

# 西门子PLC授权总经销商 S7-1200 6ES7954-8LF03-0AA0 24M 存储卡

产品名称	西门子PLC授权总经销商 S7-1200 6ES7954-8LF03-0AA0 24M 存储卡
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	西门子:全国代理 S7-1200:现货 德国:全新
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	15801997124 15801997124

## 产品详情

西门子PLC授权总经销商 S7-1200 6ES7954-8LF03-0AA0 24M 存储卡

SIMATIC S7，存储卡 用于 S7-1x00 CPU/SINAMICS，3，3V Flash，24 MB

梯形图 (LAD) 电路图的元件（如常闭触点、常开触点和线圈）相互连接构成程序段。  
要创建复杂运算逻辑，可插入分支以创建并行电路的逻辑。并行分支向下打开或直接连接到电源线。用户可向上终止分支。LAD 向多种功能（如数学、定时器、计数器和移动）提供“功能框”指令。STEP 7 不限制 LAD 程序段中的指令（行和列）数。说明 每个 LAD 程序段都必须使用线圈或功能框指令来终止。创建 LAD 程序段时请注意以下规则：  
不能创建可能导致反向能流的分支。+ \* ( ) \$ % & ' = 不能创建可能导致短路的分支。\$ % & = 编程概念  
7.5 编程语言 S7-1200 可编程控制器 186 系统手册, V4.6 11/2022, A5E02486685-AP 7.5.2 功能块图 (FBD) 与 LAD 一样，FBD 也是一种图形编程语言。逻辑表示法以布尔代数中使用的图形逻辑符号为基础。  
要创建复杂运算的逻辑，在功能框之间插入并行分支。  
算术功能和其它复杂功能可直接结合逻辑框表示。STEP 7 不限制 FBD 程序段中的指令（行和列）数。  
7.5.3 SCL 结构化控制语言 (SCL, Structured Control Language) 是用于 SIMATIC S7 CPU 的基于 PASCAL 的\*\*编程语言。SCL 支持 STEP 7 的块结构 (页 175)。可以使用以下三种编程语言之一将程序块包括到项目中：SCL、LAD 和 FBD。SCL 指令使用标准编程运算符，例如，用 (:=) 表示赋值，算术功能 (+ 表示相加，- 表示相减，\* 表示相乘，/ 表示相除)。SCL 也使用标准的 PASCAL 程序控制操作，如 IF-THEN-ELSE、CASE、REPEAT-UNTIL、GOTO 和 RETURN。SCL 编程语言中的语法元素还可以使用所有的 PASCAL 参考。许多 SCL 的其它指令（如定时器和计数器）与 LAD 和 FBD 指令匹配。有关特定指令的更多信息，请参见基本指令 (页 207) 和扩展指令 (页 325) 章节中的特定指令。  
7.5.3.1 SCL 程序编辑器 可以在创建该块时指定任何块类型（OB、FB 或 FC）以便使用 SCL 编程语言。STEP 7 提供包含以下元素的 SCL 程序编辑器：  
用于定义代码块参数的接口部分 用于程序代码的代码部分 包含 CPU 支持的 SCL 指令的指令树

可以直接在代码部分输入指令的 SCL 代码。编辑器包含用于通用代码结构和注释的按钮。要了解更复杂的指令，只需从指令树拖动 SCL 指令并将其放入程序中。也可以使用任意文本编辑器创建 SCL 程序，然后将相应文件导入 STEP 7 中。编程概念 7.5 编程语言 S7-1200 可编程控制器 系统手册, V4.6 11/2022, A5E02486685-AP 187 在 SCL 代码块接口部分，可以声明下列类型的参数：Input、Output、InOut 和 Ret\_Val：这些参数定义代码块的输入变量、输出变量和返回值。执行代码块期间局部使用此处输入的变量名称。通常不会使用变量表中的全局变量名称。Static（仅适用于 FB，上述示例适用于 FC）：代码块使用静态变量在背景数据块中存储静态中间结果。块会一直保留静态数据，直到多个周期后被覆盖。块的名称（此块将其作为多重背景调用）也存储在静态局部数据中。Temp：这些参数是执行代码块期间使用的临时变量。Constant：这些是为代码块指定的常数值。如果从其它代码块调用 SCL 代码块，该 SCL 代码块的参数会显示为输入或输出。本示例中，“Start”和“On”变量（来自项目变量表）相当于 SCL 程序声明表中的“StartStopSwitch”和“RunYesNo”。编程概念 7.5 编程语言 S7-1200 可编程控制器 188 系统手册, V4.6 11/2022, A5E02486685-AP 7.5.3.2 SCL 表达式和运算 构造 SCL 表达式 SCL 表达式是用于计算值的公式。表达式由操作数和运算符（如 \*、/、+ 或 -）组成。操作数可以是变量、常量或表达式。表达式的计算按一定的顺序进行，具体由以下因素决定：每个运算符均具有预定义的优先级，首先执行优先级高的运算。按从左至右的顺序处理优先级相同的运算符。可使用圆括号指定要一起计算的一系列运算符。表达式的结果可用于将值分配给程序使用的变量、用作由控制语句使用的条件、用作其它 SCL 指令的参数或者用于调用代码块。表格 7-2 SCL 中的运算符 类型 操作 操作员 优先级 圆括号（表达式） (, ) 1 数学 乘方 \*\* 2 符号（一元加号）+ 3 符号（一元减号）- 3 倍增 \* 4 除法 / 4 取模 MOD 4 加法 + 5 减法 - 5 比较 小于 < 6 小于或等于 <=

以下示例显示的是 FOR-TO-DO 控制语句。（两种形式的代码在语法上均有效。）FOR x := 0 TO max DO sum := sum + value(x); END\_FOR; FOR x := 0 TO max DO sum := sum + value(x); END\_FOR; 还可以为控制语句提供标签。用语句前的逗点将标签隔开：Label; TIA Portal 信息系统提供完整的 SCL 编程语言参考。条件 条件是一个比较表达式或逻辑表达式，其结果为 BOOL 类型（值为 TRUE 或 FALSE）。以下示例显示了各种类型的条件：#Temperature > 50 #Counter <= 100 #CHAR1 < 'S' 关系表达式 (#Alpha <> 12) AND NOT #Beta 比较和逻辑表达式 5 + #Alpha 算术表达式 条件可以使用算术表达式：如果结果是非零的任何值，则表达式的条件为 TRUE。如果结果为零，则表达式的条件为 FALSE。从 SCL 程序中调用其它代码块 要调用用户程序中的其它代码块，只需使用参数输入 FB 或 FC 的名称（或\*\*地址）。对于 FB，还必须提供 FB 待调用的背景数据块。（参数列表）作为单个背景调用