**大件摆轮分拣机以太网通信协议**

版本：v5.1.11

目录

[一、通讯简介 2](#_Toc1026)

[1.1 板卡连接示意图 2](#_Toc4495)

[1.2 通讯协议格式 2](#_Toc12889)

[1.3 通讯规则 2](#_Toc23259)

[二、通讯命令详解 3](#_Toc19238)

[2.0初始化信息 (摆轮板卡数量) 3](#_Toc23937)

[2.1分拣命令 3](#_Toc23204)

[2.2分拣结果回应 4](#_Toc27857)

[2.3.报警信息 4](#_Toc26366)

[2.4摆轮电滚筒报警信息（暂时没用到） 5](#_Toc15888)

[2.5发送给主控系统光电信号(可选) 6](#_Toc8446)

[2.6心跳命令 6](#_Toc21752)

[2.7 线体启停控制 7](#_Toc9001)

[2.8 线体启停状态查询 7](#_Toc14566)

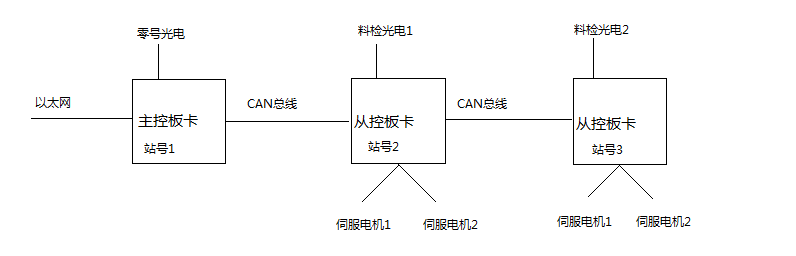
[2.9 摆轮本地/远程模式查询 8](#_Toc31406)

[2.10版本号信息上传 9](#_Toc24713)

[2.11调速命令 9](#_Toc20689)

## 一、通讯简介

### 1.1 板卡连接示意图



### 1.2 通讯协议格式

数据帧格式：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 帧头 | 帧序号 | 帧长度 | 帧校验 | 帧命令 | 数据区 |

* 帧 头：2个字节，区分每一帧数据，帧头约定为“0xAAAA”。
* 帧序号：4个字节，发送方每一帧自动加1，应答方应答时使用接收到发送方使用的帧序号。
* 帧长度：2个字节，表示整个帧的数据长度。
* 帧校验：1个字节，对帧数据中的帧命令和数据区的数据进行异或计算。
* 帧命令：2个字节，帧的命令类型代码，表示该帧数据的功能，发送命令最高位为0，应答方使用的应答命令为接收到的命令最高位或1。
* 数据区：每条命令的具体数据，详见命令说明。

**注**：通信协议帧中的数据全部为小端格式。

### 1.3 通讯规则

主控系统采用UDP方式和摆轮控制器进行通讯。主控系统使用固定IP：192.168.2.100 port：9000，摆轮控制器使用固定IP：192.168.2.110 port：9000。

## 二、通讯命令详解

### 2.0初始化信息 (摆轮板卡数量)

此协议: 自动供包摆轮共用

（1）发送时间

总控系统每次在使用摆轮前需要发送该初始化参数

（2）发送帧命令：**0x1A00**

（3）数据格式定义

typedef struct{

unsigned char cmd[11];

unsigned char board\_total;

} sPC2Board\_Init;

|  |  |  |
| --- | --- | --- |
| **变量类型** | 变量名 | **含义** |
| unsigned char | cmd[11] | 详见1.1通讯协议格式 |
| unsigned char | board\_total | 板卡数量 |

（4）应答帧命令：**0x9A00**

（5）数据格式定义

typedef struct{

unsigned char cmd[11];

} sPC2Board\_Init;

|  |  |  |
| --- | --- | --- |
| **变量类型** | **变量名** | **含义** |
| unsigned char | cmd[11] | 详见1.1通讯协议格式 |

### 2.1分拣命令

（1）发送时间

主控系统发送给摆轮控制器分拣命令。

（2）发送帧命令：**0x1A01**

（3）数据格式定义

typedef struct{

unsigned char cmd[11];

unsigned int32 msg\_id;

unsigned char controller\_num;

unsigned char direction;

