

AiLink BLEWIFI 体脂秤应用手册

版本：V1.3

更新日期：2021 年 09 月 23 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

版本	作者	审核一	审核二	发布日期	修改说明
V1.2	张雨			2021.06.07	重新修整协议
V1.3	HGB	梁永新		2021.09.23	1、支持的固件版本增加 WM07 2、增加：WM05 和 WM07 的区别及注意事项

目录

修改记录.....	- 2 -
目录.....	- 3 -
1 概述.....	- 5 -
2 关于 WMOX 模块.....	- 5 -
2.1 固件版本.....	- 5 -
2.2 WMOX 模块优点.....	- 5 -
3 说明.....	- 5 -
4 硬件参考设计.....	- 6 -
4.1 参考电路.....	- 6 -
4.2 硬件连接说明.....	- 6 -
5 MCU 和模块交互流程.....	- 7 -
6 AiLink 数据传输协议.....	- 8 -
6.1 协议格式规范.....	- 8 -
6.2.1 MCU 上传测量体重.....	- 9 -
6.2.2 MCU 上传温度数据.....	- 10 -
6.2.3 MCU 上传测量阻抗.....	- 10 -
6.2.4 MCU 请求用户信息（若不需要则不传）.....	- 11 -
6.2.5 MCU 上传测量心率（若无则不传）.....	- 12 -
6.2.6 MCU 上传测量完成.....	- 12 -
6.2.7 MCU 上传错误码.....	- 12 -
6.2.8 设置 MCU 单位.....	- 13 -
6.2.9 数据传输结束指令.....	- 14 -
7 AiLink 配置指令协议.....	- 15 -
7.1 协议格式规范.....	- 15 -
7.1.1 通用设置指令.....	- 15 -
7.1.1.1 设置模块立即进入休眠 Type = 0x19.....	- 15 -
7.1.1.2 唤醒模块 Type = 0x1a.....	- 16 -
7.1.1.3 设置模块 CIDVIDPID Type = 0x1d.....	- 17 -
7.1.1.4 恢复出厂设置 Type = 0x22.....	- 18 -
7.1.1.5 获取模块状态 Type = 0x26.....	- 19 -
7.1.2 模块下发状态.....	- 19 -
7.1.2.1 设置电池电量 Type = 0x27.....	- 20 -
7.1.2.2 MCU 请求同步时间 Type = 0x38.....	- 20 -
7.1.2.3 APP 同步时间给 MCU Type = 0x37.....	- 21 -
7.1.2.4 设置蓝牙名称 Type = 0x01.....	- 21 -
7.1.2.5 读取 BLE MAC 地址 Type = 0x0d.....	- 22 -

7.1.2.6 断开 BLE 连接 Type = 0x25.....	- 23 -
8 WM05 和 WM07 区别及注意事项.....	- 24 -
9 应用例子：体脂秤.....	- 24 -
9.1 准备阶段：启动 WM0X 模块.....	- 24 -
9.2 测量阶段：发送测量数据.....	- 25 -
10 联系我们.....	- 26 -

1 概述

- 1.1 本文档适用于接入 AiLink BLEWIFI 体脂秤的 MCU 端开发工程师使用。
- 1.2 本文档详细介绍硬件对接、固件对接。
- 1.3 本文档会持续更新，[官网链接](#)的发布最新版本。

