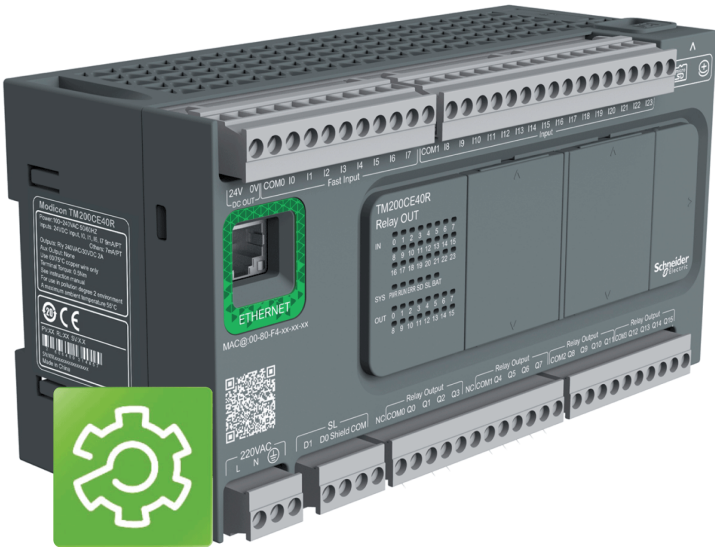
Modicon M100/M200 可编程控制器

编程指南

03/2018



EIO0000002020.04

www.schneider-electric.com

Schneider Electric

本文档中提供的信息包含有关此处所涉及产品之性能的一般说明和/或技术特性。本文档并非用于(也不代替) 确定这些产品对于特定用户应用场合的适用性或可靠性。任何此类用户或设备集成商都有责任就相关特定应用场合或使用方面对产品执行适当且完整的风险分析、评估和测试。Schneider Electric 或其任何附属机构或子公司对于误用此处包含的信息而产生的后果概不负责。如果您有关于改进或更正此出版物的任何建议、或者从中发现错误、请通知我们。

本手册可用于法律所界定的个人以及非商业用途。 在未获得施耐德电气书面授权的情况下，不得翻印传播本手册全部或部分相关内容、亦不可建立任何有关本手册或其内容的超文本链接。施耐德电气不对个人和非商业机构进行非独占许可以外的授权或许可。 请遵照本手册或其内容原义并自负风险。与此有关的所有其他权利均由施耐德电气保留。

在安装和使用本产品时，必须遵守国家、地区和当地的所有相关的安全法规。出于安全方面的考虑和为了帮助确保符合归档的系统数据，只允许制造商对各个组件进行维修。

当设备用于具有技术安全要求的应用场合时，必须遵守有关的使用说明。

未能使用施耐德电气软件或认可的软件配合我们的硬件，则可能导致人身伤害、设备损坏或不正确的运行结果。

不遵守此信息可能导致人身伤害或设备损坏。

© 2018 Schneider Electric。 保留所有权利。

目录

	安全信息	9
	关于本书	11
第I部分	**简介**	13
第1章	**关于 Modicon M100/M200 可编程控制器**	15
	M100/M200 可编程控制器 描述	15
第2章	**配置功能**	22
2.1	对象	23
	对象概述	24
	对象类型	25
	对象寻址	29
	最大对象数	32
2.2	任务结构	35
	任务和扫描模式	36
	最大任务数和优先级	38
2.3	控制器状态和行为	39
	控制器状态示意图	40
	控制器状态描述	41
	控制器状态转换	44
	持续性变量	47
	输出行为	49
2.4	后配置	52
	后配置	53
	后配置文件管理	54
第II部分	**配置 M100/M200 可编程控制器**	57
第3章	**如何配置控制器**	59
	建立配置	60
	可选 I/O 扩展模块	65
	配置 M100/M200 可编程控制器	69
	使用 Executive Loader Wizard 更新固件	70
第4章	**内置输入/输出配置**	72
4.1	配置数字量输入	73
	配置数字量输入	73
4.2	配置数字量输出	76
	配置数字量输出	76

4.3 配置脉冲发生器 78
配置脉冲发生器 78
4.4 配置频率发生器 87
配置频率发生器 (%FREQGEN) 87
4.5 配置高速计数器 89
配置高速计数器 90
配置单相和双相 94
配置频率计 99
第5章 I/O 总线配置 101
I/O 配置概述 102
配置扩展模块 105
第6章 扩展板配置 107
6.1 扩展板配置一般信息 108
一般描述 109
在配置中使用扩展板 111
配置扩展板 112
6.2 TMCR2••• 扩展板配置 113
TMCR2DM4U 114
TMCR2AI2 115
TMCR2AQ2C 117
TMCR2AQ2V 118
TMCR2AM3 119
TMCR2TI2 121
TMCR2SL1 124
TMCR2SL1A 129
TMCR2模拟量扩展板诊断 134
第7章 TM3R扩展模块配置 135
I/O 配置一般做法 136
配置 TM3R 数字量 I/O 模块 137
在配置中使用 I/O 模块 138
配置数字量 I/O 140
TM3 模拟量 I/O 模块诊断 143
第8章 内置通讯配置 145
8.1 以太网配置 146
配置以太网网络 147
配置 Modbus TCP 153

8.2 串行线路配置 163
配置串行线路 164
配置 Modbus Serial IOScanner 170
在 Modbus Serial IOScanner 上添加设备 171
8.3 支持的 Modbus 功能代码 178
支持的 Modbus 功能代码 178
第9章 Micro SD 卡 181
文件管理操作 182
SD 卡支持的文件类型 184
克隆管理 186
固件管理 188
应用程序管理 193
后配置管理 197
错误日志管理 205
存储器管理：备份和恢复控制器存储器 211
第III部分 对 M100/M200 可编程控制器 进行编程 217
第10章 如何使用源代码示例 219
如何使用源代码示例 219
第11章 I/O 对象 223
数字量输入 (%I) 224
数字量输出 (%Q) 225
模拟量输入 (%IW) 226
模拟量输出 (%QW) 227
第12章 功能块 229
12.1 快速计数器 (%FC) 230
描述 231
配置 233
编程示例 235
12.2 高速计数器 (%HSC) 236
描述 237
计数模式下的高速计数器 241
One-shot 计数模式 249
Modulo-loop 计数模式 250
频率计模式下的高速计数器 252
12.3 脉冲 (%PLS) 255
描述 256
配置 258
编程示例 261

