MIKOM
Electrical Technology

MX系列可编程控制器

编程手册

V1.5

威海麦科电气技术有限公司

手册版本信息

版本	发布时间	修订说明
V1.0	2011-09-05	首次发布
V1.1	2012-01-06	修订 1.1 产品功能特点 增加 2.5.2 编程口配置 增加 3.1 编程方面的注意事项，3.2 输入输出处理，响应延迟 修订 4.1.3 输入点和输出点，4.1.6 定时器（T）以及 4.3.3 常数 对第五章中的输入触点和输出触点指令增加了 PX、PY 元件 修订第六章 应用指令中的部分指令 增加 STRPG、DHSTI、 MODRD、 MODWR、HPRC 指令。 修订 7.5 顺序功能图编程实例 修订 9.1 理解中断程序，9.4 使用外部中断 增加第十章中关于 MODRD、MODWR 指令的相关内容。 修订 10.6.1 MXLink 简介 增加第十一章 定位功能使用指南 增加第十二章 特殊功能模块使用指南
V1.2	2013-07-25	修订第六章 应用指令中的定位指令部分 增加 ENCODER 指令 增加 MX2V 指令 对第十二章 特殊功能模块使用指南增加 MX2H-4WT 模块内容
V1.3	2013-09-15	增加 BTOW、FOLLOW、EROMWR 指令
V1.4	2015-1-15	增加 FOLLOWA 指令 增加 8.1.1 高速计数器脉冲捕获功能 修改定位指令 12.7 MX2H-2HC 增加脉冲捕获功能
V1.5	2017-2-5	修订《相关手册的介绍》部分内容 修订 FOLLOWA 指令内容 MX1H 增加可以支持部分浮点数指令 修订系统错误代码表 增加以太网口配置内容 增加监控图（示波器功能）内容 增加多窗口浮动功能

通告

承蒙购买 MX 系列可编程控制器，诚表谢意。

本手册描述了 MX 系列 PLC 的编程原理、软硬件编程资源、支持的编程语言和详细的指令说明以及高速输入输出、通讯、中断、定位等技术参考内容。

在使用之前，请阅读本书以及相关产品的手册，希望在充分理解其规格的前提下正确使用产品。此外，希望本手册能够送至最终用户。

> ⚠
> 根据本书的内容，并非对工业所有权其他的权利的实施予以保证，或是承诺实施权。此外，关于因使用本书中记载的内容而引起的工业所有权方面的各种问题，本公司不予负责。

使用时的请求

- 该产品是以一般的工业为对象制作的通用产品，因此不是以用于关系到人身安全之类的情况下使用的机器或是系统为目的而设计、制造的产品。
- 考虑将该产品用于原子能、电力、宇航、医疗、乘用移动物体用的机器或是系统等特殊用途的时候，请与本公司联系进行相关讨论。
- 虽然该产品是在严格的质量管理体系下生产的，但是用于那些因该产品故障而可能导致重大故障或是产生损失的设备的时候，请在系统上设置后备和安全功能。
- 该产品和其他产品组合使用的情况下，请用户确保应该符合的规格、法规或是规则。此外，关于用户使用的系统、机械、设备中该产品的适用性和安全性，请用户自行确认。

预先通知

- 使用产品时如有疑问，请向具有电气知识（电气施工人员或是同等以上的知识）的专业电气技术人员咨询。关于该产品的操作和使用方法有疑问时，请登陆我们的信息交流网站。
- 本手册以及相关技术资料等中穿插的应用实例是作为参考用的，不保证动作。选用的时候，请用户自行对机器、设备的功能和安全性进行确认以后再使用。
- 关于本手册的内容，有时候为了改进可能会有事先不预告就更改的情况，还请见谅。
- 关于本手册的内容期望能够做到无疏漏，若读者有疑问或是发现错误，烦请联系本书封底记载的本公司或相关网站。

我们热忱欢迎读者以多种形式咨询和交流使用方法，反馈手册中的错误和遗漏。

公司网页：http://www.mikom.com.cn

信息交流：support@mikom.com.cn

相关手册的介绍

与 MX 系列 PLC 的编程原理、软硬件编程资源、支持的编程语言和详细的指令说明以及高速输入输出、通讯、中断、定位有关的详细说明，请参考本书。可编程控制器主模块和扩展模块等的产品硬件信息请参考各自的硬件随机手册。此外，需要的手册，请向当初购入产品时的供应商咨询。

▲ 必需的手册
● 根据用途需要的手册
■ 作为详细说明另外单独编制的手册

		手册名称	内容
可编程控制器主模块用手册			
➢MX1H 系列主模块			
▲	产品随附	《MX1H 系列可编程控制器硬件手册》	关于 MX1H 可编程控制器主模块的外形结构、电气规格、功能规格、安装配线和维护等部分说明，以及选配件的相关说明和常见问题答疑等详细内容。
●	用途需要	《MX1H-2AD-1DA 集成模拟量硬件手册》	关 于 MX1H-1208M1、MX1H-1208M1-CAN、MX1H-1614M1、MX1H-1614M1-CAN 主模块上的模拟量模块的输入输出特性、信号接线及特性设置。
●	用途需要	《MX1H-4AD-2DA 集成模拟量硬件手册》	关于 MX1H-0806M1、MX1H-0806M1-CAN 主模块上的模拟量模块的输入输出特性、信号接线及特性设置。
●	用途需要	《MX1H-1AD-1PT-1DA 集成模拟量硬件手册》	关于 MX1H-1208M2 主模块上的模拟量模块的输入输出特性、信号接线及特性设置。
●	用途需要	《MX1H-2PT-1DA 集成模拟量硬件手册》	关于 MX1H-1208M3 主模块上的模拟量模块的输入输出特性、信号接线及特性设置。
●	用途需要	《MX1H-6LC 集成模拟量硬件手册》	关于 MX1H-0816MTD-6LC 主模块上的模拟量模块的输入输出特性、信号接线及特性设置。
●	用途需要	《MX1H-9LC 集成模拟量硬件手册》	关于 MX1H-9LC 主模块上的模拟量模块的输入输出特性、信号接线及特性设置。
■	另外的手册	《MX 系列可编程控制器用户手册》	关于 MX 系列可编程控制器主模块的输入输出规格、电气规格、特性设置及接线、安装、运行。
➢MX2H 系列主模块			
▲	产品随附	《MX2H系列可编程控制器硬件手册》	关于 MX2H 系列可编程控制器主模块的外形结构、电气规格、功能规格、安装配线和维护等部分说明，以及选配件的相关说明和常见问题答疑等详细内容。
■	另外的手册	《MX 系列可编程控制器用户手册》	关于 MX 系列可编程控制器主模块的输入输出规格、电气规格、特性设置及接线、安装、运行。
➢MX2V 系列主模块			
▲	产品随附	《MX2V系列可编程控制器硬件手册》	关于 MX2H 系列可编程控制器主模块的外形结构、电气规格、功能规格、安装配线和维护等部分说明，以及选配件的相关说明和常见问题答疑等详细内容。
■	另外的手册	《MX 系列可编程控制器用户手册》	关于 MX 系列可编程控制器主模块的输入输出规格、电气规格、特性设置及接线、安装、运行。
➢ 编程			
■	另外的手册	《MX 系列可编程控制器编程手册》（本手册）	关于 MX 系列 PLC 的编程原理、软硬件编程资源、支持的编程语言和详细的指令说明以及高速输入输出、通讯、中断、定位有关的详细说明

		手册名称	内容
扩展模块用手册			
➢ 特殊功能模块			
通用			
■	另外的手册	《MX 系列可编程控制器用户手册》	关于 MX 系列可编程控制器主模块的输入输出规格、电气规格、特性设置及接线、安装、运行。
模拟量输入输出、温度输入、温度控制、高速计数、运动控制 使用该产品时，请同时参考《MX 系列可编程控制器编程手册》（本手册）中相关章节			
模拟量模块			
▲	产品随附	《MX2H-2AD 模拟量输入模块硬件手册》	关于 MX2H-2AD 的外形结构、电气规格、功能规格、安装配线、维护和运行检查等部分说明，以及常见问题答疑等详细内容。
▲	产品随附	《MX2H-4AD 模拟量输入模块硬件手册》	关于 MX2H-4AD 的外形结构、电气规格、功能规格、安装配线、维护和运行检查等部分说明，以及常见问题答疑等详细内容。
▲	产品随附	《MX2H-4ADH 高精度模拟量输入模块硬件手册》	关于 MX2H-4AD 的外形结构、电气规格、功能规格、安装配线、维护和运行检查等部分说明，以及常见问题答疑等详细内容。
▲	产品随附	《MX2H-2DA 模拟量输出模块硬件手册》	关于 MX2H-2DA 的外形结构、电气规格、功能规格、安装配线、维护和运行检查等部分说明，以及常见问题答疑等详细内容。
▲	产品随附	《MX2H-4DA 模拟量输出模块硬件手册》	关于 MX2H-4DA 的外形结构、电气规格、功能规格、安装配线、维护和运行检查等部分说明，以及常见问题答疑等详细内容。
▲	产品随附	《MX2H-2PT 热电阻温度输入模块硬件手册》	关于 MX2H-2PT 的外形结构、电气规格、功能规格、安装配线、维护和运行检查等部分说明，常见问题答疑等详细内容。
▲	产品随附	《MX2H-4PT 热电阻温度输入模块硬件手册》	关于 MX2H-4PT 的外形结构、电气规格、功能规格、安装配线、维护和运行检查等部分说明，常见问题答疑等详细内容。
▲	产品随附	《MX2H-2TC 热带偶温度输入模块硬件手册》	关于 MX2H-2TC 的外形结构、电气规格、功能规格、安装配线、维护和运行检查等部分说明，以及常见问题答疑等详细内容。
▲	产品随附	《MX2H-4TC 热带偶温度输入模块硬件手册》	关于 MX2H-4TC 的外形结构、电气规格、功能规格、安装配线、维护和运行检查等部分说明，以及常见问题答疑等详细内容。
▲	产品随附	《MX2H-8TC 热带偶温度输入模块硬件手册》	关于 MX2H-8TC 的外形结构、电气规格、功能规格、安装配线、维护和运行检查等部分说明，以及常见问题答疑等详细内容。
▲	产品随附	《MX2H-4WT 称重模块硬件手册》	关于 MX2H-4WT 的外形结构、电气规格、功能规格、安装配线、维护和运行检查等部分说明，以及常见问题答疑等详细内容。

		手册名称	内容
高速输入输出模块			
▲	产品随附	《MX2H-2PG 2 轴运动控制模块硬件手册》	关于 MX2H-2PG 的外形结构、电气规格、功能规格、安装配线、维护和运行检查等部分说明，以及常见问题答疑等详细内容。
▲	产品随附	《MX2H-2HC 2 通道高速计数模块硬件手册》	关于 MX2H-2HC 的外形结构、电气规格、功能规格、安装配线、维护和运行检查等部分说明，以及常见问题答疑等详细内容。
➢I/O 扩展模块			
通用			
■	另外的手册	《MX 系列可编程控制器用户手册》	关于 MX 系列可编程控制器主模块的输入输出规格、电气规格、特性设置及接线、安装、运行。
无源 I/O 扩展模块			
▲	产品随附	《MX2H 系列无源 I/O 扩展模块硬件手册》	关于 MX2H 系列无源 I/O 模块的外形结构、电气规格、功能规格、安装配线、维护和运行检查等部分说明，以及常见问题答疑等详细内容。
有源 I/O 扩展模块			
▲	产品随附	《MX2H 系列有源 I/O 扩展模块硬件手册》	关于 MX2H 系列有源 I/O 模块的外形结构、电气规格、功能规格、安装配线、维护和运行检查等部分说明，以及常见问题答疑等详细内容。
➢ 扩展卡模块			
通用			
■	另外的手册	《MX 系列可编程控制器用户手册》	关于 MX 系列可编程控制器主模块的输入输出规格、电气规格、特性设置及接线、安装、运行。
普通通讯、集成 CAN 通讯、以太网通讯 使用该产品时请同时参考《MX 系列可编程控制器编程手册》（本手册）中相关章节			
普通通讯			
▲	产品随附	《MX2H-RS485-BD 扩展卡模块硬件手册》	关于 MX2H-RS485-BD 扩展卡的外形结构、电气规格、功能规格、安装配线、维护和运行检查等部分说明。
▲	产品随附	《MX2H-RS232-BD 扩展卡模块硬件手册》	关于 MX2H-RS232-BD 扩展卡的外形结构、电气规格、功能规格、安装配线、维护和运行检查等部分说明。
集成 CAN 通讯			
▲	产品随附	《MX2H-CAN-BD 扩展卡模块硬件手册》	关于 MX2H-CAN-BD 扩展卡的外形结构、电气规格、功能规格、安装配线、维护和运行检查等部分说明。
▲	产品随附	《MX2H-CAN-RS232-BD 扩展卡模块硬件手册》	关于 MX2H-CAN-RS232-BD 扩展卡的外形结构、电气规格、功能规格、安装配线、维护和运行检查等部分说明。
以太网通讯			
▲	产品随附	《MX2H-ETH-BD 以太网扩展卡模块硬件手册》	关于 MX2H-ETH-BD 扩展卡的外形结构、电气规格、功能规格、安装、维护和运行检查及使用说明等部分说明。

前言

随着电子科技的发展及产业应用的需要，可编程逻辑控制器（Programmable Logic Controller 简称 PLC）作为一种通用的工业自动化装置，在工业控制各个领域已得到了广泛的应用。其功能也在不断的改进和发展中日益强大，例如位置控制及网络功能，扩展了数字和模拟信号以及二者之间信号的转换功能。因此 PLC 在未来的工业控制中，仍将扮演举足轻重的角色。

本手册适用于自动化技术人员，协助掌握 MIKOM 可编程控制器的编程、系统设计和调试技术，为初步及深入学习 PLC 编程知识的人员提供了参考。

手册中细致描述了 MX 系列 PLC 的编程原理、软硬件编程资源、支持的编程语言和详细的指令说明，高速输入输出、通讯、中断、定位等技术参考内容；同时介绍了各功能的使用方法并穿插了应用实例。手册的章节编排由整体到细节，章节具有独立的内容，可以从头到尾通读逐步掌握 MX 系列 PLC 的全面内容，也可以随时查阅其中章节，以便作为一些各技术参考资料。

MX 系列小型、中小型可编程控制器主要突出特点如下：

- 高稳定性、高可靠性
  宽工作电压：额定工作电压 AC100 ～ 240V，允许工作电压 AC85 ～ 264V。
  板件严格的三防处理，能够适应温度为 - 5℃～ 55℃的恶劣现场环境，且用户程序可永久保存，具有卓越的抗干扰性，使用隔离 24V 直流电源输出，可直接给传感器、HMI、外部中间继电器等提供电源。
- 高速、大容量
  基本指令处理速度现已达到 0.084us/ 步，综合指令处理速度为 1K 步 /ms，可以满足 PID 等模拟量控制调节、数据通讯、高速定位和复杂计算等要求。程序容量可达到 24K 步既可满足复杂系统及中、大型项目的设计，也可灵活应对将来的系统扩充。
- 可靠的程序安全性
  多级密码保护，所有主程序和子程序均可单独加密，程序也可绑定指定序列号的 PLC，有效保护用户的知识产权。
- 超强的系统扩展
  扩展 I/O 模块可以将系统扩展至 512 点，特殊功能模块最大可扩展 8 个，还提供多种扩展卡，增强了系统功能，节省空间。
- 实时时钟
  主模块内置实时时钟（RTC），提供万年历功能，支持闰年补偿。
- 温控 PID
  集成带自整定温控 PID 指令（TPID），支持 32 路温控及自整定，通过 TPID 指令向导使温控变得简单。
- 集成模拟量功能
  集成多路 12 位模拟量输入和输出，简化了小型应用系统设计。
- 多种通信协议
  提供了自由口、Modbus、MXLink（MIKOM 专用通讯协议）等协议，方便系统集成，可集成 CAN 接口、485 接口、以太网接口及 2 路 RS232 接口。
- 丰富的运动和定位控制功能

本手册重点介绍 MX1H、MX2H、MX2V 系列小型、中小型、运动控制型 PLC 产品。MX3H 中型系列 PLC 产品在本书中只是作简单介绍，具体资料请参见《MX3H 系列中型 PLC 编程手册》。
阅读本手册的同时也可以参考阅读《MX 系列可编程控制器用户手册》。

指令分类索引表

基本指令索引表

类别	指令	指令名称	适用机型				页码
			MX1H	MX2H	MX2V	MX3H	
基本指令	LD	常开触点指令	√	√	√	√	5-2
	LDP	常开触点上升沿指令	√	√	√	√	5-3
	LDF	常开触点下降沿指令	√	√	√	√	5-3
	LDI	常闭触点指令	√	√	√	√	5-2
	AND	常开触点与指令	√	√	√	√	5-5
	ANDP	常开触点与上升沿指令	√	√	√	√	5-3
	ANDF	常开触点与下降沿指令	√	√	√	√	5-3
	ANI	常闭触点与指令	√	√	√	√	5-5
	OR	常开触点或指令	√	√	√	√	5-6
	ORP	常开触点或上升沿指令	√	√	√	√	5-3
	ORF	常开触点或下降沿指令	√	√	√	√	5-3
	ORI	常闭触点或指令	√	√	√	√	5-6
	OUT	线圈输出指令	√	√	√	√	5-2
	SET	线圈置位指令	√	√	√	√	5-7
	RST	线圈清除指令	√	√	√	√	5-7
	ANB	并联回路块与指令	√	√	√	√	5-8
	ORB	串联回路块或指令	√	√	√	√	5-8
	INV	取反指令	√	√	√	√	5-9
	NOP	空操作指令	√	√	√	√	5-9
	MPS	进栈指令	√	√	√	√	5-10
	MRD	读栈指令	√	√	√	√	5-10
	MPP	出栈指令	√	√	√	√	5-10
	EU	上升沿检测指令	√	√	√	√	5-13
	ED	下降沿检测指令	√	√	√	√	5-13
	TON	接通延时计时指令	√	√	√	√	5-14
	TOF	断开延时计时指令	√	√	√	√	5-15
	TMON	不重触发单稳计时指令	√	√	√	√	5-16
	TONR	记忆型接通延时计时指令	√	√	√	√	5-17
	CTU	16 位计数器增计数指令	√	√	√	√	5-18
	CTR	16 位计数器循环计数指令	√	√	√	√	5-19
	DCNT	32 位计数指令	√	√	√	√	5-20

应用指令索引表

类别	指令		P	指令名称	适用机型				页码
	16 位	32 位			MX1H	MX2H	MX2V	MX3H	
比较触点指令	LD=	DLD=		整数比较 LD 等于指令	√	√	√	√	6-168
		RLD=		浮点数比较 RLD 等于指令	√	√	√	√	6-169
	LD>	DLD>		整数比较 LD 大于指令	√	√	√	√	6-168
		RLD>		浮点数比较 RLD 大于指令	√	√	√	√	6-169
	LD>=	DLD>=		整数比较 LD 大于等于指令	√	√	√	√	6-168
		RLD>=		浮点数比较 RLD 大于等于指令	√	√	√	√	6-169
	LD<	DLD<		整数比较 LD 小于指令	√	√	√	√	6-168
		RLD<		浮点数比较 RLD 小于指令	√	√	√	√	6-169
	LD<=	DLD<=		整数比较 LD 小于等于指令	√	√	√	√	6-168
		RLD<=		浮点数比较 RLD 小于等于指令	√	√	√	√	6-169
	LD<>	DLD<>		整数比较 LD 不等于指令	√	√	√	√	6-168
		RLD<>		浮点数比较 RLD 不等于指令	√	√	√	√	6-169
	AND=	DAND=		整数比较 AND 等于指令	√	√	√	√	6-170
		RAND=		浮点数比较 RAND 等于指令	√	√	√	√	6-171
	AND>	DAND>		整数比较 AND 大于指令	√	√	√	√	6-170
		RAND>		浮点数比较 RAND 大于指令	√	√	√	√	6-171
	AND>=	DAND>=		整数比较 AND 大于等于指令	√	√	√	√	6-170
		RAND>=		浮点数比较 RAND 大于等于指令	√	√	√	√	6-171
	AND<	DAND<		整数比较 AND 小于指令	√	√	√	√	6-170
		RAND<		浮点数比较 RAND 小于指令	√	√	√	√	6-171
	AND<=	DAND<=		整数比较 AND 小于等于指令	√	√	√	√	6-170
		RAND<=		浮点数比较 RAND 小于等于指令	√	√	√	√	6-171
	AND<>	DAND<>		整数比较 AND 不等于指令	√	√	√	√	6-170
		RAND<>		浮点数比较 RAND 不等于指令	√	√	√	√	6-171
	OR=	DOR=		整数比较 OR 等于指令	√	√	√	√	6-172
		ROR=		浮点数比较 ROR 等于指令	√	√	√	√	6-173
	OR>	DOR>		整数比较 OR 大于指令	√	√	√	√	6-172
		ROR>		浮点数比较 ROR 大于指令	√	√	√	√	6-173
	OR>=	DOR>=		整数比较 OR 大于等于指令	√	√	√	√	6-172
		ROR>=		浮点数比较 ROR 大于等于指令	√	√	√	√	6-173

类别	指令 16 位	指令 32 位	P	指令名称	适用机型 MX1H	MX2H	MX2V	MX3H	页码
	OR<	DOR<		整数比较 OR 小于指令	√	√	√	√	6-172
		ROR<		浮点数比较 ROR 小于指令	√	√	√	√	6-173
	OR<=	DOR<=		整数比较 OR 小于等于指令	√	√	√	√	6-172
		ROR<=		浮点数比较 ROR 小于等于指令	√	√	√	√	6-173
	OR<>	DOR<>		整数比较 OR 不等于指令	√	√	√	√	6-172
		ROR<>		浮点数比较 ROR 不等于指令	√	√	√	√	6-173
流程控制指令	LBL			跳转标号定义指令	√	√	√	√	6-5
	CJ		√	条件跳转指令	√	√	√	√	6-5
	CALL			用户子程序调用指令	√	√	√	√	6-7
	CSRET			用户子程序条件返回指令	√	√	√	√	6-8
	CFEND			任务条件返回指令	√	√	√	√	6-8
	CIRET			用户中断服务条件返回指令	√	√	√	√	6-8
	FOR			循环开始指令	√	√	√	√	6-9
	NEXT			循环返回指令	√	√	√	√	6-9
	MC			主控指令	√	√	√	√	6-11
	MCR			主控清除指令	√	√	√	√	6-11
	WDT			用户程序看门狗清零指令	√	√	√	√	6-13
	STOP			用户程序停止指令	√	√	√	√	6-14
	EI			中断使能指令	√	√	√	√	6-14
	DI			中断禁止指令	√	√	√	√	6-14
SFC 指令	STL			SFC 状态装载指令	√	√	√	√	6-15
	RET			SFC 程序结束指令	√	√	√	√	6-15
传输比较指令	CMP	DCMP	√	整数比较指令	√	√	√	√	6-16
		RCMP	√	浮点数比较指令		√	√	√	6-17
	MOV	DMOV	√	字数据传输指令	√	√	√	√	6-18
		RMOV	√	浮点数数据传输指令		√	√	√	6-19
	BMOV		√	块数据传输指令	√	√	√	√	6-20
	SWAP		√	高低字节交换指令	√	√	√	√	6-21
	XCH	DXCH	√	字交换指令	√	√	√	√	6-22
	FMOV	DFMOV	√	数据块填充指令	√	√	√	√	6-23
	PUSH		√	数据入栈指令	√	√	√	√	6-24
	FIFO		√	先入先出指令	√	√	√	√	6-25

类别	指令		P	指令名称	适用机型				页码
	16 位	32 位			MX1H	MX2H	MX2V	MX3H	
	LIFO		√	后入先出指令	√	√	√	√	6-27
四则运算指令	ADD	DADD	√	整数加法指令	√	√	√	√	6-28
		RADD	√	浮点数加法指令	√	√	√	√	6-29
	SUB	DSUB	√	整数减法指令	√	√	√	√	6-30
		RSUB	√	浮点数减法指令	√	√	√	√	6-31
	INC	DINC	√	增一指令	√	√	√	√	6-32
	DEC	DDEC	√	减一指令	√	√	√	√	6-32
	MUL	DMUL	√	整数乘法指令	√	√	√	√	6-33
		RMUL	√	浮点数乘法指令	√	√	√	√	6-34
	DIV	DDIV	√	整数除法指令	√	√	√	√	6-35
		RDIV	√	浮点数除法指令	√	√	√	√	6-36
	VABS	DVABS	√	整数绝对值指令	√	√	√	√	6-37
		RVABS	√	浮点数绝对值指令	√	√	√	√	6-37
	NEG	DNEG	√	整数取负指令	√	√	√	√	6-38
		RNEG	√	浮点数取负指令	√	√	√	√	6-38
	SQT	DSQT	√	整数平方根指令	√	√	√	√	6-39
		RSQT	√	浮点数平方根指令	√	√	√	√	6-39
	SUM	DSUM	√	整数累加指令	√	√	√	√	6-40
		RSUM	√	浮点数累加指令	√	√	√	√	6-41
		SIN	√	浮点数 SIN 运算指令	√	√	√	√	6-42
		ASIN	√	浮点数 ASIN 运算指令	√	√	√	√	6-43
		COS	√	浮点数 COS 运算指令	√	√	√	√	6-44
		ACOS	√	浮点数 ACOS 运算指令	√	√	√	√	6-45
		TAN	√	浮点数 TAN 运算指令	√	√	√	√	6-46
		ATAN	√	浮点数 ATAN 运算指令	√	√	√	√	6-47
		RAD	√	浮点数角度→弧度转换指令	√	√	√	√	6-48
		DEG	√	浮点数弧度→角度转换指令	√	√	√	√	6-49
		LOG	√	浮点数常用对数指令	√	√	√	√	6-50
		LN	√	浮点数自然对数指令	√	√	√	√	6-51
		EXP	√	浮点数自然数幂指令	√	√	√	√	6-52
		POWER	√	浮点数求幂指令	√	√	√	√	6-53
	WAND	DWAND	√	字与指令	√	√	√	√	6-54

类别	指令 16位	指令 32位	P	指令名称	适用机型 MX1H	MX2H	MX2V	MX3H	页码
	WOR	DWOR	√	字或指令	√	√	√	√	6-55
	WXOR	DWXOR	√	字异或指令	√	√	√	√	6-56
	WINV	DWINV	√	字取非指令	√	√	√	√	6-57
移位旋转指令	ROR	DROR	√	位循环右移指令	√	√	√	√	6-58
	ROL	DROL	√	位循环左移指令	√	√	√	√	6-59
	RCR	DRCR	√	位带进位循环右移指令	√	√	√	√	6-60
	RCL	DRCL	√	位带进位循环左移指令	√	√	√	√	6-61
	SHR	DSHR	√	位右移指令	√	√	√	√	6-62
	SHL	DSHL	√	位左移指令	√	√	√	√	6-63
	SFTR		√	位串右移指令	√	√	√	√	6-64
	SFTL		√	位串左移指令	√	√	√	√	6-65
	WSFR		√	字串右移指令	√	√	√	√	6-66
	WSFL		√	字串左移指令	√	√	√	√	6-67
位处理指令	DECO		√	解码指令	√	√	√	√	6-68
	ENCO		√	编码指令	√	√	√	√	6-69
	BITS	DBITS	√	字中 ON 位统计指令		√	√	√	6-70
	BON		√	ON 位判断指令		√	√	√	6-71
	ZSET		√	批量位置位指令	√	√	√	√	6-72
	ZRST		√	批量位清零指令	√	√	√	√	6-73
	MEAN		√	平均值指令		√	√	√	6-74
	WTOB		√	字节单位的数据分离指令	√	√	√	√	6-75
	BTOW		√	字节单位的数据组合指令	√	√	√	√	6-76
	UNI		√	16 位数据的 4 位结合指令	√	√	√	√	6-77
	DIS		√	16 位数据的 4 位分离指令	√	√	√	√	6-78
	ANS		√	信号报警器置位指令		√	√	√	6-79
	ANR		√	信号报警器复位指令		√	√	√	6-80
高速I/O指令		HCNT		高速计数器驱动指令	√	√	√	√	6-81
		DHSCS		高速计数比较置位指令	√	√	√	√	6-82
		DHSCR		高速计数比较复位指令	√	√	√	√	6-83
		DHSCI		高速计数比较中断触发指令	√	√	√	√	6-84
		DHSZ		高速计数区间比较指令	√	√	√	√	6-85
		DHST		高速计数表格比较指令	√	√	√	√	6-86

类别	指令		P	指令名称	适用机型				页码
	16 位	32 位			MX1H	MX2H	MX2V	MX3H	
高速 I/O 指令		DHSTI		高速计数表格比较中断触发指令	√	√	√	√	6-88
		DHSP		高速计数表格比较脉冲输出指令	√	√	√	√	6-90
	SPD			测频指令	√	√	√	√	6-92
		PLSY		计数脉冲输出指令	√	√	√	√	6-94
		PLSR		带加减速的计数脉冲输出指令	√	√	√	√	6-96
		PLS		高速包络线输出指令	√	√	√	√	6-98
	PWM			PWM 脉冲输出指令	√	√	√	√	6-101
	MTR			矩阵输入指令	√		√	√	6-103
		ENCODER		编码器初始化指令		√	√	√	6-105
定位指令		DABS		当前值读取指令	√	√	√	√	6-106
		DZRN		原点回归指令	√	√	√	√	6-109
		PLSV		可变速脉冲输出指令	√	√	√	√	6-112
		DDRVI		相对位置控制指令	√	√	√	√	6-115
		DDRVA		绝对位置控制指令	√	√	√	√	6-118
		DSZR		带 DOG 原点回归指令	√	√	√	√	6-121
		DDVIT		中断定位指令	√	√	√	√	6-126
		HPRC		手动脉冲控制指令	√	√	√	√	6-130
		FOLLOW		跟随指令	√	√	√	√	6-132
		FOLLOWA		可设定比例补偿脉冲及输出脉冲个数的跟随指令		√	√	√	6-134
		LIN		直线插补指令			√	√	6-138
		CW		顺时针圆弧插补指令			√	√	6-141
		CCW		逆时针圆弧插补指令			√	√	6-144
外设指令	FROM		√	特殊模块缓冲寄存器字读指令		√	√	√	6-147
	TO		√	特殊模块缓冲寄存器字写指令		√	√	√	6-148
	REFF		√	设置输入滤波常数指令	√	√	√	√	6-149
	REF		√	刷新 I/O 镜像指令	√	√	√	√	6-150
	PR			ASCII 打印指令			√	√	6-151
	TKY	DTKY		16 位数字键输入指令			√	√	6-153
	HKY	DHKY		16 进制输入指令			√	√	6-155
	DSW			数字开关指令			√	√	6-157
	EROMWR		√	EEPROM 写指令		√	√	√	6-159
实时时钟指令	TRD			实时时钟读指令	√	√	√	√	6-160
	TWR			实时时钟写指令	√	√	√	√	6-161
	TADD			时间加指令	√	√	√	√	6-162

类别	指令 16 位	指令 32 位	P	指令名称	适用机型 MX1H	MX2H	MX2V	MX3H	页码
实时时钟指令	TSUB			时间减指令	√	√	√	√	6-163
	HOUR			计时表指令	√	√	√	√	6-164
	TCMP		√	时间比较指令	√	√	√	√	6-165
	HTOS	DHTOS	√	[时、分、秒] 数据的秒转换指令	√	√	√	√	6-166
	STOH	DSTOH	√	秒数据的 [时、分、秒] 转换指令	√	√	√	√	6-167
数值转换指令	FLT	DFLT	√	整数转换浮点数指令	√	√	√	√	6-174
	INT	DINT	√	浮点数转换整数指令	√	√	√	√	6-176
	BCD	DBCD	√	字转换为 BCD 码指令	√	√	√	√	6-177
	BIN	DBIN	√	BCD 码转换为字指令	√	√	√	√	6-179
	GRY	DGRY	√	字转换为格雷码指令	√	√	√	√	6-181
	GBIN	DGBIN	√	格雷码转换为字指令	√	√	√	√	6-182
	SEG		√	字转换 7 段码指令	√	√	√	√	6-183
	ASC		√	ASCII 码转换指令	√	√	√	√	6-184
	ITA		√	16 进制数转换 ASCII 码指令	√	√	√	√	6-185
	ATI		√	ASCII 码转换 16 进制数指令	√	√	√	√	6-187
	LCNV			工程转换指令	√	√	√	√	6-189
		RLCNV		浮点数工程转换指令		√	√	√	6-191
通讯指令	MODBUS			MODBUS 主站通讯指令	√	√	√	√	6-193
	MODRD			MODBUS 读指令	√	√	√	√	6-197
	MODWR			MODBUS 写指令	√	√	√	√	6-199
	XMT			自由口发送指令	√	√	√	√	6-201
	RCV			自由口接收指令	√	√	√	√	6-202
	CANXMT			CANbus 单帧发送指令	√	√	√	√	6-203
	CANRCV			CANbus 单帧接收指令	√	√	√	√	6-205
便捷指令	SORT			数据排序指令		√	√	√	6-206
	RND		√	产生随机数指令	√	√	√	√	6-208
	DUTY		√	产生定时脉冲指令	√	√	√	√	6-209
	CCITT		√	CCITT 校验指令	√	√	√	√	6-210
	CRC16		√	CRC16 校验指令	√	√	√	√	6-211
	LRC		√	LRC 校验指令	√	√	√	√	6-212
	PID			PID 功能指令	√	√	√	√	6-213
	TPID			温控 PID 功能指令	√	√	√	√	6-219

类别	指令		P	指令名称	适用机型				页码
	16 位	32 位			MX1H	MX2H	MX2V	MX3H	
便捷指令	RAMP			斜坡信号输出指令	√	√	√	√	6-224
	TRIANGLE			三角波信号输出指令	√	√	√	√	6-225
	HACKLE			锯齿波信号输出指令	√	√	√	√	6-226
	ABSD	DABSD		凸轮控制指令	√	√	√	√	6-227
	ALT		√	交替输出指令	√	√	√	√	6-228
	INCD			凸轮控制相对方式指令			√	√	6-230
	TTMR			示教定时器指令			√	√	6-232
	STMR			特殊定时器指令			√	√	6-233
数据块表处理指令	BKADD	DBKADD	√	数据块加法指令			√	√	6-235
	BKSUB	DBKSUB	√	数据块减法指令			√	√	6-237
	BKCMP=	DBKCMP=	√	数据块比较等于指令			√	√	6-239
	BKCMP>	DBKCMP>	√	数据块比较等于指令			√	√	6-239
	BKCMP>=	DBKCMP>=	√	数据块比较大于等于指令			√	√	6-239
	BKCMP<	DBKCMP<	√	数据块比较小于指令			√	√	6-239
	BKCMP<=	DBKCMP<=	√	数据块比较小于等于指令			√	√	6-239
	BKCMP<>	DBKCMP<>	√	数据块比较不等于指令			√	√	6-239
	LIMIT	DLIMIT		上下限限位控制指令			√	√	6-239
	BAND	DBAND		死区控制指令			√	√	6-243
	ZONE	DZONE		区域控制指令			√	√	6-245
	SCL	DSCL		定坐标（不同点坐标数据）指令			√	√	6-247
	SER	DSER		数据检索指令			√	√	6-250
字符串指令	STRADD		√	字符串的结合指令			√	√	6-252
	STRLEN		√	检测出字符串的长度指令			√	√	6-253
	STRRIGHT		√	从字符串的右侧开始取出指令			√	√	6-254
	STRLEFT		√	从字符串的左侧开始取出指令			√	√	6-255
	STRMIDR		√	从字符串中的任意取出指令			√	√	6-256
	STRMIDW		√	从字符串中任意替换指令			√	√	6-258
	STRINSTR		√	字符串的检索指令			√	√	6-260
	STRMOV		√	字符串的传送指令			√	√	6-261

指令排序索引表

基本指令索引表

类别	指令	指令名称	适用机型				页码
			MX1H	MX2H	MX2V	MX3H	
A	ANB	并联回路块与指令	√	√	√	√	5-8
	AND	常开触点与指令	√	√	√	√	5-5
	ANDF	常开触点与下降沿指令	√	√	√	√	5-3
	ANDP	常开触点与上升沿指令	√	√	√	√	5-3
	ANI	常闭触点与指令	√	√	√	√	5-5
C	CTR	16 位计数器循环计数指令	√	√	√	√	5-19
	CTU	16 位计数器增计数指令	√	√	√	√	5-18
D	DCNT	32 位计数指令	√	√	√	√	5-20
E	ED	下降沿检测指令	√	√	√	√	5-13
	EU	上升沿检测指令	√	√	√	√	5-13
I	INV	取反指令	√	√	√	√	5-9
L	LD	常开触点指令	√	√	√	√	5-2
	LDF	常开触点下降沿指令	√	√	√	√	5-3
	LDI	常闭触点指令	√	√	√	√	5-2
	LDP	常开触点上升沿指令	√	√	√	√	5-3
M	MPP	出栈指令	√	√	√	√	5-10
	MPS	进栈指令	√	√	√	√	5-10
	MRD	读栈指令	√	√	√	√	5-10
N	NOP	空操作指令	√	√	√	√	5-9
O	OR	常开触点或指令	√	√	√	√	5-6
	ORB	串联回路块或指令	√	√	√	√	5-8
	ORF	常开触点或下降沿指令	√	√	√	√	5-3
	ORI	常闭触点或指令	√	√	√	√	5-6
	ORP	常开触点或上升沿指令	√	√	√	√	5-3
	OUT	线圈输出指令	√	√	√	√	5-2
R	RST	线圈清除指令	√	√	√	√	5-7
S	SET	线圈置位指令	√	√	√	√	5-7
T	TMON	不重触发单稳计时指令	√	√	√	√	5-16
	TOF	断开延时计时指令	√	√	√	√	5-15
	TON	接通延时计时指令	√	√	√	√	5-14
	TONR	记忆型接通延时计时指令	√	√	√	√	5-17

应用指令索引表

类别	指令		P	指令名称	适用机型				页码
	16 位	32 位			MX1H	MX2H	MX2V	MX3H	
A	ABSD	DABSD		凸轮控制指令	√	√	√	√	6-227
		ACOS	√	浮点数 ACOS 运算指令	√	√	√	√	6-45
	ADD	DADD	√	整数加法指令	√	√	√	√	6-28
	ALT		√	交替输出指令	√	√	√	√	6-228
	AND=	DAND=		整数比较 AND 等于指令	√	√	√	√	6-170
	AND>	DAND>		整数比较 AND 大于指令	√	√	√	√	6-170
	AND>=	DAND>=		整数比较 AND 大于等于指令	√	√	√	√	6-170
	AND<	DAND<		整数比较 AND 小于指令	√	√	√	√	6-170
	AND<=	DAND<=		整数比较 AND 小于等于指令	√	√	√	√	6-170
	AND<>	DAND<>		整数比较 AND 不等于指令	√	√	√	√	6-170
	ANR		√	信号报警器复位指令		√	√	√	6-80
	ANS		√	信号报警器置位指令		√	√	√	6-79
	ASC		√	ASCII 码转换指令	√	√	√	√	6-184
		ASIN	√	浮点数 ASIN 运算指令	√	√	√	√	6-43
		ATAN	√	浮点数 ATAN 运算指令	√	√	√	√	6-47
	ATI		√	ASCII 码转换 16 进制数指令	√	√	√	√	6-187
B	BAND	DBAND		死区控制指令			√	√	6-243
	BCD	DBCD	√	字转换为 BCD 码指令	√	√	√	√	6-177
	BIN	DBIN	√	BCD 码转换为字指令	√	√	√	√	6-179
	BITS	DBITS	√	字中 ON 位统计指令		√	√	√	6-70
	BKADD	DBKADD	√	数据块加法指令			√	√	6-235
	BKCMP=	DBKCMP=	√	数据块比较等于指令			√	√	6-239
	BKCMP>	DBKCMP>	√	数据块比较等于指令			√	√	6-239
	BKCMP>=	DBKCMP>=	√	数据块比较大于等于指令			√	√	6-239
	BKCMP<	DBKCMP<	√	数据块比较小于指令			√	√	6-239
	BKCMP<=	DBKCMP<=	√	数据块比较小于等于指令			√	√	6-239
	BKCMP<>	DBKCMP<>	√	数据块比较不等于指令			√	√	6-239
	BKSUB	DBKSUB	√	数据块减法指令			√	√	6-237
	BMOV		√	块数据传输指令	√	√	√	√	6-20
	BON		√	ON 位判断指令		√	√	√	6-71
	BTOW		√	字节单位的数据组合指令	√	√	√	√	6-76

类别	指令		P	指令名称	适用机型				页码
	16 位	32 位			MX1H	MX2H	MX2V	MX3H	
C	CALL			用户子程序调用指令	√	√	√	√	6-7
	CANRCV			CANbus 单帧接收指令	√	√	√	√	6-205
	CANXMT			CANbus 单帧发送指令	√	√	√	√	6-203
	CCITT		√	CCITT 校验指令	√	√	√	√	6-210
		CCW		逆时针圆弧插补指令			√	√	6-144
	CFEND			任务条件返回指令	√	√	√	√	6-8
	CIRET			用户中断服务条件返回指令	√	√	√	√	6-8
	CJ		√	条件跳转指令	√	√	√	√	6-5
	CMP	DCMP	√	整数比较指令	√	√	√	√	6-16
		COS	√	浮点数 COS 运算指令		√	√	√	6-44
		CW		顺时针圆弧插补指令			√	√	6-141
	CRC16		√	CRC16 校验指令	√	√	√	√	6-211
	CSRET			用户子程序条件返回指令	√	√	√	√	6-8
D		DABS		当前值读取指令		√	√	√	6-106
		DDRVA		绝对位置控制指令		√	√	√	6-118
		DDRVI		相对位置控制指令		√	√	√	6-115
		DDVIT		中断定位指令		√	√	√	6-126
	DEC	DDEC	√	减一指令	√	√	√	√	6-32
	DECO		√	解码指令	√	√	√	√	6-68
		DEG	√	浮点数弧度→角度转换指令	√	√	√	√	6-49
		DHSCI		高速计数比较中断触发指令	√	√	√	√	6-84
		DHSCR		高速计数比较复位指令	√	√	√	√	6-83
		DHSCS		高速计数比较置位指令	√	√	√	√	6-82
		DHSP		高速计数表格比较脉冲输出指令	√	√	√	√	6-90
		DHST		高速计数表格比较指令	√	√	√	√	6-86
		DHSTI		高速计数表格比较中断触发指令	√	√	√	√	6-88
		DHSZ		高速计数区间比较指令	√	√	√	√	6-85
	DI			中断禁止指令	√	√	√	√	6-14
	DIS		√	16 位数据的 4 位分离指令	√	√	√	√	6-78
	DIV	DDIV	√	整数除法指令	√	√	√	√	6-35
		DSZR		带 DOG 原点回归指令		√	√	√	6-121
	DSW			数字开关指令			√	√	6-157

类别	指令		P	指令名称	适用机型				页码
	16 位	32 位			MX1H	MX2H	MX2V	MX3H	
D	DUTY		√	产生定时脉冲指令	√	√	√	√	6-209
		DZRN		原点回归指令		√	√	√	6-109
E	EI			中断使能指令	√	√	√	√	6-14
	ENCO		√	编码指令	√	√	√	√	6-69
	ENCODER		√	编码器初始化指令		√	√	√	6-105
	EROMWR		√	EEPROM 写指令		√	√	√	6-159
		EXP	√	浮点数自然数幂指令	√	√	√	√	6-52
F	FIFO		√	先入先出指令	√	√	√	√	6-25
	FLT	DFLT	√	整数转换浮点数指令	√	√	√	√	6-174
	FMOV	DFMOV	√	数据块填充指令	√	√	√	√	6-23
		FOLLOW		跟随指令	√	√	√	√	6-132
		FOLLOWA		可设定比例补偿脉冲及输出脉冲个数的跟随指令		√	√	√	6-134
	FOR			循环开始指令	√	√	√	√	6-9
	FROM		√	特殊模块缓冲寄存器字读指令		√	√	√	6-147
G	GBIN	DGBIN	√	格雷码转换为字指令	√	√	√	√	6-182
	GRY	DGRY	√	字转换为格雷码指令	√	√	√	√	6-181
H	HPRC			手动脉冲控制指令	√	√	√	√	6-130
	HACKLE			锯齿波信号输出指令	√	√	√	√	6-226
		HCNT		高速计数器驱动指令	√	√	√	√	6-81
	HKY	DHKY		16 进制输入指令			√	√	6-155
	HOUR			计时表指令	√	√	√	√	6-164
	HTOS	DHTOS	√	[时、分、秒] 数据的秒转换指令	√	√	√	√	6-166
I	INC	DINC	√	增一指令	√	√	√	√	6-32
	INCD			凸轮控制相对方式指令			√	√	6-230
	INT	DINT	√	浮点数转换整数指令	√	√	√	√	6-176
	ITA		√	16 进制数转换 ASCII 码指令	√	√	√	√	6-185
L	LBL			跳转标号定义指令	√	√	√	√	6-5
	LCNV			工程转换指令	√	√	√	√	6-189
	LD=	DLD=		整数比较 LD 等于指令	√	√	√	√	6-168
	LD>	DLD>		整数比较 LD 大于指令	√	√	√	√	6-168
	LD>=	DLD>=		整数比较 LD 大于等于指令	√	√	√	√	6-168
	LD<	DLD<		整数比较 LD 小于指令	√	√	√	√	6-168
	LD<=	DLD<=		整数比较 LD 小于等于指令	√	√	√	√	6-168
	LD<>	DLD<>		整数比较 LD 不等于指令	√	√	√	√	6-168
	LIFO		√	后入先出指令	√	√	√	√	6-27

类别	指令		P	指令名称	适用机型				页码
	16 位	32 位			MX1H	MX2H	MX2V	MX3H	
L		LIN		直线插补指令			√	√	6-138
	LIMIT	DLIMIT		上下限限位控制指令			√	√	6-241
		LN	√	浮点数自然对数指令	√	√	√	√	6-51
		LOG	√	浮点数常用对数指令	√	√	√	√	6-50
	LRC		√	LRC 校验指令	√	√	√	√	6-212
M	MC			主控指令	√	√	√	√	6-11
	MCR			主控结束指令	√	√	√	√	6-11
	MEAN		√	平均值指令		√	√	√	6-74
	MODBUS			MODBUS 主站通讯指令	√	√	√	√	6-193
	MODRD			MODBUS 读指令	√	√	√	√	6-197
	MODWR			MODBUS 写指令	√	√	√	√	6-199
	MOV	DMOV	√	字数据传输指令	√	√	√	√	6-18
	MTR			矩阵输入指令		√	√	√	6-103
	MUL	DMUL	√	整数乘法指令	√	√	√	√	6-33
N	NEG	DNEG	√	整数取负指令	√	√	√	√	6-38
	NEXT			循环返回指令	√	√	√	√	6-9
O	OR=	DOR=		整数比较 OR 等于指令	√	√	√	√	6-172
	OR>	DOR>		整数比较 OR 大于指令	√	√	√	√	6-172
	OR>=	DOR>=		整数比较 OR 大于等于指令	√	√	√	√	6-172
	OR<	DOR<		整数比较 OR 小于指令	√	√	√	√	6-172
	OR<=	DOR<=		整数比较 OR 小于等于指令	√	√	√	√	6-172
	OR<>	DOR<>		整数比较 OR 不等于指令	√	√	√	√	6-172
P	PID			PID 功能指令	√	√	√	√	6-213
		PLS		高速包络线输出指令	√	√	√	√	6-98
		PLSR		带加减速的计数脉冲输出指令	√	√	√	√	6-96
		PLSV		可变速脉冲输出指令		√	√	√	6-112
		PLSY		计数脉冲输出指令	√	√	√	√	6-94
		POWER	√	浮点数求幂指令	√	√	√	√	6-53
	PR			ASCII 打印指令			√	√	6-151
	PUSH		√	数据入栈指令	√	√	√	√	6-24
	PWM			PWM 脉冲输出指令	√	√	√	√	6-101
R		RAD	√	浮点数角度→弧度转换指令	√	√	√	√	6-48

类别	指令		P	指令名称	适用机型				页码
	16 位	32 位			MX1H	MX2H	MX2V	MX3H	
R		RADD	√	浮点数加法指令	√	√	√	√	6-29
	RAMP			斜坡信号输出指令	√	√	√	√	6-224
		RAND=		浮点数比较 RAND 等于指令	√	√	√	√	6-171
		RAND>		浮点数比较 RAND 大于指令	√	√	√	√	6-171
		RAND>=		浮点数比较 RAND 大于等于指令	√	√	√	√	6-171
		RAND<		浮点数比较 RAND 小于指令	√	√	√	√	6-171
		RAND<=		浮点数比较 RAND 小于等于指令	√	√	√	√	6-171
		RAND<>		浮点数比较 RAND 不等于指令	√	√	√	√	6-171
	RCL	DRCL	√	位带进位循环左移指令	√	√	√	√	6-61
		RCMP	√	浮点数比较指令		√	√	√	6-17
	RCR	DRCR	√	位带进位循环右移指令	√	√	√	√	6-60
	RCV			自由口接收指令	√	√	√	√	6-202
		RDIV	√	浮点数除法指令	√	√	√	√	6-36
	REF		√	刷新 I/O 镜像指令	√	√	√	√	6-150
	REFF		√	设置输入滤波常数指令	√	√	√	√	6-149
	RET			SFC 程序结束指令	√	√	√	√	6-15
	RLCNV			浮点数工程转换指令		√	√	√	6-191
		RLD=		浮点数比较 RLD 等于指令	√	√	√	√	6-169
		RLD>		浮点数比较 RLD 大于指令	√	√	√	√	6-169
		RLD>=		浮点数比较 RLD 大于等于指令	√	√	√	√	6-169
		RLD<		浮点数比较 RLD 小于指令	√	√	√	√	6-169
		RLD<=		浮点数比较 RLD 小于等于指令	√	√	√	√	6-169
		RLD<>		浮点数比较 RLD 不等于指令	√	√	√	√	6-169
		RMOV	√	浮点数数据传输指令		√	√	√	6-19
		RMUL	√	浮点数乘法指令	√	√	√	√	6-34
	RND		√	产生随机数指令	√	√	√	√	6-209
		RNEG	√	浮点数取负指令	√	√	√	√	6-38
	ROL	DROL	√	位循环左移指令	√	√	√	√	6-59
	ROR	DROR	√	位循环右移指令	√	√	√	√	6-58
		ROR=		浮点数比较 ROR 等于指令	√	√	√	√	6-173
		ROR>		浮点数比较 ROR 大于指令	√	√	√	√	6-173
		ROR>=		浮点数比较 ROR 大于等于指令	√	√	√	√	6-173

类别	指令		P	指令名称	适用机型				页码
	16 位	32 位			MX1H	MX2H	MX2V	MX3H	
R		ROR<		浮点数比较 ROR 小于指令	√	√	√	√	6-173
		ROR<=		浮点数比较 ROR 小于等于指令	√	√	√	√	6-173
		ROR<>		浮点数比较 ROR 不等于指令	√	√	√	√	6-173
		RSQT	√	浮点数平方根指令	√	√	√	√	6-39
		RSUB	√	浮点数减法指令	√	√	√	√	6-31
		RSUM	√	浮点数累加指令	√	√	√	√	6-41
		RVABS	√	浮点数绝对值指令	√	√	√	√	6-37
S	SCL	DSCL		定坐标（不同点坐标数据）指令			√	√	6-247
	SER	DSER		数据检索指令			√	√	6-250
	SEG		√	字转换 7 段码指令	√	√	√	√	6-183
	SFTL		√	位串左移指令	√	√	√	√	6-65
	SFTR		√	位串右移指令	√	√	√	√	6-64
	SHL	DSHL	√	位左移指令	√	√	√	√	6-63
	SHR	DSHR	√	位右移指令	√	√	√	√	6-62
		SIN	√	浮点数 SIN 运算指令	√	√	√	√	6-42
	SORT			数据排序指令		√	√	√	6-206
	SPD			测频指令	√	√	√	√	6-92
	SQT	DSQT	√	整数平方根指令	√	√	√	√	6-39
	STL			SFC 状态装载指令	√	√	√	√	6-15
	STMR			特殊定时器指令			√	√	6-233
	STOH	DSTOH	√	秒数据的 [时、分、秒] 转换指令	√	√	√	√	6-167
	STOP			用户程序停止指令	√	√	√	√	6-14
	STRADD		√	字符串的结合指令			√	√	6-252
	STRLEFT		√	从字符串的左侧开始取出指令			√	√	6-255
	STRLEN		√	检测出字符串的长度指令			√	√	6-253
	STRMIDR		√	从字符串中的任意取出指令			√	√	6-256
	STRMIDW		√	从字符串中任意替换指令			√	√	6-258
	STRMOV		√	字符串的传送指令			√	√	6-261
	STRRIGHT		√	从字符串的右侧开始取出指令			√	√	6-254
	STRINSTR		√	字符串的检索指令			√	√	6-260
	SUB	DSUB	√	整数减法指令	√	√	√	√	6-30
	SUM	DSUM	√	整数累加指令	√	√	√	√	6-40

类别	指令 16位	指令 32位	P	指令名称	MX1H	MX2H	MX2V	MX3H	页码
S	SWAP		√	高低字节交换指令	√	√	√	√	6-21
T	TADD			时间加指令	√	√	√	√	6-162
		TAN	√	浮点数 TAN 运算指令	√	√	√	√	6-46
	TCMP		√	时间比较指令	√	√	√	√	6-165
	TKY	DTKY		16 位数字键输入指令			√	√	6-153
	TO		√	特殊模块缓冲寄存器字写指令		√	√	√	6-148
	TPID			温控 PID 功能指令	√	√	√	√	6-219
	TRD			实时时钟读指令	√	√	√	√	6-160
	TRIANGLE			三角波信号输出指令	√	√	√	√	6-225
	TSUB			时间减指令	√	√	√	√	6-163
	TTMR			示教定时器指令			√	√	6-232
	TWR			实时时钟写指令	√	√	√	√	6-161
U	UNI		√	16 位数据的 4 位结合指令	√	√	√	√	6-77
V	VABS	DVABS	√	整数绝对值指令	√	√	√	√	6-37
W	WAND	DWAND	√	字与指令	√	√	√	√	6-54
	WDT			用户程序看门狗清零指令	√	√	√	√	6-13
	WINV	DWINV	√	字取非指令	√	√	√	√	6-57
	WOR	DWOR	√	字或指令	√	√	√	√	6-55
	WSFL		√	字串左移指令	√	√	√	√	6-67
	WSFR		√	字串右移指令	√	√	√	√	6-66
	WTOB		√	字节单位的数据分离指令	√	√	√	√	6-75
	WXOR	DWXOR	√	字异或指令	√	√	√	√	6-56
X	XCH	DXCH	√	字交换指令	√	√	√	√	6-22
	XMT			自由口发送指令	√	√	√	√	6-201
Z	ZONE	DZONE		区域控制指令			√	√	6-245
	ZRST		√	批量位清零指令	√	√	√	√	6-73
	ZSET		√	批量位置位指令	√	√	√	√	6-72

目录

第一章　产品简介

本章主要介绍 MX1H、MX2H、MX2V 系列 PLC 产品性能规格、外形及 MXProgrammer 编程软件。

1.1 产品功能特点

MX 系列 PLC 产品包括 MX0H 微型系列、MX1H 小型系列和 MX2H 中小型系列、MX2V 运动控制型系列以及 MX3H 中型系列，是适应于各种现代工业控制应用的高性能产品。

MX0H 系列 PLC

MX0H 系列 PLC 是一种微型 PLC，不带高速输入高速输出以及 RS485 接口，最大点数为 30 点，内部资源相对减少，一般应用在纯逻辑控制、成本要求较低的场合。具体资料请详见《MX0H 系列微型 PLC 手册》。

MX1H 系列 PLC

- MX1H 系列 PLC 为一体式结构。
- 自带 1 路 RS232 接口，可配置 CAN、RS485 通讯接口，内置自由口、Modbus（RTU）、MXLink（MIKOM 专用通讯协议）等协议。
- 同时可集成 AD、DA、TC 等多种模拟量通道。其中集成的温度控制通道可使用带有自整定功能的温度控制 TPID 指令，可以方便构建温控系统，降低成本，简化系统设计。
- 自带 1 路大容量隔离的 24V 电源输出。
- 高速计数器支持单相单计数、单相双计数、双相双计数三种方式，支持四倍频计数。高速计数频率总和达 80KHz，单相高速计数支持 50KHz×2 通道 +10KHz×4 通道，双相高速计数支持 20KHz×1 通道 +10KHz×2 通道。
- 主模块提供 2 路独立的高速脉冲输出（最高 100KHz），支持脉冲序列输出（PTO）和脉宽调制（PWM）两种模式。提供了包络线脉冲输出功能，用于控制伺服或步进电机的多步加速、匀速和减速过程，最多可达 256 步，适用于速度改变频繁的场合。
- 基本指令处理速度为 0.084us/ 步，综合指令处理速度为 1K 步 /ms，程序容量为 12K 步，一般应用在低成本、结构紧凑的小型控制系统中。

MX2H 系列 PLC

- MX2H 系列 PLC 可分为一体式结构和模块式组合结构。
- MX2H 主模块自带 1 路 RS232 和 1 路 RS485 接口，可通过配置 MX2H-CAN-BD（隔离 CAN 卡）、MX2H-RS232-BD（隔离 RS232 卡）、MX2H-CAN-RS232-BD（不隔离的 CAN 和 RS232 卡）、MX2H-RS485-BD（隔离 RS485 卡）、MX2H-ETH-BD（以太网卡）等通讯卡增强通讯功能，内置自由口、Modbus（RTU）、MXLink（MIKOM 专用通讯协议）等协议，可以构建较为完善的通讯网络。
- I/O 扩展模块包括 MX2H-0008ER(T)N、MX2H-0808ER(T)N、MX2H-0800ENN、MX2H-1600ENN、MX2H-0016ER(T)N 等，可以通过带交流电源输入的 MX2H-1616ER(T)A 扩展 I/O 模块将系统扩展至 512 点。
- 特殊功能模块包括 MX2H-4AD、MX2H-4DA、MX2H-4TC、MX2H-4PT、MX2H-8TC、MX2H-2HC、MX2H-2PG 等，其中温度控制模块可使用带有自整定功能的温度控制 TPID 指令，可以方便构建温控系统。
- 一体式结构的 PLC 可带 1 路大容量隔离的 24V 电源输出。
- 高速计数器支持单相单计数、单相双计数、双相双计数三种方式，支持四倍频计数。高速计数频率总和达 100KHz，单相高速计数支持 50KHz×2 通道 +10KHz×4 通道，双相高速计数支持 25KHz×1 通道 +5KHz×2 通道。
- 主模块提供 3 路独立的高速脉冲输出（最高 100KHz），另外可通过配置 MX2H-1PG-BD 扩展卡增加 1 路高速脉冲输出（最高 100KHz），支持脉冲序列输出（PTO）和脉宽调制（PWM）两种模式。提供了包络线脉冲输出功能，用于控制伺服或步进电机的多步加速、匀速和减速过程，最多可达 256 步，适用于速度改变频繁的场合，同时提供了当前值读取（DABS）、原点回归（DZRN）、相对位置控制（DDRVI）、绝对位置控制（DDRVA）等指令，方便实现精确的定位功能。
- 高速输入输出模块包括 2 路高速计数模块 MX2H-2HC、2 路高速输出模块 MX2H-2PG，其中 MX2H-2HC 支持 2 路 200KHz 高速输入。MX2H-2PG 支持 2 路独立的 200KHz 高速输出，支持直线插补、圆弧插补指令。
- 基本指令处理速度为 0.084us/ 步，综合指令处理速度为 1K 步 /ms，程序容量为 24K 步，一般应用在中型控制系统中。

MX2V 系列 PLC

MX2V 系列 PLC 可分为一体式结构和模块式组合结构，是一款运动功能强大的 PLC。容量可达 32K 步，D 寄存器数量为 32768 个。在主机上具有 8 路 200KHz 高速输入，8 路 200KHz 高速输出，可集成 4 轴具有插补功能的高速输出。支持 MX2H 系列所有功能及扩展模块。

MX3H 系列 PLC

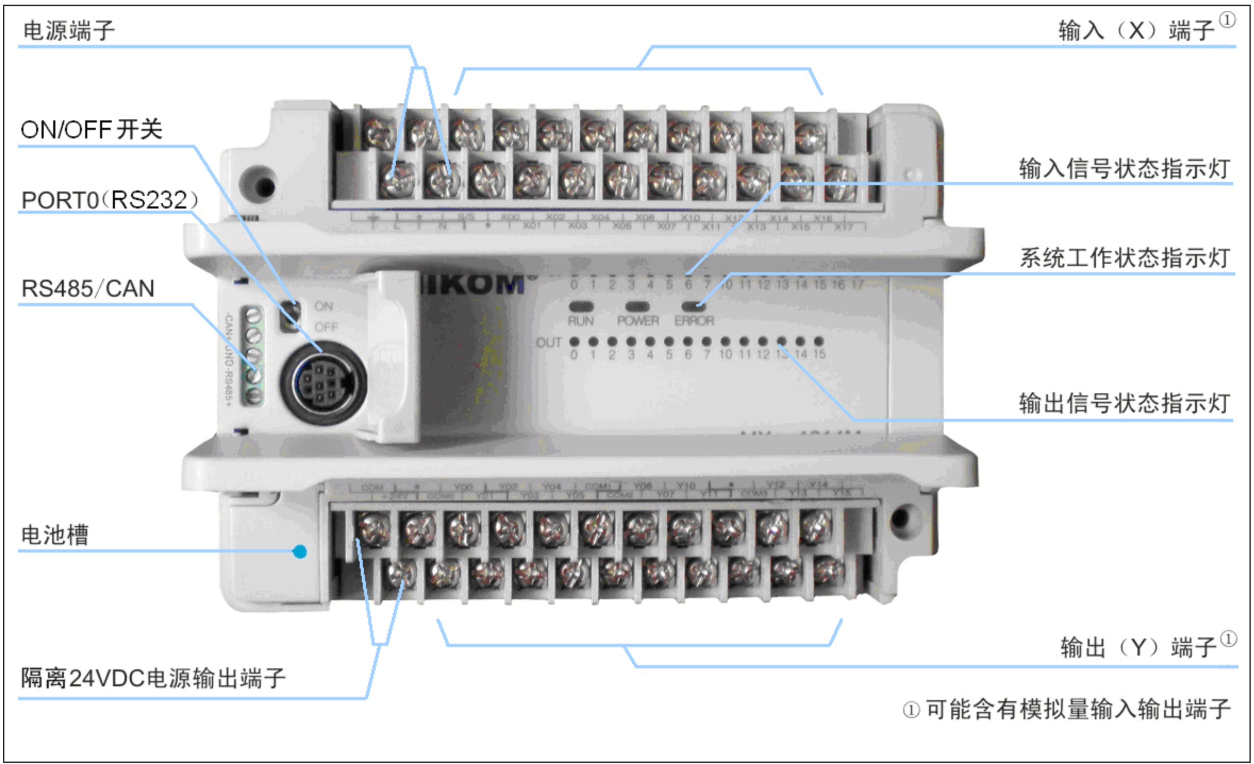
MX3H 系列 PLC 为模块式组合结构，程序容量为 64K 步，一般应用在大型控制系统中。具体资料请详见《MX3H 系列中型 PLC 编程手册》。

MX 系列 PLC 具有完善的保护机制，可通过设置“禁止格式化”、“禁止上载”、“下载密码”、“上载密码”、“监

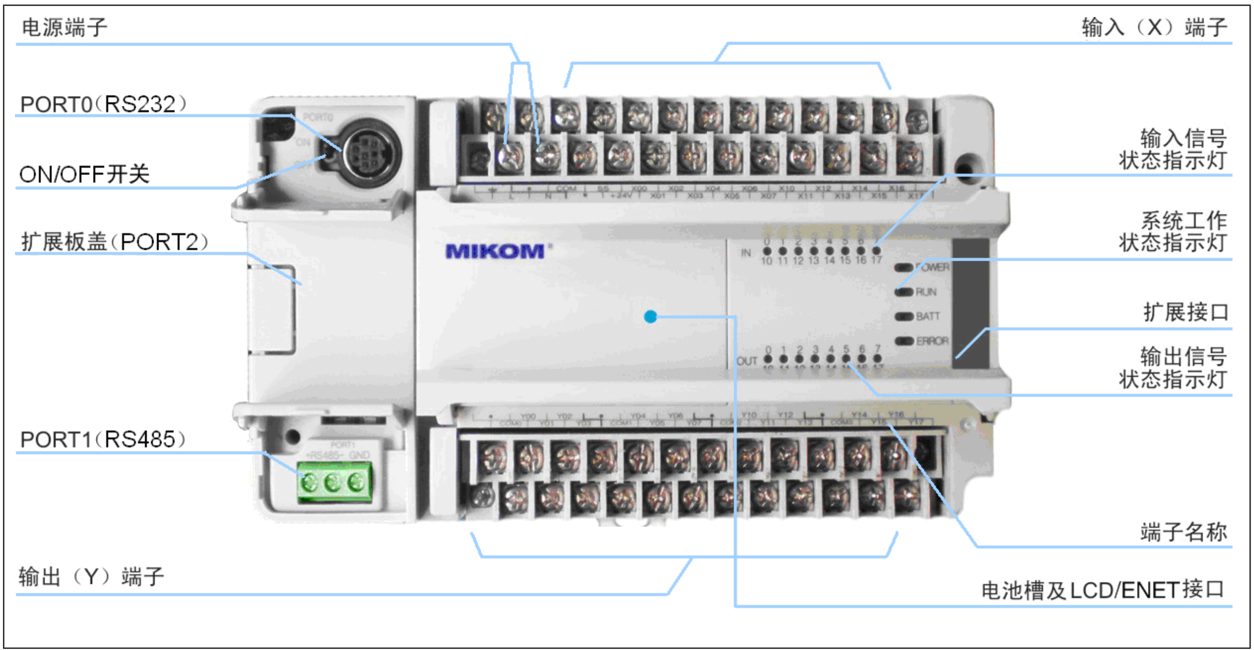
控密码”、“工程密码”、“主程序和子程序加密”及“指定 PLC 序列号”，有效保护用户的知识产权。
MX 系列 PLC 拥有丰富的内置编程资源，其编程软件 MXProgrammer 符合 IEC61131-3 国际标准，提供了梯形图（LAD）、指令列表（IL）、顺序功能图（SFC）三种标准化编程语言，三种编程语言之间可以相互转换，同时可实现程序的监控调试。

1.2 产品外形

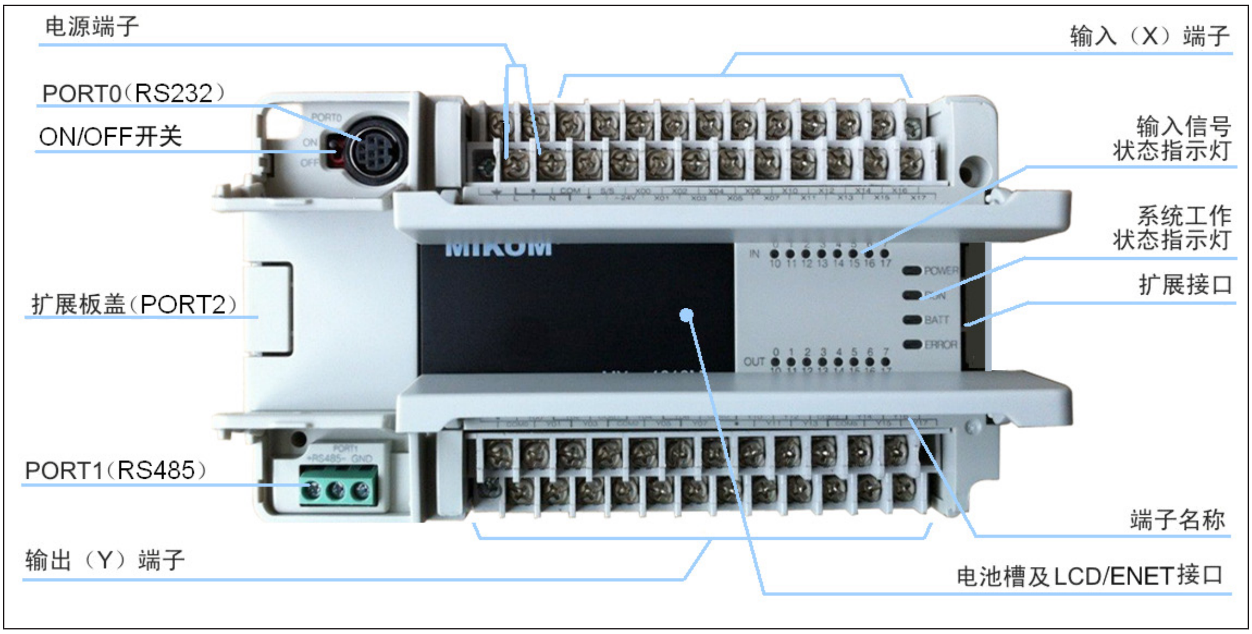
1.2.1 MX1H 系列产品外形



1.2.2 MX2H 系列产品外形



1.2.3 MX2V 系列产品外形



1.3 MX 系列产品性能规格和技术指标

- 常规性能与资源

		MX1H	MX2H	MX2V
常规性能	扫描方式	周期循环扫描		
	恒定扫描方式	支持		
	I/O 输出	每完成一次扫描都刷新一次，支持立即刷新指令（主模块及扩展模块）		
	逻辑最大 I/O 点	256	512	
	最大特殊功能模块数量	可扩展 2 个模拟量模块	可扩展 8 个特殊功能模块	
	高速脉冲输出	2×100KHz（晶体管）	3×100KHz（晶体管）	8×200KHz 支持 4 路差分输出
	单相计数通道	2×50KHz； 4×10KHz		8×200KHz 支持 4 路差分输入
	双相计数通道	1×20KHz＋2×10KHz	1×25KHz＋2×5KHz	4×200KHz
	计数频率总和	80KHz	100KHz	800KHz
	基本指令运行时间	0.084us		0.050us
	综合指令运行时间	1ms /1K 步①		0.6ms /1K 步①
	实时时钟	支持（掉电保持至少 3 年）		
	数字滤波功能	主模块和扩展模块的所有的输入端口均支持数字滤波 0.8ms，1.6ms，3.2ms，6.4ms，12.8ms，25.6ms，51.2ms 可调		
	硬件滤波功能	X0 ～ X1 端口的滤波时间为 10us；X2 ～ X7 端口的滤波时间为 50us；其余 I/O 端口约为 1ms		X0 ～ X7 端口的滤波时间为 1us；其余 I/O 端口约为 1ms
指令种类	基本指令	31 个		
	步进指令	2 个		
	应用指令	174 个	236 个	264 个

			MX1H	MX2H	MX2V
存储器	用户程序		12K 步	24K 步	32K 步
	元件保持		1K 字	用户设定	
	元件保持时间		永久保持	后备电池，保持时间至少 3 年	
	数据块		D0 ～ D3999 共 4000 个	D0 ～ D7999 共 8000 个	D0 ～ D32767 共 32768 个
元件资源	输入点		X0 ～ X177 共 128 点	X0 ～ X377 共 256 点	
	输出点		Y0 ～ Y177 共 128 点	Y0 ～ Y377 共 256 点	
	定时器	100ms	100ms 精度：T0 ～ T209 共 210 点（0.1 ～ 3276.7 秒）		
		10ms	10ms 精度：T210 ～ T251 共 42 点（0.01 ～ 327.67 秒）		
		1ms	1ms 精度：T252 ～ T255 共 4 点（0.001 ～ 32.767 秒）		
	计数器	16 位	C0 ～ C199 共 200 点		
		32 位	C200 ～ C234 共 35 点		
	高速计数器	单相单端计数输入（32 位）	C235 ～ C245 共 11 点		
		单相增减计数输入（32 位）	C246 ～ C250 共 5 点		
		双相计数输入（32 位）	C251 ～ C255 共 5 点		
	数据寄存器		D0 ～ D3999 共 4000 个	D0 ～ D7999 共 8000 个	D0 ～ D32767 共 32768 个
	局部数据寄存器		V0 ～ V63 共 64 个		
	变址寻址寄存器		Z0 ～ Z15 共 16 个		
	特殊数据寄存器		SD0 ～ SD511 共 512 个		
	辅助继电器		M0 ～ M2047 共 2048 点	M0 ～ M4095 共 4096 点	
	局部辅助继电器		LM0 ～ LM63 共 64 点		
	特殊辅助继电器		SM0 ～ SM511 共 512 点		
	状态继电器		S0 ～ S1023 共 1024 点		
中断资源	内部定时中断		3 个		
	外部中断		16 个（中断触发边沿可由用户设定，对应 X0 ～ X7 端子上升沿和下降沿）		
	高速计数器中断		7 个（可支持多用户中断）		
	串口中断		8 个	12 个	
	PTO 输出完成中断		2 个	3 个	8 个
	CAN0 接收中断		1 个		
	主电源失电中断		1 个		
通讯资源	通讯口		PORT0：RS232 PORT1：RS485 CAN 接口	PORT0：RS232　PORT1：RS485 PORT2：RS232/RS485 以太网接口及 CAN 接口	
	通讯协议		自由口 /Modbus(RTU)/MXLinK/ 编程口协议；以太网卡支持 Modbus 与 Modbus TCP 协议		

			MX1H	MX2H	MX2V
特殊功能	高速脉冲输入		X0 ～ X7 可实现高速计数、中断脉冲捕捉等功能 X0 ～ X1 端口计数最高频率达 50KHz X2 ～ X5 端口计数最高频率达 10KHz		
	高速脉冲输入扩展		无	MX2H-2HC 扩展模块增加 2 路 200K 高速输入和 4 路普通输出	
	高速脉冲输出扩展		无	MX2H-2PG 扩展模块增加 2 路 200K 的高速输出和 2 路普通输出	
	子程序调用		最多允许调用 16 个用户子程序	最多允许调用 64 个用户子程序	
			允许 5 级子程序嵌套调用，支持局部变量，每个子程序最多可提供 64 个字和 64 个位参数传递，支持变量重命名		
	安全策略	下载密码	密码不超过 8 个字符，每个字符为字母或数字，区分大小写		
		上载密码			
		监控密码			
		工程密码	密码不超过 16 个字符，每个字符为字母或数字，区分大小写		
		主程序加密	密码不超过 16 个字符，每个字符为字母、数字或符号，区分大小写		
		子程序加密			
		指定序列号	输入 PLC 序列号的后 9 位数字		
		高级策略	禁止格式化、禁止上载，密码重试次数限制②		

① 1K 步标准测试程序，其中基本指令占 60%，应用指令占 40%。
② 30 秒之内密码验证重试超过 5 次，锁定密码验证功能 10 分钟。

☞ 注意

1. X，Y 元件的地址编号是采用 8 进制编址，例如：地址 X10 代表第 9 个输入点。
2. 提供元件强制功能，方便调试和分析用户程序，提高调试效率。最多允许强制 128 个位元件和 20 个字元件。
3. 部分 PLC 内部软元件资源已经保留给内部使用，用户程序内应尽量避免使用此类元件，例如 MXLink 通信所用 D 元件 MX1H：D3000 ～ D3255；MX2H：D7000 ～ D7255；MX2V：D7000 ～ D7255。

● **数字量输入接口规格**

项目		高速输入端口 X0 ～ X7	普通输入端
信号输入方式		差分或源型 / 漏型方式，用户可通过 S/S 进行选择	
电气参数	检测电压	差分：DC5V　单端：DC24V	
	输入阻抗	差分：250Ω 单端：3.3KΩ	4.3KΩ
	输入 ON	外部回路电阻小于 400Ω	
	输入 OFF	外部回路电阻大于 24KΩ	
滤波功能	软件滤波	滤波时间可在 0.8ms、1.6 ms、3.2 ms、6.4 ms、12.8 ms、25.6 ms、51.2 ms 之间由用户编程设定	
	硬件滤波	MX1H/2H:X0 ～ X1 端口的滤波时间为 10us X2 ～ X7 端口的滤波时间为 50us MX2V: X0 ～ X7 端口的滤波时间为 50us 其余 I/O 端口约为 1ms	

● **数字量输出接口规格**

项目		继电器输出端口	晶体管输出端口	差分输出端口
回路电源额定电压		250VAC，30VDC 以下	5 ～ 24VDC	5VDC
电路绝缘		继电器机械绝缘	光耦绝缘	绝缘 IC
动作指示		继电器输出触点闭合 LED 灯亮	光耦被驱动时 LED 灯亮	输出正电压差时 LED 点亮
开路漏电流		—	小于 0.1mA/24VDC	负电平差输出
最小负载		2mA/5VDC	5 mA（5 ～ 24VDC）	—
最大输出电流	电阻负载	4A/2 点组公共端 8A/4 点组公共端 8A/8 点组公共端	0.3A/1 点 0.8A/4 点 1.6A/8 点	20mA/ 点
	感性负载	80VA	7.2W/24VDC	10mA/ 点

项目	高速输入端口 X0 ～ X7	普通输入端
高速功能	MX1H/2H： X0 ～ X5 可实现高速计数、中断脉冲捕捉等功能 X6 ～ X7 可实现中断脉冲捕捉等功能 X0 ～ X1 端口计数最高频率达 50KHz X2 ～ X5 端口计数最高频率达 10KHz 输入频率总和要求小于 100KHz MX2V： X0 ～ X7 可实现高速计数、中断、脉冲捕捉等功能端口计数最高频率达 200KHz，输入频率总和要求小于 800 KHz	
公共接线端	只有一个公共端，为 S/S	

项目	继电器输出端口	晶体管输出端口	差分输出端口
ON 响应时间	最大 20ms	Y0 ～ Y1 小于 5us/10mA 以上；其他：小于 0.5ms/100mA 以上	小于 1us/10mA 以上
OFF 响应时间	最大 20ms		
熔断器保护	无		

● **集成模拟量指标**

项目		指标
转换速度		2ms/ 通道
模拟输入量程	电压输入	- 10V ～ 10V，输入阻抗为 500KΩ
	电流输入	- 20mA ～ 20mA，输入阻抗为 250KΩ
模拟输出量程	电压输出	- 10V ～ 10VDC（外部负载阻抗大于 1KΩ）
	电流输出	0 ～ 20mA（外部负载阻抗小于 500Ω）
分辨率	电压	5mV
	电流	20uA
精度		±1% 全量程

● **高精度模拟量输入模块指标**

项目		指标
模拟电路		24VDC（ - 15% ～ 20%），最大允许纹波电压 5%，30mA（来自主单元的外部电源或外接）
数字电路		3.3VDC 25mA（源于主模块的内部电源）
转换速度		2ms/ 通道
模拟输入量程	电压输入	- 10V ～ 10V，输入阻抗为 500KΩ
	电流输入	- 20mA ～ 20mA，输入阻抗为 250KΩ
分辨率	电压输入	1mV
	电流输入	5uA
精度		±0.2% 全量程
隔离		模拟电路和数字电路之间通过数字隔离器进行隔离；模拟电路与模块输入 24VDC 电源内部隔离；模拟通道之间不隔离

● **模拟量输入模块指标**

项目		指标
模拟电路		24VDC（ - 15% ～ 20%），最大允许纹波电压 5%，48mA（来自主单元的外部电源或外接）
数字电路		3.3VDC，25mA（源于主模块的内部电源）
转换速度		2ms/ 通道
模拟输入量程	电压输入	- 10V ～ 10V，输入阻抗为 500KΩ
	电流输入	- 20mA ～ 20mA，输入阻抗为 250Ω
分辨率	电压输入	5mV
	电流输入	20uA
精度		±1% 全量程
隔离		模拟电路和数字电路之间通过数字隔离器进行隔离；模拟电路与模块输入 24VDC 电源内部隔离；模拟通道之间不隔离

● **模拟量输出模块指标**

项目		指标
模拟电路		24VDC（ - 15% ～ 20%），最大允许纹波电压 5%，158mA（来自主单元的外部电源或外接）
数字电路		3.3VDC，25mA（源于主模块的内部电源）
转换速度		1ms/ 通道（改变所有的通道数不会改变转换速度）
模拟输出量程	电压输出	- 10V ～ 10V（外部阻抗大于 1KΩ）
	电流输出	0 ～ 20mA（外部负载阻抗小于 500Ω） 4 ～ 20mA（外部负载阻抗小于 500Ω）
分辨率	电压输出	5mV
	电流输出	20uA
精度		±1% 全量程
隔离		模拟电路和数字电路之间通过数字隔离器进行隔离；模拟电路与模块输入 24VDC 电源内部隔离；模拟通道之间不隔离

- 热电偶温度输入模块指标

项目	指标			
	摄氏（℃）		华氏（℉）	
模拟电路	24VDC（－ 15% ～ 20%），最大允许纹波电压 5%，30mA（来自主单元的外部电源或外接）			
数字电路	3.3VDC 25mA（源于主模块的内部电源）			
热电偶类型	K、J、E、N、T、R、S			
转换速度	100ms/ 通道			
额定温度范围	类型 K	-100℃～ 1200℃	类型 K	-148 ℉～ 2192 ℉
	类型 J	-100℃～ 600℃	类型 J	-148 ℉～ 1112 ℉
	类型 E	-100℃～ 850℃	类型 E	-148 ℉～ 1562 ℉
	类型 N	-100℃～ 1200℃	类型 N	-148 ℉～ 2192 ℉
	类型 T	-200℃～ 300℃	类型 T	-328 ℉～ 572 ℉
	类型 R	0℃～ 1600℃	类型 R	32 ℉～ 2912 ℉
	类型 S	0℃～ 1600℃	类型 S	32 ℉～ 2912 ℉
最低分辨率	类型 K	0.3℃	类型 K	0.54 ℉
	类型 J	0.2℃	类型 J	0.36 ℉
	类型 E	0.3℃	类型 E	0.54 ℉
	类型 N	0.3℃	类型 N	0.54 ℉
	类型 T	0.2℃	类型 T	0.36 ℉
	类型 R	0.5℃	类型 R	0.9 ℉
	类型 S	0.5℃	类型 S	0.9 ℉
精度	±（1% 全范围 ±1.5℃）			
隔离	模拟电路和数字电路之间通过数字隔离器进行隔离； 模拟电路与模块输入 24VDC 电源内部隔离； 模拟通道之间不隔离			

- 热电阻温度输入模块指标

项目	指标			
	摄氏（℃）		华氏（℉）	
模拟电路	24VDC（－ 15% ～ 20%），最大允许纹波电压 5%，25mA（来自主单元的外部电源或外接）			
数字电路	3.3VDC 25mA（源于主模块的内部电源）			
热电阻类型	Pt100、JPt100、Cu100、Cu50			
转换速度	100ms/ 通道			
额定温度范围	Pt100	-150 ～ 600℃	Pt100	-238 ℉～ 1112 ℉
	JPt100	-150 ～ 600℃	JPt100	-238 ℉～ 932 ℉
	Cu100	-30 ～ 120℃	Cu100	-22 ℉～ 248 ℉
	Cu50	-150 ～ 600℃	Cu50	-22 ℉～ 248 ℉
最低分辨率	Pt100	0.2℃	Pt100	0.36 ℉
	JPt100	0.2℃	JPt100	0.36 ℉
	Cu100	0.2℃	Cu100	0.36 ℉
	Cu50	0.2℃	Cu50	0.36 ℉
精度	±1% 全量程			
隔离	模拟电路和数字电路之间通过数字隔离器进行隔离；模拟电路与模块输入 24VDC 电源内部隔离；模拟通道之间不隔离			

- 称重模块指标

项目		指标
模拟电路		24VDC（－ 15% ～ 20%），最大允许纹波电压 5%，48mA（来自主单元的外部电源或外接）
数字电路		3.3VDC，25mA（源于主模块的内部电源）
转换速度		2ms/ 通道
称重功能	传感器特性	0 ～ 4mV/V
	测量信号范围	－ 40mV ～ 40mV
	分辨率	50uV
分辨率	电压输出	10VDC（额定值）±5%
	最大电流	240mA
	负载能力	40 ～ 4010Ω
精度		±0.2% 全量程
连接传感器最大距离		100 米
隔离		模拟电路和数字电路之间通过数字隔离器进行隔离；模拟电路与模块输入 24VDC 电源内部隔离；模拟通道之间不隔离

1.4 MXProgrammer 编程软件

MXProgrammer 是 MX 系列 PLC 产品的编程软件，软件可在公司网站下载。

MXProgrammer 编程软件是标准的 Windows 软件、图形化的 PLC 编程工具，采用鼠标键盘操作。编程时可选用 3 种标准语言：梯形图（LAD）、指令列表（IL）、顺序功能图（SFC）。MXProgrammer 编程软件与 PLC 的连接采用串口编程电缆或网线（PLC 需配置 MX2H-ETH-BD 以太网卡）。

● 基本配置

MXProgrammer 编程软件可在 PC 机上运行，并且需要安装在 Microsoft Windows 系列操作系统之上。兼容的操作系统有 Windows XP、Windows7 等。

MXProgrammer 要求的最低配置和推荐配置如下：

项目	最低配置	推荐配置
CPU	相当于 Intel 公司的 Pentium233 或以上级别的 CPU	相当于 Intel 公司的 Pentium1G 或以上级别的 CPU
内存	64M	128M
显卡	640×480 分辨率，256 色模式下	800×600 以上分辨率，65535 色模式下
通讯口	使用串口连接需有一个 DB9 型插座输出的 RS232 串口（或通过 USB － RS232 转换器使用 USB 接口，需另行配备转换器） 使用网线连接需有一个 RJ45 接口（PLC 需配置 MX2H-ETH-BD 以太网卡）	
操作系统	Windows XP/ Windows 7	
其他设备	MX 系列 PLC 专用串口编程电缆	

● **MXProgrammer** 编程软件安装过程

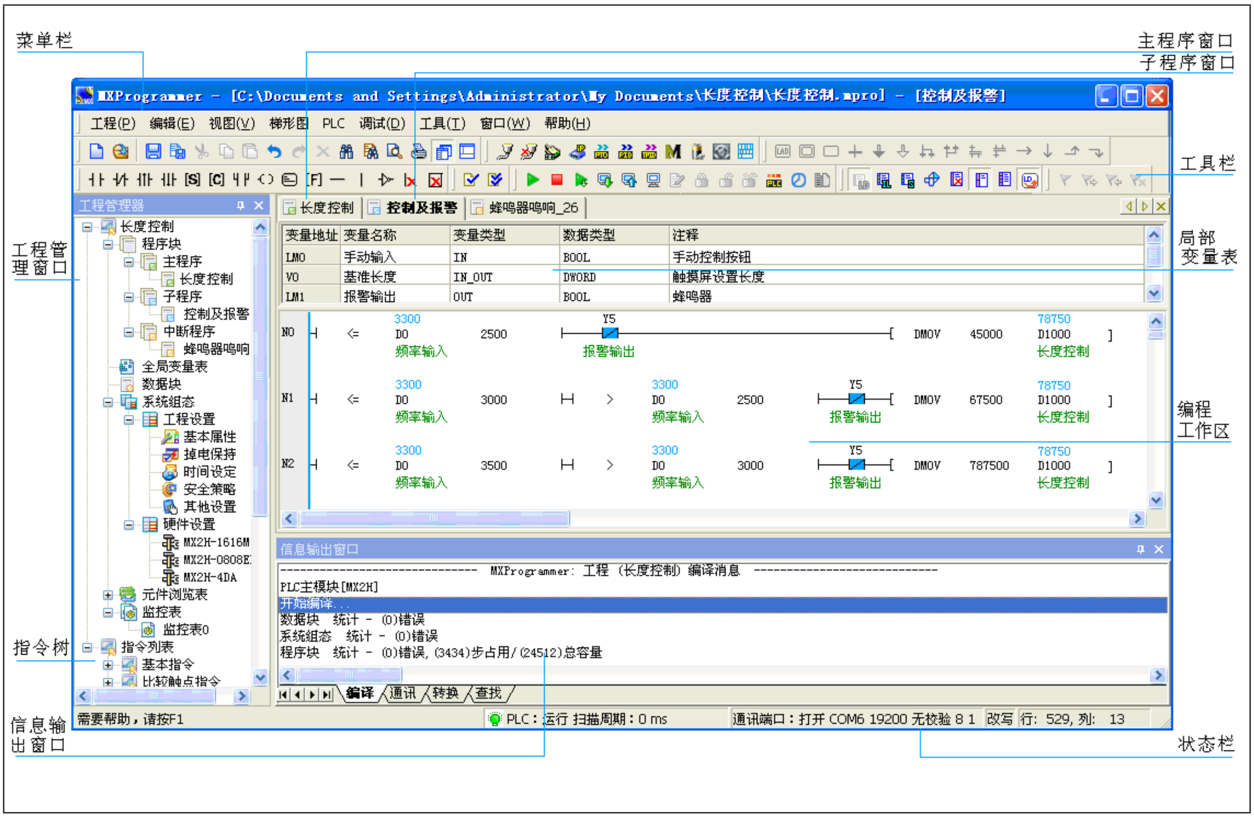
麦科电气技术有限公司发布的 MXProgrammer 安装包为单独的可执行程序，双击即可启动安装过程，根据安装向导一步步进行安装。用户可根据自己的需要选择不同的安装路径。

安装完毕后，在开始菜单中会发现 Mikom Electronics 程序组；同时安装程序还会把 MXProgrammer 快捷方式图标安装在桌面上，双击快捷方式图标可运行程序。

卸载操作：通过 Windows 控制面板可进行软件卸载。若要升级安装新版本的 MXProgrammer 软件，请先将旧版本软件卸载。

● **MXProgrammer** 运行界面

本程序运行主界面包括菜单、工程管理器窗口、指令树、信息输出窗口、状态栏、工具栏、编程工作区、局部变量表、子程序窗口和主程序窗口。



有关 MXProgrammer 软件编程环境的应用请详见软件的“帮助”→“帮助主题”。

第二章　功能说明

本章主要介绍了 MX 系列 PLC 的编程机制、系统配置的方法、程序文件的导入导出，以及如何设置 PLC 运行和操作模式，最后还介绍了 MXProgrammer 软件的有关功能。

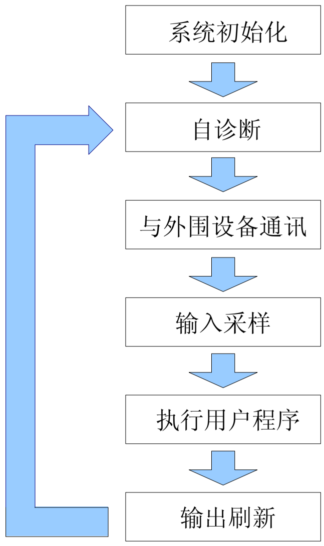
2.1 编程机制

2.1.1 PLC 运行机制

MX 系列 PLC 是按照循环扫描方式运行工作的。

PLC 运行后，首先进行系统初始化，将内部电路恢复到初始状态，然后进行自诊断，检测内部电路是否正常，以确保系统能正常运行，诊断结束后对通讯端口进行扫描，若接有外围设备则与其通讯。通讯接口无外围设备或通讯完成后，系统开始进行输入采样，然后根据输入采样结果依次执行用户程序，程序运行结束后对输出进行刷新，完成一次扫描。以上过程完成后，系统又返回重新开始自诊断，以后不断循环执行上述过程。

如下图所示：



PLC 有两种工作状态：运行（RUN）状态和停止（STOP）状态；当 PLC 工作在 RUN 状态时，系统会完整地执行上述的循环过程；当 PLC 工作在 STOP 状态时，系统不执行用户程序，但系统会执行循环过程中的其他操作。PLC 正常工作时应处于 RUN 状态，而在编写和修改程序时，应让 PLC 处于 STOP 状态。PLC 的两种工作状态可通过开关进行切换，也可以通过编程软件进行切换。

2.1.2 用户程序运行看门狗功能

每个扫描周期系统都会对用户程序的运行时间进行监控，一旦发现用户程序运行时间超过看门狗设定值，将停止用户程序运行。用户可以在 MXProgrammer 后台软件界面的“系统组态→时间设定”对话框中设置看门狗时间。

2.1.3 恒定扫描运行模式

系统在运行状态下，每个扫描周期的时间都为恒定时即为恒定扫描运行模式。用户可以在 MXProgrammer 后台软件界面的“系统组态→时间设定”对话框中设置恒定扫描时间。恒定扫描时间默认值为 0，即无恒定扫描模式；当实际扫描时间大于恒定扫描时间时，程序按照实际扫描时间运行。

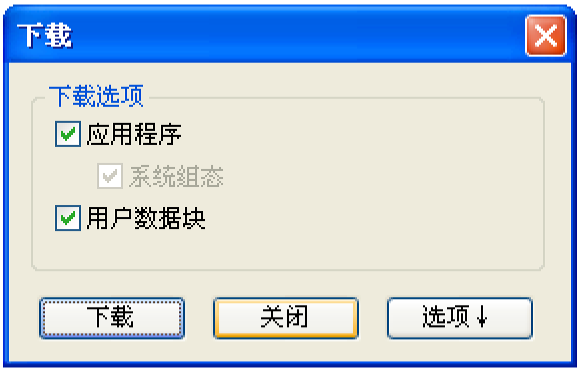
☞ 注意

恒定扫描时间设置值不能大于看门狗定时时间。

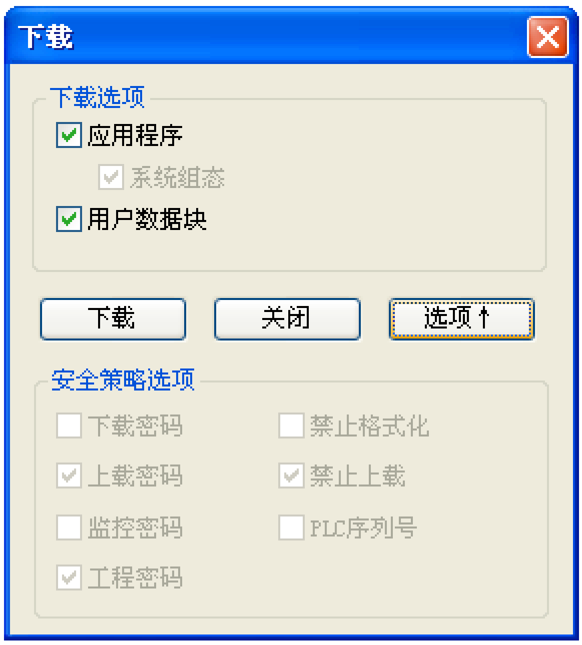
2.1.4 用户文件下载和存储

用户文件包括：应用程序及系统组态、用户数据块。用户可以选择下载应用程序及系统组态、用户数据块。相应的用户辅助信息文件也会被捆绑下载。

下载对话框如下图所示：



点击“选项”，可以看到被捆绑下载的“安全策略选项”。



☞ 注意

1. 为确保下载文件能正常固化到主模块中，文件下载完毕后一段时间内（大于 30 秒钟）应保证主模块的供电正常。
2. MX2H 系列 PLC 备份电池的失效将会导致用户程序辅助信息文件丢失，不能正确上载应用程序相关的注释信息，并且系统会报“用户信息文件错误”，但不会影响应用程序的正常执行。

2.1.5 元件初始化

PLC 在进入运行状态时（STOP → RUN），系统将会对所有的元件进行初始化，不同种类存储器的初始化有不同的优先级，如下所示：

存储器种类	电源（断电→通电）	PLC 运行（STOP → RUN）
掉电保持数据	高	高
数据块（在系统组态的其他设置中选中“数据块有效”）	中	中
元件值（在系统组态的其他设置中选中“元件值保持”）	—	低

2.1.6 掉电保持功能

当 PLC 发生掉电时，系统会停止用户程序的运行，同时将系统组态中指定为掉电保持区内的软元件的数据值保存到掉电备份文件中，PLC 重新上电后，如果掉电备份文件正确，指定的软元件的数据值将会被恢复为掉电前的保存值。非掉电保持区内的软元件在重新上电后清零，如果备份文件丢失或错误，系统将对所有元件进行清零处理。MX 系列 PLC 的掉电保持区可设定 2 组。

MX2H 系列的掉电保持区的软元件定义对话框如下：

☞ 注意

1．MX1H 系列 PLC 的掉电保持方式为 FLASH ROM，无需备份电池，其所备电池应用于时钟。

2．MX2H、MX2V 系列 PLC 的掉电保持功能在备份电池正常的情况下才能完成（掉电保持至少 3 年），如果电池不正常，BATT 指示灯常亮，SM4 置位，所有掉电保持区的元件在掉电后其数据将不确定。建议用户编程时使用 SM4 报警，提醒用户及时更换电池，更换电池时必须在 1 分钟内完成。

3．MX 系列 PLC 的掉电保持区不允许交叉使用。

2.1.7 输入点的滤波功能

主模块和扩展模块的所有输入端口除了支持硬件滤波外还支持数字滤波，用来滤除端口的干扰信号。

硬件滤波：其时间是固定的。

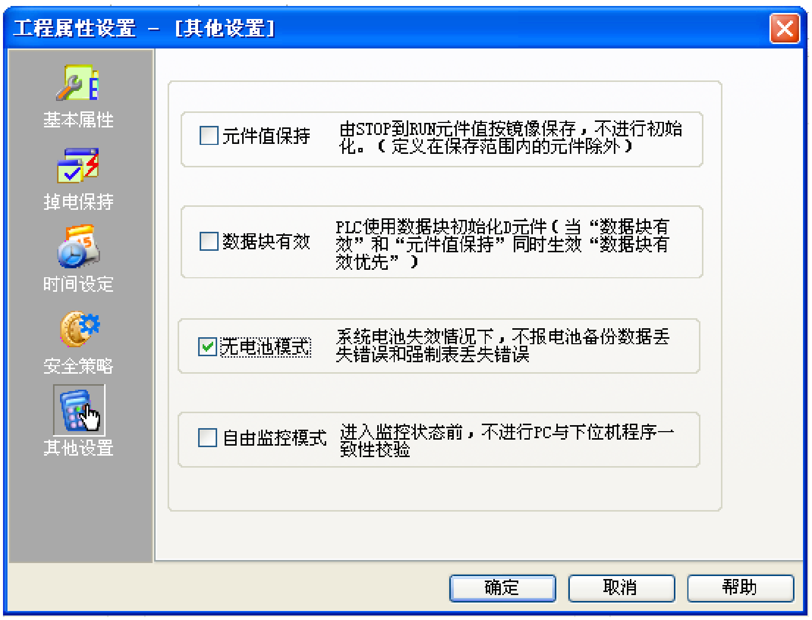
MX2V：X0 ～ X7 端口的滤波时间为 1us；其余 I/O 端口约为 1ms

MX1H、MX2H：X0 ～ X1 端口的滤波时间为 10us，X2 ～ X7 端口的滤波时间为 50us, 其余 I/O 端口约为 1ms。

数字滤波：其时间可在系统组态中设置为 0.8ms、1.6 ms、3.2 ms、6.4 ms、12.8 ms、25.6 ms、51.2 ms，也可以通过 REFF 指令来设定滤波常数，在两者同时使用的情况下，REFF 指令优先。

2.1.8 无电池模式

MX 系列 PLC 的主模块可以在“系统组态→工程设置→其他设置”中选择无电池模式，当选择无电池模式时，系统将不报因为电池缺失所引起的系统错误（电池备份数据丢失错误、强制表丢失错误），无电池模式的选择对话框如下图所示：



2.1.9 用户程序保护措施

MX 系列 PLC 设有多级密码保护等安全措施。

用户程序的保护措施包括“程序密码、禁止格式化、下载密码、禁止上载、上载密码、监控密码、设备序列号、工程密码”。

程序密码包括对主程序、子程序和中断服务子程序的加密。加密后的程序其内容不能被查看和编辑，只有打开解密对话框输入正确密码后程序方可解密，进行查看和编辑工作。

加密方法：右击要加密的程序，在右键菜单中先选中“属性”，然后在加密属性中输入“密码”和“确认密码”，即可实现程序的加密。

解密方法：右击要解密的程序，在右键菜单中先选中“属性”，然后输入正确的密码即可解密。

对于“禁止格式化、下载密码、禁止上载、上载密码、监控密码、设备序列号、工程密码”的设置请参见 *2.2.1* 系统组态。

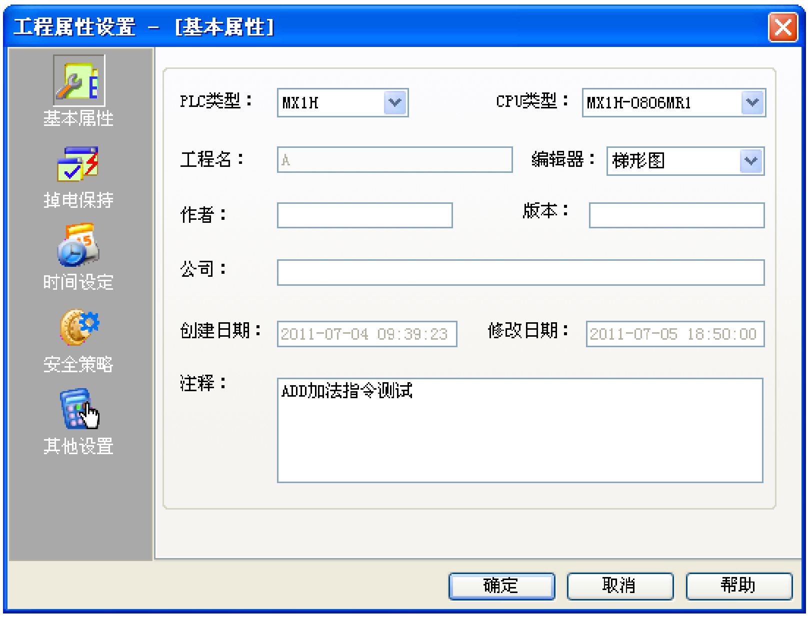
2.2 系统配置

2.2.1 系统组态

系统组态包括工程设置和硬件设置，工程设置包括基本属性、掉电保持、时间设定、安全策略及其他设置，硬件设置中包括输入模式设置、输出模式设置及通讯口设置。

1．基本属性

在基本属性对话框中可以在“系统组态→工程设置→基本属性”中选择或更改 PLC 类型和 CPU 类型及其它信息的编辑或显示，如下图所示：



设置和更改工程的基本属性

1. 更改 PLC 类型
2. 更改 CPU 类型
3. 更改默认的编程语言
4. 设置工程的作者、版本、公司名称及对工程的注释
5. 显示工程名
6. 显示创建日期及修改日期

☞ 注意

在基本属性中切换 PLC 类型和 CPU 类型后，系统组态中主模块的数字滤波、输入点开机模式、安全策略、时间设定、其他设置都保持不变。通讯口的设置会改变，需重新配置。而只有在相同系列的 PLC 之间切换时，掉电存储区保持不变。

2. 掉电保持

在掉电保持对话框中可指定软元件 M、S、D、C、T 的掉电保持区，MX1H、MX2H、MX2V 系列 PLC 掉电保持区分别可指定两组，可保存的元件地址范围和数量会因 PLC 型号的不同而异，以 MX1H 系列为例，掉电保持对话框如下所示：

掉电时：

PLC 将按照上图中定义的范围，将要保持的元件保存到掉电备份文件中。

上电时：

PLC 检查掉电备份文件是否正确，如果正确则将保持元件的值，恢复成上次掉电时的保存值；如果备份文件丢失或错误，系统将所有元件值清零。

3. 时间设定

在时间设定对话框中可设定看门狗定时值、恒定扫描时间、通信背景时间。如下图所示：

（1）看门狗定时

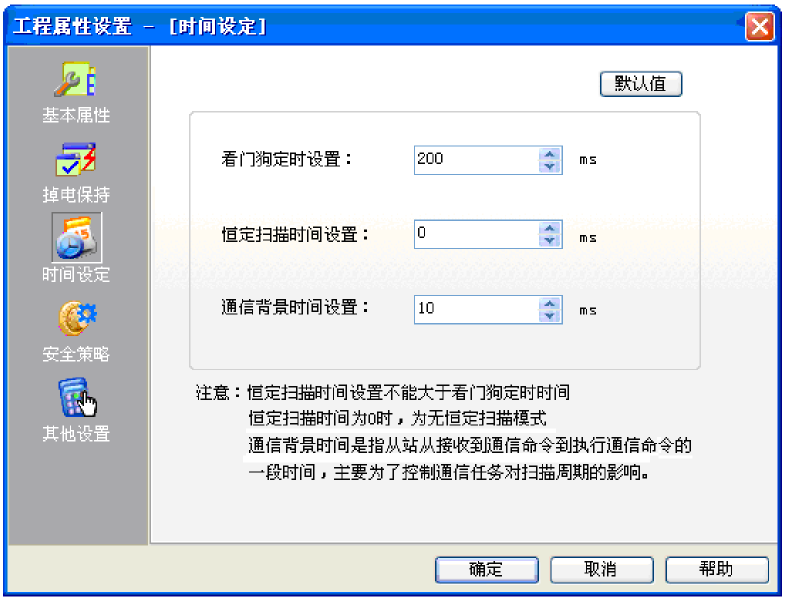
功能：设置用户程序运行看门狗时间。看门狗时间是指允许用户程序运行的最大时间，当用户程序的实际执行时间超过看门狗时间时，PLC 会停止用户程序，ERROR 灯常亮，输出按系统配置进行输出。看门狗定时器时间可设定范围为：100ms ～ 1000ms，默认值为 200ms。

（2）恒定扫描时间

功能：恒定扫描时间是指系统在恒定时间扫描寄存器的时间。读取系统的恒定时间扫描设定寄存器，在恒定时间内对用户程序只作一次扫描。恒定时间可设定范围为：0ms ～ 1000ms。默认值为 0ms，表示无恒定扫描时间。

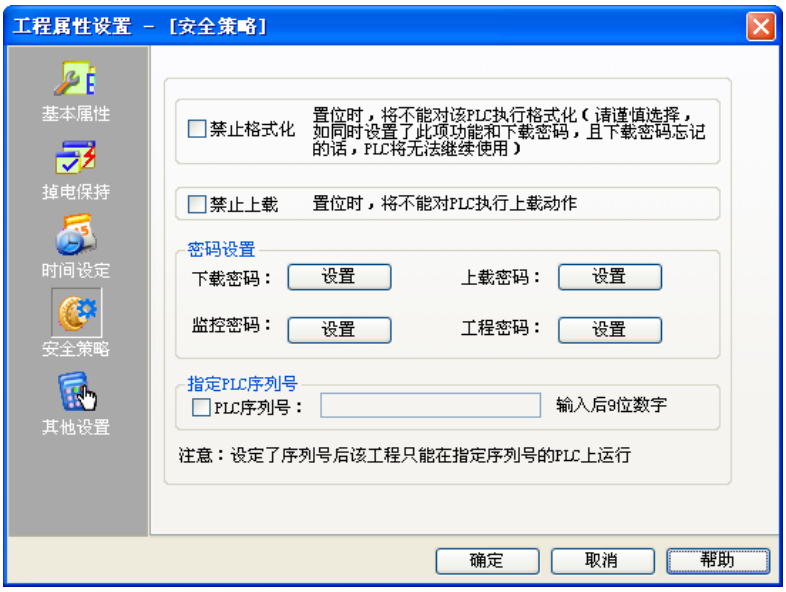
（3）通信背景时间

功能：通信背景时间是指 Modbus 从站从接收到通信命令到执行通信命令的一段时间，主要是为了控制通信任务对扫描周期的影响。



4．安全策略

在安全策略对话框中，可设置禁止格式化、禁止上载、各种密码设置及 PLC 序列号，如下图所示：



（1）禁止格式化

在系统组态选择禁止格式化并下载系统组态到 PLC 后，不能通过格式化删除 PLC 内部的用户程序、系统组态、和数据块。

要解除禁止格式化，必须通过重新下载新的系统组态，同时系统组态不应设置禁止格式化选项。（请谨慎选择，如果同时设置了此项功能和下载密码，且下载密码忘记的话，PLC 将无法继续使用）。

（2）禁止上载

在系统组态中选择禁止上载并下载系统组态到 PLC 后，则以后用户即使在有上载密码的情况下也不能上载。要解除禁止上载必须重新下载系统组态，同时系统组态不应设置禁止上载选项。

（3）密码设置

下载密码、上载密码和监控密码是需要下载到 PLC 中才生效的，绑定到 PLC 中对 PLC 进行有效保护。例如下载密码下载到 PLC 以后，下一次下载操作时，系统会对后台软件中工程的下载密码与 PLC 中绑定的下载密码进行比较，若两者相同，则不提示输入下载密码，否则，提示输入下载密码。

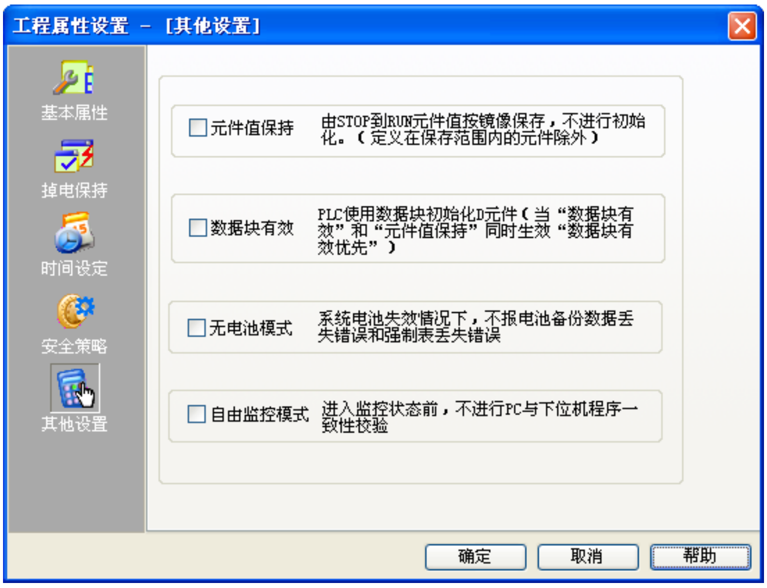
工程密码一旦设定，则在每次打开工程时都需要输入工程密码（工程密码要谨慎设置，一旦密码丢失，该工程无法恢复）。使用密码时需要注意，下载密码、上载密码和监控密码的设置不超过 8 个字符，工程密码的设置不超过 16 个字符，每个字符为字母或数字组合，区分大小写。设置密码后，相应的下载密码、上载密码、监控密码、工程密码后的“设置”将显示为“更改”。

（4）PLC 序列号

PLC 序列号指定后，该工程和其编译结果（UCD 文件）只能在对应序列号的 PLC 中运行，对其他 PLC 可以正常下载，但下载后无法运行。在实际工程应用中，在不打开工程的情况下，可直接向 PLC 中下载 UCD 文件，提高了保密性。

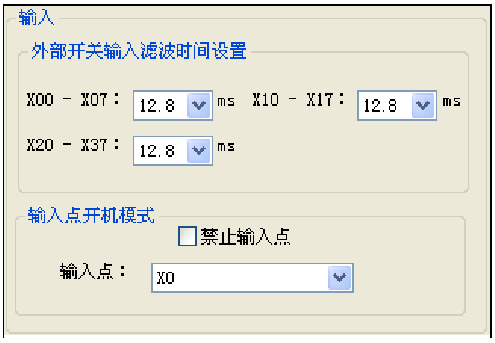
5．其他设置

在其他设置对话框中可设定元件值保持、数据块有效、无电池模式及自由监控模式。如下图所示：



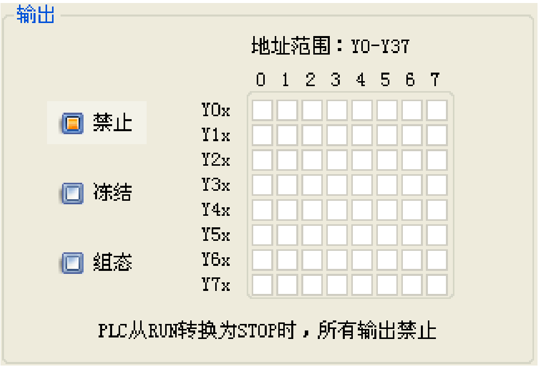
6．输入模式设置

在硬件设置中有输入模式设置区域，在其中可设置外部开关输入的滤波时间及输入点开机模式。主模块和扩展模块的所有的输入端口均支持数字滤波，滤波时间为 0.8ms、1.6 ms、3.2 ms、6.4 ms、12.8 ms、25.6 ms、51.2 ms。在没有选择“禁止输入点”时，可以使用 PLC 的外部输入点启动 PLC 运行。MX2H 系列 64 点 PLC 的输入模式设置区域如下图所示：



7．输出模式设置

在硬件设置中设有输出模式设置区域，在其中可设置 PLC 从 RUN → STOP 时的输出点状态，MX2H 系列 64 点 PLC 的输出模式设置如下图所示：



8．通讯口设置

在硬件设置中设有通讯口设置区域，在其中可对 PLC 主模块的两个通讯口进行设置，包括选择通讯协议和具体的协议参数。

通讯口 0 默认是编程口协议，通讯口 1 默认是无协议。

如下图所示：

（1）编程口协议

通讯口 0 出厂默认的是编程口协议。编程口协议是 MX 系列 PLC 编程软件通讯的专用协议。在编程口协议下，可通过 MXProgrammer 软件的串口配置来设置该通讯口的相关参数。

（2）自由口协议

自由口协议是自定义数据文件格式的通讯协议。在自由口协议对话框中，可设定的参数有：波特率、奇偶校验、数据位、停止位、有效字节、起始字符检测、结束字符检测、字符间超时和帧间超时。PLC 选择开关置于 OFF 档时，则强制为编程口协议。

（3）Modbus（RTU）协议

Modbus 协议分为主站和从站，主站可以与从站进行通讯，按照 Modbus 通讯协议的功能码，发送对从站的控制帧，从站只是对主站的控制作出响应。不向主站发送任何数据。Modbus 协议对话框中，可设定的参数有：波特率、奇偶校验、数据位、停止位、主从方式、站号、传送模式、主模块的超时时间及重复次数。

（4）MXLink 协议

MXLink 协议是麦科自主研发的一种小型 PLC 网络 N：N 通讯协议，接入 MXLink 的 PLC 可以自动的互相交换部分 D 元件的数据。

自由口协议、Modbus 协议、MXLink 协议及 CAN 的自由口协议的详细功能和使用方法请参见第十章 通讯功能使用指南。

9．扩展模块配置

MX 系列 PLC 可以在 MXProgrammer 系统组态中配置 I/O 扩展模块、模拟量模块、扩展卡等。

操作步骤：

（1）双击工程管理器中的主模块，打开系统组态界面。

（2）打开工程管理器中的 MIKOM PLC 模块的下拉菜单，该下拉菜单中有 I/O 扩展模块、模拟量模块、扩展卡模块，可 打开所需的菜单，选择指定的模块双击即实现添加。

（3）添加成功的模块会显示在表格中，MXProgrammer 会自动分配其元件地址。分配给模拟量模块的 U 元件地址在全局变量表中会根据相应的配置自动添加注释。

（4）选中模块单击可打开模块通道配置界面，用户可根据需要对此进行相应的配置。

☞ 注意

扩展模块的配置是有顺序的，顺序不同，其地址编号也不同。当实际配置与系统组态中配置不相同时，PLC 信息中报“系统模块配置错误”，ERROR 灯闪烁，但是 PLC 仍然运行。

模拟量模块 MX2H-4TC 的配置界面如下图所示：

2.2.2 数据块

数据块用于设置 D 元件中的缺省值。数据块包含 D 元件设置数值，当数据块下载到控制器中，指定 D 元件将被赋予设置值，从而达到批量设定 D 元件值的目的。在“系统组态”的“其他设置”中选择“数据块有效”，设置完成并通过编译后可以下载到 PLC 中，在 PLC 进入运行状态后 PLC 将首先使用数据块来初始化相关的 D 元件。

● 用户数据块赋值

在数据块编辑器中可以对 D 元件进行初始数据赋值，可以是字或双字形式，但是不能对字节赋值。对于值只需要占用一个字但是需要按照双字使用时，可以用 DD 寄存器，例如“DD100 10”表示将 10 作为一个双字存放到 D100 和 D101 寄存器表示的一个双字中。

数据块赋值可以通过“D 地址 值 1[值 2][值 N…]”的方式为一个或者多个地址赋值，每个数值顺序占用下一个存储区域，系统自动根据值需要占用的字节数分配存储空间和下一个值的起始地址；在行的上方已经指定过起始地址的情况下，也可以通过直接输入数字“值 1[值 2][值 N…]”来完成对存储空间的赋值。

数据块的第一行如果没有一个明确地址赋值，则默认从 D0 开始赋值，其后的行可包含明确或隐含地址赋值，已经显示或者隐含赋值的存储区域不能重复赋值。当在单地址赋值后键入多个数据值或键入仅包含数据值的行时，由编辑器指定隐含地址赋值。编辑器根据先前的地址分配及数据值大小（字或双字）指定适当的 D 内存数量。

数据块中赋值的数据类型可以是：WORD、INT、DWORD、DINT、REAL。数据格式为：二进制、八进制、十进制以及十六进制。

● 寄存器值设置

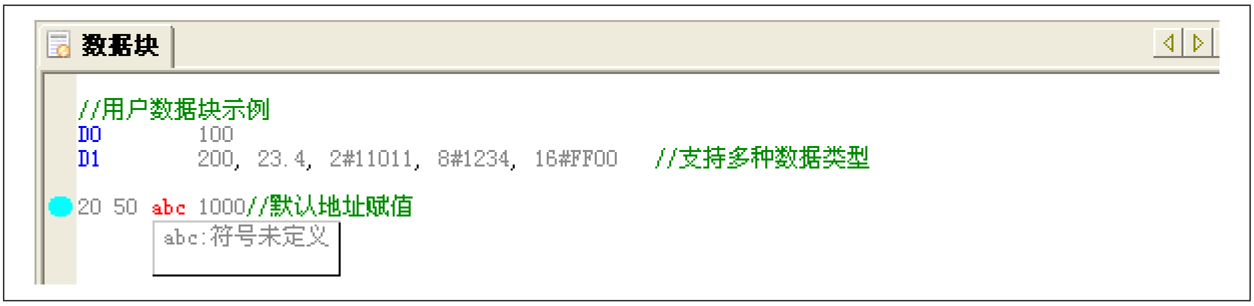
用户数据块编辑器是一个文本编辑器，用于输入寄存器地址、寄存器值和注释，它和指令列表编辑器是非常类似的，支

持文本编辑器中的常用编辑操作，包括复制、剪切、粘贴和删除。在数据块编辑器中输入 D 元件值和注释，系统会自动对齐元件值，且 D 元件用蓝色显示，D 元件数值用灰色显示，注释用绿色显示。在字符串后增加双斜线“//”可以在其后添加注释；在编辑器的任意位置输入双斜线“//”表示当前行后面的内容是注释。在全局变量表中已定义的注释在数据块中可自动显示。

● 正确性校验

输入每一行指令后，如果光标切换到其他行，则会对离开的行进行正确性校验，如果有语法错误，系统会将有错误的文本行显示为红色，当鼠标悬浮在有错误的文本行时，系统会用工具提示的方式提示错误信息。只有校验正确的语句才会格式化显示。在一次粘贴多行文本内容到编辑器中时，编辑器会对每一行作语法检查。

有语法错误时，请将鼠标放在有错误的位置，系统显示如下图所示：



在图中，由于数据格式为十进制时不能使用数值外的字符赋值，因此系统不能识别“abc”。

● 用户数据块编译

编辑完成用户数据块后，点击工具栏上的“”按钮，可以检查当前用户数据块中是否有错误，如果无错通过编译，则信息输出窗口中显示如下内容：

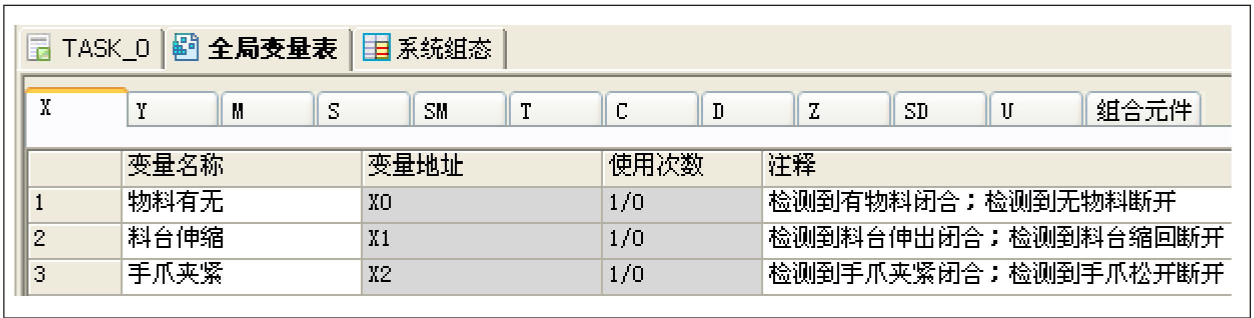
2.2.3 全局变量表

全局变量表用于定义 PLC 的某个软元件的名称、地址、使用次数及注释，该软元件名称及元件注释可在整个工程中被访问到。全局变量表包括变量名称、变量地址、使用次数及注释。

全局变量表是按元件分页进行显示的，在全局变量表中可以查看和编辑变量名称和元件注释（但对 SM、SD、U 元件的注释一栏不可编辑）。在编辑过程中可以对其进行全局变量表的合法性检查：全局变量表中输入的变量名称必须是唯一的，输入确认后，对于重复的变量名称，系统用不同的颜色显示。元件注释最大长度为 20 个字。

全局变量表的定义规则：由大写字母、小写字母、阿拉伯数字、下划线、汉字组合而成。变量名称不能是单独的数字也不能以数字开始，字母不区分大小写，长度不能超过 8 个字，不能使用元件类型字母加数字作为变量名称。变量名称中不能包含空格，不能使用和关键字相同的名称，保留的关键字包括：基本数据类型名称、指令名称、指令列表语言中的运算符。

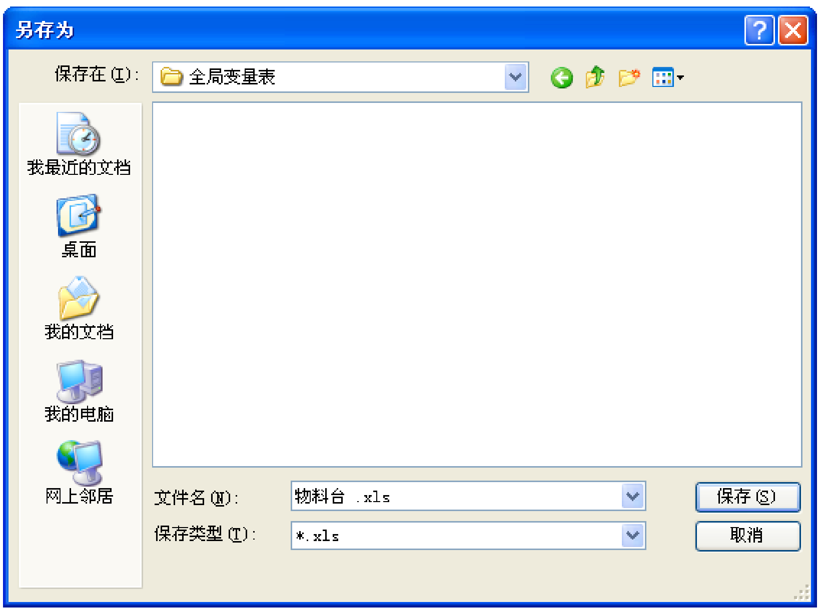
全局变量表如下所示：



软件提供了全局变量表的导入和导出功能，当多个工程需要定义相同的软元件时，可以通过全局变量表导入、导出功能，在不同的工程中多次使用，避免重复编写。MXProgrammer 使用 Excel 表格的形式管理导出的全局变量表，全局变量表首先可导出到某个目录形成可重复使用的 Excel 表格文件，在需要使用这些全局变量表时，使用导入功能可将 Excel 表格文件导入到当前工程中作为标准的全局变量表使用。

● 全局变量表导出

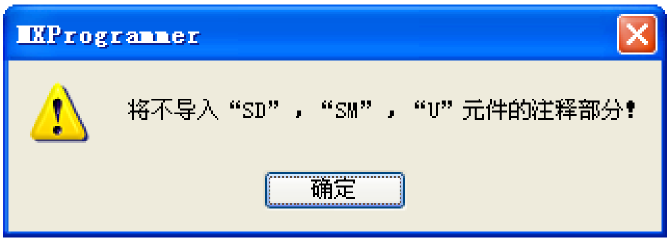
全局变量表编写完成后，选择“工程”菜单下的“全局变量表导出 (X)…”菜单项，将全局变量表保存到某个目录之下，导出的全局变量表名称不能为默认的文件名“*.xls”，对话框如下图所示：



导出的全局变量表如下图所示：

● 全局变量表导入

选择“工程”菜单下的“全局变量表导入 (W)…”菜单项，选中要导入的全局变量表，点击“打开”，由于系统无法导入“SD”，“SM”，“U”元件，会提示如下图所示对话框：



点击“确定”，即可以将需要的全局变量表导入到当前工程中。

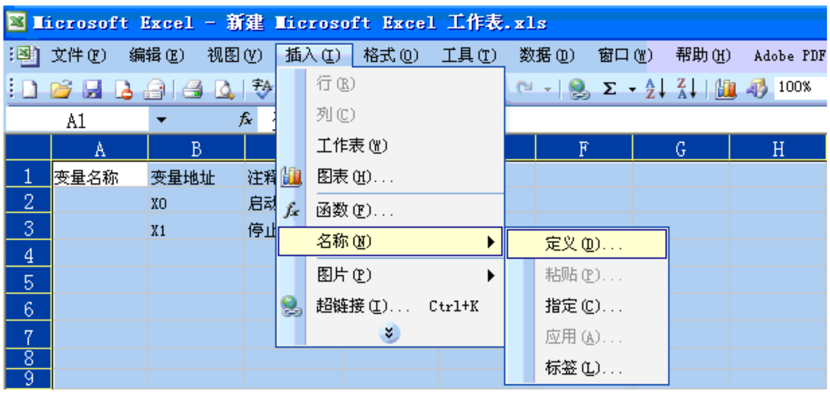
当导入不是通过 MXProgrammer 导出的全局变量表的 Excel 文件时需要经过以下步骤：

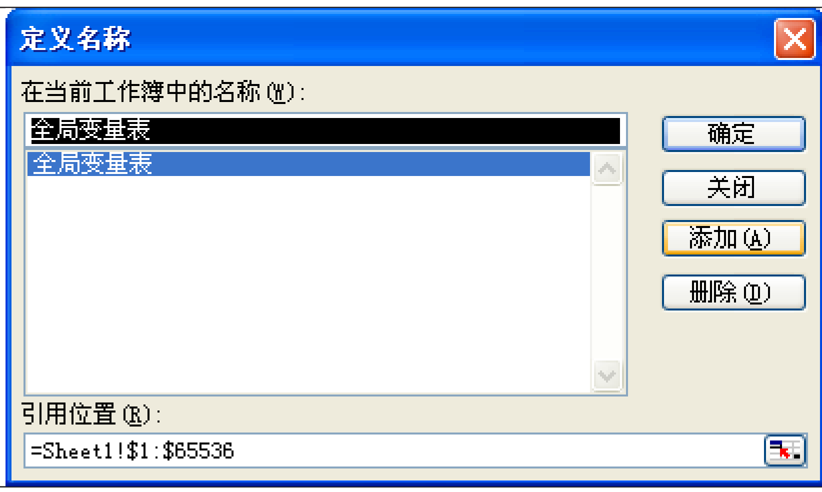
（1）首先打开全局变量表的 Excel 表格，添加表头“变量名称”、“变量地址”、“注释”，点击表格的左上端将整个 Excel 表格选中。

（2）选中后点击菜单中的“插入→名称→定义”，在定义名称对话框中输入“全局变量表”，点击添加，然后确定。

（3）点击“保存”，关闭，导入即可。

过程见下图：





☞ 注意

导入的全局变量表会将原有的全局变量表覆盖，若只修改部分元件的注释，请使用复制、粘贴功能。

2.2.4 特殊模块的 BFM 缓存区

特殊模块的地址不需要设置，PLC 主模块在上电时能自动识别和编址。

模拟量扩展模块是一类特殊模块，包括了模拟量输入模块和模拟量输出模块两大类。

这两类特殊模块通道特性、零点、最大数字量等配置数据在出厂都设有缺省值，一般无须修改；在必要的时候也可以由用户对这些参数进行修改，以满足用户对信号类型的选择、量程的缩放、误差的修正等方面的实际需要。

● 模拟量输入模块

模数转换模块与主模块之间通过 BFM 缓存区交换信息。主模块通过 FROM/TO 指令访问扩展模块内寄存器的 BFM 单元，也可使用“MOV ×× Un.b”访问扩展模块内寄存器的 BFM 单元。

主模块运行用户程序时，TO 指令将数据写入特殊模块 BFM 缓存区的相关缓存器，从而修改模数转换模块的出厂配置。可以修改的配置数据包括：零点数字量、最大数字量、输入通道信号特性、输入通道启用标志位等。

主模块通过 FROM 命令读取模数转换的 BFM 缓存区数据，包括模数转换结果和其他信息。

● 模拟量输出模块

数模转换模块与主模块之间通过 BFM 缓存区交换信息。主模块通过 FROM/TO 指令访问扩展模块内寄存器的 BFM 单元，也可使用“MOV ×× Un.b”访问扩展模块内寄存器的 BFM 单元。

主模块运行用户程序时，TO 指令将配置数据写入数模扩展模块 BFM 缓存区的相关缓存器，从而修改数模转换模块的出厂配置。可以修改的配置数据包括：零点数字量、最大数字量、输出通道信号特性、输出通道启用标志等。

主模块通过 FROM 指令读取数模转换的 BFM 缓存区的数据，通过 TO 指令将需要进行数模转换的数据写入 BFM 缓存区中的输出缓存器。

有关 TO/FROM 指令的说明，请参阅本手册第六章　应用指令说明；有关各种特殊模块 BFM 寄存器及其详细的配置方法，请参阅第十二章 特殊功能模块使用指南。

2.3 运行方式和状态控制

2.3.1 系统状态的概念

系统状态即主模块的工作状态，有两种：运行（RUN）状态和停止（STOP）状态。

运行（RUN）状态：当 PLC 工作在 RUN 状态时，系统会完整地执行循环过程（自诊断、与外围设备通讯、输入采样、执行用户程序、输出刷新）。

停止（STOP）状态：当 PLC 工作在 STOP 状态时，系统不执行用户程序，但系统会执行循环过程中的其它操作。

2.3.2 系统状态的切换

1．由停止状态切换到运行状态（STOP → RUN）

（1）主模块的选择开关切换到 ON 档位，系统自动进入运行状态。

（2）在系统组态中设置为输入点开机模式情况下，通过指定的端子由 OFF → ON 的切换使得系统进入运行状态。

（3）在主模块的选择开关为 ON 档情况下，鼠标单击编程软件的运行按钮“”，则系统会进入运行状态。

2．由运行状态切换到停止状态（RUN → STOP）

（1）主模块的选择开关切换到 OFF 档位，系统自动进入停止状态。

（2）在主模块的选择开关为 ON 档情况下，鼠标单击编程软件的停止按钮“”，则系统会进入停止状态。

（3）在运行状态下，当用户中的 STOP 指令被执行，则系统会进入停止状态。

（4）系统检测到有严重错误发生时，自动进入停止状态。

2.3.3 停止状态下输出点状态设定

在停止状态下输出点的设定模式有如下三种。

1．禁止输出模式：停止状态下，所有的输出点为 OFF。

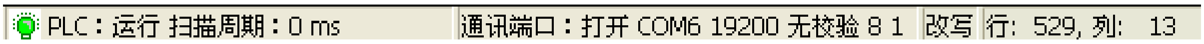
2．冻结输出模式：停止状态下，所有的输出点保持停止前的状态。

3．组态输出模式：停止状态下，用户可以根据需要设置输出点的状态。

2.3.4 系统状态的监控

1．可以通过主模块的运行（RUN）指示灯来监控系统的运行状态，指示灯闪烁说明系统已进入运行状态。

2. 在使用编程软件监控主模块的情况下，可通过查看软件右下角的“状态栏”判断 PLC 的状态，如下图所示：



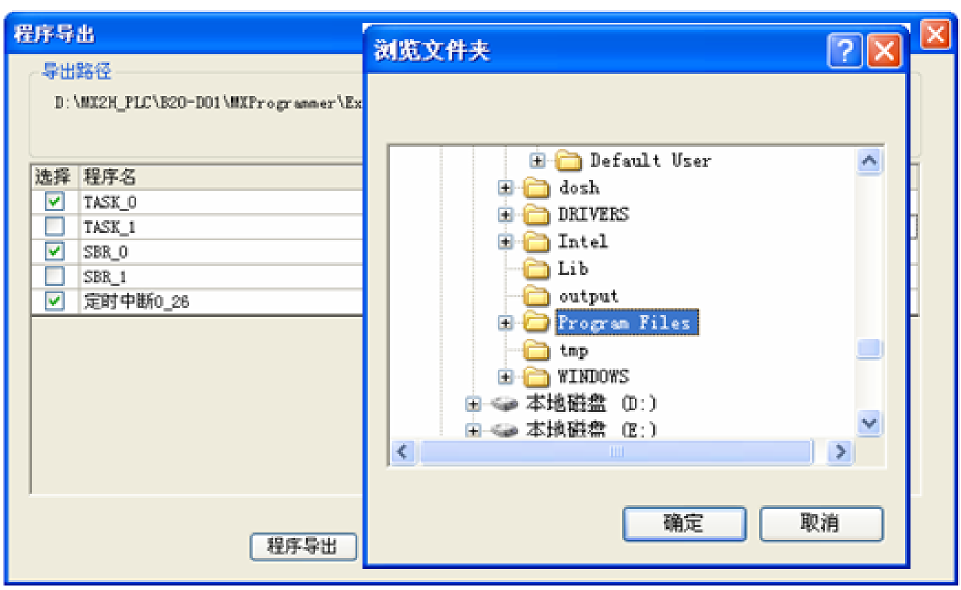
2.4 程序文件的导入导出

软件提供了程序文件的导入导出功能，对于某个通用功能，可以在程序（包括主程序、子程序、中断服务程序）编写完成后，通过导入、导出功能在不同的工程中多次使用，避免相同功能的程序重复编写。MXProgrammer 使用程序库文件的方式来管理这些程序，通用的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（MXP 文件），在需要这些程序文件的工程中，可以使用导入功能将程序库文件导入到当前工程中作为标准的程序使用。

● 程序文件导出

程序编写完成后，选择“工程”菜单下的“程序文件导出（U）”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且可以编辑“导出程序名”和“程序说明”。程序导出对话框如下图所示：

选择要导出的程序，程序导出路径第一次使用时默认为软件安装路径下的 \ExportLib\ 目录下，用户可以通过点击“…”按钮来更换导出路径。



在要导出的程序名前打勾，然后点击“程序导出”按钮。程序成功导出后，该程序名前的对勾自动消失，否则仍然处于导出前的状态。

- **程序文件导入**

选择“工程”菜单下的“程序文件导入（I）”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，如果所指定的程序名不合法或者与工程中已有的程序重名时，该程序导入不能成功，并且当导入程序名不合法时，在列表中用红颜色加下划线来提示。选择好程序后，点击“程序导入”按钮，程序导入到工程。

2.5 系统调试

2.5.1 编译

程序编辑完毕，点击工具栏上的“ ”或“ ”按钮，进行单页编译或全部编译，在“信息输出窗口”中会显示编译结果，若有错误，编译结果会显示详细的错误信息，点击错误信息可以追踪到程序中出错处。编译同时生成一个编译结果（UCD 文件），绑定了 PLC 序列号的 UCD 文件可用于保密下载。

☞ 注意

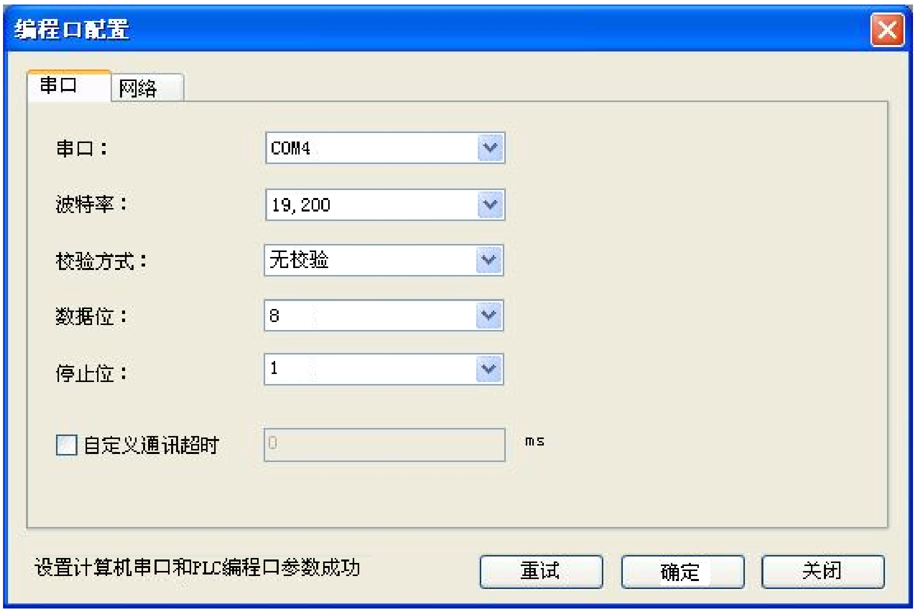
被动引用功能在编译之后有效，由于某些指令的被动引用情况由寄存器中的数据决定，而寄存器中的数据在编译后是无法得知的，因此这部分被动引用不会在全局变量表和元件使用表中显示，使用时请注意。例如 MODBUS 指令。

2.5.2 编程口配置

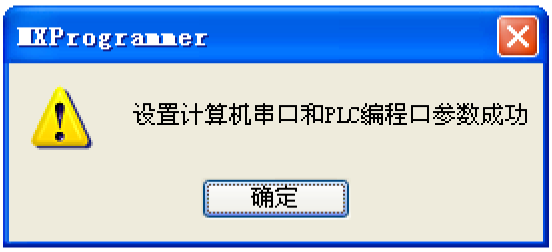
程序编译完毕后，便可以将程序下载到 PLC 中。在程序下载之前，需要先进行编程口配置，编程口配置包括串口配置（通过 PORT0 口与 PLC 通讯）和以太网口配置（通过以太网卡用网络与 PLC 通讯）。

1. 串口配置

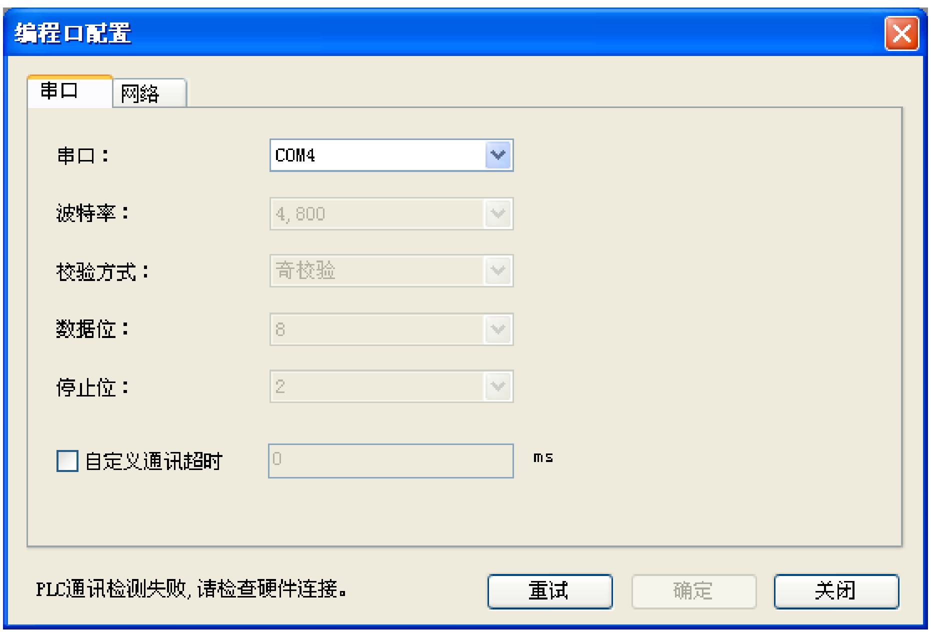
点击工具栏上的编程口配置快捷图标“ ”，系统默认进行串口配置，若配置成功，则出现如下所示对话框。



点击“确认”后，又会出现如下图所示对话框，再点击“确定”即可配置完成。



若编程口配置失败时，会进行如下对话框所示的提示：



☞ 注意

若编程口配置失败，经过检查没有硬件错误时，此时应将 PLC 的开关打到“OFF”档，然后再进行编程口配置，配置成功后，将 PLC 进行格式化，此后便可以对 PLC 进行正常操作了。

2. 以太网口配置

当通过以太网来控制 PLC 时，在开始与 PLC 硬件进行通讯前，需要建立和 PLC 以太网口的通讯连接，分为局域网和广域网两种方式。

（1）局域网

启动 MXProgrammer 应用程序，选择“工具→编程口配置”菜单或点击工具栏的编程口配置按钮“ ”，会弹出“编程口配置”的窗口，选择“网络”选项卡中的“连接到当前网络中的一台设备”，此时将显示局域网中所有设备的“类型，序列号，IP 地址，端口”，选择要连接的 PLC，点击确定，出现“连接成功”后，就可以对 PLC 进行下载、上载、监控、运行等操作。如下图所示。

（2）广域网

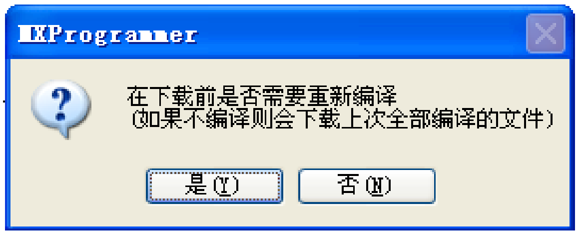
点击“编程口配置”，选择“网络”选项卡中的“我要连接到一台远程的设备”，此选项可以同时接受局域网和广域网的连接（但设备必须能够连接到广域网）。

在“请输入设备的产品序列号”文本框中，输入 PLC 的序列号，点击确定，出现“连接成功”后，就可以对 PLC 进行下载、上载、监控、运行等操作。如下图所示。

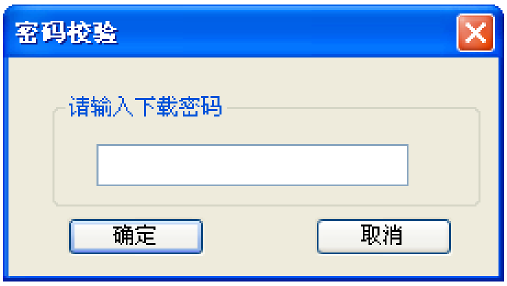
2.5.3 工程下载、监控和上载

1．下载

当 PLC 处于停止状态时，可通过编程软件将工程下载到 PLC 中，工程包括应用程序及系统组态、用户数据块，下载时，如果上次全部编译后程序发生了改变，则系统会有如下提示：



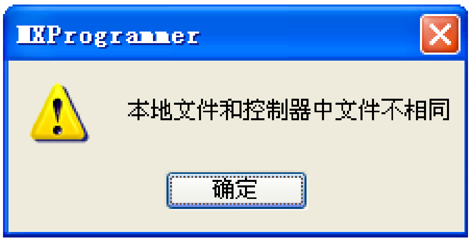
下载时，如果 PLC 中设有下载密码，并且在启动软件后没有输入过下载密码，软件会弹出“请输入下载密码”窗口要求输入密码。密码输入校验正确后开始下载，密码错误则提示重新输入密码，点击取消按钮退出下载。



2．监控

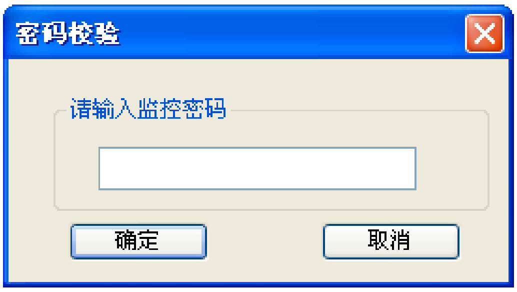
当 PLC 处于运行状态时，可通过编程软件监控程序的运行状态。

监控时，系统会自动检测本地文件与控制器中的文件是否相同，如果不同系统会弹出“本地文件和控制器中文件不相同”的对话框。



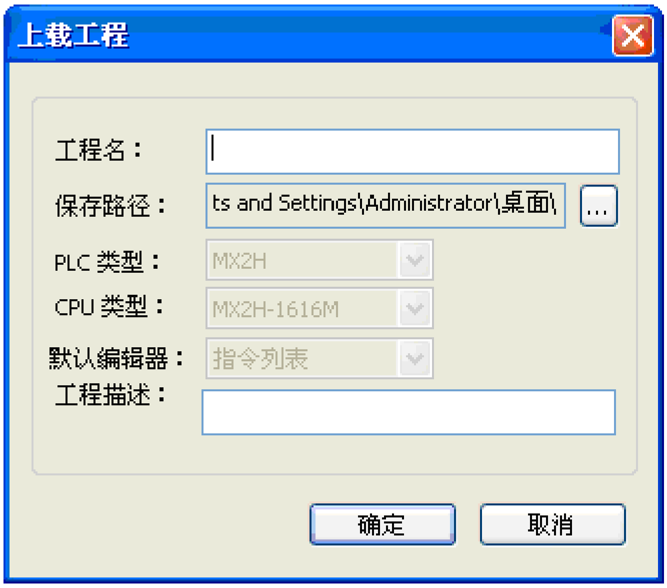
如果在“工具菜单→系统选项”中选择了“监控前不编译”，当本地文件与控制器中的文件不相同时，监控不会弹出“本地文件和控制器中文件不相同”的对话框，可以直接监控。

监控时，如果设有监控密码，并且在启动软件后没有输入过监控密码，软件会弹出“请输入监控密码”窗口要求输入密码。密码输入校验正确后开始监控，密码错误则提示重新输入密码。

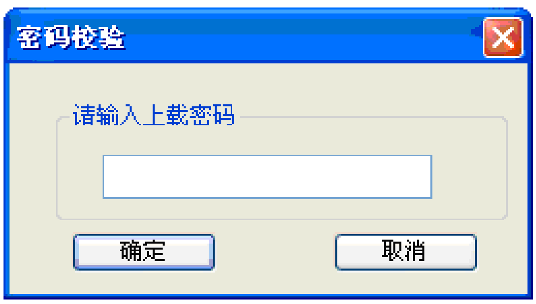


3．上载

上载功能用于把 PLC 中的应用程序及系统组态、用户数据块上载到计算机中，并且以新建工程的形式保存，在电池备份数据有效的情况下，选择上载，相应的用户辅助信息文件也会被捆绑上载。上载对话框如下图所示：



上载时，如果设有上载密码，并且在启动软件后没有输入过上载密码，软件会弹出上载密码输入窗口要求输入密码。密码输入校验正确后开始上载，密码错误则提示重新输入密码。点击取消按钮退出上载。如果在 PLC 中的工程设为禁止上载模式，则 PLC 中的此工程将无法上载。



2.5.4 报错机制

报错分两种：系统错误和用户程序运行错误。系统错误是系统异常运行导致的错误，用户运行错误是用户程序异常导致的错误。所有的错误都统一编号，每个错误编号都代表一种错误，详见附录一 系统错误代码表。

1．系统报错的机制

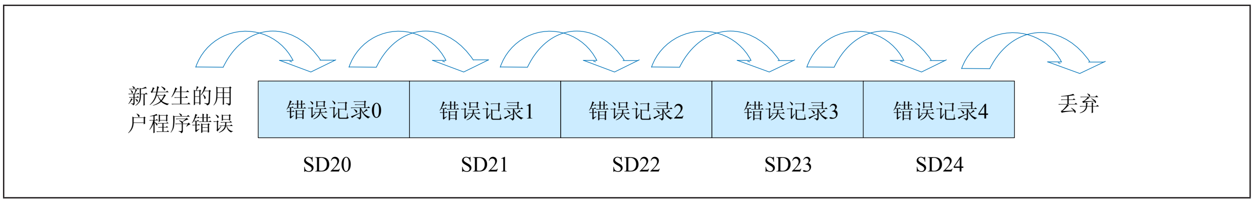
检测到系统有错误发生时，相应的系统错误编号会被写入特殊数据寄存器 SD3 中，同时特殊辅助继电器 SM3 会被置位，当多个系统错误同时发生时，系统按错误的严重程度，在 SD3 指示严重程度较高的错误。严重的系统错误将导致用户程序停止运行，并会引起主模块的 ERROR 指示灯长亮。

2．用户程序运行错误的报错机制

当检测到用户程序运行错误时，相应的错误编号会被写入特殊数据寄存器 SD20 中，同时特殊辅助继电器 SM20 会被置位。当正确执行下一条指令时，SM20 会被清零，但 SD20 中仍然记录着上一次的错误编号。

系统以错误记录栈方式记录先后发生的用户程序运行错误，特殊数据寄存器 SD20 ～ SD24 构成了一个大小为 5 的错误记录栈，SD20 ～ SD24 中记录了最近 5 次的用户程序运行错误编号。

错误编码的压栈过程如下：



3．在线检查报错信息

在与 PLC 通讯的情况下，MXProgrammer 编程软件可以读取当前的 PLC 信息，其中包括上述系统错误和用户程序运行错误。

2.5.5 清除与格式化

清除操作包括：PLC 元件值清除、PLC 错误信息清除、PLC 程序清除及 PLC 数据块清除。

格式化是指将 PLC 内全部数据和程序清除。

清除操作和格式化操作都需要 PLC 处于停止状态，这些功能都需谨慎使用，为防止误操作，操作时软件会提示确认窗口。

PLC 元件值清除：清除 PLC 中所有的元件值。它可能导致 PLC 运行不正确或中间工作数据丢失。

PLC 错误信息清除：清除 PLC 的系统错误和用户程序运行错误。

PLC 程序清除：清除 PLC 中所有的用户程序。

PLC 数据块清除：清除 PLC 中所有数据块设置。

PLC 格式化：格式化 PLC 中所有数据，包括清除用户程序、恢复默认配置、清除数据块。本操作将会丢失 PLC 中已经下载和设置的全部数据。

2.5.6 PLC 信息在线查询

1．PLC 时钟

PLC 时钟用于显示和设置 PLC 的当前时间，PLC 时钟对话框如下所示：

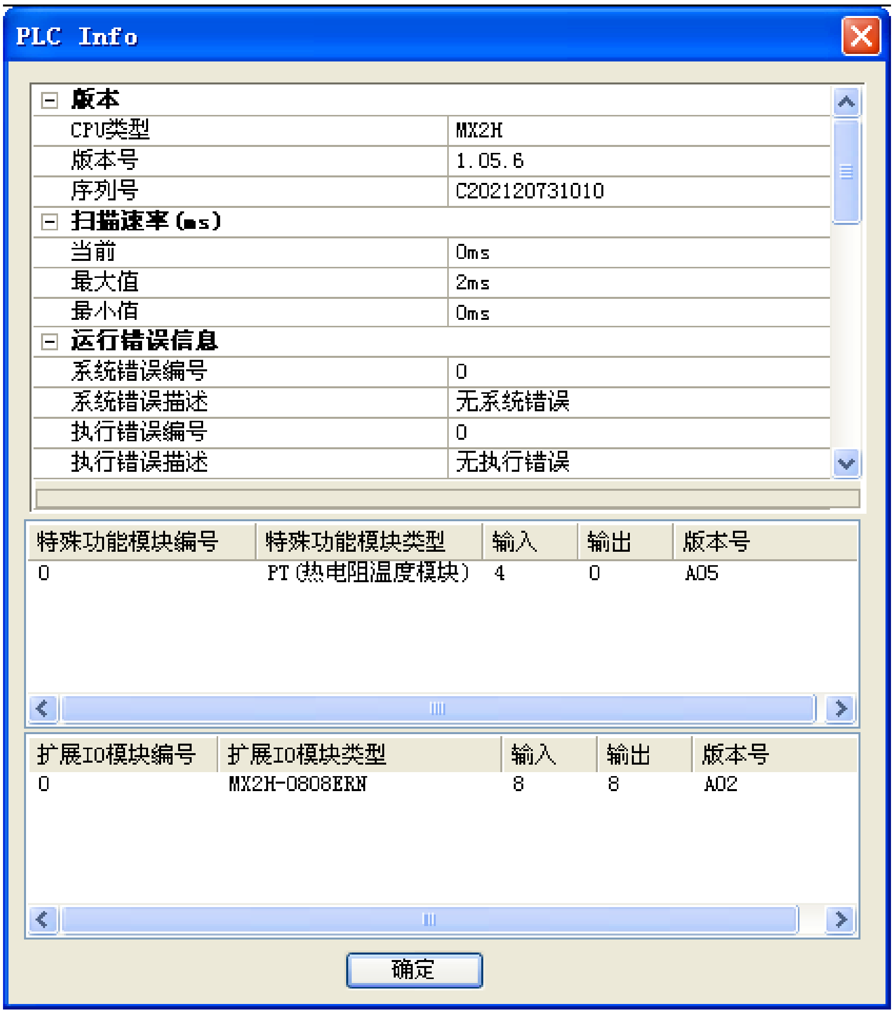
进入 PLC 时钟设置界面时，显示 PLC 当前时间。点击“同步系统时间”，界面显示 PC 机当前时间。点击“设置时间”，将界面显示时间设置到 PLC 中。

☞ 注意

如果系统设置了下载密码，必须在下载密码正确的情况下才能修改“PLC 时钟”。

2．PLC 信息

PLC 信息是从 PLC 中读取的各种运行数据和其它重要信息。在 MXProgrammer 编程软件中查看 PLC 信息可用鼠标单击菜单栏上的“PLC”选项中的“PLC 信息”选项，也可直接单击工具栏中的快捷图标“PLC”，PLC 信息对话框如下：



2.5.7 元件值写入和强制、元件监控表

1．元件值写入和强制

在调试过程中，需要手动改变某些元件的数值时，可以使用元件写入值和强制功能。写入与强制的不同之处在于：写入的元件值仅一次有效，可以被程序改变，但是强制的元件在每个扫描周期开始时向其中赋一次值，在程序执行过程中，也有可能被改变，但下一个扫描周期又被强制赋值，直至取消强制。

在监控状态下，可以在监控表中使用写入或强制功能，也可在编程区选中元件后单击鼠标右键使用写入或强制功能。对于不再需要强制的元件，可在编程区选中元件单击鼠标右键使用取消强制功能。

在监控表中，通过右键菜单中的“写入所有元件”，可以将选中的连续多个新数值写入。

在监控表中元件值写入或强制如下所示：


TASK_0 | 监控表0

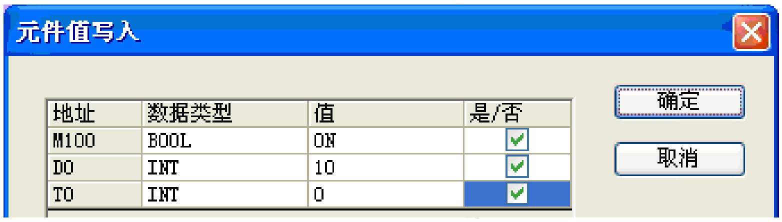
	元件名称	数据类型	显示格式	当前值	新数值	元件注释
1	X0	BOOL	二进制	ON	ON	
2	M10	BOOL	二进制	OFF	OFF	
3	M11	BOOL	二进制	ON	ON	
4	M12	BOOL	二进制	OFF	OFF	
5	D0	WORD	十进制	10	10	
6	C0	WORD	十进制	20	20	
7	C1	WORD	十进制	100	100	
8		WORD	十进制			

在梯形图界面下监控时，选中元件单击鼠标右键弹出的下拉菜单如下图所示：

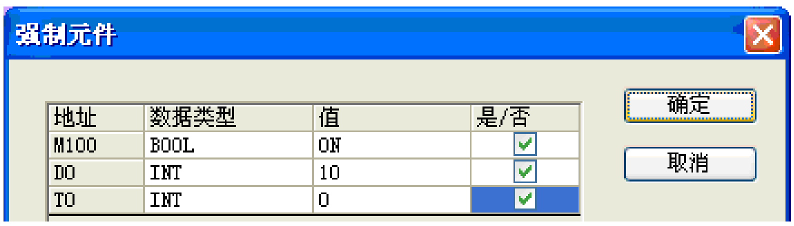
（1）对于位元件，可直接点击“写入”，写入 ON 或 OFF；也可通过点击“写入 ON”、“写入 OFF”对位元件写入；还可以点击“写入上升沿”、“写入下降沿”直接对位元件写入沿；同时还支持位元件的强制功能。

（2） 对于字元件，可通过点击“写入”，写入元件值，同时还支持字元件的强制功能。

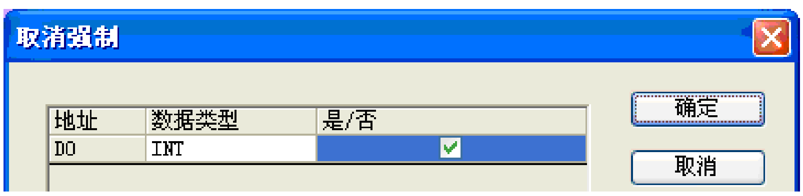
位元件和字元件的元件值写入对话框如下：



位元件和字元件的强制对话框如下：



选中元件单击鼠标右键弹出的取消强制对话框如下：



2．元件监控表

在调试过程中可以使用监控表监控元件的数值，调试程序时允许将欲监控的元件置入元件监控表中以便在将程序下载至 PLC 之后监控其状态。

在梯形图界面下，监控程序时选中元件点击鼠标右键选择“添加到监控表”选项，元件自动被加入元件监控表中。元件监控表在监控模式下自动刷新元件值。

元件监控表有两种模式：编辑模式和监控模式。

（1）编辑模式下可以执行所有的编辑功能，不能执行监控功能。

（2）在监控模式下，可以同时执行监控功能和编辑功能。编辑功能包括查找、替换、定位、剪切、复制、粘贴、插入行、删除行、添加行、排序功能（可以使用鼠标左键单击“元件名称”、“数据类型”、“显示格式”、“当前值”、“新数值”、“元件注释”进行相应的排序）、复制填充（在需要将多个元件的“数据类型”、“显示格式”及“新数值”设为相同时，可使用“复制填充”）、序列填充（在需要查看多个相邻的元件状态时，可使用“序列填充”）等功能，监控功能包括元件写入（可批量写入）、元件强制、取消元件强制、读取所有强制等功能。

元件监控表包括元件名称、数据类型、显示格式、当前值、新数值及元件注释。如下图所示：

TASK_0 | 监控表0

	元件名称	数据类型	显示格式	当前值	新数值	元件注释
1	SD3	DWORD	二进制	2#10000000000000000000		系统错误代码
2	SM3	BOOL	二进制	OFF		系统错误
3	SM66	BOOL	二进制	OFF		定时中断0使能标志
4	SD66	WORD	二进制	2#0		定时中断0周期设置值
5	M100	BOOL	二进制	ON	ON	
6	M101	BOOL	二进制	OFF		
7	M102	BOOL	二进制	ON	ON	
8	D100	WORD	十进制	255	0	
9	D101	WORD	十进制	40000	0	
10	D102	INT	十进制	-50	0	
11	D200	REAL	十进制	0.125640	0.000000	

3．元件监控图

MXProgrammer 提供了监控图（示波器功能），可以将元件值随时间的变化在监控图中显示出来，方便调试查看数据的变化细节。最多可添加 8 条曲线。

CH1 为未加抗干扰措施的波形， CH2 为加抗干扰措施后的波形。

- 支持绝对时间轴，便于曲线的存档及分析；
- 鼠标悬停实时显示采样点当前采样值、最小值、最大值及平均值；
- 可自行设置采样间隔及采样时间；
- 可跟踪 / 暂停曲线捕获，便于观察曲线；
- 支持导入曲线功能；
- 支持监控图的导入、导出功能；
- 支持单曲线、多曲线操作；
- 支持各种曲线缩放、平移功能，便于用户灵活操作；
- 支持曲线的同比显示、全景显示及平铺显示功能，适用于不同情况下对曲线进行分析；
- 支持多组监控曲线；

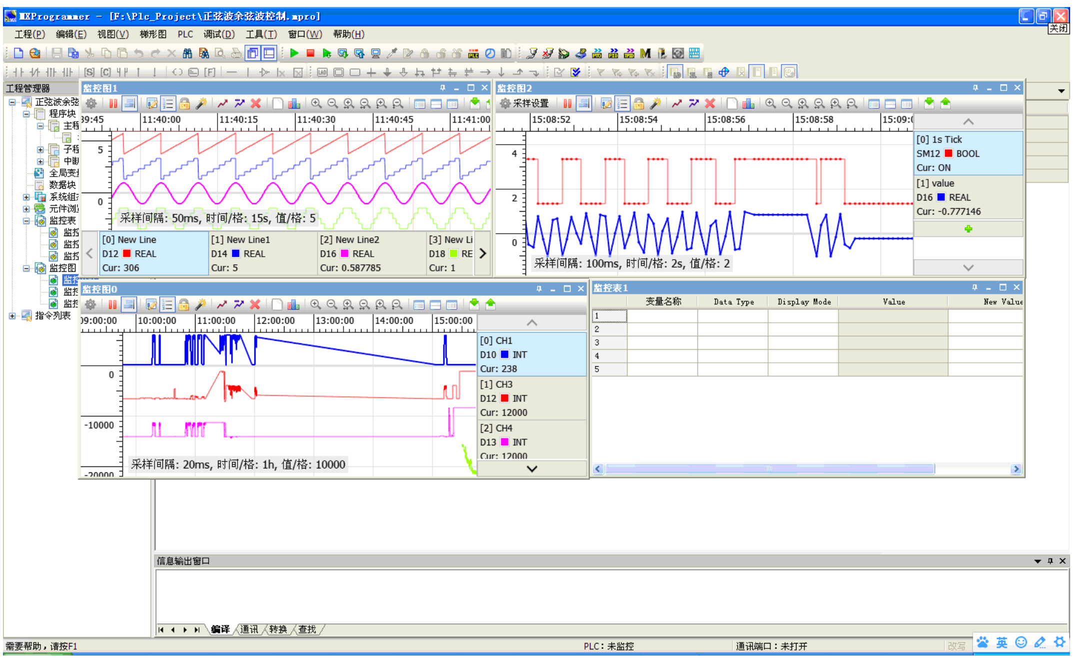
2.5.8 多窗口浮动

多窗口浮动功能支持监控表、监控图、可视化面板、信息输出窗口以及工程管理器多个窗口，以便其在 MXProgrammer 窗口框架中具有位置或大小，或作为独立于 MXProgrammer 的单独窗口而浮动。 可以停靠多个窗口，使其共同浮动在 MXProgrammer 上方或同一个组合窗口中，也可以隐藏或最小化相应的窗口。

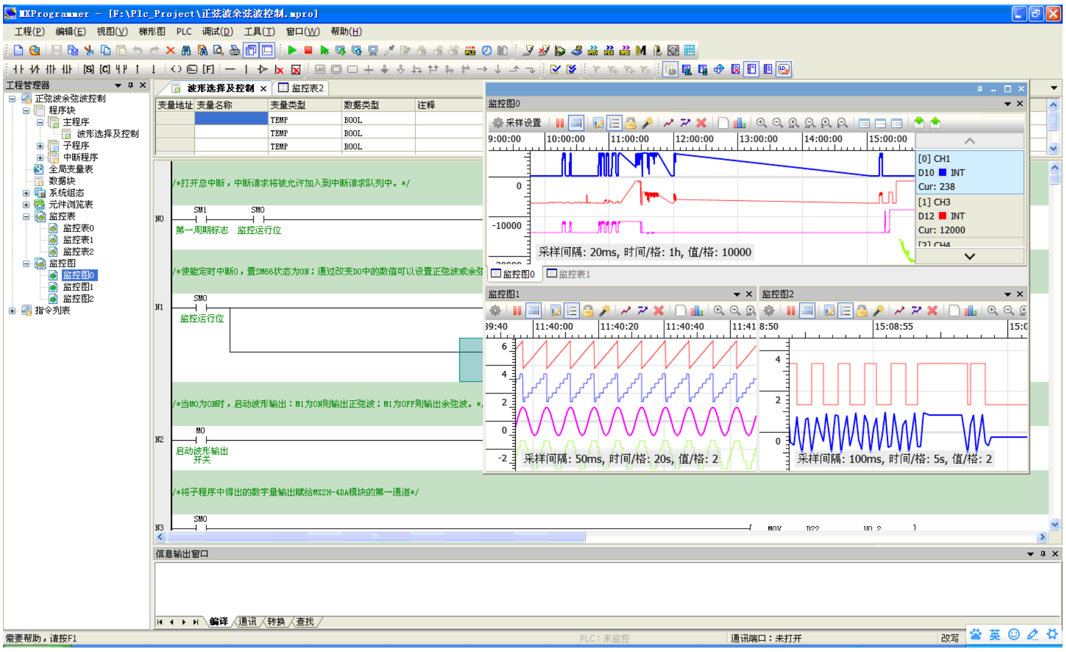
可以通过以下方式排列窗口：

- 将文档窗口固定到选项卡左侧。
- 将窗口以选项卡形式停靠到编辑框。
- 将工具窗口停靠到 MXProgrammer 中的框架边缘。
- 在 MXProgrammer 的上方或外部浮动文档窗口或工具窗口。
- 沿 MXProgrammer 的边缘隐藏工具窗口。
- 在不同监视器上显示窗口。
- 窗口位置重置为默认布局或已保存的自定义布局。
- 可以通过拖动、使用右键单击待排列窗口的标题栏，来排列相应的窗口。

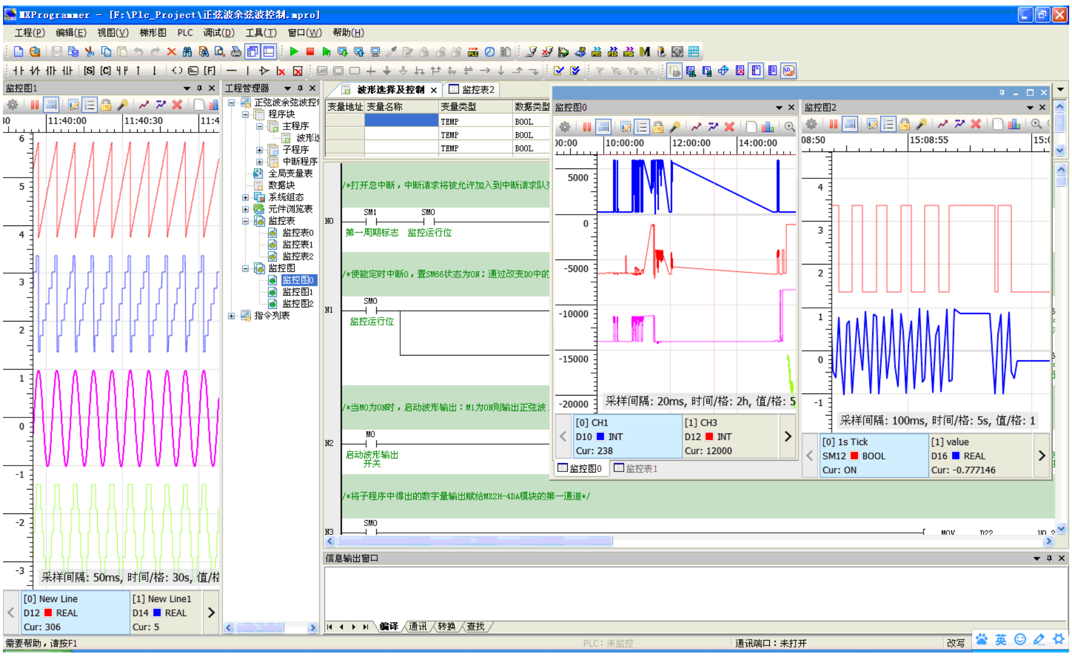
监控表与监控图支持窗口停靠功能后，用户可以在监控梯形图的同时监控多个监控表与监控图，可以同时查看、修改多组数据，便于调试。如下图：



示例 1



示例 2



示例 3

第三章 编程概念

本章详细介绍了 MX 系列 PLC 编程的相关内容，包括编程语言、程序要素、程序的块注释和变量的注释，子程序的编程和使用，最后介绍了指令的一些通用说明。

3.1 编程语言介绍

编程语言有三种：梯形图（LAD）、指令列表（IL）、顺序功能图（SFC）。

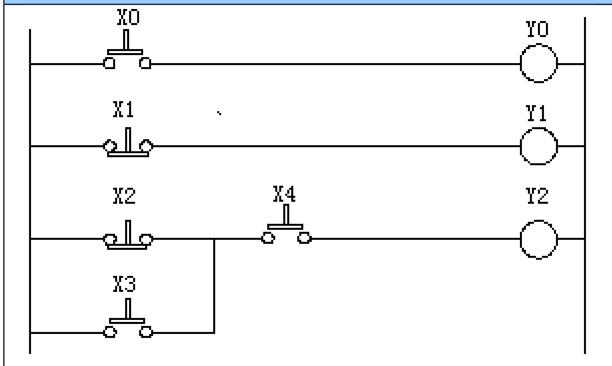
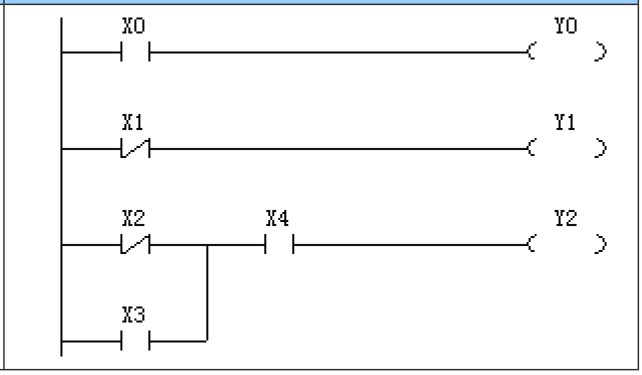
3.1.1 梯形图（LAD）

● **梯形图概念**

梯形图是一种历史最久、使用最广的编程语言，最初只有常开触点、常闭触点、输出线圈、定时器、计数器等基本器件（即配电盘），直到可编程控制器出现后，梯形图中可表示的器件，除上述以外，另增加了传统配电盘无法达成的应用指令，如加、减、乘及除等四则运算指令。

无论传统的梯形图或 PLC 梯形图其工作原理均相同，只是符号表示上传统的梯形图比较接近实体的符号表示，而 PLC 则采用较简明且易于计算机或报表上表示的符号表示。在梯形图逻辑方面可分为组合逻辑和顺序逻辑两种，如下所示：

1．组合逻辑

传统梯形图	PLC 梯形图
	

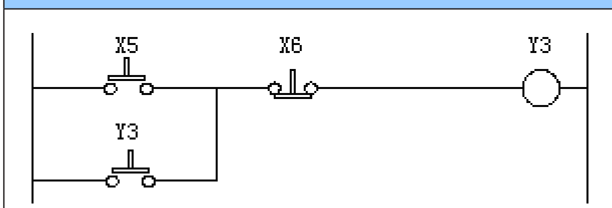
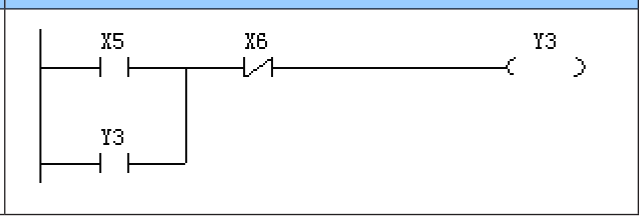
（1）使用常开开关 X0 控制输出线圈 Y0，在 X0 未动作时，其触点为开路状态，故 Y0 不得电；当 X0 动作时，其触点变为导通状态，故 Y0 得电。

（2）使用常闭开关 X1 控制输出线圈 Y1，在 X1 未动作时，其触点为导通状态，故 Y1 得电；当开关 X1 动作时，其触点反而变成开路状态，故 Y1 不得电。

（3）使用常闭开关 X2 和常开开关 X3、X4 多个输入触点组合逻辑输出，其输出 Y2 只有在 X2 不导通 X4 导通或 X3、X4 都导通时才会得电。

2．顺序逻辑

顺序逻辑为具有回授结构的回路，即将回路输出结果拉回当输入条件，如此在相同输入条件下，会因前次状态的不同而得到不同的输出结果。

传统梯形图	PLC 梯形图
	

各个输入开关的动作及输出结果如下表所示：

触点状态 / 动作顺序	X5 开关	X6 开关	Y3 状态
1	不动作	不动作	OFF
2	动作	不动作	ON
3	不动作	不动作	ON
4	不动作	动作	OFF
5	不动作	不动作	OFF

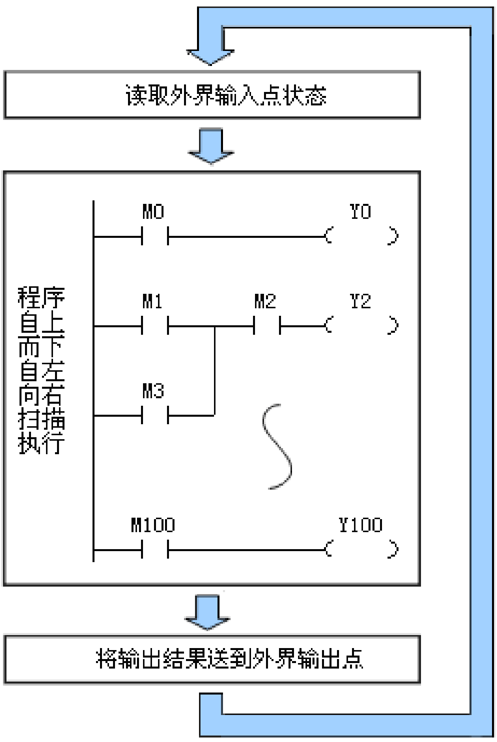
由上表可知，在不同动作顺序下，虽然输入状态完全一样，其输出结果也可能不一样，如表中的动作顺序 1 和 3，其输入 X5 和 X6 开关均不动作，在状态 1 的条件下 Y3 为 OFF，但在状态 3 的条件下 Y3 却为 ON。这种把输出状态拉回当输入条件而使回路具有顺序控制的效果是梯形图回路的主要特性。

● **传统梯形图与 PLC 梯形图的差异**

虽然传统梯形图和 PLC 梯形图的工作原理是完全一致的，但实际上 PLC 仅是利用微处理器来仿真传统梯形图的动作，即利用扫描的方式逐一地查看所有输入触点及输出线圈的状态，再将此等状态依梯形图的组态逻辑作运算得到和传统梯形图一样的输出结果，但因微处理器只有一个，只能逐一地查看梯形图程序，并依该程序及输入 / 输出的状态运算得到输出结果，再将结果送到输出接口，然后又重新读取输入状态→运算→输出，如此周而复始地循环执行上述动作，此一完整的循环动作所消耗的时间称之为扫描周期，其时间会随着程序的增大而加长，此扫描周期将造成 PLC 从输入检测到输出反应的延迟，延迟时间越长对控制所造成的误差越大，甚至造成无法胜任控制要求的情况，此时就必须选用扫描速度更快的 PLC，因此 PLC 的扫描速度是 PLC 的重要性能，随着微电脑技术的发展，现今的 PLC 在扫描速度上均有极大的改善。

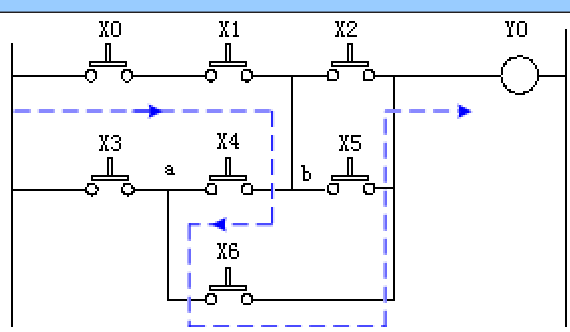
MX 系列 PLC 基本指令处理速度高达 0.084μs/ 步，综合指令处理速度为 1K 步 /ms。

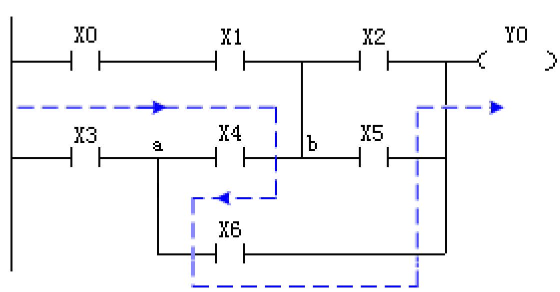
下图为 PLC 的梯形图程序扫描的示意图。

<table>
<tr>
<td>


</td>
<td>根据用户程序输出运算结果，内部状态会实时更新，但不会立即向外部输出（除设为立即刷新模式的输入输出点以外），程序周而复始的扫描执行。</td>
</tr>
</table>

除了上述扫描周期差异外，传统梯形图和 PLC 梯形图还有如下的“逆向回流”的差异，如下图传统梯形图中，若 X0、X1、X4、X6 为导通状态，其它为不导通状态，在传统梯形图中 Y0 会如虚线所示形成回路而闭合。但在 PLC 梯形图中，因运算梯形图程序是自上而下，自左向右地扫描，故在相同输入条件下，PLC 梯形图编辑工具会检查出梯形图错误。

<table>
<tr>
<th>传统梯形图的逆向回流</th>
<th>PLC 梯形图的逆向回流</th>
</tr>
<tr>
<td>


</td>
<td>


</td>
</tr>
</table>

● 梯形图的基本编程元素

梯形图根据电气（继电器）控制图的原理，抽象出几种基本编程元素：

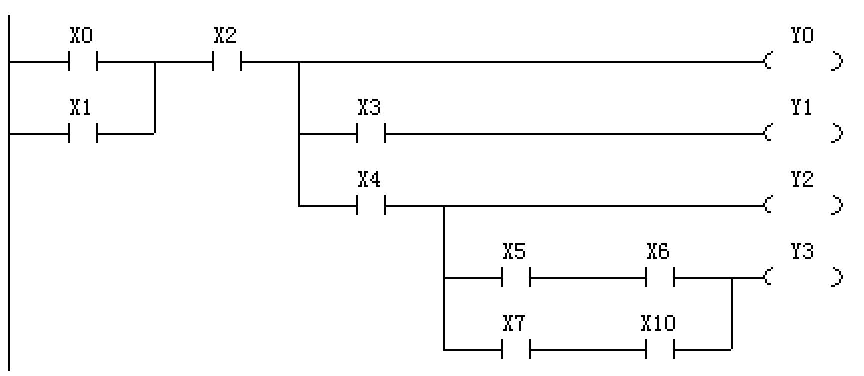
1. 左母线：对应于电气控制图中的控制母线，为控制回路提供控制电源。
2. 连接线（）：代表电气控制图的电气连接，用于导通彼此相连的其它元件。
3. 触点（、）：代表电气控制图中的输入接点，控制着回路中电流的通断和方向。触点的并联、串联的连接关系，实质上是代表了控制电路输入逻辑的运算关系，控制着电流的传递。
4. 线圈（）：代表电气控制图中的继电器输出。
5. 功能块（）：又称应用指令，对应部分电气控制图中完成特殊功能的执行机构或功能器件，功能块可以完成特定的控制功能或控制计算功能（例如数据传输、四则运算、计时器、计数器等）。

3.1.2 指令列表（IL）

指令列表是文本化的用户程序，是用户编写的指令序列集。

存储在 PLC 主模块中用于执行的用户程序，实际上是主模块可识别的指令序列，系统逐条执行序列中的每一条指令，实现用户程序的控制功能。

下图是梯形图转换成指令列表的示例：

梯形图	指令列表
	LD X0 OR X1 AND X2 MPS OUT Y0 AND X3 OUT Y1 MRD AND X4 OUT Y2 MPP LD X5 AND X6 LD X7 AND X10 ORB ANB OUT Y3

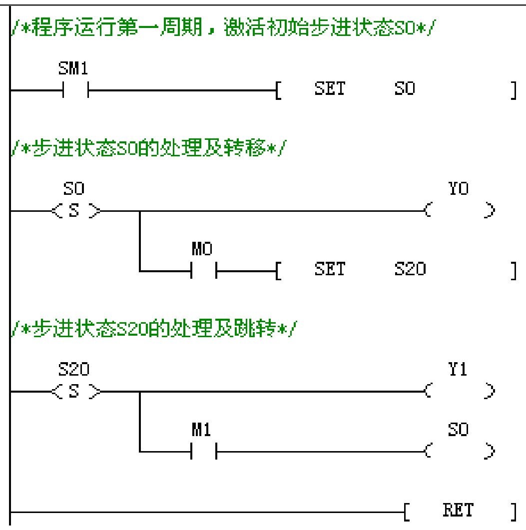
3.1.3 顺序功能图（SFC）

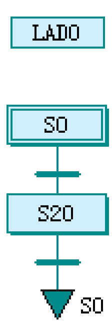
顺序功能图是一种图形化的用户程序设计语言，通常用于实现顺序控制功能。

顺序功能图可划分多个工序（处理步骤），并按一定工作顺序进行处理。

按顺序功能图设计的用户程序，程序结构与实际的顺序控制过程相符合，比较直观清晰。

下图是一个简单顺序功能图的示例：





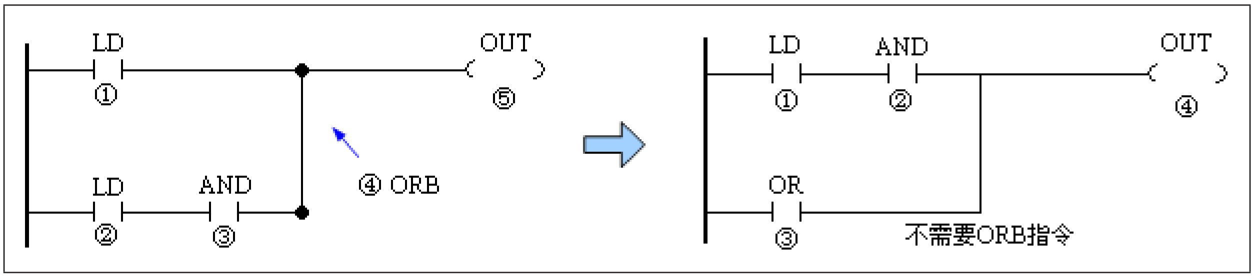
3.2 编程方面的基本注意事项

3.2.1 程序的步骤及执行顺序

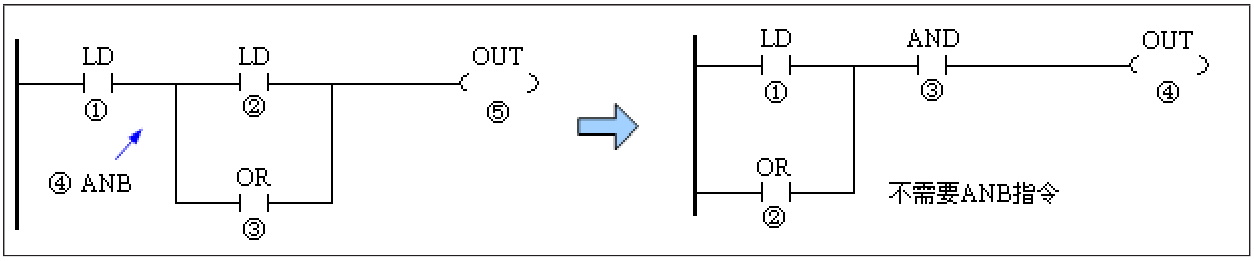
● **触点的构成和步**

即使是执行相同动作的梯形图，触点的构成方法不同，也能简化程序和节约步数。

1. 串联触点较多的梯形图写在上方比较好



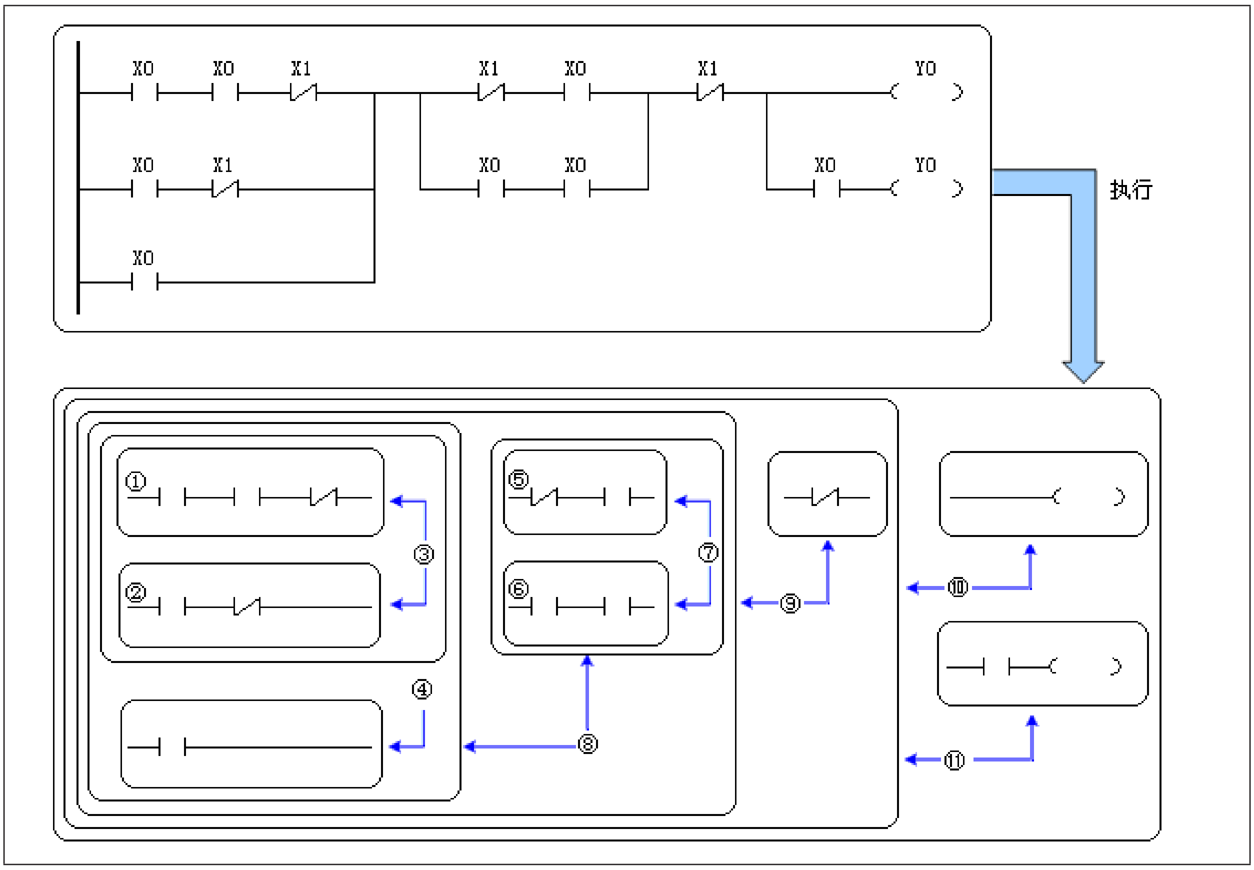
2. 并联触点较多的梯形图写在左方比较好



● **程序的执行及编程顺序**

程序是按照［从上至下］到［从左至右］的顺序进行处理的。

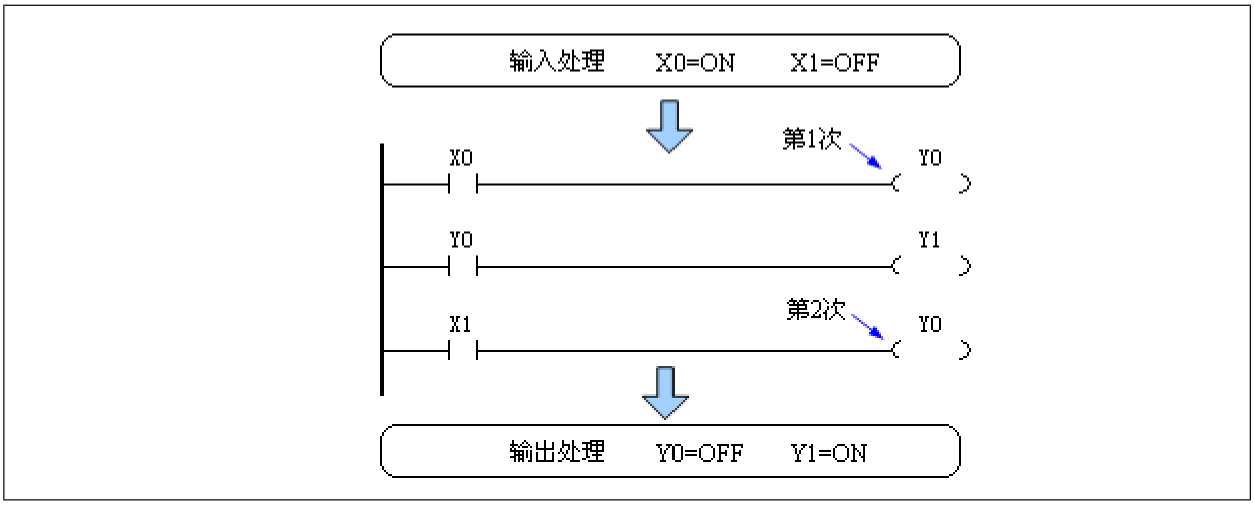
指令表也请按照这个顺序编码。



3.2.2 双重输出（双线圈）的动作及对策

- 双重输出的动作

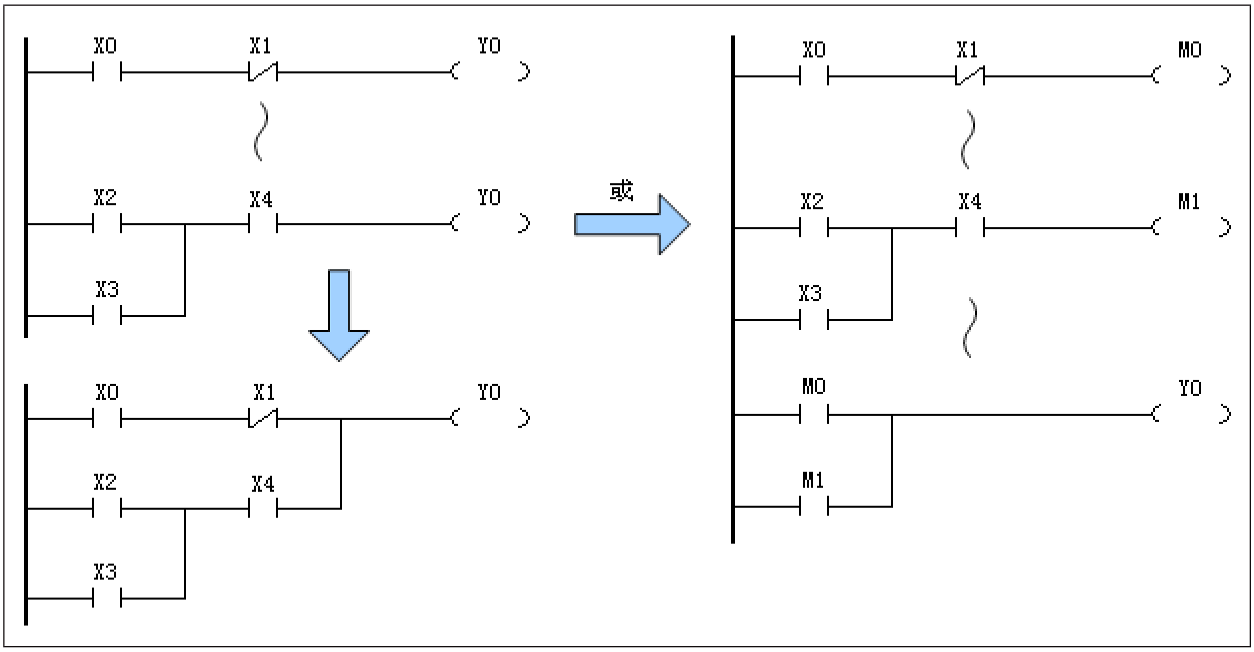
如果程序中执行线圈的双重输出 (双线圈)，则后面的线圈优先动作。如下图所示，同一线圈 Y0 在多个位置被使用。



例如 X0、X1 的最初状态为：X0=ON，X1=OFF。最初的 Y0 由于 X0 为 ON，所以其在映象存储区内为 ON，输出 Y1 也为 ON。但是，第 2 次的 Y0 由于输入 X1 为 OFF，所以其在映象存储区内被改写为 OFF。因此，实际的外部输出 Y0=OFF，Y1=ON。

