

OPL1000

ULTRA-LOW POWER 2.4GHZ WI-FI + BLUETOOTH SMART SOC

AT Instruction Set and Examples



OPULINKS

<http://www.opulinks.com/>

Copyright © 2017-2018, OpuLinks. All Rights Reserved.

OPL1000-AT-instruction-set-example-R05 | Version 0.15

| Date | Version | Contents Updated |
|------------|---------|---|
| 2018-04-16 | 0.1 | <ul style="list-style-type: none">Initial Release |
| 2018-05-18 | 0.2 | <ul style="list-style-type: none">Add wifi exampleAdd AT+CWAUTOCONN |
| 2018-05-25 | 0.3 | <ul style="list-style-type: none">Fix some mistakes |
| 2018-05-31 | 0.4 | <ul style="list-style-type: none">Add tcp server exampleFix some mistakes |
| 2018-06-20 | 0.5 | <ul style="list-style-type: none">Fix some mistakes |
| 2018-06-27 | 0.6 | <ul style="list-style-type: none">Fixed some wrong description in several AT commands suchas AT+BLECFGMTU etc. |
| 2018-06-28 | 0.7 | <ul style="list-style-type: none">Fix some description mistakes |
| 2018-07-17 | 0.8 | <ul style="list-style-type: none">Revise CIPSTATUS descriptionAdd AT+MACADDRDEF |
| 2018-07-19 | 0.9 | <ul style="list-style-type: none">Add AT+RFHPModify AT+BLEADDR—设置 BLE public 设备地址 |
| 2018-07-27 | 0.10 | <ul style="list-style-type: none">Add more description for AT+BLEADDR and AT+CIPSTAMAC |
| 2018-07-27 | 0.11 | <ul style="list-style-type: none">Add AT+READFLASH, AT+WRITEFLASH and AT+ERASEFLASH |
| 2018-08-02 | 0.12 | <ul style="list-style-type: none">Add more description for AT+CWMODE and AT+CWLAP |
| 2018-08-17 | 0.13 | <ul style="list-style-type: none">Add AT+DHCPARPCHK |
| 2018-08-27 | 0.14 | <ul style="list-style-type: none">Revise AT+CWMODE description |
| 2018-08-27 | 0.15 | <ul style="list-style-type: none">Update AT+RFHP description |

TABLE OF CONTENTS

| | |
|--|----|
| 1. 介绍 | 1 |
| 1.1. 文档应用范围 | 1 |
| 1.2. 指令说明 | 1 |
| 2. 基础 AT 指令 | 2 |
| 2.1. 基础 AT 指令一览表 | 2 |
| 2.2. 基础 AT 指令描述 | 3 |
| 2.2.1. AT—测试 AT 模块 | 3 |
| 2.2.2. AT+RST—重启模块 | 3 |
| 2.2.3. AT+GMR—查询版本信息 | 3 |
| 2.2.4. AT+GSLP—进入 Deep-sleep 模式 | 4 |
| 2.2.5. ATE—开关回显功能 | 4 |
| 2.2.6. AT+RESTORE—恢复出厂设置 | 4 |
| 2.2.7. AT+UART_CUR—设置 UART 临时配置 | 4 |
| 2.2.8. AT+UART_DEF—设置 UART 配置，保存到 Flash | 6 |
| 2.2.9. AT+SLEEP—设置 sleep 模式 | 7 |
| 2.2.10. AT+SYSRAM—查询当前剩余 RAM 大小 | 7 |
| 2.2.11. AT+MACADDRDEF—设置设备来源 Mac Address，保存到 flash | 8 |
| 2.2.12. AT+RFHP—设置 RF Power | 9 |
| 2.2.13. AT+READFLASH—读取 Flash | 9 |
| 2.2.14. AT+WRITEFLASH—写入 Flash | 10 |
| 2.2.15. AT+ERASEFLASH—抹除 Flash | 11 |
| 2.2.16. AT+DHCPARPCHK—设置 DHCP ARP 检查机制 | 12 |
| 3. WIFI 功能 AT 指令 | 13 |
| 3.1. WIFI 功能 AT 指令一览表 | 13 |
| 3.2. WIFI 功能 AT 指令描述 | 14 |
| 3.2.1. AT+CWMODE—设置 Wi-Fi 模式 | 14 |
| 3.2.2. AT+CWJAP—连接 AP | 15 |
| 3.2.3. AT+CWLAPOPT—设置 CWLAP 指令的属性 | 17 |
| 3.2.4. AT+CWLAP—扫描当前可用的 AP | 17 |
| 3.2.5. AT+CWQAP—断开与 AP 的连接 | 18 |
| 3.2.6. AT+CWDHCP—设置 DHCP - TBD | 18 |
| 3.2.7. AT+CWAUTOCONN—上电是否自动连接 AP | 19 |
| 3.2.8. AT+CIPSTAMAC—设置 OPL1000 Station 接口的 MAC 地址 | 19 |
| 3.2.9. AT+CWHOSTNAME—设置 Station 的主机名称 | 20 |

| | |
|--|----|
| 4. TCP/IP 功能 AT 指令 | 22 |
| 4.1. TCP/IP 功能 AT 指令一览表 | 22 |
| 4.2. TCP/IP 功能 AT 指令描述 | 22 |
| 4.2.1. AT+CIPSTATUS—查询网络连接信息 | 22 |
| 4.2.2. AT+CIPDOMAIN—域名解析功能 | 23 |
| 4.2.3. AT+CIPSTART—建立 TCP 连接或 UDP 传输 | 24 |
| 4.2.4. AT+CIPSEND—发送数据 | 25 |
| 4.2.5. AT+CIPSENDEX—发送数据 | 26 |
| 4.2.6. AT+CIPCLOSE—关闭 TCP/UDP 传输 | 27 |
| 4.2.7. AT+CIFSR—查询本地 IP 地址 | 28 |
| 4.2.8. AT+CIPMUX—设置多连接 | 28 |
| 4.2.9. AT+CIPSERVER—建立 TCP 服务器 | 29 |
| 4.2.10. AT+CIPSTO—设置 TCP 服务器超时时间 | 29 |
| 4.2.11. AT+CIPDINFO—接收网络数据时是否提示对端 IP 和端口 | 30 |
| 4.2.12. +IPD—接收网络数据 | 30 |
| 4.2.13. AT+PING—Ping 功能 | 31 |
| 5. BLE 相关 AT 指令 | 32 |
| 5.1. BLE 指令一览表 | 32 |
| 5.2. BLE 指令描述 | 33 |
| 5.2.1. AT+BLEINIT—BLE 初始化 | 33 |
| 5.2.2. AT+BLEADDR—设置 BLE 设备地址 | 34 |
| 5.2.3. AT+BLENAME—设置 BLE 设备名称 | 34 |
| 5.2.4. AT+BLESCANRSPDATA—设置 BLE 扫描响应 | 35 |
| 5.2.5. AT+BLEADVPARAM—设置广播参数 | 35 |
| 5.2.6. AT+BLEADVDATA—设置 BLE 广播数据 | 37 |
| 5.2.7. AT+BLEADVSTART—开始 BLE 广播 | 37 |
| 5.2.8. AT+BLEADVSTOP—结束 BLE 广播 | 37 |
| 5.2.9. AT+BLECONNPARAM—更新 BLE 连接参数 | 38 |
| 5.2.10. AT+BLEDISCONN—断开 BLE 连接 | 39 |
| 5.2.11. AT+BLEDATALEN—设置 BLE 数据包长度 | 39 |
| 5.2.12. AT+BLECFGMTU—设置 GATT MTU 的长度 | 40 |
| 5.2.13. AT+BLEGATTSSRVCRE—GATTS 创建服务 | 40 |
| 5.2.14. AT+BLEGATTSSRVSTART—GATTS 开启服务 | 41 |
| 5.2.15. AT+BLEGATTSSRVSTOP—GATTS 停止服务 | 41 |
| 5.2.16. AT+BLEGATTSSRV—GATTS 发现服务 | 41 |
| 5.2.17. AT+BLEGATTSSCHAR—GATTS 发现服务特征 | 42 |
| 5.2.18. AT+BLEGATTSENTFY—GATTS 通知服务特征值 | 43 |

5.2.19. AT+BLEGATTSIND—GATTS 指示服务特征值 _____ 44

5.2.20. AT+BLEGATTSSETATTR—GATTS 设置服务特征值 _____ 44

5.2.21. AT+BLEGATTCPRIMSRV—GATTC 发现基本服务 _____ 45

5.2.22. AT+BLEGATTCINCLSRV—GATTC 发现包含服务 _____ 46

5.2.23. AT+BLEGATTCCHAR—GATTC 发现服务特征 _____ 47

5.2.24. AT+BLEGATTCRD—GATTC 读取服务特征值 _____ 48

5.2.25. AT+BLEGATTCWR—GATTC 写服务特征值 _____ 49

5.3. BLE AT CMD Error Code _____ 50

6. AT 指令使用示例 _____ 51

6.1. 单连接 TCP 客户端 _____ 51

6.2. UDP 传输 _____ 53

6.2.1. 固定远端的 UDP 通信 _____ 54

6.2.2. 远端可变的 UDP 通信 _____ 55

6.3. 多连接 TCP 服务器 _____ 56

1. 介绍

1.1. 文档应用范围

本文档描述 OPL1000 AT 指令集功能以及使用方法。

指令集主要分为：基础 AT 指令、WIFI 功能 AT 指令、TCP/IP 功能 AT 指令、BLE 功能 AT 指令等。

OPL1000 AT 指令默认使用串口 UART1 传输，默认波特率为 115200bps，格式为 8N1。

1.2. 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。
- AT 指令不区分大小写。
- AT 指令以回车换行符结尾 \r\n。请注意设置串口工具为“新行模式”。

