

31 | 指令

本章将介绍 GP-Pro EX 逻辑指令。并将详细描述可以在逻辑程序中使用的指令。

31.1 简介	31-2
31.2 指令符号列表.....	31-6
31.3 可设置为操作数的地址	31-35
31.4 指令步数.....	31-41
31.5 位指令	31-42
31.6 脉冲指令.....	31-53
31.7 程序控制指令.....	31-57
31.8 定时器指令	31-76
31.9 计数器指令	31-89
31.10 读 / 写指令	31-95
31.11 算术运算指令.....	31-103
31.12 时间运算指令.....	31-151
31.13 逻辑运算指令.....	31-159
31.14 传送指令	31-187
31.15 移位指令	31-216
31.16 循环移位指令.....	31-254
31.17 计算函数指令.....	31-292
31.18 三角函数指令.....	31-334
31.19 算术比较指令.....	31-384
31.20 时间比较指令.....	31-414
31.21 日期比较指令.....	31-438
31.22 数据转换指令.....	31-462
31.23 类型转换指令.....	31-498
31.24 I/O 驱动程序指令	31-552

31.1 简介

下表提供了一个逻辑程序可用的指令列表。支持逻辑程序的机型可以使用所有这些指令。这些指令可分为如下 9 个类别：(1) 基本，(2) 定时器，(3) 计数器，(4) 读 / 写，(5) 运算，(6) 函数，(7) 比较，(8) 转换，(9) I/O 驱动程序指令。

类别		指令名称	指令符号
基本指令	位基本	常开	NO
		常闭	NC
		输出	OUT
		负输出	OUTN
		置位	SET
		复位	RST
	脉冲基本	上升沿	PT
		下降沿	NT
	程序控制	跳转	JMP
		跳转至子程序	JSR
		返回	RET
		循环处理	FOR
			NEXT
		反转	INV
		退出	EXIT
		母线控制开始	PBC
		母线控制复位	PBR
		逻辑等待	LWA
定时器指令		ON 延时定时器	TON
		OFF 延时定时器	TOF
		脉冲定时器	TP
		累计 ON 延时定时器	TONA
		累计 OFF 延时定时器	TOFA
计数器指令		加计数器	CTU
		减计数器	CTD
		加 / 减计数器	CTUD
读 / 写指令	时间读 / 写	时间读取	JRD
		时间设置	JSET
	日期读 / 写	日期读取	NRD
		日期设置	NSET

类别		指令名称	指令符号
运算指令	算术运算	加	ADD
		减	SUB
		乘	MUL
		除	DIV
		取模	MOD
		加 1	INC
		减 1	DEC
	时间运算	时间加	JADD
		时间减	JSUB
	逻辑运算	逻辑与	AND
		逻辑或	OR
		逻辑异或	XOR
		逻辑非	NOT
	传送	传送 (复制)	MOV
		块传送 (块复制)	BLMV
		多点传送 (多点复制)	FLMV
		交换	XCH
	移位	左移	SHL
		右移	SHR
		算术左移	SAL
		算术右移	SAR
	循环移位	循环左移	ROL
		循环右移	ROR
		带进位循环左移	RCL
		带进位循环右移	RCR
函数指令	计算函数	求和	SUM
		平均值	AVE
		平方根	SQRT
		位计数	BCNT
		PID	PID
	三角函数	正弦	SIN
		余弦	COS
		正切	TAN
		反正弦	ASIN
		反余弦	ACOS
		反正切	ATAN
		余切	COT

类别		指令名称	指令符号
函数指令	其他函数	自然指数	EXP
		自然对数	LN
		常用对数	LG10
比较指令	算术比较	算术比较 (=)	EQ
		算术比较 (>)	GT
		算术比较 (<)	LT
		算术比较 (>=)	GE
		算术比较 (<=)	LE
		算术比较 (<>)	NE
	时间比较	时间比较 (=)	JEQ
		时间比较 (>)	JGT
		时间比较 (<)	JLT
		时间比较 (>=)	JGE
		时间比较 (<=)	JLE
		时间比较 (<>)	JNE
	日期比较	日期比较 (=)	NEQ
		日期比较 (>)	NGT
		日期比较 (<)	NLT
		日期比较 (>=)	NGE
		日期比较 (<=)	NLE
		日期比较 (<>)	NNE
转换指令	数据转换	BCD 转换	BCD
		BIN 转换	BIN
		编码	ENCO
		解码	DECO
		转换至弧度	RAD
		转换至角度	DEG
		缩放	SCL
	类型转换	整型至浮点型转换	I2F
		整型至实型转换	I2R
		浮点型至整型转换	F2I
		浮点型至实型转换	F2R
		实型至整型转换	R2I
		实型至浮点型转换	R2F
		时至秒转换	H2S
		秒至时转换	S2H

类别		指令名称	指令符号
I/O 驱动程序 指令	CAN	SDO 读取	SDOR
		SDO 写入	SDOW
		主站诊断	DGMT
		从站诊断	DGSL
	STD	更改脉冲输出参数	PLSX
		更改加速 / 减速脉冲输出参数	PLSY
		读取脉冲输出参数	PLSG
		启动脉冲输出	PLS
		停止脉冲输出	PLSQ
		更改 PWM 输出参数	PWMX
		读取 PWM 输出参数	PWMG
		启动 PWM 输出	PWM
		停止 PWM 输出	PWMQ
		更改高速计数器参数	HSCX
		更改高速计数器参数	HSCG
		启动高速计数器	HSC
		停止高速计数器	HSCQ
		确认脉冲捕捉输入	PCH
		清除脉冲捕捉输入	PCHQ

31.2 指令符号列表

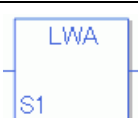
该列表分类显示指令的名称及符号。

重要

- 每个指令的步数取决于操作数的数据格式及是否使用了修饰符。
- 有关步数的详细信息，请参阅描述每条指令的章节。

31.2.1 基本指令

类别		指令名称	Pro EX 指令符号	指令步数	操作数 数量	梯形图符号
基本指令	位基本	常开	NO	1 至 5 步	1	
		常闭	NC	1 至 5 步	1	
		输出	OUT	1 至 5 步	1	
		负输出	OUTN	1 至 5 步	1	
		置位	SET	1 至 5 步	1	
		复位	RST	1 至 5 步	1	
	脉冲基本	上升沿	PT	2 至 5 步	1	
		下降沿	NT	2 至 5 步	1	
	程序控制	跳转	JMP	2 步		
		上升沿跳转	JMPP	2 至 5 步		
		跳转至子程序	JSR	2 步		
上升沿跳转至子程序		JSRP	2 步			
返回		RET	1 步			

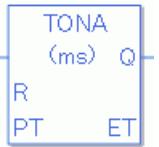
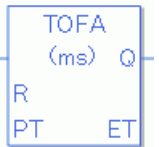
类别	指令名称	Pro EX 指令符号	指令步数	操作数 数量	梯形图符号
基本指令	程序控制	FOR	2 至 4 步	1	
		NEXT	1 步		
	反转	INV	1 步		
	退出	EXIT	1 步		
	母线控制	PBC	3 步	2	
		PBR	2 步	1	
	逻辑等待	LWA	2 步	1	

(注意)

要使用 1 步，步骤数必须小于清除位变量 (M 地址)+1536。如果根据清除位变量设置创建的位变量大于 1536 个，那么就有 2 步。

请在保持设置对话框中配置保持 / 非保持设置。

31.2.2 定时器指令

类别	指令名称	Pro EX 指令符号	指令步数	操作数 数量	梯形图符号
定时器指令	ON 延时定时器	TON	2 步	1	
	OFF 延时定时器	TOF	2 步	1	
	脉冲定时器	TP	2 步	1	
	累计 ON 延时定时器	TONA	2 步	1	
	累计 OFF 延时定时器	TOFA	2 步	1	

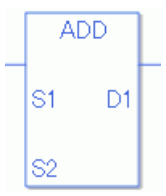
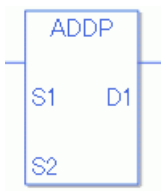
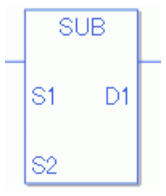
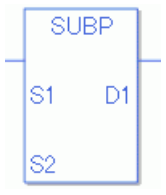
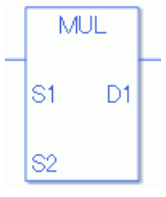
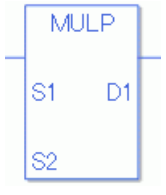
31.2.3 计数器指令

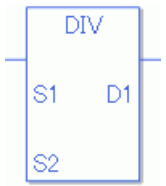
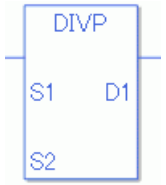
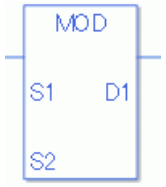
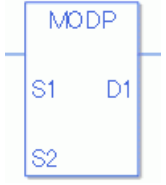
类别	指令名称	Pro EX 指令符号	指令步数	操作数 数量	触发条件	梯形图符号
计数器指令	加计数器	CTU	2 步	1	电平	
		CTUP	2 步	1	上升沿	
	减计数器	CTD	2 步	1	电平	
		CTDP	2 步	1	上升沿	
	加 / 减计数器	CTUD	2 步	1	电平	
		CTUDP	2 步	1	上升沿	

31.2.4 读 / 写指令

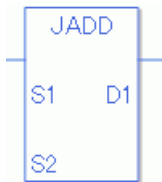
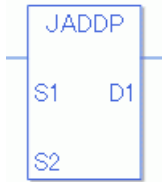
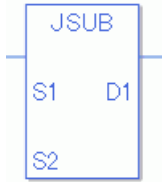
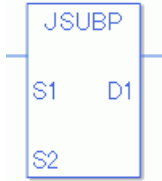
类别	指令名称	Pro EX 指令符号	指令步数	操作数 数量	触发条件	梯形图符号
读 / 写指令	时间读取	JRD	6 步	1	电平	
		JRDP	6 步	1	上升沿	
	时间设置	JSET	3 步	2	电平	
		JSETP	3 步	2	上升沿	
	日期读取	NRD	5 步	1	电平	
		NRDP	5 步	1	上升沿	
	日期设置	NSET	3 步	2	电平	
		NSETP	3 步	2	上升沿	

31.2.5 算术运算指令

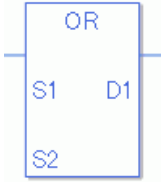
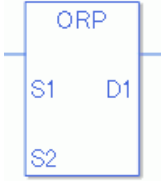
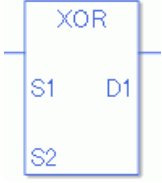
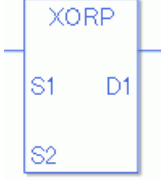
类别	指令名称	Pro EX 指令符号	指令步数	数量	触发条件	梯形图符号
运算指令	加	ADD	4 至 13 步	3	电平	
		ADDP	4 至 13 步	3	上升沿	
	减	SUB	4 至 13 步	3	电平	
		SUBP	4 至 13 步	3	上升沿	
	乘	MUL	4 至 13 步	3	电平	
		MULP	4 至 13 步	3	上升沿	

类别	指令名称	Pro EX 指令符号	指令步数	数量	触发条件	梯形图符号
运算指令	除	DIV	4 至 13 步	3	电平	
		DIVP	4 至 13 步	3	上升沿	
	取模 (%)	MOD	4 至 13 步	3	电平	
		MODP	4 至 13 步	3	上升沿	
	加 1	INC	2 至 4 步	1	电平	
		INCP	2 至 4 步	1	上升沿	
	减 1	DEC	2 至 4 步	1	电平	
		DECP	2 至 4 步	1	上升沿	
	算术运算					

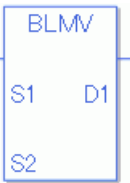
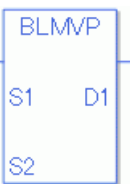
31.2.6 时间运算指令

类别		指令名称	Pro EX 指令符号	指令步数	数量	触发条件	梯形图符号
运算指令	时间运算	时间加	JADD	4 步	3	电平	
			JADDP	4 步	3	上升沿	
		时间减	JSUB	4 步	3	电平	
			JSUBP	4 步	3	上升沿	

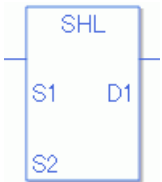
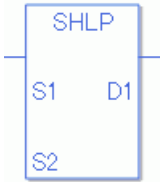
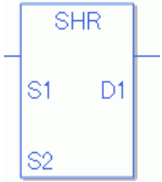
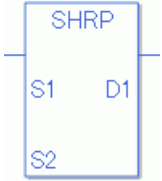
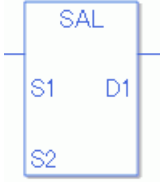
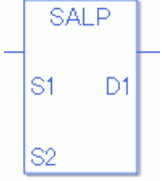
31.2.7 逻辑运算指令

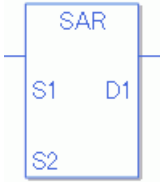
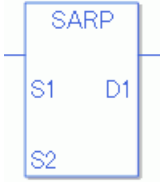
类别	指令名称	Pro EX 指令符号	指令步数	数量	触发条件	梯形图符号
运算指令	逻辑与	AND	4 至 13 步	3	电平	
		ANDP	4 至 13 步	3	上升沿	
	逻辑或	OR	4 至 13 步	3	电平	
		ORP	4 至 13 步	3	上升沿	
	逻辑异或	XOR	4 至 13 步	3	电平	
		XORP	4 至 13 步	3	上升沿	
	逻辑非	NOT	3 至 9 步	2	电平	
		NOTP	3 至 9 步	2	上升沿	

31.2.8 传送指令

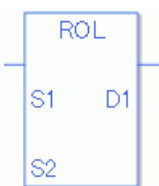
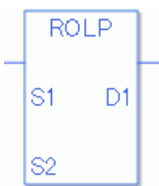
类别	指令名称	Pro EX 指令符号	指令步数	数量	触发条件	梯形图符号
运算指令	传送 (复制)	MOV	3 至 9 步	2	电平	
		MOVP	3 至 9 步	2	上升沿	
	块传送 (块复制)	BLMV	6 至 10 步	3	电平	
		BLMVP	6 至 10 步	3	上升沿	
	多点传送 (多点复制)	FLMV	4 至 10 步	3	电平	
		FLMVP	4 至 10 步	3	上升沿	
	交换	XCH	3 至 7 步	2	电平	
		XCHP	3 至 7 步	2	上升沿	

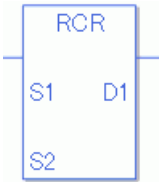
31.2.9 移位指令

类别	指令名称	Pro EX 指令符号	指令步数	数量	触发条件	梯形图符号
运算指令	左移	SHL	4 至 10 步	3	电平	
		SHLP	4 至 10 步	3	上升沿	
	右移	SHR	4 至 10 步	3	电平	
		SHRP	4 至 10 步	3	上升沿	
	算术左移	SAL	4 至 10 步	3	电平	
		SALP	4 至 10 步	3	上升沿	
	移位					

类别		指令名称	Pro EX 指令符号	指令步数	数量	触发条件	梯形图符号
运算指令	移位	算术右移	SAR	4 至 10 步	3	电平	
			SARP	4 至 10 步	3	上升沿	

31.2.10 循环移位指令

类别	指令名称	Pro EX 指令符号	指令步数	数量	触发条件	梯形图符号
运算指令	循环左移	ROL	4 至 10 步	3	电平	
		ROLP	4 至 10 步	3	上升沿	
	循环右移	ROR	4 至 10 步	3	电平	
		RORP	4 至 10 步	3	上升沿	
	带进位循环左移	RCL	4 至 10 步	3	电平	
		RCLP	4 至 10 步	3	上升沿	

类别		指令名称	Pro EX 指令符号	指令步数	数量	触发条件	梯形图符号
运算指令	循环移位	带进位循环右移	RCR	4 至 10 步	3	电平	
			RCRP	4 至 10 步	3	上升沿	

31.2.11 计算函数指令

类别	指令名称	Pro EX 指令符号	指令步数	数量	触发条件	梯形图符号
函数指令	求和	SUM	4 至 10 步	3	电平	
		SUMP	4 至 10 步	3	上升沿	
	平均值	AVE	4 至 10 步	3	电平	
		AVEP	4 至 10 步	3	上升沿	
	平方根	SQRT	3 至 7 步	2	电平	
		SQ RTP	3 至 7 步	2	上升沿	
	位计数	BCNT	3 至 9 步	2	电平	
		BCNTP	3 至 9 步	2	上升沿	

类别		指令名称	Pro EX 指令符号	指令步数	数量	触发条件	梯形图符号
函数指令	计算函数	PID	PID	10 至 18 步	5	电平	

31.2.12 三角函数指令

类别	指令名称	Pro EX 指令符号	指令步数	数量	触发条件	梯形图符号
函数指令	正弦	SIN	3 至 7 步	2	电平	
		SINP	3 至 7 步	2	上升沿	
	余弦	COS	3 至 7 步	2	电平	
		COSP	3 至 7 步	2	上升沿	
	正切	TAN	3 至 7 步	2	电平	
		TANP	3 至 7 步	2	上升沿	
	反正弦	ASIN	3 至 7 步	2	电平	
		ASINP	3 至 7 步	2	上升沿	
	三角函数					

类别		指令名称	Pro EX 指令符号	指令步数	数量	触发条件	梯形图符号
函数指令	三角函数	反余弦	ACOS	3 至 7 步	2	电平	
			ACOSP	3 至 7 步	2	上升沿	
		反正切	ATAN	3 至 7 步	2	电平	
			ATANP	3 至 7 步	2	上升沿	
		余切	COT	3 至 7 步	2	电平	
			COTP	3 至 7 步	2	上升沿	

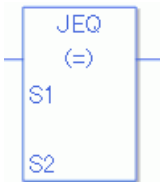
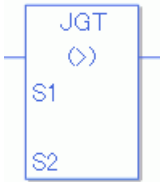
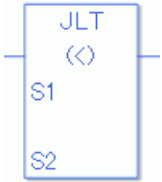
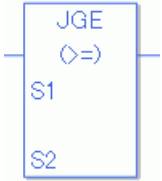
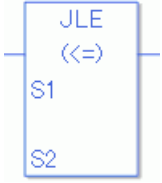
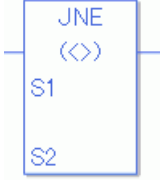
31.2.13 其他函数指令

类别	指令名称	Pro EX 指令符号	指令步数	数量	触发条件	梯形图符号
函数指令 其它函数	指数	EXP	3 至 7 步	2	电平	
		EXPP	3 至 7 步	2	上升沿	
	自然对数	LN	3 至 7 步	2	电平	
		LNP	3 至 7 步	2	上升沿	
	常用对数	LG10	3 至 7 步	2	电平	
		LG10P	3 至 7 步	2	上升沿	

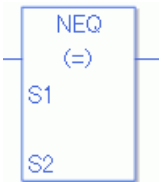
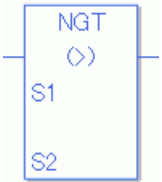
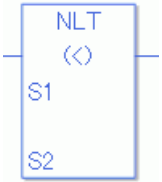
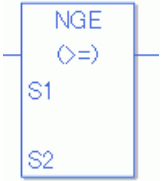
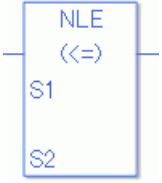
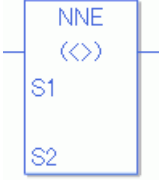
31.2.14 算术比较指令

类别	指令名称	Pro EX 指令符号	指令步数	数量	触发条件	梯形图符号
比较指令	算术比较 (=)	EQ	3 至 9 步	2	电平	
	算术比较 (>)	GT	3 至 9 步	2	电平	
	算术比较 (<)	LT	3 至 9 步	2	电平	
	算术比较 (>=)	GE	3 至 9 步	2	电平	
	算术比较 (<=)	LE	3 至 9 步	2	电平	
	算术比较 (<>)	NE	3 至 9 步	2	电平	

31.2.15 时间比较指令

类别	指令名称	Pro EX 指令符号	指令步数	数量	触发条件	梯形图符号
比较指令	时间比较 (=)	JEQ	3 步	2	电平	
	时间比较 (>)	JGT	3 步	2	电平	
	时间比较 (<)	JLT	3 步	2	电平	
	时间比较 (>=)	JGE	3 步	2	电平	
	时间比较 (<=)	JLE	3 步	2	电平	
	时间比较 (<>)	JNE	3 步	2	电平	

31.2.16 日期比较指令

类别	指令名称	Pro EX 指令符号	指令步数	数量	触发条件	梯形图符号
比较指令	日期比较 (=)	NEQ	3 步	2	电平	
	日期比较 (>)	NGT	3 步	2	电平	
	日期比较 (<)	NLT	3 步	2	电平	
	日期比较 (>=)	NGE	3 步	2	电平	
	日期比较 (<=)	NLE	3 步	2	电平	
	日期比较 (<>)	NNE	3 步	2	电平	

31.2.17 数据转换指令

类别	指令名称	Pro EX 指令符号	指令步数	数量	触发条件	梯形图符号
转换指令	BCD 转换	BCD	3 至 7 步	2	电平	
		BCDP	3 至 7 步	2	上升沿	
	BIN 转换	BIN	3 至 7 步	2	电平	
		BINP	3 至 7 步	2	上升沿	
	编码	ENCO	3 至 7 步	2	电平	
		ENCOP	3 至 7 步	2	上升沿	
	解码	DECO	3 至 7 步	2	电平	
		DECOP	3 至 7 步	2	上升沿	
	转换至弧度	RAD	3 至 7 步	2	电平	
		RADP	3 至 7 步	2	上升沿	

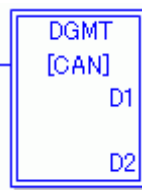
类别		指令名称	Pro EX 指令符号	指令步数	数量	触发条件	梯形图符号
转换指令	数据转换	转换至角度	DEG	3 至 7 步	2	电平	
			DEGP	3 至 7 步	2	上升沿	
		缩放	SCL	7 至 11 步	2	电平	
			SCLP	7 至 11 步	2	上升沿	

31.2.18 类型转换指令

类别	指令名称	Pro EX 指令符号	指令步数	数量	触发条件	梯形图符号
转换指令	整型至浮点型转换	I2F	3 至 7 步	2	电平	
		I2FP	3 至 7 步	2	上升沿	
	整型至实型转换	I2R	3 至 7 步	2	电平	
		I2RP	3 至 7 步	2	上升沿	
	浮点型至整型转换	F2I	3 至 7 步	2	电平	
		F2IP	3 至 7 步	2	上升沿	

类别	指令名称	Pro EX 指令符号	指令步数	数量	触发条件	梯形图符号
转换指令	浮点型至 实型转换	F2R	3 至 7 步	2	电平	
		F2RP	3 至 7 步	2	上升沿	
	实型至整 型转换	R2I	3 至 7 步	2	电平	
		R2IP	3 至 7 步	2	上升沿	
	实型至浮 点型转换	R2F	3 至 7 步	2	电平	
		R2FP	3 至 7 步	2	上升沿	
	时至秒转 换	H2S	3 至 5 步	2	电平	
		H2SP	3 至 5 步	2	上升沿	
	秒至时转 换	S2H	3 至 5 步	2	电平	
		S2HP	3 至 5 步	2	上升沿	

31.2.19 I/O 驱动程序指令

类别		指令名称	Pro EX 指令符号	指令步数	数量	梯形图符号
I/O 驱动程序指令	CAN	SDO 读取	SDOR	9 至 21 步	6	
		SDO 写入	SDOW	9 至 21 步	6	
		主站诊断	DGMT	7 至 15 步	4	
		从站诊断	DGSL	5 至 9 步	2	
	STD	更改脉冲输出参数	PLSX			
		更改加速 / 减速脉冲输出参数	PLSY			

类别		指令名称	Pro EX 指令符号	指令步数	数量	梯形图符号
I/O 驱动程序指令	STD	读取脉冲输出参数	PLSG			
		启动脉冲输出	PLS			
		停止脉冲输出	PLSQ			
		更改 PWM 输出参数	PWMX			
		读取 PWM 输出参数	PWMG			
		启动 PWM 输出	PWM			
		停止 PWM 输出	PWMQ			
		更改高速计数器参数	HSCX			
		读取高速计数器参数	HSCG			

类别		指令名称	Pro EX 指令符号	指令步数	数量	梯形图符号
I/O 驱动程序指令	STD	启动高速计数器	HSC			
		停止高速计数器	HSCQ			
		确认脉冲捕捉输入	PCH			
		清除脉冲捕捉输入	PCHQ			

31.3 可设置为操作数的地址

概括介绍可将其设置为指令操作数的符号变量、外接控制器地址和常量。
由于可配置的内容根据指令的不同而有所不同，因此请参阅各条指令的描述。

31.3.1 外接控制器地址

在通讯设置中为外接控制器指定的地址。

名称	类型	示例	描述
外部控制器 /PLC	位	[PLC1]X0000	在通讯设置中指定的通讯设备地址的位地址。
	字	[PLC1]D0000	在通讯设置中指定的外接控制器地址的字地址。

31.3.2 符号

该功能将外接控制器中的地址改为用户易于理解的名称。请务必使外接控制器的地址与它们各自的名称一一对应。

例如，要为三菱 PLC 上的寄存器地址“X0000”分配名称“RUN”，应定义“RUN”和“X0000”。

名称	类型	示例	描述
符号	位	RUN = X0000	这是一个在符号变量列表中配置的位符号，并由外接控制器地址和任意名称进行定义。
	字	Data = D0000	这是一个在符号变量列表中配置的字符符号，并由外接控制器地址和任意名称进行定义。

31.3.3 LS 区

这是 GP 中内部存储区的地址。请注意，地址的指定方法取决于通讯设置。

名称	类型	示例	描述
内部存储器	位	[#INTERNAL]LS010000	GP 内部存储器的位指定
	字	[#INTERNAL]LS0100	GP 内部存储器的字指定
Memory Link 设置	位	[#MEMLINK]010000	GP 内部存储器的位指定
	字	[#MEMLINK]0100	GP 内部存储器的字指定

注 释

- LS 区和 USR 区的字是 16 位，但是逻辑程序处理时将其视作 32 位。然而，当在显示器部件和其他部件中使用字的高 16 位被忽略。

31.3.4 USR 区

这是 GP 的内部存储区。可以使用任何指定方法。

可从 0 到 29999 进行寻址。

名称	类型	示例	描述
USR 区	位	[#INTERNAL]USR0010000	GP 内部存储器的位指定
	字	[#INTERNAL]USR00100	GP 内部存储器的字指定

注 释

- LS 区和 USR 区的字是 16 位，但是逻辑程序处理时将其视作 32 位。然而，当在显示器部件和其他部件中使用字的高 16 位被忽略。

31.3.5 系统变量

这是 GP 的系统数据区。可以使用任何外接控制器设置。

注 释

- 逻辑程序中使用的有些系统变量只有在启用逻辑程序时才起作用。因此需要注意是否将逻辑程序选为 [禁用]、或使用的是 #L**** 逻辑变量。

名称	类型	示例	描述
系统变量	位	#L_Clock100ms	GP 系统变量位类型
		#L_Clock1sec	
	整型	#L_ScanTime	GP 系统变量整数类型

31.3.6 变量

变量适用于所有 GP 机型。您可以在不知道控制器地址的情况下使用变量。变量可以含有修饰符 (*1)，也可以是数组形式 (*2)。使用修饰符可以访问整型变量中的单个位或字节。

名称	类型	示例	描述
变量	位	任意名称	位型变量。允许使用数组。
	整型	"	整型变量。允许使用数组和修饰符。
	浮点型	"	32 位浮点型变量。允许使用数组。
	实型	"	64 位实型变量。允许使用数组。
	定时器	"	定时器变量。结构 *3 变量
	计数器	"	计数器变量。结构 *3 变量
	日期	"	日期变量。结构 *3 变量
	时间	"	时间变量。结构 *3 变量
	PID	"	PID 变量。结构 *3 变量。

*1 您可以使用三种不同类型的修饰符：位修饰符、字节修饰符和字修饰符。只有整型变量支持修饰符。

指定方法：位变量名 .X[0]，字节变量名 .B[0]，字变量名 .W[0]

*2 可以使用具有以下变量类型的数组指定连续存储地址：位、整型、浮点型和实型。
指定方法：变量名 [10]

*3 多个变量组合在一起就是结构。结构变量包括：定时器、计数器、时间、日期和 PID。

■ 结构变量

定时器变量

定时器变量	变量设置	描述
变量名 .TI	位变量	当定时器开始计时时置 ON。
变量名 .Q	位变量	当定时器结束计时时置 ON。
变量名 .R	位变量	复位定时器上的当前值。清除 (0)。
变量名 .PT	整型变量	定时器上的设定值。
变量名 .ET	整型变量	定时器上的当前值。

计数器变量

计数器变量	变量设置	描述
变量名 .R	位变量	复位当前值。清除 (0)。
变量名 .Q	位变量	在当前值达到预设值时置 ON。
变量名 .UP	位变量	当加计数时置 ON。
变量名 .QU	位变量	对于加 / 减计数器，在当前值达到预设值时置 ON。
变量名 .QD	位变量	对于加 / 减计数器，在当前值达到 0 或更小时置 ON。
变量名 .PV	整型变量	计数器设定值。
变量名 .CV	整型变量	计数器上的当前值。

时间变量

时间变量	变量设置	描述
变量名 .HR	整型变量	以 BCD 形式输入小时。
变量名 .MIN	整型变量	以 BCD 形式输入分钟。
变量名 .SEC	整型变量	以 BCD 形式输入秒。

日期变量

日期变量	变量设置	描述
变量名 .YR	整型变量	以 BCD 形式输入年份。
变量名 .MO	整型变量	以 BCD 形式输入月份
变量名 .DAY	整型变量	以 BCD 形式输入日期

PID 变量

PID 变量	变量设置	描述
变量名 .Q	位变量	PID 指令处理完成标志
变量名 .PF	位变量	处理无效范围标志
变量名 .UO	位变量	输出值超出上限
变量名 .TO	位变量	输出值超出下限
变量名 .IF	位变量	积分设置
变量名 .KP	整型变量	比例系数
变量名 .TR	整型变量	积分时间
变量名 .TD	整型变量	一次微分时间
变量名 .PA	整型变量	PID 处理无效范围
变量名 .BA	整型变量	偏差 (偏移)
变量名 .ST	整型变量	采样频率

■ 当定义常量为操作数时

当输入常量值时，按如下所示输入。

实型常量	在将值与实型变量关联起来时使用。 输入格式 0r(零和小写字母 “r”), 例如, 0r0.11
浮点型常量	在将值与浮点型变量关联起来时使用。 输入格式 0f(零和小写字母 “f”), 例如, 0f0.11
常量 HEX 输入	当在整型变量中输入十六进制值时使用。 输入格式 0x(零和小写字母 “x”), 例如, 0x0.11

注 释

- 当小数部分超出 4 位时，使用如下符号。
例如, 0f0.00001 -> 0f1e-05
0f0.000001 -> 0f1e-06

31.3.7 使用地址格式时的逻辑寄存器

如果将逻辑设置为地址格式，可以使用如下寄存器。

名称	类型	名称	描述
逻辑	位	X_ /Y_ /M_	位型逻辑地址
	整型	D_ /I_ /Q_	整型逻辑地址。同变量一样，可以使用修饰符。
	浮点型	F_	浮点型逻辑地址。
	实型	R_	实型逻辑地址
	定时器	T_	定时器型逻辑地址。它是一个结构，同变量一样。
	计数器	C_	计数器型逻辑地址。它是一个结构，同变量一样。
	日期	N_	日期型逻辑地址。它是一个结构，同变量一样。
	时间	J_	定时器型逻辑地址。它是一个结构，同变量一样。
	PID	U_	PID 型逻辑地址。它是一个结构，同变量一样。

31.4 指令步数

描述每条指令的步数换算。(有关各条指令的步数的详细信息, 请参阅相关指令的描述。)

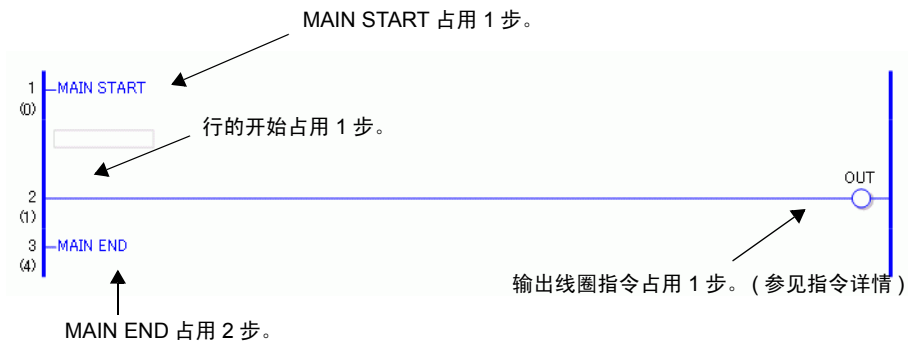
如下程序仅使用了状态一直为 ON 的输出线圈 OUT。

变量 OUT 的定义

变量名 out

保持设置 非保持

数组元素 无



总共占用 5 步。

对 1 步指令来说, 行号下显示的步数和实际的步数可能不同, 因为 1 步指令在保存和错误检查时得到了优化。

31.5 位指令

31.5.1 NO(常开)/NC(常闭)

符号和功能

梯形图指令名称	梯形图符号	功能	指令步数
NO (常开)	S1 	输入	1 至 5
梯形图指令名称	梯形图符号	功能	指令步数
NC (常闭)	S1 	输入	1 至 5

■ 操作数设置

下面描述了操作数 (S1) 的可指定内容。

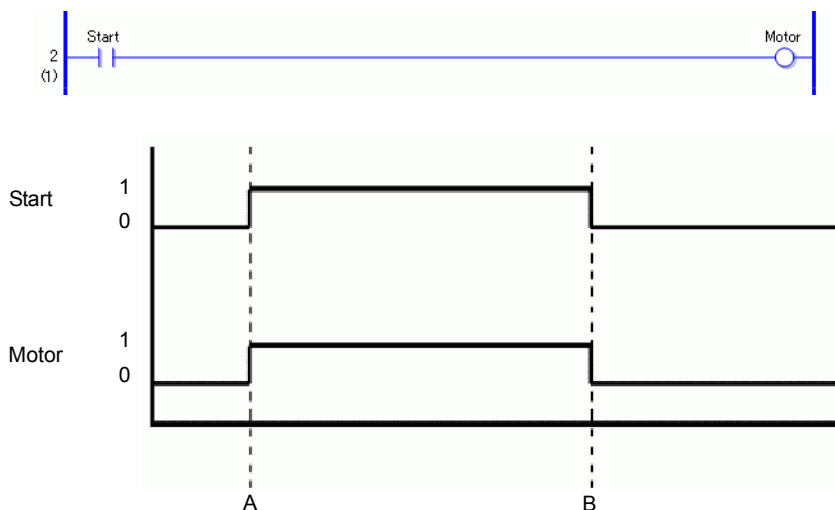
名称	类型	条件	指令步数	可用: O 不可用: X
外部寄存器地址	位		2	O
	字	指定字中的一个位。 (例如, [PLC1]D0000.00)	3	O
内部地址	位		2	O
	字	指定字中的一个位。 (例如, [#INTERNAL]LS000000)	3	O
符号	位		2	O
	字			X
变量格式	位	未指定数组。 非保持输入和输出数量最多为 1536。	1	O
		未指定数组。 非保持 (超过 1537) 或保持	2	O
		指定位数组 ([常量])	3	O
		指定位数组 ([变量])	4	O
	整型	未指定数组和修饰符		X
		指定整型变量 .X[常量]	3	O
		指定整型变量 .X[变量]	4	O
		指定整型变量 [常量 / 变量].X[常量 / 变量]	5	O
	浮点型			X
	实型			X
	定时器	仅 .Q / .TI / .R	3	O
	计数器	仅 .R / .UP / .QU / .QD / .Q	3	O

名称	类型	条件	指令步数	可用：○ 不可用：X
变量格式	日期			X
	时间			X
	PID	仅 .Q / .UO / .TO / .PF / .IF	3	○
地址格式	X_		1	○
	Y_		1	○
	M_	在清零类型范围内 (M_0000 至 M_1535)	1	○
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.X[常量]	3	○
		D_****.X[地址]	4	○
	F_			X
	R_			X
	T_	仅 .Q / .TI / .R	3	○
	C_	仅 .R / .UP / .QU / .QD / .Q	3	○
	N_			X
	J_			X
	U_	仅 .Q / .UO / .TO / .PF / .IF	3	○

■ NO 指令说明

- 使用 NO 指令来决定 ON 或 OFF 状态。该指令可以用来决定外部输入或内部线圈的 ON 或 OFF 状态。
- 使用 NO 指令时，NO 指令与右端母线之间必须包含其他指令。其他指令可以是一个输出指令或除输入以外的任何指令。

程序示例

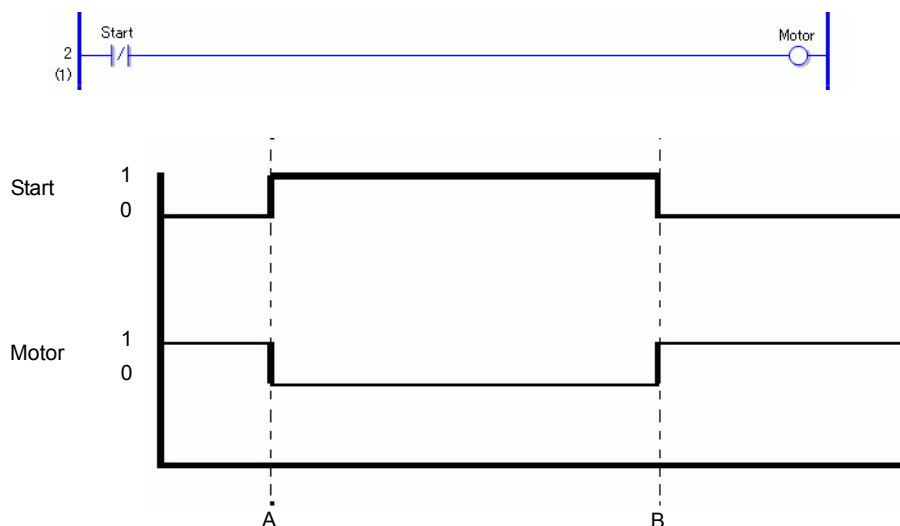


- 点 A 当位变量 Start 置 ON 时，NO 指令闭合。位变量 Motor 置 ON。
- 点 B 当位变量 Start 置 OFF 时，NO 指令断开。位变量 Motor 置 OFF。

■ NC 指令说明

- 使用 NC 指令来决定 ON 或 OFF 状态。该指令可以用来决定外部输入或内部线圈的 ON 或 OFF 状态。
- 其他指令可以是一个输出指令或除输入以外的任何指令。NC 指令与右侧母线之间必须要有其他指令。

程序示例



点 A 当位变量 Start 置 ON 时，NC 指令断开。位变量 Motor 置 OFF。

点 B 当位变量 Start 置 OFF 时，NC 指令闭合。位变量 Motor 置 ON。

注意：要在电源关闭时保持状态，请将符号变量设置为保持变量。地址格式使用保持地址。（外部输入和输出不能使用保持设置。）

31.5.2 OUT(输出线圈)/OUTN(负输出线圈)

符号和功能

梯形图指令名称	梯形图符号	功能	指令步数
OUT (输出线圈)	 D1	输出	1 至 5
梯形图指令名称	梯形图符号	功能	指令步数
OUTN (负输出线圈)	 D1	输出	1 至 5

■ 操作数设置

下面描述了操作数 (D1) 的可指定内容。

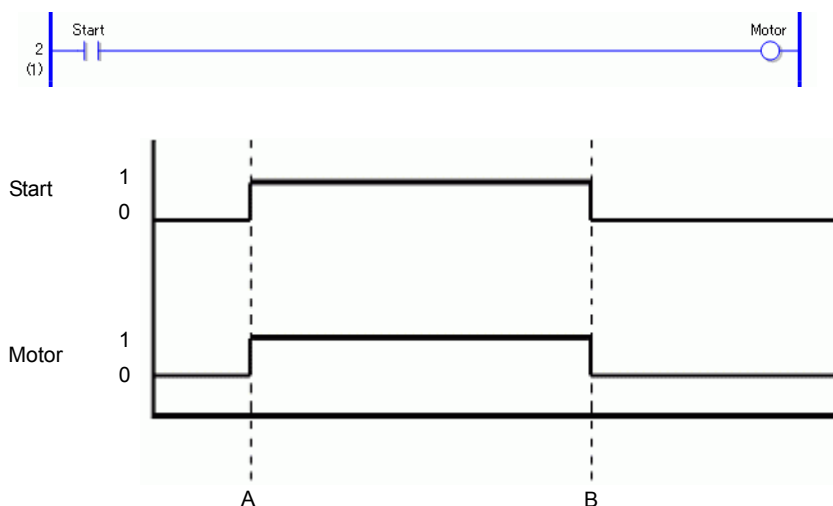
名称	类型	条件	指令步数	可用: ○ 不可用: X
外部寄存器地址	位		2	○
	字	指定字中的一个位。 (例如, [PLC1]D0000.00)	3	○
内部地址	位		2	○
	字	指定字中的一个位。 (例如, [#INTERNAL]LS000000)	3	○
符号	位		2	○
	字			X
变量格式	位	未指定数组。 非保持输出的数量最多为 1536。	1	○
		未指定数组。 非保持 (超过 1537) 或保持	2	○
		指定位数组 ([常量])	3	○
		指定位数组 ([变量])	4	○
	整型	未指定数组和修饰符		X
		指定整型变量 .X[常量]	3	○
		指定整型变量 .X[变量]	4	○
		指定整型变量 [常量 / 变量].X[常量 / 变量]	5	○
	浮点型			X
	实型			X
	定时器	仅 .Q / .TI / .R	3	○
	计数器	仅 .R / .UP / .QU / .QD / .Q	3	○
	日期			X
	时间			X
	PID	仅 .Q / .UO / .TO / .PF / .IF	3	○

名称	类型	条件	指令步数	可用：○ 不可用：X
地址格式	X_			X
	Y_		1	○
	M_	在清零类型范围内 (M_0000 至 M_1535)	1	○
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.X[常量]	3	○
		D_****.X[地址]	4	○
	F_			X
	R_			X
	T_	仅 .Q / .TI / .R	3	○
	C_	仅 .R / .UP / .QU / .QD / .Q	3	○
	N_			X
	J_			X
	U_	仅 .Q / .UO / .TO / .PF / .IF	3	○

■ OUT 指令说明

- OUT 指令用于输出一个 ON 或 OFF 结果。SET 和 RST 指令用于将外部输出或内部线圈置 ON 或 OFF。
- 一行中只能使用一个 OUT 指令。如果使用了分支指令，就可以使用多个 OUT 指令。
- 须将 OUT 指令放在紧邻右端母线的左边。

程序示例

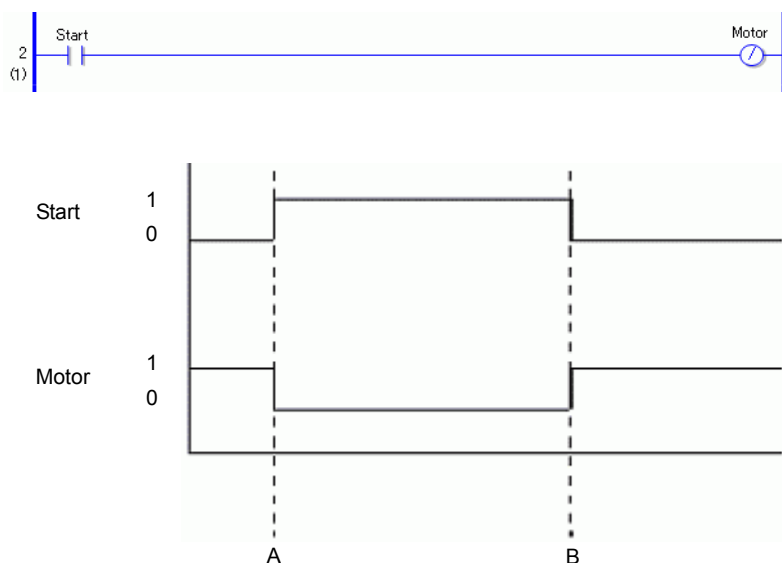


- A 点 当位变量 Start 置 ON 时，OUT 指令的位变量 Motor 置 ON。
- B 点 当位变量 Start 置 OFF 时，OUT 指令的位变量 Motor 置 OFF。

■ OUTN 指令说明

- OUTN 指令用于反转输出一个 ON 或 OFF 结果。SET 和 RST 指令用于将外部输出或内部线圈置 ON 或 OFF。
- 一行中只能使用一个 OUTN 指令。如果使用了分支指令，就可以使用多个 OUT 指令。
- 须将 OUTN 指令放在紧邻右端母线的左边。

程序示例

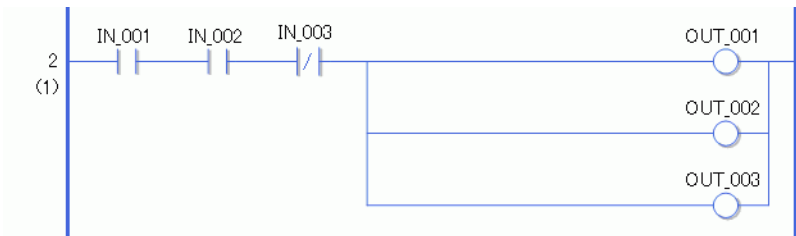


A 点 当位变量 Start 置 ON 时，OUTN 指令的位变量 Motor 置 OFF。

B 点 当位变量 Start 置 OFF 时，OUTN 指令的位变量 Motor 置 OFF。

注意：要在电源关闭时保持状态，请将符号变量设置为保持变量。地址格式使用保持地址。（外部输入和输出不能使用保持设置。）

当使用多个 OUT 和 OUTN 指令时



上面的例子显示了如何通过分支 OUT 指令来使用多个 OUT 指令。如果将 OUT_001 和 OUT_002 串联放置，就会发生错误。

31.5.3 SET(置位线圈)/RST(复位线圈)

符号和功能

梯形图指令名称	梯形图符号	功能	指令步数
SET (置位线圈)		输出	1 至 5
梯形图指令名称	梯形图符号	功能	指令步数
RST (复位线圈)		输出	1 至 5

■ 操作数设置

下面描述了操作数 (D1) 的可指定内容。

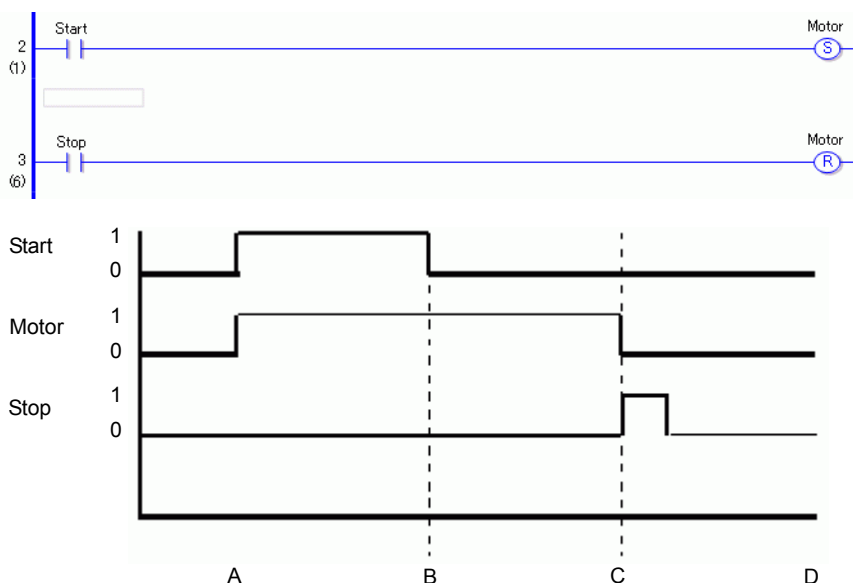
名称	类型	条件	指令步数	可用: ○ 不可用: X
外部寄存器地址	位		2	○
	字	指定字中的一个位。 (例如, [PLC1]D0000.00)	3	○
内部地址	位		2	○
	字	指定字中的一个位。 (例如, [#INTERNAL]LS000000)	3	○
符号	位		2	○
	字			X
变量格式	位	未指定数组。 非保持输出的数量最多为 1536。	1	○
		未指定数组。 非保持 (超过 1537) 或保持	2	○
		指定位数组 ([常量])	3	○
		指定位数组 ([变量])	4	○
	整型	未指定数组和修饰符		X
		指定整型变量 .X[常量]	3	○
		指定整型变量 .X[变量]	4	○
		指定整型变量 [常量 / 变量].X[常量 / 变量]	5	○
	浮点型			X
	实型			X
	定时器	仅 .Q / .TI / .R	3	○
	计数器	仅 .R / .UP / .QU / .QD / .Q	3	○
	日期			X
	时间			X
	PID	仅 .Q / .UO / .TO / .PF / .IF	3	○

名称	类型	条件	指令步数	可用：○ 不可用：X
地址格式	X_			X
	Y_		1	○
	M_	在清零类型范围内 (M_0000 至 M_1535)	1	○
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.X[常量]	3	○
		D_****.X[地址]	4	○
	F_			X
	R_			X
	T_	仅 .Q / .TI / .R	3	○
	C_	仅 .R / .UP / .QU / .QD / .Q	3	○
	N_			X
	J_			X
	U_	仅 .Q / .UO / .TO / .PF / .IF	3	○

■ SET 和 RST 指令说明

- SET 指令保持 ON 状态，无论输入状态如何。
- RST 指令保持 OFF 状态，无论输入状态如何。
- SET 和 RST 指令用于将外部输出或内部线圈置 ON 或 OFF。
- 一行中只能使用一个 OUT 指令。如果使用了分支指令，就可以使用多个 OUT 指令。

程序示例



- A 点 当位变量 Start 置 ON 时，SET 指令执行，然后位变量 Motor 置 ON。
- B 点 位变量 Start 置 OFF，但是，位变量 Motor 仍保持 ON 状态。
- C 点 位变量 Stop 置 ON，RST 指令执行。然后，位变量 Motor 置 ON。
当 RST 指令将位变量 Motor 置 ON 时，状态被清除，位变量 Motor 从 ON 变为 OFF。
- D 点 位变量 Motor 保持 OFF 状态，直到位变量 Start 置 ON。

31.6 脉冲指令

31.6.1 PT(上升沿)/NT(下降沿)

符号和功能

梯形图指令名称	梯形图符号	功能	指令步数
PT (上升沿)	S1 	输入	2 至 5
梯形图指令名称	梯形图符号	功能	指令步数
NT (下降沿)	S1 	输入	2 至 5

■ 操作数设置

下面描述了操作数 (S1) 的可指定内容。

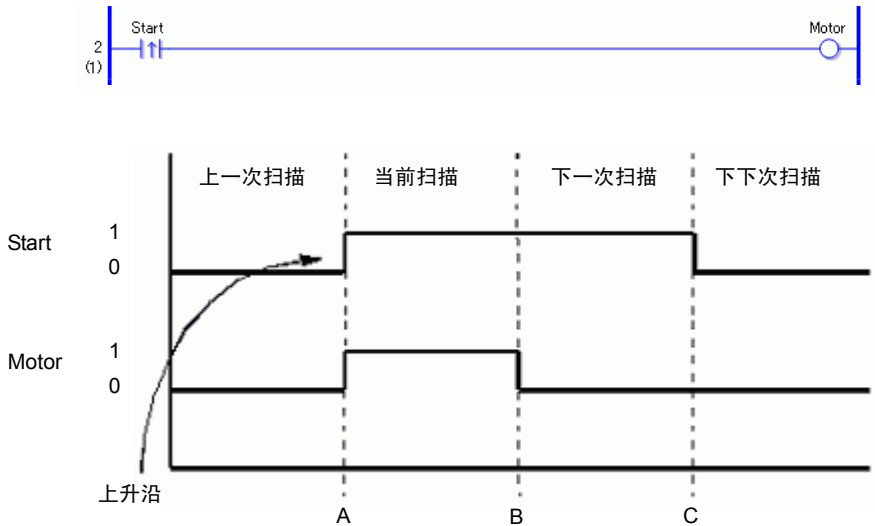
名称	类型	条件	指令步数	可用: ○ 不可用: X
外部寄存器地址	位		2	○
	字	指定字中的一个位。 (例如, [PLC1]D0000.00)	3	○
内部地址	位		2	○
	字	指定字中的一个位。 (例如, [#INTERNAL]LS000000)	3	○
符号	位		2	○
	字			X
变量格式	位	指定一个位	2	○
		指定位数组 ([常量])	3	○
		指定位数组 ([变量])	4	○
	整型	未指定数组和修饰符		X
		指定整型变量 .X[常量]	3	○
		指定整型变量 .X[变量]	4	○
		指定整型变量 [常量 / 变量].X[常量 / 变量]	5	○
	浮点型			X
	实型			X
	定时器	仅 .Q / .TI / .R	3	○
	计数器	仅 .R / .UP / .QU / .QD / .Q	3	○
	日期			X
	时间			X
	PID	仅 .Q / .UO / .TO / .PF / .IF	3	○

名称	类型	条件	指令步数	可用：○ 不可用：X
地址格式	X_		2	○
	Y_		2	○
	M_		2	○
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.X[常量]	3	○
		D_****.X[地址]	4	○
	F_			X
	R_			X
	T_	仅 .Q / .TI / .R	3	○
	C_	仅 .R / .UP / .QU / .QD / .Q	3	○
	N_			X
	J_			X
	U_	仅 .Q / .UO / .TO / .PF / .IF	3	○

■ 上升沿 (PT) 指令说明

- 当 PT 指令位变量置 ON 时，仅第一个扫描为 ON 状态。后续扫描将为 OFF 状态，即使位变量为 ON 状态。您可以用 PT 指令统计 ON 状态的数量。
- 使用 NO 指令时，NO 指令与右端母线之间必须包含其他指令。其他指令可以是一个输出指令或除输入以外的任何指令。

程序示例

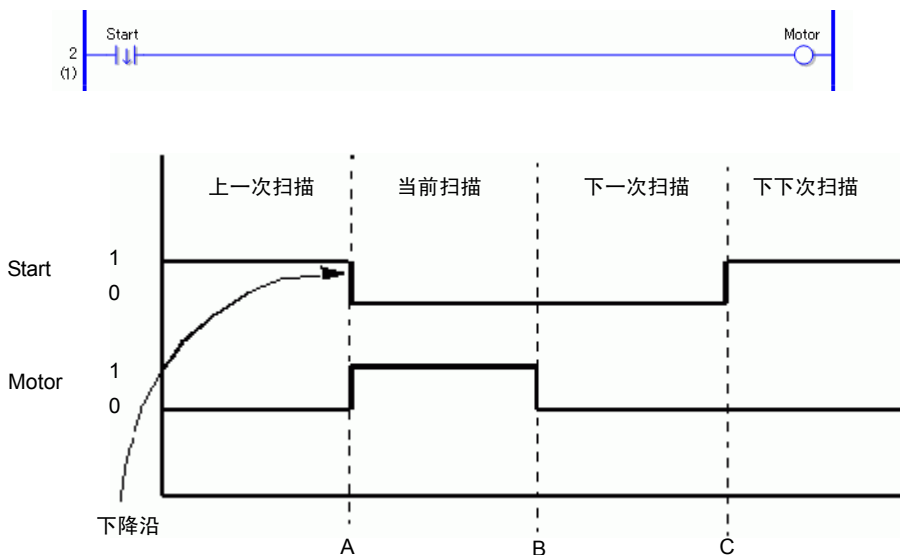


- A 点 变量 Start 置 ON，然后变量 Motor 置 ON。
B 点 在执行完一次扫描后，变量 Motor 置 OFF。
C 点 变量 Motor 保持 OFF，因为未检测到变量 Start 的下降沿。

■ 下降沿 (NT) 指令说明

- 当执行 NT 指令时，如果在上一次扫描过程中为 ON 的变量在当前扫描中变为 OFF，只在当前扫描中执行 NT 指令。NT 指令不能在初始扫描时执行，因为上一次扫描的状态总被视为 OFF。因此，在初始扫描时，即使执行了 NT 指令它也不起作用。下例描述了 NT 指令的功能。

程序示例



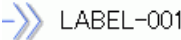
- A 点 变量 Start 置 OFF，然后变量 Motor 置 ON。
 B 点 在执行完一次扫描后，变量 Motor 将置 OFF。
 C 点 变量 Motor 保持 OFF，因为未检测到变量 Start 的下降沿。

(补充) 对于上升沿和下降沿指令的操作数，当执行每个元件的间接寻址时，尤其是当使用变量指定数组或位时，您必须要多加注意。在执行指令前，会将上一次执行的操作数中的变量与当前执行的操作数中的变量进行比较。因此，如果指定的变量值不同，比较目标也不同。

31.7 程序控制指令

31.7.1 JMP(跳转)/JMPP(上升沿跳转)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
JMP (跳转)	 LABEL-001	控制	2
梯形图指令名称	梯形图符号	功能	步数
JMPP (下降沿跳转)	 LABEL-001	控制	2

最多可为每个 JMP 指令指定 192 个标签。为 JMP 目标指定标签时，以前指定的标签名称将显示出来。如果未定义标签，则不显示标签名称。请首先插入标签，然后为跳转指令指定标签。

■ 指定标签



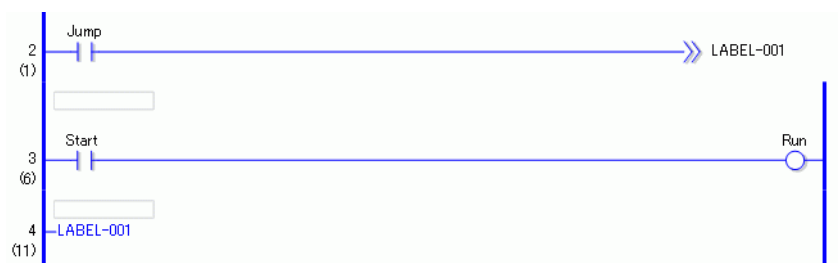
右键点击，选择 [插入标签]，或在 [逻辑] 菜单上点击 [插入标签]。
可以从 LABEL-001 至 LABEL-192 的 192 个标签范围内选择一个标签。
不能任意指定标签名称。



只显示程序中使用的标签。在 INIT、MAIN 和 SUB 画面上不能使用相同的标签名称。

执行 JMP 指令时，程序将跳转至指定标签。与 JSR 指令不同，程序不会自动返回到跳转行。不能跳过 INIT 或 SUB 块。请在一个块内创建跳转至标签的程序。此外，还需注意的是，如果程序跳转到程序上部，可能导致无限循环。
JMPP 指令是仅当检测到上升沿时才执行跳转的指令。跳转后的处理同 JMP 指令一样。

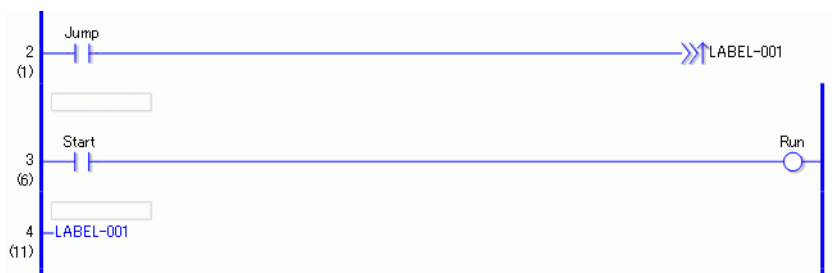
程序示例
JMP



当常开变量 Jump 置 ON 时，将执行 JMP 指令，程序将跳转第四行，其标签名为：“LABEL-001”。

跳转后，程序从第四行后继续执行。只要常开 (NO) 指令保持为 ON，就不执行第三行指令。

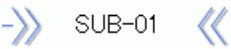
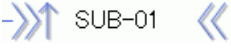
程序示例
JMPP



仅当检测到常开指令的上升沿时，才执行 JMPP 指令。然后，程序跳转至第四行，其标签名称为：“LABEL-001”。跳转后，继续执行第四行以后的程序。在后续扫描中，不执行 JMPP 指令，即便常开指令仍保持 ON 状态。在一次扫描后，才执行第三行中的程序。

31.7.2 JSR(跳转至子程序)/JSRP(上升沿跳转至子程序)

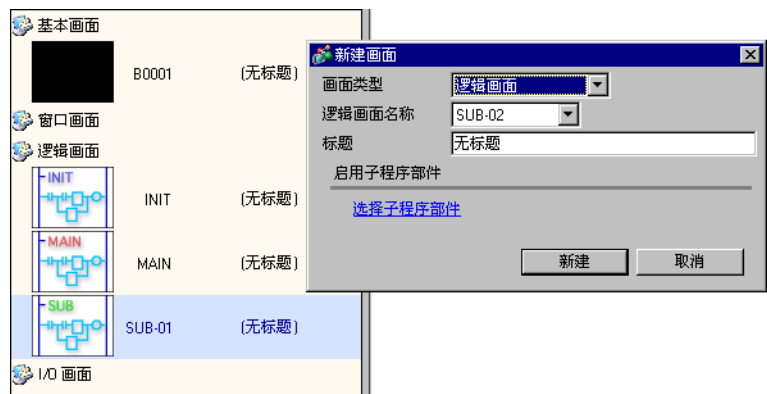
符号和功能

梯形图指令名称	梯形图符号	功能	步数
JSR (跳转至子程序)		控制	2
梯形图指令名称	梯形图符号	功能	步数
JSRP (上升沿 跳转至子程序)		控制	2

使用 JSR 指令，您最多可以指定 32 个子程序。
要定义一个到子程序的跳转，请首先创建子程序。没有子程序，您不能定义子程序跳转。您只能定义到已创建子程序的跳转。

■ 指定子程序

要创建子程序画面，在 [画面列表窗口] 中选择 [新建画面]，或在 [画面] 菜单上点击 [新建画面]。



您可以为子程序指令指定的目标是 SUB-01 到 SUB-32。
子程序名称是固定的，不能随意命名。

程序示例

JSR



当常开指令置 ON、表示发生问题时，将执行 JSR 指令。JSR 指令跳转至子程序画面“SUB-01”并执行程序。当“SUB-01”结束时，程序将返回 JSR 指令后的行并继续执行。在后续扫描中，如果常开指令仍保持为 ON，JSR 指令将执行。须将 JSR 指令放在行的末尾。

程序示例

JSRP



当检测到常开指令的上升沿时，JSRP 指令执行。JSRP 指令跳转至子程序画面“SUB-01”并执行程序。当“SUB-01”结束时，程序将返回 JSRP 指令后的行继续执行。在后续扫描中，即使常开指令仍保持为 ON，JSRP 指令也不执行。在第一次扫描后，子程序不运行，程序继续执行后面的行。在一次扫描后，不执行子程序处理，而执行下一行中的处理。须将 JSRP 指令放在最后一行。

■ 限制

(1)只能将 JSR 和 JSRP 指令放在行的右端。

(2)子程序跳转最多可达 128 次。

一个子程序跳转使用一个堆栈。逻辑程序总共可以使用 128 个堆栈。

使用堆栈的其他指令是 FOR 和 NEXT 指令。每对 FOR/NEXT 指令使用两个堆栈。

31.7.3 RET(返回)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
RET (返回)		控制	1

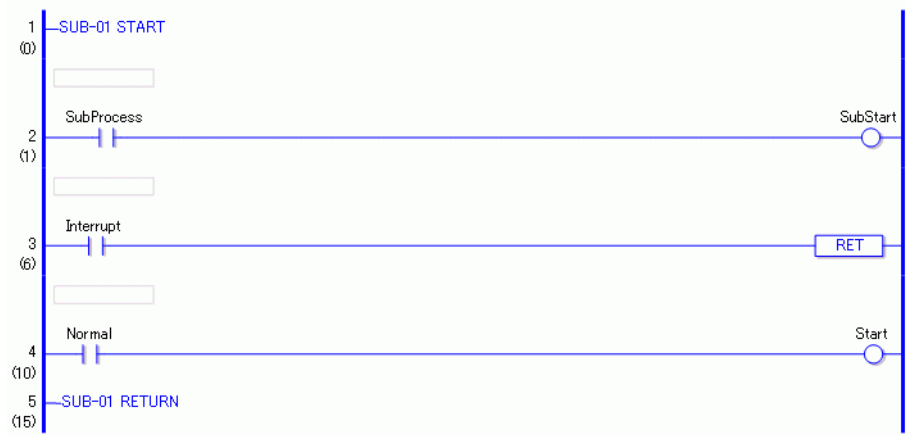
RET 指令将程序从子程序返回到原始的 JSR 指令调用处，并从后面的行开始继续执行指令。

RET 指令用于中断子程序并返回主程序。

由于程序在子程序结束后会自动返回调用处，因此有时并不需要使用 RET 指令。须将 RET 指令放在行的末尾。RET 指令只能在子程序中使用。

程序示例

RET



RET 指令只能在子程序中使用。当跳转至子程序指令在主程序中执行时，程序流就转移到子程序。子程序处理第一行和第二行中的指令。如果第三行中的常开指令变量为 ON，则执行 RET 指令，将程序流返回到主程序，而不执行第四行。

如果未执行 RET 指令，程序会执行第四行，然后在子程序结束 (END) 后将程序返回到主程序。

31.7.4 FOR NEXT(循环)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
FOR (重复)		控制	2 至 4
梯形图指令名称	梯形图符号	功能	步数
NEXT (重复)		控制	1

■ 操作数设置

下表列出了 FOR 指令中操作数 (S1) 的可配置条件。

名称	类型	条件	指令步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	2	O
内部地址	位			X
	字	指定字中的一个位。 (例如, [#INTERNAL]LS000000)	2	O
符号	位			X
	字		2	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型	未指定数组和修饰符	2	O
		指定整型变量 [常量]	3	O
		指定整型变量 [变量]	4	O
		指定整型变量 [常量 / 变量].X[常量 / 变量]		X
	浮点型			X
	实型			X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	指令步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		2	O
	Q_		2	O
	D_	未指定修饰符	2	O
		D_****.X[常量]		X
		D_****.X[地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量		0 至 2147483647	2	O

■ FOR 和 NEXT 指令说明

FOR 和 NEXT 指令重复执行 FOR 和 NEXT 之间的逻辑，重复次数在 S1 中指定。按照 S1 中指定的次数执行完 FOR 和 NEXT 指令间的处理后，NEXT 指令后的行将无条件运行。当 S1 为 0 或以下时，FOR 和 NEXT 之间的逻辑将不执行，程序将跳转至 NEXT 指令后的行。

