蓝牙通用产测授权协议

版本号	描述	作者	时间
1.0.0	初始创建	李涛	2019-05-30
1.0.1	和红伟对一版，形成最终版	李涛	2019-05-31
1.0.2	修改之前对 HID 位数描述的错误，实际为 19 位	李涛	2019-06-03
1.0.3	增加固件防拷贝芯片唯一码上传与公钥下发 [蓝底白字]	李涛	2019-07-09

一、现状：

帧格式：

字段	长度（byte）	说明
帧头	2	固定
版本	1	升级扩展用（00）
命令字	1	具体帧类型
数据长度	2	大端
数据	xxxx	
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

说明：

- 所有大于 1 个字节的数据均采用大端模式传输。

mesh 的产测授权和单点的区别：

cmd	Mesh [66 AA]	Ble 单点 [~~55 AA（彻底弃用）~~和 66 AA]
0	进入	进入（same)
1	获取 hid	获取 hid（same）
2	GPIO 测试	GPIO 测试（same)
3	写授权信息 {"auzkey":"xxxx",“uuid” :“xxxx”,“mac”:”xxxxxx”, “prod_test”:xxxx,“pid”:“ abcdefgh”} 注：是否需要写 pid 根据进入产测返回值	写授权信息 {"auzkey":"xxxx",“uuid” :“xxxx”,“mac”:”xxxxxx”, “prod_test”:xxxx,“pid”:“ abcdefgh”} 注：是否需要写 pid 根据进入产测返回值
4	获取授权信息 {“ret”:true, "auzKey":"xxxx",”hid”:”x xx”,“uuid”:“xxxx”,“mac” :”xxxxxx”,“firmName”:” esp_12F_test”, "firmVer":"1.0.0",”prod_t est”:xxxx,“pid”:“abcdef gh”}/{“ret”:false} 注：是否需要返回 pid 根据进入产测返回值	获取授权信息 {“ret”:true, "auzKey":"xxxx",”hid”:”x xx”,“uuid”:“xxxx”,“mac” :”xxxxxx”,“firmName”:” esp_12F_test”, "firmVer":"1.0.0",”prod_t est”:xxxx,“pid”:“abcdef gh”}/{“ret”:false} 注：是否需要返回 pid 根据进入产测返回值
5	复位	复位（same)
6	获取指纹	获取指纹（same)

7	写入 hid	写入 hid（same)
8	RSSI 测试	RSSI 测试（same)
9	写 OEM 配置文件	
10	读取芯片唯一码（根据产测进入返回标志位来确定是否使用）	读取芯片唯一码（根据产测进入返回标志位来确定是否使用）
11	下发固件防拷贝云端生成的公钥（根据产测进入返回标志位来确定是否使用）	下发固件防拷贝云端生成的公钥（根据产测进入返回标志位来确定是否使用）

二、升级策略：

~~1）对于老的 mesh 和 BLE 需要兼容，根据版本号判断：0x00；~~

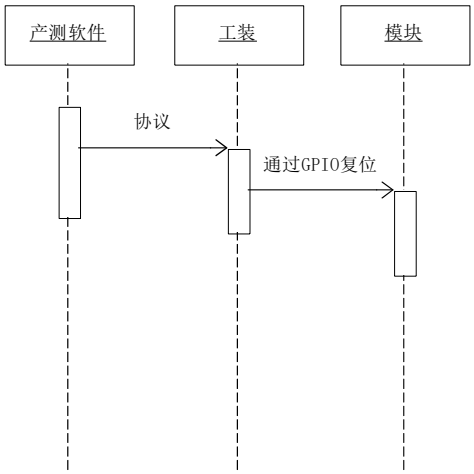
2）老的 BLE 完全弃用 0x55 0xAA 开头的方式，mesh 和 ble 全部以 66 AA 开头（产测工具不用做 55 AA 和 66 AA 交替发送操作）；