12.4 驱动器功能块 (%DRV) 262
描述 263
驱动器和逻辑控制器状态 265
添加驱动器功能块 267
功能块配置 269
MC_Jog_ATV：启动点动模式 270
MC_Power_ATV：启用/禁用电源级 273
MC_MoveVel_ATV：以指定速度移动 275
MC_Stop_ATV：停止移动 278
MC_ReadStatus_ATV：读取设备状态 280
MC_ReadMotionState_ATV：读取运动状态 282
MC_Reset_ATV：确认和复位错误 285
错误代码 287
12.5 脉冲宽度调制 (%PWM) 291
描述 292
功能块配置 293
编程示例 296
第13章 网络对象 297
输入寄存器对象 (%QWM) 298
输出寄存器 (Modbus TCP) 对象 (%IWM) 299
数字量输入 (IOScanner) 对象 (%IN) 300
数字量输出 (IOScanner) 对象 (%QN) 301
输入寄存器 (IOScanner) 对象 (%IWN) 303
输出寄存器 (IOScanner) 对象 (%QWN) 304
Modbus IOScanner 网络诊断代码 (%IWNS) 306
第14章 脉冲串输出 (%PTO) 307
14.1 描述 308
脉冲串输出 (PTO) 309
加速度/减速度斜坡 312
探测器事件 314
反向间隙补偿 316
定位限制 317
14.2 配置 319
PTO 配置 320
脉冲输出模式 321
Motion Task Table 322

14.3 编程 329
添加/删除功能块 330
PTO 功能块 331
14.4 原点模式 335
原点模式 336
位置设置 338
长参考 339
短参考无反向 341
短参考反向 343
INDEX 短参考 345
原点补偿 346
14.5 数据参数 347
功能块对象代码 347
14.6 操作模式 351
运动状态图 352
缓冲模式 354
14.7 **运动**功能块 356
MC_MotionTask_PTO 功能块 357
MC_Power_PTO 功能块 361
MC_MoveVel_PTO 功能块 364
MC_MoveRel_PTO 功能块 368
MC_MoveAbs_PTO 功能块 372
MC_Home_PTO 功能块 376
MC_SetPos_PTO 功能块 379
MC_Stop_PTO 功能块 381
MC_Halt_PTO 功能块 384
14.8 **管理**功能块 387
MC_ReadActVel_PTO 功能块 388
MC_ReadActPos_PTO 功能块 390
MC_ReadSts_PTO 功能块 392
MC_ReadMotionState_PTO 功能块 394
MC_ReadAxisError_PTO 功能块 396
MC_Reset_PTO 功能块 398
MC_TouchProbe_PTO 功能块 400
MC_AbortTrigger_PTO 功能块 402
MC_ReadPar_PTO 功能块 404
MC_WritePar_PTO 功能块 406

第15章 频率发生器 409
描述 410
配置 412
第16章 PID 功能 413
16.1 PID 操作模式 414
PID 操作模式 414
16.2 PID 自整定配置 416
PID 自整定配置 416
16.3 PID 标准配置 419
PID 字地址配置 420
使用自整定 (AT) 进行 PID 调节 423
手动模式. 426
确定采样周期 (Ts) 428
16.4 PID 助手 430
访问 PID 助手. 431
常规”选项卡 432
输入选项卡 435
PID 选项卡 436
AT 选项卡. 437
输出选项卡 439
16.5 PID 编程. 441
描述 442
编程和配置 444
PID 状态和检测到的错误代码. 445
16.6 PID 参数. 447
PID 参数的作用和影响 448
PID 参数调整方法. 450
第17章 系统对象 453
系统位 (%S) 454
系统字 (%SW) 463
输入通道状态 (%IWS) 482
输出通道状态 (%QWS) 484
术语表 487
索引 491

安全信息

重要信息

声明

在试图安装、操作、维修或维护设备之前，请仔细阅读下述说明并通过查看来熟悉设备。下述特定信息可能会在本文其他地方或设备上出现，提示用户潜在的危险，或者提醒注意有关阐明或简化某一过程的信息。

在“危险”或“警告”标签上添加此符号表示存在触电危险，如果不遵守使用说明，会导致人身伤害。

这是提醒注意安全的符号。提醒用户可能存在人身伤害的危险。请遵守所有带此符号的安全注意事项，以避免可能的人身伤害甚至死亡。

危险

危险表示若不加以避免，将**会导致**严重人身伤害甚至死亡的危险情况。

警告

警告表示若不加以避免，可能**会导致**严重人身伤害甚至死亡的危险情况。

小心

小心表示若不加以避免，可能**会导致**轻微或中度人身伤害的危险情况。

注意

注意用于表示与人身伤害无关的危害。

请注意

电气设备的安装、操作、维修和维护工作仅限于有资质的人员执行。施耐德电气不承担由于使用本资料所引起的任何后果。

有资质的人员是指掌握与电气设备的制造和操作及其安装相关的技能和知识的人员，他们经过安全培训能够发现和避免相关的危险。

关于本书

概览

文档范围

本文档介绍 SoMachine Basic 的 Modicon M100/M200 可编程控制器 的配置和编程。有关其他信息，请参见 SoMachine Basic 在线帮助内的独立文档。

有效性说明

本文档已随 SoMachine Basic V1.6 的发布进行了更新。

访问 Schneider Electric 主页 *www.schneider-electric.com/cn/zh/*。

本手册中介绍的特性应该与在线显示的那些特性相同。依据我们的持续改进政策，我们将不断修订内容，使其更加清楚明了，更加准确。如果您发现手册和在线信息之间存在差异，请以在线信息为准。

相关的文件

文件名称	参考编号
SoMachine Basic - 操作指南	*EIO0000001354 (ENG)* *EIO0000001359 (CHS)*
SoMachine Basic 通用功能 - 库指南	*EIO0000001474 (ENG)* *EIO0000001479 (CHS)*
Modicon M100/M200 可编程控制器 - 硬件指南	EIO0000002023 (ENG) EIO0000002024 (CHS)
Modicon TM3 扩展模块配置 - 编程指南	*EIO0000001396 (ENG)* *EIO0000001401 (CHS)*
Modicon TM2 扩展模块配置 - 编程指南	*EIO0000001390 (ENG)* *EIO0000001395 (CHS)*

