



ZZ-IO1600 继电器产品说明书

文件版本：Ver3.0

使

用

手

册

济南智泽贸易有限公司

目录

1. 功能简介.....	3
1.1. 产品特点.....	3
1.2. 产品功能.....	3
1.3. 检测参数.....	3
1.4. RS485 级联接线.....	4
2. 输入输出接线.....	5
2.1. 继电器接线说明.....	5
2.1.1 一般接线:	5
2.1.2 交流 220v 设备接线.....	5
2.1.3 交流 380v 设备接线.....	6
2.1.4 直流 30V 一下设备接线.....	6
2.1.5 不带零线交流 380V 设备接线.....	7
3. 测试软件说明.....	8
3.1. 软件界面.....	8
3.2. 通讯测试.....	8
4. 工作模式功能及设置.....	10
4.1. 设备地址.....	10
4.1.1 设备地址的介绍.....	10
4.1.2 拨码开关地址.....	10
4.1.3 地址的设定与读取.....	10
设备地址的读取.....	10
4.1.4 波特率的读取与设置.....	11
4.2. 闪开闪断功能及设置.....	11
4.2.1 闪开闪断功能介绍.....	11
4.2.2 闪断闪开的设置.....	11
5. 开发资料说明.....	13
5.1. 通讯协议说明.....	13
5.2. Modbus 寄存器说明.....	13
5.3. 指令生成.....	14
5.4. 指令列表.....	15
5.5. 指令详解.....	16
5.5.1 继电器输出.....	16
5.5.2 继电器状态.....	16
5.5.3 闪开闪闭指令.....	17
5.5.4 全开全关指令.....	18
6. 常见问题与解决方法.....	19
7. 更新历史.....	19
附录: CRC 校验函数.....	20

1. 功能简介

1.1. 产品特点

- DC5V、7-20V、24V 供电可选；
- 继电器输出触点隔离；
- 通讯接口支持 RS485；
- 通信波特率：2400,4800,9600,19200,38400（可以通过软件修改，默认 9600）；
- 通信协议：支持标准 modbus RTU 协议；
- 可以设置 0-255 个设备地址，5 位地址拨码开关可以设置 1-31 地址码，大于 31 的可以通过软件设置；
- 具有闪开、闪断功能，可以在指令里边带参数、操作继电器开一段时间自动关闭；
- 具有频闪功能，可以控制器继电器周期性开关。

1.2. 产品功能

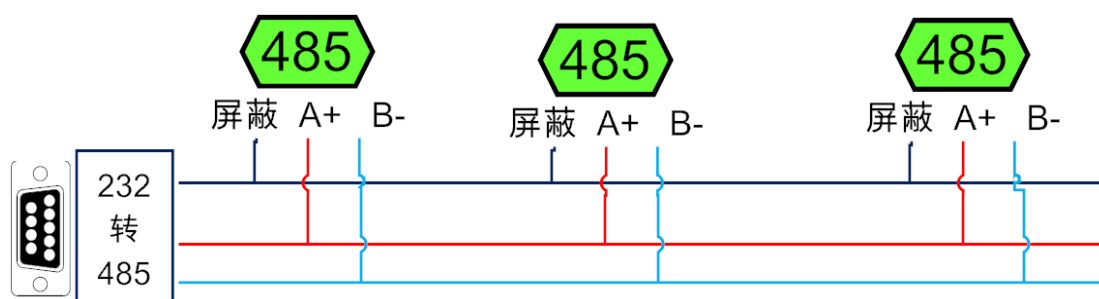
- 16 路继电器控制；
- 支持电脑软件手动控制。

1.3. 检测参数

参数	说明
触点容量	10A/30VDC 10A/250VAC
耐久性	10万次
数据接口	RS485
额定电压	DC5V； 7-20V； 24V 供电可选
电源指示	1路红色 LED 指示
输出指示	2路红色 LED 指示
工作温度	工业级，-40℃~85℃
尺寸	145*94*41mm
重量	330g
默认通讯格式	9600, n, 8, 1

