

# 温州市西门子中国（授权）一级代理商-西门子选型-西门子技术支持-西门子维修服务

产品名称	温州市西门子中国（授权）一级代理商-西门子选型-西门子技术支持-西门子维修服务
公司名称	湖南西控自动化设备有限公司
价格	.00/件
规格参数	
公司地址	中国（湖南）自由贸易试验区长沙片区开元东路1306号开阳智能制造产业园（一期）4#栋301
联系电话	15344432716 15386422716

## 产品详情

### 编写轮询程序

所谓轮询就是依次询问，假设我们有3个设备作为modbus从站，从站地址（站号）依次为1,2,3，使用case语句依次对这3个设备进行读写操作，而读出和写入的数据分别存入3套不同的变量当中。

### 建立设备变量

使用一个结构体来描述一个设备的所有信息，包括5个状态字（states:Array[0..4]

ofWord) 和5个控制字 (ctrl:Array[0..4]

of

Word) , 将结构体声明为数量为3的数组, 存放3个设备的数据。

---

在整个循环开始前, 设定起始设备地址, 然后按照“读操作触发, 读数据, 读设备地址+1, 延时, 写数据, 写操作触发, 写设备地址+1, 延时”的顺序持续循环, 按照设备地址号选择上面的结构体

变量:

读操作

---

iStep=0时, 关闭读写触发, 设定读写设备地址为1;

---

iStep=10时, 读操作触发, 模块发出读数据命令, 模块置位busy信号;

---

iStep=11时, 等待读操作完成, 模块读到设备数据后会置位done信号, 复位busy信号, 根据信号状态将读到的数据 (Read\_Data) 写入设备数据结构体 (DeviceData.states), 如果设备地址=1, 则写入 DeviceData[1].states, 设备地址变化, 写入的结构体也会相应的变化, 保证不同设备的数据不会互相干涉。这里加一个判断, 一段时间读不到数据返回10步骤重新进行读操作。

---

iStep=12时，用计数的方式做一个简单的延时功能，避免因读写频率太快导致设备反应不过来。

## 写操作

向设备写入信息，在写入操作触发前要先将相应设备结构体中的数据（DeviceData.ctrl）写入发送数据缓存区（Sent\_Data），然后再进行写操作，与读操作类似，写入完成后设备地址+1，跳转下一步骤。

## 运行程序

将程序编译写入后重启PLC，可以看到通讯板的指示灯已经开始闪烁，而轮询步骤iStep始终在10,11两步，证明读数据命令已经发出，但是没有接受到设备的反馈，始终在进行第一个设备的读操作。

虽然能够看到通讯灯在闪，但我们仍然不能直观的看到这个网络中的状态，这就需要前面提到的S  
PU软件，监视串口网络中的报文。

## 监视报文

设定端口号，选择Hex数据格式，点击开始，可以看到当前网络中所有报文，根据modbus协议的规格（可以自行百度），可以判断这些报文就是plc发出的读命令。

---

想要读到数据必须要有从站，我们使用modsim软件模拟出3个从站：

modsim

将3个模拟从站地址分别设为1,2,3，起始modbus地址与程序设为一致：40100，长度设为20，这样读写地址都能看到。点击connection设定通讯参数（波特率，数据位，停止位，校验位与程序中设为一致）。点击确定后能看到通讯板和转换器的接受发送指示灯开始闪烁，程序中的设备地址也在1-3中循环变化：

通讯指示灯

---

由于动图的帧率选的较低，会漏掉几个灯的状态。。。

变化的设备地址

---

监看程序中设备地址，能够看到地址在1-3之间循环变化。可惜的是modsim与SPU不能共用一个串口，看不到modsim反馈的报文了。

---

接下来我们在modsim中改变几个地址的值，看看PLC的设备数据结构体中能否进行相应的变化，将

设备1的数据设定为：

---

40100设置为110，

---

40101设置为111，

---

40102设置为112，

---

40103设置为113，

---

40104设置为114，

设定数据

---

数据设定后在PLC的DB块中监视DeviceData的值：

读取数据

---

可以看到DeviceData[1].states的值已经变化（16进制），而DeviceData[2]和DeviceData[3]并没有变化。

---

下面进行写数据的验证，在程序中将DeviceData[2].ctrl任意赋值，然后再modsim中查看：

写入数据赋值

写入成功

---

可以看到modsim 3中相应地址的数据也已经变化，而其他模拟设备中并没有改变。