

# T1L&T2L ailink 营养秤应用手册

版本：V1.1

更新日期：2025 年 8 月 18 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	撰写者	审核者	测试者	发布日期	修改说明
V1.0	YYT	Lxy	Zbp	2025/5/30	初版
V1.1	YYT	Lxy	Zq	2025/8/18	1、增加休眠、唤醒指令 2、增加休眠电流值

# 目录

修改记录 .....	- 2 -
目录 .....	- 3 -
1 概述 .....	- 4 -
2 说明 .....	- 4 -
3 蓝牙芯片版本 .....	- 5 -
4 硬件参考设计 .....	- 5 -
4.1 串口 UART .....	- 5 -
4.2 硬件参考设计 .....	- 5 -
5 蓝牙接口（默认） .....	- 6 -
5.1 蓝牙名称：AiLink_xxxx .....	- 6 -
5.2 UUID 说明 .....	- 6 -
5.3 蓝牙连接服务列表 1： FFE0 举例 .....	- 6 -
5.4 广播数据 .....	- 7 -
5.4.1 第一类广播设置 .....	- 7 -
6 流程及软件协议 .....	- 8 -
6.1 基础交互流程 .....	- 8 -
6.2 营养秤指令集 .....	- 9 -
7 蓝牙芯片通用指令集 .....	- 13 -
7.1 获取蓝牙芯片状态（Type: 25、26） .....	- 13 -
7.2 MCU 上报 MCU 电池状态（Type: 27、28） .....	- 14 -
7.3 APP 查询 MCU 拥有的单位（Type: 2C） .....	- 16 -
7.4 设置、读取 CID、VID、PID（Type: 1D、1E） .....	- 18 -
7.5 APP 同步时间到 MCU（Type: 37、38） .....	- 19 -
7.6 读取蓝牙芯片版本号（Type: 46） .....	- 21 -
7.7 设置芯片唤醒（Type: 1A） .....	- 21 -
7.8 设置芯片进入睡眠（Type: 19） .....	- 22 -
7.9 设置、读取芯片自动休眠时间（Type: 17、18） .....	- 23 -
8 举例说明 .....	- 25 -
9 生产测试指导 .....	- 25 -
10 联系我们 .....	- 25 -

## 1 概述

- 1.1 本文档适用于易连蓝牙芯片 T1L/T2L，接入 ailink APP。
- 1.2 本文档适用于营养秤的 MCU 端开发工程师使用。
- 1.3 本文档详细介绍硬件对接、固件对接。
- 1.4 文档会保持更新，以[官网链接](#)为最新版本。

## 2 说明

- 2.1 我们提供标准化的连接蓝牙芯片、app、云平台帮助客户的营养秤快速实现智能化，并提供 sdk、云平台配置、增值服务和技术支持帮忙客户差异化、个性化。
- 2.2 我们提供的蓝牙芯片具有功耗低、认证齐全、APP 功能强大体验好等特点。扫描下面二维码下载 APP。



(AiLink APP)

- 2.3 支持 MCU 配置蓝牙芯片 (CID、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([后台获取 CID、VID、PID 说明](#))。
- 2.4 多种规格选择
  - 2.4.1 T1L 工作电流：3.28mA、峰值电流：25.23mA、休眠电流 0.86uA。
  - 2.4.2 T2L 工作电流：4.82mA、峰值电流：42.56mA、休眠电流 1.5uA。
  - 2.4.3 休眠电流指的是无广播、无连接状态下的功耗。
- 2.5 特别注意事项
  - 2.5.1 **T1L 芯片进入休眠模式，不支持开启低频广播。**

