

Wifi ble 模块通用透传应用 手册

版本：V0.2

更新日期：2022 年 04 月 02 日

深圳市易连物联网有限公司版权所有

本产品的规格书如有变更，恕不另行通知。

深圳市易连物联网有限公司保留在不另行通知的情况下，对其中所包含的规格书和材料进行更改的权利，同时由于信任所引用的材料所造成的损害（包括结果性损害），包括但不限于印刷上的错误和其他与此出版物相关的错误，易连物联网将不承担责任。

修改记录

文档版本	作者	发布日期	修改说明
V0.1	lx1	2021/7/12	1. 初稿
V0.2	lx1	2022/4/2	1. 补充配置 wifi 相关指令

目录

修改记录.....	- 2 -
目录.....	- 3 -
1 说明.....	- 5 -
1.1 目的.....	- 5 -
2 蓝牙接口（默认配置）.....	- 5 -
2.1 蓝牙名称：AiLink_xxxx.....	- 5 -
2.2 UUID 说明.....	- 5 -
2.3 蓝牙服务 0xFFE0 结构：.....	- 5 -
3 配置 wifi 指令.....	- 6 -
3.1 扫描 AP Type = 0x80.....	- 6 -
3.2 模块上报扫描的 AP 名字 Type = 0x81.....	- 7 -
3.3 模块上报扫描的 AP 地址和安全信息等 Type = 0x82.....	- 8 -
3.4 模块上报扫描完毕 Type = 0x83.....	- 9 -
3.5 设置希望连接的 AP 地址 Type = 0x84.....	- 10 -
3.6 获取希望连接的 AP 地址 Type = 0x85.....	- 11 -
3.7 设置希望连接的 AP 密码 Type = 0x86.....	- 12 -
3.8 获取希望连接的 AP 密码 Type = 0x87.....	- 13 -
3.9 发起 wifi 连接、断开 wifi 连接 Type = 0x88.....	- 14 -
4 设置/查询指令.....	- 15 -
4.1 设置访问的 IP 地址 Type = 0x8B.....	- 15 -
4.2 获取访问的 IP 地址 Type = 0x8C.....	- 16 -
4.3 设置访问的端口号 Type = 0x8D.....	- 17 -
4.4 获取访问的端口号 Type = 0x8E.....	- 18 -
4.5 设置访问的路径 Type = 0x96.....	- 19 -
4.6 获取访问的路径 Type = 0x97.....	- 20 -
4.7 获取当前连接的 AP 名字（wifi 名称） Type = 0x94.....	- 20 -
4.8 设置蓝牙名称 Type = 0x01.....	- 21 -
4.9 获取蓝牙名称 Type = 0x02.....	- 23 -
4.10 读取 MAC 地址 Type = 0x0d.....	- 24 -
4.11 获取模块 SN 号 Type = 0x95.....	- 25 -
4.12 设置蓝牙名称 Type = 0x01.....	- 25 -
4.13 读取模块版本号 Type = 0x0e.....	- 26 -
4.14 设置模块立即进入休眠 Type = 0x19.....	- 28 -
4.15 唤醒模块 Type = 0x1a.....	- 29 -
4.16 恢复出厂设置 Type = 0x22.....	- 30 -
4.17 获取模块状态 Type = 0x26.....	- 31 -
4.18 模块上报状态.....	- 31 -
5 协议透传指令.....	- 32 -
6 数据透传.....	- 32 -

7 工作流程/应用实例.....	- 33 -
7.1 准备阶段.....	- 33 -
7.2 MCU 发送数据阶段.....	- 34 -
7.3 服务器数据处理阶段.....	- 34 -
7.4 获取模块唯一码，生成设备外壳上的二维码.....	- 34 -
7.4.1 方案一：获取模块 MAC 地址，生成设备外壳上的二维码.....	- 34 -
7.4.2 方案二：获取模块 SN，生成设备外壳上的二维码.....	- 34 -
7.5 注意事项.....	- 35 -
8 联系我们.....	- 36 -

