

西门子操作面板OP73

产品名称	西门子操作面板OP73
公司名称	浔之漫智控技术（上海）有限公司
价格	.00/件
规格参数	品牌:西门子 产品:触摸屏 产地:德国
公司地址	上海市松江区石湖荡镇塔汇路
联系电话	18771792116

产品详情

STEP7中的S7 Graph顺序控制图形编程语言属于可选的软件包。在这种语言中，工艺过程被划分为若干个顺序出现的步，步中包含控制输出的动作，从一步到另一步的转换由转换条件控制，用Graph表达复杂的顺序控制过程非常清晰，用于编程及故障诊断更为有效，使PLC程序的结构更加易读，它特别适合于生产制造过程。S7 Graph具有丰富的图形、窗口和缩放功能。系统化的结构和清晰的组织显示使S7 Graph对于顺序过程的控制更加有效。

（2）梯形图（LAD）梯形图是使用得多的PLC图形编程语言。梯形图与继电器电路图很相似，具有直观易懂的优点，很容易被工厂熟悉继电器控制的电气人员掌握，特别适合于数字量逻辑控制，有时把梯形图称为电路或程序。

梯形图由触点、线圈和用方框表示的指令框组成。触点代表逻辑输入条件，例如外部的开关、按钮和内部条件等。线圈通常代表逻辑运算的结果，常用来控制外部的指示灯、交流接触器和内部的标志位等。指令框用来表示定时器、计数器或者数学运算等附加指令。

使用编程软件可以直接生成和编辑梯形图，并将它下载到PLC。

触点和线圈等组成的独立电路称为网络（Network），编程软件自动为网络编号。

梯形图中的触点和线圈可以使用物理地址，例如I0.2、Q1.3等。如果在符号表中对某些地址定义了符号，例如令I0.0的符号为“启动”，在程序中可用符号地址“启动”来代替物理地址I0.0，使程序易于阅读和理解。

用户可以在网络号的右边加上网络的标题，在网络号的下面为网络加上注释。还可以选择在梯形图下面自动加上该网络中使用的符号的信息（Symbol Information）。

如果将两块独立电路放在同一个网络内，将会出错。本书为节约篇幅，在插图中梯形图左右两侧垂直“电源线”之间有一个左正右负的直流电源电压，当图2-2网络1中I0.0与I0.1的触点同时接通，或Q4.0与I0.1

的触点同时接通时，有一个假想的“能流”（Power Flow）流过Q4.0的线圈。利用能流这一概念，可以帮助我们更好地理解和分析梯形图，能流只能从左向右流动。

如果没有跳转指令，在网络中，程序中的逻辑运算按从左往右的方向执行，与能流的方向一致。网络之间按从上到下的顺序执行，执行完所有的网络后，下一次循环返回上面的网络（网络1）重新开始执行。

西门子操作面板OP73

浔之漫智控技术（上海）有限公司（sqw-xzm-ssm）

本公司是西门子授权代理商 自动化产品，全新，西门子PLC,西门子屏，西门子数控，西门子软启动，西门子以太网西门子电机，西门子变频器，西门子直流调速器，西门子电线电缆我公司**供应，德国进口

（3）语句表（STL）S7系列PLC将指令表称为语句表（Statement List），它是一种类似于微机的汇编语言中的文本语言，多条语句组成一个程序段。语句表比较适合经验丰富的程序员使用，可以实现某些不能用梯形图或功能块图表示的功能。

（4）功能块图（FBD）功能块图（FBD）使用类似于布尔代数的图形逻辑符号来表示控制逻辑。一些复杂的功能（例如数学运算功能等）用指令框来表示，有数字电路基础的人很容易掌握。功能块图用类似于与门、或门的方框来表示逻辑运算关系，方框的左侧为逻辑运算的输入变量，右侧为输出变量，输入、输出端的小圆圈表示“非”运算，方框被导线连接在一起，信号自左向右流动。图2-4中的控制逻辑和图2-2的控制逻辑与图2-2和图2-3中的相同。西门子公司的“LOGO！”系列微型PLC使用功能块图编程，除此之外，国内很少有人使用功能块图语言。

（5）结构文本（ST）结构文本（ST）是为IEC 61131-3标准创建的一种专用的编程语言。与梯形图相比，它能实现复杂的数学运算，编写的程序非常简洁和紧凑。

STEP7的S7 SCL（结构化控制语言）是符合IEC 61131-3标准的文本语言。它的语言结构与编程语言Pascal和C相似，所以特别适合于习惯使用编程语言的人使用。

SCL适合于复杂的公式计算和优化算法，或管理大量的数据等。

（6）S7 HiGraph 图形编程语言S7 HiGraph属于可选软件包，它用状态图（State Graphs）来描述异步、非顺序过程的编程语言。系统被分解为几个功能单元，每个单元呈现不同的状态，各功能单元的同步信息可以在图形之间交换。需要为不同状态之间的切换定义转换条件，用类似于语句表的语言描述指定状态的动作和状态之间的转换条件。

（7）S7 CFC编程语言 可选软件包CFC（Continuous Function Chart，连续功能图）用图形方式连接程序库中的以块的形式提供的各种功能，包括从简单的逻辑操作到复杂的闭环和开环控制等领域。编程时将这些块复制到图中并用线连接起来即可。

