

V5 系列可编程逻辑控制器指令及编程手册

版本变更记录

版本	发布时间	更新内容
A1.0	2019.08.28	第一次发布

第1章 前言	7
第2章 概要	10
2.1 可编程控制器简介	10
2.2 基本控制原理	11
2.2.1 可编程控制器工作原理	11
2.2.2 用户程序控制原理	12
2.3 编程软件	12
第3章 软元件	21
3.1 软元件一览表	22
3.2 输入输出继电器	24
3.2.1 输入继电器 X	24
3.2.2 输出继电器 Y	24
3.3 辅助继电器 M	24
3.4 状态继电器 S	25
3.5 定时器 T	25
3.6 计数器 C	27
3.6.1 16bit 计数器	28
3.6.2 32bit 计数器	29
3.6.3 高数计数器	29
3.7 寄存器	30
3.7.1 数据寄存器 D	30
3.7.2 变址寄存器 V、Z	31
3.7.3 文件寄存器 R	32
3.8 标号与子程序	33
3.9 常数	34
第4章 指令	35
4.1 程序逻辑指令	35
4.1.1 触点指令	36
4.1.2 结合指令	45
4.1.3 输出指令	47
4.1.4 其他处理指令	53
4.2 程序流程指令	54
4.2.1 子程序	54
4.2.2 中断	56
4.2.3 跳转	57
4.2.4 循环	59
4.3 数据比较指令	62
4.3.1 触点比较	64
4.3.2 输出比较	79
4.4 数据运算	83
4.4.1 四则运算	84
4.4.2 数据逻辑运算	95

4.4.3 三角函数.....	100
4.4.4 表格运算.....	111
4.4.5 指数运算.....	120
4.5 数据处理	125
4.5.1 数据转换.....	126
4.5.2 数据传送.....	143
4.5.3 表格操作.....	150
4.5.4 数据移位.....	151
4.5.5 其他数据处理.....	155
4.6 时钟指令	160
4.7 脉冲定位.通讯定位指令	163
4.7.1 脉冲输出.....	163
4.7.2 脉冲定位.....	163
4.7.3 刷新处理.....	164
4.8 通讯.....	166
4.9 外设.....	168
4.9.1 PID 运算	168
4.9.2 其他外设指令.....	172
4.10 电子凸轮指令.....	174
第5章 高速输出及定位指令.....	175
5.1 指令概述	175
5.1.1 高速输出指令属性表.....	175
5.1.2 脉冲输出端口说明.....	175
5.1.3 脉冲输出端口特殊软元件.....	176
5.1.4 输出频率及加减速时间.....	178
5.2 定位指令	178
5.2.1 定位指令一览.....	178
5.2.2 PLSY 脉冲输出指令	179
5.2.3 PLSV 可变脉冲输出指令	181
5.2.4 PLSV2 带加减速的可变脉冲输出指令	183
5.2.5 PLSR 带加减速脉冲输出	186
5.2.6 DRVA 绝对位置定位.....	191
5.2.7 DRVI 相对位置定位	196
5.2.8 ZRN 原点回归	201
5.2.9 DSZRDOG 搜索原点回归（开发中）	204
5.2.10 DVIT 中断定位（未开发）	209
5.2.11 DPIT 最大定长的中断定位指令.....	213
5.3 高速处理指令	217
5.3.1 PWM 脉宽调制输出指令	217
第6章 电子凸轮.....	218
6.1 电子凸轮（E-CAM）简介	218
6.2 电子凸轮的实现方式	219
6.2.1 第一步：初始设定	219

6.2.2 第二步：主轴选择/凸轮表选择	223
6.2.3 第三步：启动/停止电子凸轮	224
6.3 电子凸轮关键点修改	228
6.3.1 DCAMWR 电子凸轮数据修改	228
6.3.2 DCAMRD 读取电子凸轮数据(开发中)	230
6.3.3 全点修改指令	231
6.3.4 单数据修改指令 DCAMWR 指令	232
6.3.5 顶杆修改指令	234
6.4 飞剪	235
6.4.1 功能说明	235
6.4.2 应用范例	240
6.5 追剪	244
6.5.1 功能说明	244
6.5.2 应用范例	251
6.6 特殊功能	255
6.6.1 顶杆	255
6.6.2 主从速比计算	256
6.6.3 主轴区间位置监控	256
6.6.4 主轴累积脉冲监控	256
6.6.5 电子凸轮方向设定	257
6.6.6 编码器频率监控	257
6.6.7 主轴位置计算(开发中)	258
6.6.8 从轴位置计算	259
6.6.9 探针	260
6.6.10 运动叠加	261
6.6.11 控制周期	263
6.7 电子凸轮相关软元件一览	264
第7章 中断	266
7.1 概述	266
7.1.1 概述	266
7.1.2 中断类型	266
7.2 外部中断	266
7.2.1 概述	266
7.2.2 外部中断类型	267
7.2.3 使用举例	267
7.3 定时器中断	268
7.3.1 概述	268
7.3.2 定时器中断类型	268
7.3.3 使用举例	268
7.4 脉冲完成中断	269
7.4.1 概述	269
7.4.2 脉冲完成中断类型	269
7.4.3 使用举例	269

第8章 通信	271
8.1 概要	271
8.2 介绍	271
8.3 通信协议设置说明	273
8.3.1 COM0 协议配置	273
8.3.2 COM2 协议配置	275
8.3.3 串口通信格式	276
8.3.4 串口通信格式软元件一览表	277
8.3.5 通信错误码一览表	278
8.4 HMI 监控协议	279
8.5 MODBUS 协议	280
8.5.1 MODBUS 协议说明	280
8.5.2 MODBUS 功能码及数据编址	280
8.5.3 MODBUS 通信地址	286
8.5.4 MODBUS 的配置使用说明	287
8.5.5 MODBUS 指令使用说明	293
8.6 CANOPEN 通讯	298
8.6.1 概述	298
8.6.2 硬件配置	300
8.6.3 创建 CANopen 配置	302
8.6.4 主站配置	304
8.6.5 从站配置	307
8.6.6 CANopen 配置下载	312
8.6.7 在线调试	312
8.6.8 CAN 总线监控	314
8.6.9 CANopen 通讯故障代码及排除	315
8.6.10 CANopen 通讯变量	317
8.6.11 V5-CANopen 控制 SD700 伺服驱动器	318
第9章 子程序	324
9.1 概要	324
9.1.1 V5 子程序概述	324
9.1.2 V5 子程序执行机制	324
9.2 通用子程序应用	326
9.2.1 建立子程序	326
9.2.2 导出子程序	326
9.2.3 导入子程序	327
9.2.4 子程序属性	328
9.2.5 子程序调用	329
9.3 中断子程序应用	329
9.3.1 中断子程序属性	329
9.3.2 中断子程序调用	331
附录一 特殊软元件分配说明	335

SM 标志位分配.....	335
SD 寄存器分配.....	337
M8000 标志位、D8000 寄存器分配.....	340
附录二 系统错误码说明.....	352
系统错误码 D8061	352
系统错误码 D8062	352
系统错误码 D8063	353
系统错误码 D8064	355
系统错误码 D8065	355
系统错误码 D8066	355
系统错误码 D8067	355

第1章 前言

首先感谢您购买使用 V5 系列可编程控制器！V5 系列可编程控制器是结合全球领先的控制算法开发的一款全新产品。本产品支持点动运动、直线插补、圆弧、3D 圆弧、电子齿轮/凸轮等功能；内嵌大量方便行业功能块，如追剪、飞剪、防空等功能块，使应用变得非常简单；集成同步跟随、全闭环控制、多轴协作运动控制等功能块只需调用即可应用。

项目	V5-MC104
程序容量	64K
掉电存储容量	40K 字
基本指令速度	100ns
插补周期	125us~1ms
轴数	4+2 轴【1】
高速输入	4M（4 路）【2】，200k（4 路）
高速输出	3M（4 路）【3】，200k（2 路）
通用输入/输出	22 点数字量输入，14 点数字量输出
可编程	梯形图，凸轮指令，MC 指令，G 代码
规划模式	T/S 型，对称/非对称
运动模式	恒速，动态位置变化，速度变化，加速度变化，叠加运动
插补	6 轴直线插补、圆弧插补、3D 圆弧插补，螺旋线插补
连续轨迹	连续插补，变速，暂停
电子凸轮	6 轴电子凸轮，追剪，飞剪，电子齿轮，顶杆
探针	3 个，5us 响应时间
模拟量输入	2 路模拟量输入
通信	RS485（2 个），RS422（1 个），USB，Ethernet，CAN【4】
扩展性	最大支持扩展 6 个数字量输入输出本地扩展模块（16 入 16 出）

【1】可扩展。

【2】定义为 4 轴输入，作为手摇轮输入或者反馈输入，可接受差分输入。

【3】定义为 4 轴输出，每个轴包括 2 组差分输出，可作为 AB 相输出，CW/CCW 输出方式或者脉冲加方向方式。

【4】CAN 支持 CANopenDS301 以及 CANopenDS402 主站以及从站协议。

☞ 安全注意事项：

请由专业的操作人员按照正常步骤使用本产品，使用过程中注意以下安全相关的注意事项，否则可能导致对人体的伤害或财产损失。其中安全注意事项定义如下。

 **危险：** 如果操作错误，极有可能会导致死亡或重伤；

 **警告：** 如果操作错误，可能会导致死亡或重伤；

 **注意：** 如果操作错误，可能会导致中度伤害或受到轻伤，也有可能造成设备损坏的情况。

设计注意事项	
 危险	请在控制器的外部设置安全回路，以便在出现外部电源异常、控制器故障时，能确保整个系统在安全状态下运行。误动作、误输出有可能会发生。 请务必在控制器的外部设置紧急停止回路、保护回路、防止正反转等相反动作同时进行的互锁回路、定位上下限等防止机械破损的互锁回路等。 当控制器 CPU 通过看门狗定时器出错等的自诊断功能检测出异常时，断开所有的输出。此外，当为控制器 CPU 不能检测出输入输出控制部分等的异常时，输出控制会失效。此时，请设计外部回路及结构，以确保机械在安全状态下运行。 由于输出单元的继电器、晶体管、晶闸管等的故障，有时会导致输出一直接通，或是一直断开。为了确保机械在安全状态下运行，请为可能导致重大事故的输出信号设计外部回路以及结构。
 注意	控制线请勿与主回路或动力线捆在一起接线，或是靠近接线，原则上请离开 100mm 以上，否则会因噪音引起误动作。 使用时，请确保连接外围设备的连接器不受外力，否则会导致断线以及故障。

安装注意事项	
 危险	进行安装操作时，请务必在外部断开所有电源后方可进行操作，否则有触电危险。
 注意	请在本手册记载的一般规格的环境下使用。请勿在有灰尘、油烟、导电性灰尘、腐蚀性气体、可燃性气体的场所和暴露在高温、结露、风雨的场所，以及有振动和冲击的场所中使用。否则可能导致触电、火灾、误动作、产品损坏以及老化。 请勿直接接触产品的导电部位，否则可能导致误动作、故障。 在进行加工及配线作业时，请不要将切屑及电线屑落入控制器通风孔内，否则有可能导致火灾、故障及误动作。 产品安装时，请使用 DIN 导轨、请将产品安装在平整的表面上。 外围设备连接、输入输出等的连接电缆请牢固地安装在所规定的连接器上，避免接触不良会导致误动作。 本地扩展模块必须确保两侧锁扣锁上，否则可能导致接触不良引发误动作。

接线注意事项	
 危险	进行接线操作时，请务必在外部断开所有电源后方可进行操作，否则有触电、产品损坏的危险。
 注意	在进行加工及配线作业时，请不要将切屑及电线屑落入控制器通风孔内，否则有可能导致火灾、故障及误动作。 对欧式端子排型的产品进行接线时，请遵照以下的注意事项操作，否则有可能导致触电、故障、短路、断线、误动作、损坏产品。 绞线的末端要捻成没有金属丝发散，请勿对电线末端上锡； 请勿连接不符合规定尺寸的电线或是超出规定根数的电线。

启动维护保养	
 危险	通电时请勿触碰到端子，请务必在断开所有外部电源后清扫及插拔接线端子，否则有触电危险，并可能引起误动作。 要在运行过程中更改程序、执行强制输出、RUN、STOP 等操作前，请务必先熟读手册，在充分确认安全的情况下方可进行操作，否则可能导致机械破损及事故发生。 请勿从多个外围设备同时更改控制器中的程序，否则可能会破坏控制器的程序，引起误动作。
 注意	请勿擅自拆解、改动产品，否则有可能引起故障、误动作、火灾。 对扩展电缆等连接电缆进行拆装时请在断开电源之后在进行操作，否则可能引起故障、误动作。

废弃及运输	
 注意	废弃产品的时候，请作为工业废品来处理。对电池进行废弃处理时，请按照各地区指定的法律单独进行处理。 控制器属于精密设备，因此在运输期间请避免使其遭受一般规格值的冲击。否则可能造成控制器故障。运输之后，请对控制器进行动作确认。

第2章 概要

2.1 可编程控制器简介

可编程控制器 PLC (Programmable Logic Controller) 是一种数字运算操作的电子系统，专为工业环境下应用而设计。它主要将外部的输入信号如：按键、感应器、开关及脉冲等的状态读取后，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。

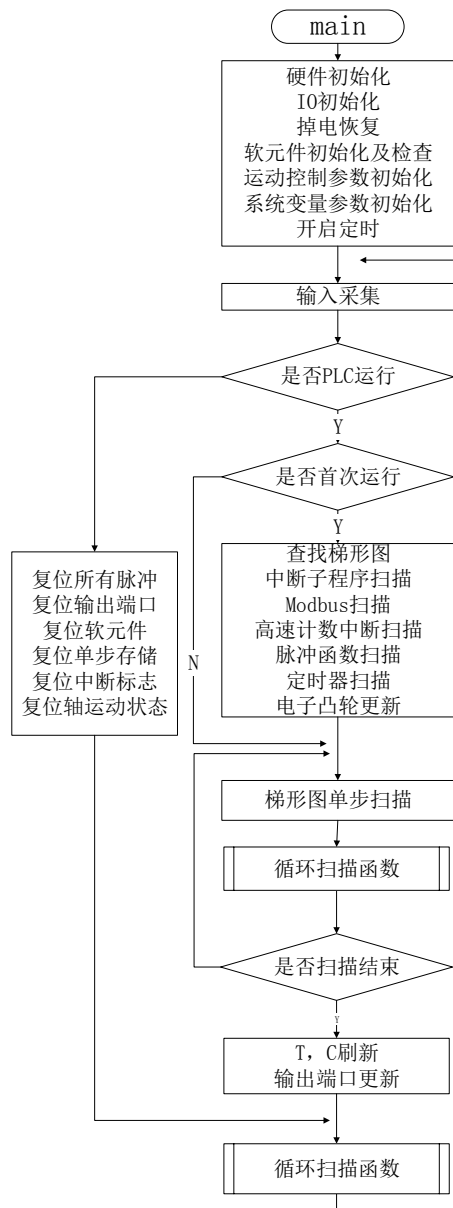
2.2 基本控制原理

2.2.1 可编程控制器工作原理

可编程控制器采用循环扫描工作方式，包含输入点扫描、用户程序执行、输出点刷新、内部处理和通讯处理几个部分。

在运行可编程控制器前，可以使用编程软件编写输入点和输出点间的控制逻辑并下载到可编程控制器中，在可编程控制器运行过程中，首先会扫描输入点信号，将之读取到可编程控制器中，然后根据控制程序完成运算和逻辑处理，运算和逻辑处理结果将输出点的值改变，最后将输出点中的值转换为电信号输出并控制各种机械设备的运行。

在可编程控制器运行过程中采用循环扫描的工作方式，通过反复的执行输入点扫描、用户程序执行、输出点刷新工作达到接收控制并操作设备的目的。



2.2.2 用户程序控制原理

在用户程序中，可编程控制器的输入点被称为触点，它的功能和工业设备中的开关触点是一样的，代表能流的导通或者关断。在可编程控制器中，输入点存储为一个软元件，当输入点为高电平时，对应的软元件为导通状态，在用户程序中参与逻辑运算并影响输出点的值；输出点称为线圈，代表输出能流的导通或者关断，输出点对应软元件的值由输入点和控制逻辑计算结果决定。在输出刷新时，软元件的值被转换为电信号在输出点的晶体管或者继电器输出，从而完成对设备的控制。

2.3 编程软件

V5 系列运动控制器采用 VCAutoDesignsoft 软件进行编程。

■ 运行环境要求：

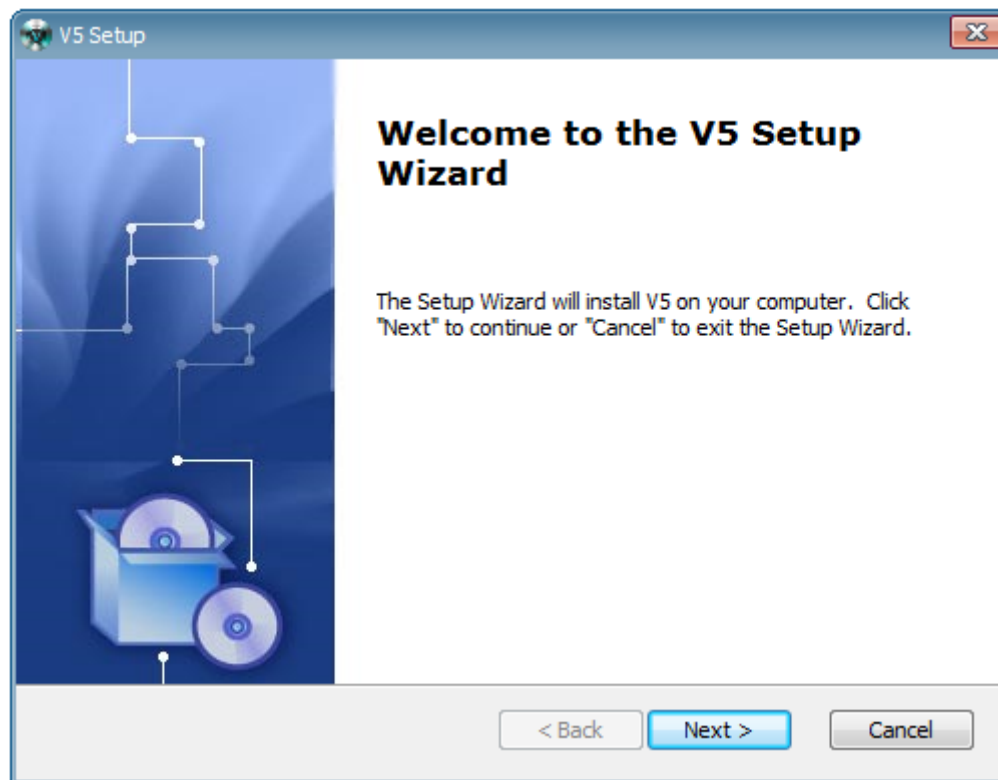
系统配置	要求
操作系统	WindowsXP、Windows2000、Windows7、Windows8、Windows10
系统类型	32 位、64 位
CPU	要求 600 兆赫(MHz)PentiumIII 处理器及以上
内存	最低 RAM 要求 1G
硬盘	需要 1G 以上的可用空间
显卡	SuperVGA(800x600)或更高分辨率的显示器
驱动	USB-SC09-FX 通讯线驱动程序
其他需求	安装 MicrosoftOffice2003 及以上

■ VCAutoDesignsoft 软件安装

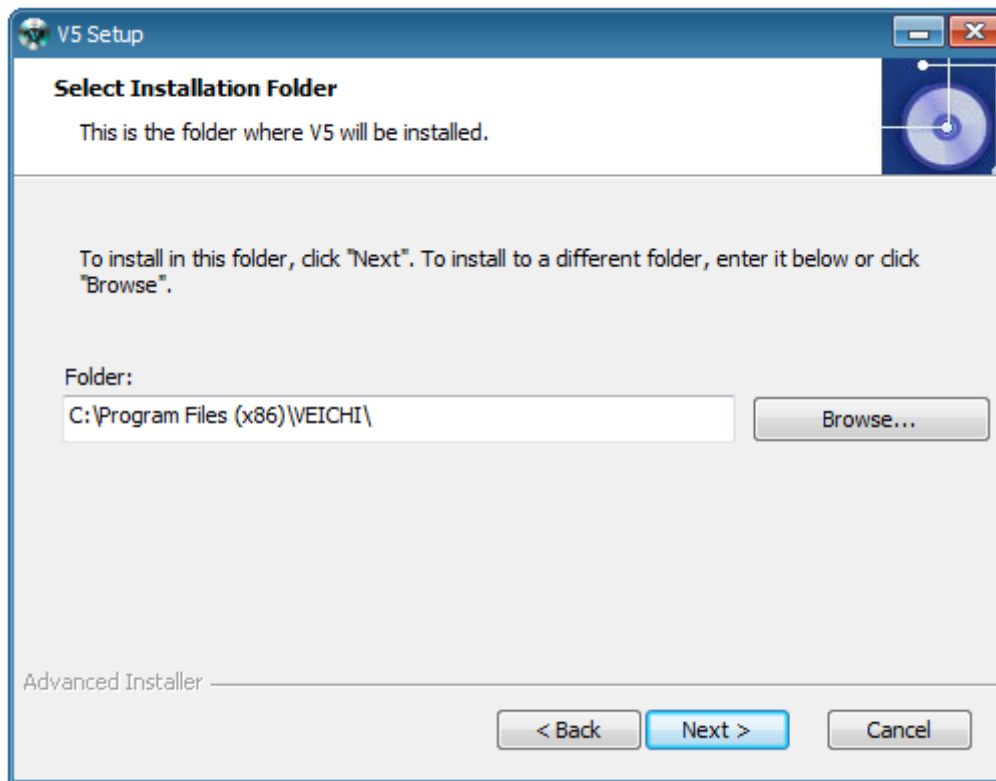
1、VEICHI 伟创电气官方网站下载最新版软件安装包 <https://www.veichi.com/>，下载得到安装包，如下图所示：



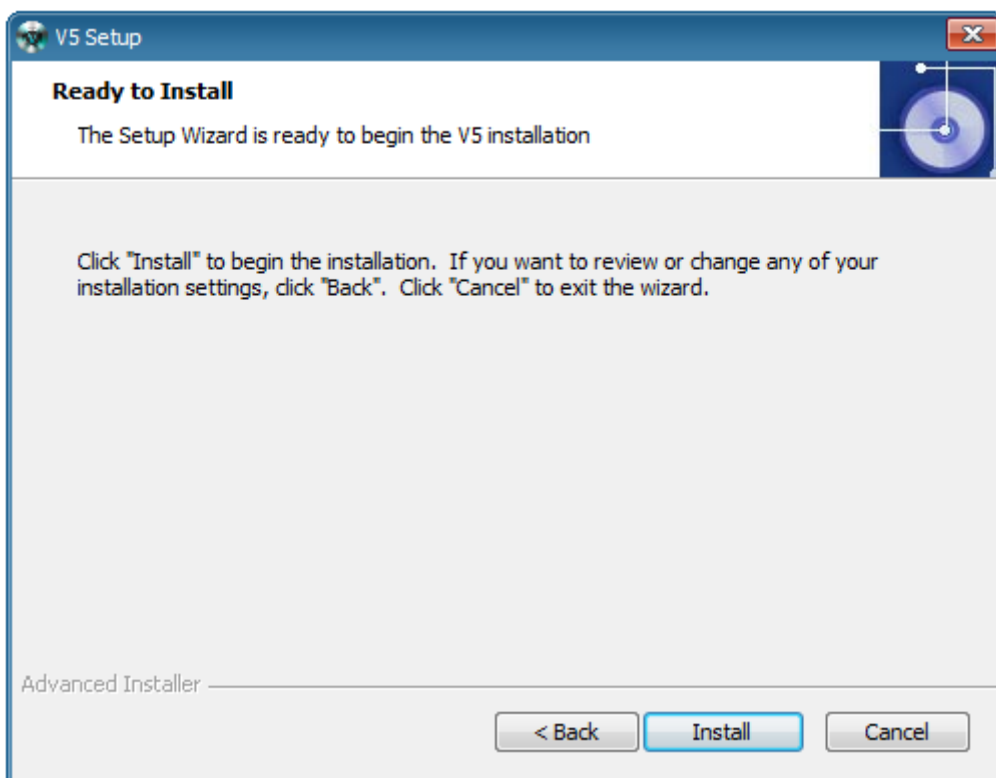
2、鼠标左键双击安装包，进入安装界面，如下图所示，点击“Next”按钮进入下一步；

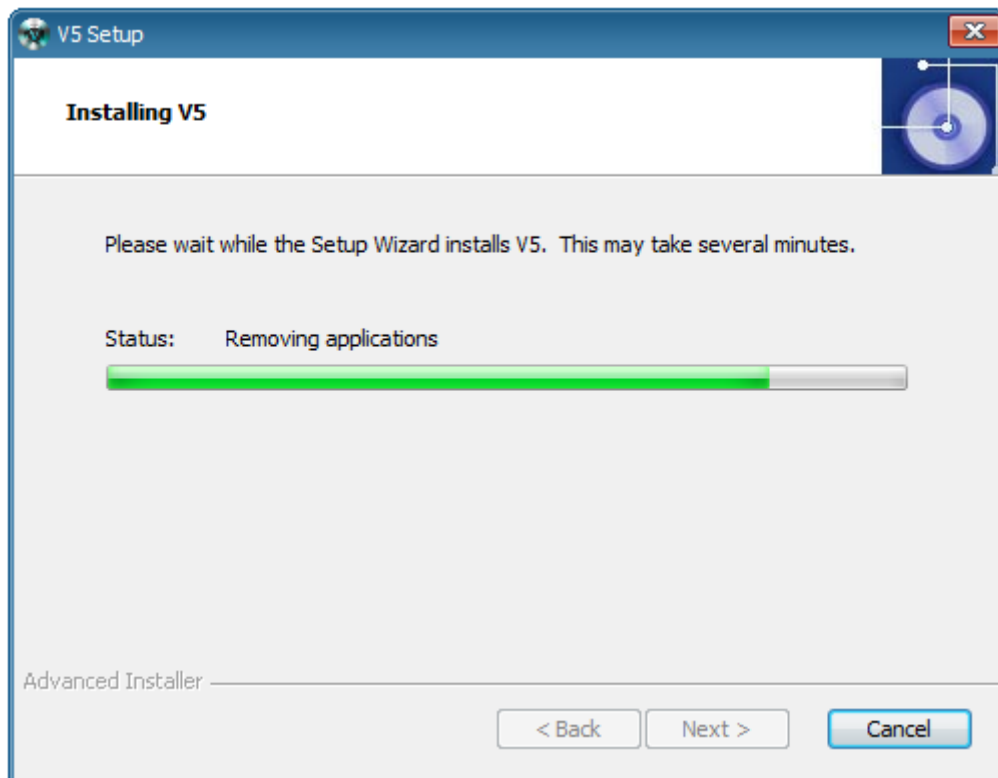


3、选择软件安装位置，点击“Next”按钮进入下一步，如下图所示：

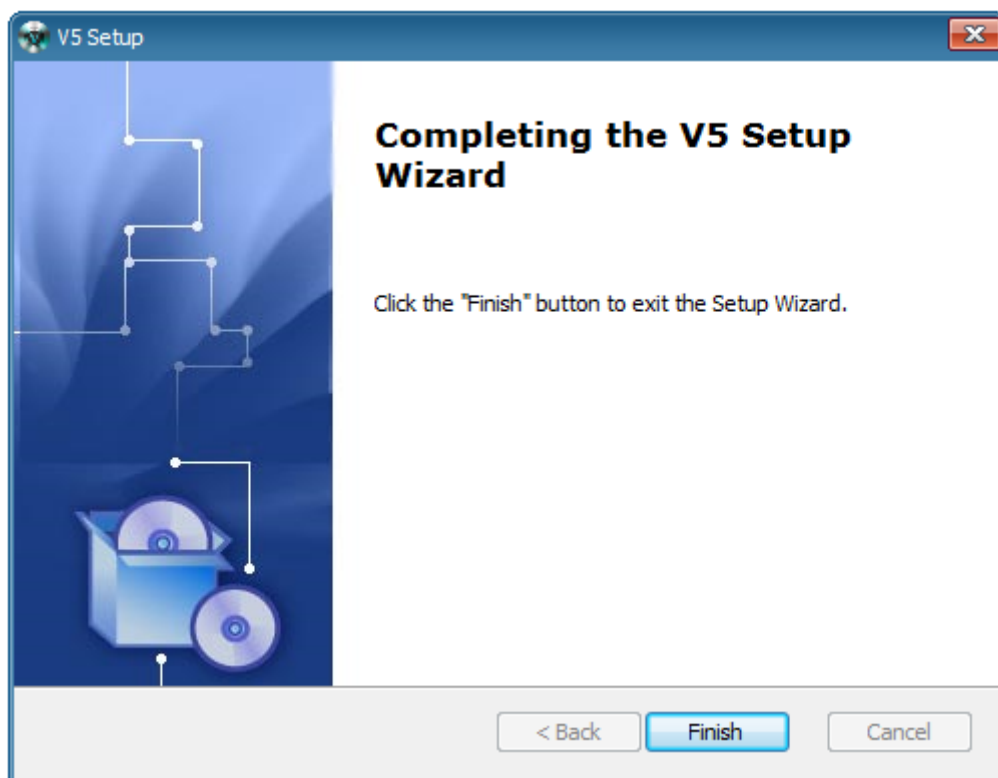


4、点击“Install”按钮，开始安装软件，等待安装完成，如下图所示：





5、安装完成，出现如下所示界面，点击“Finish”按钮；



6、在桌面找到自动生成快捷方式图标，双击即可打开软件开始编程。



■ USB-SC09-FX 通讯线驱动安装

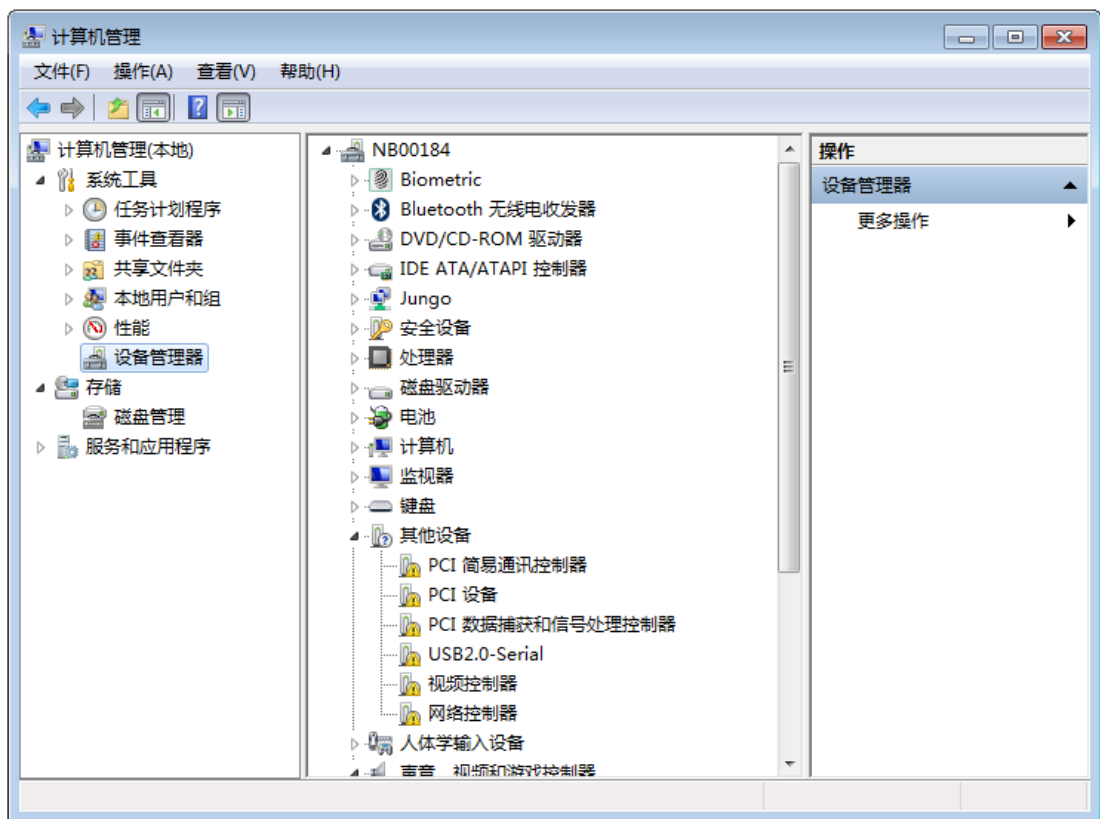
1、不同通讯卡厂家驱动程序有所不同，以 USB-SC09-FX 通讯线为例，取得该通讯线的驱动程序安装文件，如下图所示：



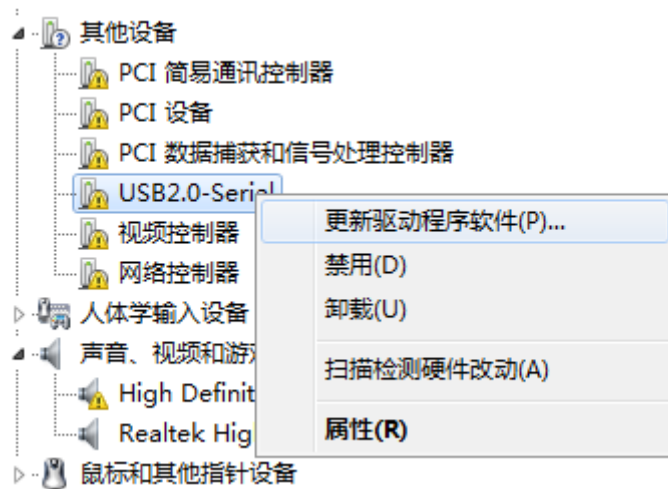
2、将 USB-SC09-FX 通讯线分别与后台软件电脑 USB 接口和 V5-MC104 控制器的 RS422 接口建立连接；

3、进入电脑的设备管理器：以 WIN7 为例（WindowXP 为“我的电脑”->“属性/设备管理器”），鼠标选中“计算机”，单击右键，选择“管理”，点击“设备管理器”，如下图所示：

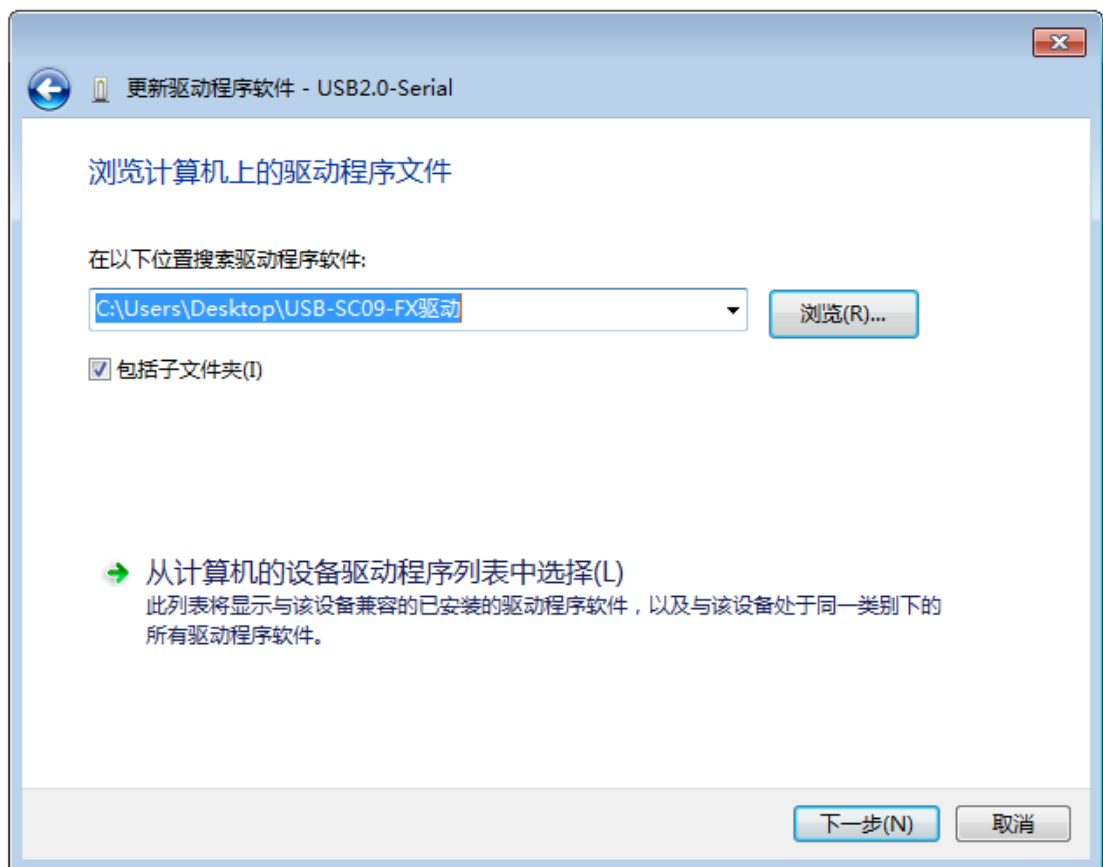
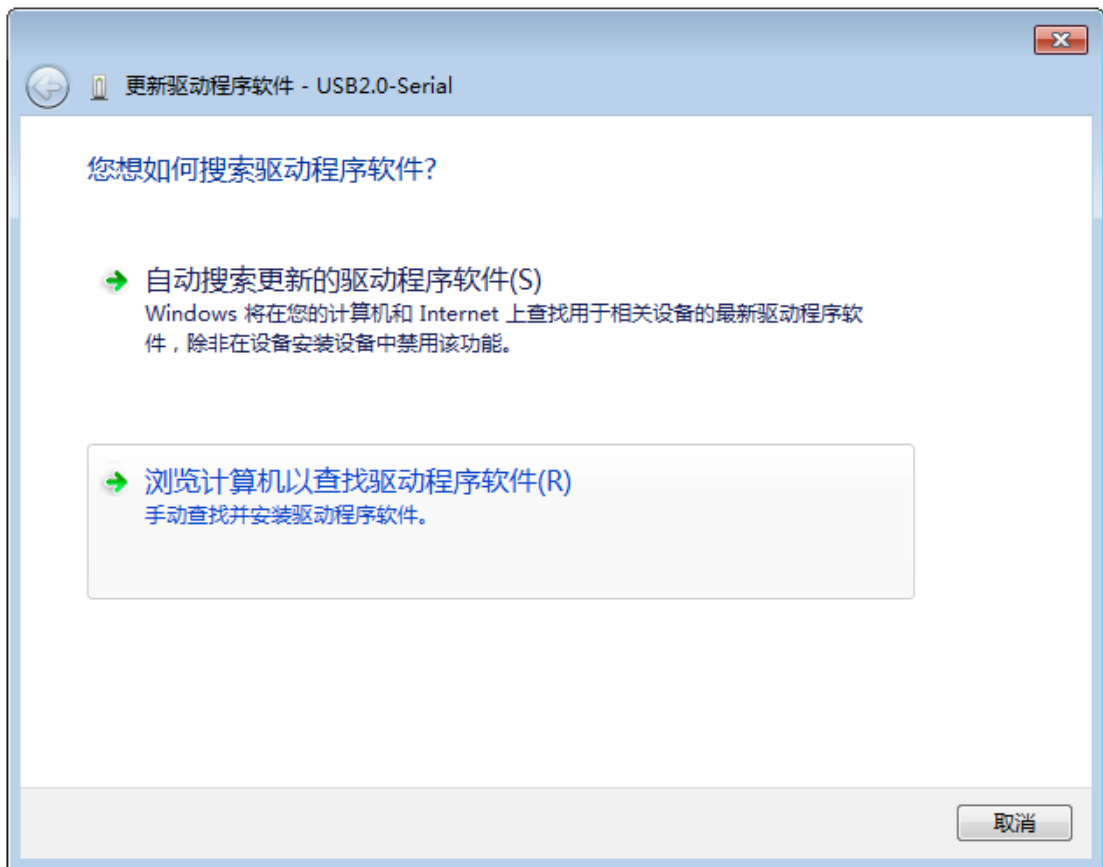




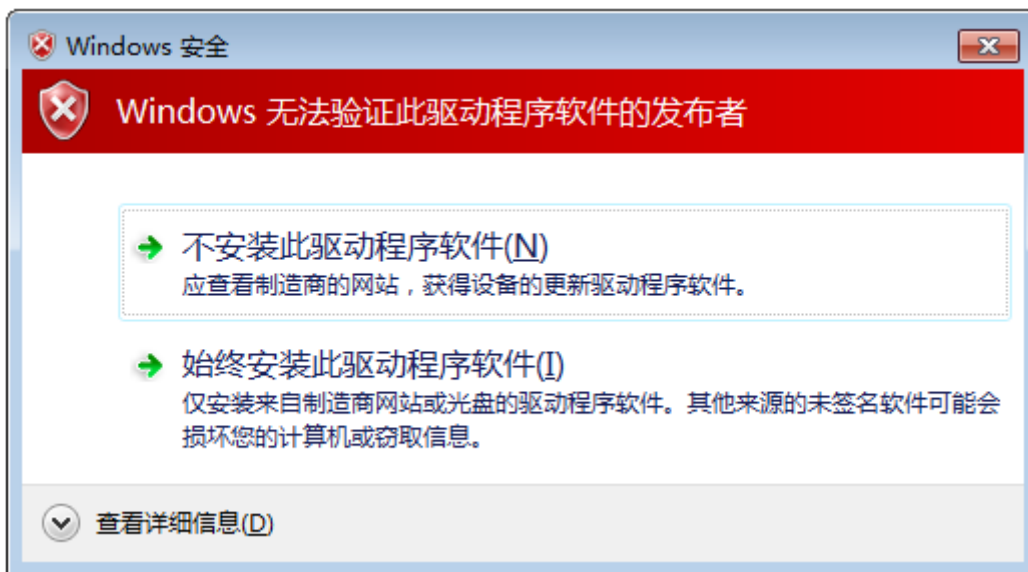
4、选中“其他设备”下的“USB2.0-Serial”，单击鼠标右键，点击“更新驱动程序软件”：



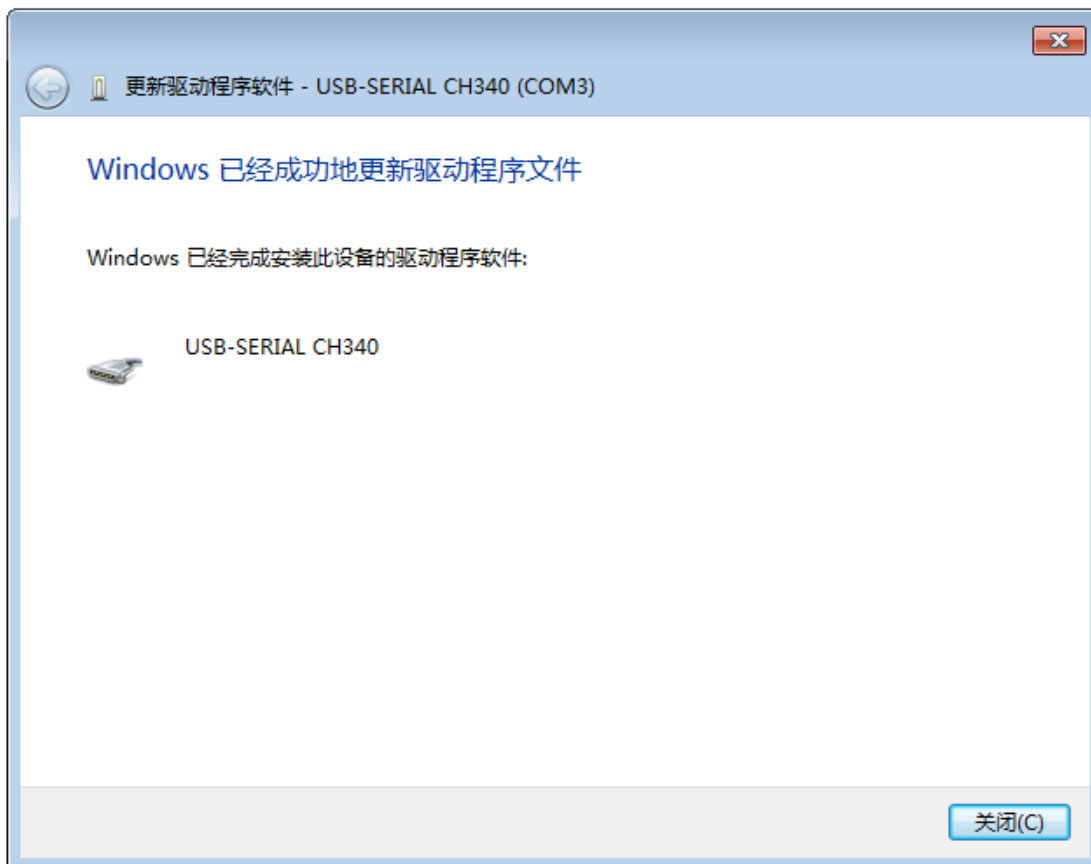
5、点击“浏览计算机以查找驱动程序软件”，点击“浏览”按钮，选择你的驱动程序所在的文件夹，比如 USB-SC09-FX 通讯线的驱动文件夹“C:\Users\Desktop\USB-SC09-FX 驱动”，如下图所示：

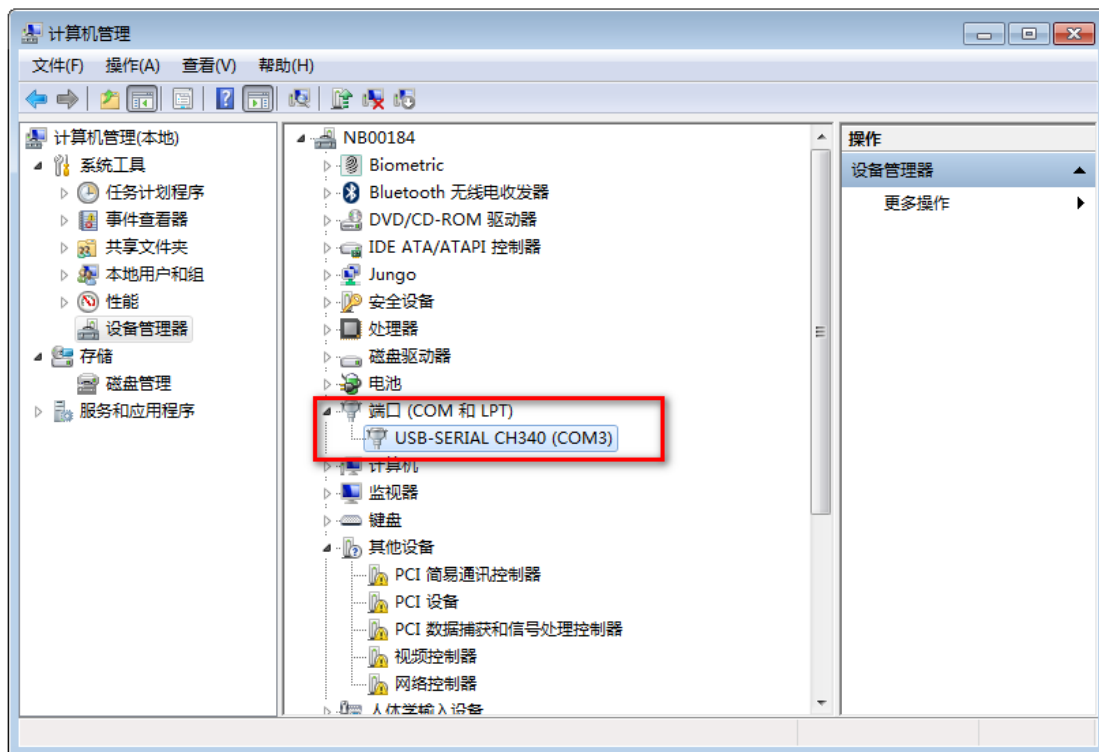


6、点击“下一步”按钮，按提示安装，如出现以下警告，点击“始终安装此驱动程序软件”即可：



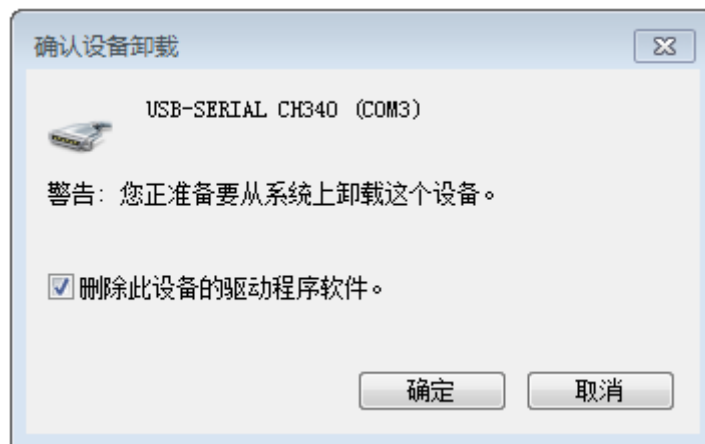
7、安装完驱动程序后，会提示“Windows 已经完成安装此设备的驱动程序软件”，点击关闭即可；在设备管理器的“端口（COM 和 LPT）”中会出现一个新的串口，如下图红色框中的“COM3”，这个串口就是 USB-SC09-FX 通讯线使用的串口号。以后每次插入和使用该通讯线就会出现该 COM 口，你只需在编程软件中选中该 COM 口即可通讯。





8、驱动程序卸载

卸载驱动程序是为了释放 COM 口资源以便其他设备使用，或驱动程序故障时需将其卸载后重新安装，按以下步骤卸载驱动程序：打开设备管理器，在“端口（COM 和 LPT）”节点中选中需要卸载的 COM 口，鼠标右键菜单选择“卸载”，选中“删除此设备的驱动程序软件”，点击“确定”按钮即可完成卸载。



第3章 软元件

系统软元件类型如下表：

序号	元件类型	功能与分类
1	输入输出继电器	对应 PLC 的硬件开关量输入的位元件 X
		对应 PLC 的控制输出的位元件 Y
2	辅助继电器	普通辅助继电器 M
		系统特殊辅助继电器 M
		系统特殊辅助继电器 SM
3	状态继电器	步进控制用状态标志位元件 S
4	定时器	1ms、10ms、100ms 步长的 16bit 定时器 T
5	计数器	16bit/32bit 增/减型计数器 C
		32bit 高速计数器 C
6	数据寄存器	普通数据寄存器 D
		系统特殊用数据寄存器 D
		系统特殊用数据寄存器 SD
		数据间接寻址寄存器 V、Z
7	文件寄存器	文件寄存器 R
8	标号	标号/跳转指针 P
9	子程序	子程序 SBR
		中断子程序 I
10	嵌套指针	嵌套指针 N
11	常数	十进制常数 K
		十六进制常数 H
		浮点数 E

3.1 软元件一览表

输入输出继电器			
输入继电器	X0～X377	256 点	软元件的编号为 8 进制编号，输入输出合计 512 点
输出继电器	Y0～Y377	256 点	
辅助继电器			
一般用	M0～M499	500 点	不支持掉电保存
保持用	M500～M1023	524 点	掉电保存
保持用	M1024～M7679	6656 点	掉电保存
系统特殊用	M8000～M8511	512 点	掉电保存
系统特殊用	SM0～SM1023	1024 点	掉电保存
状态继电器			
初始化状态	S0～S9	10 点	不支持掉电保存
一般用	S10～S499	490 点	
保持用	S500～S899	400 点	掉电保存
报警用	S900～S999	100 点	掉电保存
保持用	S1000～S4095	3096 点	掉电保存
定时器			
100ms	T0～T191	192 点	0.1～3,276.7 秒
100ms	T192～T199	8 点	0.1～3,276.7 秒,子程序，中断子程序用
10ms	T200～T245	46 点	0.01～327.67 秒
1ms 累积型	T246～T249	4 点	0.001～32.767 秒
100ms 累积型	T250～T255	6 点	0.1～3,276.7 秒
1ms	T256～T511	256 点	0.001～32.767 秒
计数器			
一般用增计数(16 位)	C0～C99	100 点	0～32,767，不支持掉电保存
保持用增计数(16 位)	C100～C199	100 点	0～32,767，掉电保存
一般用双方向(32 位)	C200～C219	20 点	-2,147,483,648～+2,147,483,647，不支持掉电保存
保持用双方向(32 位)	C220～C234	15 点	-2,147,483,648～+2,147,483,647，掉电保存
高速计数器			
单相单计数的输入双方向(32 位)	C235～C245	11 点	-2,147,483,648～+2,147,483,647，掉电保存
单相双计数的输入双方向(32 位)	C246～C250	5 点	
双相双计数的输入双方向(32 位)	C251～C255	5 点	
数据寄存器			
一般用(16 位)	D0～D199	200 点	不支持掉电保存
保持用(16 位)	D200～D511	312 点	掉电保存

保持用(16 位)	D512~D7999	7488 点	掉电保存
特殊用(16 位)	D8000~D8511	512 点	掉电保存
特殊用(16 位)	SD0~SD1023	1024 点	掉电保存
变址用(16 位)	V0~V7, Z0~Z7	16 点	掉电保存
文件寄存器			
扩展寄存器(16 位)	R0~R32767	32768 点	掉电保存
标号			
CJ 指令用	P0~P511	512 点	与 LBL 指令配套使用
子程序			
CALL 指令用	/	512 点	可设为普通子程序、加密子程序、带参数子程序、带参数加密子程序
输入中断 X000~X007	I00□, I10□ I20□, I30□ I40□, I50□ I56□, I57□	8 点	□表示: 0 下降沿中断, 1 上升沿中断。 沿中断禁止标志位寄存器置 ON 后, 则对应的输入中断被禁止。
定时中断	I6□□~I8□□	3 点	□□=01~99, 时基=1ms
计数完成中断	I010~I080	8 点	DHSCS 指令用
脉冲完成中断	I502~I506	5 点	
运动控制子程序	MC00~MC63	64 个	
嵌套指针			
主控回路用	N0~N7	8 点	MC 指令用
常数			
十进制常数 K	16 位	-32,768~+32,767	
	32 位	-2,147,483,648~+2,147,483,647	
十六进制常数 H	16 位	0~FFFF	
	32 位	0~FFFFFFFF	
实数 E	32 位	-1.0*2e128~-1.0*2e-126, 1.0*2e-126~1.0*2e128(32 位)	

3.2 输入输出继电器

3.2.1 输入继电器 X

输入继电器 X 代表 PLC 外部输入信号状态的元件，通过 X 端口来检测外部信号状态，0 代表外部信号开路，即 OFF；1 代表外部信号闭合，即 ON。用程序指令方法不能修改输入继电器的状态，其接点信号（常开型、常闭型）在用户程序中都可无限次使用。

输入继电器 X 以 8 进制方式编号，为 X0, X1…X7, X10, X11…X25。当接入本地扩展模块后，扩展模块上 X 端口的编号按紧接主模块上 X 端口的编号，依次向后编号，但扩展模块的编号总是从 8 进制个位为 0 开始的。

3.2.2 输出继电器 Y

输出继电器直接连接在外部控制端口的硬件上，在逻辑上与 PLC 的物理输出端口相对应。PLC 每次扫描完用户程序后，会将 Y 继电器的元件状态传送到 PLC 的硬件端口上，0 表示输出端口开路，即输出端口为 OFF；1 表示输出端口闭合，即输出端口为 ON。Y 继电器元件可在用户程序中无限次使用。硬件上，根据输出元件的不同，可分为继电器型、晶体管型等。

输出继电器 Y 以 8 进制方式编号，为 Y0, Y1…Y7, Y10, Y11…Y15。当接入本地扩展模块后，扩展模块上 Y 端口的编号按紧接主模块上 Y 端口的编号，依次向后编号，但扩展模块的编号总是从 8 进制个位为 0 开始的。

3.3 辅助继电器 M

辅助继电器 M 元件用作用户程序执行过程中的中间变量，如同实际电控系统中的辅助继电器，用于状态信息的传递，与外部端口没有直接的联系，但可通过程序语句将 X 复制到 M，或将 M 复制到 Y 的方式与外界发生联系。可将多个 M 变量组成为字变量使用，一个 M 变量可无限次使用。

一般用	保持用	保持用	特殊用	特殊用
M0~M499 500 点 ^[1]	M500~M1023 524 点 ^[2]	M1024~M7679 6656 点 ^[3]	M8000~M8511 512 点	SM0~SM1023 1024 点

[1]非停电保持区域。使用参数设定，可变更成停电保持区域。

[2]停电保持区域。使用参数设定，可变更成非停电保持区域。

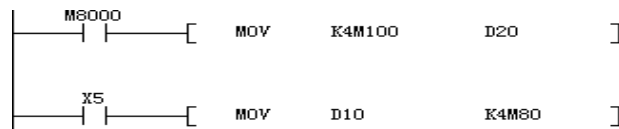
[3]停电保持区域，不能通过参数来改变。

辅助继电器 M 以 10 进制方式编号，M8000 以上的变量为系统专用变量，用于 PLC 用户程序与系统状态的交互；部分 M 变量还具有掉电保存特性。

特殊辅助继电器 SM 为系统专用变量，以 10 进制方式编号，用于 PLC 用户程序与系统状态的交互。

可编程控制器内有大量的特殊辅助继电器，这些特殊辅助继电器各有其特定的功能。具体功能请查看附录。注意，用户不可使用尚未定义的特殊辅助继电器。

可以将连续的 M 变量按字节或字来进行访问，例如：



其中 K4M100 表示将 M100、M101、M102……M115 共 16 个单元，组成一个字的单元进行读操作（M100 作为字的 bit0……M115 作为字的 bit15），这样可提高编程效率。

3.4 状态继电器 S

状态继电器 S 用于步进程序的设计和执处理，利用 STL 步进指令控制步进状态 S 的转移，简化编程设计。

若没有采用 STL 编程方式，S 可当作普通的位元件，就如 M 变量一样来使用。状态 S 变量以 S0, S1.....S999 等符号标识，其序号是以 10 进制方式编号。部分 S 变量具有掉电保存功能。

一般用		保持用	报警用	保持用
S0~S9	S10~S499	S500~S899	S900~S999	S1000~S4095
10 点[1]	490 点[1]	400 点[2]	100 点	3096 点[3]

[1]非停电保持区域。使用参数设定，可变更成停电保持区域。

[2]停电保持区域。使用参数设定，可变更成非停电保持区域。

[3]停电保持区域，无法用参数来改变。

3.5 定时器 T

定时器的功能：定时功能。定时器的组成：定时器的线圈，定时器的接点，计数时值寄存器三部分组成。

定时器的动作原理：定时器的线圈“得电”（能流有效）时，定时器开始计时，若计时值达到预设的时间值时，其接点动作，a 接点（NO 接点）闭合，b 接点（NC 接点）断开。若线圈“失电”（能流无效）时，定时器的接点恢复初始状态，计时值自动清除。也有部分定时器具有累计、掉电保持等特性，重新上电后仍维持掉电前的数值。

定时器的编号：以 10 进制方式编号，以 T0, T1...T10, T11...等符号标识。

定时器的步长：有 1ms、10ms、100ms 等，部分具有掉电保持特性。

100ms		10ms	1ms 累计型	100ms 累计型	1ms
T0~T191	T192~T199	T200~T245	T246~T249	T250~T255	T256~T511
192 点[1]	8 点[2]	46 点[1]	4 点[3]	6 点[4]	256 点[1]

[1]非停电保持区域。

[2]非停电保持区域，子程序、中断子程序用。

[3]停电保持区域，累计型。

[4]停电保持区域，累计型。

定时器的设定值：程序存储器内的常数（K）作为设定值，也可用数据寄存器（D）的内容进行间接指定。

当采用数据寄存器 D 作设定值时，D 的内容必须在开始计时前设定好，当开始计数后，D 的数据变化只有在下一次启动计时的时候才能生效。

没有用作定时器使用的定时器编号，也可用作数值存储用的数据寄存器。

累计型的定时器在计时的时间达到设定时间时，其触点只有在执行线圈指令或 END 指令时，输出触点才能动作。

从驱动定时器的线圈开始到定时器的触点动作，可能的定时长度说明如下：

（1）最长的情况为 $(T+T_0+a)$ ，其中：T 为设定的定时时间； T_0 为程序扫描执行时间；a 为定时器的计时步长。

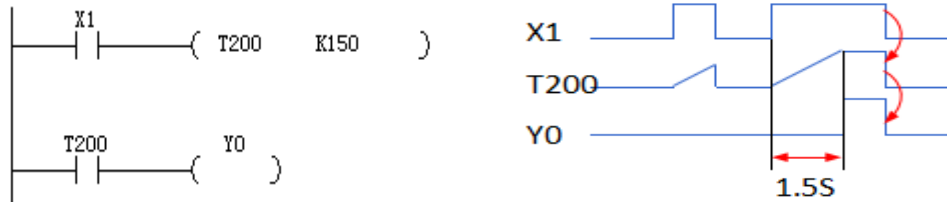
（2）最短的情况为 $(T-a)$ 。

（3）若计时器的触点指令位于线圈指令之前，最不理想的定时长度为 $(T+2T_0)$ 。

（4）利用定时器的 b 触点，可以实现延时断开、自激振荡的输出信号等。

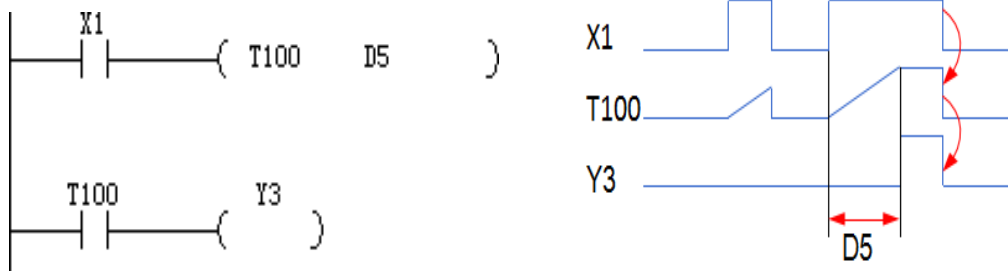
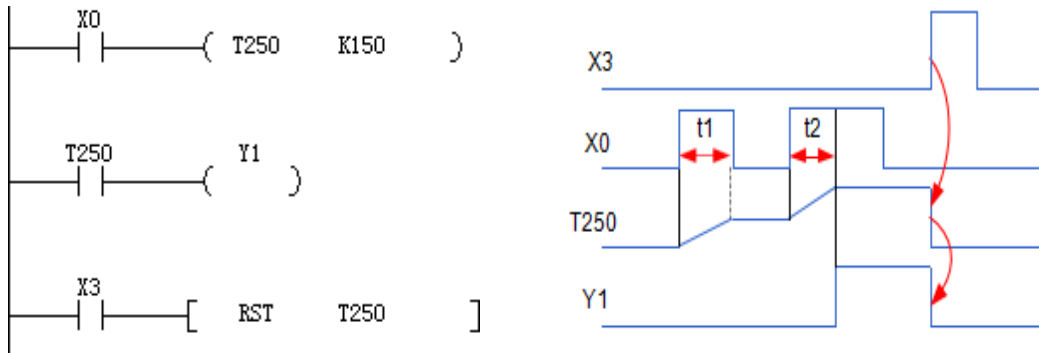
（5）PLC 还提供了特殊定时器指令，如 TTMR、STMR 等，请参见相应指令的说明。

【使用举例 1】普通计时器 T200 为 10ms 步长的计数器, 实际动作延迟为 $150 \times 10\text{ms} = 1500\text{ms}$, 即 1.50s, 动作原理为:



【使用举例 2】对于有掉电保持的累计型计时器 T250, 驱动信号为 OFF, 或 PLC 掉电时, 其内部计数值维持不变, 下次驱动信号为 ON 时, 继续计时, 直到满足计时到设定值时, 输出触点闭合。当复位计时器线圈时, 计时值清除, 输出触点断开, 如下图。因计数器 T250 为 100ms 步长的, 实际动作延迟累计为 $150 \times 100\text{ms} = 15000\text{ms}$, 即 15.0s, 即图中的 $(t_1 + t_2)$ 时间:

【使用举例 3】定时器的设定动作值可通过寄存器 D 来进行设定, 如下图。(计数器计时过程中, 若寄存器 D 内数值变化时, 在下一一次计时器启动时生效。)



3.6 计数器 C

计数器的功能：用于完成计数功能。

计数器的组成：计数器的线圈，计数器的接点，计时数据值寄存器。

计数器的工作过程：当计数器的线圈每来一个上升沿时（由 OFF→ON），计数器的计数值变化 1，当计数值达到设定值时，计数器的接点动作，常开点（NO）闭合，常闭点（NC）断开。若清除计数值，输 a 接点即断开，b 接点（NC 接点）闭合。部分计数器具有掉电保持、累计等特性，重新上电后仍维持掉电前的数值。

计数器的编号：以 10 进制方式编号，以 C0，C1...C11 等进行标识。

一般用 16bit	保持用 16bit	一般用 32bit	保持用 32bit	保持用 32bit
C0~C99 100 点增计数 ^[1]	C100~C199 100 点增计数 ^[2]	C200~C219 20 点双向计数 ^[1]	C220~C234 15 点双向计数 ^[2]	C235~C255 21 点高速计数 ^[2]

[1]非停电保持区域。使用参数设定，可变更成停电保持区域。

[2]停电保持区域。使用参数设定，可变更成非停电保持区域。

对于 32bit 计数器 C200~C234，由特殊辅助继电器 M8200~M8234 作为增计数/减计数器切换控制，见下表：

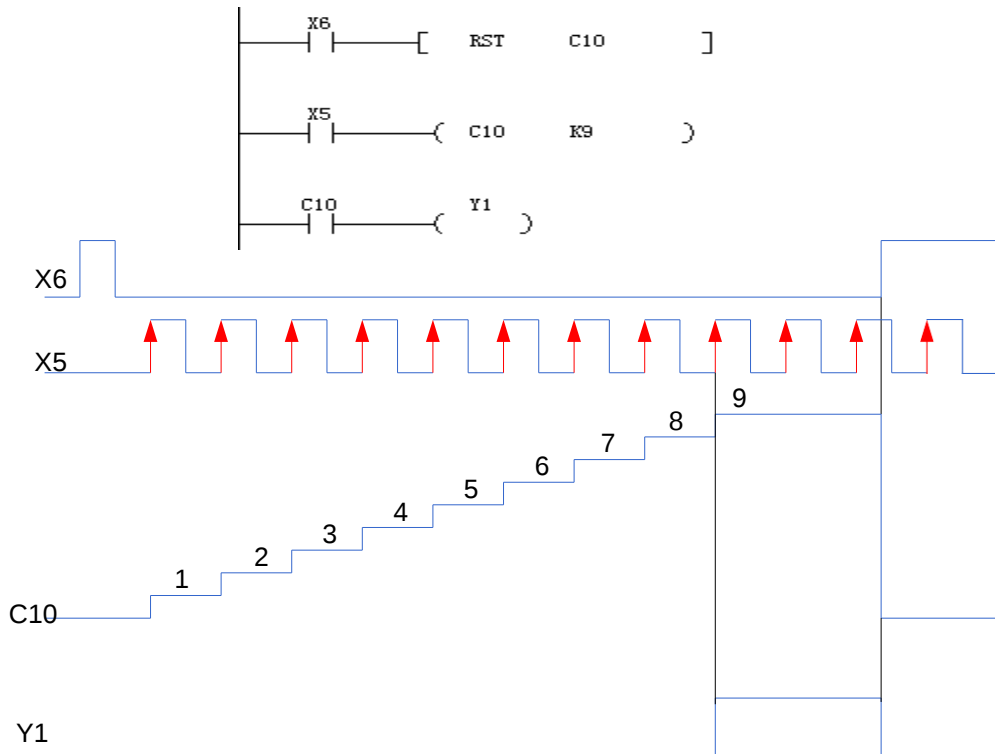
计数器 NO.	方向切换	计数器 NO.	方向切换	计数器 NO.	方向切换	计数器 NO.	方向切换
C200	M8200	C209	M8209	C218	M8218	C226	M8226
C201	M8201	C210	M8210	C219	M8219	C227	M8227
C202	M8202	C211	M8211	—	—	C228	M8228
C203	M8203	C212	M8212	C220	M8220	C229	M8229
C204	M8204	C213	M8213	C221	M8221	C230	M8230
C205	M8205	C214	M8214	C222	M8222	C231	M8231
C206	M8206	C215	M8215	C223	M8223	C232	M8232
C207	M8207	C216	M8216	C224	M8224	C233	M8233
C208	M8208	C217	M8217	C225	M8225	C234	M8234

16bit 计数器与 32bit 计数器的特点如下表所示。可按计数方向的切换与计数范围的使用条件来分开使用。

项目	16 位计数器	32 位计数器
计数方向	增计数	增减切换使用（见上表）
设定值范围	1~32,767	-2,147,483,648~+2,147,483,647
指定的设定值类型	常数 K 或数据寄存器	常数 K，也可用 2 个 D 数据寄存器
当前值的变化	顺数后不变化	顺数后变化（循环计数器）
输出接点	顺数后保持动作	顺数保持动作，倒数复位
复位动作	执行 RST 命令时，计数器的当前值为零，输出接点复位	
当前值寄存器	16 位	32 位

3.6.1 16bit 计数器

对于 16bit 计数器，其有效设定值为 K1~K32,767(10 进制常数)；设定值 K0 和 K1 具有相同效果，即在第一次计数开始时输出触点就动作。如下例：



计数输入 X5 每驱动 C10 线圈一次，计数器的当前值就增加，在执行第 9 次的线圈指令时，输出触点动作。以后即使计数输入 X5 再动作，计数器的当前值不变。如果复位输入 X6 为 ON，则执行 RST 指令，计数器的当前值清为 0，输出触点复位。

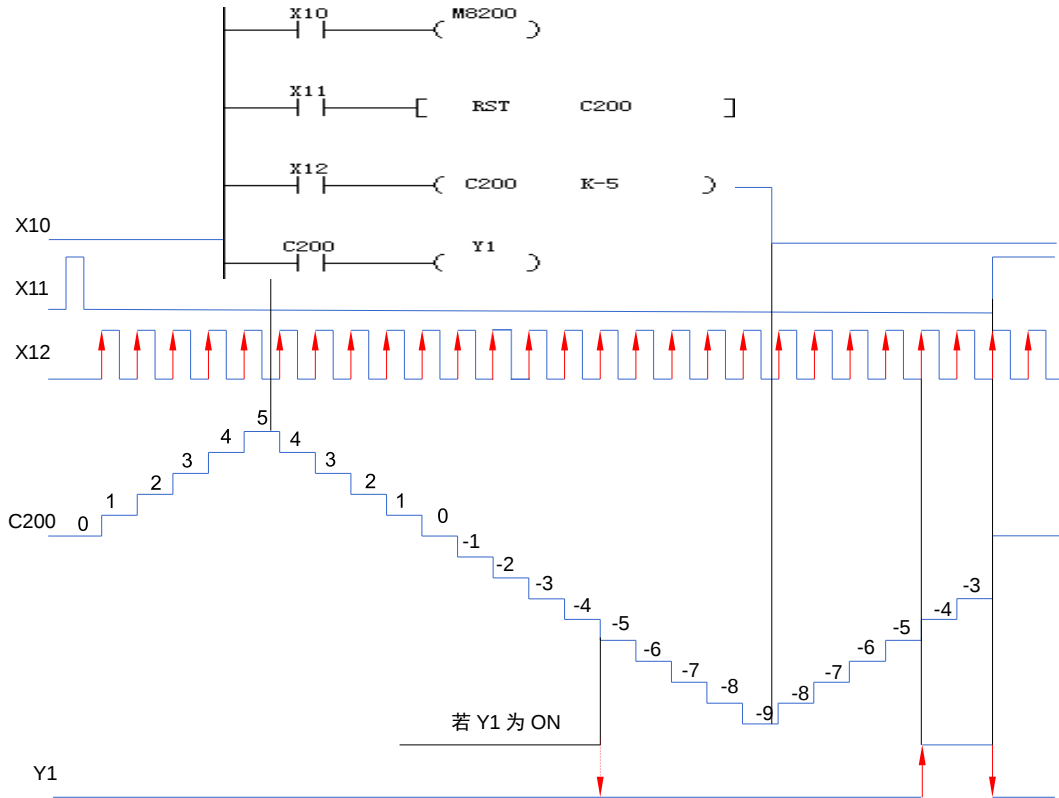
计数器的设定值，除用上述常数 K 设定外，还可由数据寄存器编号指定。如上例中，指定 D20，如果 D20 的内容为 9，则与设定 K9 是一样的。

在以 MOV 等指令将设定值以上的数据写入当前值寄存器时，则在下次输入时，输出线圈接通，当前值寄存器变为设定值。

对于一般用计数器，如果切断可编程控制器的电源，则计数器的计数值被清除，而停电保持用的计数器则可存储停电前的计数值，因此计数器可接着上次数值累计计数。

3.6.2 32bit 计数器

对于 32bit 计数器，增计数/减计数的设定值有效范围-2,147,483,648~+2,147,483,647(10 进制常数)，可用常数 K 或数据寄存器 D 的内容进行设定。利用特殊的辅助继电器 M8200~M8234 指定增计数/减计数的方向，如果对 Cxxx 驱动 M8xxx 置 1，则为减计数，不驱动时清零，则为增计数。



当前值的增减与输出触点的动作无关，但是如果从 2,147,483,647 开始增计数，再输入一个脉冲后，则成为 -2,147,483,648。同样，如果从 -2,147,483,648 开始减计数，再输入一个脉冲，则成为 2,147,483,647。（这类动作被称为环形计数）；如果复位输入 X11 为 ON，则执行 RST 指令，计数器的当前值变为 0，输出触点也复位。

使用供停电保持用的计数器时，计数器的当前值、输出触点动作与复位状态停电保持。32bit 计数器也可作为 32bit 数据寄存器使用。但是，32bit 计数器不能作为 16 位应用指令中的软元件。在以 DMOV 指令等把设定值以上的数据写入当前值数据寄存器时，则在以后的计数输入时可继续计数，触点也不变化。

对于 16bit 计数器，最高位 (bit15) 为符号位，处理的数据为 0~32767 范围，即只能为正数；对于 32bit 计数器，最高位 (bit31，即高字节的最高位) 为符号位，处理的数据范围为 -2,147,483,648~2,147,483,647。

3.6.3 高速计数器

高速计数器可以实现对外部输入信号的计数，可以实现单相单计数、单相双计数、AB 相 1/4 倍频计数等。

3.7 寄存器

寄存器的作用：用于数据的运算和存储。如对定时器、计数器、模拟量参数的运算和存储等。

寄存器的宽度：16bit。若采用 32bit 指令，则自动将相邻的 2 个寄存器组成为 32bit 寄存器使用，地址较低的为低字节，而地址较高的为高字节。

寄存器类型：数据寄存器 D、变址用数据寄存器 V 和 Z、文件寄存器 R。

一般用	保持用	保持用	特殊用	变址用	保持用	特殊用
D0~D199 200 点 ^[1]	D200~D511 312 点 ^[2]	D512~D7999 7488 点 ^[3]	D8000~D8511 512 点	V0~V7 Z0~Z7	R0~R32767 32768 点 ^[3]	SD0~SD1023 1024 点

[1]非停电保持区域。使用参数设定，可变更成停电保持区域。

[2]停电保持区域。使用参数设定，可变更成非停电保持区域。

[3]停电保持区域，不能通过参数来改变。

3.7.1 数据寄存器 D

数据寄存器的作用：处理各种数值数据，通过利用它，可以进行各种控制。如作为定时器与计数器的设定值被指定，用于数据的各种运算等。一些特殊数据寄存器，用于系统工作状态参数缓存，查询这些寄存器，可用于判断运行参数。关于特殊数据寄存器的停电保持特点请查看附录。

数据寄存器 D 是 16 位，当使用 32 位数据时，以 2 个相邻的数据寄存器来表示 32 位的数据，低 16 位数据存放在低地址（Dn）中，高 16 位数据存放在高地址（Dn+1）中。

在指定 32 位数据时，如果指定了低位（Dn），则高位为继其之后的编号（Dn+1）被自动占用。低位可用奇数或偶数的任意一种软元件编号指定，考虑到外围设备的监视功能，建议低位采用偶数软元件编号。

非停电保持区域的数据寄存器：一旦在寄存器中写入数据，只要不再写入其他数据，就不会变化。但是当停电或由 RUN→STOP 时，寄存器的数据将被清除为 0。（如果驱动特殊的辅助继电器 M8033，则可以保持）。

停电保持区域的数据寄存器：在停电或由 RUN→STOP 时，寄存器的数据将被保持。

将掉电保持专用的数据寄存器做一般用途时，请在程序的起始步采用 RST 或 ZRST 指令，将寄存器内的数据清除。

特殊用途的数据寄存器是指写入特定目的的数据，用于实现控制器的一些特殊功能，可理解为用户程序与 PLC 系统程序进行数据交互的特殊单元。

3.7.2 变址寄存器 V、Z

变址寄存器的功能：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。在应用指令的操作数中，还可以同其他的软元件编号或数值组合使用。但需注意 LD，AND，OUT 等基本顺控指令或步进梯形图指令的软元件编号不能同变址寄存器组合使用。

V0~V7，Z0~Z7 共有 16 个。在组合成 32 位时，V 在高位，Z 在低位。

V、Z 寄存器可采用 16bit 和 32bit 方式进行访问，如下图说明：

16 位访问方式时，为独立的 16 个寄存器

16 位 16 位



V0~V7:8 点 Z0~Z7:8 点

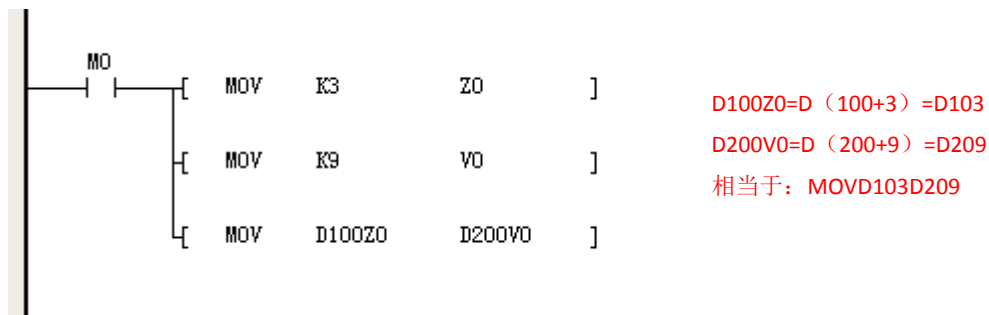
32 位访问方式时，按如下方式组合成 8 个寄存器

32 位

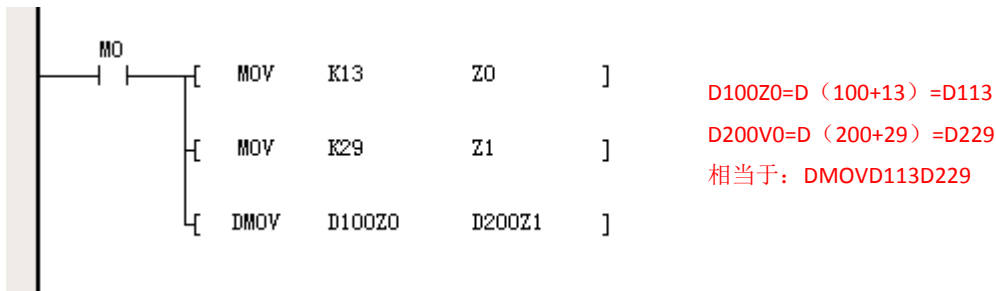
32 位	
V0 (高位)	Z0 (低位)
V1 (高位)	Z1 (低位)
V2 (高位)	Z2 (低位)
V3 (高位)	Z3 (低位)
V4 (高位)	Z4 (低位)
V5 (高位)	Z5 (低位)
V6 (高位)	Z6 (低位)
V7 (高位)	Z7 (低位)

在处理 32 位应用指令中的软元件或处理超过 16 位范围的数值时，V（高位）、Z（低位）被同时访问，指定的寄存器名必须为 Z0~Z7。即使指定了 V0~V7 的高位侧，也无法进行变址。

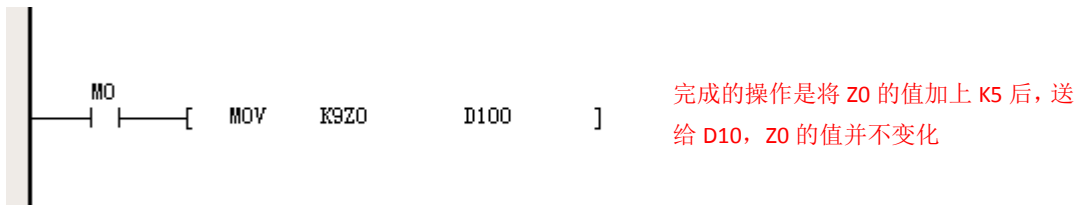
(1) 16 位变址应用举例：



(2) 32 位变址应用举例：



(3) 常数变址的特例:



当 V、Z 间接寻址方式用于循环指令中 (V、Z 随循环变量变化), 进行成片数据区的操作, 或用于查表操作等, 简化编程, 提高指令效率。

3.7.3 文件寄存器 R

文件寄存器 R 的使用和数据寄存器 D 使用方法相同。

3.8 标号与子程序

标号/跳转指针 (P) 用于跳转程序的入口地址的标识, 子程序 SBR 用于子程序起始地址的标识, 运动控制子程序以 MC 作为标识; 中断子程序 (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。

标号	子程序	概述	
P	CJ 指令用	与 LBL 指令配套使用。标号是在每个程序块内使用的, 不可以跳转到当前程序块以外的地方。所有程序块总计允许 512 个跳转标号存在	
L	CJ 指令用	与 P 等价	
SBR	CALL 指令用	最多支持 512 个子程序; 其中子程序属性可以被设置为普通子程序、加密子程序、带参数的子程序、带参数的加密子程序; 加密子程序、带参子程序与普通子程序容量一样不受限制, 共同占用系统 64K 步的容量。	
I	中断子程序	外部中断	X000-X002 输入中断, 编号 I00□, I10□, I20□, 3 点, (□表示: 0 下降沿中断, 1 上升沿中断)。沿中断禁止标志位寄存器置 ON 后, 则对应的输入中断被禁止
		定时中断	I6□□, I7□□, I8□□, 3 点 (□□=1~99, 时基=1ms)
		计数完成中断	I010、I020、I030、I040、I050、I060、I070、I080, 8 点 (DHSCS 指令用)
		脉冲完成中断	I502~I504, 3 点
MC	运动控制子程序	最多支持 64 个; 编号 MC0 - MC63; 另外支持 1 个 G-code 代码子程序, 编号 MC10000, G-code 子程序文件中支持多个 Oxxxx, 编号 O0000-O9999。 运动子程序容量和其他子程序一样, 不受限制, 共同占用系统 64K 步的容量。	

关于中断和子程序指针的具体使用方法, 参考“[第 7 章中断](#)”和“[第 9 章子程序](#)”。

3.9 常数

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），状态继电器 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，OCT	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。

[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-2,147,483,648～2,147,483,647。

[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。

16bit 指令中，常数 H 的取值范围为 0000～FFFF。

32bit 指令中，常数 K 的取值范围为 0x0～0xFFFFFFFF。

第4章 指令

运动控制系统的指令很多，可分为以下几类：程序逻辑指令；程序流程指令，数据比较；数据运算；数据处理；矩阵指令；字符串指令；时钟指令；高速输入・脉冲定位・通讯定位；运动控制；通信；外设；电子凸轮指令。

4.1 程序逻辑指令

触点指令	
LD	加载常开接点
LDI	加载常闭接点
AND	串联常开接点
ANI	串联常闭接点
OR	并联常开接点
ORI	并联常闭接点
LDP	取脉冲上升沿
LDF	取脉冲下降沿
ANDP	与脉冲上升沿检测串行连接
ANDF	与脉冲下降沿检测串行连接
ORP	或脉冲上升沿检测并行连接
ORF	或脉冲下降沿检测并行连接
INV	运算结果取反
BLD	位数据位触点
BLDI	位数据位反触点
BAND	位数据位与触点
BANI	位数据位与非触点
BOR	位数据位或触点
BORI	位数据位或非触点
结合指令	
ANB	串联回路方块
ORB	并联回路方块
MPS	存入堆栈
MRD	读出堆栈（能流指针不变）
MPP	读出堆栈
MEP	能流沿控制，运算结果脉冲化
MEF	
输出指令	
OUT	驱动线圈
SET	置位动作保存线圈指令
RST	接点或缓存器清除
PLS	脉冲上升沿检测线圈指令

PLF	脉冲下降沿检测线圈指令
BOU	位数据输出
BSET	位数据置位
BRST	位数据复位
ALT	交替输出
其他处理指令	
NOP	无动作

4.1.1 触点指令

触点指令	
LD	加载常开接点
LDI	加载常闭接点
LDP	取脉冲上升沿
LDF	取脉冲下降沿
AND	串联常开接点
ANI	串联常闭接点
ANDP	与脉冲上升沿检测串行连接
ANDF	与脉冲下降沿检测串行连接
OR	并联常开接点
ORI	并联常闭接点
ORP	或脉冲上升沿检测并行连接
ORF	或脉冲下降沿检测并行连接
INV	运算结果取反
BLD【注】	位数据位触点
BLDI【注】	位数据位反触点
BAND【注】	位数据位与触点
BANI【注】	位数据位与非触点
BOR【注】	位数据位或触点
BORI【注】	位数据位或非触点

【注】该指令用于对一个字或者双字元件选择性的取出某一位参与运算。字指令占用 5 步，双字指令占用 9 步。其操作数也是一样，第一个操作数为需要参与运算的字或者双字元件，第二个参数为取出那一位参与运算。当字指令时，第二个操作数只能取 0-15，当双字指令时，第二个操作数只能取 0-31。

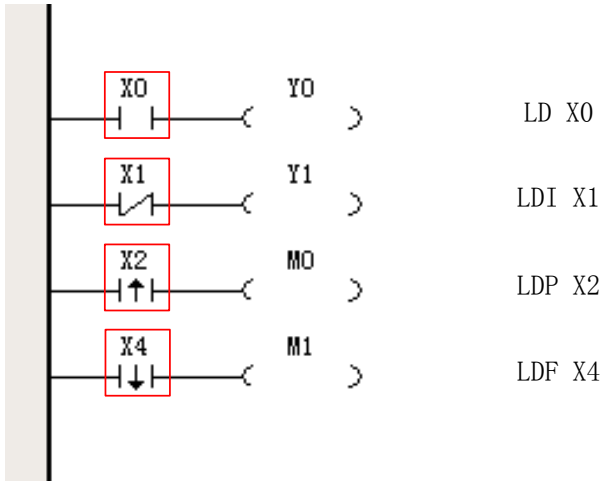
LD/LDI/LDP/LDF

LD	加载常开接点	操作数类型：S、X、Y、M、T、C 指令步长：1step
LDI	加载常闭接点	
LDP	取脉冲上升沿	
LDF	取脉冲下降沿	

指令	操作数						
LD	X0~X377	Y0~Y377	M0~M7679 M8000~M8511	S0~S4095	SM0-SM1023	T0~T511	C0~C255
LDI							
LDP							
LDF	✓	✓	✓	✓	✓	✓	✓

LD/LDI/LDP/LDF 指令用于左母线开始的接点，其中：

- LD/LDI 指令分别是把 A 接点和 B 接点的当前能流状态保存，同时把取来的接点状态存入累计缓存器内。
- LDP 指令用于取用接点信号的上升沿，若本次扫描中检测到对应信号的上升跳变，则触点有效，下一次扫描时，触点即变成无效。
- LDF 指令用于取用接点信号的下降沿，若本次扫描中检测到对应信号的下降跳变，则触点有效，下一次扫描时，触点即变成无效。

