文档编号：

版本：V1.0

生效日期：2018-01-17

深圳市通力科技开发有限公司

TCL通力电子（惠州)有限公司

西安TCL软件开发有限公司



**文件传输协议说明**

**通力电子股份有限公司 版权所有**

**内部资料 注意保密**

文档修订记录

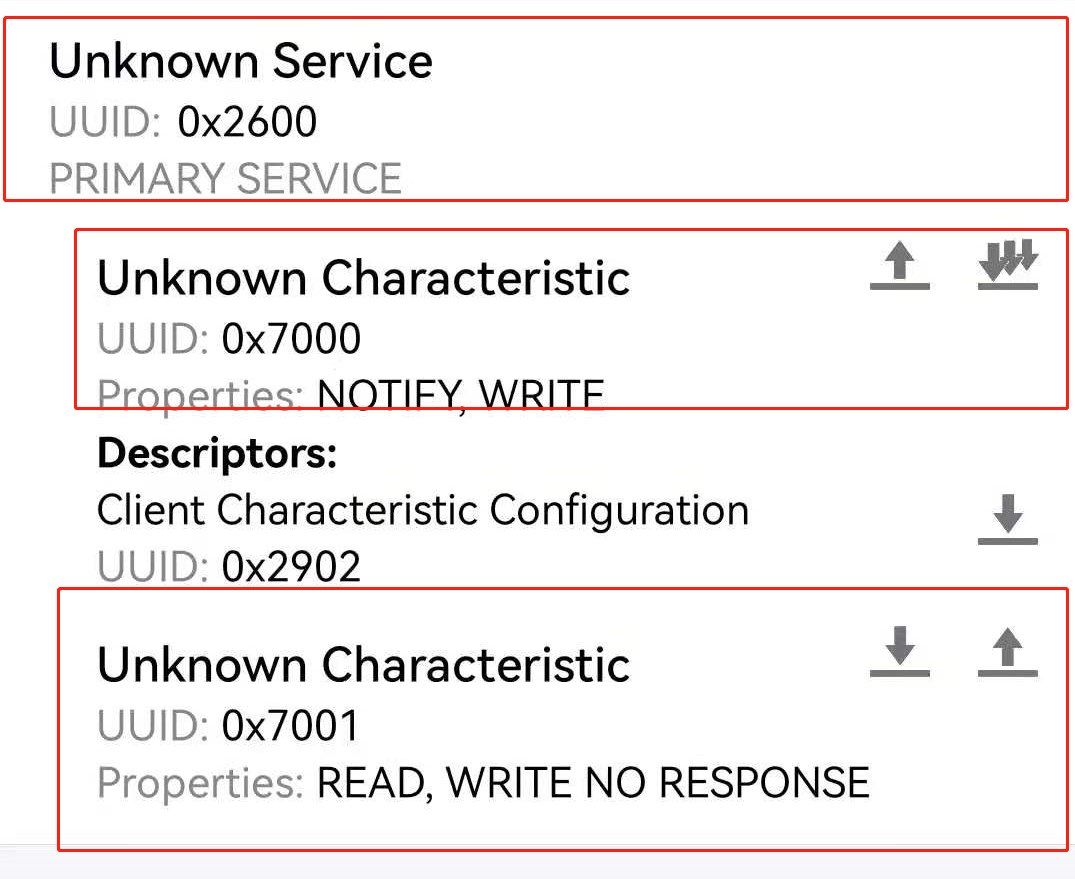
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **序号** | **版本号** | **变化状态** | **变更（+/-）说明** | **作者** | **日期** |
| 1 | V0.1 | C | 初版 | 骆惠文 | 2021-7-8 |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

\*变化状态：C――创建，A——增加，M——修改，D——删除

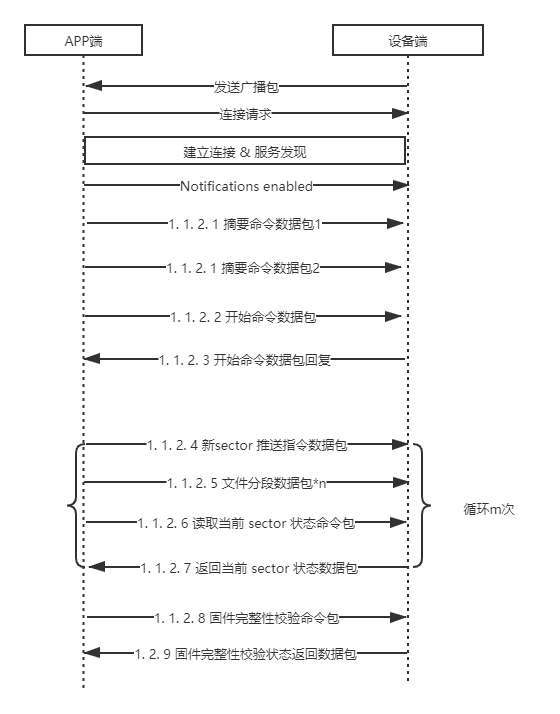
**1. 1. 1 文件传输 Profile** 定义

文件传输Profile包括一个主服务（primary service）,该服务下包含两个特征值（characteristic），一个用于传递控制信息（以下简称控制特征值），一个用于数据传输（以下简称数据特征值），其中用于传递控制信息的特征值有一个特征值描述符（characteristic descriptor）。

