**机器人遥控协议**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **文档版本号** | **更新时间** | **修订人** | **审定人** | **备注** |
| **0.0.1** | **2018-01-02** | **刘欢** |  |  |
| **0.0.2** | **2018-02-01** | **刘欢** |  | 1. 部分命令校验->所有命令校验 2. 新增命令包响应规范 3. 部分命令取消响应，当需要判断是否设置成功时，需要将状态读回判断 |
| **0.0.3** | **2018-03-15** |  |  | 1. 增加舵机控制命令，53H，54H，对一系列舵机进行动作 2. 增加文件读写相关命令 |
| **0.0.4** | **2018-05-22** |  |  | **适用软件1.0.3** |
| **0.0.5** | **2018-06-22** |  |  | **删除和添加设备类型0x0A中的部分授权命令** |
|  |  |  |  |  |

目录

[1. 通信方式描述 4](#_Toc527622272)

[2. 协议格式 5](#_Toc527622273)

[3. 一级命令对照表（此命令不再使用，保留目的仅仅为兼容老版本机器，后期逐步淘汰） 6](#_Toc527622274)

[4. 二级命令长包设备控制 8](#_Toc527622275)

[5. 二级命令短包设备控制 9](#_Toc527622276)

[6. 命令详细描述格式说明 10](#_Toc527622277)

[6.1.1. 命令描述具体细则 10](#_Toc527622278)

[6.1.2. (设备类型：0x01)语音设备 11](#_Toc527622279)

[6.1.3. (设备类型：0x02)WIFI设备 12](#_Toc527622280)

[6.1.4. (设备类型：0x06)LED指示灯 13](#_Toc527622281)

[6.1.5. (设备类型：0x07)总线设备（舵机，武器，霍尔） 14](#_Toc527622282)

[6.1.6. (设备类型：0x08)底盘设备 19](#_Toc527622283)

[6.1.7. (设备类型：0x09)整机控制命令 21](#_Toc527622284)

[6.1.8. (设备类型：0x0A)授权命令 27](#_Toc527622285)

[6.1.9. (设备类型：0x10)局域网通信（老版本APP使用，往后不再支持） 28](#_Toc527622286)

[7. Bootloader描述 30](#_Toc527622287)

[7.1. USB优盘功能 30](#_Toc527622288)

[7.2. 刷新程序文件 30](#_Toc527622289)

[7.2.1. 第一步，准备程序文件 30](#_Toc527622290)

[7.2.2. 第二步，进入刷新程序流程 30](#_Toc527622291)

[7.2.3. 第三步，等待刷新完成 30](#_Toc527622292)

[8. 武器说明 31](#_Toc527622293)

[9. 战斗模式特效 31](#_Toc527622294)

[10. LED状态说明 32](#_Toc527622295)

[11. 音效规范 33](#_Toc527622296)

[11.1. 音效说明 33](#_Toc527622297)

[11.2. 音效编号定义 33](#_Toc527622298)

[11.2.1. 技能音效 33](#_Toc527622299)

[11.2.2. 其它音效 36](#_Toc527622300)

[11.2.3. 被攻击音效 37](#_Toc527622301)

[12. 机身伤害点位编号 38](#_Toc527622302)

[13. Flash使用说明 39](#_Toc527622303)

[14. 程序版本对应 40](#_Toc527622304)

[15. 错误编号值 40](#_Toc527622305)

1. 通信方式描述

机器人采用wifi遥控方案，WIFI初始状态下默认为AP模式，在该模式下自动创建一个SSID为“MYAP-\*\*\*\*\*\*”（\*\*\*\*\*\*为改WIFI模块的MAC地址后6位）开头的热点，默认密码是“88888888”。上电时底层通信默认采用UDP传输，通信两端的端口号固定为270FH(9999)。具体的数据格式详细描述参考协议描述。

通信终端需要连接上该wifi或与wifi保持在同一个网络，创建一个UDP服务，监听指定数据端口即可获取到机器人发送的数据，或向指定的端口发送数据即可向机器人发送数据。

1. 协议格式

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **完整包** | | | | | | |
| **头1** | **头2** | **命令1** | **命令2** | **数据长度** | **数据** | **校验和取反** |
| H1 | H2 | CMD1 | CMD2 | DATA\_Length | DATA数据包 | check |

* + - 1. 协议描述：
* 头1：H1固定FEH，文章后面所有的头1用H1表示
* 头2：H2固定EFH，文章后面所有的头2用H2表示
* 命令1：

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 位 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | B0 |
| 描述 | ACK | LP | ACKF | 保留 | 保留 | 保留 | 保留 | 保留 |
| 缺省值 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**ACK：使能响应**

该位控制接收端是否响应此命令，ACK=1响应，ACK=0不响应。

**LP：使能长包**

该位控制包长，如果LP=1，则数据长度占用2字节即最大数据长度为0-65535(实际通信限制为1400)，如果LP=0，则数据长度占用1字节即最大数据长度为0-255。

**ACKF：响应包标志**

ACKF=1，如果数据发送方在发送数据包时将ACK设置为1，则数据接收方成功收到数据后会响应一个数据包，响应的数据包中ACKF=1。

* 命令2：CMD2，默认为0
* 数据长度：表示DATA的长度，参考CMD1>LP，当数据长度为多字节时，数据长度为大端MSB（即高位在前）
* 数据：协议包所包含的数据信息
* 校验和取反：求和后取反

具体计算方法为：check=~( CMD1 + CMD2 + DATA\_Length + DATA1 + DATA2… + DATAn )

