

九江市西门子中国（授权）一级代理商-西门子选型-西门子技术支持-西门子维修服务

产品名称	九江市西门子中国（授权）一级代理商-西门子选型-西门子技术支持-西门子维修服务
公司名称	广东湘恒智能科技有限公司
价格	.00/件
规格参数	变频器:西门子代理商 触摸屏:西门子一级代理 伺服电机:西门子一级总代理
公司地址	惠州大亚湾澳头石化大道中480号太东天地花园2栋二单元9层01号房（仅限办公）（注册地址）
联系电话	18126392341 15267534595

产品详情

在S7-1200PLC中，若需要根据数组下标值来对数组中元素的访问有两种方式可以实现，一是通过大家熟悉的梯形图来编写程序实现，二是可以通过SCL的编程方式来编写这样的程序实现。

一、举例说明梯形图和SCL如何实现程序编写

这里通过一个简单的例子为例，分别通过这两种方式如何实现通过索引数组的下标值来实现对数组中元素的访问。

例子说明：

假设需要从一组数据中找出一个最大值，并记录这个最大值是这组数据中的第几个数据。

先以大家相对来讲比较熟悉的梯形图的方式来实现此功能。这里我们需要用到通过读取域或写入域的指令，该指令根据索引的下标值来读取数

值中相对应元素的值或写入数到数组中相对应的元素里。此例子中只需要用到读指令，指令位于移动操作指令中的“原有”文件加中。

读取数组中元素值的指令格式如下：

举例：

表示把数据中的data这个数组中的data[5]这个元素的值读取出来放入到MW100这个变量中。

写入数组中元素值的指令格式如下：

举例：

表示把数据20写入到数据块1中的变量名为data的数组中的data[3]元素的存储器中。

在本例中，我们只需要用到第一个指令，接下来设计一个如例题中所要求的梯形图程序。

程序编写思路：

本例题要去找出最大值，并把最大值所处的位置记录下来，编程思路：假设变量MAX_DATA作为最大值的存储器，然后根据下标值（INDEX）的多少去读取相应数值中对用的元素的值放于TEMP_DATA变量中，然后与MAX_DATA做比较，若MAX_DATA的值要小，则进行交换，同时记录INDEX值。然后INDEX加1，可以指向数组中的下一个元素

注：流程图不太标准，但希望对大家理解这个编程思路有所帮助，接下来主要对程序的编写进行介绍。

二、用梯形图编写一个取最大值的程序

前面介绍了读取和写入数组中元素值的指令Field Read和Field Write两条指令,同时给大家简单的分析了程序的设计思路。接下来就使用Field Read来实现本功能，在程序的设计过程中可能还需要用到循环跳转指令。

例子说明：

假设需要从一组数据中找出一个最大值，并记录这个最大值是这组数据中的第几个数据。

第一步：

添加一个全局DB块，并在DB块中建立一个变量名为data_1#，数据类型为数组的变量，用于存储需要找出最大的数据，同时建立一些相应的变量，如下图所示。

第二步：

初始化相应的存储器并把存储最大值的存储器的值设置为最小值。程序如下所示：

第三步：

编写判断数据的挨个比较是否完成，当执行的次数与设定的次数相等时，则表示完成，可以跳出最大值查找的程序，让程序跳转到最后执行。程序如下所示：

第四步：

编写读取数组中元素的值，然后与存储最大值的存储器中的值做比较，用于判断数据存储器存储的值是否是最大值，若不是最大值进行数据交换，同时记录位置，然后INDEX的值加1，同时记录执行次数并与设定

次数做比较，如未达到设定次数，则跳转换前面继续通过Field Read指令读取数据出来继续做比较。

三、使用SCL如何实现程序编写

在前面介绍了使用梯形图的方式来编写一个取最大值的程序，但在S7-1200PLC中，还支持SCL的编程，使用SCL的编程对一些复杂的数据处理会带来很大的方便，下面还是以前面的例子为例说明使用SCL如何编写程序实现。

例子说明：

假设需要从一组数据中找出一个最大值，并记录这个最大值是这组数据中的第几个数据。

使用SCL编程来完成这个例子，这里我们需要用到两个语句

(1) 用于条件判断的语句

IF (条件) THEN (执行语句)

END_IF ;

解析：如果条件满足，则执行THEN后面的语句。

举例：IF “ DATA_A ” <100 HEN

“ DATA_A ” := “ DATA_A ” +1 ;

END_IF;

如果DATA_A的值小于100，则DATA_A的值等于自身加1.

(2) 用于循环执行的语句

```
FOR (执行变量) := (起始值) TO (结束值) BY (自增量) DO  
(后面需要执行的语句);
```

```
END_FOR;
```

解析：从“起始值”开始循环到执行，每循环一次，“执行变量”的值会根据“自增量”的多少进行变化，直到执行到“结束值”时，停止循环执行。

举例：FOR “count” := 0 TO 4 BY 1 DO

```
“Data[count]” = 10;
```

```
END_FOR;
```

把数值10填入到数值Data中的Data[0]到Data[4]的五个元素中,第一次循环时把10填入到Data[0]，第二次循环时把10填入到Data[1]，依次下去。

了解这两条语句后，接下来我们可以设计一个程序，这里我们可以把他建立为一个功能块（FB），方便以后使用。