- 双重输出的对策

双重输出 (双线圈)，并非违背了程序中的输入 (程序错误)，但是由于会使上述的动作变得复杂，因此请学习下面的例子后更改程序。

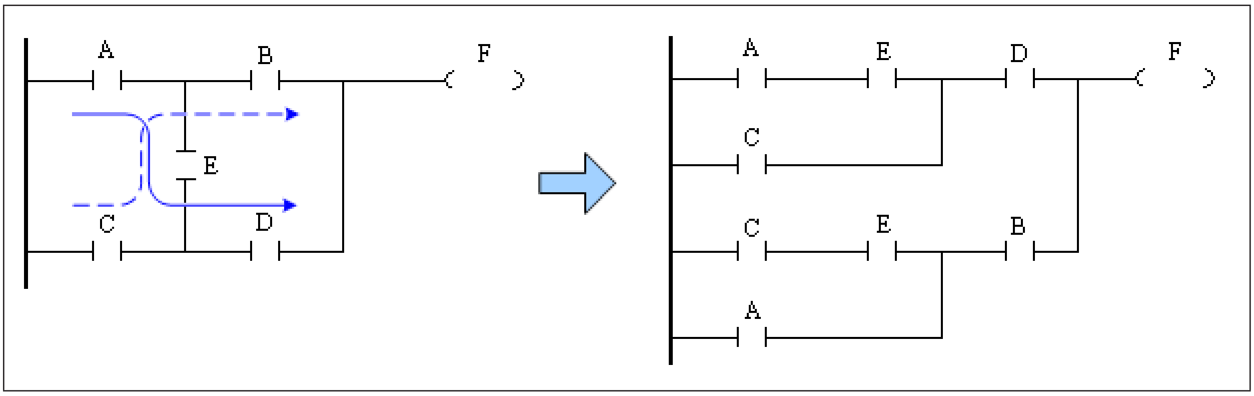


此外，还有其它编程方法，如使用 SET、RST 指令或跳转指令，以及使用步进梯形图指令，在各状态中对同一个输出线圈编程。使用步进梯形图指令的时候请注意：如果主程序中存在的输出线圈，在状态中也存在时，会被视为双重线圈。

3.2.3 不能编程的回路图及对策

- 桥式电路

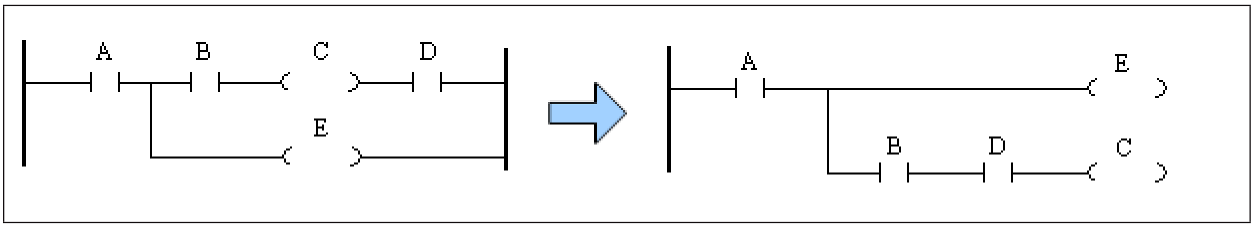
请按照如下右图所示，更改两个方向都有电流流过的回路。（将没有 D 时的回路和没有 B 时的回路进行并联的结果）



● **线圈连接的位置**

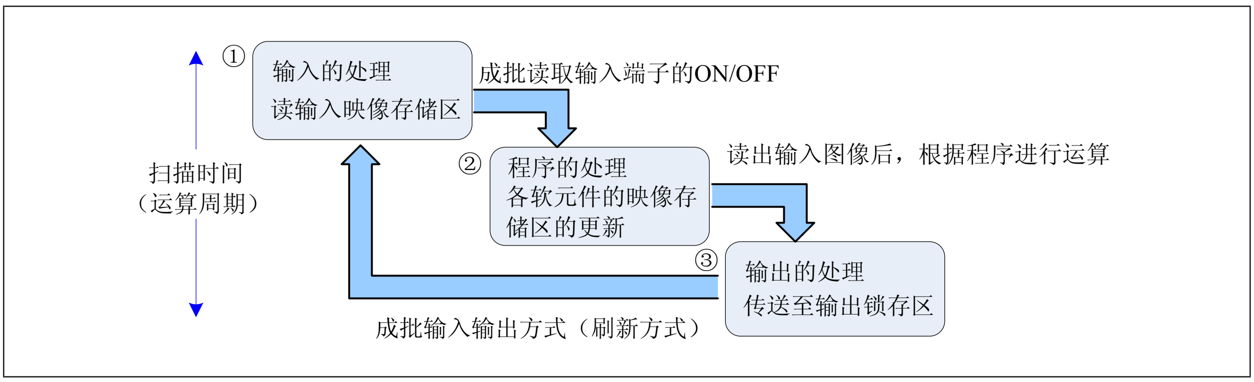
（1）线圈的右侧请勿写触点。

（2）建议触点之间的线圈放在前面编程。如触点 A 和 B 之间的线圈（E）放在程序前面，可以节省步数。



3.3 输入输出处理，响应延迟

MX 系列可编程控制器中，重复执行①～③进行输入输出处理。因此，在可编程控制器的控制中，除了输入滤波器和输出元器件的驱动时间以外，根据运算周期有时会出现响应延迟。



在上述的运算周期中，想要获取输入的最新信息时或者要将运算结果立即输出时，可以使用输入输出刷新指令。

3.4 用户程序

用户程序是用户所编写的程序，经过编译后形成可执行的指令序列，下载到 PLC 中，使 PLC 执行控制功能。

用户程序由主程序、子程序、中断服务子程序三类程序体（POU）构成。用户程序具有导入导出功能，具体内容请参考 *2.4* 程序文件的导入导出。

3.4.1 主程序（TASK）

主程序是用户程序的主体和框架。系统处于运行状态时，程序被循环执行。为了便于大型程序模块编程，MX 系列 PLC 允许有多个主程序块。

3.4.2 子程序（SBR）

子程序是一段结构和功能上独立的、可以被其它程序体调用的用户程序，通常具有调用操作数接口，只有在被调用时执行。一个用户程序可以没有子程序，也可包含一个或多个子程序。

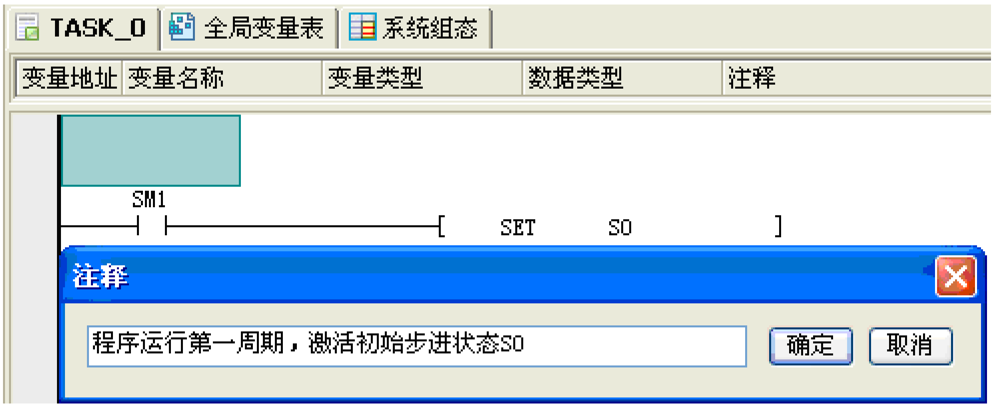
3.4.3 中断服务子程序（INT）

中断服务子程序是处理特定中断事件的用户程序。某个特定的中断事件总是对应于特定的中断服务子程序。
中断事件发生时，一个正常的扫描周期将被打断，用户程序自动跳转到中断程序中执行，直至中断程序执行完毕，系统才会返回到正常的扫描周期执行。一个用户程序可以没有中断服务子程序，也可包含一个或多个中断服务子程序。

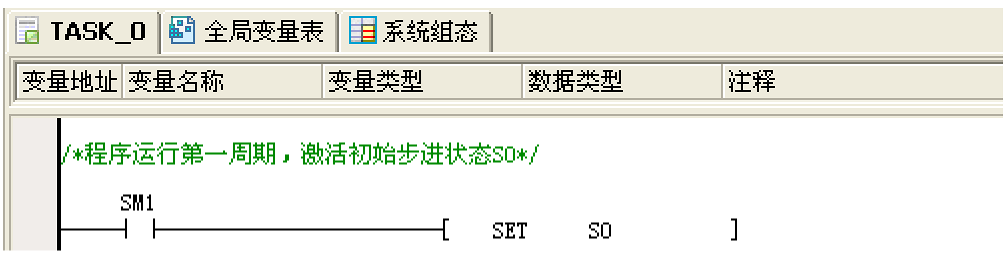
3.5 程序的块注释和变量的注释

3.5.1 块注释

在编程时，可以在程序中添加块注释，每个块注释都会占用一整行的空间。需要输入块注释时，首先选择一个空行，在输入法为英文状态下时使用键盘输入“/”，系统会弹出“块注释”对话框。在弹出的块注释对话框中输入注释文字并确认，如下图所示：



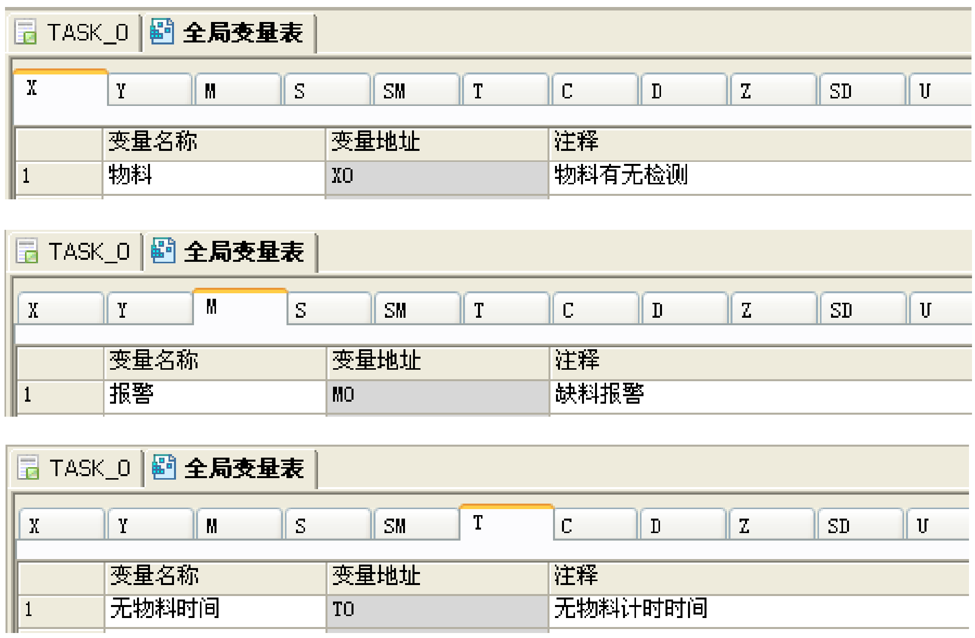
软件会自动在所输入的文字两侧分别加上“/*”和“*/”，如下图所示：



由于块注释会占用整行空间，因此如果某行已经存在其他元件，则此行不能再输入块注释；同理，已经被块注释占用的行，也不能再输入其他任何元件。

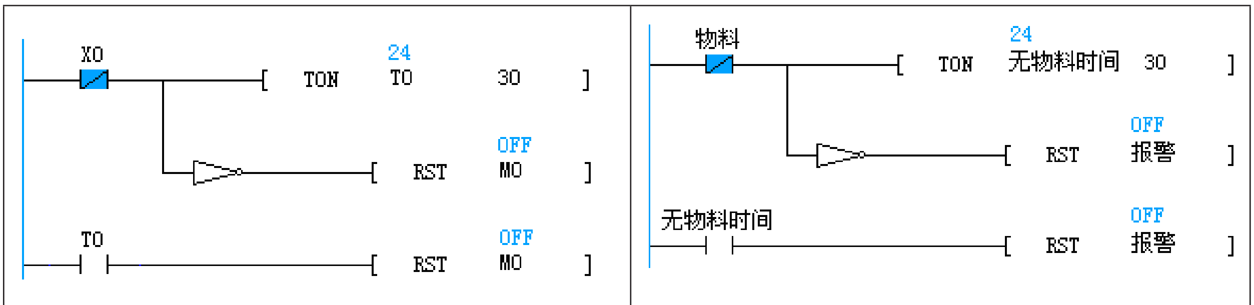
3.5.2 变量的注释

变量可以在全局变量表和局部变量表中定义（具体定义方法请参见 *2.2.3* 全局变量表 和 *3.6.3* 子程序变量表定义），正确定义的变量可以在梯形图中使用。需要使用某个地址时，可以用代表此地址的变量名来代替，以增强程序的可读性。下图是在全局变量表中定义的变量：



1．符号编址

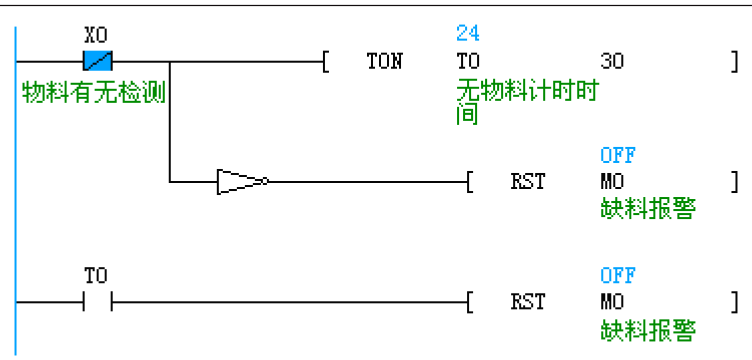
当定义了变量后，用户可以通过菜单栏中的“视图→符号编址”在变量名和元件地址之间切换，还可以通过工具栏中的快捷键“ ”进行选择。下图分别展示了两种显示模式下的同一梯形图程序：



未选中符号编址的状态　　选中了符号编址的状态

2．元件注释

可以通过菜单栏中的“视图→元件注释”选择是否在梯形图程序中显示元件注释或选择显示 2 行还是显示 4 行元件注释，还可以通过工具栏中的快捷键“ ”不显示注释，“ ”显示两行注释，“ ”显示四行注释，对元件注释进行选择。下图是在显示 2 行元件注释时的梯形图程序：



3.6 子程序

3.6.1 子程序概念

子程序是一种独立的程序体，可以被主程序或其它子程序调用，是用户程序可以选择的部分。
使用子程序编写用户程序时具有以下优点：
1．可减少用户程序的尺寸，功能相同的用户程序段，可以编写成为一个子程序反复调用。
2．使程序的结构更加清晰，特别是可以简化主程序结构。
3．提高用户程序的可移植性。

3.6.2 子程序的使用注意事项

在编写或调用子程序时，应注意以下事项：
1．支持子程序的镶嵌调用，最大嵌套调用层数为 5 层；中断服务程序只能调用 1 个子程序。
下例演示了一个合法的 5 层嵌套调用关系：
TASK → SBR1 → SBR2 → SBR3 → SBR4 → SBR5。（→代表使用 CALL 指令调用相应的子程序）
2．不支持子程序的递归调用和循环调用。
如下两种示例，演示了非法的子程序调用关系：
TASK → SBR0 → SBR0（递归调用，非法）；TASK → SBR0 → SBR1 → SBR0（循环调用，非法）
3．一个用户程序中最多可定义 64 个子程序（不包括中断程序）。
4．一个子程序中局部变量表内最多可定义 64 个位元件和 64 个字元件变量，子程序的局部变量表中 IN、OUT、IN_OUT 三种参数的定义均不能超过 16 个。
5．子程序调用时，应注意 CALL 指令中填写的操作数的属性应与子程序局部变量表中定义的变量属性相匹配，否则编译不通过。
6．每个子程序并不一定都被调用，如果空间允许，可以多放一些子程序，子程序不调用不影响扫描速度。

3.6.3 子程序变量表定义

● **子程序变量表**
子程序变量表的作用是声明子程序的接口参数和局部变量（统称为变量），并规定它们的使用属性。
● **子程序变量的属性说明**
子程序的接口参数和局部变量（统称为变量）具有以下属性：
1．变量地址
每个子程序接口参数或局部变量都被分配一个固定的 LM 元件或 V 元件地址。该地址是编程软件根据变量的数据类型，按地址连续的原则自动分配给子程序接口参数或局部变量。
2．变量名称
可以为子程序接口参数或局部变量取一个变量名称（别名），在程序中可以使用变量名称引用的方式使用变量。
3．变量类型
子程序接口参数或局部变量分为 IN 型、OUT 型、IN_OUT 型、TEMP 型。
（1）IN 型变量在子程序被调用时，用于传递子程序的输入值。
（2）OUT 型变量在子程序返回时，用于传递子程序调用返回值。
（3）IN_OUT 型变量在子程序被调用时，用于传递输入值；在子程序返回时，用于传递调用返回值。
（4）TEMP 型变量只作为子程序范围内有效的局部变量使用。
4．变量数据类型
变量数据类型属性规定了变量的数据宽度及数据范围，下表列出了变量数据类型的种类：

变量数据类型	数据类型说明	占用 LM / V 元件地址
BOOL	位型变量	占用 1 个 LM 元件地址
INT	符号整数型变量	占用 1 个 V 元件地址
DINT	符号长整数型变量	占用 2 个连续 V 元件地址
WORD	字型变量	占用 1 个 V 元件地址
DWORD	双字型变量	占用 2 个连续 V 元件地址
REAL	浮点型变量	占用 2 个连续 V 元件地址

☞ 注意

1．如果子程序中用到的局部变量在子程序变量表中未声明，则此变量的变量类型默认为 TEMP 型。

2．当全局变量表和局部变量表重名时，前面加“#”表示是局部变量。

3.6.4 子程序参数传递

在主程序里调用子程序时，如果子程序里定义了局部输入 / 输出变量，那么子程序的接口参数里需填写相应的数值或者全局 / 临时变量元件。注意局部变量与接口参数的数据类型应一致。

3.6.5 子程序的使用示例

下面用一个示例说明如何编写及调用子程序

在主程序中调用子程序，让子程序完成两个整型数据的加法运算，并将运算结果赋予 D100。

步骤 1：在工程中创建一个子程序，为创建的子程序命名为“加法运算”。

步骤 2：编写子程序

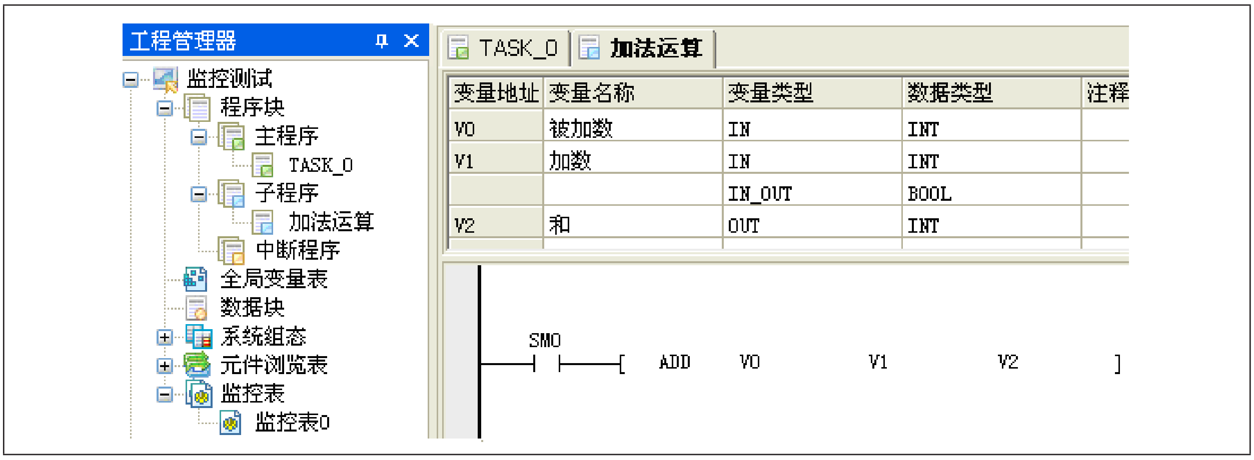
1．在子程序“加法运算”的变量表中建立子程序的调用操作数接口。

（1）定义变量 1：变量名为“被加数”，变量类型为“IN”型，数据类型为“INT”型，变量地址占用一个“V0”。

（2）定义变量 2：变量名为“加数”，变量类型为“IN”型，数据类型为“INT”型，变量地址占用一个“V1”。

（3）定义变量 3：变量名为“和”，变量类型为“OUT 型”，数据类型“INT”型，变量地址占用一个“V2”。

2．如下图编写子程序 - 加法运算：



步骤 3：编写主程序，调用子程序

在主程序中使用 CALL 指令调用子程序“加法运算”。可使用参数传递对应关系表，填写调用子程序时带入或返回的参数。

步骤 4：编译、下载、运行用户程序，验证子程序的逻辑正确性。

示例执行结果：

当 M0 为 ON 时，子程序被调用，将被加数 D0 中的数值 100 传递给 V0，加数 D1 中的数值 50 传递给 V1，子程序完成加法运算后将运算结果 V2 中的数值 150 传递给 D100，最后 D100=150。

3.7 指令通用说明

3.7.1 指令的操作数

指令的操作数可分为以下两类。

1．源操作数

指令对其数据进行读取，用于运算处理。在指令说明中用 S 来表示，当源操作数多于 1 个时，用 S_1、S_2…表示。

2．目的操作数

指令对目的操作数进行控制或输出。在指令说明中用 D 来表示，当目的操作数多于 1 个时，用 D_1、D_2…表示。

操作数可以是位元件、字元件、双字元件或常数。注意操作数为双字元件时，标号大的字元件为高位，标号小的字元件为低位。具体请参见第五章和第六章相关指令的详细说明。

3.7.2 标志位

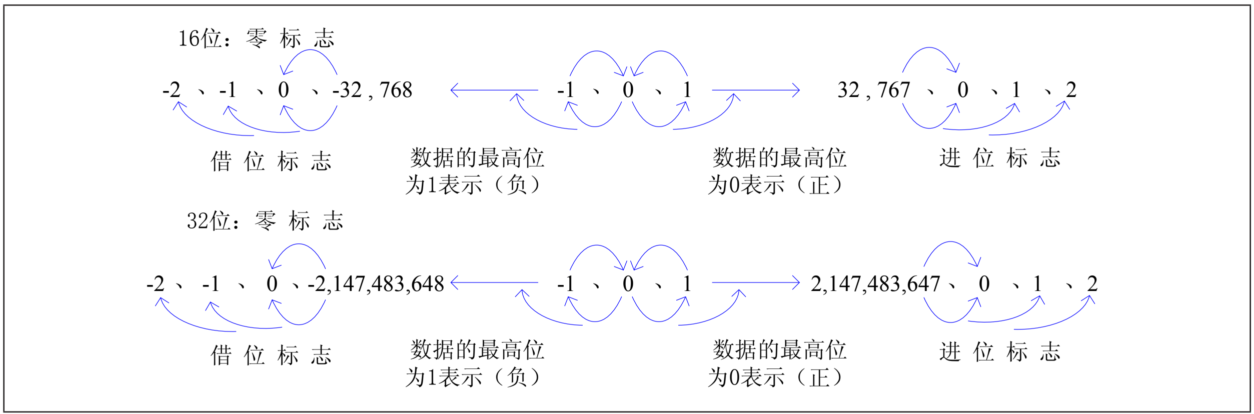
指令运算可能会对三种标志位产生影响。

1．零标志 SM180：指令运算结果为零时，SM180 置 ON。

2．进位标志 SM181：指令运算出现进位时，SM181 置 ON。

3．借位标志 SM182：指令运算出现借位时，SM182 置 ON。

标志位的动作与数值的正负关系如下：



3.7.3 指令的使用限制

一些指令的应用有一些限制条件，以下列举了其中的一部分。具体请参见指令的相关详细说明。

1．硬件资源独占

有的指令执行时，会占用硬件资源，与此硬件资源相关的其它指令就不能同时使用。

例如：高速计数指令、SPD 测频指令等，任一指令都会占用 X0 ～ X7 的某些输入点，这些指令同时使用时会相互冲突。

2．时间独占

有些指令执行时会持续一段时间。因此使用这些指令时，需要保证指令有足够的时间来完成功能，在系统运行某一时刻只能执行 1 条。

例如：由于通讯的时间性，对自由口发送指令 XMT，同一时刻只能执行 1 条；同样，自由口接收指令 RCV 也是如此。每次 Modbus 指令在执行时，也会有一段时间的独占情况。对高速输出指令、定位指令等也都有同样的情况。

3．指令应用范围限制

有些指令的使用范围受限，在某种环境下不能使用。

例如：MC/MCR 指令不能在 SFC 顺序功能图编程的步进状态中使用。

3.7.4 沿指令的产生机制

MX 系列 PLC，沿指令的产生是本扫描周期的状态与上一次扫描到此步的状态进行比较。

在系统资源中存在一些特殊程序块，例如：子程序、步进状态等，这些特殊的程序块只有在被调用时或处于步进状态有效时才被扫描，例如：步进状态 S0 无效时（即 S0 为 OFF 时），其中的程序不被扫描。在这些特殊程序块中，使用沿检测触点 LDP、LDF、ANDP、ANDF、ORP、ORF、EU、ED 时，需要注意触点的变化检出受程序块调用的影响，因

此不推荐使用。
若要使用，一定要根据程序流程仔细推敲。
如下以步进状态中使用沿检测触点为例说明（图 1 为修改前的程序，图 2 为修改后的程序）：

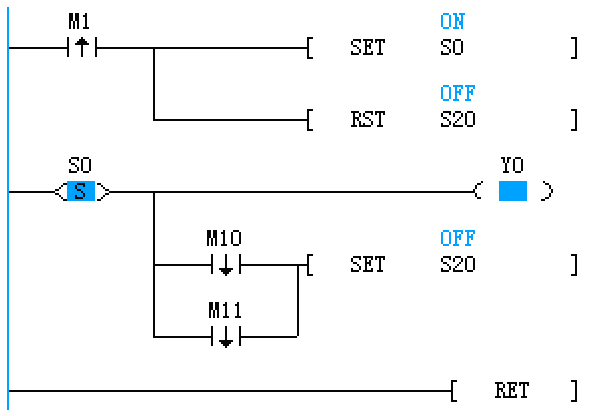


图 1

步进状态无效时变化的触点在步进状态再次有效时才被检出。对于步进状态无效时变化的条件使用沿检测时，请按下述所示修改程序。

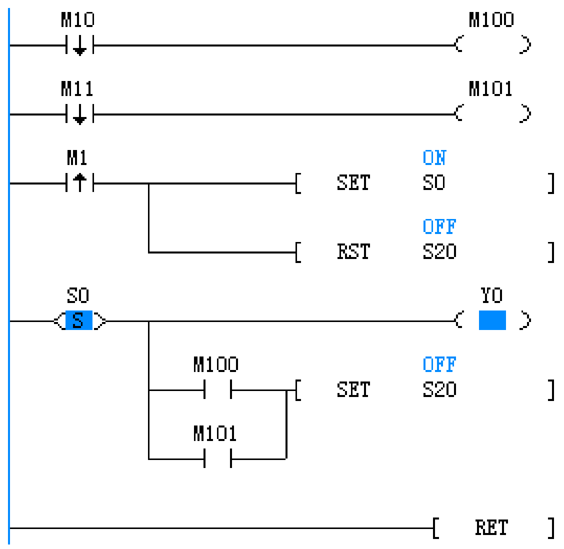
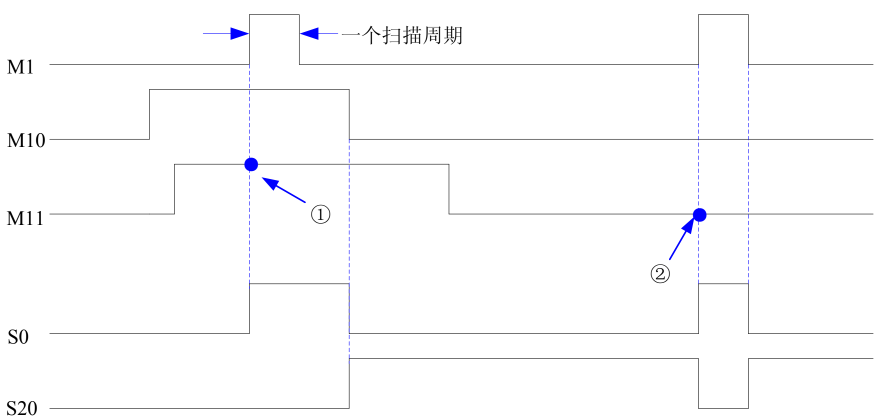


图 2

若使用修改前的程序，则步进中的沿检测与步进的跳转如下时序图所示：



当 S0 第一次有效时，检测到 M11 为 ON 状态（见①处）。
当 S0 再次有效时，检测到 M11 为 OFF 状态（见②处），比较这两个状态而产生下降沿，而此下降沿并不是用户真正需要的。

第四章 软元件和数据

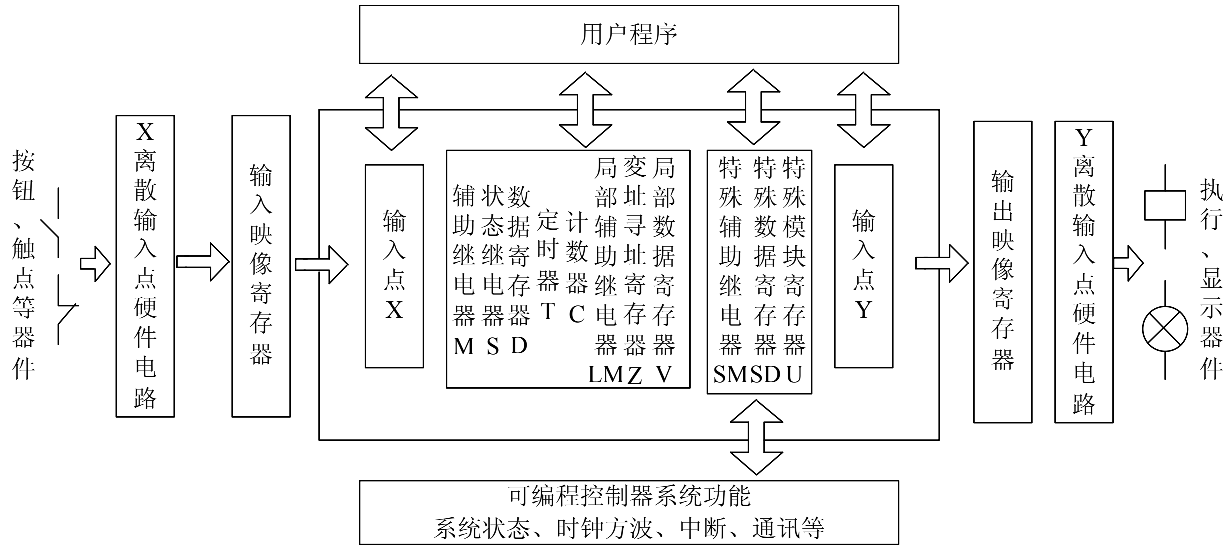
本章详细讲述了 MX 系列 PLC 各种软元件的定义、种类和作用以及数据类型。

4.1 软元件种类和作用

4.1.1 软元件概述

PLC 在系统设计中配置了多种虚拟的元件，用以代替继电器控制电路中真实的普通继电器、时间继电器等器件。这些虚拟的元件统称为软元件。PLC 采用软元件来进行程序运算和系统功能配置，从而实现所需的运算和控制功能。由于软元件是虚拟的元件，因此可以在程序中反复使用，理论上没有数量的限制（实际上与程序容量有关），而且软元件不存在真实器件的机械电气方面的故障问题，从而使 PLC 的可靠性远远高于继电器控制电路，编程和修改逻辑都更简单方便。

MX 系列 PLC 的软元件种类和作用如下图所示：



PLC 软元件的种类和作用

在本手册中，软元件按种类名称简称为“某某元件”。例如：

输入点 X 简称为“X 元件”；输出点 Y 简称为“Y 元件”；辅助继电器 M 简称为“M 元件”；数据寄存器 D 简称为“D 元件”；状态继电器 S 简称为“S 元件”。

内部软元件虽然延用了传统电气控制中的继电器、线圈及触点等名称，但 PLC 内部并不存在这些实际物理器件，例如线圈所对应的只是 PLC 内部存储器的一个基本单元（一个位），若该位为 1 表示线圈得电，该位为 0 表示线圈失电。使用常开触点即直接读取该对应位的值，使用常闭触点则读取该位的相反值。多个继电器将占有多个位，8 个位组成一个字节，两个字节称之为一个字，两个字组成双字，当多个继电器一并处理时则使用字节、字或双字。PLC 内部还有两种软元件：定时器及计数器，因此还可以进行一些字节、字或双字形式的数值处理。

4.1.2 软元件一览表

MX 系列 PLC 的软元件种类按功能编制和划分，不同元件执行不同的功能，且寻址简便。

MX 系列 PLC 的软元件一览表如下所示：

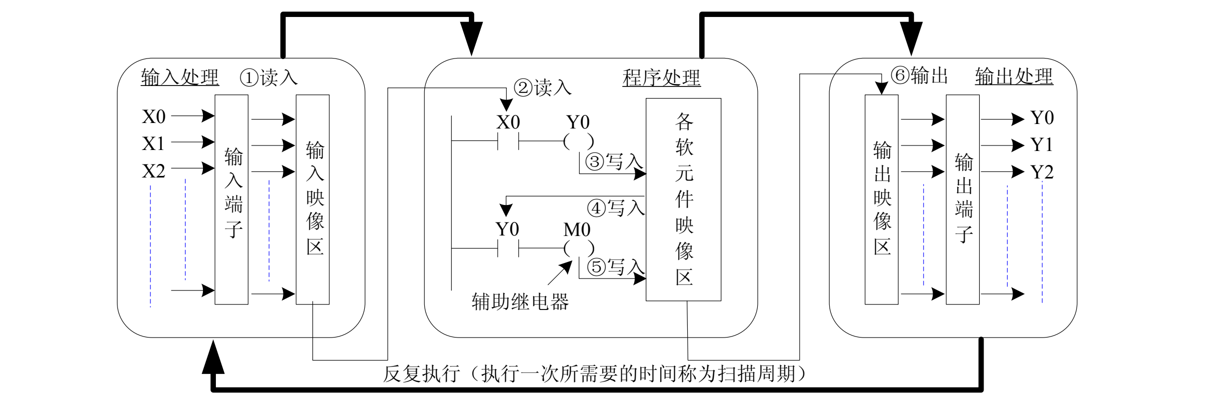
		MX1H 系列	MX2H 系列	元件编制方式
软元件资源	输入输出点	128 入 /128 出 （输入 X0 ～ X177，输出 Y0 ～ Y177）	256 入 /256 出 （输入 X0 ～ X377，输出 Y0 ～ Y377）	8 进制
	辅助继电器	2048 点 （M0 ～ M2047）	4096 点 （M0 ～ M4095）	10 进制
	局部辅助继电器	64 点 （LM0 ～ LM63）	64 点 （LM0 ～ LM63）	10 进制
	特殊辅助继电器	512 点 （SM0 ～ SM511）	512 点 （SM0 ～ SM511）	10 进制
	状态继电器	1024 点 （S0 ～ S1023）	1024 点 （S0 ～ S1023）	10 进制

		MX1H 系列	MX2H 系列	元件编制方式
软元件资源	定时器	256 个 （T0 ～ T255）	256 个 （T0 ～ T255）	10 进制
	计数器	256 个 （C0 ～ C255）	256 个 （C0 ～ C255）	10 进制
	数据寄存器	4000 个 （D0 ～ D3999）	8000 个 （D0 ～ D7999）	10 进制
	局部数据寄存器	64 个 （V0 ～ V63）	64 个 （V0 ～ V63）	10 进制
	变址寻址寄存器	16 个 （Z0 ～ Z15）	16 个 （Z0 ～ Z15）	10 进制
	特殊数据寄存器	512 个 （SD0 ～ SD511）	512 个 （SD0 ～ SD511）	10 进制
	特殊模块寄存器	U0.0 ～ U1.199	U0.0 ～ U7.199	10 进制

4.1.3 输入点和输出点

软元件名称	功能说明
输入点	输入点 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。可由外部输入信号驱动，也可以由 MXProgrammer 在监控状态时进行强制 ON/OFF 动作。它的触点可无限制地多次使用。 软元件表示：X0、X1、… X7、X10、X11…，软元件符号用 X 表示，顺序以八进制编号。在主模块及扩展模块上均有输入点编号的标示。 PX0 ～ PX7 是指 X0 ～ X7 的硬件端口，不经过数字滤波处理，对 PX0 ～ PX7 的操作不受扫描周期影响，实时性强。 单独使用可用作位元件，组合使用可用作字元件或双字元件，如 K4X0，K8X30。
输出点	输出点是直接关联到外部用户控制装置的硬件端口的软元件，在逻辑上与 PLC 的物理输出端口一一对应。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。它可以由输入触点、内部其它元件的触点以及它自身的触点驱动。它使用一个常开触点接通外部负载，其触点可无限次使用。 软元件表示：Y0、Y1、…Y7、Y10、Y11…，软元件符号用 Y 表示，顺序以八进制编号。在主模块及扩展模块上均有输出点编号的标示。 PY0 ～ PY7 是指 Y0 ～ Y7 的硬件端口，对 PY0 ～ PY7 的操作不受扫描周期影响，实时性强。 单独使用可用作位元件，组合使用可用作字元件或双字元件，如 K4Y0，K8Y30。

X 元件状态的采集是通过输入映象寄存器进行的。Y 元件状态的输出是通过输出映象寄存器来驱动输出电路实现的。这两个操作都在 PLC 扫描周期模型中的 I/O 刷新阶段进行，如下图所示。可见在用户程序运行过程中，PLC 对 I/O 的响应有短暂延迟的特性，输入滤波器和输出元件的驱动时间及运算周期也会出现响应滞后的情况。



I/O 刷新原理

1．输入处理

PLC 在执行程序之前，将可编程控制器的所有输入端子的 ON/OFF 状态读入输入映象区。在执行程序的过程中，即使输入发生变化输入映象区的内容也不变，而在下一个周期的输入处理时读入该变化。此外，即使输入触点出现 ON → OFF、OFF → ON 的变化时，在判定 ON/OFF 之前，输入滤波器会造成响应滞后。（采用数字式滤波器的输入端子可利用可编程控制器的程序来改写数值）。

2．程序处理

可编程控制器根据存储器中的指令内容，从输入映象区中读出各软元件的 ON/OFF 状态，从零步开始依次进行运算，然后将结果写入映象区。因此，各软元件的映象存储区随着程序的执行逐步改变其内容。而且，输出继电器的内部触点根据输出映象存储区的内容执行动作。

3．输出处理

一旦所有指令执行结束，将输出 Y 元件映象存储区的 ON/OFF 状态输出，这成为了可编程控制器的实际输出。可编程控制器的外部输出用的触点，按照输出用的软元件的响应滞后时间动作。

4．分类

X 元件对应的输入通道有数字滤波功能，可通过系统组态设置滤波时间。X0 ～ X5 可以作 SPD 测频指令的输入端；X0 ～ X7 还可作为高速计数器软元件的计数输入端和外部中断、脉冲捕捉的输入端。

MX1H/2H/2V 的 Y 元件中 Y0 ～ Y1/Y0 ～ Y2/Y0 ～ Y7 可以作为高速输出之用，其余为普通输出端。

4.1.4 辅助继电器（M）

软元件名称	功能说明
辅助继电器	辅助继电器与外部没有直接联系，它是 PLC 内部的一种辅助继电器，其功能与电气控制电路中的中间继电器相同，每个辅助继电器也对应着内存的一基本单元，它可由输入触点、输出触点及内部其它元件的触点驱动，它自己的触点也可以无限制地多次使用。辅助继电器无对外输出，要输出则要通过输出点（Y）。 软元件表示：M0，M1，M2…软元件符号用 M 表示，顺序以十进制编号。 M 单独使用可用作位元件，组合使用可用作字元件或双字元件，如 K4M100，K8M20。

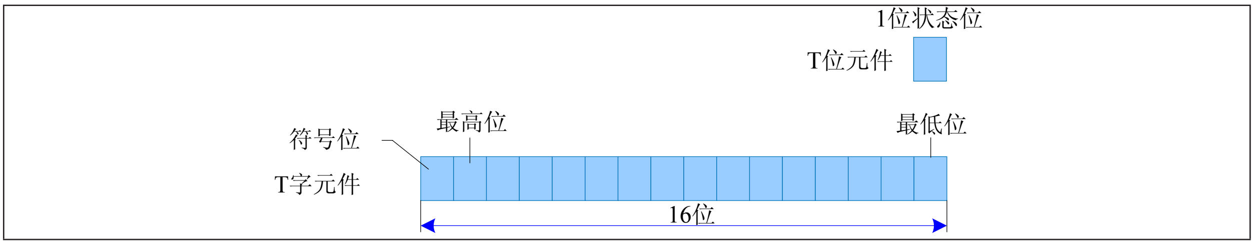
4.1.5 步进状态继电器（S）

软元件名称	功能说明
步进状态继电器	MX 系列 PLC 提供一种属于步进动作的控制程序输入方式，利用指令 STL 控制步进点 S 的转移，便可很容易写出控制程序。如果程序中完全没有使用到步进程序时，步进状态继电器可被当作辅助继电器 M 来使用，也可当作报警信号点使用。 软元件表示：S0，S1，S2…软元件符号用 S 表示，顺序以十进制编号。 S 单独使用可用作位元件，组合使用可用作字元件或双字元件，如 K2S100，K8S20。

4.1.6 定时器（T）

软元件名称	功能说明
定时器	定时器用来完成计时的功能。定时器含有线圈、触点及计时值寄存器，线圈得电定时器计时，当计时值到达预设值时，它的触点便动作。 定时器按计时精度分为 1ms 定时器、10ms 定时器、100ms 定时器。 软元件表示：T0，T1，T2…软元件符号用 T 表示，顺序以十进制编号。不同的编号范围对应不同的计时精度。 定时器包括位元件，字元件。 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。 注意：T 元件最大计时值为 32767，预设值为 －32768 ～ 32767。

定时器的位元件和字元件如下图所示：



T 元件按计时精度分类如下：

T 元件	计时精度	设定值范围	实际的设定值
T0 ～ T209	100ms 精度	0 ～ 32,767	0 ～ 3276.7s
T210 ～ T251	10ms 精度	0 ～ 32,767	0 ～ 327.67s
T252 ～ T255	1ms 精度	0 ～ 32,767	0 ～ 32.767s

100ms 精度：100ms 定时器对定时器启动后的 100ms 间隔进行计数，执行定时器指令启动计时，但是 100ms 定时器在每次扫描周期的开始刷新（也就是说，在一个扫描周期内定时器位和定时器当前值保持），把累计的 100ms 间隔数加到启动的定时器的当前值。

由于定时器在 100ms 内可以在任何地方启动，预设值必须大于最小需要的时间间隔。例如，使用 100ms 定时器时，为了保证至少 2000ms 的时间间隔，预设值应该设为 21。

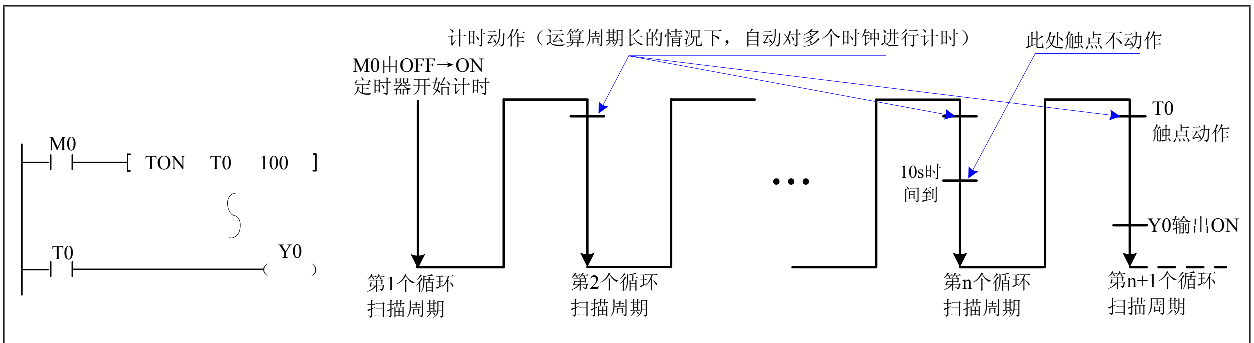
10ms 精度：10ms 定时器对定时器启动后的 10ms 间隔进行计数，执行定时器指令启动计时，但是 10ms 定时器在每次扫描周期的开始刷新（也就是说，在一个扫描周期内定时器位和定时器当前值保持），把累计的 10ms 的间隔数加到启动的定时器的当前值。

由于定时器在 10ms 内可以在任何地方启动，预设值必须大于最小需要时间间隔。例如，使用 10ms 定时器要确保至少 100ms 的时间间隔，预设值应该设为 11。

1ms 精度：1ms 定时器对定时器启动后的 1ms 间隔进行计数，执行定时器指令启动计时，但是 1ms 定时器每隔 1ms 刷新一次（定时器位和定时器当前值），不和扫描周期同步。也就是说，定时器位和定时器当前值在扫描周期大于 1ms 的一个周期中要刷新几次。

定时器指令用来启动、复位定时器，或在 TONR 指令中关闭定时器。

由于定时器在 1ms 内可以在任何地方启动，预设值必须大于最小需要时间间隔。例如，使用 1ms 定时器要确保至少 50ms 时间间隔，预设值应该设为 51。



从上面的动作图可以得知，驱动线圈开始到触点动作为止的定时器触点的动作精度 = T+T0 － α

T：定时器设定时间（计时单位 × 预设值　单位：s）

T0：运算周期（单位：s）

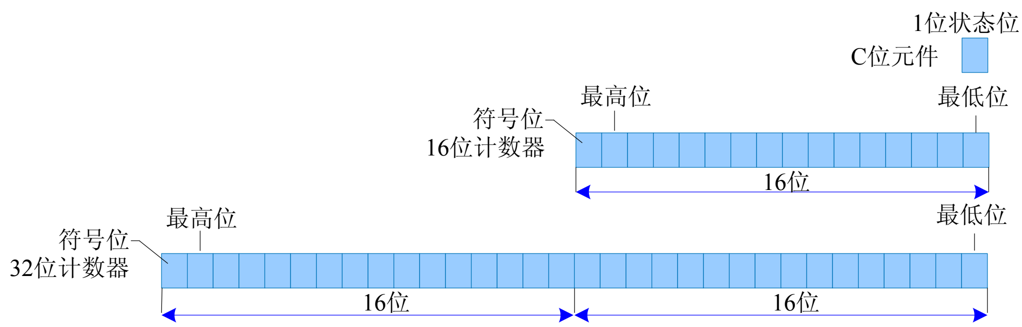
α：根据 1ms，10ms，100ms 定时器分别为 0.001、0.01、0.1（单位：s）

如果编程的时候，触点在定时器线圈前面的话，最大误差情况为 +2T0。此外，定时器的设定值为 0 的时候，在下一个循环中，线圈指令执行时，输出触点动作。中断执行型的 1ms 定时器是在线圈指令执行后，以中断方式对 1ms 的时钟脉冲进行计数。

4.1.7 计数器（C）

软元件名称	功能说明
计数器	计数器用来实现计数功能。使用计数器要事先给定计数器的预设值（即要计数的脉冲数）。计数器由线圈、触点及计数储存器组成，当线圈由 OFF → ON，即该计数器有一脉冲输入，其计数值加一，计数器有 16 位及 32 位（包括普通计数器和高速计数器）可供选择。 软元件表示：C0、C1、C2…软元件符号用 C 表示，顺序以十进制编号。 包括位元件，字元件或双字元件。 不作为计数器使用的计数器编号，可以作为数据记忆用的数据寄存器使用。

计数器的位元件和字元件如下所示：



C 元件分类如下表所示：

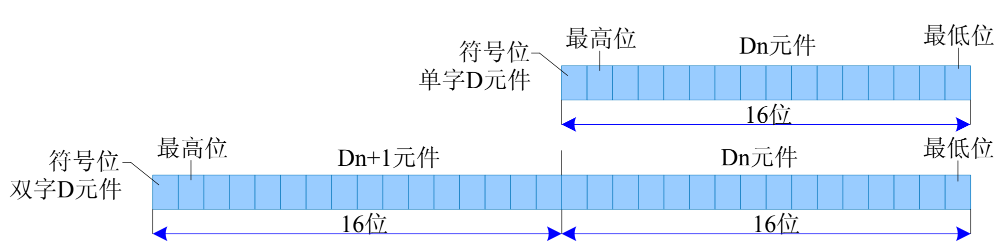
C 元件	计数功能	设定值范围	适用的计数器指令类型
C0 ～ C199	16 位增计数器	0 ～ 32,767	16 位增计数指令；16 位循环计数指令
C200 ～ C234	32 位增减计数器	－2,147,483,648 ～ +2,147,483,647	32 位增减计数指令
C235 ～ C255	32 位高速计数器		高速 I/O 指令

有关这 4 类适用指令的说明请参见第五章 基本指令和第六章 应用指令。

4.1.8 数据寄存器（D）

软元件名称	功能说明
数据寄存器	PLC 在进行各类顺序控制和计时值与计数器有关控制时，常常要作数据处理和数值运算，而数据寄存器就是专门用于储存数据或各类参数的。每个数据寄存器内有 16 位二进制数值，即存有一个字，处理双字用相邻编号的两个数据寄存器，大编号的寄存器为高位，小编号的寄存器为低位。 软元件表示：D0、D1、D2…软元件符号用 D 表示，顺序以十进制编号。 D 可用作字元件或双字元件。 注意：在采用 MXLink 协议、PTO、PID、TPID 等功能时，部分的 D 元件会被系统调用，用户在编程和修改程序时请留意。

数据寄存器的字元件与双字元件如下所示：



单字D元件数据范围是：－32,768～32,767
双字D元件数据范围是：－2,147,483,648～2,147,483,647

4.1.9 特殊辅助继电器（SM）

软元件名称	功能说明
特殊辅助继电器	特殊辅助继电器是与 PLC 系统功能密切相关的软元件，反映了 PLC 系统功能、状态。有关特殊辅助继电器的详细功能说明请参见本手册附录二 *SM* 元件一览表。 软元件表示：SM0、SM1、SM2…软元件符号用 SM 表示，顺序以十进制编号。 SM 单独使用可用作位元件，组合使用可用作字元件或双字元件，如 K4SM10，K8SM30。 注意：只读的 SM 元件不能赋值。

4.1.10 特殊数据寄存器（SD）

软元件名称	功能说明
特殊数据寄存器	特殊数据寄存器是与 PLC 系统功能密切相关的软元件，反映了 PLC 系统功能参数、状态代码值、指令运行数据。有关特殊数据寄存器的详细功能说明请参阅本手册附录三 *SD* 元件一览表。 软元件表示：SD0、SD1、SD2…软元件符号用 SD 表示，顺序以十进制编号。 SD 可以用作字元件或双字元件。 注意：只读的 SD 元件不能赋值。

4.1.11 特殊模块寄存器（Un.b）

软元件名称	功能说明
特殊模块寄存器	特殊模块寄存器是与 PLC 特殊模块功能密切相关的软元件，反映了 PLC 系统功能参数、状态代码值、指令运行数据。 软元件表示：U0.0、U0.1、U0.2…，软元件符号用 U 表示，如 U0.1 表示第 0 号单元的特殊模块的第 1 个字，顺序以十进制编号。 Un.b 可以用作字元件。 注意：只读的 U 元件不能赋值。

4.1.12 变址寻址寄存器（Zn）

软元件名称	功能说明
变址寻址寄存器	变址寻址寄存器同一般的寄存器一样都是 16 位的数据寄存器，它可以自由地被写入和读取，可用于位元件、字元件及双字元件的间接寻址。 软元件表示：Z0、Z1、Z2…软元件符号用 Z 表示，顺序以十进制编号。 Zn 只能用作字元件。

4.1.13 局部辅助继电器（LM）

软元件名称	功能说明
局部辅助继电器	局部辅助继电器是在各独立程序体（主程序、子程序和中断程序）内局部有效的变量元件。不同的程序体中不能直接共用任何 LM 元件的状态。LM 元件可用于定义子程序的接口参数，实现接口参数传递功能。详细内容请参见 *3.6* 子程序。在用户程序执行中离开了某一程序体，系统就会重新定义 LM 元件。在返回主程序或调用子程序时，重新定义的 LM 元件的值将会被清零，或者根据接口参数传递功能来获得相应的状态。有关子程序接口参数的详细内容请参见 *3.6* 子程序。 软元件表示：LM0、LM1、LM2…软元件符号用 LM 表示，顺序以十进制编号。 LM 单独使用可用作位元件，组合使用可用作字元件或双字元件，如 K4LM0，K8LM30。

4.1.14 局部数据寄存器（V）

软元件名称	功能说明
局部数据寄存器	局部数据寄存器是在各独立程序体（主程序、子程序和中断程序）内局部有效的变量元件。不同的程序体中不能直接共用任何 V 元件的数据。V 元件可用于定义子程序的接口参数，实现接口参数传递功能。在用户程序执行中离开了某一程序体，系统就会重新定义 V 元件。在返回主程序或调用子程序时，重新定义的 V 元件的值将会被清零，或者根据接口参数传递功能来获得相应的数据。有关子程序接口参数的详细内容请参见第三章 编程概念 *3.6* 子程序。 软元件表示：V0、V1、V2…软元件符号用 V 表示，顺序以十进制编号。 V 可用作字元件或双字元件。

4.2 软元件寻址方式

4.2.1 位串组合寻址方式（Kn 寻址方式）

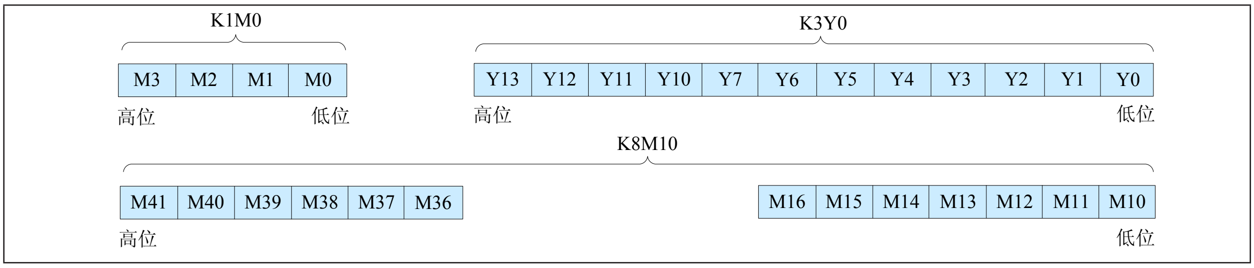
1．位串组合寻址方式概念

位串组合寻址方式（Kn 寻址方式）是将位元件串组合成字或双字使用。

2．位串组合寻址方法

位串组合寻址格式为 K（n）（E），其中 n 是一个 1 ～ 8 的整数，表示元件串长度为 n×4 位。E 代表元件串的起始位元件地址。

具体示例：



3．位串组合寻址的注意事项

如果指令的操作数使用 Kn 寻址方式，而需要存储到操作数的数据宽度大于 Kn 寻址所指定的宽度时，系统按保留低位部分，舍去高位部分的规则存储数据。

以下举例说明该种情况：

执行指令“DBITS 16# FFFFFFF0 K1M0”。

该指令执行后，操作数 2（K1M0）本应存储的运算结果为 16#1C（28），但因为 K1M0 能存储的数据的宽度为 4，不能完整地存储运算结果 16#1C，运算结果的高位部分将被舍去，所以操作数 2 的实际结果为：K1M0=16#C（12）。

4.2.2 变址寻址方式（Z 寻址方式）

1．变址寻址概念

MX 系列 PLC 提供变址寻址方式（Z 寻址方式），用户可通过使用 Z 元件（变址寻址寄存器），达到对元件进行间接寻址访问的目的。

2．Z 寻址方式使用方法

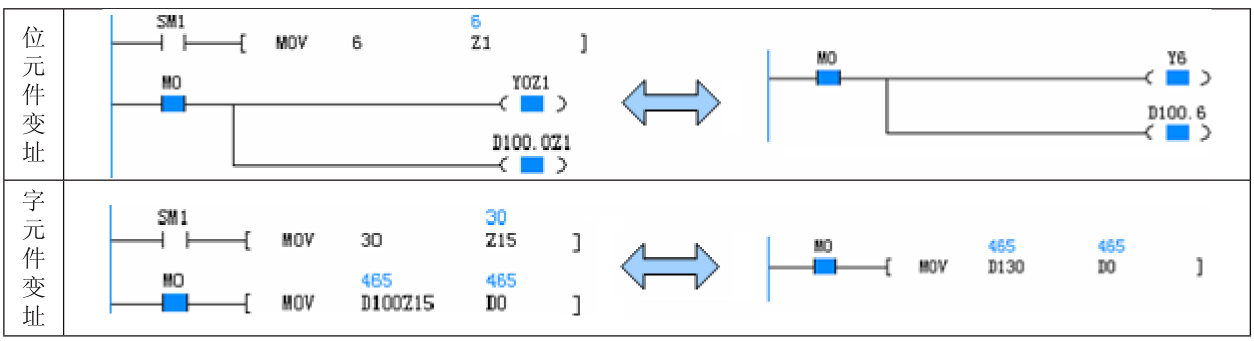
变址寻址的目标地址 ＝ 元件基地址 ＋ Z元件中存储的地址偏移量

例如：

变址寻址 D0Z0（其中 Z0=3），表示 D0 为变址寻址的基地址，变址寻址的地址偏移量存储在 Z0 中（地址偏移量等于 3），故目标地址为 D3。

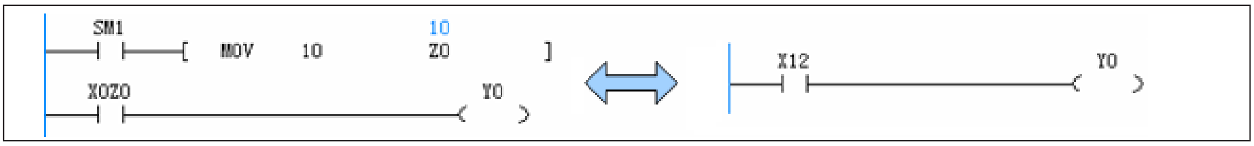
因此，在 Z0=3 的情况下，指令“MOV 45 D0Z0”和“MOV 45 D3”是等效的，指令有效执行后 D3 都会被赋值 45。

3．变址寻址方式的示例

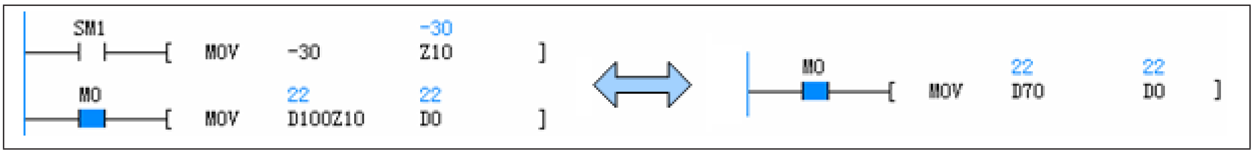


4．变址寻址方式的注意事项

（1）对于软元件编号为 8 进制数的软元件变址来说，Z 的内容要被换算成 8 进制数，然后做加法运算。如下图：



（2）在变址寻址方式（Z 寻址方式）中，Z 元件存储地址偏移量，总是被系统当作符号整数处理，即 Z 寻址方式支持负地址偏移量。如下图：



（3）SM 元件、SD 元件、S 元件不支持变址寻址方式。

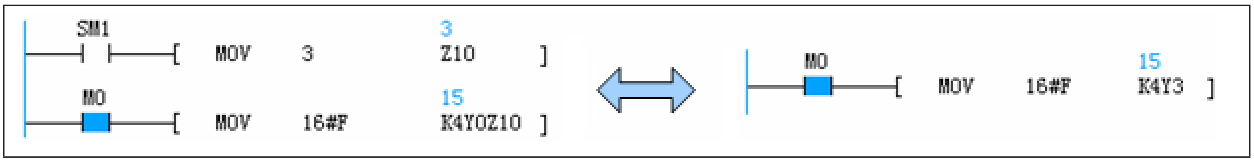
（4）在使用 Z 寻址方式时，应避免发生 Z 寻址越界的情况，例如：D7999Z0（其中 Z0=9）就发生了 Z 寻址越界情况（D 元件的最大地址为 D7999）。

4.2.3 位串组合的变址寻址方式

位串组合寻址方式也可配合变址寻址方式使用，即形如 K1X0Z10。

这种寻址方式首先通过 Z 寻址确定位串组合的起始位元件的地址，再通过 Kn 寻址确定位串的长度。

以下示例说明具体的寻址过程：



4.2.4 D、V 元件对 32 位数据的存储和寻址

1．32 位数据在 D、V 元件中的存储方式

DINT、DWORD、REAL 型的数据都是 32 位长度的，而一个 D 元件或 V 元件只有 16 位长度，因此需要两个地址连续的 D 元件或 V 元件存储 32 位数据。

MX 系列 PLC 存储 32 位数据时，小地址编号的元件用于存放 32 位长度数据的低字，大地址编号的元件存放 32 位长度数据的高字。

例如：无符号长整数数据 16# BF8A12CD 存放在（D1，D0）元件中，D1 中存放 16#BF8A，D0 中存放 16#12CD。

2．D、V 元件地址寻址 32 位数据

一个 D、V 元件地址可以寻址一个 16 位数据（如 INT、WORD 型数据），也可以寻址到一个 32 位数据（如 DINT、DWORD 型数据）。

如果指令操作数引用了 D 元件地址或 V 元件地址，那么该地址是代表一个 16 位数据，还是代表一个 32 位数据，将由操作数的数据类型决定。

例如：

在指令“MOV 16#34 D0”中，由于 MOV 指令的操作数数据类型是 INT 型，因此地址 D0 只寻址单独的 D0 元件。

而在指令“DMOV 16# FEA867DA D0”中，由于 DMOV 指令的操作数数据类型是 DINT 型，因此地址 D0 代表了 D0 开始的两个连续字元件 D0，D1。

4.3 数据

4.3.1 数据类型

数据类型	类型说明	数据长度	范围
BOOL	位	1	1（ON）、0（OFF）
INT	符号整数	16	- 32,768 ～ 32,767
DINT	符号长整数	32	- 2,147,483,648 ～ 2,147,483,647
WORD	字	16	0 ～ 65,535（16#0 ～ 16#FFFF）
DWORD	双字	32	0 ～ 4,294,967,295（16#0 ～ 16#FFFFFFFF）
REAL	浮点数	32	- 3.402823E+038 ～ 3.402823E+038

4.3.2 元件与数据类型的匹配关系

数据类型	软元件													
BOOL		X	Y	M	S	LM	SM	Dn.b		C	T			
INT	常数	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	Un.b
DINT	常数	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		Un.b
WORD	常数	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	Un.b
DWORD	常数	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		Un.b
REAL	常数							D				V		

如果指令的编写不满足匹配关系，该指令将被视为非法，例如指令“MOV 10 X0”是非法的，这是因为 MOV 指令第二个操作数的数据类型是符号整数型，而元件 X0 只能存储位型的数据。

☞ 注意

1．当操作数的数据类型为 INT，WORD 型时，使用 Kn 寻址，则 1 ≤ n ≤ 4，使用 C 元件的编号应为 C0 ～ C199。
2．当操作数的数据类型为 DINT，DWORD 型时，使用 Kn 寻址，则 5 ≤ n ≤ 8，使用 C 元件的编号应为 C200 ～ C255。

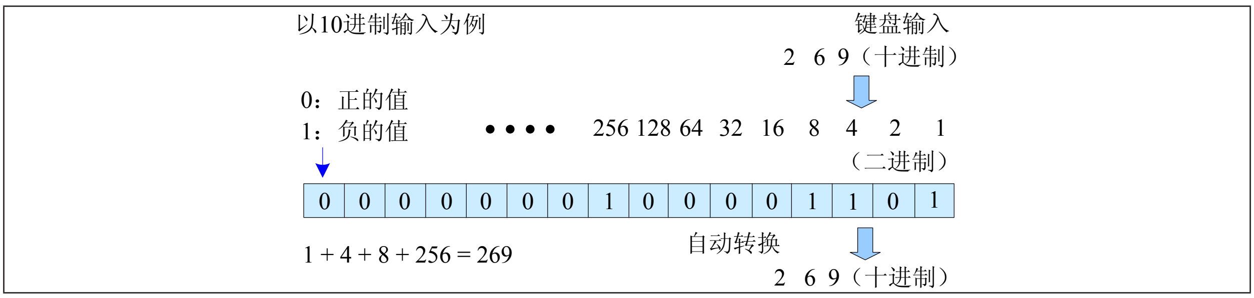
4.3.3 常数

● **常数的种类**

1．2 进制数（BIN：BINARY NUMBER）

对定时器、计数器或是数据寄存器的数值指定，是按照上述的 10 进制数和 16 进制数执行的，但是在可编程控制器的内部，这些数值都以 2 进制数进行处理。

此外，在外围设备上监控这些软元件的时候，自动转换成 10 进制数后显示（也可以切换成 16 进制），如下图所示。



负数的处理：在可编程控制器内部，负数是以 2 的补码来表现的。

2．8 进制数（OCT：OCTAL NUMBER）

MX 系列可编程控制器中，输入继电器、输出继电器的软元件编号都是以 8 进制数分配的。由于在 8 进制数中，不存在 8、9 所以按 0 ～ 7、10 ～ 17…0 ～ 77、100 ～ 107 上升排列。

3．10 进制数（DEC：DECIMAL NUMBER）

（1）定时器和计数器的设定值

（2）辅助继电器（M）、定时器（T）、计数器（C）、状态等的编号（软元件编号）

（3）应用指令的操作数中的数值指定和指令动作的指定

4．16 进制数（HEX：HEXADECIMAL NUMBER）

应用指令的操作数中的数值指定和指令动作的指定

5．BCD（BCD：BINARY CODE DECIMAL）

BCD 就是将构成 10 进制数的各位上 0 ～ 9 的数值以四位的二进制数来表现的形式。由于各位便于使用，所以使用 BCD 输出型的数字式开关和 7 段码显示器控制等用途中。

6．实数（浮点数数据）

MX 系列可编程控制器，具有能够执行高精度运算的浮点数运算功能。采用 2 进制浮点数（实数）进行浮点运算及监控。

具体的常数类型及有效范围说明见下表：

常数类型	表达示例	有效范围	说明
常十进制 16 位符号整数	- 8,949	- 32,768 ～ 32,767	
常十进制 16 位无符号整数	65,326	0 ～ 65,535	
常十进制 32 位符号整数	- 2,147,483,646	- 2,147,483,648 ～ 2,147,483,647	
常十进制 32 位无符号整数	4,294,967,295	0 ～ 4,294,967,295	
十六进制 16 位常数	16#1FE9	16#0 ～ 16#FFFF	十六进制，八进制，二进制常数本身无正负含义。选用十六进制，八进制或二进制常数作为指令操作数，操作数的正负及大小，要根据其操作数的数据类型决定
十六进制 32 位常数	16#FD1EAFE9	16#0 ～ 16#FFFFFFFF	
八进制 16 位常数	8#7173	8#0 ～ 8#177777	
八进制 32 位常数	8#71732	8#0 ～ 8#37777777777	
二进制 16 位常数	2#10111001	2#0 ～ 2#1111111111111111	
二进制 32 位常数	2#101110011111	2#0 ～ 2#11111111111111111111111111111111	
单精度浮点常数	3.1415E+3	- 3.402823E+038 ～ 3.402823E+038	符合 IEEE-754 标准。编程软件可以显示、输入 7 位有效精度的浮点常数

- **常数的转换**

MX 系列可编程控制器中处理的数值，可以按照下表的内容进行转换。

10 进制数（DEC）	8 进制数（OCT）	16 进制数（HEX）	2 进制数（BIN）		BCD	
0	0	00	0000	0000	0000	0000
1	1	01	0000	0001	0000	0001
2	2	02	0000	0010	0000	0010
3	3	03	0000	0011	0000	0011
4	4	04	0000	0100	0000	0100
5	5	05	0000	0101	0000	0101
6	6	06	0000	0110	0000	0110
7	7	07	0000	0111	0000	0111
8	10	08	0000	1000	0000	1000
9	11	09	0000	1001	0000	1001
10	12	0A	0000	1010	0001	0000
11	13	0B	0000	1011	0001	0001
12	14	0C	0000	1100	0001	0010
13	15	0D	0000	1101	0001	0011
14	16	0E	0000	1110	0001	0100
15	17	0F	0000	1111	0001	0101
16	20	10	0001	0000	0001	0110
⋮	⋮	⋮	⋮	⋮	⋮	⋮
99	143	63	0110	0011	1001	1001
⋮	⋮	⋮	⋮	⋮	⋮	⋮

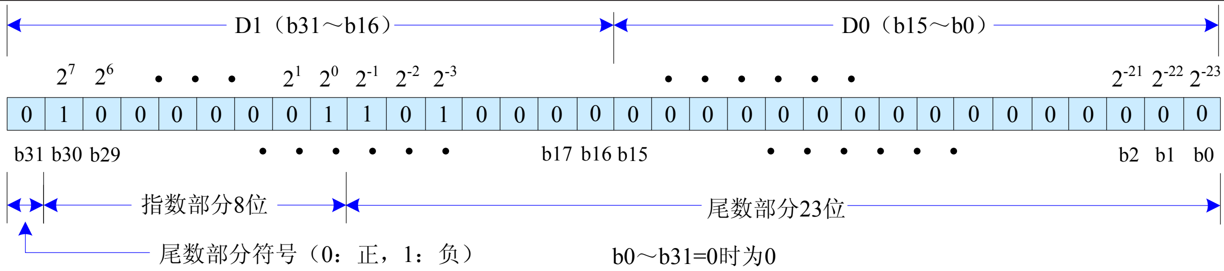
- **二进制浮点运算中数值的处理**

在可编程控制器内部使用 BIN 的整数值。在整数的除法运算中，会得出例如 40÷3=13 余 1 的答案。整数的开方运算中

小数点也被舍去。

MX 系列可编程控制器中，为了能够更加高精度地执行这些运算，可以执行浮点数运算。

在数据寄存器中处理 2 进制浮点数（实数）的时候，使用编号连续的一对数据寄存器。例如（D1，D0）时，如下图所示。



2进制浮点数（实数）$= \pm(2^0+1\times2^{-1}+0\times2^{-2}+1\times2^{-3}+\ldots+0\times2^{-23})\times2^{(1\times2^7+0\times2^6+\ldots+1\times2^0)}/2^{127} = \pm1.625\times2^{129}/2^{127} = \pm1.625\times2^2$

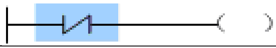
正负由b31的符号位决定，但不是补码处理。

有效位数

2 进制浮点数的有效位数如用 10 进制数表示，大约为 7 位数。2 进制浮点数的处理范围为 - 3.402823E+038 ～ 3.402823E+038。

第五章　基本指令

5.1　LD，LDI，OUT 指令

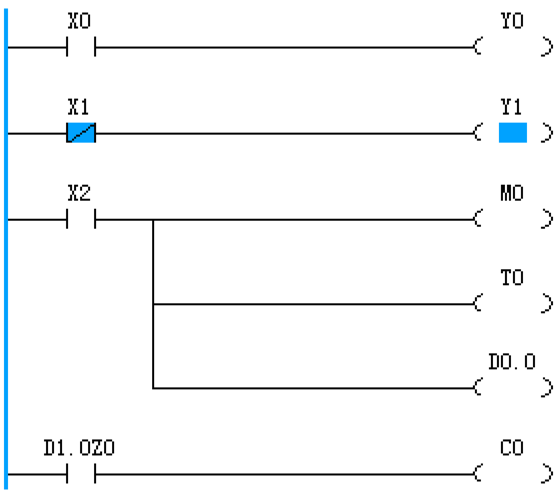
助记符、名称	梯形图和适用元件	步长
LD 常开触点	 X、Y、M、S、T、C、LM、SM、Dn.b、PX	1 步 （LM/Dn.b/PX 为 3 步）
LDI 常闭触点	 X、Y、M、S、T、C、LM、SM、Dn.b、PX	1 步 （LM/Dn.b/PX 为 3 步）
OUT 线圈输出	 Y、M、S、T、C、LM、SM、Dn.b、PY	1 步 （LM/Dn.b/S/PY 为 3 步）

＊　当指令使用 PX 元件时，可直接从端口上读取状态，不需要经过输入映象区，不受扫描周期的影响。
当指令使用 PY 元件时，可直接将输出映象区中 Yn 的状态立即输出到相应的端口，不受扫描周期的影响。
MX1H 只支持 PX 元件。

指令说明

1．LD、LDI 指令用于将触点连接到母线上。其他用法与后述的 ANB 指令组合，在分支起点处也可以使用。
2．OUT 指令是对 Y、M、S、T、C、LM、SM、Dn.b、PY 的线圈驱动指令，对输入继电器不能使用。
3．并联的 OUT 指令能多次连续使用。

使用示例




```
LD    X0
OUT   Y0
LDI   X1
OUT   Y1
LD    X2
OUT   M0
OUT   T0
OUT   D0.0
LD    D1.0Z0
OUT   C0
```

1．当 X0 为 ON 时，Y0 输出为 ON。
2．当 X1 为 OFF 时，Y1 输出为 ON。
3．当 X2 为 ON 时，并联的 OUT 指令多次连续使用，M0、T0、D0.0 输出都为 ON。
4．若 Z0=1，则 D1.0Z0 等价于 D1.1，即当 D1.1 为 ON 时，计数器 C0 的线圈输出为 ON。

5.2 LDP , LDF , ANDP , ANDF , ORP , ORF 指令

助记符 、名称	功能	梯形图和适用元件	步长
LDP 常开触点上升沿	上升沿检出运算开始	X、Y、M、S、T、C、LM、SM、Dn.b、PX	4步
LDF 常开触点下降沿	下降沿检出运算开始	X、Y、M、S、T、C、LM、SM、Dn.b、PX	4步
ANDP 常开触点与上升沿	上升沿检出串联连接	X、Y、M、S、T、C、LM、SM、Dn.b、PX	4步
ANDF 常开触点与下降沿	下降沿检出串联连接	X、Y、M、S、T、C、LM、SM、Dn.b、PX	4步
ORP 常开触点或上升沿	上升沿检出并联连接	X、Y、M、S、T、C、LM、SM、Dn.b、PX	4步
ORF 常开触点或下降沿	下降沿检出并联连接	X、Y、M、S、T、C、LM、SM、Dn.b、PX	4步

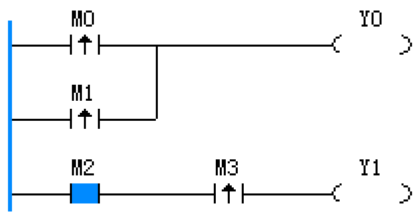
* 当指令使用 PX 元件时，可直接从端口上读取状态，不需要经过输入映象区，不受扫描周期的影响。

指令说明

1. LDP、ANDP、ORP 指令是进行上升沿检出的触点指令，仅在指定位软元件的上升沿时（OFF→ON 变化时）接通一个扫描周期。
2. LDF、ANDF、ORF 指令是进行下降沿检出的触点指令，仅在指定位软元件的下降沿时（ON→OFF 变化时）接通一个扫描周期。

使用示例

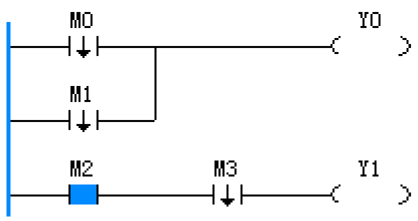
示例 1.




```
LDP   M0
ORP   M1
OUT   Y0
LD    M2
ANDP  M3
OUT   Y1
```

1. M0、M1 至少有一个由 OFF → ON 变化时，Y0 接通一个扫描周期。
2. 当 M2 为 ON 且 M3 由 OFF → ON 变化时，Y1 接通一个扫描周期。

示例 2.

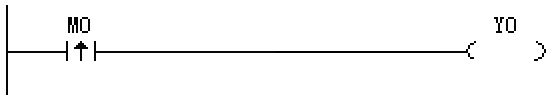


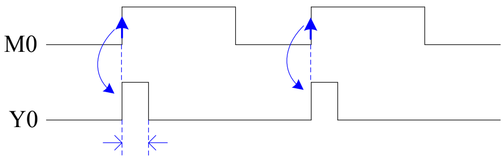

```
LDF   M0
ORF   M1
OUT   Y0
LD    M2
ANDF  M3
OUT   Y1
```

1. M0、M1 至少有一个由 ON → OFF 变化时，Y0 接通一个扫描周期。
2. 当 M2 为 ON 且 M3 由 ON → OFF 变化时，Y1 接通一个扫描周期。

示例 3. 输出驱动

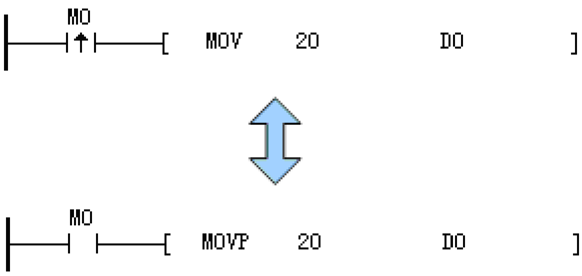
（1）





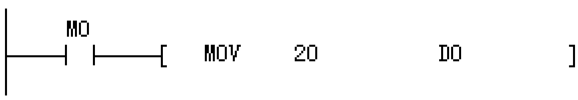
当 M0 由 OFF → ON 变化时，Y0 接通一个扫描周期。

（2）



上述两段程序等效，当 M0 由 OFF → ON 变化时，MOV、MOVP 指令都只执行一次。

（3）



与上述两种情况不同，当 M0 为 ON 时，MOV 指令每个扫描周期都执行一次。

5.3 AND , ANI 指令

助记符 、名称	功能	梯形图和适用元件	步长
AND 常开触点与	常开触点串联连接	X、Y、M、S、T、C、LM、SM、Dn.b、PX	1步（LM/Dn.b/PX为3步）
ANI 常闭触点与	常闭触点串联连接	X、Y、M、S、T、C、LM、SM、Dn.b、PX	2步（LM/Dn.b/PX为3步）

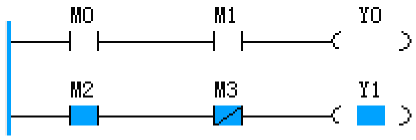
＊ 当指令使用 PX 元件时，可直接从端口上读取状态，不需要经过输入映象区，不受扫描周期的影响。

指令说明

指令 AND、ANI 将指定元件的常开触点、常闭触点与上一个触点进行逻辑“与”操作，串联触点的个数不受限制。

使用示例

示例 1.

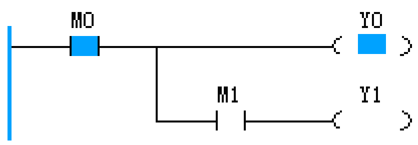



```
LD   M0
AND  M1
OUT  Y0
LD   M2
ANI  M3
OUT  Y1
```

1．当 M0 为 ON 且 M1 为 ON 时，Y0 输出为 ON。

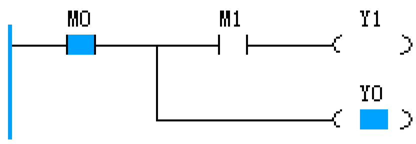
2．当 M2 为 ON 且 M3 为 OFF 时，Y1 输出为 ON。

示例 2.




```
LD   M0
OUT  Y0
AND  M1
OUT  Y1
```

图一

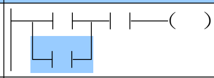
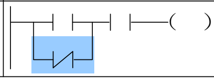



```
LD   M0
MPS
AND  M1
OUT  Y1
MPP
OUT  Y0
```

图二

如图一所示，紧接“OUT Y0”以后，通过触点 M1 可以驱动“OUT Y1”，但是当驱动顺序相反（如图二所示）时，则必须使用后述的 MPS 指令，会使得程序步增长，因此建议使用图一程序。

5.4 OR , ORI 指令

助记符、名称	功能	梯形图和适用元件	步长
OR 常开触点或	常开触点并联连接	 X、Y、M、S、T、C、LM、SM、Dn.b、PX	1步（LM/Dn.b/PX为3步）
ORI 常闭触点或	常闭触点并联连接	 X、Y、M、S、T、C、LM、SM、Dn.b、PX	2步（LM/Dn.b/PX为3步）

＊ 当指令使用 PX 元件时，可直接从端口上读取状态，不需要经过输入映象区，不受扫描周期的影响。

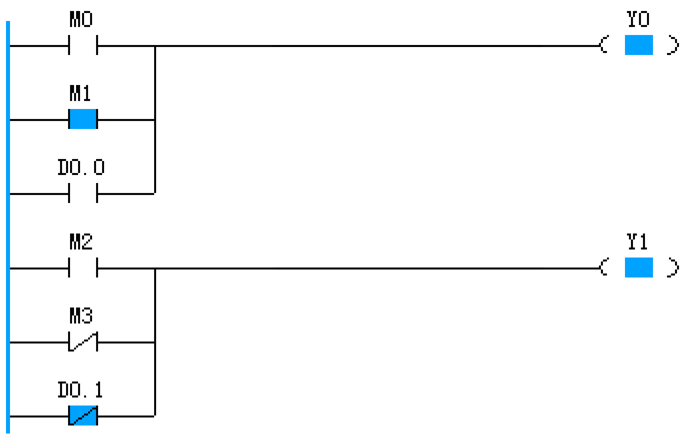
指令说明

1．OR、ORI 仅用于并联连接一个触点的指令，是对其前面 LD、LDI 指令所规定的触点再并联一个触点，而且并联触点的个数不受限制。

2．如果有两个以上的触点串联连接，并将这种串联回路块与其他回路并联连接时，采用后述的 ORB 指令。

使用示例

示例 1.



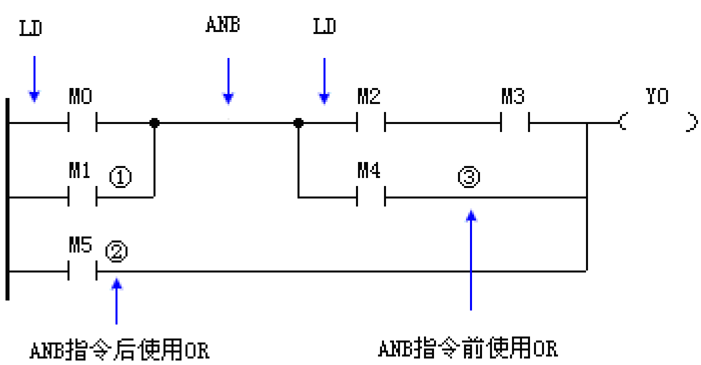

```
LD    M0
OR    M1
OR    D0.0
OUT   Y0
LD    M2
ORI   M3
ORI   D0.1
OUT   Y1
```

1．M0、M1、D0.0 中至少有一个触点为 ON，Y0 就输出为 ON。

2．M2 为 ON 或者 M3、D0.1 中至少有一个为 OFF，Y1 就输出为 ON。

示例 2.

与 ANB 指令的关系




```
LD    M0
OR    M1
LD    M2
AND   M3
OR    M4
ANB
OR    M5
OUT   Y0
```

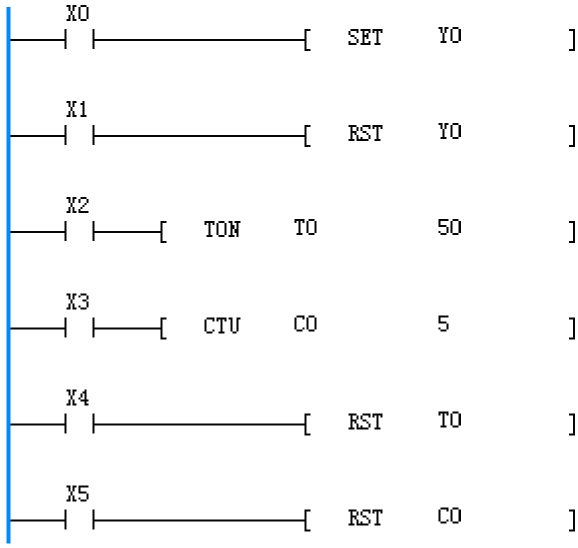
使用 OR、ORI 指令实现的并联连接，原则上是与前述的 LD、LDI 连接的（见①处），但在后述的 ANB 指令之后也可使用（见②处），并且在 ANB 指令前面加一条 LD 或 LDI 指令时也可使用（见③处）。

5.5 SET , RST 指令

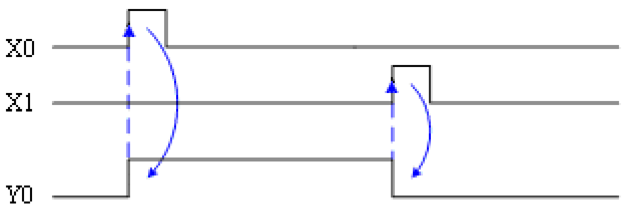
助记符 、名称	功能	梯形图和适用元件	步长
SET 线圈置位	线圈保持	[SET D] Y、M、S、SM、LM、Dn.b	1步（LM/Dn.b/S 为3步）
RST 线圈复位	线圈清除	[RST D] Y、PY、M、S、T、C、LM、SM、Dn.b	1步（LM/Dn.b/S/T/C /PY为3步）

使用示例

示例 1.



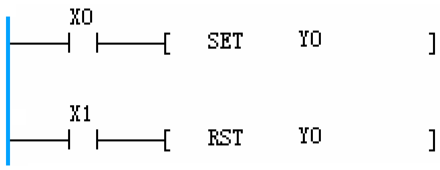

```
LD    X0
SET   Y0
LD    X1
RST   Y0
LD    X2
TON   T0 50
LD    X3
CTU   C0 5
LD    X4
RST   T0
LD    X5
RST   C0
```



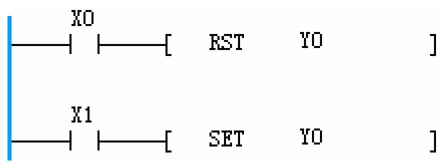
1. 在上述程序示例中，X0 接通后 Y0 输出为 ON，即使 X0 断开，Y0 仍然保持 ON 状态，直到 X1 接通 Y0 复位为 OFF。
2. 定时器、计数器当前值的清除以及触点复位也可使用 RST 指令。

示例 2.

停止优先



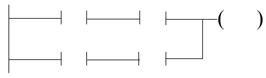
启动优先



上图是利用 RST 及 SET 指令组合成的自保电路。

1. RST 指令设置在 SET 指令之后，为停止优先。由于 PLC 是自上而下执行扫描的，因此只会输出程序最后扫描到的 Y0 的状态，所以当 X0 及 X1 同时动作时，Y0 失电，因此为停止优先。
2. SET 指令设置在 RST 指令之后，为启动优先。当 X0 及 X1 同时动作时，Y0 得电，因此为启动优先。
3. 对于同一软元件，可以多次使用 SET 或 RST 指令，顺序也可随意，但最后执行者有效。

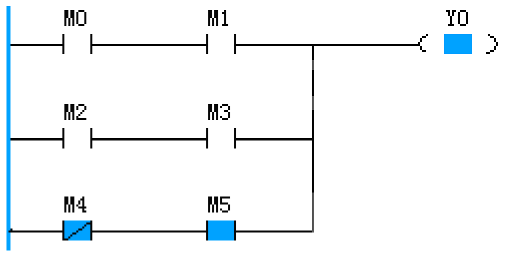
5.6 ORB : 串联回路块或指令

助记符、名称	功能	梯形图	步长
ORB 串联回路块或	串联回路块的并联连接		1 步

指令说明

1. 由两个或两个以上的触点串联连接的回路被称为“串联回路块”。将串联回路块并联连接时，需使用 ORB 指令。分支开始用 LD、LDI 指令，分支结束用 ORB 指令。
2. ORB 无操作数。
3. 有多个并联回路时，如对每个回路块使用 ORB 指令，则并联回路没有限制。
4. ORB 指令也可集中使用，但是集中使用 ORB 的次数限制在 8 次以下，请务必注意。

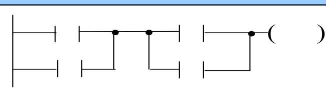
使用示例



正确的程序		不佳的程序	
LD	M0	LD	M0
AND	M1	AND	M1
LD	M2	LD	M2
AND	M3	AND	M3
ORB		LDI	M4
LDI	M4	AND	M5
AND	M5	ORB	
ORB		ORB	
OUT	Y0	OUT	Y0

ORB 有两种使用方法，一种是在要并联的两个串联回路块后面加 ORB 指令，即分散使用 ORB 指令，其并联回路块的个数没有限制（如上图正确的程序）；另一种是集中使用 ORB 指令，集中使用 ORB 的次数不允许超过 8 次，所以不推荐集中使用 ORB 指令的这种编程方法（如上图不佳的程序）。

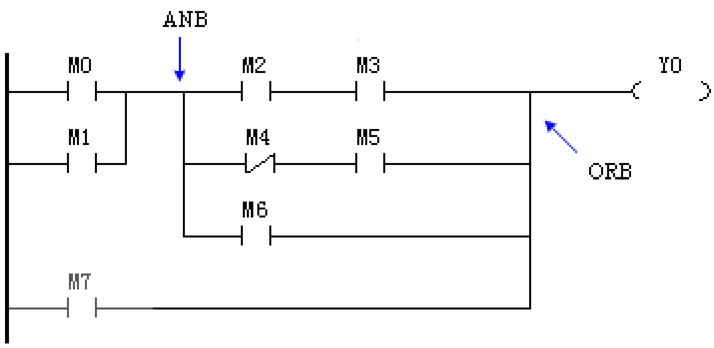
5.7 ANB : 并联回路块与指令

助记符、名称	功能	梯形图	步长
ANB 并联回路块与	并联回路块的串联连接		1 步

指令说明

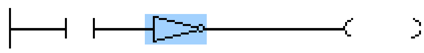
1. 由两个或两个以上的触点并联连接的回路被称为“并联回路块”。并联回路块与前面的回路串联连接时，需使用 ANB 指令。分支的起点用 LD、LDI 指令，并联回路块结束后，使用 ANB 指令与前面的回路串联连接。
2. ARB 无操作数。
3. ANB 指令的使用次数没有限制。

使用示例



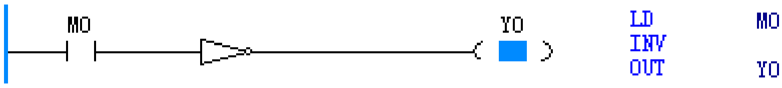
LD	M0
OR	M1
LD	M2
AND	M3
LDI	M4
AND	M5
ORB	
OR	M6
ANB	
OR	M7
OUT	Y0

5.8 INV : 取反指令

助记符、名称	功能	梯形图	步长
INV 取反	对运算结果取反		1 步

指令说明 将 INV 指令之前的逻辑运算结果取反。

使用示例

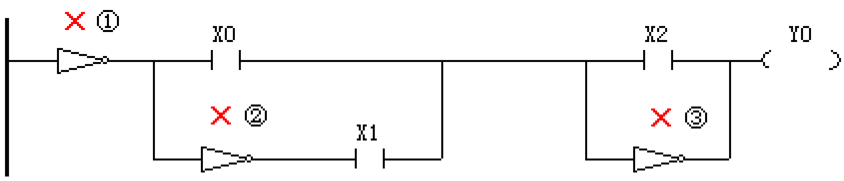


当 M0=OFF 时，将 INV 指令前的逻辑运算结果取反，此时 Y0 输出 ON。

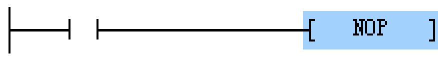
注意事项

1．在梯形图中 INV 指令不能直接连接左母线（下图①处）。
2．INV 指令不可作为并联支路中第一个指令使用（下图②处）。
3．在梯形图中 INV 指令应和触点元件串联使用，不能和其它触点元件并联使用（下图③处）。

以下是 INV 指令在梯形图中错误使用的示例：



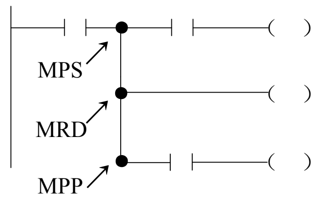
5.9 NOP : 空操作指令

助记符、名称	功能	梯形图	步长
NOP 空操作	无动作		1 步

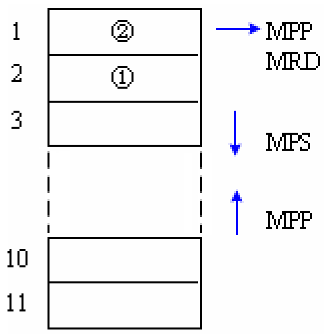
指令说明 NOP 指令为空操作指令，无实际动作，仅占程序步，但对运算结果没有影响。使用 NOP 指令可以改变程序的扫描周期。

注意事项 在梯形图中，该指令不能直接连接左母线。

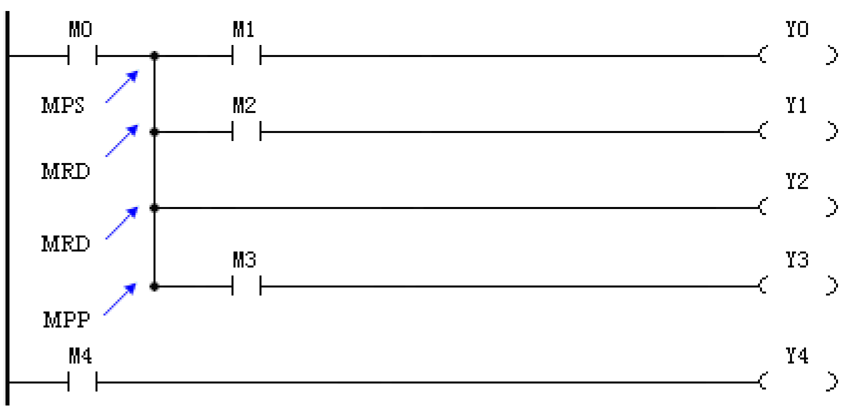
5.10 MPS , MRD , MPP 指令

助记符 、名称	功能	梯形图和适应元件	步长
MPS 进栈	运算存储		1步
MRD 读栈	存储读出		1步
MPP 出栈	存储读出与复位		1步

指令说明



1. 使用一次 MPS 指令，就将此时刻的运算结果送入栈的第一单元存储。再使用 MPS 指令，又将该时刻的运算结果送入栈的第一单元存储，而将先前送入存储的数据依次移到栈的下一单元。
2. MRD 指令是读出最上端所存的最新数据的专用指令，栈内的数据不发生移动。
3. 使用 MPP 指令，各数据按顺序向上移动，将最上端的数据读出，同时该数据就从栈中消失。
4. MPS、MRD、MPP 指令都没有操作数。



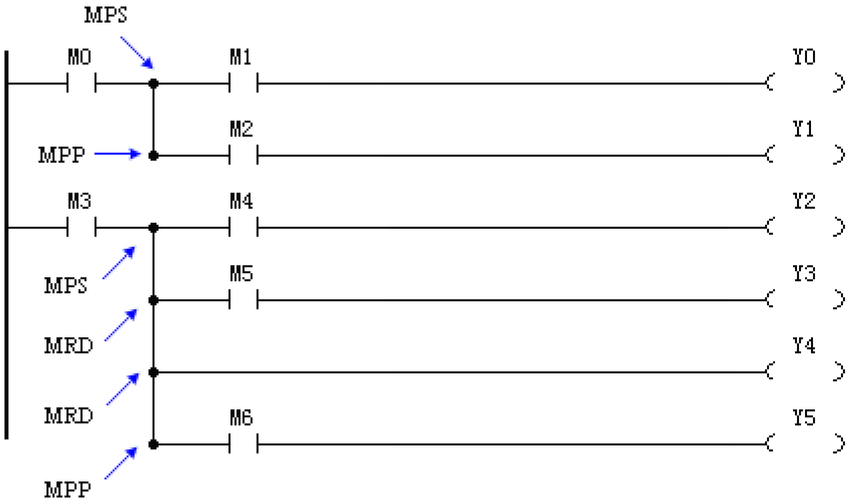

```
LD    M0
MPS
AND   M1
OUT   Y0
MRD
AND   M2
OUT   Y1
MRD
OUT   Y2
MPP
AND   M3
OUT   Y3
LD    M4
OUT   Y4
```

1. MPS、MRD、MPP 指令是进行上图所示的分支多重输出回路编程用的指令。利用 MPS 指令存储得出运算中间结果，然后驱动 Y0，用 MRD 指令将该存储读出，再驱动输出 Y1。最终输出回路以 MPP 指令替代 MRD 指令，从而在读出上述存储的同时将它复位。
2. 从上面的梯形图中对构成堆栈的三个指令的分析可知，最简单的分支即两个分支，可只有 MPS 和 MPP 构成；而三个以上的分支，则需通过反复调用 MRD 指令完成（参照上面梯形图）。也就是说，一组堆栈指令中，有且只有一个 MPS 和一个 MPP，但是可以没有或有多个 MRD。
3. MPS 指令可重复使用，但在一个梯形图网络中禁止 连续 8 次以上使用 MPS 指令（中间无 MPP 指令），否则将导致输出栈的溢出。
4. 上面梯形图中连续使用了两个 MRD 指令，目的是为了说明该指令只是读存储结果，而不影响存储结果；在执行了 MPP 后，就结束了堆栈指令，不再与 M0 的状态相关，如上面梯形图中 Y4 的状态只受 M4 控制。
5. 分支结构和并联输出结构梯形图的本质区别在于：分支结构中，分支点与输出点之间串联有触点（参照

上面梯形图），而不单纯是输出线圈。

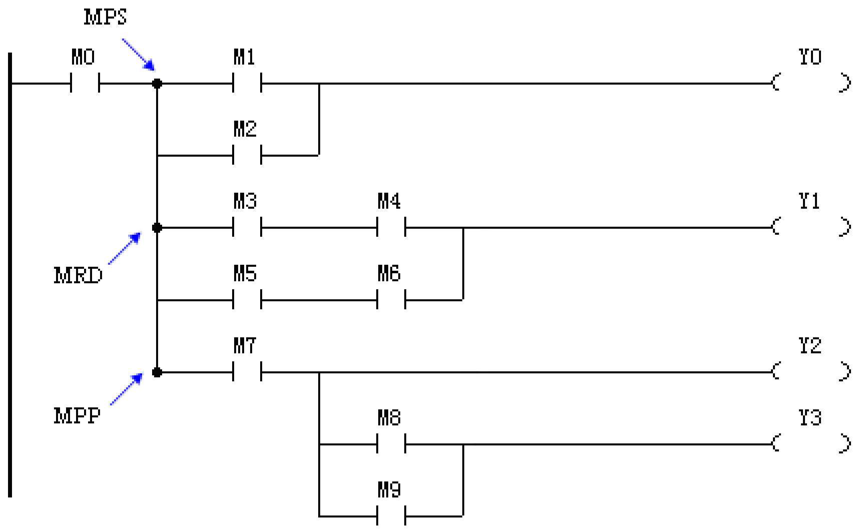
使用示例

示例 1．一段堆栈



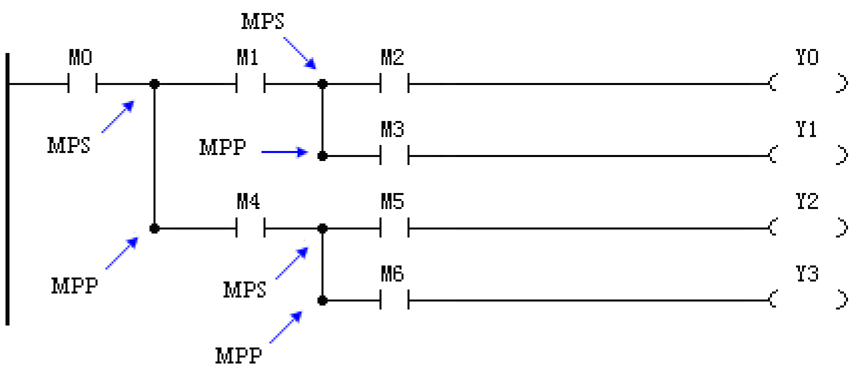

```
LD    M0
MPS
AND   M1
OUT   Y0
MPP
AND   M2
OUT   Y1
LD    M3
MPS
AND   M4
OUT   Y2
MRD
AND   M5
OUT   Y3
MRD
OUT   Y4
MPP
AND   M6
OUT   Y5
```

示例 2．堆栈与 ANB、ORB 指令并用



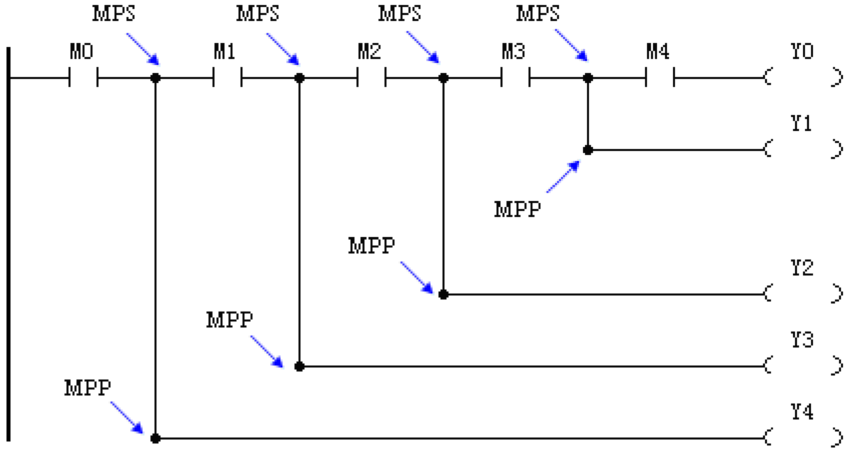

```
LD    M0
MPS
LD    M1
OR    M2
ANB
OUT   Y0
MRD
LD    M3
AND   M4
LD    M5
AND   M6
ORB
ANB
OUT   Y1
MPP
AND   M7
OUT   Y2
LD    M8
OR    M9
ANB
OUT   Y3
```

示例 3．两段堆栈



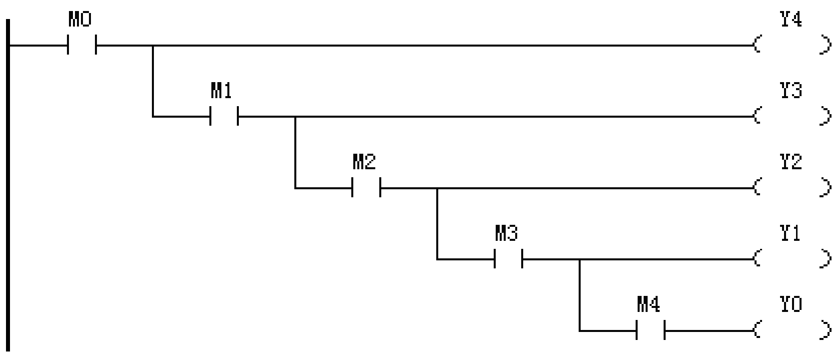

```
LD    M0
MPS
AND   M1
MPS
AND   M2
OUT   Y0
MPP
AND   M3
OUT   Y1
MPP
AND   M4
MPS
AND   M5
OUT   Y2
MPP
AND   M6
OUT   Y3
```

示例 4. 四段堆栈




```
LD    M0
MPS
AND   M1
MPS
AND   M2
MPS
AND   M3
MPS
AND   M4
OUT   Y0
MPP
OUT   Y1
MPP
OUT   Y2
MPP
OUT   Y3
MPP
OUT   Y4
```

图一

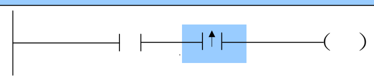
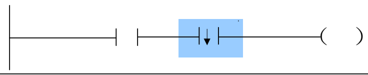



```
LD    M0
OUT   Y4
AND   M1
OUT   Y3
AND   M2
OUT   Y2
AND   M3
OUT   Y1
AND   M4
OUT   Y0
```

图二

图一所示的回路需要连续使用四次 MPS 指令。但是，如果改用图二所示的回路，则不必采用 MPS 指令，编程也很方便。

5.11 EU , ED 指令

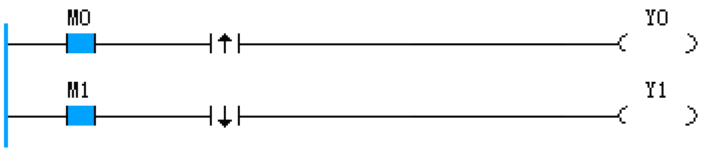
助记符 、名称	功能	梯形图	步长
EU 上升沿检测	上升沿检出运算		2步
ED 下降沿检测	下降沿检出运算		2步

指令说明

1. 比较本次扫描状态与上次扫描状态的变化。有上升沿变化（OFF → ON）时，EU 指令在本扫描周期内输出有效。
2. 比较本次扫描状态与上次扫描状态的变化。有下降沿变化（ON → OFF）时，ED 指令在本扫描周期内输出有效。

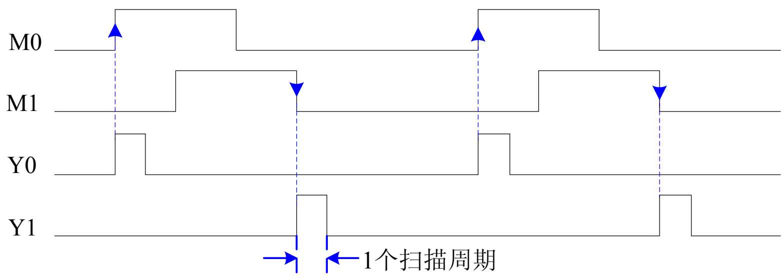
使用示例

示例 1．输出有效。

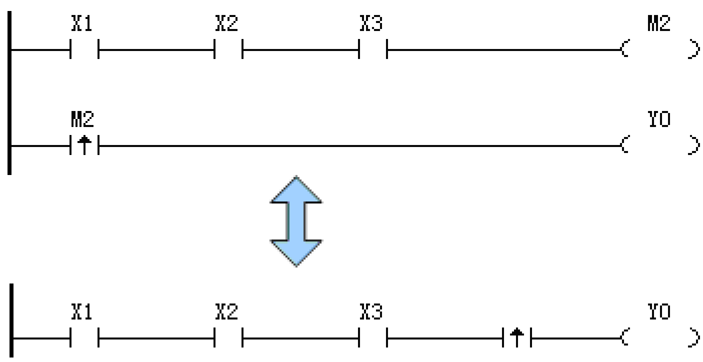



```
LD   M0
EU
OUT  Y0
LD   M1
ED
OUT  Y1
```

1．当 M0 由 OFF → ON 变化时（上升沿），EU 指令检测到上升沿变化，使得 Y0 接通一个扫描周期。
2．当 M1 由 ON → OFF 变化时（下降沿），ED 指令检测到下降沿变化，使得 Y1 接通一个扫描周期。



示例 2.
使用 EU 指令可简化程序，如下图：



注意事项

1．上升沿检测指令或下降沿检测指令应与其它触点元件串联使用，不能和其它触点元件并联使用。
2．上升沿检测触点或下降沿检测触点指令不能直接连接左母线。

5.12 TON : 接通延时计时指令

	TON		D S	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	5步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
D	INT																			●					
S	INT										●	●	●	●	●	●	●	●	＊	●	●	●	●	●	

＊ 该指令使用 C 元件时，只能使用 C0 ～ C199。

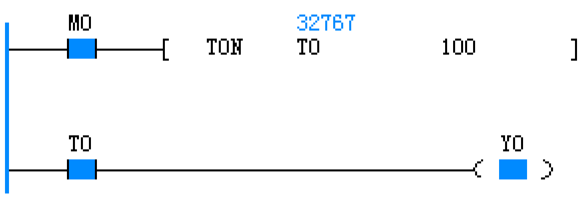
指令说明

├──┤ ├──[TON D S]

D：指定 T 元件　　**S**：预设值

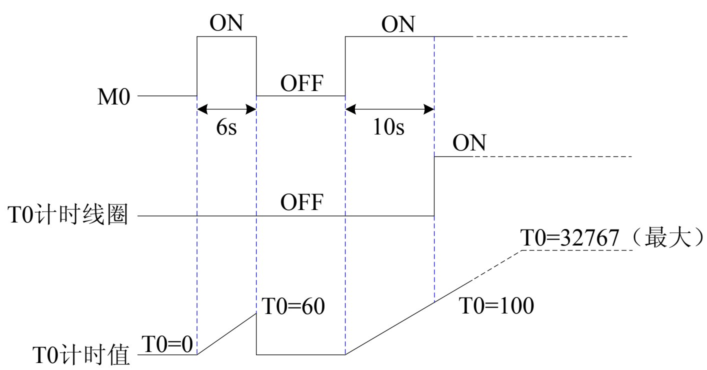
1. 定时器的实际设置时间 = 计时单位 × 预设值
2. 定时器启动后，所指定的 T 元件开始计时（计时值累加）。当计时值≥预设值时，所指定的 T 元件的计时线圈置为 ON；当计时值 =32,767 后，计时值将保持为 32,767 不变。
3. 当没有计时输入信号时，定时器停止计时，计时值清零，计时线圈被复位为 OFF。

使用示例




```
LD    M0
TON   T0 100
LD    T0
OUT   Y0
```

示例时序图：



当 M0 为 ON 时，定时器以 100ms 为单位开始计时，当计时值≥ 100 时，计时线圈 T0 置为 ON；当计时值 =32,767 后，计时值保持为 32,767 不变。

5.13 TOF : 断开延时计时指令

	TOF		D　　S	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	5步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
D	INT																			●					
S	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●	

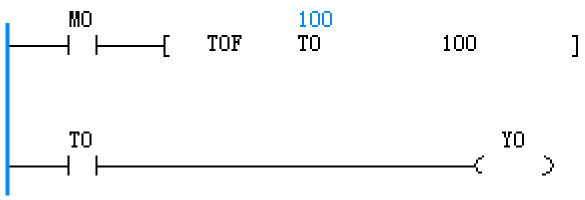
* 该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

D：指定 T 元件　　　　　　**S**：预设值

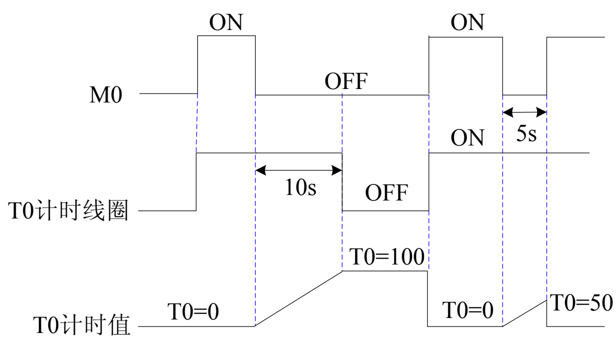
1. 定时器的实际设置时间 = 计时单位 × 预设值
2. 当定时器启动时，T 元件的计时线圈输出为 ON，同时计时值清零；当计时使能信号由 ON → OFF 时，指定定时器开始计时，直到计时值等于预设值，所指定的 T 元件的计时线圈输出为 OFF，此后计时值将保持为预设值不再变化。
3. 若定时器未启动，即使输入信号为 OFF 也不计时。

使用示例




```
LD    M0
TOF   T0 100
LD    T0
OUT   Y0
```

示例时序图：



当 M0 为 ON 时，计时线圈立即输出为 ON；当 M0 由 ON → OFF 变化（下降沿）时，定时器 T0 以 100ms 为单位开始计时，当计时值 =100 时，计时线圈 T0 被复位为 OFF，计时值保持为预设值不再变化。

5.14 TMON : 不重触发单稳计时指令

	TMON		D S	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	5步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
D	INT																			●					
S	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●	

＊ 该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

TMON D S

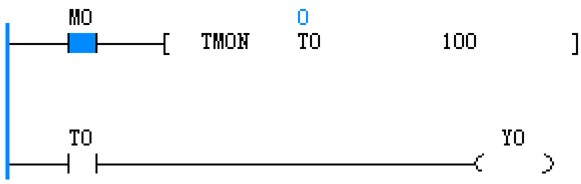
D：指定 T 元件　　　　**S**：预设值

1．定时器的实际设置时间 = 计时单位 × 预设值。

2．当定时器的计时使能信号由 OFF → ON 变化（上升沿）时，所指定的 T 元件开始计时（计时值累加），计时线圈置为 ON。当计时值 = 预设值时，计时值清零，计时线圈被复位为 OFF。

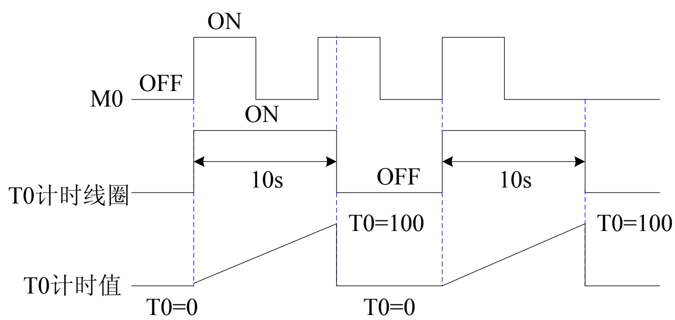
3．在计时状态下，无论计时脉冲输入信号如何变化，定时器保持计时，计时线圈输出保持为 ON。

使用示例




```
LD    M0
TMON  T0 100
LD    T0
OUT   Y0
```

示例时序图：



当 M0 由 OFF → ON 变化时，计时线圈 T0 置为 ON，定时器以 100ms 为单位开始计时，当计时值 =100（预设值）时，计时值清为零，计时线圈 T0 被复位为 OFF。

5.15 TONR：记忆型接通延时计时指令

	TONR		D S	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	5步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
D	INT																			●					
S	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●	

* 该指令使用 C 元件时，只能使用 C0 ～ C199。

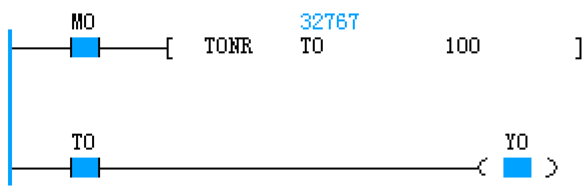
指令说明

TONR D S]

D：指定 T 元件　　**S**：预设值

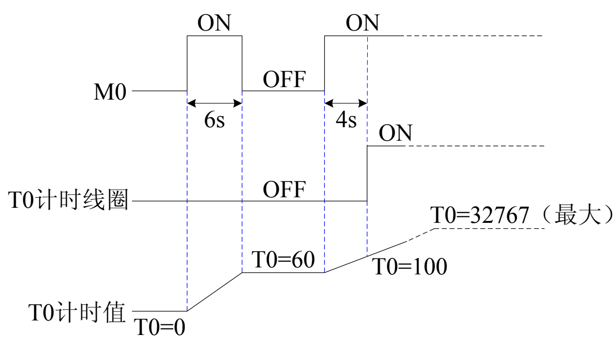
1．定时器的实际设置时间 = 计时单位 × 预设值。
2．定时器启动后，所指定的 T 元件开始计时（计时值累加）。当计时值≥预设值时，所指定的 T 元件的计时线圈置为 ON；当计时值 =32,767 后，计时值将保持为 32,767 不变。
3．当没有计时使能信号时，定时器停止计时，计时线圈与计时值保持当前状态不变。当再次检测到输入信号时，定时器继续计时。
4．若要清除计时值请使用 RST 指令。

使用示例




```
LD    M0
TONR  T0 100
LD    T0
OUT   Y0
```

示例时序图：



当 M0 为 ON 时，定时器以 100ms 为单位开始计时，当计时值≥ 100 时，计时线圈 T0 置为 ON；当计时值等于 32,767 后，计时值保持为 32,767 不变。

5.16 CTU : 16 位计数器增计数指令

	CTU		D　　S	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	6步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
D	INT																		*						
S	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●	

* 该指令使用 C 元件时，只能使用 C0 ～ C199。

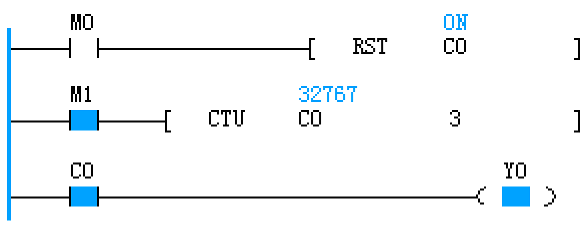
指令说明

├─┤├─[　CTU　　D　　S　]

D：指定 C 元件　　　**S**：预设值

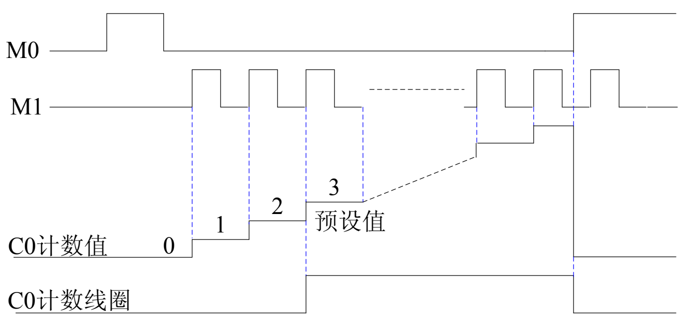
1. 当计数器的输入信号由 OFF → ON 变化时，指定的 16 位计数器计数值增 1。
2. 当计数值≥预设值时，所指定的 C 元件的线圈置为 ON；当计数值等于 32,767 后，再检测到计数器的输入信号由 OFF → ON 变化时，计数值将保持 32,767 不变。
3. 当没有检测到计数器的输入信号由 OFF → ON 变化时，计数器线圈与计数值保持不变，重新检测到计数器的输入信号由 OFF → ON 变化后会继续累计计数。若要重新从零开始计数，请使用 RST 指令将计数值清零。
4. 若使用 MXProgrammer 的写入功能或通过 MOV 指令等将一个大于等于预设值的数值传送到指定计数器的当前值寄存器中，在下次检测到计数器的输入信号由 OFF → ON 变化时，计数线圈置为 ON，同时计数器当前值增 1。

使用示例




```
LD   M0
RST  C0
LD   M1
CTU  C0 3
LD   C0
OUT  Y0
```

示例时序图：



1. 当 M0 为 ON 时，执行 RST 指令，C0 的当前值清为零，计数线圈被复位为 OFF。
2. 当 M1 由 OFF → ON 变化时，计数器当前值增 1。
3. 当计数器 C0 的计数值到达预设值 3 时，计数线圈置为 ON。如果再检测到计数器的输入信号由 OFF → ON 变化后会继续计数，当计数值等于 32,767 后保持不变。

注意事项

当预设值等于零时，计数器的输入信号由 OFF → ON 变化，计数值增 1，计数线圈置为 ON。

5.17 CTR : 16 位计数器循环计数指令

	CTR		D S	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	6步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
D	INT																		*						
S	INT										•	•	•	•	•	•	•	•	*	•	•	•	•	•	

* 该指令使用 C 元件时，只能使用 C0 ～ C199。

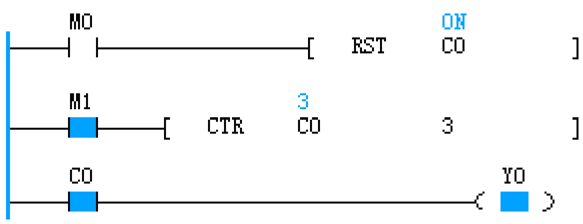
指令说明

├──┤ ├──[CTR D S]

D：指定 C 元件 **S**：预设值

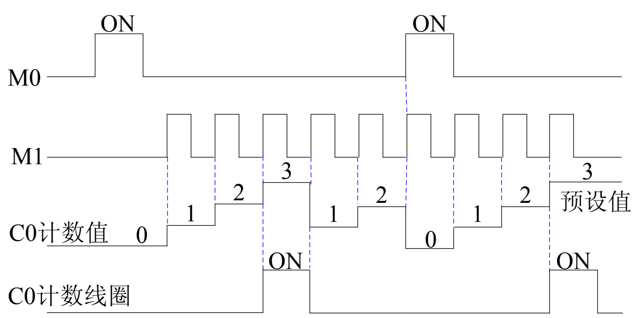
1. 当计数器的输入信号由 OFF → ON 变化时，指定的 16 位计数器计数值增 1。
2. 当计数值 = 预设值时，所指定的 C 元件的计数线圈置为 ON；如果再检测到计数器的输入信号由 OFF → ON 变化则计数值变为 1，同时计数线圈被复位为 OFF。
3. 当没有检测到计数器的输入信号由 OFF → ON 变化时，计数线圈与计数值保持不变，重新检测到计数器的输入信号由 OFF → ON 变化后会继续累计计数。若要重新从零开始计数，请使用 RST 指令。
4. 若使用 MXProgrammer 的写入功能或通过 MOV 指令等将一个大于等于预设值的数值传送到指定计数器的当前值寄存器中，在下次检测到计数器的输入信号由 OFF → ON 变化时，计数值变为 1，计数线圈被复位为 OFF。

使用示例




```
LD   M0
RST  C0
LD   M1
CTR  C0 3
LD   C0
OUT  Y0
```

示例时序图：



1. 当 M0 为 ON 时，执行 RST 指令，C0 的当前值清零，计数线圈被复位为 OFF。
2. 当 M1 由 OFF → ON 变化时，计数器当前值增 1。
3. 当计数器 C0 计数值到达预设值 3 时，计数线圈变为 ON。再检测到计数器的输入信号由 OFF → ON 变化后计数值变为 1，计数线圈被复位为 OFF。

注意事项

当 S ≤ 0 时，在 PLC 信息中报“指令操作数数值非法错误”，指令不执行。

5.18 DCNT : 32 位增减计数指令

	DCNT		D　　　S	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	8步			相关元件	SM200 ～SM234			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
D	DINT																		＊						
S	DINT										●	●	●	●	●	●	●	●	＊		●		●	●	

＊　使用 C 元件时，D 只能使用 C200 ～ C234，S 只能使用 C200 ～ C255。

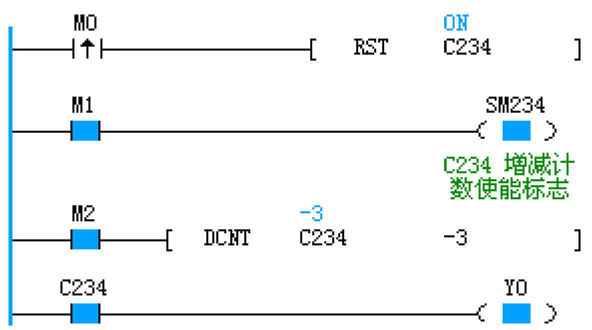
指令说明

├──┤ ├──[　DCNT　　D　　S　]

D：指定 C 元件　　　**S**：预设值

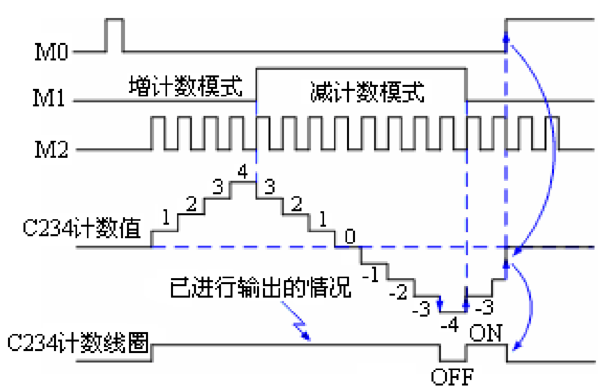
1．当计数器的输入信号由 OFF → ON 变化时，指定的 32 位计数器计数值增 1 或减 1（计数增减方向由对应 SM 标志位决定）。
2．若为增计数器，当计数值大于等于计数预设值时，计数线圈置为 ON；若为减计数器，当计数值小于计数预设值时，计数线圈复位为 OFF。
3．当计数值＝ 2,147,483,647，若再次增 1 计数时，计数值变为－ 2,147,483,648；当计数值＝－ 2,147,483,648，若再次减 1 计数时，计数值变为 2,147,483,647。
4．当没有检测到计数器的输入信号由 OFF → ON 变化时，计数线圈与计数值保持不变，重新检测到计数器的输入信号由 OFF → ON 变化后会继续累计计数。若要重新从零开始计数，请使用 RST 指令。
5．若使用 MXProgrammer 的写入功能或通过 MOV 指令等将一个大于等于预设值的数值传送到指定计数器的当前值寄存器中，在下次检测到计数器的输入信号由 OFF → ON 变化时，计数值增 1 或减 1，计数线圈置为 ON。

使用示例




```
LDP   M0
RST   C234
LD    M1
OUT   SM234
LD    M2
DCNT  C234 -3
LD    C234
OUT   Y0
```

示例时序图：



1．当 M0 为 ON 时，执行 RST 指令，C234 当前值清 0，计数线圈被复位为 OFF。
2．当 M1 为 OFF， M2 由 OFF → ON 变化时，计数器当前值由 0 → 1，计数器线圈变为 ON。
3．当 M1 为 ON 时，M2 由 OFF → ON 变化时，计数器当前值减 1。当计数器的值小于－ 3 时，计数器线圈变为 OFF。

第六章 应用指令

6.1 流程控制指令

6.1.1 CJ , CJP , LBL 指令

助记符 、名称	功能	梯形图和适用元件	步长
CJ 条件跳转	用户程序跳到编号为S的合法标号指令处执行（可缩短扫描周期）	─┤ ├──[CJ S] 常数	3步
CJP 条件跳转		─┤ ├──[CJP S] 常数	4步
LBL 跳转标号定义	无实质性操作，只为CJ标明跳转的具体位置	├────[LBL S] 常数	3步

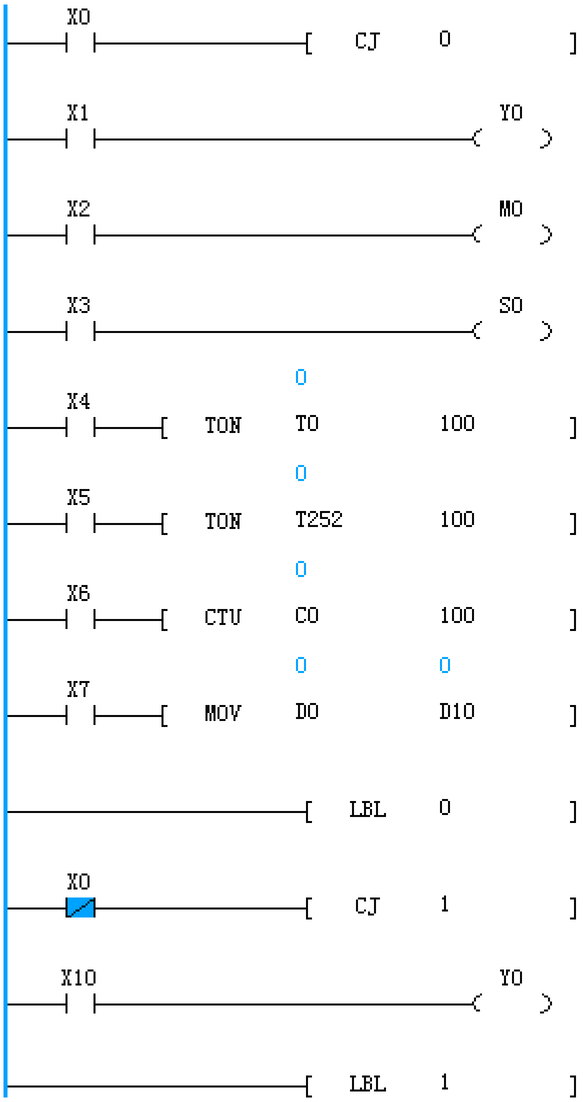
指令说明

当用户希望 PLC 不执行某一段程序以缩短扫描周期或者使用不同的条件输出双线圈时，可以用CJ 或 CJP 指令。。

使用示例

示例 1.

下段程序中 Y0 为双线圈输出，但是当 X0=OFF 时 Y0 由 X1 控制，当 X0=ON 时 Y0 由 X10 控制。

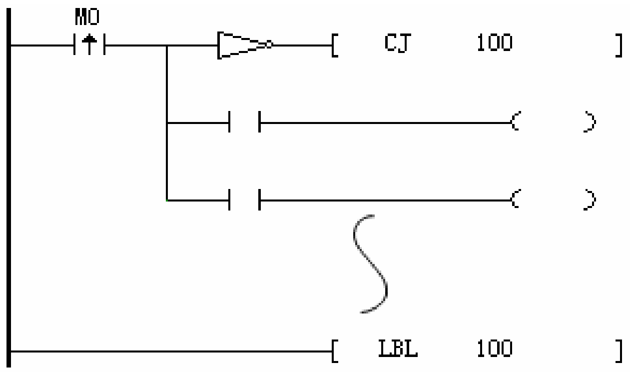



```
LD    X0
CJ    0
LD    X1
OUT   Y0
LD    X2
OUT   M0
LD    X3
OUT   S0
LD    X4
TON   T0 100
LD    X5
TON   T252 100
LD    X6
CTU   C0 100
LD    X7
MOV   D0 D10
LBL   0
LDI   X0
CJ    1
LD    X10
OUT   Y0
LBL   1
```

下列表格为上述程序在跳转过程中发生变化后产生的结果

类型	CJ 执行前接点状态	CJ 执行中接点状态	CJ 执行中输出线圈状态
Y、M、S	X1 X2 X3 OFF	X1 X2 X3 ON	Y0 M0 S0 OFF
	X1 X2 X3 ON	X1 X2 X3 OFF	Y0 M0 S0 ON
1ms、10ms、100ms 定时器	X4 X5 OFF	X4 X5 ON	定时器 T0、T252 不工作
	X4 X5 ON	X4 X5 OFF	定时器 T0、T252 继续工作
C0 ～ C234	X6 OFF	X6 ON	计数器不工作
	X6 ON	X6 OFF	计数器 C0 暂停计数，X0“OFF”后继续计数
应用指令	X7 OFF	X7 ON	应用指令不执行
	X7 ON	X7 OFF	被跳过的应用指令不执行，但是高速指令会继续工作（详见注意事项）

示例 2．CJ 使用技巧



1．当 M0 为 OFF 时，执行 CJ 指令，CJ 100 － LBL 100 之间的程序被跳过，可节省扫描周期。

2．当 M0 由 OFF → ON 变化时，CJ 100 － LBL 100 之间的程序执行一个扫描周期。

注意事项

1．CJ 指令所要跳转的标号 S（0 ≤ S ≤ 127）应是一个合法且已定义的标号，否则用户程序将不能正确通过编译。

2．在主程序、子程序、中断程序中 128 个跳转标号都可以使用且是相对独立的。

3．在同一个程序体中不允许出现两个重复定义的标号，否则用户程序将不能通过编译，但是允许多个 CJ 跳到同一个标号处。

4．相同标号的 CJ 和 LBL 指令必须同时用在同一个程序体中，不能从主程序跳到子程序或中断子程序中，同样也不能从子程序或中断子程序跳出。

5．标号 S 所指的程序若在 CJ 指令之前，会发生 WDT 逾时错误，PLC 停止运行。

6．不允许使用 CJ 指令跳转到一个 FOR － NEXT 结构中，但允许使用 CJ 从一个 FOR － NEXT 结构中跳出。

7．可以使用 CJ 指令跳入或跳出 MC － MCR 结构和 SFC 状态，但这样将破坏 MC － MCR 和 SFC 状态的逻辑，使程序复杂化，建议不要这样使用。

8．跳转指令执行时各种装置的动作情形说明：

- Y、M、S 保持跳转前的状态。
- 执行计时中的 1ms、10ms、100ms 定时器会继续工作。
- 执行计数中的高速计数器会继续计数，且输出接点正常动作。
- 一般计数器会停止计数。
- 一般应用指令不会被执行。
- 执行中的应用指令 HCNT、PLS、SPD 、PLSY、PLSR、PWM、DHSCS、DHSCR、DHSCI、DHSZ、DHSP、DHST、DHSTI 会继续执行。

6.1.2 CALL 指令

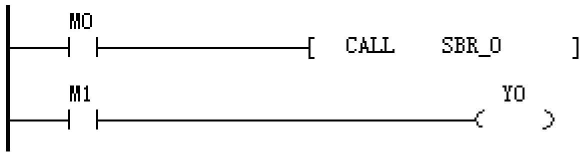
助记符 、名称	功能	梯形图	步长
CALL 用户子程序调用	调用用户指定的子程序	├──┤ ├──[CALL (子程序名) (参数1) (参数2)…]	5步

指令说明

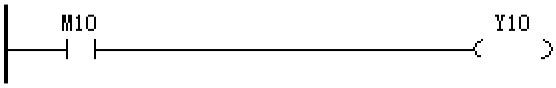
在程序中，对想要共同处理的程序进行调用的指令。可以减少程序的步数，更加有效的设计程序。

使用示例

主程序 TASK-0：



子程序 SBR-0：



1．如果 M0 为 ON，则执行子程序 SBR-0，子程序执行完成后，返回至 CALL 之后的指令继续执行。
2．子程序中可以继续使用 CALL 指令镶嵌调用子程序，整体而言子程序可做 5 层嵌套；中断服务程序只允许调用一级子程序。

注意事项

1．CALL 指令中所调用的子程序，必须在用户程序中事先定义好。当 CALL 中出现一个从未定义的子程序时，将不能通过程序编译。
2．CALL 指令中所带操作数的元件类型，应与子程序的局部变量表中定义的数据类型相匹配，否则不能通过编译。以下示例说明非法的匹配使用：
例一：SBR-1 子程序的局部变量表中，操作数的数据类型为 DINT/DWORD 型。
以下使用都是非法的：
CALL SBR-1 Z0（Z 元件不能用作数据类型为 DINT/DWORD 型）
CALL SBR-1 C199（C0 ～ C199 元件不能用作数据类型为 DINT/DWORD 型）
CALL SBR-1 K2X0（Kn 寻址 1 ≤ n ≤ 4，不能用作数据类型为 DINT/DWORD 型）
例二：SBR-1 子程序的局部变量表中，操作数的数据类型为 INT/WORD 型。
以下使用都是非法的：
CALL SBR-1 C200（C200 ～ C255 元件不能用作数据类型为 INT/WORD 型）
CALL SBR-1 K5X0（Kn 寻址 5 ≤ n ≤ 8，不能用作数据类型为 INT/WORD 型）
3．CALL 指令中所带的操作数的元件类型，应与子程序局部变量表中定义的变量类型相匹配，否则不能通过编译。以下示例说明非法的匹配使用：
例：SBR-1 子程序的局部变量表中，操作数的变量类型为 OUT 或 IN_OUT 型。
以下使用都是非法的：
CALL SBR-1 321（常数不可改变，所以与 OUT 或 IN_OUT 型操作数不匹配）
CALL SBR-1 K4X0（K4X0 只具有只读性质，所以与 OUT 或 IN_OUT 型操作数不匹配）
CALL SBR-1 SD0（SD0 只具有只读性质，所以与 OUT 或 IN_OUT 型操作数不匹配）
4．CALL 指令中所带的操作数个数，应与子程序局部变量表相配，否则不能通过编译。

6.1.3 CFEND , CSRET , CIRET 指令

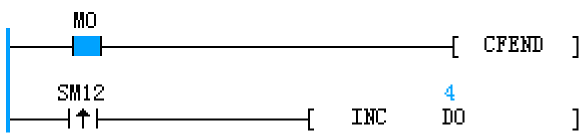
助记符、名称	功能	梯形图	步长
CFEND 任务条件返回	程序扫描到此处时返回	├─┤ ├─[CFEND]	1步
CSRET 用户子程序条件返回	返回上一级子程序	├─┤ ├─[CSRET]	1步
CIRET 用户中断服务条件返回	退出正在执行的中断子程序	├─┤ ├─[CIRET]	1步

指令说明

1. CFEND 指令执行时，主程序从当前扫描周期中返回系统，其后的主程序中的指令不被执行。需要注意的是该指令必须在主程序中使用，否则不能通过编译。
2. CSRET 指令执行时，系统退出当前正在执行的子程序，返回上一级程序。
3. CIRET 指令执行时，系统提前退出正在执行的中断子程序。

使用示例

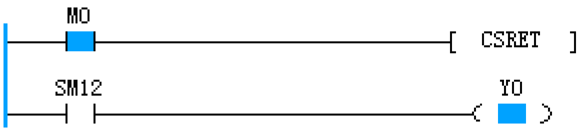
示例 1.




```
LD     M0
CFEND
LDP    SM12
INC    D0
```

当 M0 为 OFF 时，CFEND 指令不执行，其后的指令执行，D0 中的数值每秒加 1。当 M0 为 ON 时，CFEND 指令执行，其后的指令不再执行，D0 中的数值不再增加。

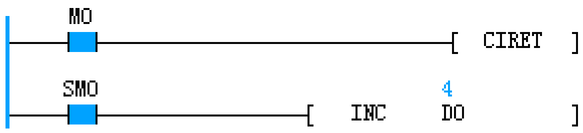
示例 2.




```
LD     M0
CSRET
LD     SM12
OUT    Y0
```

以上为子程序中的一段程序，当 M0 为 OFF 时，Y0 以 1 秒为周期闪烁。当 M0 为 ON 时，系统会退出当前正在执行的子程序，Y0 会一直保持上周期的状态（ON 或 OFF）。

示例 3.




```
LD     M0
CIRET
LD     SM0
INC    D0
```

以上为定时中断服务子程序中的程序，当 M0 为 OFF 时，中断每执行一次，D0 中的数值增 1。当 M0 为 ON 时，程序每当执行到此处时都会退出当前正在执行的中断子程序，故 D0 中的数值不会增加。

6.1.4 FOR , NEXT 指令

助记符、名称	功能	梯形图和适用元件		步长
FOR 循环开始	执行循环程序	[FOR S]	常数 KnX KnY KnS KnM KnLM KnSM D SD C T V Z	3步
NEXT 循环返回	循环程序返回	[NEXT]	无	1步

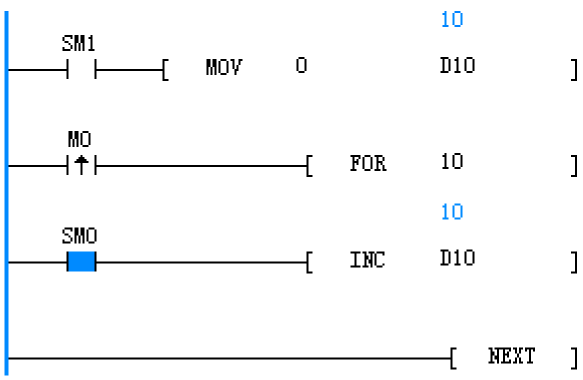
指令说明

1. FOR 指令与 NEXT 指令匹配成一个 FOR — NEXT 结构。在一个程序体中必须成对使用，若 NEXT 指令编写在 FOR 之前，或 FOR 指令无对应的 NEXT 指令，用户程序都不能正确通过编译。
2. 当 FOR 指令的驱动条件满足且循环次数（S）大于零时，FOR — NEXT 结构中间的指令被连续循环执行 S 次。当循环执行完 S 次后，继续执行 FOR — NEXT 结构后的指令。若 FOR 指令的驱动条件不满足或循环次数（S）小于等于零时，FOR — NEXT 结构中间的指令不被执行，程序直接跳转到 FOR — NEXT 结构后的指令继续执行。
3. 循环次数 S 在 1 ～ 32,767 之间时有效。例如 S=4 时，FOR — NEXT 循环执行 4 次。

使用示例

示例 1. （无嵌套）

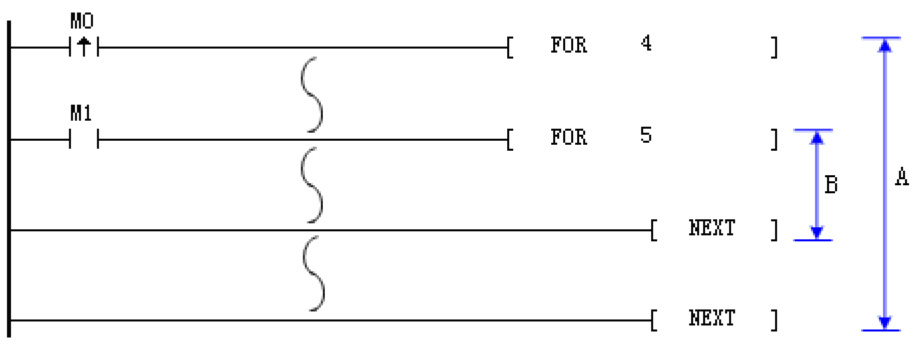
执行 FOR — NEXT 循环程序前，D10=0，M0=OFF。当 M0 由 OFF → ON 变化时，FOR — NEXT 循 环程序被执行，其内的指令被连续执行 10 次，每执行一次，D10 中的数值都会加 1，循环结束后，D10=10。




```
LD    SM1
MOV   0 D10
LDP   M0
FOR   10
LD    SM0
INC   D10
NEXT
```

示例 2. （有嵌套）

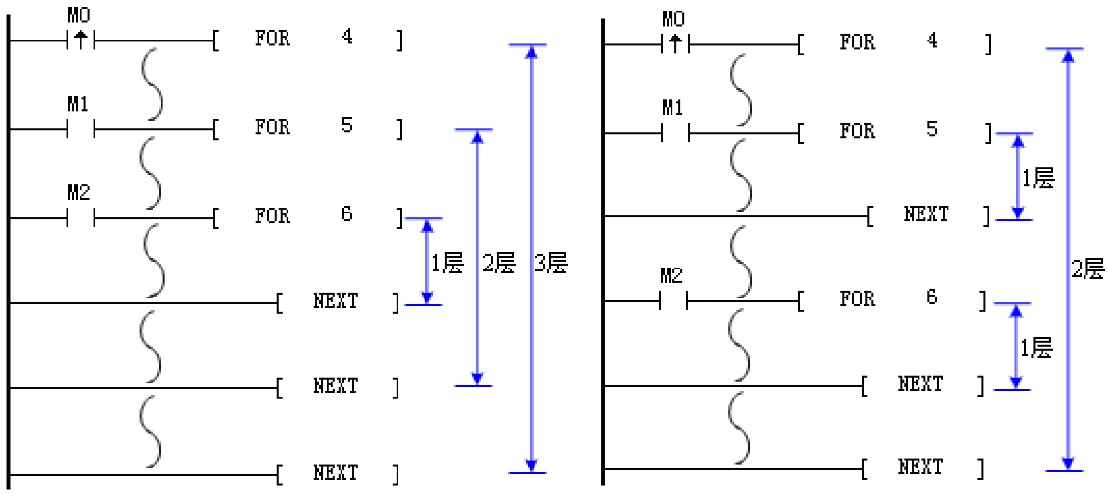
FOR — NEXT 指令最多支持 8 层结构嵌套。但应当注意的是，循环次数多时会延长 PLC 的扫描周期，可能出现看门狗超时。下图是两级嵌套程序。



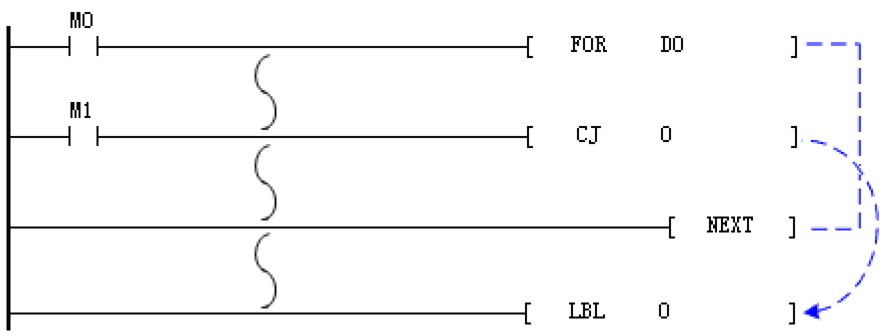
A 程序执行 4 次以后再继续执行 NEXT 指令以后的程序。而 A 程序每执行一次 B 程序会执行 5 次，所以 B 程序合计共执行 4×5=20 次。

注意事项

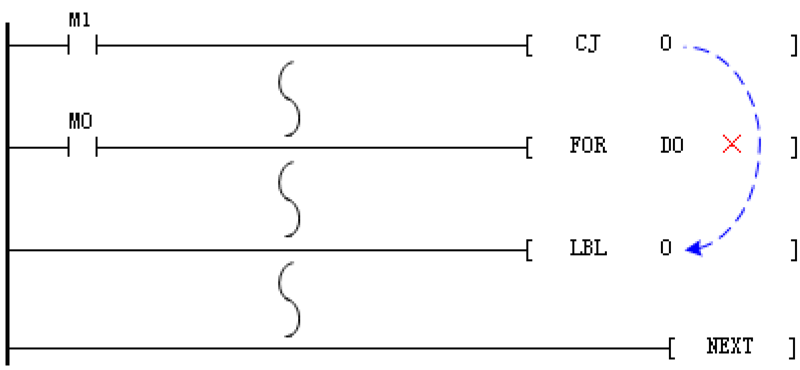
1. 在 FOR — NEXT 指令之间，将 FOR — NEXT 指令嵌套编程时，最多支持 8 层结构嵌套。



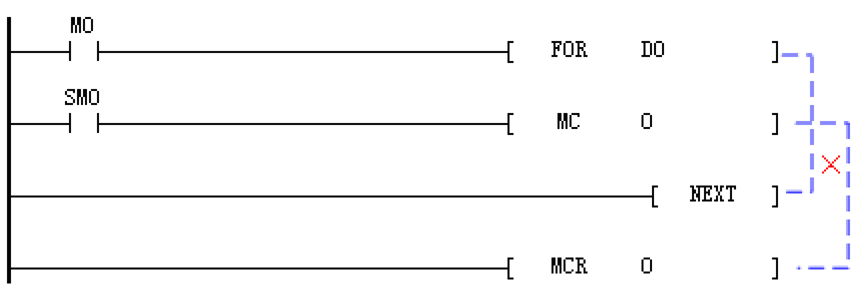
2．可以在循环体内使用条件跳转指令（CJ）跳出循环体，从而达到提前终止循环体执行的目的。



3．禁止用户使用跳转指令（CJ）跳入一个循环体，如下面梯形图将不能正确通过编译。



4．禁止 MC － MCR 结构体和 FOR － NEXT 结构体的交叉，如下面梯形图将不能正确通过编译。



6.1.5 MC , MCR 指令

助记符、名称	功能	梯形图和适用元件	步长
MC 主控	主控程序开始	[MC S] 常数	3步
MCR 主控清除	主控程序结束	[MCR S] 常数	3步

指令说明

1．MC 与 MCR 指令匹配成一个 MC － MCR 结构。MC 指令代表着一个 MC － MCR 结构的开始，MC － MCR 指令的操作数为 0 ～ 7 之间的一个常数。MCR 代表着一个 MC － MCR 结构的结束。
2．当主控 MC 指令的驱动条件满足时，执行 MC － MCR 结构中的指令。当主控 MC 指令的驱动条件不满足时，不执行 MC － MCR 结构中的指令，程序直接跳转到该结构后执行，并且该结构中的 OUT、TON、PWM、 PLSY 以及定位指令的目的操作数将被清除，沿指令状态将初始化，TOF 开始计时，TMON 保持计时，TONR 计时值保持不变， HCNT、PLSR、SPD、DHSCS、DHSCI、DHSCR、DHSZ、DHST、DHSP、DHSTI 所对应的计数器保持当前数值，不再计数，当主控 MC 指令的驱动条件再次满足时，在当前值的基础上继续计数（SPD 对应的数值重新开始计数）。

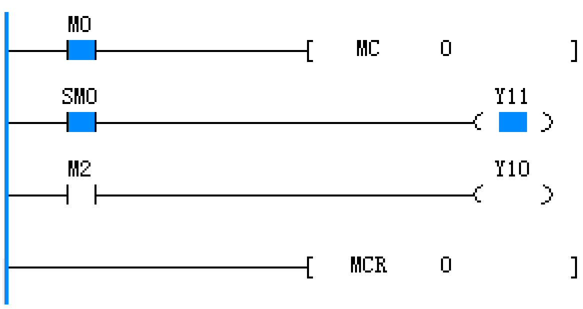
主控指令 MC － MCR 由执行到停止时各指令的动作现象

指令区分	说明
TON	计时值归零，线圈失电，触点不动作
TONR	线圈失电，计时值及触点保持目前状态
TOF、TMON	正常工作
计数器	线圈失电，计数值及触点保持目前状态
OUT 指令驱动的线圈	全部清除
SET，RST 指令驱动的元件	保持目前状态
应用指令	全部不动作，但 FOR － NEXT 循环回路仍会来回执行 N 次，但 FOR － NEXT 之间的指令动作方式与 MC － MCR 之间其他指令相同

3．MCR 为主控结束指令，置于主控程序最后，在 MCR 指令之前不可有触点指令。
4．在执行主控程序时，可以再次使用 MC － MCR 指令，这种用法称为嵌套。

使用示例

示例 1．（无嵌套）



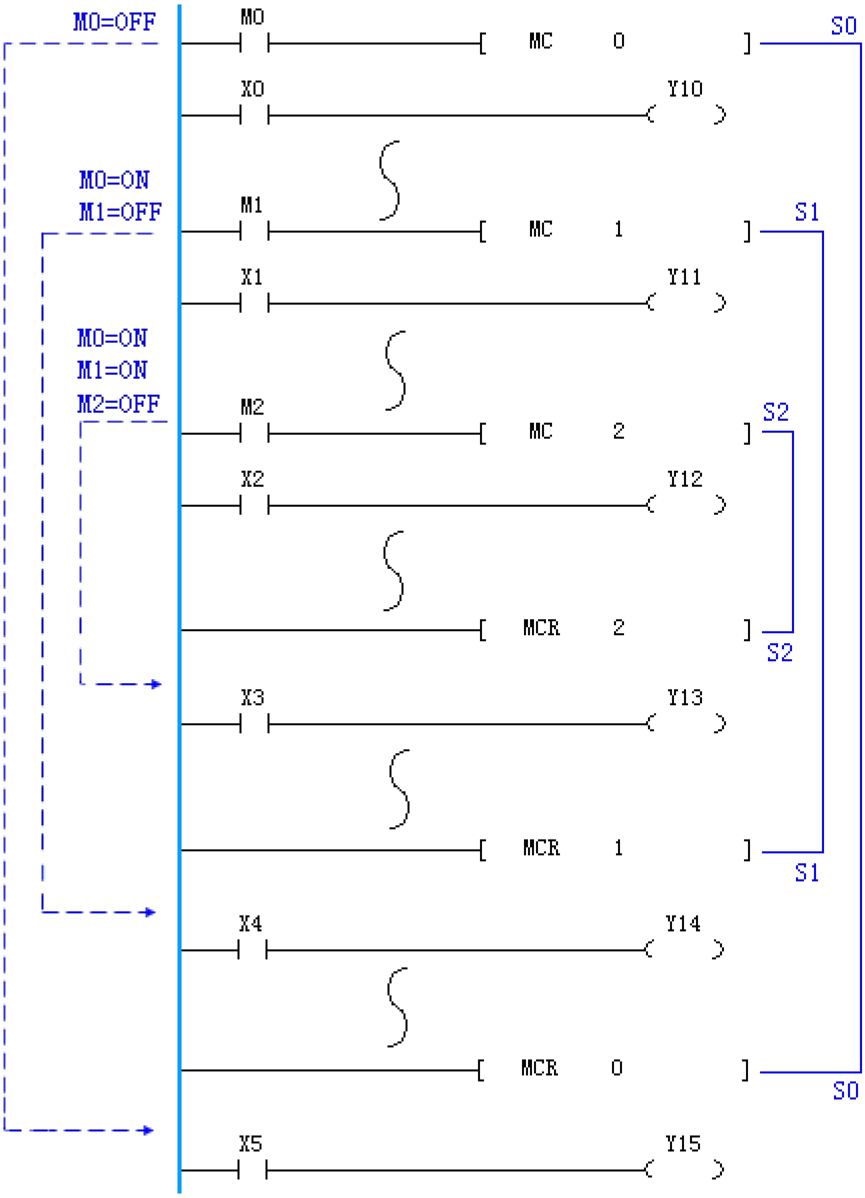

```
LD    M0
MC    0
LD    SM0
OUT   Y11
LD    M2
OUT   Y10
MCR   0
```

当 M0 为 ON 时，执行 MC 0 － MCR 0 结构内的指令，Y11 ＝ ON。
当 M0 ＝ OFF 时，不执行 MC 0 － MCR 0 结构内的指令，同时 Y11 复位。

示例 2．（有嵌套）

1. 在 MC － MCR 嵌套使用时，嵌套级按操作数 S 的标号依次增大（S0 → S1 → S2 → S3 → S4 → S5 → S6 → S7）。再将该指令返回时，采用 MCR 指令，则从大的嵌套级开始消除（S7 → S6 → S5 → S4 → S3 → S2 → S1 → S0）。

2．嵌套级最大可为 8 级。



通过 MCR 0，返回到最初的状态。所以 Y15 的接通 / 断开状态只取决于 X5 的 ON/ OFF 状态，而与 M0、M1、M2 状态无关。

6.1.6 WDT 指令

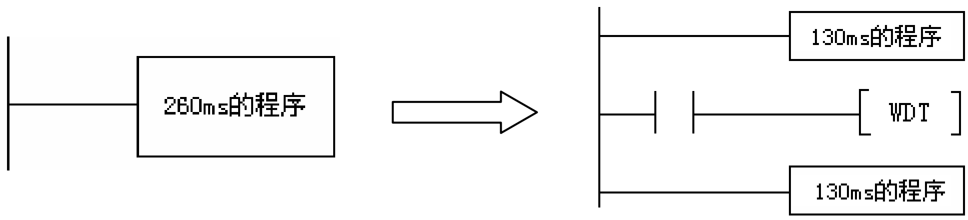
助记符、名称	功能	梯形图	步长
WDT 用户程序看门狗清零	清除看门狗时间	├──┤ ├──[WDT]	1步

指令说明

WDT 指令执行时将看门狗的计时值归零，看门狗将重新开始计时。在一段程序中，如果扫描周期大于看门狗时间，PLC 会停止用户程序，ERROR 灯常亮，输出按系统配置进行输出。像这样运算周期较长的情况，则可将 WDT 指令插入到合适的程序步中刷新看门狗时间，从而避免错误的出现。

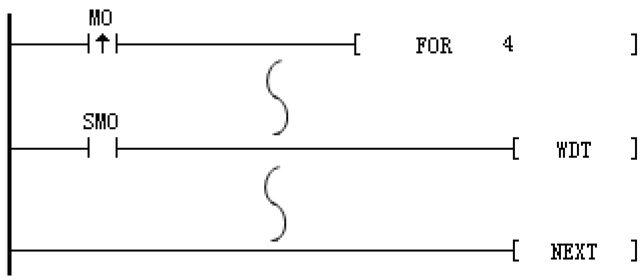
使用示例

示例 1．出现运算周期长的故障时



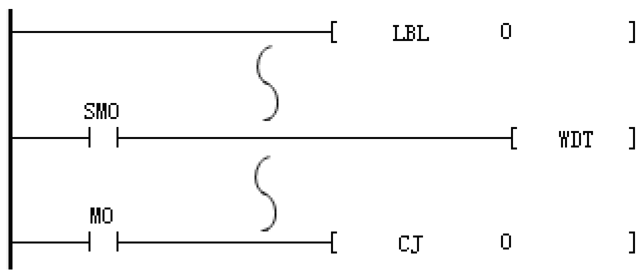
如上图所示，若看门狗时间设置采用默认值 200ms，由于程序扫描周期不能大于看门狗定时时间，所以可用 WDT 指令将一段扫描周期为 260ms 的程序分为两部分，使得这两部分的扫描时间都低于看门狗的时间。

示例 2．FOR － NEXT 指令的重复次数较多的情况



如上图所示，若 FOR － NEXT 循环程序执行时间超过看门狗定时时间，则可将 WDT 指令插入到循环程序中。

示例 3．条件跳转指令 CJ 在它对应的标号之后



如上图所示，条件跳转指令 CJ 若在它对应的标号之后，则可能因连续反复跳转使他们之间的程序被反复执行，总的执行时间就可能超过看门狗时间，为了避免这种情况，可在 CJ 指令和对应标号之间插入 WDT 指令。

注意事项

1．看门狗定时器错误的情况

（1）在连接了较多数量的特殊扩展设备 (定位、模拟量等) 的系统中，可编程控制器运行时被执行的缓冲存储区的初始化时间会变长，运算时间会延长，因此有时会出现看门狗定时器错误。

（2）较多的 FROM/TO 指令同时驱动时，传送多个缓冲存储区的时候，运算时间会延长，因此有时会出现看门狗定时器错误。

（3）编写多个高速计数器同时对高频脉冲进行计数时，运算时间会延长，因此有时会出现看门狗定时器错误。

2．更改看门狗定时器时间的方法看门狗时间可在“系统组态→工程设置→时间设定”里设置，其默认值为 200ms。

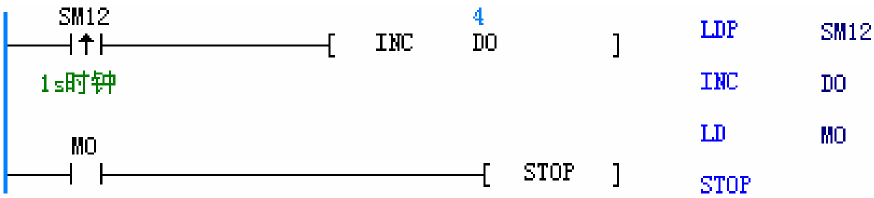
6.1.7 STOP 指令

助记符、名称	功能	梯形图	步长
STOP 用户程序停止	停止PLC	[STOP]	1步

指令说明

立即停止用户程序的执行。

使用示例



1. 当 M0 为 OFF 时，程序正常执行，D0 中的数值每秒钟增 1。
2. 当 M0 为 ON 时，程序立即停止，PLC 由 RUN → STOP 状态。

6.1.8 EI , DI 指令

助记符 、名称	功能	梯形图	步长
EI 中断使能	使能总中断	[EI]	1步
DI 中断禁止	关闭总中断	[DI]	1步

指令说明

1. 当 EI 指令的驱动条件满足时，打开总中断。中断请求将被允许加入到中断请求队列中，等待系统响应。
2. 当 DI 指令的驱动条件满足时，关闭总中断。如果中断请求队列中仍有中断请求未处理完时，剩余的中断请求仍然要被响应，但新中断事件将不能产生中断请求。
3. EI/DI 是使能 / 禁止中断总开关，当开关打开时，相应的中断使能标志位置位，中断程序才有效。（请参见第九章中断服务程序使用指南）

使用示例

1. 通过对 DI 指令的编程，可以设定中断禁止区间，在不需要中断禁止区间时，仅对 EI 指令编程，不一定必须对 DI 指令编程。
2. 当程序扫描到 EI 指令时，允许触发中断；扫描到 DI 指令时，禁止触发中断。

6.2 SFC 指令

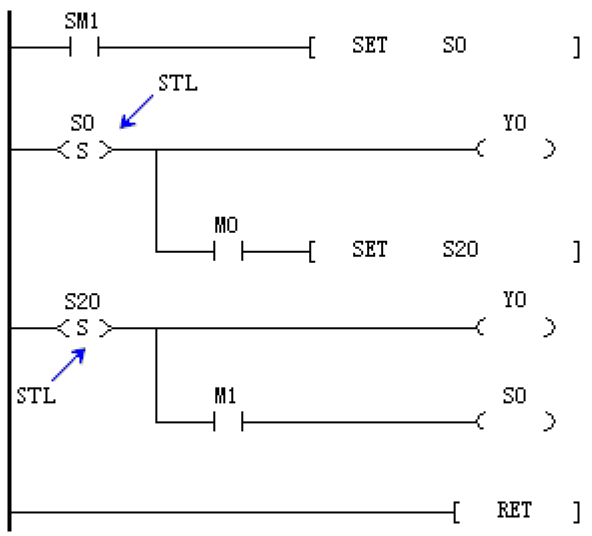
STL , RET 指令

助记符 、名称	功能	梯形图	步长
STL SFC 状态装载	步进状态开始	<S> ( ) [SET S]	4步
RET SFC 程序结束	步进程序结束	[RET]	1步

指令说明

1. STL 标志一个步进状态的开始。
2. RET 标志一段顺序功能图程序的结束。
   请求仍然要被响应，但新中断事件将不能产生中断请求。
3. 如果该步进状态有效（ON），其内置指令执行。
4. 如果该步进状态无效，其内置指令不执行，步进中的指令也不扫描，可缩短扫描周期。
5. 如果该步进状态由有效变为无效，其内置指令不被执行，并且该结构中的 OUT、TON、PWM、PLSY 以及定位指令的目的操作数将被清除，沿指令状态将初始化，TOF 开始计时，TMON 保持计时，TONR 计时值保持不变，HCNT、PLSR、SPD、DHSCS、DHSCI、DHSCR、DHSZ、DHST、DHSP、DHSTI 所对应的计数器保持当前数值，不再计数，当该步进状态再次有效时，在当前值的基础上继续计数（SPD 对应的数值重新开始计数）。
6. 连续的 STL 指令（STL 元件的串联）代表定义了一个并行汇合结构，STL 指令最大连续使用的次数为 16 次（并行分支汇合结构的最大分支数为 16）。
7. STL、RET 指令只能在主程序中使用。
8. 初始步进状态号只能用 S0 ～ S19，普通步进状态号只能用 S20 ～ S1023，且步进状态号不可重复使用。

使用示例




```
LD    SM1
SET   S0
STL   S0
OUT   Y0
LD    M0
SET   S20
STL   S20
OUT   Y0
LD    M1
OUT   S0
RET
```

如上图所示，在不同的步进状态之间，可以编写相同的输出软元件 Y0。
此时，S0 或 S20 有效时，Y0 输出为 ON。

注意事项

1． 如果对顺序图外和顺序图内的同一输出线圈编程或在同一状态内对相同的输出线圈编程时，则与普通的双重线圈一样处理，请务必注意。
2． 使用该指令时梯形图末尾请勿忘 RET 指令。

6.3 传输比较指令

6.3.1 CMP : 整数比较置位指令

D	CMP	P	S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：CMP (连续) 7步 CMPP (脉冲) 8步 32位：DCMP (连续) 9步 DCMPP (脉冲) 10步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	(D) INT										●	●	●	●	●	●	●	●	*	*	●	*	●	●	*
S2	(D) INT										●	●	●	●	●	●	●	●	*	*	●	*	●	●	*
D	BOOL		●	●	●																				

* 16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z、Un.b 元件。

指令说明

[CMP S1 S2 D]

S1：要比较的数据 1 或存放数据 1 的元件

S2：要比较的数据 2 或存放数据 2 的元件

D：输出结果的起始元件

指令执行时，比较 S1 和 S2 的数据，根据比较结果的不同（小于、等于、大于）使得 D、D+1、D+2 三个连号的位元件中相应的某一位元件置位为 ON，另外两个位元件复位为 OFF。

使用示例

M0 为 ON 时，执行 CMP 指令，相应的位元件置位。
M0 为 OFF 时，不执行 CMP 指令，M100 ～ M102 保持 M0 触点断开前的状态。

注意事项

1． 若需要得到≥、≤、≠的结果时，可将 D ～ D+2 并联取得。
2． 在指令不执行时若要清除其比较结果，请使用 RST 或 ZRST 指令。

6.3.2 RCMP : 浮点数比较置位指令

	RCMP	P	S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√	√
步长	16位： ----- ----- 32位： RCMP （连续） 9步 RCMPP （脉冲） 10步			相关元件				

操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z				Un.b
S1	REAL																●							●	●	
S2	REAL																●							●	●	
D	BOOL		●	●	●																					

指令说明

S1：要比较的数据 1 或存放数据 1 的元件

S2：要比较的数据 2 或存放数据 2 的元件

D：输出结果的起始元件

指令执行时，比较 S1 和 S2 的数据，根据比较结果的不同（小于、等于、大于）使得 D、D+1、D+2 三个号的位元件中相应的某一位元件置位为 ON，另外两个位元件复位为 OFF。

使用示例

当 M0 为 ON 时，执行 RCMP 指令，相应的位元件置位。

当 M0 为 OFF 时，不执行 RCMP 指令，M100 ～ M102 保持 M0 触点断开前的状态

注意事项

1．若需要得到≥、≤、≠的结果时，可将 D ～ D+2 并联取得。

2．若要清除其比较结果，请使用 RST 或 ZRST 指令。

6.3.3 MOV：字数据传输指令

D	MOV	P	S D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位： MOV (连续) 5步 MOVP (脉冲) 6步 32位： DMOV (连续) 7步 DMOVP (脉冲) 8步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	(D) INT										•	•	•	•	•	•	•	•	*	*	•	*	•	•	*
D	(D) INT											•	•	•	•		•	•	*	*	•	*	•		*

* 16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z、Un.b 元件。

指令说明

```
├──┤ ├──[ MOV S D ]
```

S：要传输的数据或者存放要传输数据的元件

D：存放数据的元件

当指令执行时，将 S 中的数据赋给 D，S 中的数据不变。

使用示例

示例 1．传送 16 位数据，请使用 MOV 指令。

```
   M0
├──┤ ├──[ MOV 100 D0 ]        LD   M0
                              MOV  100 D0
```

当 M0 为 ON 时，执行 MOV 指令，将 100 传送给 D0，D0 的数据变为 100。

当 M0 为 OFF 时，D0 中的数据保持不变。

示例 2．传送 32 位数据，请使用 DMOV 指令。

```
   M1
├──┤ ├──[ DMOV C200 D10 ]     LD   M1
                              DMOV C200 D10
```

当 M1 为 ON 时，执行 DMOV 指令，将 C200 的数据传送给（D11，D10）。

当 M1 为 OFF 时，（D11，D10）中的数据保持不变。

注意事项

MOV 指令支持有符号和无符号两种整数。

- 若指令的两个操作数都是软元件时，则数据类型都是有符号整数。
- 若指令的源操作数是有符号整数，如 - 50、+20，则目的操作数也是有符号整数
- 若指令的源操作数是无符号整数，如 200、55,535，则目的操作数也是无符号整数。

6.3.4 RMOV：浮点数数据传输指令

	RMOV	P	S D	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√	√
步长	16位： ----- ----- 32位： RMOV （连续） 7步 RMOVP （脉冲） 8步			相关元件				

操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z				Un.b
S	REAL																●				●		●	●	
D	REAL																●				●		●		

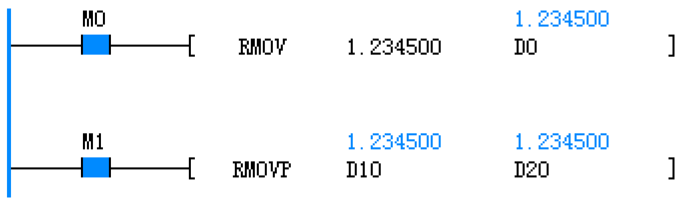
指令说明

S：要输的数据或者存放要传输数据的元件

D：存放数据的元件

当指令执行时，将 S 中的数据赋给 D，S 中的数据不变。

使用示例




```
LD     M0
RMOV   1.234500 D0
LD     M1
RMOVP  D10 D20
```

1. 当 M0 为 ON 时，执行 RMOV 指令，将 1.2345（自动转化为 1.234500）赋给（D1，D0），（D1，D0）=1.234500。

   当 M0 为 OFF 时，（D1，D0）中的数据保持不变。
2. 当 M1 由 OFF → ON 变化时，RMOVP 指令执行一次，将（D11，D10）的数据 1.2345（自动转化为 1.234500）赋给（D21，D20），（D21，D20）=1.234500。

6.3.5 BMOV : 块数据传输指令

	BMOV	P	S1 D S2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：BMOV (连续) 7步 32位： -----		BMOVP (脉冲) 8步 -----	相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT										•	•	•	•	•		•	•	*	•	•		•		•
D	INT											•	•	•	•		•		*	•	•		•		•
S2	INT										•	•	•	•	•	•	•	•	*	•	•	•	•	•	•

＊ 该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

```
├──┤ ├──[ BMOV  S1  D  S2 ]
```

S1：数据块传送的起始单元　**D**：数据块接收的起始单元　**S2**：数据块的大小

当 BMOV 指令执行时，将 S1 单元开始的 S2 个单元的内容赋给 D 单元开始的 S2 个单元，S1 单元开始的 S2 个单元的内容不变。

使用示例

```
         10        10
  M0
├─┤ ├──[ BMOV  D0   D100   5  ]          LD    M0
                                         BMOV  D0 D100 5
```

D0	→	D100
D1	→	D101
D2	→	D102
D3	→	D103
D4	→	D104

当 M0 为 ON 时，将 D0 开始的 5 个单元的内容赋给 D100 开始的 5 个单元。

- 执行前：D0=10，D1=20，D2=30，D3=40，D4=50；D100=D101=D102=D103=D104=0
- 执行后：D100=D0=10，D101=D1=20，D102=D2=30，D103=D3=40，D104=D4=50。D0 ～ D4 单元中内容不变。

注意事项

1．S1、D 同时为 Kn 寻址时，n 应相等。

```
  M0
├─┤ ├──[ BMOV  K1S0  K1Y0  2  ]
```

当 M0 为 ON 时，执行 BMOV 指令，将 K1S0 的数据赋给 K1Y0，K1S4 的数据赋给 K1Y4。

2．当传送的编号范围有重叠时， 数据的传送按如下方式。

- 当 S1<D 时，按①→②→③的顺序自动传送。

```
  M0
├─┤ ├──[ BMOV  D9  D10  3  ]
```

D9	③→	D10
D10	②→	D11
D11	①→	D12

- 当 S1>D 应避免编号只相差 1，其执行结果按①→②→③的顺序自动传送，若编号只差 1 则 D9 ～ D12 中的内容都是 D12 中的内容，请务必注意。

```
  M1
├─┤ ├──[ BMOV  D10  D9  3  ]
```

D10	③→	D9
D11	②→	D10
D12	①→	D11

3．S2>0，否则在 PLC 信息中提示“指令操作数数值非法”错误，指令不执行。

6.3.6 SWAP : 高低字节交换指令

	SWAP	P	D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：SWAP (连续) 3步 32位： -----		SWAPP (脉冲) 4步 -----	相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
D	WORD											●	●	●	●		●		*	●	●	●	●		

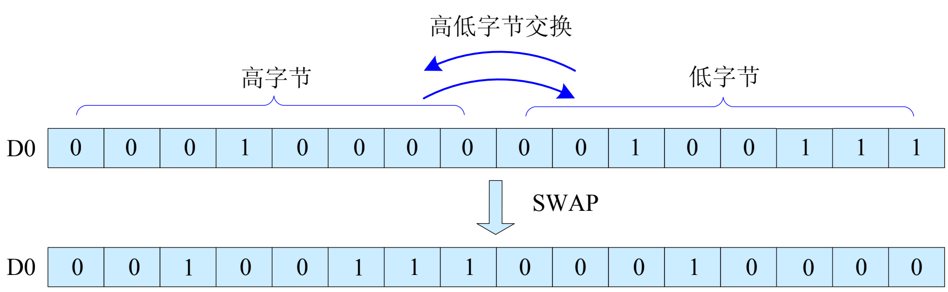
* 该指令使用 C 元件时，C 的使用范围为 C0 ～ C199。

指令说明

D：要交换高低字节的字元件

当指令执行时，将 D 单元中的内容进行高低字节交换后再保存到 D 单元中。

使用示例



当 M1 由 OFF → ON 变化时，SWAPP 指令执行一次，将 D0 单元中的内容进行高低字节交换后再保存到 D0 单元中。

- 执行前：D0=2#0001 0000 0010 0111=4,135
- 执行后：D0=2#0010 0111 0001 0000=10,000

注意事项

1. 使用连续执行型指令时，请注意每个运算周期都会执行互换。
2. 在使用 Kn 寻址方式时，n 必须等于 4。

6.3.7 XCH : 字交换指令

D	XCH	P	D1 D2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：XCH (连续) 5步 XCHP (脉冲) 6步 32位：DXCH (连续) 7步 DXCHP (脉冲) 8步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
D1	(D) INT											●	●	●	●		●		＊	＊	●	＊	●		
D2	(D) INT											●	●	●	●		●		＊	＊	●	＊	●		

＊ 16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z 元件。

指令说明

D1：要交换的数据 1 **D2**：要交换的数据 2

当指令执行时，将目的操作数 D1 中的内容与目的操作数 D2 中的内容互换后保存到 D1、D2 单元中。

使用示例

示例 1．16 位指令


```
LD    M1
XCHP  D0 D1
```

当 M1 由 OFF → ON 变化时，XCHP 指令执行一次，将 D0 与 D1 中的内容互换后保存到 D0、D1 单元中。

- 执行前：D0=100 D1=200
- 执行后：D0=200 D1=100

示例 2．32 位指令


```
LD     M0
DXCHP  D10 D20
```

当 M0 由 OFF → ON 变化时，DXCHP 指令执行一次，将（D11，D10）与（D21，D20）中的内容互换后保存到（D11，D10）、（D21，D20）中。

- 执行前：（D11，D10）=131,073 （D21，D20）=262,147
- 执行后：（D11，D10）=262,147 （D21，D20）=131,073

注意事项

1．使用连续执行型指令时，请注意每个运算周期都会执行互换。
2．D1、D2 必须同时使用字串元件或都不使用字串元件，且在使用 Kn 寻址方式时，D1 和 D2 中的 n 应相等。
3．当 D1 和 D2 为同一地址号时，PLC 信息不报错，字交换指令执行，但 D1 和 D2 中的内容保持不变。

6.3.8 FMOV：数据块填充指令

D	FMOV	P	S1　D　S2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：FMOV (连续) 7步　FMOVP (脉冲) 8步 32位：DFMOV (连续) 9步　DFMOVP (脉冲) 10步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT										●	●	●	●	●	●	●	●	*	*	●	*	●	●	*
D	INT											●	●	●	●		●		*	*	●		●		*
S2	INT										●	●	●	●	●	●	●	●	*	*	●	*	●	●	*

＊　16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z、Un.b 元件。

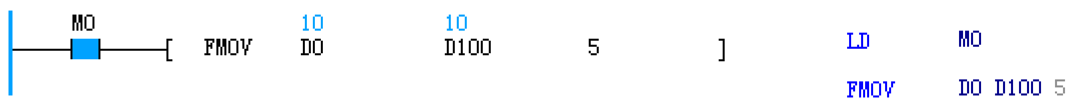
指令说明

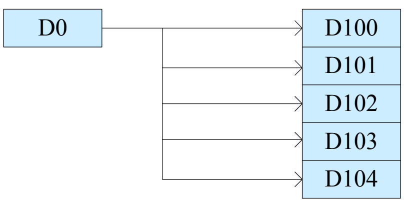
├──┤ ├──[FMOV S1 D S2]

S1：填充数据块中的数据　**D**：数据块的起始单元　**S2**：数据块的大小

当指令执行时，将 S1 单元的内容分别填充到 D 单元开始的 S2 个单元中，S1 单元的内容不变。

使用示例





当 M0 为 ON 时，执行 FMOV 指令，D0 的内容赋给 D100 开始的 5 个连续的单元中。

- 执行前：D0=10
  D100=D101=D102=D103=D104=0
- 执行后：D0=10
  D100=D101=D102=D103=D104=10

注意事项

1. 在使用 Kn 寻址方式时，S1 和 D 中的 n 应相等。
2. S2>0，否则在 PLC 信息中报“指令操作数数值非法”错误，指令不执行。

6.3.9 PUSH : 数据入栈指令

	PUSH	P	S1 D S2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：PUSH（连续） 7步 PUSHP（脉冲） 8步 32位： ----- -----			相关元件	进位标志：SM181			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●	●
D	INT																●						●		
S2	INT										●	●	●	●		●	●	●	*	●		●	●	●	

* 该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

├──┤ ├──[PUSH S1 D S2]

S1：要存入栈中的数据

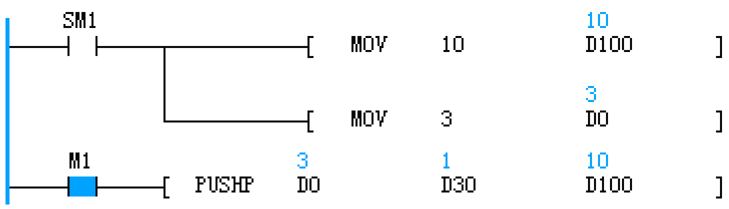
D：存储栈中元素个数，其元件标号为栈底位置

S2：栈的大小（栈尺寸），栈的大小不包括栈底元素（D 指定的元件）

1. 当指令执行时，将 S1 的值压入以 D 单元为栈底的栈顶中，同时 D 中的值增加 1，此时栈顶单元的编号为 D 的编号加 D 中数值。
2. 本指令一般使用脉冲执行型指令（PUSHP）。

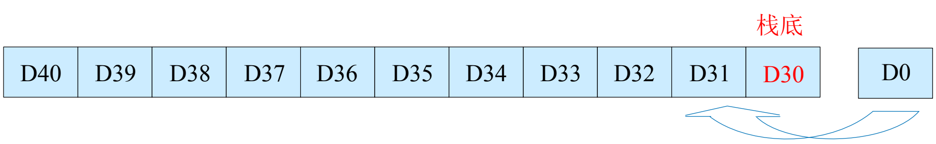
使用示例

示例 1.




```
LD    SM1
MOV   10 D100
MOV   3 D0
LD    M1
PUSHP D0 D30 D100
```

当 M1 由 OFF → ON 变化时，PUSHP 指令执行一次，将 D0 中的数值 3 压入 D31 中，D30 中的数据加 1，栈中其它软元件无变化。执行过程如下图所示：



- 执行前：D0=3，D30=D31=D32=D33=D34=D35=D36=D37=D38=D39=D40=0，D100=10。
- 执行后：D0=3，D30=1，D31=3，D32=D33=D34=D35=D36=D37=D38=D39=D40=0，D100=10。

示例 2.

