# 目录

[目录 1](#_Toc533027817)

[一、电气图 2](#_Toc533027818)

[整体系统连接功能示意图 2](file:///F:\FILE\WEGAM\AMP\Protocol\通讯协议V2.docx#_Toc533027819)

[控制板接口接示意图 3](file:///F:\FILE\WEGAM\AMP\Protocol\通讯协议V2.docx#_Toc533027820)

[控制板全功能连接示意图 5](file:///F:\FILE\WEGAM\AMP\Protocol\通讯协议V2.docx#_Toc533027821)

[主柜电气连接示意图 6](file:///F:\FILE\WEGAM\AMP\Protocol\通讯协议V2.docx#_Toc533027822)

[副柜电气连接示意图 7](file:///F:\FILE\WEGAM\AMP\Protocol\通讯协议V2.docx#_Toc533027823)

[二、通讯协议 8](#_Toc533027824)

[协议参数 8](#_Toc533027825)

[最高发送频率 8](#_Toc533027826)

[重发机制 8](#_Toc533027827)

[应答机制 8](#_Toc533027828)

[数据校验 8](#_Toc533027829)

[通讯帧格式 9](#_Toc533027830)

[说明 9](#_Toc533027831)

[消息体内容 9](#_Toc533027832)

[消息体说明 9](#_Toc533027833)

[消息类型 11](#_Toc533027834)

[应答类型 12](#_Toc533027835)

[消息传输流程 13](#_Toc533027836)

[三、消息举例说明 14](#_Toc533027837)

[应答类型举例 14](#_Toc533027838)

[控制类消息举例 16](#_Toc533027839)

[LED控制： 16](#_Toc533027840)

[锁控制 20](#_Toc533027841)

[层板供电控制 21](#_Toc533027842)

[数据类消息举例 22](#_Toc533027843)

[读卡器命令 22](#_Toc533027844)

[四、附录 24](#_Toc533027845)

[CRC16-Modbus 24](#_Toc533027846)

[拨码表及说明 25](#_Toc533027847)

# 一、电气图

PC

Address1-主柜

Address1-副柜1

Address1-副柜…

Address2-层1

Address3-位1

Address2-层…

Address3-位n

Address3-位1

Address3-位n

Address2-层n

Address2-层

Address3-位1

Address3-位n

Address2-层

Address1-副柜n

## 整体系统连接功能示意图

说明：

1. 主柜:控制柜，内含PC，数据管理中心，所有的副柜由主柜控制

任何一个柜都可以当主柜，通过拨码设定是否当作主柜，参考附录[拨码表](#_拨码表)

1. 副柜:从机，根据主柜下发的命令执行相应的操作
2. 通讯方式：主柜内部与PC机通讯用RS232接口，主柜与副柜之间通讯通过RS485接口
3. 柜号：柜号通过柜内控制板拨码地址实现，整个网络不能有相同地址，主柜地址不限，当前拨码可设定整个网络可挂载的总柜数为63个

