# ESP32

## AT 指令集与使用示例



版本 1.2

乐鑫信息科技

版权所有 © 2018

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

# 关于本手册

本文描述 ESP32 AT 指令集功能以及使用方法，并介绍几种常见的 AT 指令使用示例。

## 发布说明

| 日期 | 版本 | 发布说明 |
| --- | --- | --- |
| 2017.11 | V1.0 | 首次发布。 |
| 2018.06 | V1.1 | 更新章节 4.2.15，5.2.8，5.2.15，6.2.5，6.2.11<br>新增章节 5.2.9，6.2.29，6.2.30，9.5.2.2<br>更新第 8 章 |
| 2018.12 | V1.2 | 更新第 1 章和章节 5.2.3 |

## 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。**

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

# 目录


1. 前言 ..... 1

1.1. 自定义 AT 指令 ..... 1

1.2. 烧录 AT 固件 ..... 1

2. 指令说明 ..... 3

3. 基础 AT 指令 ..... 4

3.1. 基础 AT 指令一览表 ..... 4

3.2. 基础 AT 指令描述 ..... 4

3.2.1. AT—测试 AT 启动 ..... 4

3.2.2. AT+RST—重启模块 ..... 4

3.2.3. AT+GMR—查询版本信息 ..... 5

3.2.4. AT+GSLP—进入 Deep-sleep 模式 ..... 5

3.2.5. ATE—开关回显功能 ..... 5

3.2.6. AT+RESTORE—恢复出厂设置 ..... 5

3.2.7. AT+UART_CUR—设置 UART 当前临时设置，不保存到 Flash ..... 6

3.2.8. AT+UART_DEF—设置 UART 配置，保存到 Flash ..... 7

3.2.9. AT+SLEEP—设置 sleep 模式 ..... 8

3.2.10. AT+SYSRAM—查询当前剩余 RAM 大小 ..... 8

3.2.11. AT+SYSFLASH—读写 Flash 用户分区 * ..... 8

3.2.12. AT+FS—文件系统操作 * ..... 9

3.2.13. AT+RFPOWER—设置 RF TX Power * ..... 10

4. Wi-Fi 功能 AT 指令 ..... 12

4.1. 基础 Wi-Fi 功能 AT 指令一览表 ..... 12

4.2. 基础 Wi-Fi 功能 AT 指令描述 ..... 13

4.2.1. AT+CWMODE—设置 Wi-Fi 模式 (Station/SoftAP/Station+SoftAP) ..... 13

4.2.2. AT+CWJAP—连接 AP ..... 14


LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

4.2.3. AT+CWLAPOPT—设置 CWLAP 指令的属性.....15
4.2.4. AT+CWLAP—扫描当前可用的 AP.....16
4.2.5. AT+CWQAP—断开与 AP 的连接.....16
4.2.6. AT+CWSAP—配置 ESP32 SoftAP 参数.....17
4.2.7. AT+CWLIF—查询连接到 ESP32 SoftAP 的 Station 信息.....18
4.2.8. AT+CWDHCP—设置 DHCP.....18
4.2.9. AT+CWDHCPS—设置 ESP32 SoftAP DHCP 分配的 IP 范围，保存到 Flash.....19
4.2.10. AT+CWAUTOCONN—上电是否自动连接 AP.....19
4.2.11. AT+CWSTARTSMART—开启 SmartConfig.....20
4.2.12. AT+CWSTOPSMART—停止 SmartConfig.....20
4.2.13. AT+WPS—设置 WPS 功能.....21
4.2.14. AT+CWHOSTNAME—设置 Station 的主机名称 *.....21
4.2.15. AT+MDNS—设置 MDNS 功能 *.....22
**5. TCP/IP 相关 AT 指令.....23**
5.1. TCP/IP 指令一览表.....23
5.2. TCP/IP 指令描述.....24
5.2.1. AT+CIPSTATUS—查询网络连接信息.....24
5.2.2. AT+CIPDOMAIN—域名解析功能.....24
5.2.3. AT+CIPDNS—自定义 DNS 服务器.....25
5.2.4. AT+CIPSTAMAC—设置 ESP32 Station 接口的 MAC 地址.....25
5.2.5. AT+CIPAPMAC—设置 ESP32 SoftAP 的 MAC 地址.....26
5.2.6. AT+CIPSTA—设置 ESP32 Station 的 IP 地址.....26
5.2.7. AT+CIPAP—设置 ESP32 SoftAP 的 IP 地址.....27
5.2.8. AT+CIPSTART—建立 TCP 连接，UDP 传输或 SSL 连接.....27
5.2.9. AT+CIPSSLCCONF—配置 SSL Client *.....29
5.2.10. AT+CIPSEND—发送数据.....30
5.2.11. AT+CIPSENDEX—发送数据.....32
5.2.12. AT+CIPCLOSE—关闭 TCP/UDP/SSL 传输.....32

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

5.2.13. AT+CIFSR—查询本地 IP 地址 ....33
5.2.14. AT+CIPMUX—设置多连接 ....33
5.2.15. AT+CIPSERVER—建立/关闭 TCP 或 SSL 服务器 * ....34
5.2.16. AT+CIPSERVERMAXCONN—设置服务器允许建立的最大连接数 * ....34
5.2.17. AT+CIPMODE—设置传输模式 ....35
5.2.18. AT+SAVETRANSLINK—保存透传到 Flash ....35
5.2.19. AT+CIPSTO—设置 TCP 服务器超时时间 ....37
5.2.20. AT+CIPSNTPCFG—设置时域和 SNTP 服务器 ....37
5.2.21. AT+CIPSNTPTIME—查询 SNTP 时间 ....38
5.2.22. AT+CIUPDATE—通过 Wi-Fi 升级软件 ....38
5.2.23. AT+CIPDINFO—接收网络数据时是否提示对端 IP 和端口 ....39
5.2.24. +IPD—接收网络数据 ....39
5.2.25. AT+PING—Ping 功能 ....40
**6. BLE 相关 AT 指令 ....41**
6.1. BLE 指令一览表 ....41
6.2. BLE 指令描述 ....42
6.2.1. AT+BLEINIT—BLE 初始化 ....42
6.2.2. AT+BLEADDR—设置 BLE 设备地址 ....43
6.2.3. AT+BLENAME—设置 BLE 设备名称 ....43
6.2.4. AT+BLESCANPARAM—设置 BLE 扫描参数 ....44
6.2.5. AT+BLESCAN—使能 BLE 扫描 ....45
6.2.6. AT+BLESCANRSPDATA—设置 BLE 扫描响应 ....45
6.2.7. AT+BLEADVPARAM—设置广播参数 ....46
6.2.8. AT+BLEADVDATA—设置 BLE 广播数据 ....47
6.2.9. AT+BLEADVSTART—开始 BLE 广播 ....47
6.2.10. AT+BLEADVSTOP—结束 BLE 广播 ....47
6.2.11. AT+BLECONN—建立 BLE 连接 ....48
6.2.12. AT+BLECONNPARAM—更新 BLE 连接参数 ....48

6.2.13. AT+BLEDISCONN—断开 BLE 连接 ....49
6.2.14. AT+BLEDATALEN—设置 BLE 数据包长度 ....49
6.2.15. AT+BLECFGMTU—设置 GATT MTU 的长度....50
6.2.16. AT+BLEGATTSSRVCRE—GATTS 创建服务 ....50
6.2.17. AT+BLEGATTSSRVSTART—GATTS 开启服务....51
6.2.18. AT+BLEGATTSSRVSTOP—GATTS 停止服务 ....51
6.2.19. AT+BLEGATTSSRV—GATTS 发现服务....51
6.2.20. AT+BLEGATTSCHAR—GATTS 发现服务特征 ....52
6.2.21. AT+BLEGATTSNTFY—GATTS 通知服务特征值....53
6.2.22. AT+BLEGATTSIND—GATTS 指示服务特征值....53
6.2.23. AT+BLEGATTSSETATTR—GATTS 设置服务特征值....54
6.2.24. AT+BLEGATTCPRIMSRV—GATTC 发现基本服务 ....55
6.2.25. AT+BLEGATTCINCLSRV—GATTC 发现包含服务....55
6.2.26. AT+BLEGATTCCHAR—GATTC 发现服务特征....56
6.2.27. AT+BLEGATTCRD—GATTC 读取服务特征值 ....57
6.2.28. AT+BLEGATTCWR—GATTC 写服务特征值....58
6.2.29. AT+BLESPPCFG—配置 BLE 透传模式....58
6.2.30. AT+BLESPP—开启 BLE 透传模式....60
6.2.31. AT+BLESECPARAM—设置加密参数 ....61
6.2.32. AT+BLEENC—发起加密请求 ....62
6.2.33. AT+BLEENCRSP—回复加密请求 ....63
6.2.34. AT+BLEKEYREPLY—回复加密密钥 ....63
6.2.35. AT+BLECONFREPLY—回复确认结果 ....63
6.2.36. AT+BLEENCDEV—查询已绑定的设备 ....64
6.2.37. AT+BLEENCCLEAR—解除绑定 ....64
**7. 设置保存在 NVS 区域 AT 指令列表 ....65**
**8. 提示消息列表 ....66**

9. AT 指令使用示例 ..... 67
9.1. 单连接 TCP 客户端 ..... 67
9.2. UDP 传输 ..... 68
9.2.1. 固定远端的 UDP 通信 ..... 68
9.2.2. 远端可变的 UDP 通信 ..... 69
9.3. Wi-Fi 透传 ..... 70
9.3.1. TCP 客户端单连接透传 ..... 70
9.3.2. UDP 透传 ..... 72
9.4. 多连接 TCP 服务器 ..... 73
9.5. BLE AT 指令应用 ..... 75
9.5.1. 基于 BLE 广播的应用—iBeacon ..... 75
9.5.2. 基于 BLE 连接的应用 ..... 77
10.OTA 功能 ..... 90
11.问题反馈 ..... 95

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

# 1. 前言

本文描述 ESP32 AT 指令集功能以及使用方法。

指令集主要分为：基础 AT 指令、Wi-Fi 功能 AT 指令、TCP/IP 工具箱 AT 指令等。

目前，标注 * 号的部分 AT 指令尚未经过严格测试，为抢先试用版本。

> 📖 **说明：**
>
> *ESP32 AT* 指令集相关代码链接：*https://github.com/espressif/esp32-at*。

## 1.1. 自定义 AT 指令

自定义 AT 指令命名时，使用英文字符以及 _（下划线）等合法字符。

AT 基于 ESP-IDF 编译，***esp-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_core.a*** 的形式提供，将包含在编译生成的 AT BIN 固件中。

结构体 `at_cmd_struct` 用于定义一条指令的四种类型，用户可以参考 ***/esp32-at/main/interface/uart/at_uart_task.c*** 中提供的示例，实现自定义 AT 指令。

## 1.2. 烧录 AT 固件

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=13*。

烧录地址见 *ESP32_AT_BIN* 文件夹中的 ***download.config*** 文件。

AT 固件中的部分 bin 文件用于特定的功能，具体如下：

- ***at_customize.bin*** 包含了用户分区的详细信息，如烧录 ***ble_data.bin***、***factory_param_XXX.bin*** 和 SSL 证书的分区信息。如有需要，可通过指令 ***AT+FS***、***AT+SYSFLASH*** 操作用户分区。
- ***factory_param_XXX.bin*** 用于适配不同的 ESP 官方模组；如果客户自己设计模组，可以参考文档 *esp32-at/docs/ESP32_AT_Factory_Parameter_Bin.md* 配置，编译时将自动生成；烧录时，将 ***download.config*** 里的 ***customized_partitions/factory_param.bin*** 文件，替换为对应的 ***customized_partitions/factory_param_XXX.bin***。

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| 模组 | UART 管脚（TX, RX, CTS, RTS） | 对应 bin 文件 |
| --- | --- | --- |
| ESP32-WROOM-32 系列模组（默认值） | GPIO17, GPIO16, GPIO15, GPIO14 | customized_partitions/factory_param_WROOM-32.bin |
| ESP32-WROVER 系列模组 | GPIO22, GPIO19, GPIO15, GPIO14 | customized_partitions/factory_param_WROVER-32.bin |
| ESP32-PICO 系列模组 | GPIO22, GPIO19, GPIO15, GPIO14 | customized_partitions/factory_param_PICO-D4.bin |
| ESP32-SOLO 系列模组 | GPIO17, GPIO16, GPIO15, GPIO14 | customized_partitions/factory_param_SOLO-1.bin |

📖 **说明：**

*UART CTS* 和 *RTS* 是可选的连接管脚，并非必须的配置。

- ***ble_data.bin*** 用于 ESP32 作为 BLE server 时提供 BLE services；
- ***server_cert.bin***，***server_key.bin***，***server_ca.bin*** 用于提供 SSL 认证证书；

如果上述某个功能无需使用，则对应 bin 文件无需烧录。

如果以上功能均需要使用，即全部 bin 文件都需要烧录。那么，在 ESP Flash Download Tool 上提供了 CombineBin 按键，可以将多个 bin 合成 1 个 bin 文件用于烧录。

请注意，在 CombineBin 时，各个 bin 文件对应的地址，以及 flash 配置等信息要求配置正确。

📖 **说明：**

- 如果 *ESP32 AT* 启动失败，并且提示 *"ota data partition invalid"* 打印信息，请将 *blank.bin* 烧录到 *esp32-at/partitions_at.csv* 中 *"otadata"* 对应的地址，进行初始化。
- 用户可以自行定义，使用其他 *UART* 进行 *AT* 通信，例如，假设改为使用 *UART0* 通信，配置步骤如下：
  - *make menuconfig -> component config -> AT -> AT UART settings* 修改为 *UART 0*
  - 调试信息默认会从 *UART 0* 打印，用户可在 *menuconfig* 中关闭打印，具体步骤如下：*make menuconfig --> Component config --> ESP32-specific --> UART for console output*
- *ESP32_AT_Bin/factory* 中是针对 *ESP* 官方模组 *CombineBin* 合成的 *ESP AT* 量产固件。

如果自行编译 AT 源代码，可使用命令 `make print_flash_cmd` 查询烧录地址，具体步骤如下：

- `rm sdkconfig`：删除旧的配置文件
- `make defconfig`：设置为最新的默认配置文件
- `make print_flash_cmd`：查询烧录地址

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

# 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，可以不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`
- *AT* 指令默认波特率为 *115200*。
- *AT* 指令不区分大小写。
- *AT* 指令以回车换行符 *(CR-LF)* 结尾。请注意设置串口工具为 *"*新行模式*"*。
- *AT* 指令返回的错误码定义，位于 *esp32-at/components/at/include/esp_at.h*。

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

# 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+SYSRAM | 查询当前剩余 RAM 大小 |
| AT+SYSFLASH | 读写 flash 用户分区 |
| AT+FS | 文件系统操作 |
| AT+RFPOWER | 设置 RF TX Power |

## 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 | - |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | |
|---|---|
| 注意 | 本指令全大写（即 AT+RST）时，可用于强制重启。<br>当系统处于 busy 状态时，如果使用指令 AT+RST，系统将提示 “will force to restart!!!”，并强制重启。 |

### 3.2.3. AT+GMR—查询版本信息

| | |
|---|---|
| 执行指令 | AT+GMR |
| 响应 | <AT version info><br><SDK version info><br><compile time><br><br>OK |
| 参数说明 | • <AT version info>：AT 版本信息<br>• <SDK version info>：SDK 版本信息<br>• <compile time>：编译生成时间 |

### 3.2.4. AT+GSLP—进入 Deep-sleep 模式

| | |
|---|---|
| 设置指令 | AT+GSLP=<time> |
| 响应 | <time><br>OK |
| 参数说明 | <time>：设置 ESP32 的睡眠时长，单位：毫秒。ESP32 会在休眠设定时长后自动唤醒。 |

### 3.2.5. ATE—开关回显功能

| | |
|---|---|
| 执行指令 | ATE |
| 响应 | OK |
| 参数说明 | • ATE0：关闭回显<br>• ATE1：开启回显 |

### 3.2.6. AT+RESTORE—恢复出厂设置

| | |
|---|---|
| 执行指令 | AT+RESTORE |
| 响应 | OK |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | |
|---|---|
| **参数说明** | 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。<br>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：<br>`AT+UART_CUR?` | 设置指令：<br>`AT+UART_CUR=<baudrate>,<databits>,<stopbits>,<parity>,<flow control>` |
| **响应** | `+UART_CUR:<baudrate>,<databits>,<stopbits>,<parity>,<flow control>`<br>`OK`<br>查询返回的是 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，是正常现象。 | `OK` |
| **参数说明** | • `<baudrate>`：UART 波特率<br>• `<databits>`：数据位<br>▸ `5`：5 bit 数据位<br>▸ `6`：6 bit 数据位<br>▸ `7`：7 bit 数据位<br>▸ `8`：8 bit 数据位<br>• `<stopbits>`：停止位<br>▸ `1`：1 bit 停止位<br>▸ `2`：1.5 bit 停止位<br>▸ `3`：2 bit 停止位<br>• `<parity>`：校验位<br>▸ `0`：None<br>▸ `1`：Odd<br>▸ `2`：Even<br>• `<flow control>`：流控<br>▸ `0`：不使能流控<br>▸ `1`：使能 RTS<br>▸ `2`：使能 CTS<br>▸ `3`：同时使能 RTS 和 CTS | |
| **注意** | • 本设置**不保存**到 flash。<br>• 使用流控需要硬件支持流控：<br>▸ IO15 为 UART0 CTS<br>▸ IO14 为 UART0 RTS<br>• 波特率支持范围：80 ~ 5000000 | |
| **示例** | `AT+UART_CUR=115200,8,1,0,3` | |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

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

| | | |
|---|---|---|
| **指令** | 查询指令：<br>`AT+UART_DEF?` | 设置指令：<br>`AT+UART_DEF=<baudrate>,<databits>,<stop bits>,<parity>,<flow control>` |
| **响应** | `+UART_DEF:<baudrate>,<databits>,<stopbits>,<parity>,<flow control>`<br>`OK` | `OK` |
| **参数说明** | • `<baudrate>`：UART 波特率<br>• `<databits>`：数据位<br>  ▸ `5`：5 bit 数据位<br>  ▸ `6`：6 bit 数据位<br>  ▸ `7`：7 bit 数据位<br>  ▸ `8`：8 bit 数据位<br>• `<stopbits>`：停止位<br>  ▸ `1`：1 bit 停止位<br>  ▸ `2`：1.5 bit 停止位<br>  ▸ `3`：2 bit 停止位<br>• `<parity>`：校验位<br>  ▸ `0`：None<br>  ▸ `1`：Odd<br>  ▸ `2`：Even<br>• `<flow control>`：流控<br>  ▸ `0`：不使能流控<br>  ▸ `1`：使能 RTS<br>  ▸ `2`：使能 CTS<br>  ▸ `3`：同时使能 RTS 和 CTS | |
| **注意** | • 本设置将保存在 NVS 区，重新上电后仍生效。<br>• 使用流控需要硬件支持流控：<br>  ▸ IO15 为 UART0 CTS<br>  ▸ IO14 为 UART0 RTS<br>• 波特率支持范围：80 ~ 5000000 | |
| **示例** | `AT+UART_DEF=115200,8,1,0,3` | |

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

### 3.2.9. AT+SLEEP—设置 sleep 模式

| | |
|---|---|
| 指令 | 设置指令：<br>AT+SLEEP=<sleep mode> |
| 响应 | OK |
| 参数说明 | <sleep mode>：<br>▸ 0：禁用休眠模式<br>▸ 1：Modem-sleep 模式 |
| 示例 | AT+SLEEP=0 |

### 3.2.10. AT+SYSRAM—查询当前剩余 RAM 大小

| | |
|---|---|
| 查询指令 | AT+SYSRAM? |
| 响应 | +SYSRAM:<remaining RAM size><br>OK |
| 参数说明 | <remaining RAM size>：当前剩余 RAM 大小，单位：字节 |
| 示例 | AT+SYSRAM?<br>+SYSRAM:148408<br>OK |

### 3.2.11. AT+SYSFLASH—读写 Flash 用户分区 *

| | | |
|---|---|---|
| 指令 | 查询指令：<br>AT+SYSFLASH?<br>功能：查询 Flash 中的用户分区。 | 设置指令：<br>AT+SYSFLASH=<operation>,<partition>,<offset>,<length><br>功能：设置 Flash 中的用户分区。 |
| 响应 | +SYSFLASH:<partition>,<type>,<subtype>,<addr>,<size><br>OK | +SYSFLASH:<length>,<data><br>OK |
| 参数说明 | <partition>：用户分区名称<br><type>：用户分区类型<br><subtype>：用户分区子类型<br><addr>：地址<br><size>：分区大小 | <operation>：<br>▸ 0：擦除分区<br>▸ 1：写分区<br>▸ 2：读分区<br><partition>：用户分区名称<br><offset>：偏移地址<br><length>：长度 |

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | |
|---|---|
| 注意 | • 使用本指令需烧录 at_customize.bin。详细信息可参考 *ESP32_Customize_Partitions*。<br>• 擦除分区的注意事项：<br>▸ 擦除分区时，设置指令可省略 <offset> 和 <length> 参数，用于完整擦除该目标分区。例如，指令 AT+SYSFLASH=0,"ble_data" 可擦除整个 “ble_data” 区域。<br>▸ 如果擦除分区时，不省略 <offset> 和 <length> 参数，则这两个参数值要求是 4KB 的整数倍。<br>• 分区的定义可以参考 *ESP-IDF Partition Tables* 的介绍。 |
| 示例 | // read 100 bytes from the "ble_data" partition offset 0.<br>AT+SYSFLASH=2,"ble_data",0,100<br>// write 10 bytes to the "ble_data" partition offset 100.<br>AT+SYSFLASH=1,"ble_data",100,10<br>// erase 8192 bytes from the "ble_data" partition offset 4096.<br>AT+SYSFLASH=0,"ble_data",4096,8192 |

### 3.2.12. AT+FS—文件系统操作 *

| | |
|---|---|
| 指令 | 设置指令：<br>AT+FS=<type>,<operation>,<filename>,<offset>,<length> |
| 响应 | OK |
| 参数说明 | <type>：目前仅支持 FATFS<br>▸ 0：FATFS<br><operation>：<br>▸ 0：删除文件<br>▸ 1：写文件<br>▸ 2：读文件<br>▸ 3：查询文件大小<br>▸ 4：查询路径下文件，目前仅支持根目录<br><offset>：偏移地址，仅针对读写操作设置<br><length>：长度，仅针对读写操作设置 |
| 注意 | • 本功能默认关闭，需要用户自行通过 make menuconfig 配置后，重新编译使能。<br>• 使用本指令需烧录 at_customize.bin。详细信息可参考 *ESP32_Customize_Partitions*。 |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | |
|---|---|
| 示例 | // delete a file.<br>`AT+FS=0,0,"filename"`<br>// write 10 bytes to offset 100 of a file.<br>`AT+FS=0,1,"filename",100,10`<br>// read 100 bytes from offset 0 of a file.<br>`AT+FS=0,2,"filename",0,100`<br>// list all files in the root directory.<br>`AT+FS=0,4,"."` |

