NAP3代蓝牙私有通信协议文档

|  |  |  |  |
| --- | --- | --- | --- |
| 版本 | 内容 | 时间 | 修订人 |
| V0.3 | 建立正式版本 | 2017/12/1 |  |
| V1.0 | 补充DFU和部分指令 | 2018/1/5 | 刘运强 |
|  |  |  |  |
|  |  |  |  |

# 协议内容

蓝牙协议阐述了蓝牙设备与眼罩的通信与数据交互的基本细节和特性。包括蓝牙服务、数据格式等。

# 名词解释与约定

* 数据使用大端模式
* 广播UUID：0000FF00-1212-ABCD-1523-785FEABCD123

# 蓝牙服务列表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Service&Character | UUID | Properties |  | 数据包大小 |
| Generic Access | 0000**1800**-0000-1000-8000-00805F9B34FB |  | GAP |  |
| BLE\_UUID\_GAP\_CHARACTERISTIC\_DEVICE\_NAME | 0000**2A00**-0000-1000-8000-00805F9B34FB | Read & Write | 设备名称 |  |
| BLE\_UUID\_GAP\_CHARACTERISTIC\_APPEARANCE | 0000**2A01**-0000-1000-8000-00805F9B34FB | Read | 设备类型 |  |
| BLE\_UUID\_GAP\_CHARACTERISTIC\_PPCP | 0000**2A04**-0000-1000-8000-00805F9B34FB | Read | 连接参数 |  |
|  | | | | |
| Generic Attribute | 0000**1801**-0000-1000-8000-00805F9B34FB |  | GATT |  |
| BLE\_UUID\_GATT\_CHARACTERISTIC\_SERVICE\_CHANGED | 0000**2A05**-0000-1000-8000-00805F9B34FB | Indicate | Service change |  |
| CCCD | 0000**2902**-0000-1000-8000-00805F9B34FB |  |  |  |
|  | | | | |
| BLE\_UUID\_CONNECT\_SERVICE | 0000**FF10**-1212-abcd-1523-785feabcd123 |  | 连接服务 |  |
| BLE\_UUID\_ID\_down\_CHARACTERISTIC | 0000**FF11**-1212-abcd-1523-785feabcd123 | Write | 手机下发ID | 见设备（五：握手流程） |
| BLE\_UUID\_Shakehands\_CHARACTERISTIC | 0000**FF12**-1212-abcd-1523-785feabcd123 | Write、Notify | 三步握手 |
| BLE\_UUID\_State\_up\_CHARACTERISTIC | 0000**FF13**-1212-abcd-1523-785feabcd123 | Notify | 连接状态上传 |
|  | | | | |
| BLE\_UUID\_COMMAND\_SERVICE | 0000**FF20**-1212-abcd-1523-785feabcd123 |  | 指令传输服务 |  |
| BLE\_UUID\_COMMAND\_UPLOAD\_ CHARACTERISTIC | 0000**FF21**-1212-abcd-1523-785feabcd123 | Write | 指令下行 | 手机下发指令，数据定义见表4-1 |
| BLE\_UUID\_COMMAND\_DOWNLOAD\_ CHARACTERISTIC | 0000**FF22**-1212-abcd-1523-785feabcd123 | Notify | 指令上行 |  |
| CCCD | 0000**2902**-0000-1000-8000-00805F9B34FB |  |  |  |
|  | | | |  |
| BLE\_UUID\_BATTERY\_SERVICE | 0000**180F**-0000-1000-8000-00805F9B34FB |  | 电池电量服务 |  |
| BLE\_UUID\_BATTERY\_LEVEL\_ CHARACTERISTIC | 0000**2A19**-0000-1000-8000-00805F9B34FB | Notify & Read | 电池电量 | 电池电量上传，数据格式见表格4-2 |
| CCCD | 0000**2902**-0000-1000-8000-00805F9B34FB |  |  |  |
|  | | | |  |
| BLE\_UUID\_EEG\_SERVICE | 0000**FF30**-1212-abcd-1523-785feabcd123 |  | 脑电服务 |  |
| BLE\_UUID\_EEG\_DATA\_ CHARACTERISTIC | 0000**FF31**-1212-abcd-1523-785feabcd123 | Notify | 脑电数据 | 脑电数据上传，数据格式见表格4-3 |
| BLE\_UUID\_CONTACT\_DATA\_ CHARACTERISTIC | 0000**FF32**-1212-abcd-1523-785feabcd123 | Notify & Read | 脱落检测 | 脱落数据上传，数据格式见表格4-4 |
| CCCD | 0000**2902**-0000-1000-8000-00805F9B34FB |  |  |  |
|  | | | | |
| BLE\_DFU\_SERVICE\_UUID | 0000**FF40**-1212-abcd-1523-785feabcd123 |  | DFU 服务 |  |
| BLE\_DFU\_CTRL\_PT\_UUID | 0000**FF41**-1212-abcd-1523-785feabcd123 | Notify & Write | DFU控制指令 | DFU流程指令交互，具体见DFU升级 |
| CCCD | 0000**2902**-0000-1000-8000-00805F9B34FB |  |  |  |
| BLE\_DFU\_PKT\_CHAR\_UUID | 0000**FF42**-1212-abcd-1523-785feabcd123 | Write no response | DFU数据包 | DFU数据包下发，具体见DFU升级 |
|  | | | | |
| BLE\_UUID\_DEVICE\_INFORMATION\_SERVICE | 0000**180A**-0000-1000-8000-00805F9B34FB |  | 设备信息服务 |  |
| BLE\_UUID\_SERIAL\_NUMBER\_STRING\_ CHARACTERISTIC | 0000**2A25**-0000-1000-8000-00805F9B34FB | Read | 序列号 | 32字节，具体见Device ID和SN码规则 |
| BLE\_UUID\_MODEL\_NUMBER\_STRING\_CHAR | 0000**2A24**-0000-1000-8000-00805F9B34FB | Read | MAC | 6字节从机设备MAC地址 |
| BLE\_UUID\_FIRMWARE\_REVISION\_STRING\_ CHARACTERISTIC | 0000**2A26**-0000-1000-8000-00805F9B34FB | Read | 固件版本 | 当前固件版本号：x：x：x |
| BLE\_UUID\_HARDWARE\_REVISION\_STRING\_ CHARACTERISTIC | 0000**2A27**-0000-1000-8000-00805F9B34FB | Read | 硬件版本 | 当前硬件版本号：x：x：x |
| BLE\_UUID\_MANUFACTURER\_NAME\_STRING\_ CHARACTERISTIC | 0000**2A29**-0000-1000-8000-00805F9B34FB | Read | 制造商 | 制造商信息：公司ID+自定义数据  （暂时只在工厂测试模式下使用） |

