

自贡西门子PLC总代理商

产品名称	自贡西门子PLC总代理商
公司名称	浔之漫智控技术-西门子PLC代理商
价格	.00/件
规格参数	
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

自贡西门子PLC总代理商

一、LOGO可编程序控制器简介 LOGO可编程序控制器十分精巧（如图所示），其编程简单，具有丰富的逻辑控制功能，抗干扰能力强，可靠性高，特别适用于恶劣的工作环境。LOGO虽然属微型可编程序控制器，西门子新推出的LOGO则有下列功能特性： LOGO的功能：基本逻辑控制功能有“与”、“或”、“非”、“与非”、“或非”、“异或”等；特殊控制功能有通/断延时、RS触发器、脉冲继电器、时钟、加减计数器、随机发生器及模拟量PI控制功能等。 LOGO编程：带操作按键和文本显示的LOGO可以通过使用控制器的操作按键和LCD面板或个人电脑进行程序的输入、编制和显示及监视运行状态，这一特性优于同点数的PLC。不带有液晶显示器和按键LOGO则可以通过个人电脑上应用软件进行编程。LOGO还可通过个人电脑测试、模拟和打印控制程序等。 LOGO硬件结构：LOGO的电源有24VDC/24VAC、24VDC/12VDC、24VDC及115V~240VAC，不同电源的LOGO有不同的电压等级的开关量输入、高速脉冲输入及0-10V等模拟量输入，输出有可达10A的继电器输出和晶体管输出，可以满足较多的控制需求，适用于较多的场合。LOGO控制器的CPU主机自带有8DI/4DO且可扩展，LOGO扩展模块有4DI/4DO、8DI/8DO及2AI/2AO等模块。 LOGO程序加密及复制功能：利用LOGO编程软件可实现这二个功能，黄色模块可复制应用程序，红色模块可保护程序。另外通过操作按键和LCD面板也可设置密码。

二、PS板打孔机 PS板打孔机是印刷行业所用到的一种自动化机械，其用途是对PS板进行打孔及折弯成型方面的加工。PS板打孔机的工序变化是比较多的，用户对PS板打孔机提出的要求也多，很难进行标准化生产。传统的PS板打孔机工序控制是使用继电器来完成，工序复杂些的PS板打孔机的电控板上能用几十只继电器组成逻辑控制电路，密密麻麻的一片，设计接线调试很费工时，也跟不上时代的要求。近年来，有些PS板打孔机开始使用PLC控制，但也出现了PLC对生产过程中工序时间的变化适应困难，不能总拿着编程电脑去现场改，故一般都采取尽量延长工序时间的办法，但这样又会影响生产的速度，如加入人机介面彻底解决则又会使得PS板打孔机的成本上升很多。对于这些问题，我们与今年二月份起开始在PS板打孔机试用西门子的LOGO可编程序控制器，利用LOGO上的操作按键及LCD显示器解决上述问题。通过对PS打孔机的功能、工艺过程、结构等特点进行分析论证后，决定采取