### 3.2.13. AT+RFPOWER—设置 RF TX Power *

| | |
|---|---|
| 指令 | 设置指令：<br>`AT+RFPOWER=<wifi_power>[,<ble_adv_power>,<ble_scan_power>,<ble_conn_power>]` |
| 响应 | OK |
| 参数说明 | `<wifi_power>`：取值范围 [0, 11]<br>▸ `0`：level 0，参考 phy_init_data.bin 的第 44 字节，默认为 19.5 dBm<br>▸ `1`：level 1，参考 phy_init_data.bin 的第 45 字节，默认为 19 dBm<br>▸ `2`：level 2，参考 phy_init_data.bin 的第 46 字节，默认为 18.5 dBm<br>▸ `3`：level 3，参考 phy_init_data.bin 的第 47 字节，默认为 17 dBm<br>▸ `4`：level 4，参考 phy_init_data.bin 的第 48 字节，默认为 15 dBm<br>▸ `5`：level 5，参考 phy_init_data.bin 的第 49 字节，默认为 13 dBm<br>▸ `6`：level 5 - 2 dBm。例如，当 level 5 为 13 dBm 时，设置为 6 表示 11 dBm<br>▸ `7`：level 5 - 4.5 dBm<br>▸ `8`：level 5 - 6 dBm<br>▸ `9`：level 5 - 8 dBm<br>▸ `10`：level 5 - 11 dBm<br>▸ `11`：level 5 - 14 dBm<br>`<ble_adv_power>`：BLE 广播的 RF TX Power，取值范围 [0, 7]<br>▸ `0`：7dBm<br>▸ `1`：4dBm<br>▸ `2`：1dBm<br>▸ `3`：-2 dBm<br>▸ `4`：-5 dBm<br>▸ `5`：-8 dBm<br>▸ `6`：-11 dBm<br>▸ `7`：-14 dBm<br>`<ble_scan_power>`：BLE 扫描的 RF TX Power，取值范围 [0, 7]，同 `<ble_adv_power>`<br>`<ble_conn_power>`：BLE 连接的 RF TX Power，取值范围 [0, 7]，同 `<ble_adv_power>` |
| 注意 | • 本指令的设置并不精确，实际值可能与设置值存在差异。<br>• BLE RF TX Power 的三个参数，可以同时缺省，或者同时设置。 |

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | |
|---|---|
| 示例 | AT+RFPOWER=0<br>或者<br>AT+RFPOWER=0,0,0,0 |

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

# 4. Wi-Fi 功能 AT 指令

## 4.1. 基础 Wi-Fi 功能 AT 指令一览表

| 指令 | 说明 |
|---|---|
| AT+CWMODE | 设置 Wi-Fi 模式 (STA/AP/STA+AP) |
| AT+CWJAP | 连接 AP |
| 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 | 设置 ESP32 SoftAP 配置 |
| AT+CWLIF | 获取连接到 ESP32 SoftAP 的 Station 的信息 |
| AT+CWDHCP | 设置 DHCP |
| AT+CWDHCPS | 设置 ESP32 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+CWSTARTSMART | 开始 SmartConfig |
| AT+CWSTOPSMART | 停止 SmartConfig |
| AT+WPS | 设置 WPS 功能 |
| AT+CWHOSTNAME | 设置 ESP32 station 主机名称 |
| AT+MDNS | MDNS 功能 |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

## 4.2. 基础 Wi-Fi 功能 AT 指令描述

### 4.2.1. AT+CWMODE—设置 Wi-Fi 模式 (Station/SoftAP/Station+SoftAP)

| | | | |
|---|---|---|---|
| **指令** | 测试指令：<br>`AT+CWMODE=?` | 查询指令：<br>`AT+CWMODE?`<br>功能：查询 ESP32 当前 Wi-Fi 模式。 | 设置指令：<br>`AT+CWMODE=<mode>`<br>功能：设置 ESP32 当前 Wi-Fi 模式。 |
| **响应** | `+CWMODE:<mode>` 取值列表<br>`OK` | `+CWMODE:<mode>`<br>`OK` | `OK` |
| **参数说明** | `<mode>`：<br>▸ `0`：无 Wi-Fi 模式，并且关闭 Wi-Fi RF *<br>▸ `1`：Station 模式<br>▸ `2`：SoftAP 模式<br>▸ `3`：SoftAP+Station 模式 | | |
| **注意** | 本设置将保存在 NVS 区。 | | |
| **示例** | `AT+CWMODE=3` | | |

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

### 4.2.2. AT+CWJAP—连接 AP

| | | |
|---|---|---|
| **指令** | 查询指令：<br>`AT+CWJAP?`<br>功能：查询 ESP32 Station 已连接的 AP 信息。 | 设置指令：<br>`AT+CWJAP=<ssid>,<pwd>[,<bssid>]`<br>功能：设置 ESP32 Station 需连接的 AP。 |
| **响应** | `+CWJAP:<ssid>,<bssid>,<channel>,<rssi>`<br>`OK` | `OK`<br>或者<br>`+CWJAP:<error code>`<br>`ERROR` |
| **参数说明** | • `<ssid>`：字符串参数，AP 的 SSID<br>• `<bssid>`：AP 的 MAC 地址<br>• `<channel>`：信道号<br>• `<rssi>`：信号强度 | • `<ssid>`：目标 AP 的 SSID<br>• `<pwd>`：密码最长 64 字节 ASCII<br>• `[<bssid>]`：目标 AP 的 MAC 地址，一般用于有多个 SSID 相同的 AP 的情况<br>• `<error code>`：（仅供参考，并不可靠）<br>▸ 1：连接超时<br>▸ 2：密码错误<br>▸ 3：找不到目标 AP<br>▸ 4：连接失败<br>▸ 其他值：未知错误<br>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。 |
| **提示信息** | // If ESP32 station connects to an AP, it will prompt messages:<br>WIFI CONNECTED<br>WIFI GOT IP<br>// If the WiFi connection ends, it will prompt messages:<br>WIFI DISCONNECT | |
| **注意** | 本设置将保存在 NVS 区 | |
| **示例** | `AT+CWJAP="abc","0123456789"`<br>例如，目标 AP 的 SSID 为 `"ab\,c"`，password 为 `"0123456789"\"`，则指令如下：<br>`AT+CWJAP="ab\\\,c","0123456789\"\\"`<br>如果有多个 AP 的 SSID 均为 `"abc"`，可通过 BSSID 确定目标 AP：<br>`AT+CWJAP="abc","0123456789","ca:d7:19:d8:a6:44"` | |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

### 4.2.3. 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** | • <sort_enable>：指令 AT+CWLAP 的扫描结果是否按照信号强度 RSSI 值排序：<br>▸ 0：不排序<br>▸ 1：为根据 RSSI 排序<br>• <mask>：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：<br>▸ bit 0：设置 AT+CWLAP 的扫描结果是否显示 <ecn><br>▸ bit 1：设置 AT+CWLAP 的扫描结果是否显示 <ssid><br>▸ bit 2：设置 AT+CWLAP 的扫描结果是否显示 <rssi><br>▸ bit 3：设置 AT+CWLAP 的扫描结果是否显示 <mac><br>▸ bit 4：设置 AT+CWLAP 的扫描结果是否显示 <channel> |
| **示例** | AT+CWLAPOPT=1,31<br>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；<br>第二个参数为 31，即 0x1F，表示 <mask> 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。 |

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

### 4.2.4. AT+CWLAP—扫描当前可用的 AP

| | | |
|---|---|---|
| **指令** | 设置指令：<br>AT+CWLAP=<ssid>[,<mac>,<channel>]<br>功能：列出符合特定条件的 AP。 | 执行指令：<br>AT+CWLAP<br>功能：列出当前可用的 AP。 |
| **响应** | +CWLAP:<ecn>,<ssid>,<rssi>,<mac>,<channel><br>OK | +CWLAP:<ecn>,<ssid>,<rssi>,<mac>,<channel><br>OK |
| **参数说明** | • <ecn>：加密方式<br>▸ 0：OPEN<br>▸ 1：WEP<br>▸ 2：WPA_PSK<br>▸ 3：WPA2_PSK<br>▸ 4：WPA_WPA2_PSK<br>▸ 5：WPA2_Enterprise（目前 AT 不支持连接这种加密 AP）<br>• <ssid>：字符串参数，AP 的 SSID<br>• <rssi>：信号强度<br>• [<mac>]（选填参数）：字符串参数，AP 的 MAC 地址<br>• [<channel>]（选填参数）：信道号 | |
| **示例** | AT+CWLAP="WiFi","ca:d7:19:d8:a6:44",6<br>或者查找指定 SSID 的 AP：<br>AT+CWLAP="WiFi" | |

### 4.2.5. AT+CWQAP—断开与 AP 的连接

| | |
|---|---|
| **执行指令** | AT+CWQAP |
| **响应** | OK |
| **参数说明** | - |

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

### 4.2.6. AT+CWSAP—配置 ESP32 SoftAP 参数

| | | |
|---|---|---|
| **指令** | 查询指令：<br>`AT+CWSAP?`<br>功能：查询 ESP32 SoftAP 的配置参数。 | 设置指令：<br>`AT+CWSAP=<ssid>,<pwd>,<chl>,<ecn>[,<max conn>][,<ssid hidden>]`<br>功能：配置 ESP32 SoftAP 的参数。 |
| **响应** | `+CWSAP:<ssid>,<pwd>,<chl>,<ecn>,<max conn>,<ssid hidden>`<br>`OK` | `OK` |
| **参数说明** | • `<ssid>`：字符串参数，接入点名称<br>• `<pwd>`：字符串参数，密码长度范围：8 ~ 64 字节 ASCII<br>• `<chl>`：通道号<br>• `<ecn>`：加密方式，不支持 WEP<br>  ▸ `0`：OPEN<br>  ▸ `2`：WPA_PSK<br>  ▸ `3`：WPA2_PSK<br>  ▸ `4`：WPA_WPA2_PSK<br>• `[<max conn>]`（选填参数）：允许连入 ESP32 SoftAP 的最多 Station 数目，取值范围 [1, 10]。<br>• `[<ssid hidden>]`（选填参数）：默认为 0，开启广播 ESP32 SoftAP SSID。<br>  ▸ `0`：广播 SSID<br>  ▸ `1`：不广播 SSID | |
| **注意** | • 本设置保存到 NVS 区域。<br>• 本指令仅在 SoftAP 模式开启后有效。 | |
| **示例** | `AT+CWSAP="ESP32","1234567890",5,3` | |

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

### 4.2.7. AT+CWLIF—查询连接到 ESP32 SoftAP 的 Station 信息

| | |
|---|---|
| **执行指令** | AT+CWLIF |
| **响应** | +CWLIF:<ip addr>,<mac><br>OK |
| **参数说明** | • <ip addr>：连接到 ESP32 SoftAP 的 Station IP 地址<br>• <mac>：连接到 ESP32 SoftAP 的 Station MAC 地址 |
| **注意** | 本指令无法查询静态 IP，仅支持在 ESP32 SoftAP 和连入的 Station DHCP 均使能的情况下有效。 |

### 4.2.8. AT+CWDHCP—设置 DHCP

| | | |
|---|---|---|
| **指令** | 查询指令：<br>AT+CWDHCP? | 设置指令：<br>AT+CWDHCP=<operate>,<mode><br>功能：设置 DHCP。 |
| **响应** | +CWDHCP:<enable><br>OK | OK |
| **参数说明** | <enable>: DHCP 是否使能<br>• Bit0：<br>  ▸ 0：Station DHCP 关闭<br>  ▸ 1：Station DHCP 开启<br>• Bit1：<br>  ▸ 0：SoftAP DHCP 关闭<br>  ▸ 1：SoftAP DHCP 开启 | • <operate>：<br>  ▸ 0：关闭<br>  ▸ 1：开启<br>• <mode>：<br>  ▸ Bit0：Station DHCP<br>  ▸ Bit1：SoftAP DHCP |
| **注意** | • 本设置保存到 NVS 区域。<br>• 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：<br>  ▸ 设置使能 DHCP，则静态 IP 无效；<br>  ▸ 设置静态 IP，则 DHCP 关闭；<br>  ▸ 以最后的设置为准。 | |
| **示例** | AT+CWDHCP=1,1<br>使能 Station DHCP，如果原 DHCP mode 为 2，则现 DHCP mode 为 3。<br>AT+CWDHCP=0,2<br>禁能 SoftAP DHCP，如果原 DHCP mode 为 3，则现 DHCP mode 为 1。 | |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

### 4.2.9. AT+CWDHCPS—设置 ESP32 SoftAP DHCP 分配的 IP 范围，保存到 Flash

| | | |
|---|---|---|
| **指令** | 查询指令：<br>`AT+CWDHCPS?` | 设置指令：<br>`AT+CWDHCPS=<enable>,<lease>,<start IP>,<end IP>`<br>功能：设置 ESP32 SoftAP DHCP 服务器分配的 IP 范围。 |
| **响应** | `+CWDHCPS:<lease>,<start IP>,<end IP>`<br>`OK` | `OK` |
| **参数说明** | • `<enable>`：<br>  ▸ `0`：清除设置 IP 范围，恢复默认值，后续参数无需填写<br>  ▸ `1`：使能设置 IP 范围，后续参数必须填写<br>• `<lease>`：租约时间，单位：分钟，取值范围 [1, 2880]<br>• `<start IP>`：DHCP 服务器 IP 池的起始 IP<br>• `<end IP>`：DHCP 服务器 IP 池的结束 IP | |
| **注意** | • 本设置保存到 NVS 区域。<br>• 本指令必须在 ESP32 SoftAP 模式使能，且开启 DHCP 的情况下使用，设置的 IP 范围必须与 ESP32 SoftAP 在同一网段。 | |
| **示例** | `AT+CWDHCPS=1,3,"192.168.4.10","192.168.4.15"`<br>或者<br>`AT+CWDHCPS=0 //清除设置，恢复默认值` | |

### 4.2.10. 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** | `<enable>`：<br>▸ `0`：上电不自动连接 AP<br>▸ `1`：上电自动连接 AP<br>ESP32 Station 默认上电自动连接 AP。 |
| **注意** | 本设置保存到 NVS 区域。 |
| **示例** | `AT+CWAUTOCONN=1` |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

### 4.2.11. AT+CWSTARTSMART—开启 SmartConfig

| | | |
|---|---|---|
| **指令** | 设置指令：<br>`AT+CWSTARTSMART=<type>`<br>功能：开启某指定类型的 SmartConfig。 | 执行指令：<br>`AT+CWSTARTSMART`<br>功能：开启 ESP-TOUCH+AirKiss 兼容模式。 |
| **响应** | `OK` | `OK` |
| **参数说明** | `<type>`：<br>▸ `1`：ESP-TOUCH<br>▸ `2`：AirKiss<br>▸ `3`：ESP-TOUCH+AirKiss | none |
| **说明** | SmartConfig 连接过程中的提示信息如下：<br>smartconfig type: <type>  // AIRKISS, ESPTOUCH or UNKNOWN<br>Smart get wifi info  // got SSID and password<br>ssid:<AP's SSID><br>password:<AP's password><br>// ESP32 will try to connect to the AP<br>WIFI CONNECTED<br>WIFI GOT IP<br>smartconfig connected wifi  // if the connection failed, it will prompt "smartconfig connect fail" | |
| **注意** | • 用户可以参考 *ESP-TOUCH 用户指南* 来了解 SmartConfig 的详细介绍。<br>• 仅支持在 ESP32 单 Station 模式下调用。<br>• 消息 `Smart get wifi info` 表示 SmartConfig 成功获取到 AP 信息，之后 ESP32 尝试连接 AP，打印连接过程。<br>• 消息 `smartconfig connected wifi` 表示成功连接到 AP，此时可以调用 `AT+CWSTOPSMART` 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。 | |
| **示例** | `AT+CWMODE=1`<br>`AT+CWSTARTSMART=3` | |

### 4.2.12. 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 `AT+CWSTOPSMART` 释放快连占用的内存。 |
| **示例** | `AT+CWSTOPSMART` |

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

### 4.2.13. AT+WPS—设置 WPS 功能

| | |
|---|---|
| **设置指令** | AT+WPS=<enable> |
| **响应** | OK |
| **参数说明** | <enable>：<br>▸ 1：开启 PBC 类型的 WPS<br>▸ 0：关闭 PBC 类型的 WPS |
| **注意** | • WPS 功能必须在 ESP32 Station 使能的情况下调用。<br>• WPS 不支持 WEP 加密方式。 |
| **示例** | AT+CWMODE=1<br>AT+WPS=1 |

### 4.2.14. AT+CWHOSTNAME—设置 Station 的主机名称 *

| | | |
|---|---|---|
| **指令** | 查询指令：<br>AT+CWHOSTNAME?<br>功能：查询 ESP32 Station 的主机名称。 | 设置指令：<br>AT+CWHOSTNAME=<hostname><br>功能：设置 ESP32 Station 的主机名称。 |
| **响应** | +CWHOSTNAME:<host name><br>OK<br>如果未使能 ESP32 Station 模式，则返回<br>+CWHOSTNAME:<null><br>OK | 如果成功，返回<br>OK<br>如果未使能 ESP32 station 模式，则提示<br>ERROR |
| **参数说明** | <hostname>：主机名称，最长支持 32 字节 | |
| **注意** | • 本设置不保存到 Flash，重启后将恢复默认值。<br>• ESP32 Station 默认的主机名称为 “ESP_MAC 地址低 3 个字节”。例如，+CWHOSTNAME:<ESP_A378DA>。 | |
| **示例** | AT+CWMODE=3<br>AT+CWHOSTNAME="my_test" | |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

### 4.2.15. AT+MDNS—设置 MDNS 功能 *

| | |
|---|---|
| **设置指令** | AT+MDNS=<enable>[,<hostname>,<service_name>,<port>] |
| **响应** | OK |
| **参数说明** | • <enable>：<br>  ▸ 1：开启 MDNS 功能，后续参数需要填写<br>  ▸ 0：关闭 MDNS 功能，后续参数无需填写<br>• <hostname>：MDNS 主机名称<br>• <service_name>：MDNS 服务名称，要求以下划线开始<br>• <port>：MDNS 服务端口 |
| **注意** | • <hostname> 和 <service_name> 不能包含其他特殊字符（例如 . 符号）。 |
| **示例** | AT+MDNS=1,"espressif","_iot",8080<br>或者<br>AT+MDNS=0 |

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

# 5. TCP/IP 相关 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+CIPDNS | 自定义 DNS 服务器 |
| AT+CIPSTAMAC | 设置 ESP32 Station 的 MAC 地址 |
| AT+CIPAPMAC | 设置 ESP32 SoftAP 的 MAC 地址 |
| AT+CIPSTA | 设置 ESP32 Station 的 IP 地址 |
| AT+CIPAP | 设置 ESP32 SoftAP 的 IP 地址 |
| AT+CIPSTART | 建立 TCP 连接，UDP 传输或者 SSL 连接 |
| AT+CIPSSLCCONF | 配置 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+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 TCP 服务器允许的最大连接数 |
| AT+CIPMODE | 设置透传模式 |
| AT+SAVETRANSLINK | 保存透传连接到 Flash |
| AT+CIPSTO | 设置 ESP32 作为 TCP 服务器时的超时时间 |
| 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+CIPSNTPCFG | 设置时域和 SNTP 服务器 |
| AT+CIPSNTPTIME | 查询 SNTP 时间 |
| AT+PING | Ping 功能 |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

## 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><br>+CIPSTATUS:<link ID>,<type>,<remote IP>,<remote port>,<local port>,<tetype> |
| 参数说明 | • <stat>：ESP32 Station 接口的状态<br>  ▸ 2：ESP32 Station 已连接 AP，获得 IP 地址<br>  ▸ 3：ESP32 Station 已建立 TCP 或 UDP 传输<br>  ▸ 4：ESP32 Station 断开网络连接<br>  ▸ 5：ESP32 Station 未连接 AP<br>• <link ID>：网络连接 ID (0 ~ 4)，用于多连接的情况<br>• <type>：字符串参数，"TCP" 或者 "UDP"<br>• <remote IP>：字符串，远端 IP 地址<br>• <remote port>：远端端口值<br>• <local port>：ESP32 本地端口值<br>• <tetype>：<br>  ▸ 0：ESP32 作为客户端<br>  ▸ 1：ESP32 作为服务器 |

### 5.2.2. AT+CIPDOMAIN—域名解析功能

| | |
|---|---|
| 执行指令 | AT+CIPDOMAIN=<domain name> |
| 响应 | +CIPDOMAIN:<IP address><br>OK<br>或者<br>ERROR |
| 参数说明 | <domain name>：待解析的域名 |
| 示例 | AT+CWMODE=1 // set Station mode<br>AT+CWJAP="SSID","password" // access to the internet<br>AT+CIPDOMAIN="iot.espressif.cn" // DNS function |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

### 5.2.3. AT+CIPDNS—自定义 DNS 服务器

| | | |
|---|---|---|
| **指令** | 功能：查询当前使用的 DNS 服务器<br>查询指令：<br>`AT+CIPDNS?` | 功能：设置自定义 DNS 服务器<br>设置指令：<br>`AT+CIPDNS=<enable>[,<DNS server0>,<DNS server1>]` |
| **响应** | `+CIPDNS:<DNS server0>`<br>`[+CIPDNS:<DNS server1>]`<br>`OK` | `OK` |
| **参数说明** | • `<enable>`：<br>  ‣ `0`：不使能自定义 DNS 服务器；<br>  ‣ `1`：使能自定义 DNS 服务器。<br>• `<DNS server0>`：第一个 DNS 服务器，可不填；<br>• `<DNS server1>`：第二个 DNS 服务器，可不填。 | |
| **说明** | • 本设置将保存到 flash。<br>• 对于指令 AT+CIPDNS=1（即设置使能自定义 DNS 服务器，但未填写 `<DNS server>` 参数），则默认使用 `"208.67.222.222"` 作为 DNS 服务器。<br>• 对于指令 AT+CIPDNS=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。<br>• 自定义配置 DNS 服务器时，若设置两个地址，地址需配置为不同的地址。 | |
| **示例** | `AT+CIPDNS=1,"208.67.220.220"` | |