### 3 蓝牙芯片版本

本文档支持的固件版本：

T1L\_V2.0.0, T2L\_V3.0.0 等

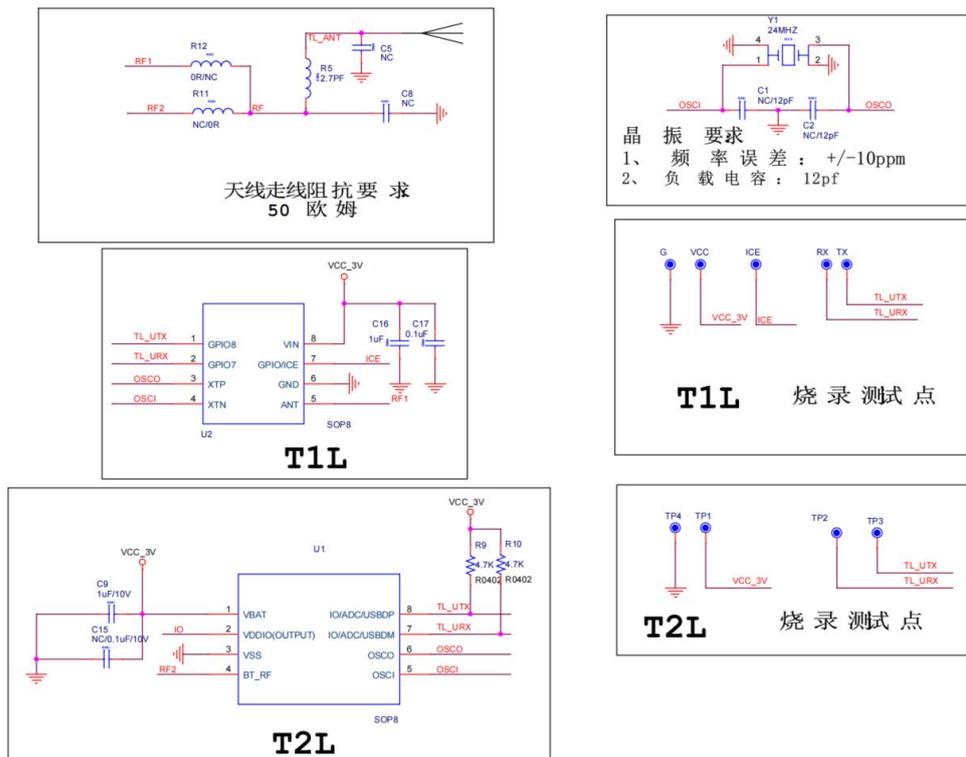
## 4 硬件参考设计

### 4.1 串口 UART

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。

### 4.2 硬件参考设计

4.2.1 MCU 底板要采用双 lay 设计、兼容两款芯片设计。硬件参考设计图如下



#### 4.2.2 设计建议：

1. 针对于 T2L amlink 营养秤产品，建议使用 3 节或以上干电池供电。
2. 画板建议使用 3.0VLDO（负载能力 $>300\text{mA}$ ）。

3. T1L 和 T2L 采用二选一方式。
4. 与 MCU 通信口：RX/TX。
5. 晶振和天线共用，选贴 T1L，R12 0R/R11 NC；选贴 T2L，R12 NC /R11 0R。
6. MCU 使用断电模式。
7. 设计原理图、 $\pi$  型天线封装可直接下载使用：

[http://doc.elinkthings.com/web/#/40?page\\_id=358](http://doc.elinkthings.com/web/#/40?page_id=358)

