中控BLE-手机通讯协议

|  |  |  |  |
| --- | --- | --- | --- |
| 版本号 | 日期 | 修改记录 | 作者 |
| V0.1 | 2020-09-04 | 创建 | Allen |
| V0.2 | 2020-10-21 | 增加第5章：蓝牙名称定义 | Allen |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# 概述

本协议为APP与中控系统之间BLE通讯的规范。

# 协议定义

我们将命令分为两种：

Request：APP主动发起， 比如查询车辆状态、解锁、激活车辆等。

Notify： 中控主动发起， 心跳数据、故障上报等。

两种命令都可以有应答response,

**UUID列表：**

Service UUID:

0x14839AC4-7D7E-415C-9A42-167340CF2339

0x14839AC4-7D7E-415C-9A42-167340CF2339

Command Characteristic UUID:

0x8B00ACE7-EB0B-49B0-BBE9-9AEE0A26E1A3

Notify Characteristic UUID:

0X0734594A-A8E7-4B1A-A6B1-CD5243059A57

## 帧格式如下

APP与中控数据通讯的是以帧单位的。每帧的最长有效长度有20byte. 这是BLE协议所定义的。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **LSB** |  | | | **MSB** |
| Head | | | Payload | |
| **CMD/RSP** | | **Data Length** | **Data** | |
| BYTE | | BYTE | BYTE[N]， N <= 20 | |

## 命令码定义

APP发起（Request）

|  |  |  |
| --- | --- | --- |
| Frame ID | 命令说明 | 备注 |
| 0x01 | 鉴权 | APP发送16字节MD5串给中控，中控直接和本地存储的字符串比较来判断鉴权是否通过 |
| 0x04 | 获取电池信息 |  |
| 0x07 | 获取体检结果 |  |
| 0x08 | 获取GPS/GPRS信息 |  |
| 0x20 | AppRom Update Start | 升级开始 |
| 0x21 | AppRom Update | 发送数据 |
| 0x22 | AppRom Update Done |  |
| 0x23 | MCU Reset |  |

## 错误码定义

表格 1错误码定义

|  |  |  |
| --- | --- | --- |
| ERROR CODE | 名称 | 错误码说明 |
| 0x00 | SUCCESS | 执行成功 |
| 0x01 | PARAM\_INVALID | 参数错误 |
| 0x02 | UNSUPPORTED | 不支持的命令 |
| 0x03 | CRC\_ERROR | 校验码错误 |
| 0x04 | DEVICE NOT READY | 设备没准备好 |
| 0x05 | USERID\_ERROR | USER ID长度错误 |
| 0x0B | ERR\_USERID\_INVALID | USER ID无效 |
| 0x0C | ERR\_BAT\_NOT\_IN\_PRESENT | 电池不在位 |
| 0x0D | ERR\_RECORD\_INVALID | 记录号无效 |
| 0x0E | ERR\_CMD\_NOT\_ALLOW | 命令不允许执行，没有权限 |
| 0xFF |  |  |

# APP请求指令格式

## Authentication (0x01)

身份认证请求，某些蓝牙命令（例如开锁，点火等）需要身份认证通过才能执行。

身份认证请求的策略如下：

* 服务器把字符串“<MAC>immotor<Role>”使用MD5加密算法生产成为一个密文，作为认证请求命令参数发送到终端，其中

<MAC>：是6个字节的蓝牙地址。

<Role>：是角色定义，取值范围[0-1]，1-Admin。

* 终端使用相同的加密算法进行解密，如果解密成功，即可认为身份认证通过。

表格 2身份认证REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x04 | 命令码 |
| 1 | Data Length | UINT8 | 0x03 | 数据包长度，包含本身和Cmd字节 |
| 2-17 | MD5 | UINT8[16] |  | 通过和本地密钥比较来决定鉴权通过与否 |

表格 3身份认证RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x04 | 命令码 |
| 1 | Data Length | UINT8 | 0x03 | 数据包长度，包含本身和Cmd字节 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |

## GetPortState (0x03) \*

获取电池槽位的电池在位信息。

表格 4获取电池槽位状态REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Command | UINT8 | 0x03 | 命令码 |
| 1 | Data Length | UINT8 |  | Data长度 |

表格 5获取电池槽位状态RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Command | UINT8 | 0x03 | 命令码 |
| 1 | Data Length | UINT8 | VAR | Data长度 |
| 2 | Result | UINT8 | 0x00-0xFF | 参考上面“错误码定义” |
| 3 | Port State | UINT8 |  | 电池槽位状态，每一个BIT表示一个槽位状态。  BIT[0]：槽位0状态，0-不在位，1-在位。  BIT[1]：槽位1状态，0-不在位，1-在位。  BIT[2-7]：保留，置零。 |
| 4 | Port Desc Count | UINT8 |  | 端口号描述符个数 |
| 5-N | Port Descriptor[0]-  Port Descriptor[n] |  |  | 端口号描述符，参考Port Descriptor定义 |

Port Descriptor定义如下：

|  |  |  |  |
| --- | --- | --- | --- |
| Index | Name | Type | Descriptor |
| 0 | Port Number | UINT8 | 槽位号，0表示槽位1，后面递增 |
| 1-2 | Nominal Voltage | UINT16 | 额定电压，单位：10mV。  0xFFFF：无效值。 |
| 3-4 | Nominal Current | INT16 | 额定电流，单位：10mA。  0xFFFF：无效值。 |
| 5-6 | Capacity | UINT16 | 设计容量，单位：100mA |

## Get Battery Info(0x04)

获取电池信息命令。

表格 6获取电池信息REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x04 | 命令码 |
| 1 | Data Length | UINT8 | 0x03 | 数据包长度，包含本身和Cmd字节 |
| 2 | PortID | UINT8 |  | 电池槽位号ID。  0：槽位0电池信息。  1：槽位1电池信息。  其他值：保留 |

注意：如果请求的电池不在位，则返回Result=0x0C，Result以下的数据无效，只有当Result=0x00时，Result以下的数据才有效。

表格 7获取电池信息RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Command | UINT8 | 0x04 | 命令码 |
| 1 | Data Length | UINT8 |  | Data长度 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义，如果请求的电池不在位，返回0x0C. |
| 3 | Battery Descriptor | UINT8 |  | 参考Battery Descriptor定义。 |

表格 8 Battery Descriptor数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Val | Descriptor |
| 0-5 | Battery ID | UINT8[6] |  | 电池唯一表示号。 |
| 6 | Port Number | UINT8 |  | 电池槽位号，最小值为0。 |
| 7 | SOC | UINT8 |  | 电池剩余电量，单位为：%。  0-100：有效值。  其他值：无效。 |
| 8-9 | Voltage | UINT16 |  | 电池电压，单位：10mV。  0xFFFF：无效值。 |
| 10-11 | Current | INT16 |  | 电池电流，单位：10mA。  < 0：放电。  > 0：充电。  (-30 A) ~ (+30 A)：正常范围。  0xFFFF：无效值。 |
| 12 | Temperature | INT8 |  | 电池问题，单位：摄氏度。  (-40) ~ (+120)：有效范围。 |
| 13 | Fault | UINT8 |  | 电池故障，每一BIT代表一个错误类型。  BIT[0]: OVP，过压。  BIT[1]: UVP，欠压。  BIT[2]: OCP，过流。  BIT[3]: OTP，过温。  BIT[4]: UTP，低位。  BIT[5]: Other，其他故障。  BIT[6-7]: 保留，置0。 |
| 14 | Damage | UINT8 |  | 电池损坏，每一BIT代表一个损坏类型。  BIT[0]: 撞击。  BIT[1]: 拆开。  BIT[2-7]: 保留，置0。 |
| 15-16 | cycleCount | UINT16 |  | 循环次数 |

## Get Battery Info(0x05) -

Request 定义：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x05 | 命令码 |
| 1 | Data Length | UINT8 | 0x03 | 数据包长度，包含本身和Cmd字节 |
| 2 | PortID | UINT8 |  | 电池槽位号ID。  0：槽位0电池信息。  1：槽位1电池信息。  其他值：保留 |

注意：如果请求的电池不在位，则返回Result=0x0C，Result以下的数据无效，只有当Result=0x00时，Result以下的数据才有效。

Response定义：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Command | UINT8 | 0x05 | 命令码 |
| 1 | Data Length | UINT8 |  | Data长度 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义，如果请求的电池不在位，返回0x0C. |
| 3 | Battery Descriptor | UINT8 |  | 参考Battery Descriptor定义。 |