### 5.2.4. AT+CIPSTAMAC—设置 ESP32 Station 接口的 MAC 地址

| | | |
|---|---|---|
| **指令** | 查询指令：<br>`AT+CIPSTAMAC?`<br>功能：查询 ESP32 Station 的 MAC 地址。 | 设置指令：<br>`AT+CIPSTAMAC=<mac>`<br>功能：设置 ESP32 Station 的 MAC 地址。 |
| **响应** | `+CIPSTAMAC:<mac>`<br>`OK` | `OK` |
| **参数说明** | `<mac>`：字符串参数，ESP32 Station 的 MAC 地址 | |
| **注意** | • 本设置保存到 NVS 区域。<br>• ESP32 SoftAP 和 Station 的 MAC 地址并不相同，请勿将其设置为同一 MAC 地址。<br>• ESP32 MAC 地址第一个字节的 bit 0 不能为 1，例如，MAC 地址可以为 "1a:..." 但不能为 "15:..."。<br>• FF:FF:FF:FF:FF:FF 和 00:00:00:00:00:00 为非法 MAC，无法进行设置。 | |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| 示例 | AT+CIPSTAMAC="18:fe:35:98:d3:7b" |
|---|---|

### 5.2.5. AT+CIPAPMAC—设置 ESP32 SoftAP 的 MAC 地址

| 指令 | 查询指令：<br>AT+CIPAPMAC?<br>功能：查询 ESP32 SoftAP 的 MAC 地址。 | 设置指令：<br>AT+CIPAPMAC=<mac><br>功能：设置 ESP32 SoftAP 的 MAC 地址。 |
|---|---|---|
| 响应 | +CIPAPMAC:<mac><br>OK | OK |
| 参数说明 | <mac>：字符串参数，ESP32 SoftAP 的 MAC 地址 | |
| 注意 | • 本设置保存到 NVS 区域。<br>• ESP32 SoftAP 和 Station 的 MAC 地址并不相同，请勿将其设置为同一 MAC 地址。<br>• ESP32 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:..."。<br>• FF:FF:FF:FF:FF:FF 和 00:00:00:00:00:00 为非法 MAC，无法进行设置。 | |
| 示例 | AT+CIPAPMAC="1a:fe:36:97:d5:7b" | |

### 5.2.6. AT+CIPSTA—设置 ESP32 Station 的 IP 地址

| 指令 | 查询指令：<br>AT+CIPSTA?<br>功能：查询 ESP32 Station 的 IP 地址。 | 设置指令：<br>AT+CIPSTA=<ip>[,<gateway>,<netmask>]<br>功能：设置 ESP32 Station 的 IP 地址。 |
|---|---|---|
| 响应 | +CIPSTA:<ip><br>OK | OK |
| 参数说明 | ⚠ **注意：**<br>ESP32 Station IP 需连上 AP 后，才可以查询。 | • <ip>：字符串，ESP32 Station 的 IP 地址<br>• [<gateway>]：网关<br>• [<netmask>]：子网掩码 |
| 注意 | • 本设置保存到 NVS 区域。<br>• 本设置指令与设置 DHCP 的指令（AT+CWDHCP 系列）互相影响：<br>▸ 设置静态 IP，则 DHCP 关闭；<br>▸ 设置使能 DHCP，则静态 IP 无效；<br>▸ 以最后的设置为准。 | |
| 示例 | AT+CIPSTA="192.168.6.100","192.168.6.1","255.255.255.0" | |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

### 5.2.7. AT+CIPAP—设置 ESP32 SoftAP 的 IP 地址

| | | |
|---|---|---|
| **指令** | 查询指令：<br>AT+CIPAP?<br>功能：查询 ESP32 SoftAP 的 IP 地址。 | 设置指令：<br>AT+CIPAP=<ip>[,<gateway>,<netmask>]<br>功能：设置 ESP32 SoftAP 的 IP 地址。 |
| **响应** | +CIPAP:<ip>,<gateway>,<netmask><br>OK | OK |
| **参数说明** | • <ip>：字符串，ESP32 SoftAP 的 IP 地址<br>• [<gateway>]：网关<br>• [<netmask>]：子网掩码 | |
| **注意** | • 本设置保存到 NVS 区域。<br>• 目前仅支持 C 类 IP 地址。<br>• 本设置指令与设置 DHCP 的指令（AT+CWDHCP 系列）互相影响：<br>▸ 设置静态 IP，则 DHCP 关闭；<br>▸ 设置使能 DHCP，则静态 IP 无效；<br>▸ 以最后的设置为准。 | |
| **示例** | AT+CIPAP="192.168.5.1","192.168.5.1","255.255.255.0" | |

### 5.2.8. 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：<br>AT+CIPSTART=<type>,<remote IP>,<remote port>[,<TCP keep alive>] | TCP 多连接 (AT+CIPMUX=1) 时：<br>AT+CIPSTART=<link ID>,<type>,<remote IP>,<remote port>[,<TCP keep alive>] |
| **响应** | OK | |
| **参数说明** | • <link ID>：网络连接 ID (0 ~ 4)，用于多连接的情况<br>• <type>：字符串参数，连接类型，"TCP"，"UDP" 或 "SSL"<br>• <remote IP>：字符串参数，远端 IP 地址<br>• <remote port>：远端端口号<br>• [<TCP keep alive>]：选填参数，TCP keep-alive 侦测时间，默认关闭此功能，建议自行设置开启此功能。<br>▸ 0：关闭 TCP keep-alive 功能<br>▸ 1 ~ 7200：侦测时间，单位为 1s | |

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | |
|---|---|
| **提示信息** | // TCP 传输建立后返回以下信息：<br>[<link ID>,] CONNECT<br>// TCP 传输结束后返回以下信息：<br>[<link ID>,] CLOSED |
| **注意** | 建议创建 TCP 连接时，开启 keep-alive 功能。 |
| **示例** | `AT+CIPSTART="TCP","iot.espressif.cn",8000`<br>`AT+CIPSTART="TCP","192.168.101.110",1000`<br>详细请参考下文第 9 章 ***AT* 指令使用示例**。 |

**建立 UDP 传输**

| | | |
|---|---|---|
| **设置指令** | 单连接模式 (`AT+CIPMUX=0`) 时：<br>`AT+CIPSTART=<type>,<remote IP>,<remote port>[,<UDP local port>,<UDP mode>]` | 多连接模式 (`AT+CIPMUX=1`) 时：<br>`AT+CIPSTART=<link ID>,<type>,<remote IP>,<remote port>[,<UDP local port>,<UDP mode>]` |
| **响应** | `OK` | |
| **参数说明** | • `<link ID>`：网络连接 ID (0 ~ 4)，用于多连接的情况<br>• `<type>`：字符串参数，连接类型，`"TCP"`，`"UDP"` 或 `"SSL"`<br>• `<remote IP>`：字符串参数，远端 IP 地址<br>• `<remote port>`：远端端口号<br>• `[<UDP local port>]`：选填参数，UDP 本地端口<br>• `[<UDP mode>]`：选填参数，UDP 传输的属性，若透传，则必须为 0<br>▸ `0`：收到数据后，不更改远端目标，默认值为 0<br>▸ `1`：收到数据后，改变一次远端目标<br>▸ `2`：收到数据后，改变远端目标<br>⚠ **注意：**<br>要使用 `<UDP mode>` 需要先设置 `<UDP local port>`。 | |
| **提示信息** | // UDP 传输建立后返回以下信息：<br>[<link ID>,] CONNECT<br>// UDP 传输结束后返回以下信息：<br>[<link ID>,] CLOSED | |
| **示例** | `AT+CIPSTART="UDP","192.168.101.110",1000,1002,2`<br>详细请参考第 9 章 ***AT* 指令使用示例**。 | |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

**建立 SSL 连接**

| | |
|---|---|
| **设置指令** | `AT+CIPSTART=[<link ID>,]<type>,<remote IP>,<remote port>[,<TCP keep alive>]` |
| **响应** | `OK` |
| **参数说明** | • `<link ID>`：网络连接 ID (0 ~ 4)，用于多连接的情况<br>• `<type>`：字符串参数，连接类型，`"TCP"`，`"UDP"` 或 `"SSL"`<br>• `<remote IP>`：字符串参数，远端 IP 地址<br>• `<remote port>`：远端端口号<br>• `[<TCP keep alive>]`：选填参数，TCP keep-alive 侦测时间，默认关闭此功能，建议自行设置开启此功能<br>  ▸ `0`：关闭 TCP keep-alive 功能<br>  ▸ `1 ~ 7200`：侦测时间，单位为 1s |
| **提示信息** | // SSL 传输建立后返回以下信息：<br>[<link ID>,] CONNECT<br>// SSL 传输结束后返回以下信息：<br>[<link ID>,] CLOSED |
| **注意** | • SSL 需要占用较多空间，如果空间不足，会导致系统重启。<br>• 建议创建 SSL 连接时，开启 keep-alive 功能。 |
| **示例** | `AT+CIPSTART="SSL","iot.espressif.cn",8443` |

### 5.2.9. AT+CIPSSLCCONF—配置 SSL Client *

| | |
|---|---|
| **设置指令** | 1. 单连接时：(`AT+CIPMUX=0`)<br>`AT+CIPSSLCCONF=<type>,<cert_key_ID>,<CA_ID>`<br>2. 多连接时：(`AT+CIPMUX=1`)<br>`AT+CIPSSLCCONF=<link ID>,<type>,<cert_key_ID>,<CA_ID>`<br>功能：配置 ESP32 作为 SSL client 时的认证方式。 |
| **响应** | `OK` |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | |
|---|---|
| 参数说明 | • [<link ID>]：网络连接 ID 号 (0 ~ 4)，用于多连接的情况；如果多连接时，省略此参数，则设置对所有的网络连接均生效。<br>• <type>：认证方式<br>  ▸ 0：不进行认证<br>  ▸ 1：加载 cert 和 private key，以供 server 认证<br>  ▸ 2：加载 CA，认证 server 的 cert 和 private key<br>  ▸ 3：双向认证，SSL client 和 server 互相认证对方的证书<br>• <cert_key_ID>：证书 ID，从 0 开始计数。ESP32 AT 支持多套证书，可参考 esp32-at/tools/readme.md 中的 PKI Bin 一节生成 bin 文件。<br>• <CA_ID>：CA ID，从 0 开始计数。ESP32 AT 支持多套证书，可参考 esp32-at/tools/readme.md 中的 PKI Bin 一节生成 bin 文件。 |
| 注意 | • 如需设置本指令，请在建立 SSL 连接之前调用。<br>• 本设置将保存到 flash NVS 区域。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。 |
| 示例 | AT+CIPMUX=1 // enable multiple connections<br>AT+CIPSSLCCONF=1,3,0,0 // to set the NO.1 link, loading certificates (with ID 0) for authentication. |

## 5.2.10. AT+CIPSEND—发送数据

| | | |
|---|---|---|
| 指令 | 设置指令：<br>1. 单连接时：(AT+CIPMUX=0)<br>AT+CIPSEND=<length><br>2. 多连接时：(AT+CIPMUX=1)<br>AT+CIPSEND=<link ID>,<length><br>3. 如果是 UDP 传输，可以设置远端 IP 和端口：<br>AT+CIPSEND=[<link ID>,]<length>[,<remote IP>,<remote port>]<br>功能：在普通传输模式时，设置发送数据的长度。 | 执行指令：<br>AT+CIPSEND<br>功能：在透传模式时，开始发送数据。 |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | | |
|---|---|---|
| **响应** | 发送指定长度的数据。<br>收到此命令后先换行返回 >，然后开始接收串口数据，当数据长度满 length 时发送数据，回到普通指令模式，等待下一条 AT 指令。<br>如果未建立连接或连接被断开，返回：<br>ERROR<br>如果数据发送成功，返回：<br>SEND OK<br>如果数据发送失败，返回：<br>SEND FAIL | 收到此命令后先换行返回 >。<br>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。<br>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。<br>本指令必须在开启透传模式以及单连接下使用。<br>若为 UDP 透传，指令 AT+CIPSTART 参数 <UDP mode> 必须为 0。 |
| **参数说明** | • <link ID>：网络连接 ID 号 (0 ~ 4)，用于多连接的情况<br>• <length>：数字参数，表明发送数据的长度，最大长度为 2048<br>• [<remote IP>]：UDP 传输可以设置对端 IP<br>• [<remote port>]：UDP 传输可以设置对端端口 | - |
| **示例** | 详细请参考第 9 章 ***AT* 指令使用示例**。 | |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

### 5.2.11. AT+CIPSENDEX—发送数据

| | |
|---|---|
| **指令** | 设置指令：<br>1. 单连接时：(+CIPMUX=0)<br>`AT+CIPSENDEX=<length>`<br>2. 多连接时：(+CIPMUX=1)<br>`AT+CIPSENDEX=<link ID>,<length>`<br>3. 如果是 UDP 传输，可以设置远端 IP 和端口：<br>`AT+CIPSENDEX=[<link ID>,]<length>[,<remote IP>,<remote port>]`<br>指令功能：在普通传输模式时，设置发送数据的长度。 |
| **响应** | 发送指定长度的数据。<br>收到此命令后先换行返回 >，然后开始接收串口数据，当数据长度满 length 或者遇到字符 \0 时，发送数据。<br>如果未建立连接或连接被断开，返回：<br>ERROR<br>如果数据发送成功，返回：<br>SEND OK<br>如果数据发送失败，返回：<br>SEND FAIL |
| **参数说明** | • <link ID>：网络连接 ID 号 (0 ~ 4)，用于多连接的情况<br>• <length>：数字参数，表明发送数据的长度，最大长度为 2048<br>• 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。<br>• 用户如需发送 \0，请转义为 \\0。 |

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

| | | |
|---|---|---|
| **指令** | 设置指令（用于多连接的情况）：<br>`AT+CIPCLOSE=<link ID>`<br>功能：关闭 TCP/UDP 传输。 | 执行指令（用于单连接的情况）：<br>`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。 | |
| **提示信息** | // When connection ends, it will prompt message as below<br>[<link ID>,] CLOSED | |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

### 5.2.13. AT+CIFSR—查询本地 IP 地址

| | |
|---|---|
| 执行指令 | AT+CIFSR |
| 响应 | +CIFSR:APIP,<SoftAP IP address><br>+CIFSR:APMAC,<SoftAP MAC address><br>+CIFSR:STAIP,<Station IP address><br>+CIFSR:STAMAC,<Station MAC address><br>OK |
| 参数说明 | <IP address>:<br>ESP32 SoftAP 的 IP 地址<br>ESP32 Station 的 IP 地址<br><MAC address>:<br>ESP32 SoftAP 的 MAC 地址<br>ESP32 Station 的 MAC 地址 |
| 注意 | ESP32 Station IP 需连上 AP 后，才可以查询。 |

### 5.2.14. AT+CIPMUX—设置多连接

| | | |
|---|---|---|
| 指令 | 查询指令：<br>AT+CIPMUX? | 设置指令：<br>AT+CIPMUX=<mode><br>功能：设置连接类型。 |
| 响应 | +CIPMUX:<mode><br>OK | OK |
| 参数说明 | <mode>：<br>▸ 0：单连接模式<br>▸ 1：多连接模式 | |
| 注意 | • 默认为单连接；<br>• 只有非透传模式 (AT+CIPMODE=0)，才能设置为多连接；<br>• 必须在没有连接建立的情况下，设置连接模式；<br>• 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。 | |
| 示例 | AT+CIPMUX=1 | |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

### 5.2.15. AT+CIPSERVER—建立/关闭 TCP 或 SSL 服务器 *

| | | |
|---|---|---|
| **指令** | 查询指令：<br>AT+CIPSERVER?<br>功能：查询服务器模式。 | 设置指令：<br>AT+CIPSERVER=<mode>[,<port>][,<SSL>,<SSL CA enable>]<br>功能：设置服务器。 |
| **响应** | +CIPSERVER:<mode>,<port>,<SSL>,<SSL CA enable><br>OK | OK |
| **参数说明** | <mode>：<br>▸ 0：关闭服务器<br>▸ 1：建立服务器<br>[<port>]：选填参数。端口号，默认为 333。<br>[<SSL>]: 字符串 “SSL”，用于建立 SSL 服务器。<br>[<SSL CA enable>]:<br>▸ 0：不使用 CA 认证<br>▸ 1：使用 CA 认证 | |
| **注意** | • 多连接情况下 (AT+CIPMUX=1)，才能开启服务器。<br>• 最多只允许创建一个服务器。<br>• 创建服务器后，自动建立服务器监听。<br>• 当有客户端接入，会自动占用一个连接 ID。 | |
| **提示信息** | // 建立连接后返回以下信息：<br>[<link ID>,] CONNECT<br>// 连接结束后返回以下信息：<br>[<link ID>,] CLOSED | |
| **示例** | • 建立 TCP 服务器：<br>AT+CIPMUX=1<br>AT+CIPSERVER=1,80<br>• 建立 SSL 服务器：<br>AT+CIPMUX=1<br>AT+CIPSERVER=1,443,"SSL",1 | |

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

| | | |
|---|---|---|
| **指令** | 查询指令：<br>AT+CIPSERVERMAXCONN?<br>功能：查询服务器允许建立的最大连接数。 | 设置指令：<br>AT+CIPSERVERMAXCONN=<num><br>功能：设置服务器允许建立的最大连接数。 |

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | | |
|---|---|---|
| 响应 | +CIPSERVERMAXCONN:<num><br>OK | OK |
| 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 | AT+CIPMUX=1<br>AT+CIPSERVERMAXCONN=2<br>AT+CIPSERVER=1,80 | |

### 5.2.17. AT+CIPMODE—设置传输模式

| | | |
|---|---|---|
| 指令 | 查询指令：<br>AT+CIPMODE?<br>功能：查询传输模式。 | 设置指令：<br>AT+CIPMODE=<mode><br>功能：设置传输模式。 |
| 响应 | +CIPMODE:<mode><br>OK | OK |
| 参数说明 | <mode>：<br>▸ 0：普通传输模式<br>▸ 1：透传模式，仅支持 TCP/SSL 单连接和 UDP 固定通信对端的情况 | |
| 注意 | • 本设置不保存到 Flash。<br>• 透传模式传输时，如果连接断开，ESP32 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。<br>• WiFi 透传与 BLE 功能无法共存，因此，使能透传模式之前，请关闭 BLE 功能（AT+BLEINIT=0）。 | |
| 示例 | AT+CIPMODE=1 | |

### 5.2.18. 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 |

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | |
|---|---|
| 参数说明 | • <mode>：<br>  ▸ 0：取消开机透传<br>  ▸ 1：保存开机进入透传模式<br>• <remote IP>：远端 IP 或者域名<br>• <remote port>：远端端口<br>• [<type>]（选填参数）：TCP，SSL 或者 UDP，缺省默认为 TCP<br>• [<TCP keep alive>]（选填参数）：TCP keep-alive 侦测，缺省默认关闭此功能<br>  ▸ 0：关闭 TCP keep-alive 功能<br>  ▸ 1 ~ 7200：侦测时间，单位为秒 |
| 注意 | • 本设置将透传模式及建立的 TCP 连接均保存在 NVS 区，下次上电自动建立 TCP 连接并进入透传。<br>• 只要远端 IP，端口的值符合规范，本设置就会被保存到 Flash。 |
| 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 | • <mode>：<br>  ▸ 0：取消开机透传<br>  ▸ 1：保存开机进入透传模式<br>• <remote IP>：远端 IP<br>• <remote port>：远端端口<br>• <type>：UDP，缺省默认为 TCP<br>• [<UDP local port>]（选填参数）：开机进入 UDP 传输时，使用的本地端口 |
| 注意 | • 本设置将透传模式及建立的 UDP 传输均保存在 NVS 区，下次上电自动建立 UDP 传输并进入透传。<br>• 只要远端 IP，端口的数值符合规范，本设置就会被保存到 Flash。 |
| 示例 | AT+SAVETRANSLINK=1,"192.168.6.110",1002,"UDP",1005 |

**保存透传（SSL 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 |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | |
|---|---|
| 参数说明 | • <mode>：<br>  ▸ 0：取消开机透传<br>  ▸ 1：保存开机进入透传模式<br>• <remote IP>：远端 IP 或者域名<br>• <remote port>：远端端口<br>• [<type>]（选填参数）：SSL，缺省默认为 TCP<br>• [<TCP keep alive>]（选填参数）：TCP keep-alive 侦测，缺省默认关闭此功能<br>  ▸ 0：关闭 TCP keep-alive 功能<br>  ▸ 1 ~ 7200：侦测时间，单位为秒 |
| 注意 | • 本设置将透传模式及建立的 SSL 连接均保存在 NVS 区，下次上电自动建立 SSL 连接并进入透传。<br>• 只要远端 IP，端口的值符合规范，本设置就会被保存到 Flash。 |
| 示例 | AT+SAVETRANSLINK=1,"192.168.6.110",443,"SSL" |

## 5.2.19. AT+CIPSTO—设置 TCP 服务器超时时间

| | | |
|---|---|---|
| 指令 | 查询指令：<br>AT+CIPSTO?<br>功能：查询 TCP 服务器超时时间。 | 设置指令：<br>AT+CIPSTO=<time><br>功能：设置 TCP 服务器超时时间。 |
| 响应 | +CIPSTO:<time><br>OK | OK |
| 参数说明 | <time>：TCP 服务器超时时间，取值范围 0 ~ 7200s。 | |
| 注意 | • ESP32 作为 TCP 服务器，会断开一直不通信直至超时了的 TCP 客户端连接。<br>• 如果设置 AT+CIPSTO=0，则永远不会超时，不建议这样设置。 | |
| 示例 | AT+CIPMUX=1<br>AT+CIPSERVER=1,1001<br>AT+CIPSTO=10 | |

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

| | | |
|---|---|---|
| 指令 | 查询指令：<br>AT+CIPSNTPCFG? | 设置指令：<br>AT+CIPSNTPCFG=<enable>[,<timezone>][,<SNTP server0>,<SNTP server1>,<SNTP server2>] |
| 响应 | +CIPSNTPCFG:<enable>,<timezone>,<SNTP server0>[,<SNTP server1>,<SNTP server2>]<br>OK | OK |

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | |
|---|---|
| 参数说明 | • <enable>：<br>  ‣ 0：SNTP 未使能；<br>  ‣ 1：SNTP 使能。<br>• <timezone>：时域，范围：[-11,13]；若 SNTP 使能，此参数必填；否则，无需填写；<br>• <SNTP server0>：第一个 SNTP 服务器，可不填；<br>• <SNTP server1>：第二个 SNTP 服务器，可不填；<br>• <SNTP server2>：第三个 SNTP 服务器，可不填。 |
| 说明 | 设置指令若未填写 SNTP server，则默认使用<br>"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" |

### 5.2.21. AT+CIPSNTPTIME—查询 SNTP 时间

| | |
|---|---|
| 查询指令 | AT+CIPSNTPTIME? |
| 响应 | +CIPSNTPTIME:SNTP 时间<br>OK |
| 参数说明 | - |
| 示例 | AT+CIPSNTPCFG=1,8,"cn.ntp.org.cn","ntp.sjtu.edu.cn"<br>OK<br>AT+CIPSNTPTIME?<br>+CIPSNTPTIME:Mon Dec 12 02:33:32 2016<br>OK |