3）~~面向未来，升级一版协议：0x01，~~实现 ble 和 mesh 产测授权协议统一：

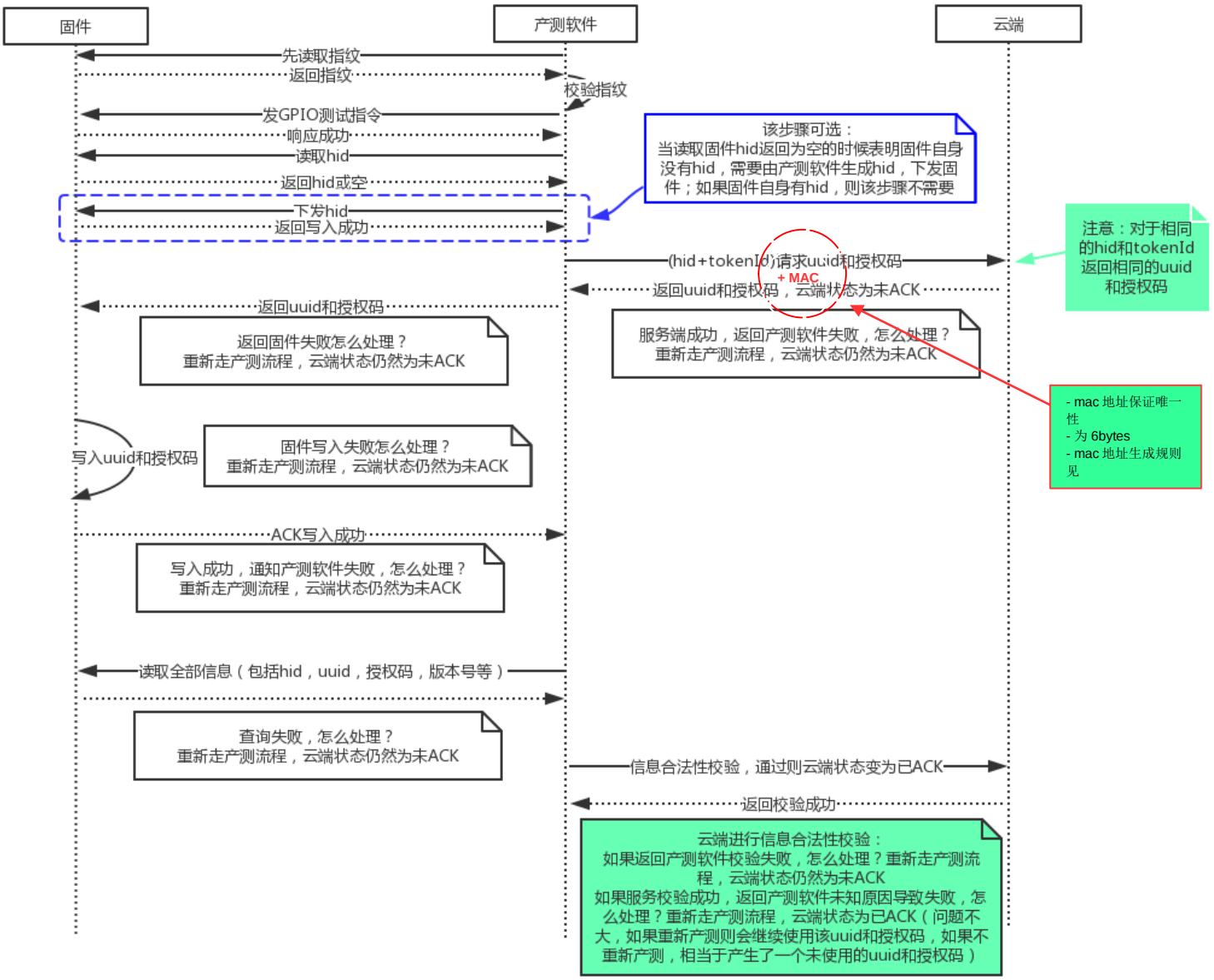
- ~~改动 3、4 两条命令实现 mesh 和 ble 统一；~~
- 增加 2-GPIO 测试失败 IO 值返回，便于查问题；
- 修改写入 9-JSON 配置文件为分包机制，支持大的 json 配置文件写入；