请参考 6.3.10 FIFO 指令的程序举例（示例 2）。

注意事项

当操作数定义非法（S2 ≤ 0，D< 0，S2< D）时，PLC 信息中报“栈定义非法”错误，指令不执行。当 D=S2 时，仍有入栈指令执行，进位标志（SM181）置 ON，不执行压栈操作。

6.3.10 FIFO：先入先出指令

	FIFO	P	S1 D S2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：FIFO (连续) 7步 FIFOP (脉冲) 8步 32位： ----- -----			相关元件	零标志：SM180			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT																•						•		
D	INT											•	•	•	•		•	•	*	•	•	•	•		•
S2	INT										•	•	•	•		•	•	•	*	•		•	•	•	

* 该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

├──┤ ├──[FIFO S1 D S2]

D1：栈中元素个数，其元件编号加 1 为栈首元件

D2：出栈值存储单元

S：栈的大小（栈尺寸），栈的大小不包括栈底元素（D1 指定的元件）

1．当指令执行时，将栈首元件中的值赋给 D2 单元，同时 D1 的值减 1，D1+1 后的 S － 1 个单元的内容由后向前移动，最后的单元填 0。

2．本指令一般使用脉冲执行型指令（FIFOP）。

使用示例

示例 1.

```
   M0                 9        100
──┤↑├────[ FIFOP    D100      D10        10      ]        LD      M0
                                                            FIFOP   D100 D10 10
```

当 M0 由 OFF → ON 变化时，FIFO 指令执行一次，将 D101 单元的内容填入 D10 单元中，同时 D100 中的值减 1，D102 ～ D110 单元的内容由后向前移动，D110 单元填 0。

											栈底	
0	D110	D109	D108	D107	D106	D105	D104	D103	D102	D101	D100	D10

- 执行前：D10=0，D100=10，D101=100，D102=200，……，D109=900，D110=1000。
- 执行后：D10=100，D100=9，D101=200，D102=300，……，D109=1000，D110=0。

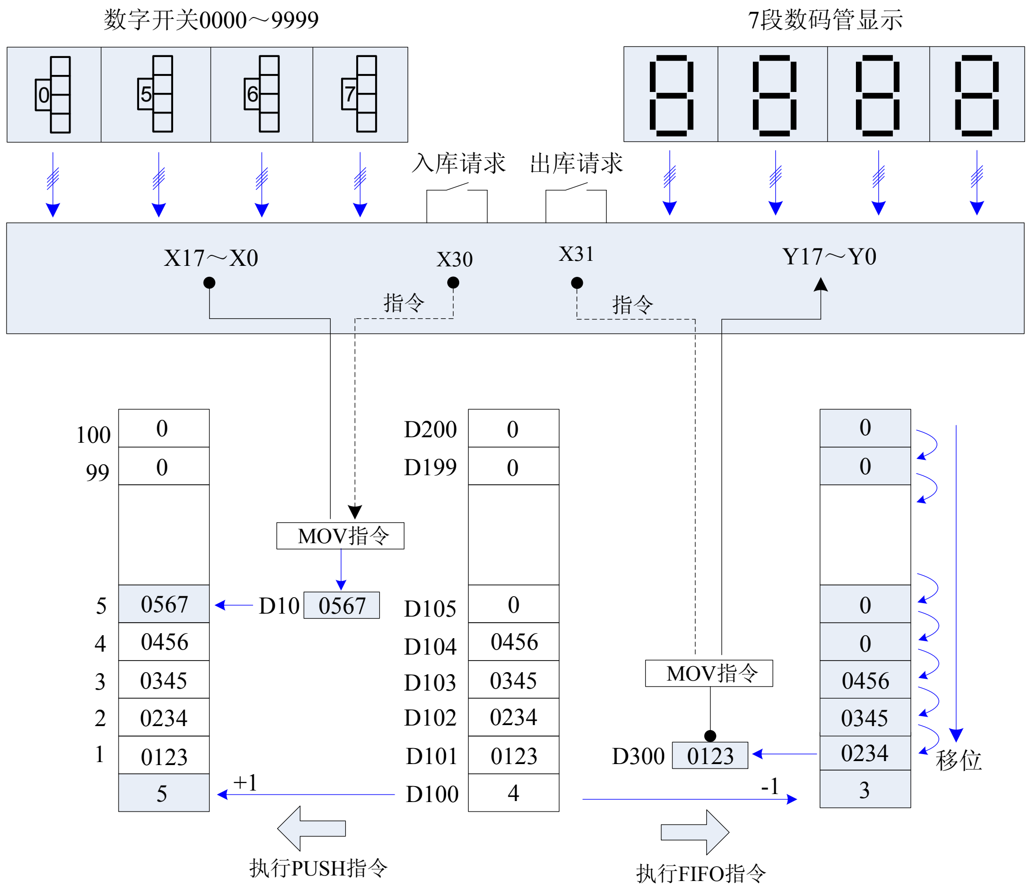
示例 2．先入先出控制的例子

下面例举了使用数据入栈指令（PUSH）、先入先出指令（FIFO）的示例加以说明。

（1）动作内容

- 登记产品编号的同时，为了能实现先入库的物品先出库这样的先入先出原则，以下介绍了输出当前应该取的产品编号的梯形图实例。
- 产品编号为 4 位数以下的 16 进制数，最大库存量在 100 点以下。

（2）程序



注意事项

当操作数定义非法（S1<0，S2 ≤ 0，S2<S1）时，PLC 信息中报“栈定义非法”错误，指令不执行。当 D1=0 时，表明栈空出，零标志（SM180）置 ON。

6.3.11 LIFO : 后入先出指令

	LIFO	P	S1 D S2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：LIFO (连续) 7步 LIFOP (脉冲) 8步 32位： ----- -----			相关元件	零标志：SM180			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT																•						•		
D	INT											•	•	•	•		•	•	*	•	•	•	•		•
S2	INT										•	•	•	•		•	•	•	*	•		•	•	•	

* 该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

```
├──┤ ├──[ LIFO S1 D S2 ]
```

D1：栈中元素个数，其元件编号加 1 为栈首元件

D2：出栈值存储单元

S：栈的大小（栈尺寸），栈的大小不包括栈底元素（D1 指定的元件）

1. 当指令执行时，将以 D1 为栈底的栈顶单元的内容赋给 D2 单元，同时 D1 的值减 1，D1 后的 S 个单元的内容保持不变。
2. 本指令一般使用脉冲执行型指令（LIFOP）。

使用示例

```
      M0            9       1000
├────┤↑├────[ LIFOP  D100    D10     10   ]
```

```
LD     M0
LIFOP  D100 D10 10
```

当 M0 由 OFF → ON 变化时，LIFOP 指令执行一次，将 D110 单元的内容填入 D10 单元中，同时 D100 中的值减 1，D101 ～ D110 单元的内容保持不变。

D10		D110	D109	D108	D107	D106	D105	D104	D103	D102	D101	D100 (栈底)

- 执行前：D10=0，D100=10，D101=100，D102=200，……，D109=900，D110=1000。
- 执行后：D10=1000，D100=9，D101=100，D102=200，……，D109=900，D110=1000。

注意事项

当操作数定义非法（S1<0，S2 ≤ 0，S2<S1）时，PLC 信息中报“栈定义非法”错误，指令不执行。当 D1=0 时，表明栈空出，零标志（SM180）置 ON。

6.4 四则运算指令

6.4.1 ADD : 整数加法指令

D	ADD	P	S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：ADD (连续) 7步 ADDP (脉冲) 8步 32位：DADD (连续) 10步 DADDP (脉冲) 11步			相关元件	零标志：SM180 进位标志：SM181 借位标志：SM182			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	(D) INT										●	●	●	●	●	●	●	●	*	*	●	*	●	●	*
S2	(D) INT										●	●	●	●	●	●	●	●	*	*	●	*	●	●	*
D	(D) INT											●	●	●	●		●		*	*	●	*	●		*

* 16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z、Un.b 元件。

指令说明

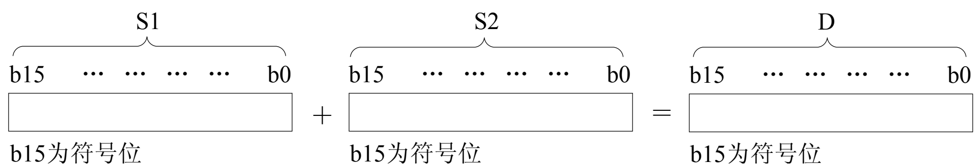
LIFO D1 D2 S] [S1] + [S2] = [D]

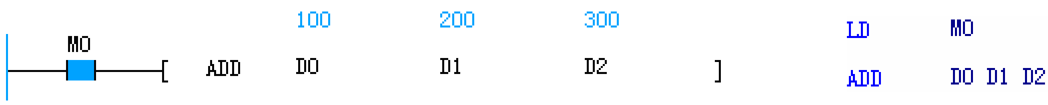
S1：被加数 **S2**：加数 **D**：和

1. 将 S1 单元中的内容与 S2 单元中的内容相加，结果存放在目的操作数 D 中。
2. 各数据的最高位为符号位：0 表示正，1 表示负，因此可做代数加法运算。例如：3+（－9）＝－6
3. 请注意 16 位及 32 位运算时，S1、S2 及 D 的符号位。

使用示例

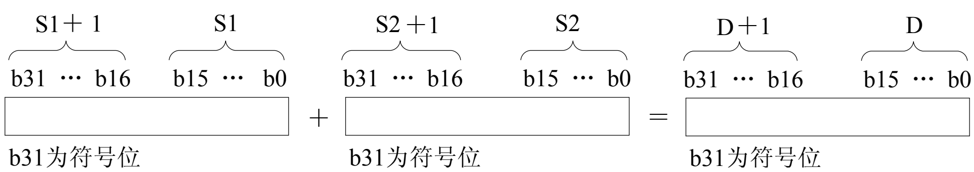
示例 1．16 位加法运算

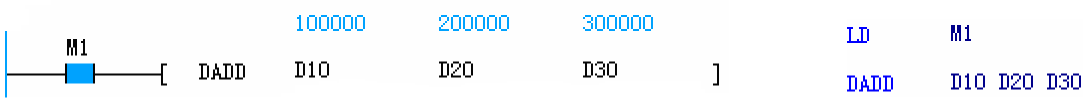




当 M0 为 ON 时，执行 ADD 指令，将 D0 和 D1 中的数据相加，结果存放在 D2 中。

示例 2．32 位加法运算





M1 为 ON 时，执行 DADD 指令，（D11，D10）和（D21，D20）中数据相加，结果放在（D31，D30）中。

注意事项

若运算结果为 0，则零标志（SM180）置 ON；若运算结果超过 32,767（16 位运算）或 2,147,483,647（32 位运算），则进位标志（SM181）置 ON；若运算结果小于－32,768（16 位运算）或－2,147,483,648（32 位运算），则借运算），则借位标志（SM182）置 ON。请参见 *3.7.2* 标志位的相关描述。

6.4.2 RADD : 浮点数加法指令

	RADD	P	S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位： ----- ----- 32位：RADD （连续） 10步 RADDP （脉冲） 11步			相关元件	零标志：SM180		进位标志：SM181	

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	REAL																●				●		●	●	
S2	REAL																●				●		●	●	
D	REAL																●				●		●		

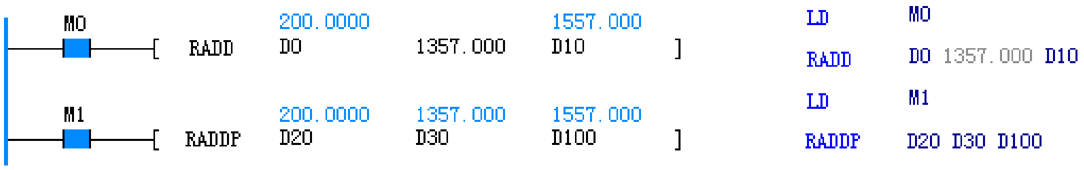
指令说明

|——| |——[RADD S1 S2 D] [S1] + [S2] = [D]

S1：被加数 **S2**：加数 **D**：和

1．将 S1 单元中的内容与 S2 单元中的内容相加，结果存放在目的操作数 D 中。

使用示例



1．当 M0 为 ON 时，执行 RADD 指令，将（D1，D0）中的数据与 1357（自动转换为 1357.000）相加，结果存放在（D11，D10）中。

2．当 M1 由 OFF → ON 变化时，RADDP 指令执行一次，将（D21，D20）和（D31，D30）中的数据相加，结果存放在（D101，D100）中。

注意事项

1．若运算结果为 0，则零标志（SM180）置 ON。

2．若运算结果大于 3.402823e+038 或小于 - 3.402823e+038，则进位标志（SM181）置 ON。

请参见 *3.7.2* 标志位的相关描述。

6.4.3 SUB：整数减法指令

D	SUB	P	S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：SUB (连续) 7步 SUBP (脉冲) 8步 32位：DSUB (连续) 10步 DSUBP (脉冲) 11步			相关元件	零标志：SM180 进位标志：SM181 借位标志：SM182			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	(D) INT										●	●	●	●	●	●	●	●	*	*	●	*	●	●	*
S2	(D) INT										●	●	●	●	●	●	●	●	*	*	●	*	●	●	*
D	(D) INT											●	●	●	●		●		*	*	●	*	●		*

* 16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z、Un.b 元件。

指令说明

├──┤ ├──[SUB S1 S2 D]　　[S1] － [S2] ＝ [D]

S1：被减数　**S2**：减数　**D**：差

1．将 S1 单元中的内容与 S2 单元中的内容相减，结果存放在目的操作数 D 中。
2．各数据的最高位为符号位：0 表示正，1 表示负，因此可做代数减法运算。例如：3 －（ - 9）＝ 12
3．请注意 16 位及 32 位运算时，S1、S2 及 D 的符号位。

使用示例

示例 1．16 位减法运算

S1		S2		D
b15 … … … … b0	－	b15 … … … … b0	＝	b15 … … … … b0
b15为符号位		b15为符号位		b15为符号位

```
M0                 300     100     200
├──┤ ├──[ SUB     D0      D1      D2    ]        LD    M0
                                                 SUB   D0 D1 D2
```

当 M0 为 ON 时，执行 SUB 指令，将 D0 中的数值减去 D1 中的数值，结果存放在 D2 中。

示例 2． 32 位减法运算

S1 +1	S1		S2 +1	S2		D +1	D
b31 … b16	b15 … b0	－	b31 … b16	b15 … b0	＝	b31 … b16	b15 … b0
b31为符号位			b31为符号位			b31为符号位	

```
M1                  3000000   1000000   2000000
├──┤ ├──[ DSUB     D10       D20       D30    ]       LD     M1
                                                      DSUB   D10 D20 D30
```

当 M1 为 ON 时，执行 DSUB 指令，将（D11，D10）中的数据减去（D21，D20）中的数据，结果存放在（D31，D30）中。

注意事项

若运算结果为 0，则零标志（SM180）置 ON。若运算结果超过 32,767（16 位运算）或 2,147,483,647（32 位运算），则进位标志（SM181）置 ON；若运算结果小于 - 32,768（16 位运算）或 - 2,147,483,648（32 位运算），则借位标志（SM182）置 ON。
请参见 *3.7.2* 标志位的相关描述。

6.4.4 RSUB：浮点数减法指令

	RSUB	P	S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位： ----- ----- 32位：RSUB (连续) 10步 RSUBP (脉冲) 11步			相关元件	零标志：SM180		进位标志：SM181	

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	REAL																●				●		●	●	
S2	REAL																●				●		●	●	
D	REAL																●				●		●		

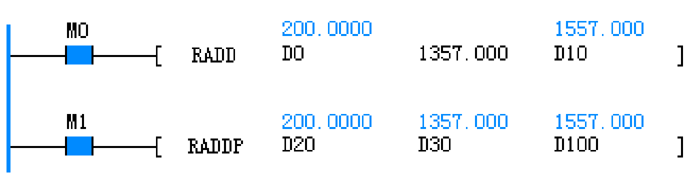
指令说明

├──┤ ├──[RSUB S1 S2 D]　　[S1] － [S2] ＝ [D]

S1：被减数　　**S2**：减数　　**D**：差

将 S1 单元中的内容与 S2 单元中的内容相减，结果存放在目的操作数中。

使用示例




```
LD     M0
RADD   D0 1357.000 D10
LD     M1
RADDP  D20 D30 D100
```

1. 当 M0 为 ON 时，执行 RSUB 指令，1357（自动转换为 1357.000）减去（D1，D0）中的数据，结果存放在（D11，D10）中。
2. 当 M1 由 OFF → ON 变化时，RSUBP 指令执行一次，将（D21，D20）中的数据减去（D31，D30）中的数据，结果存放在（D101，D100）中。

注意事项

1. 若运算结果为 0，则零标志（SM180）置 ON。
2. 若运算结果大于 3.402823e+038 或小于 - 3.402823e+038，则进位标志（SM181）置 ON。

请参见 *3.7.2* 标志位的相关描述。

6.4.5 INC : 增一指令

D	INC	P	D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：INC (连续) 3步 32位：DINC (连续) 4步		INCP (脉冲) 4步 DINCP (脉冲) 5步	相关元件	零标志：SM180		进位标志：SM181	

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
D	(D) INT											●	●	●	●		●		*	*	●	*	●		*

* 16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z、Un.b 元件。

指令说明

├──┤ ├──[INC D] [D] +1 → D

1. D 单元中的内容自加 1。
2. 本指令一般使用脉冲执行型指令（INCP、DINCP）。

使用示例

当 M0 由 OFF → ON 变化时，INCP 指令执行一次，D0 中的数据加 1。执行前：D0=0，执行后：D0=1。

注意事项

若运算结果为 0，则零标志（SM180）置 ON。若运算结果超过 32,767（16 位运算）或 2,147,483,647（32 位运算），则进位标志（SM181）置 ON；如请参见 *3.7.2* 标志位的相关描述。

6.4.6 DEC : 减一指令

D	DEC	P	D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：DEC (连续) 3步 32位：DDEC (连续) 4步		DECP (脉冲) 4步 DDECP (脉冲) 5步	相关元件	零标志：SM180		借位标志：SM182	

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
D	(D) INT											●	●	●	●		●		*	*	●	*	●		*

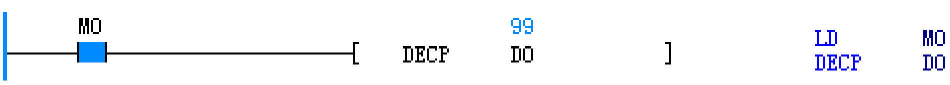
* 16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z、Un.b 元件。

指令说明

├──┤ ├──[DEC D] [D] − 1 → D

1. D 单元中的内容自减 1。
2. 本指令一般使用脉冲执行型指令（DECP、DDECP）。

使用示例



当 M0 由 OFF → ON 变化时，DECP 指令执行一次，D0 中的数据减 1。执行前：D0=100，执行后：D0=99。

注意事项

若运算结果为 0，则零标志（SM180）置 ON。若运算结果超过 32,767（16 位运算）或 2,147,483,647（32 位运算），则进位标志（SM182）置 ON；如请参见 *3.7.2* 标志位的相关描述。

6.4.7 MUL：整数乘法指令

D	MUL	P	S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：MUL (连续) 8步 MULP (脉冲) 9步 32位：DMUL (连续) 10步 DMULP (脉冲) 11步			相关元件	零标志：SM180			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	(D) INT										•	•	•	•	•	•	•	•	*	*	•	*	•	•	*
S2	(D) INT										•	•	•	•	•	•	•	•	*	*	•	*	•	•	*
D	DINT											•	•	•	•		•		*		•		•		

* 16 位指令的源操作数使用 C 元件时，只能使用 C0 ～ C199，目的操作数使用 C 元件时，只能使用 C200 ～ C255。
32 位指令的源操作数使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z、Un.b 元件。
32 位指令的目的操作数使用 C 元件时，只能使用 C200 ～ C255。

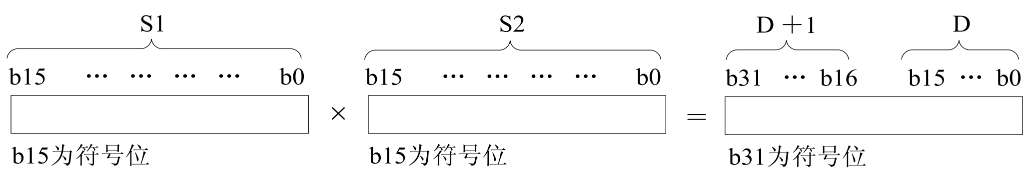
指令说明

S1：被乘数 **S2**：乘数 **D**：积

1. 将 S1 单元中的内容与 S2 单元中的内容相乘，结果存放在目的操作数 D 中。
2．16 位及 32 位运算时，请注意 S1、S2 及 D 的正负号位。

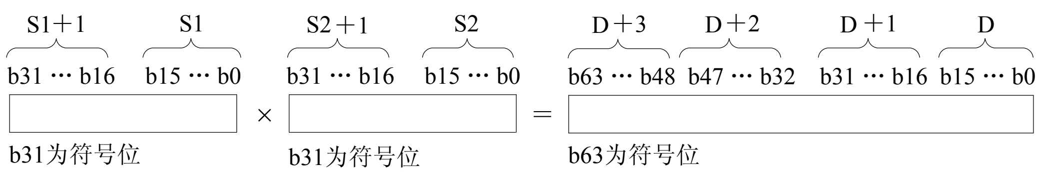
使用示例

示例 1．16 位乘法



当 M0 为 ON 时，执行 MUL 指令，将 D0 和 D1 中的数据相乘，结果存放在（D11，D10）中。

示例 2． 32 位乘法



当 M1 为 ON 时，执行 DMUL 指令，将（D101，D100）和（D201，D200）中的数据相乘，结果存放在（D303，D302）、（D301，D300）中。

注意事项

若运算结果为 0，则零标志（SM180）置 ON。

请参见 *3.7.2* 标志位的相关描述。

6.4.8 RMUL : 浮点数乘法指令

	RMUL	P	S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位： ----- ------ 32位：RMUL (连续) 10步 RMULP (脉冲) 11步			相关元件	零标志：SM180 进位标志：SM181			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	REAL																●				●		●	●	
S2	REAL																●				●		●	●	
D	REAL																●				●		●		

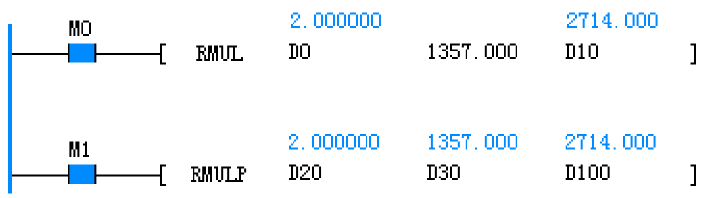
指令说明

├──┤ ├──[RMUL S1 S2 D] [S1]×[S2]= [D]

S1：被乘数 **S2**：乘数 **D**：积

将 S1 单元中的内容与 S2 单元中的内容相乘，结果存放在 D 单元中。

使用示例




```
LD      M0
RMUL    D0 1357.000 D10
LD      M1
RMULP   D20 D30 D100
```

1. 当 M0 为 ON 时，执行 RMUL 指令，将（D1，D0）中的数据和 1357（自动转换为 1357.000）相乘，结果存放在（D11，D10）中。
2. 当 M1 由 OFF → ON 变化时，RMULP 指令执行一次，将（D21，D20）和（D31，D30）中的数据相乘，结果存放在（D101，D100）中。

注意事项

1. 若运算结果为 0，则零标志（SM180）置 ON。
2. 若运算结果大于 3.402823e+038 或小于－3.402823e+038，则进位标志（SM181）置 ON。

请参见 *3.7.2* 标志位的相关描述。

6.4.9 DIV：整数除法指令

D	DIV	P	S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：DIV (连续) 7步 DIVP (脉冲) 8步 32位：DDIV (连续) 10步 DDIVP (脉冲) 11步			相关元件	零标志：SM180			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	(D) INT										●	●	●	●	●	●	●	●	*	*	●	*	●	●	*
S2	(D) INT										●	●	●	●	●	●	●	●	*	*	●	*	●	●	*
D	(D) INT											●	●	●	●		●		*	*	●	*	●		*

* 16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z、Un.b 元件。

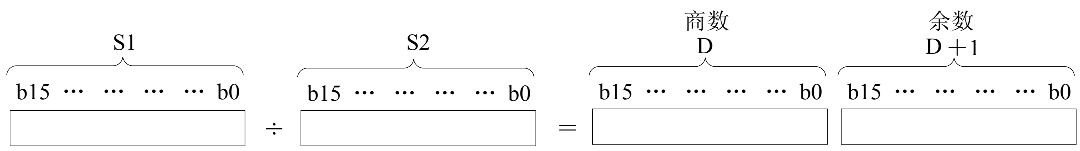
指令说明

├──┤ ├──[DIV S1 S2 D] [S1]÷[S2]＝[D]

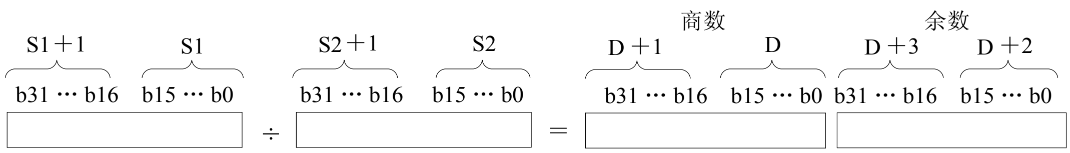
S1：被除数 **S2**：除数 **D**：商及余数

将 S1 单元中的数据除以 S2 单元中的数据，结果赋予 D 单元中。目的操作数包括两个单元，第一个单元存放商值，第二个单元存放余值。

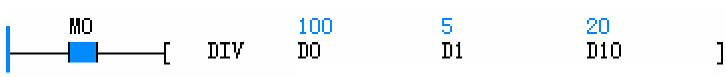
16 位除法运算



32 位除法运算



使用示例




```
LD   M0
DIV  D0 D1 D10
```

当 M0 为 ON 时，执行 DIV 指令，将 D0 中的数据除以 D1 中的数据，结果存放在 D10、D11 中，其中 D10 存放商值，D11 存放余值。

注意事项

1. S2（除数）为 0 时，PLC 信息中报“零除错误”，指令不执行。
2. 若运算结果为 0，则零标志（SM180）置 ON。请参见 *3.7.2* 标志位的相关描述。

6.4.10 RDIV : 浮点数除法指令

	RDIV	P	S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位： ----- ----- 32位：RDIV (连续) 10步 RDIVP (脉冲) 11步			相关元件	零标志：SM180			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	REAL																•				•		•	•	
S2	REAL																•				•		•	•	
D	REAL																•				•		•		

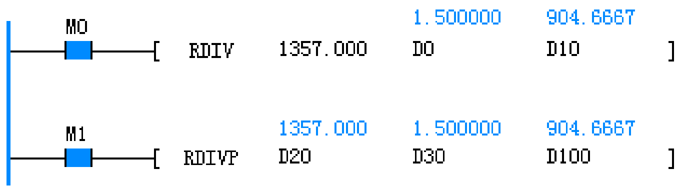
指令说明

RDIV S1 S2 D [S1]÷[S2] = [D]

S1：被除数 **S2**：除数 **D**：商

将 S1 单元中的数据除以 S2 单元中的数据，结果存放在 D 单元中。

使用示例




```
LD     M0
RDIV   1357.000 D0 D10
LD     M1
RDIVP  D20 D30 D100
```

1. 当 M0 为 ON 时，执行 RDIV 指令，用 1357（自动转换为 1357.000）除以（D1，D0）中的数据，结果存放在（D11，D10）中。
2. 当 M1 由 OFF → ON 变化时，RDIVP 指令执行一次，将（D21，D20）中的数据除以（D31，D30）中的数据，结果存放在（D101，D100）中。

注意事项

1. S2（除数）为 0 时，PLC 信息中报“零除错误”，指令不执行。
2. 若运算结果为 0，则零标志（SM180）置 ON。请参见 *3.7.2* 标志位的相关描述。

6.4.11 VABS : 整数绝对值指令

D	VABS	P	S D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：VABS (连续) 5步 32位：DVABS (连续) 7步		VABSP (脉冲) 6步 DVABSP (脉冲) 8步	相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	(D)INT										•	•	•	•	•	•	•	•	*	*	•	*	•	•	*
D	(D)INT											•	•	•	•		•		*	*	•	*	•		*

* 16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z、Un.b 元件。

指令说明

[VABS S D] [S] 取绝对值→ D

1．将 S 的值取绝对值后，结果赋予 D。
2．本指令一般使用脉冲执行型指令（VABSP、 DVABSP）。

使用示例

M0 —[VABSP D0 (-300) D1 (300)]

```
LD     M0
VABSP  D0  D1
```

当 M0 为 ON 时，VABSP 指令执行一次，将 D0 的值 - 300 取绝对值，结果赋予 D1，D1=300。

注意事项

16 位指令 S 的范围应为 - 32,767 ～ 32,767，否则 PLC 信息中报“指令操作数数值非法错误”，指令不执行。
32 位指令 S 的范围应为 - 2,147,483,647 ～ 2,147,483,647，否则 PLC 信息中报“指令操作数数值非法错误”，指令不执行。

6.4.12 RVABS : 浮点数绝对值指令

	RVABS	P	S D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位： ----- 32位：RVABS (连续) 7步		----- RVABSP (脉冲) 8步	相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	REAL																•				•		•	•	
D	REAL																•				•		•		

指令说明

[RVABS S D] [S] 取绝对值→ D

1．将 S 的值取绝对值后，结果赋予 D。
2．本指令一般使用脉冲执行型指令（RVABSP）。

使用示例

M0 —[RVABSP D10 (-300.0000) D20 (300.0000)]

```
LD      M0
RVABSP  D10 D20
```

当 M0 由 OFF → ON 变化时，RVABSP 指令执行一次，将（D11，D10）的值 - 300（自动转换为 - 300.0000）取绝对值，结果赋予（D21，D20），（D21，D20）=300.0000。

6.4.13 NEG : 整数取负指令

D	NEG	P	S D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：NEG (连续) 5步 32位：DNEG (连续) 7步		NEGP (脉冲) 6步 DNEGP (脉冲) 8步	相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	(D)INT										●	●	●	●	●	●	●	●	*	*	●	*	●	●	*
D	(D)INT											●	●	●	●		●		*	*	●	*	●		*

* 16 位指令使用 C 元件时，只能使用 C0 ～ C199。

32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z、Un.b 元件。

指令说明

[NEG S D] - [S] → D

1. 取 S 的相反数，结果赋予 D。
2. 本指令一般使用脉冲执行型指令（NEGP 、DNEGP）。

使用示例

```
   M0                  1000       -1000
|--|↑|--------[ NEGP   D0         D1        ]        LD     M0
                                                     NEGP   D0   D1
```

当 M0 由 OFF → ON 变化时，NEGP 指令执行一次，将 D0 的值 1000 取相反数，结果赋予 D1，D1= - 1000。

注意事项

16 位指令 S 的范围应为 - 32,767 ～ 32,767，否则 PLC 信息中报“指令操作数数值非法错误”，指令不执行。

32 位指令 S 的范围应为 - 2,147,483,647 ～ 2,147,483,647，否则 PLC 信息中报“指令操作数数值非法错误”，指令不执行。

6.4.14 RNEG : 浮点数取负指令

	RNEG	P	S D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位： ----- 32位：RNEG (连续) 7步		----- RNEGP (脉冲) 8步	相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	REAL																●				●		●	●	
D	REAL																●				●		●		

指令说明

[RNEG S D] - [S] → D

1. 取 S 的相反数，结果赋予 D。
2. 本指令一般使用脉冲执行型指令（RNEGP）。

使用示例

```
   M0                  400.0000   -400.000
|--|↑|--------[ RNEGP  D10        D20       ]        LD      M0
                                                     RNEGP   D10  D20
```

当 M0 由 OFF → ON 变化时，RNEGP 指令执行一次，将（D11，D10）的值 400（自动转换为 400.0000）取相反数，结果赋予（D21，D20），（D21，D20）= - 400.0000。

6.4.15 SQT : 整数平方根指令

D	SQT	P	S D	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√	√
步长	16位：SQT (连续) 5步 32位：DSQT (连续) 7步		SQTP (脉冲) 6步 DSQTP (脉冲) 8步	相关元件	零标志：SM180 借位标志：SM182			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	(D)INT										•	•	•	•	•	•	•	•	*	*	•	*	•	•	*
D	(D)INT											•	•	•	•		•		*	*	•	*	•		*

* 16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z、Un.b 元件。

指令说明

┤├ ┤├[SQT S D] [S] 取算术平方根→ D

对 S 的值取算术平方根，结果赋予 D。

使用示例

M0 ┤├[SQTP 28 D0]　(5)

LD M0
SQTP 28 D0

当 M0 由 OFF → ON 变化时，SQTP 指令执行一次，将 28 取算术平方根，结果赋予 D0，D0=5。因算结果舍去小数，故借位标志位（SM182）置 ON。

注意事项

1. S 的取值范围必须满足 S ≥ 0，否则 PLC 信息中报“指令操作数数值非法”错误，指令不执行。
2. 若运算结果为 0，零标志（SM180）置 ON；若运算结果舍去小数，借位标志位（SM182）置 ON。

请参见 *3.7.2* 标志位的相关描述。

6.4.16 RSQT : 浮点数平方根指令

	RSQT	P	S D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位： ----- 32位：RSQT (连续) 7步		----- RSQTP (脉冲) 8步	相关元件	零标志：SM180			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	REAL																•				•		•	•	
D	REAL																•				•		•		

指令说明

┤├ ┤├[RSQT S D] [S] 取算术平方根→ D

对 S 的值取算术平方根，结果赋予 D。

使用示例

M0 ┤├[RSQTP 300.0000 D0]　(17.32051)

LD M0
RSQTP 300.0000 D0

当 M0 由 OFF → ON 变化时，RSQTP 指令执行一次，将 300（自动转换为 300.0000）取算术平方根，结果赋予（D1，D0），（D1，D0）=17.32051。

注意事项

1. S 的取值范围必须满足 S ≥ 0，否则 PLC 信息中报“指令操作数数值非法”错误，指令不执行。
2. 若运算结果为 0，零标志（SM180）置 ON；若运算结果舍去小数，借位标志位（SM182）置 ON。

6.4.17 SUM：整数累加指令

D	SUM	P	S1　S2　D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：SUM（连续）8步　SUMP（脉冲）9步 32位：DSUM（连续）9步　DSUMP（脉冲）10步			相关元件	零标志：SM180			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	(D)INT										●	●	●	●	●	●	●	●	*	*	●	*	●		*
S2	(D)INT										●	●	●	●	●	●	●	●	*	*	●	*	●	●	
D	DINT											●	●	●	●		●		*		●		●		

* 16 位指令使用 C 元件时，只能使用 C0 ～ C199。操作数 D 使用 C 元件时，范围为 C200 ～ C255。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z、Un.b 元件。

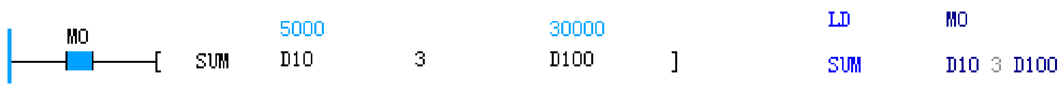
指令说明

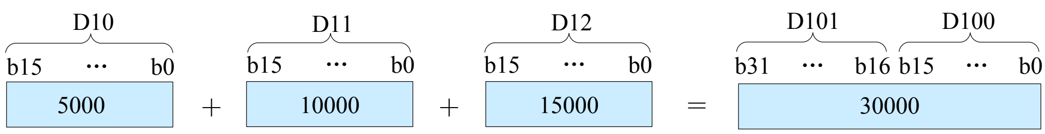
S1：累加起始单元　**S2**：累加数据个数（0 ～ 255）　**D**：累加结果

将累加起始单元 S1 开始的 S2 个单元的内容累加，运算后的结果赋给 D 单元。

使用示例

示例 1．16 位数据累加请使用 SUM 或 SUMP 指令。

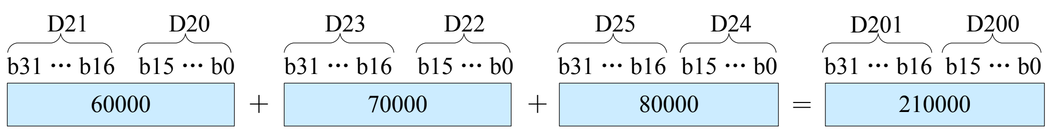




当 M0 为 ON 时，执行 SUM 指令，将 D10 起始的 3 个单元的内容累加，运算后的结果赋给（D101，D100）。

（D101，D100）=D10＋D11＋D12＝30,000。

示例 2．32 位数据累加请使用 DSUM 或 DSUMP 指令。



当 M1 由 OFF → ON 变化时，DSUMP 指令执行一次，将 D20 起始的 3×2 个单元的内容累加，运算后的结果赋给（D201，D200），（D201，D200）＝（D21，D20）＋（D23，D22）＋（D25，D24）＝ 210,000。

注意事项

1．SUM、DSUM 指令的运算结果都是 32 位数据。

2．S2 的操作数必须满足 0 ≤ S2 ≤ 255，否则 PLC 信息中报“指令操作数数值非法错误”，指令不执行。

3．若运算结果为 0，零标志（SM180）置 ON。请参见 *3.7.2* 标志位的相关描述。

6.4.18 RSUM : 浮点数累加指令

	RSUM	P	S1　　S2　　D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：----- 　----- 32位：RSUM（连续）9步　RSUMP（脉冲）10步			相关元件	零标志：SM180		进位标志：SM181	

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	REAL																•				•		•		
S2	INT										•	•	•	•	•	•	•	•			•		•		•
D	REAL																•				•		•		

指令说明

```
├──┤ ├──[ RSUM   S1   S2   D ]
```

S1：累加起始单元　**S2**：累加数据个数　**D**：累加结果

将累加起始单元 S1 开始的 S2×2 个单元的内容累加，运算后的结果赋给 D 单元。

使用示例

```
     M0                17.32051
├───┤ ├───[ RSQTP  300.0000  D0  ]

LD     M0
RSQTP  300.0000 D0
```

当 M0 为 ON 时，执行 RSUM 指令，将 D10 起始的 3×2 个单元的内容，按浮点数累加，运算后的结果赋给（D101，D100），

- 执行前：（D11，D10）=5000.100
  （D13，D12）=10000.20
  （D15，D14）=15000.30
  （D101，D100）=0
- 执行后：（D11，D10）=5000.100
  （D13，D12）=10000.20
  （D15，D14）=15000.30
  （D101，D100）=（D11，D10）+（D13，D12）+（D15，D14）=30000.60。

注意事项

1. S2 的操作数必须满足 0 ≤ S2 ≤ 255，否则 PLC 信息中报“指令操作数数值非法错误”，指令不执行。
2. 若运算结果大于 3.402823e+038 或小于 - 3.402823e+038，进位标志（SM181）置 ON；若运算结果为 0，零标志（SM180）置 ON。请参见 *3.7.2* 标志位的相关描述。

6.4.19 SIN : 浮点数 SIN 指令

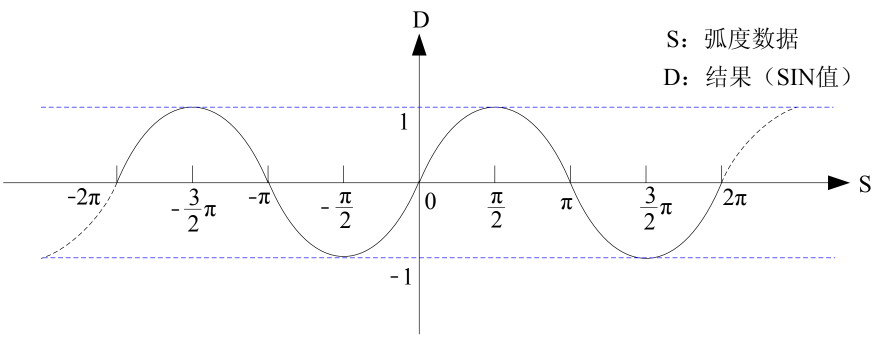
	SIN	P	S D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位： ----- ----- 32位：SIN (连续) 7步 SINP (脉冲) 8步			相关元件	零标志：SM180			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	REAL																•				•		•	•	
D	REAL																•				•		•		

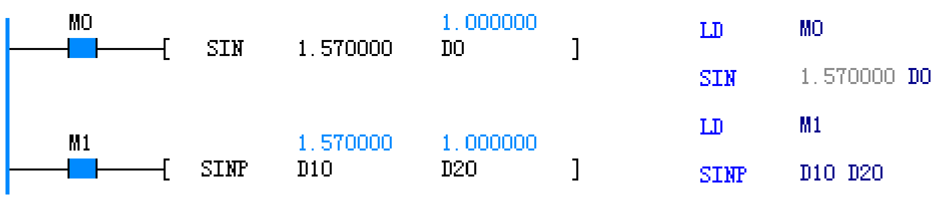
指令说明

├──┤ ├──[SIN S D] [S] 取正弦→ D

1. 求 S（单位为弧度）的 SIN（正弦）值，将运算结果赋予 D。
2. 弧度 ＝角度 ×（ π / 180）



使用示例



1. 当 M0 为 ON 时，执行 SIN 指令，求 1.57（自动转换为 1.570000）的正弦值，将运算结果赋予（D1，D0），（D1，D0）=1.000000。
2. 当 M1 由 OFF → ON 变化时，SINP 指令执行一次，求（D11，D10）=1.57（自动转换为 1.570000）的正弦值，将运算结果赋予（D21，D20），（D21，D20）=1.000000。

注意事项

若运算结果为 0，零标志（SM180）置 ON。请参见 *3.7.2* 标志位的相关描述。

6.4.20 ASIN : 浮点数 ASIN 指令

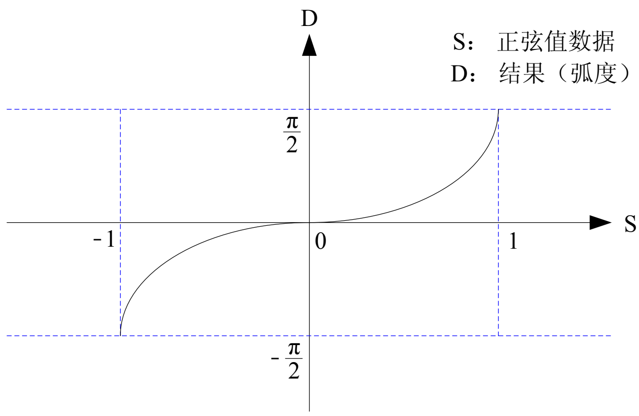
	ASIN	P	S D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位： ----- ----- 32位：ASIN (连续) 7步 ASINP (脉冲) 8步			相关元件	零标志：SM180			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	REAL																●				●		●	●	
D	REAL																●				●		●		

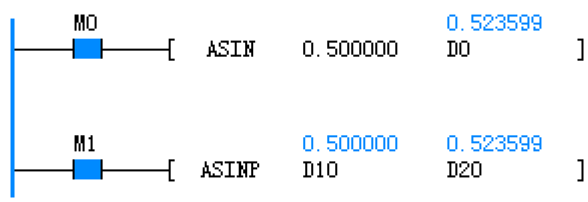
指令说明

ASIN S D [S] 反取正弦→ D

求 S 的 ASIN（反正弦）值，将运算结果赋予 D（单位为弧度）。



使用示例




```
LD     M0
ASIN   0.500000 D0
LD     M1
ASINP  D10 D20
```

1. 当 M0 为 ON 时，执行 ASIN 指令，求 0.5（自动转换为 0.500000）的反正弦值，将运算结果赋予（D1，D0），（D1，D0）=0.523599。
2. 当 M1 由 OFF → ON 变化时，ASINP 指令执行一次，求（D11，D10）=0.5（自动转换为 0.500000）的反正弦值，将运算结果赋予（D21，D20），（D21，D20）=0.523599。

注意事项

1. 若 S>1 或 S<-1，PLC 信息中报“指令操作数数值非法”错误，指令不执行。
2. 若运算结果为 0，零标志（SM180）置 ON。请参见 *3.7.2* 标志位的相关描述。

6.4.21 COS : 浮点数 COS 指令

	COS	P	S	D	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√	√	√
步长	16位： ----- 32位：COS (连续) 7步		----- COSP (脉冲) 8步		相关元件	零标志：SM180			

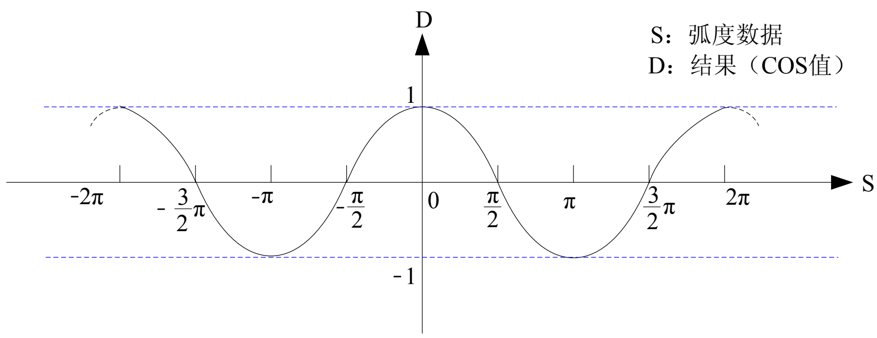
操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	REAL																●				●		●	●	
D	REAL																●				●		●		

指令说明

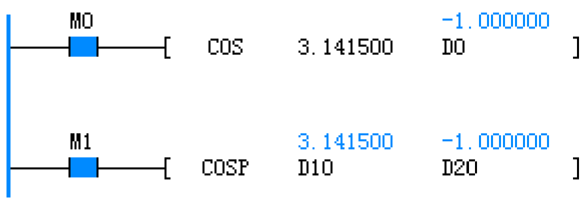
├──┤ ├──[COS S D] [S] 取余弦 → D

1．求 S（单位为弧度）的 COS（余弦）值，将运算结果赋予 D。

2．弧度 ＝角度 ×（ π / 180）



使用示例




```
LD    M0
COS   3.141500 D0
LD    M1
COSP  D10 D20
```

1．当 M0 为 ON 时，执行 COS 指令，求 3.1415（自动转换为 3.141500）的余弦值，将运算结果赋予（D1，D0），（D1，D0）＝－ 1.00000。

2． 当 M1 由 OFF → ON 变化时，COSP 指令执行一次， 求（D11，D10）=3.1415（自动转换为 3.141500）的余弦值，将运算结果赋予（D21，D20），（D21，D20）＝－ 1.00000。

注意事项

若运算结果为 0，零标志（SM180）置 ON。请参见 *3.7.2* 标志位的相关描述。

6.4.22 ACOS：浮点数 ACOS 指令

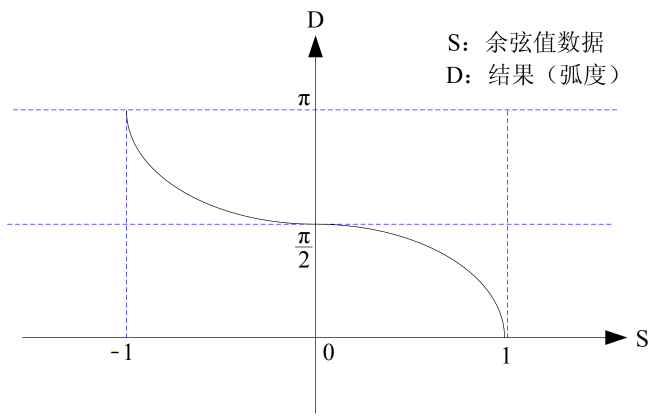
	ACOS	P	S	D	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√	√	√
步长	16位： ----- ----- 32位：ACOS (连续) 7步 ACOSP (脉冲) 8步				相关元件	零标志：SM180			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	REAL																●				●		●	●	
D	REAL																●				●		●		

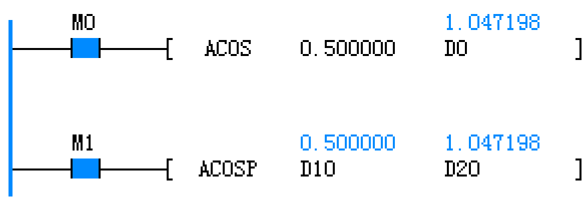
指令说明

├──┤ ├──[ACOS S D] [S] 反 取余弦→ D

求 S 的 ACOS（反余弦）值，将运算结果赋予 D（单位为弧度）。



使用示例




```
LD     M0
ACOS   0.500000 D0
LD     M1
ACOSP  D10 D20
```

1. 当 M0 为 ON 时，执行 ACOS 指令，求 0.5（自动转换为 0.500000）的反余弦值，将运算结果赋予（D1，D0），（D1，D0）=1.047198。
2. 当 M1 由 OFF → ON 变化时，ACOSP 指令执行一次，求（D11，D10）=0.5（自动转换为 0.500000）的反余弦值，将运算结果赋予（D21，D20），（D21，D20）=1.047198。

注意事项

1. 若 S>1 或 S<-1，PLC 信息中报“指令操作数数值非法”错误，指令不执行。
2. 若运算结果为 0，零标志（SM180）置 ON。请参见 *3.7.2* 标志位的相关描述。

6.4.23 TAN：浮点数 TAN 指令

<table>
<tr><td></td><td>TAN</td><td>P</td><td>S</td><td>D</td><td>适用机型</td><td>MX1H</td><td>MX2H</td><td>MX2V</td><td>MX3H</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td>√</td><td>√</td><td>√</td><td>√</td></tr>
<tr><td>步长</td><td colspan="4">16位：　-----　　　-----
32位：TAN　(连续)　7步　　TANP　(脉冲)　8步</td><td>相关元件</td><td colspan="4">零标志：SM180　　进位标志：SM181</td></tr>
</table>

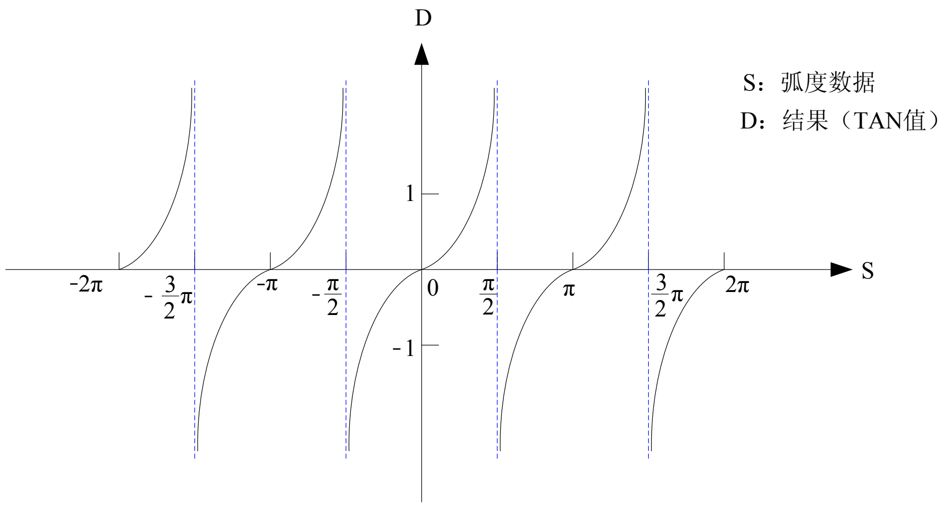
操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	REAL																●				●		●	●	
D	REAL																●				●		●		

指令说明

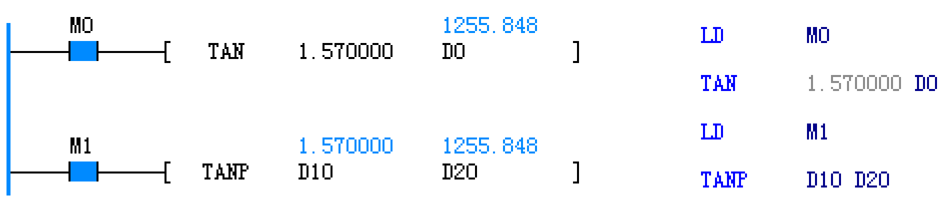
├──┤ ├──[TAN S D] [S] 取正切→ D

1．求 S（单位为弧度）的 TAN（正切）值，将运算结果赋予 D。

2．弧度 ＝角度 ×（ π / 180）



使用示例



1．当 M0 为 ON 时，执行 TAN 指令，1.57（自动转换为 1.570000）的正切值，将运算结果赋予（D1，D0），（D1，D0）=1255.848。

2．当 M1 由 OFF → ON 变化时，TANP 指令执行一次，求（D11，D10）=1.570000（自动转换为 1.570000）的正切值，将运算结果赋予（D21，D20），（D21，D20）=1255.848。

注意事项

1．若运算结果为 0，零标志（SM180）置 ON。

2．若运算结果大于 3.402823e+038 或小于 - 3.402823e+038，进位标志（SM181）置 ON。

请参见 *3.7.2* 标志位的相关描述。

6.4.24 ATAN：浮点数 ATAN 指令

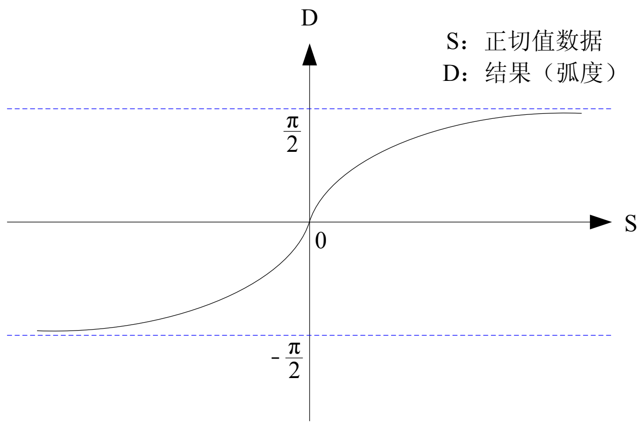
	ATAN	P	S　　　D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：----- ----- 32位：ATAN (连续) 7步 ATANP (脉冲) 8步			相关元件	零标志：SM180			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	REAL																•				•		•	•	
D	REAL																•				•		•		

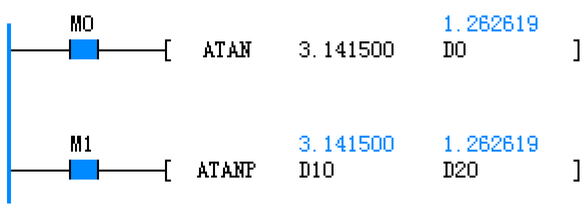
指令说明

├──┤ ├──[ATAN S D] [S] 取反正切→ D

求 S 的 ATAN（反正切）值，将运算结果赋予 D（单位为弧度）。



使用示例




```
LD     M0
ATAN   3.141500  D0
LD     M1
ATANP  D10  D20
```

1. 当 M0 为 ON 时，执行 ATAN 指令，求 3.1415（自动转换为 3.141500）的反正切值，将运算结果 D0）赋予（D1，D0），（D1，D0）=1.262619。
2. 当 M1 由 OFF → ON 变化时，ATANP 指令执行一次，求（D11，D10）=3.1415（自动转换为 3.141500）的反正切值，将运算结果赋予（D21，D20），（D21，D20）=1.262619。

注意事项

若运算结果为 0，零标志（SM180）置 ON。请参见 *3.7.2* 标志位的相关描述。

6.4.25 RAD : 浮点数角度→弧度转换指令

	RAD	P	S D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位： ----- -----			相关元件	零标志：SM180			
	32位：RAD（连续）7步 RADP（脉冲）8步							

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	REAL																●				●		●	●	
D	REAL																●				●		●		

指令说明

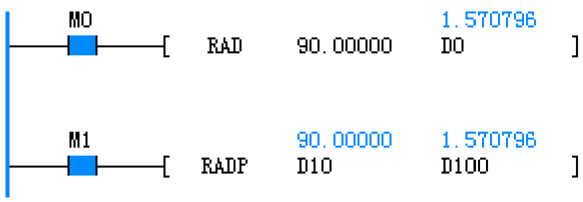
├──┤ ├──[RAD S D] [S] 转换成弧度值→ D

S：数据来源（角度） **D**：转换的结果（弧度）

1．将 S 单元的浮点数角度值转换为弧度值，结果赋予 D。

2．弧度 ＝角度 ×（π / 180）

使用示例




```
LD    M0
RAD   90.00000 D0
LD    M1
RADP  D10 D100
```

1．当 M0 为 ON 时，执行 RAD 指令，将 90（自动转换为 90.00000）（角度值）转换为弧度值，结果赋予（D1，D0），（D1，D0）=1.570796。

2．当 M1 由 OFF → ON 时，RADP 指令执行一次，（D11，D10）=90（自动转换为 90.00000）（角度值）转换为弧度值，结果赋予（D101，D100），（D101，D100）=1.570796。

注意事项

若运算结果为 0，零标志（SM180）置 ON。请参见 *3.7.2* 标志位的相关描述。

6.4.26 DEG：浮点数弧度→角度转换指令

	DEG	P	S D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：-----		-----	相关元件	零标志：SM180		进位标志：SM181	
	32位：DEG（连续）7步		DEGP（脉冲）8步					

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	REAL																●				●		●	●	
D	REAL																●				●		●		

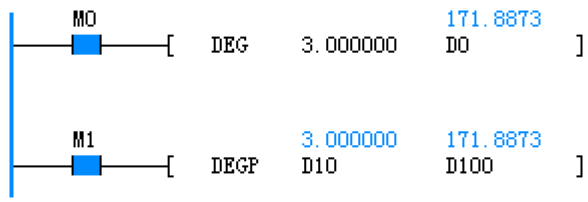
指令说明

├──┤ ├──[DEG S D] [S] 转换成角度值→ D

S: 数据来源（弧度） **D:** 转换的结果（角度）

1. 将 S 单元的浮点数弧度值转换为角度值，运算结果赋予 D 单元。
2. 弧度 ＝角度 ×（ π / 180）

使用示例




```
LD    M0
DEG   3.000000 D0
LD    M1
DEGP  D10 D100
```

1. 当 M0 为 ON 时，执行 DEG 指令，将 3（自动转换为 3.000000）（弧度值）转换为角度值，运算结果赋予（D1，D0），（D1，D0）=171.8873。
2. 当 M1 由 OFF → ON 变化时，DEGP 指令执行一次，将（D11，D10）=3.000000（自动转换为 3.000000）（弧度值）转换为角度值，运算结果赋予（D101，D100），（D101，D100）=171.8873。

注意事项

1. 若运算结果为 0，零标志（SM180）置 ON。
2. 若运算结果大于 3.402823e+038 或小于 - 3.402823e+038，进位标志（SM181）置 ON。

请参见 *3.7.2* 标志位的相关描述。

6.4.27 LOG：浮点数常用对数指令

	LOG	P	S	D	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√	√	√
步长	16位：-----		----- LOGP（脉冲）8步		相关元件	零标志：SM180		进位标志：SM181	
	32位：LOG（连续）7步								

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	REAL																•				•		•	•	
D	REAL																•				•		•		

指令说明

├──┤ ├──[LOG S D]　　$LOG_{10}{}^{[S+1,S]} = [D+1, D]$

S：数据来源　　**D**：运算的结果

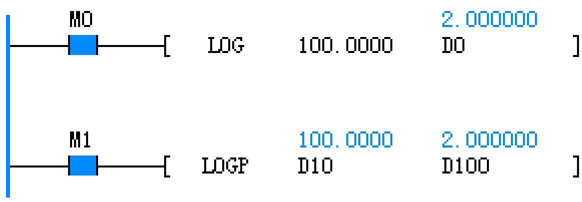
1．求 S 的 LOG 值，运算结果赋予 D。

2．LOG 是以 10 为底的常用对数运算　10^D =S，求 D 值　→　$LOG_{10}{}^{S}$ = D

例：已知 S=100，求 D= $LOG_{10}{}^{100}$ = ?

10^D =100 →　D= $LOG_{10}{}^{100}$ = 2

使用示例




```
LD      M0
LOG     100.0000 D0
LD      M1
LOGP    D10 D100
```

1．当 M0 为 ON 时，执行 LOG 指令，求 100（自动转换为 100.0000）的常用对数，运算结果赋予（D1，D0），（D1，D0）=2.000000。

2．当 M1 由 OFF → ON 变化时，LOGP 指令执行一次，求（D11，D10）=100.0000（自动转换为 100.0000）的常用对数，运算结果赋予（D101，D100），（D101，D100）=2.000000。

注意事项

1．若 S ≤ 0，PLC 信息中报“指令操作数数值非法”错误，指令不执行。

2．若运算结果为 0，零标志（SM180）置 ON；若运算结果大于 3.402823e+038 或小于－3.402823e+038，进位标志（SM181）置 ON。请参见 *3.7.2* 标志位的相关描述。

6.4.28 LN : 浮点数自然对数指令

	LN	P	S D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位： ----- ----- 32位：LN (连续）7步 LNP（脉冲）8步			相关元件	零标志：SM180 进位标志：SM181			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	REAL																•				•		•	•	
D	REAL																•				•		•		

指令说明

├──┤ ├──[LN S D] LN[S+1，S]= [D+1，D]

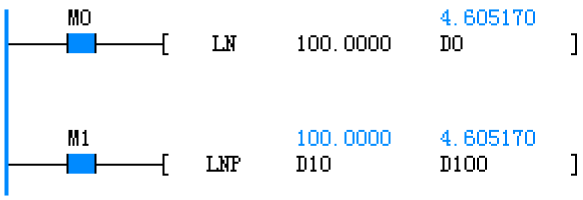
1．求 S 的自然对数，运算结果赋予 D。

2．LN 是以 e 为底的自然对数运算 $e^{D} = S \rightarrow D = LOG_e S = LNS$

例：已知 $S = e^{2}$ 求 $D = LOG_e S = LN S = ?$

$e^{D} = e^{2} \rightarrow D = LOG_e S = LN e^{2} = 2$

使用示例




```
LD   M0
LN   100.0000 D0
LD   M1
LNP  D10 D100
```

1．当 M0 为 ON 时，执行 LN 指令，求 100（自动变换为 100.0000）的自然对数，运算结果赋予（D1，D0），（D1，D0）=4.605170。

2．当 M1 由 OFF → ON 变化时，LNP 指令执行一次，求（D11，D10）=100.0000（自动变换为 100.0000）的自然对数，运算结果赋予（D101，D100），（D101， D100）=4.605170。

注意事项

1．若 S ≤ 0，PLC 信息中报“指令操作数数值非法”错误，指令不执行。

2．若运算结果为 0，零标志（SM180）置 ON；若运算结果大于 3.402823e+038 或小于 － 3.402823e+038，进位标志（SM181）置 ON。请参见 *3.7.2* 标志位的相关描述。

6.4.29 EXP : 浮点数自然数幂指令

	EXP	P	S　　　D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位： ----- 32位：EXP（连续）7步		----- EXPP（脉冲）8步	相关元件	零标志：SM180　进位标志：SM181			

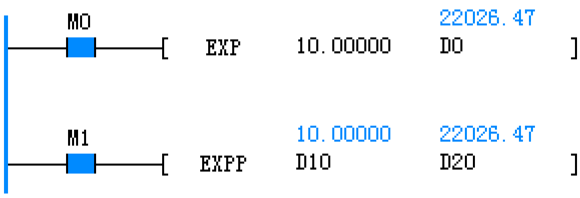
操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	REAL																●				●		●	●	
D	REAL																●				●		●		

指令说明

├──┤ ├──[EXP　S　D]　　$e^{[S+1, S]}$ = [D+1，D]

求 S 单元中内容的自然数幂，运算结果赋予 D 单元。

使用示例




```
LD    M0
EXP   10.00000 D0
LD    M1
EXPP  D10 D20
```

1. 当 M0 为 ON 时，执行 EXP 指令，求 10（自动转换为 10.00000）的自然数幂，运算结果赋予（D1，D0），（D1，D0）=22026.47。
2. 当 M1 由 OFF → ON 变化时，EXPP 指令执行一次，求（D11，D10）=10.00000（自动转换为 10.00000）的自然数幂，运算结果赋予（D21，D20），（D21，D20）=22026.47。

注意事项

1. 若运算结果为 0，零标志（SM180）置 ON。
2. 若运算结果大于 3.402823e+038 或小于－3.402823e+038，进位标志（SM181）置 ON。

请参见 *3.7.2* 标志位的相关描述。

6.4.30 POWER：浮点数求幂指令

	POWER	P	S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位： 32位：POWER(连续）10步		----- ----- POWERP（脉冲）11步	相关元件	零标志：SM180		进位标志：SM181	

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	REAL																•				•		•	•	
S2	REAL																•				•		•	•	
D	REAL																•				•		•		

指令说明

```
├──┤ ├──[ POWER S1 S2 D ]
```

[S1+1，S1]$^{[S2+1,\ S2]}$ = [D+1，D]

求 S1 的 S2 次幂，运算结果赋予 D。

使用示例

```
   M0                                         100.0000
──┤ ├──[ POWER   10.00000   2.000000   D100       ]

   M1            10.00000   2.000000   100.0000
──┤ ├──[ POWERP  D0         D10        D200       ]
```

```
LD      M0
POWER   10.00000 2.000000 D100
LD      M1
POWERP  D0 D10 D200
```

1. 当 M0 为 ON 时，执行 POWER 指令，求 10（自动转换为 10.00000）的 2（自动转换为 2.000000）次幂，运算结果赋予（D101，D100），（D101，D100）=100.00000。
2. 当 M1 由 OFF → ON 变化时，POWERP 指令执行一次，求（D1，D0）的（D11，D10）次幂（即 10.00000 的 2.000000 次幂），运算结果赋予（D201，D200），（D201，D200）=100.0000。

注意事项

1. 若 S1=0 且 S2 ≤ 0 或 S1< 0 且 S2 的尾数部分不为 0 时，PLC 信息中报“指令操作数数值非法错误”，指令不执行。
2. 若运算结果为 0，零标志（SM180）置 ON；若运算结果大于 3.402823e+038 或小于 - 3.402823e+038，进位标志（SM181）置 ON。请参见 *3.7.2* 标志位的相关描述。

6.4.31 WAND: 字与指令

D	WAND	P	S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：WAND (连续) 7步 WANDP (脉冲) 8步 32位：DWAND (连续) 10步 DWANDP (脉冲) 11步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	(D) WORD										●	●	●	●	●	●	●	●	*	*	●	*	●	●	*
S2	(D) WORD										●	●	●	●	●	●	●	●	*	*	●	*	●	●	*
D	(D) WORD											●	●	●	●		●		*	*	●	*	●		*

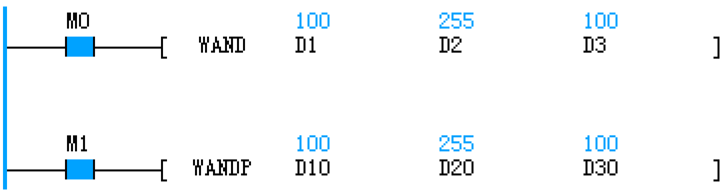
* 16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z、Un.b 元件。

指令说明

├──┤ ├──[WAND S1 S2 D] [S1] ∧ [S2] = [D]

将 S1 和 S2 中的数据按位进行字与运算，运算结果赋予 D。

使用示例




```
LD     M0
WAND   D1  D2  D3
LD     M1
WANDP  D10 D20 D30
```

1. 当 M0 为 ON 时，执行 WAND 指令，将 D1=2#1100100（100）与 D2=2#11111111（255）进行与运算，运算结果赋予 D3，D3=2#1100100（100）。

执行前：

	b15															b0
D1	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0
D2	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1

↓ WAND

执行后：

	b15															b0
D3	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0

2. 当 M1 由 OFF → ON 变化时，WANDP 指令执行一次，D10= 2#1100100（100）与 D20= 2#11111111（255）进行与运算，运算结果赋予 D30，D30=2#1100100（100）。

6.4.32 WOR：字或指令

D	WOR	P	S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：WOR (连续) 7步 WORP (脉冲) 8步 32位：DWOR (连续) 10步 DWORP (脉冲) 11步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	(D) WORD										●	●	●	●	●	●	●	●	*	*	●	●	●	●	*
S2	(D) WORD										●	●	●	●	●	●	●	●	*	*	●	●	●	●	*
D	(D) WORD											●	●	●	●		●		*	*	●	●	●		*

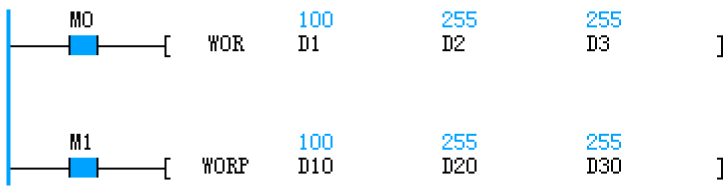
＊ 16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z、Un.b 元件。

指令说明

[WOR S1 S2 D]   [S1] ∨ [S2] = [D]

将 S1 和 S2 中的数据按位进行字字运算，运算结果赋予 D。

使用示例




```
LD    M0
WOR   D1 D2 D3
LD    M1
WORP  D10 D20 D30
```

1. 当 M0 为 ON 时，执行 WOR 指令，将 D1=2#1100100（100）与 D2=2#11111111（255）进行或运算，运算结果赋予 D3，D3=2#11111111（255）。

执行前

	b15															b0
D1	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0
D2	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1

WOR

执行后

	b15															b0
D3	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1

2. 当 M1 由 OFF → ON 变化时，WORP 指令执行一次，将 D10=2#1100100（100）与 D20=2#11111111（255）进行或运算，运算结果赋予 D30，D30=2#11111111（255）。

6.4.33 WXOR : 字异或指令

D	WXOR	P	S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：WXOR (连续) 7步 WXORP (脉冲) 8步 32位：DWXOR (连续) 10步 DWXORP (脉冲) 11步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	（D）WORD										•	•	•	•	•	•	•	•	*	*	•	*	•	•	*
S2	（D）WORD										•	•	•	•	•	•	•	•	*	*	•	*	•	•	*
D	（D）WORD											•	•	•	•		•		*	*	•	*	•		*

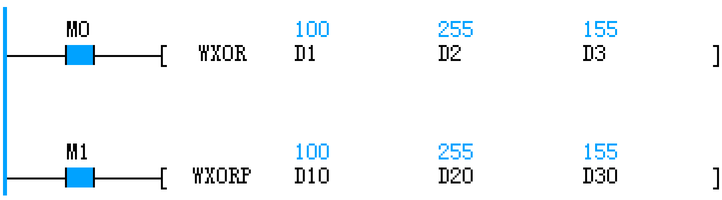
* 16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z、Un.b 元件。

指令说明

├──┤ ├──[WXOR S1 S2 D]　　[S1] ⊕ [S2] = [D]

将 S1 和 S2 中的数据按位进行字异或运算，运算结果赋予 D。

使用示例




```
LD     M0
WXOR   D1 D2 D3
LD     M1
WXORP  D10 D20 D30
```

1. 当 M0 为 ON 时，执行 WXOR 指令，D1=2#1100100（100）与 D2=2#11111111（255）进行异或运算，运算结果赋予 D3，D3=2#10011011（155）。

执行前：

	b15															b0
D1	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0
D2	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1

WXOR

执行后：

	b15															b0
D3	0	0	0	0	0	0	0	0	1	0	0	1	1	0	1	1

2. 当 M1 由 OFF → ON 变化时，WXORP 指令执行一次，D10=2#1100100（100）与 D20=2#11111111（255）进行异或运算，运算结果赋予 D30，D30=2#10011011（155）。

6.4.34 WINV : 字取非指令

D	WINV	P	S D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：WINV (连续) 5步 WINVP (脉冲) 6步 32位：DWINV (连续) 7步 DWINVP (脉冲) 8步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	(D) WORD										●	●	●	●	●	●	●	●	*	*	●	*	●	●	*
D	(D) WORD											●	●	●	●		●		*	*	●	*	●		*

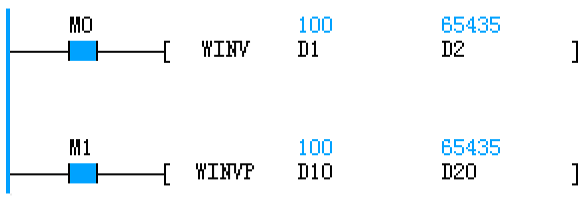
＊ 16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z 元件。

指令说明

├─┤ ├─[WINV S D] [S1] 取非 → D

将 S 中的数据按位逻辑取非，运算结果赋予 D。

使用示例




```
LD     M0
WINV   D1 D2
LD     M1
WINVP  D10 D20
```

1. 当 M0 为 ON 时，执行 WINV 指令，D1=2#1100100（100）取非后，运算结果赋予 D2，D2=2#1111111110011011（65,435）。

	b15															b0
执行前 D1	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0

⇩ WINV

	b15															b0
执行后 D2	1	1	1	1	1	1	1	1	1	0	0	1	1	0	1	1

2. 当 M1 由 OFF → ON 变化时，WINVP 指令执行一次，D10=2#1100100（100）取非后，运算结果赋予 D20，D20=2#1111111110011011（65,435）。

6.5 移位旋转指令

6.5.1 ROR : 位循环右移指令

D	ROR	P	S1 D S2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：ROR (连续) 7步 32位：DROR (连续) 9步		RORP (脉冲) 8步 DRORP (脉冲) 10步	相关元件	进位标志：SM181			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	(D) WORD										●	●	●	●	●	●	●	●	*	*	●	*	●	●	
D	(D) WORD											●	●	●	●		●		*	*	●	*	●		
S2	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●	

* 16 位指令的 S1、D 使用 C 元件时，只能使用 C0 ～ C199。
32 位指令的 S1、D 使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z 元件。
16 位指令和 32 位指令的 S2 使用 C 元件时，只能使用 C0 ～ C199。

指令说明

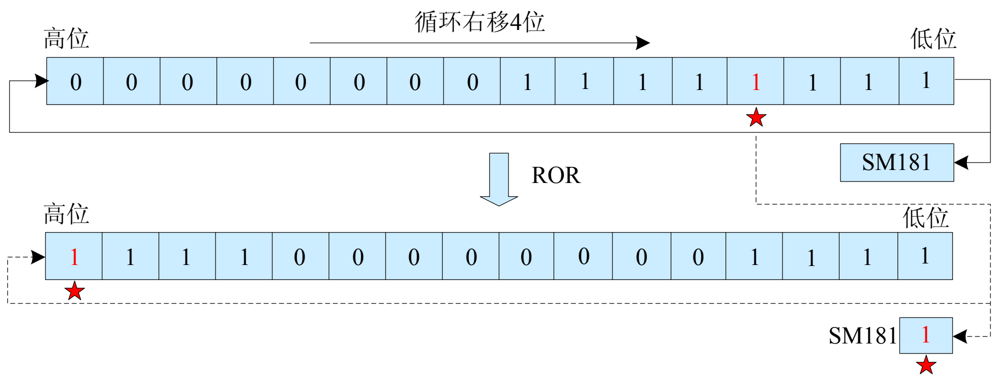
├──┤ ├──[ROR S1 D S2] [S1] 循环右移 [S2] 个位后→ D

S1：要循环右移的数据存放单元 **S2**：一次循环右移的位数 **D**：循环后的结果存储单元

1. 将源操作数 S1 中的内容循环右移 S2 位后存放在目的操作数 D 中，位移最终位存入进位标志（SM181）。
2. 本指令一般使用脉冲执行型指令（RORP，DRORP）。

使用示例

```
  M12                  255       61455
──┤ ├──[ RORP        D1        D10        4     ]     LDP    M12
                                                      RORP   D1  D10  4
```



当 M12 由 OFF → ON 变化时，RORP 指令执行一次，D1=2#11111111（255）循环右移 4 位后赋给 D10，D10=2#1111000000001111（61,455），循环右移最终位（图中标五角星处）赋给进位标志（SM181），SM181=1。

注意事项

1. S1 和 D 为 Kn 寻址时，n 必须等于 4（使用 16 位指令时）或者 8（使用 32 位指令时）。
2. S2 ≥ 0，否则 PLC 信息中报“指令操作数数值非法错误”，指令不执行。

6.5.2 ROL : 位循环左移指令

D	ROL	P	S1　D　S2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：ROL (连续) 7步　ROLP (脉冲) 8步 32位：DROL (连续) 9步　DROLP (脉冲) 10步			相关元件	进位标志：SM181			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	(D) WORD										•	•	•	•	•	•	•	•	*	*	•	*	•	•	
D	(D) WORD											•	•	•	•		•		*	*	•	*	•		
S2	INT										•	•	•	•	•	•	•	•	*	•	•	•	•	•	

* 16 位指令的 S1、D 使用 C 元件时，只能使用 C0 ～ C199。
32 位指令的 S1、D 使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z 元件。
16 位指令和 32 位指令的 S2 使用 C 元件时，只能使用 C0 ～ C199。

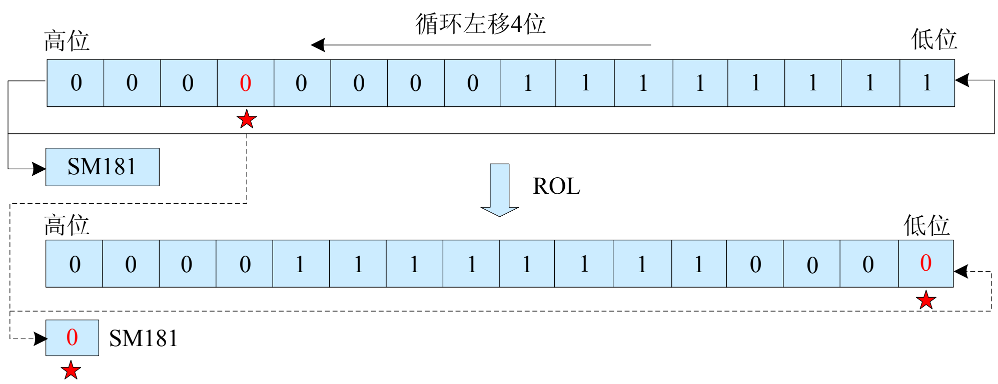
指令说明

├──┤ ├──[ROL S1 D S2] [S1] 循环左移 [S2] 个位后→ D

S1：要循环左移的数据存放单元　**S2**：一次循环左移的位数　**D**：循环后的结果存储单元

1. 将源操作数 S1 中的内容循环左移 S2 位后存放在目的操作数 D 中，位移最终位存入进位标志（SM181）。
2. 本指令一般使用脉冲执行型指令（ROLP，DROLP）。

使用示例



当 M0 由 OFF → ON 变化时，ROLP 指令执行一次，D0=2#11111111（255）循环左移 4 位后赋给 D10，D10=2#111111110000（4,080），循环左移的最终位（图中标五角星处）赋给进位标志（SM181），SM181=0。

注意事项

1. S1 和 D 为 Kn 寻址时，n 必须等于 4（使用 16 位指令时）或者 8（使用 32 位指令时）。
2. S2 ≥ 0，否则 PLC 信息中报“指令操作数数值非法错误”，指令不执行。

6.5.3 RCR : 位带进位循环右移指令

D	RCR	P	S1　D　S2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：RCR (连续) 7步　RCRP (脉冲) 8步 32位：DRCR (连续) 9步　DRCRP (脉冲) 10步			相关元件	进位标志：SM181			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	(D) WORD										•	•	•	•	•	•	•	•	*	*	•	*	•	•	
D	(D) WORD											•	•	•	•		•		*	*	•	*	•		
S2	INT										•	•	•	•	•	•	•	•	*	•	•	•	•	•	

* 16 位指令的 S1、D 使用 C 元件时，只能使用 C0 ～ C199。
32 位指令的 S1、D 使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z 元件。
16 位指令和 32 位指令的 S2 使用 C 元件时，只能使用 C0 ～ C199。

指令说明

[RCR S1 D S2] [S1] 带进位一起循环右移 [S2] 位→ D

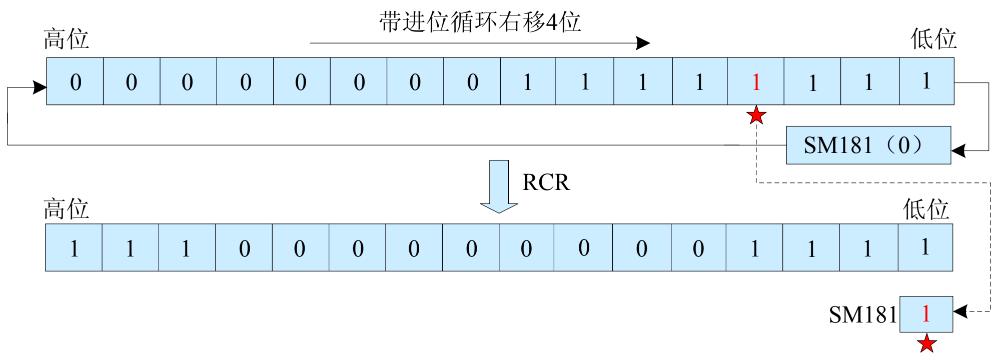
S1：要带进位循环右移的数据存放单元　**S2**：一次循环右移的位数　**D**：循环后的结果存放单元

1．将源操作数 S1 中的内容和进位标志（SM181）中的内容一起循环右移 S2 位后存放在目的操作数 D 中。

2．本指令一般使用脉冲执行型指令（RCRP，DRCRP）。

使用示例

```
 M0                 255        57359
─┤├──[ RCRP   D0         D10        4    ]        LDP   M0
                                                   RCRP  D0 D10 4
```



1．当 M0 由 OFF → ON 变化时，RCRP 指令执行一次，D0=2#11111111（255）和 SM181 中的值一起循环右移 4 位后赋给 D10，D10=2#1110000000001111（57,359）。图中标五角星处的位内容被传至进位标志（SM181），SM181=1。

2．RCR 指令会涉及到进位标志（SM181），因此在使用该指令时要注意进位标志位的状态。

注意事项

1．S1 和 D 为 Kn 寻址时，n 必须等于 4（使用 16 位指令时）或者 8（使用 32 位指令时）。

2．S2 ≥ 0，否则 PLC 信息中报“指令操作数数值非法错误”，指令不执行。

6.5.4 RCL：位带进位循环左移指令

D	RCL	P	S1 D S2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：RCL (连续) 7步 RCLP (脉冲) 8步 32位：DRCL (连续) 9步 DRCLP (脉冲) 10步			相关元件	进位标志：SM181			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	(D) WORD										•	•	•	•	•	•	•	•	*	*	•	*	•	•	
D	(D) WORD											•	•	•	•		•		*	*	•	*	•		
S2	INT										•	•	•	•	•	•	•	•	*	•	•	•	•	•	

* 16 位指令的 S1、D 使用 C 元件时，只能使用 C0 ～ C199。
32 位指令的 S1、D 使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z 元件。
16 位指令和 32 位指令的 S2 使用 C 元件时，只能使用 C0 ～ C199。

指令说明

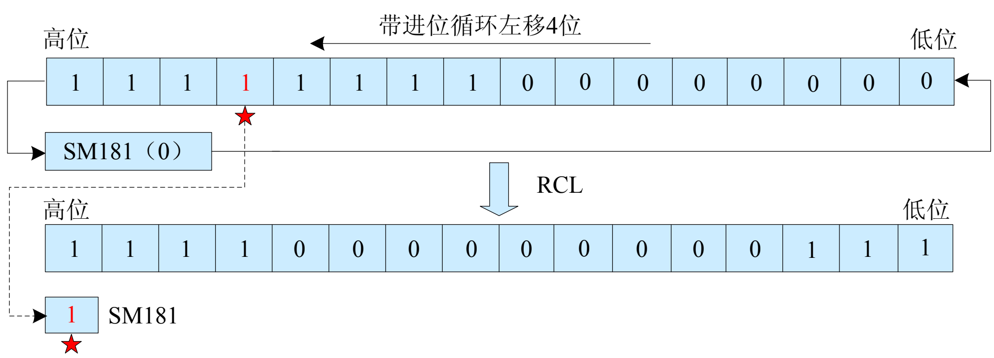
├──┤ ├──[RCL S1 D S2] [S1] 带进位一起循环左移 [S2] 位→ D

S1：要带进位循环左移的数据存放单元 **S2**：一次循环左移的位数 **D**：循环后的结果存放单元

1. 将源操作数 S1 中的内容和进位标志（SM181）中的内容一起循环左移 S2 位后存放在目的操作数 D 中。
2. 本指令一般使用脉冲执行型指令（RCLP，DRCLP）。

使用示例

```
   M0              65280      61447
├──┤ ├──[ RCLP     D0         D10        4     ]      LDP    M0
                                                      RCLP   D0 D10 4
```



1. 当 M0 由 OFF → ON 变化时，RCLP 指令执行一次，D0=2#1111111100000000（65,280）和 SM181 中的值一起循环左移 4 位后赋给 D10，D10= 2#1111000000000111（61,447）。上图中标五角星处的位内容被传至进位标志（SM181），SM181=1。
2. RCL 指令会涉及到进位标志（SM181），因此在使用该指令时要注意进位标志位的状态。

注意事项

1. S1 和 D 为 Kn 寻址时，n 必须等于 4（使用 16 位指令时）或者 8（使用 32 位指令时）。
2. S2 ≥ 0，否则 PLC 信息中报“指令操作数数值非法错误”，指令不执行。

6.5.5 SHR : 位右移指令

D	SHR	P	S1 D S2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：SHR (连续) 7步 SHRP (脉冲) 8步 32位：DSHR (连续) 9步 DSHRP (脉冲) 10步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	(D) WORD										●	●	●	●	●	●	●	●	*	*	●	*	●	●	
D	(D) WORD											●	●	●	●		●		*	*	●	*	●		
S2	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●	

* 16 位指令的 S1、D 使用 C 元件时，只能使用 C0 ～ C199。
32 位指令的 S1、D 使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z 元件。
16 位指令和 32 位指令的 S2 使用 C 元件时，只能使用 C0 ～ C199。

指令说明

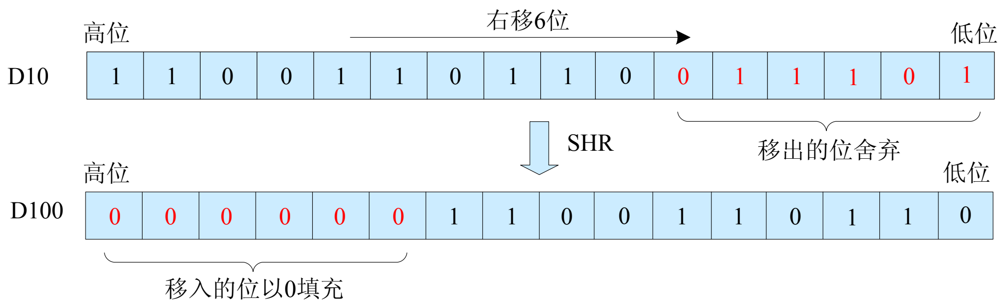
├──┤ ├──[SHR S1 D S2] [S1] 右移 [S2] 个位→ D

S1：要右移的数据存放单元 **S2**：一次右移的位数 **D**：移位后的结果存放单元

1．将源操作数 S1 右移 S2 个位后的结果存放在目的操作数 D 中，移出的位舍弃，移入的位以 0 填充。

2．本指令一般使用脉冲执行型指令（SHRP、DSHRP）。

使用示例



当 M0 由 OFF → ON 变化时，SHRP 指令执行一次，将 D10 中的数据 2#1100110110011101（52,637）右移 6 位，移出的位舍弃，移入的位以 0 填充，结果存放在 D100 中，D100=2#0000001100110110（822）。

注意事项

1．S1 和 D 为 Kn 寻址时，n 必须等于 4（使用 16 位指令时）或者 8（使用 32 位指令时）。

2．S2 ≥ 0，否则 PLC 信息中报“指令操作数数值非法错误”，指令不执行。

6.5.6 SHL : 位左移指令

D	SHL	P	S1 D S2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：SHL (连续) 7步 SHLP (脉冲) 8步 32位：DSHL (连续) 9步 DSHLP (脉冲) 10步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	(D) WORD										●	●	●	●	●	●	●	●	*	*	●	*	●	●	
D	(D) WORD											●	●	●	●		●		*	*	●	*	●		
S2	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●	

* 16 位指令的 S1、D 使用 C 元件时，只能使用 C0 ～ C199。
32 位指令的 S1、D 使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z 元件。
16 位指令和 32 位指令的 S2 使用 C 元件时，只能使用 C0 ～ C199。

指令说明

├──┤ ├──[SHL S1 D S2] [S1] 左移 [S2] 个位→ D

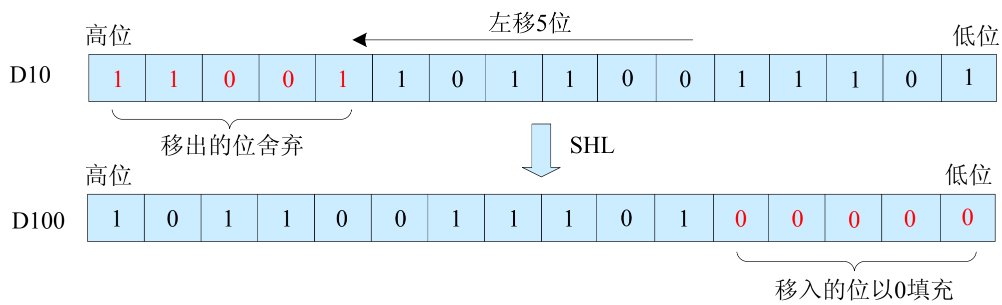
S1：要左移的数据存放单元 **S2**：一次左移的位数 **D**：移位后的结果存放单元

1．将源操作数 S1 左移 S2 个位后的结果存放在目的操作数 D 中，移出的位舍弃，移入的位以 0 填充。

2．本指令一般使用脉冲执行型指令（SHLP，DSHLP）。

使用示例

```
  M0          52637   45984
├──┤ ├──[ SHLP D10     D100    5  ]     LDP  M0
                                        SHLP D10 D100 5
```



当 M0 由 OFF → ON 变化时，SHLP 指令执行一次，将 D10 中的数据 2#1100110110011101（52,637）左移 5 位，移出的位舍弃，移入的位以 0 填充，结果存放在 D100 中，D100=2#1011001110100000（45,984）。

注意事项

1．S1 和 D 为 Kn 寻址时，n 必须等于 4（使用 16 位指令时）或者 8（使用 32 位指令时）。

2．S2 ⩾ 0，否则 PLC 信息中报“指令操作数数值非法错误”，指令不执行。

6.5.7 SFTR : 位串右移指令

	SFTR	P	S1 D S2 S3	适应机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：SFTR (连续) 9步 SFTRP (脉冲) 10步 32位： ----- -----			相关元件				

操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z				Un.b
S1	BOOL	●	●	●	●	●	●	●	●															●		
D	BOOL		●	●	●	●	●	●																●		
S2	INT										●	●	●	●	●	●	●	●	*	●	●	●		●	●	
S3	INT										●	●	●	●	●	●	●	●	*	●	●	●		●	●	

* 该指令使用 C 元件时，S2、S3 只能使用 C0 ～ C199。

指令说明

├──┤ ├──[SFTR S1 D S2 S3]

S1：要移入数据存放的起始单元　**S2**：移位的数据长度

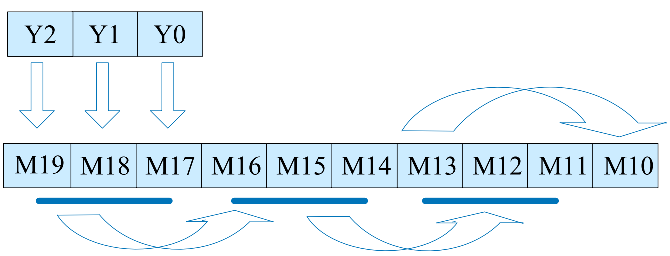
S3：一次右移的位数　**D**：要进行位串右移的数据存放单元

1．将 D 单元开始的 S2 个单元的内容右移 S3 个单元，最右端 S3 个数据将被丢弃，同时以 S1 单元开始的 S3 个单元的内容将被移入字串的左端。

2．本指令一般使用脉冲执行型指令（SFTRP）。

使用示例

```
 M0          OFF   OFF
─┤↑├──[ SFTRP  Y0    M10    10    3 ]        LD     M0
                                             SFTRP  Y0 M10 10 3
```



M12～M10 溢出丢弃

M13 ⇒ M10

M16～M14 ⇒ M13～M11

M19～M17 ⇒ M16～M14

Y2～Y0 ⇒ M19～M17

M0 由 OFF → ON 变化时，SFTRP 指令执行一次，将 M10 单元开始的 10 个单元的内容以位为单位右移 3 个单元，最右端 M12 ～ M10 将被丢弃。同时，Y0 单元开始的 3 个单元的内容被移入位串的左端。

- 执行前：Y0=1，Y1=0，Y2=1。M10=0，M11=1，M12=1，M13=0，M14=0，M15=1，M16=0，M17=0，M18=0，M19=1。
- 执行后：Y0 ～ Y2 内容不变。M10=0，M11=0，M12=1，M13=0，M14=0，M15=0，M16=1，M17=1，M18=0，M19=1。

注意事项

1．左右顺序，以大元件编号元件为左，小元件编号为右。

2. S3 的取值范围必须满足 S3 ≥ 0 且 S2 ≥ S3，否则 PLC 信息中报“指令操作数数值非法错误”，指令不执行。

6.5.8 SFTL : 位串左移指令

	SFTL	P	S1 D S2 S3	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：SFTL (连续) 9步 SFTLP (脉冲) 10步 32位： ----- -----			相关元件				

操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z				Un.b
S1	BOOL	●	●	●	●	●	●	●	●															●		
D	BOOL		●	●	●	●	●	●																●		
S2	INT										●	●	●	●	●	●	●	●	*	●	●	●		●	●	
S3	INT										●	●	●	●	●	●	●	●	*	●	●	●		●	●	

＊ 该指令使用 C 元件时，S2、S3 只能使用 C0 ～ C199。

指令说明

├──┤ ├──[SFTL S1 D S2 S3]

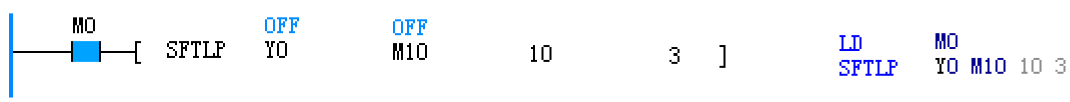
S1：要移入数据存放的起始单元 **S2**：移位的数据长度

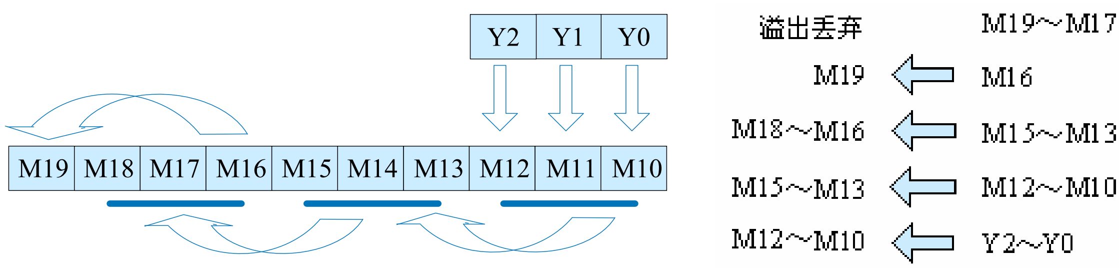
S3：一次左移的位数 **D**：要进行位串左移的数据存放单元

1．将 D 单元开始的 S2 个单元的内容左移 S3 个单元，最右端 S3 个数据将被丢弃，同时以 S1 单元开始的 S3 个单元的内容将被移入字串的右端。

2．本指令一般使用脉冲执行型指令（SFTLP）。

使用示例





M0 由 OFF → ON 变化时，SFTLP 指令执行一次，将 M10 单元开始的 10 个单元的内容以位为单位左移 3 个单元，最左端 M19 ～ M17 将被丢弃。同时，Y0 单元开始的 3 个单元的内容被移入位串的右端。

- 执行前：Y0=1，Y1=0，Y2=1。M10=0，M11=1，M12=1，M13=0，M14=0，M15=1，M16=0，M17=0，M18=0，M19=1。
- 执行后：Y0 ～ Y2 内容不变。M10=1，M11=0，M12=1，M13=0，M14=1，M15=1，M16=0，M17=0， M18=1，M19=0。

注意事项

1．左右顺序，以大元件编号元件为左，小元件编号为右。

2. S3 的取值范围必须满足 S3 ≥ 0 且 S2 ≥ S3，否则 PLC 信息中报“指令操作数数值非法错误”，指令不执行。

6.5.9 WSFR : 字串右移指令

	WSFR	P	S1 D S2 S3	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：WSFR（连续）9步　WSFRP（脉冲）10步 32位：-----　-----			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	WORD										●	●	●	●	●		●	●	*	●	●		●		
D	WORD											●	●	●	●		●		*	●	●		●		
S2	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●	
S3	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●	

＊　该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

├──┤ ├──[　WSFR　S1　D　S2　S3　]

S1：要移入数据存放的起始单元　　**S2**：字串右移的长度

S3：一次右移的字数　　**D**：要进行字串右移的数据存放单元

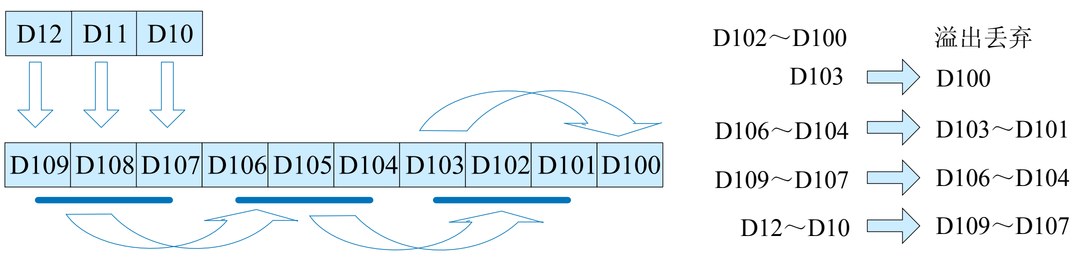
1．将 D 单元开始的 S2 个单元的内容以字为单位右移 S3 个单元，最右端 S3 个数据将被丢弃，同时以 S1 单元开始的 S3 个单元的内容将被移入字串的左端。

2．本指令一般使用脉冲执行型指令（WSFRP）。

使用示例

M0 ──[WSFRP　D10　D100　10　3]　（1　13）

```
LD     M0
WSFRP  D10 D100 10 3
```



当 M0 由 OFF → ON 变化时，WSFRP 指令执行一次，将 D100 开始的 10 个单元的内容以字为单位右移 3 个单元，最右端的 3 个数据被丢弃，同时以 D10 单元开始的连续 3 个单元的内容被移入字串的左端。

- 执行前：D10=1，D11=2，D12=3；D100=10，D101=11，D102=12，D103=13，D104=14，D105=15，D106=16，D107=17，D108=18，D109=19。
- 执行后：D10 ～ D12 不变，D100=13，D101=14，D102=15，D103=16，D104=17，D105=18，D106=19，D107=1，D108=2，D109=3。

注意事项

1．左右顺序，以大元件编号元件为右，小元件编号为左。

2．S2 的取值范围必须满足 S2 ≥ S3 且 S3 ≥ 0，否则 PLC 信息中报“指令操作数数值非法”，指令不执行。

3．当 S1、D 同时为 Kn 寻址时，n 应相等。

6.5.10 WSFL : 字串左移指令

	WSFL	P	S1　D　S2　S3	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：WSFL（连续）9步　　WSFLP（脉冲）10步 32位：-----　　-----			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	WORD										•	•	•	•	•		•	•	*	•	•		•		
D	WORD											•	•	•	•		•		*	•	•		•		
S2	INT										•	•	•	•	•	•	•	•	*	•	•	•	•	•	
S3	INT										•	•	•	•	•	•	•	•	*	•	•	•	•	•	

＊　该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

├──┤ ├──[　WSFL　S1　D　S2　S3　]

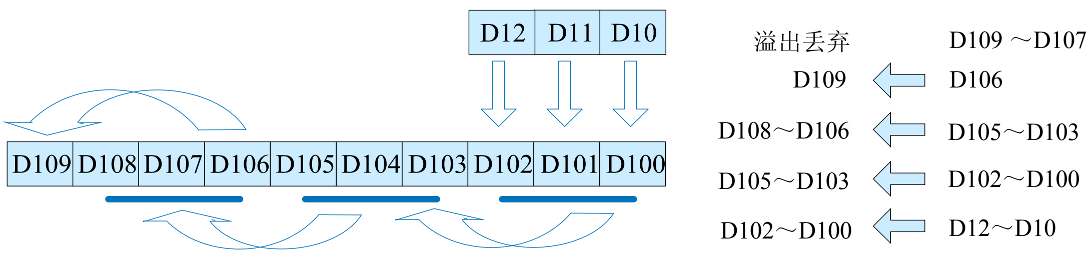
S1：要移入数据存放的起始单元　　**S2**：字串左移的长度

S3：一次左移的字数　　**D**：要进行字串左移的数据存放单元

1. 将 D 单元开始的 S2 个单元的内容以字为单位左移 S3 个单元，最左端 S3 个数据将被丢弃，同时以 S1 单元开始的 S3 个单元的内容将被移入字串的右端。
2. 本指令一般使用脉冲执行型指令（WSFLP）。

使用示例

```
     M0                         1      1
├───┤↑├───[ WSFLP   D10   D100   10   3 ]        LD      M0
                                                 WSFLP   D10 D100 10 3
```



当 M0 由 OFF → ON 变化时，WSFLP 指令执行一次，将 D100 开始的 10 个单元的内容以字为单位左移 3 个单元，最左端的 3 个数据被丢弃，同时以 D10 单元开始的连续 3 个单元的内容被移入字串的右端。

- 执行前：D10=1，D11=2，D12=3；D100=10，D101=11，D102=12，D103=13，D104=14，D105=15，D106=16，D107=17，D108=18，D109=19。
- 执行后：D10 ～ D12 不变，D100=1，D101=2,D102=3，D103=10，D104=11，D105=12，D106=13，D107=14，D108=15，D109=16。

注意事项

1. 左右顺序，以小元件编号元件为右，大元件编号为左。
2. S2 的取值范围必须满足 S2 ≥ S3 且 S3 ≥ 0，否则 PLC 信息中报“指令操作数数值非法”错误。
3. 当 S1、D 同时为 Kn 寻址时，n 应相等。

6.6 位处理指令

6.6.1 DECO : 解码指令

	DECO	P	S　　D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：DECO (连续) 5步　DECOP (脉冲) 6步 32位：----- -----			相关元件				

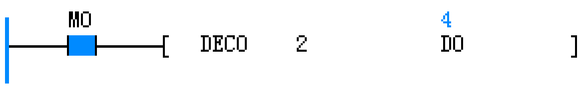
操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●	
D	WORD											●	●	●	●		●		*	●	●	●	●		

* 该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

将操作数 D 中的第 S 位置为 ON，其它位复位为 OFF。

使用示例




```
LD    M0
DECO  2 D0
```

		b15															b0
执行前	D0	0	1	1	0	1	1	0	1	0	1	1	0	0	0	0	0

⇩ DECO

		b15															b0
执行后	D0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0

当 M0 为 ON 时，执行 DECO 指令，将 D0=2#0110,1101,0110,0000（28000）的第 2 位（D0.2）置为 ON，其它位复位为 OFF。

注意事项

操作数 S 的有效范围为 0 ～ 15，否则在 PLC 信息中报“指令操作数数值非法”错误，指令不执行。

6.6.2 ENCO : 编码指令

	ENCO	P	S D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：ENCO (连续) 5步 ENCOP (脉冲) 6步 32位： ----- -----			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	WORD										•	•	•	•	•	•	•	•	*	•	•	•	•	•	
D	INT											•	•	•	•		•		*	•	•	•	•		

* 该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

```
├──┤ ├──[ ENCO S D ]
```

1. 将操作数 S 中位为 ON 的编号写入 D 中。
2. 当 S 中有多个位都为 ON 时，最小的为 ON 的位的编号被写入 D 中。

使用示例

```
      M0                          1
 ├────┤ ├────[ ENCO    2#10       D0        ]

      M1             28000        5
 ├────┤ ├────[ ENCO    D10        D100      ]
```

```
LD    M0
ENCO  2#10 D0
LD    M1
ENCO  D10 D100
```

1. 操作数 S 为 2#0010，其第 1 位为 ON，所以当 M0 为 ON 时，D0 的值为 1。

	b15															b0
2#10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
															1	0 第几位
D0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

↓ ENCO

	b15															b0
2#10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
	…	…	…	…	…	…	…	…	…	…	32	16	8	4	2	1
D0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

2. D10=2#0110 1101 0110 0000（28,000），其多个位都为 ON，且最小的为 ON 的位的编号为 5，所以当 M1 为 ON 时，D100 的值为 5。

	b15															b0
D10	0	1	1	0	1	1	0	1	0	1	1	0	0	0	0	0
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0 第几位
D100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

↓ ENCO

	b15															b0
D10	0	1	1	0	1	1	0	1	0	1	1	0	0	0	0	0
	…	…	…	…	…	…	…	…	…	…	32	16	8	4	2	1
D100	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

6.6.3 BITS : 字中 ON 位统计指令

D	BITS	P	S D	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√	√
步长	16位：BITS (连续) 5步 BITSP (脉冲) 6步 32位：DBITS (连续) 6步 DBITSP (脉冲) 7步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	(D) WORD										•	•	•	•	•	•	•	•	*	*	•	*	•	•	
D	INT											•	•	•	•		•		*	•	•	•	•		

*　16 位指令的 S 使用 C 元件时，只能使用 C0 ～ C199。
32 位指令的 S 使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z 元件。
16 位指令和 32 位指令的 D 使用 C 元件时，只能使用 C0 ～ C199。

指令说明

```
├──┤ ├──[ BITS S D ]
```

统计源操作数 S 中位为 ON 的个数，并将该个数赋给目的操作数 D。

使用示例

```
     M0                3855     8
├──┤↑├──[ BITS   D0       D2      ]      LDP   M0
                                         BITS  D0 D2
```

	b15															b0
D0	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
					8	7	6	5	←				4	3	2	1 个数
D2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

⇩ BITS

	b15															b0
D0	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
	…	…	…	…	…	…	…	…			32	16	8	4	2	1
D2	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0

当 M0 由 OFF → ON 变化时，BITS 指令执行一次，D0=2#0000 1111 0000 1111（3,855）中位为 ON 的个数为 8，所以 D2=8。

6.6.4 BON : ON 位判断指令

	BON	P	S1 D S2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：BON (连续) 7步 BONP (脉冲) 8步 32位： ----- -----			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	WORD										●	●	●	●	●	●	●	●	*	●	●	●	●	●	●
D	BOOL		●	●	●			●	●																
S2	INT																●				●			●	

* 该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

```
├──┤ ├──[ BON S1 D S2 ]
```

判断源操作数 S1 中第 S2 位的状态，并将该状态赋给目的操作数 D。

使用示例

```
   M0
├──┤ ├──[ BONP D0 Y0 8 ]
```

```
LD   M0
BONP D0 Y0 8
```

	b15															b0	
D0	0	0	1	0	0	1	0	0	0	1	0	1	0	1	0	1	⇨ Y0=OFF
								8	7	6	5	4	3	2	1	0	第几位

	b15															b0	
D0	0	0	1	0	0	1	0	1	0	1	0	1	0	1	0	1	⇨ Y0=ON

当 M0 由 OFF → ON 变化时，BONP 指令执行一次，检测 D0 中第 8 位的状态，并将检测到的状态赋给 Y0。

6.6.5 ZSET : 批量位置位指令

				适用机型	MX1H	MX2H	MX2V	MX3H
	ZSET	P	D S		√	√	√	√
步长	16位：ZSET (连续) 5步 ZSETP (脉冲) 6步 32位： ----- -----			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
D	BOOL		•	•	•			•															•		
S	BOOL/WORD		•	•	•			•			•	•	•	•	•	•	•	•	*	•	•	•	•	•	

＊ 该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

```
├──┤ ├──[ ZSET  D  S ]
```

D：待置位元件起始地址　　**S**：待置位元件个数或待置位元件结束地址

1．将 D 元件开始的 S 个连续位元件置 ON；或将 D 元件开始到 S 元件结束的连续位元件置 ON。

2．本指令一般使用脉冲执行型指令（ZSETP）。

使用示例

```
     M0
├────┤ ├──────────────[ ZSET   M10    20   ]

     M1
├────┤ ├──────────────[ ZSET   S10    S20  ]
```

1．当 M0 为 ON 时，执行 ZSET 指令，将辅助继电器 M10 ～ M29 连续 20 个位元件置 ON。

2．当 M1 为 ON 时，执行 ZSET 指令，将状态继电器 S10 ～ S20 连续 11 个位元件置 ON。

注意事项

置 ON 可单独使用 SET 指令，如位元件 Y、M、S、LM、SM。也可使用 FMOV 指令，将常数传送到位寄存器 KnY、KnM、KnS 来达到置 ON 功能。

```
     M0
├────┤ ├──┬────────────────────[ SET   M10      ]
          │
          ├────────────────────[ SET   Y0       ]
          │
          └──────[ FMOV   1      K1S0    5      ]
```

6.6.6 ZRST : 批量位清零指令

	ZRST	P	D1 D2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：ZRST (连续) 5步 ZRSTP (脉冲) 6步 32位： ----- -----			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
D1	BOOL /INT		●	●	●	●	●	●									●		●	●					
D2	BOOL /INT		●	●	●	●	●	●									●		●	●				●	

指令说明

D1：批复位的最前端的位 / 字软元件编号

D2：D2 为常数：批量位 / 字清零单元个数　　　　D2 为元件：批量复位的末尾的位 / 字软元件编号

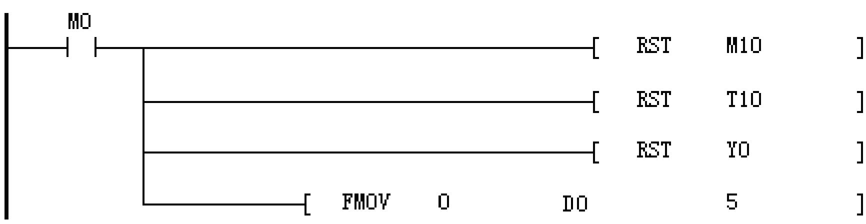
1. D2 为常数：将 D1 单元开始的 D2 个连续位元件复位为 OFF。
   D2 为元件：将同一种类的 D1 ～ D2 的软元件全部复位
2. 本指令一般使用脉冲执行型指令（ZRSTP）。

使用示例

1. 当 M0 为 ON 时，执行 ZRST 指令，将辅助继电器 M10 ～ M29 连续 20 个位元件复位为 OFF。
2. 当 M1 为 ON 时，执行 ZRST 指令，将 16 位计数器 C10 ～ C29 连续 20 个位元件复位为 OFF，对应的字元件中的计数值也将被清零。
3. 当 M2 为 ON 时，执行 ZSET 指令，将定时器 T10 ～ T29 连续 20 个字元件清零。
4. 当 M3 为 ON 时，执行 ZRST 指令，将状态继电器 S10 ～ S29 连续 20 个位元件复位为 OFF。
5. 当 M4 为 ON 时，执行 ZRST 指令，将输出点 Y10 ～ Y33 连续 20 个位元件复位为 OFF。

注意事项

复位可单独使用 RST 指令，如位元件 Y、M、S、T、C。也可使用 FMOV 指令，将常数传送到位寄存器 D 中来达到复位功能。



6.6.7 MEAN : 平均值指令

	MEAN	P	S1 D S2	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√	√
步长	16位：MEAN (连续) 7步 MEANP (脉冲) 8步 32位： ----- -----			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT										●	●	●	●	●	●	●	●	*	●			●		
D	INT											●	●	●	●		●	●	*				●		
S2	INT																●							●	

* 该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

├──┤ ├─[MEAN S1 D S2]

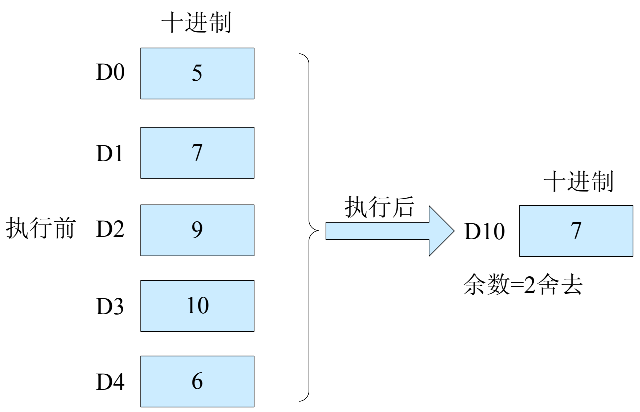
S1：取平均值的起始单元 **D**：存放平均值的单元 **S2**：取平均值的单元个数

将 S1 开始的 S2 个 16 位数据的平均值保存到 D 中，余数舍去。

使用示例

M0 ├─[MEAN D0 D10 5] (5, 7)　　LD M0 / MEAN D0 D10 5

当 M0=ON 时，执行 MEAN 指令，求 D0 开始的 5 个单元数据的平均值，保存在 D10 中，余数舍去。
（D0+D1+D2+D3+D4）/ 5 → D10



- 执行前：D0=5，D1=7，D2=9，D3=10，D4=6，D10=0。
- 执行后：D0=5，D1=7，D2=9，D3=10，D4=6，D10=7。余数 2 舍去。

注意事项

S2>0，否则 PLC 信息中报“指令操作数数值非法错误”，指令不执行。

6.6.8 WTOB：字节单位的数据分离指令

	WTOB	P	S1　　D　　S2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：WTOB　(连续)　7步　　WTOBP　(脉冲)　8步 32位：　　-----　　　　　　　-----			相关元件				

操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z				Un.b
S1	INT																•	•	*	•			•		
D	INT																•	•	*	•			•		
S2	INT																•							•	

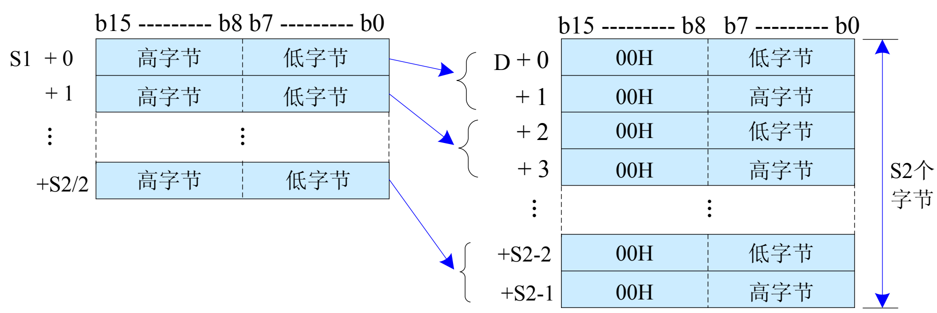
＊　该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

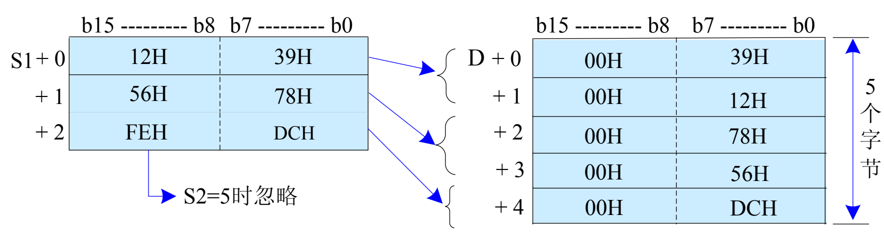
S1：按字节单位分离的数据软元件起始编号　**D**：已经按字节单位分离的结果软元件起始编号

S2：要分离的字节数据个数

1．将 S1 开始的 S2/2 个软元件中保存的 16 位数据分离成 S2 个字节，保存到以 D 开始的 S2 个软元件的低字节中，高字节清零。



2．当 S2 为奇数时，要分离的数据软元件个数为（S2+1）/ 2，在分离源（S1）的最后一个软元件中，只有低字节（8 位）为对象数据。例如，n=5，S+2 的高字节被忽略，其它数据被被保存在 D+0 ～ D+4 中。



使用示例

当 M0 为 ON 时，执行 WTOB 指令，将 D0 开始的 6/2=3 个单元的内容按照高低字节分成 6 个单元，保存在 D10 单元开始的 6 个单元中。

- 执行前：D0=16#201，D1=16#403，D2=16#605，D10=D11=D12=D13=D14=D15=0。
- 执行后：D0=16#201，D1=16#403，D2=16#605，D10=16#1，D11=16#2，D12=16#3，D13=16#4，D14=16#5，D15=16#6。

注意事项

1．S1 和 D 不能重叠。

2．S2 ＝ 0 时，PLC 不报错，指令不执行。

6.6.9 BTOW：字节单位的数据组合指令

	BTOW	P	S1 D S2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：BTOW (连续) 7步 BTOWP (脉冲) 8步 32位： ----- -----			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT																●	●	*	●			●		
D	INT																●	●	*	●			●		
S2	INT																●							●	

* 该指令使用 C 元件时，只能使用 **C0 ～ C199**。

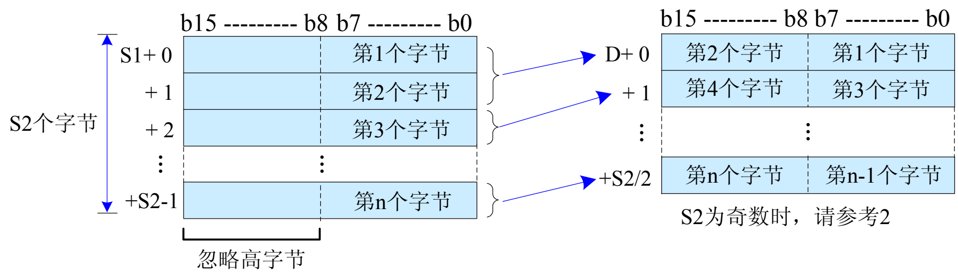
指令说明

├──┤ ├──[BTOW S1 D S2]

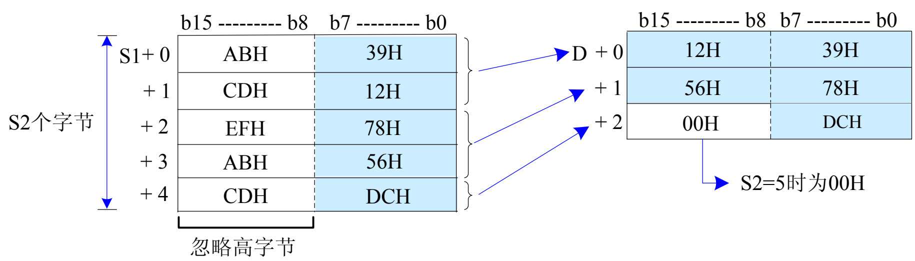
S1：保存要按字节单位结合的数据软元件起始编号

D：保存已经按字节单位结合的结果的软元件起始编号 **S2**：要结合的字节数据个数

1．将 **S1** 开始的 **S2** 个 **16** 位数据的低字节（**8** 位）结合在一起后的 **16** 位数据，保存到以 **D** 开始的 **S2/2** 个软元件中。



2．当 S2 为奇数时，组合后的数据软元件个数为（S2+1）/ 2。例如，S2=5，则 D 元件的个数为（5+1）/2= 3，D+2 的高字节用 0 填充。



使用示例

```
   M0              1          513
──┤ ├──[ BTOW     D0         D10        6     ]

LD     M0
BTOW   D0  D10  6
```

当 **M0** 为 **ON** 时，执行 **BTOW** 指令，将 **D0** 开始的 **6** 个单元的低字节结合在一起的 **16** 位数据保存在 **D10** 单元开始的 **6/2=3** 个单元中。

执行前：**D0=16#1，D1=16#2，D2=16#3，D3=16#4，D4=16#5，D5=16#6**。

执行后：**D10=16#201，D11=16#403，D12=16#605**。

注意事项

1．**S1** 和 **D** 不能重叠。

2．**S2** ＝ **0** 时，**PLC** 不报错，指令不执行。

6.6.10 UNI : 16 位数据的四位结合指令

	UNI	P	S1 D S2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：UNI (连续) 7步 UNIP (脉冲) 8步 32位： ----- -----			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT																●	●	*	●			●		
D	INT																●	●	*	●			●		
S2	INT																●							●	

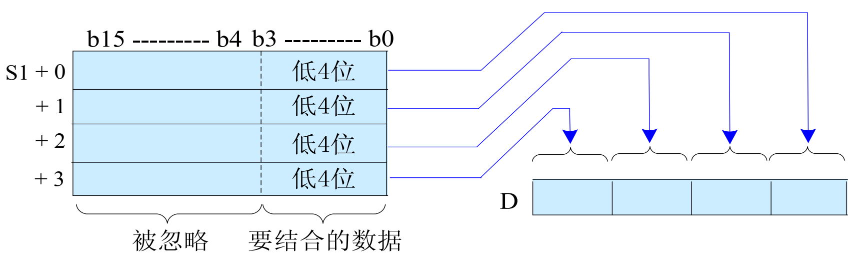
* 该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

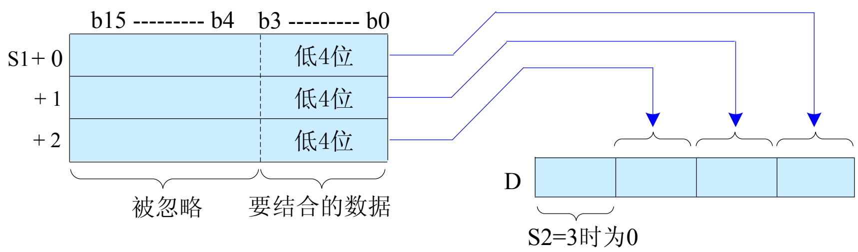
├──┤ ├──[UNI S1 D S2]

S1：要结合的数据的软元件起始编号 **D**：已结合的数据的软元件编号 **S2**：结合数

1．将 S1 开始的连续 S2 个单元中的数据的低 4 位结合，保存在 D 单元里的 4×S2 个位中。



2．当 S2 为 1 ～ 3 时，D 单元的高位中 4×（4 － S2）个位为 0。



使用示例

```
     M0                      1        17185
|───┤ ├───[ UNI   D0       D10       4    ]
```

```
LD   M0
UNI  D0 D10 4
```

当 M0 为 ON 时，执行 UNI 指令，将 D0 开始的连续 4 个单元中的数据的低 4 位结合，保存在 D10 单元中。

- 执行前：D0=16#1，D1=16#2，D2=16#3，D3=16#4，D10=0。
- 执行后：D0=16#1，D1=16#2，D2=16#3，D3=16#4，D10=16#4321。

注意事项

1．S2 的有效范围为 1 ～ 4，当 S2=0 时，PLC 不报错，指令不执行；当 S2<0 或 S2>4 时，PLC 信息中报“指令操作数数值非法”错误，指令不执行。

2．S1 和 D 不能重叠。

6.6.11 DIS : 16 位数据的四位分离指令

				适用机型	MX1H	MX2H	MX2V	MX3H
	DIS	P	S1 D S2		√	√	√	√
步长	16位：DIS (连续) 7步 32位： -----		DISP (脉冲) 8步 -----	相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT																●	●	*	●			●		
D	INT																●	●	*	●			●		
S2	INT																●							●	

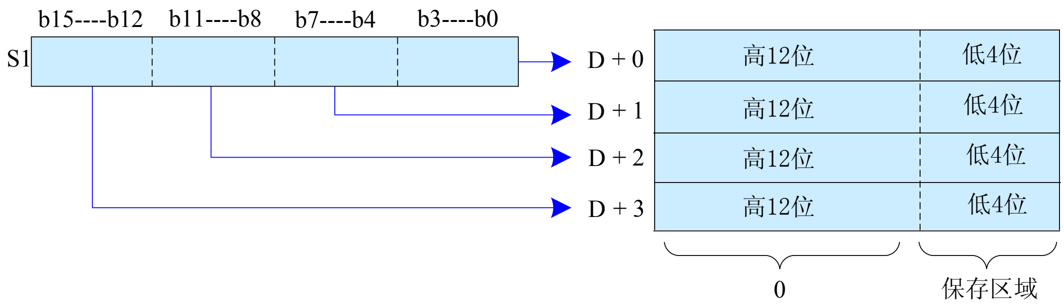
＊ 该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

[DIS S1 D S2]

S1：要分离的数据的软元件起始编号 **D**：已分离的数据的软元件起始编号 **S2**：分离数

将 S1 单元中的 4×S2 位数据每 4 位分为一组，保存到以 D 单元开始的 S2 个软元件里的低 4 位中。D 单元开始的 S2 个软元件中的高 12 位清零。



使用示例

```
M0          17185    1
─┤├──[ DIS  D0       D10    4 ]

LD   M0
DIS  D0 D10 4
```

当 M0 为 ON 时，执行 DIS 指令，将 D0 单元中的 16 位数据每 4 位分为一组，保存在 D10 单元开始的 4 个软元件里的低 4 位中。

- 执行前：D0=16#4321，D10=D11=D12=D13=0。
- 执行后：D0=16#4321，D10=16#1，D11=16#2，D12=16#3，D13=16#4。

注意事项

1. S2 的有效范围为 1 ～ 4，当 S2=0 时，PLC 不报错，指令不执行；当 S2<0 或 S2>4 时，PLC 信息中报“指令操作数数值非法”错误，指令不执行。
2. S1 和 D 不能重叠。

6.6.12 ANS : 信号报警器置位指令

	ANS	P	S1 S2 D	适应机型	MX1H	MX2H	MX2V	MX3H
						√	√	√
步长	16位：ANS (连续) 7步 ANSP (脉冲) 8步 32位： ----- -----			相关元件	SM400 SM401 SD401			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT																			*			•		
S2	INT																•							•	
D	INT				*																				

* 该指令操作数 S1 使用 T 元件时，只能使用 T0 ～ T209；操作数 D 使用 S 元件时，只能使用 S900 ～ S999。

指令说明

├──┤ ├──[ANS S1 S2 D]

S1：判断时间的定时器编号，只适用于 100ms 定时器（T0 ～ T209）

S2：判断时间的数据（0 ～ 32,767）

D：设置的信号报警器软元件（S900 ～ S999）

1．ANS 指令是用来驱动警报点输出的专用指令。

2．指令前触点闭合时间大于 S2 时置位 D，闭合时间小于 S2 时复位 S1 且不设置 D。

地址号	名称	功能
SM400	信号报警器使能标志	SM400 置 ON 后，下面的 SM401 和 SD401 工作
SM401	信号报警器动作标志	状态 S900 ～ S999 中任一动作，SM401 置 ON
SD401	信号报警器 ON 状态最小编号	保存 S900 ～ S999 中动作的最小编号

使用示例

```
    M0            32767        ON
|──┤ ├──[ ANS   T0     100    S900  ]
```

```
LD   M0
ANS  T0 100 S900
```

在 SM400 为 ON 的前提下，若 M0 闭合 10s 以上，则 S900 被置位，以后即使 M0 断开 S900 仍然保持接通状态；若 M0 闭合 10 s 以下，不置位 S900 且 T0 被清零。

6.6.13 ANR : 信号报警器复位指令

<table>
<tr><td rowspan="2"></td><td rowspan="2">ANR</td><td rowspan="2">P</td><td rowspan="2"></td><td rowspan="2">适应机型</td><td>MX1H</td><td>MX2H</td><td>MX2V</td><td>MX3H</td></tr>
<tr><td></td><td>√</td><td>√</td><td>√</td></tr>
<tr><td>步长</td><td colspan="3">16位：ANR (连续) 1步 ANRP (脉冲) 2步
32位： ----- -----</td><td>相关元件</td><td colspan="4">SM400 SM401 SD400</td></tr>
</table>

指令说明

示例 1.

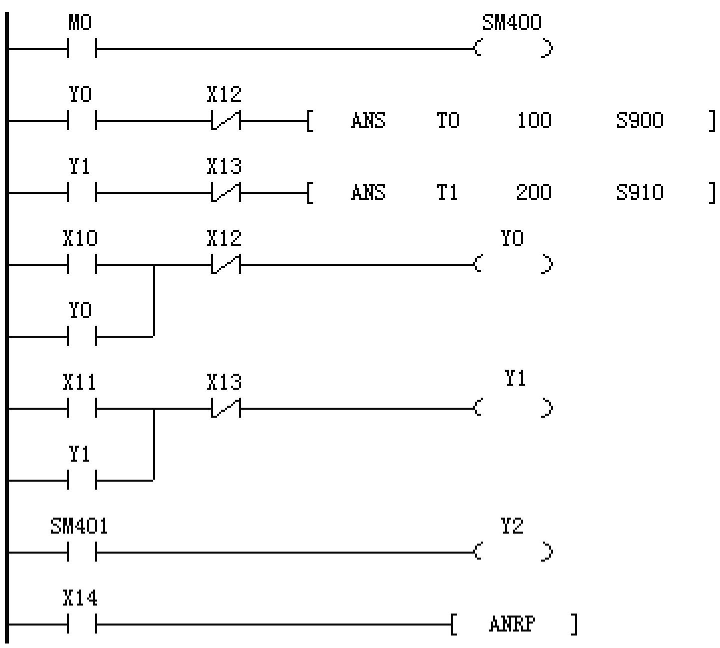

```
LD      M0
ANRP
```

M0 由 OFF → ON 时，ANRP 指令执行一次，信号报警器 S900 ～ S999 中正在运行的状态被复位。若有多个状态动作，则编号最小的状态被复位。当 M0 再次接通时，则下一个编号的状态被复位。

示例 2．警报点的应用

I/O 点配置：

输入点	状态	输出点	状态
X10	前进开关	Y0	前进
X11	后退开关	Y1	后退
X12	前端定位开关	Y2	警报指示器
X13	后端定位开关	S900	前进警报点
X14	警报点复位按钮	S910	后退警报点




```
LD      M0
OUT     SM400
LD      Y0
ANI     X12
ANS     T0 100 S900
LD      Y1
ANI     X13
ANS     T1 200 S910
LD      X10
OR      Y0
ANI     X12
OUT     Y0
LD      X11
OR      Y1
ANI     X13
OUT     Y1
LD      SM401
OUT     Y2
LD      X14
ANRP
```

1．当 SM400 有效时，SM401 和 SD401 才有效。
2．当 Y0=ON 超过 10s 而对象未到达前端定位开关 X12 时，S900 被置为 ON。
3．当 Y1=ON 超过 20s 而对象未到达后端定位开关 X13 时，S910 被置为 ON。
4．当前进开关 X10 为 ON 时，驱动前进输出 Y0 为 ON，直到对象到达前端定位开关 X12 时，Y0 才变为 OFF。
5．当后退开关 X11 为 ON 时，驱动后退输出 Y1 为 ON，直到对象到达后端定位开关 X13 时，Y1 才变为 OFF。
6．有警报点被驱动时，警报指示器 Y2 为 ON。当警报点的复位按钮 X14 每 ON 一次，动作中的编号状态就被复位一个，复位的顺序从较小的编号开始。

6.7 高速 I/O 指令

6.7.1 HCNT : 高速计数器驱动指令

	HCNT		D　　S	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	7步			相关元件	SM100～SM101　SM235～SM255			

操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z				Un.b
D	DINT																		*							
S	DINT										●	●	●	●	●	●	●	●	*		●			●	●	

＊　操作数 D 使用 C 元件时，只能使用 C235 ～ C255。操作数 S 使用 C 元件时，只能使用 C200 ～ C255。

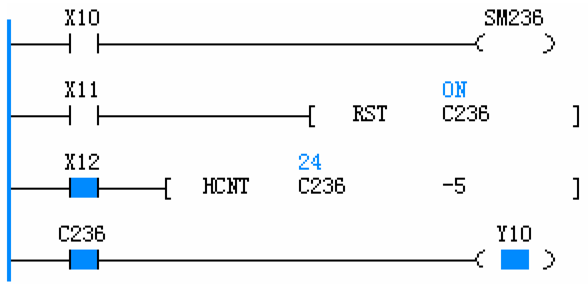
指令说明

├──┤ ├──[　HCNT　　D　　S　]

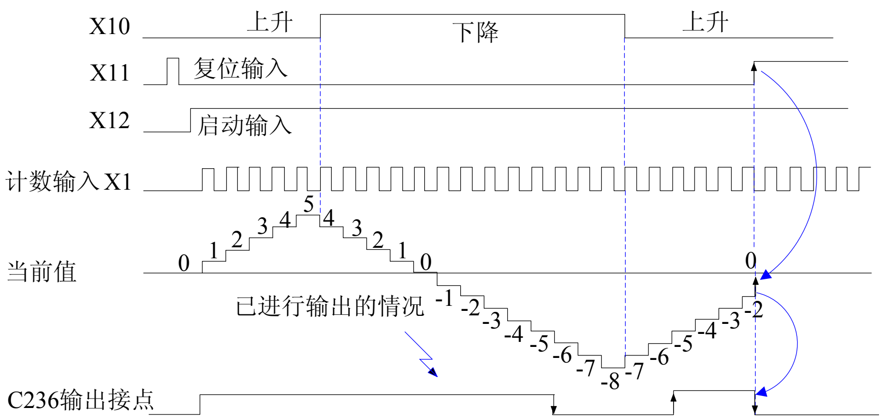
D：指定的高速计数器　　**S**：预设值

驱动指定的硬件高速计数器，所有的高速计数器只有在持续驱动的情况下，才能进行硬件高速计数，同时根据 S 对高速计数器的常开触点的动作进行判断。

使用示例




```
LD    X10
OUT   SM236
LD    X11
RST   C236
LD    X12
HCNT  C236 -5
LD    C236
OUT   Y10
```



1. X12 为 ON 时，执行 HCNT 指令，启动 C236 硬件高速计数器，X1 为 C236 的脉冲输入端，C236 对 X1 由 OFF → ON 以中断方式计数（X1 的输入频率请参见第八章高速 *I/O* 功能使用指南）。X12 为 OFF 时 X1 为一般的输入点，C236 不能对 X1 的外部脉冲计数。
2. 当计数器 C236 的当前值由 - 6 → - 5 增加时，C236 的触点被置位。当计数器 C236 的当前值由 - 5 → - 6 减小时，C236 的触点被复位。C236 的触点驱动 Y10 时，Y10 的执行受程序扫描周期的影响。
3. 当 X11 为 ON 时执行 RST 指令，C236 的计数值清零，C236 的触点断开。
4. X10 为 ON 时，SM236 为 ON，C236 为减计数；X10 为 OFF 时，SM236 为 OFF，C236 为增计数。

注意事项

HCNT 指令、SPD 指令、外部输入中断以及脉冲捕捉存在硬件冲突。请参见第八章 高速 *I/O* 功能使用指南。

6.7.2 DHSCS：高速计数比较置位指令

	DHSCS		S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位： ----- ----- 32位：DHSCS （连续） 10步 -----			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	DINT										●	●	●	●	●	●	●	●	*		●		●	●	
S2	DINT																		*						
D	BOOL		●	●	●																				

* 指令使用 C 元件时，操作数 S1 只能使用 C200 ～ C255，操作数 S2 只能使用 C235 ～ C255。

指令说明

├──┤ ├──[DHSCS S1 S2 D]

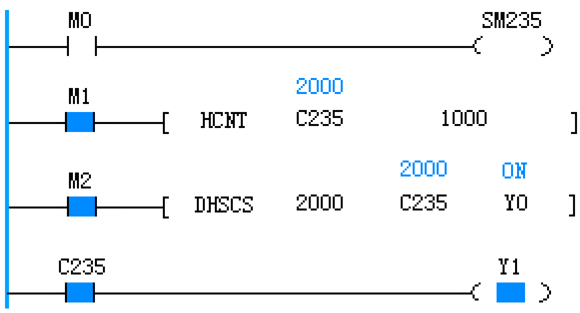
S1：高速计数器要比较的数据

S2：指定的高速计数器，高速计数器只能在 HCNT 指令驱动的条件下以中断方式计数。

D：输出位元件对象，对 Y，M，S 立即置位，输出不受扫描周期的影响。

当高速计数器的计数值等于 DHSCS 指令中的 S1 时，D 所指定的位元件立即置位。

使用示例




```
LD      M0
OUT     SM235
LD      M1
HCNT    C235 1000
LD      M2
DHSCS   2000 C235 Y0
LD      C235
OUT     Y1
```

1．M1 为 ON 时，C235 对 X0 由 OFF → ON 以中断的方式计数（X0 的输入频率请参见第八章高速 *I/O* 功能使用指南），当 C235 由 999 → 1000 时 C235 触点置位，由 1000 → 999 时 C235 触点复位。C235 的触点驱动 Y1 时，Y1 的执行受程序扫描周期影响。

2．M2 为 ON 时，执行 DHSCS 指令，当 C235 计数到 2000 时 Y0 立即输出 ON 状态，不受扫描周期影响。

3．M0 为 ON 时，SM235 为 ON，C235 计数器减计数。M0 为 OFF 时，SM235 为 OFF，C235 计数器增计数。

注意事项

1．DHSCS 指令必须要和 HCNT 指令配合使用，只有 HCNT 驱动后的高速计数器，DHSCS 才能真正使用。

2．DHSCS 指令在脉冲输入时比较结果动作。故即使使用 DMOV 或 MOV 指令等更改高速计数器值，DHSCS 也不会动作。

3．DHSCS（DHSCI，DHSCR，DHSZ，DHSP，DHST，DHSTI）和普通指令一样可以多次使用，但这些指令同时驱动个数限制在总计 6 条指令以下。对超过 6 条的指令不执行，而有效的指令按指令的有效先后决定。

4．PLC 高速计数器的最大允许频率在使用 DHSCS、DHSCI、DHSCR、DHSZ、DHSP、DHST、DHSTI 指令时，将受到很大的影响（请参见第八章高速 *I / O* 功能使用指南）。

6.7.3 DHSCR：高速计数比较复位指令

	DHSCR		S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位： ----- ----- 32位：DHSCR （连续） 10步 -----			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	DINT										●	●	●	●	●	●	●	●	*		●		●	●	
S2	DINT																		*						
D	BOOL		●	●	●		●																		

* 该指令使用 C 元件时，操作数 S1 只能使用 C200 ～ C255，操作数 S2 只能使用 C235 ～ C255。

指令说明

├──┤ ├──[DHSCR S1 S2 D]

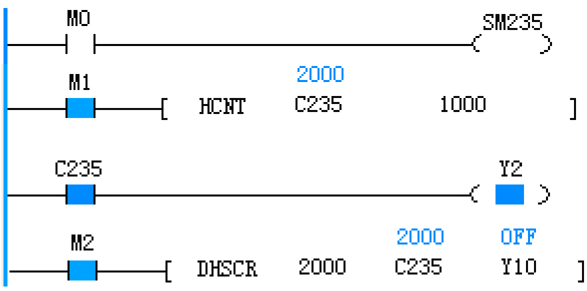
S1：高速计数器要比较的数据

S2：指定的高速计数器，高速计数器只能在 HCNT 驱动指令的条件下以中断方式计数。

D：输出位元件对象，对 Y、M、S、C 立即复位，不受扫描周期影响。

当高速计数器的值等于 DHSCR 指令中的 S1 时，D 所指定的位元件立即复位。

使用示例




```
LD     M0
OUT    SM235
LD     M1
HCNT   C235 1000
LD     C235
OUT    Y2
LD     M2
DHSCR  2000 C235 Y10
```

1. M1 为 ON 时，C235 对 X0 由 OFF → ON 以中断的方式计数（X0 的输入频率请参见第八章高速 *I/O* 功能使用指南），当 C235 由 999 → 1000 时 C235 触点置位，由 1000 → 999 时 C235 触点复位。C235 的触点驱动 Y2 时，Y2 的执行受扫描周期影响。
2. M2 为 ON 时，执行 DHSCR 指令，当 C235 计数到 2000 时 Y10 立即输出 OFF 状态，不受扫描周期影响。

注意事项

1. DHSCR 指令必须要和 HCNT 指令配合使用，只有 HCNT 驱动后的高速计数器，DHSCR 才能真正使用。
2. DHSCR 指令在脉冲输入时比较结果动作。故即使使用 DMOV 或 MOV 指令等更改高速计数器值，DHSCR 也不会有动作。
3. DHSCR（DHSCI，DHSCR，DHSZ，DHSP，DHST，DHSTI）和普通指令一样可以多次使用，但这些指令同时驱动个数限制在总计 6 条指令以下。对超过 6 条的指令不执行，而有效的指令按指令的有效先后决定。
4. PLC 高速计数器的最大允许频率。在使用 DHSCS、DHSCI、DHSCR、DHSZ、DHSP、DHST、DHSTI 指令时，将受到很大的影响（请参见第八章高速 *I / O* 功能使用指南）。

6.7.4 DHSCI : 高速计数比较中断触发指令

	DHSCI		S1 S2 S3	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位： ----- ----- 32位：DHSCI（连续） 10步 -----			相关元件	SM69			

操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z				Un.b
S1	DINT										●	●	●	●	●	●	●	●	*		●			●	●	
S2	DINT																		*							
S3	WORD																								●	

* 操作数 S1 使用 C 元件时，只能使用 C200 ～ C255。操作数 S2 使用 C 元件时，只能使用 C235 ～ C255。

指令说明

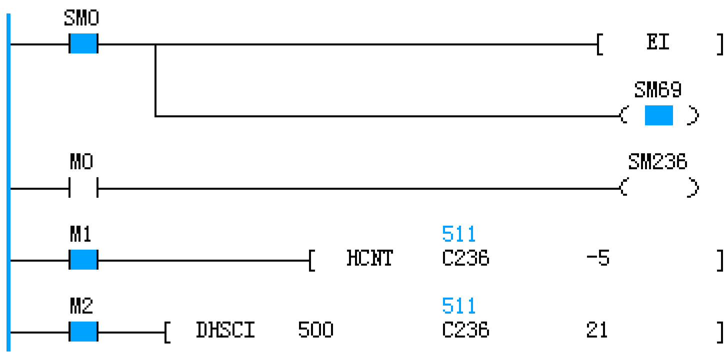
├──┤ ├──[DHSCI S1 S2 S3]

S1：高速计数器要比较的数据 **S2**：指定的高速计数器 **S3**：中断号（中断号范围 20 ～ 25）

高速计数器只能在 HCNT 驱动指令的条件下以中断方式计数，当高速计数器的值等于 DHSCI 指令中的 S1 时，执行 S3 所指定的中断服务子程序。

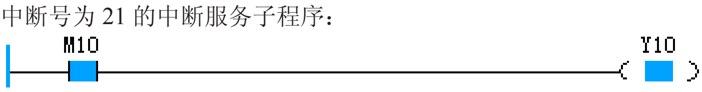
使用示例

主程序：




```
LD     SM0
EI
OUT    SM69
LD     M0
OUT    SM236
LD     M1
HCNT   C236 -5
LD     M2
DHSCI  500 C236 21
```

中断号为 21 的中断服务子程序：




```
LD     M10
OUT    Y10
```

1. 打开总中断，同时使 SM69（高速中断使能标志位）置位。当 M1 为 ON 时，C236 对 X1 由 OFF → ON 以中断的方式计数（X1 的输入频率请参见第八章高速 *I/O* 功能使用指南）。
2. 当 M0 为 ON 时，SM236 为 ON，C236 计数器减计数。当 M0 为 OFF 时，SM235 为 OFF，C236 计数器增计数。
3. 当 M2 为 ON 时，执行 DHSCI 指令，DHSCI 高速指令在满足注意事项所说的高速指令要求时，当 C236 等于 500 时马上响应中断号为 21 的中断子程序。在中断子程序中， M10 为 ON 时 Y10 驱动，但 Y10 的输出执行跟程序的扫描周期有关。

注意事项

1. DHSCI 指令必须要和 HCNT 指令配合使用，只有 HCNT 驱动后的高速计数器，DHSCI 才能使用。
2. DHSCI 指令在脉冲输入时比较结果动作。故即使使用 DMOV 或 MOV 指令等更改高速计数器值，DHSCR 也不会有动作。
3. DHSCI（DHSCS，DHSCR，DHSZ，DHSP，DHST，DHSTI）和普通指令一样可以多次使用，但这些指令同时驱动个数限制在总计 6 条指令以下。对超过 6 条的指令不执行，而有效的指令按指令的有效先后决定。
4. PLC 高速计数器的最大允许频率。在使用 DHSCS、DHSCI、DHSCR、DHSZ、DHSP、DHST、DHSTI 指令时，将受到很大的影响（请参见第八章高速 *I / O* 功能使用指南）。

6.7.5 DHSZ : 高速计数区间比较指令

	DHSZ		S1 S2 S3 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√		√
步长	16位： ----- 32位：DHSZ（连续） 13步		----- -----	相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	DINT										●	●	●	●	●	●	●	●	*		●		●	●	
S2	DINT										●	●	●	●	●	●	●	●	*		●		●	●	
S3	DINT																		*						
D	BOOL		●	●	●																				

＊ 操作数 S1、S2 使用 C 元件时，只能使用 C200 ～ C255；操作数 S3 使用 C 元件时，只能使用 C235 ～ C255。

指令说明

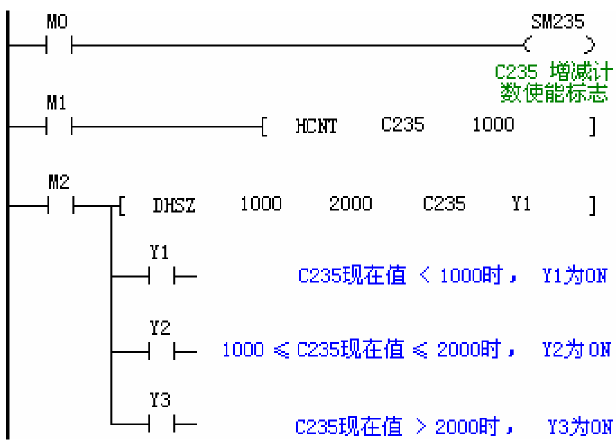
├──┤ ├──[DHSZ S1 S2 S3 D]

S1：高速计数器要比较的数据 1　**S2**：高速计数器要比较的数据 2

S3：指定的高速计数器　**D**：输出位元件对象，对 Y、M、S、C 立即复位，不受扫描周期影响。

1. 将高速计数器 S3 中的数值与 S1 中的数值和 S2 中的数值作比较，根据比较结果的不同使得 D、D+1、D+2 三个连号的位元件中对应的某一位元件置为 ON，另外两个位元件复位为 OFF（D 输出不受扫描周期影响）。
2. 当 S3<S1 时：D 元件置位，D+1 元件复位，D+2 元件复位；当 S1 ≤ S3 ≤ S2 时：D 元件复位，D+1 元件置位，D+2 元件复位；当 S3>S2 时：D 元件复位，D+1 元件复位，D+2 元件置位。
3. 如果 D 为 Y 元件，Y 元件马上输出相应的状态，输出动作与程序扫描周期无关。
4. 高速计数器只能在 HCNT 驱动指令的条件下根据计数输入信号由 OFF → ON 以中断方式计数。

使用示例



1. SM235 为高速计数器 C235 的增减计数使能位，为 OFF 时，C235 为增计数。
2. 当 M1 为 ON 时，X0（X0 为计数脉冲）由 OFF → ON 变化时，C235 增计数。
3. 当 M2 为 ON 时，执行 DHSZ 指令，此时根据 C235 中数值的大小，Y1、Y2、Y3 会有不同的输出。Y1、Y2、Y3 的输出不受扫描周期影响。

注意事项

1. DHSZ 指令必须要和 HCNT 指令配合使用，只有 HCNT 驱动后的高速计数器，DHSZ 才能真正使用。
2. DHSZ 指令在脉冲输入时比较结果动作。故即使使用 DMOV 或 MOV 指令等更改高速计数器值，DHSZ 也不会有动作。
3. DHSZ（DHSCI，DHSCS，DHSCR，DHSP，DHST，DHSTI）和普通指令一样可以多次使用，但这些指令同时驱动个数限制在总计 6 条指令以下。对超过 6 条的指令不执行，而有效的指令按指令的有效先后决定。
4. PLC 高速计数器的最大允许频率。在使用 DHSCS、DHSCI、DHSCR、DHSZ、DHSP、DHST、DHSTI 指令时，将受到很大的影响（请参见第八章高速 *I/O* 功能使用指南）。

6.7.6 DHST : 高速计数表格比较指令

	DHST		S1 S2 S3	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：DHST (连续) 10步 32位： -----		----- -----	相关元件	SM185 SD182 ～ SD184			

操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z				Un.b
S1	DINT																●									
S2	INT																								●	
S3	DINT																		*							

* 该指令使用 C 元件时，只能使用 C235 ～ C255。

指令说明

├──┤ ├──[DHST S1 S2 S3]

S1：表格比较的数据起始单元（D 元件起始号）。其后序列号相连的三个 D 元件用于指定高速计数器要比较的数据（有高低位）、Y 元件序号及其相应的输出状态。这四个序列号相连的 D 元件合称一条记录。

S2：要比较的记录数量，数据范围为 1 ～ 128。

S3：高速计数器，适用范围为 C235 ～ C255。

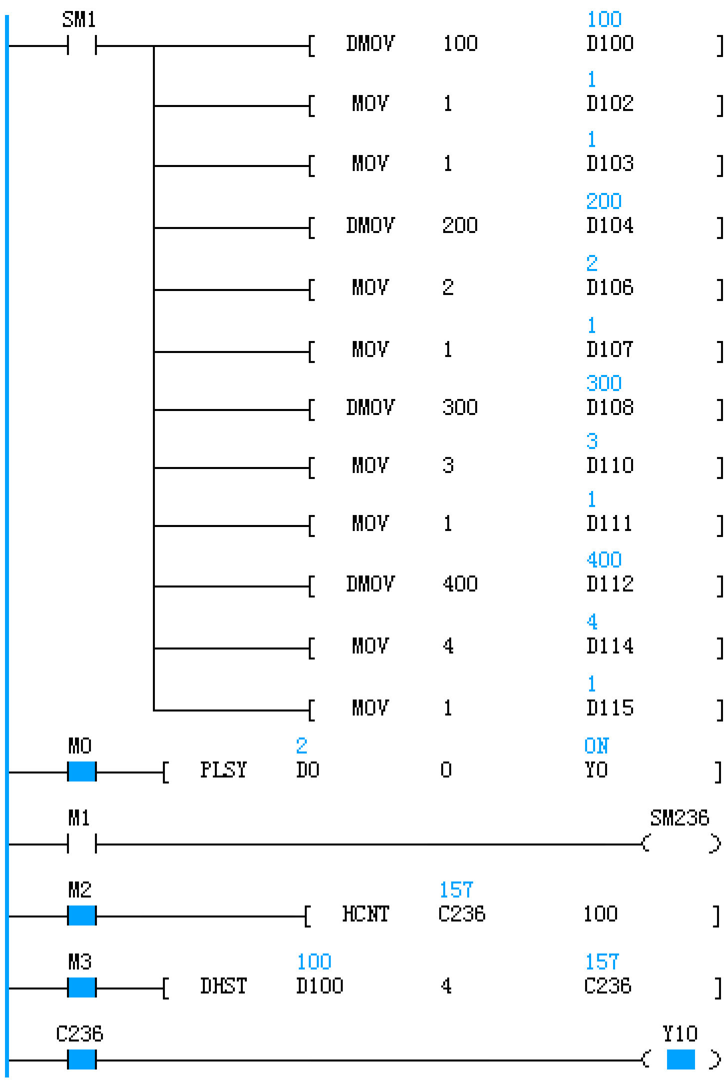
1. 当高速计数器的值等于当前要比较记录的数据，则按记录的数据输出相应的 Y 元件状态，输出的对象只能为 Y 元件。
2. 输出动作与扫描周期无关，当前记录指定的 Y 元件将立即输出指定的状态。
3. 在希望程序按某个表格指定的比较数据和 Y 元件执行立即输出的操作时，使用 DHST 表格比较输出指令。
4. 高速计数器只有在 HCNT 指令驱动的条件下，根据计数输入由 OFF → ON 以中断方式计数。

使用示例

表格数据如下所示：

比较数据		输出Y编号	置位/复位	操作流程
高位	低位			
D101=0	D100=100	D102=1	D103=1	1 ↓
D105=0	D104=200	D106=2	D107=1	2 ↓
D109=0	D108=300	D110=3	D111=1	3 ↓
D113=0	D112=400	D114=4	D115=1	4 返回到1

梯形图如下所示：




```
LD      SM1
DMOV    100 D100
MOV     1 D102
MOV     1 D103
DMOV    200 D104
MOV     2 D106
MOV     1 D107
DMOV    300 D108
MOV     3 D110
MOV     1 D111
DMOV    400 D112
MOV     4 D114
MOV     1 D115
LD      M0
PLSY    D0 0 Y0
LD      M1
OUT     SM236
LD      M2
HCNT    C236 100
LD      M3
DHST    D100 4 C236
LD      C236
OUT     Y10
```

1. 程序的第一个扫描周期，对 D100 ～ D115 赋初值，产生要比较的表格。
2. 当 M1 为 ON 时，SM236 为 ON，C236 为减计数；当 M1 为 OFF 时，SM236 为 OFF，C236 为增计数。
3. 当 M0 和 M2 为 ON 时，高速计数器 C236 由 OFF → ON 以中断方式计数 （X1 输入频率请参见第八章高速 *I/O* 功能使用指南），当 C236 由 99 → 100 时 C236 触点置位，由 100 → 99 时 C236 触点复位。C236 的触点驱动 Y10 时，Y10 的执行由用户程序的扫描周期决定。
4. 当 M3 为 ON 时，执行 DHST 指令，DHST 高速指令在满足注意事项所说的高速指令要求时，从表格的 1 号记录开始进行比较，每次只有在前一条记录比较完成后才进入下一条记录比较。当最后一条记录比较完成后，重新回到第一条记录比较，同时表格比较标志（SM185）置位。对比较完成的结果立即输出，不受扫描周期影响。SD184 表示当前要比较的记录号，（SD183，SD182）表示当前要比较的数据。

注意事项

1. DHST（DHSCI，DHSCS，DHSCR，DHSP，DHSZ，DHSTI）和普通指令一样可以多次使用，但这些指令同时驱动的个数限制在总计 6 条以下。对超过 6 条的有效指令不执行，有效的指令按指令的有效先后决定。
2. 在用户的指令中如果 DHSP 为有效指令，DHST 就不执行。反之 DHST 为有效指令，DHSP 就不执行。程序中在同一时刻只能有一条指令（DHST 或 DHSP）有效。
3. PLC 高速计数器的最大允许频率在使用 DHSCS、DHSCI、DHSCR、DHSZ、DHSP、DHST、DHSTI 指令时，将受到很大的影响（请参见第八章高速 *I/O* 功能使用指南）。
4. DHST 指令动作必须和 HCNT 指令配合使用，只有 HCNT 驱动后的高速计数器，DHST 才能正确执行。
5. DHST 指令在脉冲输入时比较结果动作。因此，即使使用 DMOV 或 MOV 指令等更改高速计数器值，DHST 也不会有动作。

6.7.7 DHSTI：高速计数表格比较中断触发指令

				适用机型	MX1H	MX2H	MX2V	MX3H
	DHSTI		S1 S2 S3		√	√	√	√
步长	16位： ----- 32位：DHSTI (连续) 13步		----- -----	相关元件	SM69			

操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z				Un.b
S1	DINT																●									
S2	DINT																		*							
S3	WORD																								●	

* 该指令使用 C 元件时，只能使用 C235 ～ C255。

指令说明

DHSTI S1 S2 S3

S1：高速计数器要比较的数据起始单元（D 元件起始号） **S2**：高速计数器，适用范围为 C235 ～ C255

S3：比较中断数，范围为 1 ～ 24，相应的中断号依次为 70 ～ 93

高速计数器只能在 HCNT 驱动指令的条件下以中断方式计数，当高速计数器的值等于 DHSTI 指令中的 S1 时，执行相应的中断服务子程序。

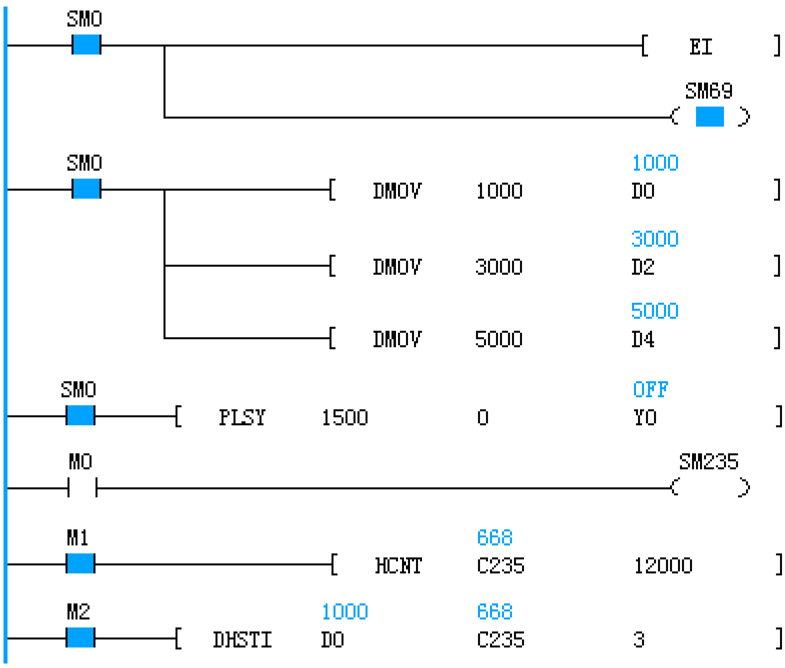
使用示例

比较数据和触发中断号如下表所示：

比较数据		触发中断号	触发顺序
高位	低位		
D1=0	D0=1000	70	1 ↓
D3=0	D2=3000	71	2 ↓
D5=0	D4=5000	72	3 ↓

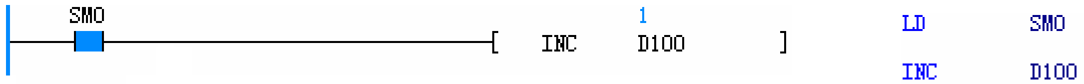
梯形图如下所示：

主程序：

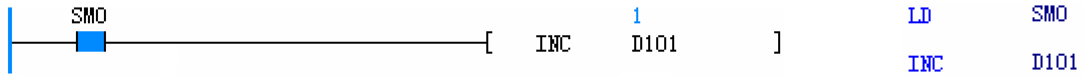



```
LD     SM0
EI
OUT    SM69
LD     SM0
DMOV   1000 D0
DMOV   3000 D2
DMOV   5000 D4
LD     SM0
PLSY   1500 0 Y0
LD     M0
OUT    SM235
LD     M1
HCNT   C235 12000
LD     M2
DHSTI  D0 C235 3
```

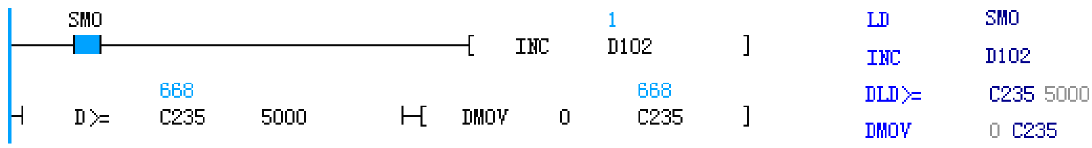
中断号为 70 的中断服务子程序：



中断号为 71 的中断服务子程序：



中断号为 72 的中断服务子程序：



1. 打开总中断，同时使 SM69（高速计数中断使能标志）置位，当 M1 为 ON 时，C235 对 X0 由 OFF → ON 以中断的方式计数（X0 的输入频率请参见第八章高速 *I/O* 功能使用指南）。
2. 当 M0 为 ON 时，SM235 为 ON，C235 为减计数；当 M0 为 OFF 时，SM235 为 OFF，C235 为增计数。
3. 当 M2 为 ON 时，执行 DHSTI 指令，DHSTI 高速指令在满足注意事项所说的高速指令要求时，当 C235 等于 1000、3000、5000 时马上分别响应中断号为 70、71、72 的中断服务子程序，执行子程序中的加 1 指令。
4. 当 C235 等于 5000 时，将 C235 计数值清零，重新开始计数，再依次触发中断。

注意事项

1. 中断号的范围为 70 ～ 93，添加高速计数器多比较中断时，中断号必须为 70 开始的连续中断，且添加的中断个数必须为 S3 个，否则编译不通过，报“高速计数器多比较中断，中断程序配置错误”。
2. 当比较数据相等，或数据顺序不是由小到大时，也是按照中断号依次触发。例如，将上述示例中的数据改为（D1，D0）=3000，（D3，D2）=1000，（D5，D4）=1000，当高速计数器计数值第一次等于，3000 时进入中断号为 70 的中断服务子程序；当高速计数器值清零后再次计数到 1000 时，进入中断号为 71 的中断服务子程序；当高速计数器值第二次清零后再次计数到 1000 时，进入中断号为 72 的中断服务子程序，依次循环。
3. DHSTI（DHST，DHSCI，DHSCS，DHSCR，DHSP，DHSZ）和普通指令一样可以多次使用，但这些指令同时驱动的个数限制在总计 6 条以下。对超过 6 条的有效指令不执行，有效的指令按指令的有效先后决定。
4. PLC 高速计数器的最大允许频率在使用 DHSCS、DHSCI、DHSCR、DHSZ、DHSP、DHST、DHSTI 指令时，将受到很大的影响（请参见第八章高速 *I /O* 功能使用指南）。
5. DHSTI 指令动作必须和 HCNT 指令配合使用，只有 HCNT 驱动后的高速计数器，DHSTI 才能正确执行。
6. DHSTI 指令在脉冲输入时比较结果动作。因此，即使使用 DMOV 或 MOV 指令等更改高速计数器值，DHSTI 也不会有动作。

6.7.8 DHSP : 高速计数表格比较脉冲输出指令

				适用机型	MX1H	MX2H	MX2V	MX3H
	DHSP		S1 S2 S3		√	√	√	√
步长	10步			相关元件	SM185 SD180 ~ SD184			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	DINT																●								
S2	INT																							●	
S3	DINT																		*						

* 该指令使用 C 元件时，只能使用 C235 ~ C255。

指令说明

├──┤ ├──[DHSP S1 S2 S3]

S1：表格比较的数据起始单元（D 元件起始号）。其后序列号相连的三个 D 元件用于指定高速计数器要比较的数据，以及输出到 SD180 和 SD181 的数据。这四个序列号相连的 D 元件合称一条记录。

S2：要比较的记录数量，数据范围为 1 ~ 128。

S3：高速计数器，适用范围为 C235 ~ C255。

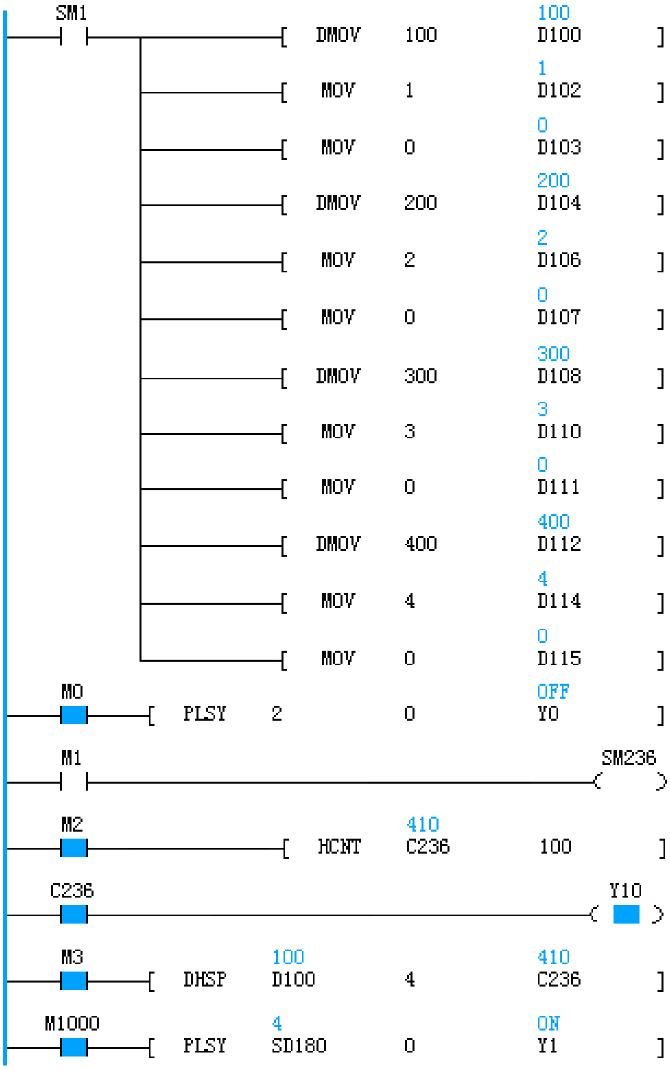
1. 当高速计数器的值等于当前要比较记录的数据时，则按当前记录的输出数据更改 SD180 和 SD181。
2. 在希望程序按某个表格决定高速输出或其它数据的赋值时，使用 DHSP 表格比较输出指令。例如可指定（SD181，SD180）为 PLSY 指令的输出频率操作数，使 PLSY 输出频率按表格比较结果进行调整。
3. DHSP 和 PLSY 配合使用时送到（SD181，SD180）的数据要满足 PLSY 的频率输出（请参见 *6.7.10 PLSY* 高速脉冲输出指令）。
4. 当希望比较在最后一条记录停止时，把表格中最后一条记录送到（SD181，SD180）的数据设为 0。在此情况下，其它的 DHST 和 DHSP 指令无效，但此时的 DHSP 指令不占用其它高速指令的总条数。
5. DHSP 指令在脉冲输入时比较结果动作。因此，即使使用 DMOV 或 MOV 指令等更改高速计数器值，DHSP 也不会有动作。
6. 高速计数器只有在 HCNT 指令驱动的条件下，根据计数输入由 OFF → ON 以中断方式计数。
7. DHSP 指令必须和 HCNT 指令配合使用，只有 HCNT 驱动后的高速计数器，DHSP 指令才能正确执行。

使用示例

表格数据如下所示：

比较数据		输出到SD181和SD180的数据		操作流程
高位	低位	高位	低位	
D101=0	D100=100	D103=0	D102=1	1 ↓
D105=0	D104=200	D107=0	D106=2	2 ↓
D109=0	D108=300	D111=0	D110=3	3 ↓
D113=0	D112=400	D115=0	D114=4	4 返回到1

梯形图如下所示：




```
LD     SM1
DMOV   100 D100
MOV    1 D102
MOV    0 D103
DMOV   200 D104
MOV    2 D106
MOV    0 D107
DMOV   300 D108
MOV    3 D110
MOV    0 D111
DMOV   400 D112
MOV    4 D114
MOV    0 D115
LD     M0
PLSY   2 0 Y0
LD     M1
OUT    SM236
LD     M2
HCNT   C236 100
LD     C236
OUT    Y10
LD     M3
DHSP   D100 4 C236
LD     M1000
PLSY   SD180 0 Y1
```

1. 程序的第一个扫描周期，对 D100 ～ D115 赋初值，产生要比较的表格。
2. 当 M1 为 ON 时，SM236 为 ON，C236 为减计数；当 M1 为 OFF 时，SM236 为 OFF，C236 为增计数。
3. 当 M0 和 M2 为 ON 时，高速计数器 C236 以中断方式计数（X1 输入频率请参见第八章高速 *I/O* 功能使用指南），当 C236 由 99 → 100 时 C236 触点置位，由 100 → 99 时 C236 触点复位。C236 的触点驱动 Y10 时，Y10 的执行由用户程序的扫描周期决定。
4. 当 M3 为 ON 时，执行 DHSP 指令，DHSP 指令在满足注意事项所说的高速指令要求时，从表格的 1 号记录开始进行比较，每次只有在前一条记录比较完成后才进入下一条记录比较。当最后一条记录比较完成后，重新回到第一条记录比较，同时表格比较标志（SM185）置位。对比较完成的结果分别放到 SD180 和 SD181 单元中，不受扫描周期影响。SD184 表示当前要比较的记录号，SD182 和 SD183 表示当前要比较的数据。
5. 指定（SD181，SD180）为 PLSY 指令的输出频率操作数。当 M1000 为 ON 时，表格比较当前记录的输出数据传送至 SD180 和 SD181，使得 PLSY 输出频率按表格比较结果进行调整。

注意事项

1. DHSP（DHSCI，DHSCS，DHSCR，DHST，DHSZ，DHSTI）和普通指令一样可以多次使用，但这些指令同时驱动的个数限制在总计 6 条以下。对超过 6 条的有效指令不执行，指令的有效性按其有效先后决定。
2. 在用户的指令中如果 DHSP 为有效指令，DHST 就不执行。反之 DHST 为有效指令，DHSP 就不执行。对用户程序中在同一时刻只能有一条指令（DHST 或 DHSP）有效，其它无效。
3. 注意 PLC 高速计数器的最大允许频率，在使用 DHSCS、DHSCI、DHSCR、DHSZ、DHSP、DHST、DHSTI 指令时，将受到很大的影响（请参见第八章高速 *I/O* 功能使用指南）。

6.7.9 SPD : 测频指令

D	SPD		S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
					*	*	√	√
步长	16位：SPD (连续) 7步 32位：DSPD (连续) 9步			相关元件				

操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z				Un.b
S1	BOOL	*																								
S2	(D)WORD										•	•	•	•	•	•	•	•	*	*	•	*	•	•	
D	(D)WORD																•						•		

* MX1H、MX2H 系列只能使用 16 位的 SPD 指令。

* MX1H、MX2H 使用 X 元件时，只能使用 X0 ～ X5。

* MX2V、MX3H 使用 X 元件时，只能使用 X0 ～ X7。

* 16 位指令的 S2 使用 C 元件时，只能使用 C0 ～ C199。
32 位指令的 S2 使用 C 元件时，只能使用 C200 ～ C255 且不能使用 T、Z 元件。

指令说明

├──┤ ├──[SPD S1 S2 D]

S1：输入端口

MX1H、MX2H 可设定范围：X0 ～ X5（不能与高速计数器及中断输入重复使用）。

MX2V、MX3H 可设定范围：X0 ～ X7（不能与高速计数器及中断输入重复使用）。

S2：输入点检测的单位时间，以 ms 为单位，S2>0。

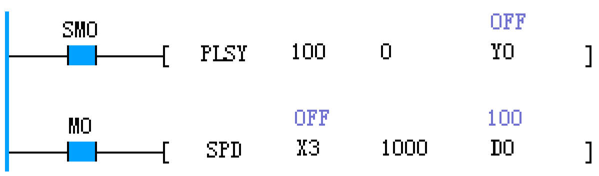
D：检测脉冲数据保存单元

16 位指令：当计数超过 65,535 时，自动溢出处理。

32 位指令：当计数超过 4,294,967,295 时，自动溢出处理。

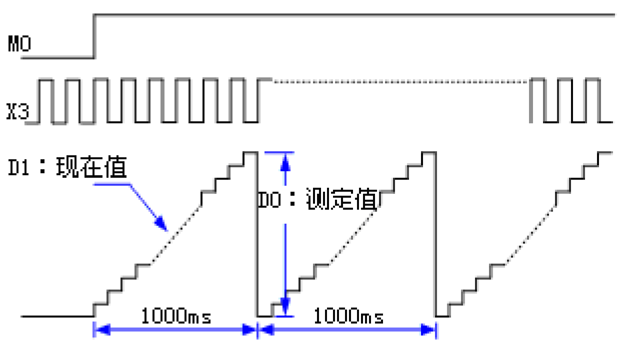
对 S1 指定的输入脉冲在 S2 指定的时间内计数，其结果存入 D 指定的软元件中，其中 D+1 为在 S2 时间内计数的现在值，D+2 为在 S2 时间内已运行的时间。

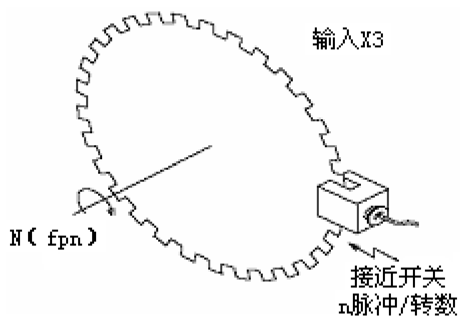
使用示例

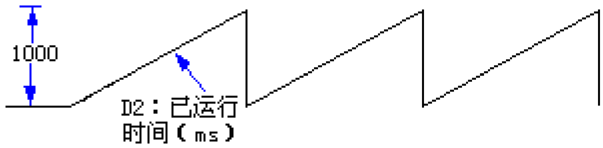



```
LD    SM0
PLSY  100  0  Y0
LD    M0
SPD   X3  1000  D0
```

对程序的操作实例的时序操作如下：







1．M0 为 ON 时，执行 SPD 指令，将 X3 指定的输入脉冲在 1000ms 内计数，将计数的结果保存在 D0 的存储单元中，其中 D1 为在 1000ms 内计数的现在值，D2 为 1000ms 内已运行的时间。

2．D0 的数据和上图中的旋转速度成正比关系：

$$N = \frac{[D0]}{nt} \times 10\text{^}3 \times 60 \text{（r/min）}$$

（N：转速　n：旋转一圈所产生脉冲数　t：S2 输入的时间宽度）

3．将 X3 在 1000ms 内的脉冲数保存到 D0 中。

6.7.10 PLSY : 高速脉冲输出指令

	PLSY		S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位： ----- -----			相关元件	SM80～SM83 SD50～SD53			
	32位：PLSY （连续） 9步 -----				SM261～SM272 SD70～SD81			

操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z				Un.b
S1	DINT										•	•	•	•	•	•	•	•	*		•			•	•	
S2	DINT										•	•	•	•	•	•	•	•	*		•			•	•	
D	BOOL		*																							

* 该指令使用 C 元件时，只能使用 C200 ～ C255。
使用位元件 Y 时，对于 MX1H 指定 Y0、Y1；MX2H 指定 Y0、Y1、Y2；MX2V 指定 Y0 ～ Y7。

指令说明

```
|—| |—[ PLSY   S1   S2   D ]
```

S1：指定频率（Hz）（MX1H/2H：0 ≤ S1 ≤ 100,000；MX2V：0 ≤ S1 ≤ 200,000）

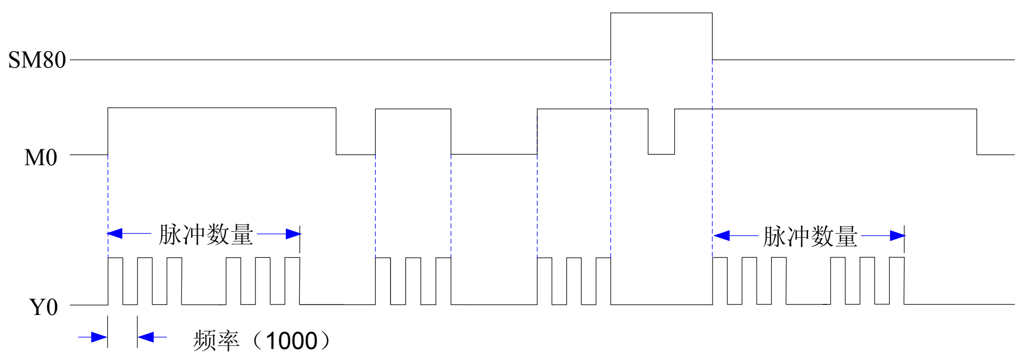
S2：产生的脉冲量（PLS）（0 ≤ S2 ≤ 2,147,483,647）

D：高速脉冲输出点。MX1H 指定 Y0、Y1；MX2H 指定 Y0、Y1、Y2；MX2V 指定 Y0 ～ Y7。

根据指令指定的频率产生指定数量的高速脉冲输出。

使用示例

```
   M0
|—| |—[ PLSY  1000  10000  Y0 ]        LD    M0
                           ON          PLSY  1000 10000 Y0
```



1．M0 为 ON 时，执行 PLSY 指令，从 Y0 端口输出频率为 1000Hz 的脉冲，输出 10000 个脉冲后，不再输出。M0 为 OFF 或 SM80 为 ON 时，端口输出 Y0 为 OFF。当 M0 再次为 ON 且 SM80 为 OFF 时，重新输出 10000 个脉冲后停止。

2．脉冲的占空比为 50％，输出控制不受扫描周期的影响，采用中断处理。在高频输出时，端口的输出占空比跟负载有关系。从输出端子得到的波形跟输出负载有关系，在不超过额定负载电流情况下，负载越大，输出波形越接近设定操作数。

3. SM80～SM81，SM261～SM266：Y0～Y1，Y2～Y7 的输出停止使能位，为 ON 的情况下停止输出脉冲。

4．SM82 ～ SM83，SM267 ～ SM272：Y0 ～ Y1，Y2 ～ Y7 的脉冲输出监控位，当输出完成或 M0 为 OFF 时监控位清除。

5．（SD51，SD50），（SD53，SD52）：Y0 ～ Y1 在 PLSY 和 PLSR 指令的脉冲输出数量（需手动清零）。（SD71，SD70）～（SD81，SD80）：Y2 ～ Y7 在 PLSY 和 PLSR 指令的脉冲输出数量（需手动清零）。

6．SD50 ～ SD53，SD70 ～ SD81 可以通过 DMOV 或 MOV 更改，也可通过监控更改。

7．如果要使用输入脉冲个数控制 PLSY 的输出脉冲频率，具体请参见 *6.7.8 DHSP* 指令。

注意事项

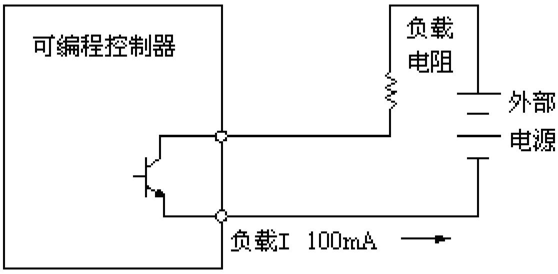
1．S1 的取值范围为 0 ～ 100,000 Hz（MX1H/2H）、0 ～ 200,000 Hz（MX2V），S2 的取值范围为 0 ～ 2,147,483,647，否则 PLC 信息中报“指 令操作数数值非法”错误，指令不执行，同时不占系统硬件资源。S1 为 0 时，PLC 信息不报错，只是不发脉冲；S2 为 0 时，在指令有效下脉冲始终输出。

2．PLC 必须使用晶体管输出方式。

3．当 PLSY 指令执行后，Y0 开始做脉冲输出。此时，若改变 S2（脉冲输出数目），对目前的输出是没有影响的，须先将 PLSY 指令停止，然后再改变脉冲输出数目。S1（频率）可在 PLSY 指令执行时更改，更改后的 S1 在程序执行到 PLSY 指令时有效。

4．PLC 执行高速脉冲输出时，必须使用下图所示的 PLC 输出晶体管规定的负载电流。

5．针对 PLSY，PWM，PLSR 的输出回路（晶体管）如下：



最小负载为：5mA（5 ～ 24VDC）

6．在高负载时晶体管的 OFF 时间较长，在 PWM、PLSY、PLSR 指令时，要求晶体管输出端接相应的负载，当输出的波形不满足指令的操作数时，可以加大晶体管的负载电流（晶体管的负载≤ 100mA），负载越大，输出波形越接近设定操作数。

7．在高速指令有效运行（包括输出完成）时，对同一端口的其它操作无效。只有在高速指令无效时，其它指令才能操作本端口。

8．使用多个 PLSY 指令能够在高速输出点得到各自独立的高速脉冲输出，也可和 PWM 或 PLSR 在不同的输出点得到各自独立的高速脉冲输出。

9．有多条 PWM、PLSY 或 PLSR 指令操作同一端口时，先有效的指令控制端口输出状态，后有效的指令对输出点的状态无影响。

10．与其它高速指令（DHSCS，DHSCR，DHSCI，DHSZ，DHSP，DHST，DHSTI，HCNT）相同，PLSY 指令要满足系统中对高速输入和高速脉冲输出的要求。

6.7.11 PLSR : 带加减速的计数脉冲输出指令

	PLSR		S1 S2 S3 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	10步			相关元件	SM80～SM83 SD50～SD53 SM261～SM272 SD70～SD81			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	WORD										●	●	●	●	●	●	●	●	*	●	●	●	●	●	●
S2	DINT										●	●	●	●	●	●	●	●	*		●		●	●	
S3	WORD										●	●	●	●	●	●	●	●	*	●	●	●	●	●	●
D	BOOL		*																						

* S1，S3 使用 C 元件时，只能使用 C0 ～ C199；S2 使用 C 元件时，只能使用 C200 ～ C255。
使用位元件 Y 时，对于 MX1H 指定 Y0、Y1；MX2H 指定 Y0、Y1、Y2；MX2V 指定 Y0 ～ Y7。

指令说明

├──┤ ├──[PLSR S1 S2 S3 D]

S1：最高频率（Hz）（10 ≤ S1 ≤ 20,000）

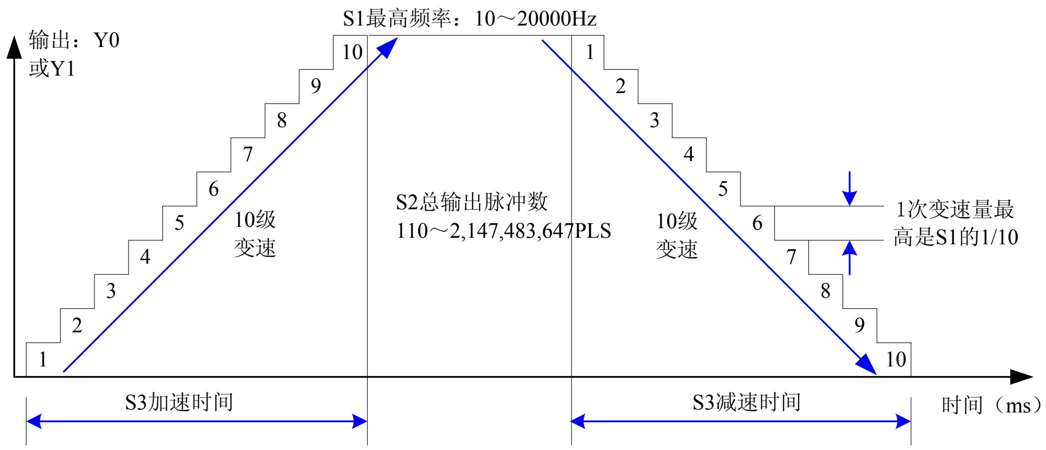
S2：总输出脉冲数（PLS）（12 ≤ S2 ≤ 2,147,483,647）

S3：加减速时间（ms）（1 ≤ S3 ≤ 10,000）。操作数 S1×S3<100,000 时，系统按 S3=100,000/S1 处理，同时 PLC 信息提示“PLSR 指令参数错误”，加减速时序不一定。操作数 S1×S3>S2×909 时。系统按 S3=S2×909/S1 处理，同时系统提示“PLSR 指令参数错误”，加减速时序不一定。加减速时的变速次数按固定的 10 次处理，每次变化量为 S1/10。

D：高速脉冲输出点。MX1H 指定 Y0、Y1；MX2H 指定 Y0、Y1、Y2；MX2V 指定 Y0 ～ Y7。

PLSR 指令为带有加减速功能的定尺寸传送用的高速脉冲输出指令。脉冲从静止状态到目标速度作加速动作，快到达目标速度时，作减速动作，到达目标距离时，脉冲停止输出。

操作过程如下：



使用示例

```
M0                                       OFF
──┤ ├──[ PLSR   10     5000    1000     Y0    ]
                10     5000    1000     ON
M1
──┤ ├──[ PLSR   D0     D10     D20      Y1    ]
```

```
LD      M0
PLSR    10 5000 1000 Y0
LD      M1
PLSR    D0 D10 D20 Y1
```

1．当 M0 为 ON 时，执行 PLSR 指令，Y0 端口输出最高频率为 10HZ，加减速时间为 1000ms 的脉冲，完成 5000 个脉冲后停止输出，M0 为 OFF 或 SM80 为 ON 时，端口输出 OFF。

2．当 M1 为 ON 时，执行 PLSR 指令，Y1 端口输出最高频率为 10HZ，加减速时间为 1000ms 的脉冲，完成 5000 个脉冲后停止输出，M1 为 OFF 或 SM81 为 ON 时，端口输出 OFF。

3．在指令执行的过程中操作数若有更改，系统仍按原有效的操作数处理，只有 M0 由 ON → OFF → ON 变化时新操作数才有效。

4. SM80 ～ SM81，SM261 ～ SM266：Y0 ～ Y1，Y2 ～ Y7 的输出停止使能位，为 ON 的情况下停止输出脉冲。

5．SM82 ～ SM83，SM267 ～ SM272：Y0 ～ Y1，Y2 ～ Y7 的脉冲输出监控位，当输出完成或 M0 为 OFF 时监控位清除。

6．（SD51，SD50），（SD53，SD52）：Y0 ～ Y1 在 PLSY 和 PLSR 指令的脉冲输出数量（需手动清零）。（SD71，SD70）～（SD81，SD80）：Y2 ～ Y7 在 PLSY 和 PLSR 指令的脉冲输出数量（需手动清零）。

7．SD50 ～ SD53，SD70 ～ SD81 可以通过 DMOV 或 MOV 更改，也可通过监控更改。

注意事项

1．S1 的取值范围为 1 ～ 200,000（Hz），设定操作数不在本范围之内时，小于 10 时按 10 处理，大于 20,000 时按 20,000 处理，PLC 信息提示“PLSR 指令参数错误”，高速脉冲输出按默认的数据输出。

2．S2 可设定范围为 12 ～ 2,147,483,647，设定操作数不在本范围之内时，PLC 信息中报“指令操作数数值非法”错误，同时脉冲不输出，也不占用本指令对应的系统硬件资源。

3．该命令的输出频率为 10 ～ 20000Hz，加减速时的变速速度超过此范围时，自动在范围内调整操作数，本指令不受扫描周期的影响。

4．当 PLSR 指令执行时，改变操作数 S1、S2 或 S3 并不影响输出。若要改变输出，须先将 PLSR 指令停止，然后才改变输出。

5．PLC 请使用晶体管输出，在高速脉冲输出时请按规定的负载电流接入输出晶体管。从输出端子得到的波形跟输出负载有关系，在不超过额定负载电流情况下，负载越大，输出波形越接近设定操作数。

6．在高速指令有效运行（包括输出完成）时，对同一端口的其它操作无效。只有在高速指令无效时，其它指令才能操作本端口。

7．使用 2 个 PLSR 指令能够在输出端得到各自独立的高速脉冲输出，也可和 PWM 或 PLSY 在不同的输出端口得到各自独立的高速脉冲输出。

8．有多条 PWM、PLSY 或 PLSR 指令操作同一端口时，先有效的指令控制端口输出状态，后有效的对输出端的状态无影响。

9．与其它高速指令（DHSCS，DHSCR，DHSCI，DHSZ，DHSP，DHST，DHSTI，HCNT）相同，PLSR 指令要满足系统中对高速输入和高速脉冲输出的要求。

6.7.12 PLS：高速包络线输出指令

	PLS		S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：PLS (连续) 7步 32位： ----- -----			相关元件	SM80～SM83 SD56～SD58			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●	●
S2	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●	●
D	BOOL		*																						

* 该指令使用 C 元件时，只能使用 C0 ～ C199。
使用位元件 Y 时，对于 MX1H 指定 Y0、Y1；MX2H 指定 Y0、Y1、Y2；MX2V 指定 Y0 ～ Y7。

指令说明

├──┤ ├──[PLS S1 S2 D]

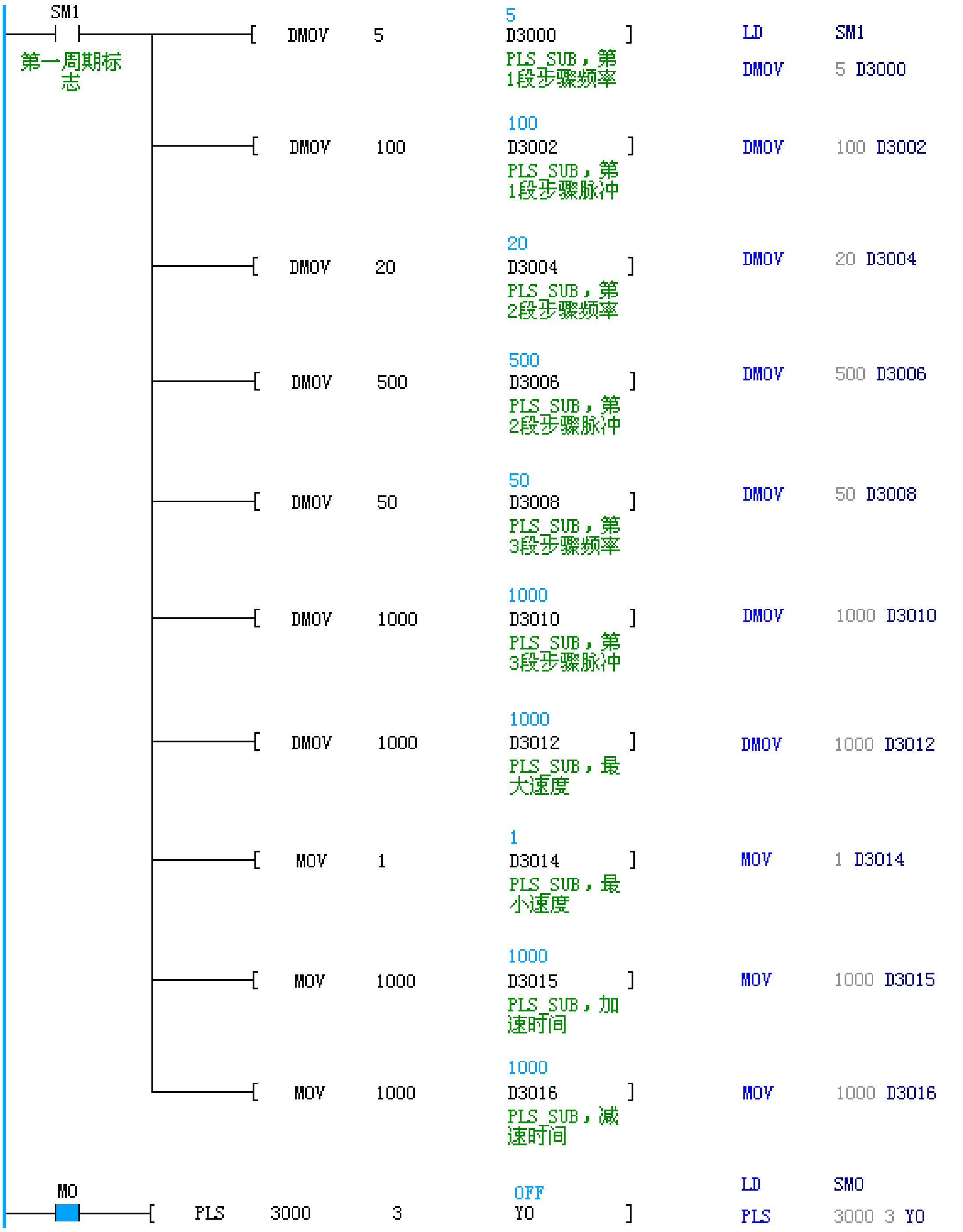
S1：存放参数的元件起始地址 **S2**：输出段数 0 ～ 255

D：高速脉冲输出点，MX1H 指定 Y0、Y1；MX2H 指定 Y0、Y1、Y2；MX2V 指定 Y0 ～ Y7。

1. 输出端口按照以 S1 起始的 n 个单元中设定的参数分 S2 段输出相应的脉冲。
2. 当输出段数为 0 时，没有输出。
3. 使用 SM80 ～ SM81、SM261 ～ SM266（脉冲输出停止使能标志）可以关断高速脉冲输出，其它的标志位与高速共用。
4. 使用 SD56 ～ SD58 可以查看 PLS 目前执行到的段数。
5. 用后台的 PTO 指令向导生成 PLS 指令，在生成指令的同时可选择是否将注释加入到全局变量表中，以子程序的形式进行调用。指令一执行系统就按照设置输出相应脉冲，可以控制发送或停止脉冲，设置频率、脉冲量。利用后台的指令向导生成的子程序 PLS_SUB 的内容（设 n 为其 D 元件号，M 为总段数）：

LD SM0
DMOV 第 1 段步骤频率 Dn
DMOV 第 1 段步骤脉冲数 Dn+2
DMOV 第 2 段步骤频率 Dn+4
DMOV 第 2 段步骤脉冲数 Dn+6
DMOV 第 3 段步骤频率 Dn+8
DMOV 第 3 段步骤脉冲数 Dn+10
………
DMOV 第 M 段步骤频率 Dn+4 M － 4
DMOV 第 M 段步骤脉冲数 Dn+4 M － 2
DMOV 最大速度 Dn+4 M
MOV 最小速度 Dn+4 M+2
MOV 加速时间 Dn+4 M+3
MOV 减速时间 Dn+4 M+4
LD SM0
PLS S1 S2 D

使用示例



1. 在此段程序中，指令 PLS 的参数设置从 D3000 开始设置，分别设置了第一段的脉冲频率和脉冲个数、第第二段的脉冲频率和脉冲个数、第三段的脉冲频率和脉冲个数、最大速度、最小速度、加速时间、减速时间。
2. PLS 指令中的 3000 表示参数设定是从 D3000 开始，3 表示程序中设定了 3 段步骤。
3. 当 M0 为 ON 时，执行 PLS 指令，Y0 端口按照设定的参数开始输出脉冲。当发送完总的脉冲数后，Y0 不再有输出。

注意事项

1. 建议使用 PTO 向导生成 PLS 指令，如果直接编写 PLS 指令，注意各步骤的脉冲个数不能过小。
2. 在设定的加速度下，每一个步骤的脉冲个数必须大于频率间变换所需要的最少脉冲数。
3. 用 p 表示某步骤输出的脉冲个数，F_N 表示第 N 段的频率，F_{max}、F_{min} 表示最高速度和最低速度，

T_{up}、T_{down} 表示加速时间和减速时间。

（1）当步骤 N 的速度大于步骤 N － 1 的速度时，步骤 N 的脉冲个数必须满足下面的条件：

$$P \geqslant \frac{(F_N+F_{N-1}) \times (F_N - F_{N-1}) \times Tup}{2000 \times (Fmax - Fmin)}$$

（2）当步骤 N 的速度小于步骤 N － 1 的速度时，步骤 N 的脉冲个数必须满足下面的条件：

$$P \geqslant \frac{(F_N+F_{N-1}) \times (F_N - F_{N-1}) \times Tdown}{2000 \times (Fmax - Fmin)}$$

4．（1）当 N=1 时，步骤 N － 1 的频率取 F_{min}，代入上述公式。

（2）当所有步骤数为 1 时，即只有 1 段时，脉冲个数必须满足下面的条件：

$$P \geqslant \frac{(F_1+Fmin) \times (F_1 - Fmin) \times (Tup + Tdown)}{2000 \times (Fmax - Fmin)}$$

（3）最后一个步骤的脉冲个数要满足下面的公式：

$$P \geqslant \frac{(F_N+F_{N-1}) \times (F_N - F_{N-1}) \times (Tup + Tdown)}{2000 \times (Fmax - Fmin)}$$

（4）各步骤中指定的频率不能大于先前设定的最高速度，也不能低于最低速度。

（5）各步骤的脉冲总数最大为 999,999。

5．PLC 请使用晶体管输出型，在高速脉冲输出时请按规定的负载电流接入输出晶体管。从输出端子得到的波形跟用户的输出负载有关系，在满足不能超过额定负载电流情况下，负载越大，输出波形越接近设定操作数。

6．在高速指令有效运行（包括输出完成）时，对同一端口的其它操作无效。只有在高速脉冲输出指令无效时，其它指令才能操作本端口。

7．PLSY、PLSR、PLS 定位指令可以利用端口输出高速脉冲。请注意不要同时对同一个高速端口使用这些指令进行高速脉冲输出。

6.7.13 PWM : PWM 脉冲输出指令

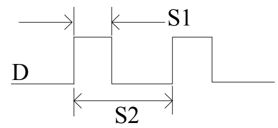
	PWM		S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：PWM (连续) 7步 32位：-----		----- -----	相关元件	SM80～SM84 SM261～SM272			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●	●
S2	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●	●
D	BOOL		*																						

＊ 该指令使用 C 元件时，只能使用 C0 ～ C199。
使用位元件 Y 时，对于 MX1H 指定 Y0、Y1；MX2H 指定 Y0、Y1、Y2；MX2V 指定 Y0 ～ Y7。

指令说明

├──┤ ├──[PWM S1 S2 D]



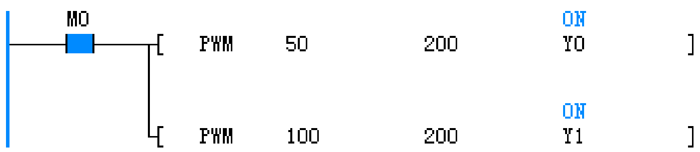
S1：脉冲幅宽（t）指定脉冲宽度（ms/μs）。在指令运行过程中更改 S1 的内容，输出脉冲也随之发生变化。SM84（PWM 微秒输出使能标志）为 OFF 时，S1 以 ms 为单位，可设定范围为 0 ～ 32767（ms）；SM84 为 ON 时，S1 以 μs 为单位，可设定范围为 10 ～ 32757（μs）。当设定数值不在范围之内时，PLC 信息报“指令操作数数值非法”错误，同时脉冲不输出。

S2：周期（T）指定脉冲周期（ms/μs）。在指令运行过程中更改 S2 的内容，输出脉冲也随之发生变化。SM84 为 OFF 时，S2 以 ms 为单位，可设定范围为 0 ～ 32767（ms）；SM84 为 ON 时，S2 以 μs 为单位，可设定范围为 20 ～ 32767（μs）。当设定数值不在范围之内时，PLC 信息报“指令操作数数值非法”错误，同时脉冲不输出。

D：高速脉冲输出点。MX1H 指定 Y0、Y1；MX2H 指定 Y0、Y1、Y2；MX2V 指定 Y0 ～ Y7。

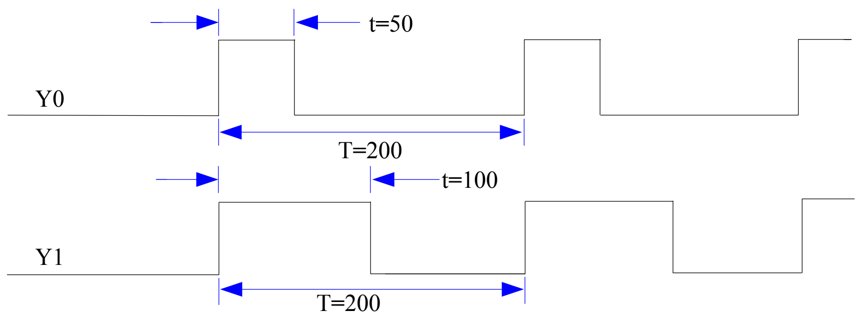
在 D 所指定的端口输出宽度为 S1、周期为 S2 的 PWM 脉冲。

使用示例




```
LD   M0
PWM  50 200 Y0
PWM  100 200 Y1
```

示例中 Y0、Y1 的脉冲波形时序图：



其中 t 为脉冲宽度，T 为脉冲周期。

1. SM84 为 OFF 且 M0 为 ON 时，执行 PWM 指令，Y0 端口输出宽度为 50ms、周期为 200ms 的 PWM 脉冲，Y1 端口输出宽度为 100ms、周期为 200ms 的 PWM 脉冲。输出状态不受扫描周期的影响。SM84 为 ON 且 M0 为 ON 时，执行 PWM 指令，Y0 端口输出宽度为 50μs 周期为 200μs 的 PWM 脉冲，Y1 端

口输出宽度为 100μs 周期为 200μs 的 PWM 脉冲，输出状态不受扫描周期的影响。

2. SM80 ～ SM81，SM261 ～ SM266 对应 Y0 ～ Y1，Y2 ～ Y7 的输出停止使能位。使能位为 ON 时，输出中止。

3. SM82 ～ SM83，SM267 ～ SM272 对应 Y0 ～ Y1，Y2 ～ Y7 的脉冲输出监控位，当 M0 为 OFF 时，监控位清为 OFF。

注意事项

1. 当 S1 为 0 时，高速输出端口输出一直为 OFF。当 SM84=0 时，S2 ≥ S1，若 S1=S2，则高速输出输端口 出一直为 ON；当 SM84=1 时，S2 ≥ S1+10。否则 PLC 信息报“指令操作数数值非法”错误，同时脉冲不输出，也不占用系统资源。
2. S1、S2 可在 PWM 指令执行时更改。
3. 从输出端子得到的波形跟输出负载有关系，在满足最大输出电流情况下，负载越大，输出波形越接近设定操作数。因此，为了输出高速脉冲，PLC 的输出晶体管上的负载电流要大，但不能超过额定负载电流。
4. 在高速指令有效运行（包括输出完成）时，对同一端口的其它操作无效。只有在高速指令无效时，其它指令才能操作本端口。
5. 使用 2 个 PWM 指令能够在输出端得到各自独立的高速脉冲输出，也可和 PLSY 或 PLSR 在不同的输出点得到各自独立的高速脉冲输出。
6. 有多条 PWM、PLSY 或 PLSR 指令操作同一端口时，先有效的指令控制端口输出状态，后有效的指令对输出点的状态无影响。
7. 与其它高速指令（DHSCS，DHSCR，DHSCI，DHSZ，DHSP，DHST，DHSTI，HCNT）相同，PWM 指令要满足系统中对高速输入和高速脉冲输出的要求。

6.7.14 MTR : 矩阵输入指令

	MTR		S1 D1 D2 S2	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√	√
步长	16位： MTR (连续) 9步 32位： -----		----- -----	相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	BOOL	●																							
D1	BOOL		●																						
D2	BOOL		●	●	●																				
S2	INT																							●	

指令说明

MTR S1 D1 D2 S2

S1：矩阵的行信号输入的起始软元件（X）。 编号 X0，X10，X20…等最下位为 0 的 X。

D1：矩阵的列信号输入的起始软元件（Y）。 编号 Y0，Y10，Y20…等最下位为 0 的 Y。

D2：ON 输出目标地址的起始软元件（Y，M，S）。 编号 Y、M、S…最下位为 0。

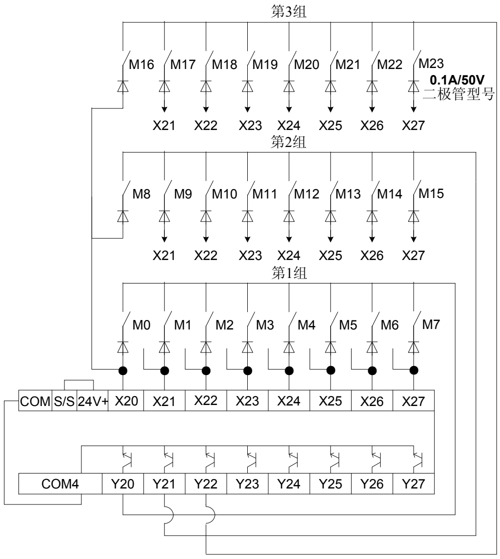
S2：16 位整数。设定矩阵输入的列数（范围：2 ～ 8）。

1. 对 8 点的 S1 输入和 S2 点的晶体管输出进行时间分割控制，以便依次读取 8 点 S2 列的输入信号，然后输出到 D2 中。
2. PLC 通过输入端子来接收外界输入信号。由于输入端子数量有限，若采用一个端子接收一路信号的普通输入方式，很难实现大量多路信号的输入。给 PLC 加设矩阵输入电路可以有效解决这个问题。

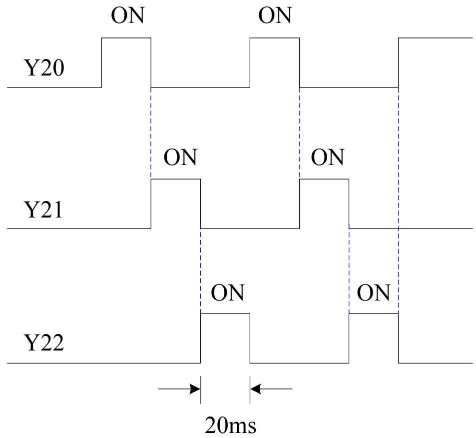
下面图一所示是一种 PLC 矩阵输入电路，它采用 X20 ～ X27 端子接收外界输入信号，这些端子外接 3 组由二极管和开关组成的矩阵输入电路，这三组矩阵电路一端都接到 X20 ～ X27 端子，另一端则分别接 PLC 的 Y20、Y21、Y22 端子。在工作时，Y20、Y21、Y22 端子内部硬触点轮流接通，如图二所示。当 Y20 接通时，Y21、Y22 断开；当 Y21 接通时，Y20、Y22 断开；当 Y22 接通时，Y20、Y21 断开，然后重复这个过程。一个周期内每个端子接通时间为 20ms。

在 Y20 端子接通期间，若第 1 组输入电路中的某个开关闭合，如 M0 开关闭合，X20 端子输出的电流经二极管、开关流入 Y20 端子，并经 Y20 端子内部闭合的硬触点流到 COM 端，X20 端子有电流输出，相当于该端子有输入信号，该输入信号在 PLC 内部被转存到辅助继电器 M0 中。在 Y20 端子接通期间，如闭合第 2 组或第 3 组中的某个开关，由于此时 Y21、Y22 端子均断开，故操作这两组开关均无效。在 Y21 端子接通期间，X20 ～ X27 端子接收第 2 组开关输入。在 Y22 端子接通期间，X20 ～ X27 端子接收第 3 组开关输入。

在采用图一所示形式的矩阵输入电路时，如果将输出端子 Y20 ～ Y27 和输入端子 X20 ～ X27 全部利用起来，则可以实现 8×8=64 个开关信号的输入。由于 Y20 ～ Y27 每个端子的接通时间是 20ms，故矩阵电路的扫描周期为 8×20=160ms。对于扫描周期长的矩阵输入电路，若输入信号时间小于扫描周期，可能会出现输入无效的情况。例如在图一中，若在 Y20 端子刚开始接通时闭合开关 M20（M20 开关控制 Y22 端子），按下时间为 30ms 再松开，由于此时 Y22 端子还未开始接通（从 Y20 到 Y22 导通时间间隔为 40ms），故操作开关 M20 无效，因此矩阵输入电路不适合用于要求快速输入的场合。



（图一）PLC 矩阵输入电路



（图二）Y20、Y21、Y22 内部硬触点轮流接通

使用示例

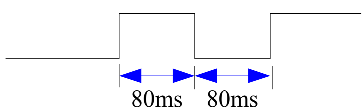

```
LD   M0
MTR  X20 Y20 M20 3
```

当 M0 为 ON 时，执行 MTR 指令，将 X20 为起始编号的 8 个连号元件作为矩阵输入行信号，将 Y20 为起始编号的 3 个连号元件作为矩阵输出列信号，将矩阵输入信号保存在以 M20 为起始编号的 3 列 8 个连号元件（M20 ～ M27、M28 ～ M35、M36 ～ M43）中。

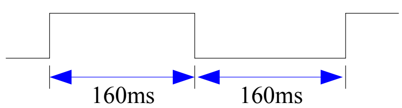
注意事项

1. 操作时请注意以 S1 指定的输入作为起始，占用 8 点输入；以 D1 指定的输出作为起始，占用 S2 点输出。
2. D2 指定的输出不要与 D1 指定的输出编号（占用 S2 点）重复。
3. 在一个扫描周期内请只使用一次 MTR 指令。
4. MTR 指令的输出，请使用晶体管输出形式。MTR 指令的输入，通常使用 X20 以后的输入。若使用输入 X0 ～ X17（16 点基本单元为 X0 ～ X17）时，获得的速度加快，但输出晶体管还原时间长并且输入灵敏度高，会发生误输入。
5. 若使 8 列 64 点的输入以 80ms 或 160ms 读取周期动作，各信号的 ON/OFF 幅度必须大于下图所示数值。

使用X0～X17输入时



使用X20以后的输入时



6.7.15 ENCODER: 编码器初始化

ENCODER	S1 S2 S3 S4 D	适应机型	MX1H	MX2H	MX2V	MX3H
				√	√	√
步长	12步	相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	DINT																		*						
S2	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●	●
S3	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●	●
S4	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●	●
D	BOOL		●	●	●																				

＊ S1 操作数使用 C 元件时，只能使用 C235 ～ C255。S2 ～ S4 操作数使用 C 元件时，只能使用 C0 ～ C199。

指令说明

├──┤ ├──[ENCODER S1 S2 S3 S4 D]

S1：高速计数器，适用于 C235 ～ C255

S2：编码器线数

S3：Z 轴偏移

S4：编码器计数器误差范围

D：误差标志位

对 S1 指定计数器进行编码器模式初始化；计数器数值只在 0 ～ S2-1 范围波动；

加计数：过 S2-1 时，计数器直接跳变为 0；

减计数：过 0 时，计数器直接跳变为 S2-1；

有 Z 相的计数器，Z 信号到达：

1．先检测计数器偏差；S1 － S3 误差范围大于 S4，则 D 置位；

2．用 S3 值更新 S1 指定计数器；

注意事项

ENCODER 指令使用前，必须使用 HCNT 指令驱动高速计数器计数。

6.8 定位指令

6.8.1 DABS : 当前值读取指令

	DABS		S D1 D2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	8步			相关元件	*			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	BOOL	●	●	●	●																				
D1	BOOL		●	●	●																				
D2	DINT											●	●	●			●	●	*						

* 该指令使用 C 元件时，只能使用 C200 ～ C255。

* 相关元件：SM29、SD200 ～ SD201、SD220 ～ SD221、SD280 ～ SD281、SD300 ～ SD301、SD320 ～ SD321、SD340 ～ SD341、SD360 ～ SD361、SD380 ～ SD381

