



IBF70系列产品是基于单片机的智能监测和控制系统，所有的用户设定的地址，波特率，数据格式，校验和状态等配置信息都储存在非易失性存储器EEPROM里。

IBF70系列产品按工业标准设计、制造，信号输入 / 输出之间不隔离，抗干扰能力强，可靠性高。工作温度范围- 45℃~+85℃。

功能简介:

IBF70 远程I/O模块，可以用来测量四路开关量信号，并有四路继电器信号输出。

1、开关量信号输入与输出

4 路开关量信号输入，可接干接点和湿接点，详细请参考接线图部分；4 路 C 型继电器信号输出，有常开与常闭接点。

2、通讯协议

通讯接口： 1 路标准的 RS-485 通讯接口或 1 路标准的 RS-232 通讯接口，订货选型时注明。

通讯协议：支持两种协议，命令集定义的字符协议和 MODBUS RTU 通讯协议。模块自动识别通讯协议，能实现与多种品牌的 PLC、RTU 或计算机监控系统进行网络通讯。

数据格式：10 位。1 位起始位，8 位数据位，1 位停止位。

通讯地址（0~255）和波特率（2400、4800、9600、19200、38400、57600 、115200bps）均可设定；通讯网络最长距离可达 1200 米，通过双绞屏蔽电缆连接。

通讯接口高抗干扰设计，±15KV ESD 保护，通信响应时间小于 100mS。

3、抗干扰

可根据需要设置校验和。模块内部有瞬态抑制二极管，可以有效抑制各种浪涌脉冲，保护模块。

产品选型:

IBF70 - □

└── 通讯接口

485: 输出为 RS-485 接口

232: 输出为 RS-232 接口

选型举例 1: 型号: **IBF70 -232** 表示通讯接口为 RS-232

选型举例 2: 型号: **IBF70 -485** 表示通讯接口为 RS-485

IBF70通用参数:

(typical @ +25℃, Vs为24VDC)

输入类型: 开关量输入, 4 通道 (DI0~DI3)。

低电平: 输入 < 1V

高电平: 输入 4~30V

输入电阻: 3KΩ

输出类型: C 型继电器输出, 4 通道 (DO0~DO3)。有常开, 常闭和公共端。

触点负载能力: 1A 125VAC 或 2A 30VDC。

触点形式: 2Z

最大切换电压: 240VAC / 120VDC

最大切换电流: 2A

通 讯: 协议 RS-485 或 RS-232 标准字符协议 和 MODBUS RTU通讯协议

波特率 (2400、4800、9600、19200、38400、57600 、115200bps) 可软件选择

地址 (0~255) 可软件选择

通讯响应时间: 100 ms 最大



工作电源: +8 ~ 32VDC 宽供电范围, 内部有防反接和过压保护电路
 功率消耗: 小于0.5W
 工作温度: -45 ~ +80°C
 工作湿度: 10 ~ 90% (无凝露)
 存储温度: -45 ~ +80°C
 存储湿度: 10 ~ 95% (无凝露)
 隔离耐压: 四路输入互相隔离, 四路输出也互相隔离, 输入输出电源之间3隔离, 隔离电压 1500VAC
 外形尺寸: 120 mm x 70 mm x 43mm

引脚定义:

| 引脚 | 名称 | 描述 | 引脚 | 名称 | 描述 |
|----|-------|---------------|----|--------|------------|
| 1 | DI0+ | 通道0 开关量信号输入正端 | 14 | RL0NC | 继电器0 常闭输出端 |
| 2 | DI0- | 通道0 开关量信号输入负端 | 15 | RL0COM | 继电器0 公共输出端 |
| 3 | DI1+ | 通道1 开关量信号输入正端 | 16 | RL0NO | 继电器0 常开输出端 |
| 4 | DI1- | 通道1 开关量信号输入负端 | 17 | RL1NC | 继电器1 常闭输出端 |
| 5 | DI2+ | 通道2 开关量信号输入正端 | 18 | RL1COM | 继电器1 公共输出端 |
| 6 | DI2- | 通道2 开关量信号输入负端 | 19 | RL1NO | 继电器1 常开输出端 |
| 7 | DI3+ | 通道3 开关量信号输入正端 | 20 | NC | 空脚 |
| 8 | DI3- | 通道3 开关量信号输入负端 | 21 | RL2NC | 继电器2 常闭输出端 |
| 9 | NC | 空脚 | 22 | RL2COM | 继电器2 公共输出端 |
| 10 | DATA+ | RS-485 信号正端 | 23 | RL2NO | 继电器2 常开输出端 |
| 11 | DATA- | RS-485 信号负端 | 24 | RL3NC | 继电器3 常闭输出端 |
| 12 | PW+ | 电源正端 | 25 | RL3COM | 继电器3 公共输出端 |
| 13 | GND | 电源负端, 通讯地线 | 26 | RL3NO | 继电器3 常开输出端 |