2 关于 WM0X 模块

2.1 固件版本

WM05:WM05_V2.0.0

WM07: WM07_V1.0.0

2.2 WM0X 模块优点

1. BLE 连接可直接使用（WiFi BLE 称可以当做蓝牙称使用）；
2. 配网更容易体验好；
3. 多用户多账户；
4. 体重匹配到用户，秤端可以显示核心体脂信息（体脂率，BMI）。
注：体脂率的算法可以联系我司业务获取；

3 说明

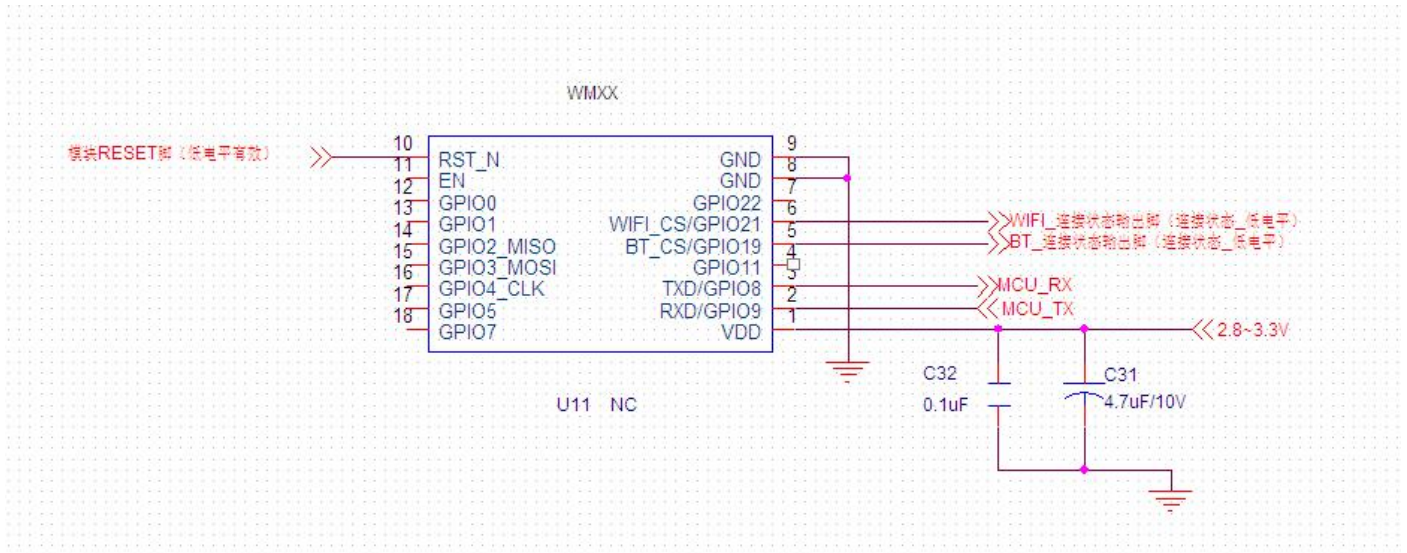
- 3.1 ElinkThings 提供标准化的连接模块、APP、云平台，帮助客户体脂秤快速实现智能化；也可提供 SDK、云平台配置、增值服务和技术支持以实现差异化、个性化的客户定制。
- 3.2 ElinkThings 提供的 WM0X 系列模块，具有 APP 功能强大、体验好等特点。扫描下面二维，下载 APP。



