蓝牙通用整机产测协议

版本号	描述	作者	时间
1.0.0	初始创建(产测授权走串口;整机测试走 dongle) 本文档高度复用《涂鸦蓝牙设备通用产测协议 1.0.1_20190219.docx》，仅仅采用 6、成品产测接口[黄底红字]	李涛	2019-06-18
1.0.1	对安全通信部分进行修改说明[绿底红字]	李涛	2019-06-27
1.0.2	增加 wifi\蓝牙 RSSI 扫描测试 统一帧头为 0x66AA[蓝底红字]	张红伟	2019-07-05
1.0.3	增加 2.20-2.24	张红伟	2019-9-2
1.0.4	增加 2.25 指纹测试	张红伟	2019-09-19
1.0.5	增加 2.3,2.4 进入和退出产测模式接口	张红伟	2019-09-25

目录

一、背景说明：

1.1、目的

- 规范蓝牙产品的 PCBA 或整机产测， 使得 PC 侧产测工具、嵌入式产测逻辑、产测 dongle 能够标准化
- 每一步都能够记录，让产测过程数字化、可追溯

1.2、产测范围定义

- 仅仅适用于整机及 PCBA 测试

1.3、蓝牙设备产测(PCBA、整机)产测拓扑

1.3.1 OTO 模式(One dongle To One DUT)

此模式适用于在一个独立的无线屏蔽的空间中，只存在一个 dongle 和一个 DUT 设备的测试环境。当 dongle 收到测试指令后，将搜索 DUT，当搜索到有且仅有一个 DUT 设备时，dongle 和 DUT 之间自动建立连接。此模式下，整个测试过程均无需外部输入 DUT 的 SN 和 MAC.

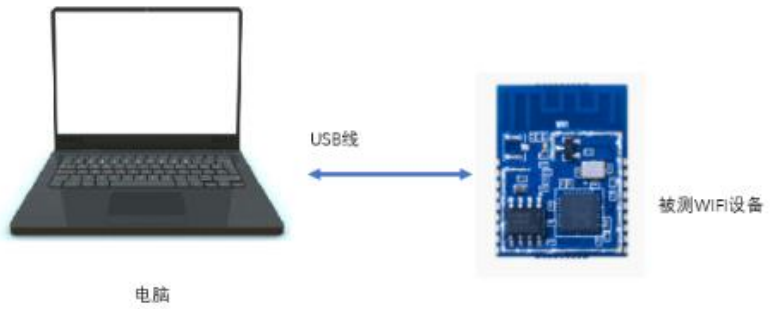
测试方案一　无线通讯一拖一测试模式 （需屏蔽箱）

1.3.2 OTM 模式（One dongle To Multiple DUT）

此模式下，在整个测试环境中仅允许存在一个 dongle，但是允许存在多个处于正常工作状态的 DUT。在某一时刻，dongle 只能与一个 DUT 设备进行通讯。产测软件通过串口与 dongle 通讯，产测软件通过串口发送命令控制 dongle 与具有指定 MAC 地址的 DUT 建立连接。

1.3.3 WCM 模式（Wired Connection Mode）

DUT 设备通过 USB 线，网线或串口线等通讯线与电脑的通讯端口连接的模式。根据需要，一台电脑可以同时连接多个 DUT 设备进行测试。



1.3.4 BCM 模式（BroadCasting Mode）

此模式下，在整个测试环境中仅允许存在一个 dongle，但是允许存在多个处于正常工作状态的 DUT。Dongle 通过广播的方式发送命令，DUT 收到命令后执行相应的动作。

