

WS63V100 AT命令

使用案例



更改记录

版本	日期	作者	备注
V1.0	2024-11-25	陈思维	

版权声明：

本文档著作权由 HiHope 所有，保留一切权利。未经书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

本文档中的信息将随着 HiHope 产品和技术的进步不断更新，恕不再通知此类信息的更新。

目录

1 概述	4
2 Wi-Fi模块连接热点案例	4
2.1 准备项	4
2.2 配置AP端	4
2.3 配置STA端	4
3 SLE模块通信案例	5
3.1 准备项	5
3.2 配置服务端	6
3.3 配置客户端	6
3.4 收发数据	7
4 BLE模块通信案例	7
4.1 准备项	7
4.2 配置服务端	7
4.3 配置客户端	8
4.4 收发数据	9

1 概述

型号: NearLink_DK_WS63

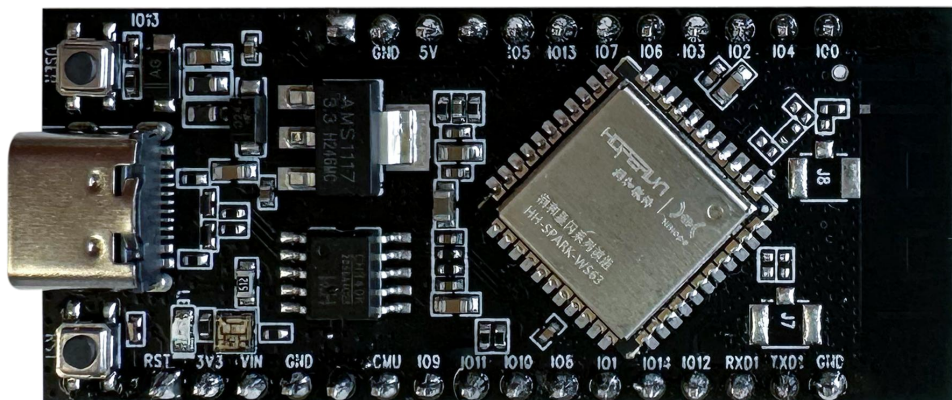


图 1-1 NearLink_DK_WS63 星闪开发板

本文主要介绍NearLink_DK_WS63 星闪开发板 的 AT 指令的常用使用案例。

2 Wi-Fi模块连接热点案例

2.1 准备项

给两块WS63开发板烧录基础固件，配置串口工具参数为，波特率为 115200、8 个数据位、1 个停止位、无校验，无流量控制。

2.2 配置AP端

- (1) 选择一块WS63开发板作为AP端。
- (2) 启动SoftAP,串口发送AT+STARTAP="wifi_test",6,2,"123456789"\n

2.3 配置STA端

- (1) 选择一块WS63开发板作为STA端。
- (2) 启动STA,串口发送AT+STARTSTA\n
- (3) 配置wifi连接重连策略，串口发送AT+RECONN=1,10,3600,50\n

(4) 开始扫描, 串口发送AT+SCAN\n

(5) 获取扫描结果, 串口发送AT+SCANRESULT\n, 扫描结果中会存在wifi_test名称的网络, 如下图。

```
[11:17:26.385]发->◇AT+SCANRESULT
□
[11:17:26.390]收<-◆AT+SCANRESULT
+SCANRESULT:wifi_test,80:2b:d2:e4:***,6,-28,2
+SCANRESULT:MiFi-D842,7c:99:2e:30:***,1,-44,3
+SCANRESULT:HR-Office,70:ba:ef:c6:***,1,-45,3
+SCANRESULT:,70:ba:ef:c6:***,1,-45,3
+SCANRESULT:Kanto,b2:ba:b0:a5:***,5,-52,2
+SCANRESULT:HR-Office,70:ba:ef:c6:***,6,-54,3
+SCANRESULT:hp_dayin,8c:be:be:40:***,11,-54,3
+SCANRESULT:,70:ba:ef:c6:***,6,-55,3
+SCANRESULT:HiHope,68:a0:3e:b5:***,1,-62,2
+SCANRESULT:DIRECT-07-HP 1005 Laser NS,32:24:a9:5d:***,11,-64,2
+SCANRESULT:HR_W9F,70:ba:ef:c6:***,1,-73,3
OK
```

