

# PLC 指令与梯形图速查手册

张高煜 屈明富 编著

人民邮电出版社

北京

# PLC 指令与梯形图速查手册

## 目 录

<b>第 1 章 位逻辑指令</b>	<b>2</b>
1.1 LD 指令：载入常开触点	2
1.2 A 指令：串联常开触点	3
1.3 O 指令：并联常开触点	4
1.4 LDN 指令：载入常闭触点	5
1.5 AN 指令：串联常闭触点	6
1.6 ON 指令：并联常闭触点	7
1.7 LDI 指令：立即载入常开触点	8
1.8 AI 指令：串联立即常开触点	9
1.9 OI 指令：并联立即常开触点	10
1.10 LDNI 指令：载入立即常闭触点	11
1.11 ANI 指令：串联立即常闭触点	12
1.12 ONI 指令：并联立即常闭触点	14
1.13 NOT 指令：改变使能位输入状态	15
1.14 EU 指令：上升边缘检测	16
1.15 ED 指令：下降边缘检测	17
1.16 ALD 指令：触点块串联	18
1.17 OLD 指令：触点块并联	19
1.18 LPS、LRD、LPP 指令：逻辑堆栈操作	20
1.19 =指令：线圈输出	22
1.20 =I指令：立即线圈输出	23
1.21 S 指令：线圈置位	24
1.22 SI 指令：线圈立即置位	25
1.23 R 指令：线圈复位	26
1.24 RI 指令：线圈立即复位	27
1.25 NOP 指令：空操作	28
1.26 X 指令：异或操作	29
<b>第 2 章 比较指令</b>	<b>31</b>
2.1 LDB=指令：载入字节等于	31
2.2 AB=指令：与运算字节等于	32
2.3 OB=指令：或运算字节等于	33
2.4 LDB 指令：载入字节不等于	34
2.5 AB 指令：与运算字节不等于	35
2.6 OB 指令：或运算字节不等于	36
2.7 LDB=指令：载入字节大于或等于	37
2.8 AB=指令：与运算字节大于或等于	38
2.9 OB=指令：或运算字节大于或等于	40

2.10	LDB=指令：载入字节小于或等于	41
2.11	AB=指令：与运算字节小于或等于	42
2.12	OOB=指令：或运算字节小于或等于	43
2.13	LDB 指令：载入字节大于	44
2.14	AB 指令：与运算字节大于	45
2.15	OB 指令：或运算字节大于	46
2.16	LDB 指令：载入字节小于	47
2.17	AB 指令：与运算字节小于	48
2.18	OB 指令：或运算字节小于	49
2.19	LDW=指令：载入字等于	50
2.20	AW=指令：与运算字等于	51
2.21	OW=指令：或运算字等于	53
2.22	LDW 指令：载入字不等于	54
2.23	AW 指令：与运算字不等于	55
2.24	OW 指令：或运算字不等于	56
2.25	LDW=指令：载入字大于或等于	57
2.26	AW=指令：与运算字大于或等于	58
2.27	OW=指令：或运算字大于或等于	59
2.28	LDW=指令：载入字小于或等于	61
2.29	AW=指令：与运算字小于或等于	62
2.30	OW=指令：或运算字小于或等于	63
2.31	LDW 指令：载入字大于	64
2.32	AW 指令：与运算字大于	65
2.33	OW 指令：或运算字大于	66
2.34	LDW 指令：载入字小于	68
2.35	AW 指令：与运算字小于	69
2.36	OW 指令：或运算字小于	70
2.37	LDD=指令：载入双字等于	71
2.38	AD=指令：与运算双字等于	72
2.39	OD=指令：或运算双字等于	73
2.40	LDD 指令：载入双字不等于	74
2.41	AD 指令：与运算双字不等于	75
2.42	OD 指令：或运算双字不等于	77
2.43	LDD=指令：载入双字大于或等于	78
2.44	AD=指令：与运算双字大于或等于	79
2.45	OD=指令：或运算双字大于或等于	80
2.46	LDD=指令：载入双字小于或等于	81

2.47	AD=指令：与运算双字小于或等于	82
2.48	OD=指令：或运算双字小于或等于	83
2.49	LDD 指令：载入双字大于	84
2.50	AD 指令：与运算双字大于	85
2.51	OD 指令：或运算双字大于	86
2.52	LDD 指令：载入双字小于	88
2.53	AD 指令：与运算双字小于	89
2.54	OD 指令：或运算双字小于	90
2.55	LDR=指令：载入实数等于	91
2.56	AR=指令：与运算实数等于	92
2.57	OR=指令：或运算实数等于	93
2.58	LDR 指令：载入实数不等于	94
2.59	AR 指令：与运算实数不等于	95
2.60	OR 指令：或运算实数不等于	96
2.61	LDR=指令：载入实数大于或等于	97
2.62	AR=指令：与运算实数大于或等于	98
2.63	OR=指令：或运算实数大于或等于	99
2.64	LDR=指令：载入实数小于或等于	101
2.65	AR=指令：与运算实数小于或等于	102
2.66	OR=指令：或运算实数小于或等于	103
2.67	LDR 指令：载入实数大于	104
2.68	AR 指令：与运算实数大于	105
2.69	OR 指令：或运算实数大于	106
2.70	LDR 指令：载入实数小于	107
2.71	AR 指令：与运算实数小于	108
2.72	OR 指令：或运算实数小于	109
<b>第3章 转换指令</b>		<b>111</b>
3.1	BTI 指令：字节转换至整数	111
3.2	ITB 指令：整数转换至字节	112
3.3	ITD 指令：整数转换至长整数	113
3.4	ITS 指令：整数转换至字符串	114
3.5	DTI 指令：长整数转换至整数	116
3.6	DTR 指令：长整数转换至实数	117
3.7	DTS 指令：长整数转换至字符串	118
3.8	ROUND 指令：取整为长整数	120
3.9	TRUNC 指令：截断为长整数	121
3.10	RTS 指令：实数转换至字符串	122

3.11	BCDI 指令: BCD 码转换为整数	124
3.12	IBCD 指令: 整数转换为 BCD 码	125
3.13	ITA 指令: 整数转换至 ASCII 码	126
3.14	DTA 指令: 长整数转换至 ASCII 码	128
3.15	RTA 指令: 实数转换至 ASCII 码	130
3.16	ATH 指令: ASCII 码转换至十六进制数字	132
3.17	HTA 指令: 十六进制数字转换至 ASCII 码	134
3.18	STI 指令: 字符串转换至整数	135
3.19	STD 指令: 字符串转换至长整数	137
3.20	STR 指令: 字符串转换至实数	138
3.21	DECO 指令: 解码	139
3.22	ENCO 指令: 编码	141
3.23	SEG 指令: 七段显示转换	142
<b>第 4 章</b>	<b>计数器指令</b>	<b>144</b>
4.1	CTU 指令: 向上计数	144
4.2	CTD 指令: 向下计数	146
4.3	CTUD 指令: 双向计数	147
4.4	HDEF 指令: 定义高速计数器	149
4.5	HSC 指令: 高速计数器	151
<b>第 5 章</b>	<b>浮点型数学运算指令</b>	<b>154</b>
5.1	+R 指令: 实数加	154
5.2	-R 指令: 实数减	155
5.3	*R 指令: 实数乘	157
5.4	/R 指令: 实数除	158
5.5	SQRT 指令: 求平方根	160
5.6	SIN 指令: 求正弦值	161
5.7	COS 指令: 求余弦值	162
5.8	TAN 指令: 求正切值	163
5.9	LN 指令: 求自然对数	164
5.10	EXP 指令: 求指数值	165
<b>第 6 章</b>	<b>整型数学运算指令</b>	<b>166</b>
6.1	+I 指令: 整数加	166
6.2	+D 指令: 长整数加	167
6.3	?I 指令: 整数减	169
6.4	?D 指令: 长整数减	170
6.5	MUL 指令: 整数与长整数相乘	172
6.6	*I 指令: 整数乘	173

6.7	*D 指令：长整数乘	174
6.8	DIV 指令：整数与长整数相除	176
6.9	/I 指令：除以整数	177
6.10	/D 指令：除以长整数	178
6.11	INCB 指令：递增字节	180
6.12	INCW 指令：递增字	181
6.13	INCD 指令：递增双字	182
6.14	DECB 指令：递减字节	184
6.15	DECW 指令：递减字	185
6.16	DECD 指令：递减双字	186
<b>第 7 章</b>	<b>中断指令</b>	<b>188</b>
7.1	CRETI 指令：从中断(INT)有条件返回	188
7.2	ENI 指令：启用中断	189
7.3	DISI 指令：禁止中断	190
7.4	ATCH 指令：中断连接	191
7.5	DTCH 指令：中断分离	192
7.6	CEVNT 指令：清除中断事件	193
<b>第 8 章</b>	<b>字逻辑指令</b>	<b>195</b>
8.1	INVB 指令：取反字节	195
8.2	INVW 指令：取反字	196
8.3	INVD 指令：取反双字	197
8.4	ANDB 指令：与运算字节	198
8.5	ANDW 指令：与运算字	200
8.6	ANDD 指令：与运算双字	201
8.7	ORB 指令：或运算字节	202
8.8	ORW 指令：或运算字	204
8.9	ORD 指令：或运算双字	205
8.10	XORB 指令：异或运算字节	206
8.11	XORW 指令：异或运算字	208
8.12	XORD 指令：异或运算双字	209
<b>第 9 章</b>	<b>移动指令</b>	<b>211</b>
9.1	MOVB 指令：移动字节	211
9.2	MOVW 指令：移动字	212
9.3	MOVD 指令：移动双字	213
9.4	MOVR 指令：移动实数	214
9.5	BMB 指令：成块移动字节	215
9.6	BMW 指令：成块移动字	216

9.7	BMD 指令：成块移动双字	218
9.8	SWAP 指令：交换字节	219
9.9	BIR 指令：移动字节立即读取	220
9.10	BIW 指令：移动字节立即写入	221
<b>第 10 章</b>	<b>程序控制指令</b>	<b>223</b>
10.1	FOR 指令：循环开始	223
10.2	NEXT 指令：循环结束	225
10.3	JMP 指令：跳接至标签	226
10.4	LBL 指令：语句标签	227
10.5	LSCR 指令：载入顺序控制继电器	227
10.6	SCRT 指令：顺序控制继电器转换	229
10.7	SCRE 指令：顺序控制继电器结束	230
10.8	CRET 指令：从子程序(SBR)有条件返回	232
10.9	END 指令：主程序(OB1)有条件结束	232
10.10	STOP 指令：转换至停止模式	233
10.11	WDR 指令：监视程序复原	233
10.12	DLED 指令：诊断 LED	234
<b>第 11 章</b>	<b>移位和循环移位指令</b>	<b>236</b>
11.1	SLB 指令：向左移位字节	236
11.2	SLW 指令：向左移位字	237
11.3	SLD 指令：向左移位双字	239
11.4	SRB 指令：向右移位字节	240
11.5	SRW 指令：向右移位字	241
11.6	SRD 指令：向右移位双字	243
11.7	RLB 指令：循环左移字节	244
11.8	RLW 指令：循环左移字	245
11.9	RLD 指令：循环左移双字	247
11.10	RRB 指令：循环右移字节	248
11.11	RRW 指令：循环右移字	250
11.12	RLD 指令：循环右移双字	251
11.13	SHRB 指令：寄存器位移	253
<b>第 12 章</b>	<b>字符串指令</b>	<b>255</b>
12.1	SLEN 指令：获取字符串长度	255
12.2	SCPY 指令：将字符串 1 复制至字符串 2	256
12.3	SSCPY 指令：从字符串复制子字符串	258

12.4	SCAT 指令：字符串连接	259
12.5	SFND 指令：在字符串 1 中查找字符串 2	261
12.6	CFND 指令：在字符串中查找字符	262
12.7	LDS=指令：载入字符串等于	264
12.8	AS=指令：与运算字符串等于	265
12.9	OS=指令：或运算字符串等于	266
12.10	LDS 指令：载入字符串不等于	267
12.11	AS 指令：与运算字符串不等于	268
12.12	OS 指令：或运算字符串不等于	269
12.13	ITS 指令：整数转换至字符串	270
12.14	DTS 指令：双整数转换至字符串	272
12.15	RTS 指令：实数转换至字符串	273
12.16	STI 指令：字符串转换至整数	275
12.17	STD 指令：字符串转换至长整数	277
12.18	STR 指令：字符串转换至实数	279
<b>第 13 章</b>	<b>表格指令</b>	<b>281</b>
13.1	FILL 指令：内存填充	281
13.2	ATT 指令：增加至表格	282
13.3	FND?TBL 指令：表格查找	284
13.4	LIFO 指令：后入先出	286
13.5	FIFO 指令：先入先出	287
<b>第 14 章</b>	<b>定时器指令</b>	<b>289</b>
14.1	TON 指令：接通延时定时器	289
14.2	TONR 指令：掉电保护性接通延时定时器	291
14.3	TOF 指令：断开延时定时器	292
14.4	BITIM 指令：开始间隔时间	294
14.5	CITIM 指令：计算间隔时间	295
<b>第 15 章</b>	<b>特殊指令</b>	<b>297</b>
15.1	网络读/网络写	297
15.2	发送/接收	300
15.3	脉冲输出	303
15.4	PID 指令	306



## 第 1 章 位逻辑指令

位逻辑指令使用两个数字：“1”和“0”。这两个数字构成二进制数字系统的基础。“1”和“0”称为二进制数字或二进制位。对触点与线圈而言，“1”表示动作或通电，“0”表示未动作或未通电。位逻辑指令解释信号状态 1 和 0，并根据布尔逻辑对它们进行组合。这些组合产生结果 1 或 0，称为“逻辑运算结果（RLO）”。

### 1.1 LD 指令：载入常开触点

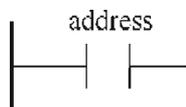
❖ 指令语法 LD <位>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<位>	BOOL	I、Q、M、SM、T、C、V、S、L

❖ 梯形图



❖ 梯形图功能说明

载入指令通常用于打开一个常开触点，同时将地址位数值置于堆栈顶部。

存储在指定<地址>的位值为“1”时，常开触点处于闭合状态，梯形图轨道能流流过触点，逻辑运算结果（RLO）=“1”。如果指定<地址>的信号状态为“0”，触点将处于断开状态，梯形图轨道能流不流过触点，逻辑运算结果（RLO）=“0”。

## 2 第1章 位逻辑指令

辑运算结果 (RLO) = “0”。

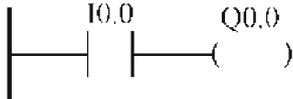
### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<address>	BOOL	I、Q、M、SM、T、C、V、S、L	选中的位

### ❖ 执行范例

⊙ 范例 1 ⊙ 载入 1 个常开触点。

梯 形 图	语 句 表 指 令
	//N.O.触点 I0.0 必须打开(关闭), 才能激活 Q0.0 LD I0.0 = Q0.0

### ❖ 相关指令

LDN、LDI、LDNI

## 1.2 A 指令：串联常开触点

❖ 指令语法 A <位>

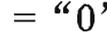
❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<位>	BOOL	I、Q、M、SM、T、C、V、S、L

❖ 梯形图 

❖ 梯形图功能说明 使用“与”指令可以检查被寻址位的信号状态是否为“1”，并将测试结果与逻辑运算结果 (RLO) 进行“与”运算。

存储在指定<地址>的位值为“1”时，常开触点处于闭合状态，梯形图轨道能流流过触点，逻辑运算结果 (RLO) = “1”。如果指定<地址>的信号状态为“0”，触点将处于断开状态，梯形图轨道能流不流过触点，逻辑运算结果 (RLO) = “0”。串联使用时，通过 AND 逻辑将  与 RLO 位进行链接。

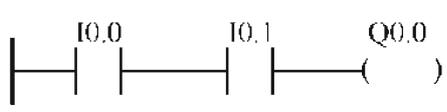
### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<address>	BOOL	I、Q、M、SM、T、C、V、S、L	选中的位

### ❖ 执行范例

⊙ 范例 2 ⊙ 串联 2 个常开触点。

梯 形 图	语 句 表 指 令
	/N.O.触点 I0.0 和 I0.1 必须打开(关闭), 才能激活 Q0.0 LD I0.0 A I0.1 = Q0.0

### ❖ 相关指令

AN、AI、ANI、ALD

## 1.3 O 指令：并联常开触点

❖ 指令语法 O <位>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<位>	BOOL	I、Q、M、SM、T、C、V、S、L

❖ 梯形图 

❖ 梯形图功能说明 使用“或”指令可以检查被寻址位的信号状态是否为“1”，并将测试结果与逻辑运算结果（RLO）进行“或”运算。

存储在指定<地址>的位值为“1”时，常开触点处于闭合状态，梯形图轨道能流流过触点，逻辑运算结果（RLO）=“1”。如果指定<地址>的信号状态为“0”，触点将处于断开状态，梯形图轨道能流不流过触点，逻辑运算结果（RLO）=“0”。并联使用时，通过 OR 逻辑将  与 RLO 位进行链接。

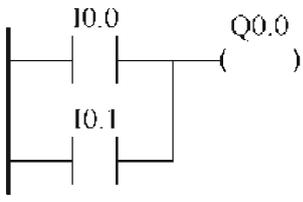
## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<address>	BOOL	I、Q、M、SM、T、C、V、S、L	选中的位

## ❖ 执行范例

● 范例 3 ● 并联 2 个常开触点。

梯 形 图	语 句 表 指 令
	<p>// (N.O.触点 I0.0 必须“打开”) 或 (N.O.触点 I0.1 必须“打开”), 才能激活 Q0.0  // 一个或多个并行梯形图分支 (OR (或) 逻辑输入) 必须为真, 才能激活输出</p> <pre>LD    I0.0 O     I0.1 =     Q0.0</pre>

## ❖ 相关指令

ON、OI、ONI、OLD

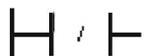
## 1.4 LDN 指令：载入常闭触点

❖ 指令语法 LDN <位>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<位>	BOOL	I、Q、M、SM、T、C、V、S、L

❖ 梯形图 

❖ 梯形图功能说明 载入取反指令通常是打开一个常闭触点，同时将地址位数值置于堆栈顶部。

存储在指定<地址>的位值为“0”时，常闭触点处于闭合状态，梯形图轨道能流流过触点，逻辑运算结果 (RLO) = “1”。如果指定<地址>的信号状态为“1”，将断开触点，梯形图轨道能流不流过触点，逻辑运算

结果 (RLO) = “0”。

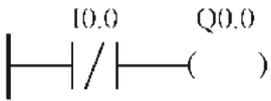
### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<address>	BOOL	I、Q、M、SM、T、C、V、S、L	选中的位

### ❖ 执行范例

● 范例 4 ● 载入 1 个常闭触点。

梯 形 图	语 句 表 指 令
	//N.O.触点 I0.0 必须关闭，才能激活 Q0.0 LDN I0.0 = Q0.0

### ❖ 相关指令

LD、LDI、LDNI

## 1.5 AN 指令：串联常闭触点

❖ 指令语法 AN <位>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<位>	BOOL	I、Q、M、SM、T、C、V、S、L

❖ 梯形图 

❖ 梯形图功能说明 使用“与非”指令可以检查被寻址位的信号状态是否为“0”，并将测试结果与逻辑运算结果 (RLO) 进行“与”运算。

存储在指定<地址>的位值为“0”时，常闭触点处于闭合状态，梯形图轨道能流流过触点，逻辑运算结果 (RLO) = “1”。如果指定<地址>的信号状态为“1”，将断开触点，梯形图轨道能流不流过触点，逻辑运算结果 (RLO) = “0”。串联使用时，通过 AND 逻辑将  与 RLO 位进行

## 6 第1章 位逻辑指令

链接。

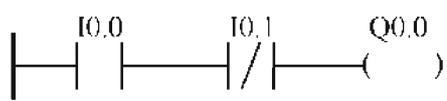
### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<address>	BOOL	I、Q、M、SM、T、C、V、S、L	选中的位

### ❖ 执行范例

● 范例 5 ● 串联 1 个常闭触点。

梯 形 图	语 句 表 指 令
	<p>//N.O.触点 I0.0 必须打开（闭合）和 I0.1 必须关闭，才能激活 Q0.0</p> <pre>LD    I0.0 AN    I0.1 =     Q0.0</pre>

### ❖ 相关指令

A、AI、ANI、ALD

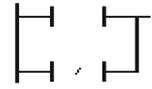
## 1.6 ON 指令：并联常闭触点

❖ 指令语法 ON <位>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<位>	BOOL	I、Q、M、SM、T、C、V、S、L

❖ 梯形图 

❖ 梯形图功能说明 使用“或非”指令可以检查被寻址位的信号

状态是否为“0”，并将测试结果与逻辑运算结果（RLO）进行“或”运算。

存储在指定<地址>的位值为“0”时，常闭触点处于闭合状态，梯形图轨道能流流过触点，逻辑运算结果（RLO）=“1”。如果指定<地址>的

信号状态为“1”，将断开触点，梯形图轨道能流不流过触点，逻辑运算结果 (RLO) = “0”。并联使用时，通过 OR 逻辑将  $\text{—|/|—}$  与 RLO 位进行链接。

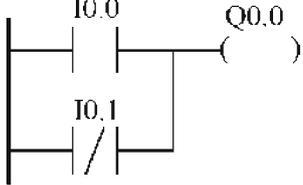
### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<address>	BOOL	I、Q、M、SM、T、C、V、S、L	选中的位

### ❖ 执行范例

⊙ 范例 6 ⊙ 并联 1 个常闭触点。

梯 形 图	语 句 表 指 令
	<p>// (N.O.触点 I0.0 必须“打开”) 或 (N.O.触点 I0.1 必须关闭), 才能激活 Q0.0</p> <p>// 一个或多个并行梯形图分支 (OR (或) 逻辑输入) 必须为真, 才能激活输出</p> <pre>LD    I0.0 ON    I0.1 =     Q0.0</pre>

### ❖ 相关指令

O、OI、ONI、OLD

## 1.7 LDI 指令：立即载入常开触点

❖ 指令语法 LDI <位>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<位>	BOOL	I

❖ 梯形图 

❖ 梯形图功能说明 执行指令时，立即指令获取实际输入值，但不更新进程映象寄存器。立即触点不依赖 PLC 的扫描周期进行更新，而是会立即更新。LDI 指令立即将实际输入的值载入至堆栈顶部。

## 8 第1章 位逻辑指令

当实际输入点（位）是“1”时，立即常开触点打开（被激活而关闭），梯形图轨道能流流过触点，逻辑运算结果（RLO）=“1”。当实际输入点（位）是“0”时，立即常开触点关闭（未被激活而保持打开状态），梯形图轨道能流不流过触点，逻辑运算结果（RLO）=“0”。

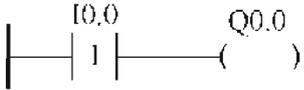
### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<address>	BOOL	I	选中的位

### ❖ 执行范例

◎ 范例 7 ◎ 载入 1 个立即常开触点。

梯 形 图	语 句 表 指 令
	//N.O.触点 I0.0 依据实际输入值而立即影响 Q0.0, 不必等扫描周期更新 LDI I0.0 = Q0.0

### ❖ 相关指令

LDN、LD、LDNI

## 1.8 AI 指令：串联立即常开触点

❖ 指令语法 AI <位>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<位>	BOOL	I

❖ 梯形图 

❖ 梯形图功能说明 执行指令时，立即指令获取实际输入值，但不更新进程映象寄存器。立即触点不依赖 PLC 的扫描周期进行更新，而是会立即更新。AI 指令立即将实际输入值和堆栈顶部当前的道做 AND（与）

运算，将结果压栈。

当实际输入点（位）是“1”时，立即常开触点打开（被激活而关闭），梯形图轨道能流流过触点，逻辑运算结果（RLO）=“1”。当实际输入点（位）是“0”时，立即常开触点关闭（未被激活而保持打开状态），梯形图轨道能流不流过触点，逻辑运算结果（RLO）=“0”。串联使用时，通过AND逻辑将  $\text{—| I |—}$  与 RLO 位进行链接。

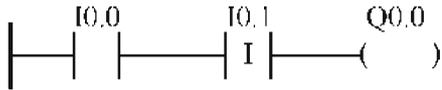
#### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<address>	BOOL	I	选中的位

#### ❖ 执行范例

⊙ 范例 8 ⊙ 串联 1 个立即常开触点。

梯 形 图	语 句 表 指 令
	<p>//N.O.触点 I0.0 必须“打开”并且触点 I0.1 实际输入值是 1 时，Q0.0 立即被激活，不必等扫描周期更新</p> <pre>LD    I0.0 AI    I0.1 =     Q0.0</pre>

#### ❖ 相关指令

AN、ANI、ALD

## 1.9 OI 指令：并联立即常开触点

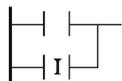
❖ 指令语法 OI <位>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<位>	BOOL	I

❖ 梯形图



❖ **梯形图功能说明** 执行指令时，立即指令获取实际输入值，但不更新进程映象寄存器。立即触点不依赖 PLC 的扫描周期进行更新，而是会立即更新。OI 指令立即将实际输入值和堆栈顶部的值做 OR（或）运算，将结果压栈。

当实际输入点（位）是“1”时，立即常开触点打开（被激活而关闭），梯形图轨道能流流过触点，逻辑运算结果（RLO）=“1”。当实际输入点（位）是“0”时，立即常开触点关闭（未被激活而保持打开状态），梯形图轨道能流不流过触点，逻辑运算结果（RLO）=“0”。并联使用时，通过 OR 逻辑将□<sub>1</sub>□<sub>2</sub>与 RLO 位进行链接。

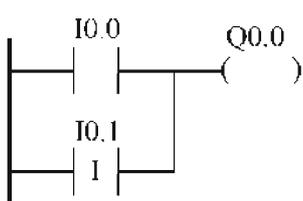
### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<address>	BOOL	I	选中的位

### ❖ 执行范例

#### ● 范例 9 ● 并联 1 个立即常开触点。

梯 形 图	语 句 表 指 令
	<p>//（N.O.触点 I0.0 必须“打开”，Q0.0 被激活）或（触点 I0.1 实际输入值是 1 时，Q0.0 立即被激活，不必等扫描周期更新）</p> <p>// 一个或多个并行梯形图分支（OR（或）逻辑输入）必须为真，才能激活输出</p> <pre>LD    I0.0 OI    I0.1 =     Q0.0</pre>

### ❖ 相关指令

ON、O、ONI、OLD

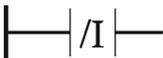
## 1.10 LDNI 指令：载入立即常闭触点

❖ **指令语法** LDNI <位>

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<位>	BOOL	I

❖ 梯形图 

## ❖ 梯形图功能说明

执行指令时，立即指令获取实际输入值，但不更新进程映象寄存器。立即触点不依赖 PLC 的扫描周期进行更新，而是会立即更新。LDNI 指令立即将实际输入点数值的逻辑 NOT（非）载入至堆栈顶部。

当实际输入点（位）是“1”时，立即常闭触点关闭（被激活而打开），能流不流过触点，逻辑运算结果（RLO）=“0”。当实际输入点（位）是“0”时，立即常闭触点打开（未被激活而保持关闭状态），能流流过触点，逻辑运算结果（RLO）=“1”。

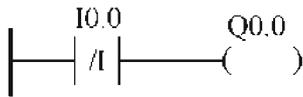
## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<address>	BOOL	I	选中的位

## ❖ 执行范例

◎ 范例 10 ◎ 载入 1 个立即常闭触点。

梯 形 图	语 句 表 指 令
	<pre>//N.O.触点 I0.0 依据实际输入值而立即影响 Q0.0, 不必等扫描周期更新 //触点 I0.0 的实际输入为 0 时, Q0.0 立即被激活 LDNI    I0.0 =       Q0.0</pre>

## ❖ 相关指令

LDN、LD、LDI

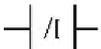
# 1.11 ANI 指令：串联立即常闭触点

## ❖ 指令语法 ANI &lt;位&gt;

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<位>	BOOL	I

❖ 梯形图 

## ❖ 梯形图功能说明

执行指令时，立即指令获取实际输入值，但不更新进程映象寄存器。立即触点不依赖 PLC 的扫描周期进行更新，而是会立即更新。ANI 指令立即将实际输入数值和堆栈当前顶部的值做 AND（与）操作求 NOT（非），再将结果压栈。

当实际输入点（位）是“1”时，立即常闭触点关闭（被激活而打开），能流不流过触点，逻辑运算结果（RLO）=“0”。当实际输入点（位）是“0”时，立即常闭触点打开（未被激活而保持关闭状态），能流流过触点，逻辑运算结果（RLO）=“1”。串联使用时，通过 AND 逻辑将  与 RLO 位进行链接。

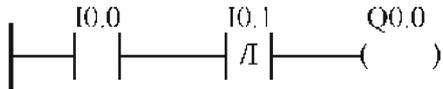
## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<address>	BOOL	I	选中的位

## ❖ 执行范例

◎ 范例 11 ◎ 串联 1 个立即常闭触点。

梯 形 图	语 句 表 指 令
	<p>//N.O.触点 I0.0 必须“打开”，并且触点 I0.1 实际输入值是 0 时，Q0.0 立即被激活，不必等扫描周期更新</p> <pre>LD    I0.0 ANI   I0.1 =     Q0.0</pre>

## ❖ 相关指令

A、AN、AI、ALD

# 1.12 ONI 指令：并联立即常闭触点

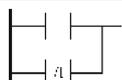
❖ 指令语法 ONI <位>

❖ 指令参数说明

参数的具体说明如下：

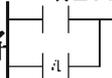
地 址	数 据 类 型	存 储 区
<位>	BOOL	I

❖ 梯形图



❖ 梯形图功能说明

执行指令时，立即指令获取实际输入值，但不更新进程映象寄存器。立即触点不依赖 PLC 的扫描周期进行更新，而是会立即更新。ONI 指令立即将实际输入点数值和堆栈当前顶部的值做 OR（或）运算后求 NOT（非），再将结果压栈。

当实际输入点（位）是“1”时，立即常闭触点关闭（被激活而打开），能流不流过触点，逻辑运算结果（RLO）=“0”。当实际输入点（位）是“0”时，立即常闭触点打开（未被激活而保持关闭状态），能流流过触点，逻辑运算结果（RLO）=“1”。并联使用时，通过 OR 逻辑将  与 RLO 位进行链接。

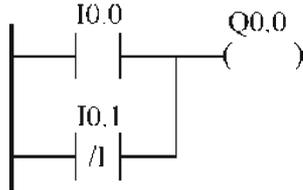
❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<address>	BOOL	I	选中的位

❖ 执行范例

◎ 范例 12 ◎ 并联 1 个立即常闭触点。

梯 形 图	语 句 表 指 令
	<p>// (N.O.触点 I0.0 必须“打开”，Q0.0 被激活) 或 (触点 I0.1 实际输入值是 0 时，Q0.0 立即被激活，不必等扫描周期更新)</p> <p>// 一个或多个并行梯形图分支 (OR (或) 逻辑输入) 必须为真，才能激活输出</p> <pre>LD    I0.0 ONI   I0.1 =     Q0.0</pre>

## ❖ 相关指令

ON、O、OI、OLD

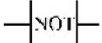
## 1.13 NOT 指令：改变使能位输入状态

## ❖ 指令语法 NOT

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
无	无	无

❖ 梯形图 

## ❖ 梯形图功能说明

执行指令时，指令将堆栈顶部的数值从 0 改变为 1，或从 1 改变为 0。

触点改变使能位输入状态。当使能位到达 NOT（取反）触点时即停止。当使能位未到达 NOT（取反）触点时，则供给使能位。

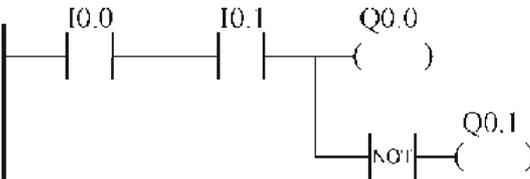
## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
无	无	无	对能流取反

## ❖ 执行范例

## ● 范例 13 ● 对输出取反。

梯 形 图	语 句 表 指 令
	<p>// N.O.接点 I0.0 和 I0.1 必须打开(关闭), 才能激活 Q0.0, NOT (取反) 指令作为反向器作业。在“运行”模式中, Q0.0 和 Q0.1 具有相反的逻辑状态</p> <pre>LD    I0.0 A     I0.1 =     Q0.0 NOT =     Q0.1</pre>

## ❖ 相关指令

ON、O、OI、OLD

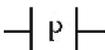
# 1.14 EU 指令：上升边缘检测

## ❖ 指令语法 EU

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
无	无	无

❖ 梯形图 

## ❖ 梯形图功能说明

执行指令时，一旦在堆栈顶部数值中检测到 0 至 1 转换时，则将堆栈顶值设为 1；否则，将其设为 0。

 触点允许一次扫描中每次执行“关闭至打开”转换时能流动。

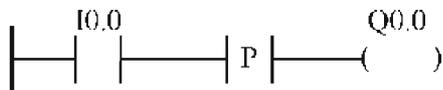
## ❖ 梯形图参数说明

参数的具体说明如下：

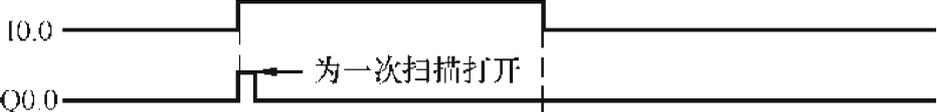
参 数	数 据 类 型	内 存 区 域	说 明
无	无	无	上升沿检测

## ❖ 执行范例

## ◎ 范例 14 ◎ 上升边缘检测。

梯 形 图	语 句 表 指 令
	<pre>// 当触点 I0.0 上有正“边缘向上” // 输入时，输出 Q0.0 带 1 个扫描周 // 期的脉冲 LD    I0.0 EU =     Q0.0</pre>

程序状态视图


--

## ❖ 相关指令

ED

## 1.15 ED 指令：下降边缘检测

## ❖ 指令语法 ED

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
无	无	无

❖ 梯形图  $\text{—|N|—}$ 

❖ 梯形图功能说明 执行指令时，一旦在堆栈顶部数值中检测到 1 至 0 转换时，则将堆栈顶值设为 1；否则，将其设为 0。

$\text{—|N|—}$  触点允许一次扫描中每次执行“打开至关闭”转换时能流流动。

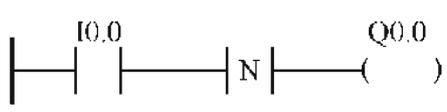
## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
无	无	无	下降沿检测

## ❖ 执行范例

● 范例 15 ● 下降边缘检测。

梯 形 图	语 句 表 指 令
	<pre>// 当触点 I0.0 上有负“边缘向下”输入 // 时，输出 Q0.0 带 1 个扫描周期的脉冲 LD    I0.0 ED =     Q0.0</pre>

程序状态视图



## ❖ 相关指令

EU

# 1.16 ALD 指令：触点块串联

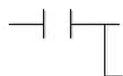
## ❖ 指令语法 ALD

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
无	无	无

## ❖ 梯形图



## ❖ 梯形图功能说明

指令采用逻辑 AND（与）操作将堆栈第一级和第二级中的数值组合，并将结果载入堆栈顶部。执行 ALD 后，堆栈深度减 1。

通常用于将多个触点的组合块进行串联。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
无	无	无	触点块串联

## ❖ 执行范例

## ● 范例 16 ● 触点块串联。

梯 形 图	语 句 表 指 令
	<pre>// 触点 I0.2 和触点 I0.3 的组合触点块与 // 触点 I0.0 进行块串联 LD    I0.0 LD    I0.2 OR    I0.3 ALD =    Q0.0</pre>

续表

梯形图	语句表指令																		
逻辑堆栈状态图																			
AND(与)最上端两个堆栈数值 $S0=iv0 \text{ AND } iv1$ 之前																			
<table border="1" style="border-collapse: collapse; width: 100px;"> <tr><td>iv0</td></tr> <tr><td>iv1</td></tr> <tr><td>iv2</td></tr> <tr><td>iv3</td></tr> <tr><td>iv4</td></tr> <tr><td>iv5</td></tr> <tr><td>iv6</td></tr> <tr><td>iv7</td></tr> <tr><td>iv8</td></tr> </table>	iv0	iv1	iv2	iv3	iv4	iv5	iv6	iv7	iv8	<table border="1" style="border-collapse: collapse; width: 100px;"> <tr><td>S0</td></tr> <tr><td>iv2</td></tr> <tr><td>iv3</td></tr> <tr><td>iv4</td></tr> <tr><td>iv5</td></tr> <tr><td>iv6</td></tr> <tr><td>iv7</td></tr> <tr><td>iv8</td></tr> <tr><td>x<sup>1</sup></td></tr> </table>	S0	iv2	iv3	iv4	iv5	iv6	iv7	iv8	x <sup>1</sup>
iv0																			
iv1																			
iv2																			
iv3																			
iv4																			
iv5																			
iv6																			
iv7																			
iv8																			
S0																			
iv2																			
iv3																			
iv4																			
iv5																			
iv6																			
iv7																			
iv8																			
x <sup>1</sup>																			
之后																			

## ❖ 相关指令

A、AN、AI

## 1.17 OLD 指令：触点块并联

## ❖ 指令语法 OLD

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
无	无	无

❖ 梯形图 

❖ 梯形图功能说明 指令采用逻辑 OR（或）操作将堆栈第一级和第二级中的数值组合，并将结果载入堆栈顶部。执行 OLD 后，堆栈深度减 1。通常用于将多个触点的组合块进行并联。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
无	无	无	触点块并联

## ❖ 执行范例

## ● 范例 17 ● 触点块并联。

梯 形 图	语句表指令
	<pre>// 触点 I2.0 和触点 I2.1 的组合触点块与触点 I0.1 进行块并联 LD    I0.0 LD    I0.1 LD    I2.0 A     I2.1 OLD ALD =     Q0.0</pre>

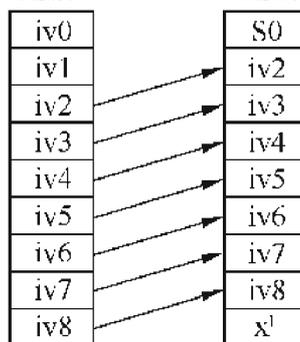
逻辑堆栈状态图

OR (或) 最上端两个堆栈数值

$S0 = iv0 \text{ OR } iv1$

之前

之后



## ❖ 相关指令

O、ON、OI、ONI

# 1.18 LPS、LRD、LPP 指令：逻辑堆栈操作

## ❖ 指令语法 LPS/LRD/LPP

## ❖ 功能说明

LD 装载指令是从梯形图最左侧母线画起的, 如果要生成一条分支的母线, 则需要利用语句表的栈操作指令来描述。逻辑进栈 (LPS) 指令复制堆栈中的顶值并使该数值进栈, 堆栈底值被推出栈并丢失。逻辑出栈 (LPP) 指令将堆栈中的一个数值出栈, 第二个堆栈数值成为堆

栈新顶值。逻辑读取（LRD）指令将第二个堆栈数值复制至堆栈顶部，不执行进栈或出栈，但旧堆栈顶值被复制破坏。

### ❖ 指令参数说明

参数的具体说明如下：

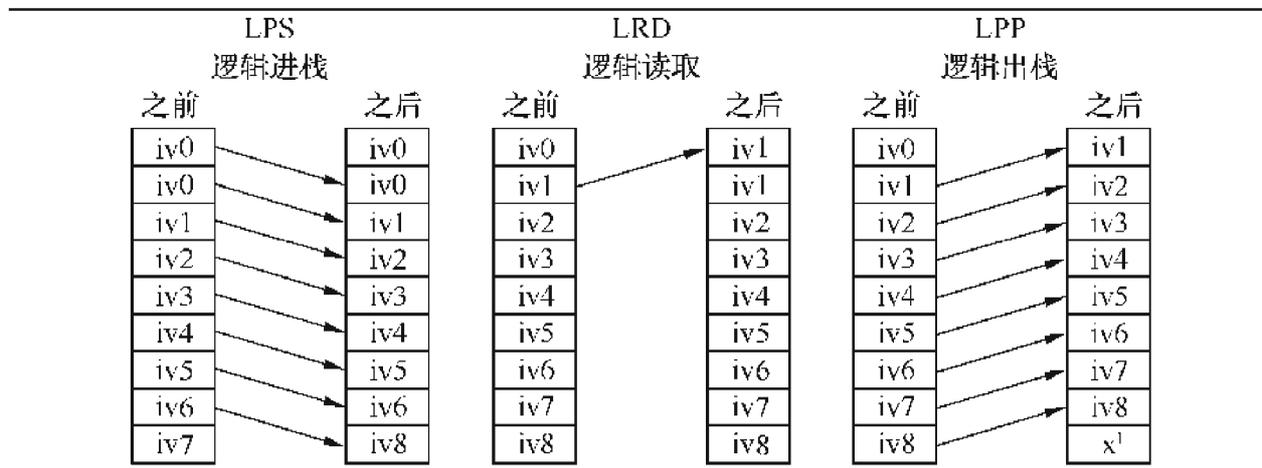
地 址	数 据 类 型	存 储 区
无	无	无

### ❖ 执行范例

#### ◎ 范例 18 ◎ 逻辑堆栈操作。

梯 形 图	语 句 表 指 令
	LD I0.0 //装入常开触点
	LPS //建立栈指针（堆栈）
	LD I0.1 //装入常开触点
	O I0.2 //或常开触点
	ALD //块与操作
	= M0.0 //输出线圈
	LRD //读栈
	LD I0.3 //装入常开触点
	O I0.4 //或常开触点
	ALD //块与操作
	= M0.1 //输出线圈
	LPP //出栈
	A I0.5 //与常开触点
= Q0.0 //输出线圈	

逻辑堆栈状态图



# 1.19 =指令：线圈输出

❖ 指令语法 = <位>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<位>	BOOL	I、Q、M、SM、T、C、V、S、L

❖ 梯形图  $\rightarrow$  ( )

❖ 梯形图功能说明 指令将输出位的新数值写入过程映象寄存器，同时位于堆栈顶端的数值被复制至指定的位。

$\rightarrow$  ( ) 的工作方式与继电器逻辑图中线圈的工作方式类似。如果有能流通过线圈 (RLO = 1)，将置位<地址>位置的位为“1”。如果没有能流通过线圈 (RLO = 0)，将置位<地址>位置的位为“0”。只能将输出线圈置于梯级的右端。可以有多个（最多 16 个）输出单元。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<address>	BOOL	I、Q、M、SM、T、C、V、S、L	分配位

❖ 执行范例

◎ 范例 19 ◎ 使用输出指令指定位值。

梯 形 图	语 句 表 指 令
	<pre>// 输出指令将位值指定给外部 I/O (I、Q) 和内存 (M、SM、T、C、V、S、L) LD    I0.0 =     Q0.0 =     V0.0</pre>

❖ 相关指令

=|

## 1.20 =|指令：立即线圈输出

❖ 指令语法 =|<位>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<位>	BOOL	Q

❖ 梯形图 —( | )

❖ 梯形图功能说明 指令将新值写入实际输出和对应的过程映象寄存器位置，同时将位于堆栈顶端的数值复制至指定的实际输出位。

—( | ) 执行“立即输出”指令时，实际输出点（位）被立即设为等于使能位。“|”表示立即参考，执行指令时，新值被写入实际输出和对应的过程映象寄存器位置。这与非立即参考不同，非立即参考仅将新值写入过程映象寄存器。

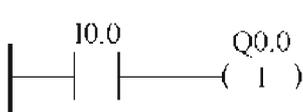
❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<address>	BOOL	Q	立即分配位

❖ 执行范例

◎ 范例 20 ◎ 使用立即输出指令。

梯 形 图	语 句 表 指 令
	<p>// 当触点 I0.0 被打开（关闭），Q0.0 立即被激活，不必等到下一次扫描周期来到</p> <pre>LD    I0.0 =     Q0.0</pre>

❖ 相关指令

=

# 1.21 S 指令：线圈置位

❖ 指令语法 S <位> <数值>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<位>	BOOL	I、Q、M、SM、T、C、V、S、L
<数值>	字节	VB、IB、QB、MB、SMB、SB、LB、AC、常数、*VD、*AC、*LD

❖ 梯形图  $\rightarrow \begin{matrix} \text{bit} \\ \text{S} \\ \text{N} \end{matrix}$

❖ 梯形图功能说明 设置（打开）指定的点数（N），从指定的地址（位）开始，可以设置 1~255 个点。

只有在前面指令的 RLO 为“1”（能流通过线圈）时，才会执行线圈置位。如果 RLO 为“1”，将把单元的指定<地址>置位为“1”。如果 RLO 为“0”，将不起作用，单元指定地址的当前状态将保持不变。