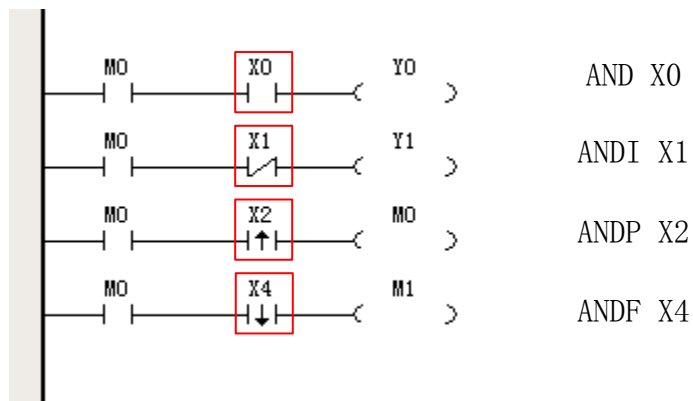


AND/ANI/ANDP/ANDF

AND	串联常开接点	操作数类型：S、X、Y、M、T、C 指令步 长：1step
ANI	串联常闭接点	
ANDP	与脉冲上升沿检测串行连接	指令步长：3step
ANDF	与脉冲下降沿检测串行连接	

指令	操作数						
AND ANI	X0~X377	Y0~Y377	M0~M7679 M8000~M8511	S0~S4095	SM0-SM1023	T0~T511	C0~C255
ANDP ANDF	✓	✓	✓	✓	✓	✓	✓

- AND/ANI/ANDP/ANDF 指令用于串联接点的状态运算，其操作是先读取目前所指定串联接点的状态再与接点之前逻辑运算结果作“与”（AND）的运算，并将结果存入累计缓存器内。
- AND/ANI 指令分别是将 A 接点和/B 接点的状态参与 AND 运算；
- ANDP 指令是将接点的上升沿跳变状态参与 AND 运算；
- ANDF 指令是将接点的下降沿跳变状态参与 AND 运算；

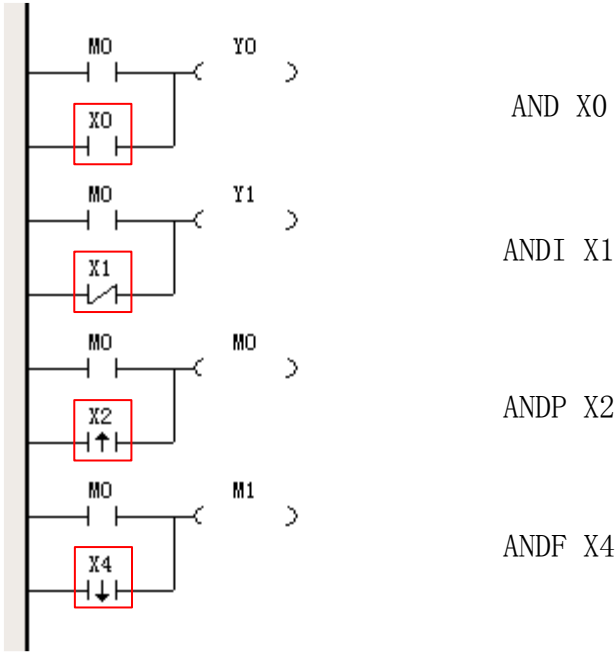


OR/ORI/ORP/ORF

OR	并联常开接点	操作数类型：S、X、Y、M、T、C 指令步长：1step
ORI	并联常闭接点	
ORP	或脉冲上升沿检测并行连接	指令步长：3step
ORF	或脉冲下降沿检测并行连接	

指令	操作数						
OR	X0~X377	Y0~Y377	M0~M7679 M8000~M8511	S0~S4095	SM0-SM1023	T0~T511	C0~C255
ORI							
ORP	✓	✓	✓	✓	✓	✓	✓
ORF							

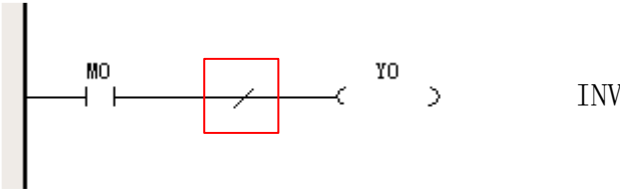
- OR/ORI 指令用于并联接点的状态运算，其操作是先读取目前所指定接点的状态，再与接点之前逻辑运算结果作“或”（OR）的运算，并将结果存入累计缓存器内。
- OR/ORI 指令分别是将 A 接点和/B 接点的状态参与 OR 运算；
- ORP 指令是将接点的上升沿跳变状态参与 OR 运算；
- ORF 指令是将接点的下降沿跳变状态参与 OR 运算。



INV 运算结果取反

INV	运算结果取反	指令步长：1step
指令	操作数	
INV	无	

将 INV 指令之前的逻辑运算结果反相后存入累计缓存器内。当 INV 指令之前能流为 ON，经过 INV 后能流变为 OFF；反之，变为 ON。



BLD 位数据位触点

1.指令形式

根据源数据指定位的状态 ON(OFF)确定该节点状态 ON(OFF)，与左母线直接相连的接点。

BLD S n			位数据位触点	指令执行	
S	源数据	源数据软元件编号		16 位指令（5step）	32 位指令（9step）
n	加载位	加载指定位，范围为 0-15(16 位指令)或 0-31(32 位指令)		BLD 连续执行	DBLD 连续执行

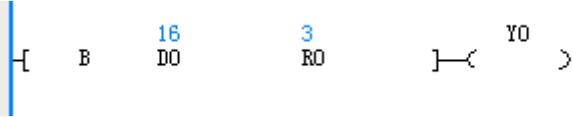
2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

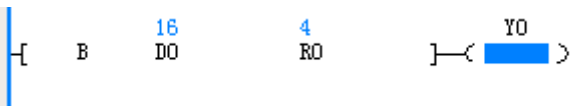
注：带灰色底纹软元件，表示可支持。

【使用举例】

n=3 情况如下：



n=4 情况如下：



BLDI 位数据位反触点

1.指令形式

根据源数据指定位的状态 ON(OFF)确定该节点状态 OFF(ON)，与左母线直接相连的节点。

BLDI S n			位数据位反触点	指令执行	
S	源数据	源数据软元件编号		16 位指令 (5step)	32 位指令 (9step)
n	加载位	加载指定位，范围为 0-15(16 位指令)或 0-31(32 位指令)		BLDI 连续执行	DBLDI 连续执行

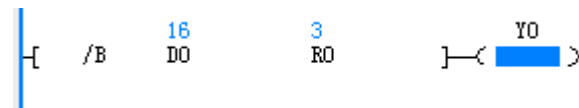
2.操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

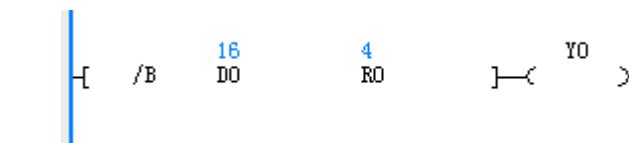
注：带灰色底纹软元件，表示可支持。

【使用举例】

n=3 情况如下：



n=4 情况如下



BAND 位数据位与触点

1.指令形式

根据源数据指定位的状态 ON(OFF)确定该节点状态 ON(OFF)，与其它节点串联连接的节点。

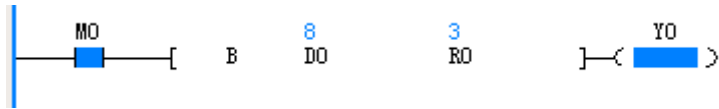
BAND S n			位数据位与触点	指令执行	
S	源数据	源数据软元件编号		16 位指令 (5step)	32 位指令 (9step)
n	加载位	加载指定位，范围为 0-15(16 位指令)或 0-31(32 位指令)		BAND 连续执行	DBAND 连续执行

2.操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

【使用举例】



BANI 位数据位与非触点

1.指令形式

根据源数据指定位的状态 ON(OFF)确定该节点状态 OFF(ON)，与其它节点串联连接的节点。

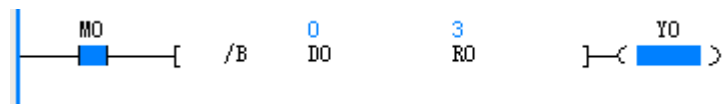
BANI S n			位数据位与非触点	指令执行	
S	源数据	源数据软元件编号		16 位指令（5step） BANI 连续执行	32 位指令 （9step） DBANI 连续执行
n	加载位	加载指定位，范围为 0-15(16 位指令)或 0-31(32 位指令)			

2.操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

【使用举例】



BOR 位数据位或触点

1. 指令形式

根据源数据指定位的状态 ON(OFF)确定该节点状态 ON(OFF)，与其它节点并联连接的节点。

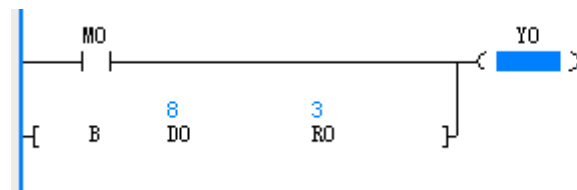
BOR S n			位数据位或触点	指令执行	
S	源数据	源数据软元件编号		16 位指令（5step） BOR 连续执行	32 位指令（9step） DBOR 连续执行
n	加载位	加载指定位，范围为 0-15(16 位指令)或 0-31(32 位指令)			

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

【使用举例】



BORI 位数据位或非触点

1. 指令形式

根据源数据指定位的状态 ON(OFF)确定该节点状态 OFF(ON)，与其它节点串联连接的节点。

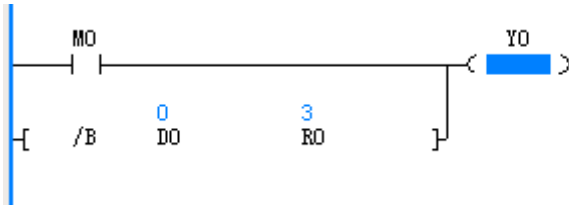
BORI S n			位数据位或非触点	指令执行	
S	源数据	源数据软元件编号		16 位指令 (5step) BORI 连续执行	32 位指令 (9step) DBORI 连续执行
n	加载位	加载指定位, 范围为 0-15(16 位指令)或 0-31(32 位指令)			

2. 操作数

操作数	位软元件								字软元件													
	系统·用户								系统·用户				位数指定					变址		常数		实数
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

【使用举例】



4.1.2 结合指令

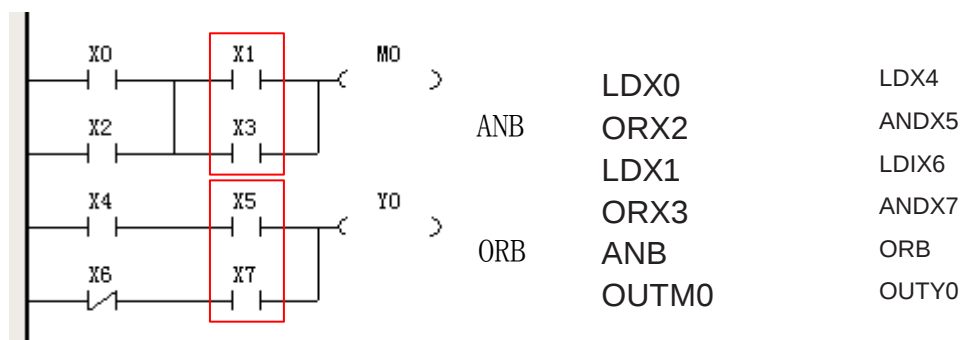
结合指令	
ANB	串联回路方块
ORB	并联回路方块
MPS	存入堆栈
MRD	读出堆栈（能流指针不变）
MPP	读出堆栈
MEP	能流沿控制，运算结果脉冲化
MEF	

ANB 串联回路方块、ORB 并联回路方块

ANB	串联回路方块	指令步长：1step
ORB	并联回路方块	指令步长：1step

指令	操作数
ANB	无
ORB	参与块运算的是最近两次 LD（或 LDI/LDP/LDF）区间的计算能流

ANB 和 ORB 是将前一保存的逻辑结果与目前累计缓存器的内容作“与”和“或”的运算。



堆栈指令 MPS. MRD. MPP

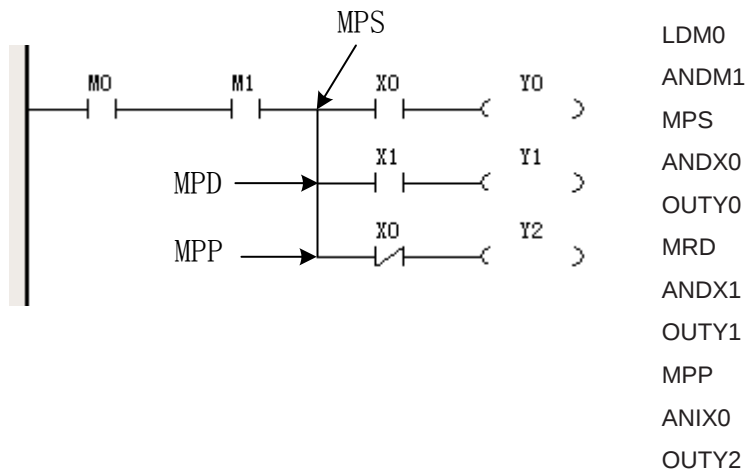
MPS	存入堆栈	指令步长：1step
MRD	读出堆栈（能流指针不变）	指令步长：1step
MPP	读出堆栈	指令步长：1step

指令	操作数
MPS MRD MPP	无

MPS：将目前累计缓存器的内容存入堆栈。（堆栈指针加一）。

MRD：读取堆栈内容存入累计缓存器。（堆栈指针不动）。

MPP：自堆栈取回前一保存的逻辑运算结果，存入累计缓存器。（堆栈指针减一）。



MEP/MEF 运算结果脉冲化

1. 指令形式

指令是使运算结果脉冲化的指令，不需要指定软元件编号

(1) MEP:在到 MEP 指令为止的运算结果，从 OFF→ON 时变为导通状态。

(2) MEF:在到 MEF 指令为止的运算结果，从 ON→OFF 时变为导通状态。

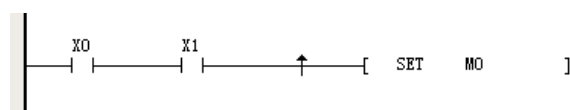
(3) 步长: MEP/MEF 都是 1step.

2. 操作数

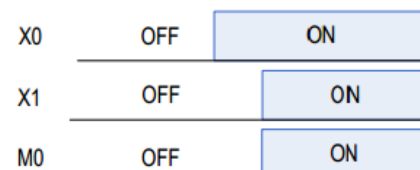
指令	位软元件	字软元件				
	系统·用户	系统·用户	位数指定	变址	常数	实数
MEP	无对象软元件					
MEF	无对象软元件					

【使用举例】

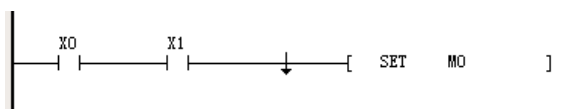
(1) MEP 指令（运算结果上升沿为 ON）



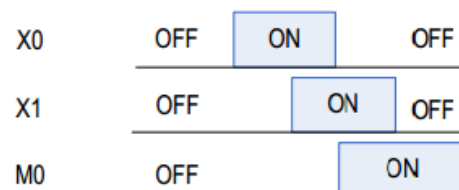
时序图



(2) MEF 指令（运算结果下降沿为 ON）



时序图



4.1.3 输出指令

输出指令	
OUT	驱动线圈
SET	置位动作保存线圈指令
RST	接点或缓存器清除
PLS	脉冲上升沿检测线圈指令
PLF	脉冲下降沿检测线圈指令
BOUT	位数据输出
BSET	位数据置位
BRST	位数据复位
ALT	交替输出

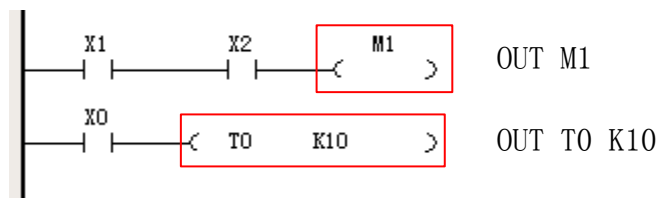
其操作数也是一样，第一个操作数为需要参与运算的字或者双字元件，第二个参数为取出那一位参与运算。当字指令时，第二个操作数只能取 0-15，当双字指令时，第二个操作数只能取 0-31。

OUT/SET/RST/PLS/PLF

OUT	驱动线圈	操作数类型：S、Y、M 指令步长：1step
SET	置位动作保存线圈指令	
RST	接点或缓存器清除	操作数类型：S、Y、M、T、C、D 指令步长：3step
PLS	脉冲上升沿检测线圈指令	
PLF	脉冲下降沿检测线圈指令	

指令	操作数						
OUT	X0~X377	Y0~Y377	M0~M7679 M8000~M8511	S0~S4095	SM0-SM1023	T0~T511	C0~C255
		✓	✓	✓	✓	✓	✓

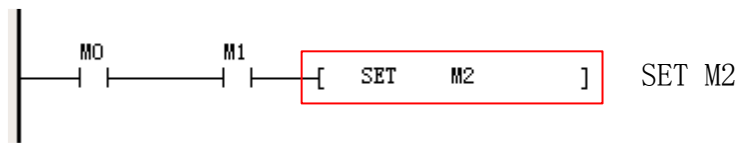
将 OUT 指令之前的逻辑运算结果输出至指定的元件。



指令	操作数						
SET	X0~X377	Y0~Y377	M0~M7679 M8000~M8511	S0~S4095	SM0-SM1023	T0~T511	C0~C255
		✓	✓	✓	✓		

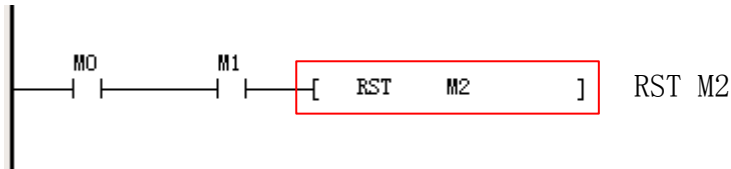
当 SET 指令被驱动，其指定的组件被设定为 ON，且被设定的组件会维持 ON，不管 SET 指令是否仍被

驱动。可利用 RST 指令将该组件设为 OFF。



指令	操作数						
RST	X0~X377	Y0~Y377	M0~M7679 M8000~M8511	S0~S4095	SM0-SM1023	T0~T511	C0~C255
		✓	✓	✓	✓	✓	✓
			D0~D8511	R0~R32767	SD0~SD1023		
			✓	✓	✓		

- 当 RST 指令被驱动，其指定的组件被设定为 OFF，且被设定的组件会维持 OFF，不管 RST 指令是否仍被驱动。可利用 SET 指令将该组件设为 ON。
- RST 指令也可用于 D、V、Z 变量复位，将指定 D、V、Z 元件的值清为 0。

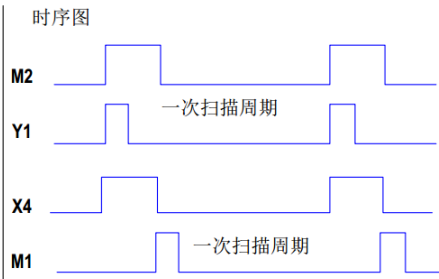
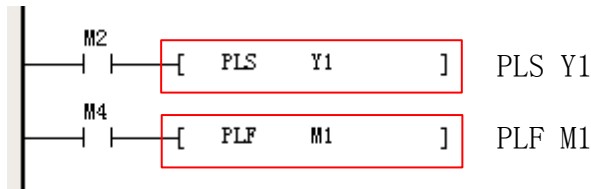


元件	操作结果
S, M, Y	线圈及接点被设定为 OFF
T, C	目前计时或计数值会被设为 0，且线圈及接点被设定为 OFF。
D, V, Z	元件的值清为 0

指令	操作数						
PLS PLF	X0~X377	Y0~Y377	M0~M7679 M8000~M8511	S0~S4095	SM0-SM1023	T0~T511	C0~C255
		✓	✓		✓		

- 当 PLS 指令被上升沿驱动时，其指定的元件被设定为 ON 状态，该 ON 状态仅持续 1 个扫描周期。
- 当 PLF 指令被下降沿驱动时，其指定的元件被设定为 ON 状态，该 ON 状态仅持续 1 个扫描周期。

【使用举例】



BOUT 位数据输出

1. 指令形式

将 BOUT 指令之前的逻辑运算结果输出至指定的位。

BOUT D n				位数据输出	指令执行	
D	执行装置	输出数据软元件编号			16 位指令 (5step)	32 位指令 (9step)
n	输出位	输出指定位, 范围为 0-15 (16 位指令) 或 0-31 (32 位指令)			BOUT 连续执行	DBOUT 连续执行

2. 操作数

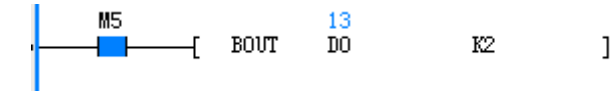
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

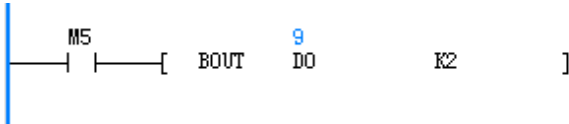
【使用举例 1】

D0 初始值=2#1001 (十进制 K9)

M5=ON 情况如下，D0 的 bit2 被置位，结果 D100=2#1110 (十进制 K13)



接着 M5=OFF 情况如下，D0 的 bit2 被复位，结果 D0=2#1101 (十进制 K13)



BSET 位数据输出

1. 指令形式

当 BSET 指令被驱动，其指定的位被设定为 ON，且被设定的位会维持 ON。不管 BSET 指令是否仍被驱动，可利用 BRST 指令将该位设为 OFF。

BSET D n			位数据置位	指令执行	
D	执行装置	输出数据软元件编号		16 位指令 (5step)	32 位指令 (9step)
n	输出位	输出指定位，范围为 0-15 (16 位指令) 或 0-31 (32 位指令)		BSET 连续执行	DBSET 连续执行

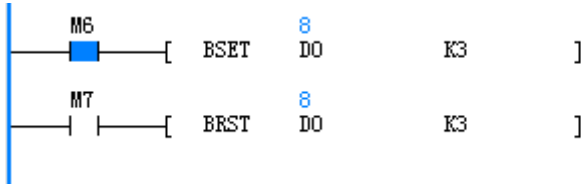
2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

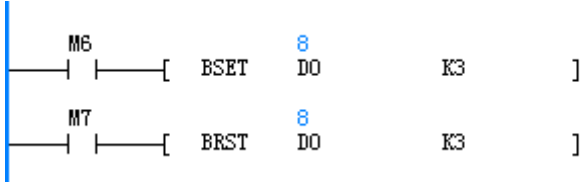
注：带灰色底纹软元件，表示可支持。

【使用举例 1】

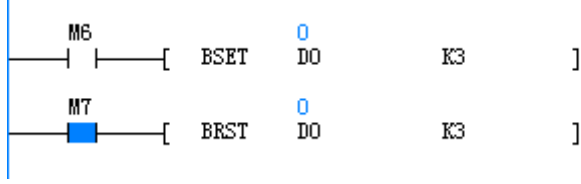
M6=ON



M6=OFF



M7=ON



BRST 位数据复位

1. 指令形式
- 当 BRST 指令被驱动，其指定的位被设定为 OFF。

BRST D n			位数据复位	指令执行	
D	执行装置	输出数据软元件编号		16 位指令（5step）	32 位指令（9step）
n	输出位	输出指定位，范围为 0-15（16 位指令）或 0-31（32 位指令）		BRST 连续执行	DBRST 连续执行

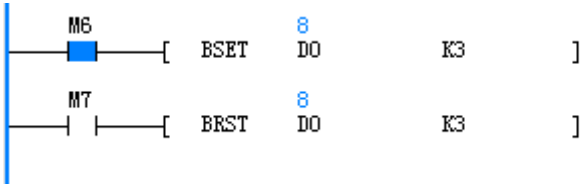
2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

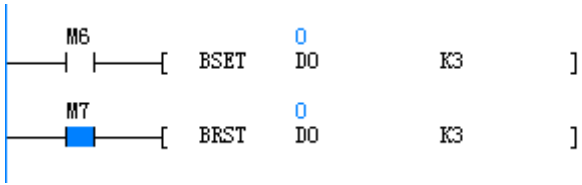
注：带灰色底纹软元件，表示可支持。

【使用举例 1】

M6=ON 时如下：



M7=ON 时如下：



ALT 交替输出

1. 指令形式
- 当驱动条件成立时，对位元件 D 执行 ON/OFF 反转。

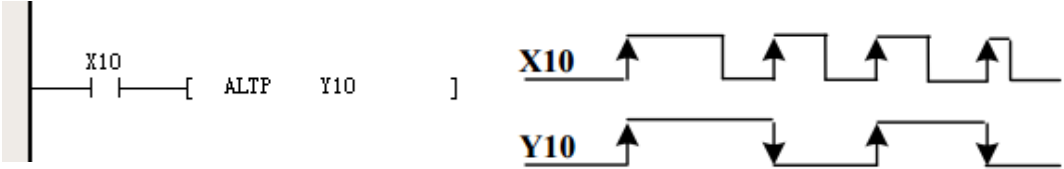
ALT D			交替输出	指令执行
D	执行装置	位元件		16 位指令（3step） ALT 连续执行 ALTP 脉冲执行

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

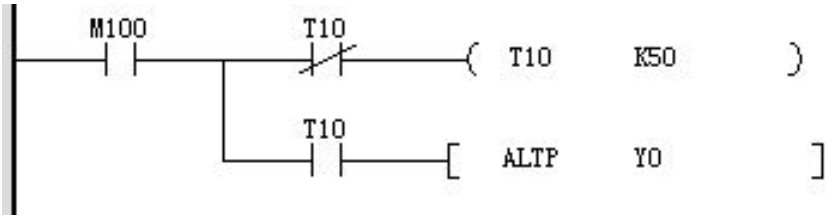
注：带灰色底纹软元件，表示可支持。

【使用举例 1】



【使用举例 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）。



4.1.4 其他处理指令

其他处理指令	
NOP	无动作
WDT	监视定时器复位

NOP

指令	操作数
NOP	无

功能说明：指令 **NOP** 在程序不做任何运算，因此执行后仍会保持原逻辑运算结果，没有实际操作。

4.2 程序流程指令

子程序	
CALL	子程序调用
SRET	子程序返回
SSRET	子程序带条件返回
IRET	中断返回
中断	
EI	中断许可
DI	中断禁止
跳转	
CJ	条件跳转
LBL	标号指令
CJEND	条件跳转到程序结束位置
循环	
FOR	循环范围开始
NEXT	循环范围结束

4.2.1 子程序

CALL 子程序调用

1. 指令形式

子程序调用指令

CALL S		子程序调用	指令执行
S	子程序名	子程序调用的目的指针标号	16 位指令（3step） CALL 连续执行 CALLP 脉冲执行

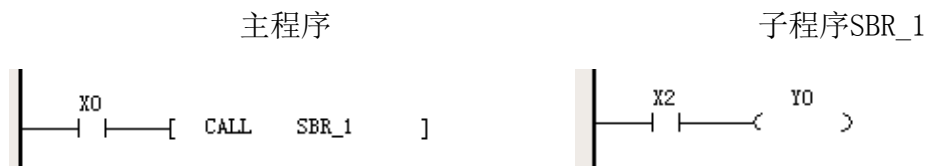
功能说明：当能流有效时，程序调用指定的子程序。子程序执行完毕，会返回到该 CALL（或 CALLP）语句的下一指令，继续执行后续语句。

- （1）子程序可被多处调用，也可被其他子程序调用，但嵌套层数不得超过 5 层。
- （2）在子程序内不得调用自身，防止死循环或程序运行超时。
- （3）在子程序中，可采用 T192~T199 或 T246~T249 作为定时器。

VCAutoDesignsoft 软件编程环境中子程序采用单独窗口编写，不存在 FEND、SRET 等指令问题，而且子程序名称支持任意修改（包括中文）。

例如：

可以通过子程序属性对话框将默认程序名改为一个更有意义的名称



SRET 子程序返回

1. 指令形式

子程序返回

SRET	子程序返回	指令执行
无需触点驱动，无操作数的单独指令		16 位指令（1step） SRET 连续执行

功能说明：在 VCAutoDesignsoft 软件编程环境中，无需用户输入 SRET 指令，系统在下载时会自动加入。

SSRET 子程序返回

1. 指令形式

子程序带条件返回

SSRET	子程序返回	指令执行
无需触点驱动，无操作数的单独指令		16 位指令（1step） SSRET 连续执行

功能说明：在 VCAutoDesignsoft 软件编程环境中，无需用户输入 SSRET 指令，系统在下载时会自动加入。

IRET 中断程序完毕

1. 指令形式

子程序带条件返回

IRET	中断程序完毕	指令执行
无需触点驱动，无操作数的单独指令		16 位指令（1step） IRET 连续执行

功能说明：IRET 语句位于中断子程序的结束处，执行了该指令后，会返回到调用该中断子程序前的语句处，继续程序执行。在 VCAutoDesignsoft 软件编程环境中，中断程序在单独窗口进行编写，无需用户输入 IRET 指令，系统在下载时会自动加入。

4.2.2 中断

EI/DI 中断允许/中断禁止

1. 指令形式

中断允许/中断禁止

EI	中断允许	指令执行
DI	中断禁止	
无需触点驱动，无操作数的单独指令		16 位指令（1step）EI/DI 连续执行

功能说明：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。

中断的种类与设置：

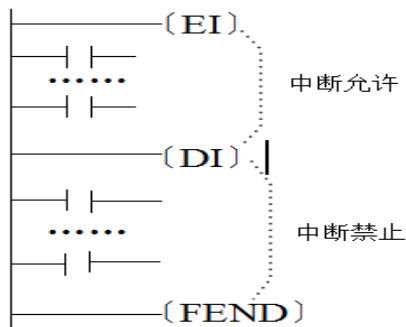
（1）外部信号输入中断：可定义 X0~X4 输入信号的上升沿或下降沿进行中断，对于不需要即时响应的 X 信号，还可以采用脉冲捕捉的功能；

（2）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；

（3）定时器中断：以 1ms~99ms 的固定周期发生的中断；

（4）脉冲完成中断：在指定脉冲个数发送完毕后，立即执行中断；

（5）多用户中断：可任意取一个高速计数器进行最多 24 个中断。



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。在中断子程序必须写在 FEND 指令之后，子程序尾部必须以 IRET 结束。在 VCAutoDesignsoft 软件编程环境中，不要写在主程序中，子程序尾可省略 IRET。
- 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、外部中断、时间中断、脉冲输出完成中断。
- 在中断例行程序的执行过程中，禁止其它的中断。
- 在中断处理过程中控制输入继电器及输出继电器时，使用输入输出刷新指令 (REFE)，可以通过读取最新的输入状态、或者立即输出运算结果，实现高速控制；
- 作为中断指针采用的输入继电器的编号，请不要与采用相同输入范围的[高速计数器][脉冲密度 (FNC56)] 等的应用命令的编号相重复。
- 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 1ms 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 REFE (FNC51) 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)。

详细内容请查看“中断介绍”。

4.2.3 跳转

CJ 条件跳转

1. 指令形式

条件满足时执行程序跳转的指令。

CJ/CJP P000~P511		条件跳转	指令执行
P	标签	条件转移目的指针标号	16 位指令（3step） CJ 连续执行 CJP 脉冲执行

注：操作数也可以使用 L，其作用与 P 等价。

功能说明：

- ① 当能流有效时，程序自动从 CJ（或 CJP）指令的地址跳转至由 P 标签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。
- ② 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。
- ③ 若被跨越的中间地址区的程序中有计数器，且已被驱动，则动作反应为：

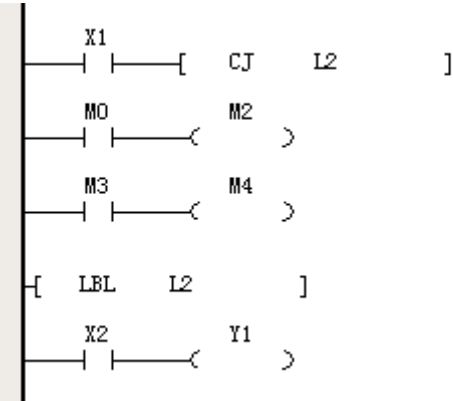
执行情况	CJ 有跳转	CJ 无跳转
T192~T199	正常执行	正常执行
其他定时器	停止计时	
C235~C255	正常执行	
其他计数器	停止计数	

对 P 标签的要求如下：

- CJ 指令必须与 LBL 指令配套使用，且目的标号必须位于当前程序块中，不能跨程序块跳转；
- P 标签的定义地址在同一个程序块中不能有重复；
- 当使用者希望某一部份程序不需要执行时，或者想使用两个线圈输出时，为避免双线圈的出现，可使用此指令；
- CJ 指令可重复指定同一指针 P。

指令举例：

在 VCAutoDesignsoft 软件编程环境中，跳转指令使用如下：



由于子程序与中断子程序在单独窗口编写，所以不必注意 FEND 等事项，跳转到结尾的指令在 VCAutoDesignsoft 软件编程环境中为 CJEND。

LBL 标号指令

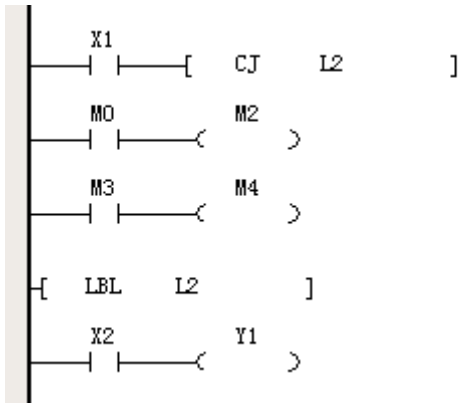
1. 指令形式

标号指令，与 CJ 指令配合使用，用于标记跳转的目标位置。

LBL P000~P511		条件跳转	指令执行
P	标签	条件转移的目标标号	16 位指令（3step） BLB 连续执行

注：操作数也可以使用 L，其作用与 P 等价。

【使用举例】



CJEND 条件跳转到程序末尾位置

1. 指令形式

条件满足时执行程序跳转到程序末尾位置，本扫描周期执行结束。

CJEND	条件跳转到程序结束	指令执行
无需触点驱动，无操作数的单独指令		16 位指令（3step） CJEND 连续执行

4.2.4 循环

FOR 循环范围开始

1. 指令形式
循环范围开始。

FOR S1			循环范围开始	指令执行
S1	循环次数	循环程序循环的次数		16 位指令 (3step) FOR 连续执行

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：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 循环范围结束

1. 指令形式

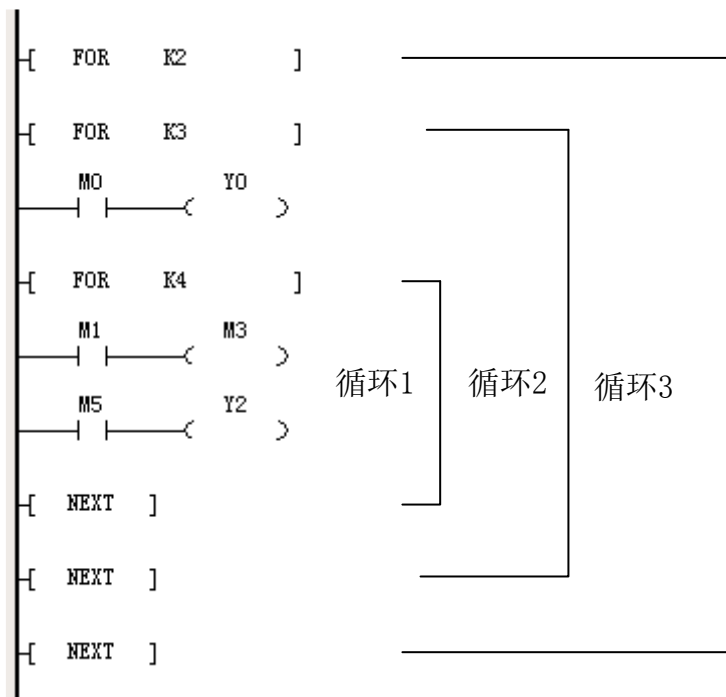
循环范围结束。

NEXT	循环范围开始	指令执行
无操作数的单独指令		16 位指令（1step） NEXT 连续执行

功能说明:

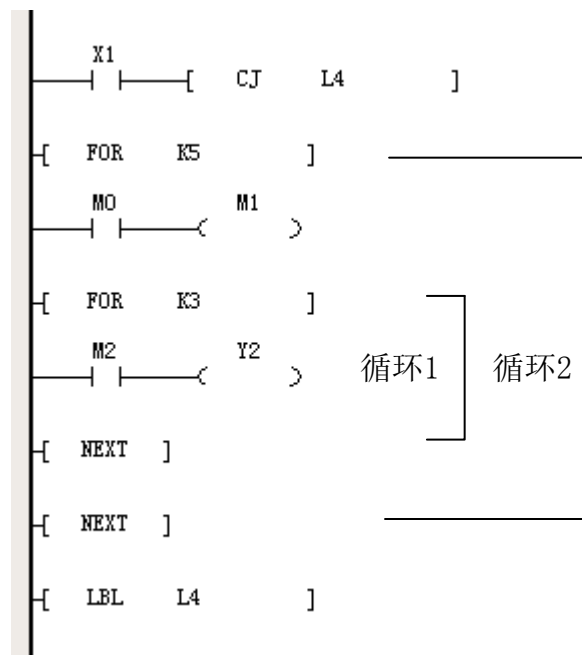
- NEXT 指令用于指示循环区域的尾部。由 FOR 指令指定 FOR~NEXT 循环来回执行 N 次后跳出 FOR~NEXT 循环往下继续执行。
- 在 FOR~NEXT 指令的循环区间，可以嵌入另一个 FOR~NEXT 循环，但规定：从最外层的 FOR~NEXT 计算，最多可内嵌 6 层 FOR~NEXT 循环。运行时 PLC 会以各 FOR~NEXT 层对应解析执行。但需要注意当循环次数过多时，会使 PLC 扫描周期延长，可能造成逾时监视定时器动作而导致错误产生。可在 FOR~NEXT 指令之间使用 WDT 指令来改善。
- 一个 FOR 指令对应一个 NEXT 指令，不能多也不能少，NEXT 指令不能写在 END/FEND 之后。FOR 指令在对应的 NEXT 指令之前。

【使用举例 1】



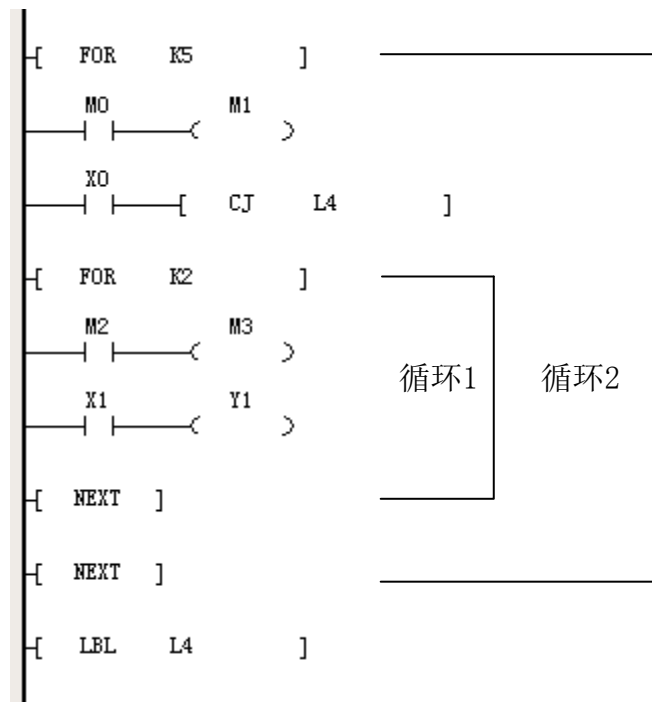
循环 3 执行 2 次后在到 NEXT 指令以后的程序继续执行，而循环 3 每执行一次循环 2 会执行 3 次，而循环 2 每执行一次循环 1 又执行 4 次，所以循环 1 共执行 $2*3*4=24$ 次，循环 2 共执行 $2*3=6$ 次。

【使用举例 2】



想跳过 FOR~NEXT 指令时，可用 CJ 跳转指令实现，范例中当 X1 为 OFF 时，执行循环 1 和循环 2 当 X0 为 ON 时，CJ 指令跳转至 L4 处，循环 1 和循环 2 之间的程序不被执行。

【使用举例 3】



想跳过循环内嵌套的 FOR~NEXT 指令或跳出循环时，也可用 CJ 跳转指令实现。当 X0 为 OFF 时，执行循环 2 内的循环 1，当 X1 为 ON 时，CJ 指令跳转至 L1 处，循环 2 内嵌套的循环 1FOR~NEXT 被 CJ 指令跳可不执行。

4.3 数据比较指令

数据比较指令可分为两大类：触点比较，比较输出。

触点指令	
LD=	LD 触点比较等于
LD>	LD 触点比较大于
LD<	LD 触点比较小于
LD<>	LD 触点比较不等于
LD>=	LD 触点比较大于等于
LD<=	LD 触点比较小于等于
AND=	AND 触点比较等于
AND>	AND 触点比较大于
AND<	AND 触点比较小于
AND<>	AND 触点比较不等于
AND>=	AND 触点比较大于等于
AND<=	AND 触点比较小于等于
OR=	OR 触点比较等于
OR>	OR 触点比较大于
OR<	OR 触点比较小于
OR<>	OR 触点比较不等于
OR>=	OR 触点比较大于等于
OR<=	OR 触点比较小于等于
LD&	LD 逻辑与运算
LD	LD 逻辑或运算
LD^	LD 逻辑异或运算
AND&	AND 逻辑与运算
AND	AND 逻辑或运算
AND^	AND 逻辑异或运算
OR&	OR 逻辑与运算
OR	OR 逻辑或运算
OR^	OR 逻辑异或运算
FLD>	浮点数>比较的状态触点 $S1 > S2$ 时导通
FLD>=	浮点数>=比较的状态触点 $S1 \geq S2$ 时导通
FLD<	浮点数<比较的状态触点 $S1 < S2$ 时导通
FLD<=	浮点数<=比较的状态触点 $S1 \leq S2$ 时导通
FLD=	浮点数=比较的状态触点 $S1 = S2$ 时导通
FLD<>	浮点数<>比较的状态触点 $S1 \neq S2$ 时导通
FAND>	浮点数>比较的与状态触点 $S1 > S2$ 时导通
FAND>=	浮点数>=比较的与状态触点 $S1 \geq S2$ 时导通
FAND<	浮点数<比较的与状态触点 $S1 < S2$ 时导通
FAND<=	浮点数<=比较的与状态触点 $S1 \leq S2$ 时导通
FAND=	浮点数=比较的与状态触点 $S1 = S2$ 时导通
FAND<>	浮点数<>比较的与状态触点 $S1 \neq S2$ 时导通

FOR>	浮点数>比较的或状态触点 $S1 > S2$ 时导通
FOR>=	浮点数>=比较的或状态触点 $S1 \geq S2$ 时导通
FOR<	浮点数<比较的或状态触点 $S1 < S2$ 时导通
FOR<=	浮点数<=比较的或状态触点 $S1 \leq S2$ 时导通
FOR=	浮点数=比较的或状态触点 $S1 = S2$ 时导通
FOR<>	浮点数<>比较的或状态触点 $S1 \neq S2$ 时导通
LDZ>	绝对值>比较的状态触点 $ S1-S2 > S3 $ 时导通
LDZ>=	绝对值>=比较的状态触点 $ S1-S2 \geq S3 $ 时导通
LDZ<	绝对值<比较的状态触点 $ S1-S2 < S3 $ 时导通
LDZ<=	绝对值<=比较的状态触点 $ S1-S2 \leq S3 $ 时导通
LDZ=	绝对值=比较的状态触点 $ S1-S2 = S3 $ 时导通
LDZ<>	绝对值<>比较的状态触点 $ S1-S2 \neq S3 $ 时导通
ANDZ>	绝对值>比较的与状态触点 $ S1-S2 > S3 $ 时导通
ANDZ>=	绝对值>=比较的与状态触点 $ S1-S2 \geq S3 $ 时导通
ANDZ<	绝对值<比较的与状态触点 $ S1-S2 < S3 $ 时导通
ANDZ<=	绝对值<=比较的与状态触点 $ S1-S2 \leq S3 $ 时导通
ANDZ=	绝对值=比较的与状态触点 $ S1-S2 = S3 $ 时导通
ANDZ<>	绝对值<>比较的与状态触点 $ S1-S2 \neq S3 $ 时导通
ORZ>	绝对值>比较的或状态触点 $ S1-S2 > S3 $ 时导通
ORZ>=	绝对值>=比较的或状态触点 $ S1-S2 \geq S3 $ 时导通
ORZ<	绝对值<比较的或状态触点 $ S1-S2 < S3 $ 时导通
ORZ<=	绝对值<=比较的或状态触点 $ S1-S2 \leq S3 $ 时导通
ORZ=	绝对值=比较的或状态触点 $ S1-S2 = S3 $ 时导通
ORZ<>	绝对值<>比较的或状态触点 $ S1-S2 \neq S3 $ 时导通
输出比较指令	
CMP	数据比较
ECMP	二进制浮点比较
ZCP	区域比较
EZCP	二进制浮点区间比较

4.3.1 触点比较

触点指令	
LD=	LD 触点比较等于
LD>	LD 触点比较大于
LD<	LD 触点比较小于
LD<>	LD 触点比较不等于
LD>=	LD 触点比较大于等于
LD<=	LD 触点比较小于等于
AND=	AND 触点比较等于
AND>	AND 触点比较大于
AND<	AND 触点比较小于
AND<>	AND 触点比较不等于
AND>=	AND 触点比较大于等于
AND<=	AND 触点比较小于等于
OR=	OR 触点比较等于
OR>	OR 触点比较大于
OR<	OR 触点比较小于
OR<>	OR 触点比较不等于
OR>=	OR 触点比较大于等于
OR<=	OR 触点比较小于等于
LD&	LD 逻辑与运算
LD	LD 逻辑或运算
LD^	LD 逻辑异或运算
AND&	AND 逻辑与运算
AND	AND 逻辑或运算
AND^	AND 逻辑异或运算
OR&	OR 逻辑与运算
OR	OR 逻辑或运算
OR^	OR 逻辑异或运算
FLD>	浮点数>比较的状态触点 S1>S2 时导通
FLD>=	浮点数>=比较的状态触点 S1 $\geq$ S2 时导通
FLD<	浮点数<比较的状态触点 S1<S2 时导通
FLD<=	浮点数<=比较的状态触点 S1 $\leq$ S2 时导通
FLD=	浮点数=比较的状态触点 S1=S2 时导通
FLD<>	浮点数<>比较的状态触点 S1 $\neq$ S2 时导通
FAND>	浮点数>比较的与状态触点 S1>S2 时导通
FAND>=	浮点数>=比较的与状态触点 S1 $\geq$ S2 时导通
FAND<	浮点数<比较的与状态触点 S1<S2 时导通
FAND<=	浮点数<=比较的与状态触点 S1 $\leq$ S2 时导通
FAND=	浮点数=比较的与状态触点 S1=S2 时导通
FAND<>	浮点数<>比较的与状态触点 S1 $\neq$ S2 时导通
FOR>	浮点数>比较的或状态触点 S1>S2 时导通

FOR>=	浮点数>=比较的或状态触点 $S1 \geq S2$ 时导通
FOR<	浮点数<比较的或状态触点 $S1 < S2$ 时导通
FOR<=	浮点数<=比较的或状态触点 $S1 \leq S2$ 时导通
FOR=	浮点数=比较的或状态触点 $S1 = S2$ 时导通
FOR<>	浮点数<>比较的或状态触点 $S1 \neq S2$ 时导通
LDZ>	绝对值>比较的状态触点 $ S1-S2 > S3 $ 时导通
LDZ>=	绝对值>=比较的状态触点 $ S1-S2 \geq S3 $ 时导通
LDZ<	绝对值<比较的状态触点 $ S1-S2 < S3 $ 时导通
LDZ<=	绝对值<=比较的状态触点 $ S1-S2 \leq S3 $ 时导通
LDZ=	绝对值=比较的状态触点 $ S1-S2 = S3 $ 时导通
LDZ<>	绝对值<>比较的状态触点 $ S1-S2 \neq S3 $ 时导通
ANDZ>	绝对值>比较的与状态触点 $ S1-S2 > S3 $ 时导通
ANDZ>=	绝对值>=比较的与状态触点 $ S1-S2 \geq S3 $ 时导通
ANDZ<	绝对值<比较的与状态触点 $ S1-S2 < S3 $ 时导通
ANDZ<=	绝对值<=比较的与状态触点 $ S1-S2 \leq S3 $ 时导通
ANDZ=	绝对值=比较的与状态触点 $ S1-S2 = S3 $ 时导通
ANDZ<>	绝对值<>比较的与状态触点 $ S1-S2 \neq S3 $ 时导通
ORZ>	绝对值>比较的或状态触点 $ S1-S2 > S3 $ 时导通
ORZ>=	绝对值>=比较的或状态触点 $ S1-S2 \geq S3 $ 时导通
ORZ<	绝对值<比较的或状态触点 $ S1-S2 < S3 $ 时导通
ORZ<=	绝对值<=比较的或状态触点 $ S1-S2 \leq S3 $ 时导通
ORZ=	绝对值=比较的或状态触点 $ S1-S2 = S3 $ 时导通
ORZ<>	绝对值<>比较的或状态触点 $ S1-S2 \neq S3 $ 时导通

LD※触点比较

1. 指令形式

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。

LD※ S1 S2			触点型数据比较	指令执行
S1	比较数 1	待比较的数据源或数据变量单元 1	16 位指令(5step) LD=连续执行	32 位指令(9step) LDD=连续执行
S2	比较数 2	待比较的数据源或数据变量单元 2		

注：※号为=、>、<、<>、>=、<=之一。

2. 操作数

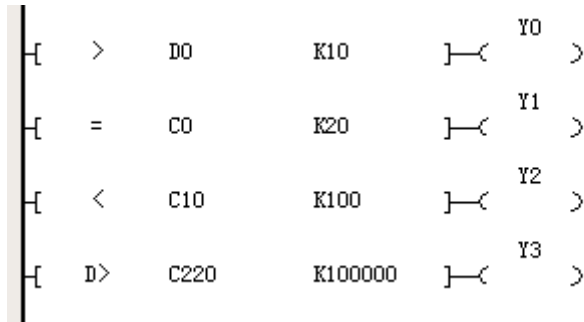
操作数	位软元件							字软元件														
	系统·用户							系统·用户				位数指定					变址		常数		实数	
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

LD 触点型比较指令有：

16bit 指令	32bit 指令	导通条件	非导通条件
LD=	LDD=	S1=S2	S1≠S2
LD>	LDD>	S1>S2	S1≤S2
LD<	LDD<	S1<S2	S1≥S2
LD<>	LDD<>	S1<>S2	S1=S2
LD≤	LDD≤	S1≤S2	S1>S2
LD≥	LDD≥	S1≥S2	S1<S2

【使用举例】



D0 的值大于 10 时，Y0 为 ON。

C0 的值等于 20 时，Y1 为 ON。

C10 的值小于 100 时，Y2 为 ON。

C220 的值大于 100000 时，Y3 为 ON。

注意:32 位计数器(C200～C255)的比较必须用 32 位(LDD=、LDD>、LDD<等)运算执行。

如指定了 16 位运算(LD=、LD>、LD<等)时，会发生程序错误或是运算错误。

AND※数据比较

1. 指令形式

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。

AND※ S1 S2			接点型数据比较	指令执行	
S1	比较数 1	待比较的数据源或数据变量单元 1	16 位指令（5step） AND=连续执行	32 位指令（9step）	ANDD=连续执行
S2	比较数 2	待比较的数据源或数据变量单元 2			

注：※号为=、>、<、<>、>=、<=之一。

2. 操作数

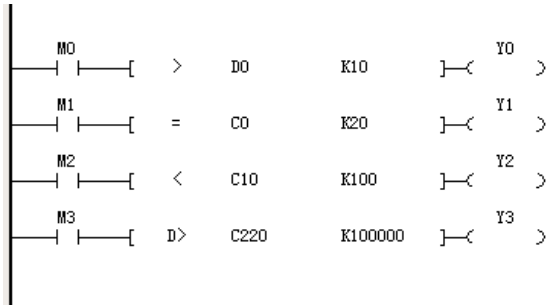
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

AND 触点型比较指令有：

16bit 指令	32bit 指令	导通条件	非导通条件
AND=	ANDD=	S1=S2	S1≠S2
AND>	ANDD>	S1>S2	S1≤S2
AND<	ANDD<	S1<S2	S1≥S2
AND<>	ANDD<>	S1<>S2	S1=S2
AND≤	ANDD≤	S1≤S2	S1>S2
AND≥	ANDD≥	S1≥S2	S1<S2

【使用举例】



当 M0 为 ON 且 D0 的值大于 10 时，Y0 为 ON。

当 M1 为 ON 且 C0 的值等于 20 时，Y1 为 ON。

当 M2 为 ON 且 C10 的值小于 100 时，Y2 为 ON。

当 M3 为 ON 且 C220 的值大于 100000 时，Y3 为 ON。

注意:32 位计数器(C200~C255)的比较必须用 32 位(ANDD=、ANDD>、ANDD<等)运算执行。如指定了 16 位运算(AND=、AND>、AND<等)时，会发生程序错误或是运算错误。

OR※数据比较

1. 指令形式

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。

OR※ S1 S2			并联接点型数据比较	指令执行	
S1	比较数 1	待比较的数据源或数据变量单元 1		16 位指令 (5step)	32 位指令 (9step)
S2	比较数 2	待比较的数据源或数据变量单元 2		OR=连续执行	ORD=连续执行

注：※号为=、>、<、<>、>=、<=之一。

2. 操作数

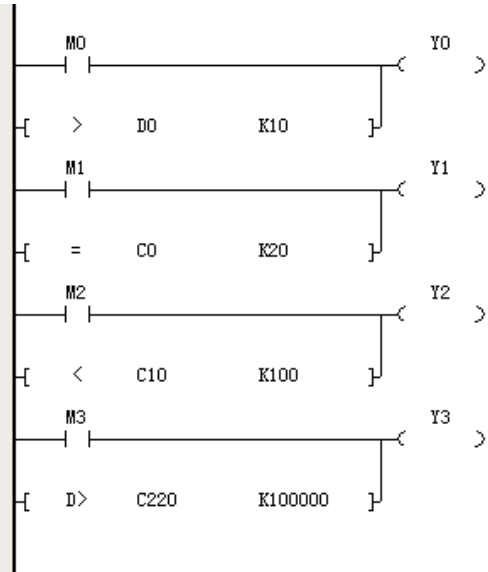
操作数	位软元件							字软元件														
	系统·用户							系统·用户				位数指定					变址		常数		实数	
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

OR 触点型比较指令有：

16bit 指令	32bit 指令	导通条件	非导通条件
OR=	ORD=	S1=S2	S1≠S2
OR>	ORD>	S1>S2	S1≤S2
OR<	ORD<	S1<S2	S1≥S2
OR<>	ORD<>	S1<>S2	S1=S2
OR≤	ORD≤	S1≤S2	S1>S2
OR≥	ORD≥	S1≥S2	S1<S2

【使用举例】



当 M0 为 ON 或 D0 的值大于 10 时，Y0 为 ON。

当 M1 为 ON 或 C0 的值等于 20 时，Y1 为 ON。

当 M2 为 ON 或 C10 的值小于 100 时，Y2 为 ON。

当 M3 为 ON 或 C220 的值大于 100000 时，Y3 为 ON。

注意:32 位计数器(C200~C255)的比较必须用 32 位(ORD=、ORD>、ORD<等)运算执行。如指定了 16 位运算(OR=、OR>、OR<等)时，会发生程序错误或是运算错误。

LD※触点状态位运算

1. 指令形式

位逻辑运算结果作为该触点导通状态，与左母线直接相连的节点。

LD※ S1 S2			触点状态位运算	指令执行	
S1	数据 1	源数据 1 软元件编号		16 位指令（5step）	32 位指令（9step）
S2	数据 2	源数据 2 软元件编号		LD※连续执行	LDD※连续执行

注：※为&、|、^之一。

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

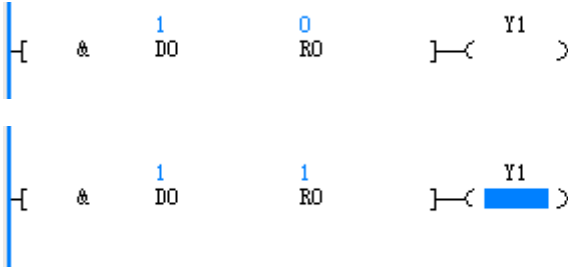
注：带灰色底纹软元件，表示可支持。

功能说明：[S1]与[S2]的内容进行逻辑运算(“与&”、“或|”、“异或^”)的指令，结果不为 0 时，该指令导通；比较结果为 0 时，该指令不导通。

运算结果如下：

16 位指令	32 位指令	导通条件	非导通条件
LD&	LDD&	$S1 \& S2 \neq 0$	$S1 \& S2 = 0$
LD	LDD	$S1 S2 \neq 0$	$S1 S2 = 0$
LD^	LDD^	$S1 \wedge S2 \neq 0$	$S1 \wedge S2 = 0$

【使用举例】



AND※触点状态位运算

1. 指令形式

位逻辑运算结果作为该触点导通状态，与其它节点串联连接的节点。

AND※ S1 S2			与触点状态位运算	指令执行	
S1	数据 1	源数据 1 软元件编号		16 位指令（5step）	32 位指令（9step）
S2	数据 2	源数据 2 软元件编号		AND※连续执行	AND※连续执行

注：※为&、|、^之一。

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

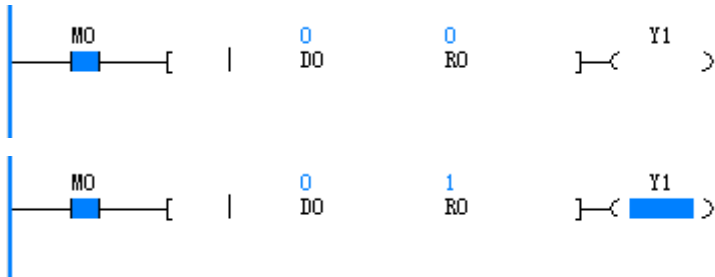
注：带灰色底纹软元件，表示可支持。

功能说明：[S1]与[S2]的内容进行逻辑运算(“与&”、“或|”、“异或^”)的指令，结果不为 0 时，该指令导通；比较结果为 0 时，该指令不导通。

运算结果如下：

16 位指令	32 位指令	导通条件	非导通条件
AND&	ANDD&	S1&S2≠0	S1&S2=0
AND	ANDD	S1 S2≠0	S1 S2=0
AND^	ANDD^	S1^S2≠0	S1^S2=0

【使用举例】



OR※触点状态位运算

1. 指令形式

位逻辑运算结果作为该触点导通状态，与其它节点并联连接的节点。

OR※ S1 S2			触点状态位运算	指令执行	
S1	数据 1	源数据 1 软元件编号		16 位指令（5step）	32 位指令（9step）
S2	数据 2	源数据 2 软元件编号		OR※连续执行	ORD※连续执行

注：※为&、|、^之一。

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

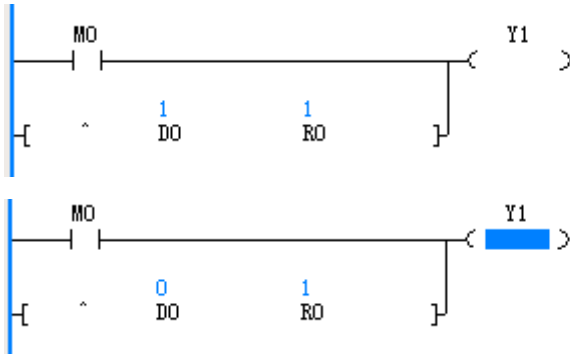
注：带灰色底纹软元件，表示可支持。

功能说明：[S1]与[S2]的内容进行逻辑运算(“与&”、“或|”、“异或^”)的指令，结果不为 0 时，该指令导通；比较结果为 0 时，该指令不导通。

运算结果如下：

16 位指令	32 位指令	导通条件	非导通条件
OR&	ORD&	S1&S2≠0	S1&S2=0
OR	ORD	S1 S2≠0	S1 S2=0
OR^	ORD^	S1^S2≠0	S1^S2=0

【使用举例】



FLD※浮点数触点比较

1. 指令形式

比较两操作数大小，根据比较结果把触点置 ON 或者 OFF，与左母线直接相连的节点。

FLD※ S1 S2			浮点数触点比较	指令执行
S1	数据 1	源数据 1 软元件编号		32 位指令（9step） FLDD※连续执行
S2	数据 2	源数据 2 软元件编号		

注：※为=、>、<、<>、>=、<=之一。

2. 操作数

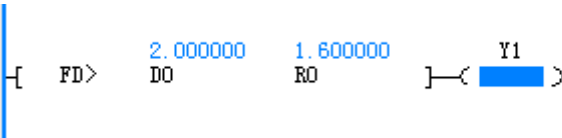
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：[S1]与[S2]比较的指令。条件满足时导通，否则不导通。

32 位指令	导通条件	非导通条件
FLDD>	S1>S2	S1<=S2
FLDD>=	S1>=S2	S1<S2
FLDD<	S1<S2	S1>=S2
FLDD<=	S1<=S2	S1>S2
FLDD=	S1=S2	S1<>S2
FLDD<>	S1<>S2	S1=S2

【使用举例】



FAND※浮点数与触点比较

1. 指令形式

比较两操作数大小，根据比较结果把触点置 ON 或者 OFF，与其它节点串联连接的节点。

FAND※ S1 S2			浮点数与触点比较	指令执行
S1	数据 1	源数据 1 软元件编号		32 位指令（9step） FAND※连续执行
S2	数据 2	源数据 2 软元件编号		

注：※为=、>、<、<>、>=、<=之一。

2. 操作数

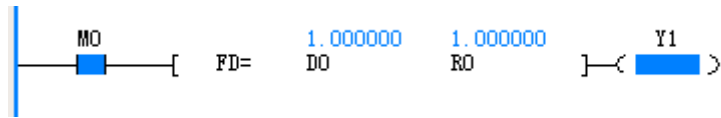
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：[S1]与[S2]比较的指令。条件满足时导通，否则不导通。

32 位指令	导通条件	非导通条件
FANDD>	S1>S2	S1<=S2
FANDD>=	S1>=S2	S1<S2
FANDD<	S1<S2	S1>=S2
FANDD<=	S1<=S2	S1>S2
FANDD=	S1=S2	S1<>S2
FANDD<>	S1<>S2	S1=S2

【使用举例】



FOR※浮点数或触点比较

1. 指令形式

比较两操作数大小，根据比较结果把触点置 ON 或者 OFF，与其它节点并联连接的节点。

FOR※		S1	S2	浮点数或触点比较	指令执行
S1	数据 1	源数据 1 软元件编号			32 位指令（9step） FORD※连续执行
S2	数据 2	源数据 2 软元件编号			

注：※为=、>、<、<>、>=、<=之一。

2. 操作数

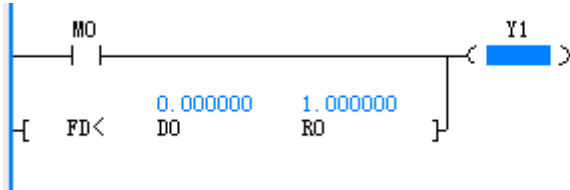
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址			常数	
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：[S1]与[S2]比较的指令。条件满足时导通，否则不导通。

32 位指令	导通条件	非导通条件
FORD>	S1>S2	S1<=S2
FORD>=	S1>=S2	S1<S2
FORD<	S1<S2	S1>=S2
FORD<=	S1<=S2	S1>S2
FORD=	S1=S2	S1<>S2
FORD<>	S1<>S2	S1=S2

【使用举例】



LDZ※绝对值比较触点

1. 指令形式

将 S1 与 S2 相减后结果的绝对值与 S3 的绝对值进行比较，根据比较结果把触点置 ON 或者 OFF，与左母线直接相连的节点。

LDZ※ S1 S2 S3				绝对值比较触点	指令执行	
S1	被减数	被减数来源元件			16 位指令（5step） LDZ※连续执行	32 位指令（9step） LDDZ※连续执行
S2	减数	减数来源元件				
S3	比较值	比较值来源元件				

注：※为=、>、<、<>、>=、<=之一。

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S3	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

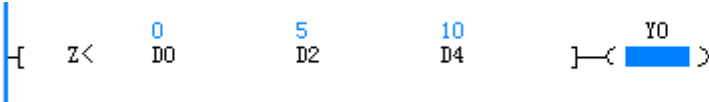
注：带灰色底纹软元件，表示可支持。

功能说明：

[S1]与[S2]相减后结果的绝对值与[S3]的绝对值作比较的指令。条件满足时导通，否则不导通。

【使用举例】

16 位指令	32 位指令	导通条件	非导通条件
LDZ>	LDDZ>	S1-S2 > S3	S1-S2 <= S3
LDZ>=	LDDZ>=	S1-S2 >= S3	S1-S2 < S3
LDZ<	LDDZ<	S1-S2 < S3	S1-S2 >= S3
LDZ<=	LDDZ<=	S1-S2 <= S3	S1-S2 > S3
LDZ=	LDDZ=	S1-S2 = S3	S1-S2 <> S3
LDZ<>	LDDZ<>	S1-S2 <> S3	S1-S2 = S3



ANDZ※绝对值比较与触点

1. 指令形式

将 S1 与 S2 相减后结果的绝对值与 S3 的绝对值进行比较，根据比较结果把触点置 ON 或者 OFF，与其他节点串联连接的节点。

ANDZ※ S1 S2 S3				绝对值比较与触点	指令执行	
S1	被减数	被减数来源元件			16 位指令（5step） ANDZ※连续执行	32 位指令（9step） ANDDDZ※连续执行
S2	减数	减数来源元件				
S3	比较值	比较值来源元件				

注：※为=、>、<、<>、>=、<=之一。

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

[S1]与[S2]相减后结果的绝对值与[S3]的绝对值作比较的指令。条件满足时导通，否则不导通。

【使用举例】

16 位指令	32 位指令	导通条件	非导通条件
ANDZ>	ANDDDZ>	S1-S2 > S3	S1-S2 <= S3
ANDZ>=	ANDDDZ>=	S1-S2 >= S3	S1-S2 < S3
ANDZ<	ANDDDZ<	S1-S2 < S3	S1-S2 >= S3
ANDZ<=	ANDDDZ<=	S1-S2 <= S3	S1-S2 > S3
ANDZ=	ANDDDZ=	S1-S2 = S3	S1-S2 <> S3
ANDZ<>	ANDDDZ<>	S1-S2 <> S3	S1-S2 = S3



ORZ※绝对值比较或触点

1. 指令形式

将 S1 与 S2 相减后结果的绝对值与 S3 的绝对值进行比较，根据比较结果把触点置 ON 或者 OFF，与其他节点并联连接的节点。

ORZ※ S1 S2 S3				绝对值比较或触点	指令执行	
S1	被减数	被减数来源元件			16 位指令（5step） ORZ※连续执行	32 位指令（9step） ORDZ※连续执行
S2	减数	减数来源元件				
S3	比较值	比较值来源元件				

注：※为=、>、<、<>、>=、<=之一。

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

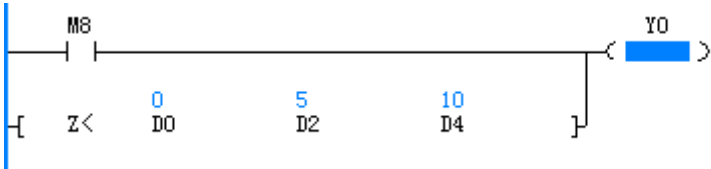
注：带灰色底纹软元件，表示可支持。

功能说明：

[S1]与[S2]相减后结果的绝对值与[S3]的绝对值作比较的指令。条件满足时导通，否则不导通。

【使用举例】

16 位指令	32 位指令	导通条件	非导通条件
ORZ>	ORDZ>	S1-S2 > S3	S1-S2 <= S3
ORZ>=	ORDZ>=	S1-S2 >= S3	S1-S2 < S3
ORZ<	ORDZ<	S1-S2 < S3	S1-S2 >= S3
ORZ<=	ORDZ<=	S1-S2 <= S3	S1-S2 > S3
ORZ=	ORDZ=	S1-S2 = S3	S1-S2 <> S3
ORZ<>	ORDZ<>	S1-S2 <> S3	S1-S2 = S3



4.3.2 输出比较

输出比较指令	
CMP	数据比较
ECMP	二进制浮点比较
ZCP	区域比较
EZCP	二进制浮点区间比较

CMP 数据比较

1. 指令形式

当驱动条件成立时，比较 S1 与 S2 的大小，并根据比较结果（S1>S2，S1=S2，S1<S2）置终址位元件 D，D+1，D+2 其中一个为 ON。

CMP S1 S2 D				两个数比较	指令执行	
S1	比较值 1	比较值 1 数据或数据存储字软元件地址			16 位指令(7step) 32 位指令(13step) CMP 连续执行 CMPP 脉冲执行	DCMP 连续执行 DCMPP 脉冲执行
S2	比较值 2	比较值 2 数据或数据存储字软元件地址				
D	比较结果	比较结果 ON/OFF 位首址，占用连续 3 个位				

2. 操作数

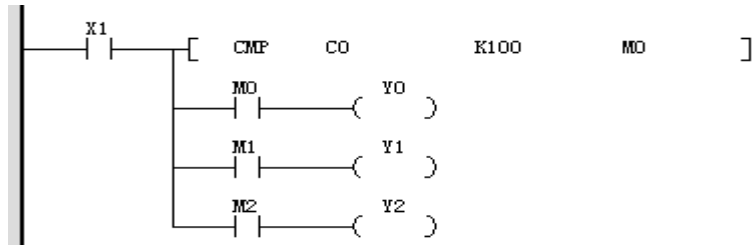
操作数	位软元件							字软元件												
	系统·用户							系统·用户					位数指定					变址	常数	实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K H E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K H E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K H E

注：带灰色底纹软元件，表示可支持。

功能说明：

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。其中 D 会占用 3 个连续地址的位变量。

【使用举例】



当 X1 为 ON 是，C0 大于 100，M0 为 ON,Y0 有输出。

当 X1 为 ON 是，C0 等于 100，M1 为 ON,Y1 有输出。

当 X1 为 ON 是，C0 小于 100，M2 为 ON,Y2 有输出。

X1 由 ON 变 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。

ECMP 二进制浮点比较

1. 指令形式

进行 2 个浮点数变量的比较，将比较的结果输出到 D 启始的 3 个变量中。

ECMP S1 S2 D				二进制浮点比较	指令执行
S1	比较值 1	比较值 1 数据或数据存储字软元件地址			32 位指令（13step） DECMP 连续执行
S2	比较值 2	比较值 2 数据或数据存储字软元件地址			
D	比较结果	比较结果存放单元，占用连续 3 个（位）变量单元			

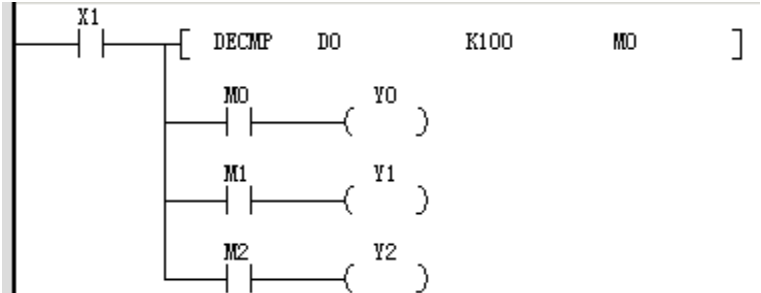
2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：该指令进行 2 个浮点数变量的比较，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。其中 D 会占用 3 个连续地址的位变量。

【使用举例】



当 X1 为 ON 是，(D1,D0)大于 100，M0 为 ON,Y0 有输出。

当 X1 为 ON 是，(D1,D0)等于 100，M1 为 ON,Y1 有输出。

当 X1 为 ON 是，(D1,D0)小于 100，M2 为 ON,Y2 有输出。

X1 由 ON 变 OFF 时，不执行 DECMP 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 $\geq$ 、 $\leq$ 、 $\neq$ 的结果时，可将 M0~M2 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。

ZCP 区间比较

1. 指令形式

当驱动条件成立时，根据 S 所在的区间（ $S < S1$ ， $S1 \leq S \leq S2$ ， $S > S2$ ）置终址位元件 D、D+1，D+2 其中一个为 ON。

ZCP S1 S2 S D					区间比较	指令执行	
S1	区间比较下线	数据或数据存储字软元件地址			16 位指令(9step) ZCP 连续执行 ZCPP 脉冲执行	32 位指令(17step) DZCP 连续执行 DZCPP 脉冲执行	
S2	区间比较上线	数据或数据存储字软元件地址					
S	比较变量	数据或数据存储字软元件地址					
D	比较结果	比较结果 ON/OFF 位首址，占用连续 3 个位					

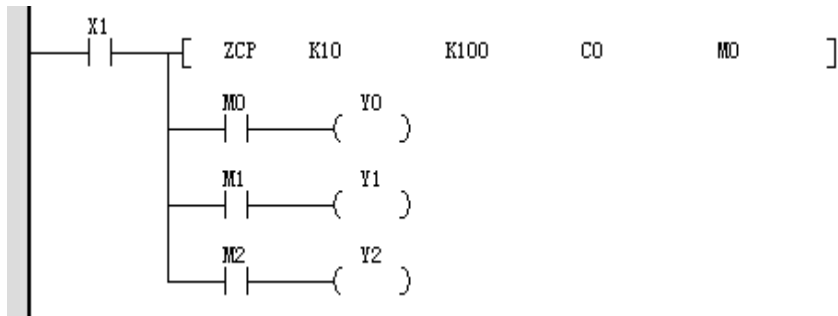
2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：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。其中 D 会占用 3 个连续地址的位变量。

【使用举例】



当 X1 为 ON 是，C0 小于 10，M0 为 ON,Y0 有输出。

当 X1 为 ON 是，C0 大于等于 10 且小于等于 100，M1 为 ON,Y1 有输出。

当 X1 为 ON 是，C0 大于 100，M2 为 ON,Y2 有输出。

X1 由 ON 变 OFF 时，不执行 ZCP 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。

EZCP 二进制浮点区间比较

1. 指令形式

进行二进制浮点数变量的区间比较，将比较的结果输出到 D 启始的 3 个变量中。

EZCP S1 S2 S D					二进制浮点区间比较	指令执行
S1	区间比较下线	二进制浮点变量区间的下限			32 位指令（17step） DEZCP 连续执行 DEZCPP 脉冲执行	
S2	区间比较上线	二进制浮点变量区间的上限				
S	比较变量	二进制浮点变量区间的变量				
D	比较结果	比较结果 ON/OFF 位首址，占用连续 3 个位				

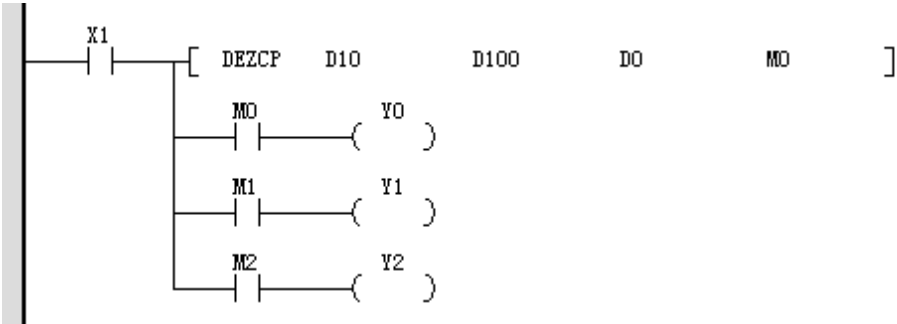
2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户				位数指定					变址		常数		实数	
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：本指令进行二进制浮点数变量的区间比较，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。其中 D 会占用 3 个连续地址的位变量。

【使用举例】



当 X1 为 ON 是，(D1,D0)小于(D11,D10)，M0 为 ON,Y0 有输出。

当 X1 为 ON 是，(D1,D0)大于等于(D11,D10)且小于等于(D101,D100)，M1 为 ON,Y1 有输出。

当 X1 为 ON 是，(D1,D0)大于(D101,D100)，M2 为 ON,Y2 有输出。

X1 由 ON 变 OFF 时，不执行 DEZCP 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。

4.4 数据运算

数据运算分为以下几类：四则运算；数据逻辑运算；三角函数；指数运算。

四则运算	
ADD	二进制数据加法
SUB	二进制数据减法
MUL	二进制数据乘法
DIV	二进制数据除法
EADD	二进制浮点加法
ESUB	二进制浮点减法
EMUL	二进制浮点乘法
EDIV	二进制浮点除法
INC	二进制数据加一
DEC	二进制数据减一
数据逻辑运算	
WAND	二进制数据逻辑与
WOR	二进制数据逻辑或
WXOR	二进制数据逻辑异或
NEG	二进制数据求补
ENEG	二进制浮点数符号取反
三角函数	
SIN	浮点 SIN 运算指令
COS	浮点 COS 运算指令指令
TAN	浮点 TAN 运算指令
ASIN	二进制浮点数 ARCSIN 运算
ACOS	二进制浮点数 ARCCOS 运算
ATAN	二进制浮点数 ARCTAN 运算
RAD	二进制浮点数角度→弧度的转换
DEG	二进制浮点数弧度→角度的转换
SINH	二进制浮点数 SINH 运算
COSH	二进制浮点数 COSH 运算
TANH	二进制浮点数 TANH 运算
指数运算	
EXP	二进制浮点数指数运算
LOGE	二进制浮点数自然对数运算
LOG	二进制浮点数以 10 为底的对数运算
ESQR	二进制浮点开方运算
SQR	二进制数据开方运算

4.4.1 四则运算

四则运算	
ADD	二进制数据加法
SUB	二进制数据减法
MUL	二进制数据乘法
DIV	二进制数据除法
EADD	二进制浮点加法
ESUB	二进制浮点减法
EMUL	二进制浮点乘法
EDIV	二进制浮点除法
INC	二进制数据加一
DEC	二进制数据减一

ADD 二进制数据加法

1. 指令形式

二进制加法指令。

ADD S1 S2 D				二进制数据加法	指令执行	
S1	被加数	数据或数据存储字软元件地址			16 位指（7step） ADD 连续执行 ADDP 脉冲执行	32 位指（13step）
S2	加数	数据或数据存储字软元件地址				DADD 连续执行
D	和	数据存储字软元件地址				DADDP 脉冲执行

2. 操作数

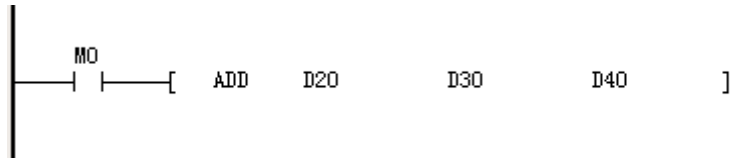
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 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。
- 若计算结果为 0，则 0 标志 (M8020) 会置位；
- 若计算结果超过 32767 (16bit 运算) 或 2147483647 (32bit 运算) 时，进位标志 (M8022) 会置位；
- 若计算结果不满 -32768 (16bit 运算) 或 -2147483648 (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。

【使用举例】



当 M0 为 ON 时，D20 的内容加上 D30 的内容放在 D40 中。

SUB 二进制数据减法

1. 指令形式

二进制减法指令。

SUB S1 S2 D				二进制数据减法	指令执行	
S1	被减数	数据或数据存储字软元件地址			16 位指（7step）	32 位指（13step）
S2	减数	数据或数据存储字软元件地址			SUB 连续执行	DSUB 连续执行
D	差	数据存储字软元件地址			SUBP 脉冲执行	DSUBP 脉冲执行

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户				位数指定					变址		常数		实数	
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 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 32767 (16bit 运算) 或 2147483647 (32bit 运算) 时，进位标志 (M8022) 会置位；
- 若计算结果不满 -32768 (16bit 运算) 或 -2147483648 (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。

【使用举例】



当 M0 为 ON 时，D20 的内容减去 D30 的内容存放在 D40 中。

MUL 二进制数据乘法

1. 指令形式

二进制乘法指令。

MUL S1 S2 D				二进制数据乘法	指令执行	
S1	被乘数	数据或数据存储字软元件地址			16 位指（7step） MUL 连续执行 MULP 脉冲执行	32 位指（13step） DMUL 连续执行 DMULP 脉冲执行
S2	乘数	数据或数据存储字软元件地址				
D	积	数据存储字软元件地址，16 位指令积为 32 位数据， 32 位指令积为 64 位数据。				

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

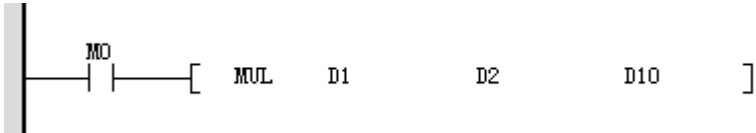
注：带灰色底纹软元件，表示可支持。

功能说明：

- 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。
- 进行 32bit 运算时，指令中变量地址位为低 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。

S1(16bin)	*	S2(16bin)	==	D+1,D(32bin)
S1(32bin)	*	S1(32bin)	==	D+3,D+2,D+1,D(64bin)

【使用举例】



当 M0 为 ON 时，D1 中的数乘 D2 中的数存放在（D11,D10）中。

当结果小于 16bin，存放在 D10 中。

当结果大于 16bin，存放在 D11,D10 中。

DIV 二进制数据除法

1. 指令形式

二进制除法指令。

DIV S1 S2 D				二进制数据除法	指令执行	
S1	被除数	数据或数据存储字软元件地址			16 位指令(7step) DIV 连续执行 DIVP 脉冲执行	32 位指令(13step) DDIV 连续执行 DDIVP 脉冲执行
S2	除数	数据或数据存储字软元件地址				
D	商及 余数	数据存储字软元件地址， 商存放在 D，余数存放在 D+1。				

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

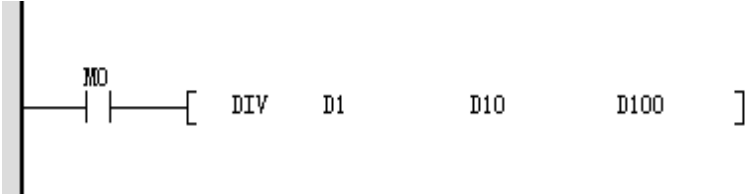
注：带灰色底纹软元件，表示可支持。

功能说明：

- 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。
- 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（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。

S1(16bin)	/	S2(16bin)	==	商 D(16bin)	余数 D+1(16bin)
S1(32bin)	/	S1(32bin)	==	商	余数 D+3,D+2(32bin)

【使用举例】



当 M0 为 ON 时，D1 中的数除以 D10 中的数，商存放在 D100 中，余数存放 D101 中。

EADD 二进制浮点加法

1. 指令形式

二进制浮点加法运算。

EADD S1 S2 D				二进制浮点加法	指令执行
S1	被加数	二进制浮点被加数			32 位指（13step） DEADD 连续执行 DEADDP 脉冲执行
S2	加数	二进制浮点加数			
D	和	加法和存放单元			

2. 操作数

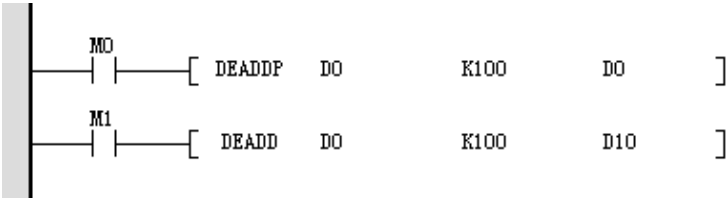
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 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加法运算；
- 若计算结果为 0，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。

【使用举例】



当 M0 为 ON 时，D1,D0 的值加上 100 后存放在 D1,D0 中。常数 K100 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。

当 M1 为 ON 时，D1,D0 的值加上 100 后存放在 D11,D10 中。

ESUB 二进制浮点减法

1. 指令形式

二进制浮点将减法运算。

ESUB S1 S2 D				二进制浮点减法	指令执行
S1	被减数	二进制浮点被减数			32 位指令（13step） DESUB 连续执行 DESUBP 脉冲执行
S2	减数	二进制浮点减数			
D	差	差的存放单元			

2. 操作数

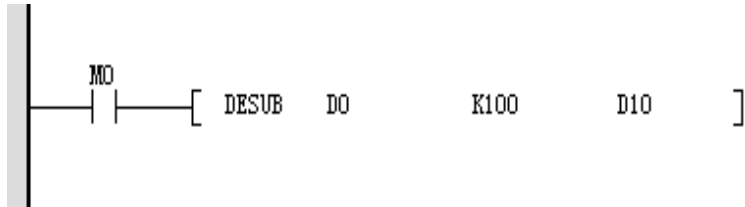
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 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加法运算；
- 若计算结果为 0，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。

【使用举例】



当 M0 为 ON 时，D1,D0 的内容减去 100 后存放在 D11，D10 中。

EMUL 二进制浮点乘法

1. 指令形式

二进制浮点乘法运算。

EMUL S1 S2 D				二进制浮点乘法	指令执行
S1	被乘数	二进制浮点被乘数			32 位指（13step） DEMUL 连续执行 DEMULP 脉冲执行
S2	乘数	二进制浮点乘数			
D	积	积的存放单元			

2. 操作数

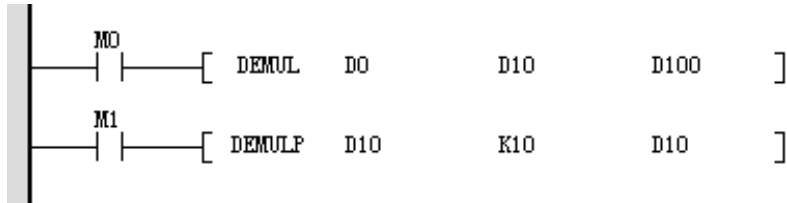
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 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。

【使用举例】



当 M0 为 ON 时，D1,D0 中的数乘 D11,D10 中的数存放在 D101,D100 中。

当 M1 为 ON 时，D11,D10 中的数乘以 10 后积存放在 D11,D10 中。

常数 10 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。

EDIV 二进制浮点除法

1. 指令形式

二进制浮点除法运算。

EDIV S1 S2 D				二进制浮点除法	指令执行
S1	被除数	二进制浮点的被除数			32 位指（13step） DEDIV 连续执行 DEDIVP 脉冲执行
S2	除数	二进制浮点的除数			
D	商	二进制浮点除法商的存放单元起始地址			

2. 操作数

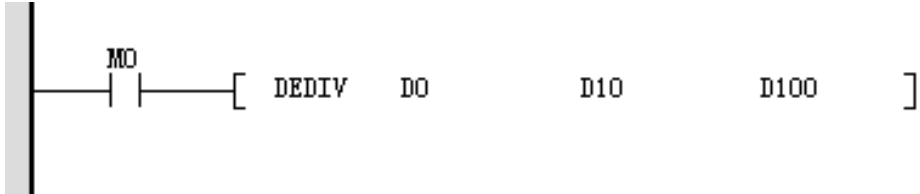
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 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。

【使用举例】



当 M0 为 ON 时，二进制浮点数（D1，D0）除以二进制浮点数（D11，D10）后，二进制浮点数商存放于（D101，D100）中。

INC 二进制数据加一

1. 指令形式

二进制加一指令。

INC D							二进制数据加一		指令执行	
D	累加结果		数据存储字软元件地址					16 位指（3step） INC 连续执行 INCP 脉冲执行	32 位指（5step） DINC 连续执行 DINCP 脉冲执行	

2. 操作数

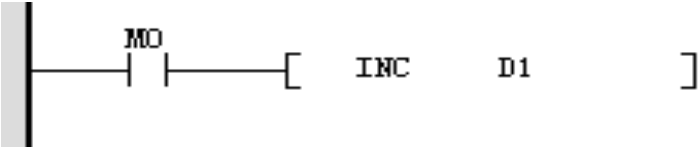
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 指令每执行一次，D 中的数值增加 1。
- 16 位运算时，32767 再加 1 变为-32768；
- 32 位运算时，2147483647 再加 1 变为-2147483648。
- 本指令对 0 标志、进位、借位标志都不刷新。
- 进行 32bit 运算时，指令中 D 变量地址位为低 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。