1 说明

1.1 目的

本文档适用于 WM 系列模块如何蓝牙配网，然后通过 wifi 发送数据给服务器。

2 蓝牙接口（默认配置）

2.1 蓝牙名称：AiLink_xxxx

其中 xxxx 为 Mac 地址后 4 个字符。

2.2 UUID 说明

模块有两个服务 UUID，一个是模块固定的服务 UUID，为 FFE0，一个是用户可以自定义的服务 UUID，默认为 FEE0。

2.3 蓝牙服务 0xFFE0 结构：

- 服务 UUID：
0000**FFE0**-0000-1000-8000-00805F9B34FB
- 特征值 UUID1：
0000**FFE1**-0000-1000-8000-00805F9B34FB
属性：read,write,write no response
功能：APP 下发的数据会通过此 UUID 透传给 MCU
- 特征值 UUID2：
0000**FFE2**-0000-1000-8000-00805F9B34FB
属性：read,notify
功能：MCU 发给 BLE 的数据由此 UUID 透传给 APP
- 特征值 UUID3：
0000**FFE3**-0000-1000-8000-00805F9B34FB
属性：read,write,write no response,notify
功能：APP 与 BLE 进行设置类指令的 UUID，有 write 和 notify

3 配置 wifi 指令

3.1 扫描 AP Type = 0x80

模块接收:

Byte	Value	Description	
0	0xA6	包头	
1	0x02	Payload 长度	
2	0x80	Type: 扫描 AP	Payload
3	0x01	Value: 1: 扫描所有频段设备 隐藏操作 (用于测试) 11-20: 扫描 Value -10 对应的频段的 AP	
4	Sum (1~3)	校验和	
5	0x6A	包尾	

模块响应:

Byte	Value	Description	
0	0xA6	包头	
1	0x02	Payload 长度	
2	0x80	Type: 回复扫描 AP 结果	Payload
3		结果值: 0x00: 成功 0x01: 失败 0x02: 不支持	
4	Sum (1~3)	校验和	
5	0x6A	包尾	

3.2 模块上报扫描的 AP 名字 Type = 0x81

模块上报:

Byte	Value	Description	
0	0xA6	包头	
1		Payload 长度	
2	0x81	Type: 上报扫描到的 AP 名字	Payload
3		本次扫描到的 AP 编号	
4~n		名称 (需要对应 ASCII 表)	
n+1	Sum (1~n)	校验和	
N+2	0x6A	包尾	

➤ 举例: 扫描到的 AP 叫做 IOT-wifi

收到: A6 0A 81 00 49 4F 54 2D 77 69 66 69 53 6A

3.3 模块上报扫描的 AP 地址和安全信息等 Type = 0x82

模块上报:

Byte	Value	Description	
0	0xA6	包头	
1		Payload 长度	
2	0x82	Type: 上报扫描到的 AP 信息	Payload
3		本次扫描到的 AP 编号	
4~9		AP 地址, 小端模式	
10		AP 信号强度, 单位-dbm	
11		安全类型: 0x00: 开放 0x01: WEP 0x02: WPA_PSK 0x03: WPA2_PSK 0x04: WPA_WPA_2_PSK 0x05: WPA2_ENTERPRISE	
12		0x00: 陌生 wifi 0x01: 保存过密码的 wifi 0x02: 目前连接着的 wifi	
13	Sum (1~12)	校验和	
14	0x6A	包尾	

➤ 举例: 扫描到的 AP 地址 E0 : 24 : 81 : A2 : A0 : D4

收到: A6 0B 82 00 E0 24 81 A2 A0 D4 2A 04 02 58 6A

3.4 模块上报扫描完毕 Type = 0x83

模块上报:

Byte	Value	Description	
0	0xA6	包头	
1	0x02	Payload 长度	
2	0x83	Type: 上报扫描完毕	Payload
3		扫描到的 AP 个数:	
4	Sum (1~3)	校验和	
5	0x6A	包尾	

3.5 设置希望连接的 AP 地址 Type = 0x84

模块接收:

Byte	Value	Description	
0	0xA6	包头	
1	0x07	Payload 长度	
2	0x84	Type: 设置连接的 AP 地址	Payload
3~8		设备地址, 小端	
9	Sum (1~8)	校验和	
10	0x6A	包尾	

模块响应:

Byte	Value	Description	
0	0xA6	包头	
1	0x02	Payload 长度	
2	0x84	Type: 回复设置连接的 AP 地址结果	Payload
3		结果值: 0x00: 成功 0x01: 失败 0x02: 不支持	
4	Sum (1~3)	校验和	
5	0x6A	包尾	