三、新 BLE+MESH 统一产测授权协议：

1）模块复位控制



2）产测流程

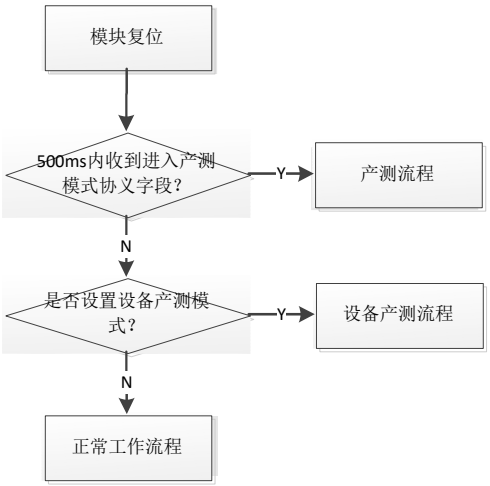


HID=...
tokenID=...
uuid=8 字节（16 位 16 进制字符串）
MAC=6 字节（12 位 16 进制字符串）
Authkey=32 字节
产测上位机给硬件的全部采用字符串形式传输（比如：MAC,发送为 12 位 16 进制字符串）

3）模块产测通信协议约定

波特率：9600
数据位：8
奇偶校验：无
停止位：1
数据流控：无

4）模块进入产测流程



5）帧格式说明

字段	长度（byte）	说明
帧头	2	固定为 0x66aa
版本	1	升级扩展用：0x00
命令字	1	具体帧类型
数据长度	2	大端
数据	xxxx	
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

说明：

- 所有大于 1 个字节的数据均采用大端模式传输。

6）产测协议

6.1、进入产测模式

产测软件：

字段	长度（byte）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0x00
数据长度	2	0x0000
数据	xxxx	无
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

模块返回：

字段	长度（byte）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0x00
数据长度	2	0x0001
数据	xxxx	0x00/0x01...
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

- 产测软件在模块上电 500ms 内通过串口发送进入产测命令，则模块进入产测流程
- 模块返回数据内容为 0x00 表示不用验证固件指纹，若数据内容为 0x01 表示需要验证固件指纹
- 返回数据右数第二 bit 为 0 表示不用从云端拉取 PID，为 1 表示用从云端拉取 PID，写入 flash
- ~~返回数据右数第三 bit 为 0 表示不用去线上拉取 OEM 的 JSON,否则需要去云端拉取并调用 json 转换工具,将其转换为 bin 文件~~
- 返回数据右数第八 bit 为 0 表示不用读取芯片唯一码、写入固件防拷贝公钥，为 1 表示需要读取芯片唯一码，写入固件防拷贝公钥

6.2、获取设备 HID

产测软件：

字段	长度（byte）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0x01
数据长度	2	0x0000
数据	xxxx	无
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

模块返回：

字段	长度（byte）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0x01
数据长度	2	表示以下字符串长度（不含 0）
数据	xxxx	{"ret":true,"hid":"xx…xxx"}/{“ret”:false}
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

说明：

- hid 固定为 19 位
- 若没有，则返回空

6.3、GPIO 测试

产测软件：

字段	长度（byte）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0x02
数据长度	2	0x0000
数据	xxxx	无
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

模块返回：

字段	长度（byte）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0x02
数据长度	2	表示以下字符串长度（不含 0）
数据	xxxx	{“ret”:true}/{“ret”:false,”result”:“x,y,...”} x,y 表示有问题的 GPIO 的序号
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

说明：

- gpio 的测试依赖于模块型号对应的工装

6.4、写入授权信息

产测软件：

字段	长度（byte）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0x03
数据长度	2	Xx
数据	xxxx	{"auzkey":"xxxx",“uuid”:“xxxx”,“mac”:”xxxxxx”,“prod_test”:xxxx,“pid”:“abcdefgh”} 注：进入产测模式返回需要从云端拉取 PID 并写入时，这里会在 prod_test 后添加一个 pid 条目；如果返回不需要从云端拉取，则写入授权信息中不需要 pid 条目
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

模块返回：

字段	长度（byte）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0x03
数据长度	2	表示以下字符串长度（不含 0）
数据	xxxx	{“ret”:true}/{“ret”:false}
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