## 5 蓝牙接口（默认）

### 5.1 蓝牙名称：AiLink\_xxxx

注：xxxx 为 Mac 地址后 4 个字符

### 5.2 UUID 说明

芯片有两个服务 UUID，一个是芯片固定的服务 UUID，为 FFE0，一个是用户可以自定义的服务 UUID，默认为 FEE0。

易联物联网的 AiLink APP 交互使用的服务 UUID 为 FFE0。

同时，两个 UUID 都可以作为普通的数据交互 UUID。

### 5.3 蓝牙连接服务列表 1：FFE0 举例

#### 5.3.1 服务 UUID：

0000**FFE0**-0000-1000-8000-00805F9B34FB

#### 5.3.2 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

#### 5.3.3 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

#### 5.3.4 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

## 5.4 广播数据

**第一类广播：AiLink 设置（默认）。**使用我司 AiLink APP 接入的设置，需根据我司要求的格式进行设置。

### 5.4.1 第一类广播设置

AiLink 设置广播数据内容包含（详情设置请查看[设置读取 CID、PID、VID](#)）：

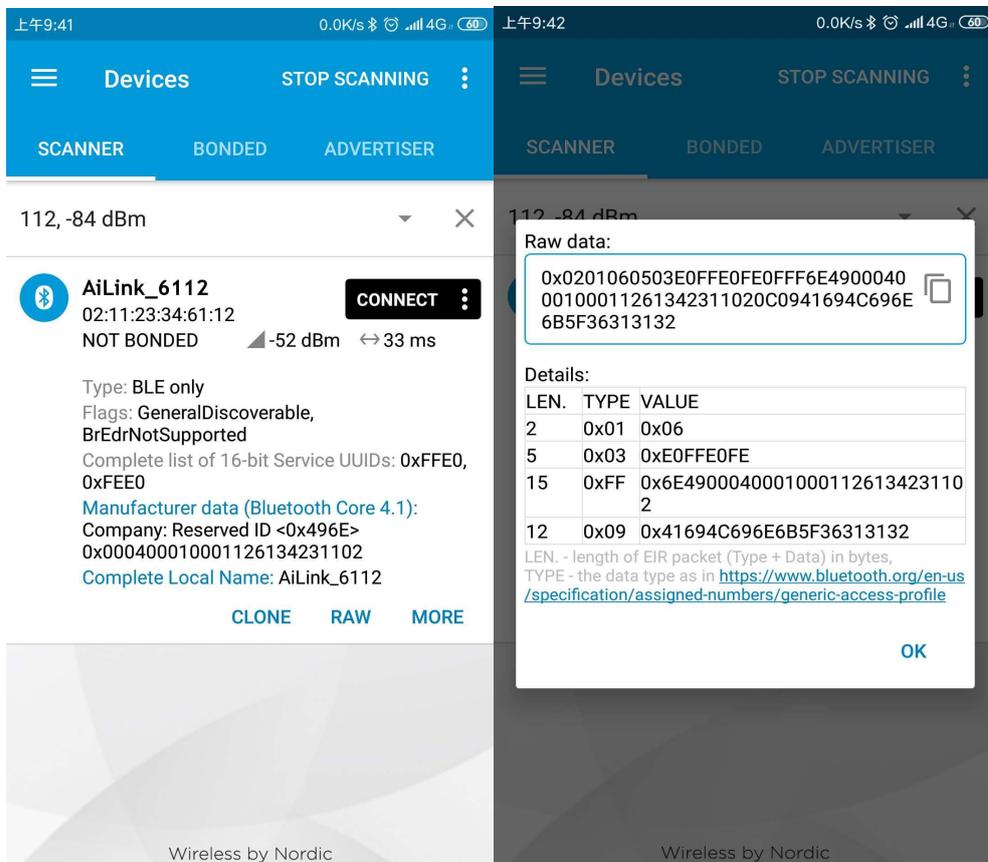
- 1、Company ID 。固定：496e（In, Inet 缩写，此处预留后续申请 SIG company 号）
- 2、CID：产品类型（2bytes）（例如血压计为 0x0001，额温枪为 0x0002，祥查看[协议透传产品介绍](#)）
- 3、VID：厂商 ID（2byte）（由我司分配）
- 4、PID：产品 ID（2byte）（由厂商分配）
- 5、Mac 地址（MAC 是固定的，[大小端序可设置](#)，默认小端序）

例如广播出来的自定义数据为：

6e49000100010001126134231102

6e49：为 In，0001 是 CID，表示产品类型，0001 是 VID，表示厂商 ID，0001 是 PID，表示产品 ID。126134231102 是 Mac 地址，因为是小端序，所以 Mac 地址是：02 : 11 : 23 : 34 : 61 : 12

蓝牙工具显示如下图：



## 6 流程及软件协议

### 6.1 基础交互流程

1. 设备上电并唤醒芯片。
2. [蓝牙芯片返回状态信息](#)。
3. [MCU 设置 CID VID PID](#)。
4. [MCU 上传自身支持的单位](#)。
5. [MCU 上传电量状态](#)（若无可不传）。
6. APP 连接，蓝牙芯片会返回连接状态。
7. MCU 发送重量指令。
8. 设备休眠关机。

## 6.2 营养秤指令集

### A7 指令传输格式（MCU 每条指令的间隔大于 100ms）：

Byte	Default	Description
0	0xA7	包头
1~2	0x0034	产品类型 CID（大端序）
3		Payload 长度（最大 15byte）
n		Payload
n+1	SUM（1~n）	(1~n)校验和=byte1+...+byte n 的值，取低 8 位
n+2	0x7A	包尾