❖ 梯形图参数说明

参数的具体说明如下：

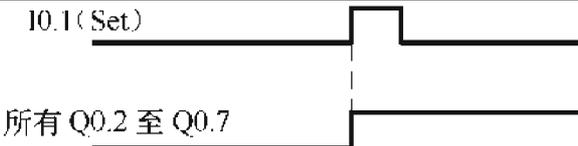
参 数	数 据 类 型	内 存 区 域	说 明
<address>	BOOL	I、Q、M、SM、T、C、V、S、L	分配位
<N>	字节	VB、IB、QB、MB、SMB、SB、LB、AC、常数、*VD、*AC、*LD	可以置位的点数

❖ 执行范例

◎ 范例 21 ◎ 使用线圈置位指令。

梯 形 图	语 句 表 指 令
	<p>// 用“设置”指令将连续 6 个位群组设为 1。指定起始位址以及需要设置的位数。“设置”指令执行锁定中继功能。当第一个位（Q0.2）的数值是 1 时，“程序状态指示器”打开</p> <pre>LD    I0.1 S     Q0.2, 6</pre>

续表

梯形图	语句表指令
程序状态视图	
	

## ❖ 相关指令

SI

## 1.22 SI 指令：线圈立即置位

❖ 指令语法 SI &lt;位&gt; &lt;数值&gt;

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<位>	BOOL	Q
<数值>	字节	VB、IB、QB、MB、SMB、SB、LB、AC、常数、*VD、*AC、*LD

❖ 梯形图  $\text{—} \begin{matrix} \text{bit} \\ \text{( SI )} \\ \text{N} \end{matrix}$ 

❖ 梯形图功能说明 立即设置（SI）指令用于立即设置（打开）点数（N），从指定的地址（位）开始，可以立即设置1~128个点。

指令中的“1”表示立即引用；执行指令时，新值被写入实际输出点和相应的过程映象寄存器位置。这与非立即参考不同，非立即参考只将新值写入过程映象寄存器。

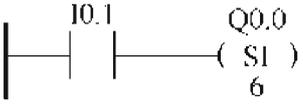
❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<address>	BOOL	Q	分配位
<N>	字节	VB、IB、QB、MB、SMB、SB、LB、AC、常数、*VD、*AC、*LD	可以置位的点数

## ❖ 执行范例

● 范例 22 ● 使用线圈立即置位指令。

梯 形 图	语句表指令
	<p>// 当触点 I0.0 被打开（关闭）时，Q0.0 到 Q0.5 立即被激活，不必等到下一次扫描周期来到</p> <pre>LD    I0.1 SI    Q0.2, 6</pre>

## ❖ 相关指令

S

## 1.23 R 指令：线圈复位

❖ 指令语法 R <位> <数值>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<位>	BOOL	I、Q、M、SM、T、C、V、S、L
<数值>	字节	VB、IB、QB、MB、SMB、SB、LB、AC、常数、*VD、*AC、*LD

❖ 梯形图  $\text{—} \begin{matrix} \text{bit} \\ \text{( R )} \\ \text{N} \end{matrix}$

❖ 梯形图功能说明 复原（R）指令复原指定的点数（N），从指定的地址（位）开始，可以复原 1~255 个点。

只有在前面指令的 RLO 为“1”（能流通过线圈）时，才会执行线圈复位。如果能流通过线圈（RLO 为“1”），将把单元的指定<地址>复位为“0”。如果没有能流通过线圈（RLO 为“0”），单元指定地址的状态将保持不变。

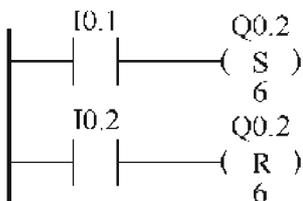
❖ 梯形图参数说明

参数的具体说明如下：

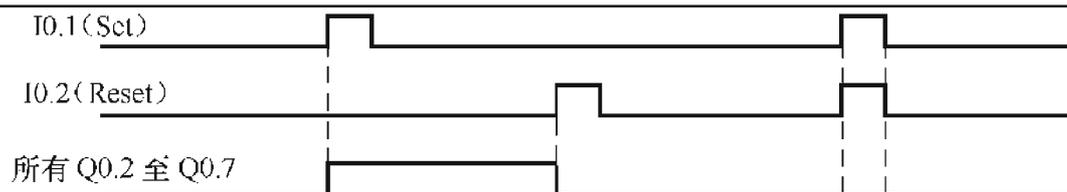
参 数	数 据 类 型	内 存 区 域	说 明
<address>	BOOL	I、Q、M、SM、T、C、V、S、L	分配位
<N>	字节	VB、IB、QB、MB、SMB、SB、LB、AC、常数、*VD、*AC、*LD	可以复位的点数

### ❖ 执行范例

#### ◎ 范例 23 ◎ 使用线圈置位/复位指令。

梯 形 图	语 句 表 指 令
	<p>// 用“设置”指令将连续 6 个位群组设为 1。指定起始位址以及需要设置的位数。“设置 / 复原”指令执行锁定中继功能。当第一个位 (Q0.2) 的数值是 1 时，“设置”的程序状态指示器“打开”</p> <pre>LD    I0.1 S     Q0.2, 6</pre> <p>// 用“复原”指令将连续 6 个位群组设为 0。指定起始位址以及需要复员的位数。当第一个位 (Q0.2) 的数值是 0 时，“设置”的程序状态指示器“打开”</p> <pre>LD    I0.2 R     Q0.2, 6</pre>

程序状态视图



“重设为 0”盖写“设为 1”，因为程序扫描在“网络 2 设置”之后执行“网络 3 重设”

### ❖ 相关指令

RI

## 1.24 RI 指令：线圈立即复位

❖ 指令语法 RI <位> <数值>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<位>	BOOL	Q
<数值>	字节	VB、IB、QB、MB、SMB、SB、LB、AC、常数、*VD、*AC、*LD

❖ 梯形图  $\rightarrow$   $\begin{matrix} \text{bit} \\ \text{---( RI )} \\ \text{N} \end{matrix}$

❖ 梯形图功能说明 立即复原 (RI) 指令立即复原 (关闭) 点数 (N)，从指定的地址 (位) 开始，可以立即复原 1~128 个点。

指令中的“I”表示立即引用，执行指令时，新值被写入实际输出点和相应的过程映象寄存器位置。这与非立即参考不同，非立即参考只将新值写入过程映象寄存器。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<address>	BOOL	Q	分配位
<N>	字节	VB、IB、QB、MB、SMB、SB、LB、AC、常数、*VD、*AC、*LD	可以复位的点数

❖ 执行范例

⊙ 范例 24 ⊙ 使用线圈立即复位指令。

梯 形 图	语 句 表 指 令
	<pre>// 当触点 I0.0 被打开 (关闭) 时，Q0.0 到 Q0.5 立即被复位，不必等到下一次扫描周期来到 LD    I0.1 RI    Q0.2, 6</pre>

❖ 相关指令

R

## 1.25 NOP 指令：空操作

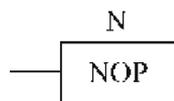
❖ 指令语法 NOP <数值>

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<数值>	字节	常数

## ❖ 梯形图



## ❖ 梯形图功能说明

指令执行空操作，常用来进行延时或者程序调试。

空操作次数  $N=0\sim 255$ 。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<N>	字节	常数	空操作次数

## ❖ 执行范例

## ● 范例 25 ● 使用空指令。

梯 形 图	语 句 表 指 令
	// 空操作次数为 5 次 NOP 5

## 1.26 X 指令：异或操作

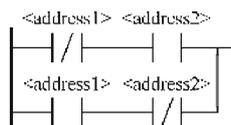
## ❖ 指令语法 X &lt;数值&gt;

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<位>	BOOL	I、Q、M、L、D、T、C

## ❖ 梯形图



❖ **梯形图功能说明** 使用“异或”指令可以检查被寻址位的信号状态是否为“1”，并将测试结果与逻辑运算结果（RLO）进行“异或”运算。也可以多次使用“异或”指令。因此，如果有奇数个被检查地址为“1”，则逻辑运算的交互结果为“1”。

如果两个指定位的信号状态不同，则创建状态为“1”的 RLO。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<address1>	BOOL	I、Q、M、L、D、T、C	扫描的位
<address2>	BOOL	I、Q、M、L、D、T、C	扫描的位

### ❖ 执行范例

● 范例 26 ● 使用异或指令。

梯 形 图	语 句 表 指 令
	<pre>// 如果 I0.0 = “0” 且 I0.1 = “1” 或者 I0.0 = “1” 且 I0.1 = “0”，输出 Q4.0 将是 “1” X   I0.0,I0.1 =   Q4.0</pre>



## 第 2 章 比较指令

根据用户选择的比较类型比较两个输入值，如果比较结果为“真”，则函数的 RLO 为“1”。如果以串联方式使用比较单元，则使用“与”运算将其链接至梯级程序段的 RLO；如果以并联方式使用比较单元，则使用“或”运算将其链接至梯级程序段的 RLO。

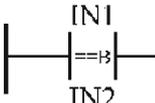
### 2.1 LDB=指令：载入字节等于

❖ 指令语法 LDB= <IN1> <IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC

❖ 梯形图 

❖ 梯形图功能说明 用来比较两个字节 IN1 和 IN2 的大小是否相等。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”。

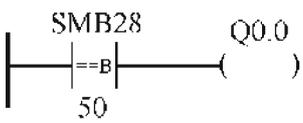
### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数

### ❖ 执行范例

◎ 范例 27 ◎ 使用载入字节等于指令。

梯 形 图	语 句 表 指 令
	<pre>// 调整模拟电位器 0，变化 SMB28 字节数值，当 SMB28 数值等于 50 时，Q0.0 被激活 LDB=  SMB28 50 =      Q0.0</pre>

### ❖ 相关指令

LDB>=、LDB>、LDB<>、LDB<=、LDB<

## 2.2 AB=指令：与运算字节等于

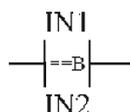
❖ 指令语法 AB= <IN1> <IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

用来比较两个字节 IN1 和 IN2 的大小是否相等。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为

“1”，并且使用“与”运算将其链接至梯级程序段的 RLO。

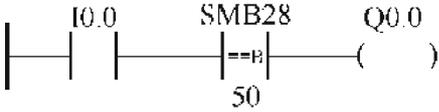
### ◆ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数

### ◆ 执行范例

◎ 范例 28 ◎ 使用与运算字节等于指令。

梯 形 图	语 句 表 指 令
	<pre>// 调整模拟电位器，变化 SMB27 字节数值，当 SMB27 // 数值等于 50 并且 I0.0 时打开，Q0.0 被激活 LD    I0.0 AB=   SMB28 50 =     Q0.0</pre>

### ◆ 相关指令

AB>=、AB>、AB<>、AB<=、AB<

## 2.3 OB=指令：或运算字节等于

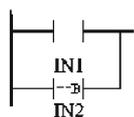
◆ 指令语法 OB= <IN1> <IN2>

### ◆ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC

### ◆ 梯形图



**❖ 梯形图功能说明** 用来比较两个字节 IN1 和 IN2 的大小是否相等。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“或”运算将其链接至梯级程序段的 RLO。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数

### ❖ 执行范例

● 范例 29 ● 使用或运算字节等于指令。

梯 形 图	语 句 表 指 令
	<pre>// 调整模拟电位器，变化 SMB28 字节数值，当 SMB28 数值等于 50 或者 I0.0 打开时，Q0.0 被激活 LD    I0.0 OB=   SMB28 50 =     Q0.0</pre>

### ❖ 相关指令

OB>=、OB>、OB<>、OB<=、OB<

## 2.4 LDB<>指令：载入字节不等于

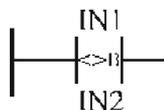
**❖ 指令语法** LDB<> <IN1> <IN2>

### ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

用来比较两个字节 IN1 和 IN2 的大小是否不相等。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数

## ❖ 执行范例

● 范例 30 ● 使用载入字节不等于指令。

梯 形 图	语 句 表 指 令
	<pre>// 将内部内存字节 MB1 与 50 相比较，当 MB1 数值 // 不等于 50 时，Q0.0 被激活 LDB&lt;&gt; MB1 50 =      Q0.0</pre>

## ❖ 相关指令

LDB==、LDB>、LDB>=、LDB<=、LDB<

## 2.5 AB<>指令：与运算字节不等于

## ❖ 指令语法

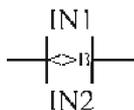
AB<> <IN1> <IN2>

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 功能说明

用来比较两个字节 IN1 和 IN2 的大小是否不相等。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“与”运算将其链接至梯级程序段的 RLO。

## ❖ 梯形图功能说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数

## ❖ 执行范例

● 范例 31 ● 使用与运算字节不等于指令。

梯 形 图	语 句 表 指 令
	<pre>//将内部内存字节 MB2 与 100 相比较，当 MB2 数值不 //等于 100 并且 I0.0 打开时，Q0.0 被激活 LD    I0.0 AB&lt;&gt; MB2 100 =     Q0.0</pre>

## ❖ 相关指令

AB>=、AB>、AB==、AB<=、AB<

## 2.6 OB<>指令：或运算字节不等于

❖ 指令语法 OB<> <IN1> <IN2>

❖ 指令参数说明

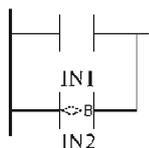
参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC

续表

地 址	数 据 类 型	存 储 区
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

用来比较两个字节 IN1 和 IN2 的大小是否不相等。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“或”运算将其链接至梯级程序段的 RLO。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数

## ❖ 执行范例

◎ 范例 32 ◎ 使用或运算字节不等于指令。

梯 形 图	语 句 表 指 令
	<pre>//将内部内存字节 MB1 与 50 相比较,当 MB1 数值不 //等于 50 或者 I0.0 打开时, Q0.0 被激活 LD    I0.0 OB&lt;&gt; MB1 50 =     Q0.0</pre>

## ❖ 相关指令

OB>=、OB>、OB==、OB<=、OB<

## 27

## LDB&gt;=指令：载入字节大于或等于

## ❖ 指令语法

LDB>= <IN1> <IN2>

### ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、 常数、*VD、*LD、*AC
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、 常数、*VD、*LD、*AC



### ❖ 梯形图功能说明

用来比较两个字节 IN1 和 IN2 是否是大于或等于关系。当比较式为真的时候，该触点打开（即常开触电闭合），且函数的 RLO 为“1”。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、 AC、常数、*VD、*LD、*AC	无符号整数
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、 AC、常数、*VD、*LD、*AC	无符号整数

### ❖ 执行范例

◎ 范例 33 ◎ 使用载入字节大于或等于指令。

梯 形 图	语 句 表 指 令
	<pre>// 将内部内存字节 MB2 与 50 相比较，当 MB2 数值 // 大于或等于 50 时，Q0.0 被激活 LDB&gt;= MB2 50 =      Q0.0</pre>

### ❖ 相关指令

LDB==、LDB>、LDB<>、LDB<=、LDB<

## 2.8

## AB>=指令：与运算字节大于或等于

### ❖ 指令语法

AB>= <IN1> <IN2>

## ◆ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC



◆ 梯形图功能说明 用来比较两个字节 IN1 和 IN2 是否是大于或等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“与”运算将其链接至梯级程序段的 RLO。

## ◆ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数

## ◆ 执行范例

◎ 范例 34 ◎ 使用与运算字节大于或等于指令。

梯 形 图	语 句 表 指 令
	<p>//将内部内存字节 MB2 与 100 相比较，当 MB2 数值大于或等于 100 并且 I0.0 打开时，Q0.0 被激活</p> <pre>LD    I0.0 AB&gt;= MB2 100 =     Q0.0</pre>

## ◆ 相关指令

AB==、AB>、AB<>、AB<=、AB<

## 2.9

## OB&gt;=指令：或运算字节大于或等于

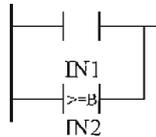
❖ 指令语法  $OB>= <IN1> <IN2>$

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

用来比较两个字节 IN1 和 IN2 是否是大于或等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“或”运算将其链接至梯级程序段的 RLO。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数

❖ 执行范例

◎ 范例 35 ◎ 使用或运算字节大于或等于指令。

梯 形 图	语 句 表 指 令
	<pre>//将内部内存字节 MB1 与 50 相比较,当 MB1 数值大于或等于 50 或者 I0.0 打开时, Q0.0 被激活 LD      I0.0 OB&gt;=   MB1 50 =       Q0.0</pre>

## ❖ 相关指令

OB==、OB&gt;、OB&lt;&gt;、OB&lt;=、OB&lt;

## 2.10 LDB<=指令：载入字节小于或等于

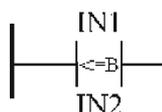
❖ 指令语法 LDB&lt;= &lt;IN1&gt; &lt;IN2&gt;

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、 常数、*VD、*LD、*AC
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、 常数、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

用来比较两个字节 IN1 和 IN2 是否是小于或等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、 AC、常数、*VD、*LD、*AC	无符号整数
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、 AC、常数、*VD、*LD、*AC	无符号整数

## ❖ 执行范例

● 范例 36 ● 使用载入字节小于或等于指令。

梯 形 图	语 句 表 指 令
	<pre>// 将内部内存字节 MB2 与 100 相比较,当 MB2 数值 // 小于或等于 100 时, Q0.0 被激活 LDB&gt;= MB2 50 =      Q0.0</pre>

## ❖ 相关指令

LDB==、LDB&gt;、LDB&lt;&gt;、LDB&gt;=、LDB&lt;

# 2.11 AB<=指令：与运算字节小于或等于

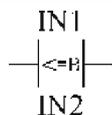
❖ 指令语法 AB&lt;= &lt;IN1&gt; &lt;IN2&gt;

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

用来比较两个字节 IN1 和 IN2 是否是小于或等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“与”运算将其链接至梯级程序段的 RLO。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数

## ❖ 执行范例

### ◎ 范例 37 ◎ 使用与运算字节小于或等于指令。

梯 形 图	语 句 表 指 令
	<pre>//将内部内存字节 MB2 与 100 相比较， //当 MB2 数值小于或等于 100 并且 I0.0 打 //开时，Q0.0 被激活 LD      I0.0 AB&lt;=   MB2 100 =       Q0.0</pre>

## ❖ 相关指令

AB==、AB&gt;、AB&lt;&gt;、AB&gt;=、AB&lt;

## 2.12 OOB<=指令：或运算字节小于或等于

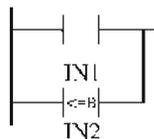
❖ 指令语法 OB&lt;= &lt;IN1&gt; &lt;IN2&gt;

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、 常数、*VD、*LD、*AC
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、 常数、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

用来比较两个字节 IN1 和 IN2 是否是小于或等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“或”运算将其链接至梯级程序段的 RLO。

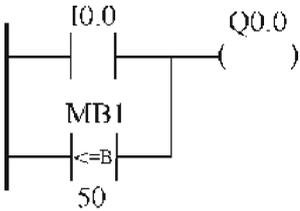
## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、 AC、常数、*VD、*LD、*AC	无符号整数
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、 AC、常数、*VD、*LD、*AC	无符号整数

## ❖ 执行范例

○ 范例 38 ○ 使用或运算字节小于或等于指令。

梯形图	语句表指令
	<p>//将内部内存字节 MB1 与 50 相比较,当 MB1 数值小于或等于 50 或者 I0.0 打开时, Q0.0 被激活</p> <pre>LD    I0.0 OB&lt;=  MB1 50 =     Q0.0</pre>

### ❖ 相关指令

OB==、OB>、OB<>、OB>=、OB<

## 2.13 LDB>指令：载入字节大于

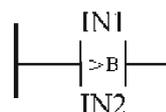
❖ 指令语法 LDB> <IN1> <IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

用来比较两个字节 IN1 和 IN2 是否是大于关系。当比较式为真时,该触点打开(即常开触电闭合),且函数的 RLO 为“1”。

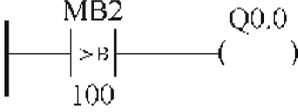
❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数

## ❖ 执行范例

## ◎ 范例 39 ◎ 使用载入字节大于指令。

梯形图	语句表指令
	<pre>// 将内部内存字节 MB2 与 100 相比较, 当 MB2 数值 // 大于 100 时, Q0.0 被激活 LDB&gt; MB2 50 =      Q0.0</pre>

## ❖ 相关指令

LDB==、LDB&lt;=、LDB&lt;&gt;、LDB&gt;=、LDB&lt;

## 2.14 AB>指令：与运算字节大于

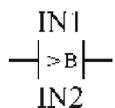
❖ 指令语法 AB&gt; &lt;IN1&gt; &lt;IN2&gt;

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC

❖ 梯形图



## ❖ 梯形图功能说明

用来比较两个字节 IN1 和 IN2 是否是大于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“与”运算将其链接至梯级程序段的 RLO。

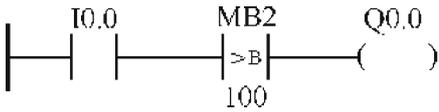
## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数

## ❖ 执行范例

## ❶ 范例 40 使用与运算字节大于指令。

梯 形 图	语句表指令
	<pre>//将内部内存字节 MB2 与 100 相比较， //当 MB2 数值大于 100 并且 I0.0 打开时， //Q0.0 被激活 LD    I0.0 AB&gt;  MB2 100 =     Q0.0</pre>

## ❖ 相关指令

AB==、AB<=、AB<>、AB>=、AB<

## 2.15 OB>指令：或运算字节大于

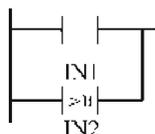
❖ 指令语法 OB><IN1> <IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、 常数、*VD、*LD、*AC
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、 常数、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

用来比较两个字节 IN1 和 IN2 是否是大于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“或”运算将其链接至梯级程序段的 RLO。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数

### ❖ 执行范例

● 范例 41 ● 使用或运算字节大于指令。

梯 形 图	语 句 表 指 令
	<pre>//将内部内存字节 MB1 与 50 相比较,当 MB1 数值大于 50 或者 I0.0 打开时, Q0.0 被激活 LD    I0.0 OB&gt;  MB1 50 =     Q0.0</pre>

### ❖ 相关指令

OB==、OB<=、OB<>、OB>=、OB<

## 2.16 LDB<指令：载入字节小于

❖ 指令语法 LDB< <IN1> <IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC

❖ 梯形图

❖ 梯形图功能说明 用来比较两个字节 IN1 和 IN2 是否是小于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”。

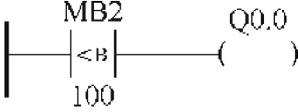
### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数

### ❖ 执行范例

◎ 范例 42 ◎ 使用载入字节小于指令。

梯 形 图	语 句 表 指 令
	<pre>// 将内部内存字节 MB2 与 100 相比较,当 MB2 数值 // 小于 100 时, Q0.0 被激活 LDB&lt; MB2 100 =      Q0.0</pre>

### ❖ 相关指令

LDB==、LDB<=、LDB<>、LDB>=、LDB>

## 2.17 AB<指令：与运算字节小于

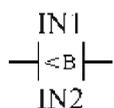
❖ 指令语法 AB< <IN1> <IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

用来比较两个字节 IN1 和 IN2 是否是小于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为

“1”，并且使用“与”运算将其链接至梯级程序段的 RLO。

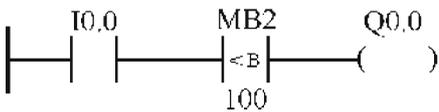
### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数

### ❖ 执行范例

◎ 范例 43 ◎ 使用与运算字节小于指令。

梯 形 图	语 句 表 指 令
	<pre>//将内部内存字节 MB2 与 100 相比较， //当 MB2 数值小于 100 并且 I0.0 打开时， //Q0.0 被激活 LD    I0.0 AB&lt;  MB2 100 =     Q0.0</pre>

### ❖ 相关指令

AB==、AB<=、AB<>、AB>=、AB>

## 2.18 OB<指令：或运算字节小于

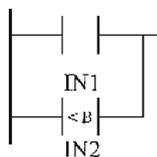
❖ 指令语法 OB<<IN1> <IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

用来比较两个字节 IN1 和 IN2 是否是小于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“或”运算将其链接至梯级程序段的 RLO。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数
<IN2>	字节	IB、QB、MB、SMB、VB、SB、LB、AC、常数、*VD、*LD、*AC	无符号整数

## ❖ 执行范例

● 范例 44 ● 使用或运算字节小于指令。

梯 形 图	语 句 表 指 令
	<pre>//将内部内存字节 MB1 与 50 相比较,当 MB1 数值小于 50 或者 I0.0 打开时, Q0.0 被激活 LD    I0.0 OB&lt;  MB1 50 =     Q0.0</pre>

## ❖ 相关指令

OB==、OB<=、OB<>、OB>=、OB>

## 2.19 LDW=指令：载入字等于

## ❖ 指令语法

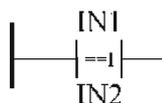
LDW= <IN1> <IN2>

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	整数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC
<IN2>	整数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC

### ❖ 梯形图



### ❖ 梯形图功能说明

用来比较两个带符号整数 IN1 和 IN2 是否是相等关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	整数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数
<IN2>	整数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数

### ❖ 执行范例

#### ● 范例 45 ● 使用载入字等于指令。

梯 形 图	语 句 表 指 令
	<pre>//将变量内存 VW0 中的数值与 + 10000 相比较， //当 VW0 数值等于 + 10000 时，Q0.0 被激活 LDW=  VW0 +10000 =      Q0.0</pre>

### ❖ 相关指令

LDW<、LDW<=、LDW<>、LDW>=、LDW>

## 2.20

## AW=指令：与运算字等于

### ❖ 指令语法

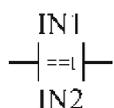
AW= <IN1> <IN2>

### ◆ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	整数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC
<IN2>	整数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC

### ◆ 梯形图



### ◆ 梯形图功能说明

用来比较两个带符号整数 IN1 和 IN2 是否是等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“与”运算将其链接至梯级程序段的 RLO。

### ◆ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	整数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数
<IN2>	整数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数

### ◆ 执行范例

#### ◎ 范例 46 ◎ 使用与运算字等于指令。

梯 形 图	语 句 表 指 令
	<pre>//将变量内存 VW0 中的数值与+10000 //相比较，当 VW0 数值等于+10000 并且 //I0.0 打开时，Q0.0 被激活 LD    I0.0 AW=   VW0 +10000 =     Q0.0</pre>

## ❖ 相关指令

AW&lt;、AW&lt;=、AW&lt;&gt;、AW&gt;=、AW&gt;

# 2.21 OW=指令：或运算字等于

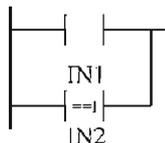
❖ 指令语法 OW=&lt;IN1&gt; &lt;IN2&gt;

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC
<IN2>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

用来比较两个字节 IN1 和 IN2 是否是相等关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“或”运算将其链接至梯级程序段的 RLO。

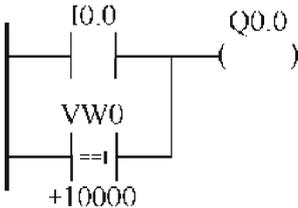
## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数
<IN2>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数

## ❖ 执行范例

○ 范例 47 ○ 使用或运算字等于指令。

梯形图	语句表指令
	<p>//将变量内存 VW0 中的数值与 +10000 相比较，当 VW0 数值等于 +10000 或者 I0.0 打开时，Q0.0 被激活</p> <pre>LD    I0.0 OW=   VW0 +10000 =      Q0.0</pre>

### ❖ 相关指令

OW<、OW<=、OW<>、OW>=、OW>

## 2.22 LDW<>指令：载入字不等于

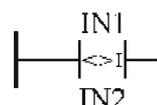
### ❖ 指令语法 LDW<> <IN1> <IN2>

### ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	整数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC
<IN2>	整数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC

### ❖ 梯形图



### ❖ 梯形图功能说明

用来比较两个带符号整数 IN1 和 IN2 是否是不相等关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”。

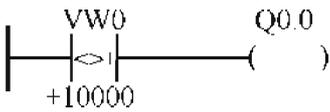
### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	整数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数
<IN2>	整数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数

## ❖ 执行范例

## ● 范例 48 ● 使用载入字不等于指令。

梯形图	语句表指令
	//将变量内存 VW0 中的数值与 + 10000 相比较， 当 VW0 数值不等于 +10000 时，Q0.0 被激活 LDW<> VW0 +10000 = Q0.0

## ❖ 相关指令

LDW&lt;、LDW&lt;=、LDW=、LDW&gt;=、LDW&gt;

## 2.23 AW<>指令：与运算字不等于

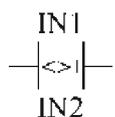
## ❖ 指令语法 AW&lt;&gt; &lt;IN1&gt; &lt;IN2&gt;

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	整 数	W、QW、MW、SW、SMW、T、C、VW、 LW、AIW、AC、常数、*VD、*LD、*AC
<IN2>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、 AIW、AC、常数、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

用来比较两个带符号整数 IN1 和 IN2 是否是不相等关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“与”运算将其链接至梯级程序段的 RLO。

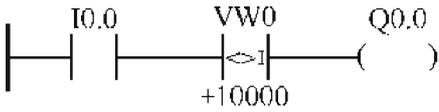
## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	整 数	W、QW、MW、SW、SMW、T、C、VW、 LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数
<IN2>	整 数	W、QW、MW、SW、SMW、T、C、VW、 LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数

## ❖ 执行范例

## ❶ 范例 49 使用与运算字不等于指令。

梯 形 图	语句表指令
	<pre>//将变量内存 VW0 中的数值与 +10000 //相比较，当 VW0 数值不等于 +10000 并 //且 I0.0 打开时，Q0.0 被激活 LD    I0.0 AW&lt;&gt;  VW0 +10000 =      Q0.0</pre>

## ❖ 相关指令

AW<、AW<=、AW=、AW>=、AW>

## 2.24 OW<>指令：或运算字不等于

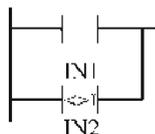
❖ 指令语法  $OW<> <IN1> <IN2>$

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	整数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC
<IN2>	整数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明 用来比较两个字 IN1 和 IN2 是否是不相等关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“或”运算将其链接至梯级程序段的 RLO。

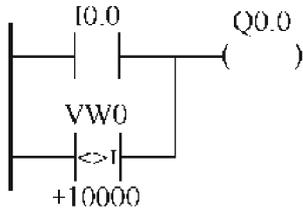
❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	整 数	W、QW、MW、SW、SMW、T、C、VW、 LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数
<IN2>	整 数	W、QW、MW、SW、SMW、T、C、VW、 LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数

### ❖ 执行范例

◎ 范例 50 ◎ 使用或运算字不等于指令。

梯 形 图	语 句 表 指 令
	<pre>//将变量内存 VW0 中的数值与 +10000 相比较,当 //VW0 数值不等于 +10000 或者 I0.0 打开时, Q0.0 //被激活 LD    I0.0 OW&lt;&gt;  VW0 +10000 =     Q0.0</pre>

### ❖ 相关指令

OW<、OW<=、OW=、OW>=、OW>

## 2.25 LDW>=指令：载入字大于或等于

❖ 指令语法 LDW>= <IN1> <IN2>

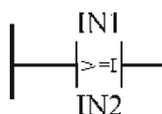
❖ 功能说明 载入字大于或等于指令将两个带符号整数做大于或等于比较，比较结果为真时，1 位于堆栈顶端，指令执行载入操作。

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、 AIW、AC、常数、*VD、*LD、*AC
<IN2>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、 AIW、AC、常数、*VD、*LD、*AC

### ❖ 梯形图



**❖ 梯形图功能说明** 用来比较两个带符号整数 IN1 和 IN2 是否是大于或等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”。

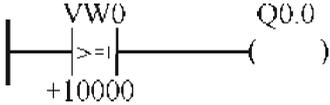
### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数
<IN2>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数

### ❖ 执行范例

◎ 范例 51 ◎ 使用载入字大于或等于指令。

梯 形 图	语 句 表 指 令
	<pre>//将变量内存 VW0 中的数值与 + 10000 相比较， //当 VW0 数值大于或等于 +10000 时，Q0.0 被激活 LDW&gt;=  VW0 +10000 =      Q0.0</pre>

### ❖ 相关指令

LDW<、LDW<=、LDW=、LDW<>、LDW>

## 2.26 AW>=指令：与运算字大于或等于

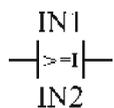
**❖ 指令语法** AW>= <IN1> <IN2>

### ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC
<IN2>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

用来比较两个带符号整数 IN1 和 IN2 是否是大于或等于关系。当比较式为真时，该触点打开（即常开触点闭合），且函数的 RLO 为“1”，并且使用“与”运算将其链接至梯级程序段的 RLO。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数
<IN2>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数

## ❖ 执行范例

●范例 52● 使用与运算字大于或等于指令。

梯 形 图	语 句 表 指 令
	<pre>//将变量内存 VW0 中的数值与+10000 //相比较，当 VW0 数值大于或等于 //+10000 并且 I0.0 打开时，Q0.0 被激活 LD      I0.0 AW&gt;=   VW0 +10000 =       Q0.0</pre>

## ❖ 相关指令

AW<、AW<=、AW=、AW<>、AW>

## 2.27

## OW&gt;=指令：或运算字大于或等于

## ❖ 指令语法

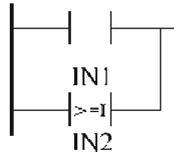
OW>= <IN1> <IN2>

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC
<IN2>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC

### ❖ 梯形图



### ❖ 梯形图功能说明

用来比较两个字 IN1 和 IN2 是否是大于或等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“或”运算将其链接至梯级程序段的 RLO。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数
<IN2>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数

### ❖ 执行范例

● 范例 53 ● 使用或运算字大于或等于指令。

梯 形 图	语 句 表 指 令
	<pre>//将变量内存 VW0 中的数值与+ 10000 相比较，当 VW0 数值大于或等于+10000 或者 I0.0 打开时，Q0.0 被激活 LD      I0.0 OW&gt;=   VW0 +10000 =       Q0.0</pre>

### ❖ 相关指令

OW<、OW<=、OW=、OW<>、OW>

## 2.28 LDW<=指令：载入字小于或等于

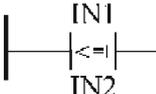
❖ 指令语法 LDW<= <IN1> <IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	整数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC
<IN2>	整数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明 用来比较两个带符号整数 IN1 和 IN2 是否是小于或等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”。

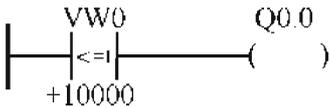
❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	整数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数
<IN2>	整数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数

❖ 执行范例

◎ 范例 54 ◎ 使用载入字小于或等于指令。

梯 形 图	语 句 表 指 令
	<pre>//将变量内存 VW0 中的数值与 + 10000 相比较， //当 VW0 数值小于或等于 +10000 时，Q0.0 被激活 LDW&lt;=  VW0 +10000 =      Q0.0</pre>

## ❖ 相关指令

LDW&lt;、LDW&gt;=、LDW=、LDW&lt;&gt;、LDW&gt;

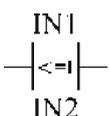
# 2.29 AW<=指令：与运算字小于或等于

❖ 指令语法 AW&lt;= &lt;IN1&gt; &lt;IN2&gt;

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC
<IN2>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC

❖ 梯形图 

❖ 梯形图功能说明 用来比较两个带符号整数 IN1 和 IN2 是否是小于或等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“与”运算将其链接至梯级程序段的 RLO。

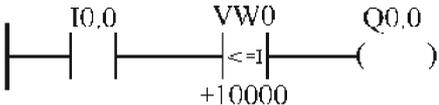
## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数
<IN2>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数

## ❖ 执行范例

○ 范例 55 ○ 使用与运算字小于或等于指令。

梯形图	语句表指令
	<p>//将变量内存 VW0 中的数值与 +10000 相比较, 当 VW0 数值小于或等于 +10000 并且 I0.0 打开时, Q0.0 被激活</p> <pre>LD    I0.0 AW&lt;=  VW0 +10000 =      Q0.0</pre>

### ❖ 相关指令

AW<、AW>=、AW=、AW<>、AW>

## 2.30 OW<=指令：或运算字小于或等于

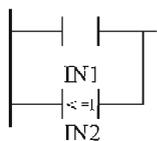
❖ 指令语法  $OW<= <IN1> <IN2>$

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	整数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC
<IN2>	整数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

比较两个字 IN1 和 IN2 是否是小于或等于关系。当比较式为真时, 该触点打开 (即常开触电闭合), 且函数的 RLO 为“1”, 并且使用“或”运算将其链接至梯级程序段的 RLO。

❖ 梯形图参数说明

参数的具体说明如下：

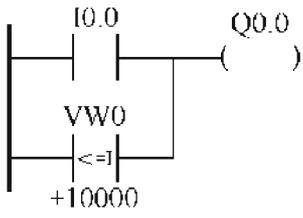
参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	整数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数

续表

参 数	数 据 类 型	内 存 区 域	说 明
<IN2>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数

## ❖ 执行范例

● 范例 56 ● 使用或运算字小于或等于指令。

梯 形 图	语 句 表 指 令
	<pre>//将变量内存 VW0 中的数值与 + 10000 相比较， //当 VW0 数值小于或等于 +10000 或者 I0.0 打开 //时，Q0.0 被激活 LD    I0.0 OW&lt;=  VW0 +10000 =     Q0.0</pre>

## ❖ 相关指令

OW<、OW>=、OW=、OW<>、OW>

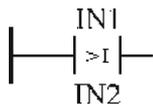
## 2.31 LDW>指令：载入字大于

❖ 指令语法 LDW> <IN1> <IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC
<IN2>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC

❖ 梯形图 

❖ 梯形图功能说明

用来比较两个带符号整数 IN1 和 IN2 是否是大于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”。

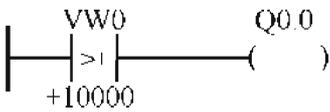
## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数
<IN2>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数

## ❖ 执行范例

◎ 范例 57 ◎ 使用载入字大于指令。

梯 形 图	语 句 表 指 令
	<p>//将变量内存 VW0 中的数值与 +10000 相比较，当 VW0 数值大于 +10000 时，Q0.0 被激活</p> <pre>LDW&gt;  VW0 +10000 =      Q0.0</pre>

## ❖ 相关指令

LDW<、LDW>=、LDW=、LDW<>、LDW<=

## 2.32 AW>指令：与运算字大于

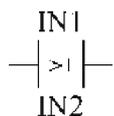
❖ 指令语法 AW> <IN1> <IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC
<IN2>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC

❖ 梯形图



## ❖ 梯形图功能说明

用来比较两个带符号整数 IN1 和 IN2 是否是大于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“与”运算将其链接至梯级程序段的 RLO。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数
<IN2>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数

## ❖ 执行范例

● 范例 58 ● 使用与运算字大于指令。

梯 形 图	语 句 表 指 令
	<pre>//将变量内存 VW0 中的数值与 +10000 //相比较，当 VW0 数值大于 +10000 并且 //I0.0 打开时，Q0.0 被激活 LD      I0.0 AW&gt;    VW0 +10000 =       Q0.0</pre>

## ❖ 相关指令

AW<、AW>=、AW=、AW<>、AW<=

## 2.33 OW>指令：或运算字大于

## ❖ 指令语法

OW> <IN1> <IN2>

## ❖ 指令参数说明

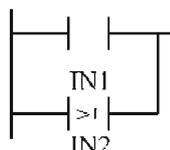
参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC

续表

地 址	数 据 类 型	存 储 区
<IN2>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

用来比较两个字 IN1 和 IN2 是否是大于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“或”运算将其链接至梯级程序段的 RLO。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数
<IN2>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数

## ❖ 执行范例

● 范例 59 ● 使用或运算字大于指令。

梯 形 图	语 句 表 指 令
	<pre>//将变量内存 VW0 中的数值与+ 10000 相比较，当 VW0 数值大于+ 10000 或者 I0.0 打开时，Q0.0 被激活 LD      I0.0 OR&gt;    VW0 +10000 =      Q0.0</pre>

## ❖ 相关指令

OW<、OW>=、OW=、OW<>、OW<=

## 2.34 LDW<指令：载入字小于

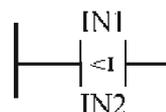
❖ 指令语法 LDW> <IN1> <IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC
<IN2>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

用来比较两个带符号整数 IN1 和 IN2 是否是小于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数
<IN2>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数

❖ 执行范例

◎ 范例 60 ◎ 使用载入字小于指令。

梯 形 图	语 句 表 指 令
	<pre>//将变量内存 VW0 中的数值与 + 10000 相比较， //当 VW0 数值小于 + 10000 时，Q0.0 被激活 LDW&gt;  VW0 +10000 =      Q0.0</pre>

## ❖ 相关指令

LDW&gt;、LDW&gt;=、LDW=、LDW&lt;&gt;、LDW&lt;=

## 2.35 AW&lt;指令：与运算字小于

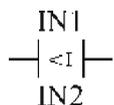
❖ 指令语法 AW&lt;&lt;IN1&gt; &lt;IN2&gt;

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC
<IN2>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

用来比较两个带符号整数 IN1 和 IN2 是否是小于关系。当比较式为真的时候，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“与”运算将其链接至梯级程序段的 RLO。

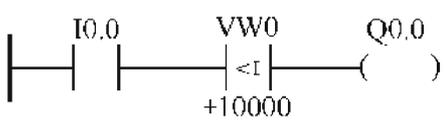
## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数
<IN2>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数

## ❖ 执行范例

◎ 范例 61 ◎ 使用与运算字小于指令。

梯 形 图	语句表指令
	<p>//将变量内存 VW0 中的数值与 +10000 相比较，当 VW0 数值大于 +10000 并且 I0.0 打开时，Q0.0 被激活</p> <pre>LD    I0.0 AW&lt;  VW0 +10000 =     Q0.0</pre>

### ❖ 相关指令

AW>、AW>=、AW=、AW<>、AW<=

## 2.36 OW<指令：或运算字小于

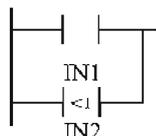
❖ 指令语法 OW< <IN1> <IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	整数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC
<IN2>	整数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明 用来比较两个字 IN1 和 IN2 是否是小于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“或”运算将其链接至梯级程序段的 RLO。

❖ 梯形图参数说明

参数的具体说明如下：

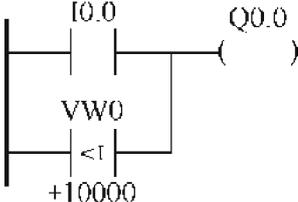
参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	整数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数

续表

参 数	数 据 类 型	内 存 区 域	说 明
<IN2>	整 数	W、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、常数、*VD、*LD、*AC	带符号整数

## ❖ 执行范例

● 范例 62 ● 使用或运算字小于指令。

梯 形 图	语 句 表 指 令
	<pre>// 将变量内存 VW0 中的数值与 + 10000 相比较， // 当 VW0 数值小于 +10000 或者 I0.0 打开时，Q0.0 // 被激活 LD    I0.0 OW&lt;  VW0 +10000 =     Q0.0</pre>

## ❖ 相关指令

OW>、OW>=、OW=、OW<>、OW<=

## 2.37 LDD=指令：载入双字等于

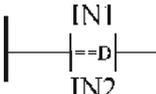
❖ 指令语法 LDD= <IN1> <IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明 用来比较两个带符号双字整数 IN1 和 IN2 是否是等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”。

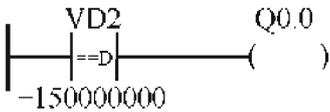
### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数

### ❖ 执行范例

⊙ 范例 63 ⊙ 使用载入双字等于指令。

梯 形 图	语 句 表 指 令
	<p>//将变量内存 VD2 中的数值与-150000000 相比较，当 VD2 数值等于-150000000 时，Q0.0 被激活</p> <pre>LDD=  VD2-150000000 =      Q0.0</pre>

### ❖ 相关指令

LDD>、LDD>=、LDD<、LDD<>、LDD<=

## 2.38 AD=指令：与运算双字等于

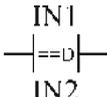
❖ 指令语法 AD= <IN1> <IN2>

### ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC

❖ 梯形图



❖ **梯形图功能说明** 用来比较两个带符号双字整数 IN1 和 IN2 是否是等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“与”运算将其链接至梯级程序段的 RLO。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数

### ❖ 执行范例

◎ 范例 64 ◎ 使用与运算双字等于指令。

梯 形 图	语 句 表 指 令
	<pre>// 将变量内存 VD2 中的数值与 -150000000 相比较，当 VD2 数值等于 -150000000 并且 I0.0 打开时，Q0.0 被激活 LD      I0.0 AD=     VD2 -150000000 =       Q0.0</pre>

### ❖ 相关指令

AD>、AD>=、AD<、AD<>、AD<=

## 2.39 OD=指令：或运算双字等于

❖ **指令语法** OD= <IN1> <IN2>

### ❖ 指令参数说明

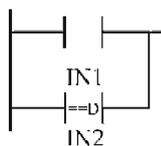
参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC

续表

地 址	数 据 类 型	存 储 区
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

用来比较两个带符号双字整数 IN1 和 IN2 是否是等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“或”运算将其链接至梯级程序段的 RLO。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数

