

邯郸西门子S7-200模块代理商|西门子CPU选型

产品名称	邯郸西门子S7-200模块代理商 西门子CPU选型
公司名称	浔之漫智控技术（上海）有限公司总部
价格	2200.00/件
规格参数	品牌:西门子 货期:现货 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	15021292620 15021292620

产品详情

邯郸西门子S7-200模块代理商|西门子CPU选型 邯郸西门子S7-200模块代理商|西门子CPU选型

邯郸西门子S7-200模块代理商|西门子CPU选型邯郸西门子S7-200模块代理商|西门子CPU选型邯郸西门子S7-200模块代理商|西门子CPU选型

部分是基本指令系统，也就是我今天要分享的内容。为什么说基本指令系统是PLC的单词，我们在学习英语的时候词汇量是必不可少的，不懂单词就看不懂英文，同样的，PLC的基本指令就像是PLC语句表程序中一个个的单词，看不懂单词就看不懂语句表。

而“可有可无”不是说指令系统没用，而是基于现在都是用梯形图编程，所以我们不需要对这些指令死记硬背，只需要懂得这些指令的功能和用法就好了，就好比学英语，其实你叫我默写某个单词我是写不出来的，但是给我看这个单词我却能知道它的意思。

说白了，其实就是我在学过这些指令之后很快就忘了，干脆就不记了，毕竟现在也很少写语句表的，对吧，大家可别误会，我不是在为偷懒找借口哦。

李老师在课程中对这些基本指令进行一个个地详细讲解，我在这里就挑几个来讲一下就好了，大家对全部内容感兴趣的话可以去看课程或者看一下编程手册都可以，下图就是我在编程手册的截图，内容说明是相当详细的。

一、逻辑取、输出指令

每一个梯级的个触点必须用取指令，并与左母线相连。这里的“取”并不单单指“LD”，而是所有取开头的指令，如LD(取常开)、LDI(取常闭)、LDP(取脉冲上升沿)、LDF(取脉冲下降沿)等。我们能用这个“取”在PLC中掏出个啥呢?其实就是上图的X、Y、M、S、T及C啦，如LD X0、LDI Y1等。简单来说，“取”就像是我们去挑选继电控制电路的各种按钮、继电器的触点一样，用谁取谁。我就想问问，为什么没有LD Money这个指令?

OUT(输出)指令是输出线圈的驱动指令，OUT谁就是将运算结果输出给谁，例如OUT Y0、OUT M1等。当然，OUT后面的线圈也不是随便来的，只能是如上图的Y、M、S、T及C，大家可别写出个OUT X0哦，被取笑了我可不负责。

二、触点串并联指令

与、或、非是逻辑代数的基本运算，我在篇文章中就有提到过。现在运用这些知识的时候到了。如图上图所示，其实“AND(与)”就相当于串联，“OR(或)”就相当于并联。上文提到，梯级的个指令是“取”，显然，“与”、“或”等指令是在“取”之后调用的，如LD X0;AND X1指的是X0“与”X1，即取X0与X1串联。

另外，与、或、非可以和其他指令相互组合成新的指令，例如LDP(取脉冲上升沿)和AND(与)组合成ANDP(与脉冲上升沿);上图的“与非”也可以理解为“与”和“非”的组合。

显然，只要涉及到串并联，与、或、非或其衍生指令的使用都是必不可少的，但是这些指令繁多且复杂，而且我们平时在画梯形图的时候一般都不会去看语句表，所以我在学习这些指令的时候，也只是看了一遍。因为我觉得，只要我们在听课的过程中能理解这些指令，那么即使我们学了就忘，当我们需要这部分知识的时候再去翻看资料就行了。