波特率	2400,4800,9600,19200,38400
软件支持	配套配置软件、控制软件; 支持各家组态软件; 支持 Labviewd 等

1.4.RS485 级联接线

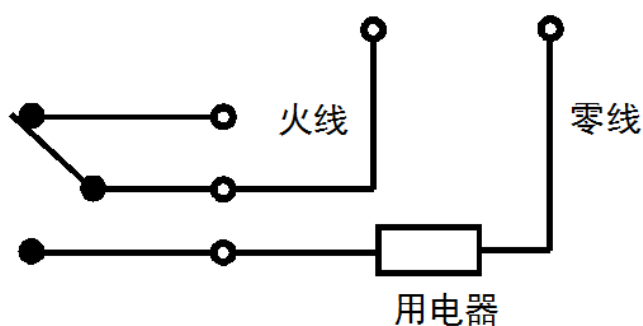


电脑自带的串口一般是 RS232，需要配 232-485 转换器（工业环境建议使用有源带隔离的转换器），或者直接用 USB 转 485。转换后 RS485 为 A、B 两线，A 接板上 A 端子，B 接板上 B 端子，485 屏蔽可以接 GND。若设备比较多建议采用双绞屏蔽线，采用链型网络结构。

2. 输入输出接线

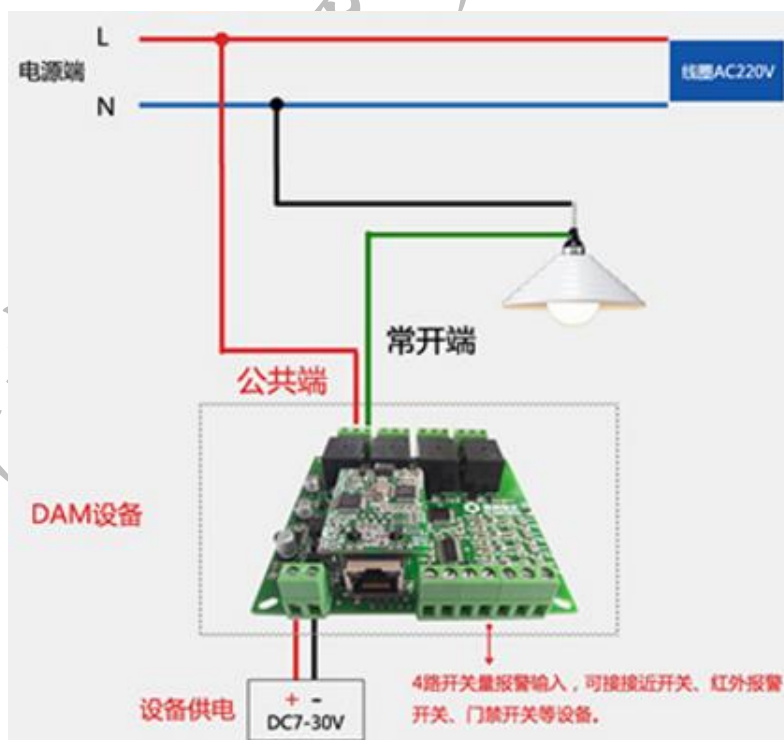
2.1. 继电器接线说明

2.1.1 一般接线：

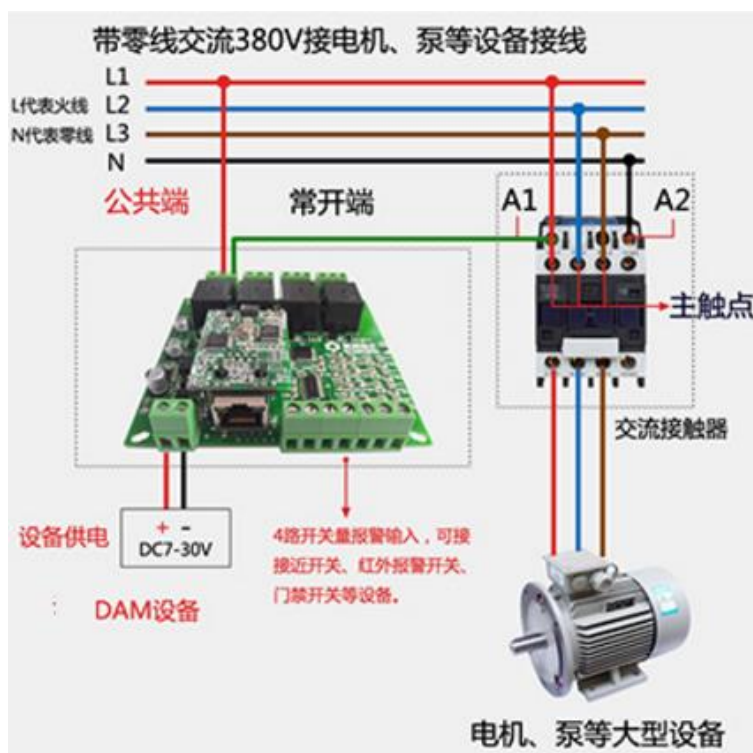


继电器输出触点容量最大为 AC250V 10A，单刀双掷。

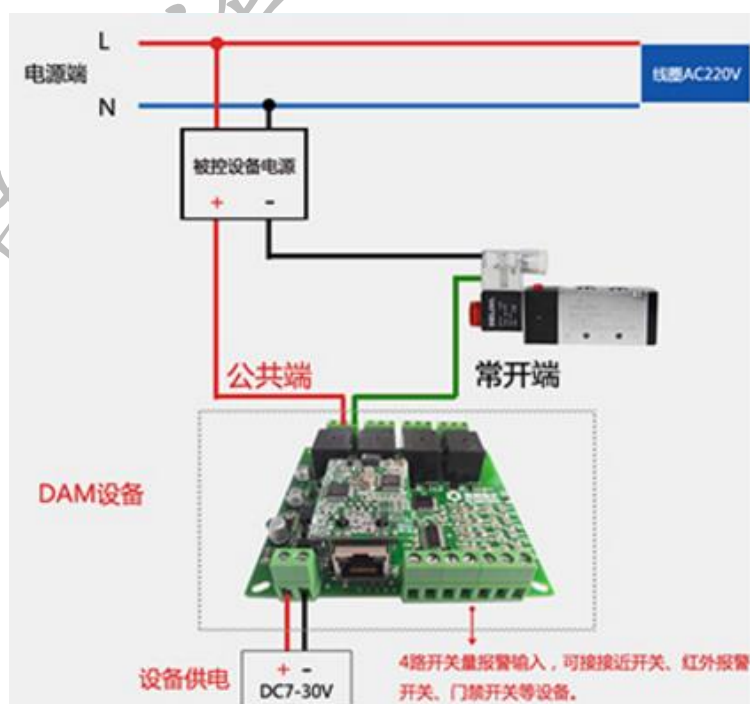
2.1.2 交流 220v 设备接线



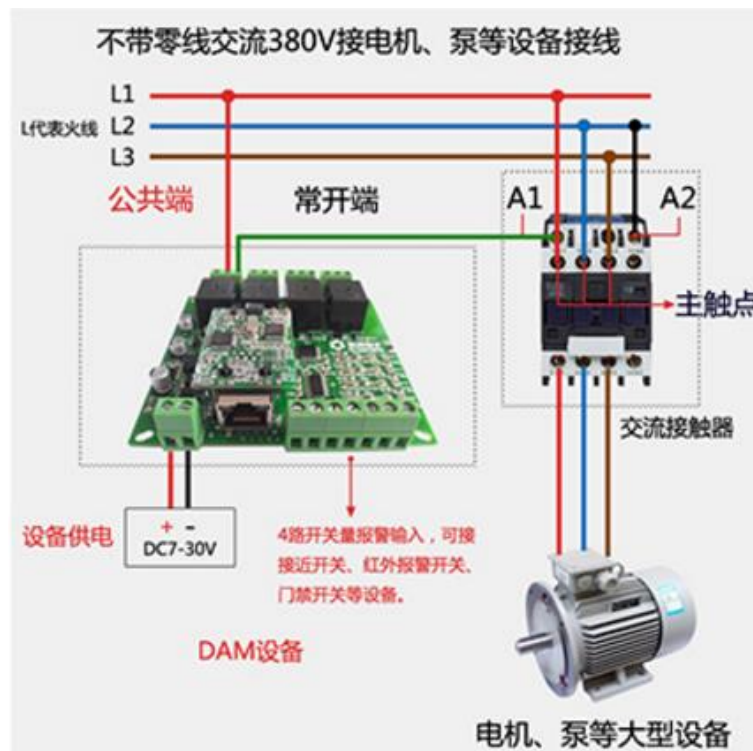
2.1.3 交流 380v 设备接线



