# **工厂协议-T1-pro 组装厂**

# **1．变更记录**

|  |  |  |  |
| --- | --- | --- | --- |
| 版本 | 时间 | 作者 | 记录 |
| V1.0 | 2024/09/13 | lewis,jeff,lc | 工厂初版 |
| V1.1 | 2024/09/28 | lewis,jeff,lc | 协议更新 |
| v1.2 | 2024/10/13 | lewis,jeff,lc | 协议更新 |
| v1.3 | 2024/10/14 | lewis,jeff,lc | 协议更新 |
| v1.４ | 2024/10/31 | lewis,jeff,lc | 协议更新 |
| v1.5 | 2024/11/6 | lewis,jeff,lc | 协议更新 |
| v1.6 | 2024/11/13 | lewis,jeff,lc | 协议更新 |

V1.1 出入水检测、按键、灯效、电池协议变更

V1.2 测试项电流阈值修改，DevInfo加入sn字段，IMU数据

v1.3 修改出入水电容检测方式， 电池电压范围， 增加温湿度,RTC字段修改"tz"，电池bat加入新的字段

v1.４

a. 修改退出工厂模式的方法，改成不使用上位机，使用组合按键方式，退出工厂模式

1. 新增返厂维修（即再进入工厂模式）

v1.5　新增关机指令，退出工厂模式，需要使用上位机控制

v1.6　SN规则更改，出水检测删除

# **２．介绍**

通过**网络**传输，上位机通过扫码枪连接到设备的ap热点上，通过mac地址确认是否连接到同一个设备上

板端IP：**192.168.4.1**

网络端口号：**12024**

# **３．协议内容**

## **3.1．数据格式**

数据使用json格式

### **示例**

上位机发送请求：

{"DeviceInfo":{}}

下位机返回响应：

{  
 "DeviceInfo":{  
 "model":"X9",  
 "version":"1.0.1"  
 },  
 "res":0  
}

协议中res字段，0代表测试项成功，-1代表测试项失败

## **3.2.　具体协议如下**

### **1.身份确认:** DevInfo

设备信息确认,如bt\_name可以用在蓝牙测试中使用

该项测试几乎不耗时

上位机　请求：

{"DevInfo":{}}

下位机　响应：

product表示产品名

bt\_name蓝牙名

mcu\_base\_ver： mcu base版本

mcu\_led\_ver：mcu led　版本

{  
 "DevInfo":{  
"ap\_addr":"c2:f5:35:bc:1b:d7",  
"ap\_name":"Scuba X1 Pro-56942",  
"bt\_name":"Aiper-Scuba X1 Pro-56942",  
"cpu\_id":"aeca188e4fb56942",  
"fw\_ver":"1.14.3.0",  
"mcu\_base\_ver":"1.2.4",  
"mcu\_led\_ver":"1.2.1",  
"product":"T1 Pro",  
"sn":"TXE4FB56942"},  
 "res":0  
}

### **2.通信测试**

#### **2.1　蓝牙测试**

蓝牙名从DevInfo协议获取

SERVICE\_UUID = "10101910-0000-1000-8000-00805f9b34fb";//订阅uuid  
BLE\_UUID\_SEND = "dfd4416e-1810-47f7-8248-eb8be3dc47f9";//上位机接收蓝牙数据uuid  
BLE\_UUID\_RECV = "9884d812-1810-4a24-94d3-b2c11a851fac";//上位机发送蓝牙数据uuid

BLE\_UUID\_RECV = "9884d812-1810-4a24-94d3-b2c11a851fac";//需要开启notify，获取设备数据,不要使用indicate测试