#### 6.2.1 MCU 发送重量

指令格式：

Byte	Default	Description
0	0xA7	包头
1~2	0x0034	产品类型： 0x0034
3		Payload 长度
4	0x01	Type: 0x01: MCU 发送重量
5		流水号 <b>MCU 每发一次数据，流水号要+1，但是一次测量中的稳定数据的流水号不能+1，否则容易导致 APP 保存两次。（即 MCU 如果要发两次稳定数据重量，流水号要一致。）</b>
6		重量高字节
7		重量中字节
8		重量低字节
9		单位 0x00: g 0x01: ml 0x02: lb: oz 0x03: oz 0x04 : kg 0x05: 斤 0x06: 牛奶 ml 0x07: 水 ml 0x08: 牛奶 floz 0x09: 水 floz

		0x0A: lb	
10		小数点 0x00: 无小数点 0x01: 一个小数点 0x02: 两个小数点 0x03: 三个小数点	
11		重量数据正负值 0x00: 正值 0x01: 负值	
12		数据类型: 0x01: 实时数据 0x02: 稳定数据	
13	SUM	校验和	
14	0x7A	包尾	

### 6.2.2 APP/MCU 切换单位

- MCU 和 APP 均可控制切换单位。因此，这条指令可在 MCU 和 BLE 之间双向发送。
- APP 所支持的列表需要和设备端同步，所以 APP 会查询设备端支持的单位列表。
- 例如：  
在秤面上选择为 g 后，手机端也会切换到 g，然后再手机端选择为 ml 后，秤面端也会被切换到 ml。当秤被连接的时候，app 会同步单位到 MCU，所以秤面单位会被切换到 APP 上的单位。

#### MCU 回复指令结果

Byte	Default	Description	
0	0xA7	包头	
1~2	0x0034	产品类型: 0x0034	
3		Payload 长度	
4	0x02	Type: APP/MCU 切换单位	Payload
5		单位 0x00: g 0x01: ml 0x02: lb: oz 0x03: oz 0x04: kg 0x05: 斤 0x06: 牛奶 ml	

		0x07: 水 ml 0x08: 牛奶 floz 0x09: 水 floz 0x0A: lb	
6	SUM (1~5)	(1~5)校验和	
7	0x7A	包尾	

### APP/MCU 回复指令结果

Byte	Default	Description	
0	0xA7	包头	
1~2	0x0034	产品类型: 0x0034	
3		Payload 长度	
4	0x03	Type: 切换单位结果	Payload
5		0x00: 成功 0x01: 失败 0x02: 不支持	
6	SUM (1~5)	(1~5)校验和	
7	0x7A	包尾	

### 6.2.3 去皮功能

➤ 一般情况下，秤面和 app 中都会支持去皮按键，两者都会触发 MCU 实现去皮功能。当秤面不为零时，去皮后，秤面数值会变为零。

数据格式:

Byte	Default	Description	
0	0xA7	包头	
1~2	0x0034	产品类型: 0x0034	
3		Payload 长度	
4	0x04	Type: 去皮	Payload
5		0x01	
6	SUM (1~5)	(1~5)校验和	
7	0x7A	包尾	

## 6.2.4 异常报警

- 当设备端出现异常时，可通过该指令通知 APP
- 数据格式：

Byte	Default	Description
0	0xA7	包头
1~2	0x0034	产品类型： 0x0034
3		Payload 长度
4	0x05	Type: 异常报警
5		Bit0:0=数据正常。1=超载 Bit1:0=电流正常。1=低电
6	SUM (1~5)	(1~5)校验和
7	0x7A	包尾

## 7 蓝牙芯片通用指令集

### 指令格式

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) byte1 + byte2 + ...+byte n 的和, 取低位 1 byte。

设置指令里, 数据的 Byte 数不能超过 20

### 7.1 获取蓝牙芯片状态 (Type: 25、26)

#### 获取蓝牙芯片状态

Byte	Value	Description	
0	0xA6	包头	
1		Payload 长度	
2	0x26	Type: 获取状态	Payload
3	Sum	(1~2)校验和	
4	0x6A	包尾	

#### 蓝牙芯片返回状态:

Byte	Value	Description	
0	0xA6	包头	
1		Payload 长度	
2	0x26	Type: 返回蓝牙芯片状态	Payload
3		连接状态: 0: 无连接 1: 已连接	
4		工作状态: 0: 唤醒 1: 进入休眠 2: 蓝牙芯片准备就绪	
5	Sum	(1~4)校验和	
6	0x6A	包尾	