FOR 和 NEXT 指令必须成对使用。这些指令总是运行。

程序示例

FOR 和 NEXT

其他指令不能和 FOR 和 NEXT 指令在同一行中出现。您可以使用 JMP 指令来指定执行 FOR 和 NEXT 指令的条件。如下 FOR 和 NEXT 指令的程序示例说明了如何使用条件来运行 FOR 和 NEXT 指令。



当常开指令的变量置 ON 时，FOR 和 NEXT 指令将不执行，程序将跳转至 “LABEL-001”。当该变量置 OFF 时，FOR 和 NEXT 指令执行。FOR 指令的操作数 S1 的值 (N) 表示 FOR 和 NEXT 指令之间的行将被重复的次数。当 S1 = 10 时，FOR 循环重复 10 次。在退出 FOR 循环后，会继续处理 NEXT 指令后的指令。

■ 限制

- (1)在插入 FOR 指令后，必须插入相应的 NEXT 指令。
- (2)不要在同一行中 FOR 至 NEXT 指令的前后插入指令。
(您不能在 FOR 或 NEXT 指令行上设置任何条件。)
- (3)您不能更改 FOR 和 NEXT 指令之间的执行次数。
- (4)您不能中途退出 FOR 和 NEXT 指令。
- (4)您最多可以嵌套 FOR 和 NEXT 指令 64 次 。如果超过 64 层嵌套，会出现重大错误，并将错误代码 4 写入 #L_FaultCode。
- (5)每层嵌套使用两个堆栈。逻辑程序总共可以使用 128 个堆栈。
除 FOR 和 NEXT 指令外，JSR 指令也使用堆栈。JSR 指令只使用一个堆栈。

31.7.5 INV(反转)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
INV (反转)		控制	1

■ 反转 (INV) 指令说明

当执行 INV 指令时，将进行反转处理。如果在执行 INV 指令前状态是 OFF，状态将反转为 ON。

如果在 INV 指令执行前状态为 ON，那么指令执行后状态就反转为 OFF。

程序示例



当常开指令的操作数为 ON 时， INV 指令将执行， OUT 线圈置 OFF。

31.7.6 EXIT(退出)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
EXIT (退出)		控制	1

■ EXIT 指令说明

EXIT 指令只能用在主程序中。在执行该指令后，程序跳转至 END。
执行完该指令后，将不执行 EXIT 和 END 之间的指令。该指令跳转至 END 标签，像
跳转指令一样。

程序示例



当开关置 ON 时，行末端的 EXIT 指令将运行。因此，将不执行 EXIT 和 END 之间的
指令。

31.7.7 PBC(母线控制开始) 和 PBR(母线控制复位)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
PBC (母线控制开始)		控制	3
梯形图指令名称	梯形图符号	功能	步数
PBR (母线控制复位)		控制	2

■ 操作数设置

下面描述了 PBC 指令中的操作数 (S1) 和 (D1) 的可指定内容。

名称	类型	条件	指令步数	可用: ○ 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	指定字中的一个位。 (例如, [#INTERNAL]LS000000)		X
符号	位			X
	字			X
变量格式	位	位指定 (仅 D1 操作数)	3	○
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型	未指定数组和修饰符		X
		指定整型变量 [常量]		X
		指定整型变量 [变量]		X
		指定整型变量 [常量 / 变量].X[常量 / 变量]		X
	浮点型			X
	实型			X
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	指令步数	可用：○ 不可用：X
地址格式	X_			X
	Y_	(仅操作数 D1)	3	○
	M_	(仅操作数 D1)	3	○
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.X[常量]		X
		D_****.X[地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量		0 至 7(仅操作数 S1)	3	○

■ 操作数设置

下面描述了 PBR 指令操作数 (S1) 的可指定内容。

名称	类型	条件	步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	指定字中的一个位。(例如, [PLC1]D0000.00)		X
内部地址	位			X
	字	指定字中的一个位。 (例如, [#INTERNAL]LS000000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型	未指定数组和修饰符		X
		指定整型变量 .X[常量]		X
		指定整型变量 .X[变量]		X
		指定整型变量 [常量 / 变量].X[常量 / 变量]		X
	浮点型			X
	实型			X
	定时器	仅 .Q / .TI / .R		X
	计数器	仅 .R / .UP / .QU / .QD / .Q		X
	日期			X
	时间			X
	PID	仅 .Q / .UO / .TO / .PF / .IF		X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.X[常量]		X
		D_****.X[地址]		X
	F_			X
	R_			X
	T_	仅 .Q / .TI / .R		X
	C_	仅 .R / .UP / .QU / .QD / .Q		X
	N_			X
	J_			X
	U_	仅 .Q / .UO / .TO / .PF / .IF		X
常量		0 至 7(仅操作数 S1)	2	O

■ 母线控制开始 (PBC) 和母线控制复位 (PBR) 指令说明

当执行了 PBC 指令时，将执行 PBC 和 PBR 之间的程序。

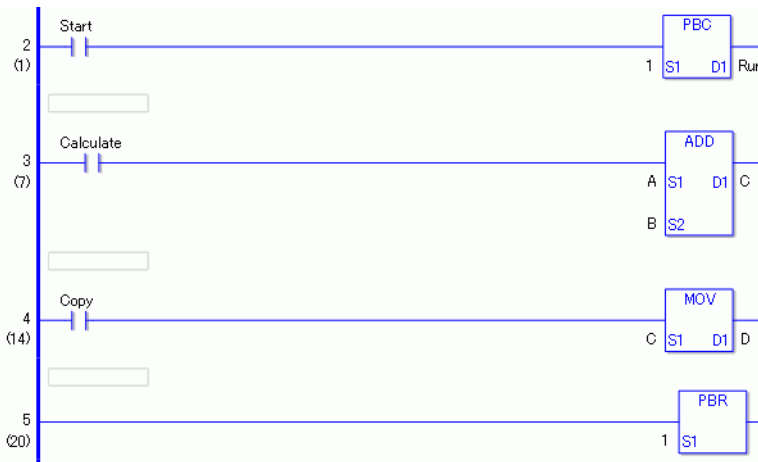
只能在主程序中使用 PBC 和 PBR 指令。不能在程序的其他部分中使用它们。

当 PBC 指令置 ON 时，D1 中的位变量将置 ON。运行于 PBC 和 PBR 指令之间的程序执行 ON 处理，直到 PBC 指令置 OFF。

每一条 PBC 指令都需要一条 PBR 指令。

PBC/PBR 指令的操作数 S1 指定嵌套层。程序将执行 PBC 和 PBR 之间指定层级的处理。

程序示例 (无嵌套)



当常开指令变量置 ON 时，PBC 指令将执行。当执行 PBC 指令时，执行 PBC 和 PBR 指令之间的处理。

(1) 当 PBC 指令为 OFF 时 (PBC 执行位为 OFF)

即使第三行中的常开指令置 ON，ADD 指令也不执行。

即使第四行中的常开指令置 ON，MOV 指令也不执行。

(2) 当 PBC 指令为 ON 时 (PBC 执行位为 ON)

当第三行中的常开指令置 ON 时执行 ADD 指令。

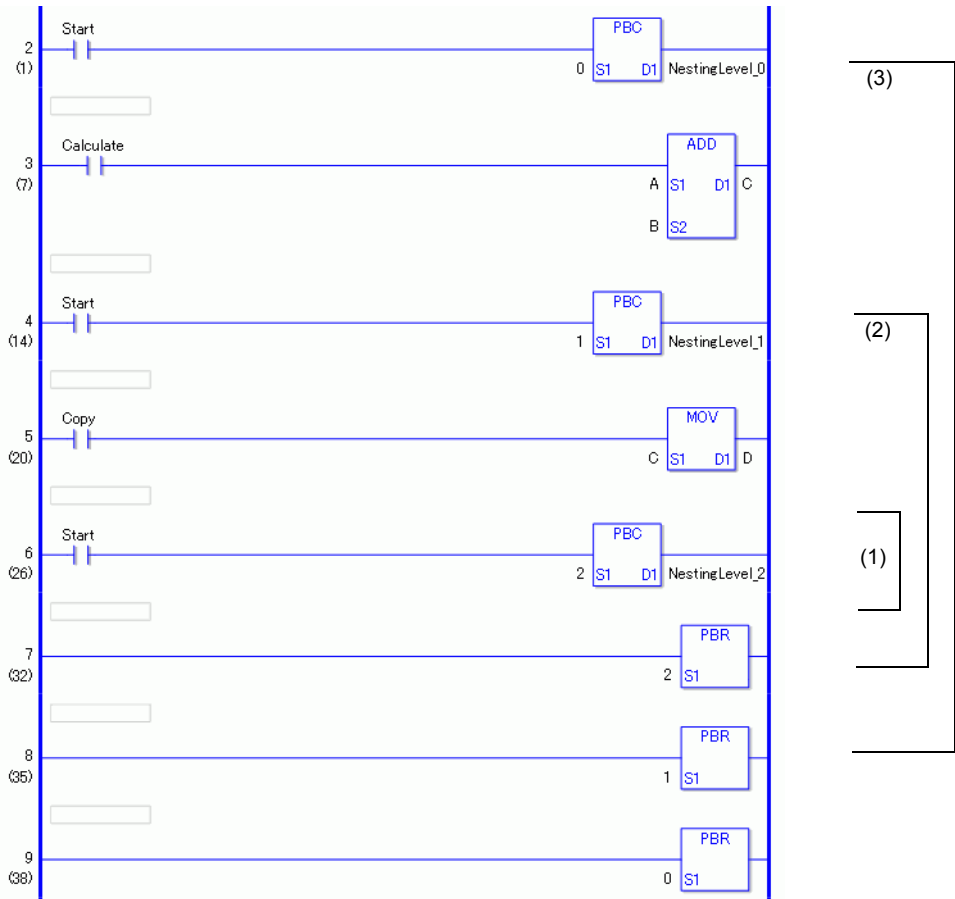
当第四行中的常开指令置 ON 时执行 MOV 指令。

■ 每条指令的状态

保持状态的元件：由累计定时器、计数器或 SET 和 RST 指令驱动的元素。

置 OFF 的元件：由定时器和 OUT 指令驱动的元素。

程序示例 (带有三层嵌套)



■ PBC 指令嵌套

PBC 指令最多可以有 8 层嵌套。

当在 PBC 指令内使用 PBC 指令时，嵌套层数 (S1) 必须增加。

(0->1->2->3->4->5->6->7)

要释放嵌套层，请使用 PBR 指令。

(7->6->5->4->3->2->1->0)

例如，如果您在未释放 PBR6 和 PBR7 的情况下释放嵌套的 PBR5，会将第五层以下的嵌套一起释放。

(1)这是嵌套层 2。在以前的程序中，状态为低。

(2)这是嵌套层 1。在以前的程序中，状态为中。

(3)这是嵌套层 0。在以前的程序中，状态为高。

31.7.8 LWA(逻辑等待)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
LWA (逻辑等待)		控制	2

■ 操作数设置

下面描述了操作数 (S1) 的可指定内容。

名称	类型	条件	步数	可用: ○ 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	指定字中的一个位。 (例如, [#INTERNAL]LS000000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型	未指定数组和修饰符		X
		指定整型变量 [常量]		X
		指定整型变量 [变量]		X
		指定整型变量 [常量 / 变量].X[常量 / 变量]		X
	浮点型			X
	实型			X
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	步数	可用：○ 不可用：×
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.X[常量]		X
		D_****.X[地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量		1 至 10	2	○

■ 逻辑等待 (LWA) 指令说明

LWA 指令将逻辑停止一段时间，该时间在 S1 中指定。如果播放视频的过程中出现闪动，请使用 LWA 指令。

您可以使用 LWA 指令来防止视频播放过程中的闪动。程序执行的能量流总能通过 LWA 指令。

(注释)

如果使用大量的 LWA 指令，可能会发生 WDT(看门狗时间) 错误。由于 WDT 错误会影响扫描时间，因此在使用 LWA 指令时必须多加注意。

使用限制

- (1) 如果使用大量的 LWA 指令，可能会出现 WDT(看门狗时间) 错误。由于 WDT 错误会影响扫描时间，因此在使用 LWA 指令时必须多加注意。
- (2) 一行中只能放一个 LWA 指令。
- (3) LWA 指令必须是行上的最后一个指令，位于紧邻右侧母线的左边。
- (4) LWA 指令只能在 MAIN 和 SUB 中使用，不能在 INIT 中使用。

程序示例

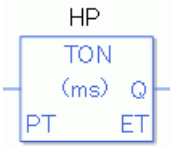
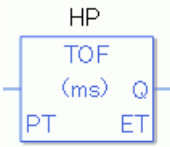


- (1) 当位变量置 ON 时，将执行 LWA 指令。
- (2) 当执行 LWA 指令时，逻辑程序停止操作数 S1 中指定的时间 (1 至 10ms)。
- (3) 在到达了指定时间后，处理将从下一行继续。

31.8 定时器指令

31.8.1 TON(ON 延时定时器) 和 TOF(OFF 延时定时器)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
TON (ON 延时定时器)		定时器	2
梯形图指令名称	梯形图符号	功能	步数
TOF (OFF 延时定时器)		定时器	2

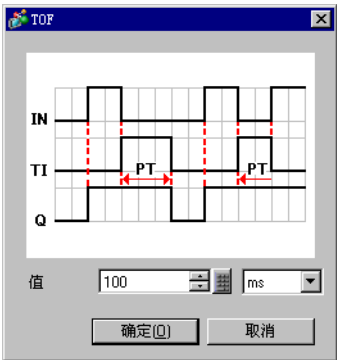
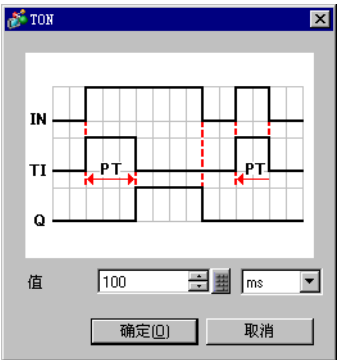
■ ON 延时定时器 (TON) 和 OFF 延时定时器 (TOF) 指令说明

TON 和 TOF 指令中使用的定时器变量是结构变量。下表列出内部结构。

定时器变量

定时器变量	变量设置	描述
变量名 .TI	位变量	当定时器开始工作时置 ON。
变量名 .Q	位变量	当定时器完成工作时置 ON。
变量名 .PT	整型变量	定时器上的设定值
变量名 .ET	整型变量	定时器上的当前值

双击定时器指令，显示如下对话框。在该对话框中输入预设时间。
输入设置值和单位。



要进行基于时间的设置，请双击定时器指令显示设置对话框。

时基	描述	PT 值 /ET 值
ms	以 1 毫秒为单位指定时间。	以毫秒为单位指定和显示 PT 值。 以毫秒为单位显示 ET 值。 设置范围 = 0 至 2147483647 x 1ms
10ms	以 10 毫秒为单位指定时间。	以 10 毫秒为单位设置和显示 PT 值。 以 10 毫秒为单位显示 ET 值。 设置范围 = 0 至 214748364 x 10ms
0.1s	以 0.1 秒为单位指定时间。	以 0.1 秒为单位指定和显示 PT 值。 以 0.1 秒为单位显示 ET 值。 设置范围 = 0 至 21474836 x 100ms
s	以 1 秒为单位指定时间。	以 1 秒为单位指定和显示 PT 值。 以 1 秒为单位显示 ET 值。 设置范围 = 0 至 2147483 x 1s

程序示例

TON



(1)当常开指令的变量置 ON 时，因为触发了 TON 指令，经过时间 .ET 将按指定的时间单位增加。

- 定时器测量位 .TI 置 ON。
- 定时器输出位 .Q 置 OFF。

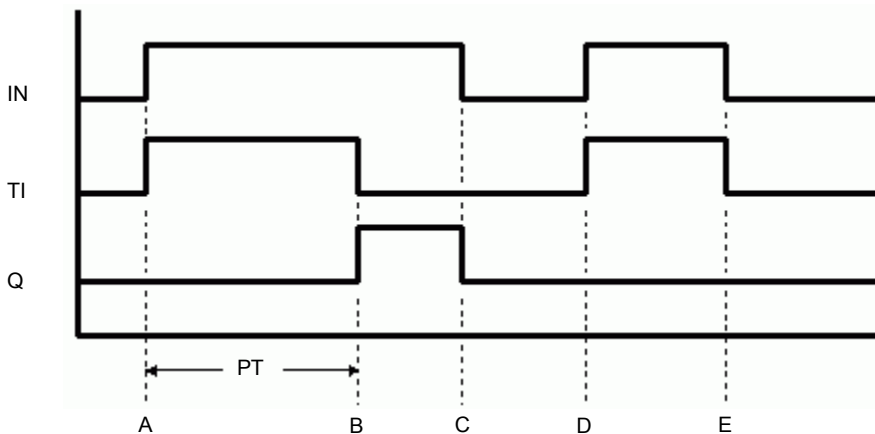
(2)当经过时间 .ET 增加到等于预设时间 .PT 时，经过时间 .ET 保持当前值。

- 定时器测量位 .TI 置 OFF。
- 定时器输出位 .Q 置 ON 并接通回路。

(3)当开始测量为 OFF(关闭) 时，经过时间 .ET 将复位为 0。

- 定时器测量位 .TI 置 OFF。
- 定时器输出位 .Q 置 OFF。

■ TON 指令操作的时序图



- A 点 定时器置 ON，定时器测量位 .TI 置 ON。定时器测量开始，经过时间 .ET 增加。定时器输出位 .Q 保持 OFF。
- B 点 当经过时间 .ET 等于预设时间 .PT 时，定时器输出位 .Q 置 ON。经过时间值。ET 仍和预设时间 .PT 相同。定时器测量位 .TI 置 OFF。
- C 点 定时器置 OFF，定时器输出位 .Q 置 OFF。经过时间 .ET 复位为 0。
- D 点 定时器置 ON，定时器测量位 .TI 置 ON。定时器测量开始，经过时间 .ET 增加。
- E 点 定时器在经过时间 .ET 达到预设时间 .PT 前置 OFF。定时器输出位 .Q 保持 OFF，经过时间 .ET 复位为 0。

程序示例

TOF



(1)常开指令变量置 ON 时，因为触发了 TON 指令，经过时间 .ET 将复位为 0。

- 定时器测量位 .TI 置 OFF。
- 定时器输出位 .Q 置 ON 并接通回路。

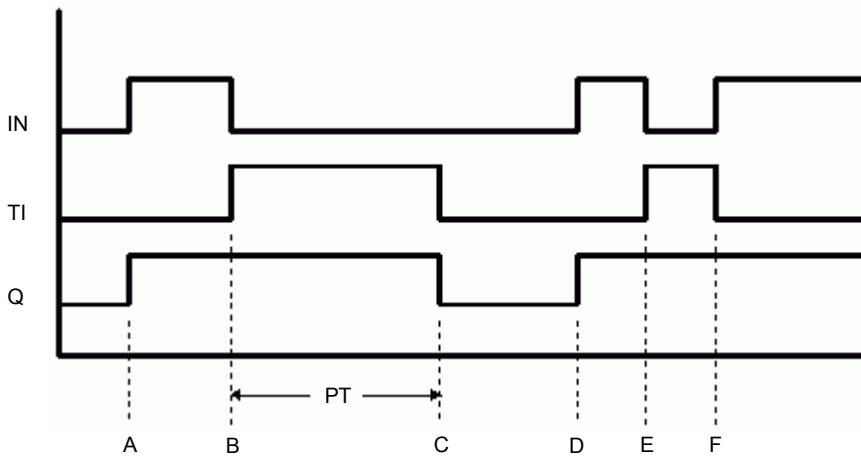
(2)当触发了 TOF 指令，测量开始位置 OFF 时，经过时间 .ET 将按指定时间单位增加。

- 定时器测量位 .TI 置 ON。
- 定时器输出位 .Q 保持为 ON。

(3)当经过时间 .ET 增加到等于预设时间 .PT 时，经过时间 .ET 保持当前值。

- 定时器测量位 .TI 置 OFF。

■ TOF 指令操作的时序图



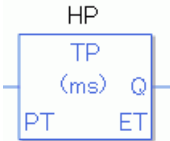
- A 点 定时器置 ON。定时器测量位 .TI 保持 OFF。定时器输出位 .Q 置 ON。经过时间 .ET 复位为 0。
- B 点 定时器置 OFF。定时器开始测量 (.TI 置 ON)。定时器输出位保持为 ON。
- C 点 经过时间 .ET 等于预设时间 .PT，定时器输出位 .Q 置 OFF。定时器停止测量 (.TI 置 OFF)。经过时间 .ET 保持等于设置时间 (ET = PT)。
- D 点 定时器置 ON。定时器测量位 .TI 保持 OFF。定时器输出位 .Q 保持为 ON。经过时间 .ET 复位为 0。
- E 点 定时器置 OFF。定时器开始测量 (.TI 置 ON)。定时器输出位 .Q 保持为 ON。
- F 点 定时器在经过时间 .ET 达到预设时间 .PT 之前置 ON，定时器停止测量。(.TI 置 OFF)。定时器输出位 .Q 保持为 ON，经过时间 .ET 复位为 0。

■ 确认执行结果

- (1)如果您输入了一个超出定义范围的值，会发生错误，并将错误代码 (6706) 写入 #L_CalcErrCode。在排除故障时，请务必检查 #L_CalcErrCode 中的错误代码。如果输入了任何超出设定范围的值，将不执行指令。

31.8.2 TP(脉冲定时器)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
TP (上升沿定时器)		定时器	2

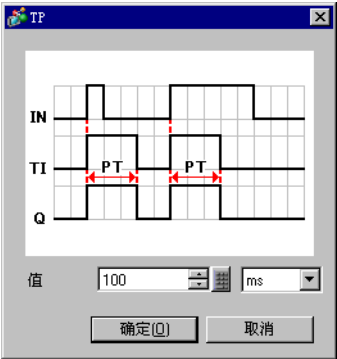
■ 脉冲定时器 (TP) 指令说明

TP 指令中使用的定时器变量是结构变量。下表列出内部结构。

定时器变量

定时器变量	变量设置	描述
变量名 .TI	位变量	当定时器开始工作时置 ON。
变量名 .Q	位变量	当定时器完成工作时置 ON。
变量名 .PT	整型变量	定时器上的设定值
变量名 .ET	整型变量	定时器上的当前值

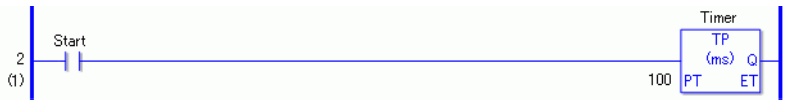
双击定时器指令，显示如下对话框。在该对话框中输入预设时间。
输入设置值和单位。



要进行基于时间的设置，请双击定时器指令显示设置对话框。

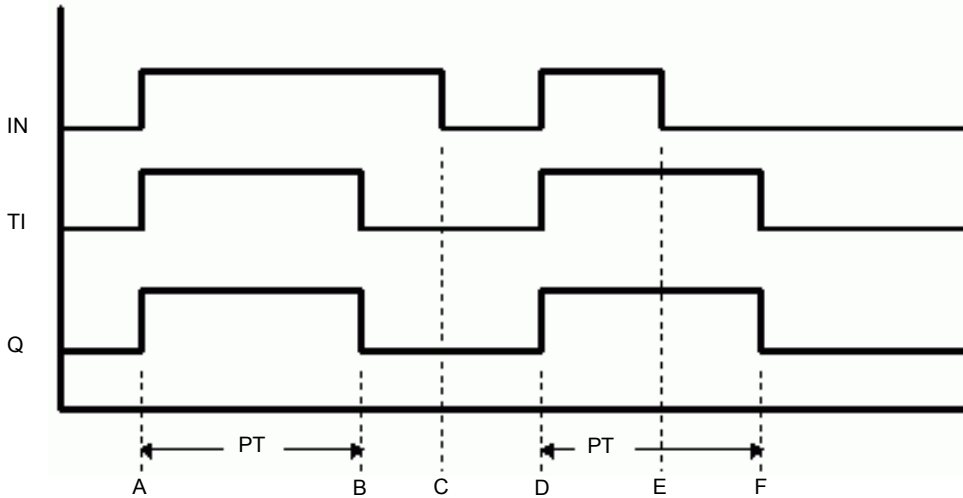
时基	描述	PT 值 /ET 值
ms	以 1 毫秒为单位指定时间。	以毫秒为单位指定和显示 PT 值。 以毫秒为单位显示 ET 值。 0 至 2147483647 x 1 毫秒
10 ms	以 10 毫秒为单位指定时间。	以 10 毫秒为单位设置和显示 PT 值。 以 10 毫秒为单位显示 ET 值。 设置范围 = 0 至 214748364 x 10 毫秒
0.1s	以 0.1 秒为单位指定时间。	以 0.1 秒为单位指定和显示 PT 值。 以 0.1 秒为单位显示 ET 值。 设置范围 = 0 至 21474836 x 100 毫秒
s	以 1 秒为单位指定时间。	以 1 秒为单位指定和显示 PT 值。 以 1 秒为单位显示 ET 值。 设置范围 = 0 至 2147483 x 1s

程序示例



- (1)当常开指令置 ON 时，将触发 TP 指令。由于 TP 指令检测到上升沿，当触发该指令时，它会启动定时器，无论定时器处于何种状态。
经过时间 .ET 按指定的时间单位增加。
- 定时器测量位 .TI 置 ON。
 - 定时器输出位 .Q 置 ON 并接通回路。
- (2)当经过时间 .ET 达到预设时间时，TP 指令置 OFF。
在达到预设时间后定时器输出位 .Q 置 OFF，无论 TP 指令左边是否有电源流过。
- 当 PT =<ET 时，它会立刻复位为 0。
 - 当经过时间 .ET 等于预设时间 .PT 时，定时器位 .TI 置 OFF。
 - 当 TP 指令为 OFF 时，定时器输出位 .Q 也为 OFF。
- (3)当常开指令变量置 OFF 时，如果经过时间 .ET 已经达到预设时间 .PT，那么经过时间 .ET 就复位为 0。
- 定时器输出位 .Q 置 OFF。
 - 否则，定时器会继续测量，定时器输出位 .Q 保持为 ON。

■ TP 指令的时序图



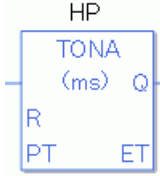
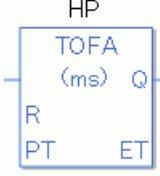
- A 点 定时器置 ON。定时器开始测量 (.TI 置 ON)。定时器输出位 .Q 置 ON。
- B 点 经过时间 .ET 等于预设时间 .PT，定时器输出位 .Q 置 OFF。定时器停止测量 (.TI 置 OFF)。经过时间 .ET 保持等于预设时间 (ET = PT)。
- C 点 定时器置 OFF。经过时间 .ET 复位为 0。
- D 点 定时器置 ON。定时器开始测量 (.TI 置 ON)。定时器输出位 .Q 置 ON。
- E 点 定时器置 OFF。定时器继续测量 (.TI 保持为 ON)。定时器输出位 .Q 保持为 ON。
- F 点 经过时间 .ET 等于预设时间 .PT，定时器输出位 .Q 置 OFF。定时器停止测量 (.TI 置 OFF)。由于定时器输入位 IN 为 OFF，经过时间 .ET 复位为 0。

■ 确认执行结果

- (1)如果您输入了一个超出定义范围的值，会发生错误，并将错误代码 (6706) 写入 #L_CalcErrCode。在排除故障时，请务必检查 #L_CalcErrCode 中的错误代码。如果输入了任何超出设定范围的值，将不执行指令。

31.8.3 TONA(累计 ON 延时定时器) 和 TOFA(累计 OFF 延时定时器)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
TONA (累计 ON 延时定时器)		定时器	2
梯形图指令名称	梯形图符号	功能	步数
TOFA (累计 OFF 延时定时器)		定时器	2

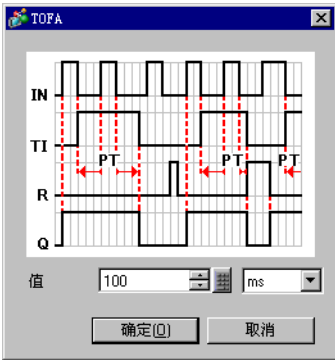
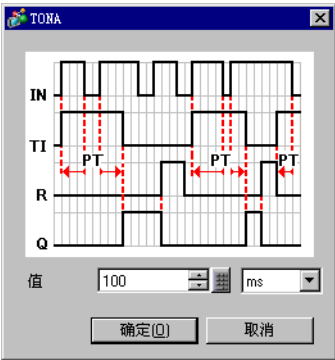
■ 累计 ON 延时定时器 (TONA) 和累计 OFF 延时定时器 (TOFA) 指令说明

TONA 和 TOFA 指令中使用的定时器变量是结构变量。下表列出内部结构。

定时器变量

定时器变量	变量设置	描述
变量名 .TI	位变量	当定时器开始工作时置 ON。
变量名 .Q	位变量	当定时器完成工作时置 ON。
变量名 .R	位变量	复位当前定时器 .0Clear(0)。
变量名 .PT	整型变量	定时器上的设定值
变量名 .ET	整型变量	定时器上的当前值

双击定时器指令，显示如下对话框。在该对话框中输入预设时间。
输入设置值和单位。

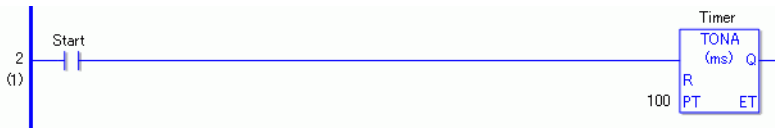


要进行基于时间的设置，请双击定时器指令显示设置对话框。

时基	描述	PT 值 /ET 值
ms	以 1 毫秒为单位指定时间。	以毫秒为单位指定和显示 PT 值。 以毫秒为单位显示 ET 值。 0 至 2147483647 x 1ms
10 ms	以 10 毫秒为单位指定时间。	以 10 毫秒为单位设置和显示 PT 值。 以 10 毫秒为单位显示 ET 值。 设置范围 = 0 至 214748364 x 10ms
0.1s	以 0.1 秒为单位指定时间。	以 0.1 秒为单位指定和显示 PT 值。 以 0.1 秒为单位显示 ET 值。 设置范围 = 0 至 21474836 x 100ms
s	以 1 秒为单位指定时间。	以 1 秒为单位指定和显示 PT 值。 以 1 秒为单位显示 ET 值。 设置范围 = 0 至 2147483 x 1s

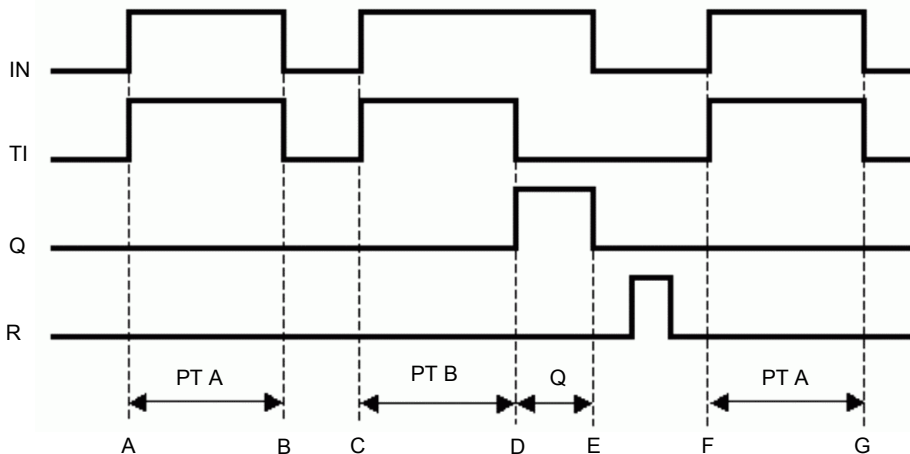
程序示例

TONA



- (1)当常开指令变量置 ON 时，因为触发了 TONA 指令，经过时间 .ET 将按指定的时间单位增加。
 - 定时器测量位 .TI 置 ON。
 - 定时器输出位 .Q 置 OFF。
- (2)当经过时间 .ET 增加到等于预设时间 .PT 时，经过时间 .ET 保持当前值。
 - 定时器测量位 .TI 置 OFF。
 - 定时器输出位 .Q 置 ON 并接通回路。
- (3)当 TONA 指令置 OFF 时，经过时间 .ET 保持当前值。
 - 定时器测量位 .TI 置 OFF。
 - 定时器输出位 .Q 置 OFF。
- (4)TONA 指令的作用就像是一个累加器，不断增加其值。将 R 线圈置 ON，复位当前值。

■ TONA 指令的时序图



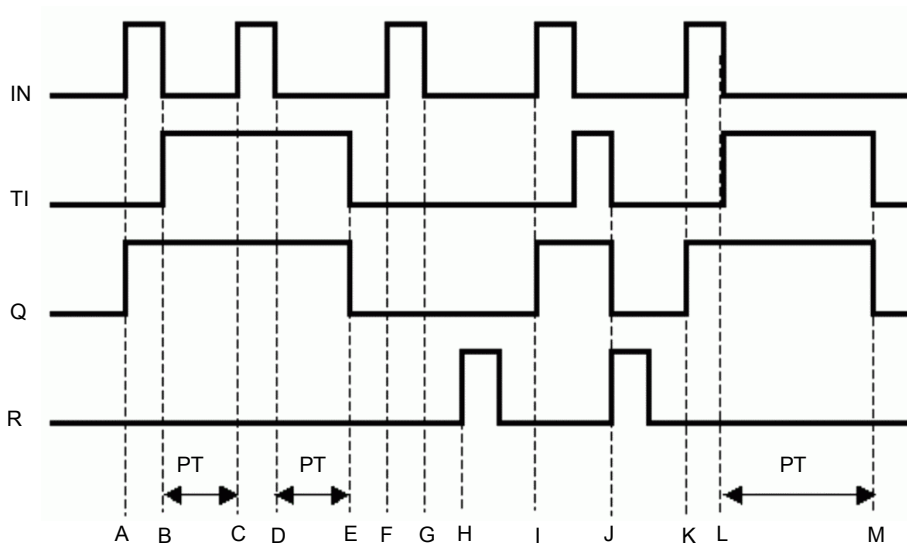
- A 和 F 点 定时器输入位 IN 置 ON，定时器测量位 TI 置 ON。定时器启动，经过时间 .ET 增加。定时器输出位 Q 保持 OFF。
- B 和 G 点 定时器输入位 IN 置 OFF，如果经过时间 ET 小于预设时间 PT，定时器输出位 Q 保持 OFF。经过时间 ET 处于保持状态。
- C 点 定时器输入位 IN 置 ON，定时器测量位 TI 置 ON。定时器再次开始测量，将经过时间 ET 累加到保持值上。定时器输出位 Q 保持 OFF。
- D 点 当经过时间 ET 达到预设时间 PT 时，定时器测量位 TI 置 OFF。
- E 点 定时器输出位 Q 置 ON。
- E 点 定时器输入位 IN 置 OFF，定时器输出位 Q 置 OFF。使用复位位 (R) 将经过时间复位为 0。

■ TOFA 指令的操作示例



- (1)当定时器置 OFF 时，由于触发了 TOFA 指令，经过时间 .ET 按指定的时间单位增加。
 - 定时器测量位 .TI 置 ON。
 - 定时器输出位 .Q 置 OFF。
- (2)当经过时间 .ET 增加到等于预设时间 .PT 时，经过时间 .ET 保持当前值。
 - 定时器测量位 .TI 置 OFF。
 - 定时器输出位 .Q 置 ON 并接通回路。
- (3)当 TONA 指令置 OFF 时，经过时间 .ET 保持当前值。
 - 定时器测量位 .TI 置 OFF。
 - 定时器输出位 .Q 置 OFF。

■ TOFA 指令的时序图



- A 点 当 IN(输入) 置 ON 时，Q(输出) 置 ON。
- B 点 当 IN(输入) 置 OFF 时，TI(定时器测量) 置 ON。当 TI 置 ON 时，定时器测量开始。
- C 点 当 IN(输入) 置 ON 时，定时器测量暂停。
- D 点 当 IN(输入) 置 OFF 时，暂停的定时器测量继续开始。
- E 点 当预设时间 (PT) 值增加到 PT 等于 ET 的点时，TI(定时器测量) 和 Q(输出) 置 OFF。
- F、G 点 无论 IN(输入) 置 ON 还是置 OFF，Q(输出) 和 TI(定时器) 都不会置 ON。
- H 点 将 R 置 ON 会复位定时器。当检测到上升沿时定时器复位。
- I 点 当 IN(输入) 置 ON 时，Q(输出) 置 ON。

- | | |
|-----|--|
| J 点 | 当 R(复位) 置 ON 时, Q(输出) 和 TI(定时器) 复位。定时器当前值 ET 也被复位并清 0。 |
| K 点 | 当 IN(输入) 置 ON 时, Q(输出) 置 ON。 |
| L 点 | 当 IN(输入) 置 OFF 时, TI(定时器测量) 置 ON。当 TI 置 ON 时, 定时器测量开始。 |
| M 点 | 当定时器设置 (PT) 值增加到 PT 等于 ET 的点时, TI(定时器测量) 和 Q (输出) 置 OFF。 |

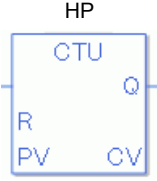
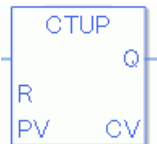
■ 确认执行结果

- (1)如果您输入了一个超出定义范围的值, 会发生错误, 并将错误代码 (6706) 写入 #L_CalcErrCode。有关错误检查的详细信息, 请参阅 #L_CalcErrCode。如果输入了任何超出设定范围的值, 将不执行指令。

31.9 计数器指令

31.9.1 CTU 和 CTUP(加计数器)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
CTU (加计数器 - 电平)		计数器	2
梯形图指令名称	梯形图符号	功能	步数
CTUP (加计数器 - 上升沿)		计数器	2

■ CTU 和 CTUP 指令说明

CTU 和 CTUP 指令中使用的计数器变量是结构变量。下表列出内部结构。

计数器变量

计数器变量	变量设置	描述
变量名 .R	位变量	复位当前值。清除 (0)。
变量名 .Q	位变量	在当前值达到预设值时置 ON。
变量名 .UP	位变量	当变量为 ON 时进行加计数。
变量名 .QU	位变量	对于加 / 减计数器，在当前值达到预设值时置 ON。
变量名 .QD	位变量	对于加 / 减计数器，在当前值达到 0 或更小时置 ON。
变量名 .PV	整型变量	预设值
变量名 .CV	整型变量	当前值

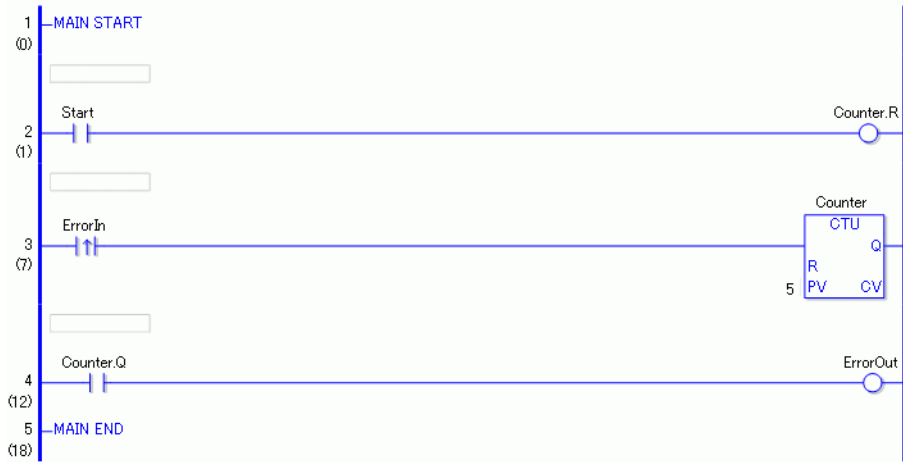
当执行 CTU 和 CTUP 指令时，如果计数器复位位变量 .R 为 OFF，无论预设值 .PV 是多少，当前值 .CV 都将加 1。当当前值 .CV 等于预设值 .PV 时，计数器输出位变量 .Q 置 ON。当计数器重置位变量 .R 置 ON 时，当前值 .CV 被重置为 0。计数器输出位变量 .Q 也置 OFF。

程序示例

CTU

在下面的例子中，如果 1 分钟内记下了 5 个操作错误，就会显示一个错误。
在程序示例中，不显示定时器指令。只显示用于定时器启动的一分钟定时器启动触发器。

要计数操作错误，请创建一个独立的错误输入触发器。



(1) 当一分钟定时器的常开指令置 ON 时，分配给计数器 .R(复位) 的 OUT 指令置 ON。

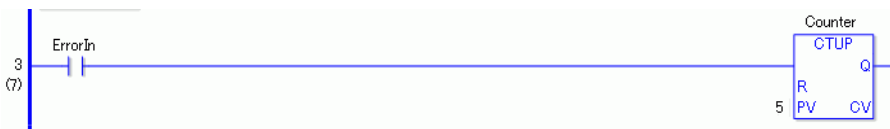
当操作错误计数器 .R(复位) 置 ON 时，CTU 指令的操作错误计数器 .CV 清 0。

(2) 当第三行中的上升沿常开指令置 ON 时，操作错误计数器 .CV 的值 (当前值) 加 1。

(3) 当操作错误计数器 .CV 的值 (当前值) 等于 .PV 值 (预设值) 时，CTU 指令的操作错误代码 .Q 置 ON，第四行中的 OUT 指令输出错误检测消息。

程序示例

CTUP



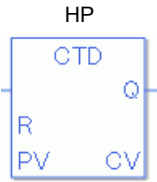
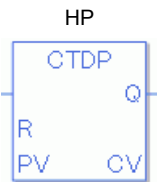
CTU 和 CTUP 指令之间的不同在于 .CV 值是作为电平计数器增加还是作为上升沿计数器增加。

程序创建中的不同在于位于第三行的用于检测操作错误的上升沿指令是常开指令。

除了输入的决定方式外，操作上没有什么不同。

31.9.2 CTD 和 CTD(减计数器)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
CTD (减计数器 - 电平)		计数器	2
梯形图指令名称	梯形图符号	功能	步数
CTDP (减计数器 - 上升沿)		计数器	2

■ CTD 和 CTDP 指令说明

CTD 和 CTDP 指令中使用的计数器变量是结构变量。下表列出内部结构。

计数器变量

计数器变量	变量设置	描述
变量名 .R	位变量	复位当前值。清除 (0)。
变量名 .Q	位变量	在当前值达到预设值时置 ON。
变量名 .UP	位变量	当变量为 ON 时进行加计数。
变量名 .QU	位变量	对于加 / 减计数器，在当前值达到预设值时置 ON。
变量名 .QD	位变量	对于加 / 减计数器，在当前值达到 0 或更小时置 ON。
变量名 .PV	整型变量	预设值
变量名 .CV	整型变量	当前值

当 CDT 和 CDTP 指令为 ON 时，如果计数器复位位变量 .R 为 OFF，当前值 .CV 将加 1。

当当前值 .CV 小于 0 时，计数器输出位 .Q 置 ON。当计数器复位位 .R 置 ON 时，预设值 .PV 被复制到当前值变量 .CV。而且，计数器输出变量 .Q 置 OFF。

注 释

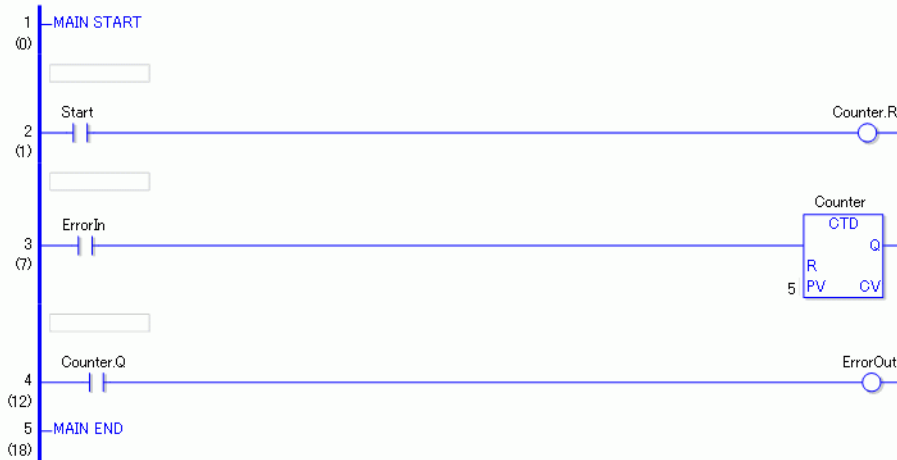
- 当减计数器的预设置设置为 0 或更小时，输出继续保持为 ON。务必将预设值设置为大于等于 1。

程序示例

CDT

在下面的例子中，如果 1 分钟内记下了 5 个操作错误，就会显示一个错误。
在程序示例中，不显示定时器指令。只显示用于定时器启动的一分钟定时器启动触发器。

要计数操作错误，请创建一个独立的错误输入触发器。



(1) 当一分钟定时器的常开指令置 ON 时，分配给计数器 .R(复位) 的 OUT 指令置 ON。

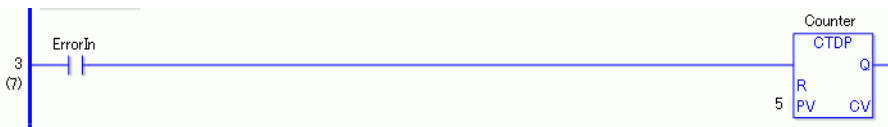
当操作错误计数器 .R(复位) 置 ON 时， CDT 指令的预设值 .PV 被复制到当前值 .CV。在程序示例中， 5 被复制到当前值 .CV。

(2) 当上升沿常开指令置 ON 时，操作错误计数器 .CV 的值 (当前值) 减 1。

(3) 当操作错误计数器 .CV 值 (当前值) 为 0 或更小时， CDT 指令的操作错误计数器 .Q 置 ON，第四行中的 OUT 指令输出错误检测消息。

程序示例

CTDP



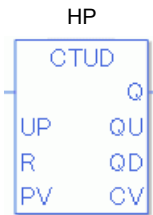
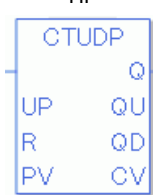
CTD 和 CTDP 指令之间的不同在于一个是在检测到电平变化时倒数，另外一个是在检测到上升沿时倒数。

程序创建中的不同在于位于第三行的用于检测操作错误的上升沿指令是常开指令。

除了输入的决定方式外，操作上没有什么不同。

31.9.3 CTUD 和 CTUDP(加 / 减计数器)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
CTUD (加 / 减计数器 - 电平)		计数器	2
梯形图指令名称	梯形图符号	功能	步数
CTUDP (加 / 减计数器 - 上升沿)		计数器	2

■ CTUD 和 CTUDP 指令说明

CTUD 和 CTUDP 指令中使用的计数器变量是结构变量。下表列出内部结构。

计数器变量

计数器变量	变量设置	描述
变量名 .R	位变量	复位当前值。清除 (0)。
变量名 .Q	位变量	在当前值达到预设值时置 ON。
变量名 .UP	位变量	当变量为 ON 时进行加计数。
变量名 .QU	位变量	对于加 / 减计数器，在当前值达到预设值时置 ON。
变量名 .QD	位变量	对于加 / 减计数器，在当前值达到 0 或更小时置 ON。
变量名 .PV	整型变量	预设值
变量名 .CV	整型变量	当前值

当 CTUD 和 CTUDP 指令的 .UP 位为 ON 时，它们的操作同 CTU 指令相同。当 .UP 位为 OFF 时，CTUD 和 CTUDP 指令的操作同 CTD 指令相同。

当 .UP 为 ON(加计数) 且 .CV(当前值) 大于 .PV(预设值) 时，.Q 在当前值达到预设值时置 ON 且 .QU 置 ON。

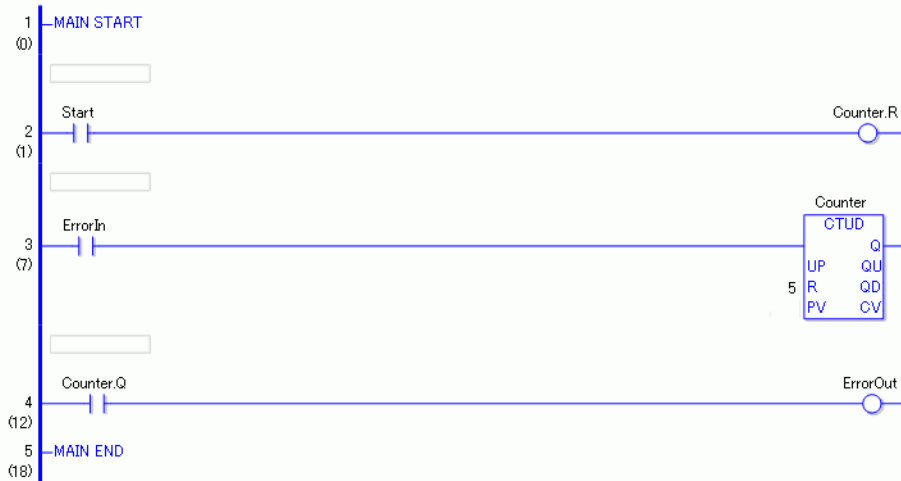
当 .UP 为 OFF(减计数) 且 .CV(当前值) 为 0 或更小时，.Q 在当前值达到预设值时置 ON 且 .QD 置 ON。

程序示例

CTUD

在下面的例子中，如果 1 分钟内记下了 5 个操作错误，就会显示一个错误。
在程序示例中，不显示定时器指令。只显示用于定时器启动的一分钟定时器启动触发器。

要计数操作错误，请创建一个独立的错误输入触发器。



- (1) 当一分钟定时器的常开指令置 ON 时，分配给计数器 .R(复位) 的 OUT 指令置 ON。

当操作错误计数器 .R(复位) 置 ON 时，如果 .UP 为 ON，将执行 CTUD 指令，并将 .CV(当前值) 清 0。如果 .UP 为 OFF，将执行 CTUD 指令，并将 .PV(预设值) 复制到 .CV(当前值)。

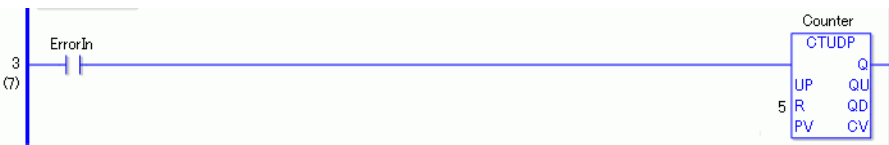
- (2) 当第三行中的上升沿常开指令置 ON 时，如果 .UP 为 ON，.CV 值加 1。如果 .UP 为 OFF，.CV 值 (当前值) 减 1。

- (3) 当 .UP 为 ON 且 .PV 值 (预设值) 和 .CV 值相等时，.Q 和 .QU 置 ON。当 .UP 为 OFF 且 .CV 值 (当前值) 小于 0 时，.Q 和 .QD 置 ON。

CTUD 指令 (当当前值达到预设值时置 ON) 的操作错误计数器 .Q 置 ON，OUT 指令输出错误检测消息。

程序示例

CTUDP



CTUD 和 CTUDP 指令之间的不同在于 .CV 值是作为电平计数器增加还是作为上升沿计数器增加。程序创建中的不同在于位于第三行的用于检测操作错误的上升沿指令是常开指令。除了输入的决定方式外，操作上没有什么不同。

31.10 读 / 写指令

31.10.1 JRD 和 JRDP(时间读取)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
JRD (时间读取 - 电平)		读取	2
梯形图指令名称	梯形图符号	功能	步数
JRDP (时间读取 - 上升沿)		读取	2

■ JR 和 JRDP 指令说明

JRD 和 JRDP 指令中使用的时间变量是结构变量。下表列出内部结构。

时间变量

时间变量	变量设置	描述
变量名 .HR	整型变量	以 BCD 形式输入小时。
变量名 .MIN	整型变量	以 BCD 形式输入分钟。
变量名 .SEC	整型变量	以 BCD 形式输入秒。

当 JRD 和 JRDP 指令通电后，当前时间将保存在 D1 的变量中。通过指定结构元素，可从时间变量中提取小时、分钟和秒。当时间变量 D1 中保存时间 12:10:45 时，.HR 时间是 12，.MIN 时间是 10，.SEC 时间是 45。

■ 确认执行结果

- (1)如果您输入了一个超出定义范围的值，会发生错误，并将错误代码 (6706) 写入 #L_CalcErrCode。在排除故障时，请务必检查 #L_CalcErrCode 中的错误代码。
- (2)当 D1 的值为 00:00:00 时，#L_CalcZero 置 ON。

程序示例

JRD

在时间变量中保存当前值。



- (1)当上升沿指令置 ON 时，将执行 JRD 指令。当执行 JRD 指令时，当前时间保存在 D1 中。

程序示例

JRDP



- (1)当常开指令置 ON 时，将执行 JRDP 指令。当执行 JRDP 指令时，当前时间保存在 D1 中。

31.10.2 JSET 和 JSETP(时间设置)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
JSET (时间设置 - 电平)		设置	6
梯形图指令名称	梯形图符号	功能	步数
JSETP (时间设置 - 上升沿)		设置	6

■ JSET 和 JSETP 指令说明

JSET 和 JSETP 指令中使用的时间变量是结构变量。下表列出内部结构。

时间变量

时间变量	变量设置	描述
变量名 .HR	整型变量	以 BCD 形式输入小时。
变量名 .MIN	整型变量	以 BCD 形式输入分钟。
变量名 .SEC	整型变量	以 BCD 形式输入秒。

当 JSET 和 JSETP 指令接通后，指定时间将保存在时间变量中。要设置时间，请使用 JSET 和 JSETP 指令。通过指定结构元素，可从 D1 的时间变量中提取小时、分钟和秒。

当 D1 中保存的当前时间是 12:10:45 时，保存在 .HR、.MIN 和 .SEC 中的时间分别是 12、10 和 45。

■ 时间设置对话框

双击 JSET 和 JSETP 指令，显示设置时间的对话框。



在上面的对话框中，在小时、分钟和秒钟中指定想要的时间。

设置范围

小时	0 - 23
分钟	0 - 59
秒钟	0 - 59

■ 确认执行结果

- (1)如果您输入了一个超出定义范围的值，会发生错误，并将错误代码 (6706) 写入 #L_CalcErrCode。在排除故障时，请务必检查 #L_CalcErrCode 中的错误代码。
- (2)当 D1 的值为 00:00:00 时，#L_CalcZero 置 ON。

程序示例

JSET

在时间变量中保存定义时间。



- (1)当上升沿指令置 ON 时，将执行 JSET 指令。当执行 JSET 指令时，定义的时间 17:30:00 保存在 D1 的时间变量中。

程序示例

JSETP



- (1)当常开指令置 ON 时，将执行 JSETP 指令。当执行 JSETP 指令时，定义的时间 17:30:00 保存在 D1 的时间变量中。

31.10.3 NRD 和 NRDP(日期读取)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
NRD (日期读取 - 电平)		读取	2
梯形图指令名称	梯形图符号	功能	步数
NRDP (日期读取 - 上升沿)		读取	2

■ NRD 和 NRDP 指令说明

NRD 和 NRDP 指令中使用的日期变量是结构变量。下表列出内部结构。

日期变量

日期变量	变量设置	描述
变量名 .YR	整型变量	以 BCD 形式输入年份。
变量名 .MO	整型变量	以 BCD 形式输入月份。
变量名 .DAY	整型变量	以 BCD 形式输入日期。

当 NRD 和 NRDP 指令接通后，当前时间就保存在 D1 中。通过指定特定日期元素，可提取日期变量的年 / 月 / 日。当 D1 中保存的当前日期是 2005/10/20 时，.YR、.MO 和 .DAY 中分别保存 05、10 和 20。

■ 确认执行结果

(1)如果输入了任何超出设定范围的数值，就会发生错误，并将“6706”错误代码设置为 #L_CalcErrCode。有关错误检查的详细信息，请参阅 #L_CalcErrCode。

程序示例

NRD

在日期变量中保存当前日期。



(1)当上升沿指令置 ON 时，将执行 NRD 指令。当执行 NRD 指令时，当前日期就保存在 D1 的日期变量中。

程序示例

NRDP



(1)当常开指令置 ON 时，将执行 NRDP 指令。当执行 NRDP 指令时，当前日期就保存在 D1 的日期变量中。

31.10.4 NSET 和 NSETP(日期设置)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
NSET (日期设置 - 电平)		设置	5
梯形图指令名称	梯形图符号	功能	步数
NSETP (日期设置 - 上升沿)		设置	5

■ NSET 和 NSETP 指令说明

NSET 和 NSETP 指令中使用的日期变量是结构变量。下表列出内部结构。

日期变量

日期变量	变量设置	描述
变量名 .YR	整型变量	以 BCD 形式输入年份。
变量名 .MO	整型变量	以 BCD 形式输入月份。
变量名 .DAY	整型变量	以 BCD 形式输入日期。

当 NSET 和 NSETP 指令接通后，将在日期变量中保存指定日期。要设置日期，请使用 NSET 和 NSETP 指令。通过指定结构元素，可从 D1 日期变量中提取年、月和日。当 D1 中保存了 JSET 指令中指定的日期 2005/10/20 时，.YR、.MO 和 .DAY 中分别保存 05、10 和 20。

■ 日期设置对话框

双击 NSET 和 NSETP 指令，显示设置日期对话框。



在上面的对话框中，在年、月和日中输入想要的日期。

设置范围

年 0 - 99

月 1 - 12

日 1 - 31(范围取决于月份。可以指定闰年。例如，2008 年 2 月有 29 天。)

■ 确认执行结果

- (1) 如果输入了任何超出设定范围的数值，就会发生错误，并将“6706”错误代码设置为 #L_CalcErrCode。有关错误检查的详细信息，请参阅 #L_CalcErrCode。

程序示例

NSET

在日期变量中保存设定日期。



- (1) 当上升沿指令置 ON 时，将执行 NSET 指令。当执行 NSET 指令时，将在 D1 的日期变量中保存对话框中指定的日期：10(月)20(日)，2005。

程序示例

NSETP

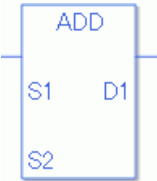


- (1) 当常开指令置 ON 时，将执行 NSETP 指令。当执行 NSETP 指令时，将在 D1 的日期变量中保存对话框中指定的日期：10(月)20(日)，2005。

31.11 算术运算指令

31.11.1 ADD 和 ADDP(加)

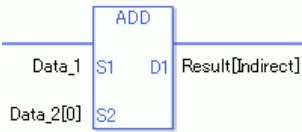
符号和功能

梯形图指令名称	梯形图符号	功能	步数
ADD (加 - 电平)		运算	4 至 13
梯形图指令名称	梯形图符号	功能	步数
ADDP (加 - 上升沿)		运算	4 至 13

■ 操作数设置

下面描述了 ADD 和 ADDP 指令操作数 (S1、S2 和 D1) 的可指定内容。
ADD 和 ADDP 指令的实际步数取决于如何指定操作数的值。下面描述如何计算步数。
操作数 S1 的步数 + 操作数 S2 的步数 + 操作数 D1 的步数 +1 = 一个指令的总步数

例如，计算 ADD 和 ADDP 指令的步数
(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {Data_2[0] = 2 步} + {Result[Indirect] = 3 步} + {1 步} = 7 步。
指令的总步数需要加最后一步。请务必加一步。

■ 操作数设置

下面描述了 ADD 和 ADDP 指令中操作数 (S1) 和 (S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]	2	O
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]	4	O
	浮点型		1	O
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型		1	O
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]	2	O
		D_****.B/W [地址]	3	O
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647	1	O
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38	1	O
	实型	± 1.175494351e308 至 ± 1.7976931348623158e+308	2	O

■ 操作数设置

下面描述了 ADD 和 ADDP 指令中操作数 (D1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (仅输出)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]	2	O
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]	4	O
	浮点型		1	O
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型		1	O
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]	2	O
		D_****.B/W [地址]	3	O
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量				X

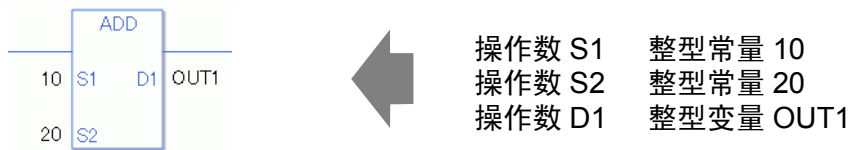
■ ADD 和 ADDP 指令说明

ADD 和 ADDP 指令是加法指令。当执行 ADD 指令时，将 S1 加到 S2，并将结果保存在 D1 中。

ADD 和 ADDP 指令总是接通。使用 ADD 和 ADDP 指令时，如果操作数 S1、S2 和 D1 指定的变量类型不同，就会发生错误。请为操作数 S1、S2 和 D1 指定相同的变量类型。

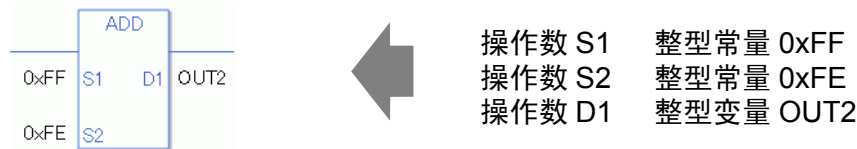
指定常量时，请参考如下内容。

当操作数 D1 是整型变量时



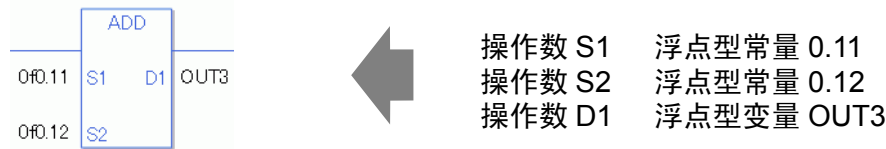
当操作数 D1 是整型变量，而您又想在操作数 S1 和 S2 中输入 16 进制值时。

当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



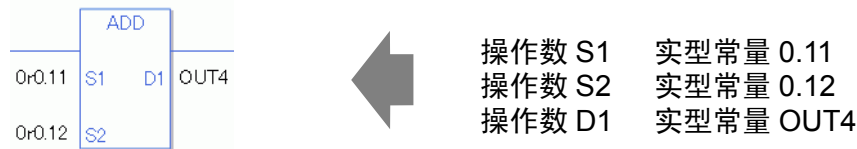
当操作数 D1 是浮点型变量时

当输入 0f(0 和小写字母 “f”) 时，将把后面的值视为浮点型数值。



当操作数 D1 是实型变量时

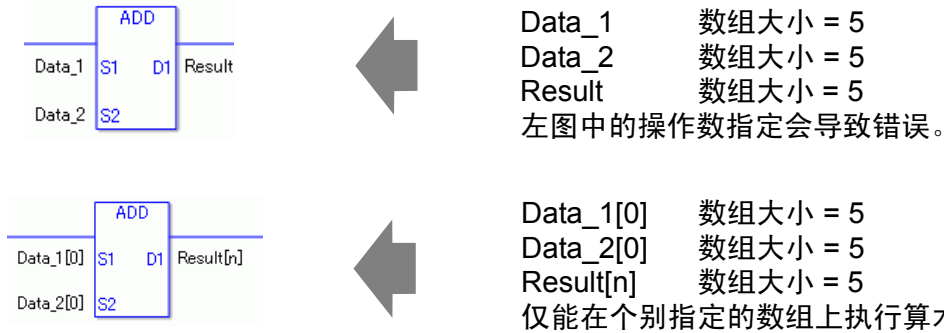
当输入 0r(0 和小写字母 “r”) 时，将把后面的值视为实数值。



当在指定数组 (整型变量数组) 中添加数据时

使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。

如果将所有操作数 S1、S2 和 D1 指定为整个数组，会发生错误，即使指定的变量是同一类型。



■ 确认执行结果

- (1)如果由于指令的执行造成了溢出错误，系统变量 (位)#L_CalcCarry 置 ON。
- (2)如果无法识别操作数 S1 或 S2 中的值 (无限或非数值)，将不执行指令。并将错误代码 (6706) 写入 #L_CalcErrCode。
D1 中的结果维持上次成功运行指令而得出的值。
- (3)#L_Error 置 ON，并将错误代码 (6706) 写入 #L_CalcErrCode。
- (4)当执行结果为 0 时，系统变量 #L_CalcZero 置 ON。

(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

ADD

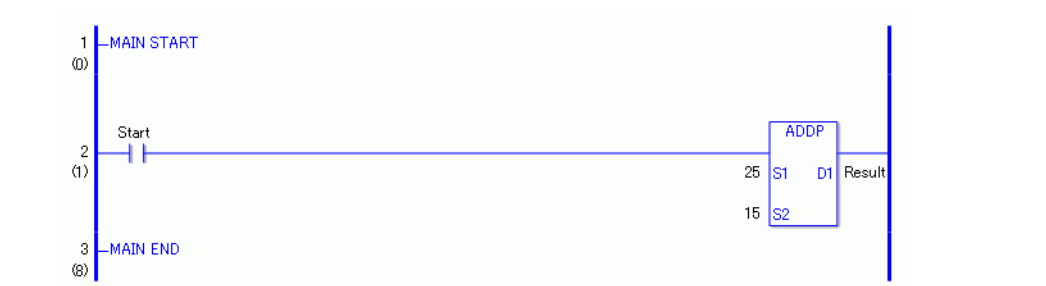
将一个常量加到另外一个常量上并将结果保存到整型变量中。



- (1)当运算中的上升沿指令置 ON 时，将执行 ADD 指令。当执行 ADD 指令时，在 D1 中保存结果值 40(从 $25 + 15 = 40$ 计算中得来)。
- 当运算是常开指令时，只要变量为 ON，就总是执行 ADD 指令。

程序示例

ADDP

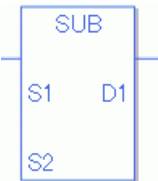
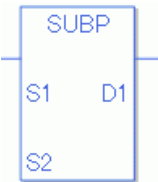


- (1)当常开指令置 ON 时，将执行 ADDP 指令。当执行 ADDP 指令时，在 D1 中保存结果值 40(从 $25 + 15 = 40$ 计算而来)。
- 即使使用的是一个常开指令，也只有在检测到上升沿时才执行 ADDP 指令。