## ❖ 执行范例

● 范例 65 ● 使用或运算双字等于指令。

梯 形 图	语 句 表 指 令
	<pre>//将变量内存 VD2 中的数值与-150000000 相比较,当 VD2 数值等于-150000000 或者 I0.0 打开时, Q0.0 被 激活 LD      I0.0 OD=     VD2 -150000000 =       Q0.0</pre>

## ❖ 相关指令

OD>、OD>=、OD>、OD<>、OD<=

# 2.40

## LDD<>指令：载入双字不等于

## ❖ 指令语法

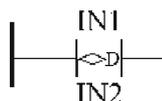
LDD<> <IN1> <IN2>

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

用来比较两个带符号双字整数 IN1 和 IN2 是否是不等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数

## ❖ 执行范例

● 范例 66 ● 使用载入双字不等于指令。

梯 形 图	语 句 表 指 令
	<pre>//将变量内存 VD2 中的数值与-150000000 相比较， //当 VD2 数值不等于-150000000 时，Q0.0 被激活 LDD&lt;&gt;  VD2 -150000000 =      Q0.0</pre>

## ❖ 相关指令

LDD>、LDD>=、LDD<、LDD=、LDD<=

## 2.41

## AD&lt;&gt;指令：与运算双字不等于

## ❖ 指令语法

AD<> <IN1> <IN2>

### ◆ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC

### ◆ 梯形图



### ◆ 梯形图功能说明

用来比较两个带符号双字整数 IN1 和 IN2 是否是不等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“与”运算将其链接至梯级程序段的 RLO。

### ◆ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数

### ◆ 执行范例

◎ 范例 67 ◎ 使用与运算双字不等于指令。

梯 形 图	语 句 表 指 令
	<pre>// 将变量内存 VD2 中的数值与 -150000000 相比较，当 VD2 数值不等于 -150000000 并且 I0.0 打开时，Q0.0 被 激活 LD      I0.0 AD&lt;&gt;   VD2 -150000000 =       Q0.0</pre>

### ◆ 相关指令

AD>、AD>=、AD<、AD=、AD<=

## 2.42 OD<>指令：或运算双字不等于

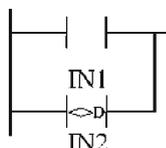
❖ 指令语法 OD<> <IN1> <IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

用来比较两个带符号双字整数 IN1 和 IN2 是否是不等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“或”运算将其链接至梯级程序段的 RLO。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数

❖ 执行范例

◎ 范例 68 ◎ 使用或运算双字不等于指令。

梯 形 图	语 句 表 指 令
	<p>//将变量内存 VD2 中的数值与-150000000 相比较，当 VD2 数值不等于-150000000 或者 I0.0 打开时，Q0.0 被激活</p> <pre>LD      I0.0 OD&lt;&gt;   VD2 -150000000 =       Q0.0</pre>

## ❖ 相关指令

OD&gt;、OD&gt;=、OD&gt;、OD=、OD&lt;=

# 2.43 LDD>=指令：载入双字大于或等于

❖ 指令语法 LDD&gt;= &lt;IN1&gt; &lt;IN2&gt;

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC



❖ 梯形图功能说明 用来比较两个带符号双字整数 IN1 和 IN2 是否是大于或等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数

## ❖ 执行范例

◎ 范例 69 ◎ 使用载入双字大于或等于指令。

梯 形 图	语 句 表 指 令
	<p>//将变量内存 VD2 中的数值与-150000000 相比较,当 VD2 数值大于或等于-150000000 时,Q0.0 被激活</p> <pre>LDD&gt;=   VD2 -150000000 =       Q0.0</pre>

## ❖ 相关指令

LDD&gt;、LDD&lt;&gt;、LDD&lt;、LDD=、LDD&lt;=

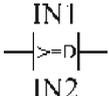
## 2.44 AD>=指令：与运算双字大于或等于

❖ 指令语法 AD&gt;= &lt;IN1&gt; &lt;IN2&gt;

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC

❖ 梯形图 

❖ 梯形图功能说明 用来比较两个带符号双字整数 IN1 和 IN2 是否是大于或等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“与”运算将其链接至梯级程序段的 RLO。

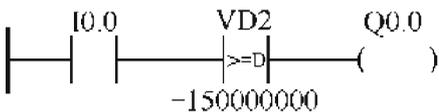
## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数

## ❖ 执行范例

○ 范例 70 ○ 使用与运算双字大于或等于指令。

梯 形 图	语句表指令
	<p>// 将变量内存 VD2 中的数值与 -150000000 相比较，当 VD2 数值大于或等于 -150000000 并且 I0.0 打开时，Q0.0 被激活</p> <pre>LD      I0.0 AD&gt;=   VD2 -150000000 =       Q0.0</pre>

### ❖ 相关指令

AD>、AD<>、AD<、AD=、AD<=

## 2.45 OD>=指令：或运算双字大于或等于

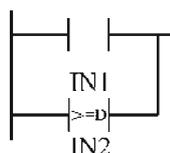
❖ 指令语法  $OD>= <IN1> <IN2>$

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

用来比较两个带符号双字整数 IN1 和 IN2 是否是大于或等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“或”运算将其链接至梯级程序段的 RLO。

❖ 梯形图参数说明

参数的具体说明如下：

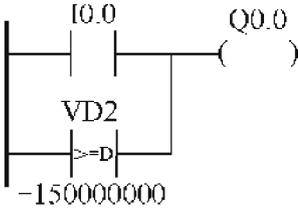
参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数

续表

参 数	数 据 类 型	内 存 区 域	说 明
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数

## ❖ 执行范例

● 范例 71 ● 使用或运算双字大于或等于指令。

梯 形 图	语 句 表 指 令
	<p>//将变量内存 VD2 中的数值与-150000000 相比较,当 VD2 数值大于或等于-150000000 或者 I0.0 打开时, Q0.0 被激活</p> <pre>LD      I0.0 OD&gt;=   VD2 -150000000 =       Q0.0</pre>

## ❖ 相关指令

OD>、OD<>、OD>、OD=、OD<=

## 2.46 LDD<=指令：载入双字小于或等于

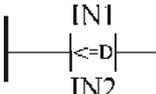
❖ 指令语法 LDD<= <IN1> <IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明 用来比较两个带符号双字整数 IN1 和 IN2 是否是小于或等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”。

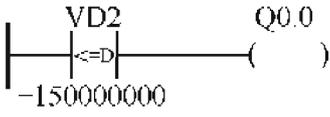
### ◆ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数

### ◆ 执行范例

◎ 范例 72 ◎ 使用载入双字小于或等于指令。

梯 形 图	语 句 表 指 令
	<p>//将变量内存 VD2 中的数值与-150000000 相比较，当 VD2 数值小于或等于-150000000 时，Q0.0 被激活</p> <pre>LDD&lt;=  VD2 -150000000 =      Q0.0</pre>

### ◆ 相关指令

LDD>、LDD<>、LDD<、LDD=、LDD>=

## 2.47 AD<=指令：与运算双字小于或等于

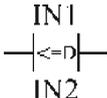
◆ 指令语法 AD<= <IN1> <IN2>

### ◆ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC

◆ 梯形图



◆ 梯形图功能说明 用来比较两个带符号双字节整数 IN1 和 IN2

是否是小于或等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的RLO为“1”，并且使用“与”运算将其链接至梯级程序段的RLO。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数

### ❖ 执行范例

#### ◎ 范例 73 ◎ 使用与运算双字小于或等于指令。

梯 形 图	语 句 表 指 令
	<pre>// 将变量内存 VD2 中的数值与 -150000000 相比较，当 VD2 数值小于或 等于-150000000 并且 I0.0 打开时，Q0.0 被激活 LD      I0.0 AD&lt;=   VD2 -150000000 =       Q0.0</pre>

### ❖ 相关指令

AD>、AD<>、AD<、AD=、AD>=

## 2.48 OD<=指令：或运算双字小于或等于

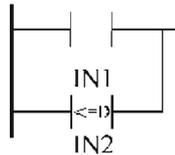
❖ 指令语法 OD<= <IN1> <IN2>

### ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

用来比较两个带符号双字整数 IN1 和 IN2 是否是小于或等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“或”运算将其链接至梯级程序段的 RLO。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数

## ❖ 执行范例

● 范例 74 ● 使用或运算双字小于或等于指令。

梯 形 图	语 句 表 指 令
	<pre>//将变量内存 VD2 中的数值与-150000000 相比较,当 VD2 数值小于或等于-150000000 或者 I0.0 打开时, Q0.0 被激活 LD      I0.0 OD&lt;=   VD2 -150000000 =       Q0.0</pre>

## ❖ 相关指令

OD>、OD<>、OD>、OD=、OD>=

## 2.49

### LDD>指令：载入双字大于

## ❖ 指令语法

LDD> <IN1> <IN2>

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC



❖ 梯形图功能说明 用来比较两个带符号双字整数 IN1 和 IN2 是否是大于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数

❖ 执行范例

○ 范例 75 ○ 使用载入双字大于指令。

梯 形 图	语 句 表 指 令
	<pre>//将变量内存 VD2 中的数值与-150000000 相比较，当 VD2 数值大于-150000000 时，Q0.0 被激活 LDD&gt;  VD2 -150000000 =      Q0.0</pre>

❖ 相关指令

LDD<=、LDD<>、LDD<、LDD=、LDD>=

## 2.50 AD>指令：与运算双字大于

❖ 指令语法 AD> <IN1> <IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC



◆ 梯形图功能说明 用来比较两个带符号双字整数 IN1 和 IN2 是否是大于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“与”运算将其链接至梯级程序段的 RLO。

### ◆ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数

### ◆ 执行范例

◎ 范例 76 ◎ 使用与运算双字大于指令。

梯 形 图	语 句 表 指 令
	<pre>//将变量内存 VD2 中的数值与-150000000 相比较，当 VD2 数值大于-150000000 并且 I0.0 打开时，Q0.0 被激活 LD      I0.0 AD&gt;    VD2 -150000000 =       Q0.0</pre>

### ◆ 相关指令

AD<=、AD<>、AD<、AD=、AD>=

## 2.51 OD>指令：或运算双字大于

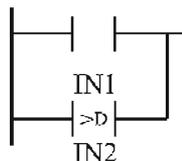
◆ 指令语法 OD> <IN1> <IN2>

## ◆ 指令参数说明

参数的具体说明如下:

地 址	数 据 类 型	存 储 区
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC

## ◆ 梯形图



## ◆ 梯形图功能说明

用来比较两个带符号双字整数 IN1 和 IN2 是否是大于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“或”运算将其链接至梯级程序段的 RLO。

## ◆ 梯形图参数说明

参数的具体说明如下:

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数

## ◆ 执行范例

◎ 范例 77 ◎ 使用或运算双字大于指令。

梯 形 图	语 句 表 指 令
	<pre>//将变量内存 VD2 中的数值与-150000000 相比较,当 VD2 数值大于-150000000 或者 I0.0 打开时, Q0.0 被 激活 LD      I0.0 OD&gt;    VD2 -150000000 =       Q0.0</pre>

## ◆ 相关指令

OD<=、OD<>、OD>、OD=、OD>=

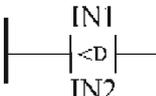
## 2.52 LDD<指令：载入双字小于

❖ 指令语法 LDD< <IN1> <IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC

❖ 梯形图 

❖ 梯形图功能说明 用来比较两个带符号双字整数 IN1 和 IN2 是否是小于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”。

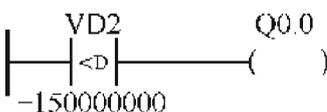
❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数

❖ 执行范例

◎ 范例 78 ◎ 使用载入双字小于指令。

梯 形 图	语 句 表 指 令
	<pre>//将变量内存 VD2 中的数值与-150000000 相比较，当 VD2 数值小于-150000000 时，Q0.0 被激活 LDD&lt;  VD2 -150000000 =      Q0.0</pre>

❖ 相关指令

LDD<=、LDD<>、LDD>、LDD=、LDD>=

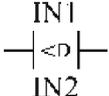
## 2.53 AD<指令：与运算双字小于

◆ 指令语法 AD< <IN1> <IN2>

◆ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC

◆ 梯形图 

◆ 梯形图功能说明 用来比较两个带符号双字整数 IN1 和 IN2 是否是小于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“与”运算将其链接至梯级程序段的 RLO。

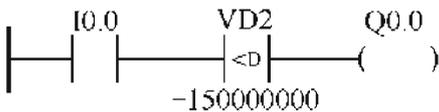
◆ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数

◆ 执行范例

◎ 范例 79 ◎ 使用与运算双字小于指令。

梯 形 图	语 句 表 指 令
	<pre>//将变量内存 VD2 中的数值与-150000000 相比较， //当 VD2 数值小于-150000000 并且 I0.0 打开时，Q0.0 //被激活 LD      I0.0 AD&lt;    VD2 -150000000 =      Q0.0</pre>

## ❖ 相关指令

AD&lt;=、AD&lt;&gt;、AD&gt;、AD=、AD&gt;=

# 2.54 OD<指令：或运算双字小于

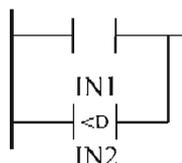
❖ 指令语法 OD&lt; &lt;IN1&gt; &lt;IN2&gt;

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

用来比较两个带符号双字整数 IN1 和 IN2 是否是小于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“或”运算将其链接至梯级程序段的 RLO。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数
<IN2>	双字节整数	ID、QD、MD、SD、SMD、VD、LD、HC、AC、常数、*VD、*LD、*AC	带符号长整数

## ❖ 执行范例

○ 范例 80 ○ 使用或运算双字小于指令。

梯形图	语句表指令
	<p>//将变量内存 VD2 中的数值与-150000000 相比较,当 VD2 数值小于-150000000 或者 I0.0 打开时, Q0.0 被激活</p> <pre>LD      I0.0 OD&lt;    VD2 -150000000 =       Q0.0</pre>

### ❖ 相关指令

OD<=、OD<>、OD>、OD=、OD>=

## 2.55 LDR=指令：载入实数等于

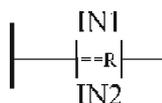
❖ 指令语法 LDR= <IN1> <IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

用来比较两个实数 IN1 和 IN2 是否是等于关系。当比较式为真时,该触点打开(即常开触电闭合),且函数的 RLO 为“1”。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值

❖ 执行范例

### 范例 81 使用载入实数等于指令。

梯形图	语句表指令
	<pre>//将变量内存 VD6 中的数值与 5.001E-006 相比较，当 VD6 数值等于 5.001E-006 时，Q0.0 被激活 LDR=   VD6 5.001E-006 =      Q0.0</pre>

#### 相关指令

LDR<=、LDR<>、LDR>、LDR<、LDR>=

## 2.56 AR=指令：与运算实数等于

指令语法 AR= <IN1> <IN2>

指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC

梯形图

梯形图功能说明 用来比较两个实数 IN1 和 IN2 是否是等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“与”运算将其链接至梯级程序段的 RLO。

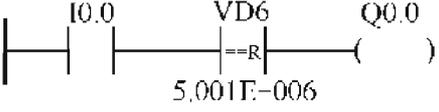
梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值

## ❖ 执行范例

## ● 范例 82 ● 使用与运算实数等于指令。

梯 形 图	语句表指令
	<p>//将变量内存 VD6 中的数值与 5.00E-006 相比较，当 VD6 数值等于 5.00E-006 并且 I0.0 打开时，Q0.0 被激活</p> <pre>LD      I0.0 AR=    VD6 5.00E-006 =      Q0.0</pre>

## ❖ 相关指令

AR&lt;=、AR&lt;&gt;、AR&gt;、AR&lt;、AR&gt;=

## 2.57 OR=指令：或运算实数等于

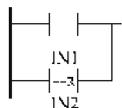
❖ 指令语法 OR= &lt;IN1&gt; &lt;IN2&gt;

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

用来比较两个实数 IN1 和 IN2 是否是等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“或”运算将其链接至梯级程序段的 RLO。

❖ 梯形图参数说明

参数的具体说明如下：

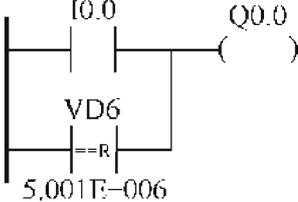
参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值

续表

参 数	数 据 类 型	内 存 区 域	说 明
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、 常数、*VD、*LD、*AC	输入值

## ❖ 执行范例

● 范例 83 ● 使用或运算实数等于指令。

梯 形 图	语 句 表 指 令
	<p>//将变量内存 VD6 中的数值与 5.001E-006 相比较， 当 VD6 数值等于 5.001E-006 或者 I0.0 打开时，Q0.0 被激活</p> <pre>LD      I0.0 OR=     VD6 5.001E-006 =       Q0.0</pre>

## ❖ 相关指令

OR<=、OR<>、OR>、OR<、OR>=

## 2.58 LDR<>指令：载入实数不等于

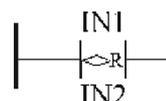
❖ 指令语法 LDR<> <IN1> <IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、 *LD、*AC
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、 *LD、*AC

❖ 梯形图



❖ 梯形图功能说明

用来比较两个实数 IN1 和 IN2 是否是不等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”。

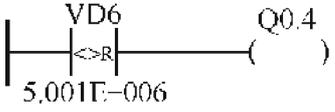
### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、 常数、*VD、*LD、*AC	输入值
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、 常数、*VD、*LD、*AC	输入值

### ❖ 执行范例

◎ 范例 84 ◎ 使用载入实数不等于指令。

梯 形 图	语 句 表 指 令
	//将变量内存 VD6 中的数值与 5.001E-006 相比较，当 VD6 数值不等于 5.001E-006 时，Q0.4 被激活 LDR<> VD6 5.001E-006 = Q0.4

### ❖ 相关指令

LDR<=、LDR=、LDR>、LDR<、LDR>=

## 2.59 AR<>指令：与运算实数不等于

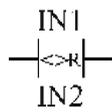
❖ 指令语法 AR<> <IN1> <IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、 *VD、*LD、*AC
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、 *VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明 用来比较两个实数 IN1 和 IN2 是否是不等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO

为“1”，并且使用“与”运算将其链接至梯级程序段的RLO。

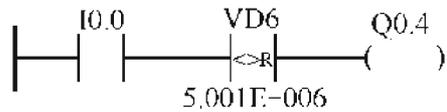
### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值

### ❖ 执行范例

◎ 范例 85 ◎ 使用与运算实数不等于指令。

梯 形 图	语 句 表 指 令
	<pre>//将变量内存 VD6 中的数值与 5.00E-006 相比较，当 VD6 数值不等于 5.00E-006 并且 I0.0 打开时，Q0.4 被激活 LD      I0.0 AR&lt;&gt;   VD6 5.00E-006 =       Q0.4</pre>

### ❖ 相关指令

AR<=、AR=、AR>、AR<、AR>=

## 2.60 OR<>指令：或运算实数不等于

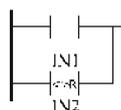
❖ 指令语法 OR<> <IN1> <IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC

❖ 梯形图



❖ **梯形图功能说明** 用来比较两个实数 IN1 和 IN2 是否是不等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“或”运算将其链接至梯级程序段的 RLO。

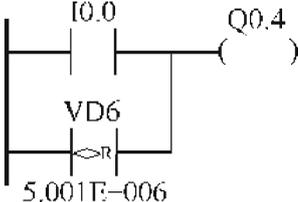
### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值

### ❖ 执行范例

⊙ 范例 86 ⊙ 使用或运算实数不等于指令。

梯 形 图	语 句 表 指 令
	<pre>//将变量内存 VD6 中的数值与 5.001E-006 相比较， //当 VD6 数值不等于 5.001E-006 或者 I0.0 打开时，Q0.0 //被激活 LD      I0.0 OR&lt;&gt;   VD6 5.001E-006 =       Q0.4</pre>

### ❖ 相关指令

OR<=、OR=、OR>、OR<、OR>=

## 2.61 LDR>=指令：载入实数大于或等于

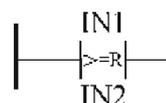
❖ **指令语法** LDR>= <IN1> <IN2>

### ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

用来比较两个实数 IN1 和 IN2 是否是大于或等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值

## ❖ 执行范例

◎ 范例 87 ◎ 使用载入实数大于或等于指令。

梯 形 图	语 句 表 指 令
	<pre>//将变量内存 VD6 中的数值与 5.001E-006 相比较，当 VD6 数值大于或等于 5.001E-006 时，Q0.4 被激活 LDR&gt;=  VD6 5.001E-006 =      Q0.4</pre>

## ❖ 相关指令

LDR<=、LDR=、LDR>、LDR<、LDR<>

## 2.62

## AR&gt;=指令：与运算实数大于或等于

## ❖ 指令语法

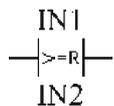
AR>= <IN1> <IN2>

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

用来比较两个实数 IN1 和 IN2 是否是大于或等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“与”运算将其链接至梯级程序段的 RLO。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值

## ❖ 执行范例

◎ 范例 88 ◎ 使用与运算实数大于或等于指令。

梯 形 图	语 句 表 指 令
	<p>//将变量内存 VD6 中的数值与 5.00E-006 相比较，当 VD6 数值大于或等于 5.00E-006 并且 I0.0 打开时，Q0.4 被激活</p> <pre>LD      I0.0 AR&gt;=   VD6 5.00E-006 =       Q0.4</pre>

## ❖ 相关指令

AR<=、AR=、AR>、AR<、AR<>

## 2.63 OR>=指令：或运算实数大于或等于

## ❖ 指令语法

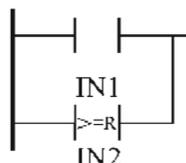
OR>= <IN1> <IN2>

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC

### ❖ 梯形图



### ❖ 梯形图功能说明

用来比较两个实数 IN1 和 IN2 是否是大于或等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“或”运算将其链接至梯级程序段的 RLO。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值

### ❖ 执行范例

◎ 范例 89 ◎ 使用或运算实数大于或等于指令。

梯 形 图	语 句 表 指 令
	<p>//将变量内存 VD6 中的数值与 5.001E-006 相比较，当 VD6 数值大于或等于 5.001E-006 或者 I0.0 打开时，Q0.0 被激活</p> <pre>LD      I0.0 OR&gt;=   VD6 5.001E-006 =       Q0.4</pre>

### ❖ 相关指令

OR<=、OR=、OR>、OR<、OR<>

## 2.64 LDR<=指令：载入实数小于或等于

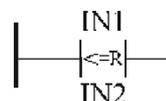
❖ 指令语法 LDR<= <IN1> <IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

用来比较两个实数 IN1 和 IN2 是否是小于或等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值

❖ 执行范例

◎ 范例 90 ◎ 使用载入实数小于或等于指令。

梯 形 图	语 句 表 指 令
	<pre>//将变量内存 VD6 中的数值与 5.001E-006 相比较，当 VD6 数值小于或等于 5.001E-006 时，Q0.4 被激活 LDR&lt;=  VD6 5.001E-006 =      Q0.4</pre>

## ❖ 相关指令

LDR&gt;=、LDR=、LDR&gt;、LDR&lt;、LDR&lt;&gt;

# 2.65 AR<=指令：与运算实数小于或等于

❖ 指令语法 AR&lt;= &lt;IN1&gt; &lt;IN2&gt;

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC

❖ 梯形图

❖ 梯形图功能说明 用来比较两个实数 IN1 和 IN2 是否是小于或等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“与”运算将其链接至梯级程序段的 RLO。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值

## ❖ 执行范例

◎ 范例 91 ◎ 使用与运算实数小于或等于指令。

梯 形 图	语 句 表 指 令
	<pre>//将变量内存 VD6 中的数值与 5.00E-006 相比较， //当 VD6 数值小于或等于 5.00E-006 并且 I0.0 打开 //时，Q0.4 被激活 LD      I0.0 AR&lt;=   VD6 5.00E-006 =       Q0.4</pre>

## ◆ 相关指令

AR&gt;=、AR=、AR&gt;、AR&lt;、AR&lt;&gt;

## 2.66 OR<=指令：或运算实数小于或等于

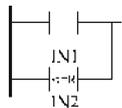
◆ 指令语法 OR&lt;= &lt;IN1&gt; &lt;IN2&gt;

## ◆ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC

## ◆ 梯形图



## ◆ 梯形图功能说明

用来比较两个实数 IN1 和 IN2 是否是小于或等于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“或”运算将其链接至梯级程序段的 RLO。

## ◆ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值

## ◆ 执行范例

◎ 范例 92 ◎ 使用或运算实数小于或等于指令。

梯 形 图	语 句 表 指 令
	<p>//将变量内存 VD6 中的数值与 5.001E-006 相比较，当 VD6 数值小于或等于 5.001E-006 或者 I0.0 打开时，Q0.4 被激活</p> <pre>LD      I0.0 OR&lt;=   VD6 5.001E-006 =       Q0.4</pre>

## ❖ 相关指令

OR&gt;=、OR=、OR&gt;、OR&lt;、OR&lt;&gt;

## 2.67 LDR>指令：载入实数大于

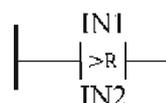
❖ 指令语法 LDR&gt; &lt;IN1&gt; &lt;IN2&gt;

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

用来比较两个实数 IN1 和 IN2 是否是大于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值

## ❖ 执行范例

● 范例 93 ● 使用载入实数大于指令。

梯 形 图	语 句 表 指 令
	<pre>//将变量内存 VD6 中的数值与 5.001E-006 相比较，当 VD6 数值大于 5.001E-006 时，Q0.4 被激活 LDR&gt;   VD6 5.001E-006 =      Q0.4</pre>

## ❖ 相关指令

LDR&gt;=、LDR=、LDR&lt;=、LDR&lt;、LDR&lt;&gt;

## 2.68 AR&gt;指令：与运算实数大于

❖ 指令语法 AR&gt; &lt;IN1&gt; &lt;IN2&gt;

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC

❖ 梯形图

❖ 梯形图功能说明 用来比较两个实数 IN1 和 IN2 是否是大于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“与”运算将其链接至梯级程序段的 RLO。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值

## ❖ 执行范例

◎ 范例 94 ◎ 使用与运算实数大于指令。

梯 形 图	语 句 表 指 令
	<pre>//将变量内存 VD6 中的数值与 5.00E-006 相比较，当 VD6 数值大于 5.00E-006 并且 I0.0 打开时，Q0.4 被激活 LD      I0.0 AR&gt;    VD6 5.00E-006 =       Q0.4</pre>

## ❖ 相关指令

AR&gt;=、AR=、AR&lt;=、AR&lt;、AR&lt;&gt;

# 2.69 OR>指令：或运算实数大于

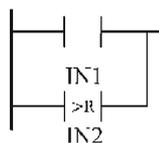
❖ 指令语法 OR&gt; &lt;IN1&gt; &lt;IN2&gt;

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

用来比较两个实数 IN1 和 IN2 是否是大于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“或”运算将其链接至梯级程序段的 RLO。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值

## ❖ 执行范例

○ 范例 95 ○ 使用或运算实数大于指令。

梯形图	语句表指令
	<p>//将变量内存 VD6 中的数值与 5.001E-006 相比较，当 VD6 数值大于 5.001E-006 或者 I0.0 打开时，Q0.4 被激活</p> <pre>LD      I0.0 OR&gt;    VD6 5.001E-006 =       Q0.4</pre>

### ❖ 相关指令

OR>=、OR=、OR<=、OR<、OR<>

## 2.70 LDR<指令：载入实数小于

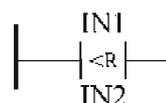
❖ 指令语法 LDR< <IN1> <IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

用来比较两个实数 IN1 和 IN2 是否是小于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”。

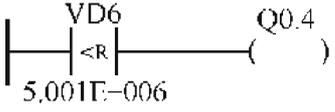
❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值

## ❖ 执行范例

## ● 范例 96 ● 使用载入实数小于指令。

梯 形 图	语句表指令
	<pre>//将变量内存 VD6 中的数值与 5.001E-006 相比较，当 //VD6 数值小于 5.001E-006 时，Q0.4 被激活 LDR&lt;   VD6 5.001E-006 =       Q0.4</pre>

## ❖ 相关指令

LDR&gt;=、LDR=、LDR&lt;=、LDR&gt;、LDR&lt;&gt;

## 2.71 AR<指令：与运算实数小于

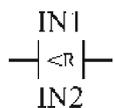
## ❖ 指令语法 AR&lt; &lt;IN1&gt; &lt;IN2&gt;

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

用来比较两个实数 IN1 和 IN2 是否是小于关系。当比较式为真时，该触点打开（即常开触电闭合），且函数的 RLO 为“1”，并且使用“与”运算将其链接至梯级程序段的 RLO。

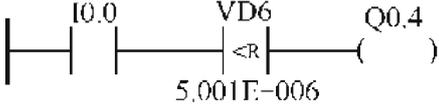
## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值

## ❖ 执行范例

## ● 范例 97 ● 使用与运算实数小于指令。

梯 形 图	语句表指令
	<p>//将变量内存 VD6 中的数值与 5.00E-006 相比较, 当 VD6 数值小于 5.00E-006 并且 I0.0 打开时, Q0.4 被激活</p> <pre>LD      I0.0 AR&lt;    VD6 5.00E-006 =      Q0.4</pre>

## ❖ 相关指令

AR>=、AR=、AR<=、AR>、AR<>

## 2.72 OR<指令：或运算实数小于

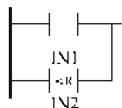
❖ 指令语法 OR< <IN1> <IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

用来比较两个实数 IN1 和 IN2 是否是小于关系。当比较式为真时, 该触点打开 (即常开触电闭合), 且函数的 RLO 为“1”, 并且使用“或”运算将其链接至梯级程序段的 RLO。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值

续表

参 数	数 据 类 型	内 存 区 域	说 明
<IN2>	实数	ID、QD、MD、SD、SMD、VD、LD、AC、常数、*VD、*LD、*AC	输入值

### ❖ 执行范例

☉ 范例 98 ☉ 使用或运算实数小于指令。

梯 形 图	语 句 表 指 令
	<pre>//将变量内存 VD6 中的数值与 5.001E-006 相比较，当 VD6 数值小于 5.001E-006 或者 I0.0 打开时，Q0.4 被激活 LD      I0.0 OR&lt;    VD6 5.001E-006 =      Q0.4</pre>

### ❖ 相关指令

OR>=、OR=、OR<=、OR>、OR<>



## 第3章 转换指令

转换指令读取参数 IN 的内容，然后进行转换或改变其符号。可通过参数 OUT 查询结果。

### 3.1 BTI 指令：字节转换至整数

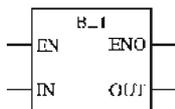
❖ 指令语法 BTI <IN (字节)> <OUT (整数)>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<字节>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<整数>	整数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

把输入端 (IN) 的字节值转换成一个整数 (OUT) 并将结果置入 OUT 指定的变量中。因为字节是无符号的，所以没有符号扩展。使 ENO=0 的错误条件是：SM4.3 (运行时间)；0006 (间接寻址)。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	输入端数据
<OUT>	整数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC	输出端数据

### ❖ 执行范例

● 范例 99 ● 使用字节转换至整数指令。

梯 形 图	语 句 表 指 令
	<p>//当 I0.0 打开时，变量内存 MB0 中的数值被转换成整数，结果被存放在内存 VW2 中</p> <pre>LD    I0.0 BTI  MB0 VW2</pre>

### ❖ 相关指令

ITB

## 3.2 ITB 指令：整数转换至字节

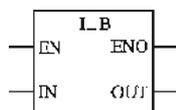
❖ 指令语法 ITB <IN (整数)> <OUT (字节)>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<整数>	整数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC
<字节>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD

❖ 梯形图



❖ 梯形图功能说明

整数转换至字节指令将整数值 (IN) 转换成字节值，并将结果置入 OUT 指定的变量中。数值在 0~255 范围内被转换，其他值转换将导致溢出，输出不受影响。使 ENO = 0 的错误条件是：“SM1.1

(溢出); SM4.3 (运行时间); 0006 (间接寻址)。

### ◆ 梯形图参数说明

参数的具体说明如下:

参 数	数 据 类 型	内 存 区 域	说 明
< IN >	整 数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC	输入端数据
< OUT >	字 节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	输出端数据

### ◆ 执行范例

◎ 范例 100 ◎ 使用整数转换至字节指令。

梯 形 图	语 句 表 指 令
	<p>//当 I0.0 打开时, 内存 VW2 中的数值被转换成字节, 结果被存放在变量内存 MB0 中</p> <pre>LD    I0.0 ITB   VW2 MB0</pre>

### ◆ 相关指令

BTI

## 3.3 ITD 指令：整数转换至长整数

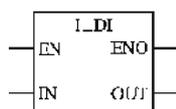
◆ 指令语法 ITD <IN (整数)> <OUT (长整数)>

◆ 指令参数说明

参数的具体说明如下:

地 址	数 据 类 型	存 储 区
<整数>	整 数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC
<长整数>	长 整 数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC

### ◆ 梯形图



**❖ 梯形图功能说明** 整数到长整数转换指令把输入端（IN）的整数转换成一个长整数（OUT），符号进行扩展。使 ENO = 0 的错误条件是：SM4.3（运行时间）；0006（间接寻址）。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
< IN >	整 数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC	输入端数据
< OUT >	长 整 数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC	输出端数据

### ❖ 执行范例

● 范例 101 ● 使用整数转换至长整数指令。

梯 形 图	语 句 表 指 令
	<pre>//当 I0.0 打开时，计数器 C10 //中的值载入 AC1 LD    I0.0 ITD   C10 AC1</pre>

### ❖ 相关指令

DTI

## 3.4 ITS 指令：整数转换至字符串

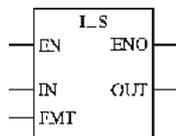
**❖ 指令语法** ITS <IN（整数）> <OUT（字符串）> <FMT（格式）>

### ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<整数>	整 数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC
<FMT（格式）>	字 节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<字符串>	字 符 串	VB、*VD、LB、*AC、*LD

### ❖ 梯形图



### ❖ 梯形图功能说明

整数转换为字符串指令将整数字 IN 转换为长度为 8 个字符的 ASCII 字符串。格式 (FMT) 指定小数点右面的转换精度，无论小数点是显示为逗号还是小数点，结果字符串写入从 OUT 开始的 9 个连续字节中。

字符串是一系列字符，每个字符作为一个字节存储。字符串的第一个字节定义字符串的长度，即字符数。如果常数字符串被直接输入程序编辑器或数据块，那么该字符串必须用双引号字符起始和结束（字符串常数）。

ITS 的格式 (FMT) 操作数定义：

MSB							LSB
7	6	5	4	3	2	1	0
0	0	0	0	c	n	n	n

C 表示逗号 (1) 或小数点 (0)，nnn 表示小数点右面的位数。

输出字符串的长度始终为 8 个字符。输出缓冲区中的小数点右面的位数由 nnn 域指定。nnn 域的有效范围是 0~5。将小数点右面的位数指定为 0 会使值不带小数点。当 nnn 值大于 5 时，输出显示为 8 个 ASCII 空格字符的字符串。c 位指定是使用逗号 (c = 1) 还是使用小数点 (c = 0) 作为整数和小数之间的分隔符。格式最上方的 4 个位必须为 0。

下表显示了几个使用小数点 (c = 0) 和小数点右面 3 位数 (nnn = 011) 格式的值范例。位于 OUT 位置的值是字符串长度。

	OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT
		+1	+2	+3	+4	+5	+6	+7	+8
IN = 12	8				0	.	0	1	2
IN = 123	8				0	.	1	2	3
IN = 1234	8				1	.	2	3	4
IN = 12345	8			1	2	.	3	4	5

设置 ENO = 0 的错误条件是：0006（间接寻址）；0091（操作数范围）；非法格式（nnn>5）。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	整 数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC	输入端数据
<FMT>	字 节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	精度格式
<OUT>	字 符 串	VB、*VD、LB、*AC、*LD	输出端数据

### ❖ 执行范例

● 范例 102 ● 使用整数转换为字符串指令。

梯 形 图	语 句 表 指 令
	<pre>//当 I0.0 打开，整数 123 被转换成字符串“0.123”放入以 VB1 开始的 9 个连续字节中 LD    I0.0 ITS   123 VB1 3</pre>

### ❖ 相关指令

STI

## 3.5 DTI 指令：长整数转换至整数

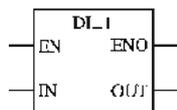
❖ 指令语法 DTI<IN（长整数）><OUT（整数）>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<长整数>	长整数	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*LD、*AC
<整数>	整数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

将长整数值 (IN) 转换成整数值, 并将结果置入 OUT 指定的变量中。如果转换的值过大, 则无法在输出中表示, 设置溢出位, 输出不受影响。使 ENO = 0 的错误条件是: SM1.1 (溢出); SM4.3 (运行时间); 0006 (间接寻址)。

## ❖ 梯形图参数说明

参数的具体说明如下:

参 数	数 据 类 型	内 存 区 域	说 明
< IN >	长整数	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*LD、*AC	输入端数据
< OUT >	整数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC	输出端数据

## ❖ 执行范例

● 范例 103 ● 使用长整数转换至整数指令。

梯 形 图	语 句 表 指 令
	<pre>//当 I0.0 打开时, AC1 中的值 载入计数器 C10 LD    I0.0 DTI   AC1 C10</pre>

## ❖ 相关指令

ITD

## 3.6 DTR 指令: 长整数转换至实数

## ❖ 指令语法

DTR <IN (长整数)> <OUT (实数)>

## ❖ 指令参数说明

参数的具体说明如下:

地 址	数 据 类 型	存 储 区
<长整数>	长整数	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*LD、*AC

续表

地 址	数 据 类 型	存 储 区
<实数>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、 *VD、*LD、*AC



❖ 梯形图功能说明 将 32 位有符号整数 (IN) 转换成 32 位实数 (OUT)，使 ENO = 0 的错误条件是：SM4.3 (运行时间)；0006 (间接寻址)。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
< IN >	长整数	VD、ID、QD、MD、SD、SMD、LD、 HC、AC、常数、*VD、*LD、*AC	输入端数据
< OUT >	实数	VD、ID、QD、MD、SD、SMD、LD、 AC、*VD、*LD、*AC	输出端数据

❖ 执行范例

◎ 范例 104 ◎ 使用长整数转换至实数指令。

梯 形 图	语 句 表 指 令
	<pre>//当 I0.0 打开时，AC1 中 的值载入 VD0 LD      I0.0 DTR    AC1 VD0</pre>

## 3.7 DTS 指令：长整数转换至字符串

❖ 指令语法 DTS <IN (长整数)> <OUT (字符串)> <FMT (格式)>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<长整数>	长整数	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*LD、*AC
<FMT (格式)>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<字符串>	字符串	VB、*VD、LB、*AC、*LD



◆ 梯形图功能说明 长整数转换为字符串指令将双整数 IN 转换为长度为 12 个字符的 ASCII 字符串。格式 (FMT) 指定小数点右面的转换精度，无论小数点是显示为逗号还是小数点。结果字符串写入从 OUT 开始的 13 个连续字节中。

DTS 的格式 (FMT) 操作数定义：

MSB							LSB
7	6	5	4	3	2	1	0
0	0	0	0	C	n	n	n

C 表示逗号 (1) 或小数点 (0)，nnn 表示小数点右面的位数。

输出字符串的长度始终为 8 个字符。输出缓冲区中的小数点右面的位数由 nnn 域指定。nnn 域的有效范围是 0~5。将小数点右面的位数指定为 0 会使值不带小数点。当 nnn 值大于 5 时，输出显示为 12 个 ASCII 空格字符的字符串。c 位指定是使用逗号 (c = 1) 还是使用小数点 (c = 0) 作为整数和小数之间的分隔符。格式最上方的 4 个位必须为零。

下表显示了使用小数点 (c = 0) 和小数点右面 4 位数 (nnn = 100) 格式的值范例。位于 OUT 位置的值是字符串长度。

IN	OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT
		+1	+2	+3	+4	+5	+6	+7	+8	+9	+10	+11	+12
12	12							0	•	0	0	1	2
1234567	12					1	2	3	•	4	5	6	7

设置 ENO = 0 的错误条件是：0006（间接寻址）；0091（操作数范围）；非法格式（nnn>5）。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	长整数	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*LD、*AC	输入端数据
<FMT>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	精度格式
<OUT>	字符串	VB、*VD、LB、*AC、*LD	输出端数据

### ❖ 执行范例

● 范例 105 ● 使用长整数转换为字符串指令。

梯 形 图	语 句 表 指 令
	<pre>//当 I0.0 打开，长整数 1234567 被转换成字符串“123.4567”放入以 VB1 开始的 13 个连续字节中 LD    I0.0 RTS   1234567 VB1 4</pre>

### ❖ 相关指令

STR

## 3.8

## ROUND 指令：取整为长整数

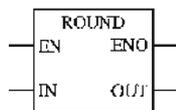
❖ 指令语法 ROUND <IN（实数）> <OUT（长整数）>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<实数>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC
<长整数>	长整数	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

将实数 (IN) 转换成长整数值 (OUT)。如果小数部分大于或等于 0.5, 则进位为整数。使 ENO = 0 的错误条件是: SM1.1 (溢出); SM4.3 (运行时间); 0006 (间接寻址)。

## ❖ 梯形图参数说明

参数的具体说明如下:

参 数	数 据 类 型	内 存 区 域	说 明
< IN >	实数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC	输入端数据
< OUT >	长整数	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*LD、*AC	输出端数据

## ❖ 执行范例

● 范例 106 ● 使用取整指令。

梯 形 图	语 句 表 指 令
	<pre>//当 I0.0 打开时, 实数 256.54 被 取整为 257 而放入 VD12 中 LD      I0.0 ROUND  256.54 VD12</pre>

## 3.9

## TRUNC 指令: 截断为长整数

## ❖ 指令语法

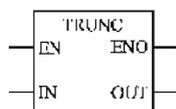
TRUNC <IN (实数)> <OUT (长整数)>

## ❖ 指令参数说明

参数的具体说明如下:

地 址	数 据 类 型	存 储 区
<实数>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC
<长整数>	长整数	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

将 32 位实数 (IN) 转换成 32 位双整数，并将结果的整数部分置入 OUT 指定的变量中。只有实数的整数部分被转换，小数部分被丢弃。如果要转换的值为无效实数或值过大，无法在输出中表示，则设置溢出位，输出不受影响。使 ENO = 0 的错误条件是：SM1.1 (溢出)；SM4.3 (运行时间)；0006 (间接寻址)。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
< IN >	实数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC	输入端数据
< OUT >	长整数	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*LD、*AC	输出端数据

## ❖ 执行范例

## ● 范例 107 ● 使用截断指令。

梯 形 图	语 句 表 指 令
	<pre>//当 I0.0 打开时，实数 256.54 被截断为 256 而放入 VD12 中 LD      I0.0 TRUNC  256.54 VD12</pre>

## 3.10 RTS 指令：实数转换至字符串

## ❖ 指令语法

RTS <IN (实数)> <OUT (字符串)> <FMT (格式)>

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<实数>	实数	VD、ID、QD、MD、SD、SMD、LD、常数、AC、*VD、*LD、*AC

续表

地 址	数 据 类 型	存 储 区
<FMT (格式)>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<字符串>	字符串	VB、*VD、LB、*AC、*LD



### 梯形图功能说明

实数转换为字符串指令将实数值 IN 转换为 ASCII 字符串。格式 (FMT) 指定小数点右面的转换精度，无论小数点是显示为逗号还是点号，也无论输出字符串的长度是多少。转换结果放置在以 OUT 开始的字符串中。结果字符串长度在格式中指定，可以是 3~15 个字符。

RTS 的格式 (FMT) 操作数定义：

MSB							LSB
7	6	5	4	3	2	1	0
s	s	s	s	c	n	n	n

ssss 表示输出字符串长度，c 表示逗号 (1) 或小数点 (0)，nnn 表述小数点右面的位数。

输出字符串的长度由 ssss 域指定。0、1 或 2 个字节无效。输出缓冲区中的小数点右面的位数由 nnn 域指定。nnn 域的有效范围是 0~5。将小数点右面的位数指定为 0 会使值不带小数点。当 nnn 值大于 5 时或当指定的输出字符串长度太小无法存储转换的值时，输出字符串用 ASCII 空格字符填充。c 位指定是使用逗号 (c = 1) 还是使用小数点 (c = 0) 作为整数和小数之间的分隔符。

下表显示了几个使用小数点 (c = 0) 和小数点右面有一位数 (nnn = 001) 以及输出字符串长度为 6 个字符 (ssss = 0110) 格式的值范例。位于 OUT 位置的值是字符串长度。

	OUT	OUT+1	OUT+2	OUT+3	OUT+4	OUT+5	OUT+6
IN = 1234.5	6	1	2	3	4	.	5

续表

	OUT	OUT+1	OUT+2	OUT+3	OUT+4	OUT+5	OUT+6
IN = -0.0004	6				0	.	0
IN = -3.67526	6			-	3	.	7
IN = 1.95	6				2	.	0

S7-200 使用的实数格式最多可支持 7 个高位数字。尝试显示 7 个以上高位数字会产生取整错误。设置 ENO = 0 的错误条件是：0006（间接寻址）；0091（操作数范围）；非法格式（ $nnn > 5$  或  $ssss < 3$  或  $ssss <$ 要求的字符数）。

### ◆ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	实数	VD、ID、QD、MD、SD、SMD、LD、常数、AC、*VD、*LD、*AC	输入端数据
<FMT>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	精度格式
<OUT>	字符串	VB、*VD、LB、*AC、*LD	输出端数据

### ◆ 执行范例

◎ 范例 108 ◎ 使用实数转换为字符串指令。

梯 形 图	语 句 表 指 令
	<p>//当 I0.0 打开时，长整数 1234.5 被转换成字符串“1234.5”，并放入以 VB1 开始的 6 个连续字节中</p> <pre>LD    I0.0 RTS   1234.5 VB1 49</pre>

### ◆ 相关指令

STR

## 3.11 BCDI 指令：BCD 码转换为整数

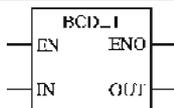
◆ 指令语法 BCDI <IN (字)> <OUT (字)>

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<字>	字	VW、IW、QW、MW、SW、SMW、LW、T、C、AIW、AC、常数、*VD、*AC、*LD
<字>	字	W、IW、QW、MW、SW、SMW、LW、T、C、AC、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

将输入的 BCD 码 (IN) 转换成整数 (OUT)，即将结果送入 OUT。输入 IN 的范围是 0~9999。使 ENO = 0 的错误条件是：0006 (间接地址)；SM1.6 (无效 BCD 数值)。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
< IN >	字	VW、IW、QW、MW、SW、SMW、LW、T、C、AIW、AC、常数、*VD、*AC、*LD	输入端数据
< OUT >	字	W、IW、QW、MW、SW、SMW、LW、T、C、AC、*VD、*LD、*AC	输出端数据

## ❖ 执行范例

● 范例 109 ● 使用 BCD 码转换至整数指令。

梯 形 图	语 句 表 指 令
	<pre>//当 I0.0 打开时，AC0 中的 BCD //值被转换成整数载入 AC0 LD    I0.0 BCDI  AC0</pre>

## ❖ 相关指令

IBCD

## 3.12 IBCD 指令：整数转换为 BCD 码

## ❖ 指令语法

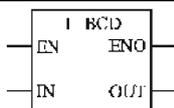
IBCD <IN (字)> <OUT (字)>

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<字>	字	VW、IW、QW、MW、SW、SMW、LW、T、C、AIW、AC、常数、*VD、*AC、*LD
<字>	字	W、IW、QW、MW、SW、SMW、LW、T、C、AC、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

将输入整数值 IN 转换成二进制编码的十进制数，并将结果载入 OUT 指定的变量中。IN 的有效范围是 0 至 9999 BCD。使 ENO = 0 的错误条件是：0006（间接地址）；SM1.6（无效 BCD 数值）。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
< IN >	字	VW、IW、QW、MW、SW、SMW、LW、T、C、AIW、AC、常数、*VD、*AC、*LD	输入端数据
< OUT >	字	W、IW、QW、MW、SW、SMW、LW、T、C、AC、*VD、*LD、*AC	输出端数据

## ❖ 执行范例

● 范例 110 ● 使用整数转换至 BCD 指令。

梯 形 图	语 句 表 指 令
	<pre>//当 I0.0 打开时，AC0 中的整数值被转换成 BCD 载入 AC0 LD    I0.0 IBCD  AC0</pre>

## ❖ 相关指令

BCDI

## 3.13 ITA 指令：整数转换至 ASCII 码

## ❖ 指令语法

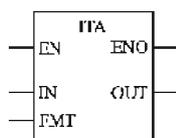
ITA <IN (整数)> <OUT (字节)> <FMT (格式)>

## ◆ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<整数>	整数	VW、IW、QW、MW、SW、SMW、LW、AIW、T、C、AC、常数、*VD、*LD、*AC
<FMT (格式)>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<字节>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC

## ◆ 梯形图



## ◆ 梯形图功能说明

将整数值(IN)转换成 ASCII 码。格式 FMT 指定小数点右侧的转换精确度，以及将小数点显示为逗号还是点号。转换结果置于从 OUT 开始的 8 个连续字节中。ASCII 字符数组总是 8 个字符。

ITA 的格式 (FMT) 操作数定义：

MSB							LSB
7	6	5	4	3	2	1	0
0	0	0	0	c	n	n	n

输出缓冲区的尺寸总是 8 个字节。输出缓冲区中小数点右侧的位数由 nnn 域指定。nnn 域的有效范围是 0~5。指定小数点右侧的数字为 0 会使数值不带小数点。对于大于 5 的 nnn 数值，用 ASCII 空格填充输出缓冲区。c 位指定是使用逗号 (c=1) 还是使用小数点 (c=0) 作为整数和小数之间的分隔符。上方 4 个位必须为 0。

根据下列规则对输出缓冲区进行格式化。

- (1) 正值写入输出缓冲区，不带符号。
- (2) 负值写入输出缓冲区，带起始负号 (-)。
- (3) 小数点左侧的起首零 (与小数点相邻的数字除外) 被压缩。
- (4) 输出缓冲区中的数值右对齐。

下表显示了几个小数点右面有 3 位数格式的数值范例。

	OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT
		+1	+2	+3	+4	+5	+6	+7
IN = 12				0	.	0	1	2
IN = -123			-	0	.	1	2	3
IN = 1234				1	.	2	3	4
IN = -12345		-	1	2	.	3	4	5

设置 ENO = 0 的错误条件是：0006（间接地址）；FMT 位 > 0（用于 FMT 值最高的 4 位）；nnn > 5。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	整数	VW、IW、QW、MW、SW、SMW、LW、AIW、T、C、AC、常数、*VD、*LD、*AC	输入端数据
<FMT>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	精度格式
<OUT >	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC	输出端数据

### ❖ 执行范例

⊙ 范例 111 ⊙ 使用整数转换为 ASCII 指令。

梯 形 图	语 句 表 指 令
	<pre>//将位于 VW2 位址的整数值转换为 8 //个 ASCII 字符，从 VB10 位置开始， //使用 16#0B 格式，用逗号代表小数点， //随后有 3 位数 LD  I2.3 ITA  VW2 VB10 16#0B</pre>

指令执行效果

	' '	' '	'1'	'2'	','	'3'	'4'	'5'	
12345	ITA	20	20	31	32	2C	33	34	35
VW2	VB10	VB11	...						