１：上位机发送如下字符串指令

aRYQGXFAOQprYDMLZhZsA29J数据未加换行符

２：等待设备给上位机发送数据

３：上位机需判断内容，数据内容如下

aRYQGXFAOQprYDMLZhZsA28YdAp3R3RCIkk=

数据未位有一个换行符

#### **2.2　wifi测试**

无线网信号测试，连接到固定wifi路由器上，路由器设置:帐号：x9-test， 密码：12345678

上位机　请求：

{"wifi\_signal":{}}

下位机　响应

{"wifi\_signal":{  
 "strength":-40  
 },  
 "res":0  
}

判断标准：

根据工厂场地，路由器的位置，信号强度在多少范围内正常（默认-55db内），要根据加现场再定具体值

范围：-3db ~ -55db

#### **2.3　岸歌通信测试**

上位机　请求：

{"angosense":{}}

下位机　响应

{"angosense":{  
 "value":10  
 },  
 "res":0  
}

判断标准：

value:表示距离，根据治具实际距离判断误差10cm内

"res":0表示岸歌通信正常，"res":-1 岸歌通信失败

### **3.RTC时间**

上位机发送时间给下位机

发送时间格式：%Y-%m-%d %H:%M:%S

时区：UTC+8

上位机　请求：

{  
 "RtcTest":{  
 "localTime":"2024-08-26 10:11:04",  
 "tz":"UTC+8"  
 }  
}

下位机　响应：

{"RtcTest":{},  
 "res":0  
}

判断标准：

"res":0表示RTC时间正常，　"res":-1 RTC时间失败

### **4.SN烧录**

上位机下发ＳＮ号给下位机

SN组成：TXB + 生产年月(3位数) + 序列号(5位)　共11位

生产年月(3位数)：

第一位是代表年份，2023=3,2024=4 ...... 以此类推

第二三位代表周01,02,03,04,05,06,07 .........

序列号(5位)：即流水号，按照00001,00002 .....以此类推，每周流水号不重复，下一周可以从00001开始

上位机　请求：

{  
 "SnBurn":{  
 "Sn":"TXB44700001"  
 }  
}

下位机　响应：

{"SnBurn":{},  
 "res":0  
}

判断标准：

"res":0表示SN烧录正常，　"res":-1 SN烧录失败

后续"工位"必须检测ＳＮ是否一致

### **5.深度计检测**

需要在水上，水下都要测试

上位机　请求：

{  
 "depthometer":{}  
}

下位机　响应：

{"depthometer":{  
 "value":0  
},  
 "res":0  
}

返回值value， 单位为pa,

判断标准：

空气中检测标准范围为(98kpa~103kpa)，

水下根据放置深度，按照1m=10kpa规律递增，

建议放置10cm水深，判断范围为99kpa~104kpa

### **6.出入水检测**

#### **~~6.1.出入水检测电容~~**

~~上位机:~~

~~{  
 "outwater":{   
 }  
}~~

~~下位机：~~

~~value， 1:出水, 0:入水~~

~~{  
 "outwater":{  
 "value":0   
 },  
 "res":0  
}~~

~~判断标准：~~

~~人工触发传感器状态变化(用水袋靠近或者其他方式)， 0和1状态都要有~~

#### **6.2.出入水检测霍尔1**

上位机:

{  
 "outwater1":{   
 }  
}

下位机：

value， 1:出水, 0:入水

{  
 "outwater1":{  
 "value":0   
 },  
 "res":0  
}

判断标准：

人工触发霍尔状态变化(用磁铁靠近或者其他方式)， 0和1状态都要有

#### **6.3.出入水检测霍尔2**

上位机:

{  
 "outwater2":{   
 }  
}

下位机：

value， 1:出水, 0:入水

{  
 "outwater2":{  
 "value":0   
 },  
 "res":0  
}

判断标准：

人工触发霍尔状态变化(用磁铁靠近或者其他方式)， 0和1状态都要有

### **7.尘盒在位霍尔**

上位机:

{  
 "dirtybox":{   
 }  
}

下位机：

value: 0:到位, 1:未到位

{  
 "dirtybox":{  
 "value":0  
 },  
 "res":0  
}

判断标准：

T0阶段：人工判断

后续：治具判断

### **8.IMU检测**

上位机:

{  
 "IMU":{   
 }  
}

下位机：

roll: 横滚角, pitch: 俯仰角, yaw: 偏航角

gyro\_x: x角速度, accel\_x: x轴线加速度, gyro\_y: y角速度, accel\_y: y轴线加速度, gyro\_z: z角速度, accel\_z: z轴线加速度

{  
 "IMU":{  
 "roll":0,  
 "pitch":0,  
 "yaw":0,  
 "gyro\_x":0,  
 "gyro\_y":0,  
 "gyro\_z":0,  
 "accel\_x":0,  
 "accel\_y":0,  
 "accel\_z":0  
 },  
 "res":0  
}

判断标准：

T0阶段：值有变化就行

后续：治具判断

### **9.有线充电**

{  
 "wired\_charging":{  
 }  
}

下位机：

vol: 电压 单位mv, cur: 电流 单位ma, temp: 温度 单位摄氏度

{  
 "wired\_charging":{  
 "vol":0,  
 "cur":0,  
 "temp":0  
 },  
 "res":0  
}