1.4、蓝牙设备产测(PCBA、整机)通信约定

1.4.1 上位机软件与 dongle 的通信约定

采用通用串口通讯,串口参数设置为：
波特率:9600,数据位：8，校验位：None，停止位：1

1.4.2 上位机软件与 DUT 通信约定

采用通用串口通讯,串口参数设置为：
波特率:9600,数据位：8，校验位：None，停止位：1

1.4.3 OTO 和 OTM 方式下加密通信策略

由于工厂中是无法连接外网的，因此走加密的流程去产测是行不通的。但是为了确保设备出厂后是安全的，采用如下策略：

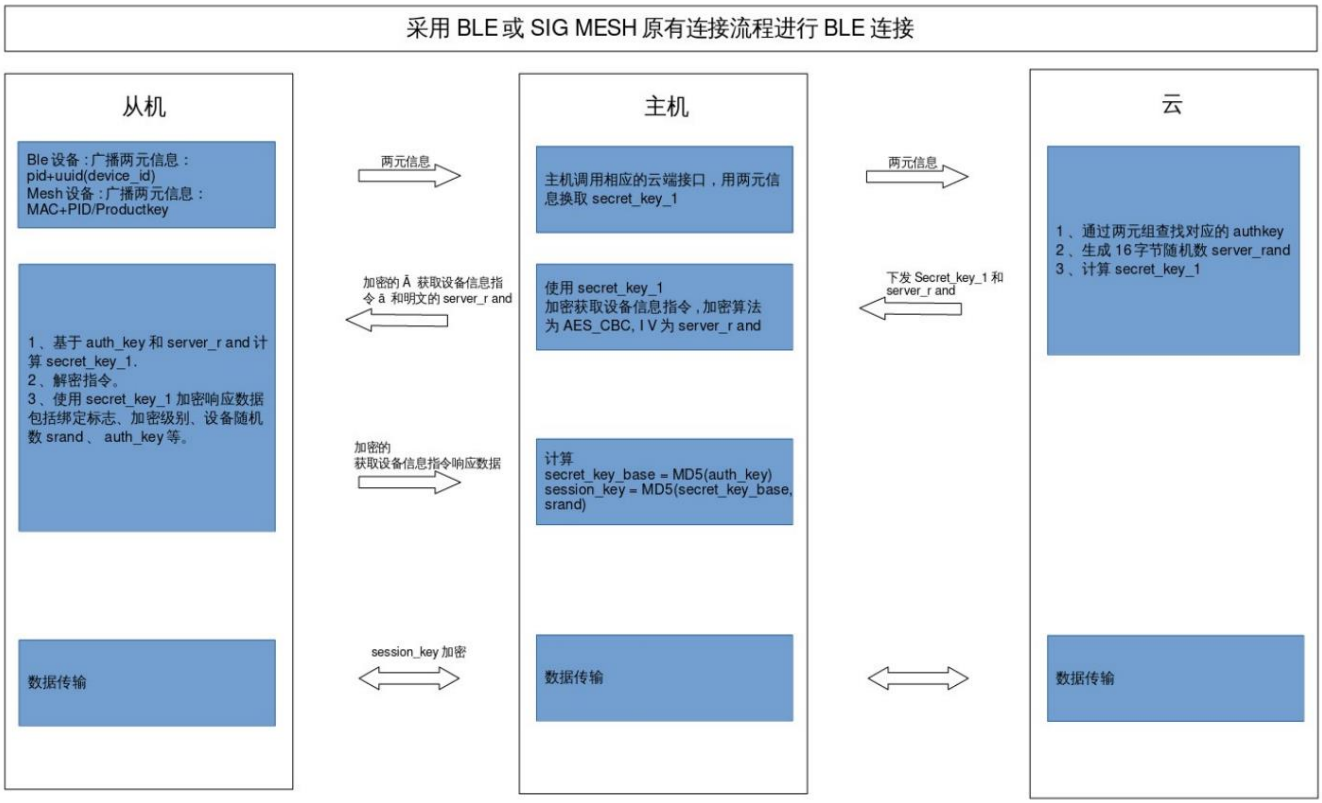
1）设备检测自身是否持续工作超过 2 小时，如果持续工作超过 2 小时，则将 dongle 产测逻辑关了；

2）留出一个特殊的基于安全的命令，可以在某些特定有网的场合，对设备使能产测；

注：第 2 节中的命令都是非加密的，第 3 节中的命令是为了加密通信服务的（具体加密解密在 dongle 和设备中完成，上位机不过是多几条命令和流程，部分接口要和服务端确认是否具备）