Battery Descriptor定义。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Val | Descriptor |
| 0-1 | Max Cell Voltage | UINT16 |  | 最大电芯电压，单位mV。 |
| 2-3 | Min Cell Voltage | UINT16 |  | 最小电芯电压，单位mV。 |
| 4 | Max Volt Cell Num | UINT8 |  | 最大电压电芯序号 |
| 5 | Min Volt Cell Num | UINT8 |  | 最小电压电芯序号 |
| 6 | Bms Pcb Temp | INT8 |  | BMS pcb温度，单位：摄氏度。  有效值：-40°~120。  其他值:无效值。 |
| 7 | Connector Temp | INT8 |  | BMS pcb温度，单位：摄氏度。  有效值：-40°~120。  其他值:无效值。 |
| 8 | Mos State | UINT8 |  | 充电管和放电管的开关状态。  BIT[0]：充电管的开关状态， 0-关；1-开。  BIT[1]：放电管的开关状态， 0-关；1-开。  BIT[2-7]：保留。 |

## Get SelfTest Result(0x07)

获取设备自检结果命令。

表格 9获取设备自检结果REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x07 | 命令码 |
| 1 | Data Length | UINT8 | 0x02 | 数据包长度，包含本身和Cmd字节 |

表格 10获取设备自检结果RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Command | UINT8 | 0x07 | 命令码 |
| 1 | Data Length | UINT8 |  | Data长度 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |
| 3 | SIM State | UINT8 |  | SIM卡状态。  BIT[0]：通信状态，0-失败；1-成功。  BIT[1]：卡在位状态，0-不在；1-在位。  BIT[2-7]：保留。 |
| 4 | GPRS State | UINT8 |  | GPRS状态。  BIT[0]：GPRS信号状态，0-没信号；1-有信号。  BIT[1]：服务器连接状态，0-失败；1-成功。  BIT[2-7]：保留。 |
| 5 | GPS State | UINT8 |  | GPS状态。  BIT[0]：定位状态，0-失败；1-成功。  BIT[1]：PmsIsCommOk，Pms是否通信正常。  BIT[2]： Pms is ready，Pms是否可以升级，0-不能升级（正在升级）。1-可以升级（没在升级）。  BIT[3-7]：保留。 |
| 6 | Device State | UINT8 |  | 设备状态定义：  BIT[0]: 点火状态，0-熄火；1-点火。  BIT[1]：是否可以充电。  BIT[2]：是否测试电池。  BIT[3]：轮毂锁状态，0-开锁；1-关锁  BIT[4]：座舱锁状态，0-开锁；1-关锁。  BIT[5-7]:保留。 |
| 7-8 | 18650 Voltage | UINT8 |  | 18650电池电压，低位在前。单位，0.1V |
| 9 | Device State2 | UINT8 |  | 设备状态2。  BIT[0]：激活状态。0-未激活，1-已激活。  BIT[1]：打卡使能。0-否，1-是。  BIT[2]：禁止放电。0-否，1-是。  BIT[3]：打卡成功。0-否，1-是。  BIT[4]：警戒模式。0-否，1-是。  BIT[5]：是否断电。0-否，1-是  BIT[6-7]：保留 |
| 10 | Battery Verify | UINT8 |  | 电池认证状态。  BIT[0]：电池认证使能。0-未开启，1-开启。  BIT[0-1]：Port0 电池认证。0-未知，1-认证成功，2-认证失败。  BIT[2-4]： Port1 电池认证。0-未知，1-认证成功，2-认证失败。  BIT[5-7]：保留 |

## Get GPS/GPRS Info (0x08)

获取GPS/GPRS位置和信号强度命令。

表格 11获取GPS/GPRS位置信息REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x08 | 命令码 |
| 1 | Data Length | UINT8 | 0x02 | 数据包长度，包含本身和Cmd字节 |

表格 12获取GPS/GPRS位置信息RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x08 | 命令码 |
| 1 | Data Length | UINT8 | 20 | 数据包长度，包含本身和Cmd字节 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |
| 3 | CSQ | UINT8 |  | GPRS信号强度, 0到31之间(99表示无信号），数值越大表明信号质量越好 |
| 4 | Satellites In View | UINT8 |  | GPS可见卫星数(0 – 16) |
| 5 | Max SNR | UINT8 |  | GPS信号强度，信噪比（00－99）dbHz，典型值在0～50之间，SNR虽可达到99，但极罕见，50已是非常好的情况 |
| 6-9 | longitude | Int32 |  | 经度，发送方乘(1E7)发送，接收方必须除(1E7)  0：表示没有获取到定位。 |
| 10-13 | latitude | Int32 |  | 纬度，发送方乘(1E7)发送，接收方必须除(1E7)  0：表示没有获取到定位。 |
| 14-17 | speed | Int32 |  | 速度，发东方乘10发送，接收方必须除10，转化为(Km/h)。 |

## Get Device Capacity(0x18) +

获取设备能力命令。

表格 13获取设备能力REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x18 | 命令码 |
| 1 | Data Length | UINT8 | 0x2 | 数据包长度，包含本身和Cmd字节 |
| 3 | Version | UINT8 |  | 协议版本号，每次修改时递增(+1)。  有效值：1-0xFF。 |

表格 14获取设备能力RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x18 | 命令码 |
| 1 | Data Length | UINT8 | 0xD | 数据包长度，包含本身和Cmd字节 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |
| 3 | Version | UINT8 |  | 协议版本号，每次修改时递增(+1)。  有效值：1-0xFF。 |
| 4 | Capacity | UINT32 |  | 设备能力描述，按位解析  BIT[0]: 是否有Smart板；0-没有，1-有。  BIT[1]: 是否有Pms板；0-没有，1-有。  BIT[2-3]: 支持最多插入电池数量。  BIT[4]：保留。  BIT[5]：是否支持陀螺仪；0-不支持，1-支持。  BIT[6]：是否支持喇叭；0-不支持，1-支持。  BIT[7]：是否支持轮毂锁；0-不支持，1-支持。  BIT[8]：是否支持座舱锁；0-不支持，1-支持。  BIT[9]：是否支持钥匙点火；0-不支持，1-支持。  BIT[10]：是否有18650电池；0-没有，1-有。  BIT[11-31]：保留。 |

## Get DeviceID (0x19) \*

获取Smart板的硬件和固件版本信息命令。

表格 15获取Smart板版本号REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x19 | 命令码 |
| 1 | Data Length | UINT8 | 0x2 | 数据包长度，包含本身和Cmd字节 |

表格 16获取Smart板版本号RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x19 | 命令码 |
| 1 | Data Length | UINT8 | 0xD | 数据包长度，包含本身和Cmd字节 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |
| 3 | Version | UINT8 |  | 协议版本号，每次修改时递增(+1)。  有效值：1-0xFF。 |
| 4 | HwMainVer | UINT8 |  | Smart板硬件主版本 |
| 5 | HwSubVer | UINT8 |  | Smart板硬件子版本 |
| 6 | AppMainVer | UINT8 |  | Smart板固件主版本 |
| 7 | AppSubVer | UINT8 |  | Smart板固件子版本 |
| 8 | AppMinorVer | UINT8 |  | Smart板固件修订版本 |
| 9-12 | AppBuildNum | UINT32 |  | Smart板固件Build号 |

## Get PMS Info (0x1A)

获取PMS设备信息

表格 17获取Pms版本号REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x1A | 命令码 |
| 1 | Data Length | UINT8 | 0x02 | 数据包长度，包含本身和Cmd字节 |

表格 18获取Pms版本号RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x1A | 命令码 |
| 1 | Data Length | UINT8 | 13 | 数据包长度，包含本身和Cmd字节 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |
| 3 | Version | UINT8 |  | PMS协议版本号，每次修改时递增(+1)。  有效值：1-0xFF。 |
| 4 | HwMainVer | UINT8 |  | PMS板硬件主版本 |
| 5 | HwSubVer | UINT8 |  | PMS板硬件子版本 |
| 6 | AppMainVer | UINT8 |  | PMS板固件主版本 |
| 7 | AppSubVer | UINT8 |  | PMS板固件子版本 |
| 8 | AppMinorVer | UINT8 |  | PMS板固件修订版本 |
| 9-12 | AppBuildNum | UINT32 |  | PMS板固件Build号 |
| 13 | State | UINT8 |  | 设备状态：  BIT[0]: IsCommOk，是否通信正常。  BIT[1]: Pms is ready，Pms是否可以升级，0-不能升级。1-可以升级。  BIT[2-7]:保留，置零。 |