# 数据格式与定义补充说明

表4-1

|  |  |
| --- | --- |
| App向Nap发出采集1291数据 | 0x01 |
| App向Nap发出停止采集1291数据 | 0x02 |

表4-2

|  |  |
| --- | --- |
| 单字节，以3.1V作为0%，4.1V作为100%发送电池电压 | 0x34-----3.62V |

表4-3

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 数据包序号 | 第一个数据 | 第二个数据 | 第三个数据 | 第四个数据 | 第五个数据 | 预留 |
| 2个字节 | 3个字节 | 3个字节 | 3个字节 | 3字节 | 3个字节 | 3个字节 |
| 00-00 | 00-01-02 | 03-04-05 | 06-07-08 | 09-0A-0B | 0C-0D-0E |  |

备注：脑电服务的脑电数据特性（UUID：FF31）一次会传输 5个脑电数据，每个数据是24位量化。

如上表：

第一个数据为：0x000102

第二个数据为：0x030405以此类推

表4-4

|  |  |
| --- | --- |
| 三个电极佩戴正常 | 0x00 |
| 活动电极佩戴不正常 | 0x08 |
| 参考电极佩戴不正常 | 0x10 |
| 活动电极和参考电极都不正常 | 0x18 |

# 设备握手流程

* 1. 数据包格式：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| UUID | 包头 | 数据 | | | | 备注 |
| FF11 | 0x00 | 00 | 00 | 00 | 01 | ID 四个字节  ID：00000001 |
| FF12 | 0x01 | 时 | 分 | 秒 | 随机数 | 时间+随机数 |
| FF12 | 0x02 | 时 | 分 | 秒 | 随机数 | 时间+随机数 |
| FF12 | 0x03 | 时 | 分 | 秒 | 随机数 | 时间+随机数 |
| FF13 | 0x04 | 00 | 00 | 00 | State | 握手状态返回 |