### 5.2.22. AT+CIUPDATE—通过 Wi-Fi 升级软件

| | | |
|---|---|---|
| 指令 | 执行指令：<br>AT+CIUPDATE<br>功能：执行普通升级。 | 设置指令：<br>AT+CIUPDATE=<ota_mode>[,<version>]<br>功能：设置升级模式及升级版本。 |
| 响应 | +CIPUPDATE:<n><br>OK<br>// 升级步骤提示说明如下：<br><n>：<br>‣ 1：找到服务器<br>‣ 2：向服务器查询版本信息；如果指定版本升级，则无此步骤<br>‣ 3：获得软件版本信息；如果指定版本升级，则无此步骤<br>‣ 4：开始升级 | |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | | |
|---|---|---|
| **参数说明** | none | <ota_mode>：<br>▸ 0：普通升级<br>▸ 1：SSL 升级<br>[<version>]：选填参数，可指定版本进行升级，如未设置，则升级默认版本。 |
| **说明** | • 升级过程由于网络条件的好坏，有快慢差异。<br>• 升级失败会提示 ERROR，请耐心等待。 | |
| **注意** | • 若直接使用乐鑫提供的 AT BIN (***/esp-idf/bin/at)***，本指令将从 Espressif Cloud 下载 AT 固件升级。<br>• 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（***/esp-idf/example/at***）。<br>• 用户可以通过 menuconfig 配置采用 SSL 升级，详细请参考后文章节 “OTA 功能”。<br>• 建议升级 AT 固件后，调用 AT+RESTORE 恢复出厂设置，重新初始化。 | |

## 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** | AT+CIPDINFO=<mode> |
| **响应** | OK |
| **参数说明** | <mode>：<br>▸ 0：不显示对端 IP 和端口<br>▸ 1：显示对端 IP 和端口 |
| **示例** | AT+CIPDINFO=1 |

## 5.2.24. +IPD—接收网络数据

| | | |
|---|---|---|
| **指令** | 单连接时：<br>(+CIPMUX=0)+IPD,<len>[,<remote IP>,<remote port>]:<data> | 多连接时：<br>(+CIPMUX=1)+IPD,<link ID>,<len>[,<remote IP>,<remote port>]:<data> |
| **参数说明** | 此指令在普通指令模式下有效，ESP32 接收到网络数据时向串口发送 +IPD 和数据。<br>• [<remote IP>]：网络通信对端 IP，由指令 AT+CIPDINFO=1 使能显示<br>• [<remote port>]：网络通信对端端口，由指令 AT+CIPDINFO=1 使能<br>• <link ID>：收到网络连接的 ID 号<br>• <len>：数据长度<br>• <data>：收到的数据 | |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

### 5.2.25. AT+PING—Ping 功能

| | |
|---|---|
| 设置指令 | AT+PING=<IP><br>功能：ping 功能。 |
| 响应 | +PING:<time><br>OK<br>或<br>+PING:TIMEOUT<br>ERROR |
| 参数说明 | • <IP>：字符串参数，IP 地址<br>• <time>：ping 响应时间 |
| 示例 | AT+PING="192.168.1.1"<br>AT+PING="www.baidu.com" |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

# 6. BLE 相关 AT 指令

## 6.1. BLE 指令一览表

| 指令 | 描述 |
| --- | --- |
| AT+BLEINIT | BLE 初始化 |
| AT+BLEADDR | 设置 BLE 设备地址 |
| AT+BLENAME | 设置 BLE 设备名称 |
| AT+BLESCANPARAM | 设置 BLE 扫描参数 |
| AT+BLESCAN | 使能 BLE 扫描 |
| AT+BLESCANRSPDATA | 设置 BLE 扫描响应 |
| AT+BLEADVPARAM | 设置 BLE 广播参数 |
| AT+BLEADVDATA | 设置 BLE 广播数据 |
| AT+BLEADVSTART | 开始 BLE 广播 |
| AT+BLEADVSTOP | 结束 BLE 广播 |
| AT+BLECONN | 建立 BLE 连接 |
| AT+BLECONNPARAM | 更新 BLE 连接参数 |
| AT+BLEDISCONN | 断开 BLE 连接 |
| AT+BLEDATALEN | 设置 BLE 数据包长度 |
| AT+BLECFGMTU | 设置 BLE MTU 的长度 |
| AT+BLEGATTSSRVCRE | GATTS 创建服务 |
| AT+BLEGATTSSRVSTART | GATTS 开启服务 |
| AT+BLEGATTSSRVSTOP | GATTS 关闭服务 |
| AT+BLEGATTSSRV | GATTS 查询服务 |
| AT+BLEGATTSCHAR | GATTS 查询服务特征 |
| AT+BLEGATTSNTFY | GATTS 通知服务特征值 |
| AT+BLEGATTSIND | GATTS 指示服务特征值 |
| AT+BLEGATTSSETATTR | GATTS 设置服务特征值 |
| AT+BLEGATTCPRIMSRV | GATTC 发现基本服务 |
| AT+BLEGATTCINCLSRV | GATTC 发现包含服务 |
| AT+BLEGATTCCHAR | GATTC 查询服务特征 |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| 指令 | 描述 |
| --- | --- |
| AT+BLEGATTCRD | GATTC 读取服务特征值 |
| AT+BLEGATTCWR | GATTC 写服务特征值 |
| AT+BLESPPCFG | 配置 BLE SPP |
| AT+BLESPP | 开启 BLE SPP |
| AT+BLESECPARAM | 设置 BLE SMP 参数 |
| AT+BLEENC | 开始 BLE 配对 |
| AT+BLEENCRSP | 设置 BLE 配对响应 |
| AT+BLEKEYREPLY | 回复 BLE 配对密钥 |
| AT+BLECONFREPLY | 回复 BLE 配对结果 |
| AT+BLEENCDEV | 列举已配对的设备 |
| AT+BLEENCCLEAR | 解除绑定 |

⚠ **注意：**

- 下载 *BLE SPEC*（当前支持为 *Core Version 4.2*）：*https://www.bluetooth.com/specifications/adopted-specifications*
- BLE 功能与 Wi-Fi 透传无法共存，因此，开启 BLE 功能之前，请确认 Wi-Fi 透传已关闭 (AT+CIPMODE=0)。

## 6.2. BLE 指令描述

### 6.2.1. AT+BLEINIT—BLE 初始化

| | | |
| --- | --- | --- |
| **指令** | 查询指令：<br>AT+BLEINIT?<br>功能：查询 BLE 是否初始化。 | 设置指令：<br>AT+BLEINIT=<init><br>功能：设置 BLE 初始化角色。 |
| **响应** | 如果 BLE 未初始化，则查询返回：<br>+BLEINIT:0<br>OK<br>如果 BLE 已初始化，则查询返回：<br>+BLEINIT:<role><br>OK | OK |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | |
|---|---|
| 参数说明 | <init>：<br>▸ 0：注销 BLE，关闭 BLE RF *<br>▸ 1：client role<br>▸ 2：server role |
| 注意 | • 使用 BLE 相关 AT 指令前，必须先调用本条设置指令，初始化 BLE 角色。<br>• BLE 角色初始化后，不能直接切换。如需切换角色，需要调用 AT+RST 重启，或者 AT+BLEINIT=0 注销后，重新初始化设置。<br>• 若使用 ESP32 作为 server：<br>▸ 需烧录 at_customize.bin 和 service bin。详细信息可参考 ESP32_Customize_Partitions。<br>▸ 如何生成 service bin 文件，请参考 esp32-at/tools/readme.md。<br>▸ service bin 文件的烧录地址，见 esp32-at/at_customize.csv 中 ble_data 对应的地址。 |
| 示例 | AT+BLEINIT=1 |

### 6.2.2. AT+BLEADDR—设置 BLE 设备地址

| | | |
|---|---|---|
| 指令 | 查询指令：<br>AT+BLEADDR?<br>功能：查询 BLE 设备的 public address。 | 设置指令：<br>AT+BLEADDR=<addr_type>,<random_addr><br>功能：设置 BLE 设备的地址。目前仅支持设置 random address。 |
| 响应 | +BLEADDR:<BLE_public_addr><br>OK | OK |
| 参数说明 | <addr_type>：<br>▸ 0：public address<br>▸ 1：random address | |
| 注意 | • 目前仅支持查询 public address，仅支持设置 random address。<br>• random address 要求最高两个 bit 必须全 1，详细可参考 BLE spec。 | |
| 示例 | AT+BLEADDR=1,"f8:7f:24:87:1c:f7" | |

### 6.2.3. AT+BLENAME—设置 BLE 设备名称

| | | |
|---|---|---|
| 指令 | 查询指令：<br>AT+BLENAME?<br>功能：查询 BLE 设备名称。 | 设置指令：<br>AT+BLENAME=<device_name><br>功能：设置 BLE 设备名称。 |
| 响应 | +BLENAME:<device_name><br>OK | OK |
| 参数说明 | <device_name>：BLE 设备名称 | |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | |
|---|---|
| 注意 | • 默认设备名称为 “BLE_AT”。<br>• 本指令设置的设备名称，需要在建立 BLE 连接之后，对端设备才能获取到，它其实设置的是 GAP service 中 device name characteristic 的值，详情请见 BLE core v4.2 vol.3 part C 12.1。<br>• 如果是需要在扫描广播包时得到的设备名称，则需要通过 AT+BLEADVDATA 设置。 |
| 示例 | AT+BLENAME="esp_demo" |

### 6.2.4. AT+BLESCANPARAM—设置 BLE 扫描参数

| | | |
|---|---|---|
| 指令 | 查询指令：<br>AT+BLESCANPARAM?<br>功能：查询 BLE 设备的扫描参数。 | 设置指令：<br>AT+BLESCANPARAM=<scan_type>,<own_addr_type>,<br><filter_policy>,<scan_interval>,<scan_window><br>功能：设置 BLE 设备的扫描参数。 |
| 响应 | +BLESCANPARAM:<scan_type>,<own_addr_type>,<br><filter_policy>,<scan_interval>,<scan_window><br>OK | OK |
| 参数说明 | <scan_type>：扫描类型<br>▸ 0：passive scan<br>▸ 1：active scan<br><own_addr_type>：地址类型<br>▸ 0：public address<br>▸ 1：random address<br>▸ 2：RPA public address<br>▸ 3：RPA random address<br><filter_policy>：扫描过滤方式<br>▸ 0：BLE_SCAN_FILTER_ALLOW_ALL<br>▸ 1：BLE_SCAN_FILTER_ALLOW_ONLY_WLST<br>▸ 2：BLE_SCAN_FILTER_ALLOW_UND_RPA_DIR<br>▸ 3：BLE_SCAN_FILTER_ALLOW_WLIST_PRA_DIR<br><scan_interval>：扫描间隔<br><scan_window>：扫描窗口 | |
| 注意 | • 扫描窗口的值不能大于扫描间隔 | |
| 示例 | AT+BLEINIT=1<br>AT+BLESCANPARAM=0,0,0,100,50 | |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

### 6.2.5. AT+BLESCAN—使能 BLE 扫描

| | |
|---|---|
| **指令** | 设置指令：<br>AT+BLESCAN=<enable>[,<interval>]<br>功能：执行扫描。 |
| **响应** | +BLESCAN:<addr>,<rssi>,<adv_data>,<scan_rsp_data>,<addr_type><br>OK |
| **参数说明** | <enable>：<br>▸ 0：停止扫描<br>▸ 1：开始扫描<br>[<interval>]：选填参数，持续扫描时间，单位：秒<br>▸ 若设置停止扫描，<interval> 无需填写；<br>▸ 若设置开始扫描，<br>- 若 <interval> 为 0 或缺省，如 AT+BLESCAN=1,0 或 AT+BLESCAN=1 表示持续扫描；<br>- 若 <interval> 不为 0，例如 AT+BLESCAN=1,3 表示扫描 3 秒后自动结束，返回扫描结果。<br><addr>：BLE 地址<br><rssi>：BLE 信号强度<br><adv_data>：BLE 广播数据<br><scan_rsp_data>：BLE 扫描响应<br><addr_type>：BLE 地址类型<br>▸ 0: public address<br>▸ 1: random address<br>▸ 2: RPA (Resolvable Private Address) public<br>▸ 3: RPA (Resolvable Private Address) random |
| **示例** | AT+BLEINIT=1 // 初始化为 client<br>AT+BLESCAN=1 // 开始扫描<br>AT+BLESCAN=0 // 结束扫描 |

### 6.2.6. AT+BLESCANRSPDATA—设置 BLE 扫描响应

| | |
|---|---|
| **指令** | 设置指令：<br>AT+BLESCANRSPDATA=<scan_rsp_data><br>功能：设置 BLE 扫描响应。 |
| **响应** | OK |
| **参数说明** | <scan_rsp_data>：扫描响应。参数实际为 HEX 字串。例如，设置扫描响应为 0x11 0x22 0x33 0x44 0x55，则设置指令为：AT+BLESCANRSPDATA="1122334455" |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | |
|---|---|
| **注意** | 扫描响应支持的最大长度为 31 字节。 |
| **示例** | AT+BLEINIT=2 // 初始化为 server<br>AT+BLESCANRSPDATA="1122334455" |

### 6.2.7. AT+BLEADVPARAM—设置广播参数

| | | |
|---|---|---|
| **指令** | 查询指令：<br>AT+BLEADVPARAM?<br>功能：查询广播参数。 | 设置指令：<br>AT+BLEADVPARAM=<adv_int_min>,<adv_int_max>,<br><adv_type>,<own_addr_type>,<channel_map>[,<adv_filter_policy>,<peer_addr_type>,<peer_addr>]<br>功能：设置广播参数。 |
| **响应** | +BLEADVPARAM:<adv_int_min>,<adv_int_max>,<adv_type>,<own_addr_type>,<channel_map>,<adv_filter_policy>,<peer_addr_type>,<peer_addr><br>OK | OK |
| **参数说明** | <adv_int_min>：最小广播间隔，取值范围：0x0020 ~ 0x4000<br><adv_int_max>：最大广播间隔，取值范围：0x0020 ~ 0x4000<br><adv_type>：广播类型<br>▸ 0：ADV_TYPE_IND<br>▸ 1: ADV_TYPE_DIRECT_IND_HIGH<br>▸ 2：ADV_TYPE_SCAN_IND<br>▸ 3：ADV_TYPE_NONCONN_IND<br><own_addr_type>：BLE 地址类型<br>▸ 0：BLE_ADDR_TYPE_PUBLIC<br>▸ 1：BLE_ADDR_TYPE_RANDOM<br><channel_map>：广播信道<br>▸ 1：ADV_CHNL_37<br>▸ 2：ADV_CHNL_38<br>▸ 4：ADV_CHNL_39<br>▸ 7：ADV_CHNL_ALL<br>[<adv_filter_policy>]（选填参数）：过滤器规则<br>▸ 0：ADV_FILTER_ALLOW_SCAN_ANY_CON_ANY<br>▸ 1：ADV_FILTER_ALLOW_SCAN_WLST_CON_ANY<br>▸ 2：ADV_FILTER_ALLOW_SCAN_ANY_CON_WLST<br>▸ 3：ADV_FILTER_ALLOW_SCAN_WLST_CON_WLST<br>[<peer_addr_type>]（选填参数）：对方 BLE 地址类型<br>▸ 0：PUBLIC<br>▸ 1：RANDOM<br>[<peer_addr>]（选填参数）：对方 BLE 地址 | |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | |
|---|---|
| 注意 | <adv_filter_policy>,<peer_addr_type>,<peer_addr> 三个参数要求同时缺省，或者同时设置。 |
| 示例 | AT+BLEINIT=2 // 初始化为 server<br>AT+BLEADVPARAM=50,50,0,0,4,0,0,"12:34:45:78:66:88" |

### 6.2.8. AT+BLEADVDATA—设置 BLE 广播数据

| | |
|---|---|
| 指令 | 设置指令：<br>AT+BLEADVDATA=<adv_data><br>功能：设置 BLE 广播数据。 |
| 响应 | OK |
| 参数说明 | <adv_data>：广播数据包。参数实际为 HEX 字串。例如，设置广播数据为 0x11 0x22 0x33 0x44 0x55，则设置指令为：AT+BLEADVDATA="1122334455" |
| 注意 | 广播包最大长度为 31 字节。 |
| 示例 | AT+BLEINIT=2 // 初始化为 server<br>AT+BLEADVDATA="1122334455" |

### 6.2.9. AT+BLEADVSTART—开始 BLE 广播

| | |
|---|---|
| 指令 | 执行指令：<br>AT+BLEADVSTART<br>功能：开始 BLE 广播。 |
| 响应 | OK |
| 参数说明 | 无 |
| 注意 | • 若未设置广播参数（AT+BLEADVPARAM=<adv_parameter>），则使用默认广播参数；<br>• 若未设置广播数据（AT+BLEADVDATA=<adv_data>），则发送全 0 数据包。 |
| 示例 | AT+BLEINIT=2 // 初始化为 server<br>AT+BLEADVSTART |

### 6.2.10. AT+BLEADVSTOP—结束 BLE 广播

| | |
|---|---|
| 指令 | 执行指令：<br>AT+BLEADVSTOP<br>功能：结束 BLE 广播。 |
| 响应 | OK |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | |
|---|---|
| **参数说明** | 无 |
| **注意** | 若开始广播后，成功建立 BLE 连接，则会自动结束 BLE 广播，无需调用本指令。 |
| **示例** | AT+BLEINIT=2 // 初始化为 server<br>AT+BLEADVSTART<br>AT+BLEADVSTOP |

### 6.2.11. AT+BLECONN—建立 BLE 连接

| | | |
|---|---|---|
| **指令** | 查询指令：<br>AT+BLECONN?<br>功能：查询 BLE 连接。 | 设置指令：<br>AT+BLECONN=<conn_index>,<remote_address>[,<addr_type>]<br>功能：建立 BLE 连接。 |
| **响应** | +BLECONN:<conn_index>,<remote_address><br>OK<br>若未建立连接，则不显示 <conn_index> 和 <remote_address> | OK<br>若连接成功建立，则提示<br>+BLECONN:<conn_index>,<remote_address> |
| **参数说明** | <conn_index>：BLE 连接号，当前只支持 index 为 0 的单连接，后续将增加支持多连接<br><remote address>：对方 BLE 设备地址<br><addr_type>：选填参数，BLE 地址类型，默认是 public address 类型<br>▸ 0: public address<br>▸ 1: random address<br>▸ 2: RPA (Resolvable Private Address) public<br>▸ 3: RPA (Resolvable Private Address) random | |
| **示例** | AT+BLEINIT=1 // 初始化为 client<br>AT+BLECONN=0,"24:0a:c4:09:34:23" | |

### 6.2.12. AT+BLECONNPARAM—更新 BLE 连接参数

| | | |
|---|---|---|
| **指令** | 查询指令：<br>AT+BLECONNPARAM?<br>功能：查询 BLE 连接参数。 | 设置指令：<br>AT+BLECONNPARAM=<conn_index>,<min_interval>,<max_interval>,<latency>,<timeout><br>功能：更新 BLE 连接参数。 |
| **响应** | +BLECONNPARAM:<conn_index>,<min_interval>,<max_interval>,<cur_interval>,<latency>,<timeout><br>OK | OK //指令已接收，将尝试更新连接参数<br>如果更新失败，将提示<br>+BLECONNPARAM：<conn_index>,-1 |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | |
|---|---|
| 参数说明 | <conn_index>：BLE 连接号，当前只支持 index 为 0 的单连接<br><min_interval>：最小连接间隔，取值范围：0x0006 ~ 0x0C80<br><max_interval>：最大连接间隔，取值范围：0x0006 ~ 0x0C80<br><cur_interval>：当前连接间隔<br><latency>：时延，取值范围：0x0000 ~ 0x01F3<br><timeout>：超时，取值范围：0x000A ~ 0x0C80 |
| 注意 | 本指令要求先建立连接，并且仅支持 BLE client 更新连接参数。 |
| 示例 | AT+BLEINIT=1 // 初始化为 client<br>AT+BLECONN=0,"24:0a:c4:09:34:23" // 建立 BLE 连接<br>AT+BLECONNPARAM=0,12,14,1,500 // 更新 BLE 连接参数 |

### 6.2.13. AT+BLEDISCONN—断开 BLE 连接

| | |
|---|---|
| 指令 | 设置指令：<br>AT+BLEDISCONN=<conn_index><br>功能：断开 BLE 连接。 |
| 响应 | +BLEDISCONN:<conn_index>,<remote_address><br>OK |
| 参数说明 | <conn_index>：BLE 连接号，当前只支持 index 为 0 的单连接<br><remote_address>：对方 BLE 设备地址 |
| 示例 | AT+BLEINIT=1 // 初始化为 client<br>AT+BLECONN=0,"24:0a:c4:09:34:23" // 建立 BLE 连接<br>AT+BLEDISCONN=0 // 断开 BLE 连接 |

### 6.2.14. AT+BLEDATALEN—设置 BLE 数据包长度

| | |
|---|---|
| 指令 | 设置指令：<br>AT+BLEDATALEN=<conn_index>,<pkt_data_len><br>功能：设置 BLE 数据包长度。 |
| 响应 | OK |
| 参数说明 | <conn_index>：BLE 连接号，当前只支持 index 为 0 的单连接<br><pkt_data_len> ：数据包长度，取值范围：0x001b ~ 0x00fb |
| 注意 | 需要先建立 BLE 连接，才能设置数据包长度。 |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | |
|---|---|
| 示例 | AT+BLEINIT=1 // 初始化为 client<br>AT+BLECONN=0,"24:0a:c4:09:34:23"<br>AT+BLEDATALEN=0,30 |

## 6.2.15. AT+BLECFGMTU—设置 GATT MTU 的长度

| | | |
|---|---|---|
| 指令 | 设置指令：<br>AT+BLECFGMTU?<br>功能：查询 GATT (Generic Attribute Profile) MTU 的长度。 | 设置指令：<br>AT+BLECFGMTU=<conn_index>,<mtu_size><br>功能：设置 GATT MTU 的长度。 |
| 响应 | +BLECFGMTU:<conn_index>,<mtu_size><br>OK | OK // 指令已接收 |
| 参数说明 | <conn_index>：BLE 连接号，当前只支持 index 为 0 的单连接<br><mtu_size> ：BLE 最大传输单元的长度 | |
| 注意 | • 仅 BLE client 支持设置 GATT MTU 长度，并且需要先建立 BLE 连接，才能设置 MTU 长度。<br>• 最终实际的 MTU 长度需经过协商，设置指令返回 OK 仅表示尝试协商 MTU，因此，设置长度不一定生效，建议设置后，使用查询指令 AT+BLECFGMTU? 查询实际的 MTU 长度。 | |
| 示例 | AT+BLEINIT=1 // 初始化为 client<br>AT+BLECONN=0,"24:12:5f:9d:91:98" // 建立 BLE 连接<br>AT+BLECFGMTU=0,300 | |