编制通用控制程序并选择使用的设计方案。通过对LOGO程序的仔细推敲，基本上做到了对于不同机型无需变更程序，只需根据不同的工艺要求选用即可。见下示意图：

三、PS板打孔机 LOGO控制系统的设计描述 控制系统是以LOGO可编程控制器为控制核心，再配24VDC电源、电源控制开关、断路器、接触器等，该控制系统LOGO可编程控制器DI、DO配置及其功能如下：“I1”工序一：吸风，打孔。打第几那组孔由有“I3”、“I4”位决定。“I2”工序二：定位、压板、折板、停折板、松压板、顶板、停顶板、放定位。打孔选定必须在0位即“I3”或“I4”位都不选。“I3”打孔选定：组一。“I4”打孔选定：组二。“I1”+“I2”均不选时为自动工序0：例：“I7”+“I8”均不选时为00功能，其工艺过程为为吸风，打孔（打第几那组孔由有I3、I4决定）、停吸风、压板、折板、停折板、松压板、顶板、停顶板。打孔选定必须在“I3”或“I4”位。“I5”启停真空泵。按一次启动真空泵吸风，再按一次停真空泵。“I6”工序及打孔启停。“I7”功能组合选择一。01功能。“I8”功能组合选择二。02功能。“I9”停止打孔磁开关：孔组1—1。“I10”停止打孔磁开关：孔组1—2。“I11”停止打孔磁开关：孔组2—1。“I12”停止打孔磁开关：孔组2—2。“Q1”定位驱动：驱动定位电磁阀。“Q2”压板驱动：驱动压板电磁阀。“Q3”折板驱动：驱动折板电磁阀。“Q4”顶板驱动：驱动顶板电磁阀。“Q5”组1-1打孔驱动：驱动1-1组打孔电动机。“Q6”组1-2打孔驱动：驱动1-2组打孔电动机。“Q7”组2-1打孔驱动：驱动2-1组打孔电动机。“Q8”组2-2打孔驱动：驱动2-组打孔电动机。“Q9”吸风真空泵驱动：驱动真空泵接触器。由于PS打孔机经常出口到韩国等地，供货距离远，现场维修不便，如出现控制问题可能会将整机送回来维修，所以如采用LOGO程控器控制，就必须做到克服任何程序上的死点（即就是乱操作也不会出问题）。

四、在PS板打孔机上使用LOGO可编程序控制器的优点 与PLC相同的是LOGO也可完全代替继电器的逻辑功能，从而简化了PS板打孔机的电气控制电路，简化了PS打孔机的制造、调试和维护等工作。使用LOGO实现了多功能并供选择使用，总体性能比原先的继电器控制更加完善，适应工艺变化的范围广，在制造和使用过程中，修改工序参数和程序方便快捷，有利于实现标准化设计。又与PLC相同的是LOGO也具有与上位计算机通信的功能，可为控制系统自动化程度的进一步提高——实现远程监控奠定了基础。由于LOGO控制的接线简单，电气控制板占地少也使得PS板打孔机的整体结构变得更加精简合理；方便使用时的工序调整和PS板品种的变更。

五、应用比较 LOGO与PLC比较：PLC虽然功能更强些但价格较贵；PLC应用得广泛与其出现的早有关，就LOGO很轻易地就能完成的普通的逻辑控制而言，有了LOGO就感到PLC是在“牛刀宰鸡”，其大多数功能都用不上，说起来也是种资源浪费——特别是对小型自动化系统和设备。编程对PLC来说都要配编程器或个人电脑，而且对软件版本还有要求，LOGO则可不用。另外，LOGO可以通过操作按键和LCD显示器随时根据需要修改程序及工序参数这一功能也是多数PLC所不及的。LOGO与的继电器控制方式比较：一般的，LOGO会比继电器控制方式的投资稍大，但对于较复杂的逻辑控制，LOGO控制也会比继电器控制的投资低，况且LOGO功能更强大，性能价格比要超过继电器控制很多。LOGO的电气接线要比继电器控制接线简单的多，调试简单，降低了制作人工成本。另外LOGO软件的设计成本要比继电器控制电路的设计成本要低且便于修改，开发周期也短。

六、结束语 通过上述比较及应用案例，我们可以知道：LOGO可编程控制器是小型系统及设备实现自动化程序控制的一种简单而快捷的方式，它不仅易满足各种不同工艺需求，且稳定可靠，易做到标准化高效生产。后，我们可以得出结论：LOGO在小型的以少数逻辑控制为主的系统或设备上使用比PLC更合适，通过不断的推广应用必将为更多的企业带来实惠。

细纱机是将粗纱或条子纺成一定支数细纱的纺织机器。它的控制系统性能稳定与否直接影响到生产成本。用PLC控制细纱机，操作简单、接线少，成纱质量好，且维修方便、利于管理。细纱机有低速运行、高速运行、吹吸风、落纱等过程。落纱分为自动落纱和中途落纱（暂停工作），自动落纱又分定长落纱和定时落纱。自动落纱的方式、落纱时间及长度均可设置，并能掉电保持。控制系统中，有8个执行元件，17个输入元件，其中热继电器FT1、F12、FT3和限位开关S为安全保护器件，可串联作为一个输入。选用西门子S7-200 CPU224型PLC能满足控制要求。1 PLC外部接线和工

