板卡通信接口协议

*版本号：V1.5*

*2020年8月*

目录

[目录 1](#_Toc14311)

[1 概述 3](#_Toc5761)

[1.1 修改记录 3](#_Toc8102)

[1.2 名词及缩写解释 4](#_Toc1168)

[2 通信协议 4](#_Toc18228)

[2.1 协议格式 4](#_Toc5073)

[2.2 通信方式 5](#_Toc30286)

[2.3 帧命令(CMD)说明 5](#_Toc9222)

[2.4 指令格式 6](#_Toc20674)

[2.5 通信命令分配表 6](#_Toc18577)

[3 系统命令 7](#_Toc18443)

[3.1 系统模块(0x0001, 0x01ff) 7](#_Toc12828)

[3.1.1 格式化存储单元 7](#_Toc8922)

[3.1.2 调试信息控制 8](#_Toc19067)

[3.1.3 获取堆栈使用情况 8](#_Toc4602)

[3.1.4 设备重启 9](#_Toc15497)

[3.1.5 设置本机 IPV4 信息 10](#_Toc17705)

[3.1.6 获取本机相关信息 11](#_Toc12309)

[3.1.7 系统进入或退出紧急停止状态 12](#_Toc18395)

[3.1.8 设置系统时间 13](#_Toc541)

[3.1.9 配置阿里云相关信息 13](#_Toc27912)

[3.1.10 获取阿里云相关信息 14](#_Toc17378)

[3.1.11 调试信息输出 15](#_Toc2674)

[3.1.12 测试通信 15](#_Toc24476)

[3.1.13 获取SN号 16](#_Toc1793)

[3.1.14 设置 SN号 16](#_Toc30681)

[3.1.15 获取软件版本信息 17](#_Toc7468)

[3.1.16 获取设备信息 17](#_Toc32460)

[3.1.17 设置 LED 状态 18](#_Toc31118)

[3.1.18 设置要监控的事件 19](#_Toc11542)

[3.1.19 更新固件请求 19](#_Toc24326)

[3.1.20 更新固件内容命令 20](#_Toc31693)

[3.1.21 更新固件完成 21](#_Toc1315)

[3.1.22 上位机信息通过 CAN 转发 21](#_Toc9168)

[3.1.23 设置设备相关信息 22](#_Toc19115)

[3.1.24 获取设备连接上位机监控状态 23](#_Toc29865)

[3.2 电机模块(0x0200, 0x02ff) 23](#_Toc25967)

[3.2.1 步进电机启动 23](#_Toc10123)

[3.2.2 步进电机停止 24](#_Toc16860)

[3.2.3 查询步进电机状态 25](#_Toc14124)

[3.2.4 直流电机启动 25](#_Toc6112)

[3.2.5 直流电机停止 26](#_Toc19342)

[3.2.6 查询直流电机状态 27](#_Toc10261)

[3.3 输入输出模块(0x0300, 0x03ff) 28](#_Toc246)

[3.3.1 DI状态获取 28](#_Toc27493)

[3.3.2 FDI状态获取 28](#_Toc19352)

[3.3.3 DO状态获取 29](#_Toc4206)

[3.3.4 DO 设置 30](#_Toc1486)

[3.3.5 设置 FDI 工作模式 30](#_Toc15735)

[3.3.6 获取FDI作为普通DI的状态 31](#_Toc25807)

[3.3.7 获取FDI作为高速输入的状态 32](#_Toc703)

[3.3.8 设置FDO作为普通DO 时的状态 32](#_Toc12345)

[3.3.9 获取 FDO作为普通DO的状态 33](#_Toc2138)

[3.3.10 启动 FDO做为高速脉冲输出 34](#_Toc15097)

[3.3.11 停止 FDO 高速脉冲输出 34](#_Toc7175)

[3.3.12 获取 FDO作为高速脉冲输出时的状态 35](#_Toc20555)

[3.3.13 启动 FDO 作为 PWM输出 36](#_Toc9888)

[3.3.14 停止FDO 的 PWM 输出 36](#_Toc7662)

[3.3.15 获取 FDO 作为 PWM 输出的状态 37](#_Toc3599)

[3.3.16 获取 ADC 的电压值 37](#_Toc11454)

[3.3.17 设置 DAC输出值 38](#_Toc28656)

[3.4 文件模块(0x0400, 0x04ff) 39](#_Toc17229)

[3.4.1 打开文件 39](#_Toc16970)

[3.4.2 读文件 39](#_Toc10817)

[3.4.3 写文件 40](#_Toc4869)

[3.4.4 关闭文件 41](#_Toc18543)

[3.4.5 删除文件 41](#_Toc9216)

[3.4.6 获取文件大小 42](#_Toc32018)

[3.4.7 获取文件列表 43](#_Toc20729)

[4 用户命令 (0xA000, 0xC000) 43](#_Toc4249)

[5 监控事件ID定义 43](#_Toc21931)

[6 返回码参数表 44](#_Toc6351)

[7 CRC校验算法示例 49](#_Toc32196)

# 概述

## 修改记录

|  |  |  |  |
| --- | --- | --- | --- |
| 版本 | 日期 | 修改人 | 说明 |
| 1.0 | 2019-5-29 | 屈孟 | 板卡和上位机通信协议初稿 |
| 1.1 | 2019-5-29 | 屈孟 | 1. 去掉命令 0x0000; 2. 修改文件读写,增加读写偏移量; 3. 最大每帧收发大小改为1500bytes. |
| 1.2 | 2019-6-13 | 屈孟 | 1. 增加直流和步进电机的状态查询命令, 相关命令码修改; 2. 增加菜单相关命令; 3. 增加查询锅的温度命令; |
| 1.3 | 2019-11-8 | 张贻众 | 1.增加冰淇淋机本体通讯功能  2.包含功能控制、定量出料时间、出料成型比例、允许出料个数。 |
| 1.4 | 2019-12-20 | 屈孟 | 重新调整命令码的结构, 同时调整命令布局, 适应多应用方便开发 |
| 1.5 | 2020-8-20 | 屈孟 | 修改部分描述错误, 包括数据最大长度, crc校验数据内容 |

## 名词及缩写解释

|  |  |
| --- | --- |
| 缩写 | 内容 |
| 下位机 | MS-TWO 或其它 |
| 上位机 | PC/手机/其它 |
| SOF | Start of frame， 数据通信起始符 |
| FRLEN | Frame length,通信帧长度 |
| DATA | Data， 通信交互数据 |
| RC | Return code, 返回码 |
| CRC | CRC 校验 |
| CRECTP | Current Encryption type, 当前数据加密类型 |

# 通信协议

## 协议格式

本文档所描述的数据均采用大端模式， 数据高字节保存在内存低地址，数据低字节保存在内存高地址。

请求报文为主动发起的命令，需要得到应答报文回复。

上下位机通信的单次data 数据最大长度为 500 (500 - 9)bytes, 单独指 DATA段， 不包含其它信息数据, 其它数据段最大 9 个字节.

上位机收发格式：

请求

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 数据项 | 字节 | 含义 | 格式 | 备注 |
| SOF | 1 | 数据包包头标识符 | HEX | 固定填0xAA |
| CMD | 2 | 命令码 | HEX | 指明当前通信报文的类型与功能 |
| FRLEN | 2 | 数据包内容长度 | HEX |  |
| DATA | N | 数据内容 | HEX | 具体的数据 |
| CRC | 2 | CRC16校验码 | HEX | 除去包头的其它所有数据进行CRC16校验运算 |

应答报文

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 数据项 | 字节 | 含义 | 格式 | 备注 |
| SOF | 1 | 数据包包头标识符 | HEX | 固定填0xAA |
| CMD | 2 | 命令码 | HEX | 指明当前通信报文的类型与功能 |
| RC | 2 | 返回码 | HEX | 返回响应状态 |
| FRLEN | 2 | 数据包内容长度 | HEX |  |
| DATA | N | 数据内容 | HEX | 具体的数据 |
| CRC | 2 | CRC16校验码 | HEX | 除去包头的其它所有数据进行CRC16校验运算 |

说明：

SOF：起始域，1字节，值固定为0XAA；

CMD : 2 字节,指明当前通信报文的类型与功能;

RC : 2 字节， 指明返回帧状态， 参见 [返回码状态表](#_返回码参数表);

FRLEN：长度域，2字节，表示DATA数据的长度（仅为 DATA长度）；

DATA ：数据域，N字节，传输的有效数据

CRC：校验域，2字节，包括 SOF+CMD+RC+FRLEN+DATA 在内的所有数据进行CRC16校验运算, 参见 [CRC算法示例](#_CRC校验算法示例);

## 通信方式

采用同步和异步通讯方式。与上位机通过网络/串口/usb/串口等进行通信,具体待定。

上下位机通过命令类型判断通讯报文是否需要回复, 具体参考[帧命令说明](#_帧命令说明).