说明：每个 id 对应唯一授权 KEY

6.5、读取模块信息

产测软件：

字段	长度（byte）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0x04
数据长度	2	0x0000
数据	xxxx	无
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

模块返回：

字段	长度（byte）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0x04
数据长度	2	表示以下字符串长度（不含 0）
数据	xxxx	{“ret”:true, "auzKey":"xxxx",”hid”:”xxx”,“uuid”:“xxxx”,“mac”:”xxxxxx”,“firmName”:”esp_12F_test”, "firmVer":"1.0.0",”prod_test”:xxxx,“pid”:“abcdefgh”}/{“ret”:false} 注：参考 6.4 是否需要回复 pid 条目
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

6.6、模块复位

产测软件：

字段	长度（byte）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0x05
数据长度	2	0x0000
数据	xxxx	无
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

模块返回：

字段	长度（byte）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0x05
数据长度	2	0x0001
数据	xxxx	0x00
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

说明：模块返回数据后复位

6.7、读取固件指纹

产测软件：

字段	长度（byte）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0x06
数据长度	2	0x0000
数据	xxxx	无
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

模块返回：

字段	长度（byte）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0x06
数据长度	2	表示以下字符串长度（不含 0）
数据	xxxx	{“ret”:true, “firmName”:”esp_12F_test”, "firmVer":"1.0.0"}/{“ret”:false}
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

说明：产测软件需要对固件返回的指纹信息做校验，即固件名称和固件版本做校验，防止用错授权 key 或烧错固件。

6.8、写入 HID

产测软件：

字段	长度（byte）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0x07
数据长度	2	Xx
数据	xxxx	{"hid":"xxxx"}
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

模块返回：

字段	长度（byte）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0x07
数据长度	2	表示以下字符串长度（不含 0）
数据	xxxx	{“ret”:true}/{“ret”:false}
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

说明：hid 由产测软件生成

6.9、RSSI 测试

产测软件：

字段	长度（byte）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0x08
数据长度	2	0x0000
数据	xxxx	无
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

模块返回：

字段	长度（byte）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0x08
数据长度	2	表示以下字符串长度（不含 0）
数据	xxxx	{“ret”:true,”rssi”:xxxx}/{“ret”:false}
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

6.10、写入线上 OEM 配置文件

产测软件：

字段	长度（byte）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0x09
数据长度	2	Xx 注： 当一包数据>249 字节时需要采用分包发送 当不需要分包发送的时候，长度的高字节为 0 当需要分包发送的时候，长度的高字节的高 4bits 表示包序号；低 4bits 表示总包数；低字节表示包长。
数据	xxxx	{"xxx":"xxxx",...}
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

模块返回：

字段	长度（byte）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0x09
数据长度	2	表示以下字符串长度（不含 0）
数据	xxxx	{“ret”:true}/{“ret”:false}
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

6.11、读取芯片唯一码

产测软件：

字段	长度（byte）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0x0A
数据长度	2	0x00 0x00
数据	0	
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

模块返回：

字段	长度（byte）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0x0A
数据长度	2	表示以下字符串长度（不含'\0'）
数据	xxxx	{“ucode”:“32453...”} //约定 16 字节 ucode，不足的话高位补 0，超过的话根据具体情况砍掉多余部分
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

6.12、上位机下发固件授权码

产测软件：

字段	长度（byte）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0x0B
数据长度	2	表示以下字符串长度（不含’\0’）
数据	n	{“key”:”xxxxxxxxxxxxx”}
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

模块返回：

字段	长度（byte）	说明
帧头	2	0x66aa
版本	1	0x00
命令字	1	0x0B
数据长度	2	表示以下字符串长度（不含’\0’）
数据	n	{“ret”:true}/{“ret”:false}
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

说明： 上位机根据 ucode 上报给服务器，服务器生成授权码下发
举例：ucode 是：0a0b0c0d01020304，那么上位机会上报 0a0b0c0d01020304tuya_ble 给服务器（0a0b0c0d01020304ble 是一个字符串）