2. 基础 AT 指令

2.1. 基础 AT 指令一览表

| 指令 | 说明 |
|-----------------|------------------------------|
| AT | 测试 AT 模块 |
| 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 | 设置 UART 当前临时配置 |
| AT+UART_DEF | 设置 UART 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+MACADDRDEF | 设置设备来源 Mac Address，保存到 flash |
| AT+RFHP | 设置 RF Power |
| AT+READFLASH | 读取 flash |
| AT+WRITEFLASH | 写入 flash |
| AT+ERASEFLASH | 抹除 flash |
| AT+DHCPARPCHECK | 设置 DHCP ARP 检查机制 |

2.2. 基础 AT 指令描述

2.2.1. AT—测试 AT 模块

| | |
|------|----|
| 执行指令 | AT |
| 响应 | OK |
| 参数说明 | - |

2.2.2. AT+RST—重启模块

| | |
|------|-----------------|
| 执行指令 | AT+RST |
| 响应 | OK |
| 参数说明 | - |
| 注意 | 执行此指令后，系统会强制重启。 |

2.2.3. AT+GMR—查询版本信息

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

2.2.4. AT+GSLP—进入 Deep-sleep 模式

| | |
|------|--|
| 设置指令 | AT+GSLP=<time> |
| 响应 | <time> OK |
| 参数说明 | <time>：设置 OPL1000 的睡眠时长，单位：毫秒。OPL1000 会在休眠设定时长后自动唤醒。 |

2.2.5. ATE—开关回显功能

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

2.2.6. AT+RESTORE—恢复出厂设置

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

2.2.7. AT+UART_CUR—设置 UART 临时配置

| | | |
|----|--|---|
| 指令 | 查询指令： AT+UART_CUR? | 设置指令： AT+UART_CUR=<baudrate>,<databits>,<stopbits>,<parity>,<flow control> |
| 响应 | +UART_CUR:<baudrate>,<databits>,<stopbits>,<parity>,<flow control> OK | |

| | | |
|--|---|---|
| 指令 | 查询指令： AT+UART_CUR? | 设置指令： AT+UART_CUR=<baudrate>,<databits>,<stopbits>,<parity>,<flow control> |
| 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。 | | |
| 参数说明 | <ul style="list-style-type: none">• <baudrate>：UART 波特率• <databits>：数据位<ul style="list-style-type: none">▸ 5：5 bit 数据位▸ 6：6 bit 数据位▸ 7：7 bit 数据位▸ 8：8 bit 数据位• <stopbits>：停止位<ul style="list-style-type: none">▸ 1：1 bit 停止位▸ 2：1.5 bit 停止位▸ 3：2 bit 停止位• <parity>：校验位<ul style="list-style-type: none">▸ 0：None▸ 1：Odd▸ 2：Even• <flow control>：流控<ul style="list-style-type: none">▸ 0：不使能流控▸ 1：保留▸ 2：保留▸ 3：同时使能 RTS 和 CTS | |
| 注意 | <ul style="list-style-type: none">• 本设置不保存到 flash。• 使用流控需要硬件支持。• 波特率支持范围：80 ~ 1000000 | |
| 示例 | AT+UART_CUR=115200,8,1,0,3 | |

2.2.8. AT+UART_DEF—设置 UART 配置 · 保存到 Flash

| | | |
|------|---|---|
| 指令 | 查询指令： AT+UART_DEF? | 设置指令： AT+UART_DEF=<baudrate>,<databits>,<stopbits>,<parity>,<flow control> |
| 响应 | +UART_DEF:<baudrate>,<data bits>,<stopbits>,<parity>,<flow control> OK | |
| 参数说明 | <ul style="list-style-type: none">• <baudrate>：UART 波特率• <databits>：数据位<ul style="list-style-type: none">▸ 5：5 bit 数据位▸ 6：6 bit 数据位▸ 7：7 bit 数据位▸ 8：8 bit 数据位• <stopbits>：停止位<ul style="list-style-type: none">▸ 1：1 bit 停止位▸ 2：1.5 bit 停止位▸ 3：2 bit 停止位• <parity>：校验位<ul style="list-style-type: none">▸ 0：None▸ 1：Odd▸ 2：Even• <flow control>：流控<ul style="list-style-type: none">▸ 0：不使能流控▸ 1：保留▸ 2：保留▸ 3：同时使能 RTS 和 CTS | |
| 注意 | <ul style="list-style-type: none">• 本设置将保存在到 flash · 重新上电后仍生效。• 使用流控需要硬件支持。• 波特率支持范围：80 ~ 1000000 | |

2.2.9. AT+SLEEP—设置 sleep 模式

2.2.10. AT+SYSRAM—查询当前剩余 RAM 大小

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

2.2.11. AT+MACADDRDEF—设置设备来源 Mac Address · 保存到 flash

| | | |
|------|---|---------------------------------------|
| 指令 | 查询指令： AT+MACADDRDEF? | 设置指令： AT+MACADDRDEF=<iface>,<type> |
| 响应 | +MACADDRDEF:<iface_0>,<type> OK <iface_1>,<type> OK | |
| 参数说明 | <ul style="list-style-type: none">• <iface>：设备介面<ul style="list-style-type: none">▸ 0：Wifi Station▸ 1：BLE• <type>：MAC address 来源设置<ul style="list-style-type: none">▸ 0：From OTP▸ 1：From Flash | |
| 注意 | <ul style="list-style-type: none">• 本设置将保存在到 flash · 重新上电后仍生效。• 默认配置来源 MAC address 为 OTP。• 設置成功後，需重新上電使 MAC address 配置生效 | |
| 示例 | AT+MACADDRDEF=0,1 | |

2.2.12. AT+RFHP—设置 RF Power

| | | |
|------|--|-----------------------------|
| 指令 | 查询指令： AT+RFHP? | 设置指令： AT+RFHP=<rf_power> |
| 响应 | +RFHP:<rf_power> OK | OK |
| 参数说明 | <ul style="list-style-type: none">• <rf_power>：RF 功率<ul style="list-style-type: none">▸ 0：Wi-Fi LPA and BLE LPA▸ 15：Wi-Fi LPA and BLE HPA▸ 240：Wi-Fi HPA and BLE LPA▸ 255：Wi-Fi HPA and BLE HPA | |
| 注意 | <ul style="list-style-type: none">• 本设置将保存在到 flash，重新上电后仍生效。• 默认配置 240：Wi-Fi HPA and BLE LPA。• 設置成功後，需重新上電使配置生效 | |
| 示例 | AT+RFHP=240 | |

2.2.13. AT+READFLASH—读取 Flash

| | | |
|------|--|--|
| 指令 | 查询指令： N/A | 设置指令： AT+READFLASH=<address>,<number_of_bytes> > |
| 响应 | N/A | 成功： <byte_0>,<byte_1>,<byte_2>,...,<byte_n> OK 失败：ERROR |
| 参数说明 | <ul style="list-style-type: none">• <address>：Flash 位址（十六进制）。▸ 允許讀取範圍為 0x00000000 ~ 0x000FFFFF• <number_of_bytes>：欲讀取的 byte 數量（十进制）。 | |
| 注意 | <ul style="list-style-type: none">• 不可超出允許範圍。 | |

| | | |
|----|--|---|
| 指令 | 查詢指令： N/A | 設置指令： AT+READFLASH=<address>,<number_of_bytes> |
| 示例 | AT+READFLASH=0x000F8000,8 FF,FF,FF,FF,FF,FF,FF,FF OK | |

2.2.14. AT+WRITEFLASH—写入 Flash

| | | |
|----|--------------|---|
| 指令 | 查詢指令： N/A | 設置指令： AT+WRITEFLASH=<address>,<number_of_bytes>,<byte_0>,<byte_1>,...,<byte_n> |
| 响应 | N/A | 成功：OK 失敗：ERROR |

- 参数说明
- <address>：Flash 位址（十六进制）。
 - 允許寫入範圍為 0x000F8000 ~ 0x000FFFFF
 - <number_of_bytes>：欲寫入的 byte 數量（十进制）。
 - <byte_0>~<byte_n>：写入值（十六进制）。

- 注意
- 不可超出允許範圍。
 - 寫入前建議請先執行 AT+READFLASH，確認欲寫入的 Flash 範圍均處於可寫入狀態（全部值均為 0xFF）；若不是，請先執行 AT+ERASEFLASH 抹除包含欲寫入範圍的 sector。
 - 由於 AT command 最多能輸入 255 個字元，依示例格式，一次最多可分別指定寫入 76 bytes。
 - 若 <byte_0>~<byte_n> 少於 <number_of_bytes> 所指定的數量，會持續寫入 <byte_n> 直到寫滿 <number_of_bytes> 為止，可藉此方式將指定範圍全部寫為同一個值。

