

# HCR 系列串行通信篇

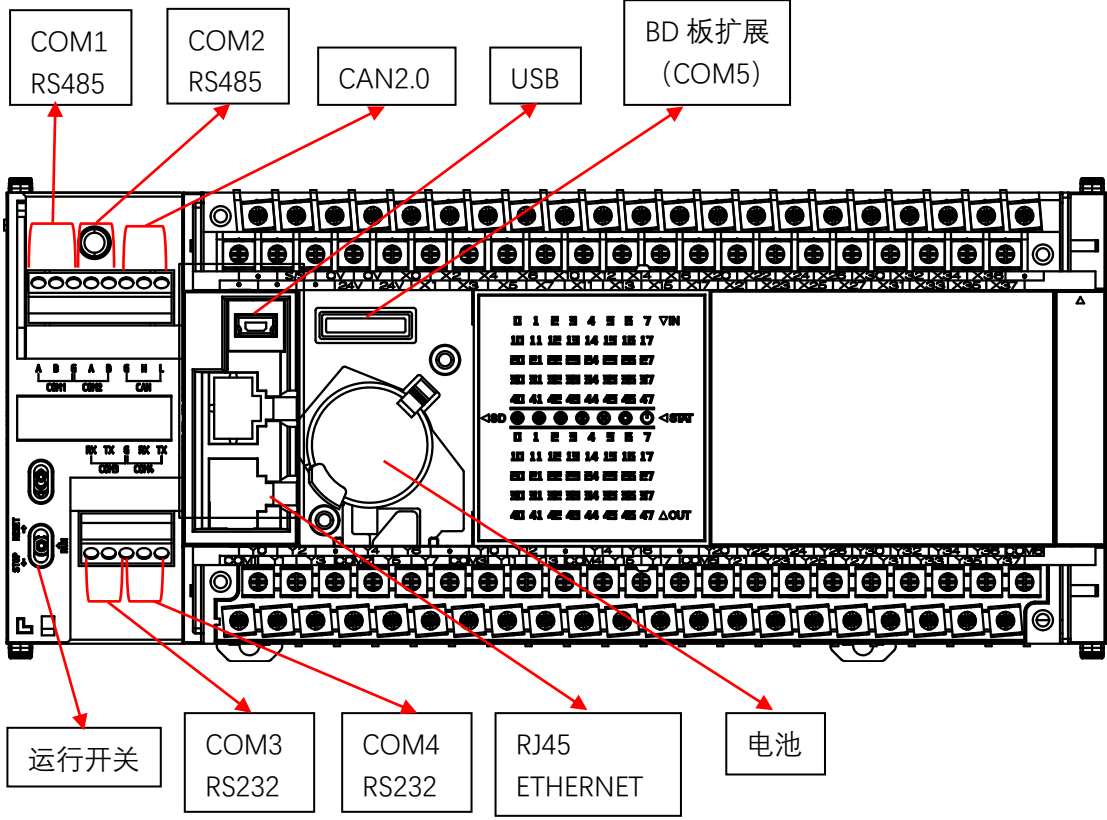
## 目录

HCR 系列串行通信篇 .....	1
1.R 系列各通信接口一览 .....	2
1.1 R 系列上的各通信接口一览 .....	2
1.2 各接口使用方法介绍 .....	3
1.3ETHERNET 接口使用方法与介绍 .....	5
1.4 COM1 接口使用方法与介绍 .....	8
1.5 COM2 接口使用方法与介绍 .....	13
1.6 COM3 口使用方法与介绍 .....	17
1.7 COM4 口使用方法与介绍 .....	19
1.8 COM5 口的使用方法与介绍 .....	22
2. R 系列的监控功能一览 .....	25
2.1 R 系列程序的转换与写入 .....	25
2.2 R 系列程序的读取 .....	26
2.3 R 系列程序的监控 .....	27
2.4 R 系列监控功能汇总 .....	28
3. 自由协议简介 .....	37
3.1 概要 .....	37
3.2 系统配置 .....	38
3.3 规格 .....	38
3.4.自由协议规则 .....	40
3.5.特殊软元件介绍 .....	40
3.6.自由通信指令格式 .....	43
3.7.应用举例 .....	49
4. MODBUS 简介 .....	49
4.1 概要 .....	49
4.2 系统配置 .....	50
4.3.规格 .....	51
4.4MODBUS 规格 .....	52
4.5 通信设置 .....	53
4.6 功能 .....	57
4.7 编程 .....	63
4.8 故障排除 .....	65
5. DIP 简介 .....	68
5.1 概要 .....	68
5.2 主要功能 .....	69
5.3 通信协议 .....	72
5.3.3 数据帧 .....	74

# 1.R 系列各通信接口一览

## 1.1 R 系列上的各通信接口一览

R 系列小型 PLC 目前支持的接口类型包含了 RS485、RS232、USB、ETHERNET 如下图所示，并且 COM5 口可使用扩展板进行通信接口的扩展。



注：各机型 COM5 位置不同，详见下表

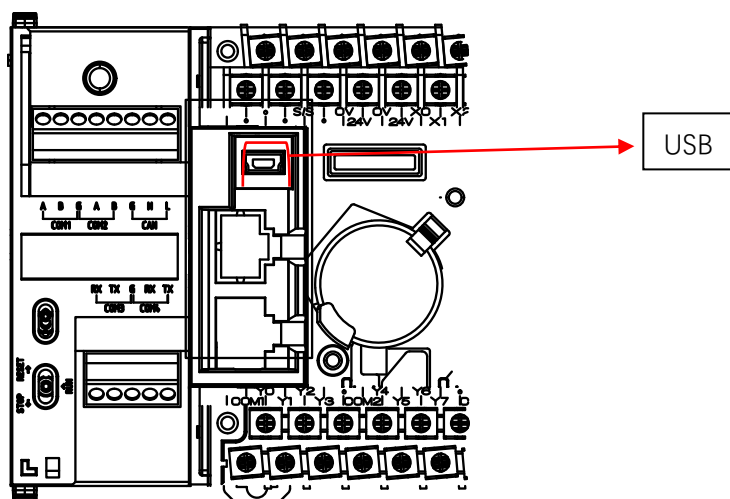
名称	接口类型	支持的协议类型	默认协议	备注
COM1	RS485	自由协议、MODBUS_RTU 主站、MODBUS_RTU 从站、不使用、DIP 从站	不使用	R1,R2 无隔离
COM2	RS485	自由协议、MODBUS_RTU 主站、MODBUS_RTU 从站、不使用、DIP 从站	不使用	R1,R2 为 BD 扩展
COM3	RS232	自由协议、下载/监控/（兼 DIP 从站）、不使用	下载/监控	R1,R2 COM3 COM4 共用圆形 8pin
COM4	RS232	自由协议、下载/监控/（兼 DIP	不使用	R1,R2 COM3

		从站)、不使用		COM4 共用圆形 8pin
COM5	扩展接口	自由协议、下载/监控/(兼 DIP 从站)、不使用	不使用	目前仅 HCA2P-422-BD,HCA2P-232-BD , HCRXB-C24-BD 支持用作 COM5 的通信扩展口 R1,R2 只支持 BD R8 BD 扩展为 COM5 R8C 只有左扩展为 COM5 R8P BD 和左扩展都支持
USB (mini-B)	通用串行总线	下载/监控、不使用	下载/监控	均支持
ETHERNET	以太网	下载/监控、MODBUS_TCP 从站, 不使用	不使用	均支持
CAN	CAN 总线	暂未完成		仅 R8P/R8C 支持

## 1.2 各接口使用方法介绍

### 1.2.1 USB 接口的使用与接线

R 系列小型 PLC 上的 USB 插座类型为 Mini B。使用标准的 Mini B 类型的插头的连接线, 连接 PLC 与安装了 HCP WORKS2 的 PC 设备时, 即可通过 HCP WORKS2 实现 PLC 程序的下载和监控功能。USB 在 R 系列 PLC 上位置图如下所示。

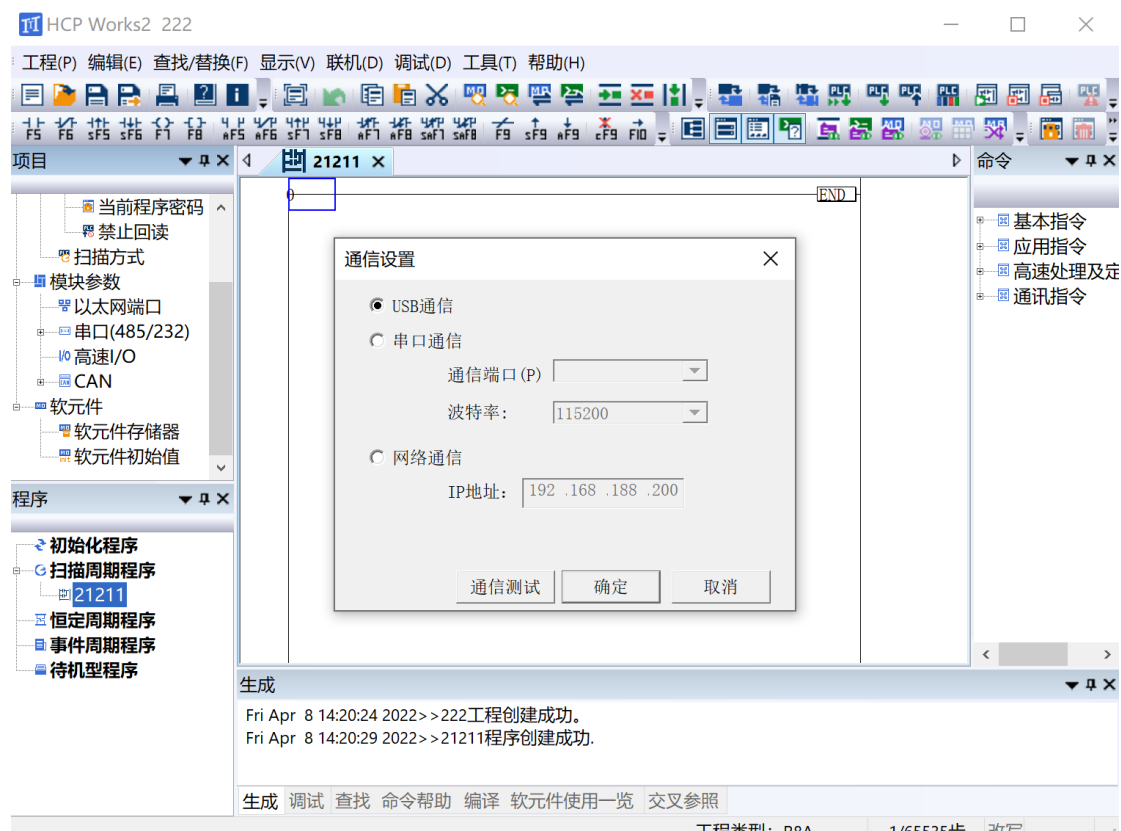


## 1.2.2 USB 接口使用设置

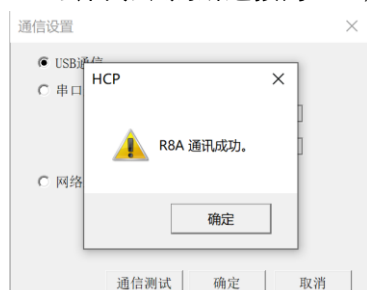
R 系列小型 PLC 的 USB 接口仅供下载/监控使用，使用的是禾川自定义的下载/监控协议，出厂时默认的通信方式是使用 USB 通信。如果设备修改了通信方式或者由于其他原因需要设置 USB 通信时，参照如下方法设置：

1. 用标准的 USB Mini B 型的线束连接 R 系列 PLC 和安装了 HCP WORKS2 的 PC 设备
2. 点击菜单栏中的设置按钮，选择通信设定，点击通信设定按钮后弹出通信设置界面
3. 选择 USB 通信后，点击通信测试按钮，通信正常时会弹出通信成功提示

注：部分 PC 设备的 USB 扩展口连接 R 系列 plc 时无法正常连接 R 系列 PLC，请使用原 USB 口连接 R 系列与 PC



IDE 会自动识别所连接的 PLC，当连接成功时会显示与相应的 PLC 通信成功

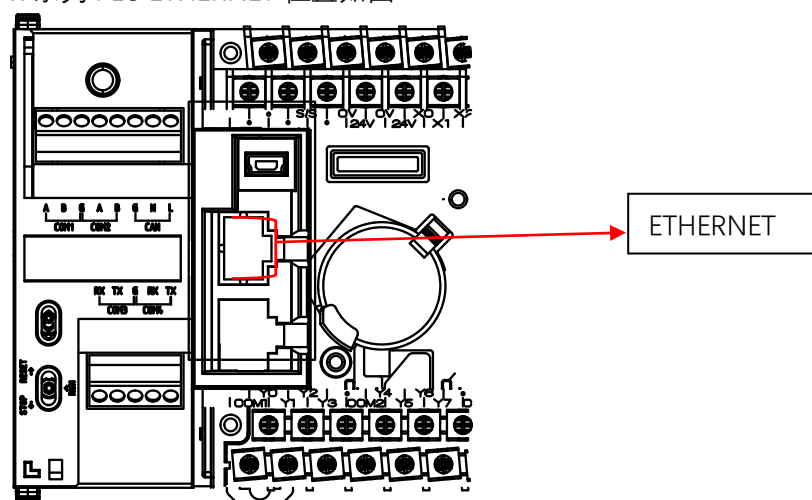


## 1.3ETHERNET 接口使用方法与介绍

### 1.3.1ETHERNET 接口的使用与接线

R 系列小型 PLC 上的 ETHERNET 接口类型为 RJ45 插座。使用标准的 RJ45 插头的连接线，连接 PLC 与安装了 HCP WORKS2 的 PC 设备时，并且设置了正确的 ip 地址后（见下小节），即可通过 HCP WORKS2 实现 PLC 程序的下载和监控功能。而当 R 系列使用 RJ45 类型连接线连接 HMI 设备或其它支持 MODBUS\_TCP 协议设备时，R 系列即可作为 MODBUS\_TCP 从站使用。

R 系列 PLC ETHERNET 位置如图



### 1.3.2 ETHERNET 接口用作监控/下载时的设置

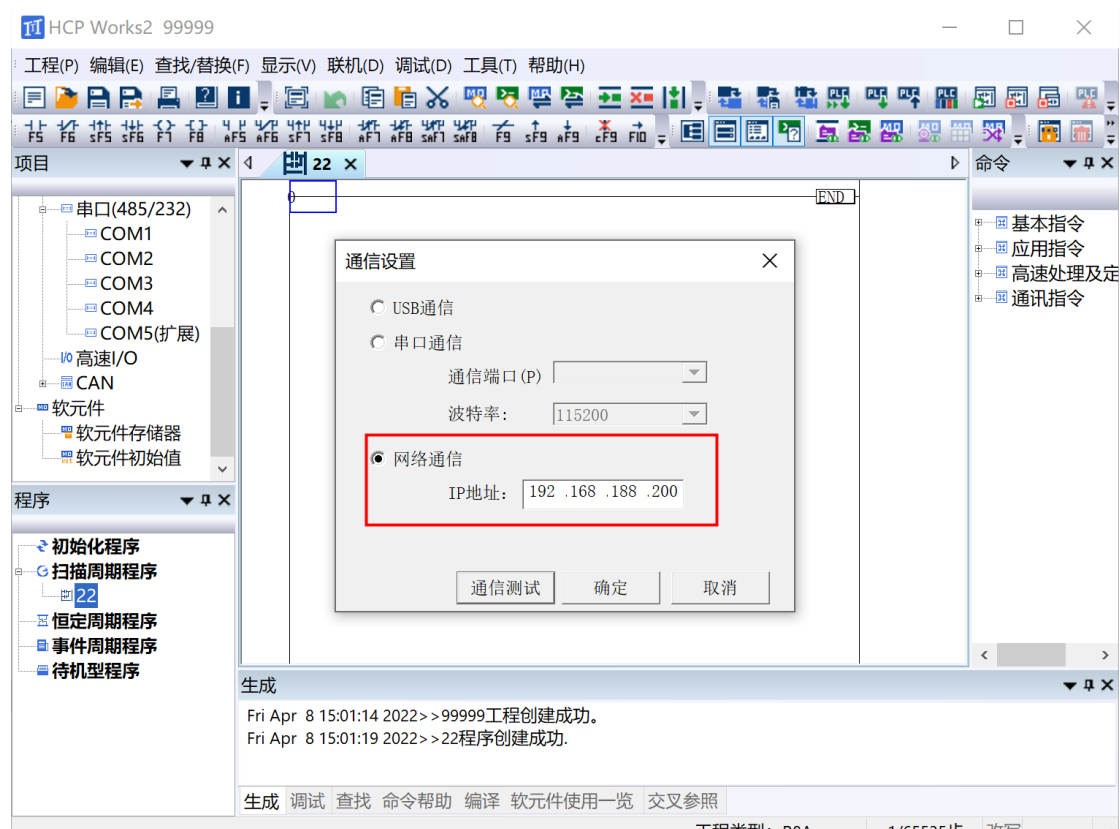
R 系列小型 PLC 的 ETHERNET 接口做下载/监控使用时，使用的是禾川自定义的下载/监控协议，出厂时默认的通信方式是使用 USB 通信。如果设备需要使用 ETHERNET 通信时，参照如下方法设置：

1. 用标准的 RJ45 型的线束连接 R 系列 PLC 和安装了 HCP WORKS2 的 PC 设备
2. 点击菜单栏中的设置按钮，选择通信设定，点击通信设定按钮后弹出通信设置界面
3. 选择网络通信选项，此时 R 系列默认 IP 地址是 192.168.188.200，用户的 PC 需要将 IP 地址 (TCP/IPV4) 中的地址修改成与 R 系列的默认 IP 地址的前三个字段一致，最后一个字段不一致（或者在已经通过其他方式连接了 PLC 的情况下，修改 PLC 的 IP 地址的前

三个网段与 PC 设备一致，最后一个网段不一致)

4. 点击通信测试按钮，通信正常时会弹出通信成功提示

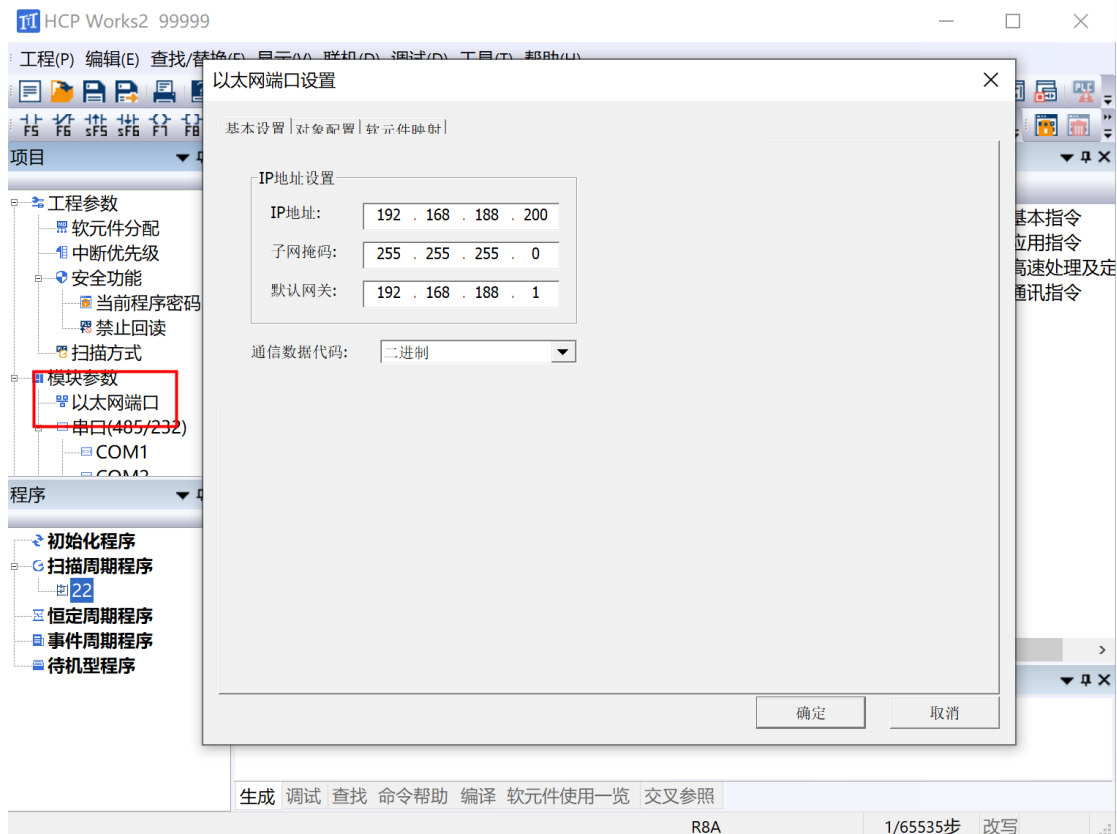
注：当弹出通信失败时，请检查线束是否连接正常，网口指示灯是否闪烁，或者设备 IP 地址是否匹配。



### 1.3.3 ETHERNET 接口用作 MODBUS\_TCP 从站时设置

R 系列小型 PLC 的 ETHERNET 接口做 MODBUS\_TCP 从站时,使用的是 MODBUS 从站协议，出厂时默认的通信方式是使用 USB 通信。如果设备需要使用 ETHERNET 通信时，参照如下方法设置：

1. 用标准的 RJ45 型的线束连接 R 系列 PLC 和支持做 MODBUS\_TCP 从站的设备
2. 点击项目栏中模块参数部分的以太网端口按钮，将 IP 地址设置为与连接设备匹配的 IP 地址，即前三个网段相同，最后网段不同。（当修改了此处的 IP 地址时，通信设定中的 IP 地址会自动保持与此处一致，而通信设定中的 IP 地址修改时则不影响此处）
3. 点击以太网端口设置中的对象配置按钮，点击红框位置，选择设备类型为 MODBUS\_TCP
4. 点击以太网端口设置中的软元件映射按钮，即可查看默认的软元件对应地址，用户也可以选择自行修改软元件映射中的软元件地址，同种类型的软元件，地址类型不能重复。
5. 当所有的操作完成之后，用户就可以通过支持做 MODBUS\_TCP 从站的设备查看或者写入数据到 R 系列对应地址的线圈，输入，输入寄存器或者保持寄存器中。



基本设置 | 对象配置 | 软元件映射 |

项目	内容	线圈	输入	输入寄存器	保持寄存器
分配01	软元件	M0	X0		D0
	起始MODBUS软元件	0	0		0
	点数	10000	256		9000
分配02	软元件	Y0			SD0
	起始MODBUS软元件	10000			10000
	点数	256			6000
分配03	软元件	SM0			
	起始MODBUS软元件	11000			
	点数	6000			
分配04	软元件				
	起始MODBUS软元件				
	点数				
分配05	软元件				
	起始MODBUS软元件				
	点数				
分配06	软元件				
	起始MODBUS软元件				
	点数				

确定 取消

## 1.4 COM1 接口使用方法与介绍

### 1.4.1 各设备 COM1 口类型和支持的协议一览

设备	R1	R2	R8	R8C	R8P
接口类型	RS485 非隔离	RS485 非 隔离	RS485	RS485	RS485
MODBUS_RTU 从站	支持	支持	支持	支持	支持
MODBUS_RTU 主站	支持	支持	支持	支持	支持
自由协议	支持	支持	支持	支持	支持
DIP 从站	支持	支持	支持	支持	支持

### 1.4.2 COM1 接口的使用与接线

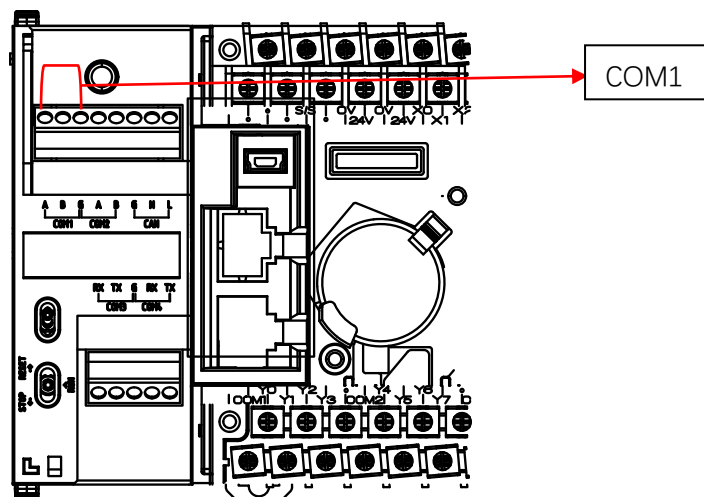
R 系列小型 PLC 上的 COM1 接口类型为 RS485。

上述所有 R 系列 PLC，使用合适的连接线，将 PLC 的 A 接相连设备的 A，PLC 的 B 接相连设备的 B。即可作 MODBUS\_TCP 主站，MODBUS\_TCP 从站，自由协议，或者 DIP 从站。