## 帧命令(CMD)说明

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 命令说明 | 命令值 | | | 12...0 |
| 15 | 14 | 13 |
| 请求命令, 上位机发送给下位机并且等待回复 | 0 | 0 | 0 | 0x0000 -- 0x1fff |
| 广播命令 | 0 | 0 | 1 | 0x0000 -- 0x1fff |
|  | 0 | 1 | 0 | 0x0000 -- 0x1fff |
|  | 0 | 1 | 1 | 0x0000 -- 0x1fff |
|  | 1 | 0 | 0 | 0x0000 -- 0x1fff |
| 用户命令区间, 由开发应用的相关人员定义 | 1 | 0 | 1 | 0x0000 -- 0x1fff |
|  | 1 | 1 | 0 | 0x0000 -- 0x1fff |
|  | 1 | 1 | 1 | 0x0000 -- 0x1fff |

## 指令格式

请求或通知报文:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 说明 | 帧头 | 帧命令 | 数据长度 | 数据 | 校验 |
| 长度 | 1Byte | 2Byte | 2Byte | (N)Byte | 2Byte |
| 数据 | 0xAA | CMD | FRLEN | DATA | CRC |

响应报文:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 说明 | 帧头 | 帧命令 | 返回码 | 数据长度 | 数据 | 校验 |
| 长度 | 1Byte | 2Byte | 2Byte | 2Byte | (N)Byte | 2Byte |
| 数据 | 0xAA | CMD | RC | FRLEN | DATA | CRC |

注意：

报文中数据长度的取值范围为 0 到 65535， 如果请求报文或者响应报文中没有数据，则数据长度的值为0 。 即当数据长度为0时，该报文中没有数据位， N 等于0。

例：测试通信

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 说明 | 帧头 | 帧命令 | 数据长度 | 数据 | 校验 |
| 长度 | 1Byte | 2Byte | 2Byte | 5Byte | 2Byte |
| 数据 | 0xAA | 0x0100 | 0x0005 | 0x01, 0x23, 0x45, 0x67, 0x89 | 0x0442 |

发送的数据为字符流(HEX)：AA010000050123456789C2C1

该指令格式为完整的数据传输格式， 所有数据必须以以上格式传输，否则为无效数据。

所有命令不可重复使用。

## 通信命令分配表

|  |  |  |
| --- | --- | --- |
| 命令模块 | 命令码(CMD) | 说明 |
| 预留命令  0x0000 |  | 暂时保留 |
| 系统模块  (0x0001, 0x00ff) |  | 用于系统控制的指令，内部使用，不对外开放 |
| 通用模块  (0x0100, 0x01ff) |  | 通用指令 |
| 锅组模块  (0x0200, 0x02ff) |  | 用于锅组模块的功能指令 |
| 迷你冰淇淋机模块  （0x0300，0x03ff） |  | 用于迷你冰淇淋机模块的功能指令 |
| 文件模块  (0x1f00, 0x1fff) |  | 用于文件操作 |

通信命令分配查看[通信命令分配表](#_通信命令分配表)， 需要修改，增加或删除命令请同步更新[通信命令分配表](#_通信命令分配表)，同时增加修改记录。命令按模块区分，每个模块分配256个命令码。

# 系统命令

## 系统模块(0x0001, 0x01ff)

说明：系统模块为内部使用部分，不对外提供接口.

### 格式化存储单元

说明：

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0001 |
| 数据长度 | 2 | 1 |
| 数据内容 | 1 | 0:系统存储区;  1:用户存储区; |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0001 |
| 返回码 | 2 | RC |
| 数据长度 | 2 | 0 |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 调试信息控制

说明： 控制调试信息

请求：(上位机)

|  |  |  |  |
| --- | --- | --- | --- |
| 域 | 字节 | | 备注 |
| 包头 | 1 | | 0xAA |
| 包命令 | 2 | | 0x0002 |
| 数据长度 | 2 | | N |
| 数据内容 | N | 1 | 串口1打印, 0 不打印, !0 打印 |
| 1 | 串口2打印, 0 不打印, !0 打印 |
| 1 | 串口3打印, 0 不打印, !0 打印 |
| 1 | 网络TCP监控设备打印, 0 不打印, !0 打印 |
| 1 | 网络UDP监控设备打印, 0 不打印, !0 打印 |
| 1 | 日志文件记录, 0不记录, !0记录 |
| CRC校验 | 2 | | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0002 |
| 返回码 | 2 | RC |
| 数据长度 | 2 | 0 |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 获取堆栈使用情况

说明： 获取堆栈使用情况

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0003 |
| 数据长度 | 2 |  |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |  |
| --- | --- | --- | --- |
| 域 | 字节 | | 备注 |
| 包头 | 1 | | 0xAA |
| 包命令 | 2 | | 0x0003 |
| 返回码 | 2 | | RC |
| 数据长度 | 2 | | N |
| 数据内容 | N | 4 | 4字节总的栈大小, 单位 bytes |
| 4 | 4字节剩余栈大小, 单位bytes |
| CRC校验 | 2 | | 除去包头的其它所有数据进行CRC16校验运算。 |

### 设备重启

说明： 重启设备

请求：(上位机)

|  |  |  |  |
| --- | --- | --- | --- |
| 域 | 字节 | | 备注 |
| 包头 | 1 | | 0xAA |
| 包命令 | 2 | | 0x0004 |
| 数据长度 | 2 | | 重启参数长度 |
| 数据内容 | 0 | 1 | 重启原因:   1. 设备故障; 2. 更新固件; |
| 32 | 加密数据 |
| CRC校验 | 2 | | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |  |
| --- | --- | --- | --- |
| 域 | 字节 | | 备注 |
| 包头 | 1 | | 0xAA |
| 包命令 | 2 | | 0x0004 |
| 返回码 | 2 | | RC |
| 数据长度 | 2 | | N |
| 数据内容 | N | 4 | 4字节总的栈大小, 单位 bytes |
| 4 | 4字节剩余栈大小, 单位bytes |
| CRC校验 | 2 | | 除去包头的其它所有数据进行CRC16校验运算。 |

### 设置本机 IPV4 信息

说明： 设置本机 IPV4 相关信息

请求：(上位机)

|  |  |  |  |
| --- | --- | --- | --- |
| 域 | 字节 | | 备注 |
| 包头 | 1 | | 0xAA |
| 包命令 | 2 | | 0x0005 |
| 数据长度 | 2 | |  |
| 数据内容 | N | a | 要设置的设备的序列号, 通过 [获取 SN 号](#_获取SN号)获得, 结尾必须以字符串结尾符 0x00 结尾 |
| b | IP内容, 字符串, 以0x00 结尾:  “xxx.xxx.xxx.xxx” |
| c | Netmask 内容, 字符串, 以0x00 结尾:  “xxx.xxx.xxx.xxx” |
| d | Gateway内容, 字符串, 以0x00 结尾:  “xxx.xxx.xxx.xxx” |
| e | 端口号, 字符串, 以0x00 结尾:  “xxxxx” |
| f | DNS内容, 字符串, 以0x00 结尾:  “xxx.xxx.xxx.xxx” |
| CRC校验 | 2 | | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0005 |
| 返回码 | 2 | RC |
| 数据长度 | 2 | 0 |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 获取本机相关信息