表1 引脚定义

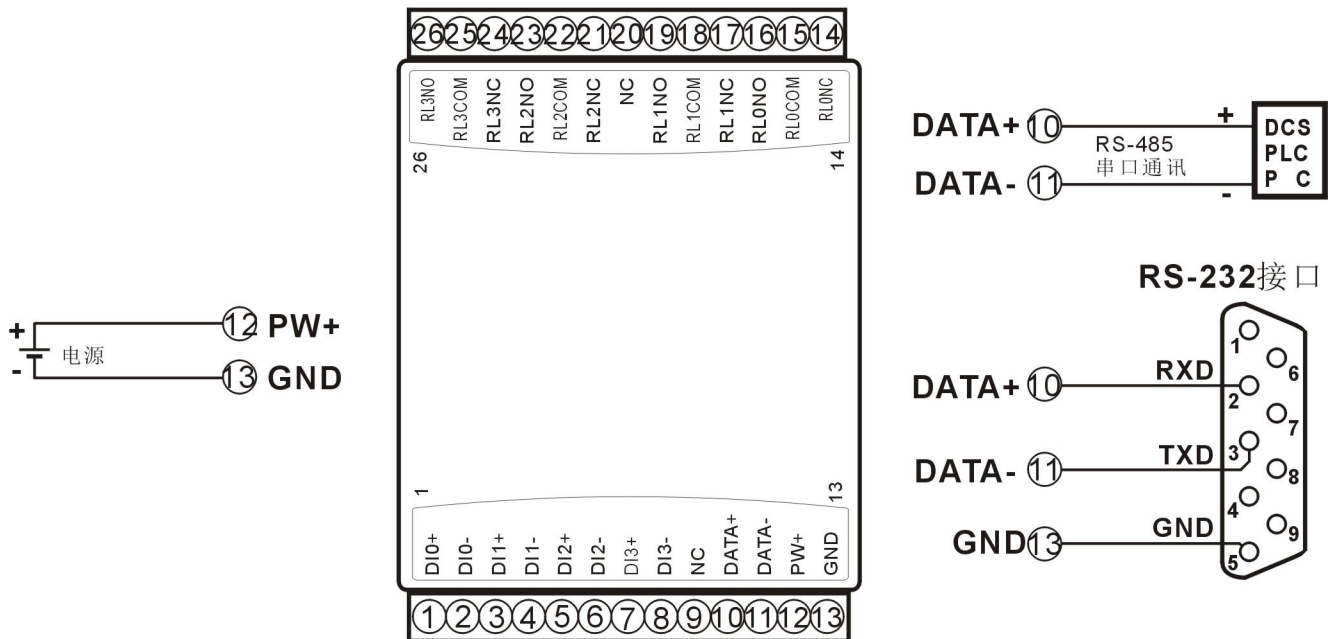
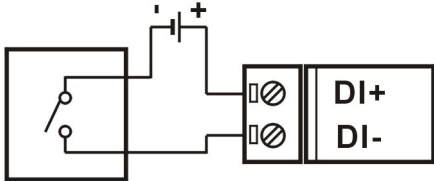
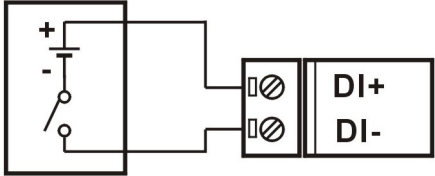
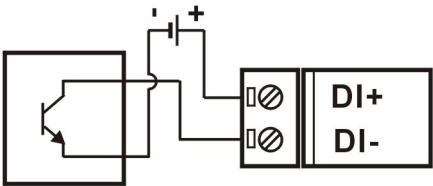
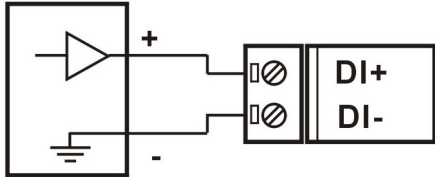
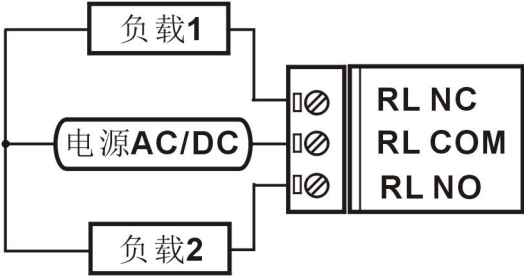


