

**颖电 Fairy 系统**

**指令参考手册**

**CORSMART Fairy System  
Instruction Reference Manual**

Ver 1.0

**上海颖电控制技术有限公司**

**Shanghai Corsmart Control Technology Co.,Ltd**



## 目 录

<b>第 1 章 概述</b> .....	<b>1</b>
1.1 目标 .....	1
1.2 指令分类 .....	1
1.3 指令参数 .....	1
1.3.1 输入参数 .....	1
1.3.2 输出参数 .....	2
1.3.3 静态参数 .....	2
<b>第 2 章 数值计算</b> .....	<b>3</b>
2.1 FADD 二输入浮点加法 .....	3
2.2 FSUB 二输入浮点减法 .....	4
2.3 FMUL 二输入浮点乘法 .....	5
2.4 FDIV 二输入浮点除法 .....	5
2.5 MFUN 数学函数 .....	6
2.6 RND 随机函数 .....	7
2.7 TRI 三角函数 .....	7
2.8 PLOY 多项式计算 .....	8
2.9 SUM8 八输入数学统计器 .....	9
2.10 WCAL 整数计算 .....	10
2.11 WNOT 整数位反运算 .....	11
<b>第 3 章 逻辑运算</b> .....	<b>13</b>
3.1 AND 八输入逻辑与 .....	13
3.2 OR 八输入逻辑或 .....	14
3.3 XOR 逻辑异或 .....	15
3.4 NOT 逻辑非 .....	15
3.5 TRIG 通用触发器 .....	16
3.6 XIC 常开检测 .....	16
3.7 XIO 常闭检测 .....	17
3.8 OTE 线圈输出 .....	17
3.9 OTL 锁定输出 .....	18
3.10 OTU 解锁输出 .....	18
浮点数数值比较 .....	19
3.11 WCMP 整数比较 .....	19
<b>第 4 章 类型转换</b> .....	<b>21</b>
4.1 WTOB16 16 位逻辑分配 .....	21
4.2 B16TOW 16 位逻辑组合 .....	22
4.3 WDC16 32 位解码输出 .....	23
4.4 WTOF 整数转实数 .....	24
4.5 FTOW 实数转整数 .....	24
4.6 WTOBCD 整数转 BCD 码 .....	25



4.7	BCDTOW BCD 码转整数 .....	25
<b>第 5 章</b>	<b>信号处理 .....</b>	<b>27</b>
5.1	TON 延时合 .....	27
5.2	TOF 延时断 .....	28
5.3	EDGE 边沿检测 .....	29
5.4	CTU 递增计数器 .....	30
5.5	CNT 计数器 .....	31
5.6	DSFT 开关量选择 .....	32
5.7	MDIG 组合开关量逻辑 .....	33
5.8	SQUR 方波发生器 .....	34
5.9	SGN 信号发生器 .....	36
5.10	PSUM 脉冲量累加器 .....	37
5.11	LST 16 段线性转换 .....	38
5.12	FST 16 段时间信号转换 .....	41
5.13	SWF 多路实数开关 .....	43
5.14	SWW 多路整数开关 .....	44
5.15	TMC 时间处理模块 .....	45
<b>第 6 章</b>	<b>设备操作 .....</b>	<b>48</b>
6.1	DMA 数字软手操 .....	48
6.2	SMA 模拟软手操 .....	49
6.3	ILK4 4 键互锁开关 .....	51
6.4	TPOS 两位位置控制 .....	52
6.5	DEV1 单向设备控制 .....	56
6.6	DEV2 双向设备控制 .....	59
6.7	STEP 步序控制 .....	62
6.8	SSA 数字驱动伺服模块 .....	65
6.9	KBML 键盘模拟量增减 .....	66
<b>第 7 章</b>	<b>过程处理 .....</b>	<b>68</b>
7.1	含手操器 PID 控制 .....	69
7.2	偏差 PID 控制 .....	72
7.3	SST 参数自整定 .....	74
7.4	ERV 偏差计算 .....	76
7.5	LDLG 超前滞后 .....	78
7.6	INTG 积分运算 .....	79
7.7	DIFF 微分运算 .....	80
7.8	FLT 滤波器 .....	81
7.9	TSEL 信号二选一 .....	82
7.10	RSEL 信号三选一 .....	83
7.11	SFT 无扰动切换 .....	84
7.12	LMT 幅度限制 .....	85
7.13	LMV 变化速率限制 .....	85
7.14	BLN2 二输出平衡模块 .....	86



---

7.15	TAS 时域统计模块.....	87
7.16	TDS 时域开关量统计模块 .....	88
7.17	DLY 滞后模块 .....	89
7.18	PTC 热力性质计算.....	90
7.19	DFML 差分方程.....	91
7.20	FUZY 模糊控制器.....	92
7.21	SALM 慢信号报警.....	94
7.22	ACCU 累积计算.....	95
7.23	FLW 流量计算.....	96
<b>第 8 章</b>	<b>特殊指令 .....</b>	<b>98</b>
8.1	STS 信号状态.....	98
8.2	EQU 实数赋值.....	99
8.3	WEQU 整数赋值 .....	99
8.4	EXEC 过程跳转 .....	100



# 第1章 概述

## 1.1 目标

本文档面向工程组态人员，用于熟悉了解每个指令的具体功能用法。

本次设计的指令指令与前一版本的主要区别在于，为了兼顾 FBD（Function Block Diagram，功能块图）设计和 LD（Lad Diagram，梯形图）设计的需要，很多指令增加了 EIN（Enable In）和 EOUT（Enable Out）引脚，一些相关的引脚的功能设计也做出了调整。

此外，对于功能相似的功能块，进行了功能合并，使当前版本的指令集更为紧凑。

## 1.2 指令分类

本次设计的指令共分 7 类

类别	说明	ID
数值计算	实数和整数的数值计算	1
逻辑运算	产生布尔结果的各种逻辑运算	2
类型转换	实数、整数、布尔之间的相互换算	3
信号处理	对信号进行处理或产生信号的指令	4
设备操作	与设备操作相关的各种指令	5
过程处理	与过程控制相关的各种指令	6
特殊指令	其他特殊用途的指令	7

## 1.3 指令参数

指令参数，在功能块图（FBD）中也有时也称为功能块引脚。

按参数作用可以分为三类，分别是输入参数、输出参数和静态参数。

### 1.3.1 输入参数

输入参数可以是常数、变量或连线这三种方式中的一种。

常数参数是输入参数的缺省方式，在添加指令后，所有输入参数系统都会赋予一个缺省数值，用户可在组态中修改。常数参数在指令运算过程中永远保持不变。

变量参数在运算过程中，引用变量的实时值。所引用的变量可以是本控制器的变量（本地引用），也可以是 Fairy 系统中其他数据源的变量（远程引用）；

连线参数只能在功能块图中使用。组态时，使用线条将输入端口与其他（或自身）指令



的输出端口相连，计算时引用该输出端口的实时值。连线参数只能引用同一功能块图中的指令。

### 1.3.2 输出参数

输出参数可以选择赋予变量、连线，或者不赋值。

通过将输出参数赋予变量，使得变量的实时值与输出端口的实时值和状态保持一致。

同样，通过连线，其他指令的输入参数可以使用输出端口的实时值（和状态）。

输出参数可以同时使用变量赋值和连线赋值。

输出参数也可以选择 not 赋值。由于本版本开始，用户可以在工艺图组态中使用“指令. 引脚”的形式直接使用指令的输出参数，因此不必如前一版本那样，将画面显示需要的数据先赋予变量。

系统不限制变量的多次赋值，即存在多个地方向同一个变量赋值。

正确应用变量多次赋值，可以构建类似子程序调用的效果，简化组态工作。

错误组态引起的变量多次赋值，会使得变量的数值发生各种不可预见的扰动。变量的多次赋值可以通过组态工具的“变量引用”功能进行检查。

### 1.3.3 静态参数

静态参数是本次新增的参数类型。这类参数是指令计算过程中用于保存中间结果的参数。

静态参数仅供用户在线调试时观察，用于帮助诊断组态错误的排除，但不可对这些参数进行任何修改，也不能在组态或画面中引用。



## 第2章 数值计算

### 介绍

进行实数和整数的运算。

该类指令中都含有 EIN 和 EOUT 引脚的指令，如果没有特殊说明，当 EIN=1 时，指令进行正常计算；当 EIN=0 时，指令的计算停止，输出保持在最后的计算结果。EOUT 的值与 EIN 始终保持一致。

该类指令包括：

序号	标记	名称	说明
1	FADD	二输入浮点加法	
2	FSUB	二输入浮点减法	
3	FMUL	二输入浮点乘法	
4	FDIV	二输入浮点除法	
5	MFUN	数学函数	取代上一版本的 EXP、LOG、ABS、SQR
6	RND	随机函数	
7	TRI	三角函数	取代上一版本的 SIN、COS、TAN、ATN 四个指令，另外新加了 acsin accos actan sec csc 指令
8	POLY	多项式计算	
9	SUM8	八输入统计器	
10	WCAL	整数计算	取代上一版本的整数加、减、乘、除、取模、逻辑与、逻辑或、逻辑异或、逻辑同或指令
11	WNOT	整数位反运算	

### 2.1 FADD 二输入浮点加法

#### 名称

标记：FADD

名称：二输入浮点加法

#### 结构

作用	标记	类型	缺省值	说明
----	----	----	-----	----



输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X1	实数	0	输入变量 1
输入	K1	实数	1	变量 1 的系数
输入	X2	实数	0	输入变量 2
输入	K2	实数	1	变量 2 的系数
输入	C	实数	0	偏置量
输出	Y	实数	0	计算结果

## 描述

这是一个含比例和偏置的二输入浮点加法指令，每个输入端可以进行比例转换。输入信号经比例转换后，再加上偏置，作为最后的输出。

$$\text{指令的表达式为: } Y = K_1 X_1 + K_2 X_2 + C$$

输入使能 EIN 和输出使能 EOUT 的作用见第一章概述及第二章介绍部分，本章其余功能块不再重复叙述。

## 2.2 FSUB 二输入浮点减法

### 名称

标记：FSUB

名称：二输入浮点减法

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X1	实数	0	输入变量 1
输入	X2	实数	0	输入变量 2
输出	Y	实数	0	计算结果

### 描述

这是一个二输入浮点减法指令，输出端的值等于输入 X1 的值减去输入 X2 的值。

$$\text{指令的表达式为: } Y = X_1 - X_2$$



## 2.3 FMUL 二输入浮点乘法

### 名称

标记：FMUL

名称：二输入浮点乘法

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X1	实数	0	输入变量 1
输入	X2	实数	0	输入变量 2
输入	K	实数	1	比例系数
输入	C	实数	0	偏置量
输出	Y	实数	0	计算结果

### 描述

这是一个二输入浮点乘法指令，计算结果可以进行比例和偏置转换。

指令的表达式为： $Y = KX_1X_2 + C$

## 2.4 FDIV 二输入浮点除法

### 名称

标记：FDIV

名称：二输入浮点除法

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X1	实数	0	输入变量 1
输入	X2	实数	1	输入变量 2



输入	K	实数	1	系数
输入	C	实数	0	偏置量
输出	Y	实数	0	计算结果

## 描述

这是一个二输入除法指令，计算结果可以进行比例和偏置转换。

$$\text{指令的表达式为: } Y = K \frac{X_1}{X_2} + C, X_2 \neq 0$$

当  $X_2=0$  时，指令计算无效，输出 Y 等于 0，输出的坏点标记被置位。

## 2.5 MFUN 数学函数

### 名称

标记：MFUN

名称：数学函数

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	FUN	整数	0	函数选择
输入	X	实数	0	输入变量
输出	Y	实数	0	计算结果

### 描述

根据 FUN 参数的设置，进行不同的函数计算，FUN 的选择见下表：

符号	作用	数值标记
X	计算 X 的输入绝对值	0
sqrt(x)	计算输入 X 的平方根，要求 $X \geq 0$	1
exp(x)	计算指数函数， $Y = e^x$	2
ln(x)	进行以 e 为底的对数计算， $Y = \ln X, X > 0$	3
rad(x)	计算角度转换弧度，X 是角度值，Y 是弧度值	4



deg(x)	计算弧度转换角度，X 是弧度值，Y 是角度值	5
--------	------------------------	---

当 FUN 输入无效时，指令停止计算，输出保持在最后一次有效计算的结果。

当输入 X 无效时，指令计算无效，输出 Y 等于 0，输出的坏点标志被置位。

## 2.6 RND 随机函数

### 名称

标记：RND

名称：随机函数

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	L	实数	0	随机数下限
输入	H	实数	1	随机数上限
输出	Y	实数	0	计算结果

### 描述

指令的表达式为： $Y = rand(L, H)$ ，输出 Y 的实数范围： $L \leq Y < H$ 。

输出值在每次功能块计算时被更新，变化周期与页面计算周期相同。

## 2.7 TRI 三角函数

### 名称

标记：TRI

名称：三角函数

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能



输出	EOUT	布尔	1	输出使能
输入	FUN	整数	0	选择计算的三角函数
输入	X	实数	0	输入变量
输出	Y	实数	0	计算结果

## 描述

根据 FUN 指定的功能，进行三角函数的计算。

FUN 的有效值：

符号	作用	数值标记
sin	正弦函数，计算 $Y=\sin(X)$	0
cos	余弦函数，计算 $Y=\cos(X)$	1
tan	正切函数，计算 $Y=\tan(X)$ , $X \neq \pi/2+k\pi, k \in \mathbb{Z}$	2
cot	余切函数，计算 $Y=\cot(X)$ , $X \neq k\pi, k \in \mathbb{Z}$	3
sec	正割函数，计算 $Y=\sec(X)$ , $X \neq \pi/2+k\pi, k \in \mathbb{Z}$	4
csc	余割函数，计算 $Y=\csc(X)$ , $X \neq k\pi, k \in \mathbb{Z}$	5
arcsin	反正弦函数，计算 $Y=\arcsin(X)$ , $-1 \leq X \leq 1$	6
arccos	反余弦函数，计算 $Y=\arccos(X)$ , $-1 \leq X \leq 1$	7
arctan	反正切函数，计算 $Y=\arctan(X)$	8

sin、cos、tan、cot、sec、csc 函数的输入 X 为弧度。

arcsin、arccos、arctan 函数的输出结果为弧度。

## 2.8 PLOY 多项式计算

### 名称

标记：PLOY

名称：多项式计算

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X	实数	0	输入变量
输入	C0	实数	0	多项式系数
输入	C1	实数	0	多项式系数
输入	C2	实数	0	多项式系数



输入	C3	实数	0	多项式系数
输入	C4	实数	0	多项式系数
输入	C5	实数	0	多项式系数
输入	C6	实数	0	多项式系数
输入	C7	实数	0	多项式系数
输入	C8	实数	0	多项式系数
输出	Y	实数	0	多项式输出

## 描述

指令的表达式为：

$$Y = C_0 + C_1X + C_2X^2 + C_3X^3 + C_4X^4 + C_5X^5 + C_6X^6 + C_7X^7 + C_8X^8$$

## 2.9 SUM8 八输入数学统计器

### 名称

标记：SUM8

名称：八输入数学统计器

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X1	实数	0	输入变量 1
输入	X2	实数	0	输入变量 2
输入	X3	实数	0	输入变量 3
输入	X4	实数	0	输入变量 4
输入	X5	实数	0	输入变量 5
输入	X6	实数	0	输入变量 6
输入	X7	实数	0	输入变量 7
输入	X8	实数	0	输入变量 8
输入	FLAG	整数	0	输入变量使能标志
输入	MODE	整数	0	计算方式，0=总和，1=平均，2=最大，3=最小。
输出	Y	实数	0	计算结果
输出	VAD	整数	0	有效的输入个数，范围 0~8



## 描述

本指令对 8 个浮点变量按照 MODE 设置的方式进行计算，计算结果通过 Y 输出。

MODE 设置计算方式：

符号	作用	数值标记
总和	计算输入信号的和， $Y = \sum_n X_n$ ，n 是有效输入信号个数	0
平均	计算输入信号的平均值， $Y = \frac{\sum_n X_n}{n}$ ，n 是有效输入信号个数	1
最大	计算输入信号中的最大值， $Y = \max(X_1, X_2, \dots, X_n)$	2
最小	计算输入信号中的最小值， $Y = \min(X_1, X_2, \dots, X_n)$	3

输入变量使能 FLAG 决定参与运算的变量，FLAG 数值二进制的每个数位与一个输入信号相对应。若数位 (bit) 上的值=1，则相应的输入参与运算，否则该输入不参与运算。

bit	7	6	5	4	3	2	1	0
输入	X8	X7	X6	X5	X4	X3	X2	X1

例如 FLAG=15，二进制表示为 00001111，即只有 X1、X2、X3、X4 四个输入参与运算；FLAG=129，二进制表示为 10000001，即只有 X1、X8 两个输入参与运算。

当 FLAG=0 时，表示所有输入信号全都不参与运算，此时输出结果不管 MODE 取何种操作，输出 Y 等于 0。

## 应用

组态中如果参与运算的变量个数固定不变，FLAG 使用常量即可。如果参与运算的变量个数会发生变化，可以使用 BTOW 指令的计算结果作为 FLAG 输入。

## 2.10 WCAL 整数计算

### 名称

标记：WCAL

名称：整数计算

### 结构



作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X1	整数	0	输入变量 1
输入	FUN	整数	0	计算方式
输入	X2	整数	0	输入变量 2
输出	Y	整数	0	计算结果

## 描述

该指令根据 FUN 参数的设置，进行整数的加、减、乘、除、取模（余数）、逻辑与、逻辑或、逻辑异或等运算。

符号	作用	数值标记
ADD	加法运算, $Y = X_1 + X_2$	0
SUB	减法运算, $Y = X_1 - X_2$	
MUL	乘法运算, $Y = X_1 \times X_2$	
DIV	除法运算, $Y = \left[ \frac{X_1}{X_2} \right], X_2 \neq 0$	
MOD	取模运算, $Y = X_1 \bmod X_2, X_2 \neq 0$	
AND	逻辑位与, 对 X1 和 X2 各数据位进行逻辑操作, 形成 Y 上相应位的结果	
OR	逻辑位或, 同上	
XOR	逻辑异或, 同上	
NXOR	逻辑同或, 同上	

## 2.11 WNOT 整数位反运算

### 名称

标记: WNOT

名称: 整数位反运算

### 结构

作用	标记	类型	缺省值	说明
----	----	----	-----	----



输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X	整数	0	输入变量
输出	Y	整数	0	计算结果

## 描述

指令的表达式为： $Y = \text{not } X$ ，Y 是 X 逐位求反后的结果。



## 第3章 逻辑运算

### 介绍

该类算法主要是涉及到布尔量的一些指令，该类指令的最后结果全部是布尔值。

该类指令包括：

序号	标记	名称
1	AND	八输入逻辑与，仅应用于功能块图
2	OR	八输入逻辑或，仅应用于功能块图
3	XOR	二输入逻辑异或，仅应用于功能块图
4	NOT	逻辑非，仅应用于功能块图
5	TRIG	通用触发器，仅用于功能块图
6	XIC	常开检测，仅应用于梯形图
7	XIO	常闭检测，仅应用于梯形图
8	OTE	线圈输出，仅应用于梯形图
9	OTL	锁定输出，仅应用于梯形图
10	OTU	解锁输出，仅应用与梯形图
11	CMP	浮点数值比较
12	WCMP	整数数值比较

### 3.1 AND 八输入逻辑与

#### 名称

标记：AND

名称：八输入逻辑与

#### 结构

作用	标记	类型	缺省值	说明
输入	B1	布尔	1	输入变量 1
输入	B2	布尔	1	输入变量 2
输入	B3	布尔	1	输入变量 3
输入	B4	布尔	1	输入变量 4
输入	B5	布尔	1	输入变量 5



输入	B6	布尔	1	输入变量 6
输入	B7	布尔	1	输入变量 7
输入	B8	布尔	1	输入变量 8
输出	Q	布尔	0	输出结果

## 描述

对所有的输入变量，求逻辑与。当所有的输入全部等于 1 时，输出 Q 等于 1；任何一个输入等于 0 时，输出 Q 等于 0。

对于不使用的输入引脚，设置为常数 1，可以不影响结果的产生。

## 3.2 OR 八输入逻辑或

### 名称

标记：OR

名称：八输入逻辑或

### 结构

作用	标记	类型	缺省值	说明
输入	B1	布尔	0	输入变量 1
输入	B2	布尔	0	输入变量 2
输入	B3	布尔	0	输入变量 3
输入	B4	布尔	0	输入变量 4
输入	B5	布尔	0	输入变量 5
输入	B6	布尔	0	输入变量 6
输入	B7	布尔	0	输入变量 7
输入	B8	布尔	0	输入变量 8
输出	Q	布尔	0	输出结果