说明： 获取本机 设备SN,IPV4 等相关信息, 可自定义, 以字符串组成

请求：(上位机)

|  |  |  |  |
| --- | --- | --- | --- |
| 域 | 字节 | | 备注 |
| 包头 | 1 | | 0xAA |
| 包命令 | 2 | | 0x0006 |
| 数据长度 | 2 | |  |
| 数据内容 | n | 1 | 过滤 SN 标识  0, 不需要过滤, 收到请求包的所有设备都需要回;  1, 需要过滤, 和本机 SN 对比, 匹配则回复; |
| a | 本机的设备的 SN 号  如果过滤 SN 标识为 1则 SN 存在, 可通过 [获取 SN 号](#_获取SN号)获得, 结尾必须以字符串结尾符 0x00 结尾;  如果过滤 SN 标识为 0, 则该段不存在 |
| 1 | 过滤项目 ID 标识  0, 不需要过滤, 收到请求包的所有设备都需要回;  1, 需要过滤, 和本机项目 ID 对比, 匹配则回复; |
| b | 本机的设备的项目 ID 号, 格式为字符串  如果过滤项目 ID 标识为1, 则该 ID 存在且以 0x00结尾;  如果过滤项目 ID 标识为 0, 则该段不存在 |
| CRC校验 | 2 | | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |  |
| --- | --- | --- | --- |
| 域 | 字节 | | 备注 |
| 包头 | 1 | | 0xAA |
| 包命令 | 2 | | 0x0006 |
| 返回码 | 2 | | RC |
| 数据长度 | 2 | | 0 |
| 数据内容 | N | a | 本机的设备的序列号, 可通过 [获取 SN 号](#_获取SN号)获得, 结尾必须以字符串结尾符 0x00 结尾 |
| b | IP内容, 字符串, 以0x00 结尾:  “xxx.xxx.xxx.xxx” |
| c | Netmask 内容, 字符串, 以0x00 结尾:  “xxx.xxx.xxx.xxx” |
| d | Gateway内容, 字符串, 以0x00 结尾:  “xxx.xxx.xxx.xxx” |
| e | 端口号, 字符串, 以0x00 结尾:  “xxxxx” |
| f | 当前和上位机监控设备连接状态, 字符串, 以0x00 结尾:  “0”未连接, “1”已建立连接 |
| g | 软件版本信息, 字符串, 以0x00 结尾:  “xxxxx” |
| h | DNS内容, 字符串, 以0x00 结尾:  “xxx.xxx.xxx.xxx” |
| i | MAC地址, 字符串, 以0x00 结尾, 显示16进制:  “xxxxxxxxxxxx” |
| j | 项目 ID, 字符串, 以0x00 结尾, 显示16进制:  “xxxxxxxxxxxx” |
| CRC校验 | 2 | | 除去包头的其它所有数据进行CRC16校验运算。 |

### 系统进入或退出紧急停止状态

说明： 让下位机系统进入或者推出紧急急停状态

请求：(上位机)

|  |  |  |  |
| --- | --- | --- | --- |
| 域 | 字节 | | 备注 |
| 包头 | 1 | | 0xAA |
| 包命令 | 2 | | 0x0007 |
| 数据长度 | 2 | |  |
| 数据内容 | 2 | 1 | 1. 退出急停;   1. 进入急停;  其它: 待定; |
| 1 | 原因:   1. 人为按键控制; 2. 待定; |
| CRC校验 | 2 | | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0007 |
| 返回码 | 2 | RC |
| 数据长度 | 2 | 0 |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 设置系统时间

说明： 设置下位机系统时间

请求：(上位机)

|  |  |  |  |
| --- | --- | --- | --- |
| 域 | 字节 | | 备注 |
| 包头 | 1 | | 0xAA |
| 包命令 | 2 | | 0x0008 |
| 数据长度 | 2 | |  |
| 数据内容 | 8 | 2 | 年: 2000 - 2099 |
| 1 | 月: 1 - 12 |
| 1 | 日: 1 - 31 |
| 1 | 时: 0 - 23 |
| 1 | 分: 0 - 59 |
| 1 | 秒: 0 - 59 |
| 1 | 星期几: 1 - 7 |
| CRC校验 | 2 | | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0008 |
| 返回码 | 2 | RC |
| 数据长度 | 2 | 0 |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 配置阿里云相关信息

说明： 配置阿里云的相关信息

请求：(上位机)

|  |  |  |  |
| --- | --- | --- | --- |
| 域 | 字节 | | 备注 |
| 包头 | 1 | | 0xAA |
| 包命令 | 2 | | 0x0009 |
| 数据长度 | 2 | |  |
| 数据内容 | n | a | Device name, 字符串, 以0x00 结尾:  “xxxxxxxxxxxx” |
| b | Device secret, 字符串, 以0x00 结尾:  “xxxxxxxxxxxx” |
| c | Product key, 字符串, 以0x00 结尾:  “xxxxxxxxxxxx” |
| CRC校验 | 2 | | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0009 |
| 返回码 | 2 | RC |
| 数据长度 | 2 | 0 |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 获取阿里云相关信息

说明： 获取阿里云的相关配置信息

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x000a |
| 数据长度 | 2 |  |
| 数据内容 |  |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |  |
| --- | --- | --- | --- |
| 域 | 字节 | | 备注 |
| 包头 | 1 | | 0xAA |
| 包命令 | 2 | | 0x000a |
| 返回码 | 2 | | RC |
| 数据长度 | 2 | | 0 |
| 数据内容 | n | a | Device name, 字符串, 以0x00 结尾:  “xxxxxxxxxxxx” |
| b | Device secret, 字符串, 以0x00 结尾:  “xxxxxxxxxxxx” |
| c | Product key, 字符串, 以0x00 结尾:  “xxxxxxxxxxxx” |
| CRC校验 | 2 | | 除去包头的其它所有数据进行CRC16校验运算。 |

### 调试信息输出

说明： 输出调试信息

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x000B |
| 数据长度 | 2 | N |
| 数据内容 | N | 调试信息输出 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 测试通信

说明：测试通信是否正常

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x000C |
| 数据长度 | 2 | N |
| 数据内容 | N | 随机长度数据 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x000C |
| 返回码 | 2 | RC |
| 数据长度 | 2 | N |
| 数据内容 | N | 接收到的随机数据 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 获取SN号

说明：获取下位机 SN 号

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x000D |
| 数据长度 | 2 | 0 |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x000D |
| 返回码 | 2 | RC |
| 数据长度 | 2 | N |
| 数据内容 | N | SN, string 类型, 以 \x00 结尾 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 设置 SN号

说明：设置下位机 SN 号

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x000E |
| 数据长度 | 2 | N |
| 数据内容 | N | SN |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x000E |
| 返回码 | 2 | RC |
| 数据长度 | 2 | 0 |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 获取软件版本信息

说明：获取下位机软件版本信息

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x000F |
| 数据长度 | 2 | 0 |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |  |
| --- | --- | --- | --- |
| 域 | 字节 | | 备注 |
| 包头 | 1 | | 0xAA |
| 包命令 | 2 | | 0x000F |
| 返回码 | 2 | | RC |
| 数据长度 | 2 | | N |
| 数据内容 | N |  | 固件版本号, string 类型, 以 \x00 结尾 |
|  | Boot版本号, string 类型, 以 \x00 结尾 |
| CRC校验 | 2 | | 除去包头的其它所有数据进行CRC16校验运算。 |

### 获取设备信息

说明：获取下位机设备信息, 包括板子信息编号,芯片信息编号,板子版本编号,应用工程编号

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0010 |
| 数据长度 | 2 | 0 |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0010 |
| 返回码 | 2 | RC |
| 数据长度 | 2 | N |
| 数据内容 | N | 设备信息, 查看[设备信息表](#_设备信息表) |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 设置 LED 状态

说明：设置 LED 状态

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0011 |
| 数据长度 | 2 |  |
| 数据内容 | 1 | LED编号,0,1,2,... |
| 1 | 状态  0:关闭;  1:打开;  2:闪烁; |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0011 |
| 返回码 | 2 | RC |
| 数据长度 | 2 |  |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 设置要监控的事件

说明：设置需要监控的事件, 设置成功之后, 数据会按照设置的频率主动上报

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0012 |
| 数据长度 | 2 | 上报周期和ID列表总长度 |
| 数据内容 | 2 | 上报周期, 单位 1ms |
| x | 要监控的事件的ID号列表, ID号定义规则参考[监控事件ID定义](#_监控模块ID定义) |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0012 |
| 返回码 | 2 | RC |
| 数据长度 | 2 | 不支持的事件ID列表大小, 如果没有则为0 |
| 数据内容 | N | 返回不支持的事件ID列表, 如果没有则不返回.当有不支持的ID返回时需要上位机去除不支持的ID后重新设置. |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 更新固件请求

说明：上位机请求更新固件, 下位机收到可以选择回复, 应答成功进入更新模式.