示例

```
1.
AT+WRITEFLASH=0x000F8000,4,F0,F1,F2,F3
OK
AT+READFLASH=0x000F8000,8
F0,F1,F2,F3,FF,FF,FF,FF
OK
```

| | | |
|----|--|---|
| 指令 | 查詢指令： N/A | 設置指令： AT+WRITEFLASH=<address>,<number_of_bytes>,<byte_0>,<byte_1>,...,<byte_n> |
| | 2. AT+WRITEFLASH=0x000F9000,8,A0 OK AT+READFLASH=0x000F9000,16 A0,A0,A0,A0,A0,A0,A0,A0,FF,FF,FF,FF,FF,FF,FF,FF OK | |

2.2.15. AT+ERASEFLASH—抹除 Flash

| | | |
|----|--------------|--|
| 指令 | 查詢指令： N/A | 設置指令： AT+ERASEFLASH=<start_address_of_sector>,<number_of_sectors> |
| 响应 | N/A | 成功：OK 失敗：ERROR |

参数说明

- <start_address_of_sector>：sector 起始位址（十六进制）。
- 允許值為 0x000F8000, 0x000F9000, 0x000FA000, ..., 或 0x000FF000。
- <number_of_sectors>：欲抹除的 sector 數量（十进制）。
- 指定從 sector 起始位址開始，要一次抹除多少 sectors。
- 單一 sector 大小為 4096 bytes。

注意

- 以 sector 為單位抹除 Flash，一次最少抹除 4096 bytes。
- 允許抹除的 sectors 如下，共 8 個。
- 0x000F8000 ~ 0x000F8FFF
- 0x000F9000 ~ 0x000F9FFF
- 0x000FA000 ~ 0x000FAFFF
- 0x000FB000 ~ 0x000FBFFF
- 0x000FC000 ~ 0x000FCFFF
- 0x000FD000 ~ 0x000FDFFF
- 0x000FE000 ~ 0x000FEFFF
- 0x000FF000 ~ 0x000FFFFF

| | | |
|----|---|---|
| 指令 | 查询指令： N/A | 设置指令： AT+ERASEFLASH= <start_address_of_sector>,<number_of_sectors> |
| 示例 | <div>1.</div> <div>抹除 0x000F9000 ~ 0x000F9FFF : 4096 bytes</div> <div>AT+ERASEFLASH=0x000F9000,1</div> <div>OK</div> <div>2.</div> <div>抹除 0x000F8000 ~ 0x000F9FFF : 8192 bytes</div> <div>AT+ERASEFLASH=0x000F8000,2</div> <div>OK</div> | |

2.2.16. AT+DHCPARPCHK—设置 DHCP ARP 检查机制

| | | |
|------|--|------------------------------------|
| 指令 | 查询指令： AT+DHCPARPCHK? | 设置指令： AT+DHCPARPCHK= <chk_mode> |
| 响应 | +DHCPARPCHK:<chk_mode> OK | OK |
| 参数说明 | <div>• < chk_mode > :</div> <div>0 : Disable</div> <div>1 : Enable</div> | |
| 注意 | <div>•本设置将保存在到 flash , 重新上电后仍生效。</div> <div>•默认配置为 Enable。</div> <div>•设置完后无需重新上电 , 重新断线连线即生效。</div> | |
| 示例 | <div>AT+DHCPARPCHK=0</div> <div>OK</div> | |

3. WIFI 功能 AT 指令

3.1. WIFI 功能 AT 指令一览表

| 指令 | 说明 |
|---------------|-------------------|
| AT+CWMODE | 设置 WIFI 模式 |
| AT+CWJAP | 连接 AP |
| AT+CWLAPOPT | 设置 CWLAP 指令的属性 |
| AT+CWLAP | 扫描当前可用的 AP |
| AT+CWQAP | 断开与 AP 连接 |
| AT+CWDHCP | 设置 DHCP |
| AT+CWAUTOCONN | 上电是否自动连接 AP |
| AT+CIPSTAMAC | 设置 STA 接口的 MAC 地址 |
| AT+CIPSAT | 设置 STA 的 IP 地址 |
| AT+CWHOSTNAME | 设置 STA 的主机地址 |

3.2. WIFI 功能 AT 指令描述

3.2.1. AT+CWMODE—设置 Wi-Fi 模式

| | | | |
|----------|--|--|--|
| 指令 | 测试指令： AT+CWMODE=? | 查询指令： AT+CWMODE? 功能：查询 OPL1000 当前 Wi-Fi 模式。 | 设置指令： AT+CWMODE=<mode> 功能：设置 OPL1000 当前 Wi-Fi 模式。 |
| 响应 | 如果 Wi-fi 未初始化，则查询 返回： +CWMODE: 0 OK 如果 Wi-fi 已初始化，则查询 返回： +CWMODE: 1 OK | +CWMODE:<mode> OK | OK |
| 参数 说明 | <mode>： ▸ 0：无 Wi-fi 模式 ▸ 1：Station 模式 | | |
| 注意 | •在使用 WIFI 与 TCPIP 相关的 AT CMD 之前，请先使用 AT+CWMODE 设定 station 模式。 •Wi-fi 初始化以后，如需切换模式，需要调用 AT+RST 重启后，重新初始化设置。 •本指令目前仅支持 station 模式。 | | |
| 示例 | AT+CWMODE=1 | | |

3.2.2. AT+CWJAP—连接 AP

| | | |
|------|---|--|
| 指令 | 查询指令： AT+CWJAP? 功能：查询 OPL1000 Station 已连接的 AP 信息。 | 设置指令： AT+CWJAP=<ssid>,<pwd>[,<bssid>] 功能：设置 OPL1000 Station 需连接的 AP。 |
| 响应 | +CWJAP:<ssid>,<bssid>,<channel>,<rssi> OK | OK 或者 +CWJAP:<error code> ERROR |
| 参数说明 | <ul style="list-style-type: none">• <ssid>：字符串参数，AP 的 SSID• <bssid>：AP 的 MAC 地址• <channel>：信道号• <rssi>：信号强度 | <ul style="list-style-type: none">• <ssid>：目标 AP 的 SSID• <pwd>：密码最长 64 字节 ASCII• [<bssid>]：目标 AP 的 MAC 地址，一般用于有多个 SSID 相同的 AP 的情况• <error code>：（仅供参考，并不可靠）<ul style="list-style-type: none">▸ 1：连接超时▸ 2：密码错误▸ 3：找不到目标 AP▸ 4：连接失败▸ 其他值：未知错误 <p>参数设置需要开启 Station 模式，若 SSID 或者 password 中含有特殊符号时，例如，“或者 \”，需要进行转义，其它字符转义无效。</p> |
| 提示信息 | // If OPL1000 station connects to an AP, it will prompt messages: WIFI CONNECTED WIFI GOT IP // If the WiFi connection ends, it will prompt messages: WIFI DISCONNECT | |
| 注意 | - | |
| 示例 | AT+CWJAP="abc","0123456789" 例如，目标 AP 的 SSID 为 "ab\c"，password 为 "0123456789\"，则指令如下： | |

| | | |
|---|---|--|
| 指令 | 查询指令： AT+CWJAP? 功能：查询 OPL1000 Station 已连接的 AP 信息。 | 设置指令： AT+CWJAP=<ssid>,<pwd>[,<bssid>] 功能：设置 OPL1000 Station 需连接的 AP。 |
| <hr/> | | |
| AT+CWJAP="ab\\c","0123456789\\" | | |
| 如果有多个 AP 的 SSID 均为 "abc"，可通过 BSSID 确定目标 AP： | | |
| AT+CWJAP="abc","0123456789","ca:d7:19:d8:a6:44" | | |

3.2.3. 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 | <ul style="list-style-type: none">• <sort_enable>：指令 AT+CWLAP 的扫描结果是否按照信号强度 RSSI 值排序：<ul style="list-style-type: none">▸ 0：不排序▸ 1：根据 RSSI 排序• <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：<ul style="list-style-type: none">▸ bit 0：设置 AT+CWLAP 的扫描结果是否显示 <ecn>▸ bit 1：设置 AT+CWLAP 的扫描结果是否显示 <ssid>▸ bit 2：设置 AT+CWLAP 的扫描结果是否显示 <rssi>▸ bit 3：设置 AT+CWLAP 的扫描结果是否显示 <mac>▸ bit 4：设置 AT+CWLAP 的扫描结果是否显示 <channel> |
| 示例 | <p>AT+CWLAPOPT=1,31</p> <p>第一个参数为 1，表示后续如果使用 AT+CWLAP 指令，扫描结果将按照信号强度 RSSI 值排序；</p> <p>第二个参数为 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，</p> <p>扫描结果将显示所有参数。</p> |