## 控制板接口接示意图

S1

S2

Sin3

GND

VCC

J1

J2

J3

J4

J8

RXD2

TXD2

GND

VCC

J7

RXD1

TXD1

GND

VCC

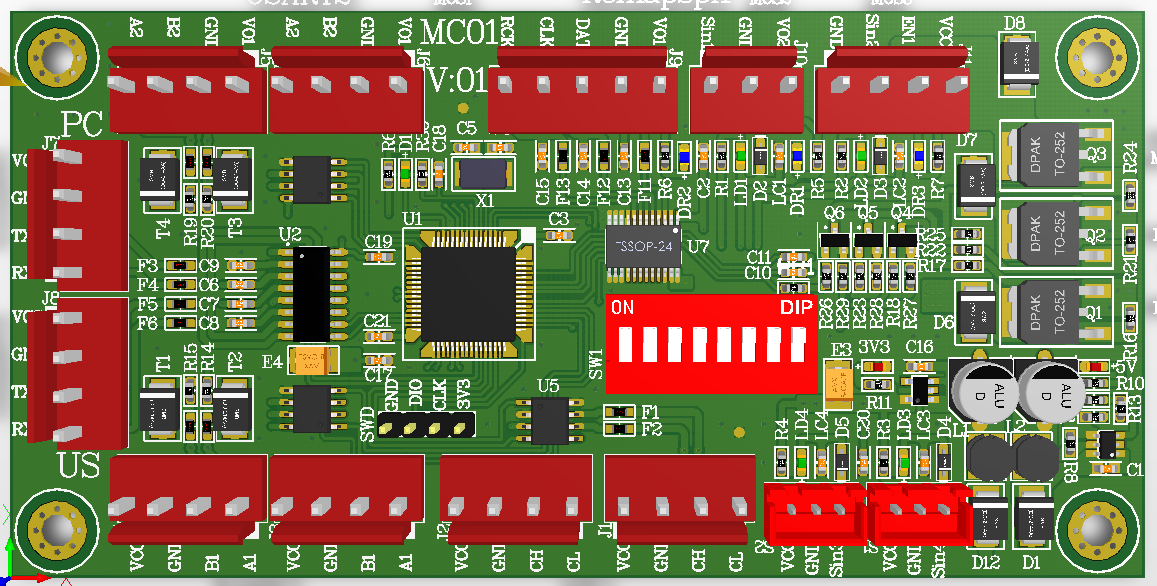
J11

J10

J6

J9

J5



Vout1

GND

485B2

485A2

Vout1

GND

485B2

485A2

Vout1

GND

DATA

CLK

RCK

Vout2

GND

Sin1

VCC

UnLock

Sin2

GND

CAN-L

CAN-H

GND

VCC

CAN-L

CAN-H

GND

VCC

485A1

485B1

GND

VCC

Sin4

GND

VCC

485A1

485B1

GND

VCC

接口说明：

1. J7,J8为RS232接口，主要为与PC机通讯或者与其它RS232接口外设通讯(例如读卡器)，接口Pin数为4，其中2Pin数据线，TXD为发送端口，RXD为接收端口，1Pin地线，1Pin电源线，电源线可作为电源输入线，如果不需要电源线，可为空
2. J3,J4为RS485接口，这两接口为并联接口，此接口为与主柜连接的总线接口，所有柜体通过这两个接口连通， Pin数为4Pin，其中2Pin为数据线，1Pin为地线，1Pin电源线，电源线可以作为本板供电输入端口，如果不需要电源线，可为空
3. J5,J6为RS485接口，这两接口为并联接口，此接口为柜体内部总线通讯接口，柜体内部所有需要用到的RS485通讯都通过此接口(如指示、显示)，Pin数为4Pin，其中2Pin为数据线，1Pin为地线，1Pin可关断电源线，电源端口由本板控制，只能用作输出
4. J1,J2为CAN接口，暂时不使用，可用作电源输入口
5. J9为SPI输出接口，无输入功能，可用来驱动595，电源端口为可控电源输出，与J5,J6共用电源开关
6. J10为带信号输入功能的电源输出端口，可用于驱动LED灯或者锁以及传感器，信号输入为低有效
7. J11为锁接口，驱动类型为NPN输出，Sin为信号输入，输入信号为低有效
8. S1，S2为传感器接口，带电源输出，电源端口与输入电源连接，无开关控制，输入信号为低有效
9. 拨码方式：全部拨码为0时为测试模式，主要为功能自检，拨码方式为二进制

最高位为SW1，最低位为SW8，SW1暂时不启用

拨码地址参考附录[拨码表](#_拨码表)

## 控制板全功能连接示意图

MC01控制板

J7-USART1:PC端口

J8-USART3：读卡器

USART2：RS485

SPI

PMOS输出

NMOS输出

UART4：RS485

CAN：100K

NPN输入型传感器

端口:J10

高输出:17A

输入:NPN型

PC端口与读卡器端口可以任意选择

端口:J11

低输出:17A

输入:NPN型

端口:J9

输出:SPI

输入:无

端口:J5,J6并联

功能:内部通讯

总线:RS485

波特率:19200

校验:无

停止位:1

端口:J3,J4并联

功能:柜体通讯

总线:RS485

波特率:19200

校验:无

停止位:1

端口:J1,J2并联

功能:备用

总线:CAN

波特率:100K

端口:S1,S2

功能:输入

类型:NPN

供电接口：

J1,J2,J3,J4,J7,J8都可以用作电源输入和输出(不受控)

J5,J6,J9,J10可用途可控电源输出

## 主柜电气连接示意图

MC01控制板

PC

读卡器

USART2：RS485

PMOS输出

NMOS输出

UART4：RS485

NPN输入型传感器

J10

LED照明灯

功率小于120W

J7,J8:PC端口与读卡器端口可以任意选择

J11

电磁锁

功率小于120W

J5,J6任意端口

内部总线(显示)

功率小于120W

J3,J4

与副柜通讯

S1,S2

按键或者传感器

说明：

主柜需要与PC连接

J3,J4接口为与副柜通讯接口,通讯线为三线(电源线不接)

## 副柜电气连接示意图

MC01控制板

读卡器

USART2：RS485

PMOS输出

NMOS输出

UART4：RS485

NPN输入型传感器

J10

LED照明灯

功率小于120W

J7,J8:PC端口与读卡器端口可以任意选择

J11

电磁锁

功率小于120W

J5,J6任意端口

内部总线(显示)

功率小于120W

J3,J4

柜与柜之间通讯

S1,S2

按键或者传感器

说明：

主柜需要与PC连接

J3,J4接口为与主柜或副柜通讯接口,通讯线为三线(电源线不接)