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0013 |
| 数据长度 | 2 | n |
| 数据内容 | 4 | 固件文件大小 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0013 |
| 返回码 | 2 | RC |
| 数据长度 | 2 | 0 |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 更新固件内容命令

说明：上位机发送固件更新命令.

请求：(上位机)

|  |  |  |  |
| --- | --- | --- | --- |
| 域 | 字节 | | 备注 |
| 包头 | 1 | | 0xAA |
| 包命令 | 2 | | 0x0014 |
| 数据长度 | 2 | | n |
| 数据内容 | X | 4 | 固件内容偏移量 |
| n | 固件内容 |
| CRC校验 | 2 | | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0014 |
| 返回码 | 2 | RC |
| 数据长度 | 2 | 0 |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 更新固件完成

说明：上位机请通知更新完成.

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0015 |
| 数据长度 | 2 | 0 |
| 数据内容 |  |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0015 |
| 返回码 | 2 | RC |
| 数据长度 | 2 | 0 |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 上位机信息通过 CAN 转发

说明：上位机要求通过 CAN 转发信息.

请求：(上位机)

|  |  |  |  |
| --- | --- | --- | --- |
| 域 | 字节 | | 备注 |
| 包头 | 1 | | 0xAA |
| 包命令 | 2 | | 0x0016 |
| 数据长度 | 2 | | N |
| 数据内容 | x | 4 | CAN ID  最高 2 个 bits 为特殊功能位, 剩下为实际 ID:  00:正常发送, 回复无 ID;  01:回复数据需要带 CAN ID; |
| 2 | 超时时间 , ms |
| n | 封好的转发包(包命令+数据内容) |
| CRC校验 | 2 | | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0016 |
| 返回码 | 2 | RC |
| 数据长度 | 2 | N |
| 数据内容 | 4/0 | CAN\_ID, 根据请求的 CAN\_ID 功能位判断是否加上 CAN\_ID |
| x | 封好的回复包(包命令+返回码+数据内容) |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 设置设备相关信息

说明：上位机设置设备相关信息

请求：(上位机)

|  |  |  |  |
| --- | --- | --- | --- |
| 域 | 字节 | | 备注 |
| 包头 | 1 | | 0xAA |
| 包命令 | 2 | | 0x0017 |
| 数据长度 | 2 | | n |
| 数据内容 | N | 4 | CAN ID |
| 未定义 | 未定义 |
| CRC校验 | 2 | | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0017 |
| 返回码 | 2 | RC |
| 数据长度 | 2 | 0 |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 获取设备连接上位机监控状态

说明：上位机设置设备相关信息

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0018 |
| 数据长度 | 2 | 0 |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0018 |
| 返回码 | 2 | RC |
| 数据长度 | 2 | 1 |
| 数据内容 | 1 | 和自研上位机连接状态:  0x00: 未连接  0x01: 已连接 |
| 1 | 和阿里云连接状态:  0x00: 未连接  0x01: 已连接 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

## 电机模块(0x0200, 0x02ff)

### 步进电机启动

说明：启动步进电机

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0200 |
| 数据长度 | 2 | 0 |
| 数据内容 | 1 | 步进电机编号 |
| 1 | 方向,0为正向, !0 为反向 |
| 4 | 频率 |
| 1 | 占空比 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0200 |
| 返回码 | 2 | RC |
| 数据长度 | 2 |  |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 步进电机停止

说明：停止步进电机

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0201 |
| 数据长度 | 2 | 0 |
| 数据内容 | 1 | 步进电机编号 |
| 1 | 停止类型:  1: 自由停止,即断电让电机自由运动后停止  2: 制动停止, 即刹车让电机停止转动 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0201 |
| 返回码 | 2 | RC |
| 数据长度 | 2 |  |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 查询步进电机状态

说明：查询步进电机状态

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0202 |
| 数据长度 | 2 | 0 |
| 数据内容 |  |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |  |
| --- | --- | --- | --- |
| 域 | 字节 | | 备注 |
| 包头 | 1 | | 0xAA |
| 包命令 | 2 | | 0x0202 |
| 返回码 | 2 | | RC |
| 数据长度 | 2 | | 状态长度 |
| 数据内容 | N | 7 | 1字节编号  1字节方向,0为正向, !0 为反向  4字节频率,若停止转动频率为0  1字节占空比 |
| ... | 同上 |
| CRC校验 | 2 | | 除去包头的其它所有数据进行CRC16校验运算。 |

### 直流电机启动

说明：启动直流电机

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0203 |
| 数据长度 | 2 | 7 |
| 数据内容 | 1 | 直流电机编号 |
| 1 | 方向,0为正向, !0 为反向 |
| 4 | 频率 |
| 1 | 起始占空比 |
| 1 | 目标占空比 |
| 2 | 调整时间, 单位毫秒 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0203 |
| 返回码 | 2 | RC |
| 数据长度 | 2 |  |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 直流电机停止

说明：停止直流电机

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0204 |
| 数据长度 | 2 |  |
| 数据内容 | 1 | 直流电机编号 |
| 1 | 停止类型:  1:自由停止,即断电让电机自由运动后停止  2:制动停止, 即刹车让电机停止转动 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0204 |
| 返回码 | 2 | RC |
| 数据长度 | 2 |  |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 查询直流电机状态

说明：查询步进电机状态

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0205 |
| 数据长度 | 2 | 0 |
| 数据内容 |  |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |  |
| --- | --- | --- | --- |
| 域 | 字节 | | 备注 |
| 包头 | 1 | | 0xAA |
| 包命令 | 2 | | 0x0205 |
| 返回码 | 2 | | RC |
| 数据长度 | 2 | | 状态长度 |
| 数据内容 | N | 7 | 1字节编号  1字节方向,0为正向, !0 为反向  4字节频率,若停止转动频率为0  1字节起始占空比  1字节目标占空比  2字节转换时间,单位毫秒  1字节运行状态,0-stop, 1-start, 2-brake,3-no\_action  1字节警报状态,0-正常, 1-错误 |
| ... | 同上 |
| CRC校验 | 2 | | 除去包头的其它所有数据进行CRC16校验运算。 |

## 输入输出模块(0x0300, 0x03ff)

### DI状态获取

说明：获取DI的高低状态

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0300 |
| 数据长度 | 2 | 0 |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0300 |
| 返回码 | 2 | RC |
| 数据长度 | 2 | 获取到的DI数据长度 |
| 数据内容 | N | 获取到的DI状态, 每byte代表每个DI的高低, 1为高, 0为低 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### FDI状态获取

说明：获取FDI的状态

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0301 |
| 数据长度 | 2 | 0 |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0301 |
| 返回码 | 2 | RC |
| 数据长度 | 2 | 获取到的FDI数据长度 |
| 数据内容 | N | 获取到的FDI状态, 每组FDI状态占5个byte大小,第一个byte为高低状态, 0为低, !0 为高, 紧跟后4bytes为脉冲量 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### DO状态获取

说明：获取DO的高低状态

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0302 |
| 数据长度 | 2 | 0 |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0302 |
| 返回码 | 2 | RC |
| 数据长度 | 2 | 获取到的DO数据长度 |
| 数据内容 | N | 获取到的DO状态, 每byte代表每个DO的高低, 1为高, 0为低 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### DO 设置