### 描述

对所有的输入变量，求逻辑或。当所有的输入全部等于 0 时，输出 Q 等于 0；任何一个输入等于 1 时，输出 Q 等于 1。

对于不使用的输入引脚，设置为常数 0，可以不影响结果的产生。



### 3.3 XOR 逻辑异或

#### 名称

标记: XOR

名称: 逻辑异或

#### 结构

作用	标记	类型	缺省值	说明
输入	B1	布尔	0	输入变量 1
输入	B2	布尔	0	输入变量 2
输出	Y	布尔	0	计算结果

#### 描述

指令的表达式为:  $Y = B1 \text{ xor } B2$

B1 和 B2 相同, 则  $Y=0$ ; B1 和 B2 不同, 则  $Y=1$ 。

### 3.4 NOT 逻辑非

#### 名称

标记: NOT

名称: 逻辑非

#### 结构

作用	标记	类型	缺省值	说明
输入	B	布尔	0	输入变量
输出	Y	布尔	0	计算结果

#### 描述

指令的表达式为:  $Y = \text{not } B$

Y 为 B 的取反。



### 3.5 TRIG 通用触发器

#### 名称

标记: TRIG

名称: 通用触发器

#### 结构

作用	标记	类型	缺省值	说明
输入	X1	布尔	0	复位端
输入	X2	布尔	0	置位端
输入	FUN	整数	0	触发器功能设置
输出	Q	布尔	0	计算结果

#### 描述

根据 FUN 的设置, 决定触发器的功能。

符号	作用	数值标记
RR-S	优先复位 RS 触发器, X1 是触发器的 R 端, X2 是触发器的 S 端。R=1 时, Q=0; R=0, S=1 时, Q=1; R=0, S=0 时, 输出 Q 不变。	0
R-SS	简称 R-SS 触发器, X1 是触发器的 R 端, X2 是触发器的 S 端。S=1 时, Q=1; R=1, S=0 时, Q=0; R=0, S=0 时, 输出 Q 不变。	1
JK	X1 是触发器的 J 端, X2 是触发器的 K 端。当 J=0, K=0 时, 输出 Q 保持不变, 即 $Q_{n+1} = Q_n$ , J=0, K=1 时, 输出 Q=0; J=1, K=0 时, 输出 Q=1; 当 J=1, K=1 时, 输出 Q 发生一次翻转, 即 $Q_{n+1} = \overline{Q_n}$ 。	2

### 3.6 XIC 常开检测

#### 名称



标记: XIC

名称: 常开检测

## 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	0	输入使能
输出	Y	布尔	0	逻辑结果
输入	X	布尔	0	检测信号

当输入 EIN=1 且 X=1 时, 输出 Y 等于 1; 其他条件输出 Y 等于 0。

## 3.7 XIO 常闭检测

### 名称

标记: XIO

名称: 常闭检测

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	0	输入使能
输出	Y	布尔	0	逻辑结果
输入	X	布尔	0	检测信号

当输入 EIN=1 且 X=0 时, 输出 Y 等于 1; 其他条件输出 Y 等于 0。

## 3.8 OTE 线圈输出

### 名称

标记: OTE

名称: 线圈输出

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	0	输入使能



输出	EOUT	布尔	0	输出使能
输出	Y	布尔	0	线圈输出

当 EIN=0 时, EOUT=0, Y=0;

当 EIN=1 时, EOUT=1, Y=1。

### 3.9 OTL 锁定输出

#### 名称

标记: OTL

名称: 锁定输出

#### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	0	输入使能
输出	EOUT	布尔	0	输出使能
输出	Y	布尔	0	线圈输出

当 EIN=0 时, EOUT=0, Y 值不变;

当 EIN=1 时, EOUT=1, Y=1;

### 3.10 OTU 解锁输出

#### 名称

标记: OTU

名称: 解锁输出

#### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	0	输入使能
输出	EOUT	布尔	0	输出使能
输出	Y	布尔	0	线圈输出

当 EIN=0 时, EOUT=0, Y 值不变;

当 EIN=1 时, EOUT=1, Y=0;



## 应用

OTU 指令通常与 OTL 指令联用，OTL 用于锁定信号输出，而 OTU 用于解锁该信号。

### 3.11 CMP 浮点数数值比较

## 名称

标记：CMP

名称：浮点数数值比较

## 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	Y	布尔	1	计算结果
输入	X1	实数	0	输入变量 1
输入	MODE	整数	0	比较方式
输入	X2	实数	0	输入变量 2

## 描述

该指令根据指定的比较方式，对两个输入的数值进行比较。

当 EIN=0 时，输出 Y 等于 0；当 EIN=1 时，输出 Y 由：比较方式 MODE、输入 X1 和 X2 的数值决定：

符号	作用	数值标记
=	相等比较，当 X1=X2 时，Y=1；否则 Y=0	0
≠	不等比较	1
<	小于比较，当 X1<X2 时，Y=1；否则 Y=0	2
>	大于比较	3
≤	小于等于比较	4
≥	大于等于比较	5

### 3.12 WCMP 整数比较

## 名称

标记：WCMP



名称：整数比较

## 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	Y	布尔	1	计算结果
输入	X1	整数	0	输入变量 1
输入	MODE	整数	0	比较方式
输入	X2	整数	0	输入变量 2

## 描述

该指令根据指定的比较方式，对两个输入的数值进行比较。

当 EIN=0 时，输出 Y 等于 0；当 EIN=1 时，输出 Y 由：比较方式 MODE、输入 X1 和 X2 的数值决定：

符号	作用	数值标记
=	相等比较，当 X1=X2 时，Y=1；否则 Y=0	0
≠	不等比较	1
<	小于比较，当 X1<X2 时，Y=1；否则 Y=0	2
>	大于比较	3
≤	小于等于比较	4
≥	大于等于比较	5



## 第4章 类型转换

### 介绍

该类指令主要是涉及到整数、布尔数、实数之间的一些转换运算。

该类指令中含有 EIN 和 EOUT 引脚的指令，如果没有特殊说明的，当 EIN=1 时，指令进行正常计算；当 EIN=0 时，指令的计算停止，输出保持在最后的计算结果。EOUT 的值与 EIN 始终保持一致。

该类指令包括：

序号	标记	名称
1	WTOB16	16 位逻辑分配
2	B16TOW	16 位逻辑组合
3	WDC16	16 位解码输出
4	WTOF	整数转实数
5	FTOW	实数转整数
6	WTOBCD	整数转 BCD 码
7	BCDTOW	BCD 码转整数

### 4.1 WTOB16 16 位逻辑分配

#### 名称

标记：WTOB16

名称：16 位逻辑分配

#### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	0	输入使能
输出	EOUT	布尔	0	输出使能
输入	X	整数	0	输入变量
输出	Y0	布尔	0	X 的第 0 位(最低位)
输出	Y1	布尔	0	X 的第 1 位
输出	Y2	布尔	0	X 的第 2 位
输出	Y3	布尔	0	X 的第 3 位



输出	Y4	布尔	0	X 的第 4 位
输出	Y5	布尔	0	X 的第 5 位
输出	Y6	布尔	0	X 的第 6 位
输出	Y7	布尔	0	X 的第 7 位
输出	Y8	布尔	0	X 的第 8 位
输出	Y9	布尔	0	X 的第 9 位
输出	Y10	布尔	0	X 的第 10 位
输出	Y11	布尔	0	X 的第 11 位
输出	Y12	布尔	0	X 的第 12 位
输出	Y13	布尔	0	X 的第 13 位
输出	Y14	布尔	0	X 的第 14 位
输出	Y15	布尔	0	X 的第 15 位（最高位）

## 描述

将一个整型数的低 16 位置给 16 个布尔变量，X 的值 0 ~ 65535。

## 4.2 B16TOW 16 位逻辑组合

### 名称

标记: B16TOW

名称: 16 位逻辑组合

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	0	输入使能
输出	EOUT	布尔	0	输出使能
输入	X0	布尔	0	Y 的第 0 位（最低位）
输入	X1	布尔	0	Y 的第 1 位
输入	X2	布尔	0	Y 的第 2 位
输入	X3	布尔	0	Y 的第 3 位
输入	X4	布尔	0	Y 的第 4 位
输入	X5	布尔	0	Y 的第 5 位
输入	X6	布尔	0	Y 的第 6 位
输入	X7	布尔	0	Y 的第 7 位
输入	X8	布尔	0	Y 的第 8 位
输入	X9	布尔	0	Y 的第 9 位
输入	X10	布尔	0	Y 的第 10 位



输入	X11	布尔	0	Y 的第 11 位
输入	X12	布尔	0	Y 的第 12 位
输入	X13	布尔	0	Y 的第 13 位
输入	X14	布尔	0	Y 的第 14 位
输入	X15	布尔	0	Y 的第 15 位（最高位）
输出	Y	整数	0	输出变量

## 描述

将 16 个布尔变量组合成一个整型数，作为输出的低 16 位，输出的高位部分补 0，Y 的值 0 ~ 65535。

## 4.3 WDC16 32 位解码输出

### 名称

标记：WDC32

名称：32 位解码输出

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	0	输入使能
输出	EOUT	布尔	0	输出使能
输入	X	整数	0	输入变量
输出	Y	整数	0	按位输出

### 描述

当 X=0 时，输出 Y 等于 0；

当 X=n 时（ $1 \leq n \leq 32$ ），输出 Y 的二进制相应位上被置 1，X=1 对应 bit 0，X=2 对应 bit 1，以此类推。

X	Y 数值（十进制）	Y 数值（二进制）
0	0	0
1	1	1
2	2	10
...	...	...
31	1073741824	10000000000000000000000000000000
32	2147483648	10000000000000000000000000000000



## 4.4 WTOF 整数转实数

### 名称

标记: WTOF

名称: 整数转实数

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	0	输入使能
输出	EOUT	布尔	0	输出使能
输入	X	整数	0	输入变量
输入	SGN	布尔	1	符号选择
输出	Y	实数	0	输出变量

### 描述

将整数变量按值转换成实数。

当 SGN=0 时, X 被作为无符号整数转换成实数。

当 SGN=1 时, X 被作为有符号整数转换成实数。

### 应用

组态工具在显示整数时采用有符号方式表示, 因此如果 SGN=1, 则输出 Y 的结果与输入 X 相同。

在 SGN=0 时, 当  $X > 0$ , 输出 Y 与输入 X 相同; 当  $X < 0$  时,  $Y = 4294967296 + X$ 。

## 4.5 FTOW 实数转整数

### 名称

标记: FTOW

名称: 实数转整数



## 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	0	输入使能
输出	EOUT	布尔	0	输出使能
输入	X	实数	0	输入变量
输出	Y	整数	0	输出变量

## 描述

将浮点变量的整数部分赋给一个整形变量，舍弃浮点数的小数部分，浮点数如果小于 0，则输出等于浮点数整数部分的补码。

## 4.6 WTOBCD 整数转 BCD 码

## 名称

标记：WTOBCD

名称：整数转 BCD 码

## 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	0	输入使能
输出	EOUT	布尔	0	输出使能
输入	X	整数	0	输入变量
输出	Y	整数	0	输出变量

## 描述

将一个整形变量转换成 BCD 码表示的整型变量。X 的值 0 ~ 99999999，针对不能成功转换成 BCD 码的 X 输入，指令计算无效，输出 Y 数值等于 0，坏点标志被置位。

## 4.7 BCDTOW BCD 码转整数

## 名称

标记：BCDTOW



名称：BCD 码转整数

## 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	0	输入使能
输出	EOUT	布尔	0	输出使能
输入	X	整数	0	输入变量
输出	Y	整数	0	输出变量

## 描述

将 BCD 码表示的整数，转换成整数，X 的数值应该是有效的 BCD 码（8 个数位）。如果输入 X 无效，输出 Y 数值等于 0，坏点标志被置位。



## 第5章 信号处理

### 介绍

该类算法主要是涉及到信号处理相关运算功能,可以利用指令构成特定的表达式或实现某种算法。

该类指令中含有 EIN 和 EOUT 引脚的指令,如果没有特殊说明的,当 EIN=1 时,指令进行正常计算;当 EIN=0 时,指令的计算停止,输出保持在最后的计算结果。EOUT 的值与 EIN 始终保持一致。

该类指令包括:

序号	标记	名称
1	TON	延时合
2	TOF	延时断
3	EDGE	边沿检测
4	CTU	递增计数器
5	CNT	双输入计数器,仅用于功能块图
6	DSFT	开关量选择,仅用于功能块图
7	MDIG	组合逻辑,仅用于功能块图
8	SQUR	方波发生器
9	SGN	信号发生器
10	PSUM	脉冲量累加器
11	LST	分段线性转换
12	FST	分段时间信号转换
13	SWF	多路实数开关
14	SWW	多路整数开关
15	TMC	时间处理模块

### 5.1 TON 延时合

#### 名称

标记: TON

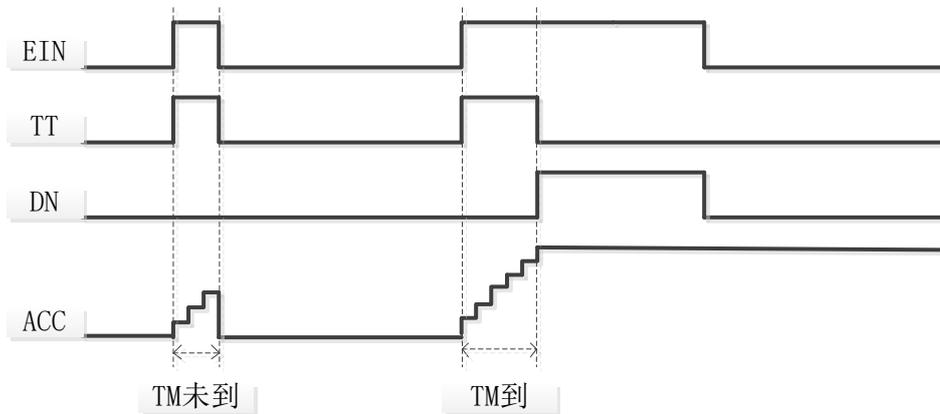
名称: 延时合

#### 结构



作用	标记	类型	缺省值	说明
输入	EIN	布尔	0	输入使能
输出	EOUT	布尔	0	输出使能
输入	TM	整数	0	以毫秒为单位的定时值
输出	TT	布尔	0	计时标志，表示正在进行计时
输出	DN	布尔	0	定时输出
输出	ACC	整数	0	计时累计时间

## 描述



当  $EIN=0$  时， $DN=0$ ， $TT=0$ ， $ACC=0$ 。

当  $EIN$  从 0 变为 1，开始进行计时。

在计时期间，输出  $ACC$  等于计时的毫秒值，输出  $TT=1$ 。

当  $ACC \geq TM$  后，计时结束， $TT=0$ ， $DN=1$ 。此后若  $EIN$  不发生变化，则输出保持不变。

## 5.2 TOF 延时断

### 名称

标记：TOF

名称：延时断

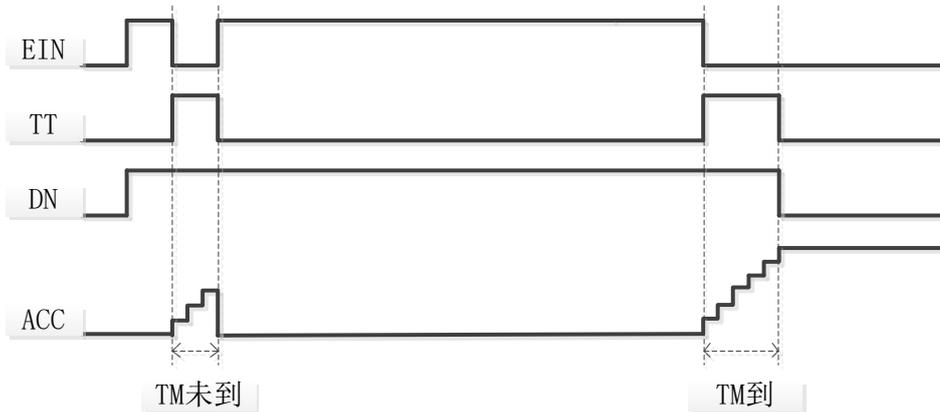
### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	0	输入使能
输出	EOUT	布尔	0	输出使能



输入	TM	整数	0	以毫秒为单位的定时值
输出	TT	布尔	0	计时标志，表示正在进行计时
输出	DN	布尔	0	定时输出
输出	ACC	整数	0	计时累计时间

### 描述



当 EIN=1 时，DN=1，TT=0，ACC=0。

当 EIN 从 1 变为 0 时，开始进行计时。

在计时期间，输出 ACC 等于计时的毫秒值，输出 TT=1。

当  $ACC \geq TM$  后，计时结束，TT=0，DN=0。此后若 EIN 不发生变化，则输出保持不变。

## 5.3 EDGE 边沿检测

### 名称

标记：EDGE

名称：边沿检测

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	0	输入变量
输出	Y	布尔	0	计算结果
输入	FUN	整数	0	边沿检测方式

### 描述



根据 FUN 设置，进行开关量的边沿检测。

FUN=0 时，该指令检测输入信号 B 的上升沿，在检测到上升沿的周期中，输出 Y 等于 1，否则 Y 等于 0；

FUN=1 时，该指令检测输入信号 B 的下降沿；

FUN=2 时，该指令检测输入信号 B 的上升或下降沿（变化）。

## 5.4 CTU 递增计数器

### 名称

标记：CTU

名称：递增计数器

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	0	输入使能
输出	EOUT	布尔	0	输出使能
输入	SP	整数	0	计数目标值
输入	RST	布尔	0	计数复位
输入	PRE	整数	0	计数起始值
输入	OC	整数	0	到达目标后计数器的行为
输出	CT	整数	0	计数输出
输出	DN	布尔	0	计数值到标志
输出	OV	布尔	0	计数溢出标志

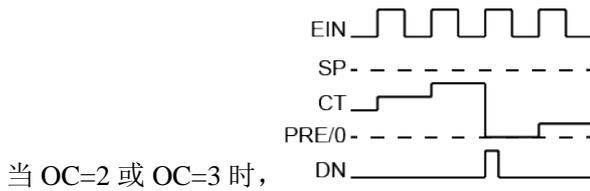
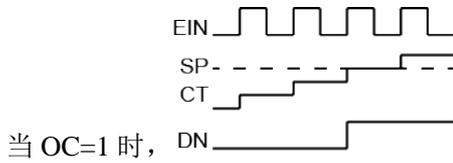
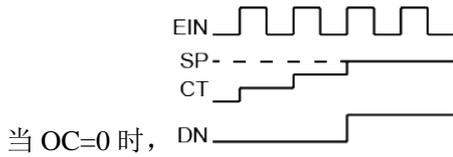
### 描述

递增计数器从计数的起始值 PRE 开始进行计数。

当 EIN 出现一个上升沿，则计数输出  $CT_{n+1} = CT_n + 1$ 。

当 CT 等于 SP 时，输出 DN=1。此后，根据 OC 的定义决定计数器的行为。

符号	作用	数值标记
冻结	CT 值不再发生变化，除非计数值被复位。	0
继续	CT 继续进行计数，直到到达最大的整数 4294967296，除非计数值被复位，在此期间 DN=1 不变。	1
循环 1	CT 从 PRE 开始重新计数，DN 在下一计算周期复位到 0。	2
循环 2	CT 从 0 开始重新计数，DN 在下一计算周期复位到 0。	3



当复位输入 RST=1 时，CT 输出被置为计数起始值 PRE，OV=0。若此时  $PRE \geq SP$ ，则 DN 立刻被置为 1，否则被置为 0。

在 OC=1 的情况下，计数达到整数的最大值，则计数溢出标志被置位 OV=1，CT 保持最大值不变。

## 5.5 CNT 计数器

### 名称

标记：CNT

名称：计数器

### 结构

作用	标记	类型	缺省值	说明
输入	UIN	布尔	0	脉冲增输入端
输入	DIN	布尔	0	脉冲减输入端
输入	SPU	整数	0	计数上目标值
输入	SPD	整数	0	计数下目标值
输入	RST	布尔	0	计数复位
输入	PRE	整数	0	计数起始值
输入	OC	整数	0	到达目标后计数器的行为
输出	CT	整数	0	计数输出
输出	UDN	布尔	0	计数值上限到标志
输出	DDN	布尔	0	计数值下限到标志
输出	UPH	布尔	0	上升过程标志