2.1.4 直流 30V 一下设备接线

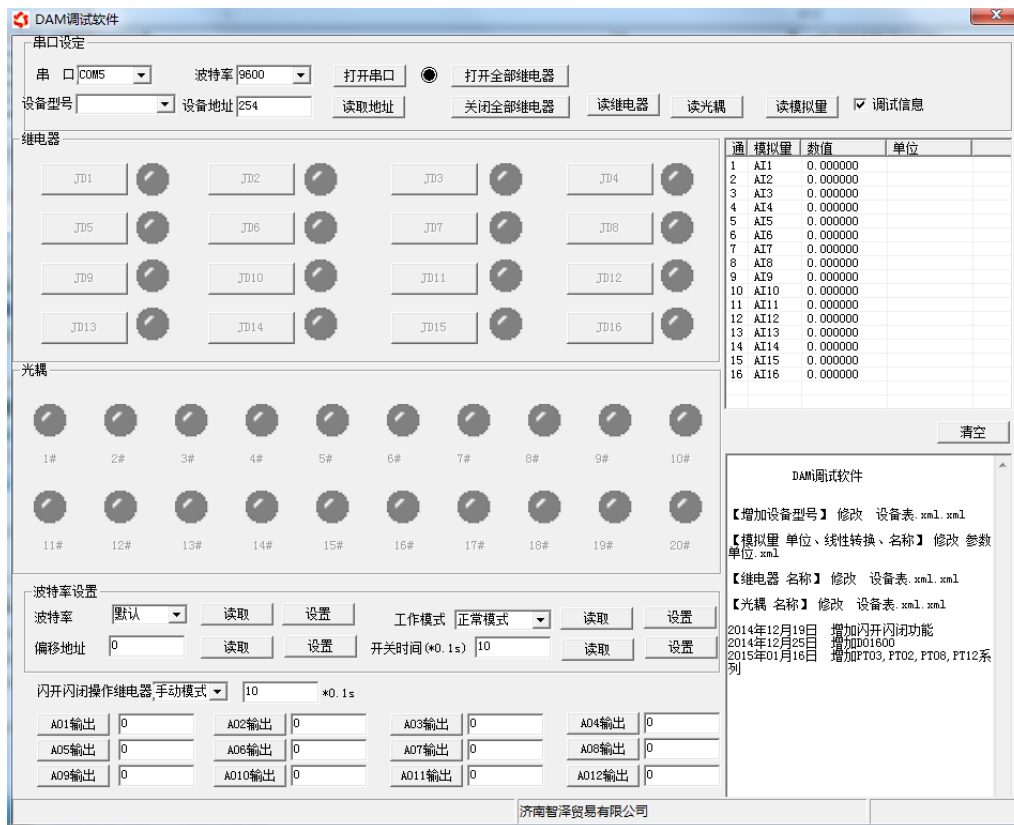


2.1.5 不带零线交流 380V 设备接线



3. 测试软件说明

3.1. 软件界面

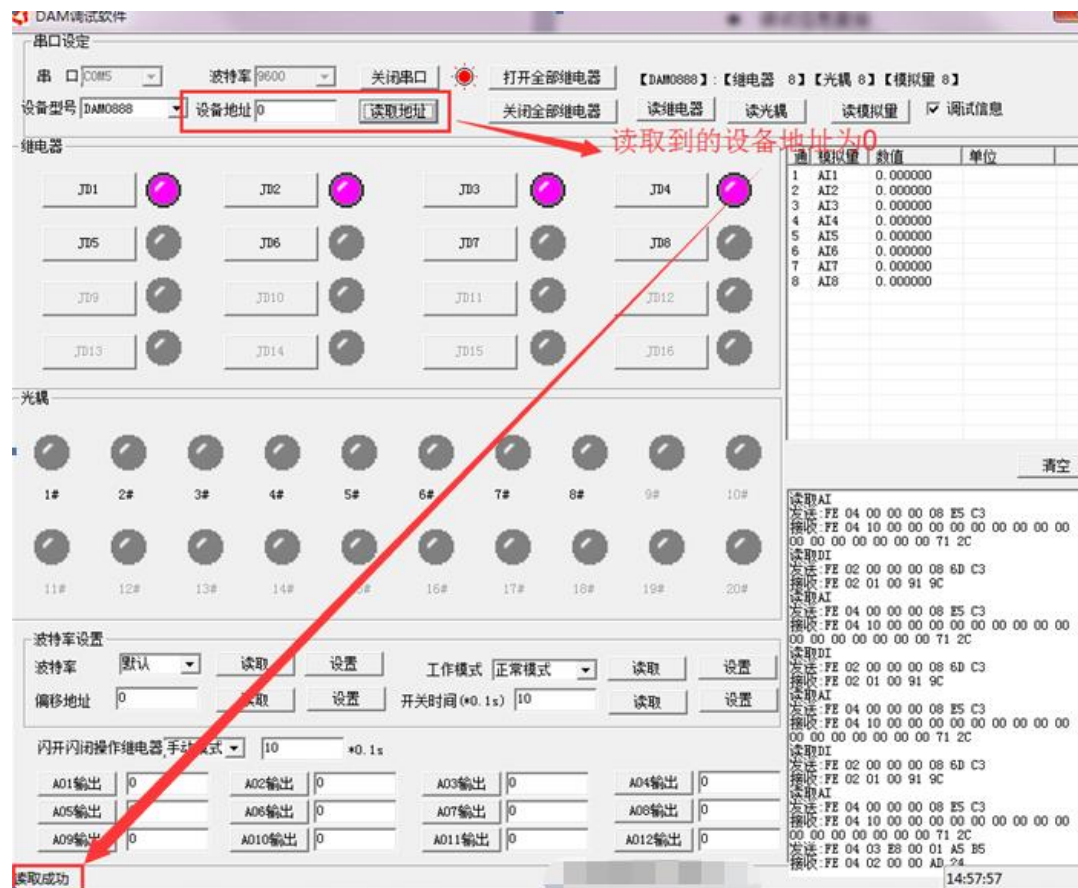


软件功能：

- 继电器状态查询
- 继电器独立控制
- 模拟量读取
- 开关量状态查询
- 调试信息查询
- 工作模式的更改
- 偏移地址的设定
- 继电器整体控制

3.2. 通讯测试

- ① 选择设备当前串口号，打开串口；
- ② 选择对应的产品型号；
- ③ 设备地址修改为 254，点击“读取地址”，软件底部提示“读取成功”，读到的设备地址为“0”，软件右下方的发送和指令正确，则说明设备与电脑通讯成功。



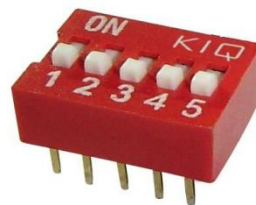
4. 工作模式功能及设置

4.1. 设备地址

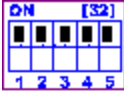
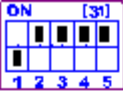
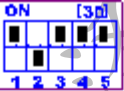