**注：**

**1.特定的命令需要特定的长度，数据包字节偏多或偏少都会使设备认为该命令错误，从而不执行。**

**2.文章中涉及到数据的地方以H结尾的数据均为十六进制数据（如：7FH），其它数据为十进制数据**

**3．长包通信数据长度限制为1400**

1. 一级命令对照表（此命令不再使用，保留目的仅仅为兼容老版本机器，后期逐步淘汰）

主命令主要用于对WIFI模块的直接通信，所有主命令均可以在未经密码授权的方式下配置生效（子命令有部分命令必须在通过密码验证后才能生效）

| 功能描述 | CMD1 | CMD2 | 数据长度 | DATA | 响应 |
| --- | --- | --- | --- | --- | --- |
| 系统内部使用 | 0 | 1 |  |  |  |
| 查询WIFI-Station连接的AP名称 | 1 | 0 | 0 |  | H1 H2 01H 00H length DATA CHECK  length：表示名称长度  DATA：为名称实体 |
| 设置WIFI-Station连接的AP名称 | 1 | 1 | 1-32 |  | H1 H2 01H 00H length DATA CHECK  length：表示名称长度  DATA：为名称实体 |
| 查询WIFI-Station连接的AP密码 | 2 | 0 | 0 |  | H1 H2 02H 00H length DATA CHECK  length：表示密码长度  DATA：为密码实体 |
| 设置WIFI-Station连接的AP密码 | 2 | 1 | 0-64 |  | H1 H2 02H 00H length DATA CHECK  length：表示密码长度  DATA：为密码实体 |
| 查询模块当前的WIFI工作模式 | 3 | 0 | 0 |  | H1 H2 03H 00H 01H mode CHECK  mode=1:STATION模式  mode=2:AP模式（默认） |
| 设置模块为Station模式，模块收到此命令后延迟0.5秒后执行 | 3 | 1 |  |  |  |
| 设置模块为AP模式，模块收到此命令后延迟0.5秒后执行 | 3 | 2 |  |  |  |
|  |  |  |  |  |  |
| 查询当前WIFI串口传输模式 | 7 | 0 |  |  | H1 H2 07H 00H 01H mode 00  mode =1:透传模式  mode =2:协议模式（默认） |
| 设置WIFI串口为透传模式 | 7 | 1 |  |  | 无返回 |
| 设置WIFI串口为协议模式 | 7 | 2 |  |  | 无返回 |
| 设置WIFI串口为透传模式，保存到flash | 7 | 3 |  |  | 无返回 |
| 设置WIFI串口为协议模式，保存到flash | 7 | 4 |  |  | 无返回 |
|  |  |  |  |  |  |
| 查询当前串口输出波特率 | 8 | 0 | 0 |  | H1 H2 08H 00H 01H 波特率 CHECK  波特率=0:出错  波特率=1:9600  波特率=2:115200  波特率=3:921600（默认） |
| 设置当前串口波特率 | 8 | 1 | 1 | 1:9600  2:115200  3:921600 | 无返回 |
| 设置当前串口波特率并保存到flash | 8 | 2 | 1 | 参考命令CMD1=8，CMD2=1 | 无返回 |
|  |  |  |  |  |  |
| 读取当前程序运行编号 | 09H | 01H | 00H |  | 返回一个字节  =0：user1  =1：user2 |
| 读取当前程序版本 | 09H | 02H | 00H |  | 返回两个字节  byte0：版本高位  byte1：版本低位 |
| 切换当前程序执行区域  且还前确保程序正确烧录到flash中否则模块将死机 | 09H | 03H |  |  |  |
|  |  |  |  |  |  |
| 查询指定IO状态 | 10 | 0 | 1 | IO编号 | H1 H2 0AH 00H 02H 编号 电平 CHECK  编号：可取12或13或14  电平：0，1 |
| 设置IO输出电平 | 10 | 1 | 2 | D0=编号  D1=电平  参考命令CMD1=10，CMD2=0 | 无返回 |
|  |  |  |  |  |  |
| 设置WIFI模块作为TCP客户端  参数为目标TCP服务器IP地址和端口号 | 11 | 1 | 6 | IP1.IP2.IP3.IP4+PORTH+PORTL | x |
| 设置WIFI模块作为TCP服务器  参数为TCP服务器的端口号 | 11 | 2 | 2 | PORTH+PORTL | x |
| 设置WIFI模块作为UDP服务器  参数为UDP服务器的端口号 | 11 | 3 | 2 | PORTH+PORTL | x |
|  |  |  |  |  |  |
| 写内部FLASH，程序升级 | 60H | 01H | \*\* | 流水号（1byte）  文件起始地址（4byte 大端四字节对齐）  数据长度（2byte 大端四字节对齐）  数据（不定长） | 流水号（1byte）  起始地址（4byte 大端）  数据长度（2byte 大端）  数据（不定长） |
|  |  |  |  |  |  |
| 读内部FLASH，程序升级 | 66H | 01H |  | 读Flash起始地址（4byte 大端四字节对齐）  数据长度（2byte 大端四字节对齐） | 文件起始地址（4byte 大端四字节对齐）  数据长度（2byte 大端四字节对齐） |

1. 二级命令长包设备控制

当一级包的参数为长包时，二级包的参数长度同样为长包，均占用两个字节长度。在实际操作过程中，可以理解为当命令参数长度超过255时将长度自动扩充为2个字节，并通过CMD1标识。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **完整包** | | | | | | | |
| **头（2）** | **CMDx（2）** | **DATA\_LEN（2）** | **DATA数据包** | | | | **校验（1字节）** |
| **设备类型（1字节）** | **设备命令（1字节）** | **命令参数长度（2）** | **命令参数（最大1400字节）** |
| H1 H2 | 00 00 | lenH lenL | DEVICE\_Type | DEVICE\_cmd | CMD\_ parameter\_len | CMD\_ parameter | check |

注：协议规定在长包中延伸出的二级命令中命令参数最大不能超过1400个字节

1. 二级命令短包设备控制

二级命令的意思是从一级命令的数据中再次将数据部分按照特定含义将数据包拆分，如下表中将数据部分再次拆分为设备类型，设备命令，命令参数长度和命令参数。

一般控制命令比较短，故大部分设备控制命令采用短包通信，并从中延伸出二级命令。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **完整包** | | | | | | | |
| **头（2）** | **CMDx（2）** | **DATA\_LEN（1）** | **DATA数据包** | | | | **校验（1字节）** |
| **设备类型（1字节）** | **设备命令（1字节）** | **命令参数长度** | **命令参数（最大230字节）** |
| H1 H2 | 00 00 | X | DEVICE\_Type | DEVICE\_cmd | CMD\_ parameter\_len | CMD\_ parameter | check |

注：协议规定在短包中延伸出的二级命令中命令参数最大不能超过230个字节

1. 命令详细描述格式说明

在命令的具体实现操作过程中，通过命令参数的长度大小自动选择长包或短包传输。

* + 1. 命令描述具体细则

协议子块：

格式为 [块描述(子块长度)] 《附加描述》

子块长度：

格式为(子块字节长度十进制)，\*表示不定长

附加描述：

用《》包含，用于备注其它信息，此信息可以没有，也可以有多个

其它说明

LSB表示小端：低字节在前

MSB表示大端：高字节在前

* + 1. (设备类型：0x01)语音设备

此设备命令需要验证机器人密码有效后生效

| 描述 | 命令 | 长度 | 发送数据包 | 响应数据包 |
| --- | --- | --- | --- | --- |
| 查询当前音量 | 01H | 00H | H1 H2 00H 00H 03H 01H 01H 00H CHECK | H1 H2 00H 00H 04H 01H 01H 01H D0 CHECK  D0:音量值 |
| 查询当前播放器状态 | 02H | 00H | H1 H2 00H 00H 03H 01H 02H 00H CHECK | H1 H2 00H 00H 04H 01H 02H 01H D0 CHECK  D0=0：空闲状态  D0=1：正在播放 |
| 读取或设置音量 | 0x08 | 01H | H1 H2 00H 00H 03H 01H 08H 01H D0 CHECK  D0=音量值,音量范围0-31 | 当数据长度为0时表示查询当前静音状态 |
| 读取或设置静音模式 | 0x09 | 01H | H1 H2 00H 00H 03H 01H 09H 01H D0 CHECK  D0=1,使能静音 D0=0：失能静音 | 当数据长度为0时表示查询当前静音状态 |
| 播放/暂停 | 0x11 | 00H | H1 H2 00H 00H 03H 01H 11H 00H CHECK |  |
| 暂停（保留） | 0x12 | 00H |  |  |
| 快进（保留） | 1AH |  |  |  |
| 快退（保留） | 1BH |  |  |  |
| 停止 | 0x1E | 00H | H1 H2 00H 00H 03H 01H 1EH 00H CHECK |  |
| 播放指定文件夹中指定文件 | 0x42 | 02H | H1 H2 00H 00H 05H 01H 42H 02H D0 D1 CHECK  D0：文件夹编号，0-9999  D1：文件编号，0-9999 |  |
| 播放指定名称的文件夹中指定名称的文件 | 70H | \* | [文件名(\*)]  《可包含完整路径，采用\分割》 | H1 H2 00H 00H 04H 01H 70H 01H D0 CHECK  D0=0:发送成功  否则发送失败 |

* + 1. (设备类型：0x02)WIFI设备

| 设备类型 | 命令 | 长度 | 发送数据包 | 响应数据包 | 描述 |
| --- | --- | --- | --- | --- | --- |
| 02H | 01H | XXH | [操作类型uint8 (1)]  《0：查询 1：station模式 2：AP模式》  《查询时没有后续参数》  [名称长度uint8 (1)]  《小于32》  [密码长度uint8 (1)]  《小于64》  [名称ASC(\*)]  [密码ASC(\*)] | * 查询返回   [模式uint8(1)]  《0：查询 1：station模式 2：AP模式》 | 设置或查询wifi模式  VES>=4000 | |
|  |  |  |  |  |  | |
|  | 03H | 00H | 无参数 | * 查询返回   [版本高位uint8(1)]  [版本低位uint8(1)] | 查询wifi固件版本  VES>=4000 | |
|  | 04H |  |  |  |  | |

