

## 江西西门子PLC代理商

产品名称	江西西门子PLC代理商
公司名称	浔之漫智控技术-西门子PLC代理商
价格	666.00/件
规格参数	品牌:西门子 产品规格:模块式 产地:德国
公司地址	上海市松江区石湖荡镇塔汇路755弄29号1幢一层A区213室
联系电话	15221406036

## 产品详情

江西西门子PLC代理商

### 5 OS编译和组态

5.1 在SIMATIC Manager中选择Options – ‘ Compile Multiple OSs ’ Wizard – Start，进行OS编译和传送。如图13

图13

5.2 在Select network connection中为SIMATIC 300 Station的CPU选择TCP/IP作为WinCC unit。如图14

图14

5.3在Select network connection中为PC Station的WinLC RTX选择Named Connection作为WinCC unit。如图15

图15

5.4 执行编译和传送，默认选择Scope中的Entire OS，再次编译可以选择Changes。如图16

图16

5.5 在SIMATIC Manager中，打开SIMATIC PC Station的WinCC Application中的OS。如图17

图17

5.6 在WinCC Explorer中，可以看到SIMATIC S7 PROTOCOL SUITE已经自动添加到Tag Management中，SIMATIC 300 Station的CPU和SIMATIC PC Station的WinLC RTX中DB Block的部分数据也已经传送到TCP/IP通道和Named Connections通道中。如图18和图19

图18

图19

西门子S7-200CPU的通信口可以设置为自由口模式。选择自由口模式后，用户程序就可以完全控制通信端口的操作，通信协议也完全受用户程序控制。

S7-200CPU上的通信口在电气

上是标准的RS-485半双工串行通信口。此串行字符通信的格式可以包括：

一个起始位

7或8位字符（数据字节）

一个奇/偶校验位，或者没有校验位

一个停止位

自由口通信速波特率可以设置为1200、2400、4800、9600、19200、38400、57600或112500。

凡是符合这些格式的串行通信设备，理论上都可以和S7-200CPU通信。

自由口模式可以灵活应用。Micro/WIN的两个指令库（USS和ModbusRTU）就是使用自由口模式编程实现的。

在进行自由口通信程序调试时，可以使用PC/PPI电缆（设置到自由口通信模式）连接PC和CPU，在PC上运行串口调试软件（或者Windows的HyperTerminal - 超级终端）调试自由口程序。

USB/PPI电缆和CP卡不支持自由口调试。

### 自由口通信要点

应用自由口通信首先要把通信口定义为自由口模式，同时设置相应的通信波特率和上述通信格式。用户程序通过特殊存储器SMB30（对端口0）、SMB130（对端口1）控制通信口的工作模式。详见下图

CPU通信口工作在自由口模式时，通信口就不支持其他通信协议（比如PPI），此通信口不能再与编程软件Micro/WIN通信。CPU停止时，自由口不能工作，Micro/WIN就可以与CPU通信。

通信口的工作模式，是可以在运行过程中由用户程序重复定义的。

如果调试时需要在自由口模式与PPI模式之间切换，可以使用SM0.7的状态决定通信口的模式；而SM0.7的状态反映的是CPU运行状态开关的位置（在RUN时SM0.7="1"，在STOP时SM0.7="0"）

自由口通信的核心指令是发送（XMT）和接收（RCV）指令。在自由口通信常用的中断有“接收指令结束中断”、“发送指令结束中断”，以及通信端口缓冲区接收中断。

与网络读写指令（NetR/NetW）类似，用户程序不能直接控制通信芯片而必须通过操作系统。用户程序使用通信数据缓冲区和特殊存储器与操作系统交换相关的信息。

XMT和RCV指令的数据缓冲区类似，起始字节为需要发送的或接收的字符个数，随后是数据字节本身。如果接收的消息中包括了起始或结束字符，则它们也算数据字节。

调用XMT和RCV指令时只需要指定通信口和数据缓冲区的起始字节地址。

XMT和RCV指令与NetW/NetR指令不同的是，它们与网络上通信对象的“地址”无

关，而仅对本地的通信端口操作。如果网络上有多个设备，消息中必然包含地址信息；这些包含地址信息的消息才是XMT和RCV指令的处理对象。

由于S7-200的通信端口是半双工RS-485芯片，XMT指令和RCV指令不能同时有效。

## XMT和RCV指令

XMT（发送）指令的使用比较简单。RCV（接收）指令所需要的控制稍多一些。

RCV指令的基本工作过程为：

在逻辑条件满足时，启动（一次）RCV指令，进入接收等待状态

监视通信端口，等待设置的消息起始条件满足，然后进入消息接收状态

如果满足了设置的消息结束条件，则结束消息，然后退出接收状态

所以，RCV指令启动后并不一定就接收消息，如果没有让它开始消息接收的条件，就一直处于等待接收的状态；如果消息始终没有开始或者结束，通信口就一直处于接收状态。这时如果尝试执行XMT指令，就不会发送任何消息。

所以确保不同时执行XMT和RCV非常重要，可以使用发送完成中断和接收完成中断功能，在中断程序中启动另一个指令。

在《S7-200系统手册》和Micro/WIN在线帮助中关于XMT和RCV指令的使用有一个例子。这个例子非常经典，强烈建议学习自由口通信时先做通这个例子。例程，见下图

## 字符接收中断

S7-200CPU提供了通信口字符接收中断功能，通信口接收到字符时会产生一个中断，接收到的字符暂存在特殊存储器SMB2中。通信口Port0和Port1共用SMB2，但两个口的字符接收中断号不同。

每接收到一个字符，就会产生一次中断。对于连续发送消息，需要在中断服务程序中将单个的字符排列到用户规定的消息保存区域中。实现这个功能可能使用间接寻址比较好。

对于高通信速率来说，字符中断接受方式需要中断程序的执行速度足够快。

一般情况下，使用结束字符作为RCV指令的结束条件比较可靠。如果通信对象的消息帧中以一个不定的字符（字节）结束（如校验码等），就应当规定消息或字符超时作为

结束RCV指令的条件。但是往往通信对象未必具有严格的协议规定、工作也未必可靠，这就可能造成RCV指令不能正常结束。这种情况下可以使用字符接收中断功能。如下图