* 1. State定义：

|  |  |
| --- | --- |
| state |  |
| 0x00 | 握手成功 |
| 0x01 | ID信息接收错误 |
| 0x02 | 设备已被绑定 |
| 0x03 | 未收到ID信息 |
| 0x04 | 握手数据错误 |
| 0x05 | 时间错误 |

* 1. 实现流程：

1. 打开app，登录账号后，检查账号是否绑定了眼罩设备，如果是，根据UUID+MAC扫描设备，并在扫描成功后发起连接。若否，根据UUID扫描设备，然后选择信号最好的设备发起连接。
2. 在UUID+MAC扫描并连接到设备后，APP端先向设备下发ID（四个字节，特征UUID为FF11）并读取MAC（特征UUID为2A24），然后进行三步握手，如果握手成功，则连接建立完成，如果握手失败，根据错误代码执行下一步操作。
3. 在UUID扫描并连接到设备后，APP端先向设备下发ID（四个字节，特征UUID为FF11）并读取MAC（特征UUID为2A24），然后进行三步握手，如果握手成功，则连接建立完成，并保存MAC信息；如果握手失败，根据错误代码执行下一步操作。
4. **三步握手流程：**（特征UUID为FF12）

第一步：APP端读取当前时分秒并获取一个随机数，分别于时分秒异或得到需要发送的数据：例如时分秒为16时38分49秒，转换为16进制为0x10：0x25：0x31，随机数为138，转换为16进制为0x8A，异或得到0x9A：0xAF：0xBB，所以下发的数据包格式为0x01：0x9A：0xAF：0xBB：0x8A。

第二步：设备端收到数据包后将0x9A：0xAF：0xBB分别与0x8A异或，得到0x10：0x25：0x31，并重新生成随机数如0x53，然后时间与随机数进行异或操作。得到0x43：0x76：0x62，所以数据包格式为：0x02：0x43：0x76：0x62：0x53。

第三步：APP端收到数据包后将0x43：0x76：0x62分别与0x53异或，得到0x10：0x25：0x31，并重新生成随机数如0x49，然后时间与随机数进行异或操作。得到0x59：0x6C：0x78，所以数据包格式为：0x03：0x59：0x6C：0x78：0x49。

1. 如果眼罩设备在发出错误的状态指令（0x01-0x05）后会在0.5s后与主机断开连接。
2. 在与眼罩设备物理层建立连接后的30s内如果握手没有成功，眼罩设备会断开连接。

# 各种连接状态列举：

1. 未绑定设备账号+未绑定设备：手机登录新账号后连接未绑定过的设备，下发ID，读取MAC，然后完成三步握手，连接成功。（第一次使用的情况）
2. 未绑定设备账号+已绑定设备：手机登录新账号后连接已绑定过的设备，下发ID，读取MAC，设备端判断出ID错误，发送0x02，0.5s后断开连接。
3. 已绑定设备账号+未绑定设备：手机登录新账号后会根据MAC筛选设备，所以无法找到新设备，如果长时间未找到新设备，需要提示用户如果连接新设备需要删除旧设备信息。（老用户更换眼罩的情况）
4. 已绑定设备账号+已绑定设备：手机登录旧账号后会根据MAC筛选设备，然后找到旧设备发下ID，读取MAC，然后完成三步握手，连接成功。（后续使用的情况）
5. 已绑定设备账号+已绑定其他设备账号：手机登录旧账号后会根据MAC筛选设备，然后找到旧设备发下ID，读取MAC，设备端判断出ID错误，发送0x02，0.5s后断开连接。（眼罩外借的情况）

# 固件升级

固件升级对应了固件升级服务（BLE\_DFU\_SERVICE\_UUID）。固件升级流程主要使用的原厂的 DFU 流程，但是前提是使用上面我们自己定义的 UUID。