艺要求 PLC外部接线如图1所示。接通电源后，吸风电动机开始工作。同时，钢领板升降电动机正转，钢领板上升。当钢领板升到了始纺位置时，其复位开关动作，电动机停止。按下低速起动按钮，主机开始低速运行，进行细纱接头。按下高速起动按钮，转换为高速运转，全机进入正常纺纱阶段。

纺纱满管后，钢领板复位开关动作，满管信号灯亮。进入工作位置，主机停止开关接通，此时，主机高速接触器释放，钢领板升降中间继电器吸合，主机断电保持惯性回转。随后，钢领板升降电动机反转，钢领板开始下降，降到极限位置时，钢领板下降限位开关动作，停止下降。撑爪电磁铁吸合时，将撑爪打开，主轴制动电磁铁吸合，主轴制动刹车。经过一段延时后，切断控制电源，落纱完毕。需要中途停车时，按下中途停车按钮，主机即可停车，并自行制动。需要提前落纱时，按下中途落纱按钮即可。当机器发生意外时，按下紧急停车按钮，可使全机立即停车。在程序中设置了各个过程、设备之间的联锁保护，使生产过程更加安全、合理。

2 编程 PL

C的控制梯形图如图2所示。在上电时用PLC内部继电器SMO.1和复位指令，程序在运行前先复位。利用模拟电位器SMB28和PLC的内部数学运算功能，设计一个0~180 s的时间继电器T37。梯形图中还使用了变量存储器V0.1、V0.2及V0.5。因为钢领板电动机在工作中要实现正反转，所以不仅要在程序中实现互锁，而且要在电气连接时实现电气互锁。满纱时，满纱信号灯一直亮；当有低速接触器触点发生热熔粘连时，信号灯以1s周期闪烁

1.引言

CPU寄存器状态字的各位给出了有关指令状态或结果的信息以及所出现的错误，我们可以将二进制逻辑操作状态位信号状态直接集成到程序中，以控制程序执行的流程。

2.状态字寄存器

先简单介绍一下CPU中状态字。检查位：状态字的0位称作检查位，如果/FC位的信号状态为“0”，则表示伴随着下一条逻辑指令，程序中将开始一个新的逻辑串。FC前面的斜杠表示对FC取反。

逻辑运算结果：状态字的第1位为RLO位（RLO=“逻辑运算结果”），在二进制逻辑运算中用作暂时存储位。比如，一串逻辑指令中的某个指令检查触点的信号状态，并根据布尔逻辑运算规则将检查的结果（状态位）与RLO位进行逻辑门运算，然后逻辑运算结果又存在RLO位中。

状态位：状态位（第2位）用以保存被寻址位的值。状态位总是向扫描指令（A,AN,O,...）或写指令（=,S,R,）显示寻址位的状态（对于写指令，保存的寻址位状态是本条写指令执行后的该寻址位的状态）。

OR位：在用指令OR执行或逻辑操作之前，执行与逻辑操作的时候，就需要用到OR这一状态位。OR位表示先前执行的与逻辑操作产生的值为“1”，于是，逻辑操作或的执行结果就已被确定为“1”。

OV位：溢出表示算术或比较指令执行时出现了错误。根据所执行的算术或逻辑指令结果对该位进行设置。

OS位：溢出存储位是与OV位一起被置位的，而且在更新算术指令之后，它能够保持这种状态，也就是说，它的状态不会由于下一个算术指令的结果而改变。这样，即使是在程序的后面部分，也还有机会判断数字区域是否溢出或者指令是否含有无效实数。OS位只有通过如下这些命令进行复位：JOS（若OS=1，则跳转）命令，块调用和块结束命令。

CC1及CC0位：CC1和CC0（条件代码）位给出有关下列结果的相关信息：•算术指令结果•比较指令结果•字逻辑指令•在移位功能中，移出位相关信息。可以用以下指令来检查条件代码CC1和CC0。