3.2.4. AT+CWLAP—扫描当前可用的 AP

| | |
|------|--|
| 执行指令 | AT+CWLAP |
| | 功能：列出当前可用的 AP。 |
| 响应 | +CWLAP:<ecn>,<ssid>,<rssi>,<mac>,<channel> OK |
| 参数说明 | <ul style="list-style-type: none">• <ecn>：加密方式<ul style="list-style-type: none">▸ 0：OPEN▸ 1：WEP |

| | |
|------|---|
| 执行指令 | AT+CWLAP |
| | 功能：列出当前可用的 AP。 |
| | <ul style="list-style-type: none">▸ 2：WPA_PSK▸ 3：WPA2_PSK▸ 4：WPA_WPA2_PSK▸ 5：WPA2_Enterprise (目前 AT 不支持连接这种加密 AP)• <ssid>：字符串参数，AP 的 SSID• <rssi>：信号强度• [<mac>] (选填参数)：字符串参数，AP 的 MAC 地址• [<channel>] (选填参数)：信道号 |
| 示例 | AT+CWLAP="WiFi","ca:d7:19:d8:a6:44",6 或者查找指定 SSID 的 AP： AT+CWLAP="WiFi" 若查到超过一台指定 AP ‘WiFi’，则会把所有 SSID 中带有 ‘WiFi’ 的 AP 都找出来，方便查找 |

3.2.5. AT+CWQAP—断开与 AP 的连接

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

3.2.6. AT+CWDHCP—设置 DHCP - TBD

| | | |
|----|------------------------|---|
| 指令 | 查询指令： AT+CWDHCP? | 设置指令： AT+CWDHCP= <operate>,<mode> 功能：设置 DHCP。 |
| 响应 | +CWDHCP:<enable> OK | OK |

| | | |
|------|---|--|
| 指令 | 查询指令： AT+CWDHCP? | 设置指令： AT+CWDHCP=<operate>,<mode> 功能：设置 DHCP。 |
| 参数说明 | <enable>: DHCP 是否使能 • Bit0 : ▸ 0 : Station DHCP 关闭 ▸ 1 : Station DHCP 开启 | • <operate> : ▸ 0 : 关闭 ▸ 1 : 开启 • <mode> : ▸ Bit0 : Station DHCP |
| 注意 | 目前是 DHCP only 模式，必须通过 DHCP 获取 IP 地址。 | |
| 示例 | AT+CWDHCP=1,1 使能 Station DHCP。 | |

3.2.7. AT+CWAUTOCONN—上电是否自动连接 AP

| | |
|------|--|
| 执行指令 | AT+CWAUTOCONN= <enable> |
| 响应 | OK |
| 参数说明 | <p><enable> :</p> <ul style="list-style-type: none"> ▸ 0 : 上电不自动连接 AP ▸ 1 : 上电自动连接 AP <p>OPL000 Station 默认上电自动连接 AP。</p> |
| 注意 | 本设置保存在 flash。 |
| 示例 | AT+CWAUTOCONN=1 |

3.2.8. AT+CIPSTAMAC—设置 OPL1000 Station 接口的 MAC 地址

| | | |
|----|--|--|
| 指令 | 查询指令： AT+CIPSTAMAC? 功能：查询 OPL1000 Station 的 MAC 地址。 | 设置指令： AT+CIPSTAMAC= <mac> 功能：设置 OPL1000 Station 的 MAC 地址。 |
| 响应 | +CIPSTAMAC:<mac> | OK |

| | | |
|------|---|---------------------------------|
| 指令 | 查询指令： AT+CIPSTAMAC? | 设置指令： AT+CIPSTAMAC=<mac> |
| | 功能：查询 OPL1000 Station 的 MAC 地址。 | 功能：设置 OPL1000 Station 的 MAC 地址。 |
| OK | | |
| 参数说明 | <mac>：字符串参数，OPL1000 Station 的 MAC 地址 | |
| 注意 | <ul style="list-style-type: none">• 本设置保存到 flash。可以用 AT+MACADDRDEF=0,1 命令设置 MAC 源为 Flash• 如果固件工作在“上电自动连接 AP”模式，则不能修改 Station MAC 地址。可以用 AT+CWAUTOCONN=0 命令关闭“上电自动连接 AP”• 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:..."。• FF:FF:FF:FF:FF:FF 和 00:00:00:00:00:00 为非法 MAC，无法进行设置。• 为使修改的 MAC address 有效，需要使用 AT+RST 进行复位。 | |
| 示例 | AT+CIPSTAMAC="18:fe:35:98:d3:7b" | |

3.2.9. AT+CWHOSTNAME—设置 Station 的主机名称

| | | |
|------|---|-----------------------------------|
| 指令 | 查询指令： AT+CWHOSTNAME? | 设置指令： AT+CWHOSTNAME=<hostname> |
| | 功能：查询 OPL1000 Station 的主机名称。 | 功能：设置 OPL1000 Station 的主机名称。 |
| 响应 | +CWHOSTNAME:<host name> | 如果成功，返回 |
| | OK | OK |
| | 如果未使能 OPL1000 Station 模式，则返回 | 如果未使能 OPL1000 station 模式，则提示 |
| | +CWHOSTNAME:<null> | ERROR |
| OK | | |
| 参数说明 | <hostname>：主机名称，最长支持 32 字节 | |
| 注意 | <ul style="list-style-type: none">• 本设置不保存到 Flash，重启后将恢复默认值。• OPL1000 Station 默认的主机名称为“opulink”。 | |
| 示例 | AT+CWMODE=1 | |

| | | |
|-------------------------|------------------------------|------------------------------|
| 指令 | 查询指令： | 设置指令： |
| | AT+CWHOSTNAME? | AT+CWHOSTNAME=<hostname> |
| | 功能：查询 OPL1000 Station 的主机名称。 | 功能：设置 OPL1000 Station 的主机名称。 |
| AT+CWHOSTNAME="my_test" | | |

4. TCP/IP 功能 AT 指令

4.1. TCP/IP 功能 AT 指令一览表

| 指令 | 说明 |
|--------------|----------------------|
| AT+CIPSTATUS | 查询网络连接信息 |
| AT+CIPDOMAIN | 域名解析功能 |
| AT+CIPSTART | 建立 TCP 连接或 UDP 传输 |
| AT+CIPSEND | 发送数据 |
| AT+CIPSENDEX | 发送数据 |
| AT+CIPCLOSE | 关闭 TCP/UDP 传输 |
| AT+CIFSR | 查询本地 IP 地址 |
| AT+CIPMUX | 设置多连接 |
| AT+CIPSERVER | 建立 TCP 服务器 |
| AT+CIPSTO | 设置 TCP 服务器超时时间 |
| AT+CIPDINFO | 接收网络数据是否提示对端 IP 和端口号 |
| +IPD | 接收网络数据 |
| AT+PING | PING 功能 |

4.2. TCP/IP 功能 AT 指令描述

4.2.1. AT+CIPSTATUS—查询网络连接信息

| | |
|------|--|
| 执行指令 | AT+CIPSTATUS |
| 响应 | STATUS:<stat> +CIPSTATUS:<link ID>,<type>,<remote IP>,<remote port>,<local port>,<tetype> |
| 参数说明 | • <stat> : OPL1000 Station 接口的状态 |

执行指令 AT+CIPSTATUS

- 2：OPL1000 Station 已连接 AP，获得 IP 地址
- 3：OPL1000 Station 已建立 TCP 或 UDP 传输
- 4：OPL1000 Station 断开网络连接
- 5：OPL1000 Station 未连接 AP
- <link ID>：网络连接 ID (0 ~ 4)，用于多连接的情况
- <type>：字符串参数，"TCP" 或者 "UDP"
- <remote IP>：字符串，远端 IP 地址
- <remote port>：远端端口值
- <local port>：OPL1000 本地端口值
- <tetype>：
- 0：OPL1000 作为客户端
- 1：OPL1000 作为服务器

4.2.2. AT+CIPDOMAIN—域名解析功能

执行指令 AT+CIPDOMAIN=<domain name>

响应 +CIPDOMAIN:<IP address>
OK
或者
ERROR

参数说明 <domain name>：待解析的域名

示例 AT+CWMODE=1 // set Station mode
AT+CWJAP="SSID","password" // access to the internet
AT+CIPDOMAIN="www.baidu.com" // DNS function

4.2.3. AT+CIPSTART—建立 TCP 连接或 UDP 传输

■ 建立 TCP 连接

| | | |
|------|--|---|
| 设置指令 | TCP 单连接 (AT+CIPMUX=0) 时 : | TCP 多连接 (AT+CIPMUX=1) 时 : |
| | AT+CIPSTART=<type>,<remote IP>,<remote port>[,<TCP keep alive>] | AT+CIPSTART=<link ID>,<type>,<remote IP>,<remote port>[,<TCP keep alive>] |
| 响应 | OK | |
| 参数说明 | <ul style="list-style-type: none">• <link ID> : 网络连接 ID (0 ~ 4) , 用于多连接的情况• <type> : 字符串参数 , 连接类型 , "TCP" , "UDP"或 "SSL"• <remote IP> : 字符串参数 , 远端 IP 地址• <remote port> : 远端端口号• [<TCP keep alive>] : 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<ul style="list-style-type: none">▸ 0 : 关闭 TCP keep-alive 功能▸ 1 ~ 7200 : 侦测时间 , 单位为 1s | |
| 提示信息 | // If the TCP connection is established, it will prompt message as below [<link ID>,) CONNECT // If the TCP connection ends, it will prompt message as below [<link ID>,) CLOSED | |
| 注意 | 建议创建 TCP 连接时 , 开启 keep-alive 功能。 | |
| 示例 | AT+CIPSTART="TCP","192.168.101.110",1000 | |