注意，如无特殊说明，本文中的将以 R8P 为例展示 R 系列 PLC 各接口连接方式



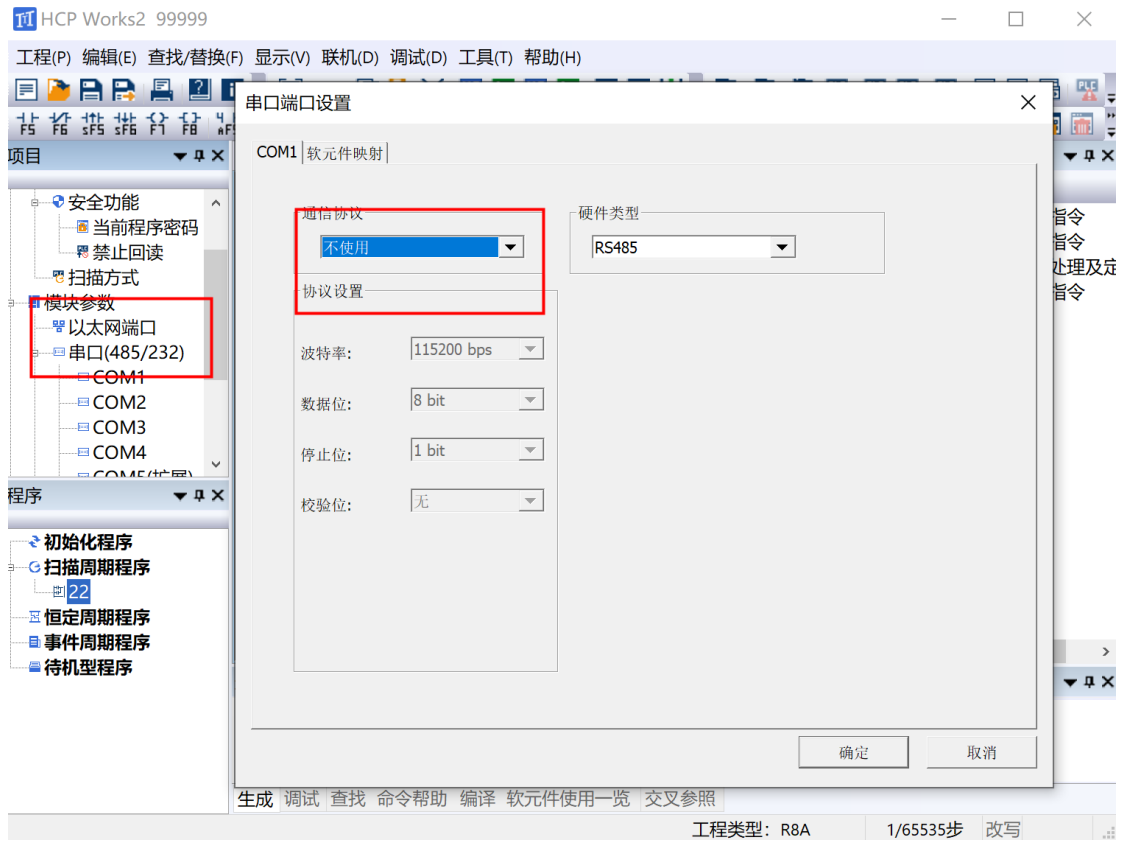
R8P 的 COM1 位置和接线排列如图



### 1.4.3 COM1 口用作 MODBUS\_RTU 主站时设置

R 系列小型 PLC 的 COM1 接口做 MODBUS\_RTU 主站时，需要自行设置参数,COM1 口默认的通信方式是不使用。如果设备需要做 MODBUS\_RTU 主站时，参照如下方法设置：

1. 用标准的线束连接 R 系列 PLC 的 COM1 和支持 RS485 口做 MODBUS\_RTU 从站的设备
2. 点击项目栏中栏中模块参数中的串口 COM1，弹出 COM1 的串口端口设置界面，此时 COM1 的通信协议默认为不使用
3. 点击并修改 COM1 为 MODBUS\_RTU 主站，其波特率，数据位，停止位，校验位需设置与连接设备一致，固有设置属性详见 MODBUS 部分
4. 点击串口端口设置中的软元件映射按钮，即可查看默认的软元件对应地址，用户也可以选择自行修改软元件映射中的软元件地址，同种类型的软元件，地址类型不能重复。
6. 当所有的操作完成之后，用户就可以通过支持 RS485 做 MODBUS\_RTU 从站的设备查看或者写入数据到 R 系列对应地址的线圈，输入，输入寄存器或者保持寄存器中。





### 1.4.4 COM1 口用作 MODBUS\_RTU 从站时设置

R 系列小型 PLC 的 COM1 接口做 MODBUS\_RTU 从站时，需要用户自行修改参数，COM1 口默认的通信方式是不使用。如果设备需要做 MODBUS\_RTU 从站时，参照如下方法设置：

1. 用标准的线束连接 R 系列 PLC 的 COM1 和支持 RS485 口做 MODBUS\_RTU 主站的设备
2. 点击项目栏中模块参数中的串口 COM1，弹出 COM1 的串口端口设置界面，此时 COM1 的通信协议默认为不使用
3. 点击并修改 COM1 为 MODBUS\_RTU 从站，其波特率，数据位，停止位，校验位需设置与连接设备一致，本站号应设置与其他从站站号不一致。
4. 点击串口端口设置中的软元件映射按钮，即可查看默认的软元件对应地址，用户也可以选择自行修改软元件映射中的软元件地址，同种类型的软元件，地址类型不能重复。
5. 当所有的操作完成之后，用户就可以通过支持 RS485 做 MODBUS\_RTU 从站的设备查看或者写入数据到 R 系列对应地址的线圈，输入，输入寄存器或者保持寄存器中。



### 1.4.5 COM1 口用作自由协议使用设置

R 系列小型 PLC 的 COM1 接口做自由协议通信时，需要用户自行修改参数，COM1 口默认的通信方式是不使用。如果设备需要做自由协议时，参照如下方法设置：

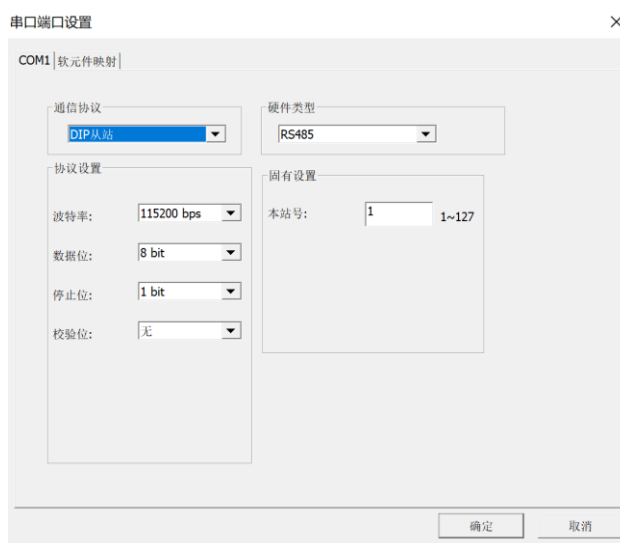
1. 用标准的线束连接 R 系列 PLC 的 COM1 和支持自由协议的设备
2. 点击项目栏中栏中模块参数中的串口 COM1，弹出 COM1 的串口端口设置界面，此时 COM1 的通信协议默认为不使用
3. 点击并修改 COM1 为自由协议
4. 参照自由协议章节按介绍的格式正确使用自由协议通信



## 1.4.6 COM1 口用作 DIP 协议使用设置

R 系列 PLC 的 COM1 口在做 DIP 通信时，仅支持做 DIP 从站设备，需要用户自行修改参数，COM1 口默认的通信方式是不使用。如果设备需要做 DIP 从站时，参照如下方法设置：

1. 用标准的线束连接 R 系列 PLC 的 COM1 和支持自由协议的设备
2. 点击项目栏中栏中模块参数中的串口 COM1，弹出 COM1 的串口端口设置界面，此时 COM1 的通信协议默认为不使用
3. 点击并修改 COM1 为 DIP 从站
4. 参照 DIP 协议章节按介绍的格式正确使用自由协议通信



# 1.5 COM2 接口使用方法与介绍

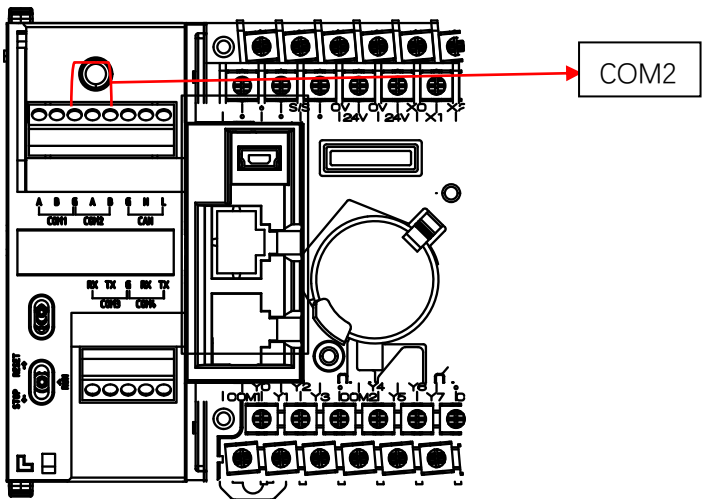
## 1.5.1 各设备 COM2 口类型和支持的协议一览

设备	R1	R2	R8	R8C	R8P
接口类型	BD 扩展口	BD 扩展口	RS485	RS485	RS485
MODBUS_RTU 从站	支持	支持	支持	支持	支持
MODBUS_RTU 主站	支持	支持	支持	支持	支持
自由协议	支持	支持	支持	支持	支持
DIP 从站	支持	支持	支持	支持	支持

## 1.5.2 COM2 接口的使用与接线

R 系列小型 PLC 上的 COM2 接口类型为 RS485。使用合适的连接线，将 PLC 的 A 接相连设备的 A，PLC 的 B 接相连设备的 B。即可作 MODBUS\_TCP 主站，MODBUS\_TCP 从站，自由协议，或使用 DIP 从站协议。

以 R8P 为例,COM2 位置和端子排列如图

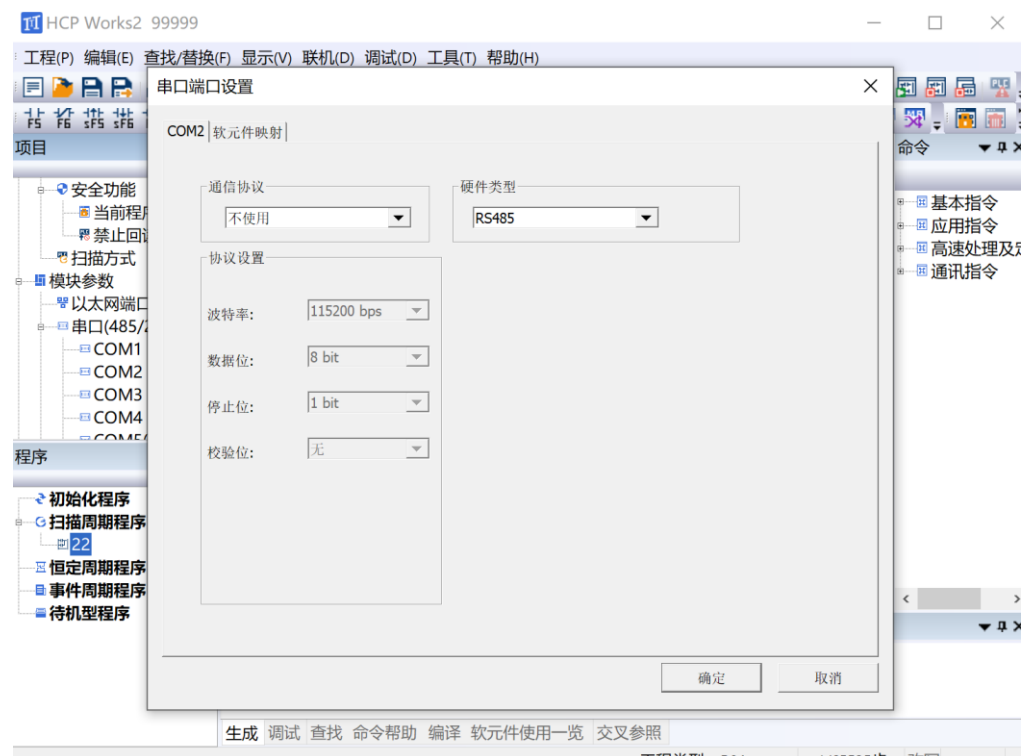


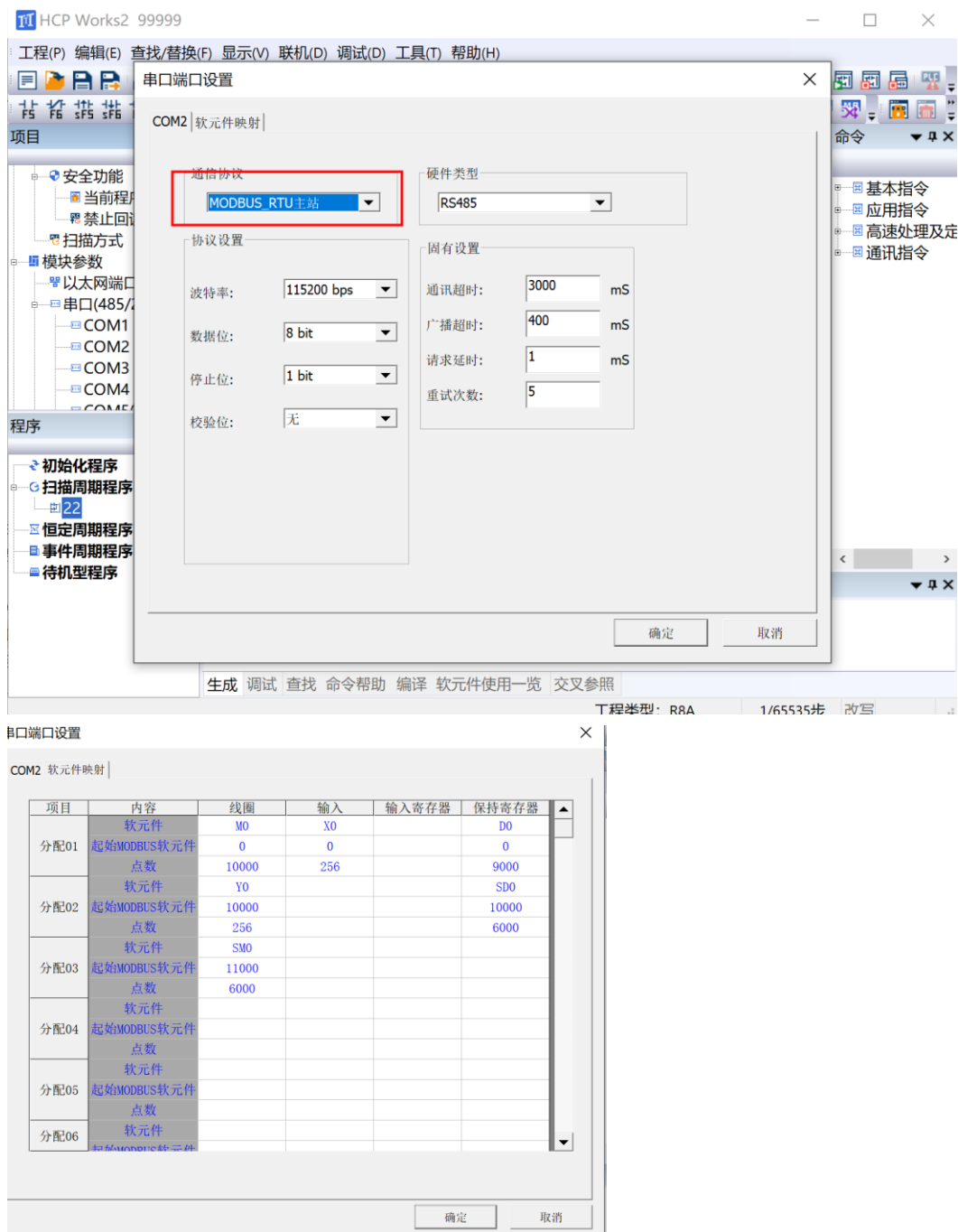
## 1.5.3 COM2 口用作 MODBUS\_RTU 主站时设置

R 系列小型 PLC 的 COM2 接口做 MODBUS\_RTU 主站时，需要自行设置参数,COM2 口默认的通信方式是不使用。如果设备需要做 MODBUS\_RTU 主站时，参照如下方法设置：

5. 用标准的线束连接 R 系列 PLC 的 COM2 和支持 RS485 口做 MODBUS\_RTU 从站的设备

6. 点击项目栏中模块参数中的串口 COM2，弹出 COM2 的串口端口设置界面，此时 COM2 的通信协议默认为不使用
7. 点击并修改 COM2 为 MODBUS\_RTU 主站，其波特率，数据位，停止位，校验位需设置与连接设备一致，固有设置属性详见 MODBUS 部分
8. 点击串口端口设置中的软元件映射按钮，即可查看默认的软元件对应地址，用户也可以选择自行修改软元件映射中的软元件地址，同种类型的软元件，地址类型不能重复。  
当所有的操作完成之后，用户就可以通过支持 RS485 做 MODBUS\_RTU 从站的设备查看或者写入数据到 R 系列对应地址的线圈，输入，输入寄存器或者保持寄存器中。





## 1.5.4 COM2 口用作 MODBUS\_RTU 从站时设置

R 系列小型 PLC 的 COM2 接口做 MODBUS\_RTU 从站时，需要用户自行修改参数，COM2 口默认的通信方式是不使用。如果设备需要做 MODBUS\_RTU 从站时，参照如下方法设置：

6. 用标准的线束连接 R 系列 PLC 的 COM2 和支持 RS485 口做 MODBUS\_RTU 主站的设备
7. 点击项目栏中模块参数中的串口 COM2，弹出 COM2 的串口端口设置界面，此时 COM2 的通信协议默认为不使用
8. 点击并修改 COM2 为 MODBUS\_RTU 从站，其波特率，数据位，停止位，校验位需设置与连接设备一致，本站号应设置与其他从站站号不一致。

9. 点击串口端口设置中的软元件映射按钮，即可查看默认的软元件对应地址，用户也可以选择自行修改软元件映射中的软元件地址，同种类型的软元件，地址类型不能重复。
10. 当所有的操作完成之后，用户就可以通过支持 RS485 做 MODBUS\_RTU 从站的设备查看或者写入数据到 R 系列对应地址的线圈，输入，输入寄存器或者保持寄存器中。



### 1.5.5 COM2 口用作自由协议使用设置

R 系列小型 PLC 的 COM2 接口做自由协议通信时，需要用户自行修改参数，COM2 口默认的通信方式是不使用。如果设备需要做自由协议时，参照如下方法设置：

5. 用标准的线束连接 R 系列 PLC 的 COM2 和支持自由协议的设备
6. 点击项目栏中栏中模块参数中的串口 COM2，弹出 COM2 的串口端口设置界面，此时 COM2 的通信协议默认为不使用
7. 点击并修改 COM2 为自由协议
8. 参照自由协议章节按介绍的格式正确使用自由协议通信



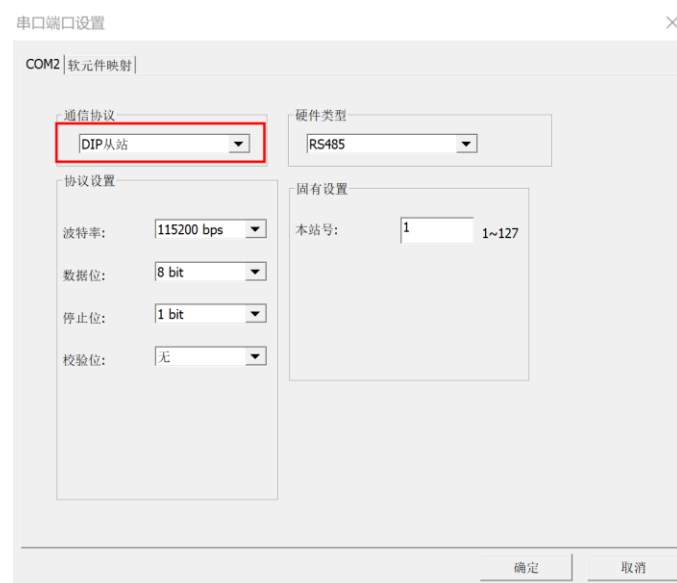
### 1.5.6 COM2 口用作 DIP 协议使用设置

R 系列 PLC 的 COM2 口在做 DIP 通信时，仅支持做 DIP 从站设备，需要用户自行修改参数，COM2 口默认的通信方式是不使用。如果设备需要做 DIP 从站时，参照如下方法设置：

5. 用标准的线束连接 R 系列 PLC 的 COM2 和支持自由协议的设备
6. 点击项目栏中栏中模块参数中的串口 COM2，弹出 COM2 的串口端口设置界面，此时



- COM2 的通信协议默认为不使用
7. 点击并修改 COM2 为 DIP 从站
  8. 参照 DIP 协议章节按介绍的格式正确使用自由协议通信



## 1.6 COM3 口使用方法与介绍

### 1.6.1 各设备 COM3 口类型和支持的协议一览

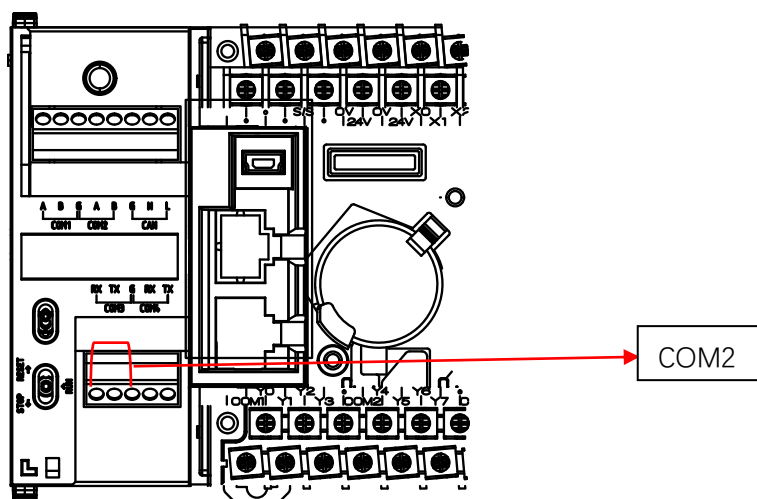
设备	R1	R2	R8	R8C	R8P
接口类型	RS422(圆8pin)	RS422(圆8pin)	RS232	RS232	RS232
下载/监控协议	支持	支持	支持	支持	支持
自由协议	支持	支持	支持	支持	支持

### 1.6.2 COM3 接口的使用与接线

R 系列小型 PLC 上的 COM3 接口类型为 RS232。

使用合适的连接线，将 PLC 的 RX 接相连设备的 TX，PLC 的 TX 接相连设备的 RX,G 接 G。即可作下载监控用，自由协议，或 DIP 协议。

以 R8P 为例，R 系列 COM3 位置和端子排列如图



### 1.6.3 COM3 口用作自由协议使用设置

R 系列小型 PLC 的 COM3 接口做自由协议通信时，需要用户自行修改参数，COM3 口默认的使用方式是下载/监控。如果设备需要做自由协议时，参照如下方法设置：

1. 用标准的线束连接 R 系列 PLC 的 COM3 和支持自由协议的设备
2. 点击项目栏中模块参数中的串口 COM3，弹出 COM3 的串口端口设置界面，此时 COM3 的通信协议默认为不使用
3. 点击并修改 COM3 为自由协议
4. 参照自由协议章节按介绍的格式正确使用自由协议通信



### 1.6.4 COM3 口用作下载监控使用设置

R 系列小型 PLC 的 COM3 接口做下载监控时，需要用户自行修改参数，COM3 口默认的使用方式是下载/监控。如果设备需要使用时，参照如下方法设置：

- 1 将串口板与 COM3 连接，连接方式与 1.7.1 描述一致，串口板与 PC 设备相连
  2. 打开设置中的通信设定，将通信方式设为串口通信，通信端口的选择需要参考设备管理器中，PC 自身给串口板命名的端口号
  3. 点击通信测试，通信成功后会显示通信成功弹窗
- 注：如遇到通信失败，请检查接线是否正确

## 1.6.5 COM3 口用作 DIP 协议使用设置

R 系列 PLC 的 COM3 口在做 DIP 通信时，仅支持做 DIP 从站设备，需要用户自行修改参数，COM3 口默认的通信方式是下载监控。如果设备需要做 DIP 从站时，参照如下方法设置：

1. 将 DIP 通信设备与 COM3 连接，连接方式与 1.7.1 描述一致
2. COM3 默认的下载/监控可直接用于 DIP 通信，此时可配置参数只有波特率
3. DIP 详情请参阅 DIP 章节

串口端口设置

COM3 | 软件映射

通信协议: 下载/监控

硬件类型: RS232

协议设置:

波特率: 115200 bps

数据位: 8 bit

停止位: 1 bit

校验位: 奇数

确定 取消

## 1.7 COM4 口使用方法与介绍

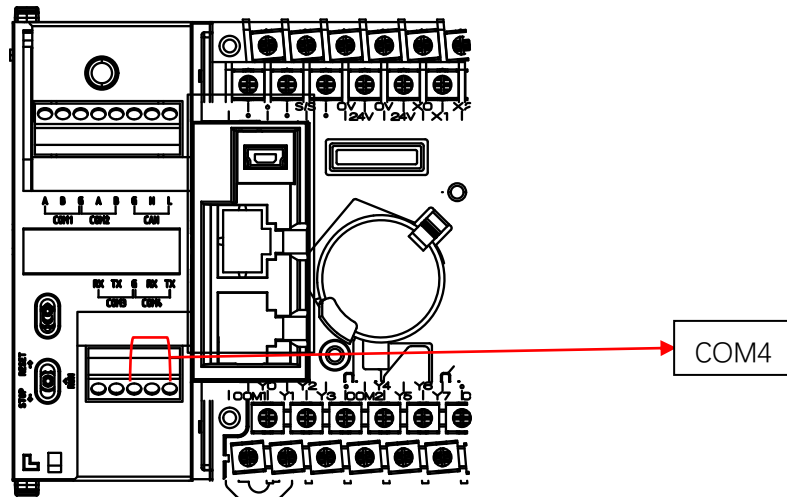
### 1.7.1 各设备 COM4 口类型和支持的协议一览

设备	R1	R2	R8	R8C	R8P
接口类型	RS232(圆8pin)	RS232(圆8pin)	RS232	RS232	RS232
下载/监控协议	支持	支持	支持	支持	支持
自由协议	支持	支持	支持	支持	支持

## 1.7.2 COM4 接口的使用与接线

R 系列小型 PLC 上的 COM4 接口类型为 RS232。使用合适的连接线，将 PLC 的 RX 接相连设备的 TX，PLC 的 TX 接相连设备的 RX，G 接 G。即可作下载监控用，或使用自由协议。

