

西门子ET-200模块一级核心供货商6ES7193-0CE30-0XA0

产品名称	西门子ET-200模块一级核心供货商6ES7193-0CE30-0XA0
公司名称	湖南西控自动化设备有限公司
价格	.00/件
规格参数	西门子:授权代理商 PLC:一级代理商 德国:售后保障服务
公司地址	中国(湖南)自由贸易试验区长沙片区开元东路1306号开阳智能制造产业园(一期)4#栋301
联系电话	17838383235 17838383235

产品详情

IEC 61131-3 定义的 46 个函数，你都掌握了吗？

引言：

IEC 61131-3 是一种广泛应用于工业自动化领域的编程标准，其中定义了许多函数用于数字运算、算术运算、位移与位运算、选择、比较和字符串操作等领域。

我们可以通过这些函数来处理任何与算术、逻辑和字符串相关的任务。比如，我可以用 ADD 函数把两个数字加在一起，用 SEL 函数根据条件来选择执行不同的操作，用 CONCAT 函数将多个字符串拼接在一起。无论是进行简单的数学计算，还是处理复杂的逻辑判断，这些函数都能派上用场。

本文将逐一介绍这些函数，并结合实例，帮助读者更好地理解和应用。

2

函数介绍：

2.1

数字运算函数：

ABS：用于计算一个数的值。

例子：如果输入为-5.5，ABS 函数将返回 5.5。

SQRT：用于计算一个数的平方根。

例子：如果输入为 16，SQRT 函数将返回 4。

LOG：用于计算一个数的自然对数，即以 e 为底的对数。

例子：如果输入为 10，LOG 函数将返回 2.302585。

LN：用于计算一个数的十进制对数，即以 10 为底的对数。

例子：如果输入为 100，LN 函数将返回 2。

EXP：用于计算指数函数 e^x 的值。

例子：如果输入为 2，EXP 函数将返回 7.389056。

SIN：用于计算一个数的正弦值。

例子：如果输入为 45，SIN 函数将返回 0.7071068。

COS：用于计算一个数的余弦值。

例子：如果输入为 30，COS 函数将返回 0.8660254。

TAN：用于计算一个数的正切值。

例子：如果输入为 60，TAN 函数将返回 1.732051。

ASIN：用于计算一个数的反正弦值。

例子：如果输入为 0.5，ASIN 函数将返回 30。

ACOS：用于计算一个数的反余弦值。

例子：如果输入为 0.5，ACOS 函数将返回 60。

ATAN：用于计算一个数的反正切值。

例子：如果输入为 1，ATAN 函数将返回 45。

2.2

算术运算函数：

ADD：将输入的两个值相加。

例子：如果输入为 3 和 5，ADD 函数将返回 8。

SUB：将第二个输入值从个输入值中减去。

例子：如果输入为 10 和 3，SUB 函数将返回 7。

MUL：将输入的两个值相乘。

例子：如果输入为 4 和 2，MUL 函数将返回 8。

DIV：将个输入值除以第二个输入值。

例子：如果输入为 10 和 2，DIV 函数将返回 5。

MOD：返回个输入值除以第二个输入值的余数。

例子：如果输入为 10 和 3，MOD 函数将返回 1。

EXPT：返回个输入值的第二个输入值次方。

例子：如果输入为 2 和 3，EXPT 函数将返回 8。

MOVE：将第二个输入值的值赋给个输入值。

例子：如果输入为变量 A 和 5，MOVE 函数将把 5 赋值给变量 A。

2.3

位移与位运算函数：

SHL：将输入值按位左移指定的位数。

例子：如果输入为 2 和 3，则 SHL 函数将把 2 左移 3 位，即将二进制数 00000010 转换为 00010000。因此，SHL 函数将返回十进制数值 16。

SHR：将输入值按位右移指定的位数。

例子：如果输入为 16 和 3，则 SHR 函数将把 16 右移 3 位，即将二进制数 00010000 转换为 00000010。因此，SHR 函数将返回十进制数值 2。

ROR：将输入值按位循环右移指定的位数。

例子：如果输入为 8 和 2，则 ROR 函数将把 8 右移 2 位，并循环移动。8 的二进制表示为 00001000，循环右移 2 位后，得到 00000010。因此，ROR 函数将返回十进制数值 2。

ROL：将输入值按位循环左移指定的位数。

例子：如果输入为 2 和 2，则 ROL 函数将把 2 左移 2 位，并循环移动。2 的二进制表示为 00000010，循环左移 2 位后，得到 00001000。因此，ROL 函数将返回十进制数值 8。

AND：对输入的两个值进行按位与运算。

例子：如果输入为 5 和 3，则 AND 函数将对它们进行按位与运算。5 的二进制表示为 00000101，3 的二进制表示为 00000011。按位与运算的规则是，只有当两个对应位都为 1 时，结果才为 1，否则为 0。因此，按位与运算的结果为：00000001，其十进制表示为 1。因此，AND 函数将返回十进制数值 1。

OR：对输入的两个值进行按位或运算。

例子：如果输入为 5 和 3，则 OR 函数将对它们进行按位或运算。5 的二进制表示为 00000101，3 的二进制表示为 00000011。按位或运算的规则是，只要两个对应位中有一个为 1，结果就为 1，否则为 0。因此，按位或运算的结果为：00000111，其十进制表示为 7。因此，OR 函数将返回十进制数值 7。