加密协议如下：

https://wiki.tuya-inc.com:446/pages/viewpage.action?pageId=26252507



在 OTO 模式，dongle 直接扫描 DUT 的两元组信息，通过串口给上位机，上位机用两元组信息去云端请求 secret_key1 和 server_rand，再通过串口给 dongle，dongle 和 DUT 之间采用上面的加密交互流程，最终以 session_key 作为数据传输秘钥；

在 OTM 模式，通过扫码枪扫描成品 SN，并根据 SN 获取待测蓝牙设备的蓝牙 MAC 地址。上位机通过串口将 MAC 地址传输给 dongle，dongle 根据 MAC 地址区扫描 DUT 的两元组信息，通过串口给 dongle,dongle 和 DUT 之间采用上面的加密交互流程，最终以 session_key 作为数据传输秘钥；

1.4.4 上位机一般操作流程

普通非加密整机产测：获取 MAC（扫码枪等） → 建立连接 → 发送测试命令

加密命令流程：获取 MAC（扫码枪等） → [dongle 默认做:根据 MAC 扫描获取对应设备的元信息] 建立连接 → … → 获取两元信息 → 发送

二、非加密协议介绍：

请求命令帧格式：

序号	字段	长度（Bytes）	说明
1	帧头	2	固定为 0x66AA
2	版本	1	升级扩展用
3	命令字	1	命令类型 （0xF0)
4	数据长度	2	如果是 JSON 格式的字符串则不包括字符串的结束符，字段 5-7 的长度之和
5	协议类型	1	1：Wifi，2：Zigbee，3，蓝牙,4:网关，5：IPC
6	子命令	2	
7	数据	XXXX	
8	校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

响应命令帧格式：

序号	字段	长度（Bytes）	说明
1	帧头	2	固定为 0x66AA
2	版本	1	升级扩展用
3	命令字	1	0XF0
4	数据长度	2	如果是 JSON 格式的字符串则不包括字符串的结束符，字段 5-7 的长度之和
5	协议类型	1	1：Wifi，2：Zigbee，3，蓝牙,4:网关，5：IPC
6	子命令	2	
7	数据	XXXX	
8	校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

2.1 Dongle 与 DUT 建立连接【子命令 0xFF00】

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0xFF00
数据	1	{”mac”:”112233445566”}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

dongle 返回：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0xFF00
数据	12	{"ret":true}/{"ret":false,”msg”:” no_hotspot”}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

2.2 Dongle 与 DUT 断开连接【子命令 0xFF01】

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0xFF01
数据	1	
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

dongle 返回：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0xFF01
数据	12	{"ret":true}/{"ret":false}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

2.3 进入测试模式【子命令 0xFF02】

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0xFF02
数据	1	
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

dongle 返回：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0xFF02
数据	12	{"ret":true}/{"ret":false}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

2.4 退出测试模式【子命令 0xFF03】

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0xFF03
数据	1	
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

dongle 返回：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0xFF03
数据	12	{"ret":true}/{"ret":false}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

2.5 LED 指示灯测试【子命令 0x0001】

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0x0001
数据	1	0x00: 全亮 0x01: 全灭 0x02: 交替闪烁（500ms）
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

dongle 返回：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0x0001
数据	12	{"ret":true}/{"ret":false}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

说明：对于灯少的设备使用交替闪烁，对于灯多的设备使用全亮；LED 动作需要人为判断。

2.6 继电器测试【子命令 0x0002】

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0x0002
数据	1	0x00: 全开 0x01: 全关 0x02: 交替开关三次（500ms）
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

dongle 返回：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0x0002
数据	12	正确返回{"ret":true}/错误不返回-->{"ret":false}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

说明：对于继电器少的设备使用交替动作，对于继电器多的设备使用全开动作；继电器动作需要人为判断。

2.7 按键测试【子命令 0x0003】

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0x00
数据长度	2	
协议类型	1	
子命令	2	0x0003
数据内容	0	
数据	1	0x00
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

dongle 返回：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0x00
子命令	1	0XF0
数据长度	2	0xXXXX
协议类型	1	
子命令	2	0x0003
数据内容	XX	{"keyID":n} N 表示按键值，例如{"keyID":0}、{"keyID":1}、{"keyID":2}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