以 R8P 为例，R 系列 PLC 的 COM4 位置和端子排布如图

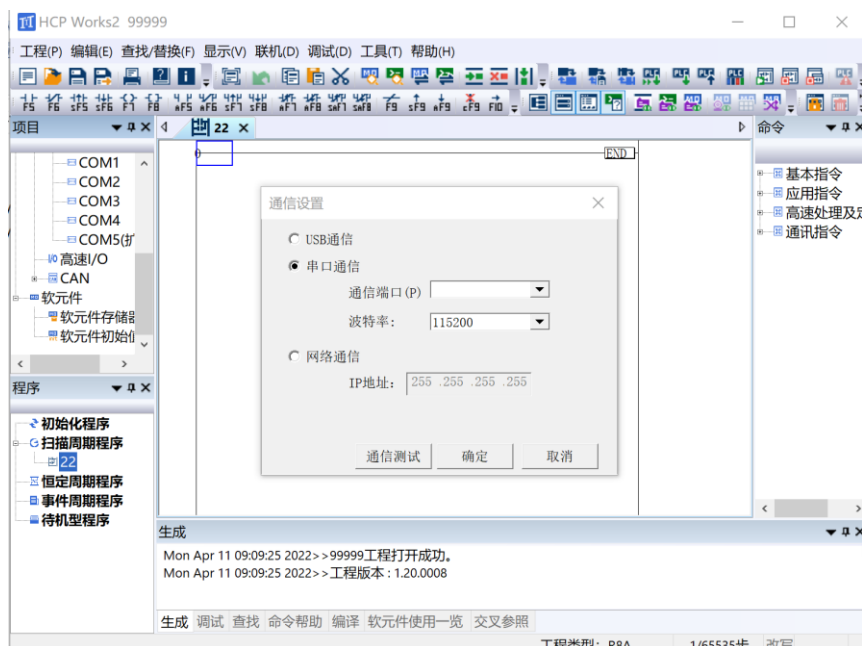


## 1.7.3 COM4 口用作下载监控使用设置

R 系列小型 PLC 的 COM4 接口做下载监控时，需要用户自行修改参数，COM4 口默认的使用方式是不使用。如果设备需要做下载监控时，参照如下方法设置：

1. 将串口板与 COM4 连接，连接方式与 1.7.1 描述一致，串口板与 PC 设备相连
2. 打开设定中的通信设定，将通信方式设为串口通信，通信端口的选择需要参考设备管理器中，PC 自身给串口板命名的端口号
3. 点击通信测试，通信成功后会显示通信成功弹窗

注：如遇到通信失败，请检查接线是否正确



## 1.7.4 COM4 口用作自由协议使用设置

R 系列小型 PLC 的 COM4 接口做自由协议通信时，需要用户自行修改参数，COM4 口默认的使用方式是下载/监控。如果设备需要做自由协议时，参照如下方法设置：

1. 用标准的线束连接 R 系列 PLC 的 COM4 和支持自由协议的设备
2. 点击项目栏中栏中模块参数中的串口 COM4，弹出 COM4 的串口端口设置界面，此时 COM4 的通信协议默认为不使用
3. 点击并修改 COM4 为自由协议
4. 参照自由协议章节按介绍的格式正确使用自由协议通信

## 1.7.5 COM4 口用作 DIP 协议使用设置

R 系列 PLC 的 COM4 口在做 DIP 通信时，仅支持做 DIP 从站设备，需要用户自行修改参数，COM4 口默认的通信方式是不使用。如果设备需要做 DIP 从站时，参照如下方法设置：

4. 将 DIP 通信设备与 COM4 连接，连接方式与 1.7.1 描述一致
5. COM4 选择下载/监控可直接用于 DIP 通信，此时可配置参数只有波特率
6. DIP 详情请参阅 DIP 章节

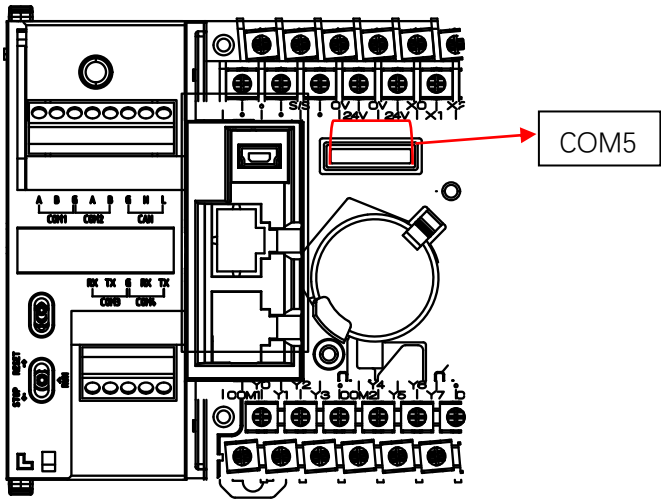
# 1.8 COM5 口的使用方法与介绍

## 1.8.1 各设备 COM5 口类型和支持的协议一览

设备	R1	R2	R8	R8C	R8P
接口类型	BD 扩展	BD 扩展	BD 扩展	左扩展	BD 扩展，左扩展
下载/监控协议	支持	支持	支持	支持	支持
自由协议	支持	支持	支持	支持	支持

## 1.8.2 COM5 口的使用与接线

R 系列小型 PLC 上的 COM5 接口类型为 BD 扩展, 位于 PLC 的上部运行指示灯左侧或者 PLC 左侧, 详见上图。此口仅支持使用 HCA2P-422-BD,HCA2P-232-BD, HCRXB-C24-BD 作为 PLC 的通信扩展口 , 使用时将 BD 板连接到 PLC 的右侧扩展口即可。  
注意 HCA2P-422-BD,HCA2P-232-BD, HCRXB-C24-BD 需要额外购买  
以 R8P 为例, COM5 在 USB 接口的右侧.

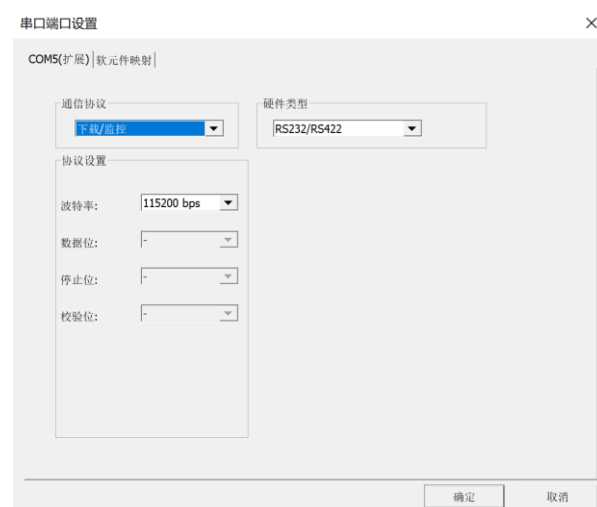


## 1.8.3 COM5 连接 HCA2P-422-BD 用作下载监控使用设置

R 系列小型 PLC 的 COM5 接口做下载监控时, 需要用户自行修改参数, COM5 口默认的使用方式是不使用。如果设备需要做下载监控使用时, 需要将该模块插在 PLC 的右扩展口上, 参照如下方法设置:

1. 用标准的 RS422 接口线束连接 BD 板与安装了 HCP WORKS2 的 PC 设备
2. 将通信方式改为下载监控协议, 硬件类型选择 RS232、RS422

3. 打开设置中的通信设定，将通信方式设为串口通信，通信端口的选择需要参考设备管理器中，PC 自身给串口板命名的端口号
4. 点击通信测试，通信成功后会显示通信成功弹窗



## 1.8.4 COM5 连接 HCA2P-232-BD 用作下载监控使用设置

R 系列小型 PLC 的 COM5 接口做下载监控使用时，需要用户自行修改参数，COM5 口默认的使用方式是不使用。如果设备需要做下载监控使用时，需要将该模块插在 PLC 的右扩展口上，参照如下方法设置：

1. 用标准的 RS232 接口线束(DB9)连接 BD 板与安装了 HCP WORKS2 的 PC 设备
2. 将通信方式改为下载监控协议，硬件类型选择 RS232、RS422
3. 打开设置中的通信设定，将通信方式设为串口通信，通信端口的选择需要参考设备管理器中，PC 自身给串口板命名的端口号
4. 点击通信测试，通信成功后会显示通信成功弹窗

## 1.8.5 COM5 连接 HCRXB-C24-BD 用作下载监控使用设置

R 系列小型 PLC 的 COM5 接口做下载监控使用时，需要用户自行修改参数。COM5 口默认的使用方式是不使用。如果设备需要做下载监控使用时，需要将该模块插在 PLC 的右扩展口上，参照如下方法设置：

1. 用标准的 RS232 或者 RS485 接口线束(DB9)连接 BD 板与安装了 HCP WORKS2 的 PC 设备
2. 将通信方式改为下载监控协议，使用 RS232 时硬件类型选择 RS232、RS422，使用 RS485 时选择为 RS485
3. 打开设置中的通信设定，将通信方式设为串口通信，通信端口的选择需要参考设备管理器中，PC 自身给串口板命名的端口号
4. 点击通信测试，通信成功后会显示通信成功弹窗

## 1.8.6 COM5 连接 HCA2P-422-BD 用作自由协议使用设置

R 系列小型 PLC 的 COM5 接口连接 HCA2P-422-BD 做自由协议通信时需要用户自行修改参数。COM5 口默认的使用方式是不使用。如果设备需要做自由协议通信使用时，需要将该模块插在 PLC 的右扩展口上，参照如下方法设置：

1. 用标准的 RS422 线束连接 BD 板和支持自由协议的设备
2. 点击项目栏中栏中模块参数中的串口 COM5，弹出 COM5 的串口端口设置界面，此时 COM5 的通信协议默认为不使用
3. 点击并修改 COM5 为自由协议，使用 RS485 时硬件类型选择 RS232、RS422
4. 参照自由协议章节按介绍的格式正确使用自由协议通信

## 1.8.7 COM5 连接 HCA2P-232-BD 用作自由协议使用设置

R 系列小型 PLC 的 COM5 接口连接 HCA2P-232-BD 做自由协议通信时需要用户自行修改参数。COM5 口默认的使用方式是不使用。如果设备需要做自由协议通信使用时，需要将该模块插在 PLC 的右扩展口上，参照如下方法设置：

1. 用标准的 RS232 线束连接 BD 板和支持自由协议的设备
2. 点击项目栏中栏中模块参数中的串口 COM5，弹出 COM5 的串口端口设置界面，此时 COM5 的通信协议默认为不使用
3. 点击并修改 COM5 为自由协议，使用 RS232 时硬件类型选择 RS232、RS422
4. 参照自由协议章节按介绍的格式正确使用自由协议通信

## 1.8.8 COM5 连接 HCRXB-C24-BD 用作自由协议使用设置

R 系列小型 PLC 的 COM5 接口连接 HCRXB-C24-BD 做自由协议通信时需要用户自行修改参数。COM5 口默认的使用方式是不使用。如果设备需要做自由协议通信使用时，需要将该模块插在 PLC 的右扩展口上，参照如下方法设置：

1. 用标准的 RS232 线束或，RS485 线束连接 BD 板和支持自由协议的设备
2. 点击项目栏中栏中模块参数中的串口 COM5，弹出 COM5 的串口端口设置界面，此时 COM5 的通信协议默认为不使用
5. 点击并修改 COM5 为自由协议，使用 RS232 时硬件类型选择 RS232、RS422，使用 RS485 时选择为 RS485
- 3.
4. 参照自由协议章节按介绍的格式正确使用自由协议通信

## 1.8.9 COM5 口用作 DIP 协议使用设置

R 系列 PLC 的 COM5 口在做 DIP 通信时，仅支持做 DIP 从站设备，需要用户自行修改参数，COM5 口默认的通信方式是不使用。如果设备需要做 DIP 从站时，参照如下方法设置：



- 7. 将 DIP 通信设备与 COM5 连接，连接方式与 1.7.1 描述一致
- 8. COM5 选择下载/监控可直接用于 DIP 通信，此时可配置参数只有波特率
- 9. DIP 详情请参阅 DIP 章节



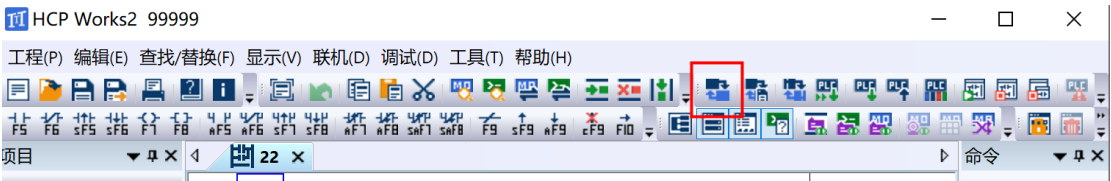
## 2. R 系列的监控功能一览

已经与 PC 通信正常的 R 系列小型 PLC，通过 R 系列的下载/监控协议，可以实现 R 系列小型 PLC 程序的写入，回读，和监控功能，本章将对这些内容进行介绍。

### 2.1 R 系列程序的转换与写入

R 系列小型 PLC 的程序需要在安装了 HCP WORKS2 的 PC 设备上，使用 HCPWORKS2 的梯形图编辑功能进行编辑。当用户按照参考文档和命令窗口的使用实例定义的规则编写了程序时，则可点击编辑栏中的转换按钮，或者按 F4 对梯形图转换。如无出错，且 PLC 与 PC 设备通信正常，则可以点击调试栏中的写入 PLC 按钮，按照弹窗提示，将梯形图写入 PLC 中，并进行重启。

转换按钮如下



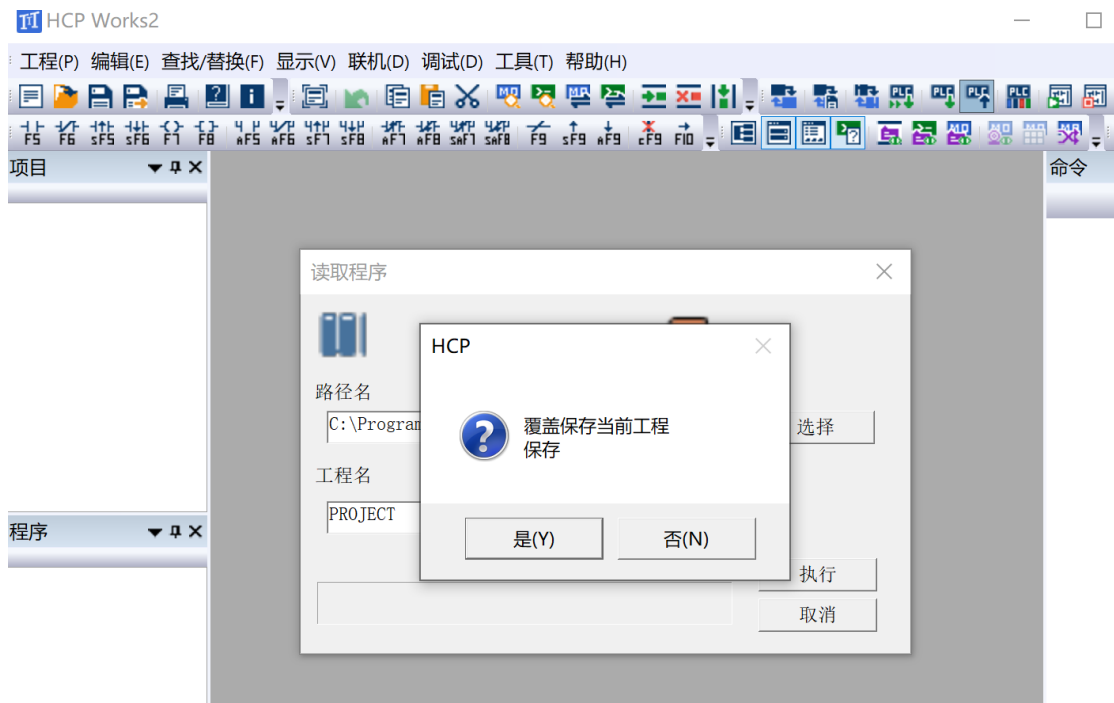
写入按钮如下



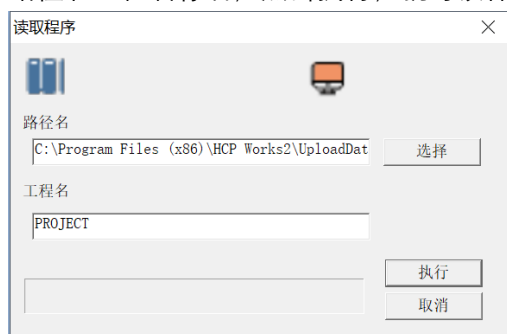
## 2.2 R 系列程序的读取

当 R 系列小型 PLC 中已经写入了程序时，且 PC 设备无程序备份时，使用 HCP WORKS2 可以将其内程序读取到 PC 设备上，其步骤如下

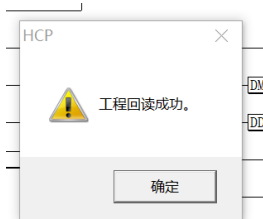
1. 确保 PLC 与 PC 设备通信正常
2. 点击调试栏中的从 PLC 读取按钮，如果有工程与目标工程名重复时会提示覆盖保存当前工程保存



3. 第 2 步之后，或者读取时无工程名重复；会出现提示保存读取工程路径的弹窗，设置好路径和工程名称后，点击执行，就可以保存读取的工程，中点击取消则会取消操作



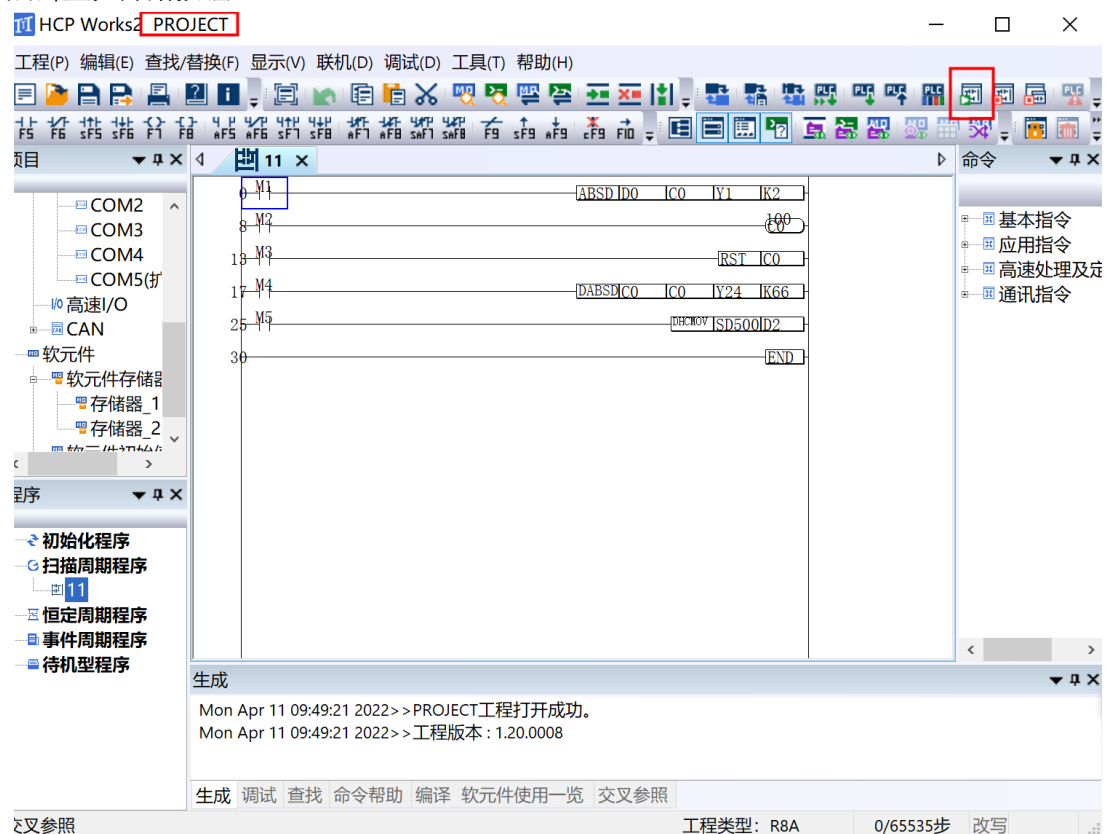
4. 读取成功则会提示工程回读成功，如下所示



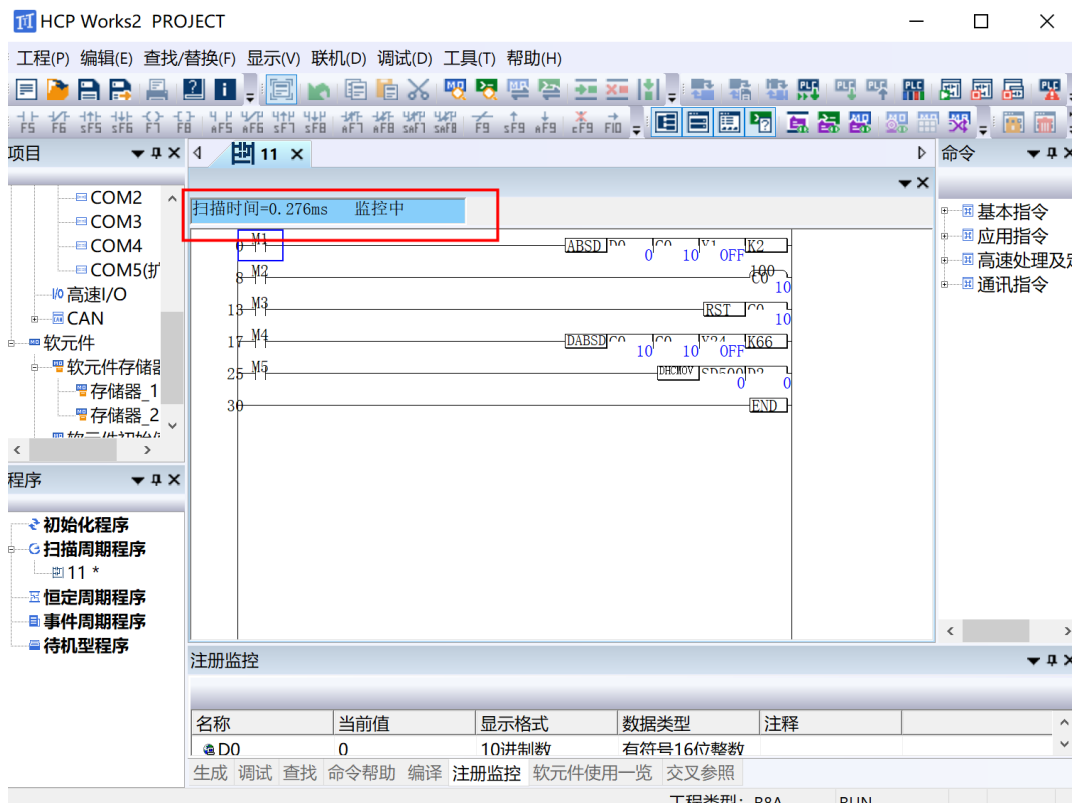
## 2.3 R 系列程序的监控

R 系列小型 PLC 的监控功能需要在 PLC 与 PC 设备通信正常, 且已经写入了程序的前提下进行, 其步骤如下

1. 选择并打开 PLC 中目前正在运行的程序
2. 点击监控开始按钮



3. 监控成功后, 会显示 PLC 当前的扫描时间和 PLC 程序中各软元件的实时状态, 其功能汇总将在下节进行汇总



注：在有多程序运行时，用户想切换监控的程序时必须先停止当前程序的监控，再重新开始下一程序的监控

## 2.4 R 系列监控功能汇总

名称	功能描述	打开方式	备注
监控开始	用于开始 PLC 程序的监控	调试栏监控开始按钮	
监控停止	用于结束当前程序的监控	调试栏监控停止按钮,或者监控的程序上鼠标右键点击监控停止(或者按 S)，或者按快捷键 F4	
全部监控	用于批量监控同类型的软元件	监控的程序上鼠标右键点击全部监控(或者按 B)，或者按快捷键 F5	
监控注册	用于监控不同类型的软元件，并将其状态汇总在注册监控窗口	监控的程序上鼠标右键点击监控注册(或者按 M)，或者按快捷键 F6,或者直接在注册监控输入	
运行/停止,切换	用于切换 PLC 当前的运行状态(运行/停止)	监控的程序上鼠标右键点击运行/停止,切换(或者按 R)，或者按快捷键 shift+F7,或者点击远程栏中的操作按钮	
软元件 设置/重置	用于进行位软元件的值查看，设置和重置	监控的程序上鼠标右键点击软元件设置/重置(或者按 E)，或者按快	

		捷键 F9,或者在全部监控和监控注册窗口进行操作	
改变当前值	用于进行字软元件的值设置, 查看	监控的程序上鼠标右键点改变当前值 (或者按 V), 或者按快捷键 shift+F9,或者在全部监控和监控注册窗口进行操作	
软元件全部清空	用于将所有非锁存软元件值恢复至默认值	监控的程序上鼠标右键点软元件全部清空 (或者按 A), 或者按快捷键 CTRL+F9	只能在 PLC 为 STOP 时操作
软元件加锁存数据全部清空	用于清空用户软元件中所有的当前值	监控的程序上鼠标右键点软元件加锁存数据全部清空 (或者按 L)	只能在 PLC 为 STOP 时操作
错误信息诊断	用于诊断 PLC 运行程序的出错	调试栏, 错误信息按钮	
重启	用于在线重启 PLC	远程栏, 重启按钮	
初始化	用于初始化 PLC	远程栏, 初始化按钮	
调试窗口	记录了 PLC 监控的程序名称, 监控时刻, 和退出时刻	IDE 下方调试按钮	