图3 IBF70 模块接线图

开关量信号输入接线图

| 干接点输入 (Dry contact) | 湿接点输入 (Wet contact) |
|--|--|
|  <p data-bbox="263 548 600 582">外接的电源可选 5 ~ 30VDC</p> |  |
| 集电极开路输入 (Open collector input) | TTL/CMOS 电平, 24V 电平输入 |
|  <p data-bbox="263 965 600 999">外接的电源可选 5 ~ 30VDC</p> |  |

开关量信号输出接线图

| | |
|---|---|
|  | <p data-bbox="863 1258 1394 1292">0: 继电器断开, 负载 1 工作, 负载 2 停止</p> <p data-bbox="863 1424 1394 1458">1: 继电器接通, 负载 1 停止, 负载 2 工作</p> |
|---|---|



IBF70 字符协议命令集:

模块的出厂初始设置, 如下所示:

地址代码为 01

波特率 9600 bps

禁止校验和

如果使用 RS-485 网络, 必须分配一个独一无二的地址代码, 地址代码取值为 16 进制数在 00 和 FF 之间, 由于新模块的地址代码都是一样的, 他们的地址将会和其他模块矛盾, 所以当你组建系统时, 你必须重新配置每一个 IBF70 模块地址。可以在接好 IBF70 模块电源线和 RS485 通讯线后, 通过配置命令来修改 IBF70 模块的地址。波特率, 校验和状态也需要根据用户的要求而调整。而在修改波特率, 校验和状态之前, 必须让模块先进入缺省状态, 否则无法修改。

让模块进入缺省状态的方法:

IBF70 模块边上都有一个 INIT 的开关, 在模块的侧面位置。将 INIT 开关拨到 INIT 位置, 再接通电源, 此时模块进入缺省状态。在这个状态时, 模块的配置如下:

地址代码为 00

波特率 9600 bps

禁止校验和

这时, 可以通过配置命令来修改 IBF70 模块的波特率, 校验和状态等参数。在不确定某个模块的具体配置时, 也可以将 INIT 开关拨到 INIT 位置, 使模块进入缺省状态, 再对模块进行重新配置。

注: 正常使用时请将 INIT 开关拨到 NORMAL 位置。

字符协议命令由一系列字符组成, 如首码、地址 ID, 变量、可选校验和字节和一个用以显示命令结束符(cr)。主机除了带通配符地址“**”的同步的命令之外, 一次只指挥一个 IBF70 模块。

命令格式: **(Leading Code)(Addr)(Command)[data][checksum](cr)**

- (Leading code)** 首码是命令中的第一个字母。所有命令都需要一个命令首码, 如%,\$,#,@,...等。 1- 字符
- (Addr)** 模块的地址代码, 如果下面没有指定, 取值范围从 00~FF (十六进制)。 2- 字符
- (Command)** 显示的是命令代码或变量值。 变量长度
- [data]** 一些输出命令需要的数据。 变量长度
- [checksum]** 括号中的Checksum (校验和) 显示的是可选参数, 只有在启用校验和时, 才需要此选项。 2- 字符
- (cr)** 识别用的一个控制代码符, (cr)作为回车结束符, 它的值为0x0D。 1- 字符

当启用校验和(checksum)时, 就需要[Checksum]。它占2-字符。命令和应答都必须附加校验和特性。校验和用来检查所有输入命令, 来帮助你发现主机到模块命令错误和模块到主机响应的错误。校验和字符放置在命令或响应字符之后, 回车符之前。

计算方法: 两个字符, 十六进制数, 为之前所发所有字符的ASCII码数值之和, 然后与十六进制数0xFF相与所得。

应用举例: 禁止校验和(checksum)

用户命令 **\$002(cr)**

模块应答 **!00020600 (cr)**

启用校验和(checksum)

用户命令 **\$002B6 (cr)**

模块应答 **!00020600 A9 (cr)**

'\$' = 0x24 '0' = 0x30 '2' = 0x32

B6=(0x24+0x30+0x30+0x32) AND 0xFF

'!' = 0x21 '0' = 0x30 '2' = 0x32 '6' = 0x36

A9=(0x21+0x30+0x30+0x30+0x32+0x30+0x36+0x30+0x30) AND 0xFF

命令的应答 :