## 6.2.16. AT+BLEGATTSSRVCRE—GATTS 创建服务

| | |
|---|---|
| 指令 | 执行指令：<br>AT+BLEGATTSSRVCRE<br>功能：GATTS 创建服务。 |
| 响应 | OK |
| 参数说明 | 无 |
| 注意 | • 使用 ESP32 作为 server 创建服务，要求烧录 service bin 文件到 Flash 中。<br>▸ 如何生成 service bin 文件，请参考 esp32-at/tools/readme.md。<br>▸ service bin 文件的烧录地址，见 esp32-at/at_customize.csv 中 ble_data 对应的地址。<br>• ESP32 作为 server 应该在初始化完成后，及时创建服务。BLE 连接建立后，无法创建服务。 |
| 示例 | AT+BLEINIT=2 // 初始化为 server<br>AT+BLEGATTSSRVCRE |

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

### 6.2.17. AT+BLEGATTSSRVSTART—GATTS 开启服务

| | | |
|---|---|---|
| **指令** | 执行指令：<br>AT+BLEGATTSSRVSTART<br>功能：GATTS 开启全部服务。 | 设置指令：<br>AT+BLEGATTSSRVSTART=<srv_index><br>功能：GATTS 开启某指定服务。 |
| **响应** | OK | |
| **参数说明** | 无 | <srv_index>：服务序号，从 1 起始递增。 |
| **示例** | AT+BLEINIT=2 // 初始化为 server<br>AT+BLEGATTSSRVCRE<br>AT+BLEGATTSSRVSTART | |

### 6.2.18. AT+BLEGATTSSRVSTOP—GATTS 停止服务

| | | |
|---|---|---|
| **指令** | 执行指令：<br>AT+BLEGATTSSRVSTOP<br>功能：GATTS 停止全部服务。 | 设置指令：<br>AT+BLEGATTSSRVSTOP=<srv_index><br>功能：GATTS 停止某指定服务。 |
| **响应** | OK | |
| **参数说明** | 无 | <srv_index>：服务序号，从 1 起始递增。 |
| **示例** | AT+BLEINIT=2 // 初始化为 server<br>AT+BLEGATTSSRVCRE<br>AT+BLEGATTSSRVSTART<br>AT+BLEGATTSSRVSTOP | |

### 6.2.19. AT+BLEGATTSSRV—GATTS 发现服务

| | |
|---|---|
| **指令** | 查询指令：<br>AT+BLEGATTSSRV?<br>功能：GATTS 发现服务。 |
| **响应** | +BLEGATTSSRV:<srv_index>,<start>,<srv_uuid>,<srv_type><br>OK |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | |
|---|---|
| 参数说明 | <srv_index> ： 服务序号，从 1 起始递增<br><start> ：<br>▸ 0：服务未开始<br>▸ 1：服务已开始<br><srv_uuid> ： 服务的 UUID<br><srv_type> ： 服务的类型<br>▸ 0：次要服务<br>▸ 1：首要服务 |
| 示例 | AT+BLEINIT=2 // 初始化为 server<br>AT+BLEGATTSSRVCRE<br>AT+BLEGATTSSRV? |

### 6.2.20. AT+BLEGATTSCHAR—GATTS 发现服务特征

| | |
|---|---|
| 指令 | 查询指令：<br>AT+BLEGATTSCHAR?<br>功能：GATTS 发现服务特征。 |
| 响应 | //对于服务特征信息，显示如下：<br>+BLEGATTSCHAR:"char",<srv_index>,<char_index>,<char_uuid>,<char_prop><br>//对于描述符信息，显示如下：<br>+BLEGATTSCHAR:"desc",<srv_index>,<char_index>,<desc_index><br>OK |
| 参数说明 | <srv_index> ： 服务序号，从 1 起始递增<br><char_index> ： 服务特征的序号，从 1 起始递增<br><char_uuid> ： 服务特征的 UUID<br><char_prop> ： 服务特征的属性<br><desc_index> ： 特征描述符序号<br><desc_uuid> ： 特征描述符的 UUID |
| 示例 | AT+BLEINIT=2 // 初始化为 server<br>AT+BLEGATTSSRVCRE<br>AT+BLEGATTSSRVSTART<br>AT+BLEGATTSCHAR? |

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

### 6.2.21. AT+BLEGATTSNTFY—GATTS 通知服务特征值

| | |
|---|---|
| **指令** | 设置指令：<br>AT+BLEGATTSNTFY=<conn_index>,<srv_index>,<char_index>,<length><br>功能：GATTS 通知服务特征值。 |
| **响应** | 收到此命令后先换行返回 >，然后开始接收串口数据，当数据长度满 <length> 时，执行通知操作。<br>若通知操作成功，则提示 OK |
| **参数说明** | <conn_index>：BLE 连接号，当前只支持 index 为 0 的单连接<br><srv_index> ：服务序号，由指令 AT+BLEGATTSCHAR? 查询可得<br><char_index> ：服务特征的序号，由指令 AT+BLEGATTSCHAR? 查询可得<br><length> ：数据长度 |
| **示例** | 以下为 notify 的简单示例，详细步骤可参看后文 **章节 *9.5 BLE AT 示例***。<br>AT+BLEINIT=2 // 初始化为 server<br>AT+BLEGATTSSRVCRE<br>AT+BLEGATTSSRVSTART<br>AT+BLEADVSTART // 开始广播，等待 client 连接，并配置接收 notify<br>AT+BLEGATTSCHAR? // 查询允许 notify 的特征<br>//例如，使用 3 号服务的 6 号特征通知长度为 4 的数据<br>AT+BLEGATTSNTFY=0,3,6,4<br>// 提示 > 符号后，输入 4 字节数据即可，例如 "1234" |

### 6.2.22. AT+BLEGATTSIND—GATTS 指示服务特征值

| | |
|---|---|
| **指令** | 设置指令：<br>AT+BLEGATTSIND=<conn_index>,<srv_index>,<char_index>,<length><br>功能：GATTS 指示服务特征值。 |
| **响应** | 收到此命令后先换行返回 >，然后开始接收串口数据，当数据长度满 <length> 时，执行指示操作。<br>若指示操作成功，则提示 OK |
| **参数说明** | <conn_index>：BLE 连接号，当前只支持 index 为 0 的单连接<br><srv_index> ：服务序号，由指令 AT+BLEGATTSCHAR? 查询可得<br><char_index> ：服务特征的序号，由指令 AT+BLEGATTSCHAR? 查询可得<br><length> ：数据长度 |

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | |
|---|---|
| 示例 | 以下为 indicate 的简单示例，详细步骤可参看后文 **章节 *9.5 BLE AT 示例***。<br>AT+BLEINIT=2 // 初始化为 server<br>AT+BLEGATTSSRVCRE<br>AT+BLEGATTSSRVSTART<br>AT+BLEADVSTART // 开始广播，等待 client 连接， client 端连接后，应该设置为接收 indication<br>AT+BLEGATTSCHAR? // 查询允许 indicate 的特征<br>//例如，使用 3 号服务的 7 号特征指示长度为 4 的数据<br>AT+BLEGATTSIND=0,3,7,4<br>// 提示 > 符号后，输入 4 字节数据即可，例如 "1234" |

### 6.2.23. AT+BLEGATTSSETATTR—GATTS 设置服务特征值

| | |
|---|---|
| 指令 | 设置指令：<br>AT+BLEGATTSSETATTR=<srv_index>,<char_index>[,<desc_index>],<length><br>功能：GATTS 设置服务特征（描述符）值。 |
| 响应 | 收到此命令后先换行返回 >，然后开始接收串口数据，当数据长度满 <length> 时，执行设置操作。<br>若设置操作成功，则提示 OK |
| 参数说明 | <srv_index> ： 服务发现结果序号，由 AT+BLEGATTSCHAR? 查询结果中获得<br><char_index> ： 服务特征的序号，由 AT+BLEGATTSCHAR? 查询结果中获得<br>[<desc_index>]（选填参数）： 特征描述符序号。若填写，则设置描述符的值；若未填写，则设置特征值。<br><length> ： 数据长度 |
| 注意 | <length> 不能超过该特征 (描述符) 支持的最大长度。例如，该服务特征值为 ”0x30 0x31“，最大长度为 2，如果设置 <lengh> 为 3 超过最大长度，则会报错。 |
| 示例 | AT+BLEINIT=2 // 初始化为 server<br>AT+BLEGATTSSRVCRE<br>AT+BLEGATTSSRVSTART<br>AT+BLEGATTSCHAR?<br>//例如，向 1号服务的 1 号特征写入长度为 4 的数据<br>AT+BLEGATTSSETATTR=1,1,,4<br>// 提示 > 符号后，输入 4 字节数据即可，例如 "1234" |

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

### 6.2.24. AT+BLEGATTCPRIMSRV—GATTC 发现基本服务

| | |
|---|---|
| **指令** | 设置指令：<br>AT+BLEGATTCPRIMSRV=<conn_index><br>功能：GATTC 发现基本服务。 |
| **响应** | +BLEGATTCPRIMSRV:<conn_index>,<srv_index>,<srv_uuid>,<srv_type><br>OK |
| **参数说明** | <conn_index>：BLE 连接号，当前只支持 index 为 0 的单连接<br><srv_index> ：服务发现结果序号，从 1 起始递增<br><srv_uuid> ：服务的 UUID<br><srv_type> ：服务的类型<br>▸ 0：次要服务<br>▸ 1：首要服务 |
| **注意** | 使用本指令，需要先建立 BLE 连接。 |
| **示例** | AT+BLEINIT=1 // 初始化为 client<br>AT+BLECONN=0,"24:12:5f:9d:91:98" // 建立 BLE 连接<br>AT+BLEGATTCPRIMSRV=0 |

### 6.2.25. AT+BLEGATTCINCLSRV—GATTC 发现包含服务

| | |
|---|---|
| **指令** | 设置指令：<br>AT+BLEGATTCINCLSRV=<conn_index>,<srv_index><br>功能：GATTC 发现包含服务。 |
| **响应** | +BLEGATTCINCLSRV:<conn_index>,<srv_index>,<srv_uuid>,<srv_type>,<included_srv_uuid>,<included_srv_type><br>OK |

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | |
|---|---|
| 参数说明 | <conn_index>：BLE 连接号，当前只支持 index 为 0 的单连接<br><srv_index> ： 服务发现结果序号，由 AT+BLEGATTCPRIMSRV=<conn_index> 查询结果中获得<br><srv_uuid> ： 服务的 UUID<br><srv_type> ： 服务的类型<br>▸ 0：次要服务<br>▸ 1：首要服务<br><included_srv_uuid> ： 包含服务的 UUID<br><included_srv_type> ： 包含服务的类型<br>▸ 0：次要服务<br>▸ 1：首要服务 |
| 注意 | 使用本指令，需要先建立 BLE 连接。 |
| 示例 | AT+BLEINIT=1 // 初始化为 client<br>AT+BLECONN=0,"24:12:5f:9d:91:98" // 建立 BLE 连接<br>AT+BLEGATTCPRIMSRV=0<br>AT+BLEGATTCINCLSRV=0,1 //根据前一条指令的查询结果，指定 index 查询 |

### 6.2.26. AT+BLEGATTCCHAR—GATTC 发现服务特征

| | |
|---|---|
| 指令 | 设置指令：<br>AT+BLEGATTCCHAR=<conn_index>,<srv_index><br>功能：GATTC 发现服务特征。 |
| 响应 | //对于服务特征信息，显示如下：<br>+BLEGATTCCHAR:"char",<conn_index>,<srv_index>,<char_index>,<char_uuid>,<char_prop><br>//对于描述符信息，显示如下：<br>+BLEGATTCCHAR:"desc",<conn_index>,<br><srv_index>,<char_index>,<desc_index>,<desc_uuid><br>OK |
| 参数说明 | <conn_index>：BLE 连接号，当前只支持 index 为 0 的单连接<br><srv_index> ： 服务发现结果序号，由 AT+BLEGATTCPRIMSRV=<conn_index> 查询结果中获得<br><char_index> ： 服务特征的序号，从 1 起始递增<br><char_uuid> ： 服务特征的 UUID<br><char_prop> ： 服务特征的属性<br><desc_index> ： 特征描述符序号<br><desc_uuid> ： 特征描述符的 UUID |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | |
|---|---|
| **注意** | 使用本指令，需要先建立 BLE 连接。 |
| **示例** | AT+BLEINIT=1 // 初始化为 client<br>AT+BLECONN=0,"24:12:5f:9d:91:98" // 建立 BLE 连接<br>AT+BLEGATTCPRIMSRV=0<br>AT+BLEGATTCCHAR=0,1 //根据前一条指令的查询结果，指定 index 查询 |

### 6.2.27. AT+BLEGATTCRD—GATTC 读取服务特征值

| | |
|---|---|
| **指令** | 设置指令：<br>AT+BLEGATTCRD=<conn_index>,<srv_index>,<char_index>[,<desc_index>]<br>功能：GATTC 读取服务特征（描述符）值。 |
| **响应** | +BLEGATTCRD:<conn_index>,<len>,<value><br>OK |
| **参数说明** | <conn_index>：BLE 连接号，当前只支持 index 为 0 的单连接<br><srv_index> ： 服务发现结果序号，由 AT+BLEGATTCPRIMSRV=<conn_index> 查询结果中获得<br><char_index> ： 服务特征的序号，由 AT+BLEGATTCCHAR=<conn_index>,<srv_index> 查询结果中获得<br>[<desc_index>]（选填参数）：特征描述符序号。若不设置，读取特征值；若设置，读取描述符的值。<br><len> ： 数据长度<br><value> ： HEX 字串<br>▸ 若由指令 AT+BLEGATTCRD=<conn_index>,<srv_index>,<char_index> 读取服务特征的值，例如指令读取返回 "+BLEGATTCRD:0,1,30" 表示特征值长度为 1 个字节，内容为 HEX 字串 "0x30"。<br>▸ 若由指令 AT+BLEGATTCRD=<conn_index>,<srv_index>,<char_index>,<desc_index> 读取服务特征描述符的值，例如指令读取返回 “+BLEGATTCRD:0,4,30313233” 表示特征描述符的值长度为 4 个字节，内容为 HEX 字串 “0x30 0x31 0x32 0x33”。 |
| **注意** | • 使用本指令，需要先建立 BLE 连接。<br>• 如果该服务特征属性不支持读操作，则指令会报错。 |
| **示例** | AT+BLEINIT=1 // 初始化为 client<br>AT+BLECONN=0,"24:12:5f:9d:91:98" // 建立 BLE 连接<br>AT+BLEGATTCPRIMSRV=0<br>AT+BLEGATTCCHAR=0,3 //根据前一条指令的查询结果，指定 index 查询<br>AT+BLEGATTCRD=0,3,2,1 //例如，读取第 3 号服务的第 2 号特征的第 1 号描述符信息 |

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

### 6.2.28. AT+BLEGATTCWR—GATTC 写服务特征值

| | |
|---|---|
| **指令** | 设置指令：<br>AT+BLEGATTCWR=<conn_index>,<srv_index>,<char_index>[,<desc_index>],<length><br>功能：GATTC 写服务特征（描述符）值。 |
| **响应** | 收到此命令后先换行返回 >，然后开始接收串口数据，当数据长度满 <length> 时，执行写操作。<br>若写操作成功，则提示 OK |
| **参数说明** | <conn_index>：BLE 连接号，当前只支持 index 为 0 的单连接<br><srv_index> ：服务发现结果序号，由 AT+BLEGATTCPRIMSRV=<conn_index> 查询结果中获得<br><char_index> ：服务特征的序号，由 AT+BLEGATTCCHAR=<conn_index>,<srv_index> 查询结果中获得<br>[<desc_index>]（选填参数）：特征描述符序号。若不设置，则写特征值；若设置，写描述符的值。<br><length> ：数据长度 |
| **注意** | • 使用本指令，需要先建立 BLE 连接。<br>• 如果该服务特征（描述符）属性不支持写操作，则指令会报错。 |
| **示例** | AT+BLEINIT=1 // 初始化为 client<br>AT+BLECONN=0,"24:12:5f:9d:91:98" // 建立 BLE 连接<br>AT+BLEGATTCPRIMSRV=0<br>AT+BLEGATTCCHAR=0,3 //根据前一条指令的查询结果，指定 index 查询<br>// 例如，向第 3 号服务的第 4 号特征，写入长度为 6 的数据<br>AT+BLEGATTCWR=0,3,4,,6<br>// 提示 > 后，通过串口输入数据 "123456" 即可 |

### 6.2.29. AT+BLESPPCFG—配置 BLE 透传模式

| | |
|---|---|
| **设置指令** | AT+BLESPPCFG=<cfg_enable>,<tx_srv_index>,<tx_char_index>,<rx_srv_index>,<rx_srv_index><br>功能：配置 BLE 透传模式。分别选择两个 characteristic，一个用于发送数据，一个用于接收数据，作为数据通信的收发通道。 |
| **响应** | OK |

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | |
|---|---|
| **参数说明** | <cfg_enable>：<br>• 0：清除 BLE SPP 设置参数；之后的四个参数无需填写。<br>• 1：设置 BLE SPP 参数；之后的四个参数必须填写。<br><tx_srv_index>：服务序号，指定用于发送数据的 characteristic 所在的 service 的序号。<br>• 如果 ESP 作为 BLE server，由 AT+BLEGATTSSRVCRE 查询结果中获得，是本地的 service ID；<br>• 如果 ESP 作为 BLE client，由 AT+BLEGATTCPRIMSRV=<conn_index> 查询结果中获得，是远端的 service ID。<br><tx_char_index>：服务 characteristic 的序号，指定用于发送数据的 characteristic 的序号。<br>• 如果 ESP 作为 BLE server，由 AT+BLEGATTSCHAR 查询结果中获得，是本地的 characteristic ID；要求此 characteristic 支持 notify 或者 indicate；<br>• 如果 ESP 作为 BLE client，由 AT+BLEGATTCCHAR=<conn_index>,<srv_index> 查询结果中获得，是远端的 characteristic ID；要求此 characteristic 支持 write。<br><rx_srv_index>：服务序号，指定用于接收数据的 characteristic 所在的 service 的序号。<br>• 如果 ESP 作为 BLE server，由 AT+BLEGATTSSRVCRE 查询结果中获得，是本地的 service ID；<br>• 如果 ESP 作为 BLE client，由 AT+BLEGATTCPRIMSRV=<conn_index> 查询结果中获得，是远端的 service ID。<br><rx_char_index>：服务特征的序号，指定用于接收数据的 characteristic 的序号；<br>• 如果 ESP 作为 BLE server，由 AT+BLEGATTSCHAR 查询结果中获得，是本地的 characteristic ID；要求此 characteristic 支持 write；<br>• 如果 ESP 作为 BLE client，由 A AT+BLEGATTCCHAR=<conn_index>,<srv_index> 查询结果中获得，是远端的 characteristic ID；要求此 characteristic 支持 notify 或者 indicate。 |
| **注意** | • 本指令在初始化 BLE 之后即可配置。<br>• 本指令可以重复调用，以最后一次的配置为准。<br>• 如果 characteristic 不支持读操作，则会返回 “ERROR"。本设置不保存到 flash，在系统重启或关闭 BLE 功能之后将被清空。 |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | |
|---|---|
| 示例 | AT+BLEINIT=2 // 初始化为 server<br>AT+BLEGATTSSRVCRE<br>AT+BLEGATTSSRVSTART<br>AT+BLEGATTSCHAR?<br>AT+BLEADVSTART // 开始广播<br>// client 连入，BLE 连接建立，将提示 +BLECONN:0,<client MAC><br>// 例如，设置第 1 号服务的第 7 号 characteristic 为写通道，设置第 1 号服务的第 5 号 characteristic 为读通道<br>AT+BLESPPCFG=1,1,7,1,5<br>// client 端需要设置为侦听写通道（例如，上述第 7 号 characteristic），允许 notify 或者 indicate。 |

## 6.2.30. AT+BLESPP—开启 BLE 透传模式

| | |
|---|---|
| 执行指令 | AT+BLESPP<br>功能：开启 BLE 透传模式。 |
| 响应 | OK<br>> // 等待输入数据。<br>执行指令后换行返回 > 符号，ESP32 进入 UART-BLE 透传模式。<br>当输入单独一包 +++ 时，ESP3 2 返回普通 AT 指令模式。<br>发送 +++ 退出透传时，请至少间隔 1 秒再发下一条 AT 指令。 |
| 注意 | • 开启透传时，会检查 AT+BLESPPCFG 的配置；如果未配置、配置清空或者配置无效（例如，characteristic 的属性不符合要求），则返回 ERROR。<br>• 如果 Wi-Fi 透传已开启，则返回 ERROR。<br>• 如果 BLE 处于未连接或者多连接的状态，则返回 ERROR。<br>• 如果 BLE 正在广播（除了不可连接的广播 ADV_TYPE_NONCONN_IND），则返回 ERROR。<br>• 如果在透传模式下，BLE 连接断开，ESP 将一直尝试重建原连接。 |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | |
|---|---|
| 示例 | AT+BLEINIT=2 // 初始化为 server<br>AT+BLEGATTSSRVCRE<br>AT+BLEGATTSSRVSTART<br>AT+BLEGATTSCHAR?<br>AT+BLEADVSTART // 开始广播<br>// client 连入，BLE 连接建立，将提示 +BLECONN:0,<client MAC><br>// 例如，设置第 1 号服务的第 7 号 characteristic 为写通道，设置第 1 号服务的第 5 号 characteristic 为读通道<br>AT+BLESPPCFG=1,1,7,1,5<br>// client 端需要设置为侦听写通道（例如，上述第 7 号 characteristic），允许 notify 或者 indicate。<br>AT+BLESPP // 开启透传 |

### 6.2.31. AT+BLESECPARAM—设置加密参数

| | | |
|---|---|---|
| 指令 | 查询指令：<br>AT+BLESECPARAM?<br>功能：查询 BLE SMP 加密参数。 | 设置指令：<br>AT+BLESECPARAM=<auth_req>,<iocap>,<key_size>,<init_key>,<rsp_key><br>功能：设置 BLE SMP 加密参数。 |
| 响应 | +BLESECPARAM:<auth_req>,<iocap>,<key_size>,<init_key>,<rsp_key><br>OK | OK |