## 3.14 DTA 指令：长整数转换至 ASCII 码

### ❖ 指令语法

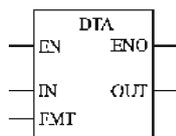
DTA <IN (整数)> <OUT (字节)> <FMT (格式)>

## ◆ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<长整数>	长整数	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*LD、*AC
<FMT (格式)>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<字节>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC

## ◆ 梯形图



## ◆ 梯形图功能说明

将输入端 (IN) 的整数转换成 ASCII 码。格式 FMT 指定小数点右侧的转换精确度。转换结果置于从 OUT 开始的 12 个连续字节中。

DTA 的格式 (FMT) 操作数定义：

MSB							LSB
7	6	5	4	3	2	1	0
0	0	0	0	c	n	n	n

输出缓冲区的尺寸总是 12 个字节。输出缓冲区中小数点右侧的位数由 nnn 域指定。nnn 域的有效范围是 0~5。指定小数点右侧的数字为 0 会使数值不带小数点。对于大于 5 的 nnn 数值，用 ASCII 空格填充输出缓冲区。c 位指定是使用逗号 (c=1) 还是使用小数点 (c=0) 作为整数和小数之间的分隔符。上方 4 个位必须为 0。

下表显示了几个小数点右面有 4 位数格式的数值范例。

OUT 字符	0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+10	+11
IN = -12						-	0	·	0	0	1	2
IN = 1234567					1	2	3	·	4	5	6	7

设置 ENO = 0 的错误条件：0006 (间接地址)；FMT 位>0 (用于 FMT 值最高的 4 位)；nnn>5。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	长整数	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*LD、*AC	输入端数据
<FMT>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	精度格式
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC	输出端数据

### ❖ 执行范例

⊙ 范例 112 ⊙ 使用长整数转换为 ASCII 指令。

梯 形 图	语 句 表 指 令
	<pre>//将位于 VD2 位址的整数值 1234567 //转换为 12 个 ASCII 字符“123.4567”， //从 VB10 位置开始，使用 16#04 格式， //小数点后有 4 位数 LD  I2.3 DTA  VD2 VB10 16#04</pre>

## 3.15 RTA 指令：实数转换至 ASCII 码

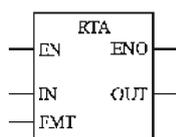
❖ 指令语法 RTA <IN (实数)> <OUT (字节)> <FMT (格式)>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<实数>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、常数、*VD、*LD、*AC
<FMT (格式)>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<字节>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC

❖ 梯形图



### ◆ 梯形图功能说明

将实数值 (IN) 转换成 ASCII 码。格式 FMT 指定小数点右侧的转换精确度, 以及将小数点表示为逗号或点号及输出缓冲区尺寸。转换结果置于从 OUT 开始的输出缓冲区中。结果 ASCII 字符的数目 (或长度) 相当于输出缓冲区的尺寸, 指定的尺寸范围为 3~15 个字符。

RTA 的格式 (FMT) 操作数定义:

MSB							LSB
7	6	5	4	3	2	1	0
s	s	s	s	c	n	n	n

输出缓冲区的大小由 ssss 域指定。0、1 或 2 个字节无效。输出缓冲区中小数点右面的位数由 nnn 域指定。nnn 域的有效范围是 0~5。将小数点右面的位数指定为 0 会使值不带小数点。当 nnn 值大于 5 时或当指定的输出字符串长度太小无法存储转换的值时, 输出缓冲区用 ASCII 空格字符填充。c 位指定是使用逗号 (c = 1) 还是使用小数点 (c = 0) 作为整数和小数之间的分隔符。

根据下列规则对输出缓冲区进行格式化。

- (1) 正值写入输出缓冲区, 不带符号。
- (2) 负值写入输出缓冲区, 带起始负号 (-)。
- (3) 小数点左侧的起首零 (与小数点相邻的数字除外) 被压缩。
- (4) 小数点右侧的数值进位, 使之符合小数点右侧指定的位数。
- (5) 输出缓冲区的尺寸必须最小比小数点右侧数位多 3 个字节。
- (6) 输出缓冲区中的数值右对齐。

下表显示几个小数点右面有一位数 (nnn = 001) 和六个字节缓冲区大小 (ssss = 0110) 格式的值范例。

	OUT	OUT	OUT	OUT	OUT	OUT
		+1	+2	+3	+4	+5
IN = 1234.5	1	2	3	4	.	5
IN = -0.0004				0	.	0
IN = -3.67526			-	3	.	7
IN = 1.95				2	.	0

设置 ENO = 0 的错误条件是：0006（间接地址）；nnn>5；ssss<3；ssss<OUT 中的字符数。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、常数、*VD、*LD、*AC	输入端数据
<FMT>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	精度格式
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC	输出端数据

### ❖ 执行范例

● 范例 113 ● 使用实数转换为 ASCII 指令。

梯形图	语句表指令
	<pre>//将位于 VD2 位置的实数值转换为 10 //个 ASCII 字符，从 VB10 开始，使用 //16#0B 格式，用句号代表小数点，随 //后有 3 位数 LD I2.3 RTA VD2 VB10 16#A3</pre>

指令执行效果

		'	'	'	'1'	'2'	'3'	'.'	'4'	'5'	'0'
12345	RIA	20	20	20	31	32	33	2E	34	35	30
VD2		VB10	VB11	...							

## 3.16

## ATH 指令：ASCII 码转换至十六进制数字

❖ 指令语法 RTA <IN (字节)> <OUT (字节)> <LEN (长度)>

❖ 指令参数说明

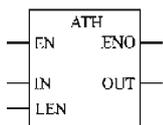
参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<字节>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC
<字节>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC

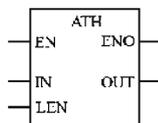
续表

地 址	数 据 类 型	存 储 区
<长度>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD

### ❖ 梯形图



### ❖ 梯形图功能说明



指令把从 IN 字符开始长度为 LEN

的 ASCII 码字符串，转换成从 OUT 开始的十六进制数。ASCII 码字符串的最大长度为 255 个字符。设置 ENO = 0 的错误条件：0006（间接地址）；0091（操作数范围）；SM1.7（非法 ASCII 值，仅限 ASCII 至十六进制数字）。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC	输入端数据
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC	输出端数据
<LEN>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	字符串长度

### ❖ 执行范例

#### ● 范例 114 ● 使用 ASCII 码至十六进制数字指令。

梯 形 图	语 句 表 指 令
	<pre>//将从 VB30 开始的 3 个 ASCII 码字符串，转换成从 VB40 开始的十六进制数 LD I3.2 ATH VB30 VB40 3</pre>

指令执行效果

'3'	'T'	'A'	ATH	3E	Ax	注释：X 表示“半字节”未改动。
VB30				VB40		

## ❖ 相关指令

HTA

# 3.17 HTA 指令：十六进制数字转换至 ASCII 码

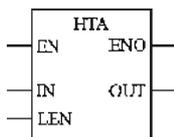
❖ 指令语法 HTA &lt;IN (字节) &gt; &lt;OUT (字节) &gt; &lt;LEN (长度) &gt;

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<字节>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC
<字节>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC
<长度>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD

## ❖ 梯形图



## ❖ 梯形图功能说明

将从输入字节 (IN) 开始的十六进制数字转换成从 OUT 开始的 ASCII 字符。欲转换的十六进制数字位数由长度 (LEN) 指定。可转换的最大十六进制数字位数为 255。设置 ENO = 0 的错误条件是：0006 (间接地址)；0091 (操作数范围)；SM1.7 (非法 ASCII 值，仅限 ASCII 至十六进制数字)。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC	输入端数据
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC	输出端数据
<LEN>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	字符串长度

## ❖ 执行范例

● 范例 115 ● 使用十六进制数字转换至 ASCII 码指令。

梯形图	语句表指令
	<pre>//将从 VB40 的 1 个十六进制数转换成从 VB30 开始的的 ASCII 码字符串 LD I3.2 HTA VB40 VB30 1</pre>
指令执行效果	

## ❖ 相关指令

ATH

## 3.18 STI 指令：字符串转换至整数

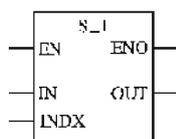
❖ 指令语法 STI<IN (字符串)><INDX (字节)><OUT (整数)>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<字符串>	字符串	VB、常数字符串、LB、*VD、*LD、*AC
<INDX (字节)>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<整数>	整数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

将字符串数值 IN 从偏移量 INDX 位置开始转换为存储在 OUT 中的整数值。INDX 值通常设为 1，从字符串的第一个字符开始转换。可将该值设为其他值，在字符串中的不同点开始转换。当输入字符串包含不属于需要转换数字一部分的文本时，可采用此种方法。

例如，如果输入字符串是“Temperature: 77.8”，可以将 INDX 值设为 13，跳过字符串开始的字“Temperature:”。

每当转换产生一个对于输出值过大的整数值时，则设置溢出错误 (SM1.1)。例如，如果输入字符串产生一个大于 32 767 或小于 -32 768 的值时，STI 指令设置溢出错误。

如果当输入字符串未包含有效值而无法执行转换时，也会设置溢出错误 (SM1.1)。例如，如果输入字符串包含“A123”，转换指令会设置 SM1.1 (溢出)，输出值保持不变。

设置 ENO = 0 的错误条件是：0006 (间接地址)；0091 (操作数范围)；009B (非法指数，指定起始位置值 0 的字符串操作)；SM1.1 (溢出或非法值)。

### 参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字符串	VB、常数字符串、LB、*VD、*LD、*AC	输入端数据
<INDX>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	偏移量
<OUT>	整数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC	输出端数据

### 执行范例

#### 范例 116 使用字符串转换为整数指令。

梯 形 图	语 句 表 指 令
	<pre>//当 I0.0 打开时，从 VB7 开始的字符串被转换为整数载入 VW100 中 LD    I0.0 STI   VB0 7 VW100</pre>

#### 指令执行效果

输入字符串举例

VB0	VB1	VB2	VB3	VB4	VB5	VB6	VB7	VB8	VB9	VB10	VB11
I	l	c	m	p			9	8	.	6	F

执行之后：

VW100 (整数)=98

## ◆ 相关指令

ITS

## 3.19 STD 指令：字符串转换至长整数

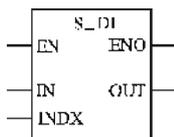
◆ 指令语法 STD <IN (字符串)> <INDX (字节)> <OUT (整数)>

## ◆ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<字符串>	字符串	VB、常数字符串、LB、*VD、*LD、*AC
<INDX (字节)>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<长整数>	长整数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC

## ◆ 梯形图



◆ 梯形图功能说明 将字符串值 IN 从偏移量 INDX 位置开始转换为存储在 OUT 中的双整数值。设置 ENO = 0 的错误条件是：0006（间接地址）；0091（操作数范围）；009B（非法指数——指定起始位置值 0 的字符串操作）；SM1.1（溢出或非法值）。

## ◆ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字符串	VB、常数字符串、LB、*VD、*LD、*AC	输入端数据
<INDX>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	偏移量
<OUT>	长整数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC	输出端数据

## ❖ 执行范例

● 范例 117 ● 使用字符串转换为长整数指令。

梯 形 图	语句表指令
	<pre>//当 I0.0 打开时，从 VB7 开始的字符串 //被转换为长整数载入 VD200 中 LD    I0.0 STD   VB0 7 VD200</pre>

## 指令执行效果

输入字符串举例

VB0	VB1	VB2	VB3	VB4	VB5	VB6	VB7	VB8	VB9	VB10	VB11
l	l	c	m	p			9	8	.	6	F

执行之后：

VD200 (双整数)=98

## ❖ 相关指令

ITS

## 3.20 STR 指令：字符串转换至实数

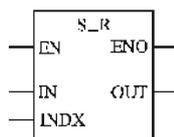
❖ 指令语法 STR <IN (字符串)> <INDX (字节)> <OUT (实数)>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<字符串>	字符串	VB、常数字符串、LB、*VD、*LD、*AC
<INDX (字节)>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<实数>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

将字符串值 IN 转换为存储在 OUT 中的实数

值，从偏移量 INDX 位置开始。设置 ENO = 0 的错误条件是：0006（间接地址）；0091（操作数范围）；009B（非法指数，指定起始位置值 0 的字符串操作）；SM1.1（溢出或非法值）。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字符串	VB、常数字符串、LB、*VD、*LD、*AC	输入端数据
<INDX>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	偏移量
<OUT>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC	输出端数据

### ❖ 执行范例

● 范例 118 ● 使用字符串转换为实数指令。

梯 形 图	语 句 表 指 令
	<p>//当 I0.0 打开时，从 VB7 开始的字符串被转换为实数载入 VD300 中</p> <pre>LD    I0.0 STR   VB0 7 VD200</pre>

指令执行效果

输入字符串举例

VB0	VB1	VB2	VB3	VB4	VB5	VB6	VB7	VB8	VB9	VB10	VB11
l	l	c	m	p			9	8	.	6	F

执行之后：

VD300（实数）=98.6

### ❖ 相关指令

ITS

## 3.21 DECO 指令：解码

❖ 指令语法 DECO <IN（字节）> <OUT（字）>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<字节>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<字>	字	VW、IW、QW、MW、SW、SMW、LW、T、C、AIW、AC、常数、*VD、*AC、*LD



◆ 梯形图功能说明 根据输入字节（IN）的低 4 位（半个字节）所表示的位号，置输出字（OUT）的相应位为 1，其他位置 0。设置 ENO = 0 的错误条件是：0006（间接寻址）。

◆ 梯形图参数说明

参数的具体说明如下：

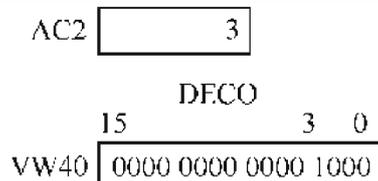
参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	输入端数据
<OUT>	字	VW、IW、QW、MW、SW、SMW、LW、T、C、AIW、AC、常数、*VD、*AC、*LD	输出端数据

◆ 执行范例

◎ 范例 119 ◎ 使用解码指令。

梯 形 图	语 句 表 指 令
	<pre>//AC2 包含错误码 3， DECO 指令按错误码将 VW40 中的第 3 位置位 LD    I3.1 DECO  AC2 VW40</pre>

应 用



◆ 相关指令

ENCO

## 3.22 ENCO 指令：编码

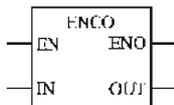
◆ 指令语法 ENCO <IN (字)> <OUT (字节)>

◆ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<字>	字	VW、IW、QW、MW、SW、SMW、LW、T、C、AIW、AC、常数、*VD、*AC、*LD
<字节>	字节	VB、IB、QB、MB、SMB、LB、SB、AC、*VD、*LD、*AC

◆ 梯形图



◆ 梯形图功能说明

将输入字 (IN) 的最低有效位 (值为 1) 的位号写入输出字节 (OUT) 的低 4 位 (半个字节)。设置 ENO = 0 的错误条件是：0006 (间接寻址)。

◆ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字	VW、IW、QW、MW、SW、SMW、LW、T、C、AIW、AC、常数、*VD、*AC、*LD	输入端数据
<OUT>	字节	VB、IB、QB、MB、SMB、LB、SB、AC、*VD、*LD、*AC	输出端数据

◆ 执行范例

● 范例 120 ● 使用编码指令。

梯 形 图	语 句 表 指 令
	<pre>//AC2 包含错误位 ENCO 指令把错误位转 换成存于 VB40 中的错 误码 LD    I3.1 ENCO  AC2 VB40</pre>

续表

梯形图	语句表指令
应 用	
<div style="display: flex; justify-content: space-around; width: 100%;"> <span>15</span> <span>9</span> <span>0</span> </div>	
AC2	0000 0010 0000 1000
ENCO	
VB40	9

## ❖ 相关指令

DECO

## 3.23 SEG 指令：七段显示转换

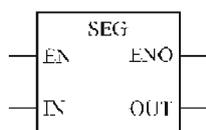
❖ 指令语法 SEG &lt;IN (字节)&gt; &lt;OUT (字节)&gt;

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<字节>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<字节>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC

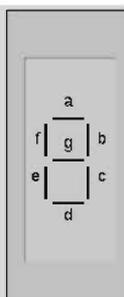
## ❖ 梯形图



## ❖ 梯形图功能说明

段 (SEG) 指令允许生成照明七段显示段的位格式。照明的段代表输入字节最低数位中的字符。下图显示“段”指令使用的七段显示编码。

(进) LSD	段显示	(OUT) -gfe dcba	(进) LSD	段显示	(OUT) -gfe dcba
0	0	0011 1111	8	0	0111 1111
1	1	0000 0110	9	9	0110 0111
2	2	0101 1011	A	A	0111 0111
3	3	0100 1111	B	b	0111 1100
4	4	0110 0110	C	c	0011 1001
5	5	0110 1101	D	d	0101 1110
6	6	0111 1101	E	e	0111 1001
7	7	0000 0111	F	F	0111 0001



指令产生点亮七段码显示器的位模式段码值（OUT）。它是根据输入字节（IN）的低位的有效数字值产生相应点亮段码。设置 ENO = 0 的错误条件是：0006（间接寻址）；SM4.3（运行时间）。

### ◆ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、 常数、*AC、*VD、*LD	输入端数据
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、 *LD、*AC	输出端数据

### ◆ 执行范例

● 范例 121 ● 使用段指令。

梯 形 图	语 句 表 指 令
	<pre>LD  I3.3 SEG  VB48 AC1</pre>

应 用

VB48 05

SEG  
AC1 6D (显示字符)



## 第4章 计数器指令

在用户 CPU 的存储器中，有为计数器保留的存储区。此存储区为每个计数器地址保留一个 16 位字。计数器指令是仅有的可访问计数器存储区的函数，计数值的范围为 0~999。

### 4.1 CTU 指令：向上计数

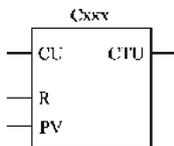
❖ 指令语法 CTU <Cxxx(字节)> <PV(字节)>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<Cxxx>	字	常数 (C0-C255)
<PV>	整数	VW、IW、QW、MW、SMW、LW、AIW、AC、T、C、常数、*VD、*AC、*LD、SW

❖ 梯形图



❖ 梯形图功能说明

每次向上计数输入 CU 从关闭向打开转换时，增计数指令从当前值向上计数。当前值(Cxxx)大于或等于预设值(PV)时，计数器位(Cxxx)打开。复原(R)输入打开或执行“复原”指令时，计数器被复原。达到最大值(32 767)时，计数器停止计数。计数器范围

Cxxx 为 C0 至 C255。CTU 复原输入是堆栈顶值，向上计数输入是装载在第二个堆栈位置的值。

### ❖ 梯形图参数说明

参数的具体说明如下：

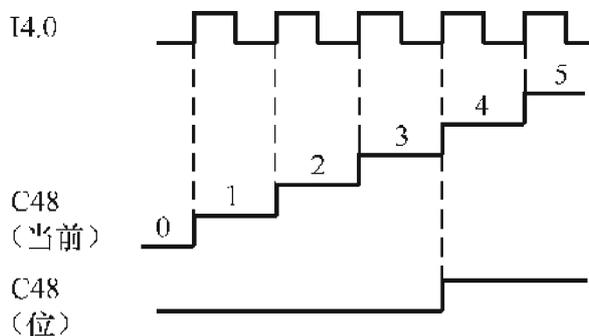
参 数	数 据 类 型	内 存 区 域	说 明
<Cxxx>	字	常数 (C0-C255)	当前值
<PV>	整数	VW、IW、QW、MW、SMW、LW、AIW、AC、T、C、常数、*VD、*AC、*LD、SW	预设值
CU	布尔	使能位	脉冲输入
R	布尔	使能位	复位

### ❖ 执行范例

● 范例 122 ● 使用向上计数指令。

梯 形 图	语 句 表 指 令
	<p>//当计数器 C48 的值达到预设值 4 时，C48 触电置位，Q0.0 被使能</p> <pre>LD    I4.0 CTU   C48 4 LD    C48 =     Q0.0</pre>

时 序 图



### ❖ 相关指令

CTD

## 4.2 CTD 指令：向下计数

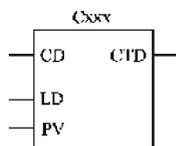
❖ 指令语法 CTD <Cxxx(字节)> <PV(字节)>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<Cxxx>	字	常数 (C0-C255)
<PV>	整数	VW、IW、QW、MW、SMW、LW、AIW、AC、T、C、常数、*VD、*AC、*LD、SW

❖ 梯形图



❖ 梯形图功能说明

每次向下计数输入 CD 从关闭向打开转换时，向下计数（CTD）指令从当前值向下计数。当前值 Cxxx 等于 0 时，计数器位(Cxxx)打开。载入输入(LD)打开时，计数器复原计数器位(Cxxx)并用预设值（PV）载入当前值。达到零时，向下计数器停止计数，计数器位 Cxxx 打开。计数器范围 Cxxx 为 C0 至 C255。CTD 载入输入是堆栈顶值，而向下计数输入是装载在第二个堆栈位置的数值。

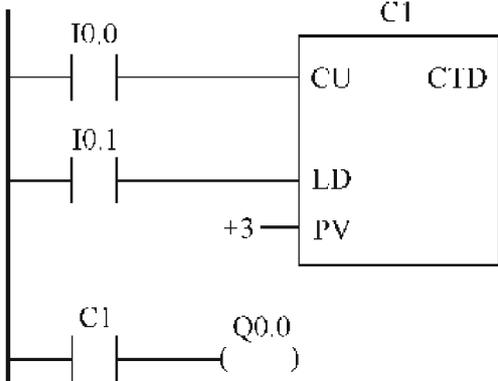
❖ 梯形图参数说明

参数的具体说明如下：

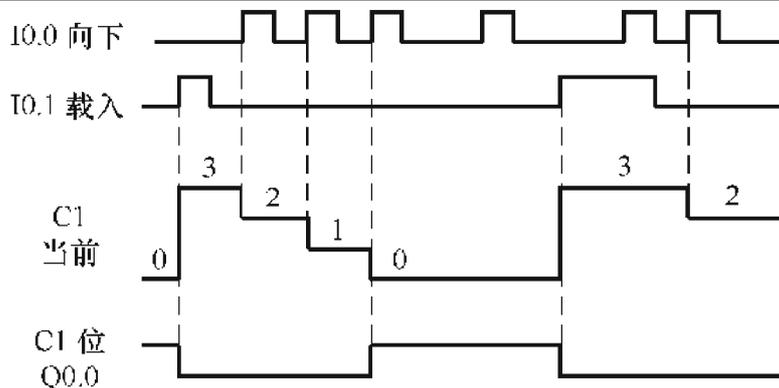
参 数	数 据 类 型	内 存 区 域	说 明
<Cxxx>	字	常数 (C0-C255)	当前值
<PV>	整数	VW、IW、QW、MW、SMW、LW、AIW、AC、T、C、常数、*VD、*AC、*LD、SW	预设值
CD	布尔	使能位	脉冲输入
LD	布尔	使能位	复位

❖ 执行范例

### 范例 123 使用向下计数指令。

梯形图	语句表指令
	<pre>// 向下计数计数器 C1 当前值从 3 计数至 0，其中 I0.1 关闭，I0.0 的“关闭-打开”动作递减 C1 当前值，I0.1 “打开”动作载入向下计数预设值 3 LD I0.0 LD I0.1 CTD C1 +3 // 当计数器 C1 当前值 = 0 时，C1 位“打开” LD C1 = Q0.0</pre>

时序图



#### ◆ 相关指令

CTU

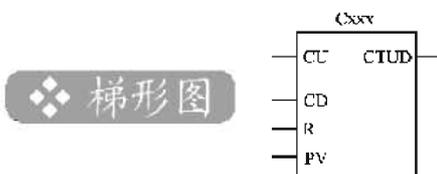
## 4.3 CTUD 指令：双向计数

◆ 指令语法 CTUD <Cxxx(字节)> <PV(字节)>

◆ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<Cxxx>	字	常数 (C0-C255)
<PV>	整数	VW、IW、QW、MW、SMW、LW、AIW、AC、T、C、常数、*VD、*AC、*LD、SW



◆ 梯形图功能说明

每次向上计数输入 CU 从关闭向打开转换时，向上/向下计时（CTUD）指令向上计数，每次向下计数输入 CD 从关闭向打开转换时，向下计数。计数器的当前值 Cxxx 保持当前计数。每次执行计数器指令时，预设值 PV 与当前值进行比较。达到最大值（32 767），位于向上计数输入位置的下一个上升沿使当前值返转为最小值（-32 768）。在达到最小值（-32 768）时，位于向下计数输入位置的下一个上升沿使当前计数返转为最大值（32 767）。当当前值 Cxxx 大于或等于预设值 PV 时，计数器位 Cxxx 打开。否则，计数器位关闭。当“复原”（R）输入打开或执行“复原”指令时，计数器被复原。达到 PV 时，CTUD 计数器停止计数。计数器范围 Cxxx 为 C0 至 C255 在 STL 中，CTUD 复原输入是堆栈顶值，向下计数输入是装载在第二个堆栈位置的值，向上计数输入是装载在第三个堆栈位置的值。

◆ 梯形图参数说明

参数的具体说明如下：

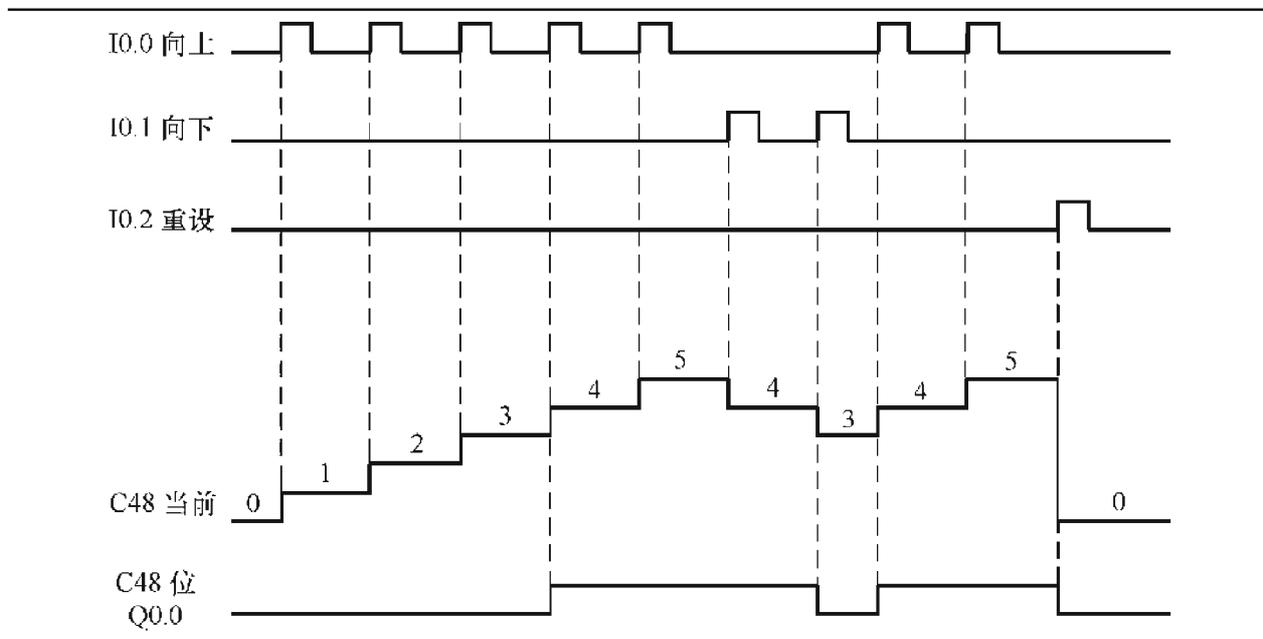
参 数	数 据 类 型	内 存 区 域	说 明
<Cxxx>	字	常数 (C0-C255)	当前值
<PV>	整数	VW、IW、QW、MW、SMW、LW、AIW、AC、T、C、常数、*VD、*AC、*LD、SW	预设值
CU	布尔	使能位	脉冲输入
CD	布尔	使能位	脉冲输入
R	布尔	使能位	复位

◆ 执行范例

- ◎ 范例 124 ◎ 使用双向计数指令。

梯形图	语句表指令
	<pre>// I0.0 向上计数，I0.1 向下计数，I0.2 将当前值复原为 0 LD I0.0 LD I0.1 LD I0.2 CTUD C48 +4  // 当前值 ≥4 时，向上 / 向下计数，计数器 C48 打开 C48 位 LD C48 = Q0.0</pre>

时序图



## ❖ 相关指令

CTU, CTD

## 4.4 HDEF 指令：定义高速计数器

## ❖ 指令语法

HDEF &lt;HSC(计数器编号)&gt; &lt;MODE(工作模式)&gt;

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<HSC>	字节	常数 (0、1、2、3、4 或 5)
<MODE>	字节	常数 (0、1、2、3、4、5、6、7、8、9、10 或 11)



#### ◆ 梯形图功能说明

选择特定的高速计数器（HSCx）的操作模式。模式选择定义高速计数器的时钟、方向、起始和复原功能。可以为每台高速计数器使用一条“高速计数器定义”指令。高速计数器累计 CPU 扫描速率不能控制的高速事件，可以配置最多 12 种不同的操作模式。

一般来说，高速计数器被用作驱动鼓形计时器设备，该设备有一个安装了增量轴式编码器的轴以恒定的速度转动。轴式编码器每圈提供一个确定的计数值和一个复位脉冲。来自轴式编码器的时钟和复位脉冲作为高速计数器的输入。高速计数器装入一组预置值中的第一个值，当前计数值小于当前预置值时，希望的输出有效。计数器设置成在当前值等于预置值和有复位时产生中断。随着每次当前计数值等于预置值的中断事件的出现，一个新的预置值被装入并重新设置下一个输出状态。当出现复位中断事件时，设置第一个预置值和第一个输出状态，这个循环又重新开始。中断事件产生的速率远低于高速计数器的计数速率，用高速计数器可实现精确控制，而与 PLC 整个扫描周期的关系不大。采用中断的方法允许在简单的状态控制中用独立的中断程序装入一个新的预置值，这样使得程序简单直接，并容易读懂。

#### ◆ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<HSC>	字节	常数 (0、1、2、3、4 或 5)	计数器编号
<MODE>	字节	常数 (0、1、2、3、4、5、6、7、8、9、10 或 11)	工作模式

## ❖ 执行范例

## ● 范例 125 ● 定义高速计数器指令。

梯形图	语句表指令
	<pre>LD    SM0.1 // 首次扫描时， MOVB 16#F8 SMB47 // 选择现用水平高的起始和复原输入，选择 4x 模式 HDEF 1 11 // 将 HSC1 配置为正交模式，具有复原和起始输入功能</pre>

## ❖ 相关指令

HSC

## 4.5 HSC 指令：高速计数器

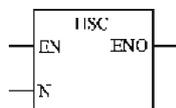
❖ 指令语法 HSC &lt;N(计数器编号)&gt;

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<N>	字节	常数 (0、1、2、3、4 或 5)

❖ 梯形图



❖ 梯形图功能说明

根据 HSC 特殊内存位的状态配置和控制高速计数器。参数 N 指定高速计数器的号码。高速计数器最多可配置为 12 种不同的操作模式。每台计数器在功能受支持的位置有专用时钟、方向控制、复原和起始输入。对于双相计数器，两个时钟均可按最高速度运行。在正交模式中，您可以选择一倍（1x）或四倍（4x）的最高计数速率。所有的计数器按最高速率运行，而不会相互干扰。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<N>	字 节	常数 (0、1、2、3、4 或 5)	计数器编号

### ❖ 执行范例

⊙ 范例 126 ⊙ 使用高速计数器指令。

梯 形 图	语 句 表 指 令
	<pre> // 主程序，首次扫描时，调用 SBR_0 LD    SM0.1 CALL  SBR_0 // 子程序 0 开始，配置 HSC1 LD    SM0.1 // 首次扫描时 MOVB  16#F8 SMB47 // 配置 HSC1: // - 启用计数器 // - 写入新当前值 // - 写入新预设值 // - 将初始方向设为向上计数 // - 选择现用水平高的起始和复 原输入 // - 选择 4x 模式 HDEF 1 11 // 将 HSC1 配置为正交 模式，具有复原和起始输入功能 MOVD +0 SMD48 //清除 HSC1 的 当前值 MOVD +50 SMD52 // 将 HSC1 预 设值设为 50 ATCH INT_0 13 // HSC1 当前值 = 预设值（事件 13），附加在中断例 行程序 INT_0 上 ENI // 全局中断启用 HSC 1 // 程序 HSC1 // 中断 0 开始 LD    SM0.0 MOVD +0 SMD48 // 清除 HSC1 的 当前值 </pre>

续表

梯形图	语句表指令
<p>The diagram shows a single rung of a ladder logic network. It starts with a normally open contact labeled SM0.0. This contact is connected to the EN (Enable) input of three instructions stacked vertically. The first instruction is MOV_DW, with its IN input set to 0 and its OUT output set to SMD48. The second instruction is MOV_B, with its IN input set to 16#C0 and its OUT output set to SMD47. The third instruction is HSC, with its N input set to 1 and its ENO output.</p>	<pre> MOV_B 16#C0 SMB47 // 选择仅写入一个新当前值，使 HSC1 保持启用状态 HSC 1 // 程序 HSC1 </pre>

### ◆ 相关指令

HDEF



## 第 5 章 浮点型数学运算指令

浮点型数学运算指令针对累加器 1 和 2 的内容进行运算，其结果保存在累加器 1 中。对于两个 CPU 的累加器，累加器 2 的内容保持不变。

### 5.1 +R 指令：实数加

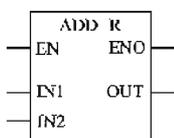
❖ 指令语法 +R <IN1(被加数)> <OUT(加法结果)>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、常数、*VD、*LD、*AC
<OUT>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC

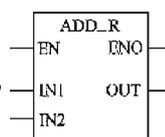
❖ 梯形图



❖ 梯形图功能说明

指令将两个 32 位实数相加，并产生一个 32

位实数结果(OUT)。在 STL 中： $IN1 + IN2 = OUT$ 。



在启用(EN)

输入端通过一个逻辑“1”来激活 ADD\_R (实数加)。IN1 和 IN2 相加，结

果通过 OUT 查看。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、常数、*VD、*LD、*AC	被加数
<IN2>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、常数、*VD、*LD、*AC	加数
<OUT>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC	加法结果

### ❖ 执行范例

● 范例 127 ● 使用实数加法指令。

梯 形 图	语 句 表 指 令												
	<pre>LD I0.0 +R AC1 AC0</pre>												
应用													
	<table border="1"> <thead> <tr> <th></th> <th>IN1</th> <th>IN2</th> <th>OUT</th> </tr> </thead> <tbody> <tr> <td>加数据</td> <td>4000.0</td> <td>+ 6000.0</td> <td>10000.0</td> </tr> <tr> <td>数据地址</td> <td>AC1</td> <td>AC0</td> <td>AC0</td> </tr> </tbody> </table>		IN1	IN2	OUT	加数据	4000.0	+ 6000.0	10000.0	数据地址	AC1	AC0	AC0
	IN1	IN2	OUT										
加数据	4000.0	+ 6000.0	10000.0										
数据地址	AC1	AC0	AC0										

### ❖ 相关指令

-R

## 5.2 -R 指令：实数减

❖ 指令语法 +R <IN1(减数)> <OUT(减法结果)>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、常数、*VD、*LD、*AC

续表

地 址	数 据 类 型	存 储 区
<OUT>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC



❖ 梯形图功能说明 指令将两个 32 位实数相减，并产生一个 32

位实数结果(OUT)。在 STL 中： $OUT - IN1 = OUT$ 。

在启用(EN)

输入端通过一个逻辑“1”来激活 SUB\_R（实数减）。IN1 减去 IN2，结果可通过 OUT 查看。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、常数、*VD、*LD、*AC	被减数
<IN2>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、常数、*VD、*LD、*AC	减数
<OUT>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC	减法结果

❖ 执行范例

● 范例 128 ● 使用实数减法指令。

梯 形 图	语 句 表 指 令
	<pre>LD I0.0 -R AC1 AC0</pre>
应用	

续表

梯形图		语句表指令	
	IN1	IN2	OUT
加数据	6000.0	- 4000.0	2000.0
数据地址	AC1	AC0	AC0

## ❖ 相关指令

+R

## 5.3 \*R 指令：实数乘

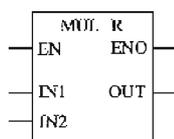
❖ 指令语法 \*R &lt;IN1(被乘数)&gt; &lt;OUT(乘法结果)&gt;

❖ 指令参数说明

参数的具体说明如下：

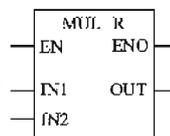
地 址	数 据 类 型	存 储 区
<IN1>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、常数、*VD、*LD、*AC
<OUT>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

指令将两个 32 位实数相乘，并产生一个 32

位实数结果 (OUT)。在 STL 中： $N1 * OUT = OUT$ 。

在启用 (EN)

输入端通过一个逻辑“1”来激活 MUL\_R (实数乘)。IN1 和 IN2 相乘，结果通过 OUT 查看。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、 常数、*VD、*LD、*AC	被乘数
<IN2>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、 常数、*VD、*LD、*AC	乘数
<OUT>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、 *VD、*LD、*AC	乘法结果

### ❖ 执行范例

● 范例 129 ● 使用实数乘法指令。

梯 形 图	语句表指令
	<pre>LD I0.0 *R AC1 VD100</pre>

### 应用

	IN1	IN2	OUT	
乘以数据	400.0	*	200.0	80000.0
数据地址	AC1	VD100	VD100	VD100

### ❖ 相关指令

/R

## 5.4 /R 指令：实数除

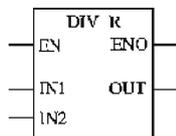
❖ 指令语法 /R <IN1(除数)> <OUT(除法结果)>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、常数、 *VD、*LD、*AC
<OUT>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、 *LD、*AC

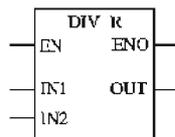
## ❖ 梯形图



## ❖ 梯形图功能说明

指令将两个 32 位实数相除，并产生一个 32

位实数商。在 STL 中： $OUT / IN1 = OUT$ 。



在启用 (EN) 输入

端通过一个逻辑“1”来激活 DIV\_R (实数除)。IN1 除以 IN2，结果可通过 OUT 查看。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、 常数、*VD、*LD、*AC	被除数
<IN2>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、 常数、*VD、*LD、*AC	除数
<OUT>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、 *VD、*LD、*AC	除法结果