## Get BMS Info (0x1B)

获取BMS设备信息命令

表格 19获取Bms版本号REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x1B | 命令码 |
| 1 | Data Length | UINT8 | 0x2 | 数据包长度，包含本身和Cmd字节 |

表格 20获取Bms版本号RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x1B | 命令码 |
| 1 | Data Length | UINT8 | 0x03 | 数据包长度，包含本身和Cmd字节 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |
| 3 | Version | UINT8 |  | BMS协议版本号，每次修改时递增(+1)。  有效值：1-0xFF，排除257。 |
| 4 | HwMainVer | UINT8 |  | BMS板硬件主版本 |
| 5 | HwSubVer | UINT8 |  | BMS板硬件子版本 |
| 6 | AppMainVer | UINT8 |  | BMS板固件主版本 |
| 7 | AppSubVer | UINT8 |  | BMS板固件子版本 |
| 8 | AppMinorVer | UINT8 |  | BMS板固件修订版本 |
| 9-12 | AppBuildNum | UINT32 |  | BMS板固件Build号 |

## Active Device (0x1C)

激活车子命令，所有车子出厂时都是去激活状态，用户第一次使用车子之前必须要激活才能正常使用。

表格 21激活设备REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x1C | 命令码 |
| 1 | Data Length | UINT8 | 0x03 | 数据包长度，包含本身和Cmd字节 |
| 2 | State | UINT8 |  | 状态：  BIT[0]：设备激活；0-否；1-是。  BIT[1]：打卡使能；0-否；1-是。  BIT[2]：电池认证使能；0-否；1-是。  BIT[3-7]：保留，置零。 |

表格 22激活设备RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x1C | 命令码 |
| 1 | Data Length | UINT8 | 0x03 | 数据包长度，包含本身和Cmd字节 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |

## SignIn Req (0x1D) -

打卡功能，当车子的中控网络功能出现故障或者在无网络状态下导致“打卡”失败时，用户可以通过手机蓝牙功能发送该命令，进行打卡。

Request定义：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x1D | 命令码 |
| 1 | Data Length | UINT8 | 0x03 | 数据包长度，包含本身和Cmd字节 |
| 2 | SignIn | UINT8 |  | 是否打卡；0: 否；1：是。 |
| 3 | Device State | UINT8 |  | 可选字节，设置设备如下状态。  BIT[0]：激活状态。0-未激活，1-已激活。  BIT[1]：打卡使能。0-否，1-是。  BIT[2]：禁止放电。0-否，1-是。  BIT[3]：打卡定时器不复位。0-否，1-是。仅用于测试  BIT[4-7]：保留，置零 |
| 4-7 | loginMaxMinute | UINT32 |  | 可选字节，设置最大打卡时间，默认值为180分钟(3小时)，单位为分钟。 |

Response定义：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x1D | 命令码 |
| 1 | Data Length | UINT8 | 0x11 | 数据包长度，包含本身和Cmd字节 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |
| 3 | Device State | UINT8 |  | 设备状态2。  BIT[0]：激活状态。0-未激活，1-已激活。  BIT[1]：打卡使能。0-否，1-是。  BIT[2]：禁止放电。0-否，1-是。  BIT[3]：打卡定时器不复位。0-否，1-是。  BIT[4]：打卡成功。0-否，1-是。  BIT[5-7]：保留 |
| 4-7 | loginMaxMinute | UINT32 |  | 最大打卡时间，分钟，默认值为3小时 |
| 7-10 | loginAfterMinute | UINT32 |  | 打卡之后时间，分钟 |
| 11-14 | remainMinute | UINT32 |  | 打卡剩余时间，分钟 |

## Battery Verify Req (0x1E)

电池身份验证请求。

表格 23电池身份验证REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x1E | 命令码 |
| 1 | Data Length | UINT8 | 0x04 | 数据包长度，包含本身和Cmd字节 |
| 2 | Port0 Battery Verify | UINT8 |  | 设置Port0电池认证状态；   1. 未知。 2. 认证成功。 3. 认证失败。   其他值：保留 |
| 2 | Port1 Battery Verify | UINT8 |  | 设置Port1电池认证状态；   1. 未知。 2. 认证成功。 3. 认证失败。   其他值：保留 |

表格 24电池身份验证RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x1E | 命令码 |
| 1 | Data Length | UINT8 | 0x11 | 数据包长度，包含本身和Cmd字节 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |

## Set Nvds (0x29) \*

设置Nvds的值。

表格 25设置设备NVDS数据REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x29 | 命令码 |
| 1 | Data Length | UINT8 | 0x03 | 数据包长度，包含本身和Cmd字节 |
| 2- | NvdsParam | NvdsParam Struct |  | NvdsParam列表,参考表格 26 |

表格 26 NvdsParam Struct数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Tag | UINT8 |  | 参数ID |
| 1 | Len | UINT8 |  | 参数长度 |
| 2 | Value | UINT8 |  | 参数值 |

参数定义如表格 27所示：

表格 27 NVDS参数定义

|  |  |  |
| --- | --- | --- |
| Tag | Len | Descriptor |
| 0x01 | 7 | Smart板固件版本号。  BYTE[0]: FwMainVer。  BYTE[1]: FwSubVer。  BYTE[2]: FwMinorVer。  BYTE[3-6]: FwBuildNum，低位字节在前。 |
| 0x02 | 7 | Pms板固件版本号。  BYTE[0]: FwMainVer。  BYTE[1]: FwSubVer。  BYTE[2]: FwMinorVer。  BYTE[3-6]: FwBuildNum，低位字节在前。 |
| 0x04 | 2 | Smart板硬件版本号。  BYTE[0]: HwMainVer。  BYTE[1]: HwSubVer。 |
| 0x10 | 1 | 服务器地址  0: 测试服务器地址  1: 正式库服务器地址  2：预发布地址  3：开发服务器地址 |
|  |  |  |

表格 28设置设备NVDS数据RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x29 | 命令码 |
| 1 | Data Length | UINT8 | 0x03 | 数据包长度，包含本身和Cmd字节 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |

## Get Nvds (0x2A)

读取设备的Nvds的参数值命令。

表格 29获取设备NVDS数据REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x2A | 命令码 |
| 1 | Data Length | UINT8 | 0x03 | 数据包长度，包含本身和Cmd字节 |
| 2- | Nvd Ind | UINT8 |  | 定义请参考表格 27 |

表格 30获取设备NVDS数据RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x29 | 命令码 |
| 1 | Data Length | UINT8 | 0x03 | 数据包长度，包含本身和Cmd字节 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |
| 3- | NvdsDesc | NvdsDesc |  | 参考上面定义。 |

## Set Factory Setting (0x2B)

恢复出厂设置。

表格 31恢复出厂设置REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x2B | 命令码 |
| 1 | Data Length | UINT8 | 0x03 | 数据包长度，包含本身和Cmd字节 |
| 2 | State | UINT8 |  | 恢复出厂设置，如果不包含该字节，表示切断18650电源。 1：恢复出厂设置。  其他值：保留 |

表格 32恢复出厂设置RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x2B | 命令码 |
| 1 | Data Length | UINT8 | 0x03 | 数据包长度，包含本身和Cmd字节 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |

## Set Alarm Mode (0x2C)

设置/解除警戒模式。

表格 33设置警戒模式REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x2C | 命令码 |
| 1 | Data Length | UINT8 | 0x03 | 数据包长度，包含本身和Cmd字节 |
| 2 | Alarm Mode | UINT8 |  | 设置警戒模式 BIT[0]：Alarm Mode, 0-DISABLE; 1-ENABLE。  BIT[1]：断电, 0-DISABLE; 1-ENABLE。  BIT[2]：断电模式, 0-停车断电; 1-立即断电。  BIT[3-7]：保留。 |

表格 34设置警戒模式RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x2C | 命令码 |
| 1 | Data Length | UINT8 | 0x03 | 数据包长度，包含本身和Cmd字节 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |

## Set LowSocPlay(0x2D) (喇叭)

设置低电量播报阈值。

表格 35设置低电量播报阈值REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x2D | 命令码 |
| 1 | Data Length | UINT8 | 0x05 | 数据包长度，包含本身和Cmd字节 |
| 2 | LP Soc 1 | UINT8 |  | 第一级播报低电量阈值。默认为10  0：不播报。  取值范围：1-30之间的任何值。 |
| 3 | LP Soc 2 | UINT8 |  | 第二级低电量阈值数组。默认值为20。  0：不播报。  取值范围：1-30之间的任何值。 |
| 4 | LP Soc 3 | UINT8 |  | 第三次低电量阈值数组。默认值为0。  0：不播报。  取值范围：1-30之间的任何值。 |

