

WECON

PLC 编程手册

(LX3V系列) Programming manual



0	LDI	X000
1	OUT	Y000
2	OUT	T0 K100
5	LD	T0
6	OUT	Y001
7	LDP	X001
9	OUT	M0
10	LDF	X002
12	OUT	M1
13	END	



福州富昌维控电子科技有限公司
FUZHOU FUCHANG WECON E AND T LTD

内 容 提 要

这是一本全面介绍 Wecon PLC Editor 基本功能及实际运用的书。本书完全针对零基础读者而开发，是入门级读者快速而全面掌握维控 PLC 和 Wecon PLC Editor 的必备参考书。

本书从维控 PLC 的基本产品、Wecon PLC Editor 的基本概念和基本操作入手，结合大量的案例和图形解析，全面而深入地阐述了 Wecon PLC Editor 软件的使用，以及 PLC 程序的编写，向读者展示了如何运用 Wecon PLC Editor 软件编写程序，灵活的运用于自动化控制设备。

本书非常适用于作为初、中级读者的入门及提高参考书，尤其是零基础读者。同时，本书所有解释权归福州富昌维控电子科技有限公司所有。

安全注意事项

在进行微型可编程控制的安装、运转、保养检修之前，请务必熟读此手册和其他相关手册，确保正确使用。请在熟练掌握操作方法、安全信息以及全部注意事项之后再进行使用。

(1) 设计注意事项

外部电源发生异常、可编程控制器发生故障时，为使整个系统安全运行，请务必在可编程控制器的外部设置安全电路。

①务必在可编程控制器的外部电路中设置紧急制动电路、保护电路、正反转电路等相反操作的互锁电路和防止机器损坏的定位上限、下限的互锁电路等。

②可编程控制器 CPU 通过自诊断检测出 WDT 错误等异常情况时，全部输出被关断。另外当可编程控制器 CPU 不能检测输入输出控制部分等的异常情况时，不能控制输出。这时为使机器能安全运转，请设计外部电路和机构。

③由于输出单元的继电器、晶体管故障，会无法控制输出为 ON 或 OFF 的状态。为使机器能安全运行，对于与重大事故相关的输出信号，请设计外部电路和机构。

(2) 安装注意事项

①请在手册中记载的一般规格环境中使用。

请勿在下列场所使用：有灰尘、油烟、导电性尘埃、腐蚀性气体、可燃性气体的场所；暴露于高温、结露、风雨的场所；有振动、冲击的场所。电击、火灾、误操作也会造成产品损坏。

②在进行螺丝孔加工和接线时，不要使铁屑或电线头落入可编程控制器的通风窗口内。可能引起火灾、故障、误操作。

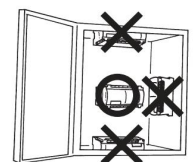
③请把连接电缆、显示模块准确插入规定插口中。接触不良有可能引起误操作。

●为防止温度上升，切勿在底部、顶部及竖直方向安装。

请务必按右图所示在壁面上水平安装。

●主机和其他设备或构造物之间请留出 50mm 以上空间。

尽量避开高压线、高压设备、动力设备。



附记

(3) 布线注意事项

可编程控制器的信号输入和输出线不能在同一电缆上通过。

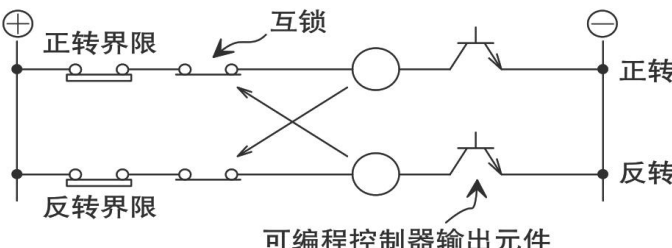
另外，不能将信号输入线和输出线与其他动力线、输出线在同一管道中通过，不能捆扎在一起。

若按上述注意事项执行，输入输出布线即使长达 50~100m，也几乎没有噪音问题。但一般为安全起见，布线长应在 20m 以内。

●必须在外部电源全部切断时进行安装、接线等操作。否则会引起触电或产品损坏。

●在安装、布线等工作结束后，通电运行前，必须先装上端子盖板，以免触电。

附记

 危险		正反转接触器同时合上十分危险，像这样的负载，除了可编程控器内部程序设定互锁以外，在可编程控制器外部也必须设置如上图所示的互锁。
 注意	空端子不要与外部接线，否则会引起产品损坏。	

- 请按照手册中记载的内容对专用接线端进行 AC 电源的接线。
如果把 AC 电源接入直流输入输出端子或直流电源端子，会烧坏可编程控制器。
- 请不要从外部电源对基本单元的 24+端供电，对空端子□请勿从外部布线，否则会损坏产品。
请把基本单元的接地端子按第三种方式接地。但请不要和强电系统共地。
- 电源出现不满 10ms 的瞬时断电，可编程控制器仍会继续工作。
- 长时间停电或电压偏低时，可编程控制器会停止工作，输出变为 OFF，但是一旦电源恢复供电，会自动地重新开始运转。

(4) 启动、保养注意事项

危险
●请不要在通电时触摸端子，否则可能引起触电、误操作。 ●请在电源关闭后进行端子的清扫和拆卸，在通电时执行有可能引起触电。 ●请在熟读手册、充分确认安全后，再进行机器运转中的程序变更、强制输出、RUN/STOP 等操作。 操作错误会损坏机器，引起事故。
注意
●请不要进行分解和改造，否则会引起故障、误动作、火灾。 ※关于修理事宜请与福州富昌维控电子科技有限公司联系。 ●请在电源关闭之后，进行扩展电缆等连接电缆的装卸工作，否则会引起故障和误动作。

(5) 维护检修

- 定期检查：可编程控制器是否装有寿命较短的消耗品。
- 继电器输出型，如果输出继电器异常高频度工作或驱动大容量负载时，必须注意其对使用寿命的影响。
- 和其他设备一起检查，请注意以下要点：
 - 有无由于其他发热体或直射阳光，导致机内温度异常升高。
 - 有无粉尘或导电性尘埃侵入机内。
 - 有无接线和端子松动及其他异常。

目 录

Wecon PLC Editor 编程软件概述	1
1.维控系列 PLC 介绍	2
1.1 PLC 主机构成	2
1.2 扩展模块构成	3
1.3 BD 板构成	3
1.4 远程模块 LCM	4
1.5 外形尺寸	4
2.产品规格	5
2.1 一般规格	5
2.2 电气规格	5
2.3 输入规格	6
2.4 输出规格	6
2.5 产品命名规则	7
3.各种软元件说明	8
3.1 输入继电器 X	9
3.2 输出继电器 Y	9
3.3 辅助继电器 M	9
3.4 状态继电器 S	11
3.5 定时器 T	12
3.6 计数器 C	14
3.7 高速计数器	15
3.8 数据寄存器 D	17
3.9 指针 P · I	18
3.10 常数 K · H · E	19
3.11 特殊软元件一览表	20
4.指令一览表	28
4.1 以指令分类为序	28
4.2 以字母顺序为序	35
5.指令详细说明	37
5.1 基本顺控指令说明	37
LD、LDI、LDP、LDF、OUT 指令	37
AND、ANI、ANDP、ANDF 指令	38
OR、ORI、ORP、ORF 指令	38
ANB、ORB 指令	39
INV 指令	39
MC、MCR 指令	40
MPS、MRD、MPP 指令	41
PLS、PLF 指令	42
SET、RST 指令	42
5.2 功能指令说明	43

5.2.1 程序流指令	43
CJ 指令	43
CALL 指令	44
EI、DI 指令	45
WDT 指令	47
FOR、NEXT 指令	48
5.2.2 传送与比较指令	49
CMP 指令	49
ZCP 指令	50
MOV 指令	51
SMOV 指令	52
CML 指令	53
BMOV 指令	54
FMOV 指令	54
XCH 指令	55
BCD 指令	56
BIN 指令	56
5.2.3 数据处理指令	57
ZRST 指令	57
DECO 指令	58
ENCO 指令	59
SUM 指令	60
BON 指令	60
MEAN 指令	61
ANS 指令	61
ANR 指令	62
SQR 指令	62
FLT 指令	63
SWAP 指令	64
5.2.4 时钟运算	65
TCMP 指令	65
TZCP 指令	66
TADD 指令	67
TSUB 指令	68
TRD 指令	69
TWR 指令	70
HOUR 指令	71
5.2.5 四则运算	72
ADD 指令	72
SUB 指令	73
MUL 指令	74
DIV 指令	75

INC 指令	76
DEC 指令	76
WAND 指令	77
WOR 指令	77
WXOR 指令	78
NEG 指令	79
5.2.6 高速处理指令	80
REF 指令	80
REFF 指令	81
MTR 指令	82
DHSCR 指令	83
DHSCS 指令	84
DHSZ 指令	86
SPD 指令	87
PLSY 指令	88
PWM 指令	89
PWM 千分比模式	89
PLSR 指令	90
PLSR2 指令	92
PTO 指令	100
5.2.7 循环移位指令	102
ROL 指令	102
ROR 指令	103
RCL 指令	104
RCR 指令	105
SFTL 指令	106
SFTR 指令	107
WSFL 指令	108
WSFR 指令	109
SFRD 指令	110
SFWR 指令	111
5.2.8 外部设备 IO 指令	112
TKY 指令	112
HKY 指令	113
DSW 指令	114
SEGD 指令	115
SEGL 指令	116
ARWS 指令	117
ASC 指令	118
PR 指令	119
FROM 指令	120
TO 指令	121

GRY 指令	122
GBIN 指令	122
5.2.9 电子凸轮指令	123
DECAM 指令	123
DEGEAR 指令	126
ECAMTBX 指令	129
5.2.10 方便指令	130
IST 指令	130
ABSD 指令	134
INCD 指令	135
TTMR 指令	136
SER 指令	137
STMR 指令	138
ALT 指令	139
RAMP 指令	140
ROTC 指令	141
SORT 指令	142
5.2.11 定位指令	144
DABS 指令	144
ZRN 指令	145
PLSV 指令	147
DRVI 指令	148
DRVA 指令	150
DVIT 指令	152
SCL 指令	154
G90G01 指令	157
G91G01 指令	159
G90G02 指令	161
G91G02 指令	163
G90G03 指令	165
G91G03 指令	167
5.2.12 外部设备 SER 指令	169
RS 指令	169
RS2 指令	172
RSLIST 指令	174
CPAVL 指令(串口通讯)	181
CPAVL 指令(以太网通讯)	183
PRUN 指令	186
ASCI 指令	187
HEX 指令	189
CCD 指令	191
CCPID 指令	192

PID 指令	200
5.2.13 浮点运算	202
DECMP 指令	202
DEZCP 指令	203
DEBCD 指令	203
DEBIN 指令	204
DEADD 指令	204
DESUB 指令	205
DEMUL 指令	206
DEDIV 指令	207
DESQR 指令	208
INT 指令	208
DSIN 指令	209
DCOS 指令	210
DTAN 指令	210
DASIN 指令	211
DACOS 指令	212
DATAN 指令	213
DSINH 指令	214
DCOSH 指令	214
DTANH 指令	215
DDEG 指令	215
DRAD 指令	216
DEXP 指令	217
DLOG10 指令	217
DLOGE 指令	218
5.2.14 触点比较指令	219
LD 系列触点比较指令	219
AND 系列触点比较指令	220
OR 系列触点比较指令	221
5.3 步进顺控指令说明	222
STL、RET 指令	222
6. 快捷键一栏	223
6.1 快捷键	223
6.2 编辑器的快捷键	224
7. 通讯运用	225
协议配置	225
7.1 Modbus 运用	226
7.1.1 modbus 主站通讯运用	227
7.1.2 modbus 从站通讯运用	229
MODBUS 从站地址映射表	232
7.2 COM2 口 N: N 通信协议	233

Wecon PLC Editor 编程软件概述

维控可编程控制器 PLC(Programmable Logic Controller)是一种专为工业环境下应用而设计的数字运算操作的电子系统。

它主要将外部的输入信号如：按键、感应器、开关及脉波等的状态读取后，依据这些输入信号的状态或数值并根据内部储存预先编写的程序，以微处理机执行逻辑、顺序、计时、计数及算术运算，产生相对应的输出信号如：继电器的开关、控制机械设备的操作。通过电脑或程序书写器可轻易的编辑/修改程序及监控装置状态，进行现场程序的维护与试机调整。维控可编程软件支持二种常用的语言：梯形图（LAD），指令列表（IL）。

梯形图编程

PLC 的梯形图在形式上沿袭了传统的继电器电气控制图，是在原继电器控制系统的继电器梯形图基础上演变而来的一种图形语言，设计中使用的元件如按钮 X、中间继电器 M、时间继电器 T、计数器 C、触点等，都和时间的电气元件的特性相似。如下图红框部分所示：



指令表编程

指令列表程序编辑器是一个文本编辑器，所有的逻辑和运算都使用指令和操作数的方式输入，根据指令所完成的功能和涉及到的操作数中的软元件，完成软元件的值读取、逻辑处理和软元件值写入。如下图红框部分所示：



程序转换

用户可以根据自己的编程习惯，在梯形图和指令列表中进行切换，以提高编程效率。如下图所示：

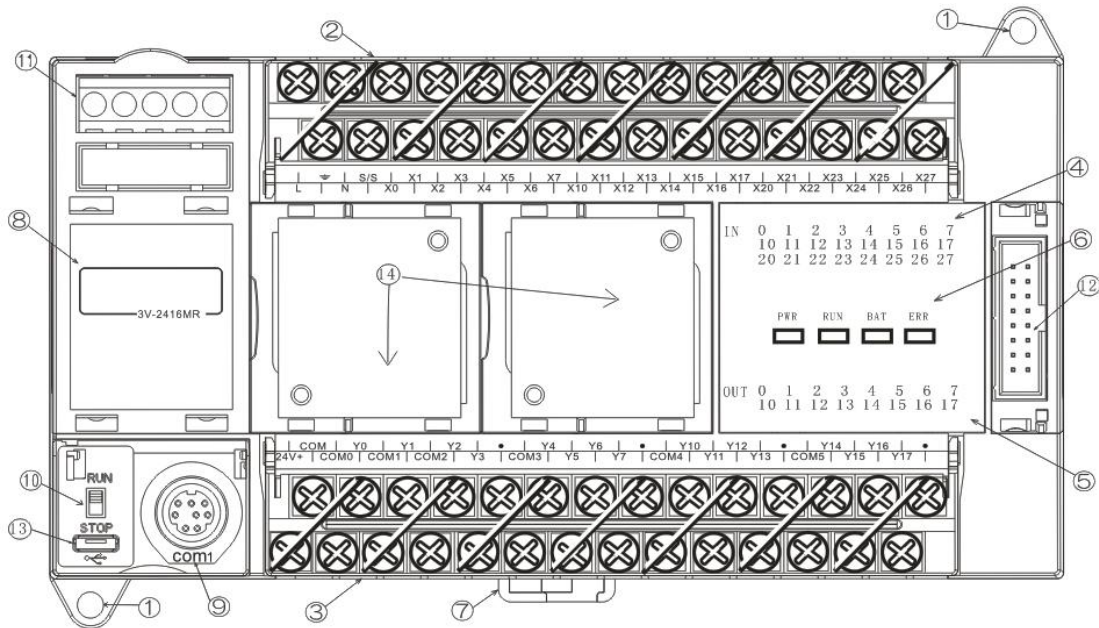


1.维控系列 PLC 介绍

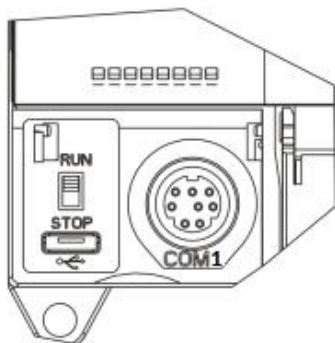
1.1 PLC 主机构成

根据硬件结构的不同，可将 PLC 分为主机、模块和 BD 板模块。

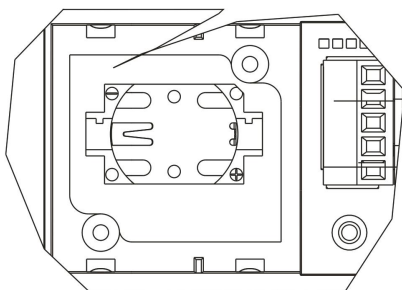
维控可编程控制器是内置电源、CPU、输入输出点的单元型可编程控制器。可编程控制器的主机称为【基本单元】；为了对其进行输入输出点数扩展，提供【扩展模块】；此外，还可以连接用于特殊控制的特殊扩展模块。



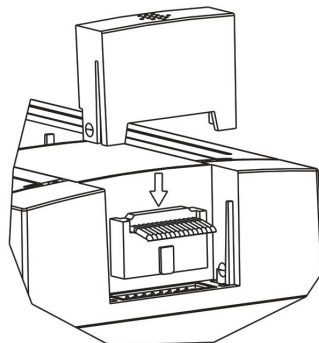
- | | |
|-----------------|------------------------|
| ① 安装孔：2 处（Φ4.5） | ⑦ DIN 导轨安装用卡扣 |
| ② 电源、输入信号端子台 | ⑧ 盖板 |
| ③ 供应电源、输出信号端子台 | ⑨ 连接编程设备(COM1) |
| ④ 输入显示 LED | ⑩ RUN/STOP 开关 |
| ⑤ 输出 LED | ⑪ COM1/COM2 口 RS485 端口 |
| ⑥ 动作指示灯： | ⑫ 扩展模块接口 |
| PWR：电源指示灯 | ⑬ USB 下载端口 |
| RUN：运行指示灯 | ⑭ BD 板插 |
| BAT：电池低压指示灯 | |
| ERR：程序错误时指示灯闪烁 | |
| ：CPU 错误时指示灯常亮 | |



设备连接口（放大）



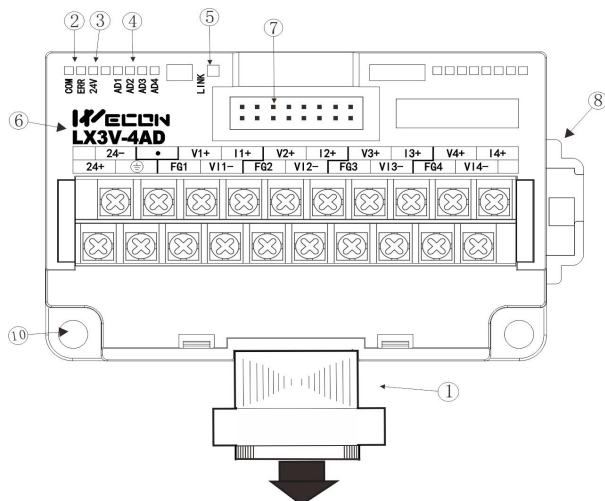
电池接口（放大）



扩展模块连接口（放大）

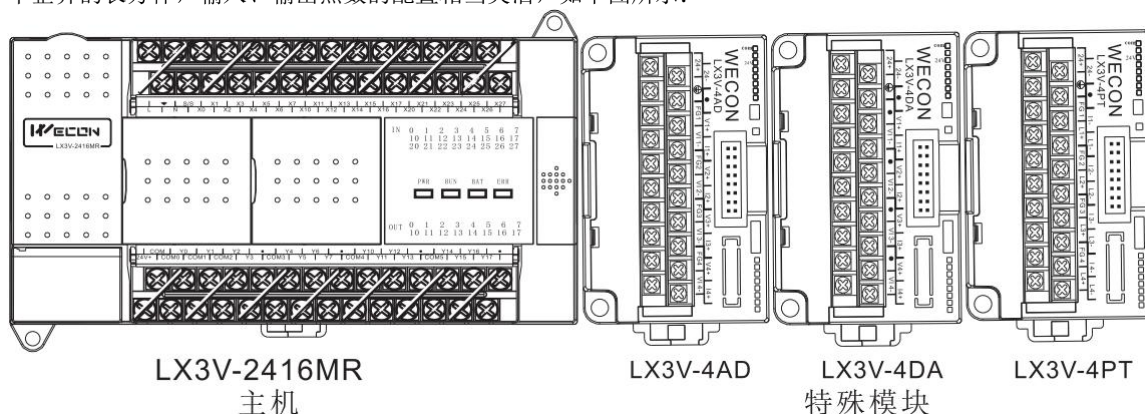
1.2 扩展模块构成

PLC 一般配备有许多专用的模块和特殊功能模块，例如 I/O 模块、模拟量输入/输出模块、高速脉冲输出模块、以太网通讯模块等，使 PLC 的功能得到扩展。



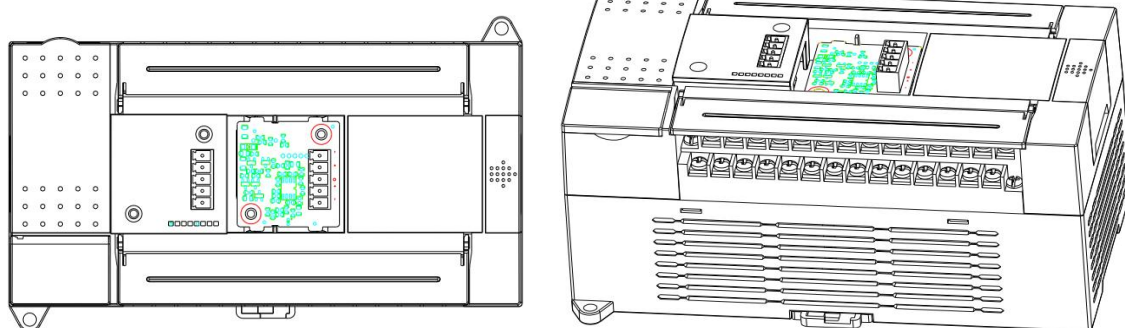
- ①扩展电缆
- ②COM 灯：通讯板与采集板的通讯指示灯
ERR 灯：通道的校准指示灯
- ③24V 灯：外接 DC24V 电源时常亮
- ④AD 灯：四个通道的输入指示灯
- ⑤LINK：PLC 和扩展模块通讯指示灯
- ⑥扩展模块的名称
- ⑦扩展模块的接口
- ⑧DIN 导轨安装用卡扣
- ⑨DIN 导轨的挂钩
- ⑩直接安装的孔：2 处（ $\Phi 4.5$ ）

维控系列 PLC 的基本单元、扩展模块的高度和深度相同，但是宽度不同。它们仅用扁平电缆连接，紧密拼装后组成一个整齐的长方体，输入、输出点数的配置相当灵活，如下图所示：



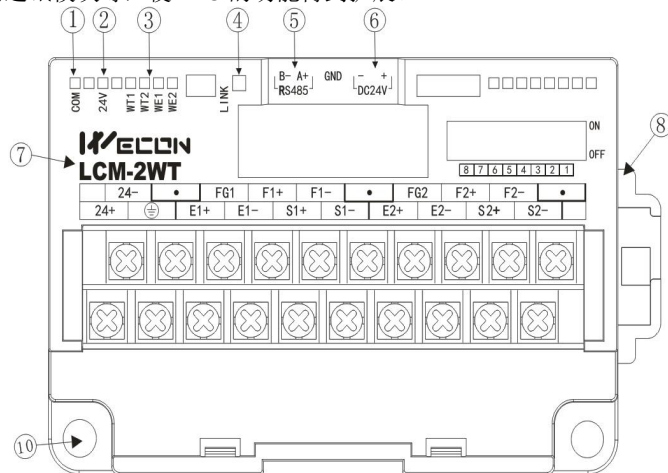
1.3 BD 板构成

相对于扩展模块而言，BD 板模块可多路混合，体积小、使用灵活、成本低、种类多，而且可直接插在基本单元上，不独立占用设备空间。如下图所示：



1.4 远程模块 LCM

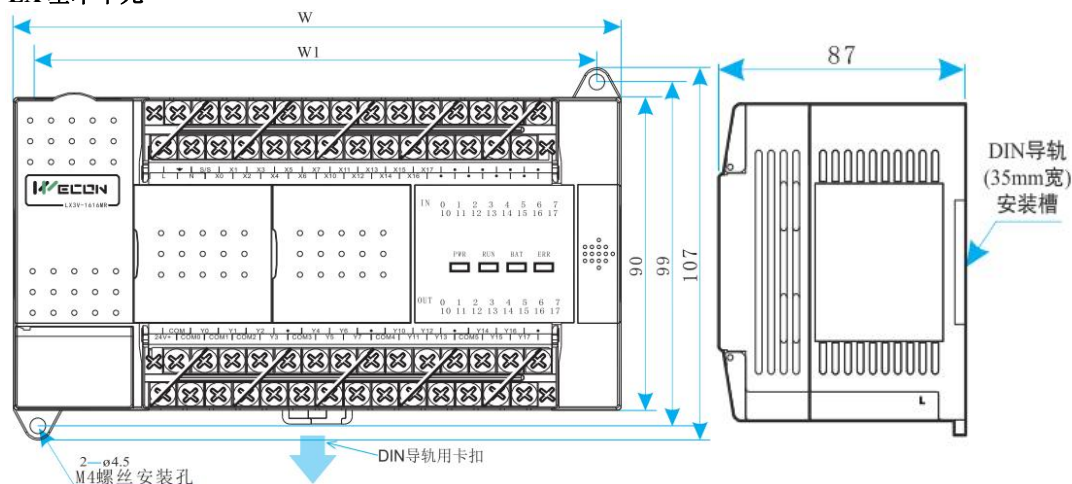
PLC 一般配备有许多专用的模块和特殊功能模块，例如 I/O 模块、模拟量输入/输出模块、高速脉冲输出模块、以太网通讯模块等，使 PLC 的功能得到扩展。



- ①COM 灯：通讯板与采集板的通讯指示灯
- ②24V 灯：外接 DC24V 电源
- ③WT 灯：通道的输入/输出指示灯
WE 灯：通道校准指示灯
- ④LINK：RS485 通讯指示灯
- ⑤RS485 通讯端子
- ⑥通讯模块 DC24V 电源
- ⑦扩展模块的名称
- ⑧DIN 导轨安装用卡扣
- ⑨DIN 导轨的挂钩
- ⑩直接安装的孔：2 处（ $\phi 4.5$ ）

1.5 外形尺寸

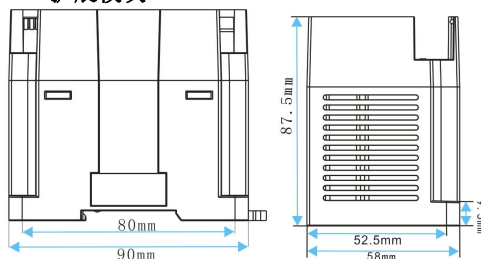
(1) LX 基本单元



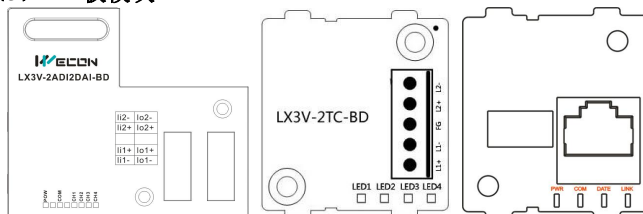
直接安装在 DIN（35mm）导轨上即可。卸下主机时，从下方轻轻拉出 DIN 导轨安装用卡扣。可利用安装孔直接用 M4 螺丝安装可编程控制器，安装孔的间距和位置请参照下表：

型号	W(mm)	W1(mm)	型号	W(mm)	W1(mm)
LX3V-0806MX	75	61	LX3V-1208MX	75	61
LX3V-1212MX	136	123	LX3V-1412MX	136	123
LX3V-1616MX	175	161	LX3V-2416MX	175	161
LX3V-2424MX	221	207	LX3V-3624MX	221	207

(2) LX 扩展模块



(3) BD 板模块

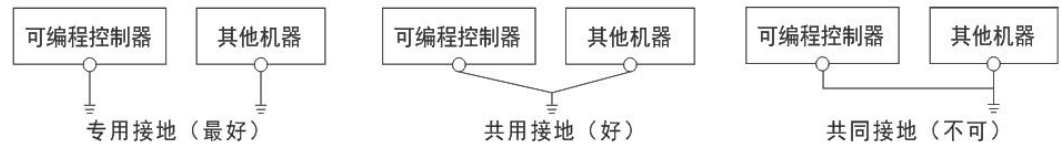


2.产品规格

2.1 一般规格

周围温度	使用时: 0~55℃, 保存时: -20~70℃				
相对湿度	使用时: 35~85%RH (无凝露)				
耐震性	符合 JISC0040 标准				
		频率	加速度	振幅	X、Y、Z 三方向 各 10 次 (各方向 80 分钟)
	DIN 导轨安装产品	10~57Hz	——	0.035mm	
		57~150Hz	4.9m/s2	——	
	直接安装产品	10~57Hz	——	0.075mm	
57~150Hz		9.8m/s2	——		
耐冲击	符合 JISC0040 标准 (147m/s², 作用时间 11ms, 正弦半波脉冲在 X、Y、Z 三方向各三次)				
耐噪声	噪声电压 1000Vp-p 噪声幅值 1us 上升 1ns 频率 30~100Hz 噪声模拟实验				
耐电压	AC1500V(1 分钟)			符合 JEM-1021 标准电源端子和	
绝缘阻抗	DC500V 绝缘测试器测得 5MΩ 以上			接地端子之间	
接地	第三种接地(不可与强电系统共地)※1				
使用环境	无腐蚀性、可燃性气体、无大量导电性尘埃(灰尘)				

※1



2.2 电气规格

(1) AC 电源型

项目	LX3V-0806/1208/1212/1410/1412MX-A	LX3V-1616/2416/2424/3624MX-A
额定电压	AC100~240V	
电压允许范围	AC85~264V	
额定频率	50/60Hz	
允许瞬停时间	100ms 以下瞬间停电, 电机能继续工作	
电源保险丝	250V 1A	250V 3.15A
冲击电流	最大 20A 5ms 以下/AC100V	
最大负载	20W	50W
传感器电源	DC24V 70mA	

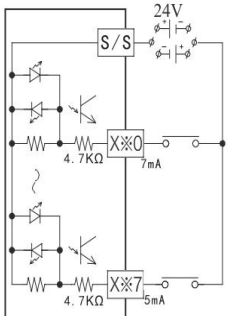
※1: 输入电流部分 (7mA/1 点、5mA/1 点) 也包含在内。

(2) DC 电源型

项目	LX3V/LX3VP/LX3VE 所有型号
额定电压	DC24V
电压允许范围	DC24V±10%
允许瞬停时间	100ms 以下瞬间停电, 电机能继续工作
电源保险丝	250V 3.15A
冲击电流	最大 15A 1ms 以下/AC100V
最大负载	30W 以下 (不记扩展模块外接电源部分)

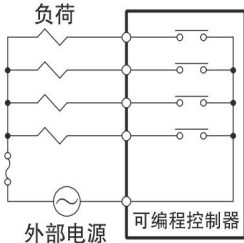
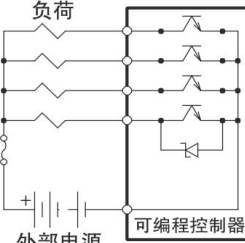
2.3 输入规格

LX 系列可编程控制器基本单元的规格如下表所示:

项目	AC 电源、DC 输入
型号	LX 系列基本单元
输入信号电压	DC $\pm 24\text{V} \pm 10\%$
输入信号电流	7mA/DC24V(X002 以后, 5mA/DC24V)
输入 OFF 电流	1.5mA 以下
输入响应时间	约 10ms 可通过内置的数字滤波器 D8020 变更为 0~15ms/PLC 参数设定
输入信号形式	触点输入或是 NPN、PNP 开集电极晶体管输入
回路绝缘	光耦绝缘
输入动作表示	输入 ON 时 LED 灯亮
输入回路构成	 图中是 NPN 接法, 若 S/S 接 24V 负极, X 接正极, 即为 PNP 接法。

※1 : X002 以后是 4.7K Ω 。

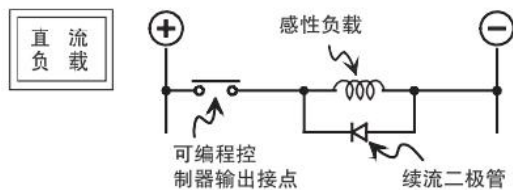
2.4 输出规格

项目		继电器输出	晶体管输出
型号		LX 系列所有机型型号	
输出回路构成			
外部电源		AC250V 以下, DC30V 以下	DC5~30V
回路绝缘		机械绝缘	光电耦合绝缘
动作表示		继电器线圈通电是 LED 灯亮	光耦合器驱动时 LED 灯亮
最大负载	电阻负载	2A/1 点, 8A/4 点	0.5A/1 点, 0.8A/4 点(Y0, Y1 是 0.3A/1 点)
	电感负载	80V/A	12W/DC24V(Y0, Y1 是 7.2W/DC24V)
	灯负载	100W	0.9W/DC24V(Y0, Y1 是 0.9W/DC24V)
开路漏电		——	0.1mA/DC30V
最小负载		DC5V2mA 参考值	——
响应时间	输入 ON 电流	约 10ms	0.2ms 以下 5μs(Y0, Y1 时)
	输入 OFF 电流	约 10ms	0.2ms 以下 5μs(Y0, Y1 时)
输出信号		——	NPN 模式

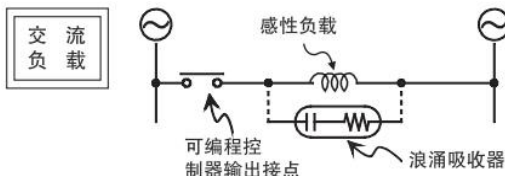
【输出回路构成】

请把直流感性负载与续流二极管并联，否则会显著降低接点寿命。

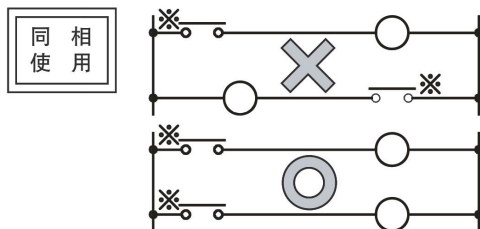
续流二极管的反向耐电压是负载电压的 5~10 倍以上，正向电流值高于负载电流。



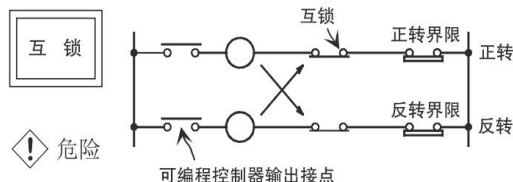
如果是交流感性负载，将负载与浪涌吸收器并联设计，可以减少噪音的发生。



最好在同相侧使用可编程控制器的输出接点。



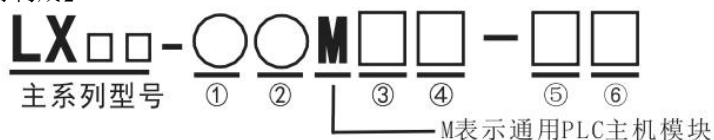
正反转用的接触器同时合上十分危险，像这样的负载除了用可编程控制器内部的程序进行互锁控制之外，在可编程控制器外部也一定要设置互锁。



2.5 产品命名规则

可编程控制器的型号名称见产品侧面的型号标签。

【基本单元的型号名称构成】



型号名称构成中的①~⑥表示以下的规格：

①输入点数

②输出点数

③输出方式：R=继电器输出（有触点、交流、直流负载两用）

T=晶体管输出（无触点、直流负载用）

④扩展口：4H=带有4轴的高速脉冲输出

2H=带有2轴的高速脉冲输出

⑤电源类型：A：AC220V 输入，省略为默认 AC220V；D：DC24V 输入。

⑥指令集：1=1S 指令集；2=2N 指令集，无=默认 2N 指令集。

3.各种软元件说明

Wecon PLC 所支持的软元件类型如下表所示：

序号	元件	功能与分类
1	输入继电器 X	对应 PLC 的硬件开关量输入的位元件
2	输入继电器 Y	对应 PLC 的控制输出的位元件
3	辅助继电器 M	普通辅助继电器 M 位元件；系统特殊继电器 M 位元件
4	状态继电器 S	步进控制用状态标志位元件
5	定时器 T	具有 1ms、10ms、100ms 步长的 16bit 定时器
6	计数器 C	具有 16bit/32bit 增/减型计数器、高速计数器、单/双相各种计数器
7	数据寄存器 D	数据寄存器 D；数据间接寻址寄存器 V、Z、文件寄存器 D
8	指针 P・I	跳转指针 P、子程序指针 P、中断子程序 I、有高速输入、定时、计数等中断
9	常数 K・H	二进制、十进制、十六进制、浮点数等

寄存器	LX3V(1S 指令集)	LX3V(2N 指令集)	LX3VP	LX3VE	附带扩展
输入继电器 X	X0~X13 最多 12 点	X0~X43 最多 36 点	X0~X43 最多 36 点	X0~X43 最多 36 点	X0~X77 最多 128 点
输出继电器 Y	Y0~Y7 最多 8 点	Y0~Y27 最多 24 点	Y0~Y27 最多 24 点	Y0~Y27 最多 24 点	Y0~Y77 最多 128 点

寄存器		LX3V/LX3VP/LX3VE							
辅助继电器 M		M0~M499 500 点 一般用※1		M500~M1023 524 点 保持用※2		M1024~M3071 2048 点 保持用※3		M8000~M8255 256 点 特殊用※4	
		S0~S499 500 点 一般用※1 初始化 S0~S9 原点回归用 S10~S19		S500~S899 400 点 保持用※2				S900~S999 100 点 信号报警用※2	
定时器 T		T0~T199 200 点 100ms 子程序用: T192~T199		T200~T245 46 点 10ms		T246~T249 4 点 1ms 累积※3		T250~T255 6 点 100ms 累积※2	
计数器 C		16 位增量计数		32 位加減计数		32 位高速加減计数器			
		C0~C99 100 点 一般用※1	C100~C199 100 点 保持用※2	C200~C219 20 点 一般用※1	C220~C234 15 点 保持用※2	C235~C245 1 相 1 输入※2	C245~C250 1 相 2 输入※2	C251~C255 2 相输入※2	
数据寄存器 D		D0~D199 200 点 一般用※1	D200~D511 312 点 保持用※2	D512~D7999 7488 点 保持用※3 文件用 D1000 以后可设定 为文件寄存器用		D8000~D8195 312 点 特殊用※2	V0~V7 Z0~Z7 16 点 变址用※1		
指针 P・I		N0~N7 8 点 主控用	P0~P127 128 点 跳跃、子程序用、 分支式指针	I00□~I50□ 6 点 输入中断用指针		I6□□~I8□□ 3 点 定时器中断用指针		I010~I060 6 点 计数器中断用指针	
常数	K	16 位 -32768~32767				32 位 -2147483648~2147483647			
	H	16 位 0~FFFFH				32 位 0~FFFFFFFFH			
	E	±1.175495 E-38~±3.402823 E+38 (有效位 7 位, 仅 LX3V\LX3VE 系列 PLC 支持)							

※1. 非停电保持领域。使用参数设定, 可变更停电保持领域。

※2. 停电保持领域。使用参数设定, 可变更非停电保持领域。

※3. 有关停电保持的特性, 无法用参数来改变。

※4. 不同系列 PLC 的对用功能请查看【[3.11 特殊软元件一览表](#)】。

3.1 输入继电器 X

输入继电器是 PLC 接收外部触点和电子开关的开关量信号的窗口，输入继电器的元件号是以八进制方式编号，八进制数只有 0~7 这 8 个数字符号，遵循“逢 8 进 1”的运算规则，因此不存在诸如 8、9 这样的数值。

PLC 通过光电耦合器，将外部信号的状态读入并存储在输入映像寄存器内，用程序指令方法是无法修改输入继电器的状态。

输入端可以外接常开触点或常闭触点，也可以接多个触点组成的串并联电路，其节点信号（常开型、常闭型）在用户梯形图程序中可无限次使用。

表 3.1 维控各系列 PLC 的输入/输出表

型号	输入	输出	型号	输入	输出
LX3V-0806MR/MT-A1(D1)	X0~X7	Y0~Y5	LX3VP-1208MR/MT-A(D)	X0~X7	Y0~Y5
LX3V-1208MR/MT-A1(D1)	X0~X13	Y0~Y7	LX3VP-1212MR/MT-A(D)	X0~X13	Y0~Y13
LX3V-0806MR/MT-A2(D2)	X0~X7	Y0~Y5	LX3VP-1412MR/MT-A(D)	X0~X15	Y0~Y13
LX3V-1208MR/MT-A2(D2)	X0~X13	Y0~Y7	LX3VP-1616MR/MT-A(D)	X0~X15	Y0~Y17
LX3V-1212MR/MT-A2(D2)	X0~X13	Y0~Y13	LX3VP-2416MR/MT-A(D)	X0~X17	Y0~Y17
LX3V-1410MR/MT-A2(D2)	X0~X15	Y0~Y11	LX3VP-2424MR/MT-A(D)	X0~X27	Y0~Y27
LX3V-1412MR/MT-A2(D2)	X0~X15	Y0~Y13	LX3VP-3624MR/MT-A(D)	X0~X43	Y0~Y27
LX3V-1616MR/MT-A2(D2)	X0~X17	Y0~Y17	LX3VE-1412MR/MT-A(D)	X0~X15	Y0~Y13
LX3V-2416MR/MT-A2(D2)	X0~X27	Y0~Y17	LX3VE-1616MR/MT-A(D)	X0~X17	Y0~Y17
LX3V-2424MR/MT-A2(D2)	X0~X27	Y0~Y27	LX3VE-2416MR/MT-A(D)	X0~X27	Y0~Y17
LX3V-3624MR/MT-A2(D2)	X0~X43	Y0~Y27	LX3VE-2424MR/MT-A(D)	X0~X27	Y0~Y27
			LX3VE-3624MR/MT-A(D)	X0~X43	Y0~Y27

其他详细系列说明，请查看“维控产品选型手册”或咨询我司业务员。

3.2 输出继电器 Y

输出继电器是 PLC 向外部负载发送信号的窗口，输入继电器的元件号是以八进制方式编号，八进制数只有 0~7 这 8 个数字符号，遵循“逢 8 进 1”的运算规则，因此不存在诸如 8、9 这样的数值。

输出继电器用来将 PLC 的输出信号传送给输出模块，再由后者驱动外部负载。输出模块中的每一个硬件继电器仅有一对常开触点，但是在梯形图中，每一个输出继电器的常开触点和常闭触点都可以多次使用。

3.3 辅助继电器 M

辅助继电器（M 元件）相当于继电器控制系统中的中间继电器，它不能接收外部的输入信号，也不能直接驱动外部负载，它的功能是用软件实现的。M8000 以上的变量为系统特殊地址，用于 PLC 用户程序与系统状态的交互；部分 M 变量具有掉电保存功能。

(1) 通用辅助继电器

LX3V 系列 PLC 的通用辅助继电器的元件号为 M0~M499，共 500 点。辅助继电器的类型与它的元件号和 PLC 的型号有关（见表 3.3）。

如果在 PLC 运行时电源突然中断，输出继电器和 M0~M499 将全部变为 OFF。若电源再次接通，除了因外部输入信号而变为 ON 之外，其余仍然保持 OFF 状态，可以用软元件将 M0~M499 设定为下面介绍的保持型辅助继电器。

表 3.3 辅助继电器

型号	一般用	停电保持用	停电保持专用	特殊用
LX3V(1S 指令集)	M0-M383 384 点※3	——	M384-M511 128 点※3	M8000-M8255 256 点
LX3V(2N 指令集)	M0-M499 500 点※1	M500-M1023 524 点※2	M1024-M3071 2048 点※3	M8000-M8255 256 点
LX3VP	M0-M499 500 点※1	M500-M1023 524 点※2	M1024-M3071 2048 点※3	M8000-M8255 256 点
LX3VE	M0-M499 500 点※1	M500-M1023 524 点※2	M1024-M3071 2048 点※3	M8000-M8255 256 点

可编程控制器内的一般用辅助继电器、停电保持用辅助继电器的区域分配，可通过参数设定来进行调整。

- ※1. 非停电保持领域。使用参数设定，可变更停电保持领域。
- ※2. 停电保持领域。使用参数设定，可变更非停电保持领域。
- ※3. 有关停电保持的特性，无法用参数来改变。

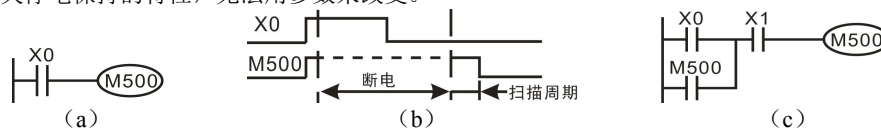


图 3.3 断电保持功能

(2) 保持型辅助继电器 M500~M3071

有些自动化控制设备要求记忆电源断开前时的状态，重新上电后再现其状态，LX3V 系列 PLC 中 M500-M3071 可以用于这种场合，其中 M500-M1023 可以用于软件设定，变为非掉电保持辅助继电器。

在电源中断时，保持它们的映像寄存器中的内容，它们只是在 PLC 重新上电后的第一个扫描周期内变为 ON，图 3.3 (b) 是图 (a) 的电路波形图。为了利用它们的断电记忆功能，采用图 (c) 所示的电路。当电源断开又重新上电后，M500 的线圈将一直“通电”，直到 X1 的常闭触点断开，其自保持功能是 M500 的常开触点实现的。

(3) 特殊辅助继电器

特殊辅助继电器用来表示 PLC 的某些状态，提供时钟脉冲和标志（例如进位、错位标志），设定 PLC 的运行方式，或者用于步进顺控、禁止中断、设定计数器是加计数或是减计数等。

这些特殊辅助继电器各有其特定的功能，可分为以下两类：

- 1) 触点利用型的特殊辅助继电器，为 PLC 系统自动驱动线圈，用户程序只能读取使用，如：
 - M8000：运行监视器（在运行中接通），常用于需用驱动信号的指令之前。
 - M8002：初始脉冲（仅在运行开始时瞬间接通），常用于只需执行一次初始化指令。
 - M8012：100ms 时钟脉冲，用于产生固定间隔翻转的信号。
- 2) 线圈驱动型特殊辅助继电器，为用户程序驱动线圈，用于控制 PLC 的工作状态和执行模式等，如：
 - M8030：电池发光两极管熄灭灯指令
 - M8033：停止时保持输出
 - M8034：输出全部禁止
 - M8039：恒定扫描

注意：M 元件存在驱动时有效与 END 指令执行后有效两种情况；用户不可使用尚未定义的特殊辅助继电器。

3.4 状态继电器 S

状态继电器 S 是用于编制顺序控制程序的一种编程元件，利用 STL（步进梯形指令）控制步进状态 S 的转移，简化编程设计。

通用状态（S0~S499）没有掉电保持功能，但是用程序可以将它们设定为有掉电保持功能的状态，其中包括供初始状态用的 S0~S9 和供返回原点用的 S10~S19。S500~S899 有掉电保持功能，S900~S999 供报警器使用。

若没有采用 STL 编程方式，S 可当做 M 变量来使用。供报警器用的状态可以用于外部故障诊断的输出。

表 3.4 状态继电器

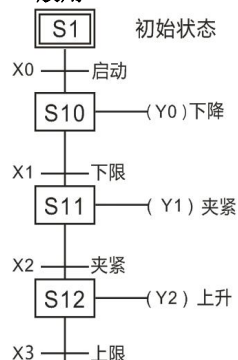
型号	一般用			停电保持用			报警器用
	——	初始化用	——	——	初始化用	——	
LX3V (1S 指令集)	——	——	——	S0-S127 128 点 ※3	S0-S9 (10 点)	S10-S19 (10 点)	——
LX3V (2N 指令集)	S0-S499 500 点※1	S0-S9 (10 点)	S10-S19 (10 点)	S500-S899 400 点※2	——	——	S900-S999 100 点※2
LX3VP	S0-S499 500 点※1	S0-S9 (10 点)	S10-S19 (10 点)	S500-S899 400 点※2	——	——	S900-S999 100 点※2
LX3VE	S0-S499 500 点※1	S0-S9 (10 点)	S10-S19 (10 点)	S500-S899 400 点※2	——	——	S900-S999 100 点※2

※1：非停电保持领域。通过参数的设定可变更停电保持的领域。

※2：停电保持领域。通过参数的设定可变更非停电保持的领域。

※3：停电保持特性。不可通过参数的设定变更。

(1) 一般用



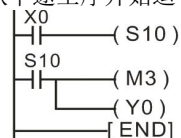
如左图所示的工序步进控制中，如果启动信号 X0=ON，则状态 S10 置位（ON），下降用的电磁阀 Y0 开始动作。其结果是，若下限限位开关 X1 为 ON，则状态 S11 置位（ON），夹紧用的电磁阀 Y1 动作。

如果夹紧动作确认的限位开关 X2=ON，则状态 S12 置位（ON）。

随着状态动作的转移，状态自动返回原状态。

(2) 停电保持型

一般用状态在电源断开后，变为 OFF 状态；但停电保持用状态能记忆电源停电前一刻的 ON/OFF 状态，因此，也能从中途工序开始运行。



状态寄存器和辅助继电器一样，有无数的常开、常闭触点，在顺控程序内可随意使用。此外，在不用于步进梯形图指令时，状态 S 也与辅助继电器一样可在一般的顺控中使用。

(3) 信号报警器用



如果 X1 和 X2 同时接通 1 秒以上，则 S900 被置位，以后即使 X1 或 X2 为 OFF 状态，S900 仍保持动作状态（但是 T0 会复位，值变成 0）。若不满 1 秒，X1 或 X2 变为 OFF 时，定时器复位。

3.5 定时器 T

PLC 中的定时器 (T) 相当于继电器系统中的时间继电器, 用于完成定时功能。它有一个设定值寄存器 (一个字长)、一个当前值寄存器 (一个字长) 和一个用来储存其输出触点状态的映像寄存器 (占二进制的一位), 这三个存储单元使用同一个元件号。

LX 系列 PLC 的定时器分为通用定时器和累计型定时器。常数 K 可作为定时器的设定值, 也可用数据寄存器 (D) 的数值来设定。外部数字开关输入的数据可存入数据寄存器, 作为定时器的设定值。

定时器的类型与它的元件号有关, 且定时器有不同的计时步长, 如 1ms、10ms、100ms 等。当达到所定的设定值时输出触点动作。

定时器 (T) 的编号如下表所示:

表 3.5 定时器

型号	100ms 型 0.1~3276.7s	100ms 型 0.1~3276.7s 0.01~327.67s	10ms 型 0.01~327.67s	1ms 型 0.001~32.767s	1ms 型累计型 0.001~32.767s	100ms 累计型 0.1~3276.7s
LX3V (1S 指令集)	T0~T31 32 点	T32~T62 31 点	T32~T62 31 点 (M8028 控制)	——	T63 1 点	——
LX3V (2N 指令集)	T0~T199 200 点 子程序用 T192~T199	——	T200~T245 46 点	——	T246~T249 4 点 执行中断的保持用	T250~T255 6 点 保持用
LX3VP	T0~T199 200 点 子程序用 T192~T199	——	T200~T245 46 点	T50~T191 42 点 (M8028 控制)	T246~T249 4 点 执行中断的保持用	T250~T255 6 点 保持用
LX3VE	T0~T199 200 点 子程序用 T192~T199	——	T200~T245 46 点	T50~T191 42 点 (M8028 控制)	T246~T249 4 点 执行中断的保持用	T250~T255 6 点 保持用

提示: 不用作定时器的定时器编号, 也可用作数值存储用的数据寄存器。

备注:要确保最小时间间隔, 请将预设值增大 1。如: 需要使用 100ms 定时器时, 为确保最小时间间隔至少为 100ms, 不建议使用 T0 K1 的方式, 建议使用 10ms 计时步长的计时器, 或是将预设值增大 1。

(1) 通用定时器 (T0~T245)



图 3.5 (a)

如上图所示: X0 的常开触点接通时, T200 的当前值计数器从零开始计时, 对 10ms 时钟脉冲进行累加计数。当前值等于设定值 223 时, 定时器的常开触点接通, 常闭触点断开, 即 T200 的输出触点在其线圈被驱动 2.23s 后动作。X0 的常开触点断开后, T200 因为线圈断电而被复位, 复位后它的常开触点断开, 常闭触点接通, 当前值恢复为零。

(2) 累计型定时器 (T246~T255)

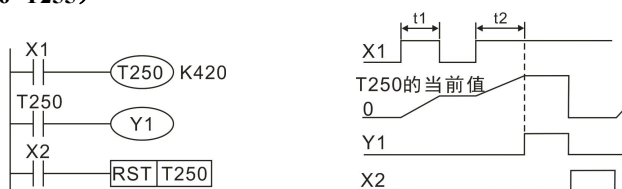
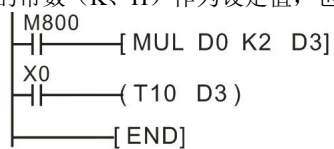


图 3.5 (b)

如图 3.5 (b) 中的 X1 常开触点接通时, T250 的当前值计数器对 100ms 时钟脉冲进行累加计数。X1 的常开触点断开或停止时停止计数, 当前值保持不变。当 X1 的常开触点再次接通时继续计数, 累计时间 t_1+t_2 为 42s 时, 当前值等于设定值 420, T250 的常开触点接通, 常闭触点断开。X2 的常开触点接通时, T250 复位 (因为累计型定时器的线圈断电时不会复位, 需要用 X2 的常开触点和复位指令使 T250 强制复位)。

(3) 设定值

定时器的时间可采用程序存储器内的常数（K、H）作为设定值，也可用数据寄存器（D）的内容进行间接指定。



PLC 上电后进行乘法运算， $D3=D0*2$ 。将 D3 的数据作为 T10 的定时时间值。

3.6 计数器 C

内部计数器（C）用于完成计数功能，每个计数器含有线圈、接点、计时值寄存器，每当计数器线圈的驱动信号由 OFF→ON 时，计数器读数增加 1，若计时值达到预设的时间值时，其接点动作。

内部计数器输入信号的接通或断开的持续时间，应大于 PLC 的扫描周期。计数器的设定值除了用常数 K 之外，还可以用指定的数据寄存器设定，这时设定值等于指定数据寄存器中的数值。32 位设定值存放在元件号相连的两个数据寄存器中。如指定的寄存器是 D0，则设定值存放在 D1 和 D0 中。32 位加/减计数器的设定值可正可负。

部分定时器的具有掉电保持、累计等特性，重新上电后仍维持掉电前的数值。

计数器中有 16bit、32bit 宽度；有单向计数型、增减计数型、双相计数型等，部分计数器的计数值还具有掉电保持特性等，使用时根据需用选择合适的计数器。

（1）16 位加计数器：

表 3.6 内部计数器

型号	16 位加计数器 0~32, 767 计数		32 位加/减计数器 -2, 147, 483, 648~+2, 147483647	
	一般用	停电保持用	一般用	停电保持用
LX3V(1S 指令集)	C0-C15 16 点※3	C16-C31 16 点※3	——	——
LX3V(2N 指令集)	C0-C99 100 点※1	C100-C199 100 点※2	C200-C219 20 点※1	C220-C234 15 点※2
LX3VP	C0-C99 100 点※1	C100-C199 100 点※2	C200-C219 20 点※1	C220-C234 15 点※2
LX3VE	C0-C99 100 点※1	C100-C199 100 点※2	C200-C219 20 点※1	C220-C234 15 点※2

※1 非停电保持领域。通过设定参数可变更停电保持领域。

※2 停电保持领域。通过设定参数可变更非停电保持领域。

※3 试通过设定参数不可改变有关停电保持的特性。

注意：不作为计数器使用的计数器编号，可以作为数据记忆用的数据寄存器使用。

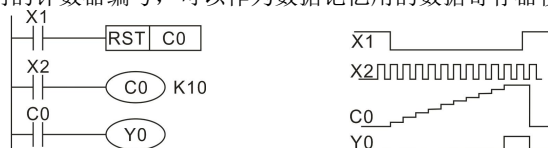


图 3.6 16 位加计数器

16 位加计数器的设定值为 1~32767。如图 3.6 的加计数器的工作过程所示，图中 X1 的常开触点接通后，C0 被复位，它对应的位存储单元被置 0，C0 的常开触点断开，常闭触点接通，同时其计数当前值置 0。X2 提供计数输入信号，当计数器的复位输入电路断开，计数输入电路由断开变为接通（即计数脉冲的上升沿）时，计数器 C0 的当前值加 1。在 10 个计数脉冲后，C0 的当前值等于设定值 10，它对应的位存储单元被置 1，此时 Y0 触点接通。再来计数脉冲时当前值不变，直到复位输入信号接通，计数器的当前值被置为 0。

（2）32 位加/减计数器：

32 位加/减计数器的设定值为-2147483648~2147483647，其中 C200~219（共 20 点）为通用型，C220~234（共 15 点）为掉电保持型。

32 位加/减计数器 C200~234 的加/减计数方式由特殊辅助继电器 M8200~M8234 设定，对应的特殊辅助继电器为 ON 时，为减计数；反之，为加计数。

3.7 高速计数器

21 点高速计数器 C235~C255 共用 PLC 的 8 个高速计数器输入端 X0-X7，某一输入端同时只能供一个高速计数器使用。，如下表所示分类，对应各高速计数器编号。输入 X0-X7 不能重复由高速计数器使用，在输入端子不作为高速计数器使用时，可用于一般输入。

表 3.7 高速计数器

中断 输入	单相单计数输入										
	C235	C236	C237	C238	C239	C240	C241	C242	C243	C244	C245
X0	U/D						U/D			U/D	
X1		U/D					R			R	
X2			U/D					U/D			U/D
X3				U/D				R			R
X4					U/D				U/D		
X5						U/D			R		
X6										S	
X7											S
中断 输入	单相双计数输入					双相双计数输入					-
	C246	C247	C248	C249	C250	C251	C252	C253	C254	C255	-
X0	U	U		U		A	A		A		-
X1	D	D		D		B	B		B		-
X2		R		R			R		R		-
X3			U		U			A		A	-
X4			D		D			B		B	-
X5			R		R			R		R	-
X6				S					S		-
X7					S					S	-

U: 增计数输入 D: 减计数输入 A: A 相输入 B: B 相输入 R: 复位输入 S: 开始计数输入

输入 X0-X7 不可重复使用。例如：一旦使用 C251，则 X0、X1 被占用，因此 C235、C236、C241、C244、C246、C247、C249、C252、C254 及中断输入指针 I00*、I10*及相应输入的 SPD 指令不能使用。用 M8235~M8245 来设置 C235~C245 的计数方向，对应的 M 为 ON 时为减计数，为 OFF 时为加计数。

(1) 单相计数

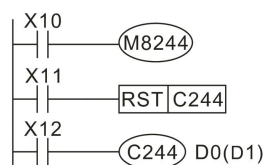


图 3.7 (a)

如图 3.7 (a) 中，C244 是一相带起动/复位端的高速计数器，由表 3.7 可得，X1 和 X6 分别为复位输入端和起动输入端。若 X12 为 ON，并且 X6 也为 ON 时立即开始计数，计数输入端为 X0，C244 的设定值由 D1 和 D0 指定。除了用 X1 来立即复位外，也可在梯形图中用 X11 复位。利用 M8244 可设置 C244 为加计数或减计数。

(2) 双相计数



图 3.7 (b)

C251~C255 为两相 (AB 相) 双计数输入高速计数器。如图 3.7 (b) 所示，当 X14 为 ON 时，C251 通过中断，对 X0 输入 A 相信号和 X1 输入 B 相信号的动作计数。X13 为 ON 时，C251 被复位，当计数值大于等于设定值时 Y2 线圈通电；若计数值小于设定值，Y2 线圈断电。

A 相输入接通时，若 B 相输入由断开变为接通，为加计数（如上图 (a) 所示）；A 相输入接通时，若 B 相输入由接通变为断开，为减计数（如上图 (b) 所示）。根据这一原理，可利用旋转轴上安装的 AB 相型编码器，在机械正

转时自动进行加计数，反转时自动进行减计数。加计数时 M8251 为 OFF，减计数时 M8251 为 ON，通过 M8251 可监视 C251 的加/减计数状态。

如使用高速计数器，则与输入编号相对应的滤波器常数自动变更，以对应高速的信号获取。

Y 输出：晶体管高速脉冲输出

最多 4 路(四轴)，且可以分别同时输出各 200K 的脉冲频率；

可将频率输出给变频器，步进电机，伺服电机等脉冲控制设备；

X 输入：单相

X0、X1 的硬件计数器(C235、C236、C246)，可同时输入 200K 的脉冲；

X0、X1 的软件计数器(C241、C244、C247、C249)，可同时输入各 100K 的脉冲；

默认硬件计数器，使用 HSCS、HSCR、HSZ 指令后切换为软件计数器；

后 4 路软件计数，分别可同时输入各 10K 的脉冲频率。

X 输入：AB 相

X0、X1 的硬件计数器(C251)，可输入 100K 的脉冲；

X0、X1 的软件计数器(C252、C254)分别可同时输入各 50K 的脉冲；

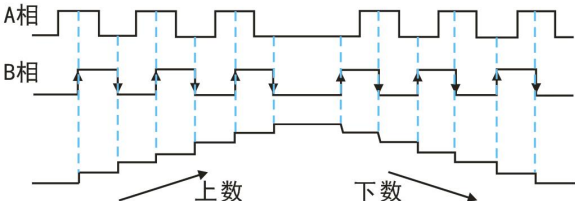
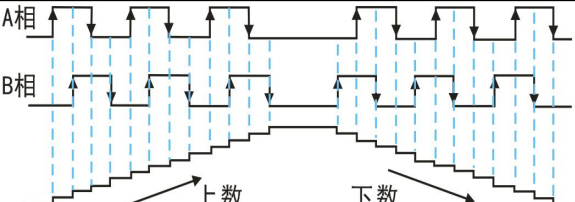
默认硬件计数器，使用 HSCS、HSCR、HSZ 指令后切换为软件计数器；

后 4 路软件计数，分别可同时输入各 5K 的脉冲频率。

* 2 相 2 输入提供 2 倍，4 倍频模式可选择（如下表所示），共享 D8200 选择倍频模式。

寄存器 C251 倍频功能，当 D8200 的值为不同数值，代表着不同的倍频，其中 0：1 倍频；1：2 倍频；2：4 倍频。

注意：HSCS、HSCR、HSCZ 不能与倍频同时使用，而且版本 V311 以上的才能支持倍频功能。

D8200 数值	计数图示
K2 (二倍频)	
K4 或其他数值 (四倍频，默认值)	

3.8 数据寄存器 D

数据寄存器（D）在模拟量检测与控制以及位置控制等场合用来储存数据和参数，如对定时器、计数器、模拟量参数的运算和运算等。数据寄存器为 16 位（最高位为符号位），两个合并起来可以存放 32 位数据。在 D0 和 D1 组成的双字数据中，D0 存放低 16 位，D1 存放高 16 位。

维控 LX 系列 PLC 多数指令中参与运算的数据是按有符号数进行处理的，对于 16bit 的寄存器，bit15 为符号位（0 表示正数，1 表示负数）；对于 32bit 的寄存器，高字节的 bit15 为符号位，数值范围为-32, 768 ~+32, 767。

表 3.8 数据寄存器

型号	一般用	停电保持用	停电保持专用		特殊用	指定用
			停电保存专用	文件用		
LX3V (1S 指令集)	D0-D127 128 点※3	——	D128-D255 128 点※3	根据参数设定, 可以将 D1000-D2499 作为文件寄存器使用	D8000-D8255 256 点	V0-V7 Z0-Z7 16 点
LX3V (2N 指令集)	D0-D199 200 点※1	D200-D511 312 点 ※2	D512-D7999 7488 点※3	根据参数设定, 可以将 D1000 以后作为文件寄存器	D8000-D8255 256 点	V0-V7 Z0-Z7 16 点※3
LX3VP	D0-D199 200 点※1	D200-D511 312 点 ※2	D512-D7999 7488 点※3	根据参数设定, 可以将 D1000 以后作为文件寄存器	D8000-D8255 256 点	V0-V7 Z0-Z7 16 点※3
LX3VE	D0-D199 200 点※1	D200-D511 312 点 ※2	D512-D7999 7488 点※3	根据参数设定, 可以将 D1000 以后作为文件寄存器	D8000-D8255 256 点	V0-V7 Z0-Z7 16 点※3

※1: 非停电保持 领域。通过设定参数可变更停电保持领域。

※2: 停电保持领域。通过设定参数可变更非停电保持领域。

※3: 通过设定参数无法变更停电保持的特性。

(1) 通用数据寄存器

将数据写入通用通用寄存器后，其值保持不变，直到下一次被改写。PLC 从 RUN 模式进入 STOP 模式时，所有的通用寄存器的数值将被写 0。如果特殊辅助继电器 M8033 为 ON，PLC 从 RUN 进入 STOP 模式时，通用寄存器数值保持不变。

(2) 保持型数据寄存器

PLC 从 RUN 进入 STOP 模式时，保持型数据寄存器的值保持不变。通过参数设置，可改变保持型数据寄存器的范围。

(3) 特殊数据寄存器

特殊数据寄存器 D8000~D8255 共 256 点，用来控制和监视 PLC 内部的各种工作方式和元件，例如电池电压、扫描时间、正在动作的状态编号等。PLC 上电时，这些数据寄存器被写入默认的数值。具体数据可查看【[3.11 章特殊地址功能说明](#)】。

(4) 文件寄存器

数据寄存器 D1000 以后是普通用的保持用寄存器，可设定作为最大 7000 点的文件寄存器使用。通过参数设定，可指定 1~14 个块（1 个块相当于 500 个文件寄存器），但是每增加 1 个记录块就要减少 500 步的程序储存区域，用于备份文件寄存器。将 D1000 以后的一部分设定为文件寄存器时，剩余部分仍可作为通用的保持寄存器使用。

(5) 变址寄存器

变址寄存器 V 与 Z 同普通的数据寄存器一样，是进行数值数据的读入、写出的 16 位数据寄存器。V0~V31，Z0~Z31 共有 64 个。

变址寄存器除了和普通的数据寄存器有相同的使用方法外，在应用指令的操作数中，还可以同其他的软件元件编号或数值组合使用。例如当 V0=12 时，数据寄存器 D6V0 相当于 D18（即 12+6=18）。通过修改变址寄存器的数值，可改变实际操作数。变址寄存器也可用来修改常数的数值，例如当 Z0=20 时，K4Z0 相当于常数 24（即 20+4=24）。

注意：LD，AND，OUT 等基本顺控指令或步进梯形图指令的软件元件编号不能同变址寄存器组合使用。

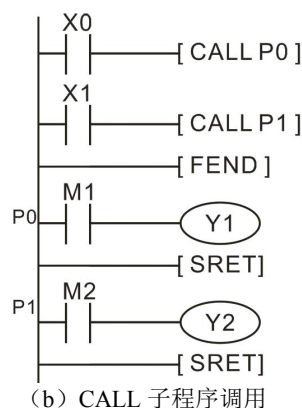
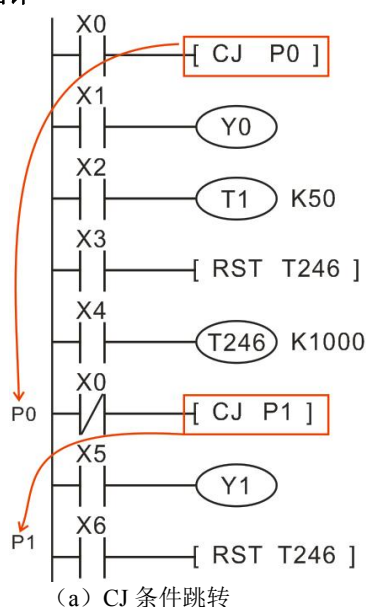
3.9 指针 P · I

指针（P）用于跳转程序的入口地址与子程序的起始地址标识；指针（I）则用于中断程序的起始地址标识，其编号采用十进制数分配，如下表所示：

型号	分支用		输入中断用				定时器中断用	计数器中断用
	无	结束跳转用					单位：ms	
LX3V (1S 指令集)	P0-P62 63 点	P63 1 点	I00_(X000) I30_(X003)	I10_(X001) I40_(X004)	I20_(X002) I50_(X005)	6 点	—	—
LX3V (2N 指令集)	P0-P62 P64-P127 127 点	P63 1 点	I00_(X000) I30_(X003)	I10_(X001) I40_(X004)	I20_(X002) I50_(X005)	6 点	I6□□ I7□□ I8□□ 3 点	I010 I020 I030 I040 I050 I060 6 点
LX3VP	P0-P62 P64-P127 127 点	P63 1 点	I00_(X000) I30_(X003)	I10_(X001) I40_(X004)	I20_(X002) I50_(X005)	6 点	I6□□ I7□□ I8□□ 3 点	I010 I020 I030 I040 I050 I060 6 点
LX3VE	P0-P62 P64-P127 127 点	P63 1 点	I00_(X000) I30_(X003)	I10_(X001) I40_(X004)	I20_(X002) I50_(X005)	6 点	I6□□ I7□□ I8□□ 3 点	I010 I020 I030 I040 I050 I060 6 点

注：在使用输入中断用指针时，给其分配的输入编号不能与采用相同输入范围的【高速计数器】或【脉冲密度】等一起使用。

(1) 分支用指针



(2) 中断用指针

中断用指针有三种类型，与应用指令 IRET（中断返回）、开中断（EI）和关中断（DI）一起使用。

- (a) 输入中断用：接收来自特定输入信号，而不受可编程控制器的扫描周期的影响。
- (b) 定时器中断用：在各指定的中断循环时间（10ms~99ms）执行中断子程序。
- (c) 计数器中断用：根据可编程控制器内置的高速计数器的比较结果，执行中断子程序。

3.10 常数 K · H · E

LX 系列可编程控制器根据不同的用途和目的，使用 5 种类型的数值。

其作用和功能如下：

类型	编程中应用说明
十进制数(DEC)	定时器和计数器的设定值 (K 常数) 辅助继电器 (M)，定时器 (T)，计数器 (C)，状态 S 等的编号 (软元件编号) 指定应用指令操作数中的数值与指令动作 (K 常数)；
十六进制数(HEX)	同 10 进制数一样，用于指定应用指令中的操作数与指定动作 (H 常数)；
二进制(BIN)	以十进制数或十六进制数对定时器、计数器或数据寄存器进行数值指定，但在可编程控制器内部，这些数字都用二进制数处理。而且，在外围设备上实时监控时，这些软元件将如图所示自动变换为十进制数（也可切换为 16 进制）；
八进制(OTC)	输入继电器、输出继电器的软元件编号以 8 进制数值进行分配。因此，可进行 [0-7，10-17，… 70-77，100-107] 的进位，在 8 进制数中，不存在 [8，9]；
BCD	BCD 是以 4 位二进制表示十进制数各位 0-9 数值的方法。各位的处理很容易，因此，可用于 BCD 输出形的数字式开关或七段码的显示器控制等方面；
BIN 浮点数	可编程控制器具有可进行高精度的浮点运算功能，内部用二进制 (BIN) 浮点数进行浮点运算；
十进制浮点数	十进制浮点值只用于监视，便于阅读。

常数 K

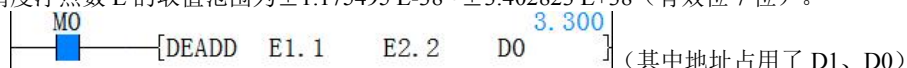
[K]是表示 10 进制整数的符号。主要用于指定定时器或计数器的设定值或应用指令操作数中的数值。16bit 指令中，常数 K 的取值范围为-32768~32767；32bit 指令中，常数 K 的取值范围为-2, 47, 483, 648~2, 147, 483, 647。

常数 H

[H]是 16 进制数的表示符号。主要用于指定应用指令的操作数的数值。常数 H 的取值范围为 0000~FFFF；32bit 指令中，常数 K 的取值范围为 0000, 0000~FFFF, FFFF。

常数 E

[E] 是单精度浮点数表示符号。主要用于指定应用指令的操作数的数值，该功能目前仅适用于 3VP 和 3VE 中的下列指令：DECMP、DEZCP、DSINH、DCOSH、DTANH、DEBCD、DEBIN、DEADD、DESUB、DEMUL、DEDIV、DEXP、DLOGE、DLOG10、DESQR、DINT、DSIN、DCOS、DTAN、DASIN、DACOS、DATAN、DRAD、DDEG。单精度浮点数 E 的取值范围为±1.175495 E-38~±3.402823 E+38（有效位 7 位）。



3.11 特殊软元件一览表

M8000~M8255, D8000~D8255 被定义为特殊元件种类, 其功能如下表所述:

M8000~M8255; D8000~D8255 被定义为特殊元件种类，其功能如下表所述：							
M 元件	M 元件的描述	适用机型		D 元件	D 元件的描述	适用机型	
		LX3V(1S 指令集)	LX3V/LX3 VP/LX3VE			LX3V (1S 指令集)	LX3V/LX3 VP/LX3VE
系统运行状态							
M8000	用户程序运行时置为 ON 状态	○	○	D8000	用户程序运行的监视定时器	○	○
M8001	M8000 状态取反	○	○	D8001	固件主版本号，其中 250**表示 3V/3V-A2 的型号；220**表示 3V-A1 的型号；251**表示 3VP 的型号；240**表示 2V 的型号。 **查看 D8101 次版本号。	○	○
M8002	用户程序开始运行的第一个周期为 ON	○	○	D8002	程序容量，2K，4K，8K，16K 等	○	○
M8003	M8002 状态取反	○	○	D8003	固定为 0X10，为可编程控制器内存储	○	○
M8004	当 M8060~M8067[除 M8062]中任意一个处于 ON，则 M8004 有效	○	○	D8004	错误的 M8060~M8067 的 BCD 值，正常为 0	○	○
M8005	电池电压过低时动作	—	○	D8005	电池电压的 BCD 当前值	—	○
M8006	电池电压低有出现就动作 [锁存]	—	○	D8006	电池电压过低的检测值，初始值为 2.6V	—	○
M8007	当交流失电 5mS 后 M8007&M8008 动作，但在 D8008 之内程序继续运行	—	○	D8007	保存 M8007 动作的次数，当失电时该单元清 0 处理	—	○
M8008	当 D8008 时间内都失电，当 M8008 由 ON→OFF 时，用户程序不运行。M8000 为 OFF	—	○	D8008	交流失电检测时间，默认为 10mS	—	○
M8009	扩展单元 24V 掉电时动作	—	○	D8009	扩展单元 24V 失电的模块号	—	○
系统时钟							
M8010	保留	○	○	D8010	当前扫描时间，从用户程序 0 步开始(0.1mS)	○	○
M8011	10mS 时钟周期的振荡时钟	○	○	D8011	扫描时间的最小值 (0.1mS)	○	○
M8012	100mS 时钟周期的振荡时钟	○	○	D8012	扫描时间的最大值 (0.1mS)	○	○
M8013	1S 时钟周期的振荡时钟	○	○	D8013	时钟秒(0~59)	○	○
M8014	1 分钟时钟周期的振荡时钟	○	○	D8014	实时时钟分(0~59)	○	○
M8015	时钟停止和预置	○	○	D8015	实时时钟小时(0~23)	○	○
M8016	时钟读取显示停止	○	○	D8016	实时时钟日(1~31)	○	○
M8017	±30 秒修正	○	○	D8017	实时时钟月(1~12)	○	○
M8018	安装检测	○	○	D8018	实时时钟公历年 (2000~2099)	○	○
M8019	实时时钟(RTC)出错	○	○	D8019	实时时钟星期	○	○
指令标志							
M8020	运算零标志	○	○	D8020	X 点的输入滤波常数 0~60[默认 10mS]	○	○
M8021	运算借位标志	○	○	D8021	保留		
M8022	运算进位标志	○	○	D8022	保留		
M8023	保留	○	○	D8023	保留		
M8024	BMOV 指令的方向	—	○	D8024	保留		
M8025	HSC 指令模式	—	○	D8025	保留		
M8026	RAMP 指令模式	—	○	D8026	保留		
M8027	PR 模式	—	○	D8027	保留		

M8028 (LX1S)	100MS/10MS 定时器切换						
M8028	当 M8028 置 ON 时, 将 T50~T191 切换为 1MS 非累计型定时器, 仅对 3VP 系列及以上版本的 PLC 有效。	○	—	D8028	和 Z0 同一变量地址	○	○
M8029	部分指令 (PLSR 等) 指令执行完成	○	○	D8029	和 V0 同一变量地址	○	○
系统模式							
M8030	为 ON 时, 屏蔽电池低告警	—	○	D8030	保留		
M8031	为 ON 时, 清除所有非保存存储器	○	○	D8031	保留		
M8032	为 ON 时, 清除所有保存存储器	○	○	D8032	保留		
M8033	为 ON 时, 停机状态所有的软元件不变	○	○	D8033	保留		
M8034	为 ON 时, PLC 所有的输出都为 OFF 状态	○	○	D8034	保留		
M8035	强制运行命令 1	○	○	D8035	保留		
M8036	强制运行命令 2	○	○	D8036	保留		
M8037	强制停止命令	○	○	D8037	保留		
M8038	通讯参数设定标志	○	○	D8038	保留		
M8039	恒定扫描控制	○	○	D8039	恒定扫描时间, 默认 0, 以 mS 为单位	○	○
步进阶梯							
M8040	转移禁止	○	○	D8040	将 S0~S899 的最小动作地址号保存在 D8040 中, 其它依次, 最大的地址号保存在 D8047 中。	○	○
M8041	转移开始	○	○	D8041		○	○
M8042	对应启动输入的脉冲输出	○	○	D8042		○	○
M8043	原点回归状态的结束标志	○	○	D8043		○	○
M8044	检测到机械原点动作	○	○	D8044		○	○
M8045	所有输出复位禁止	○	○	D8045		○	○
M8046	M8047 动作后, 当 S0~S999 中任何一个为 ON, M8046 为 ON	○	○	D8046		○	○
M8047	STL 监视有效 [D8040~D8047 有效]	○	○	D8047		○	○
M8048	M8049=ON, S900~S999 任何一个有效, M8048 有效	—	○	D8048	保留		
M8049	信号报警有效, [D8049 有效]	—	○	D8049	保存 S900~S999 的报警最小地址号	—	○
中断禁止							
M8050	驱动 I00□中断禁止	○	○	D8050	保留		
M8051	驱动 I10□中断禁止	○	○	D8051	保留		
M8052	驱动 I20□中断禁止	○	○	D8052	保留		
M8053	驱动 I30□中断禁止	○	○	D8053	保留		
M8054	驱动 I40□中断禁止	○	○	D8054	保留		
M8055	驱动 I50□中断禁止	○	○	D8055	保留		
M8056	驱动 I6□□中断禁止	—	○	D8056	保留		
M8057	驱动 I7□□中断禁止	—	○	D8057	保留		
M8058	驱动 I8□□中断禁止	—	○	D8058	保留		
M8059	驱动 计数器中断禁止	—	○	D8059	保留		
系统错误检测							
M8060	I/O 构成错误[]	—	○	D8060	I/O 构成错误的未安装 I/O 起始地址号	—	○
M8061	PLC 硬件错误	○	○	D8061	PLC 硬件错误的错误代码序号	○	○
M8062	PLC 通讯错误	—	○	D8062	PLC 通讯错误代码	—	○
M8063	模块通讯标志位	○	○	D8063	模块通讯错误码	○	○
M8064	参数错误	○	○	D8064	参数错误代码	○	○

M8065	语法错误	○	○	D8065	语法错误的代码	○	○
M8066	回路错误	○	○	D8066	回路错误的代码	○	○
M8067	运算错误	○	○	D8067	运算错误的代码	○	○
M8068	运算错误锁存	○	○	D8068	锁存运算错误程序的步号	○	○
M8069	保留			D8069	M8065~M8067 的错误发生的步号	○	○
高速环形计数器							
M8099	高速环形计数器计数启动	○	○	D8099	[0~32767]上升动作环形计数器(0.1mS)	○	○
其他功能使用							
M8100	SPD(X000)具有-脉冲个数/分钟	○	○	D8100	保留	○	○
M8101	SPD(X001)具有-脉冲个数/分钟	○	○	D8101	固件次版本号，其中160**表示 3V/3V-A2/3V-A1/3VP 的型号，240**表示 2V 的型号。D8101 的后面两位数结合固件主版本号 D8001 中的**构成完整的固件版本号。	○	○
M8102	SPD(X002)具有-脉冲个数/分钟	○	○	D8102	系统提供给用户程序的程序容量	○	○
M8103	SPD(X003)具有-脉冲个数/分钟	○	○	D8103	保留	○	○
M8104	SPD(X004)具有-脉冲个数/分钟	○	○	D8104	DRVI, DRVA 执行时加减速时间[默认 100]由 M8135 决定是否有效[Y0]，必须确保数值和 D8165 一致。	○	○
M8105	SPD(X005)具有-脉冲个数/分钟	○	○	D8105	DRVI, DRVA 执行时加减速时间[默认 100]由 M8135 决定是否有效[Y1]，必须确保数值和 D8166 一致。	○	○
M8106	保留			D8106	DRVI, DRVA 执行时加减速时间[默认 100]由 M8135 决定是否有效[Y2]，必须确保数值和 D8167 一致。	○	○
M8107	保留			D8107	DRVI, DRVA 执行时加减速时间[默认 100]由 M8135 决定是否有效[Y3]，必须确保数值和 D8168 一致。	○	○
M8108	保留			D8108	保留	○	○
M8109	输出刷新错误			D8109	输出刷新错误的输出地址编号	○	○
COM1 通讯、链接							
M8110	保留			D8110	Com1 口通讯格式，界面配置设定，默认为 0(版本 22319, 24320, 25007 以上才有效)	○	○
M8111	保留			D8111	保留		
M8112	BD 模块 1 通道 1 标志位			D8112	BD 模块 1 通道 1 数据		
M8113	BD 模块 1 通道 2 标志位			D8113	BD 模块 1 通道 2 数据		
M8114	BD 模块 1 通道 3 标志位			D8114	BD 模块 1 通道 3 数据		
M8115	BD 模块 1 通道 4 标志位			D8115	BD 模块 1 通道 4 数据		
M8116	BD 模块 2 通道 1 标志位			D8116	BD 模块 2 通道 1 数据		
M8117	BD 模块 2 通道 2 标志位			D8117	BD 模块 2 通道 2 数据		

M8118	BD 模块 2 通道 3 标志位			D8118	BD 模块 2 通道 3 数据		
M8119	BD 模块 2 通道 4 标志位			D8119	BD 模块 2 通道 4 数据		
COM2 通讯、链接							
M8120	保留			D8120	Com2 口通讯格式, 界面配置设定, 默认为 0	○	○
M8121	发送等待中 (RS 指令)	○	○	D8121	站号设置, 界面配置设定, 默认为 1	○	○
M8122	发送标志 (RS 指令) 指令执行状态 (MODBUS)	○	○	D8122	传送剩余数据数量 (仅对 RS 指令) 单位为 0.1ms。Modbus: 指令发送间隔, 0=5ms	○	○
M8123	接收完成标志 (RS) 通讯错误标志 (MODBUS)	○	○	D8123	接收到的数据数量 (仅对 RS 指令)	○	○
M8124	接收中 (仅对 RS 指令)	○	○	D8124	起始字符 STX (仅对 RS 指令)	○	○
M8125	保留			D8125	终止字符 ETX (仅对 RS 指令)	○	○
M8126	保留			D8126	通讯协议设定, 界面配置设定, 默认为 0	○	○
M8127	保留			D8127	计算机链接协议接通要求数据起始地址号	○	○
M8128	保留			D8128	计算机链接协议接通要求发送数据数量	○	○
M8129	超时判断	○	○	D8129	通讯超时时间判断, 界面配置设定, 默认为 10 (100ms)	○	○
高速&定位							
M8130	HSZ 指令平台的控制模式	○	○	D8130	HSZ 高速比较平台使用 (记录号)	○	○
M8131	和 M8130 联合使用	○	○	D8131	HSZ&PLSY 速度模型使用 (记录号)	○	○
M8132	HSZ&PLSY 速度模式	○	○	D8132	HSZ&PLSY 速度模型频率	低位 空	○ ○
M8133	和 M8132 联合使用	○	○	D8133	使用		
M8134	Y0 的千分比控制位			D8134	HSZ&PLSY 速度模型比较脉冲数使用	低位	○ ○
M8135	PWM: Y1 的千分比控制位 定位: 各轴之间的加减速时间分开标志位			D8135		高位	
M8136	PWM: Y2 的千分比控制位 定位: 每轴的加速和减速时间分开标志位 (M8135=ON 时有效)			D8136	Y000&Y001 输出脉冲合计数	低位	○ ○
M8137	Y3 的千分比控制位			D8137		高位	
M8138	保留			D8138	保留		
M8139	保留			D8139	保留		
M8140	ZRN 的 CLR 信号输出功能有效	○	○	D8140	PLSY&PLSR 输出 Y000 对应的脉冲个数累积值	低位	○ ○
M8141	当 M8141 置 ON 时, 将控制输出脉冲累积寄存器 (D8136D813, D8140D8141, D8142D8143, D8150D8151, D8152D8153) 掉电保存。	○	○	D8141		高位	
M8142	保留			D8142	PLSY&PLSR 输出 Y001 对应的脉冲个数累积值	低位	○ ○
M8143	保留			D8143		高位	
M8144	保留			D8144			
M8145	Y000 脉冲输出停止	○	○	D8145	DRVI, DRVA 执行时的偏置速度	○	○

M8146	Y001 脉冲输出停止	○	○	D8146	DRVI, DRVA 执行时的最高速度[默认 100, 000]	低位	○	○
M8147	Y000 脉冲输出监控	○	○	D8147		高位		
M8148	Y001 脉冲输出监控	○	○	D8148	DRVI, DRVA 执行时加减速时间[默认 100]		○	○
M8149	Y002 脉冲输出监控	○	○	D8149	保留			
M8150	Y003 脉冲输出监控	○	○	D8150	PLSY&PLSR 输出 Y002 对应的脉冲个数累积值	低位	○	○
M8151				D8151		高位		
M8152	Y002 脉冲输出停止	○	○	D8152	PLSY&PLSR 输出 Y003 对应的脉冲个数累积值	低位	○	○
M8153	Y003 脉冲输出停止	○	○	D8153		高位		
M8154	保留			D8154	保留			
M8155	保留			D8155	保留			
M8156	保留			D8156	保留		○	○
M8157	保留			D8157	保留		○	○
扩展功能								
M8158	保留			D8158	保留		○	○
M8159	保留			D8159	保留		○	○
M8160	(XCH)的 SWAP 功能	—	○	D8160	保留		○	○
M8161	ASC/RS/ASCII/HEX/CCD 的位处理模式	○	○	D8161	保留			
M8162	高速并联连接模式	○	○	D8162	保留			
M8163	保留			D8163	保留			
M8164	(FROM/TO)传送点数可变模式	—	○	D8164	(FROM/TO)传送点数指定模式		○	○
M8165	保留			D8165	Y0 的加减速时间, 必须确保数值和 D8104 一致。M8136=ON 时, 为 Y0 的减速时间		○	○
M8166	保留			D8166	Y1 的加减速时间, 必须确保数值和 D8105 一致。M8136=ON 时, 为 Y1 的减速时间		○	○
M8168	(SMOV)HEX 数据处理功能	—	○	D8168	Y2 的加减速时间, 必须确保数值和 D8106 一致。M8136=ON 时, 为 Y2 的减速时间		○	○
M8167	(HEY)HEX 数据处理功能	—	○	D8167	Y3 的加减速时间, 必须确保数值和 D8107 一致。M8136=ON 时, 为 Y3 的减速时间		○	○
M8169	保留			D8169	保留			
脉冲捕捉				通讯、链接				
M8170	X000 脉冲捕捉	○	○	D8170	扩展口 1 的 BD 模块型号			
M8171	X001 脉冲捕捉	○	○	D8171	扩展口 2 的 BD 模块型号			
M8172	X002 脉冲捕捉	○	○	D8172	自定义协议下的字符超时			
M8173	X003 脉冲捕捉	○	○	D8173	本站站号设定状态		○	○
M8174	X004 脉冲捕捉	○	○	D8174	通讯子站设定状态		○	○
M8175	X005 脉冲捕捉	○	○	D8175	刷新范围设定状态		○	○
M8176	保留			D8176	本站站号设定		○	○
M8177	保留			D8177	通讯子站数设定		○	○
M8178	保留			D8178	刷新范围设定		○	○
M8179	485BD 模块通道设定			D8179	重试次数设定		○	○
M8180	保留			D8180	通信超时设置		○	○

通讯、连接				变址寻址			
M8181	保留			D8181	保留		
M8182	保留			D8182	位元件地址号 No. 2/Z1 寄存器内容	○	○
M8183	数据传送主站出错	○	○	D8183	位元件地址号 No. 3/V1 寄存器内容	○	○
M8184	数据传送从站 1 出错	○	○	D8184	位元件地址号 No. 4/Z2 寄存器内容	○	○
M8185	数据传送从站 2 出错	○	○	D8185	位元件地址号 No. 5/V2 寄存器内容	○	○
M8186	数据传送从站 3 出错	○	○	D8186	位元件地址号 No. 6/Z3 寄存器内容	○	○
M8187	数据传送从站 4 出错	○	○	D8187	位元件地址号 No. 7/V3 寄存器内容	○	○
M8188	数据传送从站 5 出错	○	○	D8188	位元件地址号 No. 8/Z4 寄存器内容	○	○
M8189	数据传送从站 6 出错	○	○	D8189	位元件地址号 No. 9/V4 寄存器内容	○	○
M8190	数据传送从站 7 出错	○	○	D8190	位元件地址号 No. 10/Z5 寄存器内容	○	○
M8191	数据传送进行中	○	○	D8191	位元件地址号 No. 11/V5 寄存器内容	○	○
M8192	保留			D8192	位元件地址号 No. 12/Z6 寄存器内容	○	○
M8193	保留			D8193	位元件地址号 No. 13/V6 寄存器内容	○	○
M8194	保留			D8194	位元件地址号 No. 14/Z7 寄存器内容	○	○
M8195	保留	○	○	D8195	位元件地址号 No. 15/V7 寄存器内容	○	○
M8196	保留	○	○	D8196	保留		
M8197	保留	○	○	D8197	保留		
M8198	保留	○	○	D8198	保留		
M8199	保留	○	○	D8199	保留		
计数器增/减控制或状态				通讯、链接			
M8200	C200 控制	○	○	D8200	寄存器 C251 倍频功能，当 D8200 的值为不同数值，代表着不同的倍频，其中 0: 1 倍频; 1: 2 倍频; 2: 4 倍频。注意: HSCS、HSCR、HSCZ 不能与倍频同时使用，而且版本 V311 以上的才能支持倍频功能。	○	○
M8201	C201 控制	○	○	D8201	当前链接扫描时间		
M8202	C202 控制	○	○	D8202	最大链接的扫描时间		
M8203	C203 控制	○	○	D8203	主站错误计数		
M8204	C204 控制	○	○	D8204	从站 1 的错误计数		
M8205	C205 控制	○	○	D8205	从站 2 的错误计数		
M8206	C206 控制	○	○	D8206	从站 3 的错误计数		
M8207	C207 控制	○	○	D8207	从站 4 的错误计数		
M8208	C208 控制	○	○	D8208	从站 5 的错误计数		
M8209	C209 控制	○	○	D8209	从站 6 的错误计数		
M8210	C210 控制	○	○	D8210	从站 7 的错误计数		
M8211	C211 控制	○	○	D8211	主站的 N: N 错误码		
M8212	C212 控制	○	○	D8212	从站 1 的错误码		
M8213	C213 控制	○	○	D8213	从站 2 的错误码		
M8214	C214 控制	○	○	D8214	从站 3 的错误码		
M8215	C215 控制	○	○	D8215	从站 4 的错误码		
M8216	C216 控制	○	○	D8216	从站 5 的错误码		

M8217	C217 控制	○	○	D8217	从站 6 的错误码		
M8218	C218 控制	○	○	D8218	从站 7 的错误码		
M8219	C219 控制	○	○	D8219	保留		
M8220	C220 控制	○	○	D8220	D8220=1: 新的滤波方式, 4 点为一组的滤波, 使用此滤波方法时 D8020 设置的滤波时间无效; 而且使用此滤波方式之前要先设置好主机上的 Xn 点的滤波时间(即 D8221-D8228)。滤波时间单位 ms。 注意: 此滤波方式只对主机上的 IO 有效, 扩展 IO 无效。	○	○
M8221	C221 控制	○	○	D8221	低 8 位 为 X0~X3 的滤波时间; 高 8 位为 X4~7 的滤波时间, 单位 ms	○	○
M8222	C222 控制	○	○	D8222	低 8 位 为 X10~X13 的滤波时间; 高 8 位为 X14~17 的滤波时间, 单位 ms	○	○
M8223	C223 控制	○	○	D8223	低 8 位 为 X20~X23 的滤波时间; 高 8 位为 X24~27 的滤波时间, 单位 ms	○	○
M8224	C224 控制	○	○	D8224	低 8 位 为 X30~X33 的滤波时间; 高 8 位为 X34~37 的滤波时间, 单位 ms	○	○
M8225	C225 控制	○	○	D8225	低 8 位 为 X40~X43 的滤波时间; 高 8 位为 X44~47 的滤波时间, 单位 ms	○	○
M8226	C226 控制	○	○	D8226	低 8 位 为 X50~X53 的滤波时间; 高 8 位为 X54~57 的滤波时间, 单位 ms	○	○
M8227	C227 控制	○	○	D8227	低 8 位 为 X60~X63 的滤波时间; 高 8 位为 X64~67 的滤波时间, 单位 ms	○	○
M8228	C228 控制	○	○	D8228	低 8 位 为 X70~X73 的滤波时间; 高 8 位为 X74~77 的滤波时间, 单位 ms	○	○
M8229	C229 控制	○	○	D8229	保留		
M8230	C230 控制	○	○	D8230	保留		
M8231	C231 控制	○	○	D8231	保留		
M8232	C232 控制	○	○	D8232	保留		
M8233	C233 控制	○	○	D8233	保留		
M8234	C234 控制	○	○	D8234	保留		
M8235	单相单计数输入	C235 控制	○	○	D8235	保留	
M8236		C236 控制	○	○	D8236	保留	
M8237		C237 控制	○	○	D8237	保留	
M8238		C238 控制	○	○	D8238	保留	
M8239		C239 控制	○	○	D8239	保留	
M8240		C240 控制	○	○	D8240	保留	
M8241		C241 控制	○	○	D8241	保留	

M8242		C242 控制	○	○	D8242	保留		
M8243		C243 控制	○	○	D8243	保留		
M8244		C244 控制	○	○	D8244	保留		
M8245		C245 控制	○	○	D8245	保留		
M8246	单相双计数输入	C246 控制	○	○	D8246	保留		
M8247		C247 控制	○	○	D8247	保留		
M8248		C248 控制	○	○	D8248	保留		
M8249		C249 控制	○	○	D8249	保留		
M8250		C250 控制	○	○	D8250	保留		
M8251	双相双计数输入	C251 控制	○	○	D8251	保留		
M8252		C252 控制	○	○	D8252	保留		
M8253		C253 控制	○	○	D8253	保留		
M8254		C254 控制	○	○	D8254	保留		
M8255		C255 控制	○	○	D8255	保留		

注：符号“○”表示适用，符号“—”表示不适用。

4.指令一览表

4.1 以指令分类为序

指令类别	指令	指令功能说明	LX3V (1S 指令集)	LX3V (2N 指令集)	LX3VP	LX3VE	页码
基本指令	LD	常开触点能流装载	○	○	○	○	37
	LDI	常闭触点能流装载	○	○	○	○	37
	LDP	上升沿检出运算开始	○	○	○	○	37
	LDF	下降沿检出运算开始	○	○	○	○	37
	OUT	线圈驱动指令	○	○	○	○	37
	AND	常开触点能流与	○	○	○	○	38
	ANI	常闭触点能流与	○	○	○	○	38
	ANDP	上升沿检出串联连接	○	○	○	○	38
	ANDEF	下降沿检出串联连接	○	○	○	○	38
	OR	常开触点能流或	○	○	○	○	38
	ORI	常闭触点能流或	○	○	○	○	38
	ORP	上升沿检出并联连接	○	○	○	○	38
	ORF	下降沿检出并联连接	○	○	○	○	38
	ANB	并联回路块的串联连接	○	○	○	○	39
	ORB	串联回路块的并联连接	○	○	○	○	39
	INV	运算结果取反	○	○	○	○	39
	MC	主控	○	○	○	○	40
	MCR	主控清除	○	○	○	○	40
	MPS	运算存储	○	○	○	○	40
	MRD	存储读出	○	○	○	○	40
	MPP	存储读出与复位	○	○	○	○	40
	PLS	上升沿检出指令	○	○	○	○	42
	PLF	下降沿检出指令	○	○	○	○	42
	SET	线圈置位	○	○	○	○	42
	RST	线圈复位	○	○	○	○	42
步进梯形图指令	STL	步进梯形图开始	○	○	○	○	222
	RET	步进梯形图结束	○	○	○	○	222

指令类别	指令	指令功能说明	LX3V (1S 指令集)	LX3V (2N 指令集)	LX3VP	LX3VE	页码
程序流程指令	CJ	条件跳转	○	○	○	○	43
	CALL	子程序调用	○	○	○	○	44
	EI	中断许可	○	○	○	○	45
	DI	中断禁止	○	○	○	○	45
	WDT	监视定时器	○	○	○	○	47
	FOR	循环范围开始	○	○	○	○	48
	NEXT	循环范围结束	○	○	○	○	48
传送与比较指令	CMP	比较指令	○	○	○	○	49
	ZCP	区间比较	○	○	○	○	50
	MOV	传送指令	○	○	○	○	51
	SMOV	移位传送	——	○	○	○	52
	CML	取反传送	——	○	○	○	53
	BMOV	成批传送	○	○	○	○	54
	FMOV	多点传送	——	○	○	○	54
	XCH	交换指令	——	○	○	○	55
	BCD	BCD 交换	○	○	○	○	56
	BIN	BIN 交换	○	○	○	○	56
数据处理指令	ZRST	批量复位	○	○	○	○	57
	DECO	解码	○	○	○	○	58
	ENCO	编码	○	○	○	○	59
	SUM	ON 位统计指令	——	○	○	○	60
	BON	ON 位判断指令	——	○	○	○	60
	MEAN	平均值指令	——	○	○	○	61
	ANS	信号报警器置位	——	○	○	○	61
	ANR	信号报警器复位	——	○	○	○	62
	SQR	整数开方	——	○	○	○	62
	FLT	整数转换成二进制浮点	——	○	○	○	63
	SWAP	字节高低位变换	——	○	○	○	64

指令类别	指令	指令功能说明	LX3V (1S 指令集)	LX3V (2N 指令集)	LX3VP	LX3VE	页码
时钟运算	TCMP	时钟数据比较	○	○	○	○	65
	TZCP	时钟数据区间比较	○	○	○	○	66
	TADD	时钟数据加法运算	○	○	○	○	67
	TSUB	时钟数据减法运算	○	○	○	○	68
	TRD	时钟数据读取	○	○	○	○	69
	TWR	时钟数据写入	○	○	○	○	70
	HOUR	16 位计时表指令	○	○	○	○	71
四则运算	ADD	整数加法	○	○	○	○	72
	SUB	整数减法	○	○	○	○	73
	MUL	整数乘法	○	○	○	○	74
	DIV	整数除法	○	○	○	○	75
	INC	整数加 1	○	○	○	○	76
	DEC	整数减 1	○	○	○	○	76
	WAND	逻辑按位与	○	○	○	○	77
	WOR	逻辑按位或	○	○	○	○	77
	WXOR	逻辑按位异或	○	○	○	○	78
	NEG	求补指令	——	○	○	○	79
高速处理指令	REF	输入输出刷新	○	○	○	○	80
	REFF	滤波器调整	——	○	○	○	81
	MTR	矩形输入	○	○	○	○	82
	DHSCR	高速计数比较复位	○	○	○	○	83
	DHSCS	高速计数比较置位	○	○	○	○	84
	DHSZ	高速计数区间比较	——	○	○	○	86
	SPD	脉冲密度	○	○	○	○	87
	PLSY	脉冲输出	○	○	○	○	88
	PWM	脉宽调制	○	○	○	○	89
	PWM	千分比模式	○	○	○	○	89
	PLSR	带加减速脉冲输出	○	○	○	○	90
	PLSR2	多段速脉冲输出	○	○	○	○	92
	PTO	包络脉冲输出指令	○	○	○	○	100

指令类别	指令	指令功能说明	LX3V (1S 指令集)	LX3V (2N 指令集)	LX3VP	LX3VE	页码
循环移位指令	ROL	16 位循环左移	○	○	○	○	102
	ROR	16 位循环右移	○	○	○	○	103
	RCL	16 位带进位循环左移	——	○	○	○	104
	RCR	16 位带进位循环右移	——	○	○	○	105
	SFTL	位左移	○	○	○	○	106
	SFTR	位右移	○	○	○	○	107
	WSFL	字左移	——	○	○	○	108
	WSFR	字右移	——	○	○	○	109
	SFRD	移位读出	○	○	○	○	110
	SFWR	移位写入	○	○	○	○	111
外部设备 IO 指令	TKY	十字键输入	——	○	○	○	112
	HKY	16 键输入	——	○	○	○	113
	DSW	数字开关	○	○	○	○	114
	SEGD	7 段码译码	——	○	○	○	115
	SEGL	7 段码按时间分割显示	○	○	○	○	116
	ARWS	方向开关	——	○	○	○	116
	ASC	ASCII 码转换	——	○	○	○	118
	PR	ASCII 码打印输出	——	○	○	○	119
	FROM	特殊模块 16 位 BFM 读出	○	○	○	○	120
	TO	特殊模块 16 位 BFM 写入	○	○	○	○	121
	GRY	16 位格雷码转换	——	○	○	○	122
	GBIN	16 位格雷码逆转换	——	○	○	○	122
电子凸轮指令	DECAM	电子凸轮配置指令	——	——	——	○	123
	DEGEAR	电子齿轮/手摇轮指令	——	——	——	○	126
	ECAMTBX	电子凸轮曲线生成指令	——	——	——	○	129

指令类别	指令	指令功能说明	LX3V (1S 指令集)	LX3V (2N 指令集)	LX3VP	LX3VE	页码
方便指令	IST	状态初始化	○	○	○	○	130
	SER	16 位数据查找	——	○	○	○	137
	ABSD	凸轮控制(绝对方式)	○	○	○	○	134
	INCD	凸轮控制(增量方式)	○	○	○	○	135
	TTMR	示教定时器	——	○	○	○	136
	STMR	特殊定时器	——	○	○	○	137
	ALT	交替输出	○	○	○	○	138
	RAMP	斜坡信号	○	○	○	○	140
	ROTC	旋转工作台控制	——	○	○	○	141
	SORT	数据排序	——	○	○	○	142
定位指令	DABS	ABS 当前值读取	○	○	○	○	144
	ZRN	原点回归指令	○	○	○	○	145
	PLSV	可变速脉冲输出指令	○	○	○	○	147
	DRVI	相对位置控制指令	○	○	○	○	148
	DRVA	绝对位置控制指令	○	○	○	○	150
	DVIT	中断定位	——	——	○	○	152
	SCL	定坐标	——	——	——	——	154
	G90G01	绝对位置直线插补	——	——	——	○	157
	G91G01	相对位置直线插补	——	——	——	○	159
	G90G02	绝对位置顺圆插补	——	——	——	——	161
	G91G02	相对位置顺圆插补	——	——	——	——	163
	G90G03	绝对位置逆圆插补	——	——	——	——	165
	G91G03	相对位置逆圆插补	——	——	——	——	167
外部设备 SER 指令	RS	串行数据传送	○	○	○	○	169
	RS2	串行数据传送 2	○	○	○	○	172
	RSLIST	表格化通信指令	○	○	○	○	174
	CPAVL (串口通讯)	通讯串口参数设置	○	○	○	○	181
	CPAVL(以 以太网通讯)	通讯串口参数设置	○	○	○	○	183
	PRUN	16 位 8 进制位传送	○	○	○	○	186

指令类别	指令	指令功能说明	LX3V (1S 指令集)	LX3V (2N 指令集)	LX3VP	LX3VE	页码
外部设备 SER 指令	ASCI	HEX 向 ASCII 转换	○	○	○	○	187
	HEX	ASCII 向 HEX 转换	○	○	○	○	189
	CCD	校验码	○	○	○	○	191
	CCPID	PID 运算	——	——	——	——	192
	PID	PID 运算	○	○	○	○	200
浮点运算	DECMP	二进制浮点数比较	——	○	○	○	202
	DEZCP	二进制浮点数区间比较	——	○	○	○	203
	DEBCD	浮点数二进制向十进制转换	——	○	○	○	203
	DEBIN	浮点数十进制向二进制转换	——	○	○	○	204
	DEADD	二进制浮点数加法	——	○	○	○	204
	DESUB	二进制浮点数减法	——	○	○	○	205
	DEMUL	二进制浮点数乘法	——	○	○	○	206
	DEDIV	二进制浮点数除法	——	○	○	○	207
	DESQR	二进制浮点数开方	——	○	○	○	208
	INT	二进制浮点数转换为整数	——	○	○	○	208
	DSIN	浮点数 SIN 运算	——	○	○	○	209
	DCOS	浮点数 COS 运算	——	○	○	○	210
	DTAN	浮点数 TAN 运算	——	○	○	○	210
	DASIN	根据 SIN 值计算对应弧度值	——	——	○	○	211
	DACOS	根据 COS 值计算对应弧度值	——	——	○	○	212
	DATAN	根据 TAN 值计算对应弧度值	——	——	○	○	213
	DSINH	2 进制浮点数双曲正弦函数 SINH 运算	——	——	○	○	214
	DCOSH	2 进制浮点数双曲余弦函数 COSH 运算	——	——	○	○	214
	DTANH	2 进制浮点数双曲正切函数 TANH 运算	——	——	○	○	215
	DDEG	将 2 进制浮点弧度转成角度	——	——	○	○	215
	DRAD	将 2 进制浮点数角度转成弧度	——	——	○	○	216
	DEXP	以 e 为底数的指数运算	——	——	○	○	217
	DLOG10	以 10 为底数的对数运算	——	——	○	○	217
	DLOGE	以 e 为底数的对数运算	——	——	○	○	218

指令类别	指令	指令功能说明	LX3V (1S 指令集)	LX3V (2N 指令集)	LX3VP	LX3VE	页码
触点比较指令	<u>LD=</u>	数等于比较	○	○	○	○	219
	<u>LD></u>	数大于比较	○	○	○	○	
	<u>LD<</u>	数小于比较	○	○	○	○	
	<u>LD<></u>	数不等于比较	○	○	○	○	
	<u>LD<=</u>	数小于等于比较	○	○	○	○	
	<u>LD>=</u>	数大于等于比较	○	○	○	○	
	<u>AND=</u>	数等于比较	○	○	○	○	220
	<u>AND></u>	数大于比较	○	○	○	○	
	<u>AND<</u>	数小于比较	○	○	○	○	
	<u>AND<></u>	数不等于比较	○	○	○	○	
	<u>AND<=</u>	数小于等于比较	○	○	○	○	
	<u>AND>=</u>	数大于等于比较	○	○	○	○	
	<u>OR=</u>	数等于比较	○	○	○	○	221
	<u>OR></u>	数大于比较	○	○	○	○	
	<u>OR<</u>	数小于比较	○	○	○	○	
	<u>OR<></u>	数不等于比较	○	○	○	○	
	<u>OR<=</u>	数小于等于比较	○	○	○	○	
	<u>OR>=</u>	数大于等于比较	○	○	○	○	

4.2 以字母顺序为序

字母	指令	指令功能说明	LX3V1	LX3V2	LX3VP	LX3VE	页码	字母	指令	指令功能说明	LX3V1	LX3V2	LX3VP	LX3VE	页码
A	ABSD	凸轮控制	○	○	○	○	134	E	EI	电子凸轮曲线	-	-	-	○	45
	ADD	BIN 加法	○	○	○	○	72		ENCO	编码	○	○	○	○	59
	ALT	交替输出	○	○	○	○	138	F	FEND	主程序结束	○	○	○	○	44
	AND=	(S1)=(S2)	○	○	○	○	220		FLT	整形转浮点	-	○	○	○	63
	AND>	(S1)>(S2)	○	○	○	○	220		FMOV	多点传送	-	○	○	○	54
	AND<	(S1)<(S2)	○	○	○	○	220		FOR	循环范围开始	○	○	○	○	48
	AND<>	(S1)≠(S2)	○	○	○	○	220		FROM	BFM 读出	○	○	○	○	120
	AND<=	(S1)<=(S2)	○	○	○	○	220	G	G90G01	绝对位置直线插补	-	-	-	○	157
	AND>=	(S1)>=(S2)	○	○	○	○	220		G91G01	相对位置直线插补	-	-	-	○	159
B	ANR	信号报警器复位	-	○	○	○	62		G90G02	绝对位置顺圆插补	-	-	-	-	161
	ANS	信号报警器置位	-	○	○	○	61		G91G02	相对位置顺圆插补	-	-	-	-	163
	ARWS	方向开关	-	○	○	○	116		G90G03	绝对位置逆圆插补	-	-	-	-	165
	ASC	ASCII 码转换	-	○	○	○	118		G91G03	相对位置逆圆插补	-	-	-	-	167
C	ASCI	HEX 转 ASCII	○	○	○	○	187		GBIN	格雷码逆变换	-	○	○	○	122
	BCD	BCD 转换	○	○	○	○	56	H	GRY	格雷码变换	-	○	○	○	122
	BIN	BIN 转换	○	○	○	○	56		HEX	ASCII 转 HEX	○	○	○	○	189
	BMOV	成批传送	○	○	○	○	54		HKY	16 键输入	-	○	○	○	113
D	BON	ON 位判断	-	○	○	○	60		HOURL	计时表	○	○	○	○	71
	CALL	子程序调用	○	○	○	○	44	I	INC	BIN 加 1	○	○	○	○	76
	CCD	校验码	○	○	○	○	191		INCD	凸轮控制	○	○	○	○	135
	CCPID	PID 运算	-	-	-	-	192		INT	浮点数转整型	-	○	○	○	208
	CJ	条件跳转	○	○	○	○	43		IRET	中断返回	○	○	○	○	45
	CML	取反传送	-	○	○	○	53		IST	状态初始化	○	○	○	○	130
	CMP	比较指令	○	○	○	○	49	L	LD=	(S1)=(S2)	○	○	○	○	219
E	CPAVL	串口参数配置	○	○	○	○	181		LD>	(S1)>(S2)	○	○	○	○	219
	DABS	读取 ABS 当前值	○	○	○	○	144		LD<	(S1)<(S2)	○	○	○	○	219
	DACOS	COS 值转弧度值	-	-	○	○	212		LD<>	(S1)≠(S2)	○	○	○	○	219
	DASIN	SIN 值转弧度值	-	-	○	○	211		LD>=	(S1)<=(S2)	○	○	○	○	219
	DATAN	TAN 值转弧度值	-	-	○	○	213		LD<=	(S1)>=(S2)	○	○	○	○	219
	DCOS	浮点数 COS 运算	-	-	○	○	210	M	MEAN	平均值	-	○	○	○	61
	DCOSH	双曲余弦运算	-	-	○	○	214		MOV	传送	○	○	○	○	51
	DDEG	弧度值转角度	-	-	○	○	215		MTR	矩阵输入	○	○	○	○	82
	DEADD	浮点数加法	-	○	○	○	204		MUL	BIN 乘法	○	○	○	○	74
	DEBIN	10 进制转浮点数	-	○	○	○	204	N	NEG	求补码	-	○	○	○	79
	DEBCD	浮点数转 10 进制	-	○	○	○	203		NEXT	循环范围结束	○	○	○	○	48
	DECAM	电子凸轮配置	-	-	-	○	123	O	OR=	(S1)=(S2)	○	○	○	○	221
	DECMF	浮点数比较	-	○	○	○	202		OR>	(S1)>(S2)	○	○	○	○	221
	DECO	译码	○	○	○	○	58		OR<	(S1)<(S2)	○	○	○	○	221
	DEDIV	浮点数除法	-	○	○	○	207		OR<>	(S1)≠(S2)	○	○	○	○	221
	DEGEAR	电子齿/手摇轮	-	-	-	○	126		OR>=	(S1)<=(S2)	○	○	○	○	221
	DEMUL	浮点数乘法	-	○	○	○	206		OR<=	(S1)>=(S2)	○	○	○	○	221
F	DESQR	浮点数开方	-	○	○	○	208	P	PID	PID 运算	○	○	○	○	200
	DESUB	浮点数减法	-	○	○	○	205		PLSR	加减速脉冲输出	○	○	○	○	90
	DEXP	e 的指数运算	-	○	○	○	217		PLSR2	多段速脉冲输出	○	○	○	○	92
	DEZCP	浮点数区间比较	-	○	○	○	203		PLSV	变速脉冲输出	○	○	○	○	147
	DHSCR	比较复位	○	○	○	○	83		PLSY	脉冲输出	○	○	○	○	88
	DHSCS	比较置位	○	○	○	○	84		PRUN	8 进位传送	○	○	○	○	186
	DHSZ	区间比较	-	○	○	○	86		PR	ASCII 码打印输出	-	○	○	○	119
	DI	中断禁止	○	○	○	○	45		PTO	包络脉冲输出	○	○	○	○	100
	DIV	BIN 除法	○	○	○	○	75		PWM	脉宽调制	○	○	○	○	89
	DLOG10	10 的对数运算	-	○	○	○	217	R	RAMP	斜坡信号	○	○	○	○	140
	DLOGE	E 的对数运算	-	○	○	○	218		RCL	带进位循环左移	-	○	○	○	104
	DRAD	浮点数转弧度	-	○	○	○	216		RCR	带进位循环右移	-	○	○	○	105
	DRVA	绝对定位	○	○	○	○	150		REF	输入输出刷新	○	○	○	○	80
	DRVI	相对定位	○	○	○	○	148		REFF	滤波器调整	-	○	○	○	81
	DSIN	浮点数 SIN 运算	-	○	○	○	209		ROL	循环左移	○	○	○	○	102
	DSINH	双曲正弦运算	-	-	○	○	214		ROR	循环右移	○	○	○	○	103
	DSW	数字开关	○	○	○	○	114		ROTC	旋转工作台控制	-	○	○	○	141
	DTAN	浮点数 TAN 运算	-	○	○	○	210		RS	串行数据传送	○	○	○	○	169
	DTANH	双曲正切运算	-	-	○	○	215		RS2	串行数据传送 2	○	○	○	○	172
G	DVIT	中断定位	-	-	○	○	152		RSLIST	表格化通信	-	-	○	○	174
	ECAMTBX	电子凸轮曲线	-	-	-	○	129	S	SCL	定坐标	-	-	-	-	154

字母	指令	指令功能说明	LX3V1	LX3V2	LX3VP	LX3VE	页码
S	SEGD	7 段译码	-	○	○	○	115
	SEGL	7 段码(按时间)	○	○	○	○	116
	SER	16 位数据查找	-	○	○	○	137
	SFRD	移位读出	○	○	○	○	110
	SFTL	位左移	○	○	○	○	106
	SFTR	位右移	○	○	○	○	107
	SFWR	移位写入	○	○	○	○	111
	SMOV	移位传送	-	○	○	○	52
	SORT	数据排列	-	○	○	○	142
	SPD	脉冲密度	○	○	○	○	87
	SQR	BIN 开方	-	○	○	○	62
	STMR	特殊定时器	-	○	○	○	137
	SUB	BIN 减法	○	○	○	○	73
	SUM	ON 位数	-	○	○	○	60
	SWAP	高低字节交换	-	○	○	○	64
T	TADD	时钟数据加法	○	○	○	○	67
	TCMP	时钟数据比较	○	○	○	○	65
	TKY	数字键输入	-	○	○	○	112
	TO	BFM 写入	○	○	○	○	121
	TRD	时钟数据读出	○	○	○	○	69
	TSUB	时钟数据减法	○	○	○	○	68
	TTMR	示教定时器	-	○	○	○	136
	TWR	时钟数据写入	○	○	○	○	70
	TZCP	时钟数据区比较	○	○	○	○	66
W	WAND	逻辑按位与	○	○	○	○	77
	WDT	监控定时器	○	○	○	○	47
	WOR	逻辑按位或	○	○	○	○	77
	WSFL	字左移	-	○	○	○	108
	WSFR	字右移	-	○	○	○	109
	WXOR	逻辑按位异或	○	○	○	○	78
X	XCH	数据交换	-	○	○	○	55
Z	ZCP	区间比较	○	○	○	○	50
	ZRN	原点回归	○	○	○	○	145
	ZRST	批次复位	○	○	○	○	57

5.指令详细说明

5.1 基本顺控指令说明

LD、LDI、LDP、LDF、OUT 指令

指令描述:

LD、LDI 指令所占程序步数为 1，LDP、LDF 所占程序步数为 2。这 4 个指令的操作数都可为 X、Y、S、M、T、C。

OUT 指令的操作数可为 Y、S、T、M、C。当软元件为 Y 和一般 M 的程序步数为 1，为 S 和特殊辅助继电器 M 的程序步数为 2，为定时器 T 的程序步数为 3，为计数器的程序步数为 3-5。

LD、LDI、LDP、LDF 将触点连接到母线上。多个分支用 ANB、ORB 时也使用。

LDP 指令在上升沿（软元件由 OFF 到 ON 变化时）接通一个周期；LDF 指令在下降沿（软元件由 ON 到 OFF 变化时）接通一个周期。

LD、LDI、LDP、LDF 指令的重复次数在 8 次以下。即与后面的 ANB、ORB 指令使用时串并联使用的最多次数为 8 个。

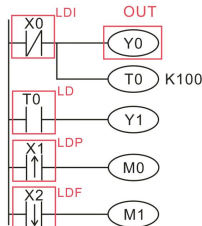
软元件 Y 和一般 M 的程序步为 1，S 和特殊辅助继电器 M 的程序步为 2，定时器 T 的程序步为 3，定时器 C 的程序步为 3-5。

OUT 指令各种软元件的线圈驱动，但对输入继电器不能使用，并列的 OUT 可多次连续使用。

OUT 指令驱动计数器时，当前面的线圈从 ON 变成 OFF，或者从 OFF 变成 ON 时，计数器才加 1。

编程示例:

梯形图模式:



指令表模式:

```

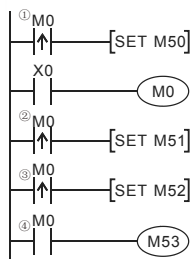
0  LDI  X000
1  OUT  Y000
2  OUT  T0    K100
5  LD   T0
6  OUT  Y001
7  LDP  X001
9  OUT  M0
10 LDF  X002
12 OUT  M1
13 END
    
```

用 LD、LDI、LDP、LDF 指令与母线连接。输出使用 OUT 指令驱动线圈。

使用 OUT 指令驱动定时器的计时线圈或者计数器的计时线圈时，不需设定定时时间和计数值，可以是常数 K，或者由寄存器间接指定数值。

在将辅助继电器（M）指定为 LDP，LDF，ANDP，ANDF，ORP，ORF 指令的软元件时，软元件的编号范围不同会造成下图所示的动作差异。

《M0-M2799》

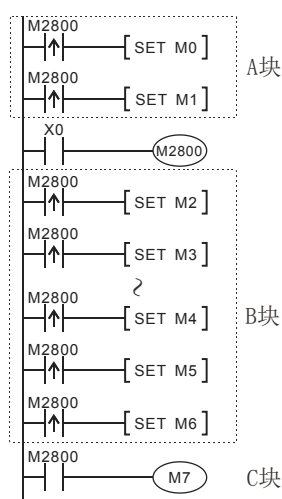


上图由 X0 驱动 M0 后，与 M0 对应的①~④的所有触点都动作。

其中：①~③执行 M0 的上升沿检出。

④为 LD 指令。因此，在 M0 接通过程中导通。

《M2800-M3071》



左图以由 X0 驱动的 M2800 为中心，分为上下 A、B 两个区域。在 A、B 两个区域内的上升沿检出与下降沿检出的触点中只有第一个触点动作。C 区域内的触点为 LD 指令，因而在 M2800 接通过程中导通。

利用这一特性，可有效地对步进梯形图中【利用同一信号进行状态转移】进行高效率的编程。

AND、ANI、ANDP、ANDF 指令

指令描述:

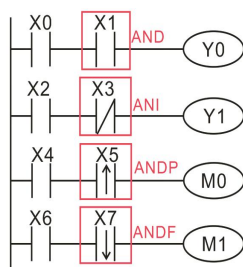
AND、ANI 程序步数为 1；ANDP、ANDF 程序步数为 2。这 4 个指令的操作数都可为 X、Y、S、M、T、C。

AND、ADNI、ANDP、ANDF 指令只能串接一个触点，两个以上的并联回路串联时使用后面的 ANB 指令。串联次数不受限制。

ANDP、ANDF 指令在上升沿（即元件由 ON 到 OFF 变化时）和下降沿（软元件由 OFF 到 ON 变化时）接通一个周期。

编程示例:

梯形图模式:



指令表模式:

```

0 LD X000
1 AND X001
2 OUT Y000
3 LD X002
4 ANI X003
5 OUT Y001
6 LD X004
7 ANDP X005
9 OUT M0
10 LD X006
11 ANDF X007
13 OUT M1
14 END
    
```

例中，X0、X3、Y1 作为串联触点与前面的触点相连。

OR、ORI、ORP、ORF 指令

指令描述:

OR、ORI 所占程序步数为 1，ORP、ORF 所占程序步数为 2；这 4 个指令操作数都可为 X、Y、S、M、T、C。

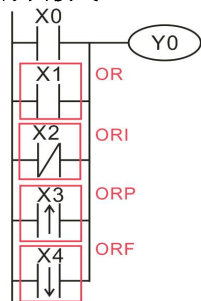
OR、ORI、ORP、ORF 只能并接一个触点，两个以上的串联回路并接时使用后面的 ORB 指令。

ORP、ORF 指令在上升沿（即软元件由 OFF 到 ON 变化时）和下降沿（即软元件由 ON 到 OFF 变化时）接通一个周期。

OR、ORI、ORP、ORF 指令和前面的 LD、LDI、LDP、LDF 指令一起使用，并联次数不受限制。

编程示例:

梯形图模式:



指令表模式:

```

0 LD X000
1 OR X001
2 ORI X002
3 ORP X003
5 ORF X004
7 OUT Y000
8 END
    
```

使用 OR、ORI、ORP、ORF 与前面的 LD、LDI、LDP、LDF 并联连接，在程序中有两个并联回路块的串联，故使用了 ANB 指令，关于 ANB 指令可参照后面章节的说明。

ANB、ORB 指令

指令描述:

ANB 指令无操作数，所占程序步数为 1；ORB 指令操作数可为 X、Y、S、M、T、C，所占程序步数为 1。

多分支回路与前面的回路串联时，使用 ANB 指令。分支以 LD、LDI、LDP、LDF 作为起点，使用 ANB 指令与前面以 LD、LDI、LDP、LDF 指令作为起点的分支串联连接。

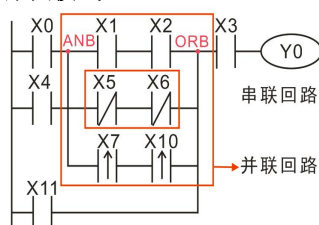
当 2 个以上的触点串接的串联回路块并联连接时，每个分支使用 LD、LDI 指令开始，ORB 指令结束。

ANB、ORB 指令都不是带软元件的指令。

ANB、ORB 使用的并串联回路的个数不受限制，但是当成批使用时，必须考虑 LD、LDI 的使用次数在 8 次以下。

编程示例:

梯形图模式:



指令表模式:

0	LD	X000	9	ANBP	X010
1	OR	X004	11	ORB	
2	LD	X001	12	ANB	
3	AND	X002	13	OR	X011
4	LDI	X005	14	AND	X003
5	ANI	X006	15	OUT	Y000
6	ORB		16	END	
7	LDP	X007			

在每个分支的最后使用 ORB 指令，不要在所有的分支后面使用 ORB 指令，如上图指令表所示。

ORB 和 ANB 指令只是对块的连接，如果不是块就不使用，如图所示：串联回路块和并联回路块的示例。

INV 指令

指令描述:

INV 指令是将 INV 指令之前，LD、LDI、LDP、LDF 指令之后的运算结果取反的指令，没有操作数，所占程序步数为 1。

编程示例:

梯形图模式:



指令表模式:

0	LD	X000
1	INV	
2	OUT	Y000
3	END	

使用 OR、ORI、ORP、ORF 与前面的 LD、LDI、LDP、LDF 并联连接，在程序中有两个并联回路块的串联，故使用了 ANB 指令，关于 ANB 指令可参照后面章节的说明。

指令描述:

MC 指令所占程序步数为 3，操作数可为 Y、M（特殊 M 除外）；MCR 指令所占程序步数为 2，操作数可为 Y、M（特殊 M 除外）。

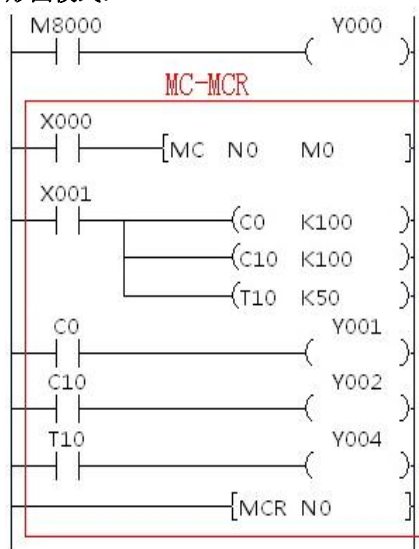
当前面的触点接通时，就执行 MC 到 MCR 的指令。执行 MC 指令时，母线向 MC 触点后移动，执行 MCR 指令返回母线。

使用 MC 指令时，嵌套级 K 的编号按顺序依次增大，也就是说只有 K0，才能嵌套 K1。相反使用 MCR 时，必须从大往小返回母线。最大嵌套级为 8（K7）。

通过不同的软元件 Y、M，可以多次使用 MC 指令，如果使用相同的元件，将同 OUT 指令一样，会出现双线圈输出。

编程示例:

梯形图模式:



指令表模式:

0	LD	M8000	
1	OUT	Y000	
2	LD	X000	
3	MC	N0	M0
6	LD	X001	
7	OUT	C0	K100
10	OUT	C10	K100
13	OUT	T10	K50
16	LD	C0	
17	OUT	Y001	
18	LD	C10	
19	OUT	Y002	
20	LD	T10	
21	OUT	Y004	
22	MCR	N0	
23	END		

该示例只使用了 MC、MCR 指令，嵌套级数也是 1，可以进行 7 级嵌套。

该示例中的 X0 接通时，执行 MC、MCR 之间的指令，当 X0 断开时，成为以下两种形式：

- ①现状保持：累计定时器的值或计数器的值，用 SET/RST 指令驱动软元件。
- ②变为断开元件：非累计定时器的值，用 OUT 指令驱动软元件。

指令描述:

MPS、MRD、MPP 指令都无操作数，这 3 个指令所占程序步数均为 1。

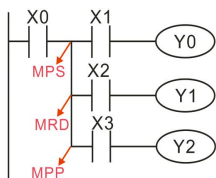
嵌入式 PLC 中有 11 个栈空间，也就是说可以压栈的最大深度为 11 级。每使用一次 MPS 将当前结果压入第一段存储，以前压入的结果依次移入下一段。MPP 指令将第一段读出，并且删除它，同时以下的单元依次向前移。MRD 指令读出第一段，但并不删除它。其他单元保持不变。使用这 3 条指令可以方便多分支编程。

在进行多分支编程时，MPS 保存前面的计算结果，以后的分支可以利用 MRD，MPP 从栈中读出前面的计算结果，再进行后面的计算。最后一个分支必须用 MPP，保证 MPS、MPP 使用的次数相同。注意，使用 MPP 以后，就不能使用 MRD 读出运算结果，也就是 MPP 必须放在最后的分支使用。

MRD 指令可以使用多次，没有限制。MPS 连续使用的最大次数为 11，但是可以多次使用。每个 MPS 指令都有一个 MPP 指令对应，MPP 的个数不能多于 MPS 的个数。

编程示例:

梯形图模式（示例 1）:

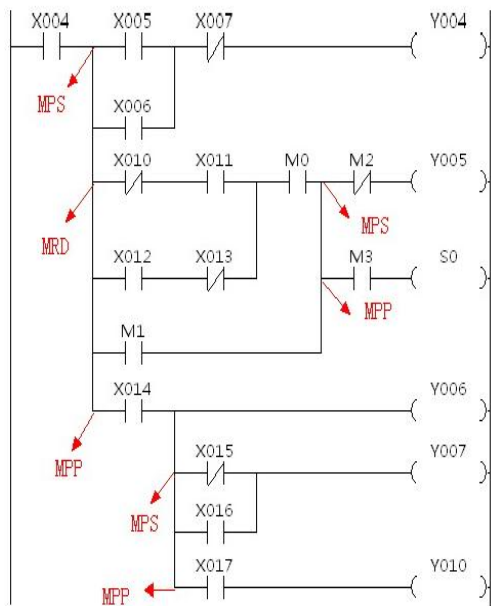


示例 1 中只使用了一级堆栈，使用一个 MPS 指令压栈，一个 MRD 指令读栈，一个 MPP 指令出栈。

指令表模式（示例 1）:

0	LD	X000
1	MPS	
2	AND	X001
3	OUT	Y000
4	MRD	
5	AND	X002
6	OUT	Y001
7	MPP	
8	AND	X003
9	OUT	Y002
10	END	

梯形图模式（示例 2）:



示例 2 中是使用一级两段堆栈，并且 OR、ORB、ANB 指令混合使用。

指令表模式（示例 2）:

0	LD	X004	17	ANI	M2
1	MPS		18	OUT	Y005
2	LD	X005	19	MPP	
3	OR	X006	20	AND	M3
4	ANB		21	OUT	S0
5	ANI	X007	23	MPP	
6	OUT	Y004	24	AND	X014
7	MRD		25	OUT	Y006
8	LDI	X010	26	MPS	
9	AND	X011	27	LDI	X015
10	LD	X012	28	OR	X016
11	ANI	X013	29	ANB	
12	ORB		30	OUT	Y007
13	ANB		31	MPP	
14	AND	M0	32	AND	X017
15	OR	M1	33	OUT	Y010
16	MPS		34	END	

PLS、PLF 指令

指令描述:

PLS、PLF 指令所占程序步数为 1，操作数都可为 Y、M（特殊 M 除外）。

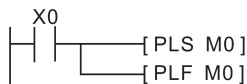
使用 PLS 指令时，只在线圈由 OFF 变成 ON 的一个扫描周期内，驱动软元件。

使用 PLF 指令时，只在线圈由 ON 变成 OFF 的一个扫描周期内，驱动软元件。

在具有停电保持功能的元件，它只在第一次运行时产生脉冲动作。

编程示例:

梯形图模式:



指令表模式:

```
0 LD X000
1 PLS M0
3 PLF M1
5 END
```

SET、RST 指令

指令描述:

SET 指令操作数可为 Y、M、S；RST 指令操作数可为 X、Y、S、M、T、C、D、V、Z。

SET、RST 这 2 个指令的程序步数都由下述规则来定:

软元件为 Y 和一般 M 的程序步为 1，S 和特殊辅助继电器 M、定时器 T、计数器 C 的程序步为 2，数据寄存器 D 以及变址既存取 V 和 Z 的程序步为 3。

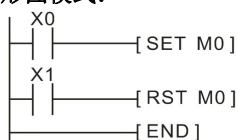
SET 指令在线圈接通的时候就对软元件进行置位，只要置位了，除非用 RST 复位，否则将保持为 1 的状态。同样，对 RST 指令只要对软元件复位，将保持为 0 的状态，除非用 SET 指令置位。

对同一软元件，SET、RST 指令可以多次使用，顺序随意。

RST 指令可以对数据寄存器 D、变址寄存器 V、Z、定时器 T、计数器 C、不论是保持还是非保持的都可以复位置零。

编程示例:

梯形图模式:



指令表模式:

```
0 LD X000
1 SET M0
2 LD X001
3 RST M0
4 END
```

5.2 功能指令说明

5.2.1 程序流指令

CJ 指令

指令描述:

使 CJ、CJP 指令开始到指针（P）为止的顺控程序不执行的指令。可以缩短循环时间（扫描周期）和执行使用双线圈的程序。

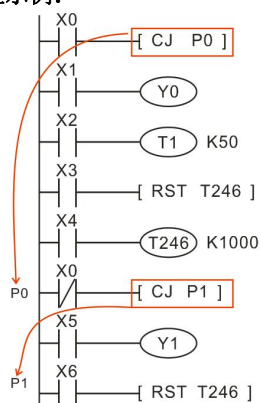
名称	功能	位数(bits)	是否脉冲型	指令格式	步数
CJ	条件跳转	16	否	CJ P* ※1	3
CJP		16	是		3

※1: LX3V(1S): P0-P63, LX3V(2N): P0-P127, LX3VP/LX3VE: P0-P127。

当能流有效时，程序自动从 CJ（或 CJP）指令的地址跳转至由 P*指定的地址后继续执行，中间地址的程序指令被跳过，不予执行；

当能流无效时，程序继续往下执行，此时 CJ（或 CJP）指令不被执行。

编程示例:



在上图示例中，如果 X0 为 ON，进行跳转，跳转中的线圈动作如下：

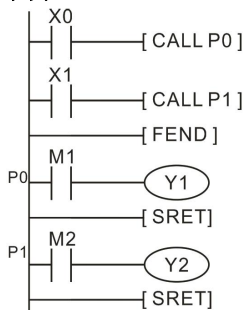
- Y、M、S 保持以前的动作。功能指令跳转后不动作。
- T 在跳转前若没有触发，跳转后即使触发，定时器也不动作。若被触发，始终继续运行，但触点不动作，当 X0 “OFF” 时，触点立即动作。
- C 在跳转前若没有触发，跳转后即使触发计数器不动作。若被触发，计数器中断，当 X0 “OFF” 时继续计数。
- 功能指令跳转后不动作。
- 定时器及计数器的复位指令在跳转外时，计时线圈及跳转的计数线圈位有效。

CALL 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
CALL	子程序调用	16	否	CALL 子程序名	3
CALLP		16	是		3
SERT	子程序返回	无	否	SERT	1
FEND	主程序结束	无	否	FEND	1

编程示例:



由上图的示例程序可知，若 X000 为 ON，则执行 CALL 跳转指令将会到子程序 P0 中去执行，执行完子程序 P0 后，继续回到主程序中执行下个语句；同样，若 X001 为 ON，则会跳转到子程序 P1 中去执行直到程序执行完毕。

在子程序内最多可以允许有 4 层嵌套，也就是说在子程序中可以嵌套调用其他子程序的次数最多可以有 4 次。

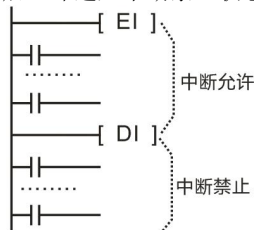
- CALL、CALLP 指令早 FEND 指令后编程，必须要有 SRET 指令。
- 中断用指针也在 FEND 之后编程，必须要有 SRET 指令。
- 在执行 CALL 指令后，IRET 指令执行前，如果执行了 FEND 指令，或者在 FOR 指令执行后，NEXT 指令执行前执行了 FEND 指令，则程序会出错。
- 在使用多个 FEND 指令的情况下，请在最后的 FEND 指令与 END 指令之间编写子程序或中断子程序。

EI、DI 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
EI	允许中断	16	否	无操作数、不需要驱动接点的单独指令	1
DI	禁止中断	16	否		1
IRET	中断返回	16	否		1

PLC 程序开始运行时，默认为中断禁止状态；执行了 EI 语句后，中断功能允许；当中断为允许状态，执行了 DI 语句后，即进入中断禁止状态。在程序中如果没有中断插入禁止的区间时，可以不使用 DI 指令。



中断的种类与设置:

- (1) 外部信号输入中断: 可定义 X0—X5 输入信号的上升沿或下降沿进行中断，对于不需要即时响应的 X 信号，还可以采用脉冲捕捉的功能；
- (2) 定时器中断: 以 1ms—99ms 的固定周期发生的中断；
- (3) 高速计数中断: 与 DHSCS 比较置位指令并用，当高速计数的当前值达到设定值时，产生中断。

外部信号输入中断指针与设置:

输入编号	指针编号		禁止中断
	上升中断	下降中断	
X000	I001	I000	M8050
X001	I101	I100	M8051
X002	I201	I200	M8052
X003	I301	I300	M8053
X004	I401	I400	M8054
X005	I501	I500	M8055

定时中断指针与设置:

输入编号	中断周期 ms	中断禁止指令
I6□□	在指令的□□中输入 1~99 的数， 如 I605 为每 5ms 执行一次定时中断	M8056
I7□□		M8057
I8□□		M8058

高速计数中断指针与设置:

输入编号	中断禁止指令
I010	M8059
I020	
I030	
I040	
I050	
I060	

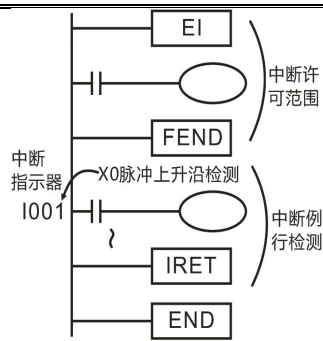
中断子程序选用不同的编号，即选择了不同的端口及中断触发沿；

外部输入中断中对于同一 X 输入，不能同时对上升中断和下降中断编号。对于一个 X 输入端口，只能使用一种触发沿，触发沿通过指针编号来设定；

外部输入中断: 如果对 M8050-M8055 在程序执行过程中"ON"，则禁止了对应 X 端口的中断功能；

定时中断: 如果对 M8056-M8058 在程序执行过程中"ON"，则禁止了对应的定时中断功能；

高速计数中断: 如果对 M8059 在程序执行过程中"ON"，则禁止了所有的高速计数中断功能。



中断的编程规定与执行特性：

在 DI-EI 指令间（中断禁止区间）发生中断，亦能对其记忆并在 EI 指令后执行；

指针编号不能重复使用；

多个中断依次发生时，以先发生的为优先。完全同时发生时，优先级别高的为优先。优先级从高到低分别为高速计数器中断、外部输入中断、时间中断；

在中断例行程序的执行过程中，禁止其它的中断。但若在中断子程序内对 EI、DI 编程，可以接受最多为二重的中断；

在中断处理过程中控制输入继电器时，使用输入刷新指令（REFF），可以通过读取最新的输入状态，实现高速控制；

子程序及中断例行程序内的定时器，请采用例行程序用的定时器 T192-T199；如果采用一般的定时器，除了不能进行计时外，在使用 1 ms 累计定时器时亦需加注意；

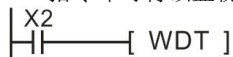
如果指定输入中断指针 I 口 0 口，则输入继电器的输入滤波特性自动关闭。因此，无需采用 REFF 指令及特殊数据寄存器 D8020(输入滤波器调整)。另外，不作为输入中断指针用的输入继电器的输入滤波器能维持 10ms(初始值)。

WDT 指令

指令描述：

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
WDT	监视用定时器的刷新	16	否	无（不需要驱动接点的单独指令）	1
WDTP		16	是		1

PLC 系统内有用于监视用户程序执行一次的时间是否超时的定时器，若超时即会停止用户程序的执行并报警，执行 WDT 指令即可将该监视定时器复位，让监视定时器重新开始计时，避免超时错误。



若用户程序所执行的操作过于复杂（例如过多的循环计算），执行时有可能出现运行超时错误，编程时若必要，可用 WDT 指令（例如在 FOR~NEXT 指令之间中插入该指令）；如果程序扫描时间大于 D8000 的值（默认 200ms），可以在程序间插入 WDT 指令将每段程序分成扫描时间低于 200ms 或者根据需要修改 D8000 的设定值。

编程示例：



此程序扫描时间是 320ms，用 WDT 指令将程序分割为 2 部分，使得每部分程序扫描时间都在 200ms 以下。

FOR、NEXT 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
FOR	循环范围开始	16	否	FOR(S1)	3
NEXT	循环范围结束	-	-	NEXT(无操作数)	1

FOR 指令用于一个循环的起始，同时指明循环执行的次数，必须与 NEXT 指令配套使用。其中：(S1)为循环次数控制变量。

NEXT 指令用于指示循环区域的尾部。由 FOR 指令指定 FOR~NEXT 循环来回执行 N 次后跳出 FOR~NEXT 循环往下继续执行。

在 FOR~NEXT 指令的循环区间，可以嵌入另一个 FOR~NEXT 循环，但规定：从最外层的 FOR~NEXT 计算，最多可内嵌 4 层 FOR~NEXT 循环。运行时 PLC 会以各 FOR~NEXT 层对应解析执行。但需要注意当循环次数过多时，会使 PLC 扫描周期延长，可能造成逾时监视定时器动作而导致错误产生。可在 FOR~NEXT 指令之间使用 WDT 指令来改善。

有下列情况者，都会出错：

NEXT 指令在 FOR 指令之前；

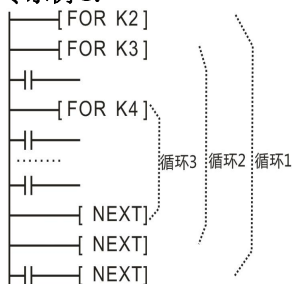
有 FOR 指令而无 NEXT 指令；

FOR 指令与 NEXT 指令个数不一致等

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√	√		√	√	√	√	√	√	√	√	√

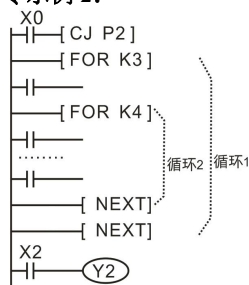
编程示例:

指令示例 1:



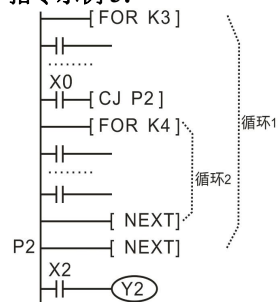
循环 1 执行 2 次后在到 NEXT 指令以后的程序继续执行，而循环 1 每执行一次循环 2 会执行 3 次，而循环 2 每执行一次循环 3 又执行 4 次，所以循环 3 共执行 $2*3*4=24$ 次，循环 2 共执行 $2*3=6$ 次。

指令示例 2:



想跳过 FOR~NEXT 指令时，可用 CJ 跳转指令实现，范例中当 X0 为 OFF 时，执行循环 1 和循环 2，当 X0 为 ON 时，CJ 指令跳转至 P2 处，循环 1 和循环 2 之间的程序不被执行。

指令示例 3:



想跳过循环嵌套的 FOR~NEXT 指令时，也可用 CJ 跳转指令实现，范例中当 X0 为 OFF 时，执行循环 1 内的循环 2，当 X0 为 ON 时，CJ 指令跳转至 P2 处，循环 1 内嵌套的循环 2 中 FOR~NEXT 指令被 CJ 指令跳过不执行。

5.2.2 传送与比较指令

CMP 指令

指令描述:

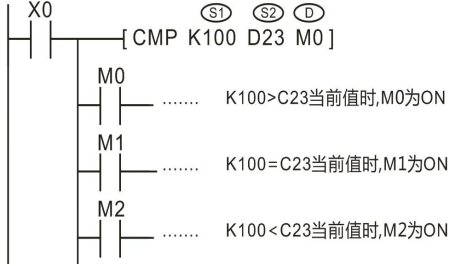
名称	功能	位数(bits)	是否脉冲型	指令格式	步数
CMP	比较指令	16	否	CMP (S1) (S2) (D)	7
CMPP		16	是		7
DCMP		32	否		13
DCMPP		32	是		13

本指令完成对两个操作变量的大小作比较，将比较结果输出给指定的位变量，操作数均按有符号数进行代数比较操作。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√	√		√	√	√	√	√	√	√	√	√
(S2)					√	√		√	√	√	√	√	√	√	√	√
(D)		√	√	√												

其中 (D) 会占用 3 个连续地址的位变量。

编程示例:



当 X0=ON 时，M0~M2 其中之一会 ON。

当 X0=OFF 时，不执行 CMP 指令，M0~M2 仍保持 X0=OFF 之前的状态，若要清除 M0~M2 的比较结果可用 RST 或者 ZRST 对 M0~M2 进行清除。

若需要得到 ≤、≥、≠ 的结果时，可将 M0~M2 串并联即可取得。

ZCP 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
ZCP	区域比较	16	否	ZCP (S1) (S2) (S) (D)	7
ZCPP		16	是		7
DZCP		32	否		13
DZCP		32	是		13

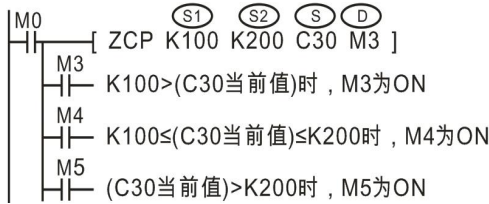
操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√	√		√	√	√	√	√	√	√	√	√
(S2)					√	√		√	√	√	√	√	√	√	√	√
(S)					√	√		√	√	√	√	√	√	√	√	√
(D)		√	√	√												

需要触点驱动，有 4 个操作变量。当控制能流有效时，按有符号数进行代数比较操作，以 (S1) (S2) 为区间，将 (S) 的值位于该区间的位置作为结果，存入 (D) 为起始地址的 3 个连续位变量中。其中：