- 3.3 支持 MCU 配置模块（VID、PID）实现 APP 连接产品时型号自定义、图标自定义等个性化设计。

4 硬件参考设计

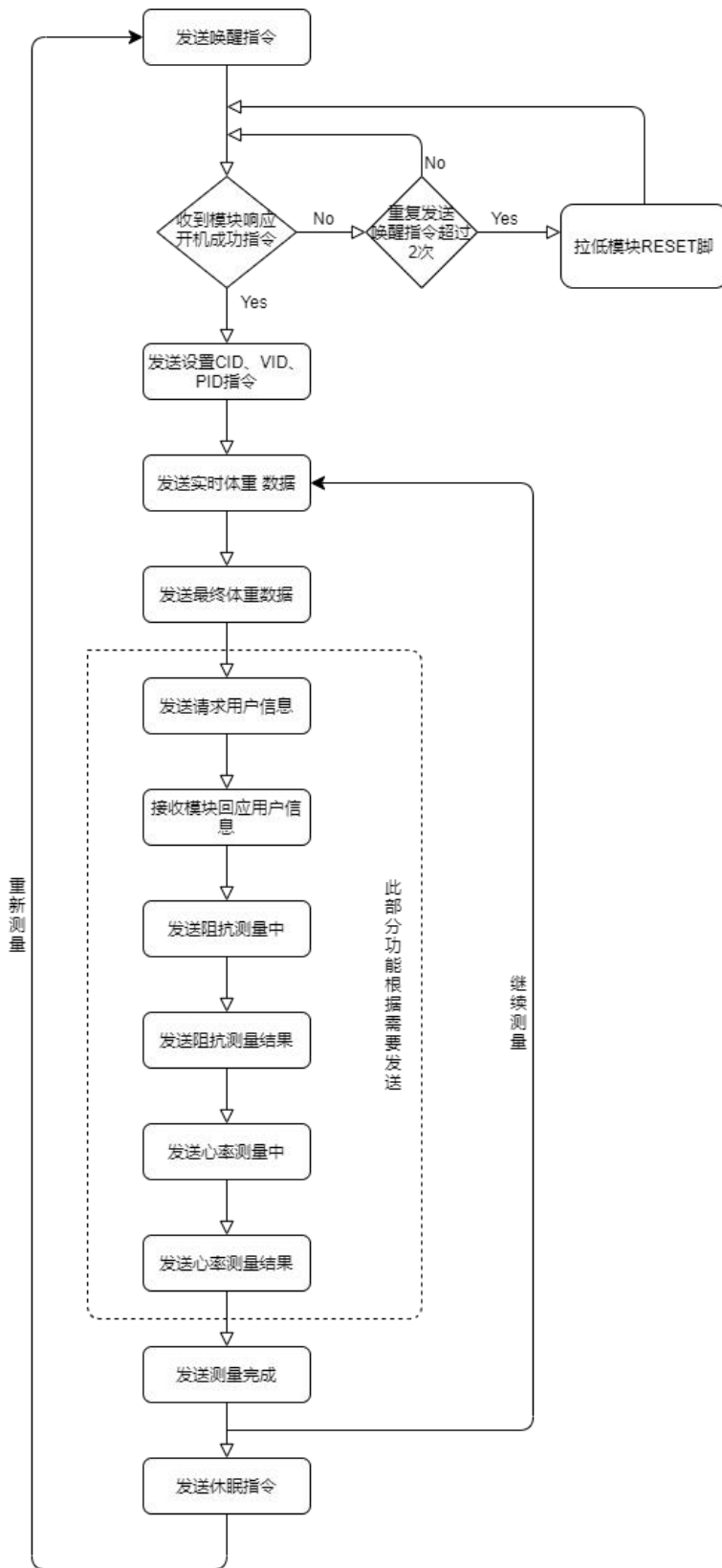
4.1 参考电路



4.2 硬件连接说明

1. 模块 RESET 脚接 MCU 的 CMS 输出 IO，引脚拉低复位。当 MCU 给模块发送唤醒指令，模块一直没有回复，或者回复出错，可以拉低 RESET 来复位模块；
2. WiFi_CS 接 MCU 的 CMS 输入 IO，WiFi 未连接时输出高电平，连接时输出低电平；
3. BT_CS 接 MCU 的 CMS 输入 IO，BLE 未连接时输出高电平，连接时输出低电平；
4. TXD 接 MCU RX 脚，RXD 接 MCU 的 TX 脚，波特率 9600，1 位开始位，8 位数据位，1 位停止位，无奇偶校验位。
5. VDD 接 2.8~3.6v 电源，采用常供电的方式。

5 MCU 和模块交互流程



6 AiLink 数据传输协议

WMOX 体脂秤产品类型 CID 为 0x0011。当模块设置 CID 为 0x0011，数据交互需遵循体脂秤的协议。

6.1 协议格式规范

WMOX 体脂秤协议格式：

Byte	Value	Description	
0	0xA7	包头	
1	0x00	WiFi+BLE 体脂秤产品类型 (CID) 高字节	
2	0x11	WiFi+BLE 体脂秤产品类型 (CID) 低字节	
3		Payload 长度	
4		二级指令	Payload
5~n		体脂秤数据	
n+1	SUM (1~n)	(1~n) 校验和	
n+2	0x7A	包尾 (注: n+2 不能超过 20)	