# 二、通讯协议

## 协议参数

1. RS232接口：波特率19200，数据位-8，校验-无，停止位-1，最大缓存255byte
2. RS485接口：波特率19200，数据位-8，校验-无，停止位-1，最大缓存255byte
3. CAN接口：波特率100K

## 最高发送频率

通讯接口由发完数据后(空闲)，需要等待2mS后再发数据(包括应答),如果未收到应答消息，则需要等待至少10mS再发送下一帧数据

## 重发机制

发送端发送完一条消息后，如果在规定时间内(10mS)未接收到接收端的应答消息，则会重发3次，直到接收到应答或者超出重发次数，此条消息不再发送

## 应答机制

接收端(PC机或者控制板)接收到数据后，需要在2mS后并且5mS内作出应答，否则发送端会认为发送失败，重新发送

## 数据校验

校验方式使用16位循环冗余校验(CRC16-Modbus)，低位在前，高位在后

计算方式参照附录**[CRC16-Modbus](#_CRC16-Modbus)**

参考资料：**[CRC16常见几个标准的算法及C语言实现](https://blog.csdn.net/leumber/article/details/54311811)**

## 通讯帧格式

|  |  |  |  |
| --- | --- | --- | --- |
| 起始符StartCode | 消息体Msg | 校验码CRC16 | 结束符EndCode |

### 说明

1. 起始符：一帧消息的起始标志，由0x7E表示
2. 消息体：消息内容，参考[**消息体内容**](#_消息体内容)
3. 校验码：消息体Msg全部数据校验，使用CRC16-Modbus校验，参考**[数据校验](#_数据校验)**
4. 结束符：一帧消息的结束标志，由0x7F表示

### 消息体内容

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 消息体Msg | | | | | | |
| 消息长度 | 消息类型 | 一级地址 | 二级地址 | 三级地址 | 数据0 | 数据N |
| Length | Cmd | Address1 | Address2 | Address3 | Data0 | Data…N |

#### 消息体说明

1. Length：整个消息体长度消息长度，不包含Length位，最大值5+128
2. Cmd：此条消息类型，

主机下发时，最高位为0，从机上传时，最高位为1；

写命令时，高第7位为0，读命令，高7位为1

Bit5~Bit0为消息类型号，最大支持63种类型，不支持0号消息，

参考[**消息类型**](#_消息类型)

1. Address1~ Address3：

Address1-一级地址，指柜地址(柜号)；

Address2-二级地址，指柜体内部设备连接结点地址(如层号、内部设备号)；

Address3-三级地址，批以二级地址为结点所连接的设备号(如位置号)；

如果PC要将消息发往指定目标，需要指定地址

例如：Address1=1，Address2=2，Address3=4(应答消息无地址)

当消息需要发送到Address3时，地址数据应该为1、2、4；

当消息只发送到Address2时，地址数据应该为1、2，Address3必须为0；

当消息只发送到Address1时，地址数据应该为1、0、0，Address2必须为0，Address3无效；

如果PC下发时消息只发往Address2，则Address1不得为0，Address3必须为0；

如果PC下发时消息发往Address3，则Address1、Address2都不得为0；

当下发地址为0xFF时，表示对所有同级地址发送

如果接收到底层消息需要上传时，则相应层级需要在相应地址位加入本地址

例如：Address1=2，Address2=3，Address3=5(应答消息无地址)

当Address3有消息需要上传到PC时，PC接收到的地址数据应该为2、3、5；

当Address2有消息需要上传到PC时，PC接收到的地址数据应该为2、3、0；

地址范围为1~63，拨码方式参照附录[**拨码表**](#_拨码表)

1. Data0~DataN：此条消息所带的数据(应答消息无数据段)

最大数据个数为128

### 消息类型

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| CMD消息类型 | | | | | | | |
| Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
| Bit7 | 数据传输方向   1. 上级往下传 2. 下级往上传 | | | | | | |
| Bit6 | 保留位 | | | | | | |
| Bit5:0 | 消息类型   1. 不支持类型 2. 应答类型，只带一个数据，不带地址，数据表示状态 3. LED控制，带三个数据，带地址，0-表示关，1-表示开 4. 开锁命令，只带一个数据，带地址，0-表示读锁状态，1表示开锁 5. 读卡器，带地址，第一个数据为0表示读卡，为1表示写卡，后面跟随卡数据 6. 层板供电控制 | | | | | | |

### 应答类型

应答类型无地址段和数据段，只对上层应答，只有一个状态码(7E 02 x1 xx xx xx 7F)

|  |  |
| --- | --- |
| 00 | 无错误 |
| 01 | 忙 |
| 02 | 不支持的消息类型/不识别 |
| 03 | 校验错误 |
|  |  |
|  |  |