2.8 开关量传感器测试【子命令 0x0004】

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	0x0001
协议类型	1	1
子命令	2	0x0004
数据内容	1	0x00 传感器类型标识 门磁:0x00 红外:0x01 光敏:0x02 烟雾:0x03 燃气:0x04 水浸:0x05
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

dongle 返回：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	1
子命令	2	0x0004
数据内容	1	{"Dtn":true} / {"Dtn":false} ‘t’表示传感器类型 ’n’表示同类型传感器下的个数 例如：两个红外{"D10":true}和{"D11":true}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

说明：要区分同一个设备上有多个不同类型开关量传感器，同时还要区分同一类型传感器下的哪一个传感器。

2.9 模拟量传感器测试【子命令 0x0006】

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	1
子命令	2	0x0006
数据内容	1	0x00 传感器类型标识 温湿度:0x00 类目待扩充
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

dongle 返回：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	0xXXXX 表示以下字符串长度（不含'\0'）
协议类型	1	1
子命令	2	0x0006
数据内容	XX	{"S1":Value, "S2":Value} 例如：S1: 温度值，S2：湿度值 如果只有一个值则 S2 为 0
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

说明：每次返回两个模拟量值，少于两个模拟量的设备只是用 sensor1，多余两个模拟量的设备，分多次返回。模拟量的值使用 int32_t 传递，传感器需要将 float 小数点消除后转换为 int32_t 发送。

2.10 电机测试【子命令 0x0007】

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	1
子命令	2	0x0007
数据内容	1	0x00: 正转 0x01: 反转 0x02: 停止 0x03: 往复测试
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

dongle 返回：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0x0007
数据内容	12	{"ret":true}/{"ret":false}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

2.11 写入电量统计校准参数【子命令 0x0008】

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	0X0011
协议类型	1	1
子命令	2	0x0008
数据内容	14	校准参数： 1：v_ref（2 字节） 2：i_ref（2 字节） 3：p_ref（2 字节） 4：e_ref（2 字节） 5：v_def（2 字节） 6：i_def（2 字节） 7：p_def（2 字节）
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

dongle 返回：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0X0008
数据	12	{"ret":true}/{"ret":false}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

2.12 电量校准【子命令 0x0009】

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0x0009
数据内容	0	
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

dongle 返回：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0x0009
数据内容	12	{"ret":true}/{"ret":false}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

2.13 调色灯色彩调试【子命令 0x000A】

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0x000A
数据内容	5	R:0-255 G:0-255 B: 0-255 C: 0-255 W: 0-255
校验和	1	

dongle 返回：

字段	长度（Bytes）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型		
子命令	2	0X000A
数据内容	12	{"ret":true}/{"ret":false}
校验和	1	

说明：调色灯的测试先单独测试每个基色，然后进入老化测试。

2.14 老化测试【子命令 0x000B】

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0X000B
数据内容	1	
校验和	1	

模块返回：

字段	长度（Bytes）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0x000B
数据内容	12	{"ret":true}/{"ret":false}
校验和	1	

2.15 红外发射【子命令 0x000C】

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0X000C
数据内容	1	6 字节 mac
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

模块返回：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0X000C
数据内容	12	{"ret":true}/{“ret”:false}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

2.16 进入红外接收模式【子命令 0x000D】

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0X000D
数据内容	0	
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

模块返回：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0X000D
数据内容	12	{"ret":true}/{“ret”:false}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

2.17 退出红外接收模式【子命令 0x000E】

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0X000E
数据内容	1	
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

模块返回：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0X000E
数据内容	12	{"ret":true}/{“ret”:false}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

2.18 产品 SN 写入【子命令 0x000F】

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0X000F
数据内容	1	{“sn”:”11100397*******”}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

模块返回：

字段	长度（Bytes）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0X000F
数据内容	12	{"ret":true}/{“ret”:false}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

2.19 产品 SN 读取【子命令 0x0010】

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0X0010
数据内容	1	
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