输出	DPH	布尔	0	下降过程标志
输出	OV	布尔	0	计数溢出标志

## 描述

置位、置数、溢出等功能与 CTU、CTD 功能类似。

目标计数值具有上限和下限两个数值。

若 UIN 和 DIN 同时出现上升沿，CT 保持不变，UPH 和 DPH 变为 0。

当 UIN 出现上升沿，DIN 没有上升沿时， $CT_{n+1} = CT_n + 1$ ，UPH=1；

当 DIN 出现上升沿，UIN 没有上升沿时， $CT_{n+1} = CT_n - 1$ ，DPH=1。

当 UDN=1 或 DDN=1 后，根据 OC 的设置确定计数器接下来的行为。

符号	作用	数值标记
冻结	CT、UDN、DDN 不再发生变化，除非计数值被复位。	0
继续	CT 继续进行计数，UDN 和 DDN 根据 CT 的变化而变化。	1
循环 1	CT 从 PRE 开始重新计数，UDN 和 DDN 在下一计算周期根据 CT 的值发生变化。	2
循环 2	若 UDN=1，下一周期 CT 从 0 开始计数；若 DDN=1，下一周期 CT 从最大整数开始计数；UDN 和 DDN 在下一计算周期根据 CT 的值发生变化。	3

## 5.6 DSFT 开关量选择

### 名称

标记：DSFT

名称：开关量选择

### 结构

作用	标记	类型	缺省值	说明
输入	Z1	布尔	0	输入信号 1
输入	Z2	布尔	0	输入信号 2
输入	S	布尔	0	选择信号
输出	D	布尔	0	选择输出



## 描述

该指令根据选择信号 S 的值，选择信号 Z1 或信号 Z2 进行输出。

当 S=0 时，输出 D 与信号 Z1 相同；

当 S=1 时，输出 D 与信号 Z2 相同。

## 5.7 MDIG 组合开关量逻辑

### 名称

标记：MDIG

名称：组合开关量逻辑

### 结构

作用	标记	类型	缺省值	说明
输入	Z1	布尔	0	输入信号 1
输入	Z2	布尔	0	输入信号 2
输入	Z3	布尔	0	输入信号 3
输入	Z4	布尔	0	输入信号 4
输入	Z5	布尔	0	输入信号 5
输入	Z6	布尔	0	输入信号 6
输入	Z7	布尔	0	输入信号 7
输入	Z8	布尔	0	输入信号 8
输入	Z9	布尔	0	输入信号 9
输入	Z10	布尔	0	输入信号 10
输入	Z11	布尔	0	输入信号 11
输入	Z12	布尔	0	输入信号 12
输入	Z13	布尔	0	输入信号 13
输入	Z14	布尔	0	输入信号 14
输入	Z15	布尔	0	输入信号 15
输入	Z16	布尔	0	输入信号 16
输入	RST	布尔	0	输出信号复位
输入	NUM	整数	1	信号判断阈值
输出	D	布尔	0	信号输出
输出	Y	整数	0	1 信号输入计数
输出	DH	布尔	0	信号输出保持
输出	FST	整数	0	最先输出端
输出	LST	整数	0	最后输出端



## 描述

该指令对 16 个布尔变量进行计数处理，输出布尔量和整形量。

根据信号判断阈值 NUM，当 16 个输入信号中有大于等于 NUM 个 1 信号，则输出 D 变为 1，否则输出 D 保持为 0。

Y 等于输入信号中，数值为 1 的信号个数。

DH 为输出保持，只要 D 曾经输出过 1，DH 就保持为 1，直到 RST 置位。

FST 表示 16 个输入信号中，自上次 RST 后，第一个变为 1 的输入信号，此后无论输入信号如何变化，该输出一直保持不变，直到下次 RST 后被清为 0。

LST 表示 16 个输入信号中，自上次 RST 后，最后一个满足 NUM 的信号，此后无论输入信号如何变化，该输出保持不变，直到下次 RST 后被清 0。

RST 为 1 时，DH、FST、LST 被复位。

## 应用

MDIG 具有很强的组合逻辑，可以作为多信号的逻辑或、多信号逻辑与、信号计数、首出、末出、多信号 RS 触发器等功能。

逻辑或的实现：将 NUM 设为 1，就能实现逻辑或功能。此时任何输入变为 1，输出 D 变为 1，所有输入信号全部为 0，输出 D 变为 0。

逻辑与的实现：根据需要使用的信号个数，假定需要判断 n 个输入信号，则将 NUM 设为 n，当所有输入信号全部为 1 时，输出 D 变为 1，任何输入信号为 0，输出 D 变为 0。

信号计数的实现：例如有三台电机运行，只要至少 2 台电机处于运行状态，系统判断正常，此时可以将 NUM 设为 2，三个输入信号分别连接 3 台电机的运行信号，这样输出 D 就能够表示系统正常状态。

首出判断：将 NUM 设为大于 0 的数字，此时任何输入信号变为 1，输出 FST 上的整数就表示最先变为 1 的信号端口号，该功能有时可应用在故障的原因判断中。

RS 触发器：利用 DH 输出，可以组成多信号置位，单信号复位的 RS 触发器。将 NUM 置位 1，此时任何输入信号变为 1，输出 DH 都为变为 1，此后即使输入端变为 0，输出 DH 仍能够保持，直到 RST 输入 1 进行复位。

## 5.8 SQUR 方波发生器

### 名称

标记：SQUR

名称：方波发生器



## 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	MODE	整数	0	脉冲模式
输入	TH1	整数	500	脉冲 A 序列的 1 信号输出时间，单位毫秒
输入	TP1	整数	1000	脉冲 A 序列的输出周期，单位毫秒
输入	CNT	整数	0	脉冲个数
输入	PE2	布尔	0	脉冲 B 序列使能信号
输入	TH2	整数	200	脉冲 B 序列的 1 信号输出时间，单位毫秒
输入	TP2	整数	1000	脉冲 B 序列的输出周期，单位毫秒
输出	Q	布尔	0	信号输出
输出	TRM	整数	0	当前脉冲输出的剩余时间，单位毫秒

## 描述

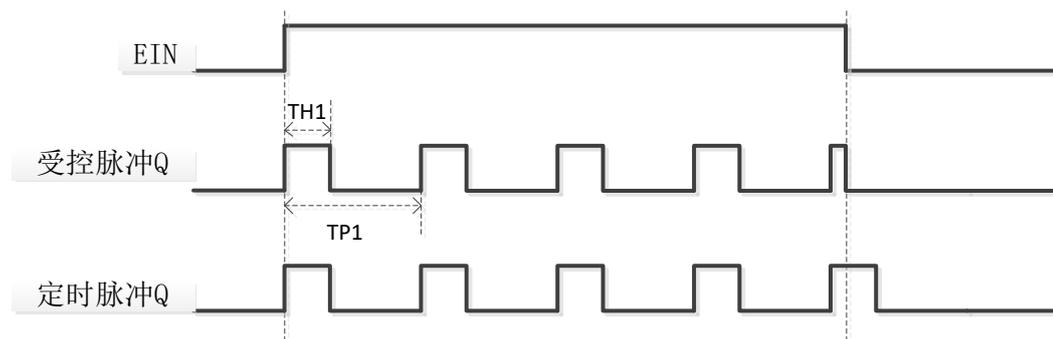
该指令按照用户定义的脉冲模式，产生各种脉冲（方波）信号。

EIN 控制指令是否计算，当 EIN=0 时，输出 Q=0；当 EIN=1 时，按用户设置的参数输出。

MODE 控制脉冲的输出方式：

MODE=0，产生受控脉冲，当 EIN 从 1 变成 0 时，如果输出 Q 等于 1，则立刻变为 0；

MODE=1，产生定时脉冲，当 EIN 从 1 变成 0 时，如果输出 Q 等于 1，则等到该脉冲完整输出后再变为 0。



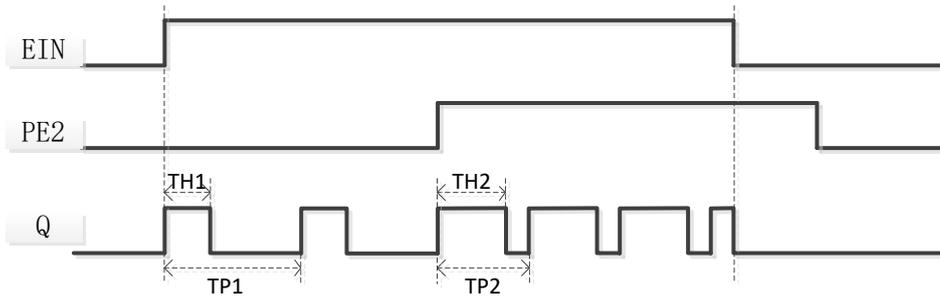
CNT 设置每次 EIN 变为 1 后，输出 Q 产生的脉冲个数。输出的脉冲到达设置数后，如果 EIN 保持 1 不变，则不再输出脉冲，直到 EIN 重新从 0 变为 1；

如果 CNT=0，则输出连续脉冲，直到 EIN=0。

PE2 设置是否启用脉冲 B 序列的参数，如果 PE2=0，则按照 TH1 和 TP1 的设置输出脉



冲 A 序列，如果 PE2=1，则按照 TH2 和 TP2 的设置输出脉冲 B 序列。



该指令集合了原先版本中 PULSE、XPULSE、SQUA、SPO 四个指令的功能，并增加了控制脉冲输出个数的功能。

## 5.9 SGN 信号发生器

### 名称

标记：SGN

名称：信号发生器

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	RST	布尔	0	复位信号
输入	MODE	整数	0	信号类型
输入	TM	整数	1000	以毫秒为单位的信号周期
输入	H	实数	100	信号上限
输入	L	实数	0	信号下限, L<H
输出	Y	实数	0	信号输出

### 描述

根据 MODE 的设置，输出 OUT 产生各种信号。

$$\text{MODE}=0, \text{ 产生等距方波, } Y = \begin{cases} L & t < \frac{TM}{2} \\ H & t \geq \frac{TM}{2} \end{cases};$$



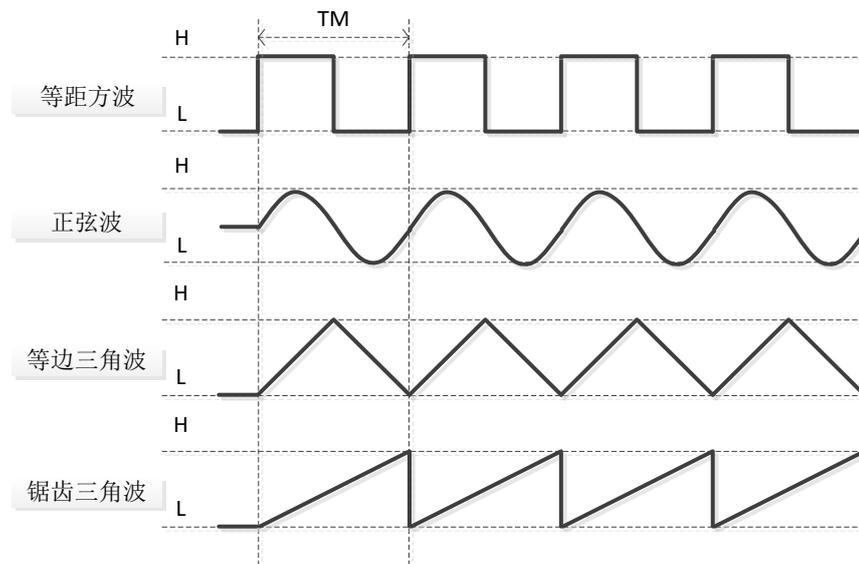
MODE=1, 产生正弦波,  $Y = \frac{H-L}{2} \sin \frac{2\pi}{TM} t + \frac{H+L}{2}$ ;

MODE=2, 产生等边三角波,  $Y = H - (H-L) \left| \frac{2t-TM}{TM} \right|$ ;

MODE=3, 产生锯齿三角波。  $Y = L + (H-L) \frac{t}{TM}$

信号波的周期由输入 TM 设置。

信号波的上下限由 L 和 H 设置。



当 EIN 输入等于 0 时, 停止信号发生, 输出 Y 维持在最后输出不变。直到 EIN 重新变为 1 后, 输出 Y 继续变化。

RST 复位信号用于重新开始信号计时。当 RST 等于 1 时, 输出 Y 等于 0, 计时清零。当 RST 从 1 变为 0 后, 输出 Y 开始按信号模式 MODE 产生信号波形的数值。

## 5.10 PSUM 脉冲量累加器

### 名称

标记: PSUM

名称: 脉冲量累加器

### 结构

作用	标记	类型	缺省值	说明
----	----	----	-----	----



输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	RST	布尔	0	累加值复位
输入	ISUM	整形	0	增脉冲量
输入	DSUM	整形	0	减脉冲量
输入	HLM	实数	100000	累计值上限
输入	LLM	实数	-100000	累计值下限
输出	Y	实数	0	累计值
输出	QHM	布尔	0	达到累计值上限
输出	QLM	布尔	0	达到累计值下限

## 描述

对脉冲量（整形值）进行累加，该指令有增脉冲和减脉冲两个输入。

$$\text{最终输出: } Y = \sum (I_{SUM} - D_{SUM}),$$

当 EIN 为 1 时，指令进行数值累加，当 EIN 为 0 时，停止累加操作，输出 Y 保持不变。

ISUM 和 DSUM 的输入只在发生变化的那个周期有效，此后如果输入的数值不变，指令不会重复对该数值进行累加。

当 Y 到达累计值上限 HLM 时，输出 QHM 被置为 1，否则 QHM 等于 0；

当 Y 到达累计值下限 LLM 时，输出 QLM 被置为 1，否则 QLM 等于 0。

当累积值复位 RST=1 时，输出 Y、QHM、QLM 被复位

## 5.11 LST 16 段线性转换

### 名称

标记：LST

名称：16 段线性转换

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X	实数	0	信号输入
输入	A0	实数	0	分段点 0 输入值
输入	B0	实数	0	分段点 0 转换值



作用	标记	类型	缺省值	说明
输入	A1	实数	1	分段点 1 输入值
输入	B1	实数	1	分段点 1 转换值
输入	A2	实数	0	分段点 2 输入值
输入	B2	实数	0	分段点 2 转换值
输入	A3	实数	0	分段点 3 输入值
输入	B3	实数	0	分段点 3 转换值
输入	A4	实数	0	分段点 4 输入值
输入	B4	实数	0	分段点 4 转换值
输入	A5	实数	0	分段点 5 输入值
输入	B5	实数	0	分段点 5 转换值
输入	A6	实数	0	分段点 6 输入值
输入	B6	实数	0	分段点 6 转换值
输入	A7	实数	0	分段点 7 输入值
输入	B7	实数	0	分段点 7 转换值
输入	A8	实数	0	分段点 8 输入值
输入	B8	实数	0	分段点 8 转换值
输入	A9	实数	0	分段点 9 输入值
输入	B9	实数	0	分段点 9 转换值
输入	A10	实数	0	分段点 10 输入值
输入	B10	实数	0	分段点 10 转换值
输入	A11	实数	0	分段点 11 输入值
输入	B11	实数	0	分段点 11 转换值
输入	A12	实数	0	分段点 12 输入值
输入	B12	实数	0	分段点 12 转换值
输入	A13	实数	0	分段点 13 输入值
输入	B13	实数	0	分段点 13 转换值
输入	A14	实数	0	分段点 14 输入值
输入	B14	实数	0	分段点 14 转换值
输入	A15	实数	0	分段点 15 输入值
输入	B15	实数	0	分段点 15 转换值
输入	A16	实数	0	分段点 16 输入值
输入	B16	实数	0	分段点 16 转换值
输出	Y	实数	0	转换输出
输出	QHM	布尔	0	越分段上限
输出	QLM	布尔	0	越分段下限
输出	SY	整数	0	当前分段号
输出	PA0	实数	0	当前分段的 X0
输出	PA1	实数	0	当前分段的 X1
输出	PB0	实数	0	当前分段的 Y0
输出	PB1	实数	0	当前分段的 Y1



## 描述

通过预先定义的分段线性转换表，将输入  $X$  转换成对应的  $Y$ 。

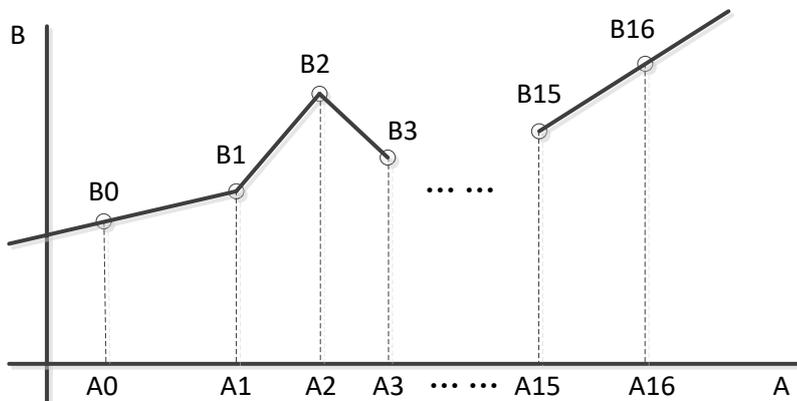
分段点为  $(A_0, B_0)$ ,  $(A_1, B_1)$ ,  $(A_2, B_2)$ , ...,  $(A_{16}, B_{16})$  共 17 点，构成 16 段折线。其中必须满足  $A_0 < A_1 < A_2 < \dots < A_{16}$ ，如果  $A_n$  的数值不满足这个不等式，即  $A_{n-1} \geq A_n$ ，则从  $A_n$  开始及以后的分段点全部无效。

对于需要定义的分段数少于 16 段的情况，例如需要设 5 段线性转换，可将  $A_6$  设为任何小于等于  $A_5$  的数即可。

分段数必须大于等于 1，即必须满足  $A_1 > A_0$ ，否则输出  $Y$  置为 0，坏点标记被置位。

假设分段数等于  $M$  ( $1 \leq M \leq 16$ )，则输出

$$Y = \begin{cases} \frac{X - A_0}{A_1 - A_0} (B_1 - B_0) + B_0 & , X < A_0 \\ \frac{X - A_{n-1}}{A_n - A_{n-1}} (B_n - B_{n-1}) + B_{n-1} & , A_{n-1} \leq X < A_n, 1 \leq n \leq M \\ \frac{X - A_{M-1}}{A_M - A_{M-1}} (B_M - B_{M-1}) + B_{M-1} & , X > A_M \end{cases}$$



当  $X < A_0$  时，输出 QLM 被置为 1，否则等于 0；

当  $X > A_M$  时，输出 QHM 被置为 1，否则等于 0。

SY 输出从 0 开始计数，0 表示当前计算范围在  $A_0 \sim A_1$  内，1 表示计算范围在  $A_1 \sim A_2$  内，以此类推。

PA0、PA1、PB0、PB1 用于输出当前线性段的四个线性转换参数。



## 5.12 FST 16 段时间信号转换

### 名称

标记: FST

名称: 16 段时间信号转换

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	SX	整数	0	起点段选择, 从 0 开始计数, 且小于 16
输入	HOLD	布尔	0	保持信号
输入	Y0	实数	0	起点数值
输入	T1	实数	1	节点 1 时间, 秒为单位, 下同
输入	Y1	实数	0	节点 1 数值
输入	T2	实数	-1	节点 2 时间
输入	Y2	实数	0	节点 2 数值
输入	T3	实数	-1	节点 3 时间
输入	Y3	实数	0	节点 3 数值
输入	T4	实数	-1	节点 4 时间
输入	Y4	实数	0	节点 4 数值
输入	T5	实数	-1	节点 5 时间
输入	Y5	实数	0	节点 5 数值
输入	T6	实数	-1	节点 6 时间
输入	Y6	实数	0	节点 6 数值
输入	T7	实数	-1	节点 7 时间
输入	Y7	实数	0	节点 7 数值
输入	T8	实数	-1	节点 8 时间
输入	Y8	实数	0	节点 8 数值
输入	T9	实数	-1	节点 9 时间
输入	Y9	实数	0	节点 9 数值
输入	T10	实数	-1	节点 10 时间
输入	Y10	实数	0	节点 10 数值
输入	T11	实数	-1	节点 11 时间
输入	Y11	实数	0	节点 11 数值
输入	T12	实数	-1	节点 12 时间
输入	Y12	实数	0	节点 12 数值
输入	T13	实数	-1	节点 13 时间
输入	Y13	实数	0	节点 13 数值



作用	标记	类型	缺省值	说明
输入	T14	实数	-1	节点 14 时间
输入	Y14	实数	0	节点 14 数值
输入	T15	实数	-1	节点 15 时间
输入	Y15	实数	0	节点 15 数值
输入	T16	实数	-1	节点 16 时间
输入	Y16	实数	0	节点 16 数值
输出	Y	实数	0	信号输出
输出	SY	整数	0	当前时段，从 SX 开始计数
输出	SD	布尔	0	到达当前段终点指示，单脉冲
输出	D	布尔	0	到达曲线终点指示，长脉冲
输出	TD	整数	0	当前时段运行时间，单位：秒
输出	TS	整数	0	总运行时间，单位：秒
输出	TACC	整数	0	当前时段累计时间，单位毫秒
输出	PY0	实数	0	当前分段 Y0
输出	PY1	实数	0	当前分段 Y1
输出	TRM	整数	0	当前分段运行剩余时间，单位：秒