} sPC2Controller\_unload\_Data;

|  |  |  |
| --- | --- | --- |
| **变量类型** | **变量名** | **含义** |
| unsigned char | cmd[11] | 详见1.1通讯协议格式 |
| unsigned int32 | msg\_id | 下料命令序号 |
| unsigned char | controller\_num | 摆轮站号 |
| unsigned char | direction | 摆动方向（沿着主线速度方向，0 左摆 1 右摆 2回正） |

（4）应答帧命令：**0x9A01**

（5）数据格式定义

typedef struct{

unsigned char cmd[11];

unsigned int32 msg\_id;（下料命令序号）

} sPC2 Controller\_unload\_Data\_ACK;

|  |  |  |
| --- | --- | --- |
| **变量类型** | **变量名** | **含义** |
| unsigned char | cmd[11] | 详见1.1通讯协议格式 |

### 2.2分拣结果回应

（1）发送时间

反馈摆轮是否能执行主控系统发送的分拣命令，由摆轮控制器主动发送。

（2）发送帧命令：**0x1A02**

（3）数据格式定义

typedef struct{

unsigned char cmd[11];

unsigned char result;

unsigned int32 msg\_id;

}  sController2PC\_Unload\_Result;

|  |  |  |
| --- | --- | --- |
| **变量类型** | **变量名** | **含义** |
| unsigned char | cmd[11] | 详见1.1通讯协议格式 |
| unsigned char | msg\_num | 回应的消息个数 |
| unsigned char | result | 摆轮0 执行了分拣动作，1未执行分拣动作 |
| unsigned int32 | msg\_id | 下料命令序号 |

（4）应答帧命令：无需应答

### 2.3.报警信息

（1）发送时间

摆轮控制器,每隔1秒钟向主控系统发送报警信息,无错误状态不发送。

（2）发送帧命令：**0x1A03**

（3）数据格式定义

typedef struct{

unsigned char cmd[11];

unsigned char error\_num;

sAlarmData error\_data[error\_num];

} sController2PC\_Alarm\_Data;

|  |  |  |
| --- | --- | --- |
| **变量类型** | **变量名** | **含义** |
| unsigned char | cmd[11] | 详见1.1通讯协议格式 |
| unsigned char | error\_num | 有报警的控制器的数量 |
| sAlarmData | error\_data[error\_num] | 每个控制器的报警数据 |

报警数据格式如下:

typedef struct{

unsigned char broad\_num; //摆轮站号

unsigned char err\_num //该摆轮报警的数量

unsigned char error\_status; // 错误码

0x01急停报警; 0x02 通讯报警; 0x03 摆轮未准备好

0x11第一组伺服报警;0x21第二组伺服报警;

0x31第三组伺服报警;0x41第四组伺服报警;

0x51第五组伺服报警;0x61第六组伺服报警;

0x71第七组伺服报警;0x81第八组伺服报警;

0x91第九组伺服报警;0XA1第十组伺服报警;

0x12 第一组寻零光电报警 0x22第二组寻零光电报警

0x32 第三组寻零光电报警 0x42第四组寻零光电报警

0x52 第五组寻零光电报警 0x62第六组寻零光电报警

0x72 第七组寻零光电报警 0x82第八组寻零光电报警

0x92 第九组寻零光电报警 0XA2第十组寻零光电报警

修改后（每一位对应一种报警）：

0x01急停报警; 0x02 通讯报警;

0x04第一组伺服报警;0x08第二组伺服报警;

0x10第三组伺服报警;0x20第四组伺服报警;

} sAlarmData

（4）应答帧命令：无需应答

### 2.4摆轮电滚筒报警信息（暂时没用到）

（1）发送时间

摆轮控制器检测到有驱动器报警状态时上传，一秒一次,一次传一个从站的报警信息，无错误状态不上传。

（2）发送帧命令：**0x1A04**

（3）数据格式定义

typedef struct{

unsigned char cmd[11];

unsigned char controller\_ID;

unsigned char error\_num;

sDriverData error\_data[error\_num];