【使用举例】



当 MO 为 ON 一次，D1 的值加 1。

DEC 二进制数据减一

1. 指令形式

二进制减一指令。

DEC D			二进制数据减一	指令执行	
D	累减结果	数据存储字软元件地址		16 位指（3step） DEC 连续执行 DECP 脉冲执行	32 位指（5step） DDEC 连续执行 DDECP 脉冲执行

2. 操作数

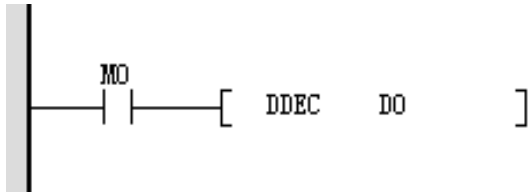
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 指令每执行一次，D 中的数值减 1。
- 16 位运算时，-32768 再减 1 变为 32767；
- 32 位运算时，-2147483648 再减 1 变为 2147483647。
- 本指令对 0 标志、进位、借位标志都不刷新。
- 进行 32bit 运算时，指令中 D 变量地址位为低 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。

【使用举例】



当 MO 为 ON 一次，D1,D0 的值减 1。

4.4.2 数据逻辑运算

数据逻辑运算	
WAND	二进制数据逻辑与
WOR	二进制数据逻辑或
WXOR	二进制数据逻辑异或
NEG	二进制数据求补
ENEG	二进制浮点数符号取反

WAND 二进制数据逻辑与

1. 指令形式

驱动条件成立时，将 S1 和 S2 按位进行逻辑与运算，并将结果存于 D。

WAND S1 S2 D				二进制数据逻辑与	指令执行	
S1	数据 1	参与与运算的数据或数据存储字软元件地址			16 位指令（7step） WAND 连续执行 WANDP 脉冲执行	32 位指（13step）
S2	数据 2	参与与运算的数据或数据存储字软元件地址				DAND 连续执行
D	运算结果	运算结果数据存储字软元件地址				DANDP 脉冲执行

2. 操作数

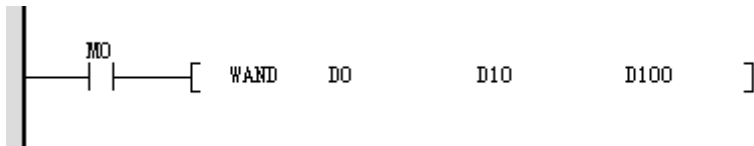
操作数	位软元件							字软元件												
	系统·用户							系统·用户					位数指定					变址	常数	实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K H E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K H E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K H E

注：带灰色底纹软元件，表示可支持。

功能说明：

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 D 变量。逻辑的‘与’(AND)运算的规则为任一为 0 结果为 0。

【使用举例】



当 M0 为 ON 时，将 D0 和 D10 以 bin 为单位进行与运算将结果存放在 D100 中。

	D0(16bin)	D10 (16bin)	D100 (16bin)
位 (bin) 单位的 逻辑与运算	0	0	0
	0	1	0
	1	0	0
	1	1	1

WOR 二进制数据逻辑或

1. 指令形式

驱动条件成立时，将 S1 和 S2 按位进行逻辑或运算，并将结果存于 D。

WOR S1 S2 D			二进制数据逻辑或	指令执行	
S1	数据 1	参与或运算的数据或数据存储字软元件地址		16 位指令（7step） WOR 连续执行 WOP 脉冲执行	32 位指（13step） DOR 连续执行 DORP 脉冲执行
S2	数据 2	参与或运算的数据或数据存储字软元件地址			
D	运算结果	运算结果数据存储字软元件地址			

2. 操作数

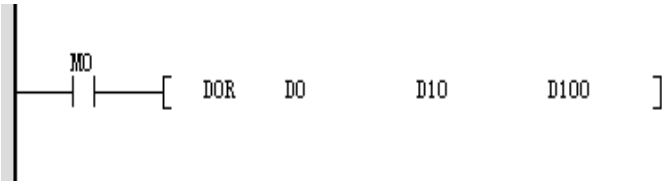
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

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。

【使用举例】



当 M0 为 ON 时，将 D1,D0 和 D11,D10 以 bin 为单位进行或运算结果存放在 D101,D100 中。

	D1,D0(32bin)	D11,D10 (32bin)	D101,D100 (32bin)
位 (bin) 单位的 逻辑或运算	0	0	0
	0	1	1
	1	0	1
	1	1	1

WXOR 二进制数据逻辑异或

1. 指令形式

驱动条件成立时，将 S1 和 S2 按位进行逻辑异或运算，并将结果存于 D。

WXOR S1 S2 D			二进制数据逻辑异或	指令执行	
S1	数据 1	参与异或运算的数据或数据存储字软元件地址		16 位指令（7step）	32 位指（13step）
S2	数据 2	参与异或运算的数据或数据存储字软元件地址			
D	运算结果	运算结果数据存储字软元件地址			
				WXOR 连续执行	DXOR 连续执行
				WXORP 脉冲执行	DXORP 脉冲执行

2. 操作数

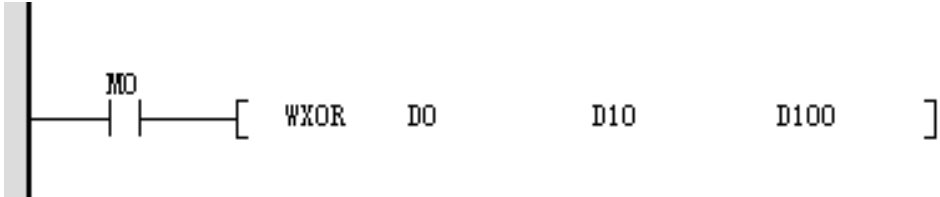
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

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）运算的规则为两个数相同时为 0，不同时为 1。

【使用举例】



当 M0 为 ON 时，将 D0 和 D10，以 bin 为单位进行异或运算结果存放在 D100 中。

	D0(16bin)	D10 (16bin)	D100 (16bin)
位（bin）单位的 逻辑异或运算	0	0	0
	0	1	1
	1	0	1
	1	1	0

驱动条件成立时，对 D 进行逐位取反、再加 1，并将结果写入 D。

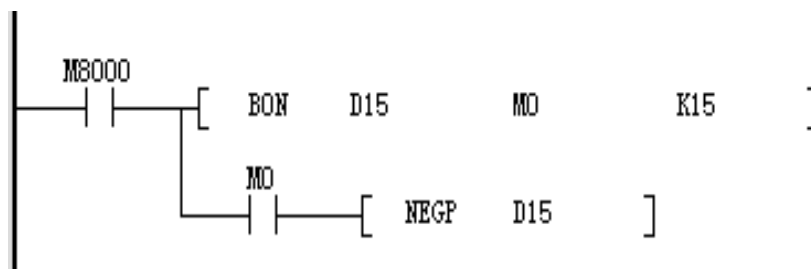
NEG D			二进制数据求补	指令执行	
D	运算结果	数据存储器地址		16 位指 (3step) NEG 连续执行 NEGP 脉冲执行	32 位指 (5step) DNEG 连续执行 DNEGP 脉冲执行

操作数	位软元件							字软元件														
	系统·用户							系统·用户				位数指定				变址		常数		实数		
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

功能说明:

- 需要触点驱动，有 1 个操作变量。将 D 的数值逐位取反、再加 1，存回 D 中。此指令一般用脉冲执行型指令。
- 使用 NEG 指令，可使负数绝对值化。

NEG 指令使负数绝对值化:



98

ENEG 二进制浮点数取反

1. 指令形式

将二进制浮点数（实数）的符号取反的指令。

ENEG S/D		二进制浮点数符号取反	指令执行
S/D	操作数	保存要执行符号翻转的二进制浮点数数据的软元件的起始编号	32 位指（5step） DENEG 连续执行 DENEGP 脉冲执行

2. 操作数

操作数	位软元件							字软元件													
	系统·用户							系统·用户					位数指定					变址		常数	
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z 修饰	K	H	E

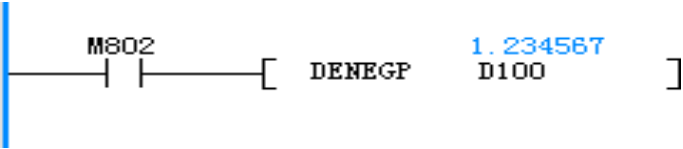
注：带灰色底纹软元件，表示可支持。

功能说明：

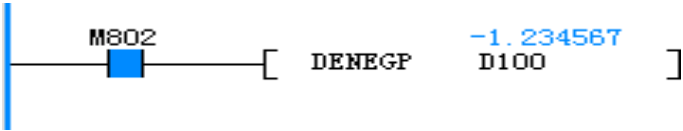
将[D+1, D]的 2 进制浮点数的符号取反，结果保存在[D+1, D]中。一般使用脉冲型指令。

【使用举例】

指令执行前



指令执行后



4.4.3 三角函数

三角函数	
SIN	浮点数 SIN 运算指令
COS	浮点数 COS 运算指令指令
TAN	浮点数 TAN 运算指令
ASIN	二进制浮点数 ARCSIN 运算
ACOS	二进制浮点数 ARCCOS 运算
ATAN	二进制浮点数 ARCTAN 运算
RAD	二进制浮点数角度→弧度的转换
DEG	二进制浮点数弧度→角度的转换
SINH	二进制浮点数 SINH 运算
COSH	二进制浮点数 COSH 运算
TANH	二进制浮点数 TANH 运算

SIN 浮点数 SIN 运算

1. 指令形式

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。

SINS D		浮点数 SIN 运算	指令执行
S	数据源	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 $0 \leq \alpha \leq 2\pi$	32 位指令（9step） DSIN 连续执行 DSINP 脉冲执行
D	运算结果	变换后 SIN 计算结果的存储单元，二进制浮点数格式。	

2. 操作数

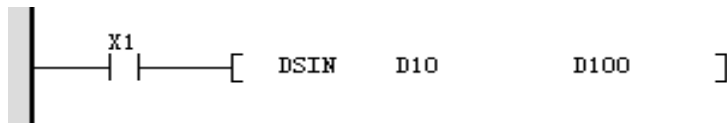
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 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。
- S 为待求正弦值的角度变量，RAD 单位，以二进制浮点数表示。
- D 为变换后 SIN 计算结果的存储单元，二进制浮点数格式。
- $\text{RAD(弧度)值} = \text{角度} \times \pi / 180^\circ$ 。

【使用举例】



当 X1 为 ON 时，将弧度(D11,D10)求 SIN 值，存放到（D101,D100）中。

这里计算的源数据、SIN 结果都为二进制浮点数格式。

COS 浮点数 COS 运算

1. 指令形式

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。

COS S D		浮点数 COS 运算	指令执行
S	数据源	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 $0 \leq \alpha \leq 2\pi$	32 位指令（9step） DCOS 连续执行 DCOSP 脉冲执行
D	运算结果	变换后 COS 计算结果的存储单元，二进制浮点数格式。	

2. 操作数

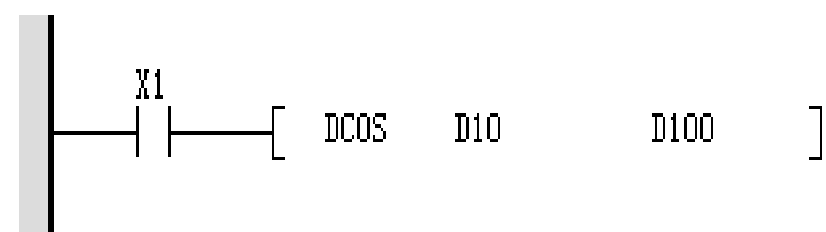
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 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。
- S 为待求余弦值的角度变量，RAD 单位，以二进制浮点数表示。
- D 为变换后 COS 计算结果的存储单元，二进制浮点数格式。
- $\text{RAD(弧度)值} = \text{角度} \times \pi / 180^\circ$ 。

【使用举例】



当 X1 为 ON 时，将弧度(D11,D10)求 COS 值，存放到（D101,D100）中。

这里计算的源数据、COS 结果都为二进制浮点数格式。

TAN 浮点数 TAN 运算

1. 指令形式

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。

TAN S D			浮点数 TAN 运算	指令执行
S	数据源	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 $0 \leq \alpha \leq 2\pi$		32 位指令（9step） DTAN 连续执行 DTANP 脉冲执行
D	运算结果	变换后 TAN 计算结果的存储单元，二进制浮点数格式。		

2. 操作数

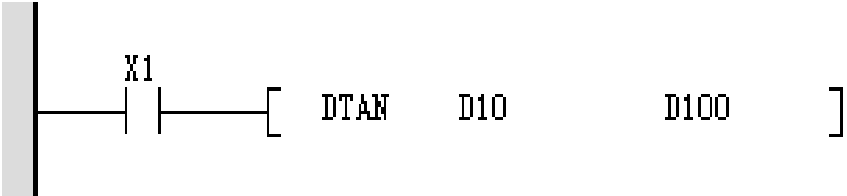
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 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。
- S 为待求正切值的角度变量，RAD 单位，以二进制浮点数表示。
- D 为变换后 TAN 计算结果的存储单元，二进制浮点数格式。
- $\text{RAD(弧度)值} = \text{角度} \times \pi / 180^\circ$ 。

【使用举例】



当 X1 为 ON 时，将弧度(D11,D10)求 TAN 值，存放到（D101,D100）中。

这里计算的源数据、TAN 结果都为二进制浮点数格式。

ASIN 二进制浮点数 ARCSIN 运算

1. 指令形式

从 SIN 值求出对应弧度的运算。

ASIN S D			二进制浮点数 ARCSIN 运算	指令执行
S	数据源	存放待求 ARCSIN（反正弦）的二进制浮点数变量		32 位指令（9step） DASIN 连续执行 DASINP 脉冲执行
D	运算结果	为计算结果(- π/2～ π/2)的存储单元		

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

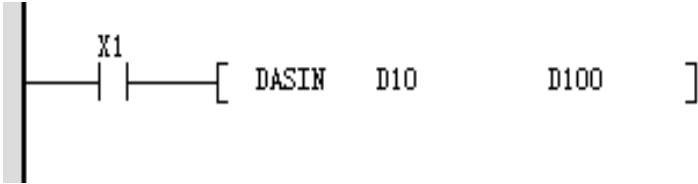
注：带灰色底纹软元件，表示可支持。

功能说明：

该指令是从 SIN 值求出对应弧度的运算。

注意：S 中的值不在-1.0~1.0 的范围内时将发生运算出错。错误代码为 K6706，K6706 会保存在 D8067 中，出错标志位 M8067 置 ON。

【使用举例】



当 X1 为 ON 时，将弧度(D11,D10)进行 SIN^{-1} 预算后，存放到（D101,D100）中。

ACOS 二进制浮点数 ARCCOS 运算

1. 指令形式

从 COS 值求出对应弧度的运算。

ACOS S D			二进制浮点数 ARCCOS 运算	指令执行
S	数据源	存放待求 ARCCOS（反余弦）的二进制浮点数变量		32 位指令（9step） DACOS 连续执行 DACOSP 脉冲执行
D	运算结果	计算结果(0~π)的存储单元		

2. 操作数

操作数	位软元件								字软元件													
	系统·用户								系统·用户				位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

该指令是从 COS 值求出对应弧度的运算。

注意：S 中的值不在-1.0~1.0 的范围内时将发生运算出错。错误代码为 K6706，K6706 会保存在 D8067 中，出错标志位 M8067 置 ON。

【使用举例】



当 X1 为 ON 时，将弧度(D11,D10)进行 $\cos^{-1}$ 预算后，存放到（D101,D100）中。

ATAN 二进制浮点数 ARCTAN 运算

1. 指令形式

从 TAN 值求出对应弧度的运算。

ATAN S D			二进制浮点数 ARCTAN 运算	指令执行
S	数据源	存放待求 ARCTAN（反正切）的二进制浮点数变量		32 位指令（9step） DATAN 连续执行 DATANP 脉冲执行
D	运算结果	计算结果(- π /2~ π /2)的存储单元		

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

该指令是从 TAN 值求出对应弧度的运算。

【使用举例】



当 X1 为 ON 时，将弧度(D11,D10)进行 TAN^{-1} 预算后，存放到（D101,D100）中。

RAD 二进制浮点数角度→弧度的转换

1. 指令形式

进行二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 $\times \pi /180$ ）

RAD S D			二进制浮点数角度→弧度的转换	指令执行
S	数据源	存放待求弧度的二进制浮点数角度变量		32 位指令（9step） DRAD 连续执行 DRADP 脉冲执行
D	运算结果	计算结果的存储单元		

2. 操作数

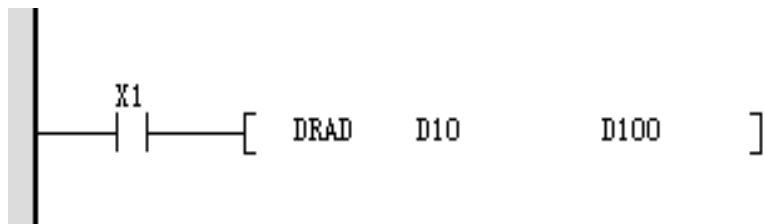
操作数	位软元件							字软元件													
	系统·用户							系统·用户					位数指定					变址		常数	
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z 修饰	K	H	E
D1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z 修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

该指令是进行二进制浮点数角度转换成弧度的运算。

【使用举例】



当 X1 为 ON 时，将(D11、D10)中的二进制浮点数数值进行角度到弧度的运算后保存到（D101,D100）中。

DEG 二进制浮点数弧度→角度的转换

1. 指令形式

进行二进制浮点数弧度转换成角度的运算。

其计算公式为（角度单位=弧度单位× π /180）

DEG S D			二进制浮点数弧度→角度的转换	指令执行
S	数据源	存放待求角度的二进制浮点数弧度变量		32 位指令（9step） DDEG 连续执行 DDEGP 脉冲执行
D	运算结果	计算结果的存储单元		

2. 操作数

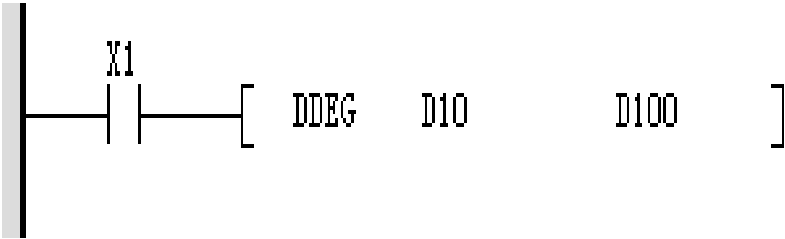
操作数	位软元件							字软元件													
	系统·用户							系统·用户				位数指定				变址		常数		实数	
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z 修饰	K	H	E
D1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z 修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

该指令是进行二进制浮点数弧度转换成角度的运算。

【使用举例】



当 X1 为 ON 时，将(D11、D10)中的二进制浮点数数值进行弧度到角度的运算后保存到（D101,D100）中。

SINH 浮点数 SINH 运算

1. 指令形式
- 进行二进制浮点数取 SINH 值。
- 其计算公式为 $\sinh(s)=(e^s-e^{-s})/2$ 。

SINH S D			浮点数 SINH 运算	指令执行
S	数据源	存放待求 SINH 值的二进制浮点数变量		32 位指令（9step） DSINH 连续执行 DSINHP 脉冲执行
D	运算结果	计算结果的存储单元（运算结果超出浮点数范围时会报错 6706）		

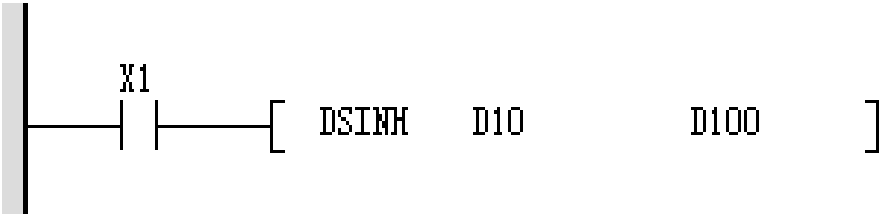
2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：该指令是进行二进制浮点数取 SINH 值。

【使用举例】



当 X1 为 ON 时，将(D11、D10)中的二进制浮点数数值求取 SINH 值后保存到（D101,D100）中。

COSH 浮点数 COSH 运算

1. 指令形式
- 进行二进制浮点数取 COSH 值。
- 其计算公式为 $\cosh(s)=(e^s+e^{-s})/2$ 。

COSH S D			浮点数 COSH 运算	指令执行
S	数据源	存放待求 COSH 值的二进制浮点数变量		32 位指令（9step） DCOSH 连续执行 DCOSHP 脉冲执行
D	运算结果	计算结果的存储单元		

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：该指令是进行二进制浮点数取 COSH 值。

【使用举例】



当 X1 为 ON 时，将(D11、D10)中的二进制浮点数数值求取 COSH 值后保存到（D101,D100）中。

TANH 浮点数 TANH 运算

1. 指令形式

进行二进制浮点数取 TANH 值。

其计算公式为 $\tanh(s)=(e^s-e^{-s})/(e^s+e^{-s})$ 。

TANH S D			浮点数 TANH 运算	指令执行
S	数据源	存放待求 TANH 值的二进制浮点数变量		32 位指令（9step） DTANH 连续执行 DTANHP 脉冲执行
D	运算结果	计算结果的存储单元		

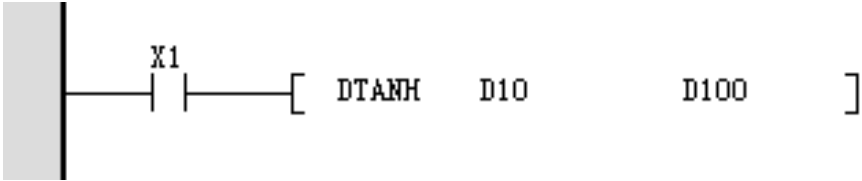
2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：该指令是进行二进制浮点数取 TANH 值。

【使用举例】



当 X1 为 ON 时，将(D11、D10)中的二进制浮点数数值求取 TANH 值后保存到（D101,D100）中。

4.4.4 表格运算

三角函数	
WSUM	算出数据合计值
MEAN	平均值计算
LIMIT	上下限限位控制
BZAND	死区控制
ZONE	区域控制

WSUM 算出数据合计值

1. 指令形式

该指令可计算出连续的 16 位或是 32 位数据的合计值。

WSUM S D n			算出数据合计值	指令执行	
S	数据源	保存要算出合计值的数据的软元件起始编号		16 位指令（7step） WSUM 连续执行 WSUMP 脉冲执行	32 位指令（13step） DWSUM 连续执行 DWSUMP 脉冲执行
D	结果	保存合计值的软元件起始编号。			
n	数据个数	数据个数(n>0)			

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

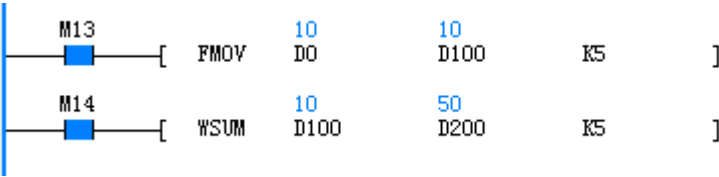
功能说明：

- 16 位指令：将[S]开始的 n 点 16 位数据的求和，结果以 32 位数据形式保存在[D+1, D]中。
- 32 位指令：将[S+1, S]开始的 n 点 32 位数据的求和，结果以 64 位数据形式保存在[D+3, D+2, D+1, D]中。

以下一些情况会报故障。出错标志位 M8067 置 ON，同时出错代码保存于 D8067。

- 如果[S]开始的 n 点软元件超出了其合法范围，报 6705 号错误。
- 在存储数据时，如果[D]超出了其合法范围，报 6705 号错误。
- 操作数 n 的有效范围：n≤0 时，报 6706 号错误。

【使用举例】



将 D100~D104 均赋值 10，将 M14 置 ON，对 D100 地址开始的 5 个 D 元件进行求和操作，结果存储于 (D201, D200)。

MEAN 平均值计算（开发中）

1. 指令形式

当驱动条件成立时，求以 S 为起始的 K 个数据的平均值，结果存放到 D。

MEAN S D n			平均值计算指令	指令执行	
S	数据首址	求平均值的数据存储字软元件首址		16位指令（7step） MEAN连续执行 MEANP脉冲执行	32位指令（13step） DMEAN连续执行 DMEANP脉冲执行
D	平均值	平均值数据存储字软元件地址			
n	数据长度	立即数，K=1～64			

2. 操作数

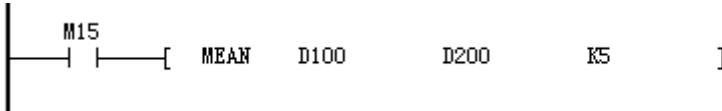
操作数	位软元件							字软元件												
	系统·用户							系统·用户					位数指定					变址	常数	实数
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K H E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K H E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K H E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 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。

【使用举例】



例如：D100=9，D101=10，D102=11，D103=12，D104=15；则 D200=11，余数被舍去。

$$(D100+D101+D102+D103+D104) / 4 = D200$$

LIMIT 上下限限位控制

1. 指令形式

设置输入数值的上限值/下限值然后输出的指令。

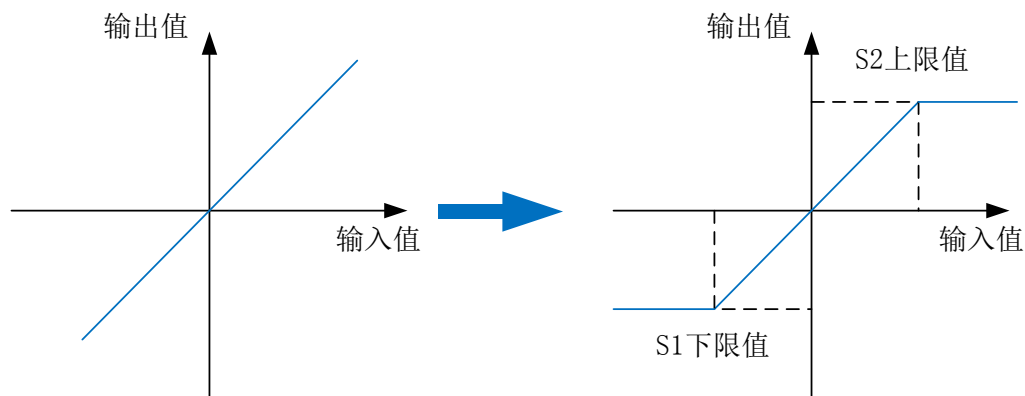
LIMIT S1 S2 S3 D					上下限限位控制	指令执行	
S1	下限值	下限限位值(最小输出界限值)			16位指令（9step） LIMIT连续执行 LIMITP脉冲执行	32位指令（17step） DLIMIT连续执行 DLIMITP脉冲执行	
S2	上限值	上限限位值(最大输出界限值)					
S3	输入值	需要通过上下限限位控制的输入值					
D	输出值	保存已经过上下限限位控制的输出值的 软元件起始编号					

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S3	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：



■ 16 位指令

- 通过在[S1]、[S2]中设定上下限值，使输出值[D]在一定范围内依据输入值[S3]进行输出。
- 当[S1]>[S3]时，[S1]→[D]。
- 当[S2]<[S3]时，[S2]→[D]。
- 当[S1]≤[S3]≤[S2]时，[S3]→[D]。
- 如果仅控制上限的话，请在下限[S1]中设定 16 位有符号最小值，即-32768。
- 如果仅控制下限的话，请在上限[S2]中设定 16 位有符号最大值，即 32767。

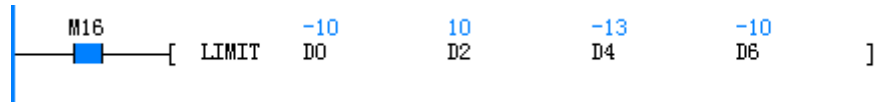
■ 32 位指令

- 通过在[S1+1, S1]、[S2+1, S2]中设定上下限值，使输出值[D+1, D]在一定范围内依据输入值[S3+1, S3]进行输出。

S3]。

- 当 $[S1+1, S1] > [S3+1, S3]$ 时, $[S1+1, S1] \rightarrow [D+1, D]$ 。
- 当 $[S2+1, S2] < [S3+1, S3]$ 时, $[S2+1, S2] \rightarrow [D+1, D]$ 。
- 当 $[S1+1, S1] \leq [S3+1, S3] \leq [S2+1, S2]$ 时, $[S3+1, S3] \rightarrow [D+1, D]$ 。
- 如果仅控制上限的话, 请在下限 $[S1+1, S1]$ 中设定 32 位有符号最小值, 即-2,147,483,648。如果仅控制下限的话, 请在上限 $[S2+1, S2]$ 中设定 32 位有符号最大值, 即 2,147,483,647。
- 以下一些情况会报故障。出错标志位 M8067 置 ON, 同时出错代码保存于 D8067。
 - 16 位指令和 32 位指令, 当下限值>上限值时, 报 6706 号错误。

【使用举例】



BZAND 死区控制（开发中）

1. 指令形式

通过判断输入值是否在指定的死区的上下限范围内，从而来控制输出值的指令。

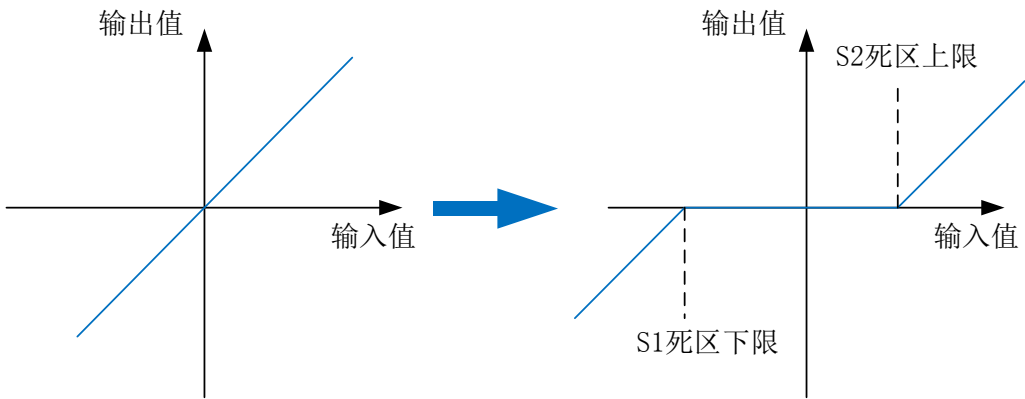
BZAND S1 S2 S3 D					死区控制	指令执行	
S1	下限值	死区（无输出区域）的下限值			16位指令（9step） BZAND连续执行 BZANDP脉冲执行	32位指令（17step） DBZAND连续执行 DBZANDP脉冲执行	
S2	上限值	死区（无输出区域）的上限值					
S3	输入值	要通过死区控制的输入值					
D	输出值	保存经过死区控制的输出值的软元件编号					

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S3	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：



■ 16 位指令

- 通过在[S1]、[S2]中设定死区范围，使输入值[S3]在死区范围外输出至[D]软元件中。输出值如下所示被控制。
- 当[S1] > [S3]时，[S3] - [S1] → [D]。
- 当[S2] < [S3]时，[S3] - [S2] → [D]。
- 当[S1] ≤ [S3] ≤ [S2]时，0 → [D]。

■ 32 位指令

- 通过在[S1+1, S1]、[S2+1, S2]中设定死区范围，使输入值[S3+1, S3]在死区范围外输出至

[D+1, D]软元件中。当[S1+1, S1]>[S3+1, S3]时, [S3+1, S3]-[S1+1, S1]→[D+1, D];

- 当[S2+1, S2]<[S3+1, S3]时, [S3+1, S3]-[S2+1, S2]→[D+1, D]; 当[S1+1, S1]≤[S3+1, S3]≤[S2+1, S2]时, 0→[D+1, D]。
- 指令在使用过程中, 数据的溢出情况符合循环处理, 即最大值加 1 则变最小值, 最小值减 1 则变最大值。
- 以下一些情况会报故障。出错标志位 M8067 置 ON, 同时出错代码保存于 D8067。
 - 16 位指令和 32 位指令, 当下限值>上限值时, 报 6706 号错误。

【使用举例】



ZONE 区域控制（开发中）

1. 指令形式

根据输入值是正数还是负数，用指定的偏差值来控制输出值的指令。

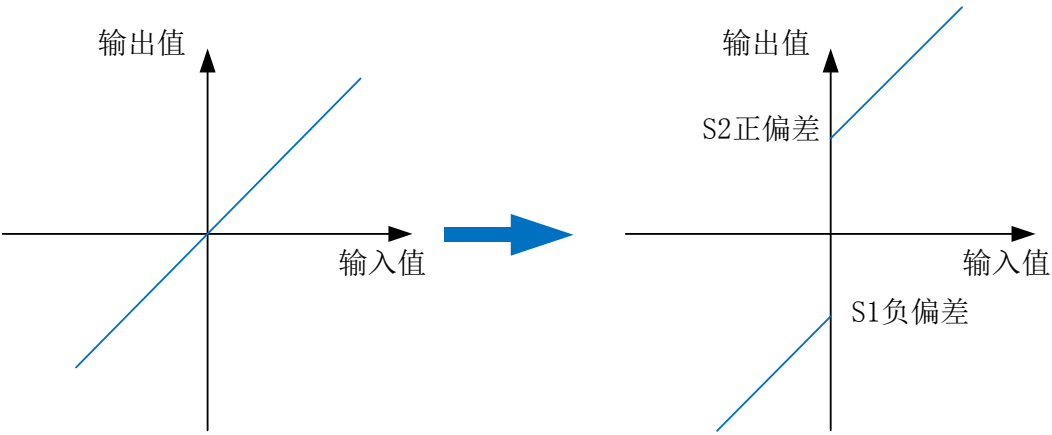
ZONE S1 S2 S3 D					区域控制	指令执行	
S1	负偏差	加在输入值上的负偏差值(可以正数、负数、0)			16位指令（9step） ZONE连续执行 ZONEP脉冲执行	32位指令（17step） DZONE连续执行 DZONEP脉冲执行	
S2	正偏差	加在输入值上的正偏差值(可以正数、负数、0)					
S3	输入值	要通过区域控制的输入值					
D	输出值	保存已通过区域控制的输出值的软元件起始编号					

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S3	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：



- 16 位指令
 - 依据输入值[S3]的符号确定，加上[S2]或[S1]，执行结果保存至[D]元件中。当[S3]<0 时，[S3]+[S1]→[D]。
 - 当[S3]>0 时，[S3]+[S2]→[D]。
 - 当[S3]=0 时，0→[D]。
- 32 位指令
 - 依据输入值[S3+1, S3]的符号确定，加上[S2+1, S2]或[S1+1, S1]，执行结果保存至[D+1, D]元件中。

【使用举例】



4.4.5 指数运算

指数运算	
EXP	二进制浮点数指数运算
LOGE	二进制浮点数自然对数运算
LOG	二进制浮点数以 10 为底的对数运算
ESQR	二进制浮点开方运算
SQR	二进制数据开方运算

EXP 二进制浮点数指数运算

1. 指令形式

进行以 e(2.71828)为底的二进制浮点数数据的指数运算。

EXP S D			二进制浮点数指数运算	指令执行
S	数据源	待求二进制浮点数指数的二进制浮点数变量		32 位指令（9step） DEXP 连续执行 DEXPP 脉冲执行
D	运算结果	指数运算后计算结果的存储单元		

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 该指令是进行以 e(2.71828)为底的二进制浮点数数据的指数运算。
- S 为待求二进制浮点数指数的二进制浮点数变量。
- D 为指数运算后计算结果的存储单元。

注意：当运算结果不在 $[2^{-126} \leq \text{运算结果} < 2^{128}]$ 时将发生运算出错。错误代码为 K6706，K6706 保存在 D8067 中，出错标志位 M8067 置 ON。

【使用举例】



当 X1 为 ON 时，以 e 为底对(D11、D10)中的二进制浮点数数值进行指数运算后保存到（D101,D100）中。

LOGE 二进制浮点自然对数运算

1. 指令形式

进行以 e(2.71828)为底的二进制浮点数数据的自然对数运算。

LOGE S D			二进制浮点自然对数运算	指令执行
S	数据源	待求二进制浮点数自然对数的二进制浮点数变量		32 位指令（9step） DLOGE 连续执行 DLOGEP 脉冲执行
D	运算结果	自然对数运算后计算结果的存储单元		

2. 操作数

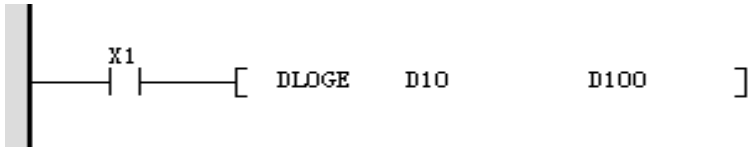
操作数	位软元件							字软元件													
	系统·用户							系统·用户					位数指定					变址		常数	
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z 修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z 修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 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。
- 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。

【使用举例】



当 X1 为 ON 时，以 e 为底对(D11、D10)中的二进制浮点数数值进行自然对数运算后保存到(D101,D100)中。

LOG 二进制浮点以 10 为底对数运算

1. 指令形式

进行以 10 为底的二进制浮点数数据的常用对数运算。

LOG S D			二进制浮点数 10 为底的对数运算	指令执行
S	数据源	待求二进制浮点数常用对数的二进制浮点数变量		32 位指令（9step） DLOG 连续执行 DLOGP 脉冲执行
D	运算结果	常用对数运算后计算结果的存储单元		

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

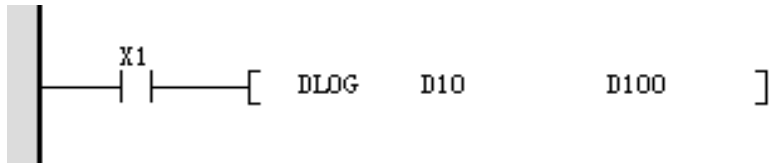
注：带灰色底纹软元件，表示可支持。

功能说明：

该指令是进行以 10 为底的二进制浮点数数据的常用对数运算。

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。

【使用举例】



当 X1 为 ON 时，以 e 为底对(D11、D10)中的二进制浮点数数值进行自然对数运算后保存到(D101,D100)中。

ESQR 二进制浮点数开方运算

1. 指令形式

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。

ESQR S D			二进制浮点数开方运算	指令执行
S	数据源	待求平方根的二进制浮点数变量		32 位指令（9step） DESQR 连续执行 DESQRP 脉冲执行
D	运算结果	二进制浮点平方根的存储单元		

2. 操作数

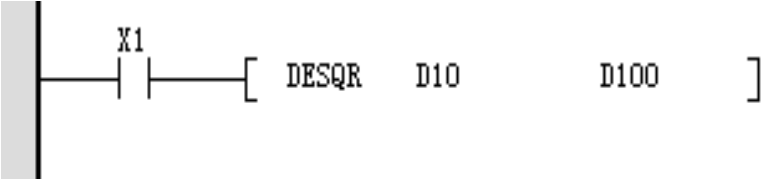
操作数	位软元件							字软元件													
	系统·用户							系统·用户				位数指定				变址		常数		实数	
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z 修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z 修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 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。
- 操作数 S 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。

【使用举例】



当 X1 为 ON 时，将二进制浮点数开方结果 $\sqrt{(D11, D10)}$ 存放到（D101,D100）中。

SQR 二进制数据开方运算

1. 指令形式

进行二进制数据开平方运算。

SQR S D			二进制数据开方运算	指令执行	
S	数据源	将开方的数据或数据存储字软元件地址		16 位指令（5step） SQR 连续执行 SQRP 脉冲执行	32 位指令（9step） DSQR 连续执行 DSQRP 脉冲执行
D	运算结果	运算结果的存储单元			

2 操作数

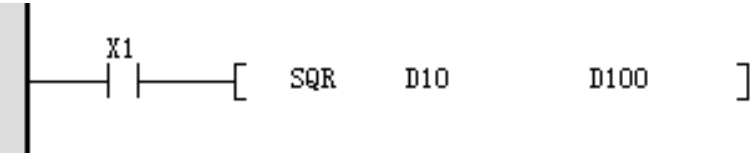
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 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。

【使用举例】



当 X1 为 ON 时，将二进制浮点数开方结果 $\sqrt{D10}$ 存放到 D100 中。

如果 D10 为 9，计算结果 D100 为 3。

如果 D10 为 10，计算记过 D100 为 3，小数部分舍去。

4.5 数据处理

数据转换	
INT	二进制浮点→BIN 整数变换
FLT	二进制数据→二进制浮点数转换
BCD	二进制数据→BCD 数据
BIN	BCD 数据→二进制数据
WTOB	字节单位的数据分离
BTOW	字节单位的数据结合
UNI	16 位数据的 4 位结合
DIS	16 位数据的 4 位分离
ASCI	HEX→ASCII 转换
HEX	ASCII→HEX 转换
数据传送	
MOV	赋值传送
EMOV	二进制浮点传送
SMOV	移位传送
BMOV	数据成批传送
FMOV	数据一对多传送
CML	数据取反传送
表格操作	
ZRST	全部数据复位
数据移位	
ROR	循环右移
ROL	循环左移
RCR	带进位的循环右移
RCL	带进位的循环左移
其他数据处理	
SWAP	上下字节交换
BON	ON 位判断
SUM	ON 为总数
RND	产生随机数据
XCH	数据交换

4.5.1 数据转换

数据转换	
INT	二进制浮点→BIN 整数变换
FLT	二进制数据→二进制浮点数转换
BCD	二进制数据→BCD 数据
BIN	BCD 数据→二进制数据
WTOB	字节单位的数据分离
BTOW	字节单位的数据结合
UNI	16 位数据的 4 位结合
DIS	16 位数据的 4 位分离
ASCI	HEX→ASCII 转换
HEX	ASCII→HEX 转换

INT 二进制浮点→BIN 整数变换

1. 指令形式

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。

INT S D			二进制浮点→BIN 整数变换	指令执行	
S	数据源	待取整变换的二进制浮点数变量		16 位指令 (5step)	32 位指令 (9step)
D	运算结果	变换后 BIN 整数结果的存储单元		INT 连续执行 INTP 脉冲执行	DINT 连续执行 DINTP 脉冲执行

2. 操作数

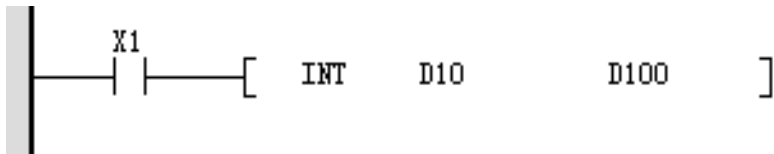
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 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。
- S=0 时，M8020 会置位。
- $|S| \leq 1$ 时，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：-32768~32767。
- 32 位指令：-2147483648~2147483647。

【使用举例】



当 X1 为 ON 时，将浮点数 (D11,D10) 取整后存放在 D100 中。

FLT 二进制整数至浮点数的转换

1. 指令形式

把二进制整数转化为二进制浮点数。

FLT S D			二进制整数至浮点数的转换	指令执行	
S	整数	要转化的二进制数或数据存储字软元件地址		16 位指令 (5step)	32 位指令 (9step)
D	浮点数	浮点数数据存储字软元件地址		FLT 连续执行 FLTP 脉冲执行	DFLT 连续执行 DFLTP 脉冲执行

2. 操作数

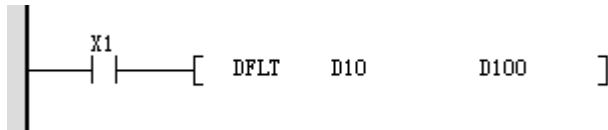
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 将整数 S 转换为浮点数，结果存入 D 和 D+1 单元。
- 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 进浮点数值变换成 BIN 整数）。

【使用举例】



当 X1 为 ON 时，将 32bit 数(D11,D10)中(32 位 BIN 整数)转换为二进制浮点数后，存放到(D101,D100)。

BCD 二进制数据转换 BCD 数据（开发中）

1. 指令形式

二进制数据转换为 BCD 数据。

BCD S D			二进制数据转换 BCD 数据	指令执行	
S	数据源	二进制码数据或数据存储字软元件地址		16 位指令（5step） BCD 连续执行 BCDP 脉冲执行	32 位指令（9step） DBCD 连续执行 DBCDP 脉冲执行
D	运算结果	BCD 码数据存储字软元件地址			

2. 操作数

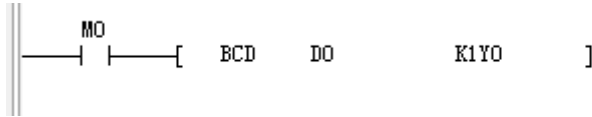
操作数	位软元件							字软元件												
	系统·用户							系统·用户				位数指定				变址		常数		实数
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K H E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K H E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 需要出点驱动，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。
- 使用 16bit 指令时，范围是 0-9999，档转换结果超过 9999 时会出错；使用 32bit 指令时，范围是 0-99999999，当转换结果超过 99999999 时出错。M8067、M8068 会置位，D8067 记录错误代码。

【使用举例】



将 D0 的 BIN 值转换成 BCD 值后，将结果的个位数存放 K1Y0 中（Y0~Y3 四个 bit 元件）。

若 D0=H00F4（十六进制）=K244（十进制），则变换后结果为 Y0~Y3=0100（BIN）。

若 D0=H0046（十六进制）=K70（十进制），则变换后结果为 Y0~Y3=0000（BIN）。

BIN BCD 数据→二进制数据

1. 指令形式

BCD 数据转换二进制数据。

BIN S D			二进制浮点→十进制浮点变换	指令执行	
S	数据源	BCD 码数据或数据存储字软元件地址		16 位指令 (5step) BIN 连续执行 BINP 脉冲执行	32 位指令 (9step) DBIN 连续执行 DBINP 脉冲执行
D	运算结果	二进制码数据存储字软元件地址			

2. 操作数

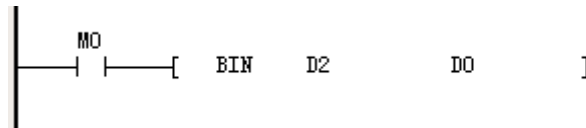
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 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。
- S (BCD) 的有效范围，16bit: 0~9999; 32bit: 0~99999999
- S 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、M8068 会置位, D8067 记录错误代码。

【使用举例】



当 M0 为 ON 时，将 D2 的 BCD 值作 BIN 转换存入 D0。

WTOB 字节单位的数据分离

1. 指令形式

将连续的 16 位数据按照字节(8 位)单位进行分离的指令。

WTOB S D n			字节单位的数据分离	指令执行
S	源数据	保存要按照字节单位进行分离的数据的软元件起始编号		16 位指令（7step） WTOB 连续执行 WTOBP 脉冲执行
D	结果	保存已经按照字节单位分离的结果的软元件起始编号		
n	分离数	要分离的字节数据个数(n≥0，且n=0时不处理)		

2. 操作数

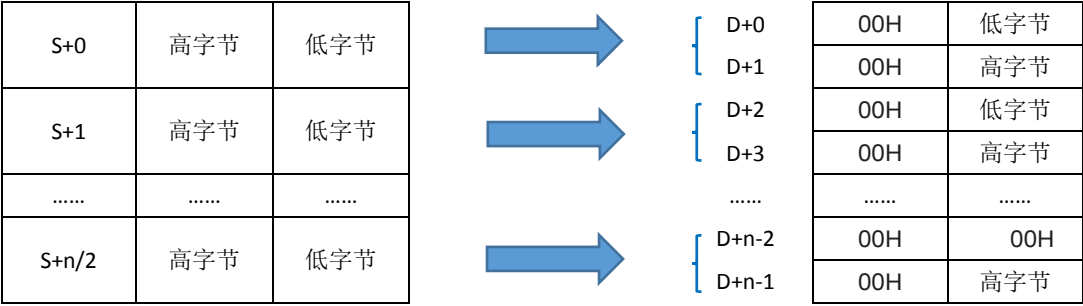
操作数	位软元件							字软元件														
	系统·用户							系统·用户				位数指定					变址		常数		实数	
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

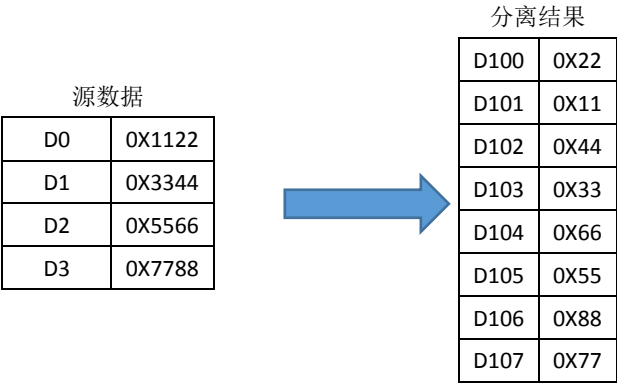
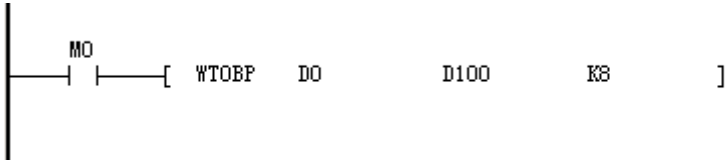
功能说明：

将[S]开始的软元件中保存的 16 位数据以字节为单位保存到的[D]开始的n 点软元件的低 8 位中,高 8 位保存 00H。

以下一些情况会报故障。出错标志位 M8067 置ON, 同时出错代码保存于 D8067。当[S]、[D]开始的软元件超出其软元件范围时, 报 6705 号错误。



【使用举例】



BTOW 字节单位的数据结合

1. 指令形式

将连续的 16 位数据的低 8 位（低字节）结合在一起的指令。

BTOW S D n			字节单位的数据结合	指令执行
S	源数据	保存要按照字节单位进行结合的数据的软元件起始编号		16 位指令（7step） BTOW 连续执行 BTOWP 脉冲执行
D	结果	保存已经按照字节单位结合的结果的软元件起始编号		
n	分离数	要结合的字节数据个数(n≥0, 且n=0时不处理)		

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

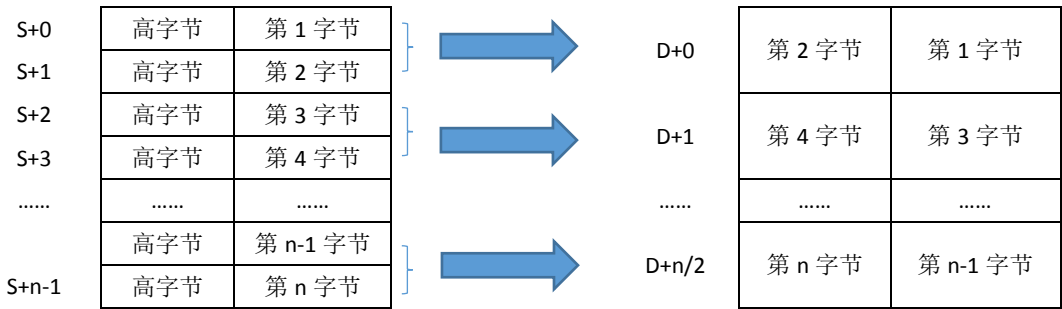
注：带灰色底纹软元件，表示可支持。

功能说明：

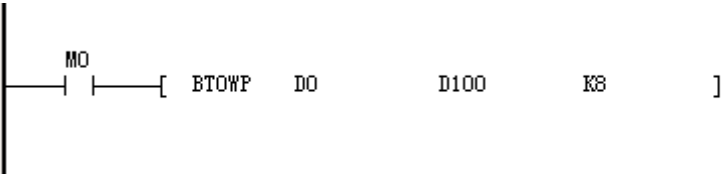
将[S]开始的 n 个 16 位数据的低 8 位结合在一起后的 16 位数据保存至[D]开始的软元件中，[S]的高 8 位被忽略。

以下一些情况会报故障。出错标志位 M8067 置 ON，同时出错代码保存于 D8067。

当[S]、[D]开始的软元件超出其软元件范围时，报 6706 号错误。



【使用举例】



源数据

D0	0X1122
D1	0X3344
D2	0X5566
D3	0X7788
D4	0X99AA
D5	0XBBCC
D6	0XDDEE
D7	0XFF00



分离结果

D100	0X4422
D101	0X8866
D102	0XCCAA
D103	0X00EE

UNI 16 位数据的 4 位结合（开发中）

1. 指令形式

将连续的 16 位数据的低 4 位结合在一起的指令。

UNI S D n			16 位数据的 4 位结合	指令执行
S	源数据	保存要结合的数据的软元件起始编号		16 位指令（7step） UNI 连续执行 UNIP 脉冲执行
D	结果	保存已结合的数据的软元件编号		
n	结合数	结合数(0~4, n=0时不处理)		

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户				位数指定					变址		常数		实数	
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

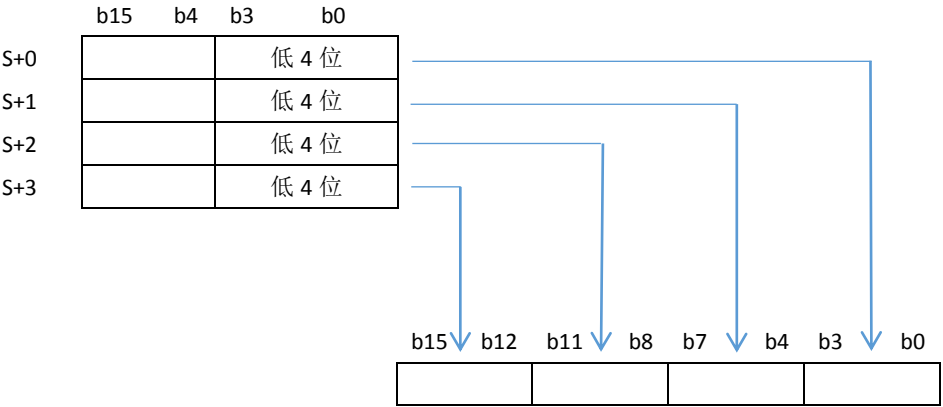
注：带灰色底纹软元件，表示可支持。

功能说明：

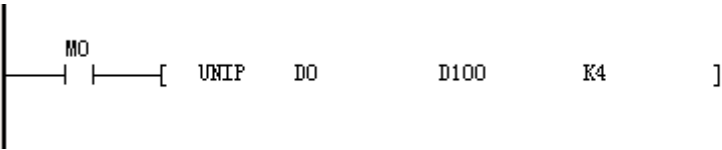
- 将 S 起始的 n 个 16 位数据的低 4 位结合成 16 位数据保存到 D 中。
- n 的取值范围 1~4。n=0 时，指令不执行。n 取值 1~3 时，剩余高位 0 填充。

以下一些情况下会发生运算错误，错误标志位 M8067 置 ON，错误代码保存在 D8067 中。

- S 中设定的软元件超范围时，报错误代码 K6705；
- n 设定超范围时，报错误代码 K6706。



【使用举例】



源数据

D0	0X1122
D1	0X3344
D2	0X5566
D3	0X7788



结合结果

D100	0X8642
------	--------

DIS 16 位数据的 4 位分离（开发中）

1. 指令形式

将 16 位数据以 4 位为单位分离的指令。

DIS S D n			16 位数据的 4 位分离	指令执行
S	源数据	保存要分离的数据的软元件起始编号		16 位指令（7step） DIS 连续执行 DISP 脉冲执行
D	结果	保存已分离的数据的软元件编号		
n	分离数	分离数(0~4, n=0时不处理)		

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户				位数指定					变址		常数		实数	
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

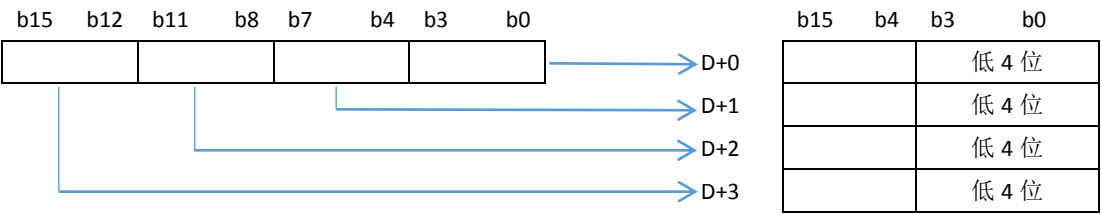
注：带灰色底纹软元件，表示可支持。 低

功能说明：

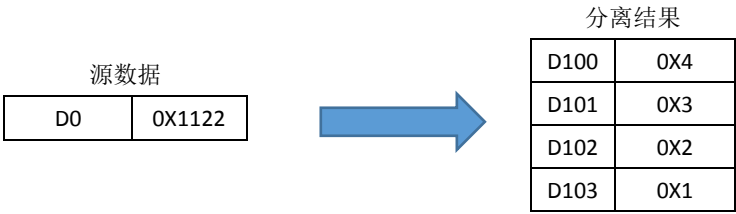
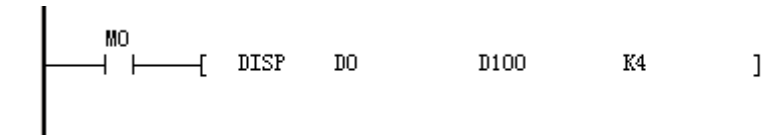
- 将 S 的 16 位数据以 4 位为单位进行分离后存储于 D 开始的软元件低 4 位中，高 12 位以 0 填充。
- n 的取值范围为 1~4。n=0 时，不执行指令的处理。

以下一些情况下会发生运算错误，错误标志位 M8067 置 ON，错误代码保存在 D8067 中。

- D 中设定的软元件超范围时，报错误代码 K6705。
- n 设定超范围时，报错误代码 K6706。



【使用举例】



ASCII HEX→ASCII 转换（开发中）

1. 指令形式

将 S 的值转换成 ASCII 码后，存储到 D 为起始地址的变量中。

ASCI S D n			HEX→ASCII 转换	指令执行
S	数据源	待转换的变量地址或常数数值		16 位指令（7step） ASCI 连续执行 ASCIP 脉冲执行
D	运算结果	转换后 ASCII 码的存放起始地址		
n	转换字符数	转换的字符位数，n 的范围为 1~256		

2. 操作数

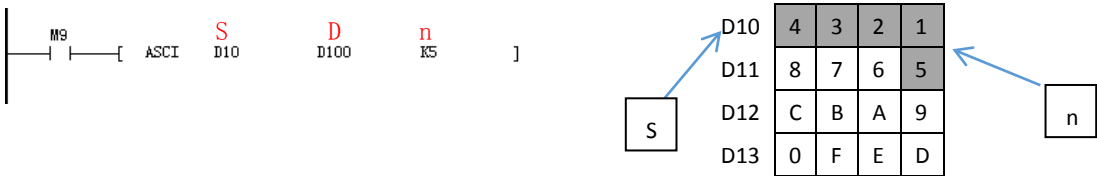
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 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。
- 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。

【使用举例】



M8161=OFF16bit 模式

1								1
2							1	2
3						1	2	3
4			不变化		1	2	3	4
5				1	2	3	4	5
6			1	2	3	4	5	6
7		1	2	3	4	5	6	7
n	H	L	H	L	H	L	H	L
	D103	D102	D101	D100				

M8161=ON8bit 模式

1								1
2							1	2
3						1	2	3
4				不变化		1	2	3
5					1	2	3	4
6				1	2	3	4	5
7			1	2	3	4	5	6
n	H	L	H	L	H	L	H	L
	D107	D106	D105	D104	D103	D102	D101	D100

- 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。

b15								D10=1234H								b0							
0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0								
1				2				3				4											

b15								D11=8765H								b0							
0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0								
1				2				3				4											

当 M8161=OFF、n=4 时，位的组成
(D10~D11) 转换

b15								D100								b0							
0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0	0	0	0	1	0	0	0	0
[2]=32H								[1]=31H															

b15								D101								b0							
0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0	0	0	0	1	0	0	0	0
[4]=34H								[3]=33H															

b15								D102								b0							
0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0	0	0	0	1	0	0	0	0

当 M8161=OFF、n=5 时，位的组成
(D10~D11) 转换

b15								D100								b0							
0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0	0	0	0	1	0	0	0	0
[1]=31H								[5]=35H															

b15								D101								b0							
0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0	0	0	0	1	0	0	0	0
[3]=33H								[2]=32H															

b15								D102								b0							
0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0	0	0	0	1	0	0	0	0
																[4]=34H							

当 M8161=ON、n=4 时，位的组成
(D10~D11) 转换

b15								D100								b0							
0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0	0	0	0	1	0	0	0	0
								[1]=31H															

b15								D101								b0							
0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0	0	0	0	1	0	0	0	0
								[2]=32H															

b15								D102								b0							
0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0	0	0	0	1	0	0	0	0
								[3]=33H															

b15								D103								b0							
0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0	0	0	0	1	0	0	0	0
								[4]=34H															

b15								D104								b0							
0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0	0	0	0	1	0	0	0	0

当 M8161=ON、n=5 时，位的组成
(D10~D11) 转换

b15								D100								b0							
0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0	0	0	0	1	0	0	0	0
								[5]=35H															

b15								D101								b0							
0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0	0	0	0	1	0	0	0	0
								[1]=31H															

b15								D102								b0							
0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0	0	0	0	1	0	0	0	0
								[2]=32H															

b15								D103								b0							
0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0	0	0	0	1	0	0	0	0
								[3]=33H															

b15								D104								b0							
0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0	0	0	0	1	0	0	0	0
																[4]=34H							

注：RS/HEX/ASCII/CCD 等指令共用 M8161 模式标志，编程时注意。

HEX ASCII→HEX 转换（开发中）

1. 指令形式

将 S 起始变量的值转换成 HEX 码后，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。

HEX S D n			ASCII→HEX 转换	指令执行
S	数据源	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		16 位指令（7step） HEX 连续执行 HEXP 脉冲执行
D	运算结果	转换后 HEX 码的存放起始地址，占用的变量空间与 S2 有关		
n	转换字符数	转换的字符位数		

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 该指令是将 S 起始变量的值转换成 HEX 码后，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。
- S 为待转换的变量地址或常数数值，若为寄存器变量，以 32bi 变量宽度（即 4 个 ASCII 字符）为单位进行转换分隔。
- D 为转换后 HEX 码的存放起始地址，占用的变量空间与 n 有关。
- n 为转换的字符位数（范围：1~256）。

【使用举例】



例如 D100 起始的数据如下：

M8161=OFF16bit 模式

S	H	L	1		0H
D10	31H (1)	30H (0)	2		01H
D11	33H (3)	32H (2)	3		012H
D12	35H (5)	34H (4)	4	不变化	0123H
D13	37H (7)	36H (6)	5	...0H	1234H
D14	39H (9)	38H (8)	6	..01H	2345H
D15	42H (B)	41H (A)	7	.012H	3456H
D16	44H (D)	43H (C)	8	0123H	4567H
D17	31H (1)	30H (0)	9	...0H	1234H
D18	33H (3)	32H (2)	n	D102	D101
					D100

M8161=OFF8bit 模式

S	H	L	1		0H
D10	31H (1)	30H (0)	2		02H
D11	33H (3)	32H (2)	3		024H
D12	35H (5)	34H (4)	4	不变化	0246H
D13	37H (7)	36H (6)	5	...0H	2468H
D14	39H (9)	38H (8)	6	..02H	468AH
D15	42H (B)	41H (A)	7	.024H	68ACH
D16	44H (D)	43H (C)	8	0246H	8AC0H
D17	31H (1)	30H (0)	9	...0H	2468H
D18	33H (3)	32H (2)	n	D102	D101
					D100

其中，M8161 标志决定了变量宽度模式，当 M8161=OFF 时，为 16 位 bit 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=4 时，位的组成
(D10~D11) 转换

b15				D100								b0			
0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0
0				1				2				3			
b15				D101								b0			
0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0

当 M8161=OFF、n=5 时，位的组成
(D10~D11) 转换

b15				D100								b0			
0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0
1				2				3				4			

b15				D101								b0			
0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0
												0			

当 M8161=ON、n=4 时，位的组成
(D10~D11) 转换

b15				D100								b0			
0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0
0				2				4				6			
b15				D101								b0			
0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0

当 M8161=ON、n=5 时，位的组成
(D10~D11) 转换

b15				D100								b0			
0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0
2				4				6				8			
b15				D101								b0			
0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0
												0			

注：

RS/HEX/ASCII/CCD 等指令共用 M8161 模式标志，编程时注意；

S 数据区的源数据必须为 ASCII 码字符，否则转换出错；

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。

4.5.2 数据传送

数据传送	
MOV	赋值传送
EMOV	二进制浮点传送
BMOV	数据成批传送
FMOV	数据一对多传送
CML	数据取反传送

MOV 赋值传送

1. 指令形式

将源址 S 中的数据复制到终址 D

MOV S D			数据移动	指令执行	
S	数据源	进行传送的数据或数据存储字软元件地址		16 位指令（5step） MOV 连续执行 MOV P 脉冲执行	32 位指令（9step） DMOV 连续执行 DMOV P 脉冲执行
D	数据赋值目的地	目的地数据存储字软元件地址			

2. 操作数

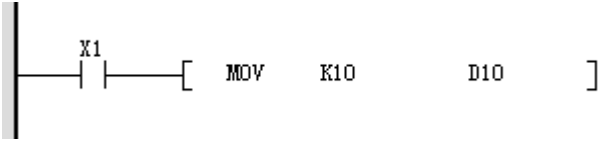
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 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：
（DMOVD1D10）的操作结果是：D1→D10；D2→D11。

【使用举例】



当 X1 为 ON 时，将常数 10 赋值给 D10 中。

EMOV 二进制浮点传送

1. 指令形式

进行二进制浮点数数据的传送。需要触点驱动，当指令执行时，将 S 的 2 进制浮点数数据值复制到 D 中。

EMOV S D			二进制浮点传送	指令执行
S	数据源	二进制浮点数数据的传送源		32 位指令（9step） DEMOV 连续执行 DEMOPV 脉冲执行
D	传送目的地	保存二进制浮点数数据的存放单元		

2. 操作数

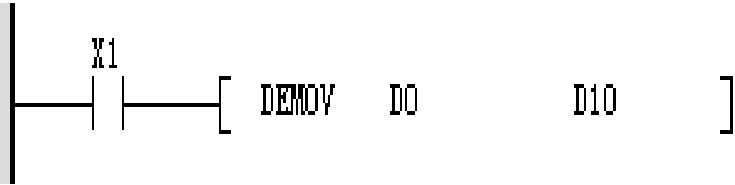
操作数	位软元件								字软元件													
	系统·用户								系统·用户				位数指定					变址		常数		实数
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 该指令是进行二进制浮点数数据的传送。需要触点驱动，当指令执行时，将 S 的 2 进制浮点数数据值复制到 D 中。其中：
- 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。

【使用举例】



当 X1 为 ON 时，（D1,D0）中的二进制浮点数保存到（D11,D10）中。

SMOV 移位传送（开发中）

1. 指令形式

将 S 中以 m1 数位为起始的共 m2 数位的数位数据移动到终址 D 中以 n 数位为起始的共 m2 数位中。

SMOV S m1 m2 D n				二进制浮点传送	指令执行
S	数据源	进行数位传送的数据存储字软元件地址			16 位指令（11step） SMOV 连续执行 SMOVP 脉冲执行
m1	传送起始数位	S中要移动的起始数位的位置			
m2	传送数位个数	S中要移动的数位的个数			
D	目的地装置	传送的目的地数据存储字软元件地址			
n	目的地起始数位	移入D中的起始数位的位置			

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
m1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
m2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

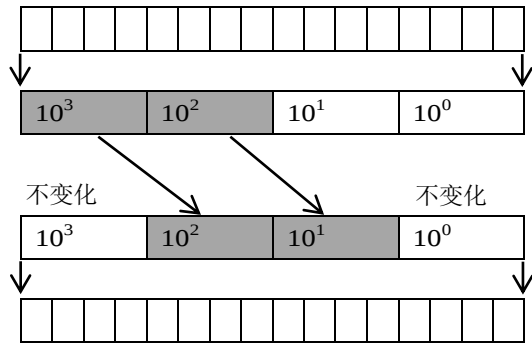
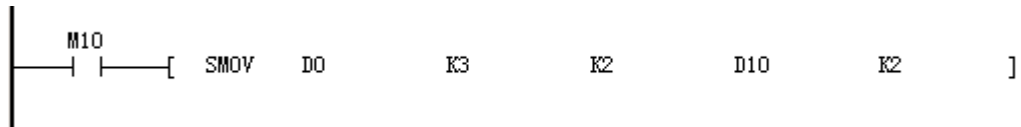
注：带灰色底纹软元件，表示可支持。

功能说明：

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，当 M8168 为 OFF 时是 BCD 模式（十进制的位），M 操作数的范围是 0000~9999，不能为负数。当 M8168 为 ON 时是 BIN 模式，S 操作数可以是负数。
- 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）。

【使用举例】



M8168=OFF 时，计算过程如下：

D0 (BIN 进制 16bit)

↓ (自动转换)

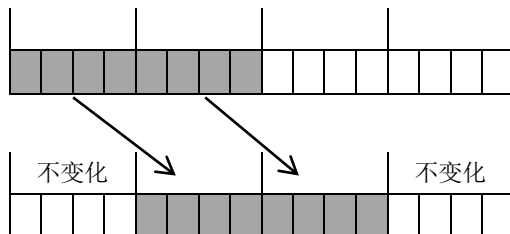
D0 (BCD 格式 4 位)

↓ (位移动)

D10 (BCD 格式 4 位)

↓ (自动转换)

D10 (BIN 格式 16bit)



M8168=ON 时，计算过程如下：

D0 (HEX 格式 4 位)

↓ (位移动)

D10 (HEX 格式 4 位)

假设 D0=K1234，D10=K5678，则当 M8168 为 OFF 时 (BCD 模式)，将 M10 置 ON，则 D10 的值变为 K5623；

当 M8168 为 ON 时 (BIN 模式)，此时 D0=H04D2=K1234，D10=H162E=K5678，将 M10 置 ON，则 D10=H12D2=K4818。

BMOV 数据成批传送

1. 指令形式

当驱动条件成立时，将以 S 为首址的 n 个寄存器的数据一一对应传送到以 D 为首址的 n 个寄存器中。

BMOV S D n			数据成批传送	指令执行
S	数据来源首址	进行批传送的数据存储字软元件首地址		16 位指令（7step） BMOV 连续执行 BMOVP 脉冲执行
D	传送目标首址	传送目的地的数据存储字软元件首地址		
n	数据长度	批传送的字软元件的点数		

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 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。其中 n 的取值范围是 1~512。
- 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。
- 当操作数为位元件时，S 和 D 位数必须相等。

【使用举例】



当 M1 为 ON 时，将（D3,D2,D1,D0）的值传送到（D13,D12,D11,D10）中。

FMOV 数据一对多传送

1. 指令形式

当驱动条件成立时，将 S 中的数据，传送到以 D 为首址的 n 个寄存器中。

FMOV S D n			数据一对多传送	指令执行	
S	数据源	进行一对多传送的数据或数据存储字软元件		16 位指令(7step) FMOV 连续执行 FMOVP 脉冲执行	32 位指令(13step) DFMOV 连续执行 DFMOVP 脉冲执行
D	数据传送目的地首址	传送目的地的数据存储字软元件首地址			
n	目标个数	多点传送的字软元件的点数			

2. 操作数

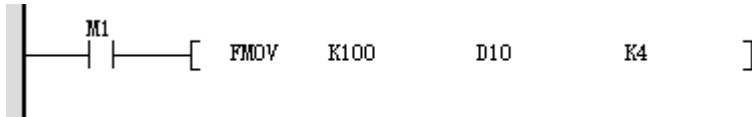
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 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。其中 n 的取值范围是 1~512。
- FMOV 是 16 位的多点传送指令，DFMOV 是 32 位多点传送。

【使用举例】



当 M1 为 ON 时，将 100 传送到 D13,D12,D11,D10 四个寄存器中。

CML 数据取反传送

1. 指令形式

将源址 S 按位取反后传送至终址 D。

CML S D			数据取反传送	指令执行	
S	取反数据源	进行数据取反的数据或数据存储字软元件地址	16 位指令(5step) CML 连续执行 CMLP 脉冲执行	32 位指 (9step)	DCML 连续执行
D	传送目的地	数据取反后传送的数据存储字软元件地址		DCMLP 脉冲执行	

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

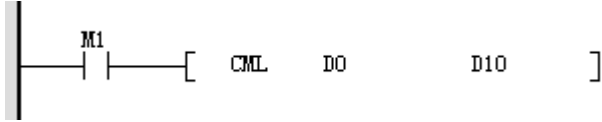
注：带灰色底纹软元件，表示可支持。

功能说明：

- 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。
- 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。

【使用举例】



当 M1 为 ON 时，将 D0 取反后存放在 D10 中。

4.5.3 表格操作

表格操作	
ZRST	全部数据复位

ZRST 全部数据复位

1. 指令形式
批量复位指令

ZRST D1 D2			全部数据复位	指令执行
D1	批复位首址	进行批复位的软元件首址		16 位指令（5step） ZRST 连续执行 ZRSTP 脉冲执行
D2	批复位终址	进行批复位的软元件终址		

2. 操作数

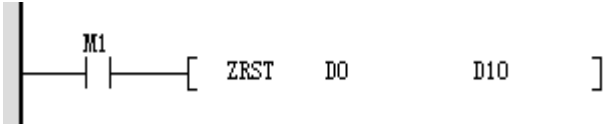
操作数	位软元件							字软元件														
	系统·用户							系统·用户				位数指定					变址		常数		实数	
D1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明

- 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。

【使用举例】



当 M1 为 ON 时，D0 到 D10 的所有寄存器全部复位。

补充说明：位元件 Y、M、S 和字元件 T、C、D 也可使用 RST 指令来单独复位；字元件 T、C、D 和位寄存器 KnY、KnM、KnS 也可以用 FMOV 来多点清除。

4.5.4 数据移位

数据移位	
ROR	循环右移
ROL	循环左移
RCR	带进位的循环右移
RCL	带进位的循环左移

ROR 循环右移

1. 指令形式

当驱动条件成立时，D 中数据向右移动 n 位，不含进位标志位 M8022，移出 D 的低位数据循环进入 D 的高位。

ROR D n			循环右移	指令执行	
D	将循环的装置	数据存储字软元件地址	有效范围: $1 \leq n \leq 16$ (16 位), $1 \leq n \leq 32$ (32 位)	16 位指令 (5step)	32 位指令 (9step)
n	单次移动位数			ROR 连续执行 RORP 脉冲执行	DROR 连续执行 DRORP 脉冲执行

2. 操作数

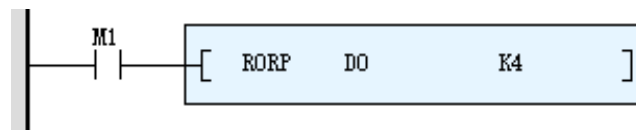
操作数	位软元件							字软元件													
	系统·用户							系统·用户					位数指定					变址		常数	
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z 修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z 修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

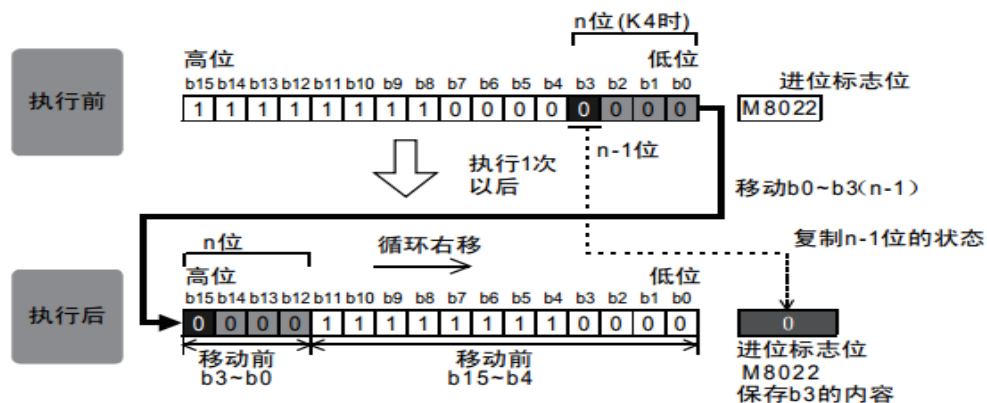
- 将循环右移 n 位。
- 本指令一般使用脉冲执行型指令。当为 32bit 指令时，寄存器变量则占用后续相邻地址的共 2 个单元。
- 若 D 中指定 KnY、KnM、KnS 时，只有 K4 (16bit) 及 K8 (32bit) 有效。

【使用举例】



当 M1 为 ON 时，D0 的数循环右移四位；

最后的位保持在进位标志位 M8022 中。



ROL 循环左移

1. 指令形式

当驱动条件成立时，D 中数据向左移动 n 位，不含进位标志位 M8022，移出 D 的高位数据循环进入 D 的低位。

ROL D n			循环左移	指令执行	
D	将循环的装置	数据存储字软元件地址		16 位指令（5step）	32 位指令（9step）
n	单次移动位数	有效范围:1≤n≤16(16 位), 1≤n≤32(32 位)		ROL 连续执行 ROLP 脉冲执行	DROL 连续执行 DROLP 脉冲执行

2. 操作数

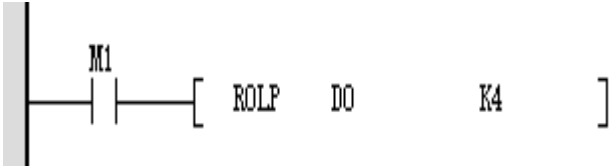
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

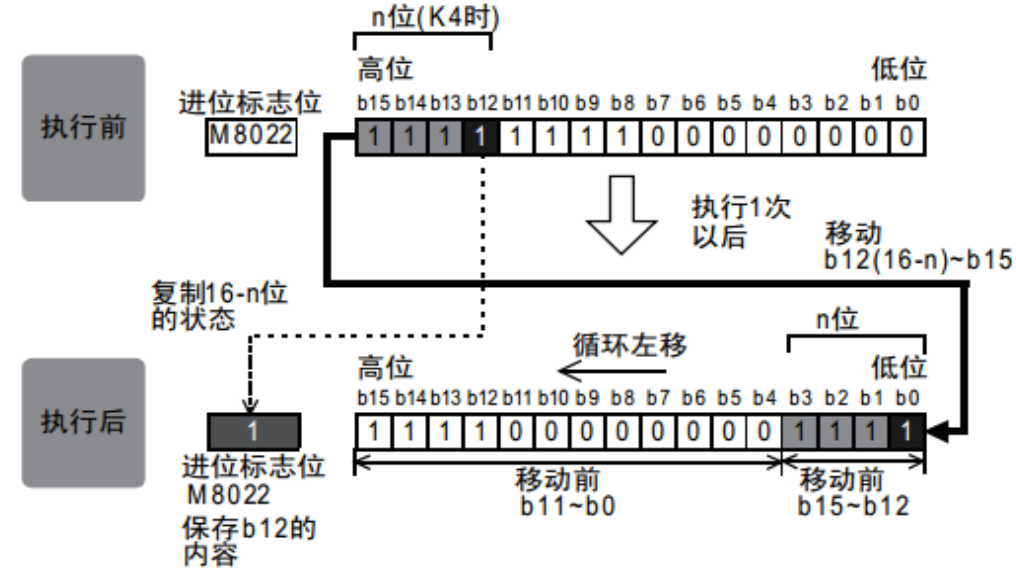
- 将 D 的内容循环左移 n 位。
- 本指令一般使用脉冲执行型指令。当为 32bit 指令时，寄存器变量则占用后续相邻地址的共 2 个单元。
- 若 D 中指定 KnY、KnM、KnS 时，只有 K4（16bit）及 K8（32bit）有效。

【使用举例】



当 M1 为 ON 时，D0 的数循环左移四位；

最后的位保持在进位标志位 M8022 中。



RCR 带进位的循环右移

1. 指令形式

当驱动条件成立时，D 中数据带进位（M8022）向右移动 n 位，移出的高位带进位（M8022）循环进入 D 的高位。

RCR D n			带进位的循环右移	指令执行	
D	将循环的装置	数据存储字软元件地址		16 位指令（5step）	32 位指令（9step）
n	单次移动位数	有效范围:1≤n≤16(16 位), 1≤n≤32(32 位)		RCR 连续执行 RCRP 脉冲执行	DRCR 连续执行 DRCRP 脉冲执行

2. 操作数

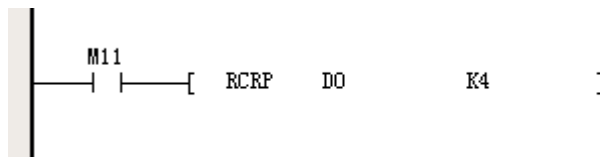
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

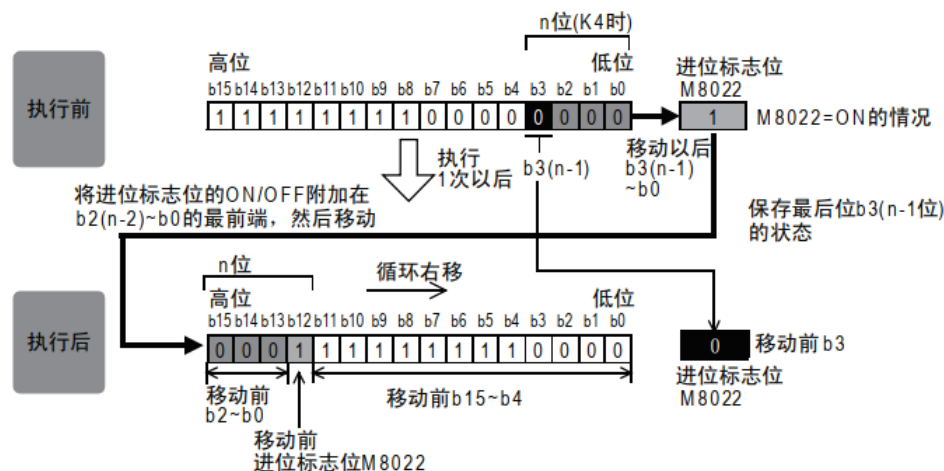
- 将 D 的内容连同进位标志 M8022 循环右移 n 位。
- 本指令一般使用脉冲执行型指令。
- 当为 32bit 指令时，寄存器变量则占用后续相邻地址的共 2 个单元。
- 若 D 中指定 KnY、KnM、KnS 时，只有 K4（16bit）及 K8（32bit）有效。

【使用举例】



当 M11 为 ON 时，D0 的数带进位循环右移四位；

最后的位保持在进位标志位 M8022 中。



RCL 带进位的循环左移

1. 指令形式

当驱动条件成立时，D 中数据带进位（M8022）向左移动 n 位，移出的高位带进位（M8022）循环进入 D 的低位。

RCL D n			带进位的循环左移	指令执行	
D	将循环的装置	数据存储字软元件地址	有效范围: $1 \leq n \leq 16$ (16 位), $1 \leq n \leq 32$ (32 位)	16 位指令 (5step)	32 位指令 (9step)
n	单次移动位数			RCL 连续执行 RCLP 脉冲执行	DRCL 连续执行 DRCLP 脉冲执行

2. 操作数

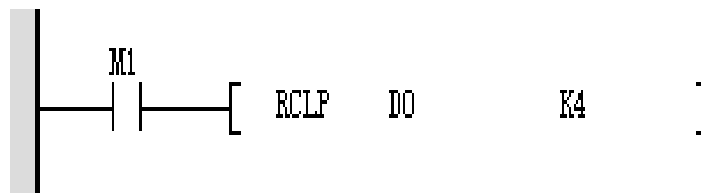
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

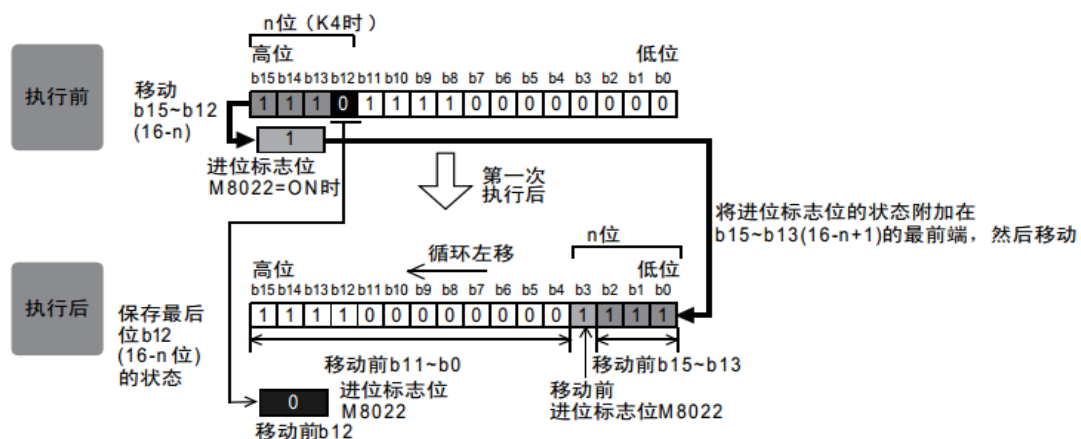
- 将 D 的内容连同进位标志 M8022 循环左移 n 位。
- 本指令一般使用脉冲执行型指令。当为 32bit 指令时，寄存器变量则占用后续相邻地址的共 2 个单元。
- 若 D 中指定 KnY、KnM、KnS 时，只有 K4（16bit）及 K8（32bit）有效。

【使用举例】



当 M1 为 ON 时，D0 的数带进位循环左移四位；

最后的位保持在进位标志位 M8022 中。



4.5.5 其他数据处理

其他数据处理	
SWAP	上下字节交换
BON	ON 位判断
SUM	ON 为总数
RND	产生随机数据
XCH	数据交换

SWAP 上下字节交换

1. 指令形式

将指定变量 S 的高低字节的值进行互相交换。

SWAP S		上下字节交换	指令执行	
S	操作数	欲执行上/下字节互换的数据存储单元	16 位指令（3step） SWAP 连续执行 SWAPP 脉冲执行	32 位指令（5step） DSWAP 连续执行 DSWAPP 脉冲执行

2. 操作数

操作数	位软元件							字软元件													
	系统·用户							系统·用户					位数指定					变址		常数	
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z 修饰	K	H	E

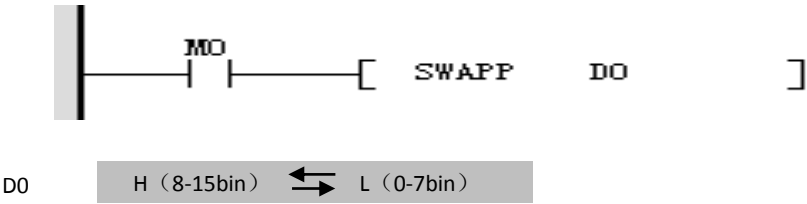
注：带灰色底纹软元件，表示可支持。

功能说明：

- 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，否则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。

【使用举例】



M0 由 OFF 变为 ON 时，将 D0 中的高字节 H 和低字节 L 的内容交换。

BON ON 位判断

1. 指令形式

当驱动条件成立时，源址 S 中二进制数据的第 K 位状态，控制 D 状态。

BON S D n			ON 位判断	指令执行	
S	源数据	数据或数据存储字软元件地址		16 位指令(7step) BON 连续执行 BONP 脉冲执行	32 位指令(13step) DBON 连续执行 DBONP 脉冲执行
D	被控位	被控位元件			
n	指定位	源址 S 中被指定的位 0≤n≤15(16 位), 0≤n≤31(32 位)			

2. 操作数

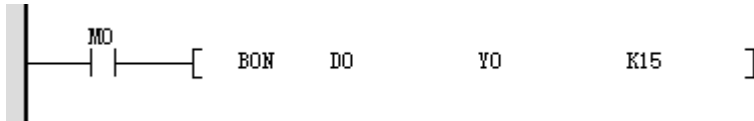
操作数	位软元件							字软元件														
	系统·用户							系统·用户				位数指定					变址		常数		实数	
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