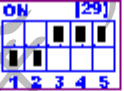
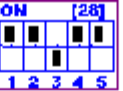
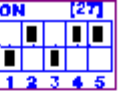
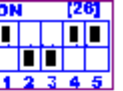
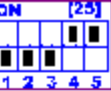
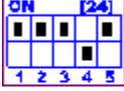
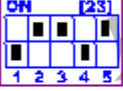

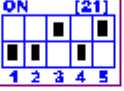
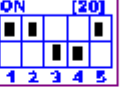
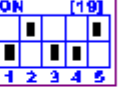
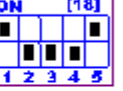
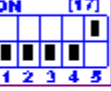
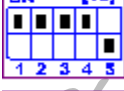

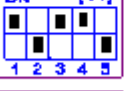
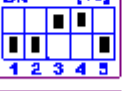
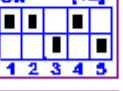
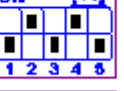
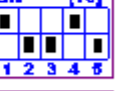
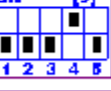



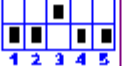
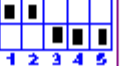
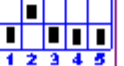


4.1.1 设备地址的介绍

ZZ-DAM 系列设备地址默认为 0，使用广播地址为 254 进行通讯，*用 0 无法通讯*。
设备地址=拨码开关地址+偏移地址。

4.1.2 拨码开关地址



- 1、五个拨码全都拨到“ON”位置时，为地址“32”；
- 2、五个拨码全都拨到“OFF”位置时，为地址“1”；
- 3、最左边 1 为二进制最低位。
- 4、地址表：