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | |
|---|---|
| 参数说明 | <auth_req>：认证要求<br>▸ 0: ESP_LE_AUTH_NO_BOND<br>▸ 1: ESP_LE_AUTH_BOND<br>▸ 2: ESP_LE_AUTH_REQ_MITM<br>▸ 4: ESP_LE_AUTH_REQ_SC_ONLY<br>▸ 5: ESP_LE_AUTH_REQ_SC_BOND<br>▸ 6: ESP_LE_AUTH_REQ_SC_MITM<br>▸ 7: ESP_LE_AUTH_REQ_SC_MITM_BOND<br><iocap>：输入输出能力<br>▸ 0: ESP_IO_CAP_OUT /*!< DisplayOnly */<br>▸ 1: ESP_IO_CAP_IO /*!< DisplayYesNo */<br>▸ 2: ESP_IO_CAP_IN /*!< KeyboardOnly */<br>▸ 3: ESP_IO_CAP_NONE /*!< NoInputNoOutput */<br>▸ 4: ESP_IO_CAP_KBDISP /*!< Keyboard display */<br><key_size>：密钥长度，取值范围: [7, 16]<br><init_key>：初始密钥<br>▸ 若 bit0 为 1，表示 ESP_BLE_ENC_KEY_MASK // to exchange the encryption key<br>▸ 若 bit1 为 1，表示 ESP_BLE_ID_KEY_MASK // to exchange the IRK key<br>▸ 若 bit2 为 1，表示 ESP_BLE_CSR_KEY_MASK // to exchange the CSRK key<br>▸ 若 bit3 为 1，表示 ESP_BLE_LINK_KEY_MASK // to exchange the link key(this key just used in the BLE & BR/EDR coexist mode)<br><rsp_key> ：响应密钥<br>▸ 若 bit0 为 1，表示 ESP_BLE_ENC_KEY_MASK // to exchange the encryption key<br>▸ 若 bit1 为 1，表示 ESP_BLE_ID_KEY_MASK // to exchange the IRK key<br>▸ 若 bit2 为 1，表示 ESP_BLE_CSR_KEY_MASK // to exchange the CSRK key<br>▸ 若 bit3 为 1，表示 ESP_BLE_LINK_KEY_MASK // to exchange the link key(this key just used in the BLE & BR/EDR coexist mode) |
| 注意 | 本指令要求在 BLE 连接建立之前调用。 |
| 示例 | AT+BLESECPARAM=1,4,16,3,3 |

### 6.2.32. AT+BLEENC—发起加密请求

| | |
|---|---|
| 指令 | 设置指令：<br>AT+BLEENC=<conn_index>,<sec_act><br>功能：发起 BLE SMP 加密请求。 |
| 响应 | OK |
| 参数说明 | <conn_index>：BLE 连接号，当前只支持 index 为 0 的单连接<br><sec_act>：<br>▸ 0: ESP_BLE_SEC_NONE<br>▸ 1: ESP_BLE_SEC_ENCRYPT<br>▸ 2: ESP_BLE_SEC_ENCRYPT_NO_MITM<br>▸ 3: ESP_BLE_SEC_ENCRYPT_MITM |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | |
|---|---|
| 示例 | AT+BLESECPARAM=1,4,16,3,3<br>AT+BLEENC=0,3 |

### 6.2.33. AT+BLEENCRSP—回复加密请求

| | |
|---|---|
| 指令 | 设置指令：<br>AT+BLEENCRSP=<conn_index>,<accept><br>功能：回复 BLE SMP 加密请求。 |
| 响应 | OK |
| 参数说明 | <conn_index>：BLE 连接号，当前只支持 index 为 0 的单连接<br><accept>：<br>▸ 0: reject<br>▸ 1: accept |
| 示例 | AT+BLEENCRSP=0,1 |

### 6.2.34. AT+BLEKEYREPLY—回复加密密钥

| | |
|---|---|
| 指令 | 设置指令：<br>AT+BLEKEYREPLY=<conn_index>,<key><br>功能：回复 BLE SMP 配对密钥。 |
| 响应 | OK |
| 参数说明 | <conn_index>：BLE 连接号，当前只支持 index 为 0 的单连接<br><key>：配对密钥 |
| 示例 | // 假设对方要求输入密钥 649784 进行配对<br>AT+BLESETKEYREPLY=0,649784 |

### 6.2.35. AT+BLECONFREPLY—回复确认结果

| | |
|---|---|
| 指令 | 设置指令：<br>AT+BLECONFREPLY=<conn_index>,<confirm><br>功能：回复配对确认结果。用于不支持回复加密密钥的情况。 |
| 响应 | OK |
| 参数说明 | <conn_index>：BLE 连接号，当前只支持 index 为 0 的单连接<br><confirm>：<br>▸ 0: No<br>▸ 1: Yes |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

| | |
|---|---|
| 示例 | // 回复对方确认配对<br>AT+BLECONFREPLY=0,1 |

## 6.2.36. AT+BLEENCDEV—查询已绑定的设备

| | |
|---|---|
| 指令 | 查询指令：<br>AT+BLEENCDEV? |
| 响应 | +BLEENCDEV:<enc_dev_index>,<mac_address><br>OK |
| 参数说明 | <enc_dev_index>：BLE 已绑定设备的连接号，当前只支持 index 为 0 的单连接<br><mac_address>：已绑定设备的 MAC 地址 |
| 示例 | AT+BLEENCDEV? |

## 6.2.37. AT+BLEENCCLEAR—解除绑定

| | | |
|---|---|---|
| 指令 | 设置指令：<br>AT+BLEENCCLEAR=<enc_dev_index><br>功能：解除某特定连接的绑定关系。 | 执行指令：<br>AT+BLEENCCLEAR<br>功能：解除所有绑定关系。 |
| 响应 | OK | |
| 参数说明 | <enc_dev_index>：BLE 已绑定设备的连接号，当前只支持 index 为 0 的单连接 | |
| 示例 | AT+BLEENCCLEAR | |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

# 7. 设置保存在 NVS 区域 AT 指令列表

| 指令 | 描述 |
| --- | --- |
| AT+UART | AT+UART=115200,8,1,0,3 |
| AT+UART_DEF | AT+UART_DEF=115200,8,1,0,3 |
| AT+CWDHCP | AT+CWDHCP=1,1 |
| AT+CIPSTAMAC | AT+CIPSTAMAC="18:fe:35:98:d3:7b" |
| AT+CIPAPMAC | AT+CIPAPMAC="1a:fe:36:97:d5:7b" |
| AT+CIPSTA | AT+CIPSTA="192.168.6.100" |
| AT+CIPAP | AT+CIPAP="192.168.5.1" |
| AT+CWDHCPS | AT+CWDHCPS=1,3,"192.168.4.10","192.168.4.15" |
| AT+SAVETRANSLINK | AT+SAVETRANSLINK=1,"192.168.6.10",1001 |
| AT+CWMODE | AT+CWMODE=3 |
| AT+CWJAP | AT+CWJAP="abc","0123456789" |
| AT+CWSAP | AT+CWSAP="ESP32","12345678",5,3 |
| AT+CWAUTOCONN | AT+CWAUTOCONN=1 |
| AT+CIPSSLCCONF | AT+CIPSSLCCONF=1,3,0,0 |

⚠ **注意：**

*NVS 参数区为 0xFA000 ~ 0x110000，88 KB。*

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

# 8. 提示消息列表

ESP32 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 | ESP station 连接到 AP |
| WIFI GOT IP | ESP station 获取到 IP 地址 |
| WIFI DISCONNECT | ESP station 的 WiFi 连接断开 |
| 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 ESP softAP |
| +DIST_STA_IP:<sta_mac>,<sta_ip> | ESP softAP 给连入的 station 分配 IP 地址 |
| +STA_DISCONNECTED:<sta_mac> | station 从 ESP softAP 断开连接 |
| +BLECONN | BLE 建立连接 |
| +BLEDISCONN | BLE 断开连接 |
| +READ | BLE 接收到读操作 |
| +WRITE | BLE 接收到写操作 |
| +NOTIFY | BLE 接收到通知操作 |
| +INDICATE | BLE 接收到指示操作 |
| +BLESECNTFYKEY | BLE SMP 密钥 |
| +BLEAUTHCMPL | BLE SMP 配对完成 |

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

# 9. AT 指令使用示例

本章介绍几种常见的 AT 指令使用示例。

## 9.1. 单连接 TCP 客户端

1. 设置 Wi-Fi 模式：

```
AT+CWMODE=3                                    // SoftAP+Station mode
```

响应：

```
OK
```

2. 连接路由：

```
AT+CWJAP="SSID","password"                     // SSID and password of router
```

响应：

```
OK
```

3. 查询设备 IP 地址：

```
AT+CIFSR
```

响应：

```
192.168.3.106                                  // device got an IP from router
```

4. PC 与 ESP32 连接同一个路由，在 PC 上使用网络调试工具，创建一个 TCP 服务器。

   - 例如，假设 TCP 服务器的 IP 为 192.168.3.116，端口为 8080。

5. ESP32 作为客户端连接到 TCP 服务器：

```
AT+CIPSTART="TCP","192.168.3.116",8080   // protocol、server IP & port
```

6. 发送数据：

```
AT+CIPSEND=4                                   // set date, such as 4 bytes
>TEST                                          // enter the data, no CR
```

响应：

```
SEND OK
```

> ⚠ **注意：**
>
> 若输入的字节数目超过了指令设定的长度 *(n)*，则会响应 busy，并发送数据的前 *n* 个字节，发送完成后响应 SEND OK。

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

7. 接收数据：

```
+IPD,n:xxxxxxxxxx                    // received n bytes, data=xxxxxxxxxx
```

## 9.2. UDP 传输

UDP 传输不区分服务器或者客户端，由指令 AT+CIPSTART 建立传输关系。

1. 设置 Wi-Fi 模式：

```
AT+CWMODE=3                          // SoftAP+Station mode
```

响应：

```
OK
```

2. 连接路由：

```
AT+CWJAP="SSID","password"           // SSID and password of router
```

响应：

```
OK
```

3. 查询设备 IP 地址：

```
AT+CIFSR
```

响应：

```
+CIFSR:STAIP,"192.168.101.104"       // IP address of ESP32 Station
```

4. PC 与 ESP32 连接同一个路由，在 PC 上使用网络调试工具，创建 UDP 传输。

   - 例如，假设 UDP 的 IP 地址为 192.168.101.110，端口为 8080。

下面介绍两种 UDP 通信的示例：

### 9.2.1. 固定远端的 UDP 通信

UDP 通信的远端固定，由 AT+CIPSTART 指令的最后参数 0 决定，分配一个连接号给这个固定连接，在通信过程中远端信息不会被改变。

1. 使能多连接：

```
AT+CIPMUX=1
```

响应：

```
OK
```

2. 创建 UDP 传输，例如，分配连接 ID 为 4。

```
AT+CIPSTART=4,"UDP","192.168.101.110",8080,1112,0
```

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

响应：

```
4,CONNECT
OK
```

> 📖 **说明：**
>
> - "192.168.101.110",8080 为 *UDP* 传输的远端 *IP* 和远端端口，也就是 *PC* 建立的 *UDP* 配置。
> - 1112 为 *ESP32* 的 *UDP* 本地端口，用户可自行设置，如不设置则为随机值。
> - 0 表示当前 *UDP* 传输建立后，*UDP* 远端不会被其他设备更改；即使有其他设备通过 *UDP* 协议发数据到 *ESP32 UDP* 端口 *1112*，*ESP32 4* 号 *UDP* 传输的远端也不会被替换，使用指令 AT+CIPSEND=4,X 发送数据，仍然是当前确定的 *PC* 端收到。

3. 发送数据：

```
AT+CIPSEND=4,7              // send 7 bytes to transmission NO.4
>UDPtest                      // enter the data, no CR
```

响应：

```
SEND OK
```

> ⚠️ **注意：**
>
> 若输入的字节数目超过了指令设定的长度 *(n)*，则会响应 busy，并发送数据的前 *n* 个字节，发送完成后响应 SEND OK。

4. 接收数据：

```
+IPD,4,n:xxxxxxxxxx          // received n bytes, data=xxxxxxxxxx
```

5. 断开 UDP 传输：

```
AT+CIPCLOSE=4
```

响应：

```
4,CLOSED
OK
```

### 9.2.2. 远端可变的 UDP 通信

1. 创建 UDP 传输，最后参数为 2：

```
AT+CIPSTART="UDP","192.168.101.110",8080,1112,2
```

响应：

```
CONNECT
OK
```

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

> 📖 **说明：**
>
> - `"192.168.101.110",8080` 为 *UDP* 传输的远端 *IP* 和远端端口，就是前述 *PC* 建立的 *UDP* 配置。
> - `1112` 为 *ESP32* 的 *UDP* 本地端口，用户可自行设置，如不设置则为随机值。
> - `2` 表示当前 *UDP* 传输建立后，*UDP* 传输远端信息仍然会更改；*UDP* 传输的远端信息会自动更改为最近一次与 *ESP32 UDP* 通信的远端 *IP* 和端口。

2. 发送数据：

```
AT+CIPSEND=7                // send 7 bytes
>UDPtest                      // enter the data, no CR
```

响应：

```
SEND OK
```

> ⚠ **注意：**
>
> 若输入的字节数目超过了指令设定的长度 *(n)*，则会响应 `busy`，并发送数据的前 *n* 个字节，发送完成后响应 `SEND OK`。

3. 若需要发 UDP 包给其他 UDP 远端，只需指定对方 IP 和端口即可。

```
AT+CIPSEND=6,"192.168.101.111",1000          // send six bytes
>abcdef                                        // enter the data, no CR
```

响应：

```
SEND OK
```

4. 接收数据：

```
+IPD,n:xxxxxxxxxx                                  // received n bytes, data=xxxxxxxxxx
```

5. 断开 UDP 传输：

```
AT+CIPCLOSE
```

响应：

```
CLOSED
OK
```

## 9.3. Wi-Fi 透传

AT Demo 仅在 ESP32 作为 TCP 客户端单连接或 UDP 传输时，支持透传。

### 9.3.1. TCP 客户端单连接透传

以下为 ESP32 作为 Station 实现 TCP 客户端单连接透传的举例。

1. 设置 Wi-Fi 模式：

LRi Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

```
AT+CWMODE=3                                    // SoftAP+Station mode
```

响应：

```
OK
```

2. 连接路由：

```
AT+CWJAP="SSID","password"                     // SSID and password of router
```

响应：

```
OK
```

3. 查询设备 IP 地址：

```
AT+CIFSR
```

响应：

```
192.168.101.105                                // device's IP that got from router
```

4. PC 与 ESP32 连接同一个路由，在 PC 上使用网络调试工具，创建一个 TCP 服务器。
   - 例如，假设 TCP 服务器的 IP 为 192.168.101.110，端口为 8080。
5. 设备作为 TCP 客户端连接 TCP 服务器：

```
AT+CIPSTART="TCP","192.168.101.110",8080     // protocol, server IP & port
```

响应：

```
OK
```

6. 使能透传模式：

```
AT+CIPMODE=1
```

响应：

```
OK
```

7. 发送数据：

```
AT+CIPSEND
```

响应：

```
>                  // From now on, data received from UART will be transparent
transmitted to server
```

8. 退出发送数据：

在透传发送数据过程中，若识别到单独的一包数据 +++，则退出透传发送。请至少间隔 1 秒，再发下一条 AT 指令。

请注意，如果直接用键盘打字输入 +++，有可能时间太慢，不被认为是连续的三个 +。

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

> ⚠ **注意：**
>
> +++ 退出透传发送数据，回到正常 *AT* 指令模式，*TCP* 连接仍然是保持的，也可以再发 AT+CIPSEND 指令，开始透传。

9. 退出透传模式：

```
AT+CIPMODE=0
```

响应：

```
OK
```

10.断开 TCP 连接：

```
AT+CIPCLOSE
```

响应：

```
CLOSED
OK
```

### 9.3.2. UDP 透传

以下为 ESP32 作为 SoftAP 实现 UDP 透传的举例。

1. 设置 Wi-Fi 模式：

```
AT+CWMODE=3                                    // SoftAP+Station mode
```

响应：

```
OK
```

2. PC 连入 ESP32 SoftAP：




3. 在 PC 上使用网络调试助手，创建一个 UDP。

   - 例如，假设 UDP 的 IP 为 192.168.4.2，端口为 1001。

4. ESP32 与 PC 对应端口建立固定对端的 UDP 传输。

```
AT+CIPSTART="UDP","192.168.4.2",1001,2233,0
```

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

响应：

```
OK
```

5. 使能透传模式：

```
AT+CIPMODE=1
```

响应：

```
OK
```

6. 发送数据：

```
AT+CIPSEND
```

响应：

```
>                    // from now on, data received from UART will be transparent
transmitted to server
```

7. 退出发送数据：

在透传发送数据过程中，若识别到单独的一包数据 +++，则退出透传发送。请至少间隔 1 秒，再发下一条 AT 指令。

请注意，如果直接用键盘打字输入 +++，有可能时间太慢，不被认为是连续的三个 +。

> ⚠ **注意：**
>
> +++ 退出透传发送数据，回到正常 *AT* 指令模式，*TCP* 连接仍然是保持的，也可以再发 AT+CIPSEND 指令，开始透传。

9. 退出透传模式：

```
AT+CIPMODE=0
```

响应：

```
OK
```

10.删除 UDP 传输：

```
AT+CIPCLOSE
```

响应：

```
CLOSED
OK
```

## 9.4. 多连接 TCP 服务器

目前 AT Demo ESP32 仅支持建立一个 TCP 服务器，且必须使能多连接，即可连接多个 TCP 客户端。

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

以下为 ESP32 作为 SoftAP ，建立 TCP 服务器的举例；如果是 ESP32 作为 Station，可在连接路由后，同理建立服务器。

1. 设置 Wi-Fi 模式：

```
AT+CWMODE=3                                    // SoftAP+Station mode
```

响应：

```
OK
```

2. 使能多连接：

```
AT+CIPMUX=1
```

响应：

```
OK
```

3. 建立 TCP 服务器：

```
AT+CIPSERVER=1                                 // default port = 333
```

响应：

```
OK
```

4. PC 连入 ESP32 SoftAP：




5. 在 PC 使用网络调试工具，作为 TCP 客户端连接设备。

> ⚠ **注意：**
>
> *ESP32 作为 TCP 服务器有超时机制，如果连接建立后，一段时间内无数据来往，ESP32 TCP 服务器会将 TCP 客户端 踢掉。请在 PC 网络工具连上 ESP32 后建立一个 2s 的循环数据发送，用于保持连接。*

6. 发送数据：

```
                               // ID number of connection is defaulted to be 0
AT+CIPSEND=0,4                 // send 4 bytes to connection NO.0
>iopd                          // enter the data, no CR
```

响应：

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

```
SEND OK
```

> ⚠ **注意：**
>
> 若输入的字节数目超过了指令设定的长度 *(n)*，则会响应 busy，并发送数据的前 *n* 个字节，发送完成后响应 SEND OK。

7. 接收数据：

```
+IPD,0,n:xxxxxxxxxx        // received n bytes, data = xxxxxxxxxx
```

8. 断开 TCP 连接：

```
AT+CIPCLOSE=0             // delete NO.0 connection
```

响应：

```
0,CLOSED
OK
```

## 9.5. BLE AT 指令应用

### 9.5.1. 基于 BLE 广播的应用—iBeacon

以下示例介绍两种关于 iBeacon 的应用方式：

- ESP32 广播 iBeacon 包，通过微信摇一摇周边来发现。
- ESP32 扫描发现其他设备广播的 iBeacon 包。

**表 9-1. iBeacon 格式**

| 类型 | 长度（字节） | 描述 |
|---|---|---|
| iBeacon prefix | 9 | 02 01 06 1A FF 4C 00 02 15 |
| Proximity UUID | 16 | 可用于区分不同设备厂商 |
| Major | 2 | 可用于区分不同商店 |
| Minor | 2 | 可用于区分同一家商店的不同位置 |

#### 9.5.1.1. ESP32 广播 iBeacon 包

1. 初始化 ESP32 模块的角色为 BLE server：

```
AT+BLEINIT=2                              // server role
```

响应：

```
OK
```

2. 进行广播

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

配置广播包内容如下：

表 9-2. iBeacon 广播包示例

| 类型 | 示例内容 |
| --- | --- |
| iBeacon prefix | 02 01 06 1A FF 4C 00 02 15 |
| Proximity UUID | FDA50693-A4E2-4FB1-AFCF-C6EB07647825 |
| Major | 27 B7 |
| Minor | F2 06 |
| TX power | C5 |

配置广播参数指令如下：

```
AT+BLEADVDATA="0201061aff4c000215fda50693a4e24fb1afcfc6eb0764782527b7f206c5"
```

响应：

```
OK
```

开始广播：

```
AT+BLEADVSTART
```

响应：

```
OK
```

打开微信摇一摇，即可发现正在广播的 ESP32 设备，如图 9-1 所示。

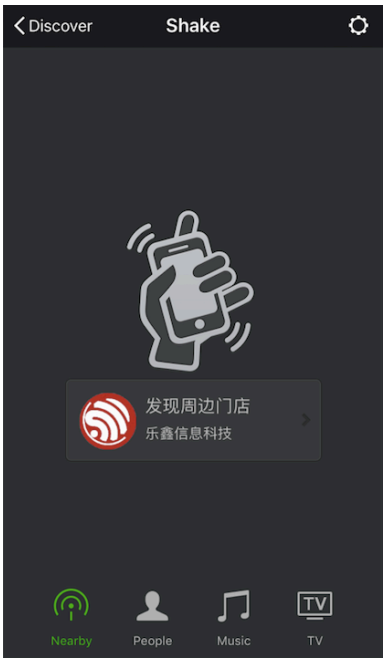



图 9-1. 微信摇一摇周边

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

#### 9.5.1.2. ESP32 扫描 iBeacon 包

ESP32 模块除了可以发 iBeacon 广播，也可以设置为 BLE client 进行扫描，获取到 iBeacon 广播包数据，再由 Host MCU 进行数据解析。

> 📖 **说明：**
>
> 如果该 *ESP32* 之前已经初始化为 *BLE server*，则需要先调用 *AT+BLEINIT=0* 注销，再重新初始化为 *BLE client*。

1. 初始化 ESP32 角色为 BLE client:

```
AT+BLEINIT=1                                    // client role
```

响应：

```
OK
```

2. 使能模块开始扫描，持续 3 秒：

```
AT+BLESCAN=1,3
```

响应：

```
OK
```

可以得到类似以下结果：

```
+BLESCAN:
24:0a:c4:02:10:0e,-33,0201061aff4c000215fda50693a4e24fb1afcfc6eb0764782527b7f206c5,

+BLESCAN:24:0a:c4:01:4d:fe,-74,02010207097a4f68664b43020aeb051220004000,

+BLESCAN:
24:0a:c4:02:10:0e,-33,0201061aff4c000215fda50693a4e24fb1afcfc6eb0764782527b7f206c5,
```