三、堆栈指令

我为什么要说这个堆栈指令呢?是因为我在学习到这个指令的时候遇到了一个问题，现在就跟大家分享一下。

由于之前的文章有提过堆栈是什么，我在这里不再赘述。堆栈指令又叫多输出指令，在梯形图中，一个梯级有一个或多个公共触点，并从该公共触点分出2条或2条以上支路且每个支路都有自己的触点及输出时，必须要用堆栈指令来编写指令语句表程序。MPS就像是打开仓库的大门，即堆栈指令的起点，MRD就像是向仓库里面堆集货物，堆一个货物就有一个MRD，而MPP就像是关闭仓库的大门，也就是该堆栈的终点。

如下图是一个含有堆栈指令的梯形图，X0、X1、X2、X3都是公共触点，因为它们有分支，控制的不仅仅是一个输出。

图右是课程中图左梯形图的指令语句表程序，我的问题恰恰是在这里。在不看课程语句表的情况下，我尝试自己写出左图的语句表，然后发现竟然与课程的语句表不一样，就是调用X1、X2、X3、X4的时候，

我用的是AND指令，课程中用的却是LD指令。关于这个差异，我尝试用编程软件GX Developer把梯形图转换为语句表程序，发现软件用的也是AND指令，但是我也不知道课程的语句表程序是否正确，希望能有大神留言评论指点一下我。

谈及这个GX Developer，由于我尝试了多遍安装、卸载都没成功(真想换电脑)，还是借用别人的电脑编程的。我的电脑安装的是GX Works2软件，所以在之后的编程学习中我都是用GX Works2，这与李老师讲课过程中用的GX Developer有所不同，GX Works2软件不能把梯形图转换为语句表，也没有时序图可以看，所以我还是建议大家安装GX Developer，因为GX Developer是具有以上两个功能的。

四、置位与复位指令

置位和复位指令的功能是编程元件进行强制操作，其中置位是把元件强制置“1”，而复位是将位元件强制置“0”或将字元件清0。

从手册截图中我们也可以看到，不管是SET还是RST，其动作都是通过触点控制的，而且SET仅仅是能对Y、M、S三种位元件起作用，而RST除了可以操作Y、M、S，还可以操作T、C、D等。

上图是课程中置位和复位的一个应用程序，这其实是起保停程序，X0闭合，Y0置“1”并保持;X1闭合，Y0被置“0”。显然这比普通的起保停程序简单得多，但我个人觉得不是很直观，依我这么笨的脑子，还是乖乖用一般的起保停程序吧。

五、主控指令

所谓“主控”就像是总开关，只有总开关闭合，其后段线路才能接通。上图的回路表示框中MC N YM的N是主控制指令嵌套的层数N0~N7，外层是N0，内层是N7，当然，如果没有嵌套也就没有N1~N7的出现。YM指的是主控指令只能用Y或M元件(不包括特殊M)。

显然，MCR N和MC N YM是成对出现的，就像是上文的MPS和MPP一样，而且N也要首尾呼应。

上图是左边是课程中的一个含有主控指令的梯形图，但是据说这个梯形图在软件中是无法编译的，经过大神指点，我现在把修正后的程序也一并给大家看一下，大家可以自己编程试试看是不是这么回事，然后把结果留言评论告诉我一声，我就可以不用再编一次了，喜滋滋。

六、运算结果取反指令

INV(运算结果取反)指令在梯形图中是一条45°的斜线，它的功能是将指令之前的逻辑运算结果取反，在梯形图中除了不能与左母线直接相连外，可以在任一地方出现。

所谓运算结果取反，为了便于理解，大家可以把INV和LDI(取反)、ANI(与非)等指令作个比较，如下图

所示，个程序是ANI(与非)的使用，第二个是INV(运算结果取反)的使用，当X0和X1均为0时，两个梯形图中的Y0显然有不同的结果。

当X0和X1均为0时，

X0 “与非” X1，此时运算结果为0，所以Y0为0;而X0 “与” X1=0，对“0”这个运算结果取反，此时Y0就为1。运算结果取反指令理解起来还是比较简单的，你们觉得呢?

邯郸西门子S7-200模块代理商|西门子CPU选型

邯郸西门子S7-200模块代理商|西门子CPU选型