## 2.4.1 监控开始功能简介

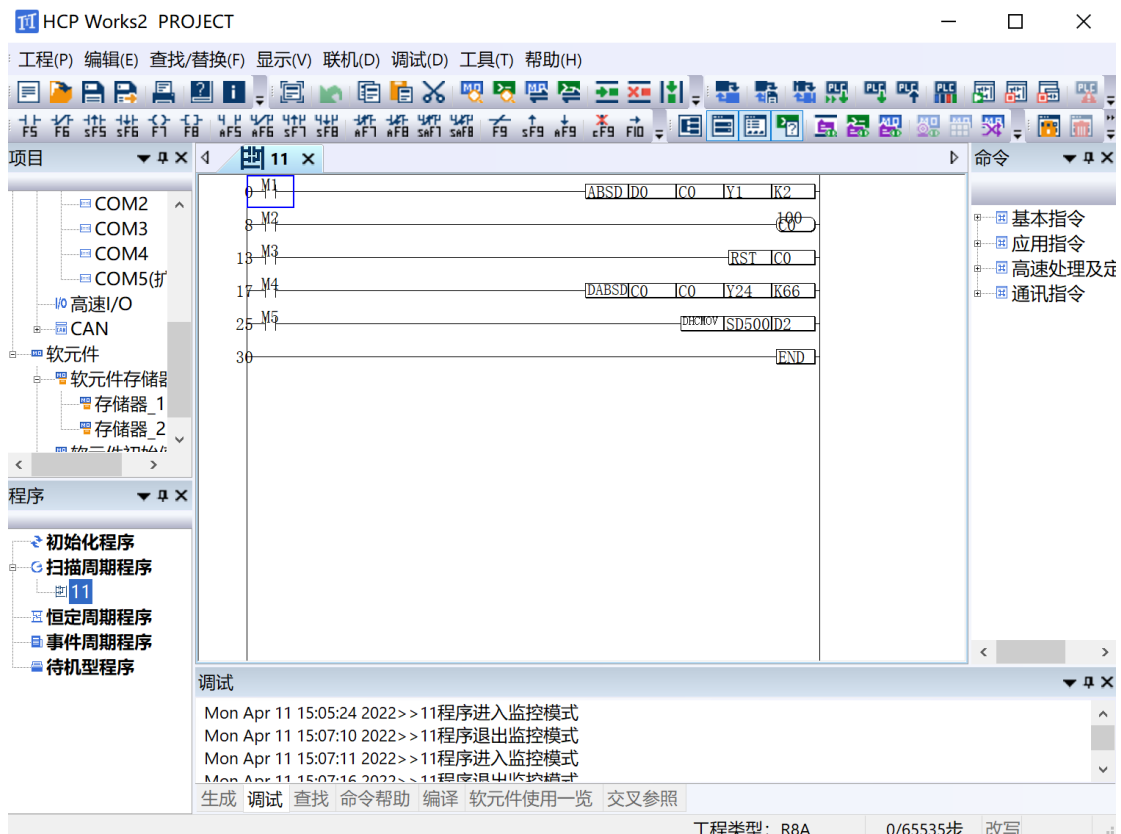
监控开始用于对已写入 PLC 的程序进行监控, 以便查看各软元件的实时状态, 分析各指令程序的运行情况。在进行监控以后, 才可以进行下述功能的使用, 具体步骤如 2.3 所示

## 2.4.2 监控停止功能简介

监控停止功能用于停止对当前程序的监控, 当用户开始监控后, 如果想停止监控, 点击调试功能栏的停止监控即可。

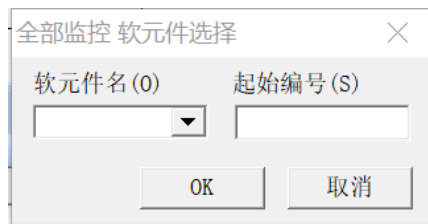


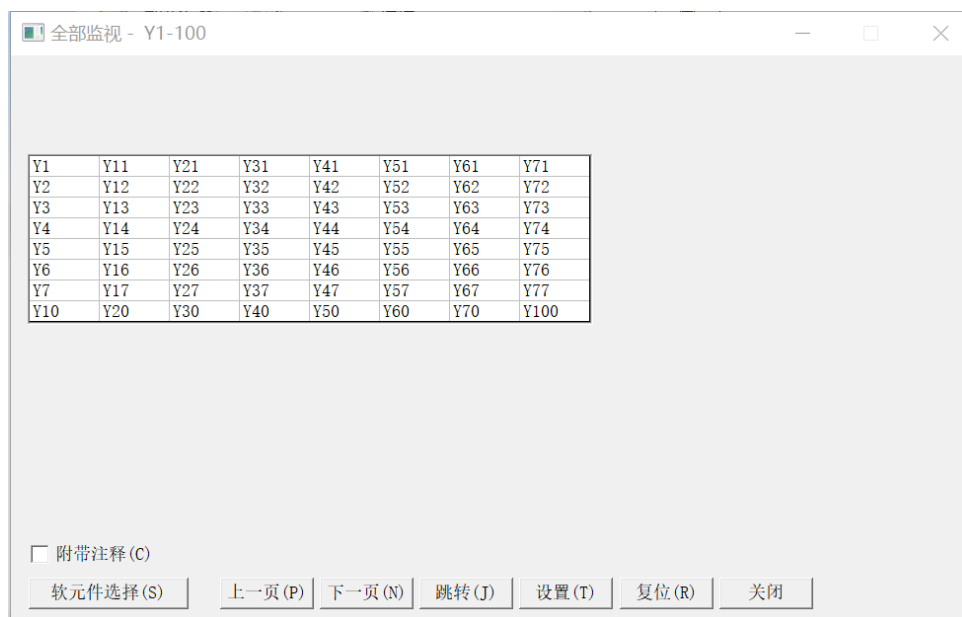
当停止监控成功时, 梯形图将不在显示软元件的实时状态, 监控窗口等监控过程才能使用的功能也将关闭, 直至下次监控开始。



### 2.4.3 全部监控功能简介

全部监控功能用作同类型软元件的批量监控，并支持其状态的修改。当进行全部监控时，在选择软元件名称和起始编号之后，就会批量显示该软元件的显示信息，并支持实时修改，支持同时进行多个软元件的批量监控





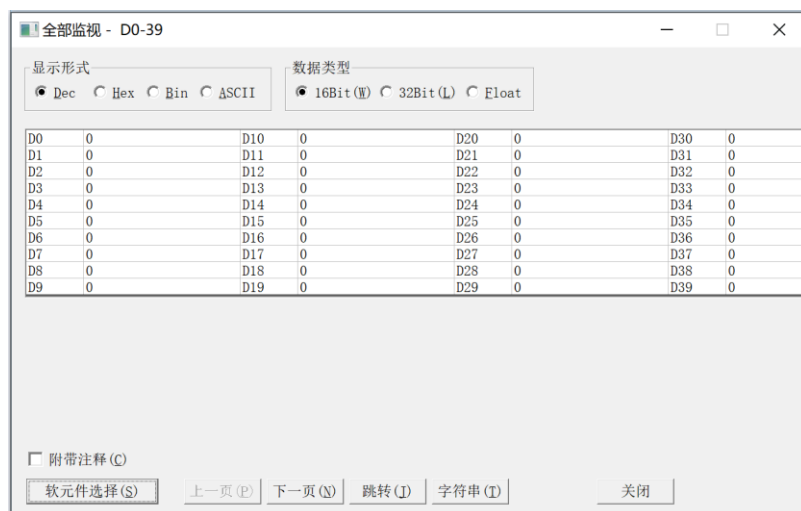
信息的显示：

1. 当批量监控位软元件时，全部监控窗口的名称将会显示当前页面该类型软元件的范围，图中是 Y1-Y100。弹窗中该位软元件所在的方格默认为白色代表该软元件为 OFF,当其为 ON 时，该软元件会变成黄色。
2. 用户在查阅时也可以选择勾选附带注释选项，此时软元件的方格将会显示其对应注释内容
3. 如果用户需要修改查阅软元件的编号，可以通过点击上一页，下一页进行页面的顺序跳转，也可点击跳转按钮直接进行目标页面的跳转
4. 用户在监控字软元件时，可以选择目标的显示形式和数据类型

位软元件数据的修改：

用户也可以双击该软元件所在的方格根据弹窗提示对其进行置位和复位操作,也可点击右下角的设置和复位按钮进行操作。

字软元件数据的修改：



用户也可以双击该软元件所在的方格根据弹窗提示对其进行值的设置。

页面的操作：

1. 如果用户需要切换软元件的监控可以电机软元件-选择按钮，点击后将会重新弹出软元件类型和编号选择的界面，用户可重新输入要监控的软元件和编号
2. 如果用户需要缩小当前的监控窗口，可以点击右上角的最小化按钮
3. 如果用户想要进行当前页面的关闭，可以点击右上角或右下角的关闭按钮

### 2.4.4 监控注册功能简介

监控注册功能主要进行不同类型的软元件的单个监控，本节内容将对此功能进行详细说明。



名称	当前值	显示格式	数据类型	注释
● D0	0	10进制数	有符号16位整数	
● Y1	OFF	2进制数	位	

注册软元件的输入方式：

1. 监控的程序上鼠标右键点击监控注册（或者按 M），或者按快捷键 F6
2. 直接在注册监控双击软元件名称下的空白格子

注册软元件的修改方式：

1. 双击注册窗口已输入的软元件名称进行修改

注册软元件的删除方式：

1. 右键点击监控注册，点击注册解除或者按 shift+F6

注册窗口的调整：

1. 用户可拉动注册监控窗口的边框边缘进行窗口的大小调整

窗口信息汇总：

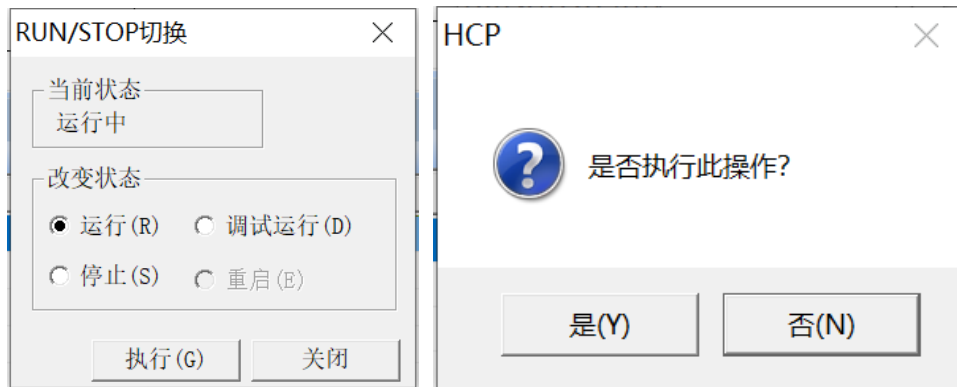
1. 名称：描述了软元件的类型及编号，用户自己写入
2. 当前值：显示了目标软元件的当前值，支持在此修改软元件的当前值
3. 显示格式：R 系列目前支持的显示格式为二进制十进制十六进制，用户可自行选择数据的格式
4. 数据类型：用户可根据实际需要选择有符号十六位整数，无符号十六位整数，有符号 32 位整数，无符号十六位整数，单精度浮点，字符串以及时间
5. 注释：此内容描述了目标软元件的注释明细

### 2.4.5 运行，停止/切换功能简介

运行，停止/切换功能用于在线切换 PLC 的运行状态，要使用此功能，使用鼠标右键菜单中的此功能按钮即可。弹出的窗口如图所示，目前只能进行运行和停止状态的切换。如果需要切换当前状态，改变了状态后，点击执行按钮会弹出是否执行弹窗，确认后点击是即可。

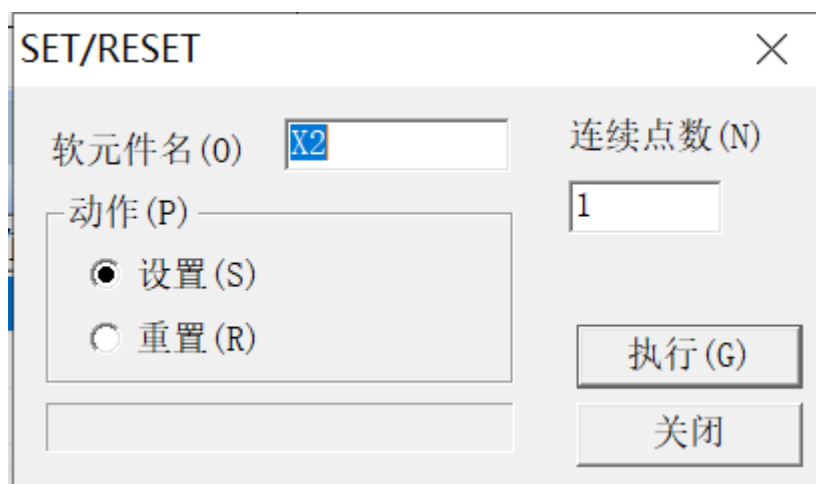
注：用户可根据 PLCrun 灯的亮灭，分别判断 PLC 处于运行还是停止状态。





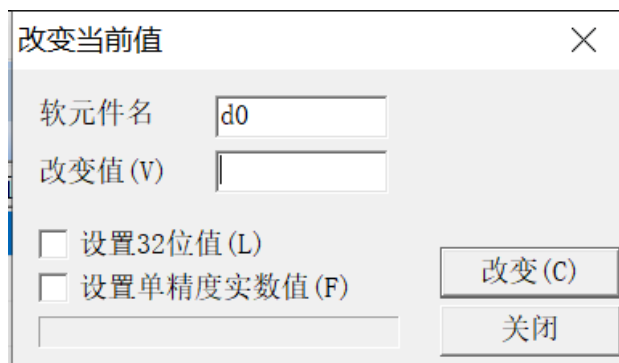
## 2.4.6 软元件 设置/重置功能简介

位软元件的值设置可以通过此功能进行，用户可以手表右键点击软元件 设置/重置按钮，或者直接双击梯形图上的位软元件进行其值的设置和复位。



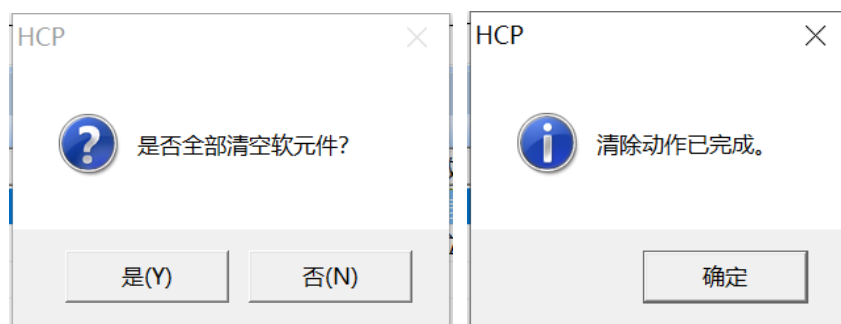
## 2.4.7 改变当前值功能简介

改变当前值功能用于修改字软元件的当前值，用户可以通过鼠标右键改变当前值按钮打开当前值修改界面，也可以双击梯形图上的字软元件进行界面的打开。用户可以选择设置值的数据类型为 32 位值，不选时为默认的 16 位类型。如果需要设置单精度实数，勾选该选项即可。完成设置值的填写和数据类型选择后，点击改变即可完成当前值的修改。



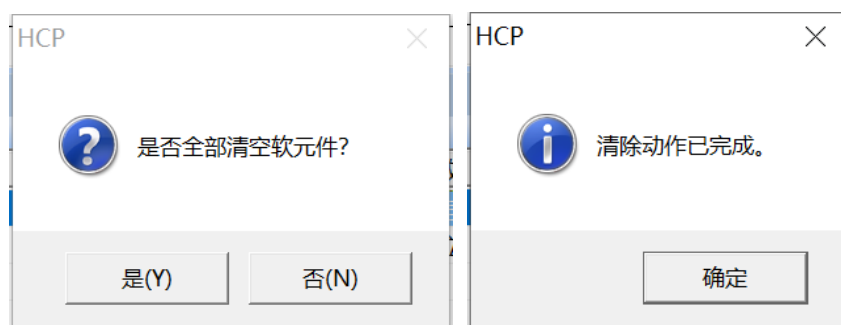
## 2.4.8 软元件全部清空功能简介

软元件全部清空功能，用于清空所有非锁存的软元件的当前值，将其值刷新为初始值，在 PLC 处于停止的状态下，鼠标右键点击软元件全部清空按钮会弹出如下提示，点击是后清除成功，则会弹出清除动作已完成。所有用户软元件的值变成初始值。



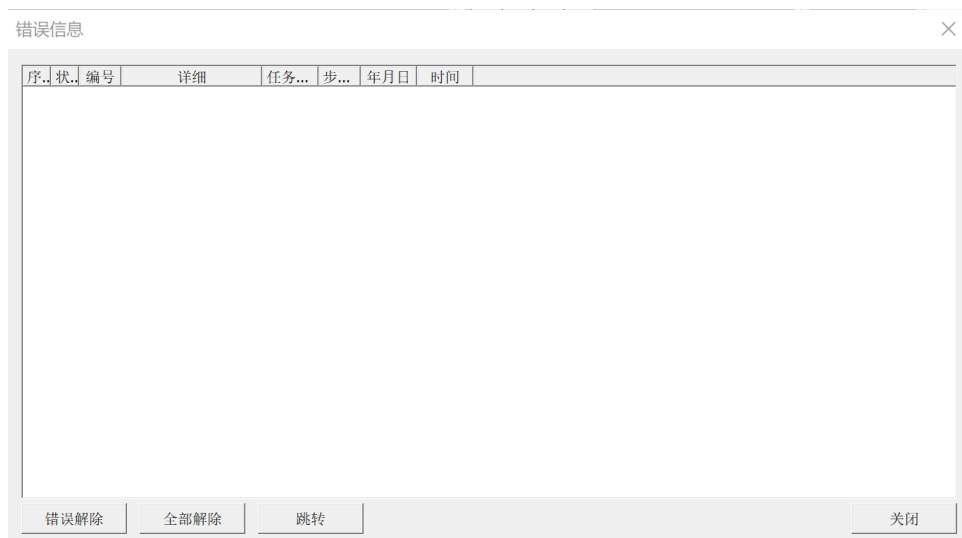
## 2.4.9 软元件加锁存数据全部清空功能简介

软元件加锁存数据全部清空功能，用于清除所有用户软元件的当前值，将其值刷新为初始值。只有在 PLC 为停止状态下可以使用，鼠标右键点击软元件加锁存数据全部清空按钮会弹出如下提示，点击是后清除成功，则会弹出清除动作已完成。所有用户软元件的值变成初始值。



## 2.4.10 错误信息诊断功能简介

R 系列小型 PLC 的错误信息诊断功能用于，在监控时查看 PLC 当前的错误信息。



序号：用于显示出错的顺序编号

状态：用于显示错误的状态，目前有轻度错误，中度错误，重度错误三种类型

编号：代表了出错的具体类型，详见软元件篇

任务名称：发生出错的 PLC 程序名称

步 NO.：发生出错位置的步号

年月日：发生出错的年月日

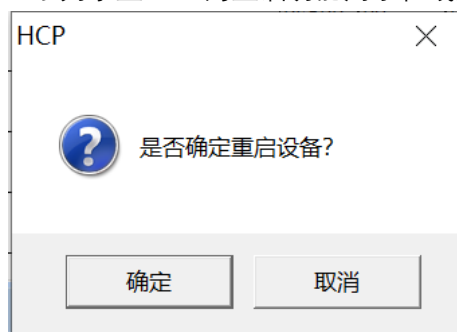
时间：发生出错的时刻

错误接触按钮：选中错误后，点击错误接触按钮即可解除单个错误

全部解除按钮：选中错误后，点击全部解除按钮即可解除所有错误

## 2.4.11 重启功能简介

R 系列小型 PLC 的重启功能用于在线重启 PLC，点击远程栏中的重启按钮会弹出如下窗口

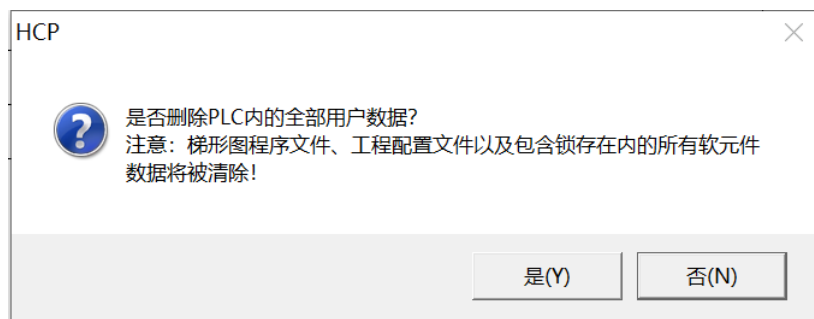


点击确定会将 PLC 重启，点击取消会中止重启。点击重启后会显示设备正在重启，重启成功后会显示重启成功



## 2.4.12 初始化功能简介

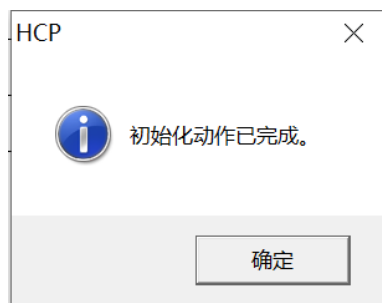
初始化功能用于将 PLC 内的所有内容清除，将其状态设置为初始状态，此功能仅在 PLC 为停止状态可以使用。点击远程栏中的初始化按钮会弹出如下弹窗



点击否可以终止初始化，点击是后会弹出操作执行中提示窗

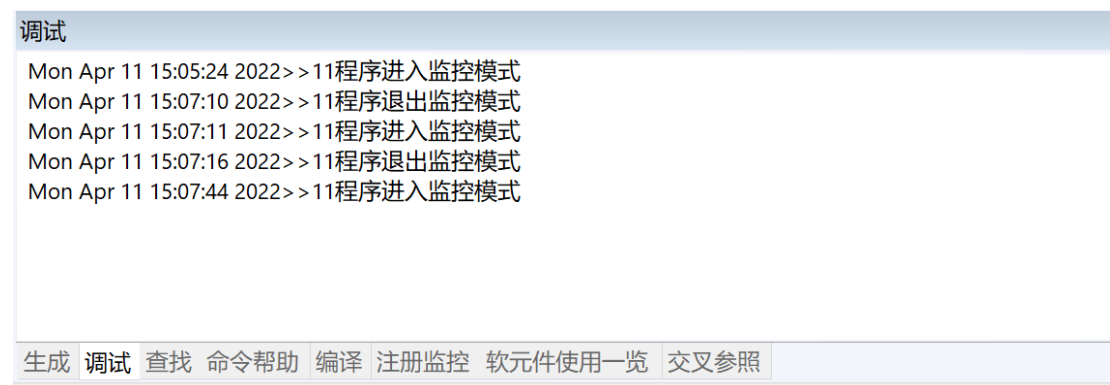


待到初始化动作完成，会提示初始化已经完成



## 2.4.13 调试窗口功能简介

输出窗口的调试窗用于显示监控程序开始监控和退出监控的时刻



## 3.自由协议简介

### 3.1 概要

本章内容对 HCR 系列的自由通信协议做简要说明。

所谓自由格式,即自定义协议通信,现在市场上很多智能设备都支持 RS232 或者 RS485 通信,而各

家产品所使用的协议不尽相同,例如:一些温度控制器厂家和一些仪表厂家就使用的自定义协议,自由格式通信是以数据块的形式进行数据传送,这样就可以按各厂家的格式,进行自由通信。

禾川 R 系列产品的串口都支持自由通信格式,其中 COM2、COM2 硬件固定为 RS485; COM3、COM4 硬件固定为 RS232; COM5 硬件可选配置为 RS485 或 RS232/RS422 类型。

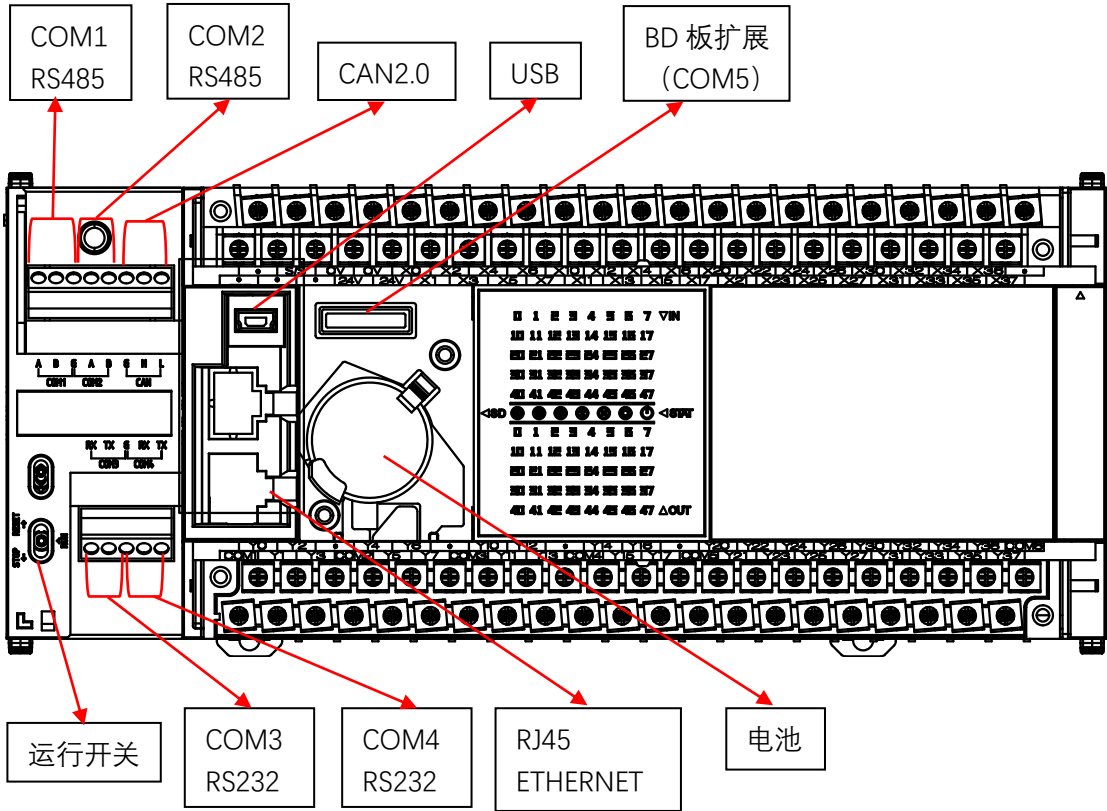
各设备支持自由协议的接口

设备	R1	R2	R8	R8C	R8P
COM1	RS485 非隔离	RS485 非隔离	RS485	RS485	RS485
COM2	BD 扩展口	BD 扩展口	RS485	RS485	RS485
COM3	RS422(圆8pin)	RS422(圆8pin)	RS232	RS232	RS232
COM4	RS232(圆8pin)	RS232(圆8pin)	RS232	RS232	RS232
COM5	BD 扩展	BD 扩展	BD 扩展	左扩展	BD 扩展,左扩展
自由协议	支持	支持	支持	支持	支持