主服务的UUID 为0x2600,控制特征值的UUID 为0x7000，数据特征值的UUID为0x7001。主服务及特征值相关的参数定义如下：



图表 1-1 Characteristic

****

文件传输交互流图

**1. 1. 2** 包格式定义  
文件传输过程中，不特殊指定的前提下，所有数据包默认 packed，不考虑数据结构内部对齐。

字节序：Little-End低字节在前  
以下命令，红色为手机到设备，蓝色为设备到手机。

首先需要Notifications enabled,即向控制特征值(UUID 为0x7000)的Client Characteristic Configuration写入0x0100。

文件分为多个sector(4096byte一个)发送，每个sector分为多个segment发送；

**1. 1. 2. 1** 摘要命令数据包

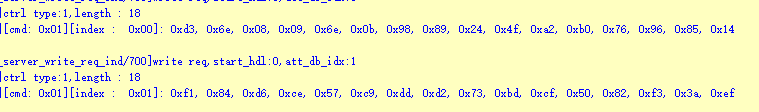
|  |  |  |  |
| --- | --- | --- | --- |
| 方向 | 目标特征值/描述符 | GATT 命令 | 长度 |
| 手机->设备 | 控制特征值（UUID=0x7000） | write request | 18 bytes |

用途：手机端将待传输文件的摘要信息发送到设备。摘要信息共32bytes，分别拆成两个16bytes 发送，外加1byte的idx和1byte的cmd，每个包共计18bytes。

摘要信息：使用**sha256算法**来计算文件的完整性。

格式：  
Struct digest\_cmd\_req  
{  
 uint8\_t cmd = 0x01;  
 uint8\_t idx;  
 uint8\_t digest[16];  
};

格式说明：  
cmd:0x01  
idx: 当前摘要的id，0或1。0表示低 16bytes摘要，1表示高16bytes摘要。  
digest: 摘要数据



example（Big-Endian）：

第一包：0x01 0x00 0xd3, 0x6e, 0x08, 0x09, 0x6e, 0x0b, 0x98, 0x89, 0x24, 0x4f, 0xa2, 0xb0, 0x76, 0x96, 0x85, 0x14

第二包：0x01 0x01 0xf1, 0x84, 0xd6, 0xce, 0x57, 0xc9, 0xdd, 0xd2, 0x73, 0xbd, 0xcf, 0x50, 0x82, 0xf3, 0x3a, 0xef

**1. 1. 2. 2** 开始命令数据包（文件信息数据包）

|  |  |  |  |
| --- | --- | --- | --- |
| 方向 | 目标特征值/描述符 | GATT 命令 | 长度 |
| 手机->设备 | 控制特征值（UUID=0x7000） | write request | 13 bytes |

用途：手机端将文件的必要信息推送到设备。

格式：  
Struct start\_cmd\_req  
{  
 uint8\_t cmd = 0x02;  
 uint8\_t file\_index\_ complete\_flag[4];  
 uint32\_t new\_file\_size;  
 uint32\_t segment\_data\_max\_length;  
};

格式说明：  
cmd:0x02  
file\_index\_ complete\_flag（Little-Endian）:

data[0]:图片索引(0x00 ~ 0x09)

data[1]:预留；

data[2]: 预留；

data[3]: 传输模式: 0x01:文件传输模式  
new\_file\_size: 文件大小（单位为 byte）  
segment\_data\_max\_length : 文件数据包实际有效数据长度，推荐设置为MTU-4



example: 0x02,0x04, 0x00, 0x00, 0x01, 0xe0, 0xc3, 0x01, 0x00, 0xfc, 0x01, 0x00, 0x00

**1. 1. 2. 3** 开始命令数据包回复

|  |  |  |  |
| --- | --- | --- | --- |
| 方向 | 目标特征值/描述符 | GATT 命令 | 长度 |
| 设备->手机 | 控制特征值（UUID=0x7000） | Notification | ≥2 bytes |

用途：设备端向手机端回复开始命令状态  
Struct start\_cmd\_rsp  
{  
 uint8\_t rsp = 0x03;  
 uint8\_t status;  
};

格式说明：  
rsp: 0x03  
status: 开始命令状态回复,0为成功,非0为失败

注：设备实际回复的数据长度可能超过 2 bytes，手机端只需要关心前面 2 bytes 即可，多余的数据不必处理

**1. 1. 2. 4** 新**sector** 推送指令数据包

|  |  |  |  |
| --- | --- | --- | --- |
| 方向 | 目标特征值/描述符 | GATT 命令 | 长度 |
| 手机->设备 | 控制特征值（UUID=0x7000） | write request | 3 bytes |