指令说明

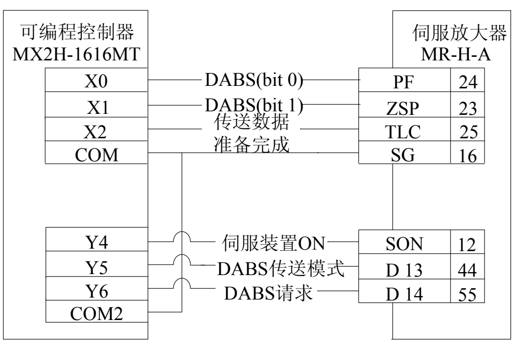
├──┤ ├──[DABS S D1 D2]

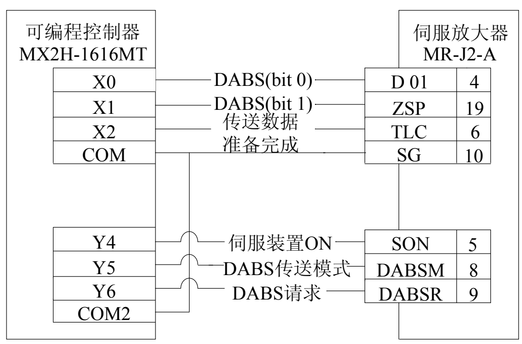
操作数设置	操作数功能描述
S	来自伺服装置的输入点，输入点占有 S、S+1、S+2 共 3 个连续输入 X 元件或其它位元件。
D1	传送至伺服装置的输出点，输出点占有 D1、D1+1、D1+2 共 3 个连续输出 Y 元件或其它位元件。
D2	从伺服装置读取的当前值数据，当前值数据占有（D2+1，D2）共 2 个字元件。读取的 DABS 数值必须写入当前值数据寄存器（SD201，SD200）/（SD221，SD220）/（SD281，SD280）/（SD301，SD300）/（SD321，SD320）/（SD341，SD340）/（SD361，SD360）/（SD381，SD380）中。

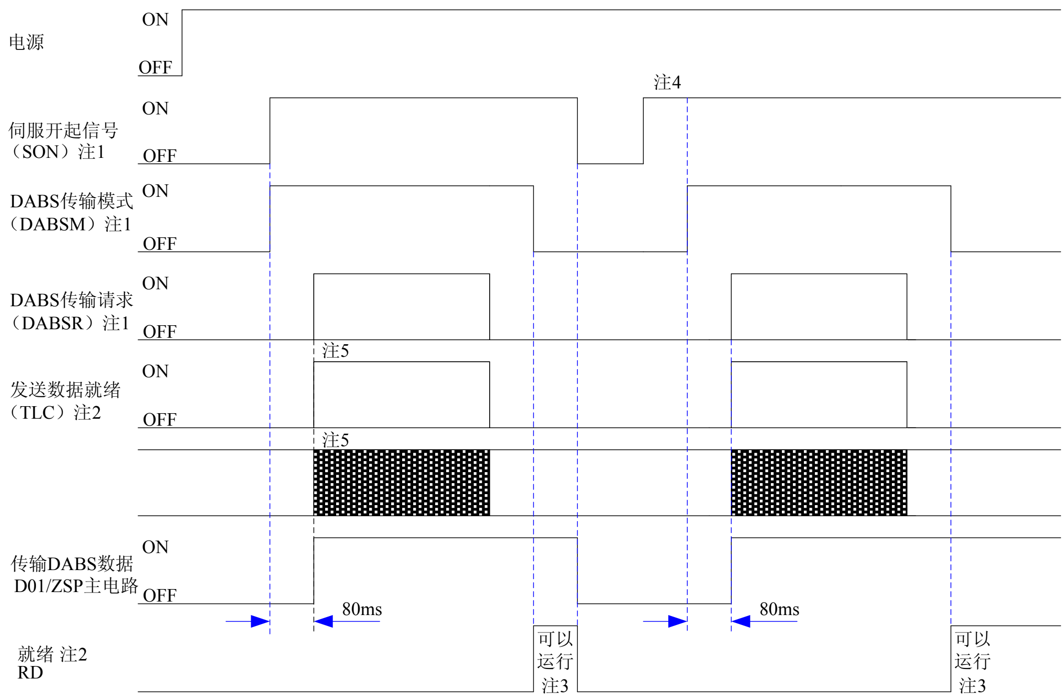
1．DABS 指令执行时，在伺服 ON 输出后，驱动 DABS 传送模式信号。

2．DABS 指令应该在 DABS 数值读出后仍然执行。当 DABS 读取操作完成后，若将指令前的驱动信号断开则伺服 ON 信号变为 OFF，不能执行运行。

DABS 指令输入输出信号接线图举例如下所示：







注 1：PLC 发给伺服放大器的信号。

注 2：伺服放大器发出的信号。

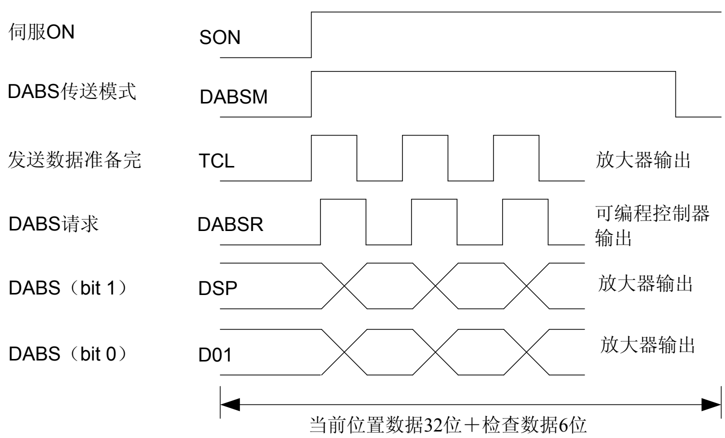
注 3：系统数据传输完毕，可以进入正常操作，RD 置位后，DABSM 信号将不再被接受。

注 4：此处 SON 信号比 DABSM 信号先置位，即使这样，主电路也必须等到 DABSM 置 ON 后才会开启。如果 DABS 传输过程中将 DABSM 置 OFF，传输模式中断，伺服放大器会出现超时警告（ALE 5）。

注 5：这几个信号针脚在 DABSM 置位 / 复位时定义会改变。

关于伺服放大器的相关详细介绍请参见三菱电机株式会社的相关产品资料。

绝对位置的检出动作：



1．可编程控制器 DABS 指令驱动后，驱动伺服 ON 输出和 DABS 传送模式输出。

2．在传送数据准备完成信号和 DABS 请求信号相互确认的同时，进行 32+6（32 位数据 +6 位校验码）位的数据的通讯。

3．数据通过 DABS bit0，bit1 的 2 位回路执行传送。

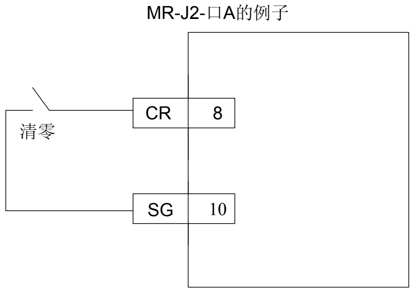
4．DABS 数据读出结束时，执行结束标志位 SM29 动作。

关于初次原点检出：

即使是带绝对位置检出功能的伺服电机，在设备制作时，也至少需要进行 1 次原点检出，对伺服电机给出清零信号。

请在下面任意一种方法中进行初次原点检出。

1．执行带 DOG 搜索的原点回归 DSZR 指令、原点回归 DZRN 指令时，附带伺服放大器输入清零信号功能，结束原点回归。
2．通过手动的方式调整位置，执行设备的原点检出后，向伺服放大器输入清零信号。清零信号的输入请使用可编程控制器的输出，或者如下图所示通过外部开关执行。



相关软元件：

地址	名称	功能
SM29	指令执行结束标志	定位指令执行结束时，置 ON。
SD200/SD220/SD280 /SD300/SD320/SD340 /SD360/SD380	Y0 ～ Y7 定位输出当前值（低位）	用于绝对位置的计算和保存。每次执行定位指令，都根据（SD53，SD52）和方向信号来计算和更新（SD221，SD220）。当开机并且从伺服驱动器读取绝对位置数据时，请将得到的绝对位置数据（32 位长整型）放入（SD221，SD220）中。
SD201/SD221/SD281 /SD301/SD321/SD341 /SD361/SD381	Y0 ～ Y7 定位输出当前值（高位）	

注意事项

1. DABS 指令支持三菱品牌 MR ～ J2 及 MR ～ J2S 等伺服放大器，利用其专用数据传输协议来读取绝对位置当前值数据。
2. DABS 指令为 32 位专用指令。而对于其它品牌的伺服放大器，需要采用通讯或指定的其它方式来获取绝对位置当前值数据。
3. 由于执行 DABS 指令时会对与该指令相关的输入输出点进行处理和操作，因此在应用其它伺服放大器的场合不能使用 DABS 指令。
4. PLC 和伺服放大器应该同时上电或先给伺服放大器上电，保证 PLC 进入运行状态在伺服放大器上电之后。
5. DABS 数值读出 D2，可设定为指定的其它字元件范围，但必须将该值传送至寄存器（SD201，SD200），（SD221，SD220）/（SD281，SD280）/（SD301，SD300）/（SD321，SD320）/（SD341，SD340）/（SD361，SD360）/（SD381，SD380）中。
6. SM82，SM83，SM267 ～ SM272 对应 Y0 ～ Y7 的输出标志，当输出完成后，标志清除。
7. 若数据传送超时，报 79 号错误；若校验错误，报 80 号错误。

6.8.2 DZRN : 原点回归指令

	DZRN		S1	S2	S3	D	适用机型	MX1H	MX2H	MX2V	MX3H
								√	√	√	√
步长	16位：-----				-----		相关元件	*			
	32位：DZRN (连续) 10步				-----						

操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z				Un.b
S1	DINT										●	●	●	●	●	●	●	●	*		●		●	●		
S2	INT										●	●	●	●	●	●	●	●	*		●		●	●		
S3	BOOL	●		●	●																					
D	BOOL		●																							

* 该指令使用 C 元件时，操作数 S1 只能使用 C200 ～ C255，操作数 S2 只能使用 C0 ～ C199。

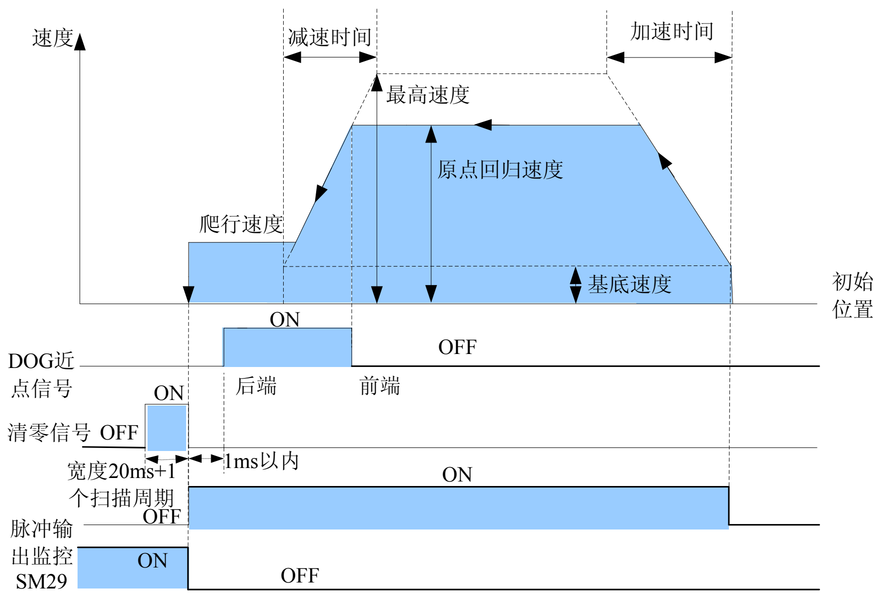
* 相关元件：SM29、SM70 ～ SM77、SM80 ～ SM83、SM261 ～ SM272、SM280 ～ SM285、SM288、SM290 ～ SM295、SM298、SM310 ～ SM315、SM318、SM320 ～ SM325、SM328、SM330 ～ SM335、SM338、SM340 ～ SM345、SM348、SM350 ～ SM355、SM358、SM360 ～ SM365、SM368、SD50 ～ SD53、SD70 ～ SD81、SD200 ～ SD210、SD220 ～ SD230、SD280 ～ SD290、SD300 ～ SD310、SD320 ～ SD330、SD340 ～ SD350、SD360 ～ SD370、SD380 ～ SD390

指令说明

├──┤ ├──[DZRN S1 S2 S3 D]

操作数设置	操作数功能描述
S1	原点回归速度，指定原点回归开始时的速度。范围是 10 ～ 100000（Hz）。
S2	爬行速度，指定近点信号（DOG）变为 ON 后的相对较低的速度，范围是 10 ～ 32767（Hz）。
S3	近点信号，指定近点信号输入 X 元件。当指定输入继电器（X）以外的元件时，由于会受到可编程控制器扫描周期的影响，会引起原点位置的偏移增大。
D	高速脉冲输出起始地址，MX1H 指定 Y0 ～ Y1，MX2H 指定 Y0 ～ Y2，MX2V 指定 Y0 ～ Y7。可编程控制器的输出必须采用晶体管输出方式。

原点回归动作：



1．调用 DZRN 指令时执行原点回归动作，以 S1 指定的原点回归速度移动，当 S3 指定的近点信号 DOG 为 ON，就开始减速，减速到 S2 指令的爬行速度后以爬行速度运行，S3 指定的近点信号 DOG 从 ON 到 OFF 后，就立即停止脉冲输出，指令执行标志位 SM29 为 ON，结束原点回归动作。

2．使用该指令时，请务必进行最高速度、基底速度、加速时间及减速时间特殊寄存器值的设定，否则指令不会正常运行。

3．加速时间是指由基底速度加速到最高速度的时间，减速时间是指由最高速度减速到基底速度的时间。

4．在原点回归过程中，当前值寄存器数值将向减少方向动作。

5．清零信号输出功能有效时，在脉冲输出停止后 1ms 以内，清零信号在 20ms+1 个扫描周期的时间内保持为 ON。

6．SM280/SM290/SM310/SM320/SM330/SM340/SM350/SM360 分别为 Y0 ～ Y7 清零功能有效。MX1H 清零信号为 Y0 ～ Y1，对应清零信号的输出点分别为 Y10 ～ Y11；MX2H 清零信号为 Y0 ～ Y2，对应清零信号的输出点分别为 Y10 ～ Y12；MX2V 清零信号为 Y0 ～ Y7，对应清零信号的输出点分别为 Y20 ～ Y27。若置位 SM280/SM290/SM310/SM320/SM330/SM340/SM350/SM360 和 SM281/SM291/SM311/SM321/SM331/SM341/SM351/SM361，则清零信号根据 SD210/SD230/SD290/SD310/SD330/SD350/SD370/SD390 指定的 Y 元件（SD210/SD230/SD290/SD310/SD330/SD350/SD370/SD390 里存储清零信号 Y 元件的序号）；若只置位 SM280/SM290，则清零信号为 Y10/Y11。

7．如果在原点回归指令运行过程中，断开指令前的驱动触点，则会减速停止。这样可以减小机械振动。

相关软元件：

地址	名称	功能
SM29	指令执行结束标志	定位指令执行结束时，置 ON
SM70 ～ SM77	Y0 ～ Y7 输出完成中断使能标志	置 ON，使能 PT0（Y0 ～ Y7）输出完成中断
SM80 ～ SM81、SM261 ～ SM266	Y0 ～ Y7 脉冲输出停止使能标志	置 ON，Y0 高速脉冲输出停止
SM82 ～ SM83 、SM267 ～ SM272	Y0 ～ Y7 脉冲输出监控位	Y0 ～ Y7 高速脉冲输出监视
SM280/SM290/SM310/SM320/SM330/SM340/SM350/SM360	Y0 ～ Y7 清零功能有效	适用 DZRN /DSZR, 定位指令原点回归（DZRN）清零信号输出功能有效（Y0）
SM281/SM291/SM311/SM321/SM331/SM341/SM351/SM361	Y0 ～ Y7 清零信号指定元件有效	适用 DZRN/DSZR，对应的 SD210 内的数值为 Y0 表示清零信号，不指定则 Y10 为 Y0 的清零信号
SM282/SM292/SM312/SM322/SM332/SM342/SM352/SM362	Y0 ～ Y7 原点回归方向标志	适用于 DZRN/DSZR，作用于 Y0 对应的轴：置 ON 时，表示原点回归方向为正转方向；置 OFF 时，表示原点回归方向为反转方向
SM283/SM293/SM313/SM323/SM333/SM343/SM353/SM363	Y0 ～ Y7 正转极限标志	适用于 DZRN/DSZR/DDRVI/DDRVA，作用于 Y0 对应的轴：置 ON 时，表示到达正转方向的极限；置 OFF 时，表示未到极限
SM284/SM294/SM314/SM324/SM334/SM344/SM354/SM364	Y0 ～ Y7 反转极限标志	适用于 DZRN/DSZR/DDRVI/DDRVA，作用于 Y0 对应的轴：置 ON 时，表示到达反转方向的极限；置 OFF 时，表示未到极限
SM285/SM295/SM315/SM325/SM335/SM345/SM355/SM365	Y0 ～ Y7 近点信号逻辑反转使能标志	适用于 DZRN/DSZR，作用于 Y0 对应的轴：置 ON 时，按负逻辑处理（输入 OFF 时，近点信号为 ON）；置 OFF 时，按正逻辑处理（输入 ON 时，近点信号为 ON）
SM288/SM298/SM318/SM328/SM338/SM348/SM358/SM368	Y0 ～ Y7 定位指令运行标志	用于监测执行 DZRN/DDRVI/DDRVA/DSZR/DDVIT 时高速输出通道 Y0 的状态，正在运行时为 ON，空闲时为 OFF

地址	名称	功能
SD50/SD52/SD70/SD72 /SD74/SD76/SD78/SD80	Y0 ～ Y7 发送脉冲总数（低位）	Y0 ～ Y7 发送脉冲总数，与脉冲方向无关
SD51/DSD53/SD71/SD73 /SD75/SD77/SD79/SD81	Y0 ～ Y7 发送脉冲总数（高位）	
SD200/SD220/SD280/SD300 /SD320/SD340/SD360/SD380	Y0 ～ Y7 定位输出当前值（低位）	用于绝对位置的计算和保存。每次执行定位指令，都根据（SD53，SD52）和方向信号来计算和更新（SD221，SD220）。当开机并且从伺服驱动器读取绝对位置数据时，请将得到的绝对位置数据（32 位长整型）放入（SD221，SD220）中
SD201/SD221/SD281/SD301 /SD321/SD341/SD361/SD381	Y0 ～ Y7 定位输出当前值（高位）	
SD202/SD222/SD282/SD302 /SD322/SD342/SD362/SD382	Y0 ～ Y7 定位输出最高速度（低位）	执行定位指令时的最高速度，范围：10 ～ 100000，单位：脉冲
SD203/SD223/SD283/SD303 /SD323/SD343/SD363/SD383	Y0 ～ Y7 定位输出最高速度（高位）	
SD204/SD224/SD284/SD304 /SD324/SD344/SD364/SD384	Y0 ～ Y7 定位输出基底速度	执行定位指令时的基底速度（最高速度的 1/10 以下）
SD205/SD225/SD285/SD305 /SD325/SD345/SD365/SD385	Y0 ～ Y7 定位输出加速时间	执行定位指令时的加速时间，范围：2 ～ 5000，单位：ms

注意事项

1. 由于 DZRN 指令没有自动搜索近点信号功能，必须从比近点检测装置前端更远处开始进行原点复归操作。需要自动搜索近点信号功能时，请使用 DSZR 指令。
2. 设计近点信号 DOG 时，请考虑有足够为 ON 的时间能充分允许减速到爬行速度。没有能够减速到爬行速度时，会导致停止位置偏移。
3. 建议使用的爬行速度足够慢。原点回归速度的停止是不进行减速停止的所以如果爬行速度过快，会由于惯性导致停止位置偏移。
4. 因为没有零点信号，所以需要对原点位置做微调，建议调整近点信号 DOG 的位置。
5. 脉冲输出监控位为 ON 时，使用该输出端口的定位指令不能执行。需要注意的是，即使指令驱动触点变为 OFF，在脉冲输出监控位为 ON 期间，也不要执行指定了相同输出端口的定位指令。
6. 以下异常结束指令时，指令结束标志位 SM29 不会置 ON：正转极限标志位或者反转极限标志位为 ON 而减速停止，以及断开指令触点而减速停止。
7. 如果将指令中的近点信号指定为 X0~X7，那么通过中断处理停止。
   以下条件下，停止处理受到扫描周期影响：
   指定了 X10 以后的输入编号或者辅助继电器等其它的软元件，另外，指定了 X10 以外的输入继电器为近点信号时，受输入滤波的影响。
8. 近点信号指定了 X0~X7 时，不能和以下的用重复使用：
   高速计数器、输入中断、脉冲捕捉、SPD 指令、DSZR 指令、DDVIT 指令。

6.8.3 PLSV：可变速脉冲输出指令

	PLSV		S D1 D2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位： ----- ----- 32位：PLSV（连续）8步 -----			相关元件	*			

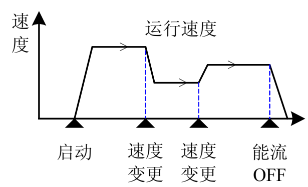
操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z				Un.b
S	DINT										●	●	●	●	●	●	●	●	*		●		●	●	
D1	BOOL		*																						
D2	BOOL		*	●	●																				

* 该指令使用 C 元件时，只能使用 C200 ～ C255。

* 对于 D2 操作数 MX2V 只支持 Y 元件。

* 对于 D1、D2 作为脉冲输出端口时，MX1H 系列只能使用 Y0 ～ Y1；MX2H 系列只能使用 Y0 ～ Y2；MX2V 系列只能使用 Y0 ～ Y7。

* 相关元件：详见指令说明。



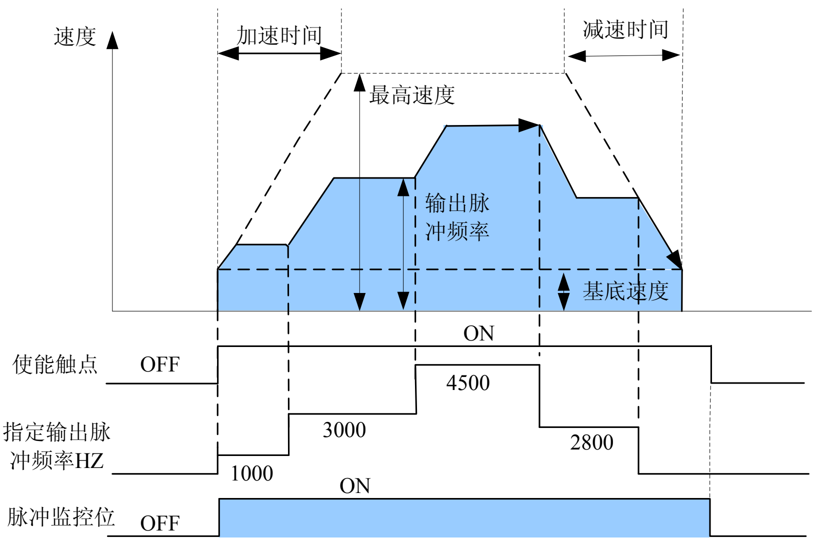
指令说明

├──┤ ├──[PLSV S D1 D2]

操作数设置	操作数功能描述
S	输出脉冲频率（Hz），范围是 10 ～ 100000（Hz）和 - 10 ～ - 100000（Hz）。
D1	脉冲输出方式。 MX1H/2H：脉冲 + 方向，方向端口不限定； MX2V：三种脉冲输出方式，脉冲 + 方向，正反向脉冲，AB 相输出，输出端口由系统组态设定，若指令输出端口与系统组态设定不同，系统会提示错误。
D2	

带加减速动作（SM85 为 ON）：

在加减速动作 SM85 为 ON 时，如果可变速脉冲输出（PLSV）指令的输出脉冲频率 S 变化则加速或者减速动作后，变为输出频率。



1．调用 PLSV 指令时执行变速度运行，指令加速至指定的输出频率，即使在高速脉冲输出状态中，仍然

能够自由改变输出脉冲频率 S，有加减速动作。

2．指令驱动触点断开后，输出脉冲减速停止。

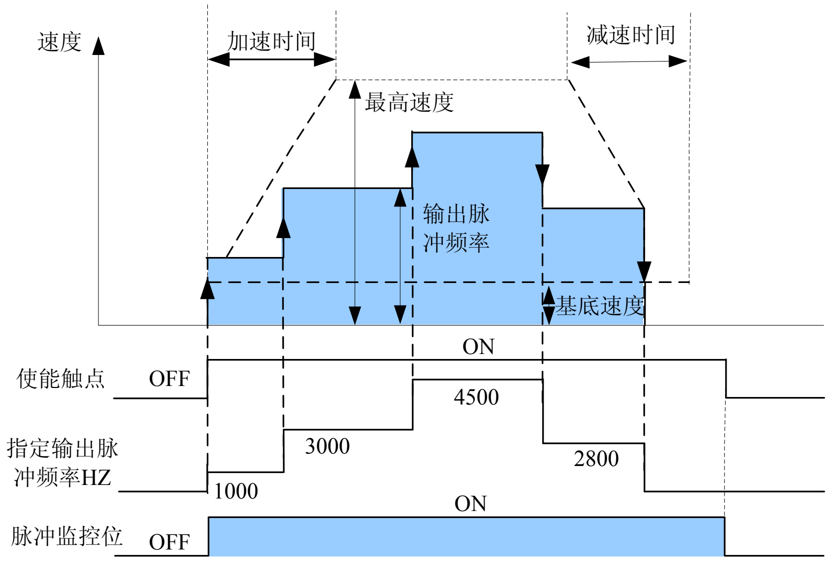
3．正 / 反方向的指定，根据输出脉冲频率 S 的正负符号决定。使用内置的晶体管输出作为脉冲输出端时，方向信号请使用晶体管输出。

4．使用该指令时，请务必进行最高速度、基底速度、加速时间及减速时间特殊寄存器值的设定，否则指令不会正常运行。

5．加速时间是指由基底速度加速到最高速度的时间，减速时间是指由最高速度减速到基底速度的时间。

无加减速动作（SM85 为 OFF）：

在加减速动作 SM85 为 OFF 时，如果可变速脉冲输出（PLSV）指令的输出脉冲频率 S 变化则无减速或者减速的变化，立即变为输出频率。



1．调用 PLSV 指令时执行变速度运行，指令立即输出指定的频率，即使在高速脉冲输出状态中，仍然能够自由改变输出脉冲频率 S，但是没有加减速动作。

2．指令驱动触点断开后，输出脉冲立即停止。

3．正 / 反方向的指定，根据输出脉冲频率 S 的正负符号决定。使用内置的晶体管输出作为脉冲输出端时，方向信号请使用晶体管输出。

4．使用该指令时，请务必进行最高速度、基底速度特殊寄存器值的设定，否则指令不会正常运行。

相关软元件：

地址	名称	功能
SM29	指令执行结束标志	定位指令执行结束时，置 ON
SM70 ～ SM77	Y0 ～ Y7 输出完成中断使能标志	置 ON，使能 PT0（Y0 ～ Y7）输出完成中断
SM80 ～ SM81、SM261 ～ SM266	Y0 ～ Y7 脉冲输出停止使能标志	置 ON，Y0 高速脉冲输出停止
SM82 ～ SM83 、SM267 ～ SM272	Y0 ～ Y7 脉冲输出监控位	Y0 ～ Y7 高速脉冲输出监视
SM85	PLSV 加减速使能标志	PLSV 指令切换有无加减速的使能标志
SM283/SM293/SM313 /SM323/SM333/SM343 /SM353/SM363	Y0 ～ Y7 正转极限标志	适用于 DZRN/DSZR/DDRVI/DDRVA，作用于 Y0 对应的轴：置 ON 时，表示到达正转方向的极限；置 OFF 时，表示未到极限
SM284/SM294/SM314 /SM324/SM334/SM344 /SM354/SM364	Y0 ～ Y7 反转极限标志	适用于 DZRN/DSZR/DDRVI/DDRVA，作用于 Y0 对应的轴：置 ON 时，表示到达反转方向的极限；置 OFF 时，表示未到极限

地址	名称	功能
SM288/SM298/SM318 /SM328/SM338/SM348 /SM358/SM368	Y0 ～ Y7 定位指令运行标志	用于监测执行 DZRN/DDRVI/DDRVA/DSZR /DDVIT 时高速输出通道 Y0 的状态，正在运行时为 ON，空闲时为 OFF
SD50/SD52/SD70/SD72 /SD74/SD76/SD78/SD80	Y0 ～ Y7 发送脉冲总数（低位）	Y0 ～ Y7 发送脉冲总数，与脉冲方向无关
SD51/DSD53/SD71/SD73 /SD75/SD77/SD79/SD81	Y0 ～ Y7 发送脉冲总数（高位）	
SD200/SD220/SD260/SD280 /SD300/SD320/SD340/SD360	Y0 ～ Y7 定位输出当前值（低位）	用于绝对位置的计算和保存。每次执行定位指令，都根据（SD53，SD52）和方向信号来计算和更新（SD221，SD220）。当开机并且从伺服驱动器读取绝对位置数据时，请将得到的绝对位置数据（32 位长整型）放入（SD221，SD220）中
SD201/SD221/SD261/SD281 /SD301/SD321/SD341/SD361	Y0 ～ Y7 定位输出当前值（高位）	
SD202/SD222/SD282/SD302 /SD322/SD342/SD362/SD382	Y0 ～ Y7 定位输出最高速度（低位）	执行定位指令时的最高速度，范围：10 ～ 100000，单位：脉冲
SD203/SD223/SD283/SD303 /SD323/SD343/SD363/SD383	Y0 ～ Y7 定位输出最高速度（高位）	
SD204/SD224/SD284/SD304 /SD324/D344/D364/ D384	Y0 ～ Y7 定位输出基底速度	执行定位指令时的基底速度（最高速度的 1/10 以下）
SD205/SD225/SD285/SD305 /SD325/SD345/SD365/SD385	Y0 ～ Y7 定位输出加速时间	执行定位指令时的加速时间，范围：2 ～ 5000，单位：ms
SD206/SD226/SD286/SD306 /SD326/SD346/SD366/SD386	Y0 ～ Y7 定位输出减速时间	执行定位指令时的减速时间，范围：2 ～ 5000，单位：ms

注意事项

1. 脉冲输出过程中，如果将输出频率变为 0，那么可编程控制器的脉冲，再带加减速时减速停止，在无加减速时立即停止。
2. 高速指令、包络线指令、定位指令可以利用 Y0 ～ Y7 八个端口输出高速脉冲。请注意不要同时对同一个高速端口使用这些指令进行高速脉冲输出。
3. 在脉冲输出监控位为 ON 时，即使指令前的触点已断开，也不能重新触发指令。
4. 在脉冲输出过程中，请不要改变输出脉冲频率的符号。如果想要变更的话，请先将输出脉冲频率设定为 0，在减速停止后，设定电机充分停止的时间，然后再变更输出脉冲频率的符号。
   如果在脉冲输出过程中改变了输出脉冲频率的符号，那么因为如下所示的动作，可能会损坏机械。
   （1）停止脉冲输出。
   （2）脉冲输出中标志位为 OFF。（脉冲输出停止，但是电机没有马上停止）
   （3）根据输出脉冲频率中指定的频率和旋转方向动作。
5. 在脉冲输出过程中，如果指令驱动触点 OFF，那么在带减速时减速停止，在无加减速时立即停止。且执行结束标志位 SM29 不动作。
6. 左 / 右极限标志位动作时，立即停止。
7. 指令执行结束后，旋转方向信号的输出 OFF。

6.8.4 DDRVI : 相对位置控制指令

	DDRVI		S1　S2　D1　D2	适应机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	11步			相关元件	*			

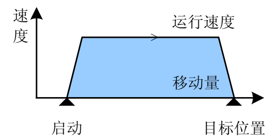
操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z				Un.b
S1	DINT										•	•	•	•	•	•	•	•	*		•		•	•	
S2	DINT										•	•	•	•	•	•	•	•	*		•		•	•	
D1	BOOL		*																						
D2	BOOL		*	•	•																				

＊　该指令使用 C 元件时，只能使用 C200 ～ C255。

＊　对于 D2 操作数 MX2V 只支持 Y 元件。

＊　对于 D1、D2 作为脉冲输出端口时，MX1H 系列只能使用 Y0 ～ Y1；MX2H 系列只能使用 Y0 ～ Y2；MX2V 系列只能使用 Y0 ～ Y7。

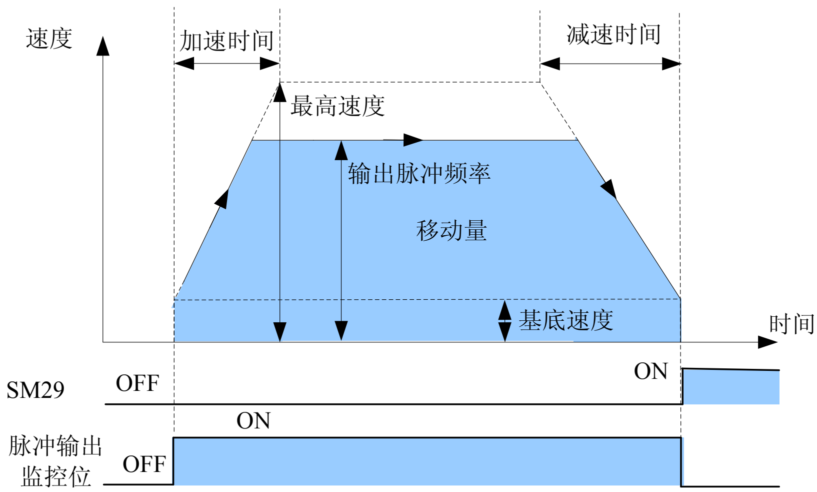
＊　相关元件：详见指令说明。



指令说明

├──┤ ├──[　DDRVI　S1　S2　D1　D2　]

操作数设置	操作数功能描述
S1	输出脉冲数（相对指定）。
S2	输出脉冲频率（Hz），范围是 10 ～ 100000。
D1	脉冲输出方式。 MX1H/2H：脉冲 + 方向，方向端口不限定； MX2V：三种脉冲输出方式，脉冲 + 方向，正反向脉冲，AB 相输出，输出端口由系统组态设定，若指令输出端口与系统组态设定不同，系统会提示错误。
D2	



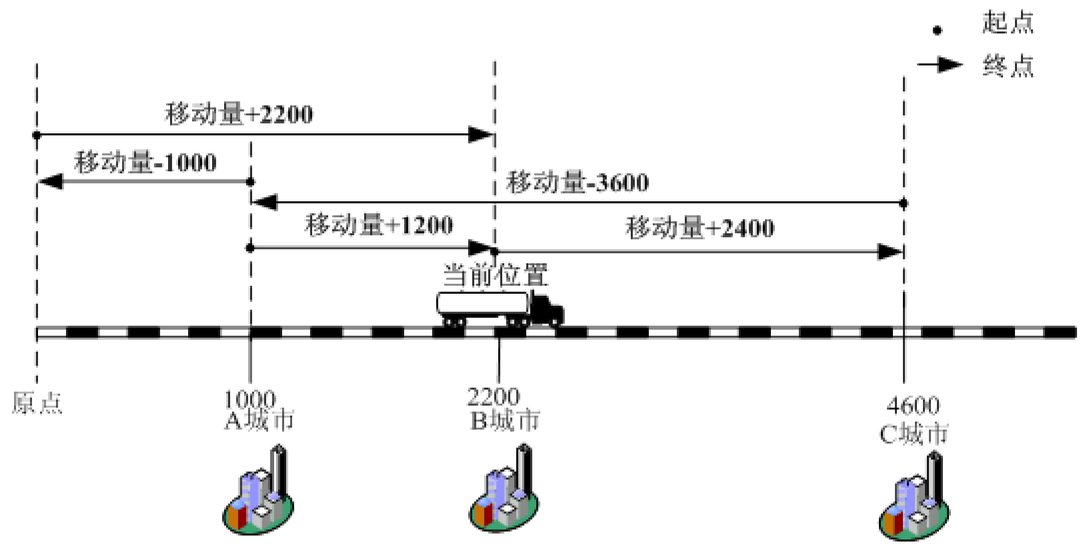
1. 调用 PLSV 指令时执行相对位置运行，指令由基底速度加速至运行速度 S2，发送完成 S1 个脉冲后减速停止。
2. 指定输出脉冲数的 S1，对应下面的当前值寄存器作为相对位置。
   Y0：（SD201，SD200）Y1：（SD221，SD220）Y2：（SD281，SD280）Y3：（SD301，SD300）

Y4：（SD321，SD320） Y5：（SD341，SD340） Y6：（SD361，SD360） Y7：（SD381，SD380 ）

3. 反转时，当前值寄存器的数值减小。
4. 旋转方向通过输出脉冲数 S1 的正负符号指定。使用内置的晶体管输出作为脉冲输出端时，方向信号请使用晶体管输出。
5. 指令执行过程中，即使改变操作数内容，也无法在当前运行中表现出来。只有在下一次指令执行时才有效。
6. 在指令执行过程中，指令前的驱动信号变为 OFF 时，将减速停止。此时执行结束标志 SM29 不动作。
7. 使用该指令时，请务必进行最高速度、基底速度、加速时间及减速时间特殊寄存器值的设定，否则指令不会正常运行。
8．加速时间是指由基底速度加速到最高速度的时间，减速时间是指由最高速度减速到基底速度的时间。

示例：

当前停止的位置作为起点，指定移动方向和移动量进行定位。终点由起点和移动量共同决定。

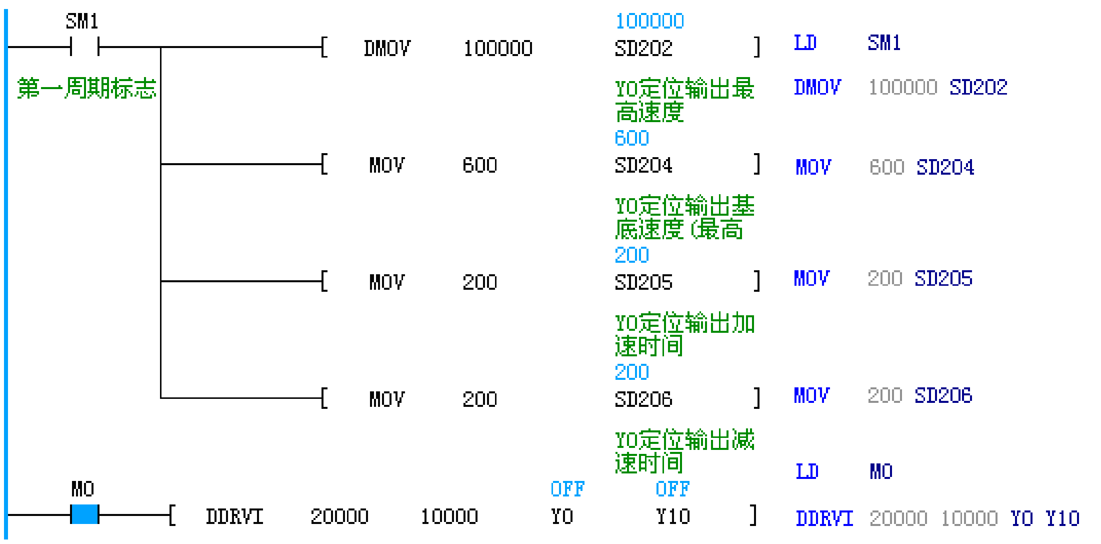


相关软元件：

地址	名称	功能
SM29	指令执行结束标志	定位指令执行结束时，置 ON
SM70 ～ SM77	Y0 ～ Y7 输出完成中断使能标志	置 ON，使能 PT0（Y0 ～ Y7）输出完成中断
SM80 ～ SM81、SM261 ～ SM266	Y0 ～ Y7 脉冲输出停止使能标志	置 ON，高速脉冲输出停止
SM82 ～ SM83、SM267 ～ SM272	Y0 ～ Y7 脉冲输出监控位	Y0 ～ Y7 高速脉冲输出监视
SM283/SM293/SM313 /SM323/SM333/SM343 / SM353/ SM363	Y0 ～ Y7 正转极限标志	适用于 DZRN/DSZR/DDRVI/DDRVA，作用于 Y0 对应的轴：置 ON 时，表示到达正转方向的极限；置 OFF 时，表示未到极限
SM284/SM294/SM314 /SM324/SM334/SM344 /SM354/ SM364	Y0 ～ Y7 反转极限标志	适用于 DZRN/DSZR/DDRVI/DDRVA，作用于 Y0 对应的轴：置 ON 时，表示到达反转方向的极限；置 OFF 时，表示未到极限
SM288/SM298/SM318 /SM328/SM338/SM348 /SM358/SM368	Y0 ～ Y7 定位指令运行标志	用于监测执 DZRN/DDRVI/DDRVA/DSZR /DDVIT 时高速输出通道 Y0 的状态，正在运行时为 ON，空闲时为 OFF

地址	名称	功能
SD50/SD52/SD70/SD72 /SD74/SD76/SD78/SD80	Y0～Y7 发送脉冲总数(低位)	Y0 ～ Y7 发送脉冲总数，与脉冲方向无关
SD51/DSD53/SD71/SD73 /SD75/SD77/SD79/SD81	Y0～Y7 发送脉冲总数(高位)	
SD200/SD220/SD260/SD280 /SD300/SD320/SD340/SD360	Y0 ～ Y7 定位输出当前值（低位）	用于绝对位置的计算和保存。每次执行定位指令，都根据（SD53，SD52）和方向信号来计算和更新（SD221，SD220）。当开机并且从伺服驱动器读取绝对位置数据时，请将得到的绝对位置数据（32 位长整型）放入（SD221，SD220）中
SD201/SD221/SD261/SD281 /SD301/SD321/SD341/SD361	Y0 ～ Y7 定位输出当前值（高位）	
SD202/SD222/SD282/SD302 /SD322/SD342/SD362/SD382	Y0 ～ Y7 定位输出最高速度（低位）	执行定位指令时的最高速度，范围：10 ～ 100000，单位：脉冲
SD203/SD223/SD283/SD303 /SD323/SD343/SD363/SD383	Y0 ～ Y7 定位输出最高速度（高位）	
SD204/SD224/SD284/SD304 /SD324/D344/D364/ D384	Y0 ～ Y7 定位输出基底速度	执行定位指令时的基底速度（最高速度的 1/10 以下）
SD205/SD225/SD285/SD305 /SD325/SD345/SD365/SD385	Y0 ～ Y7 定位输出加速时间	执行定位指令时的加速时间，范围：2 ～ 5000，单位：ms
SD206/SD226/SD286/SD306 /SD326/SD346/SD366/SD386	Y0 ～ Y7 定位输出减速时间	执行定位指令时的减速时间，范围：2 ～ 5000，单位：ms

使用示例



注意事项

1. 指令里输入的频率速度小于 SD204/SD224/SD284/SD304/SD324/D344/D364/ D384 中的值时，Y0 ～ Y7 无高速输出；当指令里输入的频率速度大于（SD203，SD202）/（SD223，SD222）/SD283，SD282）/（SD303，SD302）/（SD323，SD322）/（SD343，SD342）/（SD363，SD362）/（SD383，SD382）中的值时，会造成输出不正常。
2. 左 / 右极限标志位动作时，减速停止。
3. 脉冲输出中监控为 ON 时，使用该输出的定位指令不能执行。此外，即使指令驱动触点为 OFF，在脉冲输出中监控为 ON 期间，也请不要执行指定了同一输出编号的定位指令。

6.8.5 DDRVA : 绝对位置控制指令

				适应机型	MX1H	MX2H	MX2V	MX3H
	DDRVA		S1 S2 D1 D2		√	√	√	√
步长	11步			相关元件	*			

操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z				Un.b
S1	DINT										•	•	•	•	•	•	•	•	*		•		•	•	
S2	DINT										•	•	•	•	•	•	•	•	*		•		•	•	
D1	BOOL		*																						
D2	BOOL		*	•	•																				

* 该指令使用 C 元件时，只能使用 C200 ～ C255。

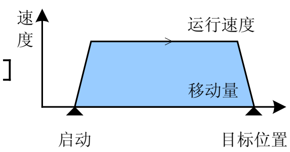
* 对于 D2 操作数 MX2V 只支持 Y 元件。

* 对于 D1、D2 作为脉冲输出端口时，MX1H 系列只能使用 Y0 ～ Y1；MX2H 系列只能使用 Y0 ～ Y2；MX2V 系列只能使用 Y0 ～ Y7。

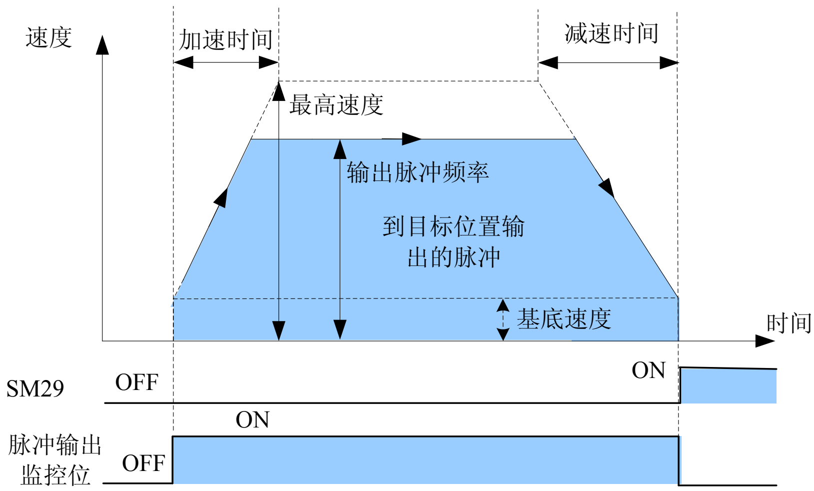
* 相关元件：详见指令说明。

指令说明

├──┤ ├──[DDRVA S1 S2 D1 D2]



操作数设置	操作数功能描述
S1	输出脉冲数（绝对指定）。
S2	输出脉冲频率（Hz），范围是 10 ～ 100000。
D1	脉冲输出方式。 MX1H/2H：脉冲 + 方向，方向端口不限定； MX2V：三种脉冲输出方式，脉冲 + 方向，正反向脉冲，AB 相输出，输出端口由系统组态设定，若指令输出端口与系统组态设定不同，系统会提示错误。
D2	



1. 调用 DDRVA 指令时执行绝对位置运行，指令由基底速度加速至运行速度 S2，运行到 S1 的位置后减速停止。
2. 指定输出脉冲数的 S1，对应下面的当前值寄存器作为绝对位置。
   Y0：（SD201，SD200） Y1：（SD221，SD220） Y2：（SD281，SD280） Y3：（SD301，SD300）

Y4：（SD321，SD320） Y5：（SD341，SD340） Y6：（SD361，SD360） Y7：（SD381，SD380 ）

3．反转时，当前值寄存器的数值减小。

4．旋转方向通过输出脉冲数 S1 的正负符号指定。当使用内置的晶体管输出作为脉冲输出端时，方向信号请使用晶体管输出。

5．指令执行过程中，即使改变操作数内容，也无法在当前运行中表现出来。只有在下一次指令执行时才有效。

6．在指令执行过程中，指令前的驱动信号断开时，将减速停止。此时执行完成标志 SM29 不动作。

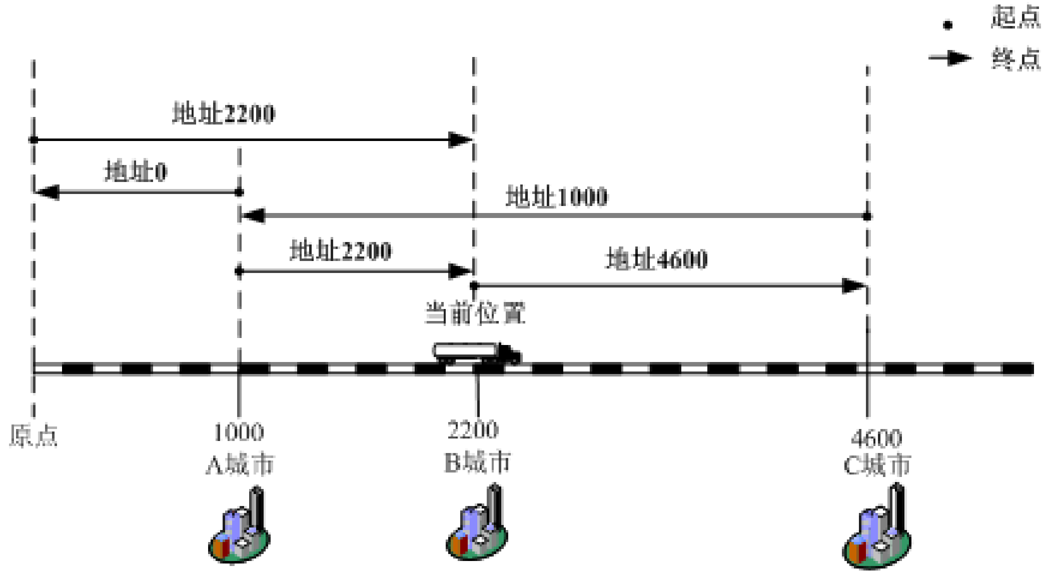
7．指令驱动信号断开后，在高速脉冲输出中标志 SM80、SM81 处于 ON 时，将不接受指令的再次驱动。

8．使用该指令时，请务必进行最高速度、基底速度、加速时间及减速时间特殊寄存器值的设定，否则指令不会正常运行。

9．加速时间是指由基底速度加速到最高速度的时间，减速时间是指由最高速度减速到基底速度的时间。

示例：

以原点为基准指定位置（绝对地址）进行定位。终点与起点没有关系，只与设定的绝对地址有关系

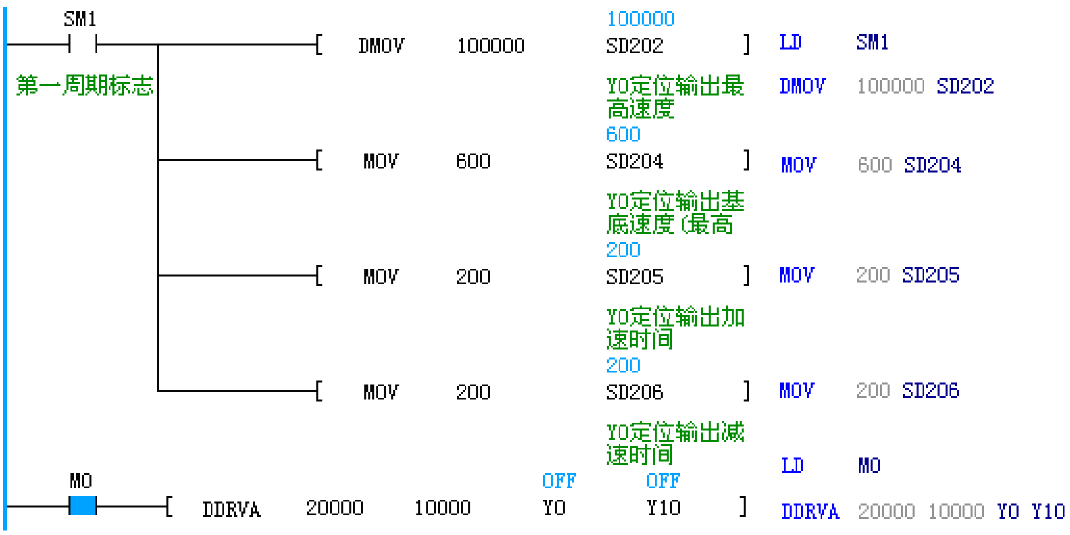


相关软元件：

地址	名称	功能
SM29	指令执行结束标志	定位指令执行结束时，置 ON
SM70 ～ SM77	Y0 ～ Y7 输出完成中断使能标志	置 ON，使能 PT0（Y0 ～ Y7）输出完成中断
SM80 ～ SM81、SM261 ～ SM266	Y0 ～ Y7 脉冲输出停止使能标志	置 ON，高速脉冲输出停止
SM82 ～ SM83、SM267 ～ SM272	Y0 ～ Y7 脉冲输出监控位	Y0 ～ Y7 高速脉冲输出监视
SM283/SM293/SM313 /SM323/SM333/SM343/ SM353/ SM363	Y0 ～ Y7 正转极限标志	适用于 DZRN/DSZR/DDRVI/DDRVA，作用于 Y0 对应的轴：置 ON 时，表示到达正转方向的极限；置 OFF 时，表示未到极限
SM284/SM294/SM314 /SM324/SM334/SM344 /SM354/SM364	Y0 ～ Y7 反转极限标志	适用于 DZRN/DSZR/DDRVI/DDRVA，作用于 Y0 对应的轴：置 ON 时，表示到达反转方向的极限；置 OFF 时，表示未到极限
SM288/SM298/SM318 /SM328/SM338/SM348 /SM358/SM368	Y0 ～ Y7 定位指令运行标志	用于监测执行 DZRN/DDRVI/DDRVA/DSZR /DDVIT 时高速输出通道 Y0 的状态，正在运行时为 ON，空闲时为 OFF

地址	名称	功能
SD50/SD52/SD70/SD72 /SD74/SD76/SD78/SD80	Y0 ～ Y7 发送脉冲总数（低位）	Y0 ～ Y7 发送脉冲总数，与脉冲方向无关
SD51/DSD53/SD71/SD73 /SD75/SD77/SD79/SD81	Y0 ～ Y7 发送脉冲总数（高位）	
SD200/SD220/SD260/SD280 /SD300/SD320/SD340/SD360	Y0 ～ Y7 定位输出当前值（低位）	用于绝对位置的计算和保存。每次执行定位指令，都根据（SD53，SD52）和方向信号来计算和更新（SD221，SD220）。当开机并且从伺服驱动器读取绝对位置数据时，请将得到的绝对位置数据（32 位长整型）放入（SD221，SD220）中
SD201/SD221/SD261/SD281 /SD301/SD321/SD341/SD361	Y0 ～ Y7 定位输出当前值（高位）	
SD202/SD222/SD282/SD302 /SD322/SD342/SD362/SD382	Y0 ～ Y7 定位输出最高速度（低位）	执行定位指令时的最高速度，范围：10 ～ 100000，单位：脉冲
SD203/SD223/SD283/SD303 /SD323/SD343/SD363/SD383	Y0 ～ Y7 定位输出最高速度（高位）	
SD204/SD224/SD284/SD304 /SD324/SD344/SD364/SD384	Y0 ～ Y7 定位输出基底速度	执行定位指令时的基底速度（最高速度的 1/10 以下）
SD205/SD225/SD285/SD305 /SD325/SD345/SD365/SD385	Y0 ～ Y7 定位输出加速时间	执行定位指令时的加速时间，范围：2 ～ 5000，单位：ms
SD206/SD226/SD286/SD306 /SD326/SD346/SD366/SD386	Y0 ～ Y7 定位输出减速时间	执行定位指令时的减速时间，范围：2 ～ 5000，单位：ms

使用示例



注意事项

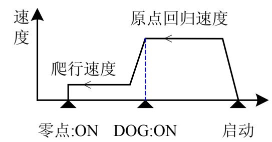
1. 当指令里输入的频率速度小于 SD204/SD224/SD284/SD304/SD324/SD344/SD364/SD384 中的值时，Y0 ～ Y7 无高速输出；当指令里输入的频率速度大于（SD203，SD202）/（SD223，SD222）/ SD283，SD282）/（SD303，SD302）/（SD323，SD322）/（SD343，SD342）/（SD363，SD362）/（SD383，SD382）中的值时，会造成输出不正常。
2. 在指令执行过程中，即使改变操作数的内容，也不会反映到当前的运行中。在下次的指令驱动时才有效。
3. 在指令执行过程中，驱动触点为 OFF 时，减速停止。且此时指令执行结束标志位不动作。
4. 动作方向的极限标志位动作时，减速停止。
5. 脉冲输出中监控为 ON 时，使用该输出的定位指令不能执行。此外，及时指令驱动触点为 OFF，在脉冲输出中监控为 ON 期间，也请不要执行指定了同一输出编号的定位指令。

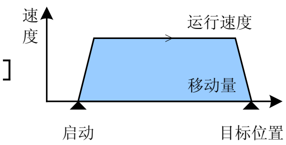
6.8.6 DSZR：带 DOG 原点回归

	DSZR		S1 S2 D1 D2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位： ----- ----- 32位：DSZR (连续) 9步 -----			相关元件	*			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	BOOL	●		●	●																				
S2	BOOL	*																							
D1	BOOL		*																						
D2	BOOL		*	●	●																				

* 使用 X 元件时，操作数 S2 只能使用 X0 ～ X7，使用 Y 元件时，操作数 D1 可指定 Y0 ～ Y7。
* 对于 D2 操作数 MX2V 只支持 Y 元件。
* 对于 D1、D2 作为脉冲输出端口时，MX1H 系列只能使用 Y0 ～ Y1；MX2H 系列只能使用 Y0 ～ Y2；MX2V 系列只能使用 Y0 ～ Y7。
* 相关元件：详见指令说明。

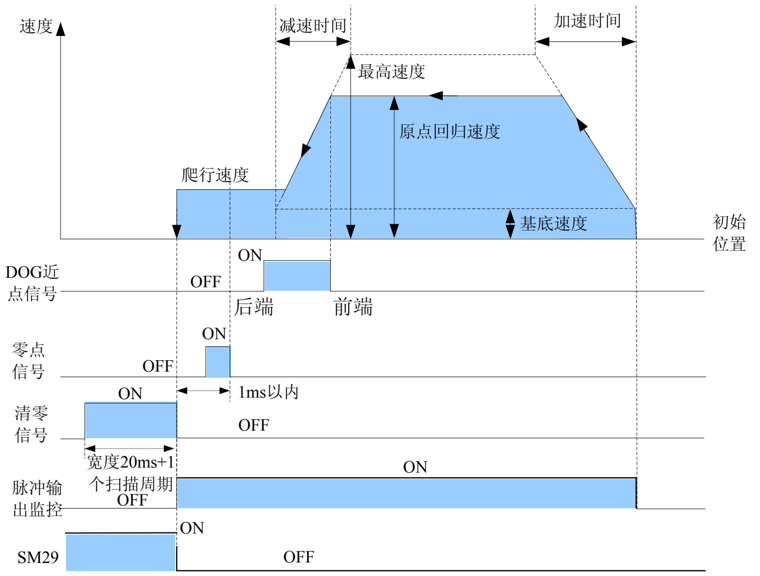




指令说明

├──┤ ├──[DSZR S1 S2 D1 D2]

操作数设置	操作数功能描述
S1	指定输入近点信号（DOG）的软元件编号。当指定输入软元件时，由于受到可编程控制器运算周期的影响，会引起原点位置的偏移增大。
S2	指定输入零点信号的软元件编号。范围：X0 ～ X7。
D1	脉冲输出方式。 MX1H/2H：脉冲 + 方向，方向端口不限定； MX2V：三种脉冲输出方式，脉冲 + 方向，正反向脉冲，AB 相输出，输出端口由系统组态设定，若指令输出端口与系统组态设定不同，系统会提示错误。
D2	



1．调用 DSZR 指令时执行原点回归动作，以原点回归速度移动，当近点信号 DOG 为 ON，就开始减速，减速到爬行速度后以爬行速度运行，当原点信号为 ON 后，就立即停止脉冲输出，指令执行标志位

SM29 为 ON，结束原点回归动作。

2．使用该指令时，请务必进行最高速度、基底速度、加速时间及减速时间特殊寄存器值的设定，否则指令不会正常运行。

3．加速时间是指由基底速度加速到最高速度的时间，减速时间是指由最高速度减速到基底速度的时间。

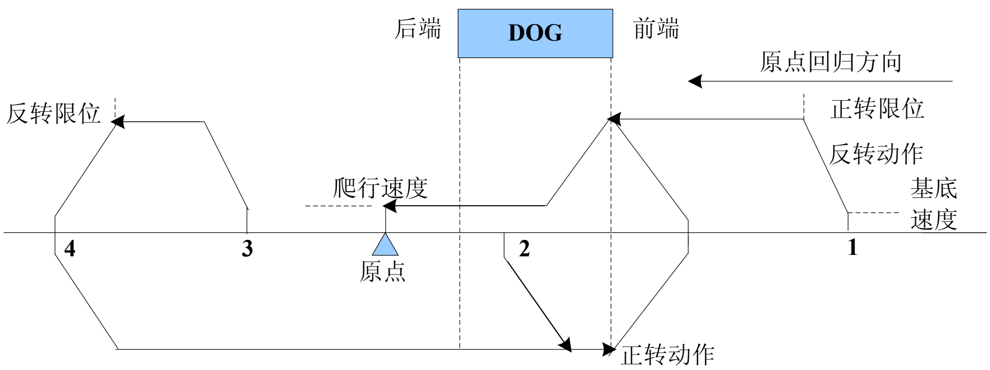
4．在原点回归过程中，当前值寄存器数值将向减少方向动作。

5．清零信号输出功能有效时，在脉冲输出停止后 1ms 以内，清零信号在 20ms+1 个扫描周期的时间内保持为 ON。

6．SM280/SM290/SM310/SM320/SM330/SM340/SM350/SM360 分别为 Y0 ～ Y7 清零功能有效。MX1H 清零信号为 Y0 ～ Y1，对应清零信号的输出点分别为 Y10 ～ Y11；MX2H 清零信号为 Y0 ～ Y2，对应清零信号的输出点分别为 Y10 ～ Y12；MX2V 清零信号为 Y0 ～ Y7，对应清零信号的输出点分别为 Y20 ～ Y27。若置位 SM280/SM290/SM310/SM320/SM330/SM340/SM350/SM360 和 SM281/SM291/SM311/SM321/SM331/SM341/SM351/SM361，则清零信号根据 SD210/SD230/SD290/SD310/SD330/SD350/SD370/SD390 指定的 Y 元件（SD210/SD230/SD290/SD310/SD330/SD350/SD370/SD390 里存储清零信号 Y 元件的序号）；若只置位 SM280/SM290，则清零信号为 Y10/Y11。

7．如果在原点回归指令运行过程中，断开指令前的驱动触点，则会减速停止。这样可以减小机械振动。

8．允许使用近点信号和零点信号的原点回归指令。设计有正转限位、反转限位。因原点回归的位置不同，原点回归的动作也不相同。指令结束后发清零信号。



1．开始位置在通过 DOG 前时：

（1）通过执行原点回归指令，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，再检测出第一个零点信号时停止。

2．开始位置在 DOG 内时：

（1）通过执行原点回归指令，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移动。
（3）检测出 DOG 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 前端就开始减速到爬行速度。
（6）检测出 DOG 后端后，再检测出第一个零点信号时停止。

3．开始位置在近点信号 OFF（通过 DOG 后）时：

（1）通过执行原点回归指令，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）检测出反转限位时减速停止。
（4）以原点回归速度向原点回归相反的方向开始移动。
（5）检测出 DOG 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。
（7）检测出 DOG 前端，开始减速到爬行速度。
（8）检测出 DOG 后端后，再检测出第一个零点信号时停止。

4．开始位置在限位开关位置上（正转限位或反转限位）：

（1）通过执行原点回归指令，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 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 前端，开始减速到爬行速度。
（6）检测出 DOG 后端后，再检测出第一个零点信号时停止。

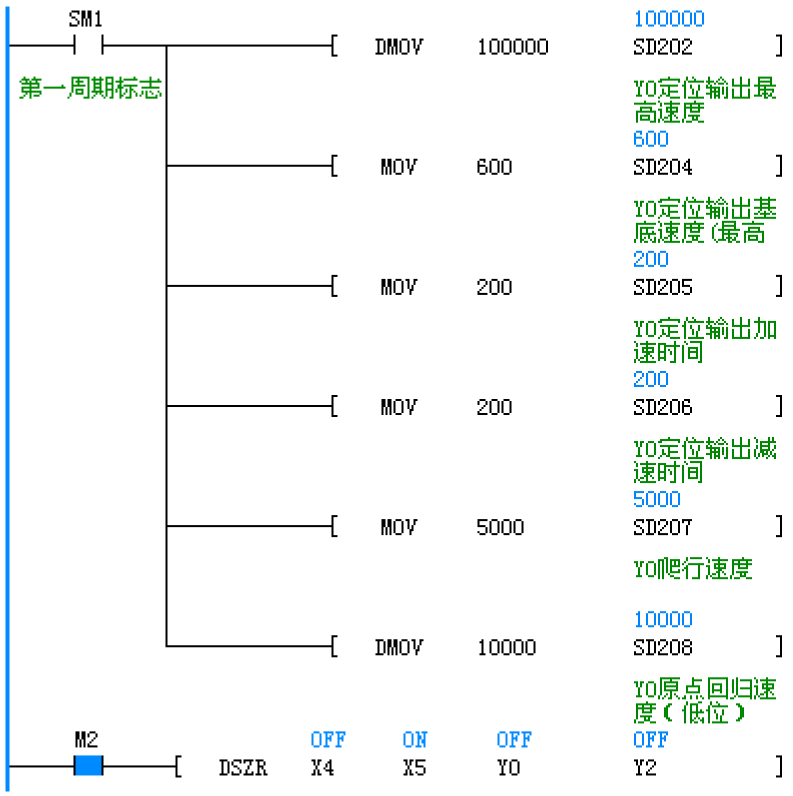
使用该指令时，请务必进行最高速度、基底速度、加速时间及减速时间特殊寄存器值的设定，否则指令不会正常运行。

相关软元件：

地址	名称	功能
SM29	指令执行结束标志	定位指令执行结束时，置 ON
SM70 ～ SM77	Y0 ～ Y7 输出完成中断使能标志	置 ON，使能 PT0（Y0 ～ Y7）输出完成中断
SM80 ～ SM81、SM261 ～ SM266	Y0 ～ Y7 脉冲输出停止使能标志	置 ON，高速脉冲输出停止
SM82/SM83 、SM267 ～ SM272	Y0 ～ Y7 脉冲输出监控位	Y0 ～ Y7 高速脉冲输出监视
SM280/SM290/SM310 /SM320/SM330/SM340 /SM350/SM360	Y0 ～ Y7 清零功能有效	适用 DZRN /DSZR, 定位指令原点回归（DZRN）清零信号输出功能有效（Y0）
SM281/SM291/SM311 /SM321/SM331/SM341 /SM351/SM361	Y0 ～ Y7 清零信号指定元件有效	适用 DZRN/DSZR，对应的 SD210 内的数值为 Y0 表示清零信号，不指定则 Y10 为 Y0 的清零信号
SM282/SM292/SM312 /SM322/SM332/SM342 /SM352/SM362	Y0 ～ Y7 原点回归方向标志	适用于 DZRN/DSZR，作用于 Y0 对应的轴：置 ON 时，表示原点回归方向为正转方向；置 OFF 时，表示原点回归方向为反转方向
SM283/SM293/SM313 /SM323/SM333/SM343 /SM353/ SM363	Y0 ～ Y7 正转极限标志	适用于 DZRN/DSZR/DDRVI/DDRVA，作用于 Y0 对应的轴：置 ON 时，表示到达正转方向的极限；置 OFF 时，表示未到极限
SM284/SM294/SM314 /SM324/SM334/SM344 /SM354/ SM364	Y0 ～ Y7 反转极限标志	适用于 DZRN/DSZR/DDRVI/DDRVA，作用于 Y0 对应的轴：置 ON 时，表示到达反转方向的极限；置 OFF 时，表示未到极限
SM285/SM295/SM315 /SM325/SM335/SM345 /SM355/SM365	Y0 ～ Y7 近点信号逻辑反转使能标志	适用于 DZRN/DSZR，作用于 Y0 对应的轴：置 ON 时，按负逻辑处理（输入 OFF 时，近点信号为 ON）；置 OFF 时，按正逻辑处理（输入 ON 时，近点信号为 ON）
SM288/SM298/SM318 /SM328/SM338/SM348 /SM358/SM368	Y0 ～ Y7 定位指令运行标志	用于监测执行 DZRN/DDRVI/DDRVA /DSZR/DDVIT 时高速输出通道 Y0 的状态，正在运行时为 ON，空闲时为 OFF
SD50/SD52/SD70/SD72 /SD74/SD76/SD78/SD80	Y0 ～ Y7 发送脉冲总数（低位）	Y0 ～ Y7 发送脉冲总数，与脉冲方向无关
SD51/DSD53/SD71/SD73 /SD75/SD77/SD79/SD81	Y0 ～ Y7 发送脉冲总数（高位）	
SD200/SD220/SD280/SD300 /SD320/SD340/SD360/SD380	Y0 ～ Y7 定位输出当前值（低位）	用于绝对位置的计算和保存。每次执行定位指令，都根据（SD53，SD52）和方向信号来计算和更新（SD221，SD220）。当开机并且从伺服驱动器读取绝对位置数据时，请将得到的绝对位置数据（32 位长整型）放入（SD221，SD220）中
SD201/SD221/SD281/SD301 /SD321/SD341/SD361/SD381	Y0 ～ Y7 定位输出当前值（高位）	
SD202/SD222/SD282/SD302 /SD322/SD342/SD362/SD382	Y0 ～ Y7 定位输出最高速度（低位）	执行定位指令时的最高速度，范围：10 ～ 100000，单位：脉冲
SD203/SD223/SD283/SD303 /SD323/SD343/SD363/SD383	Y0 ～ Y7 定位输出最高速度（高位）	
SD204/SD224/SD284/SD304 /SD324/SD344/SD364/SD384	Y0 ～ Y7 定位输出基底速度	执行定位指令时的基底速度（最高速度的 1/10 以下）

地址	名称	功能
SD205/SD225/SD285/SD305 /SD325/SD345/SD365/SD385	Y0 ～ Y7 定位输出加速时间	执行定位指令时的加速时间，范围：2 ～ 5000，单位：ms
SD206/SD226/SD286/SD306 /SD326/SD346/SD366/SD386	Y0 ～ Y7 定位输出减速时间	执行定位指令时的减速时间，范围：2 ～ 5000，单位：ms
SD207/SD227/SD287/SD307 /SD327/SD347/SD367/SD387	Y0 ～ Y7 爬行速度	适用于 DZRN/DSZR，作为执行该指令时的爬行速度
SD208/SD228/SD288/SD308 /SD328/SD348/SD368/SD388	Y0 ～ Y7 原点回归速度（低位）	适用于 DZRN/DSZR，作为执行该指令时的原点回归速度
SD209/SD229/SD289/SD309 /SD329/SD349/SD369/SD389	Y0 ～ Y7 原点回归速度（高位）	
SD210/SD230/SD290/SD310 /SD330/SD350/SD370/SD390	Y0 ～ Y7 清零信号软元件指定	SM291 置位时，使用该元件中的数值 N 对应的 Y 元件 Y（N）表示清零信号

使用示例

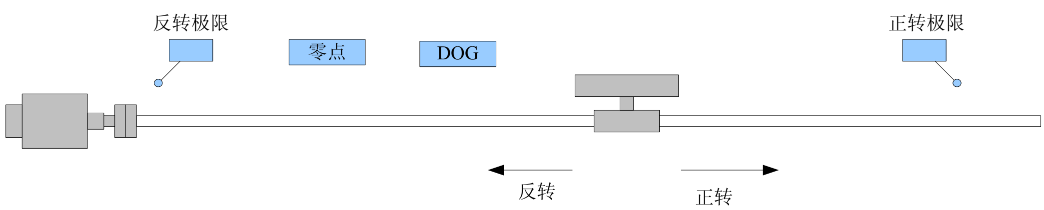



```
LD    SM1
DMOV  100000 SD202
MOV   600 SD204
MOV   200 SD205
MOV   200 SD206
MOV   5000 SD207
DMOV  10000 SD208
LD    M2
DSZR  X4 X5 Y0 Y2
```

注意事项

1．设计近点信号（DOG）时，请考虑有足够为 ON 的时间能充分减速到爬行速度。在 DOG 的后端前，没有 能够减速到爬行速度时，会导致停止位置偏移。

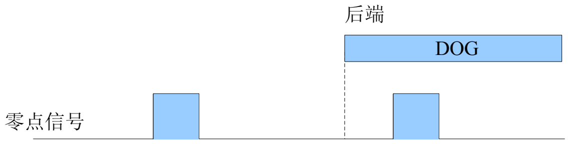
2．近点信号（DOG）请设在正转极限和反转极限之间。近点信号（DOG）、正转极限、反转极限未形成下图关系时，可能无法执行希望的动作。



3．近点信号 S1、零点信号 S2 中指定的输入信号不能和以下的用途重复使用：高速计数器、输入中断、脉冲捕捉、SPD 指令、DDVIT 指令、DZRN 指令。

4．因为使用伺服电机的零点信号，所以请按照以下关系调整 DOG 的后端和零点信号。或者在需要对原点

位置做微调时，请调整近点信号（DOG）的位置。



5．请使爬行速度足够的慢。原点回归指令的停止是不进行减速停止的，所以如果爬行速度过快，会由于惯性导致停止位置偏移。

6．如果在指令的执行过程中改变操作数的内容，那么不会反映动作上。想要将改变的内容反映到动作上时，请先断开指令的指令触点，然后再次置 ON。

7．如果原点回归过程中，指令的驱动触点变为 OFF，则减速停止。此时，指令执行结束标志位 SM29 不置 ON。

8. 即使指令驱动触点为 OFF，在脉冲输出中监控为 ON 期间，也请不要执行指定了同一输出编号的高速指令、定位指令。

9．实际的输出脉冲频率的最低频率数，根据以下公式决定：
输出脉冲频率的最低频率数＝ sqrt [最高速度 ×500÷ 加减速时间]

10. 对于输出脉冲频率数，即使指定了低于上面计算结果的数值，仍然将输出计算值的频率。加速初期和减速最终部分的频率也不可以低于上述计算结果。如果最高速度低于上述的计算结果，将没有脉冲输出。

6.8.7 DDVIT : 中断定位指令

	DDVIT		S1 S2 D1 D2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位： ----- ----- 32位：DDVIT (连续) 11步 -----			相关元件	*			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	DINT										●	●	●	●	●	●	●	●	*		●		●	●	
S2	DINT										●	●	●	●	●	●	●	●	*		●		●	●	
D1	BOOL		*																						
D2	BOOL		*	●	●																				

＊ 该指令使用 C 元件时，只能使用 C200 ～ C255。操作数 D1 使用 Y 元件时，只能使用 Y0 ～ Y7。

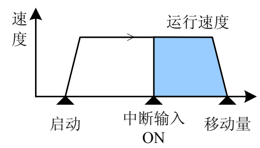
＊ 对于 D2 操作数 MX2V 只支持 Y 元件。

＊ 对于 D1、D2 作为脉冲输出端口时，MX1H 系列只能使用 Y0 ～ Y1；MX2H 系列只能使用 Y0 ～ Y2；MX2V 系列只能使用 Y0 ～ Y7。

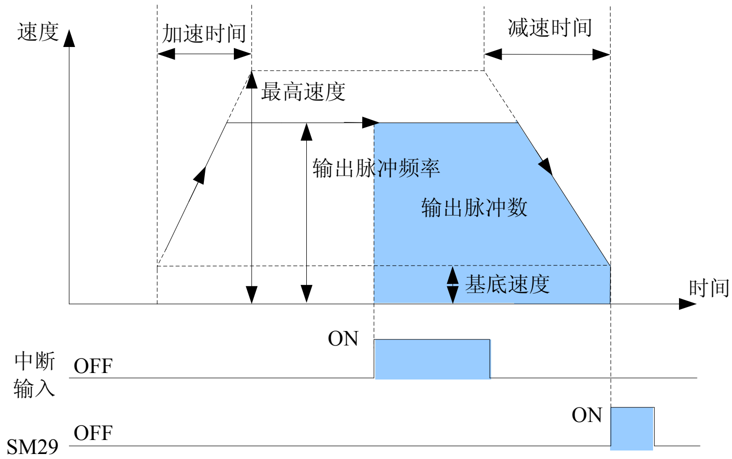
＊ 相关元件：详见指令说明。

指令说明

├──┤ ├──[DDVIT S1 S2 D1 D2]



操作数设置	操作数功能描述
S1	指定中断后的输出脉冲数（相对指令）。
S2	指定输出脉冲频率。
D1	脉冲输出方式。 MX1H/2H：脉冲 + 方向，方向端口不限定； MX2V：三种脉冲输出方式，脉冲 + 方向，正反向脉冲，AB 相输出，输出端口由系统组态设定，若指令输出端口与系统组态设定不同，系统会提示错误。
D2	



1. 调用 DDVIT 执行中断定位运行，即由基底速度加速至运行速度 S2，当中断信号输入后发送指定的脉冲数 S1 后减速停止，执行结束标志位为 ON，结束中断定位。
2. 使用该指令时，请务必进行最高速度、基底速度、加速时间及减速时间特殊寄存器值的设定，否则指令不会正常运行。
3. 加速时间是指由基底速度加速到最高速度的时间，减速时间是指由最高速度减速到基底速度的时间。

4．执行单速中断定长进给的指令。SM260 为中断输入功能指定有效；SM287 为 Y0 中断输入信号的逻辑反转使能标志。逻辑反转即确定中断软元件为 ON 还是 OFF 时产生中断。

5．中断输入的指定方法：SM260 置 ON。

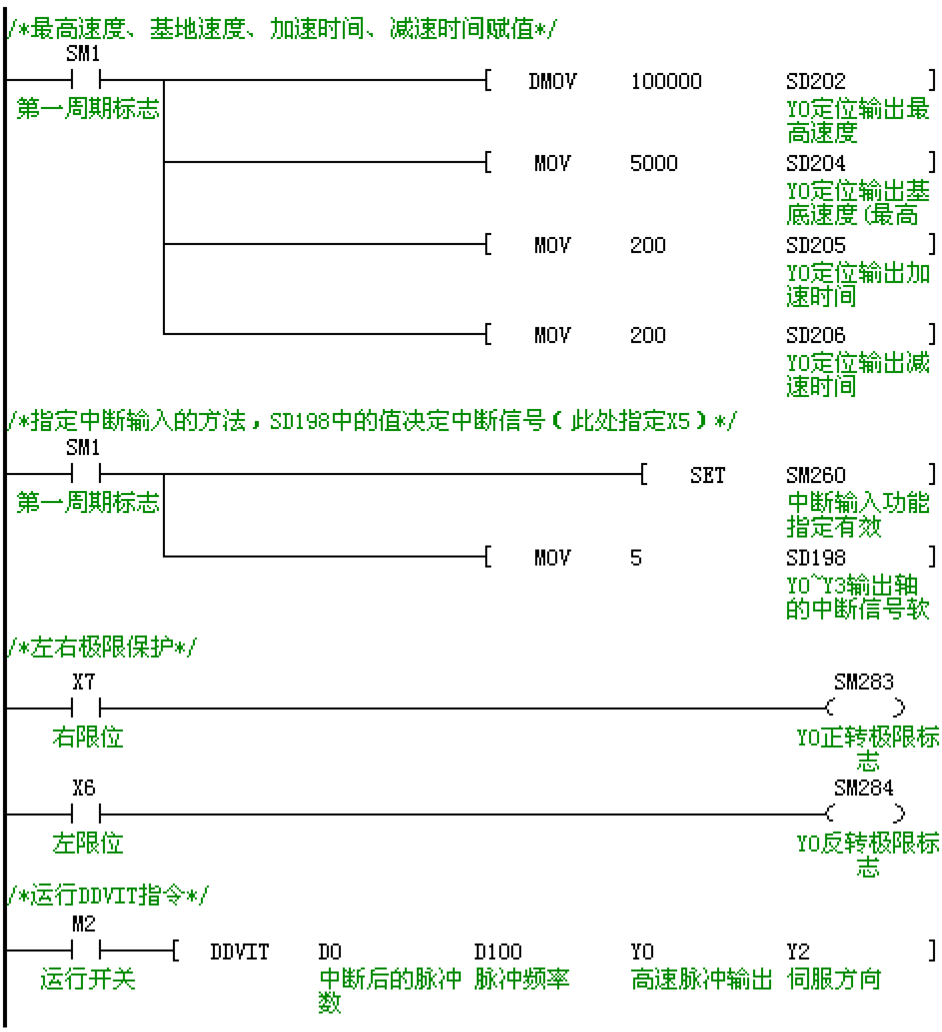
6．在 SD198 中指定作为中断输入的输入编号（X0 ～ X7），或者指定用户中断指令软元件。其中，SD198 中 b0 ～ b3 位对应脉冲输出 Y0 用的中断输入，b4 ～ b7 位对应脉冲输出 Y1 用的中断输入。

下表为 b0 ～ b3 及 b4 ～ b7 设定值所对应的关系

<table>
<tr><th>设定值</th><th>设定内容</th></tr>
<tr><td>0</td><td>将 X0 指定为中断输入信号</td></tr>
<tr><td>1</td><td>将 X1 指定为中断输入信号</td></tr>
<tr><td>……</td><td>……</td></tr>
<tr><td>7</td><td>将 X7 指定为中断输入信号</td></tr>
<tr><td>8</td><td>将 SM 元件指定为中断输入信号<table><tr><th>脉冲输出软元件</th><th>中断输入信号</th></tr><tr><td>Y0</td><td>SM289：Y0 用户中断输入使能标志</td></tr><tr><td>Y1</td><td>SM299：Y1 用户中断输入使能标志</td></tr></table></td></tr>
</table>

示例：

以下使用 Y0 输出脉冲信号，Y2 输出方向信号，指令 X5 为中断输入信号。

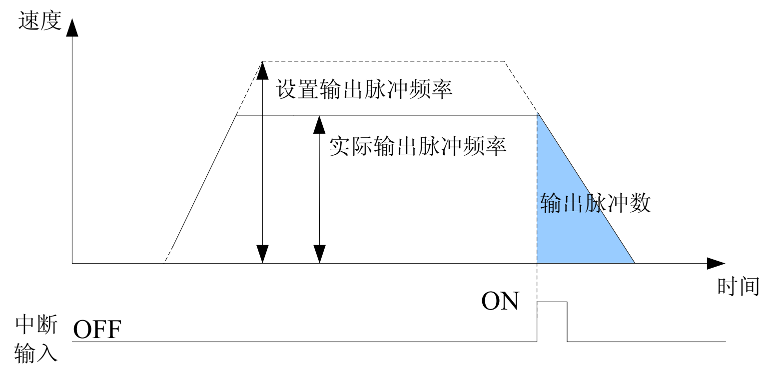


相关软元件：

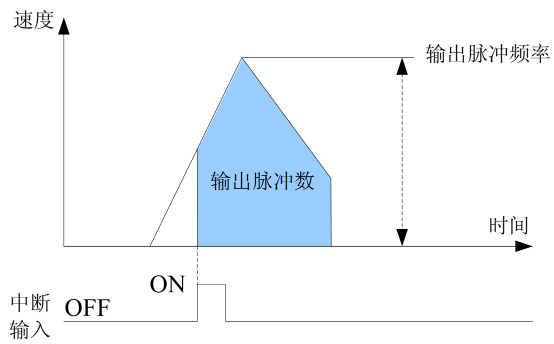
地址	名称	功能
SM29	指令执行结束标志	定位指令执行结束时，置 ON
SM70 ～ SM77	Y0 ～ Y7 输出完成中断使能标志	置 ON，使能 PT0（Y0 ～ Y7）输出完成中断
SM80 ～ SM81、SM261 ～ SM266	Y0 ～ Y7 脉冲输出停止使能标志	置 ON，高速脉冲输出停止
SM82 ～ SM83、SM267 ～ SM272	Y0 ～ Y7 脉冲输出监控位	Y0 ～ Y7 高速脉冲输出监视
SM283/SM293/SM313/SM323/SM333/SM343/SM353/ SM363	Y0 ～ Y7 正转极限标志	适用于 DZRN/DSZR/DDRVI/DDRVA，作用于 Y0 对应的轴：置 ON 时，表示到达正转方向的极限；置 OFF 时，表示未到极限
SM284/SM294/SM314/SM324/SM334/SM344/SM354/ SM364	Y0 ～ Y7 反转极限标志	适用于 DZRN/DSZR/DDRVI/DDRVA，作用于 Y0 对应的轴：置 ON 时，表示到达反转方向的极限；置 OFF 时，表示未到极限
SM288/SM298/SM318/SM328/SM338/SM348/SM358/SM368	Y0 ～ Y7 定位指令运行标志	用于监测执行 DZRN/DDRVI/DDRVA/DSZR/DDVIT 时高速输出通道 Y0 的状态，正在运行时为 ON，空闲时为 OFF
SD50/SD52/SD70/SD72/SD74/SD76/SD78/SD80	Y0 ～ Y7 发送脉冲总数（低位）	Y0 ～ Y7 发送脉冲总数，与脉冲方向无关
SD51/SD53/SD71/SD73/SD75/SD77/SD79/SD81	Y0 ～ Y7 发送脉冲总数（高位）	
SD200/SD220/SD260/SD280/SD300/SD320/SD340/SD360	Y0 ～ Y7 定位输出当前值（低位）	用于绝对位置的计算和保存。每次执行定位指令，都根据（SD53，SD52）和方向信号来计算和更新（SD221，SD220）。当开机并且从伺服驱动器读取绝对位置数据时，请将得到的绝对位置数据（32 位长整型）放入（SD221，SD220）中
SD201/SD221/SD261/SD281/SD301/SD321/SD341/SD361	Y0 ～ Y7 定位输出当前值（高位）	
SD202/SD222/SD282/SD302/SD322/SD342/SD362/SD382	Y0 ～ Y7 定位输出最高速度（低位）	执行定位指令时的最高速度，范围：10 ～ 100000，单位：脉冲
SD203/SD223/SD283/SD303/SD323/SD343/SD363/SD383	Y0 ～ Y7 定位输出最高速度（高位）	
SD204/SD224/SD284/SD304/SD324/SD344/SD364/SD384	Y0 ～ Y7 定位输出基底速度	执行定位指令时的基底速度（最高速度的 1/10 以下）
SD205/SD225/SD285/SD305/SD325/SD345/SD365/SD385	Y0 ～ Y7 定位输出加速时间	执行定位指令时的加速时间，范围：2 ～ 5000，单位：ms
SD206/SD226/SD286/SD306/SD326/SD346/SD366/SD386	Y0 ～ Y7 定位输出减速时间	执行定位指令时的减速时间，范围：2 ～ 5000，单位：ms

注意事项

1. 高速指令、包络线指令、定位指令可以利用 Y0 ～ Y7 八个端口输出高速脉冲。请注意不要同时对同一个高速端口使用这些指令进行高速输出。
2. 实际能够输出的最低频率数，根据以下公式决定：
   输出脉冲的最低频率数＝ sqrt [最高速度 ×500÷ 加减速时间]
3. 对于输出脉冲频率数 S2，即使指定了低于上面计算结果的数值，仍然将输出计算值的频率。加速初期和减速最终部分的频率也不可以低于上述计算结果。但如果最高速度低于上述的计算结果，脉冲将没有输出。
4. 输出脉冲数比减速所需脉冲数少时可以完成减速的频率动作。



5．设置输出脉冲数时，建议输出脉冲数≥加速所需脉冲数＋减速所需脉冲数。如果输出脉冲数＜加速所需脉冲数＋减速所需脉冲数，运行如下图所示：



6．在指令执行过程中，即使更改操作数的内容，也不会立即反映到动作中，只有下一次执行指令时才有效。
7．在动作过程中，指令的驱动触点为 OFF 时，减速停止，此时 SM29 指令结束标志位不动作。
8．请在输出脉冲数带到 4294967296 个脉冲前输入中断信号。如果在中断输入前，脉冲输出达到 4294967296 个脉冲，则停止，指令结束标志位 SM29 为 ON。
9．脉冲输出监控位为 ON 时，使用该输出的定位指令不能执行。即使指令驱动前的触点为 OFF，但在脉冲输出监控位为 ON 期间，也不要执行指定了同一输出编号的定位指令。
10．运行过程中，正转极限标志位或者反转极限标志位为 ON 时，减速停止。
11．中断输入不能和以下指令重复使用：高速计数器、输入中断、脉冲捕捉、SPD 指令、DSZR 指令、DZRN 指令。

6.8.8 HPRC : 手动脉冲控制指令

	HPRC		S1 S2 S3 D1 D2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位： ----- ----- 32位：HPRC (连续) 12步 -----			相关元件	*			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	DINT																		*						
S2	INT										•	•	•	•	•	•	•	•	*	•	•	•	•	•	•
S3	INT										•	•	•	•	•	•	•	•	*	•	•	•	•	•	•
D1	BOOL		*																						
D2	BOOL		*																						

* 操作数 S1 使用 C 元件时，只能使用 C235 ～ C255。操作数 S2 和 S3 使用 C 元件时，只能使用 C0 ～ C199。
* 对于 D1、D2 作为脉冲输出端口时，MX1H 系列只能使用 Y0 ～ Y1；MX2H 系列只能使用 Y0 ～ Y2；MX2V 系列只能使用 Y0 ～ Y7。
* 相关元件：详见指令说明。

指令说明

[HPRC S1 S2 S3 D1 D2]

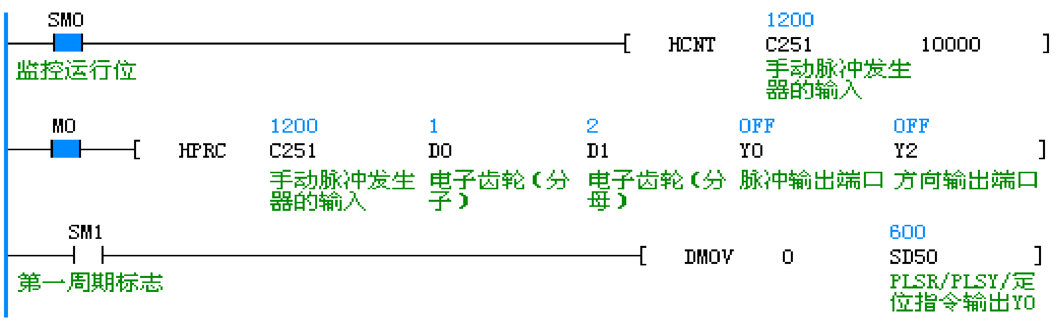
操作数设置	操作数功能描述
S1	检测脉冲输入计数器 C235~C255。
S2	脉冲输出的电子齿轮（分子），范围是 1~32767。
S3	脉冲输出的电子齿轮（分母），范围是 1~32767。
D1	脉冲输出方式。 MX1H/2H：脉冲 + 方向，方向端口不限定； MX2V：三种脉冲输出方式，脉冲 + 方向，正反向脉冲，AB 相输出，输出端口由系统组态设定，若指令输出端口与系统组态设定不同，系统会提示错误。
D2	

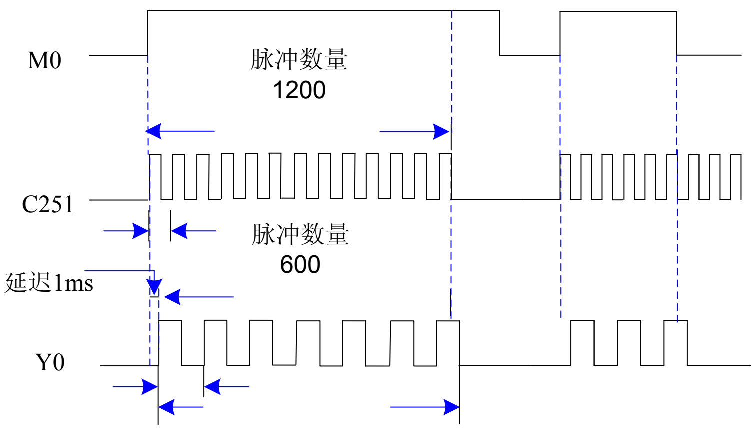
指定输出可以成比例的跟随手动脉冲发生器的输入变化，其中操作数 S2 对应输出端口的脉冲，操作数 S3 对应输入端口的脉冲。

相关软元件：

地址	名称	功能
SM80 ～ SM81、SM261 ～ SM266	Y0 ～ Y7 脉冲输出停止使能标志	置 ON，Y0 ～ Y7 高速脉冲输出停止
SM82 ～ SM83、SM267 ～ SM272	Y0 ～ Y7 脉冲输出监控位	Y0 ～ Y7 高速脉冲输出监视
SD50/SD52/SD70/SD72/SD74/SD76/SD78/SD80	Y0 ～ Y7 发送脉冲总数（低位）	Y0 ～ Y7 发送脉冲总数，与脉冲方向无关
SD51/DSD53/SD71/SD73/SD75/SD77/SD79/SD81	Y0 ～ Y7 发送脉冲总数（高位）	
SD200/SD220/SD260/SD280/SD300/SD320/SD340/SD360	Y0～Y7 定位输出当前值（低位）	用于绝对位置的计算和保存。每次执行定位指令，都根据（SD53，SD52）和方向信号来计算和更新（SD221，SD220）。当开机并且从伺服驱动器读取绝对位置数据时，请将得到的绝对位置数据（32 位长整型）放入（SD221，SD220）中
SD201/SD221/SD261/SD281/SD301/SD321/SD341/SD361	Y0～Y7 定位输出当前值（高位）	

使用示例





上例中，M0 为 ON 时，手动脉冲控制指令执行。D0：D1= 脉冲输出：手动脉冲发生器的输入脉冲

注意事项

1．请注意指令的驱动时间。

2．在启动该指令之前，先用 HCNT 指令对手动脉冲发生器的输入进行计数。

6.8.9 FOLLOW: 跟随指令

	FOLLOW		S1 S2 S3 D1 D2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位： ----- ----- 32位：FOLLOW (连续) 12步 -----			相关元件	*			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	DINT																		*						
S2	INT										•	•	•	•	•	•	•	•	*	•	•	•	•	•	•
S3	INT										•	•	•	•	•	•	•	•	*	•	•	•	•	•	•
D1	BOOL		*																						
D2	BOOL		*																						

* 操作数 S1 使用 C 元件时，只能使用 C235 ～ C255。操作数 S2 和 S3 使用 C 元件时，只能使用 C0 ～ C199。
 操作数 D1 使用 Y 元件时，MX1H 系列只能使用 Y0 ～ Y1；MX2H 系列只能使用 Y0 ～ Y2；MX2V 系列只能使用 Y0 ～ Y7。
* 相关元件：详见指令说明。
 相关功能：跟随功能，比 HPRC 指令的比例脉冲输出更准确。
 微调功能，SD212/SD232/SD292/SD312/SD332/SD352/SD372/SD392 FOLLOW 指令输出脉冲微调，脉冲输出微调值，范围为 -32,768 ～ 32,767，微调值为正值则为指令脉冲输出数 + 微调值；微调值为负值则为指令脉冲输出数 - 微调值。

指令说明

├──┤ ├──[FOLLOW S1 S2 S3 D1 D2]

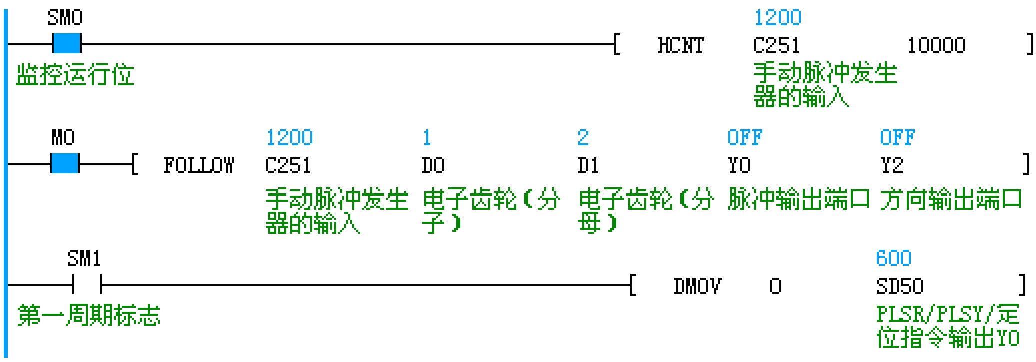
操作数设置	操作数功能描述
S1	检测脉冲输入计数器 C235~C255。
S2	脉冲输出的电子齿轮（分子），范围是 1~32767。
S3	脉冲输出的电子齿轮（分母），范围是 1~32767。
D1	脉冲输出端口。
D2	方向输出端口，只能指定使用 Y 元件。

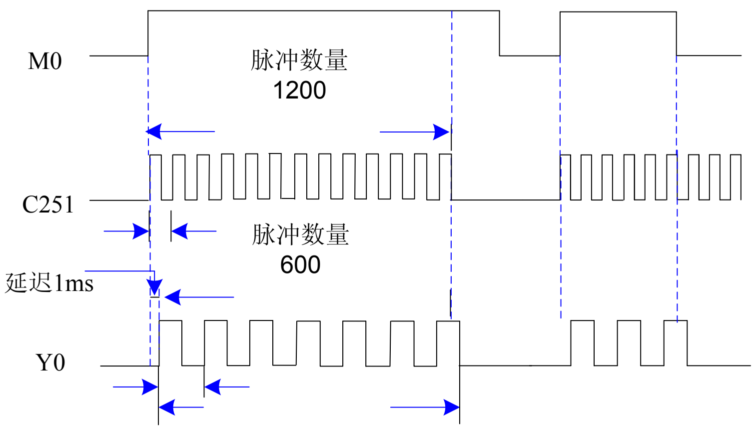
指令的指定输出可以成比例的跟随手动脉冲发生器的输入变化，其中操作数 S2 对应输出端口的脉冲，操作数 S3 对应输入端口的脉冲。

地址	名称	功能
SM80 ～ SM81、SM261 ～ SM266	Y0 ～ Y7 脉冲输出停止使能标志	置 ON，Y0 ～ Y7 高速脉冲输出停止
SM82 ～ SM83、SM267 ～ SM272	Y0 ～ Y7 脉冲输出监控位	Y0 ～ Y7 高速脉冲输出监视
SD50/SD52/SD70/SD72 /SD74/SD76/SD78/SD80	Y0 ～ Y7 发送脉冲总数（低位）	Y0 ～ Y7 发送脉冲总数，与脉冲方向无关
SD51/DSD53/SD71/SD73 /SD75/SD77/SD79/SD81	Y0 ～ Y7 发送脉冲总数（高位）	

地址	名称	功能
SD200/SD220/SD260/SD280/SD300/SD320/SD340/SD360	Y0 ～ Y7 定位输出当前值（低位）	用于绝对位置的计算和保存。每次执行定位指令，都根据（SD53，SD52）和方向信号来计算和更新（SD221，SD220）。当开机并且从伺服驱动器读取绝对位置数据时，请将得到的绝对位置数据（32 位长整型）放入（SD221，SD220）中
SD201/SD221/SD261/SD281/SD301/SD321/SD341/SD361	Y0 ～ Y7 定位输出当前值（高位）	
SD212/SD232/SD292/SD312/SD332/SD352/SD372/SD392	Y0 ～ Y7 FOLLOW 指令输出脉冲微调	Y0～Y7 FOLLOW 指令输出脉冲微调值，范围 -32768 ～ 32767

使用示例





上例中，M0 为 ON 时，跟随指令执行。D0：D1= 脉冲输出：跟随脉冲发生器的输入脉冲。

注意事项

1．请注意指令的驱动时间。
2．在启动该指令之前，先用 HCNT 指令对跟随脉冲发生器的输入进行计数。

6.8.10 FOLLOWA: 可设定比例补偿脉冲及输出脉冲个数的跟随指令

	FOLLOWA		S1 S2 S3 S4 S5 D1 D2	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√	√
步长	16位： ----- -----			相关元件	*			
	32位：FOLLOWA (连续) 20步 -----							

操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	DINT																		*						
S2	DINT										●	●	●	●	●	●	●	●	*		●		●	●	●
S3	DINT										●	●	●	●	●	●	●	●	*		●		●	●	●
S4	DINT										●	●	●	●	●	●	●	●	*		●		●	●	●
S5	DINT										●	●	●	●	●	●	●	●	*				●	●	●
D1	BOOL		*																						
D2	BOOL		*																						

* 操作数 S1 使用 C 元件时，只能使用 C235 ～ C255。操作数 S2 和 S3 使用 C 元件时，只能使用 C0 ～ C199。

* 对于 D1、D2 作为脉冲输出端口时，MX1H 系列只能使用 Y0 ～ Y1；MX2H 系列只能使用 Y0 ～ Y2；MX2V 系列只能使用 Y0 ～ Y7。

* 相关元件：详见指令说明。

相关功能：微调功能，SD212/SD232/SD292/SD312/SD332/SD352/SD372/SD392 Y0 ～ Y7 FOLLOWA 指令输出脉冲微调，脉冲输出微调值范围为 -32768 ～ 32767，微调值为正值则为指令脉冲输出数 + 微调值；微调值为负值则为指令脉冲输出数—微调值。

指令说明

├──┤ ├──[FOLLOWA S1 S2 S3 S4 S5 D1 D2]

操作数设置	操作数功能描述
S1	检测脉冲输入计数器 C235~C255。
S2	脉冲输出的电子齿轮（分子），范围是 1~32767。
S3	脉冲输出的电子齿轮（分母），范围是 1~32767。
S4	脉冲输出的微调齿轮比（分母），微调补偿完毕脉冲个数范围，范围是 0 ～ 2147483647；分子等于 SD212/SD232/SD292/SD312/SD332/SD352/SD372/SD392 的值
S5	脉冲输出指定产生的脉冲量，范围是 0 ～ 2147483647；
D1	脉冲输出端口。
D2	方向输出端口。

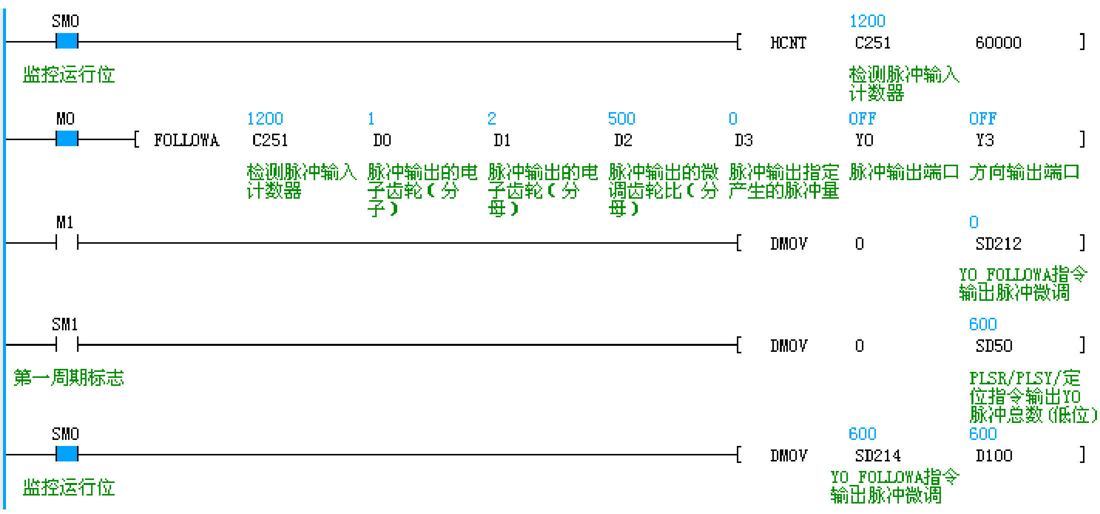
指定输出可以成比例的跟随检测脉冲输入计数器的输入变化，其中操作数 S2 对应输出端口的脉冲，操作数 S3 对应输入端口的脉冲。

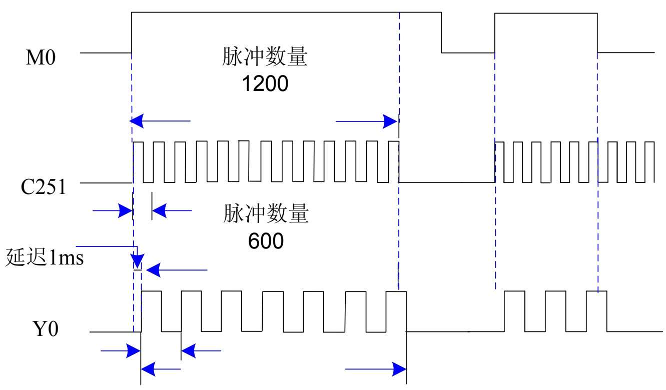
地址	名称	功能
SM29	指令执行结束标志	指令动作结束时接通
SM70 ～ SM77	Y0 ～ Y7 输出完成中断使能标志	置 ON 时，使能 PTO(Y0 ～ Y7) 输出完成中断
SM80 ～ SM81、SM261 ～ SM266	Y0 ～ Y7 脉冲输出停止使能标志	置 ON，Y0 ～ Y7 高速脉冲输出停止
SM82 ～ SM83、SM267 ～ SM272	Y0 ～ Y7 脉冲输出监控位	Y0 ～ Y7 高速脉冲输出监视

地址	名称	功能
SM150 ～ SM157	Y0 ～ Y7 FOLLOWA 指令脉冲方向强制	当补偿脉冲 + 正常输出脉冲频率 >100KHZ 时，置位相应的超调标志；当补偿脉冲 + 正常输出脉冲频率 <=100KHZ 时，清除相应的超调标志
SD50/SD52/SD70/SD72/SD74/SD76/SD78/SD80	Y0 ～ Y7 发送脉冲总数（低位）	Y0 ～ Y7 发送脉冲总数，与脉冲方向无关
SD51/DSD53/SD71/SD73/SD75/SD77/SD79/SD81	Y0 ～ Y7 发送脉冲总数（高位）	
SD200/SD220/SD260/SD280/SD300/SD320/SD340/SD360	Y0 ～ Y7 定位输出当前值（低位）	用于绝对位置的计算和保存。每次执行定位指令，都根据（SD53，SD52）和方向信号来计算和更新（SD221，SD220）。当开机并且从伺服驱动器读取绝对位置数据时，请将得到的绝对位置数据（32 位长整型）放入（SD221，SD220）中
SD201/SD221/SD261/SD281/SD301/SD321/SD341/SD361	Y0 ～ Y7 定位输出当前值（高位）	
SD212/SD232/SD292/SD312/SD332/SD352/SD372/SD392	Y0 ～ Y7 FOLLOWA 指令输出脉冲微调	Y0 ～ Y7 FOLLOWA 指令输出脉冲微调值，范围 -32768 ～ 32767

使用示例

示例 1.



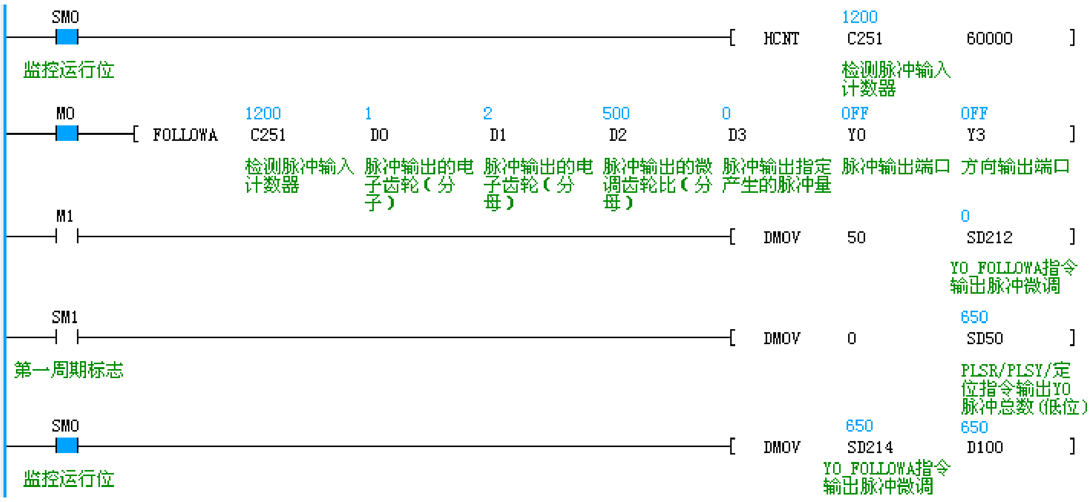


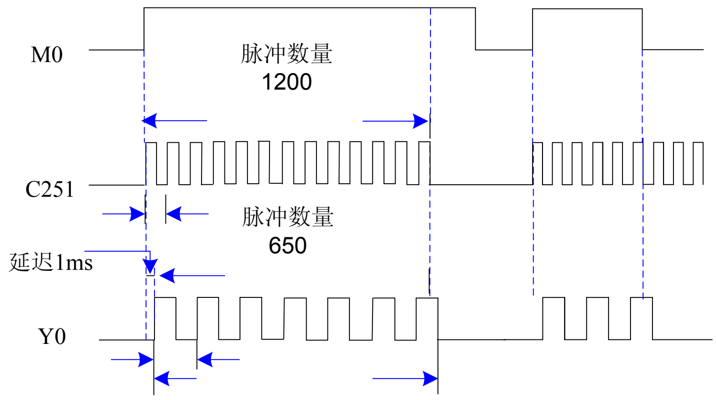
1. M0 为 ON 时，FOLLOWA 指令执行。
2. 输出脉冲个数 =（D0：D1）*（检测脉冲输入计数器当前值 - 检测脉冲输入计数器能流导通时的值）+ Y0_FOLLOWA 指令输出脉冲微调值。

   即：SD214 = (1：2)*C251 差值 +SD212，在此例中 Y0_FOLLOWA 指令输出脉冲微调 SD212=0，所以 Y0 输

出脉冲个数 =600+0=600。

示例 2.

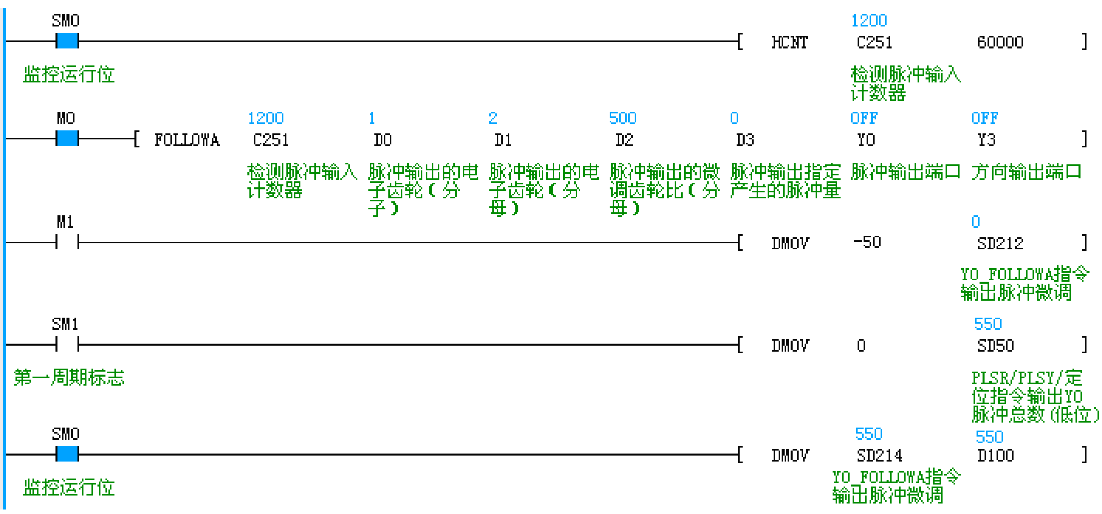


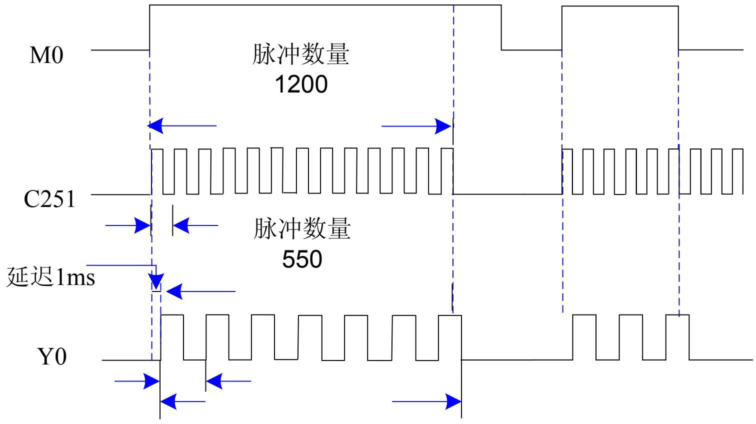


1. M1 置 ON，将 50 写入 SD212 中，然后 M1 置 OFF，M0 为 ON 时，FOLLOWA 指令执行。
2. 输出脉冲个数 =（D0：D1）*（检测脉冲输入计数器当前值 - 检测脉冲输入计数器能流导通时的值）+ Y0_FOLLOWA 指令输出脉冲微调值。

   即：SD214 = (1：2)*C251 差值 +SD212，在此例中 Y0_FOLLOWA 指令输出脉冲微调 SD212=50，所以 Y0 输出脉冲个数 =600+50=650。需补偿的 50 个脉冲在 D2（500）个脉冲的范围内均匀补完。
3. 注：FOLLOWA 指令能流导通后，当 C251 计数值有变化，则将 SD212 读入后自动清除 SD212 的值。

示例 3.





1．M1 置 ON，将 -50 写入 SD212 中，然后 M1 置 OFF，M0 为 ON 时，FOLLOWA 指令执行。
2．输出脉冲个数 =（D0：D1）*（检测脉冲输入计数器当前值 - 检测脉冲输入计数器能流导通时的值）+ Y0_FOLLOWA 指令输出脉冲微调值。
即：SD214 = (1：2)*C251 差值 +SD212，在此例中 Y0_FOLLOWA 指令输出脉冲微调 SD212=-50，所以 Y0 输出脉冲个数 =600+（-50）=550。需补偿的 -50 个脉冲在 D2（500）个脉冲的范围内均匀补完。
3．注：FOLLOWA 指令能流导通后，当 C251 计数值有变化，则将 SD212 读入后自动清除 SD212 的值。

注 意
事 项

1．请注意指令的驱动时间。
2．在启动该指令之前，先用 HCNT 指令对跟随脉冲发生器的输入进行计数。
3．S4 为脉冲输出的微调齿轮比（分母），微调补偿完毕脉冲个数范围，范围是 0-2147483647；其意义为在 S4 个脉冲范围内按照 SD212/SD232/SD292/SD312/SD332/SD352/SD372/SD392：S4 的齿轮比，均匀的将补偿脉冲补偿完。
4．S5 为脉冲输出指定产生的脉冲量，范围是 0-2147483647；当 S5 设为 0 的时候，FOLLOWA 指令的输出脉冲无个数限制；当 S5 设置为非 0 数值的时候，当输出脉冲到达 S5 的个数，FOLLOWA 指令停止输出脉冲。
5．如果断开能流后 SD212/SD232/SD292/SD312/SD332/SD352/SD372/SD392 的值不为 0，需要手动清除。

6.8.11 LIN: 直线插补指令

					适用机型	MX1H	MX2H	MX2V	MX3H
	LIN		S1	S2				√	√
步长	16位：	······		······	相关元件	*			
	32位：LIN（连续）	5步		······					

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	WORD																●								
S2	WORD																							*	

* 该指令使用 C 元件时，只能使用 C0 ～ C199。

* 该指令的操作数 S2 使用常数时，范围：2 ≤ S2 ≤ 4

* 相关元件：SM29、SM78、SM80 ～ SM83、SM261 ～ SM272、SM288、SM298、SM318、SM328、SM338、SM348、SM358、SM368、SD50 ～ SD53、SD70 ～ SD81、SD200 ～ SD201、SD211、SD220 ～ SD221、SD231、SD280 ～ SD281、SD291、SD300 ～ SD301、SD311、SD320 ～ SD321、SD331、SD340 ～ SD341、SD351、SD360 ～ SD361、SD371、SD380 ～ SD381、SD391

指令说明

├──┤ ├──[LIN S1 S2]

S1：参数指定的 D 元件起始地址

S2：插补的轴数（2 ≤ S2 ≤ 4）

轴数设定后，设置相对应 D 元件参数，指令运行，相应轴运行到目标位置。

功能说明：

D 元件号	内容
(S1+1 , S1)	合成速度：初始速度 Fmin（Hz）
(S1+3 , S1+2)	合成速度：最高速度 Fmax（Hz）
S1+4	加 / 减速时间 T（ms）
(S1+6 , S1+5)	X 轴目标值（移动距离）
S1+7	X 轴脉冲输出端口（或为正负脉冲输出的正脉冲端口号或 AB 相输出的 A 相端口号）
(S1+9 , S1+8)	Y 轴目标值（移动距离）
S1+10	Y 轴脉冲输出端口（或为正负脉冲输出的正脉冲端口号或 AB 相输出的 A 相端口号）
(S1+12 , S1+11)	Z 轴目标值（移动距离）
S1+13	Z 轴脉冲输出端口（或为正负脉冲输出的正脉冲端口号或 AB 相输出的 A 相端口号）
(S1+15 , S1+14)	U 轴目标值（移动距离）
S1+16	U 轴脉冲输出端口（或为正负脉冲输出的正脉冲端口号或 AB 相输出的 A 相端口号）

配置码分配（10 进制）：

SD***	SD211	SD231	SD291	SD311	SD331	SD351	SD371	SD391
对应 Y 端口	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7

D 元件号	内容
SD***	动作模式及输出逻辑关系： Bit0：0- 绝对地址　1- 相对地址 Bit1：0-T 型加减速　1-S 型加减速 Bit5 ～ Bit4：00- 脉冲 + 方向　01- 正负脉冲　10- AB 脉冲

使用示例

1．上电第一周期，配置相关 D 元件参数，轴数选择为 4，则要配置 D 元件号 S1 ～ S1+16 的参数。

2．当 M0 为 ON 时，LIN 指令执行，根据配置的各轴目标值，相应轴运行到目标位置。

3．脉冲输出监控位 SM82、SM83、SM267 ～ SM272：如例所示，四个轴脉冲输出端口分别为 Y0、Y2、Y4、Y6，目标值分别为 1000、2000、3000、4000，指令运行中各轴脉冲输出监控位 SM82、SM267、SM269、SM271 为 ON，指令执行完成后为 OFF。

4．定位指令运行标志位 SM288、SM298、SM318、SM328、SM338、SM348、SM358、SM368：如例所示，指令运行中 SM288、SM318、SM338、SM358 为 ON，指令执行完成后为 OFF。

5．定位输出当前值 SD200 ～ SD201、SD220 ～ SD221、SD280 ～ SD281、SD300 ～ SD301、SD320 ～

SD321、SD340 ～ SD341、SD360 ～ SD361、SD380 ～ SD381：如例所示，指令运行中 SD200 ～ SD201、SD280 ～ SD281、SD320 ～ SD321、SD360 ～ SD361 不断变化。

6. 定位脉冲输出总数 SD50 ～ SD53、SD70 ～ SD81：如例所示，指令执行后 SD50、SD70、SD74、SD78 中可看到各轴指令执行后的目标值。

7. 指令执行结束 SM29：如例所示，指令执行完成后，指令执行结束标志位 SM29 置 ON，下次指令启动时变为 OFF。

8. 脉冲输出停止使能标志 SM80、SM81、SM261 ～ SM266：如例所示，M0 为 OFF，指令没执行时，SM80、SM261、SM263、SM265 任一置 ON，指令将不能启动。

9. 配置码 SD211、SD231、SD291、SD311、SD331、SD351、SD371、SD391：如例所示，若要重复执行指令，应选择相对地址模式，相应配置码 SD211、SD291、SD331、SD371 的 Bit0 应为 1，若要脉冲输出端口输出不同脉冲模式，则利用配置码的 Bit5 ～ Bit4 配合实现。

10. 插补完成中断：如果中断总标志及插补完成中断标志 SM78 打开，插补指令执行完成后，会执行插补完成中断。

注意事项

1. 无论轴数 S2 怎样变化，运动轴都是由 X 轴开始（例如，轴数为 2 时，是 X 轴、Y 轴运行，而不是 Y 轴、Z 轴、U 轴中的任意两轴运行）。

2. 操作数 S1 设置相关 D 参数的轴数要大于等于操作数 S2 的数目（例如，轴数为 2 时，D 元件 S1 ～ S1+10 的参数必须设置，D 元件 S1+11 ～ S1+16 设不设置均可；轴数为 4 时，D 元件 S1 ～ S1+16 的参数必须设置，如果设置 D 元件数≤ S1+16，则影响指令运行）。

3. 脉冲输出端口默认值问题。例如，脉冲输出端口为 Y0，输出模式为脉冲 / 方向时，与 Y0 配合输出的方向端口只能为 Y10；脉冲输出端口为 Y0，输出模式为脉冲 / 方向或 AB 相时，与 Y0 配合输出的另一脉冲输出端口只能为 Y1。详见模块系统组态。

4. 请注意不要同时对一个高速端口使用多个高速输出和定位指令。

6.8.12 CW：顺时针圆弧插补指令

	CW		S D1 D2	适用机型	MX1H	MX2H	MX2V	MX3H
							√	√
步长	16位： ----- ----- 32位： CW（连续）7步 -----			相关元件	*			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	WORD																●								
D1	BOOL		●																						
D2	BOOL		●																						

* 该指令使用 C 元件时，只能使用 C0 ～ C199。

* 相关元件：SM29、SM78、SM80 ～ SM83、SM261 ～ SM272、SM288、SM298、SM318、SM328、SM338、SM348、SM358、SM368、SD50 ～ SD53、SD70 ～ SD81、SD200 ～ SD201、SD211、SD220 ～ SD221、SD231、SD280 ～ SD281、SD291、SD300 ～ SD301、SD311、SD320 ～ SD321、SD331、SD340 ～ SD341、SD351、SD360 ～ SD361、SD371、SD380 ～ SD381、SD391

指令说明

├──┤ ├──[CW S D1 D2]

S：参数指定的 D 元件起始地址

D1：X 轴脉冲输出端口（X 轴正向脉冲输出端口或 AB 相脉冲的 A 相端口）

D2：Y 轴脉冲输出端口（Y 轴正向脉冲输出端口或 AB 相脉冲的 A 相端口）

指令三种指定运行方式：圆心指定、半径指定、经过位置指定。

圆心指定、半径指定：相应 D 元件及两轴脉冲输出端口设置后，运行指令，指令从起始位置以顺时针圆弧轨移动到目标位置。

经过位置指定：相应 D 元件及两轴脉冲输出端口设置后，运行指令，指令以圆弧轨迹从起始位置移动到经过位置，再移动到目标位置。模块既可顺时针运行也可逆时针运行，方向取决于三点的坐标位置。

功能说明：

D 元件号	内容
S ～ S+1	保留
(S1+3 , S+2)	合成速度 Fmax（Hz）
S+4	保留
(S1+6 , S+5)	X 轴目标值（移动距离）
(S1+8) , S+7	Y 轴目标值（移动距离）
(S1+10) , S+9	X 轴圆心位置 / 当用半径指定时为半径 / 经过位置
(S1+12 , S+11)	Y 轴圆心位置 / 经过位置

配置码分配（10 进制）：

SD***	SD211	SD231	SD291	SD311	SD331	SD351	SD371	SD391
对应 Y 端口	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7

D 元件号	内容
SD***	动作模式及输出逻辑关系： Bit0：0- 绝对地址 1- 相对地址 Bit1：0-T 型加减速 1-S 型加减速 Bit5 ～ Bit4：00- 脉冲 + 方向 01- 正负脉冲 10-AB 脉冲 Bit9 ～ Bit8：00- 半径指定 01- 圆心指定 10- 经过位置指定位置指定

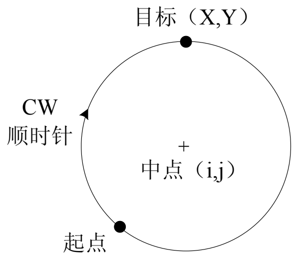


图1 圆心指定

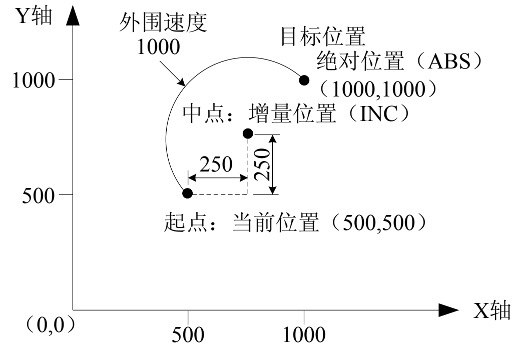


图2 圆心指定

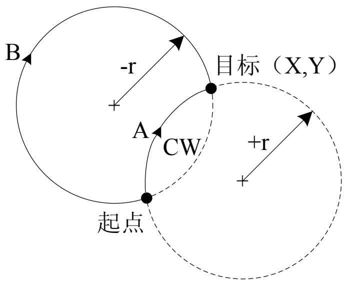


图3 半径指定

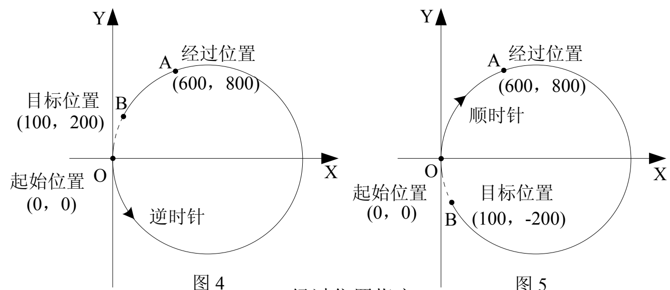
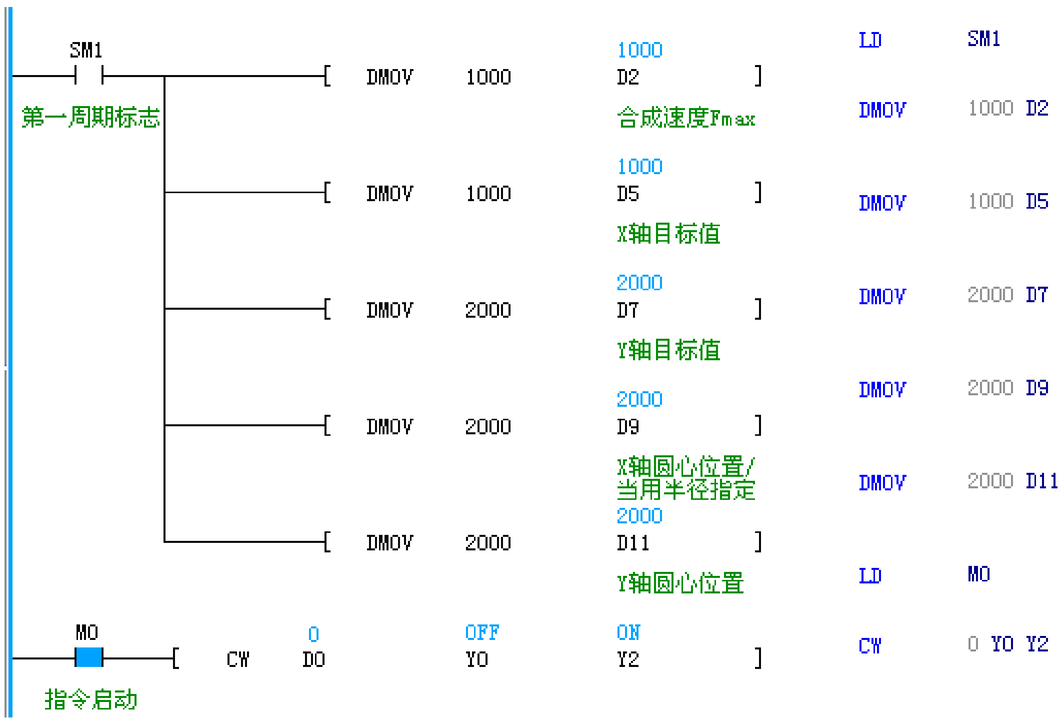


图 4 经过位置指定 图 5

使用示例



1. 上电第一周期，配置相关 D 元件参数，X 轴、Y 轴脉冲输出端口分别 Y0、Y2，则要配置 D 元件号 S+2 ～ S+11。
2. 当 M0 为 ON 时，CW 指令执行，根据配置的两轴目标值，由起始位置以顺时针圆弧轨迹运行到目标位置。
3. 脉冲输出监控位 SM82、SM83、SM267 ～ SM272：如例所示，两个轴脉冲输出端口分别为 Y0、Y2，

目标值分别为 1000、2000，指令运行中两轴脉冲输出监控位 SM82、SM267 为 ON，指令执行完成后为 OFF。

4．定位指令运行标志位 SM288、SM298、SM318、SM328、SM338、SM348、SM358、SM368：如例所示，指令运行中 SM288、SM318 为 ON，指令执行完成后为 OFF。

5．定位输出当前值 SD200 ～ SD201、SD220 ～ SD221、SD280 ～ SD281、SD300 ～ SD301、SD320 ～ SD321、SD340 ～ SD341、SD360 ～ SD361、SD380 ～ SD381：如例所示，指令运行中 SD200 ～ SD201、SD280 ～ SD281 数值不断变化。

6．定位脉冲输出总数 SD50 ～ SD53、SD70 ～ SD81：如例所示，指令执行后 SD50、SD70 中可看到各轴指令执行后的目标值（由于其运行轨迹为圆弧，其数值可能大于等于目标值）。

7．指令执行结束 SM29：如例所示，指令执行完成后，指令执行结束标志位 SM29 置 ON，下次指令启动时变为 OFF。

8．脉冲输出停止使能标志 SM80、SM81、SM261 ～ SM266：如例所示，M0 为 OFF，指令没有执行时，SM80、SM261 任一置 ON，指令将不能启动。

9．配置码 SD211、SD231、SD291、SD311、SD331、SD351、SD371、SD391：如例所示，若要重复执行指令，应选择相对地址模式，相应配置码 SD211、SD291 的 Bit0 应为 1；若要脉冲输出端口输出不同脉冲模式，则利用配置码的 Bit5 ～ Bit4 配合实现；例中所示，运行模式为半径指定模式，若要选择圆心指定模式，相应配置码 SD211、SD291 的 Bit8 应为 1。

10．插补完成中断：如果中断总标志及插补完成中断标志 SM78 打开，插补指令执行完成后，会执行插补完成中断。

注意事项

1．运行模式为指定半径时，若起始值为（X_1,Y_1），目标值为（X_2,Y_2），则半径指定 S1+9 的数值 D 应满足 $D \geqslant \frac{\sqrt{(X2-X1)^2+(Y2-Y1)^2}}{2}$。例如，起始值为（0，0），目标值为（300，400），则半径指定 S1+9 的值应大于等于 250。

2．运行模式为指定半径时，半径为正；则运行轨迹为劣弧，半径为负，则运行轨迹为优弧。（如图 3、4）

3．运行模式为指定圆心、动作模式选择为绝对模式时，假如没有设置目标值（S+5、S+7 为 0）或者起始地址等于目标地址，则运行轨迹为整圆。（如图 1）

4．运行模式为指定圆心，动作模式选择为相对模式或绝对模式条件下，起始地址不等于目标值时，运行轨迹为圆弧。（如图 2）

5．运行模式为经过位置，设置好起始位置坐标、经过位置坐标、目标位置坐标后，根据定义的三点坐标位置，模块顺时针运行或逆时针运行。如上图 4 三点坐标位置所示，模块为逆时针运行；如上图 5 三点坐标位置所示，模块为顺时针运行。

6.8.13 CCW：逆时针圆弧插补指令

	CCW		S D1 D2	适用机型	MX1H	MX2H	MX2V	MX3H
							√	√
步长	16位： ----- -----			相关元件	*			
	32位： CCW （连续） 7步 -----							

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	WORD																●								
D1	BOOL		●																						
D2	BOOL		●																						

* 该指令使用 C 元件时，只能使用 C0 ～ C199。

* 相关元件：SM29、SM78、SM80 ～ SM83、SM261 ～ SM272、SM288、SM298、SM318、SM328、SM338、SM348、SM358、SM368、SD50 ～ SD53、SD70 ～ SD81、SD200 ～ SD201、SD211、SD220 ～ SD221、SD231、SD280 ～ SD281、SD291、SD300 ～ SD301、SD311、SD320 ～ SD321、SD331、SD340 ～ SD341、SD351、SD360 ～ SD361、SD371、SD380 ～ SD381、SD391

指令说明

├──┤├──[CCW S D1 D2]

S：参数指定的 D 元件起始地址

D1：X 轴脉冲输出端口（X 轴正向脉冲输出端口或 AB 相脉冲的 A 相端口）

D2：Y 轴脉冲输出端口（Y 轴正向脉冲输出端口或 AB 相脉冲的 A 相端口）

指令三种指定运行方式：圆心指定、半径指定、经过位置指定。

圆心指定、半径指定：相应 D 元件及两轴脉冲输出端口设置后，运行指令，指令从起始位置以顺时针圆弧轨迹移动到目标位置。

经过位置指定：相应 D 元件及两轴脉冲输出端口设置后，运行指令，指令以圆弧轨迹从起始位置移动到经过位置，再移动到目标位置。模块既可顺时针运行也可逆时针运行，方向取决于三点的坐标位置。

功能说明：

D 元件号	内容
S ～ S+1	保留
(S1+3 , S+2)	合成速度 Fmax（Hz）
S+4	保留
(S1+6 , S+5)	X 轴目标值（移动距离）
(S1+8 , S+7)	Y 轴目标值（移动距离）
(S1+10 , S+9)	X 轴圆心位置 / 当用半径指定时为半径 / 经过位置
(S1+12 , S+11)	Y 轴圆心位置 / 经过位置

配置码分配（10 进制）：

SD***	SD211	SD231	SD291	SD311	SD331	SD351	SD371	SD391
对应 Y 端口	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7

D 元件号	内容
SD***	动作模式及输出逻辑关系： Bit0：0- 绝对地址 1- 相对地址 Bit1：0-T 型加减速 1-S 型加减速 Bit5 ～ Bit4：00- 脉冲＋方向 01- 正负脉冲 10-AB 脉冲 Bit9 ～ Bit8：00- 半径指定 01- 圆心指定 10- 经过位置指定

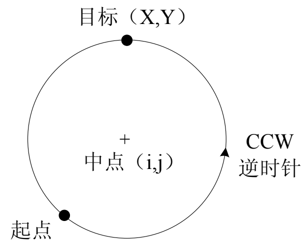


图1 圆心指定

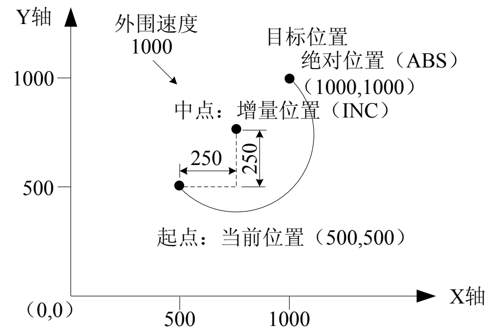


图2 圆心指定

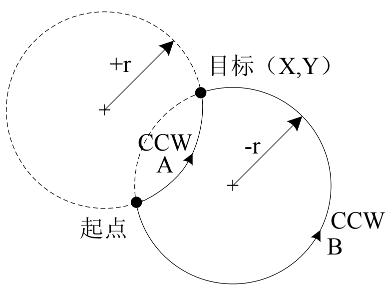


图3 半径指定

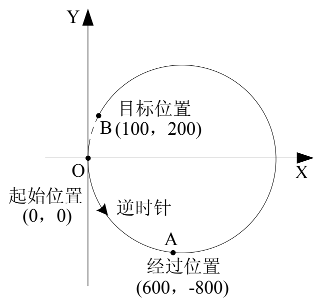


图4

经过位置指定

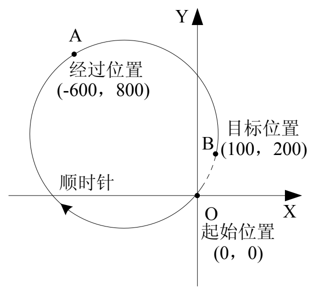
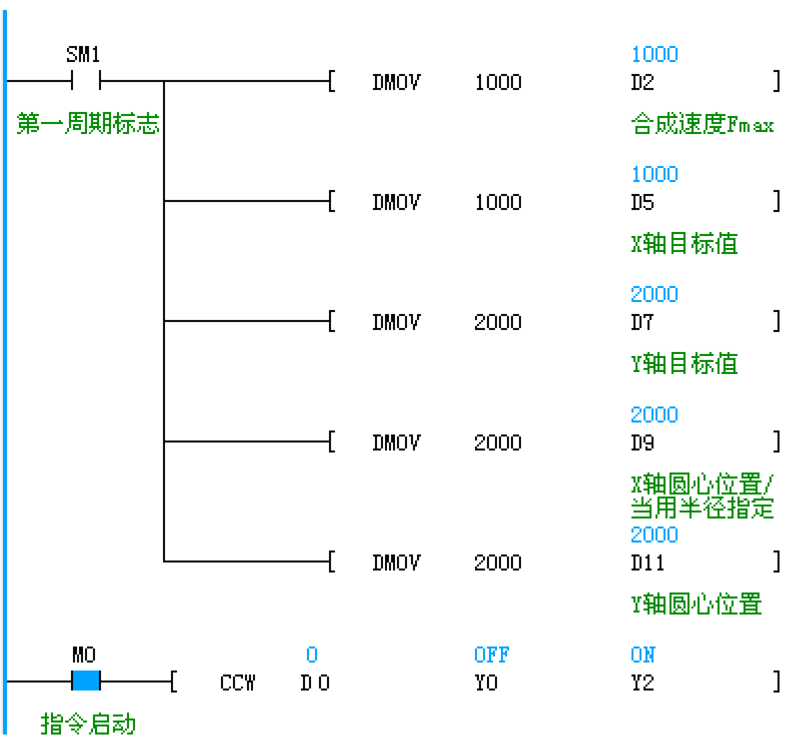


图5

使用示例




```
LD    SM1
DMOV  1000 D2
DMOV  1000 D5
DMOV  2000 D7
DMOV  2000 D9
DMOV  2000 D11
LD    M0
CCW   D0 Y0 Y2
```

1．上电第一周期，配置相关 D 元件参数，X 轴、Y 轴脉冲输出端口分别 Y0、Y2，则要配置 D 元件号 S1+2 ～ S1+11。
2．当 M0 为 ON 时，CCW 指令执行，根据配置的两轴目标值，由起始位置以逆时针圆弧轨迹运行到目标位置。
3．脉冲输出监控位 SM82、SM83、SM267 ～ SM272：如例所示，两个轴脉冲输出端口分别为 Y0、Y2，目标值分别为 1000、2000，指令运行中两轴脉冲输出监控位 SM82、SM267 为 ON，指令执行完

成后为 OFF。

4. 定位指令运行标志位 SM288、SM298、SM318、SM328、SM338、SM348、SM358、SM368：如例所示，指令运行中 SM288、SM318 为 ON，指令执行完成后为 OFF。
5. 定位输出当前值 SD200 ～ SD201、SD220 ～ SD221、SD280 ～ SD281、SD300 ～ SD301、SD320 ～ SD321、SD340 ～ SD341、SD360 ～ SD361、SD380 ～ SD381：如例所示，指令运行中 SD200 ～ SD201、SD280 ～ SD281 数值不断变化。
6. 定位脉冲输出总数 SD50 ～ SD53、SD70 ～ SD81：如例所示，指令执行后 SD50、SD70 中可看到各轴指令执行后的目标值（由于其运行轨迹为圆弧，其数值可能大于等于目标值）。
7. 指令执行结束 SM29：如例所示，指令执行完成后，指令执行结束标志位 SM29 置 ON，下次指令启动时变为 OFF。
8. 脉冲输出停止使能标志 SM80、SM81、SM261 ～ SM266：如例所示，M0 为 OFF，指令没有执行时，SM80、SM261 任一置 ON，指令将不能启动。
9. 配置码 SD211、SD231、SD291、SD311、SD331、SD351、SD371、SD391：如例所示，若要重复执行指令，应选择相对地址模式，相应配置码 SD211、SD291 的 Bit0 应为 1；若要脉冲输出端口输出不同脉冲模式，则利用配置码的 Bit5 ～ Bit4 配合实现；例中所示，运行模式为半径指定模式，若要选择圆心指定模式，相应配置码 SD211、SD291 的 Bit8 应为 1。
10. 插补完成中断：如果中断总标志及插补完成中断标志 SM78 打开，插补指令执行完成后，会执行插补完成中断。

注意事项

1. 运行模式为指定半径时，若起始值为（X_1,Y_1），目标值为（X_2,Y_2），则半径指定 S1+9 的数值 D 应满足 D ≥ $\frac{\sqrt{(X2-X1)^2+(Y2-Y1)^2}}{2}$。例如，起始值为（0，0），目标值为（300，400），则半径指定 S1+9 的值应大于等于 250。
2. 运行模式为指定半径时，半径为正；则运行轨迹为劣弧，半径为负，则运行轨迹为优弧。（如图 3、4）
3. 运行模式为指定圆心、动作模式选择为绝对模式时，假如没有设置目标值（S+5、S+7 为 0）或者起始地址等于目标地址，则运行轨迹为整圆。（如图 1）
4. 运行模式为指定圆心，动作模式选择为相对模式或绝对模式条件下，起始地址不等于目标值时，运行轨迹为圆弧。（如图 2）
5. 运行模式为经过位置，设置好起始位置坐标、经过位置坐标、目标位置坐标后，根据定义的三点坐标位置，模块顺时针运行或逆时针运行。如上图 4 三点坐标位置所示，模块为逆时针运行；如上图 5 三点坐标位置所示，模块为顺时针运行。

6.9 外设指令

6.9.1 FROM : 特殊模块缓冲寄存器字读指令

	FROM	P	S1 S2 D S3	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√	√
步长	16位：FROM (连续) 9步 FROMP (脉冲) 10步 32位： ----- -----			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT																							•	
S2	INT																							•	
D	INT																•				•		•		
S3	INT																							•	

指令说明

├──┤ ├──[FROM S1 S2 D S3]

S1：访问的特殊模块编号（0 ≤ S1 ≤ 7）。

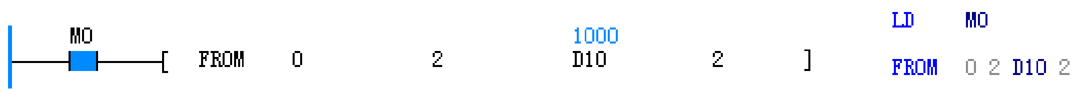
S2：要读取的特殊模块 BFM 起始地址（0 ≤ S2 ≤ 199）。

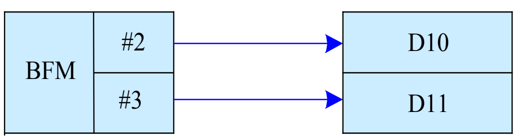
D：从特殊模块读数到数据存储单元。

S3：要读取的特殊模块的 BFM 连续缓冲寄存器（单字）数量（1 ≤ S3 ≤ 200 − S2）。

从指定特殊模块（编号为 S1）的指定 BFM 缓存区（地址为 S2）开始连续读取 S3 个字数据，依次存放到 D 起始的 S3 个字元件中。

使用示例





M0 为 ON 时，执行 FROM 指令，从 0 号特殊模块的缓冲寄存器（BFM）#2，#3 中读取 2 个字数据，读到的数据分别保存在 D10 和 D11 中。

注意事项

1. 如果连接多个特殊模块时，可编程控制器缓冲寄存器运行时的初始化时间会变长，运算时间会延长。另外执行多个 FROM 指令或传送多个缓冲寄存器数据时，运算时间也会延长。
2. 关于特殊模块的连接方法，可连接个数及输入输出编号等使用方法，请参见可编程控制器硬件手册和各模块附带的专用手册。
3. 对特殊模块的寄存器进行读或写操作时，使用系统组态或 FROM － TO 指令都可以实现，但是当两者都使用时，上电第一周期按照系统组态中配置执行，之后便按 FROM － TO 指令的设置处理。

6.9.2 TO : 特殊模块缓冲寄存器字写指令

	TO	P	S1 S2 S3 S4	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√	√
步长	16位：TO (连续) 9步 32位： -----		TOP (脉冲) 10步 -----	相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT																							●	
S2	INT																							●	
S3	INT																●				●		●		
S4	INT																							●	

指令说明

S1：访问特殊模块的编号（0 ≤ S1 ≤ 7）。

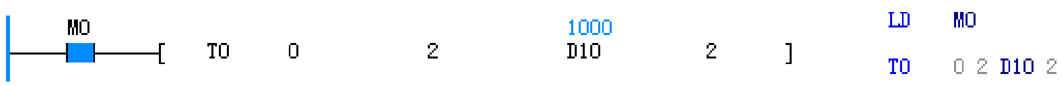
S2：要写入的特殊模块 BFM 起始地址（0 ≤ S2 ≤ 199）

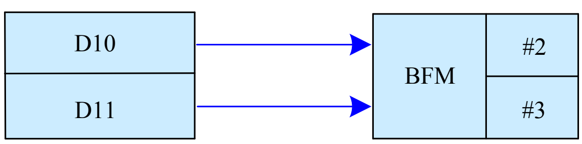
S3：要写入特殊模块的数据

S4：要写入特殊模块的 BFM 连续缓冲寄存器（单字）数量（1 ≤ S4 ≤ 200 － S2）

将 S3 起始的 S4 个连续字元件数据，依次写入指定特殊模块（编号为 S1）的指定 BFM 缓存器（地址 S2）起始的 S4 个连续字元件中。

使用示例





M0 为 ON 时，执行 TO 指令，将 D10、D11 的内容写到 0 号特殊模块的缓冲寄存器（BFM）#2，#3 中。

注意事项

1. 如果连接多个特殊模块时，可编程控制器缓冲寄存器运行时的初始化时间会变长，运算时间会延长。另外执行多个 TO 指令或传送多个缓冲寄存器数据时，运算时间也会延长。
2. 关于特殊模块的连接方法，可连接个数及输入输出编号等使用方法，请参见可编程控制器硬件手册和各模块附带的专用手册。
3. 对特殊模块的寄存器进行读或写操作时，使用系统组态或 FROM － TO 指令都可以实现，但是当两者都使用时，上电第一周期按照系统组态中配置执行，之后便按 FROM － TO 指令的设置处理。

6.9.3 REFF : 设置输入滤波常数指令

	REFF	P	S	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：REFF （连续） 3步 32位： -----		REFFP （脉冲） 4步 -----	相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	WORD										●	●	●	●	●	●	●	●	*	●	●	●	●	●	●

* 该指令使用 C 元件时，只能使用 C0 ～ C199。

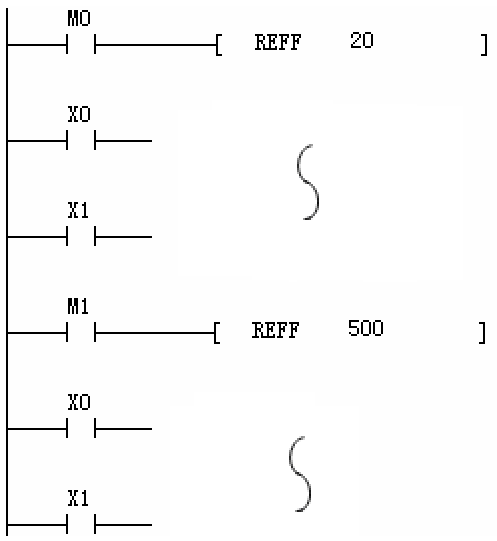
指令说明

├──┤ ├──[REFF S]

S：输入滤波常数。可设定范围：0 ～ 512（单位 0.1ms），实际有效的数据为 0.8 ms、1.6 ms、3.2 ms、6.4 ms、12.8 ms、25.6 ms、51.2ms。当参数小于 16 时，按 0.8ms 处理；当参数小于 32 大于 16 时，按 1.6ms 处理；以此类推，当参数大于 512 时，按 51.2ms 处理。

1．为了提高 PLC 输入端子的抗干扰能力，在输入端子内部都设有滤波器，滤波时间默认为 12.8ms。
2．为了有效去掉短暂的输入干扰信号，PLC 将输入端子的数字滤波时间常数设为可调。

使用示例



M0 为 ON 且 M1 为 OFF 时，滤波时间按 1.6ms 处理（16 ＜ 20 ＜ 32）。
M1 为 ON 且 M0 为 OFF 时，滤波时间按 25.6ms 处理（256 ＜ 500 ＜ 512）。

注意事项

1．输入滤波常数仅对用做普通输入的端口有效，对用做高速输入的端口无效。
2．两条 REFF 指令同时有效时，就有可能产生错乱。
3．当系统组态和 REFF 指令设置的滤波时间不相同时，只在上电第一周期按照系统组态中设置的滤波时间执行，之后便按 REFF 指令设置的滤波时间处理。

6.9.4 REF : 刷新 I/O 镜像指令

	REF	P	D S	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：REF (连续) 5步 32位： -----		REFP (脉冲) 6步 -----	相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
D	BOOL	●	●																						
S	INT																							●	

指令说明

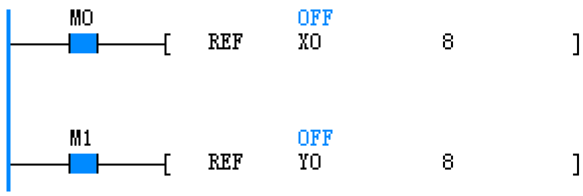
├──┤ ├──[REF D S]

D：要刷新的起始 X/Y 软元件。指定起始软元件号为 8 的整数倍，如 X0、X10、X20…或 Y0、Y10、Y20…，最低位为 0。

S：要刷新的端口数量。刷新点数应为 8，16，……（8 的倍数，除此以外的数值是错误的）。

通常 PLC 的输入和输出都是在用户程序结束后执行，在运算过程中，如果需要读取最新的输入状态或希望立即刷新输出状态时，可以使用该指令。

使用示例




```
LD   M0
REF  X0 8
LD   M1
REF  Y0 8
```

1. M0 为 ON 时，执行 REF 指令，PLC 会立即读取 X0 ～ X7 的输入点状态，输入信号立即刷新，并且没有输入延迟。
2. M1 为 ON 时，执行 REF 指令，Y0 ～ Y7 这 8 点输出信号实时被送至输出端，输出信号立即刷新，不必等到程序执行完毕。

注意事项

1. 在 FOR － NEXT 指令之间或 CJ 指令之间，一般使用 REF 指令。
2. 处理外部中断时，在中断子程序中可使用 REF 指令，获取最新的输入信息并且及时输出运算结果。
3. 输出刷新中的输出接点将在输出应答时间后动作。继电器输出型的应答时间约为 20ms；晶体管输出型时，Y0 ～ Y1 应答速度小于 5us/10mA 以上，其他小于 0.5ms/100mA 以上。

6.9.5 PR : ASCII 打印指令

	PR		S D	适用机型	MX1H	MX2H	MX2V	MX3H
							√	√
步长	16位: PR (连续) 5步 32位: -----		----- -----	相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	WORD																●	●	*	●			●		
D	BOOL		●																						

* 该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

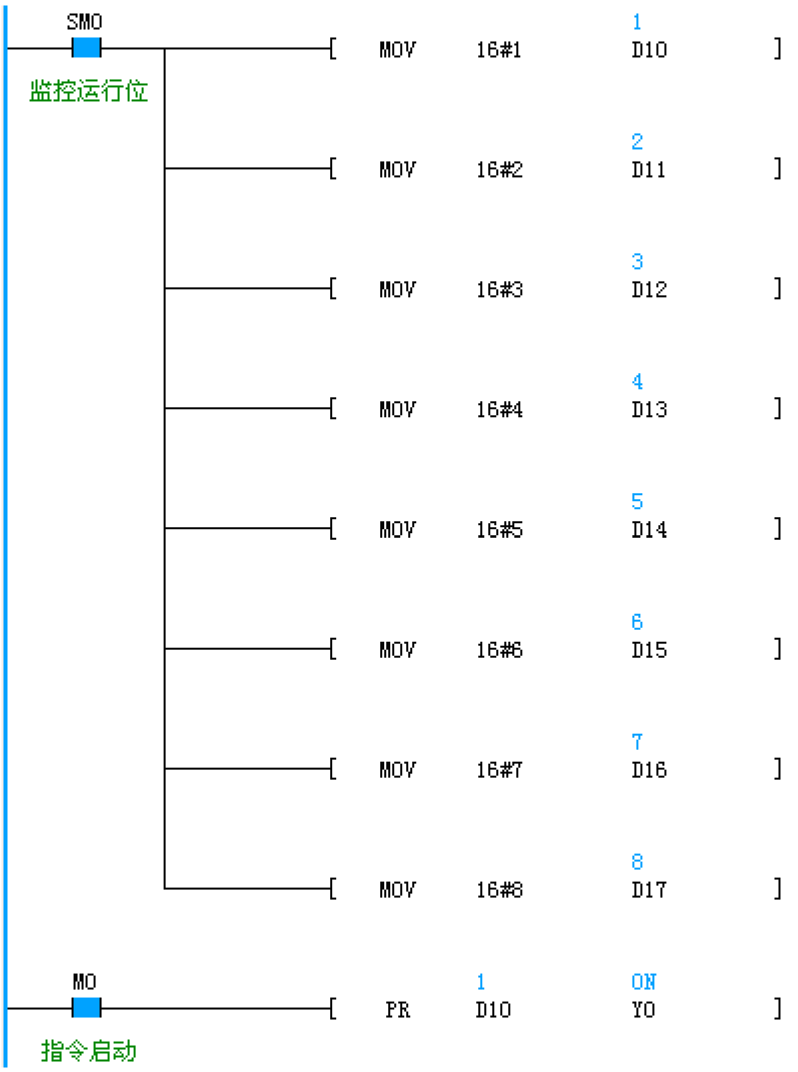
|—| |—[PR S D]

S：保存 ASCII 码数据的软元件起始编号

D：输出 ASCII 码数据的起始 Y 编号

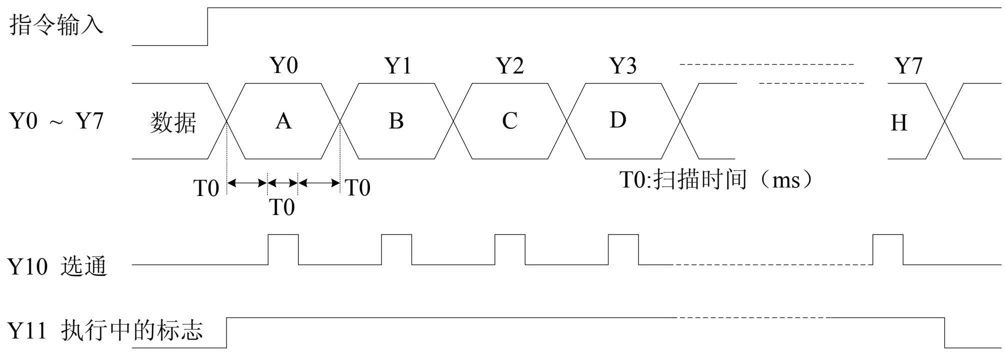
将 S ～ S+7 的低 8 位中保存的 ASCII 码，按时分方式逐个输出到 D ～ D+7 中。

使用示例




```
LD   SM0
MOV  16#1 D10
MOV  16#2 D11
MOV  16#3 D12
MOV  16#4 D13
MOV  16#5 D14
MOV  16#6 D15
MOV  16#7 D16
MOV  16#8 D17
LD   M0
PR   D10 Y0
```

M0 为 ON 执行指令，Y11 作为指令运行标志一直为 ON，Y0 ～ Y7 按时分方式输出 D10 ～ D17 的数值，随着每次时分输出，选通信号 Y10 置 ON，输出完 D17 的数值后，指令执行结束，指令运行标志 Y11 变为 OFF。



注意事项

1．指令执行后，输出 ASCII 码的顺序是从 S=A 开始，到 S+7=H 结束，每次发送 8 个字节。
2．该指令与扫描时间同步执行。扫描时间较短时，可以使用恒定时间扫描以加大扫描周期；扫描时间较长时，可以使用定时中断以减小扫描周期。

6.9.6 TKY：16 位数字键输入指令

D	TKY		S D1 D2	适用机型	MX1H	MX2H	MX2V	MX3H
							√	√
步长	16位： TKY (连续) 7步 ----- 32位：DTKY (连续) 8步 -----			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	BOOL	●	●	●	●			●	●																
D1	(D)INT											●	●	●	●	●	●	●	*	*	●	*	●		
D2	BOOL		●	●	●			●																	

* 16 位指令的 D1 使用 C 元件时，只能使用 C0 ～ C199。
32 位指令的 D1 使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z 元件。

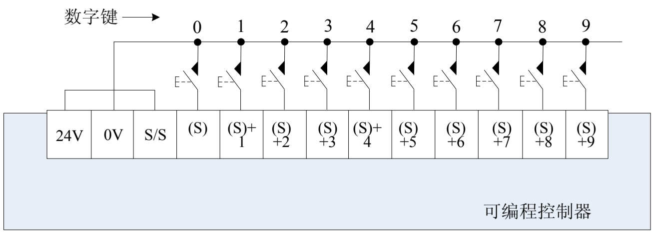
指令说明

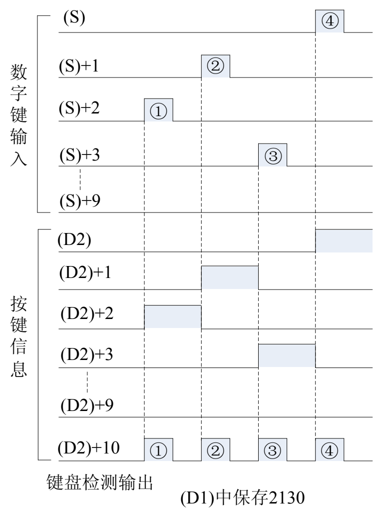
├──┤ ├──[TKY S D1 D2]

S：输入数字键的起始软元件（占用 10 个）

D1：保存数据的字软元件编号

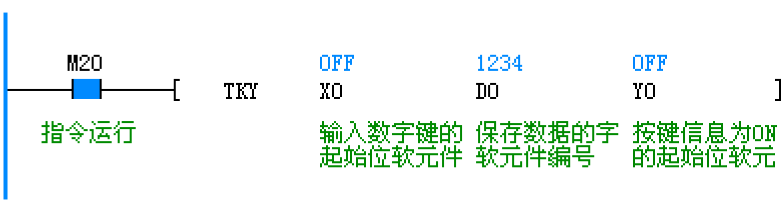
D2：按键信息为 ON 的起始位软元件编号（占用 11 个）





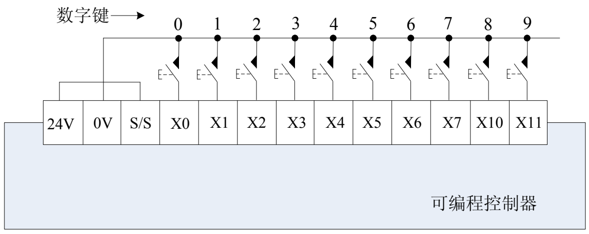
如上图所示，位软元件 S+2、S+1、S+3、S 依次输入为 ON，他们对应的数字为 2、1、3、0，数据保存单元 D1 会按输入顺序保存 2130，按键信息 D2+2、D2+1、D2+3、D2 会根据输入信息依次输出为 ON。位软元件 S ～ S+9 对应数字 0 ～ 9，位软元件为 ON 时，将其转化为相对应的数字，按输入顺序保存在 D1 中（超过 9,999，从高位溢出），D2 ～ D2+9 输出检测到的相对应的输入信息；任一 S 输入为 ON，D2+10 作为输入信息检测，输出为 ON。

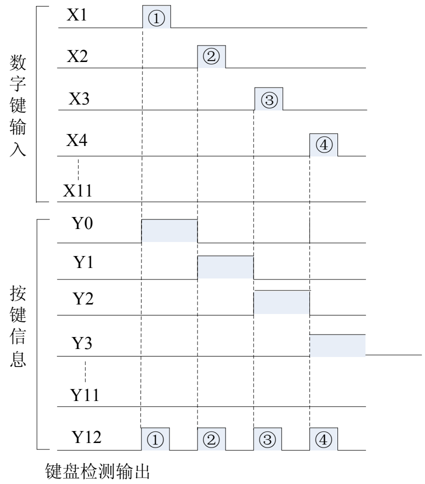
使用示例



当 M20 为 ON 时，X1 ～ X4 依次输入为 ON，其对应的数字 1 ～ 4 按输入顺序保存在 D0 中（若再输入 X5 则 1 由 D0 溢出，D0 保存数据 2,345），Y1 ～ Y4 作为相对应的信息输出单元依次输出为 ON；Y12

作为输入信息检测，当 X1 ～ X4 依次输入为 ON，Y12 分别输出为 ON。





注意事项

1．输入的数值，以十进制保存。

2．同时多个输入为 ON，只有先输入的有效。

3．指令前触点为 OFF，D1 保存的数据不变，但 D2 ～ D2+10 都变为 OFF。

4．指令占用从 S 开始的 10 个软元件，即使指令不运行，所占用的 S 软元件不能用作其他用途；指令占用 D2 开始的 11 个软元件，请注意不要与程序中使用的软元件重复。

5．TKY 指令或 DTKY 指令，程序中只能使用其中一个。

6.9.7 HKY：16 进制输入指令

D	HKY		S D1 D2 D3	适用机型	MX1H	MX2H	MX2V	MX3H
							√	√
步长	16位： HKY (连续) 9步 -----			相关元件	SM167			
	32位： DHKY (连续) 10步 -----							

操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z				Un.b
S	BOOL	●																								
D1	BOOL		●																							
D2	(D)INT											●	●		●		●		*	*	●	*	●			●
D3	BOOL		●	●	●			●																		

* 16 位指令的 D2 使用 C 元件时，只能使用 C0 ～ C199。
32 位指令的 D2 使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z 元件。

指令说明

HKY S D1 D2 D3

S：16 进制输入的起始位软元件（X）编号（占用 4 点）
D1：输出的起始软元件（Y）编号（占用 4 点）
D2：保存 16 进制输入数值的软元件编号
D3：16 进制输入（A ～ F）为 ON 的起始位软元件编号（占用 8 点）

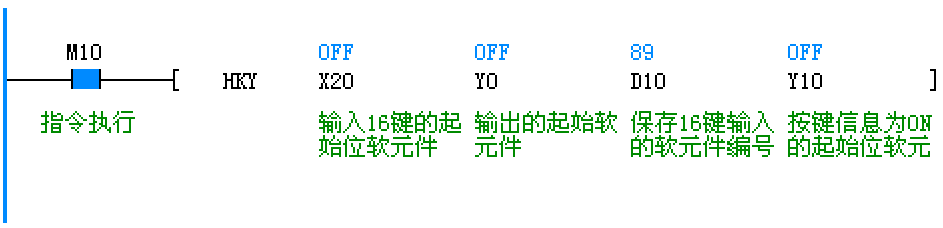
操作数D1 / 操作数S	D1	D1+1	D1+2	D1+3
S	0	4	8	C
S+1	1	5	9	D
S+2	2	6	A	E
S+3	3	7	B	F

扫描位软元件 S ～ S+3（S ～ S+3 对应数字 0 ～ 3）和位软元件 D1 ～ D1+3（D1 ～ D1+3 对应数字 0、4、8、12），将其转化为相对应的数值，按输入顺序保存在 D2 中（超过 9,999，从高位溢出），当输入数值范围为 0 ～ 9 时，任意一个数字输入，D3+7 输出为 ON；当输入数值范围为 A ～ F 时，任意一个数值输入，D3+6 输出为 ON，并且 D3 ～ D3+5 对应输出为 ON。

扩展功能标志位：SM167

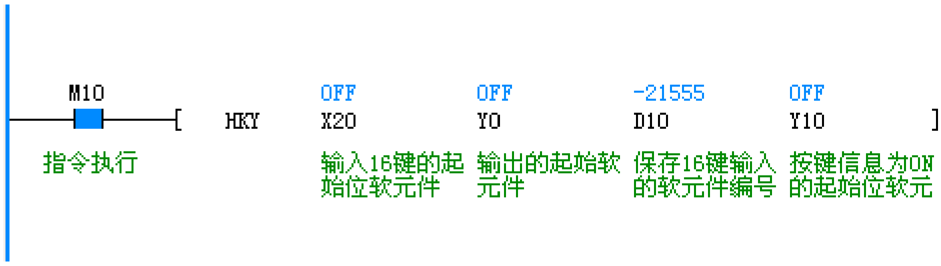
扩展功能标志位 SM167 为 OFF 时，16 位指令 HKY 操作数 D2 最大为 9,999，32 位指令 DHKY 操作数 D2 最大为 99,999,999，当输入为 A ～ F 时，寄存器不能显示，只能通过 D3 ～ D3+5 显示；扩展功能标志位 SM167 为 ON 时，16 位指令 HKY 操作数 D2 最大为 F,FFF，32 位指令 DHKY 操作数 D2 最大为 FF,FFF,FFF。

使用示例



指令执行，扩展功能标志位 SM167 为 OFF 时，依次输入 8、9、10、11（由上表得知，Y2 为 ON 的同时

X20 ～ X23 依次为 ON，D2 会按数值输入顺序保存 89，同时 Y10、Y11 依次为 ON）。其中输入数值 8、9 时，Y17 输出为 ON；输入数值为 10 时，Y16 和 Y10 输出为 ON；输入数值为 11 时，Y16 和 Y11 输出为 ON。



指令执行，扩展功能标志位 SM167 为 ON 时，依次输入 a、b、c、d（由上表得知，Y2 为 ON 的同时 X20 ～ X23 依次为 ON，Y3 为 ON 的同时 X20 ～ X21 依次为 ON），D2 会按数值输入顺序保存 abcd（梯形图中以十进制保存），Y16 和 Y10、Y11、Y12、Y13 依次为 ON。

注意事项

1. 指令的次数有限制，HKY 指令或者 DHKY 指令，在程序中可以使用其中 1 个，要使用变址修饰功能编程。
2. 同时按下多个键时，先按下的键有效。
3. 指令前的触点即使为 OFF，D2 的内容也不改变，但是 D3~D3+7 为止都变为 OFF。
4. 指令占用从 S 开始的 10 个软元件，即使指令不运行，所占用的 S 软元件不能用作其他用途；指令占用 D2 开始的 11 个软元件，请注意不要与程序中使用的软元件重复。
5. HKY、DHKY 指令与可编程控制器的运算周期同步执行，完成一系列的键盘扫描，需要 8 个运算周期的时间，为了防止因键盘输入的滤波器延迟而导致的读取遗漏，请灵活使用“恒定扫描周期”和“定时器中断”功能。

6.9.8 DSW：数字开关指令

	DSW		S D1 D2 D3	适应机型	MX1H	MX2H	MX2V	MX3H
							√	√
步长	16位：DSW (连续) 9步		-----	相关元件	SM29			
	32位： -----		-----					

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	BOOL	●																							
D1	BOOL		●																						
D2	INT																●		*	●	●	●	●		
D3	INT																							●	

* 16 位指令的 D2 使用 C 元件时，只能使用 C0 ～ C199。

* 操作数 n 使用常数时，范围为：n =1 或是 2。

指令说明

├──┤ ├──[DSW S D1 D2 D3]

S：连接数字开关的起始软元件（X）编号（占用 4*n 点）

D1：输出选通信号的起始软元件（Y）编号（占用 4 点）

D2：保存数字开关的数值的软元件编号（占用 n 点）

D3：数字开关的组数（4 位数 /1 组）[n=1 或是 2]

对 S 中连接的数字开关的值执行时分处理（通过 100ms 间隔的输出信号，从第 1 位数开始依次输入），并保存在 D2 中。

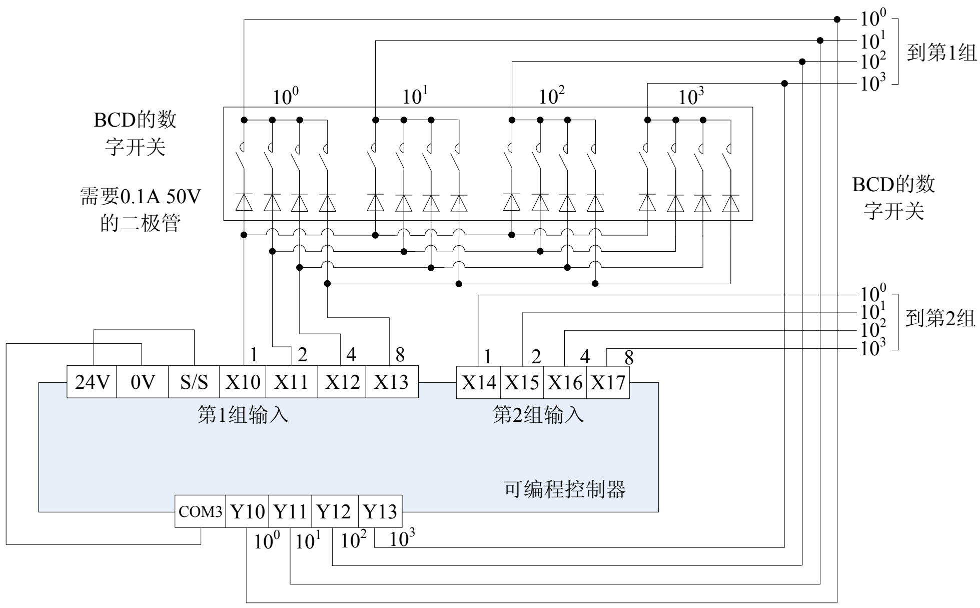
数字开关 D3=1 时

通过选通信号 D1 ～ D1+3，依次读取 S ～ S+3 中连接的 BCD4 位数的数字开关，并且将其值作为十进制保存到 D2 中。

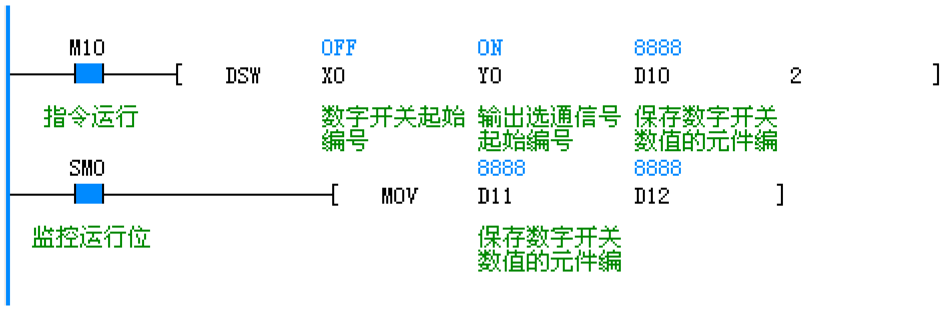
数字开关 D3=2 时

通过选通信号 D1 ～ D1+3，依次读取 S ～ S+3 中连接的 BCD4 位数的数字开关，并且将其值作为十进制保存到 D2 中；通过选通信号 S ～ S+3，依次读取 D1+4 ～ D1+7 中连接的 BCD4 位数的数字开关，并且将其值作为十进制数保存到 D2+1 中。

接线图如下：



使用示例



数字开关 n=2 时，M10 为 ON 指令执行，Y0 ～ Y3 以 100ms 间隔依次为 ON 输出信号，通过选通信号 Y0 ～ Y3 依次读取 X0 ～ X3 连接的 BCD4 位数的数字开关，并且将其值保存到 D10 中，通过选通信号 Y0 ～ Y3 依次读取 X4 ～ X7 连接的 BCD4 位数的数字开关，并且将其值保存到 D11 中。

注意事项

1．保存数据 D2 范围为 0 ～ 9,999，输入信号 OFF 时，其保存的数据会被清零，D2 中的内容不改变，但是 D1 ～ D1+3 为止都变为 OFF。
2．Y0 ～ Y3 一个循环输出后，SM29 置 ON，下次指令执行时，SM29 变为 OFF。
3．数字开关组数 n=1 时，操作数 S 占用 4 点，操作数 D2 占用 1 个；数字开关组数 n=2 时，操作数占用 8 点，作数 D2 占用 2 个。
4．为了能够连续的读取数字开关的值，请务必使用晶体管输出型的可编程控制器。
5．请使用 BCD 输出型的数字开关。

6.9.9 EROMWR : EEPROM 写指令

	EROMWR	P	S D	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√	√
步长	16位：EROMWR （连续） 5步 EROMWRP （脉冲） 6步 32位： ----- -----			相关元件				

操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z				Un.b
S	WORD																*							●		
D	INT										●	●	●	●	●	●	●	●	*	●	●	●		●	●	

指令说明

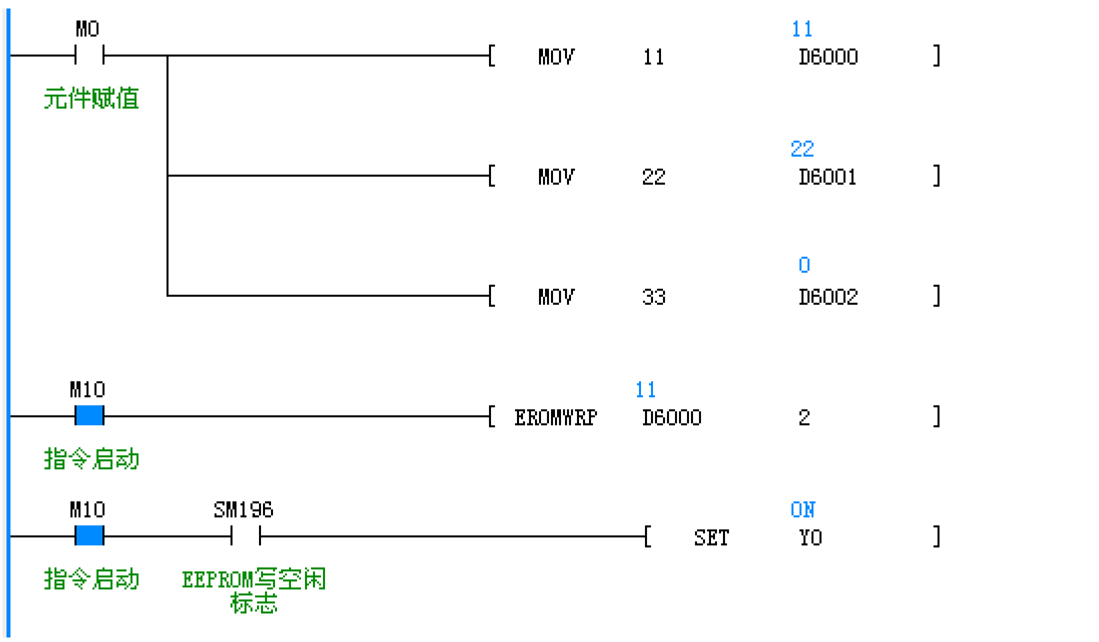
├─┤ ├─[EROMWR S D]

S：写入元件的起始地址，只能为 D6000 ～ D6999。

D：写入元件的个数，0≤D≤1000，S+D<D7000。

1. 通常 PLC 的一部分数据可以通过掉电保持功能保存，在运算过程中，如果需要将中间数据保存至 EEPROM 时，可以使用该指令。
2. EROMWR 为脉冲执行型指令，否则 PLC 报错。
3. 不能同时执行两条该指令，当 SM196（EEPROM 写空闲标志）为 OFF 时表示 EEPROM 写操作空闲，为 ON 时表示有 EROMWR 操作在执行。

使用示例



M10 为 ON 时，EROMWR 指令上升沿执行一次，D6000、D6001 数据掉电保存，指令执行期间 SM196 为 ON。

注意事项

EROMWR 指令运行期间，如果 PLC 断电，此次写区域的值是不可预知的。

6.10 实时时钟指令

6.10.1 TRD : 实时时钟读指令

	TRD		D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：TRD (连续) 3步 ----- 32位： ----- -----			相关元件	SM193 SD100～SD106			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
D	WORD																●				●		●		

指令说明

```
├───┤ ├───[   TRD      D    ]
```

D：存储时钟数据的起始单元

1．将可编程控制器的实时时钟的时钟数据读出，保存在 D 所指定的存储单元中。

2．SM193 为读写实时时钟错误标志。

使用示例

```
     M0                          2011
├───┤ ├──────────────────[ TRD   D20       ]        LD     M0
                                                     TRD    D20
```

M0 为 ON 时，执行 TRD 指令，把系统的时间分别送到 D20 开始的 7 个单元中。

指令的执行结果如下：

	元件	项目	时钟数据		元件	项目
实时时钟用的寄存器	SD100	年	2000～2099	---------→	D20	年
	SD101	月	1～12	---------→	D21	月
	SD102	日	1～31	---------→	D22	日
	SD103	时	0～23	---------→	D23	时
	SD104	分	0～59	---------→	D24	分
	SD105	秒	0～59	---------→	D25	秒
	SD106	星期	0～6	---------→	D26	星期

注意事项

1．在系统出现时钟设置错误时，TRD 读时间不成功。

2．在 MXProgrammer 中可以通过单击菜单栏中的“PLC”→“PLC 时钟”选项读取时钟数据。

6.10.2 TWR : 实时时钟写指令

	TWR		D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：TWR (连续) 3步		-----	相关元件	SM193 SD100 ~ SD106			
	32位： -----		-----					

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
D	WORD																●				●		●		

指令说明

├──┤ ├──[TWR S]

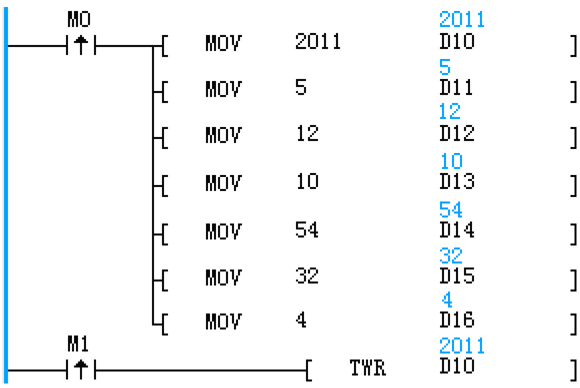
S：写入时钟数据的起始单元

1．将设定时钟的数据写入可编程控制器的实时时钟中。

2．SM193 为读写实时时钟错误志。

使用示例

设定实时时钟：2011 年 5 月 12 日 10 时 54 分 32 秒，星期四




```
LDP    M0
MOV    2011 D10
MOV    5 D11
MOV    12 D12
MOV    10 D13
MOV    54 D14
MOV    32 D15
MOV    4 D16
LDP    M1
TWR    D10
```

通过 TWR 改变系统的时间，见图：

	元件	项目	时钟数据		元件	项目
实时时钟用的寄存器	D10	年	2000～2099	→	SD100	年
	D11	月	1～12	→	SD101	月
	D12	日	1～31	→	SD102	日
	D13	时	0～23	→	SD103	时
	D14	分	0～59	→	SD104	分
	D15	秒	0～59	→	SD105	秒
	D16	星期	0～6	→	SD106	星期

1．检测到 M0 的上升沿时，给 D10 开始的连续 7 个单元写入时间设定值。

2．检测到 M1 的上升沿时，执行 TWR 指令一次，将 D10 开始的连续 7 个单元的数值写到系统时钟。

注意事项

1．写入的时间数据必须要满足公历的要求，否则指令执行不成功。

2．建议使用边沿触发作为该指令的执行条件。

3．在 MXProgrammer 中若设置了下载密码，则下载密码输入正确后，可以通过单击菜单栏中的“PLC”→“PLC 时钟”选项对时钟进行设置。

6.10.3 TADD : 时间加指令

	TADD		S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	7步			相关元件	零标志：SM180		进位标志：SM181	

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	WORD																●	●			●		●		
S2	WORD																●	●			●		●		
D	WORD																●				●		●		

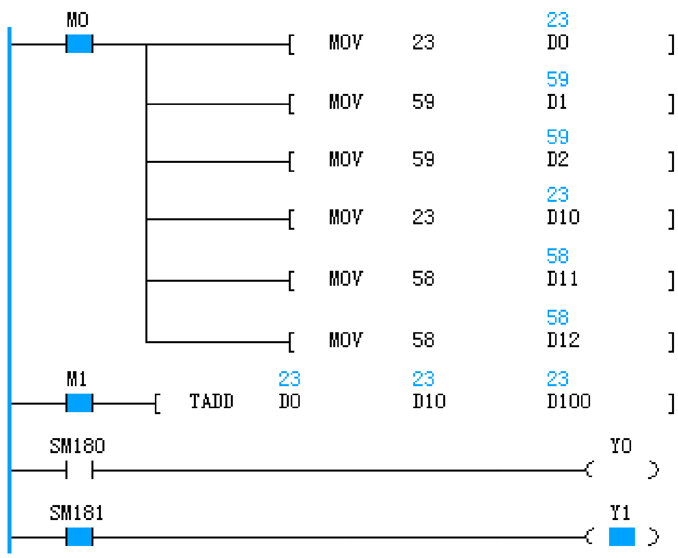
指令说明

├──┤ ├──[TADD S1 S2 D]

S1：时钟数据 1 **S2**：时钟数据 2 **D**：时间相加结果存储单元

1．对时间格式的数据进行加法运算，结果存储在 D 所指定的连续 3 个单元内，运算规则按时间格式执行。
2．处理完成的结果会影响零标志（SM180），进位标志（SM181）。

使用示例




```
LD    M0
MOV   23 D0
MOV   59 D1
MOV   59 D2
MOV   23 D10
MOV   58 D11
MOV   58 D12
LD    M1
TADD  D0 D10 D100
LD    SM180
OUT   Y0
LD    SM181
OUT   Y1
```

1．当 M0 为 ON 时，将时间数据送到 D0 开始的连续 3 个单元和 D10 开始的连续 3 个单元中。
2．当 M1 为 ON 时，执行 TADD 指令，将 D0 开始的连续 3 个单元的内容与 D10 开始的连续 3 个单元的内容相加，结果保存在 D100 开始的连续 3 个单元中。
3．运算结果出现进位时，进位标志（SM181）置 ON。

S1			S2			D	
D0	23时	+	D10	23 时	=	D100	23 时
D1	59分		D11	58 分		D101	58 分
D2	59秒		D12	58 秒		D102	57 秒

注意事项

1．在 S1，S2 所指的 3 个存储单元内保存时间数据，对不满足时间格式的数据，PLC 信息中报“指令操作数数值非法”错误，指令不执行。
2．参与运算的数据应符合时间格式：
“时”的设定范围：0 ～ 23 “分”的设定范围：0 ～ 59 “秒”的设定范围：0 ～ 59

6.10.4 TSUB：时间减指令

	TSUB		S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	7步			相关元件	零标志：SM180		借位标志：SM182	

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	WORD																•	•			•		•		
S2	WORD																•	•			•		•		
D	WORD																•				•		•		

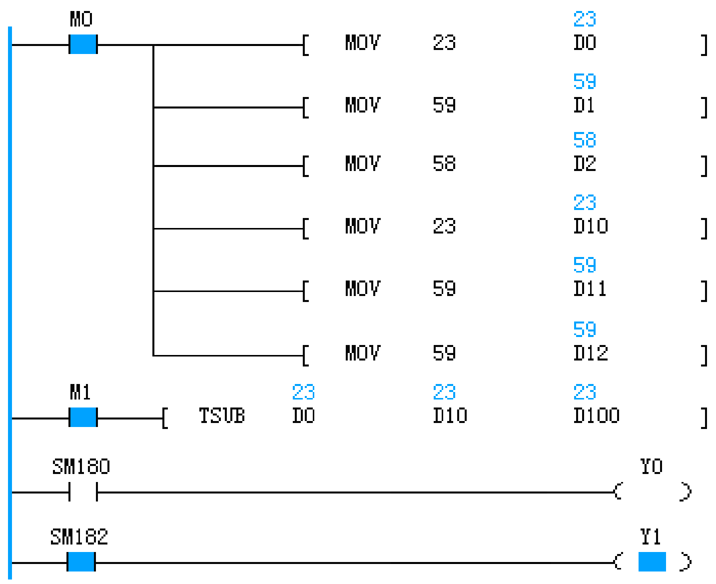
指令说明

|—| |—[TSUB S1 S2 D]

S1：时钟数据 1　**S2**：时钟数据 2　**D**：时间结果存储单元

1. 对时间格式的数据进行减法运算，结果存储在 D 开始的连续 3 个单元内，运算规则按时间格式执行。
2. 处理完成的结果会影响零标志（SM180），借位标志（SM182）。

使用示例




```
LD    M0
MOV   23 D0
MOV   59 D1
MOV   58 D2
MOV   23 D10
MOV   59 D11
MOV   59 D12
LD    M1
TSUB  D0 D10 D100
LD    SM180
OUT   Y0
LD    SM182
OUT   Y1
```

1. 当 M0 为 ON 时，将时间数据送到 D0 开始的连续 3 个单元和 D10 开始的连续 3 个单元中。
2. 当 M1 为 ON 时，执行 TSUB 指令，将 D0 开始的连续 3 个单元的内容与 D10 开始的连续 3 个单元的内容相减，结果保存在 D100 开始的连续 3 个单元中。
3. 运算结果出现借位时，借位标志（SM182）置 ON。

S1			S2			D	
D0	23时	－	D10	23时	＝	D100	23时
D1	59分		D11	59分		D101	59分
D2	58秒		D12	59秒		D102	59秒

注意事项

1. 在 S1，S2 所指的 3 个存储单元内保存时间数据，对不满足时间格式的数据，PLC 信息中报“指令操作数数值非法”错误，指令不执行。
2. 参与运算的数据应符合时间格式：
   “时”的设定范围：0 ～ 23　“分”的设定范围：0 ～ 59　“秒”的设定范围：0 ～ 59

6.10.5 HOUR : 计时表指令

	HOUR		S D1 D2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	8步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●	
D1	INT																●				●		●		
D2	BOOL		●	●	●			●															●		

* 该指令使用 C 元件时，只能使用 C0 ～ C9。

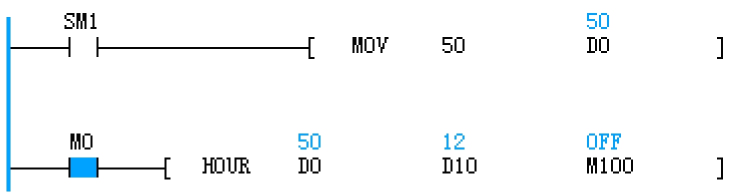
指令说明

├──┤ ├─[HOUR S D1 D2]

S：小时比较数据 **D1**：时间存储单元 **D2**：报警输出地址

对输入接点处于ON状态的时间以小时为单位进行累加，结果保存在D1中，当D1 ≥ S时，D2输出ON状态。将不满 1 小时的当前值以秒为单位保存在 D1 后一单元中。

使用示例




```
LD    SM1
MOV   50 D0
LD    M0
HOUR  D0 D10 M100
```

1. 初始上电第一周期设定 HOUR 指令的比较数据。
2. 当 M0 为 ON 时，执行 HOUR 指令，对时间以小时为单位进行累加。
3. 当 M0 的 ON 状态累加时间大于等于 50 时，M100 输出 ON 状态。
4. 不满 1 小时的当前值以秒为单位保存在 D11 单元中。

注意事项

1. 为了能在切断 PLC 的电源后仍然使用当前数据，请指定 D1 为停电保持用的软元件单元。若使用普通软元件单元，当 PLC 的电源被切断或进行 RUN → STOP 的操作时，当前数据将被清除。
2. 即使报警输出 D2 为 ON，仍然能够继续计数。
3. 本指令中 S 和 D1 为 16 位整数。当 D1 大于 32,767 时又重新从 0 开始。

6.10.6 TCMP : 时间比较指令

	TCMP	P	S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	7步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT										●	●	●	●	●	●	●	●	*	●	●	●	●		
S2	INT										●	●	●	●	●	●	●	●	*	●	●	●	●		
D	BOOL		●	●	●																				

* 该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

├──┤ ├──[TCMP S1 S2 D]

S1：时钟数据 1　**S2**：时钟数据 2　**D**：时间结果存储单元

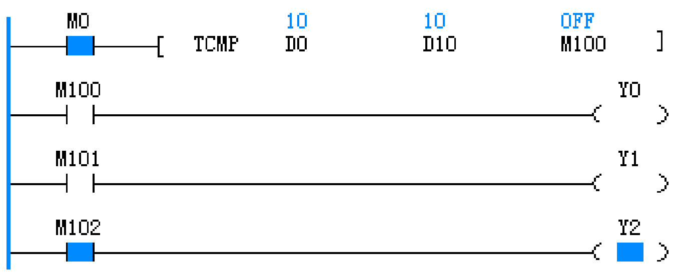
将 S1 开始的三个连续的字单元所指定的比较时间数据 1（S1，S1+1，S1+2 分别对应时，分，秒）与 S2 开始的三个连续的字单元所指定的比较时间数据 2（S2，S2+1，S2+2 分别对应时，分，秒）进行比较，比较结果决定了 D 开始的三个连续位单元的状态。

S1 （时）	<	S2 （时）	→	D ON
S1+1 （分）		S2+1 （分）		D+1 OFF
S1+2 （秒）		S2+2 （秒）		D+2 OFF

S1 （时）	=	S2 （时）	→	D OFF
S1+1 （分）		S2+1 （分）		D+1 ON
S1+2 （秒） S1+2 （秒）		S2+2 （秒）		D+2 OFF

S1 （时）	>	S2 （时）	→	D OFF
S1+1 （分）		S2+1 （分）		D+1 OFF
S1+2 （秒）		S2+2 （秒）		D+2 ON

使用示例




```
LD    M0
TCMP  D0 D10 M100
LD    M100
OUT   Y0
LD    M101
OUT   Y1
LD    M102
OUT   Y2
```

当 M0 为 ON 时，执行 TCMP 指令，对分别以 D0、D10 为起始单元的时间数据进行比较，比较的结果决定了 M100 ～ M102 三个连续位单元的状态。

- 执行前：D0=10，D1=27，D2=20，D10=10，D11=25，D12=6，M100=OFF，M101=OFF，M102=OFF。
- 执行后：D0=10，D1=27，D2=20，D10=10，D11=25，D12=6，M100=OFF，M101=OFF，M102=ON。

注意事项

1. 以 S1、S2 为起始的 3 个字单元的时间数据必须符合时间格式，否则 PLC 信息中报“指令操作数数值非法”错误（如：24，10，31 和 13，59，60 都不合法），指令不执行。
2. 参与运算的数据应符合时间格式：
   “时”的设定范围：0 ～ 23　　“分”的设定范围：0 ～ 59　　“秒”的设定范围：0 ～ 59

6.10.7 HTOS : [时、分、秒] 数据的秒转换指令

D	HTOS	P	S D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：HTOS (连续) 5步 32位：DHTOS (连续) 6步		HTOSP (脉冲) 6步 DHTOSP (脉冲) 7步	相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	INT										●	●	●	●	●	●	●	●	*	●	●		●		
D	（D）INT											●	●	●	●	●	●	●	*	*	●		●		

* S 使用 C 元件时，只能使用 C0 ～ C199。

* 16 位指令的 D 使用 C 元件时，只能使用 C0 ～ C199；
32 位指令的 D 使用 C 元件时，只能使用 C200 ～ C255 且不能使用 T 元件。

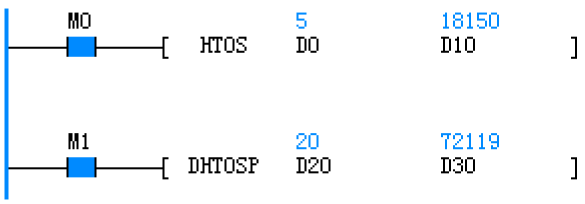
指令说明

├──┤ ├──[HTOS S D]

将 S 开始的连续三个单元（S 中存放“时”，S+1 中存放“分”，S+2 中存放“秒”）分别存放的时、分、秒时间数据转化成秒数据存放在 D 单元中，S 中数据保持不变。

S （时）		
S+1 （分）	──►	D（秒）
S+2 （秒）		

使用示例




```
LD      M0
HTOS    D0 D10
LD      M1
DHTOSP  D20 D30
```

1. 当 M0 为 ON 时，执行 HTOS 指令，将 D0、D1、D2 中的时间数据转化成秒数据存放在 D10 中。因为 HTOS 为连续执行型，所以如果 D0、D1、D2 中的时间数据发生变化，D10 中的数值也会跟着发生变化。
   - 执行前：D0=5，D1=2，D2=30，D10=0；
   - 执行后：D0=5，D1=2，D2=30，D10=18150。
2. 当 M1 由 OFF → ON 变化时，DHTOSP 指令执行一次，将 D20、D21、D22 中的时间数据转化成秒数据存放在（D31，D30）中。
   - 执行前：D20=20，D21=1，D22=59，（D31，D30）=0；
   - 执行后：D20=20，D21=1，D22=59，（D31，D30）=72119

注意事项

参与转换的分、秒时间数据必须符合时间格式：“分”的设定范围：0 ～ 59，“秒”的设定范围：0 ～ 59 否则 PLC 信息中报“指令操作数数值非法”错误，指令不执行。

6.10.8 STOH : 秒数据的 [时、分、秒] 转换指令

D	STOH	P	S D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：STOH (连续) 5步		STOHP (脉冲) 6步	相关元件				
	32位：DSTOH (连续) 6步		DSTOHP (脉冲) 7步					

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	（D）INT										●	●	●	●	●	●	●	●	＊	＊	●		●		
D	INT											●	●	●	●		●	●	＊	●	●		●		

＊ D 使用 C 元件时，只能使用 C0 ～ C199。

＊ 16 位指令的 S 使用 C 元件时，只能使用 C0 ～ C199；

32 位指令的 S 使用 C 元件时，只能使用 C200 ～ C255 且不能使用 T 元件。

指令说明

```
├──┤ ├──[ STOH S D ]
```

将 S 中存放的秒数据转化成时、分、秒时间数据存放在 D 开始的连续三个单元中。D 中存放“时”，D+1 中存放“分”，D+2 中存放“秒”，S 中数据保持不变。

S（秒）	→	D （时）
		D+1 （分）
		D+2 （秒）

使用示例

```
     M0                    3700         1
──┤ ├────[ STOH     D0           D10          ]

     M1                    3800         1
──┤ ├────[ DSTOHP   D20          D30          ]
```

```
LD      M0
STOH    D0 D10
LD      M1
DSTOHP  D20 D30
```

1. 当 M0 为 ON 时，执行 STOH 指令，将 D0 中的秒数据转化成时、分、秒时间数据分别存放在 D10、D11、D12 中。因为 STOH 为连续执行型，所以如果 D0 中的秒数据发生变化，D10、D11、D12 中的数值也会跟着发生变化。
   - 执行前：D0=3700，D10=0，D11=0，D12=0；
   - 执行后：D0=3700，D10=1，D11=1，D12=40。
2. 当 M1 由 OFF → ON 变化时，DSTOHP 指令执行一次，将（D21，D20）中的秒数据转化成时、分、秒时间数据分别存放在 D30、D31、D32 中。
   - 执行前：（D21，D20）=3800，D30=0，D31=0，D32=0；
   - 执行后：（D21，D20）=3800，D30=1，D31=3，D32=20。

6.11 比较触点指令

6.11.1 LD(=,>,>=,<,<=,<>) : 整数比较 LD 指令

D	LD		S1	S2	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√	√	√
步长	16位：LD (连续) 5步 32位：DLD (连续) 7步			----- -----	相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	(D)INT										•	•	•	•	•	•	•	•	*	*	•	*	•	•	
S2	(D)INT										•	•	•	•	•	•	•	•	*	*	•	*	•	•	

* 16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z 元件。

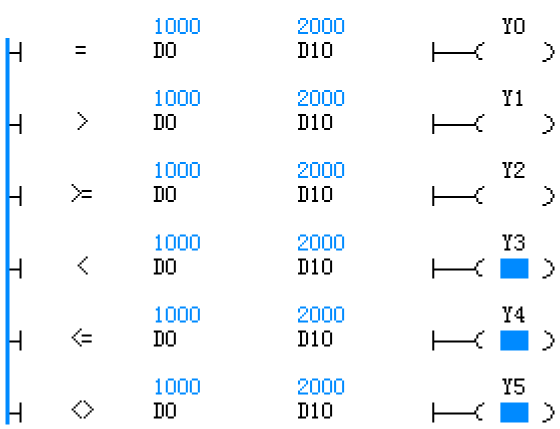
指令说明

S1：比较数据 1 **S2**：比较数据 2

对 S1、S2 单元的内容进行比较，比较后的结果用于驱动后段运算。

16位指令	32位指令	导通条件	非导通条件
LD=	DLD=	[S1] = [S2]	[S1] ≠ [S2]
LD>	DLD>	[S1] > [S2]	[S1] <= [S2]
LD>=	DLD>=	[S1] >= [S2]	[S1] < [S2]
LD<	DLD<	[S1] < [S2]	[S1] >= [S2]
LD<=	DLD<=	[S1] <= [S2]	[S1] > [S2]
LD<>	DLD<>	[S1] ≠ [S2]	[S1] = [S2]

使用示例




```
LD=    D0 D10
OUT    Y0
LD>    D0 D10
OUT    Y1
LD>=   D0 D10
OUT    Y2
LD<    D0 D10
OUT    Y3
LD<=   D0 D10
OUT    Y4
LD<>   D0 D10
OUT    Y5
```

对 D0、D10 单元的数据进行比较，比较后的结果决定后段元件输出状态。

6.11.2 RLD(=,>,>=,<,<=,<>) : 浮点数比较 RLD 指令

	RLD		S1 S2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	7步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	REAL																•				•		•	•	
S2	REAL																•				•		•	•	

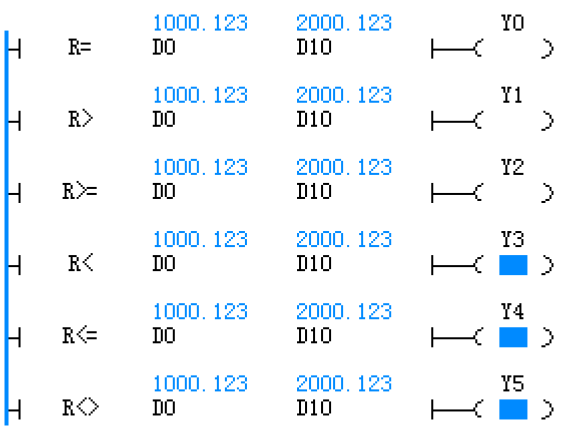
指令说明

S1：比较浮点数据 1　　**S2**：比较浮点数据 2

对 S1、S2 单元的内容进行比较，比较后的结果用于驱动后段运算。

32位指令	导通条件	非导通条件
RLD=	[S1] = [S2]	[S1] ≠ [S2]
RLD>	[S1] > [S2]	[S1] <= [S2]
RLD>=	[S1] >= [S2]	[S1] < [S2]
RLD<	[S1] < [S2]	[S1] >= [S2]
RLD<=	[S1] <= [S2]	[S1] > [S2]
RLD<>	[S1] ≠ [S2]	[S1] = [S2]

使用示例




```
RLD=    D0 D10
OUT     Y0
RLD>    D0 D10
OUT     Y1
RLD>=   D0 D10
OUT     Y2
RLD<    D0 D10
OUT     Y3
RLD<=   D0 D10
OUT     Y4
RLD<>   D0 D10
OUT     Y5
```

对（D1，D0）、（D11，D10）单元的数据进行比较，比较后的结果决定后段元件输出状态。

6.11.3 AND(=,>,>=,<,<=,<>) : 整数比较 AND 指令

D	AND		S1	S2	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√	√	√
步长	16位：AND (连续) 5步 32位：DAND (连续) 7步			----- -----	相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	(D)INT										•	•	•	•	•	•	•	•	*	*	•	*	•	•	
S2	(D)INT										•	•	•	•	•	•	•	•	*	*	•	*	•	•	

* 16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z 元件。

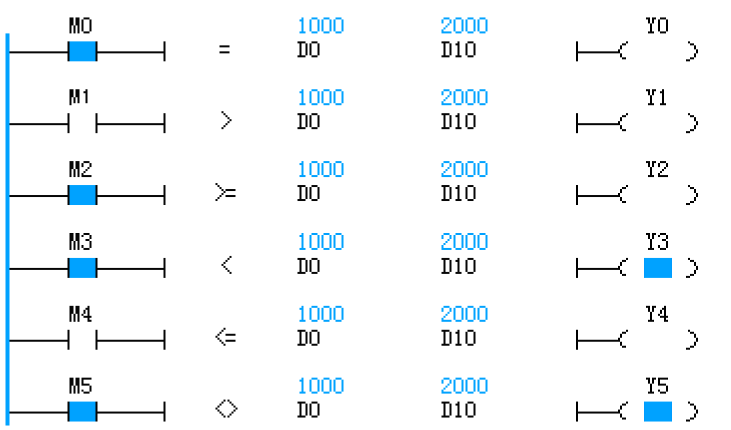
指令说明

S1：比较数据 1　　**S2**：比较数据 2

对 S1、S2 单元的内容进行比较，比较的结果再与其它节点串联，用于驱动后段运算。

16位指令	32位指令	导通条件	非导通条件
AND=	DAND=	[S1] = [S2]	[S1] ≠ [S2]
AND>	DAND>	[S1] > [S2]	[S1] <= [S2]
AND>=	DAND>=	[S1] >= [S2]	[S1] < [S2]
AND<	DAND<	[S1] < [S2]	[S1] >= [S2]
AND<=	DAND<=	[S1] <= [S2]	[S1] > [S2]
AND<>	DAND<>	[S1] ≠ [S2]	[S1] = [S2]

使用示例




```
LD      M0
AND=    D0 D10
OUT     Y0
LD      M1
AND>    D0 D10
OUT     Y1
LD      M2
AND>=   D0 D10
OUT     Y2
LD      M3
AND<    D0 D10
OUT     Y3
LD      M4
AND<=   D0 D10
OUT     Y4
LD      M5
AND<>   D0 D10
OUT     Y5
```

对 D0、D10 单元的数据进行比较，比较的结果再与其它触点串联，决定后段元件的输出状态。

6.11.4 RAND (=,>,>=,<,<=,<>) : 浮点数比较 RAND 指令

	RAND		S1 S2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	7步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	REAL																●				●		●	●	
S2	REAL																●				●		●	●	

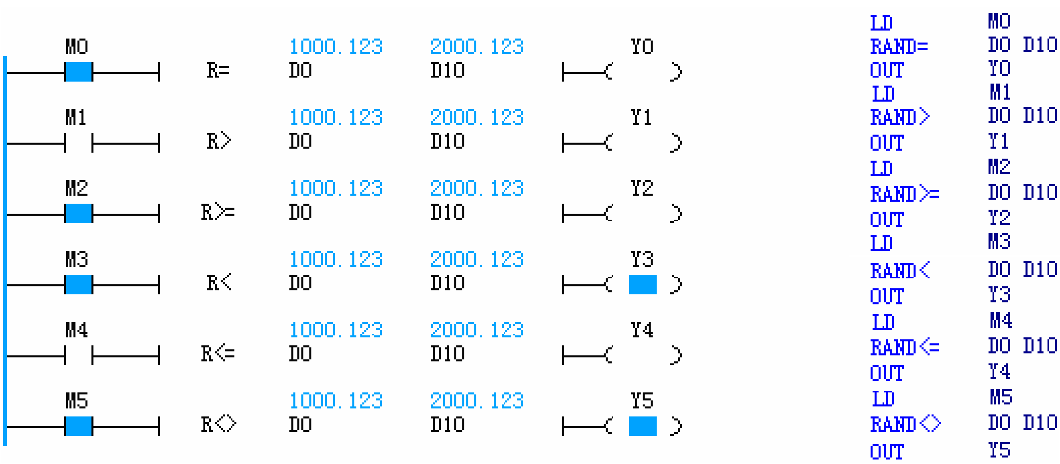
指令说明

S1：比较浮点数据 1　　**S2**：比较浮点数据 2

对 S1、S2 单元的内容进行比较，比较的结果再与其它节点串联，用于驱动后段运算。

32位指令	导通条件	非导通条件
RAND=	[S1] = [S2]	[S1] ≠ [S2]
RAND>	[S1] > [S2]	[S1] <= [S2]
RAND>=	[S1] >= [S2]	[S1] < [S2]
RAND<	[S1] < [S2]	[S1] >= [S2]
RAND<=	[S1] <= [S2]	[S1] > [S2]
RAND<>	[S1] ≠ [S2]	[S1] = [S2]

使用示例



比较（D1，D0）和（D11，D10）单元的内容，比较的结果再与其它节点串联，决定后段元件的输出状态。

6.11.5 OR (=,>,>=,<,<=,<>) : 整数比较 OR 指令

D	OR		S1	S2	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√	√	√
步长	16位：OR (连续) 5步 32位：DOR (连续) 7步			----- -----	相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	(D)INT										●	●	●	●	●	●	●	●	*	*	●	*	●	●	
S2	(D)INT										●	●	●	●	●	●	●	●	*	*	●	*	●	●	

* 16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z 元件。

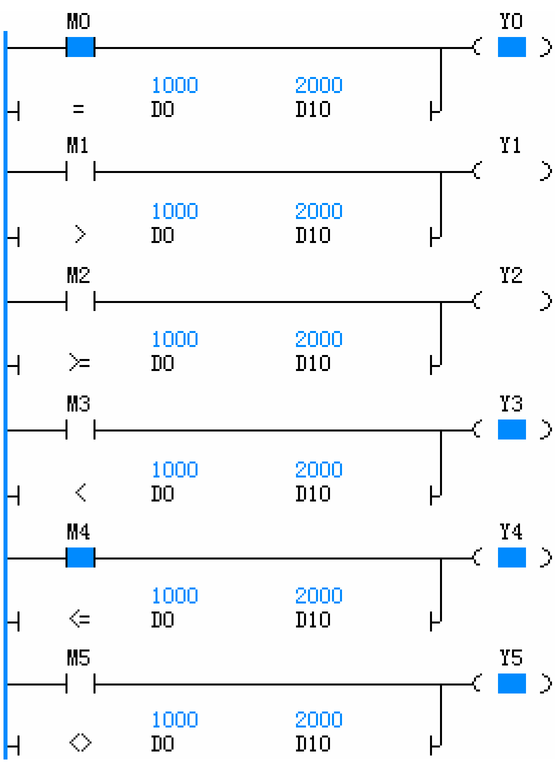
指令说明

S1：比较数据 1 **S2**：比较数据 2

对 S1、S2 单元的内容进行比较，比较的结果再与其它节点并联，用于驱动后段运算。

16位指令	32位指令	导通条件	非导通条件
OR=	DOR=	[S1] = [S2]	[S1] ≠ [S2]
OR>	DOR>	[S1] > [S2]	[S1] <= [S2]
OR>=	DOR>=	[S1] >= [S2]	[S1] < [S2]
OR<	DOR<	[S1] < [S2]	[S1] >= [S2]
OR<=	DOR<=	[S1] <= [S2]	[S1] > [S2]
OR<>	DOR<>	[S1] ≠ [S2]	[S1] = [S2]

使用示例




```
LD    M0
OR=   D0 D10
OUT   Y0
LD    M1
OR>   D0 D10
OUT   Y1
LD    M2
OR>=  D0 D10
OUT   Y2
LD    M3
OR<   D0 D10
OUT   Y3
LD    M4
OR<=  D0 D10
OUT   Y4
LD    M5
OR<>  D0 D10
OUT   Y5
```

对 D0、D10 单元的数据进行比较，比较的结果再与其它节点并联，决定后段元件的输出状态。

6.11.6 ROR (=,>,>=,<,<=,<>) : 浮点数比较 ROR 指令

	ROR		S1	S2	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√	√	√
步长	7步				相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	REAL																•				•		•	•	
S2	REAL																•				•		•	•	

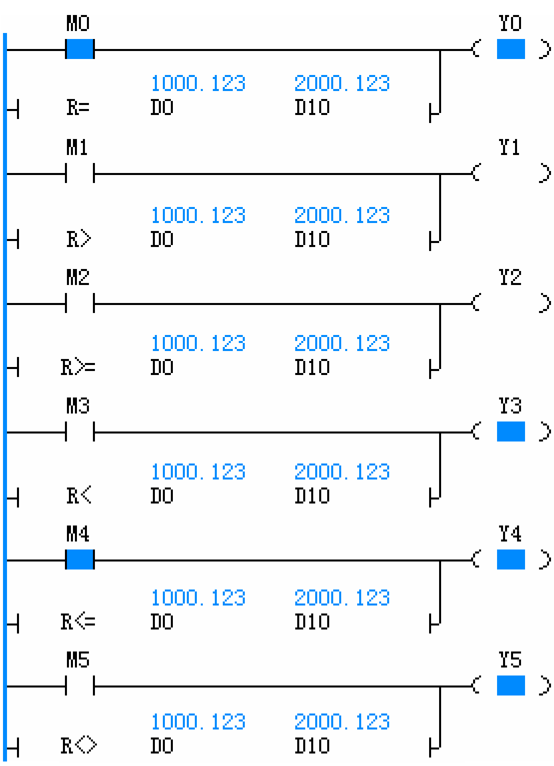
指令说明

S1：比较浮点数据 1　　**S2**：比较浮点数据 2

对 S1、S2 单元的内容进行比较，比较的结果再与其它节点并联，用于驱动后段运算。

32位指令	导通条件	非导通条件
ROR=	[S1] = [S2]	[S1] ≠ [S2]
ROR>	[S1] > [S2]	[S1] <= [S2]
ROR>=	[S1] >= [S2]	[S1] < [S2]
ROR<	[S1] < [S2]	[S1] >= [S2]
ROR<=	[S1] <= [S2]	[S1] > [S2]
ROR<>	[S1] ≠ [S2]	[S1] = [S2]

使用示例




```
LD      M0
ROR=    D0 D10
OUT     Y0
LD      M1
ROR>    D0 D10
OUT     Y1
LD      M2
ROR>=   D0 D10
OUT     Y2
LD      M3
ROR<    D0 D10
OUT     Y3
LD      M4
ROR<=   D0 D10
OUT     Y4
LD      M5
ROR<>   D0 D10
OUT     Y5
```

比较（D1，D0）和（D11，D10）单元的数据，比较的结果再与其它节点并联，决定后段元件的输出状态。

6.12 数值转换指令

6.12.1 FLT : 整数转换浮点数指令

D	FLT	P	S D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：FLT (连续) 6步 FLTP (脉冲) 7步 32位：DFLT (连续) 7步 DFLTP (脉冲) 8步			相关元件	零标志：SM180			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	(D) INT										●	●	●	●	●	●	●	●	*	*	●	*	●	●	*
D	REAL																●				●		●		

* 16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z、Un.b 元件。

指令说明

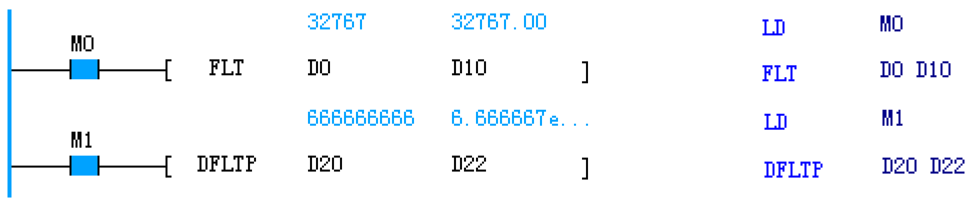
├──┤ ├──[FLT S D]

S： 要转换的来源数据 **D：** 转换的结果

将源操作数 S 由整数转换成浮点数，结果赋予目的操作数 D 中。

使用示例

示例 1.