因此，即使常开指令的变量总是为 ON，也只在第一次扫描中执行 ADDP 指令。

31.11.2 SUB 和 SUBP(减)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
SUB (减 - 电平)		运算	4 至 13
梯形图指令名称	梯形图符号	功能	步数
SUBP (减 - 上升沿)		运算	4 至 13

■ 操作数设置

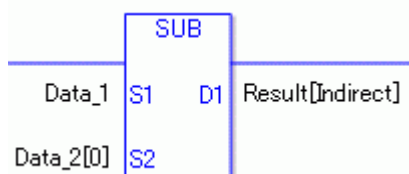
下面描述 SUB 和 SUBP 指令操作数 (S1、S2 和 D1) 的可指定内容。

SUB 和 SUBP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 S2 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 SUB 和 SUBP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {Data_2[0] = 2 步} + {Result[Indirect] = 3 步} + {1 步} = 7 步。
指令的总步数需要加最后一步。请务必加一步。

■ 操作数设置

下面描述了 SUB 和 SUBP 指令中操作数 (S1) 和 (S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]	2	O
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]	4	O
	浮点型		1	O
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型		1	O
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]	2	O
		D_****.B/W [地址]	3	O
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647	1	O
	浮点型	$\pm 1.175494351\text{e-}38$ 至 $\pm 3.402823466\text{e+}38$	1	O
	实型	$\pm 1.175494351\text{e-}308$ 至 $\pm 1.7976931348623158\text{e+}308$	2	O

■ 操作数设置

下面描述了 SUB 和 SUBP 指令中操作数 (D1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (仅输出)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]	2	O
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]	4	O
	浮点型		1	O
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型		1	O
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]	2	O
		D_****.B/W [地址]	3	O
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量				X

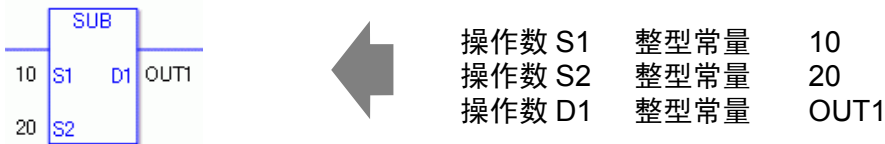
■ SUB 和 SUBP 指令说明

SUB 和 SUBP 指令是减法指令。当执行 SUB 指令时，将从 S2 中减去 S1，并将结果保存到 D1 中。

SUB 和 SUBP 指令总是接通。当使用 SUB 和 SUBP 指令时，如果在操作数 S1、S2 和 D1 中指定的变量类型不同，就会发生错误。请为操作数 S1、S2 和 D1 指定相同的变量类型。

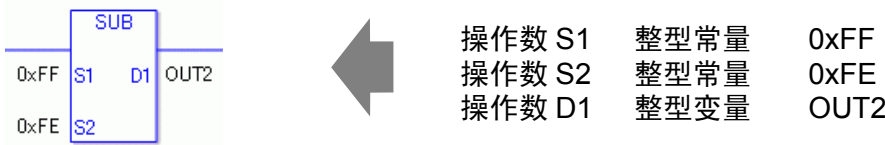
指定常量时，请参考如下内容。

当操作数 D1 是整型变量时



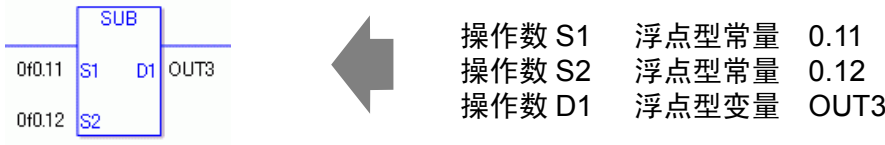
当操作数 D1 是整型变量，而您又想在操作数 S1 和 S2 中输入 16 进制值时。

当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



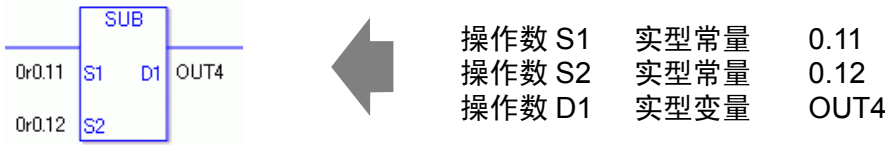
当操作数 D1 是浮点型变量时

当输入 0f(0 和小写字母 “f”) 时，将把后面的值视为浮点型数值。



当操作数 D1 是实型变量时

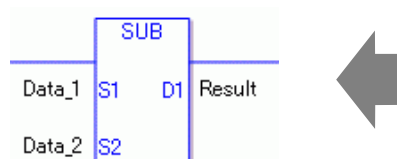
当输入 0r(0 和小写字母 “r”) 时，将把后面的值视为实数值。



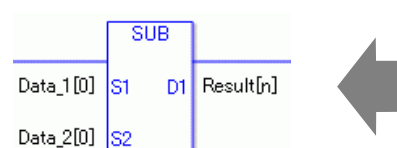
当对指定数组 (整型变量数组) 中的数据执行减法运算时

使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。

如果将所有操作数 S1、S2 和 D1 指定为整个数组，会发生错误，即使指定的变量是同一类型。



Data_1 数组大小 = 5
Data_2 数组大小 = 5
Result 数组大小 = 5
左图中的操作数指定会导致错误。



Data_1[0] 数组大小 = 5
Data_2[0] 数组大小 = 5
Result[n] 数组大小 = 5
仅能在个别指定的数组上执行算术运算。

■ 确认执行结果

- (1)如果由于指令的执行造成了溢出错误，系统变量 (位)#L_CalcCarry 置 ON。
- (2)如果无法识别操作数 S1 或 S2 中的值 (无限或非数值)，将不执行指令。错误检查发现错误时，会将错误代码 “6706” 设置为 #L_CalcErrCode。
输出结果 D1 维持上次成功执行指令时得到的值。
- (3)#L_Error 置 ON，并将错误代码 (6706) 写入 #L_CalcErrCode。
- (4)当执行结果为 0 时，系统变量 #L_CalcZero 置 ON。

(注释)

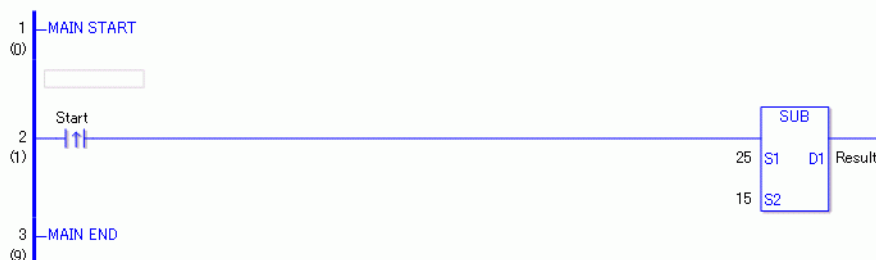
当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

SUB

从一个常量中减去另一个常量并将结果保存到整型变量中。

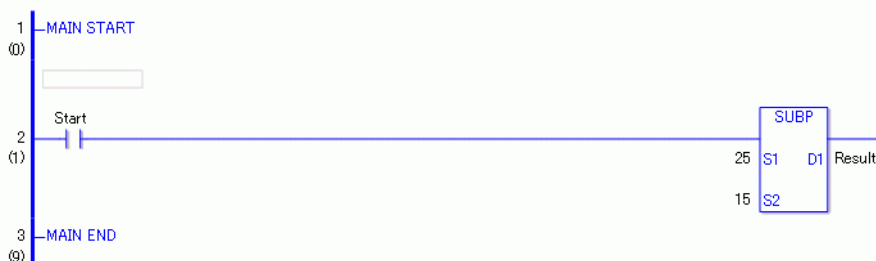


(1)当上升沿指令置 ON 时，将执行 SUB 指令。当执行 SUB 指令时，在 D1 中保存结果值 10(从 $25-15 = 10$ 计算而来)。

当使用常开指令时，只要变量为 ON，就总是执行 SUB 指令。

程序示例

SUBP



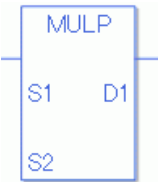
(1)当常开指令置 ON 时，将执行 SUBP 指令。当执行 SUBP 指令时，在 D1 中保存结果值 10 (从 $25-15 = 10$ 计算而来)。

即使使用常开指令，也只有在检测到上升沿时才执行 SUBP 指令。

因此，即便常开指令总是为 ON，也只在一次扫描中执行 SUBP 指令。

31.11.3 MUL 和 MULP(乘)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
MUL (乘 - 电平)		运算	4 至 13
梯形图指令名称	梯形图符号	功能	步数
MULP (乘 - 上升沿)		运算	4 至 13

■ 操作数设置

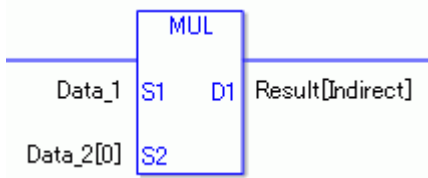
下面描述 MUL 和 MULP 指令操作数 (S1、 S2 和 D1) 的可指定内容。

MUL 和 MULP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 S2 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 MUL 和 MULP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步 } + {Data_2[0] = 2 步 } + {Result[Indirect] = 3 步 } + {1 步 } = 7 步。

指令的总步数需要加最后一步。请务必加一步。

■ 操作数设置

下面描述了 MUL 和 MULD 指令操作数 (S1) 和 (S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]	2	O
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]	4	O
	浮点型		1	O
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型		1	O
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：○ 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	○
	Q_		1	○
	D_	未指定修饰符	1	○
		D_****.B/W [常量]	2	○
		D_****.B/W [地址]	3	○
	F_		1	○
	R_		1	○
	T_	仅 .PT / .ET	2	○
	C_	仅 .PV / .CV	2	○
	N_	仅 .YR / .MO / .DAY	2	○
	J_	仅 .HR / .MIN / .SEC	2	○
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	○
常量	整型	-2147483648 至 2147483647	1	○
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38	1	○
	实型	± 1.175494351e-308 至 ± 1.7976931348623158e+308	2	○

■ 操作数设置

下面描述了 MUL 和 Mulp 指令操作数 (D1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (仅输出)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]	2	O
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]	4	O
	浮点型		1	O
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型		1	O
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]	2	O
		D_****.B/W [地址]	3	O
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量				X

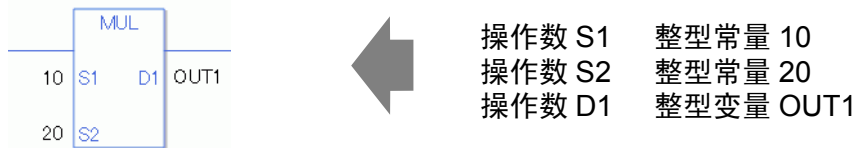
MUL 和 MULP 指令说明

MUL 和 MULP 指令是乘法指令。当执行 MUL 指令时，S1 将与 S2 相乘，结果保存在 D1 中。

MUL 和 MULP 指令总是接通电源。当使用 MUL 和 MULP 指令时，如果为操作数 S1、S2 和 D1 指定的变量类型不同，就会发生错误。请为操作数 S1、S2 和 D1 指定相同的变量类型。

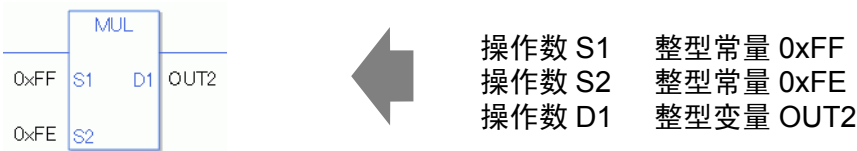
指定常量时，请参考如下内容。

当操作数 D1 是整型变量时



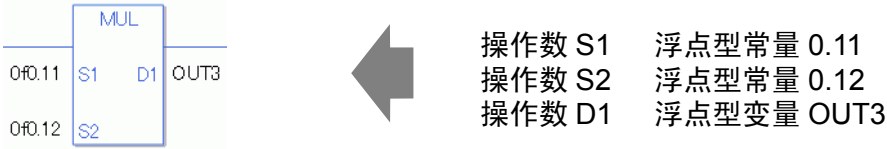
当操作数 D1 是整型变量，而您又想在操作数 S1 和 S2 中输入 16 进制值时。

当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



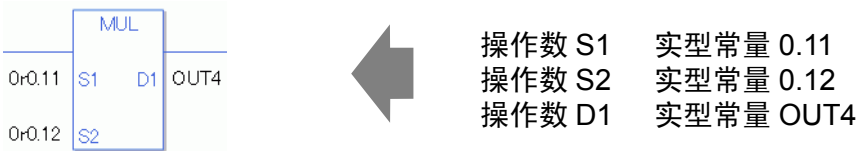
当操作数 D1 是浮点型变量时

当输入 0f(0 和小写字母 “f”) 时，将把后面的值视为浮点型数值。

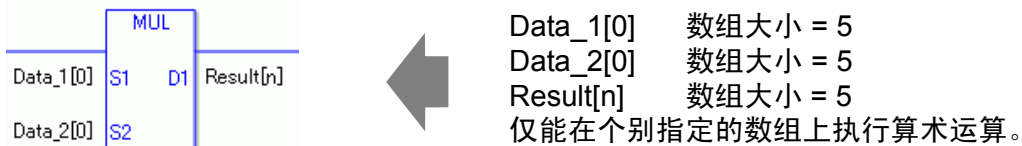


当操作数 D1 是实型变量时

当输入 0r(0 和小写字母 “r”) 时，将把后面的值视为实数值。



当乘以指定数组 (整型变量数组) 中的数据时
使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。
如果将所有操作数 S1、 S2 和 D1 指定为整个数组，会发生错误，即使指定的变量是同一类型。



■ 确认执行结果

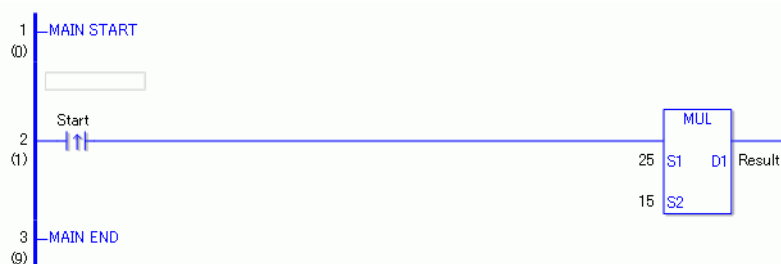
- (1)如果由于指令的执行造成了溢出错误，系统变量 (位)#L_CalcCarry 置 ON。
- (2)如果无法识别操作数 S1 或 S2 中的值 (无限或非数值)，将不执行指令。错误检查发现错误时，会将错误代码 “6706” 设置为 #L_CalcErrCode。
输出结果 D1 维持上次成功执行指令时得到的值。
- (3)#L_Error 置 ON，并将错误代码 (6706) 写入 #L_CalcErrCode。
- (4)当执行结果为 0 时，系统变量 #L_CalcZero 置 ON。

(注释)
当使用系统变量检查结果时，确保在执行完指令后进行。
当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

MUL

用一个常量乘以另外一个常量并将结果保存在整型变量中。

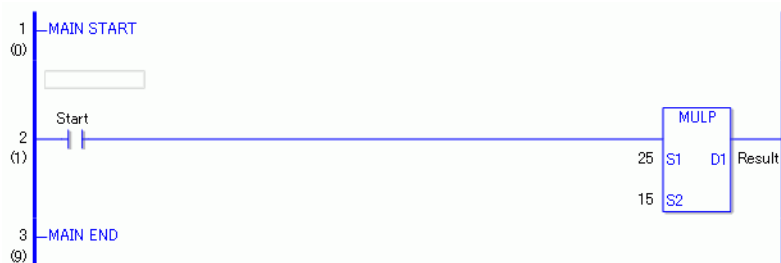


(1)当上升沿指令置 ON 时，将执行 MUL 指令。当执行 MUL 指令时，在 D1 中保存结果 375(从 $25 \times 15 = 375$ 计算而来)。

当使用常开指令时，只要指令变量为 ON，就总是执行 MUL 指令。

程序示例

MULP

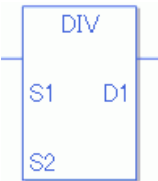
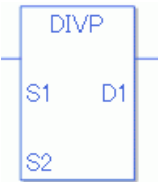


当常开指令置 ON 时，将执行 MULP 指令。当执行 MULP 指令时，在 D1 中保存结果 375(从 $25 \times 15 = 375$ 计算而来)。

即使使用常开指令，也只有在检测到上升沿时才会执行 MULP 指令。因此，即使常开指令的变量总是为 ON，也只在一次扫描中执行 MULP 指令。

31.11.4 DIV 和 DIVP(除)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
DIV (除 - 电平)		运算	4 至 13
梯形图指令名称	梯形图符号	功能	步数
DIVP (除 - 上升沿)		运算	4 至 13

■ 操作数设置

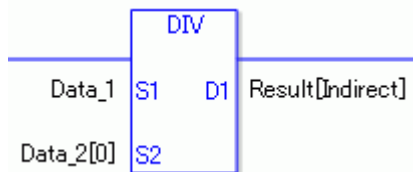
下面描述 DIV 和 DIVP 指令操作数 (S1、 S2 和 D1) 的可指定内容。

DIV 和 DIVP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 S2 的步数 + 操作数 D1 的步数 +1 = 一个指令的总步数

例如，计算 DIV 和 DIVP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {Data_2[0] = 2 步} + {Result[Indirect] = 3 步} + {1 步} = 7 步。
指令的总步数需要加最后一步。请务必加 1 步。

■ 操作数设置

下面描述了 DIV 和 DIVP 指令操作数 (S1) 和 (S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]	2	O
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]	4	O
	浮点型		1	O
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型		1	O
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：○ 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	○
	Q_		1	○
	D_	未指定修饰符	1	○
		D_****.B/W [常量]	2	○
		D_****.B/W [地址]	3	○
	F_		1	○
	R_		1	○
	T_	仅 .PT / .ET	2	○
	C_	仅 .PV / .CV	2	○
	N_	仅 .YR / .MO / .DAY	2	○
	J_	仅 .HR / .MIN / .SEC	2	○
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	○
常量	整型	-2147483648 至 2147483647	1	○
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38	1	○
	实型	± 1.175494351e-308 至 ± 1.7976931348623158e+308	2	○

■ 操作数设置

下面描述了 DIV 和 DIVP 指令中操作数 (D1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (仅输出)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]	2	O
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]	4	O
	浮点型		1	O
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型		1	O
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]	2	O
		D_****.B/W [地址]	3	O
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量				X

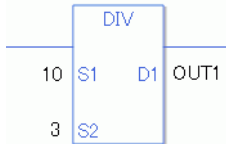
■ DIV 和 DIVP 指令说明

DIV 和 DIVP 指令是除法指令。当执行 DIV 指令时，S1 将除以 S2，并将结果保存在 D1 中。

DIV 和 DIVP 指令总是接通电源。当使用 DIV 和 DIVP 指令时，如果为操作数 S1、S2 和 D1 指定的变量类型不同，就会发生错误。请为操作数 S1、S2 和 D1 指定相同的变量类型。

指定常量时，请参考如下内容。

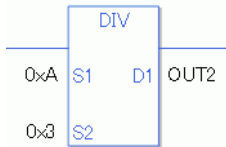
当操作数 D1 是整型变量时



操作数 S1 整型常量 10
操作数 S2 整型常量 3
操作数 D1 整型变量 OUT1
运算结果将四舍五入到最接近的整数。
例如， $10 (S1) / 3 (S2) = 3 (D1)$

当操作数 D1 是整型变量，而您又想在操作数 S1 和 S2 中输入 16 进制值时。

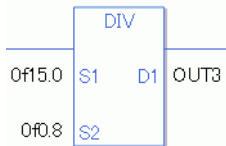
当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



操作数 S1 整型常量 0xA
操作数 S2 整型常量 0x3
操作数 D1 整型变量 OUT2
运算结果将四舍五入到最接近的整数。
例如， $0xA (S1) / 0x3 (S2) = 3 (D1)$

当操作数 D1 是浮点型变量时

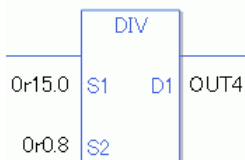
当输入 0f(0 和小写字母 “f”) 时，将把后面的值视为浮点型数值。



操作数 S1 浮点型常量 15.0
操作数 S2 浮点型常量 0.8
操作数 D1 浮点型变量 OUT3
运算结果是一个包括小数点的值。
例如， $0f 15.0 (S1) / 0f 0.8 (S2) = 18.75 (D1)$

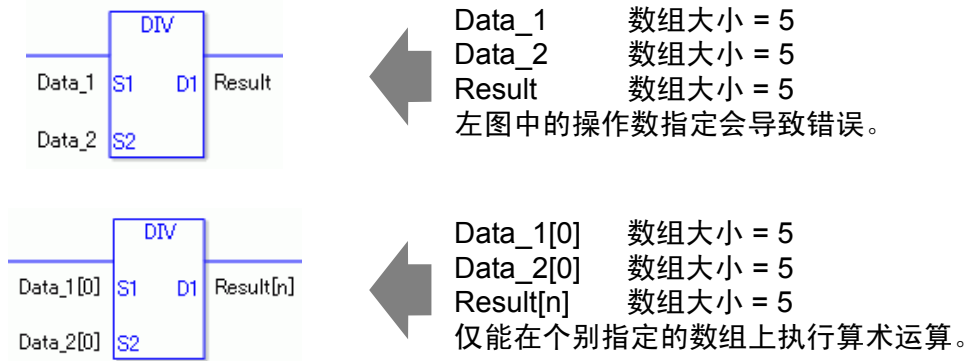
当操作数 D1 是实型变量时

当输入 0r(0 和小写字母 “r”) 时，将把后面的值视为实数值。



操作数 S1 浮点型常量 15.0
操作数 S2 浮点型常量 0.8
操作数 D1 浮点型变量 OUT4
运算结果是一个包括小数点的值。
例如， $0r15.0 (S1) / 0r0.8 (S2) = 18.75 (D1)$

当除以指定数组 (整型变量数组) 中的数据时
使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。
如果将所有操作数 S1、 S2 和 D1 指定为整个数组，会发生错误，即使指定的变量是同一类型。



■ 确认执行结果

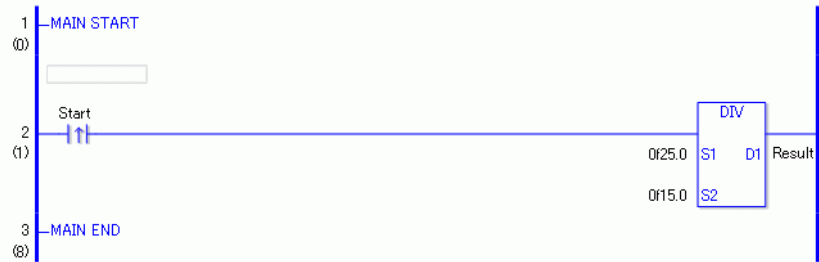
- (1)如果由于指令的执行造成了溢出错误，系统变量 (位)#L_CalcCarry 置 ON。
- (2)如果无法识别操作数 S1 或 S2 中的值 (无限或非数值)，将不执行指令。错误检查发现错误时，会将错误代码 “6706” 设置为 #L_CalcErrCode。
输出结果 D1 维持上次成功执行指令时得到的值。
- (3)#L_Error 置 ON，并将错误代码 (6706) 写入 #L_CalcErrCode。
- (4)当执行结果为 0 时，系统变量 #L_CalcZero 置 ON。

(注释)
当使用系统变量检查结果时，确保在执行完指令后进行。
当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

DIV

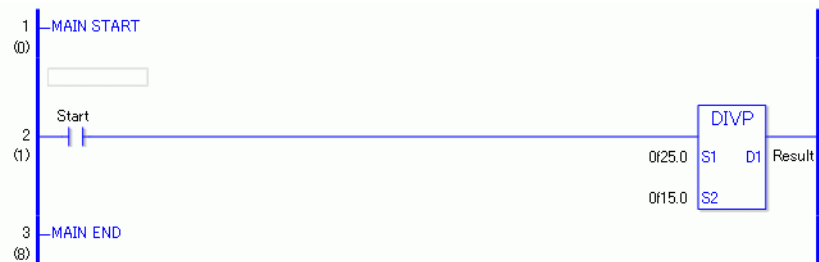
用一个常量除另外一个常量并将结果保存在浮点型变量中。



- (1)当上升沿指令置 ON 时，将执行 DIV 指令。当执行 DIV 指令时，在 D1 的结果数据 (浮点型变量) 中保存结果 1.66666(从 $25/15 = 1.66666$ 计算而来)。如果数值不能被整除，会将结果四舍五入到最接近的位。
当使用常开指令时，只要指令变量为 ON，就总是执行 DIV 指令。

程序示例

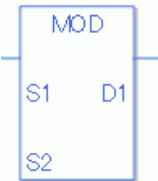
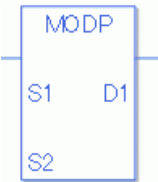
DIVP



- (1)当常开指令置 ON 时，将执行 DIVP 指令。当执行 DIVP 指令时，在 D1 的结果数据 (浮点型变量) 中保存结果 1.66666(从 $25/15 = 1.66666$ 计算而来)。如果数值不能被整除，会将结果四舍五入到最接近的位。
即使使用常开指令，也只有在检测到上升沿时才执行 DIVP 指令。
因此，即使指令总是为 ON，也只在一次扫描中执行 DIVP 指令。

31.11.5 MOD 和 MODP(取模)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
MOD (取模 - 电平)		运算	4 至 13
梯形图指令名称	梯形图符号	功能	步数
MODP (取模 - 上升沿)		运算	4 至 13

■ 操作数设置

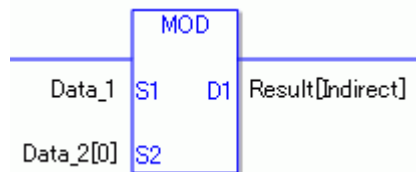
下面描述 MOD 和 MODP 指令数 (S1、S2 和 D1) 的可指定内容。

MOD 和 MODP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 S2 的步数 + 操作数 D1 的步数 +1 = 一个指令的总步数

例如，计算 MOD 和 MODP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {Data_2[0] = 2 步} + {Result[Indirect] = 3 步} + {1 步} = 7 步。
指令的总步数需要加最后一步。请务必加一步。

■ 操作数设置

下面描述了 MOD 和 MODP 指令操作数 (S1) 和 (S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]	2	O
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]	4	O
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]	2	O
		D_****.B/W [地址]	3	O
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647	1	O
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 1.175494351e-308 至 ± 1.7976931348623158e+308		X

■ 操作数设置

下面描述 MOD 和 MODP 指令中操作数 (D1) 的可指定内容。

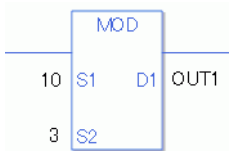
名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (仅输出)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]	2	O
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]	4	O
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：○ 不可用：×
地址格式	X_			×
	Y_			×
	M_			×
	I_			×
	Q_		1	○
	D_	未指定修饰符	1	○
		D_****.B/W [常量]	2	○
		D_****.B/W [地址]	3	○
	F_			×
	R_			×
	T_	仅 .PT / .ET	2	○
	C_	仅 .PV / .CV	2	○
	N_	仅 .YR / .MO / .DAY	2	○
	J_	仅 .HR / .MIN / .SEC	2	○
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	○
常量				×

■ MOD 和 MODP 指令说明

MOD 和 MODP 指令是取模运算。当执行 MOD 指令时，会将 S1 除以 S2，并将余数保存在 D1 中。MOD 和 MODP 指令总是处于接通状态。当使用 MOD 和 MODP 指令时，如果为操作数 S1、S2 和 D1 指定的变量类型不同，就会发生错误。请为操作数 S1、S2 和 D1 指定相同的变量类型。指定常量时，请参考如下内容。

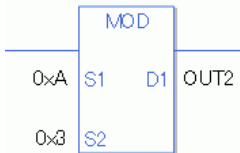
当操作数 D1 是整型变量时



操作数 S1 整型常量 10
操作数 S2 整型常量 3
操作数 D1 整型变量 OUT1
例如， $10 (S1) / 3 (S2) = 3 \text{ 余 } 1$
因此， $D1 = 1$

当操作数 D1 是整型变量，而您又想在操作数 S1 和 S2 中输入 16 进制值时。

当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。

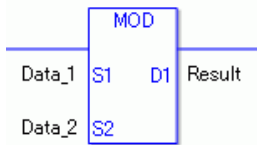


操作数 S1 整型常量 0xA
操作数 S2 整型常量 0x3
操作数 D1 整型变量 OUT2
例如， $10 (S1) / 3 (S2) = 3 \text{ 余 } 1$
因此， $D1 = 1$

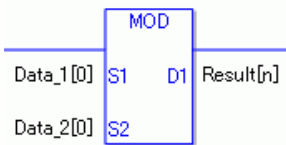
当在指定数组 (整型变量数组) 的数据上执行取模运算时

使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。

如果将所有操作数 S1、S2 和 D1 指定为整个数组，会发生错误，即使指定的变量是同一类型。



Data_1 数组大小 = 5
Data_2 数组大小 = 5
Result 数组大小 = 5
左图中的操作数指定会导致错误。



Data_1[0] 数组大小 = 5
Data_2[0] 数组大小 = 5
Result[n] 数组大小 = 5
仅能在个别指定的数组上执行算术运算。

■ 确认执行结果

- (1)如果由于指令的执行造成了溢出错误，系统变量 (位)#L_CalcCarry 置 ON。
- (2)#L_Error 置 ON，并将错误代码 (6706) 写入 #L_CalcErrCode。
- (3)当执行结果为 0 时，系统变量 #L_CalcZero 置 ON。

(注释)

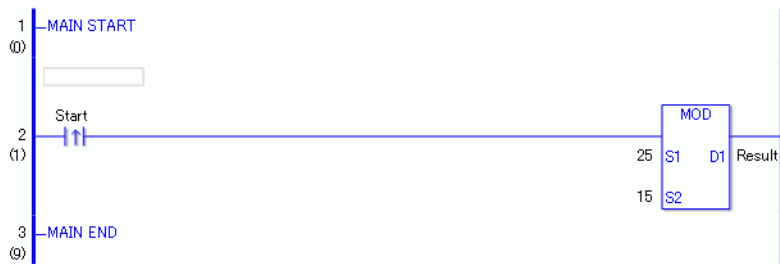
当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

MOD

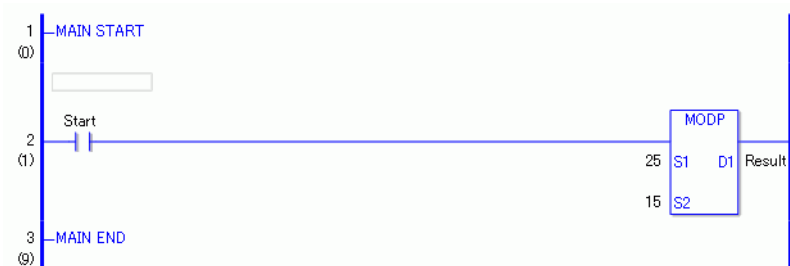
在两个常量上执行取模运算并将结果保存在整型变量中。



- (1)当上升沿指令置 ON 时，将执行 MOD 指令。当执行 MOD 指令时，在 D1 中保存结果 10(从 $25/15 = 1$ (余数 10) 计算而来)。当使用常开指令时，只要变量为 ON，就总是执行 MOD 指令。

程序示例

MODP



- (1) 当常开指令置 ON 时，将执行 MODP 指令。当执行 MODP 指令时，在 D1 中保存结果 10 (从 $25/15 = 1$ (余数 10) 计算而来)。
- 即使使用常开指令，也只有在检测到上升沿时才会执行 MODP 指令。
- 因此，即使常开指令总是为 ON，也只在一次扫描中执行 MODP 指令。

31.11.6 INC 和 INCP(加 1)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
INC (加 1 - 电平)		运算	2 至 4
梯形图指令名称	梯形图符号	功能	步数
INCP (加 1 - 上升沿)		运算	2 至 4

■ 操作数设置

下面描述了 INC 和 INCP 指令中操作数 (D1) 的可指定内容。

名称	类型	条件	指令步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	2	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	2	O
符号	位			X
	字		2	O

名称	类型	条件	指令步数	可用: O 不可用: X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (仅输出)	未指定数组和修饰符	2	O
		指定整型变量 [常量]	3	O
		指定整型变量 [变量]	4	O
		指定整型变量 [常量 / 变量].B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	3	O
	计数器	仅 .PV / .CV	3	O
	日期	仅 .YR / .MO / .DAY	3	O
	时间	仅 .HR / .MIN / .SEC	3	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	3	O
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	3	O
	C_	仅 .PV / .CV	3	O
	N_	仅 .YR / .MO / .DAY	3	O
	J_	仅 .HR / .MIN / .SEC	3	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	3	O
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 1.175494351e-308 至 ± 1.7976931348623158e+308		X

■ INC 和 INCP 指令说明

INC 和 INCP 指令是加 1 指令。当执行 INC 指令时，D1 将加 1。
INC 和 INCP 指令总是处于接通状态。

■ 确认执行结果

- (1)如果由于指令的执行造成了溢出错误，系统变量 (位)#L_CalcCarry 置 ON。
- (2)当执行结果为 0 时，系统变量 #L_CalcZero 置 ON。

(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

INC

每次 INC 指令置 ON 时，就加 1。



- (1)当上升沿指令置 ON 时，将执行 INC 指令。当执行 INC 指令时，D1 中的结果数据 (整型变量) 加 1。当使用常开指令时，只要指令接通，INC 指令将在每次扫描中不断执行加 1 操作。

程序示例

INCP



(1)当常开指令置 ON 时，将执行 INCP 指令。当执行 INCP 指令时，D1 中的结果数据 (整型变量) 加 1。

即使使用常开指令，也只有在检测到上升沿时才会执行 INCP 指令。因此，即使常开指令位保持为 ON，也只在一次扫描中执行 INCP 指令并从 D1(整型变量) 中加 1。

31.11.7 DEC 和 DECP(减 1)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
DEC (减 1 - 电平)		运算	2 至 4
梯形图指令名称	梯形图符号	功能	步数
DECP (减 1 - 上升沿)		运算	2 至 4

■ 操作数设置

下面描述了 DEC 和 DECP 指令中操作数 (D1) 的可指定内容。

名称	类型	条件	指令 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	2	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	2	O
符号	位			X
	字		2	O

名称	类型	条件	指令 步数	可用: O 不可用: X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (仅输出)	未指定数组和修饰符	2	O
		指定整型变量 [常量]	3	O
		指定整型变量 [变量]	4	O
		指定整型变量 [常量 / 变量].B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	3	O
	计数器	仅 .PV / .CV	3	O
	日期	仅 .YR / .MO / .DAY	3	O
	时间	仅 .HR / .MIN / .SEC	3	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	3	O
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	3	O
	C_	仅 .PV / .CV	3	O
	N_	仅 .YR / .MO / .DAY	3	O
	J_	仅 .HR / .MIN / .SEC	3	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	3	O
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 1.175494351e-308 至 ± 1.7976931348623158e+308		X

■ DEC 和 DECP 指令说明

DEC 和 DECP 指令是减 1 指令。当运行 DEC 指令时，将从 D1 中减 1。
DEC 和 DECP 指令总是处于接通状态。

■ 确认执行结果

- (1)如果由于指令的执行造成了溢出错误，系统变量 (位)#L_CalcCarry 置 ON。
- (2)当执行结果为 0 时，系统变量 #L_CalcZero 置 ON。

(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

DEC

每次 DEC 指令被置 ON 时，就减去 1。



- (1)当上升沿指令置 ON 时，将执行 DEC 指令。当执行 DEC 指令时，将从 D1(整型变量) 中减去 1。当使用常开指令时，只要指令接通， DEC 指令就会在每次扫描时不断执行并从 D1 中减 1。

程序示例

DECP



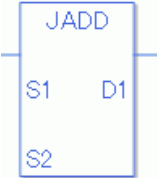
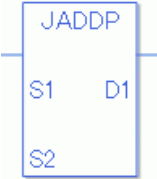
(1)当常开指令置 ON 时，将执行 DECP 指令。当 DECP 指令执行时，将从 D1(整型变量) 中减去 1。

即使使用常开指令，也只有在检测到上升沿时才会执行 DECP 指令。因此，即使常开指令位保持为 ON，也只在一次扫描中执行 DECP 指令并从 D1(整型变量) 中减去 1。

31.12 时间运算指令

31.12.1 JADD 和 JADDP(时间加)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
JADD (时间加 - 电平)		运算	4
梯形图指令名称	梯形图符号	功能	步数
JADDP (时间加 - 上升沿)		运算	4

■ 操作数设置

下面描述 JADD 和 JADDP 指令操作数 (S1、 S2 和 D1) 的可指定内容。

名称	类型	条件	指令步数	可用：○ 不可用：X
外部寄存器地址	位			X
	字	仅按字指定 (例如， [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如， [#INTERNAL]LS0000)		X
符号	位			X
	字			X

名称	类型	条件	指令 步数	可用：○ 不可用：×
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (仅输出)	未指定数组和修饰符		X
		指定整型变量 [常量]		X
		指定整型变量 [变量]		X
		指定整型变量 [常量 / 变量].B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	非 .HR / .MIN / .SEC	4	○
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	非 .HR / .MIN / .SEC	4	○
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
	常量	整型	-2147483648 至 2147483647	X
		浮点型	± 1.175494351e-38 至 ± 3.402823466e+38	X
		实型	± 1.175494351e-308 至 ± 1.7976931348623158e+308	X

■ JADD 和 JADDP 指令说明

JADD 和 JADDP 指令是时间加指令。当执行 JADD 指令时，会将操作数 S1 中的时间变量加到 S2 中的时间变量，并将结果保存在操作数 D1 中的时间变量中。JADD 和 JADDP 指令总是处于接通状态。

时间变量

时间变量	变量设置	描述
变量名 .HR	整型变量	以 BCD 形式输入小时。
变量名 .MIN	整型变量	以 BCD 形式输入分钟。
变量名 .SEC	整型变量	以 BCD 形式输入秒。

在 JADD 指令中，您不能对单个时间变量元素 (.HR .MIN .SEC) 执行时间加运算。时间变量及其中的每个元素均以 BCD 数据形式进行保存。

■ 确认执行结果

- (1)如果在执行完指令后结果达到 00:00'00"，将发生溢出。系统变量 (位)
#L_CalcCarry 置 ON。
- (2)当运算结果为 00:00'00" 时，系统变量 #L_CalcZero 置 ON。

(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

JADD

当上升沿指令置 ON 时，将执行时间加运算。

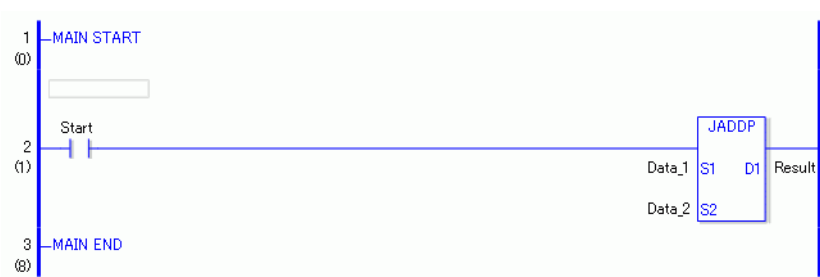


(1)当上升沿指令置 ON 时，将执行 JADD 指令。当执行 JADD 指令时，操作数 S1 中的 Data_1(时间变量) 与操作数 S2 中的 Data_2(时间变量) 相加，并将结果保存在操作数 D1 中。当使用常开指令时，只要指令接通， JADD 指令将在每次扫描时不断执行时间加运算。

例如，当操作数 S1 中的 Data_1 是 12:10:45，操作数 S2 中的 Data_2 是 6:55:20，如果执行 JADD 指令，结果就是 19:06:05，并将 19:06:05 保存在操作数 D1 中的结果数据中。

程序示例

JADDP

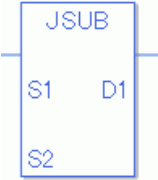
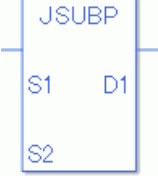


(1)当常开指令置 ON 时，将执行 JADDP 指令。当执行 JADDP 指令时，操作数 S1 中的 Data_1(时间变量) 与操作数 S2 中的 Data_2(时间变量) 相加，并将结果保存在操作数 D1 中。即使使用常开指令，也只有在检测到向上升沿时才执行 JADDP 指令。

因此，即使常开指令的变量总是为 ON，也只在一次扫描中执行 JADDP 指令。

31.12.2 JSUB 和 JSUBP(时间减)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
JSUB (时间减 - 电平)		运算	4
梯形图指令名称	梯形图符号	功能	步数
JSUBP (时间减 - 上升沿)		运算	4

■ 操作数设置

下面描述 JSUB 和 JSUBP 指令操作数 (S1、 S2 和 D1) 的可指定内容。

名称	类型	条件	操作数 步数	可用：○ 不可用：X
外部寄存器地址	位			X
	字	仅按字指定 (例如， [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如， [#INTERNAL]LS0000)		X
符号	位			X
	字			X

名称	类型	条件	操作数 步数	可用：○ 不可用：X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (仅输出)	未指定数组和修饰符		X
		指定整型变量 [常量]		X
		指定整型变量 [变量]		X
		指定整型变量 [常量 / 变量].B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	非 .HR / .MIN / .SEC	4	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	非 .HR / .MIN / .SEC	4	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 1.175494351e-308 至 ± 1.7976931348623158e+308		X

■ JSUB 和 JSUBP 指令说明

JSUB 和 JSUBP 指令是时间减指令。当执行 JSUB 指令时，会从操作数 S1 中的时间变量中减去操作数 S2 中的时间变量，并将结果保存在操作数 D1 中的时间变量中。JSUB 和 JSUBP 指令总是处于接通状态。

时间变量

时间变量	变量设置	描述
变量名 .HR	整型变量	以 BCD 形式输入小时。
变量名 .MIN	整型变量	以 BCD 形式输入分钟。
变量名 .SEC	整型变量	以 BCD 形式输入秒。

在 JSUB 指令中，您不能对单个时间变量元素 (.HR .MIN .SEC) 执行时间减运算。时间变量及其中的每个元素均以 BCD 数据形式进行保存。

■ 确认执行结果

(1)如果在执行完指令后结果未达到 00:00'00"，将发生溢出。系统变量 (位)
#L_CalcCarry 置 ON。
如果运算结果是 00(h):00(min):00(s)，系统变量 #L_CalcZero 置 ON。

(注释)

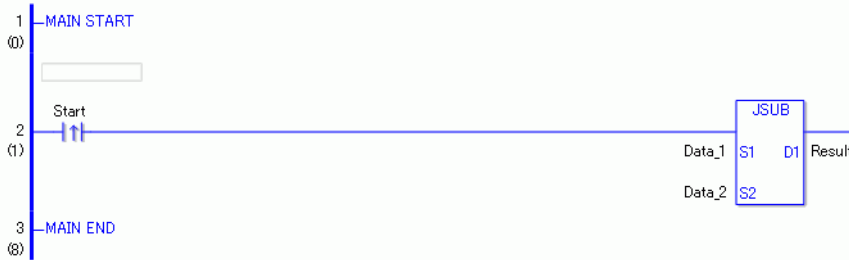
当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

JSUB

当上升沿指令置 ON 时，将执行时间减运算。

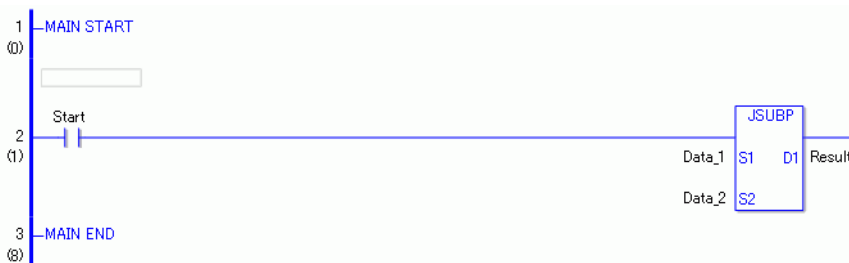


(1)当上升沿指令置 ON 时，将执行 JSUB 指令。当执行 JSUB 指令时，会从操作数 S1 的 Data_2(时间变量) 中减去操作数 S2 的 Data_1(时间变量)，并将结果保存在操作数 D1 中。当使用常开指令时，每次扫描到 JSUB 指令的接通状态，将执行日期 / 时间计算。

例如，当操作数 S1 中的 Data_1 是 12:10:45，操作数 S2 中的 Data_2 是 6:55:20，如果执行 JSUB 指令，结果就是 05:15:25，并将 05:15:25 保存在操作数 D1 的结果数据中。

程序示例

JSUBP



(1)当常开指令置 ON 时，将执行 JSUBP 指令。当执行 JSUBP 指令时，会从操作数 S1 的 Data_2(时间变量) 中减去操作数 S2 的 Data_1(时间变量)，并将结果保存在操作数 D1 中。即使使用常开指令，也只有在检测到向上升沿时才执行 JSUBP 指令。

因此，即使常开指令的变量总是为 ON，也只在一次扫描中执行 JSUBP 指令。

31.13 逻辑运算指令

31.13.1 AND 和 ANDP(逻辑与)

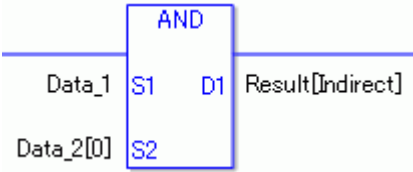
符号和功能

梯形图指令名称	梯形图符号	功能	步数
AND (逻辑与 - 电平)		运算	4 至 13
梯形图指令名称	梯形图符号	功能	步数
ANDP (逻辑与 - 上升沿)		运算	4 至 13

■ 操作数设置

下面描述了 AND 和 ANDP 指令操作数 (S1、 S2 和 D1) 的可指定内容。
AND 和 ANDP 指令的实际步数取决于操作数的指定方法。下面描述如何计算步数。
操作数 S1 的步数 + 操作数 S2 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 AND 和 ANDP 指令的步数
(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步 } + {Data_2[0] = 2 步 } + {Result[Indirect] = 3 步 } + {1 步 } = 7 步。
指令的总步数需要加最后一步。请务必加一步。

■ 操作数设置

下面描述了 AND 和 ANDP 指令操作数 (S1 和 S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: ○ 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	○
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	○
符号	位			X
	字		1	○
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	○
		指定整型变量 [常量] 或 指定整型变量 B/W [常量] 指定整型数组 (整个数组)	2	○
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]	3	○
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]	4	○
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	○
	计数器	仅 .PV / .CV	2	○
	日期	仅 .YR / .MO / .DAY	2	○
	时间	仅 .HR / .MIN / .SEC	2	○
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	○

名称	类型	条件	操作数 步数	可用：○ 不可用：×
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	○
	Q_		1	○
	D_	未指定修饰符	1	○
		D_****.B/W [常量]	2	○
		D_****.B/W [地址]	3	○
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	○
	C_	仅 .PV / .CV	2	○
	N_	仅 .YR / .MO / .DAY	2	○
	J_	仅 .HR / .MIN / .SEC	2	○
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	○
常量	整型	-2147483648 至 2147483647	1	○

■ 操作数设置

下面描述了 AND 和 ANDP 指令中操作数 (D1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (仅输出)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或 指定整型变量 B/W [常量] 指定整型数组 (整个数组)	2	O
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]	3	O
		指定整型变量 [常量 / 变量].B/W [常量 / 变量]	4	O
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]	2	O
		D_****.B/W [地址]	3	O
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647		X

■ AND 和 ANDP 指令说明

AND 和 ANDP 指令是逻辑与指令。当执行 AND 指令时，将对 S1 和 S2 执行逻辑与，并将结果保存在 D1 中。

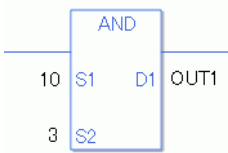
AND 和 ANDP 指令总是处于接通状态。当使用 AND 和 ANDP 指令时，如果为操作数 S1、S2 和 D1 指定的变量类型不同，就会发生错误。请为操作数 S1、S2 和 D1 指定相同的变量类型。

指定常量时，请参考如下内容。

S1	运算符	S2	D1
OFF	AND	OFF	OFF
ON		OFF	OFF
OFF		ON	OFF
ON		ON	ON

当执行 AND 指令时，仅当 S1 和 S2 均为 ON 时，D1 才置 ON。否则，D1 位为 OFF。

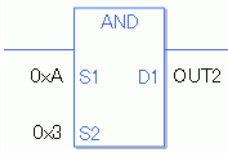
当操作数 D1 是整型变量时



操作数 S1 整型常量 10
操作数 S2 整型常量 3
操作数 D1 整型变量 OUT1

当操作数 D1 是整型变量，而您又想在操作数 S1 和 S2 中输入 16 进制值时。

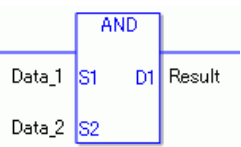
当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



操作数 S1 整型常量 0xA
操作数 S2 整型常量 0x3
操作数 D1 整型变量 OUT2

当计算指定数组 (整型变量数组) 中的数据时

指定整个数组



Data_1 数组大小 = 5
Data_2 数组大小 = 5
Result 数组大小 = 5
在整个数组上进行逻辑运算。

单个指定数组变量



Data_1[0] 数组大小 = 5
Data_2[0] 数组大小 = 5
Result[n] 数组大小 = 5
在数组中的单个变量上执行逻辑运算。

■ 确认执行结果

(1)当执行结果为 0 时，系统变量 #L_CalcZero 置 ON。

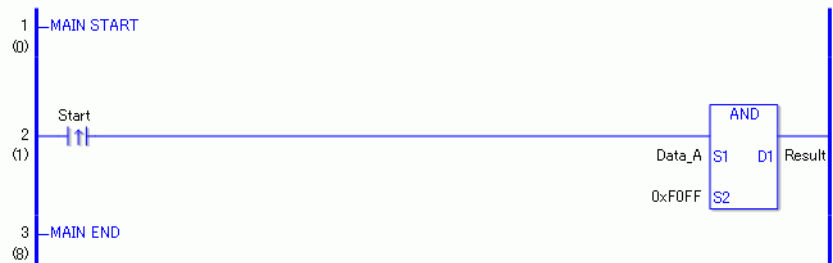
(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

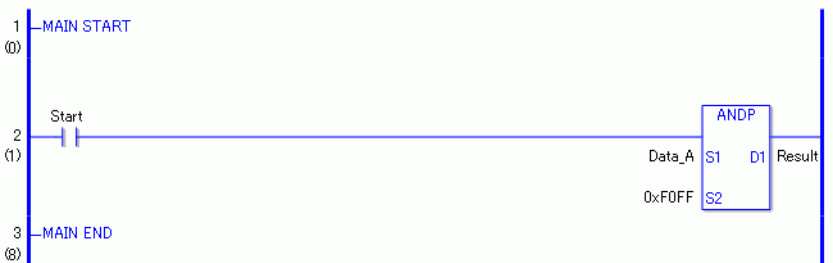
AND



(1)当上升沿指令置 ON 时，将执行 AND 指令。当执行 AND 指令时，将 Data_A 和 F0FF 执行逻辑与获得的结果保存在 D1 中。当使用常开指令时，只要指令接通，AND 指令将在每次扫描时不断执行逻辑与运算。

程序示例

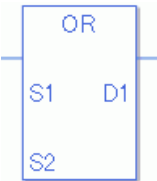
ANDP



(1)当常开指令置 ON 时，将执行 ANDP 指令。当执行 ANDP 指令时，将 Data_A 和 F0FF 执行逻辑与获得的结果保存在 D1 中。即使使用常开指令，也只有在检测到上升沿时才会执行 ANDP 指令。
因此，即便指令总是为 ON，也只在第一次扫描时执行 ANDP。

31.13.2 OR 和 ORP(逻辑或)

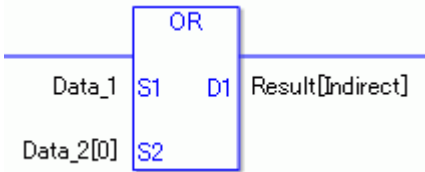
符号和功能

梯形图指令名称	梯形图符号	功能	步数
OR (逻辑或 - 电平)		运算	4 至 13
ORP (逻辑或 - 上升沿)		运算	4 至 13

■ 操作数设置

下面描述 OR 和 ORP 指令操作数 (S1、 S2 和 D1) 的可指定内容。
OR 和 ORP 指令的实际步数取决于操作数的指定方法。下面描述如何计算步数。
操作数 S1 的步数 + 操作数 S2 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 OR 和 ORP 指令的步数
(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步 } + {Data_2[0] = 2 步 } + {Result[Indirect] = 3 步 } + {1 步 } = 7 步。
指令的总步数需要加最后一步。请务必加一步。

■ 操作数设置

下面描述 OR 和 ORP 指令操作数 (S1 和 S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或 指定整型变量 B/W [常量] 指定整型数组 (整个数组)	2	O
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]	4	O
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]	2	O
		D_****.B/W [地址]	3	O
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647	1	O

■ 操作数设置

下面描述了 OR 和 ORP 指令操作数 (D1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (仅输出)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或 指定整型变量 B/W [常量] 指定整型数组 (整个数组)	2	O
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]	4	O
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：○ 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_		1	○
	D_	未指定修饰符	1	○
		D_****.B/W [常量]	2	○
		D_****.B/W [地址]	3	○
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	○
	C_	仅 .PV / .CV	2	○
	N_	仅 .YR / .MO / .DAY	2	○
	J_	仅 .HR / .MIN / .SEC	2	○
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	○
常量	整型	-2147483648 至 2147483647		X

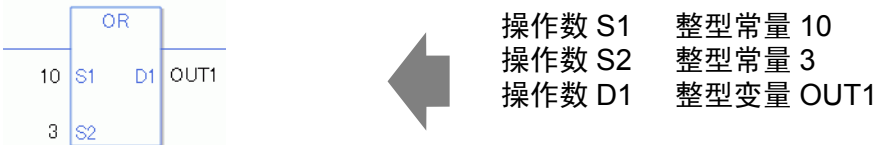
■ OR 和 ORP 指令说明

OR 和 ORP 指令是逻辑或指令。当执行 OR 指令时，将对 S1 和 S2 执行逻辑或运算，并将结果保存在 D1 中。
OR 和 ORP 指令总是处于接通状态。使用 OR 和 ORP 指令时，如果为操作数 S1、S2 和 D1 指定的变量类型不同，就会发生错误。请为操作数 S1、S2 和 D1 指定相同的变量类型。
指定常量时，请参考如下内容。

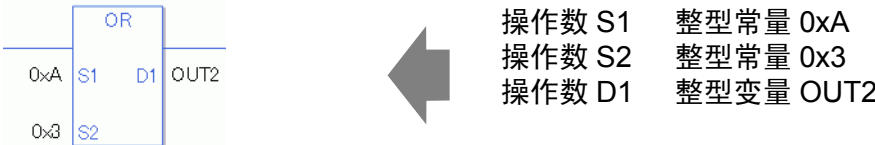
S1	运算符	S2	D1
OFF	OR	OFF	OFF
ON		OFF	ON
OFF		ON	ON
ON		ON	ON

当执行 OR 指令时，仅在 S1 和 S2 均为 OFF 时 D1 位才置 OFF。否则，D1 位为 OFF。

当操作数 D1 是整型变量时



当操作数 D1 是整型变量，而您又想在操作数 S1 和 S2 中输入 16 进制值时。
当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



当计算指定数组 (整型变量数组) 中的数据时
指定整个数组



单个指定数组变量



■ 确认执行结果

(1)当执行结果为 0 时，系统变量 #L_CalcZero 置 ON。

(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

OR

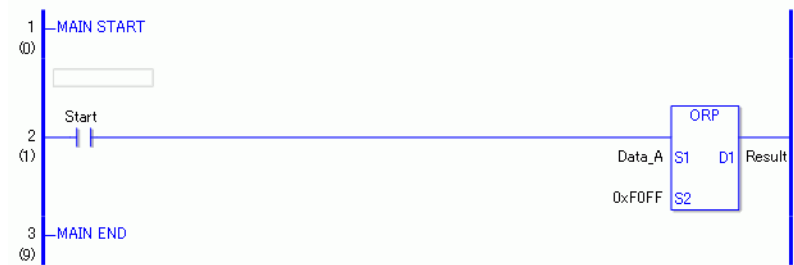


(1)当上升沿指令置 ON 时，将执行 OR 指令。当执行 OR 指令时，将 Data_A 和 F0FF 执行逻辑或的结果保存在 D1 中。

当使用常开指令时，只要指令变量为 ON，就总是执行 OR 指令。

程序示例

ORP

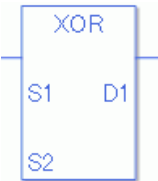


(1)当常开指令置 ON 时，将执行 ORP 指令。当执行 ORP 指令时，将 Data_A 和 F0FF 执行逻辑或的结果保存在 D1 中。

即使使用常开指令，也只有在检测到上升沿时才会执行 ORP 指令。因此，即使常开指令的变量总是为 ON，也只在一次扫描中执行 ORP 指令。

31.13.3 XOR 和 XORP(逻辑异或)

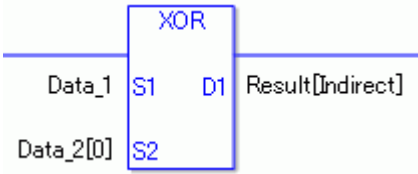
符号和功能

梯形图指令名称	梯形图符号	功能	步数
XOR (逻辑异或 - 电平)		运算	4 至 13
XORP (逻辑异或 - 上升沿)		运算	4 至 13

■ 操作数设置

下面描述了 XOR 和 XORP 指令操作数 (S1、 S2 和 D1) 的可指定内容。
XOR 和 XORP 指令的实际步数取决于操作数的指定方法。下面描述如何计算步数。
操作数 S1 的步数 + 操作数 S2 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 XOR 和 XORP 指令的步数
(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步 } + {Data_2[0] = 2 步 } + {Result[Indirect] = 3 步 } + {1 步 } = 7 步。

指令的总步数需要加最后一步。请务必加一步。

■ 操作数设置

下面描述了 XOR 和 XORP 指令操作数 (S1 和 S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或 指定整型变量 B/W [常量] 指定整型数组 (整个数组)	2	O
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]	4	O
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]	2	O
		D_****.B/W [地址]	3	O
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647	1	O

■ 操作数设置

下面描述了 XOR 和 XORP 指令操作数 (D1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (仅输出)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或 指定整型变量 B/W [常量] 指定整型数组 (整个数组)	2	O
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]	4	O
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]	2	O
		D_****.B/W [地址]	3	O
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647		X

■ XOR 和 XORP 指令说明

XOR 和 XORP 指令是异或指令。当 XOR 指令运行时，它对 S1 和 S2 执行逻辑异或运算并将结果保存在 D1 中。XOR 和 XORP 指令总是处于接通状态。使用 XOR 和 XORP 指令时，如果为操作数 S1、S2 和 D1 指定的变量类型不同，就会发生错误。请为操作数 S1、S2 和 D1 指定相同的变量类型。

指定常量时，请参考如下内容。

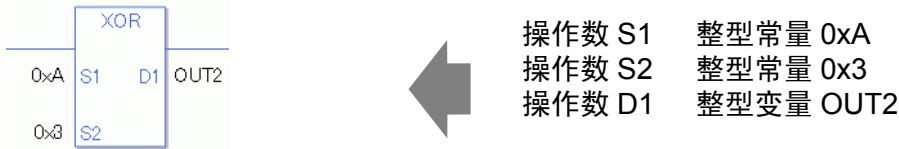
S1	运算符	S2	D1
OFF	XOR	OFF	OFF
ON		OFF	ON
OFF		ON	ON
ON		ON	OFF

当执行 XOR 指令时，仅在 S1 或 S2 其中之一为 ON 时 D1 位才置 ON。否则，D1 位为 OFF。

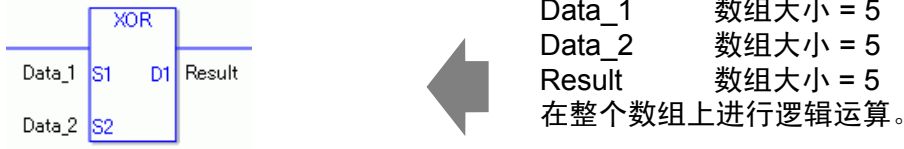
当操作数 D1 是整型变量时



当操作数 D1 是整型变量，而您又想在操作数 S1 和 S2 中输入 16 进制值时。
当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



当计算指定数组 (整型变量数组) 中的数据时
指定整个数组



单个指定数组变量



■ 确认执行结果

(1)当执行结果为 0 时，系统变量 #L_CalcZero 置 ON。

(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

XOR

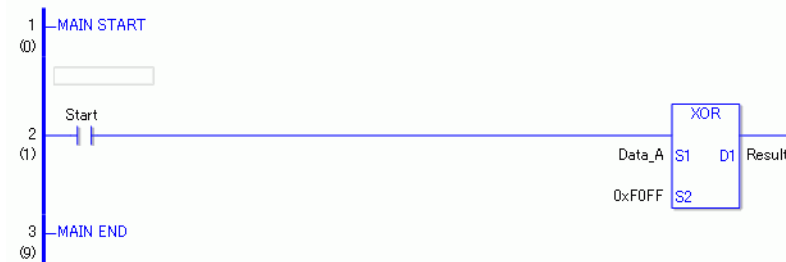


(1)当上升沿指令置 ON 时，将执行 XOR 指令。当执行 XOR 指令时，将 Data_A 和 F0FF 执行 XOR 获得的结果保存在 D1 中。

当使用常开指令时，只要指令变量为 ON，就总是执行 XOR 指令。

程序示例

XORP



(1)当常开指令置 ON 时，将执行 XORP 指令。当执行 XORP 指令时，将 Data_A 和 F0FF 执行逻辑异获得的结果保存在 D1 中。

即使使用常开指令，也只有在检测到上升沿时才会执行 XORP 指令。因此，即使常开指令的变量总是为 ON，也只在一次扫描中执行 XORP 指令。

31.13.4 NOT 和 NOTP(逻辑非)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
NOT (逻辑非 - 电平)		运算	3 至 9
梯形图指令名称	梯形图符号	功能	步数
NOTP (逻辑非 - 上升沿)		运算	3 至 9

■ 操作数设置

下面描述了 NOT 和 NOTP 指令操作数 S1 和 D1 的可指定内容。

NOT 和 NOTP 指令的实际步数取决于操作数的指定方法。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 NOT 和 NOTP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {Result[Indirect] = 3 步} + {1 步} = 5 步。

指令的总步数需要加最后一步。请务必加一步。

■ 操作数设置

下面描述了 NOT 和 NOTP 指令操作数 (S1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或 指定整型变量 B/W [常量] 指定整型数组 (整个数组)	2	O
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]	4	O
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]	2	O
		D_****.B/W [地址]	3	O
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647	1	O

■ 操作数设置

下面描述了 NOT 和 NOTP 指令操作数 (D1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (仅输出)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或 指定整型变量 B/W [常量] 指定整型数组 (整个数组)	2	O
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]	4	O
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]	2	O
		D_****.B/W [地址]	3	O
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647		X

■ NOT 和 NOTP 指令说明

NOT 和 NOTP 指令是逻辑取反指令。当运行 NOT 指令时，对 S1 进行逻辑取反，并将结果保存在 D1 中。NOT 和 NOTP 指令总是处于接通状态。如果为操作数 S1 和 D1 指定的变量类型不同，那么在使用 NOT/NOTP 指令时就会发生错误。请在操作数 S1 和 D1 中指定相同的变量类型。

指定常量时，请参考如下内容。

S1	运算符	D1
OFF	NOT	ON
ON		OFF

当执行 NOT 指令时，如果 S1 为 OFF，D1 位置 ON。如果 S1 位为 ON，DI 位置 OFF。

当操作数 D1 是整型变量时



当操作数 D1 是整型变量，而又想在操作数 S1 中输入 16 进制值时。
当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



当计算指定数组 (整型变量数组) 中的数据时
指定整个数组



单个指定数组变量



■ 确认执行结果

(1)当执行结果为 0 时，系统变量 #L_CalcZero 置 ON。

(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

NOT



(1)当上升沿指令置 ON 时，将执行 NOT 指令。当执行 NOT 指令时，将逻辑取反 Data_A 得到的结果保存在 D1 中。

当使用常开指令时，只要指令变量为 ON，就总是执行 NOT 指令。

程序示例

NOTP



(1)当常开指令置 ON 时，将执行 NOTP 指令。当执行 NOTP 指令时，将逻辑取反 Data_A 得到的结果保存在 D1 中。

即使使用常开指令，也只有在检测到上升沿时才会执行 NOTP 指令。因此，即使常开指令的变量总是为 ON，也只在一次扫描中执行 NOTP 指令。

31.14 传送指令

31.14.1 MOV 和 MOVP(复制)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
MOV (复制 - 电平)		传送	3 至 9
梯形图指令名称	梯形图符号	功能	步数
MOVP (复制 - 上升沿)		传送	3 至 9

■ 操作数设置

下面描述 MOV 和 MOVP 指令操作数 (S1 和 D1) 的可指定内容。
MOV 和 MOVP 指令的实际步数取决于操作数的指定方法。下面描述如何计算步数。
操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 MOV 和 MOVP 指令的步数
(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {Result[Indirect] = 3 步} + {1 步} = 5 步。

指令的总步数需要加最后一步。请务必加一步。

■ 操作数设置

下面描述了 MOV 和 MOVP 指令操作数 (S1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: ○ 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	○
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	○
符号	位			X
	字		1	○
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	○
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]	2	○
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]	3	○
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]	4	○
	浮点型		1	○
		指定浮点型变量 [常量]	2	○
		指定浮点型变量 [变量]	3	○
	实型		1	○
		指定实型变量 [常量]	2	○
		指定实型变量 [变量]	3	○
	定时器	仅 .PT / .ET	2	○
	计数器	仅 .PV / .CV	2	○
	日期	仅 .YR / .MO / .DAY	2	○
	时间	仅 .HR / .MIN / .SEC	2	○
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	○

名称	类型	条件	操作数 步数	可用: O 不可用: X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]	2	O
		D_****.B/W [地址]	3	O
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647	1	O
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38	1	O
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308	2	O