* + 1. (设备类型：0x06)LED指示灯

此设备命令需要验证机器人密码有效后生效

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 设备类型 | 命令 | 长度 | 发送数据包 | 响应数据包 | 描述 |
| 0x06 | 0x41 | 03H | H1 H2 00H 00H 03H 06H 41H 03H D0 D1 D2 CHECK  D0：R（红）0-255  D1：G（绿）0-255  D2：B（蓝）0-255 |  | 设置彩灯1显示的RGB值 |
|  | 0x42 | 03H | 参考同设备0x41命令 |  | 参考同设备0x41命令返回 |
|  | 0x43 | 03H | 参考同设备0x41命令 |  | 参考同设备0x41命令返回 |
|  | 0x44 | 03H | 参考同设备0x41命令 |  | 参考同设备0x41命令返回 |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

* + 1. (设备类型：0x07)总线设备（舵机，武器，霍尔）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 版本 | 命令 | 长度 | 发送包数据 | 响应包数据 | 描述 |
|  | 01H | 01H | 数据长度1字节  D0=1：脱落D0=2：安装D0=3：夹紧 |  | 设置左手臂夹持舵机状态 |
|  | 02H | 01H | 参考同设备01H命令 |  | 设置右手臂夹持舵机状态 |
|  | 03H | 00H | H1 H2 00H 00H 03H 07H 03H 00H CHECK | H1 H2 00H 00H 05H 07H 03H 02H D0 D1 CHECK  D0(左手)=1：脱落 =2：安装 =3：夹紧  D1(右手)=1：脱落 =2：安装 =3：夹紧 | 查询手臂夹持舵机状态 |
|  | 04H | 00H | H1 H2 00H 00H 03H 07H 04H 00H CHECK | H1 H2 00H 00H 05H 07H 04H 02H D0 D1 CHECK  D0=1：左手臂安装 D0=0左手臂未安装  D1=1：右手臂安装 D1=0右手臂未安装 | 查询左右手臂是否已近安装安装状态 |
|  | 05H | 01H  或  02H | 如果长度=1  D0：表示失能或使能（0=失能，1=使能）  如果长度=2  D0：表示舵机ID（254表示广播）  D1：表示失能或使能（0=失能，1=使能） |  | 控制部分或全部舵机失能或使能，失能后舵机不输出力矩 |
|  | 06H | 00H |  |  | 失能并复位动作 |
|  | 07H | 01H |  |  | 控制舵机掉电指定时间，然后恢复供电，单位时间0.1s，时间值必须大于0 |
| VES>=4000 | 08H | 00H |  |  | 重载动作文件，按照当前武器重载对应动作 |
| VES>=4000 | 09H | \*\*H |  |  | 重载指定动作文件 |
|  | 0AH | 00H |  |  | 刷新武器 |
| VES>=4000 | 0BH | 01H |  | 转换返回  [文件编号uint8(1)]  [转换结果uint8(1)]  《0：失败 1：成功》 | 老动作文件转新动作文件 |
|  | 20H |  | D0：动作编号 | 返回0表示失败  返回非0表示成功，表示技能长度 | 启动指定动作编辑 |
|  | 21H |  |  | 返回2个字节  D0：0表示命令出错，1表示成功  D1：表示当前编辑位置 | 上一个动作节点 |
|  | 22H |  |  | 返回2个字节  D0：0表示命令出错，1表示成功  D1：表示当前编辑位置 | 下一个动作节点 |
|  | 23H |  | 1：复制当前动作添加  2：添加一个默认动作 | 返回2个字节  D0=0：表示命令出错，否则表示当前动作总长度  D1表示当前动作位置 | 在当前动作后面添加一个动作节点 |
|  | 24H |  |  | 返回2个字节  D0=0：表示命令出错，否则表示当前动作总长度  D1表示当前动作位置 | 删除当前动作节点，如果当前动作为最后一个动作，删除后会添加一个默认动作 |
|  | 25H |  |  |  | 单次预览 |
|  | 26H |  |  |  | 整体预览 |
|  | 27H |  |  | 返回1个字节  0：表示命令出错  1表示保存成功 | 保存当前动作 |
|  | 28H | 00H |  | sit1,sit2,sit3,sit4,sit20,sit21,sit31,sit32,sit33,sit34：对应手臂舵机位置（uint16小端）  time1：舵机运行时间毫秒，相当于速度（uint16小端）  speed：保留（uint16小端）  delaytime：动作延迟时间毫秒（uint16小端）  MSwtich：底盘启用开关0=关，1=开（uint8）  MAngle：底盘移动角度0-360（uint16小端）  MSpeed：底盘移动速度0-100（uint16小端）  MTurn：底盘转弯速度-1000~+1000（uint16小端）  MRuntime：底盘运行时间毫秒（uint16小端） | 获取当前动作节点状态 |
|  | 29H | 23H | 参考CMD28 | 返回0表示失败，否则表示成功 | 设置当前动作节点状态 |
|  | 2AH |  |  | D0：0表示失败1：表示成功 | 清空当前动作，清空后会添加一个默认动作  技能总长度为1，且当前编辑位置为0 |
|  | 2BH |  |  |  | 定位到当前技能的起始节点动作 |
|  | 2CH | 01H | 位置参数 | D0：返回0表示失败，返回1表示成功 | 定位到技能的指定动作节点 |
|  | 2DH | 01H | 指定动作节点编号 | 返回2字节  D0：返回0表示失败，否则表示当前动作长度  D1：当前动作位置 | 删除指定动作节点 |
|  | 2EH | 00H |  | D0：返回动作节点总数  D1：返回当前节点编号值 | 读取指定动作的节点总数和当前所在节点编号 |
|  | 2FH | 02H | D0:添加的起始位置  D1：添加的动作类型  1：复制当前动作添加  2：添加一个默认动作 | 返回2个字节  D0=0：表示命令出错，否则表示当前动作总长度  D1表示当前动作位置 | 在指定位置后面添加一个指定动作 |
|  | 30H | 02H | D0=A,D1=B | 返回1字节，0表示失败，1表示成功 | 将动作节点A到复制到B |
|  | 53H | 00H |  | 返回动作名列表，不同的动作名之间使用‘,’分割 | 读取当前动作名列表 |
|  | 54H | \*\*H | 参数为指定动作名称的UTF8编码 |  | 执行指定名称动作 |
|  | 55H | 01H | H1 H2 00H 00H 04H 07H 55H 01H D0 CHECK  D0=0-3：动作1-4  D0=4: 普攻  D0=5： 失败动作  D0=6： 胜利动作  D0=7： 开机动作  D0=8： 战斗开始动作 |  | 手臂动作控制 |
|  | 56H | 01H | H1 H2 00H 00H 04H 07H 56H 01H D0 CHECK  D0=舵机ID,需要读取位置的舵机ID | H1 H2 00H 00H 06H 07H 56H 03H D0 D1 D2 CHECK  D0：读取位置的舵机ID  D1：位置高位  D2：位置低位 | 读取指定ID的舵机当前位置 |
|  | 58H | 02H | H1 H2 00H 00H 05H 07H 58H 02H D0 D1 CHECK  D0：武器ID  D1：LED状态（1亮，0灭） |  | 设置武器LED状态 |
|  | 59H | 00H | H1 H2 00H 00H 03H 07H 59H 00H CHECK |  | 一次性读取ID为1，2，3，4，20，21，31，32，33，34共10个舵机的当前位置 |
|  | 5AH | 02H | H1 H2 00H 00H 05H 07H 5AH 02H D0 D1 CHECK  D0：原设备ID  D1：现设备ID |  | 更改指定ID设备的ID,并保存到EEPROM |
|  | 5CH | 05H | H1 H2 00H 00H 08H 07H 5CH 05H D0 D1 D2 D3 D4 CHECK  D0:舵机ID  D1：位置高位  D2：位置低位  D3：时间高位  D4：时间低位 |  | 设置指定舵机运行到指定位置 |
|  | 5DH | 01H | H1 H2 00H 00H 04H 07H 5DH 01H D0 CHECK  D0：=0主武器,=1副武器 | H1 H2 00H 00H 05H 07H 5DH 02H D0 D1 CHECK  D0：=0主武器 =1副武器  D1：=0武器不存在 =1武器存在 | 查询武器状态  注：武器安装状态更改后，会自动发送当前安装状态到主机 |
|  | 5EH | 03H | D0：id  D1：起始地址  D2：查询数据长度 | D0:id  D1：起始地址  D2：数据长度  Dx-n：数据 | 查询总线设备寄存器值 |
|  | 5FH | 16H | sit1,sit2,sit3,sit4,sit20,sit21,sit31,sit32,sit33,sit34,time  sitx：舵机位置，无符号16位小端（LSB）类型  time: 无符号16位小端（LSB）类型，参数表示对应舵机x执行到sitx的执行时间，相当于设置舵机的执行速度，时间越大速度越慢。 |  | 设置一组舵机动作：  1，2，3，4，20，21，31，32，33，34 |
|  | 60H |  | sit1,sit2,sit3,sit4,sit20,sit21,sit31,sit32,sit33,sit34,time,angle,speed,turn  sitx：舵机位置，无符号16位小端（LSB）类型  time: 无符号16位小端（LSB）类型，参数表示对应舵机x执行到sitx的执行时间，相当于设置舵机的执行速度，时间越大速度越慢。  angle: 底盘角度，int16（LSB）角度0到360  speed: 底盘速度，int16（LSB）速度0-100  turn: 底盘转弯速度，int16（LSB）转弯率-1000到+1000 |  | 同时控制舵机和底盘动作 |
|  | 70H | 04H | H1 H2 00H 00H 07H 07H 70H 04H D0 D1 D2 D3 CHECK  D0：指定设备ID  D1：指定设备内存位置  D2：指定值  D3：保存到flash，1（保存）0（不保存） |  | 设置指定ID设备指定内存位置的值，并选择是否保存到flash（该命令用于程序调试） |
|  | 80H | \*\*H |  |  | 数据透传到总线 |
|  | 81H | \*\*H |  |  | 非读写数据从总线透传到终端 |
|  | E0H | XXH | XXH=0：读取  XXH=2：设置 | 返回2字节：  Uint16，小端  Bit0：左手霍尔（0x01）  Bit1：右手霍尔（0x02）  Bit2：背部霍尔（0x04） | 查询或设置所有启用状态 |
|  | E1H | XXH | XXH=0：读取  XXH=1：设置，0表示不启用，否则表示启用 |  | 查询或设置左手臂霍尔启用状态 |
|  | E2H | XXH | XXH=0：读取  XXH=1：设置，0表示不启用，否则表示启用 |  | 查询或设置右手臂霍尔启用状态 |
|  | E3H | XXH | XXH=0：读取  XXH=1：设置，0表示不启用，否则表示启用 |  | 查询或设置背部霍尔启用状态 |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