* 1. **固件升级功能说明：**

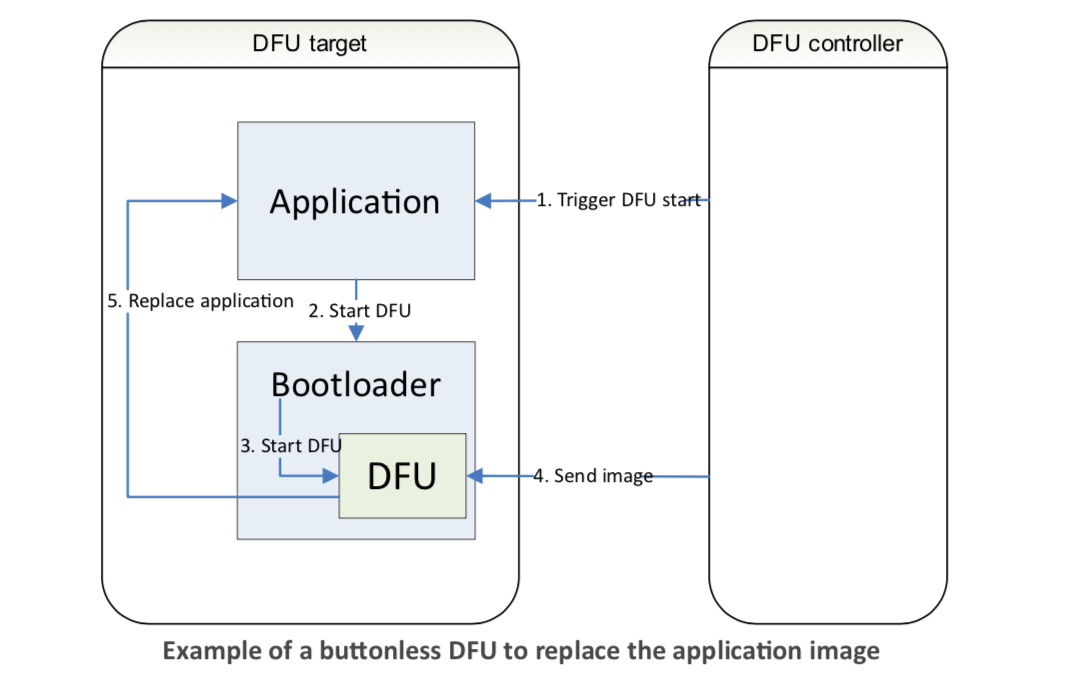
DFU： device firmware update（固件更新）

OTA: Over the Air Technology（空中升级）

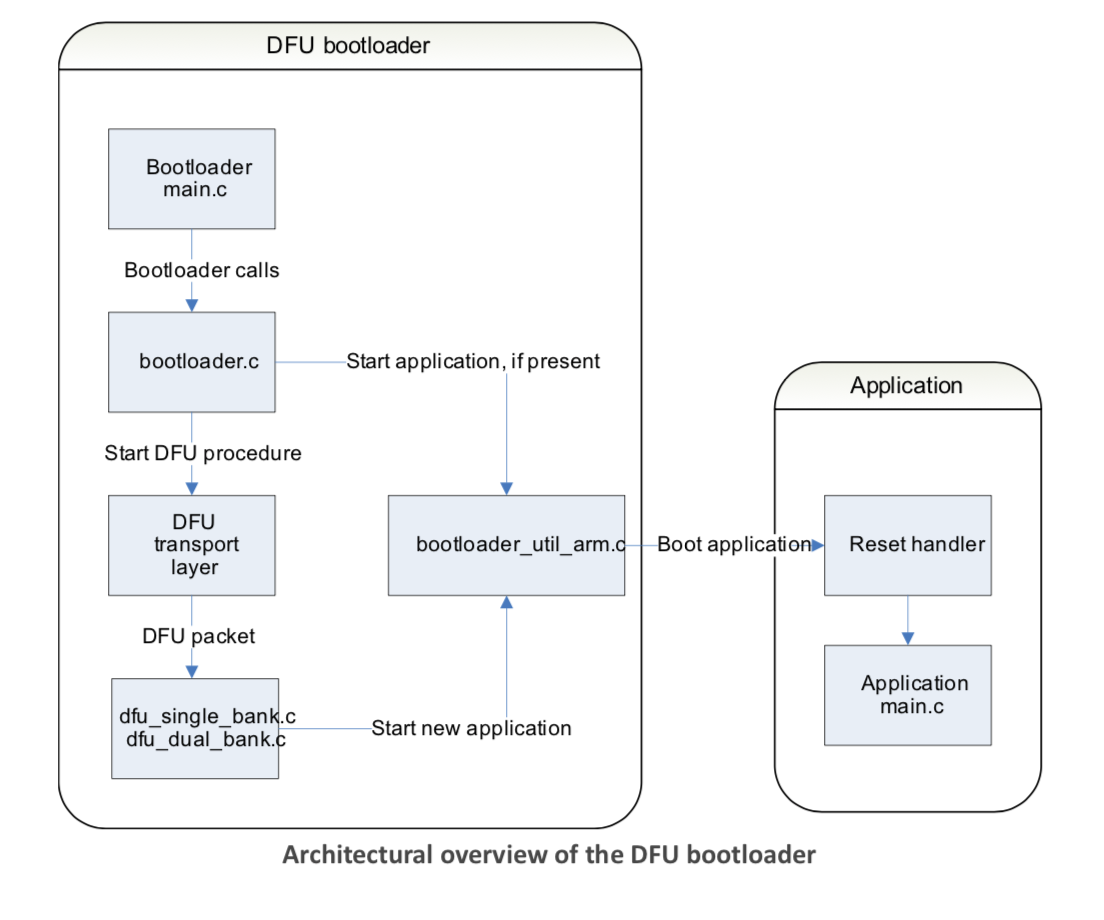
根据 flash 的特性，我们用新的固件包来替代原有的硬件程序，以此来升级固件程序。