## 消息传输流程

# 三、消息举例说明

## 应答类型举例

**应答类型无地址段和数据段，只对上层应答**

1. 主机应答无错误

|  |  |  |  |
| --- | --- | --- | --- |
| 7E 02 **01** 00 D1 90 7F | | | |
| StartCode | 7E | 起始符 |  |
| Length | 02 | 长度 | 消息段长度(不包含length) |
| Cmd | 01 | 命令 | 应答命令，上层下发最高位为0，上传时最高位为1 |
| Status | 00 | 状态码 | 00-无错误 |
| CRC16L | D1 | CRC低8位 | 从length开始计算length+1个字节 |
| CRC16H | 90 | CRC高8位 |  |
| EndCode | 7F | 结束符 |  |

1. 主机应答忙

|  |  |  |  |
| --- | --- | --- | --- |
| 7E 02 **01** 01 10 50 7F | | | |
| StartCode | 7E | 起始符 |  |
| Length | 02 | 长度 | 消息段长度(不包含length) |
| Cmd | 01 | 命令 | 应答命令，上层下发最高位为0，上传时最高位为1 |
| Status | 01 | 状态码 | 01-忙 |
| CRC16L | 10 | CRC低8位 | 从length开始计算length+1个字节 |
| CRC16H | 50 | CRC高8位 |  |
| EndCode | 7F | 结束符 |  |

1. 主机应答不支持的消息类型(未识别或无此功能)

|  |  |  |  |
| --- | --- | --- | --- |
| 7E 02 **01** 02 50 51 7F | | | |
| StartCode | 7E | 起始符 |  |
| Length | 02 | 长度 | 消息段长度(不包含length) |
| Cmd | 01 | 命令 | 应答命令，上层下发最高位为0，上传时最高位为1 |
| Status | 02 | 状态码 | 02-未识别或无此功能 |
| CRC16L | 50 | CRC低8位 | 从length开始计算length+1个字节 |
| CRC16H | 51 | CRC高8位 |  |
| EndCode | 7F | 结束符 |  |

1. 主机应答校验错误

|  |  |  |  |
| --- | --- | --- | --- |
| 7E 02 **01** 03 91 91 7F | | | |
| StartCode | 7E | 起始符 |  |
| Length | 02 | 长度 | 消息段长度(不包含length) |
| Cmd | 01 | 命令 | 应答命令，上层下发最高位为0，上传时最高位为1 |
| Status | 02 | 状态码 | 03-校验错误 |
| CRC16L | 91 | CRC低8位 | 从length开始计算length+1个字节 |
| CRC16H | 91 | CRC高8位 |  |
| EndCode | 7F | 结束符 |  |

1. 从机应答无错误

|  |  |  |  |
| --- | --- | --- | --- |
| 7E 02 **81** 00 B0 50 7F | | | |
| StartCode | 7E | 起始符 |  |
| Length | 02 | 长度 | 消息段长度(不包含length) |
| Cmd | 81 | 命令 | 应答命令，上层下发最高位为0，上传时最高位为1 |
| Status | 00 | 状态码 | 00-无错误 |
| CRC16L | B0 | CRC低8位 | 从length开始计算length+1个字节 |
| CRC16H | 50 | CRC高8位 |  |
| EndCode | 7F | 结束符 |  |

## 控制类消息举例

控制类消息需要有目标地址数据及数据段数据

### LED控制：

LED控制为开和关，命令为2，包含三字节地址，分别代表柜，层，位；带三个字节数据，按顺序分别代表红色、黄色/绿色、蓝色三种颜色的亮度值,对于单色灯，三个字节只要不为0此表示开灯

1. 柜1层1位2打开红色LED，黄色/绿色、蓝色关闭

|  |  |  |  |
| --- | --- | --- | --- |
| 7E 07 **02** 01 01 02 FF 00 00 00 A1 7F | | | |
| StartCode | 7E | 起始符 |  |
| Length | 07 | 长度 | 消息段长度(不包含length) |
| Cmd | 02 | 命令 | 命令号及方向：LED控制命令为02 |
| Address1 | 01 | 柜号 | FF表示对所有柜发送 |
| Address2 | 01 | 层号 | 如果此为0，表示消息针对柜号,FF表示对当前柜所有层发送 |
| Address3 | 02 | 位号 | 如果此为0，表示消息针对层号,FF表示对当前层有位发送 |
| Data0 | FF | 数据0 | 红色亮度值：最高0xFF,0表示关闭 |
| Data1 | 00 | 数据1 | 黄色/绿色亮度值：最高0xFF,0表示关闭 |
| Data2 | 00 | 数据2 | 蓝色亮度值：最高0xFF,0表示关闭 |
| CRC16L | 00 | CRC低8位 | 从length开始计算length+1个字节 |
| CRC16H | A1 | CRC高8位 |  |
| EndCode | 7F | 结束符 |  |