### 3.2 系统配置

以下对有关使用 MODBUS 串行通信所需的系统配置的概要内容进行说明

#### 2.1HCR8/HCR8P 的端口配置图



目前 R 系列的自由协议可在 COM2、COM2、COM3、COM4 和 COM5（扩展）上使用。其中 COM2 默认为其中 COM2 默认为 MODBUS-RTU 从站，COM2 默认为不使用

#### 3.2.2 HCR8C 的端口配置图

4. 略

#### 4.2.2 HCR1/HCR2 的端口配置图

略

### 3.3 规格

本章对有关自由协议串行通信功能的规格的内容进行说明。

### 3.3.1 通信规格

项目		规格
通信规格	波特率	9600/19200/38400/57600/115200/25600bps
	数据长度	8bit(有些从设备的数据位实际上包括了校验位，加上校验位就是 9 位)
	停止位	1bit/2bit
	校验位	可选“无(None)”、“奇校验 (Odd)”、“偶校验 (Even)”三种
	传送距离	500 米
	通信格式	自由协议
固有设置	通信超时	是指 PLC 发送请求后收不到回应，等待重发的时间。如果设置通信超时为 3000ms，PLC 等待对方 3000ms，未收到回应消息，则重新发送请求。通信超时可设置范围为 0 至 6000ms，其中设置为 0 时表示不使用“通信超时”功能，默认 3000ms
	帧超时	是指 PLC 接收的两帧数据的时间间隔，用于 PLC 区分一帧数据接收结束，当两帧数据间的间隔时间大于帧超时则表示一帧数据接收结束，帧超时可设置范围为 0 至 255ms，默认 10ms。
	重试次数	通信时 PLC 未收到回复时，则重新发送请求，若超过重发次数，则跳过当前发送，并记录到 (SD407) 重试次数。接收正常后 SD407 自动清零，默认 5
	起始符	各占 1 字节，用户可设置一个起始/终止符，设置起始/终止符之后，PLC 在发送数据时，自动加上起始/终止符，在接收数据时，自动去掉起始/终止符。不计入接收或发送数据个数。其实起始符、终止符可以看成是协议中数据帧头、帧尾，因此，如果下位机通信有起始符终止符时，既可以在软件中设定，也可以将其写在协议中，勾选时有效。注：起始符、终止符只能输入一个 8 位字符，按 16 进制输入。如 0X30。
	终止符	
	缓冲位数	可选 8 位或 16 位。选择 8 位缓冲形式进行通信时，通信过程中寄存器的高字节是无效的，PLC 只利用寄存器的低字节进行发送和接收数据。选择 16 位缓冲形式进行通信时，PLC 将发送寄存器的全部数据，且先发送低字节数据，再发送高字节数据。当需要把一个 16 位寄存器的低字节和高字节一起传送给另一个 16 位寄存器中时，必须选择 16 位缓冲形式进行通信，且通信字节数是 2 个。当一个 16 位寄存器中存放的数值只占用了低字节，则我们可以选择 8 位缓冲形式进

		行通信，通信字节数是 1 个。往往我们进行通信时数据一般不会超过一个寄存器的低字节（即 HFF），所以只要采用软件中默认的 8 位缓冲进行通信即可
--	--	---

注：参数设置好后，重新上电才能生效。

### 3.3.2 链接时间

略

## 3.4.自由协议规则

### 3.4.1 自由协议帧格式说明

帧格式有下面 4 种：

1	数据
2	起始符 数据
3	数据 终止符
4	起始符 数据 终止符

下面以格式 4 举例介绍缓冲位数 8 位和 16 位的区别，其中“0X”代表 16 进制。

起始符	高8位	低8位	高8位	低8位	终止符
0X02	0X1234		0X5678		0X03

8 位模式时：则串口发送或收到的数据依次为“0X02”，“0X34”，“0X78”，“0X03”。

16 位模式时：则串口发送或收到的数据依次为“0X02”，“0X34”，“0X12”，“0X78”，“0X56”，“0X03”。

## 3.5.特殊软元件介绍

### 3.5.1 特殊继电器

COM2	COM2	COM3	COM4	COM5	名称	详细内容	R/W
SM400	SM420	SM440	SM460	SM480	自由格式通信发送中标志	指令开始执行时，置 ON 发送完成时，置 OFF	R



SM401	SM421	SM441	SM461	SM481	自由格式通信接收完成标志	接收到一帧数据接收完成或超过重试次数时，置 ON；需要用户程序置 OFF	RW
SM402	SM422	SM442	SM462	SM482	自由格式通信缓冲位数	8 位 OFF 16 位 ON	R
SM410	SM430	SM450	SM470	SM490	MODBUS 通信出错	发生 MODBUS 通信出错时为 ON。	R
SM411	SM431	SM451	SM471	SM491	MODBUS_RTU 通信中	从开始执行命令直到命令执行结束标志 ON 为止，MODBUS 串行通信中为 ON。	R
SM412	SM432	SM452	SM472	SM492	串行通信发生重试	从站在超时设置时间内无响应时，在主站发生发送重试的动作后为 ON。 需要用户程序置 OFF，或者释放指令清除	R/W
SM413	SM433	SM453	SM473	SM493	串行通信发生超时	发生响应超时时为 ON。 需要用户程序置 OFF，或者释放指令清除	R/W

注：自由格式通信专指自由格式通信时有效，modbus 通信专指 modbus 通信有效，串行通信指所有的通信协议都有效。下同。

### 3.5.2 特殊寄存器

COM2	COM2	COM3	COM4	COM5	名称	详细内容	R/W
SD400	SD420	SD440	SD460	SD480	自由格式通信发送结果	0: 正确 2: 发送长度设置错误 如设置为 0 或大于 256 时 10: 自由格式发送缓冲区溢出 发生溢出后需要使用释放指令 (RLS) 清除错误。缓冲区大小 256	R/W
SD401	SD421	SD441	SD461	SD481	自由格式通信接收结果	0: 正确 10: 接收长度设置错误 11: 接收数据短 12: 接收数据长 13: 接收错误 14: 接收超时 15: 无起始符 16: 无终止符	R/W
SD402	SD422	SD442	SD462	SD482	自由格式通信接收数据个数	按字节计，不包含起始符、终止符	R
SD403	SD423	SD443	SD463	SD483	串行通信设置	设置在 PLC 中的通信属性会被记	R

						录。	
SD404	SD424	SD444	SD464	SD484	串行通信动作模式显示	0: 通道:未启用 1: 下载/HMI 监控协议 2: 3: 5: 自由协议通信 9: MODBUS RTU 主站 10: MODBUS RTU 从站	R
SD405	SD425	SD445	SD465	SD485	自由格式通信 起始符、终止符	起始符高 8 位, 终止符低 8 位	R
SD406	SD426	SD446	SD466	SD486	自由格式通信 帧超时	设置在 PLC 中的帧超时会被记录。 单位 mS	R
SD407	SD427	SD447	SD467	SD487	串行通信 当前的重试次数	因从站响应超时而进行通信重试 时, 当前的重试次数会被记录	R
SD408	SD428	SD448	SD468	SD488	串行通信 从站响应超时 (通信超时)	从站响应超时的设置值会被记录。 单位 mS	R
SD409	SD429	SD449	SD469	SD489	串行通信 超时时的重试次数	超时时的重试次数的设置值会被 记录。	R
SD410	SD430	SD450	SD470	SD490	MODBUS 通信出 错代码	在 MODBUS 串行通信中发生的最新 出错代码会被记录。	R
SD411	SD431	SD451	SD471	SD491	MODBUS 通信出 错的详细内容	最新出错的详细内容会被记录。	R
SD412	SD413	SD414	SD415	SD416	本站站号	设置的从站站号会被记录	R

V0.4 新增对接收错误情况的说明:

错误码	说明
10	指令设置接收的数据长度超过 256 个或者设置为 0 时(不管是 8 位还是 16 位模式)
11	帧间隔超过帧超时时间没有收到指定的个数, 报接收数据短
12	帧间隔超过帧超时时间接收超过指定的个数, 报接收数据长
13	接收数据的校验位, 数据位等设置不符, 报接收错误
14	通信超时时间内没有收到数据, 报接收超时
15	通信设置里面设置了起始符而没有收到。报无起始符
16	通信设置里面设置了终止符而没有收到。报无终止符

#### 串行通信设置

用于读取设置串行通信的格式 (只读)

位	名称	内容	
		0 (bit = OFF)	1 (bit = ON)
b0	—	—	—

b1、b2	奇偶位	(b2、b1) = (0、0): 无 (b2、b1) = (0、1): 奇数 (b2、b1) = (1、1): 偶数	
b3	停止位	1bit	2bit
b4 ~ b7	波特率 (bpS)	(b7、b6、b5、b4) = (1、0、0、0): 9600 (b7、b6、b5、b4) = (1、0、0、1): 19200 (b7、b6、b5、b4) = (1、0、1、0): 38400 (b7、b6、b5、b4) = (1、0、1、1): 57600 (b7、b6、b5、b4) = (1、1、0、1): 115200 (b7、b6、b5、b4) = (1、1、1、1): 256000	
b8 ~ b15	—	—	—

## 3.6.自由通信指令格式

### 3.6.1 发送数据 SEND

将本机内指定的数据写到目的设备数据的指令，发送个数不包括起始符和终止符。

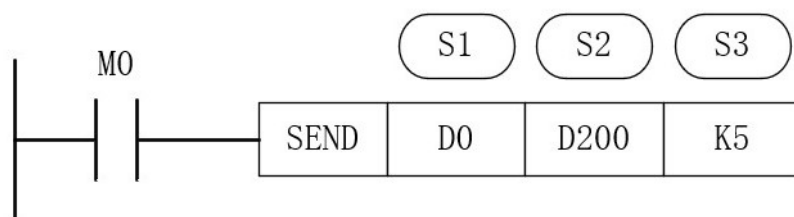
SEND S1 S2 S3		发送数据	范围	数据类型
S1	首地址	发送软元件首地址	—	S16
S2	个数	需要发送个数	K1~K256	S16
S3	通信端口	代指串口号	K1~K5	S16

操作数:

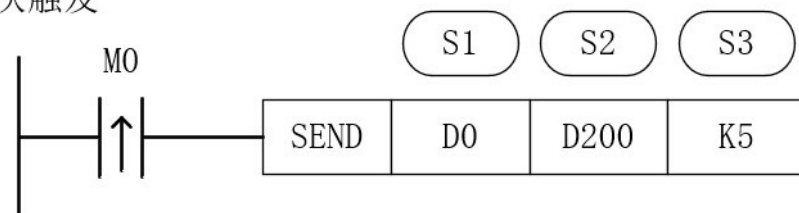
操作数	位	字		双字		常数	
	X、Y、M、L、 SM、F、B、SB、S	T、ST、C、D、 W、SD、SW、R	Z	LC	LZ	K、H	E
S1	—	○	—	—	—	—	—
S2	—	○	—	—	—	○	—
S3	—	—	—	—	—	○	—

指令式样:

### 连续触发 ①



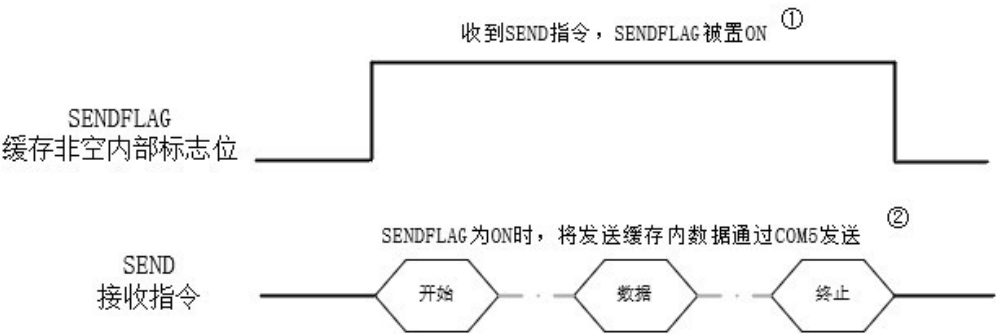
### 单次触发 ②



- S1:发送数据地址从 D0 开始
- S2:发送 D200(数值) 个数据, 若 D200=0, 不发送。比如要发送 D1~D18, 则 D200=18, 不管数据缓冲是 8 位还是 16 位格式。(后台需要根据缓冲位处理好实际发送的字节数)
- S3:指定 COM5 发送。范围 K1~K5, 代指 COM2~COM5.
- 相应端口未使能下, 使用 SEND 指令, PLC 系统应报故障码:  
7601: 自由协议未设定, SEND 指令不可执行

注意: K1-K5 任一端口未设定自由协议, 使用 SNED 指令, 均报 7601 故障码

时序图：



- ① 接收到 SEND 指令时，自由格式驱动任务被唤醒，在自由格式驱动任务内，将缓存队列内数据通过 COM5 发送，根据 IDE 设置返回发送结果。当自由格式驱动端口发送与接收空闲时，自由格式驱动任务被挂起。
- ② 如果缓存队列内有多帧数据需要发送，则全部数据发送完成后任务才会被挂起。

### 3.6.2 接收数据 REC

将目的设备的数据写到本机内指定地址的指令，接收指令不影响其他逻辑指令的执行。

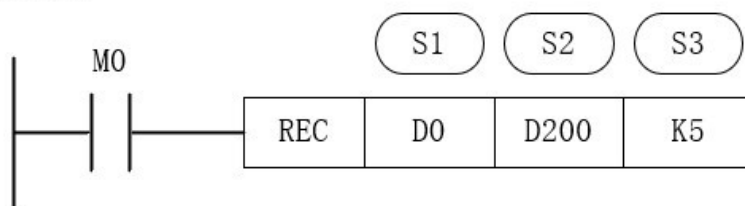
REC S1 S2 S3	发送数据	范围	数据类型
S1	首地址	接收软元件首地址	—
S2	个数	需要接收个数	K1~K256
S3	通信端口	代指串口号	K1~K5

操作数：

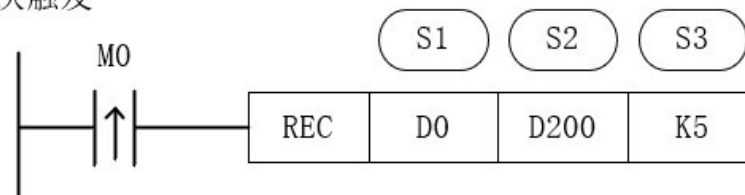
操作数	位	字		双字		常数	
	X、Y、M、L、 SM、F、B、SB、S	T、ST、C、D、 W、SD、SW、R	Z	LC	LZ	K、H	E
S1	—	○	—	—	—	—	—
S2	—	○	—	—	—	○	—
S3	—	—	—	—	—	○	—

指令式样：

### 连续触发 ①



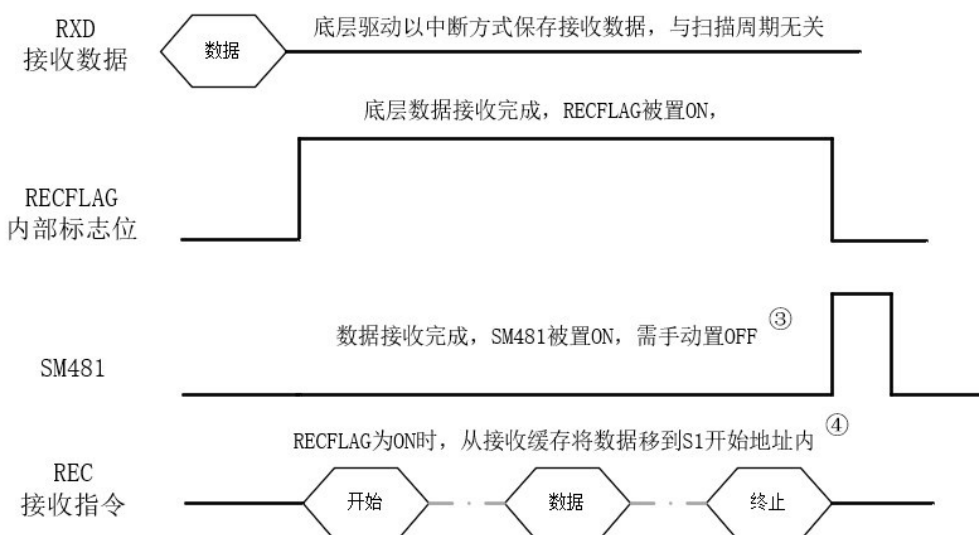
### 单次触发 ②



- S1:接收数据地址从 D0 开始
- S2:接收 D200(数值) 个数据, 若 D200=0, 不接收。比如要接收 D2~D28, 则 D200=27, 不管数据缓冲是 8 位还是 16 位格式。(后台需要根据缓冲位处理好实际接收的字节数)
- S3:指定 COM5 接收。范围 K1~K5, 代指 COM2~COM5.
- 相应端口未使能下, 使用 REC 指令, PLC 系统应报故障码:  
7602: 自由协议未设定, REC 指令不可执行

注意: K1~K5 任一端口未设定自由协议时, 使用 REC 指令, 均报 7602 故障码

### 时序图:



- ① M0 常 ON 时, 每扫描周期 REC 指令被触发一次。REC 指令内部判断 RECFLAG, RECFLAG 为 OFF 时, S1 开始的地址内数据为 0x00。当 RECFLAG 为 ON 时, 接收缓存数据被取

- 出，清除接收缓存，RECFLAG 置 OFF，SM481 置 ON。
- ② M0 闭合一次，REC 指令被触发一次。REC 指令内部判断 RECFLAG，RECFLAG 为 OFF 时，S1 开始的地址内数据为 0x00。当 RECFLAG 为 ON 时，接收缓存数据被取出，清除接收缓存，RECFLAG 置 OFF，SM481 置 ON。
- ③ 一帧数据接收完(包括接收错误情况，错误代码在 SD481 中查看)或者接收超时(指超过重发次数)，SM481 会被置 ON。
- ④ 当用户需要接收的数据长度大于实际接收到的数据长度时，差值用 0x00 填充。当用户需要接收到的数据长度小于实际接收到的数据长度时，多出的数据将会从首位开始填充接收数组。

注：接收超时会发生“重试”动作，超时时间可设置范围 0~6000mS，其中设置为 0 时表示该通道不需要接收或者接收在发送前。

### 3.6.3 释放串口 RLS

将指定的串口资源进行释放的指令。在进行自由格式通信时，如果超时时间设定过长，或者重复次数过多，可通过 RLS 指令立即释放占用的串口资源，清空错误标志(串行通信发生重试/串行通信发生超时)和清空缓存以便进行其他通信操作。

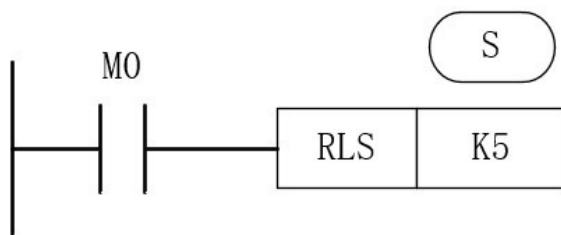
RLS S		发送数据	范围	数据类型
S	通信端口	代指串口号	K1~K5	S16

操作数：

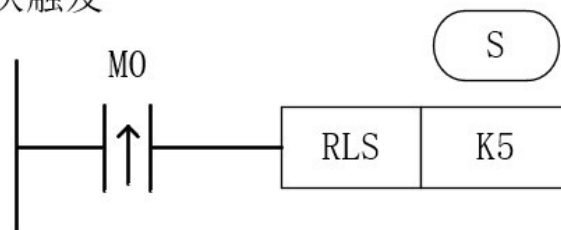
操作数	位	字		双字		常数	
	X、Y、M、L、SM、F、B、SB、S	T、ST、C、D、W、SD、SW、R	Z	LC	LZ	K、H	E
S	—	—	—	—	—	○	—

指令式样：

### 连续触发 ①



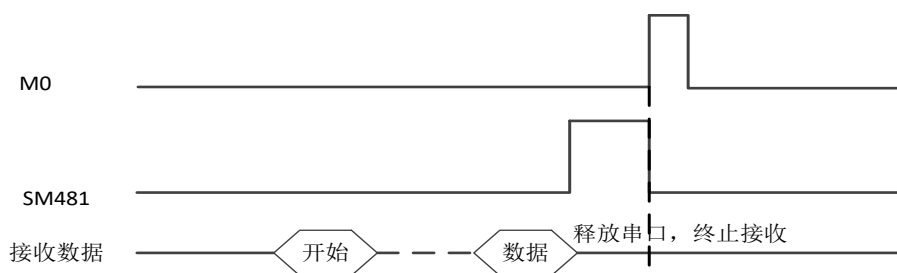
### 单次触发 ②



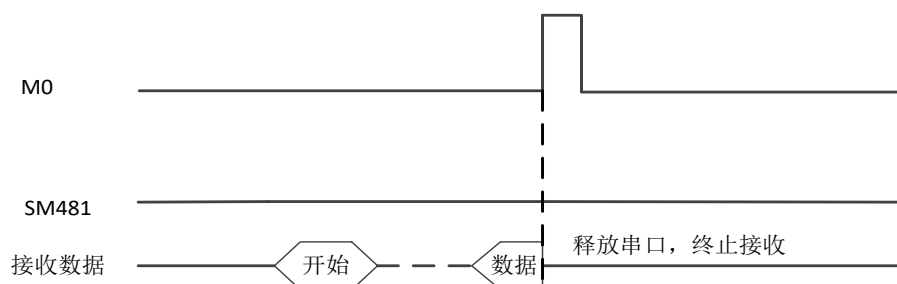
● S:释放 COM5。范围 K1~K5, 代指 COM2~COM5。

释放串口时，将接收完成标志位“SM481”或正在发送标志“SM480”置 OFF。

接收已完成：

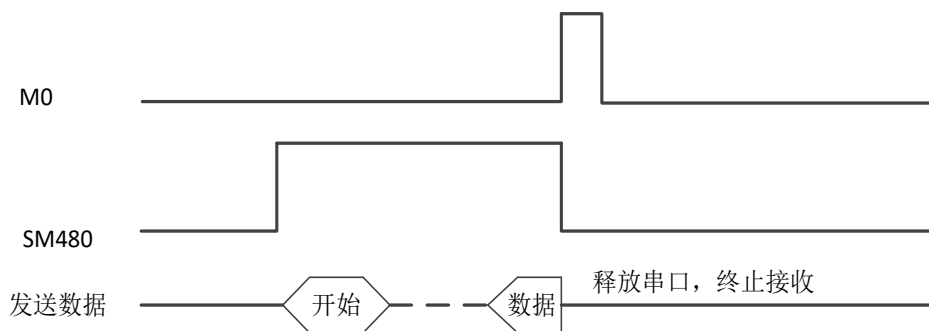


接收未完成：

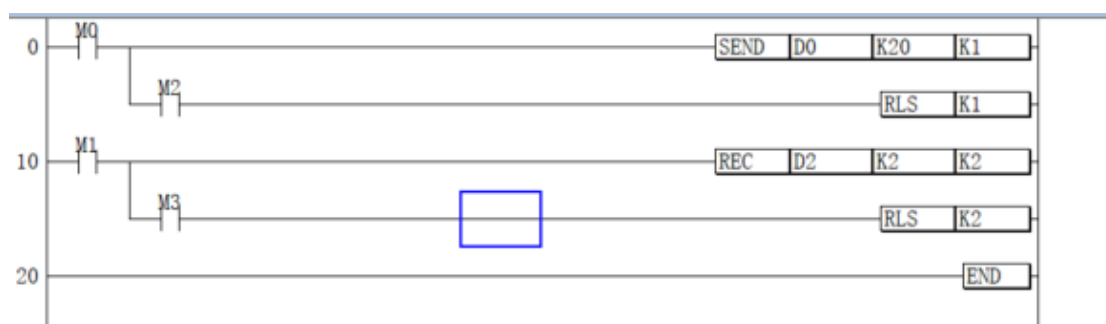


处于发送时使用释放指令的时序图：





### 3.7.应用举例



当 M0 闭合后，COM2 口将会向所连接的设备发送以 D0 为起始地址，且总数为 20 个的数据，即 D1-D21 中的数据内容。开始发送数据时，COM2 发送中标志位 SM400 将会在发送中置 ON。如果在数据发送未完成时闭合 M2，未完成的数据将会立即终止发送。如若数据正常发送完成 COM2 发送中标志位 SM400 将会在发送完成后置 OFF。

当 M1 闭合后，COM2 口将会从所连接的设备处接收 2 个数据，接着将数据内容保存到 D3-D4 中。如果在数据接收未完成时闭合 M3，未完成接收的数据将会立即终止接收。当接收超过重试次数，或者接收正常完成时，COM2 接收完成标志位将会在此时置 ON，需要用户手动置 OFF。