* + 1. (设备类型：0x08)底盘设备

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 设备类型 | 命令 | 长度 | 发送数据包 | 响应数据包 | 描述 |
| 0x08 | 0x01 | 01H | H1 H2 00H 00H 04H 08H 01H 01H D0 CHECK  D0=0：单闭环（默认）  D0=2：双闭环 | 当数据长度为0时表示要查询底盘的模式，返回数据参考发送时的数据 | 底盘运动控制模式设置,默认模式0  注：进入战斗模式后需要将模式设置为2，在APP上进入初始界面需要给底盘设置为模式0（闭环模式） |
|  | 0x02 | 6 | H1 H2 00H 00H 09H 08H 02H 06H D0-D5 CHECK  D0 D1: int16（LSB）角度0到360  D2 D3: int16（LSB）速度0-100  D4 D5: int16（LSB）转弯率-1000到+1000 |  | 控制底盘移动（角度、速度、转弯率共3个参数），数据为小端模式（LSB）  角度：正前方为0度，逆时针旋转。  转弯率：值为正方向向左，为负方向向右。 |
|  | 03H | 00H | H1 H2 00H 00H 03H 08H 03H 00H CHECK | H1 H2 00H 00H 0DH 08H 03H 0AH D0-D9 CHECK  D0 D1:uint16（LSB），电池剩余电量0-100  D2 D3:uint16（LSB），电池电压（mV）  D4 D5:int16（LSB），电池电流（mA）  D6 D7:（保留功能）  D8 D9:（保留功能） | 查询电池状态信息, 数据为小端模式（LSB） |
|  | 06H | 08H | H1 H2 00H 00H 0BH 08H 02H 08H D0-D7 CHECK  D0 D1: int16（LSB）角度0到360  D2 D3: int16（LSB）速度0-100  D4 D5: int16（LSB）转弯率-1000到+1000  D6 D7：int16（LSB）运行时间，单位毫秒 |  | 控制舵机运行指定时间 |
|  | 10H | 03H | D0:特效类型  0：正常  1：反向  2：禁止移动  D1-D2：持续时间，Uint16小端类型 |  | 操作特效设置 |
|  | 20H | 01H |  | 数据长度为0表示读取，设置成功后会返回当前设置的值。 | 设置底盘移动速度百分比限制，参数范围0-100 |
|  | 21H | 01H |  | 数据长度为0表示读取，设置成功后会返回当前设置的值。 | 设置底盘转弯速度百分比限制，参数范围0-100 |
|  |  |  |  |  |  |
|  | C0H | XXH |  |  | 数据透传到底盘 |
|  | C1H | XXH |  |  | 底盘数据透传到手机或电脑 |