■ 建立 UDP 传输

| | | |
|------|--|--|
| 设置指令 | 单连接模式 (AT+CIPMUX=0) 时 : | 多连接模式 (AT+CIPMUX=1) 时 : |
| | AT+CIPSTART=<type>,<remote IP>,<remote port>[,<UDP local port>),(<UDP mode>)] | AT+CIPSTART=<link ID>,<type>,<remote IP>,<remote port>[,<UDP local port>,<UDP mode>] |
| 响应 | OK | |
| 参数说明 | <ul style="list-style-type: none">• <link ID> : 网络连接 ID (0 ~ 4) , 用于多连接的情况 | |

| | | | |
|------|--|---|--|
| 设置指令 | | 单连接模式 (AT+CIPMUX=0) 时 : | 多连接模式 (AT+CIPMUX=1) 时 : |
| | | AT+CIPSTART=<type>,<remote IP>,<remote port>[,<UDP local port>],(<UDP mode>)] | AT+CIPSTART=<link ID>,<type>,<remote IP>,<remote port>[,<UDP local port>,<UDP mode>] |
| | | <ul style="list-style-type: none">• <type> : 字符串参数 , 连接类型 , "TCP" , "UDP"或 "SSL"• <remote IP> : 字符串参数 , 远端 IP 地址• <remote port> : 远端端口号• [<UDP local port>] : UDP 本地端口• [<UDP mode>] : UDP 传输的属性 , 若透传 , 则必须为 0<ul style="list-style-type: none">▸ 0 : 收到数据后 , 不更改远端目标 , 默认值为 0▸ 1 : 收到数据后 , 改变一次远端目标▸ 2 : 收到数据后 , 改变远端目标 <p>注意 :</p> <p>使用 <UDP mode> 必须先填写 <UDP local port> 。</p> | |
| 提示信息 | | // If the UDP transmission is established, it will prompt message as below [<link ID>,) CONNECT // If the UDP transmission ends, it will prompt message as below [<link ID>,) CLOSED | |
| 示例 | | AT+CIPSTART="UDP","192.168.101.110",1000,1002,2 | |

4.2.4. AT+CIPSEND—发送数据

| | | |
|------|------------|--|
| 设置指令 | | 1. 单连接时 : (+CIPMUX=0) AT+CIPSEND=<length> 2. 多连接时 : (+CIPMUX=1) AT+CIPSEND=<link ID>,<length> 3. 如果是 UDP 传输 , 可以设置远端 IP 和端口 : AT+CIPSEND=[<link ID>,<length>[,<remote IP>,<remote port>]]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。 | |

| | |
|------|--|
| 设置指令 | <p>1. 单连接时：(+CIPMUX=0)</p> <p>AT+CIPSEND=<length></p> <p>2. 多连接时：(+CIPMUX=1)</p> <p>AT+CIPSEND=<link ID>,<length></p> <p>3. 如果是 UDP 传输，可以设置远端 IP 和端口：</p> <p>AT+CIPSEND=[<link ID>,<length>,<remote IP>,<remote port>]功能：在普通传输模式时，设置发送数据的长度。</p> |
| | <p>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。如果未建立连接或连接被断开，返回：</p> <p>ERROR</p> <p>如果数据发送成功，返回：</p> <p>SEND OK</p> <p>如果数据发送失败，返回：</p> <p>SENDFAIL</p> |
| 参数说明 | <ul style="list-style-type: none">• <link ID>：网络连接 ID 号 (0 ~ 4)，用于多连接的情况• <length>：数字参数，表明发送数据的长度，最大长度为 2048• [<remote IP>]：UDP 传输可以设置对端 IP• [<remote port>]：UDP 传输可以设置对端端口 |
| 示例 | - |

4.2.5. AT+CIPSENDEX—发送数据

| | |
|----|---|
| 指令 | <p>设置指令：</p> <p>1. 单连接时：(+CIPMUX=0)</p> <p>AT+CIPSENDEX=<length></p> <p>2. 多连接时：(+CIPMUX=1)</p> <p>AT+CIPSENDEX=<link ID>,<length></p> <p>3. 如果是 UDP 传输，可以设置远端 IP 和端口：</p> <p>AT+CIPSENDEX=[<link ID>,<length>,<remote IP>,<remote port>]</p> <p>指令功能：在普通传输模式时，设置发送数据的长度。</p> |
| 响应 | 发送指定长度的数据。 |

| | | |
|----------------------------------|---|-----------------|
| 指令 | 设置指令： | |
| | 1. 单连接时：(+CIPMUX=0) | |
| | AT+CIPSENDEX=<length> | |
| | 2. 多连接时：(+CIPMUX=1) | |
| | AT+CIPSENDEX=<link ID>,<length> | |
| | 3. 如果是 UDP 传输，可以设置远端 IP 和端口： | |
| | AT+CIPSENDEX=[<link ID>,<length>,<remote IP>,<remote port>] | |
| | 指令功能：在普通传输模式时，设置发送数据的长度。 | |
| | | |
| | 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。 | |
| 如果未建立连接或连接被断开，返回： | | |
| ERROR | | |
| 如果数据发送成功，返回： | | |
| SEND OK | | |
| 如果数据发送失败，返回： | | |
| SENDERFAIL | | |
| 参数说明 | • <link ID>：网络连接 ID 号 (0 ~ 4)，用于多连接的情况 | |
| | • <length>：数字参数，表明发送数据的长度，最大长度为 2048 | |
| | • 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。 | |
| | • 用户如需发送 \0，请转义为 \\0。 | |
| 4.2.6. AT+CIPCLOSE—关闭 TCP/UDP 传输 | | |
| 指令 | 设置指令（用于多连接的情况）： | 执行指令（用于单连接的情况）： |
| | AT+CIPCLOSE=<link ID> | AT+CIPCLOSE |
| | 功能：关闭 TCP/UDP 传输。 | |
| 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 | |
| | [<link ID>,<length>] CLOSED | |

4.2.7. AT+CIFSR—查询本地 IP 地址

| | |
|------|--|
| 执行指令 | AT+CIFSR |
| 响应 | +CIFSR:STAIP,<Station IP address> +CIFSR:STAMAC,<Station MACaddress> OK |
| 参数说明 | <IP address> : OPL1000 Station 的 IP 地址 <MAC address> : OPL1000 Station 的 MAC 地址 |
| 注意 | OPL1000 Station IP 需连上 AP 后，才可以查询。 |

4.2.8. AT+CIPMUX—设置多连接

| | | |
|------|---|---|
| 指令 | 查询指令： AT+CIPMUX? | 设置指令： AT+CIPMUX= <mode> 功能：设置连接类型。 |
| 响应 | +CIPMUX:<mode> OK | OK |
| 参数说明 | <mode>： <ul style="list-style-type: none"> ▸ 0：单连接模式 ▸ 1：多连接模式 | |
| 注意 | <ul style="list-style-type: none"> • 默认为单连接； • 只有非透传模式 (AT+CIPMODE=0)，才能设置为多连接； • 必须在没有连接建立的情况下，设置连接模式； • 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 | |

4.2.9. AT+CIPSERVER—建立 TCP 服务器

| | | |
|------|---|---|
| 指令 | 查询指令： AT+CIPSERVER? | 设置指令： AT+CIPSERVER= <mode>[,<port>] 功能：设置服务器。 |
| 响应 | +CIPSERVER:<mode>,<port> OK | OK |
| 参数说明 | <mode>： ▸ 0：关闭服务器 ▸ 1：建立服务器 [<port>]：选填参数。端口号，默认为 333。 | |
| 提示信息 | // If the connection is established, it will prompt message as below [<link ID>,) CONNECT // If the connection ends, it will prompt message as below [<link ID>,) CLOSED | |
| 注意 | • 多连接情况下 (AT+CIPMUX=1)，才能开启服务器。 • 创建服务器后，自动建立服务器监听。 • 当有客户端接入，会自动占用一个连接 ID。 | |
| 示例 | • 建立 TCP 服务器 AT+CIPMUX=1 AT+CIPSERVER=1,80 | |

4.2.10. AT+CIPSTO—设置 TCP 服务器超时时间

| | | |
|------|---|--|
| 指令 | 查询指令： AT+CIPSTO? 功能：查询 TCP 服务器超时时间。 | 设置指令： AT+CIPSTO= <time> 功能：设置 TCP 服务器超时时间。 |
| 响应 | +CIPSTO:<time> OK | OK |
| 参数说明 | <time>：TCP 服务器超时时间，取值范围 0 ~ 7200s。 | |

| | | |
|----|--|---|
| 指令 | 查询指令： AT+CIPSTO? 功能：查询 TCP 服务器超时时间。 | 设置指令： AT+CIPSTO=<time> 功能：设置 TCP 服务器超时时间。 |
| 注意 | • OPL1000 作为 TCP 服务器，会断开一直不通信直至超时的 TCP 客户端连接。 • 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 AT+CIPSERVER=1,1001 AT+CIPSTO=10 | |

4.2.11. 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 | <mode>： |
| 说明 | ▸ 0：不显示对端 IP 和端口 ▸ 1：显示对端 IP 和端口 |
| 示例 | AT+CIPDINFO=1 |