## 描述

该指令根据设置的 T-Y 中的起点及 16 个节点，输出所需的曲线。

分段时间点要求满足  $T_n > 0$ ，否则信号变化到  $T_n$  节点即作为信号变化终点。

当输入使能 EIN=0 时，输出 SY=SX，SD=0，D=0，Y=Y<sub>sx</sub>，TD=0，TS=0；

当输入使能 EIN=1 时，指令开始按照设置的 (T, Y) 节点，从 SX 设置的起点开始匀速变化输出。

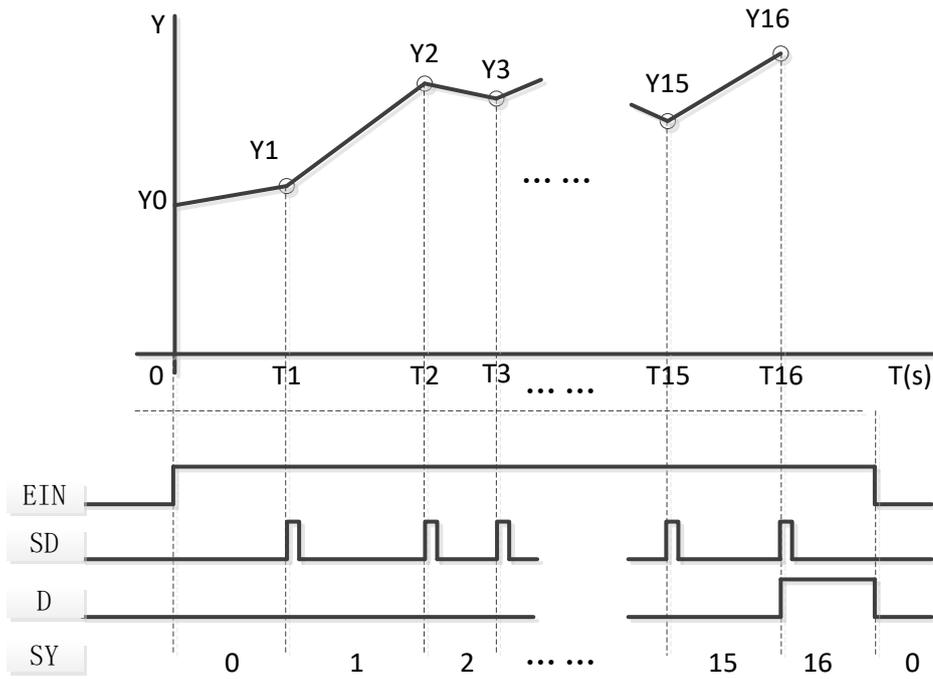
假设 EIN 在  $t_0$  时刻开始变为 1，经过  $T_x$  时间后处于第 n 个时段 ( $n \geq 0$ )，此时

$$T' < T_x \leq T'' ,$$

$$\text{其中 } T' = \begin{cases} 0 & , n = 0 \\ \sum_{a=SX+1}^{SX+n} T_a & , n > 0 \end{cases}$$

$$T'' = T' + T_{SX+n+1}$$

$$\text{输出 } Y = \frac{T_x - T'}{T_{SX+n+1}} (Y_n - Y_{n-1}) + Y_{n-1}$$



当前时段  $SY = SX + n$ ，当前段时间  $TD = T_x - T'$ ，总运行时间  $TS = T_x$

第  $n$  段时间到达后，SD 输出一个周期的 1 信号；同时 SY 增加 1。

在变化过程中，如果保持信号 HOLD=1，则变化过程被中断（计时被暂停，输出 Y 保持不变），直到 HOLD=0 后，从中断处继续开始变化。

当经过时间  $T_x$  到达变化终点（下段时间  $T_n < 0$ ）后，信号 D 变为 1，直到 EIN 变为 0。

参数 PY0 和 PY1 输出当前分段的起始值和终止值。

参数 TRM 表示当前分段剩余时间。

### 5.13 SWF 多路实数开关

#### 名称

标记：SWF

名称：多路实数开关

#### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能



输入	X1	实数	0	输入信号 1
输入	X2	实数	0	输入信号 2
输入	X3	实数	0	输入信号 3
输入	X4	实数	0	输入信号 4
输入	X5	实数	0	输入信号 5
输入	X6	实数	0	输入信号 6
输入	X7	实数	0	输入信号 7
输入	X8	实数	0	输入信号 8
输入	X9	实数	0	输入信号 9
输入	X10	实数	0	输入信号 10
输入	X11	实数	0	输入信号 11
输入	X12	实数	0	输入信号 12
输入	X13	实数	0	输入信号 13
输入	X14	实数	0	输入信号 14
输入	X15	实数	0	输入信号 15
输入	X16	实数	0	输入信号 16
输入	SEL	整数	0	选择开关
输出	Y	实数	0	选择输出

## 描述

多路实数开关用来从 16 路输入中选择一路信号输出。

选择开关 SEL 是一个 0~16 的整数。

当 SEL=0 时，不选择任何输入，输出 Y=0；

当 SEL=n,  $1 \leq n \leq 16$  时,  $Y=X_n$ 。

## 5.14 SWW 多路整数开关

### 名称

标记: SWW

名称: 多路整数开关

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X1	整数	0	输入信号 1



输入	X2	整数	0	输入信号 2
输入	X3	整数	0	输入信号 3
输入	X4	整数	0	输入信号 4
输入	X5	整数	0	输入信号 5
输入	X6	整数	0	输入信号 6
输入	X7	整数	0	输入信号 7
输入	X8	整数	0	输入信号 8
输入	X9	整数	0	输入信号 9
输入	X10	整数	0	输入信号 10
输入	X11	整数	0	输入信号 11
输入	X12	整数	0	输入信号 12
输入	X13	整数	0	输入信号 13
输入	X14	整数	0	输入信号 14
输入	X15	整数	0	输入信号 15
输入	X16	整数	0	输入信号 16
输入	SEL	整数	0	选择开关
输出	Y	整数	0	选择输出

## 描述

多路整数开关用来从 16 路输入中选择一路信号输出。

选择开关 SEL 是一个 0~16 的整数。

当 SEL=0 时，不选择任何输入，输出 Y=0；

当 SEL=n， $1 \leq n \leq 16$  时， $Y=X_n$ 。

## 5.15 TMC 时间处理模块

### 名称

标记：TMC

名称：时间处理模块

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	MODE	整数	0	处理模式
输入	YEAX	整数	0	日历输入 - 年



输入	MONX	整数	0	日历输入 - 月
输入	DAYX	整数	0	日历输入 - 日
输入	HOUX	整数	0	日历输入 - 时
输入	MINX	整数	0	日历输入 - 分
输入	SECX	整数	0	日历输入 - 秒
输入	START	布尔	0	启动计时
输入	PAUSE	布尔	0	暂停计时
输入	RST	布尔	0	计时复位标志
输出	DN	布尔	0	时间到标志
输出	TT	布尔	0	计时标志
输出	YEA1	整数	0	日历输出 1 - 年
输出	MON1	整数	0	日历输出 1 - 月
输出	DAY1	整数	0	日历输出 1 - 日
输出	HOU1	整数	0	日历输出 1 - 时
输出	MIN1	整数	0	日历输出 1 - 分
输出	SEC1	整数	0	日历输出 1 - 秒
输出	YEA2	整数	0	日历输出 2 - 年
输出	MON2	整数	0	日历输出 2 - 月
输出	DAY2	整数	0	日历输出 2 - 日
输出	HOU2	整数	0	日历输出 2 - 时
输出	MIN2	整数	0	日历输出 2 - 分
输出	SEC2	整数	0	日历输出 2 - 秒

## 描述

TMC 时间处理模块用来处理各种与日历钟（Wall Clock）相关的数据。

MODE 用来选择处理的方式：

MODE=0，日历输出，在日历输出 1 信号组中，输出控制器的日历时间。日历输入信号被忽略。

MODE=1，计时器（秒），利用 START、PAUSE 和 RST 构成秒计时器，计时时间按秒累计，在日历输出 1 组的秒信号 SEC1 输出。当 START 出现上升沿后，计时时间被复位并开始计时，计时标志 TT=1；当 START 出现下降沿后，计时过程结束，计时时间保持不变，计时标志 TT=0，当 START 下次变为 1，或 RST=1 时，SEC1 的数值被复位；当 PAUSE=1 时，计时过程被暂停，TT 和 SEC1 信号保持不变，当 PAUSE=0 后，计时过程继续进行，SEC1 继续按秒累积。当 RST=1 时，SEC1 的数值被清零。

MODE=2，计时器（时钟），利用 START、PAUSE 和 RST 构成时钟计时器，计时的时间被转换为日、时、分、秒从日历输出 1 组信号输出。START、PAUSE 和 RST 的功能参考 MODE=1。

MODE=3，计时器（日历），利用 START、PAUSE 构成日历计时器，START 出现上升沿后，在日历输出 1 组输出启动时刻的日历，在日历输出 2 组输出当前时刻的日历，计时标志 TT=1；当 START 出现下降沿后，计时过程结束，日历输出 2 组保持在最后的时刻，计



时标志 TT=0；当 PAUSE=1 时，计时过程被暂停，日历输出 2 组保持在暂停时刻的时间，TT 保持不变，当 PAUSE=0 后，计时过程继续进行，日历输出 2 继续输出当前时刻的日历。RST 信号没有作用。

MODE=4，定时器（秒），利用 START、PAUSE 和 RST 构成秒定时器，定时时间在日历输入的 SECX 设置，计时时间按秒累计，在日历输出 1 组的秒信号 SEC1 输出。当 START 出现上升沿后，计时时间被复位并开始计时，计时标志 TT=1；当 START 出现下降沿后，计时过程结束，计时时间保持不变，计时标志 TT=0，当 START 下次变为 1，或 RST=1 时，SEC1 的数值被复位；当 PAUSE=1 时，计时过程被暂停，TT 和 SEC1 信号保持不变，当 PAUSE=0 后，计时过程继续进行，SEC1 继续按秒累积。当 RST=1 时，SEC1 的数值被清零。当计时时间达到定时时间后，计时过程结束，计时标志 TT=0，时间到标志 DN=1。

MODE=5，定时器（时钟），利用 START、PAUSE 和 RST 构成时钟定时器，定时时间在日历输入的 6 个输入设置，计时的时间被转换为年、月、日、时、分、秒从日历输出 1 组信号输出。START、PAUSE 和 RST 的功能和 DN、TT 的信号输出参考 MODE=4。

MODE=6，相对秒值转换，将日历输入的“年月日时分秒”信号，转换成从 1970 年 1 月 1 日 0 时 0 分 0 秒开始到输入日期所经过的秒数，结果在 SEC1 上输出。如果输入的日期在 1970 年 1 月 1 日之前，或在 2038 年 1 月 18 日之后，则无法进行转换，所有输出被清零，输出的坏点标志被置位。

MODE=7，日历转换，将 SECX 设成从 1970 年 1 月 1 日 0 时 0 分 0 秒开始所经过的秒数，转换成按“年月日时分秒”表示的日历，在日历输出 1 组信号中进行显示。

MODE=8，绝对秒值转换，将 SECX 设成秒数，该秒值对应的“日时分秒”在日历输出 1 组信号中输出显示。



## 第6章 设备操作

### 介绍

该类算法主要是涉及到设备操作与控制相关的指令,可以利用指令构成特定的表达式或实现某种算法。

该类指令中含有 EIN 和 EOUT 引脚的指令,如果没有特殊说明的,当 EIN=1 时,指令进行正常计算;当 EIN=0 时,指令的计算停止,输出保持在最后的计算结果。EOUT 的值与 EIN 始终保持一致。

本类指令不具有品质传递功能。

该类指令包括:

序号	标记	名称
1	DMA	数字软手操
2	SMA	模拟软手操
3	ILK4	4 键互锁开关
4	TPOS	两位位置控制
5	DEV1	单向设备控制
6	DEV2	双向设备控制
7	STEP	步序控制
8	SSA	数字驱动伺服模块
9	KBML	键盘模拟量增减

### 6.1 DMA 数字软手操

#### 名称

标记: DMA

名称: 数字软手操

#### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	TR	布尔	0	跟踪点



输入	TS	布尔	0	跟踪允许
输入	CMD	整数	0	控制命令
输出	D	布尔	0	输出变量

## 描述

本指令输出一个可被操作的布尔变量。

在跟踪标志 TS=0 时，通过 CMD 参数给出“置位”、“复位”、“反相”、“脉冲”等命令，使得输出 D 根据命令发生变化。

CMD 使用的命令如下：

符号	作用	数值标记
空命令	此时功能块不执行任何命令，D 保持不变；	0
复位	输出 D 被置为 0；	1
置位	输出 D 被置为 1；	2
反相	输出 D 的状态发生改变，如果原先 D 输出状态为 1，则 D 输出变为 0；如果 D 输出状态为 0，则 D 输出变为 1；	3
脉冲	输出 D 产生为计算周期的脉冲；	4

CMD 命令只有在变化的那个周期有效，例如，如果需要再次发送“反相”命令，则 CMD 在置为“反相”命令后，必须先设成 0，然后再设为 4。

在跟踪标志 TS=1 时，不接受操作指令 CMD，D 输出跟踪点 TR 的值，其品质也随 TR。

## 6.2 SMA 模拟软手操

### 名称

标记：SMA

名称：模拟软手操

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X	实数	0	输入变量
输入	TR	实数	0	跟踪量
输入	EV	实数	0	超驰量输入
输入	DMS	实数	0	手动设定值修改量
输入	DSP	实数	0	调节设定值修改量



输入	PV	实数	0	被控调节量
输入	TOL	布尔	0	就地开关
输入	TOE	布尔	0	超驰开关
输入	TOM	布尔	0	手动开关
输入	TOA	布尔	0	自动输出开关
输入	SFT	整数	0	数值保持模式
输入	SPH	实数	100	SP 输出上限
输入	SPL	实数	0	SP 输出下限
输入	YH	实数	100	输出上限
输入	YL	实数	0	输出下限
输入	YRU	实数	0	输出上升速率限制
输入	YRD	实数	0	输出下降速率限制
输入	YB	实数	0	输出偏置量
输出	Y	实数	0	输出变量
输出	QHM	布尔	0	输出达到上限
输出	QLM	布尔	0	输出达到下限
输出	MS	实数	0	手动设定值
输出	SP	实数	0	调节设定值
输出	ST	整数	0	控制状态

## 描述

该指令实现回路的软手操。手操器具有禁操、就地、超驰、手动、自动五种工作模式，模式的优先级别是禁操>就地>超驰>手动>自动。

当 EIN=0 时，处于禁操模式，此时除 EIN 信号外，指令不接收任何信号输入，也不输出任何变化。EIN=1 时，可进入其他模式。

当就地开关 TOL=0，超驰开关 TOE=1 时，手操器处于超驰工作模式，此时模式开关 STF=2，输出 Y 等于超驰输入量 EV。

当 TOL=0，TOE=0，TOM=1 时，手操器处于手动工作模式，此时模式开关 ST=0，输出 Y 等于手动设定值 DMS。

当 TOL=0，TOE=0，TOM=0，TOA=1 时，手操器处于自动工作模式，此时模式开关 ST=1，输出 Y 等于输入 X 加上输出偏置 YB。

当四个模式开关 TOL、TOE、TOM、TOA 全部是 0 的时候，若此时 ST=1（自动模式），则继续保持在 ST=1，若 ST 是其他数值的，则 ST=0（进入手动模式）。

当就地开关 TOL=1 时，手操器处于就地工作模式，此时模式开关 ST=4，输出 Y 等于就地反馈量 TR。

在工作模式之间切换时，可以使用 SFT 参数选择是否保持调节设定值 SP 和手动值 MS 的自动跟踪。

SFT=0 时，SP 和 MS 的数值不跟踪，仅根据 DSP 和 DMS 发生变化。



SFT=1 时，SP 的数值在自动模式下根据 DSP 变化，在其他模式下等于输入 PV；MS 的数值不跟踪，仅根据 DMS 变化。

SFT=2 时，SP 的数值不跟踪，仅根据 DSP 变化，MS 的数值在手动模式下根据 DMS 变化，在其他模式下等于输出 Y。

SFT=3 时，SP 的数值在自动模式下根据 DSP 变化，在其他模式下等于输入 PV；MS 的数值在手动模式下根据 DMS 变化，在其他模式下等于输出 Y。

DMS 用来修改手动设定值 MS。 $MS = MS + DMS$ ，每次 DMS 的数值发生变化后指令修改 MS 数值，然后自动将 DMS 置成 0。

DSP 用来修改调节设定值 SP。 $SP = SP + DSP$ ，每次 DSP 的数值发生变化后指令修改 SP 数值，然后自动 DSP 置成 0。

SPH 和 SPL 为输出 SP 的上下限，SP 输出保持在上下限值上；

YRU 和 YRD 限制输出 SP 每秒的变化速率。YRU 和 YRD 都应设置为大于或等于 0 的实数。如果设定值等于 0，表示没有上升（或下降）的速率限制。

YH 和 YL 为输出 Y 的上线限，当 Y 到达设定的上下限度时，QHM 或 QLM 会被置位，Y 输出保持在上下限值上。

## 应用

SMA 一般应用在两种场景之下：

- 1) 作为对单个设备手动调节使用。这类设备只需要在画面上手动调整开度，无需自动控制功能。在这种场景下，可以将本指令看成是具有幅度、速率限制的模拟量操作器；
- 2) 配合 PID 控制指令（PID，注意不是 EPID）进行指令输出。在这种场景下，操作器具有在各种模式（手动、自动、就地等）间无扰切换的功能。

## 6.3 ILK4 4 键互锁开关

### 名称

标记：ILK4

名称：4 键互锁开关

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能



输入	KEY1	布尔	0	开关 1
输入	KEY2	布尔	0	开关 2
输入	KEY3	布尔	0	开关 3
输入	KEY4	布尔	0	开关 4
输入	RST	布尔	0	输出复位
输入	MODE	整数	0	优先选择模式
输出	Q1	布尔	0	信号输出 1
输出	Q2	布尔	0	信号输出 2
输出	Q3	布尔	0	信号输出 3
输出	Q4	布尔	0	信号输出 4

## 描述

这是一个有相互锁功能的开关量输出模块，在四个输出端的输出信号中，只能有一个信号被选择输出，当该信号输出时，会自动屏蔽其他输出端。

当复位 RST=1 时，所有输出等于 0，且不接受任何 KEY 的输入。

MODE=0 时，KEY 的上升沿有效，若同时有两个或以上 KEY 信号变化，输入信号优先级为 KEY1>KEY2>KEY3>KEY4。

MODE=1 时，KEY 的 1 信号有效，若同时有两个或以上 KEY 信号等于 1，输入信号优先级为 KEY1>KEY2>KEY3>KEY4。

MODE=2 时，KEY 的 1 信号有效。若 Q1~Q4 有输出时，会闭锁其他输入，必须使用 RST 复位输出后才能选择新的 KEY 信号。如果同一周期有两个或以上 KEY 信号等于 1，则输入信号优先级为 KEY1 > KEY2 > KEY3 > KEY4。

## 6.4 TPOS 两位位置控制

### 名称

标记：TPOS

名称：两位位置控制

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	REM	布尔	0	远程/就地标志
输入	TOM	布尔	0	切手动输入
输入	TOA	布尔	0	切自动输入



输入	ENX1	布尔	1	允许 X1 输入
输入	ENX2	布尔	1	允许 X2 输入
输入	ENP	布尔	1	允许停止输入
输入	MX1	布尔	0	手动信号输入 1
输入	MX2	布尔	0	手动信号输入 2
输入	MSTP	布尔	0	手动停止信号
输入	AX1	布尔	0	自动信号输入 1
输入	AX2	布尔	0	自动信号输入 2
输入	ASTP	布尔	0	自动停止信号
输入	EX1	布尔	0	紧急信号输入 1
输入	EX2	布尔	0	紧急信号输入 2
输入	ESTP	布尔	0	紧急停止信号
输入	ERR	布尔	0	设备故障信号
输入	EMSK	整数	0	故障停机掩码
输入	RST	布尔	0	故障复位
输入	POS1	布尔	0	位置 1 到达信号
输入	POS2	布尔	0	位置 2 到达信号
输入	TM	整数	0	运行超时设置, 单位: 毫秒
输入	DFN	整数	0	驱动信号模式
输入	TP	整数	1000	脉冲的输出时间, 单位: 毫秒
输入	TF	整数	1000	正反转切换时间, 单位: 毫秒
输出	AUTO	布尔	0	手动/自动标志
输出	Y1	布尔	0	驱动信号 1
输出	Y2	布尔	0	驱动信号 2
输出	Y3	布尔	0	驱动信号 3
输出	ESTA	整数	0	故障状态