判断标准： 电压20~25.2v, 电流 0~10A, 温度0~100度

### **10.极片充电**

{  
 "pole\_charging":{  
 }  
}

下位机：

vol: 充电电压 单位mv, cur: 充电电流 单位ma, temp: 电池温度 单位摄氏度

{  
 "pole\_charging":{  
 "vol":0,  
 "cur":0,  
 "temp":0  
 },  
 "res":0  
}

判断标准： 电压20~25.2v, 电流 0~10A, 温度0~100度

### **11.灯板按键**

测试顺序最先测

上位机:主要靠人工检测，软件检测通信是否正常

电源power按键,是开机按键，设备开机代表power按键正常，同时听一下蜂鸣器响声

发送灯板测试指令

{  
 "led\_rgb\_key":{  
   
 }  
}

下位机：

status代表有没有按下: 0:不正常, 1:正常

{  
 "led\_rgb\_key":{  
 "eco\_status":0,  
 "mode\_status":0  
 },  
 "res":0  
}

判断标准：

T0阶段：人工通过按按键

### **11.1.灯效**

上位机:主要靠人工检测，软件检测通信是否正常

发送灯板测试指令

{  
 "led\_rgb":{  
   
 }  
}

下位机：

{  
 "led\_rgb":{  
   
 },  
 "res":0  
}

判断标准：

T0阶段：人工判断灯亮

### **11.2.灯效(组装厂)**

上位机:主要靠人工检测，软件检测通信是否正常

发送灯板测试指令 0灭，1:白，2:红，3:绿，4:蓝，5:黄

{  
 "led\_colour":{  
 "colour":0  
 }  
}

下位机：

{  
 "led\_colour":{  
   
 },  
 "res":0  
}

判断标准：

5种颜色轮流亮，人工判断

### **12.三合一超声波**

上位机:

{  
 "triple\_uls":{   
 }  
}

下位机：

left: 左侧探测距离, mid:中间探测距离, right:右边探测距离,单位mm

{  
 "triple\_uls":{  
 "left":0,  
 "mid":0,  
 "right":0  
 },  
 "res":0  
}

判断标准：

放入水中， 测试结果与距离障碍物实际距离误差10%以内

### **13.左前超声波**

上位机:

{  
 "leftfrount\_uls":{   
 }  
}

下位机：

distance: 探测距离,单位mm

{  
 "leftfrount\_uls":{  
 "distance":0   
 },  
 "res":0  
}

判断标准：

放入水中， 测试结果与距离障碍物实际距离误差10%以内

### **14.左后超声波**

上位机:

{  
 "leftback\_uls":{   
 }  
}

下位机：

distance: 探测距离,单位mm

{  
 "leftback\_uls":{  
 "distance":0   
 },  
 "res":0  
}

判断标准：

放入水中， 测试结果与距离障碍物实际距离误差10%以内

### **15.左下视红外**

上位机:

{  
 "leftdown\_ray":{   
 }  
}

下位机：

value: 0:无遮挡,1有遮挡

{  
 "leftdown\_ray":{  
 "value":0   
 },  
 "res":0  
}

判断标准：

0和1都有触发

### **16.右下视红外**

上位机:

{  
 "rightdown\_ray":{   
 }  
}

下位机：

value: 0:无遮挡,1有遮挡

{  
 "rightdown\_ray":{  
 "value":0   
 },  
 "res":0  
}

判断标准：

0和1都有触发

### **17.浊度计**

上位机:

{  
 "turbidity":{   
 }  
}

下位机：

value: 浊度值，范围(0~243)

{  
 "turbidity":{  
 "value":0   
 },  
 "res":0  
}

判断标准：

放入水中， 双探头对准，数值在50以内

### **18.地磁计**

上位机:

{  
 "magnetometer":{   
 }  
}

下位机：

result: 测试结果

{  
 "magnetometer":{  
 "value":0  
 },  
 "res":0  
}

判断标准：

value： 0为成功，-1为失败

### **19.电池**

上位机:

{  
 "bat":{   
 }  
}

下位机：

pow: 当前电池电量; vol: 电池电压,单位mv；charge\_cur:电池充电电流,单位ma；discharge\_cur: 电池放电电流,单位ma；temp: 电池温度,单位度 ； temp2 电池温度2； temp3 电池温度3