判断 S 的第 n 位的状态，结果存入 D。

【使用举例】



当 M0 为 ON 时，D0 的第 15 位（15bin 为符号位）为 1 时，Y0 置位，bin15 为 0 时 Y0 复位。M0 由 ON 变成 Off 时，Y0 仍保持之前的状态。

SUM ON 位总数

1. 指令形式

当驱动条件成立时，对源址 S 中表示的二进制数据中为“1”的个数进行统计，统计结果存放到 D。

SUM S D			ON 位总数	指令执行	
S	被统计数	数据或数据存储字软元件地址		16 位指令(5step) SUM 连续执行	32 位指令 (9step) DSUM 连续执行
D	统计结果	数据存储软元件地址		SUMP 脉冲执行	DSUMP 脉冲执行

2. 操作数

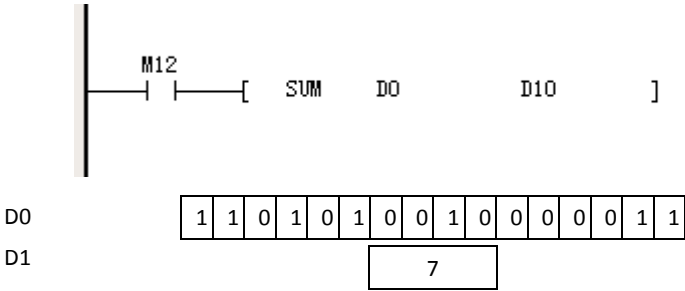
操作数	位软元件							字软元件														
	系统·用户							系统·用户				位数指定					变址		常数		实数	
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 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。

【使用举例】



RND 产生随机数据

1. 指令形式

产生随机数的指令。

RND D			产生随机数据	指令执行
D	目的地址	保存随机数的软元件		16 位指令（3step） RND 连续执行 RNDP 脉冲执行

2. 操作数

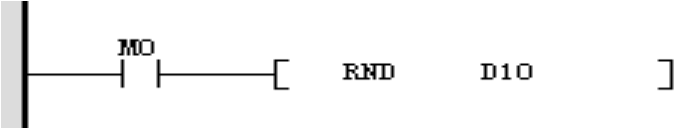
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 这个指令产生 0~32767 的伪随机数，将其数值作为随机数保存到[D]中。
- 请在 STOP→RUN 时仅仅写入一次非 0 的数值(-2, 147, 483, 648~2, 147, 483, 647) 至(D8311, D8310) 作为初始值。

【使用举例】



产生的随机数存放在 D10 中。

XCH 数据交换（开发中）

1. 指令形式

当驱动条件成立时，将 S 和 D 中的数据互换。

XCH S D			数据交换	指令执行	
S	数据 1	进行数据交换的数据存储字软元件 1		16 位指令(5step) XCH 连续执行 XCHP 脉冲执行	32 位指令 (9step) DXCH 连续执行 DXCHP 脉冲执行
D	数据 2	进行数据交换的数据存储字软元件 2			

2. 操作数

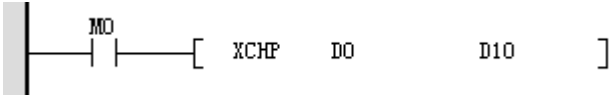
操作数	位软元件							字软元件														
	系统·用户							系统·用户				位数指定					变址		常数		实数	
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

需要触点驱动，有 2 个操作变量，将 S 和 D 的值彼此交换。

【使用举例】



当 M0 由 OFF 变为 ON 时，D0 和 D10 中的数据相互调换。

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。

4.6 时钟指令

时钟读写	
TRD	时钟数据读取
TWR	时钟数据写入

TRD 时钟数据读取

1. 指令形式

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。

TRD D		时钟数据读取	指令执行
D	时间存储首址	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	16 位指令（3step） TRD 连续执行 TRDP 脉冲执行

2. 操作数

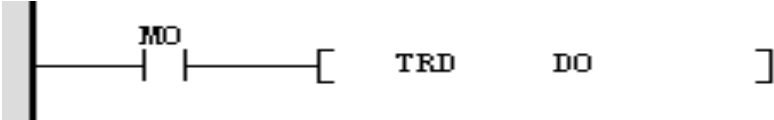
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 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：秒、分、时、日、月、年、星期等数据。

【使用举例】



当 M0 为 ON 时，指令的操作如下：

项目	系统变量		操作后 D
年(2000~2099)	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]	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 D8013~D8019 的值。

TWR 时钟数据写入

1. 指令形式

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。

TWR D			时钟数据写入	指令执行
D	时间写入数据首址	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		16 位指令 (3step) TWR 连续执行 TWRP 脉冲执行

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

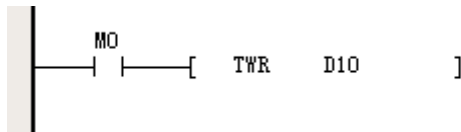
注: 带灰色底纹软元件, 表示可支持。

功能说明:

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

【使用举例一】



当 M0 为 ON 时, 指令的操作如下:

项目	系统变量		操作后 D
年(2000~2099)	D0	→	D8018
月(1~12)	D1	→	D8017
日(1~31)	D2	→	D8016
时(0~23)	D3	→	D8015
分(0~59)	D4	→	D8014
秒(0~59)	D5	→	D8013
星期[0(日)~6]	D6	→	D8019

注: 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。

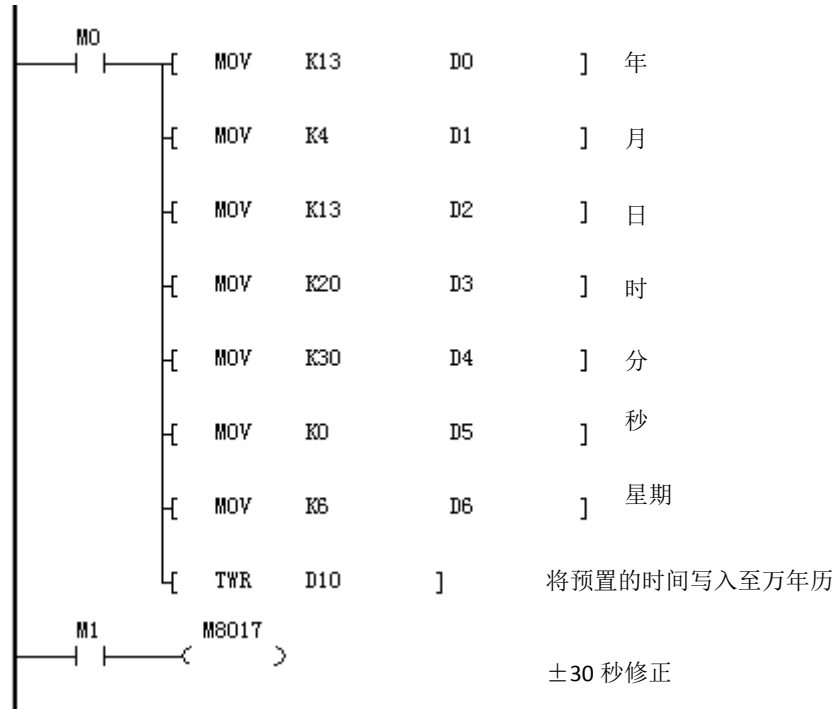
M8017 每 On 一次, PLC 内部时钟作±30 秒校正动作, 这里的校正是指当 PLC 的内部时钟的秒针于 1~29 时, 会被自动归为

“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。

M8015 置 ON 可以停止时钟计时。

PLC 内部时钟校时方法如下。

【使用举例二】



将 PLC 的现在时间调整为 2013 年 04 月 13 日 08 时 30 分 0 秒，星期六；提前一段时间写入至 D0-D6，当 M0 置 ON,将此时间设置进 PLC 中。

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 D8013~D8019，用 MOV 指令直接对 D8013~D8019 进行赋值时需要将 M8015 置位才能写入。
- 公历 4 位数模式的情况下，设定值（80-99）相当于（1980-1999 年），（00-79）相当于（2000-2079 年）。例如：80=1980 年，99=1999 年，00=2000 年，79=2079 年。

4.7 脉冲定位.通讯定位指令

脉冲输出	
PWM	脉宽调制输出
PLSY	脉冲输出
PLSR	带加减速脉冲输出
脉冲定位	
PLSV	可变速脉冲输出
PLSV2	带加减速可变速脉冲输出
DRVI	相对位置定位
DRVA	绝对位置定位
ZRN	原点回归
刷新处理	
REF	输入输出刷新
REFF	输入滤波时间调整

4.7.1 脉冲输出

脉冲输出	
PWM	脉宽调制输出
PLSY	脉冲输出
PLSR	带加减速脉冲输出

【注】：PWM，PLSY，PLSR 指令的详细使用请参考“[高速输出及定位指令](#)”章节。

4.7.2 脉冲定位

脉冲定位	
PLSV	可变速脉冲输出
PLSV2	带加减速可变速脉冲输出
DRVI	相对位置定位
DRVA	绝对位置定位
ZRN	原点回归
DVIT	中断定位
DPIT	最大定长的中断定位指令

【注】：PLSV，DRVI，DRVA，ZRN 指令的详细使用请参考“[高速输出及定位指令](#)”章节。

4.7.3 刷新处理

刷新处理	
REF	输入输出刷新
REFF	输入滤波时间调整

REF 输入输出刷新指令

1. 指令形式

立即更新输入或输出映像存储区

REF	S n	输入输出刷新	指令执行
S	位元件首址	需刷新的输入或输出位元件首址	16 位指令（5step） REF 连续执行 REFP 脉冲执行
n	位元件个数	需刷新的输入或输出位元件个数	

2. 操作数

操作数	位软元件							字软元件													
	系统·用户							系统·用户				位数指定					变址		常数		实数
S	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z 修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z 修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：将 D 地址开始的 n 个元件状态进行立即更新。由于 PLC 访问端口是按字节访问的特性，故要求：

S 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。

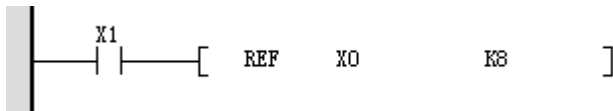
可用在 FOR~NEXT 指令之间、CJ 指令之间等。

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~X1 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。

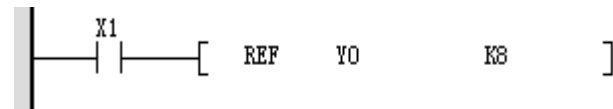
实际的输出端口状态变化延迟决定于输出元件（如继电器）的响应时间。输出刷新中的输出接点将在输出继电器（晶体管）应答时间后动作。继电器输出型的应答滞后时间约为 10ms（最大 20ms），晶体管输出型高速输出口约 10μs、普通点输出口约 0.5ms。

【使用举例 1】



当 X1 为 ON 时，会立即读取 X0-X7 的 8 个输入点状态，更新输入信号，没有产生输入延迟。

【使用举例 2】



当 X1 为 ON 时，会立即将 Y0~Y7 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 输入滤波时间调整

1. 指令形式

当驱动条件成立时，将 X0~X1 输入端口的滤波时间常数设定为 n 毫秒。

REFF n			滤波参数调整	指令执行
n	滤波时间	单位: ms		16 位指令 (5step) REFF 连续执行 REFFP 脉冲执行

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：将 X0~X1 输入端口的滤波时间常数设定为 n。

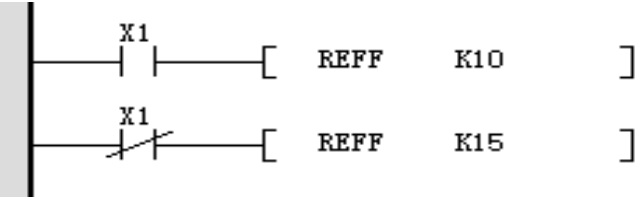
可编程控制器中，X0~X1 使用了数字滤波器，默认的滤波时间常数由 D8020 设定，通过 REFF 指令可将 D8020 改变为 0~60ms。

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。

【使用举例】



当 X1 为 ON 时，X0-X7 的滤波时间为 10ms，

当 X1 为 OFF 时，X0-X7 的滤波时间为 15ms。

4.8 通讯

通讯指令	
MODBUS	MODBUS 通讯

MODBUSMODBUS 通讯

1. 指令形式

MODBUS 通讯读写指令

MODBUS	S1	S2	n	D	MODBUS 通讯	指令执行
S1	通讯地址、功能码				从机地址（高字节）、通讯命令（低字节，按 MODBUS 协议定义）	16 位指令（9step） MODBUS 连续执行
S2	从站数据首址				访问从站的寄存器起始地址号	
n	数据长度				读或写的数据长度	
D	主机数据首址				读或写数据的存放单元起始地址，占用后续地址单元，长由 n 决定	

2. 操作数

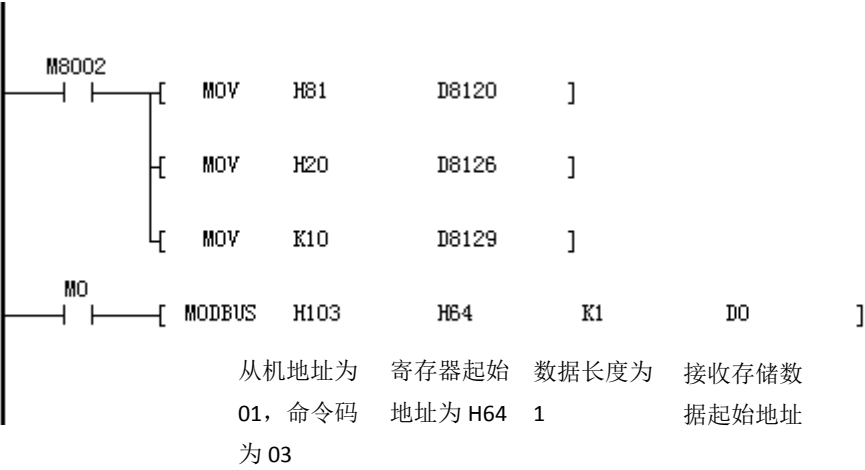
操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- S1：从机地址（高字节）、通讯命令（低字节，按 MODBUS 协议定义）。
- S2：访问从站的寄存器起始地址号。
- n：读或写的数据长度。
- D：读或写数据的存放单元起始地址，占用后续地址单元，长由 n 决定。

【使用举例】



将 M0 置 ON，PLC 不断的读取#1 从机内，地址为 H64 的寄存器的值，数据存放于 D0 单元中。

4.9 外设

PID 运算	
PID	PID 运算
其他外设指令	
ASC	ASCII 码转换

4.9.1 PID 运算

PID 运算	
PID	PID 运算

PID PID 运算

1. 指令形式

完成 PID 运算，用于闭环控制系统的控制

PID S1 S2 S3 D		PID 运算	指令执行
S1	目标值	PID 设定目标值	16 位指令（9step） PID 连续执行
S2	反馈值	实测的反馈值	
S3	运算参数	存放运算结果的起始单元	
D	输出值	PID 输出值存放单元	

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S3	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S4	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：表中灰底的软元件，表示可以支持的软元件。

3. 功能和指令说明

1) PID 功能选择

单元	功能	意义		备注
S3+0	采样周期	采样和输出刷新周期		
S3+1	运算模式选择	0x00**	增量式 PID	高 8 位控制指令运算模式
		0x01**	位置式 PID	

选择不同的功能指令，S3 单元对应不同的意义。

4. 增量式 PID 指令

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	设定范围 1~32767ms（默认为 10ms）

S3+1	动作方向	bit0: 0=正动作; 1=逆动作 bit1: 0=输入变化量报警无效; 1=输入变化量报警有效 bit2: 0=输出变化量报警无效; 1=输出变化量报警有效 bit3: 不可使用 bit4: 0=自整定不动作; 1=执行自整定 (目前版本暂时不提供此项功能) bit5: 0=输出值上下限设定无效; 1=输出值上下限设定有效 bit6~bit15 不可使用 另外, 请不要使 bit5 和 bit2 同时处于 ON。
S3+2	输入滤波常数	0~99[%], 0=没有输入滤波
S3+3	比例增益	0~32767[%]
S3+4	积分时间	0~32767 (×100ms), 0=作为∞处理 (无积分)
S3+5	微分增益	0~100[%], 0=无微分增益
S3+6	微分时间	0~32767 (×10ms), 0=无微分处理
在<ACT>的 bit1=1, bit2=1 或 bit5=1 时, S3+ (20~24) 被占用, 定义如下:		
S3+ (7~19)	PID 运算的内部处理占用, 运行前应清零	
S3+20	输入变化量 (增侧) 报警设定值	0~32767, (<ACT>的 bit1=1 时有效)
S3+21	输入变化量 (减侧) 报警设定值	0~32767, (<ACT>的 bit1=1 时有效)
S3+22	输出变化量 (增侧) 报警设定值	0~32767, (<ACT>的 bit1=1 时有效) 输出上限设定值-32768~32767, (<ACT>的 bit1=0, bit5=1 时有效) 注 1
S3+23	输出变化量 (减侧) 报警设定值	0~32767 (S3+1<ACT>的 bit2=1, bit5=0 时有效) 输出下限设定值-32768~32767, (<ACT>的 bit1=0, bit5=1 时有效) 注 1
S3+24	报警输出	bit0 输入变化量 (增侧) 溢出 bit1 输入变化量 (减侧) 溢出 Bit2 输出变化量 (增侧) 溢出 Bit3 输出变化量 (减侧) 溢出 (<ACT>的 bit1=1 或 bit2=1 时有效)
S3+25	PID 运算的内部处理占用	

注 1: 当输出被限幅后, PID 输出的是限幅后的值, 此后如果输出发生变化, 那么次变化也是在限幅后的输出值的基础上发生的变化。

5. 位置式 PID 指令 (S3+1 选择 0x01**)

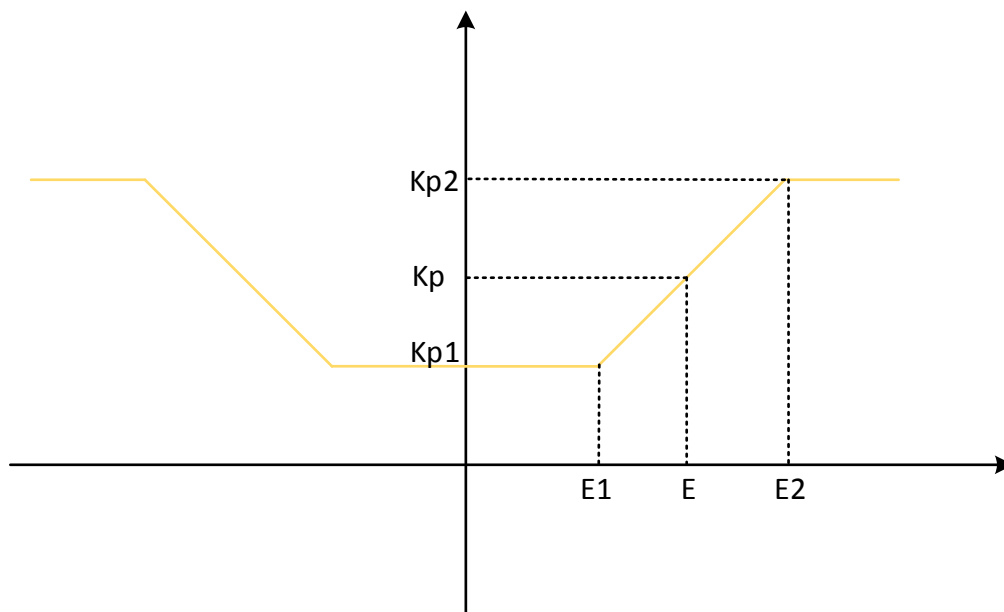
单元	功能	设定说明
S3	采样时间	设定范围 1~32767ms (默认 10ms)
S3+1	控制模式	0x0100: 正向 0x0101: 反向
S3+2	比例增益 1	0~32767[%]
S3+3	积分增益 1	0~32767[%]
S3+4	微分增益 1	0~32767[%]
S3+5	偏差死区	0~32767; 0: 不生效; 非 0: 偏差小于此值认为偏差为 0

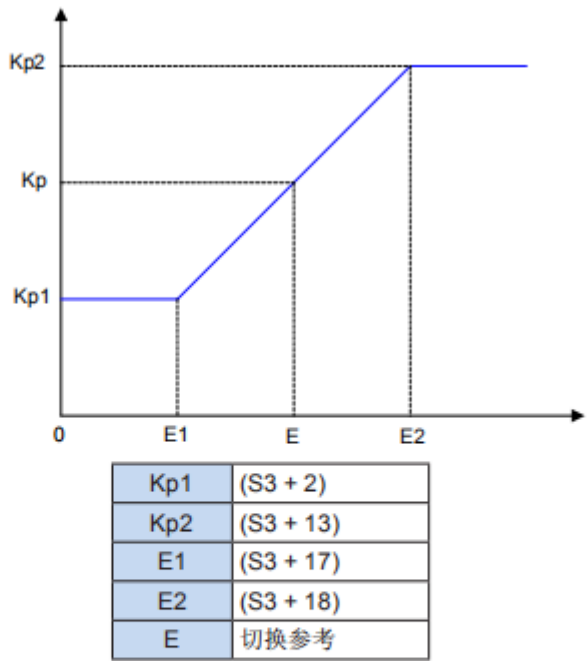
S3+6	输出上限	-32768~32767；输出最大值
S3+7	输出下限	-32768~32767；输出最小值
S3+8	积分上限	-32768~32767；累计积分最大值 ^{注1}
S3+9	积分下限	-32768~32767；累计积分最小值 ^{注1}
S3+10	累计积分	32 位浮点数
S3+11		
S3+12	上次偏差	-32768~32767；
S3+13	比例增益 2	0~32767[%]
S3+14	积分增益 2	0~32767[%]
S3+15	微分增益 2	0~32767[%]
S3+16	参数切换条件	0：不切换；1：根据偏差切换 ^{注2}
S3+17	偏差下限 ^{注3}	0~32767；偏差起点或者自定义切换起点
S3+18	偏差上限 ^{注3}	0~32767；偏差终点或者自定义切换终点
S3+19	保留	
S3+20~ S3+26	内部运算使用	

注 1：当积分上下限均设置为 0 时，积分上下限将按照上限 32767、下限-32768 生效。

注 2：偏差切换原理（比例增益 K_p 切换为例）。

注 3：偏差下限和偏差上限为偏差的绝对值。





当 $E \leq E1$, $Kp = Kp1$;
当 $E1 < E < E2$, $Kp = (Kp2 - Kp1) * E / (E2 - E1)$;
当 $E \geq E2$, $Kp = Kp2$;

S3 + 16	0	不切换
	1	$E = Sv - Pv $
	2	$E = S3 + 19$

错误码	错误表示的内容
6780	采样时间设置不合理
6781	保留
6782	输入滤波对象不合理
6783	比例系数不合理
6784	积分系数不合理
6785	微分系数不合理
6786	输出限制不正常（输出下限大于上限）

4.9.2 其他外设指令

其他外设指令	
ASC	ASCII 码转换

ASCASCII 码转换

1. 指令形式

当驱动条件成立时,将计算机输入的到 S1 中的字符串转换成 ASCII 码存放到以 D1 为首址的寄存器中。

ASCSD		ASCII 码转换	指令执行
S	数据源	欲执行ASCII码变换的英文字符串, 最大允许长度为8个字符	16 位指令 (11step) ASC 连续执行
D	转换结果	存放ASCII码的起始单元号, 占用随后的共4 (M8161=0) 或8 个变量单元 (M8161=1)	

2. 指令形式

操作数	位软元件							字软元件															
	系统·用户							系统·用户				位数指定				变址		常数		实数			
S1	用户输入相应的字母																						
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

【使用举例】



若 M8161=OFF，数字字符串转换成 ASCII 码后，按照低 8 位、高 8 位的顺序，每 2 个字符/1 个字节地保存在 D 中。

	高 8 位	低 8 位
D	42 (B)	41 (A)
D+1	44 (D)	43 (C)
D+2	46 (F)	45 (E)
D+3	00	47 (F)

若 M8161=ON，数字字符串转换成 ASCII 码后，然后将其依次传送到 D 的低 8 位（1 个字节）中。

	高 8 位	低 8 位
D	00	41 (A)
D+1	00	42 (B)
D+2	00	43 (C)
D+3	00	44 (D)
D+4	00	45 (E)
D+5	00	46 (F)

附：《ASCII 代码对照表》

10 进制位	ASCII (16 进制数)
0	30
1	31
2	32
3	33
4	34
5	35
6	36
7	37
8	38
9	39

10 进制位	ASCII (16 进制数)
0	30
1	31

英文字母	ASCII (16 进制数)	英文字母	ASCII (16 进制数)
A	41	N	4E
B	42	O	4F
C	43	P	50
D	44	Q	51
E	45	R	52
F	46	S	53
G	47	T	54
H	48	U	55
I	49	V	56
J	4A	W	57
K	4B	X	58
L	4C	Y	59
M	4D	Z	5A

4.10 电子凸轮指令

电子凸轮指令	
CAMWR	写入电子凸轮数据
CAMRD	读取电子凸轮数据
CAMSP	主轴/从轴位置计算

请参考“[第6章电子凸轮](#)”

第5章 高速输出及定位指令

5.1 指令概述

V5-M104 标准型的定位指令与轨迹控制均使用应用指令实现。特点如下：

- 高速输出频率范围：轴口为 1Hz~3MHz、高速端口为 1Hz~200kHz；
- 支持梯形加减速，S 型加减速；

5.1.1 高速输出指令属性表

指令	脉冲方向输出	梯形加减速	S 曲线加减速	加减速时间单独设定	运行中更改频率	运行中更改脉冲数	运行中换向	速度/位置控制
PLSY					√	√ (M)		速度 位置 速度+位置
PLSV	√				√		√	速度
PLSV2	√	√		√ (M)	√		√	速度
ZRN		√		√ (M)			√	速度
PLSR		√		√ (M)		√ (M)		位置
DRVA	√	√	√ (M)	√ (M)		√ (M)		位置
DRVI	√	√	√ (M)	√ (M)		√ (M)		位置
DVIT	√	√		√ (M)				速度+位置
DPIT	√	√		√ (M)				速度+位置
PWM					√			速度

- 各指令属性表中“√”表示具备/支持该属性，表中空白处表示不具备/不支持该属性；
- 各指令属性表中“√ (M)”表示需要置位特殊软元件才能使用该功能；
- 高速输出指令是否有加减速，是由指令本身的属性决定的，跟加减速时间无关。定位指令加减速时间范围：10ms-5000ms（插补指令范围加减速时间范围：10ms-500ms），小于最小范围值，以最小范围值计算，大于最大范围值，按最大范围值计算。

5.1.2 脉冲输出端口说明

V5-MC104 有 6 路高速脉冲输出口可供选择，包括 4 路轴口，2 路高速晶体管输出口，具体的端口定义见下表：

输出口编号	脉冲输出口	脉冲输出类型	结构	脉冲输出频率
1	Y300: PUL+/- Y301: DIR+/-	双路差分输出 ±5V	轴口	1Hz~3MHz
2	Y304: PUL+/- Y305: DIR+/-			
3	Y310: PUL+/- Y311: DIR+/-			
4	Y314: PUL+/- Y315: DIR+/-			
5	Y0	单路开集电极输出	IO 端子口	1Hz~200kHz

6	Y1	5~24V		
---	----	-------	--	--

5.1.3 脉冲输出端口特殊软元件

高速输出指令涉及到众多的特殊寄存器和继电器，跟脉冲相关的特殊软元件定义如下：

1) 轴口软元件定义如下：

轴 1	轴 2	轴 3	轴 4	Y0	Y1	属性
SD61	SD161	SD261	SD761	SD861	SD961	脉冲输出形式设定 0: 方向/脉冲 1: AB 正交 2: CW/CCW
Y302	Y306	Y312	Y316			轴口使能输出 ^[1]
X301	X305	X311	X315			轴口报警输入
SD56	SD156	SD256	SD756			轴口脉冲输入计数 ^[3]
SD38	SD138	SD238	SD738	SD838	SD938	当前速度

注 [1]: 通过设置软元件 ON，开启使能。

[3]: 轴口脉冲输入计数用于对编码器计数，脉冲输入形式固定为 AB 正交形式。

2) 特殊 M 元件定义如下：

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8340	M8360	M8380	M8400	M8420	M8440	脉冲输出中
M8341	M8361	M8381	M8401	M8421	M8441	ZRN 等清零信号输出有效
M8342	M8362	M8382	M8402	M8422	M8442	保留
M8343	M8363	M8383	M8403	M8423	M8443	保留
M8344	M8364	M8384	M8404	M8424	M8444	保留
M8345	M8365	M8385	M8405	M8425	M8445	保留
M8346	M8366	M8386	M8406	M8426	M8446	保留
M8349	M8369	M8389	M8409	M8429	M8449	脉冲输出停止
M8350	M8370	M8390	M8410	M8430	M8450	加减速单独设定/脉冲更改有效
M8352	M8372	M8392	M8412	M8432	M8452	脉冲输出完成中断使能
M8353	M8373	M8393	M8413	M8433	M8453	保留
M8354	M8374	M8394	M8414	M8434	M8454	DSZR 指令执行异常结束标志位
M8355	M8375	M8395	M8415	M8435	M8455	PLSV2 加速标志
M8356	M8376	M8396	M8416	M8436	M8456	PLSV2 减速标志
M8357	M8377	M8397	M8417	M8437	M8457	保留
M8358	M8378	M8398	M8418	M8438	M8458	保留
M8359	M8379	M8399	M8419	M8439	M8459	保留

3) 特殊 D 元件定义如下：

Y300	Y304	Y310	Y314	Y0	Y1	属性
D8340	D8360	D8380	D8400	D8420	D8440	脉冲输出计数值 (PLS)
D8341	D8361	D8381	D8401	D8421	D8441	
D8342	D8362	D8382	D8402	D8422	D8442	单独设定时的最高速度 (Hz)
D8343	D8363	D8383	D8403	D8423	D8443	

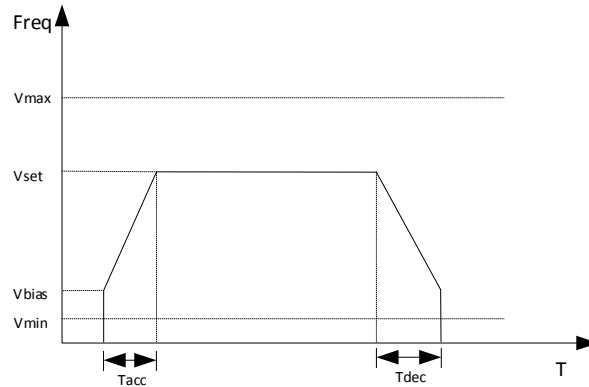
D8344	D8364	D8384	D8404	D8424	D8444	原点回归速度 (Hz)
D8345	D8365	D8385	D8405	D8425	D8445	
D8346	D8366	D8386	D8406	D8426	D8446	单独设定时的爬行速度 (Hz)
D8347	D8367	D8387	D8407	D8427	D8447	单独设定时的基底速度 (Hz)
D8348	D8368	D8388	D8408	D8428	D8448	单独设定时的加速时间 (ms)
D8349	D8369	D8389	D8409	D8429	D8449	单独设定时的减速时间 (ms)
D8350	D8370	D8390	D8410	D8430	D8450	清零软元件编号
D8500/D8501						统一设定时的最高速度 (Hz)
D8502						统一设定时的基底速度 (Hz)
D8503						统一设定时的加减速时间 (ms)

5.1.4 输出频率及加减速时间

脉冲输出的频率及加减速时间遵循以下原则：

- 控制器输出的脉冲频率要在最大频率和最小频率之间；
- 控制器输出频率在加速初期及减速末期的输出频率大于基底速度设定的频率；
- 加速时间为基底速度加速到设定速度的时间；
- 减速时间为设定速度减速到基底速度的时间。

输出频率及加减速时间的关系见下图所示：



其中：

Vset: 用户设定的脉冲输出频率；

Vmin: 最低速度；

Vbias: 基底速度，特殊软元件设定，出厂默认为 500Hz；

Vmax: 最高频率，特殊软元件设定，轴口出厂默认为 3MHz，端口出厂默认为 200kHz；

Tacc: 加速时间，从基底速度加速到设定速度的时间；

Tdec: 减速时间，从设定速度减速到基底速度的时间。

5.2 定位指令

5.2.1 定位指令一览

指令	说明
PLSY	脉冲输出指令
PLSV	可变速脉冲输出指令
PLSV2	带加减速的可变速脉冲输出指令
PLSR	带加减速的脉冲输出指令
DRVA	绝对位置控制指令
DRVI	相对位置控制指令
ZRN	原点回归指令
DSZR	DOG 搜索原点回归
DVIT	中断定位
DPIT	最大定长的中断定位指令

5.2.2 PLSY 脉冲输出指令

1. 指令形式

脉冲输出指令，以指定的脉冲输出频率输出设定的脉冲个数。

PLSY S1 S2 D				脉冲输出指令	指令执行	
S1	输出频率（Hz）			设定的脉冲输出频率	16 位指令（7step） PLSY 连续执行	32 位指令（13step） DPLSY 连续执行
S2	输出个数（PLS）			设定的脉冲输出个数		
D	输出端口			高速脉冲输出端口		

2. 操作数

操作数	位软元件							字软元件															
	系统 用户							系统 用户					位数指定					变址			常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E

注：表中灰底的软元件，表示可以支持的软元件。

- ① S1 为脉冲输出频率，单位为 Hz，对于 16 位指令 (PLSY)，设定范围为 $10 \sim 32767$ ；对于 32 位指令 (DPLSY)，设定范围为 $10 \sim 3,000,000$ ；在指令执行的过程中可以修改 S1 的值。假定设置的频率超过 3000000，系统将限制为 3000000。
- ② S2 为脉冲输出个数，单位为 PLS，对于 16 位指令 (PLSY)，设定范围为 $-32768 \sim 32767$ ；对于 32 位指令 (DPLSY)，设定范围为 $-2,147,483,648 \sim 2,147,483,647$ ；在指令执行的过程中可以修改 S2 的值。
- ③ S2 的值设置为 0 时，表示以 S1 的频率一直发送正脉冲，S2 的值设置为 $0x80000000$ （32 位指令）时，表示以 S1 的频率一直发送负脉冲，只有当指令的能流 OFF 时脉冲输出才停止。
- ④ D 为脉冲输出端口，可以指定 Y300/Y304/Y310/Y314/Y0/Y1。

3. 软元件

下面对脉冲输出相关的软元件的作用做详细说明。

◆ 脉冲输出计数

Y300	Y304	Y310	Y314	Y0	Y1	属性
D8340	D8360	D8380	D8400	D8420	D8440	脉冲输出计数值（PLS）
D8341	D8361	D8381	D8401	D8421	D8441	

脉冲输出计数值记录了端口输出的脉冲个数。该元件具有 D 软元件的特性，可以通过相关指令清零，也可以掉电清零。控制器状态由 RUN→OFF 时，该元件不会被清零。轴在运行过程中，不能被清零。

◆ 脉冲输出中

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8340	M8360	M8380	M8400	M8420	M8440	脉冲输出中

如下图所示，当 Y304 脉冲输出时，M8360 被置位，当脉冲输出完成后，M8360 被自动复位。

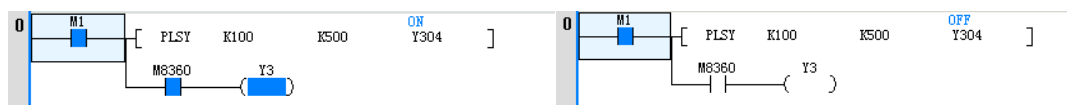


图 a：脉冲输出中

图 b：脉冲输出完成

◆ 脉冲输出停止

Y300	Y304	Y310	Y314	Y0	Y1	属性
------	------	------	------	----	----	----

M8349	M8369	M8389	M8409	M8429	M8449	脉冲输出停止
-------	-------	-------	-------	-------	-------	--------

当脉冲输出停止标志位被置位时脉冲输出停止，即使能流有效，脉冲也不会输出。软元件作用效果见下图所示，在软元件置位前，脉冲正常输出，当软元件置位后，即使能流有效，脉冲也是不会输出。

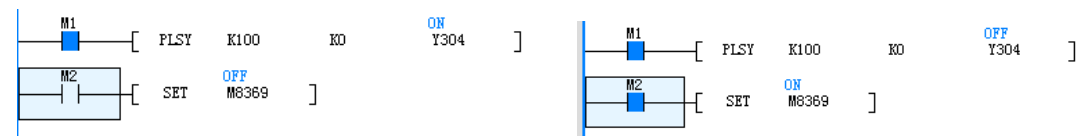


图 a：脉冲输出停止置位前

图 b：脉冲输出停止置位后

◆ 脉冲输出完成

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8029						脉冲输出完成

当本条脉冲输出完成后，相应的 M8029 会被置位，但本条脉冲输出完成并不影响别的脉冲指令的脉冲输出完成标志位，如下图中的，第一条脉冲指令执行完成后，M10 被置位，但是 M11 和 M12 均未被置位。

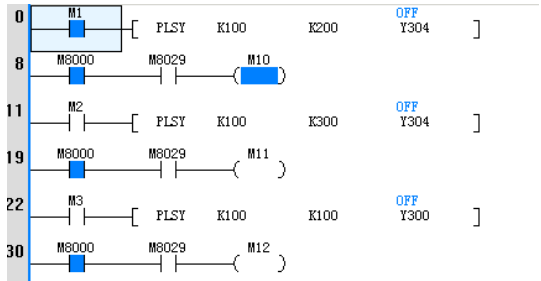


图 a：M8029 软元件动作

◆ 脉冲输出完成中断

若想在脉冲输出完成时进入中断，那么针对不同的 Y 口可开启不同的输出完成中断使能软元件。见下表，如想在 Y300 脉冲输出完成后进入中断，那么需要将 M8352 置为 1。

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8352	M8372	M8392	M8412	M8432	M8452	输出完成中断使能

◆ 不支持加减速。

4. 运行中更改参数

◆ 指令执行中，允许修改脉冲输出个数

更改前，需要置位特殊软元件，见下表，当脉冲更改有效软元件有效时，才能进行脉冲个数的更改。

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8350	M8370	M8390	M8410	M8430	M8450	加减速时间单独设定/脉冲更改有效

- ① 脉冲输出个数从 0 改成非 0 值时，输出模式由速度模式变成位置模式，更改后，控制器发出设定的脉冲个数后停止输出。
- ② 不能将脉冲输出个数从非 0 值改成 0。
- ③ 更改的脉冲个数要大于当前脉冲位置。
- ④ 后续过程中，多次修改，位置仍然是相对第一次进入时的绝对位置。

◆ 指令执行中，允许修改脉冲输出频率

- ① 在指令的输出输出过程中，允许修改脉冲输出的频率，更改的脉冲输出频率可大于也可以小于当前脉冲输出频率。

- ② 指令的脉冲输出方向是由输出脉冲个数来控制的，
当脉冲输出个数设定大于 0 时，为正向脉冲输出，
当脉冲输出个数设定小于 0 时，为负向脉冲输出。
- ③ 脉冲输出频率只能设定大于 0 的值。也即若把脉冲输出个数设定为 0，把 PLSV 指令用作速度模式使用时，只能发出正转速度指令。

5.2.3 PLSV 可变脉冲输出指令

1. 指令形式

可变脉冲输出指令，指定的脉冲频率和方向输出。

PLSV S1 D1 D2				可变速脉冲输出指令	指令执行	
S1	输出频率 (Hz)			设定的脉冲输出频率	16 位指令 (7step) PLSV 连续执行	32 位指令 (13step) DPLSV 连续执行
D1	输出端口			高速脉冲输出端口		
D2	输出方向			脉冲运行方向		

2. 操作数

操作数	位软元件							字软元件															
	系统 用户							系统 用户					位数指定					变址		常数		实数	
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
D1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
D2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E

注：表中灰底的软元件，表示可以支持的软元件。

- ① S1 为脉冲输出频率，单位为 Hz，对于 16 位指令 (PLSV)，设定范围为-32768~-1；1~32767；对于 32 位指令 (DPLSV)，设定范围为-3,000,000~-1；1~3,000,000；在指令执行的过程中可以修改 S1 的值。
- ② D1 为脉冲输出端口，可以指定 Y300/Y304/Y310/Y314/Y0/Y1。
- ③ D2 为运行方向端口或位变量，当输出脉冲为正时，元件状态为 ON，当脉冲方向为负时，元件状态为 OFF。

3. 软元件

下面对脉冲输出相关的软元件的作用做详细说明。

◆ 脉冲输出计数

Y300	Y304	Y310	Y314	Y0	Y1	属性
D8340	D8360	D8380	D8400	D8420	D8440	脉冲输出计数值 (PLS)
D8341	D8361	D8381	D8401	D8421	D8441	

脉冲输出计数值记录了端口输出的脉冲个数。该元件具有 D 软元件的特性，可以通过相关指令清零，也可以掉电清零。控制器状态由 RUN->OFF 时，该元件不会被清零。

◆ 脉冲输出中

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8340	M8360	M8380	M8400	M8420	M8440	脉冲输出中

如下图所示，当 Y304 脉冲输出时，M8360 被置位，当脉冲输出完成后，M8360 被自动复位。

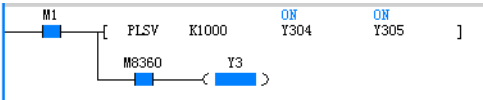


图 a：脉冲输出中

◆ 脉冲输出停止

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8349	M8369	M8389	M8409	M8429	M8449	脉冲输出停止

当脉冲输出停止标志位被置位时脉冲输出停止，即使能流有效，脉冲也不会输出。软元件作用效果见下图所示，在软元件置位前，脉冲正常输出，当软元件置位后，即使能流有效，脉冲也不会输出。

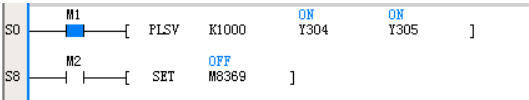


图 a：脉冲输出停止置位前



图 b：脉冲输出停止置位后

◆ 无脉冲输出完成标志。

◆ 无脉冲输出完成中断。

◆ 无加减速时间。

4. 运行中更改参数

◆ 指令执行中，允许修改脉冲输出频率

- ①在指令的输出输出过程中，允许修改脉冲输出的频率，更改的脉冲输出频率可大于也可以小于当前脉冲输出频率。
- ②脉冲输出中允许更改脉冲输出方向，可以通过更改设定的脉冲输出频率来实现；输出频率为正值，方向为正，输出频率为负值，方向为负。

5.2.4 PLSV2 带加减速的可变脉冲输出指令

1. 指令形式

带加减速的可变脉冲输出指令，指定的脉冲频率和方向输出。

PLSV2 S1 D1 D2			带加减速的可变脉冲 输出指令	指令执行		
S1	输出频率（Hz）		设定的脉冲输出频率		16 位指令(7step) PLSV2 连续执行	32 位指令（13step） DPLSV2 连续执行
D1	输出端口		高速脉冲输出端口			
D2	输出方向		脉冲运行方向			

2. 操作数

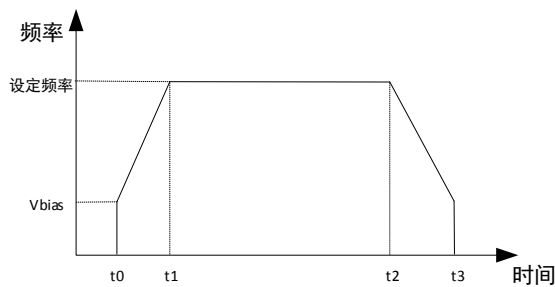
操作数	位软元件							字软元件															
	系统 用户							系统 用户					位数指定					变址			常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
D1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
D2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E

注：表中灰底的软元件，表示可以支持的软元件。

- ① S1 为脉冲输出频率，单位为 Hz，对于 16 位指令 (PLSV2)，设定范围为 $-32768 \sim -1$ ； $1 \sim 32767$ ；对于 32 位指令 (DPLSV2)，设定范围为 $-3,000,000 \sim -1$ ； $1 \sim 3,000,000$ ；在指令执行的过程中可以修改 S1 的值。
- ② D1 为脉冲输出端口，可以指定 Y300/Y304/Y310/Y314/Y0/Y1。
- ③ D2 为运行方向端口或位变量，当输出脉冲为正时，元件状态为 ON，当脉冲方向为负时，元件状态为 OFF。

3. 脉冲输出

- ① PLSV2 脉冲的输出示意图如下图所示，当指令能流 ON 时，脉冲开始输出，频率从基底速度逐渐加速到设定速度。



- ② 当脉冲输出的能流 OFF 时，脉冲输出频率将逐渐减速到基底频率后停止输出。注意在减速的过程中，脉冲输出中标志处于 ON 时，将不接受指令的再次驱动。输出停止后，当能流由 OFF→ON 时，脉冲输出处理重新开始。

4. 软元件

下面对脉冲输出相关的软元件的作用做详细说明。

◆ 脉冲输出计数

Y300	Y304	Y310	Y314	Y0	Y1	属性
D8340	D8360	D8380	D8400	D8420	D8440	脉冲输出计数值 (PLS)
D8341	D8361	D8381	D8401	D8421	D8441	

脉冲输出计数值记录了端口输出的脉冲个数。该元件具有 D 软元件的特性，可以通过相关指令清零，

也可以掉电清零。控制器状态由 RUN->OFF 时，该元件不会被清零。轴在运行过程中，不能被清零。

◆ 脉冲输出中

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8340	M8360	M8380	M8400	M8420	M8440	脉冲输出中

如下图所示，当 Y304 脉冲输出时，M8360 被置位，当脉冲输出完成后，M8360 被自动复位。

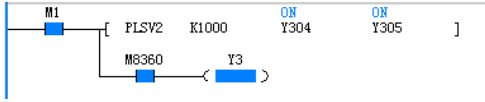


图 a：脉冲输出中

◆ 脉冲输出停止

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8349	M8369	M8389	M8409	M8429	M8449	脉冲输出停止

当脉冲输出停止标志位被置位时脉冲输出停止，即使能流有效，脉冲也不会输出。软元件作用效果见下图所示，在软元件置位前，脉冲正常输出，当软元件置位后，即使能流有效，脉冲也不会输出。



图 a：脉冲输出停止置位前

图 b：脉冲输出停止置位后

◆ 无脉冲输出完成标志。

◆ 无脉冲输出完成中断。

◆ 加减速时间单独设定/脉冲更改有效

若想各输出轴具有不同的加减速时间，或者想在运行的过程中更改脉冲个数，那么可通过设定下述软元件：

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8350	M8370	M8390	M8410	M8430	M8450	加减速时间单独设定/脉冲更改有效

注意：该软元件是“加减速单独设定”和“脉冲更改有效”共用的软元件。

当上述“加减速时间单独设定”软元件为 OFF 时，对应轴参数使用如下软元件：

Y300	Y304	Y310	Y314	Y0	Y1	属性
D8500/D8501						最高速度（Hz）
D8502						基底速度（Hz）
D8503						加减速时间（ms）

当某一轴“加减速时间单独设定”软元件为 ON 时，该对应的轴参数使用下述软元件：

Y300	Y304	Y310	Y314	Y0	Y1	属性
D8342	D8362	D8382	D8402	D8422	D8442	最高速度（Hz）
D8343	D8363	D8383	D8403	D8423	D8443	
D8347	D8367	D8387	D8407	D8427	D8447	基底速度（Hz）

D8348	D8368	D8388	D8408	D8428	D8448	加速时间 (ms)
D8349	D8369	D8389	D8409	D8429	D8449	减速时间 (ms)

5. 运行中修改参数

在运行中更改脉冲输出频率，如下图 a/b 所示。从 V1 更改到 V2 或者从 V2 更改到 V3，更改速度时，速度不会突变，而是按照设定的加减速时间来加速或者减速到更改后的速度。在图 a 中，速度从 V1 加速到 V2 的加速度跟速度从基底速度加速到 V1 的加速度相同；速度从 V2 减速到 V3 的加速度跟 V3 减速到基底速度的加速度相同。

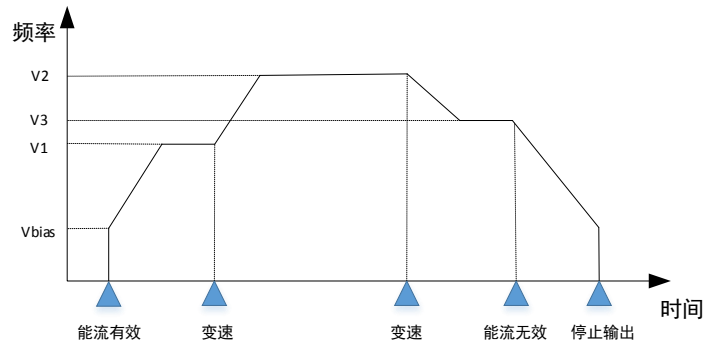


图 a: PLSV2 脉冲输出变化示意图

在图 b 中，速度从基底速度加速到 V1 的加速度跟速度从 0 加速到 V2 的加速度以及速度从 0 加速到 V3 的加速度相同；速度从 V1 减速到 0 的加速度跟速度从 V2 加速到 0 的加速度以及速度从 V3 减速到基底速度的加速度均相同。

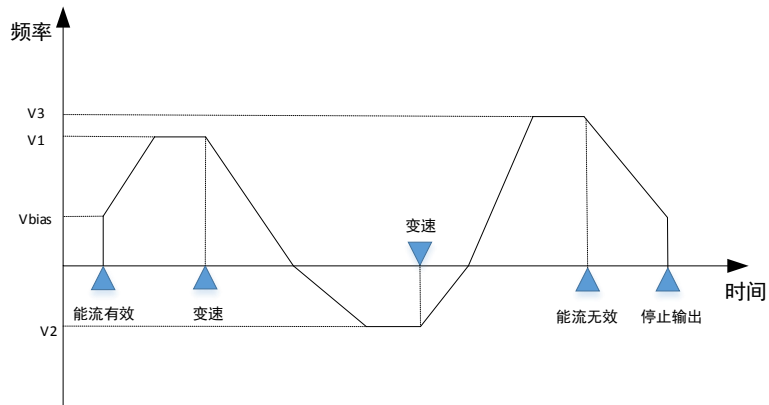


图 b: PLSV2 脉冲输出变化示意图（反向变速）

5.2.5 PLSR 带加减速脉冲输出

1. 指令形式

以设定的加减速时间，指定的脉冲频率，发出设定的脉冲个数。

PLSR S1 S2 S3 D					带加减速脉冲输出	指令执行	
S1	输出频率（Hz）		设定的脉冲输出频率			16 位指令(9step) PLSR 连续执行	32 位指令（17step） DPLSR 连续执行
S2	输出个数（PLS）		设定的脉冲输出个数				
S3	加减速时间（ms）		设定的加减速时间				
D	输出端口		高速脉冲输出端口				

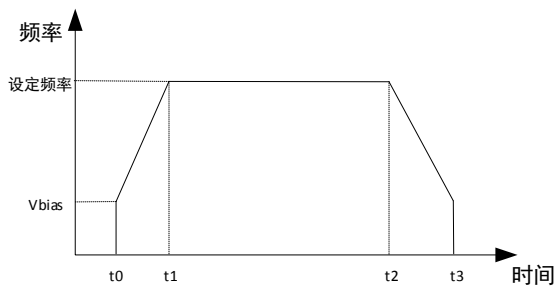
2. 操作数

操作数	位软元件							字软元件															
	系统 用户							系统 用户					位数指定					变址			常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
S3	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E

注：表中灰底的软元件，表示可以支持的软元件。

- ①S1 为脉冲输出频率，单位为 Hz，对于 16 位指令（PLSR），设定范围为 10~32767；对于 32 位指令（DPLSR），设定范围为 10~3,000,000；在指令执行的过程中可以修改 S1 的值。
- ②S2 为脉冲输出个数，单位为 PLS，对于 16 位指令（PLSR），设定范围为-32768~32767；对于 32 位指令（DPLSR），设定范围为-2,147,483,648~2,147,483,647；在指令执行的过程中可以修改 S2 的值。
- ③S3 位设定的加减速时间，范围 10~5000（ms），默认加速时间跟减速时间相同。可通过软元件修改。
- ④D1 为脉冲输出端口，可以指定 Y300/Y304/Y310/Y314/Y0/Y1。

3. 脉冲输出



- ①当使能能流后，脉冲输出频率将按照设定的加速时间（ t_1-t_0 ）从基底速度加速到设定速度，当输出频率开始减速时，将按照设定的减速时间（ t_3-t_2 ）从设定速度减速到基底速度。
- ②当脉冲输出的能流 OFF 时，脉冲输出频率将逐渐减速到基底频率后停止输出。注意在减速的过程中，脉冲输出中标志处于 ON 时，将不接受指令的再次驱动。输出停止后，当能流由 OFF->ON 时，脉冲输出处理重新开始。
- ② 行中可以修改参数详见“5. 运行中更改参数”

4. 软元件

下面对脉冲输出相关的软元件的作用做详细说明。

◆ 脉冲输出计数

Y300	Y304	Y310	Y314	Y0	Y1	属性
D8340	D8360	D8380	D8400	D8420	D8440	脉冲输出计数值（PLS）
D8341	D8361	D8381	D8401	D8421	D8441	

脉冲输出计数值记录了端口输出的脉冲个数。该元件具有 D 软元件的特性，可以通过相关指令清零，也可以掉电清零。控制器状态由 RUN->OFF 时，该元件不会被清零。

◆ 脉冲输出中

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8340	M8360	M8380	M8400	M8420	M8440	脉冲输出中

如下图所示，当 Y304 脉冲输出时，M8360 被置位，当脉冲输出完成后，M8360 被自动复位。

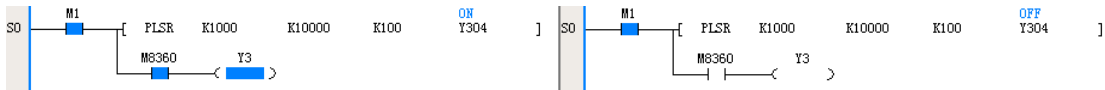


图 a：脉冲输出中

图 b：脉冲输出完成

◆ 脉冲输出停止

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8349	M8369	M8389	M8409	M8429	M8449	脉冲输出停止

当脉冲输出停止标志位被置位时脉冲输出停止，即使能流有效，脉冲也不会输出。软元件作用效果见下图所示，在软元件置位前，脉冲正常输出，当软元件置位后，即使能流有效，脉冲也会停止输出。

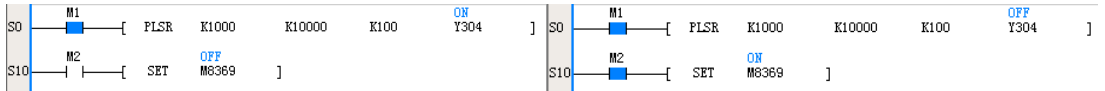


图 a：脉冲输出停止置位前

图 b：脉冲输出停止置位后

◆ 脉冲输出完成

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8029						脉冲输出完成

当本条脉冲输出完成后，相应的 M8029 会被置位，但本条脉冲输出完成并不影响别的脉冲指令的脉冲输出完成标志位，如下图中的，第一条脉冲指令执行完成后，M10 被置位，但是 M11 和 M12 均未被置位。

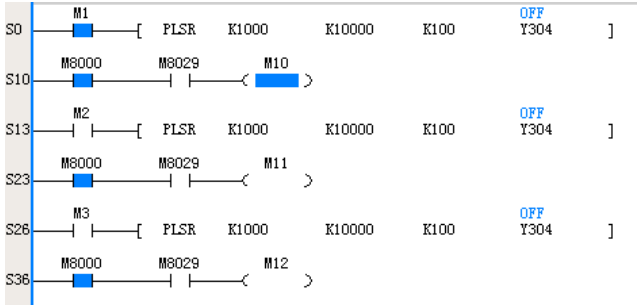


图 a：M8029 软元件动作

◆ 脉冲输出完成中断

若想在脉冲输出完成时进入中断，那么针对不同的 Y 口可开启不同的输出完成中断使能软元件。见下表，如想在 Y300 脉冲输出完成后进入中断，那么需要将 M8352 置为 1。

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8352	M8372	M8392	M8412	M8432	M8452	输出完成中断使能

◆ 加减速时间单独设定/脉冲更改有效

若想各输出轴具有不同的加减速时间，或者想在运行的过程中更改脉冲个数，那么可通过设定下述软元件：

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8350	M8370	M8390	M8410	M8430	M8450	加减速时间单独设定/脉冲更改有效

注意：该软元件是“加减速单独设定”和“脉冲更改有效”共用的软元件。

当上述“加减速时间单独设定”软元件为 OFF 时，对应轴参数使用如下软元件：

Y300	Y304	Y310	Y314	Y0	Y1	属性
D8500/D8501						最高速度 (Hz)
D8502						基底速度 (Hz)
指令本身设定 (S3)						加减速时间 (ms)

当某一轴“加减速时间单独设定”软元件为 ON 时，该对应的轴参数使用下述软元件：

Y300	Y304	Y310	Y314	Y0	Y1	属性
D8342	D8362	D8382	D8402	D8422	D8442	最高速度 (Hz)
D8343	D8363	D8383	D8403	D8423	D8443	
D8347	D8367	D8387	D8407	D8427	D8447	基底速度 (Hz)
指令本身设定 (S3)						加速时间 (ms)
D8349	D8369	D8389	D8409	D8429	D8449	减速时间 (ms)

5. 运行中更改参数

◆ 指令执行中，允许修改脉冲输出个数

更改前，需要置位特殊软元件，见下表，当脉冲更改有效软元件有效时，才能进行脉冲个数的更改。

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8350	M8370	M8390	M8410	M8430	M8450	加减速时间单独设定/脉冲/频率更改有效

更改脉冲输出个数的值可以大于当前位置，也可以小于当前位置。

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8357	M8377	M8397	M8417	M8437	M8457	位置修改生效方式

注：减速所需要的位置公式： $1/2a * (\text{当前速度}^2 - \text{基底速度}^2)$

- ① 当修改脉冲输出个数小于当前位置时，需将置位位置修改生效方式特殊软元件，否则会报警并按照原来设置的脉冲输出个数运行。
- ② 当修改脉冲输出个数大于当前位置时，如下图 a 所示，红色曲线为位置改小后的曲线，绿色曲线为位置改大之后的曲线。注意，这里的改大改小都是相对于目标位置的改大改小。
- ③ 后续过程中，多次修改，位置仍然是相对第一次进入时的绝对位置。

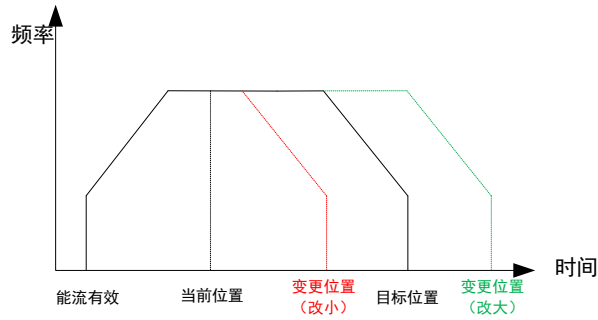


图 a: 修改脉冲输出个数大于当前位置

- ④ 当修改脉冲输出个数小于减速所需要的位置时，分为两种情况：

M8357 为 OFF：当前修改的位置将不生效，并报错“6713，定位位置偏小，不能减速”。

M8357 为 ON：如下图 b 所示，当修改后的位置小于当前位置时，修改后脉冲输出频率会逐渐减速到 0 再反向运行到修改后的位置。

注意：脉冲输出速度在减速段的斜率按照减速时间计算出来，反向加速段的加速度斜率按照加速时间计算出来。

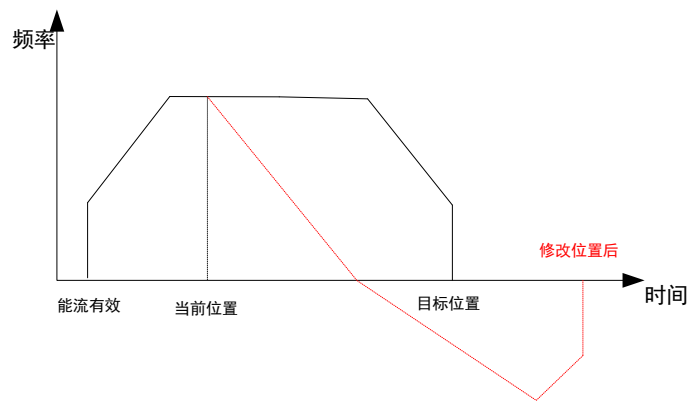
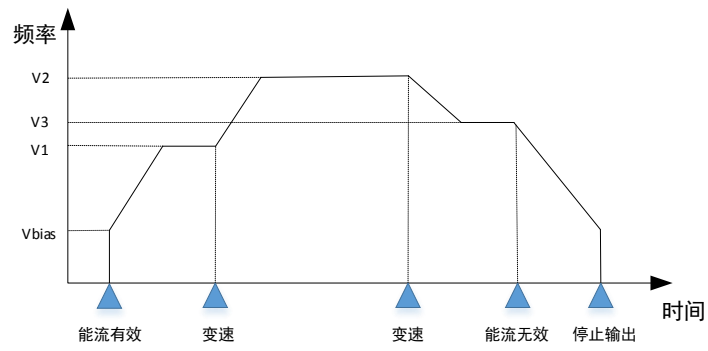


图 b: 修改脉冲输出个数小于当前位置

- ◆ 指令执行中，允许修改脉冲输出频率。

- ① 指令的输出过程中，允许修改脉冲输出的频率，更改的脉冲输出频率可大于也可以小于当前脉冲输出频率，范围为[10, 3000000]。
- ② 当频率大于 3000000 时，将限制到 3000000，当频率小于 10 时，将报错 6706，数据不合理。
- ③ 速度变化过程中，加速度与减速度斜率保持不变。
- ④ 如果同时修改脉冲个数与频率，则按照位置先生效，下一个扫描周期生效频率。



- ◆ 不支持 S 曲线。
- ◆ 不支持加速度修改。

5.2.6 DRVA 绝对位置定位

1. 指令形式

以设定的输出端口，指定的脉冲频率和方向，发出设定的脉冲个数，基于绝对位置的运动。

DRVA S1 S2 D1 D2					绝对位置定位	指令执行	
S1	脉冲个数		设定的脉冲输出个数（PLS）			16 位指令(9step) DRVA 连续执行	32 位指令（17step） DDRVA 连续执行
S2	脉冲频率		设定的脉冲输出频率（Hz）				
D1	输出端口		高速脉冲输出端口				
D2	输出方向		脉冲运行方向端口或位变量				

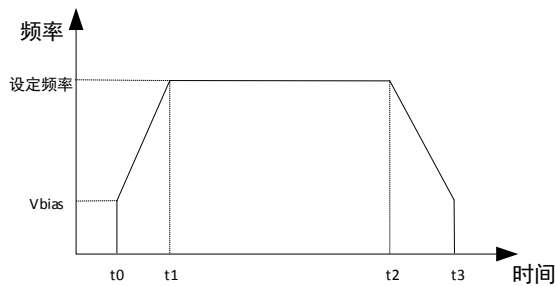
2. 操作数

操作数	位软元件							字软元件															
	系统 用户							系统 用户					位数指定					变址			常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
D1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
D2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E

注：表中灰底的软元件，表示可以支持的软元件。

- ①S1为脉冲输出个数，单位为PLS，对于16位指令(DRVA)，设定范围为-32768~32767；对于32位指令(DDRVA)，设定范围为-2,147,483,648~2,147,483,647；在指令执行的过程中可以修改S1的值。
- ②S2为脉冲输出频率，单位为Hz，对于16位指令(DRVA)，设定范围为10~32767Hz；对于32位指令(DDRVA)，设定范围为10~3,000,000Hz；在指令执行的过程中可以修改S2的值。
- ③D1为脉冲输出端口，可以指定Y300/Y304/Y310/Y314/Y0/Y1。
- ④D2为运行方向输出端口或位变量，该位的状态由控制器自身脉冲输出状态决定，当脉冲输出为正向运行时，D2的状态为ON，当脉冲输出为反向运行时，D2的状态为OFF。

3. 脉冲输出



- ①当使能能流后，脉冲输出频率将按照设定的加速时间（ t_1-t_0 ）从基底速度加速到设定速度，当输出频率开始减速时，将按照设定的减速时间（ t_3-t_2 ）从设定速度减速到基底速度。
- ②当脉冲输出的能流OFF时，脉冲输出频率将逐渐减速到基底频率后停止输出。注意在减速的过程中，脉冲输出中标志处于ON时，将不接受指令的再次驱动。输出停止后，当能流由OFF→ON时，脉冲输出处理重新开始。
- ③运行中可以修改参数详见“5.运行中更改参数”

4. 软元件

下面对脉冲输出相关的软元件的作用做详细说明。

◆ 脉冲输出计数

Y300	Y304	Y310	Y314	Y0	Y1	属性
D8340	D8360	D8380	D8400	D8420	D8440	脉冲输出计数值（PLS）
D8341	D8361	D8381	D8401	D8421	D8441	

脉冲输出计数值记录了端口输出的脉冲个数。该元件具有 D 软元件的特性，可以通过相关指令清零，也可以掉电清零。控制器状态由 RUN->OFF 时，该元件不会被清零。

◆ 脉冲输出中

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8340	M8360	M8380	M8400	M8420	M8440	脉冲输出中

如下图所示，当 Y304 脉冲输出时，M8360 被置位，当脉冲输出完成后，M8360 被自动复位。



图 a：脉冲输出中

图 b：脉冲输出完成

◆ 脉冲输出停止

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8349	M8369	M8389	M8409	M8429	M8449	脉冲输出停止

当脉冲输出停止标志位被置位时脉冲输出停止，即使能流有效，脉冲也不会输出。软元件作用效果见下图所示，在软元件置位前，脉冲正常输出，当软元件置位后，即使能流有效，脉冲也会停止输出。

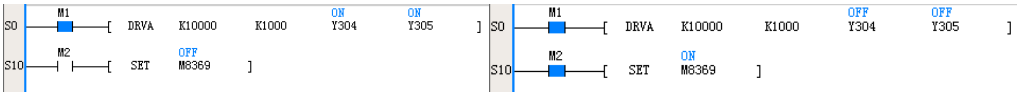


图 a：脉冲输出停止置位前

图 b：脉冲输出停止置位后

◆ 脉冲输出完成

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8029						脉冲输出完成

当本条脉冲输出完成后，相应的 M8029 会被置位，但本条脉冲输出完成并不影响别的脉冲指令的脉冲输出完成标志位，如下图中的，第一条脉冲指令执行完成后，M10 被置位，但是 M11 和 M12 均未被置位。

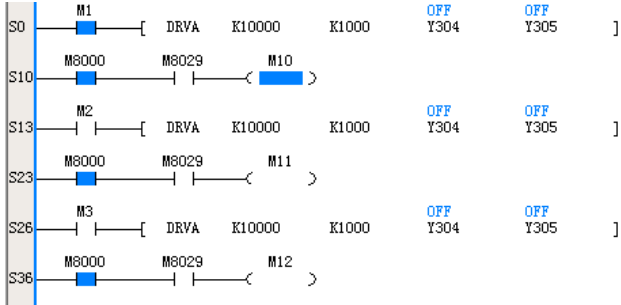


图 a：M8029 软元件动作

◆ 脉冲输出完成中断

若想在脉冲输出完成时进入中断，那么针对不同的 Y 口可开启不同的输出完成中断使能软元件。见下表，如想在 Y300 脉冲输出完成后进入中断，那么需要将 M8352 置为 1。

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8352	M8372	M8392	M8412	M8432	M8452	输出完成中断使能

◆ 加减速时间单独设定/脉冲更改有效

若想各输出轴具有不同的加减速时间，或者想在运行的过程中更改脉冲个数，那么可通过设定下述软元件：

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8350	M8370	M8390	M8410	M8430	M8450	加减速时间单独设定/脉冲更改有效

注意：该软元件是“加减速单独设定”和“脉冲更改有效”共用的软元件。

当上述“加减速时间单独设定”软元件为 OFF 时，对应轴参数使用如下软元件：

Y300	Y304	Y310	Y314	Y0	Y1	属性
D8500/D8501						最高速度（Hz）
D8502						基底速度（Hz）
D8503						加减速时间（ms）

当某一轴“加减速时间单独设定”软元件为 ON 时，该对应的轴参数使用下述软元件：

Y300	Y304	Y310	Y314	Y0	Y1	属性
D8342	D8362	D8382	D8402	D8422	D8442	最高速度（Hz）
D8343	D8363	D8383	D8403	D8423	D8443	
D8347	D8367	D8387	D8407	D8427	D8447	基底速度（Hz）
D8348	D8368	D8388	D8408	D8428	D8448	加速时间（ms）
D8349	D8369	D8389	D8409	D8429	D8449	减速时间（ms）

5. 运行中更改参数

◆ 指令执行中，允许修改脉冲输出个数

更改前，需要置位特殊软元件，见下表，当脉冲更改有效软元件有效时，才能进行脉冲个数的更改。

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8350	M8370	M8390	M8410	M8430	M8450	加减速时间单独设定/脉冲更改有效

更改脉冲输出个数的值可以大于当前位置，也可以小于当前位置。

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8357	M8377	M8397	M8417	M8437	M8457	位置修改生效方式

注：减速所需要的位置公式： $1/2a \times (\text{当前速度}^2 - \text{基底速度}^2)$

- ① 当修改脉冲输出个数小于当前位置时，需将置位位置修改生效方式特殊软元件，否则会报警并按照原来设置的脉冲输出个数运行。
- ② 当修改脉冲输出个数大于当前位置时，如下图 a 所示，红色曲线为位置改小后的曲线，绿色曲线为位置改大之后的曲线。注意，这里的改大改小都是相对于目标位置的改大改小。
- ③ 后续过程中，多次修改，位置仍然是相对第一次进入时的绝对位置。

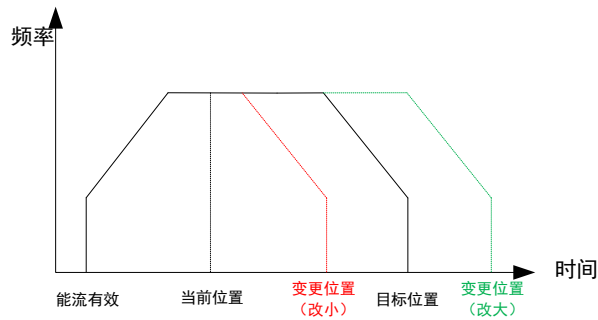


图 a: 修改脉冲输出个数大于当前位置

- ④ 当修改脉冲输出个数小于减速所需要的位置时，分为两种情况：

M8357 为 OFF：当前修改的位置将不生效，并报错“6713，定位位置偏小，不能减速”。

M8357 为 ON：如下图 b 所示，当修改后的位置小于当前位置时，修改后脉冲输出频率会逐渐减速到 0 再反向运行到修改后的位置。

注意：脉冲输出速度在减速段的斜率按照减速时间计算出来，反向加速段的加速度斜率按照加速时间计算出来。

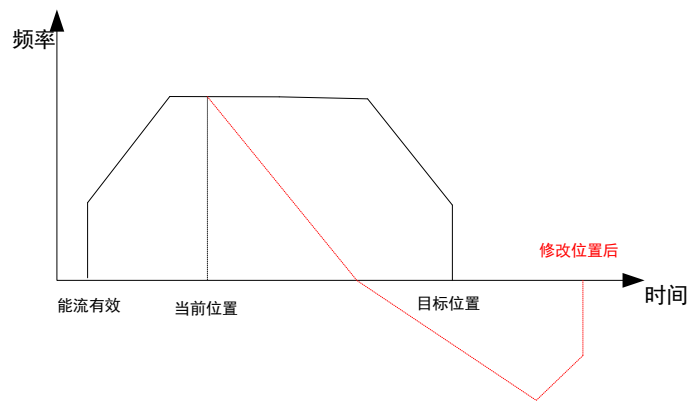
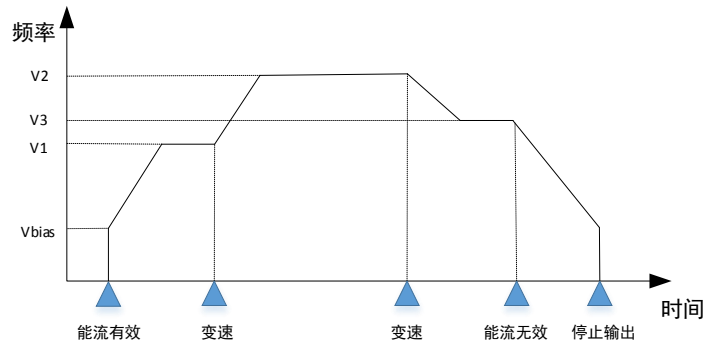


图 b: 修改脉冲输出个数小于当前位置

◆ 指令执行中，允许修改脉冲输出频率。

- ① 指令的输出过程中，允许修改脉冲输出的频率，更改的脉冲输出频率可大于也可以小于当前脉冲输出频率，范围为[10, 3000000]。
- ② 当频率大于 3000000 时，将限制到 3000000，当频率小于 10 时，将报错 6706，数据不合理。
- ③ 速度变化过程中，加速度与减速度斜率保持不变。
- ④ 如果同时修改脉冲个数与频率，则按照位置先生效，下一个扫描周期生效频率。



◆ 指令执行中，允许修改加速度。
(见补充说明文档)

◆ 支持 S 曲线功能

通过置位特殊软元件“S 曲线加减速使能标志位”来设置区分，如果标志位未置位，默认为梯形加减速。S 型曲线加减速见下表：

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8347	M8367	M8387	M8407	M8417	M8437	S 曲线加减速使能标志位
D8351	D8371	D8391	D8411	D8431	D8451	S 曲线时间

DRVA，DRVI 支持 S 曲线加减速，故在同等机械稳定性条件下可以让提升目标速度，因此可以缩短定位时间，提升加工效率。

S 曲线时间范围：[2~499].---需要扩大范围，请联系厂家。

实际加减速时间=T 型加减速+S 曲线时间。

5.2.7 DRVI 相对位置定位

1.指令形式

以设定的输出端口，指定的脉冲频率和方向，发出设定的脉冲个数，基于相对位置的运动。

DRVI S1 S2 D1 D2					相对位置定位	指令执行	
S1	脉冲个数		设定的脉冲输出个数（PLS）			16 位指令(9step) DRVA 连续执行	32 位指令（17step） DDRVA 连续执行
S2	脉冲频率		设定的脉冲输出频率（Hz）				
D1	输出端口		高速脉冲输出端口				
D2	输出方向		脉冲运行方向端口或位变量				

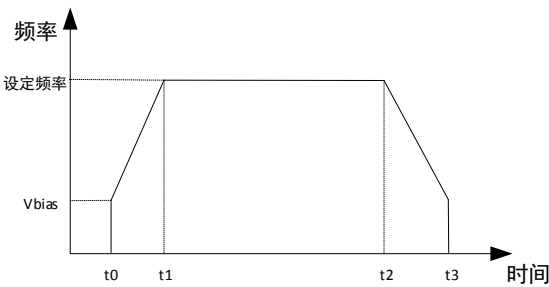
2. 操作数

操作数	位软元件							字软元件															
	系统 用户							系统 用户					位数指定					变址			常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
S3	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E

注：表中灰底的软元件，表示可以支持的软元件。

- ①S1为脉冲输出个数，单位为PLS，对于16位指令(DRVI)，设定范围为-32768~32767；对于32位指令(DDRVI)，设定范围为-2,147,483,648~2,147,483,647；在指令执行的过程中可以修改S1的值。
- ②S2为脉冲输出频率，单位为Hz，对于16位指令(DRVI)，设定范围为10~32767Hz；对于32位指令(DDRVI)，设定范围为10~3,000,000Hz；在指令执行的过程中可以修改S2的值。
- ③D1为脉冲输出端口，可以指定Y300/Y304/Y310/Y314/Y0/Y1。
- ④D2为运行方向输出端口或位变量，该位的状态由控制器自身脉冲输出状态决定，当脉冲输出为正向运行时，D2的状态为ON，当脉冲输出为反向运行时，D2的状态为OFF。

3. 脉冲输出



- ①当使能能流后，脉冲输出频率将按照设定的加速时间（t1-t0）从基底速度加速到设定速度，当输出频率开始减速时，将按照设定的减速时间（t3-t2）从设定速度减速到基底速度。
- ②当脉冲输出的能流OFF时，脉冲输出频率将逐渐减速到基底频率后停止输出。注意在减速的过程中，脉冲输出中标志处于ON时，将不接受指令的再次驱动。输出停止后，当能流由OFF->ON时，脉冲输出处理重新开始。
- ③运行中可以修改参数详见“5.运行中更改参数”

4. 软元件

下面对脉冲输出相关的软元件的作用做详细说明。

◆ 脉冲输出计数

Y300	Y304	Y310	Y314	Y0	Y1	属性
D8340	D8360	D8380	D8400	D8420	D8440	脉冲输出计数值（PLS）
D8341	D8361	D8381	D8401	D8421	D8441	

脉冲输出计数值记录了端口输出的脉冲个数。该元件具有 D 软元件的特性，可以通过相关指令清零，也可以掉电清零。控制器状态由 RUN->OFF 时，该元件不会被清零。

◆ 脉冲输出中

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8340	M8360	M8380	M8400	M8420	M8440	脉冲输出中

如下图所示，当 Y304 脉冲输出时，M8360 被置位，当脉冲输出完成后，M8360 被自动复位。



图 a：脉冲输出中

图 b：脉冲输出完成

◆ 脉冲输出停止

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8349	M8369	M8389	M8409	M8429	M8449	脉冲输出停止

当脉冲输出停止标志位被置位时脉冲输出停止，即使能流有效，脉冲也不会输出。软元件作用效果见下图所示，在软元件置位前，脉冲正常输出，当软元件置位后，即使能流有效，脉冲也会停止输出。

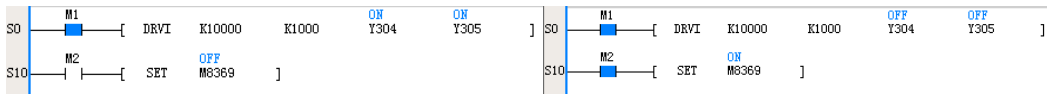


图 a：脉冲输出停止置位前

图 b：脉冲输出停止置位后

◆ 脉冲输出完成

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8029						脉冲输出完成

当本条脉冲输出完成后，相应的 M8029 会被置位，但本条脉冲输出完成并不影响别的脉冲指令的脉冲输出完成标志位，如下图中的，第一条脉冲指令执行完成后，M10 被置位，但是 M11 和 M12 均未被置位。

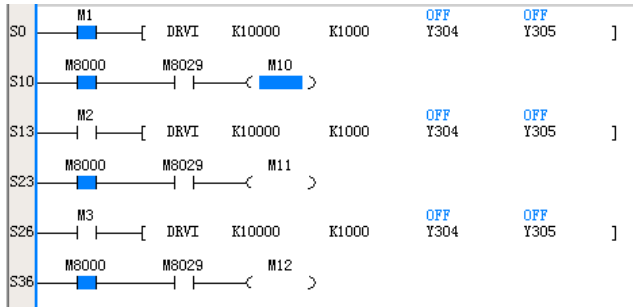


图 a：M8029 软元件动作

◆ 脉冲输出完成中断

若想在脉冲输出完成时进入中断，那么针对不同的 Y 口可开启不同的输出完成中断使能软元件。见下表，如想在 Y300 脉冲输出完成后进入中断，那么需要将 M8352 置为 1。

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8352	M8372	M8392	M8412	M8432	M8452	输出完成中断使能

◆ 加减速时间单独设定/脉冲更改有效

若想各输出轴具有不同的加减速时间，或者想在运行的过程中更改脉冲个数，那么可通过设定下述软元件：

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8350	M8370	M8390	M8410	M8430	M8450	加减速时间单独设定/脉冲更改有效

注意：该软元件是“加减速单独设定”和“脉冲更改有效”共用的软元件。

当上述“加减速时间单独设定”软元件为 OFF 时，对应轴参数使用如下软元件：

Y300	Y304	Y310	Y314	Y0	Y1	属性
D8500/D8501						最高速度（Hz）
D8502						基底速度（Hz）
D8503						加减速时间（ms）

当某一轴“加减速时间单独设定”软元件为 ON 时，该对应的轴参数使用下述软元件：

Y300	Y304	Y310	Y314	Y0	Y1	属性
D8342	D8362	D8382	D8402	D8422	D8442	最高速度（Hz）
D8343	D8363	D8383	D8403	D8423	D8443	
D8347	D8367	D8387	D8407	D8427	D8447	基底速度（Hz）
D8348	D8368	D8388	D8408	D8428	D8448	加速时间（ms）
D8349	D8369	D8389	D8409	D8429	D8449	减速时间（ms）

5. 运行中更改参数

◆ 指令执行中，允许修改脉冲输出个数

更改前，需要置位特殊软元件，见下表，当脉冲更改有效软元件有效时，才能进行脉冲个数的更改。

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8350	M8370	M8390	M8410	M8430	M8450	加减速时间单独设定/脉冲更改有效

更改脉冲输出个数的值可以大于当前位置，也可以小于当前位置。

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8357	M8377	M8397	M8417	M8437	M8457	位置修改生效方式

注：减速所需要的位置公式： $1/2a \times (\text{当前速度}^2 - \text{基底速度}^2)$

- ① 当修改脉冲输出个数小于当前位置时，需将置位位置修改生效方式特殊软元件，否则会报警并按照原来设置的脉冲输出个数运行。
- ② 当修改脉冲输出个数大于当前位置时，如下图 a 所示，红色曲线为位置改小后的曲线，绿色曲线为位置改大之后的曲线。注意，这里的改大改小都是相对于目标位置的改大改小。
- ③ 后续过程中，多次修改，位置仍然是相对第一次进入时的绝对位置。

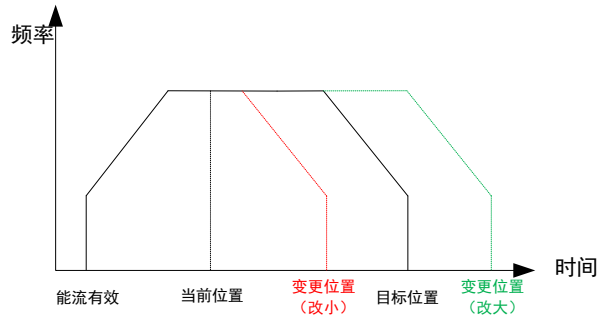


图 a: 修改脉冲输出个数大于当前位置

- ④ 当修改脉冲输出个数小于减速所需要的位置时，分为两种情况：

M8357 为 OFF：当前修改的位置将不生效，并报错“6713，定位位置偏小，不能减速”。

M8357 为 ON：如下图 b 所示，当修改后的位置小于当前位置时，修改后脉冲输出频率会逐渐减速到 0 再反向运行到修改后的位置。

注意：脉冲输出速度在减速段的斜率按照减速时间计算出来，反向加速段的加速度斜率按照加速时间计算出来。

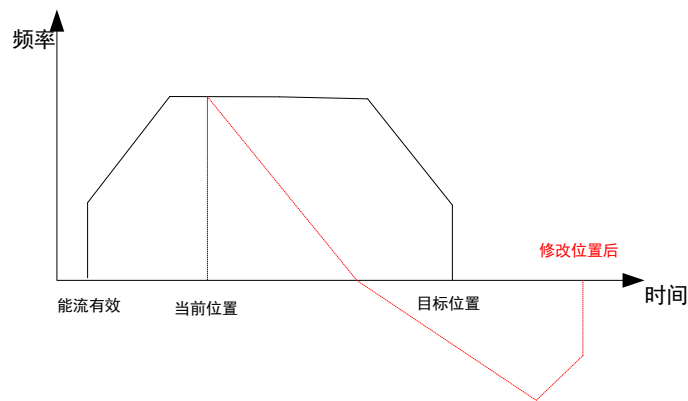
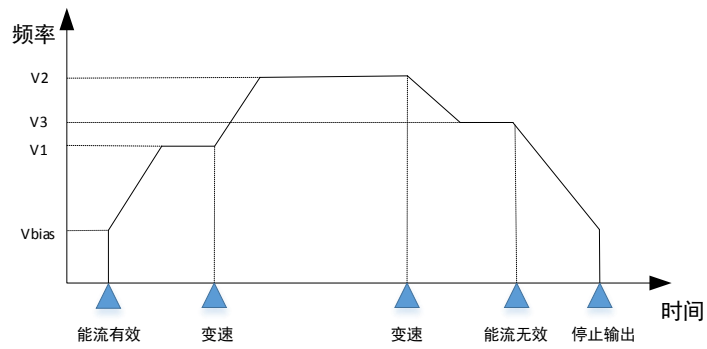


图 b: 修改脉冲输出个数小于当前位置

- ◆ 指令执行中，允许修改脉冲输出频率。

- ① 指令的输出过程中，允许修改脉冲输出的频率，更改的脉冲输出频率可大于也可以小于当前脉冲输出频率，范围为[10, 3000000]。
- ② 当频率大于 3000000 时，将限制到 3000000，当频率小于 10 时，将报错 6706，数据不合理。
- ③ 速度变化过程中，加速度与减速度斜率保持不变。
- ④ 如果同时修改脉冲个数与频率，则按照位置先生效，下一个扫描周期生效频率。



◆ 指令执行中，允许修改加速度。
(见补充说明文档)

◆ 支持 S 曲线功能

通过置位特殊软元件“S 曲线加减速使能标志位”来设置区分，如果标志位未置位，默认为梯形加减速。S 型曲线加减速见下表：

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8347	M8367	M8387	M8407	M8417	M8437	S 曲线加减速使能标志位
D8351	D8371	D8391	D8411	D8431	D8451	S 曲线时间

DRVA，DRVI 支持 S 曲线加减速，故在同等机械稳定性条件下可以让提升目标速度，因此可以缩短定位时间，提升加工效率。

S 曲线时间范围：[2~499].---需要扩大范围，请联系厂家。

实际加减速时间=T 型加减速+S 曲线时间。

5.2.8 ZRN 原点回归

1. 指令形式

启动后，加速到设定的回归输出频率，执行机构向原点（DOG）移动，检测到 DOG 信号后减速到爬行速度，DOG 信号为 OFF 时，停止输出。

ZRN S1 S2 S3 D					原点回归	指令执行	
S1	回归频率		设定的原点回归频率（Hz）			16 位指令(9step) ZRN 连续执行	32 位指令（17step） DZRN 连续执行
S2	爬行频率		设定的爬行频率（Hz）				
S3	DOG 信号		指定的原点输入信号（DOG）				
D	输出端口		高速脉冲输出端口				

2. 操作数

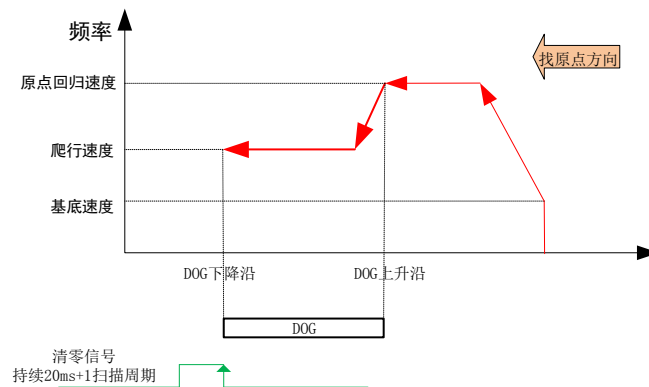
操作数	位软元件							字软元件															
	系统 用户							系统 用户					位数指定					变址			常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
S3	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E

注：表中灰底的软元件，表示可以支持的软元件。

- ①该指令是控制器与伺服驱动器配合工作时，以指定的脉冲速度从指定的端口输出，使得执行机构向原点（DOG）移动，直到遇到原点信号满足条件位置。
- ②S1 为原点回归动作频率，单位为 Hz。对于 16 位指令（ZRN），设定范围为 10~32767Hz；对于 32 位指令（DZRN），设定范围为 10~3,000,000Hz。
- ③S2 为原点信号变 ON 后的爬行速度，设定范围为 10~32767Hz。
- ④S3 为原点信号（DOG）输入，虽然 X/Y/M/S 信号都可以，但是只有 X 信号的及时性最好。
- ⑤D 为脉冲输出端口，可以指定 Y300/Y304/Y310/Y314/Y0/Y1。