■ 操作数设置

下面描述了 MOV 和 MOVP 指令操作数 (D1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (仅输出)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]	2	O
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]	4	O
	浮点型		1	O
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型		1	O
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：○ 不可用：×
地址格式	X_			×
	Y_			×
	M_			×
	I_			×
	Q_		1	○
	D_	未指定修饰符	1	○
		D_****.B/W [常量]	2	○
		D_****.B/W [地址]	3	○
	F_		1	○
	R_		1	○
	T_	仅 .PT / .ET	2	○
	C_	仅 .PV / .CV	2	○
	N_	仅 .YR / .MO / .DAY	2	○
	J_	仅 .HR / .MIN / .SEC	2	○
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	○
常量				×

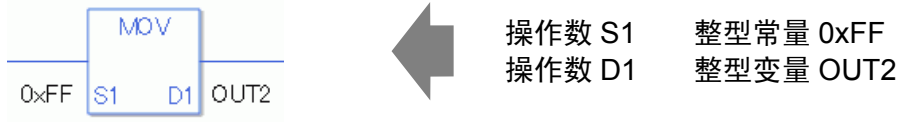
■ MOV 和 MOVP 指令说明

MOV 和 MOVP 指令是传送指令。当执行 MOV 指令时，将 S1 中的值保存到 D1 中。MOV 和 MOVP 指令总是处于接通状态。使用 MOV 和 MOVP 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。请在操作数 S1 和 D1 中指定相同的变量类型。
指定常量时，请参考如下内容。

当操作数 D1 是整型变量时



当操作数 D1 是整型变量，而您又想在操作数 S1 中输入 16 进制值时。
当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



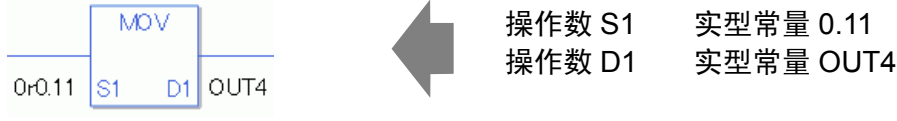
当操作数 D1 是浮点型变量时

当输入 0f(0 和小写字母 “f”) 时，将把后面的值视为浮点型数值。



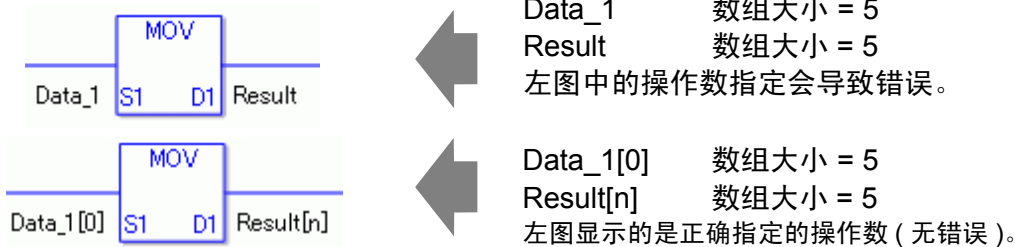
当操作数 D1 是实型变量时

当输入 0r(0 和小写字母 “r”) 时，将把后面的值视为实数值。



当传送指定数组 (整型变量数组) 中的数据时

使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。
如果将 S1 和 D1 指定为整个数组，将会发生错误，即使变量类型相同。



■ 确认执行结果

(1) 如果一个数值不能由操作数 S1 表示 (当执行结果超出范围时), 将不执行该指令。

此时 #L_Error 置 ON 并将错误代码 (6706) 写入 #L_CalcErrCode。

输出结果 D1 维持以前成功执行指令时的值。

(注释)

当使用系统变量检查结果时, 确保在执行完指令后进行。

当在执行完多个指令后检查状态时, 系统变量将只保存最后处理的指令的结果。

程序示例

MOV

将常量保存到整型变量中。



(1) 当上升沿指令置 ON 时, 将执行 MOV 指令。当执行 MOV 指令时, 将常量 10 保存到 D1 中。

当使用常开指令时, 只要指令变量为 ON, 就总是执行 MOV 指令。

程序示例

MOVP

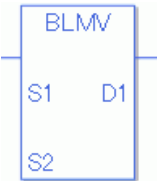
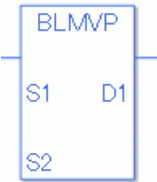


(1) 当常开指令置 ON 时, 将执行 MOVP 指令。当执行 MOVP 指令时, 将常量 10 保存在 D1 中。

即使使用常开指令, 也只有在检测到上升沿时才会执行 MOVP 指令。因此, 即使常开指令的变量总是为 ON, 也只在一次扫描中执行 MOVP 指令。

31.14.2 BLMV 和 BLMVP(块复制)

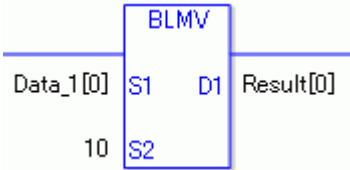
符号和功能

梯形图指令名称	梯形图符号	功能	步数
BLMV (块复制 - 电平)		传送	6 至 10
梯形图指令名称	梯形图符号	功能	步数
BLMVP (块复制 - 上升沿)		传送	6 至 10

■ 操作数设置

下面描述 BLMV 和 BLMVP 指令操作数 (S1、 S2 和 D1) 的可指定内容。
BLMV 和 BLMVP 指令的实际步数取决于操作数的指定方法。下面描述如何计算步数。
操作数 S1 的步数 + 操作数 S2 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 BLMV 和 BLMVP 指令的步数
(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1[0] = 2 步 } + {10 = 1 步 } + {Result[0] = 2 步 } + {1 步 } = 6 步。

指令的总步数需要加最后一步。请务必加一步。

■ 操作数设置

下面描述 BLMV 和 BLMVP 指令操作数 (S1 和 D1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])	2	O
		指定位数组 ([变量])	3	O
	整型 (不包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量]	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型			X
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用: O 不可用: X
地址格式	X_			X
	Y_			X
	M_		1	O
	I_			X
	Q_			X
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量	整型	-2147483648 至 2147483647		X
	浮点型	$\pm 1.175494351\text{e-}38$ 至 $\pm 3.402823466\text{e}+38$		X
	实型	$\pm 2.2250738585072014\text{e-}308$ 至 $\pm 1.7976931348623158\text{e}+308$		X

■ 操作数设置

下面描述 BLMV 和 BLMVP 指令操作数 (S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量]	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

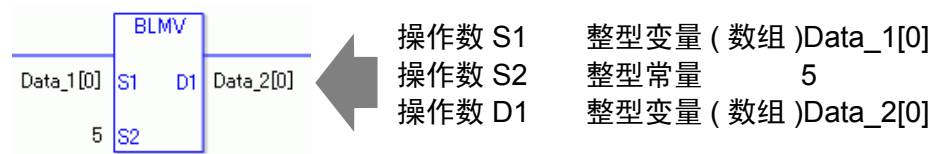
名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量		1 至 4096	1	O

■ BLMV 和 BLMVP 指令说明

BLMV 和 BLMVP 指令是块传送指令。当执行 BLMV 指令时，将 S2 中所示数量的数据元素从 S1 复制到 D1。BLMV 和 BLMVP 指令总是处于接通状态。使用 BLMV 和 BLMVP 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。请为操作数 S1 和 D1 指定相同的变量类型。

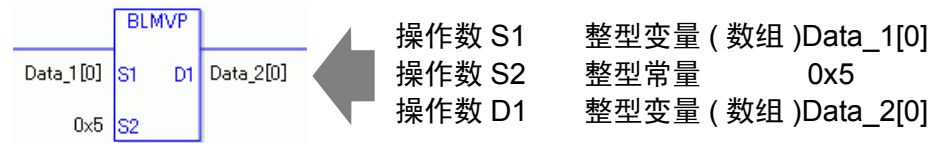
指定常量时，请参考如下内容。

当操作数 D1 是整型变量时



当操作数 D1 是整型变量，而您又想在操作数 S2 中输入 16 进制值时

当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



■ 确认执行结果

(1)如果超出数组范围 (执行结果超出范围), 将不执行指令。此时 #L_Error 置 ON, 并将错误代码写入 #L_CalcErrCode。输出结果 D1 维持上次成功运算的结果。

(注释)

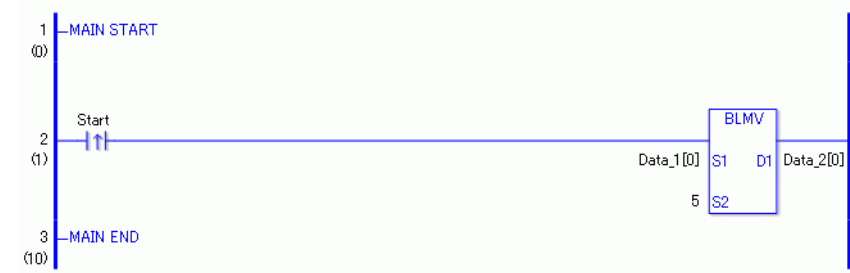
当使用系统变量检查结果时, 确保在执行完指令后进行。

当在执行完多个指令后检查状态时, 系统变量将只保存最后处理的指令的结果。

程序示例

BLMV

把 1 到 5 从 Data_1 复制到 Data_2。



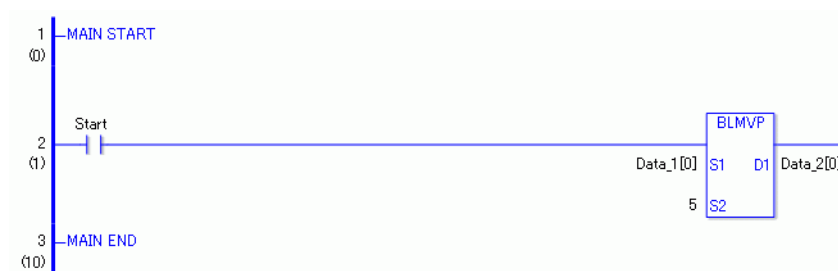
(1)当上升沿指令置 ON 时, 将执行 BLMV 指令。当执行 BLMV 指令时, 将保存在 S1 中数据 1 的 0 至 4 号数据复制到数据 2 的 0 至 4 号中。

当 start 是常开指令时, 只要 start 为 ON, 就总是执行 BLMV 指令。

数组变量名称	数据 1	5 个执行的指令	数据 2
元素	Data_1 [0]	-->	Data_2 [0]
	Data_1 [1]	-->	Data_2 [1]
	Data_1 [2]	-->	Data_2 [2]
	Data_1 [3]	-->	Data_2 [3]
	Data_1 [4]	-->	Data_2 [4]
	Data_1 [5]		Data_2 [5]
	Data_1 [6]		Data_2 [6]
	Data_1 [7]		Data_2 [7]
	Data_1 [8]		Data_2 [8]
	Data_1 [9]		Data_2 [9]
	Data_1 [10]		Data_2 [10]

程序示例

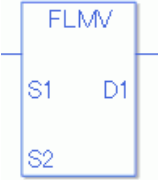
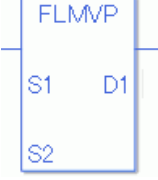
BLMVP



- (1)当常开指令置 ON 时，将执行 BLMVP 指令。当执行 BLMV 指令时，将保存在 S1 中数据 1 的 0 至 4 号数据复制到数据 2 的 0 至 4 号中。
即使使用常开指令，也只有在检测到上升沿时才执行 BLMVP 指令。
因此，即使常开指令的变量总是为 ON，也只在一次扫描中执行 BLMVP 指令。

31.14.3 FLMV 和 FLMVP(多点传送)

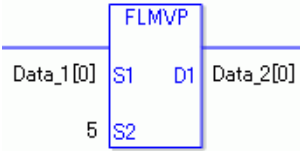
符号和功能

梯形图指令名称	梯形图符号	功能	步数
FLMV (多点传送 - 电平)		传送	4 至 10
梯形图指令名称	梯形图符号	功能	步数
FLMVP (多点传送 - 上升沿)		传送	4 至 10

■ 操作数设置

下面描述 FLMV 和 FLMVP 指令操作数 (S1、 S2 和 D1) 的可指定内容。
FLMV 和 FLMVP 指令的实际步数取决于操作数的指定方法。下面描述如何计算步数。
操作数 S1 的步数 + 操作数 S2 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 FLMV 和 FLMVP 指令的步数
(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1[0] = 2 步 } + {5 = 1 步 } + {Data_2[0] = 2 步 } + {1 步 } = 6 步。

指令的总步数需要加最后一步。请务必加一步。

■ 操作数设置

下面描述 FLMV 和 FLMVP 指令操作数 (S1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量]	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型		1	O
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型		1	O
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647	1	O
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38	1	O
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308	2	O

■ 操作数设置

下面描述 FLMV 和 FLMVP 指令操作数 (S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量]		X
		指定整型变量 [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量		1 至 4096(数组的最大数量)	1	O

■ 操作数设置

下面描述 FLMV 和 FLMVP 指令操作数 (D1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (不包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量]	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型			X
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

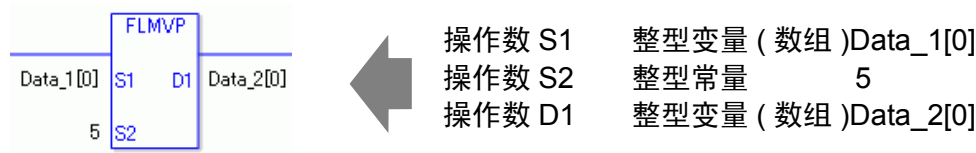
名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ FLMV 和 FLMVP 指令说明

FLMV 和 FLMVP 指令是多点传送指令。当运行 FLMV 指令时，将 S1 中的值复制到从 D1 开始的 S2 个地址中。FLMV 和 FLMVP 指令总是处于接通状态。使用 FLMV 和 FLMVP 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。请为操作数 S1 和 D1 指定相同的变量类型。

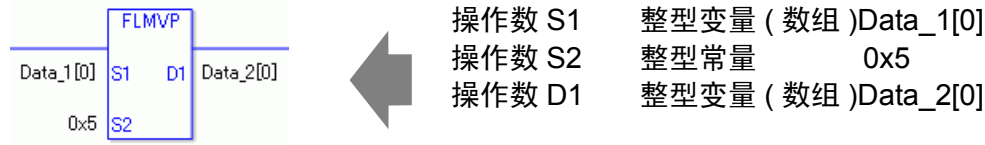
指定常量时，请参考如下内容。

当操作数 D1 是整型变量时



当操作数 D1 是整型变量，而您又想在操作数 S2 中输入 16 进制值时。

当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



■ 确认执行结果

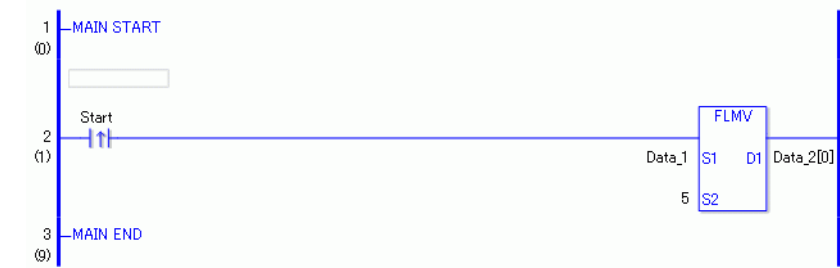
(1)如果超出数组范围 (执行结果超出范围), 将不执行指令。此时 #L_Error 置 ON, 并将错误代码写入 #L_CalcErrCode。输出结果 D1 维持以前成功执行指令时的值。

(注释)
当使用系统变量检查结果时, 确保在执行完指令后进行。
当在执行完多个指令后检查状态时, 系统变量将只保存最后处理的指令的结果。

程序示例

FLMV

将数据 1 中的数据复制到数据 2 的 0 至 4 号元素中。



(1)当上升沿指令置 ON 时, 将执行 FLMV 指令。当执行 FLMV 指令时, 将数据 1 复制到数据 2 的 0 至 4 号元素中。
当使用常开指令时, 只要指令变量为 ON, 就总是执行 FLMV 指令。

数组变量名称	数据 1	5 个执行的指令	数据 2
元素	数据 1	-->	Data_2 [0]
		-->	Data_2 [1]
		-->	Data_2 [2]
		-->	Data_2 [3]
		-->	Data_2 [4]
			Data_2 [5]
			Data_2 [6]
			Data_2 [7]
			Data_2 [8]
			Data_2 [9]
			Data_2 [10]

程序示例

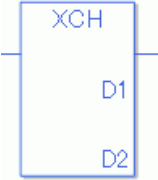
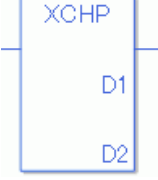
FLMVP



- (1) 当常开指令置 ON 时，将执行 FLMVP 指令。当执行 FLMVP 指令时，将数据 1 复制到数据 2 的 0 至 4 号元素中。
- 即使使用常开指令，也只有在检测到上升沿时才会执行 FLMVP 指令。因此，即使常开指令的变量总是为 ON，也只在一次扫描中执行 FLMVP 指令。

31.14.4 XCH 和 XCHP(交换)

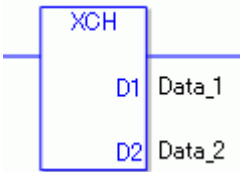
符号和功能

梯形图指令名称	梯形图符号	功能	步数
XCH (交换 - 电平)		传送	3 至 7
XCHP (交换 - 上升沿)		传送	3 至 7

■ 操作数设置

下面描述了 XCH 和 XCHP 指令操作数 (D1 和 D2) 的可指定内容。
XCH 和 XCHP 指令的实际步数取决于操作数的指定方法。下面描述如何计算步数。
操作数 D1 的步数 + 操作数 D2 的步数 + 1 = 一个指令的总步数

例如，计算 XCH 和 XCHP 指令的步数
(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {Data_2=1 步} + {1 步} = 3 步。

指令的总步数需要加最后一步。请务必加一步。

■ 操作数设置

下面描述了 XCH 和 XCHP 指令操作数 (D1 和 D2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括输出)	未指定数组和修饰符	1	O
		指定整型变量 [常量]	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ XCH 和 XCHP 指令说明

XCH 和 XCHP 指令是交换指令。当运行 XCH 指令时，操作数 D1 和 D2 中的数据相互交换。

XCH 和 XCHP 指令总是处于接通状态。使用 XCH 和 XCHP 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。请为操作数 D1 和 D2 指定相同的变量类型。

■ 确认执行结果

(1)如果超出数组范围 (执行结果超出范围), 将不执行指令。此时 #L_Error 置 ON, 并将错误代码写入 #L_CalcErrCode。D1 和 D2 返回为上次成功运行指令时的值。

(注释)

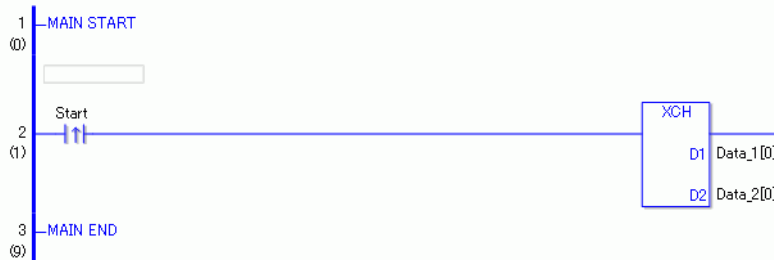
当使用系统变量检查结果时, 确保在执行完指令后进行。

当在执行完多个指令后检查状态时, 系统变量将只保存最后处理的指令的结果。

程序示例

XCH

交换 Data_1 和 Data_2 的内容。



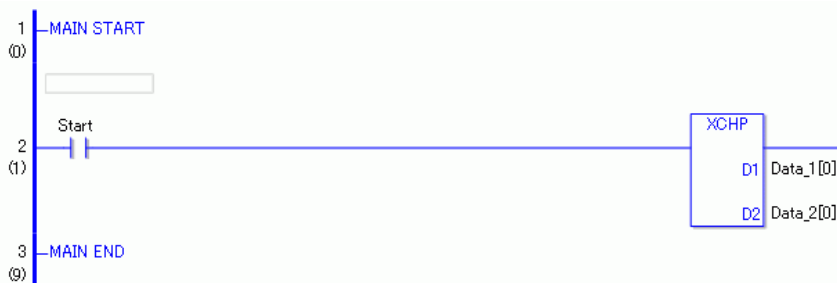
(1)当上升沿指令置 ON 时, 将执行 XCH 指令。当执行 XCH 指令时, 将交换 D1 中的 Data_1[0] 和 D2 中的 Data_2[0]。

当使用常开指令时, 只要指令变量为 ON, 就总是执行 XCH 指令。

数组变量名称	数据 1	指令执行	数据 2
元素	Data_1 [0]	<-->	Data_2 [0]
	Data_1 [1]		Data_2 [1]
	Data_1 [2]		Data_2 [2]
	Data_1 [3]		Data_2 [3]
	Data_1 [4]		Data_2 [4]

程序示例

XCHP



(1)当上升沿指令置 ON 时, 将执行 XCHP 指令。当执行 XCHP 指令时, 将交换 D1 中的 Data_1[0] 和 D2 中的 Data_2[0]。

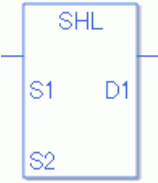
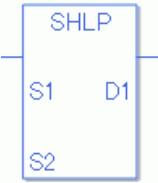
即使使用常开指令, 也只有在检测到上升沿时才执行 XCHP 指令。

因此, 即使常开指令的变量总是为 ON, 也只在一次扫描中执行 XCHP 指令。

31.15 移位指令

31.15.1 SHL 和 SHLP(左移)

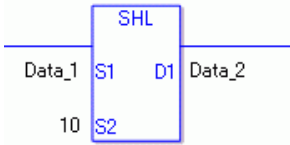
符号和功能

梯形图指令名称	梯形图符号	功能	步数
SHL (左移 - 电平)		移位	4 至 10
梯形图指令名称	梯形图符号	功能	步数
SHLP (左移 - 上升沿)		移位	4 至 10

■ 操作数设置

下表列出了 SHL 和 SHLP 指令操作数 S1、 S2 和 D1 的可指定内容。
SHL 和 SHLP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。
操作数 S1 的步数 + 操作数 S2 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 SHL 和 SHLP 指令的步数
(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {10=1 步} + {Data_2=1 步}+ {1 步} = 4 步。

指令的总步数需要加最后一步。请务必加一步。

■ 操作数设置

下面描述了 SHL 和 SHLP 指令操作数 (S1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或整个数组	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用: O 不可用: X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647	1	O
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ 操作数设置

下面描述了 SHL 和 SHLP 指令操作数 (S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量]	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用: O 不可用: X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	0 至 131071	1	O
	浮点型	$\pm 1.175494351\text{e-}38$ 至 $\pm 3.402823466\text{e}+38$		X
	实型	$\pm 2.2250738585072014\text{e-}308$ 至 $\pm 1.7976931348623158\text{e}+308$		X

■ 操作数设置

下面描述了 SHL 和 SHLP 指令操作数 (D1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: ○ 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	○
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	○
符号	位			X
	字		1	○
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括输出)	未指定数组和修饰符	1	○
		指定整型变量 [常量] 或整个数组	2	○
		指定整型变量 [变量]	3	○
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	○
	计数器	仅 .PV / .CV	2	○
	日期	仅 .YR / .MO / .DAY	2	○
	时间	仅 .HR / .MIN / .SEC	2	○
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	○

名称	类型	条件	操作数 步数	可用：○ 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	○
	Q_		1	○
	D_	未指定修饰符	1	○
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	○
	C_	仅 .PV / .CV	2	○
	N_	仅 .YR / .MO / .DAY	2	○
	J_	仅 .HR / .MIN / .SEC	2	○
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	○
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

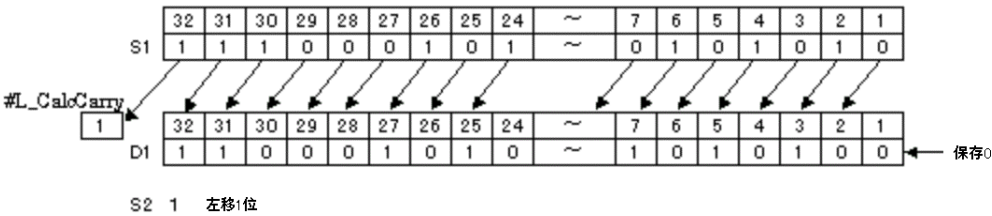
■ SHL 和 SHLP 指令说明

当执行 SHL 或 SHLP 指令时，会将 S1 的各个位左移 S2 个位。每移动一位时，最左侧的位 (最高位) 将丢失。将 0 保存在最底位的空位中。最后将结果保存到 D1 中。SHL 和 SHLP 指令总是处于接通状态。使用 SHL 和 SHLP 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。请为操作数 S1 和 D1 指定相同的变量类型。

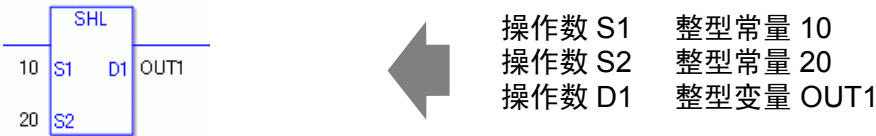
指定常量时，请参考如下内容。

- S1: 移位地址 指定移位地址。
- S2: 移位位数 指定将移动的位数。
- D1: 保存地址 指定保存移位结果的地址。

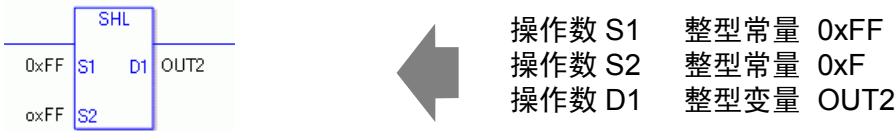
例如，当左移 1 位时



当操作数 D1 是整型变量时



当操作数 D1 是整型变量，而您又想在操作数 S1 和 S2 中输入 16 进制值时。
当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



在对指定数组 (整型变量数组) 中的数据进行移位和指定数组元素时, 请使用相同的格式。
如果格式不同就会发生错误。



如果 S1 和 D1 数组大小相同, 那么将把 S1 作为一个大的整数看待。位将从一个元素移到下一个元素。
每个元素的最高位不会丢失。但是, 最后一个元素的最高位会丢失。请为 S2 指定一个 0 至 (32 x 数组大小 -1) 中的值。



如果 S1 和 D1 均不是数组, 该指令将移动 S1 中的 32 位。为 S2 指定一个 0 至 31 间的值。

■ 显示执行结果的系统变量

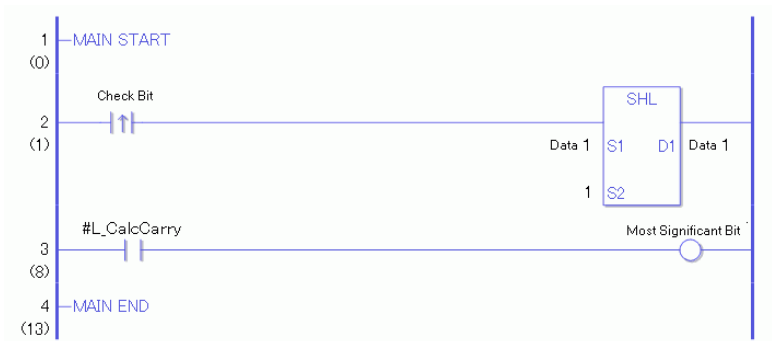
当执行结果是 0 时，#L_CalcZero 置 ON。
如果由于移位操作造成溢出，将在 #L_CalcCarry 中保存最后一个溢出的位。
当执行发生错误时，将在 #L_Status 中保存错误信息。
当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

(注释)
当使用系统变量检查结果时，确保在执行完指令后进行。
当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

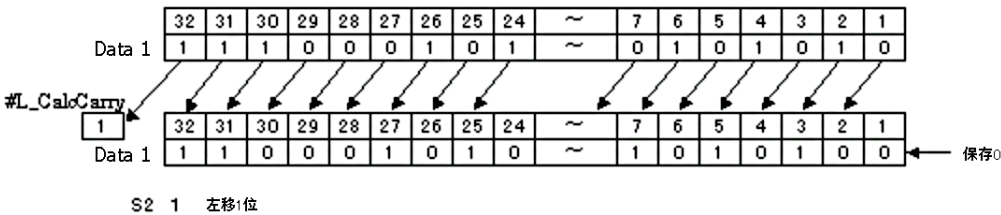
SHL

确定最高位是 ON 还是 OFF。



- (1)当上升沿指令置 ON 时，将执行 SHL 指令。当执行 SHL 指令时，将左移 1 位得到的结果保存在 D1 中。
- (2)左移 1 位时，您可以通过 #L_CalcCarry 的状态检查数据移位前最高位是 ON 还是 OFF。

(注意) 当使用常开指令时，只要常开指令为 ON，就总是执行 SHL 指令。



程序示例

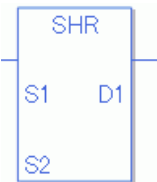
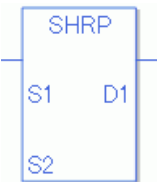
SHLP



SHLP 和 SHL 指令检测执行时间的方法不同。对于 SHLP 指令，即使使用常开指令，也只有在检测到上升沿时才执行 SHLP 指令。因此，即使常开指令保持为 ON，也只在一次扫描中执行 SHLP 指令。

31.15.2 SHR 和 SHRP(右移)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
SHR (右移 - 电平)		移位	4 至 10
SHRP (右移 - 上升沿)		移位	4 至 10

■ 操作数设置

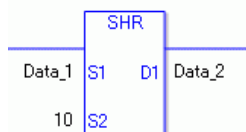
下面描述了 SHR 和 SHRP 指令操作数 S1、S2 和 D1 的可指定内容。

SHR 和 SHRP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 S2 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 SHR 和 SHRP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {10=1 步} + {Data_2=1 步} + {1 步} = 4 步。

指令的总步数需要加最后一步。请务必加一步。

■ 操作数设置

下面描述了 SHR 和 SHRP 指令操作数 (S1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或整个数组	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647	1	O
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ 操作数设置

下面描述了 SHR 和 SHRP 指令操作数 (S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量]	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	0 至 131071	1	O
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ 操作数设置

下面描述了 SHR 和 SHRP 指令操作数 (D1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: ○ 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	○
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	○
符号	位			X
	字		1	○
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括输出)	未指定数组和修饰符	1	○
		指定整型变量 [常量] 或整个数组	2	○
		指定整型变量 [变量]	3	○
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	○
	计数器	仅 .PV / .CV	2	○
	日期	仅 .YR / .MO / .DAY	2	○
	时间	仅 .HR / .MIN / .SEC	2	○
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	○

名称	类型	条件	操作数 步数	可用：○ 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	○
	Q_		1	○
	D_	未指定修饰符	1	○
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	○
	C_	仅 .PV / .CV	2	○
	N_	仅 .YR / .MO / .DAY	2	○
	J_	仅 .HR / .MIN / .SEC	2	○
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	○
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

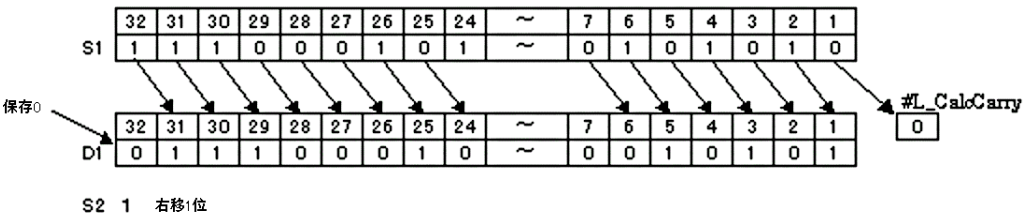
■ SHL 和 SHLP 指令说明

当执行 SHR 或 SHRP 指令时，会将 S1 的各个位右移 S2 个位。每移动一位时，最右侧的位 (最低位) 将丢失。在最高位的空位中保存 0。最后将结果保存到 D1 中。SHR 和 SHRP 指令总是处于接通状态。使用 SHR 和 SHRP 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。请为操作数 S1 和 D1 指定相同的变量类型。

指定常量时，请参考如下内容。

- S1: 移位地址 指定移位地址。
- S2: 移位位数 指定将移动的位数。
- D1: 保存地址 指定保存移位结果的地址。

例如，当右移 1 位时

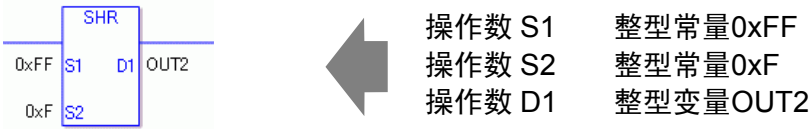


当操作数 D1 是整型变量时



当操作数 D1 是整型变量，而您又想在操作数 S1 和 S2 中输入 16 进制值时。

当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



在对指定数组 (整型变量数组) 中的数据进行移位和指定数组元素时, 请使用相同的格式。
如果格式不同就会发生错误。



如果 S1 和 D1 数组大小相同, 那么将把 S1 作为一个大的整数看待。位将从一个元素移到下一个元素。
每个元素的最底位不会丢失。但是, 第一个元素的最低位会丢失。指定 S2 为 0 或以上, 最大 (32 x 数组大小 -1)。



如果 S1 和 D1 均不是数组, 将移动 32 个位。为 S2 指定一个 0 至 31 间的值。

■ 显示执行结果的系统变量

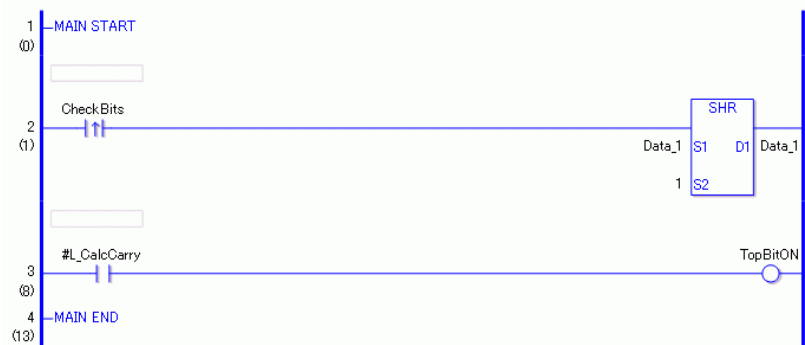
当执行结果是 0 时，#L_CalcZero 置 ON。
如果由于移位操作造成溢出，将在 #L_CalcCarry 中保存最后一个溢出的位。
当执行发生错误时，将在 #L_Status 中保存错误信息。
当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

(注释)
当使用系统变量检查结果时，确保在执行完指令后进行。
当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

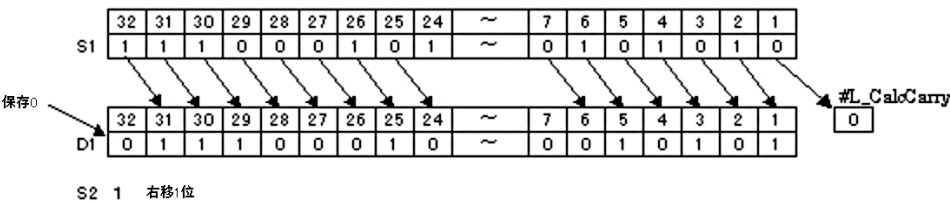
程序示例

SHR

确定最低位是 ON 还是 OFF。

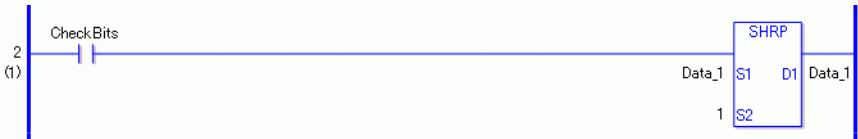


- (1)当上升沿指令的变量置 ON 时，将运行 SHR 指令。当执行 SAR 指令时，会将右移 1 位的结果保存在 D1 中。
 - (2)在移位操作完成后，通过使用 #L_CalcCarry 系统变量，您可以检查 Data_1 最低位的先前值。
- (补充) 当使用常开指令时，只要该常开位为 ON，就总是执行 SAL 指令。



程序示例

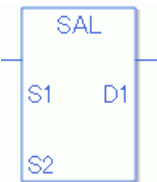
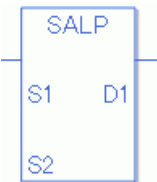
SHRP



在何时运行指令方面，SHRP 和 SHR 指令有所不同。对于 SHRP 指令，即使使用常开指令，也只有在检测到该位的上升沿时才执行 SHRP 指令。即便常开指令位保持为 ON，也只在一次扫描中执行 SHRP 指令。

31.15.3 SAL 和 SALP(算术左移)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
SAL (算术左移 - 电平)		移位	4 至 10
梯形图指令名称	梯形图符号	功能	步数
SALP (算术左移 - 上升沿)		移位	4 至 10

■ 操作数设置

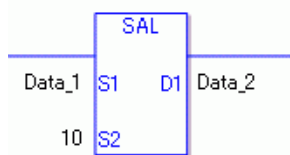
下表列出了 SAL 和 SALP 指令操作数 (S1、S2 和 D1) 的可指定内容。

SAL 和 SALP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 S2 的步数 + 操作数 D1 的步数 +1 = 一个指令的总步数

例如，计算 SAL 和 SALP 指令中的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {10=1 步} + {Data_2=1 步} + {1 步} = 4 步。

指令的总步数需要加最后一步。请务必加一步。

■ 操作数设置

下面描述了 SAL 和 SALP 指令操作数 (S1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量]	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647	1	O
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ 操作数设置

下面描述了 SAL 和 SALP 指令操作数 (S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量]	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	0 至 31	1	O
	浮点型	$\pm 1.175494351\text{e-}38$ 至 $\pm 3.402823466\text{e}+38$		X
	实型	$\pm 2.2250738585072014\text{e-}308$ 至 $\pm 1.7976931348623158\text{e}+308$		X

■ 操作数设置

下面描述了 SAL 和 SALP 指令操作数 (D1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括输出)	未指定数组和修饰符	1	O
		指定整型变量 [常量]	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ SAL 和 SALP 指令说明

当执行 SAL 或 SALP 指令时，将 S1 数据的各个位左移 S2 个位。每移动 1 位时，第 30 个位将丢失。将 0 保存在最低位的空位中。最后将结果保存到 D1 中。
SAL 和 SALP 指令总是处于接通状态。使用 SAL 和 SALP 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。请为操作数 S1 和 D1 指定相同的变量类型。
指定常量时，请参考如下内容。

- S1: 移位地址

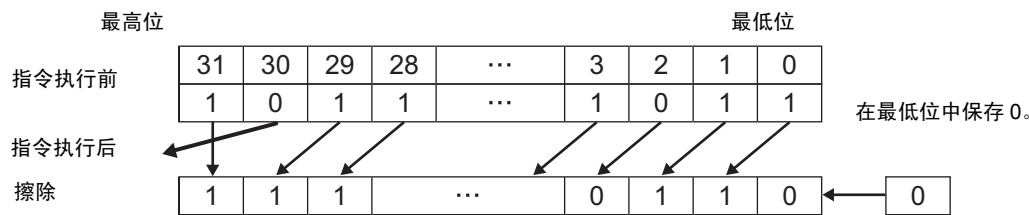
S2: 移位位数

D1: 保存地址
- 指定移位地址。

指定将移动的位数。

指定保存移位结果的地址。

例如，当左移 1 位时



当操作数 D1 是整型变量时



当操作数 D1 是整型变量，而您又想在操作数 S1 和 S2 中输入 16 进制值时。
当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



当指定数组变量时，指定数组元素。



31 个数组元素位被移位。为 S2 指定一个 0 至 31 间的值。

■ 显示执行结果的系统变量

当执行结果是 0 时，#L_CalcZero 置 ON。

当执行发生错误时，将在 #L_Status 中保存错误信息。

当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

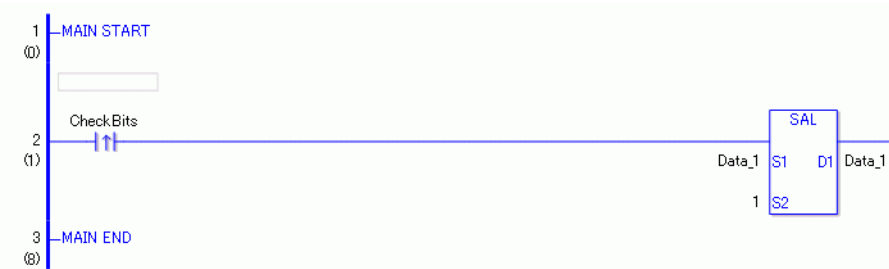
(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

SAL

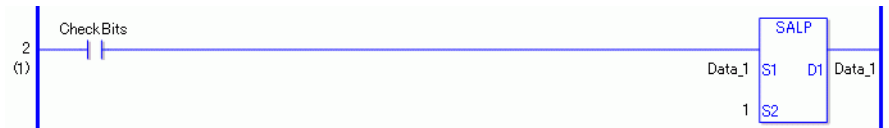


(1)当上升沿指令置 ON 时，将执行 SAL 指令。当执行 SAL 指令时，会将移位结果保存到 D1 中。最高位不发生移位，在最低位中保存 0。

(补充) 当使用常开指令时，只要该常开位为 ON，就总是执行 SAL 指令。

程序示例

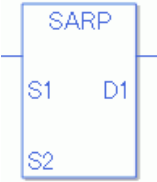
SALP



SALP 和 SAL 指令检测执行时间的方法不同。对于 SALP 指令，即使使用常开指令，也只有当检测到上升沿时才执行 SALP 指令。因此，即使常开指令保持为 ON，也只在一次扫描中执行 SALP 指令。

31.15.4 SAR 和 SARP(算术右移)

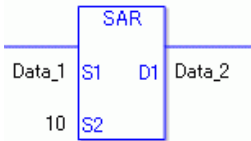
符号和功能

梯形图指令名称	梯形图符号	功能	步数
SAR (算术右移 - 电平)		移位	4 至 10
SARP (算术右移 - 上升沿)		移位	4 至 10

■ 操作数设置

下表列出了 SAR 和 SARP 指令操作数 (S1、 S2 和 D1) 的可指定内容。
SAR 和 SARP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。
操作数 S1 的步数 + 操作数 S2 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 SAR 和 SARP 指令的步数
(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {10=1 步} + {Data_2=1 步} + {1 步} = 4 步。

指令的总步数需要加最后一步。请务必加一步。

■ 操作数设置

下面描述了 SAR 和 SARP 指令操作数 (S1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量]	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：○ 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	○
	Q_		1	○
	D_	未指定修饰符	1	○
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	○
	C_	仅 .PV / .CV	2	○
	N_	仅 .YR / .MO / .DAY	2	○
	J_	仅 .HR / .MIN / .SEC	2	○
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	○
常量	整型	-2147483648 至 2147483647	1	○
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ 操作数设置

下面描述了 SAR 和 SARP 指令操作数 (S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: ○ 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	○
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	○
符号	位			X
	字		1	○
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	○
		指定整型变量 [常量]	2	○
		指定整型变量 [变量]	3	○
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	○
	计数器	仅 .PV / .CV	2	○
	日期	仅 .YR / .MO / .DAY	2	○
	时间	仅 .HR / .MIN / .SEC	2	○
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	○

名称	类型	条件	操作数 步数	可用：○ 不可用：×
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	○
	Q_		1	○
	D_	未指定修饰符	1	○
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	○
	C_	仅 .PV / .CV	2	○
	N_	仅 .YR / .MO / .DAY	2	○
	J_	仅 .HR / .MIN / .SEC	2	○
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	○
常量	整型	0 至 31	1	○
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ 操作数设置

下面描述了 SAR 和 SARP 指令操作数 (D1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括输出)	未指定数组和修饰符	1	O
		指定整型变量 [常量]	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

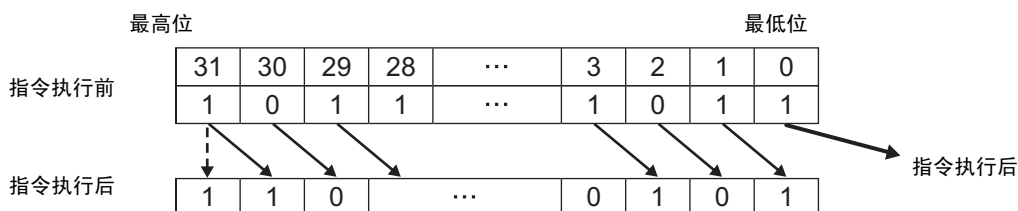
名称	类型	条件	操作数 步数	可用: O 不可用: X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647		X
	浮点型	$\pm 1.175494351\text{e-}38$ 至 $\pm 3.402823466\text{e}+38$		X
	实型	$\pm 2.2250738585072014\text{e-}308$ 至 $\pm 1.7976931348623158\text{e}+308$		X

■ SAR 和 SARP 指令说明

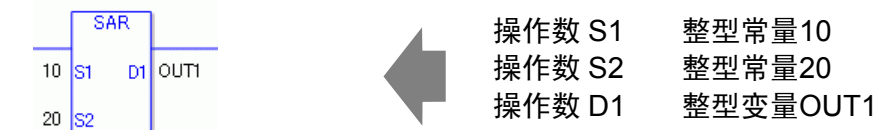
当执行 SAR 或 SARP 指令时，会将 S1 的各个位右移 S2 个位。每次移位时，最低位将丢失，最高位则被保存在最左边的空位中。最后将结果保存到 D1 中。SAR 和 SARP 指令总是处于接通状态。使用 SAR 和 SARP 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。请为操作数 S1 和 D1 指定相同的变量类型。指定常量时，请参考如下内容。

S1: 移位地址	指定移位地址。
S2: 移位位数	指定将移动的位数。
D1: 保存地址	指定保存移位结果的地址。

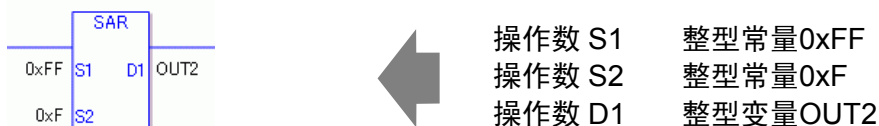
例如，当右移 1 位时



当操作数 D1 是整型变量时



当操作数 D1 是整型变量，而您又想在操作数 S1 和 S2 中输入 16 进制值时。
当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



当指定数组变量时，指定数组元素。



31 个数组元素位被移位。为 S2 指定一个 0 至 31 间的值。

■ 显示执行结果的系统变量

当执行结果是 0 时，#L_CalcZero 置 ON。

当执行发生错误时，将在 #L_Status 中保存错误信息。

当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

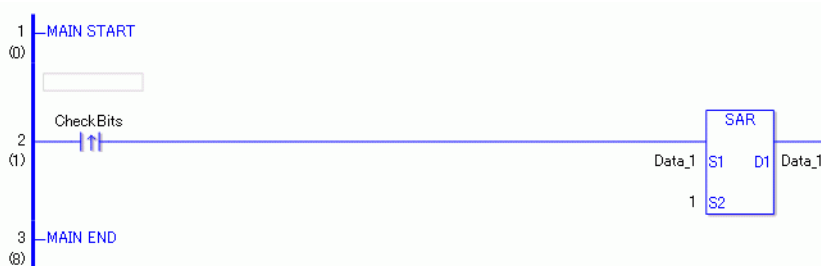
(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

SAR

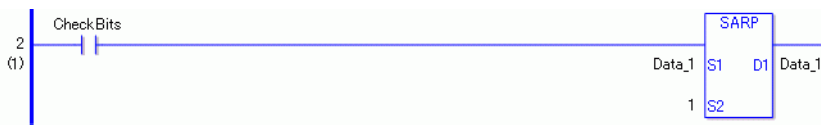


(1)当上升沿指令置 ON 时，将执行 SAR 指令。当执行 SAR 指令时，会将右移 1 位的结果保存在 D1 中。最高位不发生移位，但也被复制到 D1 中。每次移位时，都将最高位复制到最左边的空位中。

(补充) 当使用常开指令时，只要该位为 ON，就总是执行 SAR 指令。

程序示例

SALP

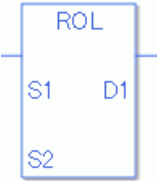
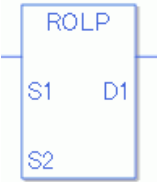


SARP 和 SAR 指令检测执行时间的方法不同。对于 SARP 指令，即使使用常开指令，也只有当检测到上升沿时才执行 SARP 指令。因此，即便常开指令位保持为 ON，也只在一次扫描中执行 SARP 指令。

31.16 循环移位指令

31.16.1 ROL 和 ROLP(循环左移)

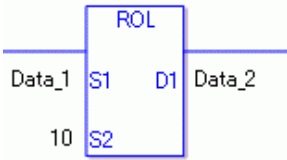
符号和功能

梯形图指令名称	梯形图符号	功能	步数
ROL (循环左移 - 电平)		循环移位	4 至 10
梯形图指令名称	梯形图符号	功能	步数
ROLP (循环左移 - 上升沿)		循环移位	4 至 10

■ 操作数设置

下表列出了 ROL 和 ROLP 指令操作数 (S1、 S2 和 D1) 的可指定内容。
ROL 和 ROLP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。
操作数 S1 的步数 + 操作数 S2 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 ROL 和 ROLP 指令的步数
(有关操作数步数的信息， 请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {10=1 步} + {Data_2=1 步}+ {1 步} = 4 步。

指令的总步数需要加最后一步。请务必加一步。

■ 操作数设置

下面描述了 ROL 和 ROLP 指令操作数 (S1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或整个数组	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647	1	O
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ 操作数设置

下面描述了 ROL 和 ROLP 指令操作数 (S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: ○ 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	○
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	○
符号	位			X
	字		1	○
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	○
		指定整型变量 [常量]	2	○
		指定整型变量 [变量]	3	○
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	○
	计数器	仅 .PV / .CV	2	○
	日期	仅 .YR / .MO / .DAY	2	○
	时间	仅 .HR / .MIN / .SEC	2	○
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	○

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	0 至 131071	1	O
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ 操作数设置

下面描述了 ROL 和 ROLP 指令操作数 (D1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括输出)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或整个数组	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ ROL 和 ROLP 指令说明

当执行 ROL 或 ROLP 指令时，会将 S1 的各个位循环左移 S2 个位。每次循环移动 1 位时，会将最左侧的位（最高位）循环移位到最右侧的位（最低位）。最后将结果保存到 D1 中。ROL 和 ROLP 指令总是处于接通状态。使用 ROL 和 ROLP 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。请为操作数 S1 和 D1 指定相同的变量类型。

指定常量时，请参考如下内容。

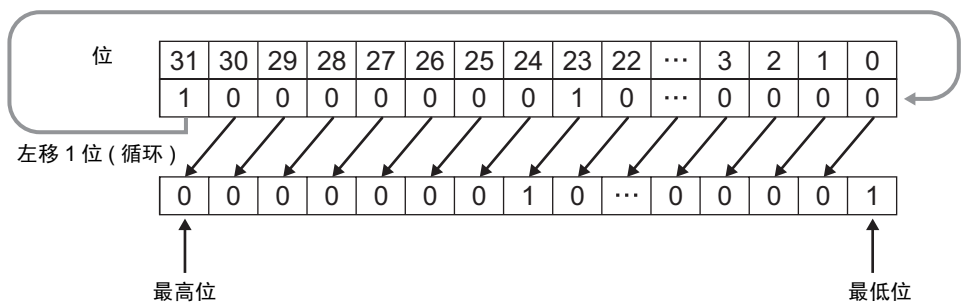
- S1: 循环移位地址

指定要执行循环移位的地址。
- S2: 循环移位位数

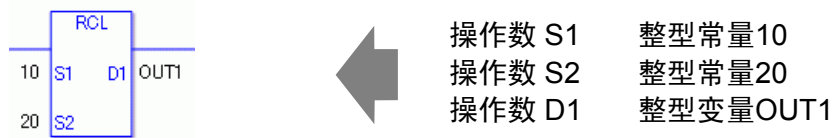
指定将循环移位的位数。
- D1: 存储地址

指定一个保存循环移位结果的地址。

例如，当循环左移 1 位时



当操作数 D1 是整型变量时

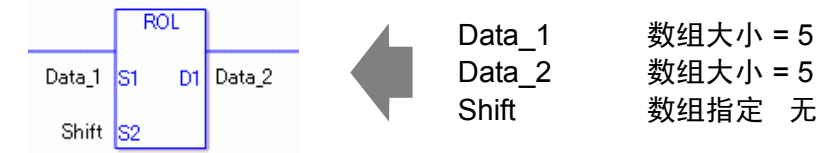


当操作数 D1 是整型变量，而您又想在操作数 S1 和 S2 中输入 16 进制值时。

当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



在循环左移指定数组 (整型变量数组) 中的数据 and 指定数组元素时, 请使用相同的格式。
如果格式不同就会发生错误。



如果 S1 和 D1 数组大小相同, 那么将把 S1 作为一个大的整数看待。将从一个元素循环移位到下一个元素。
将从一个元素循环移位到下一个元素。即对整个数组进行循环移位, 而不是每个元素之内的位。请为 S2 指定一个 0 至 (32 x 数组大小 -1) 中的值。



如果 S1 和 D1 均不是数组, 将只移动 32 个位。为 S2 指定一个 0 至 31 间的值。

■ 显示执行结果的系统变量

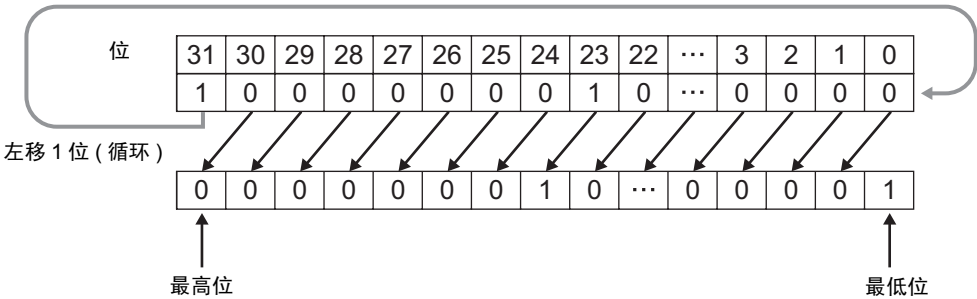
当执行结果是 0 时，#L_CalcZero 置 ON。
如果循环移位造成溢出，将在 #L_CalcCarry 中保存溢出的位。
当执行发生错误时，将在 #L_Status 中保存错误信息。
当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

(注释)
当使用系统变量检查结果时，确保在执行完指令后进行。
当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

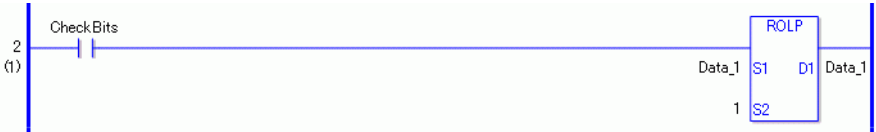
程序示例
ROL



(1)当上升沿指令置 ON 时，将执行 ROL 指令。当执行 ROL 指令时，会将循环移位 1 位的结果保存在 D1 中。
(补充) 当使用常开指令时，只要该位为 ON，就总是执行 ROL 指令。



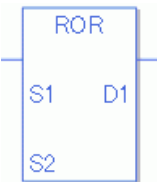
程序示例
ROLP



ROLP 和 ROL 指令检测执行时间的方法不同。对于 ROLP 指令，即使使用常开指令，也只有在检测到该位的上升沿时才执行 ROLP 指令。因此，即便常开指令位保持为 ON，也只在一次扫描中执行 ROLP 指令。

31.16.2 ROR 和 RORP(循环右移)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
ROR (循环右移 - 电平)		循环移位	4 至 10
RORP (循环右移 - 上升沿)		循环移位	4 至 10

■ 操作数设置

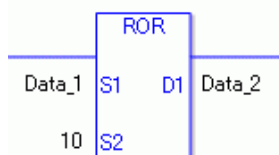
下表列出了 ROR 和 RORP 指令操作数 S1、S2 和 D1 的可指定内容。

ROR 和 RORP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 S2 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 ROR 和 RORP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {10=1 步} + {Data_2=1 步} + {1 步} = 4 步。

指令的总步数需要加最后一步。请务必加一步。

■ 操作数设置

下面描述了 ROR 和 RORP 指令操作数 (S1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或整个数组	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647	1	O
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ 操作数设置

下面描述了 ROR 和 RORP 指令操作数 (S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量]	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	0 至 131071	1	O
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ 操作数设置

下面描述了 ROR 和 RORP 指令操作数 (D1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括输出)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或整个数组	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ ROR 和 RORP 指令说明

当执行 ROR 或 RORP 指令时，会将 S1 的各个位循环右移 S2 个位。会将最右侧的位（最低位）的信息保存在最左侧的空位中。

最后将结果保存到 D1 中。ROR 和 RORP 指令总是处于接通状态。使用 ROR 和 RORP 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。请为操作数 S1 和 D1 指定相同的变量类型。

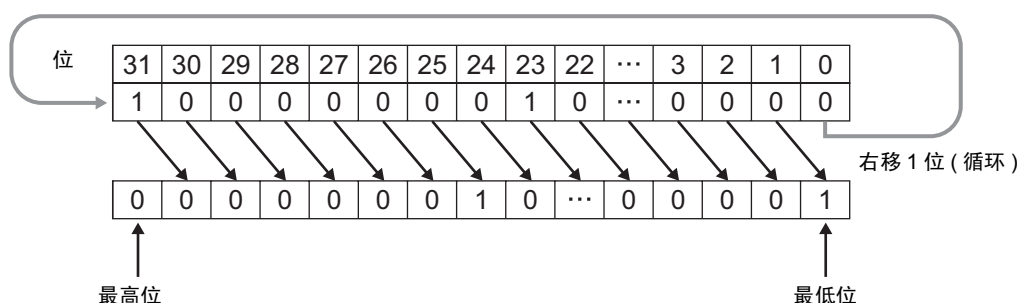
指定常量时，请参考如下内容。

S1: 循环移位地址 指定要执行循环移位的地址。

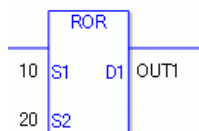
S2: 循环移位位数 指定将循环移位的位数。

D1: 存储地址 指定一个保存循环移位结果的地址。

例如，当循环右移 1 位时



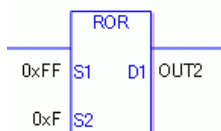
当操作数 D1 是整型变量时



操作数 S1	整型常量10
操作数 S2	整型常量20
操作数 D1	整型变量OUT1

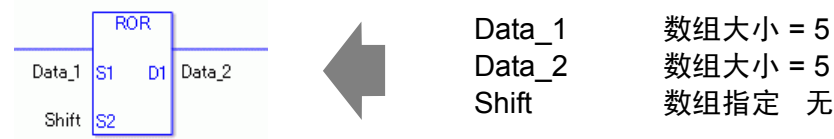
当操作数 D1 是整型变量，而您又想在操作数 S1 和 S2 中输入 16 进制值时。

当输入 0x(0 和小写字母 “x”) 时, 将把后面的值视为十六进制值。

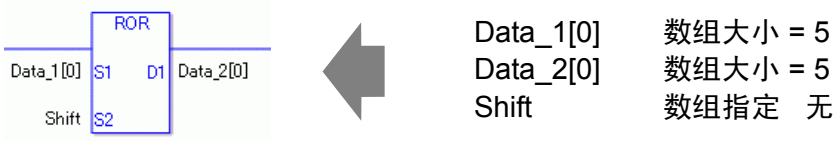


操作数 S1	整型常量0xFF
操作数 S2	整型常量0xF
操作数 D1	整型变量OUT2

在循环左移指定数组 (整型变量数组) 中的数据和指定数组元素时, 请使用相同的格式。
如果格式不同就会发生错误。



如果 S1 和 D1 数组大小相同, 那么将把 S1 作为一个大的整数看待。将从一个元素循环移位到下一个元素。
将从一个元素循环移位到下一个元素。即对整个数组进行循环移位, 而不是每个元素之内的位。请为 S2 指定一个 0 至 (32 x 数组大小 -1) 中的值。



如果 S1 和 D1 均不是数组, 将只移动 32 个位。为 S2 指定一个 0 至 31 间的值。

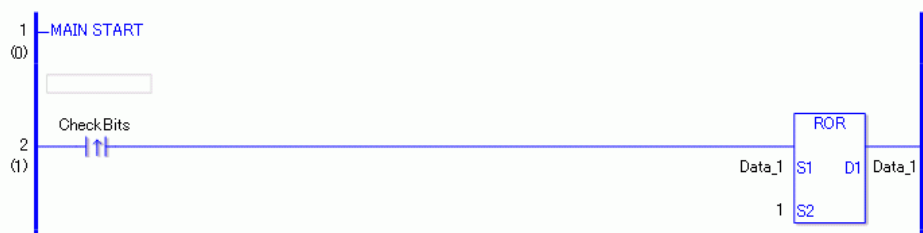
■ 显示执行结果的系统变量

当执行结果是 0 时，#L_CalcZero 置 ON。
如果循环移位造成溢出，将在 #L_CalcCarry 中保存溢出的位。
当执行发生错误时，将在 #L_Status 中保存错误信息。
当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

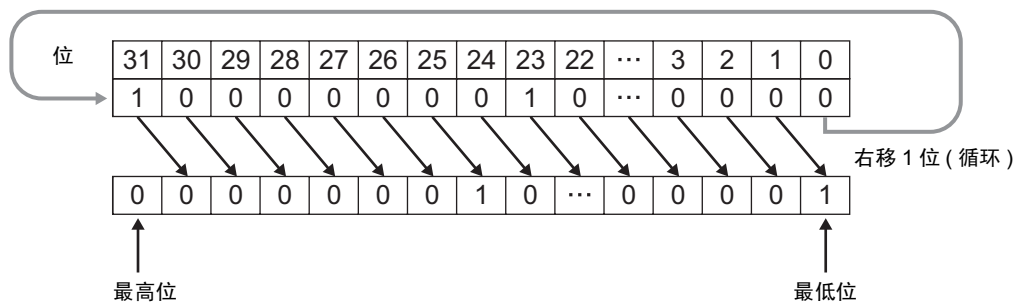
(注释)
当使用系统变量检查结果时，确保在执行完指令后进行。
当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

ROR

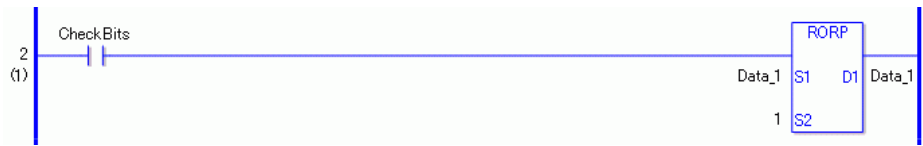


(1)当上升沿指令置 ON 时，将执行 ROR 指令。当执行 ROR 指令时，会将循环右移 1 位的结果保存在 D1 中。
(补充) 当使用常开指令时，只要该位为 ON，就总是执行 ROR 指令。



程序示例

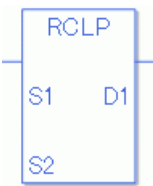
RORP



RORP 和 ROR 指令检测执行时间的方法不同。对于 RORP 指令，即使使用常开指令，也只有在检测到上升沿时才执行 RORP 指令。因此，即使常开指令位保持为 ON，也只在一次扫描中执行 RORP 指令。

31.16.3 RCL 和 RCLP(带进位循环左移)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
RCL (带进位循环左移 - 电平)		循环移位	4 至 10
梯形图指令名称	梯形图符号	功能	步数
RCLP (带进位循环左移 - 上升沿)		循环移位	4 至 10

■ 操作数设置

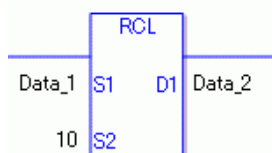
下表列出了 RCL 和 RCLP 指令操作数 S1、S2 和 D1 的可指定内容。

RCL 和 RCLP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 S2 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 RCL 和 RCLP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {10=1 步} + {Data_2=1 步} + {1 步} = 4 步。

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面描述了 RCL 和 RCLP 指令操作数 (S1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量]	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647	1	O
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ 操作数设置

下面描述了 RCL 和 RCLP 指令操作数 (S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: ○ 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	○
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	○
符号	位			X
	字		1	○
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	○
		指定整型变量 [常量]	2	○
		指定整型变量 [变量]	3	○
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	○
	计数器	仅 .PV / .CV	2	○
	日期	仅 .YR / .MO / .DAY	2	○
	时间	仅 .HR / .MIN / .SEC	2	○
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	○

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	0 至 32	1	O
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ 操作数设置

下面描述了 RCL 和 RCLP 指令操作数 (D1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: ○ 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	○
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	○
符号	位			X
	字		1	○
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括输出)	未指定数组和修饰符	1	○
		指定整型变量 [常量]	2	○
		指定整型变量 [变量]	3	○
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	○
	计数器	仅 .PV / .CV	2	○
	日期	仅 .YR / .MO / .DAY	2	○
	时间	仅 .HR / .MIN / .SEC	2	○
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	○

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ RAL 和 RCLP 指令说明

当执行 RCL 或 RCLP 指令时，会将 S1 的各个位循环左移 S2 个位。将最左侧的位（最高位）保存到进位标识中，而将进位标识（1 或 0）循环移位到最右侧的位（最低位）。最后将结果保存到 D1 中。RCL 和 RCLP 指令总是处于接通状态。使用 RCL 和 RCLP 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。请为操作数 S1 和 D1 指定相同的变量类型。