- (S1) 为比较区间的下限；
- (S2) 为比较区间的上限；
- (S) 为比较变量；
- (D) 为比较结果存储单元，会占用 3 个连续地址的位变量。

注：上限小于下限，按照上限等于下限进行处理。

编程示例:



当 M0=ON 时，M3~M5 其中之一会 ON。

M0 由 ON 变 OFF 时，不执行 ZCP 指令，M3~M5 仍保持 M0=OFF 之前的状态，若要清除 M3~M5 的比较结果，可用 RST 或者 ZRST 对 M3~M5 进行清除。

MOV 指令

指令描述:

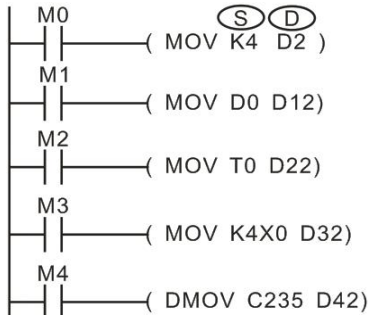
名称	功能	位数(bits)	是否脉冲型	指令格式	步数
MOV	将数据原样传送的指令	16	否	MOV(S)(D)	7
MOVP		16	是		7
DMOV		32	否		13
DMOVP		32	是		13

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)					√	√		√	√	√	√	√	√	√	√	√
(D)									√	√	√	√	√	√	√	√

需要触点驱动，有 2 个操作变量，将(S)的值复制到(D)中。

当为 32bit 指令 (DMOV) 时，(S)和(D)都会使用相邻高地址的变量单元参与运算。例如语句：(DMOV D1 D5) 的操作结果是：D1→D5；D2→D6

编程示例:



当 M0 为 ON 时，将 K4 复制到 D2 中，当 M0 由 ON→OFF 时，D2 保存 K4 的内容不变，除非用户程序再次将 D2 的值修改，或者 PLC 由 STOP→RUN 和 PLC 重新上电，D2 的值才会变为 0，停电保持寄存器上电或者由停止到运行保存原来的值不变。

SMOV 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
SMOV	位移动	16	否	SMOV (S) (m1) (m2) (D) (n)	11
SMOVP		16	是		11

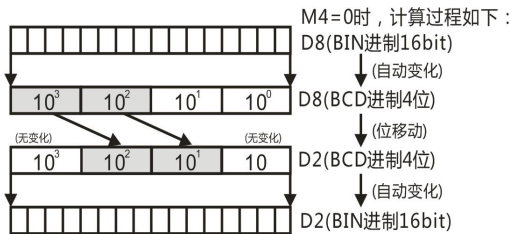
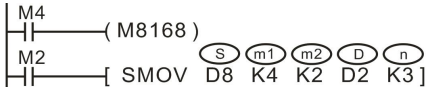
操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)								√	√	√	√	√	√	√	√	√
(m1)					√	√										
(m2)					√	√										
(D)									√	√	√	√	√	√	√	√
(n)					√	√										

需要触点驱动，最多有 5 个操作变量，其中：

- (S)为待复制的数据源变量；
- (m1)为数据源传送的起始位号，（1~4）范围；
- (m2)为数据源传送的位数，（1~m1）范围；
- (D)为数据源传送的目的变量；
- (n)为数据源传送的目的变量的起始位，（m2~4）范围。

数据位的传送过程与特殊标志 M8168 的状态有关，当 M8168 为 OFF 时是 BCD 模式（十进制的位），当 M8168 为 ON 时是 BIN 模式，在 BIN 模式下以 4 个位作为一个单位作传送（十六进制的位）。

编程示例：



假设 D8=K1234，D2=K5678，则当 M8168 为 OFF 时（BCD 模式），将 M2 置 ON，则 D2 的值变为 K5128；



当 M8168 为 ON 时（BIN 模式），此时 D8=H04D2=K1234，D2=H162E=K5678，将 M2 置 ON，D2=H104E=K4174。

CML 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
CML	将数据反向传送的指令	16	否	CML (S) (D)	5
CMLP		16	是		5
DCML		32	否		13
DCMLP		32	是		13

需要触点驱动，有 2 个操作变量，将 (S) 的 BIN 值逐位取反后复制到 (D) 中。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)					√	√		√	√	√	√	√	√	√	√	√
(D)									√	√	√	√	√	√	√	√

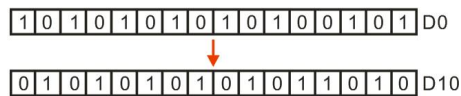
当 (D) 的位数不足 16bit 时，将 (S) 取反后按低位对齐传送到 (D) 变量中。

当为 32bit 指令 (DCML) 时，(S) 和 (D) 都会使用相邻高地址的变量单元参与运算。例如语句：(DCML D1 D5)

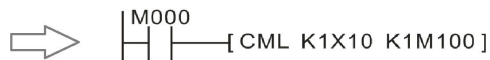
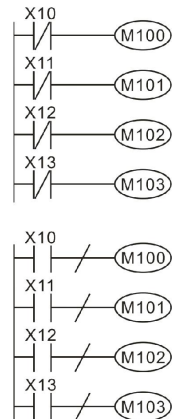
的操作结果是：/D1→D5；/D2→D6

编程示例:

指令示例 1:

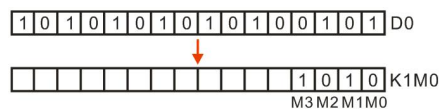


指令示例 2:



左边 2 个程序可以用 CML 指令来实现

指令示例 3:



BMOV 指令

指令描述:

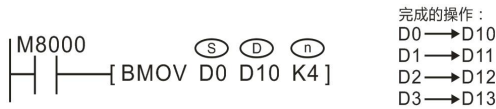
名称	功能	位数 (bits)	是否脉冲型	指令格式	步数
BMOV	将以源指定的软元件为开头的 n 点数据向以目标指定的软元件为开头的 n 点软元件成批传送	16	否	BMOV (S) (D) (n)	7
BMOV		16	是		7

需要触点驱动, 有 3 个操作变量, 将由 (S) 指定起始地址的 (n) 个变量值复制到由 (D) 指定起始地址的 (n) 个单元中。

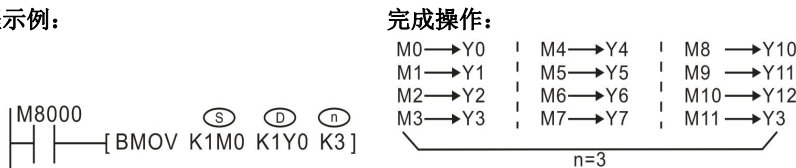
操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)								√	√	√	√	√	√	√		
(D)									√	√	√	√	√	√		
(n)	常数，n=0~512															

其中 (n) 的取值范围是 0~512。

当特殊变量 M8024=1 时, 成批传送的方向相反, 即将由 (D) 指定起始地址的 (n) 个变量值复制到由 (S) 指定起始地址的 (n) 个单元中。



编程示例:



FMOV 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
FMOV	多点传送	16	否	FMOV (S) (D) (n)	7
FMOV		16	是		7
DFMOV		32	否		13
DFMOV		32	是		13

需要触点驱动, 有 3 个操作变量, 将由 (S) 的数据复制到由 (D) 指定起始地址的 (n) 个单元中。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)								√	√	√	√	√	√	√	√	√
(D)									√	√	√	√	√	√		
(n)	常数, n=1~512															

编程示例:



当 M8 置 ON 时完成的操作是:

K100 → D100 | K100 → D102

K100 → D101 | K100 → D103

XCH 指令

指令描述:

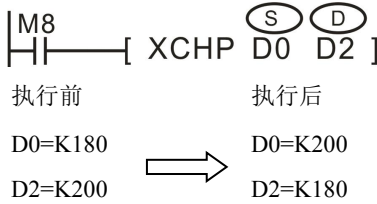
名称	功能	位数(bits)	是否脉冲型	指令格式	步数
XCH	数据交换	16	否	XCH (S) (D)	5
XCHP		16	是		5
DXCH		32	否		9
DXHCP		32	是		9

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)									√	√	√	√	√	√		
(D)									√	√	√	√	√	√		

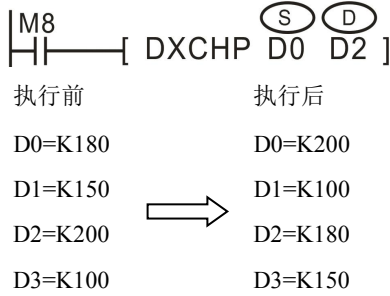
需要触点驱动，有 2 个操作变量，将 (S) 和 (D) 的值彼此交换。

编程示例:

指令示例 1:



指令示例 2:



当特殊变量 M8160=1 时，且 (D) 与 (S) 为同一地址，完成的操作将是高 8 位与低 8 位的交换，32 位的指令也一样，完成的操作将是高 8 位与低 8 位的交换，相当于 SWAP 指令的操作，一般用 SWAP 指令来实现。

BCD 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
BCD	将源 (BIN) 转换为目标 (BCD) 的指令	16	否	BCD (S) (D)	5
BCDP		16	是		5
DBCD		32	否		9
DBCDP		32	是		9

需要触点驱动, 有 2 个操作变量, 将 (S) (BIN) 的值进行 BCD 变换后存入 (D) 中。该指令常用于将数据显示前的数据格式处理。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)									√	√	√	√	√	√	√	√
(D)									√	√	√	√	√	√	√	√

使用 16bit 指令, 当转换结果超过 9999 时会出错; 使用 32bit 指令, 当转换结果超过 99999999 时会出错。M8067 会置 ON, D8067 记录错误代码。

编程示例:



将 D200 的 BIN 值转换成 BCD 值后, 将结果的个位数存于 K1Y0 中 (Y0~Y3 四个 bit 元件中)。

若 D200=H000E (十六进制) =K14 (十进制), 则变换后 Y0~Y3=0100 (BIN);

若 D200=H0028 (十六进制) =K40 (十进制), 则变换后 Y0~Y3=0000 (BIN)。

BIN 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
BIN	将源 (BCD) 转换为目标 (BIN) 的指令	16	否	BIN (S) (D)	5
BINP		16	是		5
DBIN		32	否		9
DBCDP		32	是		9

需要触点驱动, 有 2 个操作变量, 将 (S) (BCD) 的值进行 BIN 变换后存入 (D) 中。该指令常用于将外部端口读入数据 (如编码盘设置) 处理成能直接用于运算的 BIN 格式。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)									√	√	√	√	√	√	√	√
(D)									√	√	√	√	√	√	√	√

(S) (BCD) 的有效范围, 16bit: 0~9999; 32bit: 0~99, 999, 999。

(D) 的数据内容不是 BCD 值 (以 Hex 表示有任一位数不在 0~9 的范围内) 时将会产生运算错误, M8067 会置位。

编程示例:



当 M8 置位时, 将 K1Y0 的 BCD 值作 BIN 转换后存入 D200 中。

5.2.3 数据处理指令

ZRST 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
ZRST	全部复位	16	否	ZRST (D1) (D2)	5
ZRSTP		16	是		5

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(D1)		√	√	√								√	√	√		
(D2)		√	√	√								√	√	√		

将(D1)至(D2)区间的变量全部清 0。(D1)和(D2)可指定字变量，也可为 Y、M、S 位变量。

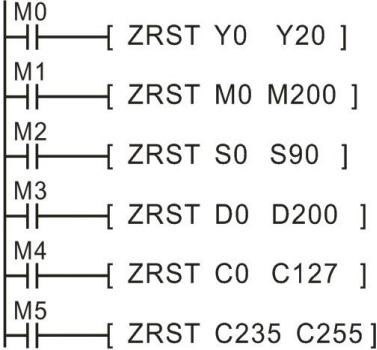
其中要求:

(D1)和(D2)必须为同一类型的软元件;

编号(D1)应不大于(D2)，若两者相同时，仅复位指定的软元件;

本指令为 16bit，但(D1)和(D2)可指定 32bit 的计数器，此时应同为 32bit 型或同为 16bit 型;

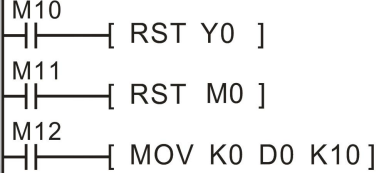
编程示例:



补充说明:

位装置 Y、M、S 和字装置 T、C、D 也可使用 RST 指令来单独复位; 字装置 T、C、D 和位寄存器 KnY、KnM、

KnS 也可以用 FMOV 来多点清除。例如:



DECO 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DECO	译码(将数据中的任意一个转换成 1 点的 ON 位)	16	否	DECO (S) (D) (n)	7
DECOP		16	是		7

计算 (S) 的最后 (2^n) 位的值, 作为 bit 位指针, 将 (D) 的对应位置 1, 其他位清 0。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)	✓	✓	✓	✓	✓							✓	✓	✓	✓	✓
(D)		✓	✓	✓								✓	✓	✓		
(n)	常数, $n=1\sim 8$ 。若 $n=0$, 指令不执行; 其他值则执行出错。															

源地址的低 n 位 ($n \leq 4$) 被解码至目标地址。 $n \leq 3$ 时, 目标的高位都转为 0;

$n=0$ 时命令不执行, $n=0\sim 8$ 以外时为运算错误;

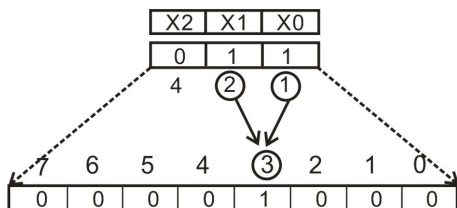
$n=8$ 时, 如果译码命令为位软元件时, 其点数是 256 点。

驱动输入为 OFF 时, 指令不执行, 正在动作的译码输出保持动作。

本指令一般使用脉冲执行型指令。

注: 当 D 参数为字软元件时 n 的范围为 1~4。

编程示例:



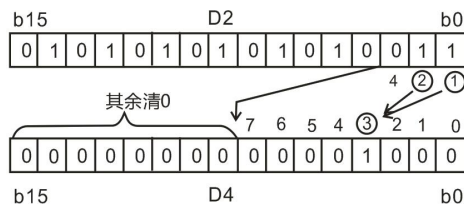
M17 M16 M15 M14 M13 M12 M11 M10

(1) 源地址是 $1+2=3$, 因此从 M0 起第 3 位的 M13 变为 1。源全部为 0 时, M10 为 1。

(2) (2) $n=0$ 时不执行, $n=0\sim 8$ 以外时会运算出错。

(3) (3) $n=8$ 时, 如果译码命令 (D) 为位元件时, 其点数是 $2^8=256$ 点。

(4) (4) 驱动输入为 OFF 时, 指令不执行, 正在动作的译码输出保持动作。



(1) 源地址的低 n 位 ($n \leq 4$) 被解码至目标地址。 $n \leq 3$ 时, 目标的高位都转为 0。

(2) $n=0$ 时不执行, $n=0\sim 4$ 以外时会运算出错。

(3) $n=8$ 时, 如果译码命令 (D) 为位元件时, 其点数是 $2^8=256$ 点。

(4) 驱动输入为 OFF 时, 指令不执行, 正在动作的译码输出保持动作。

ENCO 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
ENCO	编码(求出数据中 ON 位的位置, 并将其转换成 BIN 数据)	16	否	ENCO (S) (D) (n)	7
ENCOP		16	是		7

计算(S)的最后(n)位的值, 作为 bit 位指针, 将(D)的对应位置 1, 其他位清 0。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)	✓	✓	✓	✓	✓	✓						✓	✓	✓	✓	✓
(D)			✓	✓	✓							✓	✓	✓		
(n)	常数, n=1~8。若 n=0, 指令不执行; 其他值则执行出错。															

源地址内有多位是 1 时, 只计算高位侧的第一个为 1 的位; (S)的所有位都为 0 时会出现运算错误; 驱动输入为 OFF 时, 指令不被执行, 编码输出不变化。

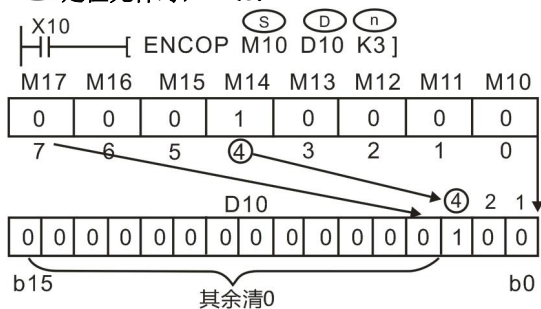
n=8 时, 编码指令(S)如果是位元件, 其点数是 $2^8=256$ 点;

本指令一般使用脉冲执行型指令。

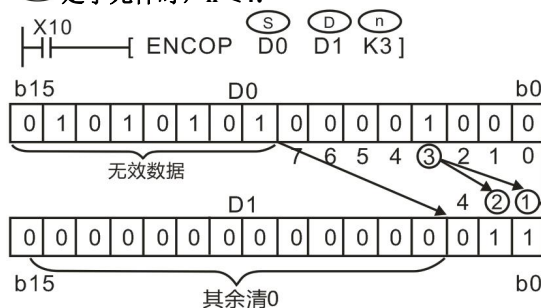
注: 当 D 参数为字软元件时 n 的范围为 1~4。

编程示例:

1、(S)是位元件时, $n \leq 8$:



2、(S)是字元件时, $n \leq 4$:



SUM 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
SUM	计算在软元件中有多少个“1”（ON）的指令	16	否	SUM (S) (D)	5
SUMP		16	是		5
DSUM		32	否		9
DSUMP		32	是		9

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)								√	√	√	√	√	√	√	√	√
(D)									√	√	√	√	√	√	√	√

计算(S)的BIN进制值中为1的位个数，存入(D)。

使用DSUM和DSUMP指令的情况下，((S)+1, (S))的32位中的1的个数写入(D)，同时(D)+1全部为0。

若(S)中的位全部为零，则零标志位M8020会置ON。

编程示例:



BON 指令

指令描述:

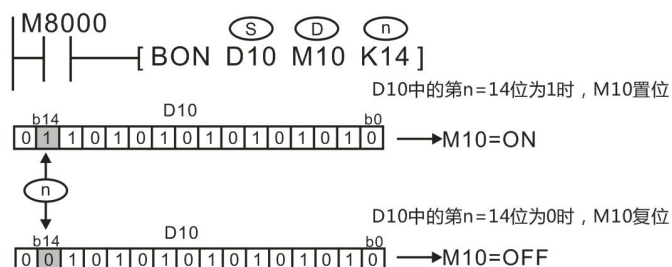
名称	功能	位数(bits)	是否脉冲型	指令格式	步数
BON	检查软元件中指定位置是ON还是OFF的指令	16	否	BON (S) (D) (n)	7
BONP		16	是		7
DBON		32	否		13
DBONP		32	是		13

判断(S)的第(n)位的状态，结果存入(D)。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)					√	√		√	√	√	√	√	√	√	√	√
(D)		√	√	√												
(n)	n=0~15 (16bit) ; n=0~31 (32bit)															

编程示例:

M8000由ON变成OFF时，M10仍保持保持之前的状态。



MEAN 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
MEAN	求数据的平均值的指令	16	否	MEAN (S) (D) (n)	7
MEANP		16	是		7
DMEAN		32	否		13
DMEANP		32	是		13

求取由(S)开始的(n)个变量的平均值（先求和，再除以n），存入(D)。

- 若计算中有余数，余数将被舍弃；
- 当n的值不在1~64的范围时，会计算出错。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)								√	√	√	√	√	√	√		
(D)									√	√	√	√	√	√	√	√
(n)	常数，n=1~64，其他值时，计算会出错。															

编程示例:

X0

(S)

(D)

(n)

[MEAN D10 D20 K4]

(D10+D11+D12+D13) /4=D20

假如 D10=K5，D11=K5，D12=K15，D13=D52，则 D20=K19，余数1被舍去。

ANS 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
ANS	信号报警器置位	16	否	ANS (S) (m) (D)	7

驱动信号报警器的方便指令。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)												√				
(D)				√												
(m)	常数，m=1~32767，（单位为100ms）。															

其中，(S)的范围为T0~T199，(D)的范围为S900~S999。

编程示例:

X1 X2

(S)

(m)

(D)

[ANS T0 K10 S900]

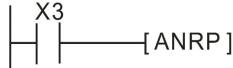
- 如果 X1 和 X2 同时接通1秒以上，则 S900 被置位，以后即使 X1 或 X2 为 OFF 状态，S900 仍保持动作状态（但是 T0 会复位，值变成0）。若不满1秒，X1 或 X2 变为 OFF 时，定时器复位。
- 如果预先将 M8049（信号报警器有效）置 ON，则信号报警器 S900~S999 中最小 ON 状态编号被存入 D8049（ON 状态最小编号）且当 S900~S999 中任意一个为 ON 时，M8048（报警器动作置 ON）。

ANR 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
ANR	信号报警器复位	16	否	ANR(无操作数)	1
ANRP		16	是		1

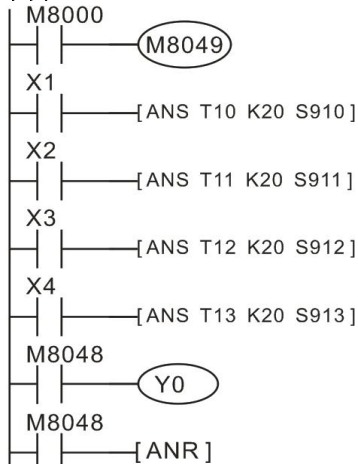
清除报警器信号的方便指令。例如:



如果 X3 接通, 则信号报警器 S900~S999 中正在动作的报警点被复位。如果同时有多个报警点动作时, 则复位最小编号为 ON 的报警点。

若将 X3 再次接通, 则下一编号的状态被复位。实际使用中多用 ANRP 指令。

编程示例:



M8049 为 ON, 当程序中 S900~S999 任意一个为 ON 时, M8048 置 ON, 且 Y0 有报警输出。

假如程序中 S910、S911、S912、S913 都为 ON, 则当 X5 第一次由 OFF 置 ON 时, S910 被复位, 当 X5 第二次由 OFF 置 ON 时, S911 被复位, 以此类推。

SQR 指令

指令描述:

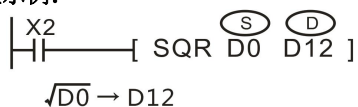
名称	功能	位数(bits)	是否脉冲型	指令格式	步数
SQR	BIN 开方运算	16	否	SQR (S) (D)	5
SQRP		16	是		5
DSQR		32	否		9
DSQRP		32	是		9

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)					√	√								√		
(D)														√		

将 (S) 按 BIN 值开平方运算, 结果存入 (D)。

只能指定 (S) 为正数, 如 (S) 为负数则运算错误标志 M8067 会置 ON, 指令不被执行; 运算结果 (D) 只取整数。舍去小数点, 有小数点被舍去时借位标志 M8021 置 ON; 运算结果是 0 时, 零位标志 M8020 置 ON。

编程示例:



假如 D0=K100, 则 X2 置 ON 的时候, D12=K10;

假如 D0=K110, 则 X2 置 ON 的时候, D12=K10, 小数被舍去。

FLT 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
FLT	BIN 整数转二进制浮点数	16	否	FLT (S) (D)	5
FLTP		16	是		5
DFLT		32	否		9
DFLTP		32	是		9

将整数(S)转换为浮点数, 结果存入(D)和(D)+1 单元。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)														√		
(D)														√		

常数 K、H 在各浮点运算指令中自动转换, 因此在本 FLT 指令中不能使用。

这个指令的逆变换指令是 INT (将 2 进浮点编程示例数值变换成 BIN 整数)。

指令示例 1:



当 M8=ON 时, 将 16bit 数 D10 中 (16 位 BIN 整数) 转换为二进制浮点数后, 存放到 (D121, D120)。

当 M10=ON 时, 将 32bit 数 (D21, D20) 中 (32 位 BIN 整数) 转换为二进制浮点数后, 存放到 (D131, D130)。

指令示例 2:

使用指令来完成下列的浮点运算: $D100 \div K125.5 \times (X17 \sim X0) = D200$



SWAP 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
SWAP	高低字节互换	16	否	SWAP (S)	3
SWAPP		16	是		3
DSWAP		32	否		5
DSWAPP		32	是		5

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)								√	√	√	√	√	√	√	√	√

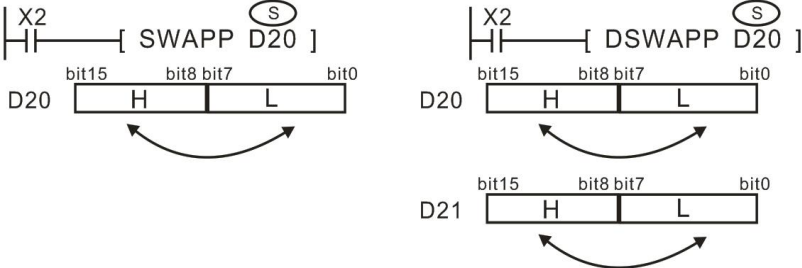
该指令是将指定变量(S)的高低字节的值进行互相交换。

16 位指令时，高 8 位与低 8 位的值进行互相交换。

32 位指令时，两个寄存器的高 8 位与低 8 位的值各自进行互相交换，

注意：此指令一般使用脉冲执行型指令，若采用连续执行指令，则程序每扫描一次，就会进行一次交换。

编程示例:



左图中将 D20 的高 8 位与低 8 位的值进行互相交换，

右图中将 D20 的高 8 位与低 8 位的值进行互相交换，D21 的高 8 位与低 8 位的值进行互相交换。

5.2.4 时钟运算

TCMP 指令

指令描述:

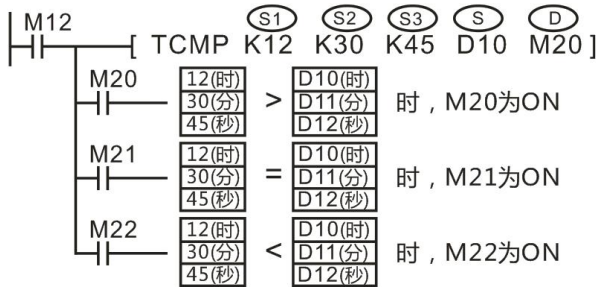
名称	功能	位数(bits)	是否脉冲型	指令格式	步数
TCMP	时钟比较	16	否	TCMP (S1) (S2) (S3) (S) (D)	9
TCMPP		16	是		9

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√	√		√	√	√	√	√	√	√	√	√
(S2)					√	√		√	√	√	√	√	√	√	√	√
(S3)					√	√		√	√	√	√	√	√	√	√	√
(S)												√	√	√		
(D)		√	√	√												

该指令是将指定的时、分、秒数值，与内部实时时钟进行比较，输出比较结果。其中：

- (S1)为指定比较用时间的“时”，范围是 0~23；
- (S2)为指定比较用时间的“分”，范围是 0~59；
- (S3)为指定比较用时间的“秒”，范围是 0~59；
- (S)为实时时钟的时间寄存器的启始地址，通常为时钟读取 TRD 或 MOV 指令读取后的存放单元；
- (D)为比较结果的存放变量启始地址，占用后续共 3 个变量单元。

编程示例:



当 M12 为 ON 时，M20~M22 其中之一会置 ON。

当 M12 由 ON 变 OFF 时，不执行 TCMP 指令，M20/M21/M22 保持 M12=OFF 以前的状态不变。要清除 M20~M22 的比较结果可用 RST 或者 ZRST 对 M20~M22 进行清除。

若需要得到 ≥、≤、≠ 的结果时，可将 M20~M22 串并联即可取得。

TZCP 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
TCZP	时钟数据区间比较	16	否	TZCP (S1) (S2) (S) (D)	9
TZCPP		16	是		9

[illegible]

该指令是将内置实时时钟数据与指定的两组时/分/秒预设值进行区间比较，输出比较结果。其中：

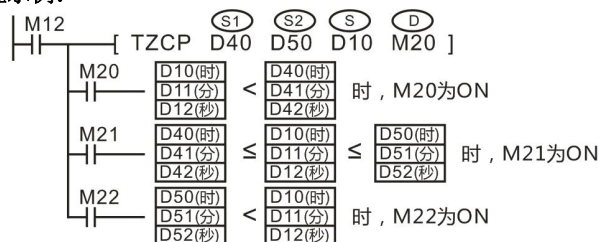
⑤S1为设定的时间比较下限，占用3个连续的变量单元，依次存储时、分、秒数据；

⑤S2为设定的时间比较上限，占用3个连续的变量单元，依次存储时、分、秒数据；

⑤为实时时钟的时间寄存器的起始地址，通常为时钟读取 TRD 或 MOV 指令读取后的存放单元；

④为比较结果的存放变量起始地址，占用后续共 3 个变量单元。

编程示例:



当 M12=ON 时, M20~M22 其中之一会置 ON。

当 M12 由 ON 变 OFF 时, 不执行 TZCP 指令。M20/M21/M22 保持 M12=OFF 以前的状态不变。要清除 M20~M22 的比较结果可用 RST 或者 ZRST 对 M20~M22 进行清除。

TADD 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
TADD	时钟数据加法运算	16	否	TADD (S1) (S2) (D)	7
TADDP		16	是		7

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)												√	√	√		
(S2)												√	√	√		
(D)												√	√	√		

该指令是将 2 组时钟数据的时/分/秒对应相加，结果保存于指定的变量中。其中：

- (S1)为时间被加数，占用 3 个连续的变量单元，依次存储时、分、秒数据；
- (S2)为时间加数，占用 3 个连续的变量单元，依次存储时、分、秒数据；
- (D)为时间相加和，存储单元，占用 3 个连续的变量单元，依次存储时、分、秒数据；

若计算结果超过 24 小时，进位标志 M8022 置 1，实际显示的时间会减去 24: 00: 00 的数值；若计算结果为 00: 00: 00，零标志 M8020 置 1。

编程示例:



完成的操作如下：

(S1)		(S2)		(D)
D10(时) 09	+	D10(时) 08	=	D10(时) 18
D11(分) 50		D11(分) 56		D11(分) 46
D11(秒) 16		D11(秒) 09		D11(秒) 25
9 时 50 分 16 秒		8 时 56 分 9 秒		18 时 46 分 25 秒

如果加法运算结果超过 24 小时，则进位标志 M8022 置 ON。

(S1)		(S2)		(D)
D10(时) 15	+	D10(时) 12	=	D10(时) 4
D11(分) 50		D11(分) 56		D11(分) 46
D11(秒) 16		D11(秒) 09		D11(秒) 25
15 时 50 分 16 秒		12 时 56 分 9 秒		4 时 46 分 25 秒

TSUB 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
TSUB	时钟数据减法运算	16	否	TSUB (S1) (S2) (D)	7
TSUBP		16	是		7

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)												√	√	√		
(S2)												√	√	√		
(D)												√	√	√		

(S1) 该指令是将 2 组时钟数据的时/分/秒对应相减, 结果保存于指定的变量中。其中:

(S2) 为时间被减数, 占用 3 个连续的变量单元, 依次存储时、分、秒数据;

(D) 为时间相减差, 存储单元, 占用 3 个连续的变量单元, 依次存储时、分、秒数据;

若计算结果为负, 借位标志 M8021 置 1, 实际显示的时间会加上 24: 00: 00 的数值; 若计算结果 00: 00: 00, 零标志 M8020 置 1。

编程示例:

M11 [TSUB (S1) (S2) (D)]
D10 D20 D40]

完成的操作如下:

(S1)		(S2)		(D)
D10(时) 09		D10(时) 08		D10(时) 00
D11(分) 50	—	D11(分) 56	=	D11(分) 54
D11(秒) 16		D11(秒) 09		D11(秒) 07
9 时 50 分 16 秒		8 时 56 分 9 秒		00 时 54 分 07 秒

如果减法运算结果为负数时, 则借位标志 M8021 置 ON。

(S1)		(S2)		(D)
D10(时) 09		D10(时) 12		D10(时) 20
D11(分) 50	—	D11(分) 56	=	D11(分) 54
D11(秒) 16		D11(秒) 09		D11(秒) 07
9 时 50 分 16 秒		12 时 56 分 9 秒		20 时 54 分 07 秒

TRD 指令

指令描述:

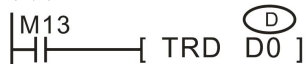
名称	功能	位数(bits)	是否脉冲型	指令格式	步数
TRD	读出时钟数据	16	否	TRD (D)	3
TRDP		16	是		3

操作数	位元件							字元件								
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(D)												√	√	√		

该指令是读取 PLC 内置的实时时钟的年/月/日/时/分/秒/星期，将该 7 个数据保存于指定的寄存器中。 其中：

(D) 为保存读取时间的起始存储单元，占用共 7 个连续的变量单元，地址由小到大依次存储：年、月、日、时、分、秒、星期等数据。

编程示例:



操作如下:

项目	系统变量		操作元件
年（公历：0~99）	D8018	→	D0
月（1~12）	D8017	→	D1
日（1~31）	D8016	→	D2
时（0~23）	D8015	→	D3
分（0~59）	D8014	→	D4
秒（0~59）	D8013	→	D5
星期：0(日)~6(六)	D8019	→	D6

注：一般情况下要使用可编程控制器的时钟，先用 TDR 指令将时钟读出来放到 (D) 寄存器里再使用，不要直接使用 D8012~D8018 的值。

TWR 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
TWR	写入时钟数据	16	否	TWR (S)	3
TWRP		16	是		3

操作数	位元件							字元件								
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)												√	√	√		

该指令是将指定时钟数据(S) (含年/月/日/时/分/秒/星期) 的 7 个数据写入 PLC 内置的实时时钟数据里。

其中: (S) 为保存读取时间的起始存储单元, 占用共 7 个连续的变量单元, 地址由小到大依次存储: 年、月、日、时、分、秒、星期等数据。

编程示例:

指令示例 1:



操作如下:

项目	系统变量		操作元件
年 (公历: 0~99)	D8018	→	D10
月 (1~12)	D8017	→	D11
日 (1~31)	D8016	→	D12
时 (0~23)	D8015	→	D13
分 (0~59)	D8014	→	D14
秒 (0~59)	D8013	→	D15
星期: 0(日)~6(六)	D8019	→	D16

注意: 在写入时钟的时候是 7 个数据全写入的, 在预先设值的时候不能缺少某个变量, 比如星期不写, 则默认是 0, 为星期天; 如果月不预先赋值, 则此时的月变量是 0, 则 PLC 认为你提供的月是错误的, 此次时钟的修改无效。

M8017 每 On 一次, PLC 内部时钟作 ±30 秒校正动作, 这里的校正是指当 PLC 的内部时钟的秒针于 1~29 时, 会被自动归为” 0” 秒而分针不变、30~59 时, 也会被自动归为” 0” 秒, 分针加 1 分钟。

通常的情况下年只显示 2 位数 (例: 2009 年只显示 09), 若需要将” 年” 采用显示 4 位的格式, 仅需在一个扫描周期执行如下语句, 就可以切换成 4 位显示格式:



若原来 D8018=09, 切换后 D8018=2009。 PLC 内部时钟校时方法如下。

指令示例 2:

将 PLC 现在的时间调整为 2017 年 6 月 28 日 10 时 30 分 20 秒, 星期三。



提前一段时间时间将时间写到 D0~D6 当中, 当这个实际时间到来时将 X7 接通, 就将正确时间写到 PLC 里面了。

M8017 在 ON 的一瞬间, 可进行正负 30 秒的调整。

注: 一般情况下要修改可编程控制器的时钟, 用 TWR 指令将时钟写入到 D8012~D8018, 不要用 MOV 指令直接对 D8012~D8018 进行赋值。

HOUR 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
HOUR	计时表	16	否	HOUR (S) (D1) (D2)	7
DHOUR		32	否		13

该指令是记录驱动条件满足的累加时间，当达到设定的时间后，令指定输出有效。其中：

- (S)：满足条件的设定的时间；
- (D1)：累计时间起始单元；
- (D2)：计时到达告警输出变量单元，当计时到达设定值后，该指定单元状态有效。

操作数	位元件				字元件												
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z	
(S)					√	√		√	√	√	√	√	√	√	√	√	
(D1)														√			
(D2)		√	√	√													

16bit 时：(D1)设置范围 K0~K32, 767, 单位：小时。(D1)+1 为未满 1 个小时的现在时间值，设置范围 K0~K3, 599, 单位：秒。此时的(D1)共占用 2 个单元；

32bit 时：(D1)+1、(D1)设置范围 K0~K2, 147, 483, 647, 单位：小时。(D1)+2 为未满 1 个小时的现在时间值，设置范围 K0~K3, 599, 单位：秒。此时的(D1)共占用 3 个单元。

指令(D1)计时没有负数，若(D1)制定为非停电保持的寄存器区域，则在 PLC 由 STOP 到 RUN 或者在掉电的时候会将(D1)的值清零。若需要在 PLC 掉电的情况下仍能保持当前值数据，请将(D1)指定为停电保持区域的寄存器。

编程示例:



当 M200=ON 时，累计该状态的持续时间，将时记录在 D300 中，将不满 1 小时的秒记录在 D301 中，当 D300 累计时间达到 2000 小时后，Y10 输出状态为 ON。计时条件满足时，到达(S)指定数值后，累计计时仍继续进行，读数会继续增大；现在时间值 D300 到达最大数值 32, 767 小时、D301 达到 3, 599 秒时会停止计时测量，要重新计时须将现在时间值 D300、301 清除为 0。

5.2.5 四则运算

ADD 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
ADD	BIN 加法运算	16	否	ADD (S1) (S2) (D)	7
ADDP		16	是		7
DADD		32	否		13
DADDP		32	是		13

需要触点驱动，有 3 个操作变量，将 (S1) 和 (S2) 的值进行 BIN 代数相加后存入 (D) 中，参与运算的变量都按有符号数处理，最高位为符号位，0 为正数，1 为负数。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√	√		√	√	√	√	√	√	√	√	√
(S2)					√	√		√	√	√	√	√	√	√	√	√
(D)									√	√	√	√	√	√	√	√

编程示例:

指令示例 1:



当 M1 置位时，将被加数 D100 的数值上加数 D110 的数值后，存放在 D120 中。假如 D100=K8，D110=K-12，则 D120=8+(-12)=K-4

指令示例 2:



当 M1 置位时，将被加数 D100 的数值上加数 D110 的数值再存放回被加数 D100 中，即 D100 进行累加。假如 D100=K8，D110=K2，则 D100 每次累加 2。

指令使用说明:

若计算结果为 0，则 0 标志(M8020)会置位。

若计算结果超过 32,767 (16bit 运算) 或 2,147,483,647 (32bit 运算) 时，进位标志(M8021)会置位。

若计算结果不满-32,768 (16bit 运算) 或-2,147,483,648 (32bit 运算) 时，借位标志(M8022)会置位。

进行 32bit 运算时，指令中变量地址为为低 16bit 地址，相邻高编号地址单元为高 16bit，编程时防重复或误覆盖。

SUB 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
SUB	BIN 减法运算	16	否	SUB (S1) (S2) (D)	7
SUBP		16	是		7
DSUB		32	否		13
DSUBP		32	是		13

操作数	位元件				字元件												
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z	
(S1)					√	√		√	√	√	√	√	√	√	√	√	
(S2)					√	√		√	√	√	√	√	√	√	√	√	
(D)									√	√	√	√	√	√	√	√	

需要触点驱动，有 3 个操作变量，将(S1)和(S2)的值进行 BIN 代数相减后存入(D)中，参与运算的变量都按有符号数处理，最高位为符号位，0 为正数，1 为负数。

若计算结果为 0，则 0 标志(M8020)会置位；

若计算结果超过 32, 767 (16bit 运算) 或-2, 147, 483, 647 (32bit 运算) 时，进位标志(M8022)会置位；

若计算结果不满-32, 768 (16bit 运算) 或-2, 147, 483, 648 (32bit 运算) 时，借位标志(M8021)会置位；

进行 32bit 运算时，指令中变量地址为为低 16bit 地址，相邻高编号地址单元为高 16bit，编程时防止重复或误覆盖。

编程示例:



当 M8 置位时将被减数 D100 的内容减去减数 D110 的内容后存放到 D120 中，假如 D100=K10; D110=K8，则 D120=10-8=K2。

MUL 指令

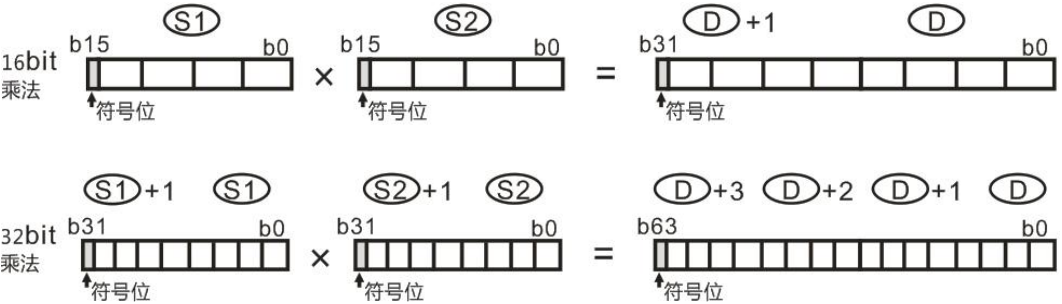
指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
MUL	BIN 乘法运算	16	否	MUL (S1) (S2) (D)	7
MULP		16	是		7
DMUL		32	否		13
DMULP		32	是		13

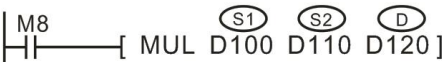
需要触点驱动，有 3 个操作变量，将(S1)和(S2)的值进行 BIN 代数相乘后存入(D)中，参与运算的变量都按有符号数处理，最高位为符号位，0 为正数，1 为负数。

当(D)是位元件时，可进行 K1~K8 的位指定。指定为 K4 时，只能求得乘积运算的低 16 位。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√	√		√	√	√	√	√	√	√	√	√
(S2)					√	√		√	√	√	√	√	√	√	√	√
(D)									√	√	√	√	√	√		



编程示例:



注意:

V、Z 元件仅在 16bit 运算时可用。

进行 32bit 运算时，指令中变量地址为低 16bit 地址，相邻高编号地址单元为高 16bit，编程时防重复或误覆盖。即使使用字元件时，也不能一下子监视 64 位数据的运算结果。

计算的结果只能为 32bit，对于超出 32bit 范围的计算，最好采用浮点运算指令 EMUL 进行计算。

DIV 指令

指令描述:

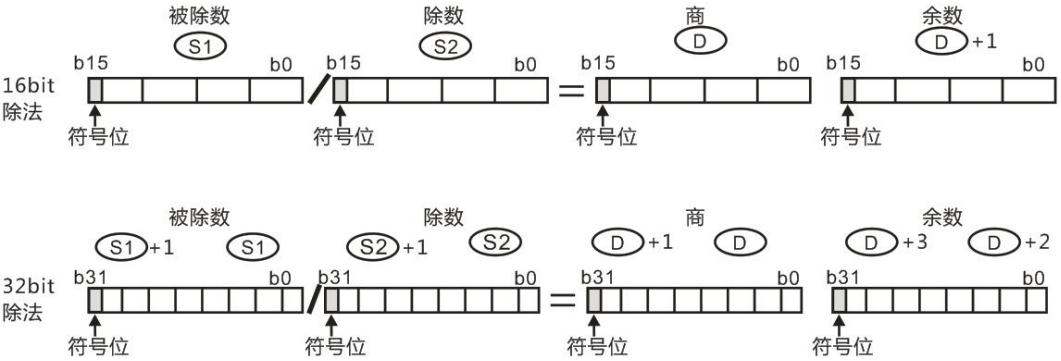
名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DIV	BIN 除法运算	16	否	DIV (S1) (S2) (D)	7
DIVP		16	是		7
DDIV		32	否		13
DDIVP		32	是		13

需要触点驱动，有 3 个操作变量，将被除数(S1)和除数(S2)的值进行 BIN 代数相除后存入(D)中，参与运算的变量都按有符号数处理，最高位为符号位，0 为正数，1 为负数。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√	√		√	√	√	√	√	√	√	√	√
(S2)					√	√		√	√	√	√	√	√	√	√	√
(D)									√	√	√	√	√	√		

表中 V、Z 元件仅在 16bit 运算时可用。进行 32bit 运算时，指令中(S1)和(S2)变量地址为低 16bit 地址，相邻高编号地址单元为高 16bit，编程时防止重复或误覆盖；计算所得的商存入(D)、(D)+1 所指单元，余数存入(D)+2、(D)+3 地址单元中。

若除数(S2)为 0，会发生计算错误；若将位元件 (KnX/KnY/KnM/KnS) 指定为(D)，不能得到余数；若被除数为负数，余数即为负数。



编程示例:



当 M0 置位时，将被除数 D100 的内容除以除数 D110 的内容后存放在 D120 中，
假如 D100=K5，D110=K2，则 D120=K2，商存放在 D121，D121=K1。

INC 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
INC	BIN 增 1	16	否	INC D	3
INCP		16	是		3
DINC		32	否		5
DINCP		32	是		5

指令每执行一次，D中的数值增加 1。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
D									√	√	√	√	√	√	√	√

16 位运算时，32，767 再加 1 变为-32，768；32 位运算时，2，147，483，647 再加 1 变为-2，147，483，648。

本指令对 0 标志、进位、借位标志都不刷新。

编程示例:



DEC 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DEC	BIN 减 1	16	否	DEC D	3
DECP		16	是		3
DDEC		32	否		5
DDECP		32	是		5

指令每执行一次，D的数值减 1。

16 位运算时，-32，768 再减 1 变为 32，767；32 位运算时，-2，147，483，648 再减 1 变为 2，147，483，647。

本指令对 0 标志、进位、借位标志都不刷新。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
D									√	√	√	√	√	√	√	√

进行 32bit 运算时，指令中D变量地址为为低 16bit 地址，相邻高编号地址单元为高 16bit，编程时防止重复或误覆盖。

编程示例:



WAND 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
WAND	逻辑按位与	16	否	WAND (S1) (S2) (D)	7
WANDP		16	是		7
DAND		32	否		13
DANDP		32	是		13

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√	√		√	√	√	√	√	√	√	√	√
(S2)					√	√		√	√	√	√	√	√	√	√	√
(D)									√	√	√	√	√	√	√	√

本指令执行时，将(S1)和(S2)中 BIN 值的各位对应作“逻辑与”运算，将结果存入变量。

逻辑的‘与’（AND）运算的规则为任一为0 结果为0。 $1 \wedge 1=1$ $1 \wedge 0=0$ $0 \wedge 1=0$ $0 \wedge 0=0$

编程示例:



WOR 指令

指令描述:

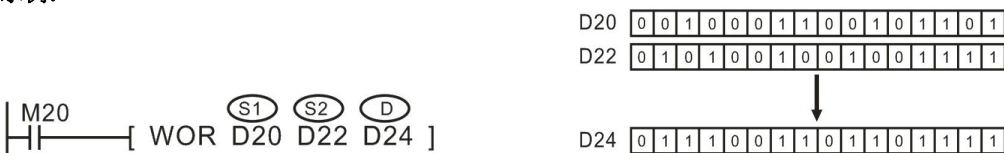
名称	功能	位数(bits)	是否脉冲型	指令格式	步数
WOR	逻辑按位或	16	否	WOR (S1) (S2) (D)	7
WORP		16	是		7
DOR		32	否		13
DORP		32	是		13

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√	√		√	√	√	√	√	√	√	√	√
(S2)					√	√		√	√	√	√	√	√	√	√	√
(D)									√	√	√	√	√	√	√	√

本指令执行时，将(S1)和(S2)中 BIN 值的各位对应作“逻辑或”运算，将结果存入(D)变量。

逻辑的‘或’（OR）运算的规则为任一为1 结果为1。 $1 \vee 1=1$ $1 \vee 0=1$ $0 \vee 1=1$ $0 \vee 0=0$

编程示例:



WXOR 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
WXOR	逻辑按位异或	16	否	WXOR (S1) (S2) (D)	7
WXORP		16	是		7
DXOR		32	否		13
DXORP		32	是		13

本指令执行时，将(S1)和(S2)中 BIN 值的各位对应作“逻辑异或”运算，将结果存入(D)变量。

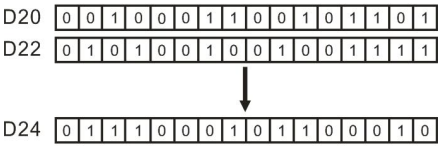
逻辑的“异或”(XOR)运算的规则为两者相同结果为 0，两者不同结果为 1。

1(1=0) 1(0=1) 0(1=1) 0(0=0)

上述三个指令的操作数适用变量类型如下表，当为 32bit 指令时，寄存器变量则占用后续相邻地址的共 2 个单元：

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√	√		√	√	√	√	√	√	√	√	√
(S2)					√	√		√	√	√	√	√	√	√	√	√
(D)									√	√	√	√	√	√	√	√

编程示例:



NEG 指令

指令描述:

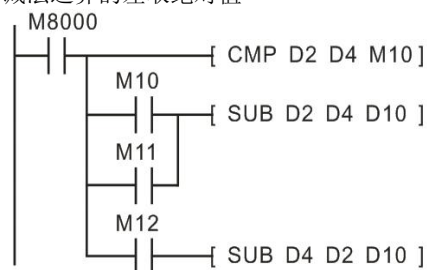
名称	功能	位数(bits)	是否脉冲型	指令格式	步数
NEG	求补	16	否	NEG $\textcircled{D}$	3
NEGP		16	是		3
DNEG		32	否		5
DNEGP		32	是		5

需要触点驱动，有 1 个操作变量。将 $\textcircled{D}$ 的数值逐位取反、再加 1，存回 $\textcircled{D}$ 中。此指令一般用脉冲执行型指令。

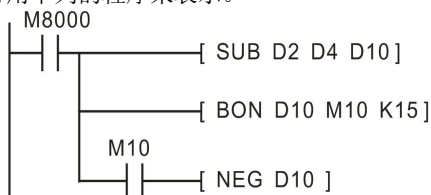
使用 NEG 指令，可得到与负的 BIN 值相对应的绝对值。

编程示例:

减法运算的差取绝对值



若 $D2 > D4$ 时，M10=On。若 $D2 = D4$ 时，M11=On。若 $D2 < D4$ 时，M12=On。由此可保证 D10 为正值此程序可用下列的程序来表示。



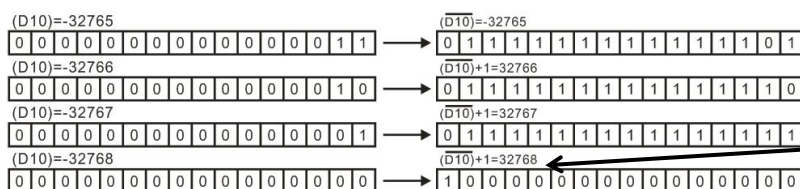
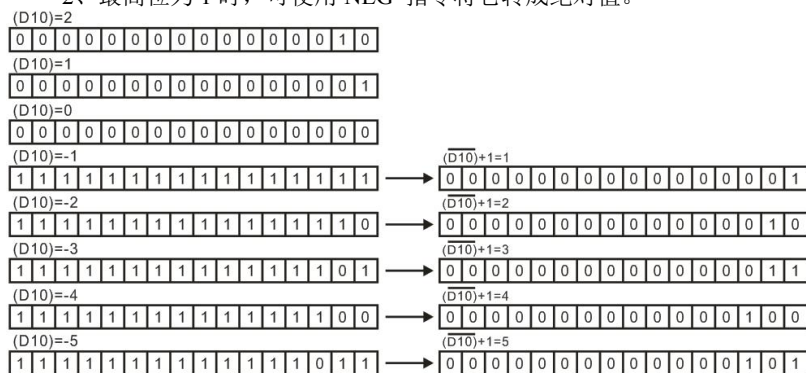
当 D10 的 bit15 为 "1" 时（表示 D10 为负数），M10=On，用 NEG 指令将 D10 取补码可得到 D10 的绝对值。

上述两例中假如 $D2=K4$ ， $D4=K8$ ；或者 $D2=K8$ ， $D4=K4$ ，D10 的结果均为 K4。

补充说明：负数的表现及绝对值：

1、正负数是以寄存器最上位（最左边）的位内容来表现，为 "0" 时为正数、为 "1" 时为负数。

2、最高位为 1 时，可使用 NEG 指令将它转成绝对值。



绝对值的最大范围只能达到 32767

5.2.6 高速处理指令

REF 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
REF	输入输出刷新	16	否	REF (D) (n)	5
REFP		16	是		5

将(D)地址开始的(n)个元件状态进行立即更新。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(D)	√	√														
(n)	常数, n=8~256, 且为 8 的倍数															

由于 PLC 访问端口是按字节访问的特性, 故要求:

(D)的地址应为 X0、X10、…Y0、Y10、。。。等最低位为 0 的编号元件;

(n)的值必须是 8 的倍数 (n=8~256)。

正常情况下, 输入端口 X 的状态读取在每次程序开始执行扫描之前进行, 输出端口 Y 的状态刷新则在每次程序执行扫描完毕 (执行到 END) 之后批次进行, 这样 IO 处理会有一定的延迟。

若应用中需要最新的输入信息以及希望立即输出运算结果时, 可以使用立即刷新指令 REF。

可用在 FOR~NEXT 指令之间、CJ 指令之间等。

可用于中断子程序中进行输入输出刷新—获取最新的输入信息并及时输出运算结果。

实际的输入端口状态变化延迟决定于输入元件的滤波时间, X0~X7 有数字滤波功能, 滤波时间在 0~60ms 范围内可设 (REFF 指令), 其余 IO 端口为硬件滤波, 滤波时间约 10ms。具体参数请参考可编程控制器的用户手册。

实际的输出端口状态变化延迟决定于输出元件 (如继电器) 的响应时间。输出刷新中的输出接点将在输出继电器 (晶体管) 应答时间后动作。

继电器输出型的应答滞后时间约为 10ms (最大 20ms), 晶体管输出型高速输出口约 10μs、普通点输出口约 0.5ms。具体参数请参考可编程控制器的用户手册。

编程示例:

指令示例 1:

```

X20
| |
| |---[ REF X0 K16 ]

```

执行上图程序时, 若 X20 为 ON 状态, 会立即读取 X0~X17 的输入点状态, 更新输入信号, 没有产生输入延迟。

指令示例 2:

```

X0
| |
| |---[ REF Y0 K16 ]

```

执行上图程序时, 若 X0 为 ON 状态, 会立即将 Y0~Y17 的状态进行刷新, 输出信号立即更新, 不必到 END 指令才输出。

REFF 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
REFF	输入刷新（带滤波器设定）	16	否	REFF (n)	3
REFFP		16	是		3

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(n)	常数，n=0~60，单位为 ms															

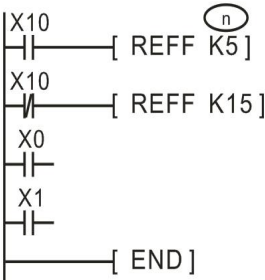
将 X0~X7 输入端口的滤波时间常数设定为(n)。

可编程控制器中，X0~X7 使用了数字滤波器，默认的滤波时间常数由 D8020 设定，通过 REFF 指令可将 D8020 改变为 0~60ms。其余的 X 端口则只有硬件 RC 滤波，滤波时间常数约为 10ms，不能修改；

当使用了高速计数器，或 X 输入端中断功能，则相关端口的滤波时间自动为最短时间，无关的端口的滤波时间仍为原设定值。

亦可用 MOV 指令直接对 D8020 进行赋值改变滤波时间。

编程示例:



X10 为 ON 时，将 X0~X7 的输入滤波时间设为 5ms，X10 为 OFF 时，将 X0~X7 的输入滤波时间设为 15ms。

MTR 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
MTR	矩阵输入	16	否	MTR (S) (D1) (D2) (n)	9

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)	√															
(D1)		√														
(D2)		√	√	√												
(n)	常数, n=2~8															

此指令只能用于晶体管输出型 PLC，通过将 8 个 X 端口与若干个 Y 端口组成矩阵输入网络，以扩大输入信号的通道数。其中：

(S)为矩阵扫描输入的硬件 X 端口的起始地址，要求为 X0、X10…等最低位为 0 的编号元件，占用连续 8 个；

(D1)为矩阵扫描输出的硬件 Y 端口的起始地址，要求为 Y0、Y10…等最低位为 0 的编号元件，占用连续 n 个

(n=2~8)；

(D2)为矩阵扫描读取状态的存放单元的起始地址，要求为 Y0、M0、S0 等最低位为 0 的编号元件；

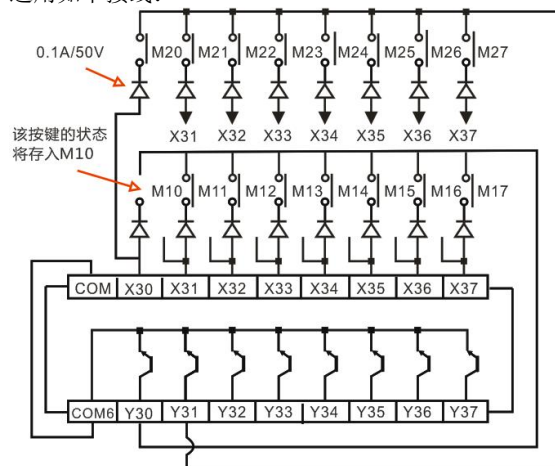
(n)为矩阵扫描的列数，即扫描用 Y 输出的个数。

本指令的条件接点一般都使用常 On 接点 M8000。

编程示例:



适用如下接线:



考虑到 X 输入滤波应答延迟 10ms，Y30、Y31 输出按每 20ms 顺序中断，进行即时输入输出处理；

每次自动读取操作完成后，标志 M8029 置 ON 一个扫描周期；

若通过 8 点 X 输入和 8 点晶体管 Y 输出，可获得最大 64 点的扫描输入，但是此时所有输入的读取需要 $20\text{ms} \times 8$ 列 = 160ms 时间，不适应高速输入操作，故一般使用 X20 以后的端口作扫描输入；

该指令在程序中只能使用一次。

DHSCR 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DHSCR	比较复位（每次计数时，将高速计数器的计数值和指定值比较，然后立即复位外部输出(Y)）	32	否	DHSCR (S1) (S2) (D)	13

当(S2)计数器的当前值等于设定值(S1)时，立即复位(D)。其中：

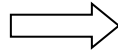
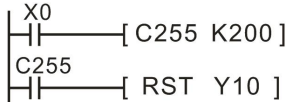
(S1)为设定的比较值，其值的宽度（bit 位数）决定于(S2)计数器的位数；

(S2)变量必须为高速计数器 C235~C255，因涉及的计数器均为 32bit 计数器，故必须采用 32bit 指令 DHSCR；

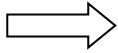
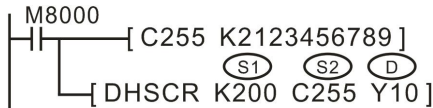
(D)为比较结果的存放单元：当为 Y0~Y17 范围端口时，为立即输出；当为 Y20 以后的端口时，会等到本次用户程序扫描完毕才会输出；当为 M、S 变量时，也为立即刷新。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√	√		√	√	√	√	√	√	√	√	√
(S2)													√			
(D)		√	√	√												

编程示例:



Y10 的动作延迟与扫描时间有关



Y10 立即动作

指令使用说明:

HSCR 指令的动作原理和 HSCS 指令相似，只是 HSCR 的比较输出动作与 HSCS 指令刚好相反，即计数器的值达到相等时，指定的输出复位，因此使用中的一些规定可参考 HSCS 的说明。

DHSCS 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DHSCS	比较复位（每次计数时，将高速计数器的计数值和指定值比较，然后立即置位外部输出(Y)）	32	否	DHSCS (S1) (S2) (D)	13

当(S2)计数器的当前值等于设定值(S1)时，立即置位(D)。其中：

(S1)为设定的比较值，其值的宽度（bit 位数）决定于(S2)计数器的位数；

(S2)变量必须为高速计数器 C235~C255，因涉及的计数器均为 32bit 计数器，故必须采用 32bit 指令 DHSCS；

(D)为比较结果的存放单元，也可以是调用计数中断子程序：当为 Y0~Y17 范围端口时，为立即输出；当为 Y20 以后的端口时，会等到本次用户程序扫描完毕才会输出；当为 M、S 变量时，也为立即刷新；

当(D)项为 I010~I060 时，即为调用高速计数器中断 0~5 的子程序。当然必需编写好相应的中断子程序、开启相应中断允许标志和全局中断允许标志等，才能正常响应定时器中断。M8059 置 ON 则禁止了所有的高速计数器中断（I010-I060）。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√	√		√	√	√	√	√	√	√	√	√
(S2)													√			
(D)		√	√	√												

一般指令 Y 输出与 DHSCS 指令 Y 输出的差异：以（指令举例一）为例。

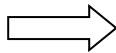
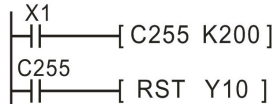
1. 当 C255 的现在值由 99→100 变化时，C255 接点立即导通，但执行到 OUT Y10 时，Y10 仍会受扫描周期影响，在 END 后才输出。
2. 当 C255 的现在值由 99→100 及 101→100 变化时，DHSCS 指令输出 Y10 是以中断方式立即输出到外部输出端，与 PLC 扫描周期无关。但仍会受输出模块继电器(10ms)或晶体管(10us)的输出延迟。

高速计数中断指针与设置：

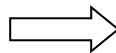
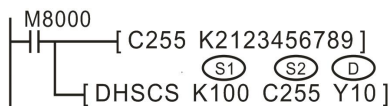
操作数	中断禁止指令
I010	M8059
I020	
I030	
I040	
I050	
I060	

编程示例:

编程示例 1:

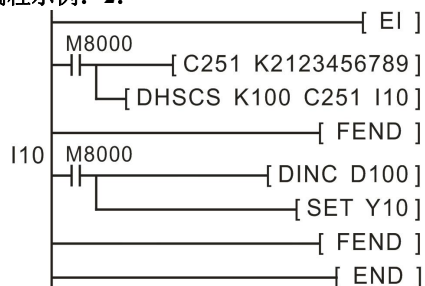


Y10 的动作延迟与扫描时间有关



Y10 立即动作

编程示例：2：



DHSCS 指令的 D 操作数范围也可指定 IO 口 0，口=1-6，作为计数器计数到达时，发生中断，执行该中断服务程序。

如果 M8059 置 ON，则禁止了所有的高速计数器中断。

注意此时的 D 装置用 I010 和 Y、M、S 输出点的 ON 信号区别：

1、用 Y 轴输出点：若 C251 的当前值变为 99→100 或 101→100 时，Y 立即置 ON，且一直保持 ON 状态，之后即使 C251 与 K100 的比较结果变成不相等，Y 仍然保持 ON 状态，除非有另外的复位指令操作；

2、用 I010：若 C251 的当前值变为 99→100 或 101→100 时，I010 只会产生一次中断，不会常 ON。

指令使用说明：

使用 HSCS 指令时，应保证所使用的计数器已被启用（见指令举例一），否则该计数器的值将不会有变化；

计数器是以中断方式响应计数器的输入信号，及时比较，若本次比较时满足匹配关系，比较输出立即置位。例如指令举例一中，若 C255 的当前值变为 99→100 或 101→100 时，Y10 立即置位，且一直保持该状态，之后即使 C255 与 K100 的比较结果变成不相等，Y10 仍然保持 On 状态，除非有另外的复位指令操作；

指令的比较输出只决定于脉冲输入时的比较结果动作，即使采用 DMOV、DADD 等指令改写高速计数器 C235~C255 的内容，若没有脉冲输入，比较输出也不会变化；单纯的指令驱动能流也不能改变比较结果；

指令输出若为 Y 端口，必须为 Y0~Y17 范围，这样才能保证输出得到立即响应；多次驱动 HSCS 指令或与 HSCR、HSZ 指令同时驱动，对象输出 Y 的高 2 位作为同一序号的软元件。例：使用 Y000 时为 Y000~Y007，Y010 时为 Y010~Y017 等；

当 HSCS 指令的输出目标为中断 I010~I060 时，每个中断号只能使用 1 次，不可重复。

HSCS、HSCR、HSZ 与普通指令一样可以多次使用，但这些指令同时驱动的个数限制在总计 6 个指令以下。

DHSZ 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DHSZ	区间比较	32	否	DHSZ (S1) (S2) (S) (D)	17

根据计数器(S)的当前值，与设定的比较区间(S1)和(S2)进行比较，将比较结果立即输出到以(D)地址起始的 3 个单元中。其中：

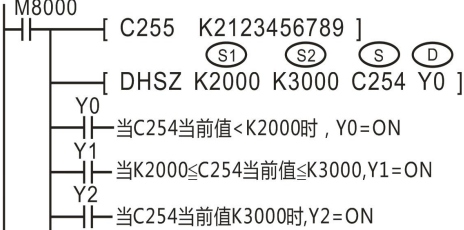
(S1)为设定的比较区间区间下限，其值的宽度（bit 位数）决定于(S)计数器的位数，其值必须不大于(S2)，即 $(S1) \leq (S2)$;

(S2)为设定的比较区间区间上限，其值的宽度（bit 位数）决定于(S)计数器的位数，其值必须不小于，即 $(S1) \leq (S2)$;

(S)变量必须为高速计数器 C235~C255，因涉及的计数器均为 32bit 计数器，故必须采用 32bit 指令 DHSZ；
(D)为比较结果的存放单元，占用以(D)起始的 3 个连续地址的单元：当为 Y0~Y17 范围端口时，为立即输出；当为 Y20 以后的端口时，会等到本次用户程序扫描完毕才会输出；当为 M、S 变量时，也为立即刷新。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√	√		√	√	√	√	√	√	√	√	√
(S2)					√	√		√	√	√	√	√	√	√	√	√
(S)													√			
(D)		√	√	√												

编程示例:



指令使用说明:

本指令的动作原理和 HSCS、HSCR 等指令相似，差别是采用了两个比较值，比较输出使用了 3 个连续的地址单元，因此使用中的一些规定可参考 HSCR 的使用说明；HSZ 指令也是以中断方式进行工作的，只有当计数器对应的输入端有计数脉冲时，比较才会进行，对应的输出才会被刷新。

SPD 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
SPD	脉冲密度	16	否	SPD (S1) (S2) (D)	7

SPD 为基本功能:

- (S1)脉冲信号输入端口, 只能为 X0~X5;
- (S2)为设定的脉冲检测时间长度(ms)1~32767;
- (D)+0: 为在 S2 时间内的脉冲个数, 为 16 位的数据;
- (D)+1: 本次时间段对脉冲的计数。
- (D)+2: 用于测定剩余时间(ms)。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)	√															
(S2)					√	√		√	√	√	√	√	√	√	√	√
(D)												√	√	√	√	√

编程示例:



- X0 为脉冲信号输入端口;
- D0 为单位时间(ms);
- D10 为 D0 时间段总脉冲个数;
- D11 为 D0 时间段正在计的数据;
- D12 为时间段剩余的时间(ms);

PLSY 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
PLSY	脉冲输出	16	否	PLSY (S1) (S2) (D)	7
DPLSY		32	否		13

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)								√	√	√	√	√	√	√	√	√
(S2)								√	√	√	√	√	√	√	√	√
(D)		√														

编程示例:



1、以指定的频率产生定量脉冲的指令。

(S1): 指定频率;

允许设定范围: 16 位指令的频率范围是 1-32767 (HZ); 32 位指令的频率范围是 1-200000 (HZ)。

在指令执行中更改(S1)指定的字软元件的内容, 输出频率也随之发生改变。

(S2): 指令产生脉冲量;

允许设定范围: 16 位指令: 1-32767 (PLS); 32 位指令: 1-2147483647 (PLS)。

将该值指定为零时, 则对产生的脉冲不做限制。

DPLSY 指令中, 将 (D1, D0) 设置为脉冲量。

在指令执行过程中, 变更(S2)指令的字软元件内容后, 将从下一个指令驱动开始执行变更内容。

(D): 指令脉冲输出端口;

对于 LX3V/LX3VP/LX3VE 型可指定 Y0~Y3; LX3V(1S)型可指定 Y0~Y1;

在 LX 系列可编程控制器中, 为了输出高频脉冲, 可编程控制器的输出晶体管上必须是额定负载的电流。

在编程示例: 中, X0 为 OFF 后, 输出中断, 再次置 ON 时, 从初始状态开始动作, 发出连续脉冲时, X0 为 OFF, Y0 也为 OFF。

脉冲的占空比为 50%ON, 50%OFF。输出控制不受扫描周期的影响, 采用中断处理。

设定脉冲发完后, 执行结束标志 M8029 动作。

指令执行中, 涉及的系统变量有:

1. D8141 (高字节), D8140 (低字节): Y000 输出脉冲数。反转时减少。(使用 32 位)
2. D8143 (高字节), D8142 (低字节): Y001 输出脉冲数。反转时减少。(使用 32 位)
3. D8151 (高字节), D8150 (低字节): Y002 输出脉冲数。反转时减少。(使用 32 位)
4. D8153 (高字节), D8152 (低字节): Y003 输出脉冲数。反转时减少。(使用 32 位)
5. M8145: Y000 脉冲输出停止 (立即停止)
6. M8146: Y001 脉冲输出停止 (立即停止)
7. M8152: Y002 脉冲输出停止 (立即停止)
8. M8153: Y003 脉冲输出停止 (立即停止)
9. M8147 : Y000 脉冲输出中监控 (BUSY/READY)
10. M8148 : Y001 脉冲输出中监控 (BUSY/READY)
11. M8149 : Y002 脉冲输出中监控 (BUSY/READY)
12. M8150 : Y003 脉冲输出中监控 (BUSY/READY)

PWM 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
PWM	脉宽调制	16	否	PWM (S1) (S2) (D)	7

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√	√		√	√	√	√	√	√	√	√	√
(S2)					√	√		√	√	√	√	√	√	√	√	√
(D)		√														

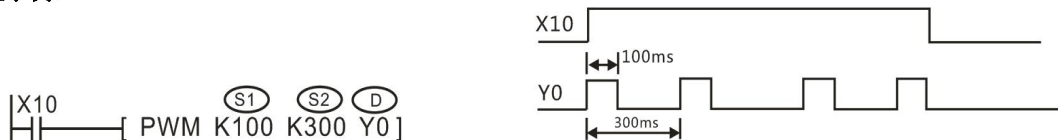
由于继电器不适合高频率动作，只有晶体管输出型 PLC 才适合使用该指令。指令功能是以(S1)指定的脉冲宽度，(S2)指定的脉冲周期，由(D)指定的端口持续输出脉冲。其中：

(S1)为设定的输出脉冲宽度，必须有(S1)≤(S2)，设定范围为 0~32，767ms；

(S2)为设定的脉冲输出周期，必须有(S1)≤(S2)，设定范围为 1~32，767ms；

(D)为脉冲输出端口，对于 LX3V(2N)/LX3V 型可指定 Y0~Y3；LX3V(1S)型可指定 Y0~Y1，不要与 PLSY，PLSR 指令的输出端口重复。本指令是以中断方式执行，当指令能流为 OFF 时，输出停止。(S1)、(S2)可在 PWM 指令执行时更改。

编程示例:



PWM 千分比模式

千分比模式：千分比模式是将周期参数，平均等份为 1000 等分，使用千分比模式方式的设置，将对应控制位置 ON，对应如下：

输出位	Y0	Y1	Y2	Y3
控制位	M8134	M8135	M8136	M8137

对应的标志位设置为 ON，设置为 OFF 则为正常模式。

由于继电器不适合高频率动作，只有晶体管输出型 PLC 才适合使用该指令。指令功能是以(S1)指定的占空比，(S2)指定的脉冲周期，由(D)指定的端口持续输出脉冲。其中：

(S1)为设定的输出脉冲占空比，必须有(S1)≤(S2)，设定范围为 0~1000；

(S2)为设定的脉冲输出周期，必须有(S1)≤(S2)，设定范围为 1~32，767ms；

(D)为脉冲输出端口对于 2N/3V/3VP/3VE 型可指定 Y0~Y3；1S 型可指定 Y0~Y1，不要与 PLSY，PLSR 指令的输出端口重复。本指令是以中断方式执行，当指令能流为 OFF 时输出停止。(S1)(S2)可在 PWM 指令执行时更改。

举例说明:



周期设置为 100ms，占空比如果设置为 500，那么输出为 50ms 高电平，50ms 低电平；占空比如果设置为 100，那么输出为 10ms 高电平，90ms 低电平；占空比如果设置为 900，那么输出为 90ms 高电平，10ms 低电平；

计算公式为：t (ms) = T0 (ms) * K / 1000

高电平时间 (ms) = 设置的周期时间 (ms) × 占空千分比 / 1000

低电平时间 (ms) = 周期 (ms) - 高电平时间 (ms)

PLSR 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
PLSR	带加减速脉冲输出	16	否	PLSR (S1) (S2) (S3) (D)	7
DPLSR		32	否		17