## 4.MODBUS 简介

### 4.1 概要

本章对有关 HCR 系列的 MODBUS 串行通信的内容进行说明。

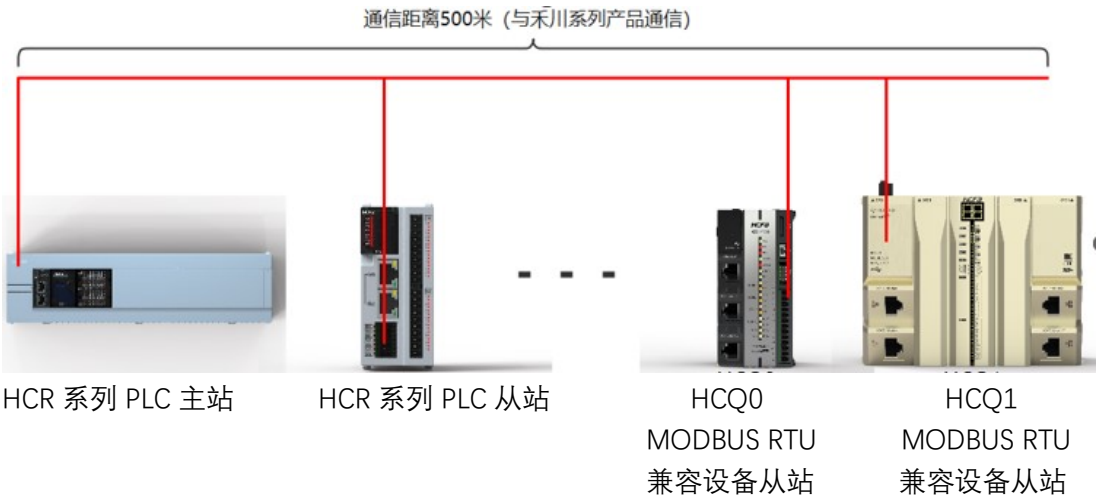
HCR 系列的 MODBUS 串行通信功能通过 1 台主站，在 RS-485 通信时可控制 32 个从站。

- 对应主站功能及从站功能，1 台 HCR 系列可同时使用为主站及从站。(但是，主站仅为单通道)
- 在主站中，使用 MODBUS 串行通信专用顺控命令控制从站。
- 通信协议支持 RTU 模式。

各设备支持 MODBUS 的接口

设备	R1	R2	R8	R8C	R8P
COM1	RS485 非隔离	RS485 非隔离	RS485	RS485	RS485
COM2	BD 扩展口	BD 扩展口	RS485	RS485	RS485
MODBUS	支持	支持	支持	支持	支持

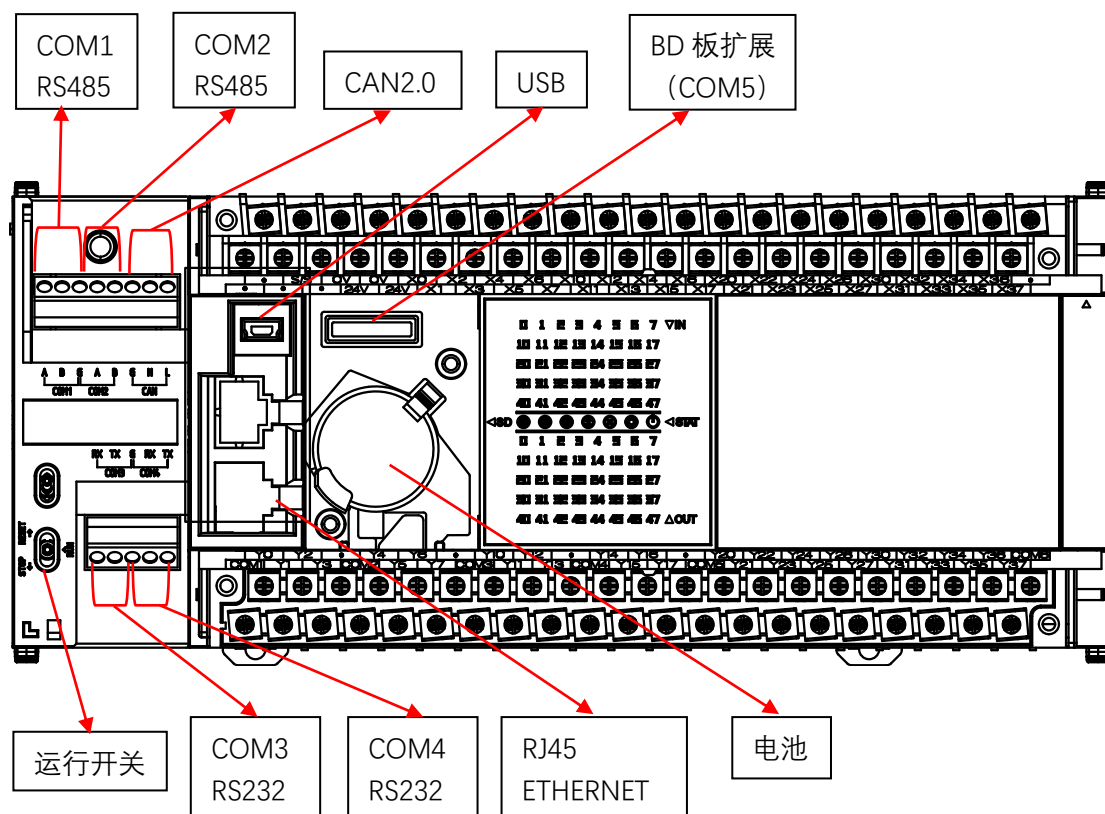
系统连接示意图



4.2 系统配置

以下对有关使用 MODBUS 串行通信所需的系统配置的概要内容进行说明

2.1 HCR8/HCR8P 的端口配置图



目前 R 系列的 MODBUS 仅可在 COM1 和 COM2 上使用。其中 COM1 默认为 MODBUS-RTU 从站，COM2 默认为不使用。

#### 4.2.1 HCR8C 的端口配置图

略

#### 4.2.2 HCR1/HCR2 的端口配置图

略

### 4.3.规格

本章对有关 MODBUS 串行通信功能的规格的内容进行说明。

#### 4.3.1 通信规格

按照以下规格执行 MODBUS 串行通信，波特率等内容是通过 HCP WorkS2 的参数进行设置的。

项目		规格
通信规格	波特率	9600/19200/38400/57600/115200/25600bps
	数据长度	8bit (有些从设备的数据位实际上包括了校验位, 加上校验位就是 9 位)
	停止位	1bit/2bit
	校验位	可选“无(None)”、“奇校验 (Odd)”、“偶校验 (Even)”三种
	传送距离	500 米
	通信格式	RTU
主站功能	可连接的从站数	32 站 (因从站类型有差异)
	功能数	待确认
	同时传送的信息数	1 个信息
	最大写入数	123 字或 1968 线圈
	最大读取数	125 字或 2000 线圈
从站功能	功能数	待确认
	可同时接收的信息数	1 个信息
	站号	1~247

### 4.3.2 链接时间

略

## 4.4MODBUS 规格

### 4.4.1MODBUS RTU 帧格式

使用 RTU 模式, 消息发送至少要以 3.5 个字符时间的停顿间隔开始。在网络波特率下多样的字符时间, 这是最容易实现的。传输的第一个域是设备地址。可以使用的传输字符是十六进制的 0...9,A...F。网络设备不断侦测网络总线, 包括停顿间隔时间内。当第一个域 (地址域) 接收到, 每个设备都进行解码以判断是否发往自己的。在最后一个传输字符之后, 一个至少 3.5 个字符时间的停顿标定了消息的结束。一个新的消息可在此停顿后开始。

整个消息帧必须作为一连续的流转输。如果在帧完成之前有超过 1.5 个字符时间的停顿时间, 接收设备将刷新不完整的消息并假定下一字节是一个新消息的地址域。同样地, 如果一个新消息在小于 3.5 个字符时间内接着前个消息开始, 接收的设备将认为它是前一消息的延续。这将导致一个错误, 因为在最后的 CRC 域的值不可能是正确的。一典型的消息帧如下所示:

起始位	设备地址	功能代码	数据	CRC 校验	结束符
3.5 个字符时间以上的间隔	8Bit	8Bit	n 个 8Bit	16Bit	3.5 个字符时间以上的间隔

### 4.4.2 支持功能一览

所对应的 MODBUS 标准功能如下所示。

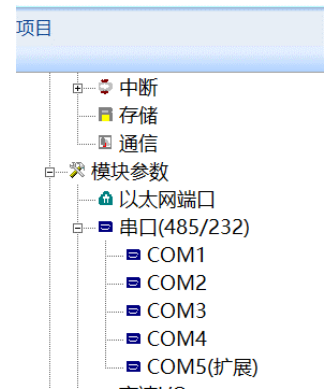
功能代码	功能名	详细内容	1 个报文可访问的 软元件数	广播
01H	线圈读取	线圈读取 （可以多点）	1 ~ 2000 点	×
02H	输入读取	输入读取 （可以多点）	1 ~ 2000 点	×
03H	保持寄存器读取	保持寄存器读取 （可以多点）	1 ~ 125 点	×
04H	输入寄存器读取	输入寄存器读取 （可以多点）	1 ~ 125 点	×
05H	1 线圈写入	线圈写入 （仅 1 点）	1 点	✓
06H	1 寄存器写入	保持寄存器写入 （仅 1 点）	1 点	✓
0FH	多线圈写入	多点的线圈写入	1 ~ 1968 点	✓
10H	多寄存器写入	多点的保持寄存器写入	1 ~ 123 点	✓

## 4.5 通信设置

本章对有关 R 系列 PLC 在使用 MODBUS 串行通信时所需的设置方法的内容进行说明。

### 4.5.1 通信参数的设置

在导航界面点击“模块参数”->“串口（485/232）”会弹出串口端口设置界面，如下图：



目前仅有 COM1,COM2 可使用 MODBUS\_RTU 主站或者从站协议，其界面如下图所示。其中 COM1 默认为 MODBUS\_RTU 从站,COM2 默认为不使用。2 者的通信协议类型，用户可选择 不使用，MODBUS\_RTU 主站，MODBUS\_RTU 从站，或者自由协议。（自由协议详见串行通信篇）

项目	设置
协议格式	MODBUS_RTU 主站、MODBUS_RTU 从站、不使用、自有协议
奇偶校验	无/奇数/偶数
停止位	1bit/2bit
波特率	9600bpS/19200bpS/38400bpS/57600bpS/115200bpS/25600bps
数据位	固定 8 位。(注：有些从设备的数据位实际上包括了校验位，加上校验位就是 9 位)

备注：只有 PLC 的这些设置和目标设备的通信设置相同方可匹配通信。

固有设置是用来设置一些固有的参数，MODBUS\_RTU 主站有如下参数需要设置：

通信超时：是指 PLC 发送请求后收不到回应，等待重发的时间。如果设置回复超时 300ms，则默认通信时，PLC 等待对方回应 300ms，时间到未收到回应，则重新发送请求。

重试次数：通信时 PLC 未收到回复时，则重新发送请求，若超过重发次数，则跳过当前发

送。并记录到 SM433 接收超时。接收正常后 SM433 自动清零。

请求延迟：主站发送报文前，插入一个固定的延迟，目的是为了让从站更容易接收到数据。

广播延迟：???

注：参数设置好后，重新上电才能生效。

广播模式时，通信超时与重试次数的设置和相应报错无效。

4.5.2 从站软元件映射

点击软元件映射弹出如下界面

串口端口设置

COM1 软元件映射

项目	内容	线圈	输入	输入寄存器	保持寄存器
分配01	软元件	M0	X0		D0
	起始MODBUS软元件	0	0		0
	点数	10000	256		9000
分配02	软元件	Y0			SD0
	起始MODBUS软元件	10000			10000
	点数	256			6000
分配03	软元件	S0			
	起始MODBUS软元件	11000			
	点数	6000			
分配04	软元件				
	起始MODBUS软元件				
	点数				
分配05	软元件				
	起始MODBUS软元件				
	点数				
分配06	软元件				
	起始MODBUS软元件				
	点数				

确定 取消

串口端口设置

COM2 软元件映射

项目	内容	线圈	输入	输入寄存器	保持寄存器
分配01	软元件	M0	X0		D0
	起始MODBUS软元件	0	0		0
	点数	10000	256		9000
分配02	软元件	Y0			SD0
	起始MODBUS软元件	10000			10000
	点数	256			6000
分配03	软元件	S0			
	起始MODBUS软元件	11000			
	点数	6000			
分配04	软元件				
	起始MODBUS软元件				
	点数				
分配05	软元件				
	起始MODBUS软元件				
	点数				
分配06	软元件				
	起始MODBUS软元件				
	点数				

确定 取消

其中已有系统默认的分配，可自行更改。

设置项目	内容
分配 1~16	各 MODBUS 软元件的分配可在 1~16 范围内进行设置
软元件	设置分配软元件的种类和起始编号
起始 MODBUS 软元件号	设置起始 MODBUS 软元件编号

分配点数	设置分配点数
------	--------

关于可使用的软元件：

软元件一览			可分配的 MODBUS 软元件			
软元件种类	软元件标记		线圈	输入	输入寄存器	保持寄存器
特殊继电器	SM		○	○	○ <sup>*1</sup>	○ <sup>*1</sup>
特殊寄存器	SD		—	—	○	○
输入	X		○	○	○ <sup>*1</sup>	○ <sup>*1</sup>
输出	Y		○	○	○ <sup>*1</sup>	○ <sup>*1</sup>
内部继电器	M		○	○	○ <sup>*1</sup>	○ <sup>*1</sup>
锁存继电器	L		○	○	○ <sup>*1</sup>	○ <sup>*1</sup>
报警器	F		○	○	○ <sup>*1</sup>	○ <sup>*1</sup>
链接继电器	B		○	○	○ <sup>*1</sup>	○ <sup>*1</sup>
数据寄存器	D		—	—	○	○
链接寄存器	W		—	—	○	○
定时器	线圈	TC	○	○	○ <sup>*1</sup>	○ <sup>*1</sup>
	触点	TS	○	○	○ <sup>*1</sup>	○ <sup>*1</sup>
	当前值	TV	—	—	○	○
累积定时器	线圈	STC	○	○	○ <sup>*1</sup>	○ <sup>*1</sup>
	触点	STS	○	○	○ <sup>*1</sup>	○ <sup>*1</sup>
	当前值	STV	—	—	○	○
计数器	线圈	CC	○	○	○ <sup>*1</sup>	○ <sup>*1</sup>
	触点	CS	○	○	○ <sup>*1</sup>	○ <sup>*1</sup>
	当前值	CV	—	—	○	○
长计数器	线圈	LC	○	○	○ <sup>*1</sup>	○ <sup>*1</sup>
	触点	LCS	○	○	○ <sup>*1</sup>	○ <sup>*1</sup>
	当前值	LCV	—	—	○	○
链接特殊继电器	SB		○	○	○ <sup>*1</sup>	○ <sup>*1</sup>
连接特殊寄存器	SW		—	—	○	○
步进继电器	S		○	○	○ <sup>*1</sup>	○ <sup>*1</sup>
变址寄存器	Z		—	—	○	○
变址寄存器	LZ		—	—	○ <sup>*2</sup>	○ <sup>*2</sup>
文件寄存器	R		—	—	○	○

\*1 软元件编号及分配点数应设置为 16 的倍数。如果不是 16 的倍数，HCP WorkS2 会发生参数设置错误。

\*2 MODBUS 软元件为长型软元件，因此需要使用 2 点分配点数。

注

- 1) 不能在线圈和输入中设置相同的软元件。
- 2) 不能再输入寄存器和保持寄存器中设置相同的软元件。
- 3) 指定的起始软元件编号+分配点数超过可编程控制器软元件的有效范围时，HCP WorkS2 会发生参数设置出错。



# 4.6 功能

本章对有关 MODBUS 串行通信的功能的内容进行说明。

## 4.6.1 读写指令 ADPRW

主站功能中，使用 ADPRW 命令与从站进行通信

ADPRW S1 S2 S3 S4 (S5/D1) D2		主站读写	范围	数据类型
S1	从站站号	0 <sup>*1</sup> ~247	K0~K247	S16
S2	功能代码	参照下文	01H ~ 06H、0FH、10H	S16
S3	与功能代码相应的功能参数	参照下文	0 ~ FFFFH	S16
S4	与功能代码相应的功能参数	参照下文	1 ~ 2000	S16
S5/D1	与功能代码相应的功能参数	--	--	Bit/S16
D2 <sup>*2</sup>	输出通信执行状态的起始位软元件编号	--	--	Bit

\*1 0 为广播模式。

\*2 指定为 (D2) 的软元件在起始处占用 3 点。注意避免与用于其他控制的软元件重复

注: u16 无符号 16 位数，S16 有符号 16 位数，u32 无符号 32 位数，S32 有符号 32 位数,后同。

操作数

操作数	位	字		双字		常数	
	X、Y、M、L、SM、F、B、SB、S	T、ST、C、D、W、SD、SW、R	Z	LC	LZ	K、H	E
S1	—	○ <sup>*1</sup>	○	—	—	○	—
S2	—	○ <sup>*1</sup>	○	—	—	○	—
S3	—	○ <sup>*1</sup>	○	—	—	○	—
S4	—	○ <sup>*1</sup>	○	—	—	○	—
S5/D1		○ <sup>*1</sup>	○	—	—	○	—
D2		○	—	—	—	—	—

\*1 不能使用 T、ST、C。

功能码和参数说明

S1: 从站号，0 为广播模式，其余从站号 1~127。

(S2)，(S3)、(S4)、(S5) / (D1) 的参数分配如下所示

S2: 功能代码	S3:MODBUS 地址	S4: 访问点数	S5/D1: 数据存储软元件起始
----------	--------------	----------	------------------

	对象软元件② (见下表详细说明)		
01H 线圈读取	MODBUS 地址: 0000H ~ FFFFH	访问点数: 1 ~ 2000	读取数据存储软元件起始
			对象软元件      字软元件① 位软元件③
			占用点数      字软元件 $(S4+15) \div 16$ 点*1 位软元件 (S4) 点
02H 输入读取	MODBUS 地址: 0000H ~ FFFFH	访问点数: 1 ~ 2000	读取数据存储软元件起始
			对象软元件      字软元件① 位软元件③
			占用点数      字软元件 $(S4+15) \div 16$ 点*1 位软元件 (S4) 点
03H 保持寄存器读取	MODBUS 地址: 0000H ~ FFFFH	访问点数: 1 ~ 125	读取数据存储软元件起始
			对象软元件      ①
			占用点数      (S4) 点
04H 输入寄存器读取	MODBUS 地址: 0000H ~ FFFFH	访问点数: 1 ~ 125	读取数据存储软元件起始
			对象软元件      ①
			占用点数      (S4) 点
05H 线圈写入	MODBUS 地址: 0000H ~ FFFFH	0 (固定)	写入数据存储软元件起始
			对象软元件*2      字软元件② 位软元件③
			占用点数      1 点
06H 保持寄存器写入	MODBUS 地址: 0000H ~ FFFFH	0 (固定)	写入数据存储软元件起始
			对象软元件      ②
			占用点数      1 点
0FH 多点的线圈写入	MODBUS 地址: 0000H ~ FFFFH	访问点数: 1 ~ 1968	写入数据存储软元件起始
			对象软元件      字软元件② 位软元件③
			占用点数      字软元件 $(S4+15) \div 16$ 点*1 位软元件 (S4) 点
10H 多点保持寄存器 写入	MODBUS 地址: 0000H ~ FFFFH	访问点数: 1 ~ 123	写入数据存储软元件起始
			对象软元件      ②
			占用点数      1 点

\*1 舍去尾数。

\*2 最低位的位为 0 时位 OFF，为 1 时位 ON。

对象软元件表

No.	对象软元件
①	T、ST、C、D、R、W、SW、SD、标签软元件
②	T、ST、C、D、R、W、SW、SD、标签软元件、K、H
③	X、Y、M、L、B、F、SB、S、SM、标签软元件

## D2:通信执行状态输出软元件

各通信状态相应的动作时间和同时动作的特殊继电器如下所示。

操作数	动作时间	同时动作的特殊继电器
D2	命令动作时 ON, 命令执行中以外 OFF	SM810
D2+1 <sup>*1</sup>	命令正常结束时 ON, 命令开始时 OFF	SM29 <sup>*2</sup>
D2+2 <sup>*1</sup>	命令异常结束时 ON, 命令开始时 OFF	SM29 <sup>*2</sup>

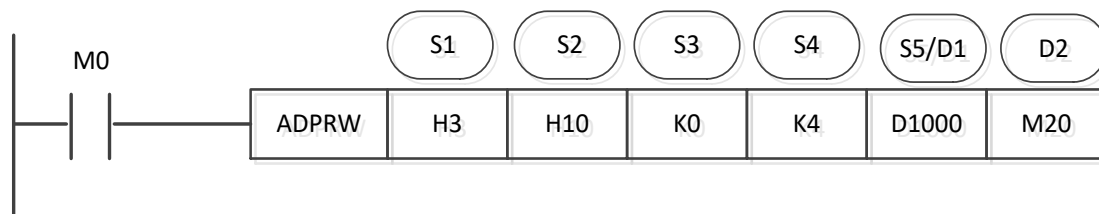
\*1 D2+1 在命令正常结束时为 ON, D2+2 在命令异常结束时为 ON, 因此可辨别正常或异常。

\*2: SM29 也使用于 MODBUS 通信以外的命令的执行结束标志。(定位命令等)

使用 SM29 时, 应在确认命令执行结束的命令的正下方使用触点。

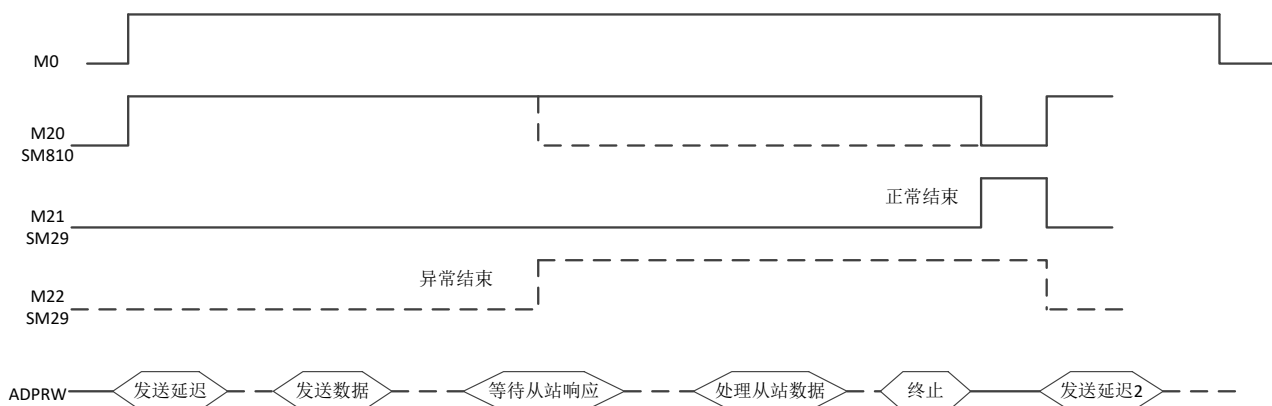
注: 通信过程中遇到错误后, 只要前置触点不断开, 下个周期指令会继续执行不会主动停止。

执行条件: 连续触发



- MODBUS 主站读写指令, M0 连续触发。
- S1: 从站地址 03H。
- S2: 功能码 10H, 多点保持寄存器写入。
- S3: K0, 写入保持寄存器的起始地址。
- S4: 写入点数 4
- S5: 从站保持寄存器地址 0~3 写入 PLC 的 D1000~D1003 的值
- D2: M20, 命令动作开始, M21, 命令正常结束, M22 命令异常结束。

时序图:



其中异常结束包括超过重发次数没有收到反馈, 或者收到错误反馈等, 详见 SD430,

## 4.6.2 特殊软元件

### 特殊继电器

COM1	COM2	COM3	COM4	COM5	名称	详细内容	R/W
SM29					命令执行结束	主站命令执行结束后为 ON。	R
SM400	SM420	SM440	SM460	SM480	自由格式通讯发送中标志	指令开始执行时，置 ON 发送完成时，置 OFF	R
SM401	SM421	SM441	SM461	SM481	自由格式通讯接收完成标志	接收到一帧数据或接收数据超时 时，置 ON； 需要用户程序置 OFF	R
SM402	SM422	SM442	SM462	SM482	自由格式通讯缓冲位数	8 位 OFF 16 位 ON	R
SM410	SM430	SM450	SM470	SM490	MODBUS 通信出错	发生 MODBUS 通信出错时为 ON。	R
SM411	SM431	SM451	SM471	SM491	MODBUS_RTU 通信中	从开始执行命令直到命令执行结束标志 ON 为止，MODBUS 串行通信中为 ON。	R
SM412	SM432	SM452	SM472	SM492	串行通信发生重试	从站在超时设置时间内无响应时，在主站发送重试的期间为 ON。	R/W
SM413	SM433	SM453	SM473	SM493	串行通信发生超时	发生响应超时时为 ON。	R/W

注：自由格式通信专指自由格式通信时有效，modbus 通信专指 modbus 通信有效，串行通信指所有的通信协议都有效。下同。

#### MODBUS 通信出错

请勿用程序或工程工具使其置为 ON。

该软元件在通信恢复正常时也不会 OFF。电源 OFF→ON、复位、STOP→RUN、SM50 （解除出错）置为 ON 时或执行下一 ADPRW 命令时会被清除

### 特殊寄存器

COM1	COM2	COM3	COM4	COM5	名称	详细内容	R/W
SD400	SD420	SD440	SD460	SD480	自由格式通讯发送结果	0: 正确 2: 长度错误 10: 自由格式发送缓冲区溢出	R/W