固件升级有以下好处：

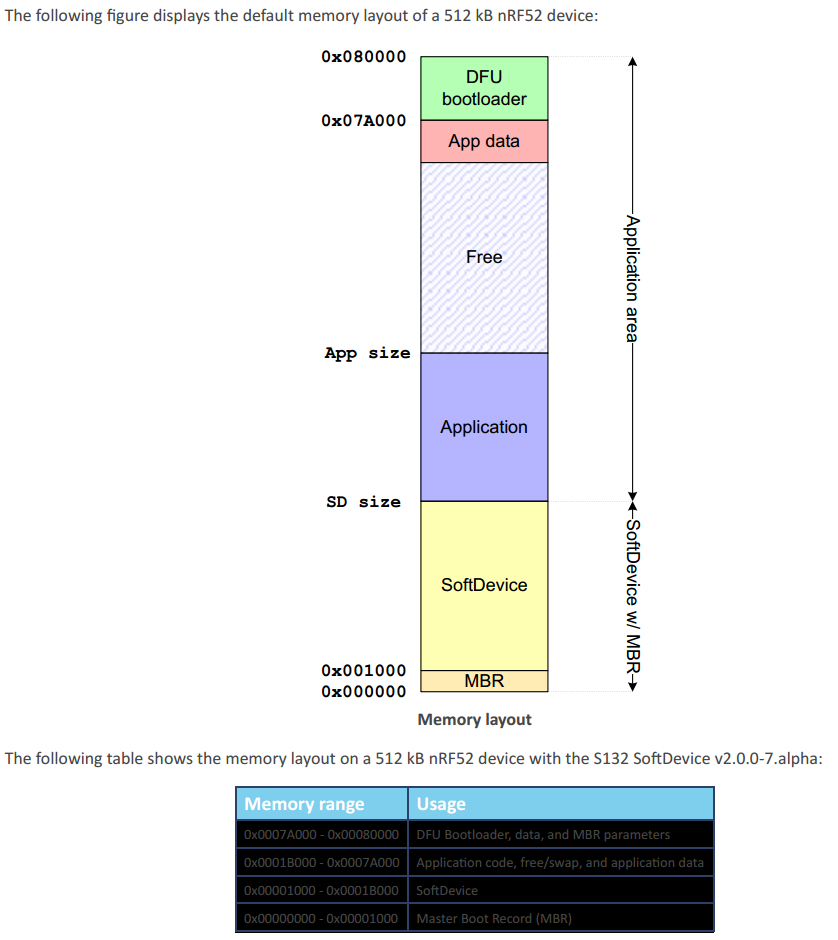
1. 解决已经存在的bug、兼容性问题。
2. 优化产品和升级产品功能。
   1. **DFU过程说明：**