您可以从我们的网站下载这些技术出版物和其它技术信息，网址是：https://www.schneider-electric.com/en/download

关于产品的资讯

⚠ 警告

失去控制

- 任何控制方案的设计者都必须考虑到控制路径可能出现故障的情况，并为某些关键控制功能提供一种方法，使其在出现路径故障时以及出现路径故障后恢复至安全状态。这些关键控制功能包括紧急停止、越程停止、断电重启以及类似的安全措施。
- 对于关键控制功能，必须提供单独或冗余的控制路径。
- 系统控制路径可包括通讯链路。必须对暗含的无法预料的传输延迟或链路失效问题加以考虑。
- 遵守所有事故预防规定和当地的安全指南。[1]
- 为了保证正确运行，在投入使用前，必须对设备的每次执行情况分别进行全面测试。

不遵循上述说明可能导致人员伤亡或设备损坏。

[1] 有关详细信息，请参阅 NEMA ICS 1.1（最新版）中的“安全指导原则 - 固态控制器的应用、安装和维护”以及 NEMA ICS 7.1（最新版）中的“结构安全标准及可调速驱动系统的选择、安装与操作指南”或您特定地区的类似规定。

⚠ 警告

意外的设备操作

- 仅使用 Schneider Electric 认可的可与本设备配合使用的软件。
- 每次更改物理硬件配置后，请更新应用程序。

不遵循上述说明可能导致人员伤亡或设备损坏。

第I部分
简介

概述

本部分提供有关 Modicon M100/M200 可编程控制器 及其配置和编程功能的一般信息。

本部分包含了哪些内容？

本部分包括以下各章：

章	章节标题	页
1	关于 Modicon M100/M200 可编程控制器	15
2	配置功能	22

第1章
关于 Modicon M100/M200 可编程控制器

M100/M200 可编程控制器 描述

概述

M100/M200 可编程控制器 具有多种强大的功能，可在广泛的应用程序中使用。

软件配置、编程和调试使用 SoMachine Basic 操作指南 *(参见 SoMachine Basic, 操作指南)*和 SoMachine Basic 通用功能库指南 *(参见 SoMachine Basic, 通用功能库指南)*中描述的 SoMachine Basic 软件完成。

编程语言

M100/M200 可编程控制器 可通过 SoMachine Basic 软件对其进行配置和编程，该软件支持以下 IEC 61131-3 编程语言：

- IL：指令列表
- LD：梯形图
- Grafcet（列表）
- Grafcet (SFC)

电源

M100 可编程控制器 的电源为 100...240 Vac。

M200 可编程控制器 的电源为 24 Vdc 或 100...240 Vac。

实时时钟

M200 可编程控制器 包括一个实时时钟 (RTC) 系统 *(参见 Modicon M100/M200 可编程控制器, 硬件指南)*。

运行/停止

M100/M200 可编程控制器 可以通过以下方式在外部进行操作：

- 硬件运行/停止切换按钮 *(参见 Modicon M100/M200 可编程控制器, 硬件指南)*
- 通过软件配置中定义的专用数字量输入进行运行/停止 *(参见 Modicon M100/M200 可编程控制器, 硬件指南)*操作。有关详细信息，请参阅配置数字量输入 (参见第 *73* 页)。
- SoMachine Basic 软件。有关详细信息，请参阅 SoMachine Basic 操作指南 *(参见 SoMachine Basic, 操作指南)*。

存储器

下表描述了不同类型的存储器：

存储器类型		大小	作用...
RAM	适用于 TM100C••RN	RAM 存储器的 128 KB：32 KB 用于内部变量，另外 96 KB 用于应用程序和数据。	执行应用程序并包含数据.
	适用于 TM100C••R 和 TM200C•••	RAM 存储器的 512 KB：256 KB 用于内部变量，另外 256 KB 用于应用程序和数据。	
闪存		1 MB，其中 256 KB 可在发生断电时用于备份应用程序和数据。	保存应用程序。

内置输入/输出

根据控制器型号，提供以下内置 I/O 类型：

- 常规输入
- 与计数器关联的快速输入
- 常规晶体管漏极输出(PMW/PLS/PTO/FREQGEN)
- 与脉冲发生器关联的快速漏极晶体管输出
- 继电器输出

可移除存储器

M100/M200 可编程控制器 包含内置微型 SD 卡插槽。

M100/M200 可编程控制器 允许使用 micro SD 卡进行以下类型的文件管理：

- 克隆管理 (参见第 *186* 页)：备份逻辑控制器的应用程序、固件和后配置（如有）
- 固件管理 (参见第 *188* 页)：将固件更新直接下载至逻辑控制器或下载至 TM3 扩展模块
- 应用程序管理 (参见第 *193* 页)：备份并恢复逻辑控制器应用程序，或者将其复制到相同应用的另一个逻辑控制器
- 后配置管理 (参见第 *197* 页)：添加、更改或删除逻辑控制器的后配置文件
- 错误日志管理 (参见第 *205* 页)：备份或删除逻辑控制器的错误日志文件
- 存储管理 (参见第 *211* 页)：备份/从控制器恢复存储位和字

内置通讯功能

根据控制器型号，以下类型的通讯端口可用：

- Ethernet *(参见 Modicon M100/M200 可编程控制器, 硬件指南)*
- USB Mini-B *(参见 Modicon M100/M200 可编程控制器, 硬件指南)*
- 串行端口 1 *(参见 Modicon M100/M200 可编程控制器, 硬件指南)*

M100 可编程控制器 参考

型号	数字量输入	数字量输出	通讯端口	电源
TM100C16R	1 路常规输入(1) 4 路快速输入 (FC)(2) 4 路高速输入 (HSC)(3)	7 路继电器输出	1 个串行线路端口 1 个 USB 编程端口	100...240 Vac
TM100C24R	6 路常规输入(1) 4 路快速输入 (FC)(2) 4 路高速输入 (HSC)(3)	10 路继电器输出	1 个串行线路端口 1 个 USB 编程端口	100...240 Vac
TM100C40R	16 路常规输入(1) 4 路快速输入 (FC)(2) 4 路高速输入 (HSC)(3)	16 路继电器输出	1 个串行线路端口 1 个 USB 编程端口	100...240 Vac
TM100C16RN	3 路常规输入(1) 4 路快速输入 (FC)(2) 2 路高速输入 (HSC)(4)	7 路继电器输出	1 个串行线路端口 1 个 USB 编程端口	100...240 Vac
TM100C24RN	8 路常规输入(1) 4 路快速输入 (FC)(2) 2 路高速输入 (HSC)(4)	10 路继电器输出	1 个串行线路端口 1 个 USB 编程端口	100...240 Vac
TM100C32RN	14 路常规输入(1) 4 路快速输入 (FC)(2) 2 路高速输入 (HSC)(4)	12 路继电器输出	1 个串行线路端口 1 个 USB 编程端口	100...240 Vac

