

# 四川攀枝花西门子PLC一级代理商

产品名称	四川攀枝花西门子PLC一级代理商
公司名称	浔之漫智控技术（上海）有限公司总部
价格	3400.00/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

四川攀枝花西门子PLC一级代理商 四川攀枝花西门子PLC一级代理商

四川攀枝花西门子PLC一级代理商四川攀枝花西门子PLC一级代理商四川攀枝花西门子PLC一级代理商

所谓数值运算，就是通过加减乘除来实现数值的变换。然，在三菱FX PLC 2N中，这些个加减乘除具体又是怎样实现的呢？

### 一、BIN数的四则运算指令

BIN数，即带符号的二进制整数，它的四则运算指令包括加、减、乘、除指令。

#### (1)ADD(加)指令、SUB(减)指令

从编程手册的截图中，可以看到，ADD指令和SUB指令都有脉冲执行型和连续执行型两种，且都可用于16位和32位。当然，它们的适用软元件为字软元件。

加、减指令都有两个源操作数(S1·)、(S2·)和一个目的操作数(D·)，它们在梯形图中的形式如下图所示。在ADD指令中，其代表的含义为(S1)(S2)=(D)，将S1中的数值加上S2中的数值，然后把和存放到D中。同理，在SUB指令中，其代表的含义为(S1)-(S2)=(D)，将S1中的数值减去S2中的数值，然后把差存放到D中。

看过上篇文章的都知道，BIN数是带有符号的，所以它们在运算的过程中，也要考虑到符号的影响。加、减指令在执行后要影响到三个标志位，如下图所示。

M8022这个进位标志位，看过我系列文章的人可能还有点印象，其实我在写到移位指令就提到过它。不记得的可以回顾一下移位指令的内容。

小学生都知道，两个数相加减会涉及到借位和进位的问题，例如十进制数运算 $7+8=15$ ，其中的1就是进位， $15-7=8$ 中的1被借位。同理，PLC的BIN数四则运算也要考虑进位和借位问题，当两个数相加，它们的和大于(D)所能保存的大值后，M8022状态为ON。同理，两个数相减，它们的差小于(D)所能保存的小值后，M8021状态为ON。

所谓(D)所能保存的大值或小值，是指这些参与运算的字软元件所能保存的数值限值，包括符号位在内。例如上图的例子，(D0)=K32767，执行指令ADD D0 K1 D10后，和为K32768，一个16位的字元件，所能保存的大正整数为K32767(包括符号位)，所以D10超出大值，此时进位标志位M8022为ON，另外，从图中可以看到，D10的值变为0，所以此时M8020也为ON。同理，减法的溢出也是大同小异，我在此就不再举例，大家感兴趣的，可以用软件编程验算一下结果。

ADD指令和SUB指令比较简单，实际运用不用我多说大家也知道，反正哪里需要加减，哪里就会出现它们。但要注意的一点是，PLC每扫描一个周期，这些指令就执行一次，此时，重复执行加、减运算可能不是我们需要的，所以应该选择脉冲执行型指令ADDP、SUBP或边沿触发型驱动条件。

## (2)MUL(乘)指令、DIV(除)指令

从编程手册的截图中，可以看到，MUL指令和DIV指令也有脉冲执行型和连续执行型两种，且都可用于16位和32位，但终址D只有在16位运算时才可以用V、Z字元件。和加、减指令一样，乘、除指令都有两个源操作数(S1·)、(S2·)和一个目的操作数(D·)，它们在梯形图中的形式如下图所示。不一样的地方在于乘法指令和除法指令的目的操作数，在用于16位运算时目的操作数占两个字元件D、D+1，用于32位运算时目的操作数占四个字元件D、D+1、D+2、D+3。这是很显然的，因为乘法所得的积往往比因数大很多，若此时仅用一个字元件(16位运算时)保存结果是不够的。另外，除法运算在无法整除的时候，就会有余数，所以就要用多一个字元件来保存余数(16位运算时)。在MUL指令中，这些操作数代表的含义为 $(S1) \times (S2) = (D, D+1)$ ，将S1中的数值乘以S2中的数值，然后把积存放到D、D+1两个连续字元件中。同理，在DIV指令中，其代表的含义为 $(S1) \div (S2) = (D) \dots (D+1)$ ，将S1中的数值除以S2中的数值，然后把商存放到D中，把余数存放到D+1中。其梯形图形式如下图所示，16位和32位所占用的字元件有所不同。

MUL指令和DIV指令也比较简单，实际运用也不用我多说。和加、减指令一样，PLC每扫描一个周期，这些指令就执行一次，此时，重复执行乘、除运算可能不是我们需要的，所以应该选择脉冲执行型指令MULP、DIVP或边沿触发型驱动条件。

除法运算的除数是不能为零的，PLC中也一样，若除数为0，错误标志位M8067=ON。

知道了BIN数的四则运算指令后，我们接着来看加1、减1指令。

## 二、加1指令INC和减1指令DEC

加1、减1，顾名思义，是指该指令执行一次，数值加1或减1。INC指令和DEC指令只有目的操作数(D·)，其梯形图形式如下图所示。

INC指令和DEC指令在执行的过程中不会影响到标志位M8020、M8021、M8022。当采用连续执行型时，显然，D会不断地进行加1或减1操作，此时若超出其限值会怎样呢？其实，INC指令和DEC指令是一个单位累加(累减)环形计数器，如下图所示。

在执行INC指令时，若当前值为-1，加1后其值就变为0，再加1就变为1；若当前值为32767(H7FFF)，加1后变为-32768(H8000)。同理，在执行DEC指令时，若当前值为1，减1后其值就变为0，再减1就变为-1；若当前值为-32768(H8000)，减1后变为32767(H7FFF)。

谈及INC指令，我就会想到一道题：求1 2 3 ..... 99的和。这道题用INC指令显然可以很简单的实现计算，如下图所示。这是我随手编的一段程序，结合INC指令和ADD指令就可以算出1 2 3 ..... 99=4950。当然，这只是随手编的，大家也可以尝试一下其他方法。

### INC指令和DEC指令

经常和变址寻址配合应用在累加或累减及检索等程序中。例如把D0~的D100的数值相加等，在这里我就不编程序啦。

知道了四则运算指令和加1、减1指令后，我们继续来学习一个比较绕的指令：求补码指令NEG。

## 三、求补码指令NEG

NEG指令和INC指令、DEC指令一样，只有一个操作数，驱动条件成立时，将D的数值求补码，然后将结果送回到D中去。

上篇文章提到，在PLC中，正数(符号位为0)直接用其原码表示，而(符号位为1)负数就用其补码表示。而NEG指令是直接对数值进行求反加1(包括符号位)，不管当前值是正数还是负数，所以，其求出来的结果，和定义上的补码有所出入。

例如-3在寄存器中用补码表示(HFFFD)，用NEG指令求补后，结果为3(H0003)；再如6在寄存器中用原码表示(H0006)，用NEG指令求补后，结果为-6的补码(HFFFA)。

显然，NEG指令其实是求相反数的指令，当然，这是基于负数本就是用补码表示的前提下的。