从扫描的结果中可以看到之前配置的广播包（比如章节 9.5.1.1 中配置的广播包），Host MCU 可以解析这串数据，其格式如前文表 9-1 所示。

### 9.5.2. 基于 BLE 连接的应用

#### 9.5.2.1. 基本通信示例

以下示例同时使用两块 ESP32，其中一块作为 Server（后文称为 "ESP32 Server"），另一块作为 Client（后文称为 "ESP32 Client"），举例如何建立 BLE 连接，完成数据通信。

> ⚠ **注意：**
>
> *ESP32 Server* 需烧录 *service bin* 文件到 *Flash* 中，用以提供服务。
>
> - 如何生成 *service bin* 文件，请参考 *esp32-at/tools/readme.md*。
> - *service bin* 文件的烧录地址，见 *esp32-at/at_customize.csv* 中 *ble_data* 对应的地址。

1. 初始化 BLE 功能：

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

- ESP32 Server：
  - 初始化为 BLE server：

```
AT+BLEINIT=2                                    // server role
```

响应：

```
OK
```

  - 创建服务：

```
AT+BLEGATTSSRVCRE                               // create services
```

响应：

```
OK
```

  - 开启服务：

```
AT+BLEGATTSSRVSTART                              // start services
```

响应：

```
OK
```

- ESP32 Client：

```
AT+BLEINIT=1                                  // client role
```

响应：

```
OK
```

2. 建立 BLE 连接：

- ESP32 Server：
  - 查询自身的 BLE 地址。假设 ESP32 Server 地址为 “24:0a:c4:03:f4:d6”，后文以此地址为例。

```
AT+BLEADDR?                                     // get server's BLE address
```

响应：

```
+BLEADDR:24:0a:c4:03:f4:d6
OK
```

配置广播参数（可选操作，一般无需配置）。如果没有配置广播参数，则使用默认广播参数。

```
AT+BLEADVPARAM=32,64,0,0,7
```

响应：

```
OK
```

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

- 配置广播数据（可选操作）。如果没有配置广播数据，那么广播包 payload 为空。

```
AT+BLEADVDATA="0201060B094573707265737369660303 02A0"

/*  The adv data is
 *  02 01 06  //<length>,<type>,<data>
 *  0A 09 457370726573736966 //<length>,<type>,<data>
 *  03 03 02A0  //<length>,<type>,<data>
 */
```

响应：

```
OK
```

除了配置广播数据之外，还可以配置 ScanRspData（扫描响应数据）：

```
AT+BLESCANRSPDATA="0201060B094573707265737369660303 02A0"
```

响应：

```
OK
```

如果 client 是 Active Scan，那么就能扫描到 ScanRspData。

- 开始广播：

```
AT+BLEADVSTART
```

响应：

```
OK
```

- ESP32 Client：
  - 配置扫描参数（可选操作，一般无需配置）。在开始扫描之前，可以配置扫描参数，假设配置的是 active scan 模式，指令如下：

```
AT+BLESCANPARAM=1,0,0,100,50
```

响应：

```
OK
```

  - 进行扫描：

```
AT+BLESCAN=1,3
```

响应：

```
+BLESCAN:<BLE address>,<rssi>,<adv data>,<scan rsp data>
OK
```

  - 成功扫描到 ESP32 Server 后，进行连接：

```
AT+BLECONN=0,"24:0a:c4:03:f4:d6"
```

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

响应：

```
OK
+BLECONN:0,"24:0a:c4:03:f4:d6"
```

> 📖 **说明：**
>
> - 如果 *BLE* 连接成功建立，则提示 +BLECONN:<conn_index>,<remote_BLE_address>
> - 如果 *BLE* 连接断开，则提示 +BLEDISCONN:<conn_index>,<remote_BLE_address>

- 更新连接参数

  在建立连接之后，可以更新连接参数：

```
AT+BLECONNPARAM=0,30,30,0,600
```

响应：

```
OK
```

也可以查询结果：

```
AT+BLECONNPARAM?
```

响应：

```
+BLECONNPARAM:0,30,30,30,0,600

OK
```

- 配置最大传输单元 (MTU)

  在连接建立之后，client 可以发起 Exchange MTU Request：

```
AT+BLECFGMTU=0,200
```

响应：

```
OK
```

也可以查询结果：

```
AT+BLECFGMTU?
```

响应：

```
+BLECFGMTU:0,200

OK
```

3. 读写服务特征：

- ESP32 Server：

  - 查询本地服务：

```
AT+BLEGATTSSRV?
```

响应：

```
+BLEGATTSSRV:1,1,0xA002,1

OK
```

- 查询本地服务特征：

```
AT+BLEGATTSCHAR?
```

响应：

```
+BLEGATTSCHAR:"char",1,1,0xC300
+BLEGATTSCHAR:"desc",1,1,1
+BLEGATTSCHAR:"char",1,2,0xC301
+BLEGATTSCHAR:"desc",1,2,1
+BLEGATTSCHAR:"char",1,3,0xC302
+BLEGATTSCHAR:"desc",1,3,1

OK
```

- ESP32 Client：
  - 查询服务：

```
AT+BLEGATTCPRIMSRV=0
```

响应：

```
+BLEGATTCPRIMSRV:0,1,0x1801,1
+BLEGATTCPRIMSRV:0,2,0x1800,1
+BLEGATTCPRIMSRV:0,3,0xA002,1

OK
```

> ⚠ **注意：**
>
> - *Client* 查询服务的结果，比 *Server* 查询服务的结果多两个默认服务（*UUID: 0x1800* 和 *0x1801*），这是正常现象。正因如此，对于同一服务，*Client* 查询的 *<srv_index>* 值 = *Server* 查询的 *<srv_index>* 值 + *2*。
> - 例如，上述示例中的服务 *0xA002*，当前在 *ESP32 Client* 侧查询到的 *<srv_index>* 为 *3*，如果在 *ESP32 Server* 侧查询（*AT+BLEGATTSSRV?*），则对应 *<srv_index>* 为 *1*。

  - 查询服务特征值：

```
AT+BLEGATTCCHAR=0,3
```

响应：

```
+BLEGATTCCHAR:"char",0,3,1,0xC300,2
+BLEGATTCCHAR:"desc",0,3,1,1,0x2901
```

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

```
+BLEGATTCCHAR:"char",0,3,2,0xC301,2
+BLEGATTCCHAR:"desc",0,3,2,1,0x2901
+BLEGATTCCHAR:"char",0,3,3,0xC302,8
+BLEGATTCCHAR:"desc",0,3,3,1,0x2901
+BLEGATTCCHAR:"char",0,3,4,0xC303,4
+BLEGATTCCHAR:"desc",0,3,4,1,0x2901
+BLEGATTCCHAR:"char",0,3,5,0xC304,8
+BLEGATTCCHAR:"char",0,3,6,0xC305,16
+BLEGATTCCHAR:"desc",0,3,6,1,0x2902
+BLEGATTCCHAR:"char",0,3,7,0xC306,32
+BLEGATTCCHAR:"desc",0,3,7,1,0x2902

OK
```

- 读取服务特征（要求该特征属性支持读取操作）：

```
AT+BLEGATTCRD=0,3,1
```

响应：

```
+BLEGATTCRD:0,1,30
OK
```

> 📖 **说明：**
>
> *ESP32 Client* 读取特征成功，*ESP32 Server* 提示 `+READ:<conn_index>,<remote BLE address>`

- 写服务特征（要求该特征属性支持写操作）：

```
AT+BLEGATTCWR=0,3,3,,2
```

响应：

```
>               // waiting for serial data
OK
```

> 📖 **说明：**
>
> *ESP32 Client* 写特征成功，*ESP32 Server* 提示 `+WRITE:<conn_index>,<srv_index>,<char_index>,[<desc_index>], <len>,<value>`

4. 通知服务特征：

- ESP32 Client：
  - 设置服务特征描述符（要求该特征属性支持通知操作）：

```
AT+BLEGATTCWR=0,3,6,1,2
```

响应：

```
>               // waiting for serial data, should input HEX string "01" here
OK
```

> 📖 **说明：**
>
> *ESP32 Client* 写特征描述符成功，*ESP32 Server* 提示
> `+WRITE:<conn_index>,<srv_index>,<char_index>,<desc_index>, <len>,<value>`

- ESP32 Server：
  - 服务特征通知（要求该特征属性支持通知操作）：

```
AT+BLEGATTSNTFY=0,1,6,3
```

响应：

```
>                // waiting for serial data
OK
```

> 📖 **说明：**
>
> - *ESP32 Client* 收到通知，将提示 `+NOTIFY:<conn_index>,<srv_index>,<char_index>,<len>,<value>`
> - 对于同一服务，*Client* 侧的 *<srv_index>* 值 = *Server* 侧的 *<srv_index>* 值 + *2*，这是正常现象。

5. 指示服务特征：

- ESP32 Client：
  - 设置服务特征描述符（要求该特征属性支持指示操作）：

```
AT+BLEGATTCWR=0,3,7,1,2
```

响应：

```
>                // waiting for serial data, should input HEX string "02" here
OK
```

> 📖 **说明：**
>
> *ESP32 Client* 写特征描述符成功，*ESP32 Server* 提示
> `+WRITE:<conn_index>,<srv_index>,<char_index>,<desc_index>, <len>,<value>`

- ESP32 Server：
  - 服务特征指示（要求该特征属性支持指示操作）：

```
AT+BLEGATTSIND=0,1,7,3
```

响应：

```
>                // waiting for serial data
OK
```

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

> 📖 **说明：**
>
> - *ESP32 Client* 收到指示，将提示 `+INDICATE:<conn_index>,<srv_index>,<char_index>,<len>,<value>`
> - 对于同一服务，*Client* 侧的 *<srv_index>* 值 = *Server* 侧的 *<srv_index>* 值 + 2，这是正常现象。

#### 9.5.2.2. BLE 透传示例

以下示例同时使用两块 ESP32，其中一块作为 Server（后文称为“ESP32 Server”），另一块作为 Client（后文称为“ESP32 Client”），举例如何建立 BLE 连接，建立透传通信。

> ⚠ **注意：**
>
> *ESP32 Server* 需烧录 *service bin* 文件到 *flash* 中，用以提供服务。
>
> - 如何生成 *service bin* 文件，请参考 *esp32-at/tools/readme.md*。
> - *service bin* 文件的烧录地址，见 *esp32-at/at_customize.csv* 中 *ble_data* 对应的地址。

1. 初始化 BLE 功能：

- ESP32 Server：
  - 初始化为 BLE server：

```
AT+BLEINIT=2                                    // server role
```

响应：

```
OK
```

  - 创建服务：

```
AT+BLEGATTSSRVCRE                               // create services
```

响应：

```
OK
```

  - 开启服务：

```
AT+BLEGATTSSRVSTART                              // start services
```

响应：

```
OK
```

- ESP32 Client：

```
AT+BLEINIT=1                                    // client role
```

响应：

```
OK
```

2. 建立 BLE 连接：

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

- ESP32 Server：
  - 查询自身的 BLE 地址。假设 ESP32 Server 地址为 “24:0a:c4:03:f4:d6”，后文以此地址为例。

```
AT+BLEADDR?                                  // get server's BLE address
```

响应：

```
+BLEADDR:24:0a:c4:03:f4:d6
OK
```

  - 配置广播数据（可选操作）。如果没有配置广播数据，那么广播包 payload 为空。

```
AT+BLEADVDATA="0201060B094573707265737369660303 02A0"

/*  The adv data is
 *  02 01 06  //<length>,<type>,<data>
 *  0A 09 457370726573736966 //<length>,<type>,<data>
 *  03 03 02A0  //<length>,<type>,<data>
 */
```

响应：

```
OK
```

  - 开始广播：

```
AT+BLEADVSTART
```

响应：

```
OK
```

- ESP32 Client：
  - 进行扫描：

```
AT+BLESCAN=1,3
```

响应：

```
+BLESCAN:<BLE address>,<rssi>,<adv data>,<scan rsp data>
OK
```

  - 成功扫描到 ESP32 Server 后，进行连接：

```
AT+BLECONN=0,"24:0a:c4:03:f4:d6"
```

响应：

```
OK
+BLECONN:0,"24:0a:c4:03:f4:d6"
```

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000



> 📖 **说明：**
>
> - 如果 *BLE* 连接成功建立，则提示 `+BLECONN:<conn_index>,<remote_BLE_address>`
> - 如果 *BLE* 连接断开，则提示 `+BLEDISCONN:<conn_index>,<remote_BLE_address>`

3. 查询服务：

- ESP32 Server：
  - 查询本地服务：

```
AT+BLEGATTSSRV?
```

响应：

```
+BLEGATTSSRV:1,1,0xA002,1

OK
```

  - 查询本地服务特征：

```
AT+BLEGATTSCHAR?
```

响应：

```
+BLEGATTSCHAR:"char",1,1,0xC300
+BLEGATTSCHAR:"desc",1,1,1
+BLEGATTSCHAR:"char",1,2,0xC301
+BLEGATTSCHAR:"desc",1,2,1
+BLEGATTSCHAR:"char",1,3,0xC302
+BLEGATTSCHAR:"desc",1,3,1

OK
```

- ESP32 Client：
  - 查询服务：

```
AT+BLEGATTCPRIMSRV=0
```

响应：

```
+BLEGATTCPRIMSRV:0,1,0x1801,1
+BLEGATTCPRIMSRV:0,2,0x1800,1
+BLEGATTCPRIMSRV:0,3,0xA002,1

OK
```

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

> ⚠ **注意：**
>
> - *Client* 查询服务的结果，比 *Server* 查询服务的结果多两个默认服务（*UUID: 0x1800* 和 *0x1801*），这是正常现象。正因如此，对于同一服务，*Client* 查询的 *<srv_index>* 值 = *Server* 查询的 *<srv_index>* 值 + *2*。
> - 例如，上述示例中的服务 *0xA002*，当前在 *ESP32 Client* 侧查询到的 *<srv_index>* 为 *3*，如果在 *ESP32 Server* 侧查询（*AT+BLEGATTSSRV?*），则对应 *<srv_index>* 为 *1*。

- 查询服务特征值：

```
AT+BLEGATTCCHAR=0,3
```

响应：

```
+BLEGATTCCHAR:"char",0,3,1,0xC300,0x02
+BLEGATTCCHAR:"desc",0,3,1,1,0x2901
+BLEGATTCCHAR:"char",0,3,2,0xC301,0x02
+BLEGATTCCHAR:"desc",0,3,2,1,0x2901
+BLEGATTCCHAR:"char",0,3,3,0xC302,0x08
+BLEGATTCCHAR:"desc",0,3,3,1,0x2901
+BLEGATTCCHAR:"char",0,3,4,0xC303,0x04
+BLEGATTCCHAR:"desc",0,3,4,1,0x2901
+BLEGATTCCHAR:"char",0,3,5,0xC304,0x08
+BLEGATTCCHAR:"char",0,3,6,0xC305,0x10
+BLEGATTCCHAR:"desc",0,3,6,1,0x2902
+BLEGATTCCHAR:"char",0,3,7,0xC306,0x20
+BLEGATTCCHAR:"desc",0,3,7,1,0x2902

OK
```

4. 配置透传：

- ESP32 Client：
  - 选择支持写操作的 characteristic 作为写通道发送数据，选择支持 notify 或者 indicate 的 characteristic 作为读通道接收数据，例如：

```
AT+BLESPPCFG=1,3,5,3,7
```

响应：

```
OK
```

- 开启透传模式：

```
AT+BLESPP
```

响应：

```
OK
>               // waiting for serial data
```

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

> 📖 **说明：**
>
> *ESP32 Client* 开启透传模式后，串口收到的数据会通过 *BLE* 传输到 *Server* 端。

- ESP32 Server：
  - 选择支持 notify 或者 indicate 的 characteristic 作为写通道发送数据，选择支持写操作的 characteristic 作为读通道接收数据，例如：

```
AT+BLESPPCFG=1,1,7,1,5
```

响应：

```
OK
```

  - 开启透传模式：

```
AT+BLESPP
```

响应：

```
OK
>               // waiting for serial data
```

> 📖 **说明：**
>
> - *ESP32 Server* 开启透传模式后，串口收到的数据会通过 *BLE* 传输到 *Client* 端。
> - 如果 *ESP32 Client* 端没有先开启透传，或者使用其他设备作为 *Client*，则 *Client* 端需要先开启侦听 *Notify* 或者 *Indicate*。例如，*ESP32 Client* 如果未开启透传，则应先调用 *AT+BLEGATTCWR=0,3,7,1,1* 开启侦听，*ESP32 Server* 才能成功实现透传。
> - 对于同一服务，*Client* 侧的 *<srv_index>* 值 = *Server* 侧的 *<srv_index>* 值 + *2*，这是正常现象。

#### 9.5.2.3. 基于 BLE 连接的应用场景

1. 蓝牙配网

用户可以通过 BLE 通信传递 Wi-Fi SSID 和 Password 信息，以实现蓝牙配置无线网络的功能。

- 如果是 BLE client 向 BLE server 传递 SSID 和 Password 信息，可以使用指令 `AT+BLEGATTCWR`；
- 如果是 BLE server 向 BLE client 传递 SSID 和 Password 信息，可以使用指令 `AT+BLEGATTSNTFY`；

详细步骤可参考前文 9.5.2.1 章节。

2. 数据透传

当前 ESP32 AT 尚未实现 BLE 透传功能，但是用户可以使用基本的数据传输方式来模拟这一过程，由 Host MCU 过滤出数据信息即可。

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

- BLE client 向 BLE server 传输数据，可由不断的调用指令 `AT+BLEGATTCWR` 实现；
- BLE server 向 BLE client 传输数据，可由不断调用指令 `AT+BLEGATTSNTFY` 实现；