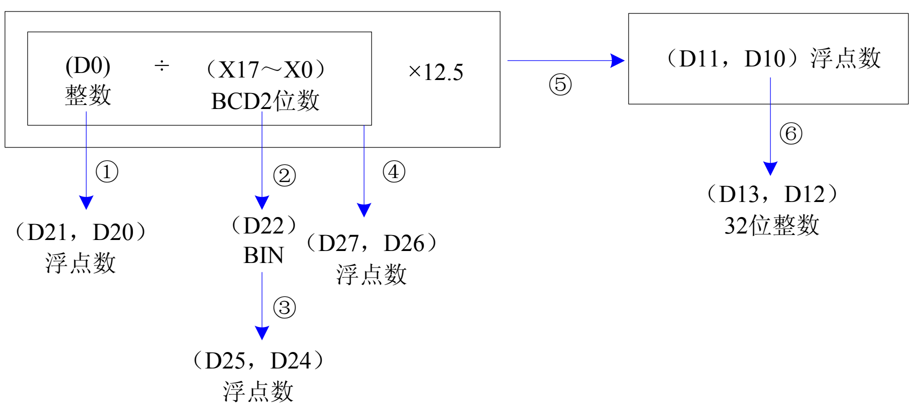


1．当 M0 为 ON 时，执行 FLT 指令，D0=32,767 由整数转换成浮点数，赋给（D11，D10），（D11，D10）=32,767.00。

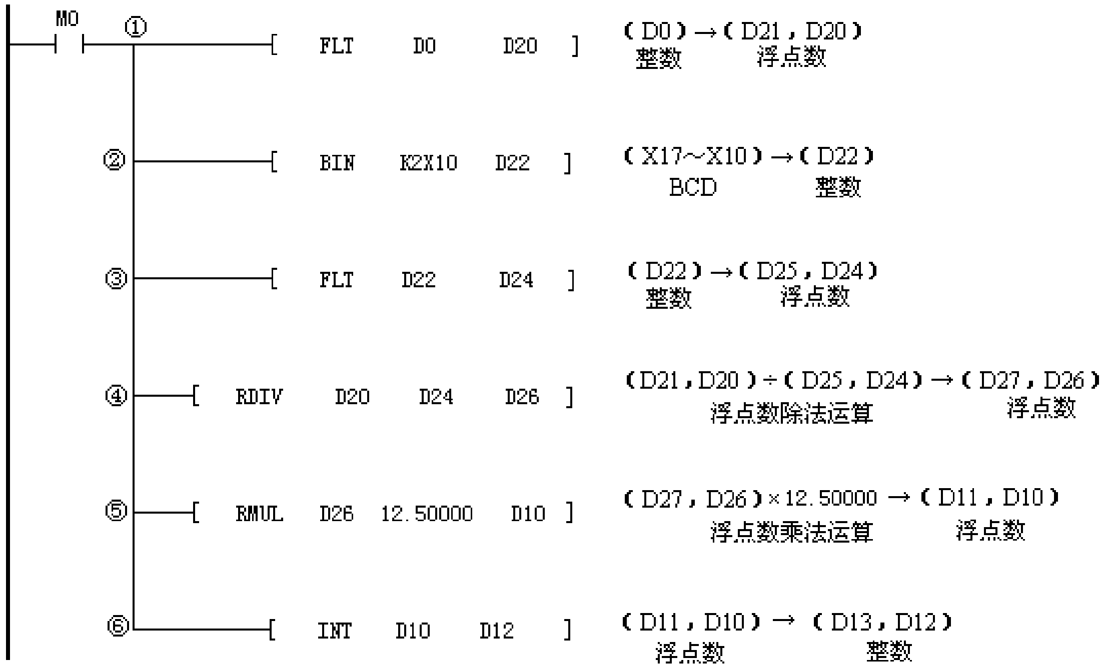
2．当 M1 由 OFF → ON 变化时，DFLTP 指令执行一次，（D21，D20）=666,666,666 由整数转换成浮点数，赋给（D23， D22），（D23，D22）=6.666667e+008。

示例 2．采用浮点数运算的四则运算

（1）计算例



（2）程序



注意事项

当转换结果为 0 时，零标志位（SM180）置 ON。

6.12.2 INT : 浮点数转换整数指令

D	INT	P	S D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：INT (连续) 6步 32位：DINT (连续) 7步		INTP (脉冲) 7步 DINTP (脉冲) 8步	相关元件	零标志: SM180 进位标志: SM181 借位标志: SM182			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	REAL																●				●		●	●	
D	(D)INT											●	●	●	●		●		*	*	●	*	●		

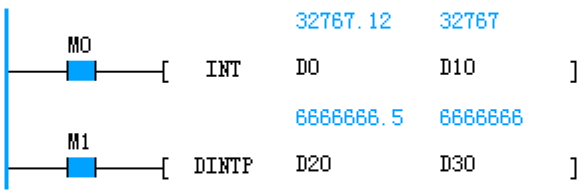
＊ 16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z 元件。

指令说明

S：要转换的来源数据　　**D**：转换的结果

1. 将源操作数 S 由浮点数转换成整数，结果赋予目的操作数 D 中。
2. 该指令影响零标志（SM180），进位标志（SM181），借位标志（SM182）。

使用示例




```
LD     M0
INT    D0 D10
LD     M1
DINTP  D20 D30
```

1. 当 M0 为 ON 时，执行 INT 指令，（D1，D0）=32767.12 由浮点数转换成整数，赋给 D10，D10=32,767。结果舍去了小数，借位标志（SM182）置 ON。
2. 当 M1 由 OFF → ON 变化时，DINTP 指令执行一次，（D21，D20）=6666666.5 由浮点数转换成整数，赋给（D31，D30），（D31，D30）=6,666,666。结果舍去了小数，借位标志（SM182）置 ON。

注意事项

1. 当转换结果为 0 时，零标志位（SM180）置 ON。
2. 当转换结果四舍五入时，进位标志位（SM181）置 ON。
3. 当转换结果舍去小数时，借位标志位（SM182）置 ON。
4. 16 位指令：当 S>32,767 时，D=32,767；当 S< － 32,768 时，D= － 32,768，同时进位标志位（SM181）置 ON。
   32 位指令：当 S>2,147,483,647 时，D=2,147,483,647；当 S< － 2,147,483,648 时，D= － 2,147,483,648，同时进位标志位（SM181）置 ON。

请参见 *3.7.2* 标志位的相关描述。

6.12.3 BCD：字转换为 BCD 码指令

D	BCD	P	S	D	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√	√	√
步长	16位：BCD（连续） 5步 32位：DBCD（连续） 7步		BCDP（脉冲） 6步 DBCDP（脉冲） 8步		相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	(D) WORD										●	●	●	●	●	●	●	●	*	*	●	*	●	●	*
D	(D) WORD											●	●	●	●		●		*	*	●	*	●		

* 16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z、Un.b 元件。

指令说明

├──┤ ├──[BCD S D]

S：要转换的来源数据 **D**：转换的结果

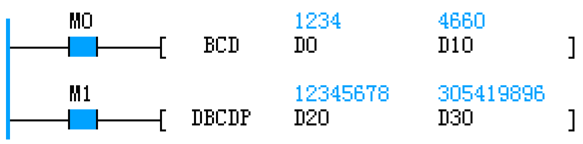
1．将整数 S 转换成 BCD 码，转换结果赋予 D。
2．16 位指令中 0 ≤ S ≤ 9,999，32 位指令中 0 ≤ S ≤ 99,999,999。
3．用 4 位二进制数组成一组代码，来表示 0 ～ 9 十个数字，这组代码称为二－十进制代码，又称 BCD 码。用 0000 ～ 1001 来表示一位十进制数 0 ～ 9，示例如下：

十进制数： 1 6
↙ ↘
BCD 码： 0 0 0 1 0 1 1 0

4．可编程控制器的运算按照 INT（整数）数据进行处理，在带 BCD 译码的 7 段码显示器中显示数值时，可使用本指令。

使用示例

示例 1.

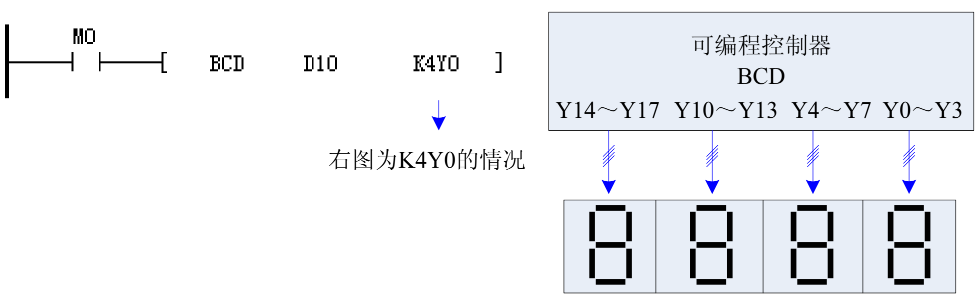



```
LD      M0
BCD     D0 D10
LD      M1
DBCDP   D20 D30
```

1． 当 M0 为 ON 时，执行 BCD 指令，将 D0 中的整数 1,234 转换成 BCD 码赋予 D10，D10=0001,0010,0011,0100
2． 当 M1 由 OFF → ON 变化时，DBCDP 指令执行一次，将（D21，D20）中的整数 12,345,678 转换成 BCD 码赋予（D31，D30），（D31，D30）=0001,0010,0011,0100,0101,0110,0111,1000。

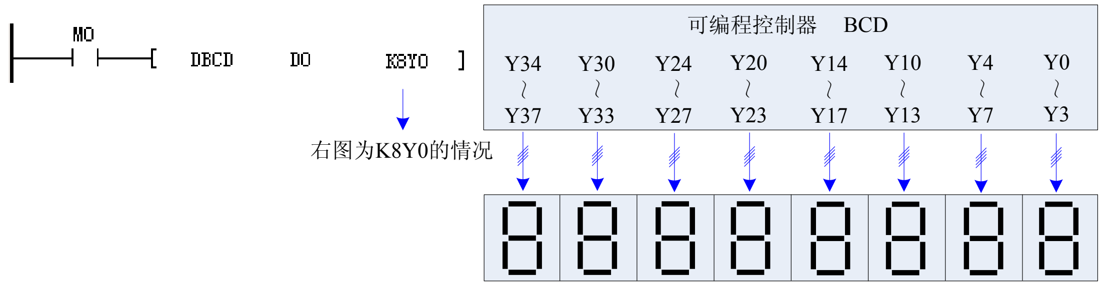
需要注意的是，由于监控时计算机无法显示 BCD 码，因此与四位二进制数一样，它从高位到低位的权值分别为 8、4、2、1，每组二进制代码各位加权系数之和便为它所代表的十进制数，就将其转换为十进制显示。

示例 2．16 位运算（BCD、BCDP），7 段数码管显示 1 位数以上、4 位数以下的情况



D	显示的数据位数	显示的数据范围
K1Y0	1 位数	0 ～ 9
K2Y0	2 位数	00 ～ 99
K3Y0	3 位数	000 ～ 999
K4Y0	4 位数	0000 ～ 9999

示例 3．32 位运算（DBCD、DBCDP），7 段数码管显示 1 位数以上、8 位数以下的情况



（D+1，D）	显示的数据位数	显示的数据范围
K1Y0	1 位数	0 ～ 9
K2Y0	2 位数	00 ～ 99
K3Y0	3 位数	000 ～ 999
K4Y0	4 位数	0000 ～ 9999
K5Y0	5 位数	00000 ～ 99999
K6Y0	6 位数	000000 ～ 999999
K7Y0	7 位数	0000000 ～ 9999999
K8Y0	8 位数	00000000 ～ 99999999

注意事项

当 16 位指令中 S>9,999 或 32 位指令中 S>99,999,999 时，PLC 信息中报“指令操作数数值非法”，不执行转换，D 内容不变。

6.12.4 BIN : BCD 码转换为字指令

D	BIN	P	S D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：BIN (连续) 5步 32位：DBIN (连续) 7步		BINP (脉冲) 6步 DBINP (脉冲) 8步	相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	(D) WORD										●	●	●	●	●	●	●	●	*	*	●	*	●	●	*
D	(D) WORD											●	●	●	●		●		*	*	●	*	●		

* 16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z、Un.b 元件。

指令说明

├──┤ ├──[BIN S D]

S: 要转换的来源数据　　**D:** 转换的结果

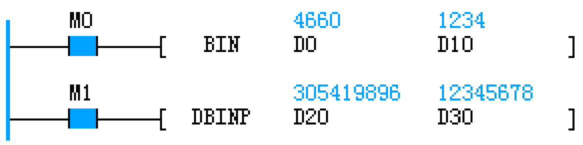
1．将 BCD 码 S 转换成整数，转换结果赋予 D。
2．S 可以转换成的数据的范围（即 D 的数据范围）为 0 ～ 9999。
3．用 4 位二进制数组成一组代码，来表示 0 ～ 9 十个数字，这组代码称为二—十进制代码，又称 BCD 码。用 0000 ～ 1001 来表示一位十进制数 0 ～ 9，与四位二进制数一样，它从高位到低位的权值分别为 8、4、2、1，每组二进制代码各位加权系数之和便为它所代表的十进制数。示例如下：

十进制数：　　1 6
　　　　　　↙ ↘
BCD 码：　　0 0 0 1　0 1 1 0

4．在将数字开关之类，以 BCD 设定的数值转换成可编程控制器中的运算中可以处理的数据后读取的情，况下可以用本指令。

使用示例

示例 1.

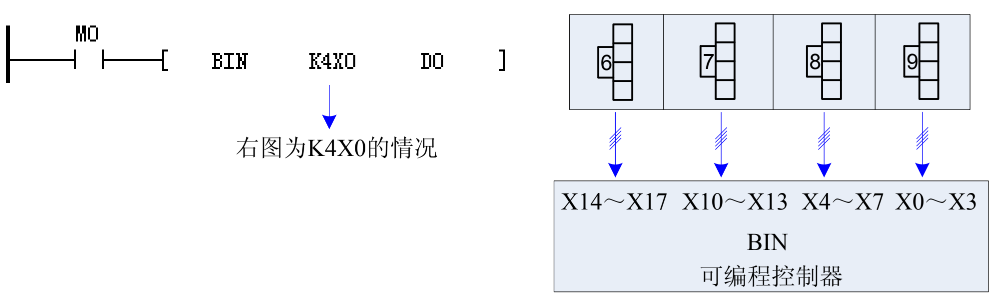



```
LD      M0
BIN     D0 D10
LD      M1
DBINP   D20 D30
```

1． 当 M0 为 ON 时，执行 BIN 指令，将 D0 中的 BCD 码转换成整数赋予 D10，D10=1,234。
2． 当 M1 由 OFF → ON 变化时，DBINP 指令执行一次，将（D21，D20）中的 BCD 码转换成整数赋予（D31，D30），（D31，D30）= 12,345,678。

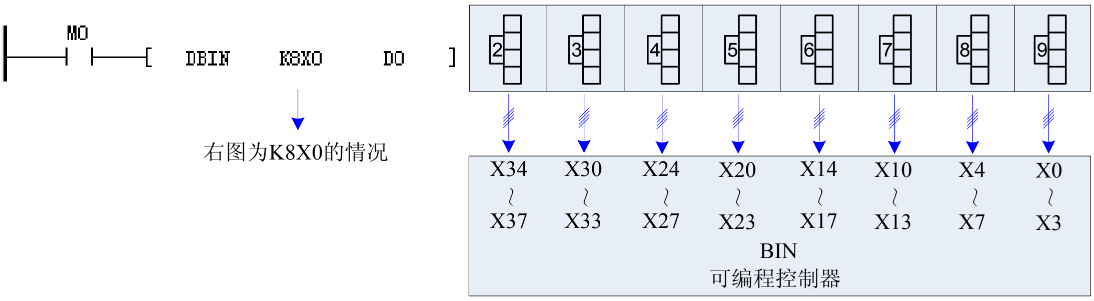
需要注意的是，由于监控时计算机无法显示 BCD 码，因此与四位二进制数一样，它从高位到低位的权值分别为 8、4、2、1，每组二进制代码各位加权系数之和便为它所代表的十进制数，就将其转换为十进制显示。

示例 2．16 位运算数字式开关（BIN、BINP），数字开关 1 位数以上、4 位数以下的情况



S	数字开关的数据位数	数字开关的数据范围
K1X0	1 位数	0 ～ 9
K2X0	2 位数	00 ～ 99
K3X0	3 位数	000 ～ 999
K4X0	4 位数	0000 ～ 9999

示例 2．32 位运算数字式开关（DBIN、DBINP），数字开关 1 位数以上、8 位数以下的情况



（D+1，D）	数字开关的数据位数	数字开关的数据范围
K1X0	1 位数	0 ～ 9
K2X0	2 位数	00 ～ 99
K3X0	3 位数	000 ～ 999
K4X0	4 位数	0000 ～ 9999
K5X0	5 位数	00000 ～ 99999
K6X0	6 位数	000000 ～ 999999
K7X0	7 位数	0000000 ～ 9999999
K8X0	8 位数	00000000 ～ 99999999

注意事项 当 S 转换成的数据的范围（即 D 的数据范围）不在 0 ～ 9999 时， PLC 信息中报“指令操作数数值非法”错误，指令不执行，D 内容不变。

6.12.5 GRY : 字转换为格雷码指令

D	GRY	P	S D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：GRY (连续) 5步 GRYP (脉冲) 6步 32位：DGRY (连续) 7步 DGRYP (脉冲) 8步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	（D）WORD										●	●	●	●	●	●	●	●	*	*	●	*	●	●	*
D	（D）WORD											●	●	●	●		●		*	*	●	*	●		

* 16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255, 且 T、Z、Un.b 不可用。

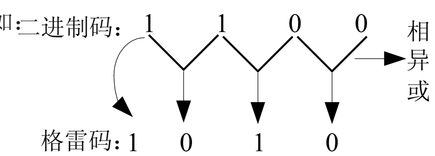
指令说明

|—| |—[GRY S D]

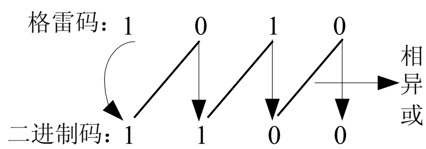
S：要转换的来源数据 **D**：转换的结果

1．将 S 中的整型数据转换成格雷码，转换结果赋予 D。
2．S 可以转换成的数据的范围（即 D 的数据范围）为 0 ～ 9999。
3．常用的可靠性代码主要有格雷码和奇偶校验码。格雷码是一种循环码，也是无权码，它的特点是任意两组相邻代码之间只有一位不同，其余各位都相同，且首位代码之间也仅有一位不同，构成了一个“循环”。这种“循环”特性使它在形成和传输过程中引起的误差较小。
格雷码与二进制码之间的转换：

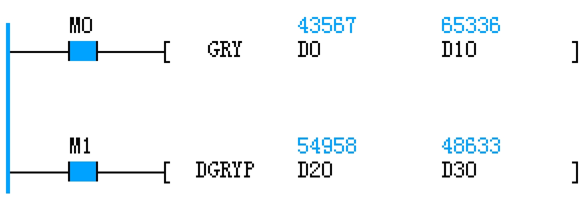
- 二进制→格雷码 保留二进制码最高位作为格雷码的最高位，次高位的格雷码为二进制的高位与次位相异或（相同为 0，不同为 1），依次类推。如：



- 格雷码→二进制 保留格雷码最高位作为自然二进制码的最高位，次高位二进制码为高位自然二进制码与次高位格雷码相异或（相同为 0，不同为 1），依此类推。如：



使用示例




```
LD     M0
GRY    D0 D10
LD     M1
DGRYP  D20 D30
```

1．当 M0 为 ON 时，执行 GRY 指令，将 D0 中的数据（43,567）转化成格雷码存放在 D10 中，D10=65,336。
2．当 M1 由 OFF → ON 变化时，DGRYP 指令执行一次，将（D21，D20）中的数据（54,958）转化成格雷码存放在（D31，D30）中，（D31，D30）=48,633。

6.12.6 GBIN : 格雷码转换为字指令

D	GBIN	P	S	D	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√	√	√
步长	16位：GBIN (连续) 5步　GBINP (脉冲) 6步 32位：DGBIN (连续) 7步　DGBINP (脉冲) 8步				相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	（D）WORD										•	•	•	•	•	•	•	•	*	*	•	*	•	•	*
D	（D）WORD											•	•	•	•		•		*	*	•	*	•		

* 16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255, 且 T、Z、Un.b 不可用。

指令说明

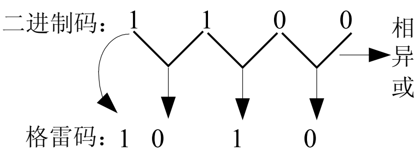
GBIN　S　D

S：要转换的来源数据　　D：转换的结果

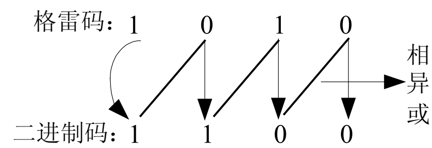
1. 将 S 中的格雷码转换成整数，转换结果赋予 D。
2. 常用的可靠性代码主要有格雷码和奇偶校验码。格雷码是一种循环码，也是无权码，它的特点是任意两组相邻代码之间只有一位不同，其余各位都相同，且首位代码之间也仅有一位不同，构成了一个“循环”。这种“循环”特性使它在形成和传输过程中引起的误差较小。

   格雷码与二进制码之间的转换：

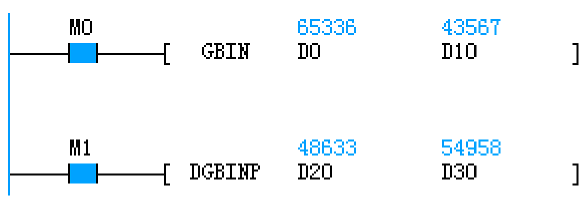
   - 二进制→格雷码　保留二进制码最高位作为格雷码的最高位，次高位的格雷码为二进制的高位与次位相异或（相同为 0，不同为 1），依次类推。如：二进制码：



   - 格雷码→二进制　保留格雷码最高位作为自然二进制码的最高位，次高位二进制码为高位自然二进制码与次高位格雷码相异或（相同为 0，不同为 1），依此类推。如：



使用示例




```
LD      M0
GBIN    D0 D10
LD      M1
DGBINP  D20 D30
```

1. 当 M0 为 ON 时，执行 GBIN 指令，将 D0 中的格雷码（65,336）转化成整数存放在 D10 中，D10=43,567。
2. 当 M1 由 OFF → ON 变化时，DGBINP 指令执行一次，将（D21，D20）中的格雷码（48,633）转化成整数存放在（D31，D30）中，（D31，D30）=54,958。

6.12.7 SEG : 字转换 7 段码指令

	SEG	P	S	D	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√	√	√
步长	16位：SEG (连续) 5步 32位： -----		SEGP (脉冲) 6步 -----		相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	WORD										●	●	●	●	●	●	●	●	*	●	●	●	●	●	●
D	WORD											●	●	●	●		●		*	●	●	●	●		

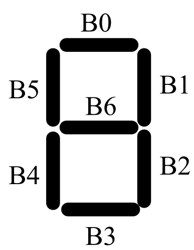
＊ 该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

├──┤ ├──[SEG S D]

将 S 单元中的数据转换成 7 段码显示的数据，结果赋予 D 单元中，使相应的 7 段码显示管点亮。

《7段码译码表》

源		7段码结构	预设定								显示数据
十六进制	二进制		B7	B6	B5	B4	B3	B2	B1	B0	
0	0000		0	0	1	1	1	1	1	1	0
1	0001		0	0	0	0	0	1	1	0	1
2	0010		0	1	0	1	1	0	1	1	2
3	0011		0	1	0	0	1	1	1	1	3
4	0100		0	1	1	0	0	1	1	0	4
5	0101		0	1	1	0	1	1	0	1	5
6	0110		0	1	1	1	1	1	0	1	6
7	0111		0	0	1	0	0	1	1	1	7
8	1000		0	1	1	1	1	1	1	1	8
9	1001		0	1	1	0	1	1	1	1	9
A	1010		0	1	1	1	0	1	1	1	A
B	1011		0	1	1	1	1	1	0	0	b
C	1100		0	0	1	1	1	0	0	1	C
D	1101		0	1	0	1	1	1	1	0	d
E	1110		0	1	1	1	1	0	0	1	E
F	1111		0	1	1	1	0	0	0	1	F

使用示例

```
   M0                    12          57           LD     M0
├──┤■├──[ SEG            D0          D10     ]    SEG    D0 D10
```

当 M0 为 ON 时，执行 SEG 指令，将 D0=2#1100（12）转换成 7 段码显示的数据，结果赋给 D10，D10=2#0111001（57），7 段码显示管 B5、B4、B3、B0 段点亮，显示 16#C。

注意事项

S 的有效范围为 0 ≤ S ≤ 15，当 S>15 时，PLC 信息中报“指令操作数数值非法”错误，指令不执行，D 单元中内容不变。

6.12.8 ASC： ASCII 码转换指令

	ASC	P	S1~S8 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：ASC (连续) 19步 ASCP (脉冲) 20步 32位： ----- -----			相关元件	SM186			

操作数	数据类型	位软元件									字软元件													变址	字符串
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			
S1～S8																									*
D	WORD																•		*	•	•	•		•	

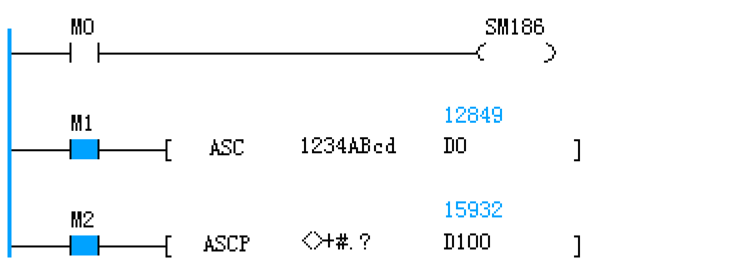
* 操作数 S1 ～ S8 是由计算机输入的 8 个或 8 个以下的 ASCII 码字符（包括数字，字母，特殊字符）组成的字符串。
* 操作数 D 使用 C 元件时，只能使用 C0 ～ C199。

指令说明

├──┤ ├──[ASC S1～S8 D]

1．将 8 个或 8 个以下的 ASCII 码字符（包括数字，字母，特殊字符）组成的字符串转化为 ASCII 码，并将转化结果赋予以 D 开始的单元序列中。
2．仅支持 ASCII 码为 16#21 ～ 16#7E 的字符（键盘输入，不足 8 个字符用 16#00 补充）。
3．当 SM186=OFF 时，每个 D 元件的高低字节分别存储 1 个 ASCII 码数据；
当 SM186=ON 时，每个 D 元件只用低字节存储 1 个 ASCII 码数据。
4．ASC II 码表请详见附录四 *ASC* II 码字符编码表。

使用示例




```
LD    M0
OUT   SM186
LD    M1
ASC   1234ABcd D0
LD    M2
ASCP  <>+#.? D100
```

1．当 M1 为 ON 时，执行 ASC 指令，将字符串（1234ABcd）进行 ASCII 码转换，并将结果存储在 D0 开始的 单元序列中。根据数据存储方式，产生结果如下：
若 SM186 为 OFF，执行结果为：D0=16#3231，D1=16#3433，D2=16#4241，D3=16#6463。
若 SM186 为 ON， 执 行 结 果 是：D0=16#31，D1=16#32，D2=16#33，D3=16#34，D4=16#41，D5=16#42
D6=16#63，D7=16#64。
2．当 M1 由 OFF → ON 变化时，ASCP 指令执行一次，将字符串（<>+#.?）进行 ASCII 码转换，并将结果存储在 D100 开始的单元序列中。根据数据存储方式，产生结果如下：
若 SM186 为 OFF，执行结果是：D100=16#3e3c，D101=16#232b，D102=16#3f2e。
若 SM186 为 ON，执行结果是：D100=16#3c，D101=16#3e，D102=16#2b，D103=16#23，D104=16#2e，D105=16#3f。

6.12.9 ITA : 16 进制数转换 ASCII 码指令

	ITA	P	S1 D S2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：ITA (连续) 7步 ITAP (脉冲) 8步 32位： ----- -----			相关元件	SM186			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	WORD										•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
D	WORD											•	•	•	•		•		•	•	•	•	•		
S2	WORD										•	•	•	•	•	•	•	•	•	•	•	•	•	•	

指令说明

├──┤ ├──[ITA S1 D S2]

S1：要转换的源 16 进制数据

D：目的操作数

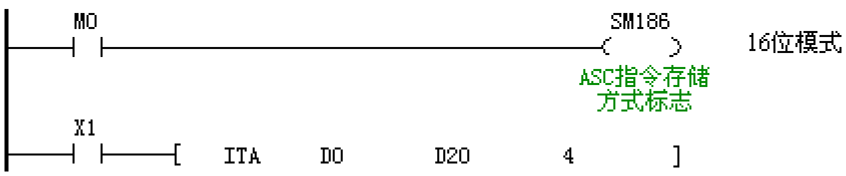
S2：ASCII 码数量（1 ≤ S2 ≤ 256）

1．将 S1 单元开始的 16 进制数转换成 S2 个 ASCII 码，结果赋给 D 开始的单元。
2．当 SM186（ASC 指令存储标志）为 OFF 时，每个 D 元件高低字节分别存储 1 个 ASCII 码数据；当 SM186 为 ON 时，每个 D 元件只用低字节存储 1 个 ASCII 码数据。
3．ASCII 码表请详见附录四 ASCII 码字符编码表。

使用示例

示例 1．16 位转换模式，SM186=OFF

D 分低 8 位，高 8 位来保存 ASCII 数据。



D0 开始的软元件中的数据为：D0=16#0123 D1=16#4567 D2=16#89AB

D \ S2	1	2	3	4	5	6	7	8	9
D20低	[3]	[2]	[1]	[0]	[7]	[6]	[5]	[4]	[B]
D20高		[3]	[2]	[1]	[0]	[7]	[6]	[5]	[4]
D21低			[3]	[2]	[1]	[0]	[7]	[6]	[5]
D21高				[3]	[2]	[1]	[0]	[7]	[6]
D22低					[3]	[2]	[1]	[0]	[7]
D22高						[3]	[2]	[1]	[0]
D23低			不变化				[3]	[2]	[1]
D23高								[3]	[2]
D24低									[3]

S2=4 时的位结构：

D0=16#0123

0	0	0	0	0	0	0	1	0	0	1	0	0	0	1	1
0				1				2				3			

D20

0	0	1	1	0	0	0	1	0	0	1	0	0	0	1	1

1 → 16#31　　0 → 16#30

D21

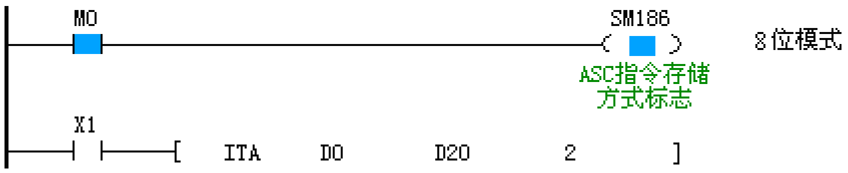
0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0

3 → 16#33　　2 → 16#32

ASCII 码：[0]=16#30　[1]=16#31　[2]= 16#32　[3]= 16#33
[4]=16#34　[5]=16#35　[6]= 16#36　[7]= 16#37
[8]=16#38　[9]=16#39　[A]= 16#41　[B]= 16#42

示例 2．8 位转换模式，SM186=ON

D 的高 8 位为 0，低 8 位来保存 ASCII 数据。



D0 开始的软元件中的数据为：D0=16#0123　D1=16#4567　D2=16#89AB

D \ S2	1	2	3	4	5	6	7	8	9
D20	[3]	[2]	[1]	[0]	[7]	[6]	[5]	[4]	[B]
D21		[3]	[2]	[1]	[0]	[7]	[6]	[5]	[4]
D22			[3]	[2]	[1]	[0]	[7]	[6]	[5]
D23				[3]	[2]	[1]	[0]	[7]	[6]
D24					[3]	[2]	[1]	[0]	[7]
D25						[3]	[2]	[1]	[0]
D26			不变化				[3]	[2]	[1]
D27								[3]	[2]
D28									[3]

S2=2 时的位结构：

D0=16#0123

0	0	0	0	0	0	0	1	0	0	1	0	0	0	1	1

0　1　2　3

D20=2 的 ASCII 码 =16#32

0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0

3　2

D21=3 的 ASCII 码 =16#33

0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	1

3　3

ASCII 码：[0]=16#30　[1]=16#31　[2]= 16#32　[3]= 16#33
[4]=16#34　[5]=16#35　[6]= 16#36　[7]= 16#37
[8]=16#38　[9]=16#39　[A]= 16#41　[B]= 16#42

注意事项

1．S1、D 使用 Kn 寻址时，n=4。

2．当 S2 不在 1 ～ 256 之间时，PLC 信息中报“指令操作数数值非法错误”，不执行转换，D 内容不变。

6.12.10 ATI：ASCII 码转换 16 进制数指令

	ATI	P	S1 D S2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：ATI (连续) 7步 32位： -----		ATIP (脉冲) 8步 -----	相关元件	SM186			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	WORD										•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
D	WORD											•	•	•	•		•		•	•	•	•	•		
S2	WORD										•	•	•	•	•	•	•	•	•	•	•	•	•	•	

指令说明

├──┤ ├──[ATI S1 D S2]

S1：待转换源 ASCII 码数据（16#30 ≤ S1 ≤ 16#39 或 16#41 ≤ S1 ≤ 16#46）

D：目的操作数

S2：ASCII 码数量（1 ≤ S2 ≤ 256）

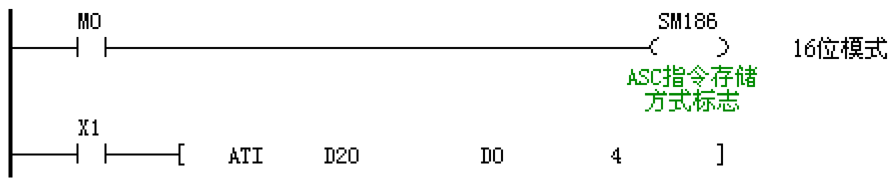
1．将 S1 单元开始的 S2 个 ASCII 码数据转换成 16 进制数据，结果每 4 位保存在 D 开始的单元中。

2．当 SM186（ASC 指令存储方式标志）为 OFF 时，S1 中的每个字元件高低字节都有效，当 SM186=ON 时，S1 中的每个字元件只有低字节有效。

使用示例

示例 1．16 位转换模式，SM186=OFF

将保存在 S1 的高低各 8 位（字节）中的 ASCII 码转换成 16 进制数。

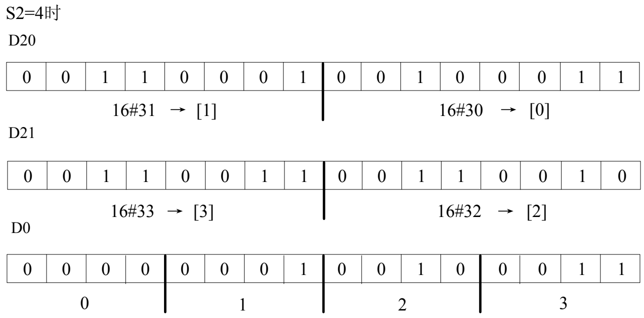


转换源数据

S1	ASCII码（16进制）	ATI转换
D20低	30	0
D20高	31	1
D21低	32	2
D21高	33	3
D22低	34	4
D22高	35	5
D23低	36	6
D23高	37	7
D24低	38	8

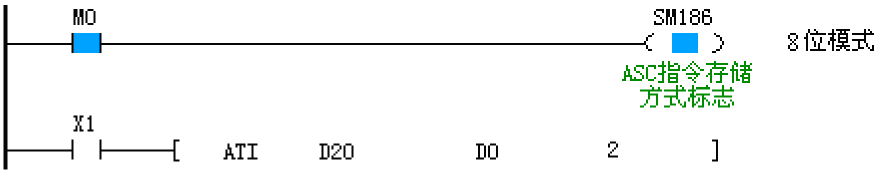
指定字符数及转换结果 [•] 为 0

S2 \ D	D2	D1	D0
1			16#•••0
2			16#••01
3	不变化		16#•012
4			16# 0123
5		16#•••0	16# 1234
6		16#••01	16# 2345
7		16#•012	16# 3456
8		16# 0123	16# 4567
9	16#•••0	16# 1234	16# 5678



示例 2．8 位转换模式，SM186=OFF

将保存在 S1 的低 8 位（字节）中的 ASCII 码转换成 16 进制数据。

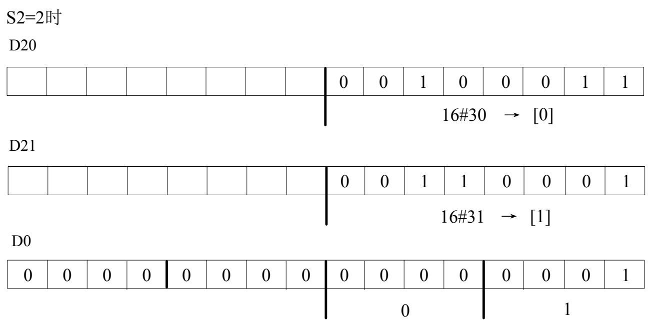


转换源数据

S1	ASCII码（16进制）	ATI转换
D20	30	0
D21	31	1
D22	32	2
D23	33	3
D24	34	4
D25	35	5
D26	36	6
D27	37	7
D28	38	8

指定字符数及转换结果　　[•] 为 0

S2 \ D	D2	D1	D0
1			16#•••0
2			16#••01
3	不变化		16#•012
4			16# 0123
5		16#•••0	16# 1234
6		16#••01	16# 2345
7		16#•012	16# 3456
8		16# 0123	16# 4567
9	16#•••0	16# 1234	16# 5678



注意事项

1．S1、D 使用 Kn 寻址时，n=4。

2．当 S1 不在 16#30 ～ 16#39 或 16#41 ～ 16#46 之间或 S2 不在 1 ～ 256 之间时，PLC 信息中报“指令操作数数值非法错误”，指令不执行，D 内容不变。

6.12.11 LCNV：工程转换指令

	LCNV		S1 S2 D S3	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	9步			相关元件				

操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT																●				●				
S2	INT																●				●				
D	INT																●				●				
S3	WORD																●				●			●	

指令说明

LCNV　S1　S2　D　S3

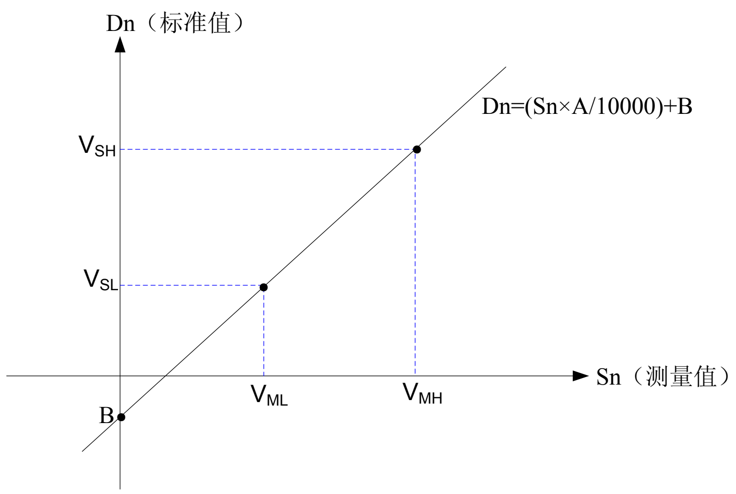
S1：要转换的源操作数起始地址　　**S2**：转换表格起始地址

D：存放转换结果起始地址　　**S3**：要转换数据个数（1 ≤ S3 ≤ 64）

1. 当使用模拟量输入模块读取外界模拟信号时，可以利用本指令将原始模拟量读值转换为相对应的工程读值。
2. 当使用模拟量模块来做模拟量测量时，如果 PLC 所测量的模拟量值与相关标准仪表所测量的结果有偏差时，可利用本指令作线性修正以作为实际测量之校正。
3. 在转换表格内填入低点测量值 V_{ML}、高点测量值 V_{MH} 及其对应的低点标准值 V_{SL} 与高点标准值 V_{SH} 共四个参数，执行线性转换时，来源数据经由如下公式运算产生与之对应的目标标准值。其中 S_n 为源输入数据，D_n 为转换结果数据。

$$A=(V_{SL}-V_{SH})/(V_{ML}-V_{MH})\times 10000$$
$$B=V_{SL}-(V_{ML}\times A/10000)$$
$$D_n=(S_n\times A/10000)+B$$

标准值与测量值对应特性图如下：

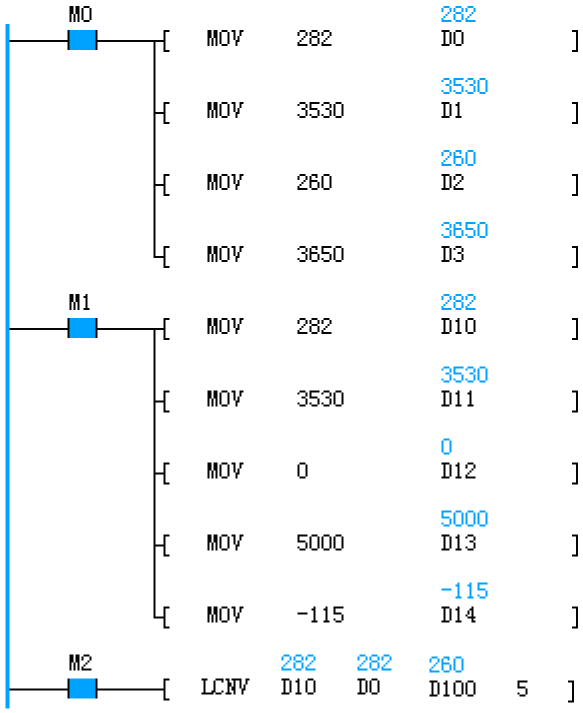


使用示例

如下表：

测量值	V_{ML}=282	V_{MH}=3530
标准值	V_{SL}=260	V_{SH}=3650

根据上图中的四个参数，执行线性转换。




```
LD    M0
MOV   282 D0
MOV   3530 D1
MOV   260 D2
MOV   3650 D3
LD    M1
MOV   282 D10
MOV   3530 D11
MOV   0 D12
MOV   5000 D13
MOV   -115 D14
LD    M2
LCNV  D10 D0 D100 5
```

1．当 M0 为 ON 时，给 D0 ～ D3 单元赋值。
2．当 M1 为 ON 时，将要转换的数据（PLC 从模拟量模块读取到的数据）赋予 D10 ～ D14 单元中。
3．当 M2 为 ON 时，执行 LCNV 转换，将 D10 ～ D14 单元中的数据，根据 D0 ～ D3 单元中的数据所建立的线性关系进行转换，产生结果如下：
D100＝260　D101＝3650　D102＝－34　D103＝5184　D104＝－154

注意事项

1．转换表格内的四个数据是有实际意义的，如低点测量值应小于高点测量值。
2．转换结果若超出符号整数（INT）的范围，结果将不准确。D_n 若大于 32,767 则为 32,767，若小于－32,768 则为－32,768。

6.12.12 RLCNV：浮点工程转换指令

	RLCNV		S1 S2 D S3	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√	√
步长	12步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	REAL																●				●				
S2	REAL																●				●				
D	REAL																●				●				
S3	REAL																●				●			●	

指令说明

├──┤ ├──[RLCNV S1 S2 D S3]

S1：要转换的源操作数起始地址　**S2**：转换表格起始地址

D：存放转换结果起始地址　**S3**：要转换数据个数（1 ≤ S3 ≤ 64）

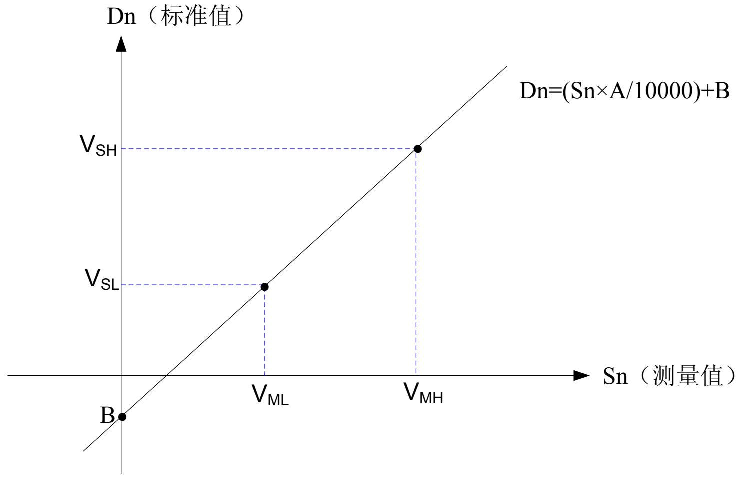
1. 当使用模拟量输入模块读取外界模拟信号时，可以利用本指令将原始模拟量读值转换为相对应的工程读值。
2. 当使用模拟量模块来做模拟量测量时，如果 PLC 所测量的模拟量值与相关标准仪表所测量的结果有偏差时，可利用本指令作线性修正以作为实际测量之校正。
3. 在转换表格内填入低点测量值 V_{ML}、高点测量值 V_{MH} 及其对应的低点标准值 V_{SL} 与高点标准值 V_{SH} 共四个参数，执行线性转换时，来源数据经由如下公式运算产生与之对应的目标标准值。其中 S_n 为源输入数据，D_n 为转换结果数据。

$$A=(V_{SL}-V_{SH})/(V_{ML}-V_{MH})\times 10000$$

$$B=V_{SL}-(V_{ML}\times A/10000)$$

$$D_n=(S_n\times A/10000)+B$$

标准值与测量值对应特性图如下：

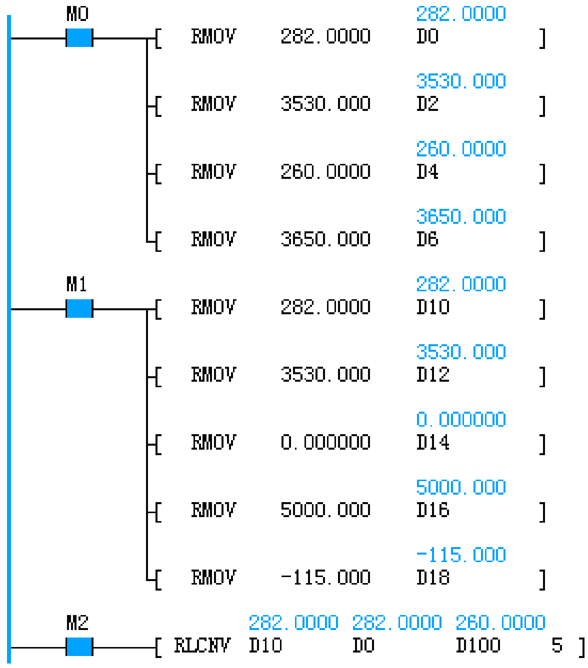


使用示例

如下表：

测量值	V_{ML}=282.0000	V_{MH}=3530.0000
标准值	V_{SL}=260.0000	V_{SH}=3650.0000

根据上图中的四个参数，执行线性转换。




```
LD     M0
RMOV   282.0000 D0
RMOV   3530.000 D2
RMOV   260.0000 D4
RMOV   3650.000 D6
LD     M1
RMOV   282.0000 D10
RMOV   3530.000 D12
RMOV   0.000000 D14
RMOV   5000.000 D16
RMOV   -115.000 D18
LD     M2
RLCNV  D10 D0 D100 5
```

1. 当 M0 为 ON 时，给 D0 ～ D3 单元赋值。
2. 当 M1 为 ON 时，将要转换的数据（PLC 从模拟量模块读取到的数据）赋予 D10 ～ D19 单元中。
3. 当 M2 为 ON 时，执行 RLCNV 转换，将 D10 ～ D19 单元中的数据，根据 D0 ～ D7 单元中的数据所建立的线性关系进行转换，产生结果如下：
   D100 = 260.0000　D102 = 3650.000　D104 = － 34.328　D106 = 5184.267　D108 = － 154.357

注意事项

1. 转换表格内的四个数据是有实际意义的，如低点测量值应小于高点测量值。
2. 转换结果若超出符号浮点数（REAL）的范围，结果将不准确。D_n 若大于 3.402823e+038 则为 3.402823e+038，若小于 － 3.402823e+038 则为 － 3.402823e+038。

6.13 通讯指令

6.13.1 MODBUS : MODBUS 主站通讯指令

	MODBUS	S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
				√	√	√	√
步长	8步		相关元件	SM124 SM134～SM140 SD140～SD142 SD145～SD147 SD149～SD152 SD154～SD157 SD159			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	WORD																							*	
S2	WORD																●				●		●		
D	WORD																●				●		●		

* S1 只能使用常数 1 ～ 3。

指令说明

```
├──┤ ├──[ MODBUS  S1  S2  D ]
```

S1：指定通讯通道（范围 1 ～ 3） **S2**：发送数据起始地址 **D**：接收数据起始地址

1. 作为主站，调用该指令，把从 S2 开始保存的数据发送出去，然后接收数据，并保存到 D 开始的地址单元中。
2. 作为从站，接收和发送数据不需要指令控制。
3. SM124/SM134 分别为 PORT1/PORT2 的串口空闲标志，可以用在 MODBUS 中标识 MODBUS 的通讯状态。
4. MODBUS 接收到数据后，进行 CRC 校验、地址校验、功能码校验，如果有错误，置错误标志（SM136 或 SM138），并将具体的错误信息记录在特殊寄存器 SD149 或 SD154 中。
5. 该指令一般使用沿检测指令驱动。

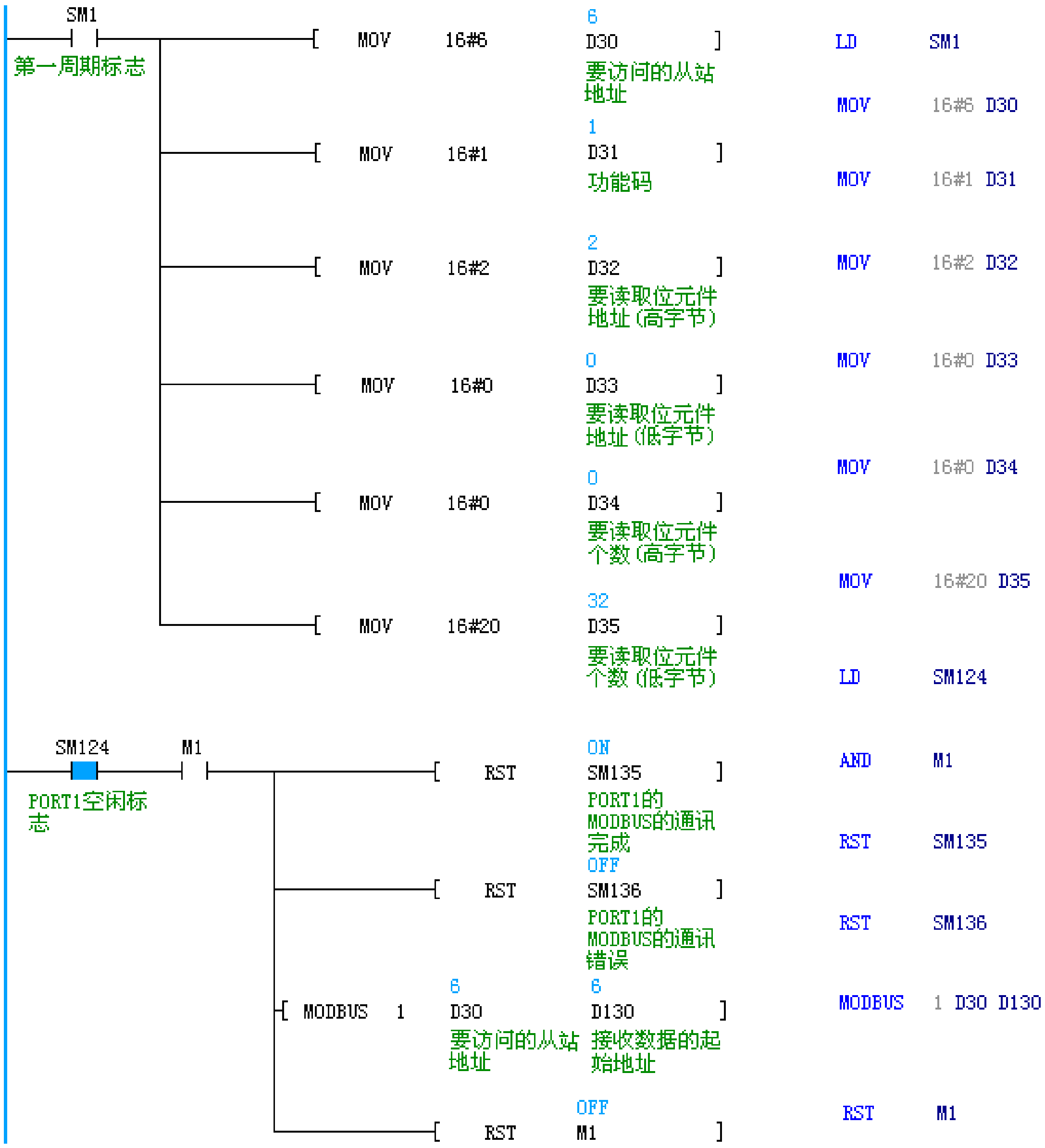
MODBUS 通讯错误代码表如下：

异常代码	异常代码意义
16#01	非法功能码
16#02	非法寄存器地址
16#03	数据个数错误
16#10	通讯超时，通讯时间超过了用户设定的通讯最大时间发生的错误
16#11	接收数据帧错误
16#12	设定主 / 从站错误
16#13	本身的站号与指令设定的站号相同，发生错误
16#14	元件地址溢出（接收或发送的数据量超出元件存储空间）

详细应用方法举例请参见第十章 通讯功能使用指南。

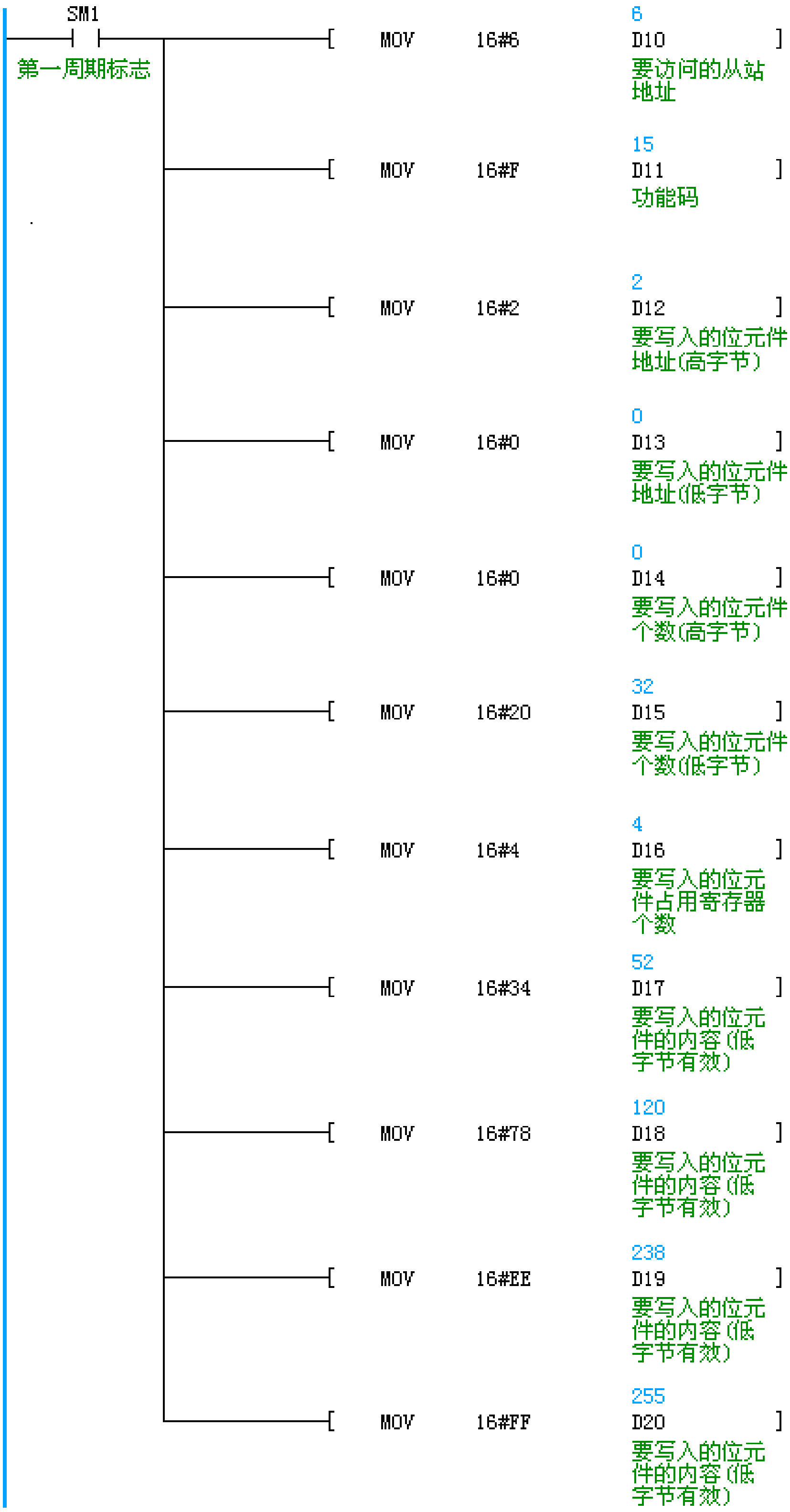
使用示例

示例 1：MODBUS 指令读线圈

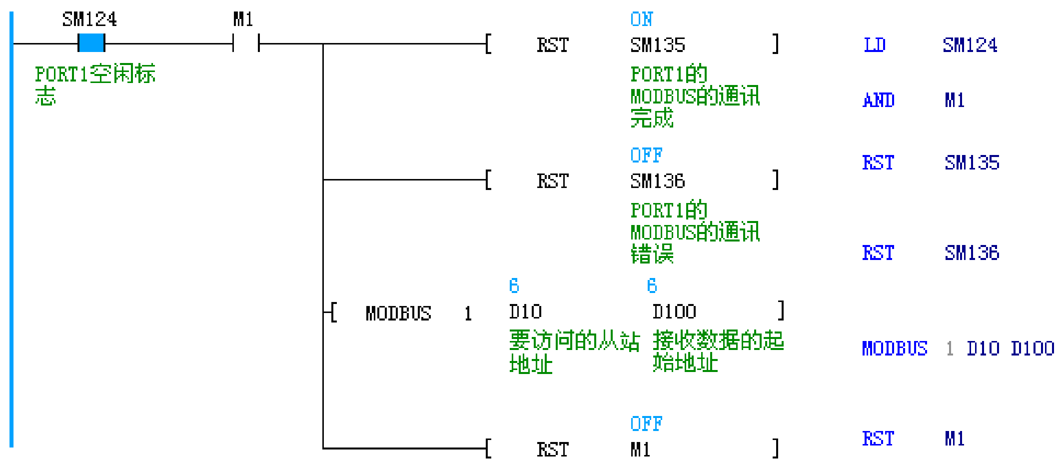


1. 初始上电第一周期给 D30 开始的单元序列赋值：访问从站地址为 6（保存在 D30 中），功能码为 1（指定读线圈功能，保存在 D31 中），要读取的位元件地址为 512（Y0 的协议地址，分高低字节保存在 D32/D33 中），要读取的位元件个数为 32（分高低字节保存在 D34/D35 中）。
2. 在 SM124 为 ON（PORT1 空闲）的前提下，当 M1 为 ON 时，复位 SM135（PORT1 的 MODBUS 的通讯完成）及 SM136（PORT1 的 MODBUS 的通讯错误）标志位，同时执行 MODBUS 指令。
3. 指令执行后接收数据保存到 D130 开始的单元序列中：D130=6（访问的从站地址），D131=1（功能码），D132=4（接收的数据占用了 4 个字节）。D133=255，D134=255，D135=255，D136=255，（读取到的从站的位元件），D137 和 D138 存放校验码。

示例 2： MODBUS 指令写线圈




```
LD   SM1
MOV  16#6 D10
MOV  16#F D11
MOV  16#2 D12
MOV  16#0 D13
MOV  16#0 D14
MOV  16#20 D15
MOV  16#4 D16
MOV  16#34 D17
MOV  16#78 D18
MOV  16#EE D19
MOV  16#FF D20
```



1．初始上电第一周期给 D10 开始的单元序列赋值：访问从站地址为 6（保存在 D10 中），功能码为 15（指定写多个线圈功能，保存在 D11 中），要写入的 位元件地址为 512（Y0 协议地址，分高低字节保存在 D12/D13 中），要写入的位元件个数为 32（分高低字节保存在 D14/D15 中），要写入的位元件占用寄存器个数为 4（32 个位元件占用 4 个寄存器，保存到 D16 中），要写入位元件的内容为 16#34,16#78,16#EE，16#FF，（分别保存在 D17/D18 /D19/D20 的低字节中）。

2．在 SM124 为 ON（PORT1 空闲）的前提下，当 M1 为 ON 时，复位 SM135（PORT1 的 MODBUS 的通讯完成）及 SM136（PORT1 的 MODBUS 的通讯错误）标志位，同时执行 MODBUS 指令。

3．指令执行后接收数据保存到 D100 开始的单元序列中：

元件地址	元件值	存放内容	元件地址	元件值	存放内容
D100	6	访问的从站地址	D104	0	写入的寄存器个数，高字节
D101	15	功能码	D105	32	写入的寄存器个数，低字节
D102	2	Y0 的协议地址，高字节	D106	84	校验码，高字节
D103	0	Y0 的协议地址，低字节	D107	28	校验码，低字节

注意事项

1. MODBUS 发送数据，用户设置成 RTU 模式，只需从 S2 开始保存 RTU 模式的数据，不需要保存起始字符，结束字符和校验和。发送过程中，自动加上所需的起始字符，结束字符和校验和。

2. 发送的数据，不需设定发送的数据长度，系统会根据指令自动按系统内部设定长度进行发送，如下图所示

S2	从站地址
S2+1	功能码
S2+2	数据1
⋮	……
S2+N+1	数据N

3．发送和接收的数据都是保存在字元件的低字节中，高字节没有使用。

4．详细介绍请参见第十章 通讯功能使用指南。

6.13.2 MODRD：MODBUS 读指令

	MODRD	S1 S2 S3 S4 D	适用机型	MX1H	MX2H	MX2V	MX3H
				√	√	√	√
步长	12步		相关元件	SM124 SM134～SM140 SD140～SD142 SD145～SD147 SD149～SD152 SD154～SD157 SD159			

操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z				Un.b
S1	WORD																								*	
S2	WORD										•	•	•	•	•	•	•	•	*	•	•	•		•	*	•
S3	WORD										•	•	•	•	•	•	•	•	*	•	•	•		•	•	•
S4	WORD										•	•	•	•	•	•	•	•	*	•	•	•		•	•	•
D	WORD			•													•									

＊ S1 只能使用常数 1 ～ 3。

＊ S2 使用常数时，只能使用 1 ～ 31。

＊ 使用 C 元件时，只能使用 C0 ～ C199。

指令说明

├──┤ ├──[MODRD S1 S2 S3 S4 D]

S1：指定通讯通道（范围 1 ～ 3） **S2**：从站站号（范围 1 ～ 31） **S3**：MODBUS 起始地址

S4：要读取的元件个数 **D**：接收数据起始地址

1．作为主站，调用该指令，将从站站号为 S2，起始协议地址为 S3 开始的 S4 个数据读取过来，保存到 D 开始的地址单元序列中。

2. 读取的元件类型由接收数据起始地址决定，若接收数据起始地址为 M 元件，则读取的元件类型为位元件；若接收数据起始地址为 D 元件，则读取的元件类型为字元件。

3．作为从站，被读取数据不需要指令控制。

4．SM124/SM134 分别为 PORT1/PORT2 的串口空闲标志，可以用在 MODBUS 中标识 MODBUS 的通讯状态。

5．MODBUS 接收到数据后，进行 CRC 校验、地址校验、功能码校验，如果有错误，置错误标志（SM136 或 SM138），并将具体的错误信息记录在特殊寄存器 SD149 或 SD154 中。

6．该指令一般使用沿检测指令驱动。

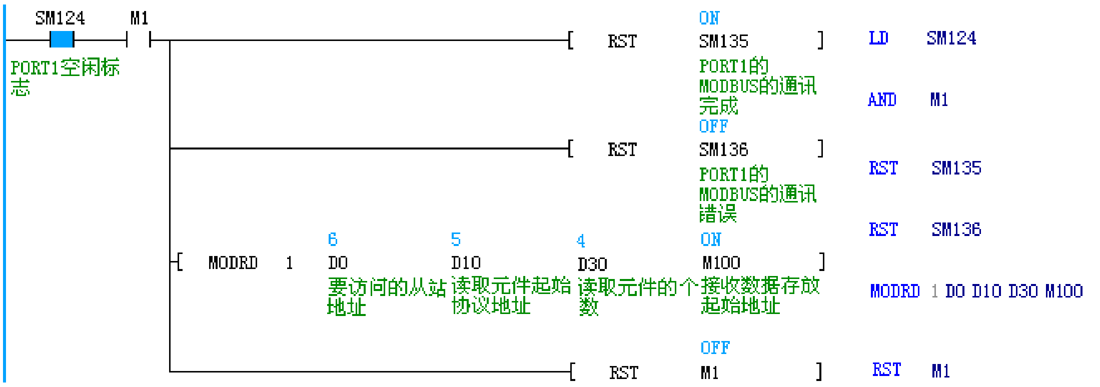
MODBUS 通讯错误代码表如下：

异常代码	异常代码意义
16#02	非法寄存器地址
16#03	数据个数错误
16#10	通讯超时，通讯时间超过了用户设定的通讯最大时间发生的错误
16#11	接收数据帧错误
16#12	设定主 / 从站错误
16#13	本身的站号与指令设定的站号相同，发生错误
16#14	元件地址溢出（接收或发送的数据量超出元件存储空间）

详细应用方法举例请参见第十章 通讯功能使用指南。

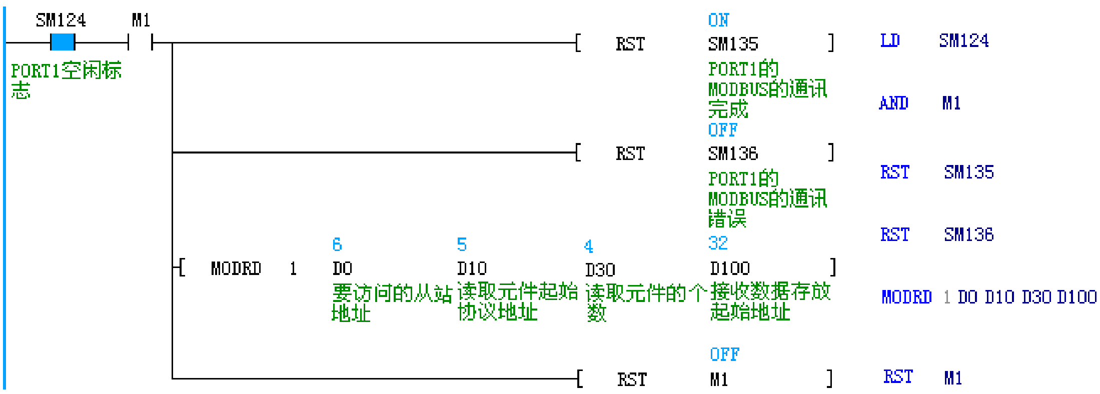
使用示例

示例 1：MODRD 指令读线圈



1. 访问从站地址为 6（保存在 D0 中），要读取的位元件起始协议地址为 5（X5 的协议地址，保存在 D10 中），要读取的位元件个数为 4（保存在 D30 中），接收数据存放起始地址为 M100（决定了读取的元件为位元件）。
2. 在 SM124 为 ON（PORT1 空闲）的前提下，当 M1 为 ON 时，复位 SM135（PORT1 的 MODBUS 的通讯完成）及 SM136（PORT1 的 MODBUS 的通讯错误）标志位，同时执行 MODRD 指令。
3. 指令执行后将 6 号站的 X5 ～ X10 的数值读取到主站的 M100 ～ M104 单元中。

示例 2：MODRD 指令读字元件



1. 访问从站地址为 6（保存在 D0 中），要读取的字元件起始协议地址为 5（D5 的协议地址，保存在 D10 中）要读取的字元件个数为 4（保存在 D30 中），接收数据存放起始地址为 D100（决定了读取的元件为字元件）。
2. 在 SM124 为 ON（PORT1 空闲）的前提下，当 M1 为 ON 时，复位 SM135（PORT1 的 MODBUS 的通讯完成）及 SM136（PORT1 的 MODBUS 的通讯错误）标志位，同时执行 MODRD 指令。
3. 指令执行后将 6 号站的 D5 ～ D8 的数值读取到主站的 D100 ～ D104 单元中。

注意事项

1. MODRD（MODBUS 读指令），用户设置成 RTU 模式与外围设备通讯。
2. 读取双字元件时，S4=2× 要读取的双字元件个数。例如：要读取 2 个双字元件，S4 应设为 4。
3. MODRD 内部使用的是 03 功能码，若用户只能使用 04 功能码，则建议使用通用通用 Modbus 指令。
4. 详细介绍请参见第十章 通讯功能使用指南。

6.13.3 MODWR：MODBUS 写指令

	MODWR	S1 S2 S3 S4 D	适用机型	MX1H	MX2H	MX2V	MX3H
				√	√	√	√
步长	12步		相关元件	SM124 SM134～SM140 SD140～SD142 SD145～SD147 SD149～SD152 SD154～SD157 SD159			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	WORD																							*	
S2	WORD										●	●	●	●	●	●	●	●	*	●	●	●	●	*	●
S3	WORD										●	●	●	●	●	●	●	●	*	●	●	●	●	●	●
S4	WORD										●	●	●	●	●	●	●	●	*	●	●	●	●	●	●
D	WORD			●													●								

* S1 只能使用常数 1 ～ 3。
* S2 使用常数时，只能使用 1 ～ 31。
* 使用 C 元件时，只能使用 C0 ～ C199。

指令说明

├──┤ ├──[MODWR S1 S2 S3 S4 D]

S1：指定通讯通道（范围 1 ～ 3） **S2**：从站站号（范围 1 ～ 31） **S3**：MODBUS 起始地址
S4：要写入的元件个数 **D**：写数据起始地址

1. 作为主站，调用该指令，将 D 开始的 S4 个单元中的数据写入到站号为 S2，起始协议地址为 S3 的元件中。
2. 写入的元件类型由写数据起始地址决定，若写数据起始地址设为 M 元件，则要写入的元件类型为位元件；若写数据起始地址为 D 元件，则要写入的元件类型为字元件。
3. 作为从站，被写入数据不需要指令控制。
4. SM124/SM134 分别为 PORT1/PORT2 的串口空闲标志，可以用在 MODBUS 中标识 MODBUS 的通讯状态。
5. MODBUS 接收到数据后，进行 CRC 校验、地址校验、功能码校验，如果有错误，置错误标志（SM136 或 SM138），并将具体的错误信息记录在特殊寄存器 SD149 或 SD154 中。
6. 操作数 S2 为 0 时为广播模式，主站向所有从站发送数据。
7. 该指令一般使用沿检测指令驱动。

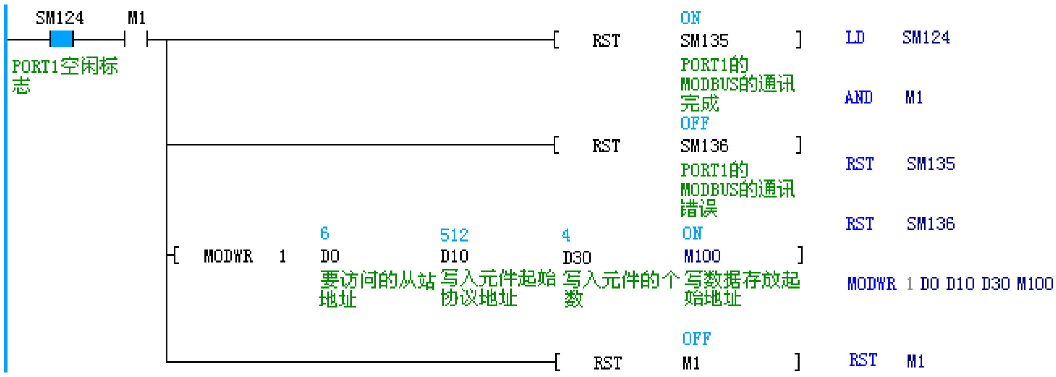
MODBUS 通讯错误代码表如下：

异常代码	异常代码意义
16#02	非法寄存器地址
16#03	数据个数错误
16#10	通讯超时，通讯时间超过了用户设定的通讯最大时间发生的错误
16#11	接收数据帧错误
16#12	设定主 / 从站错误
16#13	本身的站号与指令设定的站号相同，发生错误
16#14	元件地址溢出（接收或发送的数据量超出元件存储空间）

详细应用方法举例请参见第十章 通讯功能使用指南。

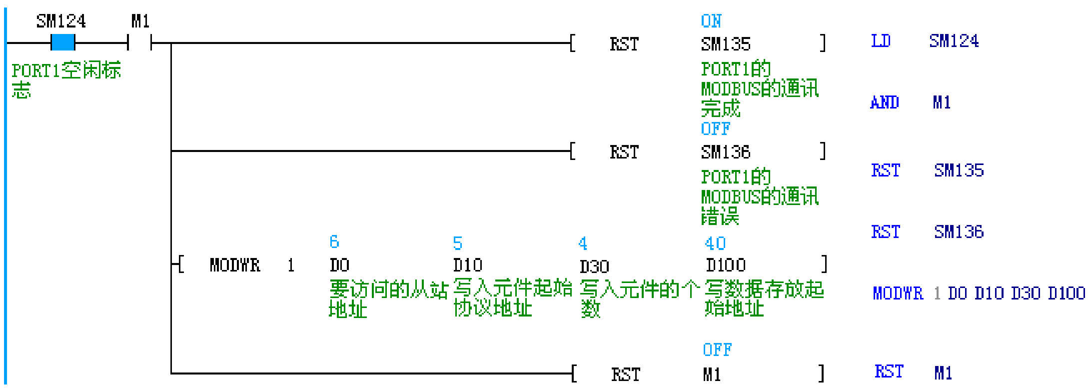
使用示例

示例 1：MODWR 指令写位元件



1. 访问从站地址为 6（保存在 D0 中），要写的位元件起始协议地址为 512（Y0 的协议地址，保存在 D10 中），要写的位元件个数为 4（保存在 D30 中），写数据存放起始地址为 M100（决定了写入的元件为位元件）。
2. 在 SM124 为 ON（PORT1 空闲）的前提下，当 M1 为 ON 时，复位 SM135（PORT1 的 MODBUS 的通讯完成）及 SM136（PORT1 的 MODBUS 的通讯错误）标志位，同时执行 MODWR 指令。
3. 指令执行后将主站的 M100 ～ M103 的值写入到 6 号从站的 Y0 ～ Y3 中。

示例 2：MODWR 指令写字元件



1. 访问从站地址为 6（保存在 D0 中），要写的字元件起始协议地址为 5（D5 的协议地址，保存在 D10 中），要写的字元件个数为 4（保存在 D30 中），写的数据存放起始地址为 D100（决定了写入的元件为字元件）。
2. 在 SM124 为 ON（PORT1 空闲）的前提下，当 M1 为 ON 时，复位 SM135（PORT1 的 MODBUS 的通讯完成）及 SM136（PORT1 的 MODBUS 的通讯错误）标志位，同时执行 MODWR 指令。
3. 指令执行后将主站的 D100 ～ D103 的值写入到 6 号从站的 D5 ～ D8 中。

注意事项

1. MODWR（MODBUS 写指令），用户设置成 RTU 模式与外围设备通讯。
2. 写入双字元件时，S4=2× 要写入的双字元件个数。例如：要写入 2 个双字元件，S4 应设为 4。
3. MODWR 内部使用的是 03 功能码，若用户只能使用 04 功能码，则建议使用通用通用 Modbus 指令。
4. 详细介绍请参见第十章 通讯功能使用指南。

6.13.4 XMT : 自由口发送指令

	XMT		S1 S2 S3	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	7步			相关元件	SM110 SM112 SM114 SM130 SM132 SM134 SD120～SD124 SD128		SM120 SM122 SM124 SD110～SD114 SD118 SD130～SD134 SD138	

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT																							*	
S2	WORD																•				•		•		
S3	INT										•	•	•	•	•		•	•	*	•	•	•	•	•	

＊ S1 只能使用常数 0 ～ 2。

＊ S3 使用 C 元件时，只能使用 C0 ～ C199。

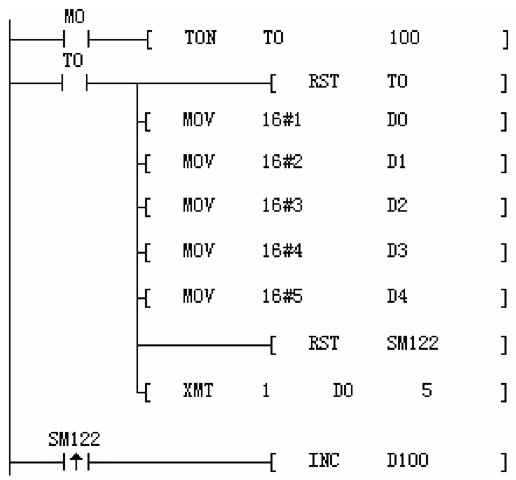
指令说明

├──┤ ├──[XMT S1 S2 S3]

S1：指定的通讯通道，取值范围为 0，1，2 **S2**：发送数据起始地址 **S3**：发送的字节数

在通讯条件满足时，按照用户指定的通道和地址发送数据。

使用示例



例程中是每隔 10s 发送一帧数据。利用串口 1 发送如下数据：

01	02	03	04	05

1．首先，在系统组态中将通讯口 1 设置为自由口，然后设定波特率、奇偶校验、数据位、停止位等。

2．将要发送的数据的低字节写到发送缓存区内。

3．发送数据之前先清发送完成标志（SM122）。

4．当发送完成，发送完成标志（SM122）置位。详细应用方法举例请参见第十章 通讯功能使用指南。

相关元件

1．SM110/SM120/SM130：PORT0/PORT1/PORT2 发送使能标志，当使用 XMT 指令时该位被置位，当发送结束后清除该位，当该位清零时，当前发送终止。

2．SM112/SM122/SM132：PORT0/PORT1/PORT2 发送完成标志，当判断发送完成的时候，发送完成标志置位。

3．SM114/SM124/SM134：PORT0/PORT1/PORT2 空闲标志，当串口没有通讯任务的时候置位，可以作为通讯的检测位。

4．SD110/SD120/SD130：PORT0/PORT1/PORT2 模式状态字

5．SD118/SD128/SD138：PORT0/PORT1/PORT2 当前发送的字符。

注意事项

1．通讯帧的大小：通讯帧根据所选用元件类型（D 或 V）的不同，发送帧的结束字符不能超过其有效范围。

2．停机的情况下，发送中止。

6.13.5 RCV：自由口接收指令

	RCV		S1 D S2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	7步			相关元件	SM111 SM113 SM114 SM121 SM123 SM124 SM131 SM133 SM134 SD110～SD117 SD120～SD127 SD130～SD137			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT																							*	
D	WORD																●				●		●		
S2	INT										●	●	●	●	●		●	●	*	●	●	●	●	●	

*　S1 只能使用常数 0 ～ 2。

*　S2 使用 C 元件时，只能使用 C0 ～ C199。

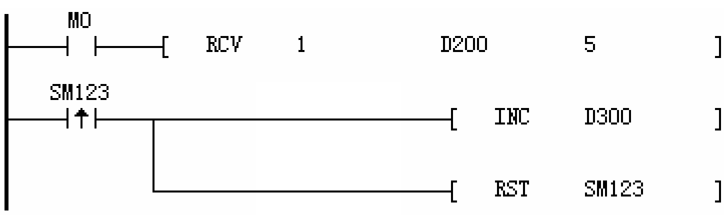
指令说明

[RCV　S1　D　S2]

S1：指定的通讯通道，取值范围为 0，1，2　**D**：存放接收数据的起始地址　**S2**：接收的最大字节数

在通讯条件满足时，按照用户指定的通道和地址的数据。

使用示例




```
LD   M0
RCV  1 D200 5
LDP  SM123
INC  D300
RST  SM123
```

当 M0 为 ON 时，RCV 指令将持续有效，如果只想接收一次，可以采用上升沿检测指令作为输入。详细应用方法举例请参见第十章 通讯功能使用指南。

相关元件

1．SM111/SM121/SM131：PORT0/PORT1/PORT2 接收使能标志，当使用 RCV 指令时该位被置位，当接收结束后清除该位，当该位清零时，当前接收终止。

2．SM113/SM123/SM133：PORT0/PORT1/PORT2 接收完成标志，当接收完成，接收完成标志置位。

3．SM114/SM124/SM134：PORT0/PORT1/PORT2 空闲标志，当串口没有通讯任务的时候置位，可以作为通讯的检测位。

4．SD111/SD121/SD131：PORT0/PORT1/PORT2 开始字符，可以在系统组态中设置。

5．SD112/SD122/SD132：PORT0/PORT1/PORT2 结束字符，可以在系统组态中设置。

6．SD113/SD123/SD133：PORT0/PORT1/PORT2 字符间超时时间，也就是接收两个字符间的最大间隔时间，可以在系统组态中设置。

7．SD114/SD124/SD134：PORT0/PORT1/PORT2 帧超时时间，从指令开始执行到接收数据结束的时间，可以在系统组态中设置。

SD115/SD125/SD135：PORT0/PORT1/PORT2 接收完成信息代码，数据位定义如下：

用户终止接收标志	收到指定结束字标志	收到最大字符数标志	字符间超时标志	（帧）接收超时标志	奇偶校验错误标志	保留，用户可忽略
第 0 位	第 1 位	第 2 位	第 3 位	第 4 位	第 5 位	第 6 ～ 15 位

SD116/SD126/SD136：PORT0/PORT1/PORT2 当前收到的字符。

SD117/SD127/SD137：PORT0/PORT1/PORT2 当前收到的字符总数。

注意事项

1．通讯帧的大小：通讯帧根据所选用元件类型（D 或 V）的不同，发送帧的结束字符不能超过其有效范围。

2．停机的情况下，发送中止。

6.13.6 CANXMT : CANbus 自由口单帧发送指令

	CANXMT	S1 S2 S3 S4 D	适用机型	MX1H	MX2H	MX2V	MX3H
				√	√	√	√
步长	13步		相关元件				

操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z				Un.b
S1	WORD																								*	
S2	DWORD																●				●			●	●	
S3	WORD																●				●			●	●	
S4	INT										●	●	●	●	●		●	●	*	●	●	●			●	
D	BOOL		●	●	●																					

* S1 只能使用常数 0。

* S4 使用 C 元件时，只能使用 C0 ～ C199。

指令说明

├──┤ ├──[CANXMT S1 S2 S3 S4 D]

S1：指定的通讯通道，只能使用常数 0　**S2**：发送数据 CANID　**S3**：发送数据起始地址

S4：发送的字节数　**D**：输出位元件对象

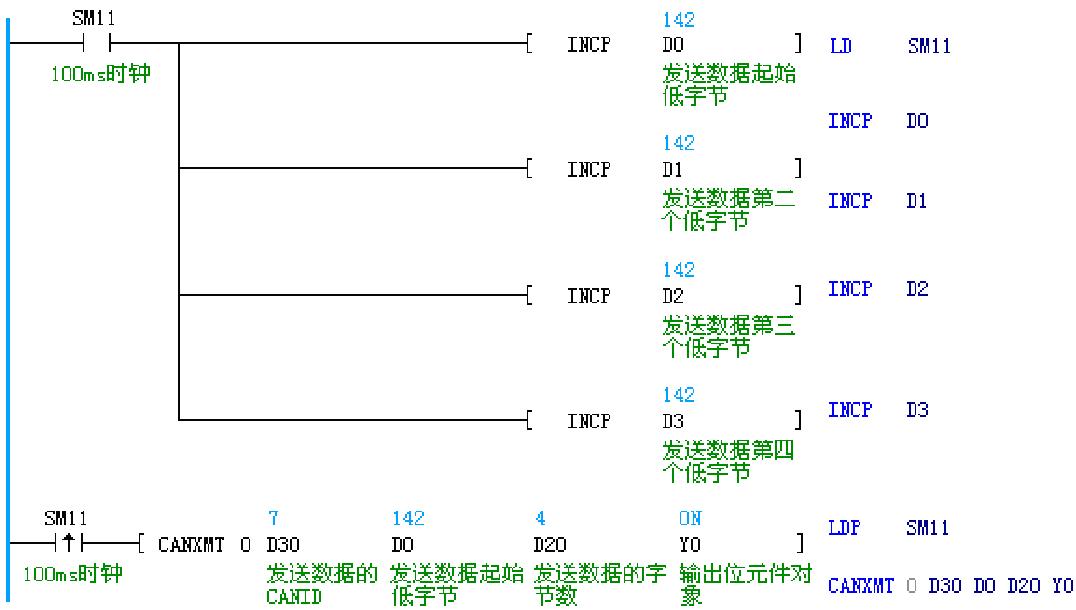
1．当通讯条件满足时，使用该指令可按照指定的通道和地址发送数据。通过 S1 通道，将 S3 单元开始的 S4 个低字节的内容发送到接收数据 CANID 为 S2 的站中，发送完成后 D 输出为 ON。

2．S2 在 CAN2.0A 下，有效位为低 11 位，在 CAN2.0B 下，有效位为低 29 位，其它位忽略。

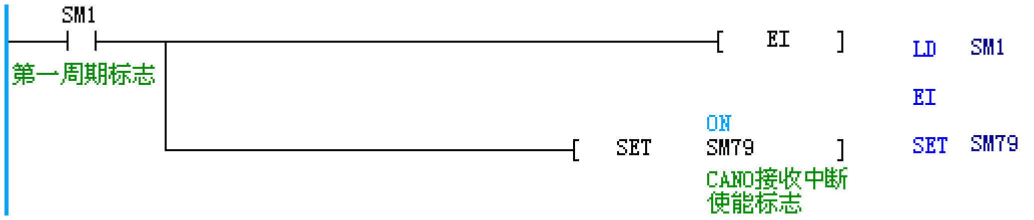
3．该指令一般使用沿检测指令驱动。

使用示例

发送站的程序：



接收站程序：



1．把要发送的数据保存在 D0 开始的单元序列中，因为在系统组态中已设定低字节有效，所以此处只有低字节有效。
2．当 SM11 由 OFF → ON 变化时，执行 CANXMT 指令。
3．通过 0 通道，从 D0 开始的 4 个单元的低字节发送到接收数据 CANID 为 7 的站中，发送完成后输出 Y0 为 ON 状态。
4．接收站可以通过开通 CAN0 接收中断来接收数据，接收的数据存放在 SD240 开始的单元中，也可以通过调用 CANRCV 指令来接收数据。需要注意的是，使用 CANRCV 指令接收数据时，必须要将发送站和接收的 CANID 设为一致。

注意事项

1．发送数据通过系统组态可以支持高低字节有效或低字节有效。
2．目的操作数 D 不可被其它指令或强制模式重复修改。
3．S4 的取值范围是大于等于 0。
4. 通讯帧的大小：通讯帧根据所选用的元件类型（D 或者 V）的不同，发送帧的结束字符不超过其有效范围，每一个 CAN 帧不能超过 255 个字节。
5．停机的情况下，发送中止。
6．网络不支持相同 CANID 的 CAN 帧同时出现；CANID 保留 0。

6.13.7 CANRCV : CANbus 自由口单帧接收指令

	CANRCV	S1 S2 S3 S4 D	适用机型	MX1H	MX2H	MX2V	MX3H
				√	√	√	√
步长	13步		相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	WORD																							*	
S2	DWORD																•				•		•	•	
S3	WORD																•				•		•	•	
S4	INT										•	•	•	•	•		•	•	*	•	•	•		•	
D	BOOL		•	•	•																				

* S1 只能使用常数 0。

* S4 使用 C 元件时，只能使用 C0 ～ C199。

指令说明

├──┤ ├──[CANRCV S1 S2 S3 S4 D]

S1：指定的通讯通道，只能使用常数 0 **S2**：发送数据 CANID **S3**：接收数据起始地址

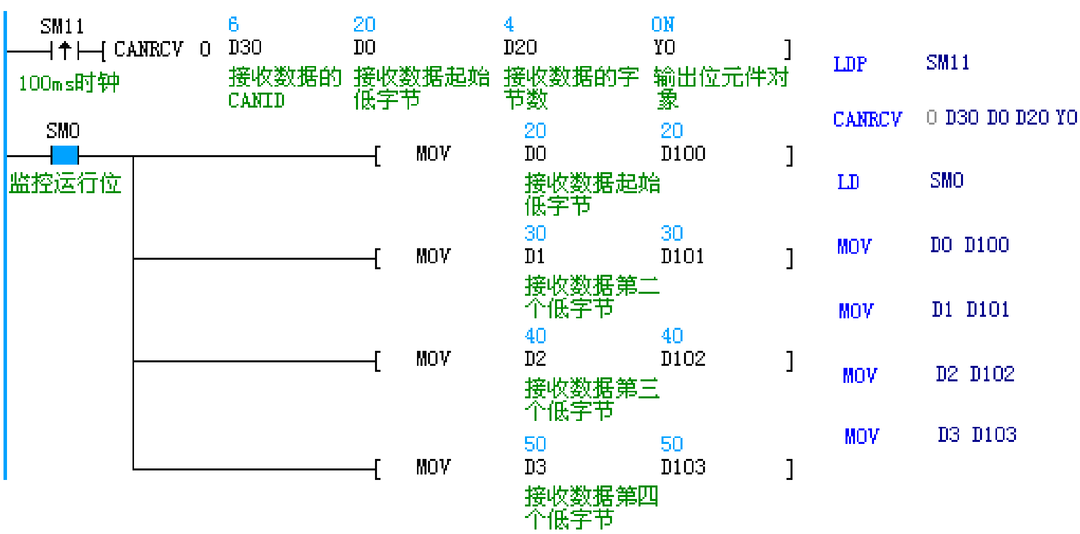
S4：发送的字节数（S4 ≥ 0） **D**：输出位元件对象

1．通过 S1 通道，接收数据保存到 S3 开始的单元序列中，接收完成后 D 输出为 ON。

2．S2 在 CAN2.0A 下，有效位为低 11 位，在 CAN2.0B 下，有效位为低 29 位，其它位忽略。

3．该指令一般使用沿检测指令驱动。

使用示例



当 SM11 由 OFF → ON 变化时，执行 CANRCV 指令。通过 0 通道，接收数据保存到 D0 开始的 4 个单元，接收完成后输出 Y0 为 ON。使用该指令接收数据时，必须要将发送站和接收站的 CANID 设为一致。

注意事项

1．接收数据通过系统组态可以支持高低字节有效或低字节有效。

2．目的操作数 D 不要被其它指令或强制模式重复修改。

3．通讯帧的大小：通讯帧根据所选用的元件类型（D 或者 V）的不同，接收帧的结束字符不能超过其有效范围，每一个 CAN 帧不能超过 255 个字节。

4．网络不支持相同 CANID 的 CAN 帧同时出现；CANID 保留 0。

6.14 便捷指令

6.14.1 SORT : 数据排序指令

	SORT	S m1 m2 D n	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√
步长	16位：SORT (连续) 11步 32位：-----	----- -----	相关元件	SM165			

操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	INT										•	•	•	•	•		•	•	*	•			•		
m1	INT										•	•	•	•	•	•	•	•	*	•	•	•	•	•	•
m2	INT										•	•	•	•	•	•	•	•	*	•	•	•	•	•	•
D	INT										•	•	•	•	•		•	•	*	•			•		
n	INT											•	•	•	•		•	•	*	•	•	•	•		•

＊ 该指令使用 C 元件时，只能使用 C0 ～ C199。

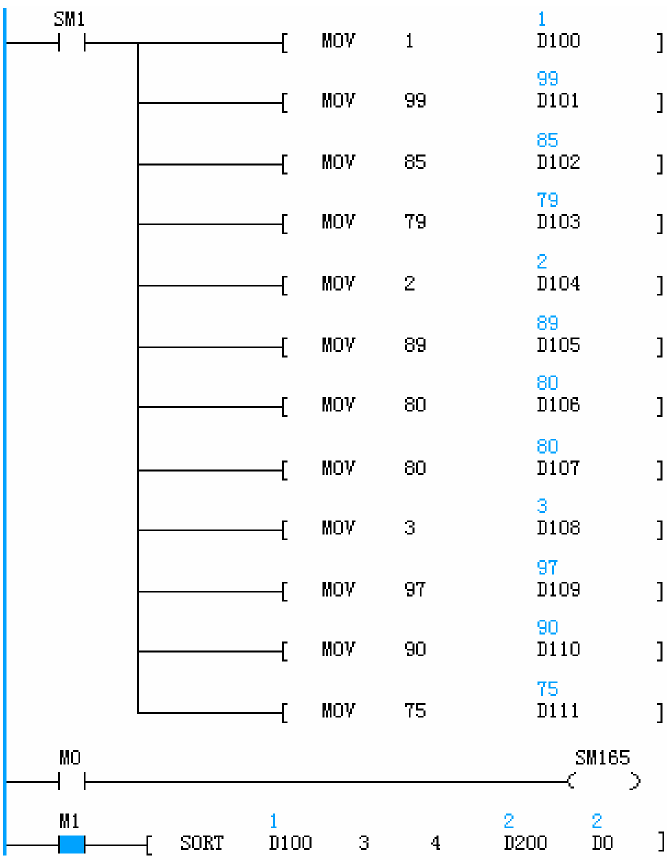
指令说明

├──┤ ├──[SORT S m1 m2 D n]

将 S 为首编号的 m1（范围 1 ～ 31）行 m2（范围 1 ～ 6）列共 m1×m2 个元件中的数据进行排序，排序以 n 指定的列作为参考，排序按由 小到大或由大到小进行（排序方向由 SM165 控制，SM165 为 OFF，升序排列；

SM165 为 ON，降序排列），排序后的数据存入以 D 为首编号的 m1×m2 个连号元件中。

使用示例




```
LD    SM1
MOV   1 D100
MOV   99 D101
MOV   85 D102
MOV   79 D103
MOV   2 D104
MOV   89 D105
MOV   80 D106
MOV   80 D107
MOV   3 D108
MOV   97 D109
MOV   90 D110
MOV   75 D111
LD    M0
OUT   SM165
LD    M1
SORT  D100 3 4 D200 D0
```

表一 以 D100 为首编号的 3 行 4 列共 12 个元件中的数据

列号 行号	1 考生号码	2 语文成绩	3 数学成绩	4 英语成绩
1	D100 1	D101 99	D102 85	D103 79
2	D104 2	D105 89	D106 80	D107 80
3	D108 3	D109 97	D110 90	D111 75

SM165 为 OFF 时，升序排列

SM165 为 ON 时，降序排列

表二 以第 2 列为参考升序排列的结果

列号 行号	1 考生号码	2 语文成绩	3 数学成绩	4 英语成绩
1	D200 2	D201 89	D202 80	D203 80
2	D204 3	D205 97	D206 90	D207 75
3	D208 1	D209 99	D210 85	D211 79

表三 以第 2 列为参考降序排列的结果

列号 行号	1 考生号码	2 语文成绩	3 数学成绩	4 英语成绩
1	D200 1	D201 99	D202 85	D203 79
2	D204 3	D205 97	D206 90	D207 75
3	D208 2	D209 89	D210 80	D211 80

1．在表一中存放了以 D100 为首编号的 3 行 4 列共 12 个元件中的数据。
2．当常开触点 M1 闭合时，SORT 指令执行，此时 M0 未闭合，SM165 为 OFF，故表一中的内容将由小到大排序，排列后的内容存放到 D200 ～ D211 中，D200 ～ D211 中的数据如表二所示。
3．如果 M0 闭合，则 SM165 为 ON，表一中的内容将由大到小排序，排列后的内容存放到 D200 ～ D211，D200 ～ D211 中的数据如表三所示。

6.14.2 RND : 产生随机数指令

	RND	P	D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：RND (连续) 3步 RNDP (脉冲) 4步 32位： ----- -----			相关元件	零标志：SM180			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
D	INT											●	●	●	●		●	●	*	●			●	●	

* 该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

├──┤ ├──[RND D]

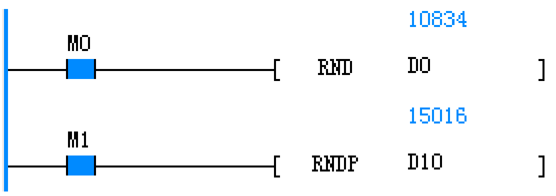
D：保存随机数的软元件编号

1．生成随机数并保存到 D 单元中。

2．产生 0 ～ 32,767 内的随机数，将其数值保存在 D 单元中，若产生的随机数为 0，零标志位（SM180）置 ON。

3．该指令一般使用脉冲执行型指令（RNDP）。

使用示例



LD	M0
RND	D0
LD	M1
RNDP	D10

1．当 M0 为 ON 时，执行 RND 指令，产生随机数，保存在 D0 中。每个扫描周期都执行一次 RND，数值保持随机变化。

2．当 M1 由 OFF → ON 变化时，RNDP 指令执行一次，产生随机数，保存在 D10 中，D10=15,016。

6.14.3 DUTY : 产生定时脉冲指令

	DUTY		S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：DUTY (连续) 7步 ----- 32位： ----- -----			相关元件	SM430～SM434 SD430～SD434			

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●	●
S2	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●	●
D	BOOL								*																

＊ 该指令使用 C 元件时，只能使用 C0 ～ C199；使用 SM 元件时，只能使用 SM430 ～ SM434。

指令说明

```
├──┤ ├──[ DUTY S1 S2 D ]
```

S1：ON 的扫描次数　　**S2**：OFF 的扫描次数

1．产生定时脉冲。

2．一旦开始输出脉冲，即使驱动信号状态改变也不停止，停机或断电时停止。

3．该指令一般使用脉冲执行型指令（RNDP）。

定时输出单元及相关 SD 元件如下：

定时输出目标地址	对扫描计数用的软元件
SM430：定时器时钟输出 1	SD430：定时器时钟输出 1 计数值
SM431：定时器时钟输出 2	SD431：定时器时钟输出 2 计数值
SM432：定时器时钟输出 3	SD432：定时器时钟输出 3 计数值
SM433：定时器时钟输出 4	SD433：定时器时钟输出 4 计数值
SM434：定时器时钟输出 5	SD434：定时器时钟输出 5 计数值

使用示例

```
                               10     10     OFF
    M0
├──┤ ├──────────[ DUTY   D0     D10    SM434  ]
```

```
LD    M0
DUTY  D0 D10 SM434
```

1．当 M0 为 ON 时，执行 DUTY 指令，SM434 按 10 个扫描周期为 ON，10 个扫描周期为 OFF 的方式输出脉冲，同时在 SD430 中保存扫描次数的计数值。

2．M0 为 ON 时，开始输出脉冲，即使 M0 状态改变也不停止，停机或断电时停止。

注意事项

该指令在一个工程中最多使用 5 次，但是多个 DUTY 指令不能使用相同的定时输出目标地址，否则编译错误。

6.14.4 CCITT : CCITT 校验指令

	CCITT	P	S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：CCITT（连续） 7步 CCITTP（脉冲） 8步 32位： ----- -----			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	WORD																•				•		•		
S2	INT										•	•	•	•	•	•	•	•	*	•	•	•	•	•	•
D	WORD																•				•		•		

＊ 该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

├─┤ ├─[CCITT S1 S2 D]

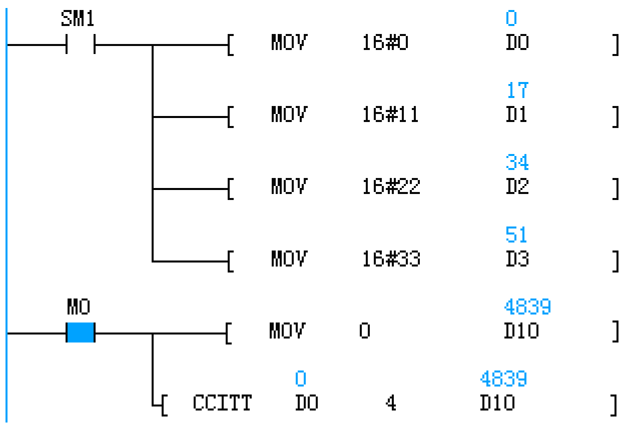
S1：待校验数据的起始单元 **S2**：待校验数据的数量

D：校验结果

1．将起始单元 S1 开始的 S2 个数据进行 CCITT 校验运算，结果赋给 D 单元。

2．CCITT 校验算法的多项式为：X^16+X^12+X^5+1。

使用示例




```
LD     SM1
MOV    16#0  D0
MOV    16#11 D1
MOV    16#22 D2
MOV    16#33 D3
LD     M0
MOV    0 D10
CCITT  D0 4 D10
```

当 M0 为 ON 时，执行 CCITT 指令，将 D0 起始的 4 个数据，进行 CCITT 校验运算，结果赋给 D10。

注意事项

1．S2 ≥ 0，否则，PLC 信息中报“指令操作数数值非法”错误，指令不执行。

2．在每次执行指令时，系统会将指令执行前的 D 的内容带入运算，所以执行前要初始化 D。

3． S1 单元起始的校验数据区内的数据默认为低字节模式，即高字节忽略为零。

4．校验结果为 16 位。

6.14.5 CRC16 : CRC16 校验指令

	CRC16	P	S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：CRC16 (连续) 7步 32位： -----		CRC16P（脉冲） 8步 -----	相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	WORD																•				•		•		
S2	INT										•	•	•	•	•	•	•	•	*	•	•	•	•	•	•
D	WORD																•				•		•		

＊ 该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

├──┤ ├──[CRC16 S1 S2 D]

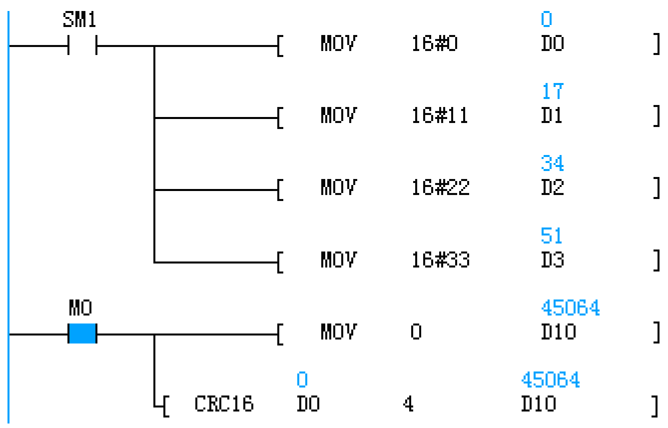
S1：待校验数据的起始单元 **S2**：待校验数据的数量

D：校验结果

1．将起始单元 S1 开始的 S2 个数据进行 CRC16 校验运算，结果赋给 D 单元。

2．CRC16 校验算法的多项式为：X^16+X^15+X^2+1。

使用示例




```
LD     SM1
MOV    16#0  D0
MOV    16#11 D1
MOV    16#22 D2
MOV    16#33 D3
LD     M0
MOV    0 D10
CRC16  D0 4 D10
```

当 M0 为 ON 时，执行 CRC16 指令，将 D0 起始的 4 个数据，进行 CRC16 校验运算，结果赋给 D10。

注意事项

1．S2 ≥ 0，否则 PLC 信息中报“指令操作数数值非法”错误，指令不执行。

2．在每次执行指令时，系统会将指令执行前的 D 的内容带入运算，所以执行前要初始化 D。

3． 如果使用标准 Modbus CRC 校验时，请给 D 元件（校验和）赋初始值 16#FFFF，且高低字节（高 8 位，低 8 位）需要交换。

4．S1 单元起始的校验数据区内的数据默认为低字节模式，即高字节忽略为零。

5．校验结果为 16 位。

6.14.6 LRC : LRC 校验指令

	LRC	P	S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：LRC（连续） 7步 32位： -----		LRCP（脉冲） 8步 -----	相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	WORD																●				●		●		
S2	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●	●
D	WORD																●				●		●		

* 该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

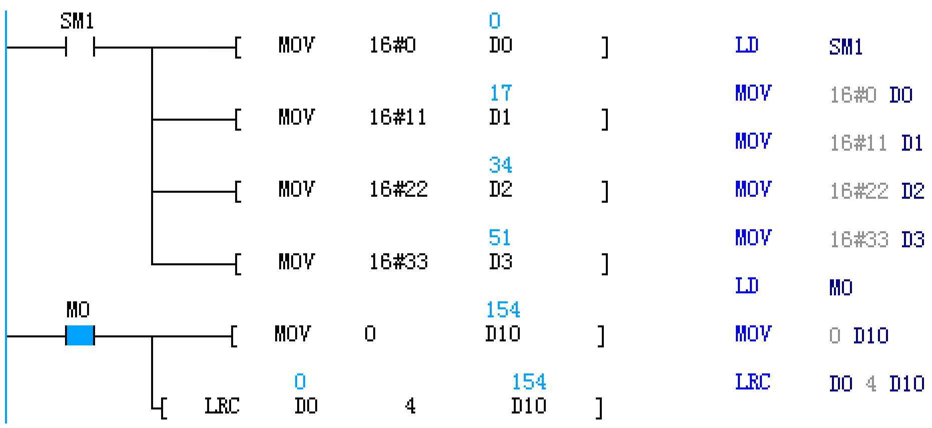
├──┤ ├──[LRC S1 S2 D]

S1：待校验数据的起始单元　　**S2**：待校验数据的数量

D：校验结果

将起始单元 S1 开始的 S2 个数据进行 LRC 校验运算，结果赋给 D 单元。

使用示例



当 M0 为 ON 时，执行 LRC 指令，将 D0 起始的 4 个数据，进行 LRC 校验运算，结果赋给 D10。

注意事项

1．S2 ⩾ 0，否则 PLC 信息中报“指令操作数数值非法”错误，指令不执行。

2．在每次执行指令时，系统会将指令执行前的 D 的内容带入运算，所以执行前要初始化 D。

3．S1 单元起始的校验数据区内的数据默认为低字节模式，即高字节忽略为零。

4．校验结果为 8 位，存放在 D 的低字节。

6.14.7 PID : PID 功能指令

	PID		S1　S2　S3　D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：PID　(连续)　10步 32位：-----		----- -----	相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT																●						●		
S2	INT																●						●		
S3	INT																●						●		
D	INT																●						●		

指令说明

├──┤ ├──[　PID　S1　S2　S3　D　]

S1：设定目标值（SV）　　**S2**：当前测量值（PV）

S3：参数　　**D**：输出运算结果数字量

1．当指令有效时，将 S1 设定值与 S2 测量值之差按 S3 设定的参数进行 PID 运算，运算结果存入 D 中。

2．PID 表示“比例、积分和微分”。PID 运算包括 P 处理、I 处理、D 处理。P（比例）处理是将偏差信号 ΔX 按比例放大，提高控制灵敏度；I（积分）处理是对偏差信号进行积分处理，缓解 P 处理比例放大过大引起的超调和振荡；D（微分）处理是对偏差信号进行微分处理，以提高控制的迅速性。

3．PID 指令在一个程序中可多次使用，但要注意运算使用的 S1、S2、S3 及 D 的软元件号不要重复覆盖。

4．PID 指令的基本运算式：

动作方向	PID运算公式
正动作	$\Delta MV = K_P\left\{\left(EV_n - EV_{n-1}\right) + \frac{T_S}{T_I}EV_n + D_n\right\}$ $EV_n = PV_{nf-1} - SV$ $D_n = \frac{T_D}{T_S + K_D * T_D}\left(PV_{nf} + PV_{nf-2} - 2PV_{nf-1}\right) + \frac{K_D * T_D}{T_S + K_D * T_D} * D_{n-1}$ $MV_n = \sum \Delta MV$
逆动作	$\Delta MV = K_P\left\{\left(EV_n - EV_{n-1}\right) + \frac{T_S}{T_I}EV_n + D_n\right\}$ $EV_n = SV - PV_{nf-1}$ $D_n = \frac{T_D}{T_S + K_D * T_D}\left(2PV_{nf-1} - PV_{nf} - PV_{nf-2}\right) + \frac{K_D * T_D}{T_S + K_D * T_D} * D_{n-1}$ $MV_n = \sum \Delta MV$

符号说明：

EVn—— 本次采样偏差

EVn-1——1 个周期前的偏差

SV—— 目标值

PVnf—— 本次采样值（滤波后）

PVnf-1——1 个周期前采样值（滤波后）

PVnf-2——2 个周期前采样值（滤波后）

ΔMV—— 输出变化量

Dn—— 本次微分项

Dn-1—— 一个周期前的微分项

Kp—— 比例增益

Ts—— 采样周期

TI—— 积分时间

TD—— 微分时间

KD—— 微分增益

MV—— 本次的操作量

5．PID 参数设置内容如下

S3：采样时间（Ts），范围为 1 ～ 32,767（ms），比运算周期短的时间数值无法执行

位	设定值及其含义	
	0	1
0	正动作	逆动作
1	输出工程量无效	输出工程量有效
2 ～ 3	保留	
4	输入变化量报警无效	输入变化量报警有效
5	输出变化量报警无效	输出变化量报警有效
6	输入值上下限报警无效	输入值上下限报警有效
7	输出值上下限报警无效	输出值上下限报警有效
8	积分分离无效	积分分离有效
9	脉冲输出无效	脉冲数出有效
10 ～ 15	保留	

S3+1：动作、报警及上下限功能设置字

S3+2：输入滤波常数（α）范围 0 ～ 99[%]，为 0 时没有输入滤波功能

S3+3：比例增益（Kp）范围 1 ～ 32767[%]

S3+4：积分时间（T_I）范围 0 ～ 32,767（×100ms），为 0 时作为∞处理（无积分）

S3+5：微分增益（K_D）范围 0 ～ 100[%]，为 0 时无微分增益

S3+6：微分时间（T_D）范围 0 ～ 32,767（×10ms），为 0 时无微分处理

S3+7：输出工程量范围 0 ～ 32,767（S3+1 的 BIT1=1 时）

S3+8：输入变化量上限报警设定值 0 ～ 32,767（S3+1 的 BIT4=1 时）

S3+9：输入变化量下限报警设定值 0 ～ 32,767（S3+1 的 BIT4=1 时）

S3+10：输出变化量上限报警设定值 0 ～ 32,767（S3+1 的 BIT5=1 时）

S3+11：输出变化量下限报警设定值 0 ～ 32,767（S3+1 的 BIT5=1 时）

S3+12：输入上限设定值－ 32,768 ～ 32,767（S3+1 的 BIT6=1 时）

S3+13：输入下限设定值－ 32,768 ～ 32,767（S3+1 的 BIT6=1 时）

S3+14：输出上限设定值 0 ～ 32,767（S3+1 的 BIT7=1 时）

S3+15：输出下限设定值 0 ～ 32,767（S3+1 的 BIT7=1 时）

S3+16：积分分离区间设定值 0 ～ 32,767（S3+1 的 BIT8=1 时）

S3+17：脉冲输出端口号（S3+1 的 BIT9=1 时，只支持 Y 端口）

S3+18 ～ S3+25：PID 运算内部数据存储寄存器

其中，S3 ～ S3+ 6 是用户设定操作数。S3+7 ～ S3+17 是用户选择设定操作数。用户可采用后台软件中的 PID 指令向导设定各操作数。

D：输出运算结果数字量

D+1：输出运算结果（MV）范围 0 ～ 1000 单位 0.1%

D+2：比例环节结果输出值范围 0 ～ 32,767

D+3：积分环节结果输出值范围 0 ～ 32,767

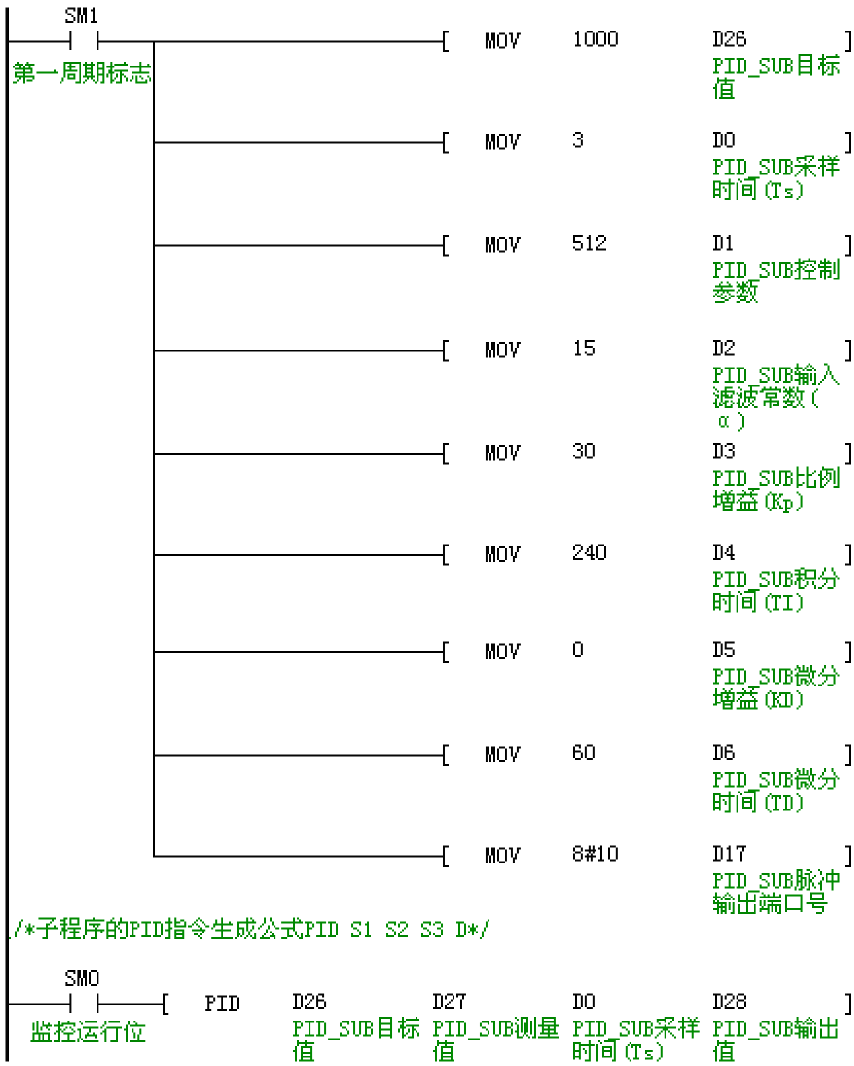
D+4：微分环节结果输出值范围 0 ～ 32,767

D+5：当前偏差值

D+6：报警输出

位	设定值及其含义	
	0	1
0	输入变化量上限没有报警	输入变化量上限报警
1	输入变化量下限没有报警	输入变化量下限报警
2	输出变化量上限没有报警	输出变化量上限报警
3	输出变化量下限没有报警	输出变化量下限报警
4	输入上限没有报警	输入上限报警
5	输入下限没有报警	输入下限报警
6	输出上限没有报警	输出上限报警
7	输出下限没有报警	输出下限报警
8 ～ 15	保留	

使用示例

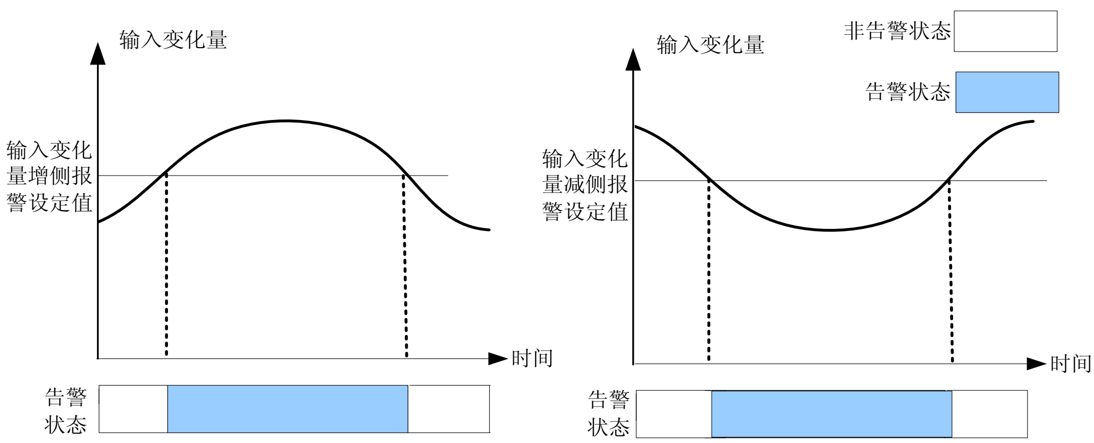


```
//该子程序由MXProgrammer的PID指令向导自动生成。
  要在程序内启用该配置，每个扫描周期中需要用户调用才能执行此子程序。
LD      SM1
MOV     1000 D26        //D26设定目标值
MOV     3 D0            //D0采样时间(Ts) 范围为1～32767(ms)但比运算周期短的时间数值无法执行
MOV     512 D1          //D1控制参数
MOV     15 D2           //D2输入滤波常数(α)范围0～99[%]，为0时没有输入滤波
MOV     30 D3           //D3比例增益(Kp)范围1～32767[%]
MOV     240 D4          //D4积分时间(TI)范围0～32767(×100ms)，为0时作为∞处理(无积分)
MOV     0 D5            //D5微分增益(KD)范围0～100[%]，为0时无微分增益
MOV     60 D6           //D6微分时间(TD)范围0～32767(×10ms)，为0时无微分处理
MOV     8#10 D17        //D17脉冲输出端口号,(只支持Y端口)
LD      SM0
PID     D26 D27 D0 D28  //子程序的PID指令生成公式PID S1 S2 S3 D
```

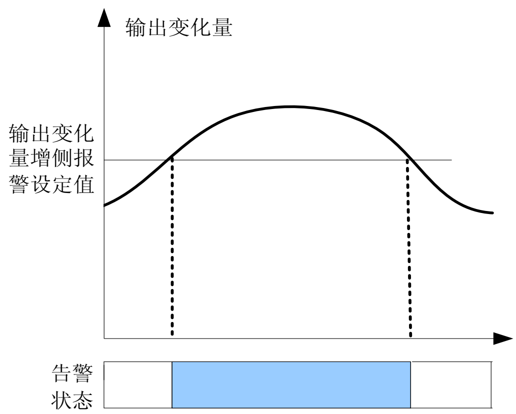
主模块开始运行第一个扫描周期初始化 PID 各操作数，以后的扫描周期中不再对其初始化。当 M0=ON 时，从外部 A/D 模块（实际应用中可以是其它方式）读入当前测量值并填充到测量值单元。执行 PID 运算，将通过外部 D/A 模块（实际应用中可以是其它方式）将运算结果转换成模拟信号，加入到被控系统当中。

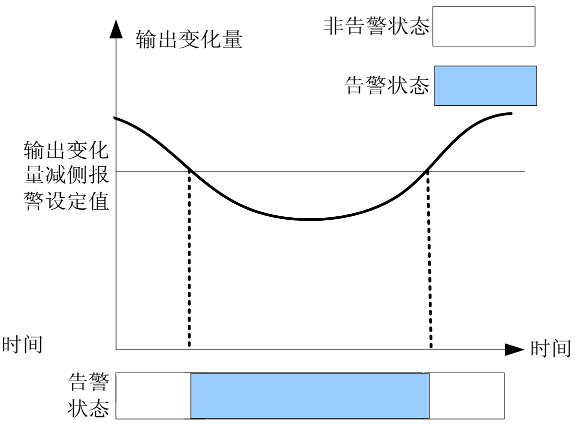
注 意 事 项

1. 当能流有效且达到采样时间时，进行 PID 运算。
2. TPID 和 PID 指令总共不能超过 32 条。
3. PID 指令在定时器中断、子程序、步进梯形图、跳转指令中也可使用。在这种情况下，执行 PID 指令前要先确认操作数设定单元及清除 S3+18 ～ S3+25 后的内部处理数据单元再使用。
4. 动作方向通过 S3+1 的 BIT0 来设定系统的正动作（正反馈）、逆动作（负反馈）模式。
5. 输出工程量的设定通过 S3+1 的 BIT1 来设定。当 BIT1=0 时，输出工程量设定无效；当 BIT1=1 时，输出工程量设定有效。当输出工程量有效时，PID 运算输出的数字量会在工程量范围之内。
6. 输入变化量报警设定通过 S3+1 的 BIT4 来设定。当 BIT4=0 时，输入变化量报警设定无效；当 BIT4=1 时，输入变化量报警设定有效。

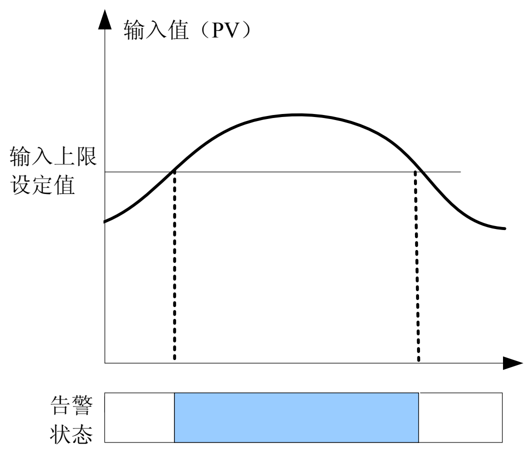


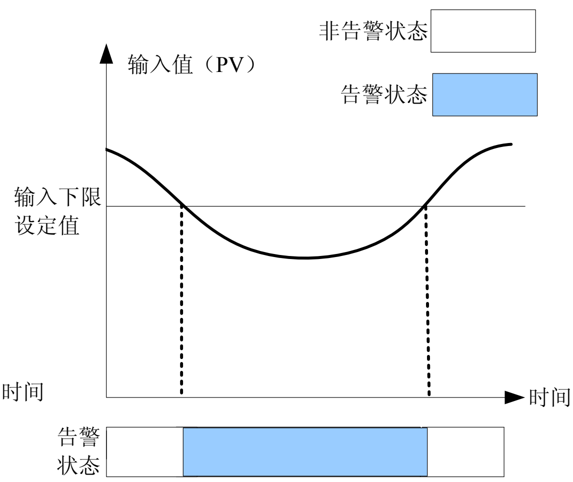
7. 输出变化量报警设定通过 S3+1 的 BIT5 来设定。当 BIT5=0 时，输出变化量报警设定无效；当 BIT5=1 时，输出变化量报警设定有效。



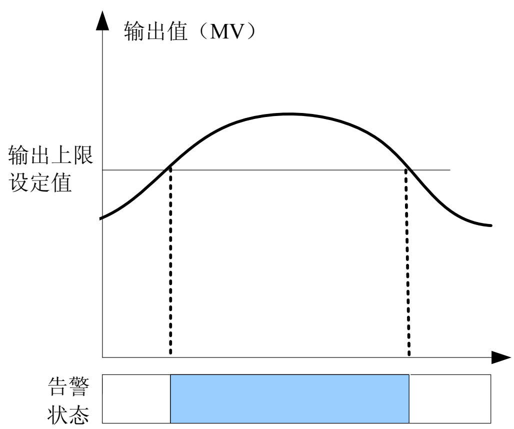


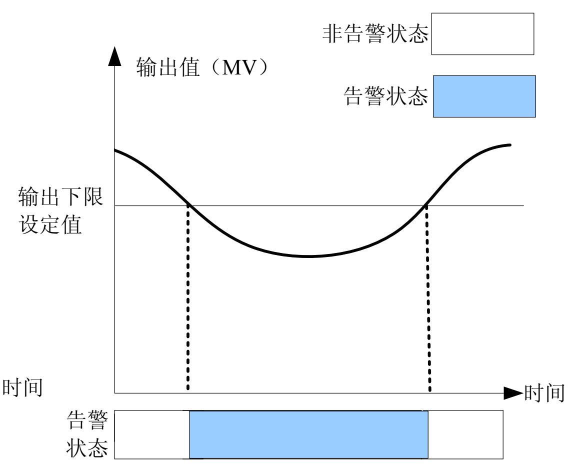
8．输入上下限报警设定 S3+1 的 BIT6 来设定。当 BIT6=0 时，输入上下限报警设定无效；当 BIT6=1 时，输入上下限报警设定有效。





9. 输出上下限报警设定通过 S3+1 的 BIT7 来设定。当 BIT7=0 时，输出上下限报警设定无效；当 BIT7=1 时，输出上下限报警设定有效。





10．积分分离设定通过 S3+1 的 BIT8 来设定。当 BIT8=0 时，不进行积分分离运算；当 BIT8=1 时，进行积分分离运算。积分分离是当测量值与设定值偏差的绝对值在积分分离区间设定值（S3+15）以内时，进行积分累加，当在积分分离区间以外时，不进行积分累加。

11．输入滤波常数有使测量值变化平滑的效果。

12．微分增益有缓和输出值剧烈变化的效果。

13. 脉冲输出端口设定（S3+17）。当用户选择脉冲输出时，设定脉冲输出的端口号，比如（S3+17）=5，则 Y5 根据运算结果来进行脉冲输出。
14. 对于 D 请指定在非停机保持的数据寄存器区。若指定在停机保持的数据寄存器区时，请在初次运行时，务必清 0（即 LD SM1 MOV 0 D****）。
15. PID 指令需要占用 S3 起始的 26 个数据寄存器。
16. 采样时间 Ts 的最大误差为 -（1 个扫描周期 +1ms）～ +1 个扫描周期。当 Ts 值较小时，将影响 PID 效果，建议最好在定时中断中使用 PID 指令。
17. 当设置输入、输出变化量报警有效时，S3+8 ～ S3+11 单元的设定值不能为负值，否则系统报操作数错误，不执行 PID 运算。
18. 当 S3+1 的 BIT2 和 BIT5 同时为 ON 时，系统会认为设置无效（等同于 BIT2 和 BIT5 同时为 OFF），不进行上下限限制或变化量超值报警。
19. 当 PID 的控制操作数（S3 ～ S3+6 单元）设定值不在有效范围内，系统会报操作数错误，不进行 PID 运算。
20. 采样时间小于等于 1 个扫描周期、运算过程中出现数据溢出、结果溢出等，不报警，PID 运算继续进行。
21. PID 指令首次执行前需要先将各操作数初始化，若运行过程中各操作数无变化，且控制操作数单元不会被其它程序覆盖，初始化程序可以只执行一次。PID 运算中间数据存储区中的数据若在 PID 运算过程中被改写，会导致运算结果有误。

6.14.8 TPID : 温度 PID 功能指令

	TPID		S1 S2 S3 D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：TPID (连续) 10步 ----- 32位： ----- -----			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT																●						●		
S2	INT																●						●		●
S3	INT																●						●		
D	INT																●						●		

指令说明

├──┤ ├──[TPID S1 S2 S3 D]

S1：设定目标值（SV） **S2**：当前测量值（PV） **D**：输出运算结果数字量 **S3**：参数

1. 当指令有效时，将 S1 设定值与 S2 测量值之差按 S3 设定的参数进行 TPID 运算，运算结果存入 D 中。
2. TPID 运算包括 P 处理、I 处理、D 处理。P（比例带）处理是将比例作用减弱、调节器放大倍数减小、提高系统稳定性和增大系统静态偏差；I（积分）处理使对偏差信号进行积分处理，缓解 P 处理比例带减小过大引起的超调和振荡；D（微分）处理是对偏差信号进行微分处理，以提高控制的迅速性。
3. TPID 指令在一个程序中可多次使用，但要注意运算使用的 S1、S2、S3 或 D 软元件号不要重复覆盖。
4. TPID 参数设置内容如下：

   S3：采样时间（**Ts**），范围为 **1** ～ **100**（**s**），比运算周期短的时间数值无法执行

   S3 ＋ 1：动作、报警及上下限功能设置字

位	设定值及其含义	
	0	1
0	正动作	逆动作
1	输出工程量无效	输出工程量有效
2	不进行自整定	进行自整定
3 ～ 5	保留	
6	输入值上下限无效	输入值上下限有效
7	输出值上下限无效	输出值上下限有效
8	保留	
9	输出脉冲无效	输出脉冲有效
10 ～ 11	保留	
12	比例带控制有效	比例带控制无效
13	积分控制有效	积分控制无效
14	微分控制有效	微分控制无效
15	保留	

S3+2：比例带增益（PB）范围 1 ～ 1,000[%]

S3+3：积分时间（TI）范围 0 ～ 3,600（s），为 0 时作为 ∞ 处理（无积分）

S3+4：微分时间（TD）范围 0 ～ 3,600（s），为 0 时无微分处理

S3+5：输出工程量范围 0 ～ 32,767（S3+1 的 BIT1=1 时）

S3+6：输入上限报警设定值 - 32,768 ～ 32,767（S3+1 的 BIT6=1 时）

S3+7：输入下限报警设定值 - 32,768 ～ 32,767（S3+1 的 BIT6=1 时）

S3+8：输出上限报警设定值 0 ～ 32,767（S3+1 的 BIT7=1 时）

S3+9：输出下限报警设定值 0 ～ 32,767（S3+1 的 BIT7=1 时）

S3+10：自整定偏差设定（自整定偏差是指开始自整定时的温度与设定温度的偏差）

S3+11：脉冲输出端口号（S3+1 的 BIT9=1 时，只支持 Y 端口）

S3+12 ～ **S3+22**：PID 运算内部存储器

S3+23 ～ **S3+43**：自整定运算内部存储器

其中，S3 ～ S3+4 是用户设定操作数。S3+5 ～ S3+11 是用户选择设定操作数。用户可采用后台软件中的 TPID 指令向导设定各操作数。

D：输出运算结果数字量（当 S3+1 的 BIT1=1，范围为输出工程量范围；当 S3+1 的 BIT1=0 时，范围为（0 ～ 32767）

D+1：输出运算结果（MV）范围 0 ～ 1000 单位 0.1%

D+2：比例带环节结果输出值 0 ～ 32767

D+3：积分环节结果输出值 0 ～ 32767

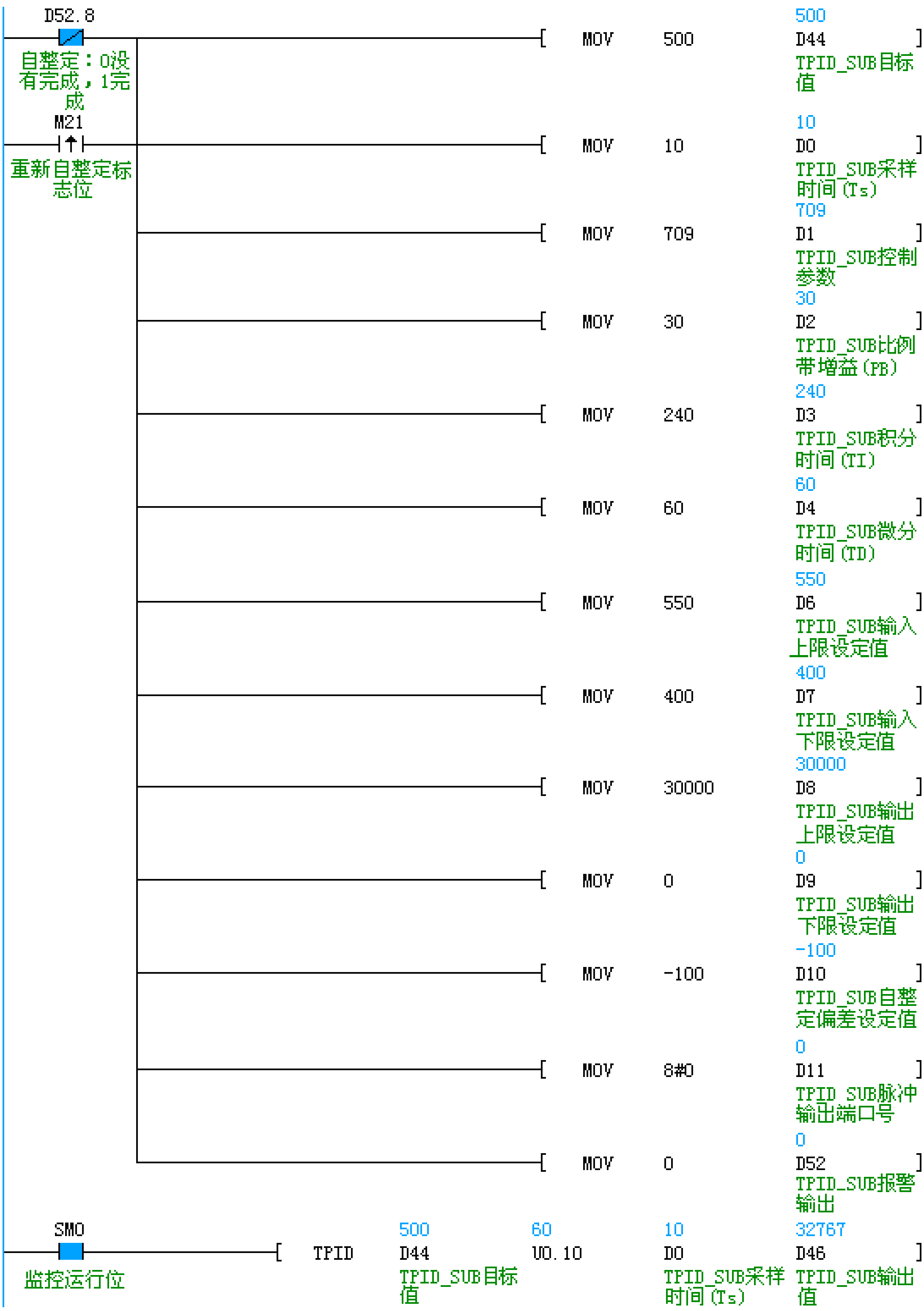
D+4：微分环节结果输出值 0 ～ 32767

D+5：当前偏差值

D+6：报警输出

位	设定值及其含义	
	0	1
0 ～ 3	保留	
4	输入上限没有报警	输入上限报警
5	输入下限没有报警	输入下限报警
6	输出上限没有报警	输出上限报警
7	输出下限没有报警	输出下限报警
8	自整定没有完成	自整定完成
9 ～ 15	保留	

使用示例

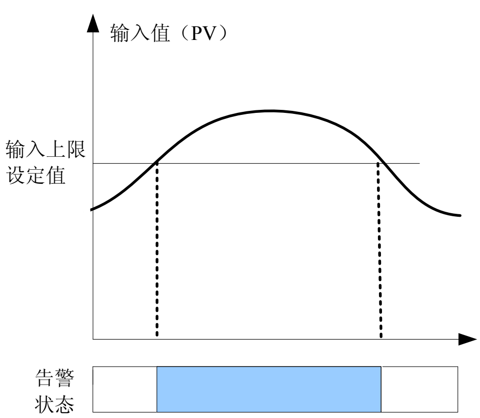

D52.8
自整定：0没有完成，1完成
M21
重新自整定标志位
MOV 500 D44] 500 TPID_SUB目标值
MOV 10 D0] 10 TPID_SUB采样时间(Ts)
MOV 709 D1] 709 TPID_SUB控制参数
MOV 30 D2] 30 TPID_SUB比例带增益(PB)
MOV 240 D3] 240 TPID_SUB积分时间(TI)
MOV 60 D4] 60 TPID_SUB微分时间(TD)
MOV 550 D6] 550 TPID_SUB输入上限设定值
MOV 400 D7] 400 TPID_SUB输入下限设定值
MOV 30000 D8] 30000 TPID_SUB输出上限设定值
MOV 0 D9] 0 TPID_SUB输出下限设定值
MOV -100 D10] -100 TPID_SUB自整定偏差设定值
MOV 8#0 D11] 0 TPID_SUB脉冲输出端口号
MOV 0 D52] 0 TPID_SUB报警输出
SM0
监控运行位
TPID
500 D44 TPID_SUB目标值
60 U0.10
10 D0 TPID_SUB采样时间(Ts)
32767 D46] TPID_SUB输出值

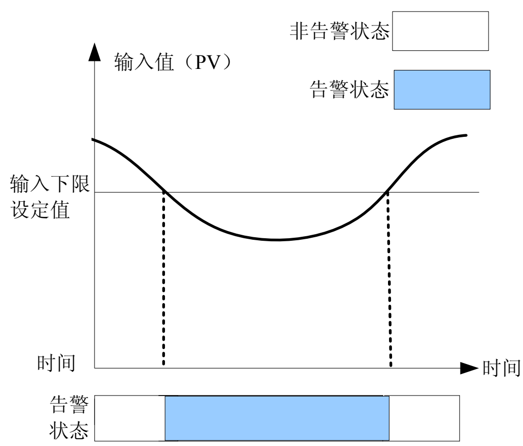
```
//该程序在MXProgrammer的TPID指令向导生成的子程序基础上根据需要略做修改，并将其放在主程序里使用。
LDI     D52.8
ORP     M21
MOV     500 D44     //D44设定目标值
MOV     10 D0       //D0采样时间(Ts) 范围为1～100(s)但比运算周期短的时间数值无法执行
MOV     709 D1      //D1控制参数
MOV     30 D2       //D2 比例带增益(PB)范围1～1000[%]
MOV     240 D3      //D3积分时间(TI)范围0～3600(s)，为0时作为∞处理(无积分)
MOV     60 D4       //D4微分时间(TD)范围0～3600(s)，为0时无微分处理
MOV     550 D6      //D6输入上限设定值-32768～32767
MOV     400 D7      //D7输入下限设定值-32768～32767
MOV     30000 D8    //D8输出上限设定值0～32767
MOV     0 D9        //D9输出下限设定值0～32767
MOV     -100 D10    //D10自整定偏差设定值-32768~32767
MOV     8#0 D11     //D11脉冲输出端口号.(只支持Y端口)
MOV     0 D52
LD      SM0
TPID    D44 U0.10 D0 D46
```

1．此工程中通过自整定标志位设置 PID 参数。当自整定未完成时，将默认 PID 参数赋给系统；当自整定结束后，自整定完成标志位 D52.8（D52 为报警输出：D52 的 bit8 为 0 时，自整定没有完成；为 1 时，自整定完成）为 ON， 此时系统开始使用自整定生成的 PID 参数。
2．此工程中的 D 设为掉电保持，这样在 PLC 重新上电时会直接使用已经整定好的 PID 参数。
3．M21 由 OFF 变 ON 时，将默认的 PID 参数重新赋给系统，并将自整定完成标志位清零，系统重新自整定。
4．系统的采样值可通过模块（如 4TC 模块）直接测得，模块的参数可在 PLC 的系统组态部分设置。
5．系统输出量较大时，Y0 一直输出，输出量较小时，Y0 间歇输出。因此，可通过 Y0 的通断时长来控制外部环境的升温和保温。

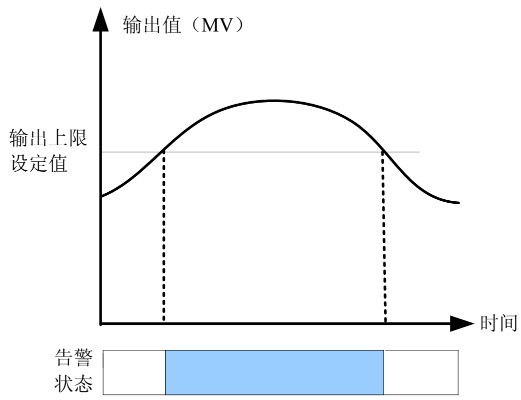
注意事项

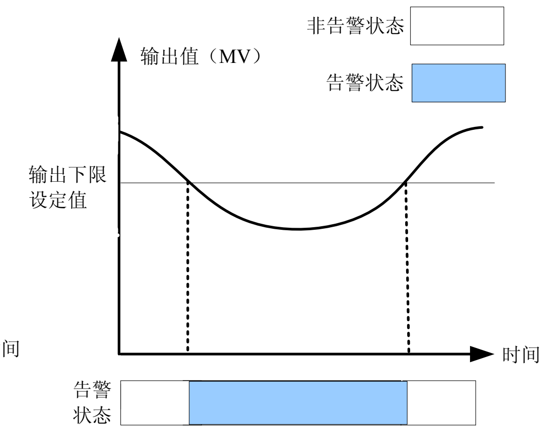
1．当能流有效且达到采样时间时，进行 PID 运算。
2．TPID 和 PID 指令总共不能超过 32 条，带自整定的 TPID 不能超过 8 条，超过 8 条只会有 8 条执行自整定，其它不执行自整定操作。
3．TPID 指令在定时器中断、子程序、步进梯形图、跳转指令中也可使用。在这种情况下，执行 TPID 指令前要先确认操作数设定单元及清除内部处理数据单元后再使用。
4．动作方向通过 S3+1 的 BIT0 来设定系统的正动作（正反馈）、逆动作（负反馈）模式。
5．自整定设置通过 S3+1 的 BIT2 来设定。
6．输出工程量通过 S3+1 的 BIT1 来设定。当 BIT1=0 时，其设定无效；反之，其设定有效。
7．输入上下限报警设定 S3+1 的 BIT6 来设定，当 BIT6=0 时，输入上下限报警设定无效；当 BIT6=1 时，输入上下限报警设定有效。





8. 输出上下限报警设定通过 S3+1 的 BIT7 来设定。当 BIT7=0 时，输出上下限报警设定无效；当 BIT7=1 时，输出上下限报警设定有效。





9. 比例带积分微分各控制环节单独使能设定可通过 S3+1 的 BIT12、BIT13、BIT14 来设定，当为 0 时，使能各环节；当为 1 时，屏蔽各环节，可组成 P、PI、PID 等控制结构。
10. 建议使用停电保持区的 D 寄存器来储存参数，以免自动调整过的参数因停电后而消失。经过自动调整过的参数，并不能保证一定适用于每个控制的环境，因此使用者可自行调整修改过的参数，不过建议最好修改 T_I 或 K_D 就好。
11. TPID 指令需要占用 S3 起始的 44 个数据寄存器和 D 起始的 6 个数据寄存器。
12. 采样时间 Ts 的最大误差为 -（1 个扫描周期+1ms）～ +1 个扫描周期。建议最好在定时中断中使用该指令。
13. 当 TPID 的控制操作数（S3 ～ S3+4）设定值不在有效范围内，PLC 信息中报“指令操作数数值非法”错误，指令不执行，不进行 TPID 运算。
14. 采样时间小于等于 1 个扫描周期、运算过程中出现数据溢出、结果溢出等，不报警，TPID 运算继续进行。
15. TPID 指令首次执行前需要先将各操作数初始化，若运行过程中各操作数无变化，且控制操作数单元不会被其它程序覆盖，初始化程序可以只执行一次。TPID 运算中间数据存储区中的数据若在 TPID 运算过程中被改写，会导致运算结果有误。

6.14.9 RAMP：斜坡信号输出指令

	RAMP		S1 S2 D1 S3 D2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	12步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●	●
S2	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●	●
D1	INT																●				●		●		●
S3	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●	●
D2	BOOL		●	●	●	●	●	●																	

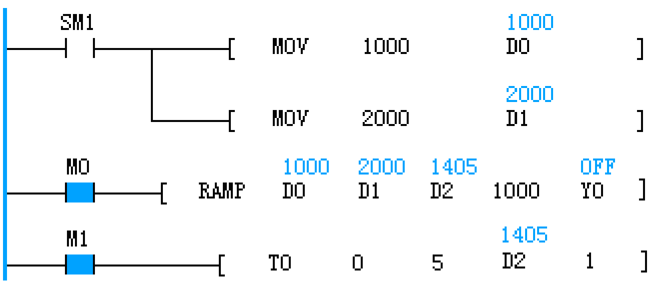
＊ 该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

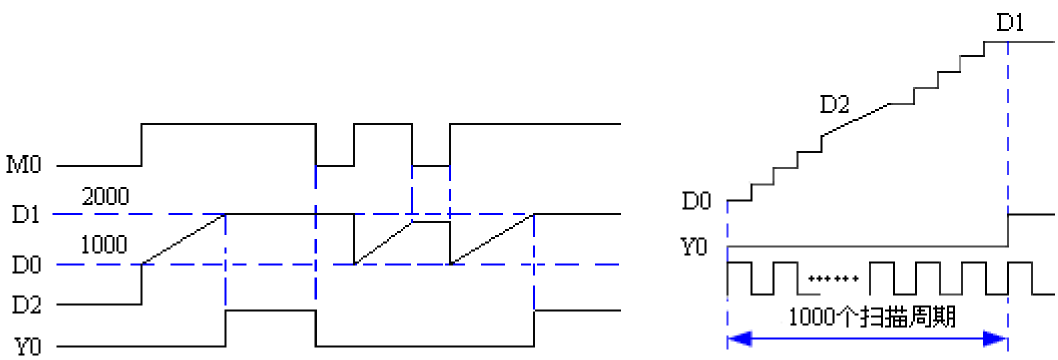
RAMP S1 S2 D1 S3 D2

D1 中的数值从 S1 变化到 S2，变化 S3 个扫描周期，D1 等于 S2 后保持当前值，D2 输出 ON。若指令执行中被打断，则 D2 输出 OFF，D1 保持当前值，直到再次执行指令时，D1 初始化为 S1，并重新开始一个新的斜坡过程。

使用示例




```
LD    SM1
MOV   1000 D0
MOV   2000 D1
LD    M0
RAMP  D0 D1 D2 1000 Y0
LD    M1
TO    0 5 D2 1
```



当 M0 为 ON 时，执行 RAMP 指令，每个扫描周期 D2 中的数值增加 1[（2000 － 1000）/ 1000]。当 D2=D1=2000 时，D2 中的数值不再变化，同时 Y0 输出 ON。斜坡产生过程中，若 M0 断开，Y0 输出 OFF，D2 保持当前数值，直到 M0 再次为 ON，D2 初始化为 D0（D0=1000），重新开始一个斜坡过程。

注意事项

1．当计算步长不能整除时，采取“四舍五入”法；当 S1=S2 时，D1=S2，D2=ON，不产生计数脉冲。
2．开关闭合，指令一直执行，产生一次斜坡。
3．可将系统的扫描时间设为恒定扫描时间，这样可以正确得到斜坡的周期。
4．将该指令与模拟输出相结合，可以输出缓冲的启动 / 停止指令。
5．RAMP、HACKLE、TRIANGLE 指令在程序中的总数不能超过 100 条。

6.14.10 TRIANGLE：三角波信号输出指令

				适用机型	MX1H	MX2H	MX2V	MX3H
	TRIANGLE		S1 S2 D1 S3 D2		√	√	√	√
步长	12步			相关元件				

操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z				Un.b
S1	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●		●
S2	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●		●
D1	INT																●				●		●			●
S3	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●		●
D2	BOOL		●	●	●	●	●	●																		

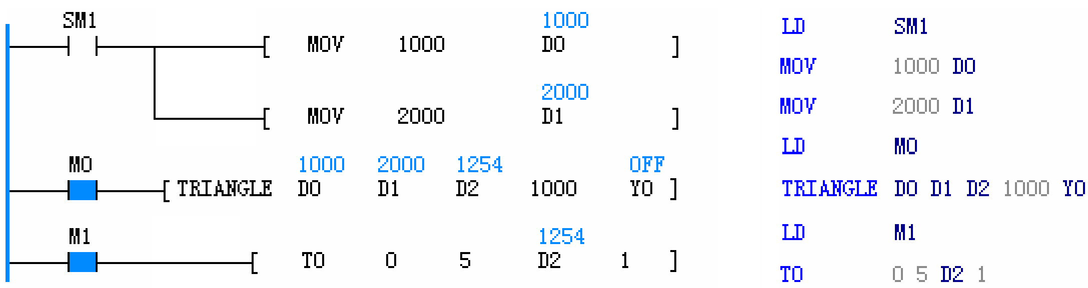
＊ 该指令使用 C 元件时，只能使用 C0 ～ C199。

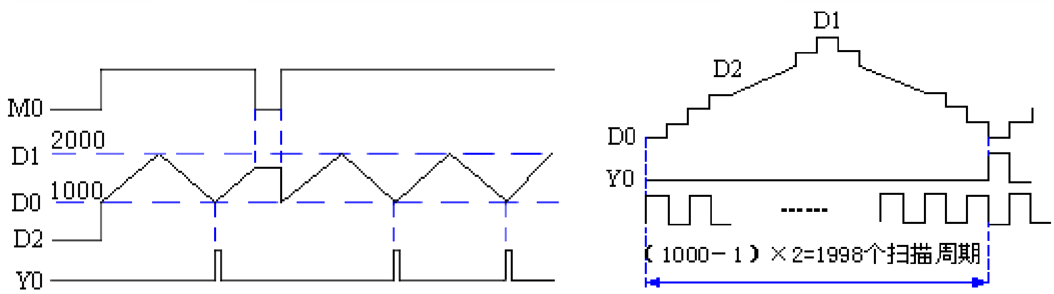
指令说明

├──┤ ├──[TRIANGLE S1 S2 D1 S3 D2]

D1 中数值从 S1 变化到 S2 时，完成三角波的前半坡，此时改变输出值的增量方向，继续产生后半坡，当 D1 再次到达 S1 值时，D2 输出 ON。下一个扫描周期若指令仍然执行，D2 输出 OFF，继续产生下一个三角波。若指令执行中被打断，则 D2 输出 OFF，D1 保持当前值，直到再次执行指令时，D1 初始化为 S1 的值，并重新开始一个新的三角波过程。三角波的周期为（S3 － 1）×2 个扫描周期，且 S3>0。

使用示例





M0 为 ON 时，执行 TRIANGLE 指令，每个扫描周期 D2 增加 1[（2000 － 1000）/ 1000]。当 D2=D1=2000 时，三角波半坡完成，此后每个扫描周期 D2 减 1。当 D2=D0=1000 时，完整三角波完成，Y0 输出 ON。下一个扫描周期，若 M0 保持 ON，Y0 输出 OFF，产生下一个三角波。若运行过程中，M0 变为 OFF，Y0 输出 OFF，D2 保持当前值，直到 M0 再次为 ON 时，D2 初始化为 D0（D0=1000），重新产生一个三角波。可以通过外部特殊模块将数据转换为模拟波形。

注意事项

1．当计算步长不能整除时，采取“四舍五入”法；当 S1=S2 时，D1=S2，D2=ON（不产生计数脉冲）。
2．开关闭合指令一直执行，产生一系列连续的三角波。
3．RAMP、HACKLE、TRIANGLE 指令在程序中的总数不能超过 100 条。
4．将该指令与模拟输出相结合，可以输出缓冲的启动 / 停止指令。

6.14.11 HACKLE：锯齿波信号输出指令

	HACKLE		S1　S2　D1　S3　D2	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	12步			相关元件				

操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z				Un.b
S1	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●	●
S2	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●	●
D1	INT																●				●		●		●
S3	INT										●	●	●	●	●	●	●	●	*	●	●	●	●	●	●
D2	BOOL		●	●	●	●	●	●																	

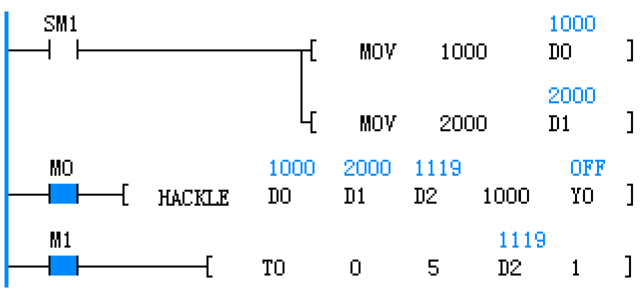
*　该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

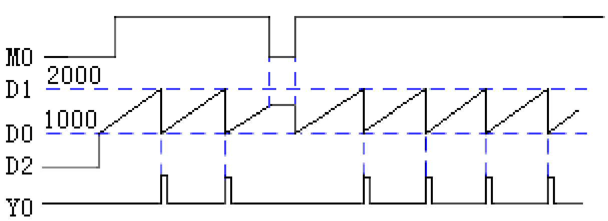
├─┤├─[　HACKLE　S1　S2　D1　S3　D2　]

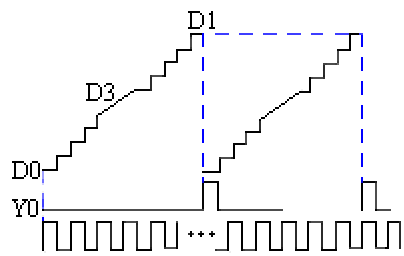
D1 中的数值从 S1 变化到 S2，变化 S3（S3>0）个扫描周期，当 D1 中的数值等于 S2 中的数值时，将 D2 置 ON。下一个扫描周期若指令仍然执行，将 D2 置 OFF，继续产生下一个锯齿波。若指令执行中被打断，则 D2 为 OFF，D1 保持当前值，直到再次执行指令时，D1 初始化为 S1，并重新开始一个新的锯齿波过程。

使用示例




```
LD      SM1
MOV     1000 D0
MOV     2000 D1
LD      M0
HACKLE  D0 D1 D2 1000 Y0
LD      M1
T0      0 5 D2 1
```





当 M0 为 ON 时，执行 HACKLE 指令，每个扫描周期 D2 增加 1[（2000 － 1000）/ 1000]。当 D2=D1=2000 时，一个锯齿波完成，Y0 输出 ON。下一个扫描周期，若 M0 保持 ON 状态，Y0 输出 OFF，开始下一个锯齿波过程。若运行过程中，M0 变为 OFF，则输出状态 Y0 为 OFF，D2 保持当前状态，直到 M0 再次为 ON 时，D2 初始化为 D0（D0=1000），重新开始一个新的锯齿波过程。可以通过外部特殊模块将数据转换为模拟波形。

注意事项

1．当计算步长不能整除时，采取“四舍五入”法。
2．开关闭合指令一直执行，产生一系列连续的锯齿波。
3．当 S1=S2 时，D1=S2，D2=ON（不再产生计数脉冲）。
4．RAMP、HACKLE、TRIANGLE 指令在程序中的总数不能超过 100 条。

6.14.12 ABSD：凸轮控制指令

D	ABSD		S1 S2 D S3	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：ABSD (连续) 9步 ----- 32位：DABSD (连续) 11步 -----			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	(D)INT										●	●	●	●			●		*	*			●		
S2	(D)INT																		*						
D	BOOL		●	●	●																				
S3	WORD																							●	

* 16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T 元件。

指令说明

├──┤ ├──[ABSD S1 S2 D S3]

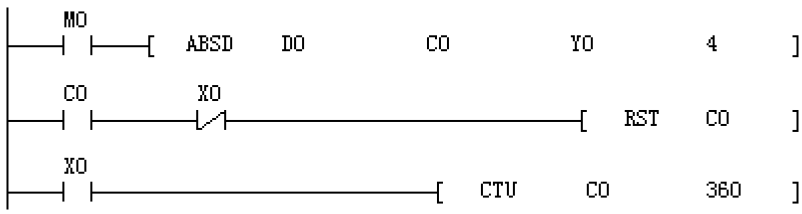
S1：保存表格数据的起始元件编号 **S2**：与表格数据比较的计数器编号

D：输出起始位元件的编号 **S3**：表格的行数以及输出的位元件的点数，1 ≤ S3 ≤ 64

将 S1 开始的 S3 行表格数据（占用 S3 行 ×2 点）与计数器 S2 的当前值作比较，对连续 S3 点的 D 输出作 ON/OFF 控制。

使用示例

平台旋转一次（0 ～ 360 度）期间控制输出 ON/OFF（1 度 1 个脉冲的旋转角度信号）。

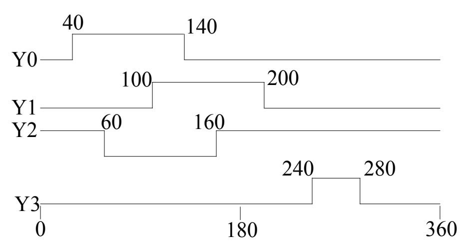



```
LD    M0
ABSD  D0 C0 Y0 4
LD    C0
ANI   X0
RST   C0
LD    X0
CTU   C0 360
```

1. 当 M0 为 ON 时，执行 ABSD 指令，将 Y0 为首编号的 4 个连号元件 Y0 ～ Y3 作为波形输出元件，将 C0 计数器当前值与 D0 为首编号的 8 个元件 D0 ～ D7 中的数据进行比较，然后让 Y0 ～ Y3 输出与 D0 ～ D7 数据相关的波形。
2. Y0 ～ Y3 输出波形与 D0 ～ D7 数据的关系如下图一。D0 ～ D7 中的数据可用 MOV 指令传送，D0 ～ D7 的偶数编号元件用来存储上升数据点，奇数编号用来存储下降数据点。
3. 在常开触点 M0 闭合期间，X0 端子外接平台每旋转 1°，该端子就输入一个脉冲，X0 闭合一次，计数器 C0 的值增 1。当平台旋转到 40° 时，C0 值为 40，C0 值与 D0 值相等，ABSD 指令则让 Y0 元件由 OFF 变为 ON。当 C0 值达到 360 时，C0 常开触点闭合，RST C0 指令执行，将计数器 C0 复位，然后重复图二的过程。

上升数据点	数值	下降数据点	数值	输出
D0	40	D1	140	Y0
D2	100	D3	200	Y1
D4	160	D5	60	Y2
D6	240	D7	280	Y3

图一 在 D0 ～ D7 单元中写入如上所示数据



图二 当 M0 为 ON 时，Y0 ～ Y3 变化如上

注意事项

1. 16 位该指令操作数使用 Kn 寻址时，n 值必须为 4；32 位该指令操作数使用 Kn 寻址时，n 值必须为 8。
2. 在 DABSD 指令中，S2 也可以使用高速计数器。但在这种情况下，对于计数器的当前值、输出波形会由于扫描循环而造成响应滞后，需要立即响应时，请使用 DHSZ 指令进行高速比较功能。

6.14.13 ALT : 交替输出指令

	ALT	P	D	适用机型	MX1H	MX2H	MX2V	MX3H
					√	√	√	√
步长	16位：ALT (连续) 3步 32位： -----		ALTP (脉冲) 4步 -----	相关元件				

操作数	数据类型	位软元件										字软元件												变址	常数	模块
		X	Y	PY	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
D	BOOL	●	●	●	●	●					●															

指令说明

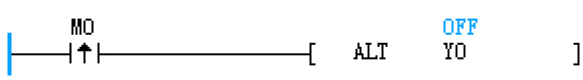
├──┤ ├──[ALT D]

D：交替输出的位元件地址

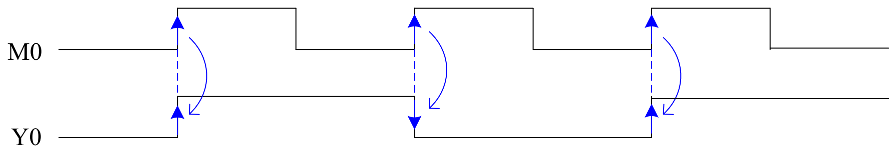
产生交替输出脉冲，每个扫描周期软元件反向动作。

使用示例

示例 1.

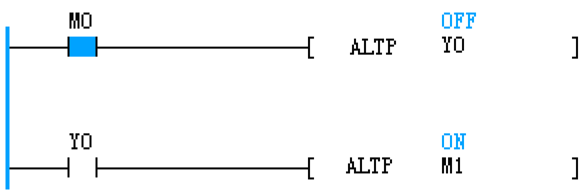



```
LDP   M0
ALT   Y0
```

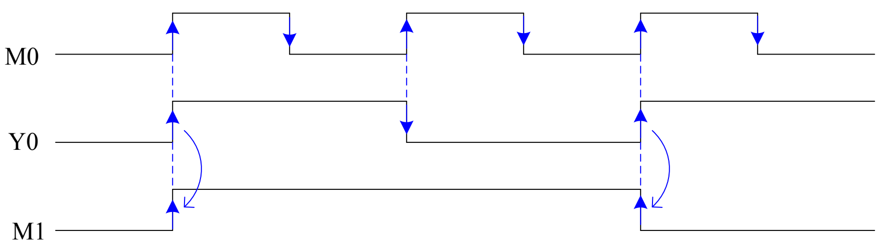


当 M0 由 OFF → ON 变化时， Y0 由 OFF → ON 变化；当 M0 再次由 OFF → ON 变化时，Y0 由 ON → OFF 变化。采用 ALT 指令，在每个扫描周期 Y0 状态都会取反一次，因此常采用 ALTP 指令。

示例 2.