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√	√		√	√	√	√	√	√	√	√	√
(S2)					√	√		√	√	√	√	√	√	√	√	√
(S3)					√	√		√	√	√	√	√	√	√	√	√
(D)		√														

由于继电器不适合高频率动作，只有晶体管输出型 PLC 才适合使用该指令。该功能是指带加减速功能的固定尺寸传送用脉冲输出指令。其中：

(S1)为设定的输出脉冲的最高频率，设定范围为 10~100, 000Hz；

(S2)为设定的输出脉冲数，16bit 指令，设定范围为 110~32, 767；32bit 指令，设定范围为 110~2, 147, 483, 647；设定的脉冲数小于 110 时，不能正常输出脉冲；

(S3)为设定的加减速时间，范围 10~32000 (ms)，减速时间与加速时间相同，ms 单位，设定时请注意；

(D)为脉冲输出口，对于 LX3V(2N)/LX3V 型可指定 Y0~Y3；LX3V(1S)型可指定 Y0~Y1，不要与 PLSY 指令的输出端口重复。

使用说明:

本指令是以中断方式执行，不受扫描周期的影响；

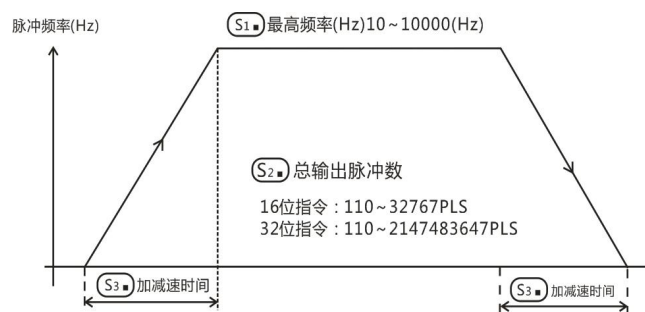
当指令能流为 OFF 时，将减速停止；当能流由 OFF→ON 时，脉冲输出处理重新开始；

在脉冲输出过程中，改变操作数，对本次输出没有影响，修改的内容在指令下次执行的时候生效。指令执行完毕，M8029 标志置为 ON；

与 PWM 指令的输出端口号不能重复；

再次启动 PLSR 指令时，需在上次脉冲输出操作结束（Y0 结束时 M8147=0；Y1 结束时 M8148=0；Y2 结束时 M8149=0；Y3 结束时 M8150=0；）后，再延迟 1 个扫描周期，方可再启动该指令。

编程示例:



各输出端口对应的特殊寄存器如下：

寄存器		定 义	备 注
D8140	低字	PLSY 或 PLSR 指令中设定的输出至 Y0 口的脉冲总数	可用指令： DMOV K0 D81xx 进行清除操作
D8141	高字		
D8142	低字	PLSY 或 PLSR 指令中设定的输出至 Y1 口的脉冲总数	
D8143	高字		
D8150	低字	PLSY 或 PLSR 指令中设定的输出至 Y2 口的脉冲总数	
D8151	高字		
D8152	低字	PLSY 或 PLSR 指令中设定的输出至 Y3 口的脉冲总数	
D8153	高字		
D8136	低字	已向 Y0 及 Y1 输出的脉冲个数的累计值	
D8137	高字		

本指令的输出频率范围为 10 ~ 100 , 000Hz 。最高速度或加减速的高速速度转换超出该范围时，将自动转换（上升或下降）至范围内的数值后执行。但是，实际能够输出的输出频率最低值取决于以公式：

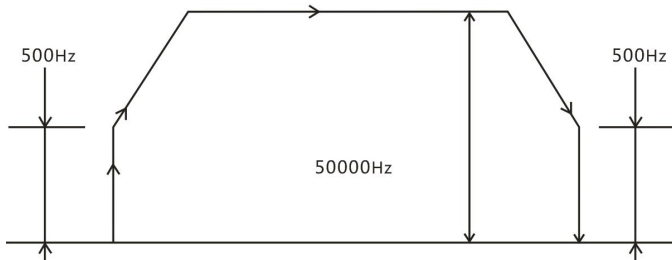
$$\sqrt{\frac{\text{最高频率 (S1) Hz} \div (2 \times (\text{加减速时间 (S2) Ms} \div 1000))}{\text{输出脉冲频率的最低频率数}}} = \text{最低频率数}$$

加速初期和减速末期的频率不得低于上述公式的计算结果。

【例】最高速度：50000Hz，加减速时间：100ms。

$$\sqrt{50000 \div (2 \times (100 \div 1000))} = 500\text{Hz}$$

最高频率 S1 指定为 50000Hz 时，加速初期和减速末期的实际输出频率为 500Hz。



指令执行中，涉及的系统变量有：

1. D8141（高字节），D8140（低字节）：Y000 输出脉冲数。反转时减少。（使用 32 位）
2. D8143（高字节），D8142（低字节）：Y001 输出脉冲数。反转时减少。（使用 32 位）
3. D8151（高字节），D8150（低字节）：Y002 输出脉冲数。反转时减少。（使用 32 位）
4. D8153（高字节），D8152（低字节）：Y003 输出脉冲数。反转时减少。（使用 32 位）
5. M8145：Y000 脉冲输出停止（立即停止）
6. M8146：Y001 脉冲输出停止（立即停止）
7. M8152：Y002 脉冲输出停止（立即停止）
8. M8153：Y003 脉冲输出停止（立即停止）
9. M8147：Y000 脉冲输出中监控（BUSY/READY）
10. M8148：Y001 脉冲输出中监控（BUSY/READY）
11. M8149：Y002 脉冲输出中监控（BUSY/READY）
12. M8150：Y003 脉冲输出中监控（BUSY/READY）

PLSR2 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
PLSR	多段速脉冲输出	16	否	PLSR2 (S1) (D1) (D2)	7

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)														√		
(D1)		√														
(D2)		√	√	√												

PLSR2 指令是以表格形式设置参数，按指定的端口、频率、运行方向和加减速时间分段产生相对、绝对位置脉冲的指令，令伺服执行机构在当前位置的基础上作给定偏移量的运动。只有晶体管输出 PLC 才能使用该指令。其中：

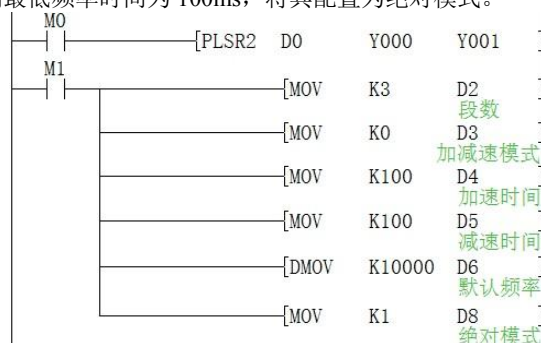
- (S1) 为参数地址，是以 Dn 为起始地址的一段区域；
- (D1) 为脉冲输出端口，可指定 Y0-Y3；
- (D2) 运行方向输出端口或位变量，输出为 ON 状态：表示为正向运行；否则为反向运行。



地址偏移	内容	说明
(S1)+0	表格标识	保留
(S1)+1	表格版本	保留
(S1)+2	脉冲总段数	1-12
(S1)+3	加减速模式	0：前加模式；1：后加模式
(S1)+4	加速时间	50-32000ms
(S1)+5	减速时间	50-32000ms
(S1)+6	默认频率	10HZ-200KHZ
(S1)+7		
(S1)+8	相对/绝对模式	0：相对；1：绝对
(S1)+9 ~ (S1)+19	保留	保留
(S1)+20	第一段脉冲频率	10HZ-200KHZ
(S1)+21		
(S1)+22	第一段脉冲总数	无
(S1)+23		
(S1)+24	第一段等待条件	0：脉冲发送完成； 1：等待时间； 2：等待信号； 3：触发信号。（与[等待条件]以及[等待寄存器]配合使用）
(S1)+25	第一段等待寄存器类型	等待条件与等待寄存器类型对应关系： 脉冲发送完成：无 等待时间：=0：D 寄存器；=1：常数； 等待信号：=0：X 位寄存器；=1：M 位寄存器； 触发信号：=0：X 位寄存器；=1：M 位寄存器。
(S1)+26	第一段 常数值/等待寄存器编号	无
(S1)+27		
(S1)+28	保留	保留
(S1)+29	保留	保留
。 。 。	。 。 。	。 。 。
(S1)+10+n*10	第 N 段脉冲频率	10HZ-200KHZ

编程示例：

下图设置为以 D0 为起始地址的 4 段脉冲的通用参数设置，从最低频率加速到默认频率（100,000Hz）的时间为 100ms，从默认频率减速到最低频率时间为 100ms，将其配置为绝对模式。



每段脉冲的频率个数及等待条件如下表：

段数	脉冲频率	脉冲个数	等待方式	条件
1	10000	10000	等待时间	K100
2	20000	20000	等待信号	M1
3	15000	40000	触发信号	X1
4	30000	60000	无等待条件	K0

按照参数进行配置：



参数说明：

1、(S1)+2：用于设置脉冲段数（单字），数据范围 1-12，支持 1-12 段数。

2、加减速模式：(S1)+3：用来设置加减速模式（单字），设置为 0 时为前加模式，设置为 1 时为后加模式，

其中：

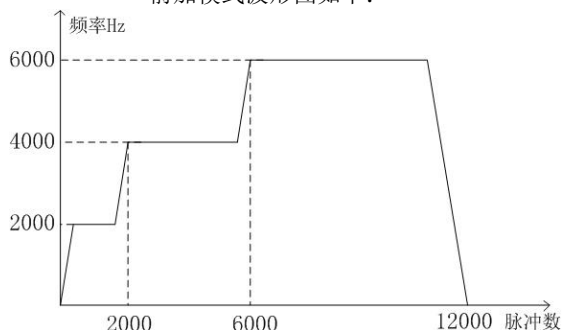
（1）前加模式：多段脉冲时，当上一段脉冲个数按照上一段脉冲输出频率发送完时，已经提前加速至下一段脉冲的频率；

（2）后加模式：多段脉冲时，当上一段脉冲个数按照上一段脉冲输出频率发送完时，才开始加速至下一段

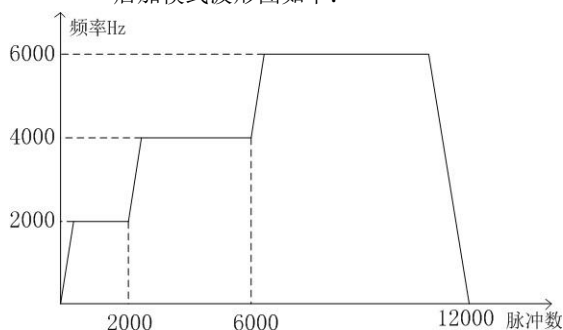
脉冲的频率。

例：现需要三段脉冲，第一段脉冲频率为 2000Hz，脉冲个数为 2000；第二段脉冲频率为 4000Hz，脉冲个数为 4000；第三段脉冲频率为 6000Hz，脉冲个数为 6000。

前加模式波形图如下：



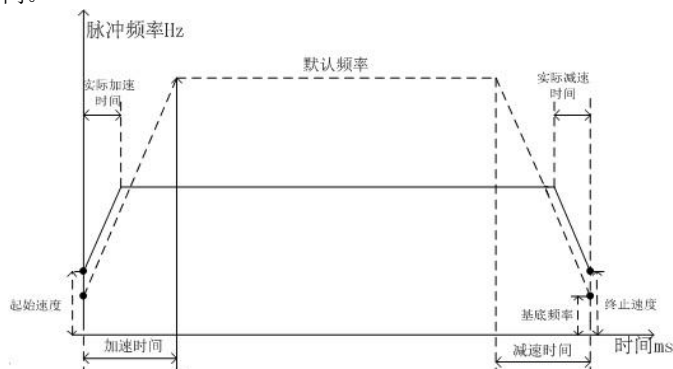
后加模式波形图如下：



3、加减速时间、默认频率：(S1)+4: 加速时间（单字）；(S1)+5: 减速时间（单字）；

(S1)+6: 默认频率（双字）。

本指令的输出频率范围为 10-200000Hz。最高速度或加减速的高速速度转换超出该范围时，将自动转换（上升或下降）至范围内的数值后执行。其中，加速时间指从基底频率加速到默认频率的时间，减速时间指从默认频率减速到基底频率的时间。



4、相对、绝对模式：(S1)+8（单字）：用来设定脉冲设置的参数用于相对模式/绝对模式。当其设置=0 时为相对模式，即脉冲数与当前寄存器为相对位置。当设置=1 时为绝对模式，即脉冲数与当前值寄存器为绝对位置。

输出端口与当前值寄存器对应关系如下：

向【Y000】输出时，当前值寄存器为【D8141（高字节），D8140（低字节）】；

向【Y001】输出时，当前值寄存器为【D8143（高字节），D8142（低字节）】；

向【Y002】输出时，当前值寄存器为【D8151（高字节），D8150（低字节）】；

向【Y003】输出时，当前值寄存器为【D8153（高字节），D8152（低字节）】；

例：现需要三段脉冲，第一段脉冲频率为 2000Hz，脉冲个数为 2000；第二段脉冲频率为 4000Hz，脉冲个数为 4000；第三段脉冲频率为 6000Hz，脉冲个数为 6000。

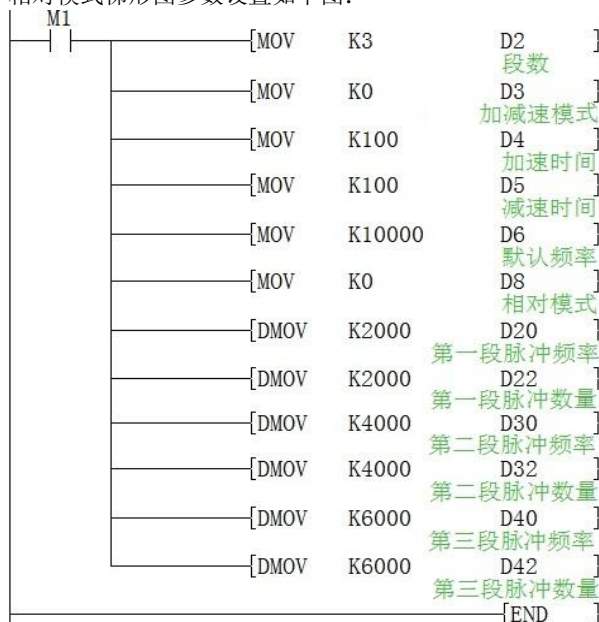
相对模式设置如下表：

段数	脉冲频率	脉冲个数	等待方式	条件
1	2000	2000	无等待条件	K0
2	4000	4000	无等待条件	K0
3	6000	6000	无等待条件	K0

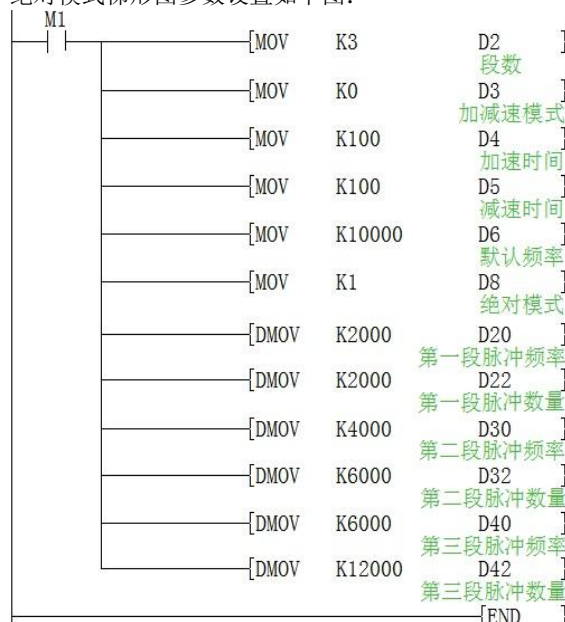
绝对模式设置如下表：

段数	脉冲频率	脉冲个数	等待方式	条件
1	2000	2000	无等待条件	K0
2	4000	4000	无等待条件	K0
3	6000	6000	无等待条件	K0

相对模式梯形图参数设置如下图：



绝对模式梯形图参数设置如下图：

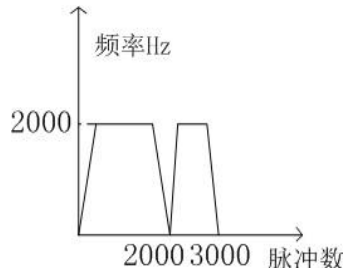


注意：绝对模式下，如果当前段位置与下一段相同，则认为下一段不存在，下一段速度将会降为 0。

下表为绝对模式下，第二段脉冲位置与第一段相同的情况：

段数	脉冲频率	脉冲个数/位置	等待方式	条件
1	2000	2000	无等待条件	K0
2	3000	2000	无等待条件	K0
3	2000	3000	无等待条件	K0

脉冲波形图如下：



5、等待条件

$(S1)+14+10*N$ (单字) 为第 N 段的等待条件， $(S1)+15+10*N$ (单字) 为等待寄存器类型， $(S1)+16+10*N$ (双字) 为等待寄存器编号/常数值。等待条件=0 时为无等待条件，=1 时为等待时间，=2 时为等待信号，=3 时为触发信号。

等待条件与等待寄存器以及等待寄存器编号/常数值配合使用。

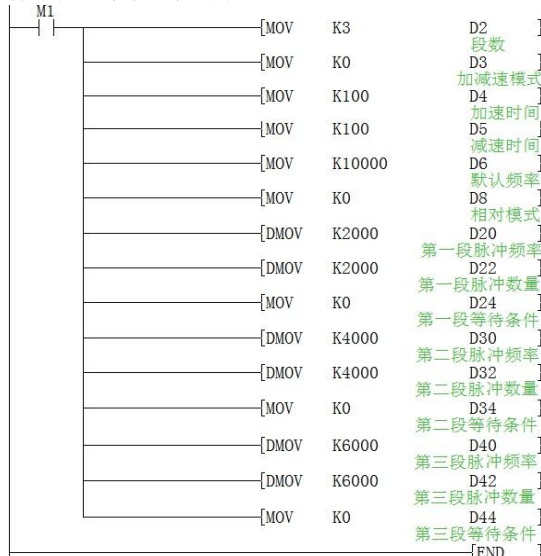
(1) 无等待条件：

$(S1)+14+10*N$ (单字) 为 0 时为无等待条件。执行完本段的设定脉冲个数后立即跳转到后面指定的脉冲段。

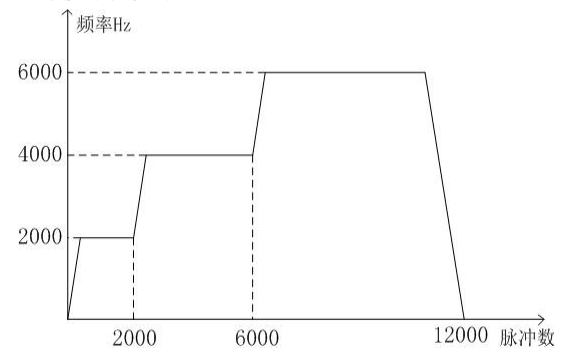
例：现需要三段脉冲，第一段脉冲频率为 2000Hz，脉冲个数为 2000；第二段脉冲频率为 4000Hz，脉冲个数为 4000；第三段脉冲频率为 6000Hz，脉冲个数为 6000，无等待条件。

段数	脉冲频率	脉冲个数	等待方式	条件
1	2000	2000	无等待条件	K0
2	4000	4000	无等待条件	K0
3	6000	6000	无等待条件	K0

梯形图参数设置如下图：



脉冲波形图如下：



(2) 等待时间：

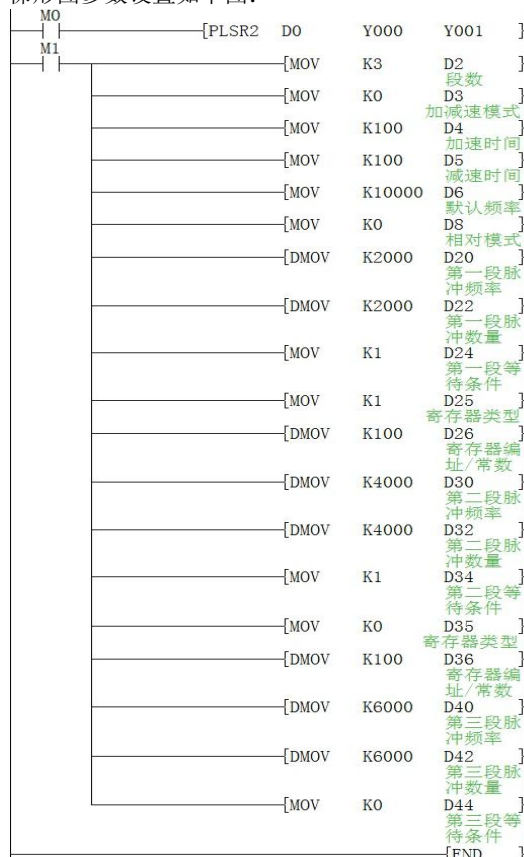
$(S1)+14+10*N$ (单字)=1 时为等待时间。 $(S1)+15+10*N$ (单字)=0 时为等待 D 寄存器，=1 时为等待常数。

当前段脉冲输出完成后开始计时，当计时时间到位后，立即跳转到指定的脉冲段；计时时间可以是常数或寄存器 D 指定，单位：ms (范围 1-65535ms)。

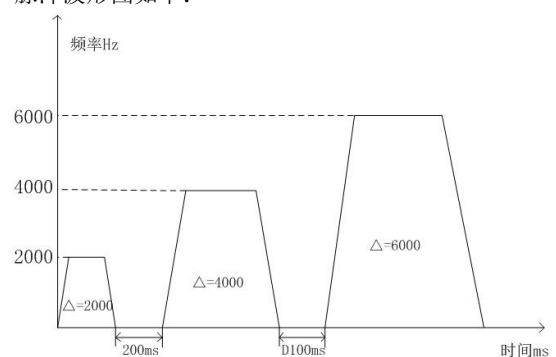
例：现需要三段脉冲，第一段脉冲频率为 2000Hz 脉冲个数为 2000 等待时间 K100 (单位：ms)；第二段脉冲频率为 4000Hz 脉冲个数为 4000 等待时间 D100；第三段脉冲频率为 6000Hz 脉冲个数为 6000 无等待条件。

段数	脉冲频率	脉冲个数	等待方式	条件
1	2000	2000	等待时间	K100
2	4000	4000	等待时间	D100
3	6000	6000	无等待条件	K0

梯形图参数设置如下图：



脉冲波形图如下：



(3) 等待信号:

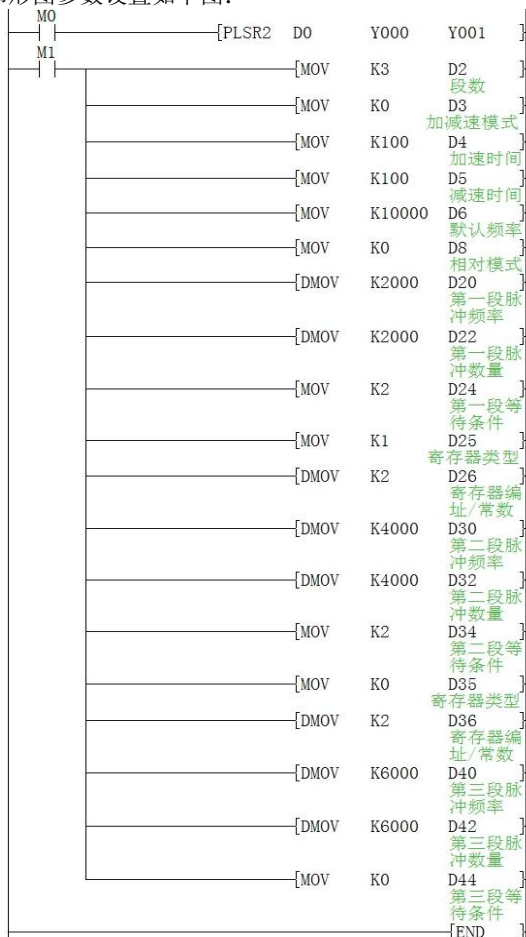
$(S1)+14+10*N$ (单字) =2 时为等待信号。 $(S1)+15+10*N$ (单字) =0 时为等待 X 信号, =1 时为等 M 信号。

当前脉冲段发送完成后, 开始等待‘等待条件’中的位信号, 当位信号为 ON 时, 开始发送下一脉冲段。位信号可以是 X、M 位寄存器。

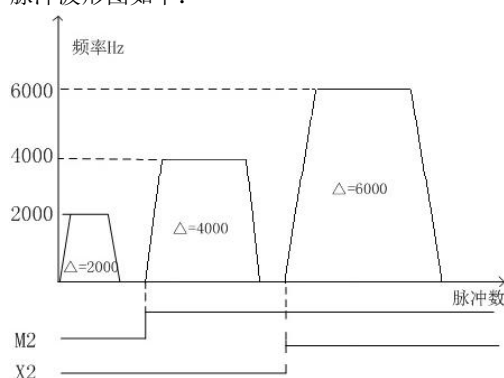
例: 现需要三段脉冲, 第一段脉冲频率为 2000Hz, 脉冲个数为 2000, 等待信号为 M2; 第二段脉冲频率为 4000Hz, 脉冲个数为 4000, 等待信号为 X2; 第三段脉冲频率为 6000Hz, 脉冲个数为 6000, 无等待条件。

段数	脉冲频率	脉冲个数	等待方式	条件
1	2000	2000	等待信号	M2
2	4000	4000	等待信号	X2
3	6000	6000	无等待条件	K0

梯形图参数设置如下图:

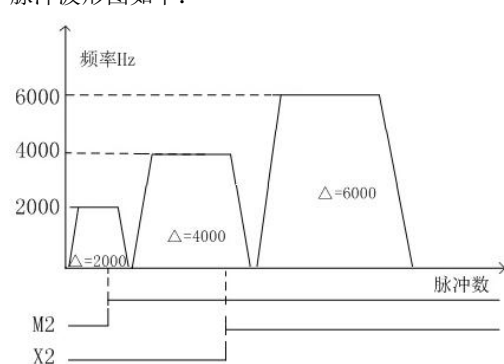


脉冲波形图如下:



若提前收到信号则在当前段结束后马上发送下一段。

脉冲波形图如下:



(4) 触发信号:

$(S1)+14+10*N$ (单字) =3 时为触发信号。

$(S1)+15+10*N$ (单字) =0 时为触发 X 信号, =1 时为触发 M 信号。

当前脉冲段开始发送脉冲后, 如果在当前脉冲数发送结束前, 外部位信号触发工作 (为 ON 状态), 则立即发送下一段脉冲。当前段的脉冲发送结束时, 如果信号没有触发 (为 OFF 状态), 则会继续发送下一段脉冲 (即表示配置好的脉冲段将会按照无等待条件的模式发脉冲, 但是当前段脉冲在发送的过程中收到触发信号, 则会直接加减速到下一段脉冲)。

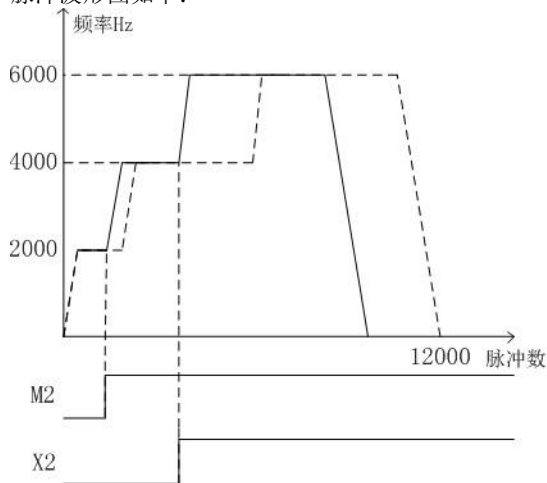
例: 现需要三段脉冲, 第一段脉冲频率为 2000Hz, 脉冲个数为 2000, 触发信号为 M2; 第二段脉冲频率为 4000Hz, 脉冲个数为 4000, 触发信号为 X2; 第三段脉冲频率为 6000Hz, 脉冲个数为 6000, 无等待条件。

段数	脉冲频率	脉冲个数	等待方式	条件
1	2000	2000	触发信号	M2
2	4000	4000	触发信号	X2
3	6000	6000	无等待条件	K0

梯形图参数设置如下图：

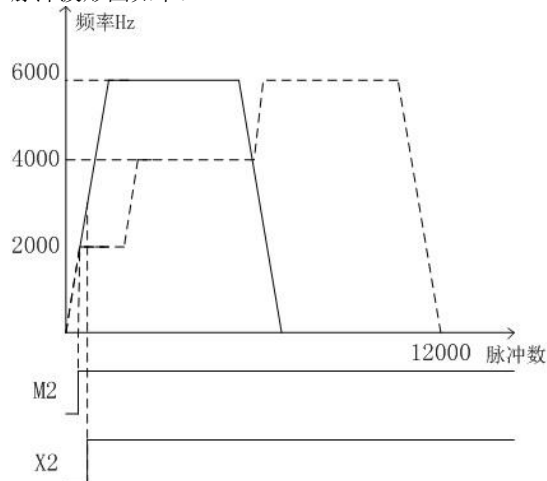


脉冲波形图如下：



若提前收到信号或在加速段收到信号，则在当前段直接加速到下一段脉冲频率。

脉冲波形图如下：



注意：触发信号在中断中进行触发响应。

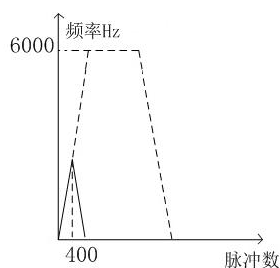
6、脉冲数不足加速到指令频率时：

(1) 单段脉冲：

若脉冲数不足加速到指令频率时，脉冲波形呈三角形。例：

段数	脉冲频率	脉冲个数	等待方式	条件
1	6000	400	无等待条件	K0

脉冲波形图如下：

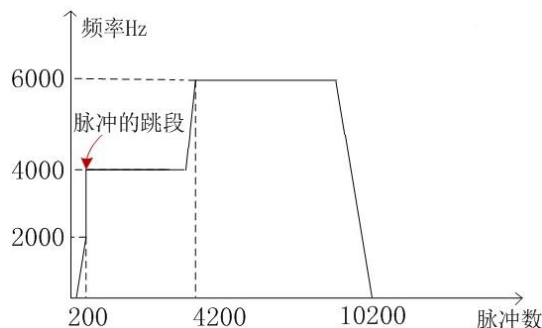


（2）多段脉冲：

前加模式下，如果当前段脉冲个数发送完成，脉冲频率未达到该段设定频率，则会出现脉冲频率的“跳段”，即直接跳变到下一段频率。例：

段数	脉冲频率	脉冲个数	等待方式	条件
1	2000	200	无等待条件	K0
2	4000	4000	无等待条件	K0
3	6000	6000	无等待条件	K0

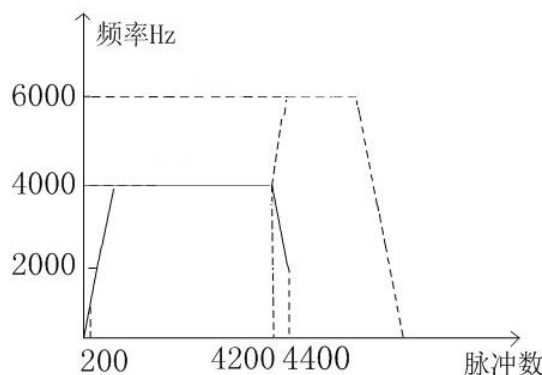
脉冲波形图如下：



后加模式下，如果当前段脉冲个数发送完成，脉冲频率未达到该段设定频率，则会按照当前频率继续进行下一段脉冲的发送，直到发送完脉冲为止。

段数	脉冲频率	脉冲个数	等待方式	条件
1	2000	200	无等待条件	K0
2	4000	4000	无等待条件	K0
3	6000	200	无等待条件	K0

脉冲波形图如下：



注意事项：

- 1、在指令执行过程中，即使改变操作数的内容，也无法在当前运行中表现出来，只能在下一次指令执行时才有效。
- 2、若在指令执行过程中，指令驱动的接点变为 OFF 时，将减速停止，此时执行完成标志 M8029 不会动作；
- 3、定位指令(ZRN/PLSV/DRVI/DRVA/PLSR2)在程序中可以多次使用，但是不能对同一输出口同时进行输出操作。
- 4、当指令的驱动能流 OFF 之后，再次驱动 ON 时，必须在状态位（Y000: [M8147]；Y001: [M8148]；Y002: [M8149]；Y003: [M8150]）OFF 后，经过一个运算周期后，方能再次驱动。
- 5、定位指令再次驱动时，必须有 1 个周期以上的 OFF 时间，若已比上述条件更短的时间内执行再驱动时，将在最初指令执行（运算）时发生[运算错误]。
- 6、请设置足够的脉冲数用于加减速。

PTO 指令

指令描述:

名称	功能	位数 (bits)	是否脉冲型	指令格式	步数
PTO	包络线脉冲输出指令	16	否	PTO (S1) (S2)	5
DPTO		32	否		9

操作数说明:

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)														√		
(S2)		√														

以操作数(S1)为起始地址的数据表说明如下:

地址偏移量	分段	描述
0		分段数目: 1~255 (为 0 则不输出脉冲)
1		记录当前执行到的分段序号
2		包络表执行次数 (-1: 不执行 0: 无限次) (重启生效)
.....		保留
10	#1	初始频率 (频率的变动范围) (0~200, 000)
11		频率增量 (有符号值: -20, 000~20, 000)
12		脉冲个数 (1-4, 294, 967, 295)
13	#2	初始频率 (频率的变动范围) (0~200, 000)
14		频率增量 (有符号值: -20, 000~20, 000)
15		脉冲个数 (1-4, 294, 967, 295)
(以此类推)	#3	(以此类推)

*: 使用 32 位指令 DPTO 时, 地址占双字, 因此地址偏移量为 2。

编程示例:



例如: 用带有脉冲包络的 PTO 来控制一台步进电机, 实现一个简单的加速、匀速和减速或者一个由最多 255 段脉冲波形组成的复杂过程, 而其中每一段波形都是加速、匀速或者减速操作。启动和最终的脉冲频率都是 2KHZ, 最大的脉冲频率是 10KHZ, 要求 4000 个脉冲才能达到期望的电机旋转数。

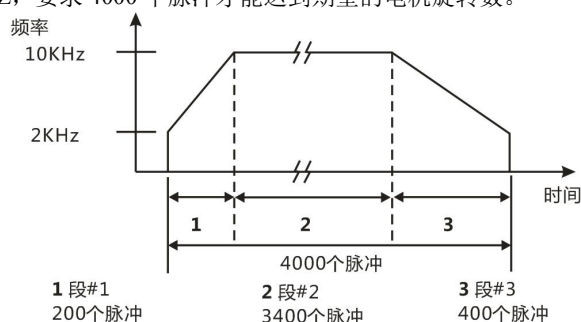


图 1

图 1 给出的包络脉冲示例要求产生一个输出信号波形包括三段: 步进电机加速 (第一段); 步进电机匀速 (第二段) 和步进电机减速 (第三段)。

根据下面的注意事项 4 中给出的频率增量计算公式得出各阶段的频率增量:

- 分段 1 (加速) 频率增量=40;
- 分段 2 (匀速) 频率增量=0;
- 分段 3 (减速) 频率增量= -20;

结合指令和图 1，给出对应的包络表：

分段	寄存器地址	数值	描述
参数设置	D0	3	总段数
	D1	0	记录当前执行到的分段序号
	D2	0	包络表执行次数
#1	D10	2KHZ	初始频率
	D11	40	频率增量
	D12	200	脉冲个数
#2	D13	10KHZ	初始频率
	D14	0	频率增量
	D15	3400	脉冲个数
#3	D16	10KHZ	初始频率
	D17	-20	频率增量
	D18	400	脉冲个数

注意事项：

- 1、以频率为基准，运行过程中启动指令；
- 2、频率变动范围：0~100kHz
- 3、若包络表超出 D 软元件有效范围，则不发送脉冲；
- 4、频率增量计算公式：频率增量=（本段的结束频率 - 本段的初始频率）/ 本段的脉冲个数；
- 5、脉冲的频率间隔（包括段内和段间）相差不能超过 20000Hz，若超出则出现运行错误（错误号 6780），指令不被执行；
- 6、若一条 PTO 指令中出现相邻脉冲(包括分段内和分段之间)的频率之差大于 20kHz，则该 PTO 指令不执行：
 - （1）循环发送模式下，最后一个分段的最后一个脉冲和第一个分段的第一个脉冲算作相邻脉冲；
 - （2）单次发送模式下最后一个分段的最后一个脉冲和第一个分段的第一个脉冲不是相邻脉冲。

5.2.7 循环移位指令

ROL 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
ROL	使 16 位或 32 位数据的各位左移位的指令	16	否	ROL (D) (n)	5
ROLP		16	是		5
DROL		32	否		9
DROLP		32	是		9

将 (D) 的内容循环左移 (n) 位。

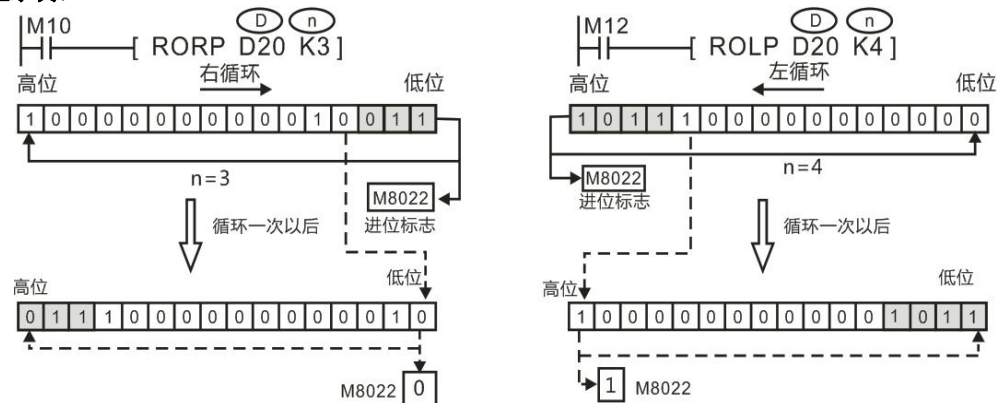
本指令一般使用脉冲执行型指令。

当为 32bit 指令时，寄存器变量则占用后续相邻地址的共 2 个单元：

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(D)								√	√	√	√	√	√	√	√	√
(n)	常数，n=1~16 (16bit)；n=1~32 (32 bit)															

若 (D) 中指定 KnY、KnM、KnS 时，只有 K4 (16bit) 及 K8 (32 bit) 有效；循环移动的最终位被存入进位标志中。

编程示例:



ROR 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
ROR	使 16 位或 32 位数据的各位右移位的指令	16	否	ROR (D) (n)	5
RORP		16	是		5
DROR		32	否		9
DRORP		32	是		9

将 (D) 的内容循环左移 (n) 位。

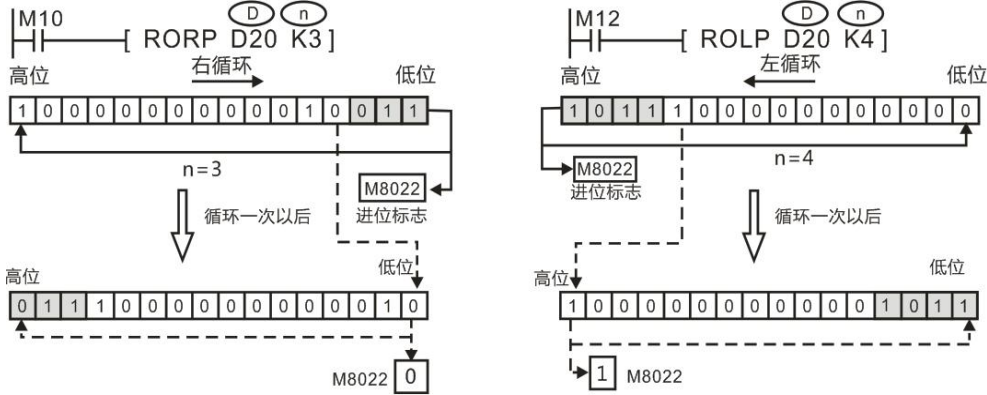
本指令一般使用脉冲执行型指令。

当为 32bit 指令时，寄存器变量则占用后续相邻地址的共 2 个单元：

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(D)								√	√	√	√	√	√	√	√	√
(n)	常数，n=1~16（16bit）；n=1~32（32 bit）															

若 (D) 中指定 KnY、KnM、KnS 时，只有 K4（16bit）及 K8（32 bit）有效；循环移动的最终位被存入进位标志中。

编程示例:



RCL 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
RCL	使 16 位或 32 位数据的各位带进位左移位的指令	16	否	RCL (D) (n)	5
RCLP		16	是		5
DRCL		32	否		9
DRCLP		32	是		9

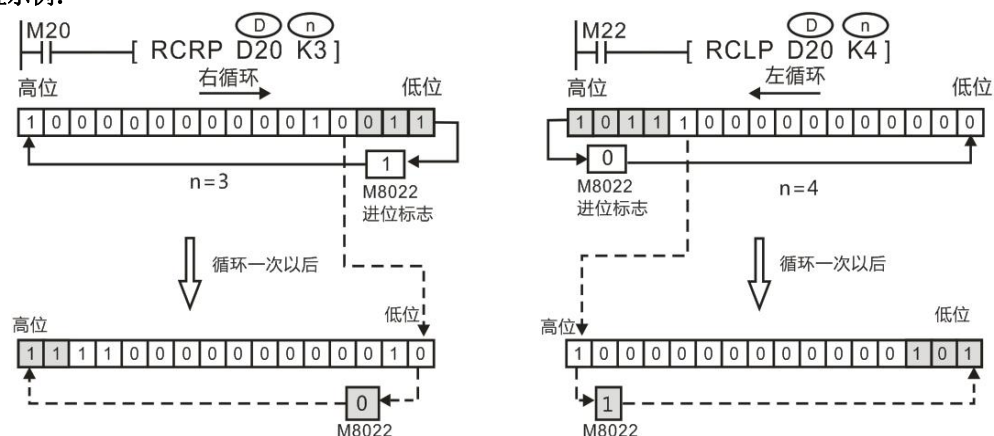
将(D)的内容连同进位标志 M8022 循环右移(n)位。本指令一般使用脉冲执行型指令。

当为 32bit 指令时，寄存器变量则占用后续相邻地址的共 2 个单元：

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(D)								√	√	√	√	√	√	√	√	√
(n)	常数，n=1~16 (16bit)；n=1~32 (32 bit)															

若(D)中指定 KnY、KnM、KnS 时，只有 K4 (16bit) 及 K8 (32 bit) 有效；

编程示例:



RCR 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
RCR	使 16 位或 32 位数据的各位带进位右移位的指令	16	否	RCR (D) (n)	5
RCRP		16	是		5
DRCR		32	否		9
DRCRP		32	是		9

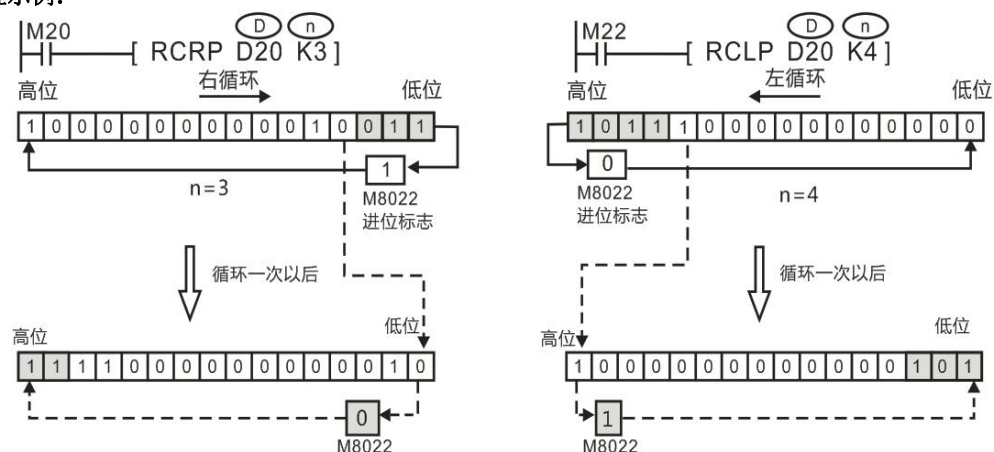
将 (D) 的内容连同进位标志 M8022 循环右移 (n) 位。本指令一般使用脉冲执行型指令。

当为 32bit 指令时，寄存器变量则占用后续相邻地址的共 2 个单元：

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(D)								√	√	√	√	√	√	√	√	√
(n)	常数，n=1~16 (16bit)；n=1~32 (32 bit)															

若 (D) 中指定 KnY、KnM、KnS 时，只有 K4 (16bit) 及 K8 (32 bit) 有效；

编程示例:



SFTL 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
SFTL	位左移	16	否	SFTL (S) (D) (n1) (n2)	7
SFTLP		16	是		7

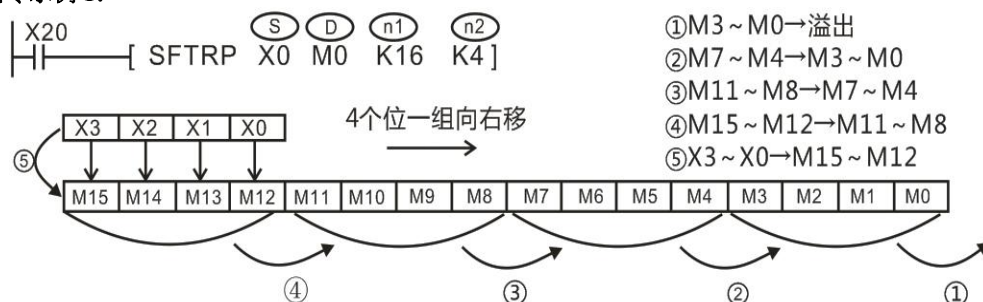
对于位变量, 将(S)地址起始的(n2)位变量与(D)地址起始的(n1)变量, 按向左方向移动(n2)位后, 将结果保存在(D)中。

本指令一般使用脉冲执行型指令。

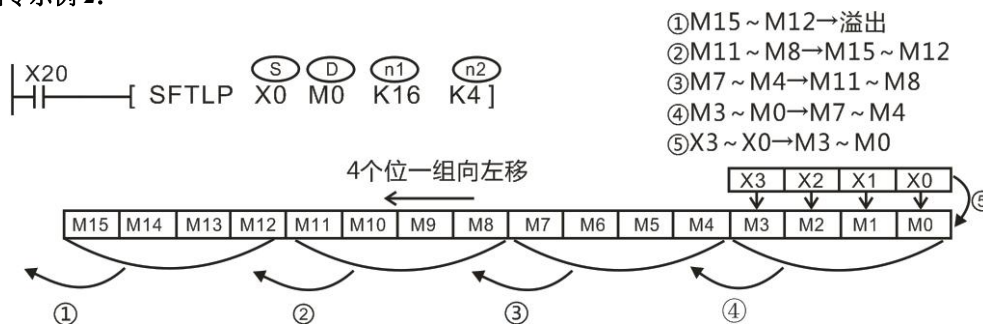
操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)		√	√	√												
(D)	√	√	√	√												
(n1)	常数, n1 ≤ 1024															
(n2)	常数, n2 ≤ n1															

编程示例:

指令示例 1:



指令示例 2:



SFTR 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
SFTR	位右移	16	否	SFTR (S) (D) (n1) (n2)	9
SFTRP		16	是		9

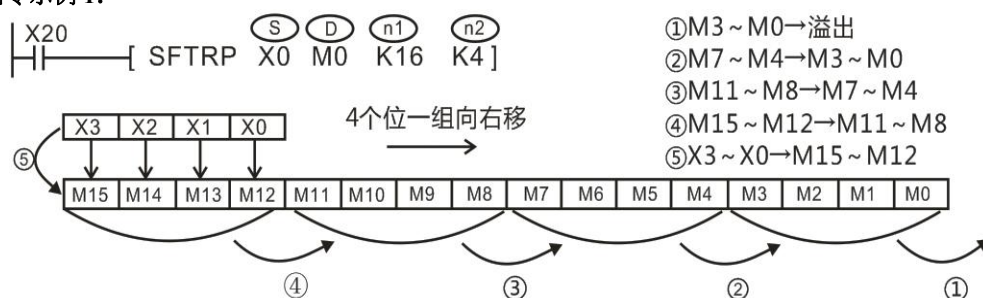
对于位变量, 将(S)地址起始的(n2)位变量与(D)地址起始的(D)变量, 按向右方向移动(n2)位后, 将结果保存在(D)中。

本指令一般使用脉冲执行型指令。

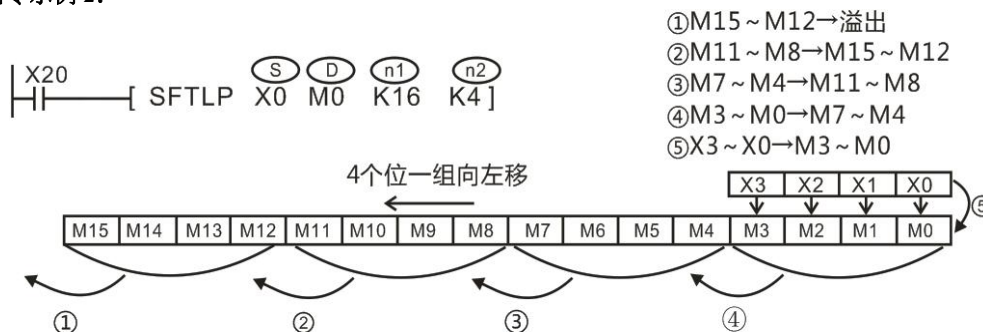
操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)		√	√	√												
(D)	√	√	√	√												
(n1)	常数, n1 ≤ 1024															
(n2)	常数, n2 ≤ n1															

编程示例:

指令示例 1:



指令示例 2:



WSFL 指令

指令描述:

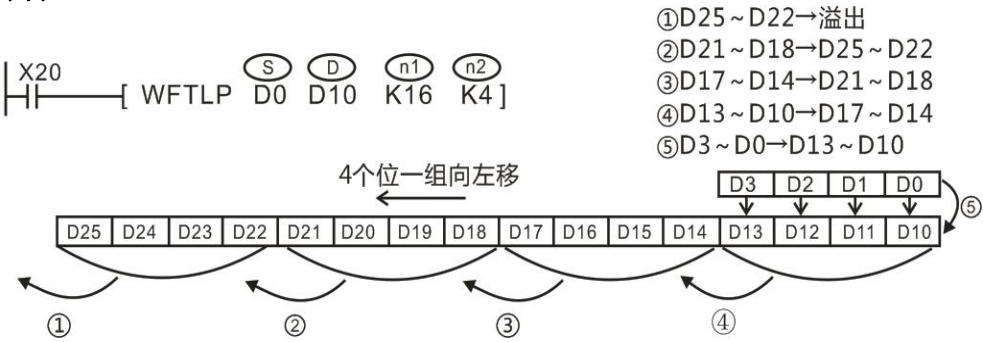
名称	功能	位数(bits)	是否脉冲型	指令格式	步数
WSFL	字左移	16	否	WSFL (S) (D) (n1) (n2)	9
WSFLP		16	是		9

以字为单位，将(S)地址起始的(n2)字变量与(D)地址起始的(n1)字变量，按向左方向移动(n2)个字。

本指令一般使用脉冲执行型指令。

操作数	位元件							字元件								
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)								√	√	√	√	√	√	√		
(D)									√	√	√	√	√	√		
(n1)	常数, n1≤2048															
(n2)	常数, n2≤n1															

编程示例:



WSFR 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
WSFR	字右移	16	否	WSFR (S) (D) (n1) (n2)	9
WSFRP		16	是		9

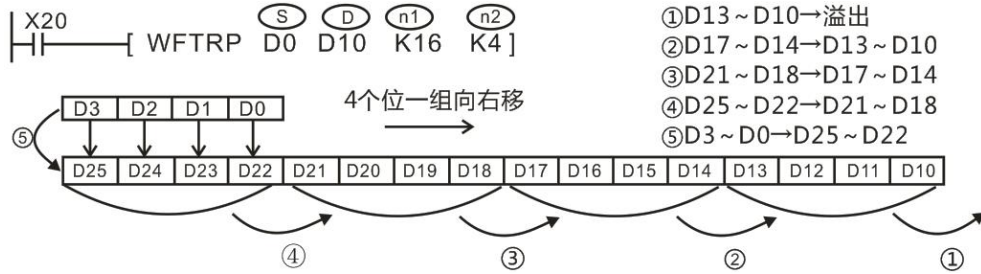
以字为单位, 将(S)地址起始的(n2)字变量与(D)地址起始的(n1)字变量, 按向左方向移动(n2)个字。

本指令一般使用脉冲执行型指令。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)								√	√	√	√	√	√	√		
(D)									√	√	√	√	√	√		
(n1)	常数, n1 ≤ 2048															
(n2)	常数, n2 ≤ n1															

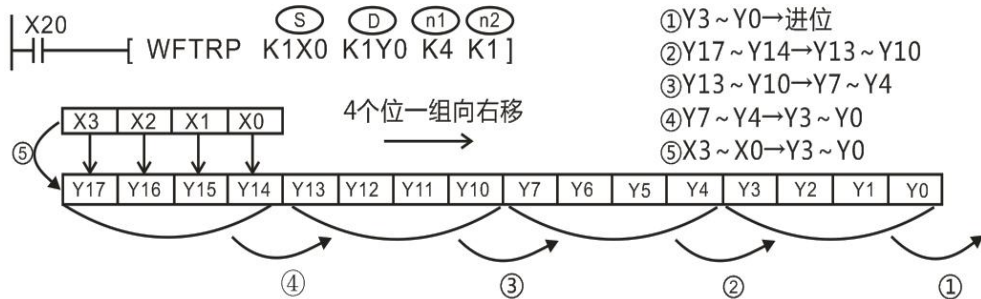
编程示例:

指令示例 1:



指令示例 2:

使用 Kn 类型元件时, 需要指定相同的位数。



SFRD 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
SFRD	移位读出（为控制先入先出的数据读入指令）	16	否	SFRD (S) (D) (n)	7
SFRDP		16	是		7

操作数	位元件				字元件										
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)					√	√	√	√	√	√	√	√	√		
(D)								√	√	√	√	√	√		
(n)	取值范围：16bit 指令：n=1~256；32bit 指令：n=1~128														

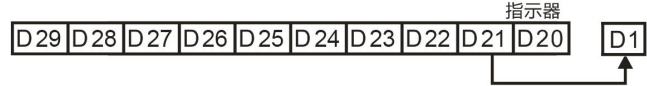
从“先进先出”队列(S)的首项读出到(D)中，然后将队列(S)逐字右移 1 个字，将队列指针递减，以第一个编号装置作为指针，当指令执行时，指针内容值先减 1，之后 S 所指定的装置其内容值会写入先入先出(D)数据串列中由指针所指定的位置。若指针已经为 0，则指令不处理前述操作，而 0 标志 M8020 会置 1。

本指令一般使用脉冲执行型指令。

使用说明:



先入先出的数据读出指令。



X0 由 OFF 到 ON 本指令动作依照下列编号 1~3 动作。（D10 内容保持不变）

- 1: D2 的内容被读出传送至 D20 当中，
- 2: D10~D3 全部往右移位一个寄存器，
- 3: 指针 D1 内容减 1。

SFWR 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
SFWR	移位写入（为控制先进先出的数据写入指令）	16	否	SFWR (S) (D) (n)	7
SWFRP		16	是		7

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)					√	√		√	√	√	√	√	√	√		
(D)									√	√	√	√	√	√		
(n)	常数，2≤n≤2048															

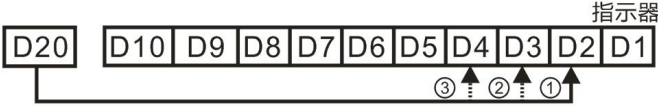
将(S)的值写入由(D)地址起始，个数为(n)的“先进先出”队列中，以第一个编号装置作为指针，当指令执行时，指针内容值先加 1，之后(S)所指定的装置其内容值会写入先入先出(D)数据串行中由指针所指定的位置。

本指令一般使用脉冲执行型指令。

编程示例:



先入先出的数据写入指令。



当 X0=1 时，D0 的内容被存入 D2，D1 的内容变为 1。当 X0 再次从 OFF→ON 时，这个 D0 的内容被存入 D3，D1 的内容变为 2，以此类推。若 D1 的内容超过 n - 1，则指令不处理，而进位标志 M8022 会置 1。

5.2.8 外部设备 IO 指令

TKY 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
TKY	十字键输入	16	否	TKY (S) (D1) (D2)	7
DTKY		32	否		13

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)	√	√	√	√												
(D1)									√	√	√	√	√	√	√	√
(D2)		√	√	√												

该指令是指定 10 个连续的位变量单元（如 X 输入端口），依次代表 10 进制的 0~9 按键，当有按键动作（状态为 ON）时，以按键的动作顺序，可输入十进制 0~9999 的 4 位数值；若采用 32bit 指令，则可输入十进制 0~99, 999, 999 的 8 位数值。其中：

(S)为按键的起始输入端口，使用随后的共 10 个位单元（如 X 端口）；

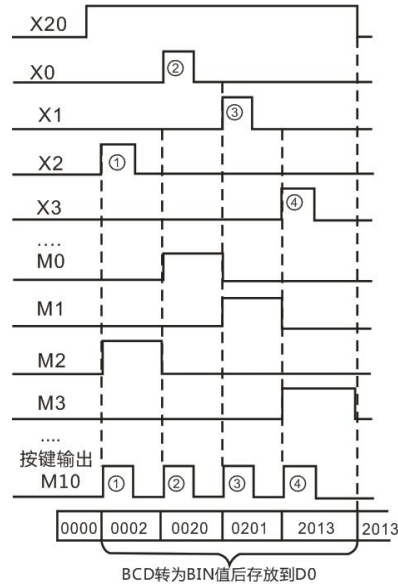
(D1)为已输入数值的存放单元；

(D2)为按键组当前状态的暂存起始单元，占用随后的共 11 个位单元。

编程示例:



若要输入数字“2013”，按顺序按键 2、0、1、3（X2、X0、X1、X3）即可，PLC 内部变量的动作如下图。若采用 32bit 指令（DTKY），(D1)占用 32bit 的变量，对上例即为 D1、D0，分别为高字和低字。



由指令中参数的设置，X0~X11 分别对应 0~9 数字键；M0~M9 对应键的状态；当有任何按键按下时，按键输出单元 M10 会置位。

按键值（如 2013）被转换为 BIN 格式后存入指定的 (D1) 单元 D0；

(D0=0x7DD)，即使驱动的能量流变为 OFF，D0 也不会改变。

当有多个按键按下时，先检测到的按键有效；当输入的数字超过 4 位后，最先输入的数字变化溢出，只留下输入的最后 4 个数字。

注意：TKY 和 DTKY 指令一个程序只可使用其中一个一次。

HKY 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
HKY	16 键输入	16	否	HKY (S) (D1) (D2) (D3)	9
DHKY		32	是		17

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)	√													√		
(D1)		√														
(D2)											√	√	√	√	√	√
(D3)		√	√	√												

该指令是读取 4×4 矩阵型共 16 个按键，依次代表 10 进制的 0~9 按键，以及 A~F 的功能按键，当有按键动作（状态为 ON）时，以按键的动作顺序，可输入十进制 0~9999 的 4 位数值，或 A~F 的功能键；若采用 32bit 指令，则可输入十进制 0~99,999,999 的 8 位数值，或 A~F 的功能键。其中：

(S) 为按键的扫描输入 X 端口的起始端口号，使用随后的共 4 个 X 端口；

(D1) 为按键的扫描输出 Y 端口的起始端口号，使用随后的共 4 个 Y 端口；

(D2) 为按键输入值存放单元，0~9999；当为 32bit 指令时可输入 0~99,999,999 范围内的数值；

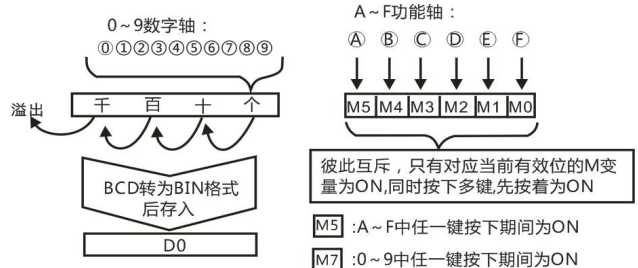
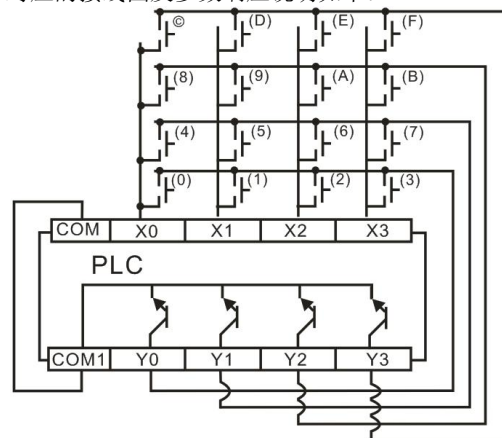
(D3) 为按键输入状态起始单元地址，共占用连续的 8 个位变量单元。

此指令只能用于晶体管输出型 PLC。

编程示例:



对应的接线图及参数响应说明如下:



1. 由于继电器输出延迟大，需要采用 MT 晶体管型控制器；
2. 驱动能流 X20 为 OFF 期间，D0 不变，而 M0~M7 均为 OFF；
3. 按键扫描的操作完成需要 8 个扫描周期，完成后 M8029 会置位 1 个扫描周期。

注:

(1) HKY 和 DHKY 指令一个程序只可使用其中一个一次。

(2) 使用该指令，需保证扫描周期大于等于 X0~7 输入滤波时间。

方法一：采用恒定扫描周期，设置 D8039 的值（单位 ms）大于等于滤波时间（D8020），然后置 ON M8039；

方法二：在该指令前加入 REFF 指令设定 REFF 的参数为小于等于扫描周期的值。

DSW 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DSW	数字开关	16	否	DWS (S) (D1) (D2) (n)	9

该指令是读取矩阵型设置开关的状态，以 4 个 BCD 设定开关为 1 组，将设定值读取后存放到指定单元，最多可读取 2 组设定开关。其中：

(S) 为按键的扫描输入 X 端口的起始端口号，若 (n)=1，使用随后的共 4 个 X 端口；若 (n)=2，则使用随后的共 8 个 X 端口；

(D1) 为按键的扫描输出 Y 端口的起始端口号，使用随后的共 4 个 Y 端口；

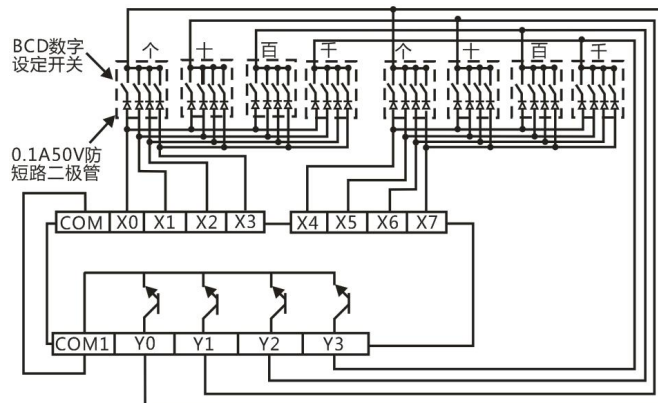
(D2) 为按键输入值存放单元，0~9999；

(n) 为数字设定开关的组数，只能取 1~2。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)	√															
(D1)		√														
(D2)												√	√	√	√	√
(n)	常数，n=1~2															

编程示例:

X20 [DSW (S) (D1) (D2) (n) X0 Y0 D0 K2]



当 X20=ON 时，执行扫描周期读取数字开关设定的操作：

1. 第一组数字开关的设定值，经 BIN 转换后存入 D0；
2. 第二组数字开关的设定值，经 BIN 转换后存入 D1；
3. 一次读取循环完毕，M8029 会置位一个扫描周期。

指令使用说明:

X20 [SET M1]
M1 [DSW X0 Y0 D0 K2]
M8029 [RST M1]

需要使用晶体管输出型 PLC 才能正常检测数字开关；

一个数字开关设定的读取操作需要多个扫描周期才能完成，若采用按键启动读取操作，建议采用如下程序语句，保证可读取周期的完整：

注：使用该指令，需保证扫描周期大于等于 X0~7 输入滤波时间。

方法一：采用恒定扫描周期，设置 D8039 的值（单位 ms）大于等于滤波时间（D8020），然后置 ON M8039；

方法二：在该指令前加入 REFF 指令设定 REFF 的参数为小于等于扫描周期的值。

SEGD 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
SEGD	七段码译码	16	否	SEGD (S) (D)	5
SEGDP		16	是		5

操作数	位元件							字元件								
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)					√	√		√	√	√	√	√	√	√	√	√
(D)									√	√	√	√	√	√	√	√

该指令是数据源的低 4 位，翻译成 7 段显示码，存放到目的变量的低 8 位中。其中：

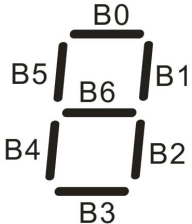
(S)为待译码的数据源（取 BIN 内容的最低四个位 b0~b3）；

(D)为译码后存放 7 段码的变量。

编程示例:

X20
[SEGDP (S) (D) D0 K2Y10]

操作时，当 X20 为 ON 时，将 D0 内数据低 4 位译码后，输出到 Y10~Y17 端口。翻译用的对应表如下表。该表格不需用户准备，PLC 系统内部已具备该对照表。

数据		数码管组合	内部译码表值								译码后字符
HEX 数	BIN 数		B7	B6	B5	B4	B3	B2	B1	B0	
0	0000	 每位对应一个笔段， 1=笔段点亮 0=笔段熄灭	0	0	1	1	1	1	1	1	0
1	0001		0	0	0	0	0	1	1	0	1
2	0010		0	1	0	1	1	0	1	1	2
3	0011		0	1	0	0	1	1	1	1	3
4	0100		0	1	1	0	0	1	1	0	4
5	0101		0	1	1	0	1	1	0	1	5
6	0110		0	1	1	1	1	1	0	1	6
7	0111		0	0	0	0	0	1	1	1	7
8	1000		0	1	1	1	1	1	1	1	8
9	1001		0	1	1	0	1	1	1	1	9
A	1010		0	1	1	1	0	1	1	1	A
B	1011		0	1	1	1	1	1	0	0	b
C	1100		0	0	1	1	1	0	0	1	c
D	1101		0	1	0	1	1	1	1	0	d
E	1110		0	1	1	1	1	0	0	1	e
F	1111		0	1	1	1	0	0	0	1	f

SEGL 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
SEGL	七段码时分显示	16	否	SEGL (S) (D) (n)	7

操作数	位元件							字元件								
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)					√	√		√	√	√	√	√	√	√	√	√
(D)		√														
(n)	常数, n=0~7															

该指令是利用 8 个或 12 个 Y 端口用于 4 位或 8 位七段锁存数码管的显示驱动, 显示方式为扫描驱动方式。其中:

(S) 为待显示的数据, 其值在 BCD 转换后才送到数码管进行显示;

(D) 为显示驱动用的 Y 端口起始地址号;

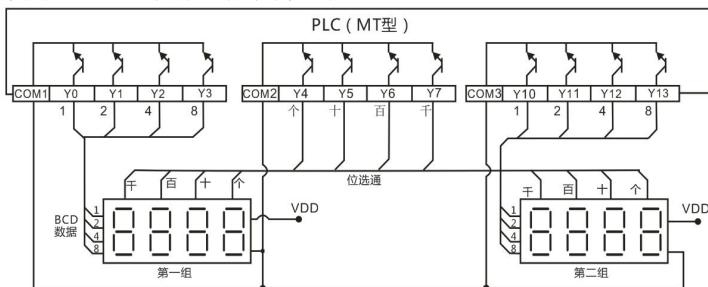
(n) 为根据显示数据的组数、信号正负逻辑等相关的设定值, 见下面的详细描述。

编程示例:

```

X10
|---[ SEGL (S) (D) (n) ]
      D0 Y0 K6
  
```

对应的硬件接线如下图, 在第一组数码管上显示 D0 的内容, 在第二组数码管上显示 D1 的内容, 若 D0 或 D1 的读数超过 9999, 程序运行就会出错:



接线图所使用的数码管, 自带显示数据锁存、7 段译码和驱动、负逻辑型 (输入端口为低电平时, 表示输入的数据为 1, 或被选通) 的 7 段数码显示管。显示处理中, PLC 的 Y4~Y7 端口会自动循环扫描, 每次只有 1 个端口为 ON, 作为位选通信号, 此时 Y0~Y3 口上的数据即为送给对应位的 BCD 码数据, 当位选通信号由 ON→OFF 时, 即被锁存到数码管内的锁存器中, 经过内部的译码和驱动后, 数码管将数字显示出来。PLC 系统依次将 Y4~Y7 循环作同样处理, 直到将 4 位都处理完毕。同样的道理, Y10~Y13 是第二组 4 位数码管的数据输出端口, 共用 Y4~Y7 的位选通线, 处理方法相同, 两组的显示处理是同时进行的。范例中若 D0=K2468, D1=K9753, 则第一组将会显示 2468, 第二组显示 9753。

完成一次显示刷新需要 12 个扫描周期, 处理完成后, M8029 标志置为 ON 一个扫描周期。(n) 的选择: 根据可编程控制器的正负逻辑、七段码的正负逻辑等因素, 按以下原则选择:

一组 4 位数的时候 n=0~3。二组 4 位数的时候 n=4~7。

显示组数	1 组				2 组			
Y 数据输出极性	PNP		NPN		PNP		PNP	
选通与数据极性	相同	相反	相同	相反	相同	相反	相同	相反
(n) 的取值	0	1	2	3	4	5	6	7

PLC 的晶体管输出极性性与 7 段显示器的输入极性是否相同或者是不同时, 可透过参数 n 的设置值来相互匹配, 该指令在程序中最多只能使用 2 次。

使用说明:

由于继电器不适合较高频率的扫描输出动作, 晶体管输出型 PLC 才能使用该指令。

该指令与可编程控制器的扫描周期 (运算周期) 同时执行。为执行一系列的显示, 可编程控制器的扫描周期需要 10ms 以上; 不足 10ms 时, 使用恒定扫描模式, 请用 10ms 以上的扫描周期定时运行。

本可编程控制器的晶体管输出的 ON 电压约为 1.5V, 7 段码请使用与此相应的输出电压。

ARWS 指令

指令说明

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
ARWS	方向开关	16	否	ARWS (S) (D1) (D2) (n)	7

该指令可指定 X 作为编辑按键、Y 端口驱动 4 位 7 段数码管，用作寄存器的参数编辑的简易界面。其中：

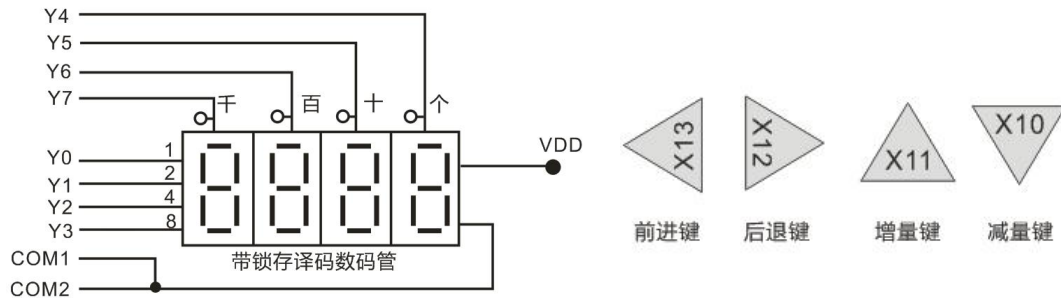
- (S) 为指定按键输入的起始地址，占用后续的共 4 个位单元；
- (D1) 用于被显示及修改的变量，只能显示一个 16bit 宽度的变量；
- (D2) 用作数码管显示驱动的 Y 端口起始地址，占用后续的共 8 个 Y 端口；
- (n) 为信号逻辑的设定值，参见前面 SEGL 指令中关于 (n) 的详细描述。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)	√	√	√	√												
(D1)												√	√	√	√	√
(D2)		√														
(n)	常数，n=0~3															