说明：设置DO的高低状态

请求：(上位机)

|  |  |  |  |
| --- | --- | --- | --- |
| 域 | 字节 | | 备注 |
| 包头 | 1 | | 0xAA |
| 包命令 | 2 | | 0x0303 |
| 数据长度 | 2 | |  |
| 数据内容 | N | 2 | 第一个字节代表DO号, DO号从0开始  第二个字节代表高低状态, 0为低, 1为高 |
| 2 | 第一个字节代表DO号, DO号从0开始  第二个字节代表高低状态, 0为低, 1为高 |
| 2 | 第一个字节代表DO号, DO号从0开始  第二个字节代表高低状态, 0为低, 1为高 |
| ... | 第一个字节代表DO号, DO号从0开始  第二个字节代表高低状态, 0为低, 1为高 |
| CRC校验 | 2 | | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0303 |
| 返回码 | 2 | RC |
| 数据长度 | 0 |  |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 设置 FDI 工作模式

说明：设置 FDI 的工作模式, 目前支持 HSI 高速输入和 DI 低速输入

请求：(上位机)

|  |  |  |  |
| --- | --- | --- | --- |
| 域 | 字节 | | 备注 |
| 包头 | 1 | | 0xAA |
| 包命令 | 2 | | 0x0304 |
| 数据长度 | 2 | | 设置的参数 |
| 数据内容 | N | 2 | Byte 1: FDI编号, 从0开始;  Byte2:模式, 0为低速输入,1为高速输入 |
| ... | ... |
| CRC校验 | 2 | | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0304 |
| 返回码 | 2 | RC |
| 数据长度 | 2 |  |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 获取FDI作为普通DI的状态

说明：将FDI作为普通DI功能使用, 获取DI的高低状态

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0305 |
| 数据长度 | 2 | 0 |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0305 |
| 返回码 | 2 | RC |
| 数据长度 | 2 | 获取到的DI数据长度 |
| 数据内容 | N | 获取到的DI状态, 每byte代表每个DI的高低, 1为高, 0为低 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 获取FDI作为高速输入的状态

说明：获取FDI作为高速输入的状态

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0306 |
| 数据长度 | 2 | 0 |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0306 |
| 返回码 | 2 | RC |
| 数据长度 | 2 | 获取到的FDI数据长度 |
| 数据内容 | N | 获取到的FDI状态, 每组FDI状态占5个byte大小,第一个byte为控制模式, 接着4个字节为已经产生的脉冲数 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 设置FDO作为普通DO 时的状态

说明：将FDO作为普通DO功能使用, 设置DO的高低状态

请求：(上位机)

|  |  |  |  |
| --- | --- | --- | --- |
| 域 | 字节 | | 备注 |
| 包头 | 1 | | 0xAA |
| 包命令 | 2 | | 0x0307 |
| 数据长度 | 2 | |  |
| 数据内容 | N | 2 | 第一个字节代表DO号, DO号从0开始  第二个字节代表高低状态, 0为低, 1为高 |
| 2 | 第一个字节代表DO号, DO号从0开始  第二个字节代表高低状态, 0为低, 1为高 |
| 2 | 第一个字节代表DO号, DO号从0开始  第二个字节代表高低状态, 0为低, 1为高 |
| ... | 第一个字节代表DO号, DO号从0开始  第二个字节代表高低状态, 0为低, 1为高 |
| CRC校验 | 2 | | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0307 |
| 返回码 | 2 | RC |
| 数据长度 | 2 |  |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 获取 FDO作为普通DO的状态

说明：将FDO作为普通DO功能使用, 获取DO的高低状态

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0308 |
| 数据长度 | 2 | 0 |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0308 |
| 返回码 | 2 | RC |
| 数据长度 | 2 | 获取到的DO数据长度 |
| 数据内容 | N | 获取到的DO状态, 每byte代表每个DO的高低, 1为高, 0为低 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 启动 FDO做为高速脉冲输出

说明：启动 FDO 做为高速脉冲输出

请求：(上位机)

|  |  |  |  |
| --- | --- | --- | --- |
| 域 | 字节 | | 备注 |
| 包头 | 1 | | 0xAA |
| 包命令 | 2 | | 0x0309 |
| 数据长度 | 2 | |  |
| 数据内容 | 10 | 1 | FDO编号, 从 0 开始 |
| 4 | 频率 |
| 1 | 是否为计数脉冲,0 不是计数脉冲, 1是计数脉冲 |
| 4 | 计数脉冲的计数值 |
| CRC校验 | 2 | | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0309 |
| 返回码 | 2 | RC |
| 数据长度 | 2 |  |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 停止 FDO 高速脉冲输出

说明：停止 FDO 高速脉冲输出

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x030A |
| 数据长度 | 2 |  |
| 数据内容 | 1 | FDO编号, 从 0 开始 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x030A |
| 返回码 | 2 | RC |
| 数据长度 | 2 |  |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 获取 FDO作为高速脉冲输出时的状态

说明：获取 FDO 状态

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x030B |
| 数据长度 | 2 |  |
| 数据内容 | 1 | FDO编号, 从 0 开始 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |  |
| --- | --- | --- | --- |
| 域 | 字节 | | 备注 |
| 包头 | 1 | | 0xAA |
| 包命令 | 2 | | 0x030B |
| 返回码 | 2 | | RC |
| 数据长度 | 2 | | 状态值 |
| 数据内容 | 5 | 1 | 脉冲启停状态, 0为停止, 1为运行 |
| 4 | 运行时的频率 |
| CRC校验 | 2 | | 除去包头的其它所有数据进行CRC16校验运算。 |

### 启动 FDO 作为 PWM输出

说明：启动 FDO 做为PWM输出

请求：(上位机)

|  |  |  |  |
| --- | --- | --- | --- |
| 域 | 字节 | | 备注 |
| 包头 | 1 | | 0xAA |
| 包命令 | 2 | | 0x030C |
| 数据长度 | 2 | |  |
| 数据内容 | 5 | 1 | FDO编号, 从 0 开始 |
| 2 | PWM T0 |
| 2 | PWM T |
| CRC校验 | 2 | | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x030C |
| 返回码 | 2 | RC |
| 数据长度 | 2 |  |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 停止FDO 的 PWM 输出

说明：停止 FDO 高速脉冲输出

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x030D |
| 数据长度 | 2 |  |
| 数据内容 | 1 | FDO编号, 从 0 开始 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x030D |
| 返回码 | 2 | RC |
| 数据长度 | 2 |  |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 获取 FDO 作为 PWM 输出的状态

说明：获取 FDO 状态

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x030E |
| 数据长度 | 2 |  |
| 数据内容 | 1 | FDO编号, 从 0 开始 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |  |
| --- | --- | --- | --- |
| 域 | 字节 | | 备注 |
| 包头 | 1 | | 0xAA |
| 包命令 | 2 | | 0x030E |
| 返回码 | 2 | | RC |
| 数据长度 | 2 | | 状态值 |
| 数据内容 | 5 | 2 | PWM T0 |
| 2 | PWM T |
| 1 | 运行状态, 0 为停止, 1为运行 |
| CRC校验 | 2 | | 除去包头的其它所有数据进行CRC16校验运算。 |

### 获取 ADC 的电压值

说明：获取 ADC 测得的电压值

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x030F |
| 数据长度 | 2 |  |
| 数据内容 | 1 | ADC编号, 从 0 开始 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x030F |
| 返回码 | 2 | RC |
| 数据长度 | 2 | 电压值长度 |
| 数据内容 | 4 | 电压值, float 类型 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 设置 DAC输出值

说明：设置 DAC的电压值

请求：(上位机)

|  |  |  |  |
| --- | --- | --- | --- |
| 域 | 字节 | | 备注 |
| 包头 | 1 | | 0xAA |
| 包命令 | 2 | | 0x0310 |
| 数据长度 | 2 | |  |
| 数据内容 | 1 | 1 | DAC编号, 从 0 开始 |
| 4 | 要设置的电压值, float类型 |
| CRC校验 | 2 | | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0310 |
| 返回码 | 2 | RC |
| 数据长度 | 2 | 0 |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