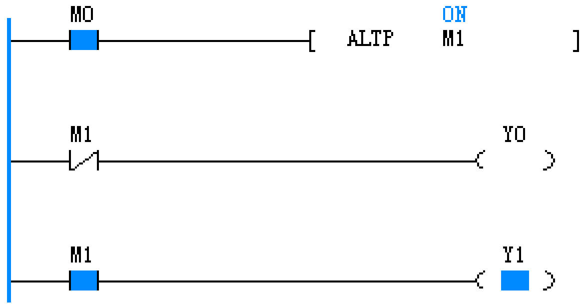



```
LD    M0
ALTP  Y0
LD    Y0
ALTP  M1
```



利用 ALTP 指令可以实现多级的分频输出。如上图所示，当 M0 按图示频率通断时，Y0 产生的脉冲频率降低一半，而 M1 产生的脉冲频率较 Y0 再降低一半，每使用一次 ALT 指令可进行一次二分频。

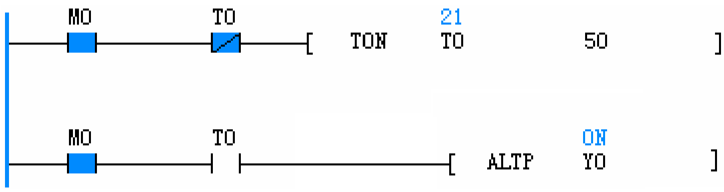
示例 3．用一个按钮控制多个负载启动、停止



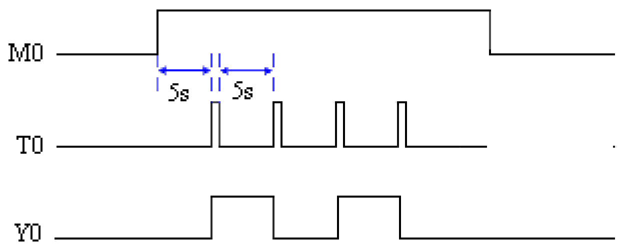

```
LD    M0
ALTP  M1
LDI   M1
OUT   Y0
LD    M1
OUT   Y1
```

当 M0 由 OFF → ON 变化时，M1 由 OFF → ON，M1 常开触点闭合，常闭触点断开，Y0 对应的负载停止，Y1 对应的负载启动。当 M0 第二次由 OFF → ON 变化时，M1 由 ON → OFF，M1 常闭触点闭合，常开触点断开，Y0 对应的负载启动，Y1 对应的负载停止。

示例 4．闪烁动作




```
LD    M0
ANI   T0
TON   T0 50
LD    M0
AND   T0
ALTP  Y0
```



当 M0 为 ON 时，定时器 T0 每隔 5s 瞬时动作，T0 每次由 OFF → ON 变化时，Y0 交替输出 ON/OFF。

6.14.14 INCD: 凸轮控制相对方式指令

	INCD		S1 S2 D N	适应机型	MX1H	MX2H	MX2V	MX3H
							√	√
步长	16位：INCD (连续) 9步 32位： ----- -----			相关元件	SM29			

操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT										●	●	●	●	●	●	●	●	*	●	●	●	●		
S2	INT																		*				●		
D	BOOL		●	●	●																				
N	INT										●	●	●	●	●	●	●	●	*	●	●	●		*	

* 该指令使用 C 元件时，S1、S2、N 只能使用 C0 ～ C199。

* 该指令的操作数 N 使用常数时，范围：0<N ≤ 64。

指令说明

├──┤ ├──[INCD S1 S2 D N]

S1：保存设定值的起始字软元件编号

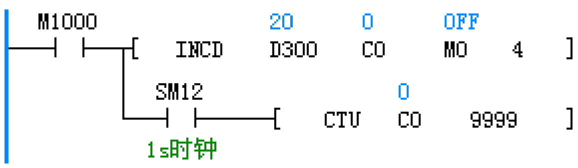
S2：监控当前值用的计数器的起始编号

D：输出的起始位软元件编号

N：输出的位软元件的点数

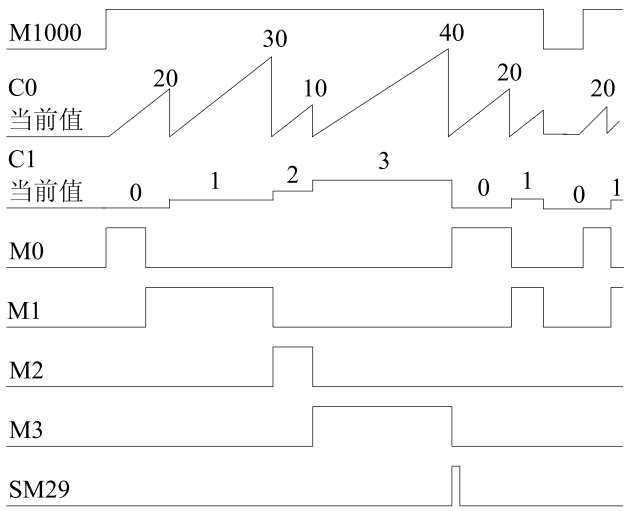
对 S1 开始的 N 行表格数据，与计数器的当前值 S2 进行比较，如果一致，则复位，并依次对输出进行 ON/OFF 控制。

使用示例



使用传送指令 D300 开始的序列单元赋值：

保存软元件		输出	
软元件编号	数据值	软元件编号	例
S1	D300=20	D	M0
S1+1	D301=30	D+1	M1
S1+2	D302=10	D+2	M2
S1+3	D303=40	D+3	M3
⋮	⋮	⋮	⋮
S1+n-1	-	D+n-1	-



1．指令触点如果为 ON，则 M0 输出也为 ON。

2．当 C0 的当前值达到比较值 D300 时，输出 M0 被复位，工序计数器 C1 的计数值被 +1，计数器 C0 的当前值也被复位。

3．M0 被复位、C0 当前值也被清零后，接下来的输出 M1 为 ON。

4．C0 当前值与比较值 D301 比较，如果达到比较值，则输出 M1 被复位，工序计数器 C1 的计数器被 +1，计数器 C0 的当前值也被复位。

5．同样的方式一直要比较到 N 指定的点数为止。

6．在 N 指定的最后工序结束后，执行结束标志位 SM29 保持 1 个运算周期为 ON。
由于 SM29 是多个指定的指令使用执行结束标志位，所以直接在指令的后面作为触点使用，请务必作为该指令专用的结束标志位。

7． 回到最初，重复输出。

注意事项

在 S1 中指定位软元件的位数时，软元件编号中，请指定 16 的倍数（0、16、32、64、…）。

6.14.15 TTMR: 示教定时器指令

	TTMR		D N	适用机型	MX1H	MX2H	MX2V	MX3H
							√	√
步长	5步			相关元件				

操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
D	INT																●					●	●		
N	INT																●							●	

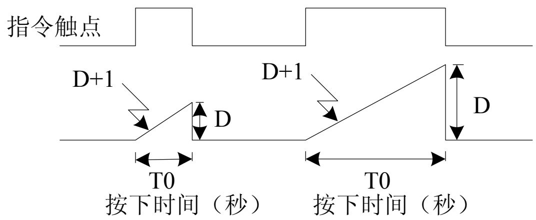
指令说明

TTMR D N

D：保存示教数据的软元件编号（D：示教时间 D+1：按下时间的当前值）

N：示教数据乘以的倍率数（0～2）

1．以秒为单位，对指令输入（按键）的按下时间进行测量，然后乘以倍率（10^N）后传送到 D 中。



2．被传到 D 中的时间是这样的，按下时间为 T0（单位：1 秒）时，根据 N 的倍率得到实际的 D 值，如下所示。

N	倍率	D
0	10^0	$T0\times10^0$
1	10^1	$T0\times10^1$
2	10^2	$T0\times10^2$

注意事项

1．当指令触点为 OFF 时，按下时间的当前值 D+1 被复位，示教时间 D 不改变。

2．软元件的占用点数，以示教时间 D 中指定的软元件为起始，占用 2 点。请注意不要与机器其他控制中使用的软元件重复。

6.14.16 STMR: 特殊定时器指令

	STMR		S M D	适用机型	MX1H	MX2H	MX2V	MX3H
							√	√
步长	7步			相关元件				

操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z				Un.b
S	INT																			*				●		
M	INT																●					●		●	
D	BOOL		●	●	●																		●		

＊ 该指令使用 T 元件时，只能使用 T0 ～ T209。

指令说明

[STMR S M D]

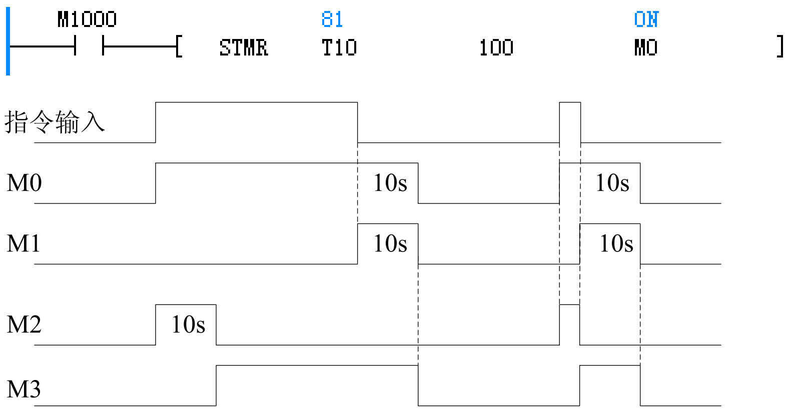
S：使用的定时器编号 T0 ～ T209（100ms 定时器） **M**：定时器的设定值（1 ～ 32,767）

D：被输出的起始位编号（占用 4 点）

将 M 指定的值作为 S 中指定的定时器的设定值，从 D 开始输出 4 点。

使用示例

断开延时定时器及单脉冲定时器：



M0：指令触点 OFF 后，经过定时器的设定时间之后，在 OFF 的断开延时定时器。

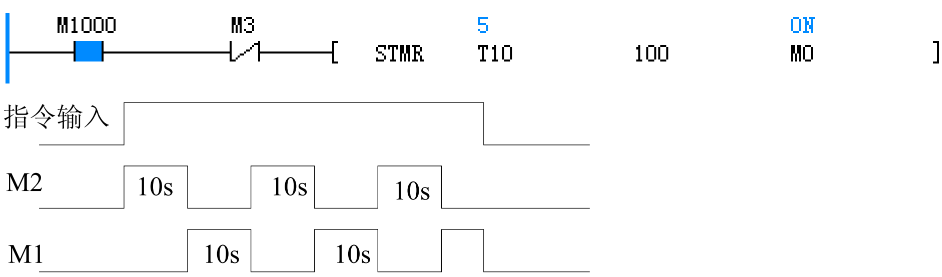
M1：指令触点从 ON 变为 OFF 后为 ON，经过定时器设定时间后变为 OFF 的单脉冲定时器。

M2：占用。用于闪烁。

M3：占用。

闪烁：

用在 D+3 的 b 触点断开该指令，通过编写如下所示的程序，在 D+1、D+2 中输出闪烁。占用 D、D+3。



M0：占用（用于断开延时定时器）。

M1：以定时器的时间间隔，重复执行 ON/OFF 的闪烁（a 触点）。

M2：以定时器的时间间隔，重复执行 ON/OFF 的闪烁（b 触点）。
M3：占用。

注意事项

1. 指定定时器的使用，这个指令中所指定的定时器编号，不能在其他普通回路中（OUT 指令等）重复使用。如果重复使用，该定时器将不能正常工作。
2. 软元件的占用点数，以 D 中指定的软元件为起始占用 4 点。请注意不要与机器其他控制中使用的软元件重复。

软元件	功能	
	断开延时定时器 / 单脉冲定时器	闪烁
D	断开延时定时器	占用
D+1	单脉冲定时器	闪烁（a 触点）
D+2	占用	闪烁（b 触点）
D+3	占用	闪烁（b 触点）

3. 当指令触点为 OFF 时，D、D+1、D+3 在经过设定时间后变为 OFF。D+2 和定时器 S 即被复位。

6.15 数据块表处理指令

6.15.1 BKADD 数据块加法指令

D	BKADD	P	S1 S2 D S3	适用机型	MX1H	MX2H	MX2V	MX3H
							√	√
步长	16位：BKADD (连续) 9步 BKADDP (脉冲) 10步 32位：DBKADD (连续) 13步 DBKADDP (脉冲) 14步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	(D)INT																●	●	*	*	●	*	●		
S2	(D)INT																●	●	*	*	●	*	●	●	
D	(D)INT																●		*	*	●	*	●		
S3	(D)INT																●	●	*	*	●	*	●	●	

* 16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z 元件。

指令说明

├──┤ ├──[BKADD S1 S2 D S3]

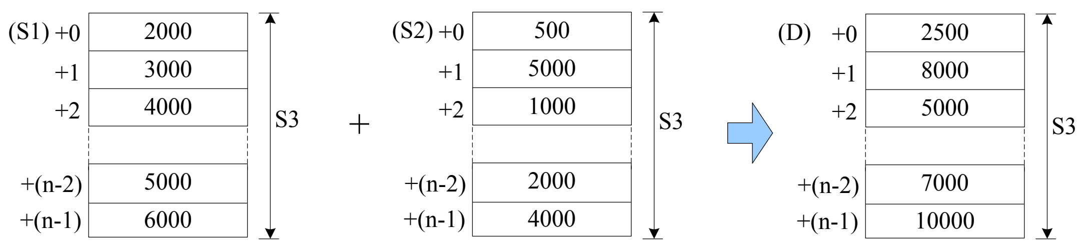
S1：保存加法运算数据的软元件起始编号

S2：要进行加法运算的常数或是保存进行加法运算的数据的软元件的起始编号

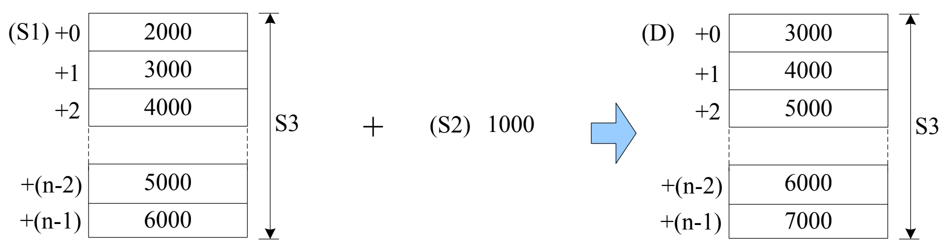
D：保存运算结果的软元件的起始编号

S3：数据的个数

将 S1 开始的 S3 个数据和 S2 开始的 S3 个数据相加后，将运算结果保存到 D 开始的 S3 个元件中。

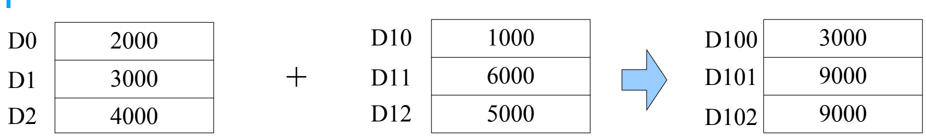


可以在 S2 中直接指定 -32768 ～ 32767（16 位）的常数。



使用示例

示例 1. 16 位运算（BKADD）



当 M0 为 ON 时，从 D0 开始的 3 个数据和 D10 开始的 3 个数据相加后，将其结果保存到 D100 开始的 3 个软元件中。

示例 2． 32 位运算（DBKADD）

(D1,D0)	2000		(D11,D10)	1000		(D101,D100)	3000
(D3,D2)	3000	＋	(D13,D12)	6000	⇨	(D103,D102)	9000
(D5,D4)	4000		(D15,D14)	5000		(D105,D104)	9000

当 M0 为 ON 时，从（D1，D0）开始的 3 个数据和（D11，D10）开始的 3 个数据相加后，将其结果保存到（D101，D100）开始的 3 个软元件中。

注意事项

1．S1、S2、D 开始的 S3 个软元件超出了相应的软元件范围时，PLC 信息中报“指令操作数元件编号范围超出”。
2．S1 开始的 S3 个软元件和 D 开始的 S3 个软元件重复时，PLC 信息中报“指令操作数数值非法”。
3．S2 开始的 S3 个软元件和 D 开始的 S3 个软元件重复时，PLC 信息中报“指令操作数数值非法”。

6.15.2 BKSUB: 数据块减法指令

D	BKSUB	P	S1 S2 D S3	适用机型	MX1H	MX2H	MX2V	MX3H
							√	√
步长	16位：BKSUB (连续) 9步 BKSUBP (脉冲) 10步 32位：DBKSUB (连续) 13步 DBKSUBP (脉冲) 14步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	(D)INT																●	●	*	*	●	*	●		
S2	(D)INT																●	●	*	*	●	*	●		
D	(D)INT																●		*	*	●	*	●		
S3	(D)INT																●	●	*	*	●	*	●	●	

＊ 16 位指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

```
├──┤ ├──[ BKSUB  S1  S2  D  S3 ]
```

S1：保存减法运算数据的软元件起始编号

S2：要进行减法运算的常数或是保存进行减法运算的数据的软元件的起始编号

D：保存运算结果的软元件的起始编号

S3：数据的个数

将 S1 开始的 S3 个数据和 S2 开始的 S3 个数据相减后，将运算结果保存到 D 开始的 S3 个元件中。

	(S1)			(S2)			(D)	
+0	2000	—	+0	500	⇒	+0	1500	S3
+1	3000		+1	5000		+1	-2000	
+2	4000		+2	1000		+2	3000	
+(n-2)	5000		+(n-2)	2000		+(n-2)	3000	
+(n-1)	6000		+(n-1)	4000		+(n-1)	2000	

可以在 S2 中直接指定 -32768 ～ 32767（16 位）的常数。

	(S1)				(D)	
+0	2000	— 1000	⇒	+0	1000	S3
+1	3000			+1	2000	
+2	4000			+2	3000	
+(n-2)	5000			+(n-2)	4000	
+(n-1)	6000			+(n-1)	5000	

使用示例

示例 1. 16 位运算（BKSUB）

```
      M0            2000     1000     1000
├──┤ ├──────[ BKSUB  D0       D10      D100     3    ]
```

D0	2000	—	D10	1000	⇒ D100	1000
D1	3000		D11	6000	D101	-3000
D2	4000		D12	5000	D102	-1000

当 M0 为 ON 时，从 D0 开始的 3 个数据和 D10 开始的 3 个数据相减后，将其结果保存到 D100 开始的 3 个软元件中。

示例 2. 32 位运算（DBKSUB）

(D1,D0)	2000		(D11,D10)	1000		(D101,D100)	1000
(D3,D2)	3000	—	(D13,D12)	6000	⇨	(D103,D102)	-3000
(D5,D4)	4000		(D15,D14)	5000		(D105,D104)	-1000

当M0为ON时，从(D1，D0)开始的3个数据和(D11，D10)开始的3个数据相减后，将其结果保存到(D101，D100）开始的 3 个软元件中。

注意事项

1. S1、S2、D 开始的 S3 个软元件超出了相应的软元件范围时，PLC 信息中报“指令操作数元件编号范围超出”。
2. S1 开始的 S3 个软元件和 D 开始的 S3 个软元件重复时，PLC 信息中报“指令操作数数值非法”。
3. S2 开始的 S3 个软元件和 D 开始的 S3 个软元件重复时，PLC 信息中报“指令操作数数值非法”。

6.15.3　BKCMP(=,>,>=,<,<=,<>)：数据块比较指令

D	BKCMP	P	S1　S2　D　S3	适用机型	MX1H	MX2H	MX2V	MX3H
							√	√
步长	16位：BKCMP　(连续)　9步　BKCMP>P　(脉冲)　10步 32位：DBKCMP　(连续)　12步　DBKCMP>P　(脉冲)　13步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	(D)INT																●	●	*	*	●	*	●		
S2	(D)INT																●	●	*	*	●	*	●		
D	BOOL		●	●	●			●	●																
S3	(D)INT																●	●	*	*	●	*	●	●	

*　16 位指令使用 C 元件时，只能使用 C0 ～ C199。
　32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z 元件。

指令说明

S1：比较值或者保存比较值的的软元件起始编号

S2：保存比较源数据的软元件起始编号

D：保存比较结果的软元件起始编号

S3：要比较的数据个数

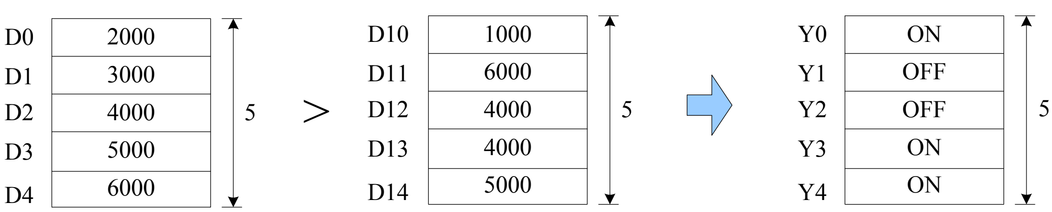
将 S1 开始的 S3 个数据和 S2 开始的 S3 个数据进行比较，将比较后的结果保存到 D 开始的 S3 个单元中。若数据符合运算，则结果输出为 ON；若数据不符合运算，则结果输出为 OFF。

16位指令	32位指令	比较结果ON的条件	比较结果OFF的条件
BKCMP=	DBKCMP=	[S1]　=　[S2]	[S1]　≠　[S2]
BKCMP>	DBKCMP>	[S1]　>　[S2]	[S1]　<=　[S2]
BKCMP>=	DBKCMP>=	[S1]　>=　[S2]	[S1]　<　[S2]
BKCMP<	DBKCMP<	[S1]　<　[S2]	[S1]　>=　[S2]
BKCMP<=	DBKCMP<=	[S1]　<=　[S2]	[S1]　>　[S2]
BKCMP<>	DBKCMP<>	[S1]　≠　[S2]	[S1]　=　[S2]

使用示例

示例 1.　16 位运算（BKCMP>）

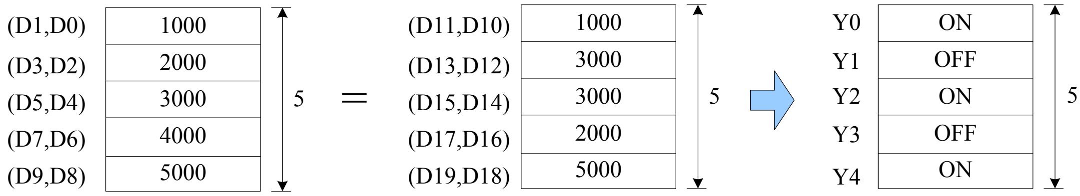
```
  M0                2000   1000   ON
--| |----[ BKCMP>   D0     D10    Y0    5  ]      LD      M0
                                                  BKCMP>  D0 D10 Y0 5
```



当 M0 为 ON 时，将 D0 ～ D4 里的 5 个数据与 D10 ～ D14 里的 5 个数据作比较，将比较后的结果给 Y0 ～ Y4。

示例 2.　32 位运算（DBKCMP=）

```
  M1                 1000   1000   ON
--| |----[ DBKCMP=   D0     D10    Y0    5  ]     LD       M1
                                                  DBKCMP=  D0 D10 Y0 5
```



当 M1 为 ON 时，将（D1，D0）～（D9，D8）里的 5 个数据与（D11，D10）～（D19，D18）里的 5 个数据作比较，将比较后的结果给 Y0 ～ Y4。

注意事项

S1、S2、D 开始的 S3 个软元件超出了相应的软元件范围时，PLC 信息中报“指令操作数元件编号范围超出”。

6.15.4 LIMIT: 上下限限位控制指令

D	LIMIT	P	S1 S2 S3 D	适用机型	MX1H	MX2H	MX2V	MX3H
							√	√
步长	16位：LIMIT (连续) 9步 LIMITP (脉冲) 10步 32位：DLIMIT (连续) 13步 DLIMITP (脉冲) 14步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	(D)INT										●	●	●	●	●	●	●	●	*	*	●	*	●	●	
S2	(D)INT										●	●	●	●	●	●	●	●	*	*	●	*	●	●	
S3	(D)INT										●	●	●	●	●	●	●	●	*	*	●	*	●		
D	(D)INT											●	●	●	●		●		*	*	●	*	●		

* 16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z 元件。

指令说明

```
├───┤ ├───[  LIMIT   S1   S2   S3   D  ]
```

S1：下限限位值（最小输出界限值）

S2：上限限位值（最大输出界限值）

S3：需要通过上下限限位控制的输入值

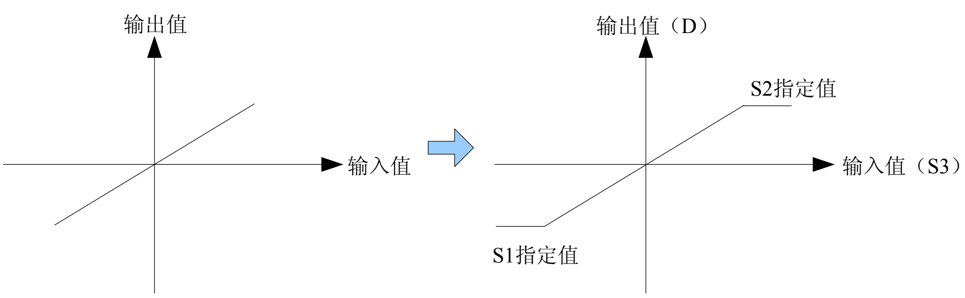
D：保存已经过上下限限位控制的输出值的软元件编号

通过判断 S3 中指定的输入值是否在 S1、S2 指定的上下限值的范围内，以此控制保存在 D 指定的软元件中的输出值。

S3 输入值 ＜ S1 下限值 时 ………………………… S1 下限值 ⟶ D 输出值

S3 输入值 ＞ S2 上限值 时 ………………………… S2 上限值 ⟶ D 输出值

S1 下限值 ≤ S3 输入值 ≤ S2 上限值 时……… S3 输入值 ⟶ D 输出值



- 仅通过上限限位值进行控制时
  16 位指令：在 S1 指定的下限限位值中设定“-32,768”。
  32 位指令：在 S1 指定的下限限位值中设定“-2,147,483,648”。
- 仅通过下限限位值进行控制时
  16 位指令：在 S2 指定的上限限位值中设定“32,767”。
  32 位指令：在 S2 指定的上限限位值中设定“2,147,483,647”。

使用示例

示例 1． 16 位运算（LIMIT）

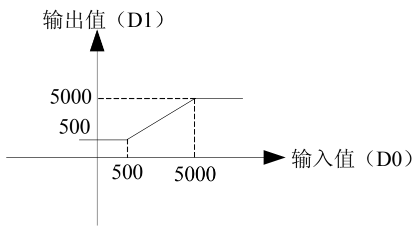
```
   M0
├───┤ ├───[ LIMIT   500   5000   D0   D1 ]
```

```
LD      M0
LIMIT   500 5000 D0 D1
```

当 M0 为 ON 时，对 D0 中的数据执行 500 ～ 5,000 的限位值控制，并将输出值保存到 D1 中。

动作：

- D0 ＜ 500 时，D1 为 500。
- 500 ＜ D0 ＜ 5000 时，D1 为 D0 的值。
- D0 ＞ 5,000 时，D1 为 5,000。



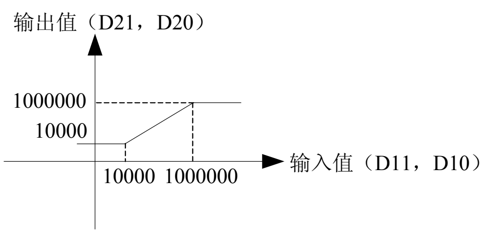
示例 2.　32 位运算（DLIMIT）

```
 M0
─┤├──[ DLIMIT   10000   1000000   D10   D20 ]      LD      M0
                                                    DLIMIT  10000 1000000 D10 D20
```

当 M0 为 ON 时，对（D11，D10）中的数据执行 10,000 ～ 1,000,000 的限位值控制，并将输出值保存到（D21，D20）中。

动作：

- （D11，D10）＜ 10000 时，（D21，D20）为 10000。
- 10,000 ＜（D11，D10）＜ 1,000,000 时，（D21，D20）为（D11，D10）的值。
- （D11，D10）＞ 1,000,000 时，（D21，D20）为 1,000,000。



注意事项

1．在使用 Kn 寻址方式时，16 位指令：1 ≤ n ≤ 4；32 位指令：5 ≤ n ≤ 8

2．当 S1 ＞ S2 时，PLC 信息中报“指令操作数数值非法错误”。

6.15.5 BAND: 死区控制指令

D	BAND	P	S1 S2 S3 D	适用机型	MX1H	MX2H	MX2V	MX3H
							√	√
步长	16位：BAND (连续) 9步 BANDP (脉冲) 10步 32位：DBAND (连续) 13步 DBANDP (脉冲) 14步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	(D)INT										•	•	•	•	•	•	•	•	*	*	•	*	•	•	
S2	(D)INT										•	•	•	•	•	•	•	•	*	*	•	*	•	•	
S3	(D)INT										•	•	•	•	•	•	•	•	*	*	•	*	•		
D	(D)INT											•	•	•	•		•		*	*	•	*	•		

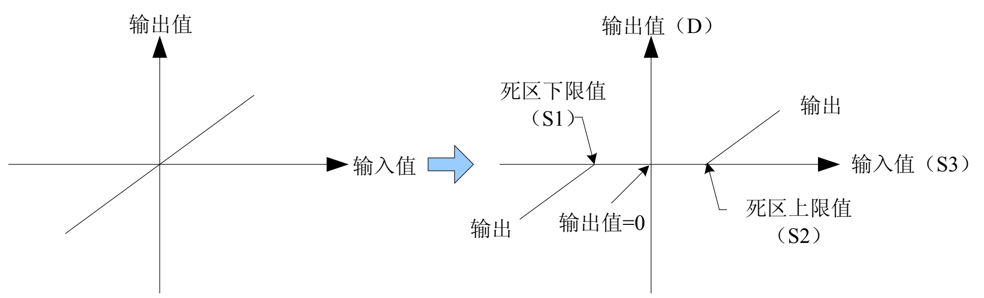
* 16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z 元件。

指令说明

├──┤ ├──[BAND S1 S2 S3 D]

S1：死区（无输出区域）的下限值
S2：死区（无输出区域）的上限值
S3：要通过死区控制的输入值
D：保存经过死区控制的输出值的软元件编号
通过判断 S3 中指定的输入值是否在 S1、S2 指定的死区上下限值的范围内，以此控制保存在 D 指定的软元件中的输出值（16 位输出范围为 -32,768 ～ 32,767，32 位输出范围为 -2,147,483,648 ～ 2,147,483,647）。

S3 输入值 ＜ S1 下限值 时 ·························· S3 输入值 − S1 下限值 ⟶ D 输出值
S3 输入值 ＞ S2 上限值 时 ·························· S3 输入值 − S2 上限值 ⟶ D 输出值
S1 下限值 ⩽ S3 输入值 ⩽ S2 上限值 时 ···································· 0 ⟶ D 输出值



输出值溢出时，如下所示。

- 16 位运算时
  输出值为带符号的 16 位 BIN 值。因此，运算结果超出 -32,768 ～ 32,767 时如下所示

死区下限值S1=10
输入值S3=-32,768 ⇨ 输出值=-32,768-10
=16#8000-16#A
=16#7FF6
=32,758

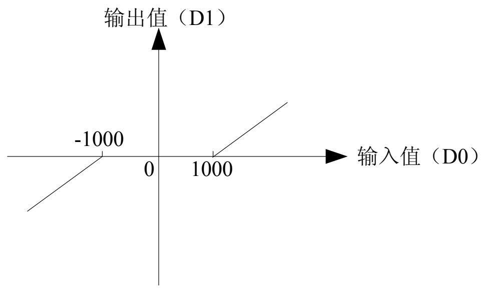
- 32 位运算时
  输出值为带符号的 32 位 BIN 值。因此，运算结果超出 -2,147,483,648 ～ 2,147,483,647 时如下所示

死区下限值S1=1000 输入值S3=-2,147,483,648 → 输出值=-2,147,483,648-1000
=16#80000000-16#000003E8
=16#7FFFFC18
=2,147,482,648

使用示例

示例 1. 16 位运算（BAND）

```
   M0
|--| |--[ BAND  -1000  1000  D0  D1 ]        LD    M0
                                             BAND  -1000 1000 D0 D1
```



当 M0 为 ON 时，对 D0 中的数据执行 -1000 ～ 1000 的死区控制，并将输出值保存到 D1 中。

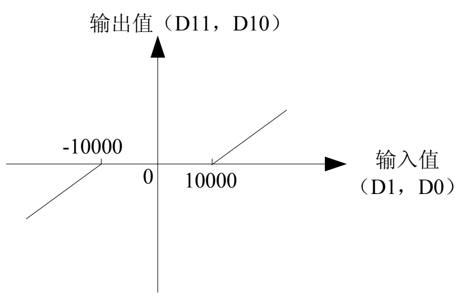
动作：

- D0 ＜ -1000 时，D1 中保存（D0）-（-1000）的值。
- -1000 ≤ D0 ≤ 1000 时，D1 中保存 0。
- 000 ＜ D0 时，D1 中保存（D0）-1000 的值。

示例 2. 32 位运算（DBAND）

```
   M0
|--| |--[ DBAND  -10000  10000  D0  D10 ]    LD     M0
                                             DBAND  -10000 10000 D0 D10
```

当 M0 为 ON 时，对（D1，D0）中的数据执行 -10,000 ～ 10,000 的死区控制，并将输出值保存到（D11，D10）中。



动作：

- （D11，D10）＜ -10,000 时，（D11，D10）中保存（D1，D0）-（-10,000）的值。
- -10,000 ≤（D1，D0）≤ 10,000 时，（D11，D10）中保存 0。
- （D11，D10）＞ 10,000 时，（D11，D10）中保存（D1，D0）-10.000 的值。

注意事项

1. 在使用 Kn 寻址方式时，16 位指令：1 ≤ n ≤ 4；32 位指令：5 ≤ n ≤ 8
2. 当 S1 ＞ S2 时，PLC 信息中报“指令操作数数值非法错误”。

6.15.6 ZONE: 区域控制指令

D	ZONE	P	S1 S2 S3 D	适用机型	MX1H	MX2H	MX2V	MX3H
							√	√
步长	16位：ZONE (连续) 9步 ZONEP (脉冲) 10步 32位：DZONE (连续) 13步 DZONEP (脉冲) 14步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	(D)INT										●	●	●	●	●	●	●	●	*	*	●	*	●	●	
S2	(D)INT										●	●	●	●	●	●	●	●	*	*	●	*	●	●	
S3	(D)INT										●	●	●	●	●	●	●	●	*	*	●	*	●		
D	(D)INT											●	●	●	●		●		*	*	●	*	●		

* 16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z 元件。

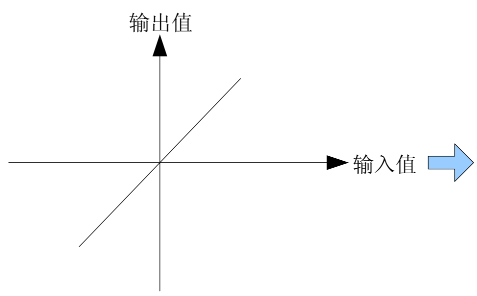
指令说明

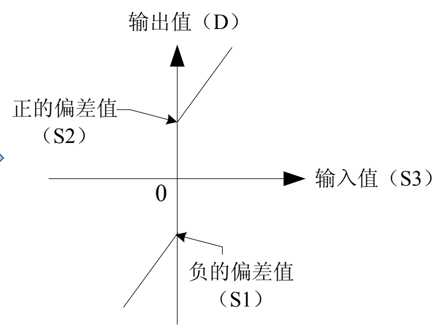
├──┤ ├──[ZONE S1 S2 D S3]

S1：加在输入值上的负偏差值
S2：加在输入值上的正偏差值
S3：要通过区域控制的输入值
D：保存已通过区域控制的输出值的软元件编号

根据 S3 中的输入值是正数还是负数，用指定的 S1 或 S2 中的偏差值来控制输出值 D 的指令（16 位输出范围为 -32,768 ～ 32,767，32 位输出范围为 -2,147,483,648 ～ 2,147,483,647）。

S3 输入值 ＜ 0 时 ………………………… S3 输入值 ＋ S1 负的偏差值 ⟶ D 输出值
S3 输入值 ＝ 0 时 …………………………………………………… 0 ⟶ D 输出值
S3 输入值 ＞ 0 时 ………………………… S3 输入值 ＋ S2 正的偏差值 ⟶ D 输出值





输出值溢出时，如下所示。

- 16 位运算时
  输出值为带符号的 16 位 BIN 值。因此，运算结果超出 -32,768 ～ 32,767 时如下所示。

  负的偏差值S1=-100
  输入值S3=-32,768
  ⇨
  输出值=-32,768+（-100）
  =16#8000+16#FF9C
  =16#7F9C
  =32,668

- 32 位运算时
  输出值为带符号的 32 位 BIN 值。因此，运算结果超出 -2,147,483,648 ～ 2,147,483,647 时如下所示。

负的偏差值S1=-1000
输入值S3=-2,147,483,648

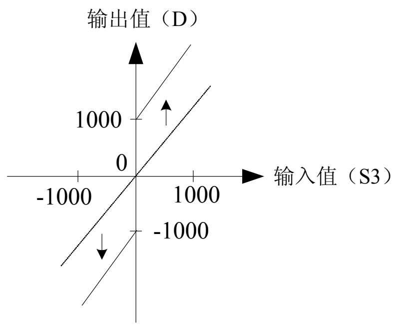
输出值=-2,147,483,648+（-1000）
=16#80000000+16#FFFFFC18
=16#7FFFFC18
=2,147,482,648

使用示例

示例 1． 16 位运算（ZONE）

```
M0
|—| |——[ ZONE  -1000  1000  D0  D1 ]
```

```
LD    M0
ZONE  -1000 1000 D0 D1
```



当 M0 为 ON 时，对 D0 中的数据执行 -1,000 ～ 1,000 的区域控制，并将输出值保存到 D1 中。

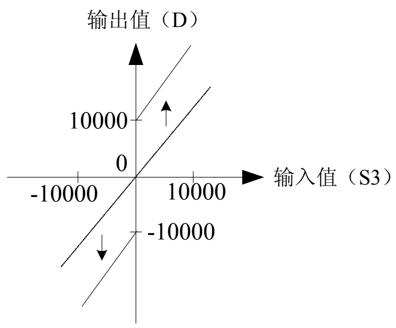
动作：

- D0 ＜ 0 时，D1 中保存（D0）+（-1,000）的值。
- D0=0 时，D1 中保存 0。
- 0 ＜ D0 时，D1 中保存（D0）+1,000 的值。

示例 2． 32 位运算（DZONE）

```
M0
|—| |——[ DZONE  -10000  10000  D0  D10 ]
```

```
LD     M0
DZONE  -10000 10000 D0 D10
```



当 M0 为 ON 时，对（D1，D0）中的数据执行 -10,000 ～ 10,000 的区域控制，并将输出值保存到（D11，D10）中。

动作：

- （D1，D0）＜ 0 时，（D11，D10）中保存（D1，D0）+（-10000）的值。
- （D1，D0）=0 时，（D11，D10）中保存 0。
- （D1，D0）＞ 0 时，（D11，D10）中保存（D1，D0）+10000 的值。

注意事项

在使用 Kn 寻址方式时，16 位指令：1 ≤ n ≤ 4；32 位指令：5 ≤ n ≤ 8。

6.15.7　SCL: 定坐标（不同点坐标数据）指令

D	SCL	P	S1　S2　D	适用机型	MX1H	MX2H	MX2V	MX3H
							√	√
步长	16位：SCL　(连续)　7步　SCLP　(脉冲)　8步 32位：DSCL　(连续)　10步　DSCLP　(脉冲)　11步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	(D)INT										●	●	●	●	●	●	●	●	*	*	●	*	●	●	
S2	(D)INT																●					*	●		
D	(D)INT											●	●	●	●		●		*	*	●	*	●		

*　16 位指令使用 C 元件时，只能使用 C0 ～ C199。
32 位指令使用 C 元件时，只能使用 C200 ～ C255，且不能使用 T、Z 元件。

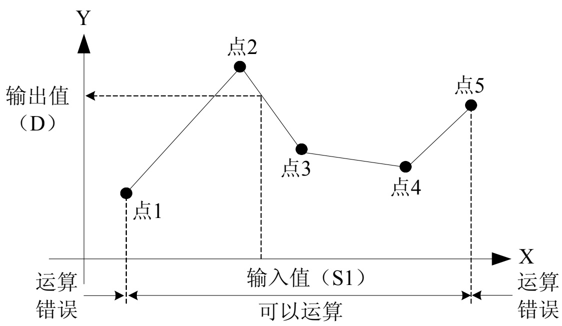
指令说明

├──┤ ├──[　SCL　S1　S2　D　]

S1：执行定坐标的输入值或保存输入值的软元件编号
S2：定坐标用的转换表格软元件的起始编号
D：保存已通过区域控制的输出值的软元件编号

根据指定的转换特性，对 S1 指定的输入值执行定坐标，然后保存在 D 指定的软元件编号中。定坐标用的转换，是依据保存在 S2 指定的软元件开始的数据表格执行的。其中，S2 中为坐标点数（假如为 5），S2+1 ～ S2+10 两两为 X、Y 坐标值。

但是，输出数据不是整数值时，小数第 1 位四舍五入后输出。



设定项目		设定数据表格的软元件分配
坐标点数（当为左图时变成“5”）		（S2）
点 1	X 坐标	（S2）+1
	Y 坐标	（S2）+2
点 2	X 坐标	（S2）+3
	Y 坐标	（S2）+4
点 3	X 坐标	（S2）+5
	Y 坐标	（S2）+6
点 4	X 坐标	（S2）+7
	Y 坐标	（S2）+8
点 5	X 坐标	（S2）+9
	Y 坐标	（S2）+10

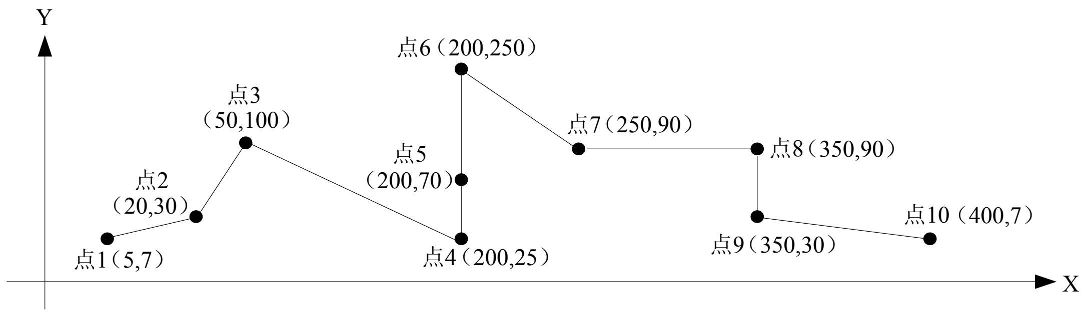
定坐标用转换表格的设定：

定坐标用的转换表格，是依据保存在 S2 指定的软元件开始的数据表格执行的。数据表格的结构如下所示。

设定项目		设定数据表格的软元件分配	
		16 位运算	32 位运算
坐标点数		（S2）	[（S2）+1，（S2）]
点 1	X 坐标	（S2）+1	[（S2）+3，（S2）+2]
	Y 坐标	（S2）+2	[（S2）+5，（S2）+4]
点 2	X 坐标	（S2）+3	[（S2）+7，（S2）+6]
	Y 坐标	（S2）+4	[（S2）+9，（S2）+8]
…	…	…	…
点 n	X 坐标	（S2）+2n-1	[（S2）+4n-1，（S2）+4n-2]
	Y 坐标	（S2）+2n	[（S2）+4n+1，（S2）+4n]

定坐标用转换表格的设定例子：

定坐标用转换表格的设定例子中例举了 16 位运算时的例子。执行 32 位运算时，请用 32 位数据设定各设定项目中的数据。当为下图所示的定坐标用转换特性时，请设定成如下所示的数据表格。



定坐标用转换设定数据表格的设定：

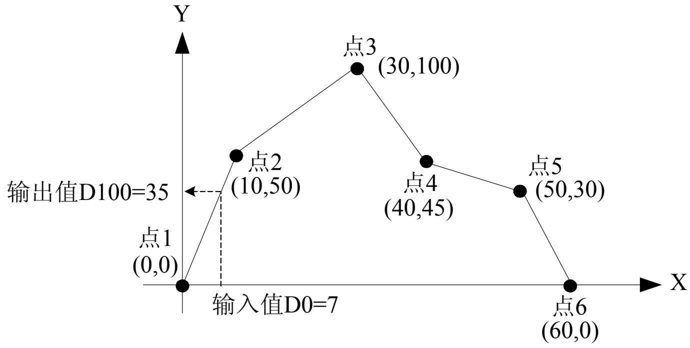
设定项目		设定数据表格的软元件分配			备注
		S2 中指定了 D0 时		设定内容	
坐标点数		（S2）	D0	10	
点 1	X 坐标	（S2）+1	D1	5	
	Y 坐标	（S2）+2	D2	7	
点 2	X 坐标	（S2）+3	D3	20	
	Y 坐标	（S2）+4	D4	30	
点 3	X 坐标	（S2）+5	D5	50	
	Y 坐标	（S2）+6	D6	100	
点 4	X 坐标	（S2）+7	D7	200	如果像这样指定 3 点的坐标，则输出值为中间值。 这个例子中，将点 5 的 Y 坐标指定为输出值（中间值）。 此外，3 点以上的 X 坐标相同时，也输出第 2 点的数值。
	Y 坐标	（S2）+8	D8	25	
点 5	X 坐标	（S2）+9	D9	200	
	Y 坐标	（S2）+10	D10	70	
点 6	X 坐标	（S2）+11	D11	200	
	Y 坐标	（S2）+12	D12	250	
点 7	X 坐标	（S2）+13	D13	250	
	Y 坐标	（S2）+14	D14	90	如果像这样指定 2 点的坐标，则输出值取后一个点的 Y 坐标值。 这个例子中，将点 9 的 Y 坐标指定为输出 值。
点 8	X 坐标	（S2）+15	D15	350	
	Y 坐标	（S2）+16	D16	90	
点 9	X 坐标	（S2）+17	D17	350	
	Y 坐标	（S2）+18	D18	30	
点 10	X 坐标	（S2）+19	D19	400	
	Y 坐标	（S2）+20	D20	7	

使用示例

```
      M0
|----| |----[ SCL    D0    D10    D100 ]        LD     M0
                                                SCL    D0 D10 D100
```

当 M0 为 ON 时，根据 D10 开始的软元件设定的坐标转换表格，对 D0 输入的值执行定坐标，然后输出到 D100 中。

动作：



设定项目		软元件	设定内容
坐标点数		D10	6
点 1	X 坐标	D11	0
	Y 坐标	D12	0
点 2	X 坐标	D13	10
	Y 坐标	D14	50
点 3	X 坐标	D15	30
	Y 坐标	D16	100
点 4	X 坐标	D17	40
	Y 坐标	D18	45
点 5	X 坐标	D19	50
	Y 坐标	D20	30
点 6	X 坐标	D21	60
	Y 坐标	D22	0

注意事项

1．在使用 Kn 寻址方式时，16 位指令：$1 \leqslant n \leqslant 4$；32 位指令：$5 \leqslant n \leqslant 8$。

2．当 S1 在数据表格设定的范围以外时，指令不执行，在 PLC 信息中报“指令操作数数值非法错误”。

3．若数据表格中的 X 坐标没有按照升序排列时，PLC 信息中报“指令操作数数值非法”错误。但是，由于运算是从数据表格的软元件编号的低位侧开始检索的，所以即使数据表格的一部分没有按照升序排列，但到这个部分为止的运算不会出现运算错误，指令会被执行。

6.15.8 SER: 数据检索指令

D	SER	P	S1 S2 D S3	适应机型	MX1H	MX2H	MX2V	MX3H
							√	√
步长	16位： SER (连续) 9步 SERP (脉冲) 10步 32位： DSER (连续) 13步 DSERP (脉冲) 14步			相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	(D)INT										●	●	●	●	●	●	●	●	*	*	●	*	●		
S2	(D)INT										●	●	●	●	●	●	●	●	*	*	●	*	●	●	
D	(D)INT											●	●	●	●		●		*	*	●	*	●		
S3	INT																●							*	

* 该指令使用 C 元件时，S1、S2、S3 只能使用 C0 ～ C199，32 位指令不能使用 T、Z。
* S3 常数范围是 1 ～ 256。

指令说明

SER S1 S2 D S3

S1：检索相同数据、最大值、最小值的起始软元件编号

S2：检索相同数据、最大值、最小值的参考值或其保存目标软元件编号

D：检索相同数据、最大值、最小值后，保存这些个数及位置的起始软元件编号

S3：检索相同数据、最大值、最小值的个数

1. 对以 S1 开始的 S3 个数据进行检索，检索与 S2 的数据相同的数据，并将结果存在 D ～ D+4 单元中。
2. 被检索的数据存在相同数据时，在以 D 开始的 5 个软元件中，保存相同数据的个数、初次 / 最终的位置以及最小值、最大值的位置；
3. 被检索的数据不存在相同数据时，在以 D 开始的 5 个软元件中，, 相同数据的个数、初次 / 最终的位置被赋值为 0，后两个软元件保存最小值、最大值的位置。

使用示例

检索结果表格的结构及数据示

被检索软元件S1	被检索数据的值	比较数据S2的值	数据的位置	检索结果		
				最大值D+4	一致D	最小值D+3
S1	100		0		○	
S1+1	111		1			
S1+2	100		2		○	
S1+3	98	100	3			
S1+4	123		4	○		
S1+5	66		5			○
S1+6	95		6			

检索结果表格：

软元件编号	内容	检索结果项目
D	2	相同数据的个数
D+1	0	相同数据的位置（初次）
D+2	2	相同数据的位置（最终）
D+3	5	最小值的最终位置
D+4	4	最大值的最终位置

注意事项

1．大小的比较，以代数方式执行。（-10<2）。

2．数据中存在多个最小值、最大值时，分别保存最后一个的位置。

3．软元件的占用点数，驱动本指令后，检索结果 D 会占用如下的软元件点数。请注意不要与机械其他控制中使用的软元件重复。

（1）16 位运算时

占用 (D、D+1、D+2、D+3、D+4)5 点

（2）32 位运算时

占用 (D+1，D)、（D+3，D+2）、(D+5，D+4)、(D+7，D+6)、（D+9，D+8）10 点

6.16 字符串指令

6.16.1 STRADD: 字符串的结合指令

STRADD	P	S1 S2 D	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√
步长	16位：STRADD (连续) 7步 STRADDP (脉冲) 8步 32位： ----- -----		相关元件				

操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT										•	•	•	•	•	•	•	•	*	•	•	•	•		
S2	INT										•	•	•	•	•	•	•	•	*	•	•	•	•		
D	INT										•	•	•	•	•		•		*	•	•	•	•		

＊ 该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

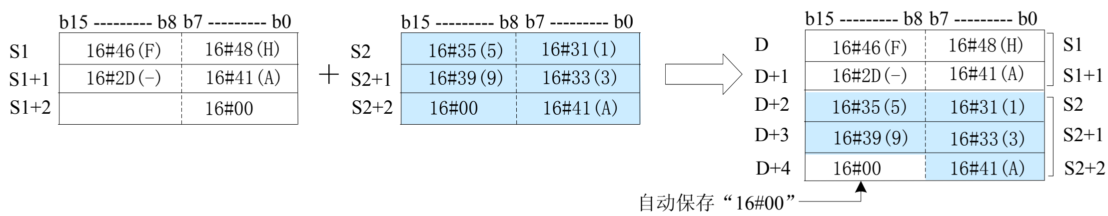
```
├────┤ ├───[ STRADD   S1   S2   D ]
```

S1：保存连接源数据（字符串）的软元件起始编号，或是被直接指定的字符串

S2：保存要连接的数据（字符串）的软元件起始编号，或是被直接指定的字符串

D：保存连接后的数据（字符串）的软元件起始编号

在 S1 开始的字符串数据后面连接 S2 开始的字符串数据，然后保存到 D 以后的软元件中。S1 和 S2 的字符串，是指以字节为单位从被指定的软元件开始到检测到第一个 16#00 的位置为止的数据。



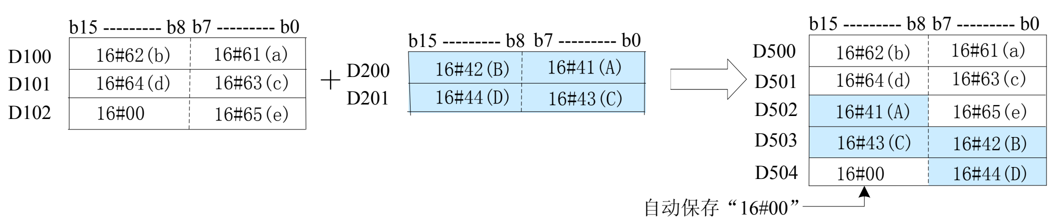
字符串的结合，就是指忽略 S1 中表示指定字符串末尾的“16#00”，接着 S1 的最后字符连接 S2 中指定的字符串。此外，执行字符串的结合后会自动将“16#00”附加在最后。

连接后的字符数为奇数时，在保存最后字符的软元件的高字节中保存“16#00”。

连接后的字符数为偶数时，在保存最后字符的软元件的下一个软元件中保存“16#0000”。

使用示例

```
      X0            25185    16961    25185
├────┤ ├───[ STRADD  D100     D200     D500    ]
```



注意事项

1. 直接指定字符串时，可以指定（输入）的字符数最多为 32 个字符。但是，S1 和 S2 中被指定了字软元件时，字符数没有限制。
2. S1、S2 中任何一个的值都是从“16#00”开始时（字符数为 0 时），在 D 中保存“16#0000”。
3. 以下一些情况会发生运算错误：D 中指定的软元件编号开始的软元件数，比保存所有已经结合的字符串所需的软元件数量更少；**S1** 和 **S2** 中指定的保存字符串的软元件和 **D** 中指定的保存字符串的软元件编号重复；**S1** 和 **S2** 指定的软元件开始的相应软元件范围中 **16#00** 未被设定。

6.16.2　STRLEN: 检测出字符串的长度

STRLEN	P	S　D	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√
步长	16位：STRLEN　(连续)　5步　STRLENP　(脉冲)　6步 32位：　-----　-----		相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S	INT										●	●	●	●	●	●	●	●	*	●	●		●		
D	INT										●	●	●	●	●		●		*	●	●	●	●		

＊ 该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

```
├──┤ ├──[　STRLEN　S　D　]
```

S：保存要检测出字符数的字符串的软元件起始编号

D：保存已检测出的字符串的长度（字节数）的软元件编号

检测出以 S 开头的字符串的长度，将字符串的长度保存到 D 中。S 以字节为单位，将从开始到 1 个保存有 16#00 的软元件编号为止的数据，作为字符串进行处理。

	b15 --------- b8	b7 --------- b0
S +0	第2个字符	第1个字符
+1	第4个字符	第3个字符
+2	第6个字符	第5个字符
≀	≀	
+n	16#00	第n个字符

表示字符串的结束

	b15 --------- --------- b0
D	字符串的长度

使用示例

```
   X0                25185      5
├──┤ ├──[ STRLEN   D100       D500      ]
```

	b15 --------- b8	b7 --------- b0
S +0	16#62 (b)	16#61 (a)
+1	16#64 (d)	16#63 (c)
+2	16#00	16#65 (e)

	b15 --------- --------- b0
D	5

D100 开始的序列单元中保存着“abcde”时，指令执行后，D500 中保存着 5

注意事项

以下一些情况会发生运算错误：

1．S 指定的软元件编号开始的相应软元件范围内没有设定 16#00。

2．检测出的字符数超过 32768 个。

6.16.3 STRRIGHT: 从字符串右侧开始取出指令

STRRIGHT	P	S1 D S2	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√
步长	16位：STRRIGHT（连续）7步 STRRIGHTP（脉冲）8步 32位： ----- -----		相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT										●	●	●	●	●	●	●	●	*	●	●		●		
D	INT											●	●	●	●		●		*	●	●		●		
S2	INT											●	●	●	●	●	●	●	*	●	●	●	●	●	

* 该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

├──┤ ├──[STRRIGHT S1 D S2]

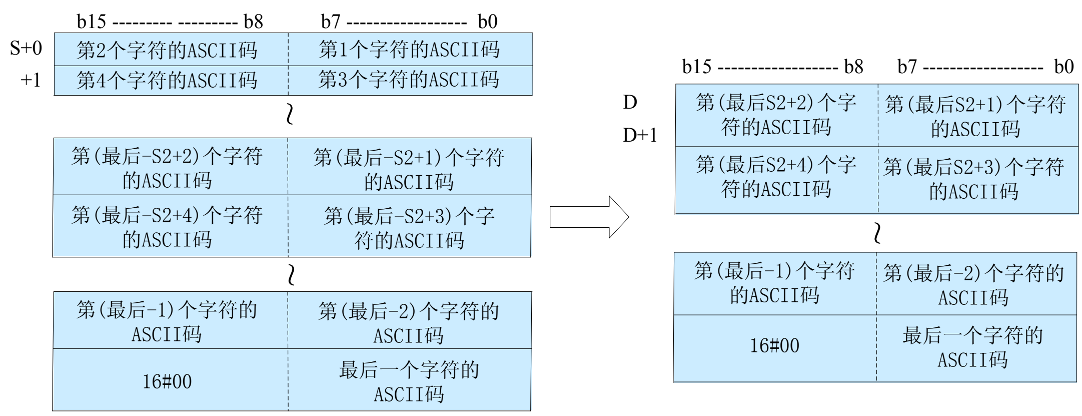
S1：保存字符串的软元件起始编号

D：保存被取出的字符串的软元件起始编号

S2：要取出的字符数

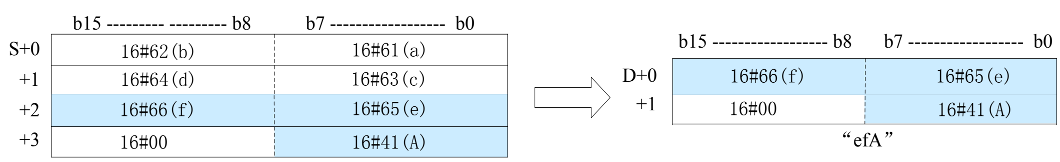
从 S1 开始的软元件中保存的字符串数据的右侧（字符串的末尾）取出 S2 个字符的数据，保存到 D 开始的软元件中。但是，S2 个指定的字符数为“0”时，D 中保存 NULL 代码（16#0000）。

此外，取出字符串时，会在最后自动附加“16#00”。（取出的字符数为奇数时，在保存最后字符的软元件的高字节中保存“16#00”；取出的字符数为偶数时，在保存最后字符的软元件的下一个软元件中保存“16#0000”）。



使用示例

```
  X0                25185       26213
├──┤ ├──[ STRRIGHT  D100        D500        3        ]
```



注意事项

1. D100 开始的序列单元中保存着“abcdef A”时，指令执行后，D500 中保存着“efA”。
2. 以下一些情况下会发生运算错误：S1 中指定的软元件开始的相应软元件范围内没有设定 16#00；S2 超出 **S1** 中指定的字符数；**D** 中指定的软元件编号开始的软元件数，比保存已经取出的字符串（**S2** 个字符）所需的软元件数更少；**S2** 为负数。

6.16.4 STRLEFT: 从字符串的左侧开始取出

STRLEFT	P	S1 D S2	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√
步长	16位：STRLEFT（连续）7步 STRLEFTP（脉冲）8步 32位： ----- -----		相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT										•	•	•	•	•	•	•	•	*	•	•		•		
D	INT											•	•	•	•		•		*	•	•		•		
S2	INT											•	•	•	•	•	•	•	*	•	•	•	•	•	

* 该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

├──┤ ├──[STRLEFT S1 D S2]

S1：保存字符串的软元件起始编号

D：保存被取出的字符串的软元件起始编号

S2：要取出的字符数

从 S1 开始的软元件中保存的字符串数据的左侧（字符串的开头）取出 S2 个字符的数据，保存到 D 开始的软元件中。但是，S2 个指定的字符数为“0”时，D 中保存 NULL 代码（16#0000）。

此外，取出字符串时，会在最后自动附加“16#00”。（取出的字符数为奇数时，在保存最后字符的软元件的高字节中保存“16#00”；取出的字符数为偶数时，在保存最后字符的软元件的下一个软元件中保存“16#0000”）。

	b15 --------- --------- b8	b7 ------------------ b0
S+0	第2个字符的ASCII码	第1个字符的ASCII码
+1	第4个字符的ASCII码	第3个字符的ASCII码
	≀	
	第S2-1个字符的ASCII码	第S2-2个字符的ASCII码
	第S2+1个字符的ASCII码	第S2个字符的ASCII码
	≀	
	16#00	最后一个字符的ASCII码

⇨

	b15 ------------------ b8	b7 ------------------ b0
D	第2个字符的ASCII码	第1个字符的ASCII码
D+1	第4个字符的ASCII码	第3个字符的ASCII码
	≀	
	第S2-1个字符的ASCII码	第S2-2个字符的ASCII码
	16#00	第S2个字符的ASCII码

使用示例

```
  X0                25185      25185
├──┤ ├──[ STRLEFT  D100       D500      3        ]
```

	b15 --------- --------- b8	b7 ------------------ b0
S+0	16#62(b)	16#61(a)
+1	16#64(d)	16#63(c)
+2	16#66(f)	16#65(e)
+3	16#00	16#41(A)

⇨

	b15 ------------------ b8	b7 ------------------ b0
D+0	16#62(b)	16#61(a)
+1	16#00	16#63(c)

“abc”

D100 开始的序列单元中保存着“abcdef A”时，指令执行后，D500 中保存着“abc”。

注意事项

以下一些情况会发生运算错误：

1. S 指定的软元件编号开始的相应软元件范围内没有设定 16#00。
2. S2 超出了 S1 中指定的字符数。
3. 中指定的软元件编号开始的软元件数，比保存已经取出的字符串（S2 个字符）所需的软元件数更少；**S2** 为负数。

6.16.5 STRMIDR: 从字符串中的任意取出指令

STRMIDR	P	S1 D S2	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√
步长	16位：STRMIDR （连续） 7步 STRMIDRP （脉冲） 8步 32位： ----- -----		相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT										●	●	●	●	●	●	●	●	＊	●	●		●		
D	INT											●	●	●	●		●		＊	●	●		●		
S2	INT										●	●	●	●	●	●	●	●	＊	●	●	●	●		

＊ 该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

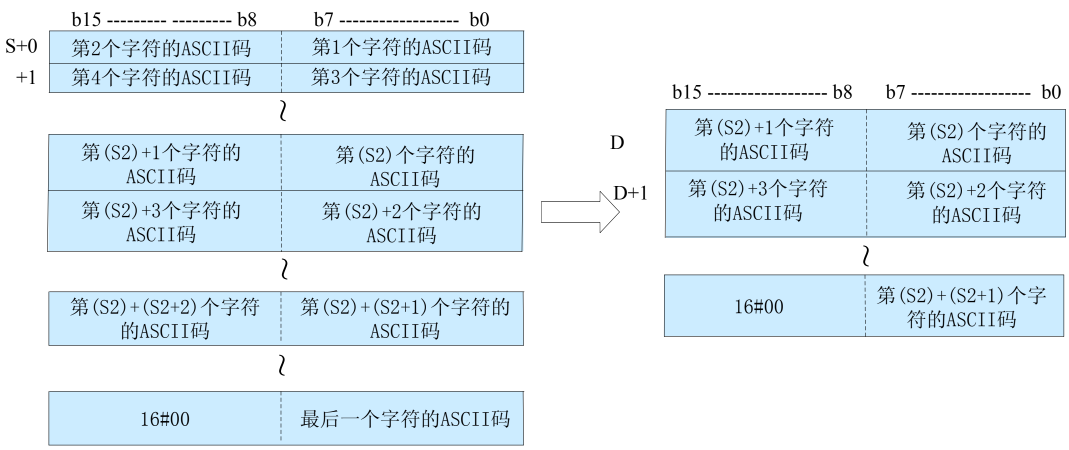
```
├──┤ ├──[ STRMIDR  S1  D  S2 ]
```

S1：保存字符串的软元件起始编号

D：保存被取出的字符串的软元件起始编号

S2：指定要取出的字符的起始位置以及字符数的软元件起始编号（S2 指定起始字符位置，S2+1 指定字符数）从 S1 开始的软元件中保存的字符串数据的左侧（字符串的开头）起第 S2 个字符的数据开始，取出 S2+1 个字符数据，保存到 D 开始的软元件中。

此外，取出字符串时，会在最后自动附加“16#00”。（取出的字符数为奇数时，在保存最后字符的软元件的高字节中保存“16#00”；取出的字符数为偶数时，在保存最后字符的软元件的下一个软元件中保存“16#0000”）。



使用示例

```
      X0                   25185     26213     5
├──┤ ├──[ STRMIDR          D100      D500      D0        ]
```

	b15 --------- --------- b8	b7 ------------------ b0
S1+0	16#62(b)	16#61(a)
+1	16#64(d)	16#63(c)
+2	16#66(f)	16#65(e)
+3	16#42(B)	16#41(A)
+4	16#44(D)	16#43(C)
+5	16#46(F)	16#45(E)
+6	16#48(H)	16#47(G)
+7	16#50(J)	16#49(I)

S2+0	5
S2+1	3

	b15 ------------------ b8	b7 ------------------ b0
D+0	16#66(f)	16#65(e)
+1	16#00	16#41(A)

“efA”

D100 开始的序列单元中保存着“abcdef　ABCDEFGHIJ”时，指令执行后，D500 中保存着“efA”。

注意事项

以下一些情况会发生运算错误：

1．S1 中指定的软元件开始的相应软元件范围内没有设定 16#00。
2．S2 超出了 S1 中指定的字符数。
3．D 开始的 S2+1 的字符数超出了 D 的软元件范围。
4．D 中指定的软元件编号开始的软元件数，比保存已经取出的字符串（S2 个字符）所需的软元件数更少；S2 为负数。
5．S2+1 为 -2 以下的值时。
6．S2+1 超出了 S1 的字符数。

6.16.6 STRMIDW: 字符串中的任意替换指令

STRMIDW	P	S1 D S2	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√
步长	16位：STRMIDW (连续) 7步 STRMIDWP (脉冲) 8步 32位： ----- -----		相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT										●	●	●	●	●	●	●	●	*	●	●		●		
D	INT											●	●	●	●		●		*	●	●		●		
S2	INT										●	●	●	●	●	●	●	●	*	●	●	●	●		

* 该指令使用 C 元件时，只能使用 C0 ～ C19

指令说明

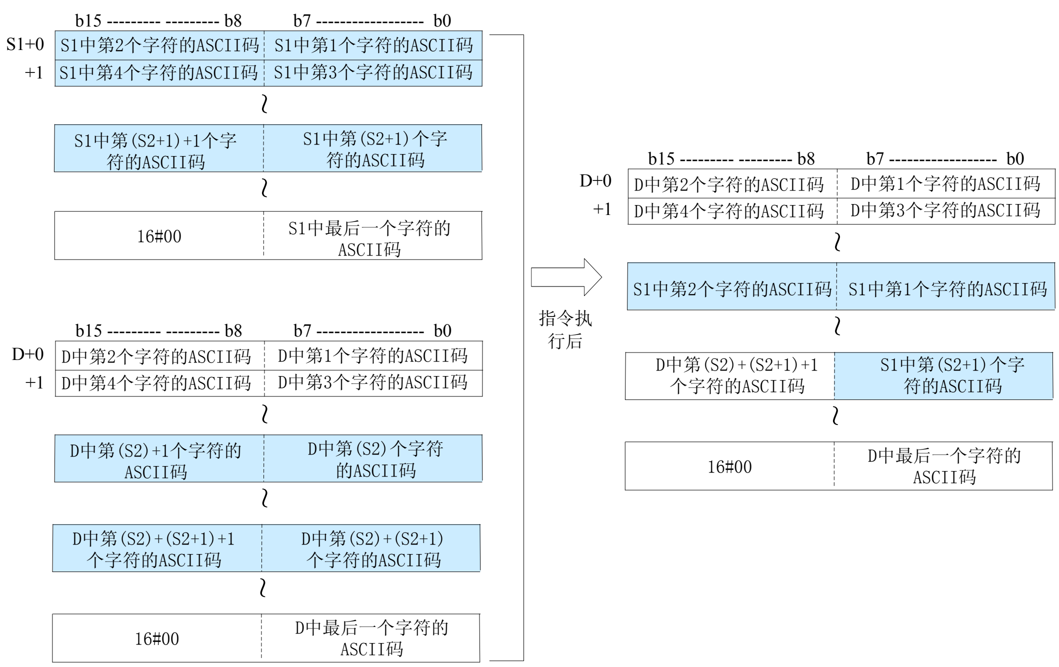
├──┤ ├──[STRMIDW S1 D S2]

S1：保存要替换的字符串的软元件起始编号

D：保存替换后的字符串的软元件起始编号

S2：指定要替换的字符的起始位置以及字符数的软元件起始编号（S2 指定被替换的字符串的起始字符位置，S2+1 指定要替换的字符数）

1．从 S1 开始的软元件中保存的字符串数据的左侧（字符串的开头）开始的字符数据，（字符个数为 S2+1），替换保存到 D 起始软元件所保存的字符串中，替换位置从 D 软元件左起第 S2 个开始，替换字符数为 S2+1。
2．S1 中 D 被指定的字符串（数据），就是指从指定的软元件开始到检测开头到第 1 个 16#00 为止的数据。
3．S2+1 中指定的要替换的字符数为“0”时不执行处理。
4．S2+1 中指定的要替换的字符数，一旦超出了 D 开始的字符串数据的最后字符时，则保存到最后字符为止的数据。
5．S2+1 中指定的字符数为“-1”时，到 S1 指定的最终字符数据为止的内容都被保存到 D 起始的软元件字 符串中。



使用示例

```
       X0                25185       16961      5
|----| |------[ STRMIDW  D100        D500       D0        ]
```

	b15 --------- --------- b8	b7 ------------------ b0
S1+0	16#62(b)	16#61(a)
+1	16#64(d)	16#63(c)
+2	16#66(f)	16#65(e)
+3	16#68(h)	16#67(g)
+4	16#70(j)	16#69(i)
+5	16#72(l)	16#71(k)

S2+0	5
S2+1	3

	b15 --------- --------- b8	b7 ------------------ b0
D+0	16#42(B)	16#41(A)
+1	16#44(D)	16#43(C)
+2	16#46(F)	16#45(E)
+3	16#48(H)	16#47(G)
+4	16#50(J)	16#49(I)
+5	16#52(L)	16#51(K)

⇨

	b15 --------- --------- b8	b7 ------------------ b0
D+0	16#42(B)	16#41(A)
+1	16#44(D)	16#43(C)
+2	16#62(b)	16#61(a)
+3	16#48(H)	16#63(c)
+4	16#50(J)	16#49(I)
+5	16#52(L)	16#51(K)

注意事项

以下一些情况会发生运算错误：

1．S1 和 D 中指定的软元件开始的相应软元件范围内没有设定 16#00。
2．S2 超出了 D 的字符数。
3．S2+1 的值超出了 S1 的软元件范围。
4．S2 为负数。
5．S2+1 为 -2 以下的值时。

6.16.7 STRINSTR: 字符串的检索指令

STRINSTR	P	S1　S2　D　n	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√
步长	16位：STRINSTR（连续）9步　STRINSTRP（脉冲）10步 32位：　-----　　-----		相关元件				

操作数	数据类型	位软元件									字软元件												变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z			Un.b
S1	INT										●	●	●	●	●	●	●	●	*	●	●		●		
S2	INT											●	●	●	●		●		*	●	●		●		
D	INT											●	●	●	●		●		*	●	●		●		
n	INT										●	●	●	●	●	●	●	●	*	●	●	●		●	

*　该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

├──┤ ├──[STRINSTR　S1　S2　D　n]

S1：保存要检索的字符串的软元件起始编号

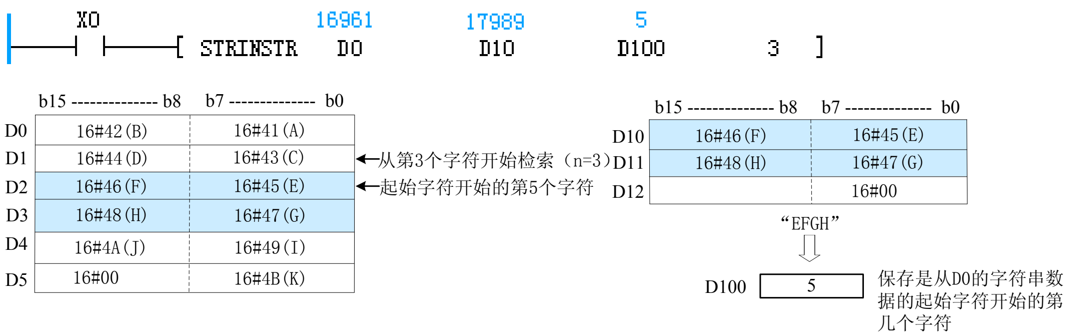
S2：保存检索源字符串的软元件起始编号

D：保存检索结果的软元件起始编号

从 S2 起始的软元件中保存的检索源字符串的坐骑（起始字符）第 n 个字符开始，检索与保存在 S1 起始的软元件中的字符串相同的字符串，然后将检索结果的字符串位置信息保存到 D 中。

检索结果，就是检索源字符串左起（起始字符）的第几个字符的字符位置信息。

使用示例



注意事项

1. 不存在一致的字符串时，在 D 中保存“0”。
2. 开始检索的位置 n 为负数或是“0”不执行处理。
3. 以下一些情况会发生运算错误

（1）开始检索的位置 n 超过了 S2 的字符数时。

（2）S1 开始的相应软元件的软元件范围内没有 16#00（NUL）时。

（3）S2 开始的相应软元件的软元件范围内没有 16#00（NUL）时。

6.16.8 STRMOV: 字符串的传送指令

STRMOV	P	S D	适用机型	MX1H	MX2H	MX2V	MX3H
						√	√
步长	16位：STRMOV（连续）5步 STRMOVP （脉冲）6步 32位： ----- -----		相关元件				

操作数	数据类型	位软元件									字软元件													变址	常数	模块
		X	Y	M	S	T	C	LM	SM	Dn.b	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z				Un.b
S	INT										●	●	●	●	●	●	●	●	*	●	●			●		
D	INT											●	●	●	●		●		*	●	●	●		●		

＊ 该指令使用 C 元件时，只能使用 C0 ～ C199。

指令说明

```
├──┤ ├──[ STRMOV   S    D ]
```

S：传送源中被直接指定的字符串，或是保存字符串的软元件起始编号

D：保存传送字符串的软元件起始编号

将 S 指定的软元件编号起始的软元件中保存的字符串数据，传送到 D 的软元件编号起始的软元件中。在传送字符串过程中，从 S 指定的软元件编号开始，一直到之后的软元件中，其高字节或低字节中包含“16#00”的软元件为止，都依次进行传送。

	b15 ------------- b8	b7 ------------- b0
S+0	第2个字符	第1个字符
+1	第4个字符	第3个字符
+2	第6个字符	第5个字符
≀	≀	
	16#00	第n个字符

⇨

	b15 ------------- b8	b7 ------------- b0
D+0	第2个字符	第1个字符
+1	第4个字符	第3个字符
+2	第6个字符	第5个字符
≀	≀	
	16#00	第n个字符

表示字符串的结束（16#00）

保存要传送的字符串数据的软元件范围 [S ～ S+n]，以及保存被传送的字符串数据的软元件范围 [D ～ D+n] 使用重复时，也能执行传送。

例如，将 D10 ～ D13 中保存的字符串传送到 D11 ～ D14 中时，如下所示：

使用示例

```
        X0              12849      12849
├──┤ ├──[ STRMOV         D0         D11   ]
```

	b15 ------------- b8	b7 ------------- b0
D10	16#32(2)	16#31(1)
D11	16#34(4)	16#33(3)
D12	16#36(6)	16#35(5)
D13		16#00
D14		

⇨

	b15 ------------- b8	b7 ------------- b0
D10	16#32(2)	16#31(1)
D11	16#32(2)	16#31(1)
D12	16#34(4)	16#33(3)
D13	16#36(6)	16#35(5)
D14	16#00	

传送前的字符串原封不动

注意事项

1．S+n 的低字节中保存了“16#00”时，D+n 的高字节、低字节都保存“16#00

	b15 ------------- b8	b7 ------------- b0
S+0	16#42(B)	16#41(A)
+1	16#44(D)	16#43(C)
+2	16#45(E)	16#00

不传送高字节

⇨

	b15 ------------- b8	b7 ------------- b0
D+0	16#42(B)	16#41(A)
+1	16#44(D)	16#43(C)
+2	16#00	16#00

高字节中，自动保存16#00

。

2．以下情况会发生运算错误：

在 D 中指定的软元件编号～指定软元件的最终编号之间的点数中，不能保存所有指定的字符串时。

第七章　顺序功能图使用指南

本章介绍了使用 MXProgrammer 进行顺序功能图编程的基本概念、编程方法以及编程的注意事项。

7.1 顺序功能图介绍

7.1.1 什么是顺序功能图

顺序功能图（Sequential Function Chart，简称 SFC）是一种近年来逐渐发展并流行起来的编程语言，用于将 PLC 编程项目划分成结构化流程。它使用 IEC61131-3 标准中规定的编程元件和语言结构，将复杂的系统过程划分为按顺序进行的多段流程步骤以及步骤间的转换过程，从而实现顺序控制功能。

由于顺序功能图编程具有直观和流程化的特点，分解后的每一步骤和每个转换条件都为相对简单的程序，符合顺序化控制应用领域的要求，因此逐渐得到了广泛应用。

MX 系列 PLC 可以使用 SFC 实现顺序控制。除了标准的顺序控制功能之外，还可在梯形图块中内置了顺序控制图流程以外的梯形图指令。

用 MX 系列 PLC 顺序功能图编写的程序可以转换成对应的梯形图和指令列表。

MX 系列 PLC 的顺序功能图程序还支持多个独立流程，数量最多可达到 20 个，这些独立流程可以分别独立运行。每个流程内的步进状态按工序分别扫描和转移，并且在各独立流程间可进行跳转。

7.1.2 顺序功能图的基本概念

顺序功能图有两个基本概念：步进状态和转移。其它的概念，如跳转、分支、多个独立流程等，都是在此基础上派生而来的。

● **步进状态**

1．步进状态的定义

一个步进状态实际为一段独立的程序，代表了顺序控制过程中的一个工作状态或一个工序。将多个步进状态进行有机的组合即可组成一个完整的顺序功能图程序。

2．步进状态的执行

在顺序功能图程序中，步进状态由固定的 S 元件来代表。

一个正在执行中的步进状态被称为是有效的步进状态，其对应的 S 元件状态为 ON，与此连接的梯形图（内置梯形图）动作。而未被执行的步进状态被称为无效的步进状态，其对应的 S 元件状态为 OFF，与此连接的梯形图（内置梯形图）不动作。

● **转移**

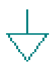
顺序控制过程是一系列的步进状态切换过程。正在执行某个步进状态的 PLC 在转移条件满足的情况下，会离开当前的步进状态，进入并执行新的步进状态。这个切换过程被称为步进状态的转移。

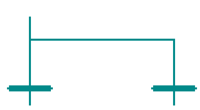
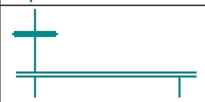
转移的发生必须满足一定的逻辑条件，这种逻辑条件被称为步进转移条件。

7.1.3 编程图元及其连接规则

● **编程图元**

MX 系列 PLC 顺序功能图由以下基本的编程图元构成。

编程图元	图形表达	具体说明
梯形图块	LAD0*	梯形图块用于表示顺序控制图流程以外的梯形图指令，可以用于起始步进的启动。
初始步进符	S0*	代表一个起始的步进状态，一个步进状态的编号为指定的 S 元件号，编号不能重复。一个顺序功能图网络的执行必须由初始步进符开始。初始步进符对应的 S 软元件地址范围是 S0 ～ S19。
普通步进符	S20*	代表一个普通的步进状态，一个步进状态的编号为指定的 S 元件号，编号不能重复。普通步进符对应的 S 软元件地址范围是 S20 ～ S1023 软元件。
转移符		代表转移，可内置使下一个步进有效的转移条件（内置梯形图），用户可以自己定义。当达到转移条件时与转移符连接的下一个 S 软元件状态置位，进入下一步进状态。转移符必须连接在步进符之间。
跳转符		跳转符，连接在转移符之后，当转移条件满足时，使指定的 S 元件为 ON。用于步进状态的循环或跳转。
重置符		重置符，连接在转移符之后，当转移条件满足时，使指定的 S 元件为 OFF。用于顺序功能图程序段的结束。

编程图元	图形表达	具体说明
选择分支符		连接在步进符之后，分别代表多个相互独立的转移条件，当达到其中任何一个转移条件时，即结束上一步进状态，进入该转移条件下对应的步进分支。用于选择多个步进分支中的一个，选择一个分支后，其它分支则不会再被选中。
选择汇合符		连接在选择分支汇合点，代表选择步进支路汇合。当达到其中一个支路的转移条件时，即转移到下一步进状态。
并行分支符		连接在步进符之后，其后的多路分支共同等待同一个转移条件。当该转移条件满足时，使后面的多个步进分支同时有效并执行。
并行汇合符		连接并行分支汇合点，转移条件代表了各个分支的结束条件之和。多个并行步进分支都执行完毕，满足转移条件后，才能使后面的一个步进状态有效。

● **编程图元连接规则**

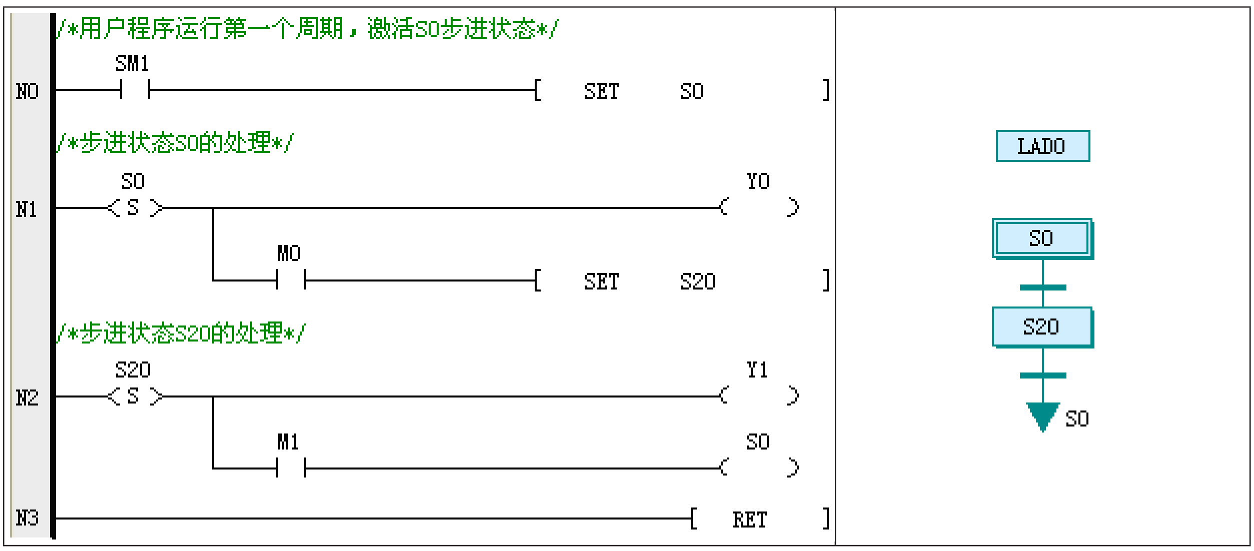
1. 初始步进符前不能连接任何图元，后接图元必须是转移符，也可不参与连接。
2. 梯形图块不能与任何图元相连接。
3. 两个普通的步进符不能直接相连，必须要有转移条件，普通步进符在图中不可孤立存在。
4. 重置符和跳转符前应连接转移符，不能后接其它元件。
5. 转移符和跳转符不可孤立存在。

7.1.4 SFC 流程结构

SFC 流程结构分为简单顺序结构、选择结构和并行结构三类。

● 简单顺序结构

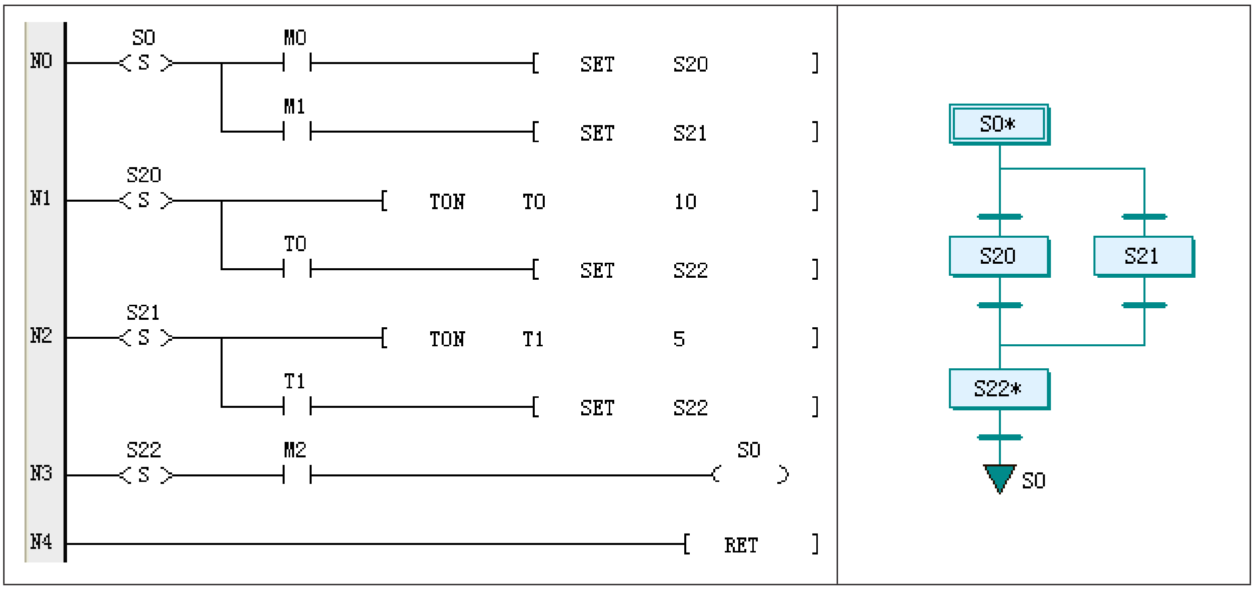
如下图为简单顺序结构及其梯形图表示的示例：



在简单的顺序结构中（其中无任何分支结构），当步进转移条件满足时，由前一个步进状态顺序转移到下一个状态。当最后一个步进状态的转移条件满足时，退出顺序功能图程序段，或者转移到初始步进状态。

● **选择分支结构**

选择分支结构示例如下图所示，左图为梯形图，右图为对应的顺序功能图。



- **选择分支**

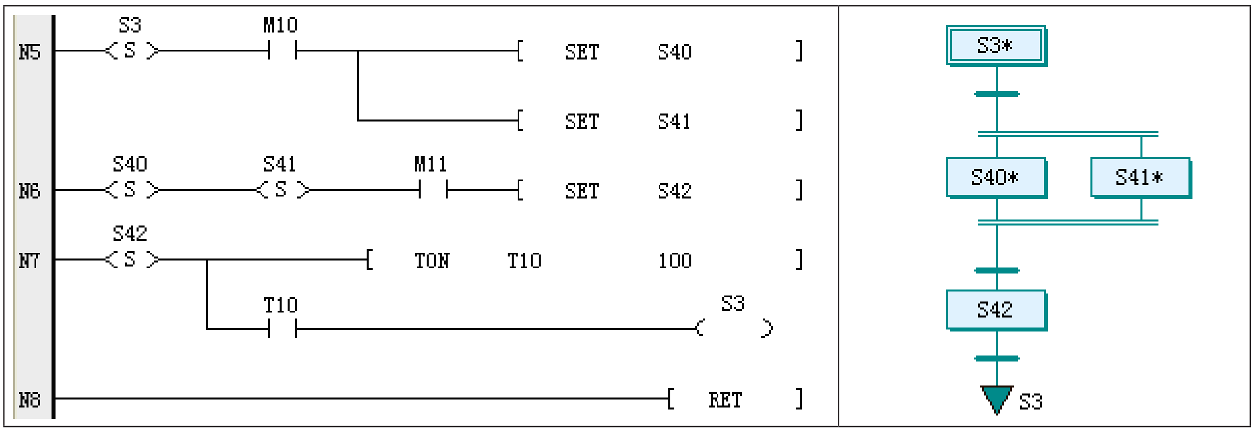
根据各分支转移条件，选择激活相应支路上的步进状态。用户必须确保分支中的转移条件互斥。因此选择结构在流程运行时，一次只能选择一个分支。如上图程序所示，N0 行程序中，S20、S21 两个步进状态分别由 M0、M1 作为转移条件，在保证 M0、M1 不会同时置位时，S20 和 S21 就只能是二者选其一。

- **选择汇合**

选择分支汇合处，所有的分支都连接到同一个步进状态，转移条件各自独立。如上图程序所示，N1 行中 S20 步进状态的转移条件是 T0 计时时间到；而 N2 行中 S21 步进状态的转移条件是 T1 计时时间到。转移结果都是进入到下一个步进状态 S22。

- **并行分支结构**

并行结构示例如下图所示，左图为梯形图，右图为对应的顺序功能图。



- **并行分支**

当并行分支结构的转移条件满足时，并行分支结构下接的所有步进状态同时激活。这也是一种常见的顺控结构，即在一定条件下，将同时处理多项并行的工序。如上图中 N5 行程序所示，M10 为转移条件，当 M10 置位，则 S40 和 S41 步进状态同时有效。

- **并行汇合**

当并行汇合结构的转移条件满足时，将会使并行分支结构上接的所有步进状态同时失效，并转移到后续步进状态。如上图 N6 行程序所示，当处在 S40 和 S41 步进状态下，M11 置位时，即转移到 S42 步进状态，结束 S40 和 S41 步进状态。汇合之前进行处理的各项独立步骤必须全部完成后，并行汇合的转移条件才能生效。

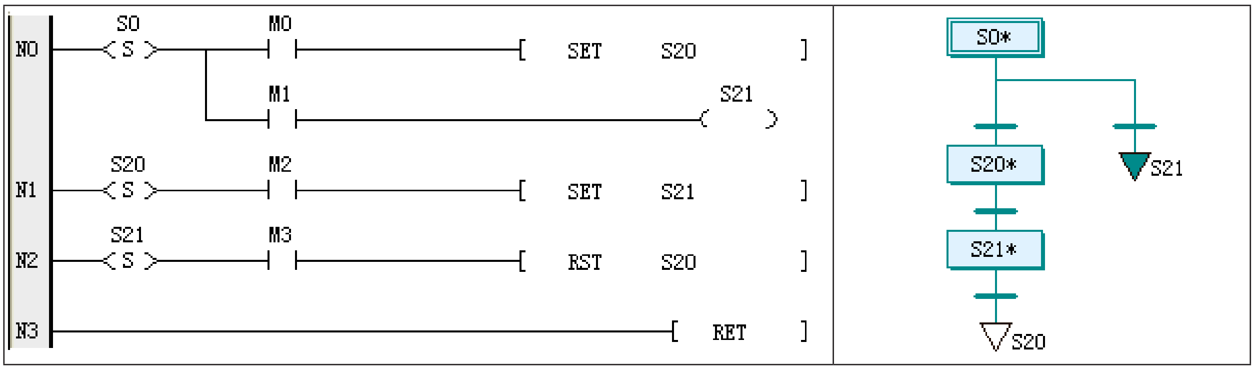
- **跳转**

跳转结构常用于：跨越部分步进状态、循环（返回起始步进状态或某一普通步进状态）、转移到其它独立流程。

1．跨越部分步进状态

一个流程中，根据一定的转移条件，当不需要顺序执行时，可以采用跳转符转移到需要执行的步进状态，跨越部分步进状态。

如下示例图，左边为梯形图，右边为对应的顺序功能图。



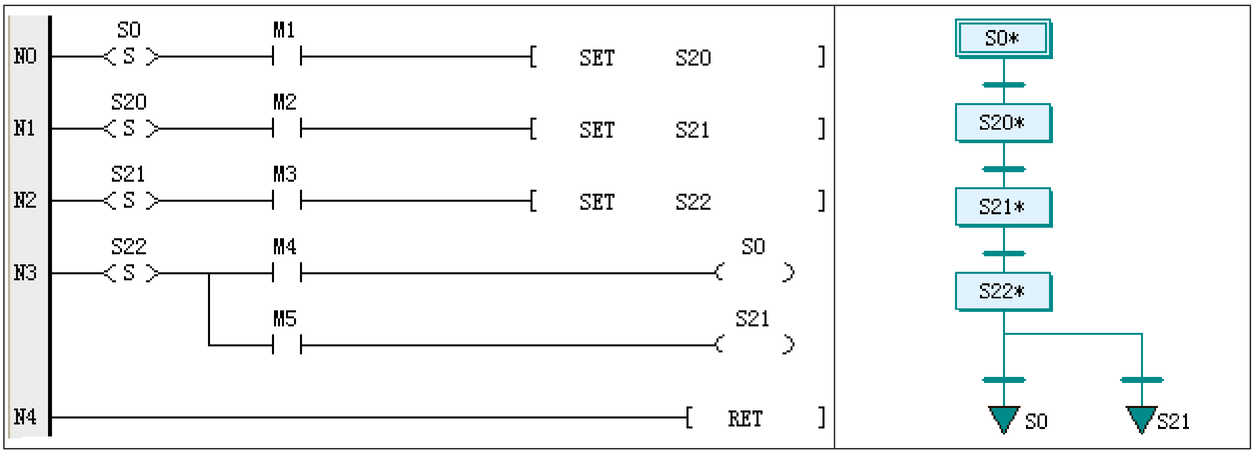
在顺序功能图中，采用 S21 跳转符表示跳转，S20 步进状态被跨越。跳转之前实际上是选择分支结构。

在梯形图中，N0 行的第二支路就是跳转指令。跳转指令采用的是 OUT 线圈的形式，而不是顺序转移的 SET 指令形式。在 S0 步进状态运行时，当 M1 为 ON，则实现跳转，跨越到 S21 状态。

2．循环

一个流程中，根据一定的转移条件，需要在部分或全部步进状态间循环时，采用跳转符实现循环的功能。在这段流程最后的转移时，跳转到之前的普通步进符，实现部分步进状态循环功能；如果是跳转到初始步进符，则实现全部步进状态循环功能。

如下为同时实现上述两种循环结构的示例程序，左图为梯形图，右图为对应的顺序功能图。



在顺序功能图中，在 S22 步进状态下，当 M5=ON 时，跳转到 S21，重新运行 S21 步进状态。当 M4=ON 时，将跳转到 S0 初始步进状态，重新运行全部步进状态。

在梯形图中，这两个循环的跳转都在 N3 行中实现，跳转使用 OUT 指令。

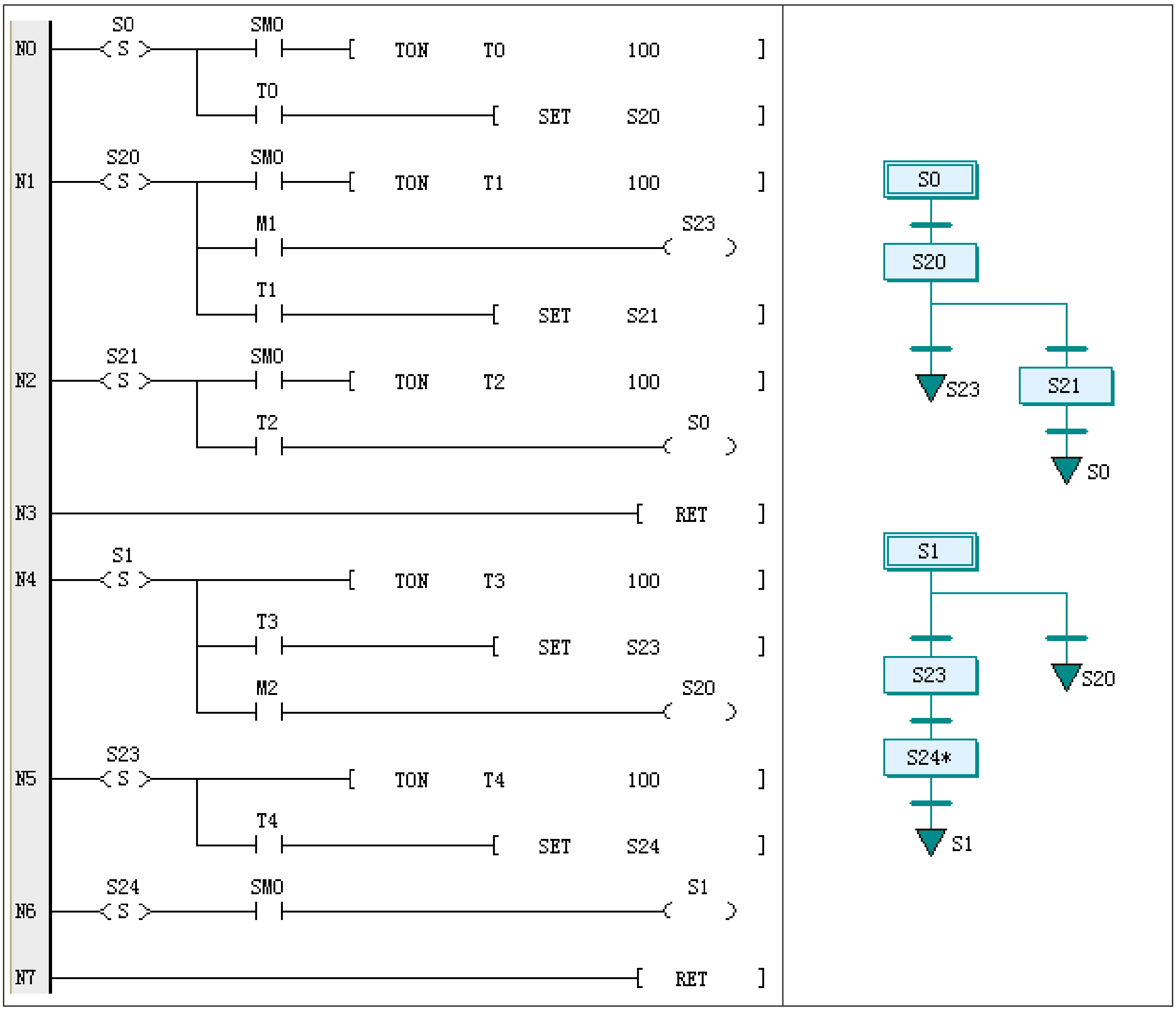
3．在不同的独立流程间跳转

MX 系列 PLC 顺序功能图程序中可以同时存在多个独立流程，支持在这些流程间的跳转。在一个独立流程中可以设置转移条件，在该条件满足的时候，直接跳转到另一个独立流程。可以跳转到另一个流程的初始步进状态，也可以跳转到普通步进状态。

☞ 注意

在多个流程之间跳转会增加 PLC 程序的复杂性，必须谨慎对待。

如下图为实现了从一个独立流程跳转到另一个独立流程的示例程序。左图为梯形图，右图为对应的顺序功能图。



在顺序功能图中，在 N1 行 S20 步进状态下，当 M1=ON 时可以跳转到 S23 步进状态，而在 N4 行 S1 步进状态下，当 M2=ON 时也可以跳转到 S20 步进状态。

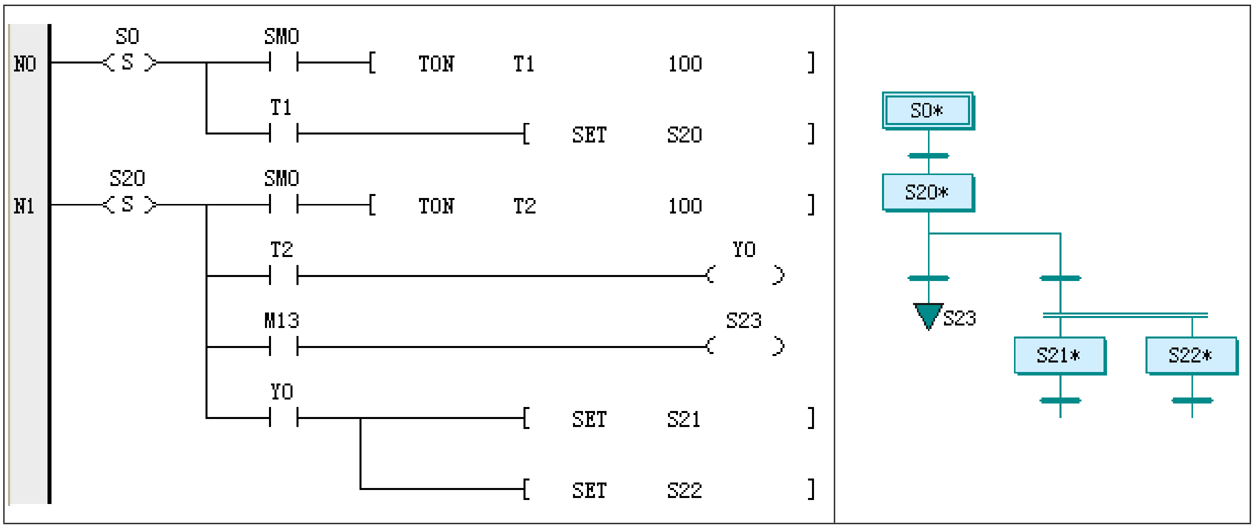
在示例图中，可以看到这种跳转是基于选择分支结构的，因此在发生不同流程间的跳转时，发生跳转的那个流程中的步进状态将全部无效。例如示例图中上面的流程 S20 步进状态下，如果转移到了下面的流程 S23 步进状态，则 S20 将会置 OFF，该独立流程所有的步进状态 S0、S20、S21 都为 OFF，也就是处于无效状态。

7.1.5 顺序功能图程序的执行

顺序功能图程序的执行过程与普通梯形图相同之处在于：两者都是连续地从上到下、自左及右地循环扫描。

顺序功能图程序的执行过程与普通梯形图不同之处在于：顺序功能图的各步进状态会根据转移条件切换有效 / 无效的状态，相应有效的步进状态的内部指令才会被扫描执行，无效的步进状态内部指令则不会被扫描执行；而普通梯形图的主程序在每次扫描周期中都会被扫描执行。

如下图所示，左图为梯形图，右图为对应的顺序功能图。当 S20 步进状态有效时，T2 计时器会被扫描并计时，在 T2 计时时间未到时不会进入 S21、S22 状态；在 M13 为 OFF 时，不会进入 S23，它们的内部指令都不会被扫描执行。



各 S 元件之间根据步进转移条件进行 ON/OFF 切换，从而实现上一步进状态到下一步进状态的转移。在一个 S 元件由 ON 跳转为 OFF 时，其内部指令的输出软元件会被复位或清零。请参见 *6.2 SFC* 指令。

☞ 注意

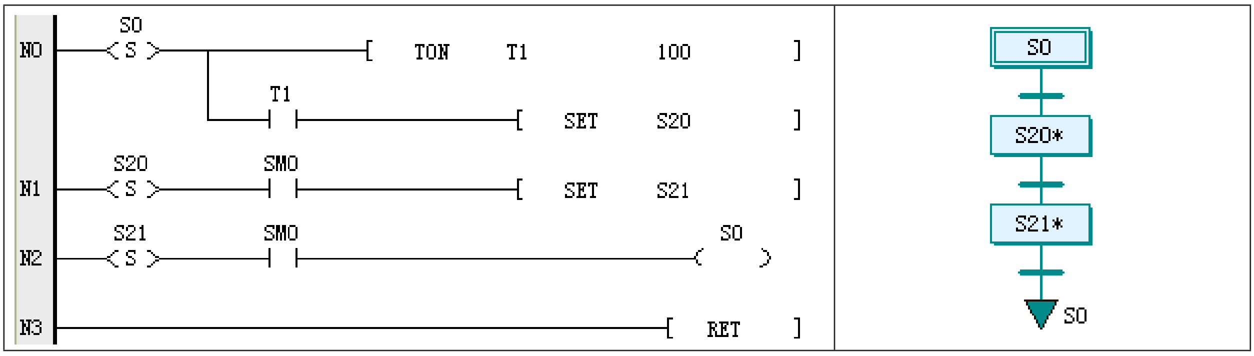
1. MX 系列 PLC 顺序功能图程序中一般都会同时包含顺序功能图和梯形图块。梯形图块用于处理流程以外的事务，包括启动顺序功能图的操作，不受任何 S 元件的控制。在每个扫描周期中，PLC 都会扫描执行梯形图块的程序。
2. 由于 S 元件的状态变化会对步进状态内置指令产生影响，且上下步进状态切换也有一个过程，因此在进行顺序功能图编程时对软元件的操作和指令的使用有一些需要注意的事项。详细情况请参见 *7.4* 顺序功能图编程注意事项及编程技巧。

7.2 顺序功能图与梯形图的对应关系

当满足一定规则后，SFC 程序和梯形图之间可以进行相互转换，且其流程是具有特定结构的。

7.2.1 STL 指令与步进状态

在梯形图中是以 STL 指令来启动一个步进状态的。每个步进状态都以 S 元件为标志。
如下左图所示为一个简单顺序结构示例程序的梯形图程序。右图为该流程的 SFC 程序。



梯形图中，S0 步进状态是从 STL 状态装载指令开始的，其后的 TON 计时器语句为 S0 步进状态的内置程序。一个步进状态的内置程序可以有多个语句，与普通梯形图程序基本一致，实际上就是一段相对完整的程序段。
初始步进状态与普通步进状态区别仅仅是选用 S 元件的范围不同。
STL 指令的详细说明请参见 *6.2 SFC* 指令。其中需要注意的是，步进状态由 ON 到 OFF 跳变时，内置的 OUT、TON、PWM、PLSY 的目的操作数将被清除，TMON 保持计数，TONR 计数值保持不变，TOF 开始计数，HCNT、PLSR、DHSCS、SPD、DHSCI、DHSCR、DHSZ、DHST、DHSP 所对应的目的操作数将被清除。

☞ 注意

由于 PLC 是连续周期性扫描的，因此，当一个步进状态转移到下一步进状态时，原步进状态的那些内置语句要到下一次扫描时才会受到 ON 到 OFF 跳变的影响。请参见 *7.4.1* 常见的编程错误。

7.2.2 SFC 状态转移

转移条件是由 SET 语句前面的常开触点元件组合而成。常开触点元件是由内置语句或外部操作来控制的。
当转移条件有效时，使指定的步进状态有效，同时使当前有效的步进状态失效，完成步进状态的转移。

7.2.3 RET 指令与 SFC 程序段

如 5.2.1 图中所示，右图的 SFC 程序，由 S0 初始步进符开始，经过 2 个普通步进符后再返回到 S0 步进符。在梯形图中，SFC 程序段必须用 RET 指令结束。
RET 指令只能在主程序中使用。

7.2.4 SFC 状态跳转符、重置符

如 5.2.1 图中，跳转符 S0 在梯形图中是 N2 行所示。采用 OUT 指令，实现了跳转。跳转可以在同一流程中，也可以在不同的独立流程之间。
如果采用了重置符 S21，在梯形图中 N2 行则为 RST 指令，实现对上一步进状态 S21 的复位。

7.2.5 SFC 选择分支、并行分支及汇合

选择分支的梯形图示例请参见 *7.1.4 SFC* 流程结构中的选择分支结构。
并行分支的梯形图示例请参见 *7.1.4 SFC* 流程结构中的 并行分支结构。

7.3 顺序功能图编程步骤

1．分析流程，确定程序流程结构
程序流程结构可分为简单顺序结构、选择结构、并行结构。采用 SFC 编程时，第一步要确定是哪一种流程结构。比如单个对象连续通过前后顺序步骤完成操作，一般是简单顺序结构；有多个产品加工选项，各选项参数不同，且不能同时加工的，则应该确定为选择结构；多个机械装置联合运行却又相对独立的，则可能是并行结构。
2．确定主要步骤和主要转移条件，得出流程草图
确定了流程结构，接下来是大体上确定主要步骤和主要转移条件。把流程结构划分为细化的操作过程，每个操作过程就是步骤，而操作过程结束的重要标志则是转移条件。这样就能得到流程的草图。
3．根据流程草图做出 SFC 顺序功能图
打开 MXProgrammer 编程软件的 SFC 编程界面，将流程草图编写成 SFC 图。此时已经能够得到可执行的 PLC 程序，只是程序有待完善。
4．列出输入输出点表，确定各步骤操作对象及实际转移条件。
输入点多为转移条件，输出点多为操作对象。根据输入输出点表，还可以进一步修正顺序功能图。
5．步骤和转移条件的输入
在 SFC 编程界面中，使用鼠标右键点击 SFC 图元，可以弹出相应的右键菜单，选择“转至内置梯形图”选项，可以打开该图元的内置梯形图编辑工作区，输入梯形图程序及条件。
6．编写梯形图程序块
不要忘记在程序中编写一些通用处理功能的梯形图程序块，比如顺序流程的启动，还有停机、报警等通用的操作。这些都需要放在梯形图块中。

☞ 注意

启动停机操作关系到人身和设备安全。要考虑到 SFC 编程的特殊性，停机时尽可能把全部应该停止运行的输出关闭。

7.4 顺序功能图编程注意事项及编程技巧

由于 STL 语句具有一些特点，且 PLC 是按语句顺序周期性扫描的，因此 SFC 编程有几个比较重要的注意事项。

7.4.1 常见的编程错误

1．重复使用步进状态符

在同一个 PLC 程序中，用于顺序控制编程的每一个步进状态符都是对应于一个唯一的 S 元件的，不可重复使用。

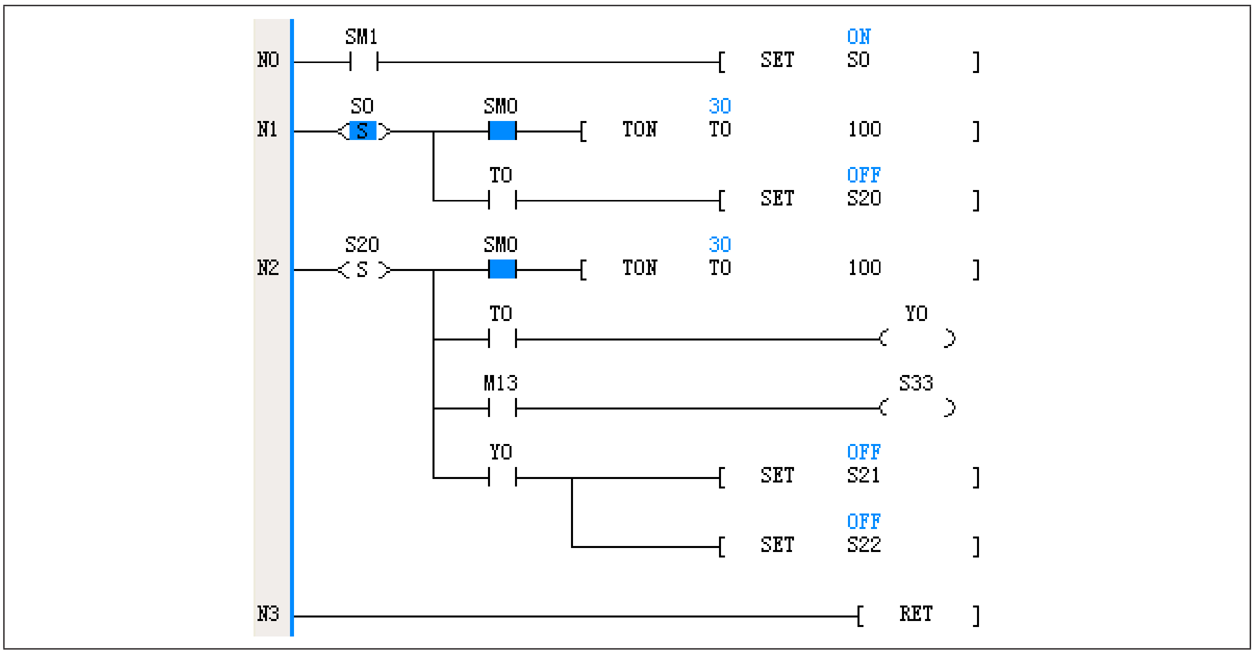
在采用梯形图输入时必须注意这个要求，否则编译不通过。

2．在相邻的步进状态重复使用软元件

PLC 执行程序时是按指令顺序循环扫描的。当从上一步进状态转移到下一步进状态时，上一步进状态内的指令序列刚扫描结束，下一步进状态指令序列也已经开通扫描，形成控制输出。

根据上述分析可知，STL 指令在由 ON 变为 OFF 时，虽然会复位其内部的一些元件（请参见 *6.2 SFC* 指令），但这个复位操作只有在下一次扫描周期时才执行。在步进状态转移的瞬间，上一步进状态内部元件仍然保持原有数据和状态直到下一次扫描经过该步进状态。

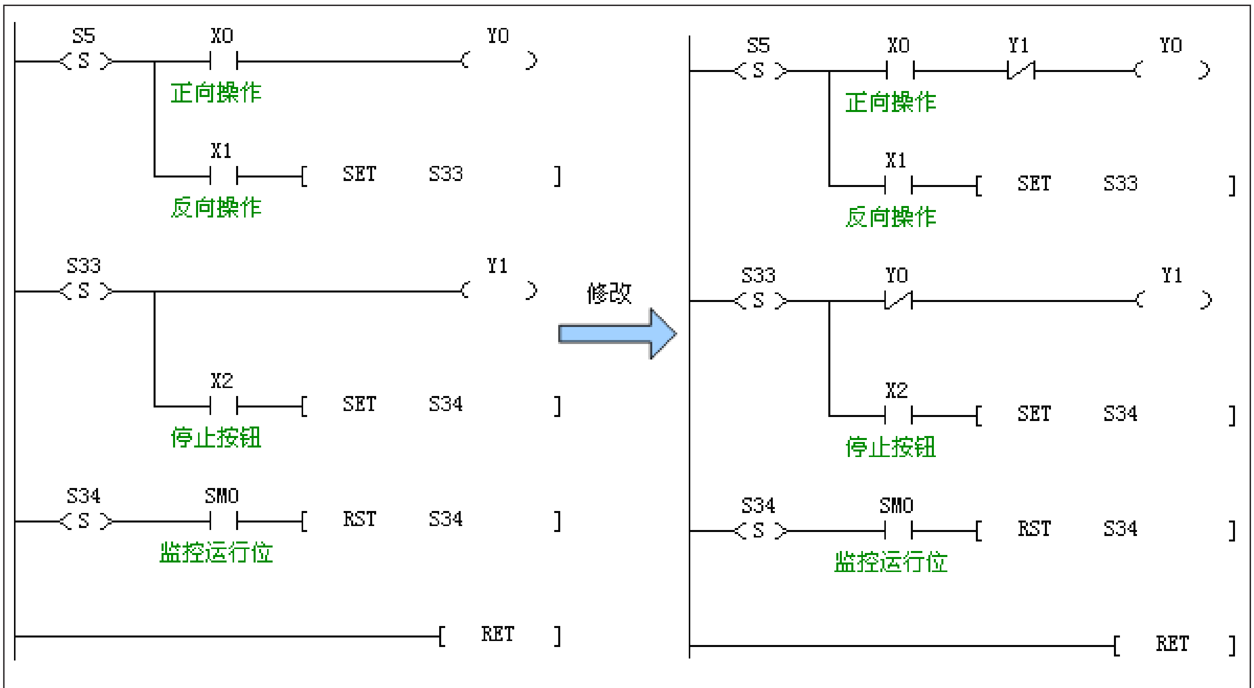
如下图所示，上下两个相联的步进状态中同时使用了 T0 计时器，当步进状态由 S0 转向 S20 时，T0 元件仍然会保持计数值和接通状态。S20 步进状态因此不能按照用户最初的设计来执行计时操作，而是直接进入下面的 S21 和 S22 步进状态。因此，在不同的步进状态中，编程软元件虽然可以重复使用，但是最好不要用在相邻的步进状态中，否则可能会造成意外的结果。



3．软元件未能互锁

在 SFC 编程中，有些软元件之间可能因为步进状态转移时的特殊情况而出现矛盾，此时需要对其进行互锁。

例如下图的正反顺序操作程序示例。Y0 和 Y1 分别是设备运行正向和反向控制输出。X0 为正向操作，X1 为反向操作，X2 为停止按钮。要求 Y0 和 Y1 互锁，即不能同时为 ON。然而在如下左图例程中，当设备正向运行时，接通 X1 使 S5 步进状态转移到 S33 步进状态的时刻，Y0 和 Y1 同时为 ON 且时间为一个程序扫描周期。



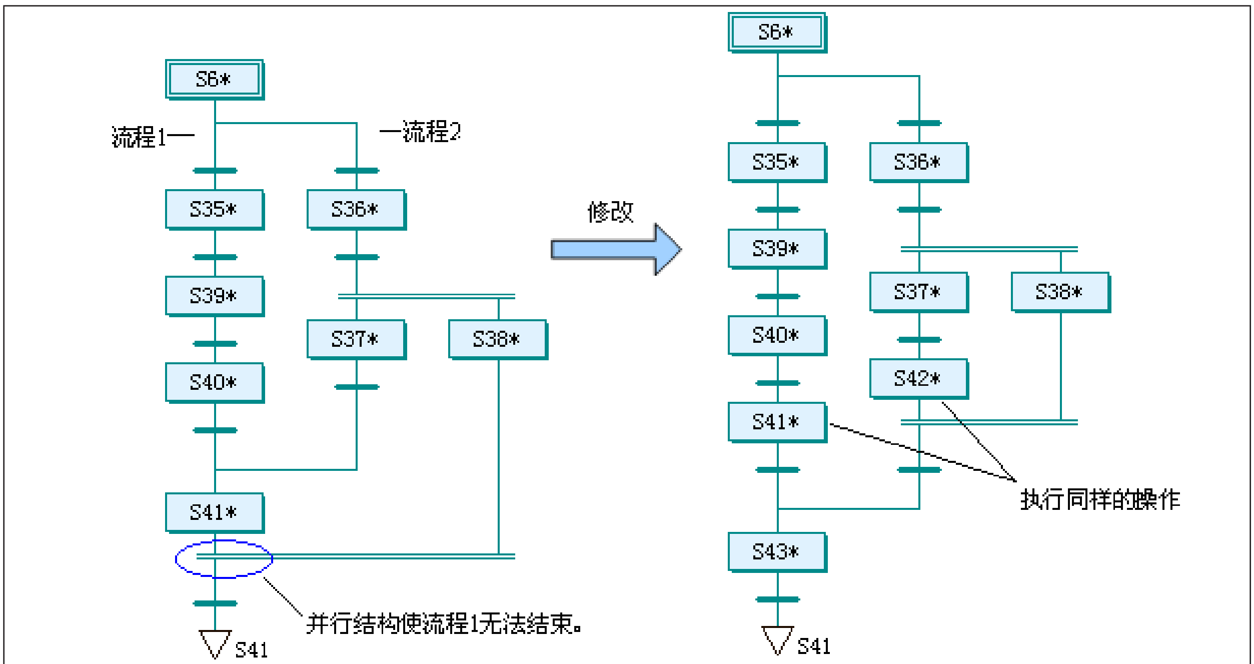
因此，程序中应该增加互锁的语句。如上图示例，在右图程序中 Y1 输出线圈前增加 Y0 的常闭触点，Y0 输出线圈前增加 Y1 的常闭触点以作为互锁，除此之外还必须在硬件上加上互锁。

4．跳转与转移

跳转是应用在不同流程、不相邻步进状态之间的切换。转移则是应用在相邻步进状态之间切换的操作。

5．选择分支转移处为并行分支汇合结构导致无法结束流程

选择分支是多选一的流程，如果其中混合了并行分支，则可能会导致选择分支运行无法结束。如下图所示，左图程序中，流程 1 在执行到步进状态 S41 时，由于转移条件为并行分支，而此刻系统不会再运行流程 2，导致该处转移条件永远无法实现，从而出现流程错误。



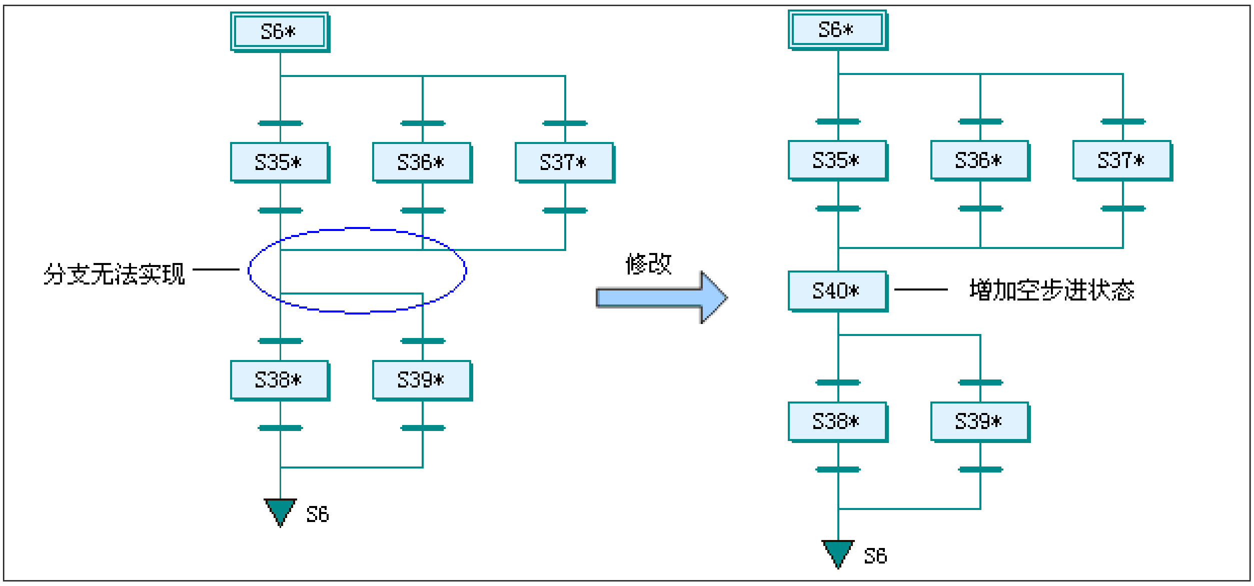
修改方法如右图所示，增加步进状态 S42，功能与 S41 完全一致；增加 S43 空步进状态，仅作为编程结构要素，没有实质操作。S38、S41、S43 的转移条件需要编程者设计，比如都采用原 S41 的转移条件即可。

7.4.2 编程技巧

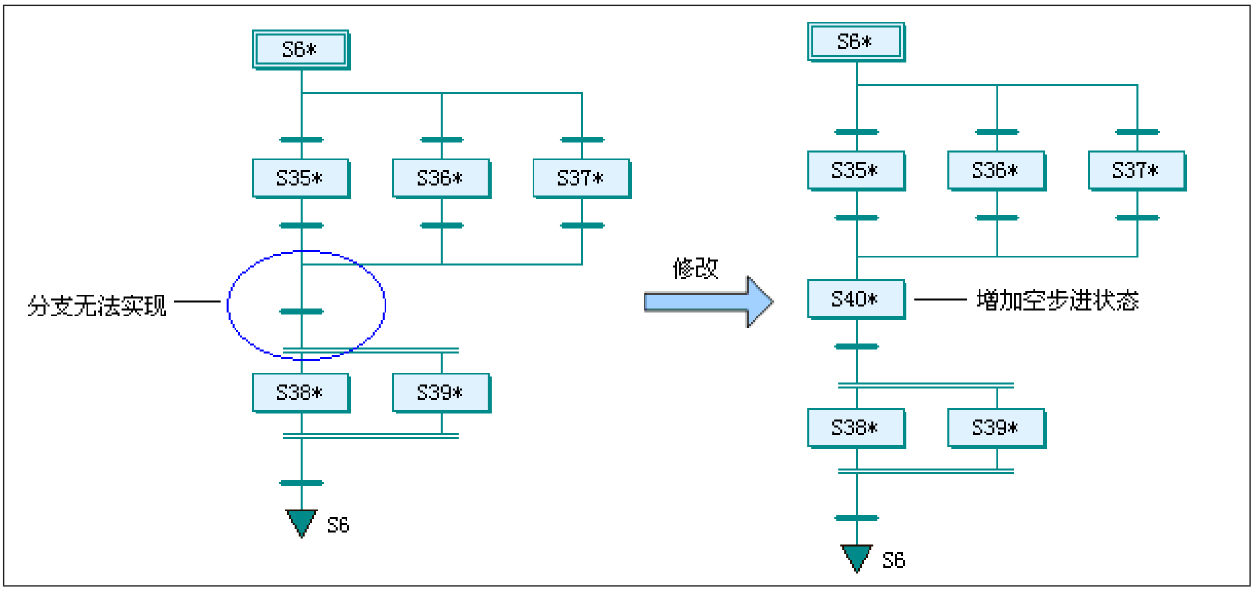
1．巧用空步进状态

一些有语法问题的分支设计，需要采用空步进状态来解决分支难题。所谓的空步进状态，就是在步进状态中没有安排有实质内容的操作，直接等待进行转移。如下是一些示例。

下面左图中，选择分支汇合后接另一个选择分支，这是不能编译通过的。可按照右图来修改，增加一个空步进状态。



下面左图中，选择分支后立即接上另一个并行也是不行的。可按照右图来修改，增加一个空步进状态即可。

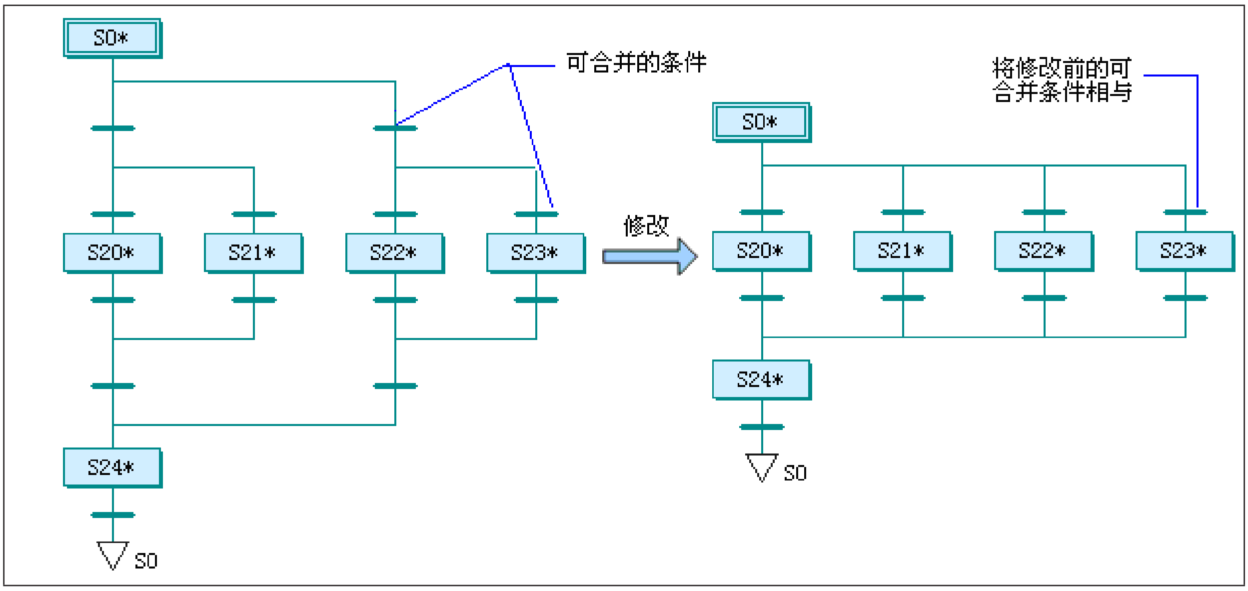


其它问题如并行汇合之后接并行分支、并行分支之后接选择分支，也可采用增加空步进状态的方式解决问题。

2．合并分支和转移条件

一些看起来复杂的分支，实际上是设计时分析不当造成的，可以适当合并或简化。

如下图，设计者先做了第一个选择分支，然后分别再做两个选择分支。实际上只需要采用一个四条支路的选择分支即可，原设计的上下两级转移符则可合并为转移条件相与的一级转移符。



3. 利用停电保持功能

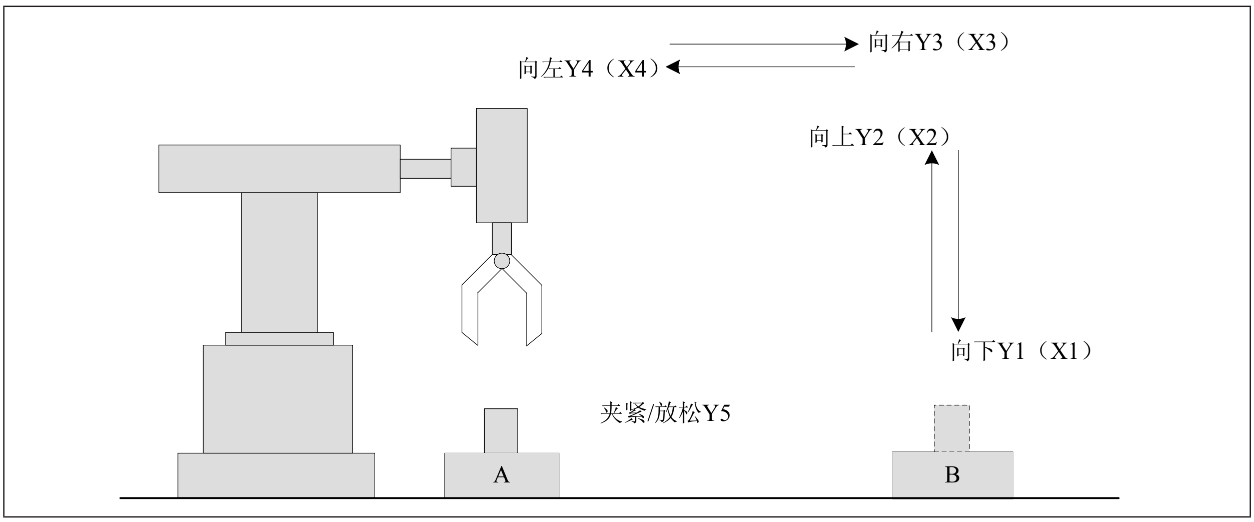
可用停电保持设置来保持 S 元件的值，在恢复上电后还可以从停电时的步进状态重新运行。

7.5 顺序功能图编程实例

本节中示例只能作为简明的顺序功能图编程，操作和条件是经过简化的。设备配置及设计表达的是粗略概念，不能当作对实际设备的设计，仅供学习参考。

7.5.1 简单结构

以某机械手的动作为例，介绍 SFC 的简单结构流程，下图为机械手动作过程。

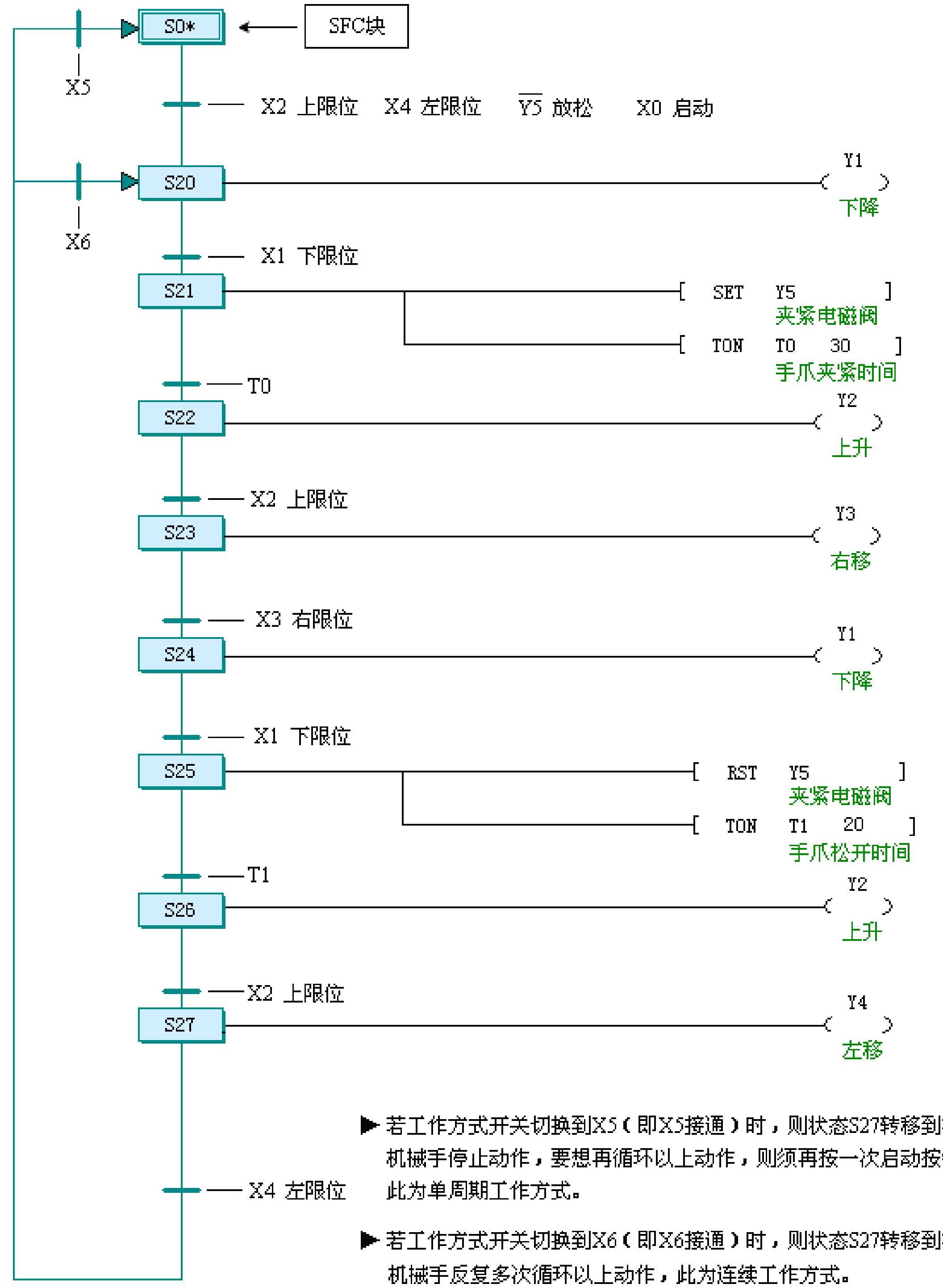


- **控制要求**

1. 当机械手处于原点时（即左限位和上限位开关均合上），按下启动按钮 X0 后，机械手下降到 A 点，夹紧工件后上升到原位，然后向右移到右限位开关 X3，再下降到 B 点后放下工件，再回到原位即完成一次操作。机械手就这样反复地将工件从 A 点移到 B 点。
2. 机械手可以实现单周期工作和连续工作两种工作方式。

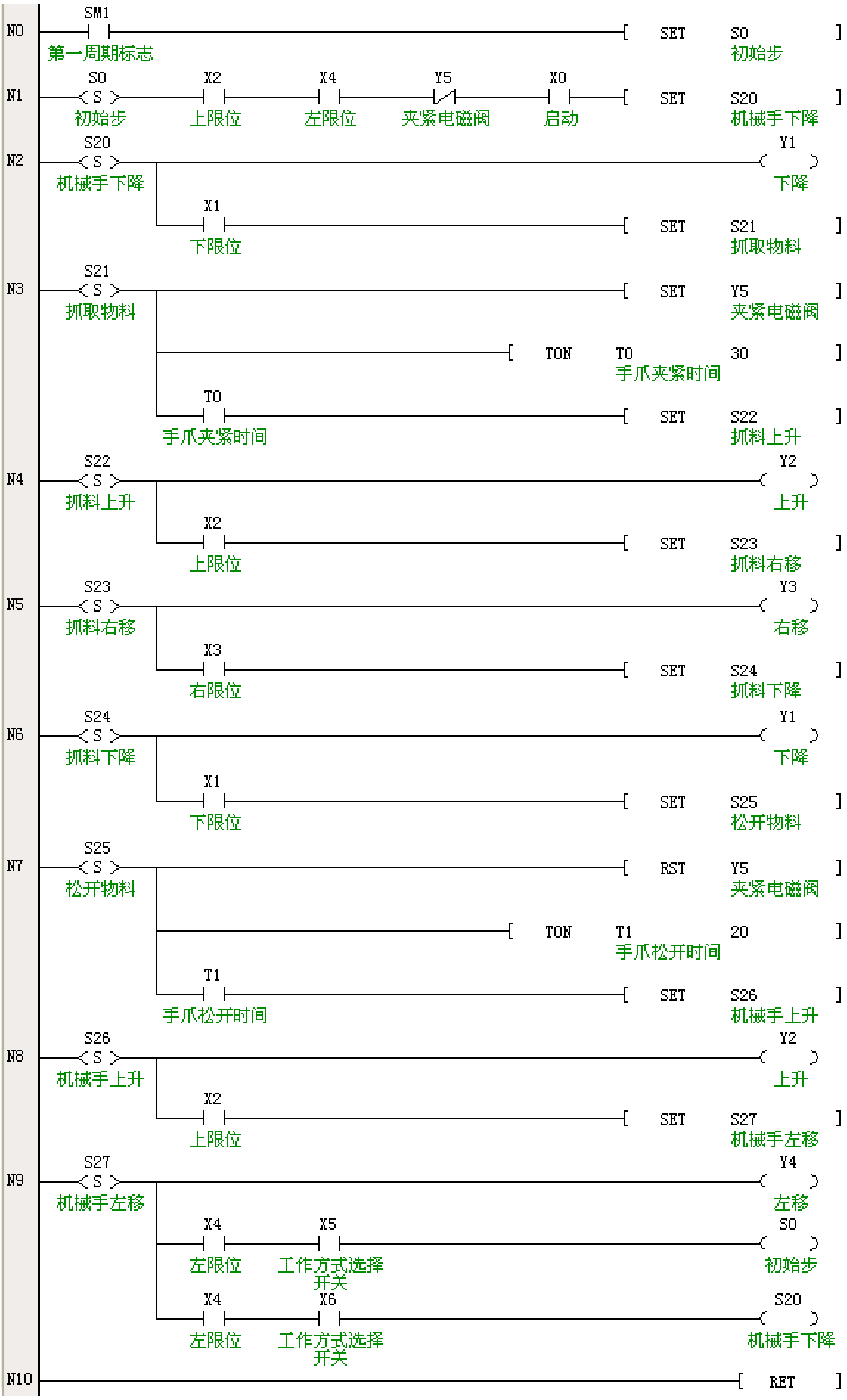
- **参考程序**

如下为顺序功能图程序以及对照的梯形图程序。



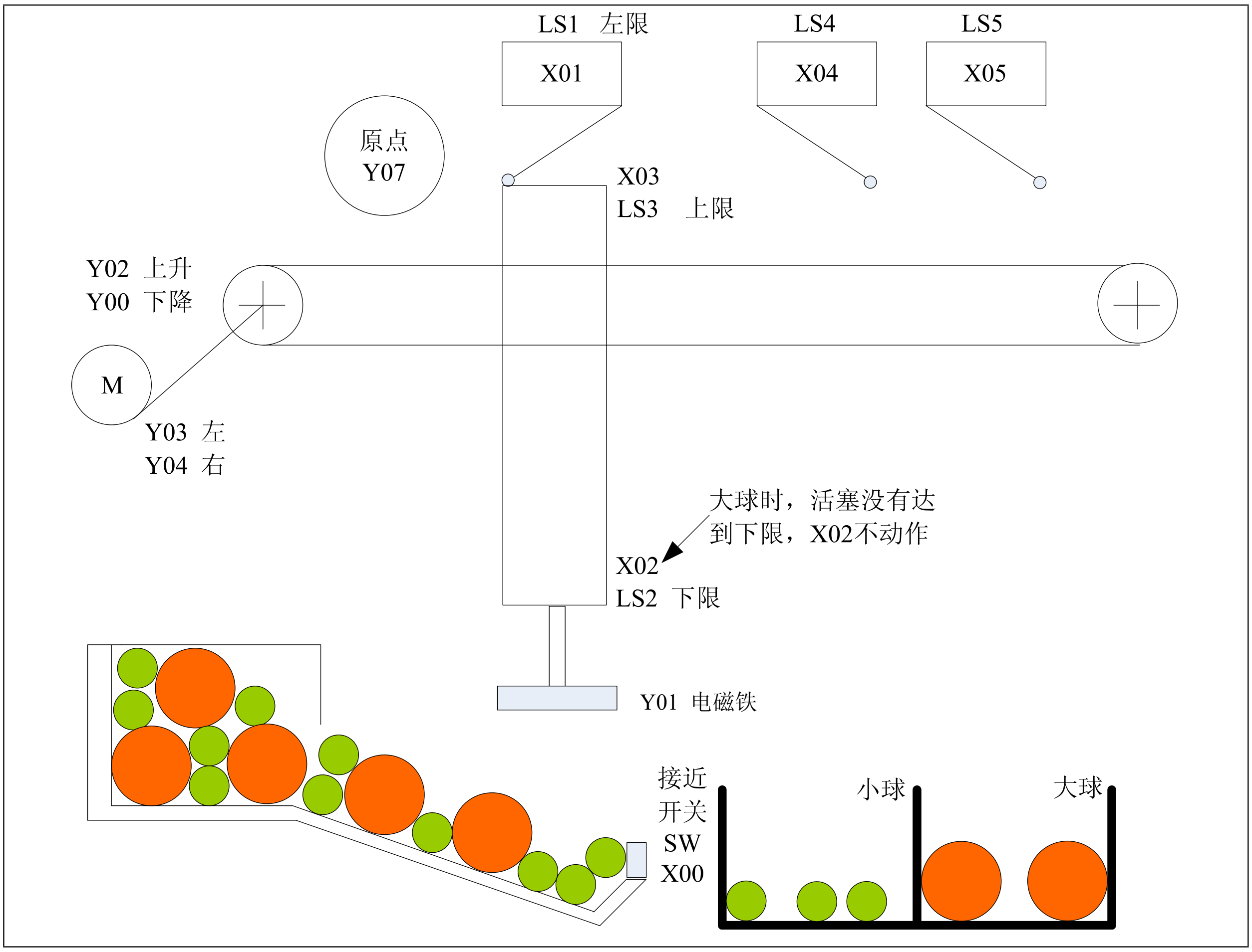
▶ 若工作方式开关切换到X5（即X5接通）时，则状态S27转移到状态S0，机械手停止动作，要想再循环以上动作，则须再按一次启动按钮X0，此为单周期工作方式。

▶ 若工作方式开关切换到X6（即X6接通）时，则状态S27转移到状态S20，机械手反复多次循环以上动作，此为连续工作方式。


N0
SM1
第一周期标志
SET S0
初始步
N1
S0
初始步
X2
上限位
X4
左限位
Y5
夹紧电磁阀
X0
启动
SET S20
机械手下降
N2
S20
机械手下降
Y1
下降
X1
下限位
SET S21
抓取物料
N3
S21
抓取物料
SET Y5
夹紧电磁阀
TON T0 30
手爪夹紧时间
T0
手爪夹紧时间
SET S22
抓料上升
N4
S22
抓料上升
Y2
上升
X2
上限位
SET S23
抓料右移
N5
S23
抓料右移
Y3
右移
X3
右限位
SET S24
抓料下降
N6
S24
抓料下降
Y1
下降
X1
下限位
SET S25
松开物料
N7
S25
松开物料
RST Y5
夹紧电磁阀
TON T1 20
手爪松开时间
T1
手爪松开时间
SET S26
机械手上升
N8
S26
机械手上升
Y2
上升
X2
上限位
SET S27
机械手左移
N9
S27
机械手左移
Y4
左移
X4
左限位
X5
工作方式选择开关
S0
初始步
X4
左限位
X6
工作方式选择开关
S20
机械手下降
N10
RET

7.5.2 选择结构

以下为大小球的选择传送例子，应用选择分支的步进流程来设计。下图为使用传送带，将大、小球分类选择传送的机械装置。

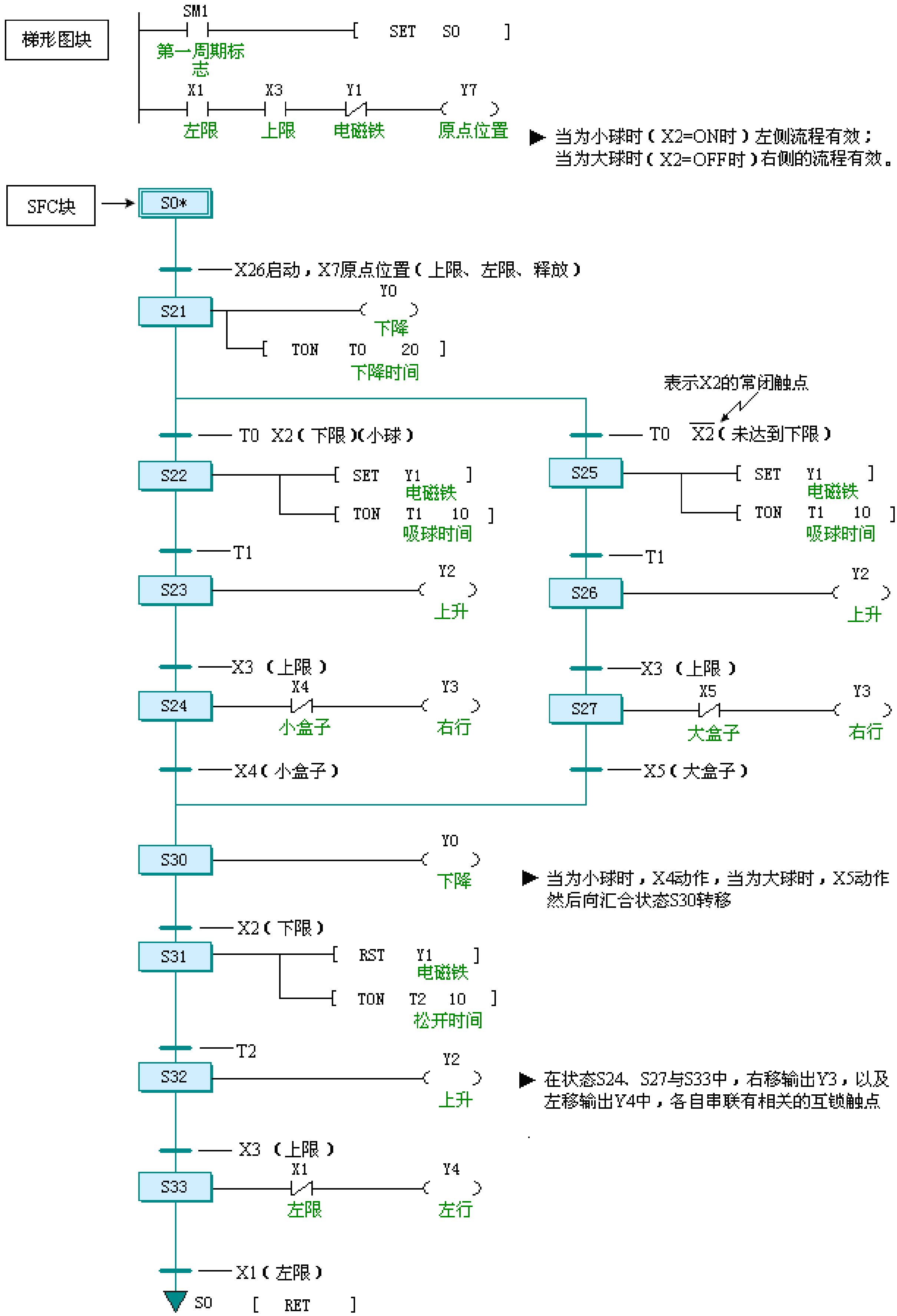


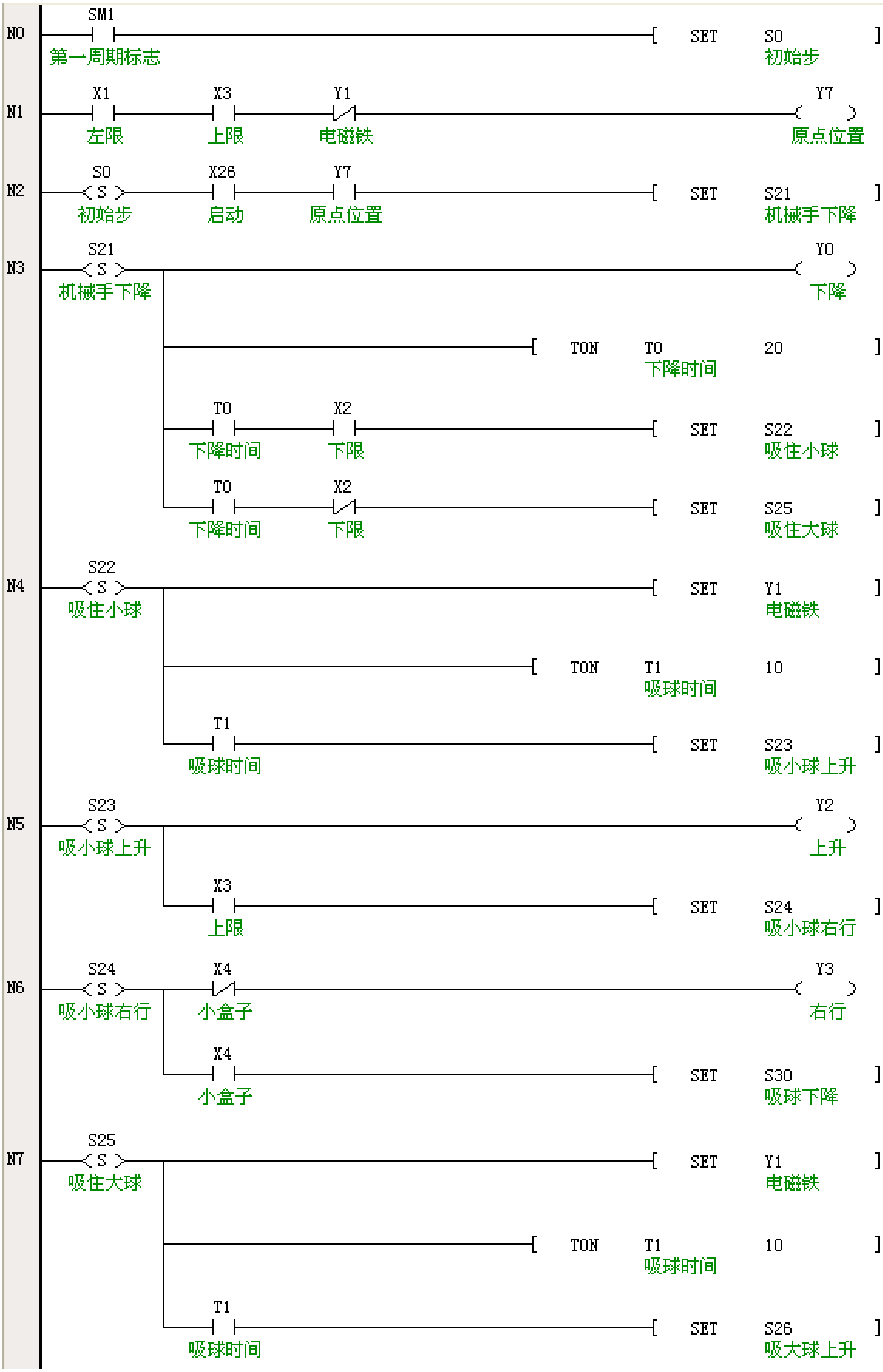
- **控制要求**

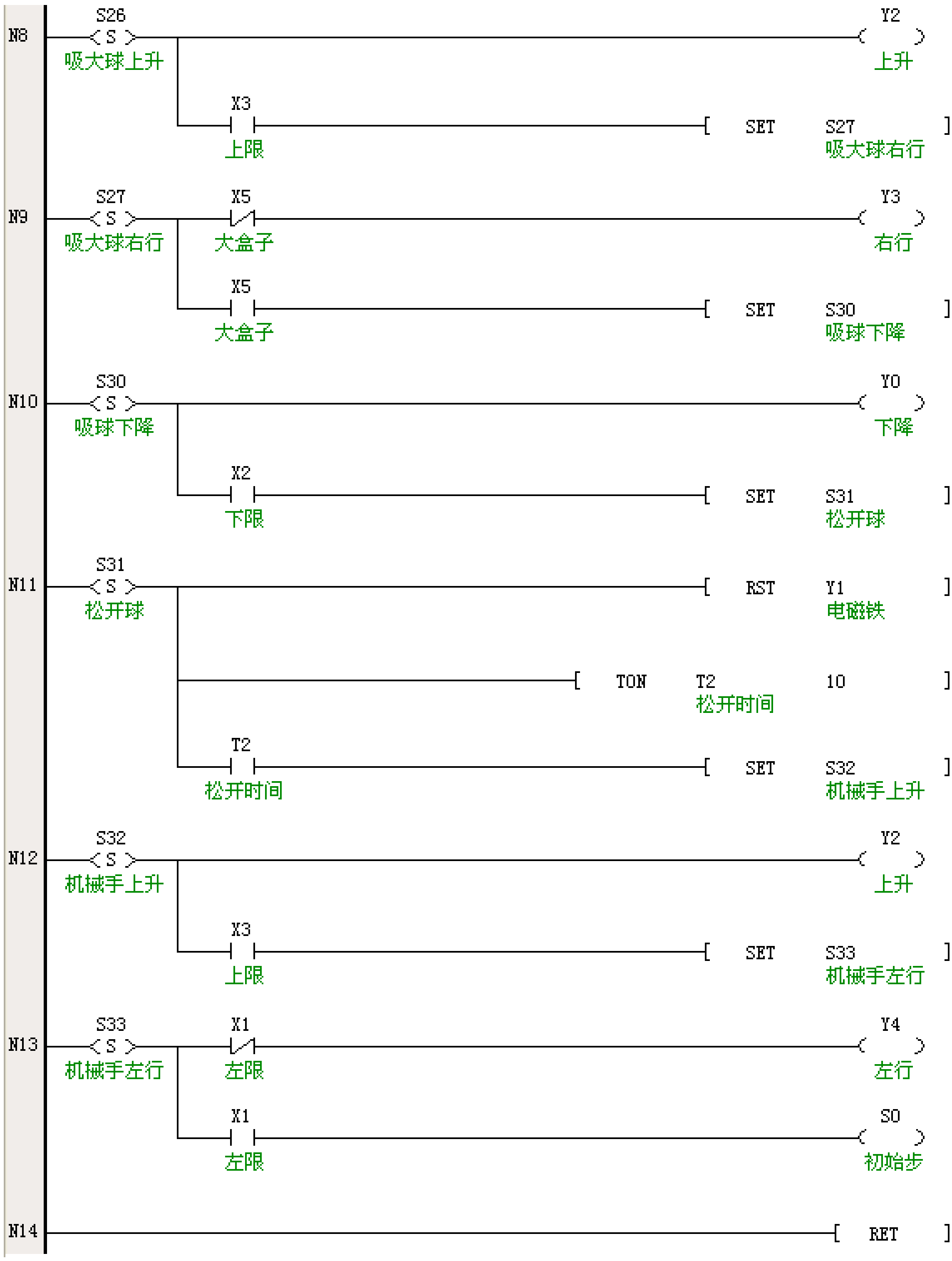
1. 左上方为原点，其动作顺序为下降、吸住、上升、右行、下降、释放、上升、左行。
2. 机械手臂下降，当电磁铁压着大球时，下限限位开关 LS2 断开；压着小球时， LS2 导通。

- **参考程序**

如下为顺序功能图程序以及对照的梯形图程序。

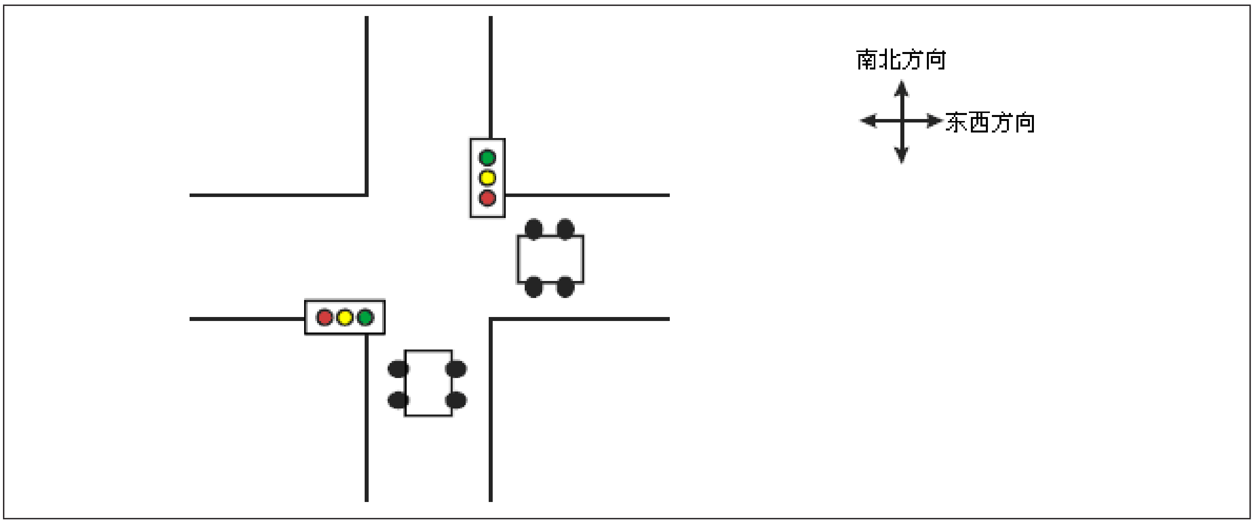

梯形图块
SM1
[SET S0]
第一周期标志
X1 X3 Y1 Y7
左限 上限 电磁铁 原点位置
▶ 当为小球时（X2=ON时）左侧流程有效；
当为大球时（X2=OFF时）右侧的流程有效。
SFC块
S0*
X26启动，X7原点位置（上限、左限、释放）
S21
Y0
下降
[TON T0 20]
下降时间
表示X2的常闭触点
T0 X2（下限）(小球）
T0 X2（未达到下限）
S22
[SET Y1]
电磁铁
[TON T1 10]
吸球时间
S25
[SET Y1]
电磁铁
[TON T1 10]
吸球时间
T1
S23
Y2
上升
T1
S26
Y2
上升
X3 （上限）
S24
X4
小盒子
Y3
右行
X3 （上限）
S27
X5
大盒子
Y3
右行
X4（小盒子）
X5（大盒子）
S30
Y0
下降
▶ 当为小球时，X4动作，当为大球时，X5动作
然后向汇合状态S30转移
X2（下限）
S31
[RST Y1]
电磁铁
[TON T2 10]
松开时间
T2
S32
Y2
上升
▶ 在状态S24、S27与S33中，右移输出Y3，以及
左移输出Y4中，各自串联有相关的互锁触点
X3 （上限）
S33
X1
左限
Y4
左行
X1（左限）
S0 [RET]


N0
SM1
第一周期标志
SET S0
初始步
N1
X1
左限
X3
上限
Y1
电磁铁
Y7
原点位置
N2
S0
S
初始步
X26
启动
Y7
原点位置
SET S21
机械手下降
N3
S21
S
机械手下降
Y0
下降
TON T0 20
下降时间
T0
下降时间
X2
下限
SET S22
吸住小球
T0
下降时间
X2
下限
SET S25
吸住大球
N4
S22
S
吸住小球
SET Y1
电磁铁
TON T1 10
吸球时间
T1
吸球时间
SET S23
吸小球上升
N5
S23
S
吸小球上升
Y2
上升
X3
上限
SET S24
吸小球右行
N6
S24
S
吸小球右行
X4
小盒子
Y3
右行
X4
小盒子
SET S30
吸球下降
N7
S25
S
吸住大球
SET Y1
电磁铁
TON T1 10
吸球时间
T1
吸球时间
SET S26
吸大球上升


N8
S26
S
吸大球上升
Y2
上升
X3
上限
SET
S27
吸大球右行
N9
S27
S
吸大球右行
X5
大盒子
Y3
右行
X5
大盒子
SET
S30
吸球下降
N10
S30
S
吸球下降
Y0
下降
X2
下限
SET
S31
松开球
N11
S31
S
松开球
RST
Y1
电磁铁
TON
T2
松开时间
10
T2
松开时间
SET
S32
机械手上升
N12
S32
S
机械手上升
Y2
上升
X3
上限
SET
S33
机械手左行
N13
S33
S
机械手左行
X1
左限
Y4
左行
X1
左限
S0
初始步
N14
RET

7.5.3 并行结构

以下是交通灯控制的示例，应用并行分支的步进流程来设计，分为东西和南北方向两个流程，两个流程同时进行。下图为交通灯控制图：

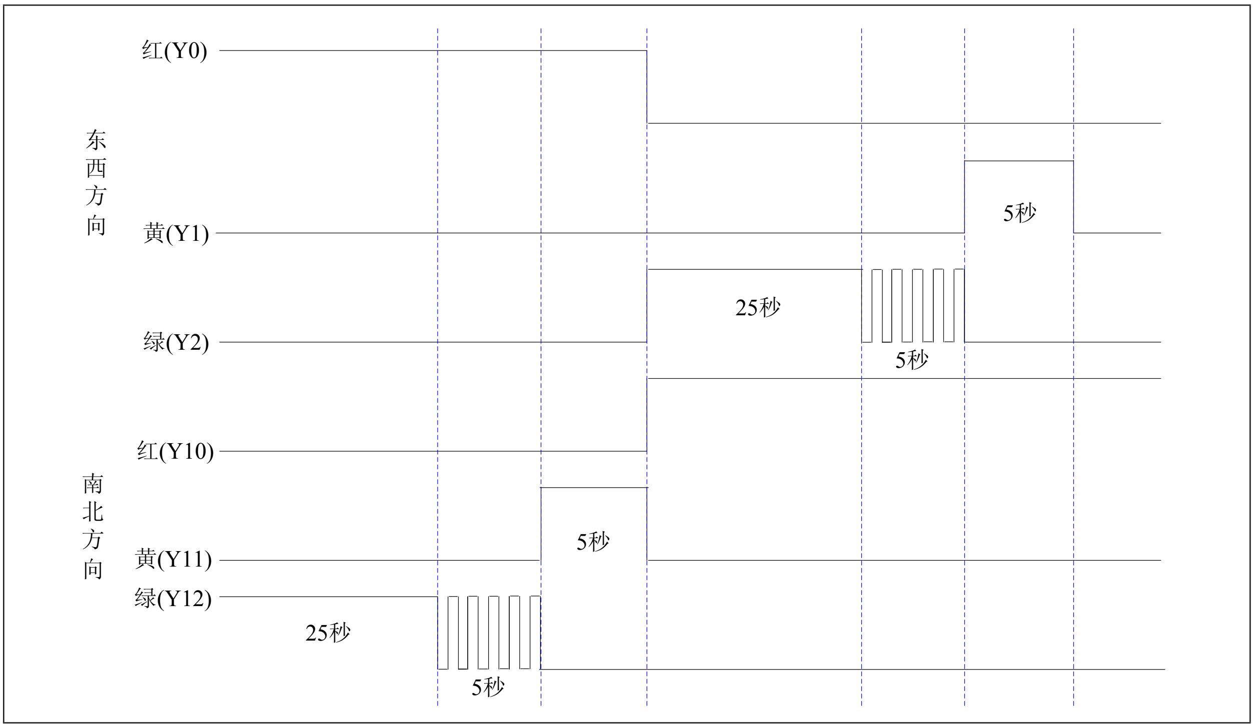


- **控制要求**

1．按下启动按钮 X0，交通灯系统开始工作；按下停止按钮 X1，交通灯系统停止运行。
2．东西方向的红灯时间就是南北方向的“绿灯时间＋绿灯闪烁时间＋黄灯时间”，反之，南北方向红灯时间就是东西向的“绿灯时间＋绿灯闪烁时间＋黄灯时间”。
3．黄灯亮时车和人不能再通过马路，黄灯亮 5 秒的目的是让正在十字路口通行的人和车有时间到达对面马路。

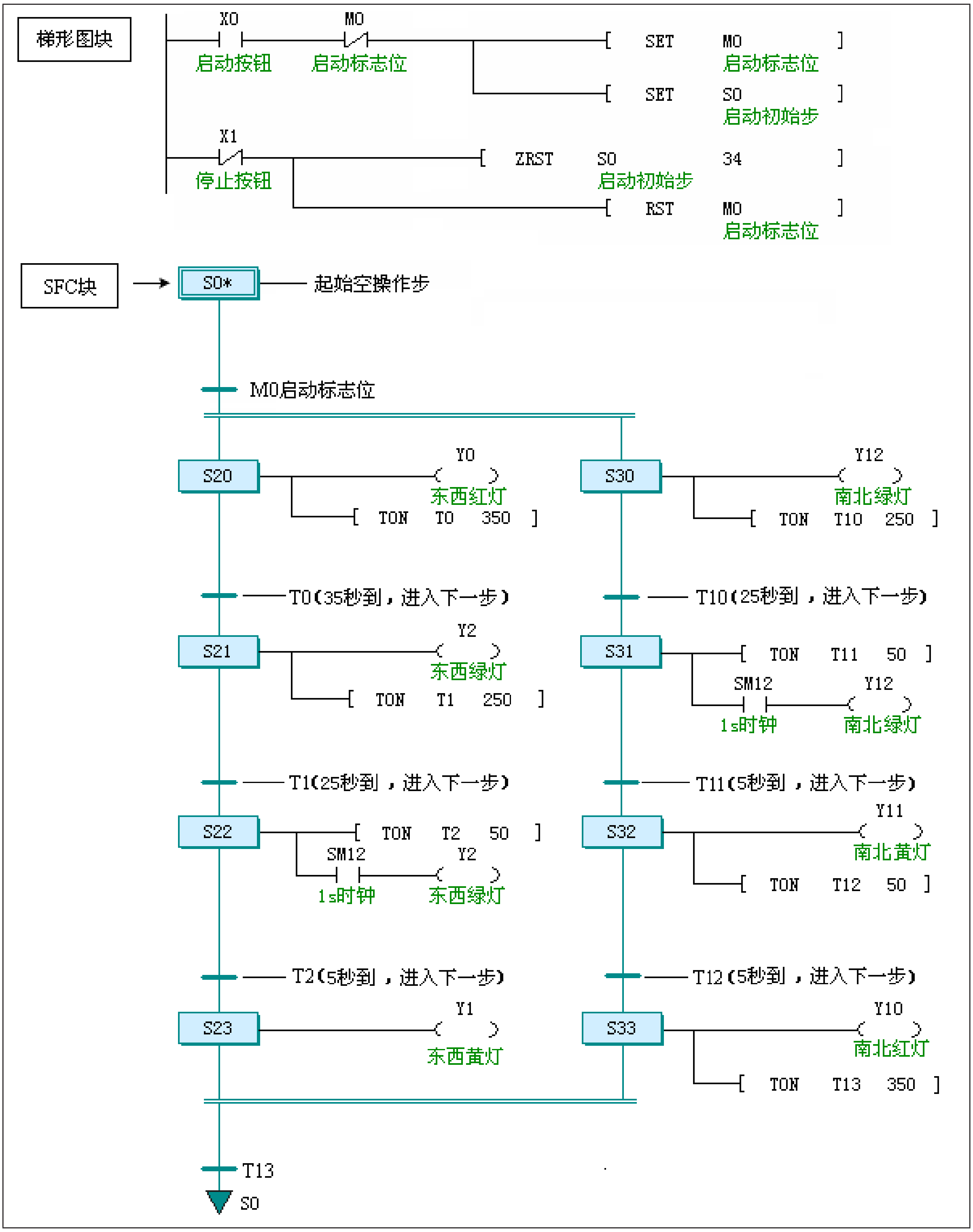
灯 / 标志	红灯	黄灯	绿灯	绿灯闪烁
东西方向信号标志	Y0	Y1	Y2	Y2
南北方向信号标志	Y10	Y11	Y12	Y12
灯亮时间	35 秒	5 秒	25 秒	5 秒

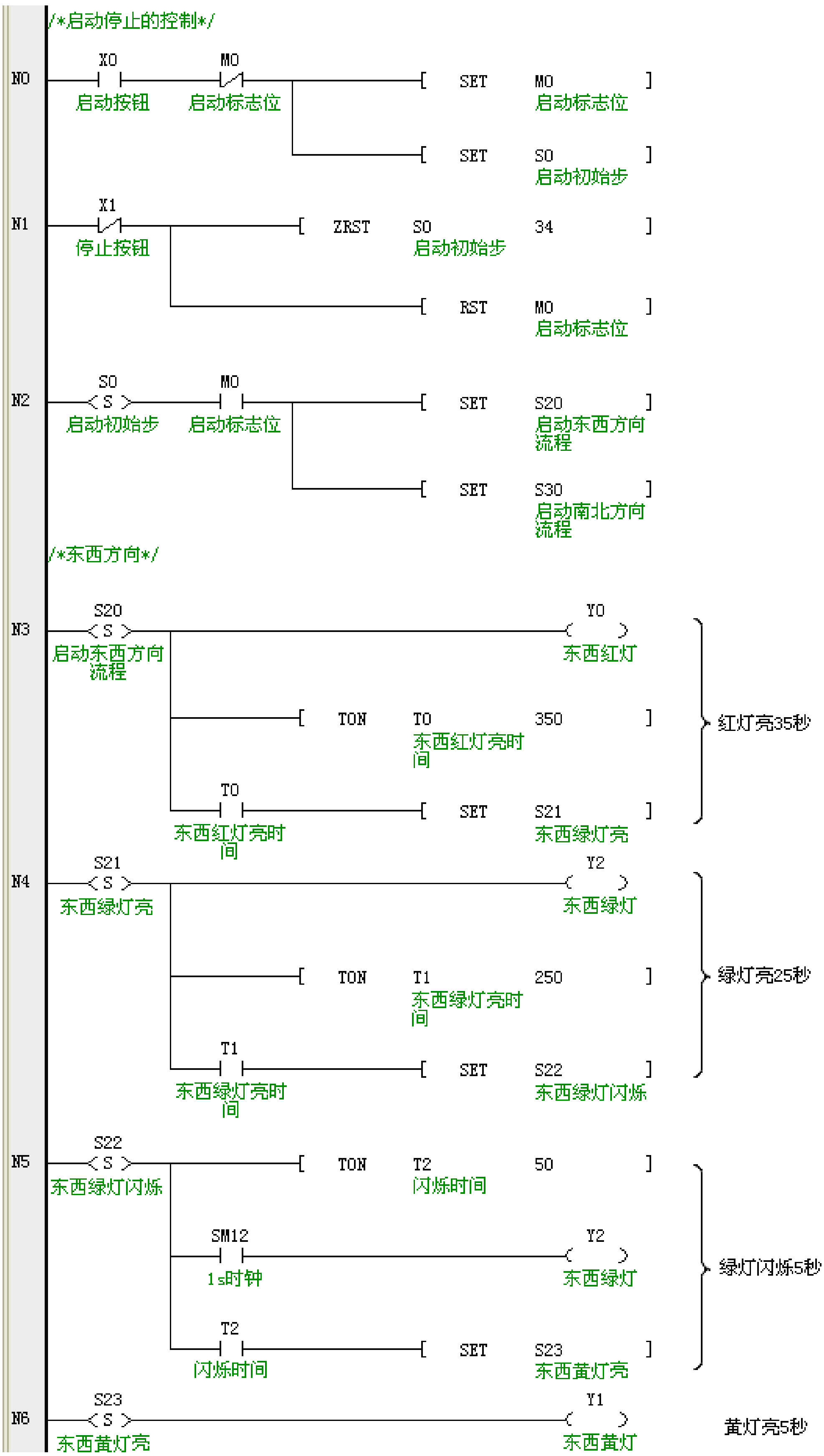
交通灯状态变化时序图：

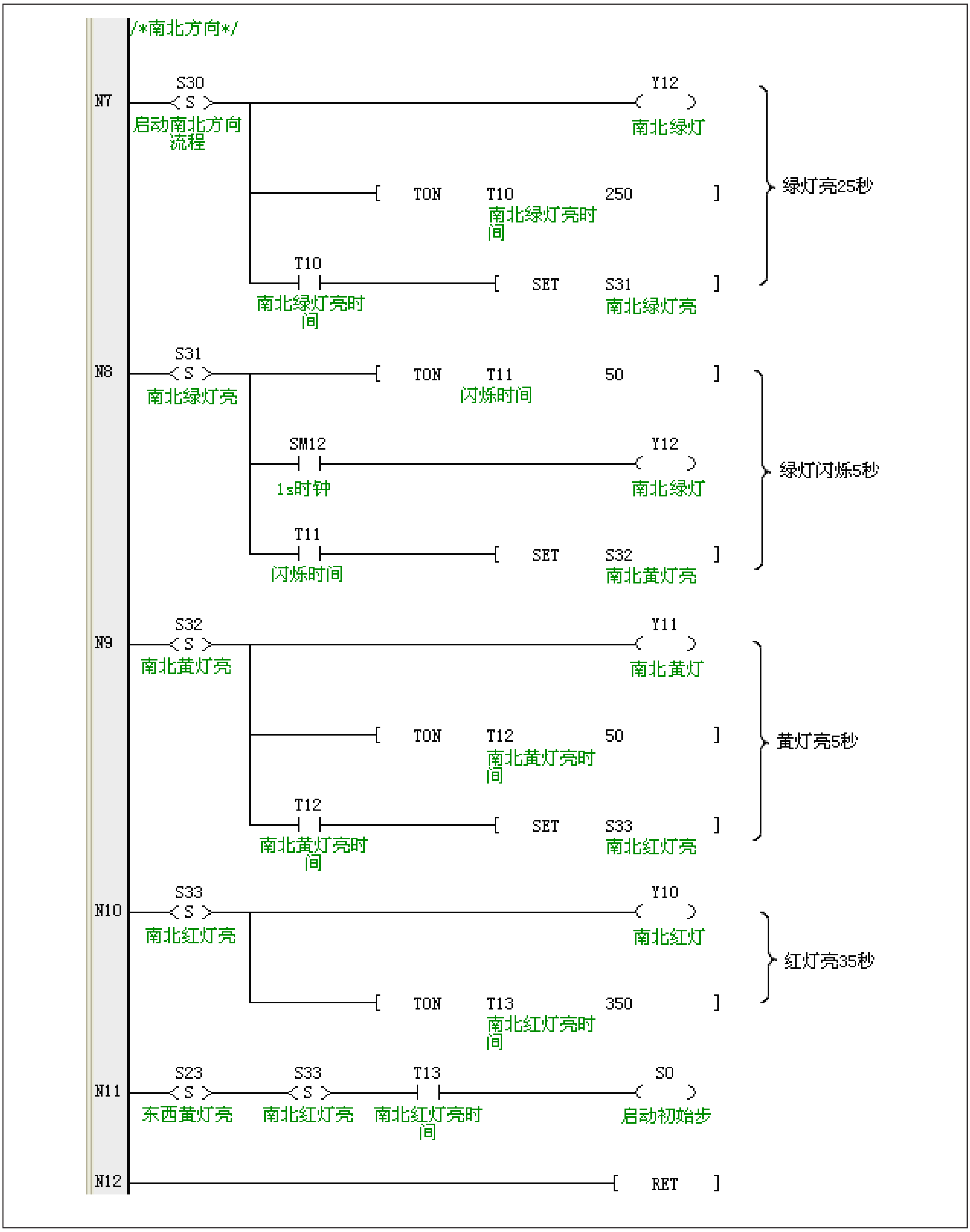


- 参考程序

如下为顺序功能图程序，以及对照的梯形图程序。




/*启动停止的控制*/
N0
X0
启动按钮
M0
启动标志位
SET M0 启动标志位
SET S0 启动初始步
N1
X1
停止按钮
ZRST S0 启动初始步 34
RST M0 启动标志位
N2
S0
启动初始步
M0
启动标志位
SET S20 启动东西方向流程
SET S30 启动南北方向流程
/*东西方向*/
N3
S20
启动东西方向流程
Y0
东西红灯
TON T0 东西红灯亮时间 350
T0
东西红灯亮时间
SET S21 东西绿灯亮
红灯亮35秒
N4
S21
东西绿灯亮
Y2
东西绿灯
TON T1 东西绿灯亮时间 250
T1
东西绿灯亮时间
SET S22 东西绿灯闪烁
绿灯亮25秒
N5
S22
东西绿灯闪烁
TON T2 闪烁时间 50
SM12
1s时钟
Y2
东西绿灯
T2
闪烁时间
SET S23 东西黄灯亮
绿灯闪烁5秒
N6
S23
东西黄灯亮
Y1
东西黄灯
黄灯亮5秒


/*南北方向*/
N7
S30
启动南北方向流程
Y12
南北绿灯
TON T10 250
南北绿灯亮时间
T10
南北绿灯亮时间
SET S31
南北绿灯亮
绿灯亮25秒
N8
S31
南北绿灯亮
TON T11 50
闪烁时间
SM12
1s时钟
Y12
南北绿灯
T11
闪烁时间
SET S32
南北黄灯亮
绿灯闪烁5秒
N9
S32
南北黄灯亮
Y11
南北黄灯
TON T12 50
南北黄灯亮时间
T12
南北黄灯亮时间
SET S33
南北红灯亮
黄灯亮5秒
N10
S33
南北红灯亮
Y10
南北红灯
TON T13 350
南北红灯亮时间
红灯亮35秒
N11
S23
东西黄灯亮
S33
南北红灯亮
T13
南北红灯亮时间
S0
启动初始步
N12
RET

第八章　高速 I/O 功能使用指南

本章详细讲述了高速计数器、外部脉冲捕捉、PLS 包络线的使用方法、高速脉冲输入 / 输出与注意事项。

8.1 高速计数器

8.1.1 高速计数器配置

MX1H/2H 高速计数器配置：

<table>
<tr><th colspan="2" rowspan="2">输入点
计数器</th><th rowspan="2">X0</th><th rowspan="2">X1</th><th rowspan="2">X2</th><th rowspan="2">X3</th><th rowspan="2">X4</th><th rowspan="2">X5</th><th rowspan="2">X6</th><th rowspan="2">X7</th><th colspan="2">最高频率 KHz</th></tr>
<tr><th>1H</th><th>2H</th></tr>
<tr><td rowspan="11">单相单端计数输入方式</td><td>C235</td><td>增 / 减</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td rowspan="2">50</td><td rowspan="2">50</td></tr>
<tr><td>C236</td><td></td><td>增 / 减</td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>C237</td><td></td><td></td><td>增 / 减</td><td></td><td></td><td></td><td></td><td></td><td rowspan="4">10</td><td rowspan="4">10</td></tr>
<tr><td>C238</td><td></td><td></td><td></td><td>增 / 减</td><td></td><td></td><td></td><td></td></tr>
<tr><td>C239</td><td></td><td></td><td></td><td></td><td>增 / 减</td><td></td><td></td><td></td></tr>
<tr><td>C240</td><td></td><td></td><td></td><td></td><td></td><td>增 / 减</td><td></td><td></td></tr>
<tr><td>C241</td><td>增 / 减</td><td>复位</td><td></td><td></td><td></td><td></td><td></td><td></td><td>50</td><td>50</td></tr>
<tr><td>C242</td><td></td><td></td><td>增 / 减</td><td>复位</td><td></td><td></td><td></td><td></td><td rowspan="2">10</td><td rowspan="2">10</td></tr>
<tr><td>C243</td><td></td><td></td><td></td><td></td><td>增 / 减</td><td>复位</td><td></td><td></td></tr>
<tr><td>C244</td><td>增 / 减</td><td>复位</td><td></td><td></td><td></td><td></td><td>启动</td><td></td><td>50</td><td>50</td></tr>
<tr><td>C245</td><td></td><td></td><td>增 / 减</td><td>复位</td><td></td><td></td><td></td><td>启动</td><td>10</td><td>10</td></tr>
<tr><td rowspan="6">单相增减计数输入方式</td><td>C246</td><td>增</td><td>减</td><td></td><td></td><td></td><td></td><td></td><td></td><td rowspan="2">50</td><td rowspan="2">50</td></tr>
<tr><td>C247</td><td>增</td><td>减</td><td>复位</td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>C248</td><td></td><td></td><td></td><td>增</td><td>减</td><td>复位</td><td></td><td></td><td>10</td><td>10</td></tr>
<tr><td>C248OP</td><td></td><td></td><td></td><td>增</td><td>减</td><td></td><td></td><td></td><td>10</td><td>10</td></tr>
<tr><td>C249</td><td>增</td><td>减</td><td>复位</td><td></td><td></td><td></td><td>启动</td><td></td><td>50</td><td>50</td></tr>
<tr><td>C250</td><td></td><td></td><td></td><td>增</td><td>减</td><td>复位</td><td></td><td>启动</td><td>10</td><td>10</td></tr>
<tr><td rowspan="7">双相计数输入方式</td><td>C251</td><td>A 相</td><td>B 相</td><td></td><td></td><td></td><td></td><td></td><td></td><td rowspan="2">20</td><td rowspan="2">25</td></tr>
<tr><td>C252</td><td>A 相</td><td>B 相</td><td>复位</td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>C253</td><td></td><td></td><td></td><td>A 相</td><td>B 相</td><td>复位</td><td></td><td></td><td>10</td><td>5</td></tr>
<tr><td>C253OP</td><td></td><td></td><td></td><td>A 相</td><td>B 相</td><td></td><td></td><td></td><td>10</td><td>5</td></tr>
<tr><td>C254</td><td>A 相</td><td>B 相</td><td>复位</td><td></td><td></td><td></td><td>启动</td><td></td><td>20</td><td>25</td></tr>
<tr><td>C254OP</td><td></td><td></td><td></td><td></td><td></td><td></td><td>A 相</td><td>B 相</td><td>10</td><td>5</td></tr>
<tr><td>C255</td><td></td><td></td><td></td><td>A 相</td><td>B 相</td><td>复位</td><td></td><td>启动</td><td>10</td><td>5</td></tr>
</table>

需要注意的是，如果多个双相计数器混合使用时，C251/C252/C254 最高频率为 15K。

高速计数器按上表所示的方式，根据特定的输入执行动作，计数的动作与 PLC 的扫描周期无关。

不作高速计数器使用的输入端子，可以作一般输入使用。

在对 PLC 编程时，每一个输入点所对应的高速计数器中只能采用其中 1 个，否则，用户程序将不能通过编译。

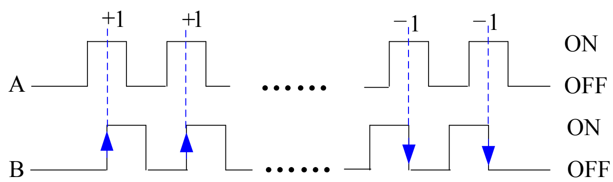
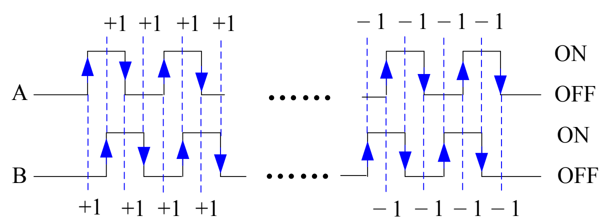
【表格的阅读举例 1】

表中 C235 为单相单端输入计数，使用 X0 输入端口，不需要中断复位与中断启动端口。如果使用 C235 计数器，即默认使用了 X0 输入端口，便不可再使用 C241，C244，C246，C247，C249，C251，C252，C254 和中断 X0 端口或者 SM90（X0 脉冲捕捉监视位），因为这些计数器、中断端口、脉冲捕捉也需用到 X0 端口，会形成端口冲突。

【表格的阅读举例 2】

表中 C255 为双相双输入计数器，即 AB 相计数器，X3 端口作为 A 相输入，X4 端口作为 B 相输入，X5 端口作为中断复位输入，X7 端口作为中断启动输入。如果使用 C255 计数器，即默认使用了 X3、X4、X5、X7 输入端口，与这些端口相关的计数器、中断端口或者脉冲捕捉，便都不能再使用了。

这类计数器是 32 位的增 / 减型的计数器，根据不同的增计数 / 减计数的切换方法，可划分为以下四种类型：

<table>
<tr><th>计数方式</th><th>计数动作</th></tr>
<tr><td>单相单端计数</td><td>SM235 ～ SM245 的 ON/OFF 状态分别对应 C235 ～ C245 增 / 减计数。</td></tr>
<tr><td>单相增减计数</td><td>对应于增计数输入或减计数输入的动作计数器 C246 ～ C250 自动的增减计数，通过 SM246 ～ SM250 可以知道对应计数器当前的计数方向，SM 元件为 OFF 时为增计数，为 ON 时为减计数。</td></tr>
<tr><td>双相一倍频计数</td><td>SM100、SM101 设置为 OFF 时，计数器 C251 ～ C255 根据双相输入作自动的普通增减计数，通过 SM251 ～ SM255 可知对应计数器当前的计数方向，SM 元件为 OFF 时为增计数，为 ON 时为减计数。计数方向定义如下：

</td></tr>
<tr><td>双相四倍频计数</td><td>SM100、SM101 设置为 ON 时，计数器 C251 ～ C255 根据双相输入作自动的四倍频增减计数，通过 SM251 ～ SM255 可知对应计数器当前的计数方向，SM 元件为 OFF 时为增计数，为 ON 时为减计数。计数方向定义如下：

</td></tr>
</table>

MX2V 高速计数器配置：

计数器 \ 输入点		X0	X1	X2	X3	X4	X5	X6	X7	最高频率 KHz
										2V
单相单端计数输入方式	C235	增 / 减								200
	C236		增 / 减							
	C237			增 / 减						
	C238				增 / 减					
	C239					增 / 减				
	C240						增 / 减			
	C244OP							增 / 减		
	C245OP								增 / 减	
	C241	增 / 减	复位							
	C242			增 / 减	复位					
	C243					增 / 减	复位			
	C250OP							增 / 减	复位	
	C244	增 / 减	复位					启动		
	C245			增 / 减	复位				启动	

计数器＼输入点		X0	X1	X2	X3	X4	X5	X6	X7	最高频率 KHz 2V
单相增减计数输入方式	C246	增	减							200
	C248OP			增	减					
	C247OP					增	减			
	C249OP							增	减	
	C247	增	减			复位				
	C248			增	减		复位			
	C249	增	减			复位		启动		
	C250			增	减		复位		启动	
双相计数输入方式	C251	A 相	B 相					CAP		200
	C253OP			A 相	B 相				CAP	
	C254OP					A 相	B 相			
	C255OP							A 相	B 相	
	C252	A 相	B 相			复位		CAP		
	C253			A 相	B 相		复位		CAP	
	C254	A 相	B 相			复位		启动		
	C255			A 相	B 相		复位		启动	

CAP：计数器在四倍频模式下脉冲捕获。需要先进行脉冲捕获设置，再开计数器，否则能流导通后再设置不起作用。

脉冲捕获相关 SD 元件：

元件地址	功能	元件地址	功能
SD412	C251/C252/C253/C253OP 捕获模式设置	SD424	C253/C253OP 上升沿捕获值低位
SD420	C251/C252 上升沿捕获值低位	SD425	C253/C253OP 上升沿捕获值高位
SD421	C251/C252 上升沿捕获值高位	SD426	C253/C253OP 下降沿捕获值低位
SD422	C251/C252 下降沿捕获值低位	SD427	C253/C253OP 下降沿捕获值高位
SD423	C251/C252 下降沿捕获值高位		

捕获模式设置（SD412 的位定义）：

位	值	功能	备注
Bit0	0	电平复位	C251/C252
	1	边沿复位	
Bit1	0	上升沿复位	
	1	下降沿复位	
Bit2	0	CAP 无效	
	1	CAP 有效	
Bit3	0	CAP 上升沿捕获	
	1	CAP 下降沿捕获	
Bit7 ～ Bit4	未定义		
Bit8	0	电平复位	C253/C253OP
	1	边沿复位	
Bit9	0	上升沿复位	
	1	下降沿复位	
Bit10	0	CAP 无效	
	1	CAP 有效	
Bit11	0	CAP 上升沿捕获	
	1	CAP 下降沿捕获	
Bit15 ～ Bit12	未定义		

8.1.2 高速计数器与 SM 辅助继电器的关系

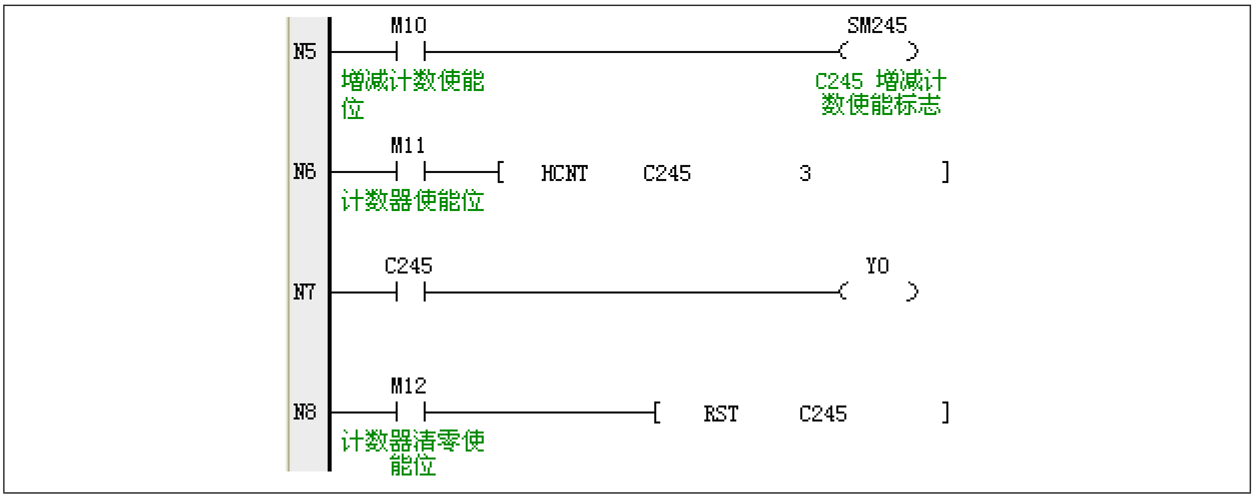
种类	计数器号	增 / 减设定	增 / 减监视位	四倍频设定
单相单端计数	C235	SM235		
	C236	SM236		
	C237	SM237		
	C238	SM238		
	C239	SM239		
	C240	SM240		
	C241	SM241		
	C242	SM242		
	C243	SM243		
	C244	SM244		
	C244OP		SM107	
	C245	SM245		
	C245OP		SM108	
	C250OP		SM107	
单相增减计数	C246		SM246	
	C247		SM247	
	C247OP		SM105	
	C248		SM248	
	C248OP		SM104	
	C249		SM249	
	C249OP		SM106	
	C250		SM250	
双相计数	C251		SM251	SM100
	C252		SM252	SM100
	C253		SM253	SM101
	C253OP		SM104	SM101
	C254		SM254	SM100
	C254OP		SM105	SM102
	C255		SM255	SM101
	C255OP		SM106	SM103

8.1.3 高速计数器使用方法

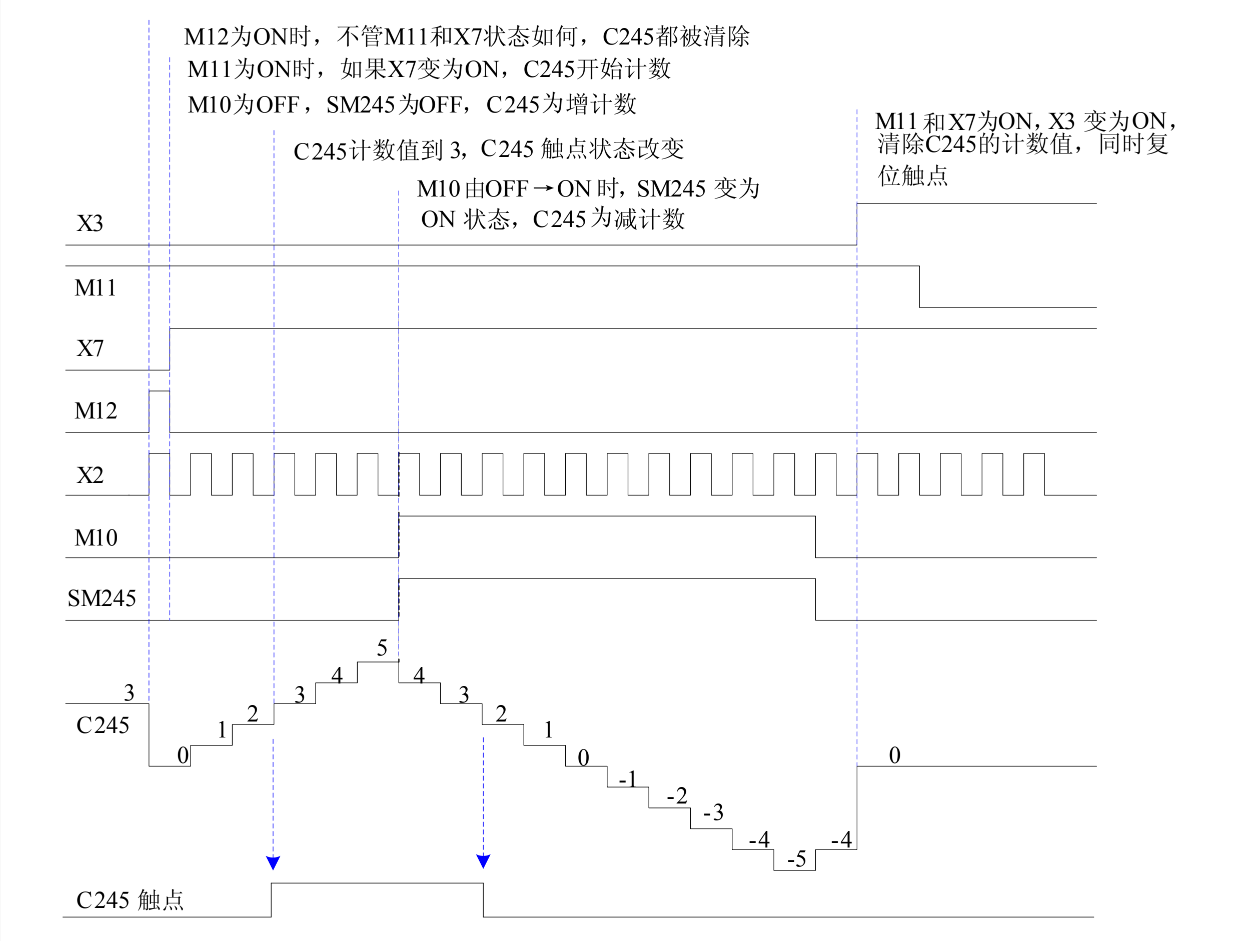
- **单相单端计数输入高速计数器的使用方法**

单相单端计数输入高速计数器的特点：只有在脉冲输入由 OFF→ON 变化时计数，计数器的增减方向由对应的特殊辅助继电器（SM 元件）决定。

动作示例如下所示：



程序中触点的时序操作图：



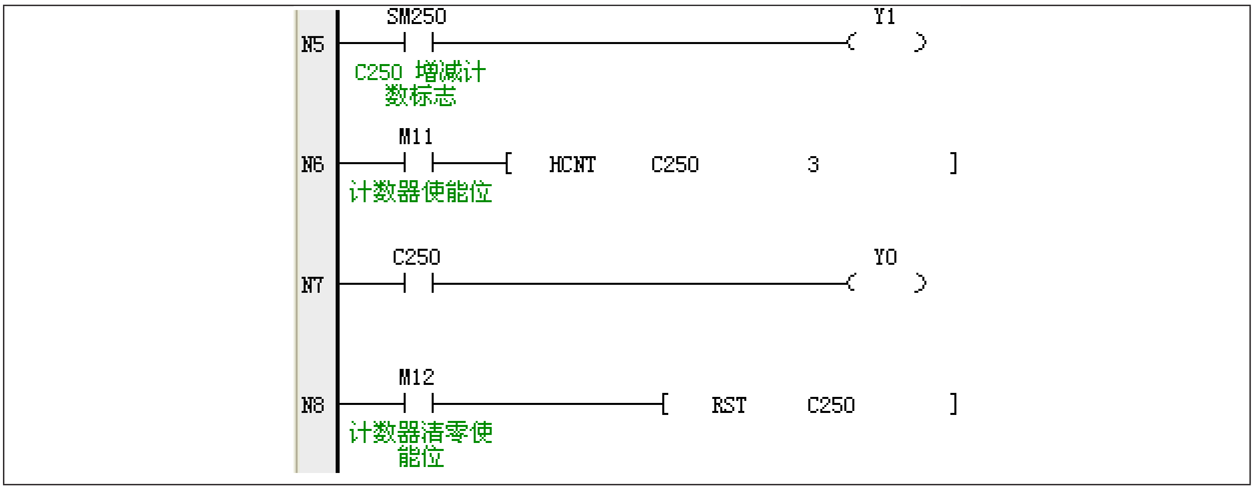
注： 1．计数器输入端为X2；

2．通过和高速指令DHSCS、DHSCI、DHSCR、DHSZ、DHSP、DHST和DHSTI可以产生与扫描周期无关的操作或控制。

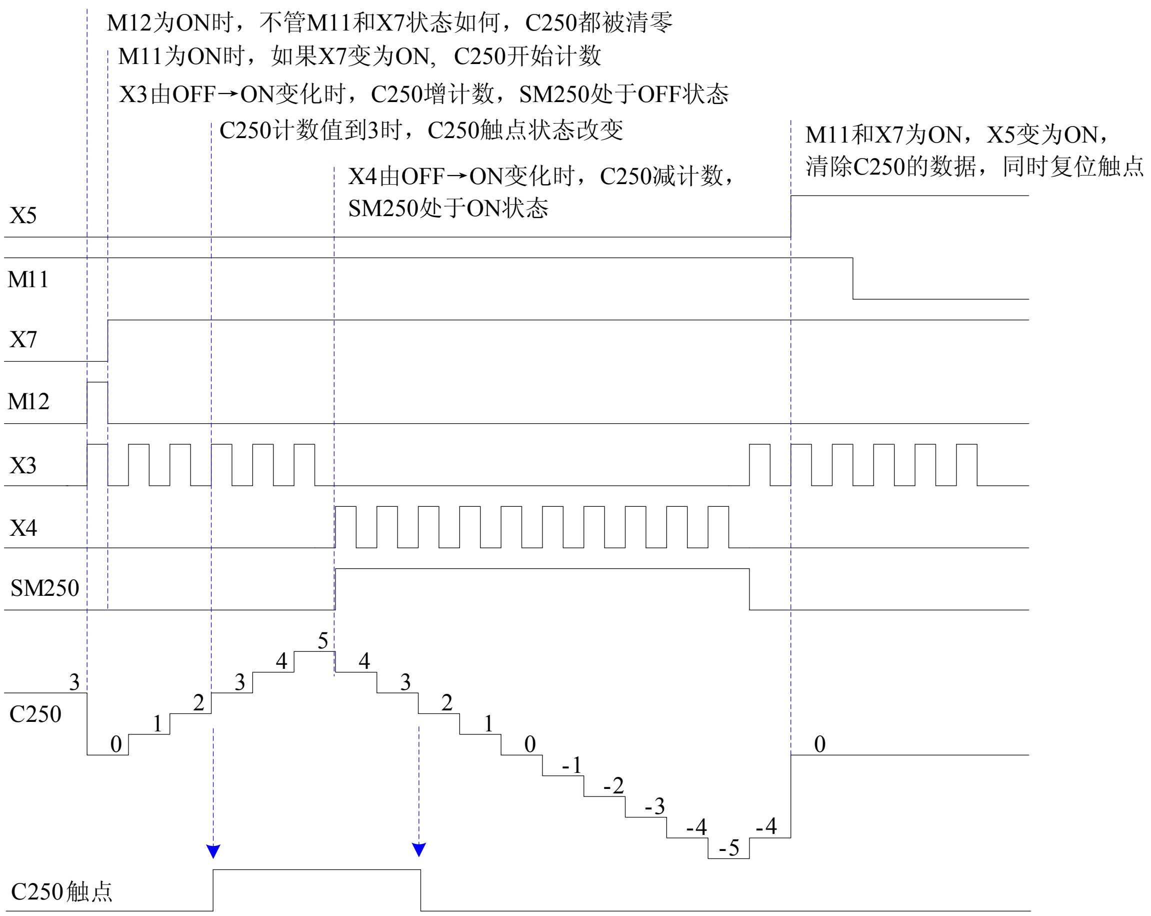
● **单相增减计数输入高速计数器的使用方法**

单相增减计数输入高速计数器的特点：只有在脉冲输入由 OFF → ON 变化时计数，两个输入点分别决定计数器的增方向和减方向。对应的特殊辅助继电器（SM 元件）显示当前高速计数器的增减状态。

动作示例如下所示：



程序中触点的时序操作图：

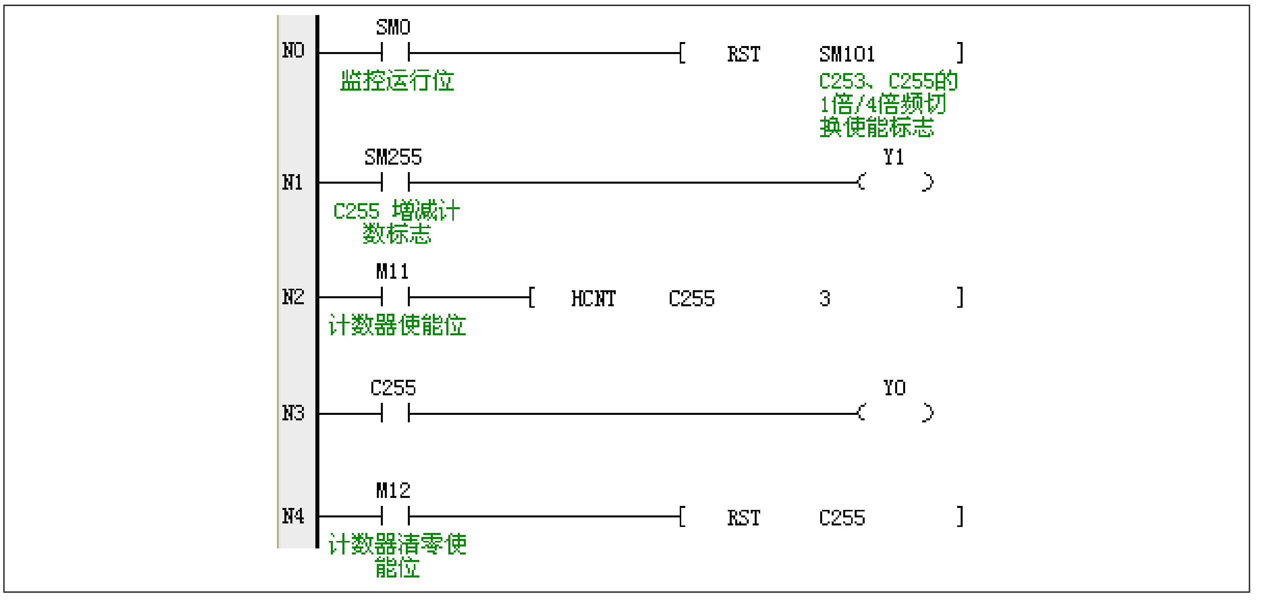


注： 1．计数器输入端为X3和X4；
2．通过和高速指令DHSCS、DHSCI、DHSCR、DHSZ、DHSP、DHST和DHSTI可以产生与扫描周期无关的操作或控制。

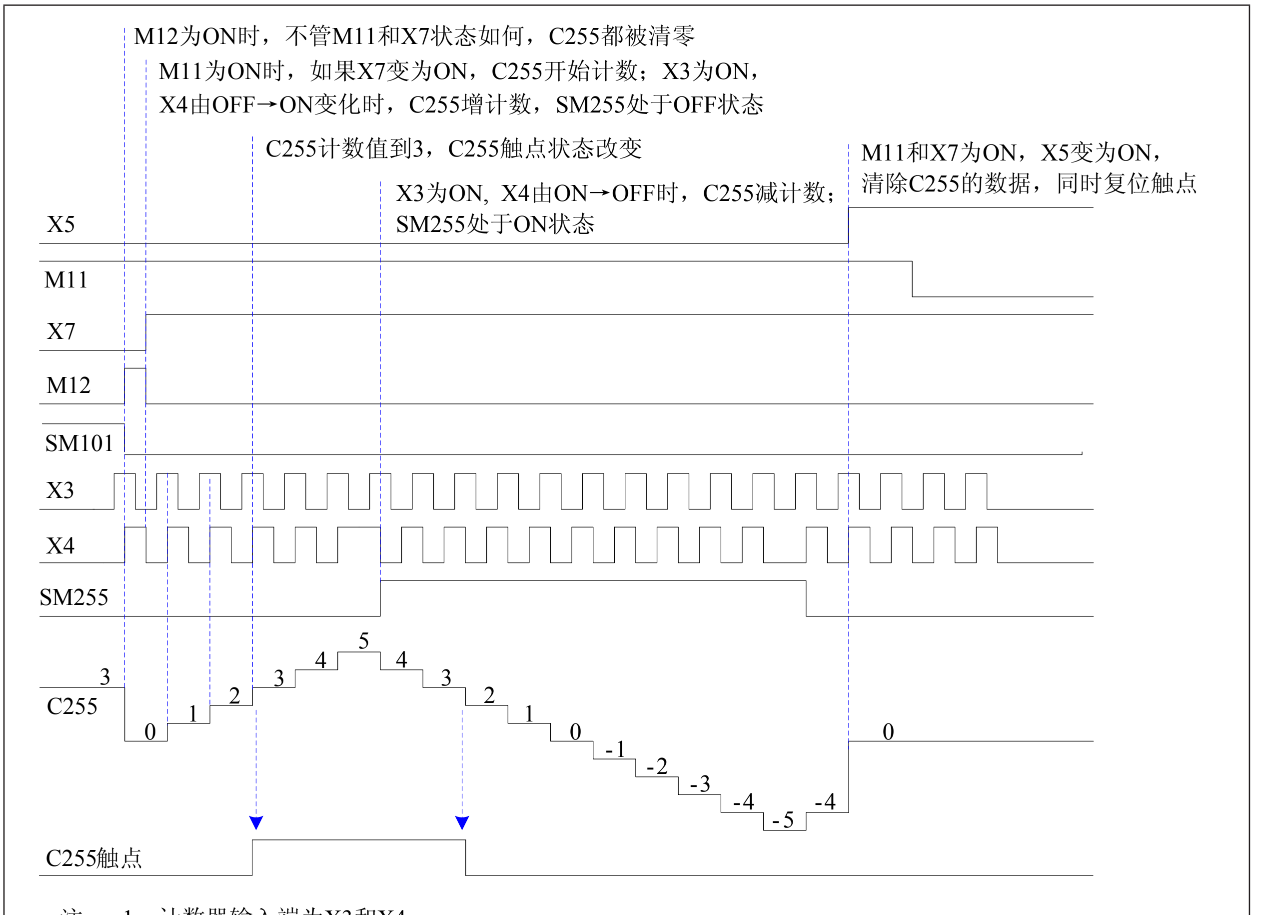
- **双相 1 倍频计数输入高速计数器的使用方法**

双相 1 倍频计数输入高速计数器的特点：只有在脉冲输入由 OFF → ON 变化时计数，计数器的增减方向分别由两个输入点的相位差决定。高速计数器对应的特殊辅助继电器（SM 元件）显示当前高速计数器的增减状态。

动作示例如下所示：



程序中触点的时序操作图：

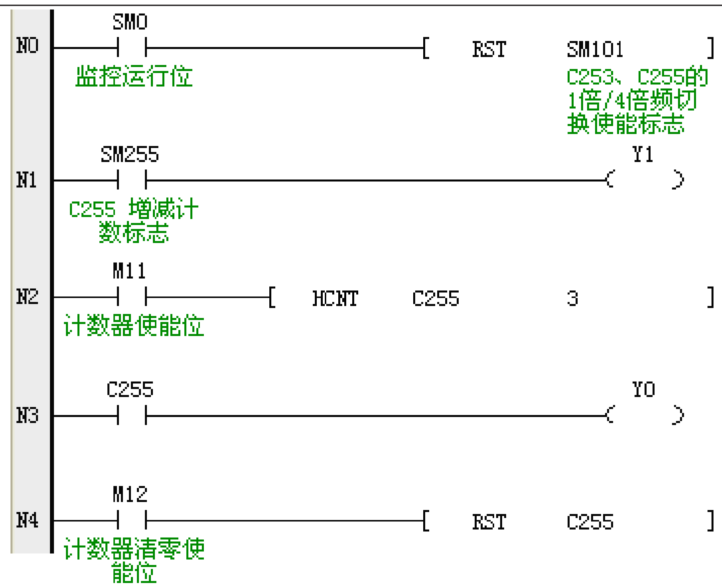


注： 1．计数器输入端为X3和X4；
2．通过和高速指令DHSCS、DHSCI、DHSCR、DHSZ、DHSP、DHST和DHSTI可以产生与扫描周期无关的操作或控制。