包头和包尾是固定的，分别为 0xA7，和 0x7A。校验和是指 byte1 + byte2 + ... + byte n 的和，数据大小的不能超过 20Byte。

6.2.1 MCU 上传测量体重

Byte	Default	Description
0	0xA7	包头
1~2	0x0011	产品类型：一级指令，表示 WiFi+BLE 体脂秤
3	0x05	Payload 长度
4		Type: 体重测量 01: 实时体重 02: 稳定体重
5		体重数据高字节
6		体重数据中字节
7		体重数据低字节
8		Bit7~4: 体重数据精度 0000: 0 位小数 0001: 1 位小数 0010: 2 位小数 0011: 3 位小数 Bit3~0: 当前单位: 0000: kg 0001: 斤 0100: st:lb 0110: lb
9	SUM (1~8)	校验和
10	0x7A	包尾

注:

0.001kg 最大量程为 7610.016kg (7610.016kg \approx 16777.215lb)

0xFFFFFFFF=16777215

举例:

A7 00 11 05 01 00 01 F4 10 1C 7A-----50.0kg

A7 00 11 05 01 00 01 F4 11 1D 7A-----50.0 斤

A7 00 11 05 01 00 01 F4 14 20 7A-----50.0lb, APP 自动按 1:14 换算为 3:8.0lb (这里的小数指的是 st:lb 中 lb 显示的小数)

A7 00 11 05 01 00 01 F4 16 22 7A-----50.0lb

6.2.2 MCU 上传温度数据

Byte	Default	Description	
0	0xA7	包头	
1~2	0x0011	产品类型：一级指令，表示 WiFi+BLE 体脂秤	
3	0x04	Payload 长度	
4	0x03	Type: 温度数据	Payload
5		Bit7: 0: 正温度 1: 负温度 Bit6~0: 温度数据高字节	
6		温度数据低字节 (精度为 0.1℃)	
7	SUM (1~6)	校验和	
8	0x7A	包尾	

6.2.3 MCU 上传测量阻抗

Byte	Default	Description	
0	0xA7	包头	
1~2	0x0011	产品类型：一级指令，表示 WiFi+BLE 体脂秤	
3	0x05	Payload 长度	
4		Type: 阻抗测量 06: 测量失败 07: 测量成功 其他: 暂不支持	Payload
5		阻抗数据高字节	
6		阻抗数据低字节 (精度为 1Ω)	
7		体脂算法编号 0: MCU 计算体脂 1-255: 体脂算法编号, APP 计算	
8	SUM (1~7)	校验和	
9	0x7A	包尾	

6.2.4 MCU 请求用户信息（若不需要则不传）

注：MCU 拿到用户信息后，仅用于计算核心体脂信息（体脂率，BMI），并显示在秤端显示屏上，不会上传到 APP。

数据格式：

Byte	Default	Description	
0	0xA7	包头	
1~2	0x0011	产品类型：一级指令，表示 WiFi+BLE 体脂秤	
3	0x02	Payload 长度	
4	0x08	Type:	Payload
5	0x01		
6	SUM (1~5)		
7	0x7A	包尾	

回复设置：

Byte	Default	Description	
0	0xA7	包头	
1~2	0x0011	产品类型：一级指令，表示 WiFi+BLE 体脂秤	
3	0x05	Payload 长度	
4	0x08	Type:	Payload
5	0X02		
6		Bit7~4: 用户特征 0000: 普通人 0001: 业余运动员 0010: 专业运动员 0011: 孕妇 Bit3~0: 当前用户所属 WM0X 体脂秤用户编号 编号 1-8	
7		7bit: 性别（第 8bit 1 代表男性，0 代表女性） 0-6bit: 年龄 (0x)	
8		身高 (0x)	
9	SUM (1~8)	校验和	
10	0x7A	包尾	