## ❖ 执行范例

● 范例 130 ● 使用实数除法指令。

梯 形 图	语 句 表 指 令
	<pre>LD I0.0 /R VD10 VD200</pre>

应用

	IN1	IN2	OUT
除以数据	4000.0	/ 41.0	97.5609
数据地址	VD200	VD10	VD200

## ❖ 相关指令

\*R

## 5.5 Sqrt 指令：求平方根

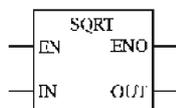
❖ 指令语法 /R <IN> <OUT>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、常数、*VD、*LD、*AC
<OUT>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

指令对 32 位实数 (IN) 取平方根，并产生一个 32 位实数结果 (OUT)，如以下等式所示：

$$OUT = \sqrt{IN}$$

当地址大于“0”时，此指令得出一个正的结果。惟一例外的是：-0 的平方根是-0。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、常数、*VD、*LD、*AC	输入值
<OUT>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC	输出值

❖ 执行范例

● 范例 131 ● 使用求平方根指令。

梯 形 图	语 句 表 指 令
	<pre>//当 I0.0 打开时，VD10 中的实数被做 求平方根运算，结果被载入 VD200 LD    I0.0 Sqrt VD10 VD200</pre>

## 5.6 SIN 指令：求正弦值

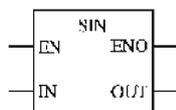
❖ 指令语法 SIN <IN> <OUT>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、常数、*VD、*LD、*AC
<OUT>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

指令对角度值 IN 进行三角正弦运算，并将结果放置在 OUT 中。输入角以弧度为单位。欲将输入角从角度转换成弧度，用角度乘以  $1.745329E-2$ （约等于  $\pi/180$ ）。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、常数、*VD、*LD、*AC	输入值
<OUT>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC	输出值

❖ 执行范例

● 范例 132 ● 使用求正弦指令。

梯 形 图	语 句 表 指 令
	<pre>//当 I0.0 打开时，AC1 中的实数被做求 正弦运算，结果被载入 AC0 LD    I0.0 SIN   AC1 AC0</pre>

## 5.7 COS 指令：求余弦值

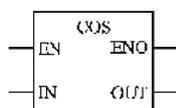
❖ 指令语法 COS <IN> <OUT>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、常数、*VD、*LD、*AC
<OUT>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

指令对角度值 IN 进行三角余弦运算，并将结果放置在 OUT 中。输入角以弧度为单位。欲将输入角从角度转换成弧度，用角度乘以  $1.745329E-2$ （约等于  $\pi/180$ ）。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、常数、*VD、*LD、*AC	输入值
<OUT>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC	输出值

❖ 执行范例

● 范例 133 ● 使用求余弦指令。

梯 形 图	语 句 表 指 令
	<pre>//当 I0.0 打开时，VD10 中的实数被做求余弦运算，结果被载入 VD20 LD    I0.0 COS   VD10 VD20</pre>

## 5.8 TAN 指令：求正切值

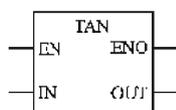
❖ 指令语法 TAN <IN> <OUT>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、常数、*VD、*LD、*AC
<OUT>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

指令对角度值 IN 进行三角正切运算，并将结果放置在 OUT 中。输入角以弧度为单位。欲将输入角从角度转换成弧度，用角度乘以  $1.745329E-2$ （约等于  $\pi/180$ ）。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、常数、*VD、*LD、*AC	输入值
<OUT>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC	输出值

❖ 执行范例

● 范例 134 ● 使用求正切指令。

梯 形 图	语 句 表 指 令
	<pre>//当 I0.0 打开时，VD10 中的实数被做求正切运算，结果被载入 VD200 LD    I0.0 TAN   VD10 VD200</pre>

## 5.9 LN 指令：求自然对数

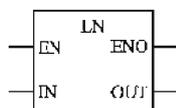
❖ 指令语法 LN <IN> <OUT>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、常数、*VD、*LD、*AC
<OUT>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

指令对 IN 中的数值进行自然对数计算，并将结果置于 OUT 中。欲从自然对数获得以 10 为底数的对数，用自然对数除以 2.302585（约等于 10 的自然对数）。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、常数、*VD、*LD、*AC	输入值
<OUT>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC	输出值

❖ 执行范例

● 范例 135 ● 使用求自然对数指令。

梯 形 图	语 句 表 指 令
	<pre>//当 I0.0 打开时，AC1 中的实数被做求自然对数运算，结果被载入 AC0 LD    I0.0 LN    AC1 AC0</pre>

❖ 相关指令

EXP

## 5.10 EXP 指令：求指数值

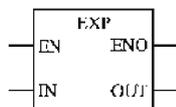
❖ 指令语法 EXP <IN> <OUT>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、常数、*VD、*LD、*AC
<OUT>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

指令进行 e 的 IN 次方指数计算，并将结果置于 OUT 中。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、常数、*VD、*LD、*AC	输入值
<OUT>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC	输出值

❖ 执行范例

● 范例 136 ● 使用求指数指令。

梯 形 图	语 句 表 指 令
	<pre>//当 I0.0 打开时，AC1 中的实数被做 求 e 的 IN 次方指数运算，结果被载入 AC0 LD    I0.0 EXP   AC1 AC0</pre>

❖ 相关指令

LN



## 第 6 章 整型数学运算指令

整型数学运算指令针对累加器 1 和 2 的内容进行运算，其结果保存在累加器 1 中。对于两个 CPU 的累加器，累加器 2 的内容保持不变。

### 6.1 +I 指令：整数加

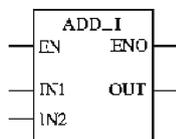
❖ 指令语法 +I <IN1(被加数)> <OUT(加法结果)>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	整 数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC
<OUT>	整 数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC

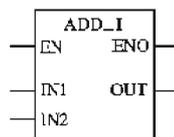
❖ 梯形图



❖ 梯形图功能说明

指令将两个 16 位整数相加，并产生一个 16

位的结果 (OUT)。在 STL 中： $IN1 + IN2 = OUT$ 。



在启用 (EN)

输入端通过一个逻辑“1”来激活 ADD\_I (整数加)。IN1 和 IN2 相加，结

果通过 OUT 查看。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC	被加数
<IN2>	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC	加数
<OUT>	整数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC	加法结果

### ❖ 执行范例

● 范例 137 ● 使用整数加法指令。

梯 形 图	语 句 表 指 令
	<pre>LD I0.0 +I AC1 AC0</pre>

应用

	IN1	IN2	OUT
加数据	40	+	60
数据地址	AC1		AC0

### ❖ 相关指令

-I

## 6.2 +D 指令：长整数加

❖ 指令语法 +D <IN1(被加数)> <OUT(加法结果)>

### ❖ 指令参数说明

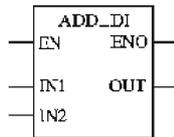
参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	双整数	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*LD、*AC

续表

地 址	数 据 类 型	存 储 区
<OUT>	双整数	VD、ID、QD、MD、SMD、SD、LD、AC、*VD、*LD、*AC

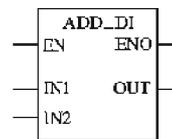
### ❖ 梯形图



### ❖ 梯形图功能说明

指令将两个 32 位整数相加，并产生一个 32

位结果 (OUT)。在 STL 中： $IN1 + IN2 = OUT$ 。



在启用 (EN)

输入端通过逻辑“1”激活 ADD\_DI (长整数加)。IN1 和 IN2 相加，结果通过 OUT 查看。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	双整数	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*LD、*AC	被加数
<IN2>	双整数	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*LD、*AC	加数
<OUT>	双整数	VD、ID、QD、MD、SMD、SD、LD、AC、*VD、*LD、*AC	加法结果

### ❖ 执行范例

● 范例 138 ● 使用长整数加法指令。

梯 形 图	语 句 表 指 令
	<pre>LD I0.0 +DI AC1 AC0</pre>
应用	

续表

梯形图		语句表指令		
	IN1	IN2	OUT	
加数据	100000	+	200000	300000
数据地址	AC1		AC0	AC0

## ❖ 相关指令

-DI

## 6.3 -I 指令：整数减

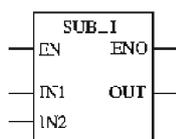
❖ 指令语法  $-I \langle IN1(\text{减数}) \rangle \langle OUT(\text{减法结果}) \rangle$ 

## ❖ 指令参数说明

参数的具体说明如下：

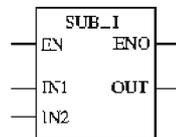
地 址	数 据 类 型	存 储 区
$\langle IN1 \rangle$	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC
$\langle OUT \rangle$	整数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

指令将两个 16 位整数相减，并产生一个 16

位的结果 (OUT)。在 STL 中： $OUT - IN1 = OUT$ 。

在启用 (EN)

输入端通过逻辑“1”激活 SUB\_I (整数减)。IN1 减去 IN2，结果可通过 OUT 查看。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	整 数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC	被减数
<IN2>	整 数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC	减数
<OUT>	整 数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC	减法结果

### ❖ 执行范例

● 范例 139 ● 使用整数减法指令。

梯 形 图	语 句 表 指 令
	<pre>LD I0.0 -I AC1 AC0</pre>

#### 应用

	IN1	IN2	OUT
加数据	60	+	40
数据地址	AC1		AC0

### ❖ 相关指令

+I

## 6.4 -D 指令：长整数减

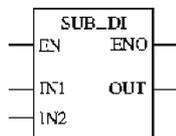
❖ 指令语法 -D <IN1(减数)> <OUT(减法结果)>

❖ 指令参数说明

参数的具体说明如下：

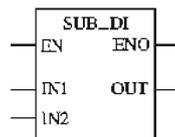
地 址	数 据 类 型	存 储 区
<IN1>	双整数	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*LD、*AC
<OUT>	双整数	VD、ID、QD、MD、SMD、SD、LD、AC、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

指令将两个 32 位整数相加或相减，并产生一个 32 位结果（OUT）。在 STL 中： $OUT - IN1 = OUT$ 。



在启

用（EN）输入端通过逻辑“1”激活 SUB\_DI（长整数减）。IN1 减去 IN2，结果可通过 OUT 查看。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	双整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC	被减数
<IN2>	双整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC	减数
<OUT>	双整数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC	减法结果

## ❖ 执行范例

● 范例 140 ● 使用长整数减法指令。

梯 形 图	语 句 表 指 令
	<pre>LD I0.0 -DI AC1 AC0</pre>

应用

	IN1	IN2	OUT
加数据	500000	- 200000	300000
数据地址	AC1	AC0	AC0

## ❖ 相关指令

+D

## 6.5 MUL 指令：整数与长整数相乘

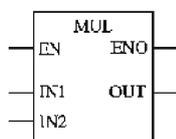
❖ 指令语法 -D <IN1(被乘数)> <OUT(乘法结果)>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	整 数	VW、IW、QW、MW、SW、SMW、T、C、AC、 LW、AIW、常数、*VD、*LD、*AC
<OUT>	双 整 数	VD、ID、QD、MD、SMD、SD、LD、AC、*VD、 *LD、*AC

❖ 梯形图



❖ 梯形图功能说明

指令将两个 16 位整数相乘，得出一个 32 位乘积。32 位 OUT 的低位字（16 位）被用作乘数之一。在 STL 中：

$IN1 * OUT = OUT$ 。在启用 (EN) 输入端通过逻辑“1”激活 MUL

(整数与长整数乘)。IN1 和 IN2 相乘，结果通过 OUT 查看。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	整 数	VW、IW、QW、MW、SW、SMW、T、C、 AC、LW、AIW、常数、*VD、*LD、*AC	被乘数
<IN2>	整 数	VW、IW、QW、MW、SW、SMW、T、C、 AC、LW、AIW、常数、*VD、*LD、*AC	乘数
<OUT>	双 整 数	VD、ID、QD、MD、SMD、SD、LD、AC、 *VD、*LD、*AC	乘法结果

❖ 执行范例

○ 范例 141 ○ 使用整数与长整数乘法指令。

梯形图	语句表指令
	<pre>LD I0.0 MUL AC1 VD100</pre>

## 应用

	IN1	IN2	OUT
乘以数据	400	*	200
数据地址	AC1	VW102	VD100
注释：VD100 包含：VW100 和 VW102			

## ❖ 相关指令

DIV

## 6.6 \*I 指令：整数乘

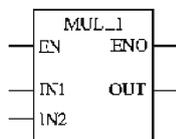
❖ 指令语法 \*I &lt;IN1(被乘数)&gt; &lt;OUT(乘法结果)&gt;

❖ 指令参数说明

参数的具体说明如下：

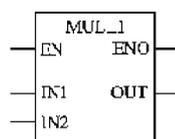
地 址	数 据 类 型	存 储 区
<IN1>	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC
<OUT>	整数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

指令将两个 16 位整数相乘，并产生一个 16

位乘积。在 STL 中： $IN1 * IN2 = OUT$ 。

在启用 (EN) 输入端通

过逻辑“1”激活 MUL\_I (整数乘)。IN1 和 IN2 相乘，结果通过 OUT 查看。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC	被乘数
<IN2>	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC	乘数
<OUT>	整数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC	乘法结果

### ❖ 执行范例

◎ 范例 142 ◎ 使用整数乘法指令。

梯 形 图	语 句 表 指 令														
	<pre>LD I0.0 *I AC1 VW100</pre>														
应用															
	<table border="1"> <thead> <tr> <th></th> <th>IN1</th> <th>IN2</th> <th>OUT</th> </tr> </thead> <tbody> <tr> <td>乘以数据</td> <td>40</td> <td>*</td> <td>20</td> <td>800</td> </tr> <tr> <td>数据地址</td> <td>AC1</td> <td>VW100</td> <td>VW100</td> <td></td> </tr> </tbody> </table>		IN1	IN2	OUT	乘以数据	40	*	20	800	数据地址	AC1	VW100	VW100	
	IN1	IN2	OUT												
乘以数据	40	*	20	800											
数据地址	AC1	VW100	VW100												

### ❖ 相关指令

/I

## 6.7 \*D 指令：长整数乘

❖ 指令语法 \*D <IN1(被乘数)> <OUT(乘法结果)>

❖ 指令参数说明

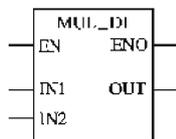
参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	双整数	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*LD、*AC

续表

地 址	数 据 类 型	存 储 区
<OUT>	双整数	VD、ID、QD、MD、SMD、SD、LD、AC、*VD、*LD、*AC

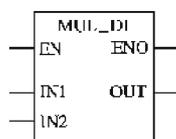
### ❖ 梯形图



### ❖ 梯形图功能说明

指令将两个 32 位整数相乘, 并产生一个 32 位

乘积。在 STL 中:  $IN1 * OUT = OUT$ 。



在启用 (EN) 输入端通过逻辑“1”激活 MUL\_DI (长整数乘)。

IN1 和 IN2 相乘, 结果通过 OUT 查看。

### ❖ 梯形图参数说明

参数的具体说明如下:

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	双整数	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*LD、*AC	被乘数
<IN2>	双整数	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*LD、*AC	乘数
<OUT>	双整数	VD、ID、QD、MD、SMD、SD、LD、AC、*VD、*LD、*AC	乘法结果

### ❖ 执行范例

● 范例 143 ● 使用长整数乘法指令。

梯 形 图	语 句 表 指 令
	<pre>LD I0.0 *D MD0 MD10</pre>

应用

	IN1	IN2	OUT
乘以数据	70000	* 300	21000000
数据地址	MD0	MD10	MD10

## ❖ 相关指令

/I

## 6.8 DIV 指令：整数与长整数相除

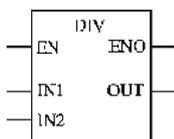
❖ 指令语法  $\text{DIV } \langle \text{IN1(除数)} \rangle \langle \text{OUT(除法结果)} \rangle$ 

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
$\langle \text{IN1} \rangle$	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC
$\langle \text{OUT} \rangle$	双整数	VD、ID、QD、MD、SMD、SD、LD、AC、*VD、*LD、*AC

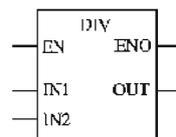
## ❖ 梯形图



## ❖ 梯形图功能说明

指令将两个 16 位整数相除，得出一个 32 位结果，其中包括一个 16 位余数（高位）和一个 16 位商（低位）。32 位 OUT

的低位字（16 位）被用作除数。在 STL 中： $\text{OUT} / \text{IN1} = \text{OUT}$ 。



在启用（EN）输入端通过逻辑“1”激活 DIV（长整数乘）。IN1 和 IN2 相除，结果通过 OUT 查看。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
$\langle \text{IN1} \rangle$	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC	被除数
$\langle \text{IN2} \rangle$	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC	除数
$\langle \text{OUT} \rangle$	双整数	VD、ID、QD、MD、SMD、SD、LD、AC、*VD、*LD、*AC	除法结果

## ❖ 执行范例

● 范例 144 ● 使用整数与长整数相除指令。

梯形图	语句表指令
	<pre>LD I0.0 DIV VW10 VD200</pre>

## 应用

	IN1	IN2	OUT
			余数 商
除以数据	4000	/ 41	23 97
数据地址	VW202	VW10	VW200 VW202

注释：VD200 包含：VW200 和 VW202。

## ❖ 相关指令

/I

## 6.9 /I 指令：除以整数

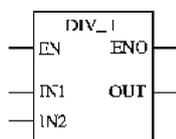
❖ 指令语法  $\text{DIV} \langle \text{IN1}(\text{除数}) \rangle \langle \text{OUT}(\text{除法结果}) \rangle$

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
$\langle \text{IN1} \rangle$	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC
$\langle \text{OUT} \rangle$	整数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

指令将两个 16 位整数相除，并产生一个 16 位商，不保留余数。如果结果大于一个字输出，则设置溢出位。在 STL 中：



(整数除)。IN1 除以 IN2，结果可通过 OUT 查看。

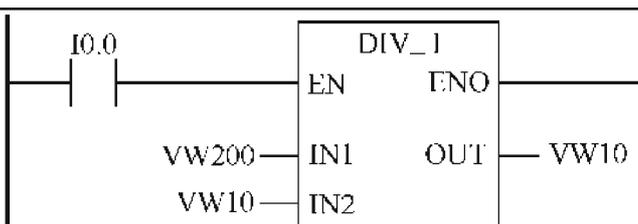
### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC	被除数
<IN2>	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC	除数
<OUT>	整数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC	除法结果

### ❖ 执行范例

● 范例 145 ● 使用整数相除指令。

梯 形 图	语 句 表 指 令												
	<pre>LD I0.0 /D VW10 VW200</pre>												
应用													
	<table border="1"> <thead> <tr> <th></th> <th>IN1</th> <th>IN2</th> <th>OUT</th> </tr> </thead> <tbody> <tr> <td>除以数据</td> <td>4000</td> <td>/ 40</td> <td>100</td> </tr> <tr> <td>数据地址</td> <td>VW200</td> <td>VW10</td> <td>VW200</td> </tr> </tbody> </table>		IN1	IN2	OUT	除以数据	4000	/ 40	100	数据地址	VW200	VW10	VW200
	IN1	IN2	OUT										
除以数据	4000	/ 40	100										
数据地址	VW200	VW10	VW200										

### ❖ 相关指令

\*I

## 6.10 /D 指令：除以长整数

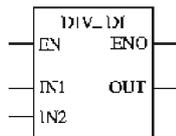
❖ 指令语法 DIV <IN1(除数)> <OUT(除法结果)>

## ◆ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	双整数	VD、ID、QD、MD、SD、SMD、LD、HC、AC、 常数、*VD、*LD、*AC
<OUT>	双整数	VD、ID、QD、MD、SMD、SD、LD、AC、*VD、 *LD、*AC

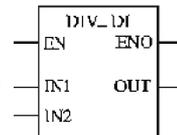
## ◆ 梯形图



## ◆ 梯形图功能说明

指令将两个 32 位整数相除，并产生一个 32

位商，不保留余数。在 STL 中： $OUT / IN1 = OUT$ 。



在启用 (EN)

输入端通过逻辑“1”激活 DIV\_DI（长整数除）。IN1 除以 IN2，结果可通过 OUT 查看。

## ◆ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	双整数	VD、ID、QD、MD、SD、SMD、LD、 HC、AC、常数、*VD、*LD、*AC	被除数
<IN2>	双整数	VD、ID、QD、MD、SD、SMD、LD、 HC、AC、常数、*VD、*LD、*AC	除数
<OUT>	双整数	VD、ID、QD、MD、SMD、SD、LD、 AC、*VD、*LD、*AC	除法结果

## ◆ 执行范例

● 范例 146 ● 使用长整数相除指令。

梯 形 图	语 句 表 指 令
	<pre>LD I0.0 /D MD10 MD20</pre>

续表

梯形图	语句表指令		
应用			
	IN1	IN2	OUT
除以数据	400000 /	80000	5
数据地址	MD20	MD10	MD20

## ❖ 相关指令

\*D

## 6.11 INCB 指令：递增字节

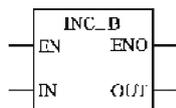
## ❖ 指令语法 INCB &lt;OUT(字节)&gt;

## ❖ 指令参数说明

参数的具体说明如下：

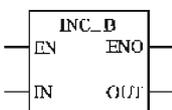
地 址	数 据 类 型	存 储 区
<IN1>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

指令在输入字节 (IN) 上加 1，并将结果置入 OUT 指定的变量中。递增字节运算不带符号。在 STL 中： $OUT + 1 = OUT$ 。



把输入字节 (IN) 加 1，并把结果存放到输出单元 (OUT)。在

LAD 中： $IN + 1 = OUT$ 。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	输入值

续表

参 数	数 据 类 型	内 存 区 域	说 明
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC	输出值

## ❖ 执行范例

● 范例 147 ● 使用递增字节指令。

梯 形 图	语 句 表 指 令
	<pre>LD    I0.0 INCB  AC0</pre>

应用

字节递增

AC0     125

增

AC0     126

## ❖ 相关指令

DECB

## 6.12 INCW 指令：递增字

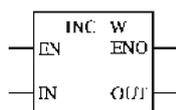
❖ 指令语法    INCW &lt;OUT(字)&gt;

❖ 指令参数说明

参数的具体说明如下：

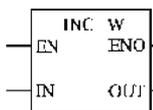
地 址	数 据 类 型	存 储 区
<IN1>	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC
<OUT>	整数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC

## ❖ 梯形图



**❖ 梯形图功能说明** 指令在输入字 (IN) 上加 1, 并将结果置入 OUT。递增字运算带符号 (16#7FFF > 16#8000)。在 STL 中:  $OUT + 1 = OUT$ 。

把输入字 (IN) 加 1, 并把结果存放到输出单元 (OUT)。在 LAD 中:  $IN + 1 = OUT$ 。



**❖ 梯形图参数说明**

参数的具体说明如下:

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC	输入值
<OUT>	整数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC	输出值

**❖ 执行范例**

● 范例 148 ● 使用递增字指令。

梯 形 图	语 句 表 指 令
	<pre>LD    I0.0 INCW  AC0</pre>

应用

字递增

AC0 125

增

AC0 126

**❖ 相关指令**

DECW

## 6.13 INCD 指令：递增双字

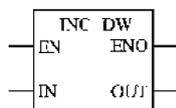
**❖ 指令语法** INCD <OUT(双字)>

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	双整数	VD、ID、QD、MD、SD、SMD、LD、HC、AC、 常数、*VD、*LD、*AC
<OUT>	双整数	VD、ID、QD、MD、SMD、SD、LD、AC、*VD、 *LD、*AC

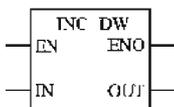
## ❖ 梯形图



## ❖ 梯形图功能说明

指令在输入双字 (IN) 上加 1, 并将结果置入 OUT。递增双字运算带符号 (16#7FFFFFFF>16#80000000)。在 STL 中:

$OUT + 1 = OUT$ 。把输入字 (IN) 加 1 并把结果存放到输出单元



(OUT)。在 LAD 中:  $IN + 1 = OUT$ 。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	双整数	VD、ID、QD、MD、SD、SMD、LD、HC、 AC、常数、*VD、*LD、*AC	输入值
<OUT>	双整数	VD、ID、QD、MD、SMD、SD、LD、AC、 *VD、*LD、*AC	输出值

## ❖ 执行范例

● 范例 149 ● 使用递增双字指令。

梯 形 图	语 句 表 指 令
	<pre>LD    I0.0 INCD  VD100</pre>

应用

双字递增

VD100      128000

增

VD100      128001

## ❖ 相关指令

DECD

# 6.14 DECB 指令：递减字节

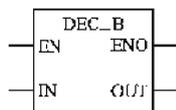
❖ 指令语法 DECB &lt;OUT(字节)&gt;

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

指令在输入字节（IN）上减 1，并将结果置入 OUT。递减字节运算不带符号。在 STL 中： $OUT - 1 = OUT$ 。在 LAD 中： $IN - 1 = OUT$ 。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	输入值
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC	输出值

❖ 执行范例

● 范例 150 ● 使用递减字节指令。

梯 形 图	语 句 表 指 令
	<pre>LD    I0.0 DECB  AC0</pre>

续表

梯形图	语句表指令
应用	
	字节递减
AC0	125 减
AC0	124

## ❖ 相关指令

INCB

## 6.15 DECB 指令：递减字

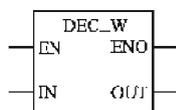
## ❖ 指令语法 DECB &lt;OUT(字)&gt;

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、 LW、AIW、常数、*VD、*LD、*AC
<OUT>	整数	VW、IW、QW、MW、SW、SMW、LW、AQW、 T、C、AC、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

指令在输入字 (IN) 上减 1，并将结果置入 OUT。递减字运算带符号 (16#7FFF > 16#8000)。在 STL 中：OUT - 1 = OUT。在 LAD 中：IN - 1 = OUT。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	整数	VW、IW、QW、MW、SW、SMW、T、C、 AC、LW、AIW、常数、*VD、*LD、*AC	输入值

续表

参 数	数 据 类 型	内 存 区 域	说 明
<OUT>	整 数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC	输出值

## ❖ 执行范例

⊙ 范例 151 ⊙ 使用递减字指令。

梯 形 图	语 句 表 指 令
	<pre>LD    I0.0 DECW  AC0</pre>

应用

字递减

AC0 125

减

AC0 124

## ❖ 相关指令

INCW

## 6.16 DECD 指令：递减双字

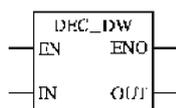
❖ 指令语法 DECD &lt;OUT(双字)&gt;

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	双 整 数	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*LD、*AC
<OUT>	双 整 数	VD、ID、QD、MD、SMD、SD、LD、AC、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

指令在输入双字 (IN) 上减 1, 并将结果置入 OUT。递减字运算带符号 (16#7FFF > 16#8000)。在 STL 中:  $OUT - 1 = OUT$ 。在 LAD 中:  $IN - 1 = OUT$ 。

## ❖ 梯形图参数说明

参数的具体说明如下:

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	双整数	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*LD、*AC	输入值
<OUT>	双整数	VD、ID、QD、MD、SMD、SD、LD、AC、*VD、*LD、*AC	输出值

## ❖ 执行范例

● 范例 152 ● 使用递减双字指令。

梯 形 图	语 句 表 指 令
	<pre>LD    I0.0 DECD  VD100</pre>

应用

	双字递减
VD100	128000
	减
VD100	127999

## ❖ 相关指令

INCD



## 第 7 章 中断指令

中断程序提供对特殊内部或外部事件的快速反应，应当优化中断例行程序，执行具体任务，然后使控制返回主例行程序。保持中断例行程序简明扼要，可加快执行速度，并且不会长时间延迟其他过程。否则，无法预测的条件会引起主程序控制的装置操作异常。

### 7.1 CRET1 指令：从中断（INT）有条件返回

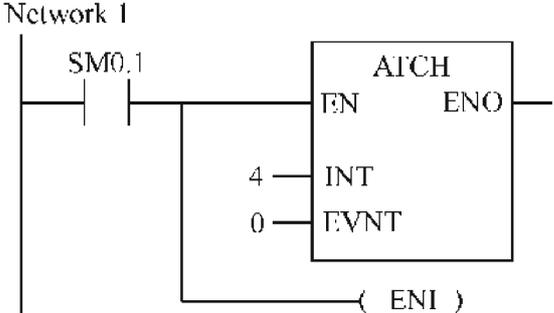
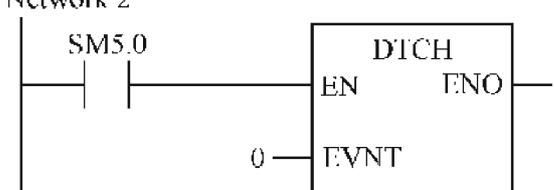
❖ 指令语法 CRET1

❖ 梯形图 —(RET1)

❖ 梯形图功能说明 指令可根据先前逻辑操作的条件从中断程序返回。可以用中断程序入口点处的中断程序标号来识别每个中断程序。中断程序由位于中断程序标号和无条件中断返回指令间的所有指令组成。中断程序在响应与之关联的内部或外部中断事件时执行。可以用无条件中断返回指令（RET1）或条件中断返回指令（CRET1）退出中断程序（从而将控制还给主程序），而无条件中断返回指令是必需的。

❖ 执行范例

● 范例 153 ● 使用有条件中断返回指令。

梯形图	语句表指令
 <p>Network 1</p> <p>SM0.1</p> <p>ATCH</p> <p>EN ENO</p> <p>4 INT</p> <p>0 EVNT</p> <p>( ENI )</p>	<p>//首次扫描时,定义中断程序4为处理 I0.0 上升沿中断事件 0 的中断程序。全局允许中断</p> <pre>LD      SM0.1 ATCH    4, 0 ENI</pre> <p>//如果发现 I/O 错误,禁止 I0.0 上升沿中断 (可选)</p> <pre>LD      SM5.0 DTCH    0</pre> <p>//当 M5.0 为 on 时,禁止所有中断</p> <pre>LD      M5.0 DISI</pre> <p>//I/O 上升沿中断程序:基于 I/O 错误的条件中断返回</p> <pre>LD      SM5.0 CRETI</pre>
 <p>Network 2</p> <p>SM5.0</p> <p>DTCH</p> <p>EN ENO</p> <p>0 EVNT</p>	
 <p>Network 3</p> <p>M5.0</p> <p>( DISI )</p>	
 <p>Network 1</p> <p>SM5.0</p> <p>( RETI )</p>	

### ❖ 相关指令

ENI

## 7.2 ENI 指令：启用中断

❖ 指令语法 ENI

❖ 梯形图 —( ENI )

❖ 梯形图功能说明 指令全局性启用所有附加中断事件进程。转换至 RUN（运行）模式时，中断开始时被禁止。一旦进入 RUN（运行）模式，可以通过执行全局中断允许指令，启用所有中断进程。

❖ 执行范例

① 范例 154 ② 使用启用中断指令。

梯 形 图	语句表指令
	<pre>//首次扫描时，将中断例行程序 INT_0 定义为下降沿中断，用于 I0.0，并全局启用中断 LD    SM0.1 ATCH  INT_0 I ENI</pre>

### ❖ 相关指令

CRETI

## 7.3 DISI 指令：禁止中断

### ❖ 指令语法 DISI

### ❖ 梯形图 —( DISI )

❖ 梯形图功能说明 指令全局性禁止所有中断事件进程。转换至 RUN（运行）模式时，中断开始时被禁止。一旦进入 RUN（运行）模式，可以通过执行全局中断允许指令，启用所有中断进程，执行中断禁止指令会禁止处理中断，但是现用中断事件将继续入队等候。

### ❖ 执行范例

#### ● 范例 155 ● 使用禁止中断指令。

梯 形 图	语句表指令
	<pre>//首次扫描时，将中断例行程序 INT_0 定义为下降沿中断，用于 I0.0，并全局启用中断 LD    SM0.1 ATCH  INT_0 I ENI  // 当 M5.0 打开时，禁用所有的中断 LD    M5.0 DISI</pre>

### ❖ 相关指令

ENI

## 7.4 ATCH 指令：中断连接

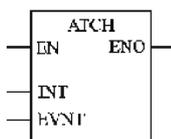
❖ 指令语法 ATCH <INT><EVNT>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<INT>	字节	常数 0-127
<EVNT>	字节	常数 CPU 221/222: 0-12, 19-23, 27-33 CPU 224: 0-23; 27-33 CPU 226 / 226XM: 0-33

❖ 梯形图



❖ 梯形图功能说明

指令将中断事件（EVNT）与中断例行程序号码（INT）相联系，并启用中断事件。在激活中断例行程序之前，必须在中断事件和希望在事件发生时执行的程序段之间建立关联。使用“中断连接”指令将中断事件（由中断事件号码指定）与程序段（由中断例行程序号码指定）联系在一起。可以将多个中断事件附加在一个中断例行程序上，但一个事件不能同时附加在多个中断例行程序上。

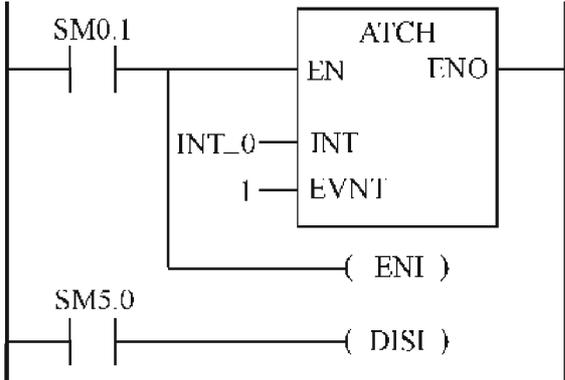
❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<INT>	字节	常数 0-127	中断标号
<EVNT>	字节	常数 CPU 221/222: 0-12, 19-23, 27-33 CPU 224: 0-23; 27-33 CPU 226 / 226XM: 0-33	中断事件

❖ 执行范例

○ 范例 156 ○ 使用中断连接指令。

梯形图	语句表指令
	<pre>//首次扫描时，将中断例行程序 INT_0 定义为下降沿中断，用于 I0.0，并全局启用中断 LD      SM0.1 ATCH   INT_0 1 ENI  // 当 M5.0 打开时，禁用所有的中断 LD      M5.0 DISI</pre>

### ❖ 相关指令

DTCH

## 7.5 DTCH 指令：中断分离

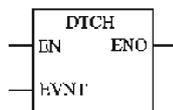
### ❖ 指令语法 DTCH <EVNT>

### ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<EVNT>	字节	PLC 221/222: 0-12, 19-23, 27-33 PLC 224: 0-23; 27-33 PLC 226 / 226XM: 0-33

### ❖ 梯形图



### ❖ 梯形图功能说明

指令取消中断事件（EVNT）与所有中断例行程序之间的关联，并禁用中断事件。当将一个中断事件附加在一个中断例行程序上时，会自动启用中断。如果用全局禁用中断指令禁用所有的中断，则每次出现的中断事件均入队等候，直至使用全局启用中断指令或中断队列溢出重新启用中断。可以使用“中断分离”指令断开中断事件与中断例行程序之间的联系，从而禁用单个中断事件。“中断分离”指令使中断返回至非现用或忽略状态。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<EVNT>	字节	PLC 221/222: 0-12, 19-23, 27-33 PLC 224: 0-23; 27-33 PLC 226 / 226XM: 0-33	中断事件

### ❖ 执行范例

◎ 范例 157 ◎ 使用中断分离指令。

梯 形 图	语 句 表 指 令
	<pre>//首次扫描时，将中断例行程序 INT_0 定义为 //下降沿中断，用于 I0.0，并全局启用中断 LD      SM0.1 ATCH   INT_0 1 ENI  //如果检测到一则 I/O 错误，禁用用于 I0.0 的 //下降沿中断 LD      M5.0 DTCH   1</pre>

### ❖ 相关指令

ATCH

## 7.6

## CEVNT 指令：清除中断事件

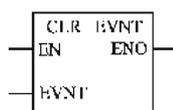
❖ 指令语法 CEVNT <EVNT>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<EVNT>	字节	常量

### ❖ 梯形图



**◆ 梯形图功能说明** 指令会删除中断队列中所有类型为 EVNT 的中断事件。此指令用于清除不必要的中断，后者可能由假传感器输出暂态造成。

### ◆ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<EVNT>	字 节	常 量	中 断 事 件

### ◆ 执行范例

● 范例 158 ● 使用中断清除指令。

梯 形 图	语 句 表 指 令
<p>网络 1 指令向导 HSC</p> <p>网络 2 消除机器颤动引起的不必要中断</p>	<p>//显示了一个处于正交模式的高速计数器，利用 CLR_EVNT 指令删除因机器振动而可能发生的假中断。假设某截光步进传感器停止在明暗转换的交界处，那么小幅度的机器振动就可能产生假中断</p> <p>NETWORK 1 //指令向导 HSC</p> <pre>LD SM0.0 MOVB 16#A0, SMB47 //设置控制位：写入预置； MOVD +6, SMD52 //PV = 6; ATCH HSC1_STEP1, 13 //中断 HSC1_STEP1: HC1 的 CV = PV</pre> <p>NETWORK 2 //清除由机器振动造成的不必要中断</p> <pre>LD SM0.0 CEVNT 13</pre>



## 第 8 章 字逻辑指令

字逻辑指令按照布尔逻辑按位比较字（16 位）和双字（32 位）对。如果输出 OUT 的结果不等于 0，将把状态字的标志位设置为“1”。如果输出 OUT 的结果等于 0，将把状态字的标志位设置为“0”。

### 8.1 INVB 指令：取反字节

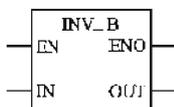
❖ 指令语法 INVB <OUT(字节)>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

指令对输入字节 IN 执行求补操作，并将结果载入内存位置 OUT。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	输入值
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC	输出值

### ❖ 执行范例

⊙ 范例 159 ⊙ 使用取反字节指令。

梯 形 图	语 句 表 指 令
	<pre>LD    I0.0 INVB  VB0</pre>
应用	
	<pre>VB0    1001 0101         取反 VB0    0110 1010</pre>

### ❖ 相关指令

ANDB、ORB、XORB

## 8.2 INVW 指令：取反字

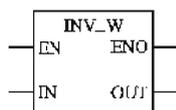
❖ 指令语法 INVW <OUT(字)>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	字	VW、IW、QW、MW、SW、SMW、LW、T、C、AIW、AC、常数、*VD、*AC、*LD
<OUT>	字	W、IW、QW、MW、SW、SMW、LW、T、C、AC、*VD、*LD、*AC

### ❖ 梯形图



### ❖ 梯形图功能说明

指令对输入字 IN 执行求补操作，并将结果

载入内存位置 OUT。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字	VW、IW、QW、MW、SW、SMW、LW、T、C、AIW、AC、常数、*VD、*AC、*LD	输入值
<OUT>	字	W、IW、QW、MW、SW、SMW、LW、T、C、AC、*VD、*LD、*AC	输出值

### ❖ 执行范例

◎ 范例 160 ◎ 使用取反字指令。

梯形图	语句表指令
	<pre>LD    I4.0 INVW  AC0</pre>
应用	
VB0	1101 0111 1001 0101
	取反
VB0	0010 1000 0110 1010

### ❖ 相关指令

ANDW、ORW、XORW

## 8.3 INVD 指令：取反双字

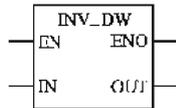
❖ 指令语法 INVD <OUT(双字)>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	双字	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*AC、*LD
<OUT>	双字	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*AC、*LD

## ❖ 梯形图



## ❖ 梯形图功能说明

指令对输入双字 IN 执行求补操作，并将结果载入内存位置 OUT。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	双字	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*AC、*LD	输入值
<OUT>	双字	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*AC、*LD	输出值

## ❖ 执行范例

☉ 范例 161 ☉ 使用取反双字指令。

梯 形 图	语 句 表 指 令
	<pre>LD    I4.0 INVD  VD0</pre>

应用

VD0 1101 0111 1100 0110 1001 0101

取反

VD0 0010 1000 0011 1001 0110 1010

## ❖ 相关指令

ANDD、ORD、XORD

## 8.4

## ANDB 指令：与运算字节

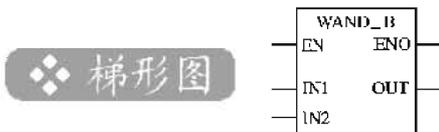
## ❖ 指令语法

ANDB <IN1(字节)><OUT(字节)>

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字节	VD、ID、QD、MD、SD、SMD、LD、HC、AC、 常数、*VD、*AC、*LD
<IN2>	字节	VD、ID、QD、MD、SD、SMD、LD、HC、AC、 常数、*VD、*AC、*LD
<OUT>	字节	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、 *AC、*LD



❖ 梯形图功能说明 指令对两个字节输入值（IN1 和 IN2）的对应位执行 AND（与运算）操作，并在内存位置（OUT）中载入结果。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字节	VD、ID、QD、MD、SD、SMD、LD、HC、 AC、常数、*VD、*AC、*LD	输入值
<IN2>	字节	VD、ID、QD、MD、SD、SMD、LD、HC、 AC、常数、*VD、*AC、*LD	输入值
<OUT>	字节	VD、ID、QD、MD、SD、SMD、LD、AC、 *VD、*AC、*LD	输出值

❖ 执行范例

● 范例 162 ● 使用与运算字节指令。

梯 形 图	语 句 表 指 令
	<pre>LD    I0.0 ANDB  VB0 VB1</pre>

应用

VB0	1001 0101
	与
VB1	1110 1110
	等于
VB1	1000 0100

## ❖ 相关指令

INVB、ORB、XORB

# 8.5 ANDW 指令：与运算字

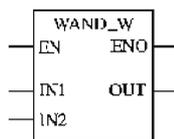
❖ 指令语法 ANDW &lt;IN1(字)&gt;&lt;OUT(字)&gt;

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字	VW、IW、QW、MW、SW、SMW、T、C、AIW、LW、AC、常数、*VD、*AC、*LD
<IN2>	字	VW、IW、QW、MW、SW、SMW、T、C、AIW、LW、AC、常数、*VD、*AC、*LD
<OUT>	字	VW、IW、QW、MW、SW、SMW、T、C、LW、AC、*VD、*AC、*LD

## ❖ 梯形图



## ❖ 梯形图功能说明

指令对两个字输入值（IN1 和 IN2）的对应位执行 AND（与运算）操作，并在内存位置（OUT）载入结果。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字	VW、IW、QW、MW、SW、SMW、T、C、AIW、LW、AC、常数、*VD、*AC、*LD	输入值
<IN2>	字	VW、IW、QW、MW、SW、SMW、T、C、AIW、LW、AC、常数、*VD、*AC、*LD	输入值
<OUT>	字	VW、IW、QW、MW、SW、SMW、T、C、LW、AC、*VD、*AC、*LD	输出值

## ❖ 执行范例

○ 范例 163 ○ 使用与运算字指令。

梯形图	语句表指令
	<pre>LD    I0.0 ANDW  AC0 AC1</pre>
应用	
AC0	0000 1111 1001 0101
	与
AC1	1100 1110 1100 1110
	等于
AC1	0000 1110 1000 0100

### ❖ 相关指令

INVW、ORW、XORW

## 8.6 ANDD 指令：与运算双字

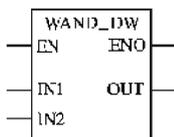
❖ 指令语法 ANDD <IN1(双字)><OUT(双字)>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	双字	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*AC、*LD
<IN2>	双字	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*AC、*LD
<OUT>	双字	VD、ID、QD、MD、SD、MD、LD、AC、*VD、*AC、*LD

### ❖ 梯形图



### ❖ 梯形图功能说明

指令对两个双字输入值的对应位执行 AND（与运算）操作，并在双字中载入结果（OUT）。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	双字	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*AC、*LD	输入值
<IN2>	双字	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*AC、*LD	输入值
<OUT>	双字	VD、ID、QD、MD、SD、MD、LD、AC、*VD、*AC、*LD	输出值

### ❖ 执行范例

● 范例 164 ● 使用与运算双字指令。

梯 形 图	语 句 表 指 令
	<pre>LD    I0.0 ANDD VD0 VD1</pre>

应用

VD0 1101 0111 1100 0110 1001 0101

与

VD1 1110 1000 1111 0011 1010 1110

等于

VD1 1100 0000 1100 0010 1000 0100

### ❖ 相关指令

INVD、ORD、XORD

## 8.7 ORB 指令：或运算字节

❖ 指令语法 ORB <IN1(字节)><OUT(字节)>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD

续表

地 址	数 据 类 型	存 储 区
<IN2>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC



◆ 梯形图功能说明 指令对两个字节输入值（IN1 和 IN2）的对应位执行 OR（或运算）操作，并在内存位址（OUT）中载入结果。

◆ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	输入值
<IN2>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	输入值
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC	输出值

◆ 执行范例

◎ 范例 165 ◎ 使用或运算字节指令。

梯 形 图	语 句 表 指 令
	<pre>LD    I0.0 ORB   VB0 VB1</pre>

应用

VB0	1001 0101
	或
VB1	1110 1110
	等于
VB1	1111 1111

## ❖ 相关指令

ANDB、INVB、XORB



## ORW 指令：或运算字

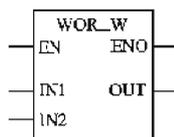
❖ 指令语法 ORW &lt;IN1(字)&gt;&lt;OUT(字)&gt;