## 7.2 MCU 上报 MCU 电池状态 (Type: 27、28)

### 上报 MCU 电池状态

Byte	Value	Description
0	0xA6	包头
1		Payload 长度
2	0x27	Type: 设置 MCU 电池状态
3		电池充电状态: 0: 没有充电 (默认) 1: 充电中 2: 充满电 3: 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

### 蓝牙芯片回复 MCU 上报结果

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: 0: 成功 (成功后会把电池电量上传到 APP) 1: 失败 2: 不支持
4	Sum	(1~3)校验和
5	0x6A	包尾

### 查询 MCU 电池状态

Byte	Value	Description	
0	0xA6	包头	
1		Payload 长度	
2	0x28	Type: 获取 MCU 电池状态	Payload
3	Sum	(1~2)校验和	
4	0x6A	包尾	

### 返回 MCU 电池状态

Byte	Value	Description	
0	0xA6	包头	
1		Payload 长度	
2	0x28	Type: 返回 MCU 电池状态	Payload
3		电池充电状态: 0: 没有充电 (默认) 1: 充电中 2: 充满电 3: 充电异常	
4		电池电量百分比 (0—100%) MCU 没有数据上传时, 默认为 0xFFFF	
5	Sum	(1~4)校验和	
6	0x6A	包尾	

### 7.3 APP 查询 MCU 拥有的单位 (Type: 2C)

- APP 端界面的单位显示是根据 MCU 端所拥有的单位来做处理的，所以当 APP 连接到 MCU 时，会发送读取指令来获取 MCU 端所拥有的单位，所以 MCU 端收到该指令时，则务必返回相应的单位数据（不返回则使用系统默认值）。
- **MCU 端需要开机后直接主动上传单位到 APP。**

APP 查询 MCU 端单位指令：  
 （蓝牙芯片直接将此指令传给 MCU）

Byte	Value	Description	
0	0xA6	包头	
1		Payload 长度	
2	0x2C	Type: APP 读取 MCU 端单位	Payload
3		Value 0x01	
4		校验和	
5	0x6A	包尾	

MCU 端返回所拥有的单位指令：  
 单位类型

类型编号	类型	支持类型 (Bit15~Bit0) Bit=0 不支持 Bit=1 支持
01	重量	Bit0: kg Bit1: 斤 Bit2: lb: oz Bit3: oz Bit4: st: lb Bit5: g Bit6: lb (纯 lb 显示) Bit7-bit15 保留
0x08	<b>营养秤单位</b> (体积重量一起使用, 营养秤专用)	Bit0: g Bit1: ml Bit2: lb: oz Bit3: oz Bit4: kg Bit5: 斤 Bit6: 牛奶 ml Bit7: 水 ml Bit8: 牛奶 floz Bit9: 水 floz

		Bit10: 1b Bit11-bit15 保留
--	--	-----------------------------

(蓝牙芯片通过 A6 指令协议传给 APP)  
 数据格式\* (每组数据长度不能多于 20 个 byte)

Byte	Value	Description	
0	0xA6	包头	
1		Payload 长度	
2	0x2C	Type: MCU 回复单位	
3		单位类型: 例如: 重量类型 =01, 长度=02, 温度=03	
4		该单位支持类型高位:	Bit15~Bit0 每一个 Bit 代表一个单位 Bit=0: 不支持 Bit=1: 支持 例如: 重量支持 kg 和 oz 则 byte4=0x00 , byte5=0x09
5		该单位支持类型低位:	
6		单位类型:	
7		单位支持类型高位	Bit15~Bit0 每一个 Bit 代表一个单位 Bit=0: 不支持 Bit=1: 支持
8		单位支持类型低位	
9		单位类型:	Bit15~Bit0 每一个 Bit 代表一个单位 Bit=0: 不支持 Bit=1: 支持
10		单位支持类型高位	
11		单位支持类型低位	
12		校验和	
13	0x6A	包尾	

➤ 举例:

APP 读取 MCU 单位, 发送指令: A6 02 2C 01 2F 6A

- 若 MCU 只拥有重量单位 kg 和斤, 则返回: A6 04 2C 01 00 03 34 6A
- 若是 MCU 支持的类型太多, 一组数据传不完, 则可以分开多组来传, 数据格式不变。