模块回复 A7 00 11 05 08 02 00 00 00 20 7A//表示没有匹配到数据

A7 00 11 05 08 02 01 99 AA 64 7A//普通人，当前用户编号 1，男，25 岁，170。

6.2.5 MCU 上传测量心率（若无则不传）

Byte	Default	Description
0	0xA7	包头
1~2	0x0011	产品类型：一级指令，表示 WiFi+BLE 体脂秤
3	0x02	Payload 长度
4		Type: 心率测量 0x0b: 测心率中 0x0c: 测心率成功，带上心率数据 0x0d: 测心率失败
5	0x00	心率数据（精度 1bpm）
6	SUM（1~5）	校验和
7	0x7A	包尾

6.2.6 MCU 上传测量完成

Byte	Default	Description
0	0xA7	包头
1~2	0x0011	产品类型：一级指令，表示 WiFi+BLE 体脂秤
3	0x01	Payload 长度
4	0x0A	Type: 测量完成
5	SUM（1~4）	校验和
6	0x7A	包尾

6.2.7 MCU 上传错误码

Byte	Default	Description
0	0xA7	包头
1~2	0x0011	产品类型：一级指令，表示 WiFi+BLE 体脂秤
3	0x02	Payload 长度
4	0xFF	Type: 错误码
5		错误内容： 1: 超重 其他: 保留
6	SUM（1~5）	校验和
7	0x7A	包尾

6.2.8 设置 MCU 单位

数据格式:

Byte	Default	Description	
0	0xA7	包头	
1~2	0x0011	产品类型: 一级指令, 表示 WiFi+BLE 体脂秤	
3	0x03	Payload 长度	
4	0x81	Type: 单位设置	Payload
5		体重单位: 00: kg 01: 斤 04: st:lb 06: lb	
6	SUM (1~5)	校验和	
7	0x7A	包尾	

回复设置:

Byte	Default	Description	
0	0xA7	包头	
1~2	0x0011	产品类型: 一级指令, 表示 WiFi+BLE 体脂秤	
3	0x02	Payload 长度	
4	0x82	Type: 回复单位设置	Payload
5		结果: 0: 设置成功 1: 设置失败 2: 不支持设置	
6	SUM (1~5)	校验和	
7	0x7A	包尾	

6.2.9 数据传输结束指令

当 WMOX 收到 MCU 发来的“测试完成”，WMOX 透传完成后，会使用“数据传输结束指令”回复 MCU。

Byte	Default	Description	
0	0xA7	包头	
1~2	0x0011	产品类型：一级指令，表示 WiFi+BLE 体脂秤	
3	0x02	Payload 长度	
4	0xFE	Type: 传输结束	Payload
5		Value: 1: 传输成功 0: 传输失败	
6	SUM (1~5)	校验和	
7	0x7A	包尾	

7 AiLink 配置指令协议

通用协议描述了关于 WiFi 和 BLE 的所有指令，MCU 可以使用这些指令去控制模块，实现需要的功能和配置。

7.1 协议格式规范

Byte	Value	Description
0	0xA6	包头
1		Payload 长度（最大 16byte）
2~n		Payload
n+1	SUM(1~n)	(1~n) 校验和
n+2	0x6A	包尾（注：n+2 不能超过 20）

包头和包尾是固定的，分别为 0xA6，和 0x6A。

校验和是指 byte1 + byte2 + ... + byte n 的和，取低位 1 byte。

整个指令，数据大小的不能超过 20Byte。

7.1.1 通用设置指令

7.1.1.1 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
3	0x01	Value: 0x01
4		休眠模式： 0x02: 深度休眠 （BLE 关闭，WiFi 关闭，串口关闭）。 0x01: 智能休眠 （BLE 正常工作，WiFi 正常工作，串口关闭）。 0x00: 定时器休眠 （BLE 关闭，WiFi 关闭，串口关闭，系统内部 rtc 时钟保持）。
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
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

7.1.1.2 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
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
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, 第一次发此指令是触发唤醒, 模块不回复。模块回复开机成功指令表示模块进入了工作状态。