## 描述

该指令用来驱动类似电动门这类设备, 驱动电机进行正反转, 到达两个固定工作位置后停止运行的设备。

该指令具有禁操、就地、超驰、手动、自动五种工作模式。其中 EX1、EX2、ESTP 这组信号用于超驰模式; MX1、MX2、MSTP 这组信号用于手动模式; AX1、AX2、ASTP 这组信号用于自动模式。

模式的优先级是, 禁操>就地>超驰>手动>自动。

在 EIN=0 时, 指令进入禁操模式, 指令不接受任何输入信号 (EIN 除外), 也不产生输出的任何变化。EIN=1 时, 指令可进入其他模式运行。

当 REM=0 时, 指令处于就地工作模式, 超驰、手动、自动输入信号不起作用, 但各种故障信号的处理会正常进行。

当 REM=1 时, 当 EX1、EX2、ESTP 输入端有上升沿信号, 指令立刻按照超驰模式运



行，根据超驰的输入信号输出相应的超驰命令，此时故障信号对输出没有影响。

当  $REM=1$ ， $TOM=1$  时，指令处于手动模式，此时  $MX1$ 、 $MX2$ 、 $MSTP$  这组输入信号有效，输出  $AUTO=0$ 。

当  $REM=1$ ， $TOA=1$  时，指令处于自动模式，此时  $AX1$ 、 $AX2$ 、 $ASTP$  这组输入信号有效，输出  $AUTO=1$ 。

如果  $TOM$  和  $TOA$  都等于 0，则指令维持在原有模式不变。

如果  $TOM$  和  $TOA$  都等于 1，则指令进入手动模式。

$ENX1$ 、 $ENX2$ 、 $ENP$  用于分别用于允许“开 ( $X1$ )”、“关 ( $X2$ )”、“停 ( $STP$ )”的输入。这几个允许信号同时对超驰、手动、自动输入起效。

$MX1$ 、 $AX1$ 、 $EX1$  有效 (=1) 且  $ENX1=1$  时，如果  $POS1=0$ ，则发送正转指令，直到  $POS1=1$ ；

$MX2$ 、 $AX2$ 、 $EX2$  有效 (=1) 且  $ENX2=1$  时，如果  $POS2=0$ ，则发送反转指令，直到  $POS2=1$ ；

$MSTP$ 、 $ASTP$ 、 $ESTP$  有效 (=1) 且  $ENP=1$  时，取消当前  $Y1$ 、 $Y2$  的输出。

如果正转指令已经发出，或已经正转到位，此时  $ENX1$  从 1 变为 0，对指令输出没有影响。同理，反转指令已经发出，或已经反转到位， $ENX2$  变为 0 对输出没有影响；正转或反转指令已经发出， $ENP$  对输出没有影响。

$TM$  输入用来设置运行超时时间，当  $Y1$  或  $Y2$  输出驱动信号，超过  $TM$  设置的时间后， $POS1$  和  $POS2$  的反馈信号尚未到达，则指令故障状态  $ESTA$  被置超时故障。 $TM$  必须小于 3600s， $TM=0$  时，不进行超时判断。

$DFN$  用来设置输出信号的方式：

$DFN=0$  时， $Y1$  发送正转信号， $Y2$  发送反转信号， $Y3$  没有作用；

$DFN=1$  时， $Y1$  发送正转脉冲， $Y2$  发送反转脉冲， $Y3$  发送停止脉冲，脉冲的输出宽度由  $TP$  设置。在使用脉冲输出时，应确保  $TP$  和  $TF$  不能同时等于 0。

$POS1$  和  $POS2$  应分别接入两个到位状态。当收到  $POS1=1$  时，指令撤销正转命令；当收到  $POS2=1$  时，指令撤销反转命令。如果  $POS1$  和  $POS2$  同时收到 1，指令产生信号故障，反映在  $ESTA$  的输出中。

$TP$  用来设置输出脉冲的宽度，在  $DFN=1$  时该输入有效，在  $DFN=0$  时该参数被忽略。

$TF$  用来设置正反转的切换时间，这是为保护机械和电气设备，在正转（或反转）后不允许立刻反转（或正转），而是延时一个短暂的时间使机械的惯性得以释放。 $TF=0$  表示可以立刻反转（应用于要求较快响应速度且惯性很小的设备中）。

当  $ERR$  被置位时，指令故障状态  $ESTA$  被置硬件故障。

故障停机掩码  $EMSK$  用来选择需要自动停机的各种故障信号。具体使用方法见下文关于故障码的说明。

$RST$  信号用于复位故障状态。当故障发生后， $ESTA$  会保持故障码输出，即使故障源已



经消失。当 RST=1，且故障源已不存在，此时 ESTA 恢复到 0，Y1 和 Y2 输出恢复正常。RST 具有自复位功能，即指令内部会将 RST 自动恢复到 0 值。

TPOS 指令为每种可能出现的故障设置了代码，如下表所示：

代码	故障	说明
1	硬件故障	当故障输入 ERR=1 产生
4	超时故障	设备未能在指定的时间内做出相应动作。 当 Y1（或 Y2）输出后，TM 时间内，POS1（或 POS2）未收到相应反馈时产生。
8	信号错误	POS1 和 POS2 同时收到 1 信号。

ESTA 用于表示当前各种故障，代码见上表。当两种或以上故障同时发生，则 ESTA 显示这几种故障码的总和，例如硬件故障（1#）和信号错误（8#）同时发生，ESTA 就显示为 9。

当各种故障发生后，用户可以选择哪些故障用于停止设备运行。通过在 EMSK 设置代码来选择需要停机的故障，将那些停机故障的代码相加设置于 EMSK。例如，希望超时故障（4#）和信号错误（8#）发生时停止设备，但硬件故障（1#）不要停止，可以将 EMSK 设置为 12。

当引起停机的故障发生时，如果是常信号驱动(DFN=0)，指令会停止 Y1 和 Y2 的输出，如果是脉冲驱动(DFN=1)，除了取消 Y1 和 Y2 的输出，还会在 Y3 输出停止脉冲。此后只有在 RST 输入 1 信号脉冲时，指令才能重新输出 Y1~Y3 的驱动信号。

## 应用

本指令通常用于电动门（电动阀）、两位式挡板（非调节型）等具有两个到位状态的设备控制。

用户应根据设备电气柜的控制方式，选择合适的驱动方式和输出信号连接。下表列出了现场电气柜常用接口场景。

场景	电气柜	DCS 指令
1	提供“开”和“关”两个信号接口，非自保持	选择“常信号”驱动，Y1 和 Y2 分别连接“开”“关”信号
2	提供“开”和“关”两个信号接口，自保持，根据到位信号自动停止	选择“脉冲信号”驱动，Y1 和 Y2 分别连接“开”“关”信号
3	提供“开”、“关”、“停”三个信号接口，自保持	选择“脉冲信号”驱动，Y1 和 Y2 分别连接“开”“关”信号，Y3 连接“停”信号

本指令虽然支持“中停”功能（即电动门既不在“开”位，也不在“关”位时，停止设备运转），但上述三种应用场景中，只有 1、3 场景可实现该功能，这主要受电气柜信号接口所限。



## 6.5 DEV1 单向设备控制

### 名称

标记: DEV1

名称: 单向设备控制

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	REM	布尔	0	远程/就地控制标志
输入	TOM	布尔	0	切手动输入
输入	TOA	布尔	0	切自动输入
输入	ENS	布尔	1	允许启动
输入	ENP	布尔	1	允许停止
输入	MSTR	布尔	0	手动启动信号
输入	MSTP	布尔	0	手动停止信号
输入	ASTR	布尔	0	自动启动信号
输入	ASTP	布尔	0	自动停止信号
输入	ESTR	布尔	0	紧急启动信号
输入	ESTP	布尔	0	紧急停机信号
输入	ERR	布尔	0	设备故障信号
输入	EMSK	整数	65535	故障停机掩码
输入	RST	布尔	0	故障复位
输入	STA	布尔	0	设备运行状态
输入	STP	布尔	0	设备停止状态
输入	TM	整数	0	启动超时设置, 单位: 毫秒
输入	DFN	整数	0	驱动信号模式
输入	TP	整数	1	启停脉冲的输出时间, 单位: 毫秒
输出	AUTO	布尔	0	手动/自动标志
输出	Y1	布尔	0	驱动信号 1
输出	Y2	布尔	0	驱动信号 2
输出	ESTA	整数	0	故障状态

### 描述

该指令用来驱动类似单向电机这类持续运行的设备。

该指令具有禁操、就地、手动、自动、超驰五种工作模式。其中 ESTR、ESTP 这组信



号用于超驰模式；MSTR、MSTP 这组信号用于手动模式；ASTR、ASTP 这组信号用于自动模式。

模式的优先级别是，禁操>就地>超驰>手动>自动。

在 EIN=0 时，指令进入禁操模式，指令不接受任何输入信号（EIN 除外），也不产生输出的任何变化。EIN=1 时，指令可进入其他模式运行。

当 REM=0 时，设备处于就地控制状态，此时超驰、手动和自动输入信号不起作用，但各种故障信号处理会正常进行。

当 REM=1，只要 ESTR、ESTP 输入端有 1 信号，指令立刻按照超驰模式运行，根据超驰的输入信号输出相应的超驰命令。

当 REM=1，TOM=1 时，指令处于手动模式，此时手动启停信号 MSTR 和 MSTP 有效，输出 AUTO=0。

当 REM=1，TOA=1 时，指令处于自动模式，此时自动启停信号 ASTR 和 ASTP 有效，输出 AUTO=1。

如果 TOM 和 TOA 都等于 0，则指令维持在原有模式不变。

如果 TOM 和 TOA 都等于 1，则指令进入手动模式。

ENS、ENP 分别用于允许“启动”、“停止”的输入信号。这几个允许信号同时对超驰、手动、自动输入起效。

ESTR、MSTR、ASTR 有效 (=1) 且 ENS=1 时，发出启动命令；ESTP、MSTP、ASTP 有效 (=1) 且 ENP=1 时，发出停止命令。

如果启动命令已经发出，此时 ENS 从 1 变为 0，对输出没有影响，启动命令不会撤销。同理，如果停止命令已经发出（常信号下，启动命令已撤销），此时 ENP 从 1 变为 0，对输出没有影响。TM 用来控制设备启动超时，当 Y1 发出命令后开始计时，若超过 TM 设置的时间后，STA 反馈信号尚未到达，则指令故障状态 ESTA 置为超时故障。TM 必须小于 3600s，TM=0 时，不进行超时判断。

DFN 用来设置 Y1 和 Y2 的输出方式：

DFN=0 时，Y1 输出常信号，Y2 没有作用；

DFN=1 时，Y1、Y2 输出长度为 TP 的脉冲信号，Y1 作为启动脉冲，Y2 作为停止脉冲。

STA 输入信号连接设备的运转状态。

如果设备额外提供了停止状态，可以连接 STP 输入端。如果设备未提供该信号，端口保持在默认值 0 即可。

一旦接入 STP 信号，如果 STA 和 STP 同时为 1，则产生信号故障，反映在 ESTA 的输出中。

TP 用来设置启动和停止脉冲的宽度，在 DFN=1 时该输入有效，在 DFN=0 时该参数被忽略。如果设置为 0，则根据状态反馈信号 STA 来清除脉冲输出。

当 ERR 被置位时，指令故障状态 ESTA 被置硬件故障。



故障停机掩码 **EMSK** 用来选择需要自动停机的各种故障信号。具体使用方法见下文关于故障码的说明。

**RST** 信号用于复位故障状态。当故障发生后，**ESTA** 会保持故障码输出，即使故障源已经消失。当 **RST=1**，且故障源已不存在，此时 **ESTA** 恢复到 0，**Y1** 输出恢复正常。**RST** 具有自复位功能，即指令内部会将 **RST** 自动恢复到 0 值。

**DEV1** 指令为每种可能出现的故障设置了代码，如下表所示：

代码	故障	说明
1	硬件故障	当故障输入 <b>ERR=1</b> 产生
2	跳闸故障	<b>REM=1</b> 时，如果设备在运行状态下，本指令未给出停止命令，设备自行停止运行，则产生本故障
4	超时故障	设备未能在指定的时间内做出相应动作。 当 <b>Y1</b> 输出后， <b>TM</b> 时间内， <b>STA</b> 未收到相应反馈时产生。
8	信号错误	<b>STA</b> 和 <b>STP</b> 同时收到 1 信号。

**ESTA** 用于表示当前各种故障，代码见上表。当两种或以上故障同时发生，则 **ESTA** 显示这几种故障码的总和，例如硬件故障（1#）和信号错误（8#）同时发生，**ESTA** 就显示为 9。

当各种故障发生后，用户可以选择哪些故障用于停止设备运行。通过在 **EMSK** 设置代码来选择需要停机的故障，将那些停机故障的代码相加设置于 **EMSK**。例如，希望超时故障（4#）和信号错误（8#）发生时停止设备，但硬件故障（1#）不要停止，可以将 **EMSK** 设置为 12。

当引起停机的故障发生时，如果是常信号驱动（**DFN=0**），指令会停止 **Y1** 的输出，如果是脉冲驱动（**DFN=1**），除了取消 **Y1** 的输出，还会在 **Y2** 输出停止脉冲。此后只有在 **RST** 输入 1 信号脉冲时，指令才能重新输出 **Y1**、**Y2** 的驱动信号。

## 应用

本指令通常用于普通单向电机、电磁阀等持续运转的设备控制。

用户应根据设备电气柜的控制方式，选择合适的驱动方式和输出信号连接。下表列出了现场电气柜常用接口场景。

场景	电气柜	DCS 指令
1	提供“运行”信号接口，非自保持	选择“常信号”驱动， <b>Y1</b> 连接“启动”信号
2	提供“运行”和“停止”两个信号接口，自保持	选择“脉冲信号”驱动， <b>Y1</b> 和 <b>Y2</b> 分别连接“运行”“停止”信号



## 6.6 DEV2 双向设备控制

### 名称

标记: DEV2

名称: 双向设备控制

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	REM	布尔	0	远程/就地控制标志
输入	TOM	布尔	0	切手动输入
输入	TOA	布尔	0	切自动输入
输入	ENF	布尔	1	允许正转
输入	ENR	布尔	1	允许反转
输入	ENP	布尔	1	允许停止
输入	MFOR	布尔	0	手动正转信号
输入	MREV	布尔	0	手动反转信号
输入	MSTP	布尔	0	手动停止信号
输入	AFOR	布尔	0	自动正转信号
输入	AREV	布尔	0	自动反转信号
输入	ASTP	布尔	0	自动停止信号
输入	EFOR	布尔	0	紧急正转信号
输入	EREV	布尔	0	紧急反转信号
输入	ESTP	布尔	0	紧急停机信号
输入	ERR	布尔	0	设备故障信号
输入	EMSK	整数	65535	故障停机掩码
输入	RST	布尔	0	故障复位
输入	SF	布尔	0	设备正转状态
输入	SR	布尔	0	设备反转状态
输入	STP	布尔	0	设备停止状态
输入	TM	整数	0	启动超时设置, 单位: 毫秒
输入	DFN	整数	0	驱动信号模式
输入	TP	整数	1	启停脉冲的输出时间, 单位: 毫秒
输入	TF	整数	3	设备正反转切换时间, 单位: 毫秒
输出	AUTO	布尔	0	手动/自动控制
输出	Y1	布尔	0	驱动信号 1
输出	Y2	布尔	0	驱动信号 2
输出	Y3	布尔	0	驱动信号 3



输出	ESTA	整数	0	故障状态
----	------	----	---	------

## 描述

该指令用来驱动类似双向电机这类能够正反两个方向持续运行的设备。

该指令具有禁操、就地、手动、自动、超驰五种工作模式。其中 EFOR、EREV、ESTP 这组信号用于超驰模式；MFOR、MREV、MSTP 这组信号用于手动模式；AFOR、AREV、ASTP 这组信号用于自动模式。

模式的优先级别是，禁操>就地>超驰>手动>自动。

在 EIN=0 时，指令进入禁操模式，指令不接受任何输入信号（EIN 除外），也不产生输出的任何变化。EIN=1 时，指令可进入其他模式运行。

当 REM=0 时，设备处于就地控制状态，此时超驰、手动和自动输入信号不起作用，但各种故障信号处理会正常进行。

当 REM=1 时，只要 EFOR、EREV、ESTP 输入端有信号，指令立刻按照超驰模式运行，根据超驰的输入信号输出相应的超驰命令。

当 REM=1，TOM=1 时，指令处于手动模式，此时手动信号 MFOR、MREV、MSTP 有效，输出 AUTO=0。

当 REM=1，TOA=1 时，指令处于自动模式，此时自动信号 AFOR、AREV、ASTP 有效，输出 AUTO=1。

如果 TOM 和 TOA 都等于 0，则指令维持在原有模式不变。

如果 TOM 和 TOA 都等于 1，则指令进入手动模式。

如果 TOM 和 TOA 都等于 1，则指令进入手动模式。

ENF、ENR、ENP 分别用于允许“正转”、“反转”、“停止”的输入信号。这几个允许信号同时对超驰、手动、自动输入起效。

EFOR、MFOR 或 AFOR 有效 (=1) 且 ENF=1 时，发出正转命令；EREV、MREV 或 AREV 有效 (=1) 且 ENR=1 时，发出反转命令；ESTP、MSTP、ASTP 有效 (=1) 且 ENP=1 时，发出停止命令。

如果正转命令已经发出，此时 ENF 从 1 变为 0，对输出没有影响。同理，反转命令已经发出后，ENR 从 1 变为 0，对输出没有影响。

TM 用来控制设备启动超时，当 Y1 或 Y2 发出命令后开始计时，若超过 TM 设置的时间而相应的 SF 或 SR 信号还没有返回，则指令故障状态 ESTA 置为超时故障。TM 必须小于 3600s，TM=0 时，不进行超时判断。

DFN 用来设置 Y1~Y3 的输出方式，Y1 通常作为正转指令，Y2 作为反转指令，Y3 作为停止指令。

DFN=0 时，Y1、Y2 输出常信号，Y3 没有作用；



DFN=1 时，Y1~Y3 输出长度等于 TP 的脉冲，Y1 发送正转脉冲，Y2 发送反转脉冲，Y3 发送停止脉冲。

SF 输入信号连接设备的正转状态，SR 输入信号连接设备的反转状态。

如果设备额外提供了停止状态，可以连接 STP 输入端。如果设备未提供该信号，端口保持在默认值 0 即可。

SF、SR 和 STP 三个信号，只要出现 2 个或 3 个信号同时为 1，则产生信号故障，反映在 EST 的输出中。TP 用来设置正转、反转和停止脉冲的宽度，在 DFN=1 时该输入有效，在 DFN=0 时该参数被忽略。如果设置为 0，则根据状态反馈信号 SF、SR 来清除脉冲输出。

TF 用来设置正反转的切换时间，这是为保护机械和电气设备，在正转（或反转）后不允许立刻反转（或正转），而是延时一个短暂的时间使机械的惯性得以释放。TF=0 表示可以立刻反转（应用于要求较快响应速度且惯性很小的设备中）。

当 ERR 被置位时，指令故障状态 ESTA 被置硬件故障。

故障停机掩码 EMSK 用来选择需要自动停机的各种故障信号。具体使用方法见下文关于故障码的说明。

RST 信号用于复位故障状态。当故障发生后，ESTA 会保持故障码输出，即使故障源已经消失。当 RST=1，且故障源已不存在，此时 ESTA 恢复到 0，Y1 和 Y2 输出恢复正常。RST 具有自复位功能，即指令内部会将 RST 自动恢复到 0 值。

DEV2 指令为每种可能出现的故障设置了代码，如下表所示：

代码	故障	说明
1	硬件故障	当故障输入 ERR=1 产生
2	跳闸故障	REM=1 时，如果设备在正转或反转状态下，本指令未给出停止命令，设备自行停止运行，则产生本故障
4	超时故障	设备未能在指定的时间内做出相应动作。 当 Y1（或 Y2）输出后，TM 时间内，SF（或 SR）未收到相应反馈时产生。
8	信号错误	SF、SR、STP 同时出现 2 个或 3 个信号为 1

ESTA 用于表示当前各种故障，代码见上表。当两种或以上故障同时发生，则 ESTA 显示这几种故障码的总和，例如硬件故障（1#）和信号错误（8#）同时发生，ESTA 就显示为 9。

当各种故障发生后，用户可以选择哪些故障用于停止设备运行。通过在 EMSK 设置代码来选择需要停机的故障，将那些停机故障的代码相加设置于 EMSK。例如，希望超时故障（4#）和信号错误（8#）发生时停止设备，但硬件故障（1#）不要停止，可以将 EMSK 设置为 12。