指定常量时，请参考如下内容。

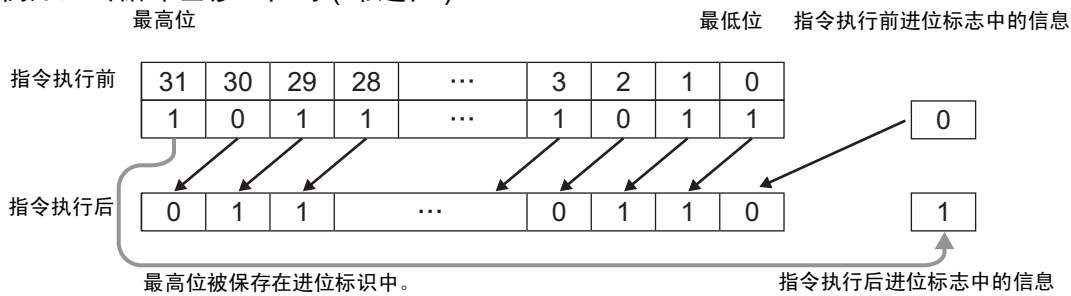
- S1: 循环移位地址

指定要执行循环移位的地址。
- S2: 循环移位位数

指定将循环移位的位数。
- D1: 存储地址

指定一个保存循环移位结果的地址。

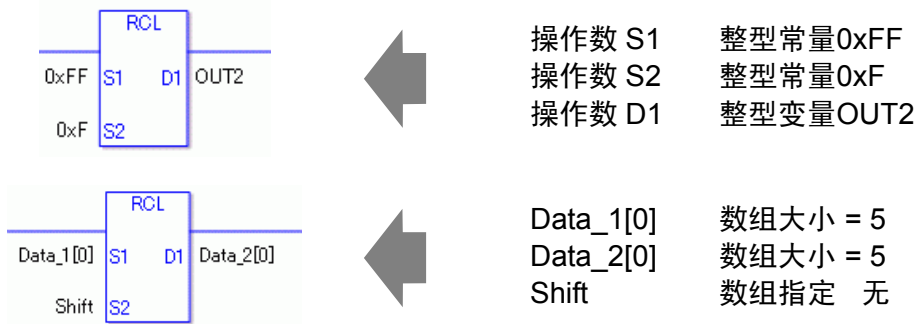
例如，当循环左移 1 位时（带进位）



当操作数 D1 是整型变量时



当操作数 D1 是整型变量，而您又想在操作数 S1 和 S2 中输入 16 进制值时。
当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



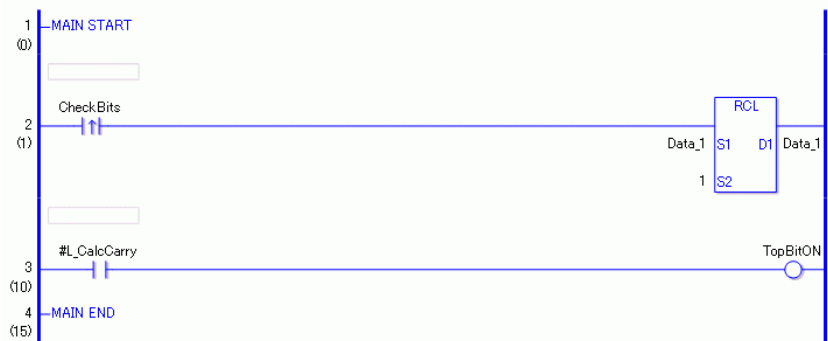
如果 S1 和 D1 均不是数组，将带进位循环移动 32 个位。
为 S2 指定一个 0 至 32 间的值。

■ 显示执行结果的系统变量

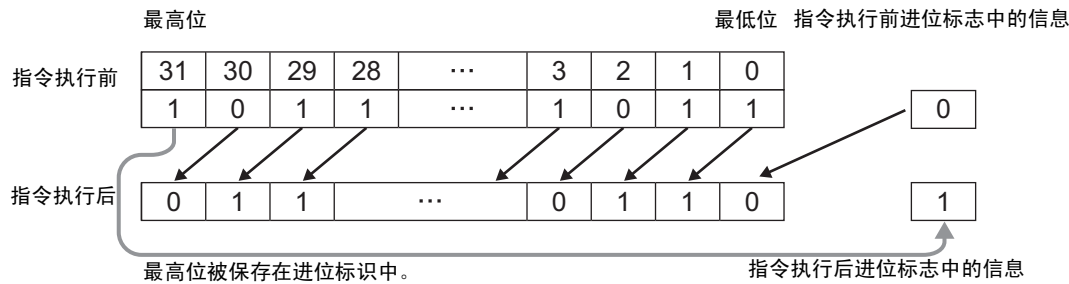
当执行结果是 0 时，#L_CalcZero 置 ON。
如果循环移位造成溢出，将在 #L_CalcCarry 中保存溢出的位。
当执行发生错误时，将在 #L_Status 中保存错误信息。
当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

(注释)
当使用系统变量检查结果时，确保在执行完指令后进行。
当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

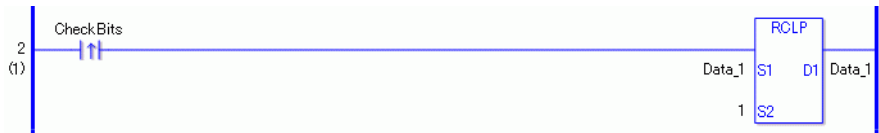
程序示例
RCL



- (1)当上升沿指令置 ON 时，将执行 RCL 指令。当执行 RCL 指令时，会将带进位循环移位 1 位的结果保存在 D1 中。
- (2)当带进位左移 1 位时，您可以用 #L_CalcCarry 来检查循环移位操作前最高位的值。
- (补充) 当使用常开指令时，只要该位为 ON，就总是执行 RCL 指令。



程序示例
RCLP



RCLP 和 RCL 指令检测执行时间的方法不同。对于 RCLP 指令，即使使用常开指令时，也只有在检测到上升沿时才会执行 RCLP 指令。因此，即使常开指令位保持为 ON，也只在一次扫描中执行 RCLP 指令。

31.16.4 RCR 和 RCRP(带进位循环右移)

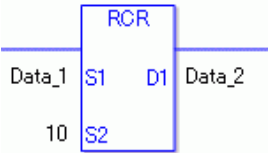
符号和功能

梯形图指令名称	梯形图符号	功能	步数
RCR (带进位循环右移 - 电平)		循环移位	4 至 10
梯形图指令名称	梯形图符号	功能	步数
RCRP (带进位循环右移 - 上升沿)		循环移位	4 至 10

■ 操作数设置

下表列出了 RCR 和 RCRP 指令操作数 S1、 S2 和 D1 的可指定内容。
RCR 和 RCRP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。
操作数 S1 的步数 + 操作数 S2 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 RCR 和 RCRP 指令的步数
(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步 } + {10=1 步 } + {Data_2=1 步 }+ {1 步 } = 4 步。

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面描述了 RCR 和 RCRP 指令操作数 (S1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量]	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647	1	O
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ 操作数设置

下面描述了 RCR 和 RCRP 指令操作数 (S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量]	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	0 至 32	1	O
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ 操作数设置

下面描述了 RCR 和 RCRP 指令操作数 (D1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括输出)	未指定数组和修饰符	1	O
		指定整型变量 [常量]	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ RCR 和 RCRP 指令说明

当执行 RCR 或 RCRP 指令时，会将 S1 的各个位循环右移 S2 个位。将最右侧的位（最低位）保存到进位标志中，而将进位标志（1 或 0）循环移位到最左侧的位（最高位）。最后将结果保存到 D1 中。RCR 和 RCRP 指令总是处于接通状态。使用 RCR 和 RCRP 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。请为操作数 S1 和 D1 指定相同的变量类型。
指定常量时，请参考如下内容。

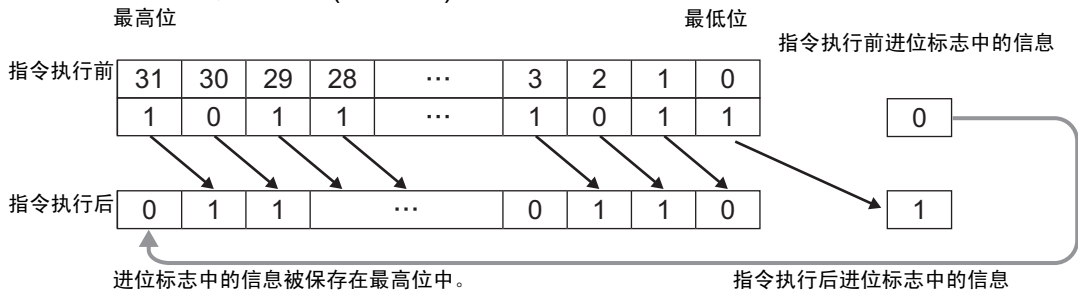
- S1: 循环移位地址

指定要执行循环移位的地址。
- S2: 循环移位位数

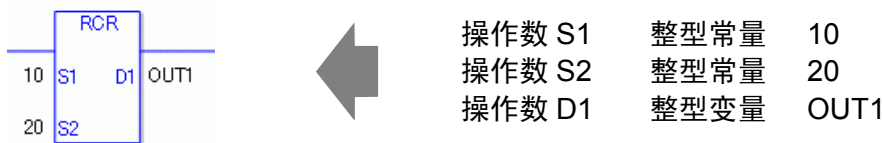
指定将循环移位的位数。
- D1: 存储地址

指定一个保存循环移位结果的地址。

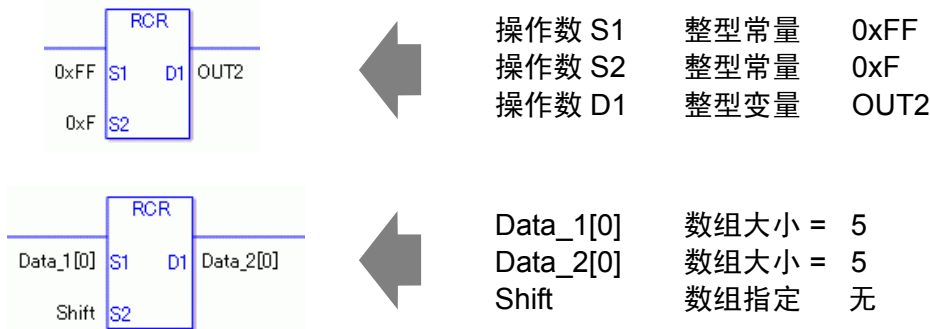
例如，当循环右移 1 位时（带进位）



当操作数 D1 是整型变量时



当操作数 D1 是整型变量，而您又想在操作数 S1 和 S2 中输入 16 进制值时。
当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



如果 S1 和 D1 均不是数组，将带进位循环移动 32 个位。
为 S2 指定一个 0 至 32 间的值。

■ 显示执行结果的系统变量

当执行结果是 0 时，#L CalcZero 置 ON。

如果循环移位造成溢出，将在 #L CalcCarry 中保存溢出的位。

当执行发生错误时，将在 #L Status 中保存错误信息。

当执行发生错误时，将在 `#L CalcErrCode` 中保存错误代码。

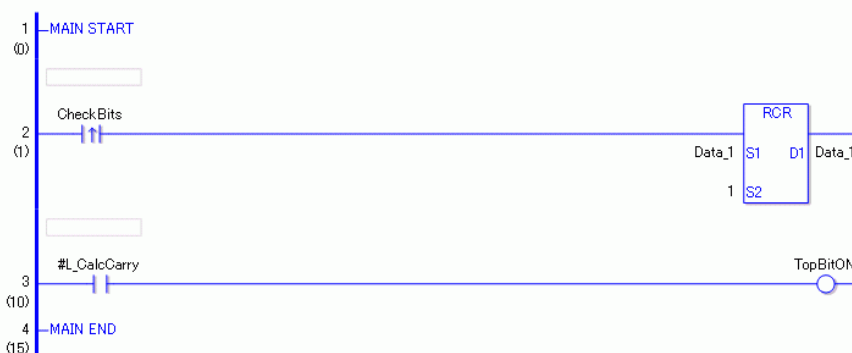
(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

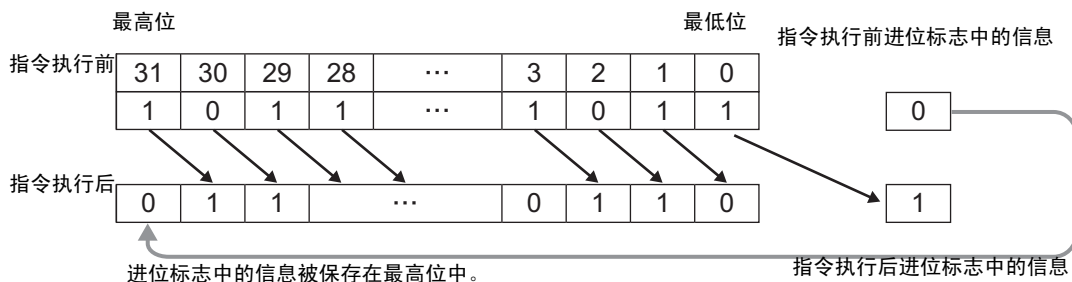
RCR



(1)当上升沿指令置 ON 时，将执行 RCR 指令。当执行 RCR 指令时，会将带进位循环移位 1 位的结果保存在 D1 中。

(2) 当带进位右移 1 位时，您可以用 `#L CalcCarry` 来检查循环移位操作前最低位的值。

(补充)当使用常开指令时,只要该位为 ON,就总是执行 RCR 指令。



程序示例

RCRP

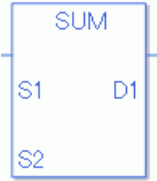
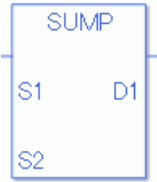


RCRP 和 RCR 指令检测执行时间的方法不同。对于 RCRP 指令，即使使用常开指令，也只有在检测到上升沿时才执行 RCRP 指令。因此，即使常开指令位保持为 ON，也只在一次扫描中执行 RCRP 指令。

31.17 计算函数指令

31.17.1 SUM/SUMP(求和)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
SUM (求和 - 电平)		函数	6 至 10
梯形图指令名称	梯形图符号	功能	步数
SUMP (求和 - 上升沿)		函数	6 至 10

■ 操作数设置

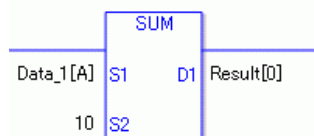
下面描述了 SUM/SUMP 指令操作数 S1、S2 和 D1 的可指定内容。

SUM/SUMP 指令的实际步数取决于操作数的指定方法。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 S2 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 SUM/SUMP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1[0] = 2 步} + {10 = 1 步} + {Result[0] = 2 步} + {1 步} = 6 步。

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面描述了 SUM/SUMP 指令操作数 S1 的可指定内容。

名称	类型	条件	操作数 步数	可用：○ 不可用：X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (不包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量]	2	○
		指定整型变量 [变量]	3	○
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]	2	○
		指定浮点型变量 [变量]	3	○
	实型			X
		指定实型变量 [常量]	2	○
		指定实型变量 [变量]	3	○
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ 操作数设置

下面描述了 SUM/SUMP 指令操作数 S2 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量]	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量		1 至 4096	1	O

■ 操作数设置

下面描述了 SUM/SUMP 指令操作数 D1 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括输出)	未指定数组和修饰符		X
		指定整型变量 [常量]	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型			X
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

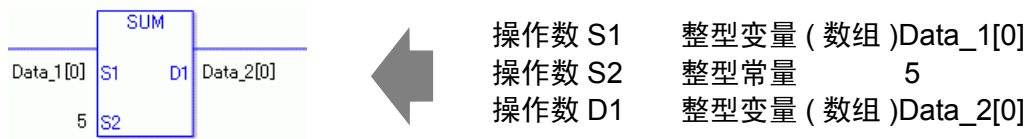
名称	类型	条件	操作数 步数	可用: O 不可用: X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ SUM 和 SUMP 指令说明

SUM/SUMP 指令均执行求和计算。当执行 SUM 指令时，将对从地址 S1 开始的 S2 个数组元素求和，并将结果保存在 D1 中。SUM/SUMP 指令总是处于接通状态。如果为操作数 S1 和 D1 指定的变量类型不同，那么在使用 SUM/SUMP 指令时就会发生错误。请为操作数 S1 和 D1 指定相同的变量类型。

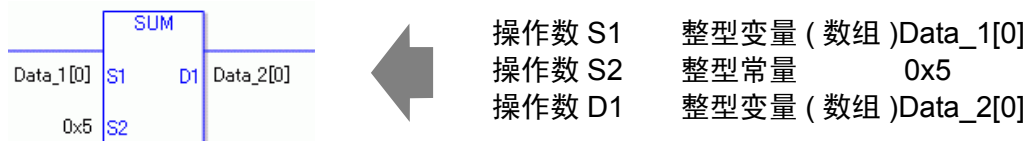
指定常量时，请参考如下内容。

当操作数 D1 是整型变量时



当操作数 D1 是整型变量，而您又想在操作数 S2 中输入 16 进制值时。

当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



确认执行结果

- (1)如果无法识别操作数 S1 或 S2 中的值 (无限或非数值)，将不执行指令。错误检查发现错误时，会将错误代码 “6706” 设置为 #L_CalcErrCode。
输出结果 D1 维持上次成功执行指令时得到的值。

■ 显示执行结果的系统变量

当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

(注释)

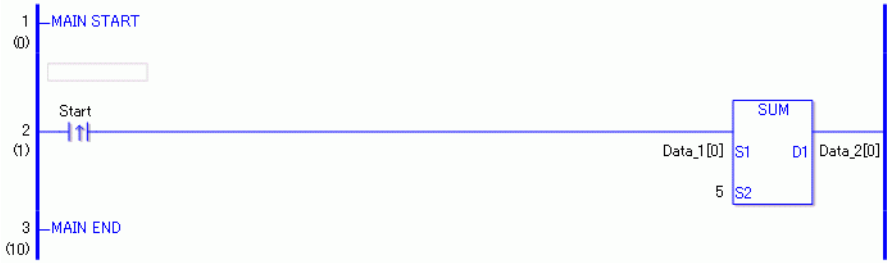
当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

SUM

对 Data_1 中的 1 至 5 求和并把结果保存在 Data_2 中。



(1)当上升沿指令置 ON 时，将执行 SUM 指令。当执行 SUM 指令时，会将数组元素 0 至 5 的总和保存在 D1 中。

当使用常开指令时，只要指令变量为 ON，就总是执行 SUM 指令。

数组变量名称	数据 1	5 个执行的指令	保存在	数据 2
元素	Data_1 [0]	+	-->	Data_2 [0]
	Data_1 [1]			Data_2 [1]
	Data_1 [2]			Data_2 [2]
	Data_1 [3]			Data_2 [3]
	Data_1 [4]			Data_2 [4]
	Data_1 [5]			Data_2 [5]
	Data_1 [6]			Data_2 [6]
	Data_1 [7]			Data_2 [7]
	Data_1 [8]			Data_2 [8]
	Data_1 [9]			Data_2 [9]
	Data_1 [10]			Data_2 [10]

程序示例

SUMP



(1)SUMP 和 SUM 指令检测指令启动的方式不同。即便使用常开指令，也只在检测到上升沿时才执行 SUMP 指令。即便常开指令的变量保持为 ON，也只执行一次 (第一次扫描时)SUMP 指令。

31.17.2 AVE/AVEP(平均值)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
AVE (平均值 - 电平)		函数	6 至 10
梯形图指令名称	梯形图符号	功能	步数
AVEP (平均值 - 上升沿)		函数	6 至 10

■ 操作数设置

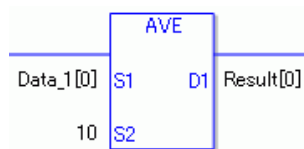
下面描述了 AVE/AVEP 指令操作数 S1、S2 和 D1 的可指定内容。

AVE/AVEP 指令的实际步数取决于操作数的指定方法。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 S2 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 AVE/AVEP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1[0] = 2 步} + {10 = 1 步} + {Result[0] = 2 步} + {1 步} = 6 步。

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面描述了 AVE/AVEP 指令操作数 S1 的可指定内容。

名称	类型	条件	操作数 步数	可用：○ 不可用：X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (不包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量]	2	○
		指定整型变量 [变量]	3	○
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]	2	○
		指定浮点型变量 [变量]	3	○
	实型			X
		指定实型变量 [常量]	2	○
		指定实型变量 [变量]	3	○
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ 操作数设置

下面描述了 AVE/AVEP 指令操作数 S2 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量]	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量		1 至 4096	1	O

■ 操作数设置

下面描述了 AVE/AVEP 指令操作数 D1 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括输出)	未指定数组和修饰符		X
		指定整型变量 [常量]	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型			X
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

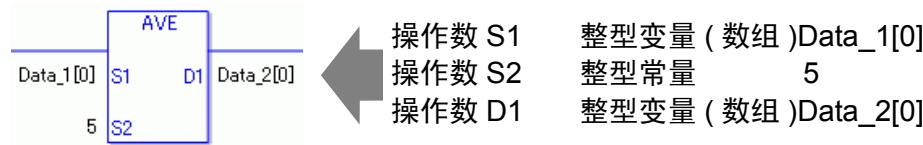
名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ AVE 和 AVEP 指令说明

AVE/AVEP 指令均用于计算平均值。当执行 AVE 指令时，将对从地址 S1 开始的 S2 个数组元素求平均值，并将结果保存在 D1 中。AVE/AVEP 指令总是处于接通状态。如果为操作数 S1 和 D1 指定的变量类型不同，那么在使用 AVE/AVEP 指令时就会发生错误。请为操作数 S1 和 D1 指定相同的变量类型。

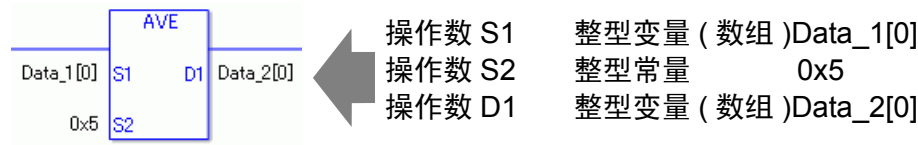
指定常量时，请参考如下内容。

当操作数 D1 是整型变量时



当操作数 D1 是整型变量，而您又想在操作数 S2 中输入 16 进制值时。

当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



确认执行结果

- (1)如果无法识别操作数 S1 或 S2 中的值 (无限或非数值)，将不执行指令。错误检查发现错误时，会将错误代码 “6706” 设置为 #L_CalcErrCode。
输出结果 D1 维持上次成功执行指令时得到的值。

■ 显示执行结果的系统变量

当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

如果没有需要计算的项目，合计为 0，结果也为 0。

程序示例

AVE

对 Data_1 中的 1 至 5 求平均值并把结果保存在 Data_2 中。

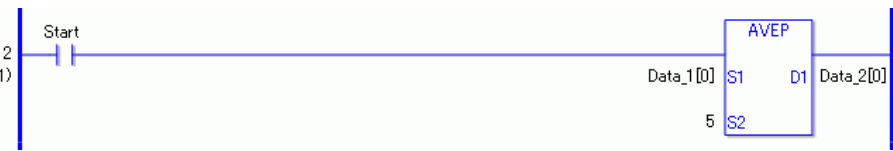


- (1)当上升沿指令置 ON 时，将执行 AVE 指令。当执行 AVE 指令时，将计算数据 1 的 0 至 4 号数组元素的平均值并将 D1 上的结果保存在 Data_2 中。
当使用常开指令时，只要指令变量为 ON，就总是执行 AVE 指令。

数组变量名称	数据 1	5 个执行的指令	保存在	数据 2
元素	Data_1 [0]	+	-->	Data_2 [0]
	Data_1 [1]	+		Data_2 [1]
	Data_1 [2]	+ ÷ 5		Data_2 [2]
	Data_1 [3]	+		Data_2 [3]
	Data_1 [4]			Data_2 [4]
	Data_1 [5]			Data_2 [5]
	Data_1 [6]			Data_2 [6]
	Data_1 [7]			Data_2 [7]
	Data_1 [8]			Data_2 [8]
	Data_1 [9]			Data_2 [9]
	Data_1 [10]			Data_2 [10]

程序示例

AVEP



- (1)AVEP 和 AVE 指令检测指令启动的方式不同。即便使用常开指令，也只在检测到上升沿时才执行 AVEP 指令。即便常开指令的变量保持为 ON，也只执行一次 (第一次扫描时)AVEP 指令。

31.17.3 SQRT/SQRTTP(平方根)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
SQRT (平方根 - 电平)		函数	3 至 7
梯形图指令名称	梯形图符号	功能	步数
SQRTTP (平方根 - 上升沿)		函数	3 至 7

■ 操作数设置

下面描述了 SQRT/SQRTTP 指令操作数 (S1 和 D1) 的可指定内容。

SQRT/SQRTTP 指令的实际步数取决于操作数的指定方法。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 SQRT/SQRTTP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 [0] = 2 步} + {Result[N] = 3 步} + {1 步} = 6 步

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面描述了 SQRT/SQRT_P 指令操作数 (S1 和 D1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (不包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量]		X
		指定整型变量 [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型	浮点型变量	1	O
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型	实型变量	1	O
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量 (不能用于 D1)	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38	1	O
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308	2	O

■ SQRT 和 SQRTP 指令说明

SQRT/SQRTTP 指令均计算平方根。当执行 SQRT 指令时，会计算 S1 的平方根，并将结果保存在 D1 中。

SQRT/SQRTTP 指令总是处于接通状态。如果为操作数 S1 和 D1 指定的变量类型不同，那么在使用 SQRT/SQRTTP 指令时就会发生错误。请在操作数 S1 和 D1 中指定相同的变量类型。

指定常量时，请参考如下内容。

当操作数 D1 是浮点型变量时

当输入 Of(0 和小写字母 “f”) 时，将把后面的值视为浮点型数值。



当操作数 D1 是实型变量时

当输入 Or(0 和小写字母 “r”) 时，将把后面的值视为实数值。



当计算指定数组中的数据时

使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。

如果将 S1 和 D1 指定为整个数组，将会发生错误，即使变量类型相同。



确认执行结果

- (1)如果无法识别操作数 S1 或 S2 中的值 (无限或非数值)，将不执行指令。错误检查发现错误时，会将错误代码 “6706” 设置为 #L_CalcErrCode。
输出结果 D1 维持上次成功执行指令时得到的值。

■ 显示执行结果的系统变量

当执行结果是 0 时，#L_CalcZero 置 ON。

当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

SQRT



(1)当上升沿指令置 ON 时，将执行 SQRT 指令。当执行 SQRT 指令时，会将 Data_A 的平方根保存在 D1 的计算结果 (实型 / 浮点型变量) 中。

当使用常开指令时，只要指令变量为 ON，就总是执行 SQRT 指令。

程序示例

SQRTP



(1)SQRTP 和 SQRT 指令检测指令启动的方式不同。即便使用常开指令，也只在检测到上升沿时才执行 SQRTP 指令。即便常开指令的变量保持为 ON，也只执行一次 (第一次扫描时)SQRTP 指令。

31.17.4 BCNT/BCNTP(位计数)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
BCNT (位计数 - 电平)		函数	3 至 9
梯形图指令名称	梯形图符号	功能	步数
BCNTP (位计数 - 上升沿)		函数	3 至 9

■ 操作数设置

下面描述了 BCNT/BCNTP 指令操作数 S1 和 D1 的可指定内容。

BCNT/BCNTP 指令的实际步数取决于操作数的指定方法。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 BCNT/BCNTP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{ Data_1[1] = 2 步 } + { Result[Indirect] = 3 步 } + { 1 步 } = 6 步

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下表列出了 BCNT/BCNTP 指令操作数 (S1 和 D1) 的可配置条件。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式 *(注释 1) S1 = 启用 IO D1 = 禁用 IO	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 * (注释 1)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]	2	O
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]	4	O
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

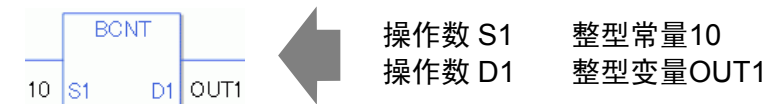
名称	类型	条件	操作数 步数	可用：○ 不可用：×
地址格式 *(注释 2) D1 = 禁用	X_			X
	Y_			X
	M_			X
	I_* (注释 2)		1	○
	Q_		1	○
	D_	未指定修饰符	1	○
		D_****.B/W [常量]	2	○
		D_****.B/W [地址]	3	○
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	○
	C_	仅 .PV / .CV	2	○
	N_	仅 .YR / .MO / .DAY	2	○
	J_	仅 .HR / .MIN / .SEC	2	○
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	○
常量 *(注释 3) D1 = 禁用	整型 * (注释 3)	-2147483648 至 2147483647	1	○
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ BCNT 和 BCNTP 指令说明

BCNT/BCNTP 指令均执行位计数。当执行 BCNT 指令时，将对 S1 数据中置 ON 的位进行计数，并将结果保存在 D1 中。BCNT/BCNTP 指令总是处于接通状态。如果为操作数 S1 和 D1 指定的变量类型不同，那么在使用 BCNT/BCNTP 指令时就会发生错误。

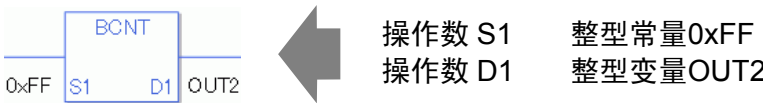
请在操作数 S1 和 D1 中指定相同的变量类型。
指定常量时，请参考如下内容。

当操作数 D1 是整型变量时



当操作数 D1 是整型变量，而您又想在操作数 S2 中输入 16 进制值时。

当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



当计算指定数组 (整型变量数组) 中的数据时

使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。

如果将 S1 和 D1 指定为整个数组，将会发生错误，即使变量类型相同。



■ 显示执行结果的系统变量

当执行结果是 0 时，#L_CalcZero 置 ON。

当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

BCNT

对处于 ON 状态的位进行计数，并将计数结果保存在一个整型变量中。



(1)当上升沿指令置 ON 时，将执行 BCNT 指令。当执行 BCNT 指令时，对数据 10 (二进制 1010) 中为 ON 的位进行计数并将结果 2 保存在结果数据中。最后将结果数据配置在 D1 中。

当使用常开指令时，只要指令变量为 ON，就总是执行 BCNT 指令。

程序示例

BCNTP



(1)BCNTP 和 BCNT 指令在运行时间上有所不同。对于 BCNTP 指令，即使使用常开指令，也只有在检测到上升沿时才执行 BCNT 指令。即便常开指令的变量保持为 ON，也只执行一次 (第一次扫描时)BCNTP 指令。

31.17.5 PID

符号和功能

梯形图指令名称	梯形图符号	功能	步数
PID (PID - 电平)		函数	10 至 18

■ PID 指令说明

PID 指令中的 PID 变量是结构变量。不能为操作数 HP 分配非 PID 变量 (地址格式: U_)。有关分配给操作数 HP 的 PID 变量的内部结构信息, 请参阅下表。

PID 变量

PID 变量	变量设置	描述
变量名 .Q	位变量	PID 指令处理完成标志
变量名 .PF	位变量	处理无效范围标志
变量名 .UO	位变量	输出值超出上限
变量名 .TO	位变量	输出值超出下限
变量名 .IF	位变量	积分设置
变量名 .KP	整型变量	比例系数
变量名 .TR	整型变量	积分时间
变量名 .TD	整型变量	微分时间
变量名 .PA	整型变量	处理无效范围
变量名 .BA	整型变量	偏差 (偏移)
变量名 .ST	整型变量	采样频率

其他操作数如下。

- S1: 目标值
- S2: 预设值
- S3: 默认值 (当禁用指令时输出设定值)
- D1: 当前值

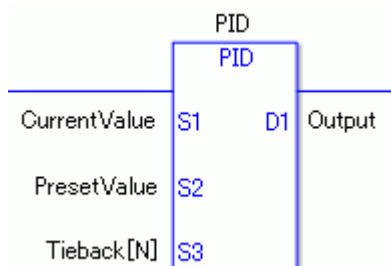
■ 操作数设置

下面描述了 PID 指令操作数 S1、S2、S3 和 D1 的可指定内容。PID 指令的实际步数取决于操作数的指定方法。下面描述如何计算步数。

HP 操作数的步数 + 操作数 S1 的步数 + 操作数 S2 的步数 + 操作数 S3 的步数 + 操作数 D1 的步数 + 5 = 一个指令的总步数

例如，计算 PID 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



(有关操作数步数的信息，请参阅下一页中的操作数设置。) {PID 控制 = 1 步 (HP 操作数中的 PID 变量固定为 1 步)} + { 当前值 = 1 步 } + { 设定值 = 1 步 } + { 默认值 [N] = 3 步 } + { 输出 = 1 步 } + { 5 步 } = 12 步

PID 指令还包括最后 5 个步骤。必须加上这 5 步。

■ 操作数设置

下面描述了 PID 指令操作数 S1、S2、S3 和 D1 的可指定内容。

名称	类型	条件	操作数 步数	可用：○ 不可用：X
外部寄存器 地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	○
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	○
符号	位			X
	字		1	○
变量格式 * (注释 1) S1, S2, S3 可使用 I/O D1 = 不可使 用 I/O	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 * (注释 1)	未指定数组和修饰符	1	○
		指定整型变量 [常量]	2	○
		指定整型变量 [变量]	3	○
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	○
	计数器	仅 .PV / .CV	2	○
	日期	仅 .YR / .MO / .DAY	2	○
	时间	仅 .HR / .MIN / .SEC	2	○
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	○

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式 * (注释 2) D1 = 不可用	X_			X
	Y_			X
	M_			X
	I_* (注释 2)		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]	2	O
		D_****.B/W [地址]	3	O
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量 * (注释 3) D1、S2 = 不可用	整型 * (注释 3)	-2147483648 至 2147483647	1	O
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ 基本功能

PID 指令比较测量值 (当前值) 和设定值 (目标值)。测量值基于模拟输入和温度输入。该指令然后会调节输出值, 缩小当前值和目标值之间的差异。在 PID 控制中您可以将 P(比例) 控制、I(积分) 控制和 D(微分) 控制结合起来。请指定下述要控制的参数。

一般用下面的公式表达 PID 控制计算出来的输出值。

$$CV=KP(E+\text{Reset} \int_0^t (E)dt +\text{Rate} \frac{d(E)}{dt})$$

KP : 比例系数

E : 偏差 (SP-PV 或 PV-SP)

Reset: 积分周期数

Rate : 微分时间

使用后面将要描述的 [微调] 标签, 调节采样时间, 以降低噪声对偏差的影响。下述公式给出了过滤偏差的结果。

$$EF_n = EF_{n-1} + \frac{T_{\text{Loop}}}{T_{\text{Filter}}} (E_n - EF_{n-1})$$

EF : 过滤偏差的结果

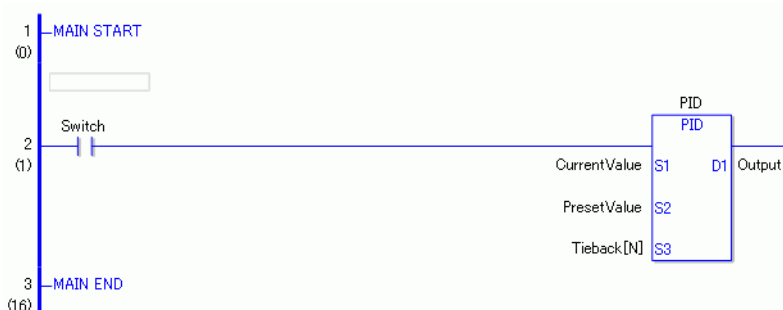
T_{loop} : 频率数据

T_{Filter} : 采样频率

E : 偏差 (SP-PV 或 PV-SP)

■ 功能摘要

当启用 PID 指令时，将计算 PID 并调节和输出 (计算) 运算量。当如下所示禁用指令时，PID 输出默认值。默认值是在 S3 中指定的。如果禁用指令后没有必要输出，则输入常量 0。



要在逻辑程序中使用 PID 指令，首先将变量分配给 PID 变量操作数 (HP) 和整型变量操作数 (S1, S2, S3 和 D1)。

PID 变量

当为 PID 指令操作数 HP 分配变量时，会自动为该变量分配一个成员。

PID 变量

PID 变量	变量设置	描述
变量名 .Q	位变量	PID 指令处理完成标志
变量名 .PF	位变量	处理无效范围标志
变量名 .UO	位变量	输出值超出上限
变量名 .TO	位变量	输出值超出下限
变量名 .IF	位变量	积分设置
变量名 .KP	整型变量	比例系数
变量名 .TR	整型变量	1 积分时间
变量名 .TD	整型变量	1 微分时间
变量名 .PA	整型变量	处理无效范围
变量名 .BA	整型变量	偏差 (偏移)
变量名 .ST	整型变量	采样频率

- 分配给比例系数、积分时间和微分时间的值，在“PID 监控”中输入的值和在程序中输入各 PID 变量时的值看起来是不同的。当在程序中输入值时，用 1000 乘以比例系数、积分时间和微分时间。

例如，比例系数 $0.1 \times 1000 \rightarrow 100$

(注释)

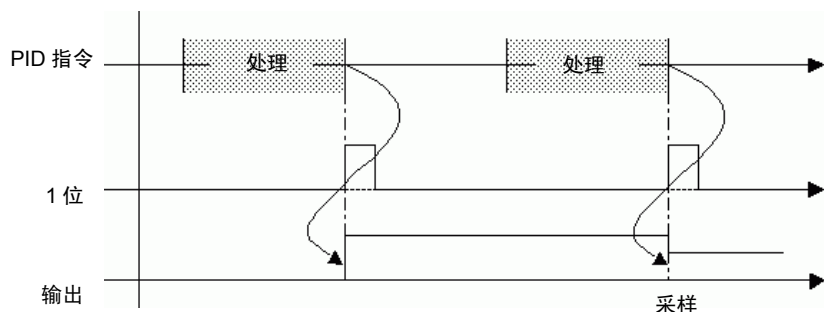
所有 PID 变量都是保持型变量。每个工程最多允许有 8 个 PID 指令。

可为 1 个 PID 变量指定一个 PID 指令。

■ PID 变量成员说明

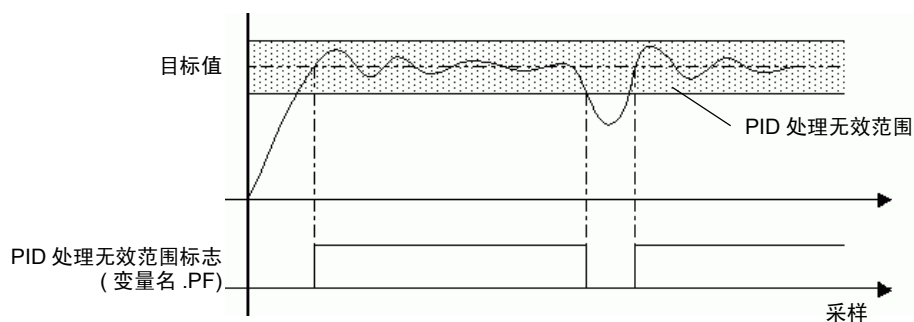
PID 指令处理完成标志 (变量名 .Q)

当处理完成后将值输出到操作数 D1 时，.Q 置 ON。当正在执行 1 个扫描时 PID 指令的完成标志置 ON。



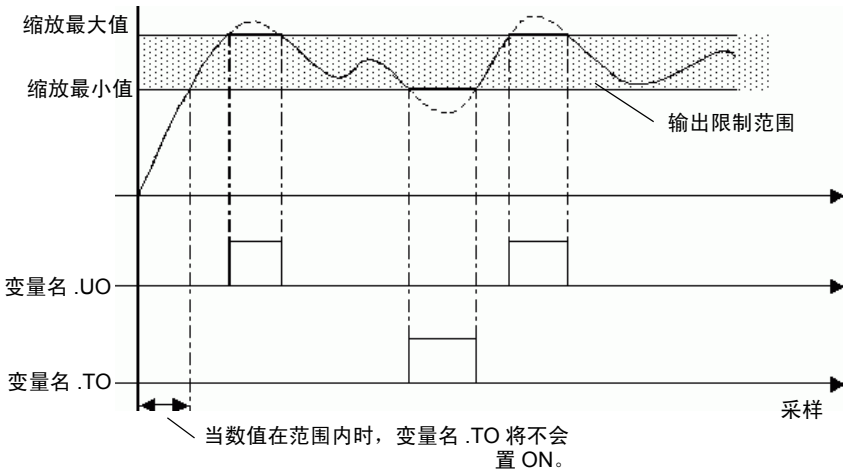
处理无效范围标志 (变量名 .PF)

当当前值达到通过指定 PID 变量 (处理无效范围变量名 .PF) 指定的范围内的目标值时，该标志置 ON，当当前值超出该范围时，该标志置 OFF。



输出值超出上限 / 下限 (变量名 .UO、变量名 .TO)

双击 PID 指令，显示指定 PID 变量输出范围的对话框。如果计算结果超出指定的输出值，变量名 .UO 置 ON。如果结果小于指定下限，变量名 .TO 置 ON。PID 继续，即便状态位置 ON，计算的值作为指定上限或下限输出。



积分设置 (变量名 .IF)

双击 PID 指令，显示对话框，设置执行 PID 指令的范围。如果结果超出指定的积分设置，.IF 置 ON。每个状态的积分设置只在范围内执行积分计算。

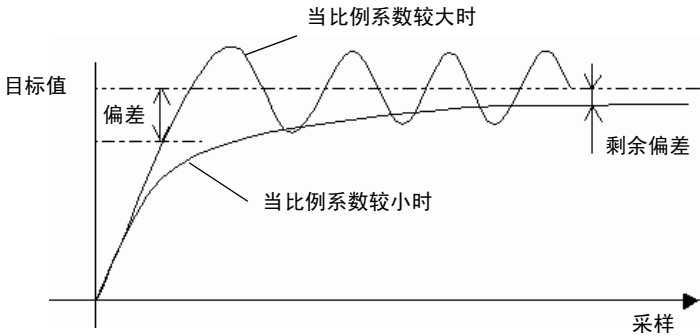
比例系数 (变量名 .KP)

指定一个比例系数 (变量名 .KP)，对应于目标值与当前值之间的偏差输出一个值。较小的比例系数会产生较小的输出值以达到目标值，能消除超调但可能会增加剩余偏差。较大的比例系数会产生较大的输出值以达到目标值，能够减少达到目标值的时间，但可能会导致振荡。

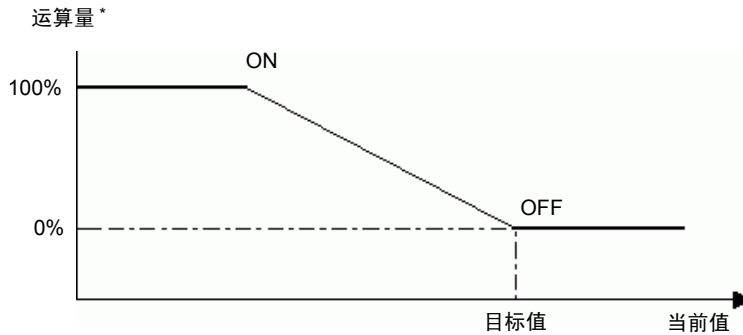
设置范围在 0.01 至 1000.00 之间内部数据是整型变量。不能使用小数。

要设置 0.01，请使用 $0.01 \times 1000 = 10$ 。

将变量名 .KP 指定为乘以 1000 的值。



(注释) 在比例控制中, 如果当前值小于目标值, 运算量将为 100%。如果目标值与当前值相等 (无偏差), 运算量将是 0%。



* 运算量: 每单位时间的输出

积分时间 (变量名 .TR)

通过设置积分时间 (.TR), 可以消除与目标值的偏差。

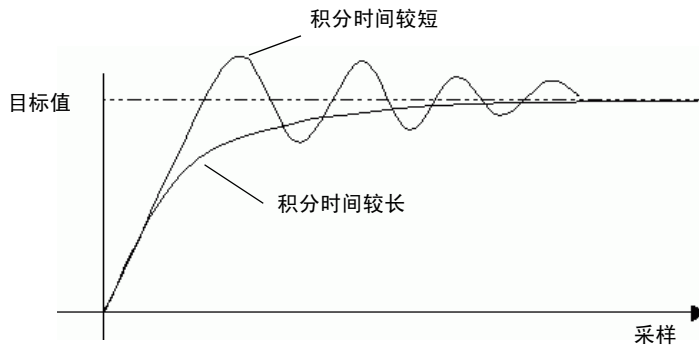
当使用比例控制使运算量接近目标值时, 偏差变得很小, 这样运算量就不足以抵消偏差。微小的偏差被称为剩余偏差。可通过积分控制来消除偏差。当偏差累积达到一定程度时, 积分控制通过增加运算量来调节偏差。当积分时间变短时, 达到目标值的运算量就变大, 从而造成超调和振荡并在较短的时间内达到目标值。同样, 随着积分时间变长, 达到目标值的运算量变小, 从而减少了超调和振荡, 但达到目标值的时间变长。

积分时间指定了执行积分处理的间隔时间 (以秒为单位)。

设置范围在 0.100 至 3000.000 之间内部数据是整型变量。不能使用小数。

要设置 0.1, 请指定 $0.1 \times 1000 = 100$ 。

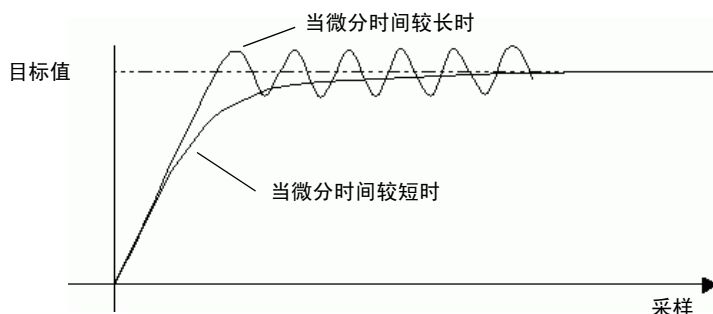
将变量名 .TR 指定为乘以 1000 的值。



微分时间 (变量名 .TD)

通过设置微分时间 (.TD), 可以快速响应任何变化。

比例控制和积分控制需要一定的时间 (时间常量), 不能立即响应外部干扰。返回原始目标值需要一定的时间。如果与外部干扰相比, 当前值和先前值之间的偏差较大时, 微分控制可立即响应并分配较大运算量。当微分时间较长时, 从外部干扰的影响中恢复所需的时间就较短, 但会导致超调和频繁的振荡。当微分时间较短时, 则能够减小超调和振荡, 但从外部干扰的影响中恢复就需要较长的时间。



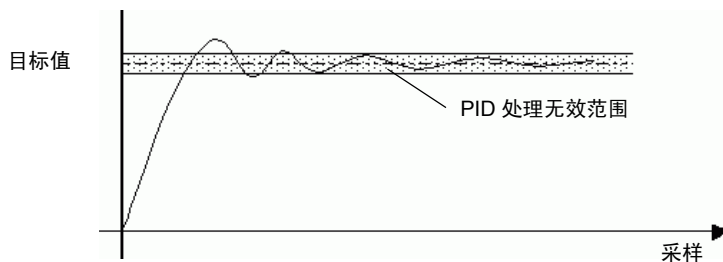
设置范围为 0.00 至 3000.00 内部数据为整型变量, 且不能使用小数。

要设置 0.1, 请使用 $0.1 \times 1000 = 100$ 。

为变量名 .TD 指定乘以 1000 的值。

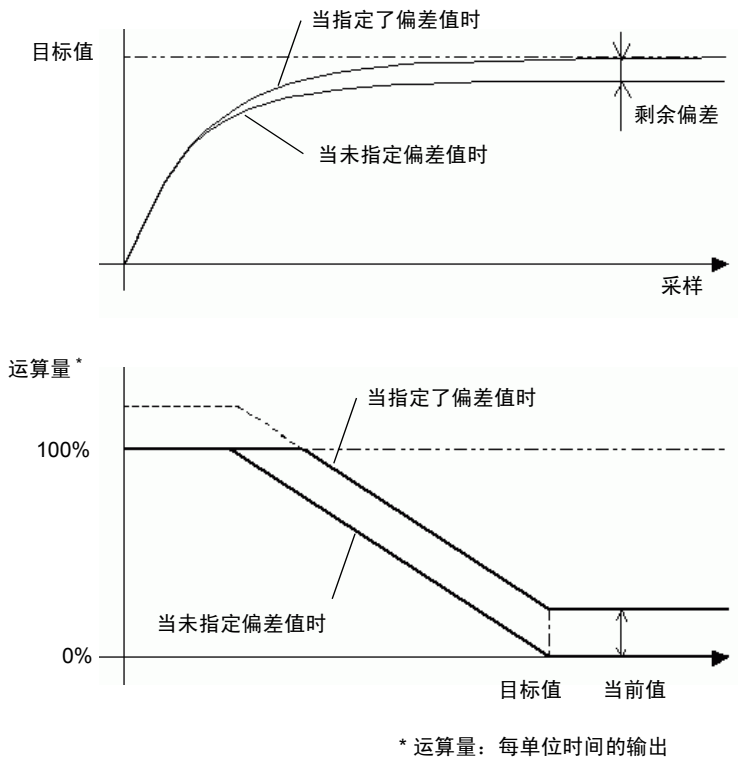
处理无效范围 (变量名 .PA)

在“处理无效范围”中, 不进行 PID 控制, 输出最小值以实现无振荡的平稳控制。



偏差 (变量名 .BA)

设置偏差值 (偏移)。这可以减少比例控制导致的任何剩余偏差。



采样频率 (变量名 .ST)

这消除了从控制设置频率中获得的 S2 值的噪声。移动平均值根据上一次过滤结果和新得到的数据进行计算。若当前数据包含意外值时, 指定采样频率可将输出值的影响降至最小。这是因为计算时使用了以前测量的数据及当前数据的平均值。请为采样频率指定一个比控制设置频率更大的值。若要禁用过滤器, 请将采样频率指定为 0。

■ 双击 PID 指令进行设置

双击 PID 指令指定 PID 变量。



输出设置 (操作数 D1 的范围)

指定输出值的上限和下限。计算结果必须在该范围内。

固定设置	输出范围是 0 到 4095。		
用户设置	根据要求指定输出范围。		
	上限范围	下限 +1 至 32767	
	下限范围	0 至上限 -1	

积分设置

指定积分设置的上限和下限。

控制设置

循环更新时间：设置获得 S2 数据的时间频率。获得数据的频率也是更新 D1 输出的频率。

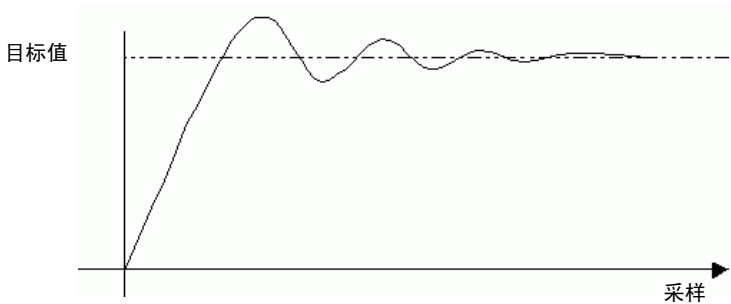
您可以使用过滤功能来指定频率。采样频率必须大于获取数据的频率。

设置范围在 10 至 65535 ms 之间

操作：

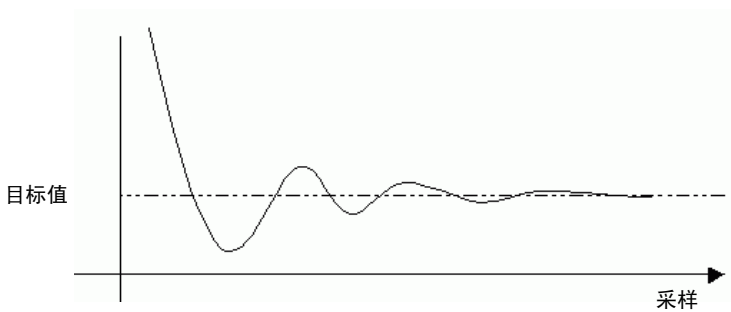
直接 (D1-D2)

当处理变量小于设定值时用来控制运算量的增加。(加热等)



反转 (D1-D2)

当处理变量大于设定值时用来控制运算量的增加。(例如，冷却)



过滤设置

指定输出值的上限和下限。如果数值超出范围，将输出上限或下限值。当数值超出范围时，超出上限和超出下限 (变量名 .UO, 变量名 .TO) 位置 ON。

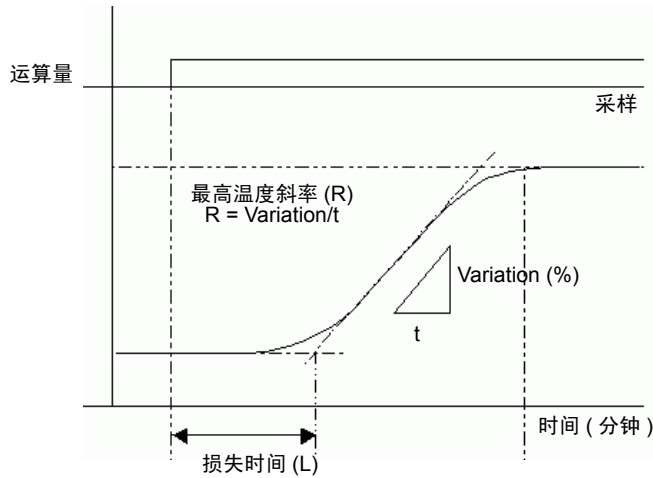
设置范围 取决于输出设置范围

上限 输出设置范围 (上限) 至 32767

下限 输出设置范围 (下限) 至 -32768

■ PID 常数调节

下面以温度控制为例进行说明。要优化 PID 控制的结果，需要优化 P(比例元素)、I(整型元素) 和 D(微分元素) 的常数值。可以使用阶跃响应法来获取不同设定值的 PID 温度常数。需要注意的是，根据使用和设定值的情况，可能不对该值进行优化。在这种情况下，执行在线监控并调整 PID 监视器窗口中的值。为阶跃响应法指定设定值并将 100% 的运算量输出到控制目标等级上。这时，测量下面显示的温度曲线的最高温度斜率 (R) 和损失时间 (L)。



在下面的公式中为最高温度斜率 (R) 和损失时间 (L) 插入测量值，以计算比例系数、积分时间和微分时间常数。将计算值分配到 PID 监控窗口中。

“比例系数” = $100 / (0.83 \cdot R \cdot L)$ [%]

“积分时间” = $1 / (2 \cdot L)$ [事件数 / 分钟] (公式 = 未确认)

“微分时间” = $0.5 \cdot L$ [分钟]

31.18 三角函数指令

31.18.1 SIN 和 SINP(正弦)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
SIN (正弦 - 电平)		三角函数	3 至 7
梯形图指令名称	梯形图符号	功能	步数
SINP (正弦 - 上升沿)		三角函数	3 至 7

■ 操作数设置

下面描述了 SIN 和 SINP 指令操作数 S1 和 D1 的可指定内容。
SIN 和 SINP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。
操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 SIN 和 SINP 指令的步数
(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 [0] = 2 步 } + {Result[N] = 3 步 } + {1 步 } = 6 步

指令的总步数需要加最后一步。请务必加一步。

■ 操作数设置

下面描述了操作数 (S1) 和 (D1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (不包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量]		X
		指定整型变量 [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型	浮点型变量	1	O
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型	实型变量	1	O
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量 (不能用于 D1)	整型	-2147483648 至 2147483647		X
	浮点型	$\pm 1.175494351\text{e-}38$ 至 $\pm 3.402823466\text{e+}38$	1	O
	实型	$\pm 2.2250738585072014\text{e-}308$ 至 $\pm 1.7976931348623158\text{e+}308$	2	O

■ SIN 和 SINP 指令说明

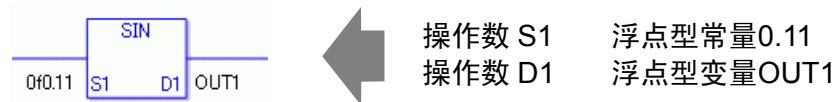
SIN 和 SINP 指令是正弦三角函数指令。SIN 指令计算 S1 数据的正弦并将结果保存在 D1 中。

在 S1 中输入一个弧度值，在 D1 中获得结果，其值是 -1.0 和 1.0 间的一个实数。SIN 和 SINP 指令总是处于接通状态。使用 SIN 和 SINP 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。请在操作数 S1 和 D1 中指定相同的变量类型。

指定常量时，请参考如下内容。

当操作数 D1 是浮点型变量时

当输入 0f(0 和小写字母 “f”) 时，将把后面的值视为浮点型数值。



当操作数 D1 是实型变量时

当输入 0r(0 和小写字母 “r”) 时，将把后面的值视为实数值。



当计算指定数组中的数据时

使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。

如果将 S1 和 D1 指定为整个数组，将会发生错误，即使变量类型相同。



显示执行结果的系统变量

当执行结果是 0 时，#L_CalcZero 置 ON。

当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

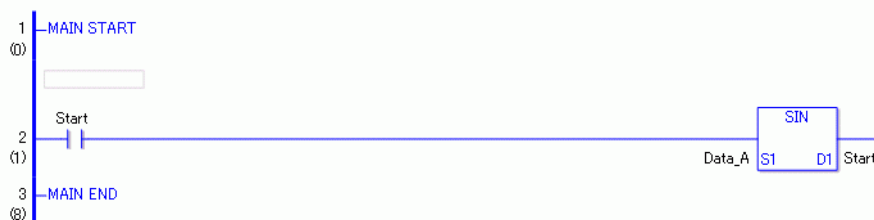
(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

SIN



(1)当上升沿指令置 ON 时，将执行 SIN 指令。SIN 指令计算 Data_A 的正弦并将结果保存在 D1 中。

当使用常开指令时，只要常开指令为 ON，就总是执行 SIN 指令。

程序示例

SINP



(1)SINP 和 SIN 指令在运行时间上有所不同。对于 SINP 指令，即使使用常开指令，也只有在检测到上升沿时才会执行 SINP 指令。因此，即使常开指令保持为 ON，也只在一次扫描中执行 SINP 指令。

31.18.2 COS 和 COSP(余弦)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
COS (余弦 - 电平)		三角函数	3 至 7
梯形图指令名称	梯形图符号	功能	步数
COSP (余弦 - 上升沿)		三角函数	3 至 7

■ 操作数设置

下面描述了 COS 和 COSP 指令操作数 S1 和 D1 的可指定内容。
 COS 和 COSP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。
 操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 COS 和 COSP 指令的步数
 (有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 [0] = 2 步 } + {Result[N] = 3 步 } + {1 步 } = 6 步

指令的总步数需要加最后一步。请务必加一步。

■ 操作数设置

下面描述了 COS 和 COSP 指令操作数 S1 和 D1 的可指定内容。

名称	类型	条件	操作数 步数	可用：O 不可用：X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (不包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量]		X
		指定整型变量 [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型	浮点型变量	1	O
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型	实型变量	1	O
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量 (不能用于 D1)	整型	-2147483648 至 2147483647		X
	浮点型	$\pm 1.175494351\text{e-}38$ 至 $\pm 3.402823466\text{e+}38$	1	O
	实型	$\pm 2.2250738585072014\text{e-}308$ 至 $\pm 1.7976931348623158\text{e+}308$	2	O

■ COS 和 COSP 指令说明

COS 和 COSP 指令是余弦三角函数指令。COS 指令计算 S1 数据的余弦并将结果保存在 D1 中。在 S1 中输入一个弧度值，在 D1 中获得结果，其值是 -1.0 和 1.0 间的一个实数。

COS 和 COSP 指令总处于接通状态。使用 COS 和 COSP 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。请在操作数 S1 和 D1 中指定相同的变量类型。

指定常量时，请参考如下内容。

当操作数 D1 是浮点型变量时

当输入 0f(0 和小写字母 “f”) 时，将把后面的值视为浮点型数值。



当操作数 D1 是实型变量时

当输入 0r(0 和小写字母 “r”) 时，将把后面的值视为实数值。



当计算指定数组中的数据时

使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。

如果将 S1 和 D1 指定为整个数组，将会发生错误，即使变量类型相同。



■ 显示执行结果的系统变量

当执行结果是 0 时，#L_CalcZero 置 ON。

当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

COS



(1)当上升沿指令置 ON 时，将执行 COS 指令。COS 指令计算 Data_A 的余弦并将结果保存在 D1 中。

当使用常开指令时，只要常开指令为 ON，就总是执行 COS 指令。

程序示例

COSP



(1)COSP 和 COS 指令在运行时间上有所不同。对于 COSP 指令，即使使用常开指令，也只有在检测到上升沿时才会执行 COSP 指令。因此，即使常开指令保持为 ON，也只在一次扫描中执行 COSP 指令。

31.18.3 TAN 和 TANP(正切)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
TAN (正切 - 电平)		三角函数	3 至 7
梯形图指令名称	梯形图符号	功能	步数
TANP (正切 - 上升沿)		三角函数	3 至 7

■ 操作数设置

下面描述了 TAN 和 TANP 指令操作数 S1 和 D1 的可指定内容。

TAN 和 TANP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 TAN 和 TANP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 [0] = 2 步 } + {Result[N] = 3 步 } + {1 步 } = 6 步

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面描述了 TAN 和 TANP 指令操作数 S1 和 D1 的可指定内容。

名称	类型	条件	操作数 步数	可用：○ 不可用：X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (不包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量]		X
		指定整型变量 [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型	浮点型变量	1	○
		指定浮点型变量 [常量]	2	○
		指定浮点型变量 [变量]	3	○
	实型	实型变量	1	○
		指定实型变量 [常量]	2	○
		指定实型变量 [变量]	3	○
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：○ 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_		1	○
	R_		1	○
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38	1	○
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308	2	○

■ TAN 和 TANP 指令说明

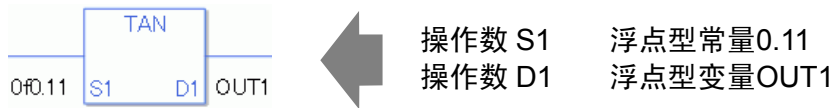
TAN 和 TANP 指令是正切三角函数指令。当 TAN 指令执行并接通回路时，将计算 S1 值的正切，并将结果保存在 D1 中。S1 值是以弧度定义的，D1 的值会出现小数数据，因此应将 D1 设置为实型变量或浮点型变量。

TAN 和 TANP 指令总是处于接通状态。使用 TAN 和 TANP 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。请在操作数 S1 和 D1 中指定相同的变量类型。

指定常量时，请参考如下内容。

当操作数 D1 是浮点型变量时

当输入 0f(0 和小写字母 “f”) 时，将把后面的值视为浮点型数值。



当操作数 D1 是实型变量时

当输入 0r(0 和小写字母 “r”) 时，将把后面的值视为实数值。



当计算指定数组中的数据时

使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。

如果将 S1 和 D1 指定为整个数组，将会发生错误，即使变量类型相同。



■ 显示执行结果的系统变量

当执行结果是 0 时，#L_CalcZero 置 ON。

当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

TAN



(1)当上升沿指令置 ON 时，将执行 TAN 指令。TAN 指令计算 Data_A 的正切并将结果保存在 D1 中。

当使用常开指令时，只要常开指令为 ON，就总是执行 TAN 指令。

程序示例

TANP



(1)TANP 和 TAN 指令在运行时间上有所不同。对于 TANP 指令，即使使用常开指令，也只有在检测到上升沿时才执行 TANP 指令。因此，即使常开指令位保持为 ON，也只在一次扫描中执行 TANP 指令。

31.18.4 ASIN 和 ASINP(反正弦)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
ASIN (反正弦 - 电平)		三角函数	3 至 7
梯形图指令名称	梯形图符号	功能	步数
ASINP (反正弦 - 上升沿)		三角函数	3 至 7

■ 操作数设置

下面描述了 ASIN 和 ASINP 指令操作数 S1 和 D1 的可指定内容。

ASIN 和 ASINP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 ASIN 和 ASINP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 [0] = 2 步 } + {Result[N] = 3 步 } + {1 步 } = 6 步

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面描述了 ASIN 和 ASINP 指令操作数 S1 和 D1 的可指定内容。

名称	类型	条件	操作数 步数	可用：○ 不可用：X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (不包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量]		X
		指定整型变量 [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型	浮点型变量	1	○
		指定浮点型变量 [常量]	2	○
		指定浮点型变量 [变量]	3	○
	实型	实型变量	1	○
		指定实型变量 [常量]	2	○
		指定实型变量 [变量]	3	○
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量 (不能用于 D1)	整型	-2147483648 至 2147483647		X
	浮点型	$\pm 1.175494351\text{e-}38$ 至 $\pm 3.402823466\text{e+}38$	1	O
	实型	$\pm 2.2250738585072014\text{e-}308$ 至 $\pm 1.7976931348623158\text{e+}308$	2	O

■ ASIN 和 ASINP 指令说明

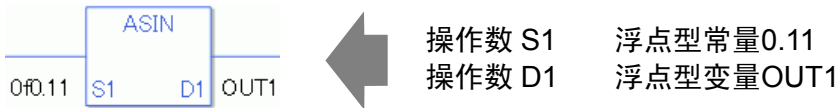
ASIN 和 ASINP 指令是反正弦三角函数指令。ASIN 指令计算 S1 的反正弦并将结果保存在 D1 中。 $\text{Sin}^{-1}(\text{S1})$ 被保存在 D1 中。请为 S1 输入一个 -1.0 至 1.0 间的值，D1 中的结果是一个以弧度表示的实数，其值在 $-\text{Pi}/2$ 和 $\text{Pi}/2$ 之间。Pi 约等于 3.1415926535897(实数)。

ASIN 和 ASINP 指令总是处于接通状态。使用 ASIN 和 ASINP 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。请在操作数 S1 和 D1 中指定相同的变量类型。

指定常量时，请参考如下内容。

当操作数 D1 是浮点型变量时

当输入 0f(0 和小写字母 “f”) 时，将把后面的值视为浮点型数值。



当操作数 D1 是实型变量时

当输入 0r(0 和小写字母 “r”) 时，将把后面的值视为实数值。



当计算指定数组中的数据时

使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。

如果将 S1 和 D1 指定为整个数组，将会发生错误，即使变量类型相同。



■ 显示执行结果的系统变量

当执行结果是 0 时，#L_CalcZero 置 ON。

当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

ASIN



(1)当上升沿指令置 ON 时，将执行 ASIN 指令。ASIN 指令计算 Data_A 的反正弦并将结果保存在 D1 中。

当使用常开指令时，只要常开指令位为 ON，就总是执行 ASIN 指令。

程序示例

ASINP



(1)ASINP 和 ASIN 指令在运行时间上有所不同。对于 ASINP 指令，即使使用常开指令，也只有在检测到上升沿时才执行 ASINP 指令。因此，即使常开指令位保持为 ON，也只在一次扫描中执行 ASINP 指令。

31.18.5 ACOS 和 ACOSP(反余弦)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
ACOS (反余弦 - 电平)		三角函数	3 至 7
梯形图指令名称	梯形图符号	功能	步数
ACOSP (反余弦 - 上升沿)		三角函数	3 至 7

■ 操作数设置

下面描述了 ACOS 和 ACOSP 指令操作数 S1 和 D1 的可指定内容。

ACOS 和 ACOSP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 ACOS 和 ACOSP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 [0] = 2 步 } + {Result[N] = 3 步 } + {1 步 } = 6 步

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面描述了 ACOS 和 ACOSP 指令操作数 S1 和 D1 的可指定内容。