编程示例：



对应的硬件接线如下图所示，PLC 应为晶体管输出型：



操作方法：

- 图中数码管显示 D0 的数值，按下 X10~X13 可修改其中的数值，D0 的数值只能在 0~9999 之间。
- 当 X20 为 ON 时，光标位为千位。每次按后退键 (X12) 时，指定位按 “千→百→十→个→千” 的顺序切换；若按前进键 (X13)，则切换顺序相反；光标位由选通脉冲信号 (Y004 ~ Y007) 连接的 LED 指示。
- 对于光标位，每次按增量键 (X11) 该位的内容按 0→1→2→……8→9→0→1 变化，按减量键 (X10) 时，则按 0→9→8→7→……1→0→9 变化，修改的值立即生效。

指令使用说明：

当用户程序扫描时间短时，请使用恒定扫描模式，或在定时中断内按固定时间间隔运行。

ASC 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
ASC	ASCII 码转换	16	否	ASC (S) (D)	11

- (S)为欲执行 ASCII 码变换的英文字母，由计算机输入，最大允许长度为 8 个字符。
- (D)为存放 ASCII 码的起始单元号，占用随后的共 4 个（M8161=0）或 8 个变量单元（M8161=1）。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)	在指令输入时，由常数变量输入，允许长度为 8 个字符。															
(D)												√	√	√		

编程示例:



	高字节	低字节
D200	54(t)	53(S)
D201	50(p)	4F(o)
D202	45(e)	50(p)
D203	20	53(d)

图 1

	高字节	低字节
D200	00	53(S)
D201	00	54(t)
D202	00	4F(o)
D203	00	50(p)
D204	00	50(p)
D205	00	45(e)
D206	00	44(d)
D207	00	20

图 2

当 X20=ON 时，D200~D203 的赋值如图 1 所示；
若将特殊寄存器 M8161 置为 ON，则每个 ASCII 字符在转换后占用 1 个 16bit 变量，如图 2，每个变量的高字节填 0 处理：。

附：《 ASCII 代码对照表 》

10 进制位	ASCII(16 进制数)	英语字母	ASCII(16 进制数)	英语字母	ASCII(16 进制数)
0	30	A	41	N	4E
1	31	B	42	O	4F
2	32	C	43	P	50
3	33	D	44	Q	51
4	34	E	45	R	52
5	35	F	46	S	53
6	36	G	47	T	54
7	37	H	48	U	55
8	38	I	49	V	56
9	39	J	4A	W	57
		K	4B	X	58
		L	4C	Y	59
		M	4D	Z	5A

代码	ASCII (16 进制数)
STX	02
ETX	03

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
PR	ASCII 码打印	16	否	PR (S) (D)	5

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)												√	√	√		
(D)		√														

该指令将指定变量单元的数值，通过 Y 输出端口以同步方式逐个字节对外输出。其中：

(S)为待输出的变量单元起始地址；

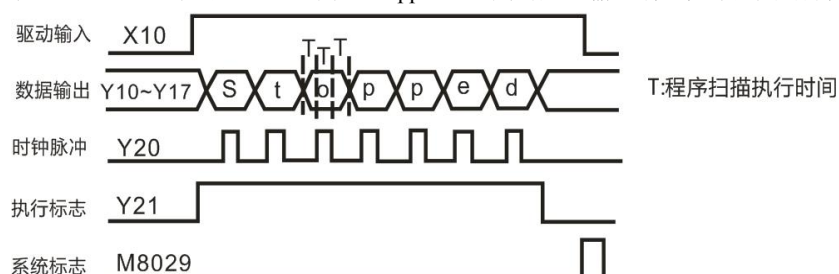
(D)为进行输出打印的 Y 端口起始号。

编程示例:

```

M8000
┌───┴───┐
└───┬───┘ [ PR (S) (D) ]
      D200 Y10
  
```

若 D200~D203 中的 ASCII 码为 “Stopped”，则对应的输出端口信号及其时序如下图：



使用说明:

必须使用晶体管输出型 PLC，才能完成该指令功能；

打印过程中，驱动信号 X10 变为 OFF 时，打印输出即被中断。X10 再次为 ON 时，打印动作重新开始；

M8027=Off 时，最多可执行 8 个字的串行的输出；当 M8027=On 时，则可执行 1~16 个字的串行输出。

打印输出过程中，遇到 “00” 的字符时，会自动结束打印操作，之后的文字不再处理；而 M8029 完成标志只有在驱动能流信号无效后才会置 ON 一个扫描周期；

指令按扫描周期（图中 T）执行，若扫描周期短时，请用恒定扫描模式；若过长则可以在定时中断程序中执行。

FROM 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
FROM	BFM 的读出	16	否	FROM (m1) (m2) (D) (n)	9
FROMP		16	是		9
DFROM		32	否		17
DFROMP		32	是		17

该指令用于读取特殊扩展模块的 BFM 区寄存器的数据读取操作。其中：

(m1)为特殊扩展模块的地址编号，最靠近主模块的为 0，依次编号（#0→#1→#2→#3。。。。。。），允许超过 8 块特殊扩展模块；

(m2)为特殊模块内 BFM 的寄存器地址号，取值范围 0~32767；

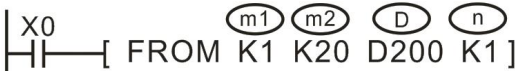
(D)为读取参数在主模块中的存放地址，当读取的寄存器数多于 1 个时，占用随后的单元；

(n)本次读取参数的个数（按 Word 计算），按寄存器地址依次读取。16bit 指令时，(n) =1-514；32bit 指令时，(n) =1-257；

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(D)									√	√	√	√	√	√	√	√

m1、 m2=0~32767；
 16bit 指令时，(n) =1-514； 32bit 指令时，(n) =1-257；
 (D)元件指定时，16bit 指令可用 K1~K4； 32bit 指令可用 K1~K8 ； m1, m2, n 不支持字符装置 D 寄存器 。

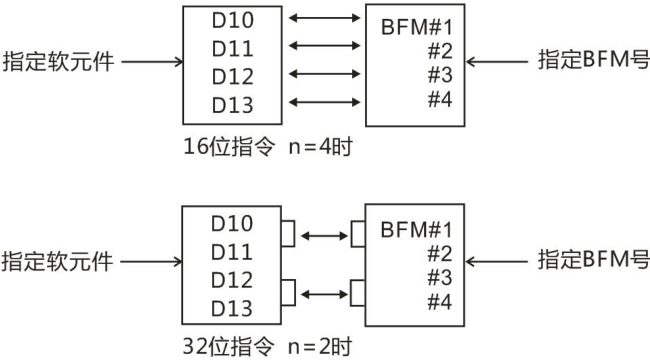
编程示例:



表示当 X0 为 ON 时，将#1 号（第 2 个）特殊模块中的缓冲寄存器（BFM）#20（16bit 宽度）的内容读出到 PLC 的 D200 寄存器中，一次读取一笔（n=1）。当 X0 为 OFF 时，不执行传送，传送地点的数据不变化，脉冲指令执行后也同样。

当用 32bit 指令时，(D)指定的地址为低 16bit 地址，(D)+1 为高 16bit 地址。

(n)表示操作的参数个数，16bit 指令中，n=2 表示 2 Word；而 32bit 指令中，n=1 表示 2 Word，即 16 位指令的 n=2 与 32 位指令的 n=1 意义相同，请注意。例如：



TO 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
TO	BFM 的写入	16	否	TO (m1) (m2) (D) (n)	9
TOP		16	是		9
DTO		32	否		17
DTOP		32	是		17

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(D)									√	√	√	√	√	√	√	√

m1; m2=0~32767; 16bit 指令时, (n)=1-514; 32bit 指令时, (n)=1-257;
(D)元件指定时, 16bit 指令可用 K1~K4; 32bit 指令可用 K1~K8; m1, m2, n 不支持字符装置 D 寄存器

该指令用于向特殊扩展模块的 BFM 区寄存器的做数据写入操作。其中:

(m1)为特殊扩展模块的地址编号, 最靠近主模块的为 0, 依次编号, 允许超过 8 块特殊扩展模块;

(m2)为特殊模块内 BFM 的寄存器地址号, 取值范围 0~32767;

(D)为主模块中参数寄存器地址, 其参数作为写操作数据的来源。当写操作的寄存器数多于 1 个时, 占用随后的单元;

(n)本次写入参数的笔数, 为按寄存器地址依次写入。16bit 指令时, (n)=1-514; 32bit 指令时, (n)=1-257;

编程示例:

指令示例 1:



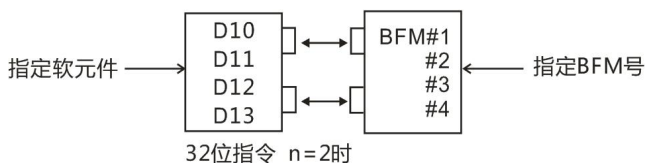
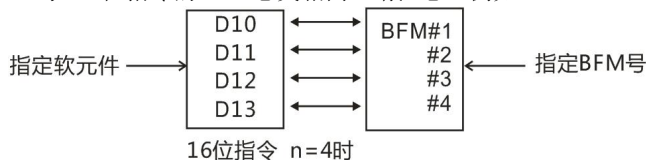
表示当 X1 为 ON 时, 将 PLC 的 D220 寄存器中的数据, 写入到 #1 号特殊模块中的第 24 号地址中, 一次只写入一笔。当 X1 为 OFF 时, 不执行操作。

当用 32bit 指令时, 指定的地址(D)为低 16bit 地址, (D)+1 为高 16bit 地址。例如:

高16位 低16位

BFM#6 BFM#5 ←指定BFM号

(n)表示操作的参数个数, 16bit 指令中, n=2 表示 2 Word; 而 32bit 指令中, n=1 表示 2 Word, 即 16 位指令的 n=2 与 32 位指令的 n=1 意义相同, 请注意。例如:



关于 FROM/TO 指令的使用说明:

1. 用 FROM/TO 指令访问扩展模块是比较耗时的操作, 执行多个 FROM / TO 指令或传送多个缓冲存储器数据时, PLC 的扫描周期会延长。为了防止运行超时, 可在 FROM / TO 前加入延长监视定时器时间的 WDT 指令, 或者错开 FROM / TO 指令的执行时间, 或者用脉冲执行型指令。

2. 关于特殊模块的连接方法和输入输出编号等使用方法, 请参阅各特殊模块附带的专用手册。

GRY 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
GRY	格雷码转换	16	否	GRY (S) (D)	5
GRYP		16	是		5
DGRY		32	否		9
DGRYP		32	是		9

该指令是将 BIN 数值，转换为格雷码（GRY）。其中：

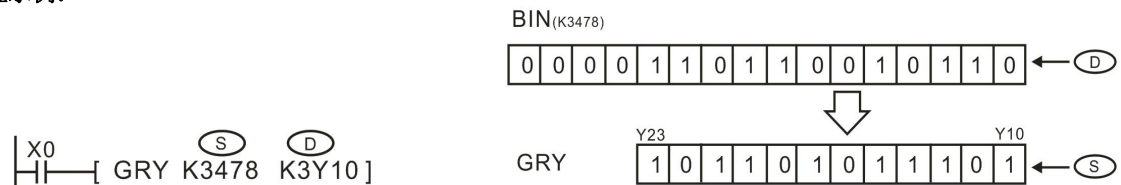
(S) 为待转换的 BIN 的数据源或数据变量单元；16bit 指令时，0~32767 范围有效，32bit 指令时，0~2147483647 范围有效。

(D) 为转换为格雷码（GRY）后的存放单元。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)					√	√		√	√	√	√	√	√	√	√	√
(D)									√	√	√	√	√	√	√	√

BIN→GRY 数学算法：从最右边一位起，依次将每一位与左边一位异或(XOR)，作为对应 GRY 该位的值，最左边一位不变(相当于左边是 0)；

编程示例:



GBIN 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
GBIN	格雷码的逆转换	16	否	GBIN (S) (D)	5
GBINP		16	是		5
DGBIN		32	否		9
DGBINP		32	是		9

该指令是将 GRY 格雷码，转换为二进制数值。其中：

(S) 为待转换的 GRY 码数据源或数据变量单元；16bit 指令时，0~32767 范围有效，32bit 指令时，0~2147483647 范围有效。

(D) 为转换为 BIN 后的存放单元。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)					√	√		√	√	√	√	√	√	√	√	√
(D)									√	√	√	√	√	√	√	√

GRY→BIN 数学算法：从左边第二位起，将每位与左边一位解码后的值异或，作为该位解码后的值（最左边一位依然不变）。

编程示例:



5.2.9 电子凸轮指令

DECAM 指令

指令描述：（详细说明书请查看“电子凸轮使用说明手册”）

仅 LX3VE 系列 PLC 支持该指令。

名称	功能	位数 (bits)	是否脉冲型	指令格式	步数
DECAM	电子凸轮配置	32	否	DECMP (S1)(S2)(S3)(D1)(D2)	21

(S1)- 主轴输入，适用软元件 C，K；

(S2)- 凸轮参数地址，适用软元件 D；

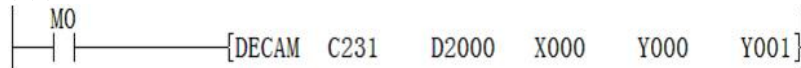
(S3)- 外部启动信号，适用软元件 X，M；

(D1)- 从轴脉冲输出，适用软元件 Y0~Y4；

(D2)- 从轴方向输出，适用软元件 Y。

操作数	位元件					字元件										
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√									√		
(S2)														√		
(S3)	√		√													
(D1)		√														
(D2)		√														

编程示例：



就(S2)起始的各单元参数值的功能和设定方法说明如下表：

偏移	说明	说明	初始值	范围
0	表格版本号			
1	标志寄存器	Bit0 - 标志位，初始化完成 电子凸轮允许信号启动电子凸轮后，计算电子凸轮相关数据，初始化完毕后自动设定 ON，使用者需自行清除此标志状态 Bit1 - 标志位，周期完成 电子凸轮完成标志。当周期式电子凸轮执行完毕后，此标志会自动设为 ON；使用者需先清除此标志状态。 Bit2 - 标志位，脉冲发送延迟了 Bit3 - 标志位，出错电子凸轮停止运行 Bit4 - 标志位，参数错误出错，电子凸轮停止运行 Bit5 - 标志位，表格出错，电子凸轮停止运行 Bit6 - 标志位，周期性电子凸轮 Bit7 - 标志位，非周期性电子凸轮 Bit9 - 标志位，当前周期完成停止 Bit10 - 标志位，同步区 Bit11 - 标志位，时间轴	0	-
2	错误寄存器	参数错误情况：显示错误码 表格错误情况：显示错误的表格段号	0	-
3	功能寄存器 (电子凸轮使用能前确定)	Bit0 - 延迟启动使能 Bit1 - 指定位置启动 Bit2 - 主轴绽放 Bit3 - 从轴缩放 Bit5 - 使用外部启动信号 Bit6 - 从当前位置启动	0	-

4	功能寄存器 (电子凸轮运行中可更改)	Bit1 - 同步信号使用能 Bit2 - 当前周期完成后停止电子凸轮	0	-
5	电子凸轮启动寄存器	0: 立即停止电子凸轮 1: 周期性电子凸轮 (启动) 2: 非周期性电子凸轮 (启动) 其他: 立即停止电子凸轮	0	-
6	电子凸轮最高输出频率设定	电子凸轮最高输出频率设定; 当小于 0 时或者大于 200K 的频率时, 则为 200K	200000	0~200000
7	电子凸轮最高输出频率设定			
8	同步信号 Y 端子号	输出端子编号: 设定同步输出端子的 Y 编号, 范围 0 ~ 255, 当启用同步输出的功能, 当在同步区时, 相应的 Y 端子输出同步信号	0	0 ~ 65535
9	CAM 同步位置下限 (LOW WORD)	电子凸轮的同步位置上/下限设定, 当同步位置下限 $\leq$ 主轴位置 $\leq$ 位置上限 且同步信号使用能 (地址 4, BIT0) 时同步信号端子 Y 输出 ON	0	32 位整数
10	CAM 同步位置下限 (HIGH WORD)		0	32 位整数
11	CAM 同步位置上限 (LOW WORD)			
12	CAM 同步位置上限 (HIGH WORD)			
13	电子凸轮脉冲余数分配设定(保留)	(保留)	(保留)	(保留)
14	非周期式电子凸轮重复次数设定	周期性电子凸轮时----保留 非周期性电子凸轮: 控制重复执行表格的周期次数, 当值为 H0000 时, 不重复执行 (总的次数为 1) 当值为 H0001 时, 将重复执行一次 (总的次数为 2), 以此类推。 当值为 HFFFF 时, 将变成周期执行电子凸轮	0	0 ~ 65535
15	电子凸轮启动延迟脉冲设定 (LOW WORD)	周期性电子凸轮----保留 非周期电子凸轮: 可通过 (地址 3, Bit0-延迟启动使能) 来启用延迟启动功能。 执行非周期电子凸轮时, 接受一个凸轮启动信号, 若没有要立即执行电子凸轮表格, 而要延迟主轴转动几个脉冲, 才运行表格, 此时此寄存设定延迟脉冲数。 如右图所示: 当系统接受一个凸轮启动信号, 主轴转动了设定的脉冲数, 才会立即执行 电子凸轮表格	0	32 位无符号整数
16	电子凸轮启动延迟脉冲设定 (HIGH WORD)			
17	主轴指定位置启动 (LOW WORD)	周期性电子凸轮----保留 非周期性电子凸轮: 可通过 (地址 3, Bit1-指定位置启动使能), 来启用指定位置启动的功能。 启动位置由该地址设置	0	32 位无符号整数
18	主轴指定位置启动 (HIGH WORD)			
19	从轴当前位置(LOW WORD)	输出轴: 从轴当前位置(转换后——缩放后), 当前凸轮执行过程中, 从轴的位置,	0	32 位整数
20	从轴当前位置(HIGH WORD)			
21	从轴当前位置(LOW WORD)	输出轴: 从轴当前位置(转换前——缩放前), 当前凸轮执行过程中, 从轴的位置, =	0	32 位整数
22	从轴当前位置(LOW WORD)			
23	从轴倍率分母	从轴缩放	1	1~65535
24	从轴倍率分子		1	1~65535
25	主轴当前位置(LOW WORD)	输入轴: 主轴当前位置(转换后——缩放后), 当前凸轮执行过程中, 主轴的位置	0	32 位整数
26	主轴当前位置(HIGH WORD)			
27	主轴当前位置(LOW WORD)	输入轴: 主轴当前位置(转换前——缩放前), 当前凸轮执行过程中, 主轴的位置, =	0	32 位整数
28	主轴当前位置(HIGH WORD)			
29	主轴倍率分母	主轴缩放	1	1~65535
30	主轴倍率分子		1	1~65535
31	保留	保留	-	-
32	保留	保留	-	-

33	保留	保留	-	-
34	保留	保留	-	-
35	保留	保留	-	-
36	保留	保留	-	-
37	保留	保留	-	-
38	表格的段数	凸轮表格数据的总数据段	0~512	
39	表格的起始偏移	指定凸轮表格的偏移地址，固定为 40	40	40
40	第 0 段主轴(LOW WORD)	第 0 段主轴位置	0	32 位无符号整数
41	第 0 段主轴(HIGH WORD)			
42	第 0 段从轴(LOW WORD)	第 0 段从轴位置	0	32 位整数
43	第 0 段从轴(HIGH WORD)			
44	第 1 段主轴(LOW WORD)	第 1 段主轴位置		32 位无符号整数
45	第 1 段主轴(HIGH WORD)			
46	第 1 段从轴(LOW WORD)	第 1 段从轴位置		32 位整数
47	第 1 段从轴(HIGH WORD)			
40+N*2	第 N 段主轴(LOW WORD)	第 N 段主轴位置		32 位无符号整数
40+N*2+1	第 N 段主轴(HIGH WORD)			
40+N*2+2	第 N 段从轴(LOW WORD)	第 N 段从轴位置		32 位整数
40+N*2+3	第 N 段从轴(HIGH WORD)			

错误代码：

6781：参数错误

6782：表格超出范围

6783：凸轮超出个数

当产生错误时，此时凸轮功能未启用。

影响标志：

D8141（高字节），D8140（低字节）：Y000 输出脉冲数。反转时减少。（使用 32 位）

D8143（高字节），D8142（低字节）：Y001 输出脉冲数。反转时减少。（使用 32 位）

D8151（高字节），D8150（低字节）：Y002 输出脉冲数。反转时减少。（使用 32 位）

D8153（高字节），D8152（低字节）：Y003 输出脉冲数。反转时减少。（使用 32 位）

M8145: Y000 脉冲输出停止（立即停止）

M8146: Y001 脉冲输出停止（立即停止）

M8152: Y002 脉冲输出停止（立即停止）

M8153: Y003 脉冲输出停止（立即停止）

M8147：Y000 脉冲输出中监控（BUSY/READY）

M8148：Y001 脉冲输出中监控（BUSY/READY）

M8149：Y002 脉冲输出中监控（BUSY/READY）

M8150：Y003 脉冲输出中监控（BUSY/READY）

DEGEAR 指令

指令描述:

名称	功能	位数 (bits)	是否脉冲型	指令格式	步数
DEGEAR	电子齿轮/ 手摇轮指令	16	否	DECMP (S1)(S2)(S3)(D1)(D2)	13
		32	否		

(S1): 指定接手摇轮之来源高速计数器, 允许寄存器 C, D。

当使用高速计数器时, PLC 内部进行外部输入的获取, 修改计数器的值不影响对输入脉冲的判断。

当使用普通计数器或者 D 时, PLC 从寄存器中获取数据做为主轴, 修改其值会影响对主轴脉冲的判断。

(S2): 指定手摇轮指令的数据缓冲区, 允许寄存器 D

(S3): 允许软件 K, D, H

其值表示响应速度, 范围 0~500。

当值为 0-1 时, 表示 1ms。

(D1): 指定脉冲输出轴, 允许软件 K, D, H Y0~Y4

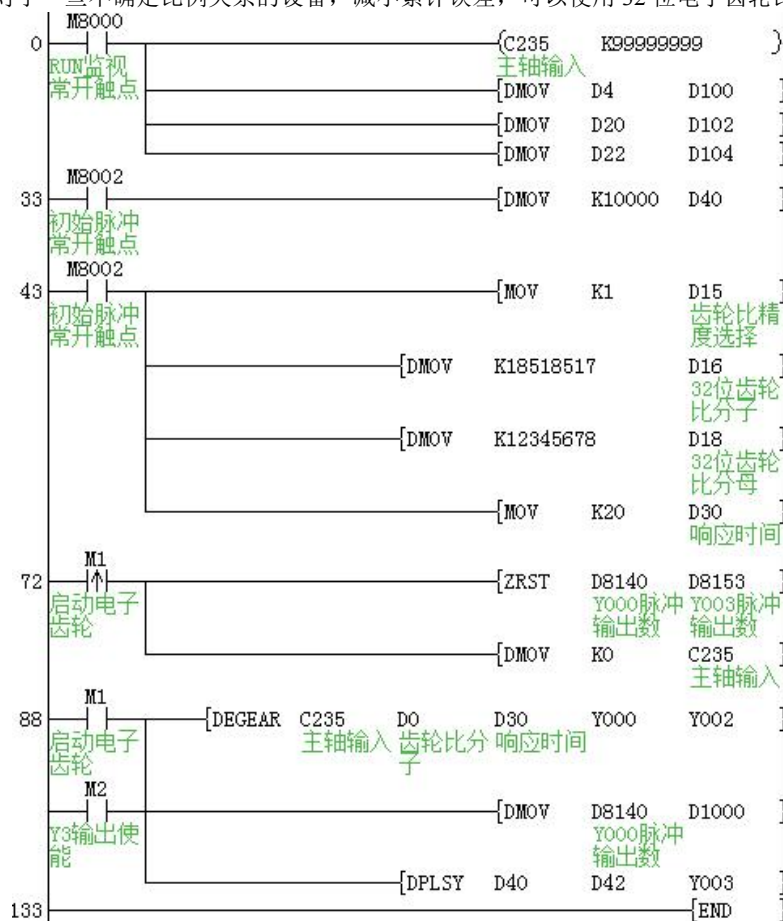
(D2): 指定脉冲输出方向, 允许软件 Yxxx。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)													✓	✓		
(S2)														✓		
(S3)					✓	✓								✓		
(D1)		✓														
(D2)		✓														

当该指令接通时, PLC 在按设定的响应时间设定外部输入的脉冲个数, 达到间隔时间, 算出间隔时间内的平均频率, 根据设定的齿轮比, 计算从轴输出频率以及脉冲个数, 并进行脉冲输出。当从轴的频率大于设定的最高输出频率时, 依设定的最高输出频率进行输出。

电子齿轮指令参数说明表				
偏移	内容	说明	范围	读写权限
0	电子齿轮比 (分子)	输出个数 = 间隔时间内输入数 * 分子 / 分母	0-32767	读/写
1	电子齿轮比 (分母)		1-32767	
2	最高输出频率 (高字)	从轴最高输出频率	1-200000	读/写
3	最高输出频率 (低字)	从轴最高输出频率		读/写
4	主轴平均频率(高字)	手摇动轮输入频率	-	只读
5	主轴平均频率(低字)	手摇动轮输入频率		只读
6	累计手摇轮输入脉冲个数 (高字)	累计手摇轮输入脉冲个数	-	只读
7	累计手摇轮输入脉冲个数 (低字)			
8	标志	保留	保留	保留
9	间隔	确认值	-	只读
10	电子齿轮比 (分子)	确认值	-	只读
11	电子齿轮比 (分母)	确认值	-	只读
12	最高输出频率 (高字)	确认值	1-200000	只读
13	最高输出频率 (低字)			只读
14	动态切换齿轮比	1: 立即切换新设置的齿轮比。并把该地址设回 0。 2: 周期完成切换齿轮比, 切换完成把该值设回 0。 (主轴计数到达分母的值做为一个周期)。	0-2	读/写
15	16 位齿轮比与 32 位齿轮比切换	0: 使用 16 位齿轮比; 1: 使用 32 位齿轮比 注: 更改此位后, 只有重新开启 DEGEAR 指令或使用动态切换齿轮比功能后生效。	0-1	读/写
16	32 位电子齿轮比分子(低字)	输出个数=间隔时间内主轴输入数*分子/分母	0-214748647	读/写
17	32 位电子齿轮比分子(高字)			
18	32 位电子齿轮比分母(低字)		1-214748647	读/写
19	32 位电子齿轮比分母(高字)			
20	32 位电子齿轮比分子(低字)	确认值	-	只读
21	32 位电子齿轮比分子(高字)			
22	32 位电子齿轮比分母(低字)		-	只读
23	32 位电子齿轮比分母(高字)			

(3) 对于一些不确定比例关系的设备，减小累计误差，可以使用 32 位电子齿轮比分子和分母。



注意：

1. 当使用高速计数器时，PLC 内部进行外部输入的获取，修改计数器的值不影响对输入脉冲的判断。
2. 当使用普通计数器或者 D 时，PLC 从寄存器中获取数据做为主轴，修改其值会影响对主轴脉冲的判断。

特殊地址：

1. D8141（高字节），D8140（低字节）：Y000 输出脉冲数。反转时减少。（使用 32 位）
2. D8143（高字节），D8142（低字节）：Y001 输出脉冲数。反转时减少。（使用 32 位）
3. D8151（高字节），D8150（低字节）：Y002 输出脉冲数。反转时减少。（使用 32 位）
4. D8153（高字节），D8152（低字节）：Y003 输出脉冲数。反转时减少。（使用 32 位）
5. M8145：Y000 脉冲输出停止（立即停止）
6. M8146：Y001 脉冲输出停止（立即停止）
7. M8152：Y002 脉冲输出停止（立即停止）
8. M8153：Y003 脉冲输出停止（立即停止）
9. M8147：Y000 脉冲输出中监控（BUSY/READY）
10. M8148：Y001 脉冲输出中监控（BUSY/READY）
11. M8149：Y002 脉冲输出中监控（BUSY/READY）
12. M8150：Y003 脉冲输出中监控（BUSY/READY）

ECAMTBX 指令

指令描述:

名称	功能	位数 (bits)	是否脉冲型	指令格式	步数
ECAMTBX	电子凸轮曲线生成指令	16	否	ECAMTBX (S1) (S2) (D1) (D2)	9

(S1)--参数地址，允许软元件：D；

说明：生成曲线的时设定的参数，（详细请查看“电子凸轮使用说明手册”中‘三种曲线参数表格’说明）；

(S2)--曲线类型，允许软元件：D、H、K；

说明：指示要生成的曲线类型；

K0-1: 以 1ms 的主轴，生成 S 型加减速曲线；

K2: 自定义指定关键点生成表格；

K100: 生成飞剪曲线；

K101: 生成追剪曲线；

(D1)--电子凸轮参数的首地址；

说明：生成的表格数据 存放于【(D1) + 40】开始的位置，【(D1) + 38】中存放表格段数；

(D2)--表格生成结果；

D2<0 生成表格出错；

D2>0 生成表格成功,D2 表示当前表格的总段数；

ECAMTBX 指令生成曲线错误码：

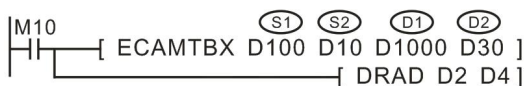
- 1: 条件参数错误
- 2: 主轴脉冲数过少，不够同步区
- 3: 未知的凸轮曲线类型
- 4: 解析度范围错误
- 5: 计算的从轴脉冲数太多
- 6: 计算的从轴脉冲数太少
- 7: 计算的主轴脉冲数超出设定长度
- 8: 设置的从轴脉冲数为 0
- 9: 设置的从轴脉冲数为负数
- 10: S 型加减速曲线计算错误
- 11: 未知的曲线类型

关键点生成曲线错误码：

- 21: 关键点个数不在范围
- 22: 总解析度超过范围
- 23: 主轴大小关系不正确
- 24: 每一段解析度设置不正确
- 25: 计算时，关键点个数不足
- 26: 未知加速曲线

操作数	位元件								字元件							
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)														√		
(S2)					√	√								√		
(D1)														√		
(D2)														√		

编程示例:



曲线类型参数地址为 D100 ；

曲线类型地址为 D10 的数值；

电子凸轮参数的首地址为 D1000；

D30 地址的数值表示生成电子凸轮表格数据是否成功。

5.2.10 方便指令

IST 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
IST	状态初始化	16	否	IST (S) (D1) (D2)	7

该应用指令可实现一个典型的多个动作循环执行机构的控制状态初始化定义，并指定运行模式的输入信号、动作状态等变量。其中：

(S)为指定运行模式的输入的起始位变量元件地址，本指令会占用(S)~(S)+7 的连续 8 个地址单元，且每个变量的对应特定功能定义，见以下描述：

(D1)为指定自动操作模式中，使用 S 状态的最小序号；

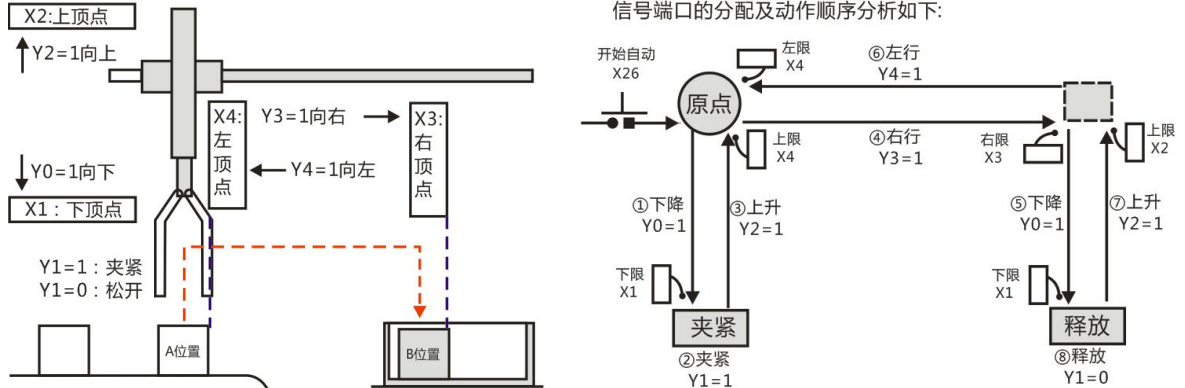
(D2)为指定自动操作模式中，使用 S 状态的最大序号；(D1)~(D2)为控制系统循环动作的状态序号，由此决定了其中的状态数。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)	√	√	√													
(D1)				√												
(D2)				√												

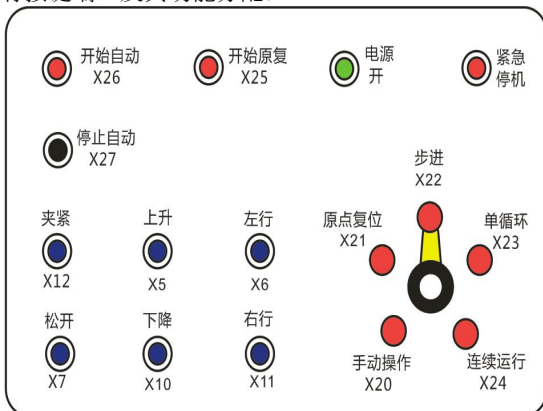
注：
 1)该指令在用户程序中只能使用 1 次；2)(D1)及(D2)只能用 S 变量 S20~S899；且必须 (D1)<(D2)，
 3)使用该指令时还用到系统专用的 M 变量。

例如下图所示，执行机构的动作依次是，抓取机构由原点下降到 A 工件位置，将工件抓取后提升到指定高度，再平移到指定位置后下降，到达指定位置后释放夹具，之后按原路返回，开始下一个循环动作。IST 指令可指定该操作机构的控制信号输入、状态转移的控制等，实现自动控制，还支持手动调试的单步动作、原点复位等。

信号端口的分配及动作顺序分析如下：



操作机构还需配备指令按键及状态切换开关，用手动试机、单次动作、循环动作等控制，以下是操作面板示意图，其中有按键端口及其功能分配：



对于如上图的应用，每个完整的循环动作分 8 个步骤（即 8 个状态），可使用如下的指令语句，实现控制系统的状态初始化：

```

M8000
┌───┐
└───┘ [ IST X20 S20 S27 ]
    
```

其中 (S) 指定 X20 为运行模式的起始输入，因此随后地址的 X21~X27 输入端口也将被使用，且功能动作特性将分别定义为：（使用其他 X、M 或 Y 等变量时同理类推）

X20：手动操作模式，用单个按钮接通或切断各控制输出信号；

X21：原点复归模式，按下原点复归用按钮时，使机械自动复归原点

X22：单步操作模式，每次按起动按钮，前进一个工序

X23：循环运行一次模式，在起点位置上按起动按钮时，进行一次循环的自动运行并在原点停止。途中按停止按钮，其工作停止，若再按起动按钮，在此继续动作至原点自动停止。

X24：连续运行模式，在起点位置上按起动按钮，开始连续运行。若按停止按钮，则运转至起点位置后停止。

X25：开始原点复归命令信号

X26：开始自动命令信号

X27：停止自动命令信号

注意：

这些端口信号中 X20~X24 决定了控制系统的运行模式，不能同时有为 ON 的状态，故建议采用旋转开关进行信号的选择切换。

(D1)和(D2)分别指定自动操作模式中，使用状态的最小和最大序号 S20~S27，共 8 个状态。

还需注意 IST 指令对如下的特殊变量的定义及使用要求：如果驱动 IST 指令，下列元件被自动切换控制，用户程序可以进行引用；为了配合 IST 指令的状态切换和控制，用户程序中还需对部分特殊变量进行操作，如下表说明：

IST 指令内部默认使用的变量		用户程序中驱动变量	
M8040	1=禁止所有状态的转移	M8043	1=原复完毕。在原复模式中，机器返回原点时，通过用户程序将该特殊 M 变量置位。
M8041	1=转移开始	M8044	1=原点条件。检测机器的原点条件，驱动该特殊辅助继电器。在全部模式中均置位。
M8042	1=启动脉冲	M8045	1=全部输出复位禁止。若在手动、原复、自动模式间进行转换时，机器不在原点位置时，则执行全部输出与动作状态的复位。但若已驱动 M8045，只有动作状态复位。
S0	手动操作初始状态	M8047	1=STL 监控有效。驱动 M8047 后，现正动作中的状态序号（S0-S899）按从小到大的顺序存入特殊辅助继电器 D8040-D8047 中。由此可以监控 8 点的动作状态序号。此外，若这些状态的任何一个动作，特殊辅助继电器 M8046 将动作。
S1	原点复归初始状态		
S2	自动运行初始状态		

在机器“自动运行”模式中，可进行自由转换：单步<-->循环一次<-->连续运行；

在机器运行中，在“手动操作”<-->“原点复归”<-->“自动运行”之间进行转换时，当全部输出复位后，转换后模式有效。（M8045 驱动时不能复位）

使用 IST 指令时，S10~S19 可作为原点复归用。因此，在编程中请勿将这些状态作为普通状态使用。另外，S0~S9 作为初始状态处理，S0~S2 作为如上述的手动操作，原点复归以及自动运行使用，而 S3~S9 则可以自由地使用。

编程时，IST 指令必须比状态 S0~S2 等一系列的 STL 电路优先编程。

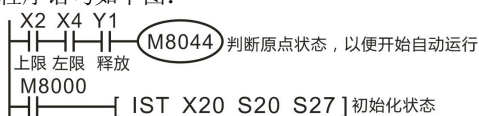
为了防止其中的 X20~X24 同时为 ON 状态，须用旋转开关。

原点复归完成（M8043）未动作时，如果在各个（X20），原点复归（X21），自动（X22、X23、X24）之间进行切换时，则所有输出进入 OFF 状态，并且自动运行在原点复归结束后，才可以再次驱动。

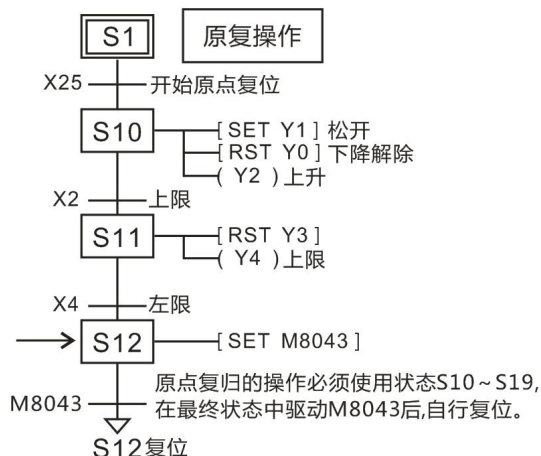
在用 IST 指令进行了控制指令的初始化之后，还需要对执行机构每个状态的动作，以及状态转移的条件进行编程，具体说明如下：

1. 系统初始化：定义原点复位条件；定义 IST 指令中使用的运行模式信号的输入端口、循环动作的状态变量。

使用的程序语句如下图：



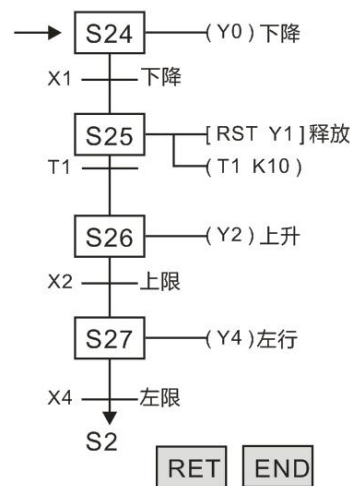
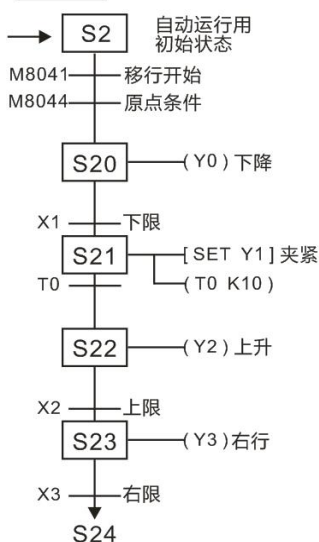
2. 手动操作：由操作盘上定义的命令信号驱动执行，见如下图中 S0 状态的程序语句。若没有手动模式，可以不需编写该部分的程序：



3. 原点复位：根据开始复位的命令信号、复位动作的顺序，设计复位程序，如上右图：

4. 自动运行：根据要求的动作条件和顺序、控制信号输出，编写程序如下图：

自动运行 单次/循环一次/连续

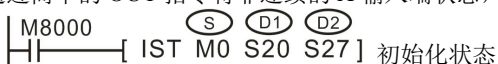


至此，控制系统就可以按前面的动作要求完成循环动作了。以上是为了方便阅读，采用了步进指令进行的编程说明，用户完全可以用等效梯形图来进行编程。

当一个控制系统的“自动运行”模式有不同的状态数，可参考上例编程，修订(D1)及(D2)的设置项，并在“自动运行”状态中进行相应的处理。

对于使用非连续 X 输入端的处理方法：

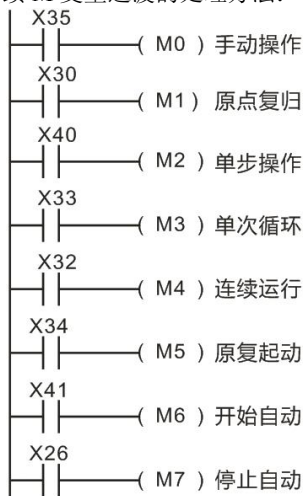
若需要采用非连续地址的 X 输入端口作为运行模式的给定输入，可采用 M 变量来进行“过渡”传送，即先通过简单的 OUT 指令将非连续的 X 输入端状态，逐个复制到某个连续地址的 M 变量，而使用如下指令：



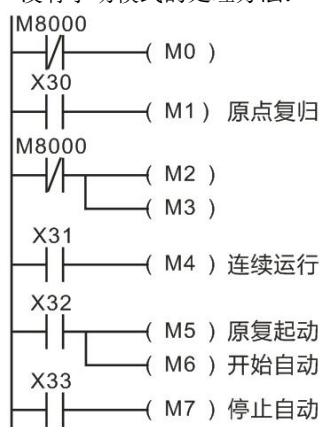
在 IST 中的 指定到该连续的 M0~M7 变量区。对于不存在的控制模式，也可用编程指令进行屏蔽，举例如下图中的 X 作为模式输入端与 M 变量的对应关系；对于不需要的模式输入 M 变量，只需将之固定清 0 就可以了：

X 输入端口不连续，采用连

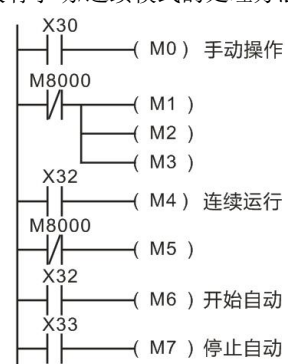
续 M 变量过渡的处理方法：



没有手动模式的处理方法：



仅有手动/连续模式的处理方法：



ABSD 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
ABSD	凸轮控制（绝对方式）	16	否	ABSD (S1) (S2) (D) (n)	9
DABSD		32	否		17

该指令完成的操作是多区段比较，用于实现凸轮控制，比较用的表格、计数器等均按绝对方式设置。该指令是主程序中扫描执行，比较结果受扫描时间的滞后影响。其中：

(S1)为比较表格的起始元件地址；

(S2)为计数器元件序号，使用 32bit 指令时，可为 32bit 计数器；

(D)为比较结果存放区的起始地址，占用(n)个连续地址的 bit 变量单元；

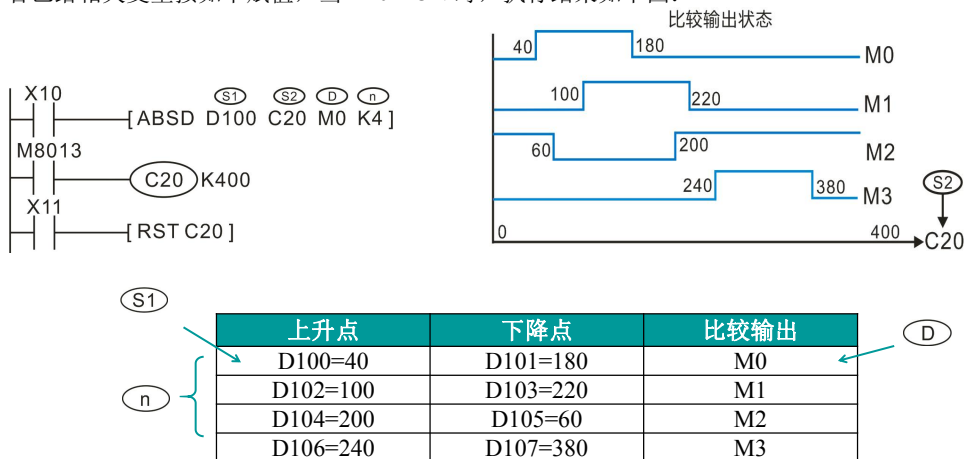
(n)为多段比较数据的组数。

当使用 32bit 指令时，(S1) (S2) (D) 均指向 32bit 变量，(n)也按 32bit 变量宽度进行计算。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)								√	√	√	√	√	√	√		
(S2)													√			
(D)				√												
(n)	常数，n=1~64；															
(S1)操作数为 KnX、KnY、KnM、KnS 时，若为 16bit 指令，必须指定 K4；若为 32bit 指令，必须指定 K8 且 X、Y、M、S 的元件编号必须是 8 的倍数；(S1)操作数在 16bit 指令时只能指定 C0~C199；32bit 指令时则只能指定 C200~C255；																

编程示例:

若已给相关变量按如下赋值，当 X10=ON 时，执行结果如下图：



使用说明:

ABSD 指令执行前，应给相关表格的各变量用 MOV 指令赋值；

即使 DABSD 指令中采用了高速指令，比较输出(D)也受用户程序扫描时间的滞后影响，对于需要及时响应的应用，可采用 HSZ 高速比较指令；

程序中只能使用 ABSD 指令一次。

INCD 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
INCD	凸轮控制增量方式	16	否	INCD (S1) (S2) (D) (n)	9

该指令完成的操作是多区段比较，用于实现凸轮控制，比较用的表格、计数器等均按增量方式设置。该指令是主程序中扫描执行，比较结果受扫描时间的滞后影响。其中：

(S1) 为比较表格的起始元件地址；

(S2) 为计数器元件序号，其相邻的 (S2)+1 单元则被用于计算比较匹配后计数器复位的次数。

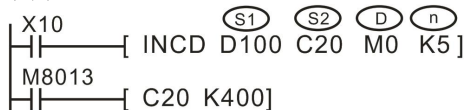
(D) 为比较结果存放区的起始地址，占用 (n) 个连续地址的 bit 变量单元；

(n) 为多段比较数据的组数。

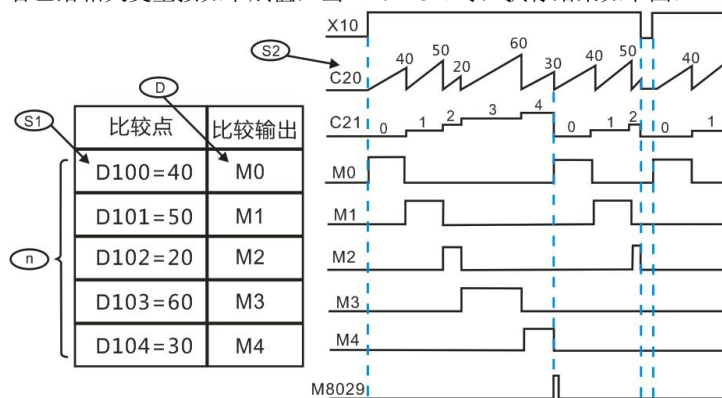
当使用 32bit 指令时，(S1) (S2) (D) 均指向 32bit 变量，(n) 也按 32bit 变量宽度进行计算。(n) 指定的最后工作完成时，标志 M8029 会置 ON，再次返回进行同样的工作。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)								√	√	√	√	√	√	√		
(S2)													√			
(D)		√	√	√												
(n)	常数，n=1~64;															
(S1)	操作数为 KnX、KnY、KnM、KnS 时，若为 16bit 指令，必须指定 K4；若为 32bit 指令，必须指定 K8 且 X、Y、M、S 的元件编号必须是 8 的倍数；															
(S1)	操作数在 16bit 指令时只能指定 C0~C199；32bit 指令时则只能指定 C200~C254；															

编程示例:



若已给相关变量按如下赋值，当 X10=ON 时，执行结果如下图：



指令使用说明:

INCD 指令执行前，应给相关表格的各变量用 MOV 指令赋值；

比较输出也受用户程序扫描时间的滞后影响，对于需要及时响应的应用，可采用 HSZ 高速比较指令；

程序中只能使用 INCD 指令一次。

TTMR 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
TTMR	示教定时器	16	否	TTMR (D) (n)	5

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(D)														√		
(n)					√	√										

该指令的功能是将指定输入端口的按键保持时间乘以 (D) 倍数后存入变量 (D)，一般用于参数设定。其中：

(D) 为按键保持时间，以秒为单位，乘以 n 倍数后的乘积，按键释放后 (D) 内容没有变化；而 (D)+1 单元则用于保存按键的按压时间，按键释放后 (D)+1 的内容被复位为 0，(D)+1 的时间单位为 100ms；

(D) 为倍数设定，注意实际倍数为 10n 计算方式，(n=0~2)

n=K0 时，实际倍数为 (D)×1； n=K1 时，实际倍数为 (D)×10； n=K2 时，实际倍数为 (D)×100。

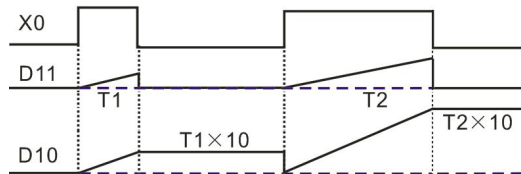
编程示例:

指令示例 1:



在 X10 闭合期间，D10=D11；

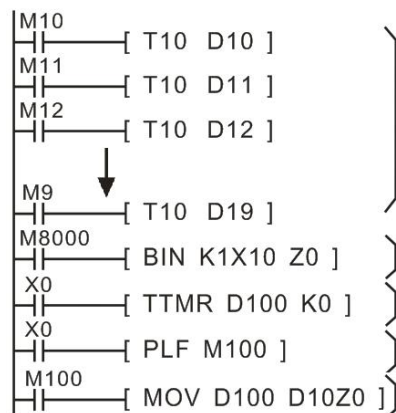
当 X10 断开后，D10 值维持不变，而 D11 则变为 0。



假如 X10 的按键保持时间为 T 秒，D10、D11 与 n 三者之间的关系如下：

n	D10	D11(单位: 100 毫秒)
K0: (单位: 秒)	1×T	D11=D10×10
K1: (单位: 100 毫秒)	10×T	D11=D10
K2: (单位: 10 毫秒)	100×T	D11=D10/10

指令示例 2:



使用 TMR 指令写入 10 组设置时间，将设置值预先写入 D10~D19，该组定时器为 100ms 型，因此示教数据的 1/10 为实际动作时间(秒)。将 1 位数字开关输入接于 X10~X13，使用 BIN 指令转换后，把数值并存放于 Z0 中。

X0 的 ON 时间（秒）存放于 D100 当中。

M100 为演示定时器按钮，X0 释放产生的一次扫描周期脉冲。

以指拨开关的设置号码当成间接指定的指针，然后将 D100 的内容传送至 D10Z0(D10~D19)中。

SER 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
SER	数据检索	16	否	SER (S1) (S2) (D) (n)	9
SERP		16	是		9
DSER		32	否		17
DSERP		32	是		17

该指令用于从一组数据中，查找相同数据的单元、同时对最大值、最小值的检索。其中：

操作数	位元件				字元件												
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z	
(S1)								√	√	√	√	√	√	√	√	√	
(S2)					√	√		√	√	√	√	√	√	√	√	√	
(D)								√	√	√	√	√	√	√	√	√	
(n)					√	√								√			

(S1)为数据组的起始地址；

(S2)为待检索的数据；

(D)为检索结果存放区的起始地址；

(n)为被检索数据区的长度，取值范围：16bit 指令：n=1~256；32bit 指令：n=1~128。

当使用 32bit 指令时，(S1) (S2) (D) 均指向 32bit 变量，(n) 也按 32bit 变量宽度进行计算。

编程示例：

X20
[SER (S1) (S2) (D) (n)
D10 D0 D80 K10]

(S1)	被检索数据例	(S2)	元件序号	参数状况
D10	(D10)=K100	比较数据 (D10) =K100	0	相等
D11	(D11)=K123		1	
D12	(D12)=K100		2	相等
D13	(D13)=K98		3	
D14	(D14)=K111		4	
D15	(D15)=K66		5	最小
D16	(D16)=K100		6	相等
D17	(D17)=K100		7	相等
D18	(D18)=K210		8	最大
D19	(D19)=K88		9	

检索结果：

(D)	参数	定义
D80	4	相等参数的个数
D81	0	第一个相等参数的序号
D82	7	最后一个相等参数的序号
D83	5	最小参数的序号
D84	8	最大相等参数的序号

使用说明：

- 1、当指令能流 X20 为 ON 时，方才进行比较；
- 2、比较的方法为有符号数的代数比较方法进行，例如 $-8 < 2$ ；
- 3、当最小值、最大值有多个时，分别显示序号最大的元件；
- 4、存储检索结果的单元占用 (D) 开始的 5 个连续单元。若不存在相等数据时，上例中的 D80~D82 均为 0。

STMR 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
STMR	特殊定时器	16	否	STMR (S) (m) (D)	7

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)												√				
(m)	常数, m=1~32767															
(D)		√	√	√												

该指令的功能是根据指令能流，产生 4 种延时动作的专用指令。其中：

(S)用于产生延迟动作的定时器序号，可用 T0~T199；

(m)为延时设定值，单位为 100ms，设定值范围为 K1~K32767；

(D)为延时动作输出元件的起始编号，共占用 4 个连续单元。

使用注意事项：在本指令中使用的定时器不得再用于其他指令中重复使用。

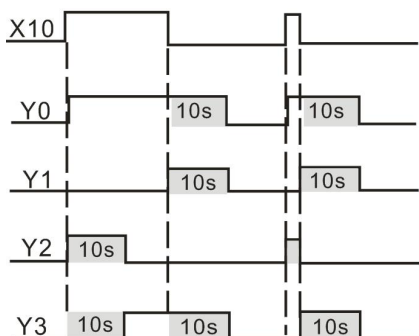
编程示例:

指令示例 1:

```

X10
|---[ STMR (S) (m) (D) ]

```



当 X10 由 OFF 变 ON 时，Y0=ON，到 X10 由 ON 变 OFF 时，延迟 10 秒后 Y0=OFF。

当 X10 由 ON 变 OFF 时，作一次 Y1=ON 输出。

当 X10 由 OFF 变 ON 时，作一次 Y2=ON 输出。

当 X10 由 OFF 变 ON 时，延迟 10 秒后 Y3=OFF；到 X10 由 ON 变为 OFF 时，延迟 10 秒后 Y3=OFF。

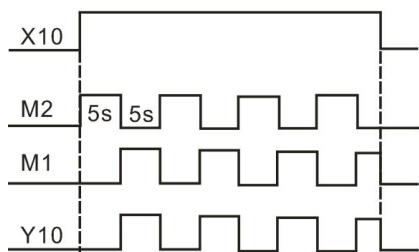
指令示例 2:

若在指令能流中引入(D)的元件，可方便地实现振荡器输出（此功能也可以用 ALT 指令来实现），如下图：

```

X10 M3
|---[ STMR (S) (m) (D) ]
M1
|---(Y10)

```



ALT 指令

指令描述:

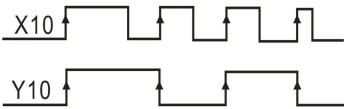
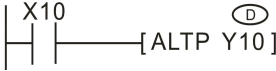
名称	功能	位数(bits)	是否脉冲型	指令格式	步数
ALT	交替输出	16	否	ATL D	3
ALTP		16	是		3

该指令的功能是能流有效时，将D元件的状态反转。其中D为位变量元件。一般使用脉冲执行型 ALTP 指令。

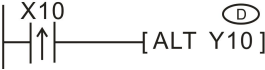
操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
D		√	√	√												

编程示例:

指令示例 1:



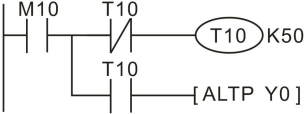
如下指令产生的动作与之相同:



指令示例 2:

若在指令能流中引入定时器，可方便地实现振荡器输出（此功能也可以用特殊定时器 STMR 指令来实现），如下

图:



RAMP 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
RAMP	斜坡信号	16	否	RAMP (S1) (S2) (D) (n)	9

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)														√		
(S2)														√		
(D)														√		
(n)	常数, 1~32767															

该指令的功能是在给定的两个数据中间, 在指定的时间区间, 进行线性插值, 按扫描执行的时间依次输出过程值, 直到区间末端的终点值为止。其中:

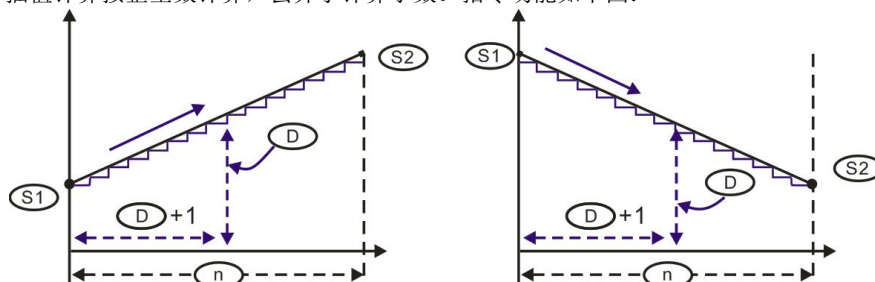
(S1) 斜坡信号的起始值单元;

(S2) 斜坡信号的终点值单元;

(D) 为线性插值信号的过程值存放单元, 而插值次数的定时器存放在 (D) + 1 单元;

(n) 完成插补过程的程序扫描执行次数。由于插值输出是在正常主循环中进行的, 为了保证插值输出的线性, 需要将程序执行设置为固定扫描方式 (见 M8039、D8039 说明)。

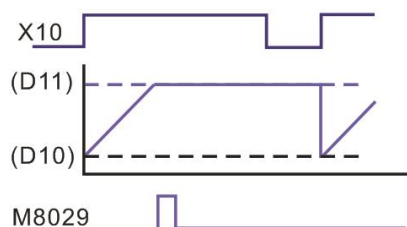
插值计算按整数计算, 丢弃了计算小数。指令功能如下图:



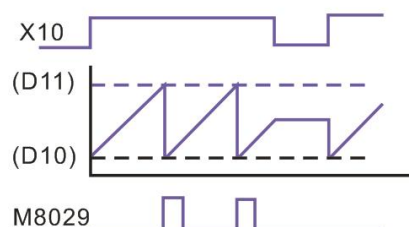
RAMP 指令执行有 2 种模式, 由 M8026 标志进行选择; 每次插值运算完毕, M8029 置位一个扫描周期。执行特点如下例:

X10 ——— [RAMP (S1) (S2) (D) (n)]

M8026=OFF



M8026=ON



ROTC 指令

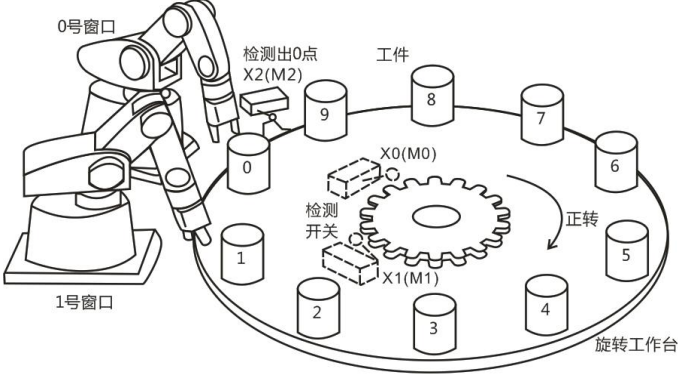
指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
ROTC	旋转工作台控制	16	否	ROTC(S)(m1)(m2)(D)	9

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)														√		
(D)		√	√	√												
(m1)	2~32767, m1 ≧ m2															
(m2)	0~32767, m1 ≧ m2															

该指令是用于旋转工作台上工件取放控制的简捷指令，旋转工作台的位置检测信号需按指定方式配置才能正常工作。其中：

- (S)计数变量的起始单元；
- (m1)旋转工作台上的工位数，必须(m1)≧(m2)；
- (m2)旋转工作台上的低速工位数，必须 (m1)≧(m2)；
- (D)为旋转台位置检测信号存放的起始单元，占用随后的 8 个位变量单元。



1 号配置方式如下图，图中 X0、X1 分别接 AB 正交编码器的 A 相和 B 相输出信号，也可采用机械开关得到正交相位的信号；X2 接用于 0 号工位的检测输入（当旋转到 0 号工位时为 ON 状态），由此 3 个信号即可检测旋转工作台的当前转动速度、转向和工位。

编程示例:



该代码实际使用的变量空间说明如下：

变量	功能定义	操作说明
D200	作为计数寄存器使用	由用户程序事先设定好该 3 个单元
D201	调用窗口号码设定	
D202	调用工件号码设定	
M0	A 相信号	
M1	B 相信号	
M2	0 点检测信号	
M3	高速正转	当 X10 为 ON 时，可以自动得到 M3~M7 的结果； 当 X10 为 OFF 时，M3~M7 均为 OFF。
M4	低速正转	
M5	停止	
M6	低速反转	
M7	高速反转	

在后续的用户程序中，将 M3~M7 从 Y 输出口中输出，控制外部执行元件即可。程序中只能使用 1 次 ROTC 指令。

SORT 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
SORT	数据排序	16	否	SORT (S) (m1) (m2) (D) (n)	17

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)														√		
(m1)	常数, 1~32															
(m2)	常数, 1~6															
(D)														√		
(n)					√	√								√		

该指令是将 m1 行×m2 列的数组（由 (S) (m1) (m2) 描述），以第 (n) 列参数排序后，存放于由 (D) 单元起始的变量区域。其中：

- (S)为第 1 行（或称第 1 条记录）的首个变量的起始单元；
 - (m1)数组的行数，或称记录数；
 - (m2)数组的列数，或称每条记录的栏目数；
 - (D)为排序后存放的起始单元，占用随后的变量单元数目与排序前的数组变量数目相同；
 - (n)为以排序为依据的数组列号。
- 其中 (n) 的值在 1~(m2) 范围。

编程示例:



当 X10=ON 时，开始排序运算，指令执行完毕，M8029 被置位 1 个程序扫描周期；若需要再次排序，需将 X10=OFF 一次。

上述指令的等效表格及其数据举例：

Diagram showing the SORT instruction parameters and data table:

Parameters: (S) points to the start of the source array. (m2) indicates the number of columns (4). (m1) indicates the number of rows (5).