## 文件模块(0x0400, 0x04ff)

### 打开文件

说明：打开文件，类似文件操作的 fd = open(const char \*filename, unsigned char mode)

请求：(上位机)

|  |  |  |  |
| --- | --- | --- | --- |
| 域 | 字节 | | 备注 |
| 包头 | 1 | | 0xAA |
| 包命令 | 2 | | 0x0400 |
| 数据长度 | 2 | | N |
| 数据内容 | N | x | 文件名，字符串格式，最后一个字节必须为0x00 |
| 1 | Mode:可以异或组合  O\_RDONLY==0x01  O\_WRONLY==0x02  O\_CREAT==0x10  O\_TRUNC==0x08  O\_APPEND==0x30 |
| CRC校验 | 2 | | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0400 |
| 返回码 | 2 | RC |
| 数据长度 | 2 | 4 |
| 数据内容 | 4 | 文件操作句柄 fd |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 读文件

说明：读文件

请求：(上位机)

|  |  |  |  |
| --- | --- | --- | --- |
| 域 | 字节 | | 备注 |
| 包头 | 1 | | 0xAA |
| 包命令 | 2 | | 0x0401 |
| 数据长度 | 2 | | 12 |
| 数据内容 | 12 | 4 | 文件操作句柄 fd |
| 4 | 文件偏移量 |
| 4 | 读取长度 |
| CRC校验 | 2 | | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0401 |
| 返回码 | 2 | RC |
| 数据长度 | 2 | N |
| 数据内容 | N | 读到的数据 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 写文件

说明：写文件

请求：(上位机)

|  |  |  |  |
| --- | --- | --- | --- |
| 域 | 字节 | | 备注 |
| 包头 | 1 | | 0xAA |
| 包命令 | 2 | | 0x0402 |
| 数据长度 | 2 | | N |
| 数据内容 | N | 4 | 文件操作句柄 fd |
| 4 | 文件偏移量 |
| x | 写入的内容 |
| CRC校验 | 2 | | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0402 |
| 返回码 | 2 | RC |
| 数据长度 | 2 | 4 |
| 数据内容 | 4 | 写入的长度 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 关闭文件

说明：关闭文件，类似文件操作的 close(int fd)

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0403 |
| 数据长度 | 2 | 4 |
| 数据内容 | 4 | 文件操作句柄 fd |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0403 |
| 返回码 | 2 | RC |
| 数据长度 | 2 | 0 |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 删除文件

说明：删除文件

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0404 |
| 数据长度 | 2 | N |
| 数据内容 | N | 文件名，字符串格式，最后一个字节必须为0x00 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0404 |
| 返回码 | 2 | RC |
| 数据长度 | 2 | 0 |
| 数据内容 | 0 |  |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 获取文件大小

说明：获取文件大小

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0405 |
| 数据长度 | 2 | N |
| 数据内容 | N | 文件名，字符串格式，最后一个字节必须为0x00 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0405 |
| 返回码 | 2 | RC |
| 数据长度 | 2 | N |
| 数据内容 | 4 | 文件大小 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

### 获取文件列表

说明：获取指定目录下的文件夹和文件名称

请求：(上位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0406 |
| 数据长度 | 2 | N |
| 数据内容 | N | 文件名，字符串格式，最后一个字节必须为0x00 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

应答：(下位机)

|  |  |  |
| --- | --- | --- |
| 域 | 字节 | 备注 |
| 包头 | 1 | 0xAA |
| 包命令 | 2 | 0x0406 |
| 返回码 | 2 | RC |
| 数据长度 | 2 |  |
| 数据内容 | N | 文件列表, xxx/ 表示目录, xxx/xxx 表示文件. 文件列表由 n 个组成, 每个文件名以 ‘\n’ 换行符结尾 |
| CRC校验 | 2 | 除去包头的其它所有数据进行CRC16校验运算。 |

# 用户命令 (0xA000, 0xC000)

用户命令用于不同工程项目各自添加自己的通信命令, 仅仅对本身的应用项目有效.不同的应用项目新增的命令可以重复. 具体参考各个项目的通信接口子协议部分.

# 监控事件ID定义

|  |  |  |
| --- | --- | --- |
| 模块名称 | ID号 | 备注 |
| 电磁模块加热线圈温度 | 0x01 |  |
| 锅胆外壁温度 | 0x02 |  |
| 当前加热挡位 | 0x03 |  |
| 当前烹饪步骤号 | 0x04 |  |
| DI状态 | 0x05 |  |
| FDI1状态 | 0x06 | 高低状态, 脉冲量 |
| DO状态 | 0x07 | 当前输出高低 |
| 步进电机状态 | 0x08 | 运行或者停止,方向,当前脉冲的频率和占空比, 当前累计脉冲量 |
| 直流电机 | 0x09 | 运行或者停止,方向,当前脉冲的频率和占空比, 电流大小 |
| 红外温度探头 | 0x0a | 温度值, AD值 |

# 返回码参数表

#define ERR\_NONE 0

//general error number

#define ERR\_INVALID\_PARM -1000 /\* parameter invalid err \*/

#define ERR\_HARDWARE\_ERROR -1001 /\* hardware err \*/

#define ERR\_BUF\_OVERFLOW -1002 /\* byte not alignment err \*/

#define ERR\_ALIGN\_ERR -1003 /\* size is limitted err \*/

#define ERR\_UNFINISHED\_ERR -1004 /\* action not finish err \*/

#define ERR\_MALLOC\_ERR -1005 /\* malloc memory err \*/

#define ERR\_LEN\_LIMIT\_ERR -1006 /\* length is limitted err \*/

#define ERR\_SIZE\_LIMIT\_ERR -1007 /\* size is limitted err \*/

#define ERR\_UNSUPPORT\_BOARD\_ERR -1008 /\* unsupportted board err \*/

#define ERR\_TIMEOUT\_ERR -1009 /\* timeout err \*/

#define ERR\_RING\_BUF\_FULL\_ERR -1010 /\* ring buffer is full err \*/

#define ERR\_RING\_BUF\_EMPTY\_ERR -1011 /\* ring buffer is empty err \*/

#define ERR\_HARDWARE\_BUSY\_ERR -1012 /\* hardware busy err \*/

#define ERR\_HARDWARE\_RESOURCE\_ERR -1013 /\* no such hardware resource err \*/

#define ERR\_BUF\_NOT\_ENOUGH -1014 /\* no enough memory err \*/

#define ERR\_FLASH\_WRITE -1015 /\* write flash err \*/

#define ERR\_FLASH\_READ -1016 /\* read flash err \*/

#define ERR\_FLASH\_ERASE -1017 /\* erase flash err \*/

#define ERR\_UNIMPLEMENTED -1018 /\* Unimplemented err \*/

#define ERR\_NOT\_EXIST\_IN\_SYSTEM -1019 /\* not exist err \*/

#define ERR\_INI\_FILE\_FORMAT -1020 /\* ini file format err \*/

#define ERR\_INI\_SECTION\_ERR -1021 /\* ini file section err \*/

#define ERR\_INI\_KEY\_ERR -1022 /\* ini file key err \*/

#define ERR\_SYS\_EMERGENCY\_STOP -1023 /\* system is in emergency stopped err \*/