* + 1. (设备类型：0x09)整机控制命令

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 功能描述 | 命令 | 长度 | 发送描述 | 返回描述 |
| 机器人模式设置或读取 | 0x01 | 01H  或  00H | 长度00H：表示读机器人当前模式  长度01H：表示设置机器人模式  D0=1：初始模式（默认）  D0=2：战斗模式（）  D0=3：失败模式（不可设置）  D0=4：胜利模式 |  |
| USB接口模式 | 02H | 01H | H1 H2 00H 00H 03H 09H 01H 01H D0 CHECK  D0=0：USB烧录口  D0=1：USB文件传输 |  |
| 清除机器人授权验证，清除后机器人处于无验证状态 | 03H | 01H | D0=1：清除机器人授权验证 | 返回1个字节  0：表示当前无验证  1：表示当前已验证 |
| 战斗模式属性来源设置 | 04H |  | * 查询（不带参数） * 设置   [来源]（1字节）  《武器：来源=0，系统当前：来源=1》 | * 查询返回   《参考发送设置》   * 设置返回   不返回 |
| 设置机器人密码，密码最大长度20字节，数据长度即密码长度  注意：该命令会将密码保存到flash中，需要至少延迟200ms，所以不能高频率使用该命令， | 06H | 14H | H1 H2 00H 00H D0+3 09H 06H D0 D1-Dn CHECK  D0:密码长度（0-20）  D1-Dn：密码ASC取值 | 数据包原样返回，由app判断密码是否设置正确 |
| 读取机器人密码 | 07H |  |  |  |
| 查询当前机器人战斗生命值 | 08H | 02H | * 查询（无参数） | * 查询返回   [声明高位](1字节)  [声明低位](1字节) |
| 查询当前机器人指定伤害部位受到的攻击次数  注意：当处于战斗模式时，如果某个部位受到攻击，机器人都会主动将该部位的攻击次数发送到主机,并且包含本次攻击的攻击值。  查询所有部位（VES>=4000） | 09H | 00H  或  01H | * 查询指定部位受攻击次数（1个参数）   H1 H2 00H 00H 04H 09H 09H 01H D0 CHECK  D0：查询的部位点位编号   * 查询所有部位收到攻击次数（0个参数）   H1 H2 00H 00H 04H 09H 09H 00H CHECK  参考：机身伤害点位编号 | * 查询指定部位和主动返回只包含三个字节   H1 H2 00H 00H 06H 09H 09H 03H D0 D1 D2 CHECK  D0：机身伤害点位编号  D1：点位被攻击次数  D2：1:轻攻击，2：重攻击（相当于两次轻攻击）   * 查询所有部位返回18个字节（18个部位）   部位参考：机身伤害点位编号 |
| 查询本机主武器属性 | 0AH | 00H | * 查询（无参数） | * 查询返回   [武器编号]（1）  [性别]（1）《保留》  [生命值]（1）  [轻攻击]（1）  [重攻击]（1）  [移动速度]（1）  [攻击速度]（1）《保留》  [防御]（1）《保留》  [冷却时间]（1）《保留》  [左手耐久]（1）  [右手耐久]（1） |
| 查询副武器属性 | 0BH | 00H |  |  |
| 查询左右手臂生命值  注意：在战斗模式下，受到伤害后，手臂会自动发送当前生命值到app | 0CH | 00H |  | D0:左手生命值uint8  D1:右手生命值uint8 |
| 查询或设置机器人名称，最大长度64字节  注意：编码方式UTF8，名称不能为空 | 0DH | XXH | 当数据长度为0时表示查询  当数据长度不为0时表示设置，最大长度64字节 | 设置成功会返回设置的名称数据  如果返回长度为0表示设置错误 |
| 查询或设置当前战斗属性  老版本：返回HPH，和HPL | 0EH |  | * 查询（无参数） * 设置   《参考命令0AH》 | * 查询返回   《参考设置》   * 设置返回 |
| 读取错误标志指定位置的值  参考15错误编号值 | 10H |  |  |  |
| 查询霍尔基准参考值 | 12H | 00H |  | 返回不定长度，每个霍尔有两个字节，LSB  4霍尔返回8字节  8霍尔返回16字节 |
| 重置霍尔参考基准  尽量在设备空闲时使用，操作耗时1秒  VES>=4000 | 13H | 00H |  | 返回1个字节参数  0：重置成功  1：重置失败，参考值严重偏离  2：失败，其它原因 |
| 临时调试信息 | 14H | 04H |  | 返回4字节  Uint16 MSB最大值  Uint16 MSB 第二大值 |
| 查询或设置敌方人物属性，进入战斗模式后，机器人需要根据敌方的武器伤害计算自己血量 | 20H |  | * 查询（ 不带参数） * 设置   《参考类型09H->命令0AH》 | * 查询返回   《参考类型09H->命令0AH》   * 设置不返回 |
|  |  |  |  |  |
| 更新指定名称文件程序必须是.Bin文件  bootVes>=4000 | 29H | xxH | * 查询（无参数） * 设置（不定长度，必须大于0）   [文件名](最大长度11字节，包括扩展名)  《设置成功后将程序更新开关设置为2》 | * 查询返回   《返回11字节，参考设置》   * 设置返回   《成功返回11字节，其中包含文件名》  《失败返回0个参数》 |
| 设置程序更新开关  注：打开程序更新前请确保SD卡中存在程序Bin文件 | 30H | 00H  或  01H | * 查询（无参数） * 设置（一个字节参数）   [开关标志](1字节)  《开关标志=0：不更新程序》  《开关标志=1：搜索目录下遇到的第一个APP打头的程序文件》  《开关标志=2：更新指定文件名称的程序文件，如果文件不存在就不更新》 | * 查询返回   《返回一个字节，参考设置》   * 设置返回   《返回一个字节，返回设置的数据，参考设置》 |
| 系统重启 | 31H | 00H |  |  |
| 查询程序版本名称 | 32H | 00H | H1 H2 00H 00H 03H 09H 32H 00H CHECK | H1 H2 00H 00H XXH 09H 32H XXH D0-Dn CHECK  D0-Dn:版本名称ASC |
| 查询bootloader版本  （APP2.0.0及以上版本支持） | 33H | 00H | H1 H2 00H 00H 03H 09H 33H 00H CHECK | H1 H2 00H 00H XXH 09H 33H 02H D0 D1 CHECK  D0:版本高位 D1：版本低位 |
| 查询程序版本编号 | 34H | 00H | H1 H2 00H 00H 03H 09H 34H 00H CHECK | H1 H2 00H 00H XXH 09H 34H 02H D0 D1 CHECK  D0:版本高位 D1：版本低位 |
| 查询wifi固件版本 | 35H | 00H |  | H1 H2 00H 00H XXH 09H 35H 02H D0 D1 CHECK  D0:版本高位 D1：版本低位 |
| 图形编程启动运行 | 40H | 00H |  |  |
| 图形编程停止运行 | 41H | 00H |  |  |
| 保留 | 42H | 00H |  |  |
| 图形编程启动/停止 | 43H | 00H |  |  |
| 树莓派启动运行 | 45H | 00H |  |  |
| 树莓派停止运行 | 46H | 00H |  |  |
| 树莓派运行指定程序文件 | 47H | \* |  | 无 |
| 树莓派停止运行指定程序文件 | 48H | \* |  | 无 |
| 树莓派查询当前正在运行的程序文件 | 49H | 00H |  | 当有多个程序文件返回时采用‘，’逗号分割，文件名中不能出现逗号  如：name1,name2,name3 … |
| 树莓派停止运行所有程序 | 4AH | 00H |  | 无 |
| 树莓派查询可运行程序文件 | 4BH | 00H |  | 当有多个程序文件返回时采用‘，’逗号分割，文件名中不能出现逗号  如：name1,name2, name3 … |
| 保存系统参数到flash，收到响应表示保存完成，保存数据期间最好不要发送其它对时间敏感的命令。 | 55H | 00H | H1 H2 00H 00H 03H 09H 55H 00H CHECK | 响应：D0  D0=1：保存成功  D0=0：保存失败 |
| 特定文件写操作（支持长包）  注：单个文件夹或文件名称长度不能超过8字节，当文件名包含完整路径时最大长度不超过64字节，路径采用'\\'分割，中文文件名和路径采用GBK编码  VES>=4000 | 70H | \* | [流水号uint8 (1)]  《需要递增，保证每次都不一样》  [文件名长度(1)]  [文件名(\*)]《最大长度64》  [文件偏移LSB(4)]  [数据长度LSB(4)]  [数据(\*)]《短包最大长度180，长包最大长度1400》 | [流水号uint8 (1)]  《和写命令的流水号一样》  [文件名长度(1)]  [文件名(\*)]《最大长度64》  [文件偏移LSB(4)]  [数据长度LSB(4)]  [操作结果(1)]《参考文件操作状态》 |
| 特定文件读操作  注：单个文件夹或文件名称长度不能超过8字节，当文件名包含完整路径时最大长度不超过64字节，路径采用'\\'分割  VES>=4000 | 71H | \* | [流水号uint8 (1)]  《需要递增，保证每次都不一样》  [文件名长度(1)]  [文件名(\*)]  [文件偏移LSB(4)]  [读取数据长度LSB(4)]  《短包最大长度180，长包最大长度1400》 | [流水号 uint8 (1)]  《和请求数据的流水号一样》  [文件名长度(1)]  [文件名(\*)]  [文件偏移LSB(4)]  [数据长度LSB(4)]  [操作结果(1)]《参考文件操作状态》  [数据(\*)] |
| 获取特定目录下的文件，不包含文件夹  注：单个文件夹或文件名称长度不能超过8字节，当文件名包含完整路径时最大长度不超过64字节，路径采用'\\'分割  VES>=4000  VES（3000-3999）只支持短包 | 72H | \* | [路径名称(\*)] | [路径长度(1)]  [路径名称(\*)]  [文件大小(4)LSB]  [文件名称(\*)] |
| 删除指定文件  VES>=3000 | 73H | \* | [文件名(\*)]  《文件名可包含路径，最大长度64》 | [文件名(\*)]  [操作结果(1)] |
| 创建文件夹 |  |  |  |  |
| 删除文件夹 |  |  |  |  |
| 扫描文件夹，返回文件夹中的文件和文件夹 |  |  |  |  |
|  |  |  |  |  |
| 竞赛状态标志锁，锁住后值无法更改 | B0H |  | * 查询（不包含参数） * 设置   [锁]（1字节）  《允许写：锁=0，不允许写：锁=1，写一次：锁=2》 | * 查询   《参考设置》   * 设置（不返回） |
| 竞赛状态标志 | B1H |  | * 查询（不包含参数） * 设置（1个字节参数） | * 查询（返回1个字节） * 设置（不返回） |
|  |  |  |  |  |
|  |  |  |  |  |