4.2.12. +IPD—接收网络数据

| | | |
|------|--|---|
| 指令 | 单连接时： (+CIPMUX=0)+IPD,<len>[,<remote IP>,<remote port>]:<data> | 多连接时： (+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，OPL1000 接收到网络数据时向串口发送 +IPD 和数据。 • [<remote IP>]：网络通信对端 IP，由指令 AT+CIPDINFO=1 使能显示 • [<remote port>]：网络通信对端端口，由指令 AT+CIPDINFO=1 使能 • <link ID>：收到网络连接的 ID 号 • <len>：数据长度 • <data>：收到的数据 | |

4.2.13. AT+PING—Ping 功能

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

5. BLE 相关 AT 指令

5.1. BLE 指令一览表

| 指令 | 说明 |
|---------------------|---------------|
| 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+BLESCANRSPDATA | 设置 BLE 扫描回应 |
| 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+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 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+BLEGATTSSCHAR | GATTs 查询服务特征 |
| AT+BLEGATTSENTFY | GATTs 通知服务特征值 |
| AT+BLEGATTSSIND | GATTs 指示服务特征值 |
| AT+BLEGATTSSSETATTR | 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 |

5.2. BLE 指令描述

| | | |
|------|---|--|
| 指令 | 查询指令： AT+BLEINIT? 功能：查询 BLE 是否初始化。 | 设置指令： AT+BLEINIT=<init> 功能：设置 BLE 初始化角色。 |
| 响应 | 如果 BLE 未初始化，则查询返回 +BLEINIT:0 OK 如果 BLE 已初始化，则查询返回 +BLEINIT:<role> OK | OK |
| 参数说明 | <init>： ‣ 1：client role ‣ 2：server + client 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。 | |
| 示例 | AT+BLEINIT=1 | |

5.2.2. AT+BLEADDR—设置 BLE 设备地址

| | | |
|----|---|---|
| 指令 | 查询指令： | 设置指令： |
| | AT+BLEADDR? | AT+BLEADDR=<addr_type>,<random_addr> |
| | 功能：查询 BLE 设备的 public address。 | 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> OK | OK |
| 参数 | <addr_type>： | |
| 说明 | ▸ 0：public address ▸ 1：random address | |
| 注意 | <ul style="list-style-type: none">• 目前可设置/查询 public address，对 random address 仅支持设置。• 在设置 BLE public address 之前需要用 AT+BLEINIT=1 指令对 BLE 完成初始化操作• 为使修改的 BLE public address 有效，需要使用 AT+RST 进行复位。• 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,"08:7f:24:87:1c:f7" | |

5.2.3. AT+BLENAME—设置 BLE 设备名称

| | | |
|----|--|--------------------------|
| 指令 | 查询指令： | 设置指令： |
| | AT+BLENAME? | AT+BLENAME=<device_name> |
| | 功能：查询 BLE 设备名称。 | 功能：设置 BLE 设备名称。 |
| 响应 | +BLENAME:<device_name> OK | OK |
| 参数 | <device_name>：BLE 设备名称 | |
| 说明 | | |
| 注意 | <ul style="list-style-type: none">• 默认设备名称为“BLE_AT”。• 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。• 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="opl_demo" | |

5.2.4. AT+BLES SCAN_RSPDATA—设置 BLE 扫描响应

| | |
|------|---|
| 指令 | 设置指令： AT+BLESCANRSPDATA=<scan_rsp_data> 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" |
| 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 AT+BLESCANRSPDATA="1122334455" |

5.2.5. AT+BLEADVPARAM—设置广播参数

| | | |
|------|---|---|
| 指令 | <p>查询指令：</p> <p>AT+BLEADVPARAM?</p> <p>功能：查询广播参数。</p> | <p>设置指令：</p> <p>AT+BLEADVPARAM=<adv_int_min>,<adv_int_max>,<adv_type>,<own_addr_type>,<channel_map>,<adv_filter_policy>,<peer_addr_type>,<peer_addr>]</p> <p>功能：设置广播参数。</p> |
| 响应 | <p>+BLEADVPARAM:<adv_int_min>,<adv_int_max>,<adv_type>,<own_addr_type>,<channel_map>,<adv_filter_policy>,<peer_addr_type>,<peer_addr></p> <p>OK</p> | |
| 参数说明 | <p><adv_int_min>：最小广播间隔，取值范围：0x0020 ~ 0x4000</p> <p><adv_int_max>：最大广播间隔，取值范围：0x0020 ~ 0x4000</p> <p><adv_type>：广播类型</p> <p>▸ 0：ADV_TYPE_IND</p> | |

| | | |
|----|---|--|
| 指令 | <p>查询指令：</p> <p>AT+BLEADVPARAM?</p> <p>功能：查询广播参数。</p> | <p>设置指令：</p> <p>AT+BLEADVPARAM=<adv_int_min>,<adv_int_max>,<adv_type>,<own_addr_type>,<channel_map>[,<adv_filter_policy>,<peer_addr_type>,<peer_addr>]</p> <p>功能：设置广播参数。</p> |
| | <p>‣ 1: ADV_TYPE_DIRECT_IND_HIGH</p> <p>‣ 2 : ADV_TYPE_SCAN_IND</p> <p>‣ 3 : ADV_TYPE_NONCONN_IND</p> <p><own_addr_type>：BLE 地址类型</p> <p>‣ 0 : BLE_ADDR_TYPE_PUBLIC</p> <p>‣ 1 : BLE_ADDR_TYPE_RANDOM</p> <p><channel_map>：广播信道</p> <p>‣ 1 : ADV_CHNL_37</p> <p>‣ 2 : ADV_CHNL_38</p> <p>‣ 4 : ADV_CHNL_39</p> <p>‣ 7 : ADV_CHNL_ALL</p> <p>[<adv_filter_policy>] (选填参数)：过滤器规则</p> <p>‣ 0 : ADV_FILTER_ALLOW_SCAN_ANY_CON_ANY</p> <p>‣ 1 : ADV_FILTER_ALLOW_SCAN_WLST_CON_ANY</p> <p>‣ 2 : ADV_FILTER_ALLOW_SCAN_ANY_CON_WLST</p> <p>‣ 3 : ADV_FILTER_ALLOW_SCAN_WLST_CON_WLST</p> <p>[<peer_addr_type>] (选填参数)：对方 BLE 地址类型</p> <p>‣ 0 : PUBLIC</p> <p>‣ 1 : RANDOM</p> <p>[<peer_addr>] (选填参数)：对方 BLE 地址</p> | |
| 注意 | <p><adv_filter_policy>,<peer_addr_type>,<peer_addr> 三个参数要求同时缺省，或者同时设置。</p> | |
| 示例 | <p>AT+BLEINIT=2 // 初始化为 server</p> <p>AT+BLEADVPARAM=50,50,0,0,4,0,0,"12:34:45:78:66:88"</p> | |

5.2.6. AT+BLEADVDATA—设置 BLE 广播数据

| | |
|------|---|
| 指令 | 设置指令： AT+BLEADVDATA=<adv_data> 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 AT+BLEADVDATA="1122334455" |

5.2.7. AT+BLEADVSTART—开始 BLE 广播

| | |
|------|---|
| 指令 | 执行指令： AT+BLEADVSTART 功能：开始 BLE 广播。 |
| 响应 | OK |
| 参数说明 | 无 |
| 注意 | <ul style="list-style-type: none"> 若未设置广播参数 (AT+BLEADVPARAM=<adv_parameter>)，则使用默认广播参数； 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 AT+BLEADVSTART |

5.2.8. AT+BLEADVSTOP—结束 BLE 广播

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

| | |
|------|--|
| 指令 | 执行指令： AT+BLEADVSTOP 功能：结束 BLE 广播。 |
| 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 AT+BLEADVSTART AT+BLEADVSTOP |

5.2.9. AT+BLECONNPARAM—更新 BLE 连接参数

| | | |
|------|--|--|
| 指令 | 查询指令： AT+BLECONNPARAM? 功能：查询 BLE 连接参数。 | 设置指令： AT+BLECONNPARAM=<conn_index>,<min_interval>,<max_interval>,<latency>,<timeout> 功能：更新 BLE 连接参数。 |
| 响应 | +BLECONNPARAM:<conn_index>,<cur_interval>,<latency>,<timeout> OK | OK //指令已接收，将尝试更新连接参数 +BLECONNPARAM: <conn_index>,0 如果更新失败，将提示 +BLECONNPARAM: <conn_index>,-1 |
| 参数说明 | <conn_index>：BLE 连接号，当前只支持 index 为 0 的单连接 <min_interval>：最小连接间隔，取值范围：0x0006 ~ 0x0C80 <max_interval>：最大连接间隔，取值范围：0x0006 ~ 0x0C80 <cur_interval>：当前连接间隔 <latency>：时延，取值范围：0x0000 ~ 0x01F3 <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 AT+BLECONN=0,"24:0a:c4:09:34:23" // 建立 BLE 连接 AT+BLECONNPARAM=0,12,14,1,500 // 更新 BLE 连接参数 | |

5.2.10. AT+BLEDISCONN—断开 BLE 连接

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

5.2.11. AT+BLEDATALEN—设置 BLE 数据包长度

| | |
|------|--|
| 指令 | 设置指令： AT+BLEDATALEN=<conn_index>,<pkt_data_len> 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 <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 packet length。 |
| 示例 | AT+BLEINIT=1 // 初始化为 client AT+BLECONN=0,"24:0a:c4:09:34:23" AT+BLEDATALEN=0,30 |