3.6 获取希望连接的 AP 地址 Type = 0x85

模块接收:

Byte	Value	Description	
0	0xA6	包头	
1	0x01	Payload 长度	
2	0x85	Type: 设置连接 AP 密码	Payload
3	0x86	校验和	
4	0x6A	包尾	

模块响应:

Byte	Value	Description	
0	0xA6	包头	
1	0x07	Payload 长度	
2	0x85	Type: 回复设置连接 AP 密码结果	Payload
3~8		设备地址 (小端)	
9	Sum (1~9)	校验和	
10	0x6A	包尾	

3.7 设置希望连接的 AP 密码 Type = 0x86

模块接收:

Byte	Value	Description
0	0xA6	包头
1		Payload 长度
2	0x86	Type: 设置连接 AP 密码
3		0: 后面没有包 1: 后面还有包
4~n		密码 (最多 14byte) 最多合计 56byte
n+1	Sum (1~n)	校验和
n+2	0x6A	包尾

模块响应:

Byte	Value	Description
0	0xA6	包头
1	0x02	Payload 长度
2	0x86	Type: 回复设置连接 AP 密码结果
3		结果值: 0x00: 成功 0x01: 失败 0x02: 不支持
4	Sum (1~3)	校验和
5	0x6A	包尾

3.8 获取希望连接的 AP 密码 Type = 0x87

模块接收:

Byte	Value	Description	
0	0xA6	包头	
1	0x01	Payload 长度	
2	0x87	Type: 获取连接 AP 密码	Payload
3	0x88	校验和	
4	0x6A	包尾	

模块响应:

Byte	Value	Description	
0	0xA6	包头	
1	0x02	Payload 长度	
2	0x87	Type: 回复连接 AP 密码	Payload
3		0: 后面没有包 1: 后面还有包	
4~n		网址 (最多 14byte) 最多联系 4 个包 56byte	
n+1	Sum (1~n)	校验和	
n+2	0x6A	包尾	

3.9 发起 wifi 连接、断开 wifi 连接 Type = 0x88

模块接收:

Byte	Value	Description	
0	0xA6	包头	
1	0x02	Payload 长度	
2	0x88	Type: 设置 wifi 状态	Payload
3		0x00: 断开连接 0x01: 发起连接	
4	Sum (1~3)	校验和	
5	0x6A	包尾	

模块响应:

Byte	Value	Description	
0	0xA6	包头	
1	0x02	Payload 长度	
2	0x88	Type: 回复设置 wifi 状态结果	Payload
3		结果值: 0x00: 成功 0x01: 失败 0x02: 不支持	
4	Sum (1~3)	校验和	
5	0x6A	包尾	

4 设置/查询指令

4.1 设置访问的 IP 地址 Type = 0x8B

模块接收:

Byte	Value	Description	
0	0xA6	包头	
1		Payload 长度	
2	0x8B	Type: 设置访问的 IP 地址	Payload
3		0: 后面没有包 1: 后面还有包	
4~n		网址 (最多 14byte) 最多合计 56byte	
n+1	Sum (1~n)	校验和	
n+2	0x6A	包尾	

模块响应:

Byte	Value	Description	
0	0xA6	包头	
1	0x02	Payload 长度	
2	0x8B	Type: 回复设置访问的 IP 地址结果	Payload
3		结果值: 0x00: 成功 0x01: 失败 0x02: 不支持	
4	Sum (1~3)	校验和	
5	0x6A	包尾	

➤ 举例: 假设网址是 <http://47.113.114.70:8092/index/>, 那么 IP 地址是 “47.113.114.70”, 对应 ascii 码是

0x34 0x37 0x2e 0x31 0x31 0x33 0x2e 0x31 0x31 0x34 0x2e 0x37 0x30

设置网址: A6 0F 8B 00 34 37 2E 31 31 33 2E 31 31 34 2E 37 30 21 6A

4.2 获取访问的 IP 地址 Type = 0x8C

模块接收:

Byte	Value	Description	
0	0xA6	包头	
1	0x01	Payload 长度	
2	0x8C	Type: 获取访问的 IP 地址	Payload
9	0x8D	校验和	
10	0x6A	包尾	

模块响应:

Byte	Value	Description	
0	0xA6	包头	
1		Payload 长度	
2	0x8C	Type: 回复访问的 IP 地址	Payload
3		0x00: 后面没有包 0x01: 后面还有包	
4~n		网址 (最多 14byte) 最多联系 4 个包 56byte	
n+1	Sum (1~n)	校验和	
n+2	0x6A	包尾	

4.3 设置访问的端口号 Type = 0x8D

模块接收:

Byte	Value	Description	
0	0xA6	包头	
1	0x03	Payload 长度	
2	0x8D	Type: 设置访问的端口号	Payload
3		端口号的高字节	
4		端口号的低字节	
5	Sum (1~4)	校验和	
6	0x6A	包尾	

模块响应:

Byte	Value	Description	
0	0xA6	包头	
1	0x02	Payload 长度	
2	0x8D	Type: 回复设置访问的端口号结果	Payload
3		结果值: 0x00: 成功 0x01: 失败 0x02: 不支持	
4	Sum (1~4)	校验和	
5	0x6A	包尾	

➤ 举例: 设置端口 8092

发送: A6 03 8D 1F 9C 4B 6A

4.4 获取访问的端口号 Type = 0x8E

模块接收:

Byte	Value	Description	
0	0xA6	包头	
1	0x01	Payload 长度	
2	0x8E	Type: 获取访问的端口号	Payload
3	0X8F	校验和	
4	0x6A	包尾	

模块响应:

Byte	Value	Description	
0	0xA6	包头	
1	0x03	Payload 长度	
2	0x8E	Type: 回复访问的端口号	Payload
3		端口号的高字节	
4		端口号的低字节	
5	Sum (1~4)	校验和	
6	0x6A	包尾	

4.5 设置访问的路径 Type = 0x96

模块接收:

Byte	Value	Description	
0	0xA6	包头	
1		Payload 长度	
2	0x96	Type: 设置访问的路径	Payload
3		0: 后面没有包 1: 后面还有包	
4~n		网址 (最多 14byte) 最多合计 56byte	
n+1	Sum (1~n)	校验和	
n+2	0x6A	包尾	

模块响应:

Byte	Value	Description	
0	0xA6	包头	
1	0x02	Payload 长度	
2	0x96	Type: 回复设置访问的路径结果	Payload
3		结果值: 0x00: 成功 0x01: 失败 0x02: 不支持	
4	Sum (1~3)	校验和	
5	0x6A	包尾	

➤ 举例: 假设网址是 `http://47.113.114.70:8092/index/`, 那么访问路径是 “/index/”, 对应 ascii 码是

0x2F 0x 69 0x 6E 0x 64 0x 65 0x 78 0x 2F

设置网址: A6 09 96 00 2F 69 6E 64 65 78 2F 15 6A

4.6 获取访问的路径 Type = 0x97

模块接收:

Byte	Value	Description	
0	0xA6	包头	
1	0x01	Payload 长度	
2	0x8C	Type: 获取访问的网址	Payload
9	0x8D	校验和	
10	0x6A	包尾	

模块响应:

Byte	Value	Description	
0	0xA6	包头	
1		Payload 长度	
2	0x8C	Type: 回复访问的网址	Payload
3		0x00: 后面没有包 0x01: 后面还有包	
4~n		网址 (最多 14byte) 最多联系 4 个包 56byte	
n+1	Sum (1~n)	校验和	
n+2	0x6A	包尾	

4.7 获取当前连接的 AP 名字 (wifi 名称) Type = 0x94

模块接收:

Byte	Value	Description	
0	0xA6	包头	
1	0x01	Payload 长度	
2	0x94	Type: 获取当前 AP 名	Payload
3	0x95	校验和	
4	0x6A	包尾	

模块响应:

Byte	Value	Description	
0	0xA6	包头	
1		Payload 长度	
2	0x94	Type: 回复当前 AP 名	Payload
3~n		AP 名字	
n+1	Sum (1~n)	校验和	

n+2	0x6A	包尾
-----	------	----

➤ 举例：发送 A6 01 94 95 6A

回复：A6 10 94 49 4F 54 2D 77 69 66 69 00 00 00 00 00 00 6C 6A

4.8 设置蓝牙名称 Type = 0x01

模块接收：

Byte	Value	Description
0	0xA6	包头
1	Len	Payload 长度（最大 16byte）
2	0x01	Type: 设置蓝牙名称
3~n	Name	名称（需要对应 ASCII 表）
n+1	Num	MAC 字符个数：名称后面跟随的 MAC 字符的个数 0x00：代表没有，则是固定蓝牙名称。 0x01：代表后面带有 mac 地址的 1 个字符，例如： Swan_x。 0x02：代表后面带有 mac 地址的 2 个字符，例如： Swan_xx。 默认 Num=4；Num 最大为 12 注：Name 长度+ “_” +Num 最大为 15
n + 2	Sum (1~n)	校验和
N+3	0x6A	包尾