用途：手机端通知设备为接收sector 数据做准备。

Struct erase\_new\_sector\_cmd  
{  
 uint8\_t cmd = 0x04;  
 uint16\_t sector\_id;  
};  
格式说明：  
cmd: 0x04

sector\_id: 下一个sector id，从0开始计数，最大值为CEILING(image\_size, 4096)，  
其中 CEILING 的定义为：  
#define CEILING(dividend,divisor) ((dividend)/(divisor) + (((dividend)%(divisor))?1:0))

example :



sector id 0:0x04 ,0x00, 0x00



sector id 1:0x04 ,0x01, 0x00

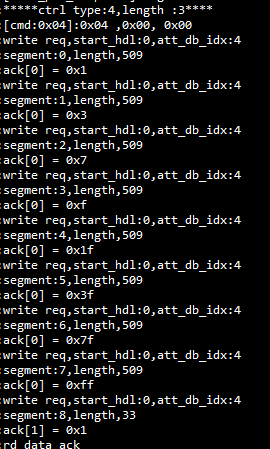
**1. 1. 2. 5 文件**分段数据包

|  |  |  |  |
| --- | --- | --- | --- |
| 方向 | 目标特征值/描述符 | GATT 命令 | 长度 |
| 手机->设备 | 控 制 特 征 值 （UUID=0x7001） | write request | ≤segment\_data\_max\_length+ 1 |

用途：发送一个segment数据。把sector拆成多个segment进行发送。  
Struct segment\_data\_packet  
{  
 uint8\_t segment\_id;  
 uint8\_t data\_array[array\_size];  
};

格式说明：  
 segment\_id: 当前segment在 sector 内部的序号，从0开始计数，递增；

取值范围是[0, CEILING(4096,segment\_data\_max\_length)]。

如左图所示：segment\_data\_max\_length = 509 -1；

segment\_id：0 ~8；

**1. 1. 2. 6** 读取当前 **sector** 状态命令包

|  |  |  |  |
| --- | --- | --- | --- |
| 方向 | 目标特征值/描述符 | GATT 命令 | 长度 |
| 手机->设备 | 控制特征值（UUID=0x7001） | read request | 0 bytes |

用途： 当手机端把一个sector 的数据全部推送到设备后，需要读取各个拆分数据包（segment data packet）的状态，以确认 4096 bytes 的数据设备都已经接收到了。  
Struct read\_segments\_status\_cmd  
{  
 // empty  
};

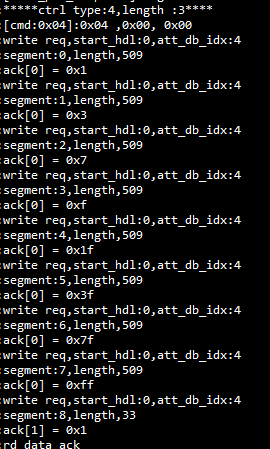
**1. 1. 2. 7** 返回当前 **sector** 状态数据包

|  |  |  |  |
| --- | --- | --- | --- |
| 方向 | 目标特征值/描述符 | GATT 命令 | 长度 |
| 设备->手机 | 控制特征值（UUID=0x7001） | read response | 27 bytes |

用途：设备端向手机端回复开始命令状态  
Struct read\_segments\_status\_rsp  
{  
 uint8\_t ack\_status[array\_size];  
};  
格式说明：  
ack\_status: 每一个 bit 代表一个segment的ack，1：已收到，0：需要重发。

segment\_id 0 对应的ack status位是最低位；

array\_size最大值是27.



如左图所示，segment0~8的ack\_status均为1即ack

**1. 1. 2. 8** 固件完整性校验命令包

|  |  |  |  |
| --- | --- | --- | --- |
| 方向 | 目标特征值/描述符 | GATT命令 | 长度 |
| 手机->设备 | 控制特征值（UUID=0x7000） | write request | 1 byte |

用途：所有文件数据推送完毕后,手机端通知设备进行数据一致性校验  
Struct image\_integrity\_check\_req  
{  
 uint8\_t cmd = 0x05;  
};  
格式说明：  
 cmd: 0x05

**1. 2. 9** 固件完整性校验状态返回数据包（**文件传输**完成数据包）

|  |  |  |  |
| --- | --- | --- | --- |
| 方向 | 目标特征值/描述符 | GATT 命令 | 长度 |
| 设备->手机 | 控制特征值（UUID=0x7000） | Notification | ≥ 2 bytes |

用途：设备端向手机端回复开始命令状态  
Struct image\_integrity\_check\_rsp  
{  
 uint8\_t rsp;  
 uint8\_t integrity\_check\_result;  
};  
格式说明：  
rsp: 0x06  
integrity\_check\_result: 完整性校验结果，0表示成功，非0表示失败

注：设备实际回复的数据长度可能超过 2 bytes，手机端只需要关心前面 2 bytes 即可，多余的数据不必处理。