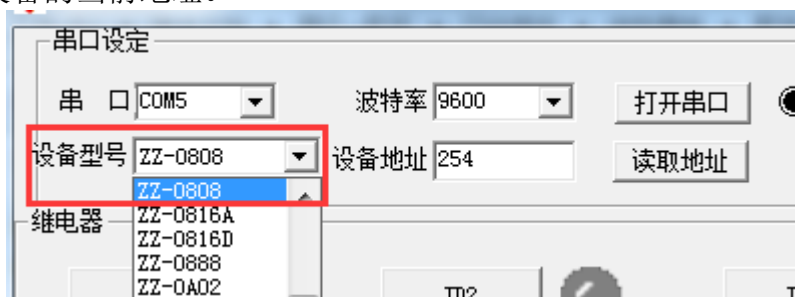
4.1.3 地址的设定与读取

设备地址的读取

点击调试软件下方偏移地址后边的“读取”或“设置”来对设备的偏移地址进行读取或设置。

设备正常通讯后，初始设备地址写入 254，然后点击软件上方“读取地址”

即可读到设备的当前地址。



4.1.4 波特率的读取与设置

点击下方波特率设置栏的“读取”和“设置”就可以分别读取和设置波特率和地址，操作后需要重启设备和修改电脑串口设置。



4.2. 闪开闪断功能及设置

4.2.1 闪开闪断功能介绍

手动模式：对继电器每操作一次，继电器则翻转一次（闭合时断开，断开时闭合）；
 闪开模式：对继电器每操作一次，继电器则闭合 1 秒（实际时间【单位秒】=设置数字*0.1）后自行断开；
 闪断模式：对继电器每操作一次，继电器则断开 1.秒（时间可调）后自行闭合；

4.2.2 闪断闪开的设置

打开“调试软件”点击继电器模式后面下拉箭头进行模式的选择。（后边时间可自行设置，实际时间=填写数字*0.1【单位秒】）

注：闪断闪开模式不能写入设备芯片内，软件上选择闪断闪开模式后，所有通道都为闪断闪开模式下，可通过发送单个通道的闪断闪开指令来进行单个通

道的控制，不影响其他通道的正常控制。

闪开闪闭操作继电器	
手动模式	10 *0.1s
A01输出	0
A02输出	0
A05输出	0
A06输出	0
A09输出	0
A10输出	0

济南智泽贸易有限公司

5. 开发资料说明

5.1. 通讯协议说明

本产品支持标准 modbus 指令，有关详细的指令生成与解析方式，可根据本文中的寄存器表结合参考《MODBUS 协议中文版》（产品相关下载里提供）即可。

本产品支持 modbus RTU 格式。

5.2. Modbus 寄存器说明

本控制卡主要为线圈寄存器，主要支持以下指令码：1、5、15

指令码	含义
1	读线圈寄存器
5	写单个线圈
15	写多个线圈寄存器

线圈寄存器地址表：

寄存器名称	寄存器地址	说明
线圈控制		
线圈 1	写线圈 1 号指令码	0x0001 第一路继电器输出
线圈 2		0x0002 第二路继电器输出
线圈 3		0x0003 第三路继电器输出
线圈 4		0x0004 第四路继电器输出
线圈 5		0x0005 第五路继电器输出
线圈 6		0x0006 第六路继电器输出
线圈 7		0x0007 第七路继电器输出
线圈 8		0x0008 第八路继电器输出
线圈 9		1x0009 第九路继电器输出
线圈 10		1x000A 第十路继电器输出
线圈 11		1x000B 第十一路继电器输出
线圈 12		1x000C 第十二路继电器输出
线圈 13		1x000D 第十三路继电器输出
线圈 14		1x000E 第十四路继电器输出
线圈 15		1x000F 第十五路继电器输出
线圈 16		1x0010 第十六路继电器输出
配置参数		

通信波特率	保持寄存器	4x1001	见下表波特率数值对应表, 默认为 0, 支持 0-5, 该寄存器同时决定 RS232 和 RS485 的通信波特率
备用		4x1002	备用, 用户不可写入任何值。
偏移地址		4x1003	设备地址=偏移地址+拨码开关地址
工作模式		4x1004	用户可以使用, 存储用户数据
延迟时间		4x1005	用户可以使用, 存储用户数据