{  
 "bat":{  
 "power":0,  
 "vol":0,  
 "charge\_cur":0,  
 "discharge\_cur":0,  
 "temp":0,  
 "temp2":0,  
 "temp3":0  
 },  
 "res":0  
}

判断标准：电压20~25.5v， 电流小于10A， 温度 -10~100度, 充电和放电时都要测

### **20.**行走轮

上位机:

value 0:前进 1后退

{  
 "wheel":{  
 "value":0  
 }  
}

下位机：

cur\_left: 左轮电流, 单位ma, speed\_left: 左轮速度，单位mm/s

cur\_left: 右轮电流, 单位ma, speed\_left: 右轮速度，单位mm/s

{  
 "wheel":{  
 "speed\_left":0,  
 "speed\_right":0,  
 "cur\_left":0,  
 "cur\_right":0  
 },  
 "res":0  
}

判断标准：

电机转动时间持续10s, 目前速度设置固定200mm/s, 转速180~220之间为pass,

读取转动电流值上报，判断返回值，合格标准：空载电流小于0.75A

### **21.**水泵

上位机:

{  
 "pump":{   
 }  
}

下位机：

cur\_left: 左水泵电流, 单位ma, rpm\_left: 左水泵每秒转速

cur\_right: 右水泵电流, 单位ma, rpm\_right: 右水泵每秒转速

{  
 "pump":{  
 "rpm\_left":0,  
 "rpm\_right":0,  
 "cur\_left":0,  
 "cur\_right":0  
 },  
 "res":0  
}

判断标准：

电机转动时间持续10s, 目前转速设置固定800, 转速720~880之间为pass,

读取转动电流值上报，判断返回值，合格标准：空载电流小于0.3A

### **~~22.指示灯~~**

~~上位机:~~

~~{  
 "indicator\_led":{   
 }  
}~~

~~下位机：~~

~~{  
 "indicator\_led":{   
 },  
 "res":0  
}~~

~~判断标准：~~

~~人工判断指示灯效果~~

### **23.**蜂鸣器

上位机:

{  
 "beep":{   
 }  
}

下位机：

{  
 "beep":{   
 },  
 "res":0  
}

判断标准：

人工判断蜂鸣器是否有响声，开机有蜂鸣器响声

### **24.**温湿度

上位机:

{  
 "humidness\_temperature":{   
 }  
}

下位机：

humidness: 湿度; temperature: 温度，单位摄氏度.

{  
 "humidness\_temperature":{  
 "humidness":0,  
 "temperature":0  
 },  
 "res":0  
}

判断标准：

温度 -10~40度， 湿度0~90

### **30.退出工厂模式，使用组合按键(一般不使用该方式，使用上位机控制退出工厂模式)**

主要做两件事：１:退出工厂模式，２：恢复出厂设置

固件需要V1.13.0以后版本支持

１：设备开机后（等待20Ｓ左右）

２：同时＂power＂ + ＂右测按键＂，听到蜂鸣器响声，黄灯灯带闪烁，

３：黄灯闪烁，然后绿灯长亮（1-2秒），设备自动关闭，即完成退出工厂模式

设备出厂前，必须保证恢复出厂设置

### **30.1退出工厂模式(上位机触发): EndFactory**

该项是工厂最后一个测试项，不可以提前进行该项测试

退出工厂模式成功后，上位机执行30.2关机指令，通知设备关机，结束整个产测，进入下一步机器打包工作

上位机　请求

~~{"EndFactory":{}}~~

下位机　响应

~~{  
 "EndFactory":{  
  
 },  
 "res":0  
}~~

判断标准：

"res":0表示响应正常，　"res":-1 响应失败

### **30.2　关机指令**

上位机　请求

~~{"shutdown":{}}~~

下位机　响应

~~{  
 "shutdown":{  
 },  
 "res":0  
}~~

判断标准：

"res":0表示响应正常，　"res":-1 响应失败

### **31．返厂维修（再进工厂模式）**

固件需要V1.13.0以后版本支持

设备退出工厂模式，如何再次进入工厂模式

１：设备开机后（等待20Ｓ左右），

２：同时power按键+左测按键，听到蜂鸣器响声以及设备的wifi灯效变成呼吸灯效

３：电脑连接设备的AP热点,运行提供的返厂上位机程序，返厂成功，

４：重启机器，机器会进入工厂模式