详细步骤可参考前文 9.5.2.1 章节。

3. 固件升级

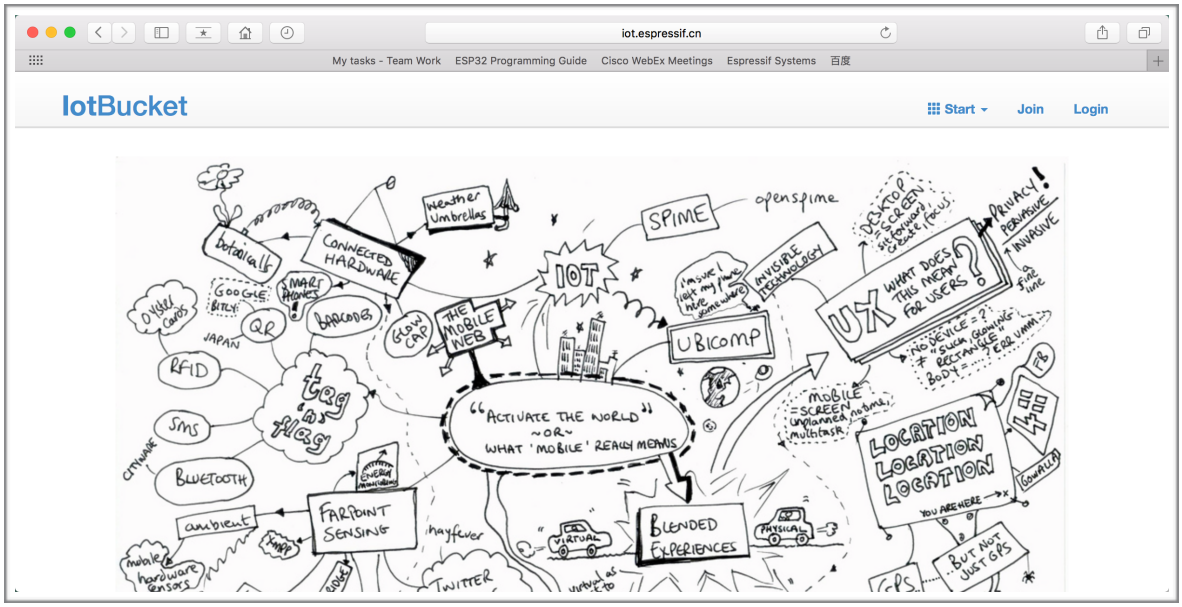
BLE 数据通信，也可以用于传输固件，实现升级功能。

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

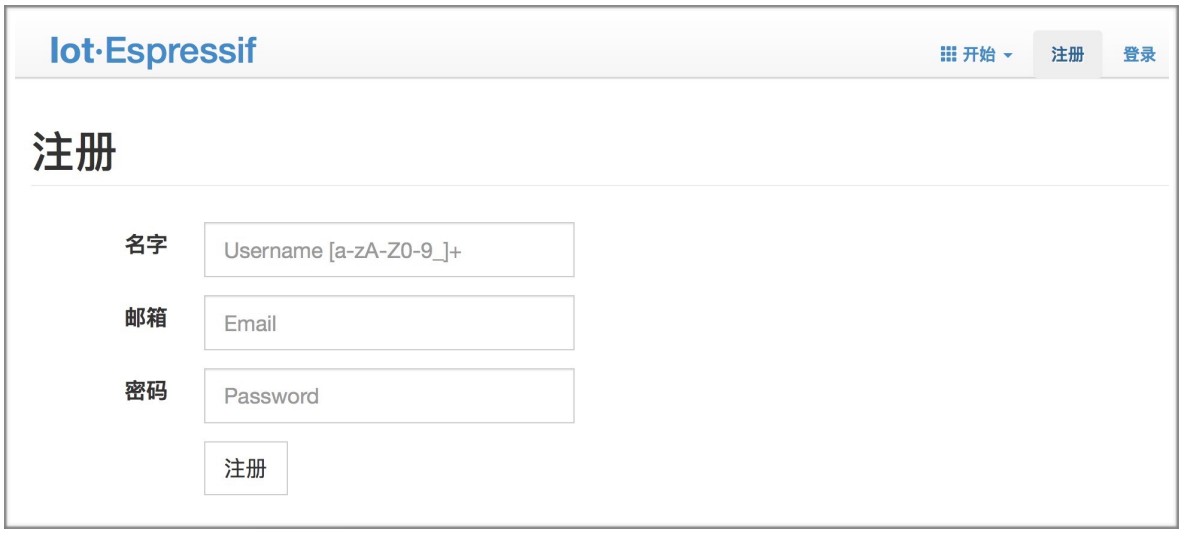
# 10. OTA 功能

本章节主要描述如何在 iot.espressif.cn 上创建自己的设备，并自定义基于 iot.espressif.cn 的 OTA 功能。

1. 打开浏览器输入 iot.espressif.cn。如果使用 SSL 升级，则网址为 https://iot.espressif.cn。



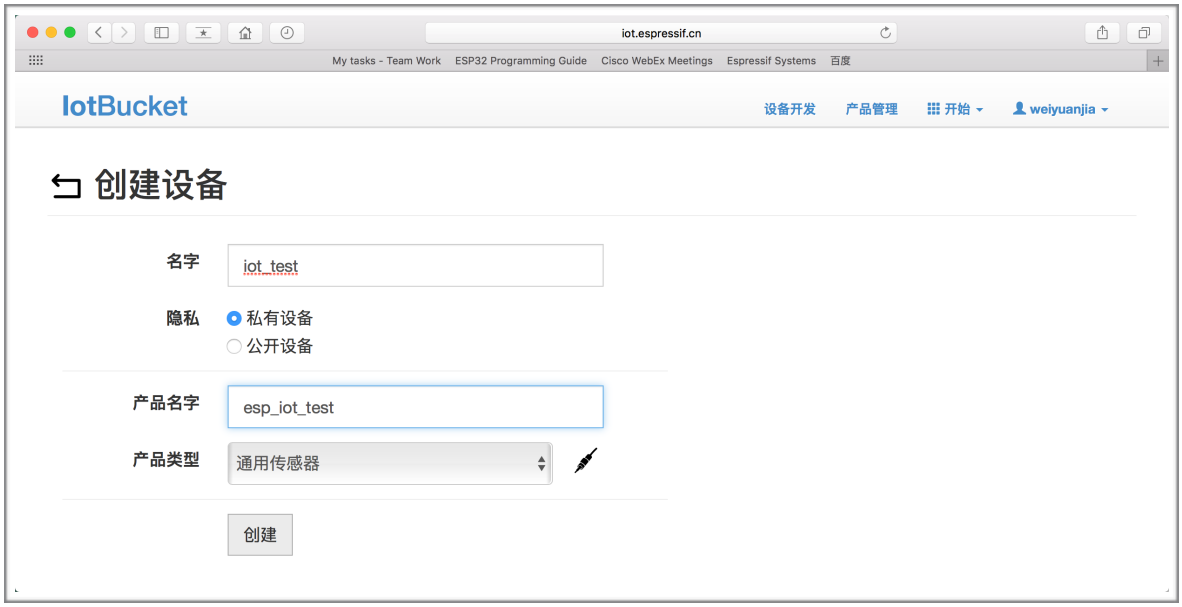

2. 点击右上角的 Join，输入用户名、邮箱和密码，注册账号。



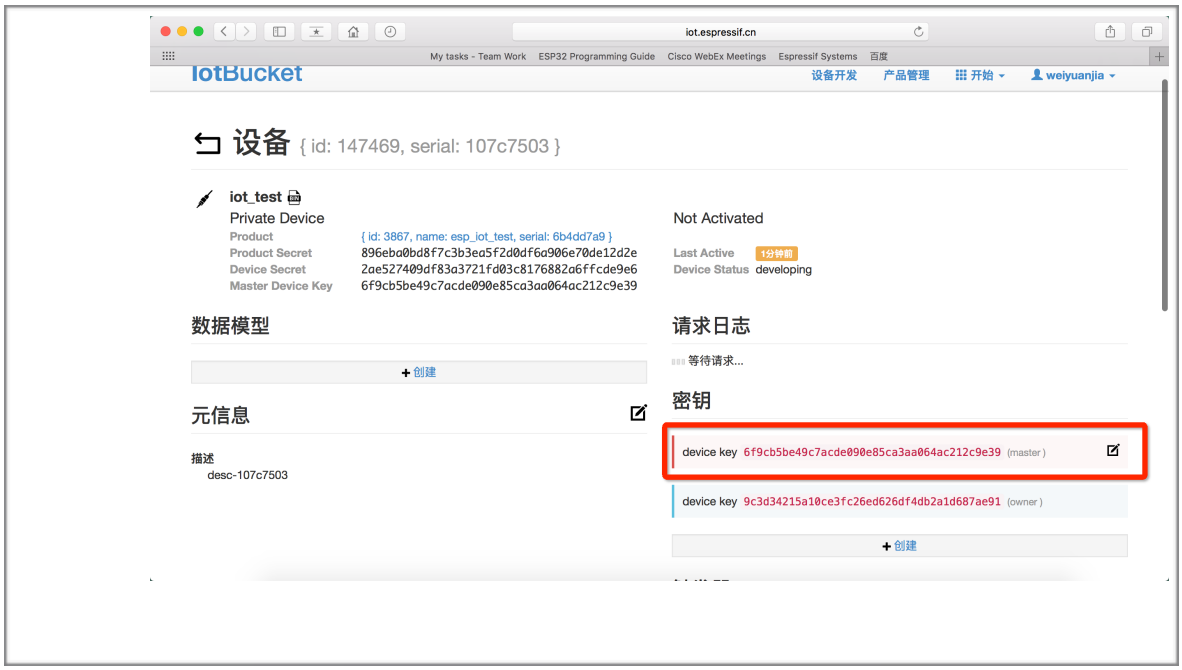

3. 然后点击“设备开发”里面的“创建”，创建一个自己的设备。

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000



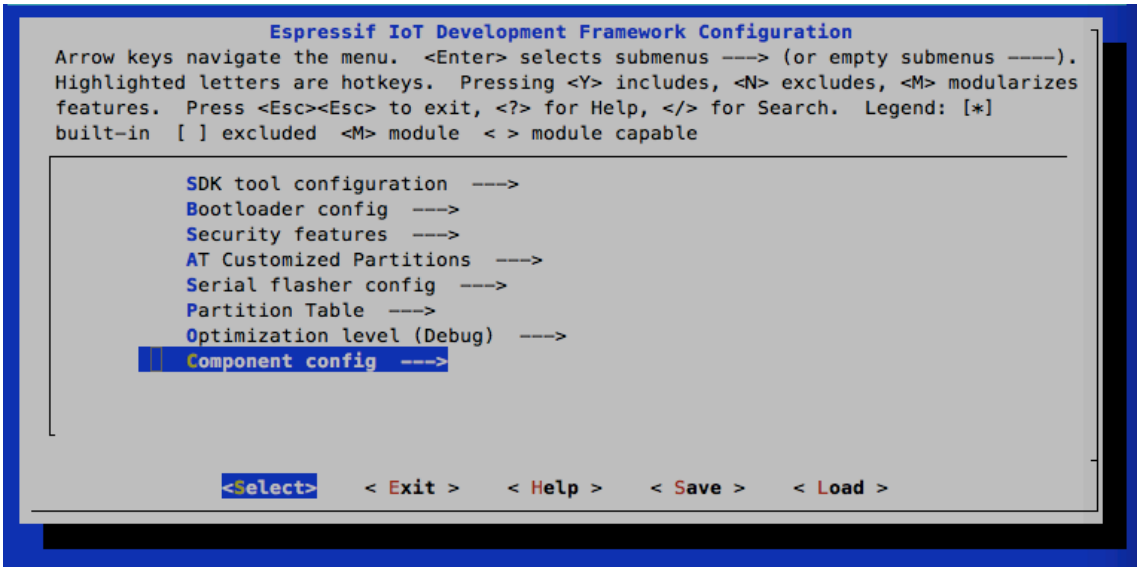



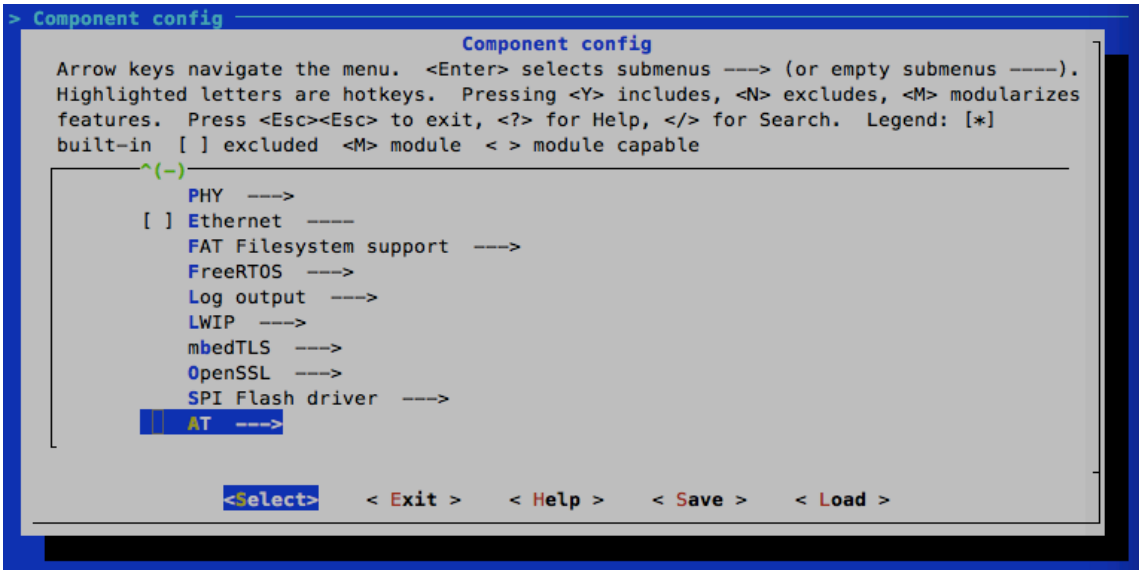

4. 创建成功后，会有一个密钥。

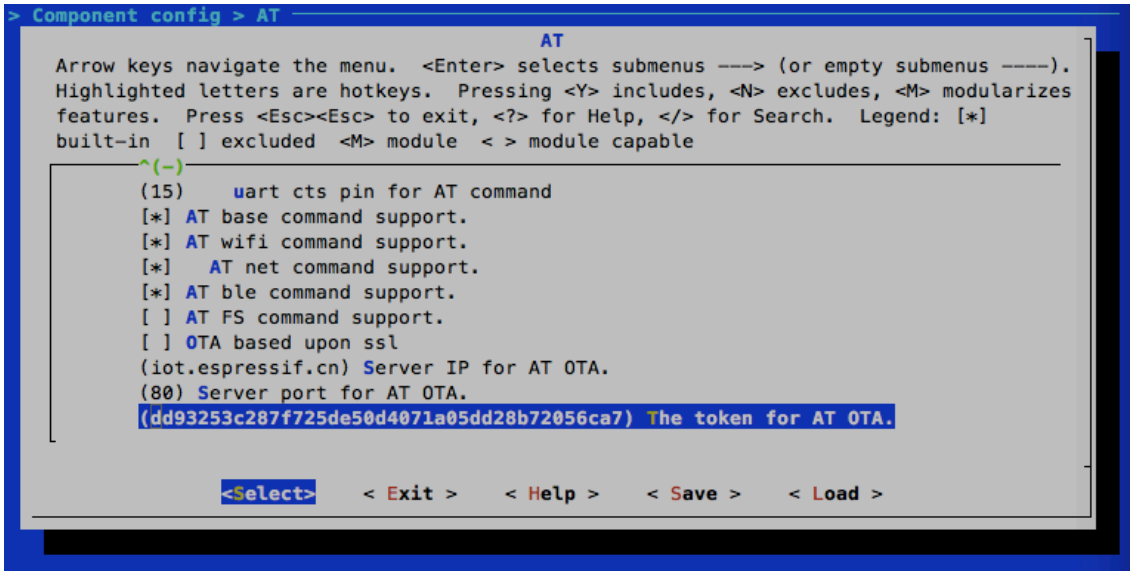



5. 使用密钥，编译自己的 OTA BIN。修改 AT OTA token 密钥的配置流程如下：

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000



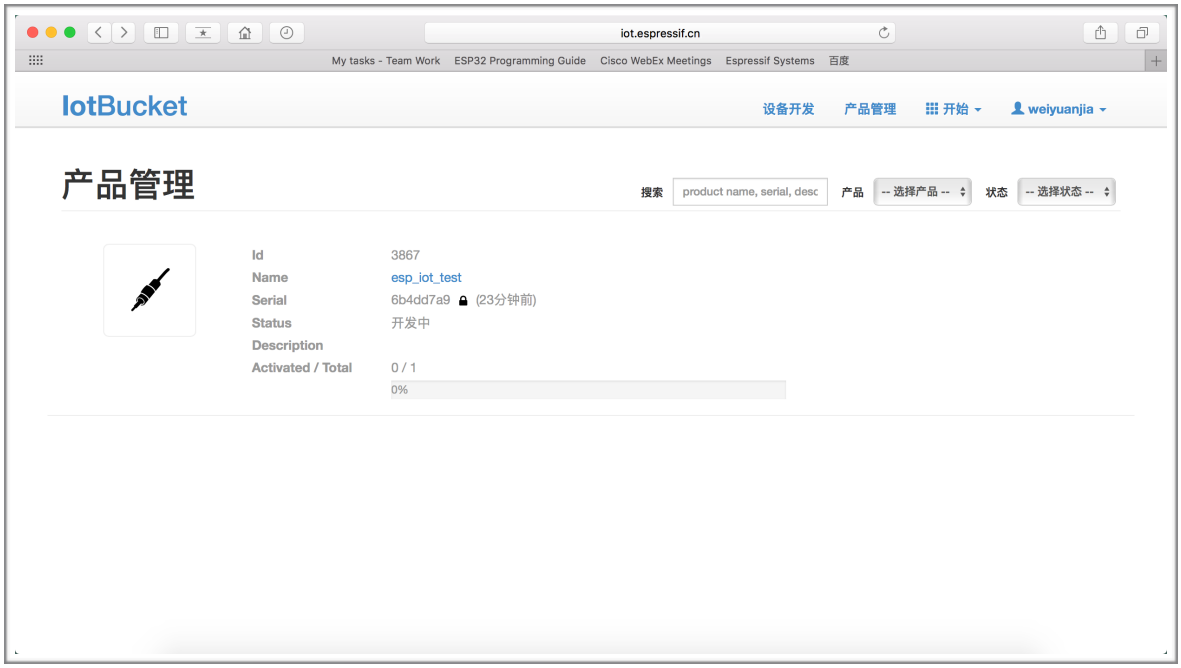



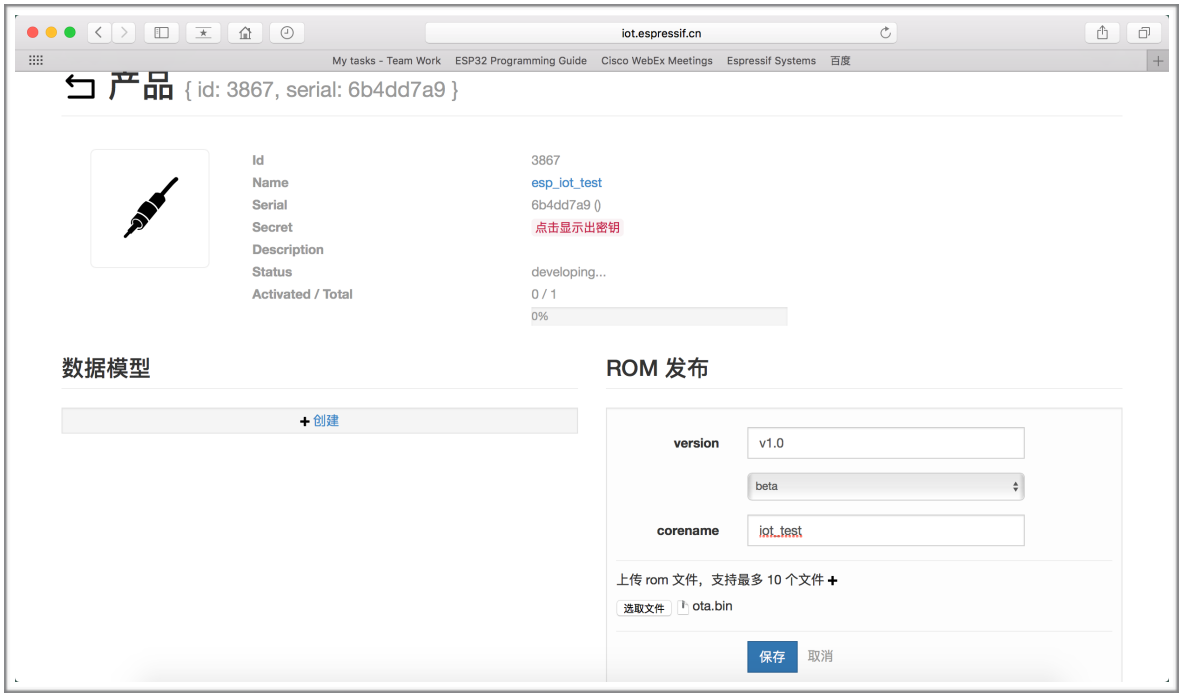




Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

> ⚠ **注意：**
>
> 如果使用 *SSL OTA* 功能，需在 *menuconfig* 中选择 "*OTA based upon ssl*"，输入空格进行选择。

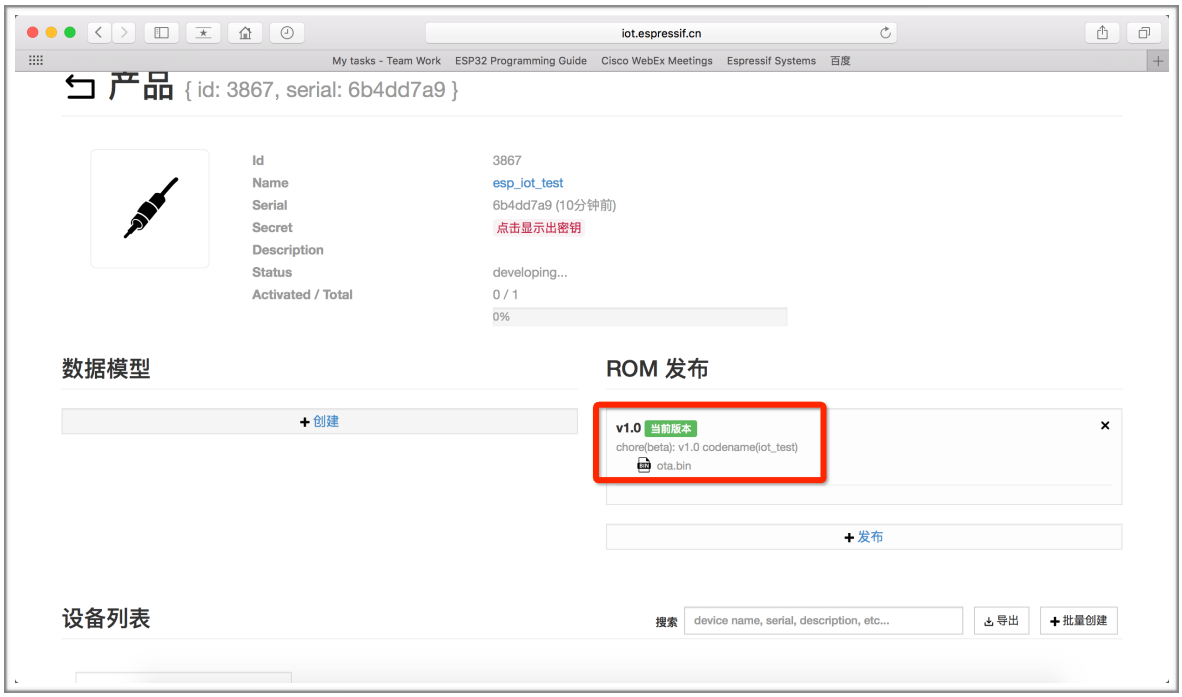
6. 进入产品管理界面，点击刚才创建的设备，在 ROM 发布下面，输入 version 和 corename，将上面编译的 BIN 文件重命名为 ota.bin，然后选取并点击保存。




7. 点击设置为当前版本。

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000



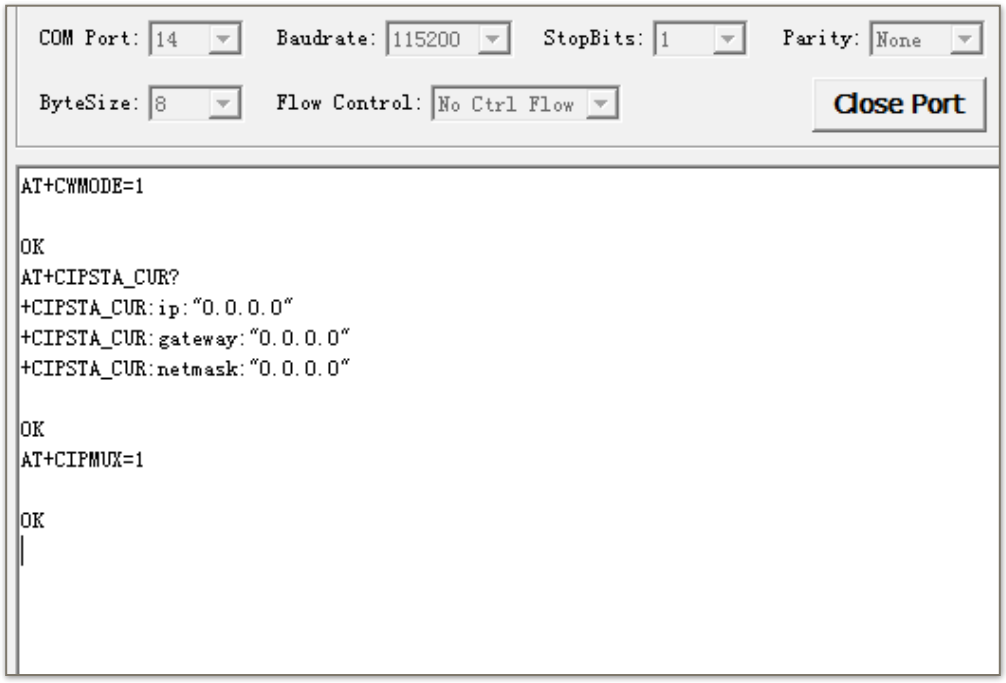

8. 发送 AT+CIUPDATE，如果网络正常，则即可完成 OTA 升级。

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

# 11. 问题反馈

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-32；
- 测试指令步骤说明或截图，例如下图：




- 如能提供 log 打印信息，请附上异常 log 信息，例如以下截屏：

```
Guru Meditation Error of type StoreProhibited occurred on core   0. Exception was unhandled.
Register dump:
PC      :  40135735  PS      :  00060f30  A0      :  800f913b  A1      :  3ffd66c0
A2      :  00000000  A3      :  3ffd6828  A4      :  00000b68  A5      :  b33f0000
A6      :  b33fffff  A7      :  3ffb004c  A8      :  00000003  A9      :  3ffd66a0
A10     :  3ffd6828  A11     :  00000b69  A12     :  00060020  A13     :  3ffc2d30
A14     :  00000003  A15     :  00060023  SAR     :  00000000  EXCCAUSE:  0000001d
EXCVADDR:  00000038  LBEG    :  00000000  LEND    :  00000000  LCOUNT  :  00000000
Rebooting...
```

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000



乐鑫 IOT 团队
*www.espressif.com*

**免责申明和版权公告**

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。**

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000