当引起停机的故障发生时，如果是常信号驱动（DFN=0），指令会停止 Y1 的输出，如果是脉冲驱动（DFN=1），除了取消 Y1 的输出，还会在 Y2 输出停止脉冲。此后只有在 RST 输入 1 信号脉冲时，指令才能重新输出 Y1、Y2 的驱动信号。



## 应用

本指令通常用于可逆皮带，双向给料机等持续运转的设备控制。

用户应根据设备电气柜的控制方式，选择合适的驱动方式和输出信号连接。下表列出了现场电气柜常用接口场景。

场景	电气柜	DCS 指令
1	提供“正转”和“反转”信号接口，非自保持	选择“常信号”驱动，Y1 连接“正转”信号，Y2 连接“反转”信号
2	提供“正转”、“反转”、“停止”三个信号接口，自保持	选择“脉冲信号”驱动，Y1 和 Y2 分别连接“正转”和“反转”信号，Y3 连接“停止”信号

## 6.7 STEP 步序控制

## 名称

标记：STEP

名称：步序控制

## 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	IN1	布尔	0	第 1 步动作完成标志
输入	TS1	整数	0	第 1 步时间，单位：毫秒
输入	IN2	布尔	0	第 2 步动作完成标志
输入	TS2	整数	0	第 2 步时间，单位：毫秒
输入	IN3	布尔	0	第 3 步动作完成标志
输入	TS3	整数	0	第 3 步时间，单位：毫秒
输入	IN4	布尔	0	第 4 步动作完成标志
输入	TS4	整数	0	第 4 步时间，单位：毫秒
输入	IN5	布尔	0	第 5 步动作完成标志
输入	TS5	整数	0	第 5 步时间，单位：毫秒
输入	IN6	布尔	0	第 6 步动作完成标志
输入	TS6	整数	0	第 6 步时间，单位：毫秒
输入	IN7	布尔	0	第 7 步动作完成标志
输入	TS7	整数	0	第 7 步时间，单位：毫秒



输入	IN8	布尔	0	第 8 步动作完成标志
输入	TS8	整数	0	第 8 步时间, 单位: 毫秒
输入	TOS	整数	8	控制步骤数
输入	START	布尔	0	步序启动及复位指令
输入	PAUSE	布尔	0	步序暂停
输入	STOP	布尔	0	步序停止指令
输入	SKIP	整数	0	跳步指令
输入	ERX1	布尔	0	停止步序执行故障
输入	ERX2	布尔	0	暂停步序执行故障
输入	EOT	整数	0	步序时间作用标志
输入	ERT	整数	0	步序超时处理选择
输入	RST	布尔	0	复位所有输出
输出	Q1	布尔	0	第 1 步指令
输出	Q2	布尔	0	第 2 步指令
输出	Q3	布尔	0	第 3 步指令
输出	Q4	布尔	0	第 4 步指令
输出	Q5	布尔	0	第 5 步指令
输出	Q6	布尔	0	第 6 步指令
输出	Q7	布尔	0	第 7 步指令
输出	Q8	布尔	0	第 8 步指令
输出	CSTP	整数	0	当前步序数
输出	RUN	布尔	0	步序运行标志
输出	HOLD	布尔	0	步序暂停
输出	END	布尔	0	步序完成标志
输出	FAIL	整数	0	步序故障信号
输出	RXT	整数	0	当前步序剩余时间, 单位: 毫秒
输出	EXT	整数	0	当前步序执行时间, 单位: 毫秒
输出	AFT	整数	0	所有步序执行时间, 单位: 毫秒

## 描述

该指令用来执行步序控制。共支持 8 个步序的执行, 通过将 STEP 指令串联, 可以扩展可控制的步序数。

利用 START、PAUSE、STOP 可以控制步序的启动、暂停、停止操作。

启动信号 START 有两种情况可以启动步序的执行。一种是复位信号输入后, 步序数等于 0, 另一种情况是步序执行完毕, 当前步序数 CSTP 大于需要执行的步序数 TOS, 结束标志 END=1。其他情况下, 该输入信号没有作用。

步序控制开始执行, 步序运行标志 RUN 输出 1, 步骤输出 CSTP=1, 步序完成标志 END=0。

步序运行, 则第一步指令 Q1=1。

对于步序的完成可以分为条件完成或时间完成两种情况, 例如第一步序, 条件完成用



IN1 输入, IN1=1 表示该步条件完成, 时间完成用 TS1 输入。如果 TS1=0, 则不判断时间完成。对于同时使用两个条件的话, 这两个条件取“或”的关系, 即条件完成或时间完成都使本步骤完成。

如果第一步条件满足, Q1=0, Q2=1, STEP=2, 进入第二步序的控制;

依此类推, 直到当前步序数大于需要控制的步序数 TOS 时, 步序控制结束, RUN=0, 步序控制结束标志 END=1。

PAUSE 用来暂停步序的执行, PAUSE 采用常信号控制, 当 PAUSE=1 时, 当前执行的步序被暂停, 当前步序输出 Qn=0, 当前步序执行时间和步序剩余时间的计时被暂停 (不清除), 步序暂停标志 HOLD=1, 步序运行标志 RUN 和当前步序数 CSTP 保持不变。此时步序条件 INn 不起作用。当 PAUSE=0 后, 继续当前步序的执行。

输入 STOP 脉冲来停止步序的执行, 当前步序输出 Qn=0, 步序运行标志 RUN=0, 当前步序数 CSTP 保持不变。步序停止后, 需要复位 RST 输入脉冲后, 使用 START 来重新启动步序。

跳步指令 SKIP 用来直接进入第 n 步运行。每次 SKIP 从 0 变化到步序数时起作用, 本指令立刻进入 SKIP 对应的步序执行。此后如果 SKIP 不复位到 0, 则该输入不再起作用。若 SKIP 指定的步序大于需要控制的步序 TOS, 该次输入无效。

该指令带有外部故障处理功能, 外部故障分为停止步序执行的故障 ERX1 和暂停步序执行的故障 ERX2 两种情况。

停止步序执行的故障, 停止步序的执行, 执行标志 RUN=0, 当前执行步数 CSTP 保持在最后执行的步序上。故障输出 FAIL 被置位。这种故障需要复位 RST 输入脉冲后, 才能重新运行。

暂停步序执行的故障和 PAUSE 的作用相似, 但在故障输出 FAIL 上会有相应的输出, 而 PAUSE 不会引起 FAIL 的变化。

步序时间作用标志 EOT 用来定义每个步序上定义的时间 TSn 是作为步序执行时间还是超时时间。使用 EOT 的低 8 位表示对应 8 个步序的定义, 当位 (Bit) =0, 表示时间是执行时间, 位=1, 表示时间是超时时间。

在判断条件完成时, 可以设置超时时间 TEn。例如在执行步序一时, 如果 TS1 设置的时间计满后, IN1 仍然等于 0, 则认为该步序执行失败。

ERT 用来设置超时故障后的处理方法, ERT=0 表示步序超时作为停止执行故障处理, ERT=1 表示超时作为暂停执行故障处理。

步序复位输入 RST 用来复位所有的输出, 同时把所有步序输出 Qn 清 0, 当前步序数 CSTP 置 0, 故障输出 FAIL 置 0。

CSTP 为当前步序数, 复位后, CSTP=0, 启动步序执行后 CSTP=1, 此后每个步序条件满足 CSTP 加 1, 直到完成 TOS 定义的步序数后, 整个步序结束, END=1。

HOLD 为步序暂停标志, 当 PAUSE=1 或出现暂停执行的故障时, HOLD=1, 否则=0。

RXT 是当前步序的剩余时间, 当 TSn 不等于 0 时, 该输出会逐秒减小; 当 TSn=0, 该输出保持在 0 上。



EXT 是当前步序的执行时间，该输出会逐秒增加。

AXT 是整个步序的执行时间，在步序运行信号 RUN=1 后开始计时，在 RUN=0 时计时停止，在步序暂停 HOLD=1 时计时暂停。

## 6.8 SSA 数字驱动伺服模块

### 名称

标记：SSA

名称：数字驱动伺服模块

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	SP	实数	0	控制设定值
输入	PV	实数	0	控制反馈值
输入	DB	实数	2	差值死区
输入	MB	实数	5	脉冲调节区
输入	TP	整数	1000	脉冲周期，单位毫秒
输入	TE	整数	500	脉冲宽度，单位毫秒
输出	INC	布尔	0	增输出
输出	DEC	布尔	0	减输出

### 描述

该指令根据设定值 SP 与反馈值 PV 的偏差，控制增减开关量的输出。

当 EIN=0 时，INC 和 DEC 不产生任何输出。

当 EIN=1 时，INC 和 DEC 根据以下规则输出：

当  $|SP-PV| \leq DB$  时，增减输出全部为 0；

当  $DB < |SP-PV| \leq MB$  时，若  $SP-PV > 0$ ，则增输出 INC 按 TP 和 TE 定义的脉冲输出，若  $SP-PV < 0$ ，则减输出 DEC 按 TP 和 TE 定义的脉冲输出；

当  $|SP-PV| > MB$  时，若  $SP-PV > 0$ ，则增输出 INC 常 1 输出，若  $SP-PV < 0$ ，则减输出 DEC 常 1 输出。

在参数定义中，应保证  $0 \leq DB \leq MB$ ， $TE < TP$ 。

若  $DB=MB$ ，则没有脉冲输出功能。



## 应用

该指令通常应用在开关控制-模拟反馈的电动阀上。

利用指令的 INC 和 DEC 输出控制阀门电机的正转和反转；利用阀门位置反馈信号 PV 与开度设定 SP 确定电机的运转方向。

为比较精确定位阀门位置，防止惯性过冲，本指令设置了两个调节区域，在偏差较大时采用常信号控制，当偏差进入一定范围（MB）后，指令自动变成脉冲输出，以降低阀门的变化速度。在偏差进入死区（DB）后，指令停止调节。

用户应根据阀门的特性，选择脉冲调节的周期和脉冲宽度。一般惯性较大的阀门选用较长的周期和脉宽，惯性较小的阀门选择短周期和短脉宽。

## 6.9 KBML 键盘模拟量增减

## 名称

标记：KBML

名称：键盘模拟量增减

## 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	SP	实数	0	设定值
输入	TR	实数	0	跟踪信号
输入	TH	实数	100	输出上限
输入	TL	实数	0	输出下限
输入	L1	实数	0.1	增减步幅 1
输入	L2	实数	0.5	增减步幅 2
输入	L3	实数	1.0	增减步幅 3
输入	T	整数	500	以毫秒为单位的增减周期
输入	TS	布尔	0	跟踪开关
输入	SET	布尔	0	置数开关
输入	INC	布尔	0	数值增开关
输入	DEC	布尔	0	数值减开关
输出	Y	实数	0	操作器输出
输出	D	布尔	0	操作指示输出



## 描述

本指令对模拟量数据进行跟踪、置数、增减三种操作。

当跟踪开关  $TS=1$  时，数据操作处于跟踪状态，输出  $Y$  等于  $TR$ 。当  $TS$  从 1 变为 0 时，输出  $Y$  保持不变：

在  $TS=0$  时，若置数开关  $SET=1$ ，则输出  $Y$  变为设定值  $SP$ ；否则输出  $Y$  保持不变。

在  $TS=0$  且  $SET=0$  时，若  $INC$  或  $DEC$  等于 1，输出  $Y$  开始按指定步幅开始连续加减，在  $INC$  或  $DEC$  等于 0 后，增减过程结束。在增减过程中，指令先对输出  $Y$  以每隔  $T$  时间增加或减少步幅  $L1$ ，在连续操作 10 次后，若  $INC$  或  $DEC$  仍旧为 1，则改用步幅  $L2$  进行增减，这样连续操作 10 次后，若  $INC$  或  $DEC$  还是等于 1，则再改用步幅  $L3$  进行增减；

步幅 1、2、3 用输出上下限  $TH$  和  $TL$  的百分比表示，即步幅 1 表示每次增减量是上下限范围( $TH-TL+1$ )的 0.1%；

若  $INC$  与  $DEC$  同时为 1， $INC$  起作用；

只要  $SET$ 、 $INC$ 、 $DEC$  开关信号起作用，则输出  $D$  变为 1。

参数设置中，要求  $TH>TL$ 。



## 第7章 过程处理

### 介绍

该类算法主要是涉及到过程控制相关的运算指令,可以利用运算指令构成特定的表达式或实现某种算法。

该类指令中含有 EIN 和 EOUT 引脚的指令,如果没有特殊说明的,当 EIN=1 时,指令进行正常计算;当 EIN=0 时,指令的计算停止,输出保持在最后的计算结果。EOUT 的值与 EIN 始终保持一致。

该类指令包括:

序号	标记	名称
1	EPID	含手操器 PID 控制
2	PID	偏差 PID 控制
3	SST	参数自整定
4	ERV	求偏差
5	LDLG	超前滞后
6	INTG	积分运算
7	DIFF	微分运算
8	FLT	滤波器
9	TSEL	信号二选一
10	RSEL	信号三选一
11	SFT	无扰动切换
12	LMT	幅度限制
13	LMV	变化速率限制
14	BLN2	二输出平衡模块
15	TAS	时域模拟量统计
16	TDS	时域开关量统计
17	DLY	纯滞后模块
18	PTC	热力性质计算
19	DFML	差分方程
20	FUZY	模糊控制器
21	SALM	慢信号报警
22	ACCU	累积计算
23	FLW	流量计算



## 7.1 含手操器 PID 控制

### 名称

标记: EPID

名称: 含手操器 PID 控制

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	MSI	实数	0	手动控制值输入
输入	SPI	实数	0	调节设定值输入
输入	PV	实数	0	被控量
输入	TR	实数	0	跟踪量
输入	EV	实数	0	超驰量输入
输入	TOL	布尔	1	就地开关
输入	TOE	布尔	0	超驰开关
输入	TOM	布尔	0	手动开关
输入	TOA	布尔	0	自动开关
输入	SFT	整数	2	无扰切换模式
输入	NFD	布尔	0	调节方向选择
输入	KP	实数	0	比例调节系数
输入	TI	实数	0	积分时间常数
输入	TD	实数	0	微分时间常数
输入	KD	实数	0	微分惯性参数
输入	SPH	实数	100	调节设定值上限
输入	SPL	实数	0	调节设定值下限
输入	YH	实数	100	输出值上限
输入	YL	实数	0	输出值下限
输入	YRU	实数	0	输出上升速率/秒
输入	YRD	实数	0	输出下降速率/秒
输入	DB	实数	0	控制量死区
输入	IDV	实数	20	积分分离阈值
输入	DISV	实数	0	前馈输入量
输入	TCP	实数	1	计算周期, 单位: 秒
输出	OV	实数	0	输出变量
输出	ST	整数	0	控制状态
输出	QHM	布尔	0	输出达到上限
输出	QLM	布尔	0	输出达到下限



作用	标记	类型	缺省值	说明
输出	MSO	实数	0	手动设定值输出
输出	SPO	实数	0	调节设定值输出
输出	ER	实数	0	实际偏差值

## 描述

本指令集成了偏差计算、PID 调节、模拟量手操器三项功能。由于指令参数很多，下面分别按功能介绍各相关参数。

### (1) 偏差计算

与偏差计算有关的参数是：SPI、PV、NFD、SPH、SPL、DB、ER。

实际偏差 ER 值由调节设定值 SP、调节量输入 PV、调节方向选择开关 NFD 决定。

指令在计算偏差前，先根据 SPH 和 SPL 检查设定值 SP 的上下限，如果 SP 大于 SPH，按 SPH 值计算偏差；如果 SP 小于 SPL，按 SPL 值计算偏差；如果 SP 在 SPH 和 SPL 之间，按 SP 实际数值计算偏差。

若 NFD=0，ER=SP-PV，输出值随偏差数值变大而增加；若 NFD=1，ER=PV-SP，即反向调节，输出值随偏差数值变大而减小。

DB 输入决定调节死区，当偏差 ER 在 DB 范围之内时  $|ER| < DB$ ，偏差按 0 值参与 PID 计算。

### (2) PID 调节

与 PID 调节相关的参数有：KP、TI、TD、KD、IDV、DISV、TCP。

KP、TI、TD 分别是 PID 调节的比例、积分和微分系数。

本指令 PID 调节计算的基本公式是：

$$\Delta OV = KP[E_n - E_{n-1} + \frac{T_p \cdot E_n}{TI} + \frac{TD}{T_p} \cdot (E_n - 2E_{n-1} + E_{n-2})],$$

$$OV_n = OV_{n-1} + \Delta OV$$

其中  $E_n$  是本周周期偏差值， $E_{n-1}$  是前一周周期偏差值， $E_{n-2}$  是前二周期偏差值， $T_p$  是计算周期， $\Delta OV$  是每个计算周期输出的改变量， $OV_n$  是本周周期输出值。

计算周期和以前周期的偏差值由指令自动提供，KP、TI、TD 根据用户设置的参数提供，本周周期偏差值由偏差计算结果提供。

为改善 PID 调节效果，本指令提供微分惯性和积分分离。



在离散系统中，微分环节如果简单按公式  $\frac{E_n - E_{n-1}}{T_p}$  代替微分量，是不能正确产生调

节效果的。通过在上述公式后串联一个惯性环节，可以有效改善微分调节在离散系统的表现。KD 参数用于调整惯性常量，当 KD=0 时，无惯性效果，当 KD>0 时，计算微分惯性，且 KD 值越大，则惯性效果越明显。

PID 调节时，若积分作用太强，会使系统产生过大的超调量，振荡剧烈，且调节时间过长。通过在偏差较大时，取消积分作用，在偏差减小到一定值后，再加上积分作用（即积分分离），这样就可以即减小了超调量，改善动态特性，又保持了积分作用。IDV 设定积分分离阈值，当 IDV=0，无积分分离效果，当 IDV>0 时，如果偏差大于 IDV 则自动取消积分环节。

DISV 是前馈输入，在 PID 计算结果上叠加一个偏置量。

TCP 是 PID 计算周期，根据控制对象的特性确定该参数，对象的惯性越大，则 TCP 的数值越大。TCP 时间必须大于过程（程序）的计算周期。

### （3）模拟量手操器

手操器相关的参数有：MSI、TR、EV、TOL、TOE、TOM、TOA、SFT、YH、YL、YRU、YRD、OV、ST、QHM、QLM、MSO、SPO。

模拟量手操器具有禁操、就地、手动、自动、超驰五种工作模式。

模式的优先级是，禁操>就地>超驰>手动>自动。

模式	模式开关	OV	ST	MSO		SPO	
				SFT=0,1	SFT=2,3	SFT=0,2	SFT=1,3
禁操	EIN=0	不变	不变	不变	不变	不变	不变
就地	EIN=1, TOL=1	TR	4	MSI	OV	SPI	PV
超驰	TOL=0, TOE=1	ER	2	MSI	OV	SPI	PV
手动	TOL=0, TOM=1	MSI	0	MSI	MSI	SPI	PV
自动	TOL=0, TOA=1	PID	1	MSI	OV	SPI	SPI

当四个模式开关 TOL、TOE、TOM、TOA 全部是 0 的时候，若此时 ST=1（自动模式），则继续保持自动模式；若 ST 是其他数值，则进入手动模式。

无扰切换参数 SFT 确定在手自动模式间切换时，是否需要保持输出不变。SFT 共有 4 种参数，分别是：

SFT=0 时，有扰切换；

SFT=1 时，切入手动有扰，输出跳变至最后一次手动操作值，切入自动时无扰，此时 SP 与 PV 相同；

SFT=2 时，切入手动无扰，输出保持在上一周期值不变，切入自动有扰，输出从上一周期值开始变化，直到 SP=PV；



SFT=3 时，切入手动和自动都无扰动。

YH 和 YL 决定输出 OV 的上下限。当 PID 的计算值大于 YH 时，OV 等于 YH，QHM=1；当 PID 的计算值小于 YL 时，OV 等于 YL，QLM=1；否则 OV 等于 PID 的计算值，OHM 和 OLM 等于 0。

YRU 和 YRD 限制输出 OV 的上升和下降每秒变化速率，当 YRU=0（或 YRD=0）时表示不限制上升（或下降）速率。

## 应用

EPID 指令最适合用于模拟量的单回路调节。由于本指令高度集成，组态工作相当简单。

首先给 EPID 指令设置一个实例名称，然后将调节阀的控制量连接 OV，阀门反馈连接 TR（可不连），被控量连接 PV。根据被控对象特点和控制要求，设置 KP、PI、PD、SFT 等各项参数。