不需要用户掌握详细的编程知识以及PLC的专门知识，只需要具有行业所必需的工艺技术方面的知识，就可以用CFC来编程。

（8）编程语言的相互转换与选用 在STEP7编程软件中，如果程序块没有错误，并且被正确地划分为网络，在梯形图、功能图和语句表之间可以转换，用语句表编写的程序不一定能转换为梯形图，不能转换的网络仍然保留语句表的形式，但是并不表示该网络有错误。

语句表可供习惯用汇编语言编程的用户使用，在运行时间和要求的存储空间方面优。语句表的输入方便

快捷，还可以在每条语句的后面加上注释，便于复杂程序的阅读和理解。在设计通信、数学运算等应用程序时建议使用语句表。

梯形图与继电器电路图的表达方式极为相似，适合于熟悉继电器电路的用户使用。语句表程序较难阅读，其中的逻辑关系很难一眼看出，在设计和阅读有复杂的触点电路的程序时好使用梯形图语言。

功能块图适合于熟悉数字电路的用户使用。

S7 SCL编程语言适合于熟悉编程语言（例如Pascal或C语言）的用户使用，合适于数据处理程序。

S7 Graph、HiGraph和CFC可供有技术背景，但是没有PLC用户使用。S7Graph对顺序控制过程的编程非常方便，HiGraph适合于异步非顺序过程的编程，CFC适合于连续过程控制的编程。

（9）S7-PLCSIM仿真软件 即使没有PLC的硬件，使用S7-PLCSIM仿真软件也可以在计算机上对SIMATIC S7用户程序块进行功能测试，它对于用户程序的调试和PLC编程的学习是非常有用的。

它可以用于用下列语言编写的程序的仿真：LAD、FBD、STL、S7 Graph、S7 HiGraph、S7 SCL和CFC。

2.2 S7-400 CPU的存储区

2.2.1 数制

（1）二进制数 二进制数的1位（bit）只能取0和1这两个不同的值，可以用来表示开关量（或称数字量）的两种不同的状态，例如触点的断开和接通，线圈的通电和断电等。如果该位为1，表示梯形图中对应的位编程元件（例如位存储器M和输出过程映像Q）的线圈“通电”，其常开触点接通，常闭触点断开，以后称该编程元件为1状态，或称该编程元件ON（接通）。如果该位为0，对应的编程元件的线圈和触点的状态与上述的相反，称该编程元件为0状态，或称该编程元件OFF（断开），二进制常数用2#表示，例如2#1111_0110_1001_0001是16位二进制常数在编程手册和编程软件中，位编程元件的1状态和0状态常用TURE和FALSE来表示。

（1）装载存储器 装载存储器可能是RAM和FEPR0M，用于保存不包含符号地址和注释的用户程序和系统数据（组态，连接和模块参数等）。有的CPU有集成的装载存储器，有的可用微存储器卡（MMC）来扩展，CPU31XC的用户程序只能装入插入式的MMC。

断电时数据保存在MMC存储器中，因此数据块的内容基本上被保留。下载程序时，用户程序（逻辑块和数据块）被下载到CPU的装载存储器，CPU把可执行部分复制到工作存储器，符号表和注释保存在编程设备中。

（2）工作存储器 它是集成的高速度存取的RAM存储器，用于存储CPU运行时的用户程序和数据，例如组织块、功能块、功能和数据块。为了保证程序执行的快速性和不过多地占用工作存储器，只有与程序执行有关的块被装入工作存储器。

STL程序中的数据块可以被标识为“与执行无关”（UNLINKED），它们只是存储在装载存储器中。有必要时可以用SFC20“BLKMOV”将它们复制到工作存储器。复位CPU的存储器时，RAM中的程序被清除，FEPR0M中的程序不会被清除。

（3）系统存储器 系统存储器是CPU为用户程序提供的存储器组件，被划分为若干个地址区域。使用指令可以在相应的地址区内对数据直接进行寻址。系统存储器为不能扩展的RAM，用于存放用户程序的操作数据，例如过程映像输入、过程映像输出、位存储器、定时器和计数器、块堆栈（B堆栈）、中断堆栈（I堆栈）和诊断缓冲区等。

系统存储器还提供临时存储器（局域数据堆栈，即L堆栈），用来存储程序块被调用时的临时数据。访问局域数据比访问数据块中的数据更快，用户生成块时，可以声明临时变量（TEMP），它们只在执行该块时有效，执行完后就被覆盖了。

（4）外设I/O存储区 通过外设I/O存储区（PI和PQ），用户可以不经过过程映像输入和过程映像输出，直接访问输入模块和输出模块。不能以位为单位访问外设I/O存储区，只能以字节、字和双字为单位访问。

2.2.5 系统存储器

（1）过程映像输入/输出（I/O）表 在扫描循环开始时，CPU读取数字量输入模块的输入信号的状态，并将它们存入过程映像输入表（Process Image Input，PII）中。

用户程序访问PLC的输入（I）和输出（Q）地址区时，不是去读写数字信号模块中的信号状态，而是访问CPU中的过程映像区。在扫描循环中，用户程序计算输出值，并将它们存入过程映像输出表（Process Image Output，PIQ）。在循环扫描开始时将过程映像输出表的内容写入数字量输出模块。