名称	类型	条件	操作数 步数	可用：O 不可用：X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (不包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量]		X
		指定整型变量 [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型	浮点型变量	1	O
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型	实型变量	1	O
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：○ 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_		1	○
	R_		1	○
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量 (不能用于 D1)	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38	1	○
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308	2	○

■ ACOS 和 ACOSP 指令说明

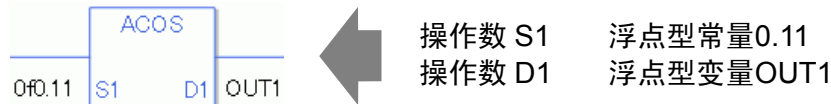
ACOS 和 ACOSP 指令是反余弦三角函数指令。ACOS 指令计算 S1 的反余弦并将结果保存在 D1 中。COS-1(S1) 被保存在 D1 中。请为 S1 输入一个 -1.0 至 1.0 间的值，D1 中的结果是一个以弧度表示的实数，在 0 和 Pi 之间。Pi 约等于 3.1415926535897(实数)。

ACOS 和 ACOSP 指令总是处于接通状态。使用 ACOS 和 ACOSP 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。请在操作数 S1 和 D1 中指定相同的变量类型。

指定常量时，请参考如下内容。

当操作数 D1 是浮点型变量时

当输入 0f(0 和小写字母 “f”) 时，将把后面的值视为浮点型数值。



当操作数 D1 是实型变量时

当输入 0r(0 和小写字母 “r”) 时，将把后面的值视为实数值。



当计算指定数组中的数据时

使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。

如果将 S1 和 D1 指定为整个数组，将会发生错误，即使变量类型相同。



■ 显示执行结果的系统变量

当执行结果是 0 时，#L_CalcZero 置 ON。

当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

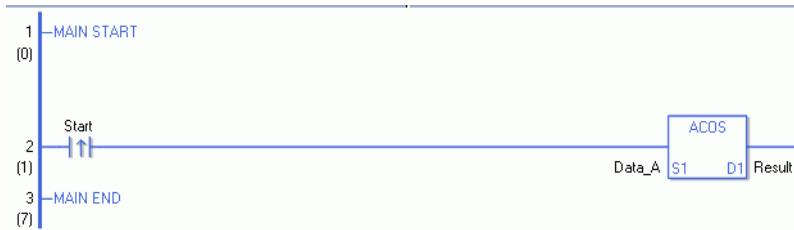
(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

ACOS



(1)当上升沿指令置 ON 时，将执行 ACOS 指令。ACOS 指令计算 Data_A 的反余弦并将结果保存在 D1 中。

当使用常开指令时，只要常开指令位为 ON，就总是执行 ACOS 指令。

程序示例

ACOSP



(1)ACOSP 和 ACOS 指令在运行时间上有所不同。对于 ACOSP 指令，即使使用常开指令，也只有在检测到上升沿时才执行 ACOSP 指令。因此，即使常开指令位保持为 ON，也只在一次扫描中执行 ACOSP 指令。

31.18.6 ATAN 和 ATANP(反正切)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
ATAN (反正切 - 电平)		三角函数	3 至 7
梯形图指令名称	梯形图符号	功能	步数
ATANP (反正切 - 上升沿)		三角函数	3 至 7

■ 操作数设置

下面描述了 ATAN 和 ATANP 指令操作数 S1 和 D1 的可指定内容。

ATAN 和 ATANP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 ATAN 和 ATANP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 [0] = 2 步 } + {Result[N] = 3 步 } + {1 步 } = 6 步

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面描述了 ATAN 和 ATANP 指令操作数 S1 和 D1 的可指定内容。

名称	类型	条件	操作数 步数	可用：O 不可用：X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (不包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量]		X
		指定整型变量 [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型	浮点型变量	1	O
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型	实型变量	1	O
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量 (不能用于 D1)	整型	-2147483648 至 2147483647		X
	浮点型	$\pm 1.175494351\text{e-}38$ 至 $\pm 3.402823466\text{e+}38$	1	O
	实型	$\pm 2.2250738585072014\text{e-}308$ 至 $\pm 1.7976931348623158\text{e+}308$	2	O

■ ATAN 和 ATANP 指令说明

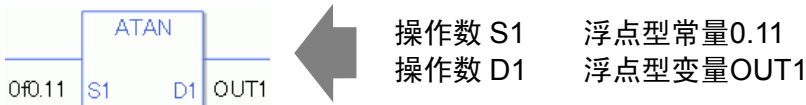
ATAN 和 ATANP 指令是反正切三角函数指令。当 ATAN 指令执行并接通回路时，计算 S1 的反正切，并将结果保存在 D1 中。TAN-1(S1) 被保存在 D1 中。请为 S1 输入一个 -1.0 至 1.0 间的值，D1 中的结果是一个以弧度表示的实数，其值在 -Pi/2 和 Pi/2 之间。Pi 约等于 3.1415926535897(实数)。

ATAN 和 ATANP 指令总是处于接通状态。使用 ATAN 和 ATANP 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。请在操作数 S1 和 D1 中指定相同的变量类型。

指定常量时，请参考如下内容。

当操作数 D1 是浮点型变量时

当输入 Of(0 和小写字母 “f”) 时，将把后面的值视为浮点型数值。



当操作数 D1 是实型变量时

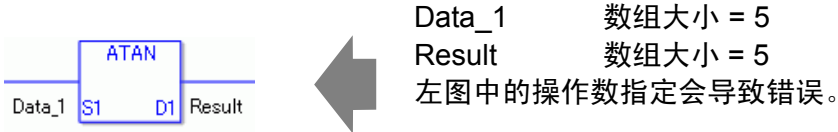
当输入 Or(0 和小写字母 “r”) 时，将把后面的值视为实数值。



当计算指定数组中的数据时

使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。

如果将 S1 和 D1 指定为整个数组，将会发生错误，即使变量类型相同。



■ 显示执行结果的系统变量

当执行结果是 0 时，#L_CalcZero 置 ON。

当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

ATAN



(1)当上升沿指令置 ON 时，将执行 ATAN 指令。ATAN 指令计算 Data_A 的反正切并将结果保存在 D1 中。

(注意) 当使用常开指令时，只要常开指令为 ON，就总是执行 ATAN 指令。

程序示例

ATANP



(1)ATANP 和 ATAN 指令在运行时间上有所不同。对于 ATANP 指令，即使使用常开指令，也只有在检测到上升沿时才会执行 ATANP 指令。因此，即使常开指令位保持为 ON，也只在一次扫描中执行 ATANP 指令。

31.18.7 COT 和 COTP(余切)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
COT (余切 - 电平)		三角函数	3 至 7
梯形图指令名称	梯形图符号	功能	步数
COTP (余切 - 上升沿)		三角函数	3 至 7

■ 操作数设置

下面描述了 COT 和 COTP 指令操作数 S1 和 D1 的可指定内容。
COT 和 COTP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。
操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 COT 和 COTP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 [0] = 2 步 } + {Result[N] = 3 步 } + {1 步 } = 6 步

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面描述了 COT 和 COTP 指令操作数 S1 和 D1 的可指定内容。

名称	类型	条件	操作数 步数	可用：○ 不可用：X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (不包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量]		X
		指定整型变量 [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型	浮点型变量	1	○
		指定浮点型变量 [常量]	2	○
		指定浮点型变量 [变量]	3	○
	实型	实型变量	1	○
		指定实型变量 [常量]	2	○
		指定实型变量 [变量]	3	○
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量 (不能用于 D1)	整型	-2147483648 至 2147483647		X
	浮点型	$\pm 1.175494351\text{e-}38$ 至 $\pm 3.402823466\text{e+}38$	1	O
	实型	$\pm 2.2250738585072014\text{e-}308$ 至 $\pm 1.7976931348623158\text{e+}308$	2	O

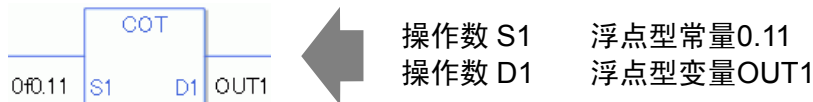
■ COT 和 COTP 指令说明

COT 和 COTP 指令是余切三角函数指令。当 COT 指令执行并接通时，将计算 S1 值的余切，并将结果 $[1/\tan(S1)]$ 保存在 D1 中。在 S1 中输入弧度值。S1 越接近于 Pi 的整数倍，在 D1 中得到的结果的绝对值就越大。该值为 $\pm 2.225\text{e-}308$ 至 $\pm 1.79\text{e+}308$ 范围内的一个实数。

Pi 约等于 3.1415926535897(实数)。COT 和 COTP 指令总是处于接通状态。使用 COT 和 COTP 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。请在操作数 S1 和 D1 中指定相同的变量类型。
指定常量时，请参考如下内容。

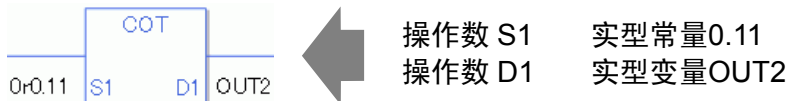
当操作数 D1 是浮点型变量时

当输入 0f(0 和小写字母 “f”) 时，将把后面的值视为浮点型数值。



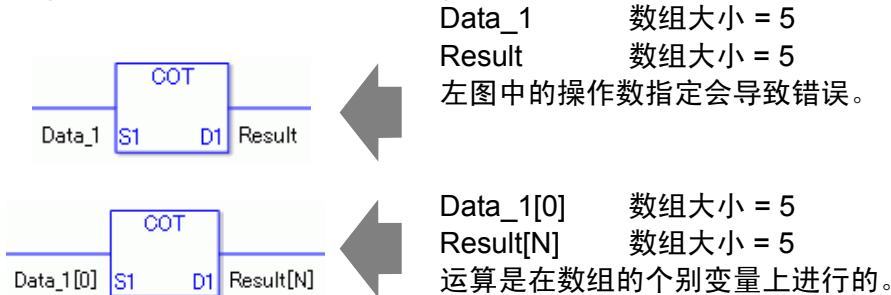
当操作数 D1 是实型变量时

当输入 0r(0 和小写字母 “r”) 时，将把后面的值视为实数值。



当计算指定数组中的数据时

使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。
如果将 S1 和 D1 指定为整个数组，将会发生错误，即使变量类型相同。



■ 显示执行结果的系统变量

当执行结果是 0 时，#L_CalcZero 置 ON。

当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

COT



(1) 当上升沿指令置 ON 时，将执行 COT 指令。COT 指令计算 Data_A 的余切并将结果保存在 D1 中。

当使用常开指令时，只要常开指令位为 ON，就总是执行 COT 指令。

程序示例

COTP



(1) COTP 和 ACOT 指令在运行时间上有所不同。对于 COTP 指令，即使使用常开指令，也只有在检测到上升沿时才会执行 COTP 指令。因此，即使常开指令位保持为 ON，也只在一次扫描中执行 COTP 指令。

31.18.8 EXP 和 EXPP(指数)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
EXP (指数 - 电平)		其他函数	3 至 7
梯形图指令名称	梯形图符号	功能	步数
EXPP (指数 - 上升沿)		其他函数	3 至 7

■ 操作数设置

下面描述了 EXP 和 EXPP 指令操作数 S1 和 D1 的可指定内容。
EXP 和 EXPP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。
操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 EXP 和 EXPP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 [0] = 2 步 } + {Result[N] = 3 步 } + {1 步 } = 6 步

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面描述了 EXP 和 EXPP 指令操作数 S1 和 D1 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (不包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量]		X
		指定整型变量 [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型	浮点型变量	1	O
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型	实型变量	1	O
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量 (不能用于 D1)	整型	-2147483648 至 2147483647		X
	浮点型	$\pm 1.175494351\text{e-}38$ 至 $\pm 3.402823466\text{e+}38$	1	O
	实型	$\pm 2.2250738585072014\text{e-}308$ 至 $\pm 1.7976931348623158\text{e+}308$	2	O

■ EXP 和 EXPP 指令说明

EXP 和 EXPP 指令是指数指令。当 EXP 指令执行时，计算 S1 的 EXP，并将结果保存在 D1 中。

S1 的指数值保存在 D1 中。D1 中以实数形式保存 e 的 S1 次幂。

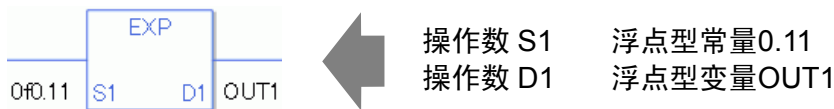
运算表达式：D1 = e^{S1}e 约等于 2.7182818284590(实数)。

EXP 和 EXPP 指令总是处于接通状态。使用 EXP 和 EXPP 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。请在操作数 S1 和 D1 中指定相同的变量类型。

指定常量时，请参考如下内容。

当操作数 D1 是浮点型变量时

当输入 0f(0 和小写字母 “f”) 时，将把后面的值视为浮点型数值。



当操作数 D1 是实型变量时

当输入 0r(0 和小写字母 “r”) 时，将把后面的值视为实数值。



当计算指定数组中的数据时

使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。

如果将 S1 和 D1 指定为整个数组，将会发生错误，即使变量类型相同。



■ 显示执行结果的系统变量

当执行结果是 0 时，#L_CalcZero 置 ON。

当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

EXP



(1)当上升沿指令置 ON 时，将执行 EXP 指令。EXP 指令计算 Data_A 次幂并将结果保存在 D1 中。

当使用常开指令时，只要常开指令位为 ON，就总是执行 EXP 指令。

程序示例

EXPP



(1)EXPP 和 EXP 指令在运行时间上有所不同。对于 EXPP 指令，即使使用常开指令，也只有在检测到上升沿时才会执行 EXPP 指令。因此，即使常开指令位保持为 ON，也只在一次扫描中执行 EXPP 指令。

31.18.9 LN 和 LNP(自然对数)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
LN (自然对数 - 电平)		其他函数	3 至 7
梯形图指令名称	梯形图符号	功能	步数
LNP (自然对数 - 上升沿)		其他函数	3 至 7

■ 操作数设置

下面描述了 LN 和 LNP 指令操作数 S1 和 D1 的可指定内容。

LN 和 LNP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 LN 和 LNP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 [0] = 2 步 } + {Result[N] = 3 步 } + {1 步 } = 6 步

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面描述了 LN 和 LNP 指令操作数 S1 和 D1 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (不包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量]		X
		指定整型变量 [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型	浮点型变量	1	O
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型	实型变量	1	O
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

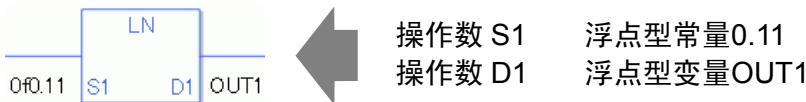
名称	类型	条件	操作数 步数	可用: O 不可用: X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量 (不能用于 D1)	整型	-2147483648 至 2147483647		X
	浮点型	$\pm 1.175494351\text{e-}38$ 至 $\pm 3.402823466\text{e+}38$	1	O
	实型	$\pm 2.2250738585072014\text{e-}308$ 至 $\pm 1.7976931348623158\text{e+}308$	2	O

■ LN 和 LNP 指令说明

LN 和 LNP 指令是对数指令。LN 指令计算 S1 的自然对数并将结果保存在 D1 中。
D1 中的结果为实数，e 的 D1 次幂等于 S1。
运算表达式：D1 = log_e S1 e 约等于 2.7182818284590(实数)。
LN 和 LNP 指令总是处于接通状态。使用 LN 和 LNP 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。请在操作数 S1 和 D1 中指定相同的变量类型。
指定常量时，请参考如下内容。

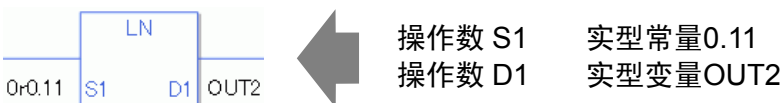
当操作数 D1 是浮点型变量时

当输入 Of(0 和小写字母 “f”) 时，将把后面的值视为浮点型数值。



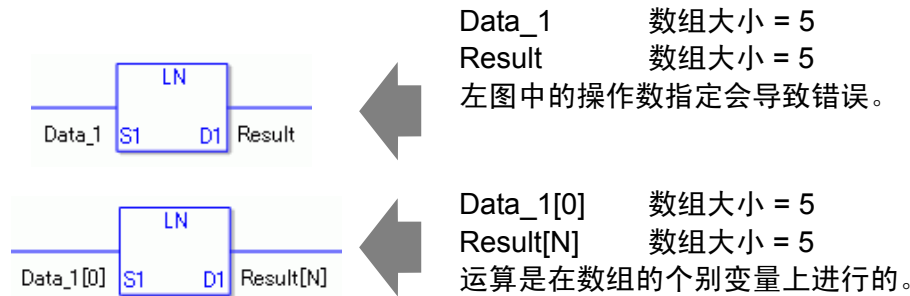
当操作数 D1 是实型变量时

当输入 Or(0 和小写字母 “r”) 时，将把后面的值视为实数值。



当计算指定数组中的数据时

使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。
如果将 S1 和 D1 指定为整个数组，将会发生错误，即使变量类型相同。



■ 显示执行结果的系统变量

当执行结果是 0 时，#L_CalcZero 置 ON。

当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

LN



(1)当上升沿指令置 ON 时，将执行 LN 指令。LN 指令计算 Data_A 的自然对数函数并将结果保存在 D1 中。

当使用常开指令时，只要常开指令位为 ON，就总是执行 LN 指令。

程序示例

LNP



(1)LNP 和 LN 指令在运行时间上有所不同。对于 LNP 指令，即使使用常开指令，也只有在检测到上升沿时才会执行 LNP 指令。因此，即使常开指令位保持为 ON，也只在一次扫描中执行 LNP 指令。

31.18.10 LG10 和 LG10P(常用对数)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
LG10 (常用对数 - 电平)		其他函数	3 至 7
梯形图指令名称	梯形图符号	功能	步数
LG10P (对数 - 上升沿)		其他函数	3 至 7

■ 操作数设置

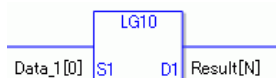
下面描述了 LG10 和 LG10P 指令操作数 S1 和 D1 的可指定内容。

LG 和 LG10P 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 LG10 和 LG10P 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 [0] = 2 步 } + {Result[N] = 3 步 } + {1 步 } = 6 步

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面描述了 LG10 和 LG10P 指令操作数 S1 和 D1 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (不包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量]		X
		指定整型变量 [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型	浮点型变量	1	O
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型	实型变量	1	O
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量 (不能用于 D1)	整型	-2147483648 至 2147483647		X
	浮点型	$\pm 1.175494351\text{e-}38$ 至 $\pm 3.402823466\text{e+}38$	1	O
	实型	$\pm 2.2250738585072014\text{e-}308$ 至 $\pm 1.7976931348623158\text{e+}308$	2	O

■ LG10 和 LG10P 指令说明

LG10 和 LG10P 指令是对数指令。LG10 指令计算 S1 的常用对数并将结果保存在 D1 中。

D1 中保存的是 $\log_{10} S1$ 的值，它是一个实数。

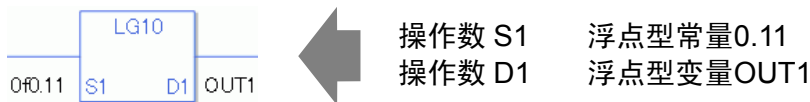
等式： $D1 = \log_{10} S1$

LG10 和 LG10P 指令总是处于接通状态。使用 LG10 和 LG10P 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。请在操作数 S1 和 D1 中指定相同的变量类型。

指定常量时，请参考如下内容。

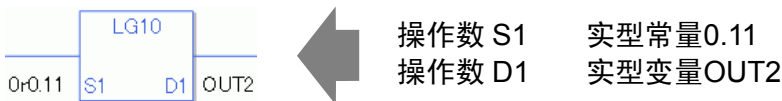
当操作数 D1 是浮点型变量时

当输入 0f(0 和小写字母 “f”) 时，将把后面的值视为浮点型数值。



当操作数 D1 是实型变量时

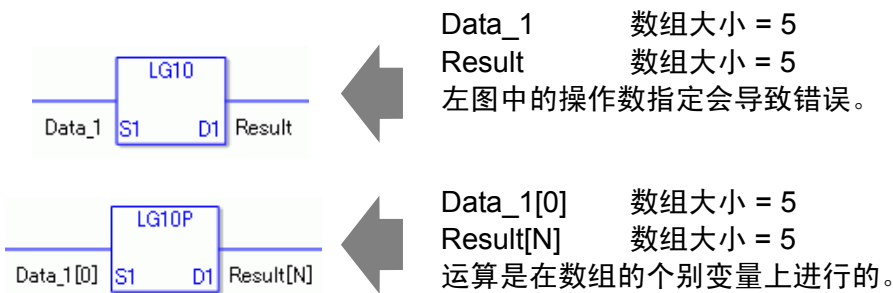
当输入 0r(0 和小写字母 “r”) 时，将把后面的值视为实数值。



当计算指定数组中的数据时

使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。

如果将 S1 和 D1 指定为整个数组，将会发生错误，即使变量类型相同。



■ 显示执行结果的系统变量

当执行结果是 0 时，#L_CalcZero 置 ON。

当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

LG10



(1)当上升沿指令置 ON 时，将执行 LG10 指令。LG10 指令计算 Data_A 的常用对数函数并将结果保存在 D1 中。

当使用常开指令时，只要常开指令位为 ON，就总是执行 LG10 指令。

程序示例

LG10P

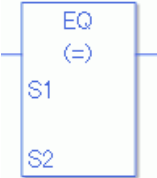


(1)LG10P 和 LG10 指令在运行时间上有所不同。对于 LG10P 指令，即使使用常开指令，也只有在检测到上升沿时才会执行 LG10P 指令。因此，即使常开指令位保持为 ON，也只在一次扫描中执行 LG10P 指令。

31.19 算术比较指令

31.19.1 EQ (=)

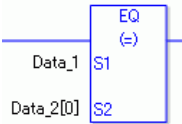
符号和功能

梯形图指令名称	梯形图符号	功能	步数
EQ (= 电平)		比较	3 至 9

■ 操作数设置

下面描述了 EQ 指令操作数 S1 和 S2 的可指定内容。
EQ 指令的实际步数取决于指定的操作数。下面描述如何计算步数。
操作数 S1 的步数 + 操作数 S2 的步数 +1 = 一个指令的总步数

例如，计算 EQ 指令的步数
(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步 } + {Data_2[0]=2 步 } + {1 步 } = 4 步。

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面描述了 EQ 指令操作数 S1 和 S2 的可指定内容。

名称	类型	条件	操作数 步数	可用：○ 不可用：X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	○
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	○
符号	位			X
	字		1	○
变量格式	位	指定一个位		X
		指定位数组 (I 常量)		X
		指定位数组 (I 变量)		X
	整型 (包括 I/O)	未指定数组和修饰符	1	○
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]	2	○
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]	3	○
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]	4	○
	浮点型		1	○
		指定浮点型变量 [常量]	2	○
		指定浮点型变量 [变量]	3	○
	实型		1	○
		指定实型变量 [常量]	2	○
		指定实型变量 [变量]	3	○
	定时器	仅 .PT / .ET	2	○
	计数器	仅 .PV / .CV	2	○
	日期	仅 .YR / .MO / .DAY	2	○
	时间	仅 .HR / .MIN / .SEC	2	○
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	○

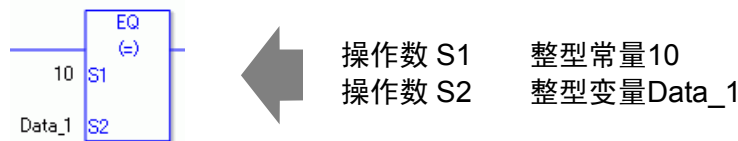
名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]	2	O
		D_****.B/W [地址]	3	O
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647	1	O
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38	1	O
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308	2	O

■ EQ 指令说明

EQ 指令是比较指令。EQ 指令比较 S1 和 S2，如果比较结果是 S1 = S2，那么指令就接通。当比较实值时需要留意。例如，如果操作数的值是 1.99999999999，它就不等于 2.00000000000。

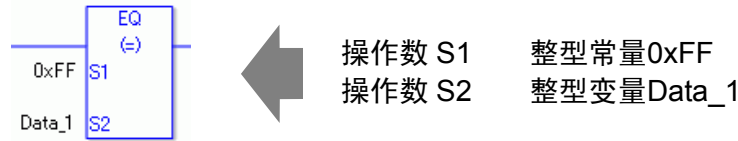
使用 EQ 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。请在操作数 S1 和 S2 中指定相同的变量类型。
指定常量时，请参考如下内容。

当操作数 S1 或 S2 是整型常量时



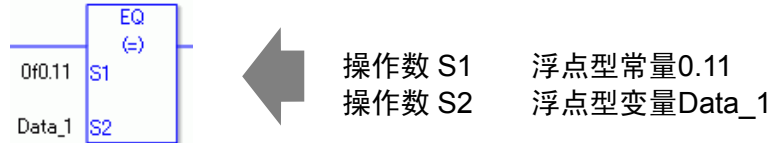
当在操作数 S1 或 S2 中输入十六进制值时

当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



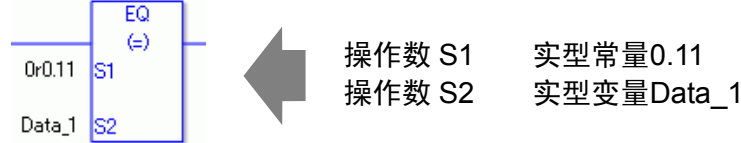
当在操作数 S1 或 S2 中输入浮点型常量时

当输入 0f(0 和小写字母 “f”) 时，将把后面的值视为浮点型数值。



当在操作数 S1 或 S2 中输入实型常量时

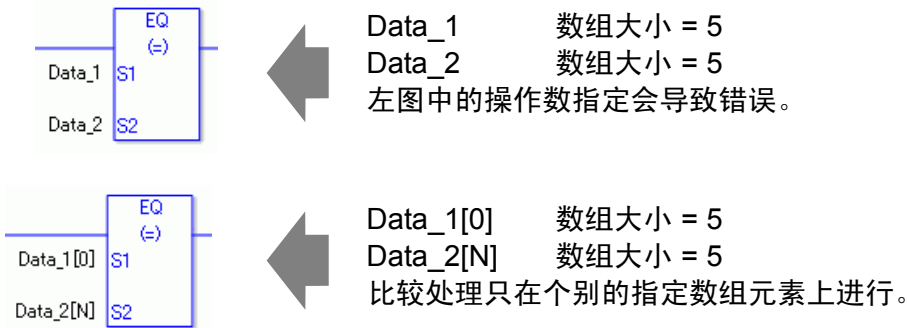
当输入 0r(0 和小写字母 “r”) 时，将把后面的值视为实数值。



当比较指定数组 (整型变量数组) 中的数据时

使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。

不能将操作数 S1 或 S2 指定为整个数组。否则，即使指定的数组变量类型相同，也会发生错误。



程序示例

EQ

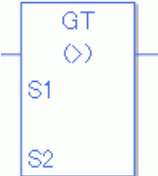
比较整型变量并在 D1 中输出结果。



(1)比较 Data_A 和 Result，确定它们是否相等。如果 EQ 指令的结果是 S1 = S2，那么 EQ 指令就接通，然后就执行 EQ 指令右边的指令。在上图中，右边的指令是 MOV 指令。

31.19.2 GT (>)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
GT (> 电平)		比较	3 至 9

■ 操作数设置

下面描述了 GT 指令操作数 S1 和 S2 的可指定内容。

GT 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 S2 的步数 + 1 = 一个指令的总步数

例如，计算 GT 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {Data_2[0]=2 步} + {1 步} = 4 步。

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面描述了 GT 指令操作数 S1 和 S2 的可指定内容。

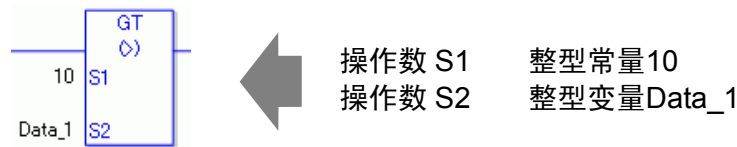
名称	类型	条件	操作数 步数	可用：O 不可用：X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]	2	O
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]	4	O
	浮点型	浮点型变量	1	O
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型	实型变量	1	O
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]	2	O
		D_****.B/W [地址]	3	O
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647	1	O
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38	1	O
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308	2	O

■ GT 指令说明

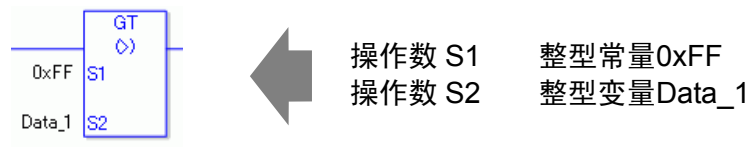
GT 指令是比较指令。GT 指令比较 S1 和 S2。如果比较结果是 $S1 > S2$ ，那么该指令就接通。当比较实值时需要留意。例如，如果操作数的值是 2.000000000001，它仍大于 2。当使用 GT 指令时，如果在操作数 S1 和 S2 中指定的不是同一类型的变量，就会发生错误。请在操作数 S1 和 S2 中指定相同的变量类型。
指定常量时，请参考如下内容。

当操作数 S1 或 S2 是整型常量时



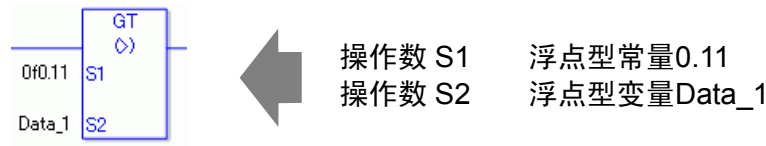
当在操作数 S1 或 S2 中输入十六进制值时

当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



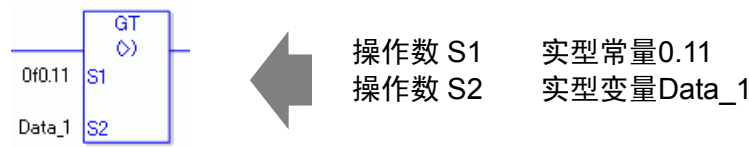
当在操作数 S1 或 S2 中输入浮点型常量时

当输入 0f(0 和小写字母 “f”) 时，将把后面的值视为浮点型数值。

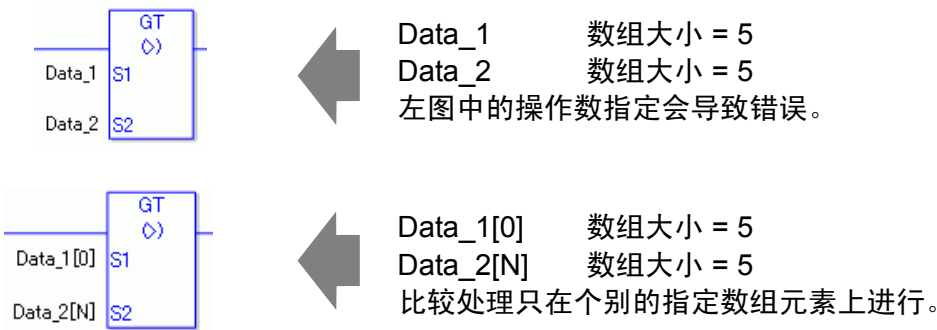


当在操作数 S1 或 S2 中输入实型常量时

当输入 0r(0 和小写字母 “r”) 时，将把后面的值视为实数值。



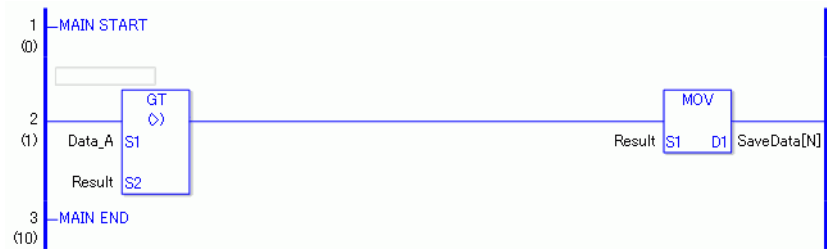
当比较指定数组 (整型变量数组) 中的数据时
使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。
不能将操作数 S1 或 S2 指定为整个数组。否则，即使指定的数组变量类型相同，也会发生错误。



程序示例

GT

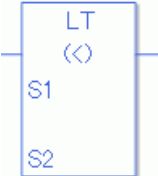
比较整型变量并在 D1 中输出结果。



(1)比较 Data_A 和 Result，确定 Data_A 是否大于 Result。如果 GT 指令的结果是 S1>S2，那么 GT 指令就接通。然后就会执行 GT 指令右边的指令。在上图中，右边的指令是 MOV 指令。

31.19.3 LT(<)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
LT (< 电平)		比较	3 至 9

■ 操作数设置

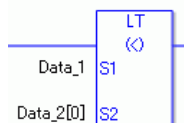
下面描述了 LT 指令操作数 S1 和 S2 的可指定内容。

LT 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 S2 的步数 + 1 = 一个指令的总步数

例如，计算 LT 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {Data_2[0]=2 步} + {1 步} = 4 步。

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面描述了 LT 指令操作数 S1 和 S2 的可指定内容。

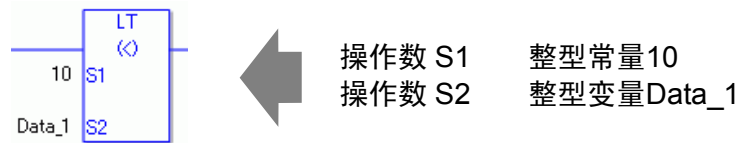
名称	类型	条件	操作数 步数	可用：O 不可用：X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]	2	O
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]	4	O
	浮点型	浮点型变量	1	O
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型	实型变量	1	O
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]	2	O
		D_****.B/W [地址]	3	O
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647	1	O
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38	1	O
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308	2	O

■ LT 指令说明

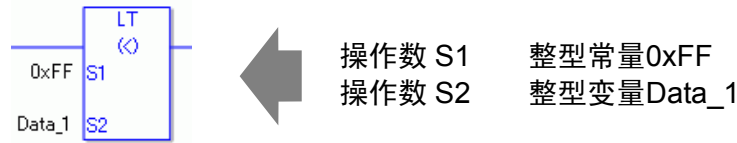
LT 指令是比较指令。LT 指令比较 S1 和 S2。如果比较结果是 $S1 < S2$ ，那么该指令就接通。当比较实值时需要留意。例如，如果操作数的值是 1.9999999999，它仍小于 2。当使用 LT 指令时，如果在操作数 S1 和 S2 中指定的不是同一类型的变量，就会发生错误。请在操作数 S1 和 S2 中指定相同的变量类型。指定常量时，请参考如下内容。

当操作数 S1 或 S2 是整型常量时



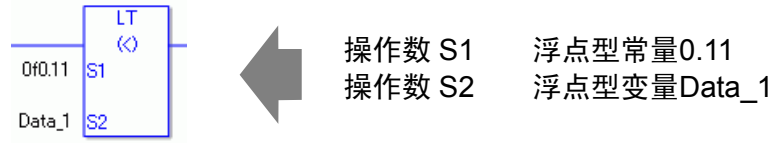
当在操作数 S1 或 S2 中输入十六进制值时

当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



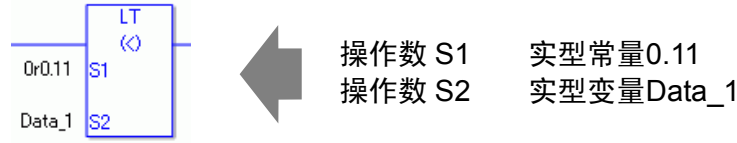
当在操作数 S1 或 S2 中输入浮点型常量时

当输入 0f(0 和小写字母 “f”) 时，将把后面的值视为浮点型数值。



当在操作数 S1 或 S2 中输入实型常量时

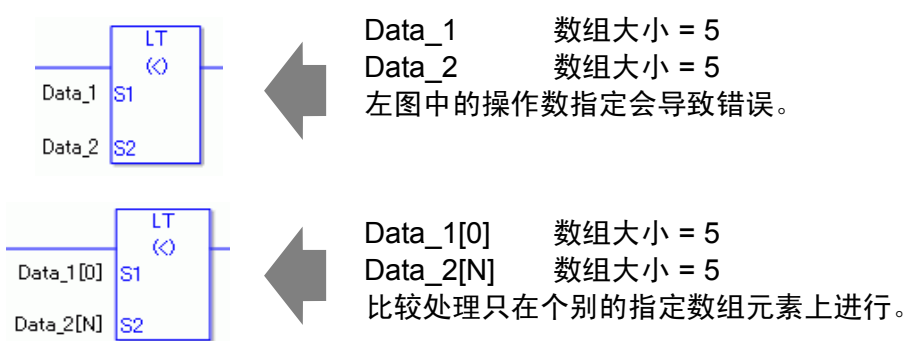
当输入 0r(0 和小写字母 “r”) 时，将把后面的值视为实数值。



当比较指定数组 (整型变量数组) 中的数据时

使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。

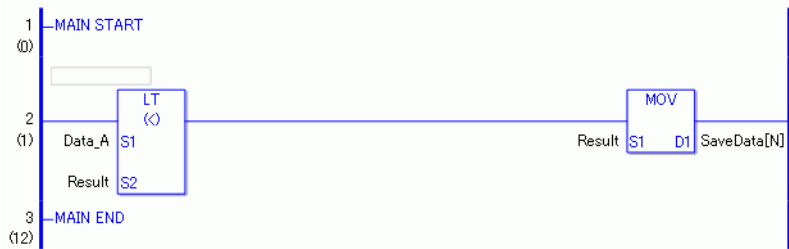
不能将操作数 S1 或 S2 指定为整个数组。否则，即使指定的数组变量类型相同，也会发生错误。



程序示例

LT

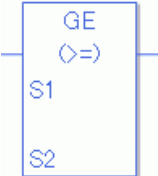
比较整型变量并在 D1 中输出结果。



(1)比较 Data_A 和 Result，确定 Data_A 是否小于 Result。如果 LT 指令的结果是 S1<S2，那么 LT 指令就接通。然后就执行 LT 指令右边的指令。在上图中，右边的指令是 MOV 指令。

31.19.4 GE (>=)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
GE (>= 电平)		比较	3 至 9

■ 操作数设置

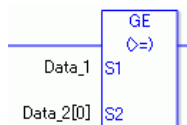
下面描述了 GE 指令操作数 S1 和 S2 的可指定内容。

GE 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 S2 的步数 + 1 = 一个指令的总步数

例如，计算 GE 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {Data_2[0]=2 步} + {1 步} = 4 步。

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面描述了 GE 指令操作数 S1 和 S2 的可指定内容。

名称	类型	条件	操作数 步数	可用：○ 不可用：×
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	○
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	○
符号	位			X
	字		1	○
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	○
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]	2	○
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]	3	○
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]	4	○
	浮点型	浮点型变量	1	○
		指定浮点型变量 [常量]	2	○
		指定浮点型变量 [变量]	3	○
	实型	实型变量	1	○
		指定实型变量 [常量]	2	○
		指定实型变量 [变量]	3	○
	定时器	仅 .PT / .ET	2	○
	计数器	仅 .PV / .CV	2	○
	日期	仅 .YR / .MO / .DAY	2	○
	时间	仅 .HR / .MIN / .SEC	2	○
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	○

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]	2	O
		D_****.B/W [地址]	3	O
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647	1	O
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38	1	O
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308	2	O

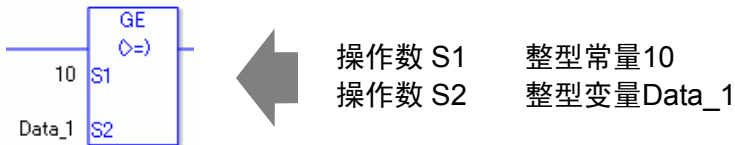
■ GE 指令说明

GE 指令是比较指令。GE 指令比较 S1 和 S2。如果比较结果是 $S1 \geq S2$ ，那么该指令就接通。

当比较实值时需要留意。例如，如果操作数的值是 1.99999999999，它就不大于 2。当使用 GE 指令时，如果在操作数 S1 和 S2 中指定的不是同一类型的变量，就会发生错误。请在操作数 S1 和 S2 中指定相同的变量类型。

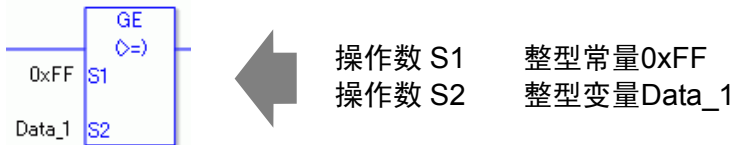
指定常量时，请参考如下内容。

当操作数 S1 或 S2 是整型常量时



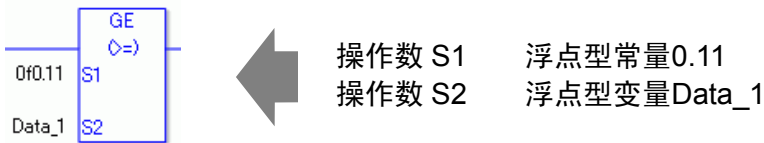
当在操作数 S1 或 S2 中输入十六进制值时

当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



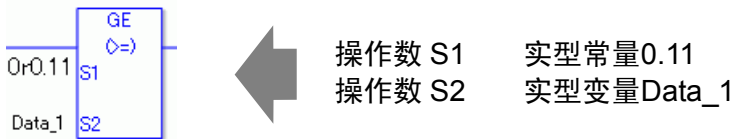
当在操作数 S1 或 S2 中输入浮点型常量时

当输入 0f(0 和小写字母 “f”) 时，将把后面的值视为浮点型数值。



当在操作数 S1 或 S2 中输入实型常量时

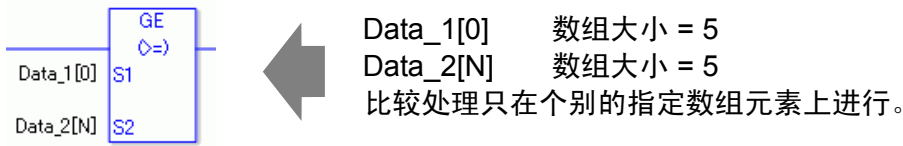
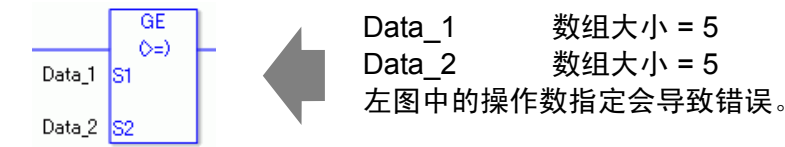
当输入 0r(0 和小写字母 “r”) 时，将把后面的值视为实数值。



当比较指定数组 (整型变量数组) 中的数据时

使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。

不能将操作数 S1 或 S2 指定为整个数组。否则，即使指定的数组变量类型相同，也会发生错误。



程序示例

GE

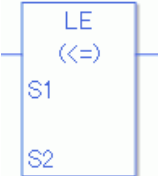
比较整型变量并在 D1 中输出结果。



(1)比较 Data_A 和 Result，确定 Data_A 是否大于等于 Result。如果 GE 指令的结果是 S1>= S2，那么 GE 指令就接通。然后就会执行 GE 指令右边的指令。在上图中，右边的指令是 MOV 指令。

31.19.5 LE (<=)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
LE (<= 电平)		比较	3 至 9

■ 操作数设置

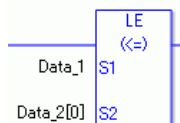
下面描述了 LE 指令操作数 S1 和 S2 的可指定内容。

LE 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 S2 的步数 + 1 = 一个指令的总步数

例如，计算 LE 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {Data_2[0]=2 步} + {1 步} = 4 步。

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面描述了 LE 指令操作数 S1 和 S2 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]	2	O
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]	4	O
	浮点型	浮点型变量	1	O
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型	实型变量	1	O
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]	2	O
		D_****.B/W [地址]	3	O
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647	1	O
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38	1	O
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308	2	O

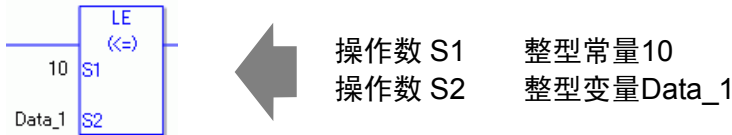
■ LE 指令的解释

LE 指令是比较指令。LE 指令比较 S1 和 S2。

当比较实值时需要留意。当比较实值时需要留意。例如，如果操作数的值是 2.000000000001，它不小于等于 2。当使用 LE 指令时，如果在操作数 S1 和 S2 中指定的不是同一类型的变量，就会发生错误。请在操作数 S1 和 S2 中指定相同的变量类型。

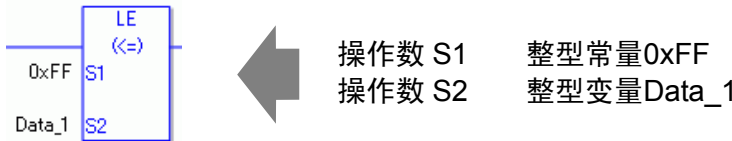
指定常量时，请参考如下内容。

当操作数 S1 或 S2 是整型常量时



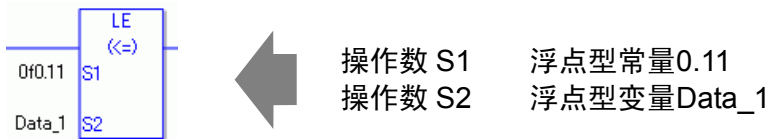
当在操作数 S1 或 S2 中输入十六进制值时

当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



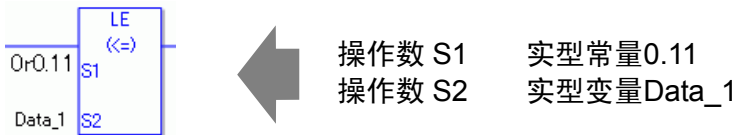
当在操作数 S1 或 S2 中输入浮点型常量时

当输入 0f(0 和小写字母 “f”) 时，将把后面的值视为浮点型数值。



当在操作数 S1 或 S2 中输入实型常量时

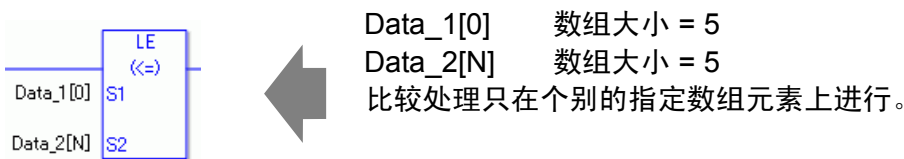
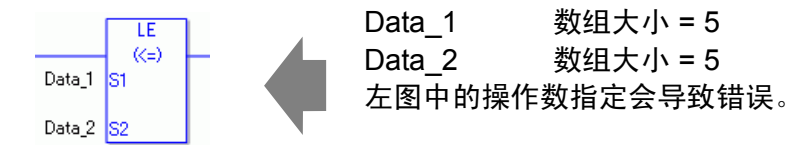
当输入 0r(0 和小写字母 “r”) 时，将把后面的值视为实数值。



当比较指定数组 (整型变量数组) 中的数据时

使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。

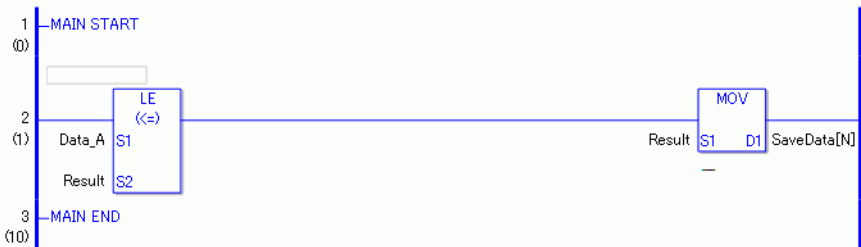
不能将操作数 S1 或 S2 指定为整个数组。否则，即使指定的数组变量类型相同，也会发生错误。



程序示例

LE

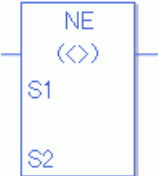
比较整型变量并在 D1 中输出结果。



(1)比较 Data_A 和 Result，看看 Data_A 是否小于或等于 Result。如果 LE 指令的结果是 S1<=S2，那么 LE 就接通。然后就执行 LE 指令右边的指令。在上图中，右边的指令是 MOV 指令。

31.19.6 NE (<>)

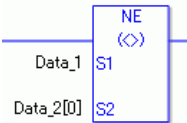
符号和功能

梯形图指令名称	梯形图符号	功能	步数
NE (<> 电平)		比较	3 至 9

■ 操作数设置

下面描述了 NE 指令操作数 S1 和 S2 的可指定内容。
NE 指令的实际步数取决于指定的操作数。下面描述如何计算步数。
操作数 S1 的步数 + 操作数 S2 的步数 +1 = 一个指令的总步数

例如，计算 NE 指令的步数
(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {Data_2[0]=2 步} + {1 步} = 4 步。

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面描述了 NE 指令操作数 S1 和 S2 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]	2	O
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]	4	O
	浮点型	浮点型变量	1	O
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型	实型变量	1	O
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用: O 不可用: X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]	2	O
		D_****.B/W [地址]	3	O
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647	1	O
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38	1	O
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308	2	O

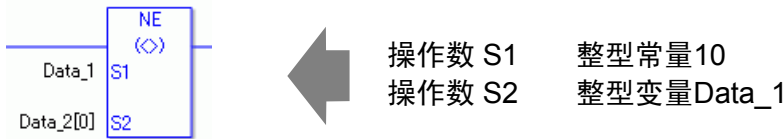
■ NE 指令说明

NE 指令是比较指令。NE 指令比较 S1 和 S2。如果比较结果是 S1<>S2，那么该指令就接通。

当比较实值时需要留意。例如，如果操作数的值是 2.000000000001，它就不等于 2。当使用 NE 指令时，如果在操作数 S1 和 S2 中指定的不是同一类型的变量，就会发生错误。请在操作数 S1 和 S2 中指定相同的变量类型。

指定常量时，请参考如下内容。

当操作数 S1 或 S2 是整型常量时



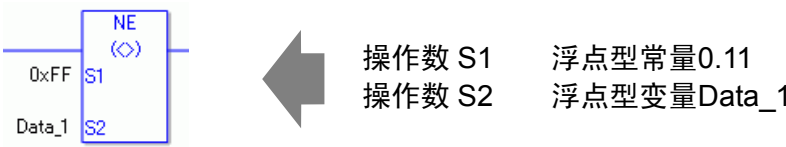
当在操作数 S1 或 S2 中输入十六进制值时

当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



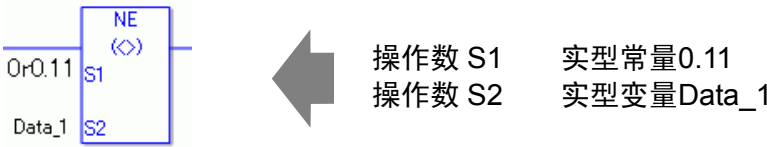
当在操作数 S1 或 S2 中输入浮点型常量时

当输入 0f(0 和小写字母 “f”) 时，将把后面的值视为浮点型数值。

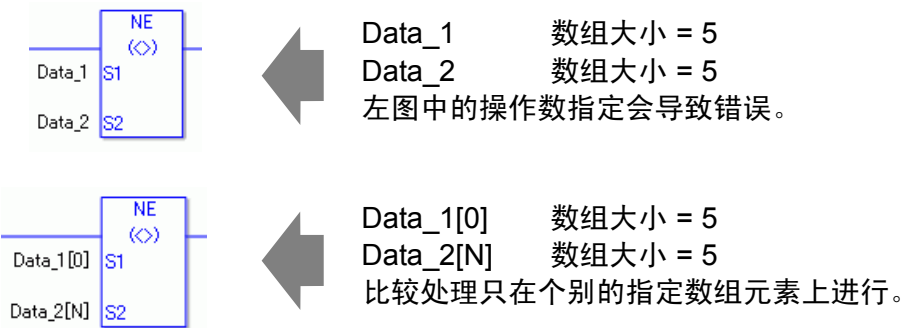


当在操作数 S1 或 S2 中输入实型常量时

当输入 0r(0 和小写字母 “r”) 时，将把后面的值视为实数值。



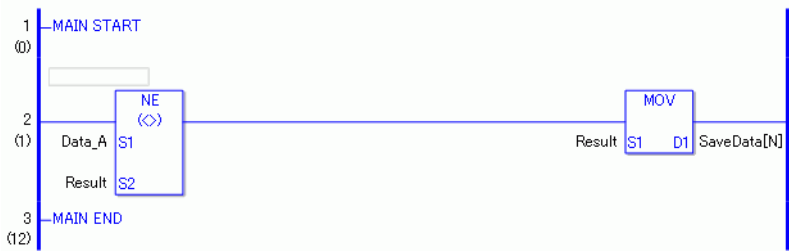
当比较指定数组 (整型变量数组) 中的数据时
使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。
不能将操作数 S1 或 S2 指定为整个数组。否则，即使指定的数组变量类型相同，也会发生错误。



程序示例

NE

比较整型变量并在 D1 中输出结果。

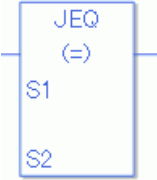


(1)比较 Data_A 和 Result，确定 Data_A 是否不等于 Result。如果 NE 指令的结果是 S1<>S2，那么 NE 指令就接通。然后执行 NE 指令右边的指令。在上图中，右边的指令是 MOV 指令。

31.20 时间比较指令

31.20.1 JEQ (=)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
JEQ (= 电平)		时间比较	3

■ 操作数设置

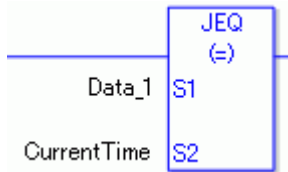
下面给出了 JEQ 指令操作数 (S1, S2) 的可配置条件。

JEQ 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 S2 的步数 + 1 = 一个指令的总步数

例如，计算 JEQ 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {CurrentTime=1 步} + {1 步} = 3 步。

指令中包括一个最终步骤。必须加上这一步。

■ JEQ 指令说明

JEQ 指令中的时间变量是结构变量。下表列出内部结构。

时间变量

时间变量	变量设置	描述
变量名 .HR	整型变量	以 BCD 形式输入小时。
变量名 .MIN	整型变量	以 BCD 形式输入分钟。
变量名 .SEC	整型变量	以 BCD 形式输入秒。

■ 操作数设置

下面描述了 JEQ 指令操作数 (S1, S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]		X
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型	浮点型变量		X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型	实型变量		X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	.HR / .MIN / .SEC 未指定结构元素。	1	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	.HR / .MIN / .SEC 未指定结构元素。	1	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ JEQ 指令说明

JEQ 指令比较时间。执行 JEQ 指令时，将比较 S1 和 S2。如果结果是 $S1=S2$ ，该指令就接通。

同时比较小时、分钟和时间变量。要比较 10:20，请在秒钟上输入 0。

当使用 JEQ 指令时，您只能在操作数 S1 和 S2 中指定时间变量。

程序示例

JEQ

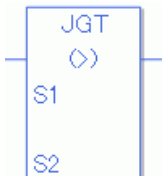
比较时间变量，用线圈来表示结果。



(1)比较 Data_1 和 CurrentTime，确定它们是否相等。如果结果是 $S1 = S2$ ，那么该指令就接通，随即执行 JEQ 指令右边的指令。在上图中，执行的是 JEQ 指令右边的 OUT 指令。

31.20.2 JGT (>)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
JGT (> 电平)		时间比较	3

■ 操作数设置

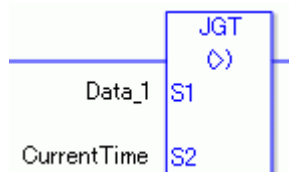
下面给出 JGT 指令操作数 (S1, S2) 的可配置条件。

JGT 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 S2 的步数 + 1 = 一个指令的总步数

例如，计算 JGT 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {CurrentTime=1 步} + {1 步} = 3 步。

指令中包括一个最终步骤。必须加上这一步。

■ JGT 指令说明

JGT 指令中的时间变量是结构变量。下表列出内部结构。

时间变量

时间变量	变量设置	描述
变量名 .HR	整型变量	以 BCD 形式输入小时。
变量名 .MIN	整型变量	以 BCD 形式输入分钟。
变量名 .SEC	整型变量	以 BCD 形式输入秒。

■ 操作数设置

下面给出 JGT 指令操作数 (S1, S2) 的可配置条件。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]		X
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	.HR / .MIN / .SEC 未指定结构元素。	1	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	.HR / .MIN / .SEC 未指定结构元素。	1	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ JGT 指令说明

JGT 指令比较时间。执行 JGT 指令时，将比较 S1 和 S2。如果结果是 $S1 > S2$ ，该指令就接通。

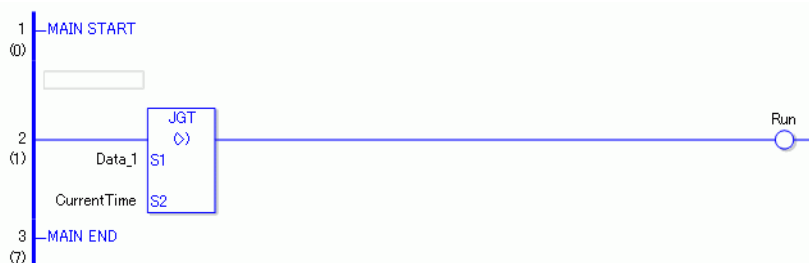
同时比较小时、分钟和时间变量。要比较 10:20，请在秒钟上输入 0。

使用 JGT 指令时，您只能在操作数 S1 和 S2 中指定时间变量。

程序示例

JGT

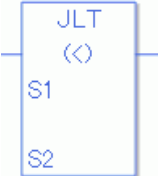
比较时间变量，用线圈来表示结果。



(1)比较 Data_1 和 CurrentTime，确定 Data_1 是否更大。如果结果是 $S1 > S2$ ，那么该指令就接通，随即执行 JGT 指令右边的指令。在上图中，执行的是 JGT 指令右边的 OUT 指令。

31.20.3 JLT (<)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
JLT (< 电平)		时间比较	3

■ 操作数设置

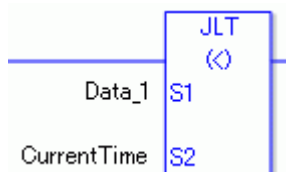
下面给出了 JLT 指令操作数 (S1, S2) 的可配置条件。

JLT 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 S2 的步数 + 1 = 一个指令的总步数

例如，计算 JLT 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {CurrentTime=1 步} + {1 步} = 3 步。

指令中包括一个最终步骤。必须加上这一步。

■ JLT 指令说明

JLT 指令中的时间变量是结构变量。下表列出内部结构。

时间变量

时间变量	变量设置	描述
变量名 .HR	整型变量	以 BCD 形式输入小时。
变量名 .MIN	整型变量	以 BCD 形式输入分钟。
变量名 .SEC	整型变量	以 BCD 形式输入秒。

■ 操作数设置

下面描述了 JLT 指令操作数 (S1, S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: ○ 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]		X
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	.HR / .MIN / .SEC 未指定结构元素。	1	○
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	.HR / .MIN / .SEC 未指定结构元素。	1	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ JLT 指令说明

JLT 指令比较时间。当执行 JLT 指令时，将比较 S1 和 S2。如果结果是 $S1 < S2$ ，该指令就接通。

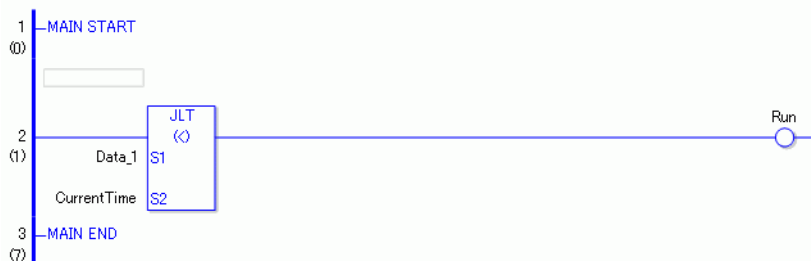
同时比较小时、分钟和时间变量。要比较 10:20，请在秒钟上输入 0。

使用 JLT 指令时，您只能在操作数 S1 和 S2 中指定时间变量。

程序示例

JLT

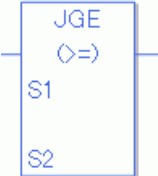
比较时间变量，用线圈来表示结果。



(1) 比较 Data_1 和 CurrentTime，确定 Data_1 是否更小。如果结果是 $S1 < S2$ ，那么该指令就接通，随即执行 JLT 指令右边的指令。在上图中，执行的是 JLT 指令右边的 OUT 指令。

31.20.4 JGE (>=)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
JGE (>= 电平)		时间比较	3

■ 操作数设置

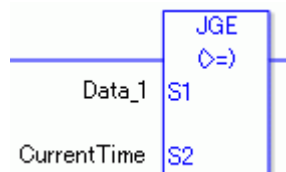
下面给出了 JGE 指令操作数 (S1, S2) 的可配置条件。

JGE 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 S2 的步数 + 1 = 一个指令的总步数

例如，计算 JGE 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {CurrentTime=1 步} + {1 步} = 3 步。

指令中包括一个最终步骤。必须加上这一步。

■ JGE 指令说明

JGE 指令中的时间变量是结构变量。下表列出内部结构。

时间变量

时间变量	变量设置	描述
变量名 .HR	整型变量	以 BCD 形式输入小时。
变量名 .MIN	整型变量	以 BCD 形式输入分钟。
变量名 .SEC	整型变量	以 BCD 形式输入秒。

■ 操作数设置

下面描述了 JGE 指令操作数 (S1, S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]		X
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	.HR / .MIN / .SEC 未指定结构元素。	1	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	.HR / .MIN / .SEC 未指定结构元素。	1	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量	整型	-2147483648 至 2147483647		X
	浮点型	$\pm 1.175494351\text{e-}38$ 至 $\pm 3.402823466\text{e}+38$		X
	实型	$\pm 2.2250738585072014\text{e-}308$ 至 $\pm 1.7976931348623158\text{e}+308$		X

■ JGE 指令说明

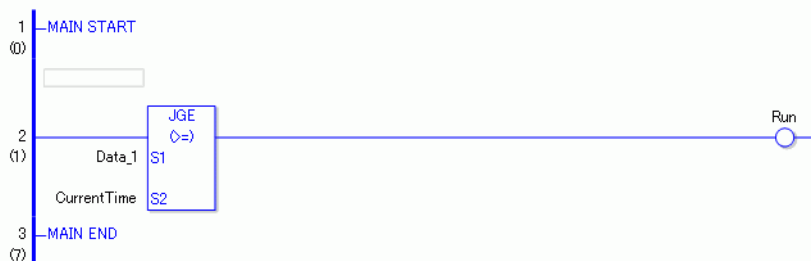
JGE 指令比较时间。执行 JGE 指令时，将比较 S1 和 S2。如果结果是 $S1 \geq S2$ ，那么该指令就接通。同时比较小时、分钟和时间变量。要比较 10:20，请在秒钟上输入 0。

使用 JGE 指令时，您只能在操作数 S1 和 S2 中指定时间变量。

程序示例

JGE

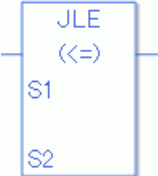
比较时间变量，用线圈来表示结果。



(1)比较 Data_1 和 CurrentTime，看看 Data_1 是否大于等于 CurrentTime。如果结果是 $S1 \geq S2$ ，那么该指令就接通，随即执行 JGE 指令右边的指令。在上图中，执行的是 JGE 指令右边的 OUT 指令。

31.20.5 JLE (<=)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
JLE (<= 电平)		时间比较	3

■ 操作数设置

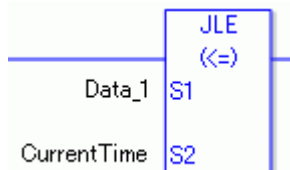
下面给出了 JLE 指令操作数 (S1, S2) 的可配置条件。

JLE 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 S2 的步数 + 1 = 一个指令的总步数

例如，计算 JLE 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {CurrentTime=1 步} + {1 步} = 3 步。

指令中包括一个最终步骤。必须加上这一步。

■ JLE 指令说明

JLE 指令中的时间变量是结构变量。下表列出内部结构。

时间变量

时间变量	变量设置	描述
变量名 .HR	整型变量	以 BCD 形式输入小时。
变量名 .MIN	整型变量	以 BCD 形式输入分钟。
变量名 .SEC	整型变量	以 BCD 形式输入秒。

■ 操作数设置

下面描述了 JLE 指令操作数 (S1, S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]		X
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	.HR / .MIN / .SEC 未指定结构元素。	1	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	.HR / .MIN / .SEC 未指定结构元素。	1	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ JLE 指令说明

JLE 指令比较时间。执行 JLE 指令时，将比较 S1 和 S2。如果结果是 $S1 \leq S2$ ，那么该指令就接通。同时比较小时、分钟和时间变量。要比较 10:20，请在秒钟上输入 0。

使用 JLE 指令时，您只能在操作数 S1 和 S2 中指定时间变量。

程序示例

JLE

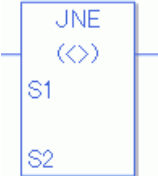
比较时间变量，用线圈来表示结果。



(1) 比较 Data_1 和 CurrentTime，确定 Data_1 是否小于等于 CurrentTime。如果结果是 $S1 \leq S2$ ，那么该指令就接通，随即执行 JLE 指令右边的指令。在上图中，执行的是 JLE 指令右边的 OUT 指令。

31.20.6 JNE (<>)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
JNE (<> 电平)		时间比较	3

■ 操作数设置

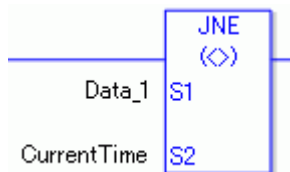
下面给出了 JNE 指令操作数 (S1, S2) 的可配置条件。

JNE 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 S2 的步数 + 1 = 一个指令的总步数

例如，计算 JNE 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {CurrentTime=1 步} + {1 步} = 3 步。

指令中包括一个最终步骤。必须加上这一步。

■ JNE 指令说明

JNE 指令中的时间变量是结构变量。下表列出内部结构。

时间变量

时间变量	变量设置	描述
变量名 .HR	整型变量	以 BCD 形式输入小时。
变量名 .MIN	整型变量	以 BCD 形式输入分钟。
变量名 .SEC	整型变量	以 BCD 形式输入秒。