在监控画面上创建一个回路操作面板的实例，设置面板实例所连接的 EPID 指令，即可完成单回路程序组态和画面组态工作。

对于串级 PID 的组态，通常不含手操器的主回路可以选用下面的 PID 指令，副回路选用带手操器的 EPID 指令。

在早期版本中没有 TCP 这一参数，采用与过程计算周期相同的速率进行 PID 计算。这样在控制大惯性对象时，会由于采样时间太短而影响控制的稳定性。

针对不同控制对象，TCP 的经验范围如下：

温度信号：5~10 秒

压力信号：1~5 秒

流量信号：0.5~2 秒

液位信号：1~5 秒

## 7.2 偏差 PID 控制

### 名称

标记：PID

名称：偏差 PID 控制

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能



作用	标记	类型	缺省值	说明
输出	EOUT	布尔	1	输出使能
输入	AUTO	整数	0	跟踪/PID 控制选择
输入	SP	实数	0	设定值修改量
输入	PV	实数	0	调节量输入
输入	TR	实数	0	跟踪输入值
输入	NFD	布尔	0	调节方向选择开关
输入	KP	实数	1	比例调节系数
输入	TI	实数	0	积分调节常数
输入	TD	实数	0	微分增益
输入	KD	实数	0	微分滞后参数
输入	SPH	实数	100	偏差值上限
输入	SPL	实数	-100	偏差值下限
输入	YH	实数	100	OV 值上限
输入	YL	实数	0	OV 值下限
输入	YRU	实数	0	输出值上升速率/秒
输入	YRD	实数	0	输出值下降速率/秒
输入	DB	实数	0	控制量死区
输入	IDV	实数	20	积分分离阈值
输入	DISV	实数	0	前馈输入量
输入	TCP	实数	1	计算周期
输出	OV	实数	0	控制输出
输出	QHM	布尔	0	OV 达到上限值
输出	QLM	布尔	0	OV 达到下限值
输出	ER	实数	0	实际偏差值

## 描述

偏差 PID 具有跟踪和自动调节两种控制方式。

当 AUTO=1 时，进入自动调节方式运行；当 AUTO≠1 时，进入跟踪方式运行。

在跟踪方式下，控制输出 OV=TR，偏差输出 ER=0；

在自动调节方式下，控制输出 OV 由设定值 SP、调节量输入 PV 计算实际偏差 ER，然后由 ER 及 PID 参数计算控制输出 OV。

指令在计算偏差前，先根据 SPH 和 SPL 检查设定值 SP 的上下限，如果 SP 大于 SPH，按 SPH 值计算偏差；如果 SP 小于 SPL，按 SPL 值计算偏差；如果 SP 在 SPH 和 SPL 之间，按 SP 实际数值计算偏差。

若 NFD=0，ER=SP-PV，输出值随偏差数值变大而增加；若 NFD=1，ER=PV-SP，即反向调节，输出值随偏差数值变大而减小。

DB 决定调节死区，当偏差 ER 在 DB 范围之内时  $|ER| < DB$ ，偏差按 0 值参与 PID 计



算。

KP、TI、TD 分别是 PID 调节的比例、积分和微分系数。

本指令 PID 调节计算的基本公式是：

$$\Delta OV = KP[E_n - E_{n-1} + \frac{T_p \cdot E_n}{TI} + \frac{TD}{T_p} \cdot (E_n - 2E_{n-1} + E_{n-2})],$$

$$OV_n = OV_{n-1} + \Delta OV$$

为改善 PID 调节效果，本指令提供微分惯性和积分分离。

KD 参数用于调整微分环节的惯性常量，当 KD=0 时，无惯性效果，当 KD>0 时，计算微分惯性，且 KD 值越大，则惯性效果越明显。

IDV 设定积分分离阈值，当 IDV=0，无积分分离效果，当 IDV>0 时，如果偏差大于 IDV 则自动取消积分环节。

DISV 是前馈输入，在 PID 计算结果上叠加一个偏置量。

TCP 是 PID 计算周期，根据控制对象的特性确定该参数，对象的惯性越大，则 TCP 的数值越大。TCP 时间必须大于等于过程（程序）的计算周期。

YH 和 YL 决定输出 OV 的上下限。当 PID 的计算值大于 YH 时，OV 等于 YH，QHM=1；当 PID 的计算值小于 YL 时，OV 等于 YL，QLM=1；否则 OV 等于 PID 的计算值，OHM 和 OLM 等于 0。

YRU 和 YRD 限制输出 OV 的上升和下降每秒变化速率，当 YRU=0（或 YRD=0）时表示不限制上升（或下降）速率。

## 应用

PID 指令适用于单纯进行 PID 计算的地方，例如在串级 PID 中主回路的控制。

将 PID 指令和 SMA 指令组合，可以构成 EPID 指令的所有功能。

## 7.3 SST 参数自整定

### 名称

标记：SST

名称：参数自整定

### 结构



作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	PV	实数	0	过程量
输入	OV	实数	0	控制量
输入	PX	实数	1	比例调节初值
输入	IX	实数	0	积分调节初值
输入	DX	实数	0	微分调节初值
输入	BETA	实数	0	控制量变化率权重
输入	DL	实数	0	纯滞后时间, 单位: 秒
输入	MIU	实数	0.98	遗忘因子, 0~1
输入	ALF	实数	0.1	柔化因子, 0~1
输入	TRE	整数	1	辨识周期, 单位: 秒
输入	TOP	整数	10	寻优周期, 单位: 秒
输入	RST	布尔	0	整定复位
输入	TS	布尔	0	跟踪开关
输出	PY	实数	1	比例值输出
输出	IY	实数	0	积分值输出
输出	DY	实数	0	微分值输出
输出	MP1	实数	0	对象模型参数 1
输出	MP2	实数	0	对象模型参数 2
输出	P11	实数	0	对象模型迭代参数 1
输出	P12	实数	0	对象模型迭代参数 2
输出	P21	实数	0	对象模型迭代参数 3
输出	P22	实数	0	对象模型迭代参数 4

## 描述

SST 指令用来自动寻找 PID 回路的最优调节参数。

当 RST=1 时, 指令内部复位, 使用输入 PX、IX 和 DX 作为最初的一组调节参数。

当 RST=0 时, 指令开始对象辨识和参数寻优的过程。对象辨识周期由 TRE 参数设置, 寻优周期由 TOP 参数设置。指令要求 TOP 比 TRE 大 10 倍或以上。

在辨识周期中, 指令根据 PV 和 OV 的数值序列, 及 BETA 的设置, 建立控制对象的数学模型, 模型特征数值使用 MP1、MP2、P11、P12、P21 和 P22 共 6 个参数表示。这组参数将来可用作系统高级优化控制的参考数据。

SST 指令具有根据对象变化, 自动调整 PID 控制参数的能力。由于实际控制对象的特性在运行中会不断变化, 用户需要调整 BETA、MIU、ALF 参数, 以兼顾控制参数 PY、IY、DY 的稳定性和响应控制对象变化的速度。一般来说这是相互矛盾的特性, 控制参数稳定性高, 那响应变化的能力就弱; 如果响应变化能力强, 那控制参数可能变化就比较频繁。

BETA 是 0~1 的系数, 用于调整对控制量 OV 或被控量 PV 的重视程度, 缺省值 0.5。数



值接近 1，表示寻找 PID 参数时，更注重控制量 OV 的平稳；数值接近 0，表示寻找 PID 参数时，更注重被控量 PV 与 SP 的偏差大小。

MIU 是 0.9~1 之间的系数，用于调整对象的历史数据在辨识过程中被遗忘的速度，缺省值 0.98。该数值越接近 1，控制对象的历史被遗忘的速度越慢，在寻优时所参考的时间范围就越宽，对象扰动对控制参数调整的影响时间也就越长；反之，控制对象的历史被遗忘得越快，寻优参考的时间范围较窄，对象扰动对控制参数的影响时间就越短。

ALF 是 0~1 之间的系数，用于调整新的辨识结果对对象的影响程度，缺省值 0.1。该数值越接近 0，则新的结果对对象的影响越大，取极限值 0，会完全忽略原对象数据；反之，则新的结果对对象的影响越小，取极限值 1，会完全忽略新的计算结果（内部的对象模型不再变化）。

DL 用于设置控制对象的纯滞后时间，即 OV 输出至对象开始响应的的时间，主要用于长滞后对象的辨识中。

TS 跟踪开关用于切除 PID 参数输出。当 TS=1 时，PY、IY、DY 与 PX、IX 和 DY 相同；当 TS=0 时，PY、IY、DY 根据寻优结果输出。

## 应用

SST 可广泛应用于各种对象的自动控制中，由于具有对象自动辨识能力，可以自动适应控制对象特性变化的地方。

根据控制对象的特性需要合理选择 TRE 和 TOP。惯性越大的对象，对应的 TRE 时间就越长。经验数值如下：

温度信号：5~10 秒

压力信号：1~5 秒

流量信号：1~2 秒

液位信号：1~5 秒

TOP 的时间一般取 10~20 倍的 TRE 时间。

还需要注意 RST 和 TS 的区别。

RST 起复位的作用，会将指令内部数据全部清除，相当于重新开始辨识对象。

TS 仅切除寻优结果的输出，但内部对象辨识仍然在进行。该功能可以方便运行时，对象控制自整定功能的投入与切除。

## 7.4 ERV 偏差计算

### 名称

标记：ERV



名称：偏差计算

## 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	SP	实数	0	设定值
输入	PV	实数	0	输入值
输入	K	实数	1	输入值的增益
输入	C	实数	0	输入值的偏置
输入	HLM	实数	100	偏差上限
输入	LLM	实数	0	偏差下限
输入	DB	实数	0	输入死区
输出	ER	实数	0	偏差值
输出	QM	布尔	0	偏差越限
输出	QHM	布尔	0	偏差越上限
输出	QLM	布尔	0	偏差越下限

## 描述

求输入值 PV 与设定值 SP 的偏差量，以 ER 信号输出，可以设定输入信号的死区范围。

输出信号具有上下限设定，当 HLM=LLM=0 时，没有上下限功能，否则偏差输出 ER 被限定在 HLM 和 LLM 之间。

指令的表达式为：

令  $X = SP - (PV * K + C)$ ，则

$$ER = \begin{cases} HLM & X \geq HLM \\ X & DB < X < HLM \\ 0 & -DB < X < DB \\ X & LLM < X < -DB \\ LLM & X \leq LLM \end{cases}$$

当  $X > HLM + DB$  时，QHM = 1；当  $X < HLM - DB$  时，QHM = 0；

当  $HLM - DB \leq X \leq HLM + DB$  时，QHM 不变。

当  $X < LLM - DB$  时，QLM = 1；当  $X > LLM + DB$  时，QLM = 0；

当  $LLM - DB \leq X \leq LLM + DB$  时，QLM 不变。

通过在 HLM 和 LLM 附近设置 DB 死区，可以避免 ER 在高低限值附近时，QHM 和



QLM 频繁变化。

当 QHM 或 QLM 等于 1 时，QM 等于 1，QHML 和 QLM 都等于 0 时，QM 等于 0，即 QM 等于 QHM 和 QLM 的逻辑或。

## 7.5 LDLG 超前滞后

### 名称

标记：LDLG

名称：超前滞后

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X	实数	0	输入变量
输入	TR	实数	0	跟踪值
输入	TS	布尔	0	跟踪选择开关
输入	LD	实数	0	超前时间常数
输入	LG	实数	1	滞后时间常数
输入	GN	实数	1	输出增益
输入	HLM	实数	100	输出上限
输入	LLM	实数	0	输出下限
输出	Y	实数	0	输出值
输出	QHM	布尔	0	输出越上限
输出	QLM	布尔	0	输出越下限

### 描述

当 TS=1 时，输出 Y 等于跟踪值 TR。

当 TS=0 时，输出 Y 产生输入信号 X 的超前滞后响应。

$$Y_n = GN \cdot \frac{(T + 2LD)X_n + (T - 2LD)X_{n-1}}{T + 2LG} - \frac{T - 2LG}{T + 2LG} Y_{n-1}$$

其中 GN 是输出增益，LD 是超前时间常数，LG 是滞后时间常数，T 是计算周期。运算中计算周期 T 由程序自动提供。

HLM 和 LLM 限定了输出 OUT 的上下限，当 HLM=LLM=0 时，输出限定不起作用。



LG 必须大于等于  $1/2 T$ ，其中 T 为计算周期

## 7.6 INTG 积分运算

### 名称

标记: INTG

名称: 积分运算

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X	实数	0	输入信号
输入	RST	布尔	0	积分值复位
输入	PRE	实数	0	积分初始值
输入	LHM	实数	100000	积分上限
输入	LLM	实数	-100000	积分下限
输入	OC	整数	0	完成模式
输出	Y	实数	0	积分输出
输出	QHM	布尔	0	达到积分上限
输出	QLM	布尔	0	达到积分下限

### 描述

本指令实现积分运算，其表达式为：

$$Y_n = \begin{cases} PRE & n = 0 \\ Y_{n-1} + \frac{X_{n-1} + X_n}{2} T & n > 0 \end{cases}$$

T 是计算周期，PRE 是积分初始值。运算中计算周期 T 由程序自动提供。

当 RST=1 时，指令进入复位状态，OUT 端输出由 PRE 初始值决定。

当 RST=0 时，指令处于积分状态，OUT 端根据公式进行积分计算。

当积分达到上限和下限时，根据完成模式 OC 决定后续行为，OC 的定义见下表：

符号	作用	数值标记
冻结	指令停止积分，Y 值不再变化，相应的 QHM/QLM 输出 1 信号，直到积分指令被复位	0
继续	指令继续积分，Y 值根据输入 X 继续变化，相应的 QHM/QLM 根据 Y 数值输出信号	1



循环 1	指令自动复位, Y 值从 PRE 开始变化, 相应的 QHM/QLM 输出一个周期的 1 信号	2
循环 2	指令自动复位, Y 值从 0 开始变化, 相应的 QHM/QLM 输出一个周期的 1 信号	3

## 应用

积分的作用是计算一段时间内, 输入数据与时间乘积的累加和。本指令在工程中的主要用途是进行流量累积计算。

积分计算公式中的计算周期  $T$ , 由程序自动提供, 单位是秒。因此本指令是计算输入数值  $X$  以秒为单位的累计值。工程中很多流量信号以  $T/h$  (吨/小时) 或  $kg/m$  (公斤/分) 为单位, 在进行计算时, 需要将输入  $X$  预先除以 3600 (小时) 或 60 (分), 或将计算结果  $Y$  除以 3600 或 60, 然后予以显示或其他后续计算。

## 7.7 DIFF 微分运算

### 名称

标记: DIFF

名称: 微分运算

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X	实数	0	输入信号
输入	TD	实数	1	微分时间常数
输入	DL	实数	1	滞后时间常数, 必须 $\geq 0$ , 单位为秒
输入	LHM	实数	100000	输出上限
输入	LLM	实数	-100000	输出下限
输出	Y	实数	0	微分输出
输出	QHM	布尔	0	达到累计值上限
输出	QLM	布尔	0	达到累计值下限

### 描述

本指令实现带一阶惯性的实际微分运算, 其功能表达式为:



$$Y_n = \frac{DL}{T + DL} [TD(X_n - X_{n-1}) + Y_{n-1}]$$

其中，TD 是微分时间，DL 是滞后时间，T 是计算周期。

当微分达到上限和下限时，置标志位 QHM 或 QLM，并停止微分操作。当微分输出 Y 重新回到上下限范围内是，QHM 和 QLM 恢复为 0 值。

## 应用

微分的作用是计算输入数据在单位时间（秒）的变化。

此处增加一阶惯性是为了改善微分方程在离散域响应特性。

## 7.8 FLT 滤波器

### 名称

标记：FLT

名称：滤波器

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X	实数	0	输入信号
输入	MODE	整数	0	滤波类型
输入	FC1	实数	0	截止频率 1，单位：Hz
输入	FC2	实数	0	截止频率 2，单位：Hz
输出	Y	实数	0	滤波后的数值

### 描述

本指令对输入模拟变量进行数字滤波。

该指令尚在开发中，对于低通滤波可使用超前滞后指令 LDLG 代替。

### 应用

工程应用中，通常可以使用低通滤波器来滤除各种过程信号（温度、压力、流量）在采样和转换中引入的高频干扰。

滤波器的截止频率对滤波的效果有直接的影响，截止频率越低，能滤除的干扰信号越多，



但也会损失部分有用信号。截止频率越高，可以保留更多的有用信号，但同时也会遗留部分干扰信号。

在应用低通滤波时，不同的过程信号最好选用与此信号相适应的截止频率，以下是一些常用信号的典型截止频率

温度信号：0.2Hz，

压力信号：0.5Hz，

流量信号：1Hz

液位信号：0.5~1Hz

另外，受计算周期的影响，截止频率太高会失去计算意义，因此目前建议截止频率最好不要高于 5Hz。

## 7.9 TSEL 信号二选一

### 名称

标记：TSEL

名称：信号二选一

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X1	实数	0	信号输入 1
输入	X2	实数	0	信号输入 2
输入	MODE	整数	0	选择方式，0=平均，1=最低，2=最高，4=信号 1，5=信号 2
输入	DB	实数	10	偏差限
输出	Y	实数	0	二选一的输出

### 描述

如果在强制方式，即  $MODE=4$  或  $5$ ，则输出值等于指定的那个输入值，本指令输出状态同输入点状态相同。

否则 (1) 如果二个输入点均为坏点，则输出为坏点，输出保持不变；

(2) 如果一个输入点为坏点，则输出等于另一个好点之值；

(3) 如果二个输入点均为好点



- (a) 如二者间偏差越限，则输出为坏点，输出保持不变；
- (b) 如二者间偏差不越限，则输出随  $MODE = 0$  或  $1$  或  $2$ ，取平均或低选或高选。

## 7.10 RSEL 信号三选一

### 名称

标记：RSEL

名称：信号三选一

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X1	实数	0	信号输入 1
输入	X2	实数	0	信号输入 2
输入	X3	实数	0	信号输入 3
输入	MODE	整数	0	选择方式，0=平均，1=最低，2=最高，3=中值，4=信号 1，5=信号 2，6=信号 3
输入	DB	实数	10	偏差限
输出	Y	实数	0	三选一的输出

### 描述

如果在强制方式，即  $MODE = 4、5、6$ ，则输出值等于指定的那个输入值，本指令输出状态同输入点状态相同。

- 否则
- (1) 如果三个都为坏点，则输出不变，输出为坏点；
  - (2) 如果二个点为坏点，则输出等于另一好点；
  - (3) 如果一个点为坏点，则：
    - (a) 如另二点间偏差越限，则输出为坏点，输出保持不变；
    - (b) 如另二者间偏差不越限，则输出随  $MODE = 0$  或  $1$  或  $2$  或  $3$ ，取平均或低选或高选或中值。
  - (4) 如果三个点均为好点，则：
    - (a) 如果二个点间的偏差不越限，而另一点对这二点的偏差越限，则



输出根据 MODE 取前二点的平均值或低选或高选。

- (b) 如果二个点间的偏差越限，而另一点对这二点的偏差不越限，则输出取后一点的值。
- (c) 如果三个点互相之间的偏差均越限，则输出不变，输出为坏点。
- (d) 如果三个点间的偏差均不超限，则输出将根据 MODE 的值取平均或低选或高选或中值。

## 7.11 SFT 无扰动切换

### 名称

标记：SFT

名称：无扰动切换

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X1	实数	0	信号输入 1
输入	X2	实数	0	信号输入 2
输入	Z	布尔	0	选择开关
输入	USL	实数	0	每秒上升变化率限制
输入	DSL	实数	0	每秒下降变化率限制
输出	Y	实数	0	无扰切换输出

### 描述

本指令按选择开关的值选择二个模拟量之一作为输出 OUT。

在 Z=0 时，选择 X1 的输入作为 OUT 的输出值，在 Z=1 时，选择输入 X2 作为 OUT 的输出值。

OUT 输出值的速率变化受 USL 和 DSL 限制。USL 限制 OUT 的上升速率，DSL 限制 OUT 的下降速率。当 USL=0 时，无上升速率限制；当 DSL=0 时，无下降速率限制。



## 7.12 LMT 幅度限制

### 名称

标记: LMT

名称: 幅度限制

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X	实数	0	信号输入
输入	H	实数	100	幅度上限
输入	L	实数	0	幅度下限
输出	Y	实数	0	限幅输出
输入	Q	布尔	0	越限标志
输出	QHM	布尔	0	输出越上限
输出	QLM	布尔	0	输出越下限

顺便增加报警功能，以方便使用，增加 bool 输出。

### 描述

本指令对输入进行限幅，输出被限幅与 H 和 L 之间。

$$Y_n = \begin{cases} L & , X \leq L \\ X & , L < X < H \\ H & , X \geq H \end{cases}$$

当  $X \leq L$  时，QLM=1，否则 QLM=0；

当  $X \geq H$  时，QHM=1，否则 QHM=0。

QLM 和 QHM 任一输出为 1 时，输出 Q=1；QLM 和 QHM 全部为 0 时，输出 Q=0。

## 7.13 LMV 变化速率限制

### 名称

标记: LMV



名称：变化速率限制

## 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X	实数	0	信号输入
输入	USL	实数	0	每秒上升变化率限制
输入	DSL	实数	0	每秒下降变化率限制
输出	Y	实数	0	速率限制器输出
输入	Q	布尔	0	越限标志
输入	QUM	布尔	0	速率达上限
输入	QDM	布尔	0	速率达下限