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字	VW、IW、QW、MW、SW、SMW、LW、T、C、AIW、AC、常数、*VD、*AC、*LD
<IN2>	字	VW、IW、QW、MW、SW、SMW、LW、T、C、AIW、AC、常数、*VD、*AC、*LD
<OUT>	字	W、IW、QW、MW、SW、SMW、LW、T、C、AC、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

指令对两个字输入值的对应位执行 OR（或运算）操作，并在内存位置（OUT）载入结果。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字	VW、IW、QW、MW、SW、SMW、LW、T、C、AIW、AC、常数、*VD、*AC、*LD	输入值
<IN2>	字	VW、IW、QW、MW、SW、SMW、LW、T、C、AIW、AC、常数、*VD、*AC、*LD	输入值
<OUT>	字	W、IW、QW、MW、SW、SMW、LW、T、C、AC、*VD、*LD、*AC	输出值

## ❖ 执行范例

○ 范例 166 ○ 使用或运算字指令。

梯形图	语句表指令
	<pre>LD    I0.0 ORW   AC0 AC1</pre>
应用	
AC0	0000 1111 1001 0101
	或
AC1	1100 1110 1100 1110
	等于
AC1	1100 1111 1101 1111

### ❖ 相关指令

INVW、ANDW、XORW

## 8.9

## ORD 指令：或运算双字

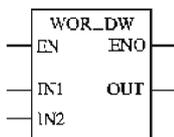
❖ 指令语法    **ORD <IN1(双字)><OUT(双字)>**

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	双字	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*AC、*LD
<IN2>	双字	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*AC、*LD
<OUT>	双字	VD、ID、QD、MD、SD、MD、LD、AC、*VD、*AC、*LD

### ❖ 梯形图



### ❖ 梯形图功能说明

指令对两个双字输入值的对应位执行 OR（或运算）操作，并在双字中载入结果（OUT）。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	双字	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*AC、*LD	输入值
<IN2>	双字	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*AC、*LD	输入值
<OUT>	双字	VD、ID、QD、MD、SD、MD、LD、AC、*VD、*AC、*LD	输出值

### ❖ 执行范例

● 范例 167 ● 使用或运算双字指令。

梯 形 图	语 句 表 指 令
	<pre>LD    I0.0 ORD   VD0 VD1</pre>

应用

VD0 1101 0001 1100 0100 1001 0001

或

VD1 1110 1000 1111 0011 1010 1110

等于

VD1 1111 1001 1111 0111 1011 1111

### ❖ 相关指令

INVD、ANDD、XORD

## 8.10 XORB 指令：异或运算字节

❖ 指令语法 XORB <IN1(字节)><OUT(字节)>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<IN2>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC



◆ 梯形图功能说明 指令对两个字节输入数值（IN1 和 IN2）的对应位执行 XOR（异-或运算）操作，并在内存位置（OUT）中载入结果。

◆ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	输入值
<IN2>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	输入值
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC	输出值

◆ 执行范例

● 范例 168 ● 使用异或运算字节指令。

梯 形 图	语 句 表 指 令
	<pre>LD    I0.0 XORB  VB0 VB1</pre>

应用

VB0	1001 0101
	异或
VB1	1110 1110
	等于
VB1	0111 1011

## ❖ 相关指令

ANDB、INVB、ORB

# 8.11 XORW 指令：异或运算字

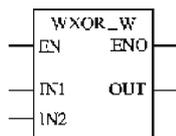
❖ 指令语法 XORW &lt;IN1(字)&gt;&lt;OUT(字)&gt;

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字	VW、IW、QW、MW、SW、SMW、T、C、AIW、LW、AC、常数、*VD、*AC、*LD
<IN2>	字	VW、IW、QW、MW、SW、SMW、T、C、AIW、LW、AC、常数、*VD、*AC、*LD
<OUT>	字	VW、IW、QW、MW、SW、SMW、T、C、LW、AC、*VD、*AC、*LD

## ❖ 梯形图



## ❖ 梯形图功能说明

指令对两个字输入数值（IN1 和 IN2）的对应位执行 XOR（异-或运算）操作，并在内存位置（OUT）载入结果。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字	VW、IW、QW、MW、SW、SMW、T、C、AIW、LW、AC、常数、*VD、*AC、*LD	输入值
<IN2>	字	VW、IW、QW、MW、SW、SMW、T、C、AIW、LW、AC、常数、*VD、*AC、*LD	输入值
<OUT>	字	VW、IW、QW、MW、SW、SMW、T、C、LW、AC、*VD、*AC、*LD	输出值

## ❖ 执行范例

● 范例 169 ● 使用异或运算字指令。

梯 形 图	语句表指令
	<pre>LD    I0.0 XORW  AC0 AC1</pre>

## 应用

AC0      0000 1111 1001 0101

异或

AC1      1100 1110 1100 1110

等于

AC1      1100 0001 0101 1011

## ❖ 相关指令

INW、ANDW、ORW

## 8.12 XORD 指令：异或运算双字

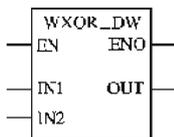
❖ 指令语法      XORD <IN1(双字)><OUT(双字)>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	双字	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*AC、*LD
<IN2>	双字	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*AC、*LD
<OUT>	双字	VD、ID、QD、MD、SD、MD、LD、AC、*VD、*AC、*LD

## ❖ 梯形图



## ❖ 梯形图功能说明

指令对两个双字输入值的对应位执行 XOR 操作，并在双字中载入结果（OUT）。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	双字	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*AC、*LD	输入值
<IN2>	双字	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*AC、*LD	输入值
<OUT>	双字	VD、ID、QD、MD、SD、MD、LD、AC、*VD、*AC、*LD	输出值

### ❖ 执行范例

◎ 范例 170 ◎ 使用异或运算双字指令。

梯 形 图	语 句 表 指 令
	<pre>LD    I0.0 XORD  VD0 VD1</pre>
应用	
VD0 1101 0001 1100 0100 1001 0001 或 VD1 1110 1000 1111 0011 1010 1110 等于 VD1 0011 1001 0011 0111 0011 1111	

### ❖ 相关指令

INVD、ANDD、ORD



## 第9章 移动指令

移动指令可以对输入或输出模块与存储区之间或在存储器之间的信息进行交换。CPU 在每次扫描中将无条件执行这些指令。

### 9.1 MOV B 指令：移动字节

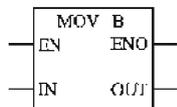
❖ 指令语法 XORD <IN(字节)><OUT(字节)>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

将输入字节（IN）移至输出字节（OUT），不改变原来的数值。设置 ENO = 0 的错误条件：0006（间接地址）。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	输入值

续表

参 数	数 据 类 型	内 存 区 域	说 明
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC	输出值

## ❖ 执行范例

● 范例 171 ● 使用移动字节指令。

梯 形 图	语 句 表 指 令
	<pre>LD    I0.0 MOV_B  VB0 VB10 //移动的结果是 VB10=VB0</pre>

## ❖ 相关指令

BMB、SWAP

## 9.2 MOVW 指令：移动字

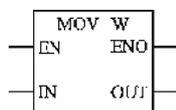
❖ 指令语法 MOVW <IN(字)><OUT(字)>

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	字、整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC
<OUT>	字、整数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

将输入字（IN）移至输出字（OUT），不改变原来的数值。设置 ENO = 0 的错误条件：0006（间接地址）。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字、整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC	输入值
<OUT>	字、整数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC	输出值

### ❖ 执行范例

◎ 范例 172 ◎ 使用移动字指令。

梯 形 图	语 句 表 指 令
	<pre>LD    I0.0 MOVW  VW0 VW10 //移动的结果是 VW10=VW0</pre>

### ❖ 相关指令

BMW

## 9.3 MOVD 指令：移动双字

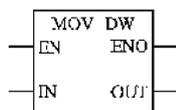
❖ 指令语法 MOVW <IN(双字)><OUT(双字)>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	双字、双整数	VD、ID、QD、MD、SD、SMD、LD、HC、&VB、&IB、&QB、&MB、&SB、&T、&C、&SMB、&AIW、&AQW、AC、常数、*VD、*LD、*AC
<OUT>	双字、双整数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明 将输入双字（IN）移至输出双字（OUT），不改变原来的数值。设置 ENO = 0 的错误条件：0006（间接地址）。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	双字、双整数	VD、ID、QD、MD、SD、SMD、LD、HC、&VB、&IB、&QB、&MB、&SB、&T、&C、&SMB、&AIW、&AQW、AC、常数、*VD、*LD、*AC	输入值
<OUT>	双字、双整数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC	输出值

### ❖ 执行范例

● 范例 173 ● 使用移动双字指令。

梯 形 图	语 句 表 指 令
	<pre>LD    I0.0 MOV_DW VD0 VD10 //移动的结果是 VD10=VD0</pre>

### ❖ 相关指令

BMD

## 9.4 MOVR 指令：移动实数

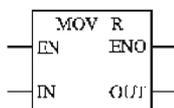
❖ 指令语法 MOVR <IN(实数)><OUT(实数)>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、常数、*VD、*LD、*AC
<OUT>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

将 32 位、实数输入双字 (IN) 移至输出双字 (OUT)，不改变原来的数值。设置 ENO = 0 的错误条件：0006 (间接地址)。

### ◆ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、 常数、*VD、*LD、*AC	输入值
<OUT>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、 *VD、*LD、*AC	输出值

### ◆ 执行范例

◎ 范例 174 ◎ 使用移动实数指令。

梯 形 图	语 句 表 指 令
	<pre>LD    I0.0 MOV_R AC0 AC2 //移动的结果是 AC2=AC0</pre>

## 9.5

## BMB 指令：成块移动字节

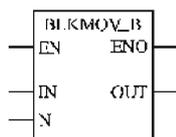
◆ 指令语法 BMB <IN(字节)><OUT(字节)> <N(个数)>

◆ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、 *AC
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、 *AC、*VD、*LD
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、 *LD、*AC

◆ 梯形图



◆ 梯形图功能说明

将字节数目 (N) 从输入地址 (IN) 移至输

出地址 (OUT)。N 的范围为 1~255。设置 ENO = 0 的错误条件：0006 (间接地址)；0091 (操作数超出范围)。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC	输入值
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	字节个数
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC	输出值

### ❖ 执行范例

● 范例 175 ● 使用成块移动字节指令。

梯 形 图	语 句 表 指 令
	<pre>//将数组 1 (VB20 至 VB23) 移至数组 2 (VB100 至 VB103) LD  I2.1 BMB  VB20 VB100 4</pre>

应用

数组 1 数据	30	31	32	33
数据地址	VB20	VB21	VB22	VB23
块移动执行载入数组 2				
数组 2 数据	30	31	32	33
数据地址	VB100	VB101	VB102	VB103

### ❖ 相关指令

MOVB、SWAP

## 9.6

## BMW 指令：成块移动字

❖ 指令语法 BMW <IN(字)><OUT(字)> <N(个数)>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	字	VW、IW、QW、MW、SW、SMW、LW、T、C、AIW、*VD、*LD、*AC
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<OUT>	字	VW、IW、QW、MW、SW、SMW、LW、T、C、AQW、*VD、*LD、*AC



◆ 梯形图功能说明 将字数目 (N) 从输入地址 (IN) 移至输出地址 (OUT)。N 的范围是 1~255。设置 ENO = 0 的错误条件：0006 (间接地址)；0091 (操作数超出范围)。

◆ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字	VW、IW、QW、MW、SW、SMW、LW、T、C、AIW、*VD、*LD、*AC	输入值
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	字节个数
<OUT>	字	VW、IW、QW、MW、SW、SMW、LW、T、C、AQW、*VD、*LD、*AC	输出值

◆ 执行范例

● 范例 176 ● 使用块移动字指令。

梯 形 图		语 句 表 指 令		
		<pre>//将数组 1 (VW10 至 VW13) 移至数组 2 (VW20 至 VW23) LD  I2.1 BMW  VW10 VW20 4</pre>		
应用				
数组 1 数据	500	501	502	503
数据地址	VW10	VW11	VW12	VW13

续表

梯 形 图		语句表指令		
块移动执行载入数组 2				
数组 2 数据	500	501	502	503
数据地址	VW20	VW21	VW22	VW23

## ❖ 相关指令

MOVW

## 9.7 BMD 指令：成块移动双字

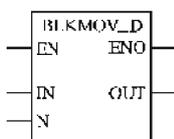
❖ 指令语法 BMD &lt;IN(双字)&gt;&lt;OUT(双字)&gt; &lt;N(个数)&gt;

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	双字	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*AC、*LD
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<OUT>	双字	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*AC、*LD

## ❖ 梯形图



## ❖ 梯形图功能说明

将双字数目 (N) 从输入地址 (IN) 移至输出地址 (OUT)。N 的范围是 1~255。设置 ENO = 0 的错误条件：0006 (间接地址)；0091 (操作数超出范围)。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	双字	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*AC、*LD	输入值

续表

参 数	数 据 类 型	内 存 区 域	说 明
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	字节个数
<OUT>	双字	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*AC、*LD	输出值

## ❖ 执行范例

● 范例 177 ● 使用块移动双字指令。

梯 形 图	语 句 表 指 令
	<pre>//将数组 1 (VD10 至 VD13) 移 至数组 2 (VD20 至 VD23) LD  I2.1 BMD  VD10 VD20 4</pre>

应用

数组 1 数据	70000	70001	70002	70003
数据地址	VD10	VD11	VD12	VD13
块移动执行载入数组 2				
数组 2 数据	70000	70001	70002	70003
数据地址	VD20	VD21	VD22	VD23

## ❖ 相关指令

MOVD

## 9.8 SWAP 指令：交换字节

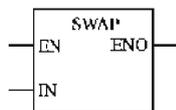
❖ 指令语法 SWAP <IN(字)>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	字	VW、IW、QW、MW、SW、SMW、T、C、LW、AC、*VD、*AC、*LD

## ❖ 梯形图



## ❖ 梯形图功能说明

交换字（IN）的最高位字节和最低位字节。  
设置 ENO = 0 的错误条件：0006（间接地址）。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字	VW、IW、QW、MW、SW、SMW、T、C、 LW、AC、*VD、*AC、*LD	输入值

## ❖ 执行范例

● 范例 178 ● 使用交换字节指令。

梯 形 图	语 句 表 指 令
	<pre>//将 VW50 的高字节与低字节 //交换 LD    I2.1 SWAP  VW50</pre>

应用

	IN		IN（执行之后）
交换数据（HEX）	D6 C3	交换字节	C3 D6
数据地址	VW50		VW50

## 9.9

## BIR 指令：移动字节立即读取

## ❖ 指令语法

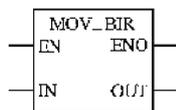
BIR <IN(字节)><OUT(字节)>

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	字节	IB、*VD、*LD、*AC
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、 *AC

## ❖ 梯形图



## ❖ 梯形图功能说明

读取实际输入 IN（作为字节），并将结果写入 OUT，但进程映象寄存器未更新。设置 ENO = 0 的错误条件：0006（间接地址）；无法存取扩充模块（仅限立即指令）。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字节	IB、*VD、*LD、*AC	输入值
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC	输出值

## ❖ 执行范例

● 范例 179 ● 使用移动字节立即读取指令。

梯 形 图	语 句 表 指 令
	<pre>LD    I2.1 BIR   IB0 VB10 //移动的结果是 VB10=IB0</pre>

## ❖ 相关指令

BIW

## 9.10 BIW 指令：移动字节立即写入

## ❖ 指令语法

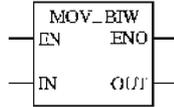
BIW <IN(字节)><OUT(字节)>

## ❖ 指令参数说明

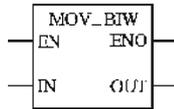
参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<OUT>	字节	QB、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明



指令从位置 IN 读取数值并写入

(以字节为单位) 实际输入 OUT 以及对应的“进程图像”位置。设置 ENO = 0 的错误条件：0006 (间接地址)；无法存取扩充模块 (仅限立即指令)。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、 常数、*AC、*VD、*LD	输入值
<OUT>	字节	QB、*VD、*LD、*AC	输出值

## ❖ 执行范例

◎ 范例 180 ◎ 使用移动字节立即写入指令。

梯 形 图	语 句 表 指 令
	<pre>LD    I2.1 BIW   VB0 QB10 //移动的结果是 QB10=VB0</pre>

## ❖ 相关指令

BIR



## 第 10 章 程序控制指令

程序控制指令可以实现 PLC 程序的循环、跳转、并行、子程序调用等流程控制功能，不仅方便编程人员的设计，也为现场调试提供方便灵活的支持。

### 10.1 FOR 指令：循环开始

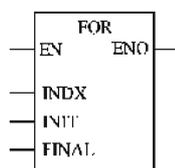
❖ 指令语法 FOR <INDEX><INITIAL><FINAL>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<INDEX>	整数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC
<INIT>	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC
<FINAL>	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

指令执行 FOR 和 NEXT 之间的指令。必须指定索引值或当前循环计数 (INDEX)、起始值 (INIT) 和结束值 (FINAL)。

FOR 指令和 NEXT 指令必须成对使用。FOR 标记循环的开始，NEXT 标记循环的结束。

例如，假定 INIT 值等于 1，FINAL 值等于 10，FOR 与 NEXT 之间的指令被执行 10 次，INDX 值递增：1、2、3、…、10。如果起始值大于结束值，则不执行循环。每次执行 FOR 和 NEXT 之间的指令后，INDX 值递增，并将结果与结束值比较。如果 INDX 大于结束值，循环则终止。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<INDX>	整数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC	输入值
<INIT>	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC	输入值
<FINAL>	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC	输入值

### ❖ 执行范例

#### ◎ 范例 181 ◎ 使用循环开始指令。

梯 形 图	语 句 表 指 令
	<pre>// I2.0 打开时，外部循环（箭头 1）被 // 执行 100 次 LD I2.0 FOR VW100 +1 +100  // I2.1 打开时，内部循环（箭头 2） // 为每次执行外部循环执行两次 LD I2.1 FOR VW225 +1 +2  // 循环 2 结束 NEXT  // 循环 1 结束 NEXT</pre>

## ❖ 相关指令

NEXT

## 10.2 NEXT 指令：循环结束

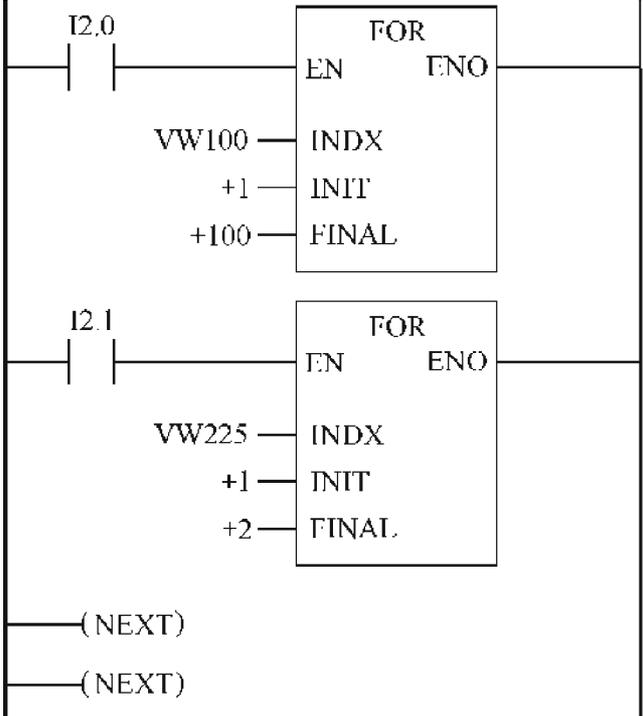
❖ 指令语法 NEXT

❖ 梯形图 —(NEXT)

❖ 梯形图功能说明 标记循环的结束并且置栈顶值为 1。每条 FOR 指令要求一个 NEXT 指令。可以复原 FOR/NEXT 循环(在 FOR/NEXT 循环中放置一个 FOR/NEXT 循环)，深度可达 8。

## ❖ 执行范例

● 范例 182 ● 使用循环结束指令。

梯 形 图	语句表指令
	<pre> // I2.0 打开时，外部循环（箭头 1） // 被执行 100 次 LD I2.0 FOR VW100 +1 +100  // I2.1 打开时，内部循环（箭头 2） // 为每次执行外部循环执行两次 LD I2.1 FOR VW225 +1 +2  // 循环 2 结束 NEXT  // 循环 1 结束 NEXT </pre>

## ❖ 相关指令

FOR

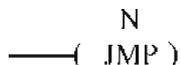
# 10.3 JMP 指令：跳接至标签

❖ 指令语法 JMP <n>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<n>	字	常数 (0~255)

❖ 梯形图 

❖ 梯形图功能说明 指令对程序中的指定标签 (n) 执行分支操作。跳转接受时，堆栈顶值始终为逻辑 1。可以在主程序、子程序或中断例行程序中使用“跳转”指令。“跳转”及其对应的“标签”指令必须始终位于相同的代码段中（主程序、子程序或中断例行程序）。不能从主程序跳转至子程序或中断例行程序中的标签，与此相似，也不能从子程序或中断例行程序跳转至该子程序或中断例行程序之外的标签。

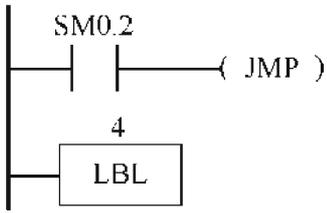
❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<n>	字	常数 (0~255)	输入值

❖ 执行范例

◎ 范例 183 ◎ 使用跳接至标签指令。

梯 形 图	语 句 表 指 令
	<pre>// 如果保留数据尚未丢失，跳转至 LBL4 LDN SM0.2 JMP 4 LBL 4</pre>

❖ 相关指令

LBL

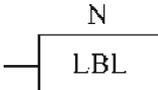
## 10.4 LBL 指令：语句标签

❖ 指令语法 LBL <n>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<n>	字	常数 (0~255)

❖ 梯形图 

❖ 梯形图功能说明 标记跳转目的地的位置 (n)。

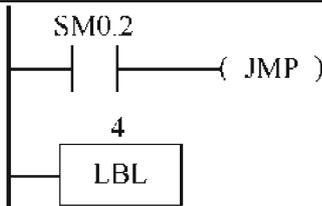
❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<n>	字	常数 (0~255)	输入值

❖ 执行范例

● 范例 184 ● 使用语句标签指令。

梯 形 图	语 句 表 指 令
	<pre>// 如果保留数据尚未丢失，跳转至 LBL4 LDN SM0.2 JMP 4 LBL 4</pre>

❖ 相关指令

JMP

## 10.5 LSCR 指令：载入顺序控制继电器

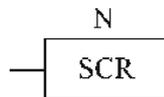
❖ 指令语法 LSCR <N>

### ◆ 指令参数说明

参数的具体说明如下：

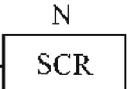
地 址	数 据 类 型	存 储 区
<N>	布尔	S

### ◆ 梯形图



### ◆ 梯形图功能说明

用指令 (N) 引用的 S 位数值载入 SCR 和逻辑堆栈。SCR 段被 SCR 堆栈的结果数值激励或取消激励。SCR 堆栈数值被复制至逻辑堆栈的顶端，以便方框或输出线圈可直接与左电源杆连接，

无需插入触点。— 标记一个顺序控制继电器 (SCR) 段的开始。当  $n=1$  时，允许该 SCR 段工作。SCR 段必须用 SCRE 指令结束。

### ◆ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<N>	布尔	S	输入值

### ◆ 执行范例

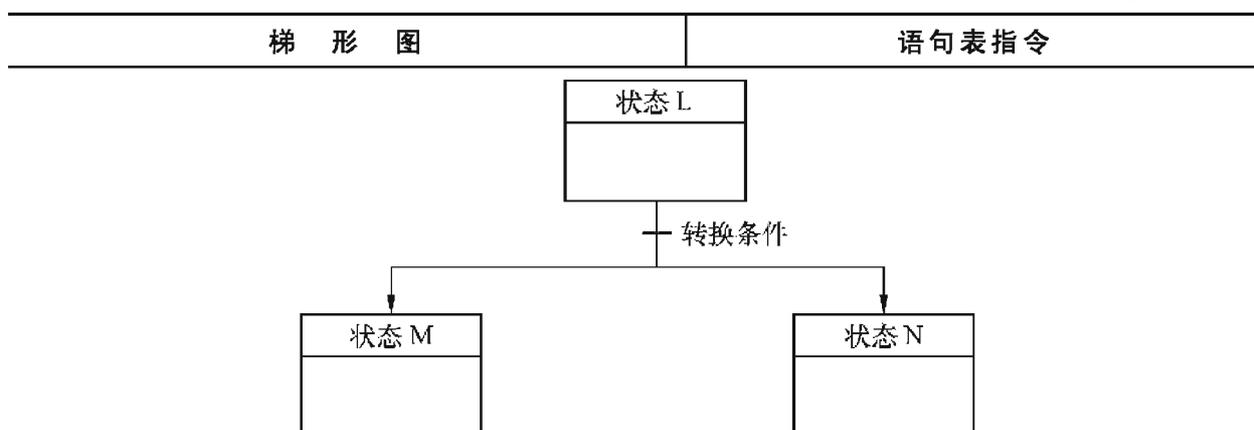
#### ● 范例 185 ● 使用载入顺序控制继电器指令。

梯 形 图	语 句 表 指 令
	<pre>// 状态 L 控制区开始 LSCR S3.4 LD M2.3 A I2.1 SCRT S3.5 // 转换至状态 M SCRT S6.5 // 转换至状态 N // 状态 L 状态区结束 SCRE</pre>

#### 分支控制说明

在很多应用程序中，一个顺序状态流必须分为两个或多个不同的状态流。如下图所示，当控制流分为多个时，所有的输出流必须同时激活。

续表



此例使用由相同的转换条件启用的多条 SCRT 指令，在 SCR 程序中实施分支控制。

### ❖ 相关指令

SCRT、SCRE

## 10.6 SCRT 指令：顺序控制继电器转换

❖ 指令语法 SCRT <N>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<N>	布尔	S

❖ 梯形图  $\text{---}_{(SCRT)}^N$

❖ 梯形图功能说明 指令识别要启用的 SCR 位(下一个要设置的  $n$  位)。当使能位进入线圈时，打开引用  $n$  位，并关闭 LSCR 指令（启用该 SCR 段）的  $n$  位。 $\text{---}_{(SCRT)}^N$  指令执行 SCR 段的转移。当  $n=1$  时，一方面对下一个 SCR 使能位（S 位）置位，以便下一个 SCR 段工作；另一方面又同时对本 SCR 使能位（S 位）复位，以使本 SCR 段停止工作。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<N>	布尔	S	输入值

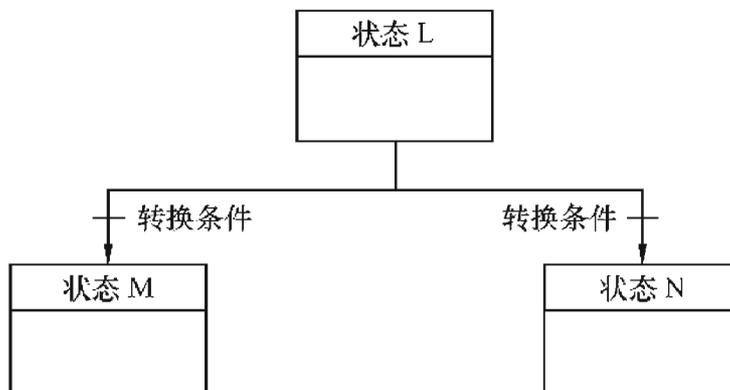
## ❖ 执行范例

## ● 范例 186 ● 使用顺序控制继电器转换指令。

梯形图	语句表指令
	<pre>// 状态 L 控制区开始 LSCR S3.4 // 转换至状态 M LD M2.3 SCRT S3.5 // 转换至状态 N LD I3.3 SCRT S6.5 // 状态 L 的 SCR 区结束 SCRE</pre>

## 转移条件分支控制说明

在有些情况下，一个控制流可能转入多个可能的控制流中某一个。到底进入哪一个，取决于控制流前面的转移条件哪一个首先为真，如下图所示。



此例使用由不同的转换条件启用的多条 SCRT 指令，在 SCR 程序中实施分支控制。

## ❖ 相关指令

LSCR、SCRE

# 10.7 SCRE 指令：顺序控制继电器结束

❖ 指令语法 SCRE

❖ 梯形图 —(SCRE)

❖ 梯形图功能说明 指令标记 SCR 段的结束。—(SCRE) 表示一个 SCR 段的结束。从 LSCR 开始到 SCRE 结束的所有指令组成 SCR 段，

它能否执行取决于 SCR 堆栈的值。SCRE 和下一个 LSCR 之间的指令逻辑与 SCR 堆栈的值无关。

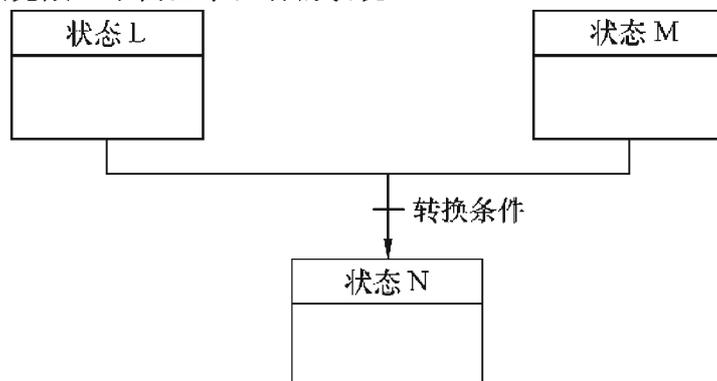
### 执行范例

#### 范例 187 使用顺序控制继电器结束指令。

梯形图	语句表指令
	<pre> // 状态 L 控制区开始 LSCR S3.4 // 转换至状态 L' LD V100.5 SCRT S3.5 // 状态 L 的 SCR 区结束 SCRE // 状态 M 控制区开始 LSCR S6.4 // 转换至状态 M' LD C50 SCRT S6.5 // 状态 M 的 SCR 区结束 SCRE LD S3.5 // 当状态 L' A S6.5 // 和状态 M'均为现用时，则： S S5.0 1 // 启用状态 N R S3.5 1 // 复原状态 L' R S6.5 1 // 复原状态 M' </pre>

#### 合并控制说明

当两个或多个连续状态流必须汇合成一个状态流时，出现一种与分散控制相似的状况。当多个状态流汇合成一条状态流时，则称为汇合。当状态流汇合时，在执行下一个状态之前，所有的输入流必须完成。下图显示汇合的状况：



此例表示当代表 L'和 M'的两个 SCR 位均为真时，可启用状态 N。

## ❖ 相关指令

LSCR、SCRT

# 10.8 CRET 指令：从子程序(SBR)有条件返回

## ❖ 指令语法 CRET

## ❖ 梯形图 —( RET )

❖ 梯形图功能说明 根据该指令前面的逻辑关系，决定是否终止子程序。执行完子程序以后，控制程序回到子程序调用指令的下一条指令。

## ❖ 执行范例

◎ 范例 188 ◎ 使用从子程序 (SBR) 有条件返回指令。

梯 形 图	语句表指令
	<pre>// 主程序 //在首次扫描时，调用子程序 0，执行初始化操作 LD SM0.1 CALL SBR_0 // 子程序 0 // 您可以在最后一个网络之前使用 // 有条件返回离开子程序 LD M14.3 CRET // 如果 M14.3 打开，该网络将被跳过 LD SM0.0 MOVB 10 VB0</pre>

# 10.9 END 指令：主程序 ( OB1 ) 有条件结束

## ❖ 指令语法 END

## ❖ 梯形图 —( END )

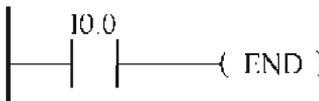
❖ 梯形图功能说明 根据前面的逻辑关系终止用户主程序。可以

在主程序中使用“有条件结束”指令，但不能在子程序或中断例行程序中

使用。

### ❖ 执行范例

● 范例 189 ● 使用主程序有条件结束指令。

梯 形 图	语句表指令
	<pre>// I0.0 打开时，中止当前扫描 LD I0.0 END</pre>

## 10.10 STOP 指令：转换至停止模式

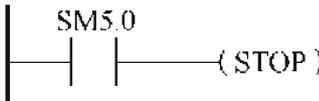
❖ 指令语法 STOP

❖ 梯形图 —(STOP)

❖ 梯形图功能说明 指令能够引起 CPU 方式发生变化，从 RUN 到 STOP，从而可以立即终止程序的执行。如果 STOP 指令在中断程序中执行，那么该中断立即终止，并且忽略所有挂起的中断，继续扫描程序的剩余部分。在本次扫描的最后，CPU 工作状态从 RUN 到 STOP 转变。

### ❖ 执行范例

● 范例 190 ● 使用转换至停止指令。

梯 形 图	语句表指令
	<pre>// 检测到 I/O 错误时，强制转换至 STOP(停止) 模式 LD SM5.0 STOP</pre>

## 10.11 WDR 指令：监视程序复原

❖ 指令语法 WDR

❖ 梯形图 —(WDR)

**❖ 梯形图功能说明** 指令重新触发 S7-200 CPU 的系统监视程序定时器，扩展扫描允许使用的时间，而不会出现看门狗错误。

### ❖ 执行范例

● 范例 191 ● 使用监视程序复原指令。

梯 形 图	语 句 表 指 令
	<pre>//M5.6 打开时，重新触发 CPU 看门狗复原和输出模块看门狗，允许扩展扫描 LD M5.6 WDR      //重新触发 CPU 看门狗复原 BIW QB2 QB2 // 重新触发第一个输出模块的看门狗</pre>

## 10.12 DLED 指令：诊断 LED

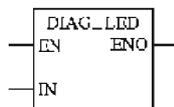
**❖ 指令语法** DLED <IN>

**❖ 指令参数说明**

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD

**❖ 梯形图**



**❖ 梯形图功能说明** 如果输入参数 IN 的数值为零，则诊断 LED 会被设置为不发光。如果输入参数 IN 的数值大于零，则诊断 LED 会被设置为发光（黄色）。

系统块（配置 LED）复选框选项如下。

- (1) SF/ DIAG LED 会在 CPU 中的某个项被强制时发黄光。
- (2) SF/ DIAG LED 会在某模块有 I/O 错误时发黄光。

取消两个复选框的选取，就会让 DIAG\_LED 指令独自控制 SF/ DIAG 黄光。CPU 系统故障（SF）用红光表示。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、 常数、*AC、*VD、*LD	输入值

### ❖ 执行范例

● 范例 192 ● 使用诊断 LED 指令。

梯 形 图	语 句 表 指 令
	<pre>LD I0.0 // I0.0 遇泵错误时开启 A SM0.5 // SM0.5 周期开启 (.5 秒)、关闭 (.5 秒) = M0.0 // M0.1 至 M0.7 必须为零且未被其他网络使用 LD SM0.0 // SM0.0 始终开启, DLED 在每次扫描时执行 DLED MB0 // DIAG LED 在发生泵错误时闪黄光</pre>



## 第 11 章 移位和循环 移位指令

可使用移位指令逐位向左或向右移动输入端 IN 的内容。向左移  $n$  位会将输入 IN 的内容乘以 2 的  $n$  次幂，向右移  $n$  位则会将输入 IN 的内容除以 2 的  $n$  次幂。设计人员为输入参数 N 提供的数值指示要移动的位数。由移位指令移空的位会用零或符号位的信号状态（0 表示正，1 表示负）补上。

### 11.1 SLB 指令：向左移位字节

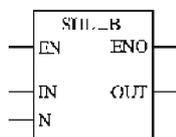
❖ 指令语法 SLB <OUT><N>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

将输入字节数值（IN）根据移位计数（N）向左移动，并将结果载入输出字节（OUT）。移位指令对每个移出位补 0。如果移位数目（N）大于或等于 8，则数值最多被移位 8 次；如果移位数目

大于 0，溢出内存位（SM1.1）采用最后一次移出位的数值；如果移位操作结果为 0，设置 0 内存位（SM1.0）。右移和向左移字节操作不带符号。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、 常数、*AC、*VD、*LD	输入值
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、 常数、*AC、*VD、*LD	移位计数
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、 *LD、*AC	输出值

### ❖ 执行范例

● 范例 193 ● 使用向左移位字节指令。

梯 形 图	语 句 表 指 令
	<p>//触点 I0.0 打开，VB0 中的字节左移 2 位后放入 VB0 中</p> <pre>LD I0.0 SLB VB0 2</pre>

应用

VB0	1100 1011
	向左移 2 位后
VB0	0010 1100

### ❖ 相关指令

SRB

## 11.2 SLW 指令：向左移位字

❖ 指令语法 SLW <OUT><N>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<OUT>	字	VW、IW、QW、MW、SW、SMW、T、C、LW、AC、*VD、*AC、*LD



◆ 梯形图功能说明 将输入字数值 (IN) 向左移动 N 位，并将结果载入输出字 (OUT)。移位指令对每个移出位补 0。如果移位数目 (N) 大于或等于 16，则数值最多被移位 16 次；如果移位数目大于 0，溢出内存位 (SM1.1) 采用最后一次移出位数值；如果移位操作结果为 0，设置 0 内存位 (SM1.0)。当使用带符号的数据类型时，符号位被移位。

#### ◆ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字	VW、IW、QW、MW、SW、SMW、LW、T、C、AIW、AC、常数、*VD、*LD、*AC	输入值
<N>	字节	VW、IW、QW、MW、SW、SMW、T、C、AIW、LW、AC、常数、*VD、*AC、*LD	移位计数
<OUT>	字	VW、IW、QW、MW、SW、SMW、T、C、LW、AC、*VD、*AC、*LD	输出值

#### ◆ 执行范例

● 范例 194 ● 使用向左移位字指令。

梯 形 图	语 句 表 指 令
	//触点 I0.0 打开，VW200 中的字左移 3 位后放入 VW200 中 LD I0.0 SLW VW200 3
应用	
VW200	1100 1011 1110 1101 向左移 3 位后
VW200	0101 1111 0110 1000

## ❖ 相关指令

SRW

## 11.3 SLD 指令：向左移位双字

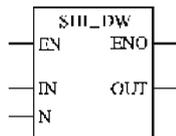
❖ 指令语法 SLD &lt;OUT&gt;&lt;N&gt;

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<OUT>	双字	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*AC、*LD

❖ 梯形图



❖ 梯形图功能说明

将输入双字数值 (IN) 向左移动 N 位，并将结果载入输出双字 (OUT)。移位指令对每个移出位补 0。如果移位数目 (N) 大于或等于 32，则数值最多被移位 32 次；如果移位数目大于 0，溢出内存位 (SM1.1) 采用最后一次移出位数值；如果移位操作结果为 0，设置 0 内存位 (SM1.0)。使用带符号数据类型时，符号位被移位。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	双字	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*AC、*LD	输入值
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	移位计数
<OUT>	字	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC	输出值

## ❖ 执行范例

● 范例 195 ● 使用向左移位双字指令。

梯 形 图	语句表指令
	<pre>//触点 I0.0 打开, VD200 中的双 字左移 4 位后放入 VD200 中 LD I0.0 SLW VD200 4</pre>

应用

VW200	1100 1011 1110 1101 0010 1000 0110 1100
	向左移 4 位后
VW200	1011 1110 1101 0010 1000 0110 1100 0000

## ❖ 相关指令

SRD

# 11.4 SRB 指令：向右移位字节

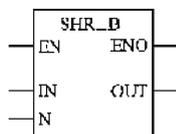
❖ 指令语法 **SRB <OUT><N>**

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

将输入字节数值 (IN) 根据移位计数 (N) 向右移动, 并将结果载入输出字节 (OUT)。移位指令对每个移出位补 0。如果移位数目 (N) 大于或等于 8, 则数值最多被移位 8 次; 如果移位数目

大于 0，溢出内存位（SM1.1）采用最后一次移出位的数值；如果移位操作结果为 0，设置 0 内存位（SM1.0）。右移和向左移字节操作不带符号。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	输入值
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	移位计数
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC	输出值

### ❖ 执行范例

● 范例 196 ● 使用向右移位字节指令。

梯 形 图	语 句 表 指 令
	<p>//触点 I0.0 打开，VB0 中的字节右移 2 位后放入 VB0 中</p> <pre>LD I0.0 SRB VB0 2</pre>

应用

VB0	1100 1011
	向右移 2 位后
VB0	0011 0010

### ❖ 相关指令

SLB

## 11.5 SRW 指令：向右移位字

❖ 指令语法 SRW <OUT><N>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<OUT>	字	VW、IW、QW、MW、SW、SMW、T、C、LW、AC、*VD、*AC、*LD



◆ 梯形图功能说明 将输入字数值 (IN) 向右移动 N 位，并将结果载入输出字 (OUT)。移位指令对每个移出位补 0。如果移位数目 (N) 大于或等于 16，则数值最多被移位 16 次；如果移位数目大于 0，溢出内存位 (SM1.1) 采用最后一次移出位数值；如果移位操作结果为 0，设置 0 内存位 (SM1.0)。当使用带符号的数据类型时，符号位被移位。

◆ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字	VW、IW、QW、MW、SW、SMW、T、C、AIW、LW、AC、常数、*VD、*AC、*LD	输入值
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	移位计数
<OUT>	字	VW、IW、QW、MW、SW、SMW、T、C、LW、AC、*VD、*AC、*LD	输出值

◆ 执行范例

● 范例 197 ● 使用向右移位字指令。

梯 形 图	语 句 表 指 令
	//触点 I0.0 打开，VW200 中的字右移 3 位后放入 VW200 中 LD I0.0 SRW VW200 3

应用

VW200	1100 1011 1110 1101
	向右移 3 位后
VW200	0001 1001 0111 1101

## ❖ 相关指令

SLW

# 11.6 SRD 指令：向右移位双字

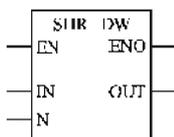
❖ 指令语法 SRD &lt;OUT&gt;&lt;N&gt;

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<OUT>	双字	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*AC、*LD

❖ 梯形图



❖ 梯形图功能说明

将输入双字数值 (IN) 向右移动 N 位，并将结果载入输出双字 (OUT)。移位指令对每个移出位补 0。如果移位数目 (N) 大于或等于 32，则数值最多被移位 32 次；如果移位数目大于 0，溢出内存位 (SM1.1) 采用最后一次移出位数值；如果移位操作结果为 0，设置 0 内存位 (SM1.0)。使用带符号数据类型时，符号位被移位。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	双字	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*AC、*LD	输入值
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	移位计数
<OUT>	字	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*AC、*LD	输出值

## ❖ 执行范例

● 范例 198 ● 使用向右移位双字指令。

梯形图	语句表指令
	<pre>//触点 I0.0 打开，VD200 中的双字右移 4 位后放入 VD200 中 LD I0.0 SLW VD200 4</pre>

应用

VW200	1100 1011 1110 1101 0010 1000 0110 1100
	向左移 4 位后
VW200	0000 1011 1110 1101 0010 1000 0110 1100

## ❖ 相关指令

SLD

# 11.7 RLB 指令：循环左移字节

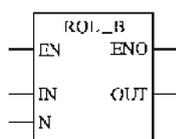
❖ 指令语法 RLB <OUT><N>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

将输入字节数值 (IN) 循环左移 N 位，并将结果载入输出字节 (OUT)。如果所需移位次数大于或等于 8，那么在执行循环移位前，先对取以 8 为底的模，其结果 0~7 为实际移动位数。如果所需移位数为 0，那就不执行循环移位。如果执行循环移位的话，那么溢出

位 (SM1.1) 值就是最近一次循环移动位的值。如果移位次数不是 8 的整数倍, 最后被移出的位就存放到溢出存储器位 (SM1.1)。如果移位操作的结果是 0, 零存储器位 (SM1.0) 就置位。字节循环移位操作是无符号的。

### ❖ 梯形图参数说明

参数的具体说明如下:

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	输入值
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	移位计数
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC	输出值

### ❖ 执行范例

● 范例 199 ● 使用循环左移字节指令。

梯 形 图	语 句 表 指 令
	<pre>//触点 I0.0 打开, VB0 中的字节左移 2 位后放入 VB0 中 LD I0.0 RLB VB0 2</pre>

应用

VB0	1100 1011
	循环左移 2 位后
VB0	0010 1111

### ❖ 相关指令

RRB

## 11.8 RLW 指令：循环左移字

❖ 指令语法 RLW <OUT><N>

❖ 指令参数说明

参数的具体说明如下:

地 址	数 据 类 型	存 储 区
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<OUT>	字	VW、IW、QW、MW、SW、SMW、T、C、LW、AC、*VD、*AC、*LD