3. 脉冲输出

- ①上电和初次运行时，一般执行原点回归指令 ZRN，以事先将机械动作的原点位置的数据写入，如果位置信息具有掉电保存功能，该指令并不需要每次上电都运行；指令执行过程中，仅能负方向运动，所以回原点的动作必须在 DOG 信号前端进行，往负向回原点。
- ②脉冲输出如下图所示，脉冲频率从基底速度开始逐渐减速到原点回归速度，驱动伺服往负方向运动，当遇到 DOG 信号的上升沿时，开始减速到爬行速度继续往负方向运动，当遇到 DOG 信号的下降沿时，认为找到原点，立即停止脉冲输出。



③当指令能流为 OFF 时，将立即停止，当能流由 OFF->ON 时，脉冲输出重新开始。

④指令执行完毕时，M8029 标志被置为 ON。

4. 软元件

下面对脉冲输出相关的软元件的作用做详细说明。

◆ 脉冲输出计数

Y300	Y304	Y310	Y314	Y0	Y1	属性
D8340	D8360	D8380	D8400	D8420	D8440	脉冲输出计数值 (PLS)
D8341	D8361	D8381	D8401	D8421	D8441	

脉冲输出计数值记录了端口输出的脉冲个数。该元件具有 D 软元件的特性，可以通过相关指令清零，也可以掉电清零。控制器状态由 RUN->OFF 时，该元件不会被清零。

◆ 脉冲输出中

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8340	M8360	M8380	M8400	M8420	M8440	脉冲输出中

◆ 脉冲输出停止

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8349	M8369	M8389	M8409	M8429	M8449	脉冲输出停止

当脉冲输出停止标志位被置位时脉冲输出停止，即使能流有效，脉冲也不会输出。

◆ 无脉冲输出完成及脉冲输出完成中断

◆ 加减速时间单独设定/脉冲更改有效

若想各输出轴具有不同的加减速时间，或者想在运行的过程中更改脉冲个数，那么可通过设定下述软元件：

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8350	M8370	M8390	M8410	M8430	M8450	加减速时间单独设定/脉冲更改有效

注意：该软元件是“加减速单独设定”和“脉冲更改有效”共用的软元件。

当上述“加减速时间单独设定”软元件为 OFF 时，对应轴参数使用如下软元件：

Y300	Y304	Y310	Y314	Y0	Y1	属性
D8500/D8501						最高速度 (Hz)

D8502	基底速度 (Hz)
D8503	加减速时间 (ms)

当某一轴“加减速时间单独设定”软元件为 ON 时, 该对应的轴参数使用下述软元件:

Y300	Y304	Y310	Y314	Y0	Y1	属性
D8342	D8362	D8382	D8402	D8422	D8442	最高速度 (Hz)
D8343	D8363	D8383	D8403	D8423	D8443	
D8347	D8367	D8387	D8407	D8427	D8447	基底速度 (Hz)
D8348	D8368	D8388	D8408	D8428	D8448	加速时间 (ms)
D8349	D8369	D8389	D8409	D8429	D8449	减速时间 (ms)

◆ 清零信号输出

可以通过置为特殊软元件“清零信号输出有效标志位”输出脉冲清零信号, 该信号可接入伺服的脉冲清除端口, 来清除伺服脉冲偏差, 使得伺服能够准确的停在 DOG 的下降沿处。清零输出信号见下表:

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8341	M8361	M8381	M8401	M8421	M8441	DSZR/ZRN 等清零信号输出有效标志位

清零信号可通过特殊寄存器指定, 只能是 Y 输出端口, 例如 D8350 指定 Y300 的清零输出端口, 当 D8350 指定为 5 时, 那么脉冲清零输出端口为 Y5, 详见下表:

Y300	Y304	Y310	Y314	Y0	Y1	属性
D8350	D8370	D8390	D8410	D8430	D8450	清零软元件编号 (十进制)
5 (默认)	6 (默认)	7 (默认)	8 (默认)	9 (默认)	10 (默认)	指定输出端口默认值

注意: 这里的清零软元件的值是十进制值, 而 Y 端口是 8 进制定义的, 所以在设定时需要进行转换, 例如 D8430 设定为 9, 那么十进制的 9 等于八进制的 11, 所以对应为 Y11 号端口。

◆ 清零逻辑位置与编码器位置

回零完成后, 系统将等待一个延迟时间 SD65, 自动清除当前逻辑位置(D8340,D8341)与编码器反馈位置(SD56, SD57)。

Y300	Y304	Y310	Y314	Y0	Y1	属性
SD65	SD165	SD265	SD765	SD865	SD965	回零延时时间(默认 20ms)
SD56	SD156	SD256	SD756	SD856	SD956	编码器反馈位置 (PLS)
SD57	SD157	SD257	SD757	SD857	SD957	

注: SD65 设置过小, 由于伺服的滞后性, 可能导致 SD56 存在残余滞留量。

5.2.9 DSZRDOG 搜索原点回归（开发中）

1. 指令形式

启动后，加速到设定的回归输出频率，执行机构向原点（DOG）移动，检测到 DOG 信号后减速到爬行速度，检测到零点信号从 OFF 到 ON，停止输出。

DSZR S1 S2 D1 D2					DOG 搜索原点回归	指令执行	
S1	DOG 信号		指定的原点输入信号（DOG）			16 位指令(9step) DSZR 连续执行	32 位指令（17step） DDSZR 连续执行
S2	零点信号		指定的零点输入信号				
D1	输出端口		高速脉冲输出端口				
D2	输出方向		脉冲运行方向端口或位变量				

2. 操作数

操作数	位软元件							字软元件															
	系统 用户							系统 用户				位数指定					变址			常数		实数	
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
D1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
D2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E

注：表中灰底的软元件，表示可以支持的软元件。

- ① 指令是控制器与伺服驱动器配合工作时，以特殊寄存器指定的脉冲速度从指定的端口输出脉冲，使得执行机构按照预先设定的动作原点移动，运行中近点信号（DOG）ON 到 OFF 后，检测到零点信号从 OFF 到 ON，则立即停止脉冲输出。
- ② 回归运动过程中输出旋转方向信号，回归结束输出清零信号。
- ③ 在设定有正反转限位的系统中，可以通过启用 DOG 搜索模式回归原点；在未设定正反转限位和不使用正反转限位进行原点回归的系统中，可以通过指定原点回归方向进行原点回归。其中：
S1 为近点信号（DOG）输入。虽然 X/Y/M/S 信号都可以，但只有 X 信号的及时性最好；S2 为零点信号输入。表示动作原点的确切位置，只能指定 X 信号；
D1 为脉冲输出端口。可指定 Y300/Y304/Y310/Y314/Y0/Y1；
D2 为旋转方向输出端口。ON：正转（脉冲输出使当前值增加）；OFF：反转（脉冲输出使当前值减少）。

3. 软元件

下面对脉冲输出相关的软元件的作用做详细说明。

◆ 特殊 D 元件定义：

Y300	Y304	Y310	Y314	Y0	Y1	属性
D8340	D8360	D8380	D8400	D8420	D8440	脉冲输出计数值（PLS）
D8341	D8361	D8381	D8401	D8421	D8441	
D8342	D8362	D8382	D8402	D8422	D8442	最高速度（HZ）[默认 3000000]
D8343	D8363	D8383	D8403	D8423	D8443	
D8344	D8364	D8384	D8404	D8424	D8444	原点回归速度（HZ）[默认 50000]
D8345	D8365	D8385	D8405	D8425	D8445	
D8346	D8366	D8386	D8406	D8426	D8446	爬行速度（HZ）[默认 2000]

D8347	D8367	D8387	D8407	D8427	D8447	基底速度 (HZ) [默认 500]
D8348	D8368	D8388	D8408	D8428	D8448	加速时间 (ms) [默认 100]
D8349	D8369	D8389	D8409	D8429	D8449	减速时间 (ms) [默认 100]
D8350	D8370	D8390	D8410	D8430	D8450	清零软元件编号

◆ 特殊 M 元件定义：

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8343	M8363	M8383	M8403	M8423	M8443	正转极限
M8344	M8364	M8384	M8404	M8424	M8444	反转极限
M8345	M8365	M8385	M8405	M8425	M8445	近点信号逻辑反转 ^[1]
M8346	M8366	M8386	M8406	M8426	M8446	零点信号逻辑反转 ^[1]
M8349	M8369	M8389	M8409	M8429	M8449	脉冲输出停止标志 ^[1]
M8350	M8370	M8390	M8410	M8430	M8450	加减速时间单独设置 和脉冲更改有效
M8351	M8371	M8391	M8411	M8431	M8451	端口的输出初始化标志
M8354	M8374	M8394	M8414	M8434	M8454	DSZR指令执行异常结束标志位

[1]:RUN→STOP, 清除。

最高速度、原点回归速度、爬行速度、基底速度，请遵循：

基底速度≤原点回归速度≤最高速度；

基底速度≤爬行速度≤最高速度；

参数设定范围：

原点回归速度，范围是 10~3100,000Hz；

爬行速度，范围是 10~32,767Hz；

基底速度，范围是 10~32,767Hz；

指定原点回归方向：按照原点回归方向指定标志位的 ON/OFF 来指定回归原点方向。由基底速度加速到原点回归速度，向原点回归方向指定标志位指定的方向移动。感应到 S1 指定的近点信号（DOG）为 ON，开始减速，直到爬行速度。S1 指定的近点信号（DOG）从 ON 到 OFF 后，如果检测到 S2 指定的零点信号由 OFF 到 ON，立即停止脉冲的输出。

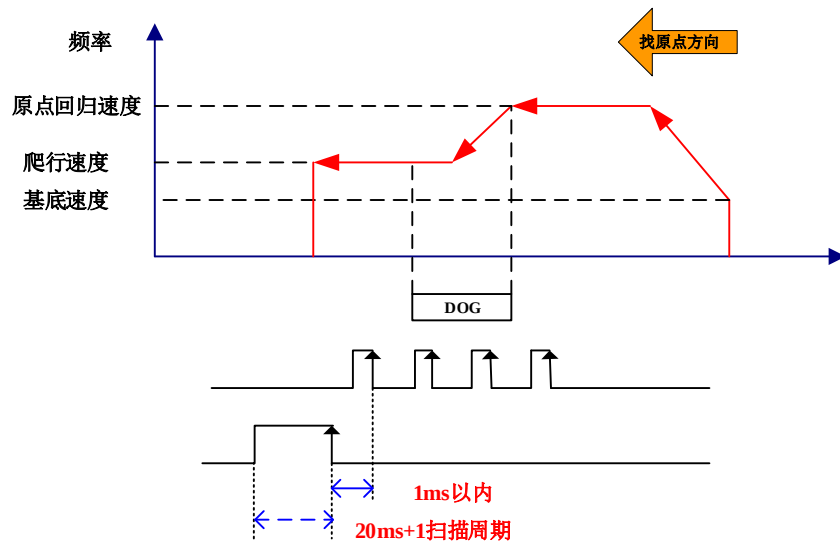
如果近点信号和零点信号指定了相同的输入，那么和 ZRN 指令一样，不使用零点信号，在近点信号从 ON 到 OFF 后，立即停止脉冲的输出；如果近点信号和零点信号指定了相同的输入，那么如果有逻辑反转标志位为 ON，以近点信号的逻辑为准。

清零信号输出功能有效（ON）时，在脉冲输出停止后（1ms 以内），清零信号在[20ms+1 个运算周期]的时间内保持为 ON。指令执行结束标志位（M8029）为 ON，结束原点回归动作。

这是近点信号逻辑反转标志位和零点信号逻辑反转标志位为 OFF 的说明。如果该逻辑标志位为 ON，对应的近点和零点信号的 ON 改成 OFF，OFF 改成 ON。

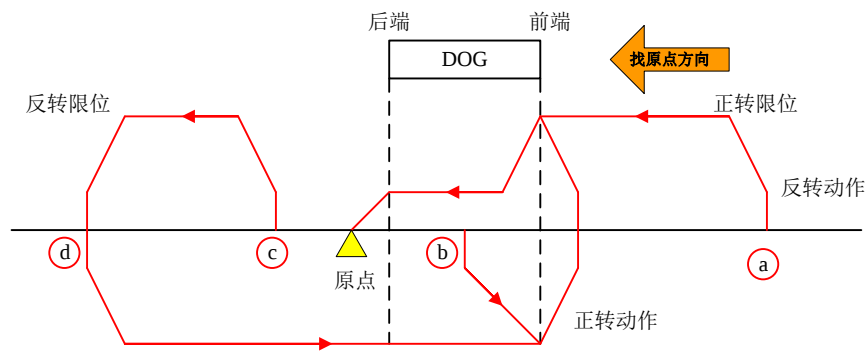
当指令能流为 OFF 时，将立即停止；当能流由 OFF→ON 时，脉冲输出处理重新开始；指令执行完毕，M8029 标志置为 ON；

脉冲输出示意图，如下：



◆ DOG 搜索功能

设计有正转限位、反转限位时，执行使用了 DOG 搜索功能的原点回归。此时，因原点回归的开始位置不同，原定回归动作也各不相同。



a) 开始位置在通过DOG前的时候，包括正转纤维1为ON的时候：

- ① 通过执行原点回归用指令，开始原点回归动作。
- ② 以原点回归速度，向原点回归方向开始移动。
- ③ 一旦检测出 DOG 的前端，就开始减速到爬行速度。
- ④ 检测出 DOG 的后端后，在检测出第一个零点信号时停止。

b) 开始位置在DOG内的时候：

- ① 通过执行原点回归用指令，开始原点回归动作。
- ② 以原点回归速度，向与原点回归方向相反的方向开始移动。
- ③ 检测出 DOG 的前端后减速停止。（离开 DOG）
- ④ 以原点回归速度，向原点回归方向开始移动。（再次进入 DOG）
- ⑤ 一旦检测出 DOG 的前端，就开始减速到爬行速度。

⑥ 检测出 DOG 的后端后，在检测出第一个零点信号时停止。

c) 开始位置在近点信号OFF（通过DOG后）的时候：

- ① 通过执行原点回归用指令，开始原点回归动作。
- ② 以原点回归速度，向原点回归方向开始移动。
- ③ 检测出反转限位 1（反转极限）时减速停止。
- ④ 以原点回归速度，向与原点回归方向相反的方向开始移动。
- ⑤ 检测出 DOG 的前端后减速停止。（检出（离开）DOG）
- ⑥ 以原点回归速度，向原点回归方向开始移动。（再次进入 DOG）
- ⑦ 一旦检测出 DOG 的前端，就开始减速到爬行速度。
- ⑧ 检测出 DOG 的后端后，在检测出第一个零点信号时停止。

d) 原点回归方向的限位开关（反转限位1）为ON的时候：

- ① 点回归用指令，开始原点回归动作。
- ② 以原点回归速度，向与原点回归方向相反的方向开始移动。
- ③ 检测出 DOG 的前端后减速停止。（检出（离开）DOG）
- ④ 以原点回归速度，向原点回归方向开始移动。（再次进入 DOG）
- ⑤ 一旦检测出 DOG 的前端，就开始减速到爬行速度。
- ⑥ 检测出 DOG 的后端后，在检测出第一个零点信号时停止。

注意：设计近点信号（DOG）时，要考虑有足够的 ON 时间减速到爬行速度；请使爬行速度尽可能的慢，因为其停止没有减速，爬行速度过快，可能会导致位置偏移。

◆ 脉冲输出中

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8340	M8360	M8380	M8400	M8420	M8440	脉冲输出中

◆ 脉冲输出停止

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8349	M8369	M8389	M8409	M8429	M8449	脉冲输出停止

当脉冲输出停止标志位被置位时脉冲输出停止，即使能流有效，脉冲也不会输出。

◆ 无脉冲输出完成及脉冲输出完成中断

◆ 加减速时间单独设定/脉冲更改有效

若想各输出轴具有不同的加减速时间，或者想在运行的过程中更改脉冲个数，那么可通过设定下述软元件：

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8350	M8370	M8390	M8410	M8430	M8450	加减速时间单独设定/脉冲更改有效

注意：该软元件是“加减速单独设定”和“脉冲更改有效”共用的软元件。

当上述“加减速时间单独设定”软元件为 OFF 时，对应轴参数使用如下软元件：

Y300	Y304	Y310	Y314	Y0	Y1	属性
D8500/D8501						最高速度 (Hz)
D8502						基底速度 (Hz)
D8503						加减速时间 (ms)

当某一轴“加减速时间单独设定”软元件为 ON 时，该对应的轴参数使用下述软元件：

Y300	Y304	Y310	Y314	Y0	Y1	属性
D8342	D8362	D8382	D8402	D8422	D8442	最高速度 (Hz)
D8343	D8363	D8383	D8403	D8423	D8443	
D8347	D8367	D8387	D8407	D8427	D8447	基底速度 (Hz)
D8348	D8368	D8388	D8408	D8428	D8448	加速时间 (ms)
D8349	D8369	D8389	D8409	D8429	D8449	减速时间 (ms)

◆ 清零信号输出

可以通过置为特殊软元件“清零信号输出有效标志位”输出脉冲清零信号，该信号可接入伺服的脉冲清除端口，来清除伺服脉冲偏差，使得伺服能够准确的停在 DOG 的下降沿处。清零输出信号见下表：

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8341	M8361	M8381	M8401	M8421	M8441	DSZR/ZRN 等清零信号输出有效标志位

清零信号可通过特殊寄存器指定，只能是 Y 输出口，例如 D8350 指定 Y300 的清零输出口，当 D8350 指定为 5 时，那么脉冲清零输出口为 Y5，详见下表：

Y300	Y304	Y310	Y314	Y0	Y1	属性
D8350	D8370	D8390	D8410	D8430	D8450	清零软元件编号（十进制）
5（默认）	6（默认）	7（默认）	8（默认）	9（默认）	10（默认）	指定输出口默认值

注意：这里的清零软元件的值是十进制值，而 Y 端口是 8 进制定义的，所以在设定时需要进行转换，例如 D8430 设定为 9，那么十进制的 9 等于八进制的 11，所以对应为 Y11 号端口。

◆ 信号逻辑反转

OFF：正逻辑（输入为 ON 时，近点/零点信号为 ON）；

ON：负逻辑（输入为 OFF 时，近点/零点信号为 ON）。

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8345	M8365	M8385	M8405	M8425	M8445	近点信号逻辑反转
M8346	M8366	M8386	M8406	M8426	M8446	零点信号逻辑反转

5.2.10 DVIT 中断定位（未开发）

1. 指令形式

启动后, 加速到设定的速度输出频率, 当检测到中断输入信号后, 立即加速或减速到位置段输出频率, 并输出设定的脉冲个数。

DVIT S1 S2 S3 D1 D2 S4							中断定位	指令执行	
S1	脉冲个数	设定的中断后位置脉冲输出个数					16 位指令 (13step) DSZR 连续执行	32 位指令 (25step) DDSZR 连续执行	
S2	输出频率 1	设定的速度段脉冲输出频率							
S3	输出频率 2	设定的中断后位置段脉冲输出频率							
D1	输出端口	高速脉冲输出端口							
D2	输出方向	脉冲运行方向端口或位变量							
S4	中断输入	中断输入信号端口（范围 X0-X4）							

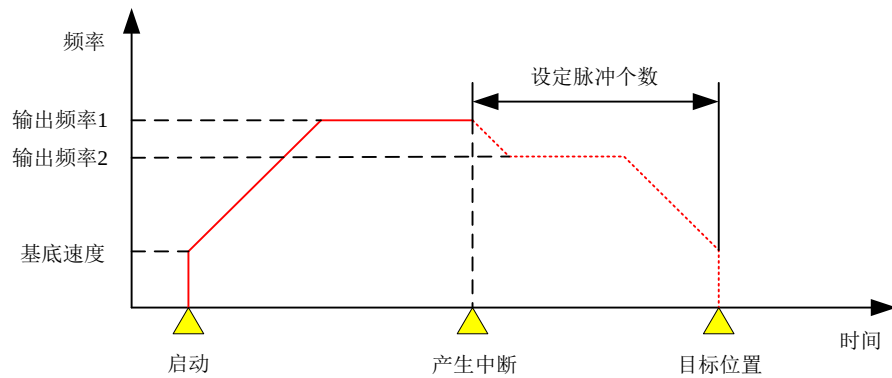
2. 操作数

操作数	位软元件							字软元件															
	系统 用户							系统 用户					位数指定					变址		常数		实数	
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
S3	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
D1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
D2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
S4	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E

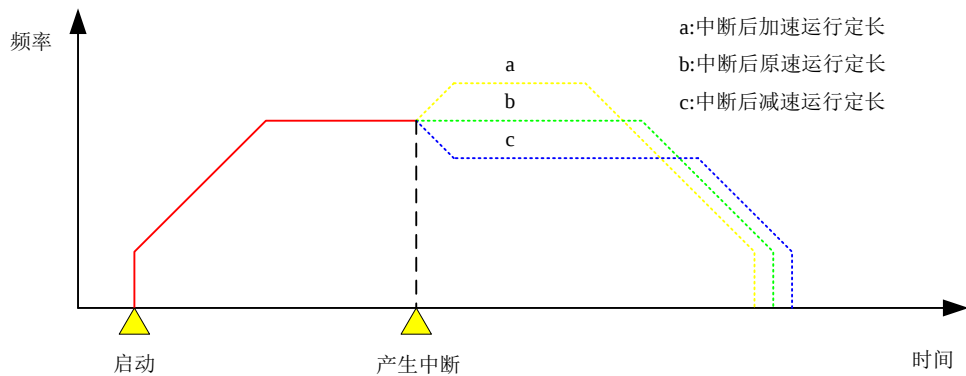
注: 表中灰底的软元件, 表示可以支持的软元件。

- ① 指令是控制器与伺服驱动器配合工作时, 以指定的脉冲速度从指定的端口输出脉冲, 运行中检测到中断信号后, 按照中断后位置段脉冲输出频率输出给定的脉冲个数, 使执行机构在当前位置的基础上作偏移的运动。
- ② S1 为脉冲输出个数, 单位为 PLS, 对于 16 位指令 (DVIT), 设定范围为-32768~32767; 对于 32 位指令 (DDVIT), 设定范围为-2, 147, 483, 648~2, 147, 483, 647;
- ③ S2 为中断前速度段脉冲输出频率, 单位为 Hz, 对于 16 位指令 (DVIT), 设定范围为 10~32767Hz; 对于 32 位指令 (DDVIT), 设定范围为 10~3, 000, 000Hz; 在指令执行的过程中可以修改 S2 的值。
- ④ S3 为中断后位置段脉冲输出频率, 单位为 Hz, 对于 16 位指令 (DVIT), 设定范围为 10~32767Hz; 对于 32 位指令 (DDVIT), 设定范围为 10~3, 000, 000Hz; 在指令执行的过程中可以修改 S2 的值。
- ⑤ D1 为脉冲输出端口, 可以指定 Y300/Y304/Y310/Y314/Y0/Y1。
- ⑥ D2 为运行方向输出端口或位变量, 该位的状态由控制器自身脉冲输出状态决定, 当脉冲输出为正向运行时, D2 的状态为 ON, 当脉冲输出为反向运行时, D2 的状态为 OFF。
- ⑦ S4 为中断信号输入端口, 可以指定 X0-X4。

3. 脉冲输出



中断后输出频率变化情况：



4. 软元件

下面对脉冲输出相关的软元件的作用做详细说明。

◆ 脉冲输出计数

Y300	Y304	Y310	Y314	Y0	Y1	属性
D8340	D8360	D8380	D8400	D8420	D8440	脉冲输出计数值 (PLS)
D8341	D8361	D8381	D8401	D8421	D8441	

脉冲输出计数值记录了端口输出的脉冲个数。该元件具有 D 软元件的特性，可以通过相关指令清零，也可以掉电清零。控制器状态由 RUN→OFF 时，该元件不会被清零。

◆ 脉冲输出中

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8340	M8360	M8380	M8400	M8420	M8440	脉冲输出中

如下图所示，当 Y304 脉冲输出时，M8360 被置位，当脉冲输出完成后，M8360 被自动复位。

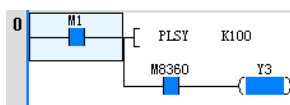


图 a：脉冲输出中

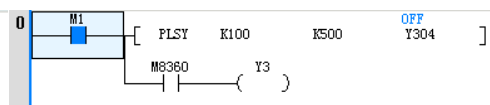


图 b：脉冲输出完成

◆ 脉冲输出停止

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8349	M8369	M8389	M8409	M8429	M8449	脉冲输出停止

当脉冲输出停止标志位被置位时脉冲输出停止，即使能流有效，脉冲也不会输出。软元件作用效果见下图所示，在软元件置位前，脉冲正常输出，当软元件置位后，即使能流有效，脉冲也会停止输出。

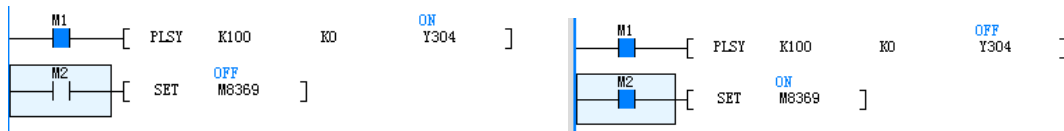


图 a: 脉冲输出停止置位前

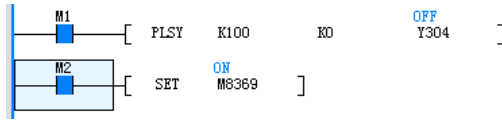


图 b: 脉冲输出停止置位后

◆ 脉冲输出完成

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8029						脉冲输出完成

当本条脉冲输出完成后，相应的 M8029 会被置位，但本条脉冲输出完成并不影响别的脉冲指令的脉冲输出完成标志位，如下图中的，第一条脉冲指令执行完成后，M10 被置位，但是 M11 和 M12 均未被置位。

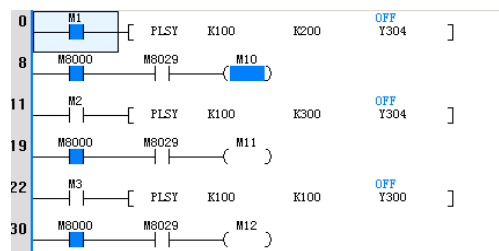


图 a: M8029 软元件动作

◆ 脉冲输出完成中断

若想脉冲输出完成时进入中断，那么针对不同的 Y 口可开启不同的输出完成中断使能软元件。见下表，如想在 Y300 脉冲输出完成后进入中断，那么需要将 M8352 置为 1。

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8352	M8372	M8392	M8412	M8432	M8452	输出完成中断使能

◆ 加减速时间单独设定/脉冲更改有效

若想各输出轴具有不同的加减速时间，或者想在运行的过程中更改脉冲个数，那么可通过设定下述软元件：

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8350	M8370	M8390	M8410	M8430	M8450	加减速时间单独设定/脉冲更改有效

注意：该软元件是“加减速单独设定”和“脉冲更改有效”共用的软元件。

当上述“加减速时间单独设定”软元件为 OFF 时，对应轴参数使用如下软元件：

Y300	Y304	Y310	Y314	Y0	Y1	属性
D8500/D8501						最高速度 (Hz)
D8502						基底速度 (Hz)
D8503						加减速时间 (ms)

当某一轴“加减速时间单独设定”软元件为 ON 时，该对应的轴参数使用下述软元件：

Y300	Y304	Y310	Y314	Y0	Y1	属性
------	------	------	------	----	----	----

D8342	D8362	D8382	D8402	D8422	D8442	最高速度 (Hz)
D8343	D8363	D8383	D8403	D8423	D8443	
D8347	D8367	D8387	D8407	D8427	D8447	基底速度 (Hz)
D8348	D8368	D8388	D8408	D8428	D8448	加速时间 (ms)
D8349	D8369	D8389	D8409	D8429	D8449	减速时间 (ms)

5. 运行中修改频率

在运行中更改脉冲输出频率，如下图 a/b 所示。从 V1 更改到 V2 或者从 V2 更改到 V3，更改速度时，速度不会突变，而是按照设定的加减速时间来加速或者减速到更改后的速度。在图 a 中，速度从 V1 加速到 V2 的加速度跟速度从基底速度加速到 V1 的加速度相同；速度从 V2 减速到 V3 的加速度跟 V3 减速到基底速度的加速度相同。

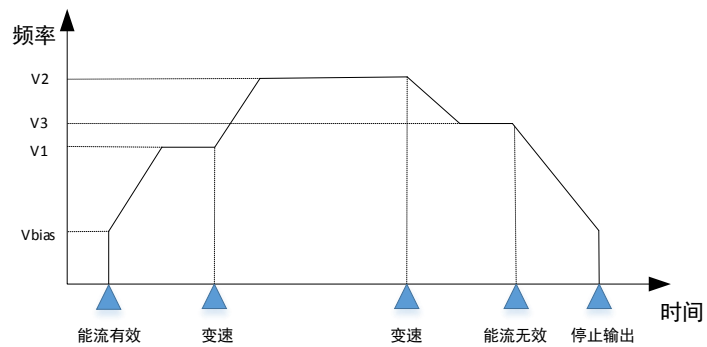


图 a: PLSV2 脉冲输出变化示意图

在图 b 中，速度从基底速度加速到 V1 的加速度跟速度从 0 加速到 V2 的加速度以及速度从 0 加速到 V3 的加速度相同；速度从 V1 减速到 0 的加速度跟速度从 V2 加速到 0 的加速度以及速度从 V3 减速到基底速度的加速度均相同。

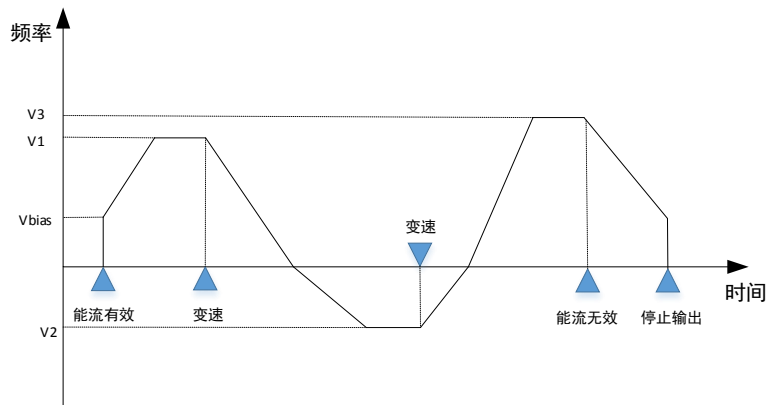


图 b: PLSV2 脉冲输出变化示意图（反向变速）

5.2.11 DPIT 最大定长的中断定位指令

1. 指令形式

启动后, 加速到设定的速度输出频率, 当检测到中断输入信号后, 立即加速或减速到位置段输出频率, 并输出设定的脉冲个数; 当没有检测到中断输入信号, 则输出设定的最大脉冲数。

DPIT S1 S2 D1 D2 S3						最大定长的中断 定位指令	指令执行	
S1	最大脉冲数	S1: 设定的中断后位置脉冲输出个数 16 位指令 S1+1,32 位指令 S1+2: 设定中断后位置段脉冲输出个数【1】				16 位指令 (11step) DSZR 连续执行	32 位指令 (21step) DDSZR 连续执行	
S2	输出频率	S2: 设定的速度段脉冲输出频率 16 位指令 S2+1,32 位指令 S2+2: 设定中断后位置段脉冲输出频率【2】						
D1	输出端口	高速脉冲输出端口						
D2	输出方向	脉冲运行方向端口或位变量						
S3	中断输入	中断输入信号端口（范围 X0-X4）						

注: 【1】当指令位 16 位时: 设定中断后位置段脉冲输出个数地址为 S1+1;

当指令位 32 位时: 设定中断后位置段脉冲输出个数地址为 S1+2。

【2】当指令位 16 位时: 设定中断后脉冲输出频率地址为 S2+1;

当指令位 32 位时: 设定中断后脉冲输出频率地址为 S2+2。

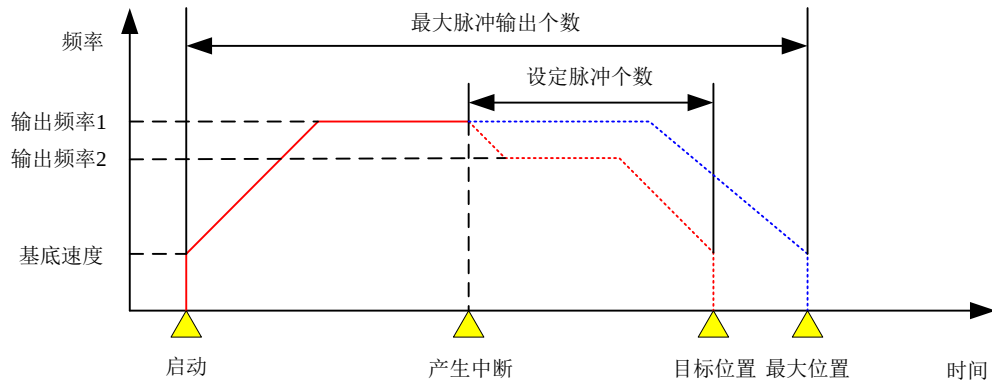
2. 操作数

操作数	位软元件							字软元件															
	系统 用户							系统 用户				位数指定						变址		常数		实数	
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
D1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
D2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
S3	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E

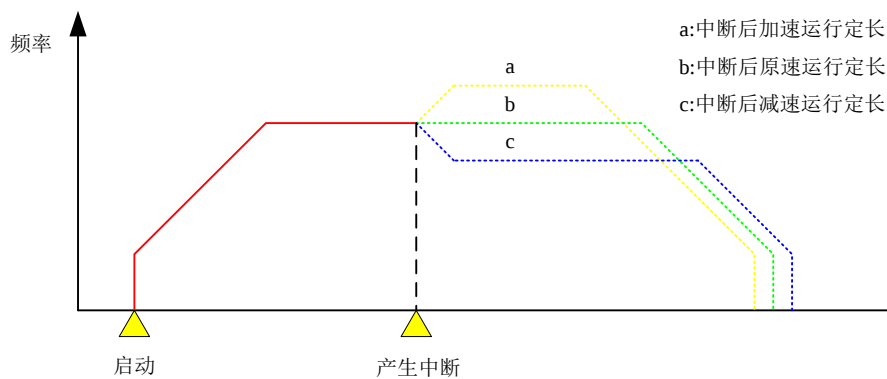
注: 表中灰底的软元件, 表示可以支持的软元件。

- ① 指令是控制器与伺服驱动器配合工作时, 以指定的脉冲速度从指定的端口输出脉冲, 运行中检测到中断信号后, 按照中断后位置段脉冲输出频率输出给定的脉冲个数, 使执行机构在当前位置的基础上作偏移的运动; 如果运行过程中未检测到中断信号, 则输出设定的最大脉冲数。
- ② S1 为脉冲输出个数, 单位为 PLS, 对于 16 位指令 (DPIT), 设定范围为-32768~32767; 对于 32 位指令 (DDPIT), 设定范围为-2, 147, 483, 648~2, 147, 483, 647; 其正负决定脉冲输出方向。
- ③ S2 为中断前和中断后脉冲输出频率, 单位为 Hz, 对于 16 位指令 (DPIT), 设定范围为 10~32767Hz; 对于 32 位指令 (DDPIT), 设定范围为 10~3, 000, 000Hz; 在指令执行的过程中可以修改 S2 的值。
- ④ D1 为脉冲输出端口, 可以指定 Y300/Y304/Y310/Y314/Y0/Y1。
- ⑤ D2 为运行方向输出端口或位变量, 该位的状态由控制器自身脉冲输出状态决定, 当脉冲输出为正向运行时, D2 的状态为 ON, 当脉冲输出为反向运行时, D2 的状态为 OFF。
- ⑥ S3 为中断信号输入端口, 可以指定 X0-X4。

3. 脉冲输出



中断后输出频率变化情况：



4. 软元件

下面对脉冲输出相关的软元件的作用做详细说明。

◆ 脉冲输出计数

Y300	Y304	Y310	Y314	Y0	Y1	属性
D8340	D8360	D8380	D8400	D8420	D8440	脉冲输出计数值 (PLS)
D8341	D8361	D8381	D8401	D8421	D8441	

脉冲输出计数值记录了端口输出的脉冲个数。该元件具有 D 软元件的特性，可以通过相关指令清零，也可以掉电清零。控制器状态由 RUN→OFF 时，该元件不会被清零。

◆ 脉冲输出中

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8340	M8360	M8380	M8400	M8420	M8440	脉冲输出中

如下图所示，当 Y304 脉冲输出时，M8360 被置位，当脉冲输出完成后，M8360 被自动复位。

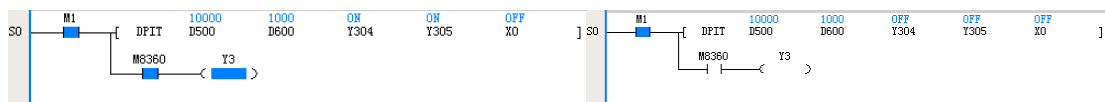


图 a：脉冲输出中

图 b：脉冲输出完成

◆ 脉冲输出停止

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8349	M8369	M8389	M8409	M8429	M8449	脉冲输出停止

当脉冲输出停止标志位被置位时脉冲输出停止，即使能流有效，脉冲也不会输出。软元件作用效果见下图所示，在软元件置位前，脉冲正常输出，当软元件置位后，即使能流有效，脉冲也会停止输出。

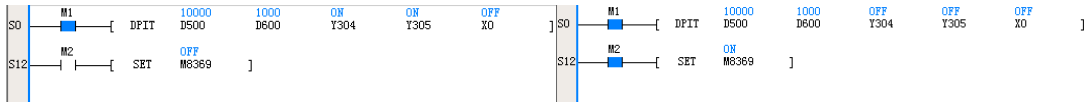


图 a: 脉冲输出停止置位前

图 b: 脉冲输出停止置位后

◆ 脉冲输出完成

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8029						脉冲输出完成

当本条脉冲输出完成后，相应的 M8029 会被置位，但本条脉冲输出完成并不影响别的脉冲指令的脉冲输出完成标志位，如下图中的，第一条脉冲指令执行完成后，M10 被置位，但是 M11 和 M12 均未被置位。

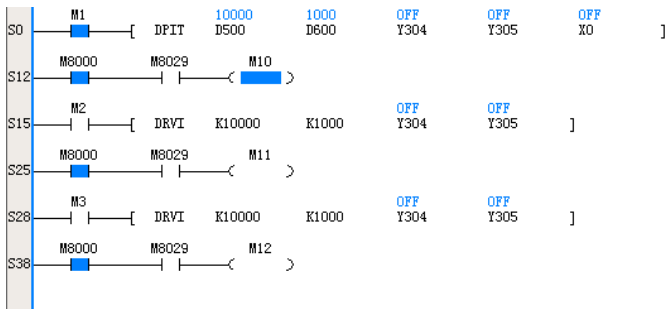


图 a: M8029 软元件动作

◆ 脉冲输出完成中断

若想脉冲输出完成时进入中断，那么针对不同的 Y 口可开启不同的输出完成中断使能软元件。见下表，如想在 Y300 脉冲输出完成后进入中断，那么需要将 M8352 置为 1。

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8352	M8372	M8392	M8412	M8432	M8452	输出完成中断使能

◆ 加减速时间单独设定/脉冲更改有效

若想各输出轴具有不同的加减速时间，或者想在运行的过程中更改脉冲个数，那么可通过设定下述软元件：

Y300	Y304	Y310	Y314	Y0	Y1	属性
M8350	M8370	M8390	M8410	M8430	M8450	加减速时间单独设定/脉冲更改有效

注意：该软元件是“加减速单独设定”和“脉冲更改有效”共用的软元件。

当上述“加减速时间单独设定”软元件为 OFF 时，对应轴参数使用如下软元件：

Y300	Y304	Y310	Y314	Y0	Y1	属性
D8500/D8501						最高速度 (Hz)
D8502						基底速度 (Hz)
D8503						加减速时间 (ms)

当某一轴“加减速时间单独设定”软元件为 ON 时，该对应的轴参数使用下述软元件：

Y300	Y304	Y310	Y314	Y0	Y1	属性
D8342	D8362	D8382	D8402	D8422	D8442	最高速度 (Hz)
D8343	D8363	D8383	D8403	D8423	D8443	

D8347	D8367	D8387	D8407	D8427	D8447	基底速度 (Hz)
D8348	D8368	D8388	D8408	D8428	D8448	加速时间 (ms)
D8349	D8369	D8389	D8409	D8429	D8449	减速时间 (ms)

5. 运行中修改频率

在运行中更改脉冲输出频率，如下图 a/b 所示。从 V1 更改到 V2 或者从 V2 更改到 V3，更改速度时，速度不会突变，而是按照设定的加减速时间来加速或者减速到更改后的速度。在图 a 中，速度从 V1 加速到 V2 的加速度跟速度从基底速度加速到 V1 的加速度相同；速度从 V2 减速到 V3 的加速度跟 V3 减速到基底速度的加速度相同。

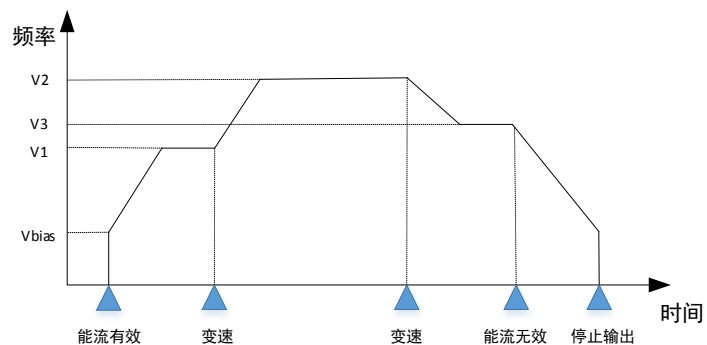


图 a: PLSV2 脉冲输出变化示意图

在图 b 中，速度从基底速度加速到 V1 的加速度跟速度从 0 加速到 V2 的加速度以及速度从 0 加速到 V3 的加速度相同；速度从 V1 减速到 0 的加速度跟速度从 V2 加速到 0 的加速度以及速度从 V3 减速到基底速度的加速度均相同。

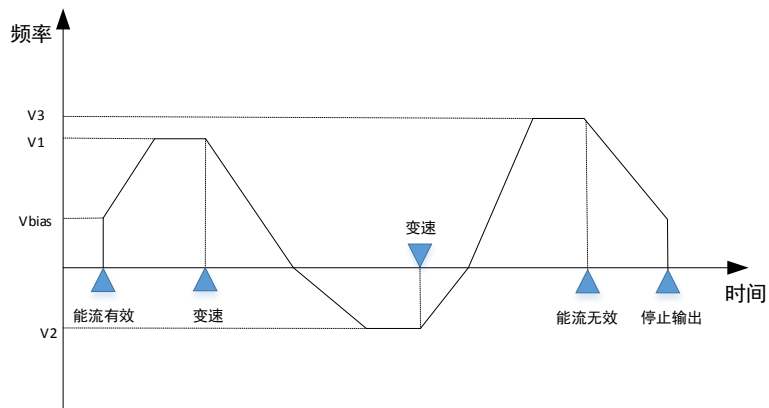


图 b: PLSV2 脉冲输出变化示意图（反向变速）

5.3 高速处理指令

5.3.1 PWM 脉宽调制输出指令

1.指令形式

按照设定的脉宽和周期输出调制方波。

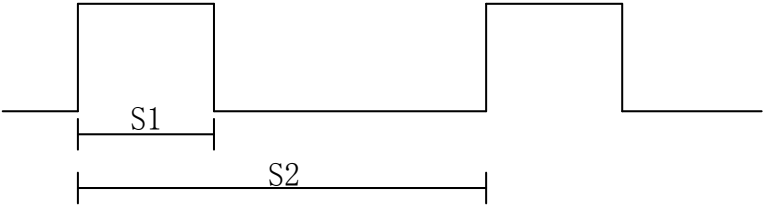
PWM S1 S2 D				脉宽调制输出指令	指令执行
S1	输出脉宽			设定的输出脉宽	16 位指令（7step） PWM 连续执行
S2	脉冲周期			设定的脉冲周期	
D	输出端口			高速脉冲输出端口	

2. 操作数

操作数	位软元件							字软元件															
	系统 用户							系统 用户					位数指定					变址		常数		实数	
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E

注：表中灰底的软元件，表示可以支持的软元件。

- ① S1 及 S2 的作用见下图所示。S1 为高电平持续时间，S2 为一个 PWM 周期的时间。S1 及 S2 之间要满足 $S1 \leq S2$ 。



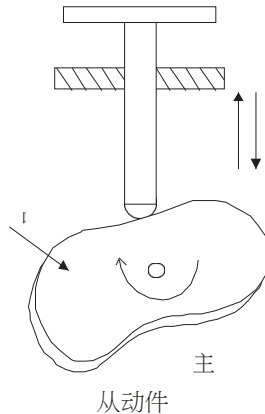
- ② S1 为设定的输出脉宽，必须满足 $S1 \leq S2$ ，设定范围为 0~32,767ms。
- ③ S2 为设定的输出周期，必须满足 $S1 \leq S2$ ，设定范围为 1~32,767ms。
- ④ D 为脉冲输出端口，可选择 Y0 或 Y1 作为 PWM 脉冲输出。
- ⑤ S1、S2 可以在 PWM 指令执行时更改。

第6章 电子凸轮

6.1 电子凸轮（E-CAM）简介

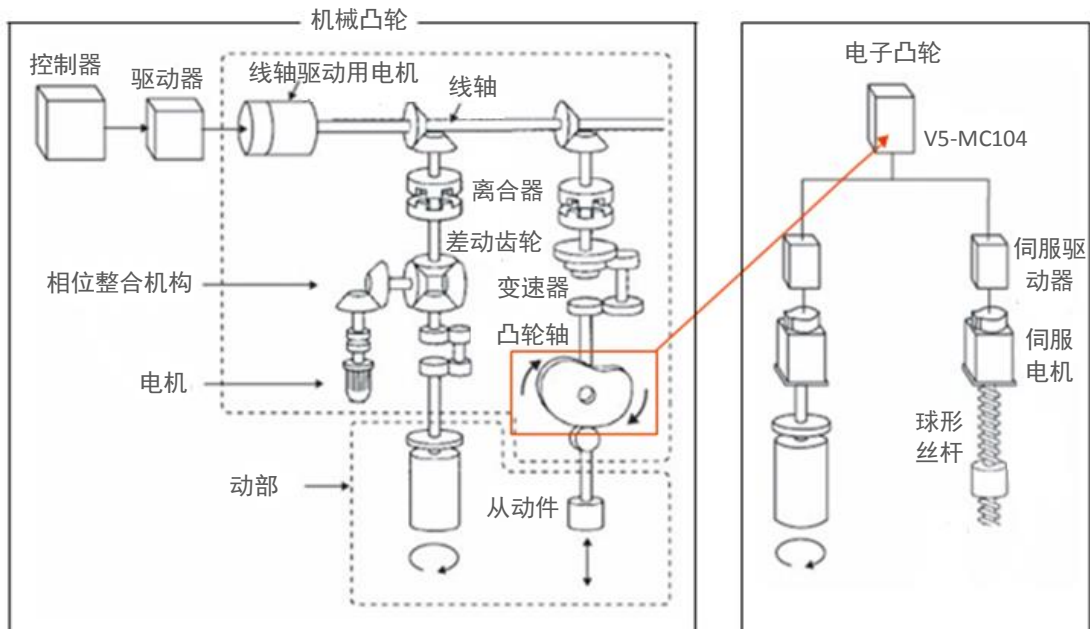
传统的机械凸轮机构，是由凸轮、从动件以及机架等三部份组成。

机械凸轮是一种不规则形状的机件，一般为等转速的输入件，可经由直接接触传递运动到从动件，使从动件依设定的规律运动。从动件为机械凸轮所驱动的被动物件，一般为产生不等速、不连续、不规则运动的输出件。机架则是用来支持机械凸轮与从动件的机件。



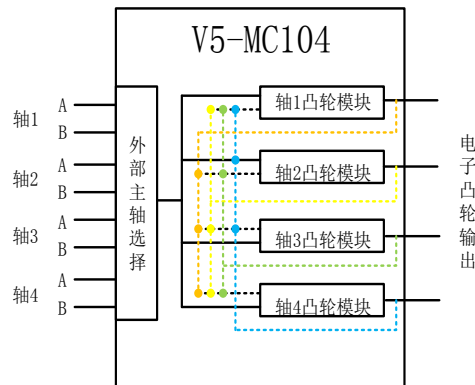
相对传统的方式，使用电子凸轮有着以下的好处：

- ◆ 较友善的用户接口
- ◆ 不同的产品需要不同的凸轮曲线，可利用软件修改电子凸轮表内电子凸轮数据，不用修改机构
- ◆ 可以有较高的加速
- ◆ 运行可以较为平顺



6.2 电子凸轮的实现方式

◆ 架构图



注：图中实线为主轴选择外部脉冲输入；图中虚线为主轴选择虚拟连接。

◆ 操作步骤

第一步	第二步	第三步
初始设定	主轴选择/凸轮表选择	启动/停止电子凸轮
①建立电子凸轮数据 ②端子设定 ③输入/出脉冲形式设定 ④输入/输出倍率 ⑤延时启动设置 ⑥电子凸轮起始位置设置	①选择实轴/虚拟轴 ②选择凸轮表	①电子凸轮启动 ②电子凸轮停止 ③电子凸轮强制停止

6.2.1 第一步：初始设定

6.2.1.1 建立电子凸轮数据

V5-MC104 提供了两种建立电子凸轮数据的方法，有其各自的优缺点，适用于不同的场合。建立凸轮数据的方法如下：

方法	适用场合
通过 VCAutoDesignsoft 的 CAM 图表生成电子凸轮数据	不需要实时更新电子凸轮数据点的场合
通过 CAMWR 指令将电子凸轮数据写入	需要实时更新电子凸轮数据点的场合

6.2.1.2 输入/输出脉冲形式设定

- 若从轴脉冲输入来源为虚拟轴，主轴脉冲输出形式可根据实际的需要来确定，从轴的脉冲输入形式默认跟随主轴脉冲输出形式，不需要用户配置。
- 从轴脉冲输入来源为实际轴，由于从轴的脉冲计数固定为 AB 正交输入，所以主轴的脉冲输出形式需要设置为 AB 正交形式（SDx61=1）。

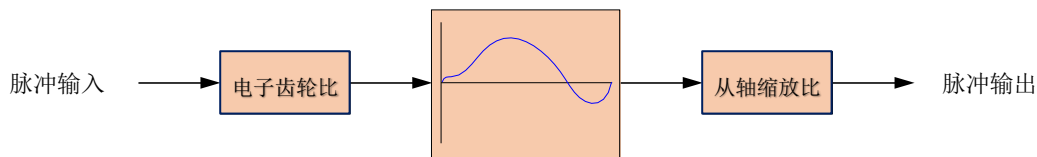
从轴脉冲来源	从轴脉冲输入形式	主轴脉冲输出形式
虚拟主轴	默认跟随虚拟主轴脉冲输出形式，不需要配置	根据实际需要选择 方向+脉冲（SDx61 ^{注1} =0） A/B 正交（SDx61=1） CW/CCW（SDx61=2）
实际主轴 ^{注2}	固定为 AB 正交输入	A/B 正交（SDx61=1）

注 1: SDx61 为脉冲输出形式选择软元件，轴 1 为 SD61，轴 2 为 SD161，轴 3 为 SD261，轴 4 为 SD761，Y0 为 SD861，Y1 为 SD961。

注 2: 实际主轴形式是指通过接线的方式，将主轴脉冲输出连接到从轴脉冲输入端口的一种形式。

6.2.1.3 输入/输出比例缩放

电子凸轮可以通过比例缩放功能实现凸轮表的比例缩放。



电子齿轮比和从轴缩放比使用的特殊元件如下表：

电子齿轮比/主轴缩放比例						从轴缩放比例					
轴 1	轴 2	轴 3	轴 4	Y0	Y1	轴 1	轴 2	轴 3	轴 4	Y0	Y1
SD44/ SD45	SD144/ SD145	SD244/ SD245	SD744/ SD745	SD844/ SD845	SD944/ SD945	SD73 /100	SD173 /100	SD173 /100	SD773 /100	SD873 /100	SD973 /100
主轴缩放比例						从轴缩放比例，SD 元件系统启动时默认为 100，即比例值为 1					

电子齿轮比和从轴缩放比值变化后，默认在下次启动凸轮的时候生效。如果需要在当前运行的凸轮中生效，需要置位凸轮表数据修改特殊 SM 元件，在当前运行的下一个凸轮周期即可生效，生效后凸轮表数据修改特殊 SM 元件自动复位。

凸轮表 数据修 改	轴 1	轴 2	轴 3	轴 4	Y0	Y1	说明
	SM83	SM183	SM283	SM783	SM883	SM983	数据修改生效后自动复位为 OFF

6.2.1.4 电子凸轮延时启动

电子凸轮/电子齿轮可以通过 SM 软元件来实现延时启动功能。

延时启动功能的特殊软元件如下表：

延时启动软元件						软元件状态	
轴 1	轴 2	轴 3	轴 4	Y0	Y1	ON	OFF
SM75	SM175	SM275	SM775	SM875	SM975	使能延时启动	禁止延时启动

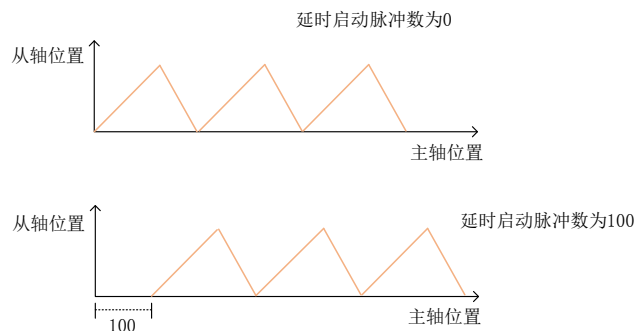
当延时启动功能开启后，凸轮使能后，等主轴运行了设定的位置后，从轴才开始运行。

从轴起始位置设置软元件：

延时启动脉冲数（32bit）						单位
轴 1	轴 2	轴 3	轴 4	Y0	Y1	
SD78、SD79	SD178、SD179	SD278、SD279	SD778、SD779	SD878、SD879	SD978、SD979	脉冲数

注：起始位置最大可设置 1000000 个脉冲

示例：



6.2.1.5 电子凸轮起始位置

电子凸轮/电子齿轮可以通过 **SM** 软元件来设定起始位置。当起始位置设置开启后，电子凸轮启动时，从轴的起始位置将发生改变。

起始位置设置 **SM** 软元件如下表：

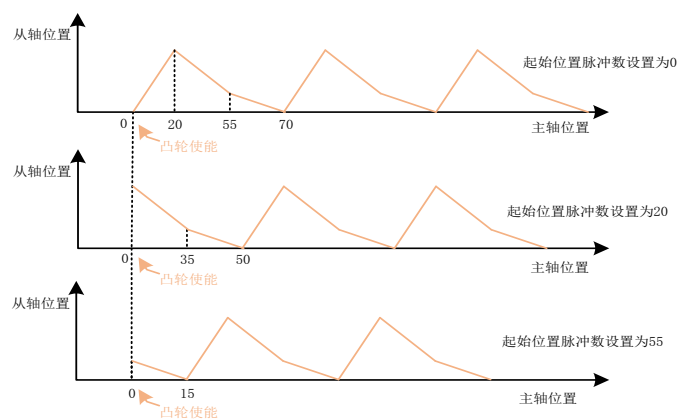
起始位置设置软元件						软元件状态	
轴 1	轴 2	轴 3	轴 4	Y0	Y1	ON	OFF
SM68	SM168	SM268	SM768	SM868	SM968	使能起始位置设置	关闭起始位置设置

当使能了起始位置设置后，使能电子凸轮，那么从轴位置跟主轴的位置的对应关系将会发生偏移，偏移量取决于起始位置脉冲数。

起始位置脉冲数 **SD** 软元件如下表：

起始位置脉冲数（32bit）						单位
轴 1	轴 2	轴 3	轴 4	Y0	Y1	
SD68、SD69	SD168、SD169	SD268、SD269	SD768、SD769	SD868、SD869	SD968、SD969	脉冲数

示例：



6.2.1.6 机械单位与脉冲单位的换算

建立凸轮表时可以选择使用机械单位（mm/min）或者脉冲单位（pls），其换算比例按照特殊寄存器设置为：

轴 1	轴 2	轴 3	轴 4	Y0	Y1	属性
SD6、SD7	SD106、SD107	SD206、SD207	SD706、SD707	SD806、SD807	SD906、SD907	电机旋转一圈所需脉冲数(A)
SD8、SD9	SD108、SD109	SD208、SD209	SD708、SD709	SD808、SD809	SD908、SD909	电机旋转一圈的运动距离(B)

$$\text{机械位置} \times \frac{\text{单圈脉冲数(A)} \times 1000}{\text{单圈距离(B)}} = \text{脉冲数}$$

$$\text{机械速度} \times \frac{\text{单圈脉冲数(A)} \times 1000}{\text{单圈距离(B)} \times 60} = \text{输出频率}$$

6.2.2 第二步：主轴选择/凸轮表选择

6.2.2.1 主轴选择

电子凸轮或齿轮功能需要主轴信号，主轴信号来源可选择外部输入或内部虚拟连接。主轴选择使用特殊元件如下表：

主轴选择设置 SM						主轴选择 SD					
轴 1	轴 2	轴 3	轴 4	Y0	Y1	轴 1	轴 2	轴 3	轴 4	Y0	Y1
SM71	SM171	SM271	SM771	SM871	SM971	SD71	SD171	SD271	SD771	SD871	SD971
OFF：内部虚拟连接						1：轴 1 作为虚拟主轴（D8340,D8341） 2：轴 2 作为虚拟主轴（D8342,D8343） 3：轴 3 作为虚拟主轴（D8344,D8345） 4：轴 4 作为虚拟主轴（D8346,D8347） 5：Y0 作为虚拟主轴（D8348,D8349） 6：Y1 作为虚拟主轴（D8350,D8351） 7：虚拟轴 1 作为虚拟主轴（D8352,D8353） 8：虚拟轴 2 作为虚拟主轴（D8354,D8355）					
ON：外部输入						1：轴 1 的脉冲输入作为主轴（SD56,SD57） 2：轴 2 的脉冲输入作为主轴（SD156,SD157） 3：轴 3 的脉冲输入作为主轴（SD256,SD257） 4：轴 4 的脉冲输入作为主轴（SD756,SD757）					

若主轴设置为内部虚拟连接，可以其他轴作为虚拟主轴，无需外部接线。例如轴 1 作为从轴，可以选择轴 2、轴 3、轴 4、Y0、Y1 作为轴 1 的虚拟主轴，但是不能选择自身作主轴。

若主轴设置为为外部输入，根据主轴选择 SD 元件的值可任意选择轴 1、轴 2、轴 3、轴 4 的输入通道作为外部输入主轴。

注意：当主轴设置为外部输入时，外部输入的脉冲形式（SD60、SD160、SD260、SD760）必须为 AB 正交形式。详见“4.2.1.3 输入/输出脉冲形式设定”

6.2.2.2 凸轮表选择

通过设置不同的凸轮表选择 SD 元件值，可以选择不同的凸轮表或者电子齿轮执行。

凸轮表选择使用的特殊软元件如下表：

凸轮表选择					
轴 1	轴 2	轴 3	轴 4	Y0	Y1
SD70	SD170	SD270	SD770	SD870	SD970

凸轮表选择特殊元件设置值与凸轮表关系：

凸轮选择 SD 元件值	说明
10	电子齿轮
11	凸轮表 1
12	凸轮表 2
13	凸轮表 3
14	凸轮表 4
15	凸轮表 5
16	凸轮表 6
其他	不执行，报 16262 错误

6.2.3 第三步：启动/停止电子凸轮

6.2.3.1 启动电子凸轮

电子凸轮/电子齿轮启动可选择为软件启动或事件触发启动；

启动模式 SM 元件如下表：

启动模式软元件						软元件状态	
轴 1	轴 2	轴 3	轴 4	Y0	Y1	ON	OFF
SM70	SM170	SM270	SM770	SM870	SM970	事件触发启动	软件启动

当启动模式选择为软件启动时，凸轮使能软元件如下表：

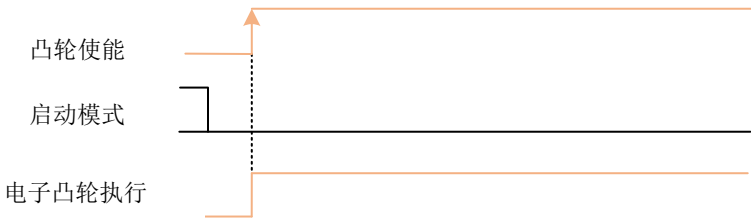
凸轮使能软元件						软元件状态	
轴 1	轴 2	轴 3	轴 4	Y0	Y1	ON	OFF
SM78	SM178	SM278	SM778	SM878	SM978	凸轮使能	凸轮不使能

当启动模式选择为事件触发启动时，触发启动方式设置如下表：（开发中）

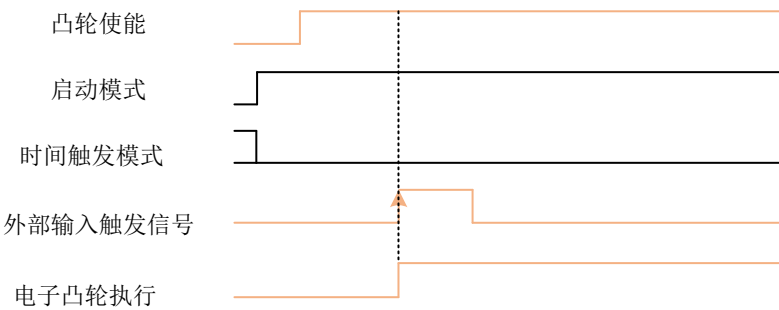
凸轮使能软元件						软元件状态	
轴 1	轴 2	轴 3	轴 4	Y0	Y1	ON	OFF
SM60	SM160	SM260	SM760	SM860	SM960	外部输入触发启动	比较中断触发启动

注：当选择启动模式为事件触发启动时，需要设定凸轮使能软元件为 ON，例如，当轴 1 的启动模式选择为事件触发启动（SM70=1），触发启动方式设置为 ON（SM60=1），那么如果想让轴 1 在外部输入触发时启动电子凸轮，还需要将凸轮使能位置为 ON（SM78=1）。

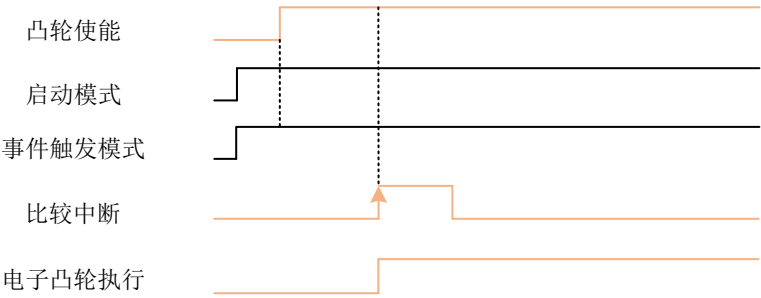
软件启动时序：



外部输入触发启动时序：（开发中）



比较中断触发启动时序：(开发中)



注：目前仅支持软件启动，SM70 需要设定为 OFF。

6.2.3.2 停止电子凸轮

电子凸轮执行时，可以将凸轮使能特殊软元件置为 OFF 来停止电子凸轮的执行。电子凸轮的停止会根据停止模式来选择是立即停止还是执行完本周期停止。

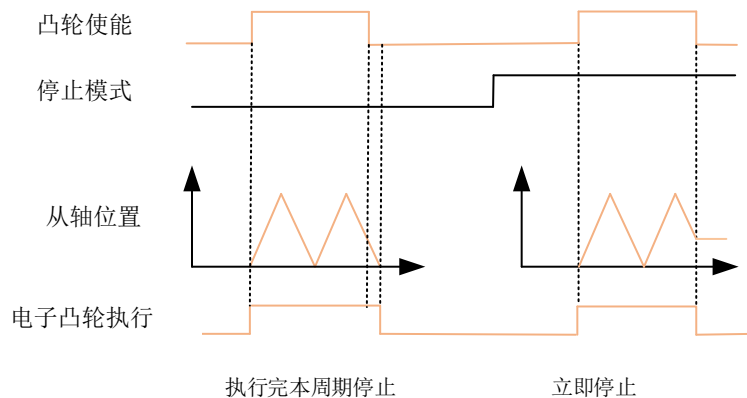
电子凸轮停止模式设置：

停止模式设置						软元件状态	
轴 1	轴 2	轴 3	轴 4	Y0	Y1	ON	OFF
SM81	SM181	SM281	SM781	SM881	SM981	立即停止 ^{注1}	执行完本周期停止 ^{注2}

注 1：立即停止是指在电子凸轮执行时，凸轮使能变为 OFF 或停止信号有效时，电子凸轮立即停止。

注 2：执行完本周期停止是指在电子凸轮执行时，凸轮使能变为 OFF 或停止信号有效时，电子凸轮执行完当前正在执行的周期后停止。

停止时序图：



电子凸轮的停止模式设置为执行完本周期停止时，凸轮使能位置为 OFF，电子凸轮会运行完本周期才停止，但是若需要立即停止电子凸轮执行，可以将电子凸轮运行中标志位由 ON 设置为 OFF，强制停止电子凸轮执行。

电子凸轮执行中/强制停止软元件：

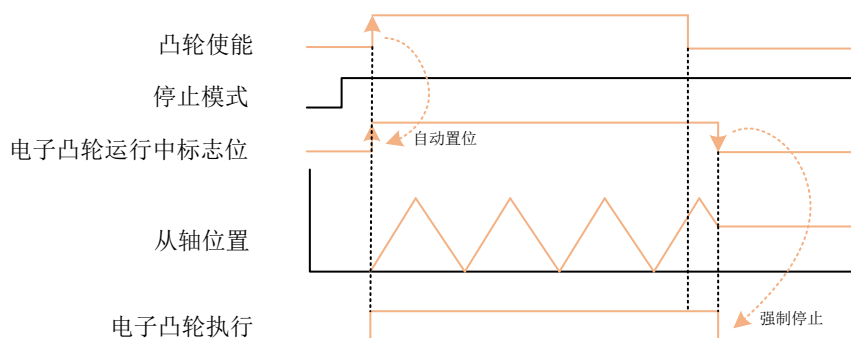
强制停止软元件						软元件状态	
轴 1	轴 2	轴 3	轴 4	Y0	Y1	ON	ON→OFF
SM89	SM189	SM289	SM789	SM889	SM989	电子凸轮执行中 ^{注1}	强制停止凸轮执行 ^{注2}

注 1：电子凸轮执行中为监控状态，表示电子凸轮正在执行，由软件自动置位。

注 2：若电子凸轮正在执行，将停止软元件置为 OFF，可使电子凸轮强制停止。

强制停止时序图：

下图为停止模式设置为执行完本周期停止的电子凸轮示意图，将电子凸轮使能置为 OFF 时，电子凸轮不会立即停止执行，但是若将电子凸轮运行中标志位置为 OFF，将强制停止电子凸轮的执行。



6.2.3.3 周期/非周期选择

电子凸轮可以选择周期或非周期执行，通过特殊 SM 和 SD 元件设置。

周期/非周期选择使用特殊元件如下表：

周期/非周期 SM						周期数设置 SD					
轴 1	轴 2	轴 3	轴 4	Y0	Y1	轴 1	轴 2	轴 3	轴 4	Y0	Y1
SM73	SM173	SM273	SM773	SM873	SM973	SD72	SD172	SD272	SD772	SD872	SD972
OFF：非周期执行						非周期执行次数，最大可设置 255 周期					
ON：周期执行						—					

周期执行：电子凸轮启动后，连续不断的周期执行电子凸轮表设定的关系，直到接收到停止命令；

非周期执行：电子凸轮启动后，执行完设定周期后自动停止，非周期执行的周期数由 SD 元件（SD72、SD172、SD272、SD772、SD872、SD972）设定，最大可设置 255 周期。

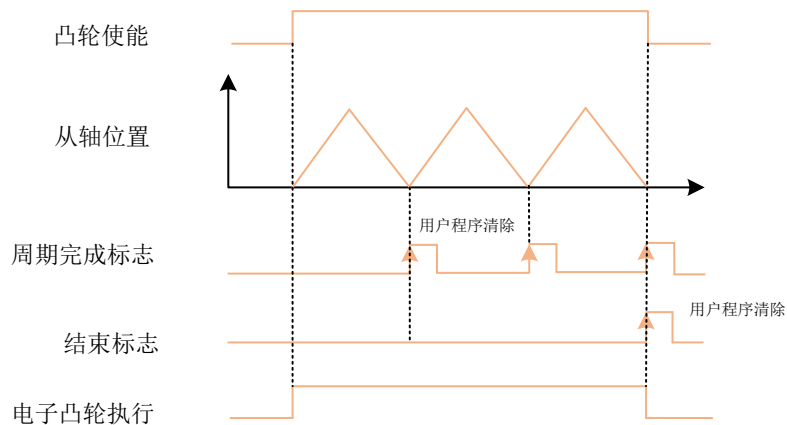
6.2.3.4 周期完成与结束标志

电子凸轮每完成一个周期，系统自动将周期完成标志特殊 SM 元件置位为 ON。周期完成标志置位后一直保持为 ON 状态，如果需要检测下一个周期的完成，需要用户程序将周期完成标志清为 OFF，在下一个周期完成时，系统再一次将周期完成标志置位为 ON。

电子凸轮/电子齿轮执行结束时，系统自动将结束标志特殊 SM 元件置位为 ON。结束标志在每次凸轮使能时由系统清为 OFF，也可由用户程序清为 OFF。

周期完成与结束标志使用的特殊 SM 元件如下表：

	轴 1	轴 2	轴 3	轴 4	Y0	Y1	说明
周期完成标志	SM79	SM179	SM279	SM779	SM879	SM979	电子凸轮每完成一个周期置位为 ON
结束标志	SM80	SM180	SM280	SM780	SM880	SM980	电子凸轮/电子齿轮执行结束时置为 ON



6.3 电子凸轮关键点修改

6.3.1 DCAMWR 电子凸轮数据修改

1. 指令形式

修改电子凸轮表数据。

DCAMWR m1 m2 D n				电子凸轮数据修改	指令执行
m1	凸轮表	指定需要修改的凸轮表			32 位指令（17step） DCAMWR 连续执行
m2	起始点	设定需要修改凸轮表数据的起始点			
D	数据	需要修改的数据存放地址			
n	点数	需要修改的关键点数据点数			

2. 操作数

操作数	位软元件							字软元件															
	系统 用户							系统 用户					位数指定					变址			常数		实数
m1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
m2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E

注：表中灰底的软元件，表示不支持的软元件。

- m1 指定需要修改的凸轮表，m1=k11~k16 分别指定电子凸轮表 ID1~ID6；
- m2 设定需要修改的电子凸轮关键点起始点，m2=k2~k360；
- D 修改数据的起始地址，占用以 D 起始的多个连续地址单元。每个关键点占用 2 个 32bit 寄存器分别标志主轴和从轴位置，即每个关键点需要占用 4 个地址单元；
- n 需修改的关键点个数，m2+n-1 需小于等于已下载关键点个数；

3. 注意要点

- DCAMWR 指令每次只能执行一条，若程序中需要 2 条以上 DCAMWR 指令，需要在前一条指令停止一个扫描周期后，才能启动下一条指令。
- DCAMWR 为多周期执行指令，特殊寄存器 SM82 由 OFF→ON 变化表示修改完成。
- DCAMWR 修改完成表示 PLC 内部的凸轮表数据已经改变，此时上载或用指令读回凸轮表数据为修改后数据；若电子凸轮正在运行，修改完成后的数据不能马上生效，需要置位特殊标志位（SM83、SM183、SM283、SM783），则可以在当前运行的下一个凸轮周期生效；若电子凸轮未执行，或者不需要在正在运行的电子凸轮中生效，下一次启动电子凸轮时，修改后的凸轮表数据自动生效。
- 电子凸轮表第 1 点为起点数据，不能修改，所以 m2 必须大于 1；指令参数 m2+n-1 需小于或者等于已下载关键点个数。
- 修改凸轮表数据时，主轴位置数据必须大于前一点的主轴位置并且小于后一点的主轴位置数据，否则运动控制器报 16268 错误。
- DCAMWR 指令指定修改的凸轮表，必须在运动控制系统中存在，即凸轮表已经通过 VCAutoDesignsoft 下载到运动控制系统中。

4. 电子凸轮表数据修改相关特殊寄存器

下面列举了电子凸轮相关的特殊功能寄存器。

软元件	轴号	含义
SM82	所有轴	SM82 由 OFF→ON 变化表示修改完成
SM83	轴 1	电子凸轮修改运行中生效标志： ON: 修改后的数据在当前运行的下一个凸轮周期生效，生效后自动复位为 OFF； OFF: 修改后的数据下一次启动凸轮时生效。
SM183	轴 2	
SM283	轴 3	
SM783	轴 4	
SM883	Y0	
SM983	Y1	

能够支持的指令

NO	m2 取值	说明
1	K2~K360	单点或者多点修改指令
2	K361	顶杆修改指令
3	K362	单个数据修改指令(一组点中，一个数据)
4	K10000	全点修改
5	K10002	飞剪
6	K10004	追剪
7	K10030	包装机专用
8	K10031	包装机专用
9	K10032	包装机专用
10	K10033	包装机专用
11	K10034	包装机专用

6.3.2 DCAMRD 读取电子凸轮数据(开发中)

1. 指令形式

读取电子凸轮表数据。

DCAMRD m1 m2 D n				读取电子凸轮数据	指令执行
m1	凸轮表	指定需要读取的凸轮表			32 位指令（17step） DCAMWR 连续执行
m2	起始点	设定需要读取凸轮表数据的起始点			
D	数据	存放读取到的凸轮表数据地址			
n	点数	需要读取的关键点数据点数			

2. 操作数

操作数	位软元件							字软元件															
	系统 用户							系统 用户				位数指定					变址		常数		实数		
m1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
m2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E

注：表中灰底的软元件，表示可以支持的软元件。

- m1 指定需要读取的凸轮表，m1=k11~k16 分别指定电子凸轮表 ID1~ID6；
- m2 设定需要修改的电子凸轮关键点起始点，m2=k1~k360；
- D 存放读取到的凸轮表数据的起始地址，占用以 D 起始的多个连续地址单元。每个关键点占用 2 个 32bit 寄存器分别标志主轴和从轴位置，即每个关键点需要占用 4 个地址单元；
- n 需修改的关键点个数，m2+n-1 需小于等于已下载关键点个数；

3. 注意要点

- DCAMRD 读取电子凸轮表数据，指定读取的凸轮表必须在运动控制系统中存在，即凸轮表已经通过 VCAutoDesignsoft 下载到运动控制器中。
- 令参数 m2+n-1 需小于等于已下载关键点个数。

6.3.3 全点修改指令

1. 指令形式

修改电子凸轮表数据。

DCAMWR m1 m2 Dn			全点修改指令	指令执行
m1	凸轮表	指定需要修改的凸轮表		32 位指令（17step） DCAMWR 连续执行
m2	起始点	设定需要修改凸轮表数据的起始点		
D	数据	需要修改的数据存放地址		
n	点数	需要修改的关键点数据点数		

2. 操作数

操作数	位软元件							字软元件															
	系统 用户							系统 用户				位数指定					变址		常数		实数		
m1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
m2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
n	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E

注：表中灰底的软元件，表示不支持的软元件。

当 m2=K10000 时，为全点修改功能。CAMWR 的参数定义如下表所示：

参数	含义	说明	取值
M1	凸轮表	凸轮表号，选择修改哪一个表	K11~K16
D	起始寄存器	数据点数存放寄存器	D 元件
n	修改点数	电子凸轮总点数	2~25

举例：

DCAMWR K11 K10000 D100 K3

该指令对电子凸轮表 1 进行修改，修改全部电子凸轮表，修改的点数为 3 个点，点的起始位置为 D100。

具体待修改的数据见下表所示

修改点数	位置	数据存放寄存器
第 1 点	主轴位置	D100D101
	从轴位置	D102D103
	速比（浮点型）	D104D105
	线型	D106D107
第 2 点	同上	D108D109
第 3 点	同上	D116D117

6.3.4 单数据修改指令 DCAMWR 指令

1. 指令形式

设置凸轮执行数据

DCAMWR S1 S2 D1 D2					单数据修改指令	指令执行
S1	凸轮表					32位指令（13setp） 连续执行
S2	固定K362					
D1	数据地址（主轴，从轴，速比，类型）					
D2	数据所在行，列					

2. 操作数

操作数	位软元件							字软元件															
	系统 用户							系统 用户				位数指定					变址			常数		实数	
s1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
s2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
D1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
D2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E

注：表中灰底的软元件，表示不支持的软元件。

功能和动作说明：

S1	凸轮表 K11：第一个凸轮表；K12，K13 依次为第二，三等。
S2	固定为 K362，十进制
D1	数据的地址
D2	数据所在行，列计算如下：行=D2/100；列=D2%100；

注 1：数据所在行，不得超过存在的最大点数。

注 2：数据列，范围为 1~4，分别对应主轴，从轴，速比，类型。

注 3：修改数据类型时，注意数据的所属类型

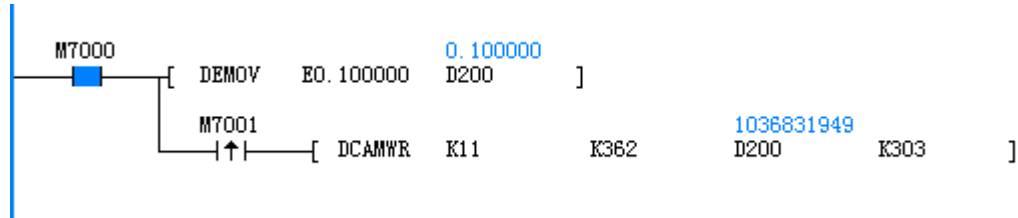
数据所在列	1	2	3	4
数据类型	主轴	从轴	速补	类型
对应	32 位	32 位	Float	16 位

示例:

原有的凸轮表为:

	Add	Del	M-Pos	S-Pos	PU Speed	Type
1			0	0	1.00	NA
2			2500	7000	2.80	Line
3			10000	30000	3.07	Line

修改指令如下:



执行的结果: 第三个点的, 第三列数据 3.07 修改为 0.1.