列号	1	2	3	4
行号	学号	语文	数学	物理
1	D100 1	D105 85	D110 78	D115 85
2	D101 2	D106 82	D111 91	D116 81
3	D102 3	D107 77	D112 89	D117 88
4	D103 4	D108 90	D113 81	D118 75
5	D104 5	D109 87	D114 95	D119 77

按指令要求 $\textcircled{n}=\text{K2}$ 排序后的表格数据结果如下：

$\textcircled{D}$ $\textcircled{n}=\text{K2}$

列号	1	2	3	4
行号	学号	语文	数学	物理
1	D200 3	D205 77	D210 89	D215 88
2	D201 2	D206 82	D211 91	D216 81
3	D202 1	D207 85	D212 78	D217 85
4	D203 5	D208 87	D213 95	D218 77
5	D204 4	D209 90	D214 81	D219 75

若指令中 $\textcircled{n}=\text{K4}$ 排序后的表格数据结果如下：

$\textcircled{D}$ $\textcircled{n}=\text{K4}$

列号	1	2	3	4
行号	学号	语文	数学	物理
1	D200 4	D205 90	D210 81	D215 75
2	D201 5	D206 87	D211 95	D216 77
3	D202 2	D207 82	D212 91	D217 81
4	D203 1	D208 85	D213 78	D218 85
5	D204 3	D209 77	D214 89	D219 88

5.2.11 定位指令

DABS 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DABS	读出 ABS 当前值	32	否	DABS (S) (D1) (D2)	13

该指令是通过高速输入端口读取伺服驱动器中电机的绝对位置（ABS）数据。其中：

(S) 为读取伺服装置的输入信号，占用后续共 3 个单元。

(D1) 为传送至伺服装置的控制信号，占用后续共 3 个单元。

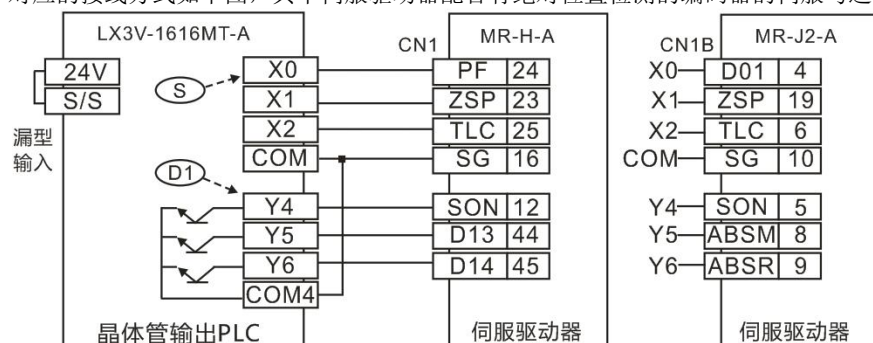
(D2) 则是从伺服读取数据的存储单元，32bit 宽度，占用 (D2)+1、(D2) 单元，通常指定 D8140。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)	√	√	√	√												
(D1)		√	√	√												
(D2)									√	√	√	√	√	√	√	√

编程示例:



对应的接线方式如下图，其中伺服驱动器配合有绝对位置检测的编码器的伺服马达：



当指令驱动 M10 变为 ON 时开始读取，读取完毕，M8029 标志置为 ON；

当指令执行过程中，驱动标志变为 OFF，读取操作即被中断；

读取 ABS 数据的编程举例如下，当 X6 端子闭合时才读取 ABS 数据，若 5s 内没有完成读取操作，超时标志 M21 被置位，编程如下左图：



ABS 读操作的信号时序如上右图，指令执行时，PLC 会按该实现自动完成与伺服驱动器的访问操作。

ZRN 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
ZRN	原点回归	16	否	ZRN (S1) (S2) (S3) (D)	9
DZRN		32	否		13

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√	√		√	√	√	√	√	√	√	√	√
(S2)					√	√		√	√	√	√	√	√	√	√	√
(S3)	√	√	√	√												
(D)		√														

该指令是 PLC 与伺服驱动器配合工作时，用指定脉冲速度和脉冲输出口，让执行机构向动作原点（DOG）移动，直到遇到原点信号满足条件为止。其中：

(S1)为原点回归动作的启始速度。16bit 指令时，范围是 10~32，767Hz；32bit 指令时，范围是 10~100，000Hz；

(S2)为原点信号变为 ON 后的爬行速度，范围是 10~32，767Hz；

(S3)原点信号（DOG）输入，虽然 XYMS 信号都可以，但只有 X 信号的及时性最好；

(D)为脉冲输出起始地址，对于 LX3V(2N)/LX3VP/LX3VE 型可指定 Y0~Y3，LX3V(1S)型可指定 Y0~Y1；

相对位置控制指令 DRVI 和绝对位置控制指令 DRVA 在执行时，设定超过最高频率，按照最高频率运行，设定小于最低频率，按照最低频率运行。

该指令的减速时间和最低频率数设定可参照绝对定位指令 DRVA。控制器会计算自身已发出的正转脉冲或反转脉冲数，并将之保存在寄存器[D8141，D8140]（Y000）和[D8143，D8142]（Y001）。但该寄存器的数据在断电时会消失，故上电时和初始运行时，必须执行原点回归指令 ZRN，以事先将机械动作的原点位置的数据写入。

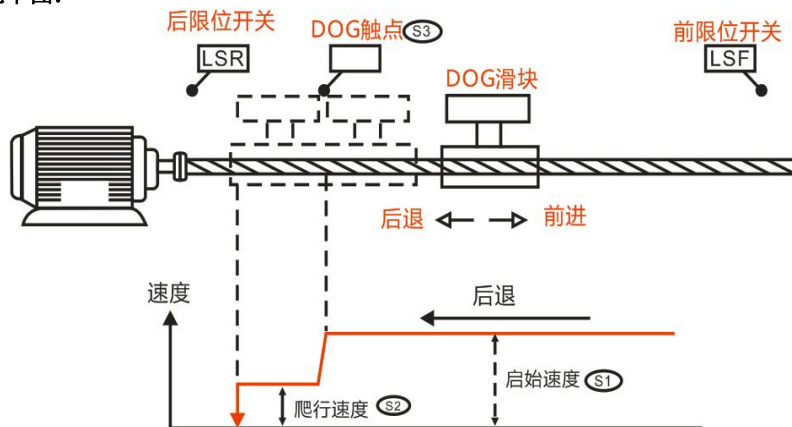
编程示例：

```

M10 ──┐
        │ [ ZRN K1000 (S1) (S2) (S3) (D) ]
        └─┘
    
```

本指令的动作是，当 M10 变为 ON 后，PLC 从 Y0 高速输出口，开始以 1000Hz 发出脉冲，令步进电机向原点作后退，当遇到 DOG 信号变为 ON 时（此时 DOG 滑块刚好碰到 DOG 触点），输出频率降为 80Hz，作低速爬行，直到 DOG 信号再变为 OFF 时，Y0 停止输出脉冲，向当前值寄存器（Y000：[D8141，D8140]，Y001：[D8143，D8142]）中写入 0。另外，M8140（清零信号输出功能）ON 时，同时输出清零信号。随后，当执行完成标志（M8029）置为 ON 的同时，脉冲输出中监控（Y000：[M8147]，Y001：[M8148]）变为 OFF。

参见下图：



本指令执行中，涉及的系统变量有：

1. D8141（高位）， D8140（低位）]： Y000 输出的当前值寄存器（使用 32 位）
2. D8143（高位）， D8142（低位）]： Y001 输出的当前值寄存器（使用 32 位）
3. M8145： Y000 脉冲输出停止（立即停止）
4. M8146： Y001 脉冲输出停止（立即停止）
5. M8147： Y000 脉冲输出中监控（BUSY/READY）
6. M8148： Y001 脉冲输出中监控（BUSY/READY）

7. 由于伺服驱动器对位置信息具有掉电保持功能，该指令并不需要每次上电时都需进行；指令执行中，仅能单方向运动（后退方向），所以回原点动作必须在 DOG 信号前端开始。

PLSV 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
PLSV	可变速脉冲输出	16	否	PLSV (S) (D1) (D2)	9
DPLSV		32	否		13

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)					√	√		√	√	√	√	√	√	√	√	√
(D1)		√														
(D2)		√	√	√												

该指令是按指定的端口、频率和运行方向输出脉冲频率，没有加减速过程，当驱动能流无效时，输出脉冲直接停止。只有晶体管输出 PLC 才能使用该指令。其中：

(S)为指定的输出脉冲频率。16bit 指令时，范围是 1~32, 767Hz；-1~-32, 768Hz ；32bit 指令时，范围是 1~200, 000Hz；-1~-200, 000Hz；32bit 指令时，当 (S)大于 200000，则按 200000 处理，(S)小于-200000，按-200000 处理其中负号表示反方向运行的指令信号；

(D1)为脉冲输出端口，对于 LX3V(2N)/LX3V 型可指定 Y0~Y3，LX3V(1S)型可指定 Y0~Y1；

(D2)运行方向输出端口或位变量，输出为 ON 状态，表示为正向运行；否则为反向运行。

编程示例:



语句表示，当 M1 为 ON 时，由 Y1 端口输出 10kHz 频率的脉冲，Y4 用于控制运行方向，若 Y4=ON 表示为正方向。

本指令执行中，涉及的系统变量有：

1. D8141（高字节），D8140（低字节）：Y000 输出脉冲数。反转时减少。（使用 32 位）
2. D8143（高字节），D8142（低字节）：Y001 输出脉冲数。反转时减少。（使用 32 位）
3. D8151（高字节），D8150（低字节）：Y002 输出脉冲数。反转时减少。（使用 32 位）
4. D8153（高字节），D8152（低字节）：Y003 输出脉冲数。反转时减少。（使用 32 位）
5. M8145：Y000 脉冲输出停止（立即停止）
6. M8146：Y001 脉冲输出停止（立即停止）
7. M8152：Y002 脉冲输出停止（立即停止）
8. M8153：Y003 脉冲输出停止（立即停止）
9. M8147：Y000 脉冲输出中监控（BUSY/READY）
10. M8148：Y001 脉冲输出中监控（BUSY/READY）
11. M8149：Y002 脉冲输出中监控（BUSY/READY）
12. M8150：Y003 脉冲输出中监控（BUSY/READY）

DRV1 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DRV1	相对定位	16	否	DRV1 (S1) (S2) (D1) (D2)	9
DDRV1		32	否		17

该指令是按指定的端口、频率和运行方向输出指定的脉冲数，令伺服执行机构在当前位置的基础上作给定偏移量的运动。只有晶体管输出 PLC 才能使用该指令。其中：

(S1)为指定的本次输出脉冲数。16bit 指令时，范围是-32768~32,767；32bit 指令时，范围是-2,147,483,648~2,147,483,647。其中负号表示反方向；

(S2)为指定的输出脉冲频率，16bit 指令时，范围为 10~32767Hz；32bit 指令时，范围为 10~100,000Hz；设定超过最高频率，按照最高频率运行，设定小于最低频率，按照最低频率运行。

(D1)为脉冲输出端口；对于 LX3V(2N) 型可指定 Y0~Y3；LX3V(1S)型可指定 Y0~Y1；

(D2)运行方向输出端口或位变量，输出为 ON 状态，表示为正向运行；否则为反向运行。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√	√		√	√	√	√	√	√	√	√	√
(S2)					√	√		√	√	√	√	√	√	√	√	√
(D1)		√														
(D2)		√	√	√												

输出脉冲数，是相对于下面的当前值寄存器作为相对位置：

向 [Y000] 输出时，当前寄存器为[D8141（高字节）， D8140（低字节）]（使用 32 位）

向 [Y001] 输出时，当前寄存器为[D8143（高字节）， D8142（低字节）]（使用 32 位）

向 [Y002] 输出时，当前寄存器为[D8151（高字节）， D8150（低字节）]（使用 32 位）

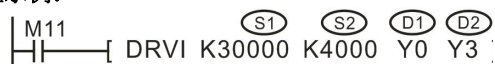
向 [Y003] 输出时，当前寄存器为[D8153（高字节）， D8152（低字节）]（使用 32 位）

在指令执行过程中，即使改变操作数的内容，也无法在当前运行中表现出来。只在下一次指令执行时才有效。

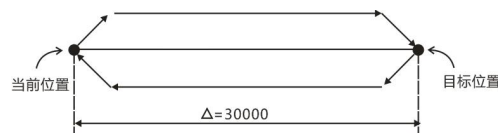
若在指令执行过程中，指令驱动的接点变为 OFF 时，将减速停止。此时执行完成标志 M8029 不会动作；

指令驱动接点变为 OFF 后，在脉冲输出标志 M8147（Y000）、M8148（Y001）、M8149（Y002）、M8150（Y003）处于 ON 时，将不接受指令的再次驱动。

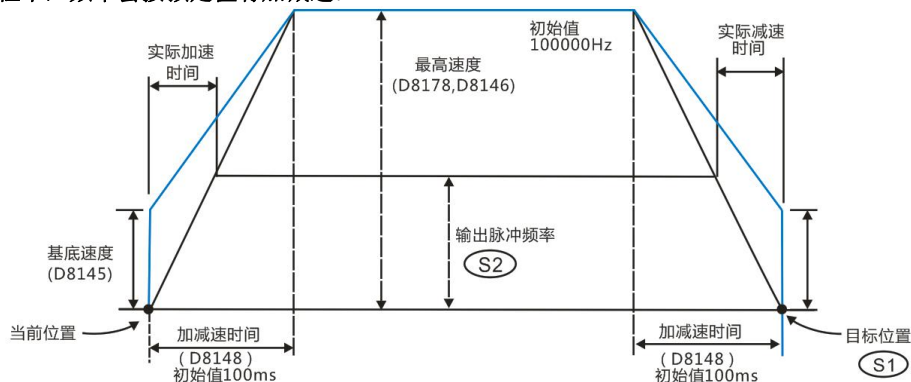
编程示例:



表示以 4kHz 的频率、由 Y0 端口输出 30000 个脉冲，令外部伺服执行机构运行，方向则由 Y3 决定。



脉冲输出过程中，频率会按预定值有加减速：



实际能够输出的输出脉冲频率的最低频率数，根据以下公式所决定：

$$\text{输出脉冲频率的最低频率数} = \sqrt{\text{最高速度}[\text{D8147}, \text{D8146}] \div (2 \times (\text{加减速时间}[\text{D8148}] \text{ms} \div 1000))}$$

即使指定了低于上面计算结果的数值，仍将输出计算值的频率。加速初期和减速最终部分的频率也不可低于上述计算结果。

指令执行中涉及的系统变量如下：

【D8145】：执行 DRV1，DRVA 指令时的基底速度。控制步进电机时，设定速度时需考虑步进电机的共振区域和自动起动频率。设定范围：最高速度（D8147，D8146）的 1/10 以下。超过该范围时，自动降为最高速度的 1/10 数值运行。

【D8147（高字节），D8146（低字节）】：执行 DRV1，DRVA 指令时的最高速度。指定的输出脉冲频率必须小于该最高速度。设定范围：10 ~ 100，000（Hz）。

【D8148】：执行 DRV1，DRVA 指令时的加减速时间。加减速时间表示到达最高速度（D8147，D8146）所需的时间。因此，当输出脉冲频率 $\leq$ 低于最高速度（D8147，D8146）时，实际加减速时间会缩短。设定范围：10 ~ 32000（ms）

【M8145】：Y000 脉冲输出停止（立即停止）

【M8146】：Y001 脉冲输出停止（立即停止）

【M8152】：Y002 脉冲输出停止（立即停止）

【M8153】：Y003 脉冲输出停止（立即停止）

【M8147】：Y000 脉冲输出中监控（BUSY/READY）

【M8148】：Y001 脉冲输出中监控（BUSY/READY）

【M8149】：Y002 脉冲输出中监控（BUSY/READY）

【M8150】：Y003 脉冲输出中监控（BUSY/READY）

注意事项：

- 1、定位指令(ZRN/PLSV。DRV1/DRVA)在程序中可以多次使用，但是不要对同一输出口同时进行输出操作。
- 2、当指令的驱动能流 OFF 之后，再次驱动 ON 时，必须在状态位（Y000：[M8147]，Y001：[M8148]，Y002：[M8149]，Y003：[M8150]）OFF 后，再经过一个运算周期后方能再次驱动。
- 3、定位指令再次驱动时，必须有 1 个周期以上的 OFF 时间，若已比上述条件更短的时间内执行再驱动时，将在最初指令执行（运算）时发生[运算错误]。

DRVA 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DRVA	绝对定位	16	否	DRVA (S1) (S2) (D1) (D2)	9
DDRVA		32	否		17

该指令是按指定的端口、频率和运行方向输出脉冲，令伺服执行机构运动到指定目的点。只有晶体管输出 PLC 才能使用该指令。其中：

(S1)为指定的目标位置(绝对位置)。16bit 指令时，范围是-32768~32, 767；32bit 指令时，范围是-2, 147, 483, 648~2, 147, 483, 647。

若(D1)=[Y000]，对应[D8I41(高字节)，D8I40(低字节)](使用 32 位)为绝对位置；

若(D1)=[Y001]，对应[D8I43(高字节)，D8I42(低字节)](使用 32 位)为绝对位置；

若(D1)=[Y002]，对应[D8I51(高字节)，D8I50(低字节)](使用 32 位)为绝对位置；

若(D1)=[Y003]，对应[D8I53(高字节)，D8I52(低字节)](使用 32 位)为绝对位置；

其中负号表示反方向。反转时，当前值寄存器的数值减小。

(S2)为指定的输出脉冲频率，范围为 10~32, 767Hz (16bit 指令)；或 10~100, 000Hz (32bit 指令)；设定超过最高频率，按照最高频率运行，设定小于最低频率，按照最低频率运行。

(D1)为脉冲输出端口；对于 LX3V(2N) 型可指定 Y0~Y3；LX3V(1S)型可指定 Y0~Y1；

(D2)运行方向输出端口或位变量，可根据(S1)和当前位置的差值决定，输出为 ON 状态，表示为正向运行；否则为反向运行。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√	√		√	√	√	√	√	√	√	√	√
(S2)					√	√		√	√	√	√	√	√	√	√	√
(D1)		√														
(D2)		√	√	√												

在指令执行过程中，即使改变操作数的内容，也无法在当前运行中表现出来。只在下一次指令执行时才有效。

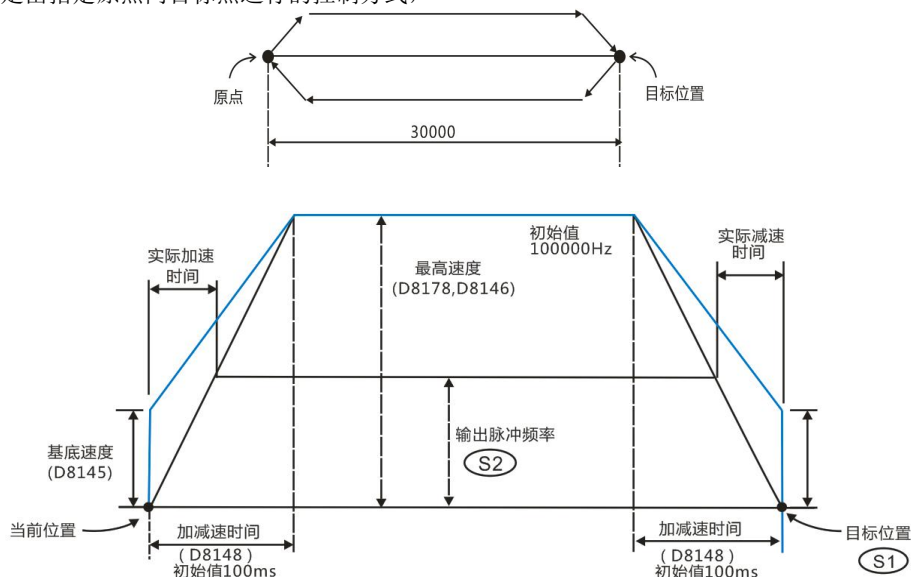
若在指令执行过程中，指令驱动的接点变为 OFF 时，将减速停止。此时执行完成标志 M8029 不会动作；

指令驱动接点变为 OFF 后，在脉冲输出标志 M8147 (Y000)、M8148 (Y001) 处于 ON 时，将不接受指令的再次驱动。

编程示例:



该指令是由指定原点向目标点运行的控制方式，



脉冲输出过程中，频率会按预定值有加减速：

实际能够输出的输出脉冲频率的最低频率数，根据以下公式所决定：

$$\text{输出脉冲频率的最低频率数} = \sqrt{\text{最高速度}[\text{D8147}, \text{D8146}] \div (2 \times (\text{加减速时间}[\text{D8148}] \text{ms} \div 1000))}$$

即使指定了低于上面计算结果的数值，仍将输出计算值的频率。加速初期和减速最终部分的频率也不可低于上述计算结果。

指令执行中涉及的系统变量如下：

【D8145】：执行 DRVI，DRVA 指令时的基底速度。控制步进电机时，设定速度时需考虑步进电机的共振区域和自动起动频率。设定范围：最高速度（D8147，D8146）的 1/10 以下。超过该范围时，自动降为最高速度的 1/10 数值运行。

【D8147（高字节），D8146（低字节）】：执行 DRVI，DRVA 指令时的最高速度。指定的输出脉冲频率必须小于该最高速度。设定范围：10 ~ 100，000（Hz）。

【D8148】：执行 DRVI，DRVA 指令时的加减速时间。加减速时间表示到达最高速度（D8147，D8146）所需的时间。因此，当输出脉冲频率 低于最高速度（D8147，D8146）时，实际加减速时间会缩短。设定范围：10 ~ 32000（ms）

【M8145】：Y000 脉冲输出停止（立即停止）

【M8146】：Y001 脉冲输出停止（立即停止）

【M8152】：Y002 脉冲输出停止（立即停止）

【M8153】：Y003 脉冲输出停止（立即停止）

【M8147】：Y000 脉冲输出中监控（BUSY/READY）

【M8148】：Y001 脉冲输出中监控（BUSY/READY）

【M8149】：Y002 脉冲输出中监控（BUSY/READY）

【M8150】：Y003 脉冲输出中监控（BUSY/READY）

注意事项：

- 1、定位指令(ZRN/PLSV。DRVI/DRVA)在程序中可以多次使用，但是不要对同一输出口同时进行输出操作。
- 2、当指令的驱动能流 OFF 之后，再次驱动 ON 时，必须在状态位（Y000: [M8147]，Y001: [M8148]，Y002: [M8149]，Y003: [M8150]）OFF 后，经过一个运算周期后，方能再次驱动。
- 3、定位指令再次驱动时，必须有 1 个周期以上的 OFF 时间，若已比上述条件更短的时间内执行再驱动时，将在最初指令执行（运算）时发生[运算错误]。

DVIT 指令

指令描述:

名称	功能	位数 (bits)	是否脉冲型	指令格式	步数
DVIT	中断定位	16	否	DVIT(S1)(S2)(D1)(D2)(D3)	9
DDVIT		32	否		13

该指令仅 3VP/3VE/3VM 系列 PLC 支持。

该指令是按指定的端口、频率和运行方向输出指定的脉冲数，当收到中断信号时，发送指定的脉冲数后停止。只有晶体管输出 PLC 才支持该指令的调用，其中：

(S1)为指定的本次输出脉冲数。16bit 指令时，范围是-32768~32, 767；32bit 指令时，范围是-2, 147, 483, 648~2, 147, 483, 647。其中负号表示反方向；

(S2)为指定的输出脉冲频率，16bit 指令时，范围为 10~32767Hz；32bit 指令时，范围为 10~200, 000Hz；

(D1)为脉冲输出端口；可指定 Y0~Y3；

(D2)运行方向输出端口或位变量，输出为 ON 状态，表示为正向运行；否则为反向运行。

(D3)为中断信号，当接收到中断信号后，运行指定的脉冲数后停止。

操作数	位元件					字元件										
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√	√	√	√	√	√	√	√	√	√	√	√
(S2)					√	√	√	√	√	√	√	√	√	√	√	√
(D1)		√														
(D2)		√	√	√												
(D3)		√	√	√												

输出脉冲数，是相对于下面的当前值寄存器作为相对位置：

向 [Y000] 输出时，当前寄存器为[D8I41（高字节）， D8I40（低字节）]（使用 32 位）

向 [Y001] 输出时，当前寄存器为[D8I43（高字节）， D8I42（低字节）]（使用 32 位）

向 [Y002] 输出时，当前寄存器为[D8I51（高字节）， D8I50（低字节）]（使用 32 位）

向 [Y003] 输出时，当前寄存器为[D8I53（高字节）， D8I52（低字节）]（使用 32 位）

1、在指令执行过程中，即使改变操作数的内容，也无法在当前运行中表现出来。只在下次指令执行时才有效。

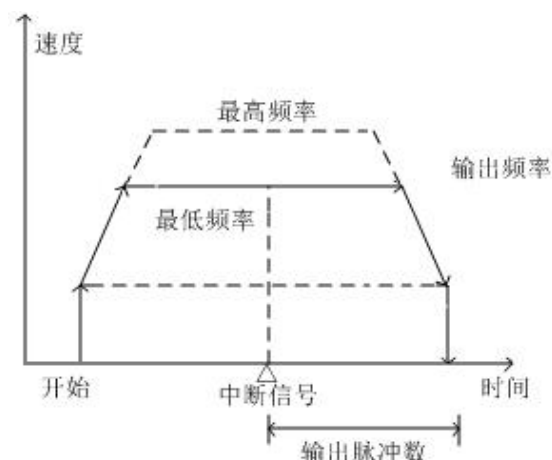
2、若在指令执行过程中，指令驱动的接点变为 OFF 时，将减速停止。此时执行完成标志 M8029 不会动作；

3、指令驱动接点变为 OFF 后，在脉冲输出中断标志 M8147（Y000）、M8148（Y001）、M8149（Y002）、M8150（Y003）处于 ON 时，将不接受指令的再次驱动。

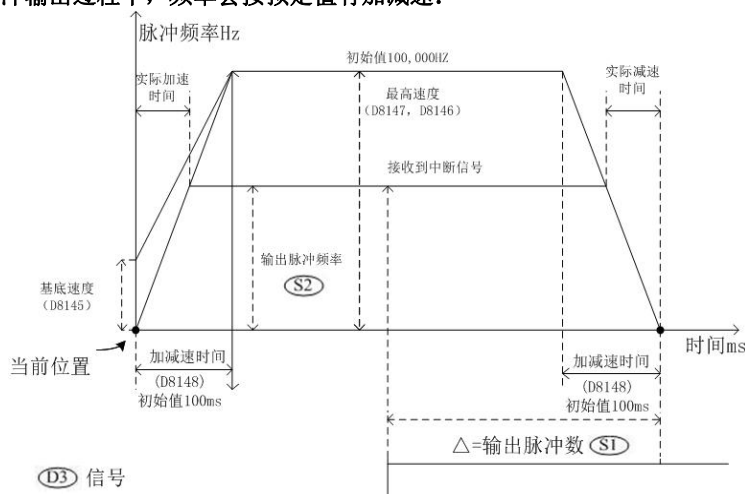
编程示例:



表示以 4kHz 的频率、由 Y0 端口输出脉冲，令外部伺服执行机构运行，当收到 M1 信号时，发送 30000 个脉冲，方向则由 Y1 决定。



脉冲输出过程中，频率会按预定值有加减速：



实际能够输出的输出脉冲频率的最低频率数，根据以下公式所决定：

$$\text{输出脉冲频率的最低频率数} = \sqrt{\frac{\text{最高速度 } [D8147, D8146] \text{ Hz} + (2 \times (\text{加减速时间 } [D8148] \text{ ms} + 1000))}{\Delta = \text{输出脉冲数 } (S1)}}$$

即使指定了低于上面计算结果的数值，仍将输出计算值的频率。加速初期和减速最终部分的频率也不可低于上述计算结果。

指令执行中涉及的系统变量如下：

[D8145]：执行 DRVI, DRVA 指令时的基底速度。控制步进电机时，设定速度时需考虑步进电机的共振区域和自动起动频率。设定范围：最高速度（D8147, D8146）的 1/10 以下。超过该范围时，自动降为最高速度的 1/10 数值运行。

[D8147（高字节），D8146（低字节）]：执行 DRVI, DRVA 指令时的最高速度。指定的输出脉冲频率必须小于该最高速度。设定范围：10 ~ 200,000（Hz）。

[D8148]：执行 DRVI, DRVA 指令时的加减速时间。加减速时间表示到达最高速度（D8147, D8146）所需的时间。因此，当输出脉冲频率（S2）低于最高速度（D8147, D8146）时，实际加减速时间会缩短。设定范围：50 ~ 5,000（ms）。

- [M8145]：Y000 脉冲输出停止（立即停止）
- [M8146]：Y001 脉冲输出停止（立即停止）
- [M8152]：Y002 脉冲输出停止（立即停止）
- [M8153]：Y003 脉冲输出停止（立即停止）
- [M8147]：Y000 脉冲输出中监控（BUSY/READY）
- [M8148]：Y001 脉冲输出中监控（BUSY/READY）
- [M8149]：Y002 脉冲输出中监控（BUSY/READY）
- [M8150]：Y003 脉冲输出中监控（BUSY/READY）

注意事项

定位指令(ZRN/PLSV, DRVI/DRVA)在程序中可以多次使用，但是不要对同一输出口同时进行输出操作。

当指令的驱动能流 OFF 之后，再次驱动 ON 时，必须在状态位（Y000: [M8147]; Y001: [M8148]; Y002: [M8149]; Y003: [M8150]）OFF 后，经过一个运算周期后，方能再次驱动。

定位指令再次驱动时，必须有 1 个周期以上的 OFF 时间，若以比上述条件更短的时间内执行再驱动时，将在最初指令执行（运算）时发生[运算错误]。

SCL 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
SCL	定坐标	16	否	SCL (S1) (S2) (D1)	7
DSCL		32	否		13

操作数	位元件				字元件										
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√	√	√	√	√	√	√	√	√	√	√
(S2)													√		
(D1)							√	√	√	√	√	√	√	√	√

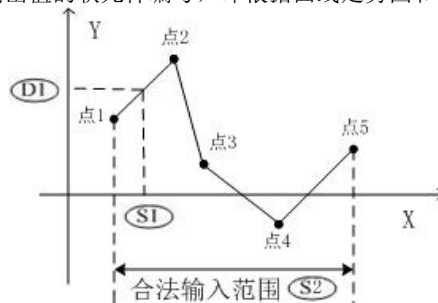
该指令根据指定的转换特性，将 (S1) 指定的输入值执行定坐标，然后保存到 (D1) 指定的软元件编号中。定坐标用的转换，是依据保存在 (S2) 指定软元件开始的数据表格执行。其中：

(S1)：执行定坐标的输入值或者保存输入值的软元件编号，即 X 轴坐标值；

(S2)：定坐标使用的转换表格的软元件起始编号，即曲线走势图；该软元件将被连续占用，具体查看下表：

16 位指令			32 位指令		
设定项目	设定数据表格的软元件分配		设定项目	设定数据表格的软元件分配	
坐标点数假设为 5	(S2)		坐标点数假设为 5	[(S2)+1, (S2)]	
点 1	X 坐标	(S2)+1	点 1	X 坐标	[(S2)+3, (S2)+2]
	Y 坐标	(S2)+2		Y 坐标	[(S2)+5, (S2)+4]
点 2	X 坐标	(S2)+3	点 2	X 坐标	[(S2)+7, (S2)+6]
	Y 坐标	(S2)+4		Y 坐标	[(S2)+9, (S2)+8]
点 3	X 坐标	(S2)+5	点 3	X 坐标	[(S2)+11, (S2)+10]
	Y 坐标	(S2)+6		Y 坐标	[(S2)+13, (S2)+12]
点 4	X 坐标	(S2)+7	点 4	X 坐标	[(S2)+15, (S2)+14]
	Y 坐标	(S2)+8		Y 坐标	[(S2)+17, (S2)+16]
点 5	X 坐标	(S2)+9	点 5	X 坐标	[(S2)+19, (S2)+18]
	Y 坐标	(S2)+10		Y 坐标	[(S2)+21, (S2)+20]

(D1)：保存被定坐标控制的输出值的软元件编号，即根据曲线走势图和 X 轴坐标值得出的 Y 轴坐标值。



1、16 位指令

根据 (S2) 的表格曲线图，找到 X 轴输入值为 (S1) 对应的 Y 轴数值 (D1)，若输出的 Y 轴数值不是整数，第一位小数将四舍五入输出。



设定项目		设定数据表格的软元件分配
坐标点数假设为 5		D10
点 1	X 坐标	D11
	Y 坐标	D12
点 2	X 坐标	D13
	Y 坐标	D14
点 3	X 坐标	D15
	Y 坐标	D16
点 4	X 坐标	D17
	Y 坐标	D18
点 5	X 坐标	D19
	Y 坐标	D20

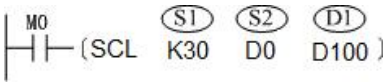
2、32 位指令

根据(S2)的表格曲线图，找到 X 轴输入值为(S1)对应的 Y 轴数值(D1)，若输出的 Y 轴数值不是整数，第一位小数将四舍五入输出。



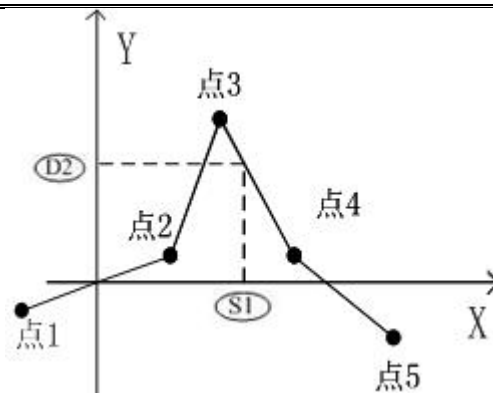
设定项目		设定数据表格的软元件分配
坐标点数假设为 5		(D10, D11)
点 1	X 坐标	(D11, D12)
	Y 坐标	(D13, D14)
点 2	X 坐标	(D15, D16)
	Y 坐标	(D17, D18)
点 3	X 坐标	(D19, D20)
	Y 坐标	(D21, D22)
点 4	X 坐标	(D23, D24)
	Y 坐标	(D25, D26)
点 5	X 坐标	(D27, D28)
	Y 坐标	(D29, D30)

3、编程实例：



其中：(S2)的数据以及形成的曲线图如下所示：

设定项目		设定数据表格的软元件分配	
		软元件地址	软元件值
坐标点数		D0	K5
点 1	X 坐标	D1	K-5
	Y 坐标	D2	K15
点 2	X 坐标	D3	K5
	Y 坐标	D4	K15
点 3	X 坐标	D5	K25
	Y 坐标	D6	K30
点 4	X 坐标	D7	K40
	Y 坐标	D8	K5
点 5	X 坐标	D9	K65
	Y 坐标	D10	K-10



可算出当 X 轴为 30 时，Y 轴的数据为 22，即 D100=22。

注意：

1、表格数据的 X 坐标未按照升序进行排列，报 6706 号错误。

由于运算从数据表格软原件编号的低位开始检索，在 X 坐标升序 排列的区间内，可以正常执行本指令进行坐标输出。

2、(S1)在表格设定的范围之外，报 6706 号错误。

3、至少需要两个点来进行运算。

4、如果运算的 X 点坐标存在两个或两个以上，输出第二个点的 Y 坐标。

G90G01 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
G90G01	绝对位置直线插补	16	否	G90G01(S1)(S2)(D1)(D2)	17

其中:

(S1): 目标位置, 占用 6 个连续地址, 范围是 -2, 147, 483, 648 ~ 2, 147, 483, 647。

(S1): X 的目标位置(绝对定位); (S1)+2: Y 的目标位置(绝对定位); (S1)+4: Z 的目标位置(绝对定位)。

(S2): 合成的输出频率。

(D1): 高速输出端口, 只能指定 Y0/Y2, 占用连续地址(Y0, Y1, Y2, Y3)。

M8108: 设置切换直线插补的模式。 M8108=0, 为二轴直线插补模式; M8108=1, 为三轴直线插补模式。

若(D1)指定 Y0 且 M8108=0, 占用 Y0, Y1; M8108=1, 占用 Y0, Y1, Y2;

若(D1)指定 Y2 且 M8108=0, 占用 Y2, Y3; M8108=1, 则软件报错;

(D2): 脉冲方向输出端口, 占用 3 个连续地址。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)														√		
(S2)					√	√								√		
(D1)		√														
(D2)		√	√													

1、使用指令插补时, 参数设置(如加减速时间等下述参数)以 X 轴(Y0)为准。

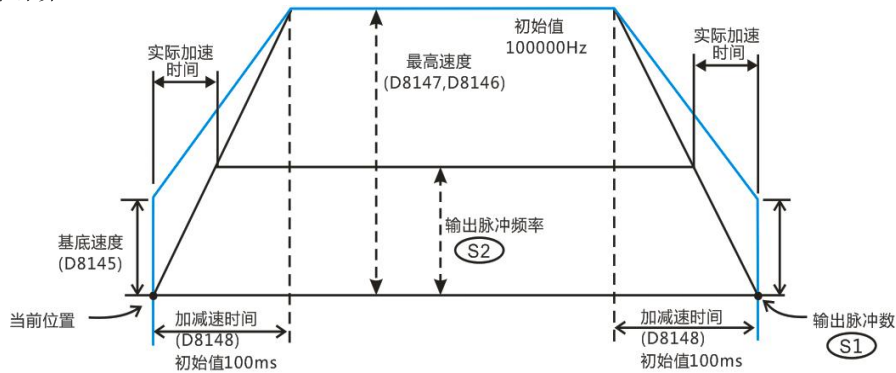
2、仅支持梯形加减速;

3、实际能够输出的合成频率 S (最低频率值) 即输出的合成频率的最低基底频率, 计算模式如下:

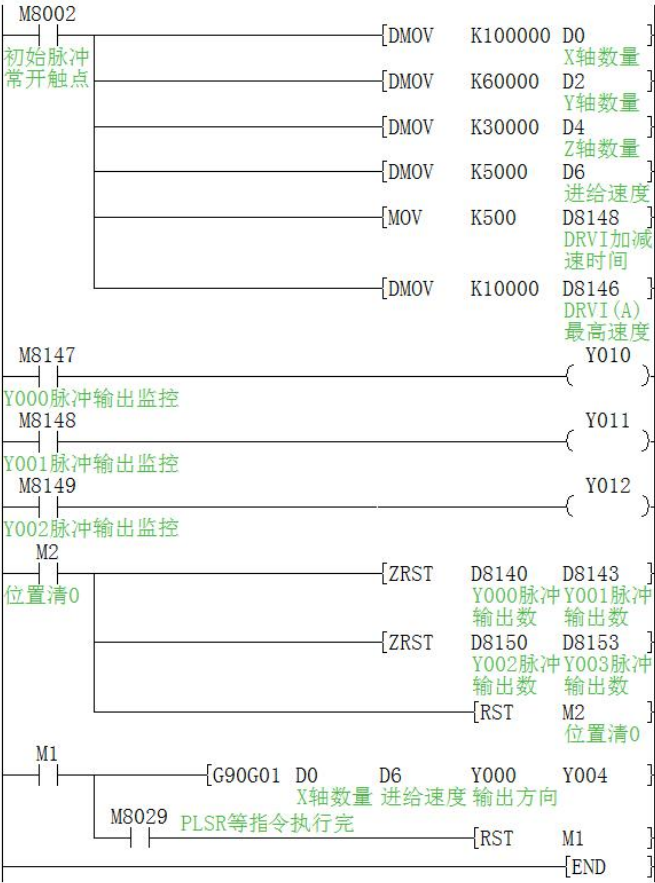
$$V_{\min} = \sqrt{\frac{\text{最高运行频率}(D8146)}{2 \times \text{加减速时间} \div 1000}}$$

4、插补输出频率范围非合成频率 10~100KHz;

5、频率计算:



编程示例：



指令执行中，涉及到的系统变量：

指令	功能	轴数	输出占用	当前位置 (双字)	加减速时间 (单字)	基底速度 (单字)	最高速度 (双字)	脉冲输出 停止	BUSY/ READY	加减速
G90G01	绝对位置 直线插补	三轴	Y0	D8140	D8148	D8104	D8145	D8146	M8145	梯形加 减速
			Y1	D8142					M8146	
			Y2	D8150					M8152	

G91G01 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
G91G01	相对位置直线插补	16	否	G91G01(S1)(S2)(D1)(D2)	17

其中:

(S1): 目标位置, 占用 6 个连续地址, 范围是 -2, 147, 483, 648 ~ 2, 147, 483, 647。

(S1): X 的相对当前位置的偏移(相对定位); (S1)+2: Y 的相对当前位置的偏移(相对定位);

(S1)+4: Z 的相对当前位置的偏移(相对定位)。

(S2): 合成的输出频率。

(D1): 高速输出端口, 只能指定 Y0/Y2, 占用连续地址(Y0, Y1, Y2, Y3)。

M8108: 设置切换直线插补的模式。 M8108=0, 为二轴直线插补模式; M8108=1, 为三轴直线插补模式。

若(D1)指定 Y0 且 M8108=0, 占用 Y0, Y1; M8108=1, 占用 Y0, Y1, Y2;

若(D1)指定 Y2 且 M8108=0, 占用 Y2, Y3; M8108=1, 则软件报错;

(D2): 脉冲方向输出端口, 占用 3 个连续地址。

操作数	位元件				字元件												
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z	
(S1)														√			
(S2)					√	√								√			
(D1)		√															
(D2)		√	√														

1、使用指令插补时, 参数设置(如加减速时间等下述参数)以 X 轴(Y0)为准。

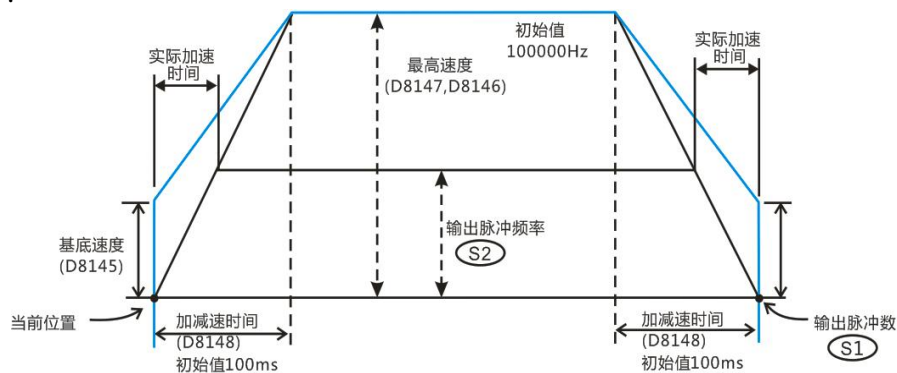
2、仅支持梯形加减速;

3、实际能够输出的合成频率 S (最低频率值) 即输出的合成频率的最低基底频率, 计算模式如下:

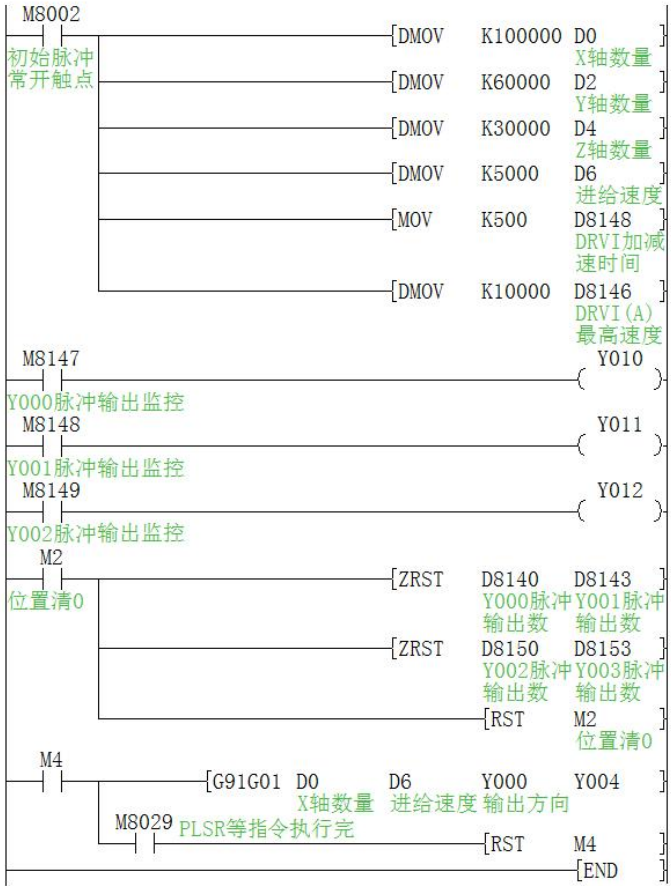
$$V_{\min} = \sqrt{\frac{\text{最高运行频率}(D8146)}{2 \times \text{加减速时间} \div 1000}}$$

4、插补输出频率范围非合成频率 10~100KHz;

5、频率计算:



编程示例：



指令执行中，涉及到的系统变量：

指令	功能	轴数	输出 占用	当前位置 (双字)	加减速时间 (单字)		基底速度 (单字)	最高速度 (双字)	脉冲输出 停止	BUSY/ READY	加减速
G91G01	相对位置 直线插补	三轴	Y0	D8140	D8148	D8104	D8145	D8146	M8145	M8147	梯形加 减速
			Y1	D8142		D8105			M8146	M8148	
			Y2	D8150		D8016			D8152	M8149	

G90G02 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
G90G02	绝对位置顺圆插补	16	否	G90G02(S1)(S2)(S3)(D1)(D2)	21

其中:

(S1): 目标位置, 占用 4 个连续地址; 范围是 -2, 147, 483, 648 ~ 2, 147, 483, 647。

(S1): X 的目标位置(绝对定位); (S1)+2: Y 的目标位置(绝对定位)。

(S2): 半径/圆, 占用 4 个连续地址。无论是相对还是绝对定位, 圆心位置都是相对值。

(S2)+0: 圆心坐标在 X 轴相对当前位置的脉冲输出数差值, 或者半径 R 的脉冲数;

(S2)+2: 圆心坐标在 Y 轴相对当前位置的脉冲输出数差值, 当使用半径时, 该值必须为 0X7FFFffff, 范围是 -2, 147, 483, 648 ~ 2, 147, 483, 647。

(S3): 合成的输出频率。

(D1): 高速输出端口, 只能指定 Y0, 占用 2 个连续地址(Y0, Y1)。

(D2): 脉冲方向输出端口, 占用 2 个连续地址。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)														√		
(S2)														√		
(S3)					√	√								√		
(D1)		√														
(D2)		√	√													

1、圆弧插补时, 弧长应大于 20 个脉冲, 否则报错;

2、圆弧插补支持的最大半径为 8000000 脉冲;

3、(S2)的设置有两种模式可选择:IJ 模式(圆心坐标模式)和 R 模式(半径模式)。当设置(S2)+2 的值为 0x7FFF FFFF, 即为 R 模式(半径模式), 否则为 IJ 模式(圆心坐标模式);

4、IJ 模式下, 无论是绝对位置插补还是相对位置插补, (S1)都只表示为圆心坐标在 XY 轴(Y0/Y1)相对当前位置的脉冲输出数差值, 都在偏移值;

5、R 模式(半径模式)下, R 的值大于 0, 表示是一个小于等于 180 度的圆弧; R 的值小于 0, 表示是一个大于 180 度的圆弧; R 模式下不能生成整圆, 因为是无穷多解。

6、(S1) 当表示目标位置的相对位置时, 需要设置合理的目标位置, 以保证能够正确的生成目标圆弧路径。当 (S1)+0=0 并且 (S1)+2=0 表示生成的是一个整圆。

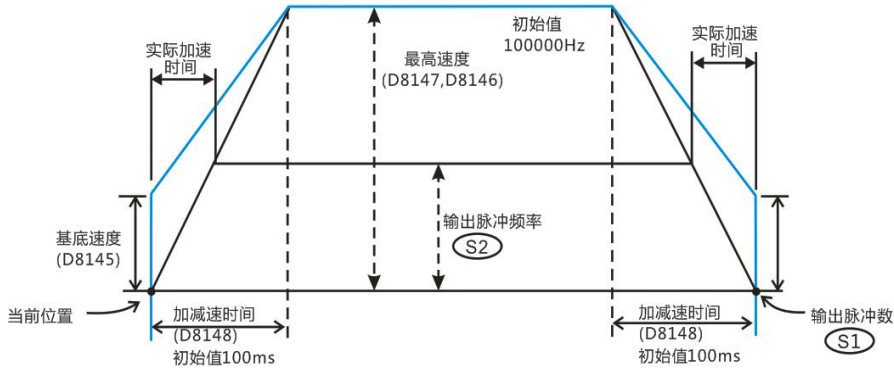
7、使用指令插补时, 参数设置(如加减速时间等下述参数)以 X 轴(Y0)为准。

8、实际能够输出的合成频率 S (最低频率值) 即输出的合成频率的最低基底频率, 计算模式如下:

$$V_{\min} = \sqrt{\frac{\text{最高运行频率}(D8146)}{2 \times \text{加减速时间} \div 1000}}$$

9、插补输出频率范围非合成频率 10~100KHz;

10、频率计算：



编程示例：

```

M8002
[DMOV K100000 D0 ]
[DMOV K60000 D2 ]
[DMOV K5000 D6 ]
[DMOV K60000 D8 ]
[DMOV H7FFFFFFF D10 ]
[MOV K500 D8148 ]
[DMOV K10000 D8146 ]

M8147
Y000脉冲输出监控
M8148
Y001脉冲输出监控
M8149
Y002脉冲输出监控

M2
位置清0
[ZRST D8140 D8143 ]
[ZRST D8150 D8153 ]
[RST M2 ]

M5
[G90G02 D0 D8 D6 Y000 Y004 ]
[M8029 PLSR等指令执行完 ]
[RST M5 ]
[END ]

```

指令执行中，涉及到的系统变量：

指令	功能	轴数	输出 占用	当前位置 (双字)	加减速时间 (单字)		基底速度 (单字)	最高速度 (双字)	脉冲输出 停止	BUSY/ READY	加减速
G90G02	绝对位置 顺圆插补	两轴	Y0	D8140	D8148	D8104	D8145	D8146	M8145	M8147	梯形加 减速
			Y1	D8142		D8105			M8146	M8148	

G91G02 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
G91G02	相对位置顺圆插补	16	否	G90G02(S1)(S2)(S3)(D1)(D2)	21

其中:

(S1): 目标位置, 占用 4 个连续地址; 范围是 -2, 147, 483, 648 ~ 2, 147, 483, 647。

(S1): X 的相对当前位置的偏移(相对定位); (S1)+2: Y 的相对当前位置的偏移(相对定位)。

(S2): 半径/圆, 占用 4 个连续地址。无论是相对还是绝对定位, 圆心位置都是相对值。

(S2)+0: 圆心坐标在 X 轴相对当前位置的脉冲输出数差值, 或者半径 R 的脉冲数;

(S2)+2: 圆心坐标在 Y 轴相对当前位置的脉冲输出数差值, 当使用半径时, 该值必须为 0X7FFFffff, 范围是 -2, 147, 483, 648 ~ 2, 147, 483, 647。

(S3): 合成的输出频率。

(D1): 高速输出端口, 只能指定 Y0, 占用 2 个连续地址(Y0, Y1)。

(D2): 脉冲方向输出端口, 占用 2 个连续地址。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)														√		
(S2)														√		
(S3)					√	√								√		
(D1)		√														
(D2)		√	√													

1、圆弧插补时, 弧长应大于 20 个脉冲, 否则报错;

2、圆弧插补支持的最大半径为 8000000 脉冲;

3、(S2)的设置有两种模式可选择:IJ 模式(圆心坐标模式)和 R 模式(半径模式)。当设置(S2)+2 的值为 0x7FFF FFFF, 即为 R 模式(半径模式), 否则为 IJ 模式(圆心坐标模式);

4、IJ 模式下, 无论是绝对位置插补还是相对位置插补, (S1)都只表示为圆心坐标在 XY 轴(Y0/Y1)相对当前位置的脉冲输出数差值, 都在偏移值;

5、R 模式(半径模式)下, R 的值大于 0, 表示是一个小于等于 180 度的圆弧; R 的值小于 0, 表示是一个大于 180 度的圆弧; R 模式下不能生成整圆, 因为是无穷多解。

6、(S1) 当表示目标位置的相对位置时, 需要设置合理的目标位置, 以保证能够正确的生成目标圆弧路径。当 (S1)+0=0 并且 (S1)+2=0 表示生成的是一个整圆。

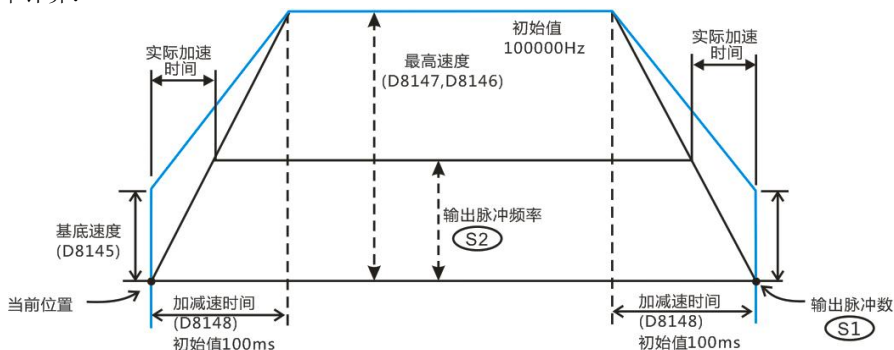
7、使用指令插补时, 参数设置(如加减速时间等下述参数)以 X 轴(Y0)为准。

8、实际能够输出的合成频率 S (最低频率值) 即输出的合成频率的最低基底频率, 计算模式如下:

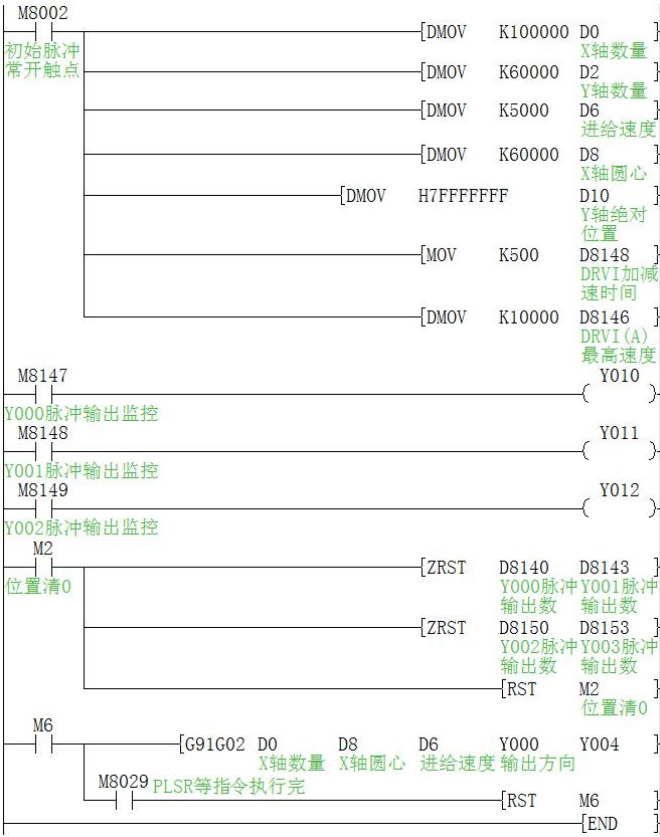
$$V_{\min} = \sqrt{\frac{\text{最高运行频率}(D8146)}{2 \times \text{加减速时间} \div 1000}}$$

9、插补输出频率范围非合成频率 10~100KHz;

10、频率计算:



编程示例：



指令执行中，涉及到的系统变量：

指令	功能	轴数	输出 占用	当前位置 (双字)	加减速时间 (单字)		基底速度 (单字)	最高速度 (双字)	脉冲输出 停止	BUSY/ READY	加减速
G91G02	相对位置 顺圆插补	两轴	Y0	D8140	D8148	D8104	D8145	D8146	M8145	M8147	梯形加 减速
			Y1	D8142		D8105			M8146	M8148	

G90G03 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
G90G03	绝对位置逆圆插补	16	否	G90G02(S1)(S2)(S3)(D1)(D2)	21

其中:

(S1): 目标位置, 占用 4 个连续地址; 范围是 -2, 147, 483, 648 ~ 2, 147, 483, 647。

(S1): X 的目标位置(绝对定位); (S1)+2: Y 的目标位置(绝对定位)。

(S2): 半径/圆, 占用 4 个连续地址。无论是相对还是绝对定位, 圆心位置都是相对值。

(S2)+0: 圆心坐标在 X 轴相对当前位置的脉冲输出数差值, 或者半径 R 的脉冲数;

(S2)+2: 圆心坐标在 Y 轴相对当前位置的脉冲输出数差值, 当使用半径时, 该值必须为 0X7FFFffff, 范围是 -2, 147, 483, 648 ~ 2, 147, 483, 647。

(S3): 合成的输出频率。

(D1): 高速输出端口, 只能指定 Y0, 占用 2 个连续地址(Y0, Y1)。

(D2): 脉冲方向输出端口, 占用 2 个连续地址。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)														√		
(S2)														√		
(S3)					√	√								√		
(D1)		√														
(D2)		√	√													

1、圆弧插补时, 弧长应大于 20 个脉冲, 否则报错;

2、圆弧插补支持的最大半径为 8000000 脉冲;

3、(S2)的设置有两种模式可选择:IJ 模式(圆心坐标模式)和 R 模式(半径模式)。当设置(S2)+2 的值为 0x7FFF FFFF, 即为 R 模式(半径模式), 否则为 IJ 模式(圆心坐标模式);

4、IJ 模式下, 无论是绝对位置插补还是相对位置插补, (S1)都只表示为圆心坐标在 XY 轴(Y0/Y1)相对当前位置的脉冲输出数差值, 都在偏移值;

5、R 模式(半径模式)下, R 的值大于 0, 表示是一个小于等于 180 度的圆弧; R 的值小于 0, 表示是一个大于 180 度的圆弧; R 模式下不能生成整圆, 因为是无穷多解。

6、(S1)当表示目标位置的相对位置时, 需要设置合理的目标位置, 以保证能够正确的生成目标圆弧路径。当 (S1)+0=0 并且 (S1)+2=0 表示生成的是一个整圆。

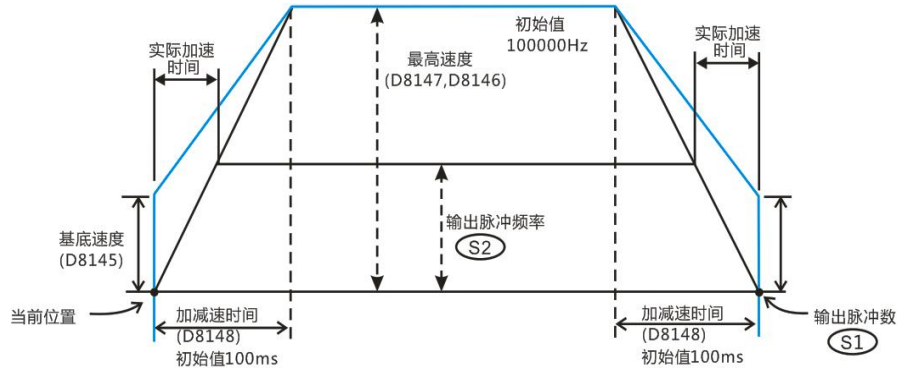
7、使用指令插补时, 参数设置(如加减速时间等下述参数)以 X 轴(Y0)为准。

8、实际能够输出的合成频率 S (最低频率值) 即输出的合成频率的最低基底频率, 计算模式如下:

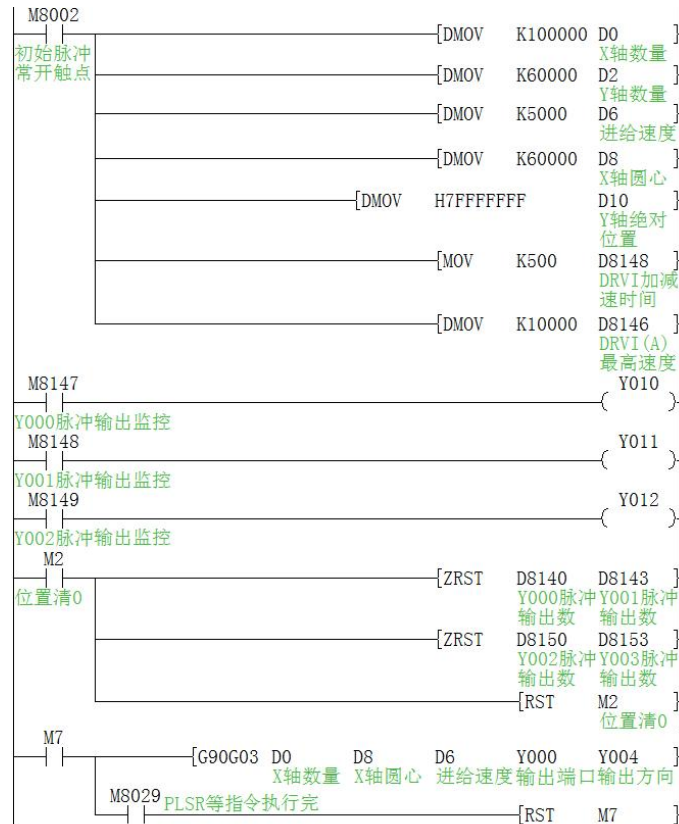
$$V_{\min} = \sqrt{\frac{\text{最高运行频率}(D8146)}{2 \times \text{加减速时间} \div 1000}}$$

9、插补输出频率范围非合成频率 10~100KHz;

10、频率计算：



编程示例：



指令执行中，涉及到的系统变量：

指令	功能	轴数	输出 占用	当前位置 (双字)	加减速时间 (单字)		基底速度 (单字)	最高速度 (双字)	脉冲输出 停止	BUSY/ READY	加减速
G90G03	绝对位置 逆圆插补	两轴	Y0	D8140	D8148	D8104	D8145	D8146	M8145	M8147	梯形加 减速
			Y1	D8142		D8105			M8146	M8148	

G91G03 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
G91G03	相对位置逆圆插补	16	否	G90G02(S1)(S2)(S3)(D1)(D2)	21

其中:

(S1): 目标位置, 占用 4 个连续地址; 范围是 -2, 147, 483, 648 ~ 2, 147, 483, 647。

(S1): X 的相对当前位置的偏移(相对定位); (S1)+2: Y 的相对当前位置的偏移(相对定位)。

(S2): 半径/圆, 占用 4 个连续地址。无论是相对还是绝对定位, 圆心位置都是相对值。

(S2)+0: 圆心坐标在 X 轴相对当前位置的脉冲输出数差值, 或者半径 R 的脉冲数;

(S2)+2: 圆心坐标在 Y 轴相对当前位置的脉冲输出数差值, 当使用半径时, 该值必须为 0X7FFFffff, 范围是 -2, 147, 483, 648 ~ 2, 147, 483, 647。

(S3): 合成的输出频率。

(D1): 高速输出端口, 只能指定 Y0, 占用 2 个连续地址(Y0, Y1)。

(D2): 脉冲方向输出端口, 占用 2 个连续地址。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)														√		
(S2)														√		
(S3)					√	√								√		
(D1)		√														
(D2)		√	√													

1、圆弧插补时, 弧长应大于 20 个脉冲, 否则报错;

2、圆弧插补支持的最大半径为 8000000 脉冲;

3、(S2)的设置有两种模式可选择:IJ 模式(圆心坐标模式)和 R 模式(半径模式)。当设置(S2)+2 的值为 0x7FFF FFFF, 即为 R 模式(半径模式), 否则为 IJ 模式(圆心坐标模式);

4、IJ 模式下, 无论是绝对位置插补还是相对位置插补, (S1)都只表示为圆心坐标在 XY 轴(Y0/Y1)相对当前位置的脉冲输出数差值, 都在偏移值;

5、R 模式(半径模式)下, R 的值大于 0, 表示是一个小于等于 180 度的圆弧; R 的值小于 0, 表示是一个大于 180 度的圆弧; R 模式下不能生成整圆, 因为是无穷多解。

6、(S1) 当表示目标位置的相对位置时, 需要设置合理的目标位置, 以保证能够正确的生成目标圆弧路径。当 (S1)+0=0 并且 (S1)+2=0 表示生成的是一个整圆。

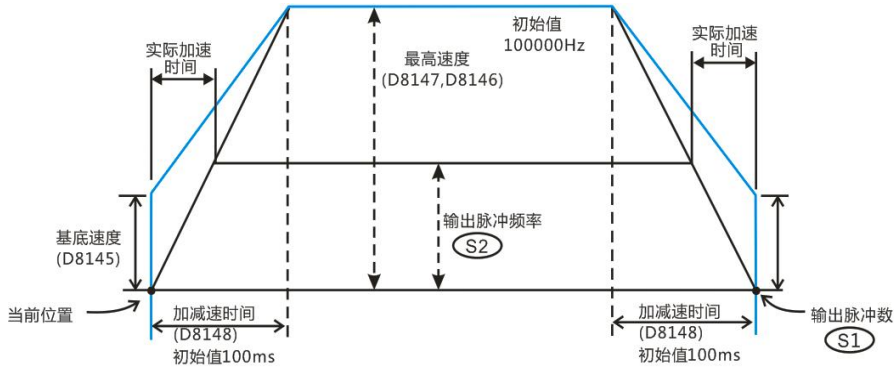
7、使用指令插补时, 参数设置(如加减速时间等下述参数)以 X 轴(Y0)为准。

8、实际能够输出的合成频率 S (最低频率值) 即输出的合成频率的最低基底频率, 计算模式如下:

$$V_{\min} = \sqrt{\frac{\text{最高运行频率}(D8146)}{2 \times \text{加减速时间} \div 1000}}$$

9、插补输出频率范围非合成频率 10~100KHz;

10、频率计算：



编程示例：

```
MS002
[DMOV K100000 D0 X轴数量]
[DMOV K60000 D2 Y轴数量]
[DMOV K5000 D6 进给速度]
[DMOV K60000 D8 X轴圆心]
[DMOV H7FFFFFFF D10 Y轴绝对位置]
[MOV K500 D8148 DRV1加减速时间]
[DMOV K10000 D8146 DRV1(A) 最高速度]

MS147
Y000脉冲输出监控 (Y010)
MS148
Y001脉冲输出监控 (Y011)
MS149
Y002脉冲输出监控 (Y012)

M2
位置清0 [ZRST D8140 D8143 Y000脉冲输出数 Y001脉冲输出数]
[ZRST D8150 D8153 Y002脉冲输出数 Y003脉冲输出数]
[RST M2 位置清0]

MS
[G91G03 D0 D8 D6 Y000 Y004 X轴数量 X轴圆心 进给速度 输出方向]
M8029 PLSR等指令执行完 [RST MS]
[END]
```

指令执行中，涉及到的系统变量：

指令	功能	轴数	输出占用	当前位置 (双字)	加减速时间 (单字)	基底速度 (单字)	最高速度 (双字)	脉冲输出 停止	BUSY/ READY	加减速
G91G03	相对位置 逆圆插补	两轴	Y0	D8140	D8148	D8145	D8146	M8145	M8147	梯形加 减速
			Y1	D8142				M8146		

5.2.12 外部设备 SER 指令

RS 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
RS	串行数据传送	16	否	RS (S) (m) (D) (n)	9

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)														√		
(m)					√	√								√		
(D)														√		
(n)					√	√								√		

该指令是一个通讯收发指令，将指定寄存器区域的数据，自动向串口依次发送，将串口接收到的数据存放到指定区域，相当于用户程序直接访问通讯缓冲区，借助用户程序对通讯收发缓冲区的处理，实现自定义协议的通讯。

COM2(通讯端口)：RS485 通信方式，支持编程口协议（即 PLC 协议）、MODBUS 协议（MODBUS-RTU 从站、MODBUS-RTU 主站、MODBUS-ASCII 主站、MODBUS-ASCII 从站）、支持 N: N 网络通信协议(目前仅 LX3VP 系列 PLC 支持)。COM2 中的通信协议由系统特殊地址 D8126 设定，串口参数由系统特殊地址 D8120 设定。具体定义如下表：

COM2 口的通信配置		
协议	D8126 数值设定	通信参数设定
HMI 监控协议（PLC 协议）	01H	由 D8120 设定
MODBUS-RTU 从站	02H	由 D8120 设定
MODBUS-ASCII 从站	03H	由 D8120 设定
RS 指令	10H	由 D8120 设定
MODBUS-RTU 主站	20H	由 D8120 设定
MODBUS-ASCII 主站	30H	由 D8120 设定

位编号	名称	内容	
		0 (off)	1 (on)
B0	数据长度	7 位	8 位
B1B2	奇偶校验	00: 无校验 01: 奇校验(ODD)	11: 偶校验 (EVEN)
B3	停止位	1 位	2 位
B4B5B6B	波特率(bps)	0111: 4800 1000: 9600 1001: 19200 1010: 38400 1011: 57600 1100: 115200	
B8	STX	无	由 D8124 设置
B9	ETX	无	由 D8125 设置

- 1、RS（自定义协议）指令说明：
- (S)为待发送数据存放的寄存器区的起始地址；

(m)为待发送数数据的长度（字节数），取值范围 0~256；

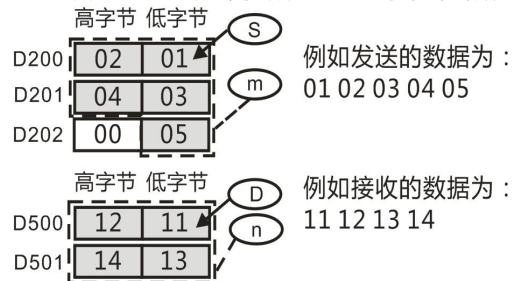
(D)为通讯接收数据的存放寄存器区的起始地址；

(n)本通讯接收的数据长度（字节数），取值范围 0~256。

编程示例:



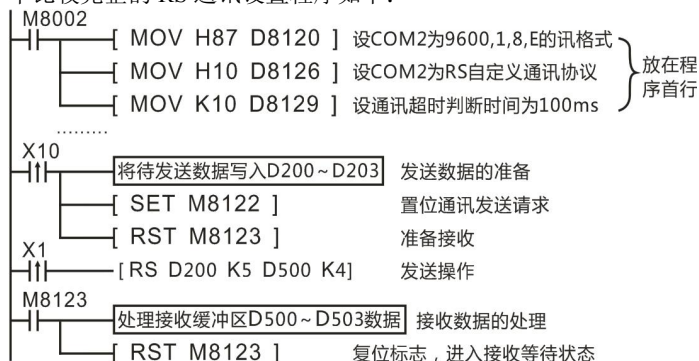
当 X1 为 ON 时，指令执行后通讯的收发数据存放如下图所示：



RS (MODBUS 模式) 指令每当完成一个发送数据、接收应答操作时，就会自动将 M8123 置位一次，利用该标志，就可以判别 RS 指令是否已执行完成。



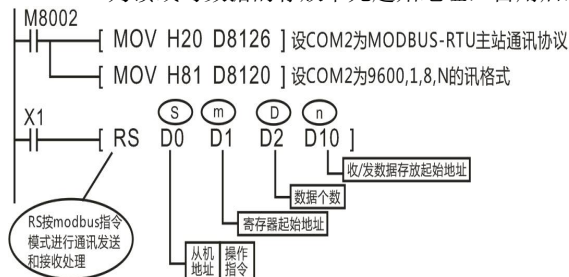
实际编程时，需要作一些串行通讯的配置和准备，如设定串口的收发模式、波特率、位数、校验位、软件协议的设定、超时判断条件、收发缓冲区的数据准备、收发标志处理等，才能按预期的要求进行通讯。仍以上述语句为例，一个比较完整的 RS 通讯设置程序如下：



2、RS (modbus 协议) 指令说明：

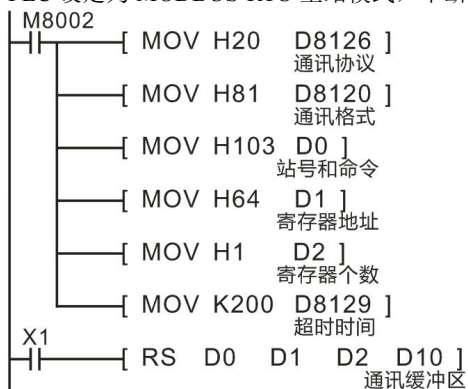
其中 RS (MODBUS 模式) 指令中的各操作数定义与标准的 RS 指令 (自定义协议) 定义不同，分别为：

- ① S 为从机地址 (高字节)、通讯命令 (低字节，按 MODBUS 协议定义)；
- ② m 为访问从站的寄存器起始地址号；
- ③ D 欲读或写的数据长度，单位为 word；
- ④ n 为读或写数据的存放单元起始地址，占用后续地址单元，长度由 ③ 决定。



编程示例:

将 PLC 设定为 MODBUS-RTU 主站模式，不断的读取从机地址为 100 的寄存器，并将读取的数据存于 D10 中。



注: 主机需从 STOP→RUN 时才有效!