应答信息取决于各种各样的命令。应答也由几个字符组成, 包括首代码, 变量和结束标识符。应答信号的首



代码有两种，‘!’或‘>’表示有效的命令而‘?’则代表无效。通过检查应答信息，可以监测命令是否有效

- 注意：1、在一些情况下，许多命令用相同的命令格式。要确保你用的地址在一个命令中是正确的，假如你用错误的地址，而这个地址代表着另一个模块，那么命令会在另一个模块生效，因此产生错误。
- 2、必须用大写字母输入命令。
- 3、(cr)代表键盘上的回车符，不要直接写出来，应该是敲一下回车键（Enter 键）。

1、读取开关状态命令

说明：从模块中读回所有输出通道开关量状态和输入通道开关量状态。

命令格式：**\$AA6(cr)**

参数说明：**\$** 分界符。十六进制为 24H

AA 模块地址，取值范围 00~FF(十六进制)。出厂地址为01，转换成十六进制为每个字符的ASCII码。如地址01换成十六进制为30H和31H。

(cr) 结束符，上位机回车键，十六进制为 0DH。

应答格式：**!(dataOutput)(dataInput)00(cr)** 命令有效。

?AA(cr) 命令无效或非法操作。

参数说明：**!** 分界符。十六进制为 21H

(dataOutput) 代表读取到的输出开关状态，两个 16 进制数，

第一个数为 0

第二个数代表 3~0 通道

位值为 0:

输出继电器断开

位值为 1:

输出继电器接通

十六进制为每个字符的 ASCII 码。

| | | | | | | | |
|------------|-------|-------|-------|-------|------|-------|-------|
| 0 | 0 | 0 | 0 | DO3 | DO2 | DO1 | DO0 |
| Bit7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit2 | Bit 1 | Bit 0 |
| dataOutput | | | | | | | |

(dataInput) 代表读取到的开关状态，两个 16 进制数，

第一个数为 0

第二个数代表 3~0 通道

位值为 0: 输入为低电平

位值为 1: 输入为高电平。

十六进制为每个字符的 ASCII 码。

| | | | | | | | |
|-----------|-------|-------|-------|-------|------|-------|-------|
| 0 | 0 | 0 | 0 | DI3 | DI2 | DI1 | DI0 |
| Bit7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit2 | Bit 1 | Bit 0 |
| dataInput | | | | | | | |

? 分界符，表示命令无效。

AA 代表输入模块地址

(cr) 结束符，上位机回车键，十六进制为 0DH。

其他说明：假如格式错误或通讯错误或地址不存在，模块不响应。

如果你使用的串口通讯软件输入不了回车键字符，请切换到十六进制格式进行通讯。

应用举例： 用户命令（字符格式） **\$016(cr)**

（十六进制格式） **243031360D**

模块应答（字符格式） **!020100 (cr)**

（十六进制格式）：**213032303130300D**

说明：读取到输出的数据是 02，转成 2 进制是 0000 0010，那么地址 01H 模块上输出开关状态是：

通道 0: 继电器断开 通道 1: 继电器接通 通道 2: 继电器断开 通道 3: 继电器断开

读取到输入的数据是 01，转成 2 进制是 0000 0001，那么地址 01H 模块上输入开关状态是：

通道 0: 高电平 通道 1: 低电平 通道 2: 低电平 通道 3: 低电平

2、设置继电器输出命令

说明：设置所有输出通道继电器状态。

命令格式：**#AABB(data) (cr)**

参数说明：**#** 分界符。十六进制为 24H

AA 模块地址，取值范围 00~FF(十六进制)。出厂地址为01，转换成十六进制为每个字符的ASCII码。如地址01换成十六进制为30H和31H。

BB 通道选择，可选择全部输出通道或单个输出通道。设置BB为00，则表示对全部输出通道进行设置。如对单个通道进行设置，则第一个字符B必须设置为1，第二个字符B可设为0-3，代表4个继电器DO输出通道。如果设置BB为FF，则表示设置全部通道的上电输出值。

(data) 输出值。

- 1, 如果是对所有通道设置
(BB=00), (BB=FF, 上电输出)
则为两个16进制数,

第一个数必须为 0

第二个数代表3~0 通道

位值为 0:

设置输出继电器断开

位值为 1:

设置输出继电器接通

- 2, 如果是对单个通道设置 (BB=1X, X表示要设定的通道), 则只能设置为00或01,

00: 设置X通道输出继电器断开

01: 设置X通道输出继电器接通

(cr) 结束符，上位机回车键，十六进制为 0DH。

应答格式：**>(cr)** 命令有效。

?AA(cr) 命令无效或非法操作。

参数说明：**>** 分界符。十六进制为 3EH。

? 分界符，表示命令无效。

AA 代表输入模块地址

(cr) 结束符，上位机回车键，十六进制为 0DH。

其他说明：假如格式错误或通讯错误或地址不存在，模块不响应。

如果你使用的串口通讯软件输入不了回车键字符，请切换到十六进制格式进行通讯。

应用举例 1: 用户命令 (字符格式) **#010002(cr)**

(十六进制格式) **233031303030320D**

模块应答 (字符格式) **>(cr)**

(十六进制格式): **3E0D**

说明：模块地址 01H，设置所有通道 (BB=00) 的输出为 02H，转成 2 进制是 0000 0010，那么地址 01H 模块上输出的开关状态是：

通道 0: 继电器断开 通道 1: 继电器接通 通道 2: 继电器断开 通道 3: 继电器断开

应用举例 2: 用户命令 (字符格式) **#011201(cr)**

(十六进制格式) **233031313230310D**

模块应答 (字符格式) **>(cr)**

(十六进制格式): **3E0D**

说明：模块地址 01H，设置通道 2 的继电器接通。

| | | | | | | | |
|------------|-------|-------|-------|-------|------|-------|-------|
| 0 | 0 | 0 | 0 | DO3 | DO2 | DO1 | DO0 |
| Bit7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit2 | Bit 1 | Bit 0 |
| dataOutput | | | | | | | |

3、配置 IBF70 模块命令

说明：对一个 IBF70 模块设置地址，波特率，校验和状态。配置信息储存在非易失性存储器 EEPROM 里。

命令格式：**%AANNTTCCFF(cr)**

参数说明：**%** 分界符。

AA 模块地址，取值范围 00~FF(十六进制)。出厂地址为 01，转换成十六进制为每个字符的 ASCII 码。如地址 01 换成十六进制为 30H 和 31H。

NN 代表新的模块 16 进制地址，数值 NN 的范围从 00 到 FF。转换成十六进制为每个字符的 ASCII 码。如地址 18 换成十六进制为 31H 和 38H。

TT 用 16 进制代表类型编码。

IBF70 产品必须设置为 00。

CC 用 16 进制代表波特率编码。

| 波特率代码 | 波特率 |
|-------|-------------|
| 04 | 2400 baud |
| 05 | 4800 baud |
| 06 | 9600 baud |
| 07 | 19200 baud |
| 08 | 38400 baud |
| 09 | 57600 baud |
| 0A | 115200 baud |

表 2 波特率代码

FF 用 16 进制的 8 位代表数据格式，校验和。注意从 bits0 到 bits5 不用必须设置为零。

| Bit7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit2 | Bit 1 | Bit 0 |
|------|-------|-------|-------|-------|------|-------|-------|
|------|-------|-------|-------|-------|------|-------|-------|

表 3 数据格式，校验和代码

Bit7: 保留位，必须设置为零

Bit6: 校验和状态，为 0: 禁止； 为 1: 允许

Bit5-bit0: 不用，必须设置为零。

(cr) 结束符，上位机回车键，十六进制为 0DH。

应答格式：**!AA(cr)** 命令有效。

?AA(cr) 命令无效或非法操作，或在改变波特率或校验和前，没有将 INIT 开关拨到 INIT 位置。

参数说明：**!** 分界符，表示命令有效。

? 分界符，表示命令无效。

AA 代表输入模块地址

(cr) 结束符，上位机回车键，十六进制为 0DH。

其他说明：假如你第一次配置模块，AA=01H，NN 等于新的地址。假如重新配置模块改变地址、输入范围、数据格式，AA 等于当前已配置的地址，NN 等于当前的或新的地址。假如要重新配置模块改变波特率或校验和状态，则必须将 INIT 开关拨到 INIT 位置，使模块进入缺省状态，此时模块地址为 00H，即 AA=00H，NN 等于当前的或新的地址。

假如格式错误或通讯错误或地址不存在，模块不响应。

应用举例： 用户命令 **%0111000600(cr)**

模块应答 **!11(cr)**

说明：**%** 分界符。

01 表示你想配置的IBF70模块原始地址为01H。

- 11 表示新的模块 16 进制地址为 11H。
- 00 类型代码, IBF70 产品必须设置为 00。
- 06 表示波特率 9600 baud。
- 00 表示禁止校验和。