表格 36设置低电量播报阈值RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x2D | 命令码 |
| 1 | Data Length | UINT8 | 0x03 | 数据包长度，包含本身和Cmd字节 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |

## Set SocPlay(0x2E) (喇叭)

设置电量播报阈值。

表格 37设置低电量播报使能REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x2E | 命令码 |
| 1 | Data Length | UINT8 | 0x04 | 数据包长度，包含本身和Cmd字节 |
| 2 | Soc1 | UINT8 |  | 10%-80%电量值播报使能。  BIT[0]: 10%，0-不播报，1-播报；默认值：0。  BIT[1]: 20%，0-不播报，1-播报；默认值：0。  **BIT[2]: 30%，0-不播报，1-播报；默认值：1。**  BIT[3]: 40%，0-不播报，1-播报；默认值：0。  **BIT[4]: 50%，0-不播报，1-播报；默认值：1。**  BIT[5]: 60%，0-不播报，1-播报；默认值：0。  BIT[6]: 70%，0-不播报，1-播报；默认值：0。  BIT[7]: 80%，0-不播报，1-播报；默认值：0。 |
| 3 | Soc2 | UINT8 |  | 90%-100%电量值播报使能。  BIT[0]: 90%，0-不播报，1-播报；默认值：0。  BIT[1]: 100%，0-不播报，1-播报；默认值：0。  BIT[2-7]:保留. |

表格 38设置低电量播报使能RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x2E | 命令码 |
| 1 | Data Length | UINT8 | 0x03 | 数据包长度，包含本身和Cmd字节 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |

## Get SocPlay(0x2F) (喇叭)

获取电量播报阈值。

表格 39获取低电量播报使能REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x2F | 命令码 |
| 1 | Data Length | UINT8 | 0x02 | 数据包长度，包含本身和Cmd字节 |

表格 40获取低电量播报使能RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x2F | 命令码 |
| 1 | Data Length | UINT8 | 0x08 | 数据包长度，包含本身和Cmd字节 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |
| 3 | Soc1 | UINT8 |  | 10%-80%电量值播报使能。  BIT[0]: 10%，0-不播报，1-播报；默认值：0。  BIT[1]: 20%，0-不播报，1-播报；默认值：0。  **BIT[2]: 30%，0-不播报，1-播报；默认值：1。**  BIT[3]: 40%，0-不播报，1-播报；默认值：0。  **BIT[4]: 50%，0-不播报，1-播报；默认值：1。**  BIT[5]: 60%，0-不播报，1-播报；默认值：0。  BIT[6]: 70%，0-不播报，1-播报；默认值：0。  BIT[7]: 80%，0-不播报，1-播报；默认值：0。 |
| 4 | Soc2 | UINT8 |  | 90%-100%电量值播报使能。  BIT[0]: 90%，0-不播报，1-播报；默认值：0。  BIT[1]: 100%，0-不播报，1-播报；默认值：0。  BIT[2-7]:保留. |
| 5 | LP Soc 1 | UINT8 |  | 第一级播报低电量阈值。默认为10。  0：不播报。  取值范围：1-30之间的任何值。 |
| 6 | LP Soc 2 | UINT8 |  | 第二级低电量阈值数组。默认值为20。  0：不播报。  取值范围：1-30之间的任何值。 |
| 7 | LP Soc 3 | UINT8 |  | 第三次低电量阈值数组。默认值为0。  0：不播报。  取值范围：1-30之间的任何值。 |

## Read Log Info(0x30)

获取Log信息，包括记录总数，记录的起始时间。

表格 41读取日志信息REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x30 | 命令码 |
| 1 | Data Length | UINT8 | 0x02 | 数据包长度，包含本身和Cmd字节 |
|  |  |  |  |  |

表格 42读取日志信息RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x30 | 命令码 |
| 1 | Data Length | UINT8 | 11 | 数据包长度，包含本身和Cmd字节 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |
| 3-6 | Records Count | UINT32 |  | Log的记录总数 |
| 7-10 | DateTime | UINT32 |  | Log的记录起始时间。 |

## Log Seek By Index (0x31)

根据Log记录号设置读指针位置，如果该记录号小于总记录数，则返回成功，否则返回失败。

设置读指针位置成功之后，如果设备接收到命令“Read Log Records”，则返回当前位置的记录内容。

表格 43设置日志读取位置REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x31 | 命令码 |
| 1 | Data Length | UINT8 | 0x02 | 数据包长度，包含本身和Cmd字节 |
| 2-5 | Index | UINT32 |  | Log记录号 |

表格 44设置日志读取位置RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x31 | 命令码 |
| 1 | Data Length | UINT8 | 0x11 | 数据包长度，包含本身和Cmd字节 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |

## Log Seek By Data Time (0x32)

根据时间日期条件设置读指针位置，设备会检索所有的Log记录，比较Log记录的时间，如果找到大于或者等于指定日志的位置，返回成功，如果没有找到该位置，返回失败。

设置读指针位置成功之后，如果设备接收到命令“Read Log Records”，则返回当前位置的记录。

表格 45设置日志读取时间REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x32 | 命令码 |
| 1 | Data Length | UINT8 | 11 | 数据包长度，包含本身和Cmd字节 |
| 2-5 | Year | int |  | 年 |
| 6 | Month | UINT8 |  | 月 |
| 7 | Day | UINT8 |  | 日 |
| 8 | Hour | UINT8 |  | 时 |
| 9 | Minute | UINT8 |  | 分 |
| 10 | Second | UINT8 |  | 秒 |

表格 46设置日志读取时间RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x32 | 命令码 |
| 1 | Data Length | UINT8 | 1 | 数据包长度，包含本身和Cmd字节 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |

## Read Log Records (0x33)

读取当前Log记录内容，该命令的功能如下：

* Log记录指针可由命令“Log Seek By Index”指定。
* 读完之后Log记录指针增1。
* 如果返回空记录（3个字节），表示所有记录已经读完，没有下一条Log记录。

表格 47读取日志REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x33 | 命令码 |
| 1 | Data Length | UINT8 | 0x03 | 数据包长度，包含本身和Cmd字节 |
| 2 | Num of Record | UINT8 |  | 记录数，最多2条 |

表格 48读取日志RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x33 | 命令码 |
| 1 | Data Length | UINT8 |  | 数据包长度，包含本身和Cmd字节 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义  如果已经读完所有记录，返回0x01 |
| 3-10 | Record Desc[] | Record Desc |  | 记录内容，参考下表。 |

表格 49日志数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Head | UINT8 | 0x21 | Log Head。  BIT[0-2]:Version;  BIT[3-7]:Reserved; |
| 1-4 | dateTime | UINT32 |  | 日期时间，unix时间戳格式，从1970年1月1日（UTC/GMT的午夜）开始所经过的秒数，不考虑闰秒。 |
| 5 | eventID | UINT8 |  | 时间ID，参考下表。 |
| 6 | Param1 | UINT8 |  | 参数1 |
| 7 | Param2 | UINT8 |  | 参数2 |

## Delete Log(0x34)

删除所有日志记录。

表格 50删除日志REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x34 | 命令码 |
| 1 | Data Length | UINT8 | 0x02 | 数据包长度，包含本身和Cmd字节 |

表格 51删除日志RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x34 | 命令码 |
| 1 | Data Length | UINT8 | 0x03 | 数据包长度，包含本身和Cmd字节 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |

## SetStopCondition(0x35)

设置停车条件， PMS 板监控电池“电流的放电大小”和“持续时间”来判定车子是否处于停车状态，主要包括如下参数。

当 PMS 板没有接收到该命令时， 使用默认值。

1：小电流阈值。

2：小电流放电持续时间。

表格 52设置停车条件REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x35 | 命令码 |
| 1 | Data Length | UINT8 | 0x05 | 数据包长度，包含本身和Cmd字节 |
| 2 | Time | UINT16 |  | 停车判定条件-小电流持续时间，单位（S）  默认值： 30S |
| 4 | Current | UINT16 |  | 停车判定条件-小电流阈值，单位（mA）  默认值： 2500 |

表格 53设置停车条件REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x35 | 命令码 |
| 1 | Data Length | UINT8 | 0x03 | 数据包长度，包含本身和Cmd字节 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |

## GetRunParam(0x36)

获取骑行参数，包括锁状态，行驶速度和电池剩余电量。

表格 54获取骑行参数REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x36 | 命令码 |
| 1 | Data Length | UINT8 | 0x02 | 数据包长度，包含本身和Cmd字节 |

表格 55获取骑行参数RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x36 | 命令码 |
| 1 | Data Length | UINT8 | 0x07 | 数据包长度，包含本身和Cmd字节 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |
| 3 | Lock State | UINT8 |  | 锁状态。  BIT[0]：轮毂锁状态，0-开锁；1-关锁。  BIT[1]：座舱锁状态，0-开锁；1-关锁。  BIT[2]：座舱锁故障，执行开锁命令后检测到没开锁。  BIT[3-7]：保留 |
| 4 | Speed | UINT16 |  | 速度或者轮毂转速。  BIT[0]：0-速度(0.01Km/小时)；1-轮毂转速(r/min)。  BIT[1-15]：速度或者转速值，取决于BIT[0] |
| 6 | Soc | UINT8 |  | 电池剩余电量 |

## SetWheelLockState(0x37) （轮毂锁）

设置轮毂锁锁状态。

表格 56设置轮毂锁状态REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x37 | 命令码 |
| 1 | Data Length | UINT8 | 0x03 | 数据包长度，包含本身和Cmd字节 |
| 2 | Lock State | UINT8 |  | 锁状态；  0：开锁；  1：关锁。 |

表格 57设置轮毂锁状态RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x37 | 命令码 |
| 1 | Data Length | UINT8 | 0x03 | 数据包长度，包含本身和Cmd字节 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |

## SetCabinLockState Cmd (0x38) （座舱锁）

设置座舱锁状态，开锁或者关锁。

表格 58设置座舱锁状态REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x38 | 命令码 |
| 1 | Data Length | UINT8 | 0x03 | 数据长度 |
| 2 | Lock Flag | UINT8 |  | 锁状态标志。  0：开锁。  1：关锁。 |

表格 59设置座舱锁状态RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x38 | 命令码 |
| 1 | Data Length | UINT8 | 1 | 数据包长度 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |

## SetAccState Cmd (0x39)

设置ACC ON/OFF状态，远程点火或者熄火。

表格 60远程点火REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x39 | 命令码 |
| 1 | Data Length | UINT8 | 0x03 | 数据长度 |
| 2 | Lock Flag | UINT8 |  | ACC状态标志。  0：熄火。  1：点火。 |

表格 61远程点火RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x39 | 命令码 |
| 1 | Data Length | UINT8 | 1 | 数据包长度 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |

## Horn Test Cmd (0x3A) (喇叭)

喇叭测试命令。

表格 62喇叭测试REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x3A | 命令码 |
| 1 | Data Length | UINT8 | 0x05 | 数据长度 |
| 2 | Audio Index | UINT8 | - | 语音文件编号。  0x0F：滴，提示音。  0x12：叭，警告音。 |
| 3 | Count | UINT8 | - | 播放次数。 |
| 4 | Vol | UINT8 | - | 音量，1-8。数字越大，音量越大。 |
|  |  |  |  |  |

表格 63喇叭测试RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x3A | 命令码 |
| 1 | Data Length | UINT8 | 1 | 数据包长度 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |

## Get Asy Info Cmd (0x3B) (调速控制器)

获取调速控制器信息。

表格 64获取调速控制器版本信息REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x3B | 命令码 |
| 1 | Data Length | UINT8 | 0x02 | 数据长度 |

表格 65获取调速控制器版本信息RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x3B | 命令码 |
| 1 | Data Length | UINT8 | 1 | 数据包长度 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |
| **3-4** | **硬件版本号** | UINT16 |  | 控制器硬件版本号  第一个字节：控制器电压等级：48,60,72 等  第二个字节：控制器硬件版本； |
| **4-5** | **SN1** | UINT16 |  | 序列号字节1、2，字符串 |
| **6-7** | **SN2** | UINT16 |  | 序列号字节3、4，字符串 |
| **8-9** | **SN3** | UINT16 |  | 序列号字节5、6，字符串 |
| **10-11** | **SN4** | UINT16 |  | 序列号字节7、8，字符串 |
| **12-13** | **SN5** | UINT16 |  | 序列号字节9、10，字符串 |
| **14-15** | **SN6** | UINT16 |  | 序列号字节11、12，字符串 |
| **16-17** | **SN7** | UINT16 |  | 序列号字节13、14，字符串 |

## Get Asy State Cmd (0x3C) (调速控制器)

获取调速控制器状态。

表格 66获取调速控制器状态REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x3C | 命令码 |
| 1 | Data Length | UINT8 | 0x02 | 数据长度 |

表格 67获取调速控制器状态RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x3C | 命令码 |
| 1 | Data Length | UINT8 | - | 数据包长度 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |
| 2-3 | State1 | UINT16 |  | BIT0 控制器故障 1：故障状态 0：正常状态  BIT1 转把故障 1：故障状态 0：正常状态  BIT2 刹车故障 1：故障状态 0：正常状态  BIT3 电机霍尔故障 1：故障状态 0：正常状态  BIT4 保养 1：需要保养； 0：无需保养  BIT5 欠压状态 1：欠压状态 0：非欠压状态  BIT6 过压状态 1：过压状态 0：非过压状态  BIT7 防盗状态 1：防盗状态 0：非防盗状态  BIT8-BIT15 预留 |
| 4-5 | State2 | UINT16 |  | BIT0 实时状态 1：运行状态 0：静止状态  BIT1 巡航状态 1：巡航状态 0：非巡航状态  BIT2 电动状态 1：电动状态 0：助力状态  BIT3 能量回收 1：回收状态 0：非回收状态  BIT4 1 档状态 1：1 档状态 0：非 1 档状态  BIT5 2 档状态 1：2 档状态 0：非 2 档状态  BIT6 3 档状态 1：3 档状态 0：非 3 档状态  BIT7 4 档状态 1：4 档状态 0：非 4 档状态  BIT8-BIT15 预留 |
| 6-7 | State3 | UINT16 |  | BIT0 总状态标志 1：正常状态 0：故障状态  BIT1 修复状态 1：修复状态 0：非修复  BIT2 驻车 P 档 1：有效 0：无效  BIT3 电机锁标志 1：锁定中 0：解锁  BIT4 通信故障 1：故障 0：正常  BIT5 转把状态 1：启动 0：停止  BIT6 刹车状态 1：启动 0：停止  BIT7 READY（童锁） 1：READY 0：非 READY  BIT8-BIT15 预留 |
| 8-9 | State4 | UINT16 |  | BIT0 限速 1: 限速模式 0：非限速模式  BIT1 电子刹车 1：启动 0：未启动  BIT2 倒车 1：倒车状态 0：正常状态  BIT3 堵转保护 1：保护状态 0：正常状态  BIT4 过流保护 1：保护状态 0：正常状态  BIT5 备用电源 1：启用 0：未启用  BIT6 启用一键通 1：启用 0：未启用  BIT7 电流标记 1：电流超过 70% 0：未超过  BIT8-BIT15 预留 |
| 10-11 | 速度 | UINT16 |  | 速度，分辨率1Km/小时 |
| 12-13 | 控制指令 | UINT16 |  | BIT 0：锁电机指令  1：锁电机；0：取消锁电机  BIT 1：限速指令  1：使能限速控制；0：取消限速控制  BIT 2：能量回收  1：使能能量回收；0：取消能量回收  BIT3-BIT15 保留 |
| 14-15 | 限速比例 | UINT16 |  | 报文要正常回复，功能无需实现 |
| 16-17 | 当前里程 | UINT16 |  | 控制器计算的当前里程；分辨率 1KM |

## Set Asy Cmd (0x3D) (调速控制器)

设置调速控制器命令。

表格 68设置调速控制器REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x3D | 命令码 |
| 1 | Data Length | UINT8 | 0x08 | 数据长度 |
| 2-3 | 控制指令 | UINT16 |  | BIT 0：锁电机指令  1：锁电机；0：取消锁电机  BIT 1：限速指令  1：使能限速控制；0：取消限速控制  BIT 2：能量回收  1：使能能量回收；0：取消能量回收  BIT3-BIT15 保留 |
| 4-5 | 限速比例 | UINT16 |  | 0-100%电压比例值，分辨率 1%。0xFFFF不设置 |
| 6-7 | 累计里程 | UINT16 |  | 累计里程：分辨率 1KM。0xFFFF不设置里程。 |