* + - 1. 文件操作状态枚举说明

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **错误描述** | **错误代码** | **值** | **文件写**  **0x70** | **文件读**  **0x71** | **扫描**  **0x72** | **删除**  **0x73** |
| 操作成功 | FILEEDIT\_STATE\_OK | 0 | ✔ | ✔ |  | ✔ |
| 内存分配错误 | FILEEDIT\_STATE\_MEMORY\_FAIL | 1 | ✔ | ✔ |  | ✔ |
| 打开错误 | FILEEDIT\_STATE\_OPEN\_FAIL | 2 |  |  |  |  |
| 创建错误 | FILEEDIT\_STATE\_CREAT\_FAIL | 3 |  |  |  |  |
| 写错误 | FILEEDIT\_STATE\_WRITE\_FAIL | 4 | ✔ |  |  |  |
| 读错误 | FILEEDIT\_STATE\_READ\_FAIL | 5 |  | ✔ |  |  |
| 没有发现文件 | FILEEDIT\_STATE\_NO\_FILE | 6 | ✔ | ✔ |  | ✔ |
| 文件错误 | FILEEDIT\_STATE\_FILE\_FAIL | 7 |  |  |  |  |
| 没有发现路径 | FILEEDIT\_STATE\_NO\_PATH | 8 | ✔ | ✔ |  | ✔ |
| 路径错误 | FILEEDIT\_STATE\_PATH\_FAIL | 9 |  |  |  |  |
| 删除错误 | FILEEDIT\_STATE\_DELETE\_FAIL | 10 |  |  |  |  |
| 其它错误 | FILEEDIT\_STATE\_OTHER\_FAIL | 11 | ✔ | ✔ |  |  |
| 参数错误 | FILEEDIT\_STATE\_PARAM\_FAIL | 12 | ✔ | ✔ |  | ✔ |
| 文件读完成（读到文件末尾） | FILEEDIT\_STATE\_READ\_FINISH |  |  | ✔ |  |  |

* + - 1. 武器属性数据结构

class WeaponProperties

{

武器编号 (uint8)

性别 (uint8)

生命值 (uint8)

轻攻击力 (uint8)

重攻击力 (uint8)

移动速度 (uint8)

攻击速度 (uint8)

防御 (uint8)

冷却时间 (uint8)

左手耐久 (uint8)

右手耐久 (uint8)

};

* + 1. (设备类型：0x0A)授权命令

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 设备类型 | 命令 | 长度 | 发送数据包 | 响应数据包 | 描述 |
| 0x0A | 71H | 0-20 | H1 H2 00H 00H D0+3 0AH 71H D0 D1-Dn CHECK  D0:密码长度（0-20）  D1-Dn：密码ASC取值 | H1 H2 00H 00H 04H 0AH 71H 01H D0 CHECK  D0=0：密码错误  D0=1：密码正确，验证通过 | 通过发送正确的密码同机器人建立连接，如果没有密码数据长度为0即可，数据长度即密码长度 |
| 0x0A | 72H | 01H | H1 H2 00H 00H 03H 0AH 72H 01H D0 CHECK  D0=0:查询验证状态  D0=1：清除验证状态 | 返回1个字节  D0=0： 未验证  D0=1：已验证 | 查询验证信息（仅授权后命令有效） |
| 0x0A | 73H | 00H |  | 响应大于等于8字节  0-5：MAC地址  6：机器人武器编号  7：是否具有密码，0无密码，否则表示有密码  8-XX：名称实体，编码方式UTF8 | 搜索机器人 |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  | 0x89 | 00H |  | 响应数据长度6字节MAC地址 | 查询机器人MAC地址 |

注：

CHECK：命令列表中的“CHECK”表示是校验结果

* + 1. (设备类型：0x10)局域网通信（老版本APP使用，往后不再支持）

UDP 端口9998

HEAD1（1） HEAD2（1） cmd1（1字节） cmd2（1字节） length（2字节） data check

HEAD1=0xFF

HEAD1=0xEE

长度高字节在前

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Cmd1 | Cmd2 | 长度 | 发送数据包 | 响应描述 | 描述 |
| 0x10 | 01H | XXXXH |  |  | 房间定时广播数据，表示等待加入，长度不定，如果房间的生命周期为0，表示要销毁该房间。 |
| 02H | 01H | D0=0表示保持房间加入状态（房间内的成员之间如果长时间未通信自动退出）  D0=1表示退出当前房间 |  | 退出房间 |
| 03H | XXXXH | RoomBaseMsg |  | 请求加入房间 |
| 04H |  |  |  | 返回-1加入失败  否则加入成功 |
|  |  |  |  |  |
| 06H |  |  |  | 同步RoomMenber数据 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
| 10H |  |  |  | 设置各自机器人进入战斗模式 |
| 11H |  |  |  | 准备完成，进入战斗模式 |
|  |  |  |  |  |
| 20H |  |  | 数据包首先应该获取Key=class，值表明当前数据属于哪一个类 | Json格式通信，状态数据设置 |
| 21H |  |  |  | Json格式通信，状态数据设置响应 |
| 11H | CMD2 |  |  |  | 对当前命令包的响应 |
|  |  |  |  |  |  |

* + 1. 新动作文件协议

动作文件后缀.mfo

文件文本格式UTF-8

例子：

VES23,R2

M0,V123.wav;456.wav

D300,P1\_p2180\_t100,P2\_p2200\_t100,P3\_p600\_t100,

D1000,B\_s0\_a0\_t1000\_d900,P1\_p2180\_t100,P2\_p2200\_t100,

* 主引导字符

用ASC大写字母表示，如果带有参数则参数直接跟在主引导符之后，主引导字符中有一种特殊的字符成为独立主引导字符，独立主引导字符必须出现在所有引导字符之前（行首），且不同的独立主引导字符采用回车换行分割。

多个主引导字符使用“;”分号分割

* 子引导字符

用ASC小写字母表示，子引导字符必须在主引导字符之后采用“\_”下划线分割，不同的主引导字符可以不带或带有多个子引导字符，子引导符之间也采用英文下划线“\_”分割；子引导字符必须带有参数，且参数紧跟在子引导字符之后。不同的主引导字符之间必须使用英文字符“,”逗号隔开。