4、读配置状态命令

说明: 对指定一个 IBF70 模块读配置。

命令格式: **\$AA2(cr)**

参数说明: **\$** 分界符。

AA 模块地址, 取值范围 00~FF(十六进制)。

2 表示读配置状态命令

(cr) 结束符, 上位机回车键, 十六进制为 0DH。

应答格式: **!AATTCFF(cr)** 命令有效。

?AA(cr) 命令无效或非法操作。

参数说明: **!** 分界符。

AA 代表输入模块地址。

TT 代表类型编码。

CC 代表波特率编码。见表 2

FF 见表 3

(cr) 结束符, 上位机回车键, 十六进制为 0DH。

其他说明: 假如格式错误或通讯错误或地址不存在, 模块不响应。

应用举例: 用户命令 **\$302(cr)**

模块应答 **!30F0600(cr)**

说明: **!** 分界符。

30 表示IBF70模块地址为30H。

00 表示输入类型代码。

06 表示波特率 9600 baud。

00 表示禁止校验和。

5、读模块名称命令

说明: 对指定一个 IBF70 模块读模块名称。

命令格式: **\$AAM(cr)**

参数说明: **\$** 分界符。

AA 模块地址, 取值范围 00~FF(十六进制)。

M 表示读模块名称命令

(cr) 结束符, 上位机回车键, 十六进制为 0DH。

应答格式: **!AA(ModuleName)(cr)** 命令有效。

?AA(cr) 命令无效或非法操作

参数说明: **!** 分界符, 表示命令有效。

? 分界符, 表示命令无效。

AA 代表输入模块地址。

(ModuleName) 模块名称 IBF70

(cr) 结束符, 上位机回车键, 十六进制为 0DH。

其他说明: 假如格式错误或通讯错误或地址不存在, 模块不响应。

应用举例: 用户命令 **\$08M(cr)**

模块应答 **!08IBF70 (cr)**

说明: 在地址 08H 模块为 IBF70。



Modbus RTU 通讯协议:

模块的出厂初始设置, 如下所示:

Modbus 地址为 01

波特率 9600 bps

让模块进入缺省状态的方法:

IBF60模块边上都有一个INIT的开关, 在模块的侧面位置。将INIT开关拨到INIT位置, 再接通电源, 此时模块进入缺省状态。在这个状态时, 模块暂时恢复为默认的状态: 地址为01, 波特率为9600。在不确定某个模块的具体配置时, 用户可以查询地址和波特率的寄存器40201-40202, 得到模块的实际地址和波特率, 也可以跟据需要修改地址和波特率。

注: 正常使用时请将 INIT 开关拨到 NORMAL 位置。

支持Modbus RTU通讯协议, 命令格式按照标准Modbus RTU通讯协议。

支持**功能码01**, 读线圈状态。1表示高电平, 0表示低电平。地址0X

支持**功能码05**, 设置单个线圈。1表示三极管导通, 0表示三极管断开。地址0X

支持**功能码03** (读保持寄存器) 和**功能码06** (写单个寄存器), 地址4X



寄存器说明:

| 地址 0X (PLC) | 地址 (PC, DCS) | 数据内容 | 属性 | 数据说明 |
|-------------|--------------|---------|-----|---|
| 00001 | 0000 | 输出继电器 | 读/写 | 通道 0 的输出状态 |
| 00002 | 0001 | 输出继电器 | 读/写 | 通道 1 的输出状态 |
| 00003 | 0002 | 输出继电器 | 读/写 | 通道 2 的输出状态 |
| 00004 | 0003 | 输出继电器 | 读/写 | 通道 3 的输出状态 |
| | | | | |
| 00011 | 0010 | 继电器上电输出 | 读/写 | 通道 0 的上电输出状态 |
| 00012 | 0011 | 继电器上电输出 | 读/写 | 通道 1 的上电输出状态 |
| 00013 | 0012 | 继电器上电输出 | 读/写 | 通道 2 的上电输出状态 |
| 00014 | 0013 | 继电器上电输出 | 读/写 | 通道 3 的上电输出状态 |
| | | | | |
| 00033 | 0032 | 输入的开关量 | 只读 | 通道 0 的电平状态 |
| 00034 | 0033 | 输入的开关量 | 只读 | 通道 1 的电平状态 |
| 00035 | 0034 | 输入的开关量 | 只读 | 通道 2 的电平状态 |
| 00036 | 0035 | 输入的开关量 | 只读 | 通道 3 的电平状态 |
| | | | | |
| 地址 4X (PLC) | 地址 (PC, DCS) | 数据内容 | 属性 | 数据说明 |
| 40001 | 0000 | 输出继电器 | 读/写 | 0x0000~0x000F, 3~0 通道 |
| 40011 | 0010 | 输出继电器 | 读/写 | 0x0000~0x000F, 3~0 通道上电输出值 |
| 40033 | 0032 | 输入的开关量 | 读/写 | 0x0000~0x000F, 3~0 通道 |
| | | | | |
| 40201 | 0200 | 模块地址 | 读/写 | 整数, 重启后生效, 范围 0x0000-0x00FF |
| 40202 | 0201 | 波特率 | 读/写 | 整数, 重启后生效, 范围 0x0004-0x000A 0x0004 = 2400 bps, 0x0005 = 4800 bps 0x0006 = 9600 bps, 0x0007 = 19200 bps 0x0008 = 38400 bps, 0x0009 = 57600 bps 0x000A = 115200bps |
| | | | | |
| 40211 | 0210 | 模块名称 | 只读 | 高位: 0x00 低位: 0x70 |