## 7.4 设置、读取 CID、VID、PID (Type: 1D、1E)

- CID、VID、PID 是 amlink 蓝牙芯片重要的一组数据，设备要接入 amlink，必须要设置。具体 CID VID PID 的值可到我司 amlink 管理后台申请，详情请查看：

[http://doc.elinkthings.com/web/#/40?page\\_id=144](http://doc.elinkthings.com/web/#/40?page_id=144)

- T1L&T2L 的 CID 限制为营养称产品，营养秤的 CID 为 0x0034。若设置为其他产品的 CID，蓝牙芯片会回复设置成功。但指令不会生效。

### 设置 ID:

Byte	Value	Description	
0	0xA6	包头	
1		Payload 长度	
2	0x1D	Type: 设置 ID	Payload
3	0x07	设置 ID 标志位	
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	

### 蓝牙芯片回复设置结果:

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

### 获取 ID:

Byte	Value	Description
0	0xA6	包头
1	0x01	Payload 长度

2	0x1E	Type: 获取 ID 设置值	Payload
3	0x1F	(1~2)校验和	
4	0x6A	包尾	

#### 蓝牙芯片返回 ID 值:

Byte	Value	Description	
0	0xA6	包头	
1	Len	Payload 长度	
2	0x1E	Type: 返回 ID	Payload
3	0x07	设置 ID 标志位	
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	

## 7.5 APP 同步时间到 MCU (Type: 37、38)

对于某些设备，具有时间功能的，此时，可利用此指令进行数据的同步。

### ● APP 下发时间。

Byte	Default	Description	
0	0xA6	包头	
1		Payload 长度 (最大 15byte)	
2	0x37	Type: APP 同步时间	Payload
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~n)	(1~n)校验和	
11	0x6A	包尾	

### ● MCU 返回同步时间结果

Byte	Value	Description	
0	0xA6	包头	
1	0x02	Payload 长度	
2	0x37	Type: MCU 返回时间同步结果	Payload
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	

### ● MCU 请求时间

设备有时间功能，且在与 APP 连接状态时，可以请求时间更新，APP 收到该请求，会下发时间同步。

Byte	Value	Description	
0	0xA6	包头	
1	0x02	Payload 长度	
2	0x38	Type: MCU 请求 APP 下发时间	Payload
3		Value 0x01	
4	Sum	(1~3)校验和	
5	0x6A	包尾	

## 7.6 读取蓝牙芯片版本号 (Type: 46)

### ■ 读取蓝牙芯片软硬件版本号:

Byte	Value	Description	
0	0xA6	包头	
1	0x01	Payload 长度	
2	0x46	Type: 读取蓝牙芯片软硬件版本号	Payload
3	0x47	(1~2)校验和	
4	0x6A	包尾	

### ➤ 蓝牙芯片返回软硬件版本号:

Byte	Value	Description	
0	0xA6	包头	
1		Payload 长度	
2	0x46	Type: 回复蓝牙芯片软硬件版本号	Payload
3		针对于某些芯片 MTU 不支持高于 23bytes,一条指令发不完版本号,固可分包发送. Bit7-4:指令条数(当只有 1 条指令时,该值为 0,类推) Bit3-0:当前指令 index(当是第一条指令时,该值为 0,类推)	
4-N		版本号(ASCII 字符) 例如:T1LH1S2.0.0_20250530	
N+1	Sum	校验和	
N+2	0x6A	包尾	

➤ 本文档适配于 T1LH1S2.0.0\_20250530、T2LH1S3.0.0\_05250423 两个版本的固件。

## 7.7 设置芯片唤醒 (Type: 1A)

### 设置芯片唤醒:

Byte	Value	Description	
0	0xA6	包头	
1	0x02	Payload 长度	
2	0x1A	Type: 设置芯片唤醒	Payload
3	0x01	Value: 1: 唤醒芯片	
4	0x1D	(1~3)校验和	
5	0x6A	包尾	

芯片回复设置结果:

Byte	Value	Description
0	0xA6	包头
1	0x02	Payload 长度
2	0x1A	Type: 回复设置芯片唤醒结果
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