1. 柜1层1位2关灯

|  |  |  |  |
| --- | --- | --- | --- |
| 7E 07 **02** 01 01 02 FF 00 00 1F 84 7F | | | |
| StartCode | 7E | 起始符 |  |
| Length | 07 | 长度 | 消息段长度(不包含length) |
| Cmd | 02 | 命令 | 命令号及方向：LED控制命令为02 |
| Address1 | 01 | 柜号 | FF表示对所有柜发送 |
| Address2 | 01 | 层号 | 如果此为0，表示消息针对柜号,FF表示对当前柜所有层发送 |
| Address3 | 02 | 位号 | 如果此为0，表示消息针对层号,FF表示对当前层有位发送 |
| Data0 | 00 | 数据0 | 红色亮度值：最高0xFF,0表示关闭 |
| Data1 | 00 | 数据1 | 黄色/绿色亮度值：最高0xFF,0表示关闭 |
| Data2 | 00 | 数据2 | 蓝色亮度值：最高0xFF,0表示关闭 |
| CRC16L | 1F | CRC低8位 | 从length开始计算length+1个字节 |
| CRC16H | 84 | CRC高8位 |  |
| EndCode | 7F | 结束符 |  |

1. 柜4、层2、位3打开黄色/绿色LED，红色、蓝色关闭

|  |  |  |  |
| --- | --- | --- | --- |
| 7E 07 **02** 04 02 03 00 FF 00 C0 A2 7F | | | |
| StartCode | 7E | 起始符 |  |
| Length | 07 | 长度 | 消息段长度(不包含length) |
| Cmd | 02 | 命令 | 命令号及方向：LED控制命令为02 |
| Address1 | 04 | 柜号 | FF表示对所有柜发送 |
| Address2 | 02 | 层号 | 如果此为0，表示消息针对柜号,FF表示对当前柜所有层发送 |
| Address3 | 03 | 位号 | 如果此为0，表示消息针对层号,FF表示对当前层有位发送 |
| Data0 | 00 | 数据0 | 红色亮度值：最高0xFF,0表示关闭 |
| Data1 | FF | 数据1 | 黄色/绿色亮度值：最高0xFF,0表示关闭 |
| Data2 | 00 | 数据2 | 蓝色亮度值：最高0xFF,0表示关闭 |
| CRC16L | C0 | CRC低8位 | 从length开始计算length+1个字节 |
| CRC16H | A2 | CRC高8位 |  |
| EndCode | 7F | 结束符 |  |

1. 柜4、层2、位3关灯

|  |  |  |  |
| --- | --- | --- | --- |
| 7E 07 **02** 04 02 03 00 00 00 5A 2D 7F | | | |
| StartCode | 7E | 起始符 |  |
| Length | 07 | 长度 | 消息段长度(不包含length) |
| Cmd | 02 | 命令 | 命令号及方向：LED控制命令为02 |
| Address1 | 04 | 柜号 | FF表示对所有柜发送 |
| Address2 | 02 | 层号 | 如果此为0，表示消息针对柜号,FF表示对当前柜所有层发送 |
| Address3 | 03 | 位号 | 如果此为0，表示消息针对层号,FF表示对当前层有位发送 |
| Data0 | 00 | 数据0 | 红色亮度值：最高0xFF,0表示关闭 |
| Data1 | FF | 数据1 | 黄色/绿色亮度值：最高0xFF,0表示关闭 |
| Data2 | 00 | 数据2 | 蓝色亮度值：最高0xFF,0表示关闭 |
| CRC16L | 5A | CRC低8位 | 从length开始计算length+1个字节 |
| CRC16H | 2D | CRC高8位 |  |
| EndCode | 7F | 结束符 |  |

1. 柜4开灯（柜灯为单色灯，只要数据区三个字节之和不为0表示开灯，为0表示关灯）

|  |  |  |  |
| --- | --- | --- | --- |
| 7E 07 **02** 04 00 00 01 00 00 72 69 7F | | | |
| StartCode | 7E | 起始符 |  |
| Length | 07 | 长度 | 消息段长度(不包含length) |
| Cmd | 02 | 命令 | 命令号及方向：LED控制命令为02 |
| Address1 | 04 | 柜号 | FF表示对所有柜发送 |
| Address2 | 00 | 层号 | 如果此为0，表示消息针对柜号,FF表示对当前柜所有层发送 |
| Address3 | 00 | 位号 | 如果此为0，表示消息针对层号,FF表示对当前层有位发送 |
| Data0 | 01 | 数据0 | 红色亮度值：最高0xFF,0表示关闭 |
| Data1 | 00 | 数据1 | 黄色/绿色亮度值：最高0xFF,0表示关闭 |
| Data2 | 00 | 数据2 | 蓝色亮度值：最高0xFF,0表示关闭 |
| CRC16L | 72 | CRC低8位 | 从length开始计算length+1个字节 |
| CRC16H | 69 | CRC高8位 |  |
| EndCode | 7F | 结束符 |  |