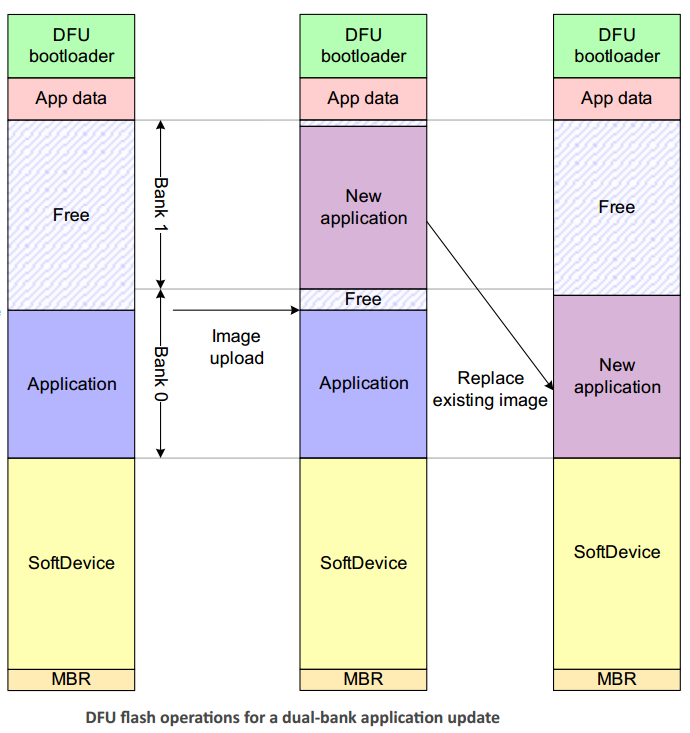
****

* 1. **DFU过程BootLoader流程说明：**

****

* 1. **DFU flash结构图：**



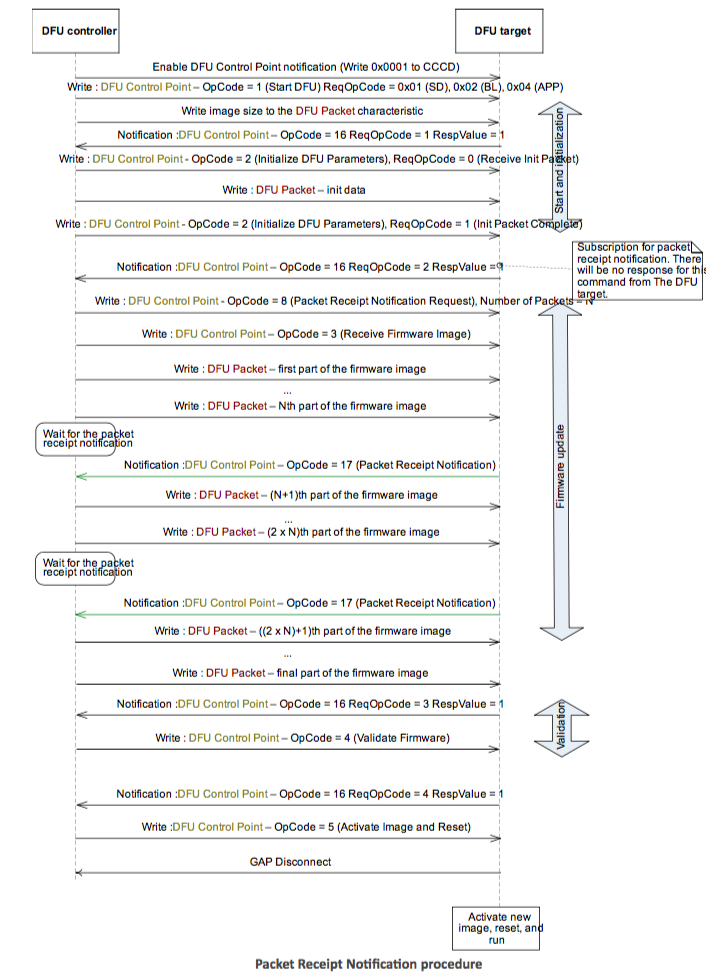


* 1. **DFU 程序实现**

要在客户端实现DFU，需要有以下几个条件：

1. 硬件支持 DFU 。
2. 蓝牙设备先处于连接状态
3. 有可烧入的升级固件包。

下面为DFU 的数据交互：



* 1. **开发库**

**1. iOS 芯片官方github Demo ：**[**https://github.com/NordicSemiconductor/IOS-nRF-Toolbox**](https://github.com/NordicSemiconductor/IOS-nRF-Toolbox)