注意： 所有 M100 逻辑控制器都使用不可插拔螺钉端子块。

(1) 常规输入的最大频率为 5 kHz。
(2) 快速输入最大频率为 5 kHz。快速输入可用作针对计数或事件功能的常规输入或快速输入。
(3) 高速输入的最大频率为 100 kHz。
(4) 高速输入的最大频率为 60 kHz。

型号	数字量输入	数字量输出	通讯端口	电源
TM100C40RN	18 路常规输入(1) 4 路快速输入 (FC)(2) 2 路高速输入 (HSC)(4)	16 路继电器输出	1 个串行线路端口 1 个 USB 编程端口	100...240 Vac

注意： 所有 M100 逻辑控制器都使用不可插拔螺钉端子块。

(1) 常规输入的最大频率为 5 kHz。
(2) 快速输入最大频率为 5 kHz。快速输入可用作针对计数或事件功能的常规输入或快速输入。
(3) 高速输入的最大频率为 100 kHz。
(4) 高速输入的最大频率为 60 kHz。

M200 可编程控制器 参考

型号	数字量输入	数字量输出	通讯端口	电源
TM200C16R	1 路常规输入(1) 4 路快速输入 (FC)(2) 4 路高速输入 (HSC)(3)	7 路继电器输出	1 个串行线路端口 1 个 USB 编程端口	100...240 Vac
TM200C16T		源极输出 5 路常规晶体管输出 2 个快速输出 (PWM / PLS / PTO / FREQGEN)(4)	1 个串行线路端口 1 个 USB 编程端口	24 Vdc
TM200C16U		漏极输出 5 路常规晶体管输出 2 个快速输出 (PWM / PLS / PTO / FREQGEN)(4)	1 个串行线路端口 1 个 USB 编程端口	24 Vdc

注意： 所有 M200 逻辑控制器都使用可插拔螺钉端子块。

(1) 常规输入的最大频率为 5 kHz。
(2) 快速输入最大频率为 5 kHz。快速输入可用作针对计数或事件功能的常规输入或快速输入。
(3) 高速输入的最大频率为 100 kHz。
(4) 快速晶体管输出可以用作常规晶体管输出，或用于 PLS/PWM/PTO/FREQGEN 功能。

型号	数字量输入	数字量输出	通讯端口	电源
TM200C24R	6 路常规输入(1) 4 路快速输入 (FC)(2) 4 路高速输入 (HSC)(3)	10 路继电器输出	1 个串行线路端口 1 个 USB 编程端口	100...240 Vac
TM200CE24R			1 个串行线路端口 1 个 USB 编程端口 1 个以太网端口	
TM200C24T		源极输出 8 路常规晶体管输出 2 个快速输出 (PWM / PLS / PTO / FREQGEN)(4)	1 个串行线路端口 1 个 USB 编程端口	24 Vdc
TM200CE24T			1 个串行线路端口 1 个 USB 编程端口 1 个以太网端口	
TM200C24U		漏极输出 8 路常规晶体管输出 2 个快速输出 (PWM / PLS / PTO / FREQGEN)(4)	1 个串行线路端口 1 个 USB 编程端口	24 Vdc
TM200CE24U			1 个串行线路端口 1 个 USB 编程端口 1 个以太网端口	
TM200C32R	12 路常规输入(1) 4 路快速输入 (FC)(2) 4 路高速输入 (HSC)(3)	12 路继电器输出	1 个串行线路端口 1 个 USB 编程端口	100...240 Vac
TM200CE32R			1 个串行线路端口 1 个 USB 编程端口 1 个以太网端口	
TM200C40R	16 路常规输入(1) 4 路快速输入 (FC)(2) 4 路高速输入 (HSC)(3)	16 路继电器输出	1 个串行线路端口 1 个 USB 编程端口	100...240 Vac
TM200CE40R			1 个串行线路端口 1 个 USB 编程端口 1 个以太网端口	
TM200C40T		源极输出 14 路常规晶体管输出 2 个快速输出 (PWM / PLS / PTO / FREQGEN)(4)	1 个串行线路端口 1 个 USB 编程端口	24 Vdc
TM200CE40T			1 个串行线路端口 1 个 USB 编程端口 1 个以太网端口	
TM200C40U		漏极输出 14 路常规晶体管输出 2 个快速输出 (PWM / PLS / PTO / FREQGEN)(4)	1 个串行线路端口 1 个 USB 编程端口	24 Vdc
TM200CE40U			1 个串行线路端口 1 个 USB 编程端口 1 个以太网端口	

注意： 所有 M200 逻辑控制器都使用可插拔螺钉端子块。

(1) 常规输入的最大频率为 5 kHz。
(2) 快速输入最大频率为 5 kHz。快速输入可用作针对计数或事件功能的常规输入或快速输入。
(3) 高速输入的最大频率为 100 kHz。
(4) 快速晶体管输出可以用作常规晶体管输出，或用于 PLS/PWM/PTO/FREQGEN 功能。

型号	数字量输入	数字量输出	通讯端口	电源
TM200C60R	28 个常规输入(1) 4 路快速输入 (FC)(2) 4 路高速输入 (HSC)(3)	24 个继电器输出	1 个串行线路端口 1 个 USB 编程端口	100...240 Vac

注意： 所有 M200 逻辑控制器都使用可插拔螺钉端子块。

(1) 常规输入的最大频率为 5 kHz。
(2) 快速输入最大频率为 5 kHz。快速输入可用作针对计数或事件功能的常规输入或快速输入。
(3) 高速输入的最大频率为 100 kHz。
(4) 快速晶体管输出可以用作常规晶体管输出，或用于 PLS/PWM/PTO/FREQGEN 功能。

第2章
配置功能

简介

本章介绍有关 M100/M200 可编程控制器 存储器映射、任务、状态、行为、对象和功能的信息。本章中涉及的主题让操作员了解在 SoMachine Basic 中配置和编程控制器主要需要的 M100/M200 可编程控制器 特色规格。