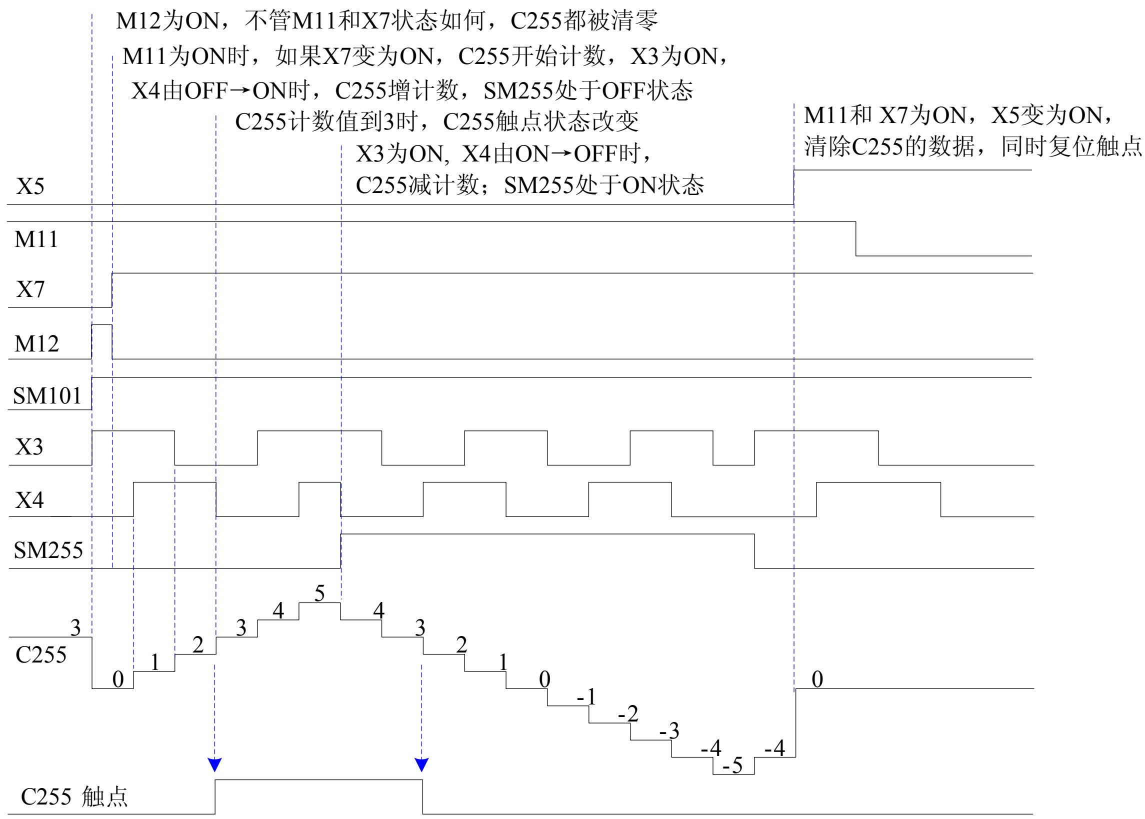
● **双相 4 倍频计数输入高速计数器的使用方法**

双相 4 倍频计数输入高速计数器的特点：脉冲双输入在 OFF → ON、ON → OFF 时均计数，计数器的增减分别由两个输入点的相位差决定。高速计数器对应的特殊辅助继电器（SM 元件）显示当前高速计数器的增减状态。

动作示例如下所示：



程序中触点的时序操作图：



注： 1．计数器输入点为X3和X4；

2．通过和高速指令DHSCS、DHSCI、DHSCR、DHSZ、DHSP、DHST和DHSTI可以产生与扫描周期无关的操作或控制。

8.1.4 MX 系列中高速计数器的注意事项

● **频率总和限制**

多个高速计数器和 SPD 同时使用的情况下，总计输入频率数为下表所示：

使用条件	总计输入频率数
DHSCS、DHSCR、DHSCI、DHSZ、DHSP、DHST、DHSTI 都没使用	≤ 100kHz
DHSCS、DHSCR、DHSCI、DHSP、DHST、DHSTI 有使用	≤ 60kHz
DHSZ 有使用	≤ 30kHz

8.2 外部脉冲捕捉功能

支持外部脉冲捕捉功能的硬件输入点为 X0 ～ X7。所对应的 SM 软元件表如下：

输入硬件端口	软元件 SM
X0	SM90
X1	SM91
X2	SM92
X3	SM93
X4	SM94
X5	SM95
X6	SM96
X7	SM97

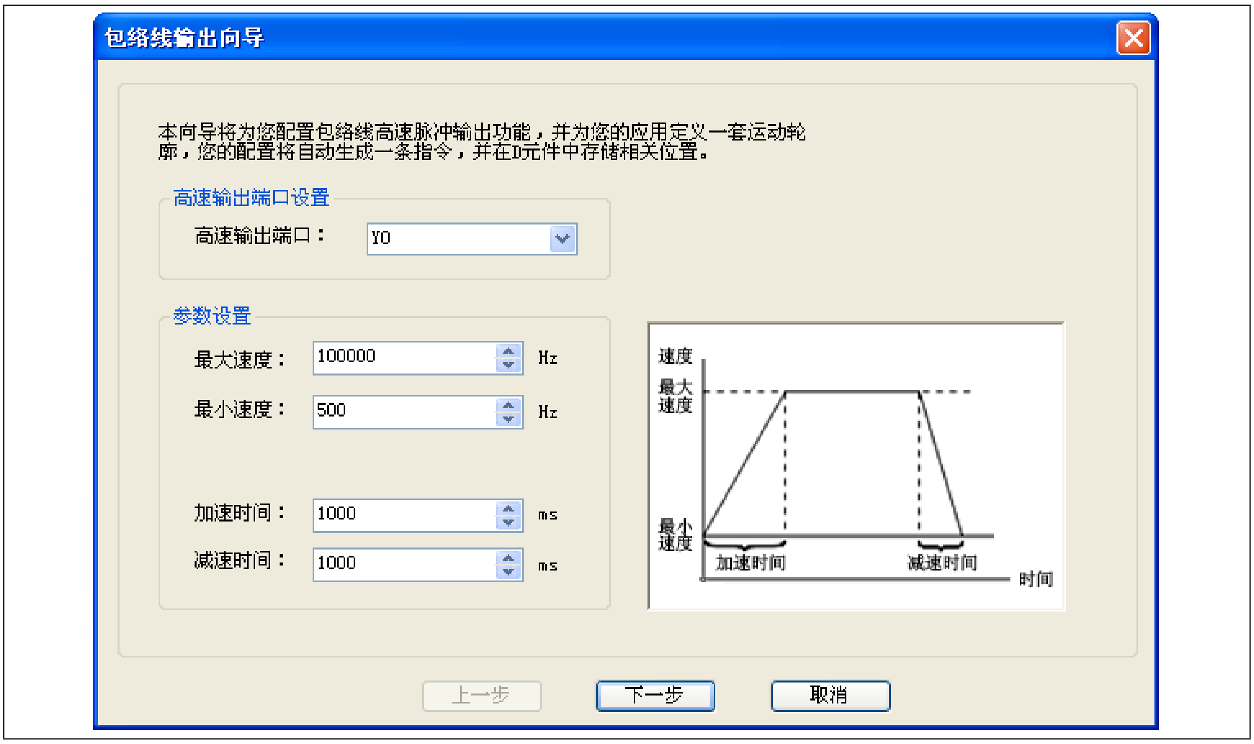
☞ 注意

1．外部输入点由 OFF → ON 变化时，对应端口的软元件 SM 置为 ON。
2．由 STOP 到 RUN 时，SM90 ～ SM97 被清除。
3．使用脉冲捕捉时，仍然须遵守各 PLC 系列对输入脉冲频率的总和限制，否则可能出现异常。
4．在本端口上有 HCNT 高速计数器驱动指令或 SPD 测频指令时，端口的脉冲捕捉无效，在其它情况下都有效。

具体说明请参见 *6.7.1 HCNT* 高速计数器驱动指令和 *6.7.9 SPD* 测频指令。

8.3 PLS 包络线指令配置方法

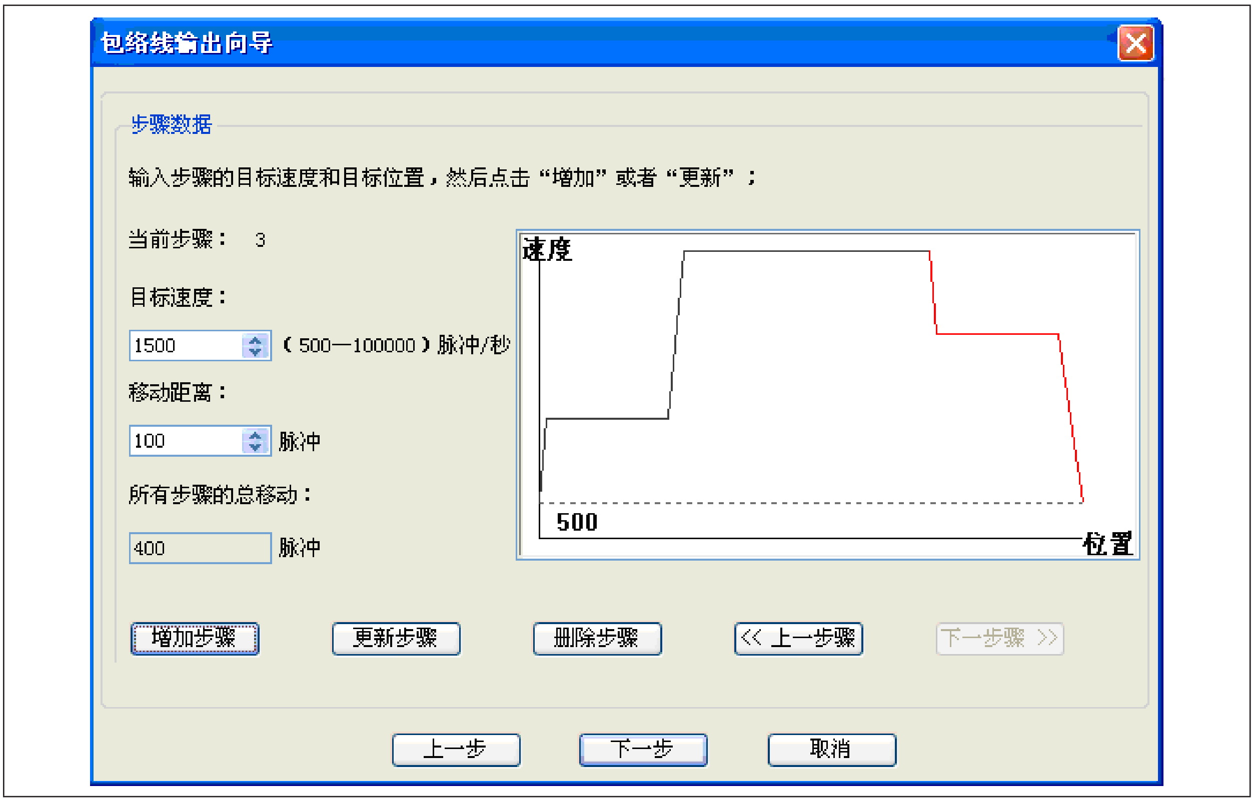
1．PLS 包络线指令可以通过 PTO 指令向导生成，还可以通过编写 PLS 指令生成，通常使用 PTO 指令向导。在 MXProgrammer 中，选择菜单栏中的“工具”→“ PTO 指令向导…”或点击工具栏中 PTO 向导快捷图标“ ”，即可对高速脉冲输出端口，高速脉冲输出的最大速度、最小速度，以及加速时间、减速时间进行配置，配置界面如下图所示：



包络线所有段的加速时间 / 减速时间即为上图中所设定的加速时间 / 减速时间。

如果对加 / 减速过程中的脉冲个数有要求时，请参考 *6.7.12 PLS* 指令。

2．点击上图中的“下一步”按钮进入下图所示界面，输入第一个步骤的目标速度和移动距离，点击“增加步骤”，即可生成第一步。若要生成多个步骤，请参照第一步骤的生成方法。若要修改某一步的参数，可以通过点击“上一步”或“下一步”按钮选中需要修改的步骤，参数重新设定完成后点击“更新步骤”即可完成修改。



3．上述配置完成后，点击“下一步”按钮，向导会要求为包络线的配置分配 D 元件（默认起始地址为 D3000），用户

可自行选择 D 元件来存储这些设置。输入子程序的名称后点击“完成”按钮，可完成 PTO 的配置。如下图所示：

在此界面中可以选择是否将注释加入到全局变量表中。

8.4 高速输入应用注意事项

输入点 X0 ～ X7 在高速计数器、SPD 测频指令、脉冲捕捉、外部中断等功能中作为输入信号。在对 PLC 编程时，每一个输入点所对应的多项功能中只能采用其中 1 项，否则，用户程序将不能通过编译。

输入点 X0 ～ X7 在高速计数器中的应用请参见 *8.1.1* 高速计数器配置。

下表标明了输入点 X0 ～ X7 在 SPD 测频指令、脉冲捕捉、外部中断等功能中的应用：

输入点 / 功能	X0	X1	X2	X3	X4	X5	X6	X7	最高频率 KHz	
									1H/2H	2V
SPD 测频指令	√	√	√	√	√	√			10	200
脉冲捕捉功能	√	√	√	√	√	√	√	√		
外部中断编号（上升）	0	1	2	3	4	5	6	7	5	200
外部中断编号（下降）	10	11	12	13	14	15	16	17		

8.5 高速脉冲输出

8.5.1 高速脉冲输出功能

高速脉冲输出是在 Y（1H：Y0、Y1；2H：Y0、Y1、Y2；2V：Y0 ～ Y7）端口输出可控脉冲，脉冲的输出与用户程序的扫描周期无关。采用 PLSY、PLSR、PLS、PWM 指令可以控制高速脉冲输出。这些指令的使用方法请参见 *6.7* 高速 *I/O* 指令。在程序中，两个不同的端口同时使用两个高速脉冲输出指令时，能够在两个不同的端口得到各自独立的高速脉冲输出，这两个高速脉冲输出可以用相同的高速指令也可以用两个不同的高速指令。

8.5.2 高速脉冲输出注意事项

在高速指令有效运行（包括输出完成）时，对同一端口的其它操作无效。只有在高速脉冲输出指令无效时，其它指令才有正确的输出。

在同一端口有多个 PWM、PLSY 或 PLSR 指令时，先有效的指令占用输出端，后有效的指令不占用输出端。

第九章　中断服务程序使用指南

本章详细讲述了各种中断的机制、处理过程及使用方法。

9.1 理解中断程序

当产生中断事件后，正在执行的程序将被打断，转而执行相应的中断程序，等中断程序执行完成后，再继续执行被中断的程序，这就是中断处理机制。用户常常需要采用这种特殊的系统处理机制来处理时效性要求高、事件触发型的控制任务。

系统内设有多种可编程的中断资源，分别对应多种中断事件，用户可通过编写相应的中断程序来处理特定的中断事件。在用户程序中，中断程序是独立的程序体。为了使得用户中断程序和指定了事件编号的中断事件建立一一对应的关系。每一个中断程序都必须指定一个中断事件编号。当该类中断事件的中断请求被响应时，系统会根据中断事件编号调用相应的用户中断程序。系统提供中断程序的加密属性选择，用户可根据实际需要给中断程序加密。

MX 系列 PLC 的中断资源列表

中断事件编号	对应中断事件	中断使能控制 SM	中断事件编号	对应中断事件	中断使能控制 SM
0	X0 输入上升沿中断	SM30	29	交流电源失电中断	SM56
1	X1 输入上升沿中断	SM31	30	PORT0 的字符发送中断	SM48
2	X2 输入上升沿中断	SM32	31	PORT0 的字符接收中断	SM49
3	X3 输入上升沿中断	SM33	32	PORT0 的帧发送中断	SM50
4	X4 输入上升沿中断	SM34	33	PORT0 的帧接收中断	SM51
5	X5 输入上升沿中断	SM35	34	PORT1 的字符发送中断	SM52
6	X6 输入上升沿中断	SM36	35	PORT1 的字符接收中断	SM53
7	X7 输入上升沿中断	SM37	36	PORT1 的帧发送中断	SM54
10	X0 输入下降沿中断	SM40	37	PORT1 的帧接收中断	SM55
11	X1 输入下降沿中断	SM41	38	PORT2 的字符发送中断	SM57
12	X2 输入下降沿中断	SM42	39	PORT2 的字符接收中断	SM58
13	X3 输入下降沿中断	SM43	40	PORT2 的帧发送中断	SM59
14	X4 输入下降沿中断	SM44	41	PORT2 的帧接收中断	SM60
15	X5 输入下降沿中断	SM45	50	PTO（Y0）输出完成中断	SM70
16	X6 输入下降沿中断	SM46	51	PTO（Y1）输出完成中断	SM71
17	X7 输入下降沿中断	SM47	52	PTO（Y2）输出完成中断	SM72
20	高速计数器中断 0	SM69	53	PTO（Y3）输出完成中断	SM73
21	高速计数器中断 1	SM69	54	PTO（Y4）输出完成中断	SM74
22	高速计数器中断 2	SM69	55	PTO（Y5）输出完成中断	SM75
23	高速计数器中断 3	SM69	56	PTO（Y6）输出完成中断	SM76
24	高速计数器中断 4	SM69	57	PTO（Y7）输出完成中断	SM77
25	高速计数器中断 5	SM69	58	插补完成中断	SM78
26	定时中断 0	设定时值：SD66 使能标志：SM66	59	CAN0 接收中断	SM79
27	定时中断 1	设定时值：SD67 使能标志：SM67	60	PORT0（1PG-BD）输出完成中断	SM61
28	定时中断 2	设定时值：SD68 使能标志：SM68	70 ～ 93	高速计数表格比较中断	SM69

9.2 中断事件的处理过程

1．中断程序不是由程序调用，而是在中断事件发生时由系统调用。当某一中断事件发生，且该中断事件已被使能，则该中断事件编号将被加入到中断请求队列记录中，中断请求队列是一个深度为 16 的先入先出队列。

2．系统处理中断请求：

（1）系统检测到中断请求队列不为空时，打断正常的主程序执行流程。

（2）系统查询中断请求队列的队首记录，队首记录的是最先发生的中断事件的编号，该中断事件编号所对应的用户中断程序将被调用执行。

（3）当对应的中断程序被执行完毕时（执行了中断返回指令），该中断请求被处理完毕，请求队列的队首记录将被

从队列中删除，请求队列中次先记录变为队首记录，后续的记录也将向前移动一个位置。

（4）系统再次检测中断请求队列是否为空，若不为空，重新循环执行以上步骤，直至中断请求队列为空。

（5）当中断请求队列为空时，系统回到被中断的主程序继续执行。

3．系统一次只处理一个中断请求，如果在执行某个中断程序时，又检测到一个中断请求，新近发生的中断事件不会被立即响应，但会被记录到中断请求队列的队尾，等待系统处理完前面的中断请求后，再处理该中断请求。

4．因为不能预知系统何时触发中断程序，所以中断程序不能改写其他程序中正在使用的存储器，中断程序应该尽可能地使用局部变量。

5．当中断请求队列中的记录数达到 16 时，系统将自动屏蔽新的中断事件，即新的中断事件不能加入到中断队列。直至中断请求队列中所有请求被处理完毕，而且被中断的主程序也被执行完毕，系统才解除屏蔽。

☞ 注意

1．编写中断程序时应该遵循“越短越好”的原则，以减少中断程序的执行时间，减少对其他处理的延迟，否则可能引起主程序控制的设备操作异常。

2．中断程序中允许调用一级子程序。

3．要想在中断中立即刷新 I/O，请使用立即刷新指令（REF），注意 REF 执行时间跟所要刷新 I/O 点数有关。

4．使用某类中断事件产生中断请求时，应确保中断请求所属的中断事件标志应已被使能（每类中断事件使能 / 禁止控制都有相关的 SM 元件。若要使能某类中断事件，应将相应的 SM 元件置为 ON）且总中断使能标志 EI 是被打开的。

5．如果产生了相应的中断请求，但用户程序中没有与之对应的中断程序，系统也将响应该中断请求，但只做空操作。

9.3 使用定时中断

1．定时中断的说明

在主程序的执行周期很长的情况下，如果要处理特定的程序或者在顺控扫描中，需要每隔一段时间执行一次特定的程序时，定时中断功能非常实用。它可以不受扫描周期的影响，每隔 Nms 执行一次定时中断程序。

设定相应 SM 元件 ON/OFF 状态，可以使能 / 禁止定时中断。设定相应 SD 元件的值，可设定中断的定时值。所设定的中断定时值的计时单位为毫秒（ms），当定时值小于等于设定值时，系统不触发中断事件。

MX 系列 PLC 的定时中断资源一览表

定时中断	中断事件号	定时设定值（SD）	范围	使能标志（SM）
0	26	SD66	1 ～ 32767ms	SM66
1	27	SD67	1 ～ 32767ms	SM67
2	28	SD68	1 ～ 32767ms	SM68

☞ 注意

1．当禁止定时中断时，已加入中断队列中的定时中断仍被执行。

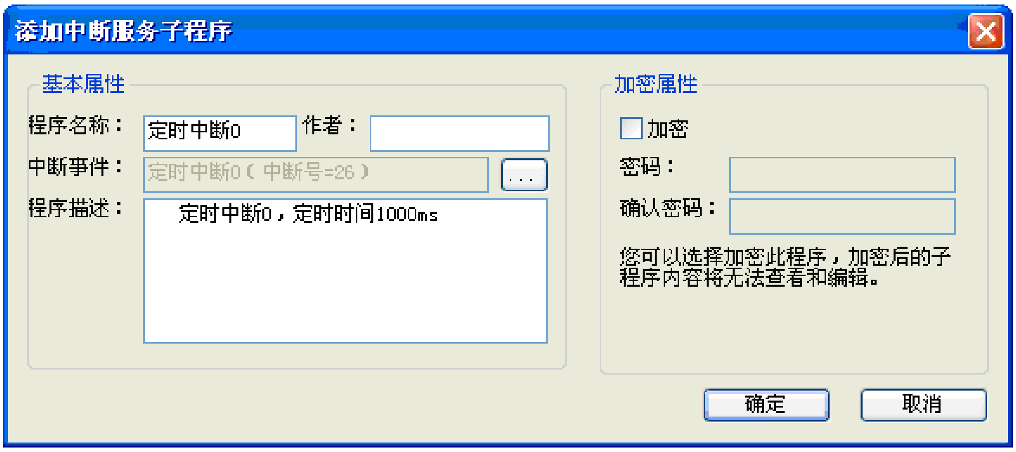
2．当定时中断被禁止后再使能时，定时的计时值将从零开始。

3．若想在程序运行中改变中断定时值的设定，请按照如下步骤操作：先禁止定时中断，改变定时值的设定，再使能该中断。

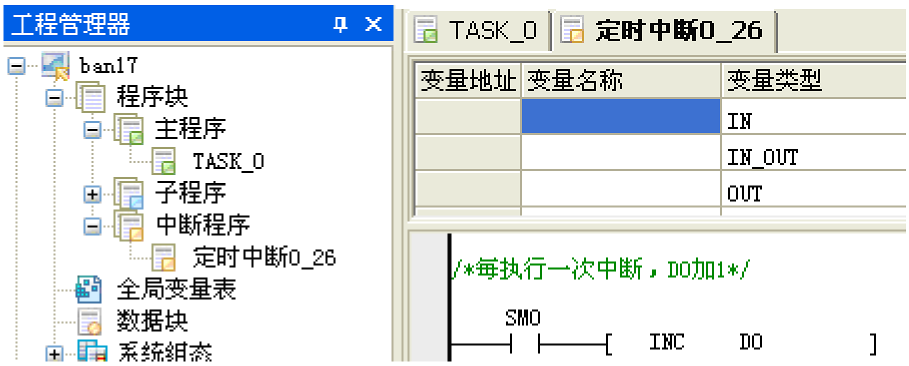
2．定时中断的使用示例

该示例是使用定时中断 0，定时时间设为 1000ms 即每隔 1000ms 执行一次中断程序。

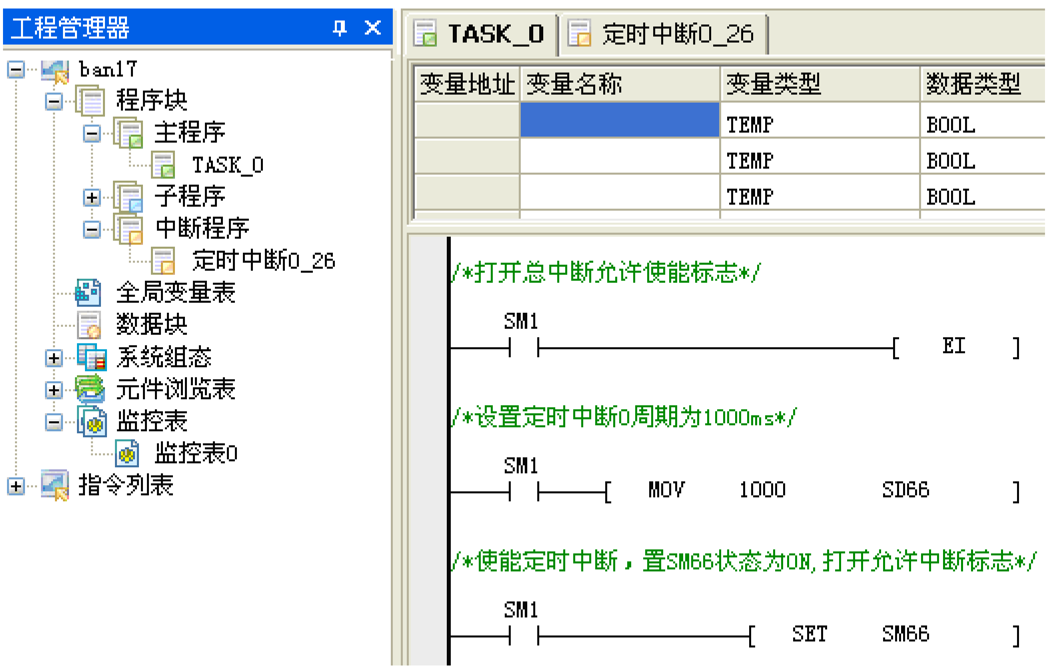
（1）右键点击中断程序选择“插入中断服务程序”，在弹出的对话框中选择需要的中断事件，如果要给选择的事件设定密码，可在右边加密属性里设置。在这个例子里我们选择“定时中断 0（中断号 =26）”，在程序描述里可填写需要的注释，然后点击确定即可，如下图所示：



（2）编写定时中断触发时要执行的中断服务程序。



（3） 在主程序中编写设置定时中断周期和使能定时中断的程序。



9.4 使用外部中断

1．外部中断说明

外部中断和 PLC 的实际输入点有关，对外部的中断分为输入上升沿中断和输入下降沿中断。在用户程序中与外部事件有关的动作放到外部中断程序中，系统对外部信号的响应频率最高为 5KHz。频率在 5KHz 以上的外部事件有可能丢失。所有的外部中断只有在总中断控制 EI 有效和对应的中断使能标志 SM 有效的情况下才有效。

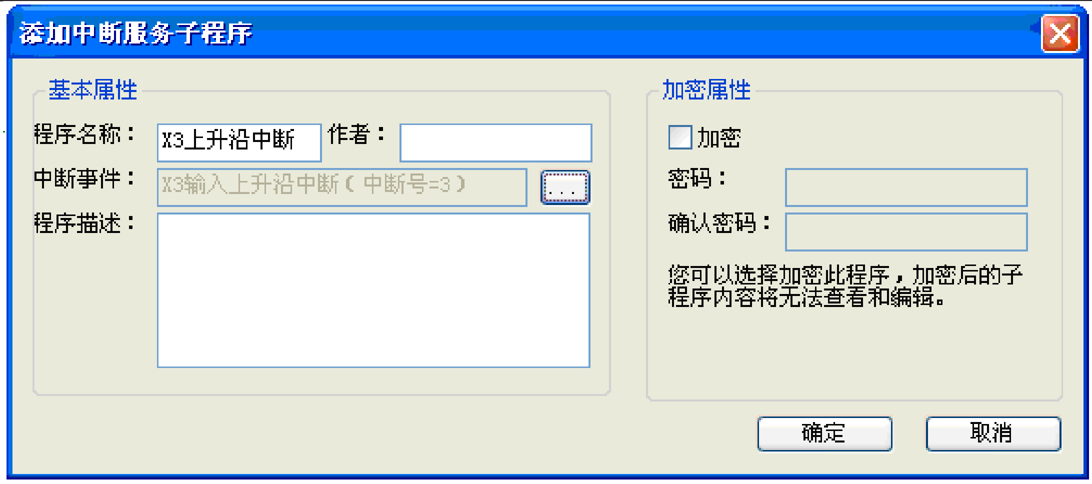
MX 系列 PLC 的外部中断资源一览表

中断编号	使能位	对应中断源		中断编号	使能位	对应中断源
0	SM30	X0 输入上升沿中断		10	SM40	X0 输入下降沿中断
1	SM31	X1 输入上升沿中断		11	SM41	X1 输入下降沿中断
2	SM32	X2 输入上升沿中断		12	SM42	X2 输入下降沿中断
3	SM33	X3 输入上升沿中断		13	SM43	X3 输入下降沿中断
4	SM34	X4 输入上升沿中断		14	SM44	X4 输入下降沿中断
5	SM35	X5 输入上升沿中断		15	SM45	X5 输入下降沿中断
6	SM36	X6 输入上升沿中断		16	SM46	X6 输入下降沿中断
7	SM37	X7 输入上升沿中断		17	SM47	X7 输入下降沿中断

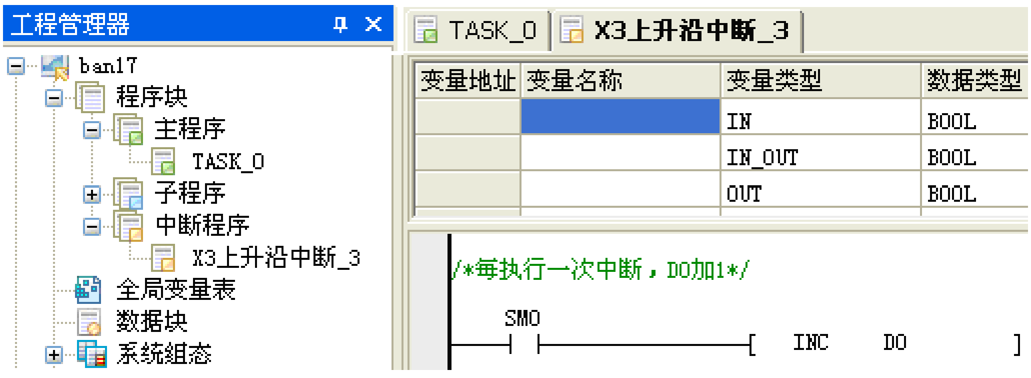
2．外部中断使用示例

该示例是使用 X3 对应的外部中断，根据 X3 的上升沿输入事件对 D0 加 1。

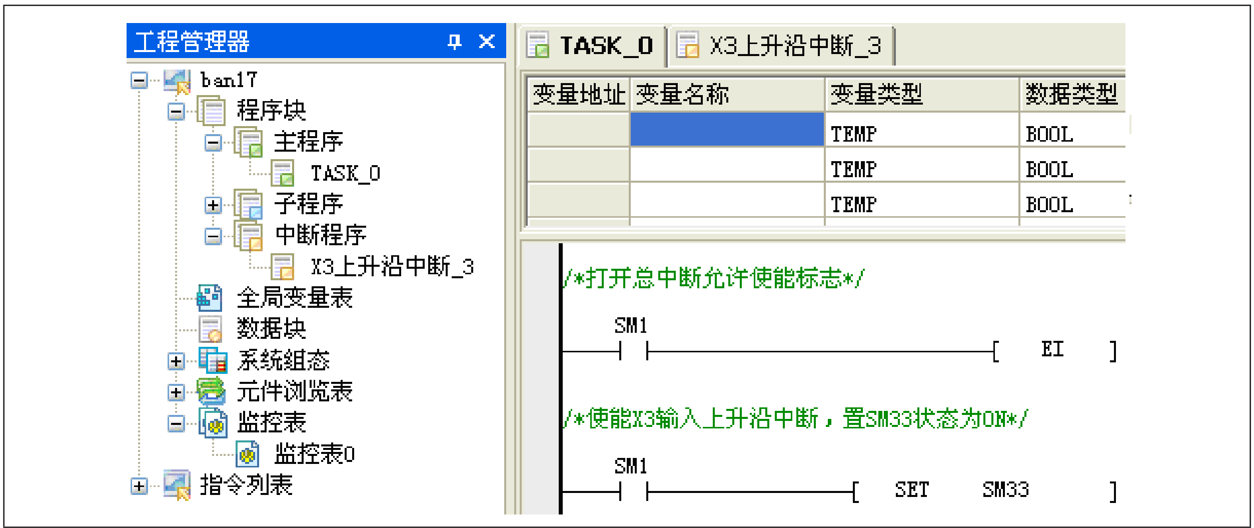
（1）添加 X3 输入上升沿中断服务子程序，如下图所示：



（2）编写中断服务程序，X3 每次由 OFF → ON 变化时，执行一次中断服务程序，D0 加 1。



（3）在主程序里写入 EI 指令，并使 X3 输入上升沿中断对应的中断使能 SM33 有效。



9.5 使用高速计数器中断

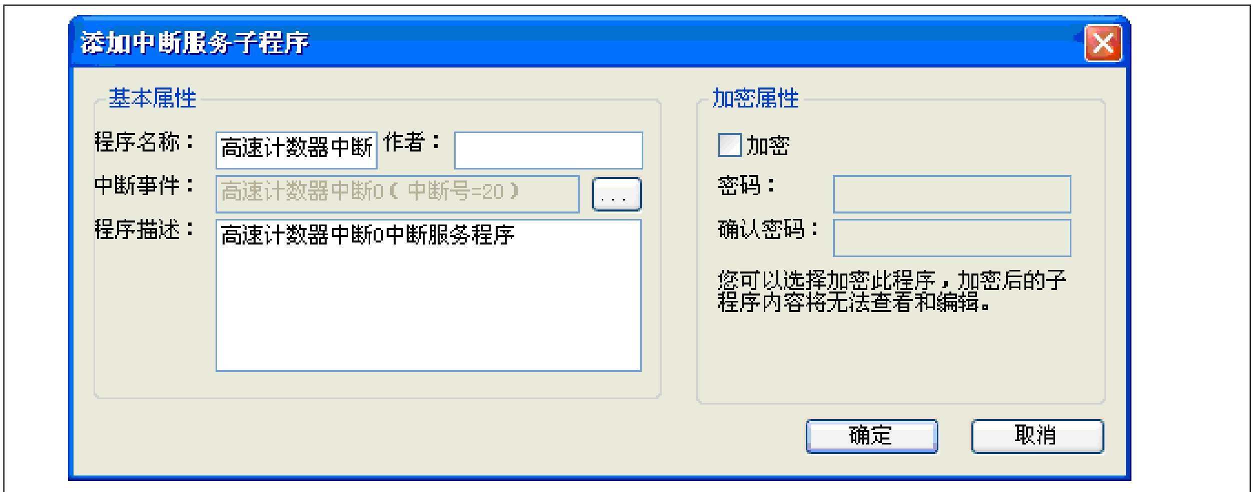
1．高速计数器中断说明

高速计数器中断必须要和高速 HCNT、DHSCI 指令配合使用才有效，根据高速计数器的计数值产生高速计数器中断。在高速中断程序中用户可编写与外部脉冲输入有关的程序。所有的高速计数器中断（中断号为 20 ～ 25）只有在总中断使能 EI 有效和对应的中断使能标志（SM69）有效的情况下才有效。

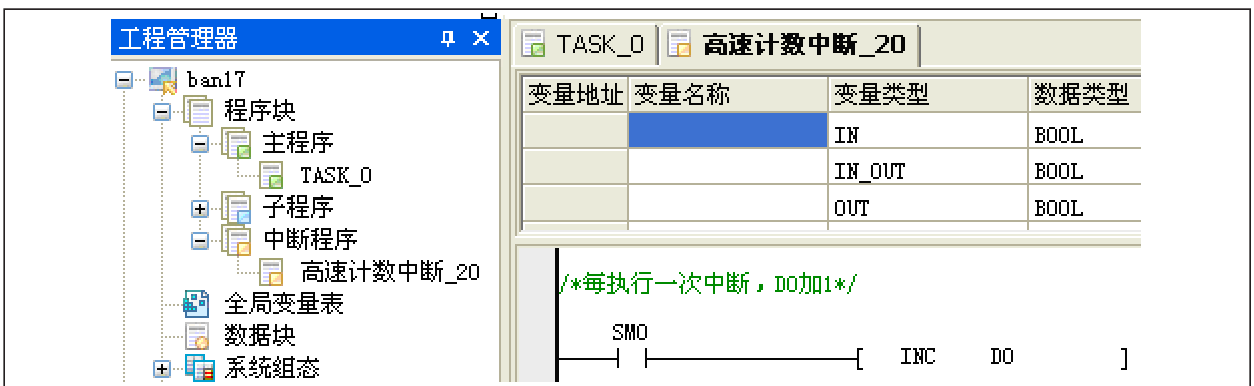
2．高速计数器中断使用示例

该示例是使用高速计数器 C235 计数，当其数值达到 DHSCI 所指定的数值时，则响应中断号为 20 的中断服务子程序。

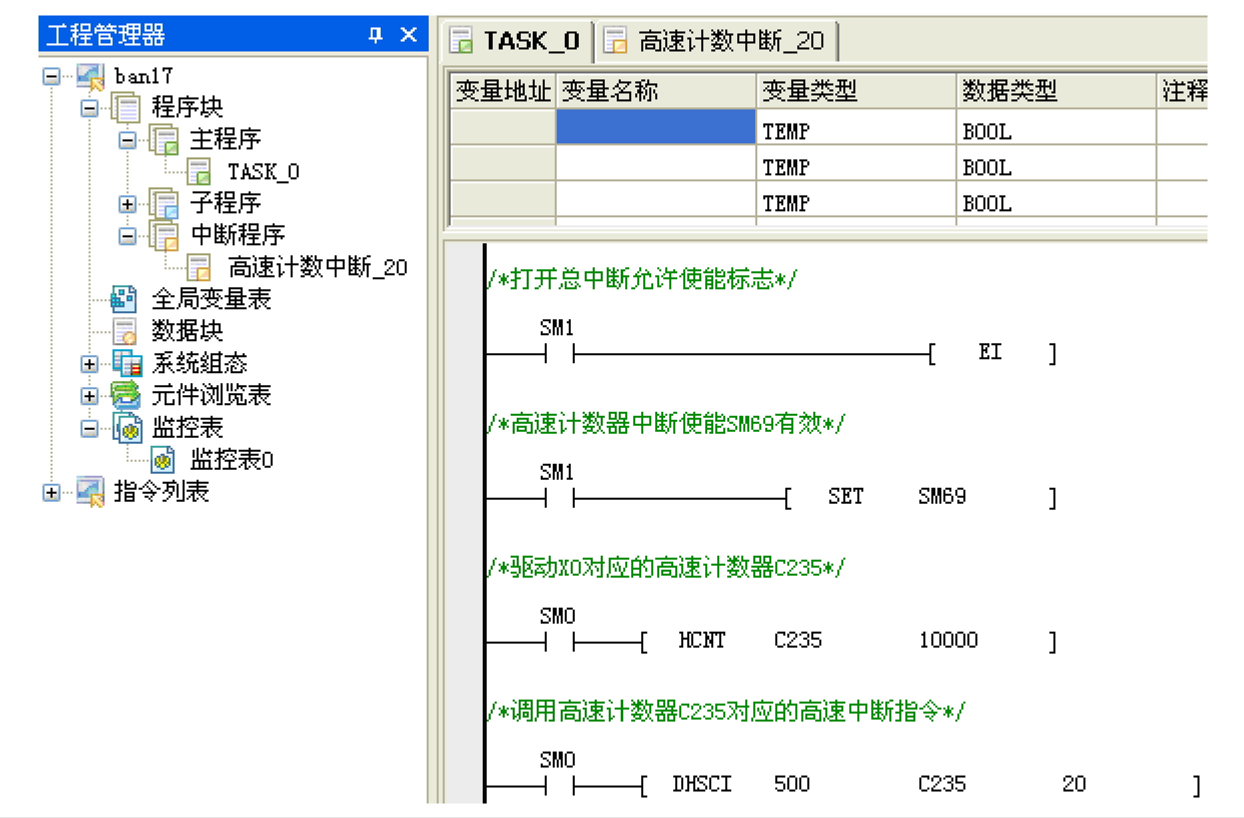
（1） 添加高速计数器中断 0 服务子程序，如下图所示：



（2） 编写中断服务子程序，每执行一次中断，D0 加 1 ，如下图所示：



（3）在主程序里写入 EI 指令，并使高速计数器中断使能标志 SM69 有效，驱动高速计数器 C235，调用高速计数比较中断触发指令 DHSCI。



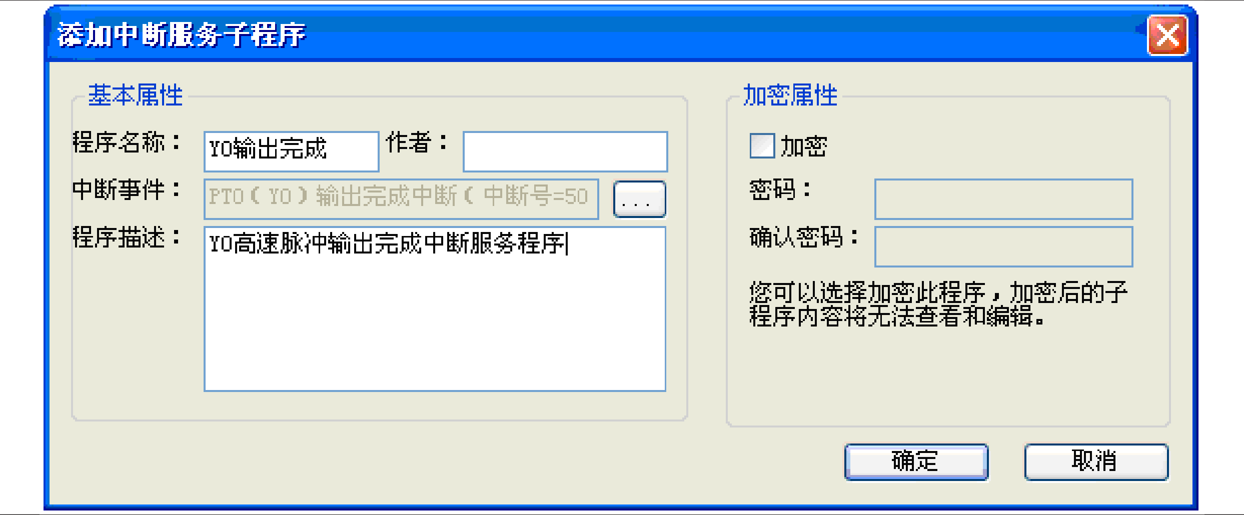
9.6 使用 PTO 输出完成中断

1．PTO 输出完成中断说明

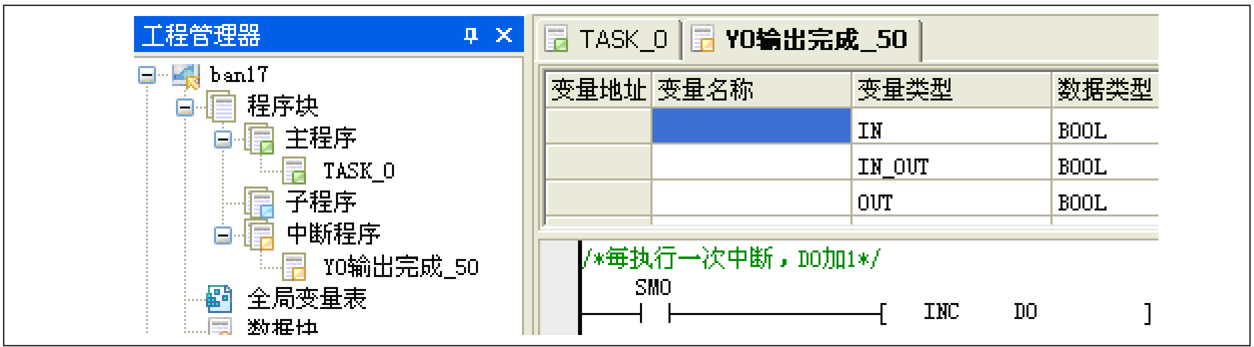
在使能标志 SM70、SM71、SM72、SM73、SM74、SM75、SM76、SM77 置位的情况下，对应 Y0、Y1、Y2、Y3、Y4、Y5、Y6、Y7 八个端口高速脉冲输出完成后，触发 PTO 输出完成中断，用户可以在该中断子程序内进行相关操作。

2．PTO 输出完成中断示例

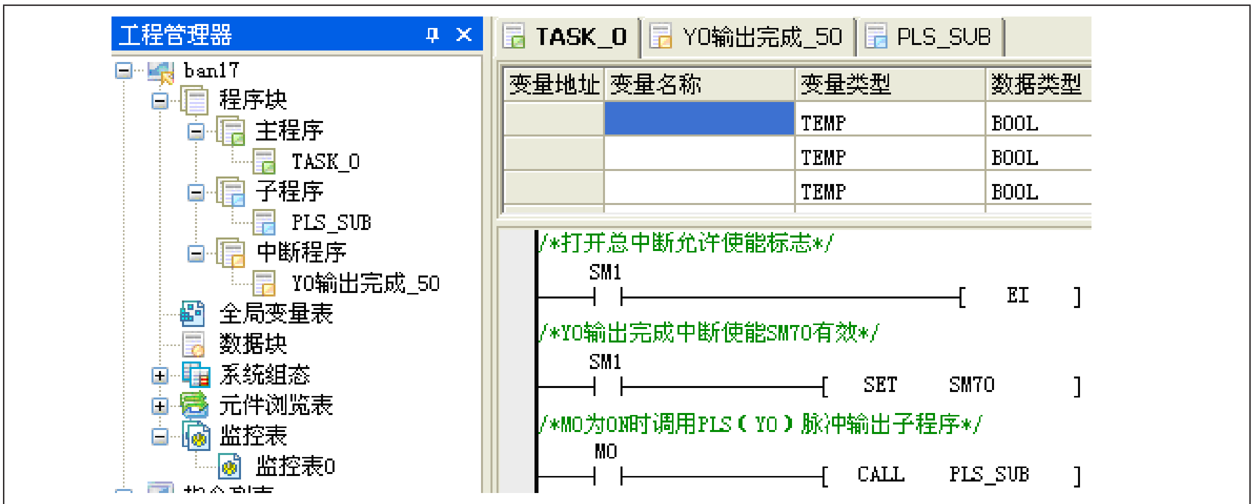
（1）该示例是使用 Y0 对应的 PTO 输出完成中断，当 Y0 高速脉冲输出完成后，则响应中断号为 50 的中断服务子程序。添加 PTO（Y0）输出完成中断服务子程序，如下图所示：



（2）编写中断服务子程序，每执行一次中断，D0 加 1，如下图所示：



（3）在主程序里写入 EI 指令，并使 PTO 输出完成中断使能标志 SM70 有效，调用 PLS 指令。



9.7 使用电源失电中断

在使能标志 SM56 置位的情况下，若主模块检测到电源失电则触发电源失电中断，用户可以在该中断子程序内进行相关操作。

由于电源失电中断子程序是在系统没有外供电源的情况下执行的，所以请控制电源失电中断子程序的执行时间不超过 5ms，否则会导致掉电保持元件无法完全保存。

9.8 使用串口中断

1．串口中断说明

当串口使用自由口协议时，根据串口的发送、接收，系统将产生中断事件。

对于每一个串口，系统给用户提供了 4 个中断资源，串口中断程序主要应用于要求对串口的字符与帧的发送、接收操作做特殊处理，且要求系统及时处理（可不受扫描周期的影响而立即响应）的场合。

设定相应 SM 元件的 ON/OFF 状态，可使能 / 禁止串口中断，当禁止串口中断时，已加入中断队列中的串口中断仍被执行。

在字符发送中断处理子程序中请不要持续调用串口发送指令（XMT），这样可能导致中断子程序嵌套而阻碍用户程序的执行。

帧发送、帧接收中断是指在串口发送指令（XMT）、串口接收指令（RCV）完成后触发的中断事件。

串口中断资源一览表

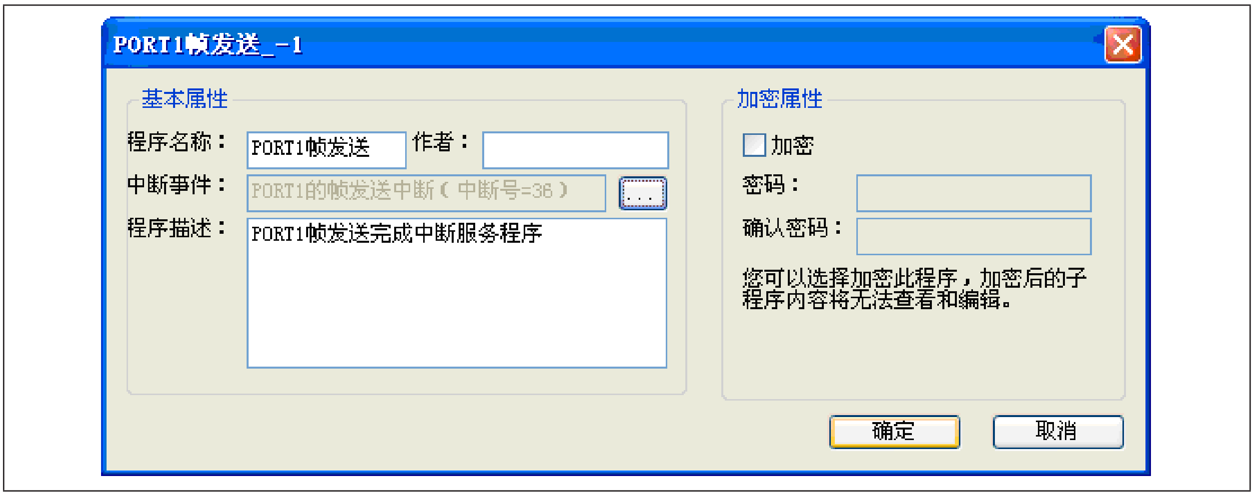
中断事件编号	对应中断事件	中断使能标志 SM 元件编号
30	PORT0 的字符发送中断	SM48
31	PORT0 的字符接收中断	SM49
32	PORT0 的帧发送中断	SM50
33	PORT0 的帧接收中断	SM51
34	PORT1 的字符发送中断	SM52
35	PORT1 的字符接收中断	SM53
36	PORT1 的帧发送中断	SM54

中断事件编号	对应中断事件	中断使能标志 SM 元件编号
37	PORT1 的帧接收中断	SM55
38	PORT2 的字符发送中断	SM57
39	PORT2 的字符接收中断	SM58
40	PORT2 的帧发送中断	SM59
41	PORT2 的帧接收中断	SM60

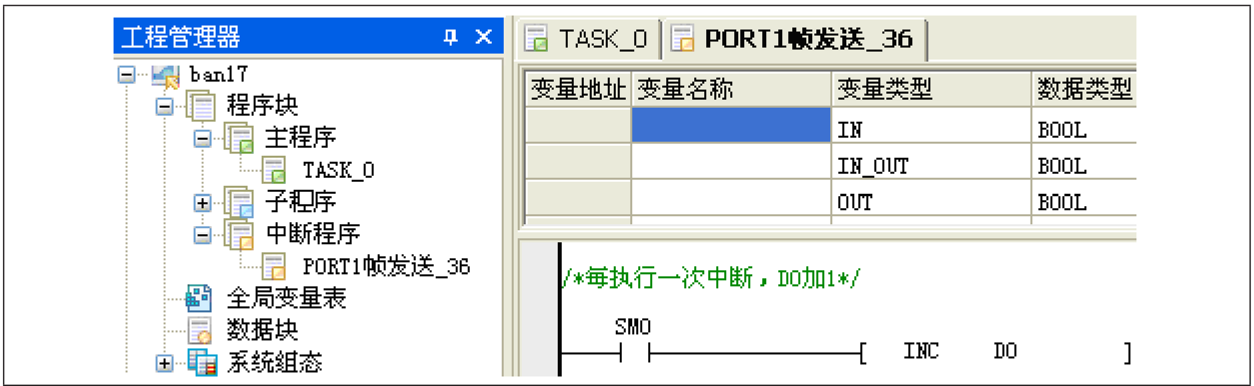
2．串口中断使用示例

该示例是使用 PORT1 的帧发送中断，每一帧数据发送完成后，都会响应中断号为 36 的中断服务子程序。

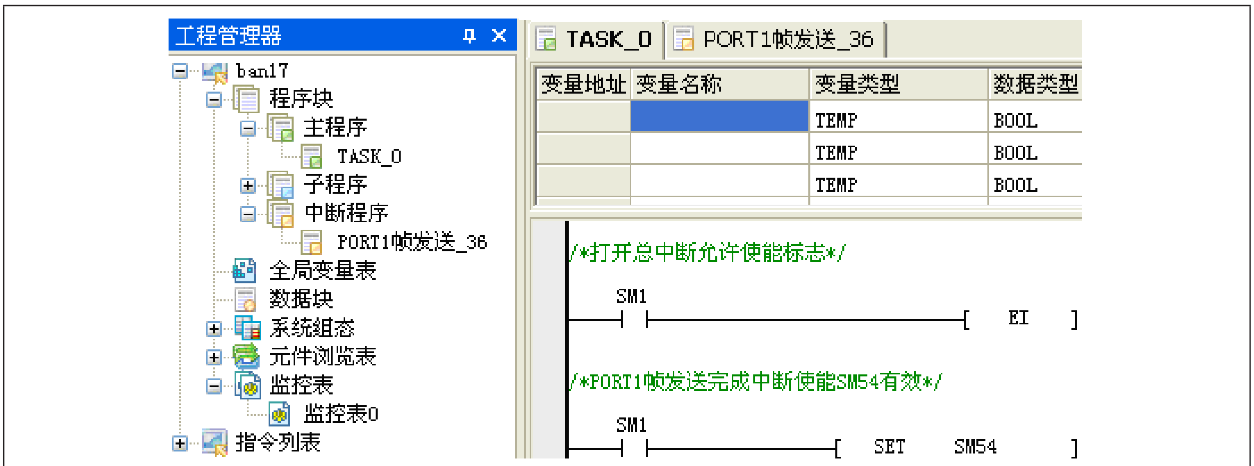
（1）选择 PORT1 的帧发送中断，如下图所示。



（2） 编写中断程序，PORT1 帧发送完成后执行中断程序，D0 加 1。



（3）在主程序里写入 EI 指令，并使 PORT1 帧发送完成中断使能标志 SM54 有效。



第十章 通讯功能使用指南

本章详细介绍了 MX 系列小型 PLC 的通讯功能，包括通讯资源、通讯协议，并通过实例加以说明。

10.1 通讯资源

MX 系列 PLC 适用的波特率：

通讯口类型	适用的波特率（BPS）
RS232/RS485	1200，2400，4800，9600，19200，38400，57600，115200
CAN	20K，40K，50K，100K，125K，250K，500K，自定义，默认为 125K

MX 系列 PLC 支持的通讯协议：

主模块	通讯口	通迅口类型	支持的协议
MX1H/2H/2V	PORT0	RS232	编程口协议，自由口协议，Modbus 通讯协议（从站）
	PORT1	RS485	自由口协议，Modbus 通讯协议（主站、从站），MXLink 通讯协议（主站，从站）
	CAN0	CAN	自由口协议， MXLink 通讯协议（主站，从站）
MX2H/2V	PORT2	RS232	编程口协议，自由口协议，Modbus 通讯协议（从站）
		RS485	自由口协议，Modbus 通讯协议（主站、从站），MXLink 通讯协议（主站，从站）

MX2H 系列 PLC 的 PORT2 和 CAN0 通讯口由通讯扩展卡提供。

10.2 编程口通讯协议

编程口协议是上位机软件 MXProgrammer 与主模块通讯的专用协议，不对外开放。

10.3 自由口通讯协议

10.3.1 简介

自由口协议是用户定义数据文件格式的通讯方式，可由指令实现数据的发送和接收。自由口协议支持 RTU 模式。只有在 PLC 处于 RUN（运行）模式时才能使用自由口通讯。

自由口的通讯指令包括 XMT（自由口发送指令）、RCV（自由口接收指令）。

10.3.2 自由口参数设置

在系统组态→硬件设置→选中主模块或者通讯扩展卡，在硬件设置中选择串口的通讯协议为自由口协议，从而激活相应的自由口设置按钮，点击自由口设置按钮，进入自由口参数设置界面，如下图所示：

可配置内容如下：

选项	设置内容	注释
波特率（BPS）	1200，2400，4800，9600，19200，38400，57600，115200，默认为 9600	—

选项	设置内容	注释
数据位	设定 7 或 8，默认为 8	不支持数据位为 7、无校验码、停止位为 1 的通讯协议。
奇偶校验位	设定为无校验、奇校验、偶校验，默认为无校验	
停止位	设定 1 或 2，默认为 1	
允许起始字符检测	允许或禁止，默认为禁止	—
起始字符检测	0 ～ 255（对应 00 ～ FF）	检测到用户指定的起始字符，开始接收，并将接收到的字符（包括起始字符）保存到用户指定的缓存区中。
允许结束字符检测	允许或禁止，默认为禁止	—
结束字符检测	0 ～ 255（对应 00 ～ FF）	当接收到用户设定的结束字符，结束接收，并将结束字符保存到缓存区内。
允许字符间超时时间	允许或禁止，默认为禁止	—
字符间超时时间	0 ～ 65,535ms	当接收到的两个字符间的时间超过用户设定的字符间超时时间，接收中止。
帧超时时间使能	有效或无效，默认为无效	—
帧超时时间	0 ～ 65,535ms	当 RCV 指令执行时，并且通讯条件满足，从通讯串口开始接收时计时，当计时时间到，还没有接收完一帧，中止接收。

10.3.3 自由口指令

- **注意事项**

自由口指令 XMT 和 RCV 可以用来向指定的通讯口发送和接收数据，自由口指令的详细使用方法请参见 *6.13* 通讯指令。如果在某个端口上使用自由口指令，需要先在 MXProgrammer 软件的系统组态中选择自由口协议，并设置通讯参数，设置完毕后系统组态需下载到 PLC 中，并且运行后才能生效。

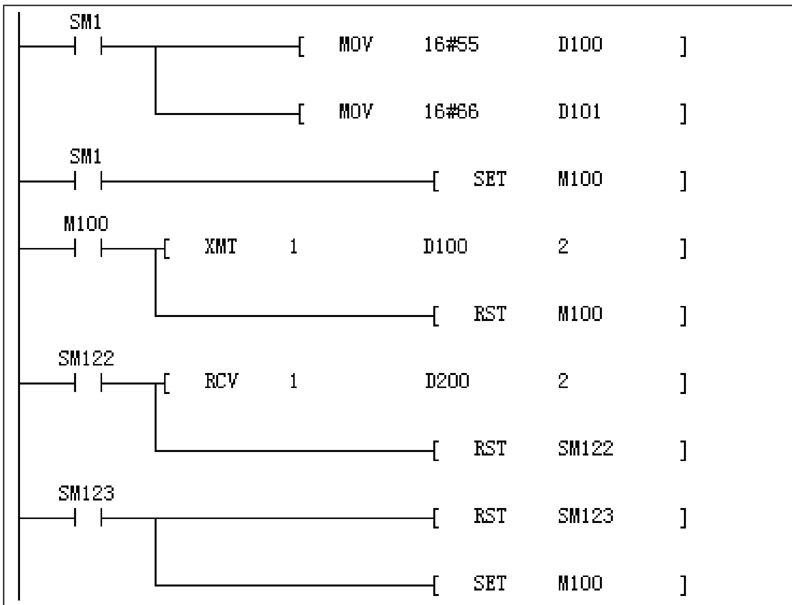
- **程序示例**

例程 1：通过通讯口 1 发送数据，然后接收数据，发送的数据为 2 个字节，接收的数据也为 2 个字节。

发送的数据	16#55	16#66
接收的数据	16#66	16#55

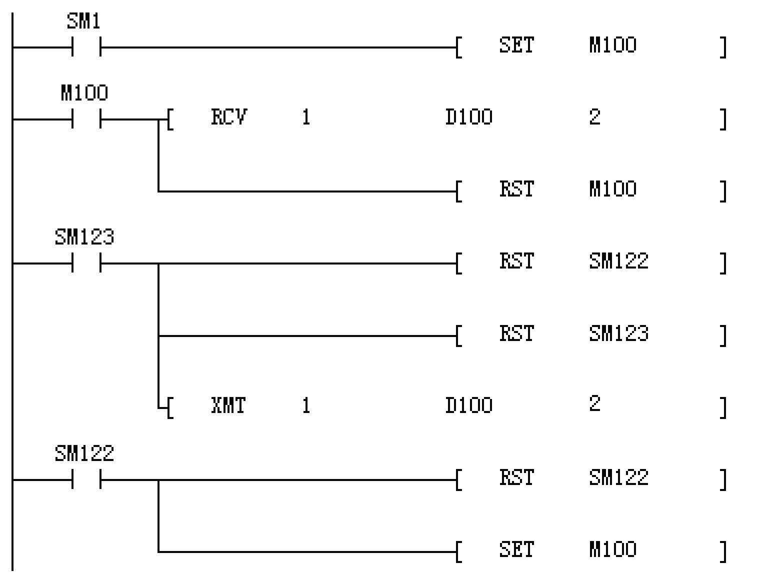
将接收到的数据保存到 D200 开始的单元序列中，每一个字节保存到一个 D 元件中，保存的方式如下表所示：

D200	D201
16#66	16#55



1. 首先在系统组态中将通讯口设置为自由口通讯，并设置波特率、奇偶校验位等参数。
2. 初始上电第一周期 SM1 导通，将待发送的数据保存到 D100 开始的通讯缓存区中，并利用 XMT 指令发送数据。
3. 发送完成时 SM122（发送完成标志位）置 ON，利用 SM122 上升沿，开始接收数据，接收的最大长度为 2，同时复位 SM122。
4. 接收完成时 SM123（接收完成标志位）置位，根据接收完成信息寄存器（SD125）的内容，进行相应操作。
5. 用 M100 作为启动一次发送和接收的使能位。

例程 2：通过通讯口 1 接收例程 1 发送的数据，然后把接收的数据发送回去。



1. 首先在系统组态中将通讯口设置为自由口通讯，并设置波特率、奇偶校验位等参数。
2. 初始上电第一周期 SM1 导通，开始接收数据，接收的最大长度为 2。
3. 接收完成时 SM123（接收完成标志位）置位，利用上升沿启动 XMT 指令发送数据，同时复位 SM122（发送完成标志位）和 SM123。
4. 发送完成时 SM122 置位后再复位，同时置位 M100。
5. 利用 M100 作为启动一次接收和发送的使能位。

10.4 CAN 自由口通讯协议

10.4.1 简介

CAN 自由口协议是用户定义数据文件格式的通讯方式，可由指令实现数据的发送和接收。只有在 PLC 位于 RUN（运行）模式时才能使用自由口通讯。

CAN 自由口的通讯指令包括 CANXMT（CANbus 单帧发送指令）、CANRCV（CANbus 单帧接收指令），为了方便用户的使用，提供了 CAN0 接收中断功能，只要使能了 CAN0 接收中断，PLC 会自动将每次接收的字节存储到相关的 SD 元件中。

10.4.2 CAN 自由口参数设置

在系统组态→硬件设置→选中通讯扩展卡（带有 CAN 接口），在协议设置栏选择自由口协议，然后在 CANbus 硬件设置栏和自由口时间设置栏设置相关的参数，如下图所示：

可配置内容如下：

选项	设置内容	注释
波特率（BPS）	20K，40K，50K，100K，125K，250K，500K 自定义①，默认为 125K	隔离扩展卡最高支持 250K 不隔离扩展卡最高支持 500K
发送超时时间	0 ～ 65,535ms，默认为 200ms	在执行 CANXMT 指令时，如果持续发送的时间超过此设置，则自动退出发送状态。
接收超时时间	0 ～ 65,535ms，默认为 200ms	在执行 CANRCV 指令时，如果持续接收的时间超过此设置，则自动退出接收状态。

①自定义波特率的设置。

BRP（发送段）：0 ～ 1023　　SEG1（时间段 1，定义采样点的位置）：0 ～ 15

SEG2（时间段 2，定义发送点的位置）：0 ～ 7　　波特率 $=\dfrac{18000}{(BRP+1)\times(SEG1+SEG2+3)}$ KHz

10.4.3 CAN 自由口指令

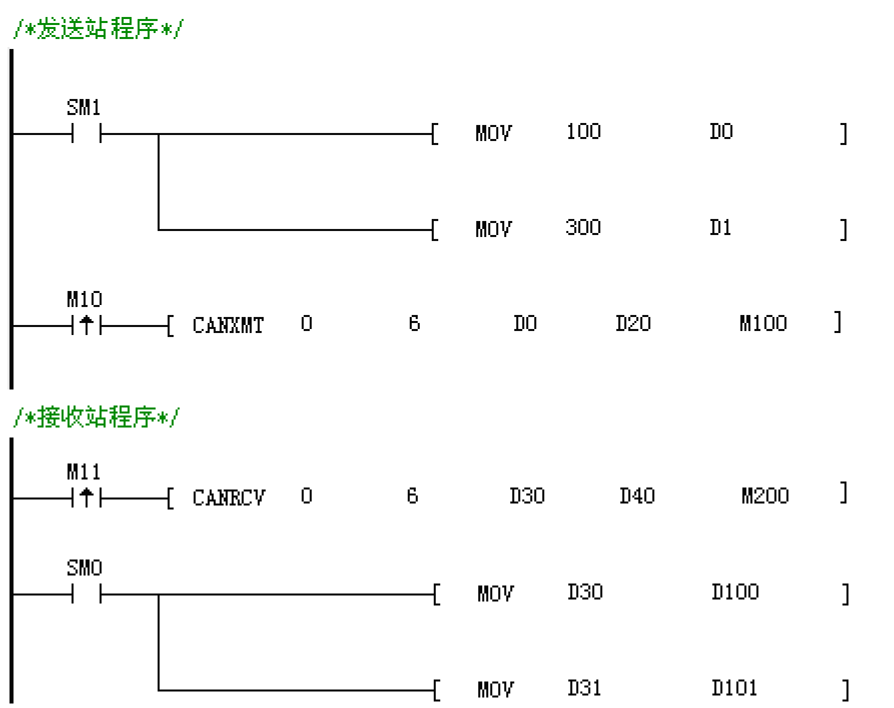
● **注意事项**

CAN 自由口指令 CANXMT 和 CANRCV 可以用来向指定的通讯口发送和接收数据，也可用 CAN0 接收中断来接收数据，CAN 自由口指令的详细使用方法请参见 *6.13* 通讯指令。

如果在某个端口上使用自由口指令，需要先在 MXProgrammer 软件的系统组态中选择自由口协议，并设置通讯参数，设置完毕后下载系统组态到 PLC 中，并且运行后才能生效。

● **程序示例**

例程 1：通过 CANXMT 指令发送数据，然后通过 CANRCV 指令接收数据。



首先在系统组态中将通讯口 CAN0 设置为自由口通讯，并设置波特率等参数。

发送站：

1. 初始上电第一周期 SM1 导通，给 D0，D1 赋值。
2. 利用 M10 上升沿启动 CANXMT 指令发送数据。

接收站：

利用 M11 上升沿启动 CANRCV 指令接收数据，并将接收到的数据存放到 D100，D101 中。

具体内容请参见 *6.13* 通讯指令。

例程 2：通过 CANXMT 指令发送数据，然后通过开通 CAN0 接收中断接收数据。

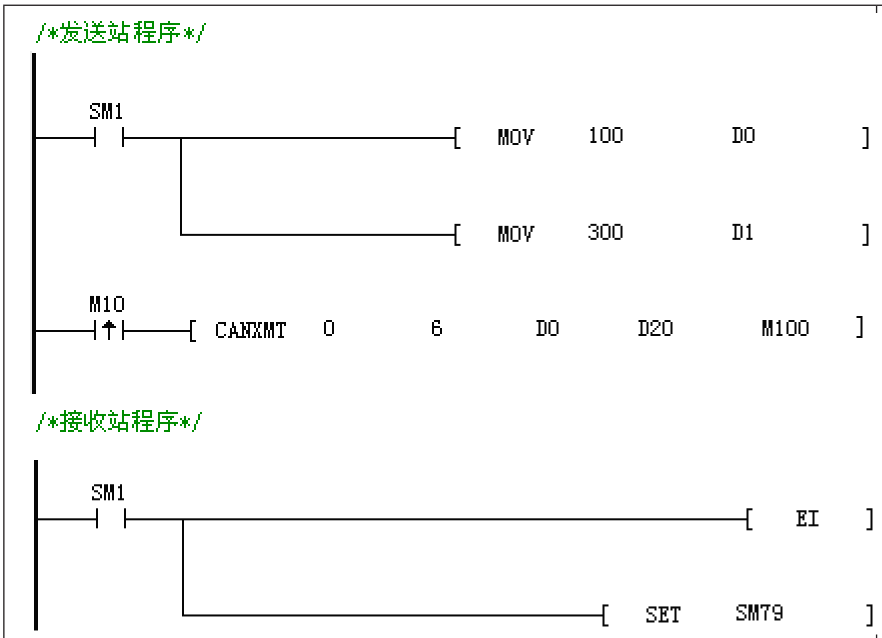
接收站可以通过开通 CAN0 接收中断来接收数据，接收的数据存放在 SD240 开始的单元中。与 CAN0 接收中断相关的 SM、SD 元件见下表所示：

表 1 SM 元件

地址	名称	动作与功能
SM79	CAN0 接收中断使能标志	置 ON 时，使能 CAN 0 接收中断

表 2 SD 元件

地址	名称
SD240	CAN0 接收中断当前接收帧 ID（低位）
SD241	CAN0 接收中断当前接收帧 ID（高位）
SD242	CAN0 接收中断当前接收帧长度
SD243	CAN0 接收中断当前接收帧第 1 个字符
SD244	CAN0 接收中断当前接收帧第 2 个字符
SD245	CAN0 接收中断当前接收帧第 3 个字符
SD246	CAN0 接收中断当前接收帧第 4 个字符
SD247	CAN0 接收中断当前接收帧第 5 个字符
SD248	CAN0 接收中断当前接收帧第 6 个字符
SD249	CAN0 接收中断当前接收帧第 7 个字符
SD250	CAN0 接收中断当前接收帧第 8 个字符



首先在系统组态中将通讯口 CAN0 设置为自由口通讯，并设置波特率等参数。

发送站：

1．初始上电第一周期 SM1 导通，给 D0，D1 赋值。

2．利用 M10 上升沿启动 CANXMT 指令发送数据。

接收站：

利用 CAN0 接收中断接收数据，并将接收到的数据存放到设定 SD240 开始的单元中。具体内容请参见 *6.13* 通讯指令。

10.5 Modbus 通讯协议

10.5.1 简介

Modbus 协议是应用于工业自动化领域设备之间传递信息的一种通用通讯协议，是一种工业标准的协议。本节主要介绍了 Modbus 协议在 MX 系列小型控制器中的应用。通讯采用主从方式：由主站发起请求，从站执行请求并且应答。同一个网络内从站通过设置地址加以区分，PLC 内部组网时，最多允许 30 个从站，地址范围为 1 ～ 31。PLC 作为主站与其他设备（做从站）组网，没有地址范围限制，但也最多允许 30 个从站。Modbus 协议支持广播方式，广播地址为 00。

10.5.2 链路特性

1．物理层：RS232、RS485

2．链路层：异步传输

（1）数据位：8 位

（2）传输速率：1200，2400，4800，9600，19200，38400，57600，115200，默认为 19200bps

（3）校验方式：偶校验，奇校验，无校验，默认为偶校验

（4）停止位： 1 位，2 位，默认为 1 位

3．组网配置：最多 31 个设备，地址范围 1 ～ 31，支持广播。

10.5.3 RTU 传输模式

帧格式：

从站地址	功能码	数据	CRC 高字节	CRC 低字节

当信息在标准的 Modbus 系列网络传输时，每个字符或字节以如下方式发送（从左到右）：

最低有效位 ... 最高有效位

采用 11 位传输，必须是 8 位数据位，位的序列如下：

有奇偶校验

起始位	1	2	3	4	5	6	7	8	奇偶位	停止位

无奇偶校验

起始位	1	2	3	4	5	6	7	8	停止位	停止位

☞ 注意

1．字符间间隔应该少于 1.5 个字符时间。如果在帧完成之前两个字符间有超过 1.5 个字符时间间隔，认为帧错误，停止接收，清缓冲，重新启动接收。

2．没有帧头和帧尾，帧间间隔至少为 3.5 个字符时。

3．使用 CRC16 校验。

4．RTU 帧的最大帧长度是 256 个字节。

5．字符间隔时间计算

字符的时间与具体的通信波特率有关，若波特率为 19200 时计算方法如下：

1.5 个字符间隔 = 1/19200 *11*1.5*1000=0.86ms，

3.5 个字符间隔 = 1/19200 *11*3.5*1000=2ms。

10.5.4 Modbus 支持的功能码及元件寻址方式

PLC 元件与 Modbus 通讯协议地址的对应关系以及功能码信息可以在 MXProgrammer 上位机软件的“工具→ MODBUS 工具”中查询。

Modbus 通讯协议中支持的功能码 01，02，03，04, 05，06，08，15，16。

1．读写元件功能码和与元件的对应关系

功能码	功能码名称	Modicon 数据地址	可操作元件类型	注释
01	读线圈	0:xxxx	X、Y、M、SM、S、T、C	读位
02	读离散量输入	1:xxxx	X	读位
03	读寄存器	4:xxxx	D、SD、Z、T、C、U	读字
04	读寄存器	3:xxxx	D、SD、Z、T、C、U	读字
05	写单个线圈	0:xxxx	X、Y、M、SM、S、T、C	写位
06	写单个寄存器	4:xxxx	D、SD、Z、T、C、U	写字
15	写多个线圈	0:xxxx	X、Y、M、SM、S、T、C	写多位
16	写多个寄存器	4:xxxx	D、SD、Z、T、C、U	写多字

功能码	功能码名称	功能描述
08	回送诊断校验	把诊断校验报文送从机，以对通信处理进行评鉴

☞ 注意

1．0 表示线圈，1 表示离散量输入，3、4 表示寄存器。0、1、3、4 并不具备物理上的意义，不参与实际的寻址。

2．xxxx 表示范围 1 ～ 9999。每一种元件类型都有独立的逻辑地址范围 1 ～ 9999（协议地址是从 0 开始的）。

2．PLC 元件与 Modbus 通讯协议地址的对应关系

元件	类型	物理元件	协议地址	支持的功能码	注释
X	位元件	X0 ～ X777	0000 ～ 0511	01、05、15、02	输入的状态，元件编号为八进制
Y	位元件	Y0 ～ Y777	0512 ～ 1023	01、05、15	输出的状态，元件编号为八进制
M	位元件	M0 ～ M4095	4096 ～ 8191	01、05、15	
SM	位元件	SM0 ～ SM511	1024 ～ 1535	01、05、15	
S	位元件	S0 ～ S1535	2560-4095	01、05、15	
T	位元件	T0 ～ T511	1536 ～ 2047	01、05、15	T 元件的状态
C	位元件	C0 ～ C511	2048 ～ 2559	01、05、15	C 元件的状态
D	字元件	D0 ～ D32767	0000 ～ 32767	03、04、06、16	
SD	字元件	SD0 ～ SD511	32768 ～ 33279	03、04、06、16	
Z	字元件	Z0 ～ Z255	33280 ～ 33535	03、04、06、16	
T	字元件	T0 ～ T511	33536 ～ 34047	03、04、06、16	T 元件的当前值
C	字元件	C0 ～ C199	34048 ～ 34247	03、04、06、16	C 元件（WORD）的当前值
C	双字元件	C200 ～ C511	34560 ～ 35182	03、04、16	C 元件（DWORD）的当前值
U	字元件	U0.0 ～ U0.199	40000 ～ 40199	03、04、06、16	
	字元件	U1.0 ～ U1.199	40200 ～ 40399	03、04、06、16	
	字元件	U2.0 ～ U2.199	40400 ～ 40599	03、04、06、16	
	字元件	U3.0 ～ U3.199	40600 ～ 40799	03、04、06、16	
	字元件	U4.0 ～ U4.199	40800 ～ 40999	03、04、06、16	
	字元件	U5.0 ～ U5.199	41000 ～ 41199	03、04、06、16	
	字元件	U6.0 ～ U6.199	41200 ～ 41399	03、04、06、16	
	字元件	U7.0 ～ U7.199	41400 ～ 41599	03、04、06、16	

☞ 注意

1．协议地址是在数据传输中使用的地址，协议地址与 Modicon 的数据的逻辑地址有对应关系，协议地址是从 0 开始，Modicon 的数据的逻辑地址是从 1 开始的，也就是说协议地址 +1=Modicon 的数据的逻辑地址，例如：D0 协议地址是 0，它对应的 Modicon 的数据的逻辑地址是 1。
2．扩展模块 Un.b 的协议地址计算公式：40000+b+200×n；n ≤ 8。
3．对于大多数 SM、SD 元件可能通过 Modbus 写入时，实际的值是写不进去的，但是 PLC 从站也会返回写入成功的信息，这是允许的。

10.5.5 Modbus 功能码叙述

1．读取线圈状态

读取从站的位元件的状态，不支持广播，最多支持 2000 个位元件。

请求帧

站号	功能码	协议地址		元件个数		校验码	
	01	H	L	H	L	H	L

响应帧

如果读取的个数不是 8 的倍数，剩下的位由 0 填充（由高位开始填充）。

站号	功能码	读取的元件个数 n（字节数）	读取的数据	••• •••	读取的数据	校验码	
	01H		No.1		No.n	H	L

B7	B6	B5	B4	B3	B2	B1	B0

2．读取寄存器

读取寄存器是读取从站的数据（字）寄存器值，不支持广播。

请求帧

站号	功能码	协议地址		元件个数		校验码	
	03/04	H	L	H	L	H	L

响应帧

站号	功能码	读取的元件个数n（字节数）	读取的数据		••• •••	读取的数据		校验码	
	03/04		H	L		H	L	H	L

3．强制（写）单线圈

强制（写）单个线圈是向从站写入位元件值，与编程协议的强制不同。允许广播（broadcast），即写入所有从站的相同元件。最多支持 1 个位元件。

请求帧

站号	功能码	协议地址		写入的元件值		校验码	
	05	H	L	H	L	H	L

注：写入元件的值：16#FF00（ON，1）；16#0000（OFF，0）。

响应帧

响应帧是请求帧的重复。

站号	功能码	协议地址		写入的元件值		校验码	
	05	H	L	H	L	H	L

4．写多个线圈

写多个线圈是向从站写入位元件值，与编程协议的强制不同。允许广播（broadcast），即写入所有从站的相同元件。

请求帧

站号	功能码	起始地址		写入位元件个数		占寄存器个数	写入元件值	••• •••	写入元件值	校验码	
	15	H	L	H	L					H	L

响应帧

响应帧是请求帧的重复。

站号	功能码	起始地址		写入的元件值		校验码	
	05	H	L	H	L	H	L

5．强制（写）单寄存器

强制（写）单个寄存器是向从站写入字元件值，与编程协议的强制不同。允许广播（broadcast），即写入所有从站的相同元件。最多支持 1 个字元件。

请求帧

站号	功能码	起始地址		写入的元件值		校验码	
	06	H	L	H	L	H	L

响应帧

响应帧是请求帧的重复。

站号	功能码	起始地址		写入的元件值		校验码	
	06	H	L	H	L	H	L

6．写多个寄存器

写多个寄存器是向从站写入字元件值，与编程协议的强制不同。允许广播（broadcast），即写入所有从站的相同元件。

请求帧

站号	功能码	起始地址		写入元件的个数		占寄存器个数	写入元件值		校验码	
	16	H	L	H	L		H	L	H	L

响应帧

响应帧是请求帧的重复。

站号	功能码	起始地址		写入的元件个数		校验码	
	16	H	L	H	L	H	L

10.5.6 Modbus 从站

Modbus 从站不主动发送任何报文，只有接收到对本地寻址的报文后才根据具体情况看是否响应主站。从站仅支持 Modbus 功能码 01，02，03，04，05，06，08，15，16，其余的响应均为“非法功能码”（除广播帧）。

10.5.7 元件的读写

除了功能码 08，MX2H 支持的其他功能码都是对元件读写操作的，原则上最多允许一帧读取 2000 个位元件，写 1968 个位元件，读取 125 个字元件，写 120 个字元件。但由于实际的协议地址对不同类型元件是分开的，不连续（例如 X377 的协议地址是 255，Y0 的协议地址是 512），因此在对元件的读写操作时，一次读取的元件只能是一种类型的元件，而读取元件的最多数目也与实际定义的该类型元件个数有关系，例如读取 Y 元件，Y0 ～ Y377（共 256 点），协议地址范围 512 ～ 767，对应的 Modicon 的数据的逻辑地址是 1 ～ 256，在读取 Y 元件时最多允许读取 256 个元件。

举例如下：

1．主站发送：05 01 02 00 00 20 10 24

05 地址，01 功能码，02 00 起始地址，00 20 读取元件个数，10 24 校验

从站应答：会返回正确的应答

2．主站发送：01 01 00 00 01 01 FC 5A

主站读取起始地址从 0000 的 01 01（257）个元件，超出了定义的 X 元件个数范围

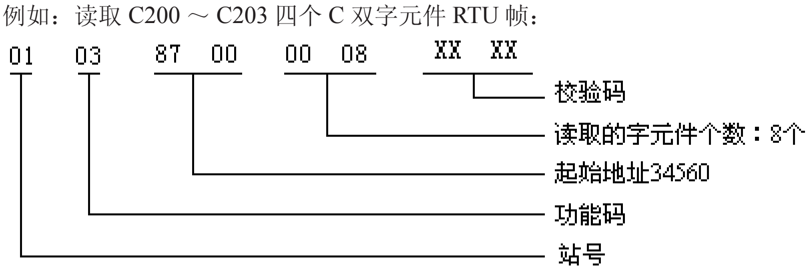
从站应答：01 81 03 00 51

从站应答是非法的数据值，原因是 257 大于 256，而 256 是最大允许的 X 元件个数

10.5.8 对双字元件的处理

C 元件的当前计数值为字元件或双字元件，C200 ～ C511 为双字元件。对 C200 ～ C511 的读写也是通过读写寄存器的功能码（03、16）来完成。每两个寄存器的地址对应一个 C 双字元件，读写时只能成对的读写寄存器。

例如：读取 C200 ～ C203 四个 C 双字元件 RTU 帧：



返回的数据中 34,560 / 34,561 两个地址表示 C200 的内容，34,560 为低 16 位，34,561 为高 16 位。

在读取双字元件时如果读取的开始地址不是偶数，会返回异常码非法地址，如果读取的寄存器个数不是偶数，会返回异常码非法数据。

举例如下：

主站发送：01 03 25 E5 00 04 5E F2

主站发送读取开始地址是 25 E5（十进制 9701）的四个字元件

从站响应：01 83 02 C0 F1

从站应答：非法的数据地址

主站发送：01 03 25 E4 00 05 CE F2

主站读取开始地址是 25 E4 的 5 个字元件

从站应答：01 83 03 01 31

从站返回非法的数据

10.5.9 对 LONGINT 的处理

一个 LONG INT 类型的数据可能存在于两个 D 元件内，例如：（D2，D1）存一个 LONG INT 型的数，D2 存储的是高 16 位，D1 存储的是低 16 位，当主站通过 Modbus 读取 LONG INT 数据时，读回数据后也应该按照上述存储原则重组 32 位数据。FLOAT 的存储原则等同于 LONG INT 的存储原则。

10.5.10 诊断功能码

诊断功能码用来提供测试主站和从站之间的通讯，或者从站的各种内部差错状态。支持的诊断子功能码如下表所示：

功能码	子功能码	子功能码名称	功能码	子功能码	子功能码名称
08	00	返回询问数据	08	12	返回总线通讯错误计数
08	01	重新启动通讯选项	08	13	返回总线异常错误计数
08	04	强制只听模式	08	14	返回从站报文计数
08	10	清除计数器	08	15	返回从站无响应计数
08	11	返回总线报文计数	08	18	返回总线字符超限计数

10.5.11 异常码

当主站发送的命令，在正常的响应中，从站在数据域中返回数据或统计值。在异常的响应中，服务器在数据域中返回异常码，异常码如下表

异常代码	异常代码意义
16#01	非法功能码
16#02	非法寄存器地址
16#03	非法的数据

另外，从站在以下几种情况接收到数据不会返回响应报文

1．广播帧中有错误，例如数据错误，地址错误等。

2．字符超限不会返回，例如 RTU 帧大于 256 个字节。

3．当使用 RTU 传输模式时，字符间的间隔时间超时，相当于收到错误帧，不会返回。

4．从站在只听模式下不会返回。

☞ 注意

读取站具有强制元件，读取的只是程序运行的值，可能与强制值不符。

10.5.12 Modbus 参数设置

在 Modbus 通讯协议操作数界面中有默认值按钮，默认值是 Modbus 通讯协议推荐的通讯设置。参数设置如下表所示。

选项	设置内容
站号	1 ～ 31
波特率（BPS）	1200，2400，4800，9600，19200，38400，57600，115200
数据位	设定 8

选项	设置内容
奇偶校验位	设定为无校验、奇校验、偶校验
停止位	设定 1 或 2，奇、偶校验时设定为 1，无校验时设定为 1, 2
Modbus 主 / 从	可设为主站或从站，通讯口 1 可设为主站或从站，通讯口 0 不可设为主站
传输模式	RTU 模式
主模式超时时间	主站等待从站响应的超时时间
注：当操作数在系统组态中设定并下载后，不是立即有效，而是必须运行一次，才能生效。	

10.5.13 Modbus 指令

Modbus 的主站可以通过系统提供的 Modbus 指令发送 Modbus 数据帧并接收回复，Modbus 指令的详细使用方法请参见 *6.13* 通讯指令。

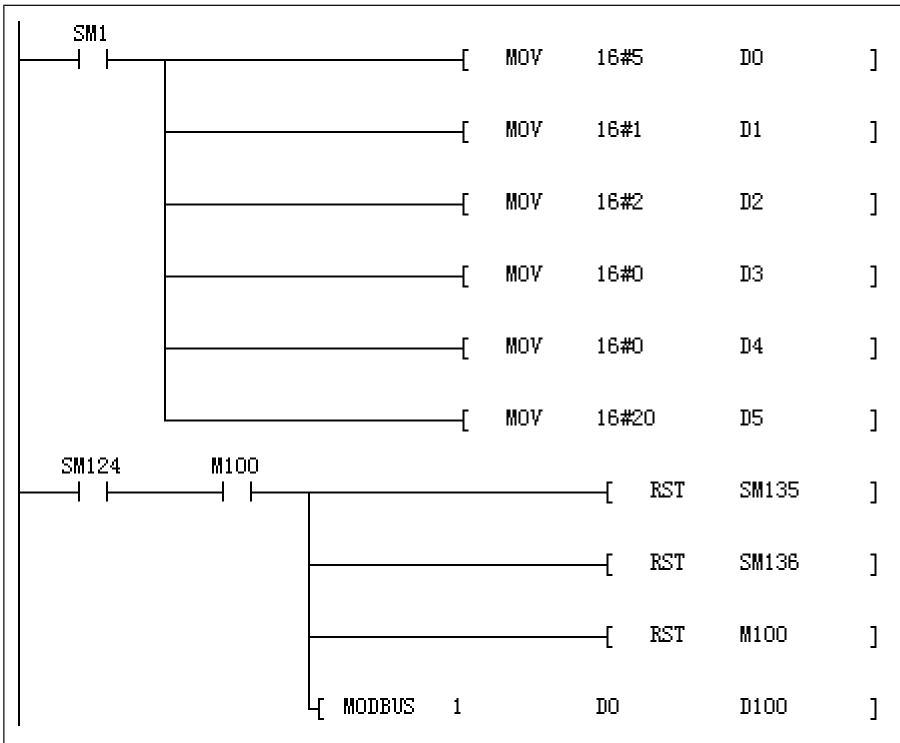
当设置 PLC 作为主站时，在系统组态中设置 Modbus 参数，存在主模式的超时时间，为了能够保证接收数据的正确性，设置时应该确保这个时间比 Modbus 从站的一个循环扫描周期要长，并且有余量。例如 MX2H 做从站，如果 MX2H 的一个扫描周期为 100ms，那么主站的主模式超时时间应该在 100ms 以上，设置 150ms 比较适合。

示例 1：Modbus 指令读线圈

Modbus 的主站和从站都为 MX2H 系列的 PLC，读取 5 号站的位元件状态，主站读取从站协议地址为 512 ～ 543 的位元件值，接收到的数据存储位置从 D100 开始，D100 保存地址，D101 保存功能码，D102 保存寄存器个数，从 D103 开始保存读取到的位元件的值。读取数据如下表格所示：

543	542	541	540	539	538	537	536	535	534	533	532	531	530	529	528
0	1	0	1	1	1	1	0	1	0	1	1	0	1	0	0
D106								D105							
527	526	525	524	523	522	521	520	519	518	517	516	515	514	513	512
1	1	1	1	0	1	1	0	0	1	0	0	1	1	0	1
D104								D103							

当读取的位元件个数不是 8 的倍数的时候，不足的部分高位补零。

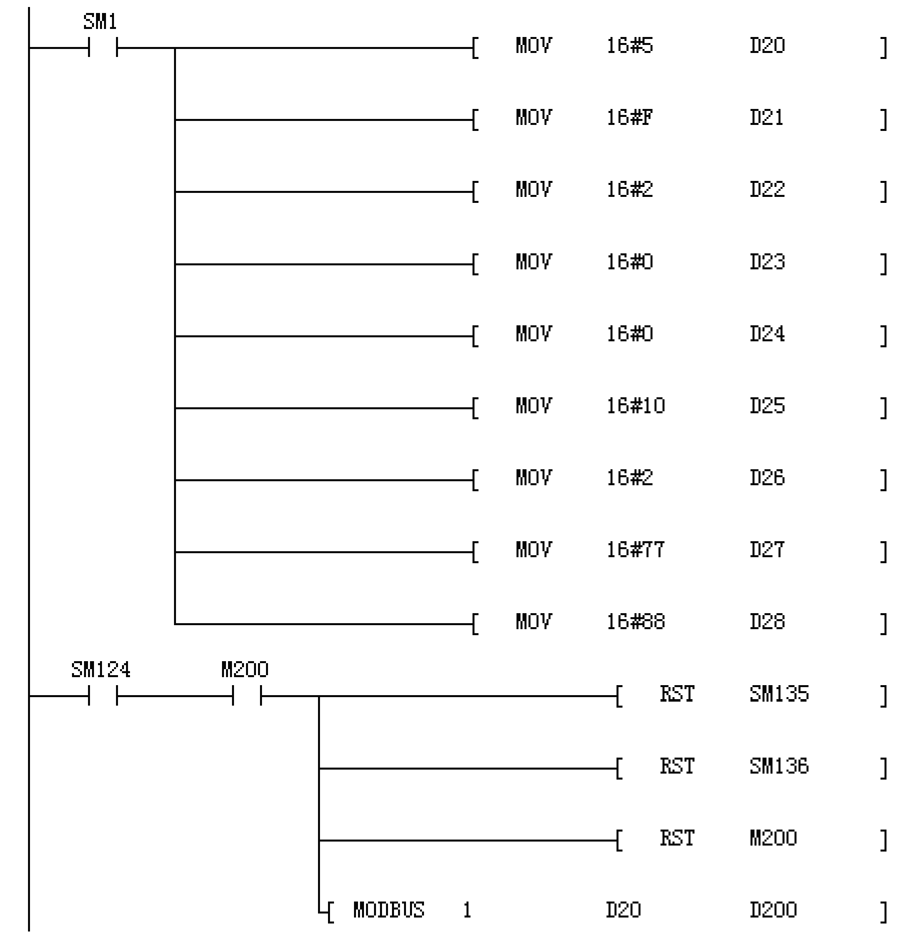


1. 指定访问的从站地址为 5（保存到 D0）。
2. 指定功能码为 1（保存到 D1 中）。
3. 要读取的位元件起始协议地址为 512（分高低字节保存到 D2/D3 中）。
4. 要读取的位元件个数 32（分高低字节保存到 D4/D5 中）。
5. 接收的数据保存在 D100 开始的单元序列中（只低字节存储）。
6. M100 是控制标志，SM124 是 PORT1 的空闲标志。

示例 2：Modbus 指令写线圈

Modbus 的主站和从站都为 MX2H 系列的 PLC，写 5 号站协议地址 512 ～ 527 的位元件。

发送数据起始地址由 D20 开始，D20 存放从站地址，D21 存放功能码，D22/D23 存放元件的协议地址，D24/D25 存放元件个数，D26 存放元件所占寄存器个数，D27/D28 存放元件的状态。

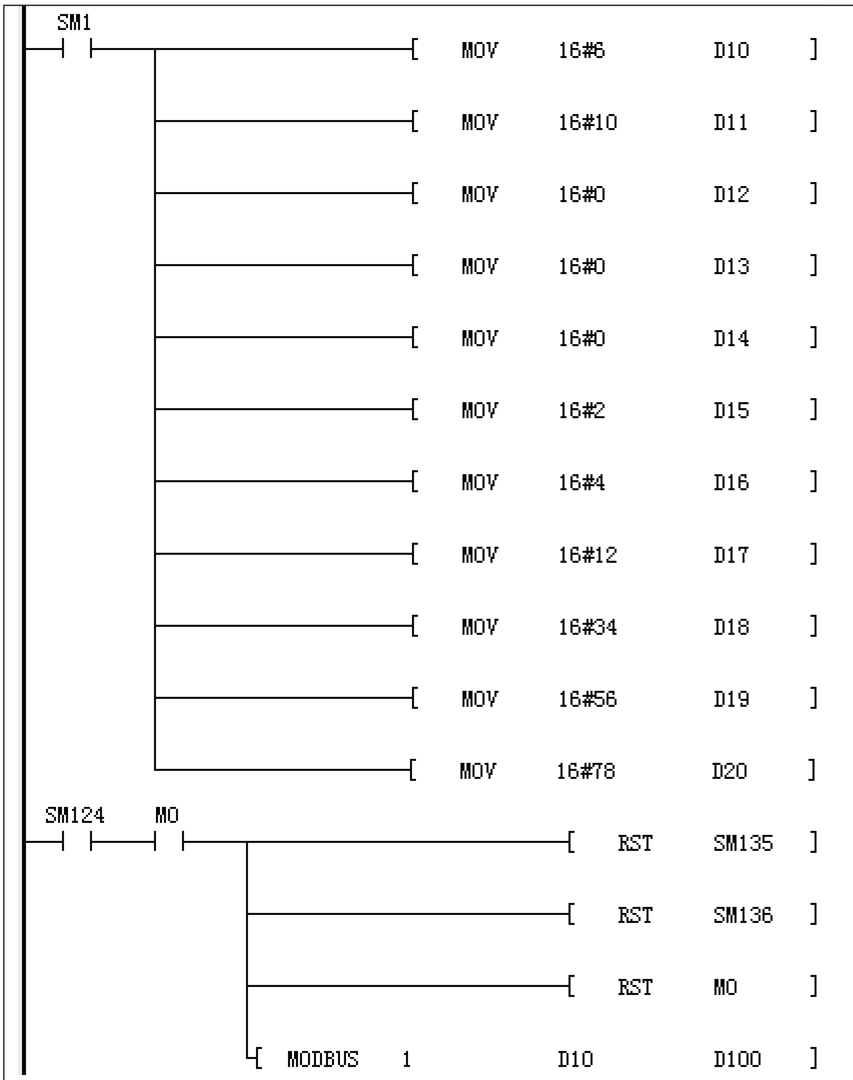


1. 指定访问的从站地址为 5（保存到 D20）。
2. 指定功能码为 15（保存到 D21）。
3. 要写的位元件的协议起始地址为 512（分高低字节保存到 D22/D23 中）。
4. 要写的位元件个数 16（分高低字节保存到 D24/D25 中）。
5. 要写的位元件占寄存器个数（保存到 D26 中）。
6. 要写的位元件状态（保存到 D27/D28 中的低字节）。
7. 返回的数据保存在 D200 开始的单元序列中。
8. M200 为指令控制标志，SM124 是 PORT1 的空闲标志。

示例 3：Modbus 指令写寄存器

Modbus 的主站和从站都为 MX2H 系列的 PLC，写 6 号站协议地址为 0000 ～ 0001 的寄存器（D0 ～ D1）。发送数据起始地址从 D10 开始，D10 存放从站地址，D11 存放功能码，D12/D13 存放寄存器的协议地址，D14/D15 存放待写入的寄存器个数，D16 存放要写的字元件所占寄存器个数，D17 ～ D20 存放写入的数据。从站接收数据的存储方式如下：

存放写入数据的地址	D17	D18	D19	D20
存放接收数据的地址	D0 B15 ～ B8	D0 B7 ～ B0	D1 B15 ～ B8	D1 B7 ～ B0

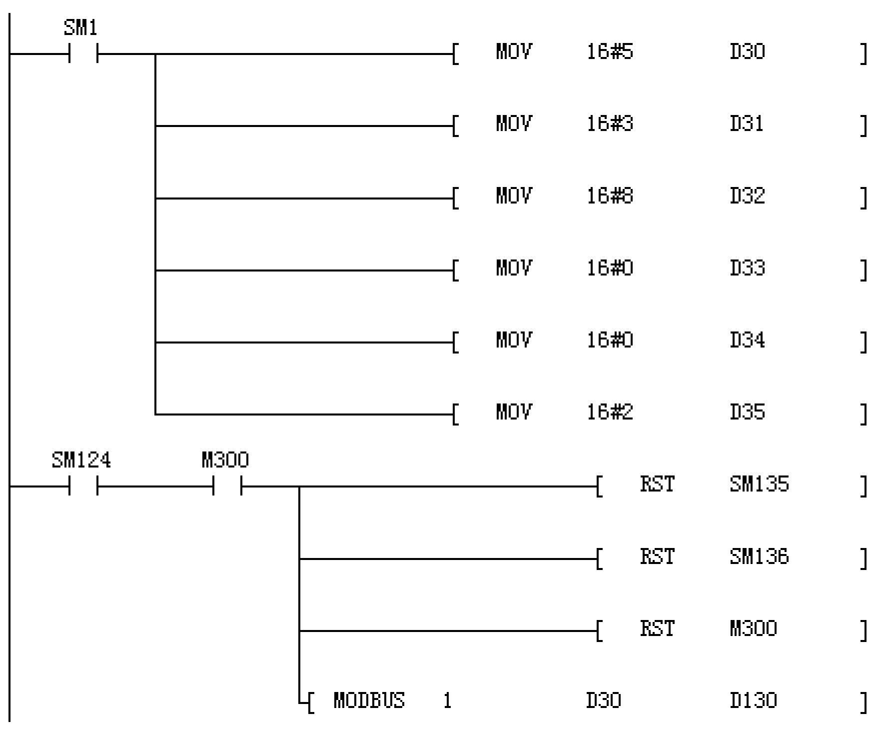


1. 访问的从站的地址为 6（保存到 D10）。
2. 指定功能码 16（保存到 D11）。
3. 要写入的寄存器起始地址为 0000（D0 的协议地址，分高低字节保存到 D12/D13 中）。
4. 要写入的寄存器个数为 2（分高低字节保存到 D14/D15 中）。
5. 写入字元件所占寄存器个数（保存到 D16 中）。
6. D17 ～ D20 存放待写入的寄存器内容（低字节有效）。
7. M0 是指令控制标志，SM124 是 PORT1 的空闲标志。

示例 4：Modbus 指令读寄存器

Modbus 的主站和从站都为 MX2H 系列的 PLC，读取 5 号站协议地址为 34048 ～ 34049（C0 ～ C1）的字元件值。读取到的数据如下，接收的帧是由 D130 开始，D130 保存地址，D131 保存功能码，D132 保存寄存器个数，D133 开始保存读取到的寄存器值。

读取的字元件	C0 B15 ～ B8	C0 B7 ～ B0	C1 B15 ～ B8	C1 B7 ～ B0
存放接收数据的地址	D133	D134	D135	D136



1. 访问的从站地址为 5（保存到 D30）。
2. 指定功能码 3（保存到 D31）。
3. 要读取的寄存器起始地址为 2048（C0 的协议地址，分高低字节保存到 D32/D33 中）。
4. 要读取的寄存器个数为 2（分高低字节保存到 D34/D35 中）。
5. 接收的数据保存到 D130 开始的单元序列中（只低字节存储）。
6. M300 是指令控制标志，SM124 是 PORT1 的空闲标志。

☞ 注意

1. 当使用逻辑地址对 MX2H 系列 PLC 的位元件寻址时，逻辑地址 1 即为协议地址 0。
2. 当本次通讯发生错误后，并不影响下一次通讯，即在一个程序中有两个 Modbus 指令发送数据，当第一个 Modbus 指令通讯失败，并且有错误代码时并不影响第二个 Modbus 指令发送数据。
3. 当从站处于只听模式时，主站对从站发送请求是不会产生响应的，因此主站不会置完成标志，会置出错标志，因此要求用户应该记得在 Modbus 组网的时候，哪些从站是处于只听模式，以便于用户能够判断 Modbus 指令发送出错不是由于此原因产生的。

10.5.14 MODRD 指令（Modbus 读指令）

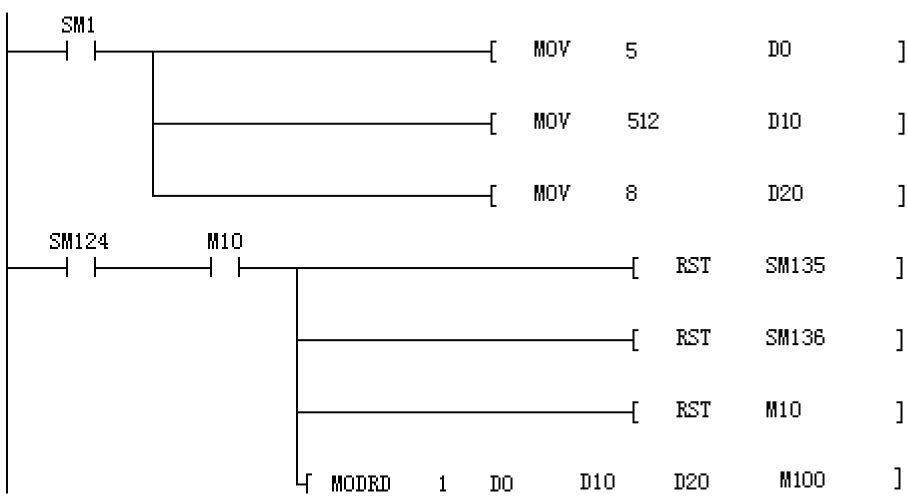
Modbus 的主站可以通过系统提供的 MODRD 指令读取从站的元件数据，MODRD 指令的详细使用方法请参见 *6.13* 通讯指令。

当设置 PLC 作为主站时，在系统组态中设置 Modbus 参数，存在主模式的超时时间，为了能够保证接收数据的正确性，设置时应该确保这个时间比 Modbus 从站的一个循环扫描周期要长，并且有余量。例如 MX2H 做从站，如果 MX2H 的一个扫描周期为 100ms，那么主站的主模式超时时间应该在 100ms 以上，设置 150ms 比较适合。

示例 1：MODRD 指令读位元件

Modbus 的主站和从站都为 MX2H 系列的 PLC，读取 5 号站的位元件的数据，主站读取从站协议地址为 512 ～ 519 的位元件，接收到的数据存储位置从 M100 开始，D0 保存地址，D2 保存读取元件个数，从 M100 开始保存读取到的位元件的值。读取数据如下表格所示：

要读取的元件	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
对应协议地址	519	518	517	516	515	514	513	512
元件中的数据	0	1	0	0	1	1	0	1
对应接收元件	M107	M106	M105	M104	M103	M102	M101	M100



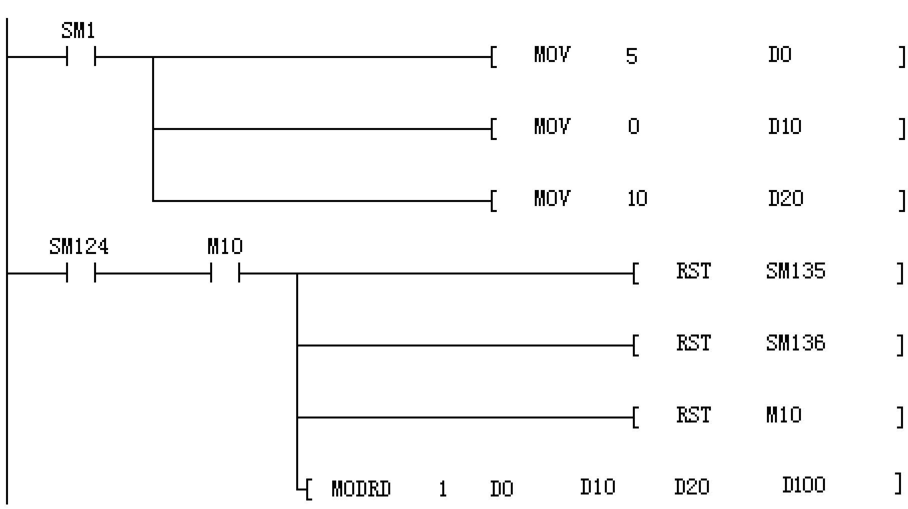
1. 指定访问的从站地址为 5（保存到 D0）。
2. 要读取的元件起始协议地址为 512 （保存到 D10 中）。
3. 要读取的位元件个数为 8（保存到 D20 中）。
4. 接收的数据保存在 M100 开始的单元序列中（该操作数为 M 或是 D，决定了要读取的是位元件还是字元件）。
5. M10 是控制标志，SM124 是 PORT1 的空闲标志。

示例 2：MODRD 指令读字元件

Modbus 的主站和从站都为 MX2H 系列的 PLC，读取 5 号站的字元件数据，主站读取从站协议地址为 0 ～ 9 的字元件，接收到的数据存储位置从 D100 开始，D0 保存地址，D2 保存读取元件个数，从 D100 开始保存读取到的字元件的值。

读取数据如下表格所示：

要读取的元件	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
对应协议地址	9	8	7	6	5	4	3	2	1	0
元件中的数据	556	45	145	456	526	89	568	4858	545	1203
对应接收元件	D109	D108	D107	D106	D105	D104	D103	D102	D101	D100



1. 指定访问的从站地址为 5（保存到 D0）。
2. 要读取的元件起始协议地址为 0 （保存到 D10 中）。
3. 要读取的位元件个数为 10（保存到 D20 中）。
4. 接收的数据保存在 D100 开始的单元序列中（该操作数为 M 或是 D，决定了要读取的是位元件还是字元件）。
5. M10 是控制标志，SM124 是 PORT1 的空闲标志。

10.5.15 MODWR 指令（Modbus 写指令）

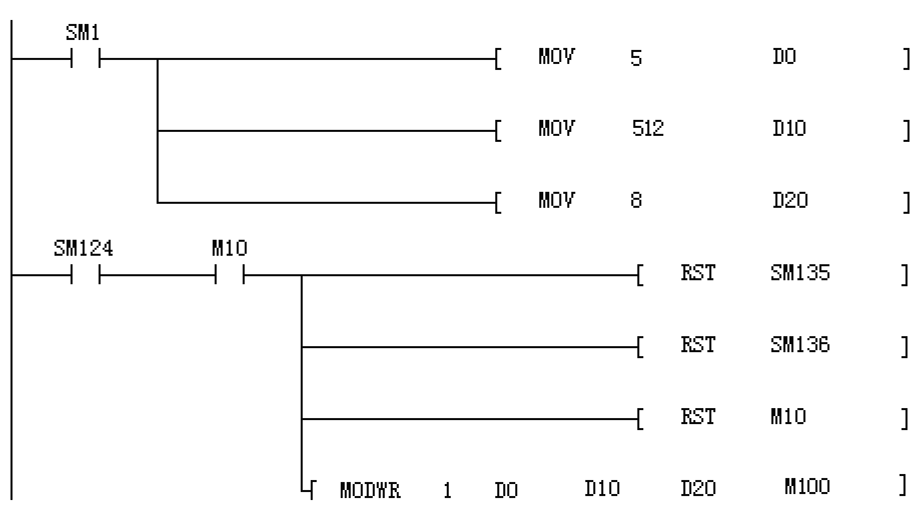
Modbus 的主站可以通过系统提供的 MODWR 指令向从站的元件中写入数据，MODWR 指令的详细使用方法请参见 *6.13* 通讯指令。

当设置 PLC 作为主站时，在系统组态中设置 Modbus 参数，存在主模式的超时时间，为了能够保证接收数据的正确性，设置时应该确保这个时间比 Modbus 从站的一个循环扫描周期要长，并且有余量。例如 MX2H 做从站，如果 MX2H 的一个扫描周期为 100ms，那么主站的主模式超时时间应该在 100ms 以上，设置 150ms 比较适合。

示例 1：MODWR 指令写位元件

Modbus 的主站和从站都为 MX2H 系列的 PLC，向 5 号站的位元件写入数值，主站写从站协议地址为 0 ～ 9 的位元件值，要写的数据存储位置从 M100 开始，D0 保存地址，D2 保存写元件个数，从 M100 开始保存要写的位元件的值。写数据如下表格所示：

要写的元件	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
对应协议地址	519	518	517	516	515	514	513	512
元件中的数值	1	1	0	0	1	0	1	1
要写的数据	M107	M106	M105	M104	M103	M102	M101	M100

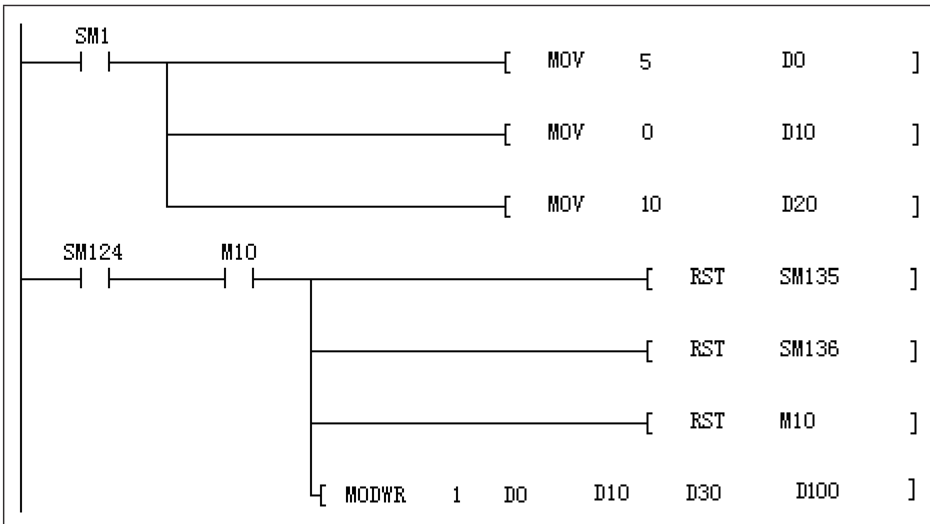


1. 指定访问的从站地址为 5（保存到 D0）。
2. 要写的元件起始协议地址为 512（保存到 D10 中）。
3. 要写的位元件个数 8（保存到 D20 中）。
4. 要写的数据保存在 M100 开始的单元序列中（该值为 M 或是 D，决定了要写的是位元件还是字元件）。
5. M10 是控制标志，SM124 是 PORT1 的空闲标志。

示例 2：MODWR 指令写字元件

Modbus 的主站和从站都为 MX2H 系列的 PLC，向 5 号站的字元件中写入数据，主站写从站协议地址为 0 ～ 9 的字元件值，要写的数据存储位置从 D100 开始，D0 保存地址，D2 保存写元件个数，从 D100 开始保存要写的字元件的值。写数据如下表格所示：

要写的元件	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
对应协议地址	9	8	7	6	5	4	3	2	1	0
元件中的数据	12	3	20	42	96	58	47	36	51	52
要写的数据	D109	D108	D107	D106	D105	D104	D103	D102	D101	D100



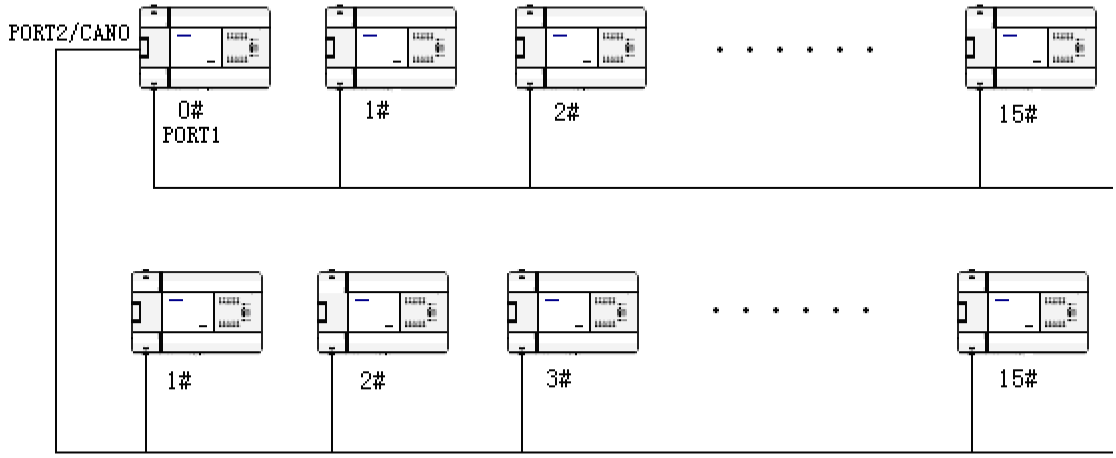
1. 指定访问的从站地址为 5（保存到 D0）。
2. 要写的元件起始协议地址为 0（保存到 D10 中）。
3. 要写的字元件个数为 10（保存到 D20 中）。
4. 要写的数据保存在 D100 开始的单元序列中（该值为 M 或是 D，决定了要写的是位元件还是字元件）。
5. M10 是控制标志，SM124 是 PORT1 的空闲标志。

10.6 MXLink 通讯协议

10.6.1 MXLink 简介

MXLink 是麦科电气技术有限公司自主研发的一种小型工业设备现场总线。MXLink 在物理层使用 RS485/CAN，PLC 可以通过 PORT1、PORT2/CAN0 接入。接入 MXLink 的 PLC 可以自动的相互交换部分 D 元件的数值，这使得对网络中其它 PLC 的元件访问，变得如同访问自身元件那样简单。MXLink 中，PLC 间的数据访问是完全对等的（N:N 通讯网络）。系统能够自动检测到新加入网络中的 PLC，任何一台 PLC 从网络中断开，其它 PLC 会继续交换数据。通过查看 MXLink 中任何一台 PLC 的相关 SM 元件的状态，都可以监测到整个网络的通讯情况。

MX2H/2V 可支持两层 MXLink 网络，PORT1（RS485 接口方式）可组成一层网络，最多 16 个站；PORT2（RS485 接口方式）或 CAN0（CAN 接口）可组成一层网络，最多 16 个站。 RS485 接口方式最高波特率 115.2K，CAN 接口方式最高波特率 500K。



MXLink 通讯示意图

10.6.2 MXLink 网络数据的传输形式

MXLink 的数据传输形式有两种：一种是持有令牌的 PLC 将自己的数据对其他 PLC 进行发送广播；另一种是事件触发（只有 CAN 网络支持）。

MXLink 一共存在两种报文：总站发放的令牌；各 PLC 对自己数据的广播。

令牌由主站统一发放，主站首先持有令牌，广播数据后，将令牌循环、依次发放到各从站。只有收到令牌的从站才能对其它 PLC(包括总站) 进行广播。如下图 1 ～图 5 表示了网络通信的主要过程。图中 1# 作主站。需要指出的是，一般情况下，默认 0# 为主站，1# 为备用主站（当主站通讯故障或者掉电的时候，切换为主站）。

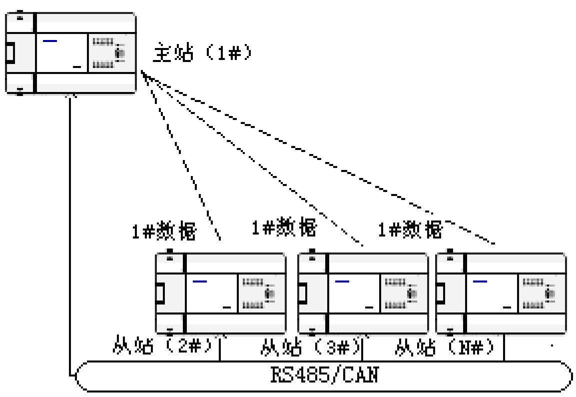


图 1 主站广播

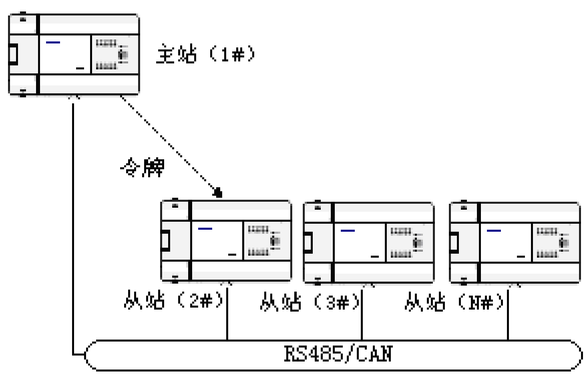


图 2 主站向 2# 从站发放令牌

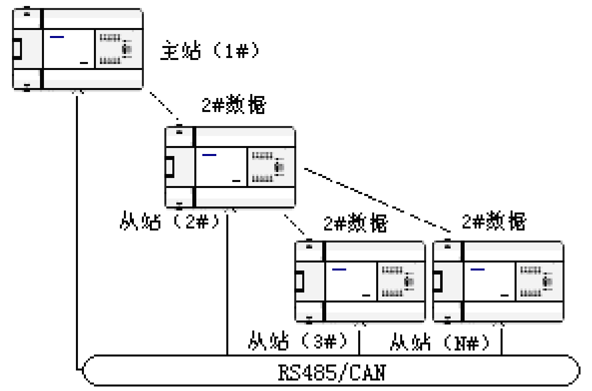


图 3 2# 从站广播

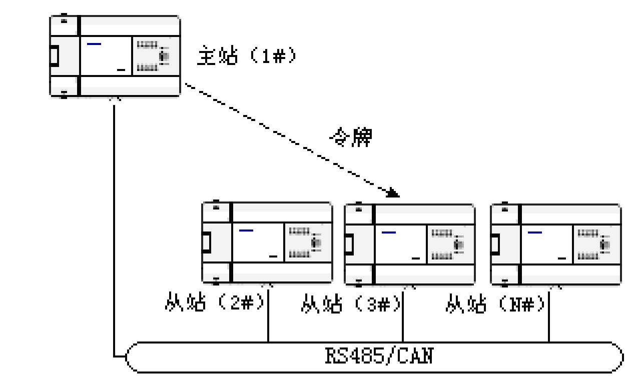


图 4 主站向 3# 从站发放令牌

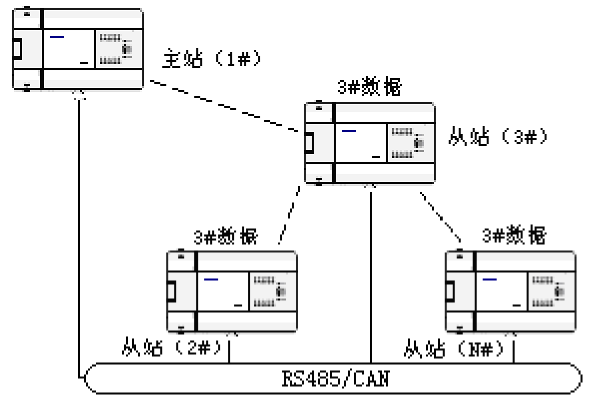


图 5 3# 从站广播

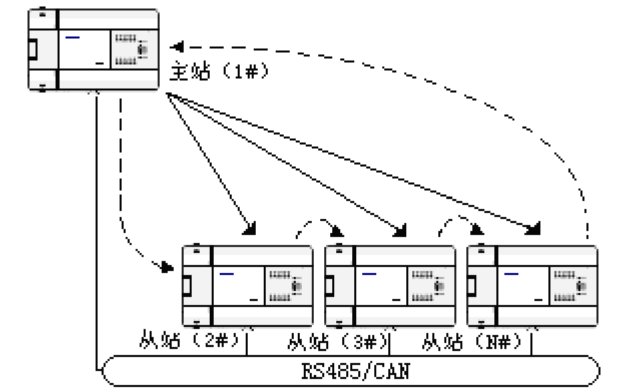


图 6 令牌的发放与流动顺序

图 6 显示了令牌的流动顺序。实线表示了实际的令牌发放过程，虚线表示了持有令牌并广播的站点的顺序。需要注意的是，令牌并不是由一个从站（例如 2#PLC）传递到另一个从站（例如 3#PLC）的，而是先由主站发放令牌到 2#PLC，然后再由主站发放到 3#PLC 的。

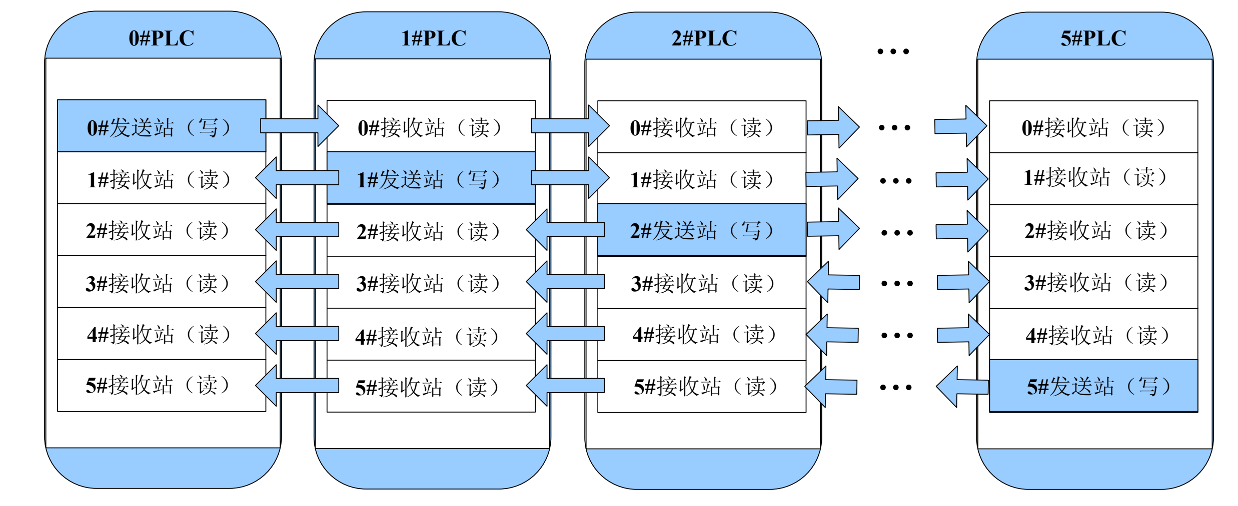
使用 SD145 可以更改该模块的站号，当模块由从站更改为主站时，共享元件的分配遵循之前系统组态配置的方式。

基于 CAN 的 MXLink 协议除了有以上的令牌流动方式外，还有一种机制是事件触发机制，具有抢先功能，即当某一个 PLC 共享区的元件有数据变化时，不管此时是否拥有令牌，都会立即抢先发送数据。

事件触发机制的实时性比令牌发送机制要高。

10.6.3 MXLink 的刷新模式

接入 MXLink 的多个 PLC 能够自动的互相交换网络中的部分 D 元件，这些 D 元件个数和编号可以通过 MXProgrammer 进行配置，这些元件称为“共享元件区”。一旦 PLC 使用了 MXLink，共享元件区的值将会自动的不停的刷新，从而使网络中每个 PLC 的共享元件区的值保持相等。



如上图所示，接入 MXLink 的每个 PLC 在这个共享元件区中，都有属于自己的一个可写的发送区，MXLink 将自动的将这个可写的发送区中的内容（特定编号的 D 元件的数值）广播给其它的 PLC，同时也接收其它 PLC 广播的内容，存入到对应的只读接收区中。

由于共享元件区的元件个数是固定的（总共 256 个 D 元件可共享），这些元件分配给多个 PLC。所以，接入网络的 PLC 数量越少，每个 PLC 分配的元件个数就越多：

MX2H/2V：PORT1 MXLink 可共享 D 元件范围 D7000 ～ D7255
PORT2 MXLink 可共享 D 元件范围 D7500 ～ D7755
CAN0 MXLink 可共享 D 元件范围 D7500 ～ D7755

MX1H：PORT1 MXLink 可共享 D 元件范围 D3000 ～ D3255
CAN0 MXLink 可共享 D 元件范围 D3500 ～ D3755

当网络中使用的 PLC 是不同型号时，其共享 D 元件的范围的偏移是相对的，例如 1#PLC（MX2H）共享数据区 D7000 ～ D7049，2#PLC（MX1H）共享数据区 D3000 ～ D3049，则 1#PLC 的 D7000 对应 2#PLC 的 D3000，1#PLC 的 D7001 对应 2#PLC 的 D3001，如此一一对应。

10.6.4 MXLink 的参数设置

例如 MX1H，在“系统组态→硬件配置”中激活相应的 MXLink 设置按钮，如下图所示：

如上图所示设置 MXLink 参数，站号的设置应该从 0 号开始，依次设置，不能将多台 PLC 设置为同一站号。0 号站是网络的启动和设置站点。网络中各站共享元件分配、附加延迟时间参数只需要对 0 号站进行设置，“控制起始～控制结束”为 0 号站的共享元件，“状态起始～状态结束”为所设站号的站点共享元件，0 号站的共享元件包括设置中所有行的“控制起始～控制结束”，且其它所有站都能接收到，其它站号的站点除了硬件设置要与 0 号一致外，只需设置自己的站号，如下图所示：

10.6.5 主站的确定

0 号站是默认的主站。只有 0 号站才能初始化和启动整个网络。MXLink 相关的设置，如刷新模式，附加延迟时间，重试次数等，只能通过 0 号站配置，0 号站在线修改相关配置并进行系统组态下载过程时，备用主站会接管网络。当 0 号站完成系统组态下载后，备用主站会将主站地位让给 0 号站。

网络中的主站策略：站号最小的站点充当主站。

第十一章　定位功能使用指南

本章详细讲述了定位的控制系统、MX 系列 PLC 定位功能、使用定位指令时的注意事项以及与定位指令相关的特殊元件等。

11.1 绝对位置系统

绝对位置系统利用伺服电机编码器的当前编码位置和总的运行圈数而获得伺服电机在行程上的绝对位置数据。根据这个原理，可建立如图 1 所示的绝对位置系统功能框图。

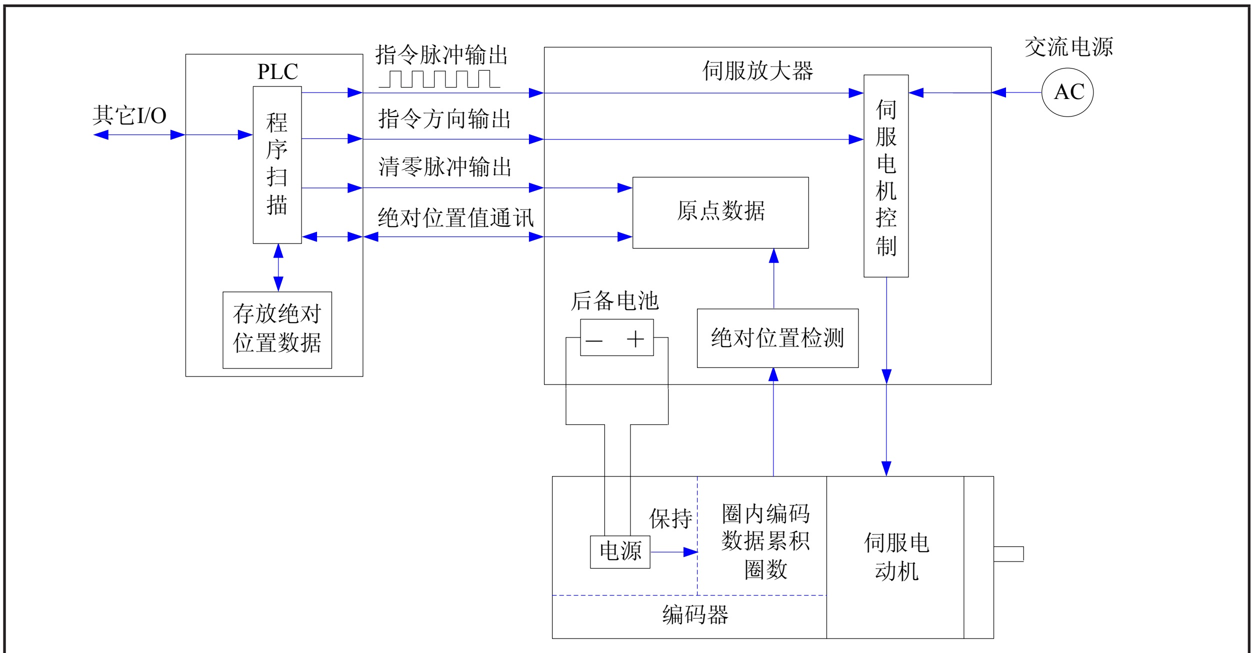


图 1 绝对位置系统功能框图

上图中，绝对位置系统的编码器当前编码位置和累积圈数计数器是可以掉电保持的，通过后备电池的供电来保持数据。即使在断电的情况下，伺服驱动器再次上电也能得出当前绝对位置数据。

PLC 可以通过通讯或其它方式从伺服驱动器获取绝对位置数据，用于确定行程的坐标位置。PLC 通过定位指令来控制伺服驱动器和电机，从而实现在行程上的精确定位，同时还可以自动对绝对位置数据进行增减刷新。由此，可构成一个基于绝对位置坐标的工作系统。

基于 MX 系列 PLC 定位指令构建的绝对位置系统机械示例图如图 2 所示。

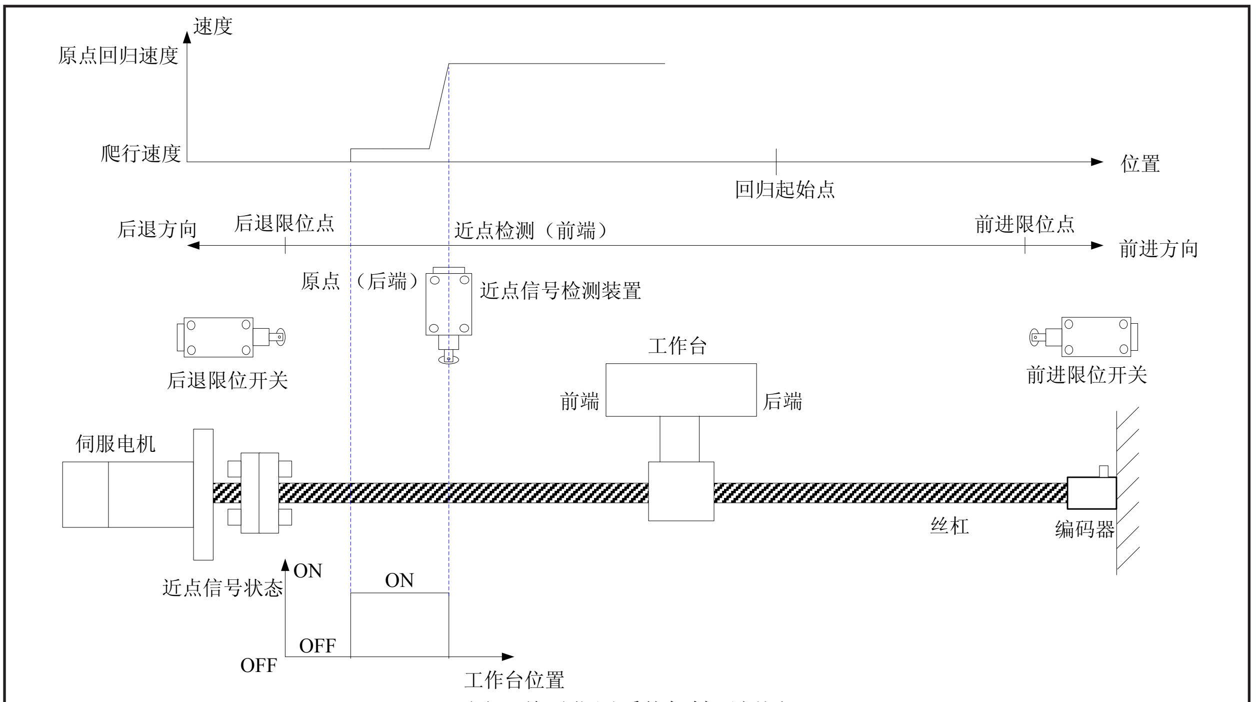


图 2 绝对位置系统机械示例图

该系统用伺服电机带动丝杠，驱动工作台运行。工作台在运行过程中的位置由绝对编码器检测。原点回归时，近点信号装置检测到工作台前端时（置位）伺服电机将减速为爬行速度；以近点信号装置检测到工作台后端处（复位）为原点位置到达信号，PLC 停止高速脉冲输出。必须设置前进限位开关和后退限位开关。必须从比近点检测装置更远的地方开始进行原点回归操作，因为原点回归指令 DZRN 没有自动搜索近点信号的功能。可以通过设计和编程，采用点动操作手工调整工作台位置。

11.2 MX 系列 PLC 定位功能概述

MX 系列 PLC 支持的定位功能包括简单的可变速脉冲输出、中断定位、（带 DOG）原点回归、当前值读取以及相对位置、绝对位置控制、直线插补和圆弧插补等功能。定位控制系统广泛用于对各种品牌的步进和伺服驱动器进行控制，可以通过相应伺服驱动器提供的方式获取绝对位置数据。

- **脉冲输出形式**

与伺服连接时，请将伺服驱动器输入信号设置为负逻辑方式，脉冲输出形式有如下的定义。

（1）第一种脉冲输出形式的定义如图 3 所示。例如：Y0 控制脉冲，Y1 控制方向。当 Y0 有脉冲输出时，若 Y1 为 ON，则伺服电机为正转；若 Y1 为 OFF，则伺服电机为反转。

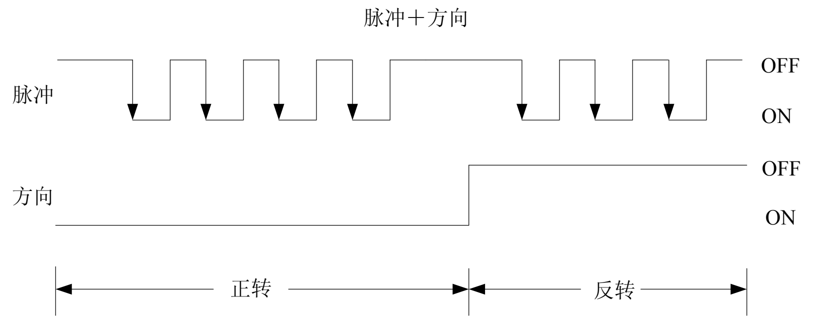


图 3 第一种脉冲输出形式

（2）第二种脉冲输出形式的定义如图 4 所示。例如：Y0 控制电机正转，Y1 控制电机反转。当 Y0 有脉冲发出时，则伺服电机为正转；当 Y1 有脉冲发出时，则伺服电机为反转。

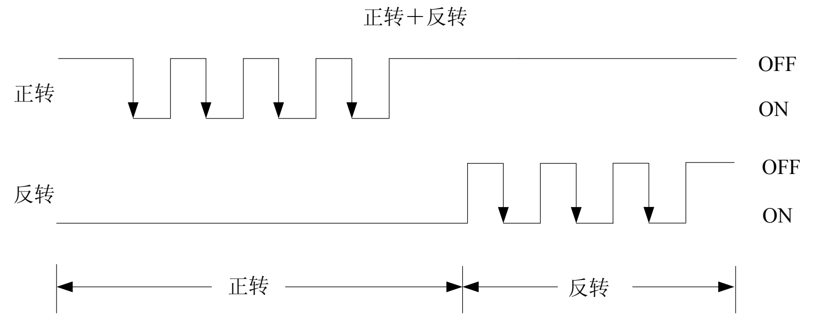


图 4 第二种脉冲输出形式

（3）第三种脉冲输出形式的定义如图 5 所示。例如：Y0 控制 A 相，Y1 控制 B 相。当 A 相超前于 B 相时，则伺服电机为正转；当 B 相超前于 A 相时，则伺服电机为反转。

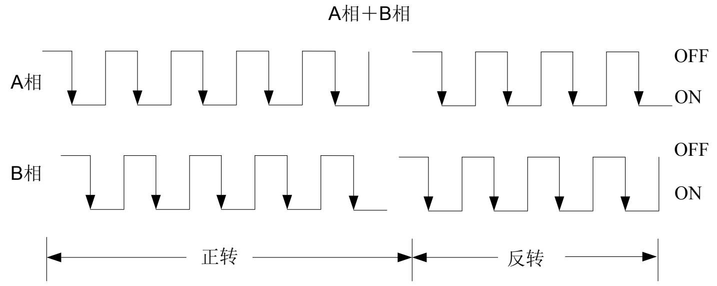


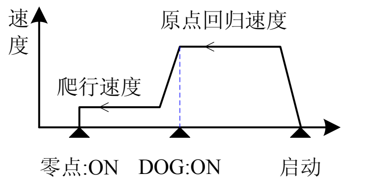
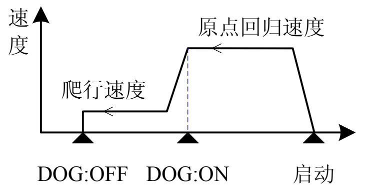
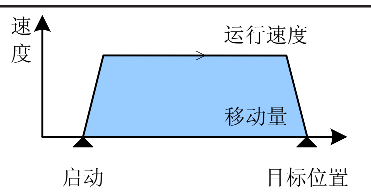
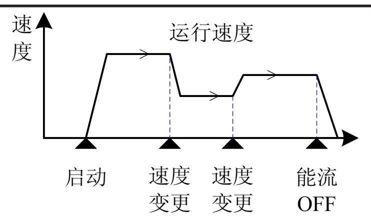
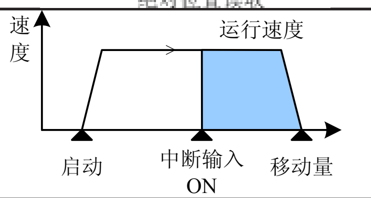
图 5 第三种脉冲输出形式

☞ 注意

高速 I/O 指令也可以控制脉冲输出，但仅能控制脉冲的输出，不能控制脉冲的方向。当使用高速 I/O 指令时，相应的 SD 元件按照正向作累加计算；希望驱动伺服正向运动时，需要用户编程将伺服方向信号置为 ON，反向运动时，置为 OFF。

- **MX 系列 PLC 支持的定位功能**

MX 系列 PLC 支持的定位功能如下表所示：

名称	动作	内容
DSZR		按照指定的原点回归速度动作，能够自动搜索 DOG 信号。检测到 DOG 后（DOG 的传感器为 ON），则减速为爬行速度。有零点信号输入时停止，完成原点回归。
DZRN		按照指定的原点回归速度动作，检测到 DOG 后（DOG 的传感器为 ON），则减速为爬行速度。DOG 传感器为 OFF 时停止，完成原点回归。
DDRVI		按照设定的运行速度动作，在目标位置停止，位置采用相对坐标。
DDRVA		按照设定的运行速度动作，在目标位置停止，位置采用绝对坐标。
PLSV		按照设定的运行速度动作。如果运行速度变化，则以新的速度运行；如果能流变为无效，则脉冲输出停止。有加减速动作的时候，在速度变更时，执行加减速。
DABS		从伺服驱动器中读出当前的绝对位置数据。
DDVIT		按照设定的运行速度动作，如果中断输入为 ON，则运行指定的脉冲数后，减速停止。

定位指令和高速指令根据设定，在高速端口输出可控脉冲，脉冲的输出与用户程序的扫描周期无关。在程序中，同时针对不同输出端口使用定位指令或高速指令能够在相应输出端口得到各自独立的高速脉冲输出。

11.3 使用定位指令时的注意事项

在定位指令或高速指令有效运行（包括输出完成）时，对同一端口的其它操作无效。只有在高速脉冲输出指令无效时，其它指令才有正确的输出。

在同一端口有多个定位指令或高速指令，先有效的指令占有输出端，后有效的指令不占有输出端。

- **晶体管输出**

必须使用具有晶体管输出的 MX 系列 PLC。

- **定位指令在编程中的要求**

定位指令可以在程序中反复使用，但是需要注意：

（1）一个高速脉冲输出点在任何时刻只能被一条定位指令或高速指令驱动。

（2）当一条定位指令的驱动触点断开后，必须经过 1 个或 1 个以上的 PLC 扫描周期后再接通驱动触点，才能再次执行。

- **同时采用高速指令和定位指令的要点**

从功能实现上，建议使用定位指令代替这些高速脉冲输出指令（PLSY，PLSR，PLS），定位指令可以完成对绝对位置 SD 元件的自动更新。

绝对位置 SD 元件可用来存储和更新定位指令使用的当前绝对位置。绝对位置 SD 元件值的自动增减是根据输出脉冲累计 SD 元件变化值，再加上调用定位指令时的运行方向来确定的，因此两者是联动关系。请不要在使用定位指令时，对脉冲累计 SD 元件进行写操作，否则可能会导致绝对位置 SD 元件数据混乱。

如果一定需要同时使用定位指令和其它高速脉冲输出指令（PLSY，PLSR，PLS），需要编写 PLC 程序，以便正确更新绝对位置寄存器绝对位置 SD 元件中的数据。

- **定位指令实际输出频率的限定条件**

定位指令执行时，实际输出脉冲的最低频率由以下公式所限制：

$$F_{\min\_acc} = \sqrt{\frac{F_{\max} \times 500}{T}}$$

上式中，F_{max} 表示最高速度，通过（SD223，SD222）或（SD203，SD202）设置。T 表示加减速时间，通过 SD225、SD226 或 SD205、SD206 设置，单位为毫秒，范围为 2 ～ 5000。计算结果 F_{min_acc} 为最低输出限制值。

如果定位指令中指定的输出频率为 F，以下为实际输出频率的三种情况：

（1）F 小于基底频率或 F 大于 F_{max}，实际没有输出。

（2）F 小于 F_{min_acc}，实际输出为 F_{min_acc}。

（3）F 大于或等于 F_{min_acc}，且小于等于 F_{max}，输出为 F。

11.4 MX 系列定位指令相关软元件

1．输出轴 0 ～输出轴 7（Y0 ～ Y7 对应的轴）的特殊辅助继电器

地址	名称	功能	R/W
SM29	指令执行结束标志	定位指令执行结束时，置 ON	R
SM70 ～ SM77	Y0 ～ Y7 输出完成中断使能标志	置 ON 时，使能 PTO 输出完成中断	R/W
SM80 ～ SM81、SM261 ～ SM266	Y0 ～ Y7 脉冲输出停止使能标志	置 ON 时，高速脉冲输出停止	R/W
SM82/SM83 、SM267 ～ SM272	Y0 ～ Y7 脉冲输出监控位	高速脉冲输出监视（调用时为 ON，未调用时为 OFF）	R
SM260	中断输入功能指定有效	适用 DDVIT。当不使用中断输入指定功能，那么 Y0 ～ Y7 对应 X0 ～ X7 的中断；当使用中断输入指定功能时则该位置 ON，然后对应的 SD198、SD199 的每一个 4 位对应着每一个输出 Y(n) 的输入	R/W
SM280/SM290/SM310 /SM320/SM330/SM340 /SM350/SM360	Y0 ～ Y7 清零功能有效	适用 DZRN /DSZR，定位指令原点回归（DZRN）清零信号输出功能有效	R/W

地址	名称	功能	R/W
SM281/SM291/SM311 /SM321/SM331/SM341 /SM351/SM361	Y0 ～ Y7 清零信号指定元件有效	适 用 DZRN/DSZR， 对 应 的 SD210/SD230/SD290/SD310/SD330/SD350/SD370/SD390 内 的数值分别为 Y0 ～ Y7 的清零 Y 元件，不指定则 Y10 ～ Y17 分别为 Y0 ～ Y7 的清零信号	R/W
SM282/SM292/SM312 /SM322/SM332/SM342 /SM352/SM362	Y0 ～ Y7 原点回归方向标志	适用于 DZRN/DSZR，作用于 Y(n) 对应的轴：置 ON 时，表示原点回归方向为正转方向；置 OFF 时，表示原点回归方向为反转方向	R/W
SM283/SM293/SM313 /SM323/SM333/SM343 /SM353/ SM363	Y0 ～ Y7 正转极限标志	适用于 DZRN/DSZR/DDRVI/DDRVA，作用于 Y (n) 对应的轴：置 ON 时，表示到达正转方向的极限；置 OFF 时，表示未到极限	R/W
SM284/SM294/SM314 /SM324/SM334/SM344 /SM354/ SM364	Y0 ～ Y7 反转极限标志	适用于 DZRN/DSZR/DDRVI/DDRVA，作用于 Y(n) 对应的轴：置 ON 时，表示到达反转方向的极限；置 OFF 时，表示未到极限	R/W
SM285/SM295/SM315 /SM325/SM335/SM345 /SM355/SM365	Y0 ～ Y7 近点信号逻辑反转使能标志	适用于 DZRN/DSZR，作用于 Y(n) 对应的轴：置 ON 时，按负逻辑处理（输入 OFF 时，近点信号为 ON）；置 OFF 时，按正逻辑处理（输入 ON 时，近点信号为 ON）	R/W
SM286/SM296/SM316 /SM326/SM336/SM346 /SM356/SM366	Y0 ～ Y7 零点信号逻辑反转使能标志	适用于 DSZR，作用于 Y (n) 对应的轴：置 ON 时，按负逻辑处理（输入 OFF 时，零点信号为 ON）；置 OFF 时，按正逻辑处理（输入 ON 时，零点信号为 ON）	R/W
SM287/SM297/SM317 /SM327/SM337/SM347 /SM357/SM367	Y0 ～ Y7 中断信号逻辑反转使能标志	适用于 DDVIT，作用于 Y (n) 对应的轴：置 ON 时，按负逻辑处理（输入 OFF 时，中断信号为 ON）；置 OFF 时，按正逻辑处理（输入 ON 时，中断信号为 ON）	R/W
SM288/SM298/SM318 /SM328/SM338/SM348 /SM358/SM368	Y0 ～ Y7 定位指令运行标志	用于监测执行 DZRN/DDRVI/DDRVA/DSZR/DDVIT 时高速输出通道 Y(n) 的状态，正在运行时为 ON，空闲时为 OFF	R/W
SM289/SM299/SM319 /SM329/SM339/SM349 /SM359/SM369	Y0 ～ Y7 用户中断输入使能标志	SD198 指定为用户中断指令软元件时，该位置 ON 则使能 Y(n) 用户中断	R/W
注：清零有效信号 SM280 置位时会导致原点回归到位时该输出轴对应的默认清零信号 Y10 发出 1 个 CLR 脉冲输出，脉冲宽度为 20ms+1 个扫描周期。如果该轴的默认清零信号用于其它用途，则应该复位清零有效信号，使这个功能无效。			

2．输出轴 0 ～输出轴 7（Y0 ～ Y7 对应的轴）的特殊数据寄存器

地址	名称	功能	R/W
SD50/SD52/SD70 /SD72/SD74/SD76 /SD78/SD80	PLSR/PLSY 指令输出 Y0 ～ Y7 脉冲总数（低位）	Y0 ～ Y7 发送脉冲总数，与脉冲方向无关	R/W
SD51/DSD53/SD71 /SD73/SD75/SD77 /SD79/SD81	PLSR/PLSY 指令输出 Y(n) 脉冲总数（高位）		R/W
SD200/SD220/SD260 /SD280/SD300/SD320 /SD340/SD360	Y0 ～ Y7 定位输出当前值（低位）	用于绝对位置的计算和保存。每次执行定位指令，都根据（SD51，SD50）和方向信号来计算和更新（SD201，SD200）。当开机并且从伺服驱动器读取绝对位置数据时，请将得到的绝对位置数据（32 位长整型）放入（SD201，SD200）中	R/W
SD201/SD221/SD261 /SD281/SD301/SD321 /SD341/SD361	Y0 ～ Y7 定位输出当前值（高位）		R/W

地址	名称	功能	R/W
SD202/SD222/SD282 /SD302/SD322/SD342 /SD362/SD382	Y0 ～ Y7 定位输出最高速度（低位）	执行定位指令时的最高速度，范围：10 ～ 100000，单位：Hz	R/W
SD203/SD223/SD283 /SD303/SD323/SD343 /SD363/SD383	Y0 ～ Y7 定位输出最高速度（高位）		R/W
SD204/SD224/SD284 /SD304/SD324/SD344 /SD364/SD384	Y0 ～ Y7 定位输出基底速度	执行定位指令时的基底速度（最高速度的 1/10 以下）	R/W
SD205/SD225/SD285 /SD305/SD325/SD345 /SD365/SD385	Y0 ～ Y7 定位输出加速时间	执行定位指令时的加速时间，范围：2 ～ 5000，单位：ms	R/W
SD206/SD226/SD286 /SD306/SD326/SD346 /SD366/SD386	Y0 ～ Y7 定位输出减速时间	执行定位指令时的减速时间，范围：2 ～ 5000，单位：ms	R/W
SD207/SD227/SD287 /SD307/SD327/SD347 /SD367/SD387	Y0 ～ Y7 爬行速度	适用于 DZRN/DSZR，作为执行该指令时的爬行速度	R/W
SD208/SD228/SD288 /SD308/SD328/SD348 /SD368/SD388	Y0 ～ Y7 原点回归速度（低位）	适用于 DZRN/DSZR，作为执行该指令时的原点回归速度	R/W
SD209/SD229/SD289 /SD309/SD329/SD349 /SD369/SD389	Y0 ～ Y7 原点回归速度（高位）		R/W
SD210/SD230/SD290 /SD310/SD330/SD350 /SD370/SD390	Y0 ～ Y7 清零信号软元件指定	SM281 置位时，使用该元件中的数值 N 对应的 Y 元件 Y(n) 表示清零信号	R/W
SD198/SD199	Y0 ～ Y3/ Y4 ～ Y7 输出轴的中断信号软元件指定（4 位代表一个轴）	指定作为中断输入的输入编号（X0 ～ X7），或者指定用户中断指令软元件。其中，SD198 中 b0 ～ b3 位对应脉冲输出 Y0 的中断输入，b4 ～ b7 位对应脉冲输出 Y1 的中断输入	R/W

注 1：SD202 ～ SD206、 SD222 ～ SD226、 SD282 ～ SD286、SD302 ～ SD306、 SD322 ～ SD326、 SD342 ～ SD346、 SD362 ～ SD366、 SD382 ～ SD386 可以由用户根据需要修改。请确保在驱动定位指令前进行赋值。在定位指令执行过程中改变上述参数时，可能会影响指令的正确执行。
注 2：基底速度必须小于最高速度的 1/10，否则基底速度将自动赋值为最高速度的 1/10。当定位指令中的速度低于基底速度或大于最高速度时，将不输出脉冲。

第十二章 特殊功能模块使用指南

12.1 特殊功能模块

MX 系列 PLC 不仅具有强大的逻辑处理、数据运算、高速处理等功能，而且具有 A/D、D/A 转换、PID、TPID 调节、2 通道高速计数、2 轴运动控制等功能，通过使用模拟量输入、输出模块、温度控制模块、高速计数模块和运动控制模块等，使 MX 系列 PLC 在温度、流量、液压、压力、定位等过程控制系统中得到了广泛的应用。

I/O 扩展模块以及各种特殊功能模块，统称为扩展模块。MX 系列 PLC 的特殊功能模块包括 MX2H-4AD、MX2H-4DA、MX2H-4PT、MX2H-4TC、MX2H-8TC、 MX2H-2HC、MX2H-2PG 等。模块的型号及规格如下表所示。

1．模拟量模块型号及规格

型号	模拟量输入		模拟量输出		尺寸（mm）长 × 宽 × 高
	输入点数	输入类型	输出点数	输出类型	
MX2H-4AD	4	电压、电流	—		58×90×83
MX2H-4DA	—		4	电压、电流	58×90×83
MX2H-4PT	4	热电阻	—		58×90×83
MX2H-4TC	4	热电偶	—		58×90×83
MX2H-8TC	8	热电偶	—		58×90×83
MX2H-4WT	4	称重传感器	—		58×90×83

2．高速输入输出模块型号及规格

型号	输入输出特性	尺寸（mm）长 × 宽 × 高
MX2H-2HC	支持 2 路 200K 的高速输入和 4 路普通输出	58×90×83
MX2H-2PG	支持 2 路 200K 的高速输出和 2 路普通输出	58×90×83

本章节以 MX2H-4AD 模块为例介绍模拟量输入，以 MX2H-4DA 为例介绍模拟量输出，以 MX2H-4PT 为例介绍热电阻温度输入，以 MX2H-4TC 为例介绍热电偶温度输入，以 MX2H-4WT 为例介绍称重模块，以 MX2H-2HC 为例介绍高速计数输入，以 MX2H-2PG 为例介绍高速脉冲输出。

12.2 MX2H-4AD 模拟量输入模块

MX2H-4AD 为模拟量输入模块，可配合 MX 系列主模块工作，实现 4 个模拟量输入通道的信号检测。每个通道都具有电压信号及电流信号输入端口，信号幅值分别可为 - 10V ～ 10V、-20mA ～ 20mA、4mA ～ 20mA。主模块通过 FROM/TO 指令访问扩展模块内寄存器的 BFM 单元，也可使用“MOV ×× Un.b”访问扩展模块内寄存器的 BFM 单元。

12.2.1 电气规格

（1） 电源指标

项目	说明
模拟电路	24VDC （ - 15% ～ 20%），最大允许纹波电压 5%，48mA（来自主单元的外部电源或外接）
数字电路	3.3VDC 25mA（源于主模块的内部电源）

（2） 电源消耗规格

最大消耗电流（3.3VDC/GND）	最大消耗电流（24VDC/COM）
25mA	48mA

（3） 性能指标

项目		指标	
转换速度		2ms/ 通道	
模拟输入量程	电压输入	- 10V ～ 10V，输入阻抗为 500KΩ	可以同时使用 4 个通道，通过设定 BFM 可进行输入量程选择（详见表 2）
	电流输入	- 20mA ～ 20mA，输入阻抗为 250Ω	

项目		指标
数字输出（16bit）		默认设置为：电压：－2000～2000　电流：－1000～1000 可由用户设定的最大量程为：－10000～10000
分辨率	电压输入	5mV
	电流输入	20μA
精度		±1% 全量程
隔离		模拟电路和数字电路之间通过数字隔离器进行隔离。模拟电路与模块输入 24VDC 电源内部隔离。模拟通道之间不隔离。

12.2.2　输入信号的接线

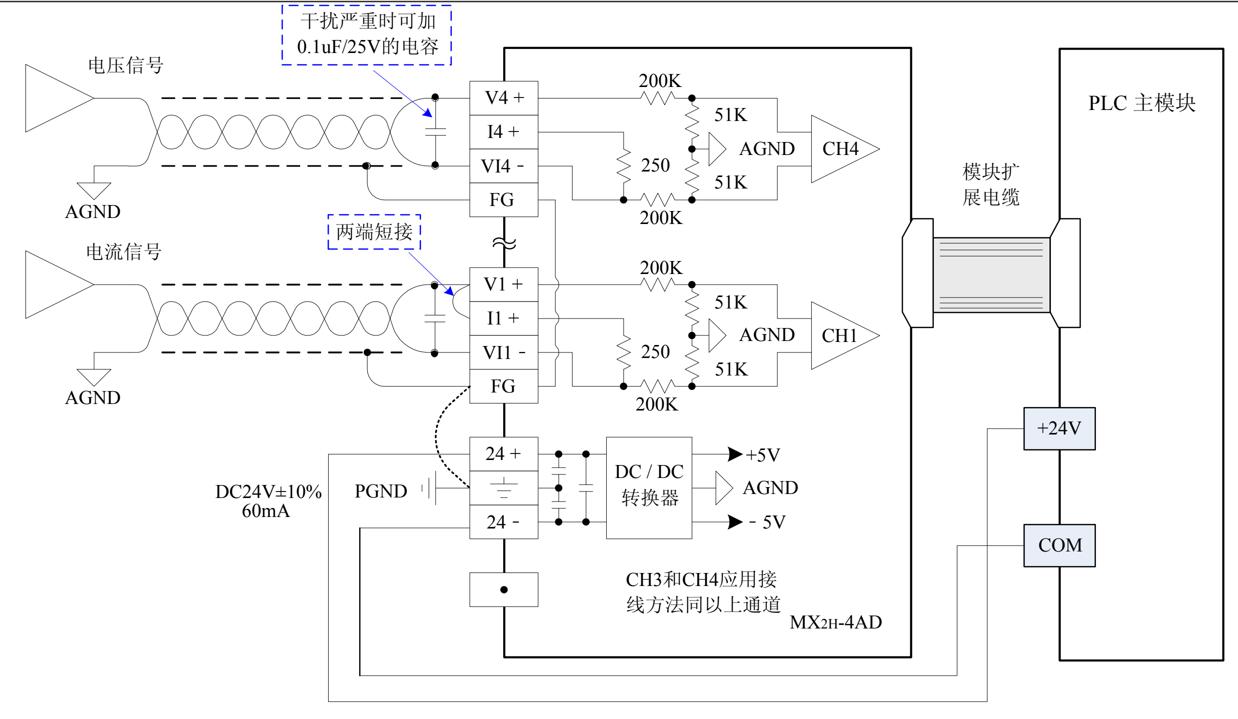


图 1　MX2H-4AD 模块输入信号接线示意图

12.2.3　缓存区（BFM）

MX2H-4AD 与主模块之间通过缓存区（BFM）交换信息，主模块通过 Un.b 元件与 MX2H-4AD 缓存区（BFM）直接映射；n：特殊模块序号；b：BFM 序号；例如，U1.0 对应特殊模块地址为 1，BFM#0 的单元。主模块可通过 FROM/TO 指令访问扩展模块内寄存器的 BFM 单元，也可使用“MOV ×× Un.b”访问扩展模块内寄存器的 BFM 单元。MX2H-4AD 的缓存区（BFM）具体内容如下表所示：

表 1　MX2H-4AD 的缓存区（BFM）内容

Un.b	内容	备注	读写属性
Un.0	通道模式字	缺省值：16#FFFF	R/W
Un.2	CH1 平均值点数	缺省值：8	R/W
Un.3	CH2 平均值点数	缺省值：8	R/W
Un.4	CH3 平均值点数	缺省值：8	R/W
Un.5	CH4 平均值点数	缺省值：8	R/W
Un.10	CH1 通道采样值		R

Un.b	内容	备注	读写属性
Un.11	CH2 通道采样值		R
Un.12	CH3 通道采样值		R
Un.13	CH4 通道采样值		R
Un.30	CH1-D0 零点数字量	缺省值：0（输入模式 0）	R/W
Un.31	CH1-A0 零点模拟量	缺省值：0（输入模式 0）	R
Un.32	CH1-D1 最大数字量	缺省值：2000（输入模式 0）	R/W
Un.33	CH1-A1 最大模拟量	缺省值：10000（输入模式 0）	R
Un.34	CH2-D0 零点数字量	缺省值：0（输入模式 0）	R/W
Un.35	CH2-A0 零点模拟量	缺省值：0（输入模式 0）	R
Un.36	CH2-D1 最大数字量	缺省值：2000（输入模式 0）	R/W
Un.37	CH2-A1 最大模拟量	缺省值：10000（输入模式 0）	R
Un.38	CH3-D0 零点数字量	缺省值：0（输入模式 0）	R/W
Un.39	CH3-A0 零点模拟量	缺省值：0（输入模式 0）	R
Un.40	CH3-D1 最大数字量	缺省值：2000（输入模式 0）	R/W
Un.41	CH3-A1 最大模拟量	缺省值：10000（输入模式 0）	R
Un.42	CH4-D0 零点数字量	缺省值：0（输入模式 0）	R/W
Un.43	CH4-A0 零点模拟量	缺省值：0（输入模式 0）	R
Un.44	CH4-D1 最大数字量	缺省值：2000（输入模式 0）	R/W
Un.45	CH4-A1 最大模拟量	缺省值：10000（输入模式 0）	R
Un.198	模块软件版本信息		R
Un.199	模块的识别码	缺省值：16#1020	R

说明：

1．输入模式选择由 Un.0 中的 4 位十六进制数字 16#$\times_4\times_3\times_2\times_1$ 控制。$\times_1$ 控制通道 1，$\times_2$ 控制通道 2，$\times_3$ 控制通道 3，$\times_4$ 控制通道 4。字符值所表示的信息如表 2 所示。

表 2 字符值信息表

×	状态信息
0	模式 0：输入量程 - 10V ～ 10V
1	模式 1：输入量程 4mA ～ 20mA
2	模式 2：输入量程 - 20mA ～ 20mA
F	通道关闭

举例：

若对 Un.0 单元写入“16#F210”，将完成如下设置：

通道 1 的输入量程：- 10V ～ 10V；通道 2 的输入量程：4mA ～ 20mA；通道 3 的输入量程：- 20mA ～ 20mA；通道 4 关闭。

2．Un.2 ～ Un.5 作为通道的平均值点数的设定缓存区，提供 2、4、8、16、32、64、128、256 供用户选择。缺省值为 8。例如，平均值点数设为 4，即求 4 次采样值的平均值。

3．Un.10 ～ Un.13 作为通道采样值的缓存区。

4．Un.30 ～ Un.45 为通道特性设置数据缓存区，使用两点法设置通道特性，D0、D1 表示通道输出的数字量，A0、A1 表示通道实际输入，A0、A1 数据的单位是 mV，每通道占用 4 个字。考虑到方便用户的设置，同时并不影响功能的实现，将 A0、A1 的值固定为当前模式下模拟量的 0 值和最大值（当选用模式 1 时，A0 的值固定为当前模式下模拟量的最小值）。

详见下表：

A \ D	D0（－ 10000 ～ 10000）	D1（－ 10000 ～ 10000）
模式 0：输出量程－ 10V ～ 10V	0V	10V
模式 1：输出量程 4mA ～ 20mA	4mA	20mA
模式 2：输出量程 -20mA ～ 20mA	0mA	20mA

对通道模式字（Un.0）进行更改时，A0、A1 会根据模式自动更改，用户对此两项设置的写入无效。

☞ 注意

若通道输入为电流信号，当前通道应选择模式 1 或 2，由于通道内部测量基于电压信号，因此，电流信号由通道的电流输入端 250Ω 电阻（参见图 1）转换为电压信号（－ 5V ～ 5V），当前通道对应的特性设置区域中的 A0、A1 值仍然以 mV 为单位，即 5000mV，也就是 20mA×250Ω=5000mV。

D0、A0、D1、A1 的更改对通道特性的改变，请参见特性设置部分。

5．Un.198 作为模块软件版本信息的缓存区。

6．Un.199 作为模块识别码的缓存区。MX2H-4AD 的识别码是 16#1020。可编程控制器中的用户程序可以在程序中使用这个识别码，以在传输 / 接收数据之前确认此特殊模块。

12.2.4 特性设置

MX2H-4AD 的输入通道特性为通道模拟输入量 A 与通道数字量 D 之间的线性关系，可由用户设置，每个通道可以理解为图 2 中所示的模型，由于其为线性特性，因此只要确定两点 P0（A0，D0）、P1（A1，D1），即可确定通道的特性，其中，D0 表示模拟量输入为 A0 时通道输出数字量，D1 表示模拟量输入为 A1 时通道输出数字量。

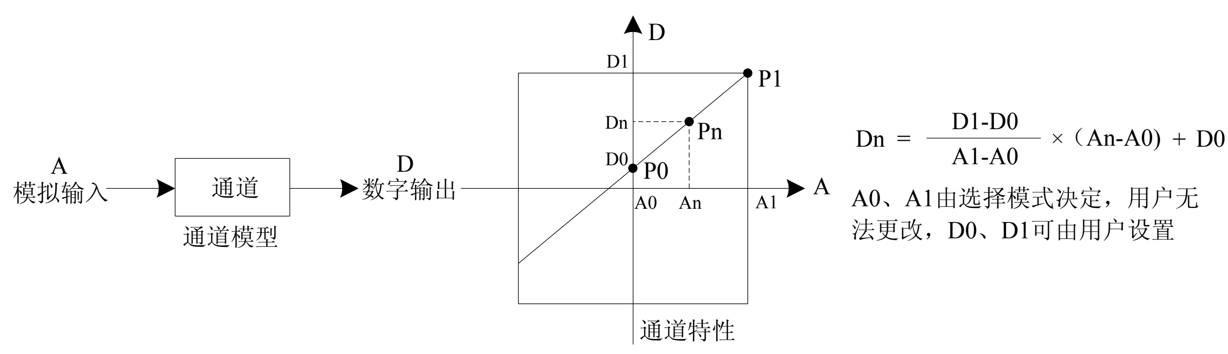
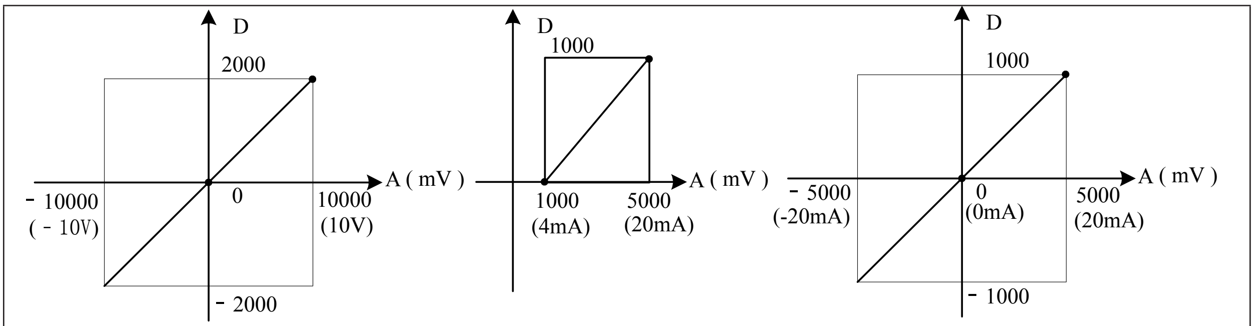


图 2 MX2H-4AD 的通道特性示意图

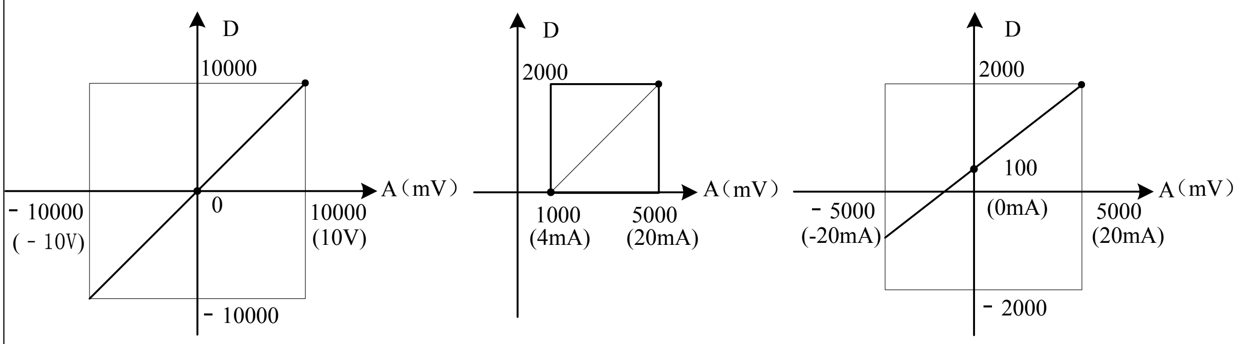
若不更改各通道的 D0、D1 值，仅设置通道的模式（Un.0），那么，每种模式对应的特性如图 3 所示。



A．模式0（出厂设定） B．模式1（出厂设定） C．模式2（出厂设定）

图 3 不更改各通道的 D0、D1 值，各种模式对应通道特性

若更改通道的 D0、D1 数值，即可更改通道特性，D0、D1 可在 - 10000 ～ 10000 之间任意设定，若设定值超出此范围，MX2H-4AD 不会接受，并保持原有有效设置，图 4 为特性更改举例。



A. 模式 0
D0=0，D1=10000
输入0V 时，对应数字量为 0
输入10V 时，对应数字量为 10000

B. 模式 1
D0=0，D1=2000
输入4 mA 时，对应数字量为 0
输入20 mA 时，对应数字量为 2000

C. 模式 2
D0=100，D1=2000
输入0 mA 时，对应数字量为 100
输入20 mA 时，对应数字量为 2000

图 4 特性更改举例

12.2.5 应用示例

例 1：系统组态配置

MX2H-4AD 模块地址为 0，使用其第 1 通道输入电压信号（ - 10V ～ 10V），通道特性更改为图 4 中的 A；第 2 通道输入电流信号（4mA ～ 20mA），通道特性更改为图 4 中的 B；第 3 通道输入电流信号（ - 20mA ～ 20mA），通道特性更改为图 4 中的 C；第 4 通道关闭；平均值点数设为 4，为了实现上述配置，可以在 MXProgrammer 中“工程管理器→系统组态→硬件设置”中进行配置，如图 5 所示。

图 5 系统组态配置

例 2：指令配置

MX2H-4AD 模块地址为 0，通道模式和平均值点数与例 1 相同，通道特性采用默认值，将各通道采样值与 D10 相加，分别存入 D11，D12，D13，D14 中，如图 6 所示。

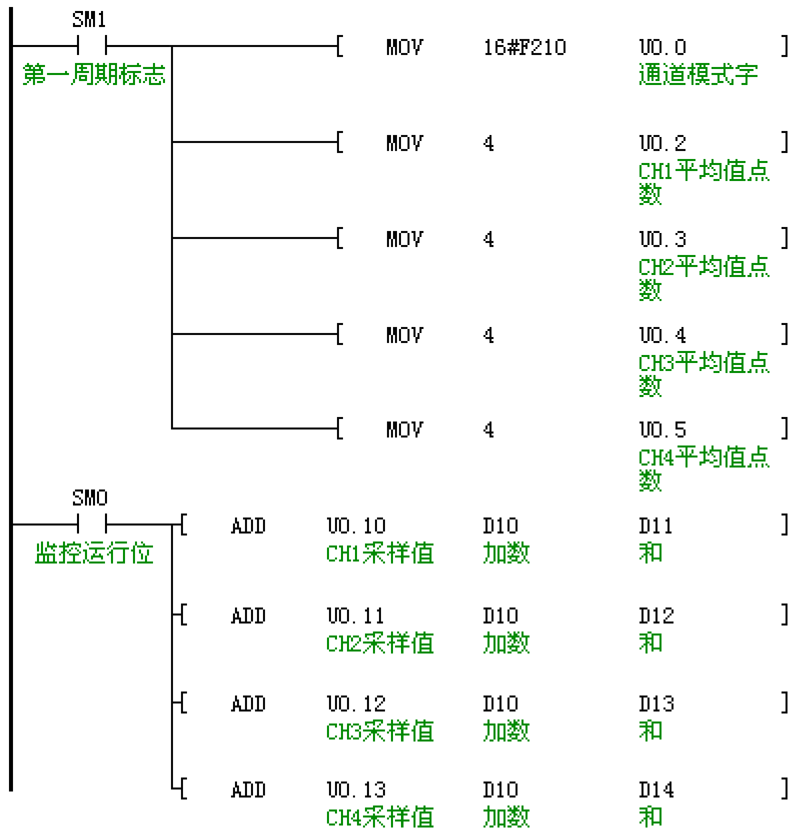


图 6 指令配置

☞ 注意

用系统组态和指令配置都可更改通道特性。当系统组态和指令同时配置且不相同时，以指令配置为准。

12.3 MX2H-4DA 模拟量输出模块

MX2H-4DA 为模拟量输出模块，可配合 MX 系列主模块工作，实现 4 个模拟信号通道的输出，每个通道都具有电压信号及电流信号输出端口，信号幅值分别可为 - 10V ～ 10V、0mA ～ 20mA、4 mA ～ 20mA。主模块通过 FROM/TO 指令访问本扩展模块内寄存器的 BFM 单元，也可使用“MOV ×× Un.b”访问扩展模块内寄存器的 BFM 单元。

12.3.1 电气规格

（1）电源指标

项目	说明
模拟电路	24VDC（ - 15% ～ 20%），最大允许纹波电压 5%，158mA（来自主单元的外部电源或外接）
数字电路	3.3VDC 25mA（源于主模块的内部电源）

（2）电源消耗规格

最大消耗电流（3.3VDC/GND）	最大消耗电流（24VDC/COM）
25mA	158mA

（3）性能指标

项目	指标
转换速度	1ms/ 通道（改变所用的通道数不会改变转换速度）

项目		指标
模拟输出	电压输出	- 10V ～ 10VDC（外部负载阻抗大于 1KΩ）
	电流输出	0mA ～ 20mA（外部负载阻抗小于 500Ω） 4 mA ～ 20mA（外部负载阻抗小于 500Ω）
数字输入		默认设置：电压输出 - 2000 ～ 2000，电流输出 0 ～ 1000 有效范围： - 10000 ～ 10000
分辨率	电压输出	5mV
	电流输出	20μA
精度		±1% 全量程
隔离		模拟电路和数字电路之间通过数字隔离器进行隔离。模拟电路与模块输入 24VDC 电源内部隔离。 模拟通道之间不隔离。

12.3.2 输出信号的接线

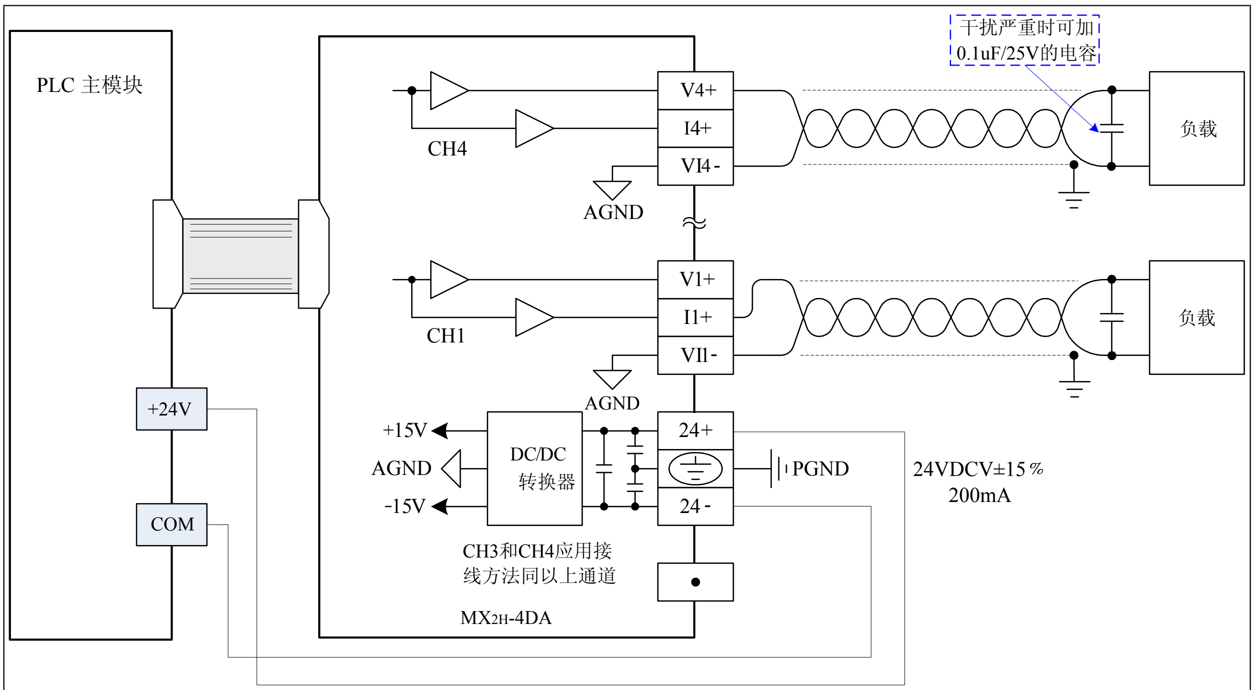


图 7 MX2H-4DA 模块输出信号接线示意图

12.3.3 缓存区（BFM）

MX2H-4DA 与主模块之间通过缓存区（BFM）交换信息，主模块通过 Un.b 元件与 MX2H-4DA 缓存区（BFM）直接映射；n：特殊模块序号；b：BFM 序号；例如，U1.0 对应特殊模块地址为 1，BFM#0 的单元。主模块通过 FROM/TO 指令访问扩展模块内寄存器的 BFM 单元，也可使用“MOV ×× Un.b”访问扩展模块内寄存器的 BFM 单元。MX2H-4DA 的缓存区（BFM）具体内容见表 3 所示。

表 3 MX2H-4DA 的缓存区（BFM）内容

Un.b	内容	备注	读写属性
Un.0	通道模式字	缺省值：16#FFFF	R/W
Un.2	CH1 通道输出数据		R/W
Un.3	CH2 通道输出数据		R/W
Un.4	CH3 通道输出数据		R/W
Un.5	CH4 通道输出数据		R/W

Un.b	内容	备注	读写属性
Un.10	停机使能字	缺省值：0	R/W
Un.12	CH1 通道停机输出量	缺省值：0	R/W
Un.13	CH2 通道停机输出量	缺省值：0	R/W
Un.14	CH3 通道停机输出量	缺省值：0	R/W
Un.15	CH4 通道停机输出量	缺省值：0	R/W
Un.30	CH1-D0 零点数字量	缺省值：0（输出模式 0）	R/W
Un.31	CH1-A0 零点模拟量	缺省值：0（输出模式 0）	R
Un.32	CH1-D1 最大数字量	缺省值：2000（输出模式 0）	R/W
Un.33	CH1-A1 最大模拟量	缺省值：10000（输出模式 0）	R
Un.34	CH2-D0 零点数字量	缺省值：0（输出模式 0）	R/W
Un.35	CH2-A0 零点模拟量	缺省值：0（输出模式 0）	R
Un.36	CH2-D1 最大数字量	缺省值：2000（输出模式 0）	R/W
Un.37	CH2-A1 最大模拟量	缺省值：10000（输出模式 0）	R
Un.38	CH3-D0 零点数字量	缺省值：0（输出模式 0）	R/W
Un.39	CH3-A0 零点模拟量	缺省值：0（输出模式 0）	R
Un.40	CH3-D1 最大数字量	缺省值：2000（输出模式 0）	R/W
Un.41	CH3-A1 最大模拟量	缺省值：10000（输出模式 0）	R
Un.42	CH4-D0 零点数字量	缺省值：0（输出模式 0）	R/W
Un.43	CH4-A0 零点模拟量	缺省值：0（输出模式 0）	R
Un.44	CH4-D1 最大数字量	缺省值：2000（输出模式 0）	R/W
Un.45	CH4-A1 最大模拟量	缺省值：10000（输出模式 0）	R
Un.198	模块软件版本信息		R
Un.199	模块的识别码	缺省值：16#2002	R

说明：

1．输入模式选择由 Un.0 中的 4 位十六进制数字 16#$\times_4\times_3\times_2\times_1$ 控制。$\times_1$ 控制通道 1，$\times_2$ 控制通道 2，$\times_3$ 控制通道 3，$\times_4$ 控制通道 4。字符值所表示的信息如表 4 所示。

表 4 字符值信息表

×	状态信息
0	模式 0：输入量程－ 10V ～ 10V
1	模式 1：输入量程 4mA ～ 20mA
2	模式 2：输入量程－ 20mA ～ 20mA
F	通道关闭

举例：

若对 Un.0 单元写入“16#F210”，将完成如下设置：

通道 1 的输入量程：－ 10V ～ 10V；通道 2 的输入量程：4mA ～ 20mA；通道 3 的输入量程：0mA ～ 20mA；通道 4 关闭。

2．Un.2 ～ Un.5 作为通道输出值的缓存区。

3．Un.10 作为停机输出使能字的缓存区，采用二进制，从低位开始每一位代表一个通道，比如当最低位为 1 时，表示当主模块处于 STOP 状态时，MX2H-4DA 的第 1 通道按 Un.12 所设置的输出值进行输出。

4．Un.12 ～ Un.15 作为通道停机输出量的缓存区。

5．Un.30 ～ Un.45 为通道特性设置数据缓存区，使用两点法设置通道特性，D0、D1 表示通道输入的数字量，A0、A1

表示通道实际输出的模拟量，A0、A1 数据的单位是 mV 或 uA，每通道占用 4 个字。考虑到方便用户的设置，同时并不影响功能的实现，将 A0、A1 的值固定为当前模式下模拟量的 0 值和最大值（当选用模式 1 时，A0 的值固定为当前模式下模拟量的最小值），详见下表：

A \ D	D0（－ 10000 ～ 10000）	D1（－ 10000 ～ 10000）
模式 0：输出量程－ 10V ～ 10V	0V	10V
模式 1：输出量程 4mA ～ 20mA	4mA	20mA
模式 2：输出量程 0mA ～ 20mA	0mA	20mA

通道模式字（Un.0）进行更改时，A0、A1 会根据模式自动更改，用户对此两项设置的写入无效。

6．Un.198 作为模块软件版本信息的缓存区。

7．Un.199 作为模块识别码的缓存区。MX2H-4DA 的识别码是 16#2002。可编程控制器中的用户程序可以在程序中使用这个识别码，以在传输 / 接收数据之前确认此特殊模块。

12.3.4 特性设置

MX2H-4DA 的输出通道特性为通道模拟输出量 A 与通道数字量 D 之间的线性关系，可由用户设置，每个通道可以理解为图 8 中所示的模型，由于其为线性特性，因此只要确定两点 P0（A0，D0）、P1（A1，D1），即可确定通道的特性，其中 D0 表示模拟量输出为 A0 时的数字量，D1 表示模拟量输出为 A1 时的数字量。

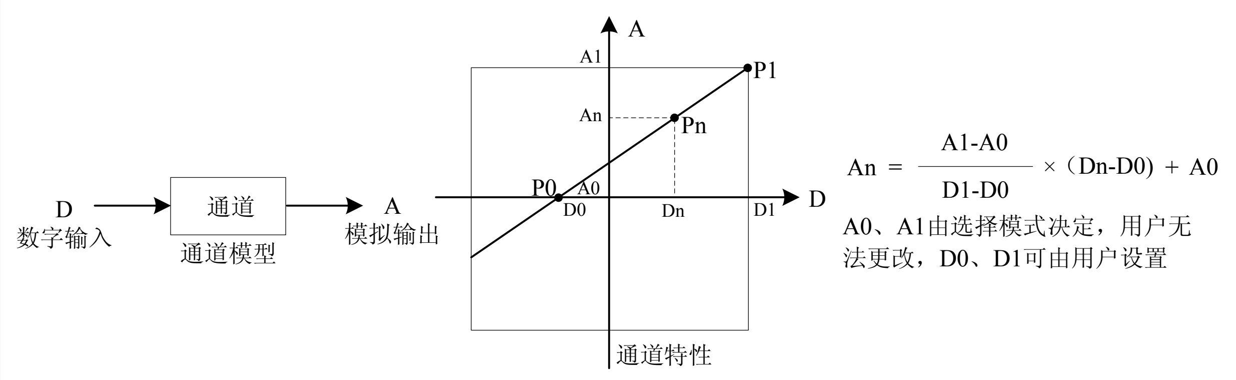
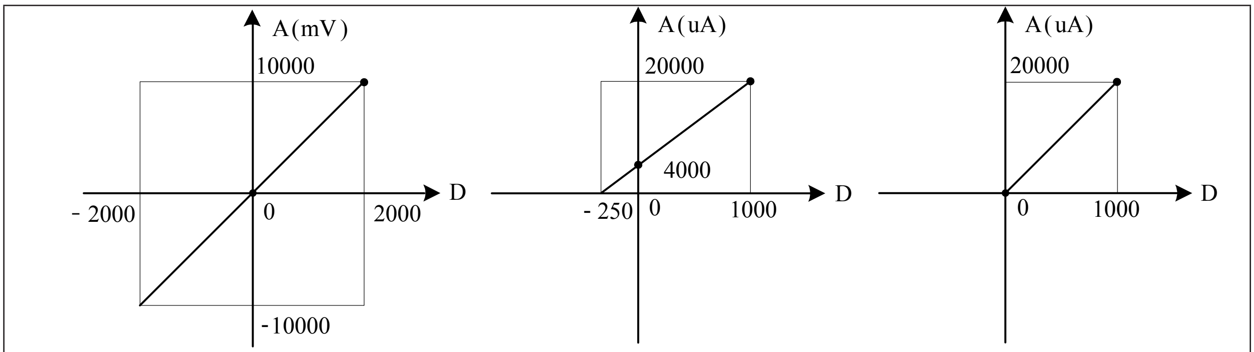


图 8 MX2H-4DA 的通道特性示意图

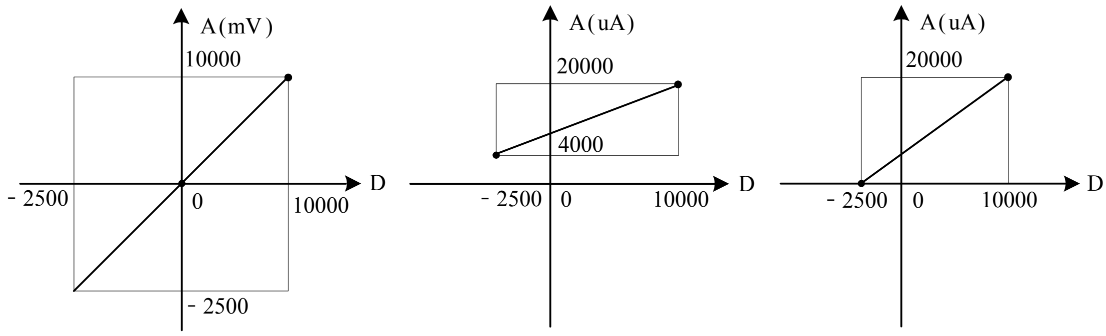
若不更改各通道的 D0、D1 值，仅设置通道的模式（Un.0），那么，每种模式对应的特性如图 9 所示。



A．模式0:（出厂设定） B．模式1:（出厂设定） C．模式2:（出厂设定）

图 9 不更改各通道的 D0、D1 值，各种模式对应通道特性

若更改通道的 D0、D1 数值，即可更改通道特性，D0、D1 可在－ 10000 ～ 10000 之间任意设定，若设定值超出此范围，MX2H-4AD 不会接受，并保持原有有效设置，图 10 为特性更改举例。



A. 模式 0
D0=0，D1=10000
输入10000时，对应输出10V
输入0时，对应输出0V

B．模式 1
D0= - 2500，D1=10000
输入10000时，对应输出20mA
输入 - 2500时，对应输出4mA

C．模式 2
D0= - 2500，D1=10000
输入10000时，对应输出20mA
输入 - 2500时，对应输出0mA

图 10 特性更改举例

12.3.5 应用示例

例 1：系统组态配置

MX2H-4DA 模块地址为 0，使用其第 1 通道输出电压信号（ - 10V ～ 10V），通道特性更改为上图中 A；第 2 通道输出电流信号（4mA ～ 20mA），通道特性更改为上图中的 B；第 3 通道输出电流信号（0mA ～ 20mA），通道特性更改为上图中的 C；第 4 通道关闭，使能第 1 通道的停机输出，当 PLC 由 RUN → STOP 时，第 1 通道输出 1V 电压。为了实现上述配置，可以在 MXProgrammer 中“工程管理器→系统组态→硬件设置”中进行配置，如图 11 所示。

图 11 系统组态配置

例 2：MX2H-4DA 模块地址为 0，通道模式与例 1 相同，通道特性采用默认值，第 1 通道输出数据为 500，则输出的模拟量为 2.5V，第 2 通道输出数据为 750，则输出模拟量为 16mA，第 3 通道输出数据为 750，则输出模拟量为 15mA，如图 12 所示。

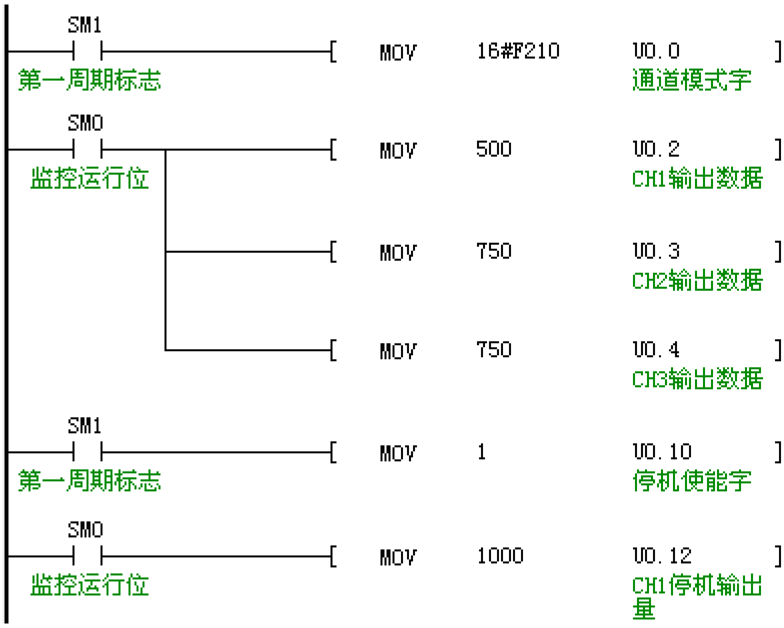


图 12 指令配置

☞ 注意

用系统组态和指令配置都可更改通道特性。当系统组态和指令同时配置且不相同时，以指令配置为准。

12.4 MX2H-4PT 热电阻温度输入模块

MX2H-4PT 为热电阻温度输入模块，可配合 MX 系列主模块工作，实现 4 路热电阻温度信号输入。主模块通过 FROM/TO 指令访问本扩展模块内寄存器的 BFM 单元，也可使用“MOV ×× Un.b”访问扩展模块内寄存器的 BFM 单元。

12.4.1 电气规格

（1）电源指标

项目	说明
模拟电路	24VDC（－ 15% ～ 20%），最大允许纹波电压 5%，50mA（来自主单元的外部电源或外接）
数字电路	3.3VDC 25mA（源于主模块的内部电源）

（2）电源消耗规格

最大消耗电流（3.3VDC/GND）	最大消耗电流（24VDC/COM）
25mA	50mA

（3）性能指标

<table>
<tr><th rowspan="2">项目</th><th colspan="4">指标</th></tr>
<tr><th colspan="2">摄氏（℃）</th><th colspan="2">华氏（℉）</th></tr>
<tr><td>输入信号</td><td colspan="4">热电阻类型：Pt100、JPt100、Cu100、Cu50</td></tr>
<tr><td>转换速度</td><td colspan="4">100ms/ 通道</td></tr>
<tr><td rowspan="4">额定温度范围</td><td>Pt100</td><td>－ 150℃～ 600℃</td><td>Pt100</td><td>－ 238°F ～ 1112°F</td></tr>
<tr><td>JPt100</td><td>－ 150℃～ 500℃</td><td>JPt100</td><td>－ 238°F ～ 932°F</td></tr>
<tr><td>Cu100</td><td>－ 30℃～ 120℃</td><td>Cu100</td><td>－ 22°F ～ 248°F</td></tr>
<tr><td>Cu50</td><td>－ 30℃～ 120℃</td><td>Cu50</td><td>－ 22°F ～ 248°F</td></tr>
</table>

项目	指标			
	摄氏（℃）		华氏（℉）	
数字输出	Pt100	- 1500 ～ 6000	Pt100	- 2380 ～ 11120
	JPt100	- 1500 ～ 5000	JPt100	- 2380 ～ 9320
	Cu100	- 300 ～ 1200	Cu100	- 220 ～ 2480
	Cu50	- 300 ～ 1200	Cu50	- 220 ～ 2480
最低分辨率	Pt100	0.2℃	Pt100	0.36°F
	Cu100	0.2℃	Cu100	0.36°F
	Cu50	0.2℃	Cu50	0.36°F
精度	±1% 全量程			
隔离	模拟电路和数字电路之间通过数字隔离器进行隔离；模拟电路与模块输入 24VDC 电源内部隔离；模拟通道之间不隔离。			