注: 尽量使用沿触发的方式修改。

注 2: 修改生效后 SM82 会置位为 ON, 失败置位为 OFF。

注 3: 该功能的使用, 请注意修改时的数据保障在至少前一点, 否则会导致不可预计的错误。

6.3.5 顶杆修改指令

1. 指令形式

修改电子凸轮表数据。

DCAMWR S1 S2 D1 D2					顶杆修改指令	指令执行
S1	凸轮表					32位指（13setp） 连续执行
S2	固定K361					
D1	顶杆起始地址					
D2	顶杆个数					

2. 操作数

操作数	位软元件							字软元件																
	系统 用户							系统 用户					位数指定					变址			常数		实数	
s1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E	
s2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E	
D1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E	
D2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E	

注：表中灰底的软元件，表示不支持的软元件

功能和动作说明：

S1	凸轮表K11：第一个凸轮表；K12，K13依次为第二，三等。
S2	固定为K361，十进制
D1	数据的起始地址，示例入下
D2	顶杆的个数，0位不设置顶杆数据

注：a.凸轮使能前，需要通过该指令设定，才能写入。

b.运行过程中不允许修改。

数据格式表：

NO	地址	位数	范围	注释
1	D1	16 位	0,1	元件类型 0: M 元件 1: Y 元件
2	D1+1	16 位	变化	元件地址 M 元件时地址为 0~767Y 元件时地址为 0~191
3	D1+2	16 位	0,1	动作类型 0: OFF, 1: ON
4	D1+3	16 位		预留
5	D1+4	32 位		顶杆数据, pulse 单位

注：Y 地址时，注意八进制与十进制转换，Y10 对应 K8。

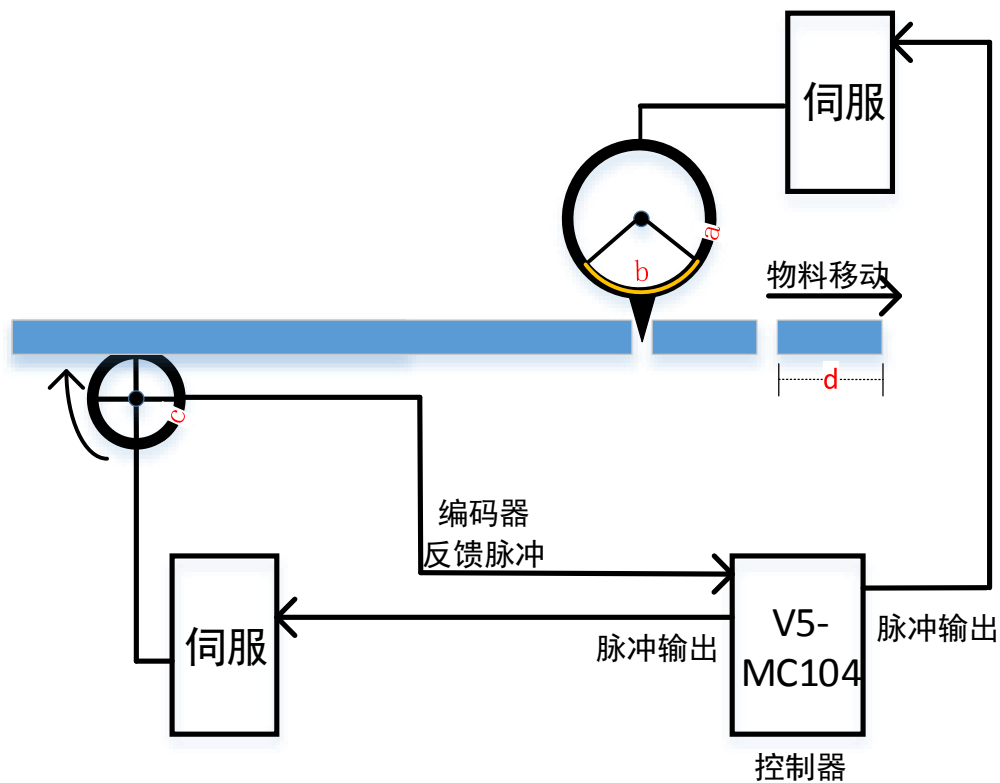
6.4 飞剪

6.4.1 功能说明

6.4.1.1 飞剪简介

在送料裁切应用上，传统的做法是使用走停式，送料轴先走到固定的长度，随后剪料轴再动作，之后不断重复“送料停”及“剪料停”的过程。这种方法的缺点在送料轴走停的过程中，所需要的加减速使生产效益无法提高，因此新的做法是采用送料不停的方式来达成，一般有二种送料裁切的方法分别为飞剪跟追剪，两者的差异为追剪是往返运动，而飞剪是同向运动，所设定的 CAM 表曲线也不同。下面将对飞剪功能进行详细说明。

6.4.1.2 飞剪原理



图中：

- a: 从轴一圈长度 (mm)
- b: 从轴同步区长度 (mm)
- c: 主轴一圈长度 (mm)
- d: 物料长度 (mm)

从轴一圈长度：从轴转动一圈的长度，即从轴周长，该长度等于从轴直径 * π

从轴同步区长度：当主轴速度和从轴速度相同的区域为同步区长度。

主轴一圈长度：主轴转动一圈的长度，即主轴周长，该长度等于主轴直径 * π

物料长度：需要裁切的物料的长度。

6.4.1.3 从轴移动方式

在实际的追剪系统中，从轴跟随主轴做电子凸轮运动。料长跟切刀周长关系不同，电子凸轮的关键点也不相同。下面介绍下从轴是如何跟随主轴运动的。

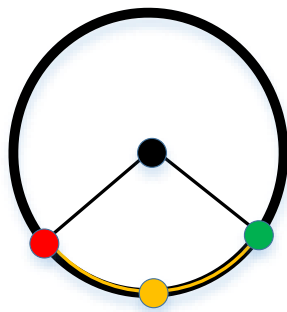
◆ 裁切短料

当 $2 \times \text{从轴一圈长度} > \text{物料长度}$ 时，系统将判定该物料为短料，按照短料的曲线规划方式来规划，短料的规划点为 4 个点。

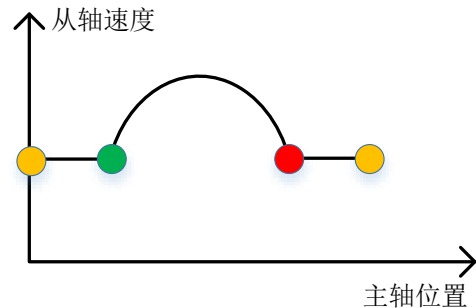
如下图 1.1 所示，凸轮啮合的起点为同步区中点，从轴从黄点同步运行到绿点，然后由绿点再移动到红点，再由红点移动到黄点，完成一个周期的移动。



图 1.2 给出了当裁切短料时，主从之间的曲线关系。黄到绿段和红到黄段均为同步区；绿到红段为变速区。



短料点规划



短料主从关系曲线

◆ 主从位置计算：

符号	名称
L	切料长度 (mm)
L1	从轴一圈长度 (mm)
P1	从轴一圈脉冲数
D	同步区长度 (mm)
L2	主轴一圈长度 (mm)
P2	主轴一圈脉冲数

关键点	位置 (脉冲个数)	计算公式
第一点	主轴	0
	从轴	0
第二点	主轴	$\frac{P2}{2 \times L2} \times D$
	从轴	$\frac{P1}{2 \times L1} \times D$
第三点	主轴	$\frac{2 \times L - D}{2 \times L2} \times P2$
	从轴	$P1 - \frac{P1}{2 \times L1} \times D$

第四点	主轴	$\frac{P2}{L2} \times L$
	从轴	P1

◆ 裁切长料

当 $2 \times \text{从轴一圈长度} < \text{物料长度}$ 时，系统将判定该物料为长料，按照长料的曲线规划方式来规划，长料的规划点为 6 个点。裁剪长料时，从轴具有一个等待区，此时主轴移动，从轴处于静止状态。

如下图 1.3 所示，凸轮啮合点为同步区中点，从轴从黄点同步运行到绿点，然后由绿点运行到蓝点，再由蓝点运行到红点，最后再由红点同步运行到黄点。

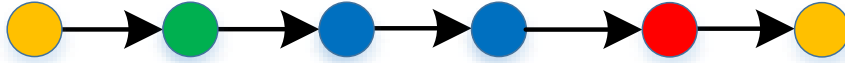
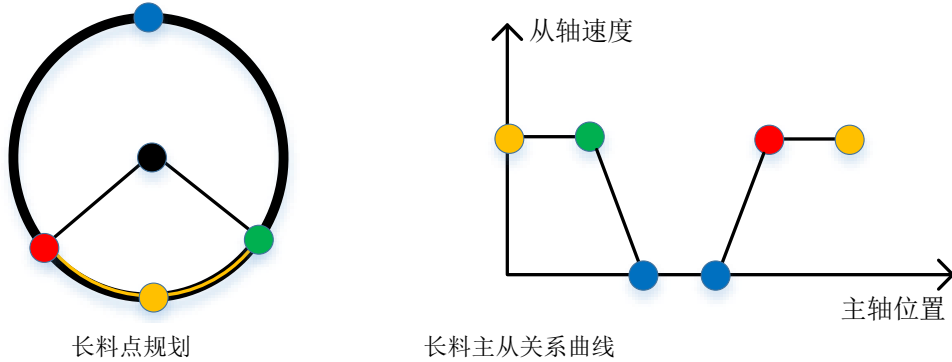


图 1.4 给出了当裁剪长料时，主从之间的曲线关系，黄点到绿点、红点到黄点均为同步区，两个蓝点之间为停止区。绿点到蓝点、蓝点到红点为非同步区。



◆ 主从位置计算：

符号	名称
L	切料长度（mm）
L1	从轴一圈长度（mm）
P1	从轴一圈脉冲数
D	同步区长度（mm）
L2	主轴一圈长度（mm）
P2	主轴一圈脉冲数

关键点	位置（脉冲个数）	计算公式
第一点	主轴	0
	从轴	0
第二点	主轴	$\frac{P2}{2 \times L2} \times D$
	从轴	$\frac{P1}{2 \times L1} \times D$
第三点	主轴	$\frac{L1 - D}{L2} \times P2 + \frac{P2}{2 \times L2} \times D$

	从轴	$\frac{P1}{2}$
第四点	主轴	$\frac{P2}{L2} \times L - \left(\frac{L1-D}{L2} \times P2 + \frac{P2}{2 \times L2} \times D \right)$
	从轴	$\frac{P1}{2}$
第五点	主轴	$\frac{P2}{L2} \times L - \frac{P2}{2 \times L2} \times D$
	从轴	$P1 - \frac{P1}{2 \times L1} \times D$
第六点	主轴	$\frac{P2}{L2} \times L$
	从轴	P1

6.4.1.4 飞剪操作步骤

第一步：回原点（若保存了主从数据，可不执行回原点）

第二步：初始化电子凸轮，并通过 CAMWR 指令将飞剪数据写入

第三步：根据当前主从位置关系，启动电子凸轮

具体的操作可参见“1.2 应用范例”

6.4.1.5 DCAMWR_K10002 指令

该功能通过指令 DCAMWR 指令写入，指令的形式如下表所示：

DCAMWR m1 m2 D n				飞剪参数写入	指令执行
m1	凸轮表	指定需要修改的凸轮表			32 位指令（17step） DCAMWR 连续执行
m2	子集选择				
D	数据	需要修改的数据存放地址			
n	保留				

- m1 位凸轮表选择功能，选择修改的凸轮表，m1 的取值范围为 K11~K16。
- m2 为应用功能子集选择，当 m2 设置为 K10002 时，为飞剪专用功能。
- D 为需要修改的数据的存放地址。
- 对于飞剪专用功能块，n 为保留位。

示例：

DCAMWR K11 K10002 D100 K0

该条指令表示将以 D100 位首的一组飞剪参数写入到凸轮表 1 中，其中相关参数存放在 D100 为首的 12 个 D 软元件中，参数如下表：

数据类型	寄存器	定义	单位
浮点型	D100D101	主轴一圈长度	mm
无符号整型	D102D103	主轴一圈脉冲数	个
浮点型	D104D105	从轴一圈长度	mm
无符号整型	D106D107	从轴一圈脉冲数	个
浮点型	D108D109	同步区长度	mm
浮点型	D110D111	切料长度	mm

6.4.1.6 参数生效

飞剪功能参数写入后，可选择与下一个凸轮周期生效，也可以选择下次启动电子凸轮时生效，生效的方式取决于 M 软元件。

生效方式						软元件状态	
轴 1	轴 2	轴 3	轴 4	Y0	Y1	ON	OFF
SM83	SM183	SM283	SM783	SM883	SM983	下一个凸轮周期生效	下次启动凸轮生效

当相应的 M 软元件设置为 ON 后，为下一个凸轮周期生效，当相应的 M 软元件设置为 OFF 后，为下次启动电子凸轮时生效。

当飞剪参数生效后，软件会自动将 SM82 软元件值 ON（该软元件对所有轴都通用）。

6.4.2 应用范例

6.4.2.1 基础案例

下面将给出一个飞剪的使用案例。其飞剪参数如下：

- 主轴直径：150mm
- 主轴传动比：1:10
- 从轴直径：200mm
- 同步区角度：60°
- 从轴传动比：1:3
- 裁剪料长：500mm

根据上述的系统参数，计算出 CAMWR 指令需要设置的值。

（伺服采用 SD700 伺服，内部设置 10000 个脉冲伺服电机选择一圈）

主轴一圈长度=150* π =471

主轴一圈脉冲数=10000*10=100000

从轴一圈长度=200* π =628

从轴一圈脉冲数=10000*3=30000

同步区长度=60°/360°*从轴一圈长度=104.667

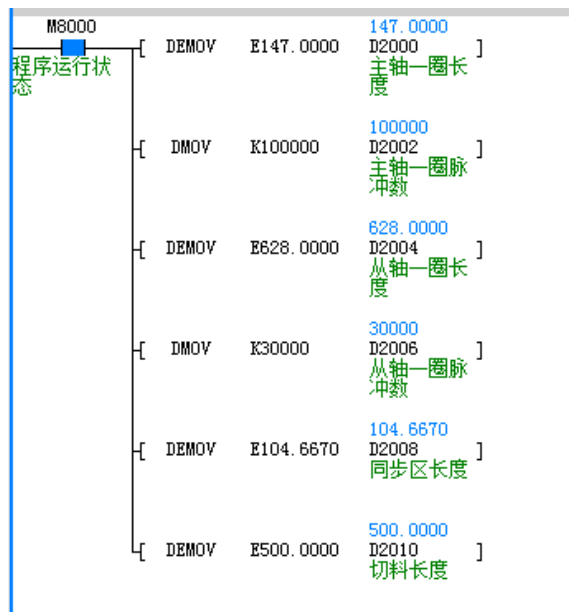
参数	单位	设定值
主轴一圈长度	mm	471
主轴一圈脉冲数	个	100000
从轴一圈长度	mm	628
从轴一圈脉冲数	个	30000
同步区长度	mm	104.667
料长	mm	500

下面我们给出一个简单的使用追剪的案例：

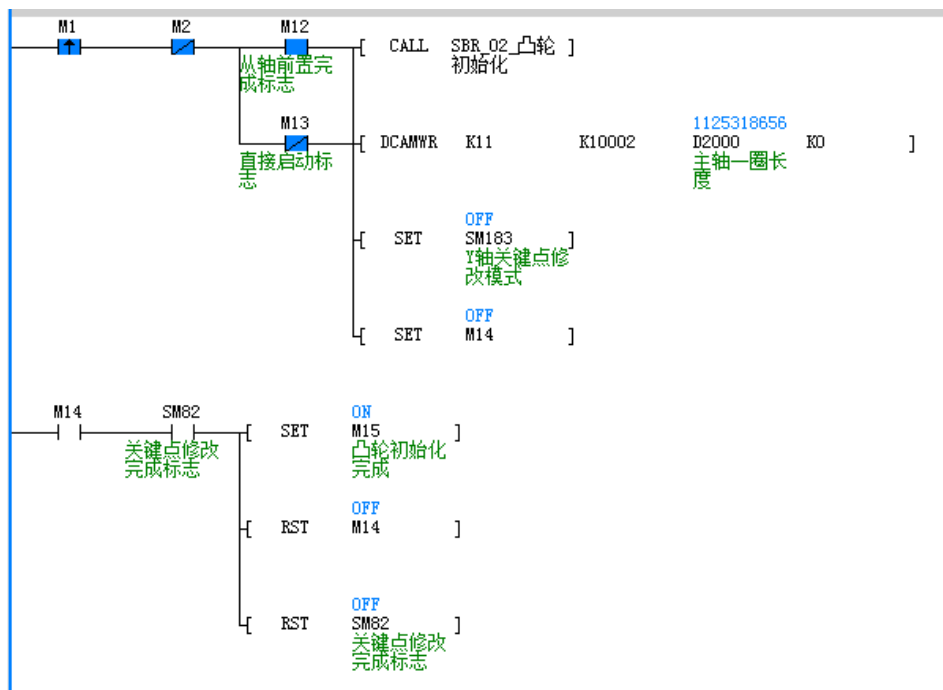
第一步：设置电子凸轮基本参数

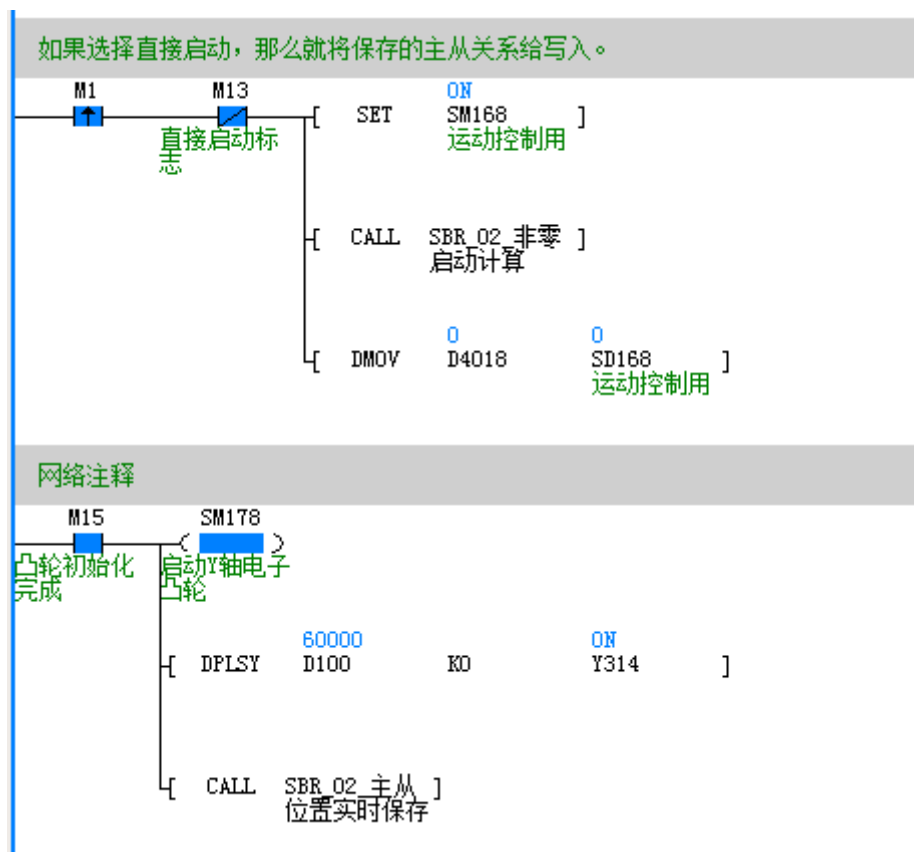


第二步：设置飞剪参数

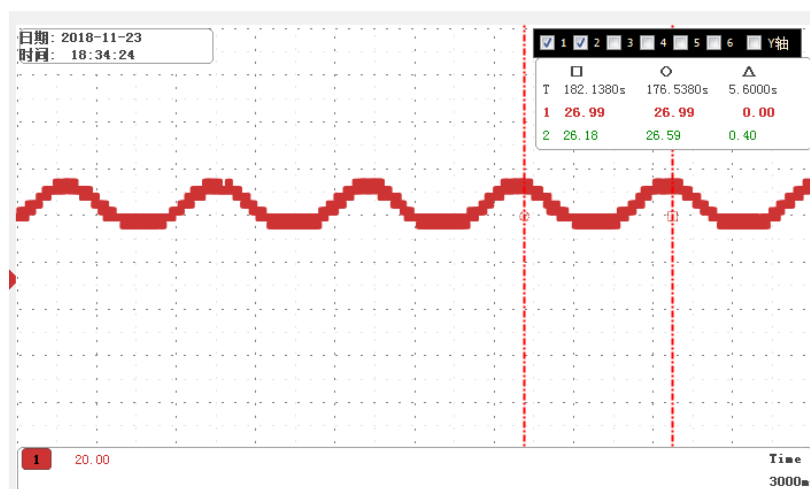


第三步：初始化完成后，启动电子凸轮即按照飞剪曲线来运行
主函数如下图所示：





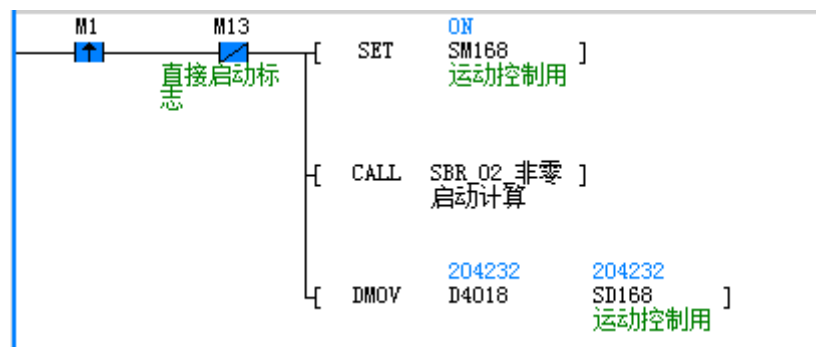
实际的运行结果如下图所示：



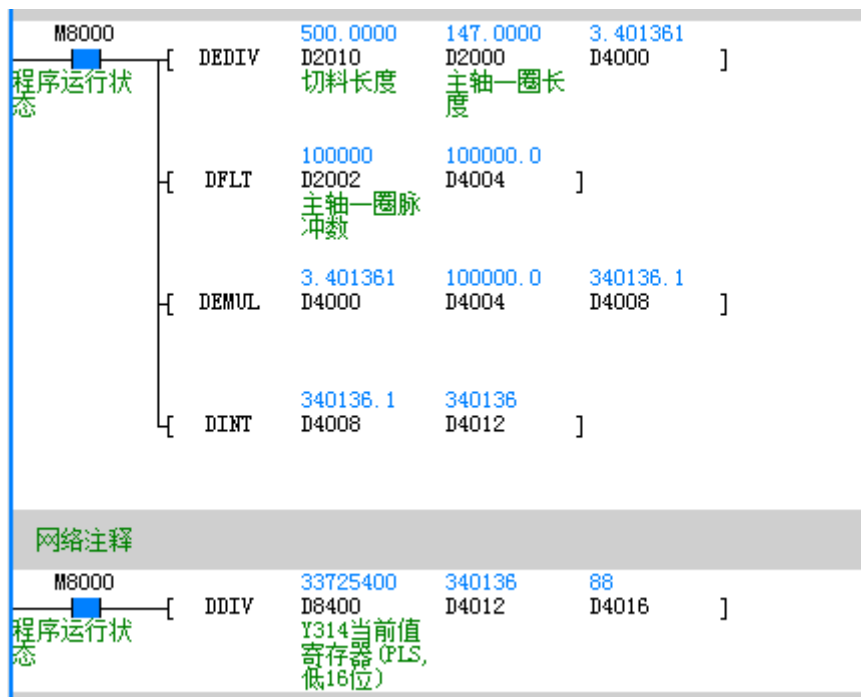
6.4.2.2 上电即运行案例

如果需要启动立即运行系统，那么可以通过非零启动功能来实现。

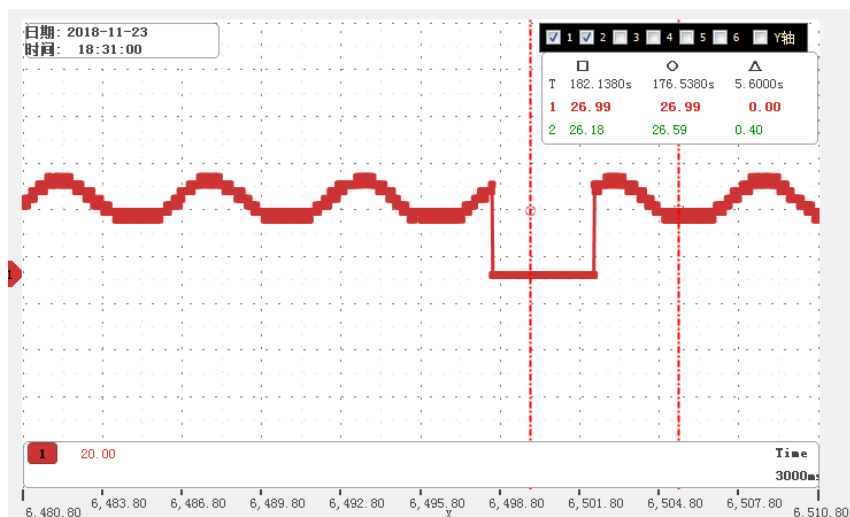
梯形图实现方式如下图所示：



其中非零启动计算的梯形图如下图所示：



最终的运行结果如下图所示，从图中可以看出，当将 PLC 置为 OFF 后，若再将 PLC 置为 ON，可立即启动，启动仍接着停止时的位置来运行。

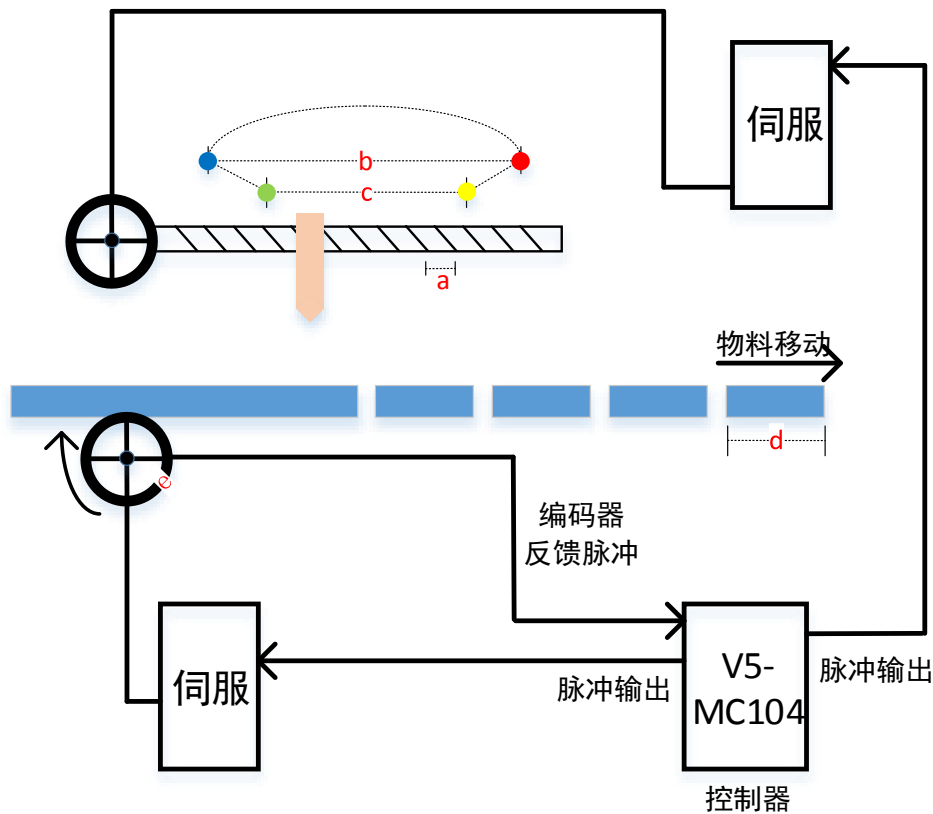


6.5 追剪

6.5.1 功能说明

6.5.1.1 追剪简介

追剪是对运动中的材料进行垂直切割、填充的工艺。其整个流程如下图所示，V5-MC104 带两个伺服轴---送料轴和追剪轴，送料轴主要是牵引着物料往前移动，追剪轴主要是追踪送料轴，从而在同步区进行加工。



图中：

- a: 从轴一圈导程 (mm)
- b: 从轴移动范围 (mm)
- c: 同步区 (mm)
- d: 物料长度 (mm)
- e: 主轴一圈长度 (mm)

从轴一圈导程是指当从轴电机旋转一周时，从轴移动的距离，若从轴采用丝杠结构，那么该值指丝杠的导程。

从轴移动范围，指从轴的移动范围，从轴只在该安全范围内移动，蓝点为从轴移动范围的起点，红点为从轴移动范围的终点。

同步区指主轴和从轴速度相同时的区域，在此区域内，主轴移动速度和从轴移动速度相同，在此区域内可以完成一些裁切等动作。

物料长度指需要裁切的物料的长度。

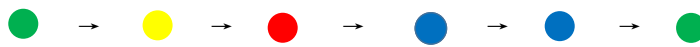
主轴一圈长度指主轴传动轮周长，其值等于主轴传动轮直径 $\times \pi$ 。

6.5.1.2 从轴移动方式

长短料判断：料长>从轴移动范围×（2+返回系数/50）-同步长度时认为是长料，否则判断为短料。

根据物料的长短，从轴的移动顺序有差异。具体的从轴移动顺序如下所示：

◆ 长料移动顺序：



第一点：第一点为绿点，即同步区的起点。

第二点：第二点为黄点，即同步区的终点，从轴以同步区速度匀速从绿点移动到黄点。

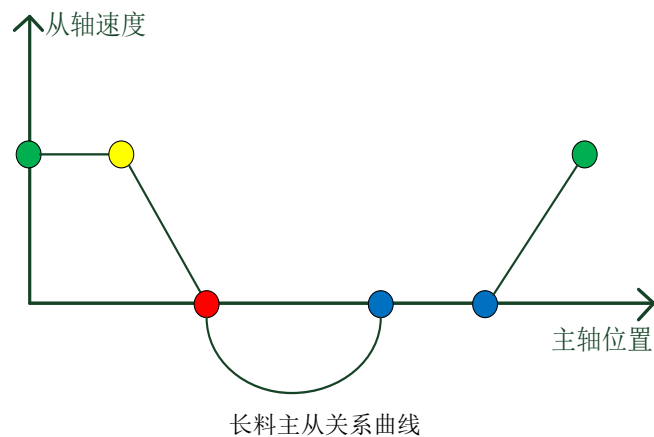
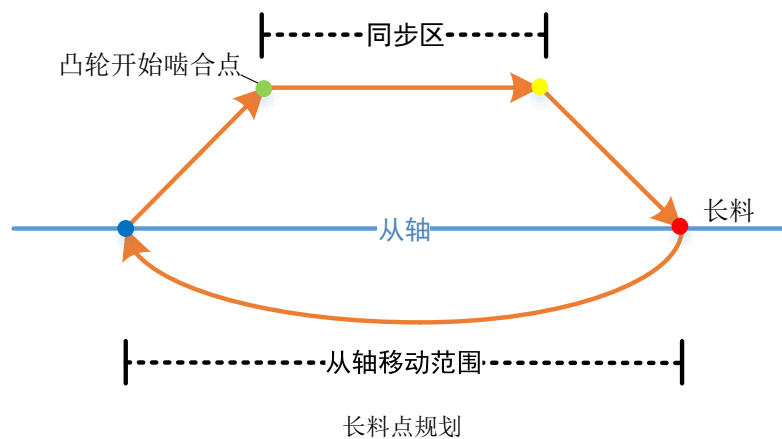
第三点：第三点为红点，即从轴移动范围的终点，从轴从第二点逐渐减速移动到第三点。

第四点：第四点为蓝点，即从轴移动范围的起点，从轴从第三点返回到第四点并停止等待点。

第五点：第五点为蓝点，即等待时间到从轴加速到同步区起点。

第六点：第六点为绿点，同第一点。

长料移动的示意图如下图所示：



◆ 主从位置计算：

符号	名称
L	物料长度 (mm)
L1	从轴一圈导程 (mm)
L2	从轴移动范围 (mm)
P1	从轴一圈脉冲数
D	同步区 (mm)
L3	主轴一圈长度 (mm)
P2	主轴一圈脉冲数
K	返回系数
R	角速度比

$$\text{角速度比: } R = \frac{(L3 \times P1)}{(P2 \times L1)}$$

关键点	位置 (脉冲个数)	计算公式
第一点	主轴	0
	从轴	0
第二点	主轴	$\frac{P2}{L3} \times D$
	从轴	$\frac{P2}{L3} \times D \times R$
第三点	主轴	$\frac{P2}{L3} \times L2$
	从轴	$\frac{L2 + D}{2} \times \frac{P2}{L3} \times R$
第四点	主轴	$\left(1 + \frac{K}{50}\right) \times \frac{P2}{L3} \times L2$
	从轴	$-\left(\frac{L2 - D}{2} \times \frac{P2}{L3} \times R\right)$
第五点	主轴	$\frac{P2}{L3} \times (L - L2 + D)$
	从轴	$-\left(\frac{L2 - D}{2} \times \frac{P2}{L3} \times R\right)$
第六点	主轴	$L \times \frac{P2}{L3}$
	从轴	0

◆ 短料移动顺序：

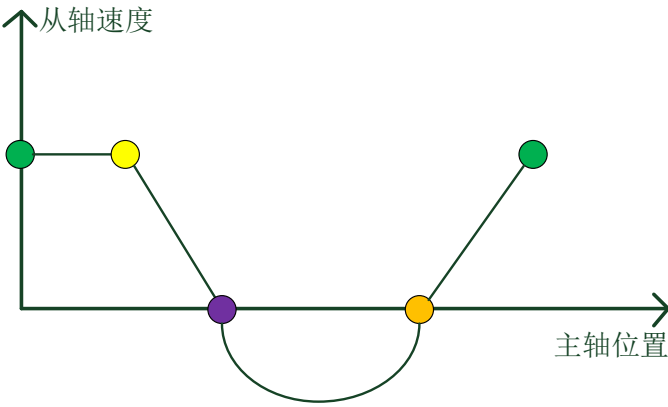
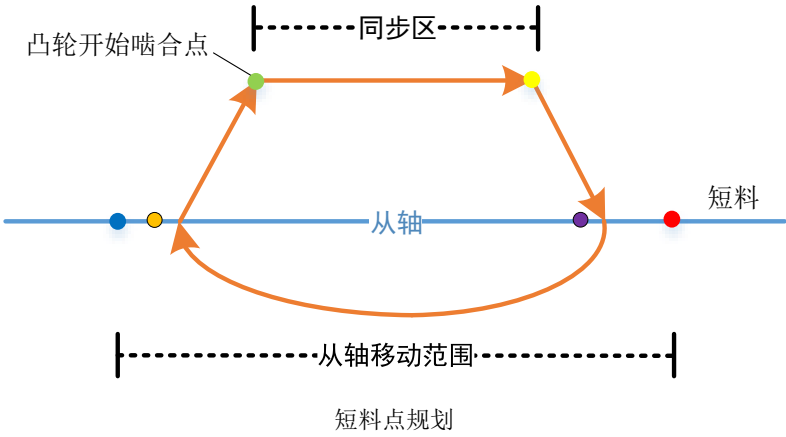
当物料较短时，若从轴仍然按照长料的五点运动，那么可能会导致从轴在返回时的速度过快而导致伺服跟踪不上。所以针对裁剪物料较短的情况，我们将移动点做了修改。

修改：

- 1) 缩小减速时间，即从轴从黄点减速，快速减速到 0，不减速到红点，而减速到红点后面的位置。
- 2) 缩小加速时间，即从轴不从蓝点加速，而是从蓝点前的位置开始加速，这样就要求从轴回车时，

不完全回车到蓝点位置，而回车到蓝点前的位置。

短料移动的示意图如下图所示：



注意：料长不能过短，料长过短时，有可能导致从轴返回速度过快，从而导致伺服报警。

◆ 主从位置计算：

符号	名称
L	物料长度（mm）
L1	从轴一圈导程（mm）
L2	从轴移动范围（mm）
P1	从轴一圈脉冲数
D	同步区（mm）
L3	主轴一圈长度（mm）
P2	主轴一圈脉冲数
K	返回系数
R	角速度比

角速度比：
$$R = \frac{(L3 \times P1)}{(P2 \times L1)}$$

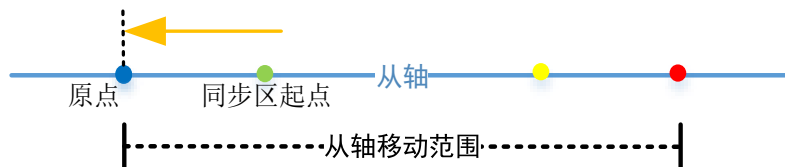
关键点	位置（脉冲个数）	计算公式
-----	----------	------

第一点	主轴	0
	从轴	0
第二点	主轴	$\frac{P2}{L3} \times D$
	从轴	$\frac{P2}{L3} \times D \times R$
第三点	主轴	$\frac{D + L2}{2} \times \frac{P2}{L3}$
	从轴	$\frac{3 \times D + L2}{4} \times \frac{P2}{L3} \times R$
第四点	主轴	$\left(\frac{D - L2}{2} + L\right) \times \frac{P2}{L3}$
	从轴	$-\left(\frac{L2 - D}{4} \times \frac{P2}{L3} \times R\right)$
第五点	主轴	$\frac{P2}{L3} \times L$
	从轴	0

6.5.1.3 追剪操作步骤

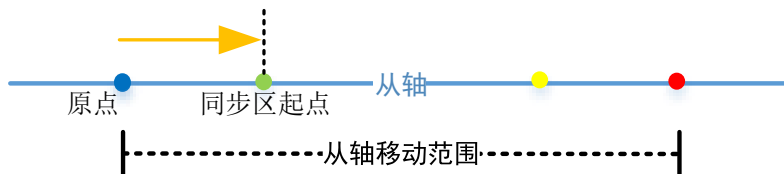
根据追剪的实际应用，物料有长短料之分，但是不管是长料还是短料，其同步区起点是固定的，所以我们这里将整个操作的起点定在了同步区的起点。整个系统的操作流程如下所示。

第一步：回原点。通过控制器的回原点指令控制从轴回到原点位置。这里我们把蓝点当做原点，即控制从轴回到蓝点位置。



第二步：从轴前置。回到原点之后，需要控制从轴移动到同步区起点位置，同步区起点位置即凸轮啮合的起点。

$$\text{前置量} = (\text{从轴移动范围} - \text{同步区长度}) / 2$$



注：第一步和第二步一般只需要在调试机械时执行一次，后续保存了主从轴的位置后，上电后可立即啮合凸轮。

第三步：配置通用凸轮，并通过 DCAMWR 指令将追剪参数写入。详细见“2.1.4DCAMWR 飞剪参数写入功能”。

第四步：使能凸轮。使得主轴和从轴啮合。

第五步：控制器控制主轴移动。主从关系按照设定的凸轮关系运行。

6.5.1.4 DCAMWR_K10004 指令

该功能通过指令 DCAMWR 指令写入，指令的形式如下表所示：

DCAMWR m1 m2 D n				追剪参数写入	指令执行
m1	凸轮表	指定需要修改的凸轮表			32 位指令（17step） DCAMWR 连续执行
m2	子集选择				
D	数据	需要修改的数据存放地址			
n	保留				

- m1 位凸轮表选择功能，选择修改的凸轮表，m1 的取值范围为 K11~K16。
- m2 为应用功能子集选择，当 m2 设置为 K10004 时，为追剪专用功能。
- D 为需要修改的数据的存放地址。
- 于追剪专用功能块，n 为保留位。

示例：

DCAMWR K11 K10004 D100 K0

该条指令表示将以 D100 位首的一组追剪参数写入到凸轮表 1 中，其中相关参数存放在 D100 为首的 16 个 D 软元件中，参数如下表：

数据类型	寄存器	定义	单位	取值范围
浮点型	D100D101	主轴一圈长度	mm	
无符号整型	D102D103	主轴一圈脉冲数	个	
浮点型	D104D105	从轴一圈导程	mm	
无符号整型	D106D107	从轴一圈脉冲数	个	
浮点型	D108D109	同步区长度	mm	
浮点型	D110D111	从轴移动范围	mm	
无符号整型	D112D113	返回系数		40-500
浮点型	D114D115	切料长度	mm	

6.5.1.5 飞剪参数作用

参数	作用	计算
主轴一圈长度	主轴一圈的长度	主轴传动轮直径 * π
主轴一圈脉冲数	主轴转动一圈的脉冲数	伺服电机转动一圈脉冲数/传动比
从轴一圈导程	丝杠转动一圈的导程	实际而定
同步区长度	同步区的长度	实际而定
从轴移动范围	从轴可以移动的范围	实际而定
返回系数	调节从轴返回速率注 1	推荐默认为 100（长料时才有效）
切料长度	需要裁剪物料的长度	实际而定

注 1：返回系数可以用来调节从轴的返回速率，其取值范围为 40—500，取值越大，返回的速度就会越慢，取值越小返回的速度就会越快。所以当物料较短时，可适当调小此值，但是需要注意是否有可能造成伺服过速。

6.5.1.6 追剪参数生效

飞剪功能参数写入后，可选择与下一个凸轮周期生效，也可以选择下次启动电子凸轮时生效，生效的方式取决于 M 软元件。

生效方式						软元件状态	
轴 1	轴 2	轴 3	轴 4	Y0	Y1	ON	OFF
SM83	SM183	SM283	SM783	SM883	SM983	下一个凸轮周期生效	下次启动凸轮生效

当相应的 M 软元件设置为 ON 后，为下一个凸轮周期生效，当相应的 M 软元件设置为 OFF 后，为下次启动电子凸轮时生效。

当飞剪参数生效后，软件会自动将 SM82 软元件值 ON（该软元件对所有轴都通用）。

示例：

在运行的过程中修改切料长度，修改后置位相应的 M 软元件使得修改生效。运行示意图如下图所示：



从图中可以看出：

①这是一个长短料切换的示意图。切料长度从短料切换到长料，在短料运行时不存在等待周期，长料运行时存在一定的等待时间。

②由于切换的只是料的长度，所以对于同步区速率无影响，图中短料切换到长料后，只是主轴长度发生了变化，同步区速率未发生该表。

③切换发生在同步区起始点，以保证切换前后物料长度准确。

6.5.2 应用范例

6.5.2.1 基础案例

下面举一个实际的范例来说明追剪的功能，系统参数如下：

- 主轴直径：60mm
- 主轴减速比：1:10
- 从轴螺杆间距：20mm
- 从轴减速比：1:1
- 从轴可移动距离：400mm
- 从轴同步区长度：300mm
- 裁切长度：1320mm

计算追剪 CAMWR 指令所需要的参数：

（使用驱动器为 SD700 驱动器，输入 10000 个脉冲，电机转动一圈）

主轴一圈长度= $60 * \pi = 188.4$

主轴一圈脉冲数= $10000 * 10 = 100000$

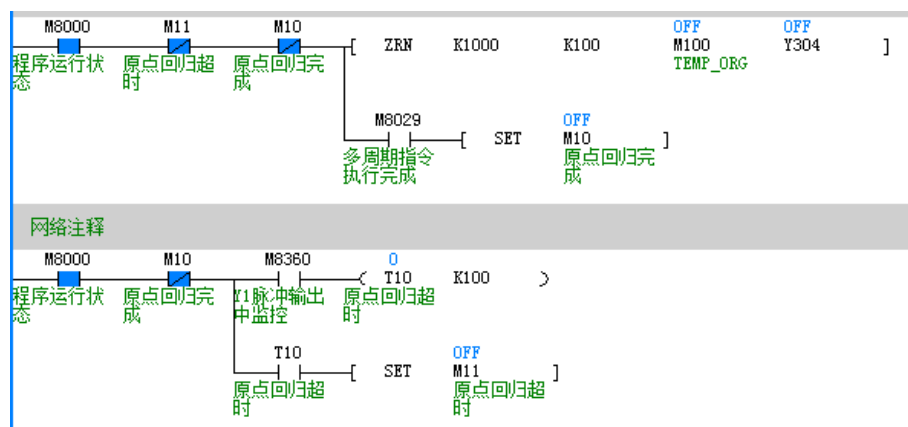
从轴一圈脉冲数= $10000 * 1 = 10000$

返回系数建议设置为 100

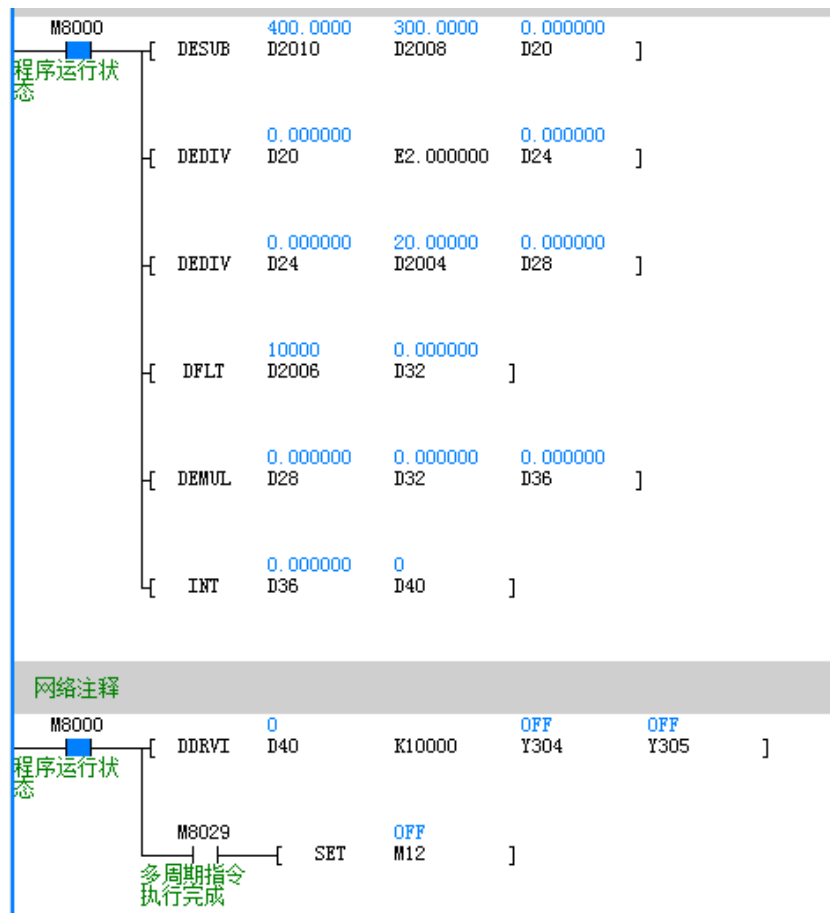
参数	单位	设定值
主轴一圈长度	mm	188.4
主轴一圈脉冲数	个	100000
从轴一圈导程	mm	20
从轴一圈脉冲数	个	10000
同步区长度	mm	300
从轴移动范围	mm	400
返回系数		100
料长	mm	1320

接下来我们将按照“2.1.3 追剪操作步骤”中的操作步骤来一步步的将我们的追剪系统运行起来。

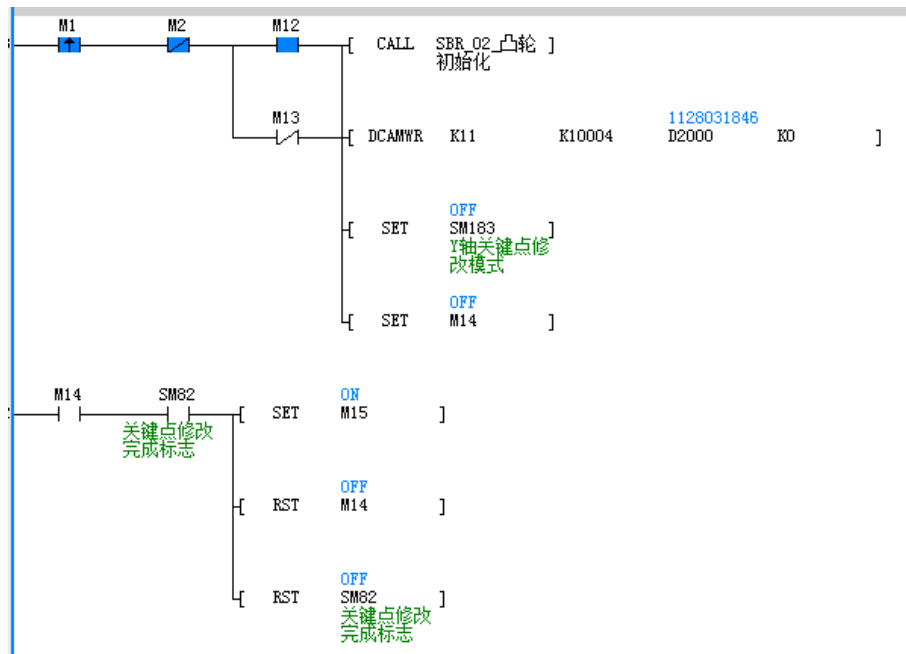
- 第一步：通过 ZRN 指令控制从轴回原点



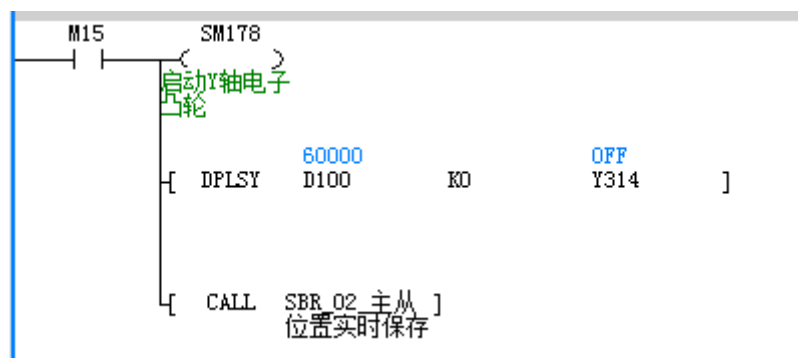
- 第二步：回完原点后，从轴前置



- 第三步：电子凸轮初始化，配置电子凸轮相关的参数；通过 DCAMWR 指令将追剪参数写入。



- 第四步：啮合电子凸轮，控制器控制主轴移动，从轴自动按照生成的电子凸轮曲线移动



整个的运行效果见图：



①为回原点的过程。这里采用 ZRN 指令回原点。

②为从轴前置过程。前置量=（从轴移动范围-同步区距离）/2。

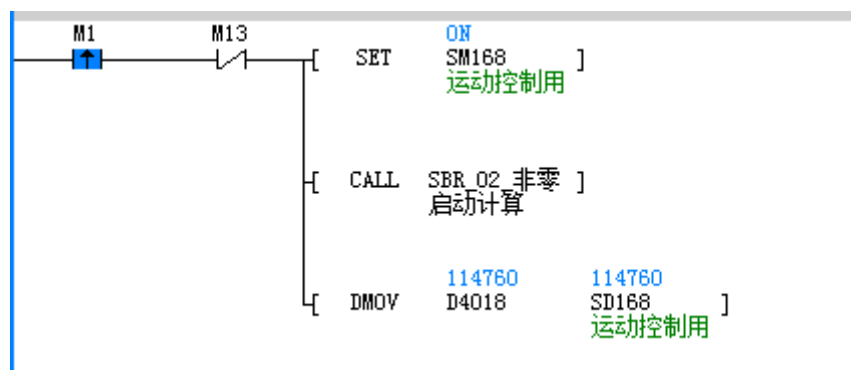
③为一个追剪周期。从图中可以看出一个追剪周期从同步区的起点开始，结束于同步区的起点。

注：过程①和过程②一般只需要执行一次，后续可将主从位置保存下来，上电后直接啮合凸轮。这样就不存在浪费物料的情况。

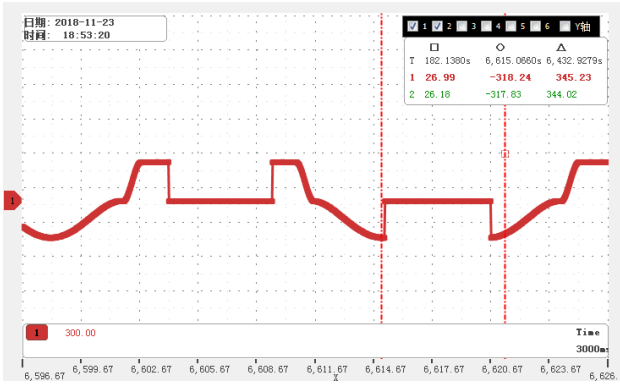
6.5.2.2 上电即运行案例

在追剪的运行过程中，如果需要上电之后不找原点直接啮合，那么可以通过电子凸轮非零启动功能来实现，具体实现原理参考说明书相关章节。

下面在上述范例的基础上，增加直接启动功能。梯形图实现如下所示：



运行结果如下图所示：

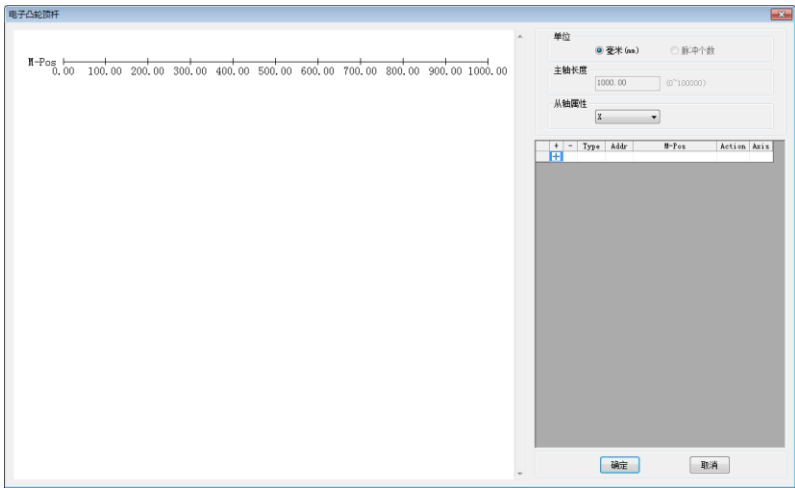


6.6 特殊功能

6.6.1 顶杆

顶杆功能可以实现位元件（M/Y）与电子凸轮主轴位置的配合，控制位元件随主轴位置变化而变化。顶杆的设置如下。

1、新建一个凸轮表在凸轮表中设置好主轴范围及单位，鼠标右击该凸轮表新建顶杆，双击打开电子凸轮顶杆编辑界面：



2、点击顶杆数据表增加或删除顶杆数据：

	+	-	Type	Addr	M-Pos	Action	Axis
	+	⊗	M	0			
		⊗			0.00	NA	X
		⊗			0.00	NA	X
	+	⊗	M	1			
		⊗			0.00	NA	X
	+						

3、设置顶杆数据：

M-Pos: 设置电子凸轮主轴位置；

Type: 设置位元件类型，支持 M 元件和 Y 元件；

Addr: 设置元件标号；

Action: 主轴位置等于 M-Pos 设置值时位元件的动作；NA 表示无动作；ON 表示置 ON；OFF 表示置 OFF；

INV 表示取反；

6.6.2 主从速比计算

该功能可以计算主从轴之间的速比，当设置完凸轮参数后，执行该功能，系统将计算出主从轴的最大和最小速比。

用户可以通过该功能计算出主从轴之间的速比，以验证凸轮点设置的是否合理。

用户也可以计算出来的速比对主轴速度进行限制，以防止从轴速度过速。

该功能的相关软元件如下表所示：

	轴 1	轴 2	轴 3	轴 4	Y0	Y1	说明
开启计算 ^{注1}	SM72	SM172	SM272	SM772	SM872	SM972	软元件置 ON，则进行主从速比的计算，计算完成后系统将自动将软元件置 OFF
最大速比 ^{注2}	SD74	SD174	SD274	SD774	SD874	SD974	主从最大速比，32 位浮点数
最小速比 ^{注2}	SD76	SD176	SD276	SD776	SD876	SD976	主从最小速比，32 位浮点数

注 1：将计算主从速比的 SM 软元件置为 ON 后，系统将开始计算主从最大/最小速比，计算完成后，系统将该软元件置为 OFF。

注 2：计算后的最大/最小速比。注意该软元件的数据类型为浮点型。

注 3：速比曲线不对称时速比计算不准确。

举例：

①若开启主从速比计算功能后，计算出来的主从最大速比为 10.0，如果从轴最大能接受到的脉冲频率为 100kHz，那么建议在使用时，主轴频率不超过 10kHz，否则有可能造成从轴过速。

②若开启主从速比计算功能后，计算出来的主从最小速比为-1.0，但是如果在加工的过程中不允许从轴出现反向，则说明设置的凸轮点不合理，需要调整凸轮点。

6.6.3 主轴区间位置监控

监控在一个凸轮周期，主轴所在的位置。

显示的是当前主轴的区间，[1,Len),其中 Len 为主轴长度。

	轴 1	轴 2	轴 3	轴 4	Y0	Y1	说明
主轴区间监控	SD85, SD86	SD185, SD186	SD285, SD286	SD785, SD786	SD885, SD886	SD985, SD986	单位：pls 32 位整型

6.6.4 主轴累积脉冲监控

凸轮运行时，主轴累积接收到的脉冲的个数可以通过 SD 元件获取。

	轴 1	轴 2	轴 3	轴 4	Y0	Y1	说明
主轴累积监控	SD48, SD49	SD148, SD149	SD248, SD249	SD748, SD749	SD848, SD849	SD948, SD949	单位：pls 32 位整型

注 1：如果是延时启动，延时启动的脉冲不计入。

注 2：内部 64 位计数，取低 32 位，比如 0x7fffffff 向上计数 0x80000000，0x80000000 向下计数 0x00000000。

6.6.5 电子凸轮方向设定

通过运行方式设定，可以设定从轴在不同主轴方向时的运行模式。其中沿凸轮曲线正运行表示从轴跟随主轴信号，沿设定凸轮曲线运行；沿凸轮曲线逆运行表示从轴跟随主轴信号，沿凸轮曲线反方向运行。

当主轴沿一方向运行一段距离再反向时，从轴沿设定凸轮曲线反向运行，即以凸轮曲线终点作为从轴运行起点，从轴逆凸轮曲线相对于凸轮曲线的终点运行。直到运行到主轴回到凸轮啮合位置，如果此时主轴继续方向运行从轴不动作并且报故障。

	轴 1	轴 2	轴 3	轴 4	Y0	Y1	说明
凸轮方向 设定 SD	SD84	SD184	SD284	SD784	SD884	SD984	0: 主轴方向为正时沿凸轮曲线正运行 1: 主轴方向为负时沿凸轮曲线正运行 2: 主轴方向为正时沿凸轮曲线逆运行 3: 主轴方向为负时沿凸轮曲线逆运行

注：在电子凸轮刚啮合位置如果伺服参数没调试好，会导致反馈脉冲波动，导致报故障。

6.6.6 编码器频率监控

监控每秒编码器频率，单位 pls。

	轴 1	轴 2	轴 3	轴 4	Y0	Y1	说明
编码器频率 监控	SD46, SD47	SD146, SD147	SD246, SD247	SD746, SD747	SD846, SD847	SD946, SD947	单位: pls 32 位整型

6.6.7 主轴位置计算（开发中）

1.指令形式

根据电子凸轮主轴位置计算主轴位置。

DCAMSP S1 S2 D				主轴位置计算	指令执行
S1	从轴位置				32位指令（13setp） 连续执行
S2	凸轮从轴轴号+1				
D	D+0: 总的点数 D+2: 点 1: 区间（区间从 1 开始） D+4: 点 1: 主轴位置 D+6: 点 2: 区间 D+8: 点 2: 主轴位置				

2.操作数

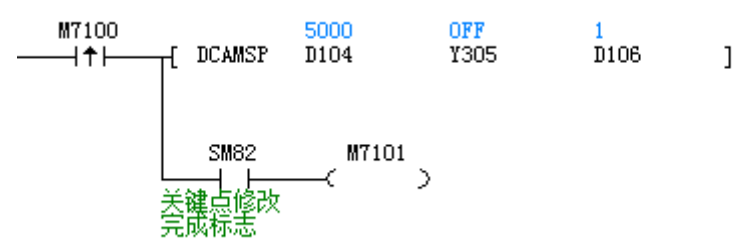
操作数	位软元件							字软元件															
	系统 用户							系统 用户					位数指定					变址			常数		实数
s1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
s2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E

注：带灰色底纹软元件表示不支持。

功能说明

- 凸轮必须被啮合过才能获取有效从轴位置。
- 开启功能，需要 SM82 清零，计算完成 SM82 自动置位。
- 尽量采用沿触发，防止多次计算。
- 与上一个计算的区别：凸轮从轴轴号+1。

例子：



	元件名称	数据类型	显示格式	当前值
1	D106	32位整数	十进制	1
2	D108	32位整数	十进制	1
3	D110	32位整数	十进制	5000

6.6.8 从轴位置计算

1. 指令形式

根据电子凸轮主轴位置计算从轴位置。

DCAMSP S1 S2 D				从轴位置计算	指令执行
S1	主轴位置				32位指令（13setp） 连续执行
S2	凸轮从轴轴号				
D	从轴位置				

2. 操作数

操作数	位软元件							字软元件															
	系统 用户							系统 用户				位数指定					变址		常数		实数		
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V	Z	修饰	K	H	E

注：带灰色底纹软元件表示不支持。

功能和动作说明：

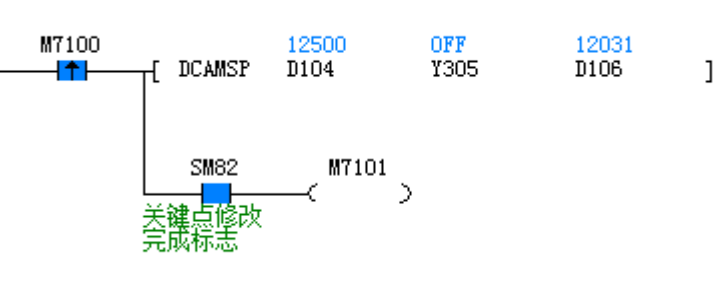
- S1: 凸轮主轴位置，单位为脉冲数，[S1]的设定值应该在凸轮一个周期的主轴范围内。
- S2: 凸轮从轴轴号，Y300、Y304、Y310、Y314分别对应轴1、轴2、轴3、轴4。
- D: 根据[S1]设定的主轴位置，计算出[S2]使用的凸轮表对应的从轴位置。

注：a.凸轮必须被啮合过才能获取有效从轴位置。

b.开启功能，需要 SM82 清零，计算完成 SM82 自动置位。

c.尽量采用沿触发，防止多次计算。

例子：



6.6.9 探针

1. 功能说明

探针功能为硬件实现。具备实时锁存数据特点。在总多场合下，能够显著提高系统的精度，并且具备相位的计算能够准确计算出偏差。

2. 操作寄存器

IN/OUT	说明	寄存器	长度	说明
输入	使能	M8150	BOOL	ON 使能；OFF 失能
	端口选择	D8150:bit[2:0]	3 位	0: X0;1: X1 2: EZ[0];3: EZ[1] 4: EZ[2];5: EZ[3]
	沿选择	D8150:Bit[3]	1 位	0:上升沿 1:下降沿
	轴设定	D8150:bit[6:4]	3 位	0~5: 对应轴 1~6
	逻辑与编码器选择	D8150:bit[7]	1 位	0: 逻辑位置 D8356 1: 编码器位置 SD56
输出	锁存位置	D8151: :32 位	32 位	相应数据
	锁存状态	M8151	BOOL	ON:有锁存；OFF 无锁存

3. 使用方法

- ① 设定模式。（开启使能后，在修改模式无效）
- ② 开启使能。
- ③ 实时读取锁存状态，如果为 ON，则读取到锁存信号。（可连续读取）
- ④ 关闭使能

注 1: V5 具备三个探针，具体下面寄存器列表

注 2: 关闭使能，锁存器的状态也会为 OFF。

注 3: 一直打开使能，则一直保持保持触发探针的功能，锁存位置将持续更新，状态也一直为 on。

注 4: 对应轴，需要添加 20，比如轴 1: D8356，轴 2: D8376

注 5: 相比 D8340（周期刷新），D8356 具有实时性，是硬件发送脉冲的实时数据，每发送一个脉冲，该数据都会变化。

三个探针各自对应的寄存器如下表所示

NO	说明	读/写	探针 1	探针 2	探针 3
1	使能	W	M8150	M8153	M8156
2	模式	W	D8150	D8153	D8156
3	锁存位置	R	D8151, D8152	D8154, D8155	D8157, D8158
4	锁存相位值	R	D8162, D8163	D8164, D8165	D8166, D8167
5	锁存状态	R	M8151	M8154	M8157

注：锁存相位值=当前锁存位置-上一刻锁存位置(第一次锁存不生效)

6.6.10 运动叠加

MOVIMPOS 运动叠加

1. 指令形式

运动脉冲叠加

MOVIMPOS S1 S2 S3 D					运动叠加	指令执行	
S1	脉冲数		设定叠加总脉冲数			16 位指令（9step） MOVIMPOS 连续执行	32 位指令（17step） DMOVIMPOS 连续执行
S2	频率		设定叠加脉冲频率				
S3	模式		叠加模式				
D	输出端口		设定叠加脉冲输出端口				

2. 操作数

操作数	位软元件							字软元件														
	系统·用户							系统·用户					位数指定					变址		常数		实数
S1	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S2	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
S3	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E
D	X	Y	M	T	C	S	SM	D	R	T	C	SD	KnX	KnY	KnM	KnS	KnSM	V, Z	修饰	K	H	E

注：带灰色底纹软元件，表示可支持。

功能说明：

- 当前脉冲输出基础上，按设定的叠加频率，叠加一定的脉冲。
- S1：叠加总脉冲数，16 位：-32767~32767，32 位：-2147483648~2147483647，符号表示叠加方向，为正时表示脉冲增加，为负时表示脉冲减少。注意：脉冲增加或减少只能在一个方向进行，脉冲减少不能改变当前脉冲输出指令的方向，最大减少到输出脉冲频率为 0；
- S2：叠加脉冲频率，16 位：1~65535，32 位：1~3M 单位为 Hz，实际可达的叠加频率可能和设定频率存在差异，实际可达的最大叠加加速度按照 1K/ms。实际端口输出的脉冲频率为：定位指令频率±叠加频率；
- S3：叠加模式，0：加速/减速模式；其它：保留；
- D：脉冲输出端口，轴 1、轴 2、轴 3、轴 4、Y0、Y1。

注意要点：

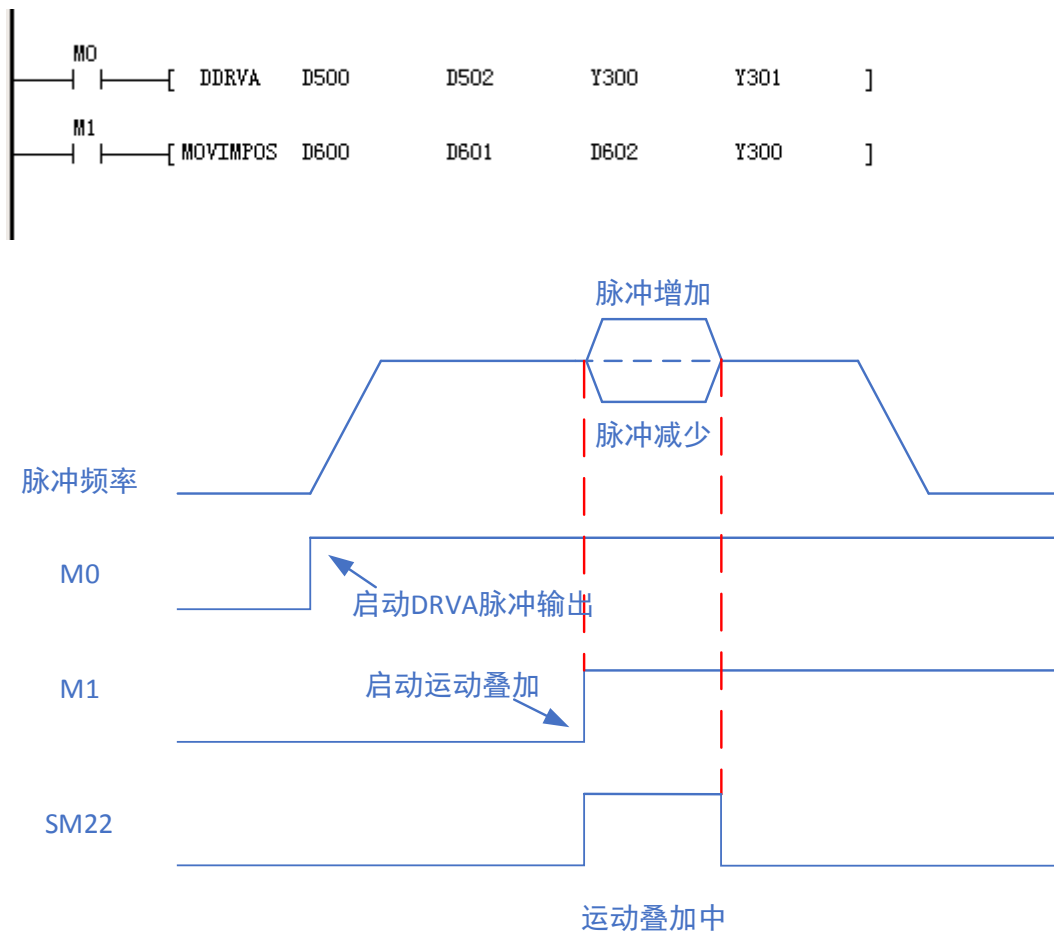
一个轴梯形图上只能使用一条叠加指令，否则会造成指令不能正常运行。

使用的特殊标志位，可以监控运动叠加状态：

轴 1	轴 2	轴 3	轴 4	Y0	Y1	功能	
SM22	SM122	SM222	SM722	SM822	SM922	运动叠加状态	ON:运动叠加中
							OFF: 运动叠加完成

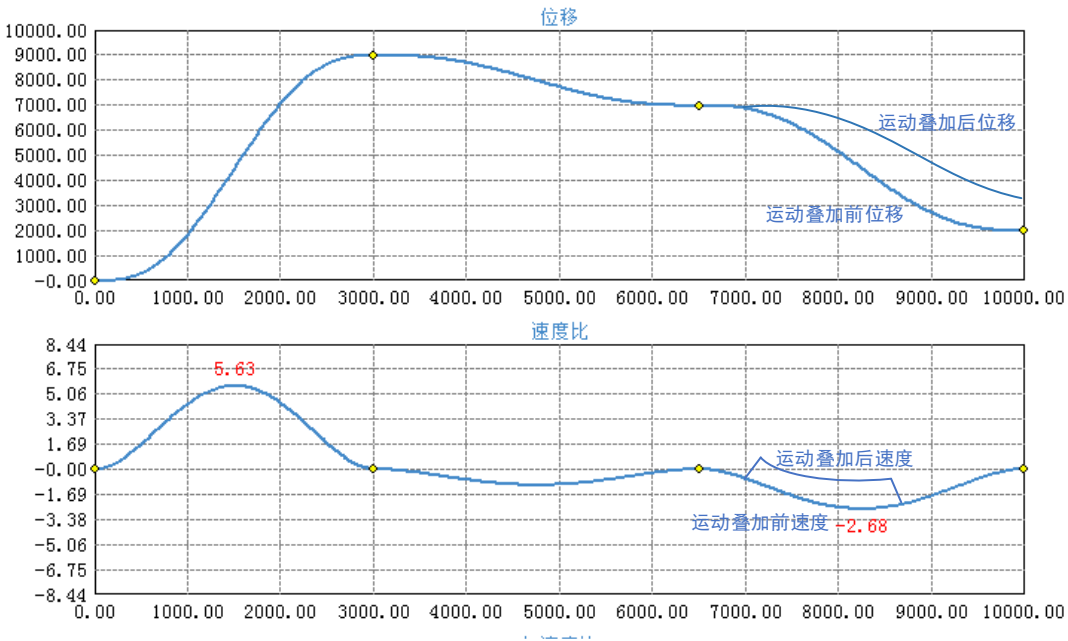
运动叠加启动后，脉冲输出端口在当前输出频率的基础上，按 1KHz/ms 的加速度改变当前输出频率，在当前输出频率的基础上增加或减少叠加频率，使输出脉冲总数增加或减少设定的脉冲个数。

【使用举例】



M0 启动 DRVA 定位指令输出脉冲后，置位 M1 启动运动叠加功能，可以改变叠加一个运动改变当前脉冲频率，增加或减少定位脉冲数。

在电子凸轮追标应用中，可以通过检测 2 个色标间的时间或距离，启动运动叠加功能，对送料轴实时补偿，提高追标精度。



6.6.11 控制周期

控制周期是在固定的时间计算一个数据，默认为 500，单位 us。修改改值能够影响到系统的响应时间，参数为 SD87,16 位。

设定范围 125,250,500,1000，分别对应 125us，250us，500us，1ms。其中，设定时可参考当前开销，最大开销来调节，建议设定不超过 70%。例如当前 SD88=100，那么 $100/0.7=142$ ，那么设定的数据，就设定 250,500,1000。

NO	参数变量	说明	位数
1	SD87	设定控制周期（R/W）	16 位
2	SD88	实时采样任务开销（R）	16 位
3	SD89	最大采样任务开销（R）	16 位

注：对于响应要求较高的场合，建议设定尽量小。

6.7 电子凸轮相关软元件一览

SD 软元件:

名称	轴 1	轴 2	轴 3	轴 4	Y0	Y1	含义
	D8340 D8341	D8360 D8361	D8380 D8381	D8400 D8401	D8420 D8421	D8440 D8441	内部逻辑位置
	SD61	SD161	SD261	SD761	SD861	SD961	脉冲输出设置
	SD56 SD57	SD156 SD157	SD256 SD257	SD756 SD757	SD856 SD857	SD956 SD957	外部编码器位置
	SD68 SD69	SD168 SD169	SD268 SD269	SD768 SD769	SD868 SD869	SD968 SD969	非零启动主轴位置
	SD70	SD170	SD270	SD770	SD870	SD970	凸轮表选择
	SD71	SD171	SD271	SD771	SD871	SD971	凸轮主轴选择
	SD72	SD172	SD272	SD772	SD872	SD972	凸轮设定循环次数
	SD73	SD173	SD273	SD773	SD873	SD973	凸轮从轴缩放比
	SD74 SD75	SD174 SD175	SD274 SD275	SD774 SD775	SD874 SD875	SD974 SD975	凸轮最大速率比(浮点数)
	SD76 SD77	SD176 SD177	SD276 SD277	SD776 SD777	SD876 SD877	SD976 SD977	凸轮最小速率比(浮点数)
	SD78 SD79	SD178 SD179	SD278 SD279	SD778 SD779	SD878 SD879	SD978 SD979	延时启动脉冲数
	SD80	SD180	SD280	SD780	SD880	SD980	预留
	SD81 SD82	SD181 SD182	SD281 SD282	SD781 SD782	SD881 SD882	SD981 SD982	凸轮已执行周期数
	SD83	SD183	SD283	SD783	SD883	SD983	预留
	SD84	SD184	SD284	SD784	SD884	SD984	凸轮方向设定
	SD85 SD86	SD185 SD186	SD285 SD286	SD785 SD786	SD885 SD886	SD985 SD986	凸轮主轴区间长度监控

SM 软元件

名称	轴 1	轴 2	轴 3	轴 4	Y0	Y1	含义
	SM68	SM168	SM268	SM768	SM868	SM968	凸轮非零启动使能
	SM70	SM170	SM270	SM770	SM870	SM970	凸轮触发模式
	SM71	SM171	SM271	SM771	SM871	SM971	凸轮主轴选择
	SM72	SM172	SM272	SM772	SM872	SM972	凸轮速比计算使能
	SM73	SM173	SM273	SM773	SM873	SM973	凸轮周期模式使能
	SM74	SM174	SM274	SM774	SM874	SM974	预留
	SM75	SM175	SM275	SM775	SM875	SM975	凸轮延时启动使能
	SM76 SM77	SM176 SM177	SM276 SM277	SM776 SM777	SM876 SM877	SM976 SM977	预留
	SM78	SM178	SM278	SM778	SM878	SM978	凸轮使能
	SM79	SM179	SM279	SM779	SM879	SM979	凸轮周期循环完成

	SM80	SM180	SM280	SM780	SM880	SM980	凸轮停止状态监控
	SM81	SM181	SM281	SM781	SM881	SM981	凸轮停止模式
	SM82						关键点修改完成标志
	SM83	SM183	SM283	SM783	SM883	SM983	凸轮关键点修改模式
	SM84	SM184	SM284	SM784	SM884	SM984	预留
	SM88	SM188	SM288	SM788	SM888	SM988	
	SM89	SM189	SM289	SM789	SM889	SM989	凸轮完成标志

第7章 中断

7.1 概述

7.1.1 概述

不受主程序的运算周期影响，将中断功能作为触发信号，立即执行中断程序(中断子程序)的功能。

在一般的顺控程序处理中，由运算周期造成的延迟以及时间的偏差给机械动作带来影响，这样的情况可以得到改善。

7.1.2 中断类型

		概述	
I	中断	外部中断	X0~X2 输入中断，编号 I00□，I10□，I20□，3 点，(□表示：0 下降沿中断，1 上升沿中断)。沿中断禁止标志位寄存器置 ON 后，则对应的输入中断被禁止
		定时中断	I6□□，I7□□，I8□□，3 点(□□=1~99，时基=1ms)
		脉冲完成中断	I502~I504，3 点

上位机		V5 控制器
I001	X 轴 PG0 上升沿	X0 上升沿
I101	Y 轴 PG0 上升沿	X1 上升沿
I201	Z 轴 PG0 上升沿	X2 上升沿
I000	X 轴 PG0 下降沿	X0 下降沿
I100	Y 轴 PG0 下降沿	X1 下降沿
I200	Z 轴 PG0 下降沿	X2 下降沿
I6□□	定时器中断 0	定时器中断 0
I7□□	定时器中断 1	定时器中断 1
I8□□	定时器中断 2	定时器中断 2
I502	X 轴输出完成中断	Axis1 输出完成中断
I503	Y 轴输出完成中断	Axis2 输出完成中断
I504	Z 轴输出完成中断	Axis3 输出完成中断

7.2 外部中断

7.2.1 概述

使用输入 X000~X002 的输入信号，执行中断子程序。

其中 X0,X1 为高速输入，X2 为低速输入。

由于可以不受可编程控制器的运算周期的影响处理外部输入信号，所以适用于执行高速控制和获取短时间脉冲。

7.2.2 外部中断类型

输入编号	中断编号		禁止中断
	上升沿中断	下降沿中断	
X0	I001	I000	M8050
X1	I101	I100	M8051
X2	I201	I200	M8052

注：当 M8050~M8052 为 ON 时，其各自对应的输入编号的中断事件被禁止。

外部中断事件注意要点：

禁止输入的重复使用：用作为中断指针的输入继电器的编号，请勿与使用相同输入范围的“高速计数器”、“脉冲捕捉功能”、“脉冲密度”等应用指令重复。

关于输入滤波器的自动调节：指定了输入中断指针 I□0□后，输入继电器的输入滤波器会被自动更改为高速读取用。因此，不需要使用 REFF 指令和特殊数据寄存器 D8020(输入滤波器的调节)更改滤波器的调节。

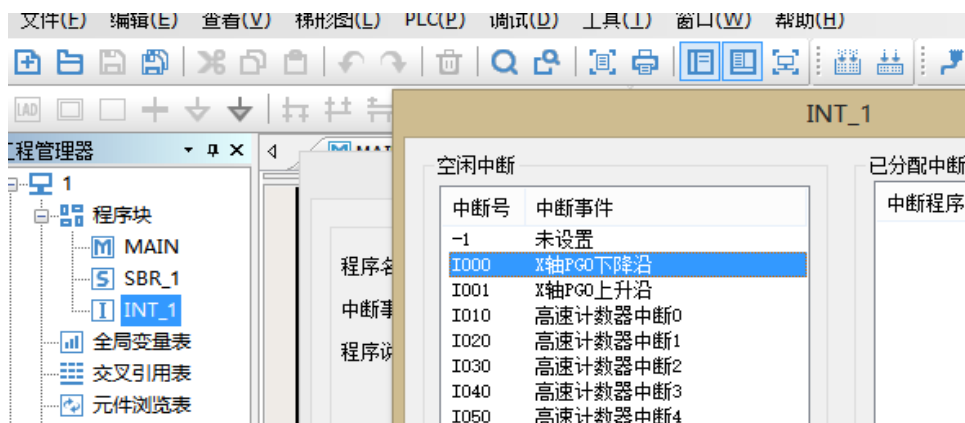
此外，没有作为输入中断指针使用的输入继电器的输入滤波器以 10ms(初始值)动作。

指针编号的重复使用：对像 I001 和 I000 等那样的同一输入的上升沿中断和下降沿中断，不能被同时编写。

7.2.3 使用举例

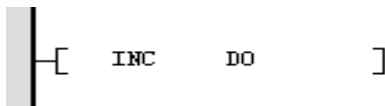
对外部输入中断 X0 下降沿进行计数的程序

(1) 建立 X0 的下降沿中断子程序，编号 I000

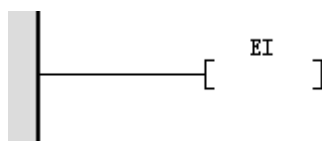


(2) 中断子程序内容

当 X0 来一个下降沿时，进入中断子程序 D0 加 1



(3) 在主程序中开启对应的 X0 中断允许



7.3 定时器中断

7.3.1 概述

不受可编程控制器运算周期的影响，每隔 1ms~99ms 执行一次中断程序。

适用于当主程序的运算周期较长时，对特定程序进行高速处理，或者需要在顺控运算时间隔一定时间执行程序时的情况。

7.3.2 定时器中断类型

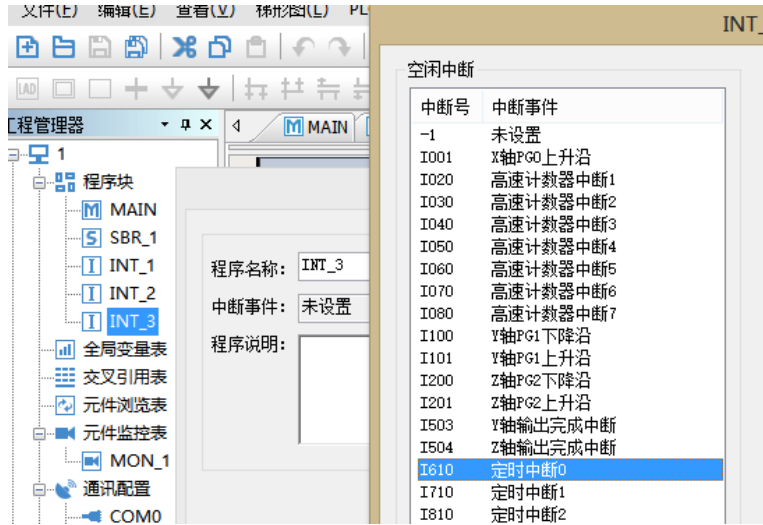
指针编号	中断周期	禁止中断
I6□□	在指针名的□□中，输入 1~99 的整数。 例如:I650=每 50ms 的定时器中断	M8056
I7□□		M8057
I8□□		M8058

当 M8056~M8057 为 ON 时，其各自对应的输入编号的中断事件被禁止；
指针编号(I6、I7、I8)不能重复使用。

7.3.3 使用举例

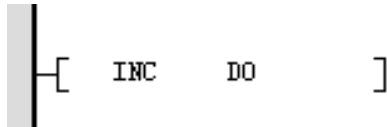
每隔 50ms 对数据 D0 加 1。

(1) 建立 50ms 的定时中断子程序，编号 I650。



(2) 中断子程序内容

每隔 50ms 进入一次定时器中断，进入中断子程序 D0 加 1



(3) 主程序开中断 EI



7.4 脉冲完成中断

7.4.1 概述

通过使用特殊位 M8352、M8372、M8392、M8412、M8432、M8452（分别对应 Axis1，Axis2，Axis3，Axis4，Y0，Y1）为 ON 时，在 PLSY、PLSR、DRVA、DRVI 等定位指令中可以实现脉冲输出完成中断。

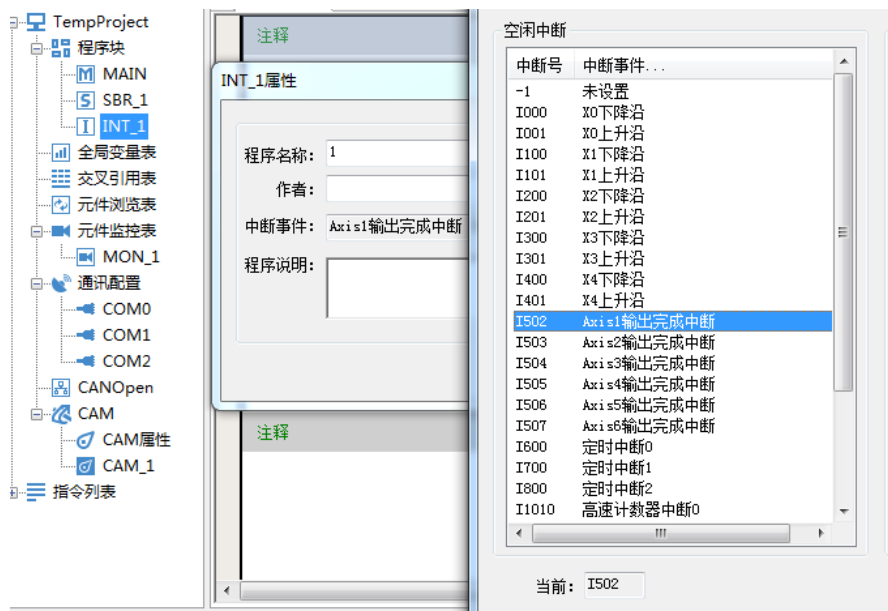
7.4.2 脉冲完成中断类型

其脉冲完成中断对应关系如下：

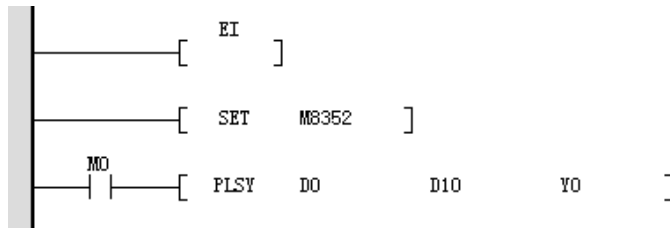
	中断对象	与 V5 对应关系
Axis1	Axis1 输出完成中断	Axis1 脉冲输出中断 I502
Axis2	Axis2 输出完成中断	Axis2 脉冲输出中断 I503
Axis3	Axis3 输出完成中断	Axis3 脉冲输出中断 I504
Axis4	Axis4 输出完成中断	Axis4 脉冲输出中断 I505
Y0	Y0 输出完成中断	Y0 脉冲输出中断 I506
Y1	Y1 输出完成中断	Y1 脉冲输出中断 I507

7.4.3 使用举例

(1) 建立 Y300 高速输出口脉冲完成中断，编号 I502



(1) 主程序开启中断

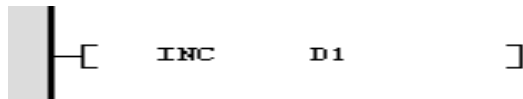


开启全局中断 EI；

M8352 为 ON 开启 Y0 脉冲输出完成中断标志；

在 Y0 口输出以 D0 为频率,以 D10 为指定的脉冲个数。

(2) 脉冲完成中断子程序



第8章 通信

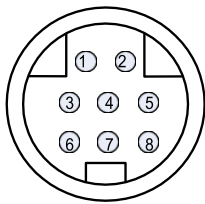
8.1 概要

V5 运动控制模块自带，CAN 通信，支持 CANopen 协议，MODBUS 协议。

包含 3 个独立物理串行通信口，分别为 COM0 和 COM1 和 COM2。COM0 具有编程、监控功能；COM1 包含 RS485 通信功能完全由用户自由定义、CAN 通信功能；COM2 为网口，支持 CAN/RS485(232)通信。

8.2 介绍

COM0 硬件为标准 RS422，接口端子为 8 孔鼠标头母座。接口定义如下：



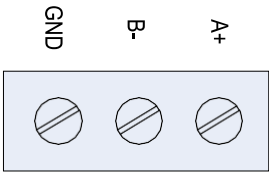
管脚号	信号	描述
1	RXD-	接收负
2	RXD+	接收正
3	GND	地线，9、10 没有电气连接
4	TXD-/RXD-	对外发送负，若为 RS485，也作接收负
5	+5V	对外供电+5V，与内部用的逻辑+5V 相同
6	CCS	通信方向控制线，高电平表示发，低电平表示收，在串口作 RS485 时由 PLC 控制 4、7 脚是接收还是发送。若为 RS422 时固定 4、7 脚一直处于发送
7	TXD+/RXD+	对外发送正，若为 RS485，也作接收正
8	NC	空脚

通过 COM0 口，V5 与计算机或触摸屏的连接有两种方式。

①方式 1：V5 侧为 RS422，计算机侧为 USB。计算机通过专用的 USB 下载电缆连接到 COM0 的程序下载口。

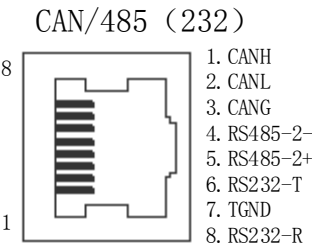
②方式 2：V5 侧为 RS422，计算机侧为 RS232，计算机通过专用的串口下载电缆连接到 COM0 的程序下载口。

COM1 为 RS485，接口为接线端子，接口定义：



COM1 与其它设备通信连接方式。通过接线端子，用户现场配线。

COM2 为 CAN/RS485（232），接口定义如下：



RS232-R、RS232-T	RS232 通信信号（不能与 RS485-2 同时使用）
TGND	RS232/RS485-2 参考地
RS485-2+、RS485-2-	第二组 RS485 通信信号线
CANH、CANL、TGND	CAN 通信线，其参考电平为 TGND，多台使用时注意请确保各个 TGND 互相连接

8.3 通信协议设置说明

8.3.1 COM0 协议配置

工程管理器中的“通讯配置”右键点击 COM0，点击打开：



COM0 的通讯参数配置如下图：



COM0 协议：默认为下载/HMI 监控协议

COM0 协议	D8116	半双工/全双工模式	COM0 通信格式
下载/HMI 监控协议	非 02h	不支持	固定
MODBUS-RTU 从站	02h	半双工	由 COM0 配置决由 D8110 查看。

COM1 协议配置

COM1 配置如图：



协议切换条件：系统上电运行或用户程序的停机状态 COM1 都可以切换协议，运行状态中不对协议进行更改。

注：COM1 做主站时，需将 M8260 标志位复位。

M8260=OFF，COM1 做主站有效。

M8260=ON，COM2 做主站有效。

COM1 协议：

COM1 协议	D8126	半双工/全双工模式	COM1 通信格式
MODBUS-RTU /QLINK 从站	02h	半双工	由 COM1 配置决定，由 D8120 查看。
MODBUSRTU 主站	20h	半双工	

8.3.2 COM2 协议配置

COM2 配置如图：



协议切换条件：系统上电运行或用户程序的停机状态 COM2 都可以切换协议，运行状态中不对协议进行更改。

注：COM2 作为主站时，需将 M8260 标志位置位。

M8260=OFF，COM1 做主站有效。

M8260=ON，COM2 做主站有效。

COM2 协议：

COM2 协议	D8266	半双工/全双工模式	COM2 通信格式
MODBUS-RTU /QLINK 从站	02h	半双工	由 COM1 配置决定，由 D8260 查看。
MODBUSRTU 主站	20h	半双工	

8.3.3 串口通信格式

完成通信配置后，系统会自动生成通信格式，具体格式定义如下表：

端口	对应的 D 元件
COM0	D8110
COM1	D8120
COM2	D8260

协议与通信格式对照表

协议名称	波特率 (Bit7-Bit4)	停止位 (Bit3)	校验位 (Bit2-Bit1)	数据位 (Bit0)
MODBUS-RTU 从站	0011b:300Bits/s 0100b:600Bits/s 0101b:1200Bits/s 0110b:2400Bits/s0111b:4800			0b:7Bits 1b:8Bits
MODBUS-RTU 主站	Bits/s 1000b:9600Bits/s 1001b:19200Bits/s 1010b:38400Bits/s 1011b:57600Bits/s 1100b:115200Bits/s	0:1Bits 1:2Bits	00b: 无校验(N) 01b: 奇校验(O) 11b: 偶校验(E)	注: MODBUS-RTU 从站协议及主站只 支持 8 位数据位, 否 则将造成通信出错

8.3.4 串口通信格式软元件一览表

COM0:

M8110	保留	D8110	通信格式
M8111	保留	D8111	通信站号
M8112	Modbus-通信执行状态	D8112	下载、HMI监控协议-通信格式
M8113	Modbus-通信错误标志	D8113	保留
M8114	保留	D8114	保留
M8115	保留	D8115	保留
M8116	保留	D8116	通信协议
M8117	保留	D8117	保留
M8118	保留	D8118	Modbus-通信出错站号
M8119	超时判断	D8119	通信超时时间

COM1:

M8120	保留	D8120	通信格式
M8121	保留	D8121	通信站号
M8122	Modbus-通信执行状态 RS 指令-发送标志	D8122	保留
M8123	保留	D8123	保留
M8124	保留	D8124	保留
M8125	保留	D8125	保留
M8126	保留	D8126	通信协议
M8127	保留	D8127	保留
M8128	保留	D8128	保留
M8129	保留	D8129	通信超时时间

COM2:

M8260	切换主站 M8260=0:COM1 主站 M8260=1:COM2 主站	D8260	通信格式
M8261	保留	D8261	通信站号
M8262	Modbus-通信执行状态 RS 指令-发送标志	D8262	保留
M8263	保留	D8263	保留
M8264	保留	D8264	保留
M8265	保留	D8265	保留
M8266	保留	D8266	通信协议
M8267	保留	D8267	保留
M8268	保留	D8268	保留
M8269	保留	D8269	通信超时时间

8.3.5 通信错误码一览表

并行连接通信出错 M8063(D8063)继续运行		
0000	无异常	
6301	奇偶出错超过出错成帧出错	检查双方的可编程控制器的电源是否为 ON，适配器和控制器之间，以及适配器之间连接是否正确。
6302	通信字符有误	
6303	通信数据的和数不一致	
6304	数据格式有误	
6305	指令有误	
6306	监视定时器溢出	
6307~6311	无	
6312	并行连接字符出错	
6313	并行连接和数出错	
6314	并行连接格式出错	
6330	MODBUS 从站地址设置错误	COM0 通信出错请检查 COM0 的通信电缆是否正确连接。
6331	数据帧长度错误	
6332	地址错误	
6333	CRC 检验错误	
6334	不支持的命令码	
6335	接收超时	
6336	数据错误	
6337	缓冲区溢出	
6338	帧错误	
6339	串口协议错误	
6340	MODBUS 从站地址设置错误	COM1 通信出错，请检查 COM1 的通信电缆是否正确连接； 检查通信双方通信格式是否匹配；
6341	数据帧长度错误	
6342	地址错误	
6343	CRC 检验错误	
6344	不支持的命令码	
6345	接收超时	
6346	数据错误	
6347	缓冲区溢出	
6348	帧错误	
6349	串口协议错误	

8.4 HMI 监控协议

1. 硬件连接

通过 COM0 通信，连接方式为 RS422，只能通过下载口连接。

2. 软件配置



COM0配置对话框，用于配置HMI监控协议的通信参数。

协议类型

- ☒ 下载/HMI监控协议 (RS422)
- ☐ MODBUS-RTU从站 (RS485)

参数配置

波特率	115200
数据位	7位
校验位	偶
停止位	1位
站号	1 (1~247)
通讯超时	10 *10ms (1~255)

底部按钮：读取配置、写入配置、确定、取消

3. 协议说明

HMI 监控协议为 V5 运动控制器内部协议，用 VCAutoDesignsoft 软件与 V5 通信，VCAutoDesignsoft 通过该协议，可以擦除、读取和下载用户程序；可以对 V5 实施遥测、遥调与遥控。具体为可监测 V5 中任意元件的状态，可强制更改任何元件，还可以控制 V5 的启动和停止。

8.5 MODBUS 协议

8.5.1 MODBUS 协议说明

MODBUS 通信的底层为 RS485 信号，采用双绞线进行联接就可以了，因此传输距离较远，可达 1000 米，抗干扰性能比较好，且成本低，在工业控制设备的通信中被广泛使用，现在众多厂家的变频器、控制器都采用了该协议。

传送数据格式有 HEX 码数据和 ASCII 码两种，分别称为 MODBUS-RTU 和 MODBUS-ASC 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 协议的通信效率较高，处理简单，使用得更多。

MODBUS 为单主多从通信方式，采用的是主问从答方式，每次通信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。

8.5.2 MODBUS 功能码及数据编址

MODBUS 协议功能码 0x01, 0x03, 0x05, 0x06, 0x0f, 0x10; 通过这些功能码，可读写的线圈有 M, S, T, C, X (只读), Y 等变量；寄存器有 D, T, C。

MODBUS 通信主机在访问（读取或改写）PLC 从机的内部变量时，必须遵循如下的通信命令帧定义，以及变量地址索引方法，才能进行正常的通信操作。当通信出现错误时，见“(g) 错误响应帧”。

1. MODBUS 帧格式（以 MODBUS-RTU 为例）

◆ 功能码 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（功能码）	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（功能码）	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，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

N 最大为 255。

◆ 功能码0x02 (02): 读线圈

请求帧格式: 从机地址+0x02+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	0x02 (功能码)	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+0x02+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	0x02 (功能码)	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, 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