表 5 Modbus Rtu 寄存器说明

**Modbus RTU 通讯协议应用举例：**

1, 支持Modbus RTU通讯协议**功能码01**（读线圈状态），命令格式按照标准Modbus RTU通讯协议。

通讯举例：假如模块地址为 01，以 16 进制发送：**010100000083DCC**，即可取得寄存器的数据。

| | | | | | | | |
|------|-------|--------|--------|--------|--------|----------|----------|
| 01 | 01 | 00 | 00 | 00 | 08 | 3D | CC |
| 模块地址 | 读线圈状态 | 线圈地址高位 | 线圈地址低位 | 线圈数量高位 | 线圈数量低位 | CRC 校验低位 | CRC 校验高位 |

假如模块回复：**010101031189** 即读到的数据为 0x03，最后一位换成 2 进制即 0011。

即表明现在输出继电器通道 2 和 3 断开，通道 1 和 0 接通。

| | | | | | | | |
|------|-------|--------|----|----------|----------|--|--|
| 01 | 01 | 01 | 03 | 11 | 89 | | |
| 模块地址 | 读线圈状态 | 数据的字节数 | 数据 | CRC 校验低位 | CRC 校验高位 | | |

2, 支持Modbus RTU通讯协议**功能码05**（设置单个线圈），命令格式按照标准Modbus RTU通讯协议。

通讯举例：假如模块地址为 01，以 16 进制发送：**01050000FF008C3A**，数据为 0xFF00 表示设置继电器接通。

如果数据为 0x0000 则表示断开继电器（命令：**010500000000CDCA**）

| | | | | | | | |
|------|--------|--------|--------|------|------|----------|----------|
| 01 | 05 | 00 | 00 | FF | 00 | 8C | 3A |
| 模块地址 | 设置单个线圈 | 线圈地址高位 | 线圈地址低位 | 数据高位 | 数据低位 | CRC 校验低位 | CRC 校验高位 |

假如模块回复：**01050000FF008C3A** 即设置成功

| | | | | | | | |
|------|--------|--------|--------|------|------|----------|----------|
| 01 | 05 | 00 | 00 | FF | 00 | 8C | 3A |
| 模块地址 | 设置单个线圈 | 线圈地址高位 | 线圈地址低位 | 数据高位 | 数据低位 | CRC 校验低位 | CRC 校验高位 |

3, 支持Modbus RTU通讯协议**功能码03**（读保持寄存器），命令格式按照标准Modbus RTU通讯协议。

通讯举例：假如模块地址为 01，以 16 进制发送：**01030000001840A**，即可取得寄存器的数据。

| | | | | | | | |
|------|--------|---------|---------|---------|---------|----------|----------|
| 01 | 03 | 00 | 00 | 00 | 01 | 84 | 0A |
| 模块地址 | 读保持寄存器 | 寄存器地址高位 | 寄存器地址低位 | 寄存器数量高位 | 寄存器数量低位 | CRC 校验低位 | CRC 校验高位 |

假如模块回复：**0103020003F845** 即读到的数据为 0x0003，最后一位换成 2 进制即 0011。

即表明现在输出继电器通道 2 和 3 断开，通道 1 和 0 接通。

| | | | | | | | |
|------|--------|--------|------|------|----------|----------|--|
| 01 | 03 | 02 | 00 | 03 | F8 | 45 | |
| 模块地址 | 读保持寄存器 | 数据的字节数 | 数据高位 | 数据低位 | CRC 校验低位 | CRC 校验高位 | |