模块返回：

字段	长度（Bytes）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0X0010
数据内容	12	{“sn”:”11100397*******”}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

2.20 蓝牙 RSSI 测试【子命令 0x0011】

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0X0011
数据内容	n	“ssid”:”ty_med”/”ty_prod”
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

模块返回：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0X0011
数据内容	n	{“ret”:true,”rssi”:-50}/{“ret”:false}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

注：当上位机给 dongle MAC 地址的时候，先执行 2~3S 的蓝牙 scan，获取对应 MAC 的数 10 个 rssi 值（因为有些连接情况下获取不到 RSSI）。

2.21 Wi-Fi RSSI 测试【子命令 0x0100】

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0X0100
数据内容	n	“ssid”:””tuya_mdev_test”
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

模块返回：

字段	长度（Bytes）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0X0100
数据内容	n	{“ret”:true,”rssi”:-50}/{“ret”:false}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

2.22 Gsensor 三轴传感器测试【子命令 0x0012】

三轴传感器测试成功返回 true，否则返回 false

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0X0012
数据内容	1	
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

模块返回：

字段	长度（Bytes）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0X0012
数据内容	12	{"ret":true}/{“ret”:false}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

2.23 电池电压测试【子命令 0x0013】

读取电池电压，上位机判定是否在正常范围内。

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0X0013
数据内容	1	
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

模块返回：

字段	长度（Bytes）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0X0013
数据内容	12	{"ret":true,"volt":3.6}/{"ret":false}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

2.24 读取 MAC【子命令 0x0014】

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0X0014
数据内容	n	
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

模块返回：

字段	长度（Bytes）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0X0014
数据内容	n	{“ret”:true,”mac”:“1122334455”}/{“ret”:false}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

备注：读取的 mac 地址高位在前地位在后。

2.25 读取 PID【子命令 0x0015】

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0X0015
数据内容	n	
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

模块返回：

字段	长度（Bytes）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0X0015
数据内容	n	{“ret”:true,”pid”:“1122334455"}/{“ret”:false}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

备注:

2.26 读取固件指纹【子命令 0x0016】

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0X0016
数据内容	n	
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

模块返回：

字段	长度（Bytes）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0X0016
数据内容	n	{“ret”:true,”firmName”:“xxxx”,”firmVer”:”x.x.x”}/{“ret”:false}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

备注:

2.27 指纹测试【子命令 0x0017】

包括指纹录入，指纹校验、指纹删除三个过程，三个步骤全部通过返回 true,否则，返回 false，和错误信息。

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0X0017
数据内容	n	
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

模块返回：

字段	长度（Bytes）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0X0016
数据内容	n	{“ret”:true }/{“ret”:false,”err”:”input_err”} 错误类型为：“input_err”，“check_err”，” delete_err”
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

备注:

三、加密协议介绍：

3.1、Dongle 获取 DUT 两元组信息【子命令 0xFE00】

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0xFE00
数据	1	{”mac”:”112233445566”}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

dongle 返回：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0xFE00
数据	n	{"ret":true,”pid”:”abcdefgh”,“mac”:”BC234C000000”} ← mesh {"ret":true,”pid”:”abcdefgh”,“uuid”:”2B3C4D0123456789”} ← 单点 {"ret":false,”msg”:” no_hotspot”}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

3.2、Dongle 下发 DUT 加密验证信息【子命令 0xFE01】

产测软件发送：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0xFE01
数据	n	{“secret_key1”：“<16bytes>”，“server_rand”：“<16bytes>”}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

dongle 返回：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0xFE01
数据	n	{"ret":true}//认证加密成功 {"ret":false}//认证加密失败
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

3.3、Dongle 激活 DUT 整机产测【子命令 0xFE02】

产测软件发送（加密认证成功才能发送）：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0xFE02
数据	0	
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余

dongle 返回：

字段	长度（Bytes）	说明
帧头	2	0x66AA
版本	1	0x00
命令字	1	0XF0
数据长度	2	
协议类型	1	
子命令	2	0xFE02
数据	n	{"ret":true} {"ret":false}
校验和	1	从帧头累加到数据字段的最后一个字节再对 256 求余