模块响应：

Byte	Value	Description
0	0xA6	包头
1	0x02	Payload 长度
2	0x01	Type: 回复设置蓝牙名称结果
3		结果值： 0x00：成功（立即生效） 0x01：失败 0x02：不支持
4	Sum (1~3)	校验和
5	0x6A	包尾

设置蓝牙名称可以设置为固定字符作为蓝牙名称，例如设置为 swan，所有的模块都会显示为 swan。同时也可以设置为固定蓝牙名称+ “_” + Mac 地址的方式，这样子有利于每个模块的名称都有差异。

➤ 举例： 蓝牙的 MAC 地址为 12 : 34 : 56 : 78 : 9A : BC。

如果设置蓝牙名称为 swan, 且不带 MAC 地址时, 那么发送 A6 06 01 73 77 61 6E 00 C0 6A , 则蓝牙名称为 swan;

如果设置蓝牙名称为 swan, 且带 MAC 地址 2 个字符, 那么发送 A6 06 01 73 77 61 6E 02 C2 6A , 则蓝牙名称为 swan_BC;

如果设置蓝牙名称为 swan, 且带 MAC 地址 4 个字符, 那么发送 A6 06 01 73 77 61 6E 04 C4 6A , 则蓝牙名称为 swan_9ABC;

整个蓝牙名称长度最长为 15 个字符。

4.9 获取蓝牙名称 Type = 0x02

模块接收:

Byte	Value	Description	
0	0xA6	包头	
1	0x01	Payload 长度	
2	0x02	Type: 获取蓝牙名称	Payload
3	0x03	校验和	
4	0x6A	包尾	

模块响应:

Byte	Value	Description	
0	0xA6	包头	
1	Len	Payload 长度 (最大 16 byte)	
2	0x02	Type: 回复蓝牙名称	Payload
3 ~ n	Name	蓝牙名称 (最长 15 byte)	
n + 1	Sum (1~n)	校验和	
n + 2	0x6A	包尾	

➤ 举例: 蓝牙名称为 swan_BC

发送查询指令: A6 01 02 03 6A

BM 返回名称: A6 08 02 73 77 61 6E 5F 42 43 A7 6A

4.10 读取 MAC 地址 Type = 0x0d

模块接收:

Byte	Value	Description	
0	0xA6	包头	
1	0x01	Payload 长度	
2	0x0D	Type: 读取 MAC 地址值	Payload
3	0x0E	校验和	
4	0x6A	包尾	

模块响应:

Byte	Value	Description	
0	0xA6	包头	
1	0x07	Payload 长度	
2	0x0D	Type: 回复 MAC 地址值	Payload
3~8		Mac 地址值 (6byte、小端序)	
9	Sum (1~8)	校验和	
10	0x6A	包尾	

➤ 举例: MAC 地址为 11 : 22 : 33 : 44 : 55 : 66

返回: A6 07 0D 66 55 44 33 22 11 79 6A

4.11 获取模块 SN 号 Type = 0x95

模块接收:

Byte	Value	Description	
0	0xA6	包头	
1	0x01	Payload 长度	
2	0x95	Type: 获取 SN 号	Payload
3	0x96	校验和	
4	0x6A	包尾	

模块响应:

Byte	Value	Description	
0	0xA6	包头	
1	0x10	Payload 长度	
2	0x95	Type: 回复 SN 号	Payload
3~17		SN 号	
18	Sum (1~17)	校验和	
19	0x6A	包尾	

➤ 举例: 发送 A6 01 95 96 6A

回复: A6 10 95 57 4D 05 88 4A 18 32 23 CD 1E 55 8B C2 F8 2F 41 6A

4.12 设置蓝牙名称 Type = 0x01

模块接收:

Byte	Value	Description	
0	0xA6	包头	
1	Len	Payload 长度 (最大 16byte)	
2	0x01	Type: 设置蓝牙名称	Payload
3 ~ n	Name	名称 (需要对应 ASCII 表)	
n+1	Num	MAC 字符个数: 名称后面跟随的 MAC 字符的个数 0x00: 代表没有, 则是固定蓝牙名称。 0x01: 代表后面带有 mac 地址的 1 个字符, 例如: Swan_x。 0x02: 代表后面带有 mac 地址的 2 个字符, 例如: Swan_xx。 默认 Num=4; Num 最大为 12	

		注：Name 长度+ “_” +Num 最大为 15	
n + 2	Sum (1~n)	校验和	
N+3	0x6A	包尾	

模块响应:

Byte	Value	Description	
0	0xA6	包头	
1	0x02	Payload 长度	
2	0x01	Type: 回复设置蓝牙名称结果	Payload
3		结果值: 0x00: 成功 (立即生效) 0x01: 失败 0x02: 不支持	
4	Sum (1~3)	校验和	
5	0x6A	包尾	

设置蓝牙名称可以设置为固定字符作为蓝牙名称，例如设置为 swan，所有的模块都会显示为 swan。同时也可以设置为固定蓝牙名称+ “_” + Mac 地址的方式，这样子有利于每个模块的名称都有差异。

➤ 举例： 蓝牙的 MAC 地址为 12 : 34 : 56 : 78 : 9A : BC。

如果设置蓝牙名称为 swan，且不带 MAC 地址时，那么发送 A6 06 01 73 77 61 6E 00 C0 6A ，则蓝牙名称为 swan；

如果设置蓝牙名称为 swan，且带 MAC 地址 2 个字符，那么发送 A6 06 01 73 77 61 6E 02 C2 6A ，则蓝牙名称为 swan_BC；

如果设置蓝牙名称为 swan，且带 MAC 地址 4 个字符，那么发送 A6 06 01 73 77 61 6E 04 C4 6A ，则蓝牙名称为 swan_9ABC；

整个蓝牙名称长度最长为 15 个字符。

4.13 读取模块版本号 Type = 0x0e

模块接收:

Byte	Value	Description	
0	0xA6	包头	
1	0x01	Payload 长度	
2	0x0E	Type: 读取 BM 模块软硬件版本号	Payload
3	0x0F	校验和	
4	0x6A	包尾	

模块响应:

Byte	Value	Description
0	0xA6	包头
1	0x0A	Payload 长度
2	0x0E	Type: 回复 BM 模块软硬件版本号
3		产品型号。byte3 、byte4 为 ASCII 字符，byte5 为数字。
4		
5		
6		
7		硬件版本号 H
8		软件版本号 S
9		定制版本号 P
10		年 实际年份=年+2000 例如：2019 年 年=2019-2000=19
11		月 1~12
12		日 1~31
12	Sum (1~11)	校验和
13	0x6A	包尾

➤ 举例：如软硬件版本号为 WM05H1S1.0P0_20190507

解析：WM05 为产品型号，对应实际数据为 0x57 0x4D 0x05

H1 为硬件版本号 1，对应实际数据为 0x01

S1.0 为软件版本号 1.0 ，对应实际数据为：0x0A（带 1 位小数点）

P0 为定制版本号，对应实际数据为 0

年：2019-2000=19，对应实际数据 0x13

返回：A6 0A 0E 57 4D 05 01 0A 00 13 05 07 EB 6A

4.14 设置模块立即进入休眠 Type = 0x19

模块接收:

Byte	Value	Description	
0	0xA6	包头	
1	Len	Payload 长度	
2	0x19	Type: 设置进入睡眠	Payload
3	0x01	Value: 0x01	
4		休眠模式: 0x00: BLE 关闭, WIFI 关闭, 串口关闭 (掉电模式)。 0x01: BLE 正常工作, WIFI 正常工作, 串口关闭 (正常睡眠)。	
5		保留位	
6		保留位	
7	Sum (1~6)	校验和	
8	0x6A	包尾	

模块响应:

Byte	Value	Description	
0	0xA6	包头	
1	Len	Payload 长度	
2	0x19	Type: 回复设置进入睡眠的结果	Payload
3		结果值: 0x00: 成功 (成功后 100ms 后进入睡眠) 0x01: 失败 0x02: 不支持	
4	Sum (1~3)	校验和	
5	0x6A	包尾	