执行结果: 开机后，PLC 不断读从机地址为 100 的寄存器，通过 COM2 发送以下一帧数据（16 进制）：

01 03 00 64 00 01 C5 D5

01: 代表从机地址，D0 的高 8 位；

03: MODBUS 命令码，D0 的低 8 位，意义为读从机寄存器；

00 64: 所要读取的从机寄存器地址，D1 的值；

00 01: 所要读取的寄存器个数，D2 的值，

C5 D5: CRC 校验码。

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
RS2	串行数据传送 2	16	否	RS2 (S) (m) (D) (n) (n1)	11

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)														√		
(m)					√	√								√		
(D)														√		
(n)					√	√								√		
(n1)					√	√								√		

该指令主要用于 BD 板模块的串行数据传送指令。

该指令是一个通讯收发指令，将指定寄存器区域的数据，自动向串口依次发送，将串口接收到的数据存放到指定区域，相当于用户程序直接访问通讯缓冲区，借助用户程序对通讯收发缓冲区的处理，实现通讯。其中：分为自定义协议的运用和 modbus 的运用两种。

RS2 指令中关于通讯协议的选择，是根据 CPAVL 指令进行配置，相关说明请查看“LX3V-2RS485-BD 使用手册”、“LX3V-ETH-BD 使用手册”或者“CPAVL 指令说明”。

一、LX3V-2RS485-BD 模块

1、自定义协议的运用说明

(S) 为待发送数据存放的寄存器区的起始地址；

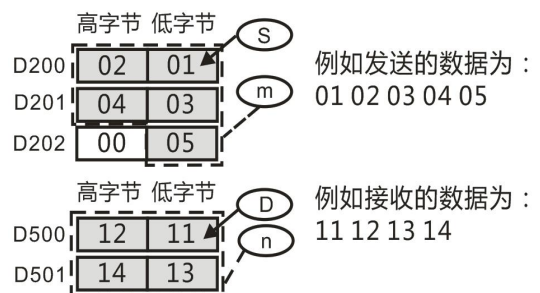
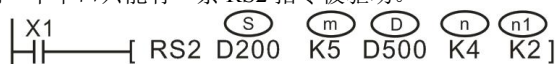
(m) 为待发送数数据的长度（字节数），取值范围 0~256；

(D) 为通讯接收数据的存放寄存器区的起始地址；

(n) 本通讯接收的数据长度（字节数），取值范围 0~256。

(n1) 为使用串口号，2-com2；3-com3；4-com4；5-com5；6-com6；用户程序可写多条 RS2 指令，但是同一时间

一个串口只能有一条 RS2 指令被驱动。



(n1) 设置为 K2：RS2 指令用于 COM2 口的通讯。

当 X1 为 ON 时，指令执行后通讯的收发数据存放如右图：

2、modbus 协议的运用说明

其中 RS（MODBUS 模式）扩展指令中的各操作数定义与标准的 RS 指令定义不同，分别为：

(S) 为从机地址（高字节）、通讯命令（低字节，按 MODBUS 协议定义）；

(m) 为访问从站的寄存器起始地址号；

(D) 欲读或写的数据长度，单位为 word；

(n) 为读或写数据的存放单元起始地址，占用后续地址单元，长度由 (D) 决定。

(n1) 为使用串口号，2-com2；3-com3；4-com4；5-com5；6-com6；用户程序可写多条 RS2 指令，但是同一时间

一个串口只能有一条 RS2 指令被驱动。

(n1) 设置为 K2: RS2 指令用于 COM2 口的通讯。当 X1 为 ON 时, 指令执行后通讯的收发数据存放如下图:



根据示例: 发送的数据为: 01 03 00 64 00 04 + CRC 校验

指令意义: 读取站号为 1 的从机中地址 100 到 103 四个地址的数据, 将读取到的数据存放在 D202 到 D205 四个地址中。

二、LX3V-ETH-BD 模块

1、MODBUS TCP 协议的运用说明

其中 RS (MODBUS 模式) 扩展指令中的各操作数定义与标准的 RS 指令定义不同, 分别为:

(S) 为从机地址 (高字节)、通讯命令 (低字节, 按 MODBUS 协议定义);

(m) 为访问从站的寄存器起始地址号;

(D) 为读或写的数据长度, 单位为 word (取值范围查看下表);

Modbus 数据长度限制标准		
功能码	十进制长度	16 进制长度
写线圈	1968	0x7B0
读线圈	2000	0x7D0
写寄存器	123	0x7B
读寄存器	125	0x7D

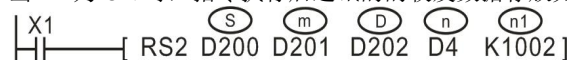
(n) 为读或写数据的存放单元起始地址, 占用后续地址单元, 长度由 (D) 决定。

(n1) 为使用的连接编号, 根据以太网口对应的连接编号来设置 (具体设置查看连接编号)。

连接编号说明					
以太网口 1		使用的连接编号	以太网口 2		使用的连接编号
RS2 指令	连接 1	1000	RS2 指令	连接 1	1100
	连接 2	1001		连接 2	1101
	连接 3	1002		连接 3	1102
	连接 4	1003		连接 4	1103
	连接 5	1004		连接 5	1104
	连接 6	1005		连接 6	1105
	连接 7	1006		连接 7	1106
	连接 8	1007		连接 8	1107

(n1) 设置为 K1002: RS2 指令用于以太网口连接 3 的设备通讯。

当 X1 为 ON 时, 指令执行后通讯的收发数据存放如下图所示:



根据示例: 发送的数据为: 01 03 00 00 00 06 01 03 00 64 00 04

指令意义: 读取站号为 1 的从机中地址 100 到 103 四个地址的数据, 将读取到的数据存放在 D202 到 D205 四个地址中。



指令格式: 发送次数(2bytes) 00 00 modbus 长度(2bytes) 站号功能码 起始地址 地址长度

00 03: 发送次数, 该次数是设备自动去累加的, 表示已发送 3 次。

00 06: modbus 指令的长度, 如后面的 01 03 00 64 00 04。

RSLIST 指令

指令描述:

名称	功能	位数	是否脉冲型	指令格式	步数
RSLIST	表格化通信指令	16	否	RSLIST(S1)(S2)(m1)	7

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)														√		
(S2)														√		
(m1)			√													

RSLIST 指令仅 LX3VP 系列及以上的 plc 才支持，且 LX3VP 系列 plc 的主版本号 and 此版本号必须是“25103”和“16001”及以上版本才支持。 LX3VE 系列 plc 的主版本号 and 此版本号必须是“25201”和“16001”及以上版本才支持。

该指令是将 RS 指令进行表格化的指令，通过 D8126（通讯协议）的更改进行更换通讯协议。该指令的大部分功能与 RS 指令相同，但简化了繁琐的工程编写，直接由表格设定各个传输命令的参数。

(S1)为表格控制的起始软元件地址，取值范围 D0~D7999； 起始软元件(S1)~(S1)+4+n×6 之间不能被其他指令使用，（尽量存放在掉电保存区域（D200-D7999）避免数据丢失）。

RSLIST 指令的通讯表格（MODBUS 协议）：

地址	名称	功能说明
(S1)+0	表头	表头；=50h，正确的 MODBUS 以及 MODBUSASC 协议通讯表格。(严禁修改)
(S1)+1	通讯命令个数	通讯命令个数；一次传输需要用六个软元件来描述，即六个软元件描述一次数据传输。(严禁修改)
(S1)+2	表头和通讯命令个数的校验位	针对表头和通讯命令个数作校验(严禁修改)
(S1)+3	错误重试次数（针对全部命令）	错误重试次数；(0-255)
(S1)+4	当前要通讯的从站站号	站号：0~255（若为 0：代表主站对所有从站作广播式发送，从站不回应信息）功能码：请查看下表。
(S1)+5	从站数据起始地址	Word 有效，定义运作从站数据起始地址。
(S1)+6	数据长度	Word 有效，范围 1~126(字数据)，1~2039(位数据)。
(S1)+7	主站储存数据起始软元件	Word 有效，定义运作主站数据起始软元件。
(S1)+8	发送控制位	发送控制位：只要非 0 都暂时不运行
(S1)+9	单独错误重试次数	单独错误重试次数（0-255）
(S1)+10	从站站号	第二数据传输描述。
(S1)+11	从站数据起始地址	
(S1)+12	数据长度	
(S1)+13	主站储存数据起始软元件	
(S1)+14	发送控制位	
(S1)+15	单独错误重试次数	
.....		
(S1)+4+n×6	保留	n 为数据传输命令总条数。

RSLIST 指令的通讯表格（自定义协议）：

地址	名称	功能说明
(S1)+0	表头	表头：=51h，正确的 自定义协议通讯表格。(严禁修改)
(S1)+1	通讯命令个数	通讯命令个数；一次传输需要用六个软元件来描述，即六个软元件描述一次数据传输。(严禁修改)
(S1)+2	表头和通讯命令个数的校验位	针对表头和通讯命令个数作校验(严禁修改)
(S1)+3	错误重试次数(针对全部命令)	错误重试次数；（0-255）
(S1)+4	主站发送数据起始软元件	Word 有效，定义运作主站接收数据起始软元件。
(S1)+5	主站数据长度	Word 有效，PLC 根据缓存区块判断大小(3VP: 0~528)
(S1)+6	主站接收数据起始软元件	Word 有效，定义运作主站发送数据起始软元件。
(S1)+7	从站数据长度	Word 有效，PLC 根据缓存区块判断大小(3VP: 0~528)
(S1)+8	发送控制位	发送控制位：只要非 0 都暂时不运行
(S1)+9	单独错误重试次数	单独错误重试次数（0-255）
(S1)+10	主站数据长度	第二数据传输描述。
(S1)+11	从站数据长度	
(S1)+12	主站发送数据起始软元件	
(S1)+13	主站接收数据起始软元件	
(S1)+14	发送控制位	
(S1)+15	单独错误重试次数	
.....		
(S1)+4+ n×6	保留	n 为数据传输命令总条数。

(S2)为表格缓存的起始软元件地址，取值范围 D0~D7999；缓存起始软元件(S2)~(S2)+12 之间不能被其他指令使用。注意：以上不能操作的位置如果随意写入数据将会导致通讯异常。

地址	High Byte	Low Byte	R/W
(S2)+0	运作序号：当前运行到第几条命令		R
(S2)+1	结果码：=0，正常； =其它值，异常		R
(S2)+2	从站设备站号	功能码	R
	自定义协议：主站发送起始软元件		R
(S2)+3	主站软元件起始地址（自定义协议：主站接收数据起始软元件）		R
(S2)+4	接收或发送的数据量（自定义协议：主站接收数据量）		R
(S2)+5	当前命令错误重复次数		R
(S2)+6	错误次数记录		R
(S2)+7	Bit0=1，Port 已被占用，本命令等待取得数据交易传输权； Bit4，通讯传输输出指示由“M1”输出表示； Bit5，通讯错误输出指示由“M1+1”输出表示； Bit6，通讯完成输出指示由“M1+2”输出表示。		R
(S2)+8	选择第几条命令执行		W
(S2)+9	选择第几条命令开启和关闭：第几条命令		W
(S2)+10	选择第几条命令开启和关闭： 0：无，1：关闭，2：开启；		W
(S2)+11	（系统占用）		R

名称	数值	功能说明
功能码	01H	读取从站连续多个单点的状态。
	03H	读取从站连续多个缓存器的数据。
	05H	对从站写入单个单点状态。
	06H	对从站写入单个缓存器数据。
	0FH	对从站写入连续多个单点状态。
	10H	对从站写入连续多个缓存器数据。
结果码 (错误码)	0x00	通讯交易成功
	0x01	帧错误
	0x02	不合法的通讯表格。(表头错误)
	0x04	数据长度错误(命令所读取或写入的位置超出软元件大小范围)
	0x05	设置的读写长度范围超出软元件范围(起始软元件加上长度后超出D0-D7999的范围)
	0x06	功能码错误(不正确的功能码或不支持此功能码)。
	0x07	从机站号错误
	0x08	从机地址错误
	0x09	从站无反应(Time-out 异常)。
	0x0A	通讯异常(接收到错误数据或从站响应错误信息)。
	0x0B	选择的命令超出最大命令数
	0x0F	跳过此条命令(该命令的发送控制位不为0)

(m1)为通讯标志的起始地址，取值范围 M0~M3068；((m1)~(m1)+2)不能被其他指令使用。

地址	功能
(m1)+0	传输标志
(m1)+1	错误标志
(m1)+2	完成标志

其他相关设置见下表：

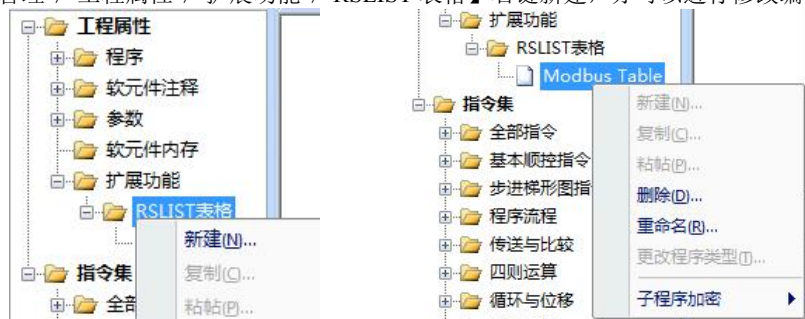
D8120 通讯参数设置			
位编号	名称	内容	
		0 (off)	1 (on)
B0	数据长度	7 位	8 位
B1B2	奇偶校验	00: 无校验 01: 奇校验(ODD) 11: 偶校验(EVEN)	
B3	停止位	1 位	2 位
B4B5B6 B7	波特率 (bps)	0111: 4800 1000: 9600 1001: 19200 1010: 38400 1011: 57600 1100: 115200	
B8	STX	无	由 D8124 设置
B9	ETX	无	由 D8125 设置

地址	说明
D8120	Com2 口通讯格式，界面配置设定(详情见上表)。
D8122	自定义协议：传送剩余数据数量(仅对 RS 指令)； MODBUS 协议：命令发送间隔，0=5ms。单位 0.1ms
D8124	起始字符 STX (仅对 RS 自定义协议)
D8125	终止字符 ETX (仅对 RS 自定义协议)
D8126	通讯协议设定，界面配置设定
D8129	MODBUS：通讯超时时间判断，界面配置设定，默认为 10(10ms) RS 自定义协议：字符间超时，界面配置设定，默认为 10(10ms)
D8172	首字符超时，界面配置设定，默认为 10 (10ms) M8172 为 0 时不计算首字符超时。(仅对 RS 自定义协议)

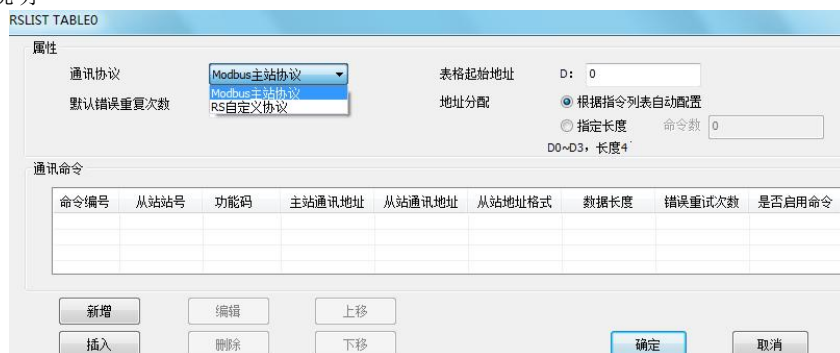
表格配置和案例分析

一、新建 RSLIST 表格

点击【工程管理->工程属性->扩展功能->RSLIST 表格】右键新建，亦可以进行修改编辑。如下图：



表格配置说明



通讯协议：应与通讯协议控制地址（D8126 等）的配置一致（modbus 主站/RS 自定义协议）。

错误重复次数：单独一条的错误重复次数不为 0 时用此条错误重复次数，为 0 时用表头设置的错误重复次数，都是 0 则默认 3 次。所设置的重复次数包括第一次运行的次数，也就是设置重复次数 2 表示运行一次后错误则重复一次再次错误则转向下一条。

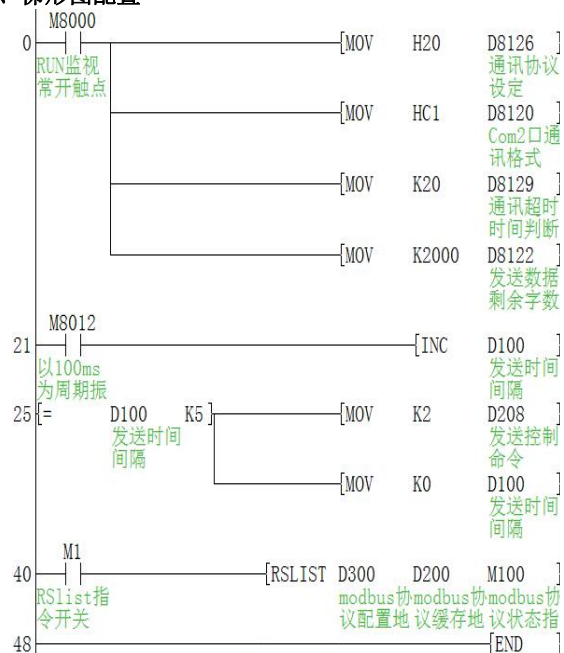
表格起始地址：应与对应的 RSLIST 指令中的(S1)一致。

地址分配：表格空间可自动配置也可选择固定长度（ $4+n*6$ ），其中 n 为通讯命令的条数。

注意表格的命令条数应小于 255 条，且做好地址规划，以免地址被重复占用，导致数据错乱。

二、Modbus 协议配置

1、梯形图配置



先设置通讯协议为“MODBUS-RTU 主站”：

D8126=H20。

设置通讯格式为“115200, 1, 8, None”：D8120=HC1。

设置通讯超时时间。

设置命令发送间隔。

设置数据发送时间间隔

D208 为 RSLIST 表格指令控制命令地址（详情查看“S2)地址分配列表”）

启动 RSLIST 的 modbus 协议表格，协议控制起始地址为 D300，缓存控制起始地址为 D200，状态指示起始地址为 M100。

3、表格配置

属性

通讯协议

Modbus主站协议

默认错误重复次数

5

表格起始地址

D: 300

地址分配

☒ 根据指令列表自动配置

☐ 指定长度

命令数 3

D300~D321, 长度22

通讯命令

命令编号	从站站号	功能码	主站通讯地址	从站通讯地址	从站地址格式	数据长度	错误重试次数	是否启用命令
1	1	01H	0	20	10进制	10	0	是
2	1	03H	500	100	10进制	20	0	否
3	1	06H	600	300	10进制	10	0	是

表格起始地址为 D300，与梯形图中的 RSLIST 指令对应。

当 M1=ON 时，RSLIST 指令开始执行。表格中的指令选择“是否启用命令”为“是”时，将循环的执行（周期执行表格中编号 1 和编号 3 的指令）。若指令命令选择为“否”时，将由“(S2)+8”=D208 控制执行。如上图的梯形图，每隔 500ms 触发一次 D208=2，即每 500ms 执行一次表格中命令编号为 2 的指令（读取从站 100 开始的 20 个地址的数据，将读取的数据分别存放在 D500-D519 中）。

4、通讯命令配置

命令编辑

命令类型: MODBUS主站协议

从站站号

1

功能码

03H 读多寄存器

主站通讯地址

D: 1020

从站通讯地址

10进制 2000

数据长度

10

当前命令地址占用

D1020~D1029, 长度10

错误重复次数

2

默认启用此命令

☐ 是

☒ 否

确定

取消

从站账号：（限制在 0-255 之间）0 账号为广播不接收。

参看“功能码列表”。

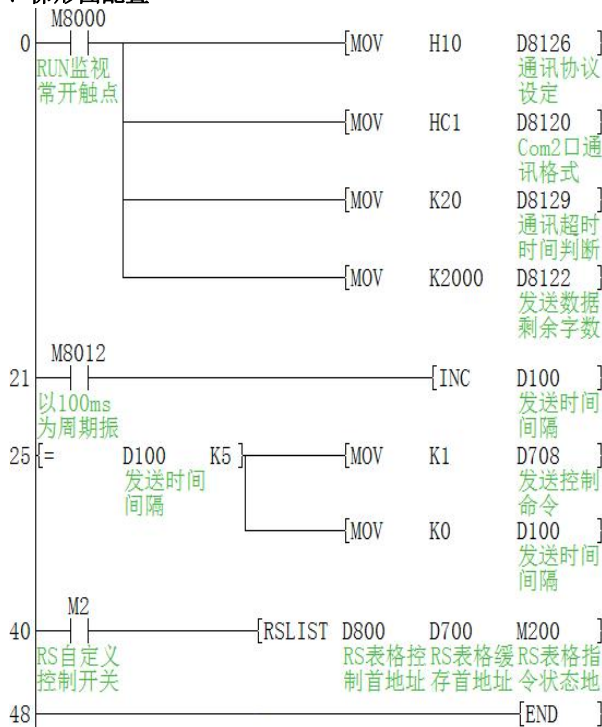
主站储存数据起始地址（D 软元件）范围（D0-D7999），如果需要保留相应数据应当设置在掉电保存（D200-D7999）里面避免数据丢失。从站存储数据的起始地址。

数据长度（位的长度以位为单位，字的长度以字为单位），范围 1~126(字数据), 1~2039(位数据)。此条命令的错误重复次数（0 表示为本条错误重复次数与表格编辑中的错误重复次数相同）。

是：RSLIST 开启时就循环执行，否：只有(S2)+8 有动作时才执行。

三、RS 自定义协议

1、梯形图配置



先设置“通讯协议为“MODBUS-RTU 主站”：
D8126=H10。

设置“通讯格式为“115200, 1, 8, None”：D8120=HC1。

设置通讯超时时间。

设置命令发送间隔。

设置数据发送时间间隔

D708 为 RSLIST 表格指令控制命令地址（详情查看”
(S2)地址分配列表“）

启动 RSLIST 的 RS 协议表格，协议控制起始地址为
D800，缓存控制起始地址为 D700，状态指示起始地
址为 M200。

2、表格配置

属性

通讯协议: RS自定义协议

默认错误重复次数: 5

表格起始地址: D: 800

地址分配: ☒ 根据指令列表自动配置 ☐ 指定长度 命令数: 2

D800~D815, 长度16

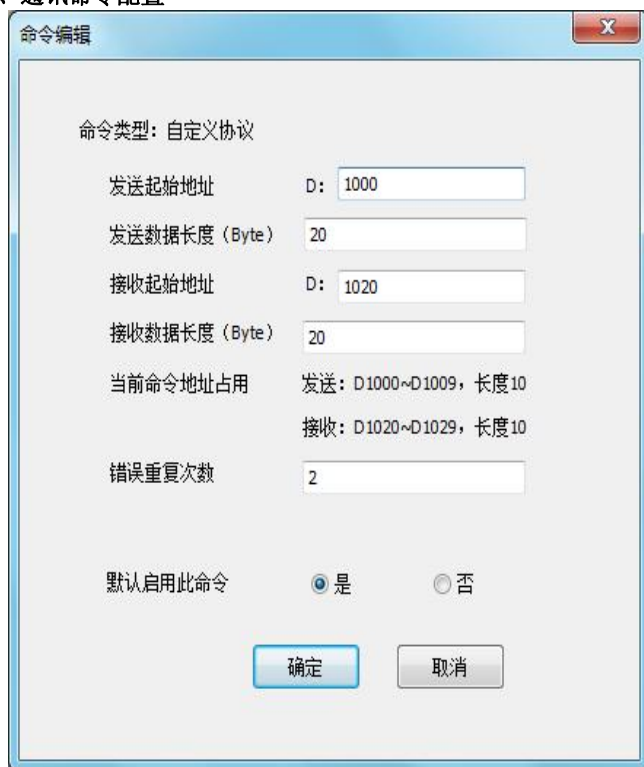
通讯命令

命令编号	发送起始地址	发送数据长度	接收起始地址	接收数据长度	错误重试次数	是否启用命令
1	1000	20	1020	20	2	否
2	0	0	1040	1	0	是

表格起始地址为 D800，与梯形图中的 RSLIST 指令对应。

当 M2=ON 时，RSLIST 指令开始执行。表格中的指令选择“是否启用命令”为“是”时，将循环的执行（周期执行表格中编号 2 的指令）。若指令命令选择为“否”时，将由“(S2)+8”=D208 控制执行。如上图的梯形图，每隔 500ms 触发一次 D708=1，即每 500ms 执行一次表格中命令编号为 1 的指令（将 D1000 开始的 20 个地址的数据发给设备，将设备返回的数据存放在 D1020 开始的 20 个地址中）。

3、通讯命令配置



命令编辑

命令类型：自定义协议

发送起始地址 D: 1000

发送数据长度 (Byte) 20

接收起始地址 D: 1020

接收数据长度 (Byte) 20

当前命令地址占用 发送: D1000~D1009, 长度10
接收: D1020~D1029, 长度10

错误重复次数 2

默认启用此命令 ☒ 是 ☐ 否

确定 取消

发送起始地址 (D 软元件) 范围 (D0-D7999), 如果需要保留相应数据应当设置在掉电保存 (D200-D7999) 里面避免数据丢失。

发送数据长度: 这里的长度以 BYTE 为单位, (范围 3VP: 0-528)。当发送数据长度为 0 时, 表示不发送。

接收起始地址 (D 软元件) 范围 (D0-D7999), 如果需要保留相应数据应当设置在掉电保存 (D200-D7999) 里面避免数据丢失。

接收数据长度: 这里的长度以 BYTE 为单位, (范围 3VP: 0-528)。当接收数据长度为 0 时, 表示不接收。

此条命令的错误重复次数 (0 表示为本条错误重复次数与表格编辑中的错误重复次数相同)。

是: RSLIST 开启时就循环执行, 否: 只有 (S2)+8 有动作时才执行。

CPAVL 指令(串口通讯)

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
CPAVL	BD 板的通讯串口参数设置	16	否	CPAVL (S) (D) (m)	7

通过本指令，配置 LX 主机上 LX3V-2RS485-BD 模块中通讯串口的参数。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)														√		
(D)			√													
(m)					√	√										

(S) 为 D 软元件的起始地址;

(D) 为 M 软元件的起始地址;

(m) 为通讯串口号，0-com0; 1-com1; 2-com2; 3-com3; 4-com4; 5-com5; 6-com6; 用户程序可写多条 RS2 指令，但是同一时间一个串口只能有一条 RS2 指令被驱动。

地址含义



通讯串口为 4 的参数在 D0, M0 起始的 20 个地址中设置。

1、modbus 协议中，通讯串口的参数含义:

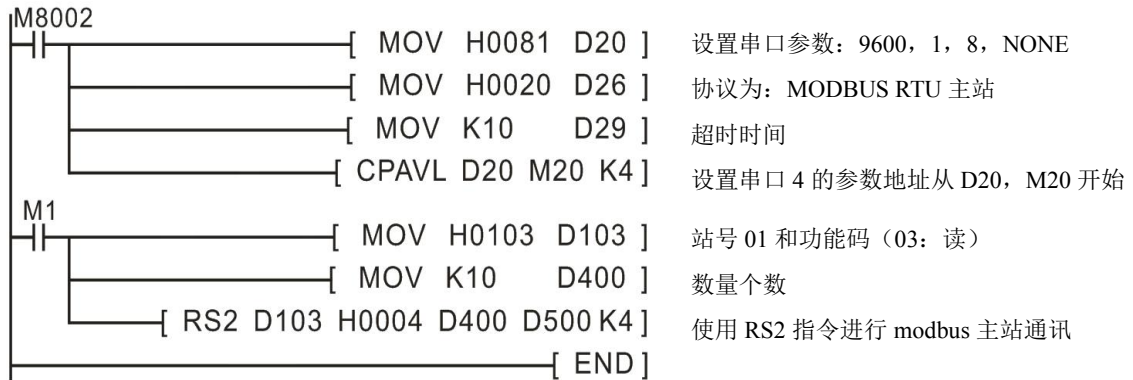
位地址	内容	字地址	内容
M0	保留	M0	通讯格式，界面配置设定，默认为 0
M1	发送等待中(RS2 指令)	M1	站号设置，界面配置设定，默认为 0
M2	发送等待中(RS2 指令) 指令执行状态 (modbus)	M2	传送剩余数据数量 (RS2 指令)，单位为 0.1ms。Modbus: 指令发送间隔，0=5ms
M3	接收完成标志 (RS2 指令) 通讯错误标志 (modbus 协议)	M3	接收到的数据数量 (RS2 指令)
M4	接收中 (仅对 RS2 指令)	M4	起始字符 STX (仅 RS2 指令)
M5	保留	M5	终止字符 ETX (仅 RS2 指令)
M6	保留	M6	通讯协议设定，界面配置设定，默认为 0
M7	保留	M7	保留
M8	保留	M8	保留
M9	超时判断标志位	M9	通讯超时时间判断，界面配置设定，默认 10 (10ms)
M10~M19	保留	M10~M19	保留

2、自定义协议中，通讯串口的参数含义:

位地址	内容	字地址	内容
(D)+0	保留	(S)+0	通讯格式，界面配置设定，默认为 0
(D)+1	发送等待中(RS2 指令)	(S)+1	站号设置，界面配置设定，默认为 0
(D)+2	发送等待中(RS2 指令)	(S)+2	传送剩余数据数量
(D)+3	接收完成标志 (RS2 指令)	(S)+3	接收到的数据数量
(D)+4	接收中 (仅对 RS2 指令)	(S)+4	起始字符 STX
(D)+5	保留	(S)+5	终止字符 ETX
(D)+6	保留	(S)+6	通讯协议设定，界面配置设定，默认为 0
(D)+7	保留	(S)+7	保留
(D)+8	保留	(S)+8	保留
(D)+9	超时判断标志位	(S)+9	通讯超时时间判断，界面配置设定，默认 10 (10ms)
(D)+10~ (D)+19	保留	(S)+10~ (S)+19	保留

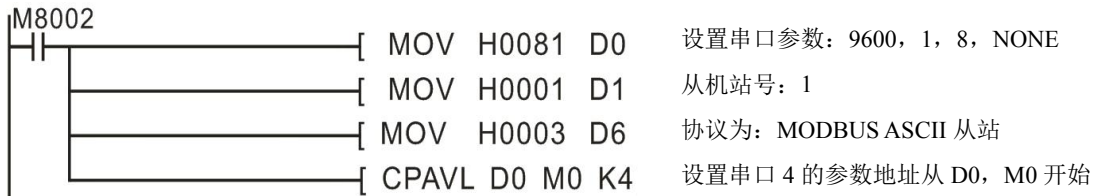
编程示例：

1、modbus 主站协议配置



通过以上设置，即可通过使用 RS2 指令进行 modbus 主站的通信。RS2 指令是一个通讯接收指令，将指定寄存器区域的数据，自动向串口依次发送，将串口接收到的数据存放到指定区域，相当于用户程序直接访问通讯缓存区，借助用户程序对通讯发缓存区的处理，实现协议的通讯。RS 指令是固定 com2 口使用，而 RS2 指令可以使用在 com3/com4/com5/com6 口。

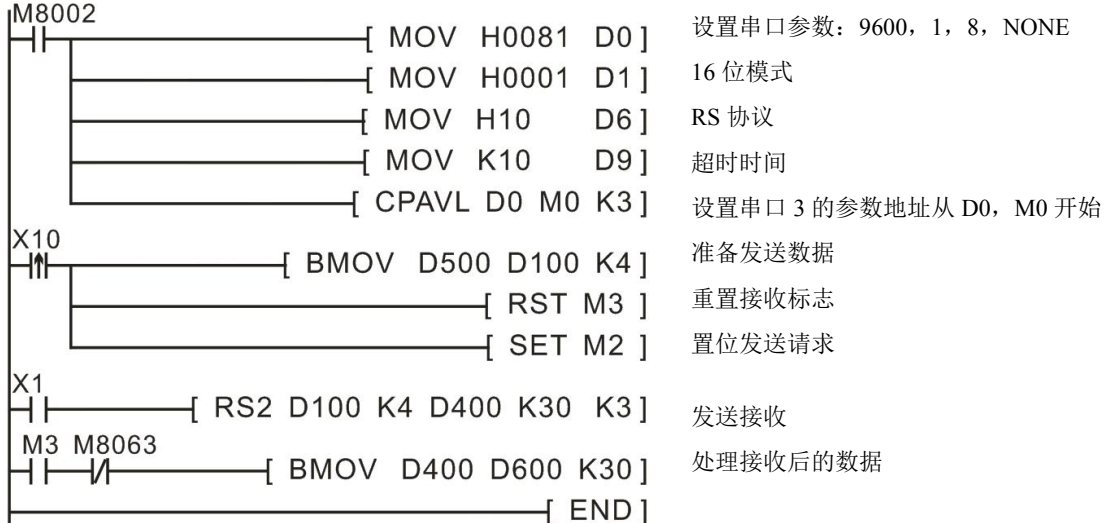
2、modbus 从站协议配置



通过以上设置，当 plc 从 stop 切换到 run 状态后，即可进行 modbus 从站通信，支持的功能码以及地址映射关系与 com2 一致。

变量名称	起始地址	寄存器数量	说明
D0~D8255	0 (0)	8256	
T0~T255	0xF000 (61440)	256	
C0~C199	0xF400 (62464)	200	
C200~C255	0xF700 (63232)	56	32 位寄存器

3、自定义协议配置



CPAVL 指令(以太网通讯)

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
CPAVL	BD 板的通讯串口参数设置	16	否	CPAVL (S) (D) (m)	7

通过本指令，配置 LX 主机上 LX3VP-ETH-BD 模块中以太网通讯接口的参数。

支持版本:

LX3VP: 25103 、16050, LX3VE: 25201、16050。

LX3VP: 25105、16000 及以上版本, LX3VE: 25203、16000 及以上版本。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)														√		
(D)			√													
(m)					√	√										

(S) 为 D 软元件参数表起始地址，地址范围为：D0-D7999（尽量存放在掉电保存区域（D200-D7999）避免数据丢失）。

(D) 为 M 软元件参数表起始地址，地址范围为：M0-M3071（尽量存放在掉电保存区域（D500-D3071）避免数据丢失）。

(m) 为使用的连接编号，根据以太网口对应连接编号来设置（具体设置查看下表）。

连接编号说明		
CPAVL 指令	端口	使用的连接编号
	以太网口 1	1000
	以太网口 2	1100

注意：指令使用时注意 CPAVL 指令只需要一条可以配置多个连接。RS 指令需要对应连接来进行发送。

地址含义

M 字地址含义表

M 位地址	说明 M	设备号	M 位地址	说明 M	设备号
(D)+0	保留	连接 1 的配置	(D)+10	保留	连接 2 的配置
(D)+1	指令执行中		(D)+11	指令执行中	
(D)+2	指令执行状态		(D)+12	指令执行状态	
(D)+3	通讯错误标志		(D)+13	通讯错误标志	
(D)+4	保留		(D)+14	保留	
(D)+5	保留		(D)+15	保留	
(D)+6	保留		(D)+16	保留	
(D)+7	保留		(D)+17	保留	
(D)+8	保留		(D)+18	保留	
(D)+9	超时标志		(D)+19	超时标志	
			(D)+20		
					

D 字地址含义表

D 字地址	说明 D	详细说明	其他说明	读写特性	D 字地址	说明 D	详细说明	其他说明	读写特性
(S)+0		版本号	BD 板参数设置	R	(S)+23	协议	通讯协议	连接 1 的配置	R/W
(S)+1	BD 板 IP 地址	IP 第 1 节		R/W	(S)+24	从站 IP	IP 第 1 节		R/W
(S)+2		IP 第 2 节		R/W	(S)+25		IP 第 2 节		R/W
(S)+3		IP 第 3 节		R/W	(S)+26		IP 第 3 节		R/W
(S)+4		IP 第 4 节		R/W	(S)+27		IP 第 4 节		R/W
(S)+5	端口	0, 默认 K502		R/W	(S)+28	端口	0 默认 K502		R/W
(S)+6	网关	网关第 1 节		R/W	(S)+29	保留			R/W
(S)+7		网关第 2 节		R/W	(S)+30	指令发送间隔	用于设置指令发送间隔单位 0.1ms		R/W
(S)+8		网关第 3 节		R/W	(S)+31	保留			R/W
(S)+9		网关第 4 节		R/W	(S)+32	保留			R/W
(S)+10	子网掩码	子网掩码第 1 节		R/W	(S)+33	保留			R/W
(S)+11		子网掩码第 2 节		R/W	(S)+34	超时			R/W
(S)+12		子网掩码第 3 节		R/W	(S)+35	协议	通讯协议	连接 2 的配置	R/W
(S)+13		子网掩码第 4 节		R/W	(S)+36	从站 IP	IP 第 1 节		R/W
(S)+14	MAC	MAC 第 1 节		R	(S)+37		IP 第 2 节		R/W
(S)+15		MAC 第 2 节		R	(S)+38		IP 第 3 节		R/W
(S)+16		MAC 第 3 节		R	(S)+39		IP 第 4 节		R/W
(S)+17		MAC 第 4 节		R	(S)+40	端口	0 默认 K502		R/W
(S)+18		MAC 第 5 节		R	(S)+41	保留			R/W
(S)+19		MAC 第 6 节		R	(S)+42	指令发送间隔			R/W
(S)+20	保留			R/W	(S)+43	保留			R/W
(S)+21	保留			R/W	(S)+44	保留			R/W
(S)+22	连接数量	设置模块连接多少个设备（最大 8 个）		R/W	(S)+45	保留			R/W
				R/W	(S)+46	超时			R/W
				R/W	(S)+47				R/W

编程示例：

M8002

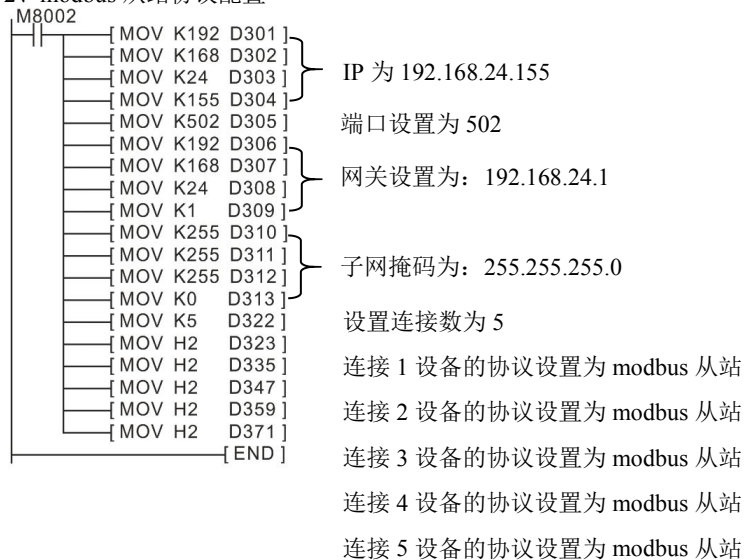

使用 LX3V-ETH-BD 模块第 1 个以太网口，以太网口的参数地址由 D300，M300 开始

1、modbus 主站协议配置

LX3V-ETH-BD 模块的以太网参数设置如下：



2、modbus 从站协议配置



PRUN 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
PRUN	8 进制位传送	16	否	PRUN (S) (D)	5
PRUNP		16	是		5
DPRUN		32	否		9
DPRUNP		32	是		9

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)								√		√						
(D)									√	√						

该指令是将(S)起始连续地址的位变量（以 8 进制为宽度单位），成批复制到(D)起始位变量组中。其中：

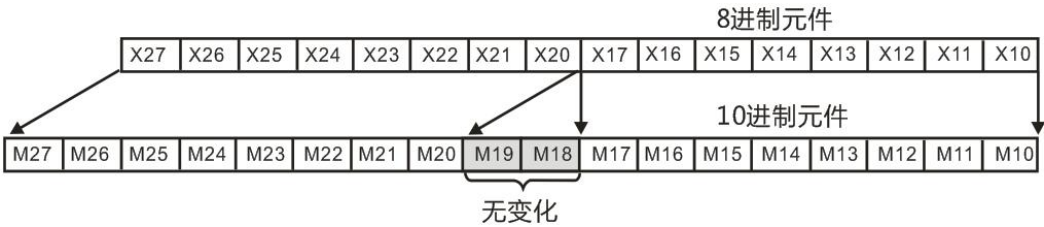
(S)为待复制的位变量的起始地址，要求地址的个位必需为 0，如 X10，M20 等；

(D)为复制的目的位变量起始地址，同样要求地址的个位必需为 0，如 M30，Y10 等。

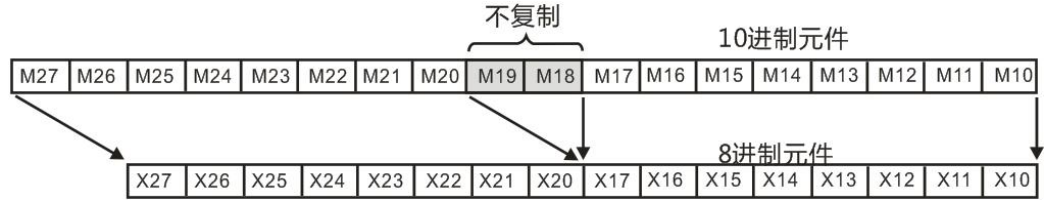
其中 Kn 中，允许 n=1~8。

编程示例:

指令示例 1:



指令示例 2:



ASCII 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
ASCII	十六进制转换	16	否	ASCII (S) (D) (n)	7
ASCIP	ASCII	16	是		7

该指令是将(S)的值转换成 ASCII 码后, 存储到(D)为起始地址的变量中。其中:

(S)为待转换的变量地址或常数数值。

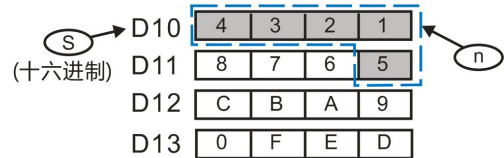
(D)为转换后 ASCII 码的存放起始地址。

(n)为转换的字符位数。

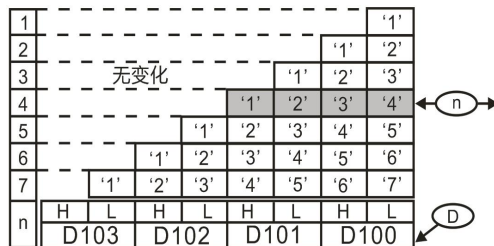
操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)					√	√		√	√	√	√	√	√	√	√	√
(D)									√	√	√	√	√	√		
(n)	常数, n=1~256															

ASCII 数值转换遵照 ASCII 与 HEX 进制数值对照表, 如: ASCII '0' 对应 HEX 'H30'; ASCII 'F' 对应 HEX 'H46' 等。关于 HEX 和 ASCII 的对照关系请参考 ASC 指令后面的附录。

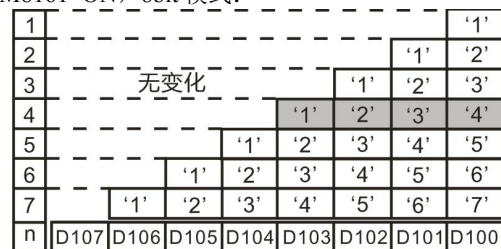
编程示例:



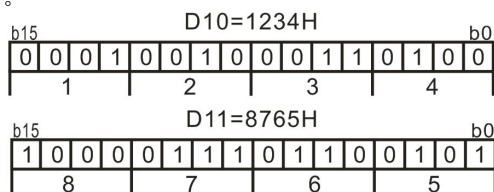
M8161=OFF, 16bit 模式:



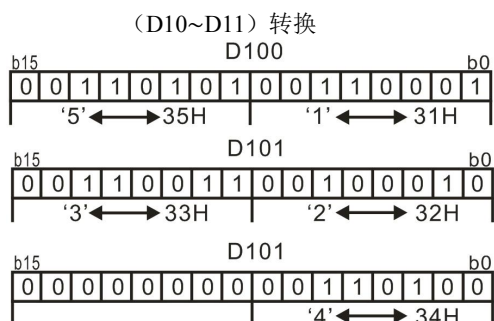
M8161=ON, 8bit 模式:



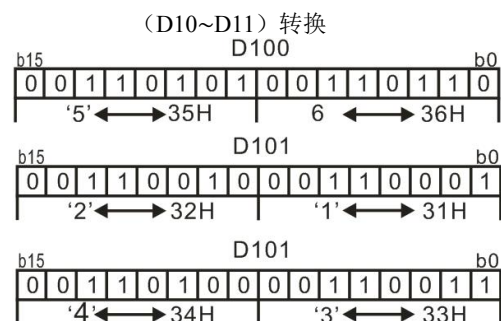
其中, M8161 标志决定了计算结果存放目的变量的宽度模式, 当 M8161=OFF 时, 为 16bit 模式, 即变量的高字节和低字节分别存储; 当 M8161=ON 时, 为 8bit 模式, 只有变量的低字节存储结果, 因此实际使用变量区域的长度增加。



当 M8161=OFF、n=5 时, 位的组成

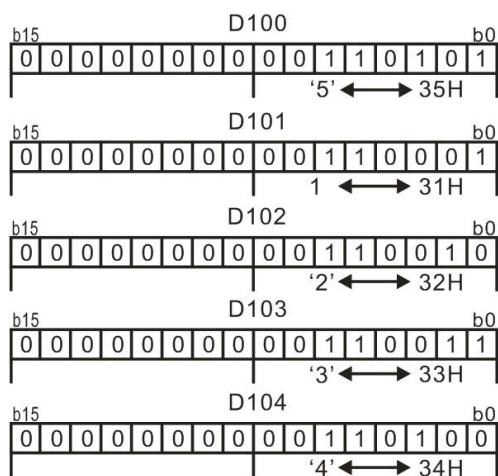


当 M8161=OFF、n=6 时, 位的组成



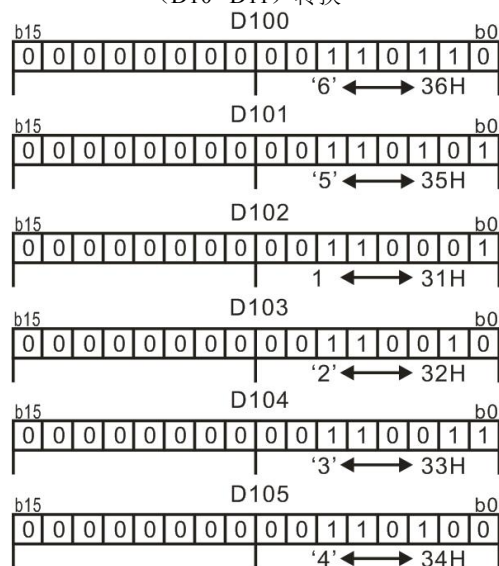
当 M8161=ON、n=5 时，位的组成

(D10~D11) 转换



当 M8161=ON、n=6 时，位的组成

(D10~D11) 转换



注：RS/ HEX/ ASCII/ CCD 等指令共用 M8161 模式标志，编程时注意。

HEX 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
HEX	HEX 的转换	16	否	HEX (S) (D) (n)	7
HEXP		16	是		7

该指令是将(S)起始变量的值(ASCII 码)转换成 HEX 数据后,存储到(D)为起始地址的变量中,转换的字符数、存储模式可以设定。其中:

(S)为待转换的变量地址或常数数值,若为寄存器变量,以 32bit 变量宽度(即 4 个 ASCII 字符)为单位进行转换分隔;

(D)为转换后 ASCII 码的存放起始地址,占用的变量空间与(n)有关;

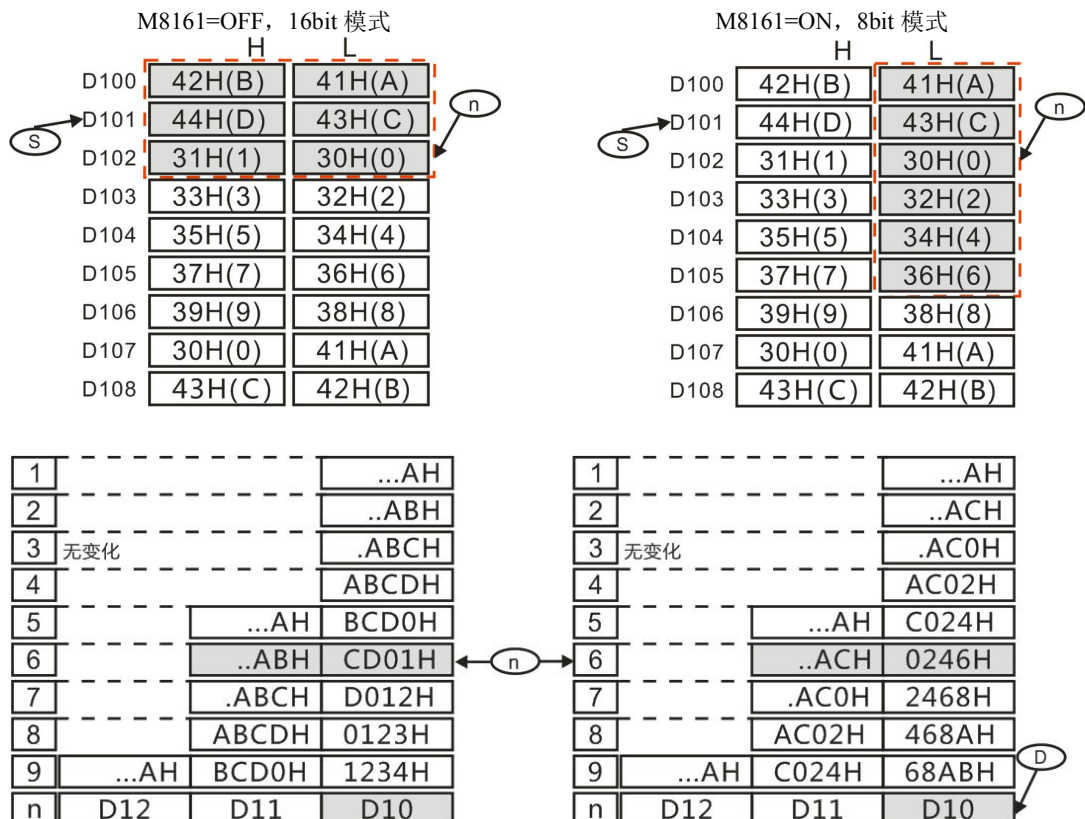
(n)为转换的 ASCII 字符位数。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)					√	√		√	√	√	√	√	√	√	√	√
(D)									√	√	√	√	√	√	√	√
(n)	常数, n=1~256															

编程示例:



例如 D100 起始的数据如下:



其中, M8161 标志决定了变量宽度模式,当 M8161=OFF 时,为 16bit 模式,即变量的高字节和低字节都参与运算;当 M8161=ON 时,为 8bit 模式,只有变量的低字节参与运算,高字节的内容丢弃,因此实际使用变量区域(S)的长度增加。

当 M8161=OFF、n=5 时，位的组成

用了 D100~D102(高低字节)转换

D10															
1	0	1	1	1	1	0	0	1	1	0	1	0	0	0	0
B				C				D				0			
D11															
0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0
								A							

当 M8161=ON、n=5 时，位的组成

用了 D100~D104(高低字节)转换

D10															
1	1	0	0	0	0	0	0	0	0	1	0	0	1	0	0
C				0				2				4			
D11															
0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
								A							

注意事项：

- 1、RS/ HEX/ ASCII/ CCD 等指令共用 M8161 模式标志，编程时注意；
- 2、(S)数据区的源数据必需为 ASCII 码字符，否则转换出错；
- 3、若输出的数据为 BCD 格式，HEX 转换后，需要进行 BCD—BIN 转换，才是正确的数值。

当 M8161=OFF、n=6，位的组成

用了 D100~D102(高低字节)转换

D10															
1	1	0	0	1	1	0	1	0	0	0	0	0	0	0	1
C				D				0				1			
D11															
0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	1
								A				B			

当 M8161=ON、n=6，位的组成

用了 D100~D105(高低字节)转换

D10															
0	0	0	0	0	0	1	0	0	1	0	0	0	1	1	0
0				2				4				6			
D11															
0	0	0	0	0	0	0	0	1	0	1	0	1	1	0	0
						A						C			

CCD 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
CCD	校验码	16	否	CCD (S) (D) (n)	7
CCDP		16	是		7

该指令是对(S)起始的(n)个变量进行两种校验和运算，将直接加法的求和运算结果存于(D)，将逐个异或逻辑运算的结果存于(D)+1单元中。本指令用于作通信时，为了确保数据传输时的正确性所做的字符串总和检查(SumCheck)。其中：(S)为待求校验和运算的变量起始地址，使用后续地址的变量单元；(D)用于存放校验和；(D)+1单元用于存放逐项异或逻辑运算结果；(n)为用于校验的变量字节数。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)								√	√	√	√	√	√	√		
(D)									√	√	√	√	√	√		
(n)	常数, n=1~256															

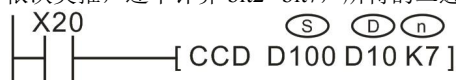
编程示例:

M8161 标志决定了变量宽度模式，当 M8161=OFF 时，为 16bit 模式，即变量的高字节和低字节都参与运算；当 M8161=ON 时，为 8bit 模式，只有变量的低字节参与运算，高字节的内容被丢弃，因此实际使用变量区域的长度增加，见下例图：

“累加和”就是将指定的 n 个变量直接相加的计算结果；

“异或”逻辑运算的方法则是：将参与运算的变量都换算为二进制数；

先统计每个变量的 bit0 为 1 的个数，若为偶数个，则异或结果的 bit0 为 0；若为奇数个，则异或结果的 bit0 为 1；再统计每个变量的 bit1 为 1 的个数，若为偶数个，则异或结果的 bit1 为 0；若为奇数个，则异或结果的 bit1 为 1；依次类推，逐个计算 bit2~bit7，所得的二进制数换算为 HEX 数值即为异或结果（或称极性值）。



例如 D100 起始的数据如下：

M8161=OFF, 16bit 模式:

	H	L
D100	12H=00010010	01H=00000001
D101	44H=01000100	A3H=10100011
D102	21H=00100001	3FH=00111111
D103	33H=00110011	D2H=11010010
D104	65H=01100101	A1H=10100001
D105	37H=00110111	C6H=11000110
D106	A9H=10101001	02H=00000010

累加和 :D10 22CH

异或(极性) D11 00111000

M8161=ON, 8bit 模式:

	H	L
D100	12H=00010010	01H=00000001
D101	44H=01000100	A3H=10100011
D102	21H=00100001	3FH=00111111
D103	33H=00110011	D2H=11010010
D104	65H=01100101	A1H=10100001
D105	37H=00110111	C6H=11000110
D106	A9H=10101001	02H=00000010

累加和 :D10 31EH

异或(极性) D11 00101010

RS/ HEX/ ASCII/ CCD 等指令共用 M8161 模式标志，编程时注意该标志的处理。

CCPID 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
CCPID	PID 运算	16	否	CCPID (S1)(S2)(S3)(D)	9

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)														√		
(S2)														√		
(S3)														√		
(D)														√		

该指令仅 CC3V 系列 PLC 支持，其中：

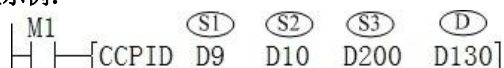
(S1)为 PID 控制的目标值 (SV)；

(S2)为实测的反馈值 (PV)；

(S3)为 PID 运算所需设定参数、中间结果保存的缓存区起始地址，占用随后地址的共 52 个变量单元（建议保留 100 个连续空间），取值范围是 D0~D7975，最好指定停电保持区，在电源 OFF 后仍保持设定值，否则首次启动运算前需对缓存区作赋值处理。其中每个单元的功能和参数描述详见本节说明；

(D)为 PID 计算结果的存放单元 (MV)，请指定(D)为非电池保持区，否则首次启动运算前需对其作初始化清 0 处理。

编程示例:



其中参数说明如下:

D9 中存放的是 PID 调节的目标值，D10 为闭环反馈值，注意 D9、D10 必需为同一量纲，如都为 0.01MPa 单位，或都为 1℃单位等等；

D200~D224 共 52 个单元用于存放 PID 运算的设定值与过程值，这些值在首次 PID 计算前就必需逐项进行设定；

D130 单元则用于存放计算后的控制输出值，用于控制执行的动作。

(S3)起始的各单元参数值的功能和设定方法说明如下表：

(S3)~(S3)+12，可设置的参数范围（CCPID 执行时设置的参数）。其中(S3)+13~(S3)+21 为 CCPID 控制内部使用的空间。而(S3)+22~(S3)+51 则是自整定过程使用的参数空间。

单元	功能	设定说明	补充说明
(S3)	采样时间 (TS)	设定范围 1~32767 (ms)，但大于 PLC 程序扫描周期	为本指令每隔多少时间计算一次，并更新输出值 (MV)。Ts 小于一次扫描时间时，PID 指令以一次扫描时间来执行，报警 6740。当 Ts 小于 0 时，报警 6730，不执行
(S3)+1	动作方向 (ACT)	bit0: 0=正动作；1=逆动作 bit2: 自整定过渡区开关 0=不打开 1= 打开 bit3: 0=单向；1= 双向 bit4: 0=自整定不动作；1=执行自整定 【bit6: 0=两段式自整定不动作 1=执行两段式自整定 (bit4 必须置 1) bit7: 0=三段式自整定不动作 1=执行三段式自整定 (bit4 必须置 1)】 其它的不可使用	bit0: 正动作：类加热系统，当温度低于设定值输出加大； 逆动作：类冷却系统，当温度大于设定值输出加大。 bit2: 自整定过渡区开关，打开后有在 1.5℃的过渡区大小。 bit3: 双向表示根据输出正负值给加热系统或冷却系统。实现一个 PID 控制两个外部系统。 bit4: a.当 bit4=1，而 bit6、bit7 都不为 1 时，自整定操作不执行。B.而当 bit4=0，而 bit6、bit7 其中一个为 1 时，不执行自整定操作。 C.当 bit4=1，而 bit6、bit7 其中都为 1 时，执行三段式自整定

(S3)+2	滤波系数	反馈量的一阶惯性滤波 (0~100%)，取值范围 [0, 100]	当取值大于等于 100 时，将以 0 执行，即不执行滤波；（未处理）
(S3)+3	比例增益(Kp)	设定范围: [1, 30000][%]	超限报错 6733
(S3)+4	积分增益(Ti)	Ti 为积分时间，取值范围 [0, 3600]s	超限报错 6734
(S3)+5	微分增益(Td)	Td 为微分时间，取值范围 [0, 1000]s	超限报错 6735
(S3)+6	工作区间	PID 启用的工作温度设置 (0 表示不作用)，取值范围 [0, 1000]	建议大于 5℃，即 50（精度 0.1℃）；超限将取边界值。
(S3)+7	输出下限	取值范围: [-10000, 10000]; 推荐设定范围 -2000 或 0 (S3+1 bit3=0 时，下限=0; bit3=1 时，下限=-2000)	1. 自整定初始化; 2. ①单向控制时: 下限取值为 0; 3. ②双向控制时: 下限>0 时，调整为下限=0; 上限=下限=0 时，默认调整为下限=-2000。说明: 设定-2000，则输出值 (MV) 小于-2000 时将以-2000 输出。 4. 控制过程中，下限动态可调，若下限>=上限时，报错 6736。
(S3)+8	输出上限	取值范围: (-10000, 10000); 推荐设定值 2000	1. 自整定初始化; 2. ①单向控制时: 上限<0 时，默认调整为上限=2000; 3. ②双向控制时: 上限<0 时，调整为上限=0; 上限=下限=0 时，默认调整上限=2000。说明: 设定 2000，则输出值 (MV) 大于 2000 时将以 2000 输出。 4. 控制过程中，上限动态可调，若下限>=上限时，报错 6736。
(S3)+9	模式设置	0: 允许超调 1: 小超调或不超调 2: 动态设置	0: 允许超调 (ukd = 100) 1: 小超调或不超调模式 (ukd = 300)
(S3)+10	比例系数 (ukp)	通常设定值 100 (默认 100) 【S3+9 设置成 2 时启用】；取值范围[1, 300]	当该值小于等于 0 或大于 500 时，超限将取边界值。
(S3)+11	积分系数 (uki)	通常设定值 50 (默认 50) 【S3+9 设置成 2 时启用】；取值范围[0, 300]	当该值小于 0 或大于 300 时，超限将取边界值。
(S3)+12	微分系数 (ukd)	通常设定值 50 (默认 100, 小超调要求时可设置 300~400) 【S3+9 设置成 2 时启用】；取值范围[0, 500]	当该值小于 0 或大于 500 时，超限将取边界值。
(S3)+13	保留内部控制使用	内部控制占用空间	
⋮			
(S3)+21			
(S3)+22	自整定使用空间	新自整定内部使用的空间	
⋮			
(S3)+53			

①自整定过程占用 S3+22~S3+51 的空间，当自整定成功后，将会把整定后的参数写入 S3+2~S3+21 的空间内。

②(S3)+2 滤波系数 α ：（采用一阶惯性滤波处理）

$$\text{公式: } T_{\text{now}} = (100 - \alpha) \times T_a + \alpha \times T_{\text{old}}$$

其中， T_a 为当前实测的温度； T_{old} 为前一次参与 PID 计算的温度； T_{now} 当前 PID 运算用的温度。 α 为滤波系数（当 $\alpha=0$ 时，不进行滤波，取值范围 $\alpha \in [0, 100)$ ）；

（如果存在超调很小的温度但稳定时间很长时，可设置该参数 80，具体问题具体分析）。

③(S3)+6 工作区间: T_{work} （比如 170，代表 17℃）

$$OUT = \begin{cases} 100\% \text{功率输出} & PV < SV - T_{work} \\ PidOut & PV \geq SV - T_{work} \end{cases}$$

正动作:

$$OUT = \begin{cases} 100\% \text{功率输出} & PV < SV + T_{work} \\ PidOut & PV \geq SV + T_{work} \end{cases}$$

逆动作:

④(S3)+9 工作模式:

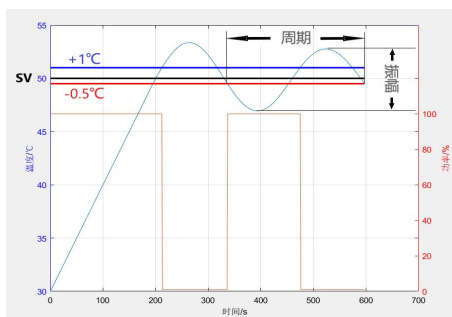
0: 允许超调的工作模式

1: 小超调或不超调工作模式

2: 自定义设置; 通过设置(S3)+10, (S3)+11, (S3)+12 三个系数来实现。

⑤(S3)+1 bit2 自整定过渡区开关: (上限 1°C, 下限 0.5°C)

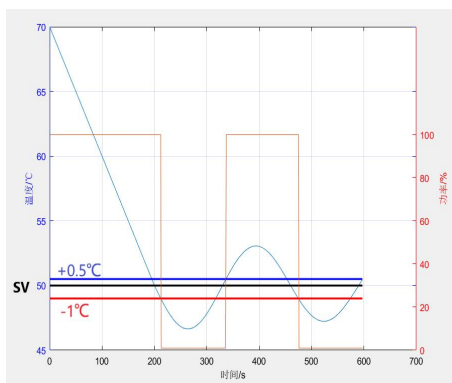
正向控制时, 过渡区说明:



在升温过程中, 当 $PV \leq SV + 1^\circ\text{C}$, 100%功率输出; 当 $PV > SV + 1^\circ\text{C}$, 不输出。

在降温过程中, 当 $PV < SV - 0.5^\circ\text{C}$, 100%功率输出。当 $PV \geq SV - 0.5^\circ\text{C}$, 不输出。

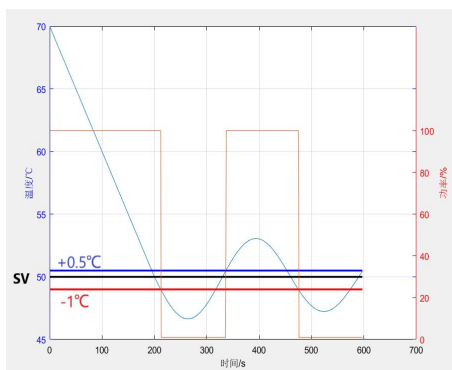
反向控制时, 过渡区说明:



在降温过程中, 当 $PV \geq SV - 1^\circ\text{C}$, 100%功率输出; 当 $PV < SV - 1^\circ\text{C}$, 不输出。

在升温过程中, 当 $PV > SV + 0.5^\circ\text{C}$, 100%功率输出。当 $PV \leq SV + 0.5^\circ\text{C}$, 不输出。

双向控制时, 过渡区说明:



在升温过程中, 当 $PV \leq SV + 1^\circ\text{C}$, 100%功率加热输出; 当 $PV > SV + 1^\circ\text{C}$, 100%功率冷却输出。

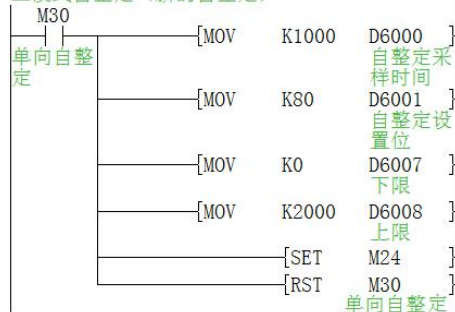
在降温过程中, 当 $PV < SV - 0.5^\circ\text{C}$, 100%功率加热输出。当 $PV \geq SV - 0.5^\circ\text{C}$, 100%功率冷却输出输出。

编程案例

1、CCPID 运用配置

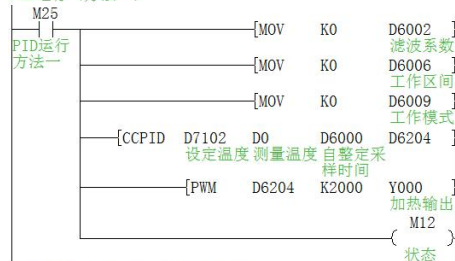
①、参数设定

三段式自整定（新的自整定）

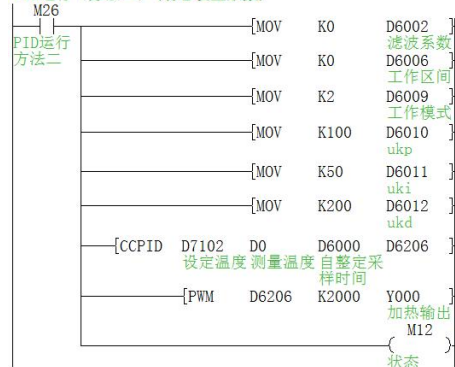


②、CCPID 控制过程设置

PID运行（方法一）

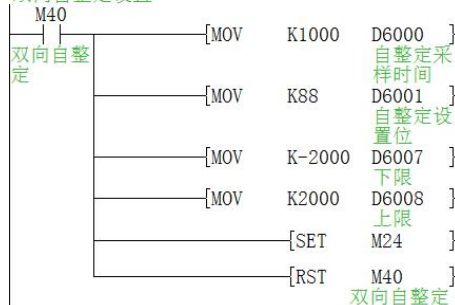


PID运行（方法二）（动态设置系数）



③、双向控制

双向自整定设置



注意：

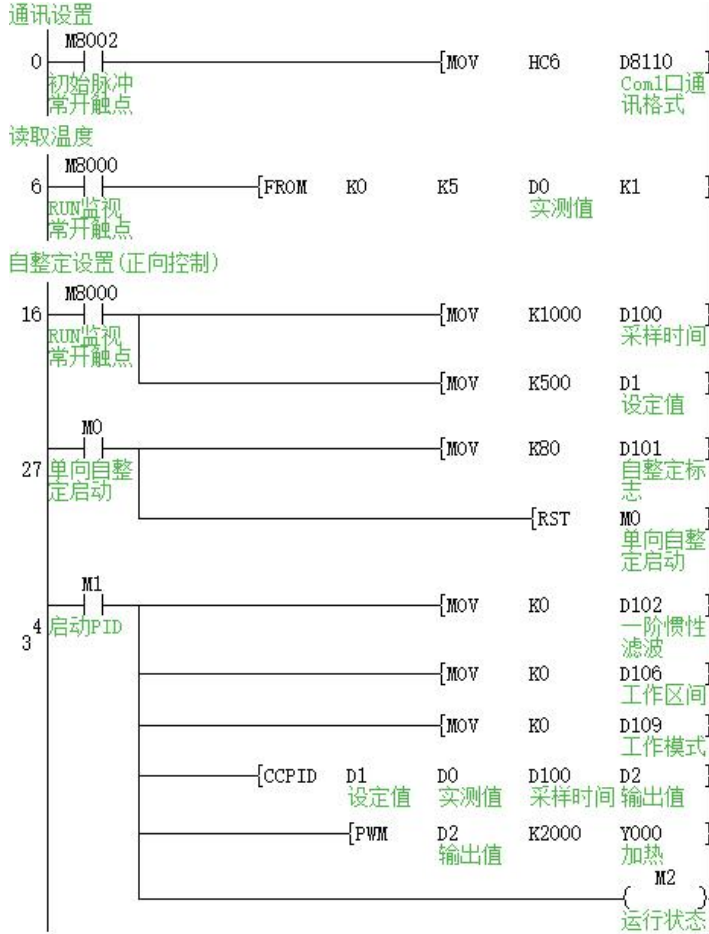
- 1、CCPID 是运算控制的专用指令，于采样时间到达后的该次扫描才执行 CCPID 运算动作。
- 2、CCPID 指令的使用次数没有限制，但是(S3)~(S3)+51 不可以重复。
- 3、CCPID 指令执行前，需要把 CCPID 参数设定完成。

案例分析

1、控制要求

本范例的控制环境为烧水壶，配置以 PLC-3V2416 主机搭配 4PT 模块控制，采用 PI8070 屏进行数据存储及过程曲线查看。

2、示例程序



3、参数说明

PLC 装置	控制说明
M0	设置自整定
M1	CCPID 指令运算启动
M2	CCPID 运行状态
Y0	可调变脉冲宽度的脉冲输出
D0	温度实测值
D1	温度设定值
D100	采样时间
D101	控制细节设置
D102	一阶惯性滤波系数
D106	工作区间
D109	工作模式

4、参数控制效果说明

① 烧水实验

A.自整定过程及控制过程(无过渡区设置), 以两段式自整定为例

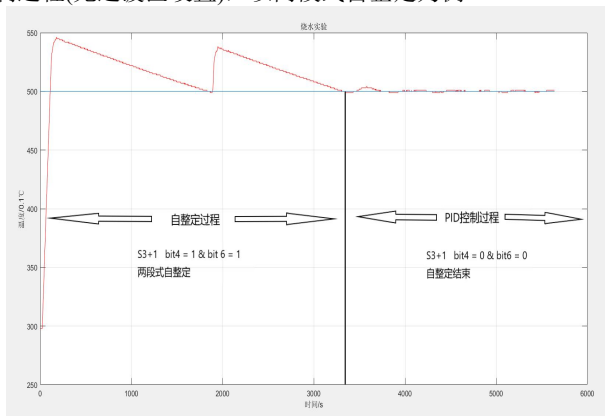


图 1 无过渡区自整定过程曲线

当控制系统为单温控的系统或者环境干扰不会引起大的波动现象的系统, 通常选择无过渡区自整定, 这样比有过渡区的方法更加快速的完成自整定过程。

B.自整定过程及控制过程(过渡区设置)

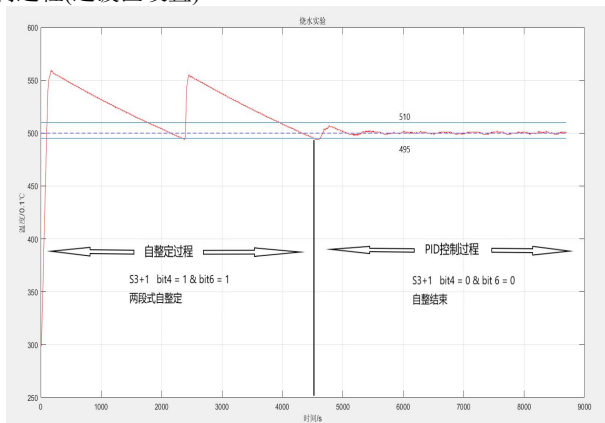


图 2 有过渡区自整定过程曲线

有过渡区自整定过程, 在双路控制系统中更加适用, 过渡区有 1.5°C 的大小。上限 1°C, 下限 0.5°C。

② 工作区间设置的区别

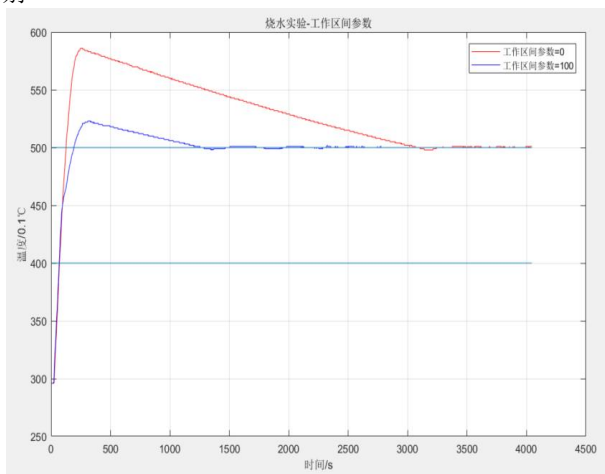


图 3 不同工作区间参数下的过程曲线

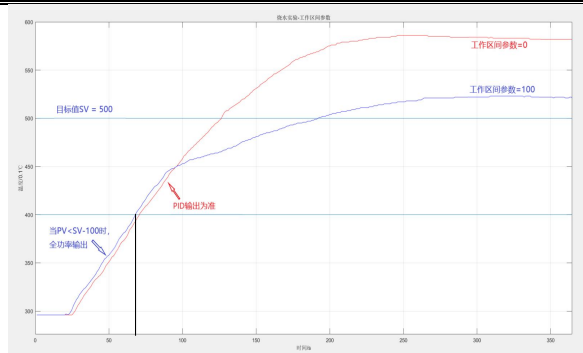


图 4 无不同工作区间参数下的过程曲线（升温过程图）

从局部放大的图形可以看出，工作区间的参数对超调量与稳定时间有一定的影响。在允许超调的情形下，设置工作区间参数，可使得超调量小些，这是因为 PID 开始工作的偏差 E 比较小，积分累计不会快速饱和。

③ 滤波系数设置的结果

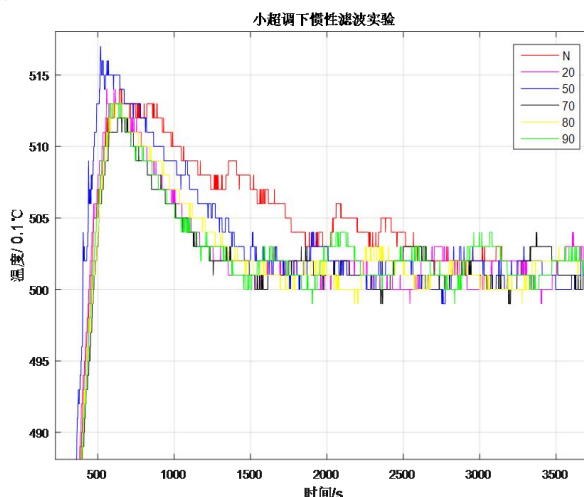


图 5 不同滤波参数下的过程曲线

上图为小超调系数下的实验结果，采样时间为 1s。而一阶惯性滤波的系数分别为（20，50，70，80，90）。加入惯性系数后很大程度上加快了系统控制的稳定时间，针对烧水实验提高 6 分钟左右。超调量在 1.2~1.7℃ 左右。

所以引入一阶惯性滤波在一定程度上对温度来回跳动的 PID 环境有很大的改善。对稳定时间的提高有改善。

注意：滤波系数这个参数对滞后性不是很大的系统有帮助。或者控制量存在来回波动的现象的控制效果有较大改善。

④ 模式选择的区别

0：允许超调（ukd = 100）；1：小超调或不超调（ukd = 300）。

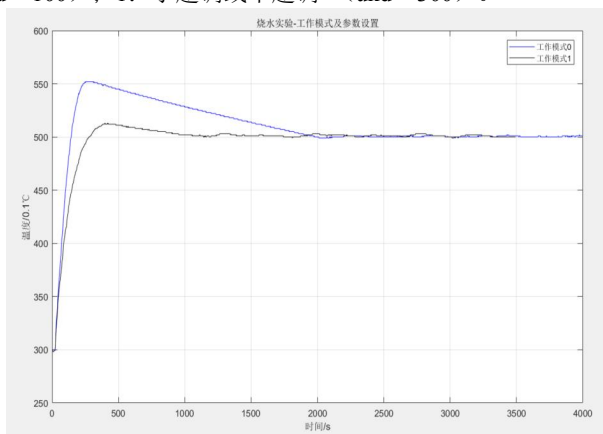


图 6 不同工作模式下的过程曲线

当选择模式 1 时（小超调或不超调），可能会出现的现象，稳定后的温度会略高于设定温度（在设定温度的上方波动）。

⑤ 说明系数的作用

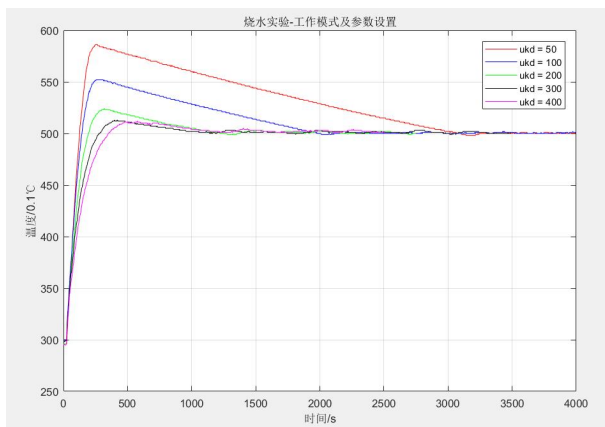


图 7 动态设置下的过程曲线

当选择工作模式 2 时，对应的就有 3 个可调节的参数 $ukp[S3+10]$ ， $uki[S3+11]$ ， $ukd[S3+12]$ 。通常对 ukp 与 uki 采用默认的参数即可，我们只调整 ukd 的取值就可实现控制效果。

Ukp 的调节是当 Kp 的值达到最大值时才进行调节的，通常取默认值 100。

Uki 是出现周期性振荡时进行调节。具体的调节方法，可逐步增大 uki 的取值，跟踪控制效果。适当调整 ukd 的取值。

PID 指令

指令描述：

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
PID	PID 运算	16	否	PID (S1)(S2)(S3)(D)	9

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)														√		
(S2)														√		
(S3)														√		
(D)														√		

该指令是完成 PID 运算，用于闭环控制系统参数的控制。PID 控制在机械设备、气动设备、恒压供水和电子设备等等中有着广泛的应用。其中：

(S1)为 PID 控制的目标值；

(S2)为实测的反馈值；

(S3)为 PID 运算所需设定参数、中间结果保存的缓存区起始地址，占用随后地址的共 26 个变量单元，取值范围是 D0~D7974，最好指定停电保持区，在电源 OFF 后仍保持设定值，否则首次启动运算前需对缓存区作赋值处理。其中每个单元的功能和参数描述详见本节说明；

(D)为 PID 计算结果的存放单元，请指定(D)为非电池保持区，否则首次启动运算前需对其作初始化清 0 处理。

编程示例：



其中参数说明如下：

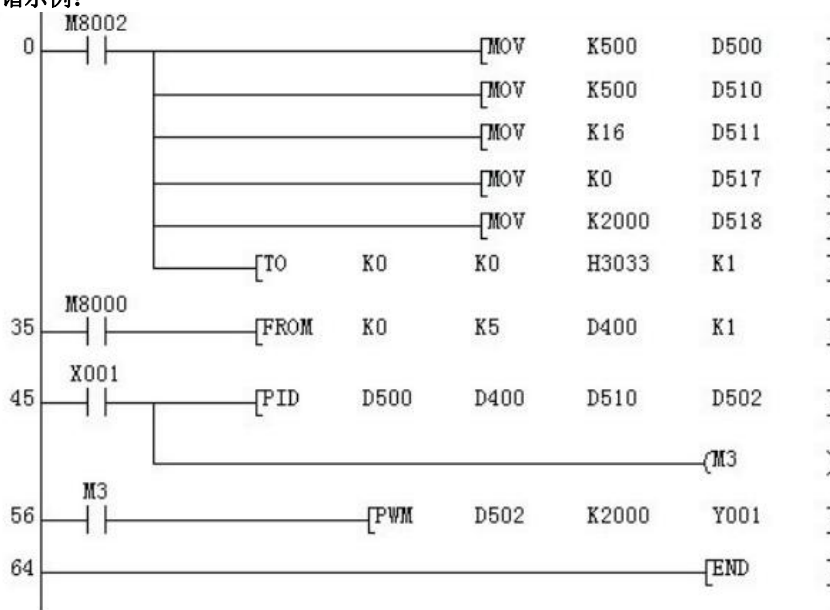
D9 中存放的是 PID 调节的目标值，D10 为闭环反馈值，注意 D9、D10 必需为同一量纲，如都为 0.01MPa 单位，或都为 1℃单位等等；

D200~D225 共 26 个单元用于存放 PID 运算的设定值与过程值，这些值在首次 PID 计算前就必需逐项进行设定；D130 单元则用于存放计算后的控制输出值，用于控制执行的动作。

就(S3)起始的各单元参数值的功能和设定方法说明如下表：

单元	功能	设定说明
(S3)	采样时间 (TS)	设定范围 1~32767 (ms)，但需大于 PLC 程序扫描周期
(S3)+1	动作方向 (ACT)	bit0: 0=正动作；1=逆动作 bit3: 0=单向；1=双向 bit4: 0=自整定不动作；1=执行自整定，其他不可使用。
(S3)+2	最大上升率 (DeltaT)	设定范围 0~320 即滤波
(S3)+3	比例增益 (Kp)	设定范围 0~32767，注意此值放大 256 倍，实际值为 Kp/256
(S3)+4	积分增益 (Ki)	设定范围 0~32767，Ki=16384Ts/Ti，Ti 作为积分时间
(S3)+5	微分增益 (Kd)	设定范围 0~32767，Kd≈Td/Ts，Td 为微分时间
(S3)+6	滤波 (C0)	设定范围 0~1024，积分部分滤波
(S3)+7	输出下限	推荐设定范围 -2000~2000 S3+1 bit3=0 0；S3+1 bit3=1-2000
(S3)+8	输出上限	推荐设定值 2000
(S3)+9	保留	保留内部使用
⋮	⋮	⋮
(S3)+25	保留	保留内部使用