#define ERR\_NULL\_POINT -1024 /\* NULL point err \*/

#define ERR\_CHECKSUM\_ERR -1025 /\* checksum err \*/

#define ERR\_INVALID\_DATA\_ERR -1026 /\* invalid data err \*/

#define ERR\_LOG\_SYSTEM\_ERR -1027 /\* log system err \*/

#define ERR\_COMMUNICATION\_ERR -1028 /\* communication err \*/

//os operation error number

#define ERR\_OS\_TIMEOUR -2000 /\* timeout err \*/

#define ERR\_OS\_UNSUPPORTTED\_API -2001 /\* unsupported api error \*/

#define ERR\_OS\_CREAT\_TASK -2002 /\* creat task error \*/

#define ERR\_OS\_NO\_TASK -2003 /\* no task error \*/

#define ERR\_OS\_WAKEUP -2004 /\* wakeup task error \*/

#define ERR\_OS\_CREAT\_SEM -2005 /\* creat semaphore error \*/

#define ERR\_OS\_DELETE\_SEM -2006 /\* delete semaphore error \*/

#define ERR\_OS\_SEM\_TAKE -2007 /\* take semaphore error \*/

#define ERR\_OS\_SEM\_GIVE -2008 /\* give semaphore error \*/

#define ERR\_OS\_SEM\_LIMIT -2009 /\* give semaphore limit error \*/

#define ERR\_OS\_SEM\_GET\_COUNT -2010 /\* get semaphore count error \*/

#define ERR\_OS\_CREAT\_MUTEX -2011 /\* creat mutex error \*/

#define ERR\_OS\_CREAT\_QUEUE -2012 /\* creat queue error \*/

#define ERR\_OS\_DELETE\_QUEUE -2013 /\* delete queue error \*/

#define ERR\_OS\_SEND\_QUEUE -2014 /\* send queue error \*/

#define ERR\_OS\_RECV\_QUEUE -2015 /\* receive queue error \*/

#define ERR\_OS\_CREAT\_EVENT -2016 /\* creat event group error \*/

#define ERR\_OS\_CREAT\_TIMER -2017 /\* creat software timer error \*/

#define ERR\_OS\_DEL\_TIMER -2018 /\* delete software timer error \*/

#define ERR\_OS\_START\_TIMER -2019 /\* start software timer error \*/

#define ERR\_OS\_SCHEDULER\_FAIL -2020 /\* os scheduler failed error \*/

#define ERR\_OS\_NEW\_INSTANCE -2021 /\* new instance error \*/

#define ERR\_OS\_CREAT\_MESG -2022 /\* creat a os message error \*/

#define ERR\_NO\_SUCH\_DEVICE -2023 /\* no such device error \*/

#define ERR\_OS\_CREAT\_ENVENT -2024 /\* envent group have not created error \*/

#define ERR\_OS\_TRUSTEESHIP\_REG -2025 /\* trusteeship service register error \*/

//fs operation error number

#define ERR\_FS\_IOCTL -3000 /\* fs ioctl error \*/

#define ERR\_FS\_OPEN -3001 /\* fs open error \*/

#define ERR\_FS\_DEVICE\_NOT\_OPEN -3002 /\* device not open error \*/

#define ERR\_DEV\_REG\_REPEAT\_ERR -3003 /\* repeat register device error \*/

#define ERR\_DEV\_REG\_LIMIT\_ERR -3004 /\* too much device registered error \*/

#define ERR\_DEV\_NOTFOUND\_ERR -3005 /\* no this device found error \*/

#define ERR\_DEV\_NULL\_POINT\_ERR -3006 /\* point NULL error \*/

#define ERR\_DEV\_OPEN\_ERR -3007 /\* open device error \*/

#define ERR\_DEV\_NOT\_REDAY\_ERR -3008 /\* device not ready error \*/

#define ERR\_FS\_IOCTL\_UNSUPPORTTED -3009 /\* fs ioctl unsupportted error \*/

//hardware error number

#define ERR\_CAN\_SEND\_ERR -4000 /\* can send data error \*/

#define ERR\_CAN\_GET\_ERR -4001 /\* can get data error \*/

#define ERR\_CAN\_RX\_TIMEOUT -4002 /\* can recieve timeout error \*/

#define ERR\_CAN\_TX\_TIMEOUT -4003 /\* can transmit timeout error \*/

#define ERR\_UXART\_SEND\_ERR -4004 /\* uxart send data error \*/

#define ERR\_UXART\_RECV\_ERR -4005 /\* uxart recieve data error \*/

#define ERR\_UXART\_RX\_EMPTY\_ERR -4006 /\* uxart no rx data error \*/

#define ERR\_UXART\_NOT\_READY -4007 /\* uxart not ready error \*/

#define ERR\_ADD\_CHANNEL -4008 /\* adc add channel error \*/

//file operation error number, return from fatfs

#define ERR\_FILE\_BASE\_ERR -5000 /\* error base number \*/

#define ERR\_FILE\_DISK\_ERR -5001 /\* (1) A hard error occurred in the low level disk I/O layer \*/

#define ERR\_FILE\_INT\_ERR -5002 /\* (2) Assertion failed \*/

#define ERR\_FILE\_NOT\_READY -5003 /\* (3) The physical drive cannot work \*/

#define ERR\_FILE\_NO\_FILE -5004 /\* (4) Could not find the file \*/

#define ERR\_FILE\_NO\_PATH -5005 /\* (5) Could not find the path \*/

#define ERR\_FILE\_INVALID\_NAME -5006 /\* (6) The path name format is invalid \*/

#define ERR\_FILE\_DENIED -5007 /\* (7) Access denied due to prohibited access or directory full \*/

#define ERR\_FILE\_EXIST -5008 /\* (8) Access denied due to prohibited access \*/

#define ERR\_FILE\_INVALID\_OBJECT -5009 /\* (9) The file/directory object is invalid \*/

#define ERR\_FILE\_WRITE\_PROTECTED -5010 /\* (10) The physical drive is write protected \*/

#define ERR\_FILE\_INVALID\_DRIVE -5011 /\* (11) The logical drive number is invalid \*/

#define ERR\_FILE\_NOT\_ENABLED -5012 /\* (12) The volume has no work area \*/

#define ERR\_FILE\_NO\_FILESYSTEM -5013 /\* (13) There is no valid FAT volume \*/

#define ERR\_FILE\_MKFS\_ABORTED -5014 /\* (14) The f\_mkfs() aborted due to any problem \*/

#define ERR\_FILE\_TIMEOUT -5015 /\* (15) Could not get a grant to access the volume within defined period \*/

#define ERR\_FILE\_LOCKED -5016 /\* (16) The operation is rejected according to the file sharing policy \*/

#define ERR\_FILE\_NOT\_ENOUGH\_CORE -5017 /\* (17) LFN working buffer could not be allocated \*/

#define ERR\_FILE\_TOO\_MANY\_OPEN\_FILES -5018 /\* (18) Number of open files > FF\_FS\_LOCK \*/

#define ERR\_FILE\_INVALID\_PARAMETER -5019 /\* (19) Given parameter is invalid \*/

#define ERR\_ILLEGAL\_FW\_ERR -5020

#define ERR\_FILE\_OPEN\_LIMIT\_ERR -5021

#define ERR\_FW\_TOO\_LARGE\_ERR -5022

#define ERR\_FW\_NOT\_FOUND\_ERR -5023

#define ERR\_DIR\_OPEN\_LIMIT\_ERR -5024

//upper communication error number

#define ERR\_UPPER\_MSG\_INCOMPLETE -6000 /\* message from upper is not complete error \*/

#define ERR\_UPPER\_MSG\_CRC -6001 /\* upper crc error \*/

#define ERR\_UPPER\_MSG\_UNSUPPORT -6002 /\* upper message not supported error \*/

#define ERR\_UPPER\_UPDATE\_MENU -6003 /\* update menu error \*/

#define ERR\_UPPER\_MISSING\_HEAD -6004 /\* head of frame is missing error \*/

#define ERR\_UPPER\_MSG\_LEN -6005 /\* length of frame is error \*/

#define ERR\_NO\_VALID\_MENU -6006 /\* no valid menu error \*/

#define ERR\_GET\_LOCAL\_MENU -6007 /\* get local menu error \*/

//network error number

#define ERR\_NETWORK\_LOST\_CONNECT -7000 /\* network lost connection error \*/

#define ERR\_NETWORK\_RECV -7001 /\* network receive error \*/

#define ERR\_NETWORK\_SEND -7002 /\* network send error \*/

#define ERR\_NETWORK\_IGNORE\_MSG -7003 /\* network ignore this message error \*/

#define ERR\_GET\_IP\_BY\_HOST -7004 /\* network get ip by hostname error \*/

#define ERR\_NETWORK\_ADDR\_INVALID -7005 /\* network address invalid error \*/

#define ERR\_NETWORK\_NOT\_READY -7006 /\* network not ready error \*/

#define ERR\_NETWORK\_MQTT\_SUB -7007 /\* network mqtt subcribe error \*/

#define ERR\_NETWORK\_MQTT\_PUB -7008 /\* network mqtt publish error \*/

//VL53L0X error number

#define ERR\_DATA\_INVALID -8000 /\* VL53L0X measure data invalid error \*/

#define ERR\_WORK\_INNORMAL -8001 /\* VL53L0X work innormal error \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