## 描述

本指令对输入的变化速率进行限制，输出的每秒钟的正向的变化量不大于 USL，每秒钟的反向的变化量不大于 DSL。

当输入 X 的正向变化速率（增加速率） $\geq$  USL 时，输出 Y 的正向变化速率=USL；

当输入 X 的正向变化速率 $<$ USL 时，输出 Y 的正向变化速率=输入 X 的正向变化速率；

当输入 X 的反向变化速率（减小速率） $\geq$ DSL 时，输出 Y 的反向变化速率=DSL；

当输入 X 的反向变化速率 $<$ DSL 时，输出 Y 的反向变化速率=输入 X 的反向变化速率；

USL、DSL 的值大于等于 0。

当输入 X 的变化速率达上限后，QUM=1，否则 QUM=0。

当输入 X 的变化速率达下限后，QDM=1，否则 QDM=0。

QUM 和 QDM 任一输出为 1 时，输出 Q=1；QUM 和 QDM 全部为 0 时，输出 Q=0。

## 7.14 BLN2 二输出平衡模块

## 名称

标记：BLN2

名称：二输出平衡模块

## 结构



作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X	实数	0	信号输入
输入	DB	实数	0	差值输入
输入	TR1	实数	0	第一路跟踪量
输入	TS1	布尔	0	第一路跟踪切换
输入	TR2	实数	0	第二路跟踪量
输入	TS2	布尔	0	第二路跟踪切换
输入	YH	实数	100	Y1 和 Y2 的上限
输入	YL	实数	0	Y1 和 Y2 的下限
输出	Y1	实数	0	第一路平衡输出
输出	Y2	实数	0	第二路平衡输出

## 描述

二输出平衡模块的计算逻辑如下：

如果  $TS1 = 1$ ,  $TS2 = 1$ , 则  $Y1 = TR1$ ,  $Y2 = TR2$ ;

如果  $TS1 = 0$ ,  $TS2 = 0$ , 则  $Y1 = X + DB$ ,  $Y2 = X - DB$ ;

如果  $TS1 = 1$ ,  $TS2 = 0$ , 则  $Y1 = TR1$ ,  $Y2 = 2X - Y1$ ;

如果  $TS1 = 0$ ,  $TS2 = 1$ , 则  $Y2 = TR2$ ,  $Y1 = 2X - Y2$ ;

在 Y1 或 Y2 达到上限时，闭锁增；在 Y1 或 Y2 达到下限时，闭锁减。

## 7.15 TAS 时域统计模块

### 名称

标记：TAS

名称：时域统计模块

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X	实数	0	模拟量信号
输入	MODE	整数	1	统计方式
输入	SMP	整数	60	采样个数， $SMP > 0$



输入	RST	布尔	0	复位开关
输入	Y0	实数	0	Y 的统计初值
输入	CJ	实数	1	记忆转换系数
输出	Y	实数	0	即时统计结果
输出	YJ	实数	0	前一次的统计结果

## 描述

本指令在 EIN=1 的一段时间内对输入模拟量进行累加、平均或取最大值、最小值、梯形累加和，并对前次统计值进行转换记录。

当 EIN 从 0 变为 1 时统计开始， $YJ_n = CJ \cdot Y_{n-1}$ ；本次统计结果被初始化  $Y_n = Y0$ 。

MODE 用来选择统计的方式：

MODE=0，对输入 X 取矩形累加和， $Y_n = Y_{n-1} + T \cdot X_n$ ；

MODE=1，对输入 X 取梯形累加和， $Y_n = Y_{n-1} + \frac{T}{2}(X_n + X_{n-1})$ ；

MODE=2，对输入 X 取移动平均值， $Y_n = \frac{\sum_{k=n-m+1}^n X_k}{m}$ ，当采样个数小于 SMP 时，公式

中的 m 等于采样个数，当采样个数大于等于 SMP 时， $m=SMP$ ，SMP 必须大于 0；

如果 Mode=3，取最大值， $Y_n = \max(Y_{n-1}, X_n)$ ；

如果 Mode=4，取最小值， $Y_n = \min(Y_{n-1}, X_n)$ ；

上面公式中，T 为计算周期。

当 EIN 从 1 变为 0 时，本次统计结束，输出 Y 保持不变。

当 RST=1 时，清除本次统计结果，置  $Y=Y0$ ，不进行结果的转换，YJ 保持不变。

## 7.16 TDS 时域开关量统计模块

### 名称

标记：TDS

名称：时域开关量统计模块

### 结构



作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X	布尔	1	开关量信号输入
输入	MODE	整数	1	统计方式, 0=对 0 累加, 1=对 1 累加
输入	RST	布尔	0	复位开关
输入	Y0	实数	0.0	Y 的统计初值
输入	CJ	实数	1.0	记忆转换系数
输出	Y	实数	0.0	时间统计结果, 以秒为单位
输出	YJ	实数	0.0	前一次的统计结果

## 描述

本指令在 EIN=1 的一段时间内, 对输入开关量的状态时间进行累计统计, 并对前次统计值进行转换记录。

当 EIN 从 0 变为 1 时统计开始,  $YJ_n = CJ \cdot Y_{n-1}$ ; 本次统计结果被初始化为  $Y_n = Y0$ 。

如 Mode=0, 则对输入 X 的 0 信号进行时间累计,  $Y_n = \begin{cases} Y_{n-1} + T & , X = 0 \\ Y_{n-1} & , X = 1 \end{cases}$

如 Mode=1, 则对输入 X 的 1 信号进行时间累计,  $Y_n = \begin{cases} Y_{n-1} + T & , X = 1 \\ Y_{n-1} & , X = 0 \end{cases}$ ;

上面公式中, T 为计算周期时间。

当 EIN 从 1 变为 0 时, 本次统计结束。

当 RST=1 时, 清除本次统计结果, 但不进行结果的转换, 也就是放弃本次到 RST=1 时刻的统计结果。

## 7.17 DLY 滞后模块

### 名称

标记: DLY

名称: 滞后模块

### 结构

作用	标记	类型	缺省值	说明
----	----	----	-----	----



输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X	实数	0	输入信号
输入	TR	实数	0	被跟踪量
输入	TS	布尔	0	跟踪切换开关
输入	K	实数	1	比例增益, K>0
输入	DT	实数	0	纯滞后时间常数
输入	LT	实数	1	惯性时间常数
输出	Y	实数	0	滞后输出

## 描述

本指令对输入进行滞后惯性运算：

当 TS=1 时，Y 等于被跟踪量 TR。

当 TS=0 时，进行滞后及惯性计算，

$$Y_n = \begin{cases} 0 & , n < m \\ \frac{LT}{LT+T} Y_{n-1} + \frac{K \cdot T}{LT+T} X_{n-m} & , n \geq m \end{cases}$$

上式中， $m = \frac{DT}{T}$ ，T 是计算周期。

当 DT<120\*T 时，输出 Y 按输入 X 的序列，然后计算惯性滞后输出；

当 DT>120\*T 时，输出 Y 对输入 X 取几个周期的平均值形成序列，然后计算惯性滞后输出。

## 7.18 PTC 热力性质计算

### 名称

标记：PTC

名称：热力性质计算

### 结构

作用	标记	类型	缺省值	说明
输入	P	实数	0.0	压力
输入	T	实数	0.0	温度
输入	S	实数	0.0	焓值
输入	H	实数	0.0	焓值
输入	MODE	整数	0	计算方式



输出	Y	实数	0.0	计算结果
----	---	----	-----	------

## 描述

该指令根据设定的计算模式，进行热力性能中温度、压力、焓值、焓值之间关系的计算。

当 MODE=0 时，根据压力和温度，计算水或汽的焓值；

当 MODE=1 时，根据压力和温度，计算水或汽的比容；

当 MODE=2 时，根据压力和温度，计算水或汽的焓值；

当 MODE=3 时，根据压力和焓值，计算水或汽的温度；

当 MODE=4 时，根据压力和温度，计算水的焓值；

当 MODE=5 时，根据压力和温度，计算水的比容；

当 MODE=6 时，根据压力和温度，计算水的焓值；

当 MODE=7 时，根据焓值和温度，计算水或汽的压力；

当 MODE=8 时，根据压力和温度，计算汽的焓值；

当 MODE=9 时，根据压力和温度，计算汽的比容；

当 MODE=10 时，根据压力和温度，计算汽的焓值；

当 MODE=11 时，根据压力，计算水沸点温度；

当 MODE=12 时，根据压力和焓值，计算水或汽的温度；

当 MODE=13 时，根据焓值和温度，计算水或汽的压力；

当 MODE=14 时，根据温度，计算水沸点压力；

在以上各种计算模式中，对于不使用的输入信号，需要置为 0。

## 应用说明

在模式 1~3、7、12、13 中，指令首先根据温度或压力自动识别气态或液态，然后再按照水或汽进行热力性质的计算。

由于热力性质的计算量比较大，对于能够确定物质状态（液态或气态）的情况，尽量使用 4~6 或 8~10 的计算。

## 7.19 DFML 差分方程

### 名称

标记：DFML



名称：差分方程

## 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X	实数	0	输入信号
输入	A0	实数	0	滤波器系数
输入	B1	实数	0	滤波器系数
输入	A1	实数	0	滤波器系数
输入	B2	实数	0	滤波器系数
输入	A2	实数	0	滤波器系数
输入	B3	实数	0	滤波器系数
输入	A3	实数	0	滤波器系数
输入	B4	实数	0	滤波器系数
输入	A4	实数	0	滤波器系数
输入	B5	实数	0	滤波器系数
输入	A5	实数	0	滤波器系数
输入	B6	实数	0	滤波器系数
输入	A6	实数	0	滤波器系数
输出	Y	实数	0	滤波后的数值

## 描述

本指令进行差分方程的计算。

指令的表达式为：

$$Y_n = \sum_{k=0}^6 A_k X_{n-k} - \sum_{k=1}^6 B_k Y_{n-k}$$

## 应用

本指令应用于某些高级过程计算，用于实现更为一般化的方程计算。

有关差分方程可以参考工程数学、离散系统、数字信号处理等各种技术书籍。

## 7.20 FUZY 模糊控制器

## 名称



标记: FUZY

名称: 模糊控制器

## 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	ER	实数	0	偏差信号
输入	EV	实数	0	偏差变化速率
输入	YH	实数	100	输出上限
输入	YL	实数	0	输出下限
输入	TR	实数	0	跟踪量
输入	TS	布尔	0	跟踪开关
输入	K	实数	1	调节增益
输入	MODE	整数	0	工作模式
输入	ESEL	整数	0	偏差速率选择
输入	ER(-3)	实数	-40	偏差分割值, 负偏大值
输入	ER(-2)	实数	-20	偏差分割值, 负偏中值
输入	ER(-1)	实数	-10	偏差分割值, 负偏小值
输入	ER(0)	实数	0	偏差分割值, 偏差零值
输入	ER(1)	实数	10	偏差分割值, 正偏小值
输入	ER(2)	实数	20	偏差分割值, 正偏中值
输入	ER(3)	实数	40	偏差分割值, 正偏大值
输入	EV(-3)	实数	-10	速率分割值, 负速大值
输入	EV(-2)	实数	-5	速率分割值, 负速中值
输入	EV(-1)	实数	-1	速率分割值, 负速小值
输入	EV(0)	实数	0	速率分割值, 速率零值
输入	EV(1)	实数	1	速率分割值, 正速小值
输入	EV(2)	实数	5	速率分割值, 正速中值
输入	EV(3)	实数	10	速率分割值, 正速大值
输入	U(0,0)	实数	0	原点调整参数
输入	U(-1,0)	实数	-10	对应 ER(-1),EV(0)的控制值
输入	U(1,0)	实数	10	对应 ER(1),EV(0)的控制值
输入	U(0,-1)	实数	-1	对应 ER(0),EV(-1)的控制值
输入	U(0,1)	实数	1	对应 ER(0),EV(1)的控制值
输入	U(-2,0)	实数	-20	对应 ER(-2),EV(0)的控制值
输入	U(2,0)	实数	20	对应 ER(2),EV(0)的控制值
输入	U(0,-2)	实数	-2	对应 ER(0),EV(-2)的控制值
输入	U(0,2)	实数	2	对应 ER(0),EV(2)的控制值
输入	U(-3,0)	实数	-30	对应 ER(-3),EV(0)的控制值
输入	U(3,0)	实数	30	对应 ER(3),EV(0)的控制值



作用	标记	类型	缺省值	说明
输入	U(0,-3)	实数	-5	对应 ER(0),EV(-3)的控制值
输入	U(0,3)	实数	5	对应 ER(0),EV(3)的控制值
输出	Y	实数	0	控制输出量

## 描述

本指令按查表方式进行模糊控制，设定点以外的调整值按差值计算。

当 TS=1 时，处于跟踪状态，此时  $Y_n = TR_n$

当 TS=0 时，根据 MODE 的设置调节控制输出量。

MODE=0，进行增量型控制，此时  $Y_n = Y_{n-1} + K(U_n - U_{n-1})$

MODE=1，进行绝对型控制，此时  $Y_n = KU_n$

其中 U 是查表插值后的数值。

## 7.21 SALM 慢信号报警

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X	实数	0	输入信号
输入	H	实数	90	报警高限
输入	L	实数	10	报警低限
输入	DL	整数	1	报警延时：秒
输入	DB	实数	1	报警死区
输入	HH	实数	100	有效值高限
输入	LL	实数	0	有效值低限
输入	VU	实数	10	每秒上升速率
输入	VD	实数	10	每秒下降速率
输入	MUK	整数	0	解锁选择
输入	RST	布尔	0	闭锁复位
输出	ALM	布尔	0	报警输出
输出	QT	布尔	0	品质坏标志
输出	LK	布尔	0	报警闭锁标志



## 描述

用于温度等慢速信号的报警。

当信号 X 越过高限 H+DB 或低于低限 L-DB 时，处于越限状态。在进入越限状态后，当信号 X 低于高限 H-DB 且高于低限 L+DB 时，退出越限状态。

当进入越限状态且经过 DL（秒）延时后，报警输出 ALM 被置位。当退出越限状态后，报警输出 ALM 立刻被复位。信号 X 进入越限状态，且在延期内退出越限状态，报警输出 ALM 不发生变化。

但出现下列任一情况时，报警动作闭锁（失效），报警输出 ALM 不会置位，此时闭锁标志 LK 被置位：

1. 信号的品质坏，同时品质坏标志 QT 被置位；
2. 信号越过有效高低限 HH 或 LL；
3. 信号突然越过 H 和 L 的范围，即上升或下降的速度超过 VU 或 VD；

当上述情况消失后，根据 MUK 的选择，解除闭锁状态。若 MUK=0，则 X 进入 H 和 L 的区间，或 RST 置位，都可以解除闭锁；若 MUK=1，仅 RST 置位可以解除闭锁；若 MUK=2，仅 X 进入 H 和 L 的区间时解除闭锁，RST 无效；

如果引起报警失效的条件存在，则任何解锁动作无效。

## 7.22 ACCU 累积计算

## 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X	实数	0	输入信号
输入	HLM	实数	100000	累积高限值
输入	LSH	实数	0	小信号阈值
输入	PRE	实数	0	复位预置值
输入	RST	布尔	0	累积复位
输出	Y0	实数	0	当前累积值
输出	TS0	实数	0	当前有效累积时间，秒
输出	TB0	实数	0	当前无效累积时间，秒
输出	QH	布尔	0	累积达上限标志
输出	Y1	实数	0	上次累积值
输出	TS1	实数	0	上次有效累积时间，秒
输出	TB1	实数	0	上次无效累积时间，秒



## 描述

对输入信号 X 的有效值进行累积，累积结果输出在 Y0 端口，同时有效累积时间不断更新。

若 X 品质坏，则停止累积计算，同时无效累积时间 TB0 开始更新，直到输入信号恢复有效。

当输入信号 X 的数值小于 LSH 时，按 0 值累积，即 Y0 不变，TS0 不断更新。

当累积值 Y0 大于等于 HLM 时，到达上限标志 QH 被置位。累积过程会继续进行，直到复位信号 RST 被置位。

当 RST 信号置位时，当前的累积过程结束。当前的累积的 Y0、TS0 和 TB0 被移入 Y1、TS1 和 TB1。然后 TS0 和 TB0 被清零，Y0 赋予 PRE 的数值。

ACCU 指令具有自动断电保存功能，保存周期为 5 分钟。当控制器意外断电后，下次将会从最后保存点开始数值累积。

## 7.23 FLW 流量计算

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X	实数	0	输入信号
输入	FUN	整数	0	功能选择
输入	PR	实数	100	差压量程
输入	FM	实数	100	流量量程
输入	SHD	实数	1	小信号阈值%
输入	TX	实数	0	流体实际温度℃
输入	TS	实数	0	流体额定温度℃
输入	PX	实数	0	流体实际压力 MPa（表压）
输入	PS	实数	0	流体额定压力 Mpa（表压）
输入	P0	实数	0.1013	标准大气压 Mpa
输出	FL	实数	0	流量值
输出	F0	实数	0	标准流量

### 描述

进行流量计算，并具有小信号切除和温压补偿功能。



当 FUN=0 时，输入 X 作为差压信号，按公式  $F0 = FM \sqrt{\frac{X}{PS}}$  计算；

当 FUN=1 时，输入 X 作为差压量程的百分比，根据  $F0 = FM \cdot X\%$  计算；

当 FUN=2 时，输入 X 直接作为流量信号，即  $F0 = X$ 。

如果设置了小信号切除功能 (SHD>0)，当 F0 小于 FM\*SHD% 时，流量 FL=0。

否则根据标准流量 F0，进行后续温压补偿计算，

$$\text{流量值 } FL = F0 \sqrt{\frac{(TS + 273.15)(PX + P0)}{(TX + 273.15)(PS + P0)}}$$

此外，当输入 X 品质为坏点时，输出流量 FL=0，品质为坏点；若输入 X 品质为好点，但数值小于 0 时，输出流量 FL=0，品质为好点。



## 第8章 特殊指令

### 介绍

该类算法主要是涉及到特定功能运算的指令,可以利用指令构成特定的表达式或实现某种算法。该类指令输入输出的信号依据变量设置里的类型进行连接,不能连接其他类型的信号,否则会产生意料不到的后果。

该类指令包括：<sup>1</sup>

标记	名称
STS	信号状态（仅用于功能块图）
EQU	实数赋值
WEQU	整数赋值
EXEC	过程跳转

### 8.1 STS 信号状态

#### 名称

标记：STS

名称：信号状态

#### 结构

作用	标记	类型	缺省值	说明
输入	X	任何类型	无	信号输入
输入	STA	整数	0	信号状态选择
输出	Q	布尔	0	状态输出

#### 描述

将信号的状态根据 STA 的选择进行输出。

符号	作用	数值标记
品质	当 X 的品质好输出 0，品质坏输出 1	0

<sup>1</sup> 取消 SAVE 功能块，组态对象具有“动态保存”标志，一旦置上该标志，这个对象就可以被动态保存，不再需要使用 SAVE 功能块进行数据保存。



报警	当 X 有报警出 1，无报警出 0	1
3 级报警	当 X 处于超高限或超低限时出 1，否则出 0	2
2 级报警	当 X 处于高高限或低低限时出 1，否则出 0	3
1 级报警	当 X 处于高限或低限时出 1，否则出 0	4
报警确认	当 X 有报警且用户未确认时出 1，否则出 0	5

## 8.2 EQU 实数赋值

### 名称

标记: EQU

名称: 实数赋值

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	X	实数	0	输入变量
输出	Y	实数	0	输出变量

### 描述

该指令用来对实数进行赋值操作，而且其赋值与否可以通过使能端进行控制。当 EIN=1 时，进行赋值，当 EIN=0 时，停止赋值。

## 8.3 WEQU 整数赋值

### 名称

标记: WEQU

名称: 整数赋值

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能



输入	X	整数	0	输入变量
输出	Y	整数	0	输出变量

## 描述

该指令用来对整数进行赋值操作，而且其赋值与否可以通过使能端进行控制。当 EIN=1 时，进行赋值，当 EIN=0 时，停止赋值。

## 8.4 EXEC 过程跳转

### 名称

标记：EXEC

名称：过程跳转

### 结构

作用	标记	类型	缺省值	说明
输入	EIN	布尔	1	输入使能
输出	EOUT	布尔	1	输出使能
输入	ID	整数	1	过程编号

### 描述

通过该指令，可以调用其他的过程。

在输入有效的 ID 的情况下，且 EIN=1 时，进行过程的跳转执行，EIN=0 时，不执行指定的过程。