表格 69设置调速控制器RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x3A | 命令码 |
| 1 | Data Length | UINT8 | 1 | 数据包长度 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |

## Get Beacon Scan Cmd (0x3E) (信标)

获取信标信息命令。

表格 70获取信标信息REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x3E | 命令码 |
| 1 | Data Length | UINT8 | 0x02 | 数据长度 |

表格 71获取信标信息RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x3E | 命令码 |
| 1 | Data Length | UINT8 | 1 | 数据包长度 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |
| 3 | Beacon Count | UIN8 | 0-5 | 信标数量，最多5个。 |
| 4~ | Beacon Desc | BeaconDesc[5] |  | 信标描述符，最多5个，定义参考下表 |

表格 72 Beacon Description数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0-1 | Major | UINT16 | - | Major标识 |
| 2-3 | Minor | UINT16 | - | Minor标识 |
| 4 | RSSI | INT8 |  | RSSI值，0 ~ -127 |

蓝牙使用RSSI计算距离：

计算公式：

d = 10^((abs(RSSI) - A) / (10 \* n))

其中：

d - 计算所得距离

RSSI - 接收信号强度（负值）

A - 发射端和接收端相隔1米时的信号强度，默认值：59

n - 环境衰减因子，默认值：2.0

由于所处环境不同，每台发射源（蓝牙设备）对应参数值都不一样。按道理，公式里的每项参数都应该做实验（校准）获得。

当你不知道周围蓝牙设备准确位置时，只能给A和n赋经验值（如本例）。

我们应该再安装道钉设备时，同时在实际环境中计算出A和N。

## Get Beacon Cmd (0x3F) (信标)

获取信标信息命令。

表格 73获取信标配置信息REQ数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x3F | 命令码 |
| 1 | Data Length | UINT8 | 0x08 | 数据长度 |
| 2-3 | Major | UINT16 | - | Major标识 |
| 4-5 | Minor | UINT16 | - | Minor标识 |

表格 74获取信标配置信息RSP数据格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Cmd | UINT8 | 0x3E | 命令码 |
| 1 | Data Length | UINT8 | 1 | 数据包长度 |
| 2 | Result | UINT8 |  | 参见附录1，错误码的定义 |
| 3-8 | Mac | UINT8[6] |  | 信标MAC |
| 9 | SOC | UINT8 |  | 电池电量百分比, 0-无效值，没获取到。 |
| 10-11 | Voltage | UINT16 |  | 电池实际电压值，单位为mv,  0-无效值，没获取到 |
| 12 | rssiCalib | INT8 |  | 取值范围：-127 ~ 20 |
| 13-14 | interval | UINT16 |  | 单位ms, 范围：100-10000 |
| 15 | txPower | UINT8 |  | 发射功率,  取值范围：4,0,-4,-8,-12,-16,-20,-30 |

# APPROM Update Process

APPROM OTA升级软件APP运行在手机中，通过UART发送命令给DEVICE，他们的通信连接示意图如下图：



APPROM OTA 通信连接示意图

## 升级协议定义

APP与设备间数据通讯的是以帧单位的。每帧的数据最长长度为132 bytes (128 + 4)。

帧格式定义:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **LSB** |  | | | **MSB** |
| **Head** | Payload | | | Tail |
| 0x7E | **CMD/RSP** | **Data Length** | **Data** | 0xFF |
| 1 octet | 1 octet | 1 octet | Variable | 1 octet |

1. 所有数据域以小端格式表示，即低字节先发送，高字节后发送。
2. Header： 0x7E，表示一帧数据的开始，后面是被传数据。
3. Tail： 0xFF，表示一帧数据的结束。
4. CMD/RSP：命令或者响应。
5. Data Length：传输的Data长度。
6. Data：传输的数据。
7. 在发送方，如果Payload存在如下字节，必须要做转码处理，接收方接收到数据之后，必须做相反的转码处理。

|  |  |
| --- | --- |
| 字符 | 转码 |
| 0x7E | 0x8C 0x81 |
| 0xFF | 0x8C 0x00 |
| 0x8C | 0x8C 0x73 |

## 升级命令定义

### FwUpdate Start (0x20)

固件更新开始请求

Request定义：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Head | UINT8 | 0x7E | 包头 |
| 1 | Cmd | UINT8 | 0x20 | 命令码 |
| 2 | Length | UINT8 | 0x0D | 数据包长度，不包含本身和Cmd字节 |
| 3 | Target | UINT8 |  | 升级目标：  0：Smart板固件。  1：PMS板固件。  2：BMS板固件。 |
| 4 | BMS Port | UINT8 |  | BMS槽位号：  只有当Target 是BMS板时，该域才有意义；  当Target为其他值时，该域保持为0. |
| 5-8 | File Length | UINT32 |  | 文件长度 |
| 9 | MainVer | UINT8 |  | 固件主版本号 |
| 10 | SubVer | UINT8 |  | 固件子版本号 |
| 11 | MinorVer | UINT8 |  | 固件修订版本号 |
| 12-15 | BuildNum | UINT32 |  | 固件Build版本号 |
| 16 | Tail | UINT8 | 0xFF | 包尾 |

Response定义：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Head | UINT8 | 0x7E | 包头 |
| 1 | Cmd | UINT8 | 0x20 | 命令码 |
| 2 | Data Length | UINT8 | 0x03 | 数据包长度，不包含本身和Cmd字节 |
| 3 | Result | UINT8 |  | 参见附录1，错误码的定义 |
| 4 | Tail | UINT8 | 0xFF | 包尾 |

### FwUpdate (0x21)

固件数据更新请求。

Request定义：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Head | UINT8 | 0x7E | 包头 |
| 1 | Cmd | UINT8 | 0x21 | 命令码 |
| 2 | Length | UINT8 | 132 | 数据包长度，不包含本身和Cmd字节 |
| 3-6 | offset | UINT32 |  | 偏移 |
| 7-134 | File Data | UINT8[128] |  | 文件数据 |
| 135 | Tail | UINT8 | 0xFF | 包尾 |

Response定义：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Head | UINT8 | 0x7E | 包头 |
| 1 | Cmd | UINT8 | 0x21 | 命令码 |
| 2 | Data Length | UINT8 | 0x01 | 数据包长度，不包含本身和Cmd字节 |
| 3 | Result | UINT8 |  | 参见附录1，错误码的定义 |
| 4 | Tail | UINT8 | 0xFF | 包尾 |

### FwUpdate Done (0x22)

固件更新结束请求。

Request定义：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Head | UINT8 | 0x7E | 包头 |
| 1 | Cmd | UINT8 | 0x22 | 命令码 |
| 2 | Length | UINT8 | 0x04 | 数据包长度，不包含本身和Cmd字节 |
| 3-6 | CRC | UINT32 |  | 检验和 |
| 7 | Tail | UINT8 | 0xFF | 包尾 |

Response定义：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Head | UINT8 | 0x7E | 包头 |
| 1 | Cmd | UINT8 | 0x22 | 命令码 |
| 2 | Data Length | UINT8 | 0x01 | 数据包长度，不包含本身和Cmd字节 |
| 3 | Result | UINT8 |  | 参见附录1，错误码的定义 |
| 4 | Tail | UINT8 | 0xFF | 包尾 |

### Mcu Reset (0x23)

Request定义：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Head | UINT8 | 0x7E | 包头 |
| 1 | Cmd | UINT8 | 0x23 | 命令码 |
| 2 | Length | UINT8 | 0x00 | 数据包长度，不包含本身和Cmd字节 |
| 3 | Tail | UINT8 | 0xFF | 包尾 |

该命令无应答。

## APPROM OTA的升级定义

1. APPROM OTA文件的格式必须为标准的BIN文件格式。
2. APP和被升级目标DEVICE之间采用请求-响应的工作方式，APP向DEVICE发送”REQ”，DEVICE接收到”REQ”之后，必须回复一个“RSP”， APP如果在规定的时间内没有收到”RSP”, 则必须重发”REQ”，直到接收到”RSP”。

## APPROM OTA文件格式定义

APPROM的文件格式定义如下：文件的前段数据是纯的固件数据，文件将会被逻辑划分成若干个块，每个块的大小是64字节对齐，文件尾部不完整块用FFh填充，文件的最后一个块包含文件的CRC，文件有效长度信息，版本信息等。