########################## Applications error number ##########################

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* shuangcengguo specific error number \*/

#define ERR\_PAN\_ZHANWEI -10000 /\* pan error number start \*/

#define ERR\_PAN\_INVALID\_CTRL -10001 /\* ileagal coommand sent to double\*/

/\* railway specific error number \*/

#define ERR\_RAILWAY\_MOTOR\_ALM -10200 /\* railway motor alarm from motor drive\*/

#define ERR\_RAILWAY\_TOEC\_ERR -10201 /\* railway toec sensor error, signal triggered from both side\*/

#define ERR\_RAILWAY\_RUN\_TIME -10202 /\* railway beyond run time, maybe the structure is blocked\*/

/\* AGV specific error number \*/

#define ERR\_AGV\_ZHANWEI -10400 /\* agv error number start \*/

/\* SHANG CAI KOU specific error number \*/

#define ERR\_SHANGCAIKOU\_ZHANWEI -10600 /\* shang cai kou error number start \*/

/\* XIA CAI KOU specific error number \*/

#define ERR\_XIACAIKOU\_ZHANWEI -10800 /\* xia cai kou error number start \*/

/\* PIZZA specific error number \*/

#define ERR\_PIZZA\_ZHANWEI -11000 /\* pizza error number start \*/

#define ERR\_PIZZA\_OVERTEMPERATURN -11001 /\* Over temperature \*/

#define ERR\_PIZZA\_CS\_NUM\_OVER -11002 /\* CS of the MAX6675 chip is over \*/

#define ERR\_PIZZA\_TEMP\_MODEL\_CONNET -11003 /\* Temperature model is losed with connection\*/

#define ERR\_PIZZA\_TEMP\_MODEL\_WRITE -11004 /\* Can not write data to the register in Temperature model \*/

#define ERR\_PIZZA\_VFD\_MODEL\_CONNET -11005 /\* VFD model is losed with connection\*/

#define ERR\_PIZZA\_VFD\_MODEL\_WRITE -11006 /\* Can not write data to the register in VFD model\*/

#define ERR\_PIZZA\_MODBUS\_MASTER\_CRC\_CHECK -11005 /\*The Modbus\_master CRC check is error\*/

/\* Mini\_Ice\_Cream Error number \*/

#define ERR\_MIC\_AST\_WRITE -12001 /\* Writting data to the ast machine is error\*/

/\*Coffee Error number\*/

#define ERR\_COFFEE\_BODY\_WRITE -12101 /\*Writting data to the coffee body is error\*/

/\* special error number \*/

#define ERR\_THIRDPART\_ERR -20000 /\* error baseline error, this error is for third part source error return \*/

# CRC校验算法示例

/\*\* CRC-16-ANSI table values \*/

#defineCOMMAND\_CRC16\_TABLE { \

0x0000, 0xc0c1, 0xc181, 0x0140, 0xc301, 0x03c0, 0x0280, 0xc241, \

0xc601, 0x06c0, 0x0780, 0xc741, 0x0500, 0xc5c1, 0xc481, 0x0440, \

0xcc01, 0x0cc0, 0x0d80, 0xcd41, 0x0f00, 0xcfc1, 0xce81, 0x0e40, \

0x0a00, 0xcac1, 0xcb81, 0x0b40, 0xc901, 0x09c0, 0x0880, 0xc841, \

0xd801, 0x18c0, 0x1980, 0xd941, 0x1b00, 0xdbc1, 0xda81, 0x1a40, \

0x1e00, 0xdec1, 0xdf81, 0x1f40, 0xdd01, 0x1dc0, 0x1c80, 0xdc41, \

0x1400, 0xd4c1, 0xd581, 0x1540, 0xd701, 0x17c0, 0x1680, 0xd641, \

0xd201, 0x12c0, 0x1380, 0xd341, 0x1100, 0xd1c1, 0xd081, 0x1040, \

0xf001, 0x30c0, 0x3180, 0xf141, 0x3300, 0xf3c1, 0xf281, 0x3240, \

0x3600, 0xf6c1, 0xf781, 0x3740, 0xf501, 0x35c0, 0x3480, 0xf441, \

0x3c00, 0xfcc1, 0xfd81, 0x3d40, 0xff01, 0x3fc0, 0x3e80, 0xfe41, \

0xfa01, 0x3ac0, 0x3b80, 0xfb41, 0x3900, 0xf9c1, 0xf881, 0x3840, \

0x2800, 0xe8c1, 0xe981, 0x2940, 0xeb01, 0x2bc0, 0x2a80, 0xea41, \

0xee01, 0x2ec0, 0x2f80, 0xef41, 0x2d00, 0xedc1, 0xec81, 0x2c40, \

0xe401, 0x24c0, 0x2580, 0xe541, 0x2700, 0xe7c1, 0xe681, 0x2640, \

0x2200, 0xe2c1, 0xe381, 0x2340, 0xe101, 0x21c0, 0x2080, 0xe041, \

0xa001, 0x60c0, 0x6180, 0xa141, 0x6300, 0xa3c1, 0xa281, 0x6240, \

0x6600, 0xa6c1, 0xa781, 0x6740, 0xa501, 0x65c0, 0x6480, 0xa441, \

0x6c00, 0xacc1, 0xad81, 0x6d40, 0xaf01, 0x6fc0, 0x6e80, 0xae41, \

0xaa01, 0x6ac0, 0x6b80, 0xab41, 0x6900, 0xa9c1, 0xa881, 0x6840, \

0x7800, 0xb8c1, 0xb981, 0x7940, 0xbb01, 0x7bc0, 0x7a80, 0xba41, \

0xbe01, 0x7ec0, 0x7f80, 0xbf41, 0x7d00, 0xbdc1, 0xbc81, 0x7c40, \

0xb401, 0x74c0, 0x7580, 0xb541, 0x7700, 0xb7c1, 0xb681, 0x7640, \

0x7200, 0xb2c1, 0xb381, 0x7340, 0xb101, 0x71c0, 0x7080, 0xb041, \

0x5000, 0x90c1, 0x9181, 0x5140, 0x9301, 0x53c0, 0x5280, 0x9241, \

0x9601, 0x56c0, 0x5780, 0x9741, 0x5500, 0x95c1, 0x9481, 0x5440, \

0x9c01, 0x5cc0, 0x5d80, 0x9d41, 0x5f00, 0x9fc1, 0x9e81, 0x5e40, \

0x5a00, 0x9ac1, 0x9b81, 0x5b40, 0x9901, 0x59c0, 0x5880, 0x9841, \

0x8801, 0x48c0, 0x4980, 0x8941, 0x4b00, 0x8bc1, 0x8a81, 0x4a40, \

0x4e00, 0x8ec1, 0x8f81, 0x4f40, 0x8d01, 0x4dc0, 0x4c80, 0x8c41, \

0x4400, 0x84c1, 0x8581, 0x4540, 0x8701, 0x47c0, 0x4680, 0x8641, \

0x8201, 0x42c0, 0x4380, 0x8341, 0x4100, 0x81c1, 0x8081, 0x4040 \

}

/\*\* CRC-16-ANSI table \*/

staticunsignedshort \_crc16\_table[] = COMMAND\_CRC16\_TABLE;

/\*\*

\* Calculate a CRC-16-ANSI checksum from data in the buffer and crc specified.

\* @param p Buffer

\* @param crc CRC base.

\* @param len Length

\* @return CRC calculated

\*/

unsignedshortcrc16(void \*p, unsignedshortcrc, unsignedintlen) {

unsignedchar\* ptemp = (unsignedchar \*)p;

inti = 0;

for (i = 0; i<len; i++) {

crc = (((crc / 256 ) &0xff) ^ \_crc16\_table[(crc ^ (ptemp[i])) &0xff]) &0x0000ffff;

}

return (crc&0xFFFF);

}