} sController2PC\_Driver\_Alarm\_Data;

|  |  |  |
| --- | --- | --- |
| **变量类型** | **变量名** | **含义** |
| unsigned char | cmd[11] | 详见1.1通讯协议格式 |
| unsigned char | board\_ID | 报警的摆轮站号 |
| unsigned char | error\_num | 报警的电滚筒总数量 |
| sAlarmData | error\_data[error\_num] | 每个电滚筒的报警数据 |

报警数据格式如下:

typedef struct {

unsigned char driver\_count; //电滚筒序号;

unsigned char error\_status; //错误状态

0x01：滚筒电机堵转

0x02：滚筒电机过流

0x03：48V电源过压/欠压

0x04：滚筒驱动器报警

0x0f：滚筒驱动器未连接

0x10: 设置驱动器速度失败

0x20: 设置驱动器分组失败

0x40: 设置驱动器型号失败

} sDriverData

（4）应答帧命令：无需应答

### 2.5发送给主控系统光电信号(可选)

（1）发送时间

如果有包裹检测光电 接到了摆轮控制器，则收到光电开关触发信号时，控制器将立即向主控系统发送该光电触发的信息，滤波时间50ms； 如果没接光电 则不发。

（2）发送帧命令：**0x1A05**

（3）数据格式定义

typedef struct{

unsigned char cmd[11];

unsigned char photoSwitch\_num;

} sBoard2PC\_Photoelectric\_Data;

|  |  |  |
| --- | --- | --- |
| **变量类型** | **变量名** | **含义** |
| unsigned char | cmd[11] | 详见1.1通讯协议格式 |
| unsigned char | photoSwitch\_num | 光电开关序号（与摆轮站号相同） |

（4）应答帧命令：无需应答

### 2.6心跳命令

（1）发送时间

摆轮控制器与上位机连续5秒未发生数据交互时发送。

（2）发送帧命令：**0x1150**

（3）数据格式定义

typedef struct{

unsigned char cmd[11];

} sBoard2PC\_Heart\_Data;

|  |  |  |
| --- | --- | --- |
| **变量类型** | **变量名** | **含义** |
| unsigned char | cmd[11] | 详见1.1通讯协议格式 |

（4）应答帧:无需应答

### 2.7 线体启停控制

（1）发送时间：

主动系统在需要控制线体启停时，主动发送给摆轮控制器.

（2）发送帧命令：**0x1A06**

（3）数据格式定义

typedef struct{

unsigned char cmd[11];

unsigned char num;

sStartStopMsg StartOrStop[num];

} sPC2Controller\_StartOrStop\_cmd

|  |  |  |
| --- | --- | --- |
| **变量类型** | 变量名 | **含义** |
| unsigned char | cmd[11] | 详见1.1通讯协议格式 |
| sStartStopMsg | StartOrStop; | 启停信息 |

Typedef\_struct{

unsigned char controller\_id； 摆轮站号

unsigned char run\_state； 0 停止, 1人工速度启动,2 正常速度启动

}sStartStopMsg；

（4）应答帧命令：**0x9A06**

（5）数据格式定义

typedef struct{

unsigned char cmd[11];

} sPC2Controller\_StartOrStop\_Ack

|  |  |  |
| --- | --- | --- |
| **变量类型** | 变量名 | **含义** |
| unsigned char | cmd[11] | 详见1.1通讯协议格式 |

### 2.8 线体启停状态查询

（1）发送时间

主控系统在需要时主动发送给摆轮控制器.在启停状态改变的时，摆轮控制器主动发送给主控系统.