本章包含了哪些内容？

本章包含了以下部分：

节	主题	页
2.1	对象	23
2.2	任务结构	35
2.3	控制器状态和行为	39
2.4	后配置	52

第2.1节
对象

本节包含了哪些内容？

本节包含了以下主题：

主题	页
对象概述	24
对象类型	25
对象寻址	29
最大对象数	32

对象概述

定义

在 SoMachine Basic 中，*对象*一词用于表示保留供应用程序使用的逻辑控制器存储器的区域。对象可能是：

- 简单软件变量，例如存储器位和字
- 数字量及模拟量输入和输出的地址
- 控制器内部变量，例如系统字和系统位
- 预定义的系统功能或功能块，例如定时器和计数器。

将为某些对象类型预先分配控制器存储器，或者在应用程序下载到逻辑控制器时自动分配。

分配存储器后，对象只能由程序进行寻址。使用前缀% 对对象进行寻址。例如，%MW12 是存储器字的地址，%Q0.3 是内置数字量输出的地址，而 %TM0 是 Timer 功能块的地址。

对象类型

简介

下表介绍了 M100/M200 可编程控制器 的语言对象类型：

对象类型	对象	对象功能	描述
存储器对象	%M	存储器位 *(参见 SoMachine Basic, 通用功能库指南)*	存储存储器位。
	%MW	字存储器 *(参见 SoMachine Basic, 通用功能库指南)*	存储 16 位存储器字。
	%MD	双字存储器 *(参见 SoMachine Basic, 通用功能库指南)*	存储 32 位存储器字。
	%MF	存储器浮动点 *(参见 SoMachine Basic, 通用功能库指南)*	将存储器浮点数存储在其表达式中具有小数的数学参数中。
	%KW	常量字 *(参见 SoMachine Basic, 通用功能库指南)*	存储 16 位常量字。
	%KD	双字常量 *(参见 SoMachine Basic, 通用功能库指南)*	存储 32 位常量字。
	%KF	常量浮点数 *(参见 SoMachine Basic, 通用功能库指南)*	将常量浮点数存储在其表达式中具有小数的数学参数中。
系统对象	%S	系统位 (参见第 *454* 页)	存储系统位。
	%SW	系统字 (参见第 *463* 页)	存储系统字。
	%IWS	输入通道状态字 (参见第 *482* 页)	包含有关模拟量输入通道的诊断信息。
	%QWS	输出通道状态字 (参见第 *484* 页)	包含有关模拟量输出通道的诊断信息。

对象类型	对象	对象功能	描述
I/O 对象	%I	输入位 (参见第 *224* 页)	存储数字量输入的值。
	%Q	输出位 (参见第 *225* 页)	存储数字量输出的值。
	%IW	模拟量输入字 (参见第 *226* 页)	存储模拟量输入的值。
	%QW	模拟量输出字 (参见第 *227* 页)	存储模拟量输出的值。
	%FC	快速计数器 (参见第 *230* 页)	在单字或双字计算模式下，用作加计数器或减计数器并对离散量输入的上升沿进行计数。
	%HSC	高速计数器 (参见第 *236* 页)	在单字或双字计算模式下，对离散量输入进行计数。
	%PLS	脉冲 (参见第 *255* 页)	在专用输出通道上生成方波脉冲信号。
	%PWM	脉冲宽度调制 (参见第 *291* 页)	在专用输出通道上生成具有可变占空比的调制波信号。
	%PTO	脉冲串输出 (参见第 *307* 页)	在开环模式下，生成脉冲串输出以控制线性单轴步进驱动器或伺服驱动器。
	%FREQGEN	频率发生器 (参见第 *409* 页)	以可编程的频率和 50% 的占空比在专用输出通道上生成方波信号。
网络对象	%QWM	输入寄存器 (Modbus TCP) (参见第 *298* 页)	由逻辑控制器发送的 Modbus 映射表 Input registers 的值。
	%IWM	输出寄存器 (Modbus TCP) (参见第 *299* 页)	由逻辑控制器接收的 Modbus 映射表 Output registers 的值。
	%IN	数字量输入 (IOScanner) (参见第 *300* 页)	Modbus Serial 或 TCP IOScanner 数字量输入位的值。
	%QN	数字量输出 (IOScanner) (参见第 *301* 页)	Modbus Serial 或 TCP IOScanner 数字量输出位的值。
	%IWN	输入寄存器 (IOScanner) (参见第 *303* 页)	Modbus Serial 或 TCP IOScanner 数字量输入字的值。
	%QWN	输出寄存器 (IOScanner) (参见第 *304* 页)	Modbus Serial 或 TCP IOScanner 数字量输出字的值。
	%IWNS	IOScanner 网络诊断代码 (参见第 *306* 页)	Modbus Serial 或 TCP IOScanner 网络诊断位的值。

对象类型	对象	对象功能	描述
软件对象	%TM	定时器 *(参见 SoMachine Basic, 通用功能库指南)*	指定触发动作之前的时间。
	%C	计数器 *(参见 SoMachine Basic, 通用功能库指南)*	提供动作的加减计数。
	%MSG	消息 *(参见 SoMachine Basic, 通用功能库指南)*	存储通讯端口的状态消息。
	%R	LIFO/FIFO 寄存器 *(参见 SoMachine Basic, 通用功能库指南)*	存储以 2 种不同的方式（队列和堆栈）各存储最多 16 个字的 16 位的存储器。
	%DR	磁鼓 *(参见 SoMachine Basic, 通用功能库指南)*	工作原理与根据外部事件更改步进的机电鼓控制器类似。
	%SBR	移位寄存器 *(参见 SoMachine Basic, 通用功能库指南)*	提供二进制数据位（0 或 1）的左移或右移。
	%SC	步进计数器	提供可向其分配动作的一系列步进。
	SCH	计划块 *(参见 SoMachine Basic, 通用功能库指南)*	按照预定义的月、日和时间控制动作。
	%RTC	RTC	允许在逻辑控制器上读取或写入实时时钟 (RTC) 的值。
	PID	PID (参见第 *413* 页)	提供一般控制回路反馈，其中输出是输入的比例、积分和微分。
	%X	Grafcet 步骤	与各 Grafcet (SFC) 步骤关联的位对象。当相应步骤处于活动状态时，将对象设置为 1；而当相应步骤处于禁止状态时，将对象设置为 0。
PTO 对象	请参阅脉冲串输出 (参见第 *307* 页)。		
驱动器对象	请参阅驱动器对象 (参见第 *262* 页)。		
通讯对象	%READ_VAR	读取变量 *(参见 SoMachine Basic, 通用功能库指南)*	%READ_VAR 功能块用于从 Modbus SL 或 Modbus TCP 上的远程设备读取数据
	%WRITE_VAR	写入变量 *(参见 SoMachine Basic, 通用功能库指南)*	%WRITE_VAR 功能块用于按照 Modbus SL 或 Modbus TCP 协议向外部设备写入数据
	%WRITE_READ_VAR	写读变量 *(参见 SoMachine Basic, 通用功能库指南)*	%WRITE_READ_VAR 功能块用于按照 Modbus SL 或 Modbus TCP 协议向外部设备读写存储在内部存储器字中的数据。
	%SEND_RECV_MSG	发送接收消息 *(参见 SoMachine Basic, 通用功能库指南)*	%SEND_RECV_MSG 功能块用于在针对 ASCII 协议配置的串行线路上发送或接收数据。