SD401	SD421	SD441	SD461	SD481	自由格式通讯接收结果	0: 正确 10: 发送数据长度溢出 11: 接收数据短 12: 接收数据长 13: 接收错误 14: 接收超时 15: 无起始符 16: 无终止符	R/W
SD402	SD422	SD442	SD462	SD482	自由格式通讯接收数据个数	按字节计, 不包含起始符、终止符	R
SD403	SD423	SD443	SD463	SD483	串行通信设置	设置在 PLC 中的通信属性会被记录。	R
SD404	SD424	SD444	SD464	SD484	串行通信动作模式显示	0: 通道: 未启用 1: 下载/HMI 监控协议 2: 3: 5: 自由协议通信 9: MODBUS RTU 主站 10: MODBUS RTU 从站	R
SD405	SD425	SD445	SD465	SD485	自由格式通讯起始符、终止符	起始符高 8 位, 终止符低 8 位	R
SD406	SD426	SD446	SD466	SD486	自由格式通讯帧超时	设置在 PLC 中的帧超时会被记录。 单位 mS	R
SD407	SD427	SD447	SD467	SD487	串行通信当前的重试次数	因从站响应超时而进行通信重试时, 当前的重试次数会被记录	R
SD408	SD428	SD448	SD468	SD488	串行通信从站响应超时 (通讯超时)	从站响应超时的设置值会被记录。 单位 mS	R
SD409	SD429	SD449	SD469	SD489	串行通信超时时的重试次数	超时时的重试次数的设置值会被记录。	R
SD410	SD430	SD450	SD470	SD490	MODBUS 通信出错代码	在 MODBUS 串行通信中发生的最新出错代码会被记录。	R
SD411	SD431	SD451	SD471	SD491	MODBUS 通信出错的详细内容	最新出错的详细内容会被记录。	R
SD412	SD413	SD414	SD415	SD416	本站站号	设置的从站站号会被记录	R

#### 串行通信设置

用于读取设置串行通信的格式 (只读)

位	名称	内容	
		0 (bit = OFF)	1 (bit = ON)
b0	—	—	—

b1、b2	奇偶位	(b2、b1) = (0、0)：无 (b2、b1) = (0、1)：奇数 (b2、b1) = (1、1)：偶数	
b3	停止位	1bit	2bit
b4 ~ b7	波特率 (bpS)	(b7、b6、b5、b4) = (1、0、0、0)：9600 (b7、b6、b5、b4) = (1、0、0、1)：19200 (b7、b6、b5、b4) = (1、0、1、0)：38400 (b7、b6、b5、b4) = (1、0、1、1)：57600 (b7、b6、b5、b4) = (1、1、0、1)：115200 (b7、b6、b5、b4) = (1、1、1、1)：256000	
b8 ~ b15	—	—	—

## 命令执行结束

确认命令执行结束的软元件。

通道 1 (从站)	通道 2 (主站)	内容	R/W
SM29		命令执行结束后为 ON。	R

## 注意事项

请勿用程序或工程工具使其置为 ON。

电源 OFF→ON、复位、STOP→RUN 或执行下 ADPRW 命令时会被清除。

## 串行通信设置

用于读取设置串行通信的格式（只读）

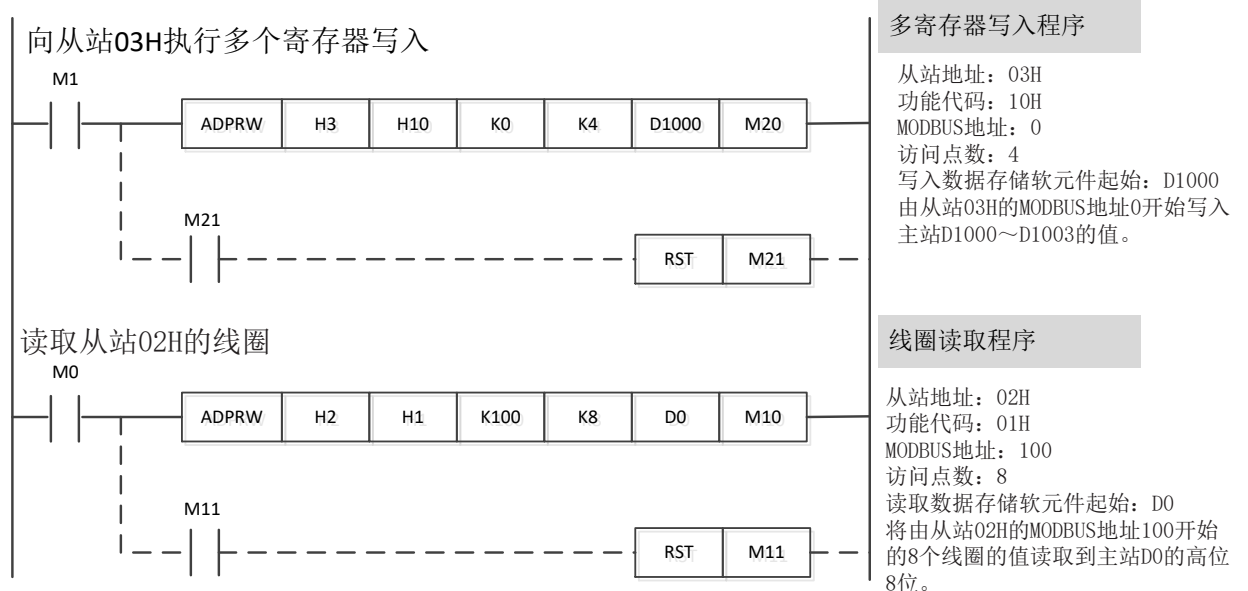
位	名称	内容	
		0 (bit = OFF)	1 (bit = ON)
b0	—	—	—
b1、b2	奇偶位	(b2、b1) = (0、0)：无 (b2、b1) = (0、1)：奇数 (b2、b1) = (1、1)：偶数	
b3	停止位	1bit	2bit
b4 ~ b7	波特率 (bpS)	(b7、b6、b5、b4) = (1、0、0、0)：9600 (b7、b6、b5、b4) = (1、0、0、1)：19200 (b7、b6、b5、b4) = (1、0、1、0)：38400 (b7、b6、b5、b4) = (1、0、1、1)：57600 (b7、b6、b5、b4) = (1、1、0、1)：115200	
b8 ~ b15	—	—	—

注：从站

## 4.7 编程

本章中对有关 MODBUS 串行通信中主站程序的编写示例的内容进行说明。

### 4.7.1 主站程序的编写举例



指令中的虚线部分程序用户可以不写，可以由后台自动识别完成当前通信。

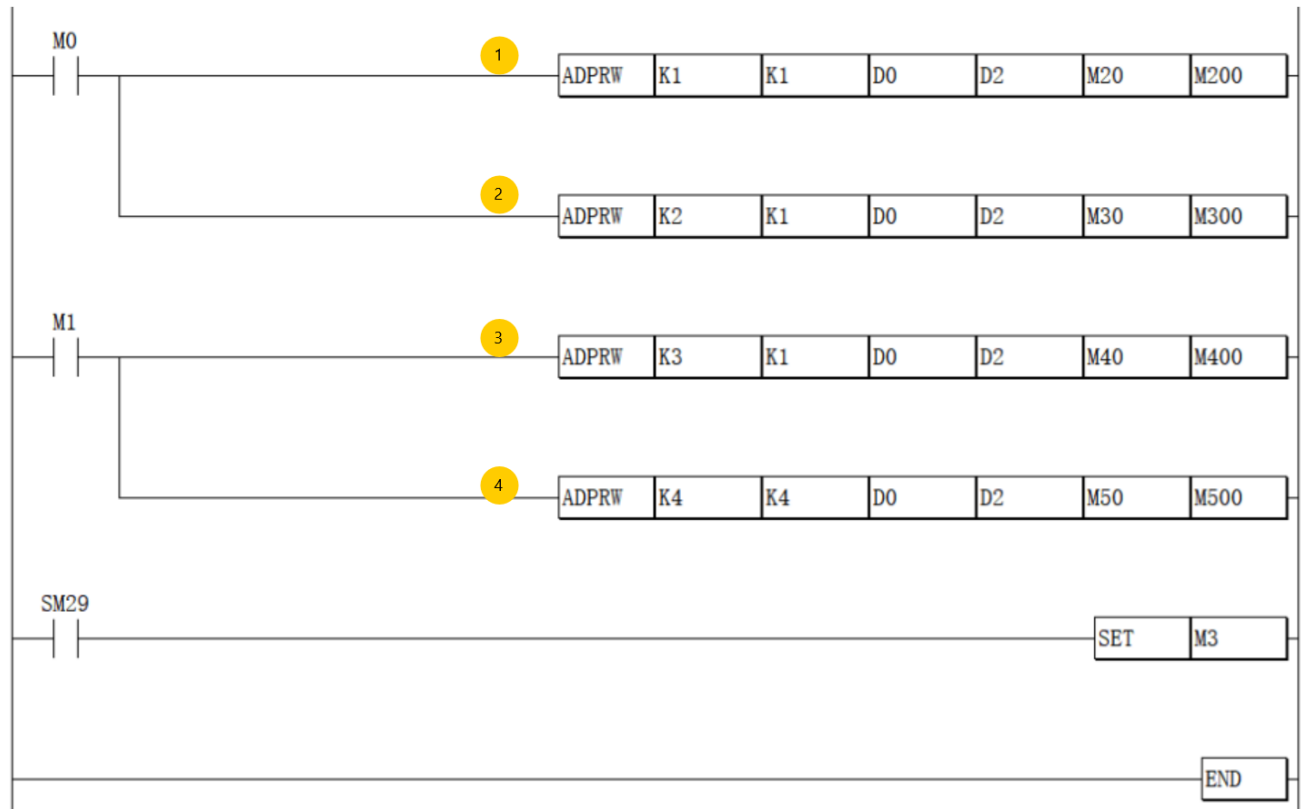
注意：

- 请勿在 ADPRW 命令结束前将驱动触点置为 OFF。(如错误操作可能导致当前通信出错)
- 可以触发多条 ADPRW,但是不要超过后台缓冲区。
- 主站线程独立不占用扫描时间。

注意：

- 请勿在 ADPRW 命令结束前将驱动触点置为 OFF。指令未执行完成断开前置触点指令执行状态将被设置为异常，当前置触点再次设置为 ON 时，执行动作状态变为执行中。
- 可以出发多条 ADPRW,但是不要超过后台缓冲区。
- 同时驱动多个 ADPRW 命令时，后台自动轮询发送每条指令。
- 使用线圈读取功能或输入读取功能，并在读取目标软元件中指定字软元件时，仅通过 ADPRW 命令的访问点数所指定的位会被改写。字软元件的剩余位不会变化
- 主站线程独立不占用扫描时间。

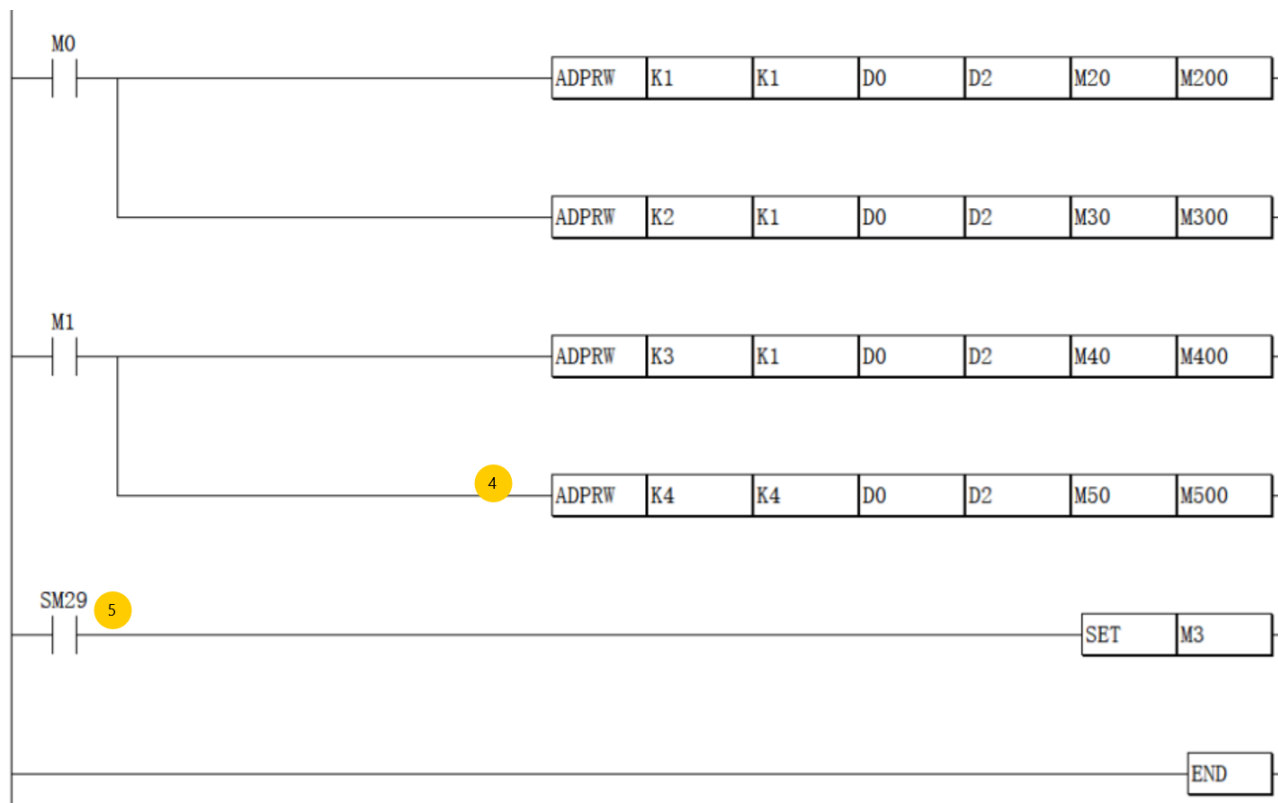
(2) 同时驱动多个 ADPRW 命令，时序由后台控制。



- 程序执行中，M0 (OFF->ON) 时，指令(1) (ADPRW) 的状态重新设置为执行中 (M200 = 1, M201=0, M202=0)，无论之前的状态是执行完成状态，还是异常状态。
- 程序执行中，M0 (OFF->ON) 时，指令(1) (ADPRW) 执行开始，指令(1)完成之后，状态设置为执行完了 (M200 =0, M201=1, M202=0)，指令(2)执行开始，直到指令(2)执行完了(异常完了)之后再次执行指令(1.)
- 程序执行中，M0 (OFF->ON) 时，指令 (1) 与指令 (2) 交替执行 (指令 (1) 执行 -> 指令 (2) 完了 ->指令 (2) 执行 ->指令 (2) 完了->指令 (1) 执行 ->指令 (2) 完了...)
- 程序执行中指令 (1) 或者指令 (2) 异常完了后，在状态位没有清除前将不再执行。异常状态位清除之后将会被再次执行。
- 程序执行中，M0 和 M1 同时 (OFF-ON) 执行顺序为指令 (1) 执行->指令 (1) 完了 ->指令 (2) 执行->指令 (2) 完了->->指令 (3) 执行->指令 (3) 完了->指令 (4) 执行->指令 (4) 完了。

(3) 运行结束标志 (SM29)。





SM29 在命令执行结束后设置为 ON。上图中位置（5）SM29 仅仅反映指令（4）的状态，当 M1 为 OFF 时指令（4）不会被执行，位置（5）SM29 总为 OFF 状态。

请勿使用跳转指令跳过 ADPRW 指令，如使用跳转指令跳过上图指令（4）到 SM29（5），那么 SM29（5）将不会反映指令（4）的执行状态。

## 4.7.2 编程上注意的事项

- 请勿在 ADPRW 命令结束前将驱动触点置为 OFF。
- 同时驱动多个 ADPRW 命令时，需在当前的命令结束后执行下一 ADPRW 命令。（一次仅可执行 1 个命令）
- 使用线圈读取功能或输入读取功能，并在读取目标软元件中指定字软元件时，仅通过 ADPRW 命令的访问          点数所指定的位会被改写。字软元件的剩余位不会变化。

## 4.8 故障排除

应按如下方法确认通信的故障

### ■ 通过 LED 显示确认通信状态

应确认“TX”、“RX”的 LED 显示状态。

TX	RX	状态
亮灯	亮灯	正在进行数据的收发。

亮灯	熄灯	正在进行数据的接收，但是发送不成功。
熄灯	亮灯	正在进行数据的发送，但是接收不成功。
熄灯	熄灯	数据的收发都不成功

MODBUS 通信正常执行时，两个 LED 都将亮灯。

LED 不亮灯时，应确认配线或主站和各从站的设置情况及出错状态。

#### ■ 安装及配线的确认

应确认通信设备和 PLC、扩展插板以及扩展适配器是否连接牢固。通信设备的连接不稳定时，将无法通信。应确认各通信设备之间的配线是否正确。配线不正确时，将无法通信。

关于安装和配线方法，请参照各通信设备的手册。

#### ■ 参数的确认

应确认通信设置参数是否符合使用用途。不符合使用用途时，通信将无法正确进行。更改设置后，必须将 PLC 的电源由 OFF -> ON 或进行复位。

#### ■ 出错标志

当串行通信中发生通信出错时，串行通信出错标志为 ON。应确认以下软元件是否为 ON。

通道 1	通道 2	名称	内容
SM410	SM430	串行通信出错	串行通信中发生出错时为 ON。

#### ■ 出错代码

通道 1	通道 2	名称	内容
SD410	SD430	串行通信出错代码	串行通信中发生出错时，存储出错代码。
SD411	SD431	串行通信出错的详细内容	串行通信中发生出错时，存储出错的详细内容。

被存储在各软元件中的出错代码（16 进制数）、出错详细内容如下所示。

出错代码	出错详细内容	出错内容	发生出错的站点
7001H	—	使用通道与 MODBUS RTU 以外的通信设置重复	从站
7002H	—	MODBUS RTU 主站缓冲区溢出	主站
7010H	—	奇偶位出错、溢出出错、成帧出错	主站/从站
7302H	发生出错的特殊软元件编号会被存储	MODBUS 通信参数设置异常 (特殊软元件的设置异常)	主站/从站
7304H	—	CRC 出错	主站/从站
7305H	—	字符溢出 接收数据为 256 个字节以上时	主站/从站
7306H	—	报文格式不正确	主站/从站

		接收的字节数与指定字节数不一致	
7307H	—	接收到不支持的功能代码	从站
7308H	—	访问未分配软元件的 MODBUS 地址	从站
7309H	—	从站响应超时 经过通信参数超时中设置的时间后，从站仍无响应	主站
730AH	以下的响应报文格式会被存储。 异常响应功能代码：高位字节 异常响应代码：低位字节	异常响应报文的接收 通过从站响应返回异常响应报文	主站
730BH	以下的响应报文格式会被存储。 请求站号：高位字节 响应站号：低位字节	站号不一致 请求报文的站号和响应报文的站号不一致	主站
730CH	以下的响应报文格式会被存储。 请求功能代码：高位字节 响应功能代码：低位字节	功能代码不一致 请求报文的功能代码和响应报文的功能代码不一致	主站
730DH	响应功能代码会被存储	广播请求出错 读取命令为广播命令	从站
730EH		请求报文数据异常 请求报文中包含超出设置范围的值。	从站

#### ■ 从站异常响应码（标准协议规定）

主站接收到从站的异常响应时，以下的响应报文格式会被存储。

b15 — b8	b7 — b0
异常响应功能代码 (高位字节)	异常响应代码 (低位字节)

异常响应功能代码（高位字节）的详细内容如下所示。

异常响应功能代码	功能名	详细内容
81H	线圈读取	线圈读取时异常结束
82H	输入读取	输入读取时异常结束
83H	保持寄存器读取	保持寄存器读取时异常结束
84H	输入寄存器读取	输入寄存器读取时异常结束
85H	1 线圈写入	线圈写入时异常结束
86H	1 寄存器写入	保持寄存器写入时异常结束
8FH	多线圈写入	多点的线圈写入时异常结束
90H	多寄存器写入	多点的保持寄存器写入时异常结束

异常响应代码（低位字节）的详细内容如下所示。

异常响应代码	异常响应代码名	详细内容
01H	功能代码异常	接收到不支持的功能代码时
02H	软元件地址异常	访问未分配软元件的 MODBUS 地址时
03H	数据异常	请求报文的数据区域不正确时

04H	处理中断	在进行请求报文的处理时，发生致命性出错而导致处理中断时
-----	------	-----------------------------

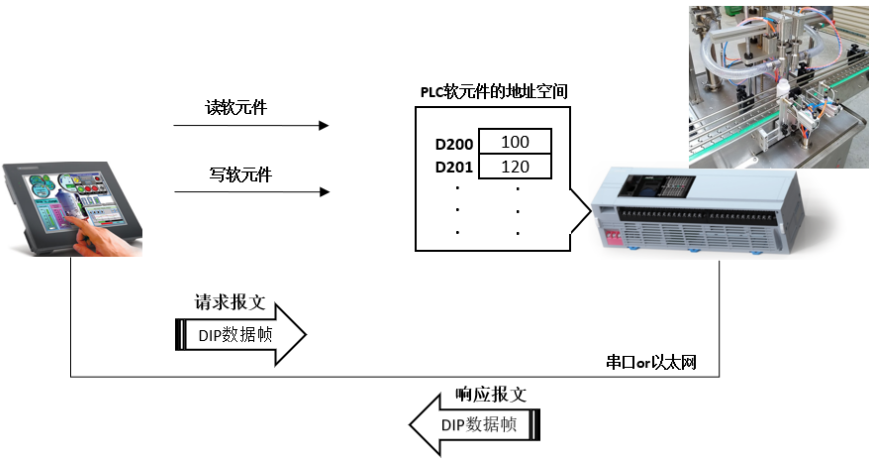
# 5. DIP 简介

## 5.1 概要

设备接口协议（以下简称 DIP 协议）是 R 系列 PLC 的私有通信协议，使用该协议的第三方设备可以对 R 系列 PLC 的软元件进行读写操作。在物理层，DIP 协议支持① RS-232 接口、② RS-422 接口、③ RS-485 接口 这三种串口传输，以及以太网接口传输。

各设备支持 DIP 的串口

设备	R1	R2	R8	R8C	R8P
COM1	RS485 非隔离	RS485 非隔离	RS485	RS485	RS485
COM2	BD 扩展口	BD 扩展口	RS485	RS485	RS485
DIP 协议	支持	支持	支持	支持	支持

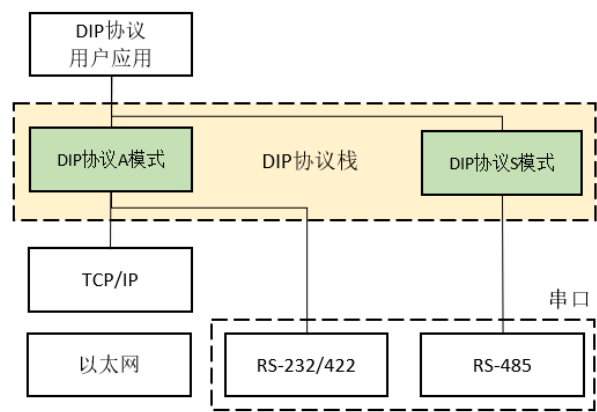


图表 1 使用 DIP 协议读写 PLC 软元件

DIP 协议分为同步通信的 S 模式和异步通信的 A 模式。S 模式和 A 模式的报文编码有所不同。DIP 协议 S 模式运行于 RS-485 接口、A 模式运行于 RS-232/422 串口以及以太网接口的设备间通信。

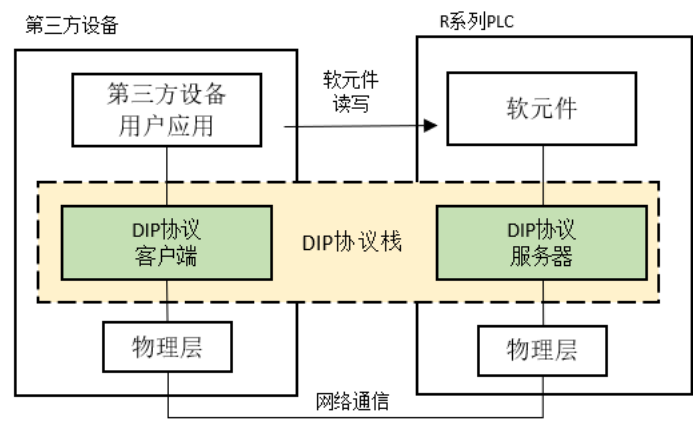
使用 RS-232 接口的 DIP 协议和使用 RS-422 协议的 DIP 协议，除物理层之外协议内容

完全相同。在本手册中如无特殊说明，RS-232 接口相关描述均适用于 RS-422 接口。



图表 2 协议位于 OSI 模型的应用层

DIP 协议采用服务器/客户端通信模式，PLC 的 DIP 协议服务器拥有站号，DIP 协议客户端的用户应用需要指定服务器站号来进行通信。



图表 3 DIP 协议的客户端服务器模式

## 5.2 主要功能

注意，R 系列小型 PLC 当前仅支持 COM1-COM5 口使用 DIP 从站协议（V1.20.0008）。

### 5.2.1 通信

用户通过使用 HCP works2 设置作为服务器的 R 系列 PLC 的站号、通信接口、波特率等必要配置参数。请注意，RS-485 与 RS-232 使用不同的 DIP 协议模式，RS-485 接口使用 S 模式，RS-232 接口使用 A 模式。

表格 1 协议的串口规格

项目		规格
通信接口		RS-232/422      RS-485
通信协议		DIP 协议 A 模式      DIP 协议 S 模式
通信方式		客户端/服务器
连接方式		1: 1      1: N (最大 127 台)
服务器站号		服务器站号=1 (固定)      1-127
通信速度		9600/19200/38400/57600/115200bps
单报文最大写入数		123 字或 1968 位
单报文最大读取数		125 字或 2000 位
数据形式	数据长	8bit
	停止位	1bit
	奇偶校验位	偶/奇/无
帧错误		CRC16

以太网通信使用 DIP 协议的 A 模式。使用 HCP works2 设置作为服务器的 PLC 的站号。  
DIP 协议端口号固定为 8080。

表格 2 DIP 协议的以太网接口规格

项目		规格
通信接口		RJ-45
通信协议		DIP 协议 A 模式
通信方式		N: N
服务器站号		服务器站号=1 (固定)
单报文最大写入数		123 字或 1968 位
单报文最大读取数		125 字或 2000 位
TCP/IP	服务器的最大连接数	8 个
	服务器端口号	8080
	Socket 连接方式	长连接和短连接

## 5.2.2 读写软元件

R 系列 PLC 的软元件，除了 B/SB/W/SW/Z/LZ 以外的类型，均可通过 DIP 协议对软元件的全部地址空间进行读写访问。

由于 T/ST/C/LC 软元件兼有 bit 类型和 word 类型值，DIP 协议通过不同的功能码分别访问这些软元件的 bit 类型和 word 类型。

读写 dword 型的 LC 软元件的报文中，dword 数据按照从地址低位 word 到地址高位 word 的顺序排列通信；word 型的数据按照从地址低位 byte 到地址高位 byte 的顺序排列通信。