（2） 发送帧命令：**0x1A07**

（3）数据格式定义

typedef struct{

unsigned char cmd[11];

} sPC2Board\_QueryIsRunning\_cmd

|  |  |  |
| --- | --- | --- |
| unsigned char | cmd[11] | 详见1.1通讯协议格式 |

（4）应答帧命令：**0x9A07**

（5）数据格式定义

typedef struct{

unsigned char cmd[11];

unsigned char num;

srunstate IsRunning[num] ; //线体启停状态(0 停止, 1人工速度启动,2 正常速度启动)

} sPC2Board\_ IsRunning \_cmd；

|  |  |  |
| --- | --- | --- |
| **变量类型** | 变量名 | **含义** |
| unsigned char | cmd[11] | 详见1.1通讯协议格式 |
| unsigned char | num | 摆轮数量 |
| srunstate | IsRunning[num] | 摆轮启停状态信息(0 停止, 1人工速度启动,2 正常速度启动) |

typedef struct{

unsigned char balance\_index; 摆轮站号

unsigned char runstate\_Value; 启停状态信息

} srunstate;

### 2.9 摆轮本地/远程模式查询

（1）发送时间

主控系统在需要时主动发送给摆轮控制器查询.本地远程状态改变时，摆轮控制器将各个摆轮的本地远程状态主动发给主控系统。

（2） 发送帧命令：**0x1A08**

（3）数据格式定义

typedef struct{

unsigned char cmd[11];

} sPC2Board\_LocalRemote\_state \_cmd

|  |  |  |
| --- | --- | --- |
| 变量类型 | 变量名 | 含义 |
| unsigned char | cmd[11] | 详见1.1通讯协议格式 |

（4）应答帧命令：**0x9A08**

（5）数据格式定义

typedef struct{

unsigned char cmd[11];

unsigned char num;

CModeInfo [num] ; //摆轮控制模式(0：本地 1：远程)

}  sPC2Board\_LocalRemote\_state \_cmd\_ack；

|  |  |  |
| --- | --- | --- |
| **变量类型** | 变量名 | **含义** |
| unsigned char | cmd[11] | 详见1.1通讯协议格式 |
| unsigned char | num | 摆轮数量 |
| s\_CModeInfo | CModeInfo[num] | 摆轮控制模式信息(0：本地 1：远程) |

typedef struct{

unsigned char board\_index; 摆轮站号

unsigned char Elocal\_Value; 本地远程状态信息

} s\_CModeInfo;

### 2.10版本号信息上传

（1）发送时间

主控系统发送查询命令时，主板卡回复给主控系统

（2）发送帧命令：0x1A09

（3）数据格式定义

typedef struct{

unsigned char cmd[11];

unsigned int32 version;

} sPC2Board\_ version;

|  |  |  |
| --- | --- | --- |
| **变量类型** | **变量名** | **含义** |
| unsigned char | cmd[11] | 详见1.1通讯协议格式 |

（4）应答帧命令：0x9A09  
（5）数据格式定义

typedef struct{

unsigned char cmd[11];

}sPC2Board\_ version；

|  |  |  |
| --- | --- | --- |
| **变量类型** | **变量名** | **含义** |
| unsigned char | cmd[11] | 详见1.1通讯协议格式 |
| unsigned int32 | version | 版本号（比如1.2.1.6 则 对应 0x06 0x01 0x02 0x01） |

### 2.11调速命令

（1）发送时间

上位软件在需要时主动发送给板卡.

（2）发送帧命令：**0x1A10**

（3）数据格式定义

typedef struct{

unsigned char cmd[11];

unsigned char driverSpeedValue;

} sPC2Board\_DriverSpeedValue\_cmd

|  |  |  |
| --- | --- | --- |
| **变量类型** | 变量名 | **含义** |
| unsigned char | cmd[11] | 详见1.1通讯协议格式 |
| unsigned char | driverSpeedValue | 驱动器速度值 |

驱动器速度值部分数据解释：

|  |  |
| --- | --- |
| 速度(M/S) | 调速指令3字节（速度只是对于现有摆轮63.5MM直径参考，准确的需要根据实际调节） |
| 1.5 | 18 |
| 1.8 | 1D（默认值） |
| 2.0 | 20 |
| 2.5 | 28 |
| 3 | 2F |
| 3.5 | 38 |

（4）应答帧命令：**0x9A10**

（5）数据格式定义

typedef struct{

unsigned char cmd[11];

} sPC2Board\_DriverSpeedValue\_cmd；

|  |  |  |
| --- | --- | --- |
| **变量类型** | **变量名** | **含义** |
| unsigned char | cmd[11] | 详见1.1通讯协议格式 |

|  |  |
| --- | --- |
| 版本 | 更新说明 |
|  |  |