对象类型	对象	对象功能	描述
用户定义功能和用户定义功能块对象	%RET0	返回值	用户定义功能的返回值。
	%PARAM	参数	用户定义功能或用户定义功能块的参数。 每种对象类型的参数不同。
	%VAR	局部变量	用户定义功能或用户定义功能块的局部变量。 每种对象类型的局部变量不同。

存储器对象、软件对象和通讯对象是 SoMachine Basic 中使用的一般对象，而系统对象和 I/O 对象是控制器特定的对象。将在*编程* (参见第 *217* 页)一节中讨论所有的控制器特定的对象。

有关存储器对象、软件对象和通讯对象的编程详细信息，请参阅《SoMachine Basic 通用功能库指南 *(参见 SoMachine Basic, 通用功能库指南)*》。

有关 PID、驱动器和 PTO 对象的编码详情，请参阅本文档的编程部分 (参见第 *217* 页)。

有关用户定义功能和用户定义功能块的更多详情，请参阅 M100/M200 操作指南。

对象寻址

寻址示例

下表显示了各种对象类型的寻址示例：

对象类型	语法	示例	描述
存储器对象			
存储器位	%M*i*	%M25	内部存储器位 25。
字存储器	%MW*i*	%MW15	内部存储器字 15。
双字存储器	%MD*i*	%MD16	内部存储器双字 16。
存储器浮点数	%MF*i*	%MF17	内部存储器浮点数 17。
常量字	%KW*i*	%KW26	常量字 26。
双字常量	%KD*i*	%KD27	内部常量双字 27。
常量浮点数	%KF*i*	%KF28	内部常量浮点数 28。
系统对象			
系统位	%S*i*	%S8	系统位 8。
系统字	%SW*i*	%SW30	系统字 30。
I/O 对象			
数字量输入	%I*y.z*	%I0.5	控制器上的数字量输入 5（内置 I/O）。
数字量输出	%Q*y.z*	%Q3.4	地址 3 处扩展模块上的数字量输出 4（扩展模块 I/O）。
模拟量输入	%IW0.*y0z*	%IW0.101	扩展板 1 上的模拟量输入 1。
模拟量输出	%QW0.*m0n*	%QW0.202	扩展板 2 上的模拟量输出 2。
快速计数器	%FC*i*	%FC2	控制器上的快速计数器 2。
高速计数器	%HSC*i*	%HSC1	控制器上的高速计数器 1。
脉冲	%PLS*i*	%PLS0	控制器上的脉冲输出 0。
脉冲宽度调制	%PWM*i*	%PWM1	控制器上的脉冲宽度调制输出 1。
脉冲串输出	%PTO*i*	%PTO1	控制器上的脉冲串输出 1。
频率发生器	%FREQGEN*i*	%FREQGEN1	控制器上的频率发生器 1。
网络对象			

a 100 + SL1 上的设备数，200 + SL2 上的设备数，300 + ETH1 上的设备数。
b Modbus Serial IOScanner 或 Modbus TCP IOScanner 设备的通道数。
c 通道中的对象实例标识符。
i 对象实例标识符，表示控制器上对象的实例。有关每个对象的最大实例数，请参阅*对象的最大数量*(参见第 *32* 页)。如果 n 为对象的最大数量，则实例范围为 0...n-1。
m 控制器上的扩展板数量。
n 扩展板上的通道数量。
y 表示 I/O 类型。对于控制器，取值为 0，对于扩展模块，取值为 1、2 等。
z 控制器或扩展模块上的通道编号。

对象类型	语法	示例	描述
输入寄存器 (Modbus TCP)	%QWM*i*	%QWM1	Input register 实例 1。
输出寄存器 (Modbus TCP)	%IWM*i*	%IWM0	Output register 实例 0。
数字量输入 (IOScanner)	%IN*a. b. c*	%IN300. 2. 1	ETH1 上的 Modbus TCP IOScanner 从站设备 0，通道 2，数字量输入 1。
数字量输出 (IOScanner)	%QN*a. b. c*	%QN101. 1. 0	SL1 上的 Modbus Serial IOScanner 从站设备 1，通道 1，数字量输出 0。
输入寄存器 (IOScanner)	%IWN*a. b. c*	%IWN302. 3. 0	ETH1 上的 Modbus TCP IOScanner 从站设备 2，通道 3，输入寄存器 0。
输出寄存器 (IOScanner)	%QWN*a. b. c*	%QWN205. 0. 4	SL2 上的 Modbus Serial IOScanner 从站设备 5，通道 0，输出寄存器 4。
IOScanner 网络诊断代码	%IWNS*a*	%IWNS302	ETH1 上 Modbus TCP IOScanner 从站设备 2 的状态。
	%IWNS*a.b*	%IWNS205. 3	串行线路 SL2 上 Modbus Serial IOScanner 从站设备 5 的通道 3 的状态
软件对象			
定时器	%TM*i*	%TM5	定时器实例 5。
计数器	%C*i*	%C2	计数器实例 2。
消息	%MSG*i*	%MSG1	程序编译状态消息 1。
LIFO/FIFO 寄存器	%R*i*	%R3	FIFO/LIFO 寄存器实例 3。
磁鼓	%DR*i*	%DR6	控制器上的磁鼓寄存器 6。
移位寄存器	%SBR*i*	%SBR5	控制器上的移位寄存器 5。
步进计数器	%SC*i*	%SC5	控制器上的步进计数器 5。
计划块	SCH *i*	SCH 3	控制器上的计划块 3。
RTC	RTC*i*	RTC 1	实时时钟 (RTC) 实例 1。
PID	PID *i*	PID 7	控制器上的 PID 反馈对象 7。
Grafcet 步骤	X*i*	X1	Grafcet 步骤 1。
PTO 对象			
MC_Power_PTO（运动功能块）	%MC_POWER_PTO*i*	%MC_POWER_PTO1	MC_POWER_PTO 功能块实例 1。 有关 PTO 功能块的详细信息，请参阅*脉冲串输出 (%PTO)* (参见第 *307* 页)。

a 100 + SL1 上的设备数，200 + SL2 上的设备数，300 + ETH1 上的设备数。
b Modbus Serial IOScanner 或 Modbus TCP IOScanner 设备的通道数。
c 通道中的对象实例标识符。
i 对象实例标识符，表示控制器上对象的实例。有关每个对象的最大实例数，请参阅*对象的最大数量* (参见第 *32* 页)。如果 n 为对象的最大数量，则实例范围为 0...n-1。
m 控制器上的扩展板数量。
n 扩展板上的通道数量。
y 表示 I/O 类型。对于控制器，取值为 0，对于扩展模块，取值为 1、2 等。
z 控制器或扩展模块上的通道编号。