备注：

所有的名称和参数里面都不能出现分隔符（“,” “\_”）

第一行为文件头，包含动作文件版本信息VES必须在第一行的开始，日期等信息等（年4位，月两位，日两位），角色属性R，

VES45,DATE20180506,R0

第二行开始记录动作信息引导字符含义查看下表：

|  |  |  |
| --- | --- | --- |
| **含义** | **引导字符** | **备注** |
| VES为版本引导字符，x为版本号参数，必须出现在文件开始，否则不能认为是正常的动作文件  后面可以出现的引导字符有：  Rx（保留） | VESx | 独立主引导 |
| 指定动作，x为动作名称，长度不能超过63个字节,必须出现在行起始  后面可以出现的引导字符有：  V（语音参数） | Mx | 独立主引导 |
| 当前行执行动作从开始到执行下一行动作的时间，必须出现在行起始，x为具体时间的ASC码，单位毫秒  后面可以出现的引导字符有：  B（底盘参数）  V（语音参数）  Px（动作点位参数） | Dx | 独立主引导 |
|  |  |  |
| 舵机位置引导，x表示id  Px后面必须有子引导符参数（p,s,t） | Px | 主引导 |
| 参数，位置 | p | 子引导 |
| 参数，速度 | s | 子引导 |
| 参数，时间 | t | 子引导 |
|  |  |  |
| 语音引导字符  例如：V234.WAV(后面紧跟语音参数，234.WAV为语音文件名，可以包含路径，如：V0254\234.WAV，多个语音采用英文分号” ; ”分割，如V0000\0013.WAV;0000\0014.WAV;0000\0015.WAV) | V | 主引导 |
|  |  |  |
| 底盘执行参数引导  B后面必须有子引导符参数（s,a,t,d） | B | 主引导 |
| 底盘子参数，速度int16（0~100） | s | 子引导 |
| 底盘子参数，角度int16（0~360） | a | 子引导 |
| 底盘子参数，转弯速度,int16（-1000~+1000） | t | 子引导 |
| 底盘子参数，执行时间Uint16 | d | 子引导 |

1. Bootloader描述

Bootloader用于对程序进行更新，上电前按住功能按键，即可进入bootloader模式，长按3-5秒即可让bootloader进入USB优盘模式，可以对TF卡中的文件进行读写。

* 1. USB优盘功能

bootloader支持USB优盘功能，该功能可以方便用户对TF卡的读写操作。进入方法如下：

在bootloader模式下长按功能按键，等待机身指示灯呈蓝色闪烁时松开（大约2-4秒）。然后使用typeC数据线连接机器人和电脑即可对机器人内部的TF卡进行文件拷贝或写入操作。

* 1. 刷新程序文件

刷新程序文件目前仅仅支持通过USB离线刷新的方式

* + 1. 第一步，准备程序文件

可以将程序文件命名为appXXX.bin形式（xxx可以为版本号），同时将文件放置在software文件夹下，software文件夹下不能放置其它文件，否则不能刷新程序。

* + 1. 第二步，进入刷新程序流程

长按功能按键，当机身指示灯呈红色快速闪烁状态时立即松开（大约6-8秒），即可进入刷新程序流程。

* + 1. 第三步，等待刷新完成

程序刷新过程中，机身指示灯呈绿色闪烁状态，如果刷新失败，机身指示灯呈红色，如果刷新成功机器人将自动重启，运行新的程序

1. 武器说明

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 属性编号 | 名称 | 动作文件名 | 生命值 | 攻击力 | 重攻击 | 移动速度 | 攻击速度 | 防御 | 冷却 | 手臂生命值 |  |
| 缺省 | 无武器 | 无 | 60 | 10 | 20 | 80 | 保留 | 保留 | 保留 | 30 |  |
| 1 | 剑 | 剑 | 90 | 10 | 20 | 50 | 保留 | 保留 | 保留 | 45 |  |
| 2 | 锤 | 锤 | 100 | 10 | 20 | 20 | 保留 | 保留 | 保留 | 50 |  |
| 3 | 刀 | 刀 | 80 | 10 | 20 | 70 | 保留 | 保留 | 保留 | 40 |  |

动作编辑器：2.0.16 对应 程序版本："4.1.7"或5.1.7（4017，5017）

1. 战斗模式特效
2. 头部霍尔重击，操作反向5秒
3. 腰部霍尔重击，底盘不能移动5秒

注：特效时间不叠加，按最近一次触发计时为准。后产生的特效会覆盖先产生的特效

轻击：被攻击位置闪黄色

重击：被攻击位置闪红色

胸前LED指示

根据队伍属性显示不同的主题颜色

|  |  |  |  |
| --- | --- | --- | --- |
| 组号 | 颜色 |  |  |
| 1 | 红 |  |  |
| 2 | 蓝 |  |  |
| 3 | 绿 |  |  |
| 4 | 黄 |  |  |
| 5 | 紫 |  |  |
| 6 | 白 |  |  |
|  |  |  |  |

1. LED状态说明

* 空闲模式：

显示wifi状态和手臂安装状态

左右LED指示手臂安装状态（安装：蓝色，脱落：红色，加紧：绿色）

AP模式：上下LED指示WIFI连接状态（有设备：绿色常量，无设备：绿色闪烁）

STATION模式：

连接路由中：上下两个灯，蓝色灯闪烁

连接成功：上下两个灯两个灯保蓝色持常亮

连接错误（密码错误或者路由不存在）：上下两个灯黄色闪烁

* 战斗模式：

由当前角色在房间中的序号显示不同的颜色，同时亮度表示当前血量，血量越少，灯越暗。

收到攻击后LED状态参考： 9战斗模式特效

* 失败模式（机器人血量为0）：

机身所有灯呈红色渐暗，循环

1. 音效规范
   1. 音效说明

硬件要求：TF卡推荐1GB，使用前推荐格式为FAT，16KB模式（其它容量的TF卡没有测试，其它的文件系统格式没有测试）。

语音部分：语音播放格式为WAV，文件夹和文件名称命名形式为4位数字的方式如（0001.WAV）。语音文件统一处理为采样率44K，8bit，单声道

不同武器的音频文件，根据武器的ID放置与对应名称的文件夹下（文件夹名称长度为4位），比如ID为1的武器，音频文件放置在0001文件夹下。

注：新版本采样率定为22K，采用flash代替SD卡，短文件名系统

* 1. 音效编号定义
     1. 技能音效

编号从0001开始，每个技能预留3个声音， 共5个技能，但是只支持技能1-3和胜利和失败有声音。

|  |  |  |  |
| --- | --- | --- | --- |
| **编号范围** | **功能** | **默认使用** | **说明** |
|  | 上电初始动作 |  | 无声音 |
|  |  |  |  |
|  |  |  |  |
|  | 失败动作 |  |  |
|  |  |  |  |
|  |  |  |  |
|  | 胜利动作 |  |  |
|  |  |  |  |
|  |  |  |  |
|  | 进入战斗模式 |  |  |
|  |  |  |  |
|  |  |  |  |
|  | 技能1音效 |  |  |
|  | 技能1音效 |  |  |
|  | 技能1音效 |  |  |
|  | 技能2音效 |  |  |
|  | 技能2音效 |  |  |
|  | 技能2音效 |  |  |
|  | 技能3音效 |  |  |
|  | 技能3音效 |  |  |
|  | 技能3音效 |  |  |
|  | 技能4音效 |  |  |
|  | 技能4音效 |  |  |
|  | 技能4音效 |  |  |
|  | 技能5音效 |  |  |
|  | 技能5音效 |  |  |
|  | 技能5音效 |  |  |
|  | 技能6音效 | 保留 | 保留 |
|  | 技能6音效 | 保留 | 保留 |
|  | 技能6音效 | 保留 | 保留 |
|  | 技能7音效 | 保留 | 保留 |
|  | 技能7音效 | 保留 | 保留 |
|  | 技能7音效 | 保留 | 保留 |
|  | 技能8音效 | 保留 | 保留 |
|  | 技能8音效 | 保留 | 保留 |
|  | 技能8音效 | 保留 | 保留 |
|  | 技能9音效 | 保留 | 保留 |
|  | 技能9音效 | 保留 | 保留 |
|  | 技能9音效 | 保留 | 保留 |
|  |  | 保留 | 保留 |
|  |  | 保留 | 保留 |
|  |  | 保留 | 保留 |
|  |  | 保留 | 保留 |
|  |  | 保留 | 保留 |
|  |  | 保留 | 保留 |
|  |  | 保留 | 保留 |
|  |  | 保留 | 保留 |
|  |  | 保留 | 保留 |
|  |  | 保留 | 保留 |
|  |  | 保留 | 保留 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