(6) 发起与AP的连接, 串口发送AT+CONN="wifi_test","123456789"\n, 成功连接会有如下打印

```
cmd_sta_connect ssid="wifi_test"
ConnectToNetwork
OK
OK
[osal_event_read:61]:LOS_EventRead failed! ret=0. mask=0x7. mode=0x2
[osal_event_read:61]:LOS_EventRead failed! ret=0. mask=0x7. mode=0x2
drv_soc_ioctl ioctl_cmd->cmd=14.

[11:17:35.089]收<-◆APP|[SYS INFO] mem: used:145040, free:180884; log: drop/all[0/0], at_recv 165.

[11:17:35.188]收<-◆hmac_single_hal_device_scan_complete:vap[1] time[771] chan_cnt[13] chan_0[1] back[0] event[
Scan:vap[1] find bss_num[11] in regdomain, other bss_num[0]

[11:17:35.297]收<-◆drv_soc_ioctl ioctl_cmd->cmd=47.
drv_soc_ioctl ioctl_cmd->cmd=47.
[osal_event_read:61]:LOS_EventRead failed! ret=0. mask=0x7. mode=0x2
drv_soc_ioctl ioctl_cmd->cmd=47.
drv_soc_ioctl ioctl_cmd->cmd=16.
drv_soc_ioctl ioctl_cmd->cmd=47.
drv_soc_ioctl ioctl_cmd->cmd=6.
drv_soc_ioctl ioctl_cmd->cmd=5.
drv_soc_ioctl ioctl_cmd->cmd=6.
drv_soc_ioctl ioctl_cmd->cmd=6.
drv_soc_ioctl ioctl_cmd->cmd=5.
drv_soc_ioctl ioctl_cmd->cmd=1.
drv_soc_ioctl ioctl_cmd->cmd=3.
drv_soc_ioctl ioctl_cmd->cmd=1.
+NOTICE:CONNECTED
drv_soc_ioctl ioctl_cmd->cmd=6.
```

3 SLE模块通信案例

3.1 准备项

给两块WS63开发板烧录基础固件, 配置串口工具参数为, 波特率为 115200、8 个数据位、1 个停止位、无校验, 无流量控制。

3.2 配置服务端

(1) 选择一块WS63开发板作为服务端

(2) 使能星闪，串口发送AT+SLEENABLE\n

(3) 设置地址，串口发送AT+SLESETADDR=0,0x112233445566\n

(4) 注册服务端回调函数，串口发送AT+SSAPSREGCBK\n

(5) 创建一个服务端，串口发送AT+SSAPSADDSRV=0x1234\n

(6) 添加一个服务，串口发送AT+SSAPSADDSERV=0x2222,1\n

(7) 给服务添加一个属性，串口发送

AT+SSAPSADDPROPERTY=1,0x2323,5,5,2,0x1234\n

(8) 给属性添加一个描述符，串口发送

AT+SSAPSADDDESCRIPTOR=1,2,0x3333,5,5,2,0x0200\n

(9) 启动服务，串口发送AT+SSAPSSTARTSERV=1\n

(10) 设置SLE广播参数，串口发送，

AT+SLESETADVPAR=1,3,200,200,0,0x112233445566,0,0x000000000000\n

(11) 设置 SLE 广播数据，串口发送，

AT+SLESETADVDATA=1,10,4,aabbccddeeff11223344,11224455\n

(12) 开始发送 SLE 广播，串口发送AT+SLESTARTADV=1\n

3.3 配置客户端

(1) 选择一块WS63开发板作为客户端

(2) 使能星闪，串口发送AT+SLEENABLE\n

(3) 设置地址，串口发送AT+SLESETADDR=0,0x112233445577\n

(4) 注册客户端回调函数，串口发送AT+SSAPCREGCBK\n

(5) 设置 SLE 扫描参数，串口发送AT+SLESETSCANPAR=1,0x48,0x48\n

(6) 开始扫描，串口发送AT+SLESTARTSCAN\n

客户端串口会打印服务端发送的广播数据，如下图：

[18:15:57.299]收←◆AT+SLESTARTSCAN

at_uart_rx_callback fail:0x80003020

JK

[adv_report] event_type: 0x03, addr_type: 0x0000, addr: 11:**:**:**:55:66

[adv_report] data length: 10, data: 0xaa 0xbb 0xcc 0xdd 0xee 0xff

[adv_report] event_type: 0x0b, addr_type: 0x0000, addr: 11:**:**:**:55:66

[adv_report] data length: 4, data: 0x11 0x22 0x44 0x55 0xee 0xff

[18:15:58.967]收←◆[adv_report] event_type: 0x03, addr_type: 0x0000, addr: 11:**:**:**:55:66

[adv_report] data length: 10, data: 0xaa 0xbb 0xcc 0xdd 0xee 0xff

[18:15:59.133]收←◆[adv_report] event_type: 0x0b, addr_type: 0x0000, addr: 11:**:**:**:55:66

[adv_report] data length: 4, data: 0x11 0x22 0x44 0x55 0x00 0x00

(7) 停止扫描，串口发送AT+SLESTOPSCAN\n

(8) 连接服务端，串口发送AT+SLECONN=0,112233445566\n，连接成功会有如下打印。

```
[18:19:12.083]收←◆AT+SLECONN=0,112233445566
```

OK

```
[18:19:12.186]收←◆[Connected]
addr:11:**:**:**:55:66, handle:00
```

(9) 进行加密配对，串口发送AT+SLEPAIR=0,112233445566\n

(10) 发现服务，串口发送AT+SSAPCFNDSTRU=1,0,1\n

3.4 收发数据

3.4.1 客户端向服务端发送5个字节数据 (0x11, 0x22, 0x33, 0x44, 0x55)