对象类型	语法	示例	描述
MC_Reset_PTO （管理功能块）	%MC_RESET_PTO*i*	%MC_RESET_PTO0	MC_RESET_PTO 功能块实例 0。 有关 PTO 功能块的详细信息，请参阅*脉冲串输出 (%PTO)*(参见第 *307* 页)。
通讯对象			
读取变量	%READ_VAR*i*	%READ_VAR2	READ_VAR 功能块实例 2。
写入变量	%WRITE_VAR*i*	%WRITE_VAR4	WRITE_VAR 功能块实例 4。
读写变量	%WRITE_READ_VAR*i*	%WRITE_READ_VAR0	WRITE_READ_VAR 功能块实例 0。
发送接收消息	%SEND_RECV_MSG*i*	%SEND_RECV_MSG6	SEND_RECV_MSG 功能块实例 6。
用户定义功能和功能块对象			
返回值	%RET*i*	%RET0	用户定义功能的返回值。
参数	%PARAM*i*	%PARAM0	用户定义功能的参数。
局部变量	%VAR*i*	%VAR0	用户定义功能的局部变量。

a 100 + SL1 上的设备数，200 + SL2 上的设备数，300 + ETH1 上的设备数。
b Modbus Serial IOScanner 或 Modbus TCP IOScanner 设备的通道数。
c 通道中的对象实例标识符。
i 对象实例标识符，表示控制器上对象的实例。有关每个对象的最大实例数，请参阅*对象的最大数量* (参见第 *32* 页)。如果 n 为对象的最大数量，则实例范围为 0...n-1。
m 控制器上的扩展板数量。
n 扩展板上的通道数量。
y 表示 I/O 类型。对于控制器，取值为 0，对于扩展模块，取值为 1、2 等。
z 控制器或扩展模块上的通道编号。

最大对象数

描述

下表提供了有关 M100/M200 可编程控制器 支持的最大对象数的信息：

对象类型	对象	允许的最大数量
存储器对象	%M	1024
	%MW	● 4000（对于 TM100C••R）和 TM100C••RN ● 8000（对于 TM200C•••）
	%MD / %MF	● 3999（对于 TM100C••R 和 TM100C••RN) ● 7999（对于 TM200C•••）
	%KW	512
	%KD / %KF	511
系统对象	%S	160
	%SW	234
	%IWS	为每个模拟量输入自动创建 1
	%QWS	为每个模拟量输处自动创建 1
I/O 对象	%I	● 9（对于 TM100C16R / TM100C16RN）/ TM200C16• ● 14（对于 TM100C24R / TM100C24RN / TM200C•24•） ● 20（对于 TM100C32RN / TM200C•32R） ● 24（对于 TM100C40R / TM100C40RN / TM200C•40•） ● 36（对于 TM200C60R）
	%Q	● 7（对于 TM100C16R / TM100C16RN）/ TM200C16• ● 10（对于 TM100C24R / TM100C24RN / TM200C•24•） ● 12（对于 TM100C32RN / TM200C•32R） ● 16（对于 TM100C40R / TM100C40RN / TM200C•40•） ● 24（对于 TM200C60R）
	%IW	**注意：** M100/M200 可编程控制器 没有内置模拟量 I/O。使用扩展板或扩展模块将模拟量 I/O 添加到配置。
	%QW	
	%FC	4
	%HSC	4
	%PLS / %PWM / %PTO / %FREQGEN	● 0（对于 TM100C••R / TM100C••RN / TM200C•••） ● 2（对于 TM200C•••U）/ TM200C•••T

对象类型	对象	允许的最大数量
网络对象	%QWM	20
	%IWM	20
	%IN	128
	%QN	128
	%IWN	128
	%QWN	128
	%IWNS	1（对于每个已配置的 Modbus Serial IOScanner 或 Modbus TCP IOScanner 设备），加 1（对于每个通道）
	%QWNS	1（对于每个已配置的 Modbus Serial IOScanner 或 Modbus TCP IOScanner 设备），加 1（对于每个通道）
软件对象	%TM	255
	%C	255
	%MSG	1
	%R	4
	%DR	8
	%SBR	8
	%SC	8
	%SCH	16
	%RTC	2
	PID	14

对象类型	对象	允许的最大数量
PTO 对象	%MC_MOTIONTASK_PTO	2
	%MC_POWER_PTO	86
	%MC_MOVEVEL_PTO	86
	%MC_MOVEREL_PTO	86
	%MC_MOVEABS_PTO	86
	%MC_HOME_PTO	86
	%MC_READACTVEL_PTO	40
	%MC_READACTPOS_PTO	40
	%MC_READSTS_PTO	40
	%MC_READMOTIONSTATE_PTO	40
	%MC_READAXISERROR_PTO	40
	%MC_RESET_PTO	40
	%MC_TOUCHPROBE_PTO	40
	%MC_ABORTTRIGGER_PTO	40
	%MC_READPAR_PTO	40
	%MC_WRITEPAR_PTO	40
驱动器对象	%DRV	16
通讯对象	%READ_VAR	16
	%WRITE_VAR	16
	%WRITE_READ_VAR	16
	%SEND_RCV_MSG	16
用户定义功能和用户定义功能块对象	%RETO	每个用户定义功能 1 个
	%PARAM	每个用户定义功能和用户定义功能块 5 个
	%VAR	每个用户定义功能和用户定义功能块 10 个