说明：

1. BIN DATA CRC: 使用CRC校验算法对BIN DATA计算出一个CRC值, 不包含补位的数据。
2. BIN DATA LEN: BIN DATA的长度，不包含补位的数据。
3. Firwware version: 固件的版本号。
4. FILE INFO CRC: 使用CRC校验算法对文件最后的一个区块字节，从BIN DATA CRC位置开始，长度为128 - 4，计算出的CRC值，用于校验最后的区块是否是有效的内容。
5. CRC算法如下：

uin电池仓6\_t crc16\_compute(const uint8\_t \* p\_data, uint32\_t size, const uin电池仓6\_t \* p\_crc)

{

uint32\_t i;

uin电池仓6\_t crc = (p\_crc == NULL) ? 0xffff : \*p\_crc;

for (i = 0; i < size; i++)

{

crc = (unsigned char)(crc >> 8) | (crc << 8);

crc ^= p\_data[i];

crc ^= (unsigned char)(crc & 0xff) >> 4;

crc ^= (crc << 8) << 4;

crc ^= ((crc & 0xff) << 4) << 1;

}

return crc;

}

## APPROM OTA文件的升级过程

APP的升级过程如下序列图：



1. APP读取OTA文件，检验合法性，从最后一个扇区中取出文件的版本号和有效文件长度。
2. APP发送命令” UpdateApp Start REQ”给DEVICE
   1. 如果在2S内没有接收到RSP，则必须重发当前请求，重发总时长最大10秒。
   2. 如果接收到RSP，错误码为“NOT READY”，则必须延时2秒重发当前请求，重发总时长最大30秒。
   3. 如果接收到RSP，错误码为“SUCCESS”，则执行下一步。
3. APP发送命令“FwUpdate”，每次发送固件数据的长度固定为128字节。每次发送请求之后等待2秒接收响应。
   1. 如果接收到RSP[SUCCESS]，则跳到第3步，发送下一个区块，直至所有数据发送完毕，接着执行第4步。
   2. 如果接收到RSP[其他错误码]，则终止当前升级。
   3. 如果没有接收到RSP，则终止当前升级
4. APP发送命令” UpdateAppDone REQ”给DEVICE。
   1. 如果接收到RSP[SUCCESS]，则认为固件升级成功。
   2. 如果接收到RSP[其他错误码]，认为固件升级失败，终止当前升级。
5. 升级结束。

注意：无论最终升级成功或者失败，Device都会自动复位。APP无须再发送复位命令“MCU Reset”。

# 蓝牙名称定义

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 制造商代码 | 蓝牙广播名称 | 产品描述 |
| 0 | 41 | IMTSEB-###### | 电动助力车, ######表示蓝牙MAC的后6个字符。 |
| 1 | 30 | IMT60-###### | 普通中控, ######表示蓝牙MAC的后6个字符。 |
| 2 | 30 | IMT-Bxx-###### | Bxx，xx代表数字, 电池的外置模组, ######表示蓝牙MAC的后6个字符。  例如B9电池BLE的MAC=24:B1:3C:2A:1B:0F  蓝牙名称为IMT-B9-2A1B0F |
| 3 | 30 | IMT-Cxx-###### | Cxx，中控的外置模组, ######表示蓝牙MAC的后6个字符。  例如C7电池BLE的MAC=24:B1:3C:2A:1B:0F  蓝牙名称为IMT-C7-2A1B0F |

# 日志格式定义

日志存储空间：为32K，分为8个扇区存储，每个扇区为4K Bytes。

日志存储方式：循环写入，当所有扇区全部写满内容，则删除最早的记录内容，写入新的记录内容。

日志由多条记录组成，每条记录的长度固定，为8个字节，因此，日志空间最多可存储32K/8 = 4096条记录。

记录格式如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Name | Type | Value | Descriptor |
| 0 | Head | UINT8 | 0x21 | Log Head。  BIT[0-2]:Version;  BIT[3-7]:Reserved; |
| 1-4 | dateTime | UINT32 |  | 日期时间，unix时间戳格式，从1970年1月1日（UTC/GMT的午夜）开始所经过的秒数，不考虑闰秒。 |
| 5 | eventID | UINT8 |  | 时间ID，参考下表。 |
| 6 | Param1 | UINT8 |  | 参数1 |
| 7- | Param2 | UINT8 |  | 参数2 |

## 日志EventID定义

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 日志Event ID | 值 | 参数1 | | | | 参数2 | | | 说明 |
| ET\_SYS\_RESET = 1  (硬复位源，软复位原因) | 1 | 00000001 | | | 上电复位 | 0 | | 看门狗，上电等 | 系统复位 |
| 00000010 | | | 复位脚复位 | 1 | | 修改固件版本号 |
| 00000100 | | | 看门狗复位 | 2 | | 修改硬件版本号 |
| 00010000 | | | 欠压复位 | 3 | | 修改服务器地址 |
| 00100000 | | | MO复位信号 | 4 | | 固件升级失败 |
| 10000000 | | | CPU 复位 | 5 | | 网络复位 |
|  | | | | 6 | | 升级固件成功 |
| 7 | | 收到蓝牙复位命令 |
| 8 | | 恢复出厂设置 |
| 9 | | 切断18650供电 |
| 10 | | 设备激活 |
| 11 | | 拔出所有电池 |
| 12 | | 短消息复位 |
| 13 | | pms固件升级成功 |
|  | |  |
|  | |  |
| ET\_SYS\_SLEEP | 2 | 打卡剩余分钟 | | | | 18650电压（0.1V） | | | 系统休眠 |
| ET\_SYS\_WAKEUP  (设备状态值，复位原因) | 3 | 设备状态值。  BIT[0]:去激活  BIT[1]: 打卡使能  BIT[2]:是否断电  BIT[3]:是否警戒模式  BIT[4]:电池身份验证使能  BIT[5-7]:保留 | | | | 0 | 上电复位 | | 系统唤醒 |
| 1 | 陀螺仪唤醒 | |
| 2 | PMS唤醒 | |
| 3 | SIM唤醒 | |
| 4 | 蓝牙唤醒 | |
| 5 | RTC唤醒 | |
|  |  | |
| ET\_SYS\_ACTIVE | 4 | 设备状态值，同上 | | | | 0 | | | 设备激活 |
| ET\_SYS\_INACTIVE | 5 | 设备状态值，同上 | | | | 0 | | | 设备去激活 |
| ET\_SYS\_DISCHARGE\_ON | 6 | 设备状态值，同上 | | | | 0 | | | 远程断电-开 |
| ET\_SYS\_DISCHARGE\_OFF | 7 | 设备状态值，同上 | | | | 0 | | | 远程断电-关 |
| ET\_SYS\_ALARM\_MODE\_ON | 8 | 设备状态值，同上 | | | | 0 | | | 警戒模式开 |
| ET\_SYS\_ALARM\_MODE\_OFF | 9 | 设备状态值，同上 | | | | 0 | | | 警戒模式关 |
| ET\_SYS\_SIGN\_FAILED | 10 | 设备状态值，同上 | | | |  | | | 打卡失败 |
|  | | | | | | | | | |
| ET\_SIM\_PWR\_RST  (复位原因，CSQ) | 20 | 0 | 上电复位 | | | CSQ | | | SIM复位 |
| 1 | 网络看门狗 | | |
| 2 | AT命令错误重试失败 | | |
| 3 | 网络远程复位 | | |
| 4 | TCP连接服务器错误 | | |
| 5 | 关闭网络连接错误 | | |
| 6 | 网络状态异常 | | |
| 7 | SIM卡异常 | | |
| 8 | SIM卡不在位 | | |
| 9 | 服务器连接错误 | | |
| 10 | 调试串口指令 | | |
| 11 | SIM模组初始化失败 | | |
| ET\_SIM\_SLEEP | 21 | HbCount（发送心跳次数） | | | | GpsCount  （发送定位次数） | | | SIM模组睡眠 |
| ET\_SIM\_WAKEUP | 22 | 设备状态值，同上 | | | | 18650电压（0.1V） | | | SIM唤醒 |
|  | | | | | | | | | |
| ET\_GPRS\_CNT | 30 | 打卡剩余分钟 | | | | CSQ | | | 网络连接 |
| ET\_GPRS\_DIS\_CNT  (中断原因，CSQ) | 31 | 0 | 其他原因 | | | CSQ | | | 网络断开 |
| 1 | TCP连接关闭 | | |
| 2 | IP INITIAL | | |
| 3 | PDP DEACT | | |
| ET\_GPRS\_SEND\_FAILED | 32 | 18650电压（0.1V） | | | | CSQ | | | 发送数据失败 |
| ET\_GPRS\_HEARBEAT\_COUNT | 33 | HbCount,心跳次数 | | | | GpsCount，定位次数 | | | 心跳技术，每10次记录一次 |
| ET\_GPRS\_SMS | 34 | 1 | | 寻车 | | CSQ | | | 接收到短消息 |
| 2 | | 系统复位 | |
| FF | | 其他。 | |
| ET\_GPRS\_UPG\_START | 35 | 升级固件开始 | | | | 是否点火,  0: ACC OFF，可以升级  1: ACC ON,不能启动升级 | | | BIT0：升级PMS固件。  BIT1：升级SMART固件  其他值：保留 |
| ET\_GPRS\_UPG\_PROC | 36 | 升级固件过程 | | | | 0：网络协议切换到正常 | | | 18650电压（0.1V） |
| 1：网络协议切换到升级 | | | 18650电压（0.1V） |
| 2：获取固件URL | | | 0：成功；其他值：失败。 |
| 3：获取固件文件长度 | | | 0：获取成功，状态码=200。  1：获取失败。  2：文件长度无效。 |
| 4：下载文件数据 | | | 包技术，每包4K，最有一个包可能不满4K |
| 5：下载完毕 | | | 0：成功；1：失败。 |
|  | | | | | | | | | |
| ET\_GPS\_PWR\_ON | 40 | 18650电压（0.1V） | | | | CSQ | | | GPS上电 |
| ET\_GPS\_PWR\_OFF | 41 | 18650电压（0.1V） | | | | CSQ | | | GPS关电 |
| ET\_GPS\_LOC\_OK | 42 | 可见卫星数 | | | | SNR | | | 定位成功 |
| ET\_GPS\_LOC\_FAILED | 43 | 可见卫星数 | | | | SNR | | | 定位失败 |
|  | 44 |  | | | |  | | |  |
|  | | | | | | | | | |
| ET\_BLE\_CNT | 50 | 0 | | | | 0 | | | 蓝牙连接 |
| ET\_BLE\_DIS\_CNT | 51 | 0 | | | | 0 | | | 蓝牙断开 |
|  | | | | | | | | | |
| ET\_PMS\_ACC\_ON | 60 | 打卡剩余分钟 | | | | SOC | | | 点火 |
| ET\_PMS\_ACC\_OFF | 61 | 打卡剩余分钟 | | | | SOC | | | 熄火 |
| ET\_PMS\_BAT\_PLUG\_IN | 62 | 电池总数 | | | | SOC | | | 电池插入 |
| ET\_PMS\_BAT\_PLUG\_OUT | 63 | 电池总数 | | | | SOC | | | 电池拔出 |
| ET\_PMS\_BAT\_VERIFY | 64 | 电池验证 | | | | 0 | | | SOC |
| ET\_PACK\_STATE\_CHANGED | 65 | 电池状态改变 | | | | 电池状态：  0-电池休眠  1-电池充电。  2-电池放电。 | | | SOC |
| ET\_PMS\_COMM\_EVENT | 66 | PMS通信事件 | | | | 通信状态。  0-中断。  1-连接。 | | | SOC |
| ET\_PMS\_PWR\_EVENT | 67 | PMS上电事件 | | | | PMS端口上电状态。  0-断电。  1-上电。 | | | SOC |
| ET\_PMS\_SET\_DISCHARGE | 68 | 设置PMS放电模式 | | | | Bit[0-1]:Port 0电池身份校验。  Bit[6]: 是否允许放电。  Bit[7]: 是否警戒模式 | | | Port1电池身份校验。  0-没校验。  1-校验成功。  2-校验失败 |
|  | | | | | | | | | |
| ET\_UPGRADE\_SMART\_START | 80 | 18650电压（0.1V） | | | | 0 | | | 升级SMART板固件开始 |
| ET\_UPGRADE\_SMART\_DONE | 81 | 18650电压（0.1V） | | | | 0-成功；1-失败 | | | 升级SMART板固件结束 |
| ET\_UPGRADE\_PMS\_START | 82 | 18650电压（0.1V） | | | | 0 | | | 升级SMART板固件开始 |
| ET\_UPGRADE\_PMS\_DONE | 83 | 18650电压（0.1V） | | | | 0-成功；1-失败 | | | 升级PMS板固件结束 |