## 7.8 设置芯片进入睡眠 (Type: 19)

➤ **T1L 芯片进入休眠模式，不支持开启低频广播。**

- T2L 支持蓝牙连接唤醒（需要开启睡眠后带广播功能，详情看下面设置进入睡眠指令格式）
- 当芯片进入休眠后，支持串口唤醒（MCU 可以发任意数据唤醒芯片，或者发送唤醒指令）

设置睡眠唤醒:

Byte	Value	Description
0	0xA6	包头
1	Len	Payload 长度
2	0x19	Type: 设置进入睡眠
3		Value: 0x01
4		睡眠后是否断开连接，是否开启低频广播: 0: 断开连接，关闭广播。 1: 保持连接，开启广播。 2: 断开连接，开启广播。 3: 保持连接，关闭广播。
5		低频广播间隔时间的高字节
6		低频广播间隔时间的低字节
7	Sum	(1~6)校验和
8	0x6A	包尾

芯片回复设置结果:

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: 0: 成功 (成功后 100ms 后进入睡眠) 1: 失败 2: 不支持	Payload
4	Sum	(1~3)校验和	
5	0x6A	包尾	

- MCU 和 APP 都可以设置芯片进入睡眠，芯片在回复 MCU/APP 时，同时向 APP/MCU 发送芯片当前状态“[返回芯片状态](#)”。

## 7.9 设置、读取芯片自动休眠时间 (Type: 17、18)

- **T1L 芯片设置定时休眠模式，不支持开启低频广播。**

可以设置芯片无数据 (UART/APP) 自动进入休眠模式 (低功耗模式，此时设备的 Tx 要保持为高)，但是要注意的是，当芯片进入 (休眠模式时，MCU 在发数据前，需要提前发一组数据过来唤醒芯片，芯片才能正常开始接受数据。)

注意：该设置芯片会保存。

设置自动睡眠时间：

Byte	Value	Description	
0	0xA6	包头	
1		Payload 长度	
2	0x17	Type: 设置无连接自动休眠时间	
3		自动睡眠标志位: 0: 不开启自动休眠 1: 开启自动休眠，芯片没有连接自动进入低功耗模式	
4		自动睡眠时间的最高字节	Payload 单位 : s 范围: 5 ~ 0xffff/100 (建议设为: 60s)
5		自动睡眠时间的次高字节	
6		自动睡眠时间的次低字节	
7		自动睡眠时间的最低字节	
8		睡眠后是否立刻断开连接，是否开启低频广播: 0: 断开连接，停止广播。 1: 保持连接，开启广播。 2: 断开连接，开启广播。 3: 保持连接，停止广播。	
9		低频广播间隔时间的高字节	Payload 单位: ms ; 范围: 20~2000; (建议设为 1000);
10		低频广播间隔时间的低字节	
11	Sum	(1~10)校验和	
12	0x6A	包尾	

**芯片回复设置结果：**

Byte	Value	Description	
0	0xA6	包头	
1	0x02	Payload 长度	
2	0x17	Type: 回复设置自动睡眠时间结果	Payload
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	

**获取自动睡眠时间：**

Byte	Value	Description	
0	0xA6	包头	
1	0x01	Payload 长度	
2	0x18	Type: 获取自动睡眠时间值	Payload
3	0x19	(1~2)校验和	
4	0x6A	包尾	

**芯片返回自动睡眠时间值：**

Byte	Value	Description		
0	0xA6	包头		
1	Len	Payload 长度		
2	0x18	Type: 返回无连接时自动睡眠时间	Payload	
3		自动睡眠标志位： 0: 不开启自动休眠 1: 开启自动休眠，芯片没有连接		
4		自动睡眠时间的最高字节		单位：s
5		自动睡眠时间的次高字节		
6		自动睡眠时间的次低字节		
7		自动睡眠时间的最低字节		
8		睡眠后是否开启低频广播： 0: 不开启 1: 开启		
9		低频广播间隔的高字节		单位：ms；范围 20~2000
10		低频广播间隔的低字节		
11	Sum	(1~9)校验和		
12	0x6A	包尾		

## 8 举例说明

## 9 生产测试指导

T1L&T2L 生产测试指导说明详细文档请点击连接下载:

[http://doc.elinkthings.com/web/#/40?page\\_id=389](http://doc.elinkthings.com/web/#/40?page_id=389)

## 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](mailto:hw@elinkthings.com)

Web: [www.elinkthings.com](http://www.elinkthings.com)