1. 柜4端口关灯（柜灯为单色灯，只要数据区三个字节之和不为0表示开灯，为0表示关灯）

|  |  |  |  |
| --- | --- | --- | --- |
| 7E 07 **02** 04 00 00 00 00 00 23 A9 7F | | | |
| StartCode | 7E | 起始符 |  |
| Length | 07 | 长度 | 消息段长度(不包含length) |
| Cmd | 02 | 命令 | 命令号及方向：LED控制命令为02 |
| Address1 | 04 | 柜号 | FF表示对所有柜发送 |
| Address2 | 00 | 层号 | 如果此为0，表示消息针对柜号,FF表示对当前柜所有层发送 |
| Address3 | 00 | 位号 | 如果此为0，表示消息针对层号,FF表示对当前层有位发送 |
| Data0 | 00 | 数据0 | 红色亮度值：最高0xFF,0表示关闭 |
| Data1 | 00 | 数据1 | 黄色/绿色亮度值：最高0xFF,0表示关闭 |
| Data2 | 00 | 数据2 | 蓝色亮度值：最高0xFF,0表示关闭 |
| CRC16L | 23 | CRC低8位 | 从length开始计算length+1个字节 |
| CRC16H | A9 | CRC高8位 |  |
| EndCode | 7F | 结束符 |  |

1. 柜4所有层及位亮蓝灯

|  |  |  |  |
| --- | --- | --- | --- |
| 7E 07 **02** 04 FF FF 00 00 FF 47 E9 7F | | | |
| StartCode | 7E | 起始符 |  |
| Length | 07 | 长度 | 消息段长度(不包含length) |
| Cmd | 02 | 命令 | 命令号及方向：LED控制命令为02 |
| Address1 | 04 | 柜号 | FF表示对所有柜发送 |
| Address2 | FF | 层号 | 如果此为0，表示消息针对柜号,FF表示对当前柜所有层发送 |
| Address3 | FF | 位号 | 如果此为0，表示消息针对层号,FF表示对当前层有位发送 |
| Data0 | 00 | 数据0 | 红色亮度值：最高0xFF,0表示关闭 |
| Data1 | 00 | 数据1 | 黄色/绿色亮度值：最高0xFF,0表示关闭 |
| Data2 | FF | 数据2 | 蓝色亮度值：最高0xFF,0表示关闭 |
| CRC16L | 47 | CRC低8位 | 从length开始计算length+1个字节 |
| CRC16H | E9 | CRC高8位 |  |
| EndCode | 7F | 结束符 |  |

1. 柜4所有层及位关灯

|  |  |  |  |
| --- | --- | --- | --- |
| 7E 07 **02** 04 FF FF 00 00 FF 07 A9 7F | | | |
| StartCode | 7E | 起始符 |  |
| Length | 07 | 长度 | 消息段长度(不包含length) |
| Cmd | 02 | 命令 | 命令号及方向：LED控制命令为02 |
| Address1 | 04 | 柜号 | FF表示对所有柜发送 |
| Address2 | FF | 层号 | 如果此为0，表示消息针对柜号,FF表示对当前柜所有层发送 |
| Address3 | FF | 位号 | 如果此为0，表示消息针对层号,FF表示对当前层有位发送 |
| Data0 | 00 | 数据0 | 红色亮度值：最高0xFF,0表示关闭 |
| Data1 | 00 | 数据1 | 黄色/绿色亮度值：最高0xFF,0表示关闭 |
| Data2 | 00 | 数据2 | 蓝色亮度值：最高0xFF,0表示关闭 |
| CRC16L | 07 | CRC低8位 | 从length开始计算length+1个字节 |
| CRC16H | A9 | CRC高8位 |  |
| EndCode | 7F | 结束符 |  |

### 锁控制

1. 柜4开锁

|  |  |  |  |
| --- | --- | --- | --- |
| 7E 05 **03** 04 00 00 00 45 7E 7F | | | |
| StartCode | 7E | 起始符 |  |
| Length | 05 | 长度 | 消息段长度(不包含length) |
| Cmd | 03 | 命令 | 命令号及方向：锁控制命令为03 |
| Address1 | 04 | 柜号 | FF表示对所有柜发送 |
| Address2 | 00 | 层号 | 开锁命令时无效 |
| Address3 | 00 | 位号 | 开锁命令时无效 |
| Data0 | 00 | 数据0 | 下发时0-开锁，1-关锁，上报时0-开状态，1-关状态 |
| CRC16L | 45 | CRC低8位 | 从length开始计算length+1个字节 |
| CRC16H | 7E | CRC高8位 |  |
| EndCode | 7F | 结束符 |  |

