[1. 名词解释 1](#_Toc479253800)

[2. 开始运动 1](#_Toc479253801)

[3. 校准 1](#_Toc479253802)

[4. 真正开始运动 2](#_Toc479253803)

[5. 结束运动 2](#_Toc479253804)

[6. 同步数据 2](#_Toc479253805)

[6.1 开始同步数据： 2](#_Toc479253806)

[6.2 同步数据包： 2](#_Toc479253807)

## 名词解释

|  |  |  |  |
| --- | --- | --- | --- |
| 数据项 | 处理端 | 算法 | 备注 |
| 总距离 | BLE输出 |  |  |
| 总步数 | BLE输出 |  | 包括：总跑步数，总冲刺数，总步行数；其中只有总步行数没有直接给出，其余均由BLE直接提供 |
| 纵向距离 | APP计算 | 总距离-横向距离 |  |
| 横向距离 | BLE输出 |  |  |
| 时间 |  |  |  |
| 总时间 | APP处理 |  |  |
| 活跃时间 | APP处理 | 总跑次数\*每次跑的单位时间+冲刺数量\*每次冲刺的单位时间 | 数据项均由BLE提供 |
| 活跃时间比 | APP处理 | 活跃时间/总时间 |  |
| 速度 |  |  |  |
| 平均移动速度 | APP处理 | 总距离/运动时间 | 运动时间：总跑次数\*每次跑的单位时间+冲刺数量\*每次冲刺的单位时间+步行数\*每步的单位时间；单位时间相关的数值由BLE提供 |
| 最大移动速度 | BLE输出 |  |  |
| 冲刺次数 | BLE输出 |  |  |
| 变向次数 | BLE输出 |  |  |
| 变向平均触地时间 | BLE输出 |  |  |
| 纵跳 |  |  | 【纵跳高度， 滞空时间， 着地翻转角度】组成的BYTE数组 |
| 纵跳次数 | BLE输出 |  |  |
| 平均纵跳高度 | APP计算 | N次跳跃高度的和/跳跃次数 |  |
| 平均滞空时间 | APP计算 | N次纵跳时间的和/纵跳次数 |  |
| 平均着地翻转角度 | APP计算 | N次翻转角度的和/着地翻转次数 |  |
| 着地姿势 | APP计算 | 统计得出。算法见备注说明 | 对于右脚而言，着地内翻角度为负数为外翻，正表示内翻，需要统计外翻和内翻哪个占多数来决定。APP需要设置左右脚。对于左脚数值与右脚相反 |

以下协议数据传输建立在蓝牙正常连接

## 开始运动

app发送ff01

ble回复

ff0100 表示没有数据需要同步

ff0101 表示有数据需要同步

## 校准

App发送ff69

Ble收到后读取校准结果

ble回复 ff69xx

xx==03 代表校准成功,如：ff6903

xx为其他值均表示未校准成功(默认是00)

## 4. 真正开始运动

app发送 ff08 userId createtime longitude latitude

示例：

四字节userid 四字节createtime 四字节longitude 四字节 latitude

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ff08 | 00003002 | 000e0300 | Ff040300 | 0f804000 |

ble回复ff08ff

## 5结束运动

App主动结束运动时

app发送ff07

ble回复 ff07ff

注：如APP未结束运动，ble会在特定的情况下自动结束运动

## 6同步数据

### 6.1 开始同步数据：

app发送ff09

ble回复ff09ff

然后ble开始发送数据

### 6.2 同步数据包：

<------------------数据包1---------->

<-------------------数据包2-------->

<-------------------....------------->

<------------------数据包n-------------->

<----------------656e64-------------->

注：Ble根据蓝牙协议规定大小来定的数据包个数，656e64 表示蓝牙数据包传输结束

数据包格式 0--n个字节数据 (小端字节序)

完整数据包，例如：

03020000 03030404 0e050403 04040302 03020000 04040303 0e050403 04040302 03020000 04040303 0e050403 04040302

#### 0--47 (前48个字节)

总计12个数据,每个数据占4个字节

总步数(long)+总跑步数(long)+总冲刺数(long)+横向步数(long)+横向跑步数(long)+横向冲刺数(long)+总距离(float,米)

+横向距离(float,米)+最大移动速度(float m/s)+变向次数(long)+变向平均触地时间(long,毫秒)+纵跳次数(long)

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 03020000 | 03030404 | 0e050403 | 04040302 | 03020000 | 04040303 | 0e050403 | 04040302 | 03020000 | 04040303 | 0e050403 | 04040302 |
| 总步数 | 总跑步数 | 总冲刺数 | 横向步数 | 横向跑步数 | 横向冲刺数 | 总距离 | 横向距离 | 最大移动速度 | 变向次数 | 变向平均触地时间 | 纵跳次数 |

#### 48—64（第48字节到第64字节）

APP传入到ble的user先关的原始数据（见4.发送用户运动id指令）

|  |  |  |  |
| --- | --- | --- | --- |
| 00003002 | 000e0300 | Ff040300 | 0f804000 |

#### 65----n (第49个字节到第n个字节)

trak03020000 04040303 0e050403 04040302 tick03020000 trak03020000 03030404 0e050403 04040302 tick03020000 trak03020000 04040303 0e050403 04040302 tick03020000 trak03020000 03030404 0e050403 04040302 jump03020000 03030404 0e050403 04040302 tick03020000 jump03020000 04040303 0e050403 04040302 tick03020000 jump03020000 04040303 0e050403 04040302 tick03020000 jump03020000 04040303 0e050403 04040302 tick03020000

(注：真实数据中没有空格。其中trak,tick,jump为了直观表示在真实数据中是4字节数据，它转换成的英文字母)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 7472616B | 03020000 | 04040303 | 0e050403 | 04040302 | 7469636B | 03020000 |
| trak | x坐标 | y坐标 | 步长 | 横向步数 | tick | 相对时间单位 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 6A756D70 | 0e050403 | 0e050403 | 0e050403 | 7469636B | 03020000 |
| jump | 滞空时间 | 跳跃高度 | 翻转角度 | Tick | 相对时间单位 |

后面数据为坐标数据和纵跳数据

每组坐标数据为 trak(4个字节)+x坐标(4个字节,float,单位为米)+y坐标(4个字节,float,单位为米)+步长(4个字节,float,单位为米)+tick(4个字节,long) // tick为相对时间单位 tick\*1000/512=毫秒

每组纵跳数据为 jump(4个字节)+滞空时间(4个字节,float,秒)+跳跃高度(4个字节,float,米)+翻转角度(4个字节,float,度)+tick(4个字节,long) // tick为相对时间单位 tick\*1000/512=毫秒