自动调谐示例：



错误代码：

控制参数的设定值或 PID 运算中的数据发生错误时，则运算错误 M8067 变为 ON 状态，根据其错误内容 D8067 中存入下述数据。

代码	错误内容	处理状态	处理方法
K6705	应用指令的操作数在对象软元件范围外	PID 命令运算停止	请确认控制数据的内容
K6706	应用指令的操作数在对象软元件范围外		
K6730	采样时间(TS)在对象软元件范围外(TS<0)		
K6732	滤波(C0)范围外(C0<0 或 1024≤C0)		
K6732	最大上升率(DeltaT)范围外 ΔT<0 或 320≤ΔT		
K6733	比例增益(KP)在对象范围外(KP<0)		
K6734	积分增益(KI)在对象范围外(KI<0)		
K6735	微分增益(KD)在对象范围外(KD<0)		
K6740	采样时间(Ts)≤运算周期	PID 命令运算继续	
K6742	测定值变化量超过((PV<-32768 或 32767<(PV)		
K6751	自整定动作方向不一致	自整定继续	从自整定开始时的测定值预测的动作方向和自整定用输出时实际动作方向不一致。请使目标值、自整定用输出值、测定值的关系正确后，再次进行自整定。
K6752	自整定动作不良	自整定	自整定中测定值因上下变化不能正确动作。请使采样时间远远大于输出的变化周期，增大输入滤波常数。设定变更后请再次进行自整定。

要点：必须在 PID 运算执行前，将正确的测定值读入 PID 测定值（PV）中，特别对模拟量输入模块的数值进行 PID 运算时，需注意其转换时间。

编程说明：

1、PID 指令在程序中可多次使用，可同时执行，但各 PID 指令中使用的变量区域不要有重叠；在步进指令、跳转指令、定时中断、子程序中也可以使用，在此情况下执行 PID 指令时，需事先清除 S3+9 缓存单元。

2、采样时间 TS 的最大误差为 - (1 运算周期+1ms) ~ + (1 运算周期)。如果采样时间 TS ≤可编程控制器的 1 个运算周期，则发生下述的 PID 运算错误 (K6740)，并以 TS =运算周期执行 PID 运算。在这种情况下，建议最好使用恒定扫描模式或在定时器中断 (16□□~18□□) 中使用 PID 指令。

5.2.13 浮点运算

DECMP 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DECMP	二进制浮点数比较	32	否	DECM (S1) (S2) (D)	13
DECMPP		32	是		13

该指令是进行 2 个浮点数变量的比较，将比较的结果输出到 (D) 起始的 3 个变量中。其中：

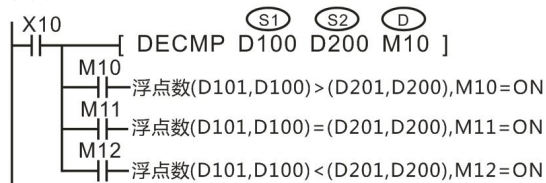
(S1) 为待比较的二进制浮点数 1；

(S2) 为待比较的二进制浮点数 2；

(D) 为比较结果的存放单元，共占用 3 个（位）变量单元。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√	√	√							√		
(S2)					√	√	√							√		
(D)			√	√	√											

编程示例:



当 X10=ON 时，M10~M12 其中之一会 ON。

X10 由 ON 变 OFF 时，不执行 DECP 指令，M10-M12 仍保持 X10=OFF 之前的状态，要清除 M10~M12 的比较结果，可用 RST 或者 ZRST 对 M10~M12 进行清除。

若需要得到 $\geq$ 、 $\leq$ 、 $\neq$ 的结果时，可将 M10~M12 串并联即可取得。

若 (S1) 或 (S2) 为 K、H 常数，系统会自动转换为浮点数参与运算。

DEZCP 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DEZCP	二进制浮点数区域比较	32	否	DEZCP (S1) (S2) (S) (D)	9
DEZCPP		32	是		17

该指令是进行二进制浮点数变量的区间比较，将比较的结果输出到启始的 3 个变量中。其中：

(S1)为二进制浮点变量区间的下限；

(S2)为二进制浮点变量区间的上限；

(S)待比较的二进制浮点变量；

(D)为比较结果的存放单元，共占用 3 个（位）变量单元。

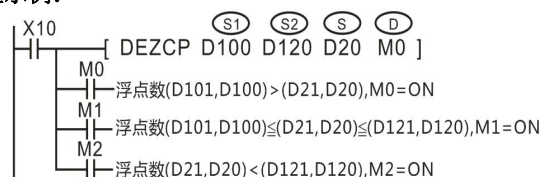
若 $(S) < (S1)$ ，则 M0=ON；

若 $(S2) > (S) > (S1)$ ，则 M1=ON；

若 $(S) > (S2)$ ，则 M2=ON；

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√	√	√							√		
(S2)					√	√	√							√		
(S)					√	√	√							√		
(D)		√	√	√												

编程示例:



DEBCD 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DEBCD	二进制浮点数转十进制浮点数	32	否	DEBCD (S) (D)	9
DEBCDP		32	是		9

该指令是进行二进制浮点数转换为十进制浮点的运算。其中：

(S)为二进制浮点变量；

(D)为转换为十进制浮点数结果的存放单元。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)							√							√		
(D)														√		

编程示例:



将二进制浮点数 (D3, D2) 转换成十进制浮点数后，存放于 (D11, D10) 单元。其中二进制浮点数 (D3, D2) 实数 23 位，指数 8 位。符号位 1 位。十进制浮点数 (D11, D10) 指数 (D3)，实数 (D2)，用科学计算式表示为 $D2 * 10^{D3}$ 。

PLC 内部浮点数据计算均为二进制形式，转换为十进制，可方便监控。

DEBIN 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DEBIN	十进制浮点数转二进制浮点数	32	否	DEBIN (S) (D)	9
DEBINP		32	是		9

该指令是进行十进制浮点数转换为二进制浮点的运算。其中:

(S) 为十进制浮点变量;

(D) 为转换为二进制浮点数结果的存放单元。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)							√							√		
(S2)														√		

编程示例:



将十进制浮点数 3.142 (先放于 D11, D10) 转换成二进制浮点数后, 存放于 (D3, D2) 单元。

DEADD 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DEADD	二进制浮点加法	32	否	DEADD (S1) (S2) (D)	13
DEADDP		32	是		13

该指令是进行二进制浮点的加法运算。其中:

(S1) 和 (S2) 分别为二进制浮点的被加数和加数;

(D) 为二进制浮点加法结果的存放单元。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√	√	√							√		
(S2)					√	√	√							√		
(D)														√		

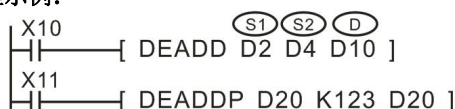
(S1) 或 (S2) 来源操作数若是常数 K 或 H, 会自动将该常数转换成二进制浮点数值来作加法运算;

若计算结果为零, 则 0 标志 (M8020) 会置位。

若运算结果的绝对值大于可表示的最大浮点值, 则进位标志 (M8022) 会置位。

若运算结果的绝对值小于可表示的最小浮点值, 则借位标志 (M8021) 会置位。

编程示例:



当 X10=ON 时, 二进制浮点数 (D3, D2) 与二进制浮点数 (D5, D4) 相加后, 二进制浮点数和存放于 (D11, D10) ;

当 X11 由 OFF 变为 ON 时, 二进制浮点数 (D21, D20) 的值增大 123。这里的常数 K123 在运算前已自动被调整为二进制浮点数;

和的存放单元可以与加数或被加数为同一单元, 此时请使用脉冲执行型指令 DEADDP, 否则若采用连续执行指令, 则程序每扫描一次, 计算就会被执行一次。

DESUB 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	操作数适用的软元件	步数
DESUB	二进制浮点数减法	32	否	DESUB (S1) (S2) (D)	13
DESUBP		32	是		13

该指令是进行二进制浮点的减法运算。其中:

(S1)和(S2)分别为二进制浮点的被减数和减数;

(D)为二进制浮点减法差的存放单元。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√	√	√							√		
(S2)					√	√	√							√		
(D)														√		

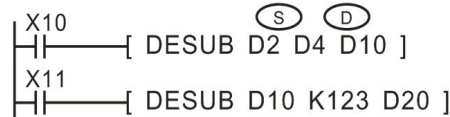
(S1)或(S2)来源操作数若是常数 K 或 H , 会自动将该常数变换成二进制浮点数值来作减法运算;

若计算结果为零, 则 0 标志 (M8020) 会置位。

若运算结果的绝对值大于可表示的最大浮点值, 则进位标志 (M8022) 会置位。

若运算结果的绝对值小于可表示的最小浮点值, 则借位标志 (M8021) 会置位。

编程示例:



当 X10=ON 时, 二进制浮点数 (D3, D2) 减去二进制浮点数 (D5, D4) 后, 二进制浮点数差存放于 (D11, D10) ;

当 X11 由 OFF 变为 ON 时, 二进制浮点数 (D21, D20) 的值减小 123。这里的常数 K123 在运算前已自动被调整为二进制浮点数;

差的存放单元可以与减数或被减数为同一单元, 此时请使用脉冲执行型指令 DESUBP, 否则若采用连续执行指令, 则程序每扫描一次, 计算就会被执行一次。

DEMUL 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DEMUL	二进制浮点数乘法	32	否	DEMUL (S1) (S2) (D)	13
DEMULP		32	是		13

该指令是进行二进制浮点的乘法运算。其中:

(S1)和(S2)分别为二进制浮点的被乘数和乘数;

(D)为二进制浮点乘法积的存放单元。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√	√	√							√		
(S2)					√	√	√							√		
(D)														√		

(S1)或(S2)来源操作数若是常数 K 或 H , 会自动将该常数变换成二进制浮点数值来作乘法运算;

若计算结果为零, 则 0 标志 (M8020) 会置位。

若运算结果的绝对值大于可表示的最大浮点值, 则进位标志 (M8022) 会置位。

若运算结果的绝对值小于可表示的最小浮点值, 则借位标志 (M8021) 会置位。

编程示例:



当 X12=ON 时, 二进制浮点数 (D3, D2) 乘以二进制浮点数 (D5, D4) 后, 二进制浮点数积存放于 (D11, D10) ;

当 X13 由 OFF 变为 ON 时, 二进制浮点数 (D21, D20) 的值乘以 3 倍后存回 (D21, D20)。这里的常数 K3 在运算前已自动被调整为二进制浮点数;

积的存放单元可以与乘数或被乘数为同一单元, 此时请使用脉冲执行型指令 DEMULP, 否则若采用连续执行指令, 则程序每扫描一次, 计算就会被执行一次。

DEDIV 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DEDIV	二进制浮点数除法	32	否	DEDIV (S1) (S2) (D)	13
DEDIVP		32	是		13

该指令是进行二进制浮点的除法运算。其中:

(S1)和(S2)分别为二进制浮点的被除数和除数;

(D)为二进制浮点除法商的存放单元起始地址。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√	√	√							√		
(S2)					√	√	√							√		
(D)														√		

(S1)或(S2)来源操作数若是常数 K 或 H , 会自动将该常数变换成二进制浮点数值来作除法运算;

若计算结果为零, 则 0 标志 (M8020) 会置位。

若运算结果的绝对值大于可表示的最大浮点值, 则进位标志 (M8022) 会置位。

若运算结果的绝对值小于可表示的最小浮点值, 则借位标志 (M8021) 会置位。

除数不得为 0, 否则计算出错, M8067、M8068 会置 ON。

编程示例:



当 X14=ON 时, 二进制浮点数 (D3, D2) 除以二进制浮点数 (D5, D4) 后, 二进制浮点数商存放于 (D11, D10) ;

当 X15 由 OFF 变为 ON 时, 二进制浮点数 (D11, D10) 的值除以 10 后存回 (D11, D10)。这里的常数 K10 在运算前已自动被调整为二进制浮点数;

商的存放单元可以与除数或被除数为同一单元, 此时请使用脉冲执行型指令 DEDIVP, 否则若采用连续执行指令, 则程序每扫描一次, 计算就会被执行一次。

DESQR 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DESQR	二进制浮点数开方	32	否	DESQR (S) (D)	9
DESQRP		32	是		9

该指令是进行二进制浮点的开平方运算，即求二进制浮点数的平方根。其中：

(S) 为待求平方根的二进制浮点数变量；

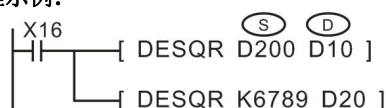
(D) 为二进制浮点平方根的存储单元。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)					√	√	√							√		
(D)														√		

操作数 (S) 是常数 K 或 H，会自动将该常数变换成二进制浮点数值来作开方运算；若计算结果为零，则 0 标志 (M8020) 会置位。

(S) 只有正数有效，如果是负数则计算出错，M8067、M8068 会置 ON。

编程示例:



案例 1: 将二进制浮点数的开方结果 √(D201, D200) 存放到 (D11, D10)。

案例 2: 将二进制浮点数 K6789 做开方，结果存放到 (D21, D20)，这里的常数 K6789 在运算前已自动被调整为二进制浮点数。

INT 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
INT	二进制浮点数转 BIN 整数变换	16	否	INT (S) (D)	5
INTP		16	是		5
DINT		32	否		9
DINTP		32	是		9

该指令是进行二进制浮点的取整运算，丢弃小数部分，将二进制结果存于 (D) 中。其中：

(S) 为待取整变换的二进制浮点数变量；

(D) 为变换后 BIN 整数结果的存储单元。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)							√							√		
(D)														√		

若计算结果为零，则 0 标志 (M8020) 会置位。

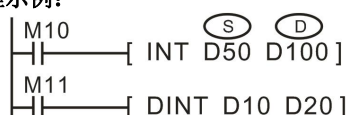
若运算结果有浮点数被舍弃时，则借位标志 (M8021) 会置位。

若运算结果若超出下列范围时 (溢位)，则进位标志 (M8022) 会置位。

16 位指令: -32, 768~32, 767

32 位指令: -2, 147, 483, 648~2, 147, 483, 647

编程示例:



当 M10 置 ON 后，将浮点数 (D51, D50) 取整后，存放于 D100 中。

当 M11 置 ON 后，将浮点数 (D11, D10) 取整后，存放于 (D21, D20) 中。

注意 INT 和 DINT 指令存放结果的区别。

DSIN 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DSIN	浮点数 SIN 运算	32	否	DSIN (S) (D)	9
DSINP		32	是		9

该指令是求指定角度（RAD，弧度）的 SIN（正弦）值，变量为二进制浮点存储格式。其中：

(S)为待求正弦值的角度变量，RAD 单位，以二进制浮点数表示。取值范围 $0 \leq \alpha \leq 2\pi$ ；

(D)为变换后 SIN 计算结果的存储单元，二进制浮点数格式。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)														√		
(D)														√		

编程示例:

指令示例 1:

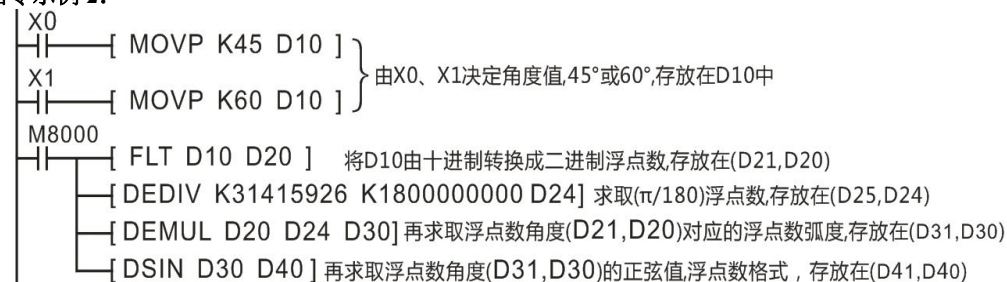


将弧度（D21，D20）求 SIN 值，存放到(D31，D30)。

这里计算的源数据、SIN 结果都为二进制浮点数格式。

$RAD(弧度)值 = 角度 \times \pi / 180^\circ$ ，如角度 360° 对应的弧度 $= 360^\circ \times \pi / 180^\circ = 2\pi$ 。

指令示例 2:



DCOS 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DCOS	浮点数 COS 运算	32	否	DCOS (S) (D)	9
DCOSP		32	是		9

该指令是求指定角度（RAD，弧度）的 COS（余弦）值，变量为二进制浮点存储格式。其中：

(S)为待求余弦值的角度变量，RAD 单位，以二进制浮点数表示。取值范围 $0 \leq \alpha \leq 2\pi$ ；

(D)为变换后 COS 计算结果的存储单元，二进制浮点数格式。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)														√		
(D)														√		

编程示例:

M10 ——— [DCOS (S) (D)]
 D20 D30

将弧度（D21，D20）求 COS 值，存放到（D31，D30）。

这里计算的源数据、COS 结果都为二进制浮点数格式。

RAD(弧度)值=角度 $\times \pi/180^\circ$ ，如角度 360° 对应的弧度= $360^\circ \times \pi/180^\circ = 2\pi$ 。

关于以角度求取 COS 值的编程语句，可参考 SIN 指令中的举例。

DTAN 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DTAN	浮点数 TAN 运算	32	否	DTAN (S) (D)	9
DTANP		32	是		9

该指令是求指定角度（RAD，弧度）的 TAN（正切）值，变量为二进制浮点存储格式。其中：

(S)为待求正切值的角度变量，RAD 单位，以二进制浮点数表示。取值范围 $0 \leq \alpha < 2\pi$ ；

(D)为变换后 TAN 计算结果的存储单元，二进制浮点数格式。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)														√		
(D)														√		

编程示例:

X2 ——— [DTAN (S) (D)] 将弧度(D21，D20)求 TAN 值，存放在(D31，D30)中。
 D20 D30

这里计算的源数据、TAN 结果都为二进制浮点数格式。

RAD(弧度)值=角度 $\times \pi/180^\circ$ ，如角度 360° 对应的弧度= $360^\circ \times \pi/180^\circ = 2\pi$ 。

关于以角度求取 TAN 值的编程语句，可参考 SIN 指令中的举例。

DASIN 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DASIN	根据 SIN 值计算对应弧度值	32	否	DASIN (S) (D)	9
DASINP		32	是		9

该指令是从 SIN 值求出对应弧度的运算。其中:

(S)为存放待求的 2 进制浮点数变量。取值范围-1.0<= α <=1.0;

(D)为计算结果的存储单元(-0.5 π ~+0.5 π) 。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)							√							√		
(D)														√		

编程示例:

指令示例 1:



当 M10 为 ON 时，将 (D1、D0) 中二进制浮点数数值进行 SIN -1 运算后，保存到 (D3、D2) 中。

SIN^{-1} (D1、D0) $\longrightarrow$ (D3、D2)

这里计算的源数据、ASIN 结果都为二进制浮点数格式。

$RAD(弧度)值 = 角度 \times \pi / 180^\circ$ ，如角度 360° 对应的弧度 $= 360^\circ \times \pi / 180^\circ = 2\pi$ 。

指令示例 2:



假如 (D1、D0) 为 0.707106781，当 M10 由 OFF 到 ON 的时候，(D3、D2) 的值为 0.78539815，(D5、D4) 的值为 45，(D7、D6) 的值为 45。

DACOS 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DACOS	根据 COS 值计算对应弧度值	32	否	DACOS (S) (D)	9
DACOSP		32	是		9

该指令是从 COS 值求出对应弧度的运算。其中:

(S)为存放待求的 2 进制浮点数变量。取值范围-1.0<= α <=1.0;

(D)为计算结果的存储单元(0~ 11)。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)							√							√		
(D)														√		

编程示例:

指令示例 1:



当 M10 为 ON 时, 将 (D1、D0) 中二进制浮点数数值进行 COS-1 运算后, 保存到 (D3、D2) 中。

COS-1 (D1、D0) ——> (D3、D2)

这里计算的源数据、DACOS 结果都为二进制浮点数格式。

RAD(弧度)值=角度× π /180° , 如角度 360° 对应的弧度=360° × π /180° =2 π 。

指令示例 2:



假如 (D1、D0) 为 0.866025404, 当 M10 由 OFF 到 ON 的时候, (D3、D2) 的值为 0.52359877, (D5、D4) 的值为 30, (D7、D6) 的值为 30。

DATAN 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DATAN	根据 TAN 值计算对应弧度值	32	否	DATAN (S) (D)	9
DATANP		32	是		9

该指令是从 TAN 值求出对应弧度的运算。其中:

(S) 为存放待求的 2 进制浮点数变量。

(D) 为计算结果的存储单元($-\pi/2 \sim \pi/2$)。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)							√							√		
(D)														√		

编程示例:

指令示例 1:



当 M10 为 ON 时, 将 (D1、D0) 中二进制浮点数数值进行 TAN -1 运算后, 保存到 (D3、D2) 中。

TAN -1 (D1、D0) $\longrightarrow$ (D3、D2)

这里计算的源数据、ATAN 结果都为二进制浮点 数格式。

RAD(弧度)值=角度 $\times \pi/180^\circ$, 如角度 360° 对应的弧度= $360^\circ \times \pi/180^\circ = 2\pi$ 。

指令示例 2:



假如 (D1、D0) 为 1.732050808, 当 M10 由 OFF 到 ON 的时候, (D3、D2) 的值为 1.04719753, (D5、D4) 的值为 60, (D7、D6) 的值为 60。

DSINH 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DSINH	2 进制浮点数双曲正弦函数 SINH 运算	32	否	DSINH (S) (D)	9
DSINHP		32	是		9

该指令是进行 2 进制浮点数取 SINH 值。其计算公式为 $\text{SINH 值} = (e^s - e^{-s})/2$ ，其中：

(S) 为存放待求 SINH 值的 2 进制浮点数变量；

(D) 为计算结果的存储单元。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)							√							√		
(D)														√		

编程示例:

```

M10 ┌───┐ (S) (D)
    │   │ [ DSINH D0 D2 ]
M11 └───┘
M11 ┌───┐
    │   │ [ DSINHP D10 D12 ]
M11 └───┘
  
```

当 M10 为 ON 时，将 (D1、D0) 中二进制浮点数数值进行 DSINH 运算后，保存到 (D3、D2) 中。

DCOSH 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DCOSH	2 进制浮点数双曲余弦函数 COSH 运算	32	否	DCOSH (S) (D)	9
DCOSHHP		32	是		9

该指令是进行 2 进制浮点数取 COSH 值。其计算公式为 $\text{cosh 值} = (e^s + e^{-s})/2$ ，其中：

(S) 为存放待求 COSH 值的 2 进制浮点数变量；

(D) 为计算结果的存储单元。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)							√							√		
(D)														√		

编程示例:

指令示例 1:

```

M10 ┌───┐ (S) (D)
    │   │ [ DCOSH D0 D2 ]
M11 └───┘
M11 ┌───┐
    │   │ [ DCOSHHP D10 D12 ]
M11 └───┘
  
```

当 M10 为 ON 时，将 (D1、D0) 中二进制浮点数数值进行 COSH 运算后，保存到 (D3、D2) 中。

DCOSH (D1、D0) → (D3、D2)

DTANH 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DTANH	2 进制浮点数双曲正切函数 TANH 运算	32	否	DTANH (S) (D)	9
DTANHP		32	是		9

该指令是进行 2 进制浮点数取 TANH 值。其计算公式为 $\tanh \text{ 值} = (e^s - e^{-s}) / (e^s + e^{-s})$ ，其中：

(S) 为存放待求 TANH 值的 2 进制浮点数变量；

(D) 为计算结果的存储单元。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)							√							√		
(D)														√		

编程示例:



DDEG 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DDEG	将 2 进制浮点数弧度转换成角度	32	否	DDEG (S) (D)	9
DDEGP		32	是		9

该指令是进行 2 进制浮点数弧度转换成角度的运算。其计算公式为 $[\text{角度单位} = \text{弧度单位} \times \pi / 180]$ ，其中：

(S) 为存放待求角度的 2 进制浮点数弧度变量；

(D) 为计算结果的存储单元。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)							√							√		
(D)														√		

编程示例:

指令示例 1:



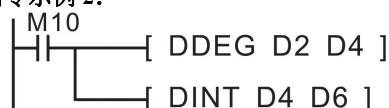
当 M10 为 ON 时，将 (D1、D0) 中二进制浮点数数值进行弧度到角度的运算后，保存到 (D3、D2) 中。

DDEG (D1、D0) → (D3、D2)

这里计算的源数据、DEG 结果都为二进制浮点数格式。

DEG(弧度)值 = 角度 $\times \pi / 180^\circ$ ，如角度 360° 对应的弧度 = $360^\circ \times \pi / 180^\circ = 2\pi$ 。

指令示例 2:



假如 (D1、D0) 为 3.1415926，当 M10 由 OFF 到 ON 的时候，(D3、D2) 的值为 180，经过浮点数变整型的运算后，得到 (D5、D4) 的值为 180。

DRAD 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
DRAD	进行 2 进制浮点数角度转换成弧度的运算	32	否	DRAD (S) (D)	9
DRADP		32	是		9

该指令是进行 2 进制浮点数角度转换成弧度的运算。其计算公式为【弧度单位=角度单位 $\times \pi/180$ 】，其中：

(S) 为存放待求弧度的 2 进制浮点数角度变量；

(D) 为计算结果的存储单元。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)							√							√		
(D)														√		

编程示例:

指令示例 1:



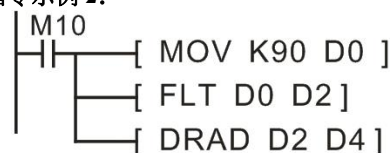
当 M10 为 ON 时，将 (D1、D0) 中二进制浮点数数值进行角度到弧度的运算后，保存到 (D3、D2) 中。

DRAD (D1、D0) $\rightarrow$ (D3、D2)

这里计算的源数据、RAD 结果都为二进制浮点数格式。

RAD(弧度)值=角度 $\times \pi/180^\circ$ ，如角度 360° 对应的弧度= $360^\circ \times \pi/180^\circ = 2\pi$ 。

指令示例 2:



当 M10 由 OFF 到 ON 的时候，将 90 赋给 D0，D0 经过整数到浮点数的运算赋给 (D3、D2)，(D3、D2) 经过角度到弧度的运算赋给 (D5、D4)，最后得 (D5、D4) 的值为 $\pi/2$ ，即为 1.570796。

DEXP 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	操作数适用的软元件	步数
DEXP	进行以 e(2.71828)为底的二进制浮点数数据的指数运算	32	否	DEXP (S) (D)	5
DEXPP		32	是		9

该指令是将(S)以 e 为底数的 2 进制浮点数的指数运算数据值复制到(D)中。

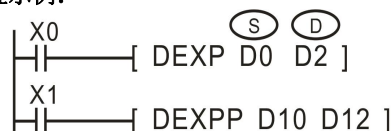
(S)为二进制浮点数数据的传送源;

(D)为保存二进制浮点数数据的存放单元。

注意: 当运算结果不在【 $2 \cdot 126 \leq \text{运算结果} < 2128$ 】时将发生运算出错。错误代码为 K6706, K6706 保存在 D8067 中, 出错时标志位 M8067 置 ON。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)					√	√	√							√		
(D)														√		

编程示例:



当 X0 为 ON 时, 以 e 为底对 (D1, D0) 中的 2 进制浮点数数值进行指数运算后保存到 (D3, D2) 中。E(D1, D0)→(D3, D2), 由于 $\log_e 2128 = 88.7$, 因此当 (D1, D0) 大于 88.7 时, D8067 为 K6706, M8067 置 ON。

DLOG10 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	操作数适用的软元件	步数
DLOG10	进行以 10 为底的二进制浮点数数据的常用对数运算	32	否	DLOG10 (S) (D)	9
DLOG10P		32	是		9

指令是进行以 10 为底的二进制浮点数数据的常用对数运算。其中:

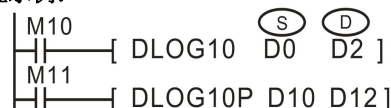
(S)为待求二进制浮点数常用对数的 2 进制浮点数变量;

(D)为常用对数运算后计算结果的存储单元。

注意: (S)中的值只能为正数, 当(S)的内容值为 0 或者负数时将发生运算出错。错误代码为 K6706, K6706 会保存在 D8067 中, 出错标志位 M8067 置 ON。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)							√							√		
(D)														√		

编程示例:



当 M10 为 ON 时, 以 10 为底对 (D1, D0) 中的 2 进制浮点数数值进行常用对数运算后保存到 (D3, D2) 中。
 $\log_{10}(D1, D0) \rightarrow (D3, D2)$

DLOGE 指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	操作数适用的软元件	步数
DLOGE	进行以 e(2.71828)为底的二进制浮点数数据的自然对数运算	32	否	DLOGE (S) (D)	9
DLOGEP		32	是		9

该指令是进行以 e(2.71828)为底的二进制浮点数数据的自然对数运算。其中:

(S)为待求二进制浮点数自然对数的 2 进制浮点数变量;

(D)为自然对数运算后计算结果的存储单元。

注意: (S)中的值只能为正数, 当(S)的内容值为 0 或者负数时将发生运算出错。错误代码为 K6706, K6706 会保存在 D8067 中, 出错标志位 M8067 置 ON。

操作数	位元件				字元件											
	X	Y	M	S	K	H	E	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S)							√							√		
(D)														√		

编程示例:



当 X0 为 ON 时, 以 e 为底对 (D1、D0) 中的 2 进制浮点数数值进行自然对数运算后保存到 (D3、D2) 中。log_e(D1, D0)→(D3, D2)

从自然对数向常用对数的转换 (用 0.4342945 分割常用对数的值) 公式: $10^x = e^{\frac{x}{0.4342945}}$

5.2.14 触点比较指令

LD 系列触点比较指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
LD=	(s1)=(s2)导通	16	否	LD* (S1) (S2) *比较符中有=、>、<、>=、<=、◇等。	5
LDD=	(s1)=(s2)导通	32	否		9
LD>	(s1)>(s2)导通	16	否		5
LDD>	(s1)>(s2)导通	32	否		9
LD<	(s1)<(s2)导通	16	否		5
LDD<	(s1)<(s2)导通	32	否		9
LD◇	(s1)≠(s2)导通	16	否		5
LDD◇	(s1)≠(s2)导通	32	否		9
LD≤	(s1)≤(s2)导通	16	否		5
LDD≤	(s1)≤(s2)导通	32	否		9
LD≥	(s1)≥(s2)导通	16	否		5
LDD≥	(s1)≥(s2)导通	32	否		9

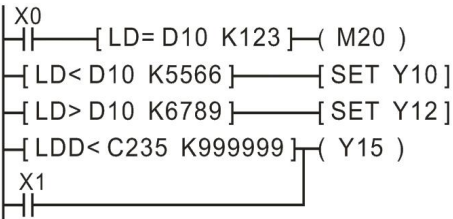
操作数	位元件				字元件										
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√	√	√	√	√	√	√	√	√	√	√
(S2)					√	√	√	√	√	√	√	√	√	√	√

该指令将两个操作数进行比较，将比较结果以逻辑状态输出，参与比较的变量都按有符号数处理，其中：

(S1)为待比较的数据源或数据变量单元 1；

(S2)为待比较的数据源或数据变量单元 2。

编程示例:



当 X0=ON 且 D10=K123 时，M20=ON。

当 D10<K5566 时，Y10=ON 并保持住。

当 D10>K6789 时，Y12=ON 并保持住。

当 C235<K999999，或者 X1=ON 时，Y15=ON。

对于参与比较的数为 32bit 宽度的计数器，应使用 32bit 指令 LDD，否则出错。 32 位计数器(C200~C255)以本指令作比较时，一定要使用 32 位指令。

AND 系列触点比较指令

指令描述:

名称	功能	位数(bits)	是否脉冲型	指令格式	步数
AND=	(s1)=(s2)导通	16	否	AND* (S1) (S2) *比较符中有=、>、<、>=、<=、<>等。	5
ANDD=	(s1)=(s2)导通	32	否		9
AND>	(s1)>(s2)导通	16	否		5
ANDD>	(s1)>(s2)导通	32	否		9
AND<	(s1)<(s2)导通	16	否		5
ANDD<	(s1)<(s2)导通	32	否		9
AND<>	(s1)≠(s2)导通	16	否		5
ANDD<>	(s1)≠(s2)导通	32	否		9
AND<=	(s1)<=(s2)导通	16	否		5
ANDD<=	(s1)<=(s2)导通	32	否		9
AND>=	(s1)>=(s2)导通	16	否		5
ANDD>=	(s1)>=(s2)导通	32	否		9

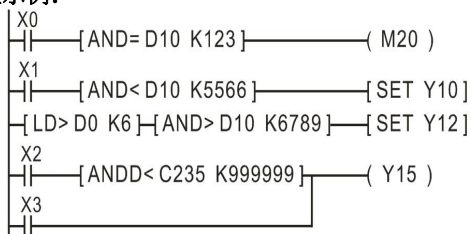
操作数	位元件				字元件										
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z
(S1)					√	√	√	√	√	√	√	√	√	√	√
(S2)					√	√	√	√	√	√	√	√	√	√	√

该指令之前已有其他逻辑操作。该指令将两个操作数进行比较，将比较结果以逻辑状态形式参与程序能流的运算，指令中参与比较的变量都按有符号数处理，其中：

(S1)为待比较的数据源或数据变量单元 1；

(S2)为待比较的数据源或数据变量单元 2。

编程示例:



当 X0=ON 且 D10=K123 时，M20=ON。

当 X1=ON 且 D10<K5566 时，Y10=ON 并保持住。

当 D0>K6 且 D10>K6789 时，Y12=ON 并保持住。

当 X2=ON 且 C235<K999999，或者 X3=ON 时，Y15=ON。

对于参与比较的数为 32bit 宽度的计数器，应使用 32bit 指令 ANDD，否则出错。32 位计数器(C200~C255)以本指令作比较时，一定要使用 32 位指令。

OR 系列触点比较指令

指令描述:

名称	功能	位数 (bits)	是否脉冲型	指令格式	步数
OR=	(s1)=(s2)导通	16	否	OR* $\textcircled{S1}$ $\textcircled{S2}$ *比较符号中有=、>、<、>=、<=、<>等。	5
ORD=	(s1)=(s2)导通	32	否		9
OR>	(s1)>(s2)导通	16	否		5
ORD>	(s1)>(s2)导通	32	否		9
OR<	(s1)<(s2)导通	16	否		5
ORD<	(s1)<(s2)导通	32	否		9
OR<>	(s1)≠(s2)导通	16	否		5
ORD<>	(s1)≠(s2)导通	32	否		9
OR<=	(s1)<=(s2)导通	16	否		5
ORD<=	(s1)<=(s2)导通	32	否		9
OR>=	(s1)>=(s2)导通	16	否		5
ORD>=	(s1)>=(s2)导通	32	否		9

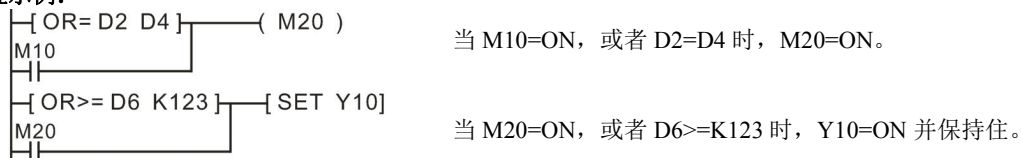
[illegible]

该指令将两个操作数进行比较，将比较结果以逻辑状态形式参与程序能流的或运算，指令中参与比较的变量都按有符号数处理，其中：

⑤(S1)为待比较的数据源或数据变量单元 1;

⑤S2为待比较的数据源或数据变量单元2。

编程示例：



对于参与比较的数为 32bit 宽度的计数器，应使用 32bit 指令 ORD，否则出错。32 位计数器(C200~C255)以本指令作比较时，一定要使用 32 位指令。

5.3 步进顺控指令说明

STL、RET 指令

助记符、名称	功能	可用软元件	程序步
STL	步序动作开始	S	1
RET	步序动作结束	无	1

步进控制方式（STL）是将控制被划分为多个工序状态（S），依据条件进行状态转移（SET），逐步完成控制过程。步进控制方式的特点是将复杂的控制分步后，分别考虑好每一步的控制，从而降低了各步的关联，降低编程的复杂度。各状态内执行的动作由梯形图其他指令编写。

STL 是一个步序动作的开始指令，RET 是一个步序的结束指令，其后指令返回母线。

SET S[k]([k]表示十进制数)是 STL 状态发生转移的唯一指令。

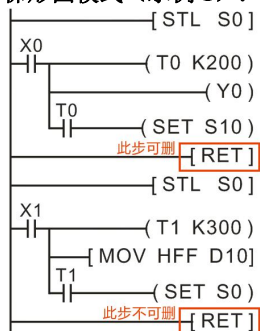
规定：子程序内不能使用 STL---RET 指令。

当前状态（S0）向下一状态（S1）转移时，该扫描周期两个状态内的动作均得到执行；下一扫描周期执行时，当前状态（S0）被下一状态（S1）所复位，当前状态（S0）内的动作不被执行，所有 OUT 元件的输入均被断开。

步序与步序之间一般省去 RET，因此看起来是多个 STL 共享一个 RET。有 STL 没有 RET 的程序在编译时会出错。

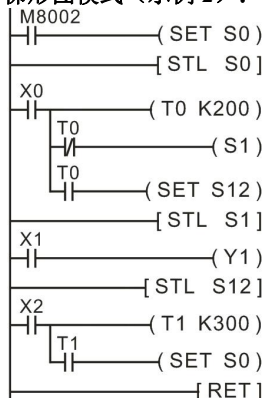
编程示例：

梯形图模式（示例 1）：



如左图所示，步序与步序之间一般可以省略 RET，因此看起来是多个 STL 共享一个 RET。而如果有 STL 没有 RET，程序编译时会报错。

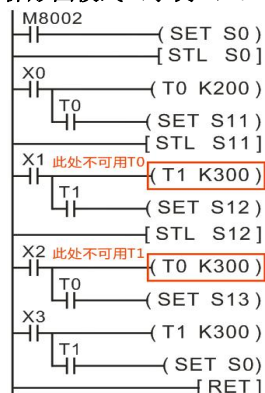
梯形图模式（示例 2）：



如左图所示，状态转移只能用 SET 指令，不能用 OUT 指令。

使用 OUT S 时，S 作为辅助继电器使用，而不是状态寄存器。

梯形图模式（示例 3）：



如左图所示，时间继电器 T 可重复使用，但相邻的两个状态不能重复使用同一时间继电器。

6. 快捷键一栏

6.1 快捷键

通用快捷键如下表所示：

快捷键	对应的菜单(功能名)	概要
Ctrl + N	新建工程	新建工程
Ctrl + O	打开工程	打开已存在的工程
Ctrl + S	保存工程	保存工程
Ctrl + X	剪切	剪切选择的数据和范围
Ctrl + C	复制	复制选择的数据和范围
Ctrl + V	粘贴	在光标位置剪切/粘贴复制的数据
Ctrl + Z	撤销	撤销前一操作
Ctrl + Y	恢复	恢复通过[撤销]取消的操作
Ctrl + F	搜索软元件	搜索软元件
Ctrl + F1	显示菜单工具/隐藏栏	显示/隐藏菜单工具栏
F3	监视开始	当前，开始监视作为操作对象的窗口
Ctrl + F3	监视停止	当前，停止监视作为操作对象的窗口
F4	变换/变换+编译	编译(变换)当前编辑中的程序
Alt+F4	退出软件	退出软件

6.2 编辑器的快捷键

梯形图编辑器中可使用的快捷键如下表所示：

快捷键	对应的菜单(功能名)	概要
F5	常开触点	将常开触点写入至光标位置
Shift + F5	常开触点 OR	将常开触点 OR 写入至光标位置
F6	常闭触点	将常闭触点写入至光标位置
Shift + F6	常闭触点 OR	将常闭触点 OR 写入至光标位置
F7	线圈	将线圈写入至光标位置
F8	应用指令	将应用指令写入至光标位置
F9	横线输入	将横线写入至光标位置
F11	竖线输入	将竖线写入至光标位置
Ctrl + F9	横线删除	删除光标位置中的横线
Ctrl + F11	竖线删除	删除光标位置中的竖线
Shift + F7	上升沿脉冲	将上升沿脉冲写入至光标位置
Shift + F8	下降沿脉冲	将下降沿脉冲写入至光标位置
Ctrl + Alt + F7	并联上升沿脉冲	将并联上升沿脉冲写入至光标位置
Ctrl + Alt + F8	并联下降沿脉冲	将并联下降沿脉冲写入至光标位置
Ctrl + Alt + F11	运算结果取反	将运算结果取反写入至光标位置
Ctrl + Shift + Insert	行间声明插入	将声明行插入至光标位置
Shift + Insert	程序行插入	将程序行插入至光标位置
Shift + Delete	行删除	删除光标位置的行
Ctrl + Insert	列插入	将列插入至光标位置
Ctrl + Delete	列删除	删除光标位置的列
Ctrl + →	右方向的横线输入/删除	从光标位置向右输入/删除横线
Ctrl + ←	左方向的横线输入/删除	从光标位置向左输入/删除横线
Ctrl + ↓	下方向的竖线输入/删除	从光标位置向下输入/删除竖线
Ctrl + ↑	上方向的竖线输入/删除	从光标位置向上输入/删除竖线
Ctrl + /	常开/常闭触点切换	切换常开触点与常闭触点
Ctrl + G	跳转	显示跳转指定的行
Ctrl + F5	注释显示	显示软元件注释、标签注释
Ctrl + F7	声明显示	显示声明
Ctrl + F8	注解显示	显示注解
F1	打开指令帮助	显示指令帮助

7.通讯运用

协议配置

维控 LX 系列 plc 中支持的协议如下：

COM1(圆头 8 针)：RS422 通信方式，支持编程口协议（即 PLC 协议），串口参数（波特率、数据位、校验位、停止位等）中波特率可修改，其他参数不可修改。波特率通过系统特殊地址 D8110 设置。

COM1(端子)：RS485 通信方式，支持编程口协议（即 PLC 协议），串口参数（波特率、数据位、校验位、停止位等）中波特率可修改，其他参数不可修改。波特率通过系统特殊地址 D8110 设置。

注意：COM1（RS422）和 COM1(RS485)不能同时使用。

COM2(端子)：RS485 通信方式，支持编程口协议（即 PLC 协议）、MODBUS 协议（MODBUS-RTU 从站、MODBUS-RTU 主站、MODBUS-ASCII 主站、MODBUS-ASCII 从站）、支持 N：N 网络通信协议(目前仅 LX3VP 系列 PLC 支持)。

1、COM1 口参数配置

COM1 中波特率的修改由 D8110 设定。D8110 的定义如下表：

位号	名称	内容	
		0 (OFF)	1 (ON)
B0	数据长度	7 位	——
B2B1	奇偶性	11：偶校验 (EVEN)	
B3	停止位	1 位	——
B7B6B5B4	波特率 (bps)	0111： 4800 1000： 9600 1001： 19200 1010： 38400 1011： 57600 1100： 115200	

2、COM2 口参数配置

COM2 中的通信协议由系统特殊地址 D8126 设定，具体定义如下表：

COM2 口的通信配置		
协议	D8126 数值设定	通信参数设定
RS 指令	00H	由 D8120 设定
HMI 监控协议 (PLC 协议)	01H	由 D8120 设定
MODBUS-RTU 从站	02H	由 D8120 设定
MODBUS-ASCII 从站	03H	由 D8120 设定
N：N 网络通信协议从站 (仅 LX3VP/LX3VE 系列 PLC 支持)	04H	由 D8120 设定
RS 指令	10H	由 D8120 设定
MODBUS-RTU 主站	20H	由 D8120 设定
MODBUS-ASCII 主站	30H	由 D8120 设定
N：N 网络通信协议主站 (仅 LX3VP/LX3VE 系列 PLC 支持)	40H	由 D8120 设定

串口参数由系统特殊地址 D8120 设定。具体定义如下表：

位编号	名称	内容	
		0 (OFF)	1 (ON)
B0	数据长度	7 位	8 位
B1B2	奇偶校验	00：无校验 01：奇校验(ODD)	11：偶校验 (EVEN)
B3	停止位	1 位	2 位
B4B5B6B	波特率(bps)	0111： 4800 1000： 9600 1001： 19200 1010： 38400 1011： 57600 1100： 115200	
B8	STX	无	由 D8124 设置
B9	ETX	无	由 D8125 设置

7.1 Modbus 运用

1、MODBUS 协议说明（com2 口通讯）

COM2 口采用 RS485 信号，采用双绞线进行联接即可，因此传输距离较远，可达 1000 米。由于 RS485 抗干扰性能比较好，且成本低，在工业控制设备的通讯中被广泛使用。

MODBUS 通讯协议是一种工业自动化总线通信协议，其标准、开发和免费等特点，支持数千种工业智能仪器仪表厂家数据通信，同时支持多种通信接口 RS232、RS485、TTL、光纤和无线等方式，协议公开、格式通俗易懂，广泛应用于多种工业自动化领域。现有众多厂家的变频器、控制器都采用了该通讯模式（RS485+MODBUS）。

传送数据格式有 HEX 码数据和 ASCII 码两种，分别称为 MODBUS-RTU 和 MODBUS-ASCII 协议，前者为数据直接传送，而后者需将数据变换为 ASCII 码后传送，因此 MODBUS-RTU 协议的通讯效率较高，处理简单，使用得更多。

MODBUS 为单主多从通讯方式，采用的是主问从答方式，每次通讯都是由主站首先发起，从站被动应答。因此，如变频器之类的被控设备，一般内置的是从站协议，而 PLC 之类的控制设备，则需具有主站协议、从站协议。

现以 MODBUS-RTU 协议为例，说明通讯帧的典型格式：

请求帧格式：从机地址+0x03+寄存器起始地址+寄存器数量+CRC 校验

例如： 01 03 00 01 00 02 CRC 校验

序号	数据(字节)意义	例子	字节数量	说明
1	从机地址	0x01	1 个字节	取值 1~247，由 D8121 设定
2	0x03（操作码）	0x03	1 个字节	读寄存器
3	寄存器起始地址	0x00 01	2 个字节	高位在前，低位在后，见寄存器编址
4	寄存器数量	0x00 02	2 个字节	高位在前，低位在后（N）
5	CRC 校验	0xc4 0b	2 个字节	高位在前，低位在后

正常响应帧格式：从机地址+0x03+字节数+寄存器值+CRC 校验：

例如：若设备返回的两个数据分别是 100 和 200，

则响应帧为： 01 03 04 00 64 00 C8 CRC 校验。

序号	数据(字节)意义	例子	字节数量	说明
1	从机地址	0x01	1 个字节	取值 1~247，由 D8121 设定
2	0x03（功能码）	0x03	1 个字节	读寄存器
3	字节数	0x04	1 个字节	值：N*2
4	寄存器值	0x00 64 0x00 c8	N*2 个字节	每两字节表示一个寄存器值，高位在前低位在后。寄存器地址小的排在前面
5	CRC 校验	0xba 7a	2 个字节	高位在前，低位在后

若是主站发送的通讯帧错误，或操作失败，就发送错误响应帧，反馈给主站：

错误响应帧：从机地址+（功能码+0x80）+错误码+CRC 校验。

序号	数据(字节)意义	字节数量	说明
0	帧头	3.5 个字符空闲时间	
1	从机地址	1 个字节	取值 1~247，由 D8121 设定
2	功能码+0x80	1 个字节	错误功能码
3	错误码	1 个字节	1~4
4	CRC 校验	2 个字节	高位在前，低位在后
5	END	3.5 个字符空闲时间	

2、PLC 编程时需关注如下信息：

从机地址：主站发送帧中，该地址表示目标接收从机的地址；从机应答帧中，表示本机地址；从机地址的设定范围为 1~247，0 为广播通信地址。

操作类型：表示读或写操作；0x1=读线圈操作；0x03=读寄存器操作；0x05=改写线圈操作；0x06=改写寄存器操作。对于变频器而言，只支持 0x03 读、0x06 写的操作。

寄存器起始地址：表示对从机中要访问的寄存器地址，对于 MD280、MD320 系列变频器的访问时，对应的就是“功能码号”、“命令地址”、“运行参数地址”；

数据个数：即从“寄存器起始地址”开始要连续访问的数据个数，对于寄存器变量，以 word 为单位。

寄存器参数（数据）：要改写的数据（主机改写），或读取的数据（从机应答）；

校验和：本帧数据的 CRC 校验和，plc 中会自动计算处理，用户不需关注。

通讯过程中，难免有通讯出错或失败的情况，系统软件中提供的特殊变量 M8063、D8063 用于报告该故障信息，若 M8063 置位，表明出现通讯故障，用户读取 D8063 的内容，即可得知故障的原因。

在 LX 系列 PLC 的系统软件内已封装了 MODBUS 协议，包括 MODBUS-RTU 主站和从站、MODBUS-ASCII 主站和从站，可应用于 COM2 通讯口，只需给系统寄存器 D8126 设置相应的数值就可使用了。

7.1.1 modbus 主站通讯运用

1、MODBUS 主站通讯应用

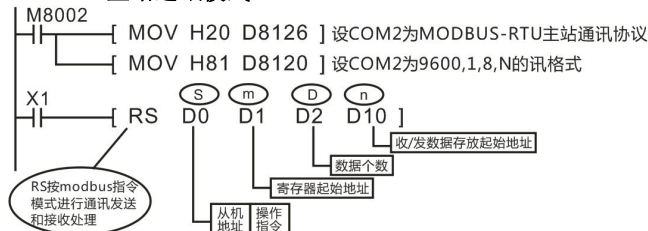
在 LX 系列 PLC 的 COM2 通讯口可使用 MODBUS-RTU 和 MODBUS-ASCII 等通讯协议，只需给系统寄存器 D8126 设置相应的数值就可使用，用 RS 指令实现的 MODBUS 的通讯编程。

关于通讯协议通讯参数的设置如下表所示：

COM2 口的通信配置		
协议	D8126 数值设定	通信参数设定
HMI 监控协议（PLC 协议）	01H	由 D8120 设定
MODBUS-RTU 从站	02H	由 D8120 设定
MODBUS-ASCII 从站	03H	由 D8120 设定
N: N 网络通信协议从站 （仅 LX3VP 系列 PLC 支持）	04H	由 D8120 设定
RS 指令	10H	由 D8120 设定
MODBUS-RTU 主站	20H	由 D8120 设定
MODBUS-ASCII 主站	30H	由 D8120 设定
N: N 网络通信协议主站 （仅 LX3VP 系列 PLC 支持）	40H	由 D8120 设定

D8120 通信参数定义			
位号	名称	内容	
		0 (OFF)	1 (ON)
B0	数据长度	7 位	8 位
B2B1	奇偶性	00: 无校验 01: 奇校验(ODD) 11: 偶校验(EVEN)	
B3	停止位	1 位	2 位
B7B6B5B4	波特率 (bps)	0111: 4800 1000: 9600 1001: 19200 1010: 38400 1011: 57600 1100: 115200	

2、Modbus 主站通讯模式



其中 RS（MODBUS 模式）指令中的各操作数定义与标准的 RS 指令（自定义协议）定义不同，分别为：

- ⑤为从机地址（高字节）、通讯命令（低字节，按 MODBUS 协议定义）；
- ④为访问从站的寄存器起始地址号；
- ③欲读或写的数据长度，单位为 word；
- ②为读或写数据的存放单元起始地址，占用后续地址单元，长度由③决定。

编程时，在每个 RS（MODBUS 模式）指令的前面，要根据要进行的通讯操作对象地址、操作类型、操作寄存器地址、数据个数、发送或接收的单元等各操作数单元赋值完毕，一旦开始执行，系统程序会自动计算 CRC 校验，组织通讯帧，完成发送数据、接收应答操作。

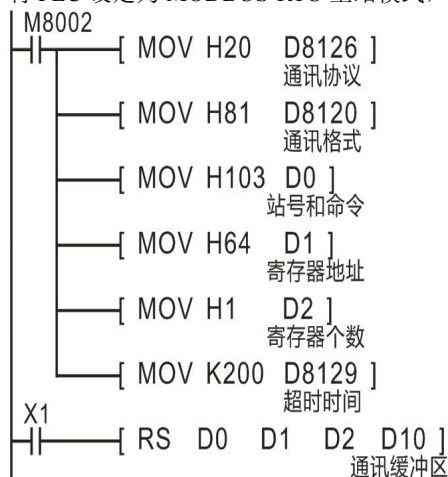
在 LX 系列的 plc 程序中，若有多个 RS（MODBUS 模式）指令被驱动，系统程序在执行时，仍是将一个 RS 指令的“发送、等待回答、接收、校验解析存放”等环节进行完毕后，再对下一个 RS 指令的同样处理，直到所有 RS 指令执行完毕，重新开始，用户无需关心其执行的时序和过程，简化了 PLC 编程设计，这是 MODBUS 中 RS 指令的优点所在。

3、MODBUS 主站功能一览表

功能码	功能名	详细内容
0x01	线圈读出	线圈读出（可以多点）
0x02	输入读出	输入读出（可以多点）
0x03	保持寄存器读出	保持寄存器读出（可以多点）
0x04	输入寄存器读出	输入寄存器读出（可以多点）
0x05	1 线圈写入	线圈写入（仅 1 点）
0x06	1 寄存器写入	保持寄存器写入（仅 1 点）
0x0F	批量线圈写入	多点的线圈写入
0x10	批量寄存器写入	多点的保持寄存器写入

4、编程案例

将 PLC 设定为 MODBUS-RTU 主站模式，不断的读取从机地址为 100 的寄存器，并将读取的数据存于 D10 中。



注：主机需从 STOP→RUN 时才有效！

执行结果：开机后，PLC 不断读从机地址为 100 的寄存器，通过 COM2 发送以下一帧数据（16 进制）：

01 03 00 64 00 01 C5 D5

01：代表从机地址，D0 的高 8 位；

03：MODBUS 命令码，D0 的低 8 位，意义为读从机寄存器；

00 64：所要读取的从机寄存器地址，D1 的值；

00 01：所要读取的寄存器个数，D2 的值，

C5 D5：CRC 校验码。

案例解析：当 X1=ON 时，读取站号为 01 的设备中地址为 100 的数据，并将该读取的数据存放在 D10 寄存器中。

用户程序中，循环执行的 RS（MODBUS）指令越少，通讯数据的更新就越频繁，读数刷新速度就越快，提高了实时性，合理安排一些不重要参数的读取频度，可以改善通讯效果。

利用特殊变量 M8129，还可判断通讯超时故障，就可作相应的保护或告警处理。

若使用 MODBUS-ASCII 协议通讯（将 D8126 设定为 H30），其中收发数据的 HEX-ASC 格式变换由 PLC 系统程序自动完成，用户使用 RS（MODBUS 模式）指令的方法与使用 MODBUS-RTU 协议的方法完全相同。

7.1.2 modbus 从站通讯运用

1、MODBUS 从站通讯应用

有些工业应用中，PLC 控制器作为工业自动化系统的一部分，要接受自动化控制网络的监控，典型的上位机有如 DCS、运行组 2 态软件的工业 PC 等，作为监控主机，以 MODBUS 主站协议与 PLC 等设备进行通讯，此时 PLC 的通讯口需要以 MODBUS 从站协议与上位机通讯。LX 系列 PLC 内置有 MODBUS-RTU 从站协议和 MODBUS-ASCII 从站协议，并且只有 com2 口可以支持该协议运行。

2、MODBUS 从站协议的相关寄存器

COM2 口的通信配置		
协议	D8126 数值设定	通信参数设定
HMI 监控协议（PLC 协议）	01H	由 D8120 设定
MODBUS-RTU 从站	02H	由 D8120 设定
MODBUS-ASCII 从站	03H	由 D8120 设定
N: N 网络通信协议从站 (仅 LX3VP 系列 PLC 支持)	04H	由 D8120 设定
RS 指令	10H	由 D8120 设定

位编号	名称	内容	
		0 (off)	1 (on)
B0	数据长度	7 位	8 位
B1B2	奇偶校验	00: 无校验 01: 奇校验(ODD)	11: 偶校验 (EVEN)
B3	停止位	1 位	2 位
B4B5B6B	波特率(bps)	0111: 4800 1000: 9600 1010: 38400 1011: 57600	1001: 19200 1100: 115200
B8	STX	无	由 D8124 设置
B9	ETX	无	由 D8125 设置

PLC 程序中，将上述几个寄存器配置完毕，当相应通讯口有 MODBUS 主站发送给本机地址的通讯帧时，PLC 系统程序即会根据通讯要求，自动组织 MODBUS 通讯帧进行应答，无需用户程序的参与。

3、MODBUS 从站支持的操作

LX 系列 plc 作为 MODBUS 从站时，支持 MODBUS 的 0x01, 0x03, 0x05, 0x06, 0x0f, 0x10 等通讯操作命令；通过这些命令，可读写 PLC 的线圈有 M, S, T, C, X（只读），Y 等变量；寄存器变量有 D, T, C。

MODBUS 从站功能一览表：

功能码	功能名	详细内容
0x01	线圈读出	线圈读出（可以多点）
0x02	输入读出	输入读出（可以多点）
0x03	保持寄存器读出	保持寄存器读出（可以多点）
0x04	输入寄存器读出	输入寄存器读出（可以多点）
0x05	1 线圈写入	线圈写入（仅 1 点）
0x06	1 寄存器写入	保持寄存器写入（仅 1 点）
0x0F	批量线圈写入	多点的线圈写入
0x10	批量寄存器写入	多点的保持寄存器写入

MODBUS 通讯主机在访问（读取或改写）PLC 从机的内部变量时，必须遵循如下的通讯命令帧定义，以及变量地址索引方法，才能进行正常的通讯操作。

3.1 命令码 0x01（01）：读线圈

请求帧格式：从机地址+0x01+线圈起始地址+线圈数量+CRC 检验

序号	数据(字节)意义	字节数量	说明
1	从机地址	1 个字节	取值 1~247，由 D8121 设定
2	0x01（命令码）	1 个字节	读线圈
3	线圈起始地址	2 个字节	高位在前，低位在后，见线圈编址
4	线圈数量	2 个字节	高位在前，低位在后（N）
5	CRC 校验	2 个字节	高位在前，低位在后

响应帧格式：从机地址+0x01+字节数+线圈状态+CRC 检验

序号	数据(字节)意义	字节数量	说明
1	从机地址	1 个字节	取值 1~247，由 D8121 设定
2	0x01（命令码）	1 个字节	读线圈
3	字节数	1 个字节	值： $[(N+7)/8]$
4	线圈状态	$[(N+7)/8]$ 个字节	每 8 个线圈合为一个字节，最后一个若不足 8 位，未定义部分填 0。前 8 个线圈在第一个字节，最地址最小的线圈在最低位。依次类推
5	CRC 校验	2 个字节	高位在前，低位在后

错误响应：见错误响应帧

3.2 命令码 0x03（03）：读寄存器

请求帧格式：从机地址+0x03+寄存器起始地址+寄存器数量+CRC 检验

序号	数据(字节)意义	字节数量	说明
1	从机地址	1 个字节	取值 1~247，由 D8121 设定
2	0x03（命令码）	1 个字节	读寄存器
3	寄存器起始地址	2 个字节	高位在前，低位在后，见寄存器编址
4	寄存器数量	2 个字节	高位在前，低位在后（N）
5	CRC 校验	2 个字节	高位在前，低位在后

响应帧格式：从机地址+0x03+字节数+寄存器值+CRC 检验

序号	数据(字节)意义	字节数量	说明
1	从机地址	1 个字节	取值 1~247，由 D8121 设定
2	0x03（命令码）	1 个字节	读寄存器
3	字节数	1 个字节	值： $N*2$
4	寄存器值	$N*2$ 个字节	每两字节表示一个寄存器值，高位在前低位在后。寄存器地址小的排在前面
5	CRC 校验	2 个字节	高位在前，低位在后

错误响应：见错误响应帧

3.3 命令码 0x05（05）：写单线圈

请求帧格式：从机地址+0x05+线圈地址+线圈状态+CRC 检验

序号	数据(字节)意义	字节数量	说明
1	从机地址	1 个字节	取值 1~247，由 D8121 设定
2	0x05（命令码）	1 个字节	写单线圈
3	线圈地址	2 个字节	高位在前，低位在后，见线圈编址
4	线圈状态	2 个字节	高位在前，低位在后。非 0 即为有效
5	CRC 校验	2 个字节	高位在前，低位在后

响应帧格式：从机地址+0x05+线圈地址+线圈状态+CRC 检验

序号	数据(字节)意义	字节数量	说明
1	从机地址	1 个字节	取值 1~247，由 D8121 设定
2	0x05（命令码）	1 个字节	写单线圈
3	线圈地址	2 个字节	高位在前，低位在后，见线圈编址
4	线圈状态	2 个字节	高位在前，低位在后。非 0 即为有效
5	CRC 校验	2 个字节	高位在前，低位在后

错误响应：见错误响应帧

3.4 命令码 0x06（06）：写单个寄存器

请求帧格式：从机地址+0x06+寄存器地址+寄存器值+CRC 检验

序号	数据(字节)意义	字节数量	说明
1	从机地址	1 个字节	取值 1~247，由 D8121 设定
2	0x06（命令码）	1 个字节	写单寄存器
3	寄存器地址	2 个字节	高位在前，低位在后，见寄存器值编址
4	寄存器值	2 个字节	高位在前，低位在后。非 0 即为有效
5	CRC 校验	2 个字节	高位在前，低位在后

响应帧格式：从机地址+0x06+寄存器地址+寄存器值+CRC 检验。

序号	数据(字节)意义	字节数量	说明
1	从机地址	1 个字节	取值 1~247, 由 D8121 设定
2	0x06 (命令码)	1 个字节	写单寄存器
3	寄存器地址	2 个字节	高位在前, 低位在后, 见寄存器编址
4	寄存器值	2 个字节	高位在前, 低位在后。非 0 即为有效
5	CRC 校验	2 个字节	高位在前, 低位在后

3.5 命令码 0x0f (15) : 写多个线圈

请求帧格式: 从机地址+0x0f+线圈起始地址+线圈数量+字节数+线圈状态+CRC 校验。

序号	数据(字节)意义	字节数量	说明
1	从机地址	1 个字节	取值 1~247, 由 D8121 设定
2	0x0f (命令码)	1 个字节	写多个单线圈
3	线圈起始地址	2 个字节	高位在前, 低位在后, 见线圈编址
4	线圈数量	2 个字节	高位在前, 低位在后。N, 最大为 1968
5	字节数	1 个字节	值: 值: $\lceil (N+7)/8 \rceil$
6	线圈状态	$\lceil (N+7)/8 \rceil$ 个字节	每 8 个线圈合为一个字节, 最后一个若不足 8 位, 未定义部分填 0。前 8 个线圈在第一个字节, 最地址最小的线圈在最低位。依次类推
7	CRC 校验	2 个字节	高位在前, 低位在后

响应帧格式: 从机地址+0x0f+线圈起始地址+线圈数量+CRC 校验

序号	数据(字节)意义	字节数量	说明
1	从机地址	1 个字节	取值 1~247, 由 D8121 设定
2	0x0f (命令码)	1 个字节	写多个单线圈
3	线圈起始地址	2 个字节	高位在前, 低位在后, 见线圈编址
4	线圈数量	2 个字节	高位在前, 低位在后。
5	CRC 校验	2 个字节	高位在前, 低位在后

错误响应: 见错误响应帧。

3.6 命令码 0x10 (16) : 写多个寄存器

请求帧格式: 从机地址+0x10+寄存器起始地址+寄存器数量+字节数+寄存器值+CRC 校验。

序号	数据(字节)意义	字节数量	说明
1	从机地址	1 个字节	取值 1~247, 由 D8121 设定
2	0x10 (命令码)	1 个字节	写多个寄存器
3	寄存器起始地址	2 个字节	高位在前, 低位在后, 见寄存器编址
4	寄存器数量	2 个字节	高位在前, 低位在后。N, 最大为 120
5	字节数	1 个字节	值: $N*2$
6	寄存器值	$N*2$ ($N*4$)	
7	CRC 校验	2 个字节	高位在前, 低位在后

响应帧格式: 从机地址+0x10+寄存器起始地址+寄存器数量+CRC 校验。

序号	数据(字节)意义	字节数量	说明
1	从机地址	1 个字节	取值 1~247, 由 D8121 设定
2	0x10 (命令码)	1 个字节	写多个寄存器
3	寄存器起始地址	2 个字节	高位在前, 低位在后, 见寄存器编址
4	寄存器数量	2 个字节	高位在前, 低位在后。N, 最大为 120
5	CRC 校验	2 个字节	高位在前, 低位在后

错误响应: 见错误响应帧。

3.7 错误响应帧

错误响应: 从机地址+ (命令码+0x80) +错误码+CRC 校验。

序号	数据(字节)意义	字节数量	说明
1	从机地址	1 个字节	取值 1~247, 由 D8121 设定
2	命令码+0x80	1 个字节	错误命令码
3	错误码	1 个字节	1~4
4	CRC 校验	2 个字节	高位在前, 低位在后

MODBUS 从站地址映射表

1、线圈地址映射表

线圈：指位变量，只有两种状态 0 和 1。在本 PLC 中包含 M, S, T, C, X, Y 等变量。

变量名称	起始地址	线圈数量	说明
M0~3071	0 (0)	3072	
M8000~M8256	0x1F40 (8000)	256	
S0~S999	0xE000 (57344)	1000	
T0~T256	0xF000 (61440)	256	
C0~C255	0xF400 (62464)	256	
X0~X255	0xF800 (63488)	256	
Y0~Y255	0xFC00 (64512)	256	

2、寄存器地址映射表

寄存器：指 16 位(字)或 32 位(双字)变量，在本 PLC 中，16 位变量包含 D, T, C0~199; 32 位变量为 C200~255。

变量名称	起始地址	寄存器数量	说明
D0~D8255	0 (0)	8256	
T0~T255	0xF000 (61440)	256	
C0~C199	0xF400 (62464)	200	
C200~C255	0xF700 (63232)	56	32 位寄存器

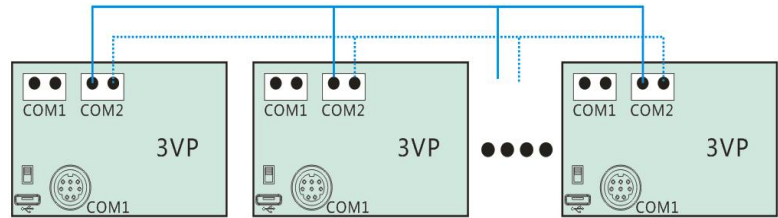
说明：通过 MODBUS 访问 C200~C255 段 32 位寄存器时，一个寄存器作两寄存器看待，一个 32 位寄存器占用两个 16 寄存器空间。比如用户要读或写 C205~C208 这 4 个寄存器，MODBUS 地址为 0xF 70A (0xF700+10)，寄存器数量 8 (4*2)。

32 位寄存器不支持写单个寄存器 (0x06) 命令码。

7.2 COM2 口 N：N 通信协议

1、N：N 通信协议模式

当一个设备中有多台（2~8 台）PLC 需要互相交换信息，协调地工作，可采用 PLC 内置的 N：N 网络协议，实现 PLC 之间的多方通讯。硬件上只需用双绞线将所有 LX3VP 系列 PLC 中 COM2 端口的 RS485 信号端对应并联即可组成通讯网络。模式如下所示：



2、N：N 通信功能描述

PLC 运行时，即可实现多台 PLC 间互相交换数据，用户程序在本 PLC 内部特定的数据区可读取到其他 PLC 的发送状态数据；本机用户程序将需要广播的数据复制到特定数据单元，便可让其他 PLC 读取。根据所需的通讯交互的信息量与通讯刷新速度两个指标要求，有五种模式可供选择，对应各自的变量区域定义如下：

站号		模式 0		模式 1		模式 2		模式 3		模式 4	
		位软元件 (M)	字软元件 (D)	位软元件 (M)	字软元件 (D)	位软元件 (M)	字软元件 (D)	位软元件 (M)	字软元件 (D)	位软元件 (M)	字软元件 (D)
		0 点	各站 4 点	各站 32 点	各站 4 点	各站 64 点	各站 8 点	各站 64 点	各站 16 点	各站 64 点	各站 32 点
主站	站号 0	——	D00— D03	M1000— M1031	D00— D03	M1000— M1063	D00— D07	M1000— M1063	D00— D15	M1000— M1063	D00— D31
	站号 1	——	D32— D35	M1064— M1095	D32— D35	M1064— M1127	D32— D39	M1064— M1127	D32— D47	M1064— M1127	D32— D63
	站号 2	——	D64— D67	M1128— M1159	D64— D67	M1128— M1191	D64— D71	M1128— M1191	D64— D79	M1128— M1191	D64— D95
	站号 3	——	D96— D99	M1192— M1223	D96— D99	M1192— M1255	D96— D103	M1192— M1255	D96— D111	M1192— M1255	D96— D127
	站号 4	——	D128— D131	M1256— M1287	D128— D131	M1256— M1319	D128— D135	M1256— M1319	D128— D143	M1256— M1319	D128— D159
	站号 5	——	D160— D163	M1320— M1351	D160— D163	M1320— M1383	D160— D167	M1320— M1383	D160— D175	M1320— M1383	D160— D191
	站号 6	——	D192— D195	M1384— M1415	D192— D195	M1384— M1447	D192— D199	M1384— M1447	D192— D207	M1384— M1447	D192— D223
	站号 7	——	D224— D227	M1448— M1479	D224— D227	M1448— M1511	D224— D231	M1448— M1511	D224— D239	M1448— M1511	D224— D255

数据链接最多可在 8 台可编程控制器之间自动更新；数据的链接如下图所示(图示为 3 台可编程控制器的数据交互情况)：



3、特殊地址

特殊地址	说明
D8120	通信格式的设置
D8122	通信格式的设置 NN 协议发送间隔。主从站都需要设置。建议设置为所有站中扫描周期最长的 PLC 中最大扫描周期的 1.5 倍。默认值 10，单位 100us。（注意：3VP: 2510716000、3VE: 2520616000、3VM: 2530416000、3VT: 2550016001 及以上版本有效）
D8126	COM2 口通信协议的配置，设为 40h 代表 N: N 主站，设为 04h 代表 N: N 从站
D8176	站点号，范围 0~7，0 表示主站
D8177	从站点的总数，范围 1~7，仅主站需要设置
D8178	刷新范围（模式）设置，范围 0~5，仅主站需要设置
D8179	重试次数设定，主站必须设置
D8180	通信超时设置，单位为 10ms，主站必须设置
D8201	当前链接扫描时间
D8202	最大链接扫描时间
D8203	主站错误计数
D8204~D8210	从站错误计数
D8211	主站 N: N 错误码
D8212~D8218	从站 N: N 错误码
M8183	主站数据传输序列错误
M8183~M8190	通信出错标志，M8183 对应第 0 号站点（主站），M8184 对应第 1 号站点，依次类推，M8190 对应第 7 号站点；
M8191	正在执行数据传送

N: N 通信错误码（D8211~D8218）

错误码	说明
1	超过了监视时间，从站仍未来自主站的发送请求作出响应
2	对于主站的发送请求，其他的从站已经响应
3	参数数据中的计数值与从站响应的计数值不一致
4	从站作出响应的报文不正确
17	超过了监视时间，主站仍未对下一从站发出请求
20	来自主站的报文不正确
33	从站无响应错误
34	对于主站的发送请求，其他的从站已经响应-
35	参数数据中的计数值与从站响应的计数值不一致
49	在尚未接收参数的状态下，已接收到来自主站的发送请求

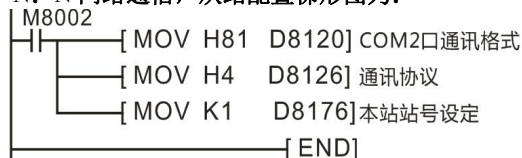
4、功能案例

设置 LX3VP 中 COM2 通信口使用串口参数为（9600，1，8，NONE）。使用模式 3 刷新范围，使两台 PLC 进行 N: N 通信为例。

1) N: N 网络通信，主站配置梯形图为：



2) N: N 网络通信，从站配置梯形图为：



5、注意

LX3VP 系列 PLC 中 COM2 端口的 N: N 通信协议，不能与 LX3V-2RS485-BD 模块的 N: N 功能同时使用，如果 COM2 和 485BD 同时配置为 N: N 功能，则以 PLC 中 COM2 端口优先执行，LX3V-2RS485-BD 模块的 N: N 功能则无效。即一台 PLC 主机（包括负载模块的情况下）在同一时间内只能存在一个 N: N 通信协议。



好团队·好产品·助力智能制造

福州富昌维控电子科技有限公司

FUZHOU FUCHANG WECON E AND T LTD

网站: www.we-con.com.cn

Email: info@we-con.com.cn

电话: 400-799-8189

传真: 0591-87843899

福建省福州市软件园E区10号楼 (生产总部)

福建省福州市软件园F区6号楼5-6层 (管理和研发中心)



培训视频



公众号



抖音