1. 柜4锁开状态上报

|  |  |  |  |
| --- | --- | --- | --- |
| 7E 05 **83** 04 00 00 00 44 A0 7F | | | |
| StartCode | 7E | 起始符 |  |
| Length | 05 | 长度 | 消息段长度(不包含length) |
| Cmd | 83 | 命令 | 命令号及方向：锁控制命令为03 |
| Address1 | 04 | 柜号 | FF表示对所有柜发送 |
| Address2 | 00 | 层号 | 开锁命令时无效 |
| Address3 | 00 | 位号 | 开锁命令时无效 |
| Data0 | 00 | 数据0 | 下发时0-开锁，1-关锁，上报时0-开状态，1-关状态 |
| CRC16L | 44 | CRC低8位 | 从length开始计算length+1个字节 |
| CRC16H | A0 | CRC高8位 |  |
| EndCode | 7F | 结束符 |  |

1. 柜4锁关状态上报

|  |  |  |  |
| --- | --- | --- | --- |
| 7E 05 **83** 04 00 00 01 85 60 7F | | | |
| StartCode | 7E | 起始符 |  |
| Length | 05 | 长度 | 消息段长度(不包含length) |
| Cmd | 83 | 命令 | 命令号及方向：锁控制命令为03 |
| Address1 | 04 | 柜号 | FF表示对所有柜发送 |
| Address2 | 00 | 层号 | 开锁命令时无效 |
| Address3 | 00 | 位号 | 开锁命令时无效 |
| Data0 | 01 | 数据0 | 下发时0-开锁，1-关锁，上报时0-开状态，1-关状态 |
| CRC16L | 44 | CRC低8位 | 从length开始计算length+1个字节 |
| CRC16H | A0 | CRC高8位 |  |
| EndCode | 7F | 结束符 |  |

### 层板供电控制

**地址只有一个字节，为柜地址，数据位只有一位，为开关状态**

**7E 03 05 XX XX XX XX 7F**

1. 关闭1柜层板供电(针对整柜)

|  |  |  |  |
| --- | --- | --- | --- |
| 7E 03 **05** 01 00 11 F1 7F | | | |
| StartCode | 7E | 起始符 |  |
| Length | 03 | 长度 | 消息段长度(不包含length) |
| Cmd | 05 | 命令 | 命令号及方向：锁控制命令为03 |
| Address1 | 01 | 柜号 | FF表示对所有柜发送 |
| Data0 | 00 | 数据0 | 下发时0-关电，1-通电，上报时0-关电状态，1-通电状态 |
| CRC16L | 11 | CRC低8位 | 从length开始计算length+1个字节 |
| CRC16H | F1 | CRC高8位 |  |
| EndCode | 7F | 结束符 |  |

1. 打开1柜层板供电(针对整柜)

|  |  |  |  |
| --- | --- | --- | --- |
| 7E 03 **05** 01 01 D0 31 7F | | | |
| StartCode | 7E | 起始符 |  |
| Length | 03 | 长度 | 消息段长度(不包含length) |
| Cmd | 05 | 命令 | 命令号及方向：锁控制命令为03 |
| Address1 | 01 | 柜号 | FF表示对所有柜发送 |
| Data0 | 01 | 数据0 | 下发时0-关电，1-通电，上报时0-关电状态，1-通电状态 |
| CRC16L | D0 | CRC低8位 | 从length开始计算length+1个字节 |
| CRC16H | 31 | CRC高8位 |  |
| EndCode | 7F | 结束符 |  |

## 数据类消息举例

### 读卡器命令

1. 向2号柜读卡器发送数据

|  |  |  |  |
| --- | --- | --- | --- |
| 7E 05 **83** 02 00 00 data0~dataN xx xx 7F | | | |
| StartCode | 7E | 起始符 |  |
| Length | xx | 长度 | 消息段长度(不包含length), length=4+N+1,N为要发往读卡器的全部数据长度 |
| Cmd | 04 | 命令 | 命令号及方向：读卡器相关命令为04 |
| Address1 | 02 | 柜号 | FF表示对所有柜发送 |
| Address2 | 00 | 层号 | 如果此为0，表示消息针对柜号,FF表示对当前柜所有层发送 |
| Address3 | 00 | 位号 | 如果此为0，表示消息针对柜号,FF表示对当前柜所有层发送 |
| Data0 | xx |  | 根据读卡器协议发送相关数据 |
| ~~ | xx |  |  |
| DataN | xx |  |  |
| CRC16L | xx | CRC低8位 | 从length开始计算length+1个字节 |
| CRC16H | xx | CRC高8位 |  |
| EndCode | 7F | 结束符 |  |