# 附录1：Firmware Version定义

MCU Firmware version adopts **GNU** style(Including 4 parts):

Major\_Version\_Number**.**Minor\_Version\_Number**.**Revision\_Number**.**Build\_Number

* Main Version Number (**主版号，1字节**)

从1开始，当项目在进行重大修改或局部修正较多，而导致项目整体发生全局变

化时，主版本号加1.

* Minor Version Number (**子版本号，1字节**)

当项目在原有的基础上增加了部分功能时，主版本号不变，子版本号加1，修正

版本号复位成0.

* Revision Number **(修正版本号，1字节)**

当项目进行了局部修改或bug修正时，主版本号和子版本号都不变，修正版本号

加1.

* Build Number **(编译版本号，4字节)**

Build Number是不断递增的，如果IDE比较智能的话，每次打包发布时，会自 动加1。如果不是自动加1的话，每次对外发布新的firmware时，都需要手动加1或者参考SVN修订号.

# 附录2：Hardware Version定义

Hardware adopts the below version style(Including 2 parts):

Device\_Typer. Major\_Version\_Number

Product Typer定义如下：

|  |  |
| --- | --- |
| Product Typer | Description |
| 1 | 电池仓 Smart |
| 其他值 | 保留 |

# 附录3：设备类型定义 +

C7和C9代表不同的设备型号，C7支持PMS板和双电池，C9仅支持单电池，没有PMS板，具体的设备特性请参考表格 75

表格 75设备类型支持模块定义

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 功能描述 | C7 | C9 |
|  | 2G/4G | **✔** | **✔** |
|  | GPS功能 | **✔** | **✔** |
|  | BLE | **✔** | **✔** |
|  | PMS板 | **✔** | **×** |
|  | 电池 | 2 | 1 |
|  | 18650电池 | **✔** | **×** |
|  | 陀螺仪 | **✔** | **✔** |
|  | 喇叭 | **✔** | **×** |
|  | 轮毂锁 | **✔** | **×** |
|  | 座舱锁 | **✔** | **×** |
|  | 控制器 | **×** | **×** |
|  | 点火线 | **✔** | **×** |
|  |  |  |  |

# 附录4：设备支持命令定义 +

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 命令 | 是否需要  身份认证 | C7 | C9 |
|  | Authentication | **×** | **✔** | **✔** |
|  | GetPortState | **×** | **✔** | **✔** |
|  | Get Battery Info | **×** | **✔** | **✔** |
|  | Get SelfTest Result | **×** | **✔** | **✔** |
|  | Get GPS/GPRS Info | **×** | **✔** | **✔** |
|  | Get Device Capacity | **×** | **✔** | **✔** |
|  | Get SmartInfo | **×** | **✔** | **✔** |
|  | Get PMS Info | **×** | **✔** | **✔** |
|  | Get BMS Info | **×** | **✔** | **✔** |
|  | Active Device | **✔** | **✔** | **✔** |
|  | Battery Verify Req | **✔** | **✔** | **✔** |
|  | Set Nvds | **✔** | **✔** | **✔** |
|  | Get Nvds | **×** | **✔** | **✔** |
|  | Set Factory Setting | **✔** | **✔** | **✔** |
|  | Set Alarm Mode | **✔** | **✔** | **✔** |
|  | Set LowSocPlay | **×** | **×** | **×** |
|  | Set SocPlay | **×** | **×** | **×** |
|  | Get SocPlay | **×** | **×** | **×** |
|  | Read Log Info | **×** | **✔** | **✔** |
|  | Log Seek By Index | **×** | **✔** | **✔** |
|  | Log Seek By Data Time | **×** | **✔** | **✔** |
|  | Read Log Records | **×** | **✔** | **✔** |
|  | Delete Log | **✔** | **✔** | **✔** |
|  | SetStopCondition | **✔** | **✔** | **✔** |
|  | GetRunParam | **×** | **✔** | **✔** |
|  | SetWheelLockState | **✔** | **✔** | **×** |
|  | SetCabinLockState | **✔** | **✔** | **×** |
|  | SetAccState | **✔** | **✔** | **×** |
|  | Horn Test | **×** | **×** | **×** |
|  | Get Asy Info | **×** | **×** | **×** |
|  | Get Asy State | **×** | **×** | **×** |
|  | Set Asy | **✔** | **×** | **×** |
|  | Get Beacon Scan | **×** | **✔** |  |
|  | Get Beacon | **×** | **✔** |  |