CC1 CC0 检查完成后，如果：

00 A == 0 结果 = 0

10 A > 0 结果 > 0

01 A < 0 结果 < 0

BR位：状态字的第8位称为二进制结果位。它将字处理程序与位处理联系起来，在一段既有位操

作又有字操作的程序中，用于表示字逻辑是否正确。将BR位加入程序后，无论字操作结果如何，都不会造成二进制逻辑链中断。在梯形图的方块指令中，BR位与ENO位有对应关系，用于表明方块指令是否被正确执行：如果执行出现了错误，BR位为0，ENO位也为0；如果功能被正确执行，BR位为1，

ENO位也为1。在用户编写的FB/FC程序中，应该对BR位进行管理，功能块正确执行后，使BR位为1，否则使其为0。使用SAVE指令将RLO存入BR中，从而达到管理BR位目的。

状态字的9-15位未使用。

3.具体使用

下面我们结合STEP7中的指针编程来具体介绍条件码CC0/CC1的用法。

不同的指令在CPU中执行时间是不同的。浮点数比定点数执行时间要长；字逻辑指令比位逻辑指令执行时间要长；在某些程序中适当使用状态字来进行编程可以减少CPU程序的执行时间。

例1：比如说要比较一个DB中块的DBBO-DBB99这100个字节是正数是负数还是0，正数用1来表示；负数用-1来表示；0用0来表示。并且将对应结果存入MB200开始的100个字节中。我们通常的做法可能为：

如果利用条件码来进行编程，既可以减少程序的大小还会减少一定的指令执行时间，我们只需要将中间的比较程序加以优化，即可以达到目的。

例2：根据状态位C0和CC1的状态而跳转的跳转功能指令JZ不改变任何状态位的状态，而且逻辑操作结果RLO值也会“随着”该跳转功能带到跳转程序段中，供用户程序其它逻辑操作之用（不改变/FC状态）。示例 两个整数相减并需进行连续判断：
LMW2 LMW8 -I JZ ZERO //
如果结果等于“0”，则跳转至标号ZERO处 //
结果不等于“0”时所执行的指令 ZERO: // 结果等于“0”时，所要执行的指令 如果用户不熟悉JZ指令和状态位C0和CC1的具体含义，编程时就需要通过比较指令将比较结果存入一个二进制位中，再根据这个二进制位通过JC/JCN指令来控制程序的执行了。

例3：我们实际应用中可能要利用某些协议转换网关（比如说Hilscher公司的NTTAP系列网关）来和某些串口协议的仪表进行通信时，会遇到CRC校验的问题，关于CRC校验时需要判断溢出位是否为1的问题来进行程序的进一步计算。我们以EURO2408的MODBUS通信时需要的CRC校验为例说明CRC校验的步骤：

- 1、装载16#FFFF到一个16位CRC寄存器；
- 2、将CRC寄存器的高8位字节与信息中的个8位字节相异或，结果返回到CRC寄存器中；
- 3、将CRC寄存器数据向右移动一位；
- 4、如果溢出的位等于1，则将CRC寄存器与16#A001相异或，结果返回到CRC寄存器中；
- 4、如果溢出的位等于0，则重复第3步；
- 5、重复第3、4步骤，直到已经移位了8次；
- 6、将CRC寄存器的高8位字节与信息中的下一个8位字节相异或，结果返回到CRC寄存器中；

7、重复第3步到第6步，直到信息中所有字节都与CRC寄存器相异或，并都移位了8次；

8、后的CRC寄存器中的结果即为CRC校验码，后被添加到信息（数据）的末尾（交换！低8位在前，高8位在后；）

在第4步中需要判断溢出的位是否为1，如何判断对于整个程序有着重要的影响。我们可以用A>0指令来判断这个条件，具体代码的编写，有兴趣时大家可以根据上面的步骤编写一个自己的CRC程序。

4.结束语

在一般情况下，我们不必考虑这些状态位，但在某些情况下，利用这些状态位并结合一定的指令，可以给我们的编程带来更大的灵活性，同时对于进一步提高自己的编程水平也有一定的作用。