XOR：对输入的两个值进行按位异或运算。

例子：如果输入为 5 和 3，则 XOR 函数将对它们进行按位异或运算。5 的二进制表示为 00000101，3 的二进制表示为 00000011。按位异或运算的规则是，只有当两个对应位不相同，结果为 1，否则为 0。因此，按位异或运算的结果为：00000110，其十进制表示为 6。因此，XOR 函数将返回十进制数值 6。

NOT：对输入的值进行按位取反运算。

例子：如果输入为 5，则 NOT 函数将对其进行按位取反运算。5 的二进制表示为 00000101。按位取反运算的规则是，将每个位上的 0 变为 1，将每个位上的 1 变为 0。因此，按位取反运算的结果为：11111010，其十进制表示为 -6（根据补码表示法）。因此，NOT 函数将返回十进制数值 -6。

2.4

选择函数：

SEL：根据条件选择返回两个值中的一个。

例子：如果条件表达式为真，SEL 函数将返回第一个输入值；否则返回第二个输入值。

MAX：返回输入的两个值中较大的一个。

例子：如果输入为 3 和 7，MAX 函数将返回 7。

MIN：返回输入的两个值中较小的一个。

例子：如果输入为 3 和 7，MIN 函数将返回 3。

LIMIT：根据上下限限制输入值的取值范围。

例子：如果输入为 8 和 5 到 10 的范围，LIMIT 函数将返回 5，因为 8 超出了范围。

MUX：根据输入的选择位选择返回多个值中的一个。

例子：如果选择位为 1，MUX 函数将返回个输入值；如果选择位为 0，返回第二个输入值。

2.5

比较函数：

GT：判断个输入值是否大于第二个输入值。

例子：如果输入为 3 和 2，GT 函数将返回 1。

GE：判断个输入值是否大于等于第二个输入值。

例子：如果输入为 3 和 3，GE 函数将返回 1。

LT：判断个输入值是否小于第二个输入值。

例子：如果输入为 3 和 5，LT 函数将返回 1。

LE：判断个输入值是否小于等于第二个输入值。

例子：如果输入为 3 和 3，LE 函数将返回 1。

EQ：判断两个输入值是否相等。

例子：如果输入为 5 和 5，EQ 函数将返回 1。

NE：判断两个输入值是否不相等。

例子：如果输入为 4 和 3，NE 函数将返回 1。

2.6

字符串操作函数：

CONCAT：将两个字符串连接在一起。

例子：如果输入为"Hello"和"World"，CONCAT 函数将返回"HelloWorld"。

LEN：返回字符串的长度。

例子：如果输入为"Hello"，LEN 函数将返回 5。

LEFT：从字符串的左侧截取指定长度的子字符串。 例子：如果输入为 "HelloWorld" 和 3，LEFT

函数将返回 "Hel"。RIGHT：从字符串的右侧截取指定长度的子字符串。例子：如果输入为 "HelloWorld" 和 5，RIGHT 函数将返回 "World"。MID：从字符串的指定位置开始截取指定长度的子字符串。例子：如果输入为 "HelloWorld"、2 和 5，MID 函数将返回 "lloWo"。INSERT：将一个字符串插入到另一个字符串的指定位置。例子：如果输入为 "Hello"、3 和 "World"，INSERT 函数将返回 "HelWorldlo"。DELETE：从字符串的指定位置开始删除指定长度的字符。例子：如果输入为 "HelloWorld"、5 和 5，DELETE 函数将返回 "Hello"。REPLACE：替换字符串中指定位置和长度的子字符串。例子：如果输入为 "HelloWorld"、2、5 和 "abc"，REPLACE 函数将返回 "Heabcorld"。FIND：在字符串中查找指定子字符串的个匹配位置。例子：如果输入为 "HelloWorld" 和 "o"，FIND 函数将返回 5。

3

总结

需要注意的是，具体实现这些函数的方法可能因编程语言和开发环境而异，以上示例仅为了演示基本用法。在实际使用时，请参考相关的编程文档和规范来正确使用这些函数。

这些函数涵盖了各种算术运算、位移与位运算、选择、比较和字符串操作等功能，为程序员们提供了丰富的工具来处理各种任务。无论是做简单的加减乘除，还是进行复杂的逻辑判断和字符串操作，这些函数都能够提供jingque、高效的解决方案。它们就像是程序员的法宝，能够让他们的代码更加简洁、清晰而又高效。

想象一下，当我们使用这些函数时，他们好像是在谱写一段美妙的音乐。他们能够从函数的海洋中挑选出合适的旋律，将代码编织成一幅绚丽多彩的画面。无论是解决实际问题，还是创造出令人惊艳的应用程序，这些函数都是我们的得力助手。

总之，这些 IEC 61131-3 定义的 46 个函数是编程世界的宝藏。它们丰富多样，功能强大，可以帮助程序员们轻松应对各种编程任务。无论是初学者还是经验丰富的人士，都可以从中受益，轻松编写出高效、可靠的代码。