注意：R 系列 PLC 允许用户调整不同种类的软元件实际地址空间，客户端的应用程序请按照所访问的服务器实际地址空间进行编程。

表格 3 可访问软元件类型和地址空间

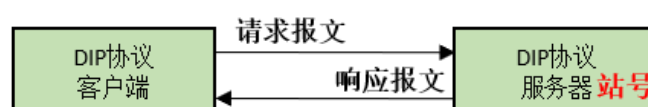
R 系列 PLC 软元件		软元件数据			访问	机型例
类型	名称	类型	标记	地址空间	许可	R 系列
X	输入	bit	8 进制	X0-X377	✓	128 点
Y	输出	bit	8 进制	Y0-Y377	✓	128 点
M	内部继电器	bit	10 进制	M0-M10239	✓	10240 点
B	链接继电器	bit	16 进制	B0-B27FF	✗	10240 点
SB	链接特殊继电器	bit	16 进制	SB0-SB27FF	✗	10240 点
F	报警器	bit	10 进制	F0-F10239	✓	10240 点
S	步进继电器	bit	10 进制	S0-S10239	✓	4096 点
L	锁存继电器	bit	10 进制	L0-L10239	✓	10240 点
T	定时器	bit/word	10 进制	T0-T10239	✓	2048 点
ST	累计定时器	bit/word	10 进制	ST0-ST10239	✓	2048 点
C	计数器	bit/word	10 进制	C0-C10239	✓	1024 点
LC	长计数器	bit/dword	10 进制	LC0-LC10239	✓	1024 点
D	数据寄存器	word	10 进制	D0-D10239	✓	10240 点
W	链接寄存器	word	16 进制	W0-W27FF	✗	10240 点
SW	链接特殊寄存器	word	16 进制	SW0-SW27FF	✗	10240 点
R	文件寄存器	word	10 进制	R0-R10239	✓	32768 点
SM	特殊继电器	bit	10 进制	SM0-SM16383	✓	12288 点
SD	特殊寄存器	word	10 进制	SD0-SD16383	✓	12288 点
Z	变址寄存器	word	10 进制	Z0-Z11	✗	12 点
LZ	超长变址寄存器	dword	10 进制	LZ0-LZ1	✗	2 点

## 5.3 通信协议

DIP 协议仅用于第三方设备与 R 系列 PLC 的通信。R 系列 PLC 内部预先安装了 DIP 协议的服务器，用户可以通过 HC-works2 配置相关参数在 R 系列 PLC 上实现 DIP 协议通信。第三方设备需要内置自行开发的 DIP 协议客户端，第三方设备中的用户应用通过 DIP 协议客户端与 R 系列 PLC 进行通信。

### 5.3.1 通信方式

DIP 协议采用客户端/服务器的通信方式，客户端应用需要指定服务器站号，与服务器进行通信。为保持数据帧规格的统一性，即使在 1: 1 通信时，客户端也需要指定通信对手的主站站号。



图表 4 客户端/服务器的通信方式

### 5.3.2 典型拓扑连接

DIP 协议支持多种连接方式。用户可以根据需要选择使用 RS-232/422 串口的 1: 1 的连接方式，使用 RS-485 串口的 1: N 的连接方式，以及使用以太网的 N: N 的连接方式。

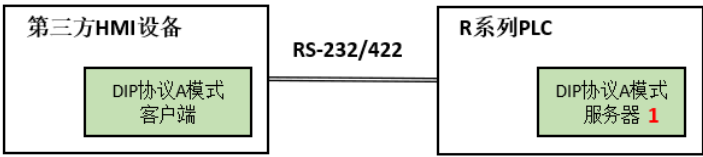
在使用 1: N 的多服务器的网络中，请为服务器指定不同的站号。

使用 DIP 协议不同模式的设备，无法进行互联以及通信。也就是说，仅仅实装了 S 模式的 HMI 设备，只能通过 RS-485 接口访问 R 系列 PLC，无法通过 RS-232 接口访问 R 系列 PLC。

以下为三种常见的连接方式。图中的设备 HMI 设备也可替换成其他不同的、实装了 DIP 协议客户端的其他设备。

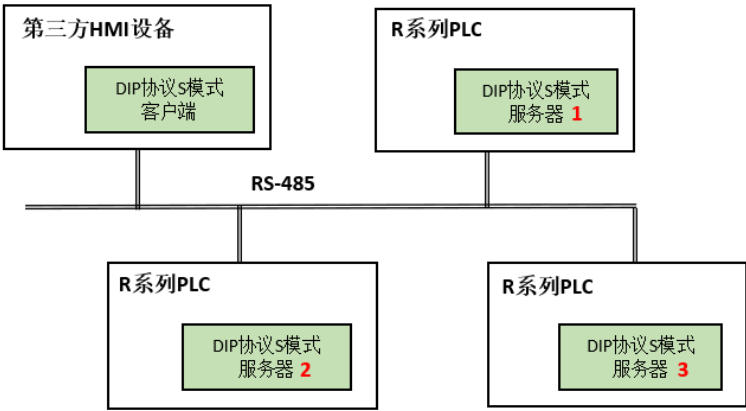


1) 使用 RS-232 网路, HMI 等设备可以单独与 R 系列 PLC 连接, 通过 DIP 协议访问软元件。



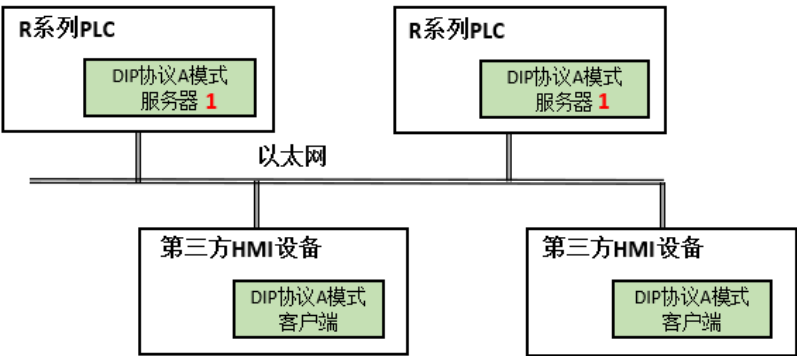
图表 5 串口的 1：1 连接

2) 通过组成 RS-485 网络, 一台 HMI 等客户端设备可以同时访问多台 R 系列 PLC。



图表 6 串口的 1：N 连接

3) 在以太网组成的网路中, 多台 HMI 等客户端设备, 使用 DIP 协议可以同时访问多台 R 系列 PLC 软元件。注意, R 系列 PLC 允许的同时连接数最大不超过 8 台。



图表 7 以太网的 N：N 连接

### 5.3.3 数据帧

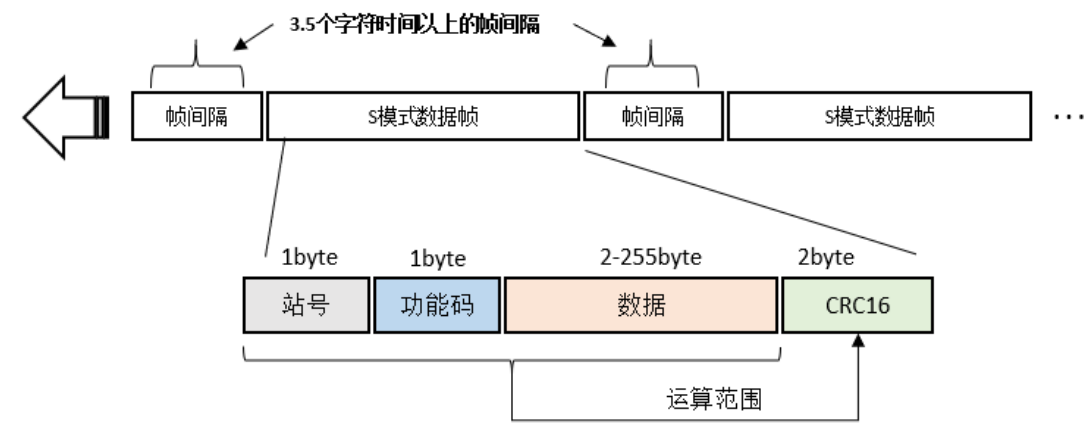
DIP 协议的数据帧包括①站号、②功能码、③数据这三项基本内容和校验信息。S 模式的数据帧应用于 RS-485 接口，A 模式的数据帧应用于 RS-232/422 和以太网接口。A 模式仅比 S 模式的数据帧多了 2byte 的帧识别标识符和 2byte 的帧数据长，后续说明中除非特别注明，数据帧均指 S 模式的数据帧。

DIP 协议的站号、功能码、数据这三项基本内容和校验方式，为 S 模式和 A 模式共通。DIP 协议的应答数据帧格式与请求数据帧格式相同。

#### 5.3.3.1 S 模式

S 模式的数据帧通过帧间隔进行识别，客户端发送的请求报文，间隔时间为 3.5 个字符时间以上。

数据帧包含的 CRC16（计算逻辑参见 3.3.5.9 节）验证码，用来校验数据帧的有效性。校验码运算范围为站号、功能码和数据。接收到 CRC16 校验码错误的数据帧，PLC 作为错误数据丢弃掉，不会向客户端返送错误报文。

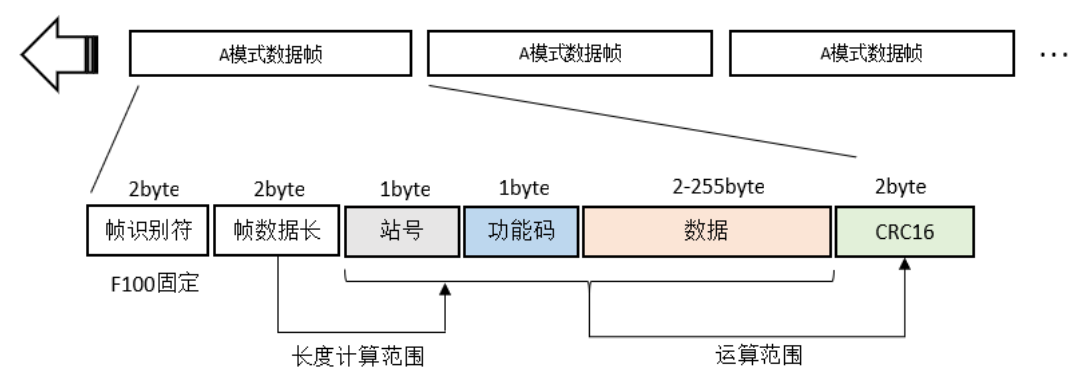


图表 8 S 模式的数据帧

5.3.3.2 A 模式

A 模式的帧识别符固定为 2 个字节的 F1（低位字节）00（高位字节）。如图所示，帧数据长为 2 个字节（先低位后高位），长度计算对象包含站号、功能码、数据这三个基本项。

数据帧包含的 CRC16（计算逻辑参见 3.3.5.9 节）验证码，用来校验数据帧的有效性。校验码运算范围为站号、功能码和数据。接收到帧识别符错误或者长度错误的帧数据，PLC 作为错误数据丢弃掉，不会向客户端返送错误报文。

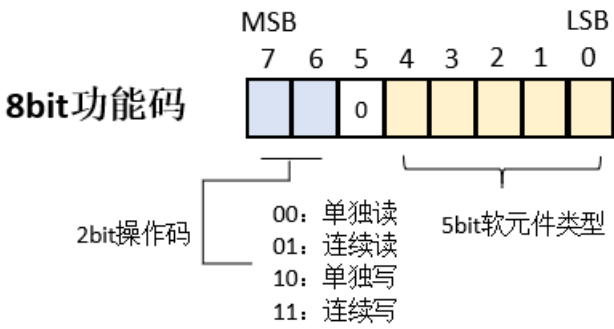


图表 9 A 模式的数据帧

5.3.3.3 功能码格式

DIP 协议 1 个字节的 功能码，包含 bit6-bit7 的 2bit 操作码和 bit0-bit4 的 5bit 软元件代码。功能码的 bit5 固定为 0。

如图 10 所示，操作码分别表示单独读取、连续读取、单独写入、连续写入这四种操作。软元件类型为 R 系列 PLC 支持的所有可访问软元件，如表 4 所示。



图表 10 功能码格式

DIP 协议根据软元件属性（bit 软元件或 word 软元件）对软元件进行相应的操作。

T/ST/C/LC 软元件的 bit/word 双属性，对应不同软元件编码。表 4 为使用 DIP 协议访问 R 系列 PLC 所有软元件的功能码一览。

表格 4 功能码一览

R 系列 PLC 软元件			8bit 功能码			
符号	名称	类型码 5bit	单独读	连续读	单独写	连续写
位软元件及对应的功能码						
X	输入	00000	00H	40H	80H	C0H
Y	输出	00001	01H	41H	81H	C1H
M	内部继电器	00010	02H	42H	82H	C2H
F	报警器	00011	03H	43H	83H	C3H
S	步进继电器	00100	04H	44H	84H	C4H
L	锁存继电器	00101	05H	45H	85H	C5H
SM	特殊继电器	00110	06H	46H	86H	C6H
TC	定时器（线圈）	00111	07H	47H	87H	C7H
TS	定时器（触点）	01000	08H	48H	88H	C8H
STC	累计定时器（线圈）	01001	09H	49H	89H	C9H
STS	累计定时器（触点）	01010	0AH	4AH	8AH	CAH
CC	计数器（线圈）	01011	0BH	4BH	8BH	CBH
CS	计数器（触点）	01100	0CH	4CH	8CH	CCH
LCC	长计数器（线圈）	01101	0DH	4DH	8DH	CDH
LCS	长计数器（触点）	01110	0EH	4EH	8EH	CEH
字软元件及对应的功能码						
TV	定时器（当前值）	10000	10H	50H	90H	D0H
STV	累计定时器（当前值）	10001	11H	51H	91H	D1H
CV	计数器（当前值）	10010	12H	52H	92H	D2H
D	数据寄存器	10011	13H	53H	93H	D3H
R	文件寄存器	10100	14H	54H	94H	D4H
SD	特殊寄存器	10101	15H	55H	95H	D5H
双字软元件及对应的功能码						
LCV	长计数器（当前值）	11100	1CH	5CH	9CH	DCH

5.3.3.4 数据段格式

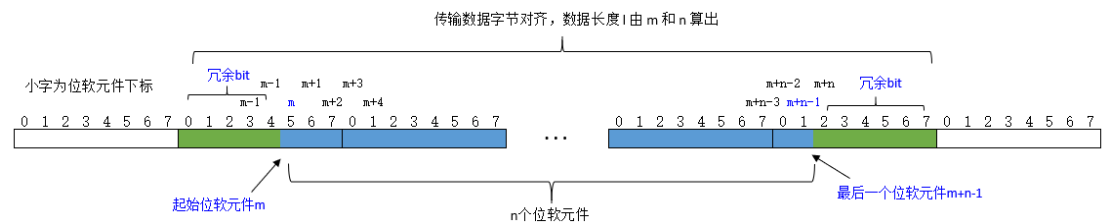
DIP 协议报文中，不同操作码后续的数据段长度也不相同。其中的多软元件读响应报文和多软元件写请求报文，由于要编码软元件的数据值，因此数据段的长度为可变长。

在 DIP 协议的数据段中，字软元件的数据值按照由低到高的顺序进行编码，字和双字的编码也是先低位字节后高位字节。

位软元件数据的编码，客户端和服务端均采用**字节整列的对位排列方式**。字节整列的数据即从包含首个位软元件的字节开始发送，一直到包含最后一个目标位软元件的字节为止。同样，在数据接收端也按照同样的方式进行数据处理。注意，数据编码时，对范围外的首位数据下标值，请填写 0。

5.3.3.4.1 位软元件数据的对位排列

如下图所示，DIP 协议在生成连续位软元件数据值的时候，按照字节单位整理，不进行数值的位移调整。因此在传输字节的首尾，可能产生一些冗余的 bit 传输。这些冗余 bit 需被置位为 0。同样，在收到连续位软元件数值后，接收端也需要根据软元件起始位（图中的 m）和数据长（图中的 n），就算出字节数值中位软元件的偏移量进行合理取值。



图表 11 位软元件值的传输数据对齐

5.3.3.4.2 位软元件数据的长度计算

起始下标为 m 的字软元件，连续 n 个 bit 的读写操作时，送信侧报文的送信字节数 l 的计算公式，以及受信侧报文解析时首个软元件值的下标计算公式如下：

1) 数据字节长度 l 计算公式

$$l = (m \% 8 \neq 1 : 0) + (n - m \% 8) / 8 + (n - m \% 8) \% 8 \neq 1 : 0$$

## 2) 首个软元件下标 i 的计算公式

$$i = m \% 8$$

<举例说明>

读取 R 系列 PLC 中内部继电器 M100 至 M299，一共 200 个位软元件的值时，R 系列 PLC 侧发送的应答报文中，数据字节长度计算如下：

由于首个字软元件下标  $m = 100$ ，字软元件数量  $n = 200$ ，因此

1) 应答报文的数据字节长度  $l =$

$$(100 \% 8 ? 1 : 0) + (200 - 100 \% 8) / 8 + (200 - 100 \% 8) ? 1 : 0 = 26$$

2) 解析应答报文时，首个软元件下标  $i = 100 \% 8 = 4$

需要注意的是，上述公式仅对 10 进制下标有效。8 进制下标的软元件计算，在编码实现时请自行换算。

### 5.3.3.5 报文格式

DIP 协议的报文格式由①读写操作类型（读或者写、连续或者单独）、②软元件类型（位或者字）的排列组合决定，一共有八种格式的请求报文和应答报文。另外，还有一种异常应答报文，一共支持九种报文格式。标记站号

表格 5 报文操作码和数据长

操作功能		操作码	请求报文数据长 (单位：字节)	应答报文数据长 (单位：字节)
位 软 元 件	单个位读出	00	2	1
	单个位写入	10	4	4
	多个位读出	01	4	报文可变长
	多个位写入	11	报文可变长	4
字 软 元 件	单个字读出	00	2	2
	单个字写入	10	4	4
	多个字读出	01	4	报文可变长
	多个字写入	11	报文可变长	3
双 字 软 元 件	单个字读出	00	2	4
	单个字写入	10	6	6
	多个字读出	01	4	报文可变长
	多个字写入	11	报文可变长	3

注 1：操作码是功能码的 bit7 和 bit6。

注 2：没有标注报文可变长的报文，均为固定长报文。

请求报文中的站号为通信对手（服务器端）的站号，功能码代表需要通信对手执行的功能。应答报文中的站号和功能码与收到的请求报文相同。

服务器端收到的请求报文如果是校验失败，则不返回错误报文。如果是报文校验成功但功能码编码错误，或者是软元件地址范围错误，则返回失败报文。

**注意：**应答报文的最大长度和请求报文一样。因此需要在请求报文中限制需要操作的软元件个数。如果单次操作的最大软元件个数不能满足要求，可以分为多次请求。



5.3.3.5.1 单个位软元件的读

简要说明和举例。

单位读请求

站号	
功能码	
软元件地址	高位
	低位
CRC16	低位
	高位

单位读应答

站号	
功能码	
数据	
CRC16	低位
	高位

图表 12 read bit 报文

5.3.3.5.2 单个位软元件的写

简要说明和举例。

单位写请求

站号	
功能码	
软元件地址	高位
	低位
变更数据	高位
	低位
CRC16	低位
	高位

单位写应答

站号	
功能码	
软元件地址	高位
	低位
变更数据	高位
	低位
CRC16	低位
	高位

置ON: FF(高) 00(低)  
置OFF: 00(高) 00(低)

图表 13 write bit 报文

5.3.3.5.3 多个位软元件的读

简要说明和举例。

一个数据帧可读取的位软元件个数最大值是 $(255-1) \times 8 = 2032$  个。

多位读请求

站号	
功能码	
软元件开始地址	高位
	低位
读取软元件数	高位
	低位
CRC16	低位
	高位

多位读应答

站号	
功能码	
数据字节数	
数据首个字节	
• • •	
CRC16	低位
	高位

图表 14 read bits 报文

5.3.3.5.4 多个位软元件的写

简要说明和举例。  
一个数据帧可写入的位软元件个数最大值是 $(255-5) \times 8 = 2000$  个。

多位写请求

站号	
功能码	
软元件开始地址	高位
	低位
写入软元件数	高位
	低位
数据字节数	
数据首个字节	
...	
CRC16	低位
	高位

多位写应答

站号	
功能码	
软元件开始地址	高位
	低位
写入软元件数	高位
	低位
CRC16	低位
	高位

图表 15 write bits 报文

5.3.3.5.5 单个字软元件的读

简要说明和举例。

单字读请求

站号	
功能码	
软元件地址	高位
	低位
CRC16	低位
	高位

单字读应答

站号	
功能码	
软元件值	高位
	低位
CRC16	低位
	高位

图表 16 read word 报文

5.3.3.5.6 单个字软元件的写

简要说明和举例。

单字写请求

站号	
功能码	
软元件地址	高位
	低位
软元件值	高位
	低位
CRC16	低位
	高位

单字写应答

站号	
功能码	
软元件地址	高位
	低位
软元件值	高位
	低位
CRC16	低位
	高位

图表 17 write word 报文

5.3.3.5.7 多个字软元件的读

简要说明和举例。  
一个数据帧可读取的字软元件个数最大值是 $(255-1)/2=127$  个。

多字读请求

站号	
功能码	
软元件开始地址	高位
	低位
读取软元件数	高位
	低位
CRC16	低位
	高位

多字读应答

站号	
功能码	
数据字节数	
第1个字的上位字节	
第1个字的下位字节	
第2个字的上位字节	
第2个字的下位字节	
• • •	
CRC16	低位
	高位

图表 18 read words 报文

5.3.3.5.8 多个字软元件的写

简要说明和举例。  
一个数据帧可写入的字软元件个数最大值是 $(255-5)/2=125$  个。

多字写请求

站号	
功能码	
软元件开始地址	高位
	低位
写入软元件数	高位
	低位
数据字节数	
第1个字的上位字节	
第1个字的下位字节	
第2个字的上位字节	
第2个字的下位字节	
• • •	
CRC16	低位
	高位

多字写应答

站号	
功能码	
软元件开始地址	高位
	低位
写入软元件数	高位
	低位
CRC16	低位
	高位

图表 19 write words 报文

5.3.3.5.9 单个双字软元件的读

简要说明和举例。

单双字读请求

站号	
功能码	
软元件地址	高位
	低位
CRC16	低位
	高位

单双字读应答

站号	
功能码	
软元件值	最高位
	第2位
	第3位
	最低位
CRC16	低位
	高位

图表 20 read dword 报文

5.3.3.5.10 单个双字软元件的写

简要说明和举例。

单双字写请求

站号	
功能码	
软元件地址	高位
	低位
软元件值	最高位
	第2位
	第3位
	最低位
CRC16	低位
	高位

单双字写应答

站号	
功能码	
软元件地址	高位
	低位
软元件值	最高位
	第2位
	第3位
	最低位
CRC16	低位
	高位

图表 21 write dword

5.3.3.5.11 多个双字软元件的读

简要说明和举例。  
一个数据帧可读取的双字软元件个数最大值是 $(255-1)/4=63$  个。

多双字读请求

站号	
功能码	
软元件开始地址	高位
	低位
读取软元件数	高位
	低位
CRC16	低位
	高位

多双字读应答

站号	
功能码	
数据字节数	
第1个双字的最高位字节	
第1个双字的第2位字节	
第1个双字的第3位字节	
第1个双字的最低位字节	
第2个双字的最高位字节	
• • •	
CRC16	低位
	高位

图表 22 read dwords

5.3.3.5.12 多个双字软元件的写

简要说明和举例。  
一个数据帧可读取的双字软元件个数最大值是 $(255-5)/4=62$  个。

多双字写请求

站号	
功能码	
软元件开始地址	高位
	低位
写入软元件数	高位
	低位
数据字节数	
第1个双字的最高位字节	
第1个双字的第2位字节	
第1个双字的第3位字节	
第1个双字的最低位字节	
第2个双字的最高位字节	
• • •	
CRC16	低位
	高位

多双字写应答

站号	
功能码	
软元件开始地址	高位
	低位
写入软元件数	高位
	低位
CRC16	低位
	高位

图表 23 write dwords

5.3.3.5.13 错误应答

简要说明和举例。

错误应答

站号	
功能码	
错误码	
CRC16	低位
	高位

图表 24 错误应答报文

表格 6 错误码

No	错误码	名称	内容
1	01	功能码错误	请求报文中的功能码无法识别
2	02	软元件错误	请求报文中软元件无法识别
3	03	软元件地址错误	请求报文中指定的软元件地址越界
4	04	数据错误	请求报文中指定的数据有错误

### 5.3.3.6 CRC16 校验

DIP 协议 RTU 串行通信 (RTU 模式) 的出错检查通过 CRC16 进行, 计算逻辑与 modbusRTU 的 CRC 校验计算逻辑相同。计算步骤如下所示:

1. 加载 FFFFH (16 位全部为“1”) 的寄存器。将此寄存器作为 CRC 寄存器。
2. 计算 CRC 寄存器低位字节和报文前 8 位的逻辑异或, 并将结果放入 CRC 寄存器。
3. 将 CRC 寄存器向右方 (最低位的方向) 移动 1 位, 将最高位设为 0。
4. 确认进位标志。
5. 进位标志为 0 时: 重复上述步骤 3。(重新移位。)
6. 进位标志为 1 时: 计算生成多项式 A001H (1010 0000 0000 0001) 和 CRC 寄存器的逻辑异或。
7. 重复上述步骤 3 和 4 的操作, 直至位移动达到 8 次。通过该操作, 8 位都将被处理。
8. 对报文的下一个 8 位重复上述步骤 2~5 的操作。继续该操作, 直至全部字节都被处理。
9. CRC 寄存器最后的值为 CRC 值。
10. 将 CRC 值存储到报文中时, 顺序为低位 8 位→高位 8 位。