**2. pod ‘iOSDFULibrary’, ‘~> 4.0’**

**3. iOS库修改 uuid 的地方：**由于 DFU 服务的 UUID 与官方不一样，需要修改库里面官方提供的 UUID，将下面对应类里面的 UUID 分别改成我们相应的 UUID 就行了：

* DFUControlPoint改成： 0000**FF41**-1212-abcd-1523-785feabcd123
* DFUPacket改成：0000**FF42**-1212-abcd-1523-785feabcd123
* DFUVersion改成：0000**FF44**-1212-abcd-1523-785feabcd123
* LegacyDFUService改成： 0000**FF40**-1212-abcd-1523-785feabcd123

**Android 芯片官方 github Demo：**[**https://github.com/NordicSemiconductor/Android-nRF-Toolbox**](https://github.com/NordicSemiconductor/Android-nRF-Toolbox)

build.gradle 文件加入以下内容：

implementation 'no.nordicsemi.android:dfu:1.6.1'

自己 fork 库，，然后修改 uuid 相关内容即可。UUIDHelper 修改相关的 UUID 即可：

* ControlPoint改成： 0000**FF41**-1212-abcd-1523-785feabcd123
* Packet改成：0000**FF42**-1212-abcd-1523-785feabcd123
* Version改成：0000**FF44**-1212-abcd-1523-785feabcd123
* Service改成： 0000**FF40**-1212-abcd-1523-785feabcd123

# 工厂测试模式指令

* 1. **数据交互协议**

|  |  |  |  |
| --- | --- | --- | --- |
| Nap—Tool--1 | | Tool—Nap--2 | |
| Nap已进入工厂测试模式 | 0x10 | Tool正在给Nap充电中状态 | 0x21 |
| Nap充电中 | 0x11 | Tool给Nap充电完成状态 | 0x22 |
| Nap充电完成 | 0x12 |  |  |
| Nap充电中状态获取错误 | 0x13 |  |  |
| Nap充电完成状态获取错误 | 0x14 |  |  |
| Nap连接App成功 | 0x15 |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Nap—App--3 | | App—Nap--4 | |
| Nap上传Device ID | 0x31+Device ID | App向Nap写入Device ID | 0x41+DeviceID |
| Nap上传SN | 0x32+SN | App向Nap写入SN | 0x42+SN |
| Nap上传UserID | 0x33+UserID | App向Nap删除UserID指令 | 0x43+UserID |
| Nap按键被按下 | 0x34 | App向Nap发出LED测试指令 | 0x44 |
|  |  | App向Nap发出关机指令 | 0x45 |
|  |  | App向Nap发出采集1291数据 | 0x46 |
|  |  | App向Nap发出停止采集1291数据 | 0x47 |

|  |  |  |  |
| --- | --- | --- | --- |
| Tool—App--5 | | App—Tool--6 | |
| Tool连接Nap成功 | 0x50 | Tool脱落信号输出 | 0x61 |
| Tool前端信号输出成功 | 0x51 | 充电输出关闭 | 0x62 |
| Nap充电电流 | 0x52+电流（mA） |  |  |
| Nap充电完成电流 | 0x53+电流（mA） |  |  |
| Nap右腿电压 | 0x54+电压 |  |  |
| Nap充电中测试正常 | 0x55 |  |  |
| Nap充电完成测试正常 | 0x56 |  |  |
| Nap充电测试失败 | 0x57 |  |  |
| ADV的制造商数据 | 0x58+0xXX(两次) |  |  |

* 1. **数据传输通道**
* Nap和Tool数据交互使用串口
* Nap和App数据交互使用蓝牙的命令服务（UUID为FF20）。
* Tool和App数据交互使用蓝牙的命令服务（UUID为FF20）。
* 脱落信号和脑电数据走脑电服务。