12.4.2 输入信号的接线

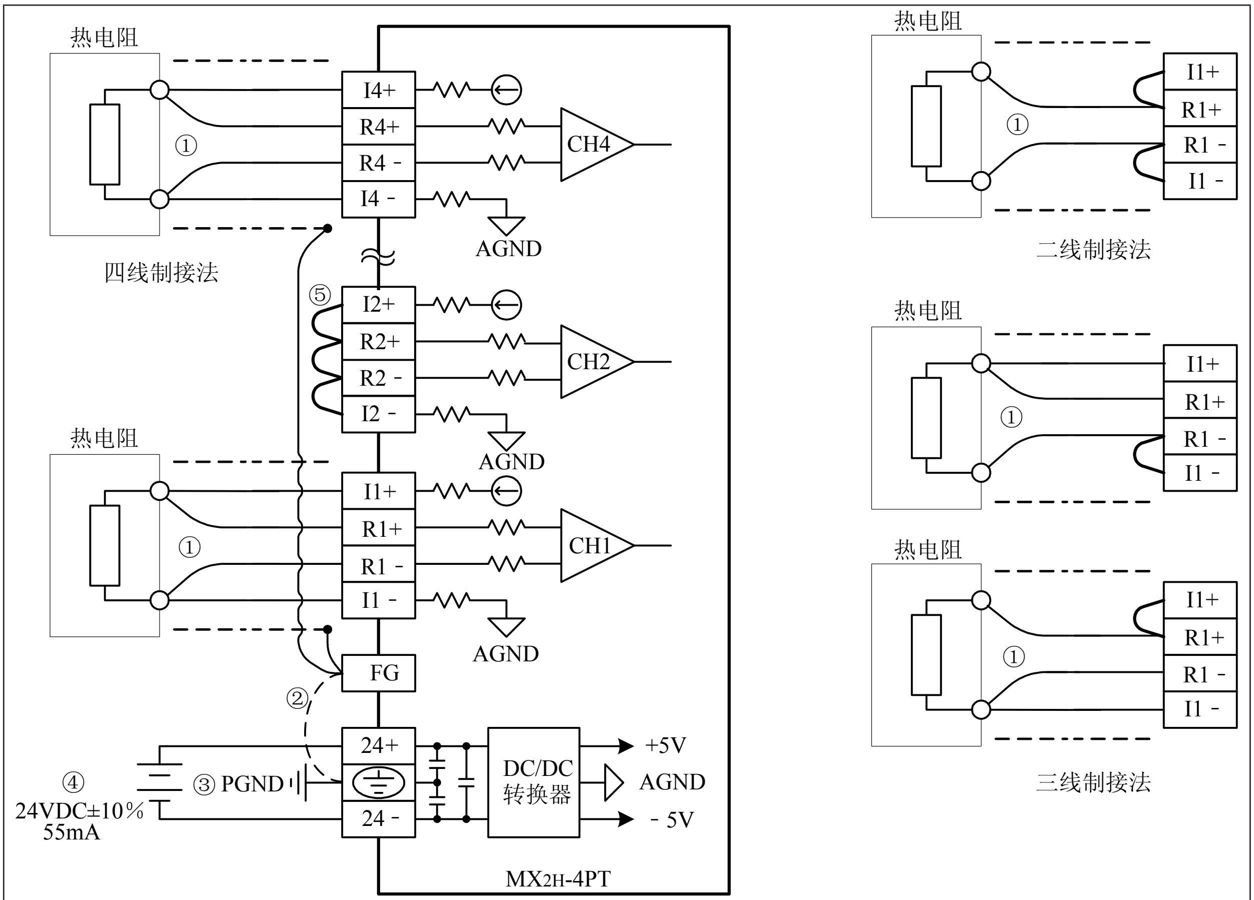


图 13 MX2H-4PT 输入信号接线示意图

12.4.3 缓存区（BFM）

MX2H-4PT 与主模块之间通过缓存区（BFM）交换信息，主模块通过 Un.b 元件与 MX2H-4PT 缓存区（BFM）直接映射；n：特殊模块序号；b：BFM 序号；例如，U1.0 对应特殊模块地址为 1，BFM#0 的单元。主模块可通过 FROM/TO 指令访问扩展模块内寄存器的 BFM 单元，也可使用“MOV ×× Un.b”访问扩展模块内寄存器的 BFM 单元。MX2H-4PT 的缓存区（BFM）具体内容如表 5 所示。

表 5 MX2H-4PT 的缓存区（BFM）内容

Un.b	内容	备注	读写属性
Un.0	通道模式字	缺省值：16#FFFF	R/W
Un.2	CH1 平均值点数	缺省值：8	R/W
Un.3	CH2 平均值点数	缺省值：8	R/W
Un.4	CH3 平均值点数	缺省值：8	R/W
Un.5	CH4 平均值点数	缺省值：8	R/W
Un.10	CH1 通道采样值	单位：℃	R
Un.11	CH2 通道采样值	单位：℃	R
Un.12	CH3 通道采样值	单位：℃	R
Un.13	CH4 通道采样值	单位：℃	R
Un.20	CH1 通道采样值	单位：°F	R
Un.21	CH2 通道采样值	单位：°F	R
Un.22	CH3 通道采样值	单位：°F	R
Un.23	CH4 通道采样值	单位：°F	R
Un.30	CH1-D0 零点数字量	缺省值：0（输入模式 0）	R/W
Un.31	CH1-A0 零点模拟量	缺省值：0（输入模式 0）	R
Un.32	CH1-D1 最大数字量	缺省值：6000（输入模式 0）	R/W
Un.33	CH1-A1 最大模拟量	缺省值：6000（输入模式 0）	R
Un.34	CH2-D0 零点数字量	缺省值：0（输入模式 0）	R/W
Un.35	CH2-A0 零点模拟量	缺省值：0（输入模式 0）	R
Un.36	CH2-D1 最大数字量	缺省值：6000（输入模式 0）	R/W
Un.37	CH2-A1 最大模拟量	缺省值：6000（输入模式 0）	R
Un.38	CH3-D0 零点数字量	缺省值：0（输入模式 0）	R/W
Un.39	CH3-A0 零点模拟量	缺省值：0（输入模式 0）	R
Un.40	CH3-D1 最大数字量	缺省值：6000（输入模式 0）	R/W
Un.41	CH3-A1 最大模拟量	缺省值：6000（输入模式 0）	R
Un.42	CH4-D0 零点数字量	缺省值：0（输入模式 0）	R/W
Un.43	CH4-A0 零点模拟量	缺省值：0（输入模式 0）	R
Un.44	CH4-D1 最大数字量	缺省值：6000（输入模式 0）	R/W
Un.45	CH4-A1 最大模拟量	缺省值：6000（输入模式 0）	R
Un.198	模块软件版本信息		R
Un.199	模块的识别码	缺省值：16#4020	R

说明：

1. 输入模式选择由 Un.0 中的 4 位十六进制数字 16#$\times_4\times_3\times_2\times_1$ 控制。$\times_1$ 控制通道 1，$\times_2$ 控制通道 2，$\times_3$ 控制通道 3，$\times_4$ 控制通道 4。字符值所表示的信息如表 6 所示。

表 6 字符值信息表

×	状态信息
0	模式 0：Pt100 型热电阻，输入范围 - 150.00℃～ 600.0℃
1	模式 1：JPt100 型热电阻，输入范围 - 150.00℃～ 500.0℃

×	状态信息
2	模式 2：Cu100 型热电阻，输入范围 - 30.00℃～ 120.0℃
3	模式 3：Cu50 型热电阻，输入范围 - 30.00℃～ 120.0℃
F	通道关闭

举例，若对 Un.0 单元写入“16#F210”，将完成如下设置：
通道 1 的输入量程：Pt100 型热电阻，输入范围： - 150.00℃～ 600.0℃；
通道 2 的输入量程：JPt100 型热电阻，输入范围： - 150.00℃～ 500.0℃；
通道 3 的输入量程：Cu100 型热电阻，输入范围： - 30.00℃～ 120.0℃；
通道 4 关闭。

2．Un.2 ～ Un.5 作为通道的平均值点数的设定缓存区，提供 2、4、8、16、32、64、128、256 供用户选择。缺省值为 8。例如，平均值点数设为 4，即求 4 次采样值的平均值。

3．Un.10 ～ Un.13 作为通道采样值的缓存区，单位为摄氏度（℃）；Un.20 ～ Un.23 作为通道采样值缓存区，单位为华氏度（° F）。

4．Un.30 ～ Un.45 为通道特性设置数据缓存区，使用两点法设置通道特性，D0、D1 表示通道输出的数字量，单位为 0.1℃，A0、A1 表示通道实际输入温度值，A0、A1 数据的单位是 0.1℃，每通道占用 4 个字。考虑到方便用户的设置，同时并不影响功能的实现，将 A0、A1 的值固定为当前模式下温度输入的 0℃和最大温度值，详见下表：

A \ D	D0	D1
模式 0：Pt100 型热电阻	0℃	600.0℃
模式 1：JPt100 型热电阻	0℃	500.0℃
模式 2：Cu100 型热电阻	0℃	120.0℃
模式 3：Cu50 型热电阻	0℃	120.0℃

对通道模式字（Un.0）进行更改时，A0、A1 会根据模式自动更改，用户对此两项设置的写入无效。

☞ 注意

1．特性参数中均以 0.1℃为数据单位，对于华氏度（°F）参数，请按下述表达式进行转换后再写入特性设置中。
摄氏℃（Celsius）= 5/9 ×（华氏 °F（Fahrenheit）－ 32）

2．D0、A0、D1、A1 的更改对通道特性的改变，请参见下述的特性设置。

5．Un.198 作为模块软件版本信息的缓存区。

6．Un.199 作为模块识别码的缓存区。MX2H-4PT 的识别码是 16#4020。可编程控制器中的用户程序可以在程序中使用这个识别码，以在传输 / 接收数据之前确认此特殊模块。

12.4.4 特性设置

MX2H-4PT 的输入通道特性为通道模拟输入量 A 与通道数字量 D 之间的线性关系，可由用户设置，每个通道可以理解为图 14 所示的模型，由于其为线性特性，因此只要确定两点 P0（A0，D0）、P1（A1，D1），即可确定通道的特性，其中，D0 表示模拟量输入为 A0 时通道输出数字量，D1 表示模拟量输入为 A1 时通道输出数字量。

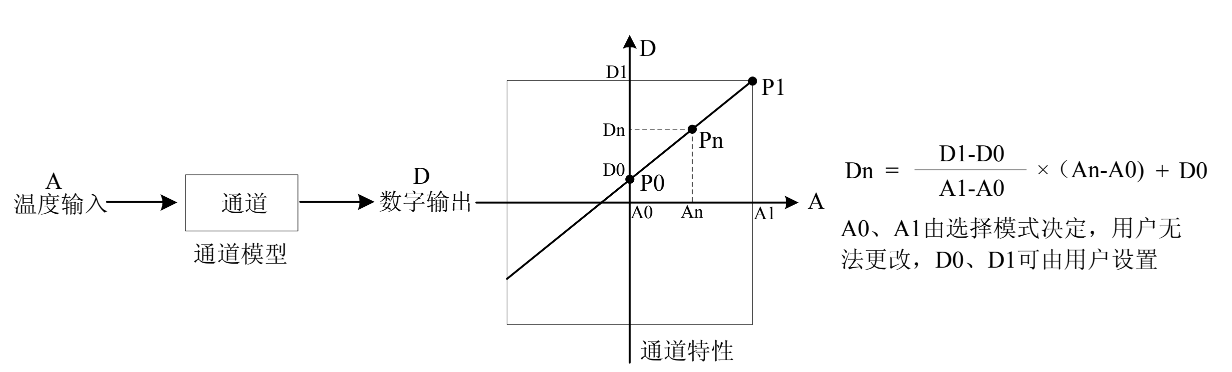


图 14 MX2H-4PT 的通道特性示意图

若不更改各通道的 D0、D1 值，仅设置通道的模式（Un.0），那么模式 0，模式 1，模式 2 对应的特性如图 15 所示：

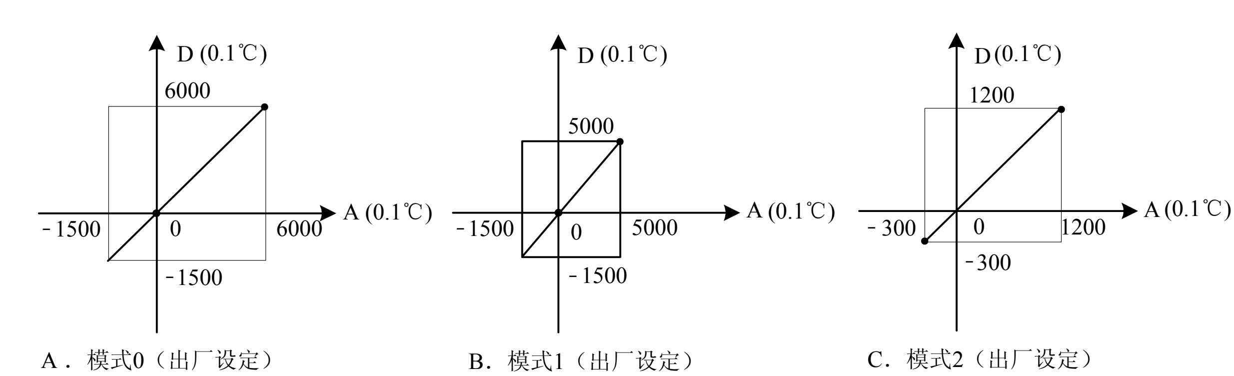
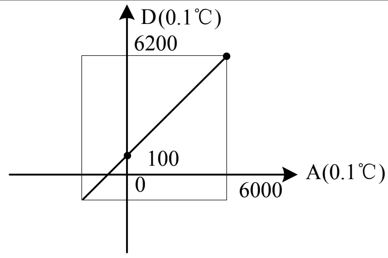


图 15 不更改各通道的 D0、D1 值，各种模式对应通道特性

若更改通道的 D0、D1 数值，即可更改通道特性，D0、D1 允许在出厂设定基础上调整 ±1000（0.1℃），即当模式 0 时，D0 允许调整的范围为 - 1000 ～ 1000（0.1℃），D1 允许调整的范围为 5000 ～ 7000（0.1℃），若设定值超出此范围，MX2H-4PT 不会接受，并保持原有有效设置，图 16 为特性更改举例。



模式0

D0=100，D1=6200

输入0℃时，对应数字量为100

输入600℃时，对应数字量为6200

图 16 特性更改举例

12.4.5 应用示例

例 1：系统组态配置

MX2H-4PT 模块地址为 0，使用其第 1 通道输入模式为 0，通道特性更改为图 13 所示；第 2 通道输入模式为 1，通道特性采用默认值；第 3 通道输入模式为 2，通道特性采用默认值；第 4 通道关闭；平均值点数设为 4，为了实现上述配置，可以在 MXProgrammer“工程管理器→系统组态→硬件设置”中进行配置，如图 17 所示。

图 17 系统组态配置

例 2：指令配置

MX2H-4PT 模块地址为 0，通道模式和平均值点数与例 1 相同，通道特性采用默认值，将各通道采样值与 D10 相加，分别存入 D11，D12，D13，D14 中，如图 18 所示。

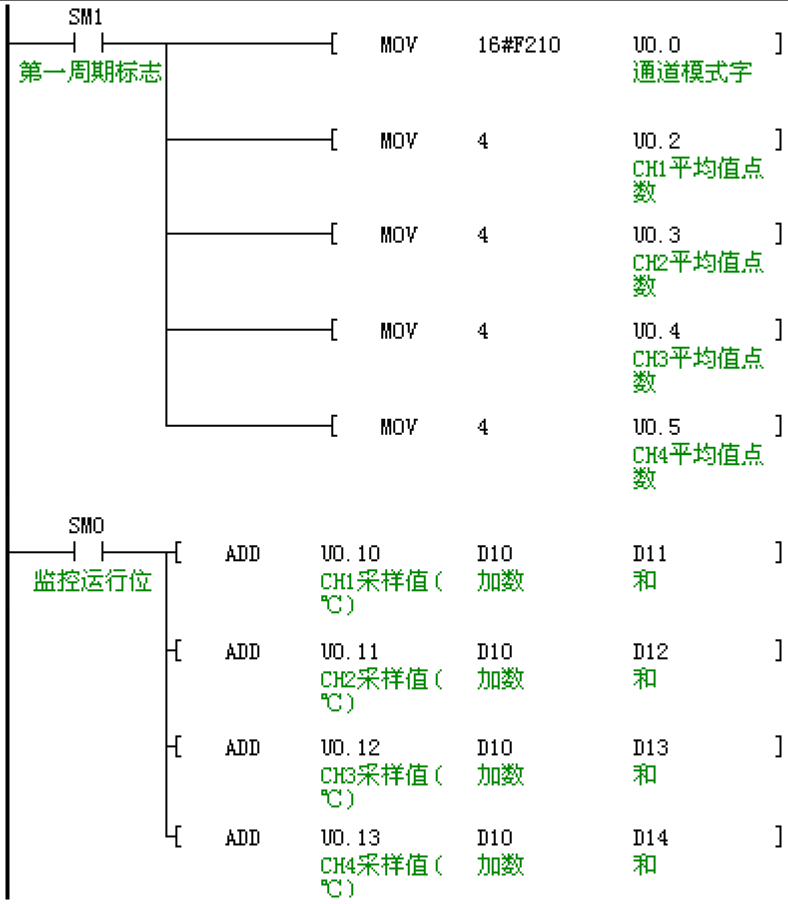


图 18 指令配置示例

☞ 注意

用系统组态和指令配置都可更改通道特性。当系统组态和指令同时配置且不相同时，以指令配置为准。

12.5 MX2H-4TC 热电偶温度输入模块

MX2H-4TC 为热电偶温度输入模块，可配合 MX2H 系列主模块工作，实现 4 路热电偶温度信号输入。主模块通过 FROM/TO 指令访问本扩展模块内寄存器的 BFM 单元，也可使用“MOV ×× Un.b”访问扩展模块内寄存器的 BFM 单元。

12.5.1 电气规格

（1）电源指标

项目	说明
模拟电路	24VDC（－ 15%～20%），最大允许纹波电压 5%，30mA(来自主单元的外部电源或外接)
数字电路	3.3VDC 25mA（源于主模块的内部电源）

（2）电源消耗规格

最大消耗电流（3.3VDC/GND）	最大消耗电流（24VDC/COM）
25mA	30mA

（3）性能指标

项目	指标			
	摄氏（℃）		华氏（℉）	
输入信号	热电偶类型： K、J、E、N、T、R、S（每个通道 7 种都可使用）			
转换速度	100ms/ 通道			
额定温度范围	类型 K	- 100℃～ 1200℃	类型 K	- 148°F ～ 1112°F
	类型 J	- 100℃～ 600℃	类型 J	- 148°F ～ 1112°F
	类型 E	- 100℃～ 850℃	类型 E	- 148°F ～ 1562°F
	类型 N	- 100℃～ 1200℃	类型 N	- 148°F ～ 2192°F
	类型 T	- 200℃～ 300℃	类型 T	- 328°F ～ 572°F
	类型 R	0℃～ 1600℃	类型 R	32°F ～ 2912°F
	类型 S	0℃～ 1600℃	类型 S	32°F ～ 2912°F
数字输出	类型 K	- 1000 ～ 12000	类型 K	- 1480 ～ 21920
	类型 J	- 1000 ～ 6000	类型 J	- 1480 ～ 11120
	类型 E	- 1000 ～ 8500	类型 E	- 1480 ～ 15620
	类型 N	- 1000 ～ 12000	类型 N	- 1480 ～ 21920
	类型 T	- 2000 ～ 3000	类型 T	- 3280 ～ 7520
	类型 R	0 ～ 16000	类型 R	320 ～ 29120
	类型 S	0 ～ 16000	类型 S	320 ～ 29120
最低分辨率	类型 K	0.3℃	类型 K	0.54°F
	类型 J	0.2℃	类型 J	0.36°F
	类型 E	0.3℃	类型 E	0.54°F
	类型 N	0.3℃	类型 N	0.54°F
最低分辨率	类型 T	0.2℃	类型 T	0.36°F
	类型 R	0.5℃	类型 R	0.9°F
	类型 S	0.5℃	类型 S	0.9°F
总精度校正点	±（1% 全范围 ±1.5℃）			
隔离	模拟电路和数字电路之间通过数字隔离器进行隔离；模拟电路与模块输入 24VDC 电源内部隔离；模拟通道之间不隔离。			

12.5.2 输入信号的接线

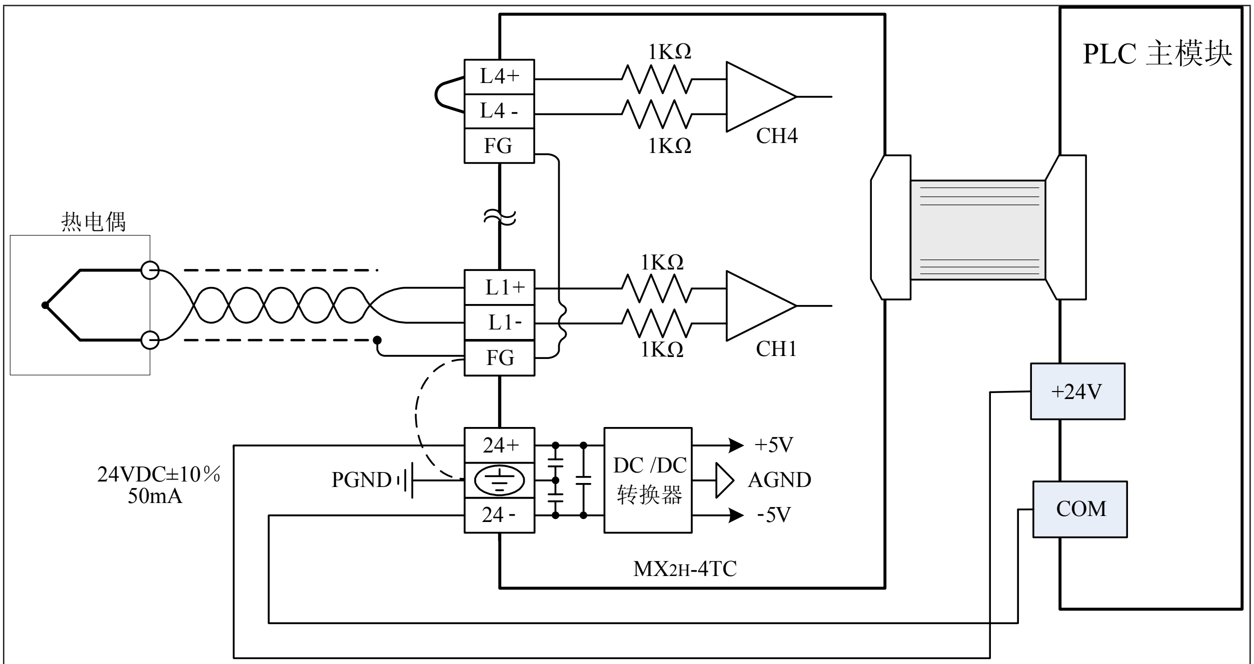


图 19 MX2H-4TC 模块输入信号接线示意图

12.5.3 缓存区（BFM）

MX2H-4TC 与主模块之间通过缓存区（BFM）交换信息，主模块通过 Un.b 元件与 MX2H-4TC 缓存区（BFM）直接映射；n：特殊模块序号；b：BFM 序号；U1.0 对应特殊模块地址为 1，BFM#0 的单元。主模块可通过 FROM/TO 指令访问扩展模块内寄存器的 BFM 单元，也可使用“MOV ×× Un.b”访问扩展模块内寄存器的 BFM 单元。MX2H-4TC 的缓存区（BFM）具体内容如表 7 所示。

表 7 MX2H-4TC 的缓存区（BFM）内容

Un.b	内容	备注	读写属性
Un.0	通道模式字	缺省值：16#FFFF	R/W
Un.2	CH1 平均值点数	缺省值：8	R/W
Un.3	CH2 平均值点数	缺省值：8	R/W
Un.4	CH3 平均值点数	缺省值：8	R/W
Un.5	CH4 平均值点数	缺省值：8	R/W
Un.10	CH1 通道采样值	单位：℃	R
Un.11	CH2 通道采样值	单位：℃	R
Un.12	CH3 通道采样值	单位：℃	R
Un.13	CH4 通道采样值	单位：℃	R
Un.20	CH1 通道采样值	单位：°F	R
Un.21	CH2 通道采样值	单位：°F	R
Un.22	CH3 通道采样值	单位：°F	R
Un.23	CH4 通道采样值	单位：°F	R
Un.30	CH1-D0 零点数字量	缺省值：0（输入模式 0）	R/W
Un.31	CH1-A0 零点模拟量	缺省值：0（输入模式 0）	R
Un.32	CH1-D1 最大数字量	缺省值：12000（输入模式 0）	R/W

Un.b	内容	备注	读写属性
Un.33	CH1-A1 最大模拟量	缺省值：12000（输入模式 0）	R
Un.34	CH2-D0 零点数字量	缺省值：0（输入模式 0）	R/W
Un.35	CH2-A0 零点模拟量	缺省值：0（输入模式 0）	R
Un.36	CH2-D1 最大数字量	缺省值：12000（输入模式 0）	R/W
Un.37	CH2-A1 最大模拟量	缺省值：12000（输入模式 0）	R
Un.38	CH3-D0 零点数字量	缺省值：0（输入模式 0）	R/W
Un.39	CH3-A0 零点模拟量	缺省值：0（输入模式 0）	R
Un.40	CH3-D1 最大数字量	缺省值：12000（输入模式 0）	R/W
Un.41	CH3-A1 最大模拟量	缺省值：12000（输入模式 0）	R
Un.42	CH4-D0 零点数字量	缺省值：0（输入模式 0）	R/W
Un.43	CH4-A0 零点模拟量	缺省值：0（输入模式 0）	R
Un.44	CH4-D1 最大数字量	缺省值：12000（输入模式 0）	R/W
Un.45	CH4-A1 最大模拟量	缺省值：12000（输入模式 0）	R
Un.195	模块断偶状态字	缺省值：16#0000	R
Un.198	模块软件版本信息	0x1000	R
Un.199	模块的识别码	缺省值：16#3020	R

说明：

1. 输入模式选择由 Un.0 中的 4 位十六进制数字 16#$\times_4\times_3\times_2\times_1$ 控制。$\times_1$ 控制通道 1，$\times_2$ 控制通道 2，$\times_3$ 控制通道 3，$\times_4$ 控制通道 4。字符值所表示的信息如表 8 所示。

表 8 字符值信息表

×	状态信息
0	模式 0：K 型热电偶，输入范围 - 100.00℃～ 1200.0℃
1	模式 1：J 型热电偶，输入范围 - 100.00℃～ 600.0℃
2	模式 2：E 型热电偶，输入范围 - 100.00℃～ 850.0℃
3	模式 3：N 型热电偶，输入范围 - 100.00℃～ 1200.0℃
4	模式 4：T 型热电偶，输入范围 - 200.00℃～ 300.0℃
5	模式 5：R 型热电偶，输入范围 0.00℃～ 1600.0℃
6	模式 6：S 型热电偶，输入范围 0.00℃～ 1600.0℃
F	通道关闭

举例，若对 Un.0 单元写入“16#F420”，将完成如下设置：
通道 1 的输入量程：K 型热电偶，输入范围： - 100.00℃～ 1200.0℃；
通道 2 的输入量程：E 型热电偶，输入范围： - 100.00℃～ 850.0℃；
通道 3 的输入量程：T 型热电偶，输入范围： - 200.00℃～ 300.0℃；
通道 4 关闭。

2. Un.2 ～ Un.5 作为通道的平均值点数的设定缓存区，提供 2、4、8、16、32、64、128、256 供用户选择。缺省值为 8。
3. Un.10 ～ Un.13 作为通道采样值的缓存区，单位为摄氏度（℃）。
4. Un.20 ～ Un.23 作为通道采样值的缓存区，单位为华氏度（°F）。
5. Un.30 ～ Un.45 为通道特性设置数据缓存区，使用两点法设置通道特性，D0、D1 表示通道输出的数字量，单位为

0.1℃，A0、A1 表示通道实际输入温度值，A0、A1 数据的单位是 0.1℃，每通道占用 4 个字。考虑到方便用户的设置，同时并不影响功能的实现，将 A0、A1 的值固定为当前模式下温度输入的 0℃和最大温度值，详见下表：

A \ D	D0	D1
模式 0：K 型热电偶	0℃	1200.0℃
模式 1：J 型热电偶	0℃	600.0℃
模式 2：E 型热电偶	0℃	850.0℃
模式 3：N 型热电偶	0℃	1200.0℃
模式 4：T 型热电偶	0℃	300.0℃
模式 5：R 型热电偶	0℃	1600.0℃
模式 6：S 型热电偶	0℃	1600.0℃

对通道模式字（Un.0）进行更改时，A0、A1 会根据模式自动更改，用户对此两项设置的写入无效。

☞ 注意

特性参数中均以 0.1℃为数据单位，对于华氏度（°F）参数，请按下述表达式进行转换后写入特性设置中。

摄氏℃（Celsius）= 5/9 ×（华氏 °F（Fahrenheit）－ 32）

D0、A0、D1、A1 的更改对通道特性的改变，请参见下述的特性设置。

6．Un.195 作为模块断偶状态字的缓存区，由 4 位十六进制数字 16#$\times_4\times_3\times_2\times_1$ 表示。$\times_1$ 表示通道 1 的断偶状态，如果 $\times_1$ 为 1，则表示通道 1 断偶，为 0 则表示没有断偶；$\times_2$ 表示通道 2 的断偶状态，$\times_3$ 通道 3 的断偶状态，$\times_4$ 通道 4 的断偶状态。

7．Un.198 作为模块软件版本信息的缓存区。

8．Un.199 作为模块识别码的缓存区。MX2H-4PT 的识别码是 16#4020。PLC 中的用户程序可以在程序中使用这个识别码，以在传输 / 接收数据之前确认此模块。

12.5.4 特性设置

MX2H-4TC 的输入通道特性为通道模拟输入量 A 与通道数字量 D 之间的线性关系，可由用户设置，每个通道可以理解为图 20 所示的模型，由于其为线性特性，因此只要确定两点 P0（A0，D0）、P1（A1，D1），即可确定通道的特性，其中，D0 表示模拟量输入为 A0 时通道输出数字量，D1 表示模拟量输入为 A1 时通道输出数字量。

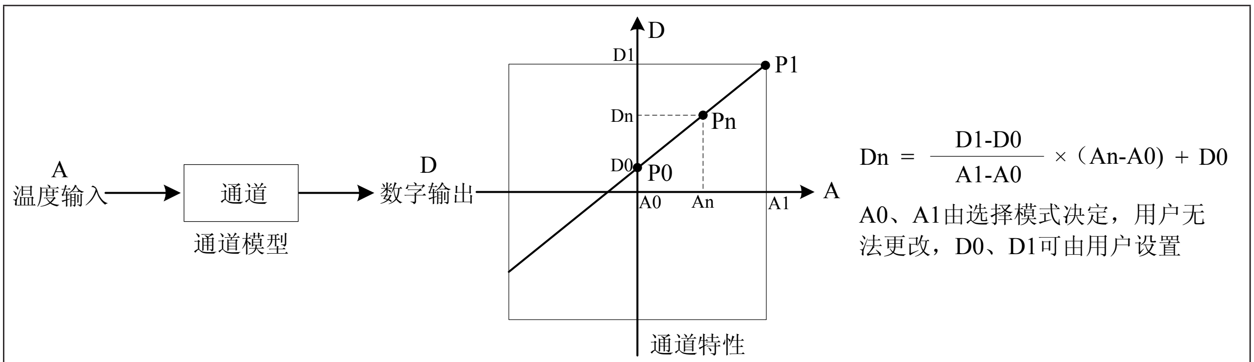
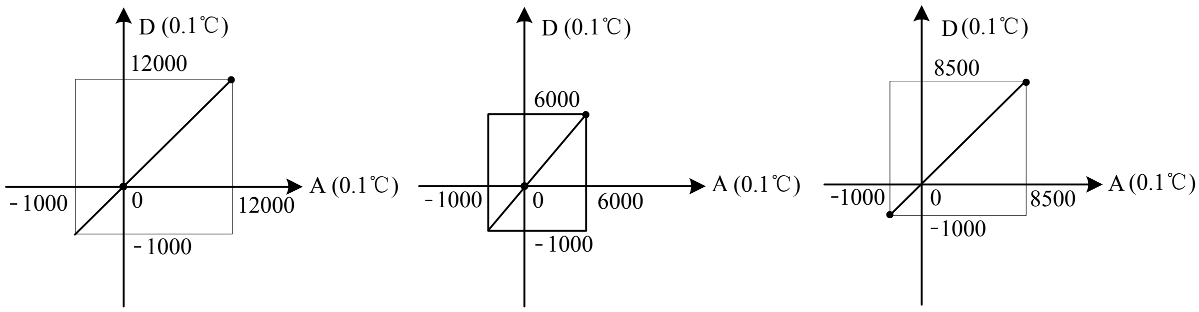


图 20 MX2H-4TC 的通道特性示意图

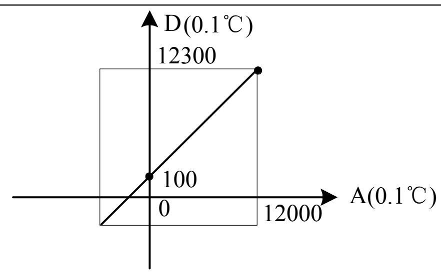
若不更改各通道的 D0、D1 值，仅设置通道的模式（Un.0），那么模式 0，模式 1，模式 2 对应的特性如图 21 所示：



A． 模式0（出厂设定） B． 模式1（出厂设定） C． 模式2（出厂设定）

图 21 不更改各通道的 D0、D1 值，各种模式对应通道特性

若更改通道的 D0、D1 数值，即可更改通道特性，D0、D1 允许在出厂设定基础上调整 ±1000（0.1℃），即当模式 0 时，D0 允许调整的范围为 - 1000 ～ 1000（0.1℃），D1 允许调整的范围为 11000 ～ 13000（0.1℃），若设定值超出此范围，MX2H-4TC 不会接受，并保持原有有效设置，图 22 为特性更改举例。



模式 0
D0=100，D1=12300
输入0℃时，对应数字量为100
输入1200℃时，对应数字量为12300

图 22 特性更改举例

12.5.5 应用示例

例 1：系统组态配置

MX2H-4TC 模块地址为 0，使用其第 1 通道输入模式为 0，通道特性更改为图 21 所示；第 2 通道输入模式 1，通道特性采用默认值；第 3 通道输入模式 2，通道特性采用默认值；第 4 通道关闭；平均值点数设为 4，为了实现上述配置，可以在 MXProgrammer“工程管理器→系统组态→硬件设置”中进行配置，如图 23 所示。

图 23 系统组态配置

例 2：指令配置

MX2H-4TC 模块地址为 0，通道模式和平均值点数与例 1 相同，通道特性采用默认值，将各通道采样值与 D10 相加，分别存入 D11，D12，D13，D14 中，如图 24 所示。

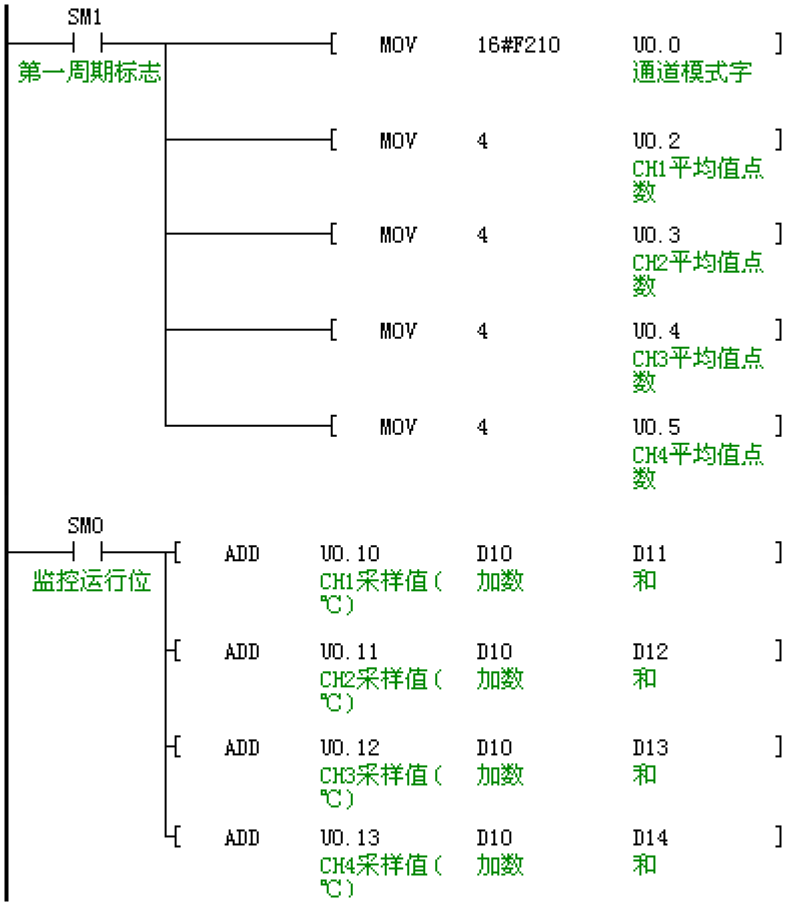


图 24 指令配置

☞ 注意

用系统组态和指令配置都可更改通道特性。当系统组态和指令同时配置且不相同时，以指令配置为准。

12.6 MX2H-4WT 称重模块

MX2H-4WT 扩展模块可配合 MX2H 系列主模块工作，实现 4 路称重信号的检测。每个通道都具有电压信号输入端口，通过将检测到的电压信号转换成称重的重量。

主模块通过 FROM/TO 指令访问扩展模块内寄存器的 BFM 单元，也可使用 “MOV ×× Un.b” 访问扩展模块内寄存器的 BFM 单元。

12.6.1 电气规格

（1）电源指标

项目	说明
模拟电路	24VDC（－ 15% ～ 20%），最大允许纹波电压 5%，180mA(来自主单元的外部电源或外接)
数字电路	3.3VDC 25mA（源于主模块的内部电源）

（2）电源消耗规格

最大消耗电流（3.3VDC/GND）	最大消耗电流（24VDC/COM）
25mA	180mA

（3）性能指标

项目	指标
测量范围	0 ～ 40mV（传感器特性 0 ～ 4mV /V）
传感器特性	使用本模块输出激励时最大支持 4mV /V； 使用外部激励时只需满足信号电压不超过 40mV 即可；
传感器最小阻抗	42Ω（单通道）（多通道时按电阻并联法计算）
激励电源	本模块可提供的激励电源：10V，（最大可提供 240mA）
AD 位数	16bit
采样周期	2ms/ 通道（同时开多个通道时为 25ms/ 通道）
精度	±0.2% 全量程
隔离	模拟电路和数字电路之间通过数字隔离器进行隔离；模拟电路与模块输入 24VDC 电源内部隔离；模拟通道之间不隔离。

12.6.2 输入信号的接线

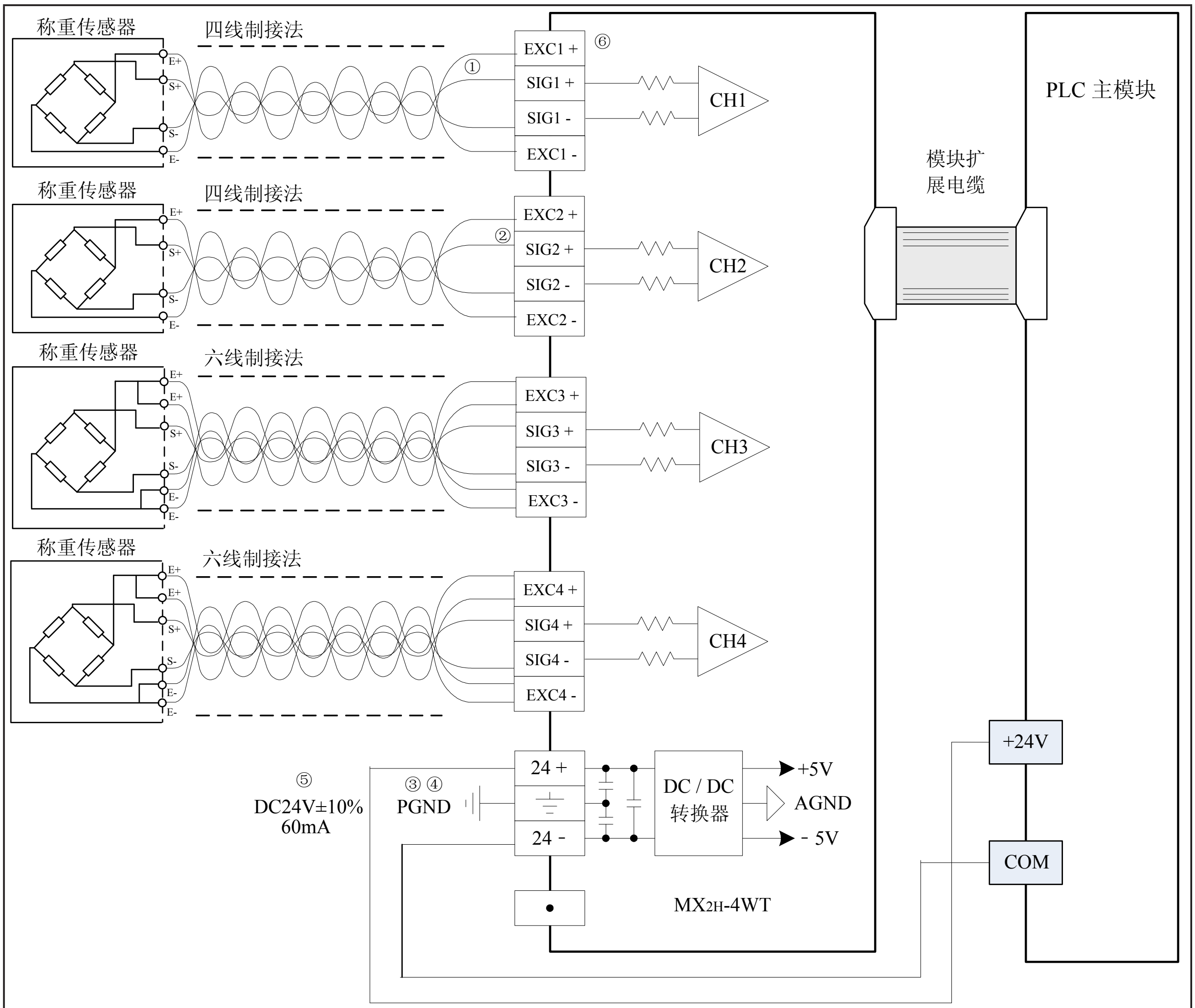


图 25 MX2H-4WT 模块输入信号接线示意图

图中的①～⑥表示布线时必须注意的 6 个方面：

①传感器信号建议通过屏蔽电缆连接电缆接入。电缆应远离电源线或其他可能产生电气干扰的电线。使用长的连接电缆

容易受到噪声的干扰，建议使用长度小于 100 米的连接电缆。连接电缆存在阻抗，会引入测量误差，通道校正可解决此问题。

②建议将不使用的通道关闭，以防止在这个通道上检测出错误的数据。

注：当选择 1 个通道时，采样周期为 2ms；同时选择多个通道时，由于 mV 信号的放大倍数较大，切换通道需要一段稳定时间，采样周期为 25ms。如果需要高速，则只能设一路。

③在现场接地良好的情况下，将模块的接地端 PG 良好连接现场地，否则不建议将 PG 连接，可能将会干扰引入模块。

④在现场 PG 接地良好的情况下，如果传感器电缆上存在过多的电气干扰，请将屏蔽线的屏蔽层与模块接地端相连接。

24V 供电电源可以使用主模块的辅助输出 24VDC 电源，也可以使用其它满足的电源。

⑥模块提供激励电源为 10V。

12.6.3 缓存区（BFM）

MX2H-4WT 与主模块之间通过缓存区（BFM）交换信息，主模块通过 Un.b 元件与 MX2H-4WT 缓存区（BFM）直接映射；n：模拟量模块序号；b：BFM 序号。在系统组态中系统会自动分配其地址，模拟量模块会自动排序，两个模拟量模块之间有 I/O 模块时不参与分配地址。例如，U1.0 对应模拟量模块地址为 1，BFM#0 的单元。主模块通过 FROM/TO 指令访问扩展模块内寄存器的 BFM 单元，也可使用 “MOV ×× Un.b” 访问扩展模块内寄存器的 BFM 单元。MX2H-4WT 的缓存区（BFM）具体内容见表 2。

表 9 MX2H-4WT 的缓存区（BFM）内容

Un.b	内容	备注	读写属性
Un.0	通道模式字 0	缺省值：16#FFFF	RW
Un.2	CH1 平均值点数	缺省值：8	RW
Un.3	CH2 平均值点数	缺省值：8	RW
Un.4	CH3 平均值点数	缺省值：8	RW
Un.5	CH4 平均值点数	缺省值：8	R/W
Un.10	CH1 当前毛重	缺省值：0	R
Un.11	CH2 当前毛重	缺省值：0	R
Un.12	CH3 当前毛重	缺省值：0	R
Un.13	CH4 当前毛重	缺省值：0	R
Un.20	CH1 当前净重	缺省值：0	R
Un.21	CH2 当前净重	缺省值：0	R
Un.22	CH3 当前净重	缺省值：0	R
Un.23	CH4 当前净重	缺省值：0	R
Un.29	错误状态字	缺省值：0	R
Un.30	CH1 皮重	缺省值：0	RW
Un.31	CH2 皮重	缺省值：0	RW
Un.32	CH3 皮重	缺省值：0	RW
Un.33	CH4 皮重	缺省值：0	RW
Un.40	CH1 工作状态字	缺省值：0	R
Un.41	CH2 工作状态字	缺省值：0	R
Un.42	CH3 工作状态字	缺省值：0	R
Un.43	CH4 工作状态字	缺省值：0	R
Un.50	CH1 校正 1 点的砝码重量	缺省值：0	RW
Un.51	CH1 校正 2 点的砝码重量	缺省值：0	RW
Un.52	CH1 校正 3 点的砝码重量	缺省值：0	RW
Un.53	CH1 校正 4 点的砝码重量	缺省值：0	RW

Un.b	内容	备注	读写属性
Un.54	CH1 校正 5 点的砝码重量	缺省值：0	RW
Un.55	CH1 校正 6 点的砝码重量	缺省值：0	RW
Un.56	CH1 校正命令	缺省值：0	RW
Un.60	CH2 校正 1 点的砝码重量	缺省值：0	RW
Un.61	CH2 校正 2 点的砝码重量	缺省值：0	RW
Un.62	CH2 校正 3 点的砝码重量	缺省值：0	RW
Un.63	CH2 校正 4 点的砝码重量	缺省值：0	RW
Un.64	CH2 校正 5 点的砝码重量	缺省值：0	RW
Un.65	CH2 校正 6 点的砝码重量	缺省值：0	RW
Un.66	CH2 校正命令	缺省值：0	RW
Un.70	CH3 校正 1 点的砝码重量	缺省值：0	RW
Un.71	CH3 校正 2 点的砝码重量	缺省值：0	RW
Un.72	CH3 校正 3 点的砝码重量	缺省值：0	RW
Un.73	CH3 校正 4 点的砝码重量	缺省值：0	RW
Un.74	CH3 校正 5 点的砝码重量	缺省值：0	RW
Un.75	CH3 校正 6 点的砝码重量	缺省值：0	RW
Un.76	CH3 校正命令	缺省值：0	RW
Un.80	CH4 校正 1 点的砝码重量	缺省值：0	RW
Un.81	CH4 校正 2 点的砝码重量	缺省值：0	RW
Un.82	CH4 校正 3 点的砝码重量	缺省值：0	RW
Un.83	CH4 校正 4 点的砝码重量	缺省值：0	RW
Un.84	CH4 校正 5 点的砝码重量	缺省值：0	RW
Un.85	CH4 校正 6 点的砝码重量	缺省值：0	RW
Un.86	CH4 校正命令	缺省值：0	RW
Un.90	CH1 传感器电压	缺省值：10000 （mV）	RW
Un.91	CH1 传感器输出灵敏度	缺省值：2（mV/V）	RW
Un.92	CH1 传感器量程	缺省值：5000 表示传感器最大量程	RW
Un.100	CH2 传感器电压	缺省值：10000 （mV）	RW
Un.101	CH2 传感器输出灵敏度	缺省值：2（mV/V）	RW
Un.102	CH2 传感器量程	缺省值：5000 表示传感器最大量程	RW
Un.110	CH3 传感器电压	缺省值：10000 （mV）	RW
Un.111	CH3 传感器输出灵敏度	缺省值：2（mV/V）	RW
Un.112	CH3 传感器量程	缺省值：5000 表示传感器最大量程	RW
Un.120	CH4 传感器电压	缺省值：10000 （mV）	RW
Un.121	CH4 传感器输出灵敏度	缺省值：2（mV/V）	RW
Un.122	CH4 传感器量程	缺省值：5000 表示传感器最大量程	RW
Un.193	恢复出厂设置	缺省值：0 由 0 变 1 时，恢复出厂设置	RW
Un.194	模块参数保存	缺省值：0 由 0 变 1 时，进行参数保存	RW
Un.198	模块软件版本信息	16#1000	R
Un.199	模块的识别码	16#8020	R

说明：

1．输入模式选择由 Un.0 中的 4 位十六进制数字 16#×4×3×2×1 控制。×1 控制通道 1，×2 控制通道 2，×3 控制通道 3，×4 控制通道 4。字符值所表示的信息如表 10 所示。

表 10 字符值信息表

×	状态信息
0	打开通道
F	关闭通道

举例：

若对 Un.0 单元写入 “16#FFF0”，将完成如下设置：

通道 1 打开；通道 2、通道 3、通道 4 关闭。

2．Un.2 ～ Un.5 作为通道的平均值点数的设定缓存区，提供 2、4、8、16、32、64、128、256 供用户选择。缺省值为 8。例如，平均值点数设为 4，即求 4 次采样值的平均值。

3．Un.10 ～ Un.13 表示当前称重的重量。

4．Un.20 ～ Un.23 表示当前称重的重量去掉皮重后的净重。

5．Un.29 是错误状态字，为 1 时，代表 24V 错误；为 0 时，代表没错误。

6．Un.30 ～ Un.33 表示皮重，允许直接设定或通过标定命令获得，范围 0 ～ 32,767。

7．Un.40 ～ Un.43：通道工作状态字。

位	定义	状态
Bit0	校正零点	0：零点未校正或校正不成功 1：零点校正成功
Bit1	校正 1 点	0：1 点未校正或校正不成功 1：第 1 点已校正成功
Bit2	校正 2 点	0：2 点未校正或校正不成功 1：第 2 点已校正成功
Bit3	校正 3 点	0：3 点未校正或校正不成功 1：第 3 点已校正成功
Bit4	校正 4 点	0：4 点未校正或校正不成功 1：第 4 点已校正成功
Bit5	校正 5 点	0：5 点未校正或校正不成功 1：第 5 点已校正成功
Bit6	校正 6 点	0：6 点未校正或校正不成功 1：第 6 点已校正成功
Bit7	标定皮重	0：皮重未标定或标定不成功 1：皮重标定成功
Bit8	断线	0：未断线 1：断线
Bit9	校正值错误	0：校正值无错误 1：校正值错误
其余	保留	

8．Un.56、Un.66、Un.76、Un.86：校正命令

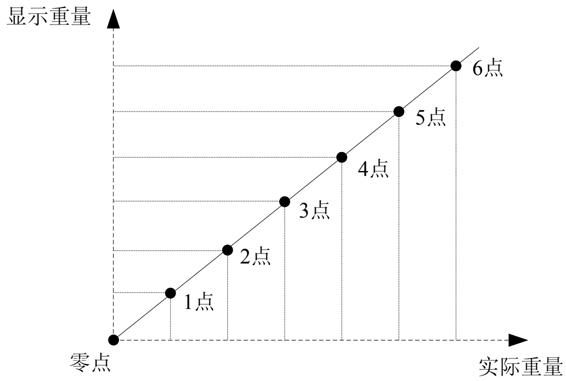
位	定义	状态
Bit0	校正零点	0→1 的上升沿：启动零点校正
Bit1	校正 1 点	0→1 的上升沿：启动第 1 点校正
Bit2	校正 2 点	0→1 的上升沿：启动第 2 点校正
Bit3	校正 3 点	0→1 的上升沿：启动第 3 点校正
Bit4	校正 4 点	0→1 的上升沿：启动第 4 点校正
Bit5	校正 5 点	0→1 的上升沿：启动第 5 点校正
Bit6	校正 6 点	0→1 的上升沿：启动第 6 点校正
Bit7	标定皮重	0→1 的上升沿：将当前重量标定为皮重
Bit8	清除校正错误	0→1 的上升沿：清除校正错误标志（会同时清掉所有校正值）
Bit9	清除校正	0→1 的上升沿：清除所有校正值
Bit10	传感器电源选择	0：本模块提供 1：外部提供
其余	保留	

9．Un.50 ～ Un.55、Un.60 ～ Un.65、Un.70 ～ Un.75、Un.80 ～ Un.85：校正参数区域。每个通道允许设定 6 个校正点，校正点用于写入当前实际放在称上的砝码重量。

写入校正点的砝码重量时，需注意以下几点：①所有写入的数值必须不大于 32,767；② 6 个校正点可仅适用其中部分校正点；③ 6 个校正点的重量必须从小到大依次设定，如下图所示；④每个通道的所有校正点的数据单位必须统一，可设定为（t、0.1t、0.01t、kg、0.1kg、0.01kg、g、0.1g、0.01g），当前重量（毛重、净重、皮重）都依据校正点的

数据单位显示。

校正操作过程如下，当秤上没有放置砝码时，使用校正零点命令，进行零点校正；然后再进行其他校正点的校正；将选定的校正砝码放置在秤上，在相应校正点的 BFM 单元写入砝码重量，再写入校正命令（设置 Un.56、Un.66、Un.76、Un.86 的相应位），提示校正成功后（查看 Un.40 ～ Un.43 的相应位），进行下一个校正点的校正。所有校正点完成后，利用参数保存命令（Un.194）将校正参数保存到模块内部。



10．Un.90 ～ Un.92、Un.100 ～ Un.102、Un.110 ～ Un.112、Un.120 ～ Un.122：传感器参数区域，当秤只进行了零点校正，不方便通过其他砝码校正时，模块可通过直接设定传感器参数来工作；当模块进行了 6 点校正中的任一点校正时，当前重量（毛重、净重）都依据校正数据显示，当模块没有进行 6 点校正中的任一点校正时，当前重量（毛重、净重）都依据传感器量程数据显示。例如：传感器最大量程为 500kg，设定量程数据为 500 时，当前重量（毛重、净重）显示的单位就为 1kg 而传感器数据设定为 5000 时，当前重量显示的单位就为 0.1kg。

11．Un.193 恢复出厂值，初始化模块。

12．Un.198 作为模块软件版本信息的缓存区。

13．Un.199 作为模块识别码的缓存区。MX2H-4WT 的识别码是 16#8020。PLC 中的用户程序可以在程序中使用这个识别码，在传输 / 接收数据之前确认此模块。

12.6.4 应用示例

例 1：系统组态配置

MX2H-4WT 模块地址为 0，将其第 1 通道打开，传感器灵敏度采用默认值 2（例如 1 个 50Kg 的传感器，灵敏度为 2，激励电压是 10V，则在 50Kg 的拉力作用下，输出端的压力变化就是 2*10=20mV），传感器电压不设置（默认采用外部电压供给传感器），传感器量程设置为 5000（传感器的实际最大量程所对应的数字量最大量程），平均值点数设置为 8；将其第 2、3、4 通道都关闭；为了实现上述配置，可以在 MXProgrammer 中“工程管理器 → 系统组态 → 硬件设置”中进行配置，如图 26 所示。

图 26 系统组态配置

例 2：指令配置

MX2H-4WT 模块地址为 0，通道模式和平均值点数与例 1 相同，通道特性采用默认值，校正零点到 6 点的砝码重量，校正完成之后进行质量测量，如图 27 所示。

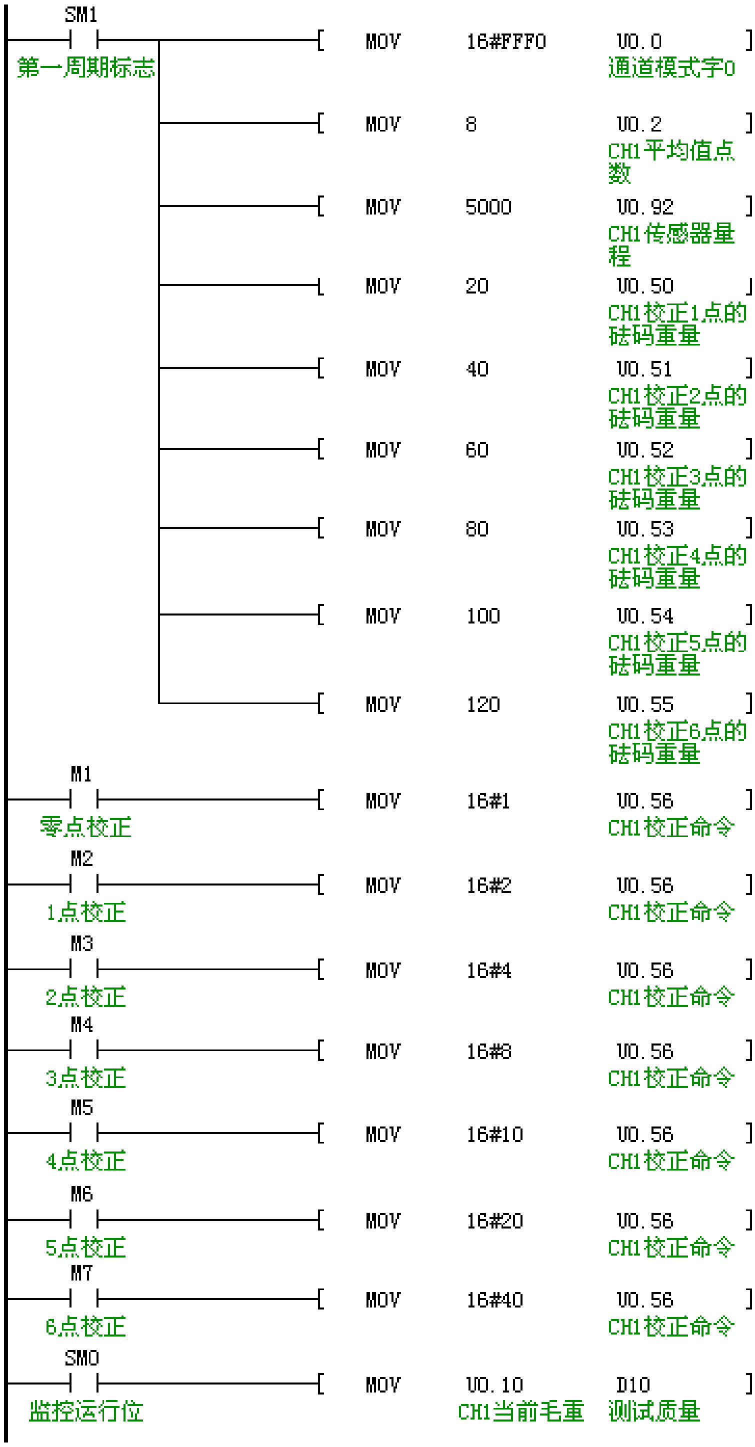


图 27　指令配置

☞ 注意

用系统组态和指令配置都可更改通道特性。当系统组态和指令同时配置且不相同时，以指令配置为准。

12.7 MX2H-2HC 高速计数输入模块

MX2H-2HC 是 MX2H 系列 PLC 扩展模块中的一种，支持 2 路 200K 的高速输入和 4 路普通输出，主要是对 MX2H 系列主模块高速输入计数器的补充。MX2H-2HC 的计数频率比 PLC 内置的高速计数器的计数频率高，而且它可直接进行比较和输出。高速输入作计数脉冲信号使用，普通输出做硬件置位使用。主模块通过 FROM/TO 指令访问本扩展模块内寄存器的 BFM 单元，也可使用“MOV ×× Un.b”访问扩展模块内寄存器的 BFM 单元。

各种计数模式可用 PLC 命令进行选择，如 1 相或 2 相，16 位或 32 位模式等。只有这些模式参数设定之后，MX2H-2HC 模块才能运行。

输入信号源必须是 1 相或 2 相编码器。可使用 5V、12V 电源，也可使用禁止计数输入信号（DIS）和复位输入信号（PRE）。输入配线有源型或漏型两种模式可供选择。

MX2H-2HC 硬件高速计数输入模块有 4 个输出点 YH0(CH1/ CH2)、YH1(CH1/ CH2)，当计数值与设定值相等时，相对输出将执行。输出晶体管是被单独隔离的。

12.7.1 电气规格

（1）电源消耗规格

最大消耗电流（3.3VDC/GND）	最大消耗电流（24VDC/GND）
110mA	0mA

（2）性能指标

表格一：

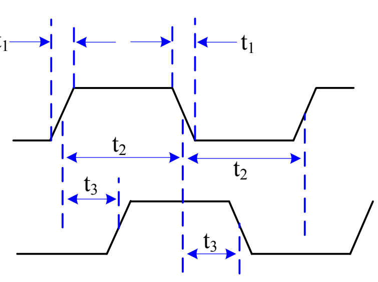
项目	A，B		PRE，DIS	
输入电压	12VDC (9.8 ～ 13.2V)	5VDC (4.5 ～ 5.5V)	12VDC ～ 24VDC (9.8 ～ 26.4V)	5VDC (4.5-5.5V)
输入电流（标准的）	8mA	7mA	8mA	7mA
ON 电压（最小）	9.8V	4.5V	9.8V	4.5V
OFF 电压（最大）	2.5V	1.5V	2.5V	1.5V

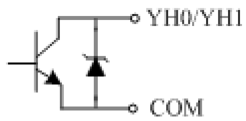
☞ 注意

1．计数器输入（A，B，PRE）相互隔离，和 I/O 总线也相互隔离并且反向极性保护。

2．A、B、PRE 和 DIS 的信号（+）端输入，仅能就它们支持的输入电压中选择一个使用。

表格二：

<table>
<tr><th colspan="2" rowspan="2">项目</th><th colspan="2">1 相输入</th><th colspan="3">2 相输入</th></tr>
<tr><th>1 个输入</th><th>2 个输入</th><th>1 边缘计数</th><th>2 边缘计数</th><th>4 边缘计数</th></tr>
<tr><td rowspan="3">输入信号</td><td>信号水平</td><td colspan="5">详见表格一</td></tr>
<tr><td>最大频率</td><td colspan="5">200KHZ</td></tr>
<tr><td>脉冲形状</td><td colspan="5">
t_1：上升下降时间≤0.8μs
t_2：ON/OFF脉冲宽度≥2.5μs
t_3：A相与B相的相位差≥1μs
复位输入（PRE）：输入脉冲宽度≥50μs
禁止计数输入（DIS）：输入脉冲宽度≥50μs</td></tr>
</table>

<table>
<tr><th colspan="2" rowspan="2">项目</th><th colspan="2">1 相输入</th><th colspan="3">2 相输入</th></tr>
<tr><th>1 个输入</th><th>2 个输入</th><th>1 边缘计数</th><th>2 边缘计数</th><th>4 边缘计数</th></tr>
<tr><td rowspan="3">计数特性</td><td>格式</td><td colspan="5">自动 UP/DOWN（当为 1 相 1 输入模式时，UP/DOWN 由 PLC 命令或输入端子决定）</td></tr>
<tr><td>范围</td><td colspan="5">当使用 32 位时：-2,147,483,648 ～ +2,147,483,647
当使用 16 位时：0 ～ 65,535（上限可由用户指定）</td></tr>
<tr><td>比较类型</td><td colspan="5">当计数器的当前值与比较值（由 PLC 传送）相匹配时，每个输出被设置，而且 PLC 的复位命令可将其转向 OFF 状态。
YH：由硬件处理的直接输出</td></tr>
<tr><td rowspan="2">输出信号</td><td>输出类型</td><td colspan="5">YH0：YH 的晶体管输出
YH1：YH 的晶体管输出
COM：晶体管输出

</td></tr>
<tr><td>输出容量</td><td colspan="5">5 ～ 24VDC，0.5A</td></tr>
<tr><td colspan="2">基本单元供电</td><td colspan="5">3.3V 50mA（由主单元或有源扩展单元提供的内部单元供电）</td></tr>
</table>

12.7.2 输入输出信号的接线

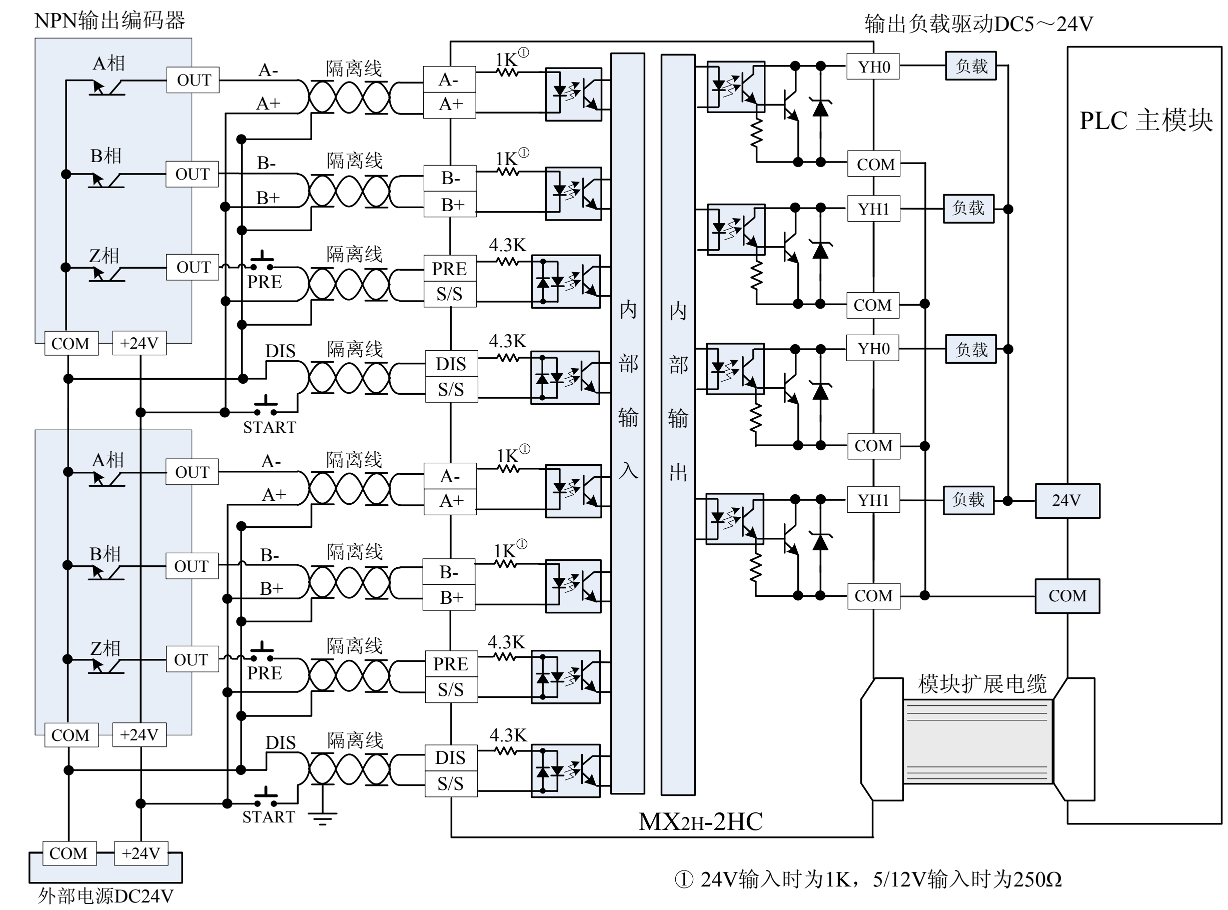


图 25 MX2H-2HC 模块输入输出信号接线示意图

☞ 注意

1. 对于上图中的“⏚”表示连接 MX2H 的主模块上的接地端子。
2. 若使用 PNP 输出型的编码器，请注意与 MX2H-2HC 模块输入配线的极性。

12.7.3 缓存区（BFM）

MX2H-2HC 与主模块之间通过缓存区（BFM）交换信息，主模块通过 Un.b 元件与 MX2H-2HC 缓存区（BFM）直接映射；n：特殊功能模块序号；b：BFM 序号。在系统组态中系统会自动分配其地址，特殊功能模块会自动排序，两个特殊功能模块之间有 I/O 模块时不参与分配地址。例如，U1.0 对应模块地址为 1，BFM#0 的单元。主模块通过 FROM/TO 指令访问扩展模块内寄存器的 BFM 单元，也可使用“MOV ×× Un.b”访问扩展模块内寄存器的 BFM 单元。MX2H-2HC 的缓存区（BFM）具体内容见表 9。

表 9 MX2H-2HC 的缓存区（BFM）内容

Un.b		内容	备注	读写属性
CH1	CH2			
Un.0	Un.30	计数模式 0 到 11	缺省值：0	R/W
Un.1	Un.31	DOWN/UP 命令（1 相 1 输入模式）	缺省值：0	R/W
Un.3，Un.2	Un.33，Un.32	环长度高 / 低位	缺省值：65536	R/W
Un.4	Un.34	命令	缺省值：0	R/W
Un. 6，Un. 5	Un.36，Un.35	预设置数据高 / 低位	缺省值：0	R/W
Un.8，Un.7	Un.38，Un.37	YH0 比较值高 / 低位	缺省值：32767	R/W
Un.10，Un.9	Un.40，Un.39	YH1 比较值高 / 低位	缺省值：32767	R/W
Un.12，Un.11	Un.42，Un.41	最大计数值高 / 低位	缺省值：0	R
Un.14，Un.13	Un.44，Un.43	最小计数值高 / 低位	缺省值：0	R
Un.16，Un.15	Un.46，Un.45	计数器当前值高 / 低位	缺省值：0	R/W
Un.17	Un.47	比较结果	缺省值：0	R
Un.18	Un.48	端子状态	缺省值：0	R
Un.19	Un.49	错误状态	缺省值：0	R
Un.21，Un.20	Un.51，Un.50	上升沿捕获的计数值	缺省值：0	R
Un.23，Un.22	Un.53，Un.52	下升沿捕获的计数值	缺省值：0	R
Un.24 ～ Un.28	Un.54 ～ Un.58	保留		
Un.29		错误状态字	缺省值：0	R
Un.198		模块软件版本信息		R
Un.199		模块的识别码	16#5010	R

说明：

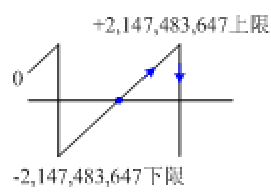
1．Un.0，Un.30 计数模式

计数模式		32 位	16 位
2 相输入（相位差脉冲）	1 边缘计数	0	1
	2 边缘计数	2	3
	4 边缘计数	4	5
1 相 2 输入（加 / 减脉冲）		6	7
1 相 1 输入	硬件上 / 下	8	9
	软件上 / 下	10	11

计数模式由 PLC 进行选择，当有数据写到 Un.0 或 Un.30 时，Un.1 到 Un.19 或 Un.31 到 Un.49 的值复位成默认值。

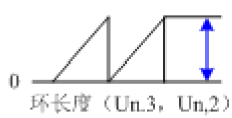
（1）32 位计数模式

32 位计数模式的计数范围为 -2147483648 ～ 2147483647，当发生溢出时，进行 UP/DOWN 计数的 32 位计数器由下限改变成上限，或由上限改变成下限。上限和下限都是固定值，上限值为 +2147483647，下限值为 -2147483648。



（2）16 位计数器模式

16 位计数器只处理 0 ～ 65536 的正数值。当发生溢出时，计数值会由上限值变为 0 或 0 变为上限值，其中 CH1 的上限值由（U0.3，U0.2）决定，CH2 的上限值由（U0.7，U0.6）决定。



（3）1 相 1 输入计数器（8 到 11）

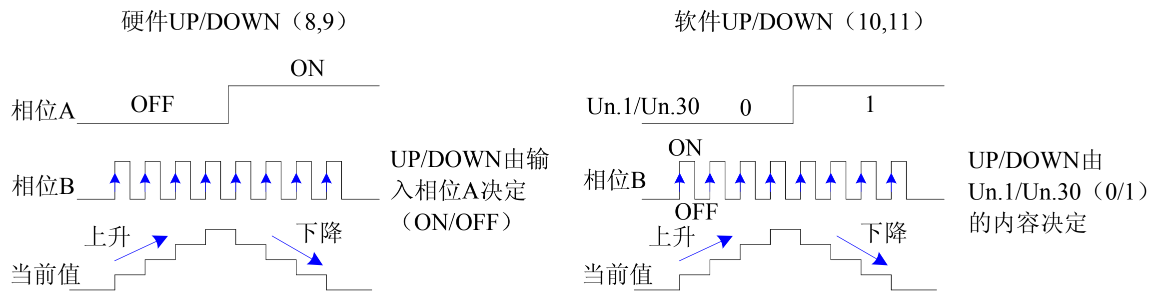


图 26 1 相 1 输入计数时序图

（4）1 相 2 输入计数器（6，7）

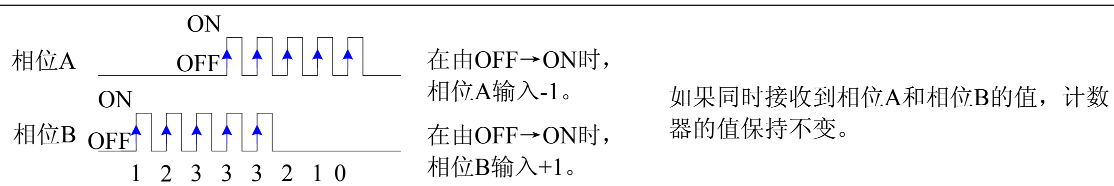


图 27 1 相 2 输入计数时序图

（5）2 相计数器（0 到 5）

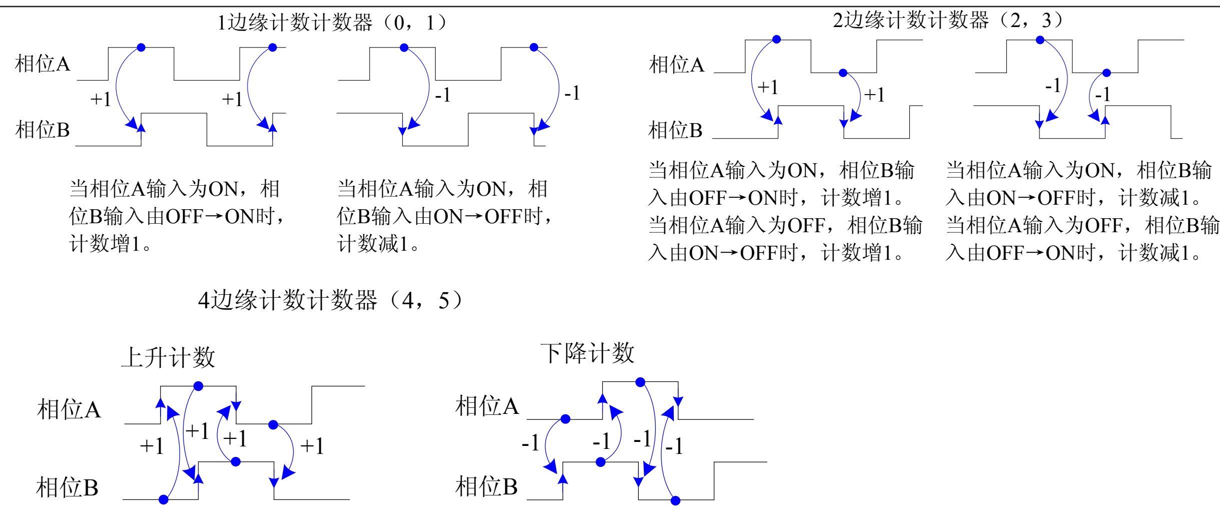


图 28 2 相计数时序图

2．Un.1，Un.31 计数方向

（1）Un.1 对应 CH1 的计数方向，Un.31 对应 CH2 的计数方向。

（2）当 Un.1=0 或 Un.31=0 时，对应的通道软件计数方向向上；当 Un.1=1 或 Un.31=1 时，对应的通道软件计数方向向下。

3．（Un.3，Un.2），（Un.33，Un.32）环长度

（1）（Un.3，Un.2）对应 CH1 的环长度，（Un.33，Un.32）对应 CH2 的环长度。

（2）只针对 16 位计数器模式，默认值为 65536。当工作为 16 位模式下，计数范围为 0 ～ 65,536。当发生溢出时，计数值由 0 变为上限值（即为环长度设定值）或由上限值变为 0。

4．Un.4，Un.34 命令

Un.4 对应 CH1 的命令设置，Un.34 对应 CH2 的命令设置。

Un.4，Un.34	“0”（OFF）	“1”（ON）
Bit0	计数禁止	计数允许
Bit1	YH0 输出禁止	YH0 输出允许
Bit2	YH1 输出禁止	YH1 输出允许
Bit3	预先复位禁止	预先复位允许

Un.4，Un.34	“0”（OFF）	“1”（ON）
Bit4	无动作	YH0 输出置位
Bit5	无动作	YH1 输出置位
Bit6	无动作	YH0 输出复位
Bit7	无动作	YH1 输出复位
Bit8	无动作	错误标志复位
Bit9	PRE 电平有效（默认）	PRE 边沿有效
Bit10	PRE 上升沿有效（默认）	PRE 下降沿有效
Bit11	普通模式（DIS 作为禁止计数功能）（默认）	色标模式（DIS 作为边沿捕获计数值功能）
Bit12 ～ Bit15	未定义	

（1）Bit0 设置为 ON 时，并且 DIS 输入端子为 OFF 时，计数器被允许开始计数输入脉冲。
（2）当 Bit1 设置为 ON 时，YH0 输出允许。
（3）当 Bit2 设置为 ON 时，YH1 输出允许。
（4）当 Bit3 设置为 OFF 时，PRE 输入端子的预先设置功能失去作用。
（5）当 Bit4 设置为 ON 时，YH0 输出被设置为 ON。
（6）当 Bit5 设置为 ON 时，YH1 输出被设置为 ON。
（7）当 Bit6 设置为 ON 时，YH0 输出被复位。
（8）当 Bit7 设置为 ON 时，YH1 输出被复位。
（9）当 Bit8 设置为 ON 时，所有的错误标志被复位。

5．（Un.6，Un.5），（Un.36，Un.35）预设值

（1）（Un.6，Un.5）对应 CH1 的预设值，（Un.36，Un.35）对应 CH2 的预设值。
（2） 在 16 位计数模式下，当 PLC 设置预设值时，Un.5 填充为 0。
（3） 当 PRE 信号变为高电平且 Un.4 或 Un.34 中的允许预设已为 ON 时，计数器就从预设值开始计数。

6．（Un.8，Un. 7），（Un.10，Un. 9），（Un.38，Un. 37），（Un.40，Un. 39）比较值

（1）（Un.8，Un. 7），（Un.10，Un. 9）对应 CH1 的输出比较值，（Un.38，Un. 37），（Un.40，Un. 39）对应 CH2 的输出比较值。
（2） 当对计数值的当前值和（Un.8，Un. 7），（Un.10，Un. 9）中的值进行比较后，对应的通道就会输出比较结果。
（3） 在 16 位计数模式下，当 PLC 设置比较时，比较值的高位填充为 0。
（4） 当前计数值等于比较值且比较输出允许时，就置位比较输出并保持，当收到复位比较输出命令时才复位比较输出（当复位比较输出后，把 Un.4，Un.34 命令中的 Bit6 或 Bit7 也复位成 0）。
（5） 当通过写计数器当前值或通过复位预设值使计数器值等于比较值时，不置位比较输出。

7．（Un.12，Un. 11），（Un.42，Un. 41）最大计数值

（1）（Un.12，Un. 11）对应 CH1 的最大计数，（Un.,42，Un. 41）对应 CH2 的最大计数。
（2） 存储计数器所能达到的最大计数值。
（3） 如果掉电，存储的数据被清除。

8．（Un.14，Un. 13），（Un.44，Un. 43）最小计数值

（1）（Un.14，Un. 13）对应 CH1 的最小计数，（Un.44，Un. 43）对应 CH2 的最小计数。
（2） 存储计数器所能达到的最小计数值。
（3） 如果掉电，存储的数据被清除。

9．（Un.16，Un. 15），（Un.46，Un. 45）计数器当前值

（1）（Un.16，Un. 15）对应 CH1 的计数器当前值，（Un.46，Un. 45）对应的 CH2 的计数器当前值。
（2） 计数器的当前值可通过 PLC 进行读操作。
（3） 在高速运行时，它并不是准确的值，因为存在通信延迟。
（4） 计数器的当前值可通过可编程控制器的写命令改变。

10．Un.17，Un.47 比较状态

（1） Un.17 对应 CH1 的比较状态，Un.47 对应 CH2 的比较状态。
（2） 比较状态为只读，可编程控制器的写命令对其不起作用。

Un.17，Un.47		“0” (OFF)	“1” (ON)
YH0	Bit0	设定值 ≤ 当前值	设定值＞当前值
	Bit1	设定值 ≠ 当前值	设定值＝当前值
	Bit2	设定值 ≥ 当前值	设定值＜当前值

Un.17，Un.47		“0” (OFF)	“1” (ON)
YH1	Bit3	设定值 ≤ 当前值	设定值＞当前值
	Bit4	设定值 ≠ 当前值	设定值＝当前值
	Bit5	设定值 ≥ 当前值	设定值＜当前值

11．Un.18，Un.48 端子状态

Un.18，Un.48	“0” （OFF）	“1” （ON）
Bit0	预先复位输入为 OFF	预先复位输入为 ON
Bit1	禁止计数输入为 OFF	禁止计数输入为 ON
Bit2	YH0 输出为 OFF	YH0 输出为 ON
Bit3	YH1 输出为 OFF	YH1 输出为 ON
Bit4 ～ Bit15	未定义	

12． Un.19，Un.49 错误状态

Un.19，Un.49	错误状态	
Bit0	当环长度写错（不是 0 ～ 65536），被设置	
Bit1	当预设值写错，被设置	在 16 位计数器模式下， 当前值 > 环长度
Bit2	当比较值写错，被设置	
Bit3	当当前值写错，被设置	
Bit4	当计数器超出上限时，被设置	当超出 32 位计数器上限值或下限值时
Bit5	当计数器超出下限时，被设置	
Bit6	当计数器模式写错（超出 0 ～ 11），被设置	
Bit7	当 BFM 号写错（超出 0 ～ 19，或 30 ～ 49），被设置	
Bit8 ～ Bit15	未定义	

13．Un.29 是错误状态字，为 1 时，代表 24V 错误；为 0 时，代表没错误。

12.7.4　应用示例

例 1：系统配置组态配置（以 CH1 通道为例）

MX2H-2HC 模块地址为 0，使用其第 1 通道高速输入信号，计数模式设置为 1 相 1 输入软件计数，软件计数方向向上，环长度为 1000，YH0 比较输出值为 200，YH1 比较输出值为 500，选择计数控制，YH0 输出控制，YH1 输出控制。选择错误标志复位有效。

若要满足上述功能，可在 MXProgrammer 中“工程管理器→系统组态→硬件设置”中进行配置，通道参数配置如图 29 所示。

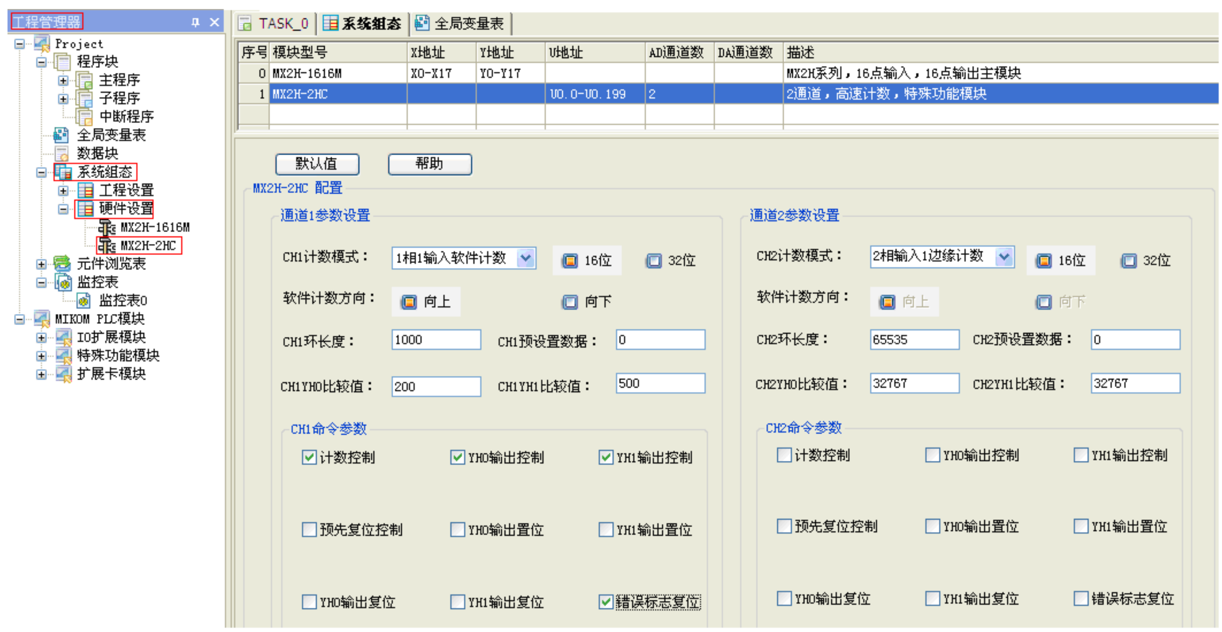


图 29 系统组态配置

例 2：指令配置（以 CH1 通道为例）

MX2H-2HC 模块地址为 0，通道设置与例 1 相同，如图 30 所示。

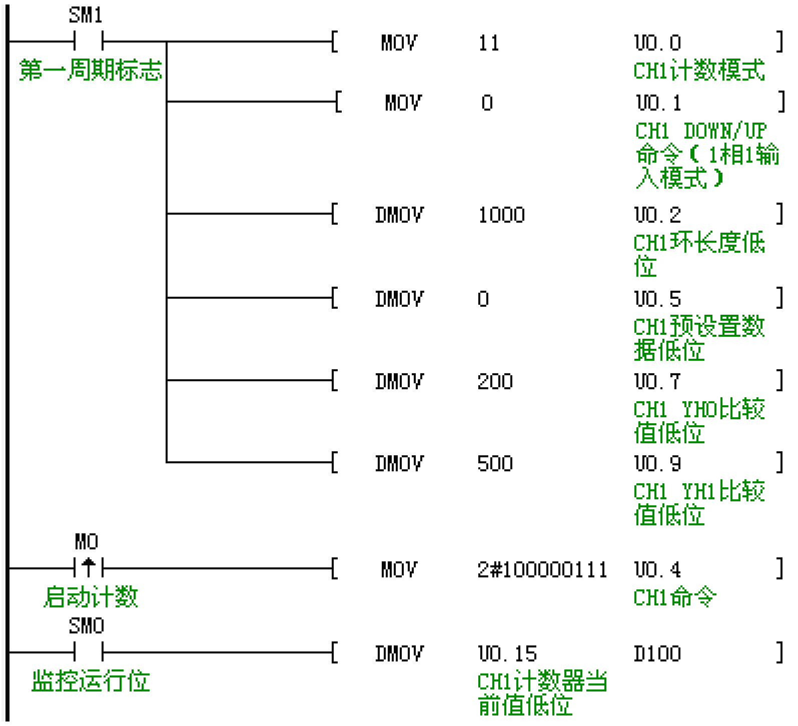


图 30 指令配置

☞ 注意

用系统组态和指令配置都可更改通道特性。当系统组态和指令同时配置且不相同时，以指令配置为准。

12.8 MX2H-2PG 运动控制模块

MX2H-2PG 是 MX2H 系列 PLC 扩展模块中的一种，支持 2 路 200K 的高速输出和 2 路普通输出，主要是对 MX2H 系列主模块高速输出的补充。高速输出作脉冲输出使用，普通输出做脉冲的方向信号和清零信号使用。主模块通过 FROM/TO 指令访问本扩展模块内寄存器的 BFM 单元，也可使用“MOV ×× Un.b”访问扩展模块内寄存器的 BFM 单元。

12.8.1 电气规格

（1）电源消耗规格

最大消耗电流（3.3VDC/GND）	最大消耗电流（24VDC/GND）
140mA	10mA

（2）输入特性与信号规格

项目		信号输入端 X0、X1
信号输入方式		源型 / 漏型方式，用户可通过 S/S 进行选择
电气参数	检测电压	DC24V
	输入阻抗	4.3KΩ
	输入 ON	外部回路电阻小于 400Ω（端口电压大于 18V）
	输入 OFF	外部回路电阻大于 24kΩ（端口电压小于 4V）

项目	信号输入端 X0、X1
公共接线端	只有一个公共端，为 S/S

输入响应时间：
PG：脉宽不小于 50ms；DG：不大于 1ms；STR、X0、X1：不大于 0.1ms。
输出 CLR：当归零操作完成时为 ON，输出脉冲 20ms。
（3）输出端口规格

项目	指标
输出方式	门极开路的晶体管输出
回路电源电压	5 ～ 24V
最大回路电源电压	30V
回路电流	0.3A 24VDC
开路时漏电流	小于 0.1mA/30VDC
最小负载	5mA（5 ～ 24VDC）

12.8.2 输入输出信号的接线

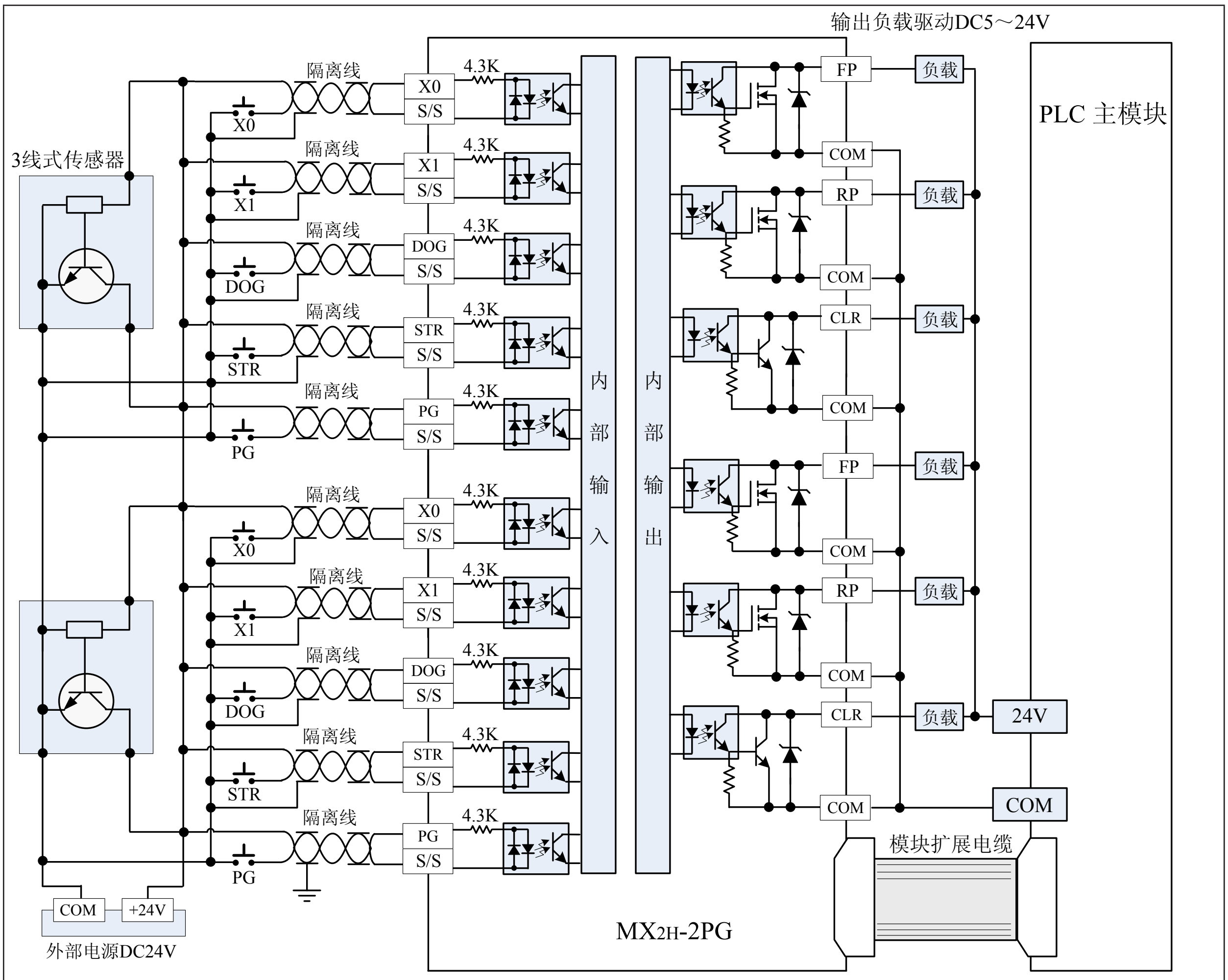


图 31 MX2H-2PG 模块输入信号接线示意图

☞ 注意

1．对于上图中的“⏚”表示连接 MX2H 的主模块上的接地端子。

2．若使用 PNP 输出型的传感器，请注意与 MX2H-2PG 模块输入配线的极性。

12.8.3 缓存区（BFM）

MX2H-2PG 与主模块之间通过缓存区（BFM）交换信息，主模块通过 Un.b 元件与 MX2H-2PG 缓存区（BFM）直接映射；n：特殊功能模块序号；b：BFM 序号。在系统组态中系统会自动分配其地址，特殊功能模块会自动排序，两个特殊功能模块之间有 I/O 模块时不参与分配地址。例如，U1.0 对应模块地址为 1，BFM#0 的单元。主模块通过 FROM/TO 指令访问扩展模块内寄存器的 BFM 单元，也可使用“MOV ×× Un.b”访问扩展模块内寄存器的 BFM 单元。MX2H-2PG 的缓存区（BFM）具体内容见表 10。

表 10 MX2H-2PG 的缓存区（BFM）内容

Un.b		内容	备注	读写属性
CH1	CH2			
Un.1，Un.0	Un.101，Un.100	最大速度高 / 低位	缺省值：200000，单位 Hz，范围：1 ～ 200000，指每个运行模式中的速度上限	R/W
Un.2	Un.102	偏置速度	缺省值：5000，单位 Hz，范围：1 ～ 65535，指每个运行模式中的速度下限	R/W
Un.4，Un.3	Un.104，Un.103	点动速度高 / 低位	缺省值：10000，单位 Hz，范围：1 ～ 200000，指点动操作过程中的速度	R/W
Un.6，Un.5	Un.106，Un.105	归零速度（高速）高 / 低位	缺省值：30000，单位 Hz，范围：1 ～ 200000，指设备处于归零运行中的高速返回速度	R/W
Un.7	Un.107	归零速度（爬行）	缺省值：5000，单位 Hz，范围：1 ～ 65535，指机器处于归零运行过程中，输入近点信号之后设定的低速运行速度	R/W
Un.8	Un.108	零点信号数量	缺省值：1，范围：0 ～ 32767，设定从 DOG 输入 ON 或 OFF（计数时间通过 Un.36 或者 Un.136 设定）一直到停止为止的零点信号的数量。当检测为上升沿时对零点信号进行计数。当设定值为“0”的时候，则会在 DOG 输入变为 ON 或者 OFF 时会从归零速度（低速）立即停止	R/W
Un.10，Un.9	Un.110，Un.109	零点地址高 / 低位	缺省值：0，范围：－2,147,483,648 ～ 2,147,483,467，当归零操作完成以后，该值被写入到当前地址	R/W
Un.11	Un.111	加速时间	缺省值：200，单位 ms，范围：1 ～ 5000，指从偏置速度达到最大速度所用的时间	R/W
Un.12	Un.112	减速时间	缺省值：200，单位 ms，范围：1 ～ 5000，指从最大速度降到偏置速度所用的时间	R/W
Un.14，Un.13	Un.114，Un.113	目标地址 1 高 / 低位	缺省值：0，范围：－2,147,483,648 ～ 2,147,483,467，指设定每个运行模式中的目标位置（绝对地址）或是移动距离（相对地址）	R/W
Un.16，Un.15	Un.116，Un.115	运行速度 1 高 / 低位	缺省值：100000，单位 Hz，范围：1 ～ 200000 指每个运行模式中的时间运行速度	R/W
Un.18，Un.17	Un.118，Un.117	目标地址 2 高 / 低位	缺省值：0，范围：－2,147,483,648 ～ 2,147,483,467 指第二速度定位模式中的目标位置（绝对地址）或是移动距离（相对地址）	R/W
Un.20，Un.19	Un.120，Un.119	运行速度 2 高 / 低位	缺省值：100000，单位 Hz，范围：1 ～ 100000 指第二速度定位模式和中断第二速度模式运行过程中的实际运行速度。	R/W
Un.23，Un.22	Un.123，Un.122	当前速度高位 / 低位	单位 Hz，当输出脉冲串的时候，保存运行速度的当前值	R

Un.b		内容	备注	读写属性
CH1	CH2			
Un.25，Un.24	Un.125，Un.124	当前地址高位 / 低位	保存当前地址数据，保存的地址数据始终为绝对地址	R/W
Un.26	Un.126	运行指令	缺省值：0	R/W
Un.27	Un.127	运行模式	缺省值：0	R/W
Un.28	Un.128	状态信息		R
Un.36	Un.136	运行参数	缺省值：0	R/W
Un.37	Un.137	出错代码	在出现错误的时候保存错误代码	R
Un.38	Un.138	端子信息		R
Un.60	Un.160	恢复出厂设置	当值从 0 → 16#FFFF 跳变时恢复当前通道参数到出厂设置状态 注意：只有在 PLC 主机处于 STOP 状态并且 MX2H-2PG 没有发脉冲串的时候才能使用此命令	R/W
Un.198		模块软件版本信息		R
Un.199		模块的识别码	16#6001	R

说明：

1．（Un.1，Un.0），（Un.101，Un.100）最大速度

（1）指所有速度的上限。

（2）将偏置速度、点动速度、归零速度（高速）、归零速度（爬行）、运行速度 1 和运行速度 2 都应该小于最大速度。

2．Un.2，Un.102 偏置速度

（1）指起始速度。

（2）加减速的斜度由最大速度、偏置速度、加速时间和减速时间共同决定。

（3）在配合步进电机使用时，偏置速度的设定需要考虑到共振区域和自行起动的频率。

3．（Un.4，Un.3），（Un.104，Un.103）点动速度

（1）指进行手动正向 / 反向操作过程中的速度。

（2）偏置速度＜点动速度＜最大速度。

（3）当点动速度＞最大速度或点动速度＜偏置速度时，报错，不发送脉冲。

4．（Un.6，Un.5），（Un.106，Un.105）归零速度（高速）

（1）指处于归零运行中的高速返回速度。

（2）偏置速度＜归零速度（高速）＜最大速度。

5．Un.7 ，Un.107 归零速度（爬行）

（1）指处于归零运行过程中，近点信号有效以后的低速返回的速度。

（2）在允许的范围内该值越小，则起始位置的精确度越高。

6．Un.8 ，Un.108 零点信号数量

（1）设定从 DOG 输入有效一直到停止为止的零点信号的数量。

（2）设定值为 0 的情况下，则会在 DOG 输入有效时立刻停止。

（3）如果设备因为紧急停止可能损坏，该数值可以设定为达到了爬行速度的安全速度以后才进行停止。

7．（Un.10，Un.9），（Un.110，Un.109）零点地址

（1）指归零操作完成时当前值的地址。

（2）当归零操作完成以后，该值被写入到当前地址。

8．Un.11，Un.111 加速时间

（1）指从偏置速度达到最大速度所用的时间。

（2）加速度的斜率取决于偏置速度、最大速度和加速时间。可以设定的范围为 1 ～ 5000ms。

（3）当设定值≤ 0ms 时，以 1ms 动作；当设定值≥ 5001ms 时，以 5000ms 动作。

9. （Un.14，Un.13），（Un.114，Un.113）目标地址 1

（1）指设定每个运行模式中的目标位置（绝对地址）或是移动距离（相对地址）。

（2）当使用到绝对地址的时候，定位就会按照当前地址所指定的位置进行定位。此时，运行方向取决于当前地址是大于还是小于目标地址。此时，运行方向取决于目标地址的符号。

10. （Un.16，Un.15），（Un.116，Un.115）运行速度 1

（1）指每个运行模式中的实际运行速度。

（2）偏置速度＜运行速度＜最大速度。

（3）当运行速度＞最大速度或运行速度＜偏置速度时，会以偏置速度运行。

11. （Un.18，Un.17），（ Un.118，Un.117）目标地址 2

（1）指设定在第二速度定位操作中的目标位置（绝对地址）或是移动距离（相对地址）。

（2）当使用到绝对地址的时候，定位就会按照当前地址所指定的位置进行定位。当使用到相对地址的时候，定位仅仅根据第一速度的目标地出发的移动距离进行定位。

12. （Un.20，Un.19），（Un.120，Un.119）运行速度 2

（1）指在第二速度定位运行和中断第二速度运行过程中的实际运行速度。

（2）偏置速度＜运行速度＜最大速度。

（3）当运行速度＞最大速度或运行速度＜偏置速度时，会以偏置速度运行。

（4） 当运行速度的更改处于允许的状态下，在定位中的速度允许被更改。

13. （Un.23，Un.22），（ Un.123，Un.122）运行速度当前值

（1）当输出脉冲串的时候，会保存运行速度的当前值。（当其停止或是处于手动脉冲发生器输入的时候，会变成“0”。）

14. （Un.25，Un.24），（Un.125，Un.124）当前地址

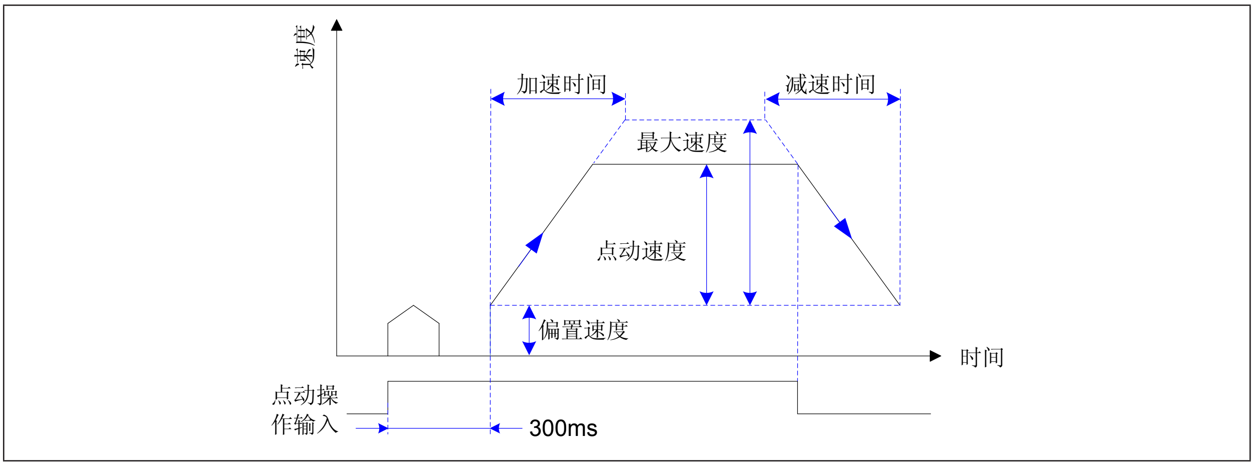
（1）保存当前地址数据。

（2）保存的地址数始终为绝对地址。

15．Un.26，Un.126 运行指令

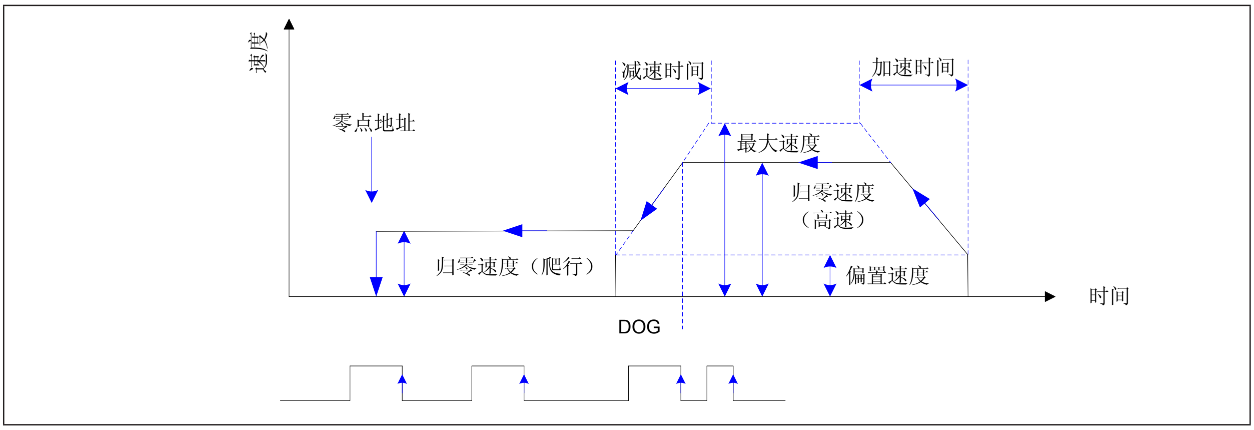
位	定义	说明
Bit0	出错复位	OFF → ON 变化时清除错误状态和错误代码
Bit1	STOP	在定位操作过程中通过该位来减速停止
Bit2	正向极限	在正向脉冲输出过程中通过该位来减速停止
Bit3	反向极限	在反向脉冲输出过程中通过该位来减速停止
Bit4	正向点动	为 ON 时输出正向脉冲
Bit5	反向点动	为 ON 时输出反向脉冲
Bit6	归零指令	OFF → ON 变化时启动机器归零
Bit7	数据设定型归零指令	OFF → ON 变化时输出 CLR 信号，零点地址的数据被传送给当前地址，并且归零完成标志位置 ON
Bit8	相对 / 绝对地址	为 OFF 时，通过绝对地址进行操作（离开地址值 0 的距离）； 为 ON 时，通过相对地址指令进行操作（离开当前地址的距离）
Bit9	START	OFF → ON 变化时启动运行模式中选择的定位操作，在该位变为 ON 之前请先选择运行模式
Bit10 ～ Bit15	保留	

（1）点动操作

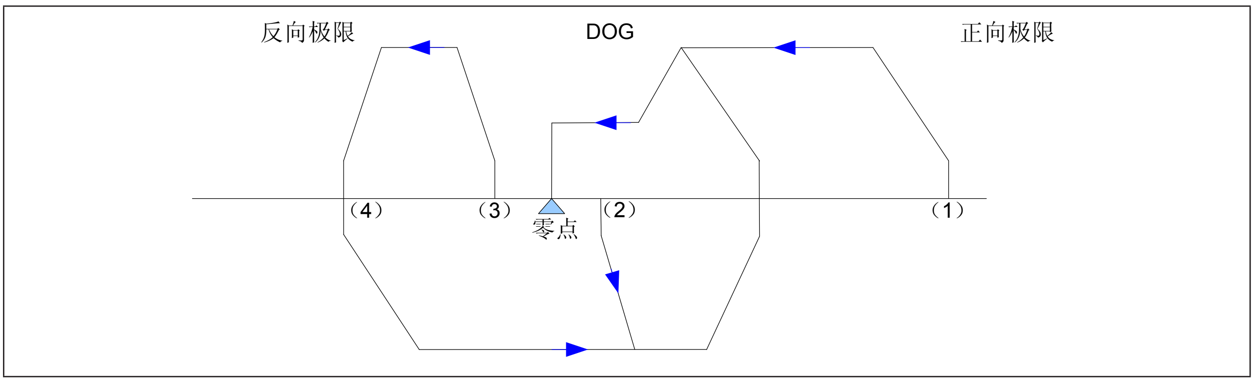


当正向 / 反向点动控制位（BFM#26/#126 的 Bit4，Bit5）为 ON 的时间小于 300ms 时，只会输出一个脉冲，当大于 300ms 时，脉冲串会持续输出，直到控制位变为 OFF，当正反极限为 ON 时，就开始减速停止。

（2）机械归零



归零操作的启动是通过归零命令（BFM#26，BFM#126 的 Bit6）的上升沿启动。在以归零速度（高速）按照归零方向移动后，当检测到工件到达 DOG 端子后减至归零速度（爬行），同时开始对零点信号计数，当计数到指定的数量后工件停止。DOG 搜索按照下面的曲线动作，下图上半部分箭头所指方向为归零方向。

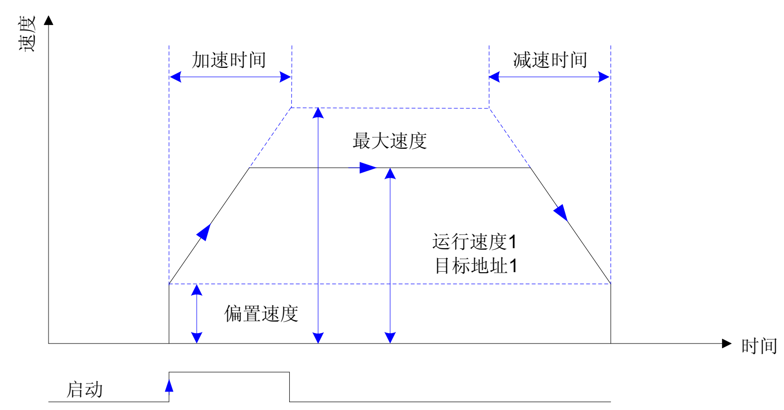


16．Un.27，Un.127 运行模式

位	定义	说明
Bit0	第一速度定位运行	为 ON 时，就选择了第一速度定位动作。Bit0 变为 ON，把相关数据写入到目标地址 1 和运行速度 1 中，然后使得 START 输入为 ON 或 START 标志位为 ON，此时定位开始。

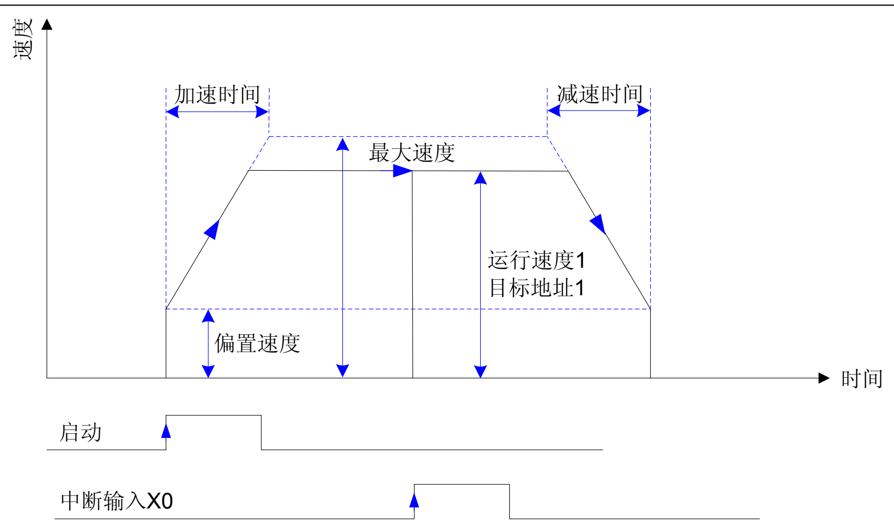
位	定义	说明
Bit1	中断第一速度定位运行	为 ON 时，就选择了中断第一速度定位动作。Bit1 变为 ON，把相关数据写入到目标地址 1 和运行速度 1 中，然后使得 START 输入为 ON 或 START 标志位为 ON，此时定位开始。使用 X0 作为中断输入。
Bit2	第二速度定位运行	为 ON 时，就选择了第一速度定位动作。Bit2 变为 ON，把相关数据写入到目标地址 1、运行速度 1、目标地址 2、运行速度 2 中，然后使得 START 输入为 ON 或 START 标志位为 ON，此时定位开始。
Bit3	中断第二速度定位运行	为 ON 时，就选择了中断第二速度定位运行。Bit3 变为 ON，把相关数据写入到目标地址 1、运行速度 1 和运行速度 2 中，然后使得 START 输入为 ON 或 START 标志位为 ON，此时定位开始。使用 X0、X1 作为中断输入。
Bit4	中断停止运行	为 ON 时，就选择了中断停止运行。Bit4 变为 ON，把相关数据写入到目标地址 1、运行速度 1 中，然后使得 START 输入为 ON 或 START 标志位为 ON，此时定位开始。使用 X0 作为中断输入。
Bit5	保留	
Bit6	变速度运行	为 ON 时，就选择了变速度运行。Bit6 变为 ON，把相关数据写入到运行速度 1 中，然后使 START 输入为 ON 或 START 标志位为 ON。此时将立即开始变速度运行。
Bit7 ～ Bit15	保留	

（1）第一速度定位操作



当选择第一速度定位操作后，如果 START 为 ON，定位操作就以运行速度 1 开始启动至目标地址 1 减速停止。

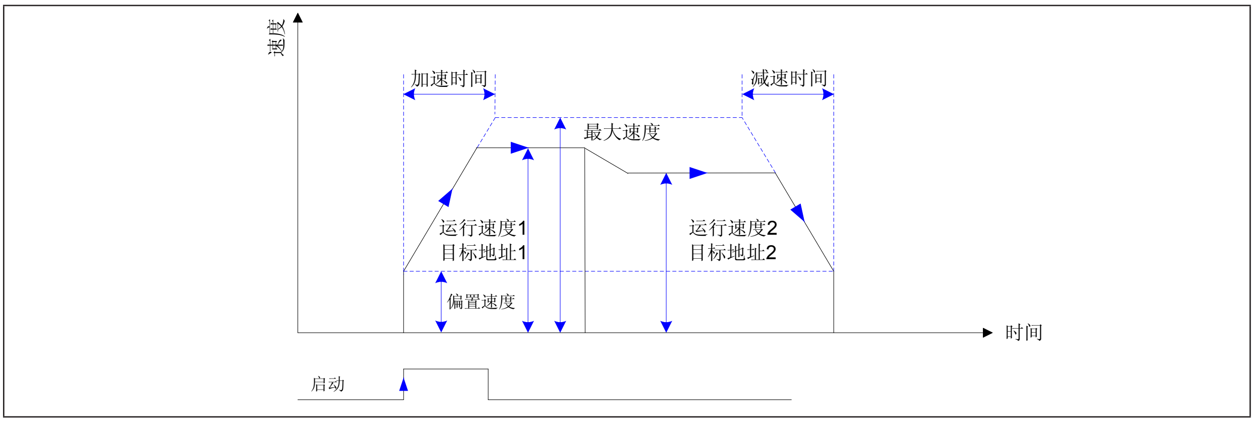
（2）中断第一速度定位操作



当选择中断第一速度定位操作后，如果 START 为 ON，定位操作就以运行速度 1 开始启动。中断输入（X0 的边沿检测）

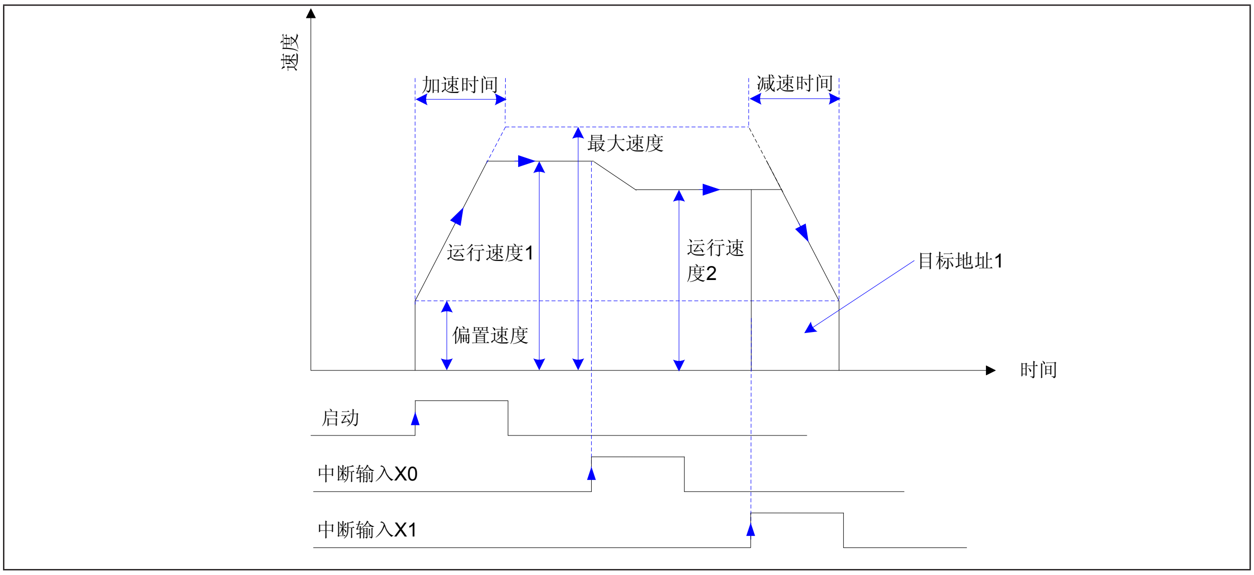
启动后，工件就移动到目标地址 1 减速停止。

（3）第二速度定位操作



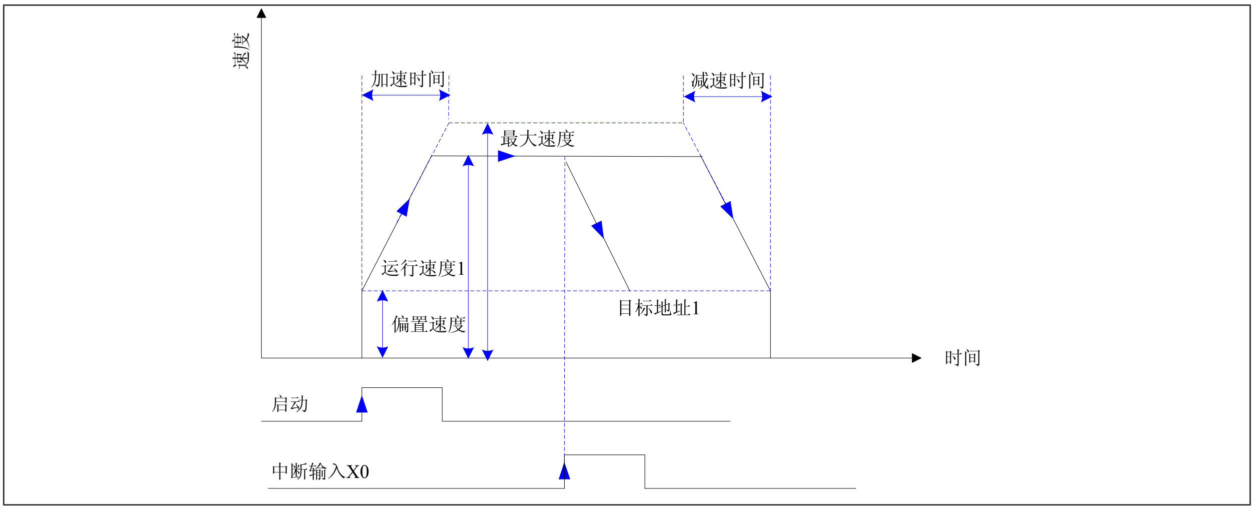
当选择第二速度定位操作后，如果 START 为 ON，定位操作就以运行速度 1 移动到目标地址 1。然后以运行速度 2 移动到目标地址 2，最后减速停止。

（4）中断第二速度定位操作



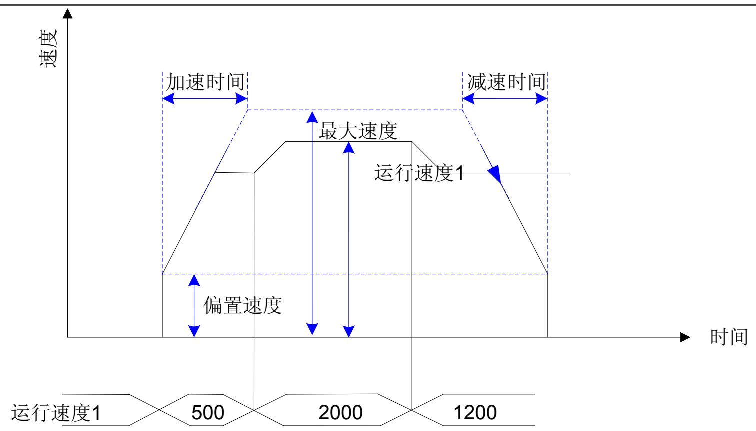
当选择中断第二速度定位操作后，如果 START 为 ON，工件就以运行速度 1 移动。当中断输入 X0 为 ON，速度就切换到运行速度 2。当中断输入 X1 为 ON，工件就移动到目标地址 1，然后减速停止。

（5）中断停止操作



当选择中断停止定位操作后，如果 START 为 ON，工件就以运行速度 1 移动。当工件到达目标地址 1 或者中断输入 X0 为 ON 时，就减速停止。

（6）变速度运行操作



当选择变速度运行定位操作后，工件就加速至运行速度 1，然后开始运行，START 不需要为 ON。当运行速度 1 更改时，加 / 减速就根据加速时间或者减速时间来更改运行速度 1 的速度。当运行速度 1 为 0，工件就减速并且停止。

17．Un.28，Un.128 状态信息

位	定义	说明
Bit0	READY/BUSY	ON：READY（脉冲输出停止） OFF：BUSY（脉冲输出中）
Bit1	正向脉冲输出	当正向脉冲输出时为 ON
Bit2	反向脉冲输出	当反向脉冲输出时为 ON
Bit3	归零完成	当归零正常结束时置 ON。通过断电、归零指令和数据设定型归零指令来复位
Bit4	设定值溢出	当移动距离超出 32 位数据范围时置 ON。通过断电、归零指令和数据设定型归零指令来复位
Bit5	出现错误	出现错误时置 ON，同时错误代码被保存到 Un.37 或者 Un.137 中，可以通过出错复位指令来复位
Bit6	定位结束	当定位完成时置 ON。通过归零，定位动作 START 和 START 指令（Un.26）来清除
Bit7	保留	
Bit8 ～ Bit15	保留	

18．Un.36 ，Un.136 运行参数

位	定义	说明
Bit0 ～ Bit6；Bit9	保留	
Bit7；Bit8	脉冲输出格式	Bit7 为 OFF 且 Bit8 为 OFF 时：FP/RP= 脉冲 + 方向 Bit7 为 ON 且 Bit8 为 OFF 时：FP/RP= 正向脉冲 + 反向脉冲 Bit7 为 OFF 且 Bit8 为 ON 时：FP/RP=AB 相输出
Bit10	归零方向	为 OFF 时：当前值减少方向 为 ON 时：当前值增加方向
Bit11	保留	
Bit12	DOG 输入极性	为 OFF 时：N/O 接点 为 ON 时：N/C 接点
Bit13	开始计数定时	为 OFF 时：DOG 前端 为 ON 时：DOG 后端
Bit14	保留	
Bit15	保留	

19．Un.37，Un.137 错误代码

错误代码	说明
0	没有错误
4	达到了正向极限或是反向极限
6	在 Un. 26 或 Un. 126 中，若归零指令 Bit6、数据设定类型归零指令（Bit7）、启动（Bit9）同时为 ON。（正向点动和反向点动同时为 ON，不属于出错）
7	在 Un. 27 或 Un. 127 中，选择了多个运行模式
×××2	数据的设定范围出错。例如 302 表示 Un. 30 的值有错

20．Un.38，Un.138 端子信息

位	定义	说明
Bit0	输入 START	当 STR 端子输入为 ON 时 Bit0 为 ON
Bit1	输入 DOG	当 DOG 端子输入为 ON 时 Bit1 为 ON
Bit2	输入 PG	当 PG0 端子输入为 ON 时 Bit2 为 ON
Bit3	输入 X0	当 X0 端子输入为 ON 时 Bit3 为 ON
Bit4	输入 X1	当 X1 端子输入为 ON 时 Bit4 为 ON
Bit5	保留	
Bit6	保留	
Bit7	CLR 信号	当 CLR 端子输出为 ON 时 Bit7 为 ON，也可通过写 Bit7 强制 CLR 端子为 ON 或是 OFF
Bit8 ～ Bit15	保留	

12.8.4 应用示例

例 1：系统组态配置（以 CH1 通道为例）

MX2H-2PG 模块地址为 0，使用其第 1 通道，通道参数设置如下：将最大速度设为 100000Hz，最小速度设为 5000 Hz，加速时间 / 减速时间都设为 200ms，归零速度设为 30000 Hz，爬行速度设为 10000 Hz，零点信号设为 2，零点地址为 0，点动速度设为 10000 Hz，运行模式设为第一速度定位运行。

运行参数设置：脉冲输出格式为脉冲 / 方向，归零方向设为当前值减少方向，DG 输入极性设为 N/O 接点，开始计数定时设为 DG 前端。

若要满足上述功能，可在 MXProgrammer 中“工程管理器→系统组态→硬件设置”中进行配置，通道参数配置如图 32 所示。

图 32 系统组态配置

例 2：指令配置（以 CH1 通道为例）

MX2H-2HC 模块地址为 0，通道设置与例 1 相同，如图 33 所示。

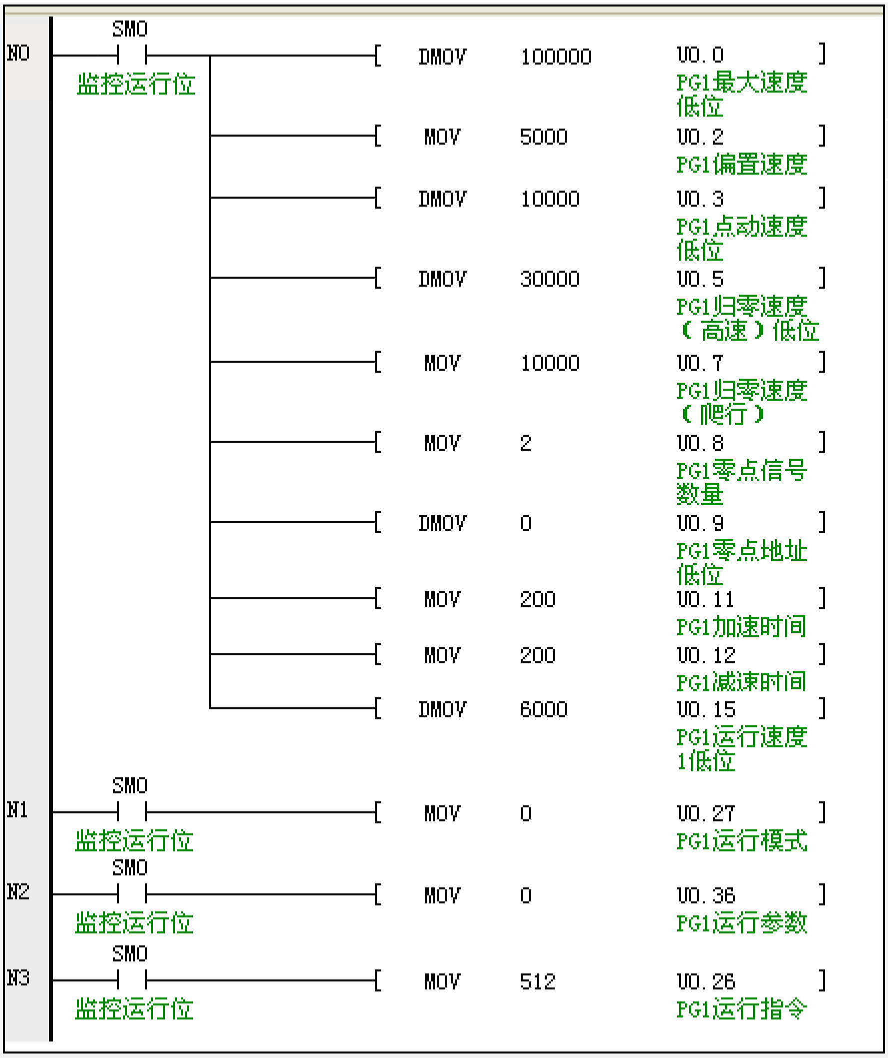


图 33 指令配置

☞ 注意

用系统组态和指令配置都可更改通道特性。当系统组态和指令同时配置且不相同时，以指令配置为准。

附录一 系统错误代码表

错误代码	含义	错误类型	说明	MX1H	MX2H	MX2V
0	无错误发生			√	√	√
1～9	系统保留			√	√	√
系统硬件错误						
10	SRAM 错误	系统错误	停止用户程序；错误灯常亮；消除本错误需停电检查硬件	√	√	√
11	FLASH 错误	系统错误	停止用户程序；错误灯常亮；消除本错误需停电检查硬件	√	√	√
12	通讯口错误	系统错误	停止用户程序；错误灯常亮；消除本错误需停电检查硬件	√	√	√
13	实时时钟错误	系统错误	停止用户程序；错误灯常亮；消除本错误需停电检查硬件	√	√	√
14	12C 错误	系统错误	停止用户程序；错误灯常亮；消除本错误需停电检查硬件	√	√	√
外设错误（20～23）						
20	本机 I/O 严重错误	系统错误	停止用户程序；错误灯常亮；消除本错误需停电检查硬件	√	√	√
21	扩展 I/O 严重错误	系统错误	错误灯闪烁；错误消失，自动消除	√	√	√
22	特殊模块严重错误	系统错误	错误灯闪烁；错误消失，自动消除	√	√	√
23	实时时钟错误刷新（系统刷新时发现读出错误的时间）	系统错误	错误灯闪烁；错误消失，自动清除	√	√	√
24	EEPROM 读写操作错误	系统错误	错误灯闪烁；错误消失，自动清除	√	√	√
25	本地模拟量错误	系统错误	错误灯闪烁；错误消失，自动清除	√	√	√
26	系统特殊模块配置错误	系统错误	错误灯闪烁；错误消失，自动清除		√	√
27	系统扩展卡配置错误	系统错误	错误灯闪烁；错误消失，自动清除	√	√	√
28	I/O 扩展模块配置错误	系统错误	错误灯闪烁；错误消失，自动清除	√	√	√
29	电池电压过低	系统错误	错误灯闪烁；错误消失，自动清除	√	√	√
30	本机序列号与设置序列号不一致	系统错误	错误灯闪烁；错误消失，自动清除	√	√	√
存储错误（40～45）						
40	用户程序文件错误	系统错误	停止用户程序；错误灯常亮；消除本错误需下载新程序 / 格式化	√	√	√
41	系统配置文件错误	系统错误	停止用户程序；错误灯常亮；消除本错误需下载新系统配置文件 / 格式化	√	√	√
42	数据块文件错误	系统错误	停止用户程序；错误灯常亮；消除本错误需下载新数据块文件 / 格式化	√	√	√
43	保持元件数据丢失错误	系统错误	不停止用户程序；错误灯闪烁；消除本错误需消除元件 / 强制操作 / 格式化 / 复位	√	√	√
44	强制表丢失错误	系统错误	不停止用户程序；错误灯闪烁；消除本错误需消除元件 / 强制操作 / 格式化 / 复位	√	√	√
45	用户信息文件错误	系统错误	不停止用户程序；错误灯指示；消除本错误需下载新程序和新数据块文件 / 格式化	√	√	√

错误代码	含义	错误类型	说明	MX1H	MX2H	MX2V
46	全局变量表错误	系统错误	不停止用户程序；错误灯指示；消除本错误需下载新程序和新数据块文件 / 格式化	√	√	√
47	EEPROM 指令数据丢失	系统错误	不停止用户程序；错误灯闪烁；消除本错误需消除元件 / 强制操作 / 格式化 / 复位	√	√	√
48	U 元件保存数据丢失	系统错误	不停止用户程序；错误灯闪烁；消除本错误需消除元件 / 强制操作 / 格式化 / 复位	√	√	√
49 ～ 59	保留					
指令执行错误（60 ～ 75）						
60	用户程序编译错误	执行错误	停止用户程序；错误灯常亮	√	√	√
61	用户程序运行超时	执行错误	停止用户程序；错误灯常亮	√	√	√
62	执行到非法的用户程序指令	执行错误	停止用户程序；错误灯常亮	√	√	√
63	指令操作数的元件类型非法	执行错误	停止用户程序；错误灯常亮	√	√	√
64	指令操作数数值非法	执行错误	不停止用户程序执行，错误灯不指示，但会在 SD20 中指示错误类型代码	√	√	√
65	指令操作数元件编号范围超出	执行错误		√	√	√
66	子程序栈溢出	执行错误		√	√	√
67	用户中断请求队列溢出	执行错误		√	√	√
68	非法的标号跳转或子程序调用	执行错误		√	√	√
69	零除错误	执行错误		√	√	√
70	栈定义非法	执行错误	当栈尺寸、栈中元素个数小于零 栈中元素个数大于栈尺寸限制	√	√	√
71	保留			√	√	√
72	未定义用户子程序或中断子程序	执行错误		√	√	√
73	特殊模块地址无效	执行错误		√	√	√
74	访问特殊模块出错	执行错误		√	√	√
75	I/O 立即刷新	执行错误		√	√	√
76	时钟设置出错	执行错误		√	√	√
77	PLSR 指令参数错误	执行错误		√	√	√
78	特殊模块 BFM 缓存器超界	执行错误		√	√	√
79	DABS 数据读取超时	执行错误		√	√	√
80	DABS 数据读取和校验错误	执行错误		√	√	√
81	DSZR 指令进入异常状态	执行错误		√	√	√
82	DABS 指令数目过多	执行错误		√	√	√
83	TPID 自整定指令超过 8 条	执行错误		√	√	√

附录二 SM 元件一览表

所有特殊辅助继电器，在 STOP → RUN 时被系统初始化，在系统组态中被设置的特殊辅助继电器将在前面的初始化完成后，重新根据系统组态的设置值赋值。

☞ 注意
保留的 SM 在表格中不列出，其读写属性默认为只读（R）。

1. PLC 工作状态标志

地址	名称	动作与功能	R/W	1H	2H	2V
SM0	监控运行位	RUN 状态下，始终为 ON；STOP 状态下，始终为 OFF	R	√	√	√
SM1	第一周期标志	系统由 STOP → RUN 的第一个周期为 ON，之后保持 OFF	R	√	√	√
SM2	上电标志	当系统上电后为 ON，程序运行一个周期后为 OFF	R	√	√	√
SM3	系统错误	系统上电后或由 STOP → RUN 时检测有系统错误发生时为 ON，如果没有任何系统错误发生时清零	R	√	√	√
SM4	电池电压过低	当电池电压过低时为 ON，当检测到电池电压高于 2.4V 时清零	R	√	√	√
SM5	交流停电检测标志	检测到交流失电时为 ON（检测交流失电时间窗口长 40ms），检测延时时间（在 SD5 中设定）测定后，若掉电情况消失则清零	R	√	√	√
SM6	DC24V 失电	检测到 DC24V 失电时为 ON（检测时间窗口长 50ms），延迟 50ms 后再次检测 DC24V，若失电情况消失则清零	R	√	√	√
SM7	无电池工作模式	当该位为 ON 时，系统电池失效情况下，不报电池备份数据丢失和强制表丢失错误（只能通过系统组态配置）	R	√	√	√
SM8	恒定扫描模式	扫描时间恒定（只能通过系统组态配置）时为 ON	R	√	√	√
SM9	输入点启动模式	设定输入点启动模式时（只能通过系统组态配置），该位为 ON	R	√	√	√
SM16	禁止所有输出使能标志	当该位为 ON 时，所有输出被禁止	R/W	√	√	√
SM17	强制运行使能标志	当该位为 ON 时，PLC 被强制运行	R/W	√	√	√

2. 时钟运行位

地址	名称	动作与功能	R/W	1H	2H	2V
SM10	10ms 时钟	以 10ms 为一个周期的时钟振荡（半个周期翻转，用户程序运行时首半个周期为 0）	R	√	√	√
SM11	100ms 时钟	以 100ms 为一个周期的时钟振荡（半个周期翻转，用户程序运行时首半个周期为 0）	R	√	√	√
SM12	1s 时钟	以 1s 为一个周期的时钟振荡（半个周期翻转，用户程序运行时首半个周期为 0）	R	√	√	√
SM13	1min 时钟	以 1min 为一个周期的时钟振荡（半个周期翻转，用户程序运行时首半个周期为 0）	R	√	√	√
SM14	1hour 时钟	1hour 为周期的时钟振荡（半个周期翻转，用户程序运行时首半个周期为 0）	R	√	√	√
SM15	扫描周期振荡位	该位每个扫描周期翻转一次（用户程序运行时第一个周期为 0）	R	√	√	√

3. 用户执行错误

地址	名称	动作与功能	R/W	1H	2H	2V
SM20	指令执行错误标志	当指令执行错误时为 ON，同时具体的错误类型代码填入 SD20。指令执行正确后清零	R	√	√	√
SM21	指令元件编号溢出标志	当指令元件编号溢出时为 ON，同时将错误类型代码填入 SD20	R	√	√	√
SM22	指令参数非法标志	当指令参数非法时为 ON，同时将错误类型代码填入 SD20	R	√	√	√
SM29	指令执行结束标志	指令动作结束时接通	R	√	√	√

4. 中断控制

地址	名称	动作与功能	R/W	1H	2H	2V
SM30	X0 上升沿中断使能标志	置 ON 时，使能 X0 上升沿中断	R/W	√	√	√
SM31	X1 上升沿中断使能标志	置 ON 时，使能 X1 上升沿中断	R/W	√	√	√
SM32	X2 上升沿中断使能标志	置 ON 时，使能 X2 上升沿中断	R/W	√	√	√
SM33	X3 上升沿中断使能标志	置 ON 时，使能 X3 上升沿中断	R/W	√	√	√
SM34	X4 上升沿中断使能标志	置 ON 时，使能 X4 上升沿中断	R/W	√	√	√
SM35	X5 上升沿中断使能标志	置 ON 时，使能 X5 上升沿中断	R/W	√	√	√
SM36	X6 上升沿中断使能标志	置 ON 时，使能 X6 上升沿中断	R/W	√	√	√
SM37	X7 上升沿中断使能标志	置 ON 时，使能 X7 上升沿中断	R/W	√	√	√
SM40	X0 下降沿中断使能标志	置 ON 时，使能 X0 下降沿中断	R/W	√	√	√
SM41	X1 下降沿中断使能标志	置 ON 时，使能 X1 下降沿中断	R/W	√	√	√
SM42	X2 下降沿中断使能标志	置 ON 时，使能 X2 下降沿中断	R/W	√	√	√
SM43	X3 下降沿中断使能标志	置 ON 时，使能 X3 下降沿中断	R/W	√	√	√
SM44	X4 下降沿中断使能标志	置 ON 时，使能 X4 下降沿中断	R/W	√	√	√
SM45	X5 下降沿中断使能标志	置 ON 时，使能 X5 下降沿中断	R/W	√	√	√
SM46	X6 下降沿中断使能标志	置 ON 时，使能 X6 下降沿中断	R/W	√	√	√
SM47	X7 下降沿中断使能标志	置 ON 时，使能 X7 下降沿中断	R/W	√	√	√
SM48	PORT0 字符发送中断使能标志	置 ON 时，允许	R/W	√	√	√
SM49	PORT0 字符接收中断使能标志	置 ON 时，允许	R/W	√	√	√
SM50	PORT0 帧发送中断使能标志	置 ON 时，允许	R/W	√	√	√
SM51	PORT0 帧接收中断使能标志	置 ON 时，允许	R/W	√	√	√
SM52	PORT1 字符发送中断使能标志	置 ON 时，允许	R/W	√	√	√
SM53	PORT1 字符接收中断使能标志	置 ON 时，允许	R/W	√	√	√
SM54	PORT1 帧发送中断使能标志	置 ON 时，允许	R/W	√	√	√
SM55	PORT1 帧接收中断使能标志	置 ON 时，允许	R/W	√	√	√
SM56	电源失电中断	置 ON 时，使能主电源失电中断	R/W	√	√	√
SM57	PORT2 字符发送中断使能标志	置 ON 时，允许	R/W		√	√
SM58	PORT2 字符接收中断使能标志	置 ON 时，允许	R/W		√	√
SM59	PORT2 帧发送中断使能标志	置 ON 时，允许	R/W		√	√
SM60	PORT2 帧接收中断使能标志	置 ON 时，允许	R/W		√	√
SM61	MX2H-1PG-BD 输出完成中断使能标志	置 ON 时，允许	R/W		√	√
SM66	定时中断 0 使能标志	置 ON 时，使能定时中断 0	R/W	√	√	√
SM67	定时中断 1 使能标志	置 ON 时，使能定时中断 1	R/W	√	√	√

地址	名称	动作与功能	R/W	1H	2H	2V
SM68	定时中断 2 使能标志	置 ON 时，使能定时中断 2	R/W	√	√	√
SM69	高速计数中断使能标志	置 ON 时，使能高速计数中断	R/W	√	√	√
SM70	Y0 输出完成中断使能标志	置 ON 时，使能 PT0（Y0）输出完成中断	R/W	√	√	√
SM71	Y1 输出完成中断使能标志	置 ON 时，使能 PT0（Y1）输出完成中断	R/W	√	√	√
SM72	Y2 输出完成中断使能标志	置 ON 时，使能 PT0（Y2）输出完成中断	R/W	√	√	√
SM73	Y3 输出完成中断使能标志	置 ON 时，使能 PT0（Y3）输出完成中断	R/W	√	√	√
SM74	Y4 输出完成中断使能标志	置 ON 时，使能 PT0（Y4）输出完成中断	R/W	√	√	√
SM75	Y5 输出完成中断使能标志	置 ON 时，使能 PT0（Y5）输出完成中断	R/W	√	√	√
SM76	Y6 输出完成中断使能标志	置 ON 时，使能 PT0（Y6）输出完成中断	R/W	√	√	√
SM77	Y7 输出完成中断使能标志	置 ON 时，使能 PT0（Y7）输出完成中断	R/W	√	√	√
SM79	CAN0 接收中断使能标志	置 ON 时，使能 CAN 0 接收中断	R/W	√	√	√

5. 高速脉冲输出控制

地址	名称	动作与功能	R/W	1H	2H	2V
SM80	Y0 脉冲输出停止使能标志	置 ON 时，Y0 高速脉冲输出停止	R/W	√	√	√
SM81	Y1 脉冲输出停止使能标志	置 ON 时，Y1 高速脉冲输出停止	R/W	√	√	√
SM82	Y0 脉冲输出监控位	Y0 高速脉冲输出监视（调用时为 ON，未调用时为 OFF）	R	√	√	√
SM83	Y1 脉冲输出监控位	Y1 高速脉冲输出监视（调用时为 ON，未调用时为 OFF）	R	√	√	√
SM84	PWM 微秒输出使能标志	置 ON 时，时间基本单位为微秒；置 OFF 时，时间基本单位为毫秒	R/W	√	√	√
SM86	MX2H-1PG-BD 脉冲输出停止使能标志	置 ON 时，MX2H-1PG-BD 高速脉冲输出停止	R/W		√	√
SM87	MX2H-1PG-BD 脉冲输出监控位	MX2H-1PG-BD 高速脉冲输出监视（调用时为 ON，未调用时为 OFF）	R		√	√
SM261	Y2 脉冲输出停止使能标志	置 ON 时，Y2 高速脉冲输出停止	R/W	√	√	√
SM262	Y3 脉冲输出停止使能标志	置 ON 时，Y3 高速脉冲输出停止	R/W	√	√	√
SM263	Y4 脉冲输出停止使能标志	置 ON 时，Y4 高速脉冲输出停止	R/W	√	√	√
SM264	Y5 脉冲输出停止使能标志	置 ON 时，Y5 高速脉冲输出停止	R/W	√	√	√
SM265	Y6 脉冲输出停止使能标志	置 ON 时，Y6 高速脉冲输出停止	R/W	√	√	√
SM266	Y7 脉冲输出停止使能标志	置 ON 时，Y7 高速脉冲输出停止	R/W	√	√	√
SM267	Y2 脉冲输出监控位	Y2 高速脉冲输出监视（调用时为 ON，未调用时为 OFF）	R	√	√	√
SM268	Y3 脉冲输出监控位	Y3 高速脉冲输出监视（调用时为 ON，未调用时为 OFF）	R	√	√	√
SM269	Y4 脉冲输出监控位	Y4 高速脉冲输出监视（调用时为 ON，未调用时为 OFF）	R	√	√	√
SM270	Y5 脉冲输出监控位	Y5 高速脉冲输出监视（调用时为 ON，未调用时为 OFF）	R	√	√	√

地址	名称	动作与功能	R/W	1H	2H	2V
SM271	Y6 脉冲输出监控位	Y6 高速脉冲输出监视（调用时为 ON，未调用时为 OFF）	R	√	√	√
SM272	Y7 脉冲输出监控位	Y7 高速脉冲输出监视（调用时为 ON，未调用时为 OFF）	R	√	√	√
SM150	Y0_FOLLOWA 指令脉冲输出方向强制	FOLLOWA 指令执行前，SM150 ～ SM157 置 ON，脉冲输出方向改变；若 FOLLOWA 指令执行中 SM150 ～ SM157 置 ON，对方脉冲输出方向无影响	R/W		√	√
SM151	Y1_FOLLOWA 指令脉冲输出方向强制		R/W		√	√
SM152	Y2_FOLLOWA 指令脉冲输出方向强制		R/W		√	√
SM153	Y3_FOLLOWA 指令脉冲输出方向强制		R/W			√
SM154	Y4_FOLLOWA 指令脉冲输出方向强制		R/W			√
SM155	Y5_FOLLOWA 指令脉冲输出方向强制		R/W			√
SM156	Y6_FOLLOWA 指令脉冲输出方向强制		R/W			√

6. 脉冲捕捉监视位

地址	名称	动作与功能	R/W	1H	2H	2V
SM90	X0 脉冲捕捉监视位	X0 上升沿脉冲捕捉	R/W	√	√	√
SM91	X1 脉冲捕捉监视位	X1 上升沿脉冲捕捉	R/W	√	√	√
SM92	X2 脉冲捕捉监视位	X2 上升沿脉冲捕捉	R/W	√	√	√
SM93	X3 脉冲捕捉监视位	X3 上升沿脉冲捕捉	R/W	√	√	√
SM94	X4 脉冲捕捉监视位	X4 上升沿脉冲捕捉	R/W	√	√	√
SM95	X5 脉冲捕捉监视位	X5 上升沿脉冲捕捉	R/W	√	√	√
SM96	X6 脉冲捕捉监视位	X6 上升沿脉冲捕捉	R/W	√	√	√
SM97	X7 脉冲捕捉监视位	X7 上升沿脉冲捕捉	R/W	√	√	√

注意：

1. 由 STOP 到 RUN 时脉冲捕捉监视位清除。在端口上若有 HCNT 高速计数器驱动指令和 SPD 测频指令，则此端口的脉冲捕捉无效。在其它情况都有效。具体说明参考 HCNT 高速计数器驱动指令和 SPD 测频指令。
2. 对 X0 ～ X7 脉冲的总计输入频率数在脉冲捕捉、SPD 指令时 1H/2H 不超过 10K，使用外部中断时不超过 5KHz。

7. 四倍频

地址	名称	动作与功能	R/W	1H	2H	2V
SM100	C251，C252，C254 的 1 倍 / 4 倍频切换使能标志	置 ON 时，设定为 4 倍频	R/W	√	√	√
SM101	C253, C255,C253OP 的 1 倍 / 4 倍频切换使能标志	置 ON 时，设定为 4 倍频	R/W	√	√	√
SM102	C254OP 的 1 倍 / 4 倍频切换使能标志	置 ON 时，设定为 4 倍频	R/W	√	√	√
SM103	C255OP 的 1 倍 / 4 倍频切换使能标志	置 ON 时，设定为 4 倍频	R/W	√	√	√

8. 自由端口（PORT0）

地址	名称	动作与功能	R/W	1H	2H	2V
SM110	PORT0 发送使能标志	当使用 XMT 指令时该位被置位，当发送结束后该位将被自动清零。用户也可对其手动清零，以中止端口 0 的当前发送任务。当再次置 ON 时，该端口的发送任务可以继续	R/W	√	√	√

地址	名称	动作与功能	R/W	1H	2H	2V
SM111	PORT0 接收使能标志	当使用 RCV 指令时该位被置位，当接收结束后该位将被自动清零。用户也可对其手动清零，以中止端口 0 的当前接收任务。当再次置 ON 时，该端口的接收任务可以继续	R/W	√	√	√
SM112	PORT0 发送完成标志	当使用 XMT 指令，发送完成时置位	R/W	√	√	√
SM113	PORT0 接收完成标志	当使用 RCV 指令，接收完成置位	R/W	√	√	√
SM114	PORT0 空闲标志	当端口没有通讯任务时为 ON	R	√	√	√
注意： SM114 是针对于适用 PORT0 的所有通信协议而言的一个空闲标志。比如：PLC 的 PORT0 可用于 MXLink、Modbus 及自由口协议，无论采用哪种协议，SM114 都适用。						

9. 自由端口（PORT1）

地址	名称	动作与功能	R/W	1H	2H	2V
SM120	PORT1 发送使能标志	当使用 XMT 指令时该位被置位，当发送结束后该位将被自动清零。用户也可对其手动清零，以中止端口 1 的当前发送任务。当再次置 ON 时，该端口的发送任务可以继续	R/W	√	√	√
SM121	PORT1 接收使能标志	当使用 RCV 指令时该位被置位，当接收结束后该位将被自动清零。用户也可对其手动清零，以中止端口 1 的当前接收任务。当再次置 ON 时，该端口的接收任务可以继续	R/W	√	√	√
SM122	PORT1 发送完成标志	当使用 XMT 指令，发送完成时置位	R/W	√	√	√
SM123	PORT1 接收完成标志	当使用 RCV 指令，接收完成置位	R/W	√	√	√
SM124	PORT1 空闲标志	当端口没有通讯任务时为 ON	R	√	√	√
注意： SM124 是针对于适用 PORT1 的所有通信协议而言的一个空闲标志。比如：PLC 的 PORT1 可用于 MXLink、Modbus 及自由口协议，无论采用那种协议，SM124 都适用。						

10. 自由端口（PORT2）

地址	名称	动作与功能	R/W	1H	2H	2V
SM130	PORT2 发送使能标志	当使用 XMT 指令时该位被置位，当发送结束后该位将被自动清零。用户也可对其手动清零，以中止端口 2 的当前发送任务。当再次置 ON 时，该端口的发送任务可以继续	R/W		√	√
SM131	PORT2 接收使能标志	当使用 RCV 指令时该位被置位，当接收结束后该位将被自动清零。用户也可对其手动清零，以中止端口 2 的当前接收任务。当再次置 ON 时，该端口的接收任务可以继续	R/W		√	√
SM132	PORT2 发送完成标志	当使用 XMT 指令，发送完成时置位	R/W		√	√
SM133	PORT2 接收完成标志	当使用 RCV 指令，接收完成置位	R/W		√	√
SM134	PORT2 空闲标志	当端口没有通讯任务时为 ON	R		√	√
注意： SM134 是针对于适用 PORT2 的所有通信协议而言的一个空闲标志。比如：PLC 的 PORT2 可用于 MXLink、Modbus 及自由口协议，无论采用那种协议，SM134 都适用。						

11. Modbus 通讯

地址	名称	动作与功能	R/W	1H	2H	2V
SM135	PORT1 的 Modbus 的通讯完成	通讯完成时置位	R/W	√	√	√
SM136	PORT1 的 Modbus 的通讯错误	通讯错误时置位	R/W	√	√	√
SM137	PORT2 的 Modbus 的通讯完成	通讯完成时置位	R/W		√	√
SM138	PORT2 的 Modbus 的通讯错误	通讯错误时置位	R/W		√	√

地址	名称	动作与功能	R/W	1H	2H	2V
SM139	PORT3 的 Modbus 的通讯完成	通讯完成时置位	R/W		√	√
SM140	PORT3 的 Modbus 的通讯错误	通讯错误时置位	R/W		√	√

12. SORT 指令

地址	名称	动作与功能	R/W	1H	2H	2V
SM165	SORT 排序方向使能标志	置 ON 时，降序排列	R/W	√	√	√

13. 运算标志及 ASCII 码转换指令标志

地址	名称	动作与功能	R/W	1H	2H	2V
SM180	零标志	相关指令执行的结果为 0 时，该位置位，可手动清零	R/W	√	√	√
SM181	进位 / 溢出标志	相关指令执行的结果有进位时，该位置位，可手动清零	R/W	√	√	√
SM182	借位标志	相关指令执行的结果有借位时，该位置位，可手动清零	R/W	√	√	√
SM185	表格比较标志	当整个表格记录完成时为 ON	R	√	√	√
SM186	ASC 指令存储方式标志	置 ON 时，每个字只有低字节存放 1 个 ASCII 码； 置 OFF 时，每个字高低字节各存放 1 个 ASCII 码	R/W	√	√	√

14. 系统总线错误标志

地址	名称	动作与功能	R/W	1H	2H	2V
SM190	主模块总线错误标志	1. 当发生主模块总线操作错误时为 ON，引起系统停机 2. 上电编址正确清零 3. STOP → RUN 无主模块错误清除 4. 下载新程序时清除	R	√	√	√
SM191	一般模块总线错误标志	当发生一般模块总线操作错误时为 ON，系统报警。系统故障消除该标志位自动清除	R		√	√
SM192	特殊模块总线错误标志	当发生特殊模块总线操作错误时为 ON，系统报警。系统故障消除该标志位自动清除	R		√	√

15. 实时时钟错误标志

地址	名称	动作与功能	R/W	1H	2H	2V
SM193	读写实时时钟错误标志	当发生实时时钟错误时为 ON，系统故障消除该标志位自动清零	R	√	√	√

16. 增 / 减型计数器计数方向

地址	名称	动作与功能	R/W	1H	2H	2V
SM200	C200 增减计数使能标志	当 SM 置 ON 时，其对应的计数器为减型计数 当 SM 置 OFF 时，其对应的计数器为增型计数	R/W	√	√	√
SM201	C201 增减计数使能标志		R/W	√	√	√
SM202	C202 增减计数使能标志		R/W	√	√	√
SM203	C203 增减计数使能标志		R/W	√	√	√
SM204	C204 增减计数使能标志		R/W	√	√	√
SM205	C205 增减计数使能标志		R/W	√	√	√
SM206	C206 增减计数使能标志		R/W	√	√	√
SM207	C207 增减计数使能标志		R/W	√	√	√
SM208	C208 增减计数使能标志		R/W	√	√	√
SM209	C209 增减计数使能标志		R/W	√	√	√

地址	名称	动作与功能	R/W	1H	2H	2V
SM210	C210 增减计数使能标志	当 SM 置 ON 时，其对应的计数器为减型计数 当 SM 置 OFF 时，其对应的计数器为增型计数	R/W	√	√	√
SM211	C211 增减计数使能标志		R/W	√	√	√
SM212	C212 增减计数使能标志		R/W	√	√	√
SM213	C213 增减计数使能标志		R/W	√	√	√
SM214	C214 增减计数使能标志		R/W	√	√	√
SM215	C215 增减计数使能标志		R/W	√	√	√
SM216	C216 增减计数使能标志		R/W	√	√	√
SM217	C217 增减计数使能标志		R/W	√	√	√
SM218	C218 增减计数使能标志		R/W	√	√	√
SM219	C219 增减计数使能标志		R/W	√	√	√
SM220	C220 增减计数使能标志		R/W	√	√	√
SM221	C221 增减计数使能标志		R/W	√	√	√
SM222	C222 增减计数使能标志		R/W	√	√	√
SM223	C223 增减计数使能标志		R/W	√	√	√
SM224	C224 增减计数使能标志		R/W	√	√	√
SM225	C225 增减计数使能标志		R/W	√	√	√
SM226	C226 增减计数使能标志		R/W	√	√	√
SM227	C227 增减计数使能标志		R/W	√	√	√
SM228	C228 增减计数使能标志		R/W	√	√	√
SM229	C229 增减计数使能标志		R/W	√	√	√
SM230	C230 增减计数使能标志		R/W	√	√	√
SM231	C231 增减计数使能标志		R/W	√	√	√
SM232	C232 增减计数使能标志		R/W	√	√	√
SM233	C233 增减计数使能标志		R/W	√	√	√
SM234	C234 增减计数使能标志		R/W	√	√	√

17. 高速计数器的计数方向及监控

区分	地址	名称	寄存器内容	R/W	1H	2H	2V
单向单端计数输入方式	SM235	C235 增减计数使能标志	当 SM 置 ON 时，其对应的计数器为减型计数当 SM 置 OFF 时，其对应的计数器为增型计数	R/W	√	√	√
	SM236	C236 增减计数使能标志		R/W	√	√	√
	SM237	C237 增减计数使能标志		R/W	√	√	√
	SM238	C238 增减计数使能标志		R/W	√	√	√
	SM239	C239 增减计数使能标志		R/W	√	√	√
	SM240	C240 增减计数使能标志		R/W	√	√	√
	SM241	C241 增减计数使能标志		R/W	√	√	√
	SM242	C242 增减计数使能标志		R/W	√	√	√
	SM243	C243 增减计数使能标志		R/W	√	√	√
	SM244	C244 增减计数使能标志		R/W	√	√	√
	SM107	C244OP 增减计数标志		R			√
	SM245	C245 增减计数使能标志		R/W	√	√	√
	SM108	C245OP 增减计数标志		R			√
	SM107	C250OP 增减计数标志		R			√

区分	地址	名称	寄存器内容	R/W	1H	2H	2V
单向增减计数输入方式	SM246	C246 增减计数标志	当单相增减计数输入计数器和双相计数输入计数器处于减型计数模式时，其对应的 SM 为 ON，处于增型计数模式时，其对应的 SM 为 OFF	R	√	√	√
	SM247	C247 增减计数标志		R	√	√	√
	SM105	C247OP 增减计数标志		R/W			√
	SM248	C248 增减计数标志		R	√	√	√
	SM104	C248OP 增减计数标志		R	√	√	√
	SM249	C249 增减计数标志		R	√	√	√
	SM106	C249OP 增减计数标志		R			√
	SM250	C250 增减计数标志		R	√	√	√
双向计数输入方式	SM251	C251 增减计数标志		R	√	√	√
	SM252	C252 增减计数标志		R	√	√	√
	SM253	C253 增减计数标志		R	√	√	√
	SM104	C253OP 增减计数标志		R	√	√	√
	SM254	C254 增减计数标志		R	√	√	√
	SM105	C254OP 增减计数标志		R	√	√	√
	SM255	C255 增减计数标志		R	√	√	√
	SM106	C255OP 增减计数标志		R			√

18. 增强定位

地址	名称	功能	R/W	1H	2H	2V
SM260	中断输入功能指定有效	适用 DDVIT。当不使用中断输入指定功能，那么 Y0 ～ Y7 对应 X0 ～ X7 的中断；当使用中断输入指定功能时则该位置 ON，然后对应的 SD198、SD199 的每一个 4 位对应着每一个输出 Y(n) 的输入	R/W		√	√
SM280/SM290/SM310 /SM320/SM330/SM340 /SM350/SM360	Y0 ～ Y7 清零功能有效	适用 DZRN /DSZR，定位指令原点回归（DZRN）清零信号输出功能有效 z	R/W		√	√
SM281/SM291/SM311 /SM321/SM331/SM341 /SM351/SM361	Y0 ～ Y7 清零信号指定元件有效	适用 DZRN/DSZR，对应的 SD210/SD230/ SD290/SD310/SD330/SD350/SD370/SD390 内的数值分别为 Y0 ～ Y7 的清零 Y 元件，不指定则 Y10 ～ Y17 分别为 Y0 ～ Y7 的清零信号	R/W		√	√
SM282/SM292/SM312 /SM322/SM332/SM342 /SM352/SM362	Y0 ～ Y7 原点回归方向标志	适用于 DZRN/DSZR，作用于 Y(n) 对应的轴：置 ON 时，表示原点回归方向为正转方向；置 OFF 时，表示原点回归方向为反转方向	R/W		√	√
SM283/SM293/SM313 /SM323/SM333/SM343 /SM353/ SM363	Y0 ～ Y7 正转极限标志	适用于 DZRN/DSZR/DDRVI/DDRVA，作用于 Y(n) 对应的轴：置 ON 时，表示到达正转方向的极限；置 OFF 时，表示未到极限	R/W		√	√
SM284/SM294/SM314 /SM324/SM334/SM344 /SM354/ SM364	Y0 ～ Y7 反转极限标志	适用于 DZRN/DSZR/DDRVI/DDRVA，作用于 Y(n) 对应的轴：置 ON 时，表示到达反转方向的极限；置 OFF 时，表示未到极限	R/W		√	√
SM285/SM295/SM315 /SM325/SM335/SM345 /SM355/SM365	Y0 ～ Y7 近点信号逻辑反转使能标志	适用于 DZRN/DSZR，作用于 Y (n) 对应的轴：置 ON 时，按负逻辑处理（输入 OFF 时，近点信号为 ON）；置 OFF 时，按正逻辑处理（输入 ON 时，近点信号为 ON）	R/W		√	√

地址	名称	功能	R/W	1H	2H	2V
SM286/SM296/SM316 /SM326/SM336/SM346 /SM356/SM366	Y0 ～ Y7 零点信号逻辑反转使能标志	适用于 DSZR，作用于 Y (n) 对应的轴：置 ON 时，按负逻辑处理（输入 OFF 时，零点信号为 ON）；置 OFF 时，按正逻辑处理（输入 ON 时，零点信号为 ON）	R/W		√	√
SM287/SM297/SM317 /SM327/SM337/SM347 /SM357/SM367	Y0 ～ Y7 中断信号逻辑反转使能标志	适用于 DDVIT，作用于 Y (n) 对应的轴：置 ON 时，按负逻辑处理（输入 OFF 时，中断信号为 ON）；置 OFF 时，按正逻辑处理（输入 ON 时，中断信号为 ON）	R/W		√	√
SM288/SM298/SM318 /SM328/SM338/SM348 /SM358/SM368	Y0 ～ Y7 定位指令运行标志	用于监测执行 DZRN/DDRVI/DDRVA/DSZR/ DDVIT 时高速输出通道 Y(n) 的状态，正在运行时为 ON，空闲时为 OFF	R/W		√	√
SM289/SM299/SM319 /SM329/SM339/SM349 /SM359/SM369	Y0 ～ Y7 用户中断输入使能标志	SD198 指定为用户中断指令软元件时，该位置 ON 则使能 Y (n) 用户中断	R/W		√	√

19. 信号报警器

地址	名称	功能	R/W	1H	2H	2V
SM400	信号报警器使能标志	SM400 置 ON 时，SM401 和 SD401 工作	R/W	√	√	√
SM401	信号报警器动作标志	状态 S900 ～ S999 中任一动作，SM401 置 ON	R/W	√	√	√

20. 定时器时钟

地址	名称	功能	R/W	1H	2H	2V
SM430	定时器时钟输出 1	DUTY 指令用	R/W	√	√	√
SM431	定时器时钟输出 2	DUTY 指令用	R/W	√	√	√
SM432	定时器时钟输出 3	DUTY 指令用	R/W	√	√	√
SM433	定时器时钟输出 4	DUTY 指令用	R/W	√	√	√
SM434	定时器时钟输出 5	DUTY 指令用	R/W	√	√	√

21. PORT1 MXLink 通讯

地址	名称	动作与功能	R/W	1H	2H	2V
SM440	0 号站状态标志	ON：活动　OFF：僵死	R	√	√	√
SM441	1 号站状态标志	ON：活动　OFF：僵死	R	√	√	√
SM442	2 号站状态标志	ON：活动　OFF：僵死	R	√	√	√
SM443	3 号站状态标志	ON：活动　OFF：僵死	R	√	√	√
SM444	4 号站状态标志	ON：活动　OFF：僵死	R	√	√	√
SM445	5 号站状态标志	ON：活动　OFF：僵死	R	√	√	√
SM446	6 号站状态标志	ON：活动　OFF：僵死	R	√	√	√
SM447	7 号站状态标志	ON：活动　OFF：僵死	R	√	√	√
SM448	8 号站状态标志	ON：活动　OFF：僵死	R	√	√	√
SM449	9 号站状态标志	ON：活动　OFF：僵死	R	√	√	√
SM450	10 号站状态标志	ON：活动　OFF：僵死	R	√	√	√

地址	名称	动作与功能	R/W	1H	2H	2V
SM451	11 号站状态标志	ON：活动　OFF：僵死	R	√	√	√
SM452	12 号站状态标志	ON：活动　OFF：僵死	R	√	√	√
SM453	13 号站状态标志	ON：活动　OFF：僵死	R	√	√	√
SM454	14 号站状态标志	ON：活动　OFF：僵死	R	√	√	√
SM455	15 号站状态标志	ON：活动　OFF：僵死	R	√	√	√

22. PORT2 MXLink 通讯

地址	名称	动作与功能	R/W	1H	2H	2V
SM460	0 号站状态标志	ON：活动　OFF：僵死	R		√	√
SM461	1 号站状态标志	ON：活动　OFF：僵死	R		√	√
SM462	2 号站状态标志	ON：活动　OFF：僵死	R		√	√
SM463	3 号站状态标志	ON：活动　OFF：僵死	R		√	√
SM464	4 号站状态标志	ON：活动　OFF：僵死	R		√	√
SM465	5 号站状态标志	ON：活动　OFF：僵死	R		√	√
SM466	6 号站状态标志	ON：活动　OFF：僵死	R		√	√
SM467	7 号站状态标志	ON：活动　OFF：僵死	R		√	√
SM468	8 号站状态标志	ON：活动　OFF：僵死	R		√	√
SM469	9 号站状态标志	ON：活动　OFF：僵死	R		√	√
SM470	10 号站状态标志	ON：活动　OFF：僵死	R		√	√
SM471	11 号站状态标志	ON：活动　OFF：僵死	R		√	√
SM472	12 号站状态标志	ON：活动　OFF：僵死	R		√	√
SM473	13 号站状态标志	ON：活动　OFF：僵死	R		√	√
SM474	14 号站状态标志	ON：活动　OFF：僵死	R		√	√
SM475	15 号站状态标志	ON：活动　OFF：僵死	R		√	√

附录三 SD 元件一览表

☞ 注意

1. SD50 ～ SD55 以外的所有特殊数据寄存器，在 STOP → RUN 时被系统初始化。
2. 保留的 SD 表格中不列出，保留的 SD 元件读写属性默认为只读（R）。

1. PLC 工作状态数据

地址	名称	动作与功能	范围	R/W	1H	2H	2V
SD0	PLC 类型	本机类型识别码		R	√	√	√
SD1	版本号	例如：100 为 1.00		R	√	√	√
SD2	用户程序的容量	例如：8 表示 8k 步程序		R	√	√	√
SD3	系统错误代码	存储发生的系统错误代码		R	√	√	√
SD4	电池电压值	以 0.1V 为单位，3.2V 为 32		R	√	√	√
SD5	交流失电检测延迟时间设置值	若设置值小于 10ms 按 10ms 处理。若设置值大于 100ms 按 100ms 处理（只能通过系统组态配置）	10 ～ 100ms	R	√	√	√
SD7	扩展 I/O 模块个数			R		√	√
SD8	特殊模块个数			R		√	√
SD9	开机输入点设置值	运行状态（运行、停止）控制的输入点设定采用 8 进制（X0 显示为 0，X10 显示为 8，最大到 15）（只能通过系统组态配置）	0 ～ 15	R	√	√	√
SD10	主模块 IO 的点数	高字节：输入；低字节：输出		R	√	√	√
SD11	扩展 IO 的点数	高字节：输入；低字节：输出		R		√	√
SD12	扩展卡类型及点数	高字节：输入；低字节：输出		R	√	√	√

2. 运行错误代码 FIFO 区

地址	名称	动作与功能	R/W	1H	2H	2V
SD20	运行错误代码栈 0	按队列顺序，保留 5 条最近的运行错误类型代码，SD20 总保存新近发生的错误的类型代码	R	√	√	√
SD21	运行错误代码栈 1		R	√	√	√
SD22	运行错误代码栈 2		R	√	√	√
SD23	运行错误代码栈 3		R	√	√	√
SD24	运行错误代码栈 4		R	√	√	√

3. 扫描时间

地址	名称	动作与功能	范围	R/W	1H	2H	2V
SD30	当前扫描值	当前扫描时间（ms 为单位）		R	√	√	√
SD31	最小扫描时间	扫描时间的最小值（ms 为单位）		R	√	√	√
SD32	最大扫描时间	扫描时间的最大值（ms 为单位）		R	√	√	√
SD33	恒定扫描时间设定值	初始值为 0ms（只能通过系统组态配置）以 1ms 为单位，若恒定扫描时间大于监控超时设定值，程序超时报警。若程序某个扫描周期大于恒定扫描，该周期恒定扫描模式自动失效，不作报警处理。设定值大于 1000ms 时，按 1000 处理。	0 ～ 1000ms	R	√	√	√

地址	名称	动作与功能	范围	R/W	1H	2H	2V
SD34	用户程序超时设定值	初始值为 100ms（只能通过系统块配置）。设定值小于 100 时，按 100 处理; 大于 1000 时，按 1000 处理	1 0 0 ～ 1000ms	R	√	√	√
SD35	通信背景时间设定值	PORT 接收到数据后延迟 35ms 处理	10 ～ 100ms	R	√	√	√

☞ 注意

1．SD30、SD31、SD32 有 1ms 的误差。

2．当恒定扫描时间设定值 SD33 与程序超时设定值 SD34 值相近时，受系统工作情况及用户程序影响，易发生程序超时错误，建议程序超时设定值大于恒定扫描时间设定值（SD33）5ms。

4. 输入滤波常数设置

地址	名称	动作与功能	范围	R/W	1H	2H	2V
SD36	主单元 X40 以后输入滤波常数	可通过系统组态配置	0.8 ～ 51.2ms	R		√	√
SD37	主单元 X40 以后输入滤波常数	可通过系统组态配置	0.8 ～ 51.2ms	R		√	√
SD38	主单元 X20 ～ X37 输入滤波常数	可通过系统组态配置	0.8 ～ 51.2ms	R		√	√
SD39	主单元 X0 ～ X17 输入滤波常数	可通过系统组态配置	0.8 ～ 51.2ms	R	√	√	√
SD40	扩展 I/O 0 输入滤波常数	可通过系统组态配置	0.8 ～ 51.2ms	R		√	√
SD41	扩展 I/O 1 输入滤波常数	可通过系统组态配置	0.8 ～ 51.2ms	R		√	√
SD42	扩展 I/O 2 输入滤波常数	可通过系统组态配置	0.8 ～ 51.2ms	R		√	√
SD43	扩展 I/O 3 输入滤波常数	可通过系统组态配置	0.8 ～ 51.2ms	R		√	√
SD44	扩展 I/O 4 输入滤波常数	可通过系统组态配置	0.8 ～ 51.2ms	R		√	√
SD45	扩展 I/O 5 输入滤波常数	可通过系统组态配置	0.8 ～ 51.2ms	R		√	√
SD46	扩展 I/O 6 输入滤波常数	可通过系统组态配置	0.8 ～ 51.2ms	R		√	√
SD47	扩展 I/O 7 输入滤波常数	可通过系统组态配置	0.8 ～ 51.2ms	R		√	√
SD48	扩展 I/O 8 输入滤波常数	可通过系统组态配置	0.8 ～ 51.2ms	R		√	√
SD49	扩展 I/O 9 输入滤波常数	可通过系统组态配置	0.8 ～ 51.2ms	R		√	√

5. 高速脉冲输出监控

地址	名称	R/W	1H	2H	2V
SD50/SD52/SD70 /SD72/SD74/SD76 /SD78/SD80	PLSR/PLSY 指令输出 Y0 ～ Y7 脉冲总数（低位）	R/W	√	√	√
SD51/DSD53/SD71 /SD73/SD75/SD77 /SD79/SD81	PLSR/PLSY 指令输出 Y0 ～ Y7 脉冲总数（高位）	R/W	√	√	√
SD52	PLSR/PLSY 指令输出 Y1 脉冲总数（低位）	R/W	√	√	√
SD53	PLSR/PLSY 指令输出 Y1 脉冲总数（高位）	R/W	√	√	√
SD54	MX2H-1PG-BD 脉冲输出总数（低位）	R/W		√	√
SD55	MX2H-1PG-BD 脉冲输出总数（高位）	R/W		√	√
SD56	PLS 指令输出 Y0 指令目前执行到的段数	R	√	√	√
SD57	PLS 指令输出 Y1 指令目前执行到的段数	R	√	√	√
SD58	MX2H-1PG-BD PLS 指令目前执行到的段数	R		√	√

6. 定时中断周期

地址	名称	动作与功能	范围	R/W	1H	2H	2V
SD66	定时中断 0 周期设置值	当值不在 1 ～ 32767 范围内时该中断不触发	1 ～ 32767ms	R/W	√	√	√
SD67	定时中断 1 周期设置值	当值不在 1 ～ 32767 范围内时该中断不触发	1 ～ 32767ms	R/W	√	√	√
SD68	定时中断 2 周期设置值	当值不在 1 ～ 32767 范围内时该中断不触发	1 ～ 32767ms	R/W	√	√	√
系统在处理用户定时中断时有 ±1ms 误差，为了保证定时中断能正常工作，建议用户定时中断周期设置值≥ 2ms。							

7. 实时时钟

地址	名称	动作与功能	范围	R/W	1H	2H	2V
SD100	年	实时时钟用	2000 ～ 2099 年	R	√	√	
SD101	月	实时时钟用	1 ～ 12 月	R	√	√	
SD102	日	实时时钟用	1 ～ 31 日	R	√	√	
SD103	时	实时时钟用	0 ～ 23 时	R	√	√	
SD104	分	实时时钟用	0 ～ 59 分	R	√	√	
SD105	秒	实时时钟用	0 ～ 59 秒	R	√	√	
SD106	星期	实时时钟用	0（周日）～ 6（周六）	R	√	√	

8. PORT0 口接收控制及状态（通讯口 0）

<table>
<tr><th>地址</th><th colspan="2">名称</th><th>寄存器功能</th><th>范围</th><th>R/W</th><th>1H</th><th>2H</th><th>2V</th></tr>
<tr><td rowspan="13">SD110</td><td rowspan="13">PORT0 模式状态字</td><td>SD110.0 ～ SD110.2
端口波特率</td><td>b2 b1 b0
0 0 0：1,200 波特率
0 0 1：2,400 波特率
0 1 0：4,800 波特率
0 1 1：9,600 波特率
1 0 0：19,200 波特率
1 0 1：38,400 波特率
1 1 0：57,600 波特率
1 1 1：115,200 波特率</td><td rowspan="13"></td><td rowspan="13">R/W</td><td rowspan="13">√</td><td rowspan="13">√</td><td rowspan="13">√</td></tr>
<tr><td>SD110.3 停止位</td><td>0：1 位停止位；　1：2 位停止位</td></tr>
<tr><td>SD110.4 奇偶校验</td><td>0：偶校验；　1：奇校验</td></tr>
<tr><td>SD110.5 奇偶校验允许</td><td>0：不校验；　1：校验</td></tr>
<tr><td>SD110.6 字符数据位</td><td>每个字符的数据位
0：8 位字符 ；　1：7 位字符</td></tr>
<tr><td>SD110.7 自由口
接收开始字符模式</td><td>1：有特定起始字符
0：无特定起始字符</td></tr>
<tr><td>SD110.8 自由端口接收
结束字符模式</td><td>1：有特定起始字符
0：无特定起始字符</td></tr>
<tr><td>SD110.9 自由口字符间
超时有效</td><td>1：有字符间超时有效
0：无字符间超时有效</td></tr>
<tr><td>SD110.10 自由口帧间
超时有效</td><td>1：有帧间超时
0：无帧间超时</td></tr>
<tr><td>SD110.11</td><td>保留</td></tr>
<tr><td>SD110.12 高低字节有效</td><td>0：字元件低字节有效
1：字元件高低字节有效</td></tr>
<tr><td>SD110.13 ～ SD110.15</td><td>保留</td></tr>
</table>

地址	名称	寄存器功能	范围	R/W	1H	2H	2V
SD111	PORT0 开始字符			R/W	√	√	√
SD112	PORT0 结束字符			R/W	√	√	√
SD113	PORT0 字符间超时时间	默认 0ms（忽略字符间超时）	1 ～ 32767ms	R/W	√	√	√
SD114	PORT0 帧超时时间	默认 0ms（忽略帧超时）	1 ～ 32767ms	R/W	√	√	√
SD115	PORT0 接收完成信息代码	第 0 位：用户终止接收置位 第 1 位：收到指定结束字置位 第 2 位：收到最大字符数置位 第 3 位：字符间超时置位 第 4 位：（帧）接收超时置位 第 5 位：奇偶检验错误，置位 第 6 ～ 15 位：保留，用户可忽略		R	√	√	√
SD116	PORT0 当前收到的字符			R	√	√	√
SD117	PORT0 当前收到的字符总数			R	√	√	√
SD118	PORT0 当前发送的字符			R	√	√	√

9. PORT1 口接收控制及状态（通讯口 1）

地址	名称		寄存器功能	范围	R/W	1H	2H	2V
SD120	PORT1 模式状态字	SD120.0 ～ SD120.2 端口波特率	b2 b1 b0 0 0 0： 1,200 波特率 0 0 1： 2,400 波特率 0 1 0： 4,800 波特率 0 1 1： 9,600 波特率 1 0 0： 19,200 波特率 1 0 1： 38,400 波特率 1 1 0： 57,600 波特率 1 1 1：115,200 波特率		R/W	√	√	√
		SD120.3 停止位	0：1 位停止位；1：2 位停止位					
		SD120.4 奇偶校验	0：偶校验； 1：奇校验					
		SD120.5 奇偶校验允许	0：不校验； 1：校验					
		SD120.6 字符数据位	每个字符的数据位 0：8 位字符 1：7 位字符					
		SD120.7 自由口接收开始字符模式	1：有特定起始字符 0：无特定起始字符					
		SD120.8 自由端口接收结束字符模式	1：有特定起始字符 0：无特定起始字符					
		SD120.9 自由口字符间超时有效	1：有字符间超时有效 0：无字符间超时有效					
		SD120.10 自由口帧间超时有效	1：有帧间超时 0：无帧间超时					
		SD120.11	保留					
		SD120.12 高低字节有效	0：字元件低字节有效 1：字元件高低字节有效					
		SD120.13 ～ SD120.15	保留					
SD121	PORT1 开始字符				R/W	√	√	√
SD122	PORT1 结束字符				R/W	√	√	√
SD123	PORT1 字符间超时时间		默认 0ms（忽略字符间超时）	1 ～ 32767ms	R/W	√	√	√
SD124	PORT1 帧超时时间		默认 0ms（忽略帧超时）	1 ～ 32767ms	R/W	√	√	√

地址	名称	寄存器功能	范围	R/W	1H	2H	2V
SD125	PORT1 接收完成信息代码	第 0 位：用户终止接收置位 第 1 位：收到指定结束字置位 第 2 位：收到最大字符数置位 第 3 位：字符间超时置位 第 4 位：（帧）接收超时置位 第 5 位：奇偶检验错误，置位 第 6 ～ 15 位：保留，用户可忽略		R	√	√	√
SD126	PORT1 当前收到的字符			R	√	√	√
SD127	PORT1 当前收到的字符总数			R	√	√	√
SD128	PORT1 当前发送的字符			R	√	√	√

10. PORT2 口接收控制及状态（通讯口 2）

<table>
<tr><th colspan="2">地址</th><th>名称</th><th>寄存器功能</th><th>范围</th><th>R/W</th><th>1H</th><th>2H</th><th>2V</th></tr>
<tr><td rowspan="13">SD130</td><td rowspan="13">PORT2
模
式
状
态
字</td><td>SD130.0 ～ SD130.2
端口波特率</td><td>b2 b1 b0
0 0 0： 1,200 波特率
0 0 1： 2,400 波特率
0 1 0： 4,800 波特率
0 1 1： 9,600 波特率
1 0 0： 19,200 波特率
1 0 1： 38,400 波特率
1 1 0： 57,600 波特率
1 1 1：115,200 波特率</td><td rowspan="13"></td><td rowspan="13">R/W</td><td rowspan="13"></td><td rowspan="13">√</td><td rowspan="13">√</td></tr>
<tr><td>SD130.3 停止位</td><td>0：1 位停止位；1：2 位停止位</td></tr>
<tr><td>SD130.4 奇偶校验</td><td>0：偶校验； 1：奇校验</td></tr>
<tr><td>SD130.5 奇偶校验允许</td><td>0：不校验； 1：校验</td></tr>
<tr><td>SD130.6 字符数据位</td><td>每个字符的数据位
0：8 位字符 1：7 位字符</td></tr>
<tr><td>SD130.7 自由口
接收开始字符模式</td><td>1：有特定起始字符
0：无特定起始字符</td></tr>
<tr><td>SD130.8 自由端口接收
结束字符模式</td><td>1：有特定起始字符
0：无特定起始字符</td></tr>
<tr><td>SD130.9 自由口
字符间超时有效</td><td>1：有字符间超时有效
0：无字符间超时有效</td></tr>
<tr><td>SD130.10 自由
口帧间超时有效</td><td>1：有帧间超时
0：无帧间超时</td></tr>
<tr><td>SD130.11</td><td>保留</td></tr>
<tr><td>SD130.12 高低字节有效</td><td>0：字元件低字节有效
1：字元件高低字节有效</td></tr>
<tr><td>SD130.13 ～ SD130.15</td><td>保留</td></tr>
<tr><td colspan="0" style="display:none"></td></tr>
<tr><td>SD131</td><td colspan="2">PORT2 开始字符</td><td></td><td></td><td>R/W</td><td></td><td>√</td><td>√</td></tr>
<tr><td>SD132</td><td colspan="2">PORT2 结束字符</td><td></td><td></td><td>R/W</td><td></td><td>√</td><td>√</td></tr>
<tr><td>SD133</td><td colspan="2">PORT2 字符间超时时间</td><td>默认 0ms（忽略字符间超时）</td><td>1 ～ 32767ms</td><td>R/W</td><td></td><td>√</td><td>√</td></tr>
<tr><td>SD134</td><td colspan="2">PORT2 帧超时时间</td><td>默认 0ms（忽略帧超时）</td><td>1 ～ 32767ms</td><td>R/W</td><td></td><td>√</td><td>√</td></tr>
<tr><td>SD135</td><td colspan="2">PORT2 接收完成信息代码</td><td>第 0 位：用户终止接收置位
第 1 位：收到指定结束字置位
第 2 位：收到最大字符数置位
第 3 位：字符间超时置位
第 4 位：（帧）接收超时置位
第 5 位：奇偶检验错误，置位
第 6 ～ 15 位：保留，用户可忽略</td><td></td><td>R</td><td></td><td>√</td><td>√</td></tr>
</table>

地址	名称	寄存器功能	范围	R/W	1H	2H	2V
SD136	PORT2 当前收到的字符			R		√	√
SD137	PORT2 当前收到的字符总数			R		√	√
SD138	PORT2 当前发送的字符			R		√	√

11. Modbus/MXlink 设定

地址	名称	范围	R/W	1H	2H	2V
SD140	PORT 0 本站站号	Modbus（1 ～ 32）	R	√	√	√
SD141	PORT 0 最大超时时间		R	√	√	√
SD142	PORT 0 重试次数	0 ～ 100	R	√	√	√
SD145	PORT 1 本站站号	Modbus（1 ～ 32）MXLink（0 ～ 15）	R	√	√	√
SD146	PORT 1 最大超时时间		R	√	√	√
SD147	PORT 1 重试次数	0 ～ 100	R	√	√	√
SD149	PORT 1 Modbus 主站的错误代码		R	√	√	√
SD150	PORT 2 本站站号	Modbus（1 ～ 32）	R		√	√
SD151	PORT 2 最大超时时间		R		√	√
SD152	PORT 2 重试次数	0 ～ 100	R		√	√
SD154	PORT 2 Modbus 主站的错误代码		R		√	√
SD155	PORT 3 本站站号设定	Modbus（1 ～ 32）	R	√	√	√
SD156	PORT 3 最大超时时间设定		R	√	√	√
SD157	PORT 3 重试次数	0 ～ 100	R	√	√	√
SD159	PORT 3 Modbus 主站的错误代码		R	√	√	√
SD160	CAN0 本站号	MXLink（0 ～ 15）	R	√	√	√
SD161	CAN0 最大超时时间		R	√	√	√
SD162	CAN0 重试次数		R	√	√	√
SD163	CAN0 发送超时时间	0 ～ 12000（ms）	R	√	√	√
SD164	CAN0 接收超时时间	0 ～ 12000（ms）	R	√	√	√

12. 以太网卡

地址	名称	R/W	1H	2H	2V
SD165	本机以太网卡站号	R		√	√
SD166	本机以太网卡 IP 地址 IP1	R		√	√
SD167	本机以太网卡 IP 地址 IP2	R		√	√
SD168	本机以太网卡 IP 地址 IP3	R		√	√
SD169	本机以太网卡 IP 地址 IP4	R		√	√
SD170	本机以太网卡子网掩码 IP1	R		√	√
SD171	本机以太网卡子网掩码 IP2	R		√	√
SD172	本机以太网卡子网掩码 IP3	R		√	√
SD173	本机以太网卡子网掩码 IP4	R		√	√
SD174	本机以太网卡网关 IP1	R		√	√
SD175	本机以太网卡网关 IP2	R		√	√
SD176	本机以太网卡网关 IP3	R		√	√
SD177	本机以太网卡网关 IP4	R		√	√
SD178	本机以太网卡端口号	R		√	√
SD179	本机以太网卡单板软件版本号	R		√	√

13. DHSP 和 DHST 指令使用

地址	名称	R/W	1H	2H	2V
SD180	DHSP 表格比较输出数据（低位）	R/W	√	√	√
SD181	DHSP 表格比较输出数据（高位）	R/W	√	√	√
SD182	DHST 或 DHSP 表格比较数据（低位）	R/W	√	√	√
SD183	DHST 或 DHSP 表格比较数据（高位）	R/W	√	√	√
SD184	当前正执行表格的记录号	R/W	√	√	√

14. 增强定位指令

地址	名称	寄存器功能	R/W	1H	2H	2V
SD198	Y0 ～ Y3 输出轴的中断信号软元件指定（4 位代表一个轴）	指定作为中断输入的输入编号（X0 ～ X7），或者指定用户中断指令软元件	R/W		√	√
SD199	Y4 ～ Y7 输出轴的中断信号软元件指定（MX2H-1PG: 高四位）	指定作为中断输入的输入编号（X0 ～ X7），或者指定用户中断指令软元件	R/W		√	√
SD200/SD220/SD260 /SD280/SD300/SD320 /SD340/SD360	Y0 ～ Y7 定位输出当前值（低位）	作为 Y 输出定位指令的当前值数据寄存器	R/W		√	√
SD201/SD221/SD261 /SD281/SD301/SD321 /SD341/SD361	Y0 ～ Y7 定位输出当前值（高位）	作为 Y 输出定位指令的当前值数据寄存器	R/W		√	√
SD202/SD222/SD282 /SD302/SD322/SD342 /SD362/SD382	Y0 ～ Y7 定位输出最高速度(10～100000)(低位)	执行 DZRN，PLSV，DDRVI，DDRVA，DSZR，DDVIT 指令时的最高速度（10 ～ 100000）	R/W		√	√
SD203/SD223/SD283 /SD303/SD323/SD343 /SD363/SD383	Y0 ～ Y7 定位输出最高速度(10～100000)(高位)	执行 DZRN，PLSV，DDRVI，DDRVA，DSZR，DDVIT 指令时的最高速度（10 ～ 100000）	R/W		√	√
SD204/SD224/SD284 /SD304/SD324/SD344 /SD364/SD384	Y0 ～ Y7 定位输出基底速度（最高速度的 1/10 以下）	执行 DZRN，PLSV，DDRVI，DDRVA，DSZR，DDVIT 指令时的基底速度 (最高速度的 1/10 以下)	R/W		√	√
SD205/SD225/SD285 /SD305/SD325/SD345 /SD365/SD385	Y0 ～ Y7 定位输出加速时间	执行 DZRN，DDRVI，DDRVA，DDSZR，DDVIT 指令时由基底速度上升到最高速度的加速时间，范围：2 ～ 5000，单位：ms	R/W		√	√
SD206/SD226/SD286 /SD306/SD326/SD346 /SD366/SD386	Y0 ～ Y7 定位输出减速时间	执行 DZRN，DDRVI，DDRVA，DDSZR，DDVIT 指令时由基底速度上升到最高速度的减速时间，范围：2 ～ 5000，单位：ms	R/W		√	√
SD207/SD227/SD287 /SD307/SD327/SD347 /SD367/SD387	Y0 ～ Y7 爬行速度	适用于 DZRN/DSZR，作为执行该指令时的爬行速度	R/W		√	√
SD208/SD228/SD288 /SD308/SD328/SD348 /SD368/SD388	Y0 ～ Y7 原点回归速度（低位）	适用于 DZRN/DSZR，作为执行该指令时的原点回归速度	R/W		√	√
SD209/SD229/SD289 /SD309/SD329/SD349 /SD369/SD389	Y0 ～ Y7 原点回归速度（高位）		R/W		√	√

地址	名称	寄存器功能	R/W	1H	2H	2V
SD210/SD230/SD290 /SD310/SD330/SD350 /SD370/SD390	Y0 ～ Y7 清零信号软元件指定	SM281 置位时，使用该元件中的数值 N 对应的 Y 元件 Y（N）表示清零信号	R/W		√	√
SD211/SD231/SD291 /SD311/SD331/SD351 /SD371/SD391	Y0 ～ Y7 配置码	动作模式及输出关系： Bit0：0- 绝对位置　　1- 相对地址 Bit5 ～ Bit4：00- 脉冲 + 方向 01- 正反向脉冲 10-AB 脉冲	R/W			√

15. CANbus 单帧接收指令

地址	名称	R/W	1H	2H	2V
SD240	CAN0 接收中断当前接收帧 ID（低位）	R	√	√	√
SD241	CAN0 接收中断当前接收帧 ID（高位）	R	√	√	√
SD242	CAN0 接收中断当前接收帧长度	R	√	√	√
SD243	CAN0 接收中断当前接收帧第 1 个字符	R	√	√	√
SD244	CAN0 接收中断当前接收帧第 2 个字符	R	√	√	√
SD245	CAN0 接收中断当前接收帧第 3 个字符	R	√	√	√
SD246	CAN0 接收中断当前接收帧第 4 个字符	R	√	√	√
SD247	CAN0 接收中断当前接收帧第 5 个字符	R	√	√	√
SD248	CAN0 接收中断当前接收帧第 6 个字符	R	√	√	√
SD249	CAN0 接收中断当前接收帧第 7 个字符	R	√	√	√
SD250	CAN0 接收中断当前接收帧第 8 个字符	R	√	√	√

16. 信号报警器指令

地址	名称	寄存器功能	R/W	1H	2H	2V
SD401	信号报警器 ON 状态最小值编号	保持 S900 ～ S1023 中动作的最小编号	R		√	√

17. 实时时钟指令

地址	名称	R/W	1H	2H	2V
SD430	定时器时钟输出 1 计数值	R	√	√	√
SD431	定时器时钟输出 2 计数值	R	√	√	√
SD432	定时器时钟输出 3 计数值	R	√	√	√
SD433	定时器时钟输出 4 计数值	R	√	√	√
SD434	定时器时钟输出 5 计数值	R	√	√	√

18. PLC 工作状态数据

地址	名称	R/W	1H	2H	2H
SD510	PLC 序列号（低位）	R	√	√	√
SD511	PLC 序列号（高位）	R	√	√	√

附录四 ASCII 码字符编码表

ASCII 码为 7 位代码，使用 16 进制表示，如下表所示：

字符	ASCII（16 进制数）	字符	ASCII（16 进制数）	字符	ASCII（16 进制数）
NUL	00	+	2B	V	56
SOH	01	，(逗号)	2C	W	57
STX	02	－(减号)	2D	X	58
ETX	03	.	2E	Y	59
EOT	04	/	2F	Z	5A
ENQ	05	0	30	[	5B
ACK	06	1	31	\	5C
BEL	07	2	32	]	5D
BS	08	3	33	^	5E
HT	09	4	34	-(下划线)	5F
LF	0A	5	35	`(顿号)	60
VT	0B	6	36	a	61
FF	0C	7	37	b	62
CR	0D	8	38	c	63
SO	0E	9	39	d	64
SI	0F	：	3A	e	65
DLE	10	；	3B	f	66
DC1	11	<	3C	g	67
DC2	12	=	3D	h	68
DC3	13	>	3E	i	69
DC4	14	?	3F	j	6A
NAK	15	@	40	k	6B
SYN	16	A	41	l	6C
ETB	17	B	42	m	6D
CAN	18	C	43	n	6E
EM	19	D	44	o	6F
SUB	1A	E	45	p	70
ESC	1B	F	46	q	71
FS	1C	G	47	r	72
GS	1D	H	48	s	73
RS	1E	I	49	t	74
US	1F	J	4A	u	75
SPACE	20	K	4B	v	76
!	21	L	4C	w	77
“	22	M	4D	x	78
#	23	N	4E	y	79
$	24	O	4F	z	7A
%	25	P	50	{	7B
&	26	Q	51	|(竖线)	7C
'(单引号)	27	R	52	}	7D
(	28	S	53	～	7E
)	29	T	54	DEL	7F
*	2A	U	55		

威海麦科电气技术有限公司

地址：威海市高技区丹东路85号

邮编：264209

电话：0631-5808951

传真：0631-5801751

信息交流：support@mikom.com.cn

技术支持热线：400-680-8951

MIKOM

Electrical Technology

www.mikom.com.cn