7.1.1.3 设置模块 CIDVIDPID Type = 0x1d

模块接收:

Byte	Value	Description
0	0xA6	包头
1		Payload 长度
2	0x1D	Type: 设置 ID
3		ID 标志位 [Bit0] 0: 不设置 CID (CID 值清 0); 1: 设置 CID [Bit1] 0: 不设置 VID (VID 值清 0); 1: 设置 VID [Bit2] 0: 不设置 PID (PID 值清 0); 1: 设置 PID
4		CID: 产品类型 ID 的高字节
5		CID: 产品类型 ID 的低字节
6		VID: 厂商 ID 的高字节
7		VID: 厂商 ID 的低字节
8		PID: 产品 ID 的高字节
9		PID: 产品 ID 的低字节
10	Sum (1~9)	校验和
11	0x6A	包尾

模块响应:

Byte	Value	Description
0	0xA6	包头
1	Len	Payload 长度
2	0x1D	Type: 回复设置 ID 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

➤ 举例: 设置模块 CID 为 WiFi+BLE 体脂秤

发送: A6 08 1D 07 00 11 00 01 01 00 3F 6A

7.1.1.4 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

7.1.1.5 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
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
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

7.1.2 模块下发状态

当 BLE、WiFi、工作状态改变时, 模块会通过‘获取模块状态’的响应包格式主动进行下发状态, 参照 7.1.1.5。

7.1.2.1 设置电池电量 Type = 0x27

模块接收:

Byte	Value	Description
0	0xA6	包头
1	0x03	Payload 长度
2	0x27	Type: 设置 MCU 电池状态
3		电池充电状态: 0x00: 没有充电 (默认) 0x01: 充电中 0x02: 充满电 0x03: 充电异常
4		电池电量百分比 (0—100)
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	0x27	Type: 回复 MCU 设置电池结果
3		结果值: 0x00: 成功 (成功后会把电池电量上传到 APP) 0x01: 失败 0x02: 不支持
4	Sum (1~3)	校验和
5	0x6A	包尾

➤ 举例: 设置电池正在充电, 电量为 1

发送: A6 03 27 01 01 2C 6A

7.1.2.2 MCU 请求同步时间 Type = 0x38

Byte	Value	Description
0	0xA6	包头
1	0x02	Payload 长度
2	0x38	Type: 请求同步时间
3	0x01	Value: 0x01
4	0x3b	校验和
5	0x6A	包尾

7.1.2.3 APP 同步时间给 MCU Type = 0x37

APP 下发:

Byte	Default	Description
0	0xA6	包头
1		Payload 长度 (最大 15byte)
2	0x37	Type: 设置 MCU 时间
3~9		时间: 7 个 byte 年 (当前年份-2000) 月 日 时 分 秒 星期 (1~7 1=周一~7=周日)
10	SUM (1~9)	校验和
11	0x6A	包尾

MCU 响应:

Byte	Value	Description
0	0xA6	包头
1	0x02	Payload 长度
2	0x37	Type: 回复设置时间结果
3		结果值: 0: 成功 1: 失败 2: 不支持
4	SUM (1~3)	校验和
5	0x6A	包尾

7.1.2.4 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	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。

7.1.2.5 读取 BLE 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

7.1.2.6 断开 BLE 连接 Type = 0x25

模块接收:

Byte	Value	Description	
0	0xA6	包头	
1		Payload 长度	
2	0x25	Type: 设置蓝牙连接状态	Payload
3		主动断开连接标志位 0x01: 立刻断开连接 0x00: 不断开连接	
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	0x25	Type: 回复设置蓝牙连接状态结果	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	