1. 2号柜读卡器上传数据

|  |  |  |  |
| --- | --- | --- | --- |
| 7E 05 **83** 02 00 00 data0~dataN xx xx 7F | | | |
| StartCode | 7E | 起始符 |  |
| Length | xx | 长度 | 消息段长度(不包含length), length=4+N+1,N为从读卡器接收到的的全部数据长度 |
| Cmd | 84 | 命令 | 命令号及方向：读卡器相关命令为04 |
| Address1 | 02 | 柜号 | FF表示对所有柜发送 |
| Address2 | 00 | 层号 | 如果此为0，表示消息针对柜号,FF表示对当前柜所有层发送 |
| Address3 | 00 | 位号 | 如果此为0，表示消息针对柜号,FF表示对当前柜所有层发送 |
| Data0 | xx |  | 根据读卡器协议发送相关数据 |
| ~~ | xx |  |  |
| DataN | xx |  |  |
| CRC16L | xx | CRC低8位 | 从length开始计算length+1个字节 |
| CRC16H | xx | CRC高8位 |  |
| EndCode | 7F | 结束符 |  |

# 四、附录

## CRC16-Modbus

void InvertUint8(unsigned char \*dBuf,unsigned char \*srcBuf)

{

int i;

unsigned char tmp[4];

tmp[0] = 0;

for(i=0;i< 8;i++)

{

if(srcBuf[0]& (1 << i))

tmp[0]|=1<<(7-i);

}

dBuf[0] = tmp[0];

}

void InvertUint16(unsigned short \*dBuf,unsigned short \*srcBuf)

{

int i;

unsigned short tmp[4];

tmp[0] = 0;

for(i=0;i< 16;i++)

{

if(srcBuf[0]& (1 << i))

tmp[0]|=1<<(15 - i);

}

dBuf[0] = tmp[0];

}

unsigned short CRC16\_MODBUS(unsigned char \*puchMsg, unsigned int usDataLen)

{

unsigned short wCRCin = 0xFFFF;

unsigned short wCPoly = 0x8005;

unsigned char wChar = 0;

while (usDataLen--)

{

wChar = \*(puchMsg++);

InvertUint8(&wChar,&wChar);

wCRCin ^= (wChar << 8);

for(int i = 0;i < 8;i++)

{

if(wCRCin & 0x8000)

wCRCin = (wCRCin << 1) ^ wCPoly;

else

wCRCin = wCRCin << 1;

}

}

InvertUint16(&wCRCin,&wCRCin);

return (wCRCin) ;

}

## 拨码表及说明

说明：1号位和2号位为特殊功位

LED板当未拨码时，LED会闪烁

当柜控制板未拨码时，背光灯会闪烁

|  |  |
| --- | --- |
| 12345678 | 参数设定 |
| 1xxxxxxx | 主柜拨码设定 |
| 0xxxxxxx | 副柜拨码设定 |
| x0xxxxxx | 读卡器接口波特率选择19200 |
| x1xxxxxx | 读卡器接口波特率选择9600 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ID | 12345678 | ID | 12345678 | ID | 12345678 | ID | 12345678 |
| 0 | xx000000 | 16 | xx010000 | 32 | xx100000 | 48 | xx110000 |
| 1 | xx000001 | 17 | xx010001 | 33 | xx100001 | 49 | xx110001 |
| 2 | xx000010 | 18 | xx010010 | 34 | xx100010 | 50 | xx110010 |
| 3 | xx000011 | 19 | xx010011 | 35 | xx100011 | 51 | xx110011 |
| 4 | xx000100 | 20 | xx010100 | 36 | xx100100 | 52 | xx110100 |
| 5 | xx000101 | 21 | xx010101 | 37 | xx100101 | 53 | xx110101 |
| 6 | xx000110 | 22 | xx010110 | 38 | xx100110 | 54 | xx110110 |
| 7 | xx000111 | 23 | xx010111 | 39 | xx100111 | 55 | xx110111 |
| 8 | xx001000 | 24 | xx011000 | 40 | xx101000 | 56 | xx111000 |
| 9 | xx001001 | 25 | xx011001 | 41 | xx101001 | 57 | xx111001 |
| 10 | xx001010 | 26 | xx011010 | 42 | xx101010 | 58 | xx111010 |
| 11 | xx001011 | 27 | xx011011 | 43 | xx101011 | 59 | xx111011 |
| 12 | xx001100 | 28 | xx011100 | 44 | xx101100 | 60 | xx111100 |
| 13 | xx001101 | 29 | xx011101 | 45 | xx101101 | 61 | xx111101 |
| 14 | xx001110 | 30 | xx011110 | 46 | xx101110 | 62 | xx111110 |
| 15 | xx001111 | 31 | xx011111 | 47 | xx101111 | 63 | xx111111 |