5.2.12. AT+BLECFGMTU—设置 GATT MTU 的长度

| | | |
|----|---|--|
| 指令 | 设置指令： AT+BLECFGMTU? 功能：查询 GATT (Generic Attribute Profile) MTU 的长度。 | 设置指令： AT+BLECFGMTU=<conn_index>,<mtu_size> 功能：设置 GATT MTU 的长度。 |
| 响应 | +BLECFGMTU:<conn_index>,<mtu_size> OK // 指令已接收, OK | |
| 参数 | <conn_index>：BLE 连接号，当前只支持 index 为 0 的单连接 | |
| 说明 | <mtu_size>：BLE 最大传输单元的长度 | |
| 注意 | • 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 AT+BLECONN=0,"24:12:5f:9d:91:98"// 建立 BLE 连接 AT+BLECFGMTU=0,300 | |

5.2.13. AT+BLEGATTSSRVCRE—GATTS 创建服务

| | |
|------|--|
| 指令 | 执行指令： AT+BLEGATTSSRVCRE 功能：GATTS 创建服务。 |
| 响应 | OK |
| 参数说明 | 无 |
| 注意 | • OPL1000 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 AT+BLEGATTSSRVCRE |

5.2.14. AT+BLEGATTSSRVSTART—GATTS 开启服务

| | | |
|------|-----------------------------|----------------------------------|
| 指令 | 执行指令： | 设置指令： |
| | AT+BLEGATTSSRVSTART | AT+BLEGATTSSRVSTART= <srv_index> |
| | 功能：GATTS 开启全部服务。 | 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 | |
| | AT+BLEGATTSSRVCRE | |
| | AT+BLEGATTSSRVSTART | |
| | | |

5.2.15. AT+BLEGATTSSRVSTOP—GATTS 停止服务

| | | |
|------|-----------------------------|---------------------------------|
| 指令 | 执行指令： | 设置指令： |
| | AT+BLEGATTSSRVSTOP | AT+BLEGATTSSRVSTOP= <srv_index> |
| | 功能：GATTS 停止全部服务。 | 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 | |
| | AT+BLEGATTSSRVCRE | |
| | AT+BLEGATTSSRVSTART | |
| | AT+BLEGATTSSRVSTOP | |
| | | |

5.2.16. AT+BLEGATTSSRV—GATTS 发现服务

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

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

5.2.17. AT+BLEGATTCHAR—GATTS 发现服务特征

| | |
|------|--|
| 指令 | 查询指令： AT+BLEGATTCHAR? 功能：GATTS 发现服务特征。 |
| 响应 | //对于服务特征信息，显示如下： +BLEGATTCHAR:"char",<srv_index>,<char_index>,<char_uuid>,<char_prop> //对于描述符信息，显示如下： +BLEGATTCHAR:"desc",<srv_index>,<char_index>,<desc_index> OK |
| 参数说明 | <srv_index>：服务序号，从 1 起始递增 <char_index>：服务特征的序号，从 1 起始递增 <char_uuid>：服务特征的 UUID <char_prop>：服务特征的属性 <desc_index>：特征描述符序号 |

指令 查询指令：

AT+BLEGATTCHAR?

功能：GATTS 发现服务特征。

示例 AT+BLEINIT=2 // 初始化为 server

AT+BLEGATTSSRVCRE

AT+BLEGATTSSRVSTART

AT+BLEGATTCHAR?

5.2.18. AT+BLEGATTSNTFY—GATTS 通知服务特征值

指令 设置指令：

AT+BLEGATTSNTFY=<conn_index>,<srv_index>,<char_index>,<length>

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。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

 <srv_index>：服务序号，由指令 AT+BLEGATTCHAR? 查询可得

 <char_index>：服务特征的序号，由指令 AT+BLEGATTCHAR? 查询可得

 <length>：数据长度

示例 以下为 notify 的简单示例，

AT+BLEINIT=2 // 初始化为 server

AT+BLEGATTSSRVCRE

AT+BLEGATTSSRVSTART

AT+BLEADVSTART// 开始广播，等待 client 连接，并配置接收 notify

AT+BLEGATTCHAR?// 查询允许 notify 的特征

//例如，使用 3 号服务的 6 号特征通知长度为 4 的数据

AT+BLEGATTSNTFY=0,3,6,4

// 提示 > 符号后，输入 4 字节数据即可，例如 "1234"

5.2.19. AT+BLEGATTSIND—GATTS 指示服务特征值

| | |
|------|---|
| 指令 | 设置指令： AT+BLEGATTSIND=<conn_index>,<srv_index>,<char_index>,<length> 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。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 <srv_index>：服务序号，由指令 AT+BLEGATTSCHAR? 查询可得 <char_index>：服务特征的序号，由指令 AT+BLEGATTSCHAR? 查询可得 <length>：数据长度 |
| 示例 | 以下为 indicate 的简单示例， AT+BLEINIT=2 // 初始化为 server AT+BLEGATTSSRVCRE AT+BLEGATTSSRVSTART AT+BLEADVSTART// 开始广播，等待 client 连接，client 端连接后，应该设置为接收 indication AT+BLEGATTSCHAR?// 查询允许 indicate 的特征 //例如，使用 3 号服务的 7 号特征指示长度为 4 的数据 AT+BLEGATTSIND=0,3,7,4 // 提示 > 符号后，输入 4 字节数据即可，例如 "1234" |

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

| | |
|------|---|
| 指令 | 设置指令： AT+BLEGATTSSETATTR=<srv_index>,<char_index>[,<desc_index>],<length> 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。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 <char_index>：服务特征的序号，由 AT+BLEGATTSCHAR? 查询结果中获得 |

| | |
|----|---|
| 指令 | <p>设置指令：</p> <p>AT+BLEGATTSSSETATTR=<srv_index>,<char_index>[,<desc_index>],<length></p> <p>功能：GATTS 设置服务特征（描述符）值。</p> |
| | <p>[<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。</p> <p><length>：数据长度</p> |
| 注意 | <p><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 <length> 为 3 超过最大长度，则会报错。</p> |
| 示例 | <p>AT+BLEINIT=2 // 初始化为 server</p> <p>AT+BLEGATTSSRVCRE</p> <p>AT+BLEGATTSSRVSTART</p> <p>AT+BLEGATTSSCHAR?</p> <p>//例如，向 1 号服务的 1 号特征写入长度为 4 的数据</p> <p>AT+BLEGATTSSSETATTR=1,1,,4</p> <p>// 提示 > 符号后，输入 4 字节数据即可，例如 "1234"</p> |

5.2.21. AT+BLEGATTCPRIMSRV—GATTC 发现基本服务

| | |
|------|--|
| 指令 | <p>设置指令：</p> <p>AT+BLEGATTCPRIMSRV=<conn_index></p> <p>功能：GATTC 发现基本服务。</p> |
| 响应 | <p>+BLEGATTCPRIMSRV:<conn_index>,<srv_index>,<srv_uuid>,<srv_type></p> <p>OK</p> |
| 参数说明 | <p><conn_index>：BLE 连接号，当前只支持 index 为 0 的单连接</p> <p><srv_index>：服务发现结果序号，从 1 起始递增</p> <p><srv_uuid>：服务的 UUID</p> <p><srv_type>：服务的类型</p> <ul style="list-style-type: none">▸ 0：次要服务▸ 1：首要服务 |
| 注意 | <p>使用本指令，需要先建立 BLE 连接。</p> |
| 示例 | <p>AT+BLEINIT=1 // 初始化为 client</p> |

| | |
|---|---|
| 指令 | 设置指令： AT+BLEGATTCPRIMSRV=<conn_index> 功能：GATTC 发现基本服务。 |
| | AT+BLECONN=0,"24:12:5f:9d:91:98"// 建立 BLE 连接 AT+BLEGATTCPRIMSRV=0 |
| 5.2.22. AT+BLEGATTCINCLSRV—GATTC 发现包含服务 | |
| 指令 | 设置指令： AT+BLEGATTCINCLSRV= <conn_index>,<srv_index> 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> OK |
| 参数说明 | <conn_index>：BLE 连接号，当前只支持 index 为 0 的单连接 <srv_index>：服务发现结果序号，由 AT+BLEGATTCPRIMSRV=<conn_index> 查询结果中获得 <srv_uuid>：服务的 UUID <srv_type>：服务的类型 ▸ 0：次要服务 ▸ 1：首要服务 <included_srv_uuid>：包含服务的 UUID <included_srv_type>：包含服务的类型 ▸ 0：次要服务 ▸ 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 AT+BLECONN=0,"24:12:5f:9d:91:98"// 建立 BLE 连接 AT+BLEGATTCPRIMSRV=0 AT+BLEGATTCINCLSRV=0,1//根据前一条指令的查询结果，指定 index 查询 |