➤ 举例: 设置进入掉电模式

发送: A6 05 19 01 00 00 00 1f 6A

4.15 唤醒模块 Type = 0x1a

模块接收:

Byte	Value	Description	
0	0xA6	包头	
1	0x02	Payload 长度	
2	0x1A	Type: 设置模块唤醒	Payload
3	0x01	Value: 0x01	
4	0x1D	校验和	
5	0x6A	包尾	

模块响应:

Byte	Value	Description	
0	0xA6	包头	
1	0x02	Payload 长度	
2	0x1A	Type: 回复设置模块唤醒结果	Payload
3		结果值: 0x00: 成功 0x01: 失败 0x02: 不支持	
4	Sum (1~3)	校验和	
5	0x6A	包尾	

如果模块处于睡眠状态，第一次发指令是没有响应的，此指令只是唤醒模块。

4.16 恢复出厂设置 Type = 0x22

模块接收:

Byte	Value	Description
0	0xA6	包头
1	0x02	Payload 长度
2	0x22	Type: 设置恢复出厂设置
3	0x01	Value: 0x01
4	0x25	校验和
5	0x6A	包尾

模块响应:

Byte	Value	Description
0	0xA6	包头
1	0x02	Payload 长度
2	0x22	Type: 回复设置模块重启结果
3		结果值: 0x00: 成功 (成功后, 100ms 后恢复出厂设置) 0x01: 失败 0x02: 不支持
4	Sum (1~3)	校验和
5	0x6A	包尾

4.17 获取模块状态 Type = 0x26

模块接收:

Byte	Value	Description	
0	0xA6	包头	
1	0x01	Payload 长度	
2	0x26	Type: 获取状态	Payload
3	0x27	校验和	
4	0x6A	包尾	

模块响应:

Byte	Value	Description	
0	0xA6	包头	
1	0x03	Payload 长度	
2	0x26	Type: 返回模块状态	Payload
3		模块状态: bit0-bit3 表示 BLE 状态: 0: 无连接 1: 已连接 2: 配对完成 Bit4-bit7 表示 wifi 状态: 0: 没连接 AP; 1: 连接 AP 失败, 连接时密码错误、AP 信号不好、主动断开都会是这个状态; 2: 连接的 AP 信号不好; 3: 成功连接上 AP; 4: 正在连接 AP;	
4		工作状态: 0: 唤醒 1: 进入休眠 2: 模块准备就绪	
5	Sum (1~4)	校验和	
6	0x6A	包尾	

4.18 模块上报状态

当 BLE、WIFI、功耗模式改变时, 模块都会通过获取模块状态的响应包格式主动进行上报状态变化。

5 协议透传指令

根据易连已定好的 AiLink 协议，做数据的传输。

传输格式：

Byte	Default	Description
0	0xA7	包头
1~2		产品类型 CID
3		Payload 长度（最大 15byte）
n		Payload
n+1	SUM (1~n)	(1~n)校验和
n+2	0x7A	包尾