8 WM05 和 WM07 区别及注意事项

- 8.1 WM0X 用于传输体脂秤数据，应设置 CID、VID、PID，WiFi+BLE 体脂秤 CID 为 0x0011；VID,PID 联系我司项目进行分配。
- 8.2 **WM05 支持两种低功耗模式：深度休眠模式功耗 21.7ua，定时器休眠模式 42.7ua；若要支持离线历史记录，必须使用定时器休眠模式。进入休眠模式的指令请参照 7.1.1.1。**
- 8.3 **WM07 支持一种低功耗模式：深度休眠模式功耗 19.9ua，不支持离线历史记录功能。**
- 8.4 模块上电需要初始化，进入就绪状态后主动给 MCU 下发状态信息，详情请查看“模块下发状态”。

9 应用例子：体脂秤

WM0X 的串口通信协议遵循 AiLink 协议，体脂秤数据部分的协议请参照第 6 部分（A7 开头的指令），通用配置指令请参照第 7 部分（A6 开头的指令）。

这里测量的数据包含重量、阻抗、心率，如果不支持某项，可忽略相关项的指令。

9.1 准备阶段：启动 WM0X 模块

- 1) 用户称重，MCU 发送唤醒指令 A6 02 1A 01 1D 6A
模块回复网络连接状态（A6 03 26 00 02 2B 6A 表示 WiFi 和 BLE 未连接）
然后回复开机成功（A6 02 1A 00 1C 6A）。
（若 MCU 发送唤醒指令后，一直没有收到开机成功指令，可拉低一下模块的 RESET 脚进行复位，确保模块开机成功。）
- 2) MCU 设置 CIDPIDVID，首次使用模块时候设置一次即可，模块会保存：
A6 08 1D 07 00 11 00 00 00 00 3D 6A
WM0X 回复设置成功：
A6 02 1D 00 1F 6A
- 3) 期间模块会根据状态变化下发状态信息：
A6 03 26 01 02 2C 6A --WiFi 未连接，BLE 已连接，模块处于就绪状态；
A6 03 26 31 02 5C 6A --WiFi 已连接，BLE 已连接，模块处于就绪状态；
A6 02 91 03 96 6A --模块正在进行 OTA 升级，
A6 02 91 00 93 6A --OTA 升级成功，随后模块将自动复位；
A6 02 91 01 94 6A --OTA 升级失败
- 4) 期间如果 APP 下发测量单位 kg，MCU 将会收到：
A7 00 11 03 81 00 95 7A
MCU 如果支持，则回复：
A7 00 11 02 82 00 95 7A

9.2 测量阶段：发送测量数据

- 5) MCU 发送实时体重 50.0kg（可多次发送，每次发送间隔 100ms 以上）：
A7 00 11 05 01 00 01 F4 10 1C 7A
- 6) MCU 发送最终体重 50.0kg：
A7 00 11 05 02 00 01 F4 10 1D 7A
- 7) MCU 发送阻抗测量中：
A7 00 11 03 04 00 00 18 7A
- 8) MCU 发送阻抗测量成功，MCU 计算体脂，阻抗 560Ω/失败：
A7 00 11 04 07 02 30 01 4F 7A / A7 00 11 04 06 00 00 00 1B 7A
- 9) 如果 MCU 需要请求这条体重的用户信息（若不需要则不发这条指令）
A7 00 11 05 08 02 00 00 00 20 7A
模块回复：A7 00 11 05 08 02 01 99 AA 64 7A
表明这条体重的主人：普通人，男，25 岁，170cm
- 10) MCU 发送心率测量中，如不支持可不发：
A7 00 11 02 0B 00 1E 7A
- 11) MCU 发送心率测量成功 60pbm/失败，如不支持可不发：
A7 00 11 02 0C 3C 5B 7A / A7 00 11 02 0D 00 20 7A
- 12) MCU 发送测量完成：
A7 00 11 01 0A 1C 7A
- 13) MCU 接收到上报数据结果，模块回复上传结果（如果连接有问题，不会收到此结果）
A7 00 11 02 FE 01 12 7A / A7 00 11 02 FE 00 11 7A，上传成功/失败
- 14) MCU 收到上报结果或者等待结果超时（例如 5 秒），则发送休眠指令给模块休眠。
- 15) 再次使用，启动 WMOX 后，从第 5) 步重新开始即可。

10 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