4, 支持Modbus RTU通讯协议**功能码06**（写单个寄存器），命令格式按照标准Modbus RTU通讯协议。

通讯举例：假如模块地址为 01，以 16 进制发送：**0106000000FC9CE**，最后一位换成 2 进制为 1111，即输出继电器所有通道接通。

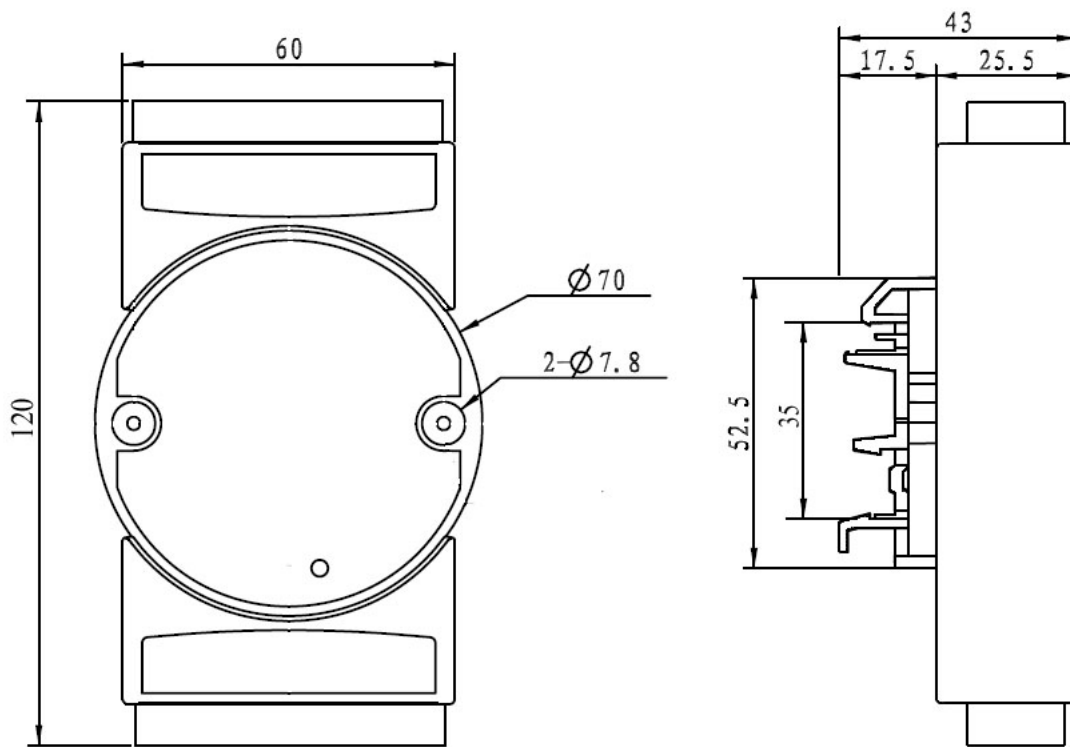
| | | | | | | | |
|------|--------|---------|---------|------|------|----------|----------|
| 01 | 06 | 00 | 00 | 00 | 0F | C9 | CE |
| 模块地址 | 写单个寄存器 | 寄存器地址高位 | 寄存器地址低位 | 数据高位 | 数据低位 | CRC 校验低位 | CRC 校验高位 |

假如模块回复：**0106000000FC9CE** 即设置成功

| | | | | | | | |
|------|--------|---------|---------|------|------|----------|----------|
| 01 | 06 | 00 | 00 | 00 | 0F | C9 | CE |
| 模块地址 | 写单个寄存器 | 寄存器地址高位 | 寄存器地址低位 | 数据高位 | 数据低位 | CRC 校验低位 | CRC 校验高位 |



外形尺寸: (单位: mm)



可以安装在标准 DIN35 导轨上

保修:

本产品自售出之日起两年内,凡用户遵守贮存、运输及使用要求,而产品质量低于技术指标的,可以返厂免费维修。因违反操作规定和要求而造成损坏的,需交纳器件费用和维修费。

版权:

版权 © 2018 深圳市贝福科技有限公司。

如未经许可,不得复制、分发、翻译或传输本说明书的任何部分。本说明书如有修改和更新,恕不另行通知。

商标:

本说明书提及的其他商标和版权归各自的所有人所有。

版本号: V1.1

日期: 2018 年 11 月