协议透传指令的识别：

包头和包尾是固定的：0xA7，0x7A。

校验和是指 byte1 + ... + byte n 的值，取低 8 位。

6 数据透传

不符合设置指令与协议透传指令的数据一律采用数据透传，即收到什么数据就传什么数据。

7 工作流程/应用实例

7.1 准备阶段

- (1) MCU 上电，等待模块回复
- (2) 模块上电成功，回复模块状态指令
- (3) **MCU 设置网址（IP、端口），模块回复设置网址成功**

举例：

目标网址：<http://test.ailink.revive.bak.aicare.net.cn/api/v1/wifiDevice/register>

其中，IP 地址为：<http://test.ailink.revive.bak.aicare.net.cn>

访问的路径为：</api/v1/wifiDevice/register>

MCU 设置 IP 地址：

发→◇A6 10 8B 01 74 65 73 74 2E 61 69 6C 69 6E 6B 2E 72 65 07 6A □

发→◇A6 10 8B 01 76 69 63 65 2E 62 61 6B 2E 61 69 63 61 72 CD 6A □

发→◇A6 0A 8B 00 65 2E 6E 65 74 2E 63 6E 6E 6A □

MCU 收到模块返回的设置 IP 地址成功：

收←◆A6 02 8B 00 8D 6A

MCU 设置访问的路径：

发→◇A6 10 96 01 2F 61 70 69 2F 76 31 2F 77 69 66 69 44 65 6D 6A □

发→◇A6 10 96 00 76 69 63 65 2F 72 65 67 69 73 74 65 72 2F 10 6A □

MCU 收到模块返回的设置访问路径成功：

收←◆A6 02 96 00 98 6A

- (4) 设备连接小程序
- (5) 模块回复 MCU 连接成功的状态
- (6) 用户在小程序上进行配网操作。注：配网时，设备需要保持亮屏状态。

小程序配网 demo：http://doc.elinkthings.com/web/#/36?page_id=140

- (7) 配网成功，模块回复 MCU 配网成功，小程序显示配网成功。

MCU 收到模块回复的配网成功：

收←◆A6 03 26 31 02 5C 6A

7.2 MCU 发送数据阶段

(1) MCU 按照数据透传发送数据，如 MCU 发送数据“1122”。

7.3 服务器数据处理阶段

- (1) 服务器接收到设备上传的测量结果数据
- (2) 服务器解析、处理测量数据。
- (3) 服务器推送处理后的数据到小程序。

举例：

模块上传数据：1112，模块 sn=574d05884a1871004fcd043a71a7de00。

服务器收到设备的数据请求"params":"V00FiEoYcQBPzQQ6cafeABES"，

服务器接收到数据，进行 base64 解码，得出 1112。

parameter	params	sn
1112	V00FiEoYcQBPzQQ6cafeABES	574d05884a1871004fcd043a71a7de00
1122334455667788990011223344556677889900	V00FiEoYcQA11w5Ee7HoABEIMORVZnelmQARijNEVWZ3iJkA	574d05884a18710035d70e447bb1e800
1122334455667788990011223344556677889900	V00FiEoYcQA11w5Ee7HoABEIMORVZnelmQARijNEVWZ3iJkA	574d05884a18710035d70e447bb1e800
3132	V00FiEoYcQA11w5Ee7HoADEy	574d05884a18710035d70e447bb1e800

7.4 获取模块唯一码，生成设备外壳上的二维码

7.4.1 方案一：获取模块 MAC 地址，生成设备外壳上的二维码

- (1) MCU 发送“读取 BLE 地址”指令给模块
- (2) 模块返回 MAC 地址
- (3) MCU 截取 MAC 地址内容，生成对应二维码

发→◇A6 01 0D 0E 6A □

收←◆A6 07 0D 88 4A 18 71 00 35 A4 6A

其中，88 4A 18 71 00 35 是设备的 MAC 地址。

7.4.2 方案二：获取模块 SN，生成设备外壳上的二维码

- (1) MCU 发送“读取模块 SN 号”指令给模块
- (2) 模块返回 SN 号信息
- (3) MCU 截取 SN 号，生成对应二维码

发→◇A6 01 95 96 6A □

收←◆A6 10 95 57 4D 05 88 4A 18 71 00 35 D7 0E 44 7B B1 E8 1B 6A

其中，57 4D 05 88 4A 18 71 00 35 D7 0E 44 7B B1 E8 ，是设备的 SN 号

7.5 注意事项

- (1) WM 系列模块是 ble wifi 模块，MCU 发送数据给模块，模块会优先从蓝牙上传数据。所以使用 WM 系列模块发送数据到服务器的时候，**需要断开蓝牙连接**。
- (2) BLE (Slave) 与 APP (Master) 交互的每包数据默认最大为 20byte，当 MCU 端一次性发送超过 20byte 时，模块会将数据进行分包发送，需 50byte 则分为 20+20+10，分 3 次发送。
- (3) 模块上电需要时间进行配置，当配置完成，进入就绪时，模块会主动给 MCU 返回模块状态信息，详情请查看“模块上报状态”。
- (4) 当模块 BLE 连接时候，数据通道是串口和 BLE；如果 BLE 没有连接，而 WIFI 连接上时，数据通道是串口和 WIFI。

8 联系我们

深圳市易连物联网有限公司

地址：深圳市宝安区西乡街道银田工业区侨鸿盛文化创意园写字楼 A 栋五层 502 室

Tel: + (86) 0755-81773367

Email: hw@elinkthings.com

Web: www.elinkthings.COM