* + 1. 其它音效

编号从0080开始，每个动作预留3个声音，目前手臂掉落无声音。

|  |  |  |  |
| --- | --- | --- | --- |
|  | 左手臂掉落声音 |  |  |
|  |  |  |  |
|  |  |  |  |
|  | 右手臂掉落声音 |  |  |
|  |  |  |  |
|  |  |  |  |
|  | 左手安装武器声音 |  |  |
|  |  |  |  |
|  |  |  |  |
|  | 右手安装武器声音 |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

* + 1. 被攻击音效

编号从0110开始，每个动作预留3个声音，目前所有轻攻击音效一样，所有重攻击音效一样,默认使用身体轻攻击和身体重攻击。

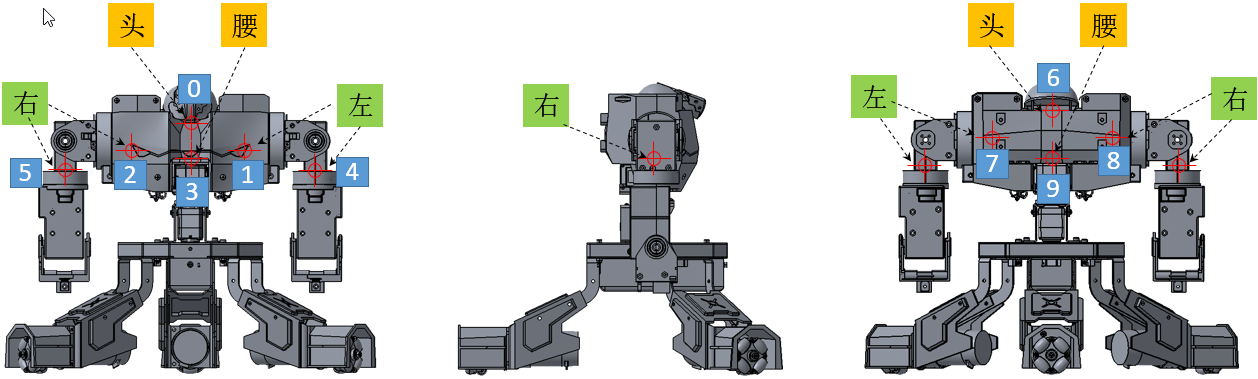
|  |  |  |  |
| --- | --- | --- | --- |
|  | 左手臂被攻击-轻 |  |  |
|  |  |  |  |
|  |  |  |  |
|  | 左手臂被攻击-重 |  |  |
|  |  |  |  |
|  |  |  |  |
|  | 右手臂被攻击-轻 |  |  |
|  |  |  |  |
|  |  |  |  |
|  | 右手臂被攻击-重 |  |  |
|  |  |  |  |
|  |  |  |  |
|  | 身体被攻击-轻 |  |  |
|  |  |  |  |
|  |  |  |  |
|  | 身体被攻击-重 |  |  |
|  |  |  |  |
|  |  |  |  |

1. 机身伤害点位编号

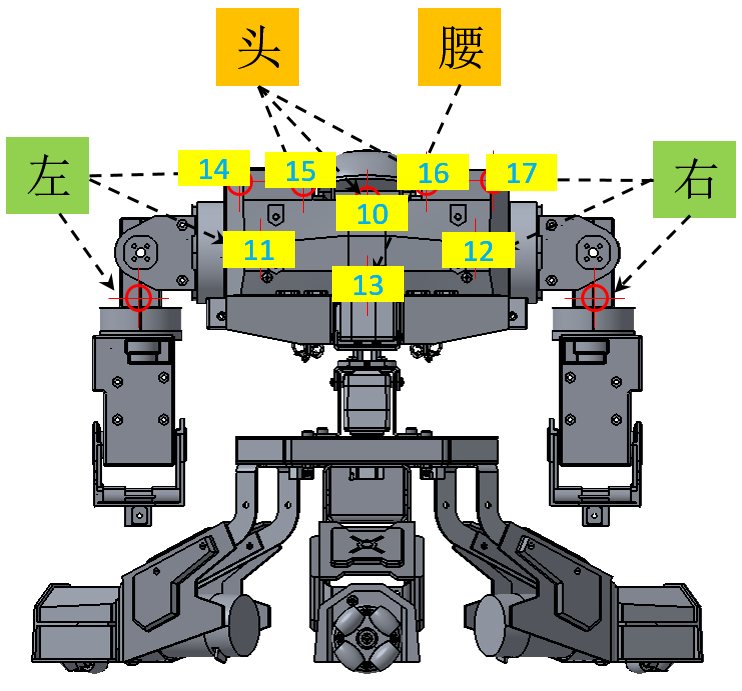
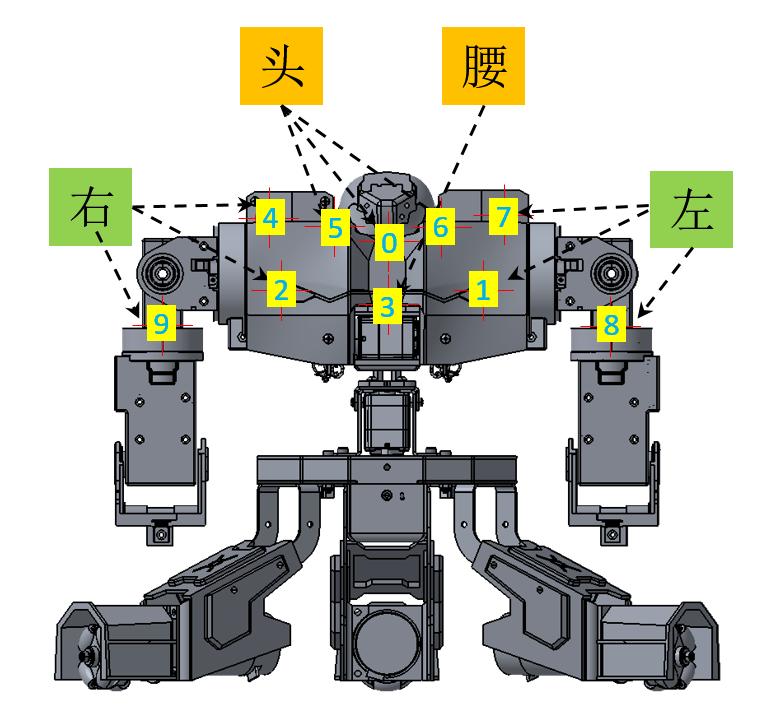
编号原则按照机器人站立姿态，以机器人机身作为0点坐标参考，从上到下从左到右的原则依次递增，其中4号和5号为手臂伤害计算点

下图示例是以人的角度看到的机器人伤害点位编号：

* 1. 老版本点位：



* 1. 新版本点位：



1. 程序版本对应

|  |  |  |
| --- | --- | --- |
| **软件版本** | **硬件版本** | **描述** |
| V1.0.1 | MControl-003 |  |
|  |  |  |

1. 错误编号值

typedef enum {

DEVICE\_NONE=0,

DEVICE\_ERR\_ID40=1,

DEVICE\_ERR\_ID60,

DEVICE\_ERR\_ID70,

DEVICE\_COUNT\_ID40,

DEVICE\_COUNT\_ID60,

DEVICE\_COUNT\_ID70,

}DeviceErrorflag;