备注:

①: Modbus 设备指令支持下列 Modbus 地址:

00001 至 09999 是离散输出(线圈)

10001 至 19999 是离散输入(触点)

30001 至 39999 是输入寄存器(通常是模拟量输入)

40001 至 49999 是保持寄存器(通常存储设备配置信息)

采用 5 位码格式, 第一个字符决定寄存器类型, 其余 4 个字符代表地址。地址 1 从 0 开始, 如 00001 对应 0000。

②: 波特率数值对应表

数值	波特率
0	38400
1	2400
2	4800
3	9600
4	19200
5	38400

③: 继电器状态, 通过 30002 地址可以查询, 也可以通过 00001---00002 地址来查询, 但控制只能使用 00001---00002 地址。

30002 地址数据长度为 16bit。最多可表示 16 个继电器。

对应结果如下:

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
继电器位置	8	7	6	5	4	3	2	1	16	15	14	13	12	11	10	9

即 寄存器 30009 数据 的 bit8 与寄存器 00001 的数据一样。

同理: 光耦输入也是如此。寄存器 30003 的 bit8、bit9 与寄存器 10001、10002 都对应到指定的硬件上。

寄存器地址按照 PLC 命名规则, 真实地址为去掉最高位, 然后减一。

5.3. 指令生成

应用举例及其说明: 本机地址除了偏移地址地址之外, 还有默认的 254 为广播地址。当总线上只有一个设备时, 无需关心拨码开关地址, 直接使用 254 地址

即可，当总线上有多个设备时通过拨码开关选择为不同地址，发送控制指令时通过地址区别。

注意：RS232 总线为 1 对 1 总线，即总线上只能有两个设备，例如电脑与继电器板卡，只有 485 总线才可以挂载多个设备。

指令可通过“调试软件”，的调试信息来获取。



指令生成说明：对于下表中没有的指令，用户可以自己根据 modbus 协议生成，对于继电器线圈的读写，实际就是对 modbus 寄存器中的线圈寄存器的读写，上文中已经说明了继电器寄存器的地址，用户只需生成对寄存器操作的读写指令即可。例如读或者写继电器 1 的状态，实际上是对继电器 1 对应的线圈寄存器 00001 的读写操作。

5.4. 指令列表

情景	RTU 格式（16 进制发送）
查询四路状态	FE 02 00 00 00 04 6D C6

查询指令返回信息	FE 01 01 00 61 9C
控制第一路开	FE 05 00 00 FF 00 98 35
控制返回信息	FE 05 00 00 FF 00 98 35
控制第一路关	FE 05 00 00 00 00 D9 C5
控制返回信息	FE 05 00 00 00 00 D9 C5
控制第二路开	FE 05 00 01 FF 00 C9 F5
控制第二路关	FE 05 00 01 00 00 88 05

5.5. 指令详解

5.5.1 继电器输出

控制 1 路继电器（以第一路开为例，其他通道参照本例）

发送码: FE 05 00 00 FF 00 98 35

字段	含义	备注
FE	设备地址	这里为广播地址
05	05 指令	单个控制指令
00 00	地址	要控制继电器寄存器地址
FF 00	指令	继电器开的动作
98 35	CRC16	前 6 字节数据的 CRC16 校验和

继电器卡返回信息:

返回码: FE 05 00 00 FF 00 98 35

字段	含义	备注
FE	设备地址	这里为广播地址
05	05 指令	单个控制指令
00 00	地址	要控制继电器寄存器地址
FF 00	指令	继电器开的动作
98 35	CRC16	前 6 字节数据的 CRC16 校验和

5.5.2 继电器状态

查询 2 路继电器

发送码: FE 01 00 00 00 02 A9 C4

字段	含义	备注
FE	设备地址	这里为广播地址
01	01 指令	查询继电器状态指令
00 00	起始地址	要查询的第一个继电器寄存器地址
00 02	查询数量	要查询的继电器数量
A9 C4	CRC16	前 6 字节数据的 CRC16 校验和

继电器卡返回信息:

返回码: FE 01 01 00 61 9C

字段	含义	备注
FE	设备地址	
01	01 指令	返回指令: 如果查询错误, 返回 0x81
01	字节数	返回状态信息的所有字节数。1+(n-1)/8
00	查询的状态	返回的继电器状态。 Bit0:第一个继电器状态 Bit1:第二个继电器状态 Bit7:第八个继电器状态
61 9C	CRC16	前 6 字节数据的 CRC16 校验和

5.5.3 闪开闪闭指令

闪开闪闭指令解析

闪开发送码: FE 10 00 03 00 02 04 00 04 00 0A 00 D8

闪断发送码: FE 10 00 03 00 02 04 00 02 00 14 21 62

字段	含义	备注
FE	设备地址	
10	10 指令	查询输入寄存器指令
00 03	继电器地址	要控制的器地址
00 02	控制命令数量	要对继电的命令个数
04	字节数	控制信息命令的的所有字节数。1+(n-1)/8
00 04 或 00 02	指令	00 04 为闪开指令 00 02 为闪闭命令
00 0A	间断时间	00 0A 为十六进制换为十进制则为 10 间隔时间为 (0.1 秒*10)
00 D8	CRC16	校验方式

返回码: FE 10 00 03 00 02 A5 C7

字段	含义	备注
FE	设备地址	
10	10 指令	返回指令: 如果查询错误, 返回 0x82
00 03	设备地址	查询设备的地址

00 02	接收命令数	设备接受的命令个数
A5 C7	CRC16	校验位

5.5.4 全开全关指令

全开全关指令解析

全开发送码: *FE 0F 00 00 00 02 01 FF D1 D3*

全断发送码: *FE 0F 00 00 00 02 01 00 91 93*

字段	含义	备注
FE	设备地址	
0F	0F 指令	返回指令: 如果查询错误, 返回 0x82
00 00	起始地址	
00 02	控制数量	控制的继电器数量
01	字节数	发送命令字节数
FF (或 00)	全开全关命令	FF 全开命令 00 全关命令
<i>D1 D3 (或 91 93)</i>	CRC16	校验位

全断全开返回码: *FE 0F 00 00 00 02 C0 05*

字段	含义	备注
FE	设备地址	
0F	0F 指令	返回指令: 如果查询错误, 返回 0x82
00 00	起始地址	
00 10	数量	返回信息的继电器数量
<i>C0 05</i>	CRC16	校验位

6. 常见问题与解决方法

①232 通讯，设备控制无响应，不动作

设备与上位机进行通信使用的是 232 直连线。即 RX 对 RX，TX 对 TX，GND 对 GND

②继电器只能开不能关

读取地址是否读到的是实际设备地址，调试信息栏内是否有返回指令，返回指令是否正确，如果读取地址失败，没有返回指令或返回指令异常，检查通讯线和通讯转换器

③485 总线上挂有多个设备时，每个设备地址不能一样，不能使用广播地址 254 来进行通讯。

广播地址在总线上只有一个设备时可以使用，大于 1 个设备时请以拨码开关区分地址来控制，否则会因为模块在通信数据的判断不同步上导致指令无法正确执行。

7. 更新历史

日期	更新内容	版本
2018/02/23	修改文档框架	V3.0

附录: CRC 校验函数

```
int CRC_Check(char *m_Data, short m_Size)
{
    int i0, i1;
    char CRC16Lo, CRC16Hi;    //CRC 寄存器
    char SaveHi, SaveLo;
    CRC16Lo = 0xFF;
    CRC16Hi = 0xFF;
    for(i0=0; i0<m_Size; i0++)
    {
        CRC16Lo = CRC16Lo ^ *(m_Data+i0); //每一个数据与 CRC 寄存器进行
        异或
        for(i1=0; i1<8; i1++)
        {
            SaveHi = CRC16Hi;
            SaveLo = CRC16Lo;
            CRC16Hi >>=1;                //高位右移一位
            CRC16Lo >>=1;                //低位右移一位
            if((SaveHi & 1) == 1) //如果高位字节最后一位为 1
            {
                CRC16Lo |=0x80; //则低位字节右移后前面补 1
            }
            if((SaveLo & 1) == 1) //如果 LSB 为 1, 则与多项式码进行异或
            {
                CRC16Hi ^=0xA0;
                CRC16Lo ^=1;
            }
        }
    }
    return ( CRC16Hi << 8 )| CRC16Lo;
}
```