N 最大为 255。

◆ 功能码 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 (功能码)	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 (功能码)	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

N 最大为 125。

◆ 功能码 0x04 (04): 读寄存器

请求帧格式: 从机地址+0x04+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	0x04 (功能码)	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+0x04+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	0x04 (功能码)	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

N 最大为 125。

◆ 功能码 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 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	0x05 (功能码)	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。FF00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 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	0x05 (功能码)	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。FF00即为有效
5	CRC校验	2个字节	高位在前, 低位在后

◆ 功能码 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 (功能码)	1个字节	写单个寄存器
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。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 (功能码)	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

◆ 功能码 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 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	0x0f (功能码)	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: $[(N+7)/8]$
6	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
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 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	0x0f (功能码)	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

N 最大为 255。

◆ 功能码 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 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	0x10 (功能码)	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 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	0x10 (功能码)	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

N 最大为 123。

◆ 错误响应帧

错误响应: 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
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

2. 变量编址

■ 线圈地址

线圈：指位变量，只有两种状态 0 和 1。在 V5 中包含 M, S, T, C, X, Y 等变量。

变量名称	起始地址	线圈数量
M0~M7679	0x0000 (0)	7680
M8000~M8511	0x1F40 (8000)	512
SM0~SM1023	0x2400 (9216)	1024
S0~S4095	0xE000 (57344)	4096
T0~T511	0xF000 (61440)	512
C0~C255	0xF400 (62464)	256
X0~X377	0xF800 (63488)	256
Y0~Y377	0xFC00 (64512)	256

■ 寄存器编址

寄存器：指 16 位或 32 位变量，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~D8511	0 (0)	8512	16位寄存器
SD0~SD1023	0x2400	1024	16位寄存器
R0~R32767	0x3000	32768	16位寄存器
T0~T511	0xF000 (61440)	512	16位寄存器
C0~C199	0xF400 (62464)	200	16位寄存器
C200~C255	0xF700 (63232)	56	32位寄存器

注意：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 0xF70A (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) 功能码。

8.5.3 MODBUS 通信地址

V5 运动控制器作为 MODBUS 从站时，软元件地址如下：

1. 字型变量寄存器的地址

表示 16 位（字）或 32 位（双字）变量，这些变量类型的首地址如下表，各寄存器的具体地址（首地址+变量序号）如下：

变量名称	起始地址	寄存器数量	说明
D0~D8511	0x0000（0）	8512	16位寄存器
SD0~SD1023	0x2400（9216）	1024	16位寄存器
R0~R32767	0x3000（12288）	32768	16位寄存器
T0~T511	0xF000（61440）	512	16位寄存器
C0~C199	0xF400（62464）	200	16位寄存器
C200~C255	0xF700（63232）	56	32位寄存器

注意：通过 MODBUS 访问 32 位寄存器时，一个寄存器作两寄存器看待。如要读或写 C200-C205 这 5 个寄存器，MODBUS 地址为 0xF700，寄存器的数量为 10（5*2）。

2. 位变量的线圈地址

位变量也称线圈，如 M/S/T/C/X/Y 等变量，只有 0 或 1 两种状态。这些为变量地址如下，具体地址（首地址+变量序号）。

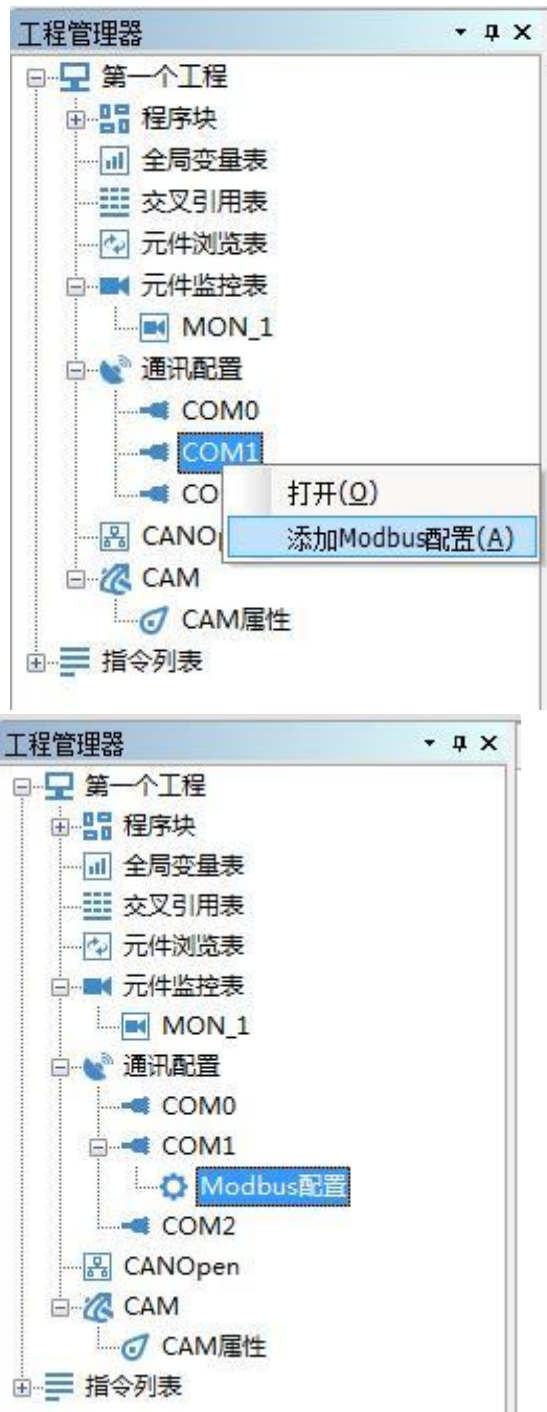
变量名称	起始地址	线圈数量
M0~M7679	0x0000（0）	7680
M8000~M8511	0x1F40（8000）	512
SM0~SM1023	0x2400（9216）	1024
S0~S4095	0xE000（57344）	4096
T0~T511	0xF000（61440）	512
C0~C255	0xF400（62464）	256
X0~X377	0xF800（63488）	256
Y0~Y377	0xFC00（64512）	256

8.5.4 MODBUS 的配置使用说明

Modbus 配置可实现对 Modbus 通讯数据的集中配置,将配置数据下载到 Modbus 主站,PLC 单板软件根据配置信息实现与从站设备通讯,从而达到数据交换的目的;使用 Modbus 配置降低了 Modbus 的使用难度、减少了工作量、提升了用户体验的效果。

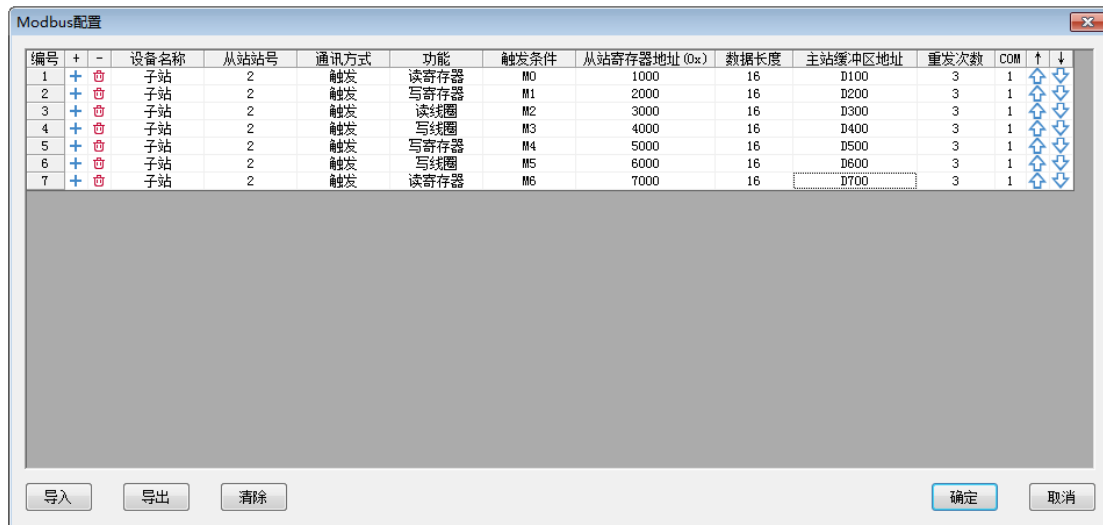
■ 创建 Modbus 配置

在工程管理器中,鼠标右键“通讯配置”节点下的“COM1”或“COM2”节点,在弹出的菜单中选择“添加 Modbus 配置”,即可添加 Modbus 配置节点,如下图所示:



■ Modbus 配置数据添加

鼠标左键双击“Modbus 配置”节点，即可打开 Modbus 配置界面，对相关参数进行设置如下图：



界面操作说明：

按钮	图标	功能
新增	+	在下一行增加一条新的配置记录
删除	-	删除当前行的配置记录
上移	↑	将当前行向上移动一行，第一行上移时位置不变
下移	↓	将当前行向下移动一行，最后一行下移时位置不变
导入		从Excel导入数据到界面，方便数据处理
导出		导出界面数据到Excel，方便数据处理
清除		清除窗口中所有配置信息

配置数据说明：

配置参数	数据说明
设备名称	自定义文字信息，不会下载到PLC，只是为了界面上方便区分主从站
从站站号	十进制表示的从站站号，范围为（1至247）
通讯方式	通讯方式分为“触发”、“循环”； 触发：“触发条件”置“ON”通讯操作被触发；触发元件类型：M、S； 循环：“触发条件”为“空”，通讯操作被循环执行；
功能	4种功能：读寄存器、写寄存器、读线圈、写线圈
触发条件	通讯方式为“触发”时，可输入触发条件元件，元件类型为M、S。
从站寄存器地址	通讯操作从站端起始操作地址，用十六进制表示
数据长度	读寄存器、写寄存器数据长度表示寄存器个数，即“从站寄存器地址”开始的n个寄存器；读线圈、写线圈数据长度表示线圈个数，即“从站寄存器地址”开始的n个线圈；
主站缓冲区地址	主、从站交换数据时主站端缓存区；缓存区为PLC的D元件，元件个数有数据长度决定；
重发次数	通讯方式为“触发”时，发送次数
COM	主站 Modbus 通讯串口

■ MODBUS 做主站时的配置表的协议设置

除了配置 Modbus 配置参数外,还需要对 Modbus 配置所对应的 COM 端口进行参数配置,COM1 和 COM2 都可以作为 Modbus 主站的通讯端口。假如 Modbus 主站的通讯端口为 COM1 口,在“工程管理器”中,双击“通讯配置”节点下的“COM1”节点或右键“COM1”节点选择“打开”,进入“COM1 配置”界面,如下图所示:



COM1配置对话框的初始状态。对话框标题为“COM1配置”。顶部有一个复选框“启用通讯参数”，当前未勾选。下方是“协议类型”区域，包含两个下拉菜单，分别显示“自由协议”和“RS485”。再下方是“配置”区域，包含以下参数：波特率（9600）、数据位（未显示）、校验位（无）、起始符（2）、停止位（1位）、结束符（3）、站号（1，范围1~247）、通讯超时（10，单位*10ms，范围1~255）。底部有四个按钮：“读取配置”、“写入配置”、“确定”和“取消”。

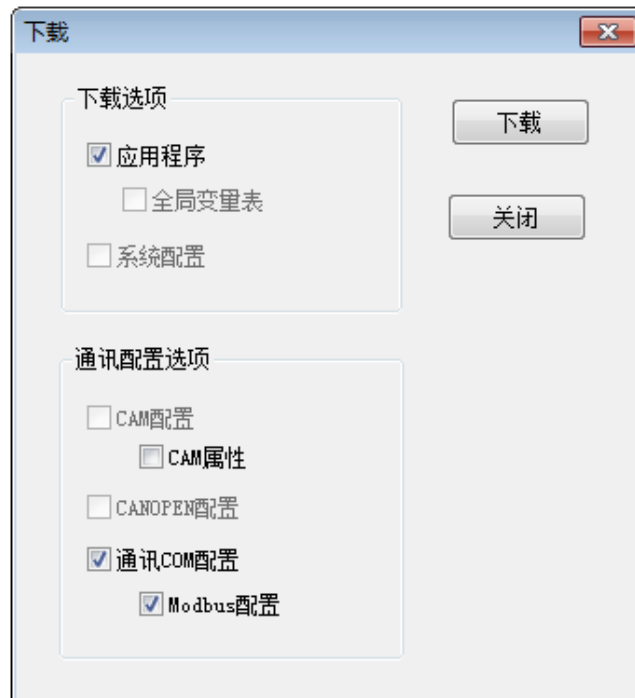
然后选择“启用通讯参数”选项,协议类型选择“MODBUS-RTU 主站”,可修改需要的“波特率”、“校验位”、“停止位”、“通讯超时”等参数,主站站号默认使用“1”号,修改完成的界面如下图所示,可以点击“读取配置”或“写入配置”来进行配置的读取和设置,点击“确定”按钮完成主站通讯设置。



COM1配置对话框修改后的状态。对话框标题为“COM1配置”。顶部复选框“启用通讯参数”已被勾选。下方“协议类型”区域的下拉菜单显示为“MODBUS-RTU主站”，右侧的“RS485”下拉菜单保持不变。再下方“配置”区域的参数如下：波特率（9600）、数据位（8位）、校验位（无）、起始符（2）、停止位（2位）、结束符（3）、站号（1，范围1~247）、通讯超时（10，单位*10ms，范围1~255）。底部按钮布局与初始状态一致。

Modbus 主站配置下载:

点击“PLC”菜单中的“下载”菜单，或使用快捷键“F8”，或点击“PLC 工具栏”的下载按钮（）打开下载窗口，选择“通讯 COM 配置”和“Modbus 配置”，点击“下载”按钮即可完成 Modbus 主站通讯协议配置下载，如下图所示：



注意：通讯协议如果在之前已经设置完成，只需下载“Modbus 配置”即可。

■ MODBUS 的从站配置

除了设置主站通讯配置外,还需要对从站通讯参数进行相应的配置,COM1 和 COM2 都可以作为 Modbus 从站的通讯端口。假如 Modbus 从站的通讯端口为 COM1 口,在“工程管理器”中,双击“通讯配置”节点下的“COM1”节点或右键“COM1”节点选择“打开”,进入“COM1 配置”界面,如下图所示:



COM1配置对话框的初始状态。对话框标题为“COM1配置”。顶部有一个复选框“启用通讯参数”，当前未勾选。下方是“协议类型”区域，包含两个下拉菜单，分别显示“自由协议”和“RS485”。再下方是“配置”区域，包含以下参数：波特率（9600）、数据位（未指定）、校验位（无）、起始符（2）、停止位（1位）、结束符（3）、站号（1，范围1~247）、通讯超时（10，单位*10ms，范围1~255）。底部有四个按钮：“读取配置”、“写入配置”、“确定”和“取消”。

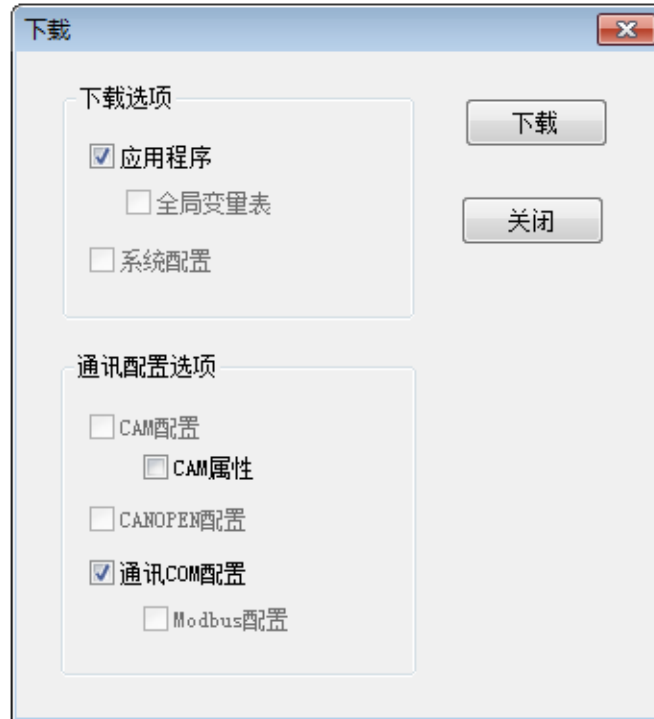
然后选择“启用通讯参数”选项,协议类型选择“MODBUS-RTU/QLINK 从站”,可修改需要的“波特率”、“校验位”、“停止位”、“站号”、“通讯超时”等参数,这里设置从站站号为“2”号,修改完成的界面如下图所示,点击“确定”按钮完成从站通讯设置:



COM1配置对话框修改后的状态。对话框标题为“COM1配置”。顶部复选框“启用通讯参数”已被勾选。下方“协议类型”区域的下拉菜单显示为“MODBUS-RTU/QLINK从站”，右侧的“RS485”下拉菜单保持不变。再下方“配置”区域的参数已修改：波特率（9600）、数据位（8位）、校验位（无）、起始符（2）、停止位（2位）、结束符（3）、站号（2，范围1~247）、通讯超时（10，单位*10ms，范围1~255）。底部按钮保持不变。

Modbus 从站配置下载:

从站无“Modbus 配置”数据,在“下载”窗口中选择“通讯 COM 配置”,点击“下载”按钮即可完成 Modbus 从站通讯协议配置下载,如下图所示:

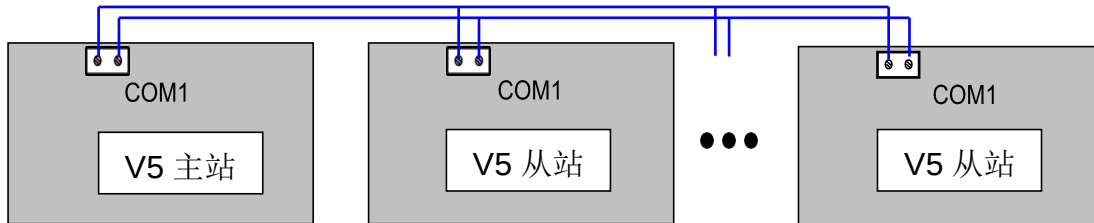


8.5.5 MODBUS 指令使用说明

■ MODBUS 主站的通信应用

在 V5 运动控制器的 COM1 通信口可以设置为 MODBUS-RTU 或 MODBUS-ASC 主站。

(1) 硬件连接



(2) 软件设置



设置为 MODBUS-RTU 主站，波特率为 9600，校验位为无，停止位设为 2，站号为 1，通信超时设为 10，设置好后点击确定。

(3) 协议说明

MODBUS 指令对串口 COM1 有效，用户可通过 MODBUS 指令编程，把 V5 作为主站与 MODBUS 从站设备通信。

MODBUS 指令可以同时存在多条并且全部被驱动，系统内部会协调指令的顺序执行，MODBUS 协议要求无论写还是读，从站均需要有应答（广播除外）。一条 MODBUS 指令可能需要执行较长时间，一般需要多个扫描周期。在一个扫描周期内，指令被驱动，但不一定被执行。

若存在多条 MODBUS 指令，其执行顺序是这样的：从开机开始，扫描第一条被驱动的 MODBUS 指令，若扫描到，把该 MODBUS 的参数记录下来，在后台执行。执行完后，返回用户程序，从刚执行的 MODBUS 指令位置开始扫描下一条被驱动的 MODBUS 指令并执行，周而复始。

指令格式：MODBUS(S1, S2, n, D)

①S1：从机地址和 MODBUS 功能码，高 8 位表示从机地址，即目标设备地址。低 8 位表示 MODBUS 功能码，由标准 MODBUS 协议定义，目前支持功能码有 0x01, 0x02, 0x03, 0x04, 0x05, 0x06, 0x0f, 0x10。具体含义请参照标准 MODBUS 协议或目标设备 MODBUS 协议。

②S2：所要读或写的从机线圈（1 位）或寄存器（16 位）地址，取值参考从机 MODBUS 协议。可为元

件或常数。

③n: 所要读写的从机线圈或寄存器个数, 可为元件或常数。

④D: 只能为 D 元件。本机用于存放数据的起始寄存器, 即数据缓冲区。缓冲区长度与 RELEN 相关, 至少取 1。若 MODBUS 命令为读, 指令成功执行完后, 把从机数据读到缓冲区中, 若 MODBUS 命令为写, 把缓冲区发送给从机。用户在设计程序时需要计算缓冲区长度, 预留足够的寄存器作缓冲区。

相关状态标志

①M8122: MODBUS 指令执行状态指示, OFF 时表示指令执行完毕, ON 时为执行中。若 M8122 为 OFF, 且指令在一个扫描周期内能流有效, M8122 置为 ON, 系统将会把指令参数记录下来, 转入后台执行该指令的通信要求。通信执行完后, 当再次运行到此指令的位置时, 无论该指令能流是否有效, 均会把 M8122 复位为 OFF, 立即扫描下一条能流有效的指令, 记录指令参数并转入后台执行该指令的通信要求。

②M8123: 指令通信情况指示, ON 表示通信异常, OFF 表示通信正常。

③M8063: 指令错误指示, 错误码存于 D8063。

④D8063: 错误码 (见通信错误码一览表)。

注意:

读写寄存器时, V5 满足 MODBUS 协议标准要求, 最大支持一次读 125 个寄存器, 写 123 个寄存器。若读写超出最大值, 将报参数错。

读写线圈时, V5 满足 MODBUS 协议标准要求, 最大支持一次读 255 个线圈, 写 255 个线圈。若读写超出最大值, 将报参数错。

(4) 例一：读从机地址为 100 的寄存器，数据存放在 D100 中。



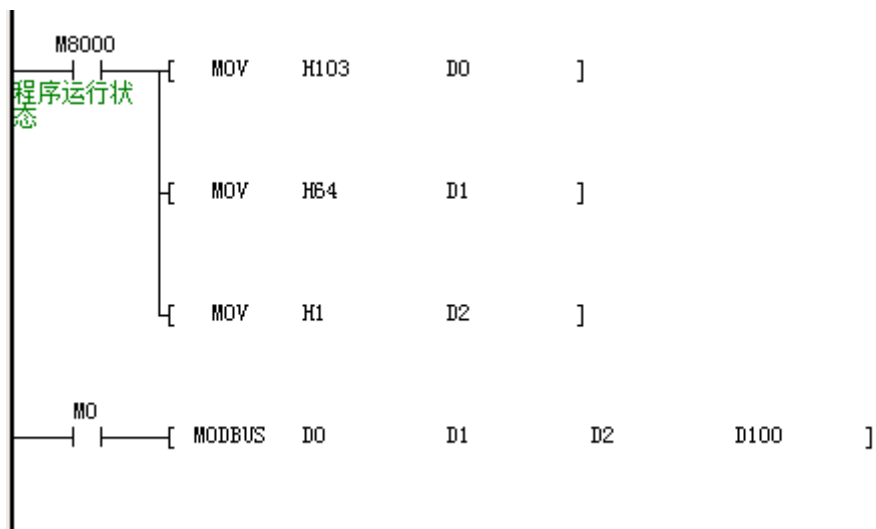
D0=H0103 从机地址为 01，功能码为 03；

D1=H0064 要操作的从机的寄存器地址；

D2=H0001 要操作的寄存器个数；

D100 数据缓冲区，将读到的数据存放在 D100 中。

梯形图如下：



当 M0 为 ON 时，读取从机地址为 100 的寄存器，存放在 D100 中。

通过 COM1 发送以下一帧数据（16 进制）：010300640001D5E5

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；

0064：所要读从机寄存器地址，D1 的值；

0001：所要读的寄存器个数，D2 的值；

D5E5：CRC 校验码。

若从机也是 V5 运动控制器，设定为 MODBUSRTU 从站协议，梯形图如下：



从站配置如下：



从机正确的相应数据帧（16 进制）：0203020520F2C3

从机把 D100（寄存器地址为 H0064）发给主机：

02：代表从机地址；

03：MODBUS 命令码；

02：表示回复 2 个字节的有效数据；

0520：寄存器数据，即 D100 的值；

F2C3：CRC 校验码。

■ MODBUS 从站通信应用

有些工业应用中，V5 运动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、运行组态软件的工业 PC 等，作为监控主机，以 MODBUS 主站协议与 V5 运动控制器等设备进行通信，此时 V5 的通信口需要以 MODBUS 从站协议与上位机通信。V5 运动控制器内置有 MODBUS-RTU 从站协议，并且在 COM1 和 COM2 口都可以运行该协议。

1. COM1 从站的配置



COM1配置对话框，用于配置 COM1 从站的通信参数。对话框包含以下配置项：

- ☒ 启用通讯参数
- 协议类型：MODBUS-RTU/QLINK从站
- RS485
- 配置：
 - 波特率：9600
 - 数据位：8位
 - 校验位：无
 - 起始符：2
 - 停止位：2位
 - 结束符：3
 - 站号：21 (1~247)
 - 通讯超时：10 *10ms (1~255)
- 底部按钮：读取配置、写入配置、确定、取消

2. 协议说明

MODBUS 从站协议包括 MODBUSRTU 协议（以下简称 RTU 协议）和 MODBUSASC 协议（以下简称 ASC 协议），两者区别在与数据链路，通信传送的数据 RTU 协议为真实数据，ASC 协议传送的数据是转换为 ASC 码的数据。另外两者在帧结构上也有区别，RTU 协议是以时间来区分数据帧的，若通信中有 3.5 个字节的时间没有接收到数据，则认为对方数据传送完毕；ASC 协议即是以 ASC 码“:”为帧起始符，以\CR\LF(0D0Ah)为帧结束符，从通信效率来看，RTU 协议高于 ASC 协议，大概 RTU 协议大概为 ASC 协议到两倍。

8.6 CANopen 通讯

8.6.1 概述

1、CAN 是控制器局域网网络(ControllerAreaNetwork,CAN)的简称,是一种串行通信方式,通信波特率可达到 1Mbps。CANopen 是 CAN 网络的一种应用层协议,CANopen 应用层通讯协议规范称为 CiA301,也称 DS301。针对个别设备的子协定义 CiA301 为基础再进行扩充,如针对 I/O 模组的 CiA401 及针对运动控制的 CiA402。

2、V5 运动控制器支持 CANopen 协议,只支持主站模式。通过组态 CANopen 配置界面,设置主站波特率和节点号。理论支持 126 个从站节点,实际应用中,最大节点数量取决于所用 CAN 收发器的性能,目前可带 10 个从站。

型号	V5-MC104
支持 CANopen 协议	DS301V4.02
最大 TPDO/RPDO 数量	64
从站节点数	10
支持的波特率与对应的通讯距离	1Mbps/25m 800Kbps/50m 500Kbps/100m 250Kbps/250m 125Kbps/500m 100Kbps/600m 50Kbps/1000m 20Kbps/2500m 10Kbps/6700m
从站 PDO 映射寄存器范围	D0-D7999

3、CANopen 通讯对象种类及说明

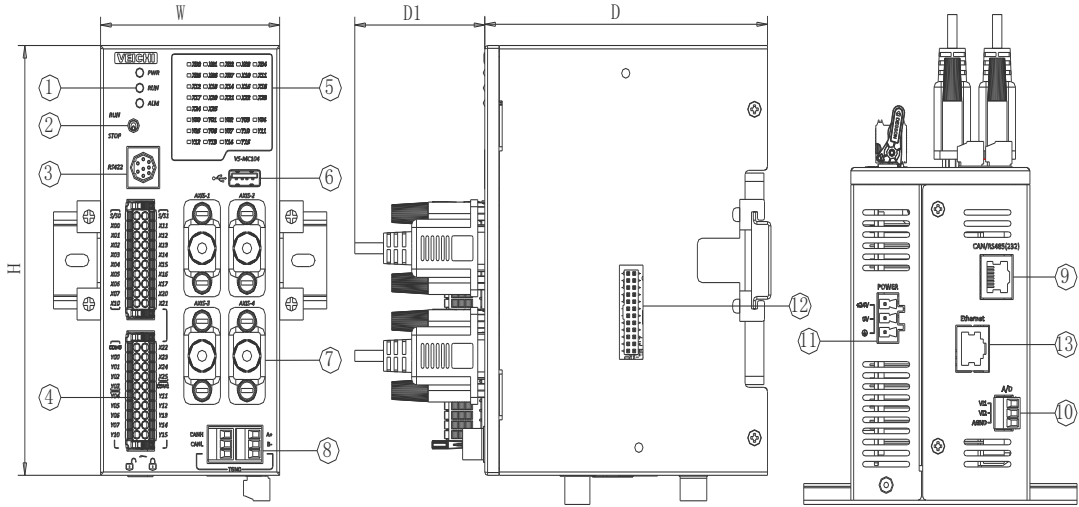
对象	CAN-ID	说明
NMT 网络管理命令	000h	主机通过 NMT 报文管理从站,进行“启动节点”、“停止节点”等相应操作
Sync 同步报文	080h	通过同步信号,让所有节点能在同一时刻进行上传数据或者执行下达的应用指令
Emergency 紧急报文	080h+从站节点号	当 CANopen 节点发生错误时,节点会发送一帧紧急报文。当主站收到此报文后,会进行相应的处理。
PDO 过程数据对象	见表 1	用于主站对从站的读写操作
SDO 服务数据对象	“答”: 580h+从站节点号 “问”: 600h+从站节点号	主要对从站进行参数设置时使用。对从站对象字典内容进行配置时,一问一答形式

表 1:

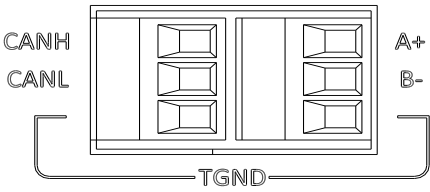
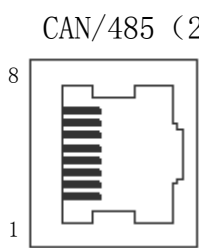
对象	CAN-ID
TPDO1	181h-1FFh (180h+node-ID)
RPDO1	201h-27Fh (200h+node-ID)
TPDO2	281h-2FFh (280h+node-ID)
RPDO2	301h-37Fh (300h+node-ID)
TPDO3	381h-3FFh (380h+node-ID)
RPDO3	401h-47Fh (400h+node-ID)
TPDO4	481h-4FFh (480h+node-ID)
RPDO4	501h-57Fh (500h+node-ID)

8.6.2 硬件配置

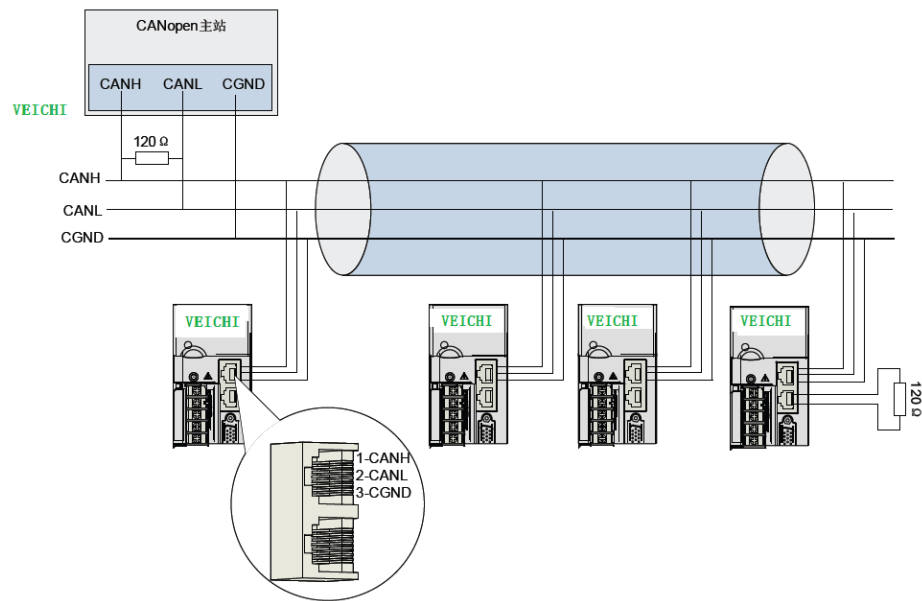
1、V5 控制器 CAN 通讯接口有两种形式：端子（下图序号 8）和 RJ45（下图序号 9）。两都内部接线是相通的，两者只能选其一使用，内部已接有 120 欧电



2、端子说明：

名称	功能描述	示意图
RS485+ RS485-	第一组 RS485 通信信号线	
CANH CANL	CAN 通信线	
TGND	RS485/CAN 参考电平 TGND，多台使用时注意请确保各个 TGND 互相连接	
RS232-R RS232-T	RS232 通信信号（不能与 RS485-2 同时使用）	 CAN/485 (232) 1. CANH 2. CANL 3. CANG 4. RS485-2- 5. RS485-2+ 6. RS232-T 7. TGND 8. RS232-R
TGND	RS232/RS485-2 参考地	
RS485-2+ RS485-2-	第二组 RS485 通信信号线	
CANH CANL TGND	CAN 通信线，其参考电平为 TGND，多台使用时注意请确保各个 TGND 互相连接	

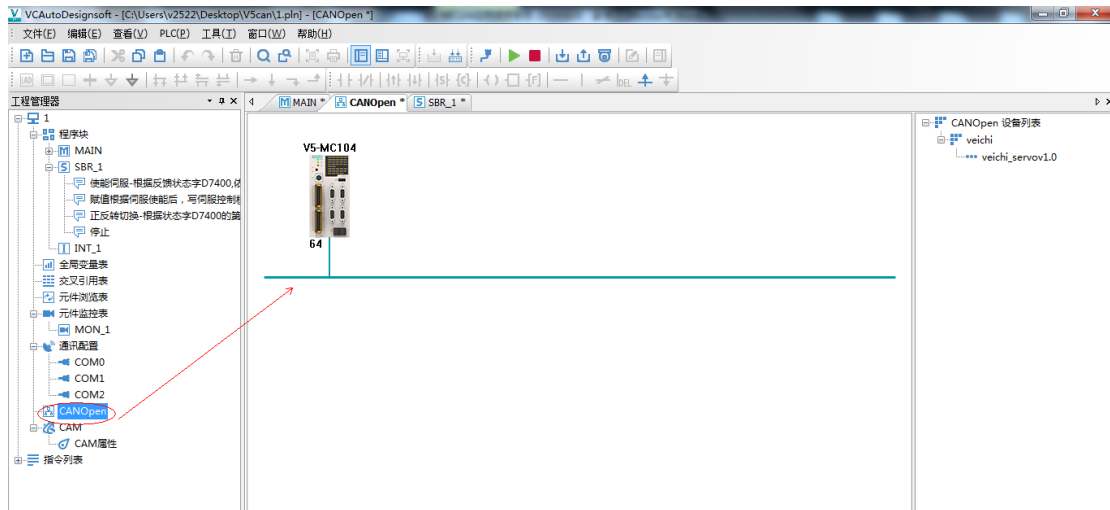
3、接线说明（为提高通讯抗干扰性，保证通讯质量从站终端要并联 120 欧电阻）：



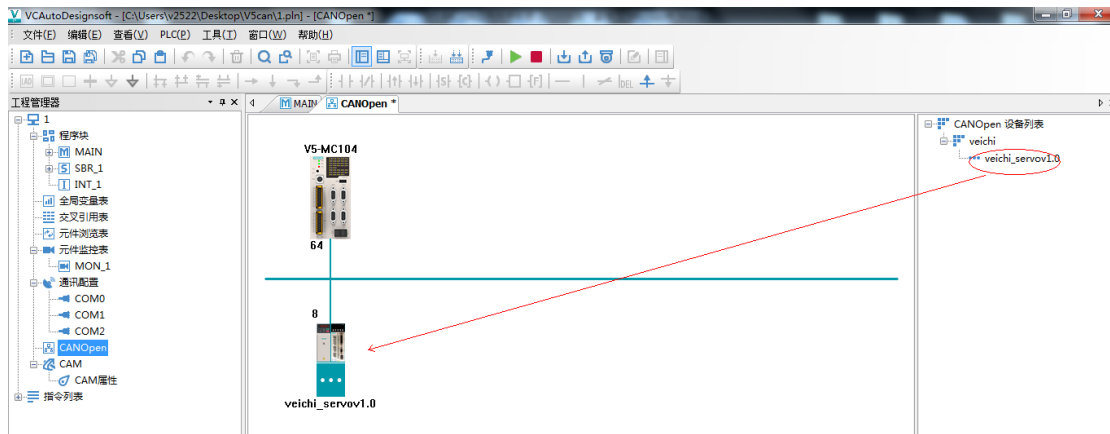
8.6.3 创建 CANopen 配置

CANopen 功能用于配置 CANopen 通信，包括 1 个 CANopen 主站，负责管理网络中的所有从站，从站数量最多为 126 个，每个设备都有一个独立的节点地址(Node-ID)。

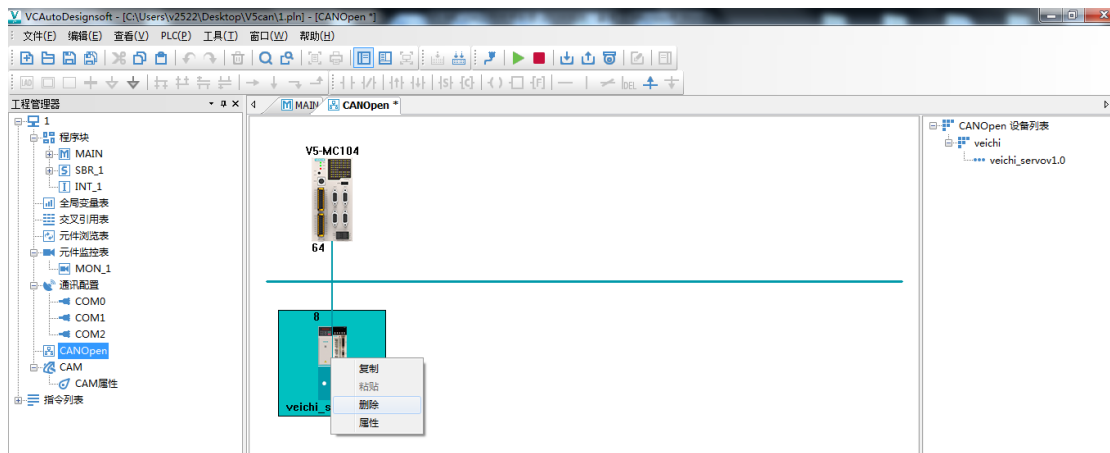
1、鼠标左键双击“工程管理器”中的“CANopen”节点，或鼠标右键打开菜单选择“打开”，打开如下配置界面：



2、添加从站：在窗口右侧的“CANopen 设备列表”中选中设备双击即可添加从站，如下图所示：



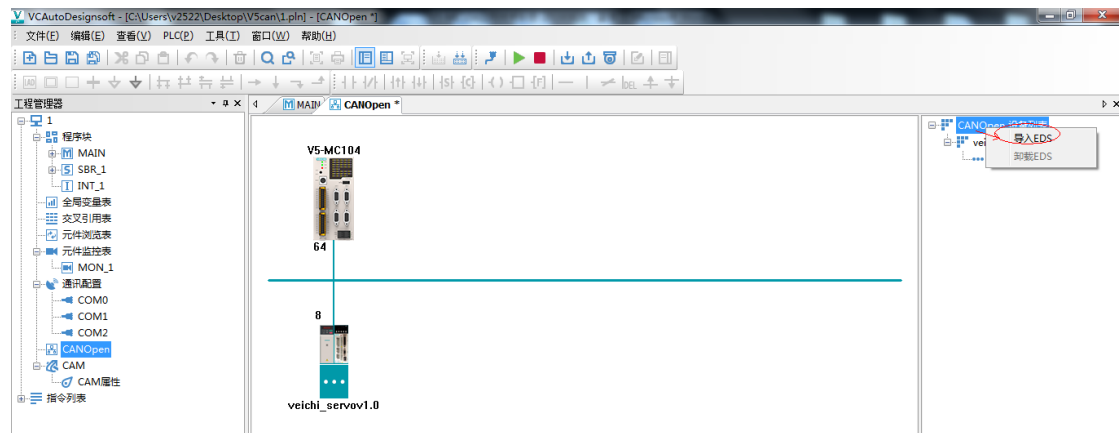
3、删除从站：在组态窗口中选中从站，单击右键弹出菜单执行“删除”即可删除选中从站，如下图所示：



4、复制、粘贴

在组态窗口选中“从站”，单击右键在弹出菜单上执行“复制”项，在空白处单击右键在弹出菜单上执行“粘贴”项实现。

5、添加新的 EDS 文件



8.6.4 主站配置

1、在窗口中选中“V5-MC104”主站图片，双击鼠标左键或单击鼠标右键弹出菜单执行“属性”进入主站配置界面，如下图所示：

主站参数说明	
参数	功能
主站节点 ID	设置主站站号, 当此站号与 PLC 本身站号相同时, 此 PLC 被初始化成 CANopen 主站。
波特率	主站生效的通信波特率。
使能同步生产	勾选此项, 本站将会按照“同步周期(ms)”设置的时间循环发送同步帧。
COB-ID	同步帧发送 ID, 此项使用默认值 0x80, 不允许设置。
同步周期(ms)	发送同步帧的循环周期。默认 200, 单位 ms。
使能心跳生产	勾选此项, 本站将按照“生产时间(ms)”设置的时间循环发送心跳帧。
生产时间(ms)	发送心跳的循环周期。默认 300, 单位 ms。
程序运行过程中禁止 SDO、NMT 访问	勾选此功能后, 运行过程中将不能使用在线调试功能。此功能仅针对后台软件的限制。
所有 SDO 错误继续配置	勾选此功能后, 如果出现 SDO 配置错误 (校验错误除外), 将继续进行配置。此功能对所有从站都有效。不勾选此功能, 发生 SDO 错误时, 主站将广播复位从站。

SDO 超时时间	默认 500，单位 ms。SDO 帧主要作为网络配置。SDO 在重发 3 次没有按时收到返回帧，主站判定配置超时。每帧的等待间隔时间为此时时间。
使能站点监控	勾选此项主站将会把从站状态写入对应设置的寄存器，此项默认勾选。
监控寄存器起始地址	默认值为 7800。即设置 D7800 为设备状态监控起始地址。D7800 为主站状态，D（7800+从站站号）为相应从站状态。状态值意义如下：0 为初始状态，4 为停止状态，5 为运行状态，127 为预运行状态，255 为掉线状态。如果相应的从站不存在，那么相应的寄存器也不会被更新。例如，3 号站不存在，D7803 的数据不会被更新。从站需设置心跳或者节点保护功能，此功能才有意义。因为此状态是由从站的心跳或者节点保护帧反馈。

2、节点列表配置

自动分配：勾选此功能，主从站数据交互的寄存器地址将自动分配；不勾选此功能，用户需手动设置数据交互的起始地址（单独设置每一个 PDO 的起始地址），此功能默认勾选。

从站接收映射寄存器起始地址：自动分配主站发送的数据起始地址。

从站发送映射寄存器起始地址：自动分配主站接收数据的起始地址。

1、自动分配时，选择可用节点，单击向右箭头，自动分配地址。

配置列表

未配置节点:

Node-ID	设备名称

已配置节点:

Node-ID	设备名称
8	veichi_servov1.0

接收PDO映射寄存器				发送PDO映射寄存器			
地址变量	名称	索引	长度	地址变量	名称	索引	长度
D7000, D7001	target_position...	607A:0	32	D7400	status word	6041:0	16
D7002	control word	6040:0	16	D7401, D7402	Position ABS Value	6064:0	32
D7003, D7004	profile_velocity	6081:0	32	D7403	Error Code	603F:0	16
D7005_L	modes of operation	6060:0	8				
D7006	profile_acceler...	6083:0	16				
D7007	profile_deceler...	6084:0	16				

II、取消自动分配时，需手动输入地址，在已添加的节点地址表中，双击要手动分配的 D 寄存器，弹出映射寄存器设置界面，手动输入要分配的地址，注意手动分配时，地址不要重复。

主站

主站配置 节点列表配置

☐ 自动分配PDO映射寄存器 从站接收映射寄存器起始地址 (D) 7000

重设PDO映射寄存器 从站发送映射寄存器起始地址 (D) 7400

配置列表

未配置节点:

Node-ID	设备名称

已配置节点:

Node-ID	设备名称

接收PDO映射寄存器

地址变量	名称	索引	长度
D7000, D7001	target_position...	607A:0	32
D7002	control word	6040:0	16
D7003, D7004	profile_velocity	6081:0	32
D7005_L	modes of operation	6060:0	8
D7006	profile_acceler...	6083:0	16
D7007	profile_deceler...	6084:0	16

映射寄存器设置

映射索引 (0x) 1600

映射名称 Receive PDO1 mapping

元件类型 D 映射长度 48

起始地址数值 7000 元件使用个数 3

映射起始地址 D7000 映射结束地址 D7002

确定 取消

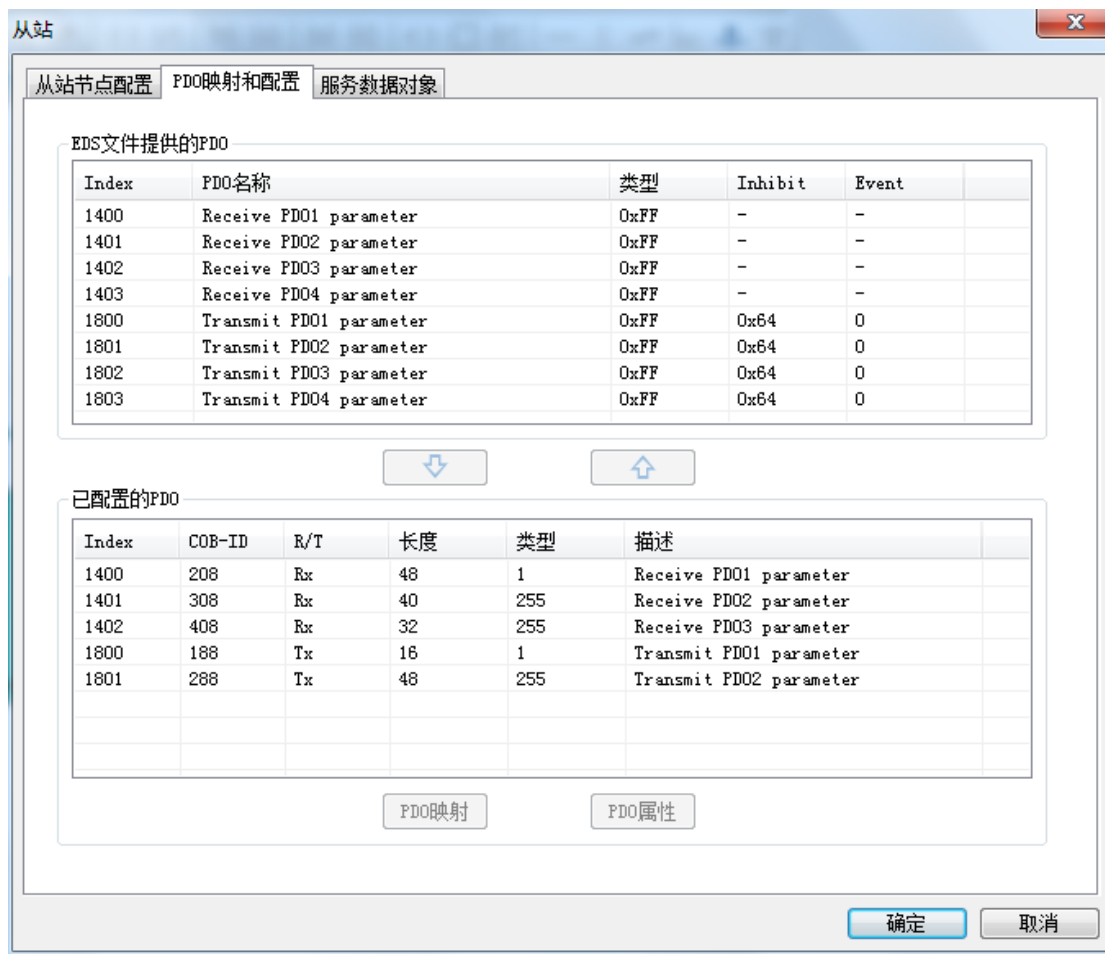
8.6.5 从站配置

1、在窗口双击从站图标，进入从站配置界面，如下图所示：

从站参数说明	
参数	功能
从站节点 ID	设置从站站号, 当此站号与 PLC 本身站号相同时, 此 PLC 被初始化成 CANopen 从站。
不初始化	选择此功能后, 此从站将不进行初始化配置 (在使用默认配置情况下才可以选择)。默认不勾选。
出厂设置	勾选此功能后, 将可以选择后面的相应操作。默认不勾选 (此功能需所选从站支持相应功能才可以勾选)。
SDO 错误继续配置	有效时出现配置错误将继续配置下一条配置 (校验类型错误除外), 无效时出现配置错误主站将不继续进行配置, 并且在网络启动的情况下会停止整个网络。此选项默认为不勾选。
创建所有 SDO	选择此功能后, 将添加所有 EDS 中可写的对象字典, 在配置过程中初始化。默认不勾选。
紧急报文	勾选此功能, 在配置过程中将进行紧急报文 COB-ID 设置。默认不勾选。
心跳产生	勾选功能后, 从站将会产生心跳。默认勾选。从站勾选心跳后, 主站默认监控此从站心跳状态。
心跳产生时间(ms)	心跳循环发送的时间。
消费者心跳属性	此功能用于设置本从站将要监控的其他站点心跳。此功能默认不选择。此功能还需要从站支持心跳监控功能。

节点保护	勾选此功能后，从站的节点保护功能将被设置，默认不勾选。节点保护是一种有回帧的主站与从站间互相监控的网络评估功能。心跳和节点保护功能仅可以选择其中一种。 节点保护超时时间=保护时间*生命周期因子。
保护时间(ms)	节点保护时间，默认 200ms
生命周期因子	节点保护因子，默认为 3。
使能同步生产	勾选此项，本站将会按照“同步周期(ms)”设置的时间循环发送同步帧，一个网络里面只可以存在一个同步帧发送。前提是从站要支持发同步功能。
COB-ID	同步帧发送 ID，此项使用默认值 0x80，不允许设置。
同步周期(ms)	发送同步帧的循环周期。默认 200，单位 ms。
重启时检查	检测供应商 ID，检测产品 ID，检测版本：勾选相应的功能，在从站开始配置前将进行相应的校验。如果校验不通过，网络将无法启动。

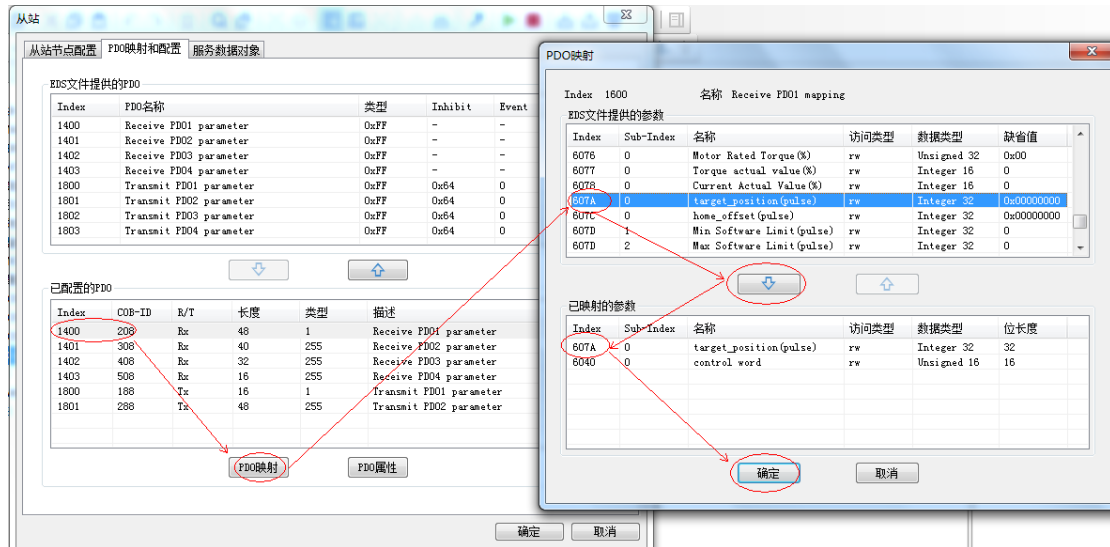
2、PDO 映射和配置界面



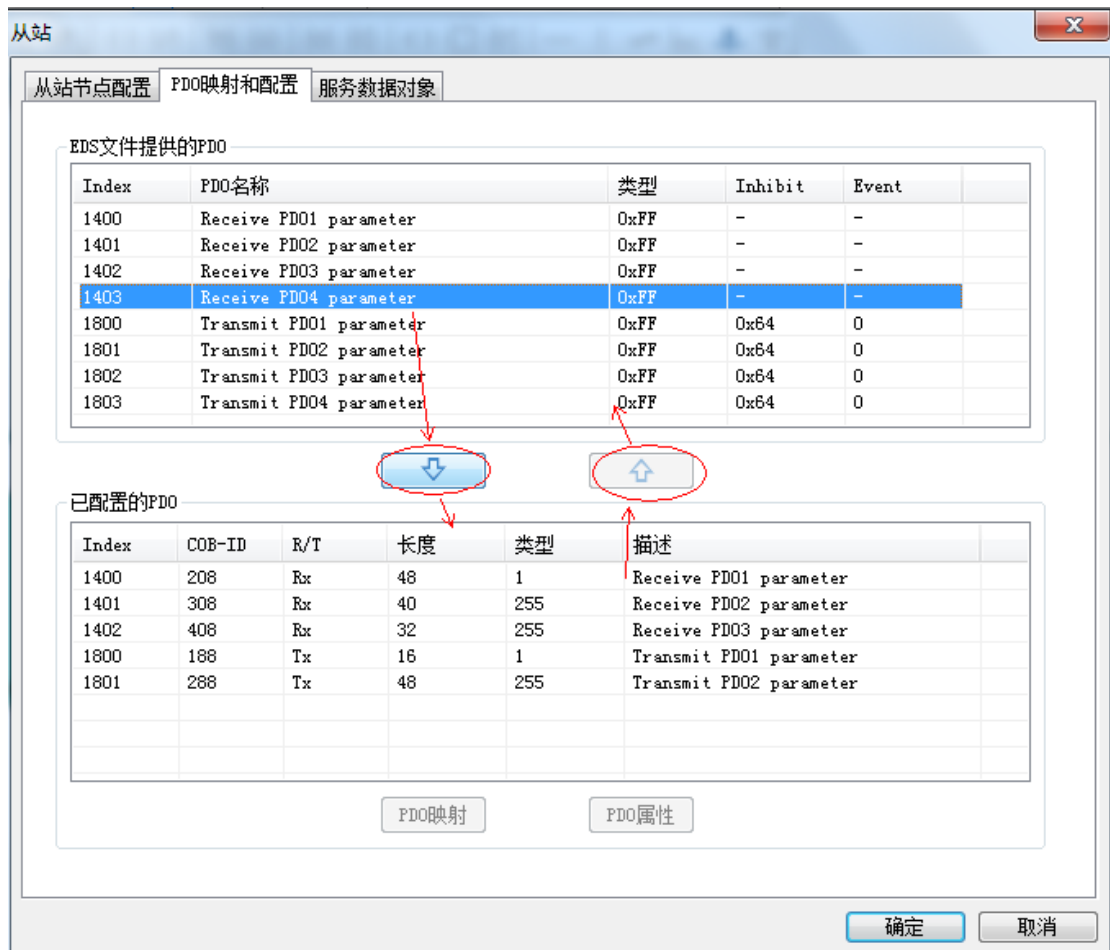
- I、选择“EDS 文件提供的 PDO”，可以单击向下箭头添加 PDO。
- II、选择“已配置的 PDO”，可以单击向上箭头删除 PDO。
- III、选择“已配置的 PDO”，点击“PDO 属性”按钮设置其属性，如下图所示：

PDO 属性说明			
参数	功能		
COB-ID	PDO 发送使用的 ID 号。根据 CANopenDS301 协议的规定，前 4 条 PDO 有默认的 COB-ID 初始值，其他的需要用户自行设置（如果从站支持）。设置原则为整个网络不可以出现重复的 COB-ID，设置范围为 0x180-0x57F。		
传输类型	类型	数据发送条件	数据生效条件
	0: 循环-同步	数据发生变化，并且接收到一帧同步帧	接收到数据后立即生效，需再接收到一帧同步才生效
	1~240: 循环-同步	在接收到相应的“同步数”帧同步后数据发送	接收到数据后立即生效，需再接收到一帧同步才生效
	252: 异步-只有 RTR	不支持	不支持
	253: 异步-只有 RTR	不支持	不支持
	254: 异步-生产厂商指定	由各个厂家自定义	由各个厂家自定义
	255: 异步-设备配置文件指定	数据变化或满足事件时间，并且变化频率小于抑制时间	立即生效
抑制时间	当抑制时间小于 0.5ms 或等于 0 时，抑制时间间隔为 0.5ms，当抑制时间大于 0.5ms 时，抑制间隔为抑制时间*0.1ms。		
事件时间	为 0 时此功能表示按数据变化发送。不为 0 时，表示按定时周期发送。（此发送情况也要受抑制时间的限制）		

IV、选择“已配置的 PDO”，点击“PDO 映射”按钮，打开 PDO 映射配置界面，每个 PDO 的映射长度最大为 64 位。如下图所示：



V、从“EDS 文件提供的参数”中选择需要映射的参数，单击向下箭头添加映射；从“已映射的参数”中选择要删除的映射，点击向上箭头移除映射；如下图所示：



3、服务数据对象界面

此表里面的信息是根据用户的设置自动生成的 SDO 配置数据。也可以点击增加，手动再增加数据字典。

从站

从站节点配置 PDO映射和配置 服务数据对象

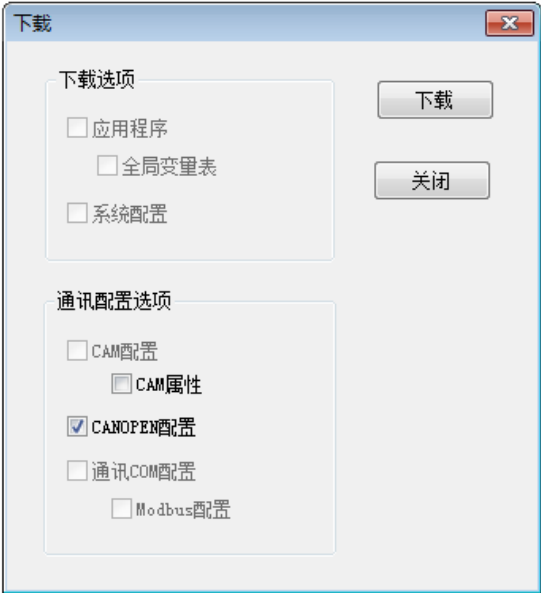
编号	索引	子索引	名称	值	位长度	下载
1	1000	00	Device Type	0xFFFF0402	16	*
2	1018	01	Vendor ID	0xF00002B5	16	
3	1018	02	Product code	0x0006	16	
4	1018	03	Revision number	0x0002	16	
5	1400	01	Disable PDO	0x80000208	32	*
6	1401	01	Disable PDO	0x80000308	32	*
7	1402	01	Disable PDO	0x80000408	32	*
8	1403	01	Disable PDO	0x80000508	32	*
9	1600	00	Clear PDO mapping	0x00	8	*
10	1601	00	Clear PDO mapping	0x00	8	*
11	1602	00	Clear PDO mapping	0x00	8	*
12	1603	00	Clear PDO mapping	0x00	8	*
13	1800	01	Disable PDO	0x80000188	32	*
14	1801	01	Disable PDO	0x80000288	32	*
15	1802	01	Disable PDO	0x80000388	32	*
16	1803	01	Disable PDO	0x80000488	32	*
17	1A00	00	Clear PDO mapping	0x00	8	*
18	1A01	00	Clear PDO mapping	0x00	8	*
19	1A02	00	Clear PDO mapping	0x00	8	*
20	1A03	00	Clear PDO mapping	0x00	8	*
21	1400	02	Set transmission type	0x01	8	*
22	1401	02	Set transmission type	0xFF	8	*
23	1402	02	Set transmission type	0xFF	8	*
24	1403	02	Set transmission type	0xFF	8	*
25	1600	01	Receive PDO mapping	0x607A0020	32	*

增加 修改 删除

确定 取消

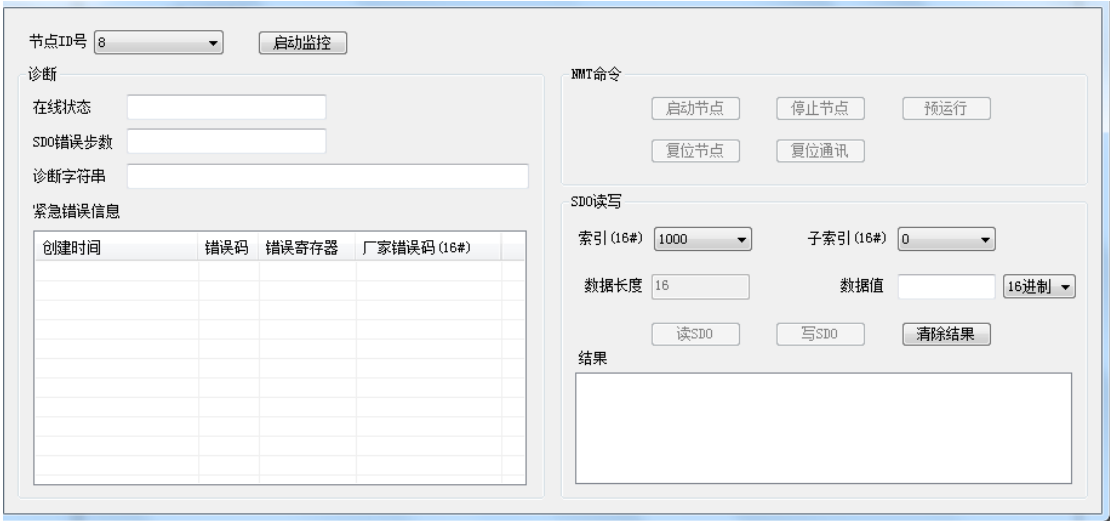
8.6.6 CANopen 配置下载

点击“PLC”菜单中的“下载”菜单，或使用快捷键“F8”，或点击“PLC 工具栏”的下载按钮（）打开下载窗口，点击“下载”按钮即可完成 CANopen 配置的下载，如下图所示：



8.6.7 在线调试

单击“工具”菜单中的“SDO 调试”打开 SDO 调试界面，选择相应的节点号，启动监控，可执行相应的对象字典读写操作及 NMT 命令，如下图所示。



SDO 读写说明	
参数	说明
节点 ID 号	需要读写的 PLC 站点。
索引	PLC 站点对象字典的索引。
子索引	PLC 站点对象字典的子索引。
数据长度	在该索引、子索引的数据的位长度。
数据值	为发送或返回的数据，可以选择 16 进制和 10 进制显示。

读 SDO	执行相应的对象字典读操作。
写 SDO	执行相应的对象字典写操作。
SDO 接收	从 PLC 站点接收到的数据内容显示。

诊断功能	
参数	说明
在线状态	显示选择节点的状态
SDO 错误步数	显示服务数据对象表中 SDO 配置错误的位置
诊断字符串	SDO 配置错误的中止码（见 6.1.9 章节 SDO 中止码）
紧急错误信息	显示主站或从站产生的紧急报文（见 6.1.9 章节紧急报文错误码）

8.6.8 CAN 总线监控

单击“工具”菜单中的“CAN 总线监控”，打开 CAN 总线监控界面，如下图所示。

点击启动监控，可以监控设备所处的状态。网络负载是 1s 内网络总线传输数据所占带宽的百分率，反映整个 CAN 总线通讯数据量的大小，是判断 CAN 组态数据通讯方式是否合理的依据。

还可监控紧急错误信息和服务数据对象表配置错误对应的从站、服务对象表错误位置和错误代码。



The screenshot shows the 'CAN总线监控' (CAN Bus Monitoring) window. It has a title bar with a close button. The main area is divided into three sections: '站点状态' (Station Status), '紧急错误信息' (Emergency Error Information), and 'SDO配置错误' (SDO Configuration Error).

站点状态 (Station Status): This section displays the '网络负载' (Network Load) as a green bar at 4%. To the right is a green button labeled '停止监控' (Stop Monitoring). Below this is a table with two columns: '站点' (Station) and '状态' (Status).

站点	状态
64	运行
8	掉线

紧急错误信息 (Emergency Error Information): This section contains a table with five columns: '创建时间' (Creation Time), '站点' (Station), '错误码' (Error Code), '错误寄存器' (Error Register), and '厂家错误码 (16#)' (Manufacturer Error Code (16#)).

创建时间	站点	错误码	错误寄存器	厂家错误码 (16#)

SDO配置错误 (SDO Configuration Error): This section has three input fields: '站号' (Station Number), '错误步数' (Error Step Number), and '错误码 (16#)' (Error Code (16#)).

8.6.9 CANopen 通讯故障代码及排除

1、SDO 中止码（16 进制）

中止代码	说明
0x05040001	控制命令无效(SDO 只支持 0x40、0x2F、0x2B、0x23 指令)
0x06010002	试图写只读对象
0x06020000	对象字典中的对象不存在
0x06040041	对象不能映射到 PDO
0x06040042	映射的对象的数目和长度超出 PDO 长度
0x06070010	写入长度不匹配(对象字典定义的长度与写入的长度不一致)
0x06070012	数据类型不匹配，服务参数长度不匹配
0x06090011	子索引不存在
0x06090031	写入参数数值太大
0x06090032	写入参数值太小

2、紧急报文错误码（16 进制）

紧急错误码	说明	紧急错误码	说明
0000	错误复位或无错误	6300	数据设置
1000	一般错误	7000	附加模块错误
2000	电流错误	8000	监控错误
2100	设备输入电流	8100	一般错误
2200	设备内部电流	8110	CAN 通不过载
2300	设备输出电流	8120	CAN 被动错误
3000	电压错误	8130	节点保护或心跳错误
3100	电源错误	8140	总线关闭恢复
3200	设备内部电压	8150	CAN-ID 冲突
3300	输出电压	8200	协议错误
4000	温度错误	8210	PDO 长度错误
4100	环境温度	8220	PDO 长度超限
4200	设备温度	8240	不能识别 sync 数据长度
5000	设备硬件错误	8250	RPDO 超时
6000	设备软件错误	9000	外部错误
6100	内部软件	F000	附加功能错误
6200	用户软件	FF00	特殊设备错误

3、通讯失败故障排除

(1) 推荐使用带屏蔽双绞线连接，总线两端分别连接两个 120 欧终端匹配电阻防止信号反射，屏蔽层一般使用单点可靠接地。

(2) 用万用表测量 CANH 和 CANL 之间的阻值可以确认现场端接电阻是否正确，正常阻值应为 60Ω 左右（两个电阻并联），若无 60Ω 左右电阻，不接入总线情况下，分别测量头尾设备 CANH 和 CANL 间阻值是否有 120 欧，若有，则可能中间 CAN 设备有匹配电阻，去掉中间 CAN 设备匹配电阻。CAN 总线通讯时中间设备不能接入匹配电阻，否则会影响其后设备的正常通讯。

(3) 主从波特率设置一定要一致，否则无法和从站通讯。

(4) 站地址不能重复，否则主站无法正确读写有重复地址的从站。

(5) CAN 设备长距离通信时，须将不同 CAN 电路的公共地 CGND 相互连接，以保证不同通信设备之间参考电位相等，并且适当降低波特率保证正常通讯。

(6) 对于干扰很大，上述办法没有效果时，适当总线降低波特率。

(7) 当总线负载率过高时（大于 90%），总线堵塞，也会出现通讯失败、掉线的情况，出现负载率高，往往时程序规划数据的问题。此时应合理规划交换的数据量，选择不同的类型的传输方式进行数据交换。一般在同步模式下，选择同步周期短，同步数小，又传输数据量大时，会出现通讯失败、掉线，此时要提高同步周期或同步数。异步模式下，抑制时间过短也会通讯失败、掉线的，要加大抑制时间。

8.6.10 CANopen 通讯变量

CANopen 通讯变量表:

M 变量	说明	D 变量	说明
M8282	0: 仅复位当前的节点 1: 当前节点和总网络复位	D8240	总线负载率
M8284	设定 CAN 地址	D8241	错误--主站号
M8285	设定波特率	D8242	错误--错误码
		D8243	错误--0x11
		D8244	错误--辅助字节 L
		D8245	错误--辅助字节 L
		D8246	在线--节点
		D8247	在线--索引
		D8248	在线--字节长度+子索引
		D8249	在线--数据 1
		D8250	在线--数据 2
		D8251	在线--数据 3
		D8252	在线--数据 4
		D8253	在线--命令 0:写 SDO 1:读 SDO 2:启动节点 3:停止节点 4:预运行 5:复位节点 6:复位通讯 0xff:等待
		D8254	SDO 错误中止代码 L
		D8255	SDO 错误中止代码 H
		D8284	设定 CAN 地址
		D8285	设定波特率
		D8287	CANOpen 配置错误站号
		D8288	CANOpen 配置错误编号
		D8289	CAN 总线错误
		D8290	CAN 接收错误计数
		D8291	CAN 每秒收发帧总数
		D7800	状态--主站
		D7801	状态—从站 1
		D78xx	最后一个用 255 填充

8.6.11 V5-CANopen 控制 SD700 伺服驱动器

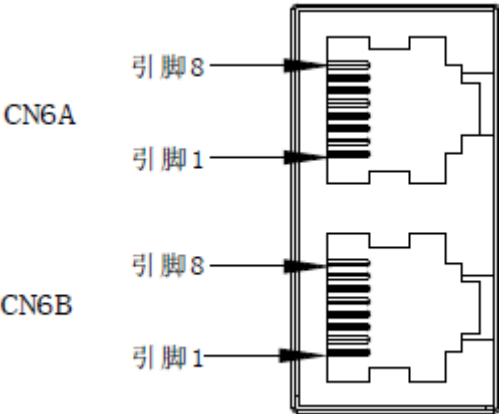
1、SD700 伺服命名（确定伺服驱动器是否带 CAN 通讯功能）：

SD700-3R3A-PA*

A B C D E F G

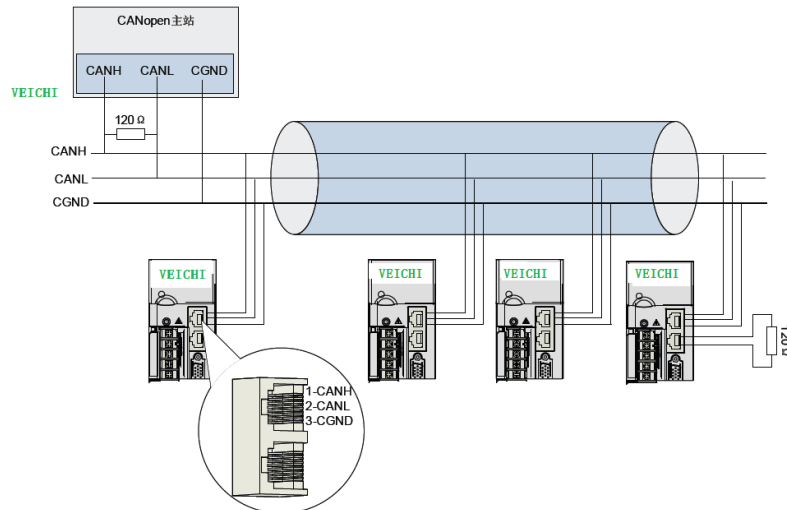
字段标识	字段详细说明	
A	SD: 伺服产品代号	
B	700: 产品系列	
C	电流等级:	
	1R1: 1.1A	1R8: 1.8A 3R3: 3.3A 5R5: 5.5A 7R6: 7.6A 9R5: 9.5A
	2R5: 2.5A	3R8: 3.8A 6R0: 6A 8R4: 8.4A 110: 11A 170: 17A
	240: 24A	300: 30A
D	输入电压等级:	
	A: 220VAC; D: 400VAC	
E	机器类型:	
	P: 脉冲型	
	S: 标准型	
	C: CANopen 总线型	
	N: EtherCAT 总线型	
	M: MECHATROLINK-II 总线型	
F	支持的编码器类型:	
	A	绝对值型
	B	增量型
	T	旋转变压器型
G	产品管理号, 标准产品缺省	

2、SD700 通讯端子 CN6A 和 CN6B 定义（可以用网线连接 V5 和 S700 的 can 通讯口）：



CN6A/CN6B 接口定义					
针号	信号名称	功能	针号	信号名称	功能
1	CANH	CAN 数据+	6	-	
2	CANL	CAN 数据-	7	GND	485 信号地
3	CANG	CAN 信号地	8	-	-

3、V5 和 SD700 之间 CANopen 配线示意图：



4、伺服 CANopen 通讯参数（站地址、波特率、内部位置指令选择选择 4-Canopen）设置：

功能码	参数名称	当前值	单位	范围	默认值
✓ Pn030	保留参数	0x1706	-	0~65535	0x6AA
✓ Pn080	本机通信地址	0x8	-	0~127	0x1
✓ Pn083	CANopen 通讯波特率选择	[6]1Mbps		0~6	[4]250...
✓ Pn085	通讯数据存储EEPROM选择	0x0	-	0~65535	0x1
✓ Pn204	电子齿轮比分子	0	-	0~1073741824	64
✓ Pn206	电子齿轮比分母	10000	-	1~1073741824	1
▶ ✓ Pn208	内部位置指令选择	[4]CanOpen		0~4	[0]位置...
✓ Pn791	编码器类型	[9]23位单圈绝...	-	1~18	[7]24...
✓ Pn880	Pr加减速时间0	50	ms	0~60000	100
✓ Pn898	Pr指令通讯触发	20000		0~65535	10000

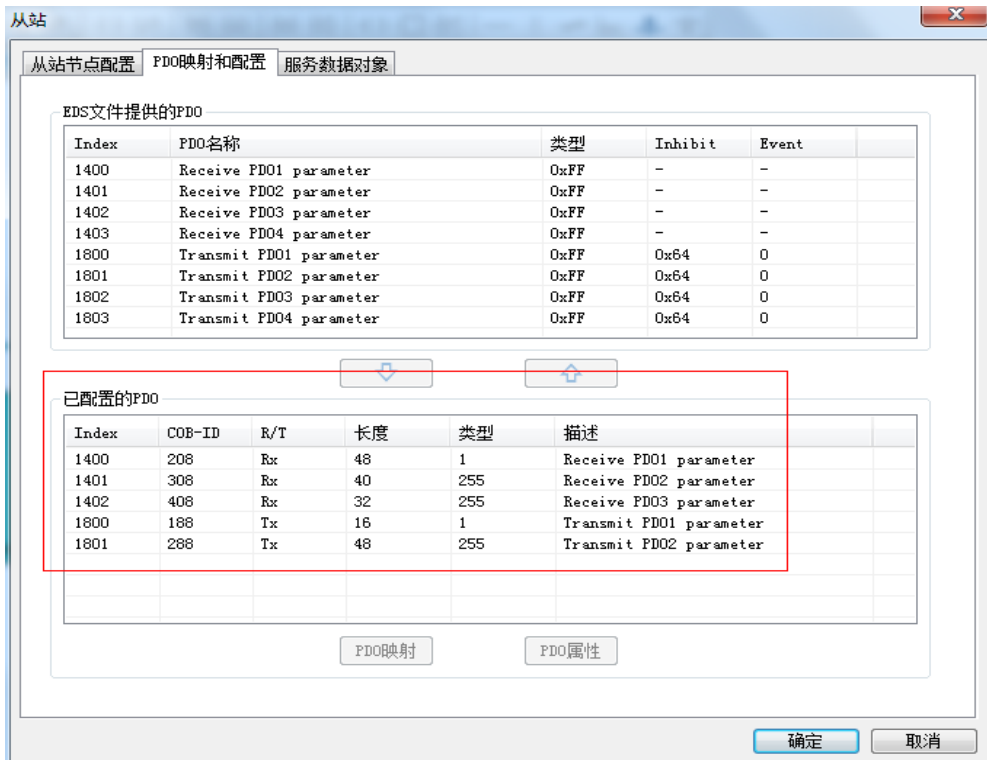
5、编写 V5 控制伺服正反转程序：

（1）PDO 属性设置：

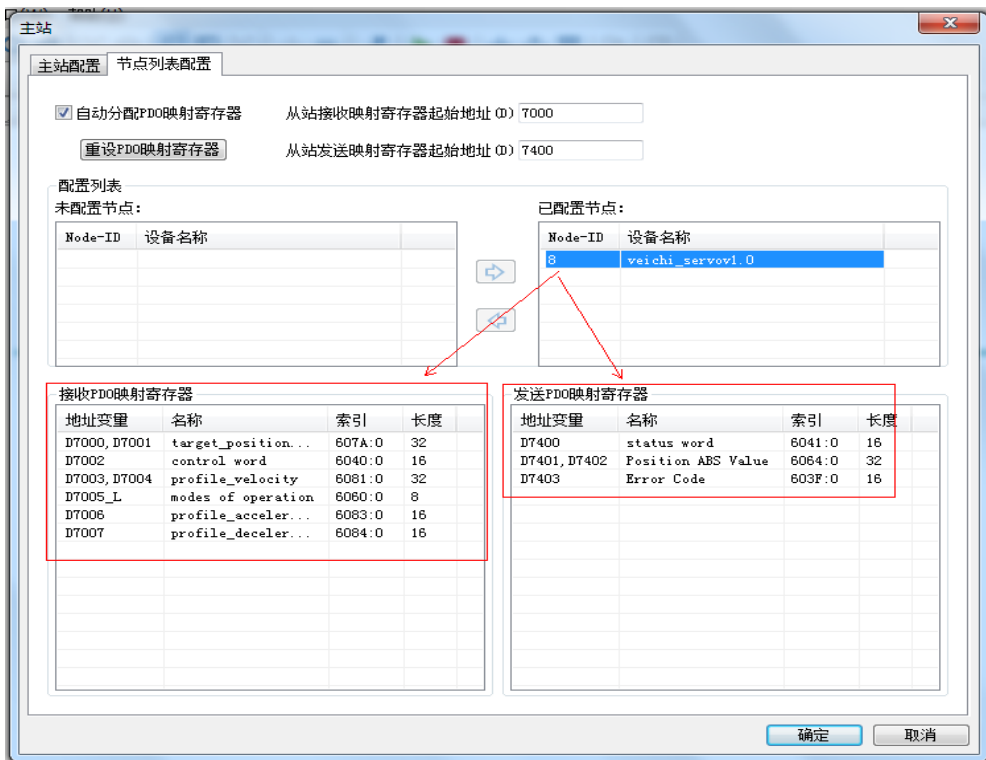
TPDO1 和 RPDO1 设置为同步模式，同步周期为 4ms（同步周期或同步数可以根据所带从站数和通讯数据量设置）。同步数为 1。

TPDO2 设置为同步模式，同步数为 5。RPDO2 和 RPDO3 设置为异步模式，抑制时间和事件时间都设为 0，按数据变化发送。

TPDO2 设置为异步模式，事件时间为 10ms，抑制时间为 5ms。

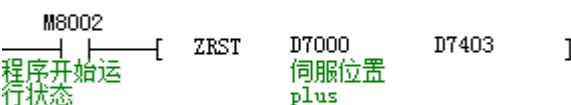


(2) IO 映射表:

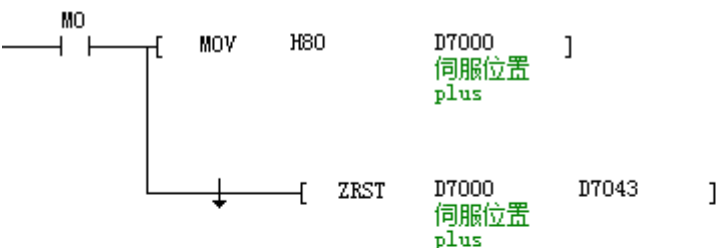


(3) 主程序:

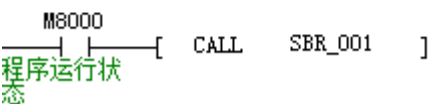
（因IO映射表中的D寄存器为掉电保持类，为防止上电误动作，上电将IO映射区D寄存器清零。）



复位伺服 对控制字D7000赋值H80。



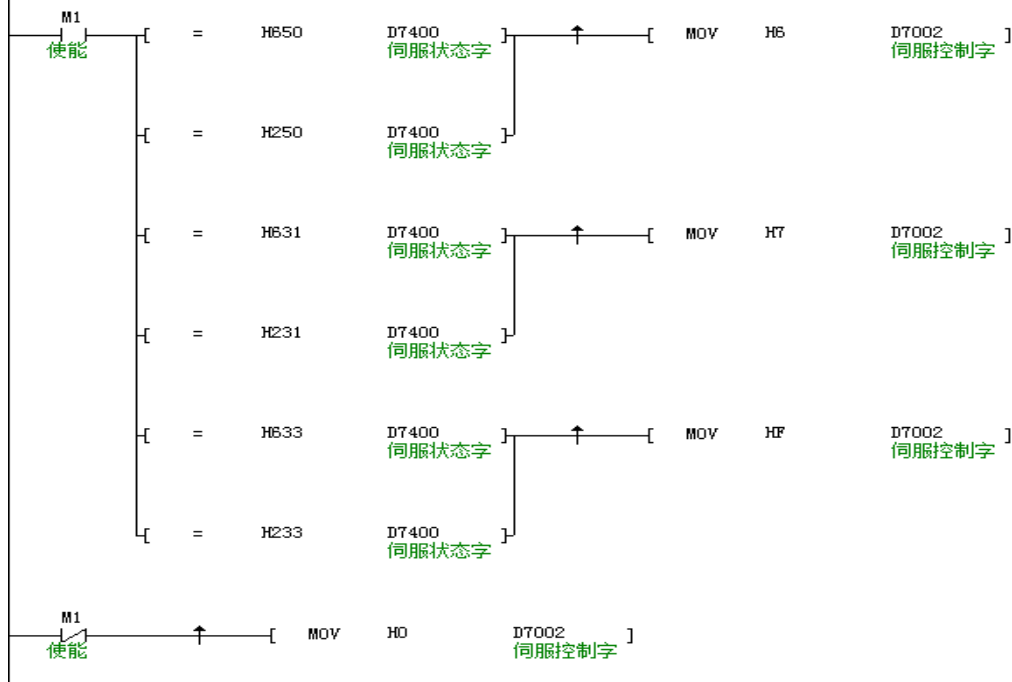
调用控制从站1子程序



(4) 子程序:

网络 1:

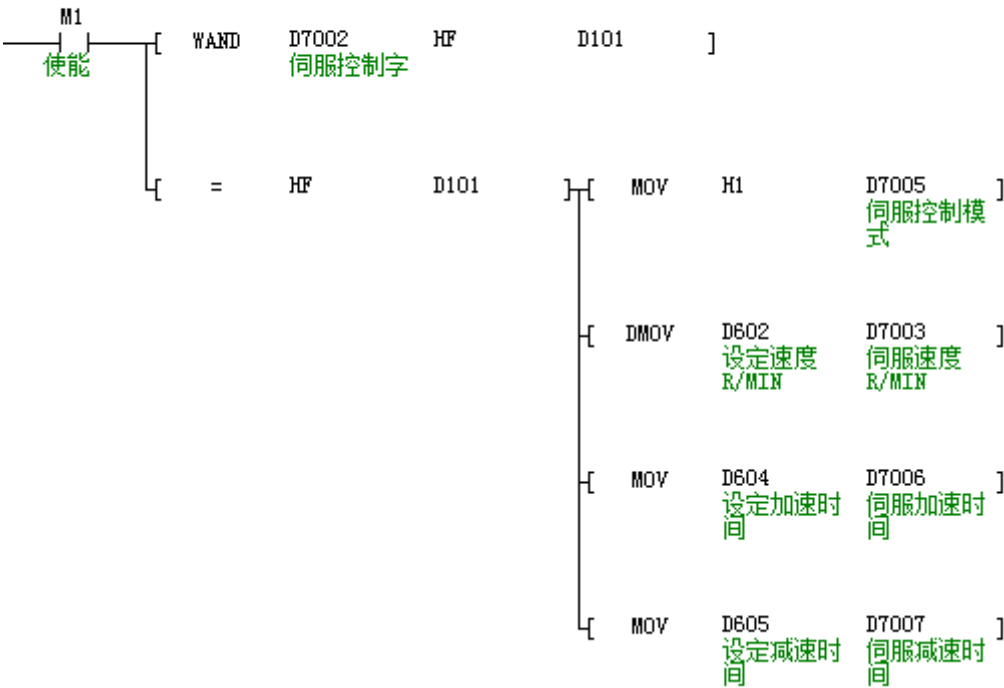
使能伺服-根据反馈状态字D7400,依次写H6、H7、HF给控制字D7000。



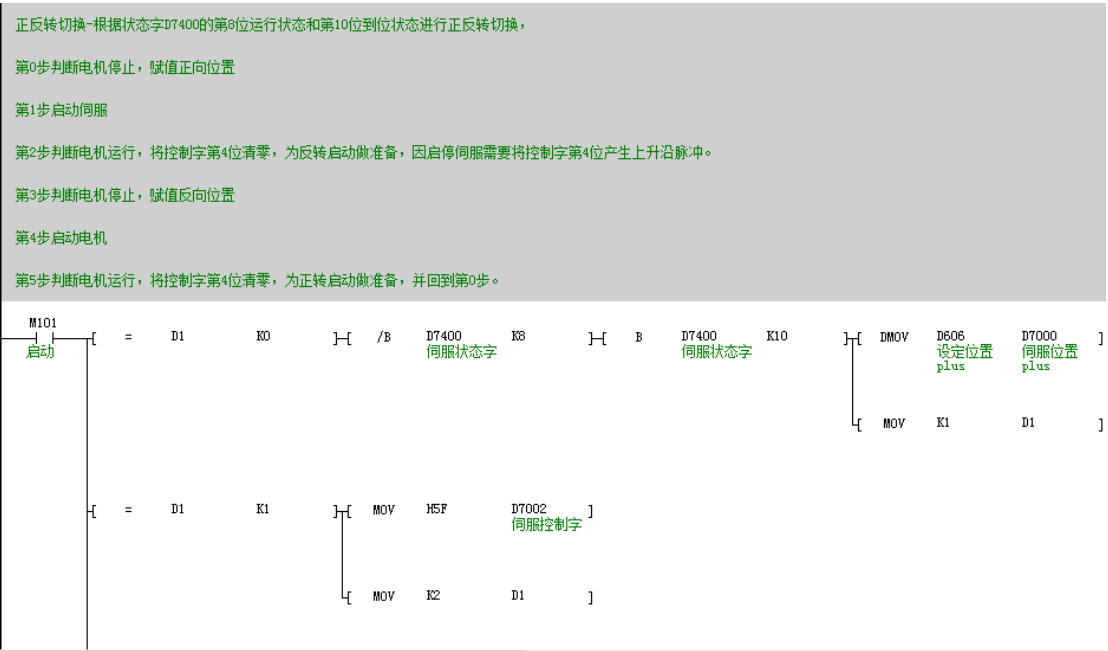
网络 2:

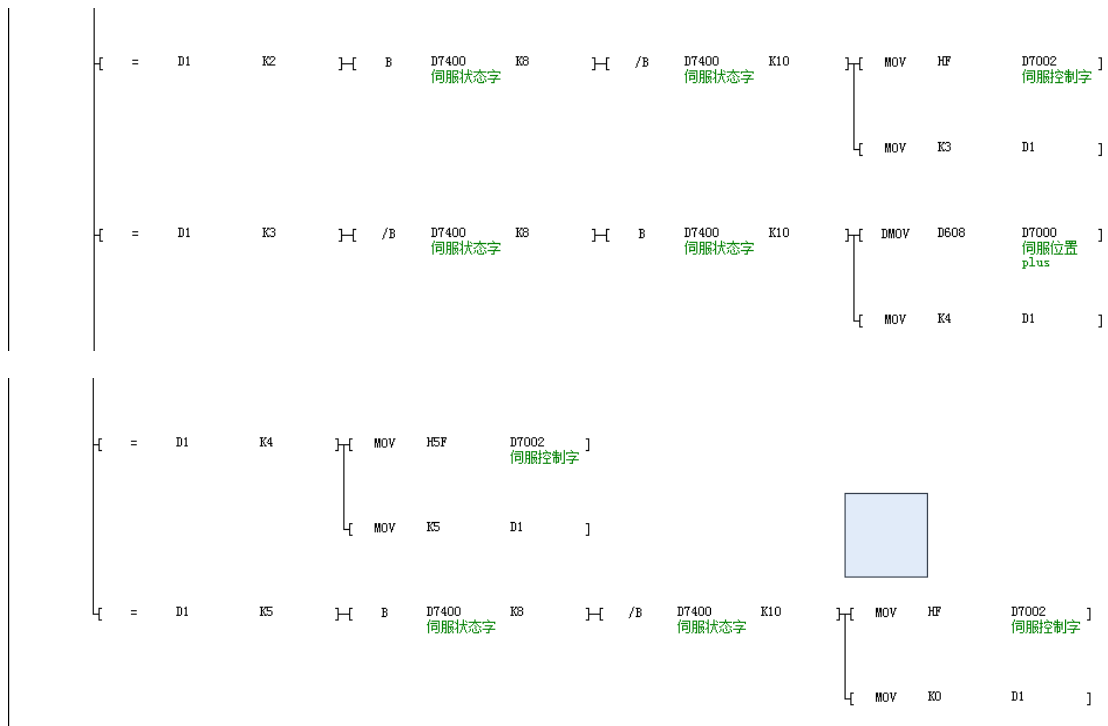
赋值

根据伺服使能后，写伺服控制模式、速度、加速时间、减速时间

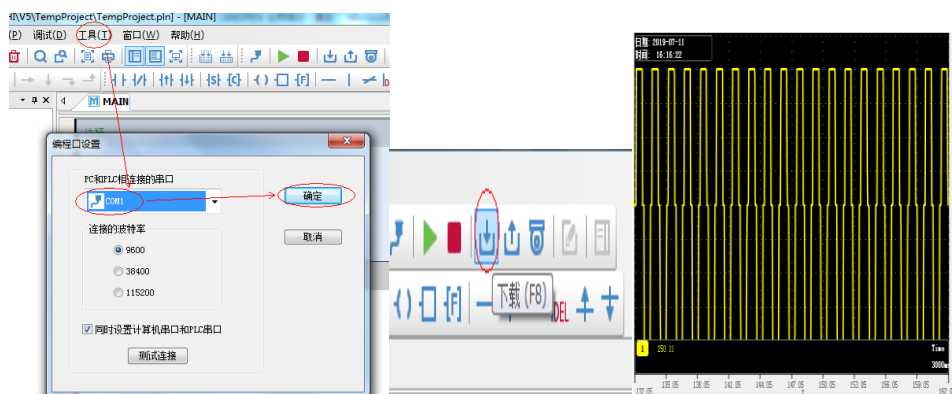


网络 3:





(5) 下载程序，并用 SD700 伺服上位机软件示波器功能监控电机参数指令速度：



第9章 子程序

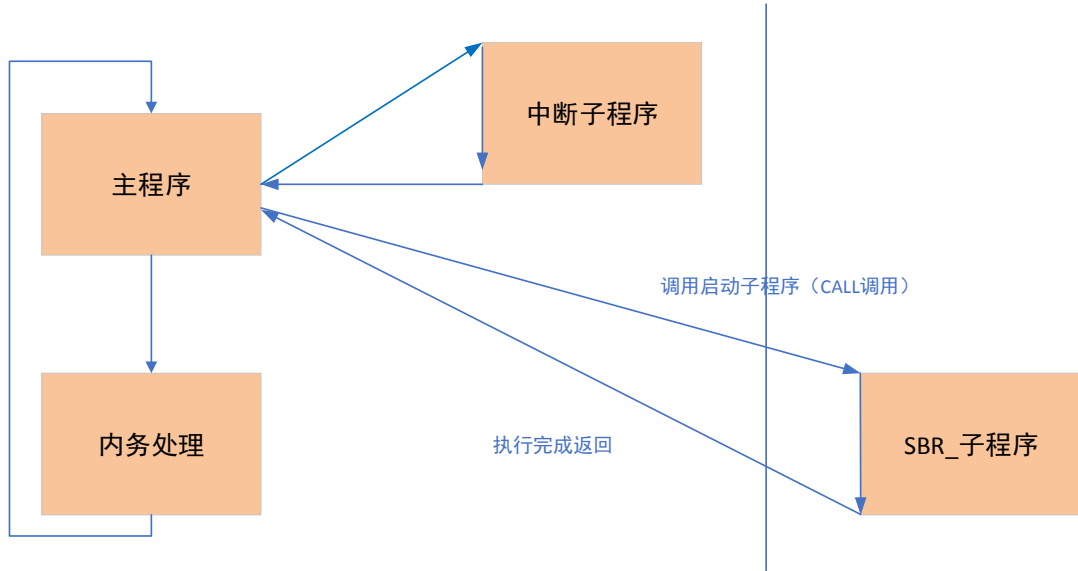
9.1 概要

9.1.1 V5 子程序概述

		概述	
SBR	CALL 指令用	一个工程内可以有多个子程序，不能超过 64 个，子程序由主程序或者其他子程序调用，主要完成一些常用或者需要重复使用的功能，子程序只能由梯形图或者指令列表编写。	
P	CJ 指令用	512 点，与 LBL 指令配套使用	
I	中断子程序	外部中断	PG0-PG2 输入中断，编号 I000, I001, I100, I101, I200, I201 (I000 表示 X0 下降沿中断, I001 表示 X0 上升沿中断, I100 表示 X1 下降沿中断, I101 表示 X1 上升沿中断, I200 表示 X2 下降沿中断, I201 表示 X2 上升沿中断)。其中断禁止标志位寄存器置 ON 后，则对应的输入中断被禁止。
		脉冲完成中断	I502, I502, I504。
		定时中断	I600 (I6 表示定时中断 0, 00 表示时基, 时基范围为 1 至 99), I700 (定时中断 1), I800 (定时中断 2)。
		计数完成中断	I010、I020、I030、I040、I050、I060、I070、I080，8 点 (DHSCS 指令用)。

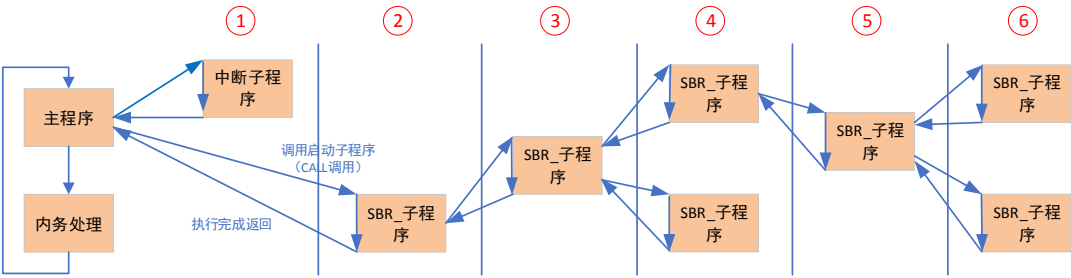
9.1.2 V5 子程序执行机制

主程序、子程序执行逻辑，循环扫描方式。



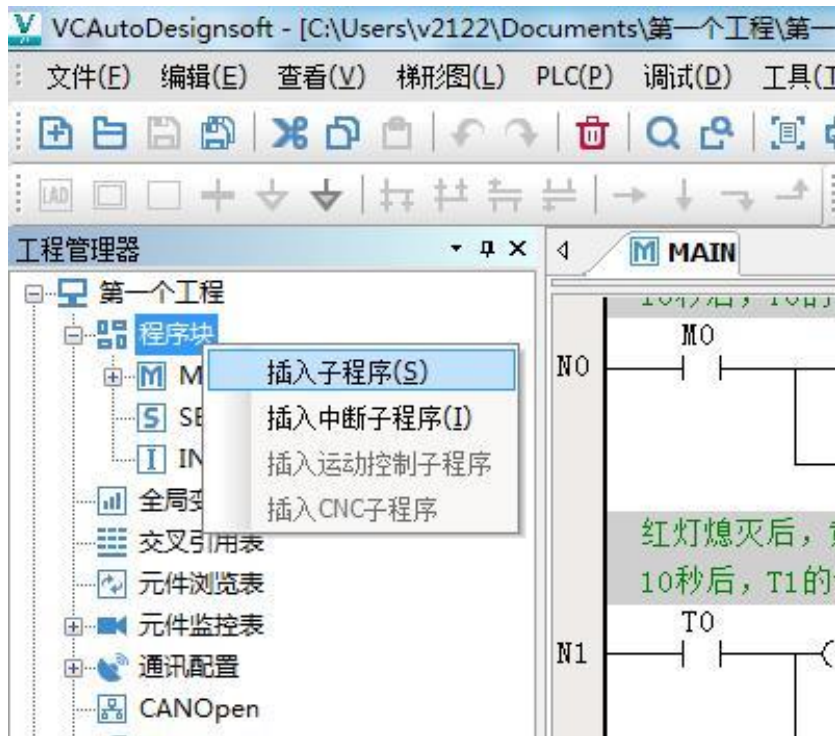
子程序嵌套层级

子程序最多嵌套 6 层，以主程序调用子程序作为第一层，没调用一次，增加一层嵌套，如果嵌套返回，则不增加嵌套层级，如下图所示。



9.2 通用子程序应用

9.2.1 建立子程序



右键单击“工程管理器”窗口的“程序块”节点，选择“插入子程序”或者“插入中断子程序”来插入新的子程序，新的子程序名的默认名称是 SBR_*，新的中断子程序名的默认名称是 INT_*（其中的*是由软件自动计算的一个数字），建立完成后可以通过子程序属性对话框将默认程序名改为一个更有意义的名称。在工程树上插入新程序节点的同时，该程序也会被打开，可以马上对它进行编辑。

9.2.2 导出子程序

软件提供了子程序库的导入和导出功能，对于某个通用功能，可以在子程序中编写完成后，通过导入/导出功能，在不同的工程中再次使用，避免相同功能的重复代码编写。VCAutoDesignsoft 使用程序库文件的方式来管理这些程序，通用的子程序首先可以导出到某个目录形成可以重复使用的程序库文件，在需要使用这些子程序的工程中，使用导入功能可以将程序库文件导入到当前工程中作为标准的子程序使用。

选择菜单“文件”下的“程序文件导”菜单项，打开程序导出对话框，对话框中自动把该工程中可以导出的程序列出来，程序导出的路径第一次使用时默认为软件安装路径下的\Lib\目录下，用户可以点击“...”按钮来更换该路径。用户选择的路径自动成为程序默认的导出/导入路径。



可以编辑程序的导出程序名以及程序说明，最后在需要导出的程序前打勾，然后点击“导出程序”按钮。程序正确导出后该程序前的勾自动消失，否则仍然处于导出前的状态。



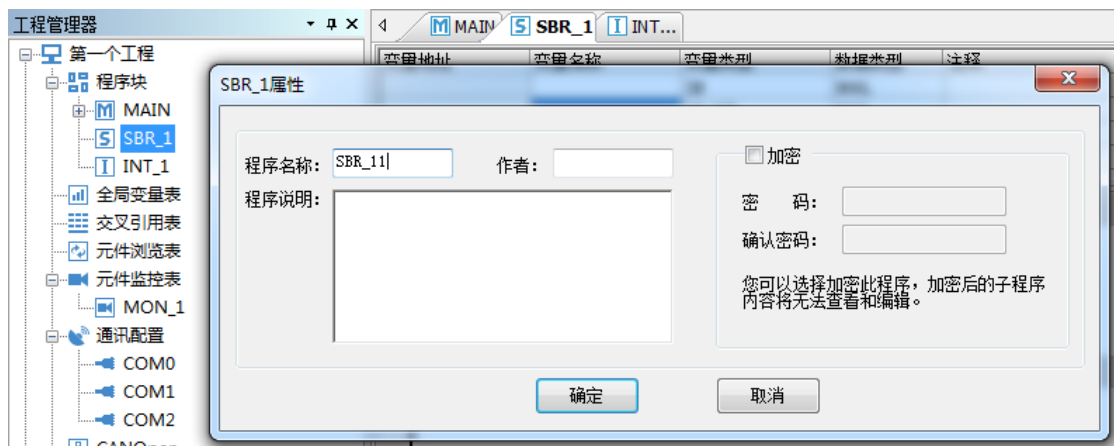
9.2.3 导入子程序

选择菜单“文件”下的“程序文件导”菜单项，打开程序导入对话框：对话框中自动列出了在默认导入导出路径下可以导入的程序信息，可以点击“…”按钮来更改路径。导入前可以编辑程序导入工程后的名称，当导入程序名不合法时，在列表用红颜色加下划线来提示，如果所指定的程序名与工程中已有的程序重名时，会提示是否要覆盖。选择好程序后，点击“导入程序”按钮，程序导入到工程。



9.2.4 子程序属性

选中子程序节点，然后点击鼠标右键弹出菜单，选择“属性”来打开子程序属性对话框，在子程序属性对话框中可以修改该子程序的程序名称、作者、程序说明及加密，如下图所示：



单击“确定”保存，子程序名将以新输入的“程序名称”重命名，如下图所示：



9.2.5 子程序调用

子程序调用规则：

- 1、在主程序中，您可以嵌套子程序（在子程序中放置子程序调用指令），最大嵌套深度为 5，中断程序中不允许调用子程序；
- 2、用户程序之间禁止显式循环调用，如子程序 A 和 B 互相调用对方；
- 3、用户程序禁止递归调用，例如用户程序 A 调用程序 B，程序 B 调用程序 C，程序 C 再调用程序 A，形成一个环，此外子程序不允许调用本身。

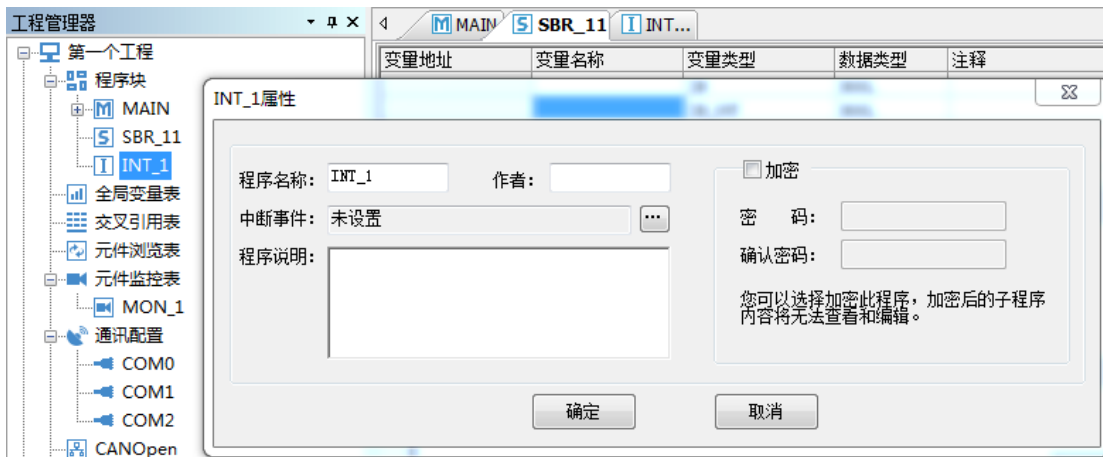
其他程序单元调用子程序：子程序可使用 Call、Callp 指令被主程序、其他子程序调用。

子程序中调用其他程序块：子程序中可调用其他子程序、电子凸轮、控制中断子程序等。

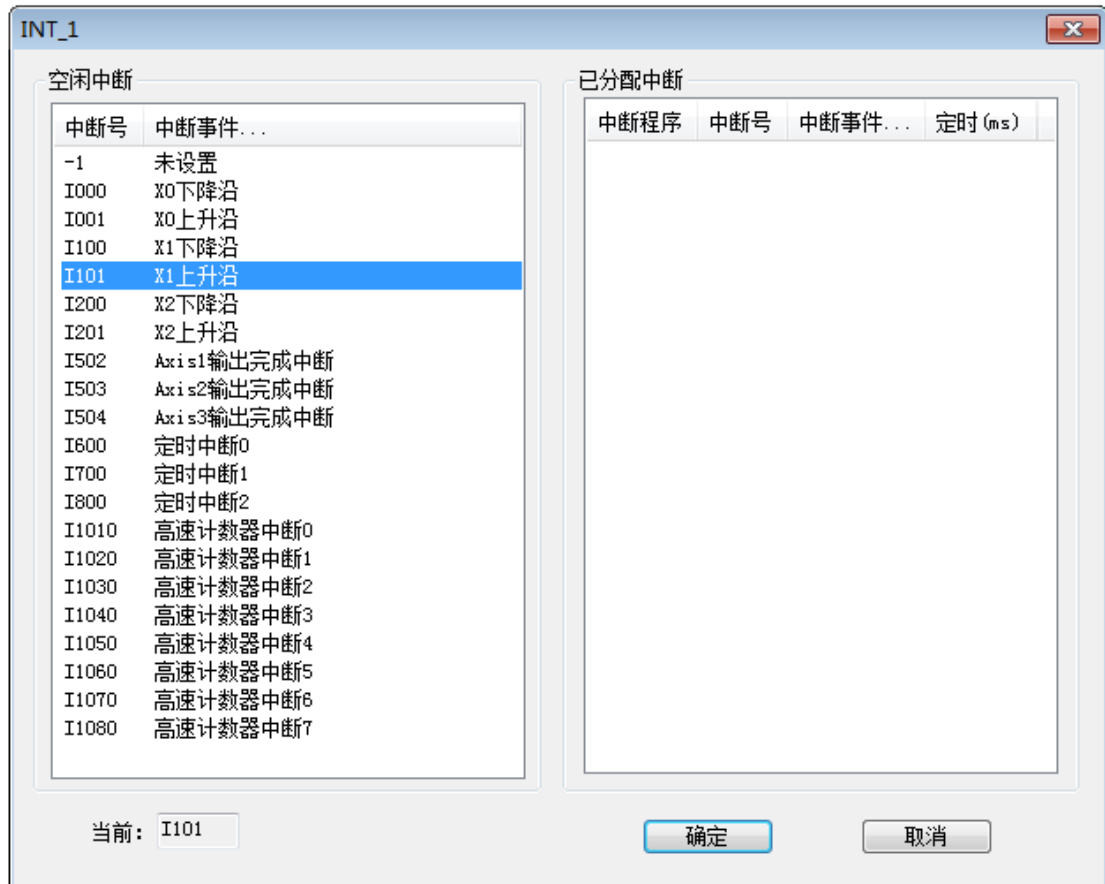
9.3 中断子程序应用

9.3.1 中断子程序属性

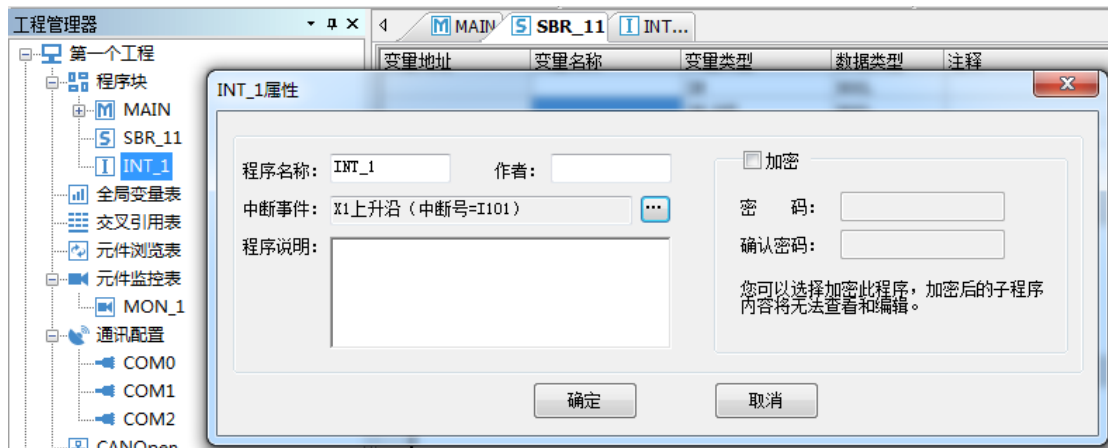
选中中断子程序节点，然后点击鼠标右键弹出菜单，选择“属性”来打开中断子程序属性对话框。对于中断子程序，可以为其指定中断号（新建的中断子程序默认中断号为-1，表示未设置），中断子程序属性对话框如下图所示：



点击“...”按钮，会弹出中断分配对话框，在这里显示当前可用的空闲中断以及已经被其他中断子程序使用的中断，您可以在空闲中断中选择一个中断号指定给当前中断子程序，如下图所示：



在中断设置窗口选中中断号后单击“确定”返回到中断属性窗口，新设置的中断如下图所示：



中断属性窗口单击“确定”按钮完成中断设置，中断子程序属性窗口可同时设置程序名称、作者、程序说明以及加密。

9.3.2 中断子程序调用

1、中断说明

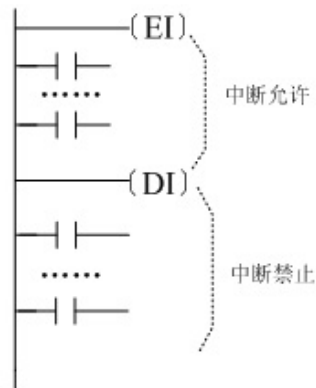
中断子程序由中断功能触发后立即执行，不受用户程序扫描周期影响；在一般的顺控程序处理中，由运算周期造成的延迟以及时间的偏差给机械动作带来影响，在中断子程序中可以得到改善。

V5-MC104 支持的中断类型如下表所示：

中断类型	中断号
外部中断	PG0-PG2 输入中断，编号 I000, I001, I100, I101, I200, I201 (I000 表示 X0 下降沿中断, I001 表示 X0 上升沿中断, I100 表示 X1 下降沿中断, I101 表示 X1 上升沿中断, I200 表示 X2 下降沿中断, I201 表示 X2 上升沿中断)。其中断禁止标志位寄存器置 ON 后，则对应的输入中断被禁止。
脉冲完成中断	I502, I502, I504。
定时中断	I600 (I6 表示定时中断 0, 00 表示时基, 时基范围为 1 至 99), I700 (定时中断 1), I800 (定时中断 2)。
计数完成中断	I010、I020、I030、I040、I050、I060、I070、I080, 8 点 (DHSCS 指令用)。

2、开启、禁用中断

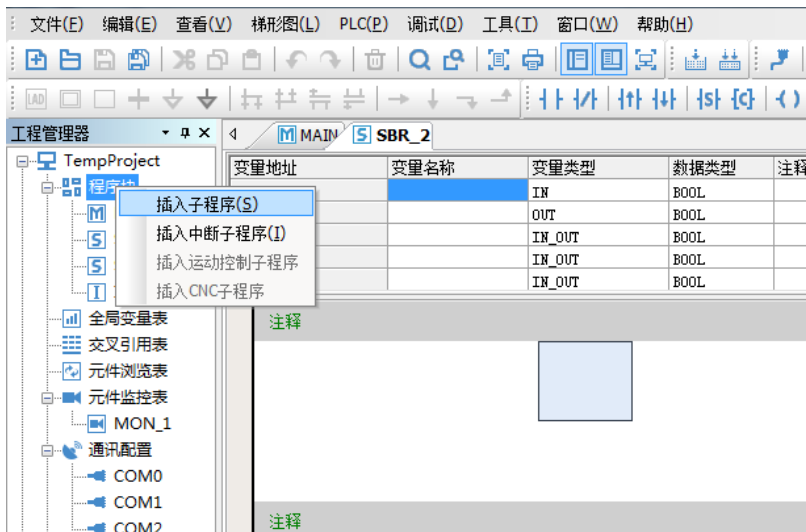
PLC 程序开始运行时，默认为中断禁止状态；执行了 EI 语句后，中断功能允许；当中断为允许状态，执行了 DI 语句后，即进入中断禁止状态。在程序中如果没有中断插入禁止的区间时，可以不使用 DI 指令。



9.4 带参子程序应用

V5 系列可编程控制器支持带参数的子程序调用，提供局部变量（VM、VD），允许参数传递，可以为子程序定义输入和输出。

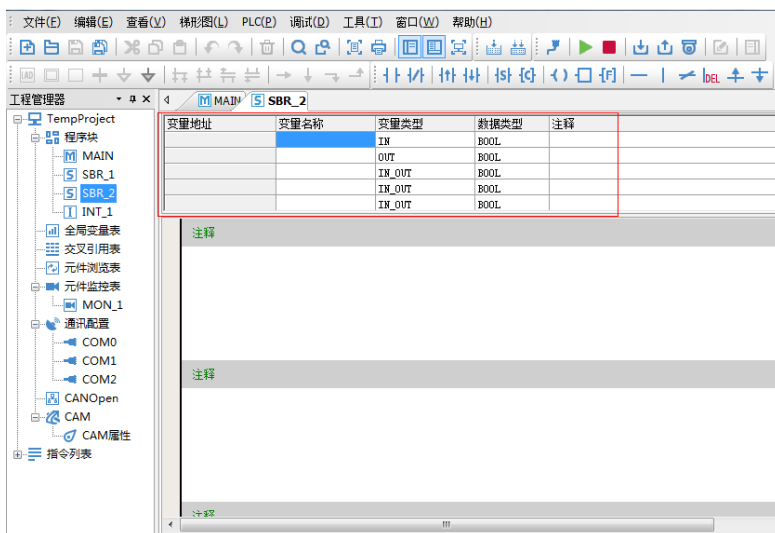
9.4.1 带参子程序的建立



9.4.2 输入输出参数的定义

1、变量地址

最多支持 32 个 VM 元件，96 个 VD 元件，除去输入和输出传递参数外，其他 VM、VD 可以作为临时变量使用。



2、定义变量名称的语法规则：

- 变量名称可包含字母数字字符、下划线，第一个字符不能为数字。
- 不可使用关键字作为变量名称，如 IN、OUT、LD、ADD 等。
- 符号名的最大长度为 8 个字符（或 4 个汉字）。

3、定义变量名称错误提示：

变量名称如果定义错误，上位机软件会用红色和下划线提示

变量地址	变量名称	变量类型
	<u>IN</u>	IN
	<u>OUT</u>	OUT
	<u>LD</u>	IN_OUT
	<u>ADD</u>	IN_OUT
		IN_OUT

- 红色文本表示语法无效。
- 变量名称不能以数字开头；
- 变量名称不能使用关键字；
- 变量名称不能重复。

4、变量类型

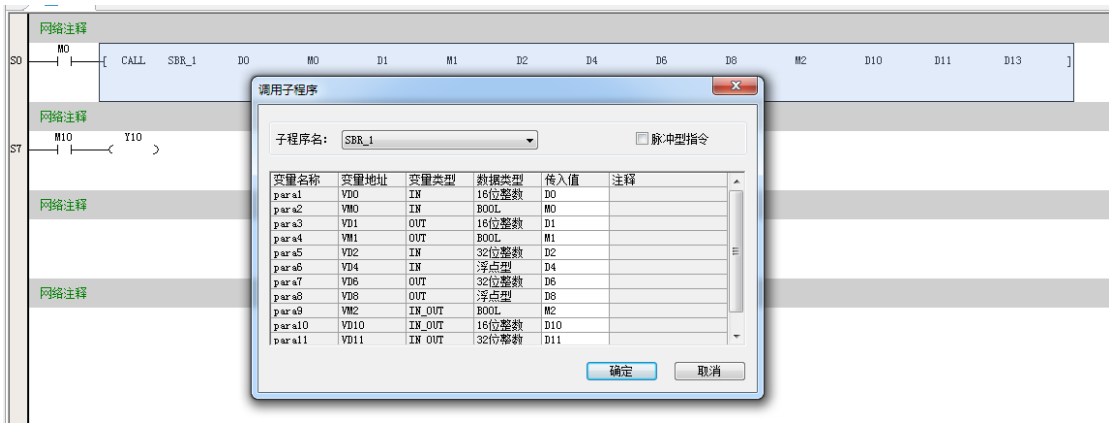
变量类型	说明
IN	如果要将数据值传递至子程序，则在子程序变量表中创建一个变量，并将其类型指定为 IN
OUT	如果要将子程序中建立的数据值传回至调用程序，则在子程序的变量表中创建一个变量，并将其类型指定为 OUT。
IN_OUT	指定参数位置的值被传到子程序，从子程序的结果值被返回到同样地址

- 定义 IN、OUT、IN_OUT 三种参数的总个数最多 16 个

5、数据类型

支持 BOOL、16 位整数、32 位整数和浮点数。

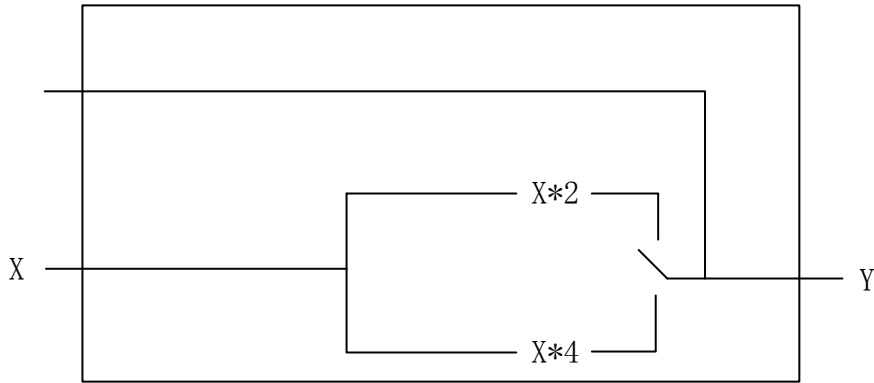
9.4.3 带参子程序的调用



在程序中输入 CALL 后敲空格键弹出调用子程序框，选择需要的带参子程序，其参数名称已经填写过，故可以看出其别名。在调用方输入需要传入的参数以及输出参数。

举例：

带参子程序实现一个数学运算，如 SEL = ON 时输出为 $\text{para1} \times 2$ ，否则为 $\text{para1} \times 4$



其中 para1 为 32 位浮点数，para2 为 32 位浮点数。SEL 为位变量。

变量地址	变量名称	变量类型	数据类型	注释
VM0	para0	IN	BOOL	
VD0	para1	IN	浮点型	
VD2	para2	IN	浮点型	
		OUT	浮点型	
		OUT	浮点型	

注释
<pre> VM0 ├── [DEMUL VD0 E2.000000 VD2] └── / [DEMUL VD0 E4.000000 VD2] </pre>

9.4.4 带参子程序的主要事项

- 子程序不可以重入，因此不可以递归调用，也不可以在中断中调用
- 不能使用如下指令 OUT C(235-255), PLSY, DPLSY, PLSR, DPLSR, DRVI, DDRVI, DRVA, DDRVA, PLSV, DPLSV, SPD 等高速输入、输出指令。

附录一 特殊软元件分配说明

SM 标志位分配

轴 1	轴 2	轴 3	轴 4	Y0	Y1	说明
SM0	SM100	SM200	SM700	SM800	SM900	运动控制用
SM1	SM101	SM201	SM701	SM801	SM901	运动控制用
SM2	SM102	SM202	SM702	SM802	SM902	运动控制用
SM3	SM103	SM203	SM703	SM803	SM903	运动控制用
SM4	SM104	SM204	SM704	SM804	SM904	运动控制用
SM5	SM105	SM205	SM705	SM805	SM905	运动控制用
SM6	SM106	SM206	SM706	SM806	SM906	运动控制用
SM7	SM107	SM207	SM707	SM807	SM907	运动控制用
SM8	SM108	SM208	SM708	SM808	SM908	运动控制用
SM9	SM109	SM209	SM709	SM809	SM909	运动控制用
SM10	SM110	SM210	SM710	SM810	SM910	运动控制用
SM11	SM111	SM211	SM711	SM811	SM911	运动控制用
SM12	SM112	SM212	SM712	SM812	SM912	DRVZ 原点回归方向
SM13	SM113	SM213	SM713	SM813	SM913	ZRN 原点信号指定
SM14	SM114	SM214	SM714	SM814	SM914	运动控制用
SM15	SM115	SM215	SM715	SM815	SM915	运动控制用
SM16	SM116	SM216	SM716	SM816	SM916	运动控制用
SM17	SM117	SM217	SM717	SM817	SM917	G01S 曲线使能
SM18	SM118	SM218	SM718	SM818	SM918	DRVZ 指令使能
SM19	SM119	SM219	SM719	SM819	SM919	反向补偿使能标志
SM20	SM120	SM220	SM720	SM820	SM920	连续插补使能标志
SM21	SM121	SM221	SM721	SM821	SM921	运动控制用
SM22	SM122	SM222	SM722	SM822	SM922	运动叠加状态标志
SM23	SM123	SM223	SM723	SM823	SM923	运动控制用
SM24	SM124	SM224	SM724	SM824	SM924	运动控制用
SM25	SM125	SM225	SM725	SM825	SM925	运动控制用
SM26	SM126	SM226	SM726	SM826	SM926	运动控制用
SM27	SM127	SM227	SM727	SM827	SM927	运动控制用
SM28	SM128	SM228	SM728	SM828	SM928	运动控制用
SM29	SM129	SM229	SM729	SM829	SM929	运动控制用
SM30	SM130	SM230	SM730	SM830	SM930	运动控制用
SM31	SM131	SM231	SM731	SM831	SM931	运动控制用
SM32	SM132	SM232	SM732	SM832	SM932	运动控制用
SM33	SM133	SM233	SM733	SM833	SM933	运动控制用
SM34	SM134	SM234	SM734	SM834	SM934	运动控制用
SM35	SM135	SM235	SM735	SM835	SM935	运动控制用
SM36	SM136	SM236	SM736	SM836	SM936	运动控制用
SM37	SM137	SM237	SM737	SM837	SM937	运动控制用

SM38	SM138	SM238	SM738	SM838	SM938	运动控制用
SM39	SM139	SM239	SM739	SM839	SM939	运动控制用
SM40	SM140	SM240	SM740	SM840	SM940	运动控制用
SM41	SM141	SM241	SM741	SM841	SM941	运动控制用
SM42	SM142	SM242	SM742	SM842	SM942	运动控制用
SM43	SM143	SM243	SM743	SM843	SM943	运动控制用
SM44	SM144	SM244	SM744	SM844	SM944	运动控制用
SM45	SM145	SM245	SM745	SM845	SM945	运动控制用
SM46	SM146	SM246	SM746	SM846	SM946	运动控制用
SM47	SM147	SM247	SM747	SM847	SM947	运动控制用
SM48	SM148	SM248	SM748	SM848	SM948	运动控制用
SM49	SM149	SM249	SM749	SM849	SM949	运动控制用
SM50	SM150	SM250	SM750	SM850	SM950	运动控制用
SM51	SM151	SM251	SM751	SM851	SM951	运动控制用
SM52	SM152	SM252	SM752	SM852	SM952	运动控制用
SM53	SM153	SM253	SM753	SM853	SM953	运动控制用
SM54	SM154	SM254	SM754	SM854	SM954	运动控制用
SM55	SM155	SM255	SM755	SM855	SM955	运动控制用
SM56	SM156	SM256	SM756	SM856	SM956	运动控制用
SM57	SM157	SM257	SM757	SM857	SM957	运动控制用
SM58	SM158	SM258	SM758	SM858	SM958	运动控制用
SM59	SM159	SM259	SM759	SM859	SM959	运动控制用
SM60	SM160	SM260	SM760	SM860	SM960	电子凸轮端子启动
SM61	SM161	SM261	SM761	SM861	SM961	运动控制用
SM62	SM162	SM262	SM762	SM862	SM962	运动控制用
SM63	SM163	SM263	SM763	SM863	SM963	运动控制用
SM64	SM164	SM264	SM764	SM864	SM964	运动控制用
SM65	SM165	SM265	SM765	SM865	SM965	运动控制用
SM66	SM166	SM266	SM766	SM866	SM966	运动控制用
SM67	SM167	SM267	SM767	SM867	SM967	运动控制用
SM68	SM168	SM268	SM768	SM868	SM968	电子凸轮使能起始位置
SM69	SM169	SM269	SM769	SM869	SM969	运动控制用
SM70	SM170	SM270	SM770	SM870	SM970	电子凸轮触发模式选择: OFF: 软件触发 ON: 硬件触发
SM71	SM171	SM271	SM771	SM871	SM971	电子凸轮输入来源选择 OFF: 内部虚拟连接 ON: 外部输入
SM72	SM172	SM272	SM772	SM872	SM972	速比计算使能
SM73	SM173	SM273	SM773	SM873	SM973	周期使能
SM74	SM174	SM274	SM774	SM874	SM974	运动控制用
SM75	SM175	SM275	SM775	SM875	SM975	电子凸轮延时启动
SM76	SM176	SM276	SM776	SM876	SM976	运动控制用
SM77	SM177	SM277	SM777	SM877	SM977	运动控制用
SM78	SM178	SM278	SM778	SM878	SM978	启动电子凸轮

SM79	SM179	SM279	SM779	SM879	SM979	凸轮周期完成标志位
SM80	SM180	SM280	SM780	SM880	SM980	停止标志
SM81	SM181	SM281	SM781	SM881	SM981	停止模式选择
SM82	SM182	SM282	SM782	SM882	SM982	关键点修改完成标志
SM83	SM183	SM283	SM783	SM883	SM983	关键点修改模式
SM84	SM184	SM284	SM784	SM884	SM984	运动控制用
SM85	SM185	SM285	SM785	SM885	SM985	运动控制用
SM86	SM186	SM286	SM786	SM886	SM986	运动控制用
SM87	SM187	SM287	SM787	SM887	SM987	运动控制用
SM88	SM188	SM288	SM788	SM888	SM988	运动控制用
SM89	SM189	SM289	SM789	SM889	SM989	凸轮运行中标志/强制停止
SM90	SM190	SM290	SM790	SM890	SM990	运动控制子程序使能
SM91	SM191	SM291	SM791	SM891	SM991	运动控制子程序执行完成
SM92	SM192	SM292	SM792	SM892	SM992	运动控制用
SM93	SM193	SM293	SM793	SM893	SM993	运动控制用
SM94	SM194	SM294	SM794	SM894	SM994	运动控制用
SM95	SM195	SM295	SM795	SM895	SM995	运动控制用
SM96	SM196	SM296	SM796	SM896	SM996	运动控制用
SM97	SM197	SM297	SM797	SM897	SM997	运动控制用
SM98	SM198	SM298	SM798	SM898	SM998	运动控制用
SM99	SM199	SM299	SM799	SM899	SM999	运动控制用

SD 寄存器分配

轴 1	轴 2	轴 3	轴 4	Y0	Y1	说明
SD0	SD100	SD200	SD700	SD800	SD900	正向极限端口号
SD1	SD101	SD201	SD701	SD801	SD901	负向极限端口号
SD2	SD102	SD202	SD702	SD802	SD902	运动控制用
SD3	SD103	SD203	SD703	SD803	SD903	运动控制用
SD4	SD104	SD204	SD704	SD804	SD904	运动控制用
SD5	SD105	SD205	SD705	SD805	SD905	运动控制用
SD6	SD106	SD206	SD706	SD806	SD906	马达旋转一圈所需脉冲数
SD7	SD107	SD207	SD707	SD807	SD907	马达旋转一圈所需脉冲数
SD8	SD108	SD208	SD708	SD808	SD908	马达旋转一圈运动距离
SD9	SD109	SD209	SD709	SD809	SD909	马达旋转一圈运动距离
SD10	SD110	SD210	SD710	SD810	SD910	G00 最高速度
SD11	SD111	SD211	SD711	SD811	SD911	G00 最高速度
SD12	SD112	SD212	SD712	SD812	SD912	G00 基底速度
SD13	SD113	SD213	SD713	SD813	SD913	G00 轴基底速度
SD14	SD114	SD214	SD714	SD814	SD914	运动控制用
SD15	SD115	SD215	SD715	SD815	SD915	运动控制用
SD16	SD116	SD216	SD716	SD816	SD916	运动控制用
SD17	SD117	SD217	SD717	SD817	SD917	运动控制用
SD18	SD118	SD218	SD718	SD818	SD918	运动控制用

SD19	SD119	SD219	SD719	SD819	SD919	运动控制用
SD20	SD120	SD220	SD720	SD820	SD920	G00 加速时间
SD21	SD121	SD221	SD721	SD821	SD921	G00 减速时间
SD22	SD122	SD222	SD722	SD822	SD922	运动控制用
SD23	SD123	SD223	SD723	SD823	SD923	运动控制用
SD24	SD124	SD224	SD724	SD824	SD924	运动控制用
SD25	SD125	SD225	SD725	SD825	SD925	运动控制用
SD26	SD126	SD226	SD726	SD826	SD926	电气零点位置
SD27	SD127	SD227	SD727	SD827	SD927	电气零点位置
SD28	SD128	SD228	SD728	SD828	SD928	目标位置 I
SD29	SD129	SD229	SD729	SD829	SD929	目标位置 I
SD30	SD130	SD230	SD730	SD830	SD930	运转速度 I
SD31	SD131	SD231	SD731	SD831	SD931	运转速度 I
SD32	SD132	SD232	SD732	SD832	SD932	目标位置 II
SD33	SD133	SD233	SD733	SD833	SD933	目标位置 II
SD34	SD134	SD234	SD734	SD834	SD934	运转速度 II
SD35	SD135	SD235	SD735	SD835	SD935	运转速度 II
SD36	SD136	SD236	SD736	SD836	SD936	当前位置 (PLS)
SD37	SD137	SD237	SD737	SD837	SD937	当前位置 (PLS)
SD38	SD138	SD238	SD738	SD838	SD938	当前速度 (PPS)
SD39	SD139	SD239	SD739	SD839	SD939	当前速度 (PPS)
SD40	SD140	SD240	SD740	SD840	SD940	当前位置 (mm)
SD41	SD141	SD241	SD741	SD841	SD941	当前位置 (mm)
SD42	SD142	SD242	SD742	SD842	SD942	当前速度 (mm/min)
SD43	SD143	SD243	SD743	SD843	SD943	当前速度 (mm/min)
SD44	SD144	SD244	SD744	SD844	SD944	电子齿轮分子
SD45	SD145	SD245	SD745	SD845	SD945	电子齿轮分母
SD46	SD146	SD246	SD746	SD846	SD946	当前输入频率
SD47	SD147	SD247	SD747	SD847	SD947	当前输入频率
SD48	SD148	SD248	SD748	SD848	SD948	电子凸轮累计输入脉冲数
SD49	SD149	SD249	SD749	SD849	SD949	电子凸轮累计输入脉冲数
SD50	SD150	SD250	SD750	SD850	SD950	偏移补偿值
SD51	SD151	SD251	SD751	SD851	SD951	偏移补偿值
SD52	SD152	SD252	SD752	SD852	SD952	圆心坐标偏移补偿值
SD53	SD153	SD253	SD753	SD853	SD953	圆心坐标偏移补偿值
SD54	SD154	SD254	SD754	SD854	SD954	运动控制用
SD55	SD155	SD255	SD755	SD855	SD955	运动控制用
SD56	SD156	SD256	SD756	SD856	SD956	脉冲输入值计数
SD57	SD157	SD257	SD757	SD857	SD957	脉冲输入值计数
SD58	SD158	SD258	SD758	SD858	SD958	运动控制用
SD59	SD159	SD259	SD759	SD859	SD959	运动控制用
SD60	SD160	SD260	SD760	SD860	SD960	脉冲输入设置
SD61	SD161	SD261	SD761	SD861	SD961	脉冲输出设置

SD62	SD162	SD262	SD762	SD862	SD962	运动控制用
SD63	SD163	SD263	SD763	SD863	SD963	运动控制用
SD64	SD164	SD264	SD764	SD864	SD964	运动控制用
SD65	SD165	SD265	SD765	SD865	SD965	回零等待时间，默认 1000 (ms)
SD66	SD166	SD266	SD766	SD866	SD966	运动控制用
SD67	SD167	SD267	SD767	SD867	SD967	运动控制用
SD68	SD168	SD268	SD768	SD868	SD968	电子凸轮使能起始位置
SD69	SD169	SD269	SD769	SD869	SD969	电子凸轮使能起始位置
SD70	SD170	SD270	SD770	SD870	SD970	凸轮表格设定
SD71	SD171	SD271	SD771	SD871	SD971	凸轮输入轴号设定
SD72	SD172	SD272	SD772	SD872	SD972	非周期执行的次数
SD73	SD173	SD273	SD773	SD873	SD973	凸轮从轴缩放比例
SD74	SD174	SD274	SD774	SD874	SD974	最大速比（浮点数）
SD75	SD175	SD275	SD775	SD875	SD975	最大速比（浮点数）
SD76	SD176	SD276	SD776	SD876	SD976	最小速比（浮点数）
SD77	SD177	SD277	SD777	SD877	SD977	最小速比（浮点数）
SD78	SD178	SD278	SD778	SD878	SD978	延时启动脉冲数
SD79	SD179	SD279	SD779	SD879	SD979	延时启动脉冲数
SD80	SD180	SD280	SD780	SD880	SD980	运动控制用
SD81	SD181	SD281	SD781	SD881	SD981	凸轮已执行周期数
SD82	SD182	SD282	SD782	SD882	SD982	凸轮已执行周期数
SD83	SD183	SD283	SD783	SD883	SD983	运动控制用
SD84	SD184	SD284	SD784	SD884	SD984	电子凸轮方向选择设置
SD85	SD185	SD285	SD785	SD885	SD985	主轴周期长度
SD86	SD186	SD286	SD786	SD886	SD986	主轴周期长度
SD87	SD187	SD287	SD787	SD887	SD987	控制周期(默认 500) (us)
SD88	SD188	SD288	SD788	SD888	SD988	实时控制周期时间监控 s(us)
SD89	SD189	SD289	SD789	SD889	SD989	最大控制周期时间监控(us)
SD90	SD190	SD290	SD790	SD890	SD990	运动控制子程序标号设定
SD91	SD191	SD291	SD791	SD891	SD991	(内部使用)
SD92	SD192	SD292	SD792	SD892	SD992	(内部使用)
SD93	SD193	SD293	SD793	SD893	SD993	(内部使用)
SD94	SD194	SD294	SD794	SD894	SD994	运动控制用
SD95	SD195	SD295	SD795	SD895	SD995	运动控制用
SD96	SD196	SD296	SD796	SD896	SD996	运动控制用
SD97	SD197	SD297	SD797	SD897	SD997	运动控制用
SD98	SD198	SD298	SD798	SD898	SD998	运动控制用
SD99	SD199	SD299	SD799	SD899	SD999	运动控制用

M8000 标志位、D8000 寄存器分配

M 元件	元件功能	D 元件	元件功能
M8000	程序运行状态	D8000	程序监视定时器
M8001	M8000 状态取反	D8001	ARM 版本
M8002	程序开始运行状态	D8002	用户程序最大容量
M8003	M8002 状态取反	D8003	用户数据最大容量
M8004	监控 PLC 系统错误	D8004	错误的 M8060~M8067 的 BCD 值
M8005	系统保留	D8005	电池电压的 BCD 当前值
M8006	系统保留	D8006	电池电压过低的检测值
M8007	系统保留	D8007	保存 M8007 动作的次数
M8008	系统保留	D8008	交流失电检测时间(ms)
M8009	系统保留	D8009	错误显示轴号
M8010	系统保留	D8010	当前扫描时间
M8011	10ms 震荡时钟	D8011	扫描时间最小值(0.1ms)
M8012	100ms 震荡时钟	D8012	扫描时间最大值(0.1ms)
M8013	1s 震荡时钟	D8013	实时时钟秒
M8014	1 分钟震荡时钟	D8014	实时时钟分
M8015	时钟停止和预置	D8015	实时时钟时
M8016	时钟读取显示停止	D8016	实时时钟日
M8017	±30 秒修正	D8017	实时时钟月
M8018	系统保留	D8018	实时时钟年
M8019	实时时钟 RTC 出错	D8019	实时时钟星期
M8020	运算零标志	D8020	X0~X7 通用输入滤波常数
M8021	运算借位标志	D8021	X0~X7 高速输入滤波常数
M8022	运算进位标志	D8022	FPGA 版本号(低 16 位)
M8023	系统保留	D8023	FPGA 版本号(高 16 位)
M8024	BMOV 指令的方向	D8024	库文件版本
M8025	系统保留	D8025	硬件版本
M8026	系统保留	D8026	系统保留
M8027	系统保留	D8027	ARM 的温度
M8028	系统保留	D8028	系统保留
M8029	多周期指令执行完成	D8029	系统保留
M8030	系统保留	D8030	ADC0 读取值
M8031	系统保留	D8031	ADC1 读取值
M8032	系统保留	D8032	梯形图执行次数低位
M8033	软元件状态不变	D8033	梯形图执行次数高位
M8034	PLC 所有输出为 OFF	D8034	ADC0 的滤波值
M8035	系统保留	D8035	ADC1 的滤波值
M8036	系统保留	D8036	系统保留
M8037	系统保留	D8037	系统保留
M8038	系统保留	D8038	系统保留
M8039	系统保留	D8039	系统保留

M8040	系统保留	D8040	系统保留
M8041	系统保留	D8041	系统保留
M8042	系统保留	D8042	系统保留
M8043	系统保留	D8043	系统保留
M8044	系统保留	D8044	系统保留
M8045	系统保留	D8045	系统保留
M8046	系统保留	D8046	系统保留
M8047	系统保留	D8047	系统保留
M8048	系统保留	D8048	系统保留
M8049	信号报警有效	D8049	系统保留
M8050	I00□(X0)中断禁止	D8050	CPU 唯一 id
M8051	I10□(X1)中断禁止	D8051	CPU 唯一 id
M8052	I20□(X2)中断禁止	D8052	CPU 唯一 id
M8053	I30□(X3)中断禁止	D8053	CPU 唯一 id
M8054	I40□(X4)中断禁止	D8054	CPU 唯一 id
M8055	系统保留	D8055	CPU 唯一 id
M8056	驱动 I6□□(定时 0)中断禁止	D8056	CPU 的 flashsize
M8057	驱动 I7□□(定时 1)中断禁止	D8057	CPU 的 flashsize
M8058	驱动 I8□□(定时 2)中断禁止	D8058	CPU 的唯一 id 校验码
M8059	系统保留	D8059	上电时的 AD 采样值
M8060	I/O 构成错误	D8060	I/O 错误起始地址号
M8061	PLC 硬件错误	D8061	PLC 硬件错误代码序号
M8062	PLC 通讯错误	D8062	PLC 通讯错误代码
M8063	联机/通讯错误	D8063	并行联机错误代码
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	锁存错误程序的步号
M8069	系统错误标志	D8069	M8065~M8067 的错误步号
M8070	系统保留	D8070	系统保留
M8071	系统保留	D8071	系统保留
M8072	系统保留	D8072	系统保留
M8073	系统保留	D8073	系统保留
M8074	系统保留	D8074	系统保留
M8075	系统保留	D8075	系统保留
M8076	系统保留	D8076	系统保留
M8077	系统保留	D8077	系统保留
M8078	系统保留	D8078	系统保留
M8079	系统保留	D8079	系统保留
M8080	系统保留	D8080	系统保留
M8081	系统保留	D8081	系统保留
M8082	系统保留	D8082	系统保留

M8083	系统保留	D8083	系统保留
M8084	系统保留	D8084	系统保留
M8085	系统保留	D8085	系统保留
M8086	系统保留	D8086	系统保留
M8087	系统保留	D8087	系统保留
M8088	系统保留	D8088	系统保留
M8089	系统保留	D8089	系统保留
M8090	系统保留	D8090	系统保留
M8091	系统保留	D8091	系统保留
M8092	系统保留	D8092	系统保留
M8093	系统保留	D8093	系统保留
M8094	系统保留	D8094	系统保留
M8095	系统保留	D8095	系统保留
M8096	系统保留	D8096	系统保留
M8097	系统保留	D8097	系统保留
M8098	系统保留	D8098	系统保留
M8099	系统保留	D8099	系统保留
M8100	系统保留	D8100	系统保留
M8101	系统保留	D8101	系统保留
M8102	系统保留	D8102	系统保留
M8103	系统保留	D8103	系统保留
M8104	系统保留	D8104	系统保留
M8105	系统保留	D8105	系统保留
M8106	系统保留	D8106	内部占用
M8107	系统保留	D8107	内部占用
M8108	系统保留	D8108	内部占用
M8109	系统保留	D8109	内部占用
M8110	系统保留	D8110	COM0 通讯格式设定
M8111	系统保留	D8111	COM0 通讯站号设定
M8112	系统保留	D8112	COM0 通讯格式设定
M8113	系统保留	D8113	系统保留
M8114	系统保留	D8114	系统保留
M8115	系统保留	D8115	系统保留
M8116	系统保留	D8116	COM0 通讯协议设定
M8117	系统保留	D8117	系统保留
M8118	系统保留	D8118	Modbus 通讯出错站号
M8119	系统保留	D8119	通讯超时时间判断(100ms)
M8120	系统保留	D8120	COM1 通讯格式设定
M8121	系统保留	D8121	COM1 通讯站号设定
M8122	Modbus 执行状态/RS 发送标志	D8122	COM1 通讯格式设置/RS 传送剩余量
M8123	系统保留	D8123	系统保留
M8124	系统保留	D8124	系统保留
M8125	系统保留	D8125	系统保留

M8126	系统保留	D8126	COM1 通讯协议设定
M8127	系统保留	D8127	系统保留
M8128	系统保留	D8128	系统保留
M8129	系统保留	D8129	COM1 通讯超时时间判断(100ms)
M8130	系统保留	D8130	系统保留
M8131	系统保留	D8131	系统保留
M8132	系统保留	D8132	系统保留
M8133	系统保留	D8133	系统保留
M8134	系统保留	D8134	系统保留
M8135	系统保留	D8135	系统保留
M8136	系统保留	D8136	系统保留
M8137	系统保留	D8137	系统保留
M8138	系统保留	D8138	系统保留
M8139	系统保留	D8139	系统保留
M8140	系统保留	D8140	系统保留
M8141	系统保留	D8141	系统保留
M8142	系统保留	D8142	系统保留
M8143	系统保留	D8143	系统保留
M8144	系统保留	D8144	系统保留
M8145	系统保留	D8145	系统保留
M8146	系统保留	D8146	系统保留
M8147	系统保留	D8147	系统保留
M8148	系统保留	D8148	系统保留
M8149	系统保留	D8149	系统保留
M8150	探针 1, 使能	D8150	探针 1, 模式设定
M8151	探针 1, 状态	D8151	探针 1, 锁存位置低位
M8152	系统保留	D8152	探针 1, 锁存位置高位
M8153	探针 2, 使能	D8153	探针 2, 模式设定
M8154	探针 2, 状态	D8154	探针 2, 锁存位置低位
M8155	系统保留	D8155	探针 2, 锁存位置高位
M8156	探针 3, 使能	D8156	探针 3, 模式设定
M8157	探针 3, 状态	D8157	探针 3, 锁存位置低位
M8158	系统保留	D8158	探针 3, 锁存位置高位
M8159	系统保留	D8159	系统保留
M8160	(XCH) 的 SWAP 功能	D8160	系统保留
M8161	8/16 位处理模式	D8161	系统保留
M8162	系统保留	D8162	探针 1, 锁存相位偏差低位
M8163	BINDA 输出字符切换标志	D8163	探针 1, 锁存相位偏差高位
M8164	系统保留	D8164	探针 2, 锁存相位偏差低位
M8165	系统保留	D8165	探针 2, 锁存相位偏差高位
M8166	系统保留	D8166	探针 3, 锁存相位偏差低位
M8167	系统保留	D8167	探针 3, 锁存相位偏差高位
M8168	系统保留	D8168	系统保留

M8169	系统保留	D8169	系统保留
M8170	系统保留	D8170	系统保留
M8171	系统保留	D8171	系统保留
M8172	系统保留	D8172	系统保留
M8173	系统保留	D8173	系统保留
M8174	系统保留	D8174	系统保留
M8175	系统保留	D8175	系统保留
M8176	系统保留	D8176	系统保留
M8177	系统保留	D8177	系统保留
M8178	系统保留	D8178	系统保留
M8179	系统保留	D8179	系统保留
M8180	系统保留	D8180	系统保留
M8181	系统保留	D8181	系统保留
M8182	系统保留	D8182	Z1 寄存器
M8183	系统保留	D8183	V1 寄存器
M8184	系统保留	D8184	Z2 寄存器
M8185	系统保留	D8185	V2 寄存器
M8186	系统保留	D8186	Z3 寄存器
M8187	系统保留	D8187	V3 寄存器
M8188	系统保留	D8188	Z4 寄存器
M8189	系统保留	D8189	V4 寄存器
M8190	系统保留	D8190	Z5 寄存器
M8191	系统保留	D8191	V5 寄存器
M8192	系统保留	D8192	Z6 寄存器
M8193	系统保留	D8193	V6 寄存器
M8194	系统保留	D8194	Z7 寄存器
M8195	系统保留	D8195	V7 寄存器
M8196	系统保留	D8196	系统保留
M8197	系统保留	D8197	系统保留
M8198	系统保留	D8198	系统保留
M8199	系统保留	D8199	系统保留
M8200	系统保留	D8200	ARM 辅版本
M8201	系统保留	D8201	系统保留
M8202	系统保留	D8202	系统保留
M8203	系统保留	D8203	系统保留
M8204	系统保留	D8204	系统保留
M8205	系统保留	D8205	系统保留
M8206	系统保留	D8206	系统保留
M8207	系统保留	D8207	系统保留
M8208	系统保留	D8208	系统保留
M8209	系统保留	D8209	系统保留
M8210	系统保留	D8210	系统保留
M8211	系统保留	D8211	系统保留

M8212	系统保留	D8212	系统保留
M8213	系统保留	D8213	系统保留
M8214	系统保留	D8214	系统保留
M8215	系统保留	D8215	系统保留
M8216	系统保留	D8216	系统保留
M8217	系统保留	D8217	系统保留
M8218	系统保留	D8218	系统保留
M8219	系统保留	D8219	系统保留
M8220	系统保留	D8220	系统保留
M8221	系统保留	D8221	系统保留
M8222	系统保留	D8222	系统保留
M8223	系统保留	D8223	系统保留
M8224	系统保留	D8224	系统保留
M8225	系统保留	D8225	系统保留
M8226	系统保留	D8226	系统保留
M8227	系统保留	D8227	系统保留
M8228	系统保留	D8228	系统保留
M8229	系统保留	D8229	系统保留
M8230	系统保留	D8230	系统保留
M8231	系统保留	D8231	系统保留
M8232	系统保留	D8232	系统保留
M8233	系统保留	D8233	系统保留
M8234	系统保留	D8234	系统保留
M8235	C235 高速计数器方向控制	D8235	系统保留
M8236	C236 高速计数器方向控制	D8236	系统保留
M8237	C237 高速计数器方向控制	D8237	系统保留
M8238	系统保留	D8238	系统保留
M8239	系统保留	D8239	系统保留
M8240	系统保留	D8240	CAN 功能占用, 不可使用
M8241	系统保留	D8241	CAN 功能占用, 不可使用
M8242	系统保留	D8242	CAN 功能占用, 不可使用
M8243	系统保留	D8243	CAN 功能占用, 不可使用
M8244	系统保留	D8244	CAN 功能占用, 不可使用
M8245	系统保留	D8245	CAN 功能占用, 不可使用
M8246	系统保留	D8246	CAN 功能占用, 不可使用
M8247	系统保留	D8247	CAN 功能占用, 不可使用
M8248	系统保留	D8248	CAN 功能占用, 不可使用
M8249	系统保留	D8249	CAN 功能占用, 不可使用
M8250	系统保留	D8250	CAN 功能占用, 不可使用
M8251	系统保留	D8251	CAN 功能占用, 不可使用
M8252	系统保留	D8252	CAN 功能占用, 不可使用
M8253	系统保留	D8253	CAN 功能占用, 不可使用
M8254	系统保留	D8254	CAN 功能占用, 不可使用

M8255	系统保留	D8255	CAN 功能占用，不可使用
M8256	系统保留	D8256	系统保留
M8257	系统保留	D8257	系统保留
M8258	系统保留	D8258	系统保留
M8259	系统保留	D8259	系统保留
M8260	modbus 主站切换	D8260	COM2 通讯格式设定
M8261	系统保留	D8261	COM2 通讯站号设定
M8262	com2-Modbus 执行状态/RS 发送标志	D8262	COM2 通讯格式设置/RS 传送剩余量
M8263	系统保留	D8263	系统保留
M8264	系统保留	D8264	系统保留
M8265	系统保留	D8265	系统保留
M8266	系统保留	D8266	COM2 通讯协议设定
M8267	系统保留	D8267	系统保留
M8268	系统保留	D8268	系统保留
M8269	系统保留	D8269	COM2 通讯超时时间判断(100ms)
M8270	系统保留	D8270	系统保留
M8271	系统保留	D8271	系统保留
M8272	系统保留	D8272	系统保留
M8273	系统保留	D8273	系统保留
M8274	系统保留	D8274	系统保留
M8275	系统保留	D8275	系统保留
M8276	系统保留	D8276	系统保留
M8277	系统保留	D8277	系统保留
M8278	系统保留	D8278	系统保留
M8279	系统保留	D8279	系统保留
M8280	CAN 协议切换标志	D8280	CAN 生效协议显示
M8281	系统保留	D8281	系统保留
M8282	系统保留	D8282	CanLink 心跳
M8283	生效 CAN 在线监控地址	D8283	CAN 在线监控起始地址
M8284	设定 CAN 地址	D8284	CAN 地址设定/显示地址
M8285	设定波特率	D8285	波特率显示
M8286	系统保留	D8286	CANlink 波特率设置
M8287	系统保留	D8287	CANopen 配置错误站号
M8288	系统保留	D8288	CANopen 配置错误编号
M8289	系统保留	D8289	CAN 总线错误
M8290	系统保留	D8290	CAN 接收错误
M8291	系统保留	D8291	系统保留
M8292	系统保留	D8292	系统保留
M8293	系统保留	D8293	系统保留
M8294	系统保留	D8294	系统保留
M8295	系统保留	D8295	系统保留
M8296	设备地址错误	D8296	系统保留
M8297	系统保留	D8297	系统保留

M8298	系统保留	D8298	系统保留
M8299	系统保留	D8299	系统保留
M8300	系统保留	D8300	系统保留
M8301	系统保留	D8301	系统保留
M8302	系统保留	D8302	系统保留
M8303	系统保留	D8303	系统保留
M8304	零位标志	D8304	系统保留
M8305	系统保留	D8305	系统保留
M8306	进位标志	D8306	系统保留
M8307	系统保留	D8307	系统保留
M8308	系统保留	D8308	系统保留
M8309	进位标志	D8309	系统保留
M8310	系统保留	D8310	(RND) 随机数低 16 位
M8311	系统保留	D8311	(RND) 随机数高 16 位
M8312	进位标志	D8312	系统保留
M8313	系统保留	D8313	系统保留
M8314	系统保留	D8314	系统保留
M8315	进位标志	D8315	系统保留
M8316	系统保留	D8316	系统保留
M8317	系统保留	D8317	系统保留
M8318	进位标志	D8318	系统保留
M8319	系统保留	D8319	系统保留
M8320	进位标志	D8320	系统保留
M8321	系统保留	D8321	系统保留
M8322	系统保留	D8322	系统保留
M8323	进位标志	D8323	系统保留
M8324	系统保留	D8324	系统保留
M8325	系统保留	D8325	系统保留
M8326	进位标志	D8326	系统保留
M8327	系统保留	D8327	系统保留
M8328	系统保留	D8328	系统保留
M8329	进位标志	D8329	系统保留
M8330	进位标志	D8330	系统保留
M8331	系统保留	D8331	系统保留
M8332	系统保留	D8332	系统保留
M8333	进位标志	D8333	系统保留
M8334	系统保留	D8334	系统保留
M8335	系统保留	D8335	系统保留
M8336	进位标志	D8336	系统保留
M8337	系统保留	D8337	系统保留
M8338	系统保留	D8338	系统保留
M8339	进位标志	D8339	系统保留
M8340	Y300 脉冲输出中监控	D8340	Y300 当前值寄存器 (PLS, 低 16 位)

M8341	Y300 清零信号输出有效	D8341	Y300 当前值寄存器(PLS, 高 16 位)
M8342	Y300 原点回归方向指定	D8342	Y300 最高速度(Hz, 低 16 位)
M8343	Y300 正转极限	D8343	Y300 最高速度(Hz, 高 16 位)
M8344	Y300 反转极限	D8344	Y300 原点回归速度(Hz, 低 16 位)
M8345	Y300 近点信号逻辑反转	D8345	Y300 原点回归速度(Hz, 高 16 位)
M8346	Y300 零点信号逻辑反转	D8346	Y300 爬行速度(Hz)
M8347	Y300S 曲线加减速使能	D8347	Y300 基底速度(Hz)
M8348	Y300 回零后保留当前位置的标志	D8348	Y300 加速时间(ms)
M8349	Y300 脉冲输出停止标志	D8349	Y300 减速时间(ms)
M8350	Y300 加减速时间设置和脉冲更改有效	D8350	Y300 清零软元件编号
M8351	系统保留	D8351	Y300S 曲线滤波值
M8352	Y300 输出完成中断使能	D8352	系统保留
M8353	Y300 加速度时间修改使能	D8353	系统保留
M8354	Y300 异常结束标志位	D8354	系统保留
M8355	Y300PLSV2 指令加速中	D8355	系统保留
M8356	Y300PLSV2 指令减速中	D8356	Y300 当前值实时寄存器(PLS, 低 16 位)
M8357	Y300 允许减速反向	D8357	Y300 当前值实时寄存器(PLS, 高 16 位)
M8358	系统保留	D8358	系统保留
M8359	系统保留	D8359	系统保留
M8360	Y304 脉冲输出中监控	D8360	Y304 当前值寄存器(PLS, 低 16 位)
M8361	Y304 清零信号输出有效	D8361	Y304 当前值寄存器(PLS, 高 16 位)
M8362	Y304 原点回归方向指定	D8362	Y304 最高速度(Hz, 低 16 位)
M8363	Y304 正转极限	D8363	Y304 最高速度(Hz, 高 16 位)
M8364	Y304 反转极限	D8364	Y304 原点回归速度(Hz, 低 16 位)
M8365	Y304 近点信号逻辑反转	D8365	Y304 原点回归速度(Hz, 高 16 位)
M8366	Y304 零点信号逻辑反转	D8366	Y304 爬行速度(Hz)
M8367	Y304S 曲线加减速使能	D8367	Y304 基底速度(Hz)
M8368	Y304 回零后保留当前位置的标志	D8368	Y304 加速时间(ms)
M8369	Y304 脉冲输出停止标志	D8369	Y304 减速时间(ms)
M8370	Y304 加减速时间设置和脉冲更改有效	D8370	Y304 清零软元件编号
M8371	系统保留	D8371	Y304S 曲线滤波值
M8372	Y304 输出完成中断使能	D8372	系统保留
M8373	Y304 加速度时间修改使能	D8373	系统保留
M8374	Y304 异常结束标志位	D8374	系统保留
M8375	Y304PLSV2 指令加速中	D8375	系统保留
M8376	Y304PLSV2 指令减速中	D8376	Y304 当前值实时寄存器(PLS, 低 16 位)
M8377	Y304 允许减速反向	D8377	Y304 当前值实时寄存器(PLS, 高 16 位)
M8378	系统保留	D8378	系统保留
M8379	系统保留	D8379	系统保留
M8380	Y310 脉冲输出中监控	D8380	Y310 当前值寄存器(PLS, 低 16 位)
M8381	Y310 清零信号输出有效标志	D8381	Y310 当前值寄存器(PLS, 高 16 位)
M8382	Y310 原点回归方向指定	D8382	Y310 最高速度(Hz, 低 16 位)
M8383	Y310 正转极限	D8383	Y310 最高速度(Hz, 高 16 位)

M8384	Y310 反转极限	D8384	Y310 原点回归速度(Hz, 低 16 位)
M8385	Y310 近点信号逻辑反转	D8385	Y310 原点回归速度(Hz, 高 16 位)
M8386	Y310 零点信号逻辑反转	D8386	Y310 爬行速度(Hz)
M8387	Y310S 曲线加减速使能	D8387	Y310 基底速度(Hz)
M8388	Y310 回零后保留当前位置的标志	D8388	Y310 加速时间(ms)
M8389	Y310 脉冲输出停止标志	D8389	Y310 减速时间(ms)
M8390	Y310 加减速时间设置和脉冲更改有效	D8390	Y310 清零软元件编号
M8391	系统保留	D8391	Y310S 曲线滤波值
M8392	Y310 输出完成中断使能	D8392	系统保留
M8393	Y310 加速度时间修改使能	D8393	系统保留
M8394	Y310 异常结束标志位	D8394	系统保留
M8395	Y310PLSV2 指令加速中	D8395	系统保留
M8396	Y310PLSV2 指令减速中	D8396	Y310 当前值实时寄存器(PLS, 低 16 位)
M8397	Y310 允许减速反向	D8397	Y310 当前值实时寄存器(PLS, 高 16 位)
M8398	系统保留	D8398	系统保留
M8399	系统保留	D8399	系统保留
M8400	Y314 脉冲输出中监控	D8400	Y314 当前值寄存器(PLS, 低 16 位)
M8401	Y314 清零信号输出有效标志	D8401	Y314 当前值寄存器(PLS, 高 16 位)
M8402	Y314 原点回归方向指定	D8402	Y314 最高速度(Hz, 低 16 位)
M8403	Y314 正转极限	D8403	Y314 最高速度(Hz, 高 16 位)
M8404	Y314 反转极限	D8404	Y314 原点回归速度(Hz, 低 16 位)
M8405	Y314 近点信号逻辑反转	D8405	Y314 原点回归速度(Hz, 高 16 位)
M8406	Y314 零点信号逻辑反转	D8406	Y314 爬行速度(Hz)
M8407	Y314S 曲线加减速使能	D8407	Y314 基底速度(Hz)
M8408	Y314 回零后保留当前位置的标志	D8408	Y314 加速时间(ms)
M8409	Y314 脉冲输出停止标志	D8409	Y314 减速时间(ms)
M8410	Y314 加减速时间设置和脉冲更改有效	D8410	Y314 清零软元件编号
M8411	系统保留	D8411	Y314S 曲线滤波值
M8412	Y314 输出完成中断使能	D8412	系统保留
M8413	Y314 加速度时间修改使能	D8413	系统保留
M8414	Y314 异常结束标志位	D8414	系统保留
M8415	Y314PLSV2 指令加速中	D8415	系统保留
M8416	Y314PLSV2 指令减速中	D8416	Y314 当前值实时寄存器(PLS, 低 16 位)
M8417	Y314 允许减速反向	D8417	Y314 当前值实时寄存器(PLS, 高 16 位)
M8418	系统保留	D8418	系统保留
M8419	系统保留	D8419	系统保留
M8420	Y0 脉冲输出中监控	D8420	Y0 当前值寄存器(PLS, 低 16 位)
M8421	Y0 清零信号输出有效标志	D8421	Y0 当前值寄存器(PLS, 高 16 位)
M8422	Y0 原点回归方向指定	D8422	Y0 最高速度(Hz, 低 16 位)
M8423	Y0 正转极限	D8423	Y0 最高速度(Hz, 高 16 位)
M8424	Y0 反转极限	D8424	Y0 原点回归速度(Hz, 低 16 位)
M8425	Y0 近点信号逻辑反转	D8425	Y0 原点回归速度(Hz, 高 16 位)
M8426	Y0 零点信号逻辑反转	D8426	Y0 爬行速度(Hz)

M8427	Y0S 曲线加减速使能	D8427	Y0 基底速度(Hz)
M8428	Y0 回零后保留当前位置的标志	D8428	Y0 加速时间(ms)
M8429	Y0 脉冲输出停止标志	D8429	Y0 减速时间(ms)
M8430	Y0 加减速时间设置和脉冲更改有效	D8430	Y0 清零软元件编号
M8431	系统保留	D8431	Y0S 曲线滤波值
M8432	Y0 输出完成中断使能	D8432	系统保留
M8433	Y0 加速度时间修改使能	D8433	系统保留
M8434	Y0 异常结束标志位	D8434	系统保留
M8435	Y0PLSV2 指令加速中	D8435	系统保留
M8436	Y0PLSV2 指令减速中	D8436	Y0 当前值实时寄存器(PLS, 低 16 位)
M8437	Y0 允许减速反向	D8437	Y0 当前值实时寄存器(PLS, 高 16 位)
M8438	系统保留	D8438	系统保留
M8439	系统保留	D8439	系统保留
M8440	Y1 脉冲输出中监控	D8440	Y1 当前值寄存器(PLS, 低 16 位)
M8441	Y1 清零信号输出有效	D8441	Y1 当前值寄存器(PLS, 高 16 位)
M8442	Y1 原点回归方向指定	D8442	Y1 最高速度(Hz, 低 16 位)
M8443	Y1 正转极限	D8443	Y1 最高速度(Hz, 高 16 位)
M8444	Y1 反转极限	D8444	Y1 原点回归速度(Hz, 低 16 位)
M8445	Y1 近点信号逻辑反转	D8445	Y1 原点回归速度(Hz, 高 16 位)
M8446	Y1 零点信号逻辑反转	D8446	Y1 爬行速度(Hz)
M8447	Y1S 曲线加减速使能	D8447	Y1 基底速度(Hz)
M8448	Y1 回零后保留当前位置的标志	D8448	Y1 加速时间(ms)
M8449	Y1 脉冲输出停止标志	D8449	Y1 减速时间(ms)
M8450	Y1 加减速时间设置和脉冲更改有效	D8450	Y1 清零软元件编号
M8451	系统保留	D8451	Y1S 曲线滤波值
M8452	Y1 输出完成中断使能	D8452	系统保留
M8453	Y1 加速度时间修改使能	D8453	系统保留
M8454	Y1 异常结束标志位	D8454	系统保留
M8455	Y1PLSV2 指令加速中	D8455	系统保留
M8456	Y1PLSV2 指令减速中	D8456	Y1 当前值实时寄存器(PLS, 低 16 位)
M8457	Y1 允许减速反向	D8457	Y1 当前值实时寄存器(PLS, 高 16 位)
M8458	系统保留	D8458	系统保留
M8459	系统保留	D8459	系统保留
M8460	系统保留	D8460	系统保留
M8461	系统保留	D8461	系统保留
M8462	系统保留	D8462	系统保留
M8463	系统保留	D8463	系统保留
M8464	系统保留	D8464	系统保留
M8465	系统保留	D8465	系统保留
M8466	系统保留	D8466	系统保留
M8467	系统保留	D8467	系统保留
M8468	系统保留	D8468	系统保留
M8469	系统保留	D8469	系统保留

M8470	系统保留	D8470	系统保留
M8471	系统保留	D8471	系统保留
M8472	系统保留	D8472	系统保留
M8473	系统保留	D8473	系统保留
M8474	系统保留	D8474	系统保留
M8475	系统保留	D8475	系统保留
M8476	系统保留	D8476	系统保留
M8477	系统保留	D8477	系统保留
M8478	系统保留	D8478	系统保留
M8479	系统保留	D8479	系统保留
M8480	系统保留	D8480	系统保留
M8481	系统保留	D8481	系统保留
M8482	系统保留	D8482	系统保留
M8483	系统保留	D8483	系统保留
M8484	系统保留	D8484	系统保留
M8485	系统保留	D8485	系统保留
M8486	系统保留	D8486	系统保留
M8487	系统保留	D8487	系统保留
M8488	系统保留	D8488	系统保留
M8489	系统保留	D8489	系统保留
M8490	系统保留	D8490	系统保留
M8491	系统保留	D8491	系统保留
M8492	系统保留	D8492	系统保留
M8493	系统保留	D8493	系统保留
M8494	系统保留	D8494	系统保留
M8495	系统保留	D8495	系统保留
M8496	系统保留	D8496	系统保留
M8497	系统保留	D8497	系统保留
M8498	系统保留	D8498	系统保留
M8499	系统保留	D8499	系统保留
M8500	系统保留	D8500	Y0-Yn 定位指令执行时最高速度(低 16 位)
M8501	系统保留	D8501	Y0-Yn 定位指令执行时最高速度(高 16 位)
M8502	系统保留	D8502	Y0-Yn 定位执行时基底速度
M8503	系统保留	D8503	Y0-Yn 定位执行时加减速时间
M8504	系统保留	D8504	系统保留
M8505	系统保留	D8505	系统保留
M8506	系统保留	D8506	系统保留
M8507	系统保留	D8507	系统保留
M8508	系统保留	D8508	系统保留
M8509	系统保留	D8509	系统保留
M8510	系统保留	D8510	系统保留
M8511	系统保留	D8511	系统保留

附录二 系统错误码说明

系统错误码 D8061

错误码	错误表示内容	错误时动作
6101	硬件 SRAM 故障	停止运行
6106	硬件 SPIFLASH 故障	停止运行
6107	系统 IO 设定错误	停止运行
6108	FPGA 加载故障	停止运行
6109	FPGA 版本故障	停止运行
6110	硬件以太网错误	停止运行
6119	硬件电容故障	停止运行
6120	文件系统初始化故障	停止运行
6121	FPGA 文件异常	停止运行
6122	FLASH 操作错误	停止运行
6180	RAM 内存申请出错	停止运行
6181	FLASH 读写出错	停止运行
16100	硬件 EEPROM 故障	停止运行
16101	硬件 422 故障	停止运行
16102	硬件 COM1 故障	停止运行
16103	硬件 COM2 故障	停止运行
16104	硬件 AD 故障	停止运行
16105	硬件时钟故障	停止运行
16110	IO 模块搜索失败	停止运行
16111	IO 模块连续三次收到 ID 号错误	停止运行
16112	IO 模块主站连续三次收到 CRC 校验错误	停止运行
16113	IO 模块连续三次从站报错	停止运行
16114	IO 模块从站掉线	停止运行
16115	IO 模块从站类型错误	停止运行

系统错误码 D8062

错误码	错误表示内容	错误时动作
16204	以太网应答出错	继续运行
16260	机械单位设定值有误（具体轴错误请查看 D8009）	继续运行
16261	电子齿轮比设定值有误（具体轴错误请查看 D8009）	继续运行
16262	使用了后台没有配置的凸轮表（具体轴错误请查看 D8009）	继续运行
16263	电子凸轮未选择外部输入主轴（具体轴错误请查看 D8009）	继续运行
16264	电子凸轮从轴速度过大，超出最大允许输出速度（具体轴错误请查看 D8009）	继续运行
16265	同步下限大于同步上限（具体轴错误请查看 D8009）	继续运行
16266	主轴关键点非法、主轴位置偏差过大（具体轴错误请查看 D8009）	继续运行
16267	延时启动脉冲数设定有误（具体轴错误请查看 D8009）	继续运行
16268	凸轮关键点写入指令，关键点值非法（具体轴错误请查看 D8009）	继续运行
16269	凸轮已加密，不允许指令读取关键点数据（具体轴错误请查看 D8009）	继续运行

16270	电子凸轮从轴缩放比例错误（具体轴错误请查看 D8009）	继续运行
16271	电子凸轮配置单位错误（具体轴错误请查看 D8009）	继续运行
16272	电子凸轮运行中修改不成功（具体轴错误请查看 D8009）	继续运行
16273	电子凸轮修改指令重复使用（具体轴错误请查看 D8009）	继续运行
16274	非周期个数超过最大值（具体轴错误请查看 D8009）	继续运行
16275	飞剪数据不合理（具体轴错误请查看 D8009）	继续运行
16276	飞剪数据不合理（具体轴错误请查看 D8009）	
16278	方向设置不合理（具体轴错误请查看 D8009）	
16279	速比计算出错（具体轴错误请查看 D8009）	
16281	飞剪返回系数设置不合理（具体轴错误请查看 D8009）	继续运行
16282	切刀移动范围设置不合理（具体轴错误请查看 D8009）	继续运行
16283	切刀移动范围、物料长度、同步区长度三者之间的关系不合理（具体轴错误请查看 D8009）	继续运行
16285	关键点的线性类型错误（具体轴错误请查看 D8009）	继续运行
16286	关键点修改个数错误（具体轴错误请查看 D8009）	继续运行
16287	顶杆点数据错误（具体轴错误请查看 D8009）	继续运行
26200	轴通讯错误	继续运行
26201	伺服故障	继续运行
26202	轴未使能	继续运行
26203	轴忙	继续运行
26207	正向超程	继续运行
26208	反向超程	继续运行
26210	轴号超范围或未配置	继续运行
26211	CANOPEN 配置出错	继续运行
26212	CANOPEN 监控地址设置错误	继续运行
26213	CANOPEN 接收溢出	继续运行
26214	原点回归失败	继续运行
26220	轴控指令同时导通条数超过最大值	继续运行

系统错误码 D8063

错误码	错误表示内容	错误时动作
6320	422 通讯地址位错误	继续运行
6321	422 通讯地址字错误	继续运行
6330	COM0_MODBUS 从站地址设置错误，地址大于 247；	继续运行
6331	COM0_数据帧长度错误，帧长度不符合要求，或帧长小于 5；	继续运行
6332	COM0_地址错误，标准错误帧；或收发地址不一致；	继续运行
6333	COM0_CRC 检验错误	继续运行
6334	COM0_不支持的命令码，标准错误帧；或收发命令不一致；或不支持的命令；	继续运行
6335	COM0_接收超时	继续运行
6336	COM0_数据错误，标准错误帧；	继续运行
6337	COM0_缓冲区溢出，暂无	继续运行
6338	COM0_帧错误，标准错误帧；	继续运行
6339	COM0_串口协议错误，使用 modbus 指令或 RS 指令时，未定义成对应的协议；	继续运行

6340	COM1_MODBUS 从站地址设置错误, 地址大于 247;	继续运行
6341	COM1_数据帧长度错误, 帧长度不符合要求, 或帧长小于 5;	继续运行
6342	COM1_地址错误, 标准错误帧; 或收发地址不一致;	继续运行
6343	COM1_CRC 检验错误	继续运行
6344	COM1_不支持的命令码, 标准错误帧; 或收发命令不一致; 或不支持的命令;	继续运行
6345	COM1_接收超时	继续运行
6346	COM1_数据错误, 标准错误帧;	继续运行
6347	COM1_缓冲区溢出, 暂无	继续运行
6348	COM1_帧错误, 标准错误帧;	继续运行
6349	COM1_串口协议错误, 使用 modbus 指令或 RS 指令时, 未定义成对应的协议;	继续运行
6350	COM2_MODBUS 从站地址设置错误, 地址大于 247;	继续运行
6351	COM2_数据帧长度错误, 帧长度不符合要求, 或帧长小于 5;	继续运行
6352	COM2_地址错误, 标准错误帧; 或收发地址不一致;	继续运行
6353	COM2_CRC 检验错误	继续运行
6354	COM2_不支持的命令码, 标准错误帧; 或收发命令不一致; 或不支持的命令;	继续运行
6355	COM2_接收超时	继续运行
6356	COM2_数据错误, 标准错误帧;	继续运行
6357	COM2_缓冲区溢出, 暂无	继续运行
6358	COM2_帧错误, 标准错误帧;	继续运行
6359	COM2_串口协议错误, 使用 modbus 指令或 RS 指令时, 未定义成对应的协议;	继续运行
6380	CAN 下载数据出错	继续运行
6381	CAN 未知错误	继续运行
6382	CANOPEN 发送缓存溢出	继续运行
6383	CANOPEN 接收缓存溢出	继续运行
6384	CAN 一般错误	继续运行
6385	CAN 被动错误	继续运行
6386	CAN 总线关闭	继续运行
6387	CAN 心跳错误	继续运行
6388	CAN 协议错误	继续运行
6389	CANPDO 长度错误	继续运行
6390	CANRPDO 超时	继续运行
6391	CAN 过载	继续运行
6392	CANPDO 发送接收处理错误	继续运行
6393	CANPDO 传输类型错误	继续运行
6394	CAN 接收到错误的报文	继续运行
6395	CAN 收到紧急报文	继续运行
6396	CAN 从站数量超限	继续运行
6397	CANSDO 返回错误的命令码	继续运行
6398	CAN 下载错误	继续运行
6399	CAN 写错误的数据	继续运行

系统错误码 D8064

错误码	错误表示内容	错误时动作
6426	用户程序不完整	停止运行
16400	Gcode 字符串错误	停止运行
16401	插补周期最大值溢出	停止运行
16402	设置控制周期时间错误	停止运行
16403	系统倍率设置错误	停止运行
16404	系统正在运行	停止运行

系统错误码 D8065

错误码	错误表示内容	错误时动作
6503	指令参数错误	停止运行
6504	标号定义重复	停止运行
6506	使用未定义指令	停止运行
6507	标号 P 定义不良	停止运行
6508	标号 I 定义不良	停止运行
6511	高数计数冲突	停止运行
6520	文件打开失败	停止运行
16500	错误的中断 ROM 标签	停止运行
16501	子程序 ROM 后缀	停止运行
16502	梯形图地址错误	停止运行
16503	梯形图指令错误	停止运行
16504	脉冲指令个数超限	停止运行
16505	定义的子程序超过最大编码	停止运行
16506	定义的 T 超过最大编码	停止运行
16507	ZRST 的类型错误	停止运行
16508	ZRST 无效类型	停止运行
16509	ZRST 操作数 1 编号大于操作数 2 编号	停止运行

系统错误码 D8066

错误码	错误表示内容	错误时动作
6627	梯形图无 RET 指令	停止运行
6630	梯形图无 SRET 或 IRET 指令	停止运行
6631	SRET 处于无法使用的位置	停止运行
6632	RET 处于无法使用的位置	停止运行
6633	IRET 处于无法使用的位置	停止运行

系统错误码 D8067

错误码	错误表示内容	错误时动作
6701	跳转标签超限	继续运行

6702	调用超过 6 次	继续运行
6705	VZ 数据计算溢出	继续运行
6706	数据不合理或返回超限	继续运行
6720	CALL 与 SRET 不对应	继续运行
6713	定位位置偏小，不满足减速	继续运行
6714	当前运动模式有误或此时正在运动	继续运行
6715	减速停止未断开，再次驱动无效	继续运行
6716	轴还在运动，切换到凸轮模式失败	继续运行
6732	输入滤波参数不正确	继续运行
6748	输入输出类型错误	继续运行
6749	输入输出个数错误	继续运行
6760	高速计数超出限制	继续运行
6761	高速计数输出地址错误，元件范围超限	继续运行
6762	高速计数输出类型错误	继续运行
6763	高速计数输入计数端口元件超过范围	继续运行
6770	高速端口冲突	继续运行
6771	脉冲指令端口错误	继续运行
6772	MC 指令错误	继续运行
6773	调用 MC 子程序错误的子程序编号	继续运行
6775	MC 子程序位置错误	继续运行
6776	探针设置错误	继续运行
6780	PID 采样周期小于等于 0	继续运行
6782	PID 滤波时间错误	继续运行
6783	PID 比例系数错误	继续运行
6784	PID 积分系数错误	继续运行
6785	PID 微分系数错误	继续运行
6786	PID 下限大于上限	继续运行
6787	PID 增益选择错误	继续运行
16700	MC 地址错误	继续运行
16701	MC 子程序，ROM 校验和出错	继续运行
16702	MC 结束符错误	继续运行
16703	MC 状态错误	继续运行
16704	MC 速度错误	继续运行
16705	MC 类型错误	继续运行
16706	MC 轴正在运行	继续运行
16707	MC 设定速度不合法	继续运行
16740	VZ 数据 D 计算溢出	继续运行
16741	VZ 数据 T 计算溢出	继续运行
16742	VZ 数据 C 计算溢出	继续运行
16743	VZ 数据 C200 计算溢出	继续运行
16744	VZ 数据 SD 计算溢出	继续运行
16745	VZ 数据 R 计算溢出	继续运行
16746	VZ 数据 K 计算溢出	继续运行

16747	VZ 数据 H 计算溢出	继续运行
16748	VZ 数据 S 计算溢出	继续运行
16749	VZ 数据 M 计算溢出	继续运行
16750	VZ 数据 SM 计算溢出	继续运行
16751	VZ 数据 X 计算溢出	继续运行
16752	VZ 数据 Y 计算溢出	继续运行
16753	未知的 VZ 类型	继续运行
16754	32 位指令使用错误的 V	继续运行
16755	未知的字元件	继续运行
16756	未知的软元件类型	继续运行