■ 操作数设置

下面描述了 JNE 指令操作数 (S1, S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: ○ 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]		X
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	.HR / .MIN / .SEC 未指定结构元素。	1	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	.HR / .MIN / .SEC 未指定结构元素。	1	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ JNE 指令说明

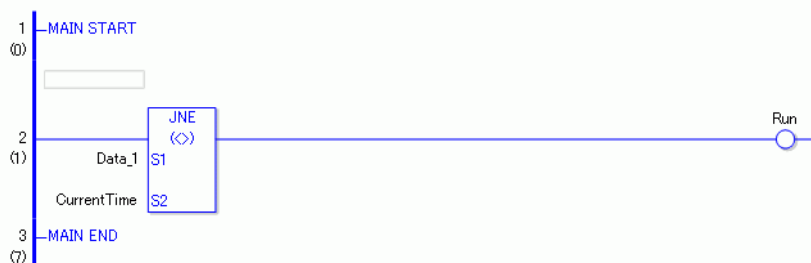
JNE 指令比较时间。执行 JNE 指令时，将比较 S1 和 S2。如果结果是 $S1 \neq S2$ ，那么该指令就接通。同时比较小时、分钟和时间变量。要比较 10:20，请在秒钟上输入 0。

使用 JNE 指令时，您只能在操作数 S1 和 S2 中指定时间变量。

程序示例

JNE

比较时间变量，用线圈来表示结果。

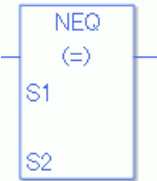


(1)比较 Data_1 和 CurrentTime，确定它们是否不相等。如果结果是 $S1 \neq S2$ ，那么该指令就接通，随即执行 JNE 指令右边的指令。在上图中，执行的是 JNE 指令右边的 OUT 指令。

31.21 日期比较指令

31.21.1 NEQ (=)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
NEQ (= 电平)		日期比较	3

■ 操作数设置

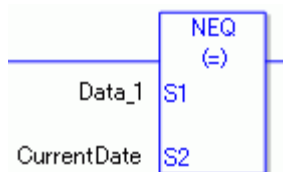
下面给出了 NEQ 指令操作数 (S1, S2) 的可配置条件。

NEQ 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 S2 的步数 + 1 = 一个指令的总步数

例如，计算 NEQ 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {CurrentTime=1 步} + {1 步} = 3 步。

指令中包括一个最终步骤。必须加上这一步。

■ NEQ 指令说明

NEQ 指令中的日期变量是结构变量。下表列出内部结构。

日期变量

日期变量	变量设置	描述
变量名 .YR	整型变量	以 BCD 形式输入年份。
变量名 .MO	整型变量	以 BCD 形式输入月份
变量名 .DAY	整型变量	以 BCD 形式输入日期

■ 操作数设置

下面描述了 NEQ 指令操作数 (S1, S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]		X
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	.YR/ .MO/ .DAY 未指定结构元素。	1	O
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	.YR/ .MO/ .DAY 未指定结构元素。	1	O
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ NEQ 指令说明

NEQ 指令比较日期。执行 NEQ 指令时，将比较 S1 和 S2。如果结果是 S1= S2，那么该指令就接通。同时比较年、月和日变量。使用 NEQ 指令时，您只能在操作数 S1 和 S2 中指定日期变量。

程序示例

NEQ

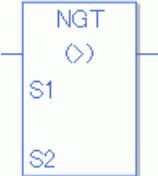
比较日期变量，用线圈来表示结果。



(1)比较 Data_1 和 CurrentDate，确定它们是否相等。如果结果是 S1 = S2，那么该指令就接通，随即执行 NEQ 指令右边的指令。在上图中，执行的是 NEQ 指令右边的 OUT 指令。

31.21.2 NGT (>)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
NGT (> 电平)		日期比较	3

■ 操作数设置

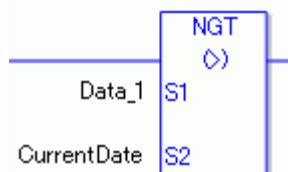
下面给出了 NGT 指令操作数 (S1, S2) 的可配置条件。

NGT 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 S2 的步数 + 1 = 一个指令的总步数

例如，计算 NGT 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {CurrentDate=1 步} + {1 步} = 3 步。

指令中包括一个最终步骤。必须加上这一步。

■ NGT 指令说明

NGT 指令中的日期变量是结构变量。下表列出内部结构。

日期变量

日期变量	变量设置	描述
变量名 .YR	整型变量	以 BCD 形式输入年份。
变量名 .MO	整型变量	以 BCD 形式输入月份。
变量名 .DAY	整型变量	以 BCD 形式输入日期。

■ 操作数设置

下面描述了 NGT 指令操作数 (S1, S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]		X
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	.YR/ .MO/ .DAY 未指定结构元素。	1	O
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	.YR/ .MO/ .DAY 未指定结构元素。	1	O
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

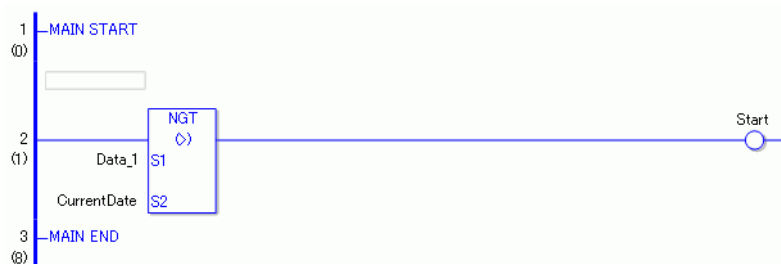
■ NGT 指令说明

NGT 指令比较日期。执行 NGT 指令时，将比较 S1 和 S2。如果结果是 $S1 > S2$ ，那么该指令就接通。同时比较年、月和日变量。使用 NGT 指令时，您只能在操作数 S1 和 S2 中指定日期变量。

程序示例

NGT

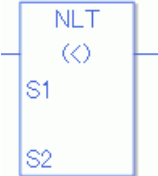
比较日期变量，用线圈来表示结果。



(1)比较 Data_1 和 CurrentDate，确定 Data_1 是否更大。如果结果是 $S1 > S2$ ，那么该指令就接通，随即执行 NGT 指令右边的指令。在上图中，执行的是 NGT 指令右边的 OUT 指令。

31.21.3 NLT (<)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
NLT (< 电平)		日期比较	3

■ 操作数设置

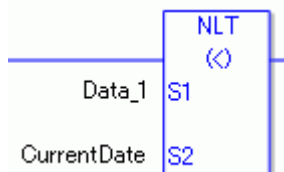
下面给出了 NLT 指令操作数 (S1, S2) 的可配置条件。

NLT 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 S2 的步数 + 1 = 一个指令的总步数

例如，计算 NLT 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {CurrentDate=1 步} + {1 步} = 3 步。

指令中包括一个最终步骤。必须加上这一步。

■ NLT 指令说明

NLT 指令中的日期变量是结构变量。下表列出内部结构。

日期变量

日期变量	变量设置	描述
变量名 .YR	整型变量	以 BCD 形式输入年份。
变量名 .MO	整型变量	以 BCD 形式输入月份
变量名 .DAY	整型变量	以 BCD 形式输入日期

■ 操作数设置

下面描述了 NLT 指令操作数 (S1, S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]		X
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	.YR/ .MO/ .DAY 未指定结构元素。	1	O
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	.YR/ .MO/ .DAY 未指定结构元素。	1	O
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

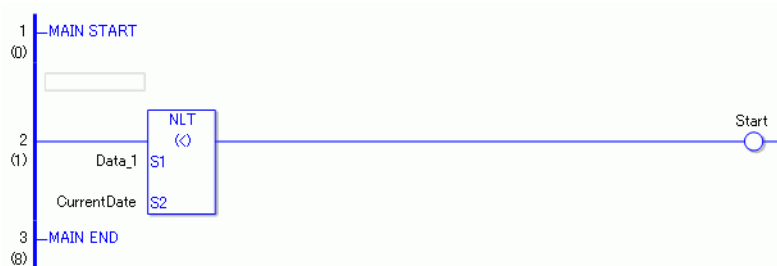
■ NLT 指令说明

NLT 指令比较日期。执行 NLT 指令时，将比较 S1 和 S2。如果结果是 $S1 < S2$ ，那么该指令就接通。同时比较年、月和日变量。使用 NLT 指令时，您只能在操作数 S1 和 S2 中指定日期变量。

程序示例

NLT

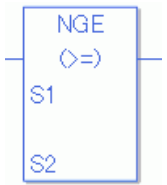
比较日期变量，用线圈来表示结果。



(1)比较 Data_1 和 CurrentDate，确定 Data_1 是否更小。如果结果是 $S1 < S2$ ，那么该指令就接通，随即执行 NLT 指令右边的指令。在上图中，执行的是 NLT 指令右边的 OUT 指令。

31.21.4 NGE (>=)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
NGE (>= 电平)		日期比较	3

■ 操作数设置

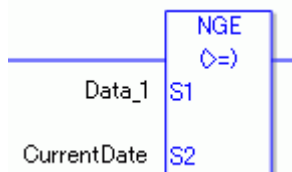
下面给出了 NGE 指令操作数 (S1, S2) 的可配置条件。

NGE 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 S2 的步数 + 1 = 一个指令的总步数

例如，计算 NGE 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {CurrentDate=1 步} + {1 步} = 3 步。

指令中包括一个最终步骤。必须加上这一步。

■ NGE 指令说明

NGE 指令中的日期变量是结构变量。下表列出内部结构。

日期变量

日期变量	变量设置	描述
变量名 .YR	整型变量	以 BCD 形式输入年份。
变量名 .MO	整型变量	以 BCD 形式输入月份。
变量名 .DAY	整型变量	以 BCD 形式输入日期。

■ 操作数设置

下面描述了 NGE 指令操作数 (S1, S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]		X
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	.YR / .MO / .DAY 未指定结构元素。	1	O
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	.YR/ .MO/ .DAY 未指定结构元素。	1	O
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

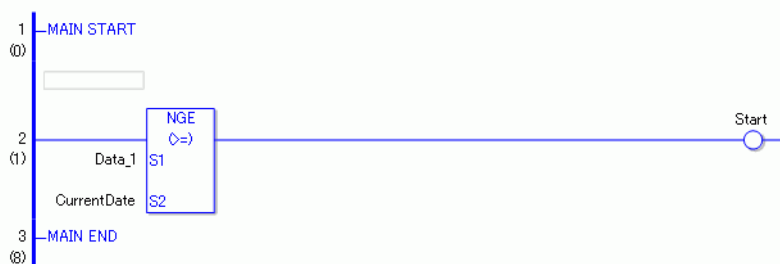
■ NGE 指令说明

NGE 指令比较日期。执行 NGE 指令时，将比较 S1 和 S2。如果结果是 $S1 \geq S2$ ，那么该指令就接通。同时比较年、月和日变量。使用 NGE 指令时，您只能在操作数 S1 和 S2 中指定日期变量。

程序示例

NGE

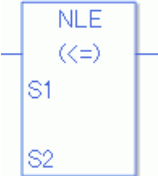
比较日期变量，用线圈来表示结果。



(1)比较 Data_1 和 CurrentDate，确定 Data_1 是否大于等于 CurrentDate。如果结果是 $S1 \geq S2$ ，那么该指令就接通，随即执行 NGE 指令右边的指令。在上图中，执行的是 NGE 指令右边的 OUT 指令。

31.21.5 NLE (<=)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
NLE (<= 电平)		日期比较	3

■ 操作数设置

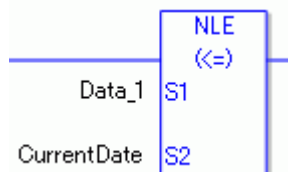
下面给出了 NLE 指令操作数 (S1, S2) 的可配置条件。

NLE 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 S2 的步数 + 1 = 一个指令的总步数

例如，计算 NLE 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {CurrentDate=1 步} + {1 步} = 3 步。

指令中包括一个最终步骤。必须加上这一步。

■ NLE 指令说明

NLE 指令中的日期变量是结构变量。下表列出内部结构。

日期变量

日期变量	变量设置	描述
变量名 .YR	整型变量	以 BCD 形式输入年份。
变量名 .MO	整型变量	以 BCD 形式输入月份
变量名 .DAY	整型变量	以 BCD 形式输入日期

■ 操作数设置

下面描述了 NLE 指令操作数 (S1, S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]		X
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	.YR/ .MO/ .DAY 未指定结构元素。	1	O
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	.YR/ .MO/ .DAY 未指定结构元素。	1	O
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

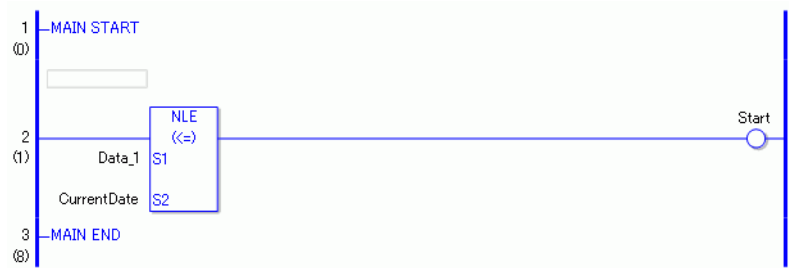
■ NLE 指令说明

NLE 指令比较日期。执行 NLE 指令时，将比较 S1 和 S2。如果结果是 $S1 \leq S2$ ，那么该指令就接通。同时比较年、月和日变量。使用 NLE 指令时，您只能在操作数 S1 和 S2 中指定日期变量。

程序示例

NLE

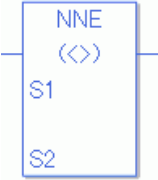
比较日期变量，用线圈来表示结果。



(1)比较 Data_1 和 CurrentDate，确定 Data_1 是否小于等于 CurrentDate。如果结果是 $S1 \leq S2$ ，那么该指令就接通，随即执行 NLE 指令右边的指令。在上图中，执行的是 NLE 指令右边的 OUT 指令。

31.21.6 NNE (<>)

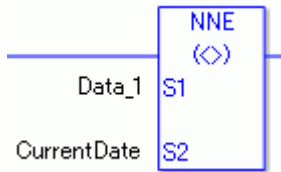
符号和功能

梯形图指令名称	梯形图符号	功能	步数
NNE (<> 电平)		日期比较	3

■ 操作数设置

下面给出了 NNE 指令操作数 (S1, S2) 的可配置条件。
NNE 指令的实际步数取决于指定的操作数。下面描述如何计算步数。
操作数 S1 的步数 + 操作数 S2 的步数 + 1 = 一个指令的总步数

例如，计算 NNE 指令的步数
(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 = 1 步} + {CurrentDate=1 步} + {1 步} = 3 步。

指令中包括一个最终步骤。必须加上这一步。

■ NNE 指令说明

NNE 指令中的日期变量是结构变量。下表列出内部结构。
日期变量

日期变量	变量设置	描述
变量名 .YR	整型变量	以 BCD 形式输入年份。
变量名 .MO	整型变量	以 BCD 形式输入月份
变量名 .DAY	整型变量	以 BCD 形式输入日期

■ 操作数设置

下面描述了 NNE 指令操作数 (S1, S2) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量] 或 指定整型变量 B/W [常量]		X
		指定整型变量 [变量] 或 指定整型变量 B/W [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	.YR / .MO / .DAY 未指定结构元素。	1	O
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	.YR/ .MO/ .DAY 未指定结构元素。	1	O
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

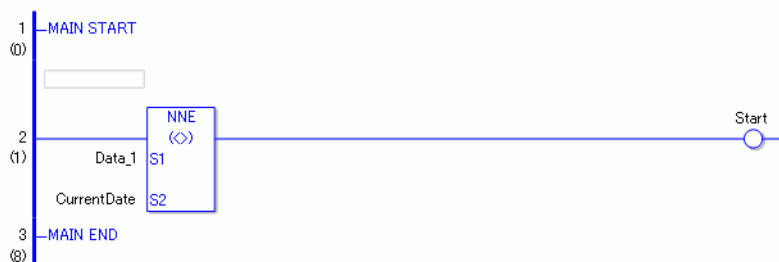
■ NNE 指令说明

NNE 指令比较日期。执行 NNE 指令时，将比较 S1 和 S2。如果结果是 $S1 \neq S2$ ，那么该指令就接通。同时比较年、月和日变量。使用 NNE 指令时，您只能在操作数 S1 和 S2 中指定日期变量。

程序示例

NNE

比较日期变量，用线圈来表示结果。



(1)比较 Data_1 和 CurrentDate，确定它们是否不相等。如果结果是 $S1 \neq S2$ ，那么该指令就接通，随即执行 NNE 指令右边的指令。在上图中，执行的是 NNE 指令右边的 OUT 指令。

31.22 数据转换指令

31.22.1 BCD/BCDP(BCD 转换)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
BCD (BCD 转换 - 电平)		数据转换	3 至 7
梯形图指令名称	梯形图符号	功能	步数
BCDP (BCD 转换 - 上升沿)		数据转换	3 至 7

■ 操作数设置

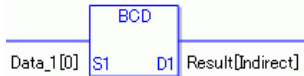
下面给出了 BCD/BCDP 指令操作数 (S1, D1) 的可配置条件。

BCD/BCDP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 BCD/BCDP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



$$\{ \text{Data_1}[0] = 2 \text{ 步} \} + \{ \text{Result}[\text{Indirect}] = 3 \text{ 步} \} + \{ 1 \text{ 步} \} = 6 \text{ 步}$$

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面描述了 BCD/BCDP 指令操作数 (S1, D1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: ○ 不可用: X
外部寄存器 地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	○
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	○
符号	位			X
	字		1	○
变量格式 *(注释 1) S1 = I/O 可用 D1 = 不能输入	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 * (注释 1)	未指定数组和修饰符	1	○
		指定整型变量 [常量]	2	○
		指定整型变量 [变量]	3	○
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	○
	计数器	仅 .PV / .CV	2	○
	日期	仅 .YR / .MO / .DAY	2	○
	时间	仅 .HR / .MIN / .SEC	2	○
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	○

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式 * (注释 2) D1 = 不可用	X_			X
	Y_			X
	M_			X
	I_* (注释 2)		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量 * (注释 3) D1 = 不可用	整型 * (注释 3)	0 至 99999999	1	O
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ BCD/BCDP 指令说明

BCD/BCDP 指令将数值转换成 BCD(二进制编码的十进制数) 码。将 S1 中的值转换成 BCD 码并保存在 D1 中。

BCD 和 BCDP 指令总是处于接通状态。在操作数 S1 中可以转换的最大值是 0x5F5E0FF。

如果您试图转换一个不能转换的值， D1 中的值将变成未定义的值。

使用 BCD/BCDP 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。

请在操作数 S1 和 D1 中指定相同的变量类型。

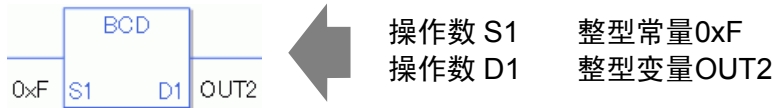
指定常量时，请参考如下内容。

当操作数 D1 是整型变量时



当操作数 D1 是整型变量，而您又想在操作数 S1 中输入十六进制值时。

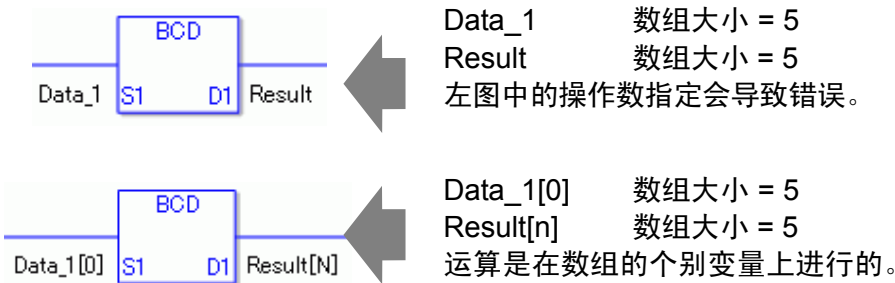
在数值前加上 0x(0 和小写字母 x) 即表示该数值为十六进制值。



当转换指定数组 (整型变量数组) 中的数据时

使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。

如果将 S1 和 D1 指定为整个数组，将会发生错误，即使变量类型相同。



■ 显示执行结果的系统变量

当执行结果是 0 时，#L_CalcZero 置 ON。
当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

(注释)
当使用系统变量检查结果时，确保在执行完指令后进行。
当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

BCD

将一个常量转换成二进制编码的十进制数并将它保存在结果数据中。



(1)当上升沿指令置 ON 时，将执行 BCD 指令。当执行 BCD 指令时，10(二进制时是 1010) 被转换为二进制编码的十进制数 0001 0000，并在 D1 中保存 0001 0000。当使用常开指令时，只要常开指令变量保持为 ON，就总是执行 BCD 指令。

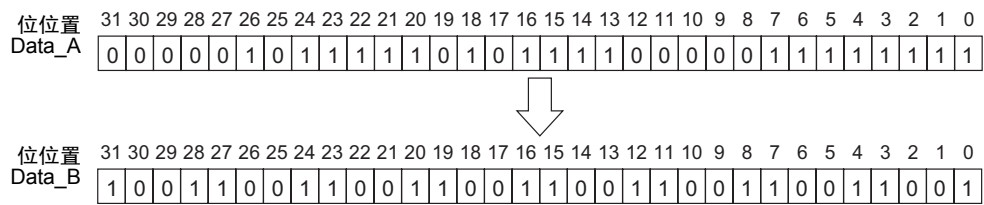
程序示例

BCDP



(1)BCDP 和 BCD 指令检测执行时间的方法不同。对于 BCDP 指令，即使使用常开指令，也只有在检测到上升沿时才会执行 BCDP 指令。即便常开指令变量保持为 ON，也只执行一次 BCDP 指令 (在 1 次扫描中执行)。

例如，S1(Data_A) 的 BCD 转换 = “99999999” 至 D1(Data_B)。



31.22.2 BIN/BINP(BIN 转换)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
BIN (BIN 转换 - 电平)		数据转换	3 至 7
梯形图指令名称	梯形图符号	功能	步数
BINP (BIN 转换 - 上升沿)		数据转换	3 至 7

■ 操作数设置

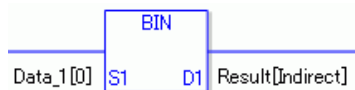
下面给出了 BIN/BINP 指令操作数 (S1, D1) 的可配置条件。

BIN/BINP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如：计算 BIN/BINP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{ Data_1[0] = 2 步 } + { Result[Indirect] = 3 步 } + { 1 步 } = 6 步

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面给出了 BIN/BINP 指令操作数 (S1, D1) 的可配置条件。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器 地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式 * (注释 1) S1 = I/O 可用 D1 = 不能输入	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 * (注释 1)	未指定数组和修饰符	1	O
		指定整型变量 [常量]	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用: O 不可用: X
地址格式 * (注释 2) D1 = 不可用	X_			X
	Y_			X
	M_			X
	I_* (注释 2)		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量 * (注释 3) D1 = 不可用	整型 * (注释 3)	0 至 999999999(BCD 值)	1	O
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ BIN/BINP 指令说明

BIN/BINP 指令将 BCD 值转换为二进制值。将 S1 中的值转换成二进制值并保存在 D1 中。

BIN 和 BINP 指令总是处于接通状态。在操作数 S1 中可以转换的最大值是 0x5F5E0FF。

如果您试图转换一个不能转换的值，D1 中的值将变成未定义的值。

使用 BIN/BINP 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。

请在操作数 S1 和 D1 中指定相同的变量类型。

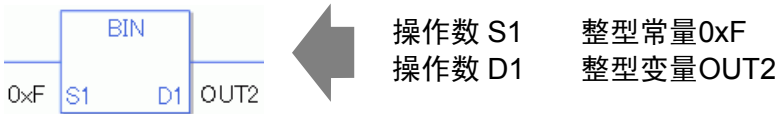
指定常量时，请参考如下内容。

当操作数 D1 是整型变量时



当操作数 D1 是整型变量，而您又想在操作数 S1 中输入十六进制值时。

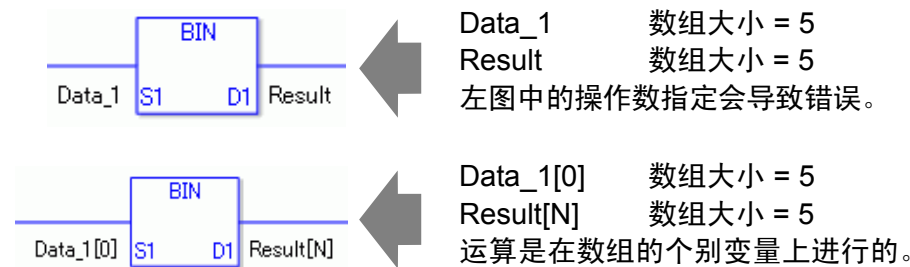
在数值前加上 0x(0 和小写字母 x) 即表示该数值为十六进制值。



当转换指定数组 (整型变量数组) 中的数据时

使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。

如果将 S1 和 D1 指定为整个数组，将会发生错误，即使变量类型相同。



■ 显示执行结果的系统变量

当执行结果是 0 时，#L_CalcZero 置 ON。
当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

(注释)
当使用系统变量检查结果时，确保在执行完指令后进行。
当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

BIN

将一个常量从 BCD 码转换为二进制并在结果数据中保存转换后的值。



(1)当上升沿指令置 ON 时，将执行 BIN 指令。当执行 BIN 指令时，将 0001 0000 (BCD 的 10) 转换成二进制并在 D1 中保存值 1010 。当使用常开指令时，只要常开指令变量保持为 ON，就总是执行 BIN 指令。

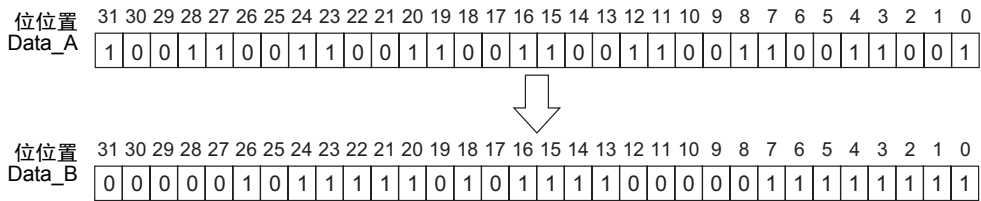
程序示例

BINP



(1)BINP 和 BIN 指令检测执行时间的方法不同。对于 BINP 指令，即使使用常开指令，也只有在检测到上升沿时才会执行 BINP 指令。即便常开指令变量保持为 ON，也只执行一次 BINP 指令 (在 1 次扫描中执行)。

例如S1(Data_A) 的 BIN 转换 = "99999999"BCD 至 D1(Data_B)。



31.22.3 ENCO/ENCOP(编码)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
ENCO (编码 - 电平)		数据转换	3 至 7
梯形图指令名称	梯形图符号	功能	步数
ENCOP (编码 - 上升沿)		数据转换	3 至 7

■ 操作数设置

下面给出了 ENCO/ENCOP 指令操作数 (S1, D1) 的可配置条件。
 ENCO/ENCOP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。
 操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如：计算 ENCO/ENCOP 指令的步数
 (有关操作数步数的信息，请参阅下一页中的操作数设置。)



{ Data_1[0] = 2 步 } + { Result[Indirect] = 3 步 } + { 1 步 } = 6 步

指令的总步数需要加最后一步。请务必加一步。

■ 操作数设置

下面给出了 ENCO/ENCOP 指令操作数 (S1, D1) 的可配置条件。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器 地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式 * (注释 1) S1 = I/O 可用 D1 = 不能输入	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 * (注释 1)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 数组	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用: O 不可用: X
地址格式 * (注释 2) D1 = 不可用	X_			X
	Y_			X
	M_			X
	I_* (注释 2)		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量 * (注释 3) D1 = 不可用	整型 * (注释 3)	-2147483648 至 2147483647	1	O
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

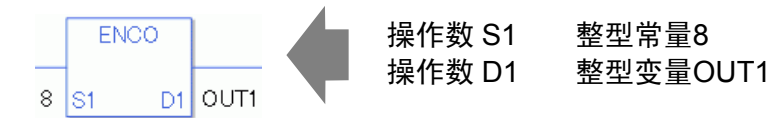
■ ENCO/ENCOP 指令说明

ENCO/ENCOP 指令对数值进行编码。将 S1 中的值编码并保存在 D1 中。在 S1 的 32 个位中，将 ON 位的位置以二进制形式输出到 D1。当 S1 中有多个位为 ON 时，输出最高位的位置。ENCO/ENCOP 指令总是处于接通状态。

使用 ENCO/ENCOP 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。请在操作数 S1 和 D1 中指定相同的变量类型。

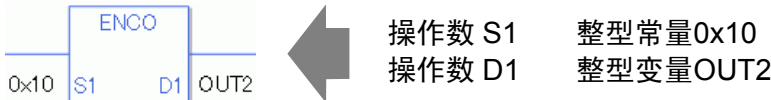
指定常量时，请参考如下内容。

当操作数 D1 是整型变量时



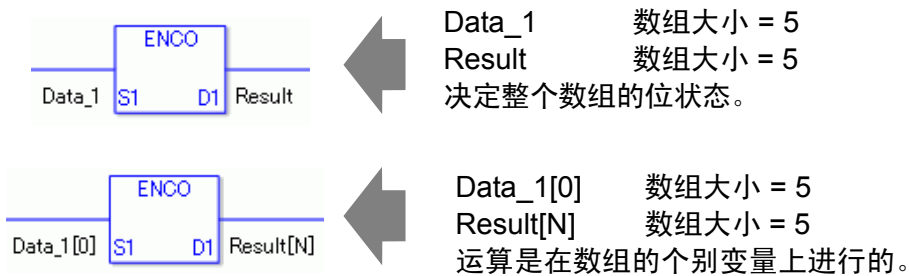
当操作数 D1 是整型变量，而您又想在操作数 S1 中输入十六进制值时。

当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



当转换指定数组 (整型变量数组) 中的数据时

既可以将操作数 S1 和 D1 指定为整个数组，也可以指定为单个的数组元素。



■ 显示执行结果的系统变量

当执行结果是 0 时，#L_CalcZero 置 ON。
当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

(注释)
当使用系统变量检查结果时，确保在执行完指令后进行。
当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

ENCO

转换一个常量并将转换后的值保存在结果数据中。



(1)当上升沿指令置 ON 时，将执行 ENCO 指令。当执行 ENCO 指令时，对 0000 1000 (十六进制的 8) 进行转换，将二进制值 0011(3) 保存在 D1 中。当使用常开指令时，只要常开指令变量保持为 ON，就总是执行 ENCO 指令。

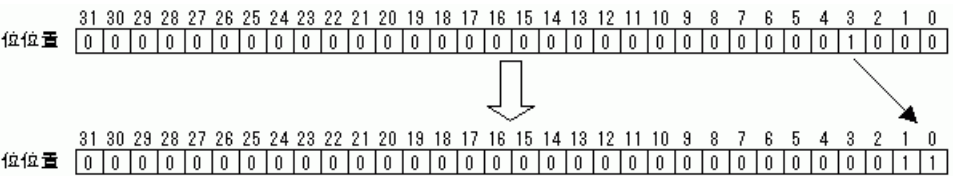
程序示例

ENCOP



(1)ENCOP 和 ENCO 指令检测执行时间的方法不同。对于 ENCOP 指令，即使使用常开指令，也只有在检测到上升沿时才会执行 ENCOP 指令。即便常开指令变量保持为 ON，也只执行一次 ENCOP 指令 (在 1 次扫描中执行)。

例如，当在 S1 中输入了 0x00000008，D1 中的输出值将是 0x00000003。



31.22.4 DECO/DECOP(解码)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
DECO (解码 - 电平)		数据转换	3 至 7
梯形图指令名称	梯形图符号	功能	步数
DECOP (解码 - 上升沿)		数据转换	3 至 7

■ 操作数设置

下面给出了 DECO/DECOP 指令操作数 (S1, D1) 的可配置条件。
 DECO/DECOP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。
 操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如：计算 DECO/DECOP 指令的步数
 (有关操作数步数的信息，请参阅下一页中的操作数设置。)



{ Data_1[0] = 2 步 } + { Result[Indirect] = 3 步 } + { 1 步 } = 6 步

指令的总步数需要加最后一步。请务必加一步。

■ 操作数设置

下面给出了 DECO/DECOP 指令操作数 (S1, D1) 的可配置条件。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器 地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式 * (注释 1) S1 = I/O 可用 D1 = 不能输入	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 * (注释 1)	未指定数组和修饰符	1	O
		指定整型变量 [常量] 数组	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用: O 不可用: X
地址格式 * (注释 2) D1 = 不可用	X_			X
	Y_			X
	M_			X
	I_* (注释 2)		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量 * (注释 3) D1 = 不可用	整型 * (注释 3)	0 至 131071(指定的数组)	1	O
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

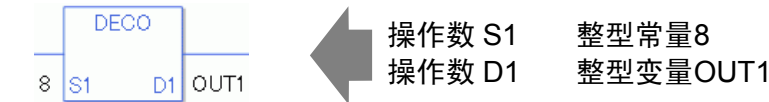
■ DECO/DECOP 指令说明

DECO/DECOP 指令对数值进行解码。对 S1 中的值解码并将结果保存在 D1 中。D1 中与 S1 值对应的位位置 ON。当使用输出数组时，可以解码的最大位位置是 (4096 x 32 - 1 = 131071)。

DECO/DECOP 指令总是处于接通状态。使用 DECO/DECOP 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。请在操作数 S1 和 D1 中指定相同的变量类型。

指定常量时，请参考如下内容。

当操作数 D1 是整型变量时



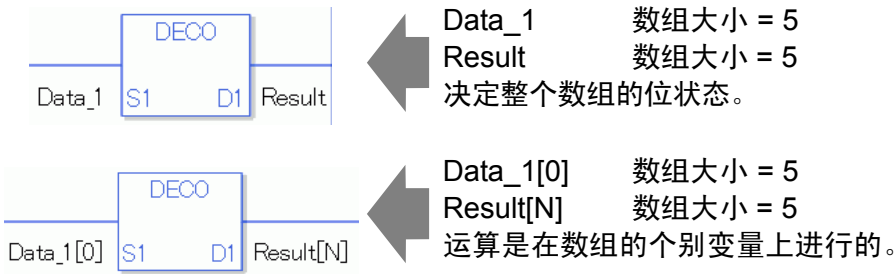
当操作数 D1 是整型变量，而您又想在操作数 S1 中输入十六进制值时。

当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



当转换指定数组 (整型变量数组) 中的数据时

既可以将操作数 S1 和 D1 指定为整个数组，也可以指定为单个的数组元素。



■ 显示执行结果的系统变量

当执行结果是 0 时，#L CalcZero 置 ON。

当执行发生错误时，将在 #L CalcErrCode 中保存错误代码。

(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

DECO

转换一个常量并将转换后的值保存在结果数据中。



- (1) 当上升沿指令置 ON 时，将执行 DECO 指令。当执行 DECO 指令时，将对 0000 1000(十六进制的 8) 进行转换，在 D1 中保存二进制值 1 0000 0000。
当使用常开指令时，只要常开指令变量保持为 ON，就总是执行 DECO 指令。

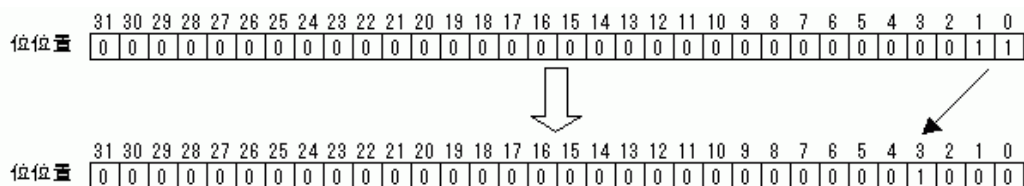
程序示例

DECOP



- (1)DECOP 和 DECO 指令检测执行时间的方法不同。对于 DECOP 指令，即使使用常开指令，也只有在检测到上升沿时才会执行 DECOP 指令。即便常开指令保持为 ON，也只执行一次 DECOP 指令（在 1 次扫描中执行）。

例如，当在S1 中输入3 时，D1 中的输出值就变成8。



31.22.5 RAD/RADP(转换至弧度)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
RAD (转换至弧度 - 电平)		数据转换	3 至 7
梯形图指令名称	梯形图符号	功能	步数
RADP (转换至弧度 - 上升沿)		数据转换	3 至 7

■ 操作数设置

下面给出了 RAD/RADP 指令操作数 (S1, D1) 的可配置条件。

RAD/RADP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如：计算 RAD/RADP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 [0] = 2 步} + {Result[N] = 3 步} + {1 步} = 6 步

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面给出了 RAD/RADP 指令操作数 (S1, D1) 的可配置条件。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器 地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (不包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量]		X
		指定整型变量 [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型	浮点型变量	1	O
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型	实型变量	1	O
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

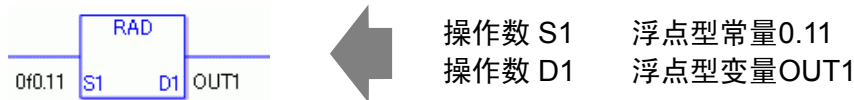
名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量 * (注释 1) D1 = 不可用	整型	-2147483648 至 2147483647		X
	浮点型 * (注释 1)	$\pm 1.175494351\text{e-}38$ 至 $\pm 3.402823466\text{e+}38$	1	O
	实型 * (注释 1)	$\pm 2.2250738585072014\text{e-}308$ 至 $\pm 1.7976931348623158\text{e+}308$	2	O

■ RAD/RADP 指令说明

RAD 和 RADP 指令是将角度转换为弧度的弧度转换指令。当执行 RAD 指令且接通回路后，将在 S1 中输入角度，并将转换后的弧度保存在 D1 中。Pi 约等于 3.1415926535897(实数)。RAD 和 RADP 指令总是处于接通状态。
使用 RAD/RADP 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。
请在操作数 S1 和 D1 中指定相同的变量类型。
指定常量时，请参考如下内容。

当操作数 D1 是浮点型变量时

当输入 0f(0 和小写字母 “f”) 时，将把后面的值视为浮点型数值。



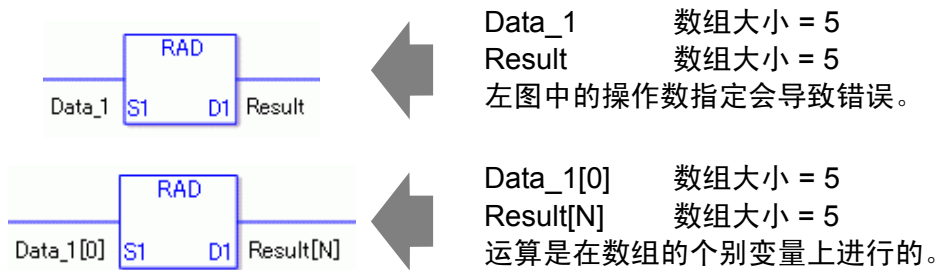
当操作数 D1 是实型变量时

当输入 0r(0 和小写字母 “r”) 时，将把后面的值视为实数值。



当计算指定数组中的数据时

使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。
如果将 S1 和 D1 指定为整个数组，将会发生错误，即使变量类型相同。



■ 显示执行结果的系统变量

当执行结果是 0 时，#L_CalcZero 置 ON。

当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

RAD



(1)当上升沿指令置 ON 时，将执行 RAD 指令。当执行 RAD 指令时，会将结果 Data_A 保存在 D1 中。

当使用常开指令时，只要常开指令变量保持为 ON，就总是执行 RAD 指令。

程序示例

RADP



(1)RADP 和 RAD 指令检测执行时间的方法不同。对于 RADP 指令，即使使用常开指令，也只有在检测到上升沿时才会执行 RADP 指令。即便常开指令保持为 ON，也只执行一次 RADP 指令 (在 1 次扫描中执行)。

31.22.6 DEG/DEGP(转换至角度)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
DEG (转换至角度 - 电平)		数据转换	3 至 7
梯形图指令名称	梯形图符号	功能	步数
DEGP (转换至角度 - 上升沿)		数据转换	3 至 7

■ 操作数设置

下面给出了 DEG/DEGP 指令操作数 (S1, D1) 的可配置条件。

DEG/DEGP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如：计算 DEG/DEGP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 [0] = 2 步 } + {Result[N] = 3 步 } + {1 步 } = 6 步

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面给出了 DEG/DEGP 指令操作数 (S1, D1) 的可配置条件。

名称	类型	条件	操作数 步数	可用：○ 不可用：X
外部寄存器 地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (不包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量]		X
		指定整型变量 [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型	浮点型变量	1	○
		指定浮点型变量 [常量]	2	○
		指定浮点型变量 [变量]	3	○
	实型	实型变量	1	○
		指定实型变量 [常量]	2	○
		指定实型变量 [变量]	3	○
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量 * (注释 1) D1 = 不可用	整型	-2147483648 至 2147483647		X
	浮点型 * (注释 1)	$\pm 1.175494351\text{e-}38$ 至 $\pm 3.402823466\text{e+}38$	1	O
	实型 * (注释 1)	$\pm 2.2250738585072014\text{e-}308$ 至 $\pm 1.7976931348623158\text{e+}308$	2	O

■ DEG 和 DEGP 指令说明

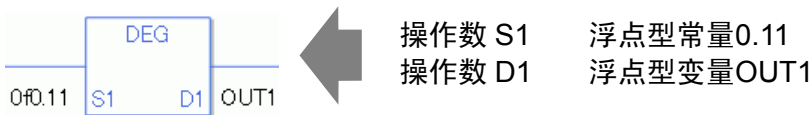
DEG/DEGP 指令将数值转换成角度。该指令将衡量角度的单位弧度转换成角度并保存在 D1 中。

Pi 约等于 3.1415926535897(实数)。DEG 和 DEGP 指令总是处于接通状态。使用 DEG/DEGP 指令时，如果为操作数 S1 和 D1 指定的变量类型不同，就会发生错误。请在操作数 S1 和 D1 中指定相同的变量类型。

指定常量时，请参考如下内容。

当操作数 D1 是浮点型变量时

当输入 0f(0 和小写字母 “f”) 时，将把后面的值视为浮点型数值。



当操作数 D1 是实型变量时

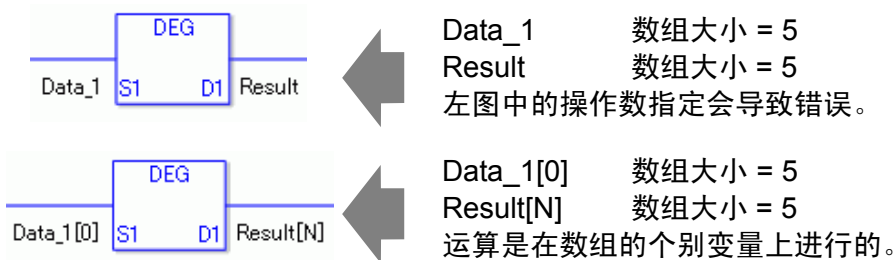
当输入 0r(0 和小写字母 “r”) 时，将把后面的值视为实数值。



当计算指定数组中的数据时

使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。

如果将 S1 和 D1 指定为整个数组，将会发生错误，即使变量类型相同。



■ 显示执行结果的系统变量

当执行结果是 0 时，#L_CalcZero 置 ON。

当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

DEG



(1)当上升沿指令置 ON 时，将执行 DEG 指令。当执行 DEG 指令时，会将结果 Data_A 保存在 D1 中。

当使用常开指令时，只要常开指令变量保持为 ON，就总是执行 DEG 指令。

程序示例

DEGP



(1)DEGP 和 DEG 指令检测执行时间的方法不同。对于 DEGP 指令，即使使用常开指令，也只有在检测到上升沿时才会执行 DEGP 指令。即便常开指令变量保持为 ON，也只执行一次 DEGP 指令 (在 1 次扫描中执行)。

31.22.7 SCL/SCLP(比例转换)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
SCL (比例转换 - 电平)		数据转换	7 至 11
梯形图指令名称	梯形图符号	功能	步数
SCLP (比例转换 - 上升沿)		数据转换	7 至 11

■ 操作数设置

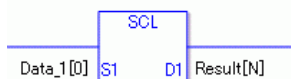
下面给出了 SCL/SCLP 指令操作数 (S1, D1) 的可配置条件。

SCL/SCLP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 SCL/SCLP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 [0] = 2 步 } + {Result[N] = 3 步 } + {5 步 } = 10 步

指令中包含最后 5 个步骤。必须加上这 5 步。

■ 操作数设置

下面给出了 SCL/SCLP 指令操作数 (S1, D1) 的可配置条件。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器 地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式 * (注释 1) D1 = 不能输入	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 * (注释 1)	未指定数组和修饰符	1	O
		指定整型变量 [常量]	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型	浮点型变量	1	O
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型	实型变量	1	O
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

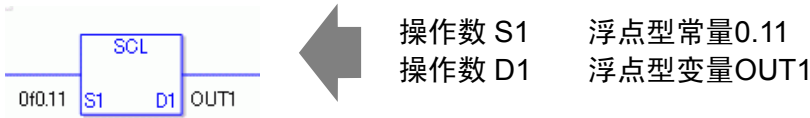
名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式 * (注释 2) D1 = 不可用	X_			X
	Y_			X
	M_			X
	I_* (注释 2)		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_		1	O
	R_		1	O
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量 * (注释 3) D1 = 常量不可用	整型	-2147483648 至 2147483647	1	O
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38	1	O
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308	2	O

■ SCL/SCLP 指令说明

SCL/SCLP 指令将数值转换成比例。该指令将根据上限和下限转换 S1 中的值，并将转换后的值保存在 D1 中。如果在操作数 S1 和 D1 中指定的变量类型不同，就会发生错误。请在操作数 S1 和 D1 中指定相同的变量类型。
指定常量时，请参考如下内容。

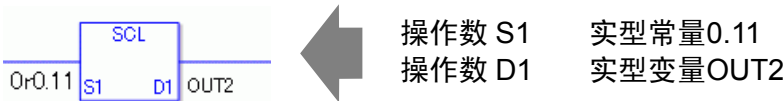
当操作数 D1 是浮点型变量时

当输入 Of(0 和小写字母 “f”) 时，将把后面的值视为浮点型数值。



当操作数 D1 是实型变量时

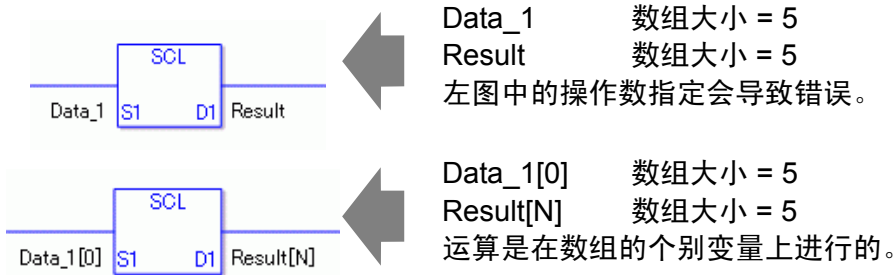
当输入 Or(0 和小写字母 “r”) 时，将把后面的值视为实数值。



当计算指定数组中的数据时

使用数据 [0] 或数据 [N] 指定数组 (N 表示整型变量)。

如果将 S1 和 D1 指定为整个数组，将会发生错误，即使变量类型相同。



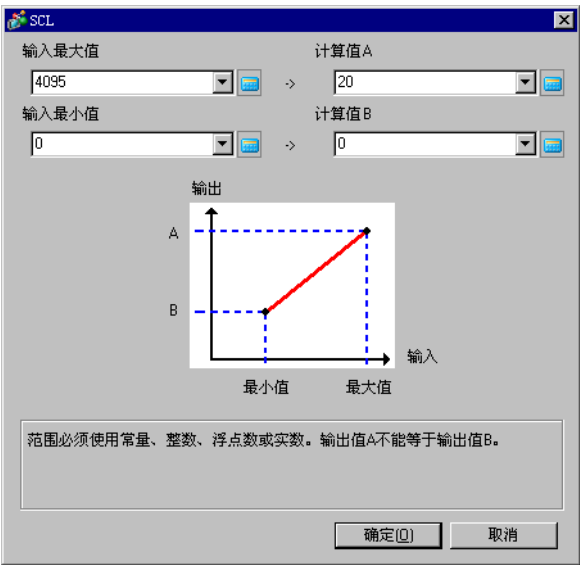
■ 显示执行结果的系统变量

当执行结果是 0 时，#L_CalcZero 置 ON。
当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

(注释)
当使用系统变量检查结果时，确保在执行完指令后进行。
当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

■ 输入和输出的上限和下限

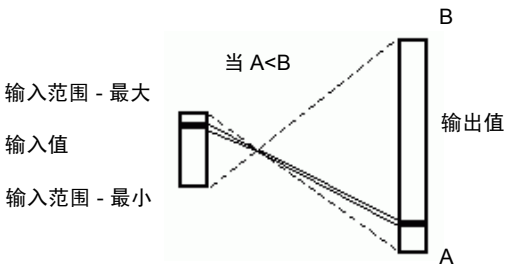
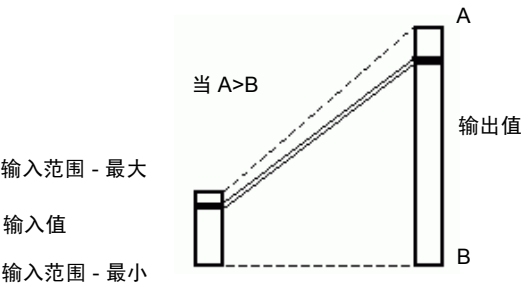
双击 SCL 指令，显示如下对话框。在对话框中，设定最大和最小输入值及输出 A 和输出 B 的值。



- (注释 1) 当设置最大 / 最小输入值及输出值 A 和 B 时，您不能间接指定数组元素。
数组变量名称 数据
数组 5
O 数据 [0] X 数据 [N]
- (注释 2) 当在操作数 S1 或 D1 中使用实型或浮点型变量，并使用常量来定义最大 / 最小输入值以及输出 A 和输出 B 的值时，请用 “Or” 和 “Of” 来标明实数和浮点数。

当输出值 A > 输出值 B

当输出值 A < 输出值 B

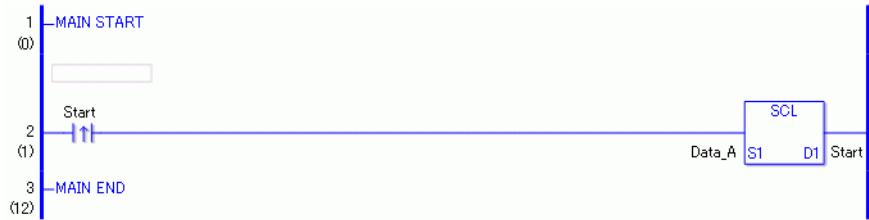


程序示例

SCL

将模拟输入值 (0 至 4095) 转换为 4 至 20[最大] 范围内的当前值并用十进制表示该值。

在 SCL 指令设置对话框中，设置最大输入值 = 0r4095，最小输入值 = 0r0， A = 0r20， B =0r4。



(1)当上升沿指令置 ON 时，将执行 SCL 指令。当执行 SCL 指令时，会将结果 Data_A 保存在 D1 中。

当使用常开指令时，只要常开指令变量保持为 ON，就总是执行 SCL 指令。

程序示例

SCLP



(1)SCLP 和 SCL 指令的运行时间不同。对于 SCLP 指令，即使使用常开指令，也只有在检测到上升沿时才会执行 SCLP 指令。因此，即使常开指令位保持为 ON，也只在一次扫描中执行 SCLP 指令。

31.23 类型转换指令

31.23.1 I2F/I2FP(整型至浮点型转换)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
I2F (整型至浮点型转换 - 电平)		类型转换	3 至 7
梯形图指令名称	梯形图符号	功能	步数
I2FP (整型至浮点型转换 - 上升沿)		类型转换	3 至 7

■ 操作数设置

下面给出了 I2F/I2FP 指令操作数 (S1, D1) 的可配置条件。
I2F/I2FP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。
操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如：计算 I2F/I2FP 指令的步数
(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 [0] = 2 步 } + {Result[N] = 3 步 } + {1 步 } = 6 步

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面描述了 I2F/I2FP 指令操作数 (S1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: ○ 不可用: X
外部寄存器 地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	○
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	○
符号	位			X
	字		1	○
变量格式 * (注释 1) 包括 I/O	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 * (注释 1)	未指定数组和修饰符	1	○
		指定整型变量 [常量]	2	○
		指定整型变量 [变量]	3	○
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型	未指定数组和修饰符		X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型	未指定数组和修饰符		X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	○
	计数器	仅 .PV / .CV	2	○
	日期	仅 .YR / .MO / .DAY	2	○
	时间	仅 .HR / .MIN / .SEC	2	○
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	○

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	O
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647	1	O
	浮点型	$\pm 1.175494351\text{e-}38$ 至 $\pm 3.402823466\text{e+}38$		X
	实型	$\pm 2.2250738585072014\text{e-}308$ 至 $\pm 1.7976931348623158\text{e+}308$		X

■ 操作数设置

下面描述了 I2F/I2FP 指令操作数 (D1) 的可指定内容。

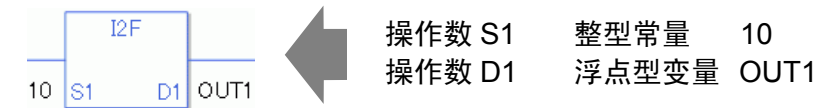
名称	类型	条件	操作数 步数	可用：○ 不可用：X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (不包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量]		X
		指定整型变量 [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型	浮点型变量	1	○
		指定浮点型变量 [常量]	2	○
		指定浮点型变量 [变量]	3	○
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_		1	O
	R_			X
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量	整型	-2147483648 至 2147483647		X
	浮点型	$\pm 1.175494351\text{e-}38$ 至 $\pm 3.402823466\text{e+}38$		X
	实型	$\pm 2.2250738585072014\text{e-}308$ 至 $\pm 1.7976931348623158\text{e+}308$		X

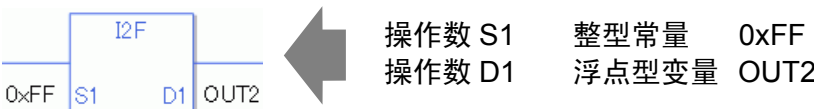
■ I2F/I2FP 指令说明

I2F/I2FP 指令将整型变量转换为浮点型变量。在 S1 中指定您想转换的整型变量或常量并为 D1 中的转换输出结果指定浮点型变量。您只能为 S1 中的输入指定一个整型变量，为 D1 中的输出指定一个浮点型变量。当您在计算或比较中使用不同的变量类型时请使用转换指令。
指定常量时，请参考如下内容。

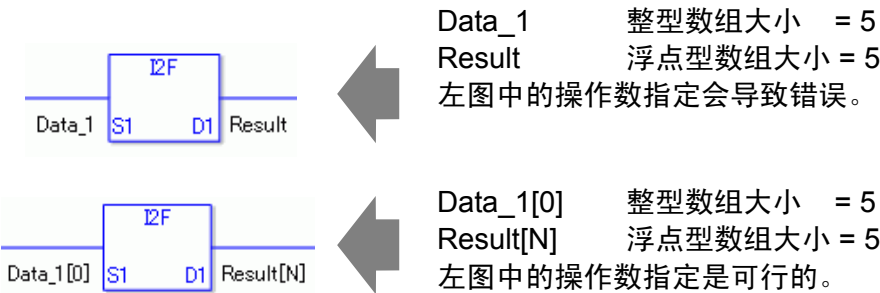
当操作数 S1 是整型常量时



当操作数 S1 是整型常量，而您想在其中输入一个十六进制值时。
当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



需要注意的是不能转换指定数组 (整个数组)。
如果将 S1 和 D1 指定为整个数组，将会发生错误，即使变量类型相同。



■ 显示执行结果的系统变量

当执行结果是 0 时，#L_CalcZero 置 ON。

当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

I2F



(1)当上升沿指令置 ON 时，将执行 I2F 指令。当执行 I2F 时，会将 Data_A 的 I2F 转换结果保存在 D1 中。

当使用常开指令时，只要常开指令变量保持为 ON，就总是执行 I2F 指令。

程序示例

I2FP



(1)I2FP 和 I2F 指令检测执行时间的方法不同。对于 I2FP 指令，即使使用常开指令，也只有在检测到上升沿时才会执行 I2FP 指令。即便常开指令变量保持为 ON，也只执行一次 I2FP 指令 (在 1 次扫描中执行)。

31.23.2 I2R/I2RP(整型至实型转换)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
I2R (整型至实型转换 - 电平)		类型转换	3 至 7
梯形图指令名称	梯形图符号	功能	步数
I2RP (整型至实型转换 - 上升沿)		类型转换	3 至 7

■ 操作数设置

下面给出了 I2R/I2RP 指令操作数 (S1, D1) 的可配置条件。

I2R/I2RP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如：计算 I2R/I2RP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 [0] = 2 步 } + {Result[N] = 3 步 } + {1 步 } = 6 步

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面描述了 I2R/I2RP 指令操作数 (S1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: ○ 不可用: X
外部寄存器 地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	○
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	○
符号	位			X
	字		1	○
变量格式 * (注释 1) 包括 I/O	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 * (注释 1)	未指定数组和修饰符	1	○
		指定整型变量 [常量]	2	○
		指定整型变量 [变量]	3	○
		指定整型变量 [常量 / 变量]		X
		.B/W [常量 / 变量]		
	浮点型	未指定数组和修饰符		X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型	未指定数组和修饰符		X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	○
	计数器	仅 .PV / .CV	2	○
	日期	仅 .YR / .MO / .DAY	2	○
	时间	仅 .HR / .MIN / .SEC	2	○
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	○

名称	类型	条件	操作数 步数	可用：○ 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_		1	○
	Q_		1	○
	D_	未指定修饰符	1	○
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	○
	C_	仅 .PV / .CV	2	○
	N_	仅 .YR / .MO / .DAY	2	○
	J_	仅 .HR / .MIN / .SEC	2	○
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	○
常量	整型	-2147483648 至 2147483647	1	○
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ 操作数设置

下面描述了 I2R/I2RP 指令操作数 (D1) 的可指定内容。

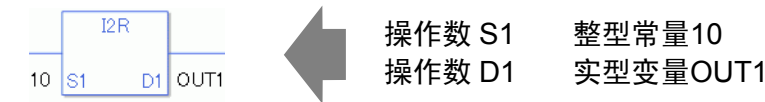
名称	类型	条件	操作数 步数	可用: ○ 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (不包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量]		X
		指定整型变量 [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型	实型变量	1	○
		指定实型变量 [常量]	2	○
		指定实型变量 [变量]	3	○
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_		1	O
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量	整型	-2147483648 至 2147483647		X
	浮点型	$\pm 1.175494351\text{e-}38$ 至 $\pm 3.402823466\text{e+}38$		X
	实型	$\pm 2.2250738585072014\text{e-}308$ 至 $\pm 1.7976931348623158\text{e+}308$		X

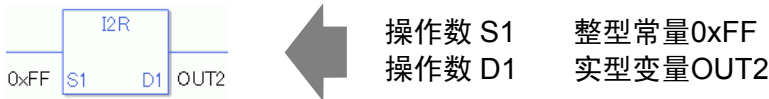
■ I2R/I2RP 指令说明

I2R/I2RP 指令将整型变量转换为实型变量。在 S1 中指定您想转换的整型变量或常量并为 D1 中的转换输出结果指定实型变量。您只能为 S1 中的输入指定整型变量，为 D1 中的输出指定实型变量。当您想在计算或比较中使用不同的变量类型时请使用转换指令。
指定常量时，请参考如下内容。

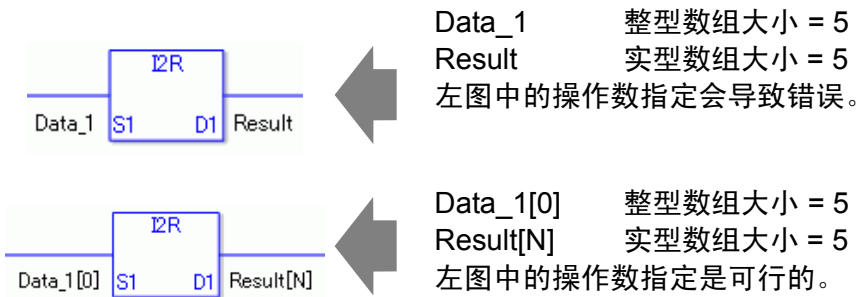
当操作数 S1 是整型常量时



当操作数 S1 是整型常量，而您想在其中输入一个十六进制值时。
当输入 0x(0 和小写字母 “x”) 时，将把后面的值视为十六进制值。



需要注意的是不能转换指定数组 (整个数组)。
如果将 S1 和 D1 指定为整个数组，将会发生错误，即使变量类型相同。



■ 显示执行结果的系统变量

当执行结果是 0 时，#L_CalcZero 置 ON。

当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

I2R



(1)当上升沿指令置 ON 时，将执行 I2R 指令。当执行 I2R 时，会将 Data_A 的 I2R 转换结果保存在 D1 中。

当使用常开指令时，只要常开指令变量保持为 ON，就总是执行 I2R 指令。

程序示例

I2RP



(1)I2RP 和 I2R 指令检测执行时间的方法不同。对于 I2RP 指令，即使使用常开指令，也只有在检测到上升沿时才会执行 I2RP 指令。即便常开指令变量保持为 ON，也只执行一次 I2RP 指令 (在 1 次扫描中执行)。

31.23.3 F2I/F2IP(浮点型至整型转换)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
F2I (浮点型至整型转换 - 电平)		类型转换	3 至 7
梯形图指令名称	梯形图符号	功能	步数
F2IP (浮点型至整型转换 - 上升沿)		类型转换	3 至 7

■ 操作数设置

下面给出了 F2I/F2IP 指令操作数 (S1, D1) 的可配置条件。
F2I/F2IP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。
操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如：计算 F2I/F2IP 指令的步数
(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 [0] = 2 步 } + {Result[N] = 3 步 } + {1 步 } = 6 步

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面描述了 F2I/F2IP 指令操作数 (S1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (不包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量]		X
		指定整型变量 [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型	浮点型变量	1	O
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_		1	O
	R_			X
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量	整型	-2147483648 至 2147483647		X
	浮点型	$\pm 1.175494351\text{e-}38$ 至 $\pm 3.402823466\text{e}+38$	1	O
	实型	$\pm 2.2250738585072014\text{e-}308$ 至 $\pm 1.7976931348623158\text{e}+308$		X

■ 操作数设置

下面描述了 F2I/F2IP 指令操作数 (D1) 的可指定内容。

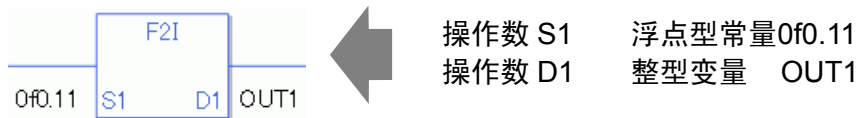
名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式 * (注释 1) 仅输出	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 * (注释 1)	未指定数组和修饰符	1	O
		指定整型变量 [常量]	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型	未指定数组和修饰符		X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型	未指定数组和修饰符		X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

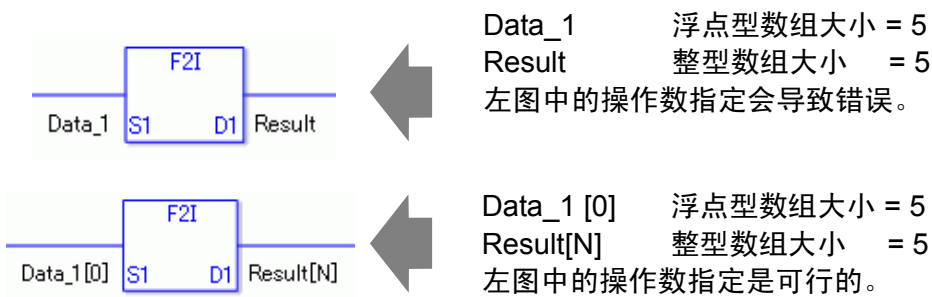
■ R2I/R2IP 指令说明

F2I/F2IP 指令将浮点型变量转换为整型变量。在 S1 中指定您想转换的浮点型变量或常量并为 D1 中的转换输出结果指定整型变量。您只能为 S1 中的输入指定浮点型变量，为 D1 中的输出指定整型变量。当您想在计算或比较中使用不同的变量类型时请使用转换指令。
指定常量时，请参考如下内容。

当操作数 D1 是浮点型常量时



需要注意的是不能转换指定数组 (整个数组)。
如果将 S1 和 D1 指定为整个数组，将会发生错误，即使变量类型相同。



■ 显示执行结果的系统变量

#L_CalcZero 当结果为 0 时置 ON 的系统变量。

#L_CalcCarry 当结果溢出时置 ON 的系统变量。

#L_CalcErrCode 当发生运算错误时保存错误代码的系统变量。

(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

F2I



(1) 当上升沿指令置 ON 时，将执行 F2I 指令。当执行 F2I 指令时，会将 Data_A 的 F2I 转换结果保存在 D1 中。

当使用常开指令时，只要常开指令变量保持为 ON，就总是执行 F2I 指令。

程序示例

F2IP



(1) F2IP 和 F2I 指令检测执行时间的方法不同。对于 F2IP 指令，即使使用常开指令，也只有在检测到上升沿时才会执行 F2IP 指令。即便常开指令变量保持为 ON，也只执行一次 F2IP 指令 (在 1 次扫描中执行)。

31.23.4 F2R/F2RP(浮点型至实型转换)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
F2R (浮点型至实型转换 - 电平)		类型转换	3 至 7
梯形图指令名称	梯形图符号	功能	步数
F2RP (浮点型至实型转换 - 上升沿)		类型转换	3 至 7

■ 操作数设置

下面给出了 F2R/F2RP 指令操作数 (S1, D1) 的可配置条件。
F2R/F2RP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。
操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如：计算 F2R/F2RP 指令的步数
(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 [0] = 2 步 } + {Result[N] = 3 步 } + {1 步 } = 6 步