◆ 梯形图功能说明 将输入字数值 (IN) 循环左移 N 位，并将结果载入输出字 (OUT)。如果移动位数 (N) 大于或等于 16，在循环执行之前的移动位数 (N) 上执行模数 16 操作，从而使移动位数在 0~15 之间。如果移动位数为 0，则不执行循环移位。如果执行循环移位，移位的最后一位数值被复制至溢出位 (SM1.1)。如果移动位数不是 16 的整倍数，移位出的最后一位数值被复制至溢出内存位 (SM1.1)。如果移位操作结果为 0，设置 0 内存位 (SM1.0)。

◆ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字	VW、IW、QW、MW、SW、SMW、T、C、AIW、LW、AC、常数、*VD、*AC、*LD	输入值
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	移位计数
<OUT>	字	VW、IW、QW、MW、SW、SMW、T、C、LW、AC、*VD、*AC、*LD	输出值

◆ 执行范例

◎ 范例 200 ◎ 使用循环左移字指令。

梯 形 图	语 句 表 指 令
	<pre>//触点 I0.0 打开，VW200 中的字循环左移 3 位后放入 VW200 中 LD I0.0 RLW VW200 3</pre>

续表

梯形图	语句表指令
应用	
VW200	1100 1011 1110 1101 循环左移 3 位后
VW200	0101 1111 0110 1110

## ❖ 相关指令

RRW

## 11.9 RLD 指令：循环左移双字

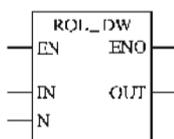
❖ 指令语法 RLD &lt;OUT&gt;&lt;N&gt;

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<OUT>	双字	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*AC、*LD

❖ 梯形图



❖ 梯形图功能说明

将输入双字数值 (IN) 循环左移 N 位，并将结果载入输出双字 (OUT)。如果所需移位次数 N 大于或等于 32，那么在执行循环移位前，先对 N 取以 32 为底的模，其结果 0~31 为实际所移位次数。如果所需移位数为 0，那就不执行循环移位。如果执行循环移位的话，那么溢出位 (SM1.1) 上的值就是最近一次循环移动的值。如果移位次数不是 32 的整数倍，最后被移出的位就存放到溢出存储器位 (SM1.1)。如果移位操作的结果是 0，零存储器位 (SM1.0) 就置位。字循环移位操作是无符号的。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	双字	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*AC、*LD	输入值
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	移位计数
<OUT>	字	W、IW、QW、MW、SW、SMW、LW、T、C、AC、*VD、*LD、*AC	输出值

### ❖ 执行范例

⊙ 范例 201 ⊙ 使用循环左移字指令。

梯 形 图	语 句 表 指 令
	//触点 I0.0 打开, VD200 中的双字左移 4 位后放入 VD200 中 LD I0.0 RLD VD200 4
应用	
VW200	1100 1011 1110 1101 0010 1000 0110 1100
	向左移 4 位后
VW200	0000 1011 1110 1101 0010 1000 0110 1100

### ❖ 相关指令

RRD

## 11.10 RRB 指令：循环右移字节

❖ 指令语法 RRB <OUT><N>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD

续表

地 址	数 据 类 型	存 储 区
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC



### ◆ 梯形图功能说明

将输入字节数值 (IN) 循环右移 N 位, 并将结果载入输出字节 (OUT)。如果所需移位次数大于或等于 8, 那么在执行循环移位前, 先对取以 8 为底的模, 其结果 0~7 为实际移动位数。如果所需移位数为 0, 那就不执行循环移位。如果执行循环移位的话, 那么溢出位 (SM1.1) 值就是最近一次循环移动位的值。如果移位次数不是 8 的整数倍, 最后被移出的位就存放到溢出存储器位 (SM1.1)。如果移位操作的结果是 0, 零存储器位 (SM1.0) 就置位。字节循环移位操作是无符号的。

### ◆ 梯形图参数说明

参数的具体说明如下:

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	输入值
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	移位计数
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC	输出值

### ◆ 执行范例

◎ 范例 202 ◎ 使用循环右移字节指令。

梯 形 图	语 句 表 指 令
	<pre>//触点 I0.0 打开, VB0 中的字节右移 2 位后放入 VB0 中 LD I0.0 RRB VB0 2</pre>

续表

梯形图	语句表指令
应用	
	VB0 1100 1011 循环右移 2 位后
	VB0 1111 0010

## ❖ 相关指令

RLB

# 11.11 RRW 指令：循环右移字

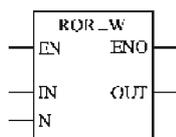
## ❖ 指令语法 RRW &lt;OUT&gt;&lt;N&gt;

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<OUT>	字	VW、IW、QW、MW、SW、SMW、T、C、LW、AC、*VD、*AC、*LD

## ❖ 梯形图



## ❖ 梯形图功能说明

将输入字数值 (IN) 循环右移 N 位，并将结果载入输出字 (OUT)。如果移动位数 (N) 大于或等于 16，在循环执行之前的移动位数 (N) 上执行模数 16 操作，从而使移动位数在 0~15 之间。如果移动位数为 0，则不执行循环移位。如果执行循环移位，移位的最后一位数值被复制至溢出位 (SM1.1)。如果移动位数不是 16 的整倍数，移位出的最后一位数值被复制至溢出内存位 (SM1.1)。如果移位操作结果为 0，设置 0 内存位 (SM1.0)。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字	VW、IW、QW、MW、SW、SMW、T、C、AIW、LW、AC、常数、*VD、*AC、*LD	输入值
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	移位计数
<OUT>	字	VW、IW、QW、MW、SW、SMW、T、C、LW、AC、*VD、*AC、*LD	输出值

### ❖ 执行范例

#### ● 范例 203 ● 使用循环右移字指令。

梯 形 图	语 句 表 指 令
	//触点 I0.0 打开，VW200 中的字循环右移 3 位后放入 VW200 中 LD I0.0 RRW VW200 3

#### 应用

VW200	1100 1011 1110 1101
	循环右移 3 位后
VW200	1011 1001 0111 1101

### ❖ 相关指令

RLW

## 11.12 RLD 指令：循环右移双字

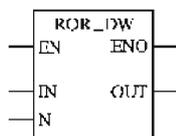
❖ 指令语法 RRD <OUT><N>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<OUT>	双字	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*AC、*LD

### ❖ 梯形图



### ❖ 梯形图功能说明

将输入双字数值 (IN) 循环右移 N 位, 并将结果载入输出双字 (OUT)。如果所需移位次数 N 大于或等于 32, 那么在执行循环移位前, 先对 N 取以 32 为底的模, 其结果 0~31 为实际所移位次数。如果所需移位数为 0, 那就不执行循环移位。如果执行循环移位的话, 那么溢出位 (SM1.1) 上的值就是最近一次循环移动的值。如果移位次数不是 32 的整数倍, 最后被移出的位就存放到溢出存储器位 (SM1.1)。如果移位操作的结果是 0, 零存储器位 (SM1.0) 就置位。字循环移位操作是无符号的。

### ❖ 梯形图参数说明

参数的具体说明如下:

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	双字	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*AC、*LD	输入值
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	移位计数
<OUT>	字	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC	输出值

### ❖ 执行范例

● 范例 204 ● 使用循环右移字指令。

梯 形 图	语 句 表 指 令
	<pre>//触点 10.0 打开, VD200 中的双字右移 4 位后放入 VD200 中 LD 10.0 RRD VD200 4</pre>
应用	
VW200	1100 1011 1110 1101 0010 1000 0110 1100
	循环右移 4 位后
VW200	1100 1100 1011 1110 1101 0010 1000 0110

## ◆ 相关指令

RLD

## 11.13 SHRB 指令：寄存器位移

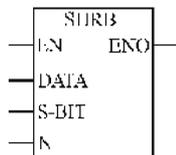
◆ 指令语法 RRD &lt;DATA&gt;&lt;S\_BIT&gt;&lt;N&gt;

## ◆ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<DATA>	布尔	I、Q、M、SM、T、C、V、S、L
<S_BIT>	布尔	I、Q、M、SM、T、C、V、S、L

## ◆ 梯形图



## ◆ 梯形图功能说明

将 DATA 数值移入移位寄存器。S\_BIT 指定移位寄存器的最低位。N 指定移位寄存器的长度和移位方向（移位加 = N，移位减 = -N）。SHRB 指令移出的每个位被放置在溢出内存位（SM1.1）中。该指令由最低位（S\_BIT）和由长度（N）指定的位数定义。

## ◆ 梯形图参数说明

参数的具体说明如下：

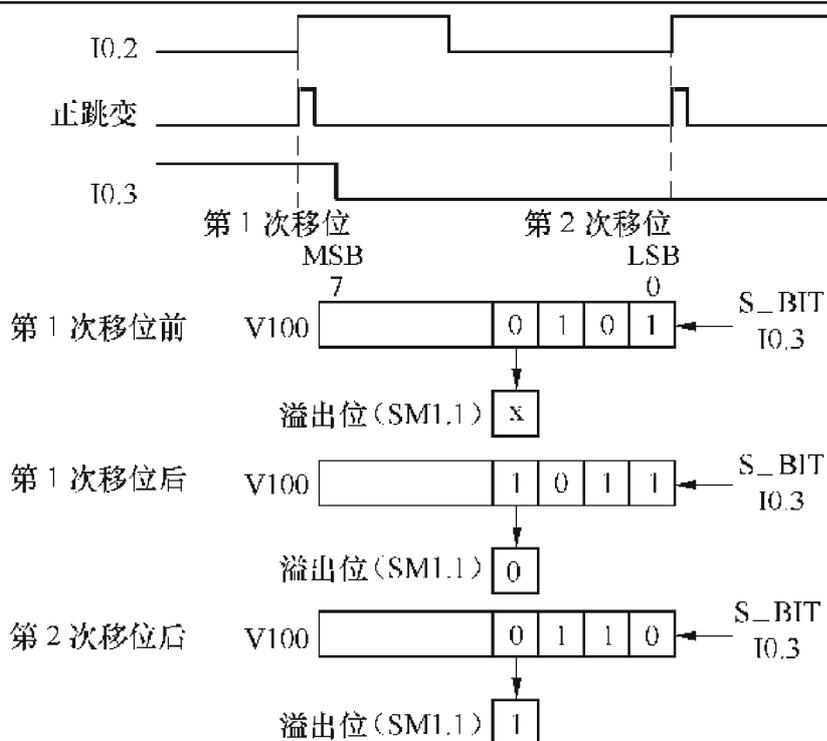
参 数	数据类型	内 存 区 域	说 明
<DATA>	布尔	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	输入值
<S_BIT>	布尔	I、Q、M、SM、T、C、V、S、L	最低位
<N>	字节	I、Q、M、SM、T、C、V、S、L	移位计数

## ◆ 执行范例

○ 范例 205 ○ 使用寄存器位移指令。

梯形图	语句表指令
	<pre>LD I0.2 EU SHRB I0.3 V100.0 +4</pre>

时序图





## 第 12 章 字符串指令

字符串指令可以完成字符串的查找、比较和转换功能，为人机交互界面的编程提供了较大的设计空间。

### 12.1 SLEN 指令：获取字符串长度

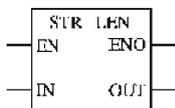
❖ 指令语法 SLEN <IN><OUT>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	字符串	VB、常数字符串、LB、*VD、*LD、*AC
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

指令返回 IN 指定的字符串长度。常数字符串参数最长为 126 个字节。设置 ENO = 0 的错误条件：0006（间接地址）；0091（操作数范围）。

下面的内存图显示了字符串数据类型的格式。字符串的长度可为 0~254 个字符。字符串最长为 255 个字节（254 个字符加长度字节）。

长度	字符 1	字符 2	字符 3	字符 4	...	字符 254
字节 0	字节 1	字节 2	字节 3	字节 4		字节 254

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字符串	VB、常数字符串、LB、*VD、*LD、*AC	输入值
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC	输出值

### ❖ 执行范例

● 范例 206 ● 使用获取字符串长度指令。

梯 形 图	语 句 表 指 令
	<pre>LD I0.0 SCAT "World", VB0 // 将字符串 "World" 附加在字符串 "Hello" 之后 SCPY VB0, VB100 // 将位于 VB0 位置的字符串复制至位于 VB100 位置的一个新字符串 SLEN VB100, AC0 // 获取从 VB100 位置开始的字符串长度</pre>

应用

执行网络程序之前位于VB0的字符串：

VB0	VB1	VB2	VB3	VB4	VB5	VB6
6	H	e	l	l	o	

执行网络程序之后：

VB0	VB1	VB2	VB3	VB4	VB5	VB6	VB7	VB8	VB9	VB10	VB11
11	H	e	l	l	o		W	o	r	l	d

VB100	VB101	VB102	VB103	VB104	VB105	VB106	VB107	VB108	VB109	VB110	VB111
11	H	e	l	l	o		W	o	r	l	d

AC0 = 11

# 12.2

## SCPY 指令：将字符串 1 复制至字符串 2

### ❖ 指令语法

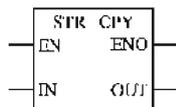
SCPY <IN><OUT>

## ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	字符串	VB、常数字符串、LB、*VD、*LD、*AC
<OUT>	字符串	VB、LB、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

将 IN 指定的字符串复制至 OUT 指定的字符串。常数字符串参数最长为 126 个字节。设置 ENO = 0 的错误条件：0006（间接地址）；0091（操作数范围）。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字符串	VB、常数字符串、LB、*VD、*LD、*AC	输入值
<OUT>	字符串	VB、LB、*VD、*LD、*AC	输出值

## ❖ 执行范例

● 范例 207 ● 使用字符串复制指令。

梯 形 图	语句表指令
	<pre>LD I0.0 SCAT "World", VB0 // 将字符串 "World" 附加在字符串 "Hello" 之后 SCPY VB0, VB100 // 将位于 VB0 位置的字符串复制至位于 VB100 位置的一个新字符串 SLEN VB100, AC0 // 获取从 VB100 位置开始的字符串长度</pre>

续表

梯 形 图						语 句 表 指 令					
应用											
执行网络程序之前位于VB0的字符串:											
VB0	VB1	VB2	VB3	VB4	VB5	VB6					
6	H	e	l	l	o						
执行网络程序之后:											
VB0	VB1	VB2	VB3	VB4	VB5	VB6	VB7	VB8	VB9	VB10	VB11
11	H	e	l	l	o		W	o	r	l	d
VB100	VB101	VB102	VB103	VB104	VB105	VB106	VB107	VB108	VB109	VB110	VB111
11	H	e	l	l	o		W	o	r	l	d
AC0 = 11											

## 12.3 SSCPY 指令：从字符串复制子字符串

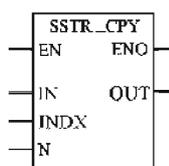
❖ 指令语法 SSCPY <IN><INDX><N><OUT>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	字符串	VB、常数字符串、LB、*VD、*LD、*AC
<INDX><N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<OUT>	字符串	VB、LB、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

将（从索引（INDX）开始）N 指定的具体字符串数目复制至 OUT 指定的字符串。常数字符串参数最长为 126 个字节。设置 ENO = 0 的错误条件：0006（间接地址）；0091（操作数范围）；009B（非法索引：指定起始位置 0 的字符串操作）。

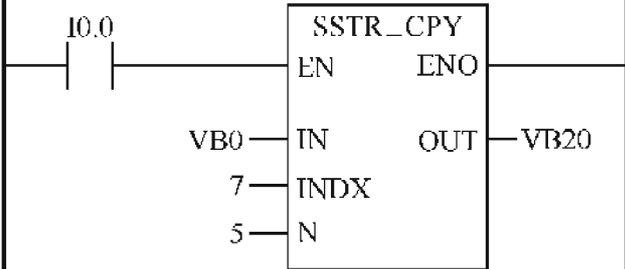
❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字符串	VB、常数字符串、LB、*VD、*LD、*AC	输入值
<INDX><N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	索引/字符串复制数目
<OUT>	字符串	VB、LB、*VD、*LD、*AC	输出值

### 执行范例

#### 范例 208 使用子字符串复制指令。

梯 形 图	语 句 表 指 令
	<pre>LD I0.0 SSCPY VB0,7,5,VB20 // 从第 7 个           字符开始，将 5 个字符复制至位于           VB20 位置的一个新字符串</pre>

#### 应用

执行网络程序之前位于VB0位置的字符串：

VB0	VB1	VB2	VB3	VB4	VB5	VB6	VB7	VB8	VB9	VB10	VB11
11	H	e	l	l	o		W	o	r	l	d

执行网络程序之后位于VB20位置的字符串：

VB20	VB21	VB22	VB23	VB24	VB25
5	W	o	r	l	d

## 12.4 SCAT 指令：字符串连接

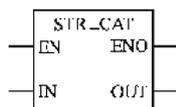
指令语法 SCAT <IN><OUT>

指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	字符串	VB、常数字符串、LB、*VD、*LD、*AC
<OUT>	字符串	VB、LB、*VD、*LD、*AC

梯形图



### ❖ 梯形图功能说明

将 IN 指定的字符串附加至 OUT 指定的字符串之后。常数字符串参数最长为 126 个字节。设置 ENO = 0 的错误条件：0006（间接地址）；0091（操作数范围）。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字符串	VB、常数字符串、LB、*VD、*LD、*AC	输入值
<OUT>	字符串	VB、LB、*VD、*LD、*AC	输出值

### ❖ 执行范例

#### ● 范例 209 ● 使用字符串连接指令。

梯 形 图	语 句 表 指 令
	<pre>LD I0.0 SCAT "World", VB0 // 将字符串 "World" 附加在字符串 "Hello" 之后 SCPY VB0, VB100 // 将位于 VB0 位置的字符串复制至位于 VB100 位置的一个新字符串 SLEN VB100, AC0 // 获取从 VB100 位置开始的字符串长度</pre>

#### 应用

执行网络程序之前位于 VB0 的字符串：

VB0	VB1	VB2	VB3	VB4	VB5	VB6
6	H	e	l	l	o	

执行网络程序之后：

VB0	VB1	VB2	VB3	VB4	VB5	VB6	VB7	VB8	VB9	VB10	VB11
11	H	e	l	l	o		W	o	r	l	d

VB100	VB101	VB102	VB103	VB104	VB105	VB106	VB107	VB108	VB109	VB110	VB111
11	H	e	l	l	o		W	o	r	l	d

AC0 = 11

## 12.5 SFND 指令: 在字符串 1 中查找字符串 2

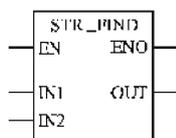
❖ 指令语法 SFND <IN1><IN2><OUT>

❖ 指令参数说明

参数的具体说明如下:

地 址	数 据 类 型	存 储 区
<IN1><IN2>	字符串	VB、常数字符串、LB、*VD、*LD、*AC
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

在字符串 IN1 中搜索首次出现的字符串 IN2。搜索从 OUT 起始位置开始。如果找到一个与字符串 IN2 完全符合的字符系列, 该系列的第一个字符位置被写入 OUT。如果在字符串 IN1 中未找到字符串 IN2, OUT 被设为 0。单个常数字符串最长为 126 个字节, 两个常数字符串综合最长为 240 个字节。设置 ENO = 0 的错误条件: 0006 (间接地址); 0091 (操作数范围); 009B (非法索引: 指定起始位置 0 的字符串操作)。

❖ 梯形图参数说明

参数的具体说明如下:

参 数	数 据 类 型	内 存 区 域	说 明
<IN1><IN2>	字符串	VB、常数字符串、LB、*VD、*LD、*AC	输入值
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC	输出值

❖ 执行范例

- 范例 210 ○ 使用字符串查找指令。

梯形图	语句表指令
	<pre>// 将 AC0 设为 1。 // （AC0 被用作 OUT 参数。） // 在位于 VB0 位置的字符串中搜索字符串 “On” // 从第一个字符位置开始（AC0=1） LD I0.0 MOV B 1 AC0 SFND VB0, "On", AC0</pre>

## 应用

本例将一个存储在 VB0 位置的字符串用作打开或关闭泵的命令。对字符“On”的搜索会在存储于 VB0 的字符串中执行。“在字符串内查找字符串”指令结果存储在 AC0 位置（OUT 参数）。如果结果不是 0，则在命令串（VB12）中查找字符串“打开”。

执行网络程序之前位于 VB0 的字符串：

VB0	VB1	VB2	VB3	VB4	VB5	VB6	VB7	VB8	VB9	VB10	VB11	VB12
12	T	u	i	n		P	u	m	p		0	n

执行网络程序之后的数据数值：

如果在 VB20 中找到字符串：	AC0	11	如果在 VB20 中未找到字符串：	AC0	0
------------------	-----	----	-------------------	-----	---

## 12.6

## CFND 指令：在字符串中查找字符

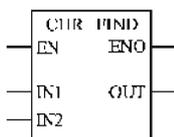
❖ 指令语法 CFND <IN1><IN2><OUT>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1><IN2>	字符串	VB、常数字符串、LB、*VD、*LD、*AC
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

在首次出现的字符串 IN1 中搜索字符串 IN2

中描述的字符集中的任何字符。搜索从起始位置 OUT 开始。如果找到一个相符的字符，该字符位置被写入 OUT。如果未找到相符的字符，OUT 被设为 0。单个常数字符串最长为 126 个字节，两个常数字符串综合最长为 240 个字节。设置 ENO = 0 的错误条件：0006（间接地址）；0091（操作数范围）；009B（非法索引：指定起始位置 0 的字符串操作）。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1><IN2>	字符串	VB、常数字符串、LB、*VD、*LD、*AC	输入值
<OUT>	字节	VB、IB、QB、MB、SB、SMB、LB、*VD、*LD、*AC	输出值

### ❖ 执行范例

#### ● 范例 211 ● 使用字符查找指令。

梯 形 图	语 句 表 指 令
	<pre>// 将 AC0 设为 1(AC0 被用作 OUT 参数，并指向字符串的第一个位置)。在位于 VB0 位置的字符串中查找数字字符。将字符串转换为实数 LD I0.0 MOVB 1, AC0           // 指向带有 AC0 的字符串的起始位置 CFND                  VB0, "0123456789+-", AC0 // 查找位于 VB0 的字符串中的该数字字符 STR  VB0, AC0, VD200 // 将该字符串转换为实数</pre>

#### 应用

本例中，一个存储在 VB0 位置的字符串包含温度。位于 VB20 位置的字符串存储能够识别字符串中温度的所有数字字符（以及+和-）。举例程序在该字符串中找到一个数字的起始位置，然后将该数字字符转换为实数。VD200 存储温度的实数数值。

续表

梯 形 图						语 句 表 指 令					
执行网络程序之前位于VB0的字符串：											
VB0	VB1	VB2	VB3	VB4	VB5	VB6	VB7	VB8	VB9	VB10	VB11
11	T	e	m	p			9	8	.	6	F
执行网络程序之后的数据数值：											
温度起始位置存储在VB0中						温度的实数值					
AC0						VD200					
7						98.6					

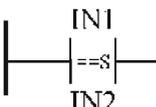
## 12.7 LDS=指令：载入字符串等于

❖ 指令语法 LDS= <IN1><IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字符串	VB、常数字符串、LB、*VD、*LD、*AC
<IN2>	字符串	VB、LB、*VD、*LD、*AC

❖ 梯形图 

❖ 梯形图功能说明 比较两个 ASCII 字符串，如果比较为真，使能位流过比较触点执行载入操作，并将值置于堆栈顶端。单个常数字符串的最大长度为 126 个字节，两个常数字符串的最大组合长度为 242 个字节。

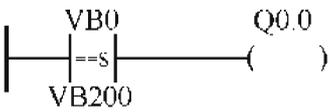
❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字符串	VB、常数字符串、LB、*VD、*LD、*AC	输入值
<IN2>	字符串	VB、LB、*VD、*LD、*AC	输入值

❖ 执行范例

○ 范例 212 ○ 使用载入字符串等于指令。

梯形图	语句表指令
	//当 VB0 中的字符串等于 VB200 中的字符串时， 输出位 Q0.0 被激活 LDS= VB0,VB200 = Q0.0

### ◆ 相关指令

LDS<>

## 12.8

## AS=指令：与运算字符串等于

### ◆ 指令语法

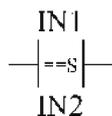
AS= <IN1><IN2>

### ◆ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字符串	VB、常数字符串、LB、*VD、*LD、*AC
<IN2>	字符串	VB、LB、*VD、*LD、*AC

### ◆ 梯形图



### ◆ 梯形图功能说明

比较两个 ASCII 字符串，如果比较为真，使能位流过比较触点，执行 AND（与）操作，并将值置于堆栈顶端。单个常数字符串的最大长度为 126 个字节，两个常数字符串的最大组合长度为 242 个字节。

### ◆ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字符串	VB、常数字符串、LB、*VD、*LD、*AC	输入值
<IN2>	字符串	VB、LB、*VD、*LD、*AC	输入值

### ◆ 执行范例

◎ 范例 213 ◎ 使用与运算字符串等于指令。

梯形图	语句表指令
	<pre>//当 VB0 中的字符串等于 VB200 中的字符串， //并且触点 I0.0 打开时，输出位 Q0.0 被激活 LD I0.0 AS= VB0,VB200 = Q0.0</pre>

### ❖ 相关指令

AS<>

## 12.9 OS=指令：或运算字符串等于

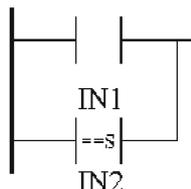
❖ 指令语法 OS= <IN1><IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字符串	VB、常数字符串、LB、*VD、*LD、*AC
<IN2>	字符串	VB、LB、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

比较两个 ASCII 字符串，如果比较为真，使能位流过比较触点，执行 OR（或）操作，并将值置于堆栈顶端。单个常数字符串的最大长度为 126 个字节，两个常数字符串的最大组合长度为 242 个字节。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字符串	VB、常数字符串、LB、*VD、*LD、*AC	输入值
<IN2>	字符串	VB、LB、*VD、*LD、*AC	输入值

## ❖ 执行范例

● 范例 214 ● 使用或运算字符串等于指令。

梯形图	语句表指令
	<pre>//当 VB0 中的字符串等于 VB200 中的字符串， //或者触点 I0.0 打开时，输出位 Q0.0 被激活 LD I0.0 OS= VB0,VB200 = Q0.0</pre>

## ❖ 相关指令

OS<>

## 12.10

## LDS&lt;&gt;指令：载入字符串不等于

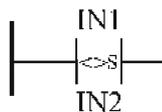
❖ 指令语法 LDS<> <IN1><IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字符串	VB、常数字符串、LB、*VD、*LD、*AC
<IN2>	字符串	VB、LB、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

比较两个 ASCII 字符串 IN1<>IN2，如果比较为真，使能位流过比较触点，执行载入操作，并将值置于堆栈顶端。单个常数字符串的最大长度为 126 个字节，两个常数字符串的最大组合长度为 242 个字节。

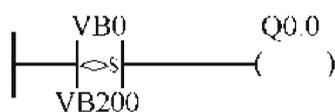
❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字符串	VB、常数字符串、LB、*VD、*LD、*AC	输入值
<IN2>	字符串	VB、LB、*VD、*LD、*AC	输入值

## ❖ 执行范例

● 范例 215 ● 使用载入字符串不等于指令。

梯 形 图	语句表指令
	<pre>//当 VB0 中的字符串不等于 VB200 中的字符串时， 输出位 Q0.0 被激活 LDS&lt;&gt; VB0,VB200 =      Q0.0</pre>

## ❖ 相关指令

LDS=

## 12.11 AS<>指令：与运算字符串不等于

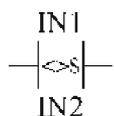
❖ 指令语法 AS<> <IN1><IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字符串	VB、常数字符串、LB、*VD、*LD、*AC
<IN2>	字符串	VB、LB、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

比较两个 ASCII 字符串 IN1<>IN2，如果比较为真，使能位流过比较触点，执行 AND（与）操作，并将值置于堆栈顶端。单个常数字符串的最大长度为 126 个字节，两个常数字符串的最大组合长度为 242 个字节。

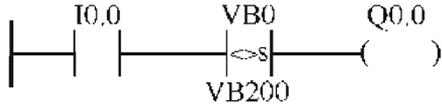
❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字符串	VB、常数字符串、LB、*VD、*LD、*AC	输入值
<IN2>	字符串	VB、LB、*VD、*LD、*AC	输入值

## ❖ 执行范例

● 范例 216 ● 使用与运算字符串不等于指令。

梯 形 图	语句表指令
	<pre>//当 VB0 中的字符串不等于 VB200 中的字符串时， 输出位 Q0.0 被激活 LD  I0.0 AS&lt;&gt; VB0,VB200 =   Q0.0</pre>

## ❖ 相关指令

AS=

## 12.12 OS<>指令：或运算字符串不等于

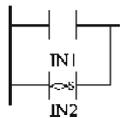
❖ 指令语法 OS<> <IN1><IN2>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN1>	字符串	VB、常数字符串、LB、*VD、*LD、*AC
<IN2>	字符串	VB、LB、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

比较两个 ASCII 字符串 IN1<>IN2，如果比较为真，使能位流过比较触点，执行 OR（或）操作，并将值置于堆栈顶端。单个常数字符串的最大长度为 126 个字节，两个常数字符串的最大组合长度为 242 个字节。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN1>	字符串	VB、常数字符串、LB、*VD、*LD、*AC	输入值
<IN2>	字符串	VB、LB、*VD、*LD、*AC	输入值

## ❖ 执行范例

● 范例 217 ● 使用或运算字符串不等于指令。

梯形图	语句表指令
	<pre>//当 VB0 中的字符串不等于 VB200 中的字符串， //或者 I0.0 打开时，输出位 Q0.0 被激活 LD    I0.0 OS&lt;&gt; VB0,VB200 =     Q0.0</pre>

## ❖ 相关指令

OS=

## 12.13 ITS 指令：整数转换至字符串

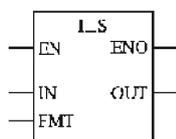
❖ 指令语法 ITS <IN><OUT><FMT>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC
<FMT>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<OUT>	字符串	VB、LB、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

将整数 IN 转换为长度为 8 个字符的 ASCII 字符串。格式 (FMT) 指定小数点右面的转换精度，无论小数点是显示为逗号还是句点。结果字符串写入从 OUT 开始的 9 个连续字节中。

ITS 的格式 (FMT) 操作数定义：

MSB							LSB
7	6	5	4	3	2	1	0
0	0	0	0	c	n	n	n

C 表示逗号 (1) 或小数点 (0), nnn 表示小数点右面的位数。

输出字符串的长度始终为 8 个字符。输出缓冲区中的小数点右面的位数由 nnn 域指定。nnn 域的有效范围是 0~5。将小数点右面的位数指定为 0 会使值不带小数点。当 nnn 值大于 5 时, 输出显示为 8 个 ASCII 空格字符的字符串。c 位指定是使用逗号 (c = 1) 还是使用小数点 (c = 0) 作为整数和小数之间的分隔符。格式最上方的 4 个位必须为 0。

下表显示了几个使用小数点 (c = 0) 和小数点右面 3 位数 (nnn = 011) 格式的值范例。位于 OUT 位置的值是字符串长度。

OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT
		+1	+2	+3	+4	+5	+6	+7	+8
IN=12	8				0	.	0	1	2
IN=123	8				0	.	1	2	3
IN=1234	8				1	.	2	3	4
IN=12345	8			1	2	.	3	4	5

设置 ENO = 0 的错误条件: 0006 (间接地址); 0091 (操作数范围); 非法格式 (nnn > 5)。

### ◆ 梯形图参数说明

参数的具体说明如下:

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC	输入值
<FMT>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	转换精度
<OUT>	字符串	VB、LB、*VD、*LD、*AC	输出值

### ◆ 执行范例

#### ◎ 范例 218 ◎ 使用整数转换至字符串指令。

梯 形 图	语 句 表 指 令
	<pre>//当 I0.0 打开时, 整数 12345 被转换成字符串 "12.345" 放入 VB0 中 LD    I0.0 ITS   12345,VB0,3</pre>

## ❖ 相关指令

STI

## 12.14

## DTS 指令：双整数转换至字符串

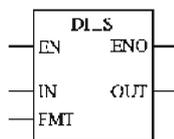
❖ 指令语法 DTS &lt;IN&gt;&lt;OUT&gt;&lt;FMT&gt;

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	双整数	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*LD、*AC
<FMT>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<OUT>	字符串	VB、LB、*VD、*LD、*AC

## ❖ 梯形图



## ❖ 梯形图功能说明

将双整数 IN 转换为长度为 12 个字符的 ASCII 字符串。格式 (FMT) 指定小数点右面的转换精度，无论小数点是显示为逗号还是句点。结果字符串写入从 OUT 开始的 13 个连续字节中。

DTS 的格式 (FMT) 操作数定义：

MSB							LSB
7	6	5	4	3	2	1	0
0	0	0	0	c	n	n	n

C 表示逗号 (1) 或小数点 (0)，nnn 表示小数点右面的位数。

输出字符串的长度始终为 8 个字符。输出缓冲区中的小数点右面的位数由 nnn 域指定。nnn 域的有效范围是 0~5。将小数点右面的位数指定为 0 会使值显示为不带小数点。当 nnn 值大于 5 时，输出显示为 12 个 ASCII 空格字符的字符串。c 位指定是使用逗号 (c = 1) 还是使用小数点 (c = 0) 作为整数和小数之间的分隔符。格式最上方的 4 个位必须为 0。

下表显示了使用小数点 (c=0) 和小数点右面 4 位数 (nnn=100) 格式的值范例。位于 OUT 位置的值是字符串长度。

	OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT
		+1	+2	+3	+4	+5	+6	+7	+8	+9	+10	+11	+12
IN=12	12							0	.	0	0	1	2
IN=1234567	12					1	2	3	.	4	5	6	7

设置 ENO = 0 的错误条件：0006 (间接地址)；0091 (操作数范围)；非法格式 (nnn > 5)。

### ◆ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	双整数	VD、ID、QD、MD、SD、SMD、LD、HC、AC、常数、*VD、*LD、*AC	输入值
<FMT>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	转换精度
<OUT>	字符串	VB、LB、*VD、*LD、*AC	输出值

### ◆ 执行范例

◎ 范例 219 ◎ 使用双整数转换至字符串指令。

梯 形 图	语 句 表 指 令
	<pre>//当 I0.0 打开时，整数 1234567 //被转换成字符串“123.4567”放 //入 VB0 中 LD    I0.0 DTS   1234567,VB0,4</pre>

### ◆ 相关指令

STD

## 12.15

## RTS 指令：实数转换至字符串

### ◆ 指令语法

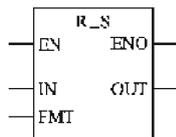
RTS <IN><OUT><FMT>

## ◆ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、常数、*VD、*LD、*AC
<FMT>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<OUT>	字符串	VB、LB、*VD、*LD、*AC

## ◆ 梯形图



## ◆ 梯形图功能说明

将实数值 IN 转换为 ASCII 字符串。格式 (FMT) 指定小数点右面的转换精度，无论小数点是显示为逗号还是句点，亦无论输出字符串的长度是多少。转换结果放置在以 OUT 开始的字符串中。结果字符串长度在格式中指定，可以是 3~15 个字符。

RTS 的格式 (FMT) 操作数定义：

MSB							LSB
7	6	5	4	3	2	1	0
s	s	s	s	c	n	n	n

ssss 表示输出字符串长度，c 表示逗号 (1) 或小数点 (0)，nnn 表示小数点右面的位数。

输出字符串的长度由 ssss 域指定。0、1 或 2 个字节无效。输出缓冲区中的小数点右面的位数由 nnn 域指定。nnn 域的有效范围是 0~5。将小数点右面的位数指定为 0 会使值显示为不带小数点。当 nnn 值大于 5 时或当指定的输出字符串长度太小无法存储转换的值时，输出字符串用 ASCII 空格字符填充。c 位指定是使用逗号 (c = 1) 还是使用小数点 (c = 0) 作为整数和小数之间的分隔符。

下表显示了几个使用小数点 (c = 0) 和小数点右面有一位数 (nnn = 001) 以及输出字符串长度为 6 个字符 (ssss = 0110) 格式的值范例。位于 OUT 位置的值是字符串长度。

	OUT	OUT+1	OUT+2	OUT+3	OUT+4	OUT+5	OUT+6
IN=1234.5	6	1	2	3	4	.	5
IN=-0.0004	6				0	.	0
IN=-3.67526	6			-	3	.	7
IN=1.95	6				2	.	0

设置 ENO = 0 的错误条件：0006（间接地址）；0091（操作数范围）；非法格式（ $nnn > 5$ ； $ssss < 3$ ； $ssss <$  要求的字符数）。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、常数、*VD、*LD、*AC	输入值
<FMT>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	转换精度
<OUT>	字符串	VB、LB、*VD、*LD、*AC	输出值

### ❖ 执行范例

● 范例 220 ● 使用实数转换至字符串指令。

梯 形 图	语 句 表 指 令
	<pre>//当 I0.0 打开时，实数 1234.5 被转换成 字符串“1234.5”放入 VB0 中 LD    I0.0 DTS   1234.5,VB0,2#1100001</pre>

### ❖ 相关指令

STR

## 12.16 STI 指令：字符串转换至整数

❖ 指令语法 STI <IN><INDX><OUT>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	字符串	VB、常数字符串、LB、*VD、*LD、*AC
<INDX>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<OUT>	整数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC



### ◆ 梯形图功能说明

将字符串数值 IN 转换为存储在 OUT 中的整数值，从偏移量 INDX 位置开始。

INDX 值通常设为 1，从字符串的第一个字符开始转换。可将该值设为其他值，在字符串中的不同点开始转换。当输入字符串包含不属于需要转换数字一部分的文本时，可采用此种方法。例如，如果输入字符串是“Temperature: 77.8”，可以将 INDX 设为值 13，跳过字符串开始的“Temperature:”。当达到字符串结束时或当发现第一个无效字符时，转换终止。无效字符是数字（0~9）以外的任何字符。

每当转换产生一个对于输出值过大的整数值时，则设置溢出错误（SM1.1）。例如，如果输入字符串产生一个大于 32 767 或小于 -32 768 的值时，STI 指令设置溢出错误。

如果当输入字符串未包含有效值而无法执行转换时，也会设置溢出错误（SM1.1）。例如，如果输入字符串包含“A123”，转换指令会设置 SM1.1（溢出），输出值保持不变。设置 ENO = 0 的错误条件：0006（间接地址）；0091（操作数范围）；009B（非法指数：指定起始位置值 0 的字符串操作）。

### ◆ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字符串	VB、常数字符串、LB、*VD、*LD、*AC	输入值
<INDX>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	偏移地址
<OUT>	整数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC	输出值

## ❖ 执行范例

◎ 范例 221 ◎ 使用字符串转换至整数指令。

梯形图	语句表指令
	<pre>LD I0.0 STI VB0 7 VW100</pre>

应用

## 输入字符串示例

V <sub>B0</sub>	V <sub>B1</sub>	V <sub>B2</sub>	V <sub>B3</sub>	V <sub>B4</sub>	V <sub>B5</sub>	V <sub>B6</sub>	V <sub>B7</sub>	V <sub>B8</sub>	V <sub>B9</sub>	V <sub>B10</sub>	V <sub>B11</sub>
11	T	e	m	p			9	8	.	6	F

## 执行网络之后:

VW100 (整数) = 98

## ❖ 相关指令

ITS

## 12.17 STD 指令：字符串转换至长整数

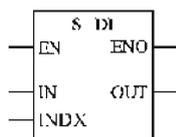
❖ 指令语法 **STD <IN><INDX><OUT>**

❖ 指令参数说明

参数的具体说明如下:

地 址	数 据 类 型	存 储 区
<IN>	字符串	VB、常数字符串、LB、*VD、*LD、*AC
<INDX>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<OUT>	双整数	VD、ID、QD、MD、SMD、SD、LD、AC、*VD、*LD、*AC

## ❖ 梯形图



### ❖ 梯形图功能说明

将字符串值 IN 转换为存储在 OUT 中的长整数值，从偏移量 INDX 位置开始。

每当转换产生一个对于输出值过大的整数值时，则设置溢出错误 (SM1.1)。

如果当输入字符串未包含有效值而无法执行转换时，也会设置溢出错误 (SM1.1)。例如，如果输入字符串包含“A123”，转换指令会设置 SM1.1 (溢出)，输出值保持不变。设置 ENO = 0 的错误条件：0006 (间接地址)；0091 (操作数范围)；009B (非法指数：指定起始位置值 0 的字符串操作)。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字符串	VB、常数字符串、LB、*VD、*LD、*AC	输入值
<INDX>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	偏移地址
<OUT>	双整数	VD、ID、QD、MD、SMD、SD、LD、AC、*VD、*LD、*AC	输出值

### ❖ 执行范例

● 范例 222 ● 使用字符串转换至长整数指令。

梯 形 图	语 句 表 指 令
	<pre>LD I0.0 STD VB0 7 VD200</pre>

应用

#### 输入字符串示例

VB0	VB1	VB2	VB3	VB4	VB5	VB6	VB7	VB8	VB9	VB10	VB11
1	1	1	1	1			9	8	.	6	F

#### 执行网络之后：

VD200 (双整数) = 98

### ❖ 相关指令

DTS

# 12.18 STR 指令：字符串转换至实数

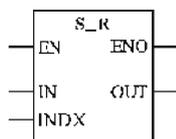
❖ 指令语法 STR <IN><INDX><OUT>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	字符串	VB、常数字符串、LB、*VD、*LD、*AC
<INDX>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<OUT>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

将字符串值 IN 转换为存储在 OUT 中的实数值，从偏移量 INDX 位置开始。

“子字符串至实数”指令不会使用实数的科学计数法或指数形式转换字符串。该指令不会产生溢出错误 (SM1.1)，但只会将字符串转换为直至指数的实数，然后终止转换。例如，字符串“1.234E6”将转换为实数值 1.234，而不产生错误信息。

达到字符串结束时或当发现第一个无效字符时，转换终止。无效字符是数字 (0~9) 以外的任何字符。

每当转换产生一个对于输出值过大的整数值时，则设置溢出错误 (SM1.1)。如果当输入字符串未包含有效值而无法执行转换时，也会设置溢出错误 (SM1.1)。例如，如果输入字符串包含“A123”，转换指令会设置 SM1.1 (溢出)，输出值保持不变。设置 ENO = 0 的错误条件：0006(间接地址)；0091(操作数范围)；009B (非法指数：指定起始位置值 0 的字符串操作)。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	字符串	VB、常数字符串、LB、*VD、*LD、*AC	输入值
<INDX>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	偏移地址
<OUT>	实数	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*LD、*AC	输出值

### ❖ 执行范例

◎ 范例 223 ◎ 使用字符串转换至实数指令。

梯 形 图	语 句 表 指 令
	<pre>LD I0.0 STR VB0 7 VD300</pre>

应用

输入字符串单例

VB0	VB1	VB2	VB3	VB4	VB5	VB6	VB7	VB8	VB9	VB10	VB11
11	T	e	m	p			9	8	.	6	F

执行网络之后：

VD300 (实数) = 98.6

### ❖ 相关指令

RTS



## 第 13 章 表格指令

表格指令直接对 PLC 内存进行操作，一方面提高了程序运行速度，另一方面为大量的数据处理提供了完善的支持。

### 13.1 FILL 指令：内存填充

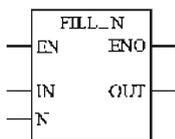
❖ 指令语法 FILL <IN><OUT><N>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC
<N>	字节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD
<OUT>	整数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

用包含在地址 IN 中的字值写入 N 个连续字，从地址 OUT 开始。N 的范围是 1~255。设置 ENO = 0 的错误条件：0006

(间接地址)；0091（操作数范围）。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	整 数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC	输入值
<N>	字 节	VB、IB、QB、MB、SB、SMB、LB、AC、常数、*AC、*VD、*LD	填充数目
<OUT>	整 数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC	输出值

### ❖ 执行范例

⊙ 范例 224 ⊙ 使用内存填充指令。

梯 形 图	语 句 表 指 令															
	<pre>LD I2.1 FILL +0 VW200 10</pre>															
应用																
<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;">IN</td> <td style="text-align: center;">VW200</td> <td style="text-align: center;">VW202</td> <td style="text-align: center;">...</td> <td style="text-align: center;">VW218</td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">...</td> <td style="text-align: center;">0</td> </tr> <tr> <td style="text-align: center;">填充</td> <td></td> <td></td> <td></td> <td></td> </tr> </table>	IN	VW200	VW202	...	VW218	0	0	0	...	0	填充					
IN	VW200	VW202	...	VW218												
0	0	0	...	0												
填充																

## 13.2 ATT 指令：增加至表格

❖ 指令语法 ATT <DATA><TABLE>

❖ 指令参数说明

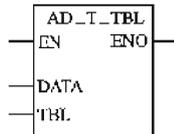
参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<DATA>	整 数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC

续表

地 址	数 据 类 型	存 储 区
<TBL>	字	VW、IW、QW、MW、SW、SMW、T、C、LW、AC、*VD、*AC、*LD

### 梯形图



### 梯形图功能说明

向表格 (TBL) 中加入字值 (DATA)。表格中的第一个数值是表格的最大长度 (TL)。第二个数值是条目计数 (EC)，指定表格中的条目数。新数据被增加至表格中的最后一个条目之后。每次向表格中增加新数据后，条目计数加 1。表格最多可包含 100 个条目，不包括指定最大条目数和实际条目数的参数。设置 ENO = 0 的错误条件：0006 (间接地址)；0091 (操作数范围)。

### 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<DATA>	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC	字值
<TBL>	字	VW、IW、QW、MW、SW、SMW、T、C、LW、AC、*VD、*AC、*LD	表格

### 执行范例

● 范例 225 ● 使用增加至表格指令。

梯 形 图	语 句 表 指 令
	<pre>// 载入最大表格长度 LD SM0.1 MOVW +6 VW200 //填充表格 LD I0.0 ATT VW100 VW200</pre>

续表

梯形图		语句表指令	
应用			
在执行 ATT 之前		在执行 ATT 之后	
VW100	1234	VW200	0006 TL(最大条目数)
VW200	0006	VW202	0002 EC(条目计数)
VW202	0002	VW204	5431 d0 数据 0
VW204	5431	VW206	8942 d1 数据 1
VW206	8942	VW208	1234 d2 数据 2
VW208	XXXX	VW210	XXXX
VW210	XXXX	VW212	XXXX
VW212	XXXX	VW214	XXXX
VW214	XXXX		

## 13.3

## FND?TBL 指令：表格查找

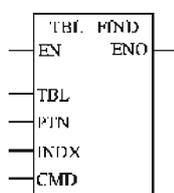
❖ 指令语法 FND? &lt;TBL&gt;&lt;PATRN&gt;&lt;INDX&gt;

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<TBL>	字	VW、IW、QW、MW、SW、SMW、T、C、LW、AC、*VD、*AC、*LD
<PTN>	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC
<INDX>	字	VW、IW、QW、MW、SW、SMW、T、C、LW、AC、*VD、*AC、*LD

❖ 梯形图



❖ 梯形图功能说明

在表格（TBL）中搜索与某些标准相符的数据。“表格查找”指令搜索表，从 INDX 指定的表格条目开始，寻找与 CMD（梯形图中定义的参数）定义的搜索标准相匹配的数据数值（PTN）。命令

参数 (CMD) 被指定一个 1~4 的数值, 分别代表 =、<>、<、>。如果找到匹配条目, 则 INDX 指向表格中的匹配条目。欲查找下一个匹配条目, 再次激活“表格查找”指令之前必须在 INDX 上加 1。如果未找到匹配条目, INDX 的数值等于条目计数。一个表格最多可有 100 个条目, 数据项目 (搜索区域) 从 0 排号至最大值 99。设置 ENO = 0 的错误条件: 0006 (间接地址); 0091 (操作数范围)。

### ❖ 梯形图参数说明

参数的具体说明如下:

参 数	数 据 类 型	内 存 区 域	说 明
<TBL>	字	VW、IW、QW、MW、SW、SMW、T、C、LW、AC、*VD、*AC、*LD	表格
<PTN>	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC	搜索数据
<INDX>	字	VW、IW、QW、MW、SW、SMW、T、C、LW、AC、*VD、*AC、*LD	索引
<CMD>	字节	常数	等式条件

### ❖ 执行范例

● 范例 226 ● 使用表格查找指令。

梯 形 图	语 句 表 指 令
	<pre>LD I2.1 FND= VW202 16#3130 AC1</pre>

应用

当 I2.1 打开时, 在表格中搜索一个等于 3130HEX 的数值。

VW202	0006	EC (entry count)
VW204	3133	d0 (data.0)
VW206	4142	d1 (data.1)
VW208	3130	d2 (data.2)
VW210	3030	d3 (data.3)
VW212	3130	d4 (data.4)
VW214	4541	d5 (data.5)

# 13.4 LIFO 指令：后入先出

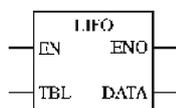
❖ 指令语法 LIFO <TABLE><DATA>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<TBL>	字	VW、IW、QW、MW、SW、SMW、T、C、LW、AC、*VD、*AC、*LD
<DATA>	整数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

将表格中的最新（或最后）一个条目移至输出内存地址，方法是移除表格（TBL）中的最后一个条目，并将数值移至 DATA 指定的位置。每次执行指令时，表格中的条目数减 1。设置 ENO = 0 的错误条件：0006（间接地址）；0091（操作数范围）。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<TBL>	字	VW、IW、QW、MW、SW、SMW、T、C、LW、AC、*VD、*AC、*LD	表格
<DATA>	整数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC	内存变量

❖ 执行范例

⊙ 范例 227 ⊙ 使用后入先出指令。

梯 形 图	语 句 表 指 令
	<pre>LD I0.1 LIFO VW200 VW300</pre>

续表

梯形图		语句表指令	
应用			
执行 LIFO 之前		VW300	1234
VW200	0006	TL(最大条目数)	
VW202	0003	EC(条目计数)	
VW204	5431	d0(数据0)	
VW206	8942	d1(数据1)	
VW208	1234	d2(数据2)	
VW210	XXXX		
VW212	XXXX		
VW214	XXXX		
执行 LIFO 之后		VW300	1234
VW200	0006	TL(最大条目数)	
VW202	0002	EC(条目计数)	
VW204	5431	d0(数据0)	
VW206	8942	d1(数据1)	
VW208	XXXX		
VW210	XXXX		
VW212	XXXX		
VW214	XXXX		

### ◆ 相关指令

FIFO

## 13.5 FIFO 指令：先入先出

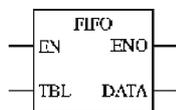
◆ 指令语法 `FIFO <TABLE><DATA>`

◆ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<TBL>	字	VW、IW、QW、MW、SW、SMW、T、C、LW、AC、*VD、*AC、*LD
<DATA>	整数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC

◆ 梯形图



◆ 梯形图功能说明

通过移除表格 (TBL) 中的第一个条目，并将数值移至 DATA 指定位置的方法，移动表格中的最早 (或第一个) 条目。表格中的所有其他条目均向上移动一个位置。每次执行指令时，表格中的条目数减 1。设置 ENO = 0 的错误条件：0006 (间接地址)；0091 (操作数范围)。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<TBL>	字	VW、IW、QW、MW、SW、SMW、T、C、LW、AC、*VD、*AC、*LD	表格
<DATA>	整数	VW、IW、QW、MW、SW、SMW、LW、AQW、T、C、AC、*VD、*LD、*AC	内存变量

### ❖ 执行范例

◎ 范例 228 ◎ 使用先入先出指令。

梯 形 图	语 句 表 指 令
	<pre>LD I4.1 FIFO VW200 VW400</pre>

应用

执行 LIFO 之前

VW200	0006	T1.(最大条目数)
VW202	0003	EC(条目计数)
VW204	5431	d0(数据 0)
VW206	8942	d1(数据 1)
VW208	1234	d2(数据 2)
VW210	XXXX	
VW212	XXXX	
VW214	XXXX	

VW400 5431 执行 FIFO 之后

VW200	0006	T1.(最大条目数)
VW202	0002	EC(条目计数)
VW204	8942	d0(数据 0)
VW206	1234	d1(数据 1)
VW208	XXXX	
VW210	XXXX	
VW212	XXXX	
VW214	XXXX	

### ❖ 相关指令

LIFO



## 第 14 章 定时器指令

定时器指令在 PLC 编程中是非常重要的，尤其在实时控制的工业现场自动线的程序中，定时器指令可以完成工步、工序的时间流程控制功能。

### 14.1 TON 指令：接通延时定时器

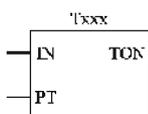
❖ 指令语法 TON <T><PT>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<Txxx>	字	常数(T0-T255)
<PT>	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

启用输入为“打开”时，开始计时。当前值（Txxx）大于或等于预设时间（PT）时，定时器位为“打开”。启用输入为“关闭”时，接通延时定时器当前值被清除。达到预设值后，定时器仍继续计时，达到最大值 32 767 时，停止计时。

TON 定时器有 3 种分辨率。分辨率由下表所示的定时器号码决定。每一个当前值都是时间基准的倍数。例如，10ms 定时器中的计数 50 表示 500ms。

定时器类型	分辨率	最大值	定时器号码
TON	1ms	32.767s	T32、T96
	10ms	327.67s	T33-T36、T97-T100
	100ms	3276.7s	T37-T63、T101-T255

### ❖ 梯形图参数说明

参数的具体说明如下：

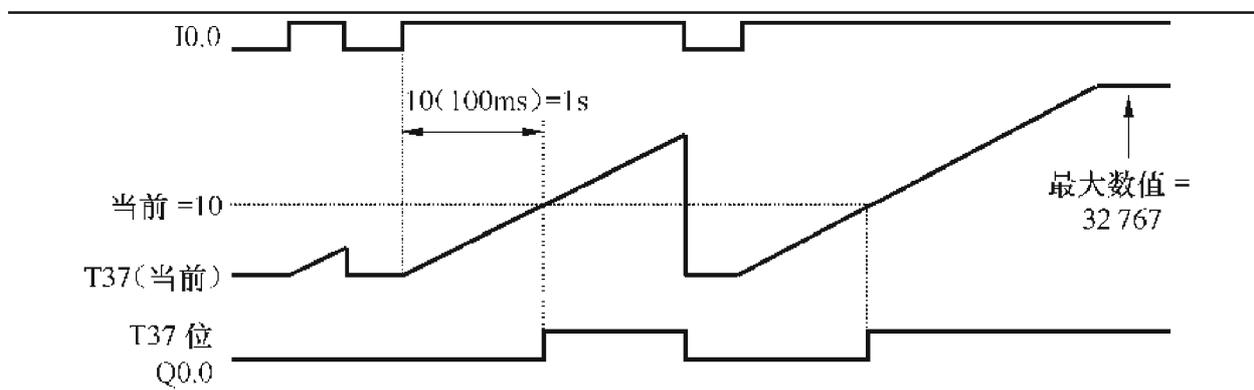
参数	数据类型	内存区域	说明
<Txxx>	字	常数 (T0-T255)	当前值
<IN>	布尔	使能位	定时器触发
<PT>	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC	预设时间

### ❖ 执行范例

● 范例 229 ● 使用接通延时定时器指令。

梯形图	语句表指令
	<pre>// 在(10)100ms 或 1s 之后，100ms 定时器 T37 超时，I0.0 打开=T37 被启用，I0.0 关闭=禁止和复原 T37 LD I0.0 TON T37 +10 // T37 位由定时器 T37 控制 LD T37 = Q0.0</pre>

时序图



### ❖ 相关指令

TOF

## 14.2

## TONR 指令：掉电保护性接通延时定时器

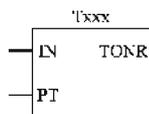
❖ 指令语法 TONR &lt;T&gt;&lt;PT&gt;

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<Txxx>	字	常数(T0-T255)
<PT>	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC

❖ 梯形图



❖ 梯形图功能说明

输入为“打开”时，开始计时。当前值（Txxx）大于或等于预设时间（PT）时，计时位为“打开”。当输入为“关闭”时，保持保留性延迟定时器当前值。可使用保留性接通延时定时器为多个输入“打开”阶段累计时间。使用“复原”指令（R）清除保留性延迟定时器的当前值。达到预设值后，定时器继续计时，达到最大值 32 767 时，停止计时。

TONR 定时器有 3 种分辨率。分辨率由下表所示的定时器号码决定。每一个当前值都是时间基准的倍数。例如，10ms 定时器中的计数 50 表示 500ms。

定时器类型	分 辨 率	最 大 值	定 时 器 号 码
TONR	1ms	32.767s	T0、T64
	10ms	327.67s	T1-T4、T65-T68
	100ms	3276.7s	T5-T31、T69-T95

❖ 梯形图参数说明

参数的具体说明如下：

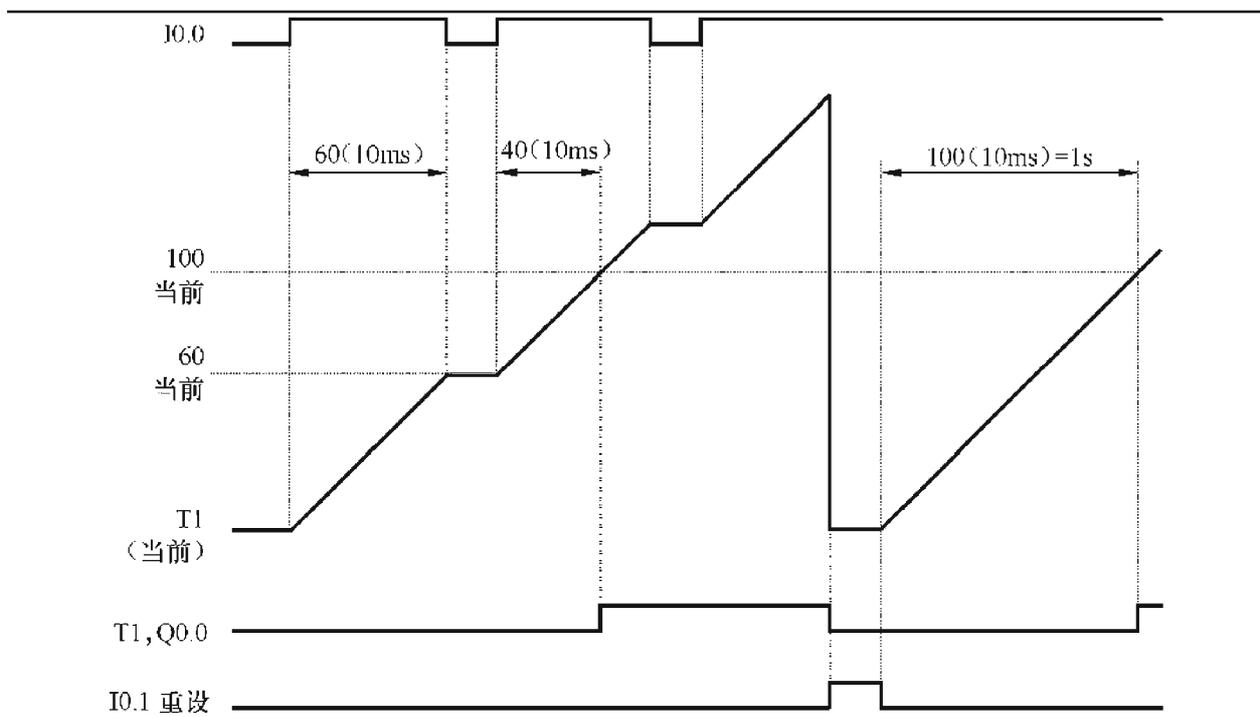
参 数	数 据 类 型	内 存 区 域	说 明
<Txxx>	字	常数(T0-T255)	当前值
<IN>	布尔	使能位	定时器触发
<PT>	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC	预设时间

## ❖ 执行范例

## ● 范例 230 ● 使用有记忆接通延时定时器指令。

梯形图	语句表指令
	<pre>// 10ms TONR 定时器在 PT= (100×10ms) // 或 1s 时超时 LD I0.0 TONR T1 +100 // T1 位由定时器 T1 控制，在定时器总共累 // 计 1s 后，打开 Q0.0 LD T1 = Q0.0 // TONR 定时器必须由带有 T 地址的复原指 // 令复原,当 I0.1 打开时，复原定时器 T1 (当 // 前和位) LD I0.1 R T1 1</pre>

时序图



## 14.3

## TOF 指令：断开延时定时器

## ❖ 指令语法

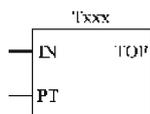
TOF &lt;T&gt;&lt;PT&gt;

## ◆ 指令参数说明

参数的具体说明如下:

地 址	数 据 类 型	存 储 区
<Txxx>	字	常数(T0-T255)
<PT>	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC

## ◆ 梯形图



## ◆ 梯形图功能说明

用于在输入关闭后,延迟固定的一段时间再关闭输出。启用输入打开时,定时器位立即打开,当前值被设为 0。输入关闭时,定时器继续计时,直到消逝的时间达到预设时间。达到预设值后,定时器位关闭,当前值停止计时。如果输入关闭的时间短于预设数值,则定时器位仍保持在打开状态。TOF 指令必须遇到从“打开”至“关闭”的转换才开始计时。如果 TOF 定时器位于 SCR 区域内部,而且 SCR 区域处于非现用状态,则当前值被设为 0,计时器位被关闭,而且当前值不计。

TON 定时器有 3 种分辨率。分辨率由下表所示的定时器号码决定。每一个当前值都是时间基准的倍数。例如,10ms 定时器中的计数 50 表示 500ms。

定时器类型	分 辨 率	最 大 值	定 时 器 号 码
TOF	1ms	32.767s	T32、T96
	10ms	327.67s	T33-T36、T97-T100
	100ms	3276.7s	T37-T63、T101-T255

## ◆ 梯形图参数说明

参数的具体说明如下:

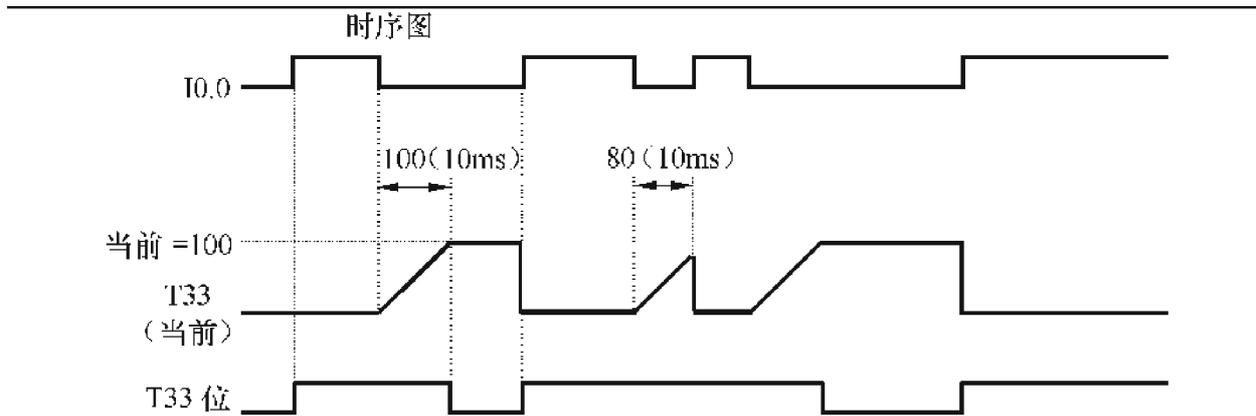
参 数	数 据 类 型	内 存 区 域	说 明
<Txxx>	字	常数(T0-T255)	当前值
<IN>	布尔	使能位	定时器触发
<PT>	整数	VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、*VD、*LD、*AC	预设时间

## ◆ 执行范例

● 范例 231 ● 使用断开延时定时器指令。

梯形图	语句表指令
	<pre>// 在 (100×10ms) 或 1s 之后，10ms 定时器 T33 超时，I0.0 打开至关闭=T33 被启用，I0.0 关闭至打开=禁止和复原 T33 LD I0.0 TOF T33 +100 // 定时器 T33 通过定时器触点 T33 控制 Q0.0 LD T33 = Q0.0</pre>

时序图



## ❖ 相关指令

TON

## 14.4 BITIM 指令：开始间隔时间

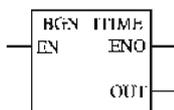
❖ 指令语法 BITIM &lt;OUT&gt;

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<OUT>	双字	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*AC、*LD

❖ 梯形图



❖ 梯形图功能说明

读取内置 1ms 计数器的当前值，并将该值存

储于 OUT。双字毫秒值的最大计时间隔为 2 的 32 次方，即 49.7 日。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<OUT>	双字	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*AC、*LD	输出值

### ❖ 执行范例

◎ 范例 232 ◎ 使用开始间隔时间指令。

梯 形 图	语 句 表 指 令
	<pre>// 捕获开启 Q0.0 的时间 LD Q0.0 EU BITIM VD0  // 计算 Q0.0 处于开启状态的持续时间 LD Q0.0 CITIM VD0, VD4</pre>

### ❖ 相关指令

CITIM

## 14.5 CITIM 指令：计算间隔时间

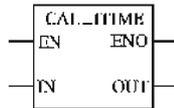
❖ 指令语法 CITIM <IN><OUT>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<IN>	双字	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*AC、*LD
<OUT>	双字	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*AC、*LD

## ❖ 梯形图



## ❖ 梯形图功能说明

计算当前时间与 IN 所提供时间的时差，将该时差存储于 OUT。双字毫秒值的最大计时间隔为 2 的 32 次方，即 49.7 日。取决于 BGN\_ITIME 指令的执行时间，CAL\_ITIME 指令将自动处理发生在最大间隔内的 1ms 定时器翻转。

## ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<IN>	双字	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*AC、*LD	输入值
<OUT>	双字	VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*AC、*LD	输出值

## ❖ 执行范例

● 范例 233 ● 使用计算间隔时间指令。

梯 形 图	语 句 表 指 令
	<pre>// 捕获开启 Q0.0 的时间 LD Q0.0 EU BITIM VD0 // 计算 Q0.0 处于开启状态的持续时间 LD Q0.0 CITIM VD0, VD4</pre>

## ❖ 相关指令

BITIM



## 第 15 章 特殊指令

特殊指令包含有网络通信、脉冲输出和 PID 控制指令，这些指令的使用要求设计人员具备有相当的专业背景和理论知识，而且都是在复杂系统设计时使用。

### 15.1 网络读/网络写

❖ 指令语法

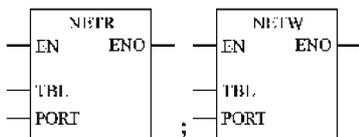
```
NETR <TABLE><PORT>
NETW <TABLE><PORT>
```

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<TBL>	字节	VB、MB、*VD、*LD、*AC
<PORT>	字节	常数(0 用于 CPU 221/222/224; 0 或 1 用于 CPU 226/226XM)

❖ 梯形图



❖ 梯形图功能说明

网络读取 (NETR) 指令开始一项通信操作，通过指定的端口 (PORT) 根据表格 (TBL) 定义从远程设备收集数据。

网络写入 (NETW) 指令开始一项通信操作，通过指定的端口 (PORT) 根据表格 (TBL) 定义向远程设备写入数据。

NETR 指令可从远程站最多读取 16 字节信息, NETW 指令可向远程站最多写入 16 字节信息。

### ❖ 梯形图参数说明

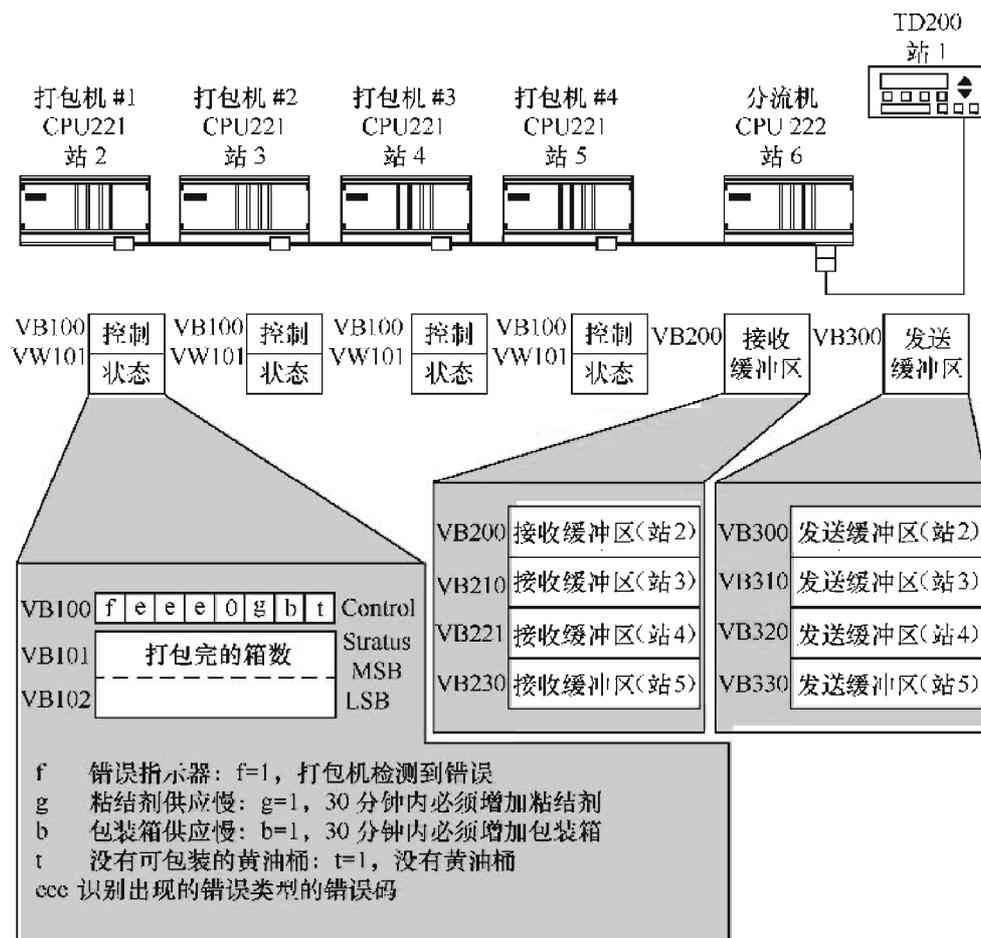
参数的具体说明如下:

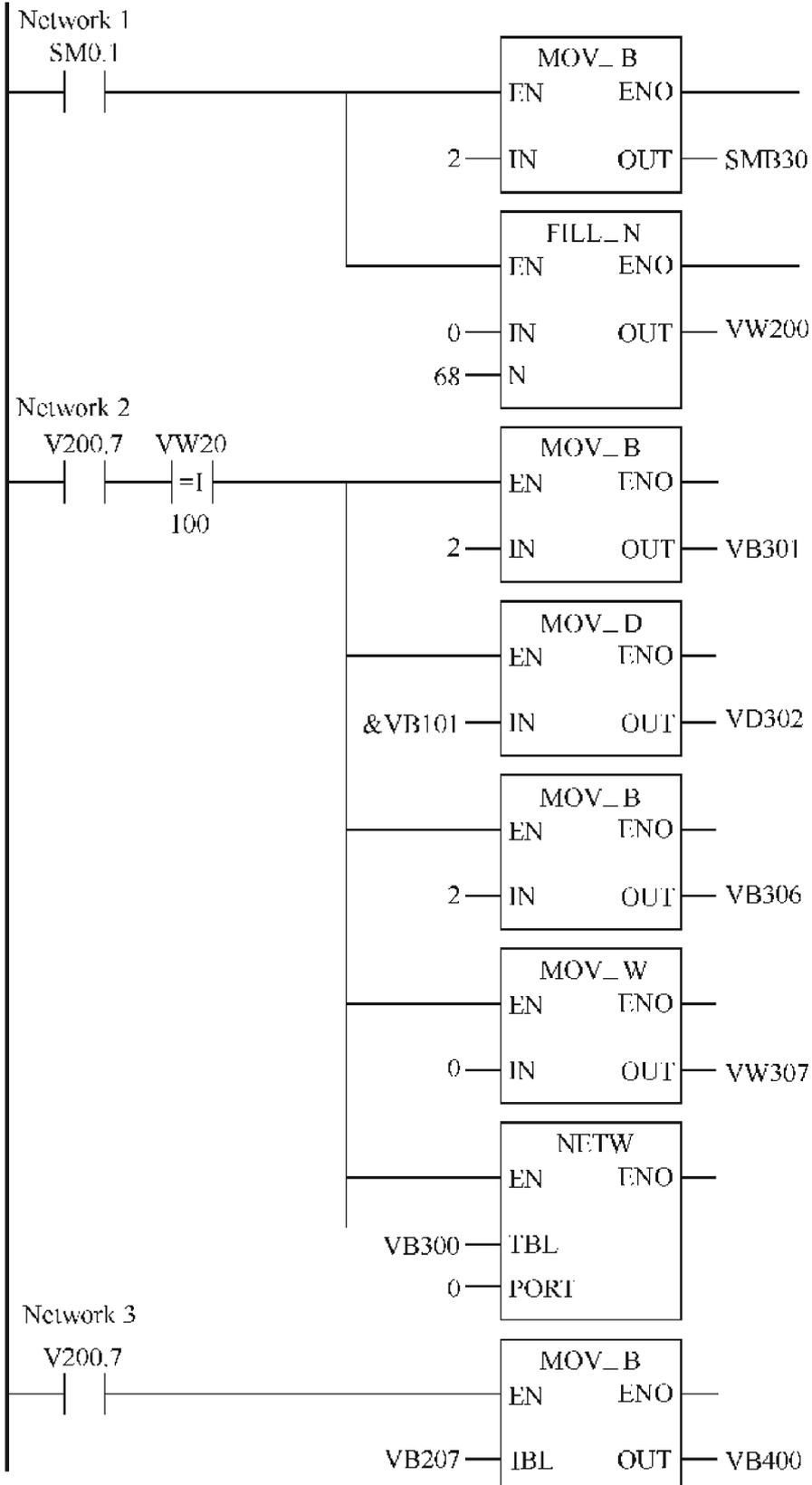
参 数	数 据 类 型	内 存 区 域	说 明
<TBL>	字节	VB、MB、*VD、*LD、*AC	表
<PORT>	字节	常数 (0 用于 CPU 221/222/224; 0 或 1 用于 CPU 226 / 226XM)	端口

### ❖ 执行范例

#### ● 范例 234 ● 使用网络读/网络写指令。

本示例中, 考虑一条生产线正在灌装黄油桶并将其送到 4 台包装机(打包机)中的一台上。打包机把 8 个黄油桶包装到一个纸板箱中。一个分流机控制着黄油桶流向各个打包机。4 个 CPU221 模块用于控制打包机, 一个 CPU222 模块安装了 TD200 操作器接口, 被用来控制分流机。下图给出了该网络配置。



梯 形 图	语句表指令
 <p>Network 1</p> <p>SM0.1</p> <p>MOV_B</p> <p>2 — IN    2 — OUT — SMB30</p> <p>FILL_N</p> <p>0 — IN    68 — N    0 — OUT — VW200</p> <p>Network 2</p> <p>V200.7    VW20 = 100</p> <p>MOV_B</p> <p>2 — IN    2 — OUT — VB301</p> <p>MOV_D</p> <p>&amp;VB101 — IN    0 — OUT — VD302</p> <p>MOV_B</p> <p>2 — IN    2 — OUT — VB306</p> <p>MOV_W</p> <p>0 — IN    0 — OUT — VW307</p> <p>NETW</p> <p>VB300 — TBL    0 — PORT</p> <p>Network 3</p> <p>V200.7</p> <p>MOV_B</p> <p>VB207 — IBL    0 — OUT — VB400</p>	<pre> LD SM0.1 // 首次扫描时, MOV_B 2 SMB30 // 启用 PPI 主模式, FILL +0 VW200 68 // 并清除所有的接收和传输缓冲区 LD V200.7 // 当“NETR 完成”位被设置 AW= VW208 +100 // 且 100 种情况被组装, MOV_B 2 VB301 // 载入情况分组#1 的站址 MOVD &amp;VB101 VD302 // 将指针载入远程站中的数据 MOV_B 2 VB306 // 载入需要传输的数据长度 MOVW +0 VW307 // 载入需要传输的数据 NETW VB300 0 // 复原由情况分组#1 组装的情况数目 LD V200.7 // 当“NETR 完成”位被设置, MOV_B VB207 VB400 // 保存来自情况分组#1 的控制数据。 LDN SM0.1 // 如果不是首次扫描, AN V200.6 // 并没有错误, AN V200.5 </pre>

续表

梯 形 图	语句表指令
<p>Network 4</p>	<pre> MOV_B 2 VB201 // 载入情况分组#1 的 站址  MOVD   &amp;VB100 VD202 // 将指针载 入远程站中的数据  MOV_B 3 VB206 // 载入将要接收的数据 长度  NETR VB200 0 // 读 取情况分组#1 中的 控制和状态数据 </pre>

## 15.2 发送/接收

### ❖ 指令语法

XMT <TABLE><PORT>

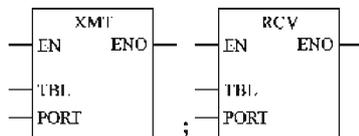
RCV <TABLE><PORT>

### ❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<TBL>	字节	VB、MB、*VD、*LD、*AC
<PORT>	字节	常数(0用于CPU 221/222/224; 0或1用于CPU 226/226XM)

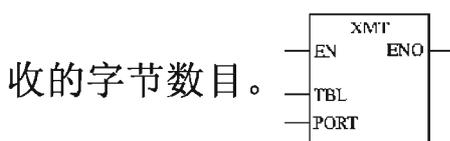
### ❖ 梯形图



### ❖ 梯形图功能说明

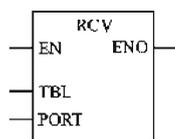
发送 (XMT) 指令在自由端口模式中使用，通过通信端口传送数据。

接收 (RCV) 指令开始或终止“接收信息”服务。必须指定一个开始条件和一个结束条件，“接收”方才能操作。通过指定端口 (PORT) 接收的信息存储在数据缓冲区 (TBL) 中。数据缓冲区中的第一个条目指定接收的字节数目。



在自由端口模式中使用，通过通信端口传送数

据。设置  $ENO = 0$  的错误条件：0006 (间接地址)；0009 (端口 0 中同时执行 XMT/RCV)；000B (端口 1 中同时执行 XMT/RCV)。



通过指定端口 (PORT) 接收的信息存储在数据缓冲区

(TBL) 中。数据缓冲区中的第一个条目指定接收的字节数目。设置  $ENO = 0$  的错误条件：0006 (间接地址)；0009 (端口 0 中同时执行 XMT/RCV)；000B (端口 1 中同时执行 XMT/RCV)。

### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<TBL>	字节	VB、MB、*VD、*LD、*AC	表
<PORT>	字节	常数 (0 用于 CPU 221/222/224；0 或 1 用于 CPU 226 / 226XM)	端口

### ❖ 执行范例

#### ● 范例 235 ● 使用发送/接收指令。

本程序展示了接收和发送的使用，它将接收一串字符，直到接收到回车符。信息又发回到发送方。

梯 形 图	语 句 表 指 令
	<pre> NETWORK 1 // 主程序 LD SM0.1 // 首次扫描时, MOVB 16#09 SMB30 // 初始化自由端口: // - 选择 9600 波特 // - 选择 8 个数据位 // - 选择无校验 MOVB 16#B0 SMB87 // 初始化 RCV 信息控制字节: // - RCV 被启用 // - 检测到信息字符结束 // - 将空闲行条件检测为 // 信息开始条件  MOVB 16#0A SMB89 // 将信息字符结束设为 hex 0A (换行符) MOVW +5 SMW90 // 将空闲行超时设为 5ms MOVB 100 SMB94 // 将最大字符数设为 100 ATCH INT_0 23 // 将中断附加在接收完成事件上 ATCH INT_2 9 // 将中断 2 附加在传送完成事件上 ENI // 启用用户中断 RCV VB100 0 // 为端口 0 在 VB100 位置启用带缓冲区的接收方 </pre>

续表

梯形图	语句表指令
	<pre> NETWORK 1 // 中断 0 // 接收完成中断例行程序 LDB= SMB86 16#20 // 如果接收状态显示接收结束字符, MOVB 10 SMB34 // 则附加一个 10ms 计时器, ATCH INT_1 10 // 触发传送  CRETI // 并返回  NOT RCV VB100 0 // 如果因任何其他原因接收完成, // 则开始新的接收  NETWORK 1 // 中断 1 // 10ms 计时器中断 LD SM0.0 DTCH 10 // 分离计时器中断 XMT VB100 0 // 将信息送回端口 0 中的用户 </pre>

## 15.3 脉冲输出

❖ 指令语法 PLS <Q>

❖ 指令参数说明

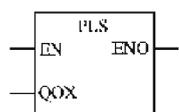
参数的具体说明如下:

地 址	数 据 类 型	存 储 区
<Q>	字	常数 (0 或 1)



❖ 梯形图功能说明 指令被用于控制在高速输入 (Q0.0 和 Q0.1) 中提供的“脉冲串输出”(PTO)和“脉宽调制”(PWM)功能。

PTO 提供方波 (50%占空比) 输出、配备周期和脉冲数用户控制功能。PWM 提供连续性变量占空比输出、配备周期和脉宽用户控制功能。脉冲输出范围为 Q0.0 至 Q0.1。



检测为脉冲输出 (Q0.0 或 Q0.1) 设置的特殊存储器位，然后激活由特殊存储器位定义的脉冲操作。

#### ❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<Q>	字	常数 (0 或 1)	输出位

#### ❖ 执行范例

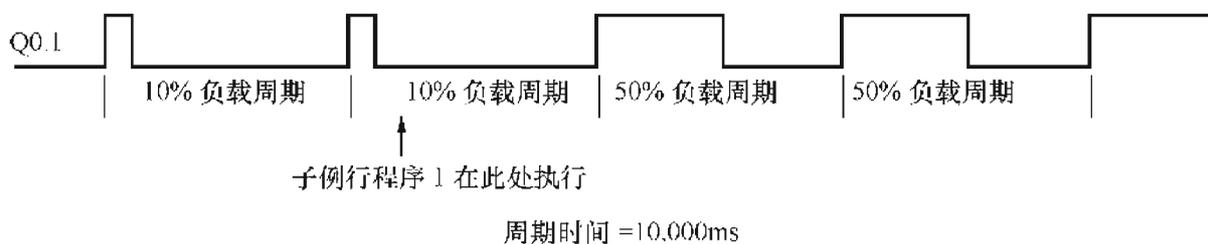
● 范例 236 ● 使用脉冲输出指令。

梯 形 图	语 句 表 指 令
	<pre> NETWORK 1 // 用于 PWM 范例 的主程序 // 首次扫描时，将映像寄存器位 设为低，并调用 SBR_0 LD SM0.1 R Q0.1 1 CALL SBR_0  NETWORK 2 // 设置程序中其他 位置的 M0.0，将脉宽更改为 50% 占空比 LD M0.0 EU CALL SBR_1 </pre>

续表

梯 形 图	语句表指令
	<pre> NETWORK 1 // 子程序 0 开始 LD SM0.0 MOVB 16#DB SMB77 // 设置控制字节  // - 选择 PWM 操作 // - 选择毫秒递增和同步更新 // - 设置脉宽和周期值 // - 启用 PWM 功能 MOVW +10000 SMW78 // 将周期设为 10 000ms MOVW +1000 SMW80 // 将脉宽设为 1 000ms  PLS 1 // 激活 PWM 操作： PLS1=&gt;Q0.1  MOVB 16#DA SMB77 // 重新载入控制字节，用于随后的脉宽改动 </pre>
	<pre> NETWORK 1 // 子程序 1 开始 LD SM0.0 MOVW +5000 SMW80 // 将脉宽设为 5 000ms PLS 1 // 断言脉宽改动 </pre>

脉冲宽度调制 (PWM) 时序图



# 15.4 PID 指令

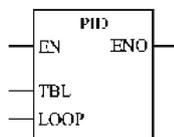
❖ 指令语法 PID <TABLE><LOOP>

❖ 指令参数说明

参数的具体说明如下：

地 址	数 据 类 型	存 储 区
<TBL>	字节	VB
<LOOP>	字节	常数 (0~7)

❖ 梯形图



❖ 梯形图功能说明

指令根据表格 (TBL) 中的输入和配置信息对引用 LOOP 执行 PID 回路计算。提供 PID 回路指令 (成比例、积分、微分回路) 进行 PID 计算。逻辑堆栈 (TOS) 顶值必须是“打开” (使能位) 状态, 才能启用 PID 计算。本指令有两个操作数: 表示回路表起始地址的 TBL 地址和 0~7 常数的“回路”号。

❖ 梯形图参数说明

参数的具体说明如下：

参 数	数 据 类 型	内 存 区 域	说 明
<TBL>	字节	VB	表起始地址
<LOOP>	字节	常数 (0~7)	回路号

❖ 执行范例

◎ 范例 237 ◎ 使用 PID 指令。

在本例中, 有一水箱需要维持一定的水位, 该水箱里的水以变化的速度流出。这就需要有一个水泵以不同的速度给水箱供水, 以维持水位不变, 这样才能使水箱不断水。本系统的给定值是水箱满水位的 75% 时的水位, 过程变量由漂浮在水面的水位测量仪给出。输出值是进水泵的速度, 可以

从允许最大值的 0% 变到 100%。

给定值可以预先设定后直接输入到回路表中，过程变量值是来自水位表的单极性模拟量，回路输出值也是一个单极性模拟量，用来控制进水泵速度。这两个模拟量的范围是 0.0~1.0，分辨率为 1/32000（标准化）。在本系统中，只使用比例和积分控制，其回路增益和时间常数可以通过工程计算初步确定。但还需要进一步调整以达到最优控制效果。初步确定的增益和时间常数为：增益为 0.25，采样时间为 0.1s，积分时间是 30ms。

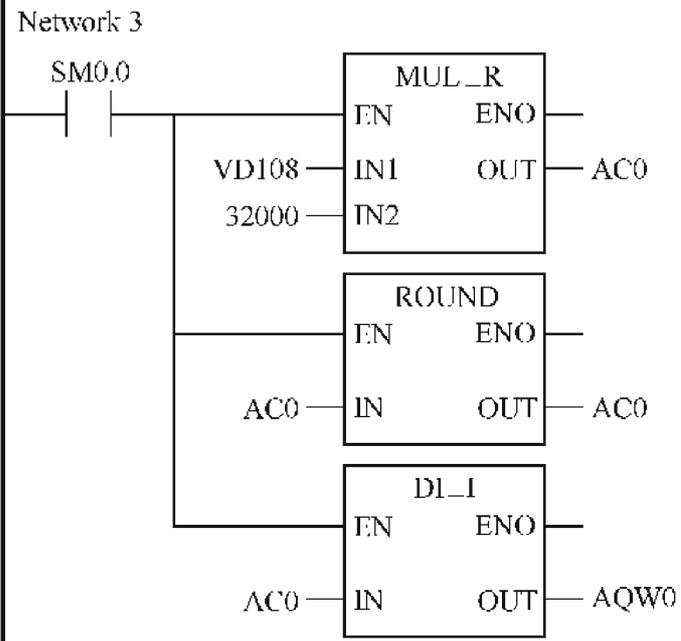
系统启动时，关闭出水口，用手动控制进水泵速度，使水位达到满水位的 75%，然后打开出水口，同时水泵控制从手动方式切换到自动方式。这种切换由一个输入的数字量控制，I0.0 位控制手动到自动的切换：0 代表手动，1 代表自动。当工作在手动控制方式下，可以把水泵速度（0.0~1.0 之间的实数）写到 VD108（VD10 是回路表中保存输出的寄存器）。

梯 形 图	语句表指令
<p>Network 1 SM0.1 —   — [ SBRO ]           EN</p> <p>Network 2 SM0.0 —   — [ MOV_R ]                   EN   TNO   —                   IN   OUT — VD104                   0.75</p> <p>                  [ MOV_R ]                   EN   TNO   —                   IN   OUT — VD112                   0.25</p> <p>                  [ MOV_R ]                   EN   ENO   —                   IN   OUT — VD116                   0.10</p> <p>                  [ MOV_R ]                   EN   ENO   —                   IN   OUT — VD120                   30.0</p>	<pre> Network 1 LD SM0.1 //在首次扫描调用初始化子程序 0 CALL 0 Network 1 LD SM0.0 MOV_R 0.75, VD104 //装入设定值 75% MOVR 0.25, VD112 //装入回路增益=0.25 MOVR 0.10, VD116 //装入采样时间=0.1s MOVR 30.0, VD120 //装入积分时间=30min </pre>

续表

梯形图	语句表指令
	<pre> MOV_R 0.0, VD124 //关闭微分作用 MOV_B 100, SMB34 //设定定时中断 0 发生的时间 //间隔为 100 ms AT_C 0, 10 //设置定时中断以执行 PID 计算 ENI //允许中断 //子程序 0 结束 </pre>
	<pre> NETWORK 1 //把 PV 转换成一个标准的实数 //PV 是一个单极 LD SM0.0 ITD AIW0, AC0 //把单极性模拟量存入累加器 DTR AC0, AC0 //把 32 位整数变换成实数 /R 32000.0, AC0 //标准化累加器中的值 MOV_R AC0, VD100 //把标准化 PV 值存 TABLE 表  NETWORK 2 //在自动方式下执行 PID 控制 LD I0.0 //一旦进入自动方式就执行 PID PID VB100, 0 </pre>

续表

梯形图	语句表指令
<p>Network 3</p> 	<pre> NETWORK 3 //把 Mn 转换成 16 位整数 Mn 为 单极性且非负 LD SM0.0 MOVR VD108, AC0 //把输出值 传送到累加器 *R 32000.0, AC0 //累加器中为刻 度 ROUND AC0, AC0 //把实数转换 为 32 位整数 DTI AC0, AQW0 //把 16 位整数 写到模拟输出 </pre>