5.2.23. AT+BLEGATTCCHAR—GATTC 发现服务特征

| | |
|------|--|
| 指令 | 设置指令： AT+BLEGATTCCHAR=<conn_index>,<srv_index> 功能：GATTC 发现服务特征。 |
| 响应 | //对于服务特征信息，显示如下： +BLEGATTCCHAR:"char",<conn_index>,<srv_index>,<char_index>,<char_uuid>,<char_prop> //对于描述符信息，显示如下： +BLEGATTCCHAR:"desc",<conn_index> , <srv_index>,<char_index>,<desc_index>,<desc_uuid> OK |
| 参数说明 | <conn_index>：BLE 连接号，当前只支持 index 为 0 的单连接 <srv_index>：服务发现结果序号，由 AT+BLEGATTCPRIMSRV=<conn_index> 查询结果中获得 <char_index>：服务特征的序号，从 1 起始递增 <char_uuid>：服务特征的 UUID <char_prop>：服务特征的属性 <desc_index>：特征描述符序号 <desc_uuid>：特征描述符的 UUID |
| 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 AT+BLECONN=0,"24:12:5f:9d:91:98"// 建立 BLE 连接 AT+BLEGATTCPRIMSRV=0 AT+BLEGATTCCHAR=0,1//根据前一条指令的查询结果，指定 index 查询 |

5.2.24. AT+BLEGATTCRD—GATTC 读取服务特征值

| | |
|------|---|
| 指令 | 设置指令： AT+BLEGATTCRD=<conn_index>,<srv_index>,<char_index>[,<desc_index>] 功能：GATTC 读取服务特征（描述符）值。 |
| 响应 | +BLEGATTCRD:<conn_index>,<len>,<value> OK |
| 参数说明 | <p><conn_index>：BLE 连接号，当前只支持 index 为 0 的单连接</p> <p><srv_index>：服务发现结果序号，由 AT+BLEGATTCPRIMSRV=<conn_index> 查询结果中获得</p> <p><char_index>：服务特征的序号，由 AT+BLEGATTCCHAR=<conn_index>,<srv_index> 查询结果中获得 [<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。</p> <p><len>：数据长度</p> <p><value>：HEX 字符串</p> <p>▸ 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 字符串 "0x30"。</p> <p>▸ 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 字符串 "0x30 0x31 0x32 0x33"。</p> |
| 注意 | <ul style="list-style-type: none">• 使用本指令，需要先建立 BLE 连接。• 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 AT+BLECONN=0,"24:12:5f:9d:91:98"// 建立 BLE 连接 AT+BLEGATTCPRIMSRV=0 AT+BLEGATTCCHAR=0,3//根据前一条指令的查询结果，指定 index 查询 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 |

5.2.25. AT+BLEGATTCWR—GATTC 写服务特征值

| | |
|------|--|
| 指令 | <p>设置指令：</p> <p>AT+BLEGATTCWR=<conn_index>,<srv_index>,<char_index>[,<desc_index>],<length></p> <p>功能：GATTC 写服务特征（描述符）值。</p> |
| 响应 | <p>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。若写操作成功，则提示 OK</p> |
| 参数说明 | <p><conn_index>：BLE 连接号，当前只支持 index 为 0 的单连接</p> <p><srv_index>：服务发现结果序号，由 AT+BLEGATTCPRIMSRV=<conn_index> 查询结果中获得</p> <p><char_index>：服务特征的序号，由</p> <p>AT+BLEGATTCCHAR=<conn_index>,<srv_index> 查询结果中获得</p> <p>[<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。</p> <p><length>：数据长度</p> |
| 注意 | <ul style="list-style-type: none">• 使用本指令，需要先建立 BLE 连接。• 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 | <p>AT+BLEINIT=1 // 初始化为 client</p> <p>AT+BLECONN=0,"24:12:5f:9d:91:98"// 建立 BLE 连接</p> <p>AT+BLEGATTCPRIMSRV=0</p> <p>AT+BLEGATTCCHAR=0,3//根据前一条指令的查询结果，指定 index 查询</p> <p>// 例如，向第 3 号服务的第 4 号特征，写入长度为 6 的数据</p> <p>AT+BLEGATTCWR=0,3,4,,6</p> <p>// 提示 > 后，通过串口输入数据 "123456" 即可</p> |

5.3. BLE AT CMD Error Code

| Code | Description |
|------|--|
| 1 | BLE is not initialized |
| 2 | The memory is not enough |
| 3 | No such command |
| 4 | Invalid parameter |
| 5 | Invalid state |
| 6 | Command is in progress |
| 7 | Fail |
| 8 | Already (in the wanted state) |
| 9 | Wrong role |
| 10 | Busy |
| 11 | No random address |
| 12 | No peer address |
| 13 | The number of connections is out of max (only one connection is supported) |
| 14 | Service does not start |
| 15 | Invalid characteristic property |
| 16 | No GATT service |
| 17 | No GATT include service |
| 18 | No GATT characteristic |
| 19 | No GATT characteristic descriptor |
| 20 | No read permission |
| 21 | No write permission |
| 22 | GATT read fail |
| 23 | GATT write fail |
| 24 | Invalid characteristic value length |

6. AT 指令使用示例

本章介绍几种常见的 AT 指令使用示例。

6.1. 单连接 TCP 客户端

- 1 · 设置 Wi-Fi 模式：

```
AT+CWMODE=1      // Station mode
```

响应：

OK

- 2 · 扫描 AP：

```
AT+CWLAP
```

响应：

```
+CWLAP:2,Opulinks,-22,30:fc:68:90:a8:a1,1
```

```
+CWLAP:3,Opulinks-S,-24,00:d0:41:df:1e:25,6
```

OK

- 3 · 连接路由：

```
AT+CWJAP="SSID","password"
```

响应：

OK

WIFI CONNECTED

WIFI GOT IP

- 4 · 查询设备 IP 信息：

```
AT+CIFSR
```

响应：

```
+CIFSR:STAIP,"169.254.119.102"
```

```
+CIFSR:STAMAC,"22:33:44:55:66:76"
```

OK

- 5 · 设置 PC 与 OPL000 连接同一个路由，在 PC 上使用网络调试助手，创建一个 TCP 服务器：



6 · OPL00 作为客户端连接到 TCP 服务器：

AT+CIPSTART="TCP","192.2168.1.101",8080 // protocol 、 server IP & port

响应：

CONNECT

OK

7 · 发送数据：

AT+CIPSEND=4

OK

>ABCD

Recv 4 bytes

SEND OK

8 · 接收数据：

+IPD,n:xxxxx // received n bytes, data=xxxxx

6.2. UDP 传输

1. 设置 Wi-Fi 模式：

AT+CWMODE=1 // Station mode

响应：

OK

2. 扫描 AP：

AT+CWLAP

响应：

+CWLAP:2,Opulinks,-22,30:fc:68:90:a8:a1,1

+CWLAP:3,Opulinks-S,-24,00:d0:41:df:1e:25,6

OK

3. 连接路由：

AT+CWJAP=" SSID" ," password"

响应：

OK

WIFI CONNECTED

WIFI GOT IP

4. 查询设备 IP 信息：

AT+CIFSR

响应：

+CIFSR:STAIP,"169.254.119.102"

+CIFSR:STAMAC,"22:33:44:55:66:76"

OK

5. 设置 PC 与 OPL000 连接同一个路由，在 PC 上使用网络调试助手，创建 UDP 传输：



下面介绍两种 UDP 通信的示例：

6.2.1. 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","192.168.1.101",8080,1112,0

响应：

4,CONNECT

OK

3. 发送数据：

AT+CIPSEND=4,5

OK

> ABCDE

Recv 5 bytes

SEND OK

4. 接收数据：

+IPD,n:xxxxx // received n bytes, data=xxxxx

5. 断开 UDP 传输：

4,CLOSED

OK

6.2.2. 远端可变的 UDP 通信

1. 创建 UDP 传输，最后参数为 2：

AT+CIPSTART="UDP","192.168.1.101",8080,1112,2

响应：

CONNECT

OK

2. 发送数据

AT+CIPSEND=5

OK

> ABCDE

Recv 5 bytes

SEND OK

3. 接收数据：

+IPD,n:xxxxx // received n bytes, data=xxxxx

4. 断开 UDP 传输：

0,CLOSED

OK

6.3. 多连接 TCP 服务器

目前 OPL000 仅支持建立一个 TCP 服务器，且必须使能多连接。

因为 OPL000 只能作为 Station，所以需要连接路由后再建立服务器。

1. 设置 Wi-Fi 模式：

AT+CWMODE=1

响应：

OK

2. 扫描 AP：

AT+CWLAP

响应：

+CWLAP:2,Opulinks,-22,30:fc:68:90:a8:a1,1

+CWLAP:3,Opulinks-S,-24,00:d0:41:df:1e:25,6

OK

3. 连接路由：

AT+CWJAP=" SSID" ," PASSWD"

响应：

OK

WIFI CONNECTED

WIFI GOT IP

4. 查看 IP 信息：

AT+CIFSR

响应：

+CIFSR:STAIP,"192.168.1.103"

+CIFSR:STAMAC,"22:33:44:55:66:76"

OK

5. 使能多连接：

AT+CIPMUX=1

响应：

OK

6. 建立 TCP SERVER：

AT+CIPSERVER=1,8080

响应：

OK

0,CONNECT //在 PC 上建立 TCP Client 并连接后显示

7. 接收数据

+IPD,0,7,192.168.1.104,54789:abcdefg

+IPD,0,7,192.168.1.104,54789:abcdefg0,CLOSED

CONTACT

sales@Opulinks.com