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面描述了 F2R/F2RP 指令操作数 (S1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (不包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量]		X
		指定整型变量 [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型	浮点型变量	1	O
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_		1	O
	R_			X
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量	整型	-2147483648 至 2147483647		X
	浮点型	$\pm 1.175494351\text{e-}38$ 至 $\pm 3.402823466\text{e}+38$	1	O
	实型	$\pm 2.2250738585072014\text{e-}308$ 至 $\pm 1.7976931348623158\text{e}+308$		X

■ 操作数设置

下面描述了 F2R/F2RP 指令操作数 (D1) 的可指定内容。

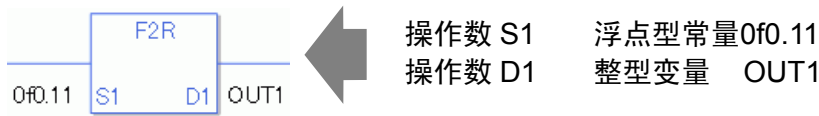
名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式 * (注释 1) 仅输出	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 * (注释 1)	未指定数组和修饰符		X
		指定整型变量 [常量]		X
		指定整型变量 [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型	实型变量	1	O
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_		1	O
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量	整型	-2147483648 至 2147483647		X
	浮点型	$\pm 1.175494351\text{e-}38$ 至 $\pm 3.402823466\text{e+}38$		X
	实型	$\pm 2.2250738585072014\text{e-}308$ 至 $\pm 1.7976931348623158\text{e+}308$		X

■ F2R/F2RP 指令说明

F2R/F2RP 指令将浮点型变量转换为实型变量。在 S1 中指定您想转换的浮点型变量或常量并为 D1 中的转换输出结果指定实型变量。您只能为 S1 中的输入指定浮点型变量，为 D1 中的输出指定实型变量。当您想在计算或比较中使用不同的变量类型时请使用转换指令。
指定常量时，请参考如下内容。

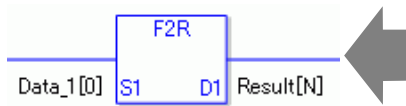
当操作数 D1 是浮点型常量时



需要注意的是不能转换指定数组 (整个数组)。
如果将 S1 和 D1 指定为整个数组，将会发生错误，即使变量类型相同。
Data_1 浮点型数组大小 = 5
Result 整型数组大小 = 5
左图中的操作数指定会导致错误。



Data_1 [0] 浮点型数组大小 = 5
Result[N] 整型数组大小 = 5
左图中的操作数指定是可行的。



■ 显示执行结果的系统变量

当执行结果是 0 时，#L_CalcZero 置 ON。

当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

F2R



(1)当上升沿指令置 ON 时，将执行 F2R 指令。当执行 F2R 指令时，会将 Data_A 的 F2R 转换结果保存在 D1 中。

当使用常开指令时，只要常开指令变量保持为 ON，就总是执行 F2R 指令。

程序示例

F2RP



(1)F2RP 和 F2R 指令检测执行时间的方法不同。对于 F2RP 指令，即使使用常开指令，也只有在检测到上升沿时才会执行 F2RP 指令。即便常开指令变量保持为 ON，也只执行一次 F2RP 指令 (在 1 次扫描中执行)。

31.23.5 R2I/R2IP(实型至整型转换)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
R2I (实型至整型转换 - 电平)		类型转换	3 至 7
梯形图指令名称	梯形图符号	功能	步数
R2IP (实型至整型转换 - 上升沿)		类型转换	3 至 7

■ 操作数设置

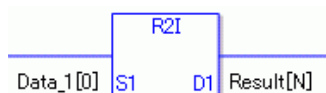
下面给出了 R2I/R2IP 指令操作数 (S1, D1) 的可配置条件。

R2I/R2IP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如：计算 R2I/R2IP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 [0] = 2 步 } + {Result[N] = 3 步 } + {1 步 } = 6 步

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面描述了 R2I/R2IP 指令操作数 (S1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (不包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量]		X
		指定整型变量 [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型	实型变量	1	O
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_		1	O
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量	整型	-2147483648 至 2147483647		X
	浮点型	$\pm 1.175494351\text{e-}38$ 至 $\pm 3.402823466\text{e+}38$		X
	实型	$\pm 2.2250738585072014\text{e-}308$ 至 $\pm 1.7976931348623158\text{e+}308$	1	O

■ 操作数设置

下面描述了 R2I/R2IP 指令操作数 (D1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式 * (注释 1) 仅输出	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 * (注释 1)	未指定数组和修饰符	1	O
		指定整型变量 [常量]	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型	未指定数组和修饰符		X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型	未指定数组和修饰符		X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

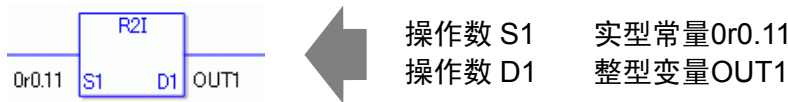
名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ R2I/R2IP 指令说明

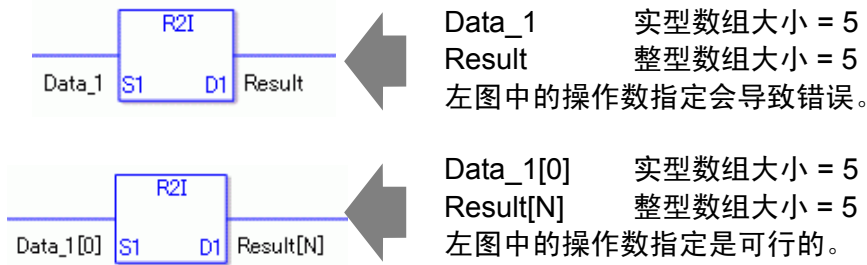
R2I/R2IP 指令将实型变量转换为整型变量。在 S1 中指定您想转换的实型变量或常量并为 D1 中的转换输出结果指定整型变量。您只能为 S1 中的输入指定实型变量，为 D1 中的输出指定整型变量。当您想在计算或比较中使用不同的变量类型时请使用转换指令。

指定常量时，请参考如下内容。

当操作数 S1 是实型常量时



需要注意的是不能转换指定数组 (整个数组)。
如果将 S1 和 D1 指定为整个数组，将会发生错误，即使变量类型相同。



■ 显示执行结果的系统变量

当执行结果是 0 时，#L_CalcZero 置 ON。

当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

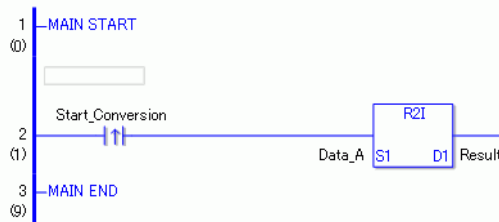
(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

R2I



(1)当上升沿指令置 ON 时，将执行 R2I 指令。当执行 R2I 指令时，会将 Data_A 的 R2I 转换结果保存在 D1 中。

当使用常开指令时，只要常开指令变量保持为 ON，就总是执行 R2I 指令。

程序示例

R2IP



(1)R2IP 和 R2I 指令检测执行时间的方法不同。对于 R2IP 指令，即使使用常开指令，也只有在检测到上升沿时才会执行 R2IP 指令。即便常开指令变量保持为 ON，也只执行一次 R2IP 指令 (在 1 次扫描中执行)。

31.23.6 R2F/R2FP(实型至浮点型转换)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
R2F (实型至浮点型转换 - 电平)		类型转换	3 至 7
梯形图指令名称	梯形图符号	功能	步数
R2FP (实型至浮点型转换 - 上升沿)		类型转换	3 至 7

■ 操作数设置

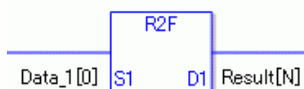
下面给出了 R2F/R2FP 指令操作数 (S1, D1) 的可配置条件。

R2F/R2FP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如：计算 R2F/R2FP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{Data_1 [0] = 2 步 } + {Result[N] = 3 步 } + {1 步 } = 6 步

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面描述了 R2F/R2FP 指令操作数 (S1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (不包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量]		X
		指定整型变量 [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型	实型变量	1	O
		指定实型变量 [常量]	2	O
		指定实型变量 [变量]	3	O
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_		1	O
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量	整型	-2147483648 至 2147483647		X
	浮点型	$\pm 1.175494351\text{e-}38$ 至 $\pm 3.402823466\text{e+}38$		X
	实型	$\pm 2.2250738585072014\text{e-}308$ 至 $\pm 1.7976931348623158\text{e+}308$	1	O

■ 操作数设置

下面描述了 R2F/R2FP 指令操作数 (D1) 的可指定内容。

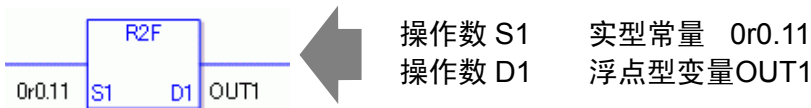
名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式 * (注释 1) 仅输出	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 * (注释 1)	未指定数组和修饰符		X
		指定整型变量 [常量]		X
		指定整型变量 [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型	浮点型变量	1	O
		指定浮点型变量 [常量]	2	O
		指定浮点型变量 [变量]	3	O
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_		1	O
	R_			X
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量	整型	-2147483648 至 2147483647		X
	浮点型	$\pm 1.175494351\text{e-}38$ 至 $\pm 3.402823466\text{e+}38$		X
	实型	$\pm 2.2250738585072014\text{e-}308$ 至 $\pm 1.7976931348623158\text{e+}308$		X

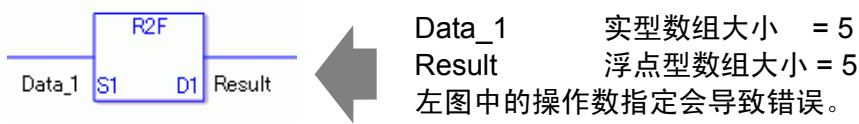
■ R2F/R2FP 指令说明

R2F/R2FP 指令将实型变量转换为浮点型变量。在 S1 中指定您想转换的实型变量或常量并为 D1 中的转换输出结果指定浮点型变量。您只能为 S1 中的输入指定实型变量，为 D1 中的输出指定浮点型变量。当您想在计算或比较中使用不同的变量类型时请使用转换指令。
指定常量时，请参考如下内容。

当操作数 S1 是实型常量时



需要注意的是不能转换指定数组 (整个数组)。
如果将 S1 和 D1 指定为整个数组，将会发生错误，即使变量类型相同。



■ 显示执行结果的系统变量

当执行结果是 0 时，#L_CalcZero 置 ON。

当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

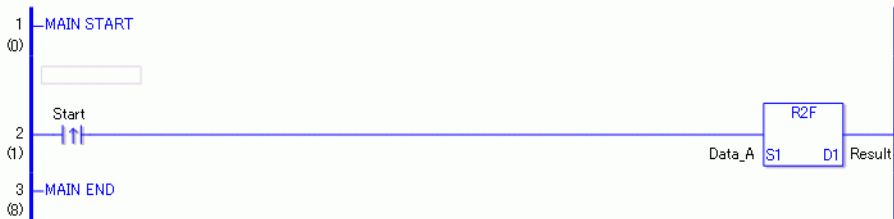
(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

R2F

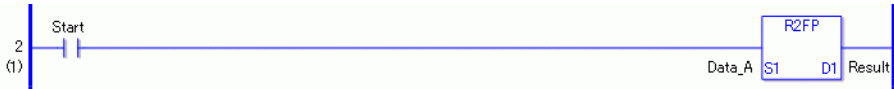


(1)当上升沿指令置 ON 时，将执行 R2F 指令。当执行 R2F 指令时，会将 Data_A 的 R2F 转换结果保存在 D1 中。

当使用常开指令时，只要常开指令变量保持为 ON，就总是执行 R2F 指令。

程序示例

R2FP



(1)R2FP 和 R2F 指令检测执行时间的方法不同。对于 R2FP 指令，即使使用常开指令，也只有在检测到上升沿时才会执行 R2FP 指令。即便常开指令变量保持为 ON，也只执行一次 R2FP 指令 (在 1 次扫描中执行)。

31.23.7 H2S/H2SP(时至秒转换)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
H2S (时至秒转换 - 电平)		类型转换	3 至 5
梯形图指令名称	梯形图符号	功能	步数
H2SP (时至秒转换 - 上升沿)		类型转换	3 至 5

■ 操作数设置

下面给出了 H2S/H2SP 指令操作数 (S1, D1) 的可配置条件。

H2S/H2SP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如：计算 H2S/H2SP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{CurrentTime = 1 步} + {Total[0] = 2 步} + {1 步} = 4 步

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面描述了 H2S/H2SP 指令操作数 (S1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: ○ 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (不包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量]		X
		指定整型变量 [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	非 .HR / .MIN / .SEC	1	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	非 .HR / .MIN / .SEC	1	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量	整型	-2147483648 至 2147483647		X
	浮点型	$\pm 1.175494351\text{e-}38$ 至 $\pm 3.402823466\text{e+}38$		X
	实型	$\pm 2.2250738585072014\text{e-}308$ 至 $\pm 1.7976931348623158\text{e+}308$		X

■ 操作数设置

下面描述了 H2S/H2SP 指令操作数 (D1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器 地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式 * (注释 1) 仅输出	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 * (注释 1)	未指定数组和修饰符	1	O
		指定整型变量 [常量]	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量]		X
		.B/W [常量 / 变量]		
	浮点型	未指定数组和修饰符		X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型	未指定数组和修饰符		X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ H2S/H2SP 指令说明

H2S/H2SP 指令将时间变量中的秒转换为整型变量。在 S1 中指定您想转换的时间变量并为 D1 中的转换输出结果指定整型变量。您只能为 S1 中的输入指定时间变量，为 S2 中的输出指定整型变量。不能用数组配置时间变量。0:30 将被转换成 1800 秒。

14:00 被转换成 50400 秒。

■ 显示执行结果的系统变量

当执行结果是 0 时，#L_CalcZero 置 ON。

当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

H2S



(1)当上升沿指令置 ON 时，将执行 H2S 指令。当执行 H2S 时，会将 Data_A 的 H2S 转换结果保存在 D1 中。

当使用常开指令时，只要常开指令变量保持为 ON，就总是执行 H2S 指令。

程序示例

H2SP



(1)H2SP 和 H2S 指令检测执行时间的方法不同。对于 H2SP 指令，即使使用常开指令，也只有在检测到上升沿时才执行 H2SP 指令。即便常开指令变量保持为 ON，也只执行一次 H2SP 指令 (在 1 次扫描中执行)。

31.23.8 S2H/S2HP(秒至时转换)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
S2H (秒至时转换 - 电平)		类型转换	3 至 5
梯形图指令名称	梯形图符号	功能	步数
S2HP (秒至时转换 - 上升沿)		类型转换	3 至 5

■ 操作数设置

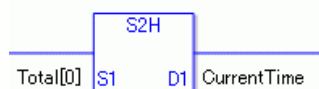
下面给出了 S2H/S2HP 指令操作数 (S1, D1) 的可配置条件。

S2H/S2HP 指令的实际步数取决于指定的操作数。下面描述如何计算步数。

操作数 S1 的步数 + 操作数 D1 的步数 + 1 = 一个指令的总步数

例如，计算 S2H/S2HP 指令的步数

(有关操作数步数的信息，请参阅下一页中的操作数设置。)



{CurrentTime = 1 步} + {Total[0] = 2 步} + {1 步} = 4 步

指令中包括一个最终步骤。必须加上这一步。

■ 操作数设置

下面描述了 S2H/S2HP 指令操作数 (S1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器 地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)	1	O
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)	1	O
符号	位			X
	字		1	O
变量格式 * (注释 1) 仅输出	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 * (注释 1)	未指定数组和修饰符	1	O
		指定整型变量 [常量]	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 [常量 / 变量]		X
		.B/W [常量 / 变量]		
	浮点型	未指定数组和修饰符		X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型	未指定数组和修饰符		X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET	2	O
	计数器	仅 .PV / .CV	2	O
	日期	仅 .YR / .MO / .DAY	2	O
	时间	仅 .HR / .MIN / .SEC	2	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_		1	O
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET	2	O
	C_	仅 .PV / .CV	2	O
	N_	仅 .YR / .MO / .DAY	2	O
	J_	仅 .HR / .MIN / .SEC	2	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST	2	O
常量	整型	-2147483648 至 2147483647		X
	浮点型	$\pm 1.175494351\text{e-}38$ 至 $\pm 3.402823466\text{e}+38$		X
	实型	$\pm 2.2250738585072014\text{e-}308$ 至 $\pm 1.7976931348623158\text{e}+308$		X

■ 操作数设置

下面描述了 S2H/S2HP 指令操作数 (D1) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器地址	位			X
	字	仅按字指定 (例如, [PLC1]D0000)		X
内部地址	位			X
	字	仅按字指定 (例如, [#INTERNAL]LS0000)		X
符号	位			X
	字			X
变量格式	位	指定一个位		X
		指定位数组 ([常量])		X
		指定位数组 ([变量])		X
	整型 (不包括 I/O)	未指定数组和修饰符		X
		指定整型变量 [常量]		X
		指定整型变量 [变量]		X
		指定整型变量 [常量 / 变量] .B/W [常量 / 变量]		X
	浮点型			X
		指定浮点型变量 [常量]		X
		指定浮点型变量 [变量]		X
	实型			X
		指定实型变量 [常量]		X
		指定实型变量 [变量]		X
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	非 .HR / .MIN / .SEC	1	O
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用：O 不可用：X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符		X
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	非 .HR / .MIN / .SEC	1	O
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
常量	整型	-2147483648 至 2147483647		X
	浮点型	± 1.175494351e-38 至 ± 3.402823466e+38		X
	实型	± 2.2250738585072014e-308 至 ± 1.7976931348623158e+308		X

■ S2H/S2HP 指令说明

S2H/S2HP 指令将整型变量转换为时间变量中的秒。在 S1 中指定您想转换的整型变量并为 D1 中的转换输出结果指定时间变量。您只能为 S1 中的输入指定整型变量，为 D1 中的输出指定时间变量。不能用数组配置时间变量。0:30 将被转换成 1800 秒。

14:00:00 将被转换成 50400 秒。

■ 显示执行结果的系统变量

当执行结果是 0 时，#L_CalcZero 置 ON。

当执行发生错误时，将在 #L_CalcErrCode 中保存错误代码。

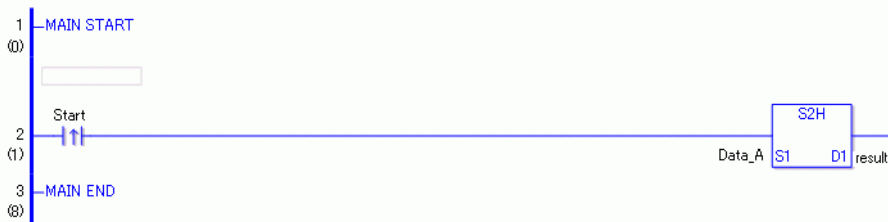
(注释)

当使用系统变量检查结果时，确保在执行完指令后进行。

当在执行完多个指令后检查状态时，系统变量将只保存最后处理的指令的结果。

程序示例

S2H



(1)当上升沿指令置 ON 时，将执行 S2H 指令。当执行 S2H 指令时，会将 Data_A 的 S2H 转换结果保存在 D1 中。

当使用常开指令时，只要常开指令变量保持为 ON，就总是执行 S2H 指令。

程序示例

S2HP

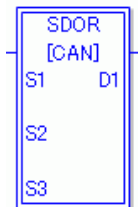
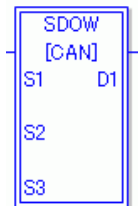
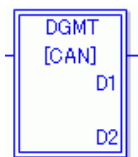


(1)S2HP 和 S2H 指令检测执行时间的方法不同。对于 S2HP 指令，即使使用常开指令，也只有在检测到上升沿时才执行 S2HP 指令。即便常开指令变量保持为 ON，也只执行一次 S2HP 指令 (在 1 次扫描中执行)。

31.24 I/O 驱动程序指令

31.24.1 SDOR, SDOW, DGMT, DGSL(CANopen 驱动程序)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
SDOR		定义的节点 读取对象字典	9 至 21
SDOW		定义的节点 写入对象字典	9 至 21
DGMT		读取主站状态	5 至 9
DGSL		读取从站状态	5 至 9

■ 操作数设置

下面描述了操作数 (S1 至 S10 和 D1 至 D5) 的可指定内容。

I/O 驱动程序指令中的步数取决于指定方法及使用的操作数数量。下面描述如何计算步数。

3 + 操作数 S1 中的步数 + ... + 操作数 S10 中的步数 + 操作数 D1 中的步数 + ... + 操作数 D5 中的步数 = 一个指令中的总步数

注 释

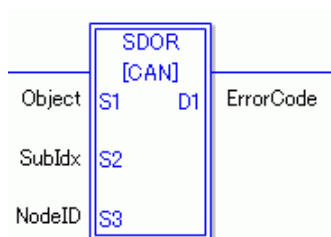
- 有关各个操作数的更多信息，请参阅下面的内容。

☞ "30.7.4 使用 I/O 驱动程序指令" (p30-150)

例如，计算 SDOR, SDOW, DGMT 和 DGSL 中的步数

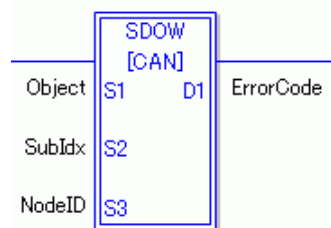
(有关操作数步数的信息，请参阅下一页中的操作数设置。)

SDOR



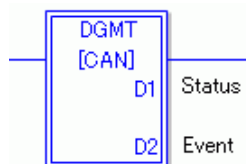
3 步 + {Object = 1 步} + {SubIdx = 1 步} + {NodeID = 1 步} + {Length = 1 步} + {Offset = 1 步} + {ErrorCode = 1 步} = 9 步

SDOW



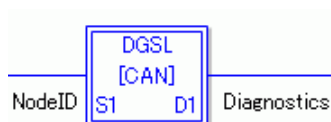
3 步 + {Object = 1 步} + {SubIdx = 1 步} + {NodeID = 1 步} + {Length = 1 步} + {Offset = 1 步} + {ErrorCode = 1 步} = 9 步

DGMT



3 步 + {Status = 1 步} + {Event = 1 步} = 5 步

DGSL



3 步 + {NodeID = 1 步} + {Diagnostics = 1 步} = 5 步

前三步是所有 I/O 驱动程序指令都要求的步数。一定要为 I/O 驱动程序指令加上这三步。

■ 操作数设置

下面描述了操作数 (S1 至 S10 和 D1 至 D5) 的可指定内容。

名称	类型	条件	操作数 步数	可用: O 不可用: X
外部寄存器 地址	位			X
	整型			X
内部地址	位			X
	整型			X
符号	位			X
	整型			X
变量格式 * (注释 1) Sx= 不允许 输入和输出 Dx= 不允许 输入和输出	位			X
	整型 * (注释 1)	未指定数组和修饰符	1	O
		指定整型变量 [常量]	2	O
		指定整型变量 [变量]	3	O
		指定整型变量 B/W [变量] 指定整型变量 B/W [常量]		X
	浮点型			X
	实型			X
	定时器	仅 .PT / .ET		X
	计数器	仅 .PV / .CV		X
	日期	仅 .YR / .MO / .DAY		X
	时间	仅 .HR / .MIN / .SEC		X
	PID	仅 .KP / .TR / .TD / .PA / .BA / .ST		X
地址格式	X_			X
	Y_			X
	M_			X
	I_			X
	Q_			X
	D_	未指定修饰符	1	O
		D_****.B/W [常量]		X
		D_****.B/W [地址]		X
	F_			X
	R_			X
	T_	仅 .PT / .ET		X
	C_	仅 .PV / .CV		X
	N_	仅 .YR / .MO / .DAY		X
	J_	仅 .HR / .MIN / .SEC		X
	U_	仅 .KP / .TR / .TD / .PA / .BA / .ST		X

名称	类型	条件	操作数 步数	可用: O 不可用: X
常量	整型	2147483648 至 2147483647		X
	浮点型	$\pm 1.17549435138 \text{ e-}38$ 至 $\pm 3.402823466\text{e}+38$		X
	实型	$\pm 2.2250738585072014\text{e-}308$ 至 $\pm 1.7976931348623158\text{e}+308$		X

■ SDOR、SDOW、DGMT 和 DGSL 指令说明

- 从 I/O 驱动程序中，SDOR 指令运行在源操作数中定义的 SDO 命令 (读)。在 I/O 驱动程序完成 SDO 命令后，将执行结果设置到目标操作数。
- 从 I/O 驱动程序中，SDOW 指令运行在源操作数中定义的 SDO 命令 (写)。在 I/O 驱动程序完成 SDO 命令后，将执行结果设置到目标操作数。
- 从 I/O 驱动程序中，DGMT 指令读取主站诊断结果。I/O 驱动程序然后将读取结果设置到目标操作数。
- 从 I/O 驱动程序中，DGSL 指令读取在源操作数中定义的从站诊断结果。I/O 驱动程序然后将读取结果设置到目标操作数。
- 指令在接通时执行。指令执行完成后，指令维持接通一次扫描周期的时间。

注 释

- 有关 SDOR、SDOW、DGMT 和 DGSL 指令 (CANopen 驱动程序) 的更多信息，请参阅以下内容。

☞ "30.7 用 CANopen 控制外部 I/O" (p30-141)

■ 设置指南

- 如果在未设置 CANopen 驱动程序的情况下使用这些指令，或当用无效的数据类型设置操作数时，会发生错误。
- 最多可使用 15 个 I/O 驱动程序指令，包括使用其他 I/O 驱动程序。
- 这些指令只能用在 MAIN 和 SUB 程序中。不能用在 INIT 中。
- 不支持只能在启动时执行的指令，如 MOVP。要想只在启动时执行指令，请使用 PT 指令。

31.24.2 PLSX(STD 驱动程序)

符号和功能

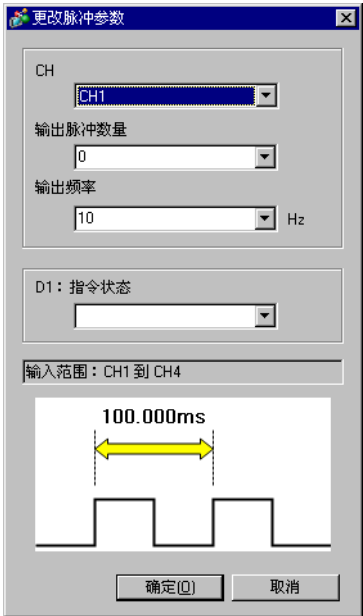
梯形图指令名称	梯形图符号	功能	步数
PLSX		更改脉冲输出参数	5 至 11

■ 操作数设置

双击 PLSX 指令，显示如下对话框。在如下对话框中，指定各项设置。

注 释

- 有关 LT 的特殊 I/O 的详细信息，请参阅以下内容。
☞ "30.5 控制 LT 的外部 I/O" (p30-23)



设置	操作数	描述	设置范围
CH	S1	指定通道号，将为该通道分配要更改参数的脉冲输出。	CH1 - CH4
输出脉冲数量	S2	为要更改参数的脉冲输出指定脉冲数 (次数)。这里指定的值会被保存在 #L_PLS*_NUM 中。	• 数值 0 至 2147483647 • 变量 仅整型变量
输出频率	S3	指定参数将被更改的输出频率 (Hz)。这里指定的值被保存在 #L_PLS*_LHZ 中。	• 数值 10 至 65000 • 变量 仅整型变量
D1：指令状态	D1	指定用于保存指令执行状态的整型变量。	仅整型变量

■ PLSX 指令说明

指令执行状态



注 释

- 可以用系统变量 #L_ExIOSpParmErr(特殊 I/O 参数错误) 来查看错误状态。

☞ "30.5.2 映射 I/O 端子 (通用) ◆ 特殊 I/O 参数错误 (#L_ExIOSpParmErr)" (p30-35)

启用条件

- 当指令正在执行且参数已被更改时。
- 如果未发生错误 (请参阅以下注释)。

禁用条件

- 当指令正在执行且参数已被更改时，参数更改完成标志置 OFF。
- * 当禁用时，操作数 D1 被清 0。

注意

- 在执行指令时，会确认参数是否正被更改。如果是，就会发生错误，且不再进行任何处理。
- 当参数更改出现问题时，就变成一个错误。
- 当操作数 D1 不能被指令识别时，将发生错误。
- 当使用脉冲来执行指令时 (例如，PT 指令)，需要将指令置 OFF，然后再置 ON 才能重新执行指令。
- 执行指令后，它将处于运行状态。
- 如果指令处于运行状态，无论启用 / 禁用条件如何，都会持续执行该指令。

31.24.3 PLSXY(STD 驱动程序)

符号和功能

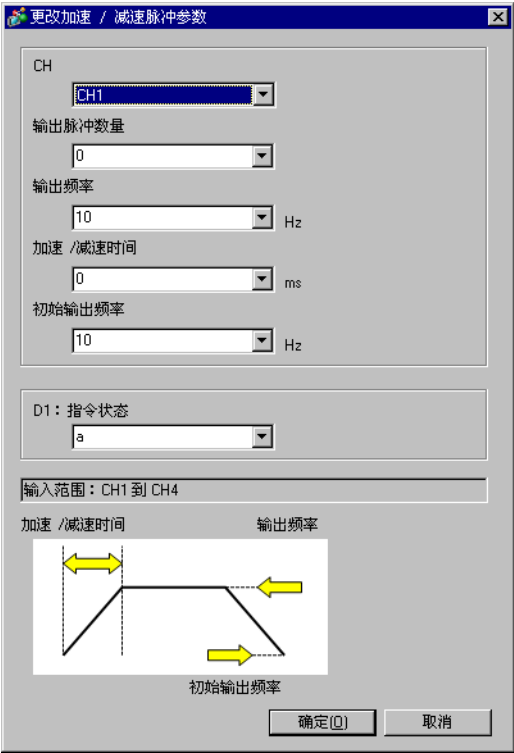
梯形图指令名称	梯形图符号	功能	步数
PLSY		更改加速 / 减速脉冲输出参数	7 至 17

■ 操作数设置

双击 PLSXY 指令，显示如下对话框。在如下对话框中，指定各项设置。

注 释

- 有关 LT 的特殊 I/O 的详细信息，请参阅以下内容。
☞ "30.5 控制 LT 的外部 I/O" (p30-23)



设置	操作数	描述	设置范围
CH	S1	指定通道号，将为该通道分配要更改参数的脉冲输出。	CH1 - CH4
输出脉冲数量	S2	指定要更改的脉冲输出数量 (次数)。这里指定的值会被保存在 #L_PLS*_NUM 中。	- 数值 0 至 2147483647 - 变量 仅整型变量

设置	操作数	描述	设置范围
输出频率	S3	指定要更改的输出频率 (Hz)。这里指定的值被保存在 #L_PLS*_LHZ 中。	• 数值 10 至 65000 • 变量 仅整型变量
加速 / 减速时间	S4	指定要更改的加速 / 减速时间 (ms)。这里指定的值被保存在 #L_PLS*_ACC 中。	• 数值 0 至 65535 • 变量 仅整型变量
初始输出频率	S5	指定要更改的初始输出频率 (Hz)。这里指定的值被保存在 #L_PLS*_SHZ 中。	• 数值 10 至 65000 • 变量 仅整型变量
D1: 指令状态	D1	指定用于保存指令执行状态的整型变量。	仅整型变量

■ PLSY 指令说明

指令执行状态



注 释

- 可以用系统变量 #L_ExIOSpParmErr(特殊 I/O 参数错误) 来查看错误状态。

☞ "30.5.2 映射 I/O 端子 (通用) ◆ 特殊 I/O 参数错误 (#L_ExIOSpParmErr)" (p30-35)

启用条件

- 当指令正在执行且参数已被更改时。
- 如果未发生错误 (请参阅以下注释)。

禁用条件

- 当指令正在执行且参数已被更改时，参数更改完成标志置 OFF。
- * 当禁用时，操作数 D1 被清 0。

注意

- 当输出加速 / 减速脉冲时，您不能更改参数。如果在输出加速 / 减速脉冲的同时执行指令，就会发生错误。
- 在执行指令时，会确认参数是否正被更改。如果是，就会发生错误，且不再进行任何处理。
- 当参数更改出现问题时，就会发生错误。
- 当操作数 D1 不能被指令识别时，将发生错误。
- 当使用脉冲来执行指令时 (例如，PT 指令)，需要将指令置 OFF，然后再置 ON 才能重新执行指令。
- 执行指令后，它将处于运行状态。
- 如果指令处于运行状态，无论启用 / 禁用条件如何，都会持续执行该指令。

31.24.4 PLSG(STD 驱动程序)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
PLSG		读取脉冲输出参数	8 至 20

■ 操作数设置

双击 PLSG 指令，显示如下对话框。在如下对话框中，指定各项设置。

注 释	有关 LT 的特殊 I/O 的详细信息，请参阅以下内容。 ☞ "30.5 控制 LT 的外部 I/O" (p30-23)
----------------	---

读取脉冲参数

CH

CH2

脉冲输出模式

1

D1: 指令状态

输出脉冲数量

输出频率

 Hz

加速 / 减速时间

 ms

初始输出频率

 Hz

常量: 1 (脉冲) 3 (加速 / 减速脉冲)

变量: 仅整型变量

确定(O)

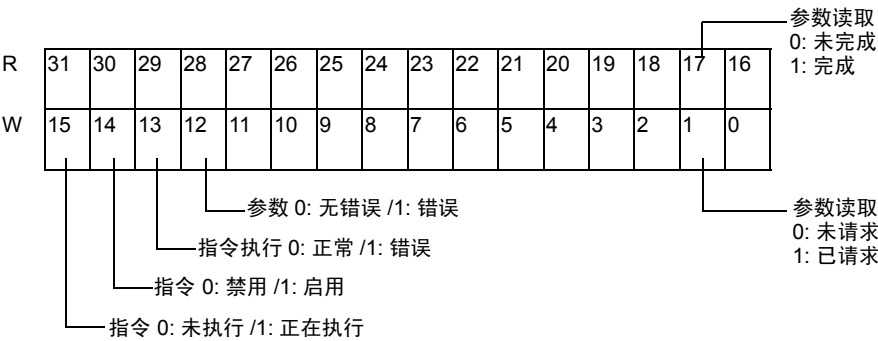
取消

设置	操作数	描述	设置范围
CH	S1	指定通道号， 将为该通道分配要读取参数的脉冲输出。	CH1 - CH4
脉冲输出模式	S2	指定输出脉冲模式 (正常或加速 / 减速)。这里指定的值被保存在 #L_ExIOSPCtrl 的控制部分。	- 常量 1 (PLS) 或 3(加速 / 减速 PLS) - 变量 仅整型变量
D1: 指令状态	D1	指定用于保存指令执行状态的整型变量。	仅整型变量
输出脉冲数量	D2	指定变量， 用于保存要更改的脉冲输出数量 (次数)。 #L_PLS*_NUM 的值被保存在这里指定的变量中。	仅整型变量

设置	操作数	描述	设置范围
输出频率	D3	指定变量，用于保存要更改的脉冲输出频率 (Hz)。#L_PLS*_LHZ 的值被保存在这里指定的变量中。	仅整型变量
加速 / 减速时间	D4	指定变量，用于保存要更改的加速 / 减速时间 (ms)。#L_PLS*_ACC 的值被保存在这里指定的变量中。	仅整型变量
初始输出频率	D5	指定变量，用于保存要更改的初始脉冲输出频率 (Hz)。#L_PLS*_SHZ 的值被保存在这里指定的变量中。	仅整型变量

■ PLSG 指令说明

指令执行状态



注 释

- 可以用系统变量 #L_ExIOSpParmErr(特殊 I/O 参数错误) 来查看错误状态。

☞ "30.5.2 映射 I/O 端子 (通用) ◆ 特殊 I/O 参数错误 (#L_ExIOSpParmErr)" (p30-35)

启用条件

- 当指令处于运行状态且参数读取已经完成时
- 如果未发生错误 (请参阅以下注释)。

禁用条件

- 当指令处于运行状态且已经确认参数读取完成标志被清除时。
(参数读取请求被取消, 参数读取完成标志被清除。)
* 当禁用时, 操作数 D1 被清 0。

注意

- 在执行指令时, 会确认是否正在读取参数。如果是, 就会发生错误, 且不再进行任何处理。
- 当操作数 D1 不能被指令识别时, 将发生错误。
- 当使用脉冲来执行指令时 (例如, PT 指令), 需要将指令置 OFF, 然后再置 ON 才能重新执行指令。
- 如果脉冲输出模式是 “正常”, 则设置在操作数 D2(输出脉冲计数) 和 D3(输出频率) 中读取的数据。将不更新操作数 D4(加速 / 减速时间) 和 D5(初始输出频率) 的数据。
- 如果脉冲输出模式是 “加速 / 减速”, 则设置在操作数 D2 至 D5 中读取的数据。
- 执行指令后, 它将处于运行状态。
- 如果指令处于运行状态, 无论启用 / 禁用条件如何, 都会持续执行该指令。

31.24.5 PLS(STD 驱动程序)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
PLS		启动脉冲输出	4 至 8

■ 操作数设置

双击 PLS 指令，显示如下对话框。在如下对话框中，指定各项设置。

注 释

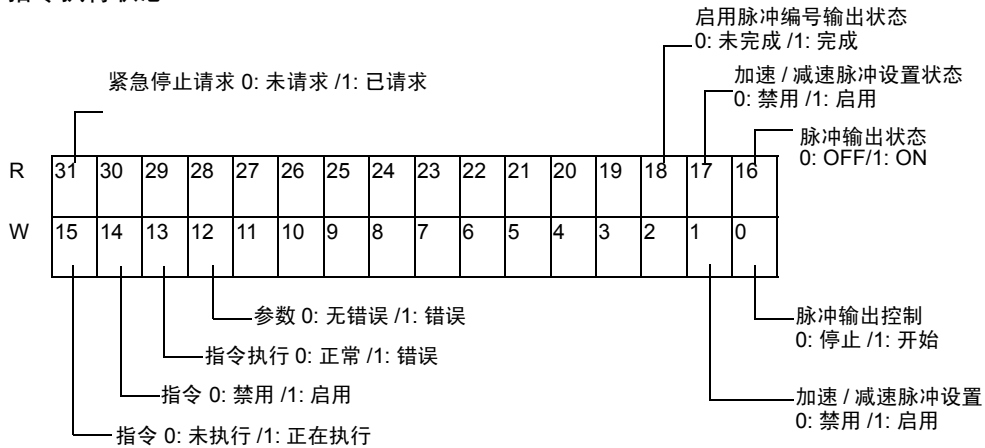
- 有关 LT 的特殊 I/O 的详细信息，请参阅以下内容。
"30.5 控制 LT 的外部 I/O" (p30-23)



设置	操作数	描述	设置范围
CH	S1	指定通道号，将为该通道分配要更改参数的脉冲输出。 这里指定的通道号的脉冲输出开始。	CH1 - CH4
脉冲输出模式	S2	指定输出脉冲模式（正常或加速 / 减速）。 这里指定的值被保存在 #L_ExIOSPCtrl 的控制部分。	- 常量 1 (PLS) 或 3(加速 / 减速 PLS) - 变量 仅整型变量
D1: 指令状态	D1	指定用于保存指令执行状态的整型变量。	仅整型变量

■ PLS 指令说明

指令执行状态



注 释

- 可以用系统变量 #L_ExIOSpParmErr(特殊 I/O 参数错误) 来查看错误状态。

☞ "30.5.2 映射 I/O 端子 (通用) ◆ 特殊 I/O 参数错误 (#L_ExIOSpParmErr)" (p30-35)

启用条件

- 当指令处于运行状态且确认脉冲输出正在执行时
- 指令处于运行状态，当检测到紧急停止请求且指令停止时
- 如果未发生错误 (请参阅以下注释)。

禁用条件

- 当确认了已设定的脉冲计数输出完成时
(脉冲输出停止，已设定的脉冲计数输出完成标志被清除)
- 当执行了紧急停止请求且确认停止时
* 当禁用时，操作数 D1 被清 0。

注意

- 在执行指令时，会确认是否正在输出脉冲。如果是，就会发生错误，且不再进行任何处理。
- 对于加速 / 减速脉冲，确认正在请求创建加速 / 减速表 (请求和完成位不是 0)，并且有加速 / 减速表。当正在创建加速 / 减速表或没有加速 / 减速表时，就会发生错误。
- 当操作数 D1 不能被指令识别时，将发生错误。
- 当使用脉冲来执行指令时 (例如，PT 指令)，需要将指令置 OFF，然后再置 ON 才能重新执行指令。
- 执行指令后，它将处于运行状态。
- 如果指令处于运行状态，无论启用 / 禁用条件如何，都会持续执行该指令。

31.24.6 PLSQ(STD 驱动程序)

符号和功能

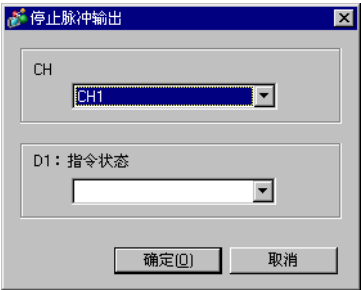
梯形图指令名称	梯形图符号	功能	步数
PLSQ		停止脉冲输出	3 至 5

■ 操作数设置

双击 PLSQ 指令，显示如下对话框。在如下对话框中，指定各项设置。

注 释

- 有关 LT 的特殊 I/O 的详细信息，请参阅以下内容。
"30.5 控制 LT 的外部 I/O" (p30-23)



设置	操作数	描述	设置范围
CH	S1	指定通道号，将为该通道分配要更改参数的脉冲输出。 这里指定的通道号的脉冲输出停止。（仅适用于 PLS 指令的事件发生通知）	CH1 - CH4
D1: 指令状态	D1	指定用于保存指令执行状态的整型变量。 这里一定要使用与 PLS D1 相同的变量： 指令执行状态	仅整型变量

■ PLSQ 指令说明

指令执行状态



注 释

- 可以用系统变量 #L_ExIOSpParmErr(特殊 I/O 参数错误) 来查看错误状态。

☞ "30.5.2 映射 I/O 端子 (通用) ◆ 特殊 I/O 参数错误 (#L_ExIOSpParmErr)" (p30-35)

启用条件

- 当能流为 ON 时

禁用条件

- 当能流为 OFF 时

注意

- 只会将紧急停止请求通知给 PLS 指令。脉冲输出的停止控制用 PLS 指令执行。

31.24.7 PWMX(STD 驱动程序)

符号和功能

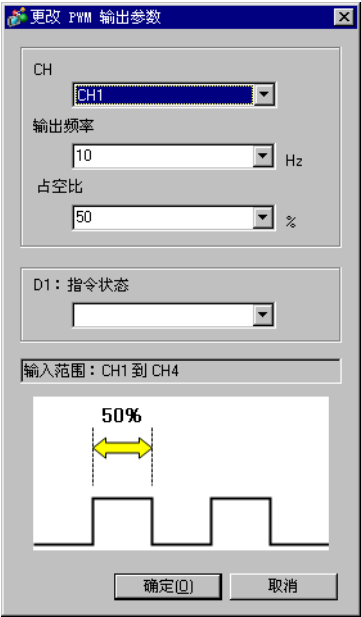
梯形图指令名称	梯形图符号	功能	步数
PWMX		更改 PWM 输出参数	5 至 11

■ 操作数设置

双击 PWMX 指令，显示如下对话框。在如下对话框中，指定各项设置。

注 释

- 有关 LT 的特殊 I/O 的详细信息，请参阅以下内容。
☞ "30.5 控制 LT 的外部 I/O" (p30-23)



设置	操作数	描述	设置范围
CH	S1	指定通道号，将为该通道分配要更改参数的 PWM 输出。	CH1 - CH4
输出频率	S2	指定要更改的输出频率 (Hz)。这里指定的值被保存在 #L_PWM*_WHZ 中。	- 数值 10 至 65000 - 变量 仅整型变量
占空比	S3	指定要更改的 PWM 占空比值 (%)。这里指定的值被保存在 #L_PWM*_DTY 中。有关占空比值有效范围的更多信息，请参阅以下内容。 ☞ "30.5.9 PWM输出 ◆ 占空比值有效范围" (p30-84)	- 数值 0 至 100 - 变量 仅整型变量
D1: 指令状态	D1	指定用于保存指令执行状态的整型变量。	仅整型变量

■ PWMX 指令说明

指令执行状态



注 释

- 可以用系统变量 #L_ExIOSpParmErr(特殊 I/O 参数错误) 来查看错误状态。

☞ "30.5.2 映射 I/O 端子 (通用) ◆ 特殊 I/O 参数错误 (#L_ExIOSpParmErr)" (p30-35)

启用条件

- 当指令正在执行且参数已被更改时。
- 如果未发生错误 (请参阅以下注释)。

禁用条件

- 当指令正在执行且参数已被更改时，参数更改完成标志置 OFF。
- * 当禁用时，操作数 D1 被清 0。

注意

- 在执行指令时，会确认参数是否正被更改。如果是，就会发生错误，且不再进行任何处理。
- 当参数更改出现问题时，就会发生错误。
- 当操作数 D1 不能被指令识别时，将发生错误。
- 当使用脉冲来执行指令时 (例如，PT 指令)，需要将指令置 OFF，然后再置 ON 才能重新执行指令。
- 执行指令后，它将处于运行状态。
- 如果指令处于运行状态，无论启用 / 禁用条件如何，都会持续执行该指令。

31.24.8 PWMG(STD 驱动程序)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
PWMG		读取 PWM 输出参数	5 至 11

■ 操作数设置

双击 PWMG 指令，显示如下对话框。在如下对话框中，指定各项设置。

注 释

- 有关 LT 的特殊 I/O 的详细信息，请参考以下内容。
☞ "30.5 控制 LT 的外部 I/O" (p30-23)



设置	操作数	描述	设置范围
CH	S1	指定通道号，将为该通道分配要读取参数的 PWM 输出。	CH1 - CH4
D1: 指令状态	D1	指定用于保存指令执行状态的整型变量。	仅整型变量
输出频率	D2	指定变量，用于保存要更改的脉冲输出频率 (Hz)。#L_PWM*_WHZ 的值被保存在这里指定的变量中。	仅整型变量
占空比	D3	指定变量，用于保存要更改的 PWM 占空比值 (%)。#L_PWM*_DTY 的值被保存在这里指定的变量中。	仅整型变量

■ PWMG 指令说明

指令执行状态



注 释

- 可以用系统变量 #L_ExIOSpParmErr(特殊 I/O 参数错误) 来查看错误状态。

☞ "30.5.2 映射 I/O 端子 (通用) ◆ 特殊 I/O 参数错误 (#L_ExIOSpParmErr)" (p30-35)

启用条件

- 当指令处于运行状态且参数读取已经完成时
- 如果未发生错误 (请参阅以下注释)。

禁用条件

- 当指令处于运行状态且已经确认参数读取完成标志被清除时。
(参数读取请求被取消, 参数读取完成标志被清除。)
- * 当禁用时, 操作数 D1 被清 0。

注意

- 在执行指令时, 会确认是否正在读取参数。如果正在读取, 就会发生错误, 且不再进行任何处理。
- 当操作数 D1 不能被指令识别时, 将发生错误。
- 当使用脉冲来执行指令时 (例如, PT 指令), 需要将指令置 OFF, 然后再置 ON 才能重新执行指令。
- 执行指令后, 它将处于运行状态。
- 如果指令处于运行状态, 无论启用 / 禁用条件如何, 都会持续执行该指令。

31.24.9 PWM(STD 驱动程序)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
PWM		开始 PWM 输出	3 至 5

■ 操作数设置

双击 PWM 指令，显示如下对话框。在如下对话框中，指定各项设置。

注 释

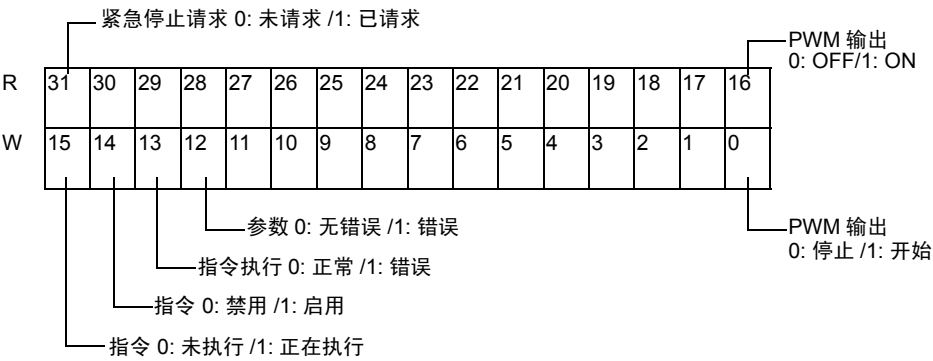
- 有关 LT 的特殊 I/O 的详细信息，请参考以下内容。
☞ "30.5 控制 LT 的外部 I/O" (p30-23)



设置	操作数	描述	设置范围
CH	S1	指定通道号，将为该通道分配要更改参数的 PWM 输出。 这里指定的通道号的 PWM 输出开始。	CH1 - CH4
D1: 指令状态	D1	指定用于保存指令执行状态的整型变量。	仅整型变量

■ PWM 指令说明

指令执行状态



注 释

- 可以用系统变量 #L_ExIOSpParmErr(特殊 I/O 参数错误) 来查看错误状态。

☞ "30.5.2 映射 I/O 端子 (通用) ◆ 特殊 I/O 参数错误 (#L_ExIOSpParmErr)" (p30-35)

启用条件

- 当指令处于运行状态且确认 PWM 输出正在执行时
- 指令处于运行状态, 当检测到紧急停止请求且指令停止时
- 如果未发生错误 (请参阅以下注释)。

禁用条件

- 当执行了紧急停止请求且确认停止时
- * 当禁用时, 操作数 D1 被清 0。

注意

- 在执行指令时, 会确认是否正在输出 PWM。如果是, 就会发生错误, 且不再进行任何处理。
- 当操作数 D1 不能被指令识别时, 将发生错误。
- 当使用脉冲来执行指令时 (例如, PT 指令), 需要将指令置 OFF, 然后再置 ON 才能重新执行指令。
- 执行指令后, 它将处于运行状态。
- 如果指令处于运行状态, 无论启用 / 禁用条件如何, 都会持续执行该指令。

31.24.10 PWMQ(STD 驱动程序)

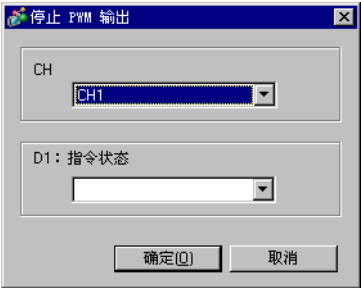
符号和功能

梯形图指令名称	梯形图符号	功能	步数
PWMQ		停止 PWM 输出	3 至 5

■ 操作数设置

双击 PWMQ 指令，显示如下对话框。在如下对话框中，指定各项设置。

注 释	有关 LT 的特殊 I/O 的详细信息，请参考以下内容。 "30.5 控制 LT 的外部 I/O" (p30-23)
----------------	---



设置	操作数	描述	设置范围
CH	S1	指定通道号，将为该通道分配要更改参数的 PWM 输出。 这里指定的通道号的 PWMQ 输出停止。 (仅适用于 PWM 指令的事件发生通知)	CH1 - CH4
D1: 指令状态	D1	指定用于保存指令执行状态的整型变量。 这里一定要使用与 PWM D1 相同的变量： 指令执行状态	仅整型变量

■ PWMQ 指令说明

指令执行状态

紧急停止请求 0: 未请求 / 1: 已请求

R	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
W	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

注 释

- 可以用系统变量 #L_ExIOSpParmErr(特殊 I/O 参数错误) 来查看错误状态。

☞ "30.5.2 映射 I/O 端子 (通用) ◆ 特殊 I/O 参数错误 (#L_ExIOSpParmErr)" (p30-35)

启用条件

- 当能流为 ON 时

禁用条件

- 当能流为 OFF 时

注意

- 只会将紧急停止请求通知给 PWM 指令。PWM 输出的停止控制用 PWM 指令执行。

31.24.11 HSCX(STD 驱动程序)

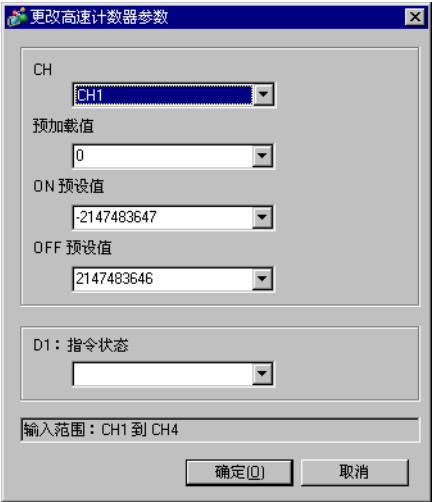
符号和功能

梯形图指令名称	梯形图符号	功能	步数
HSCX		更改高速计数器参数	6 至 14

■ 操作数设置

双击 HSCX 指令，显示如下对话框。在如下对话框中，指定各项设置。

注 释	有关 LT 的特殊 I/O 的详细信息，请参考以下内容。 ☞ "30.5 控制 LT 的外部 I/O" (p30-23)
----------------	---



设置	操作数	描述	设置范围
CH	S1	指定通道号，将为该通道分配要更改参数的高速计数器。	CH1 - CH4
预加载值	S2	指定要更改其参数的高速计数器的预加载值。这里指定的值被保存在 #L_HSC*_PLV 中。	- 数值 -2147483648 至 2147483647 - 变量 仅整型变量
ON 预设值	S3	指定要更改其参数的高速计数器的 ON 预设值。这里指定的值被保存在 #L_HSC*_ONP 中。 注 释 - ON 预设值和 OFF 预设值必须不同。 - 不能将低 16 位为 0xFFFF 或 0x0000 的值设置为预设值。	- 数值 -2147483647 至 2147483646 - 变量 仅整型变量

设置	操作数	描述	设置范围
OFF 预设值	S4	指定要更改其参数的高速计数器的 OFF 预设值。这里指定的值被保存在 #L_HSC*_OFF 中。 注 释 - ON 预设值和 OFF 预设值必须不同。 - 不能将低 16 位为 0xFFFF 或 0x0000 的值设置为预设值。	- 数值 -2147483647 至 2147483646 - 变量 仅整型变量
D1: 指令状态	D1	指定用于保存指令执行状态的整型变量。	仅整型变量

■ HSCX 指令说明

指令执行状态



注 释

- 可以用系统变量 #L_ExIOSpParmErr(特殊 I/O 参数错误) 来查看错误状态。

☞ "30.5.2 映射 I/O 端子 (通用) ◆ 特殊 I/O 参数错误 (#L_ExIOSpParmErr)" (p30-35)

启用条件

- 当指令正在执行且参数已被更改时。
- 如果未发生错误 (请参阅以下注释)。

禁用条件

- 当指令正在执行且参数已被更改时，参数更改完成标志置 OFF。
- * 当禁用时，操作数 D1 被清 0。

注意

- 在执行指令时，会确认参数是否正被更改。如果是，就会发生错误，且不再进行任何处理。
- 当参数更改出现问题时，就会发生错误。
- 当操作数 D1 不能被指令识别时，将发生错误。
- 当使用脉冲来执行指令时 (例如，PT 指令)，需要将指令置 OFF，然后再置 ON 才能重新执行指令。
- 执行指令后，它将处于运行状态。
- 如果指令处于运行状态，无论启用 / 禁用条件如何，都会持续执行该指令。

31.24.12 HSCG(STD 驱动程序)

符号和功能

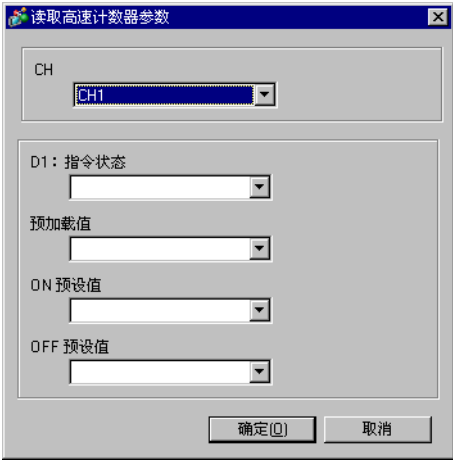
梯形图指令名称	梯形图符号	功能	步数
HSCG		读取高速计数器参数	7 至 17

■ 操作数设置

双击 HSCG 指令，显示如下对话框。在如下对话框中，指定各项设置。

注 释

- 有关 LT 的特殊 I/O 的详细信息，请参考以下内容。
☞ "30.5 控制 LT 的外部 I/O" (p30-23)



设置	操作数	描述	设置范围
CH	S1	指定通道号，将为该通道分配要读取其参数的高速计数器。	CH1 - CH4
D1: 指令状态	D1	指定用于保存指令执行状态的整型变量。	仅整型变量
预加载值	D2	指定变量，用于保存要更改其参数的高速计数器的预加载值。#L_HSC*_PLV 的值被保存在这里指定的变量中。	仅整型变量
ON 预设值	D3	指定变量，用于保存要更改其参数的高速计数器的 ON 预设值。#L_HSC*_ONP 的值被保存在这里指定的变量中。	仅整型变量
OFF 预设值	D4	指定变量，用于保存要更改其参数的高速计数器的 OFF 预设值。#L_HSC*_OFP 的值被保存在这里指定的变量中。	仅整型变量

■ HSCG 指令说明

指令执行状态



注 释

- 可以用系统变量 #L_ExIOSpParmErr(特殊 I/O 参数错误) 来查看错误状态。
- ☞ "30.5.2 映射 I/O 端子 (通用) ◆ 特殊 I/O 参数错误 (#L_ExIOSpParmErr)" (p30-35)

启用条件

- 当指令处于运行状态且参数读取已经完成时
- 如果未发生错误 (请参阅以下注释)。

禁用条件

- 当指令处于运行状态且已经确认参数读取完成标志被清除时。
(参数读取请求被取消, 参数读取完成标志被清除。)
- * 当禁用时, 操作数 D1 被清 0。

注意

- 在执行指令时, 会确认是否正在读取参数。如果正在读取, 就会发生错误, 且不再进行任何处理。
- 当操作数 D1 不能被指令识别时, 将发生错误。
- 当使用脉冲来执行指令时 (例如, PT 指令), 需要将指令置 OFF, 然后再置 ON 才能重新执行指令。
- 执行指令后, 它将处于运行状态。
- 如果指令处于运行状态, 无论启用 / 禁用条件如何, 都会持续执行该指令。

31.24.13 HSC(STD 驱动程序)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
HSC		启动高速计数器	4 至 8

■ 操作数设置

双击 HSC 指令，显示如下对话框。在如下对话框中，指定各项设置。

注 释

- 有关 LT 的特殊 I/O 的详细信息，请参考以下内容。
"30.5 控制 LT 的外部 I/O" (p30-23)



设置	操作数	描述	设置范围
CH	S1	指定通道号，将为该通道分配要更改参数的高速计数器。 这里指定的通道号的高速计数器启动。	CH1 - CH4
同步输出设置	S2	指定高速计数器模式 (同步输出 On/Off)。这里指定的值被保存在 #L_ExIOSPCtrl 的控制部分。	- 常量 1 (同步输出 Off) 或 3 (同步输出 On) - 变量 仅整型变量
D1: 指令状态	D1	指定用于保存指令执行状态的整型变量。	仅整型变量

■ HSC 指令说明

指令执行状态



注 释

- 可以用系统变量 #L_ExIOSpParmErr(特殊 I/O 参数错误) 来查看错误状态。

☞ "30.5.2 映射 I/O 端子 (通用) ◆ 特殊 I/O 参数错误 (#L_ExIOSpParmErr)" (p30-35)

启用条件

- 当指令处于运行状态且确认高速计数器正在执行时
- 指令处于运行状态，当检测到紧急停止请求且指令停止时
- 如果未发生错误 (请参阅以下注释)。

禁用条件

- 当执行了紧急停止请求且确认停止时
- * 当禁用时，操作数 D1 被清 0。

注意

- 执行指令时，会确认高速计数器是否正在执行。如果是，就会发生错误，且不再进行任何处理。
- 当操作数 D1 不能被指令识别时，将发生错误。
- 当使用脉冲来执行指令时 (例如，PT 指令)，需要将指令置 OFF，然后再置 ON 才能重新执行指令。
- 执行指令后，它将处于运行状态。
- 如果指令处于运行状态，无论启用 / 禁用条件如何，都会持续执行该指令。

31.24.14 HSCQ(STD 驱动程序)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
HSCQ		停止高速计数器	3 至 5

■ 操作数设置

双击 HSCQ 指令，显示如下对话框。在如下对话框中，指定各项设置。

注 释

- 有关 LT 的特殊 I/O 的详细信息，请参考以下内容。
☞ "30.5 控制 LT 的外部 I/O" (p30-23)



设置	操作数	描述	设置范围
CH	S1	指定通道号，将为该通道分配要更改参数的高速计数器。这里指定的通道号的高速计数器停止。（仅适用于 HSC 指令的事件发生通知）	CH1 - CH4
D1: 指令状态	D1	指定用于保存指令执行状态的整型变量。这里一定要使用与 HSC D1 相同的变量：指令执行状态	仅整型变量

■ HSCQ 指令说明

指令执行状态



注 释

- 可以用系统变量 #L_ExIOSpParmErr(特殊 I/O 参数错误) 来查看错误状态。

☞ "30.5.2 映射 I/O 端子 (通用) ◆ 特殊 I/O 参数错误 (#L_ExIOSpParmErr)" (p30-35)

启用条件

- 当能流为 ON 时

禁用条件

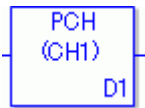
- 当能流为 OFF 时

注意

- 只会将紧急停止请求通知给 HSC 指令。高速计数器的停止控制用 HSC 指令执行。

31.24.15 PCH(STD 驱动程序)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
PCH		确认脉冲捕捉输入	3 至 5

■ 操作数设置

双击 PCH 指令，显示如下对话框。在如下对话框中，指定各项设置。

注 释

- 有关 LT 的特殊 I/O 的详细信息，请参考以下内容。
☞ "30.5 控制 LT 的外部 I/O" (p30-23)



设置	操作数	描述	设置范围
CH	S1	指定通道号，将为该通道分配要更改参数的脉冲捕捉输入。 针对这里指定的通道号的“检测脉冲捕捉输入”启动。	CH1 - CH4
D1: 指令状态	D1	指定用于保存指令执行状态的整型变量。	仅整型变量

■ PCH 指令说明

指令执行状态

R	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
W	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

指令 0: 禁用 / 1: 启用

注 释

- 可以用系统变量 #L_ExIOSpParmErr(特殊 I/O 参数错误) 来查看错误状态。

☞ "30.5.2 映射 I/O 端子 (通用) ◆ 特殊 I/O 参数错误 (#L_ExIOSpParmErr)" (p30-35)

启用条件

- 当能流为 ON 且确认检测到脉冲捕捉输入时

禁用条件

- 当能流为 OFF 且确认没检测到脉冲捕捉输入时

31.24.16 PCHQ(STD 驱动程序)

符号和功能

梯形图指令名称	梯形图符号	功能	步数
PCHQ		清除脉冲捕捉输入	3 至 5

■ 操作数设置

双击 PCHQ 指令，显示如下对话框。在如下对话框中，指定各项设置。

注 释

- 有关 LT 的特殊 I/O 的详细信息，请参考以下内容。
☞ "30.5 控制 LT 的外部 I/O" (p30-23)



设置	操作数	描述	设置范围
CH	S1	指定通道号，将为该通道分配要更改参数的脉冲捕捉输入。 针对这里指定的通道号的“清除脉冲捕捉输入”启动。（仅适用于 PCH 指令的事件发生通知）	CH1 - CH4
D1: 指令状态	D1	指定用于保存指令执行状态的整型变量。	仅整型变量

■ PCHQ 指令说明

指令执行状态



注 释

- 可以用系统变量 #L_ExIOSpParmErr(特殊 I/O 参数错误) 来查看错误状态。

☞ "30.5.2 映射 I/O 端子 (通用) ◆ 特殊 I/O 参数错误 (#L_ExIOSpParmErr)" (p30-35)

启用条件

- 当指令处于运行状态，且确认存在脉冲捕捉输入清除状态时
- 如果未发生错误

禁用条件

- 在将脉冲捕捉输入清除设置为否后，当确认无脉冲捕捉输入清除状态时
* 当禁用时，操作数 D1 被清 0。

注意

- 当执行该指令时，会确认是否正在清除脉冲捕捉输入。如果是，就会发生错误。
- 当操作数 D1 不能被指令识别时，将发生错误。
- 当使用脉冲来执行指令时 (例如， PT 指令)，需要将指令置 OFF，然后再置 ON 才能重新执行指令。
- 执行指令后，它将处于运行状态。
- 如果指令处于运行状态，无论启用 / 禁用条件如何，都会持续执行该指令。

31.24.17 I/O 驱动程序指令限制

- 即使使用了不同于 I/O 驱动程序设置的指令，也不会发生错误。
例如，I/O 驱动程序设置 CH1：脉冲
例如，在上述情况中使用 PWMX(CH1)
- 即便是上面这种情况，当执行 PWMX 指令时，脉冲参数也会由于 I/O 驱动程序设置而改变。无论是否是 PWMX，都会发生上述情况。
- 不能在线编辑特殊 I/O 驱动程序指令。
- 不要用 #L 系统变量和 I/O 驱动程序指令同时控制同一通道 (CH)。
在现有工程中使用 I/O 驱动程序指令时尤其需要注意。即便是在上述情况中，也不会发生错误。
- 如果在更改或读取参数的同时关闭 GP，在重启 GP 后，参数更改或读取继续进行。这时，当执行 I/O 驱动程序指令时，会认为处理已经启动，并生成一个错误。
- 当发生错误时，指令不能使用。如果错误是在启用指令后发生的，当错误被确认时，指令将被禁用。
- 对于同一通道 (CH)，一定要在确认已启用了当前指令后再执行下一条指令。即使没有确认启用状态，指令也将执行，但可能不能正常发挥作用。
例如，运行带有脉冲输出的 PLSX 和 PLS 指令
在上面的例子中，由于已经用 PLSX 指令更改了设置，脉冲输出可能不能执行。
- 当使用一个来自 PBC-PBR 指令中的指令时，将会发生以下情况。
当 PBC 为 ON 时 (启用)
指令将根据其规范运行。
当 PBC 为 OFF 时 (禁用)
指令将不运行。
然而，一旦指令已经开始运行，无论 PBC 的 ON/OFF 状态如何，它都将运行下去。
- 当 I/O 设置启用时，指令运行。