串口发送AT+SSAPCWRITECMD=0,0,1,0,5,0x1122334455\n，

服务端会有如下打印：

```
[18:20:58.413]收←◆[AT] ssaps write request cb server_id(1) conn_id(0) handle(1) status(0).
[AT] ssaps write request cb request_id(0) type(0) need_rsp(0) need_authorize(0).
[AT] ssaps write request cb data len(5).
data
11 22 33 44 55
```

3.4.2 服务端向客户端发送5个字节数据 (0x66, 0x77, 0x88, 0x99, 0xAA)

串口发送AT+SSAPSSNDNTFY=0,1,0,5,0x66778899AA\n，

客户端会有如下打印：

```
[18:21:53.537]收←◆[ssap client] notification info cbk client 0,handle:1, type:0, status:0
data
66 77 88 99 aa
```

4 BLE模块通信案例

4.1 准备项

给两块WS63开发板烧录基础固件，配置串口工具参数为，波特率为 115200、8 个数据位、1 个停止位、无校验，无流量控制。

4.2 配置服务端

(1) 选择一块WS63开发板作为服务端

(2) 使能蓝牙，串口发送AT+BLEENABLE\n

- (3) 设置蓝牙名称, 串口发送AT+BLESETNAME=10,ble_client\n
- (4) 设置蓝牙地址, 串口发送AT+BLESETADDR=0,0x112233445566\n
- (5) 注册服务端回调函数, 串口发送AT+GATTSREGCBK\n
- (6) 创建一个 GATT server, 串口发送AT+GATTSREGSRV=0x1122\n
- (7) 添加一个 GATT 服务, 串口发送AT+GATTSSYNCADDSERV=1,0x1122,1\n
- (8) 给服务添加一个特征, 串口发送

AT+GATTSSYNCADDCHAR=1,14,0x2a4a,0x03,0,1,0x01\n

- (9) 给特征添加一个描述符, 串口发送

AT+GATTSSYNCADDDESCRIPTOR=1,14,0x2a4a,0x03,1,0x02\n

- (10) 启动服务, 串口发送AT+GATTSSTARTSERV=1,14\n
- (11) 设置蓝牙广播参数, 串口发送

AT+BLESETADVPAR=0x48,0x48,0,0x000000000000,0,0x000000000000,7,0,1,0,1\n

- (12) 设置蓝牙广播数据, 串口发送

AT+BLESETADVDATA=6,0x112233445566,0,0,1\n

- (13) 开始发送广播, 串口发送AT+BLESTARTADV=1\n

4.3 配置客户端

- (1) 选择一块WS63开发板作为客户端
- (2) 使能蓝牙, 串口发送AT+BLEENABLE\n
- (3) 设置地址, 串口发送AT+BLESETADDR=0,0x112233445577\n
- (4) 注册客户端回调函数, 串口发送AT+GATTCREGCBK\n
- (5) 创建一个 GATT client, 串口发送AT+GATTCREG=0x3434\n
- (6) 连接服务端, 串口发送AT+BLECONN=0,0x112233445566\n,有如下打印表示连接成功

接成功

```
[18:31:29.994]收←◆AT+BLECONN=0,0x112233445566
OK
[Connected] addr:11:***:***:22:66, handle:00
[GATTClient]Mtu changed——client:1 conn_id:0 mtu size:23
status:0
```

- (7) 配对服务端, 串口发送AT+BLEPAIR=0,0x112233445566\n
- (8) 获取配对信息, 串口发送AT+BLEGETPAIREDDEV\n
- (9) 发现服务, 串口发送AT+GATTCFNDSERV=1,0,0x1122\n

4.4 收发数据

4.4.1 客户端向服务端发送消息

客户端发送写请求，发送五个数据 (0x11,0x22,0x33,0x44,0x55),串口发送

```
AT+GATTTCWRITEREQ=1,0,17,5,0x1122334455\n ,
```

服务端的串口打印接收到的数据，说明通信正常。

```
[18:34:32.491]收←◆[GATTServer]WriteReqReceiveCallback—server_id:1 conn_id:0
request_id:4 handle:17 offset:0 need_authorize:0 need_rsp:1 is_prep:0
data_len:5 data:
11 22 33 44 55
status:0
```

4.4.2 服务端向客户端发送消息

(1) 服务端发送通知，发送五个数据 (0x66,0x77,0x88,0x99,0xAA)，在服务端的串口发送，

```
AT+GATTSSNDNTFY=1,0,14,5,0x66778899AA\n ,
```

(2) 客户端读取服务端发送消息，在客户端的串口发送

```
AT+GATTCREADBYHDL=1,0,14\n,
```

客户端打印接收数据说明收发正常。

```
[18:35:54.140]收←◆[GATTClient]Read result—client:1 conn_id:0
handle:14 data_len:5
data:66 77 88 99 aa
status:0
```