第2.2节
任务结构

本节包含了哪些内容？

本节包含了以下主题：

主题	页
任务和扫描模式	36
最大任务数和优先级	38

任务和扫描模式

概述

SoMachine Basic 具有以下扫描模式：

- **正常模式**
  连续循环扫描模式（自由运行模式），前一个扫描完成后立即开始新的扫描。
- **周期模式**
  周期性循环扫描模式，仅在经过前一个扫描的配置的扫描时间之后开始新的扫描。因此，每次扫描的持续时间都相同。

SoMachine Basic 提供以下任务类型：

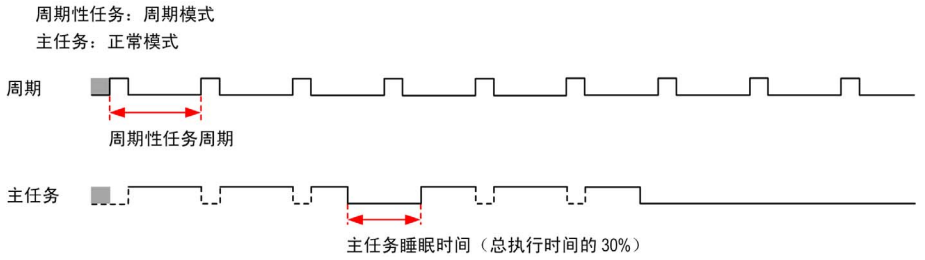
- **主任务**：应用程序的主任务。
  对于主任务，通过连续循环扫描（在正常扫描模式中）、软件定时器（在周期性扫描模式中），或者指定扫描周期 1...150 毫秒（缺省值 10 毫秒）来触发主任务。
- **周期性任务**：定期处理持续时间较短的子程序。
  对于周期性任务，通过在周期扫描模式下指定扫描周期 1...255 毫秒（缺省值 255 毫秒），借助因此而配置的软件定时器来触发。
- **事件任务**：持续时间很短的子程序，用于减少应用程序的响应时间。
  事件任务由物理输入或 HSC 功能块触发。这些事件与内置数字量输入 (%I0.2...%I0.5)（上升沿、下降沿或两者）或与高速计数器（%HSC0 和 %HSC1）（当计数达到高速计数器阈值时）相关联。您可以为每个 HSC 功能块配置 2 个事件。

周期性任务和事件在周期性扫描模式中配置。主任务在正常扫描模式或周期性扫描模式中配置。

有关详细信息，请参阅配置程序行为和任务 *(参见 SoMachine Basic, 操作指南)*。

正常扫描模式中的主任务

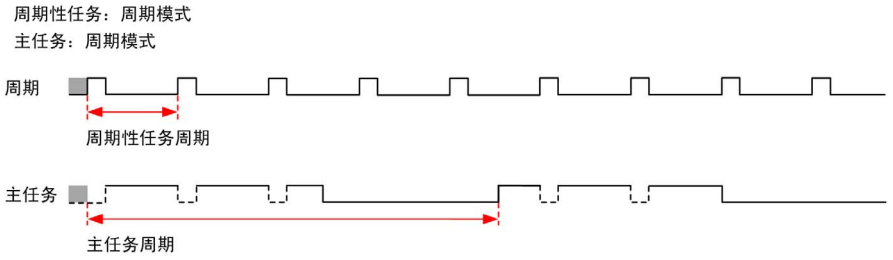
下图显示了在正常扫描模式中配置主任务时，主任务和周期性任务执行之间的关系：



周期性扫描模式中的主任务

在周期模式下，Logic Controller 一直等到配置的扫描时间过去后，才会开始一个新扫描。因此，每次扫描的持续时间都相同。

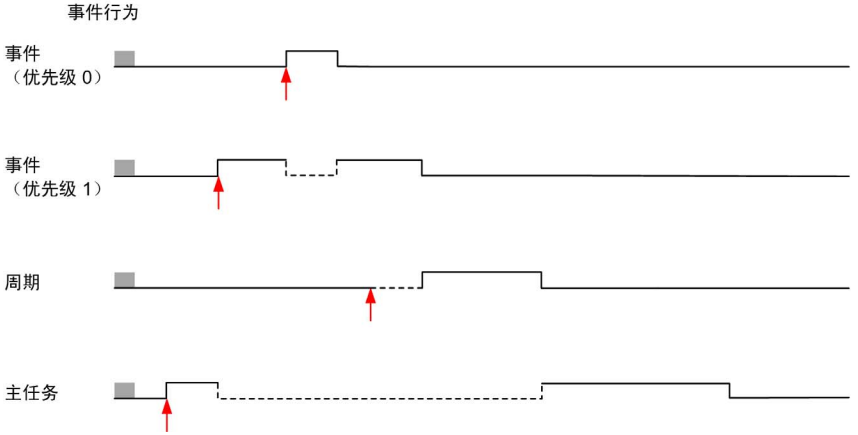
下图显示了在周期性扫描模式中配置主任务时，主任务和周期性任务之间的关系：



主任务与周期性任务的事件优先级

事件优先级控制事件任务、主任务和周期性任务之间的关系。事件任务将中断主任务和周期性任务的执行。

下图显示了在周期模式下，事件任务、主任务和周期性任务之间的关系：



事件任务由将任务事件发送到事件任务的硬件中断触发。

警戒时钟定时器

您可以为主任务和周期性任务配置特定的专用看门狗定时器。如果任务执行时间超过配置的警戒时钟定时器周期，则逻辑控制器会进入 HALTED 状态。

系统警戒时钟定时器将验证程序是否正在使用超过 80% 的处理能力。在这种情况下，逻辑控制器会进入 HALTED 状态。

最大任务数和优先级

任务优先级

下表概述了任务类型及其优先级：

任务类型	扫描模式	触发条件	可配置范围	最大任务数	优先级
主任务	正常	正常	不适用	1	最低
	周期性任务	软件定时器	1...150 ms		
周期	周期性任务	软件定时器	1...255 毫秒	1	高于主任务但低于事件任务
事件任务	周期	物理输入	%I0.2...%I0.5	4	最高
		%HSC 功能块	每个 %HSC 对象最多两个事件	4	

事件优先级

请参阅事件优先级和队列 *(参见 SoMachine Basic, 操作指南)*。

第2.3节
控制器状态和行为

简介

本节提供控制器状态、状态转换及响应系统事件时的行为的相关信息。本章开头详细介绍了控制器状态图并描述了各种状态。接着定义了输出状态与控制器状态的关系，然后解释导致状态转换的命令和事件。最后介绍了与保持型变量有关的信息以及 SoMachine Basic 任务编程选项对系统行为的影响。

本节包含了哪些内容？

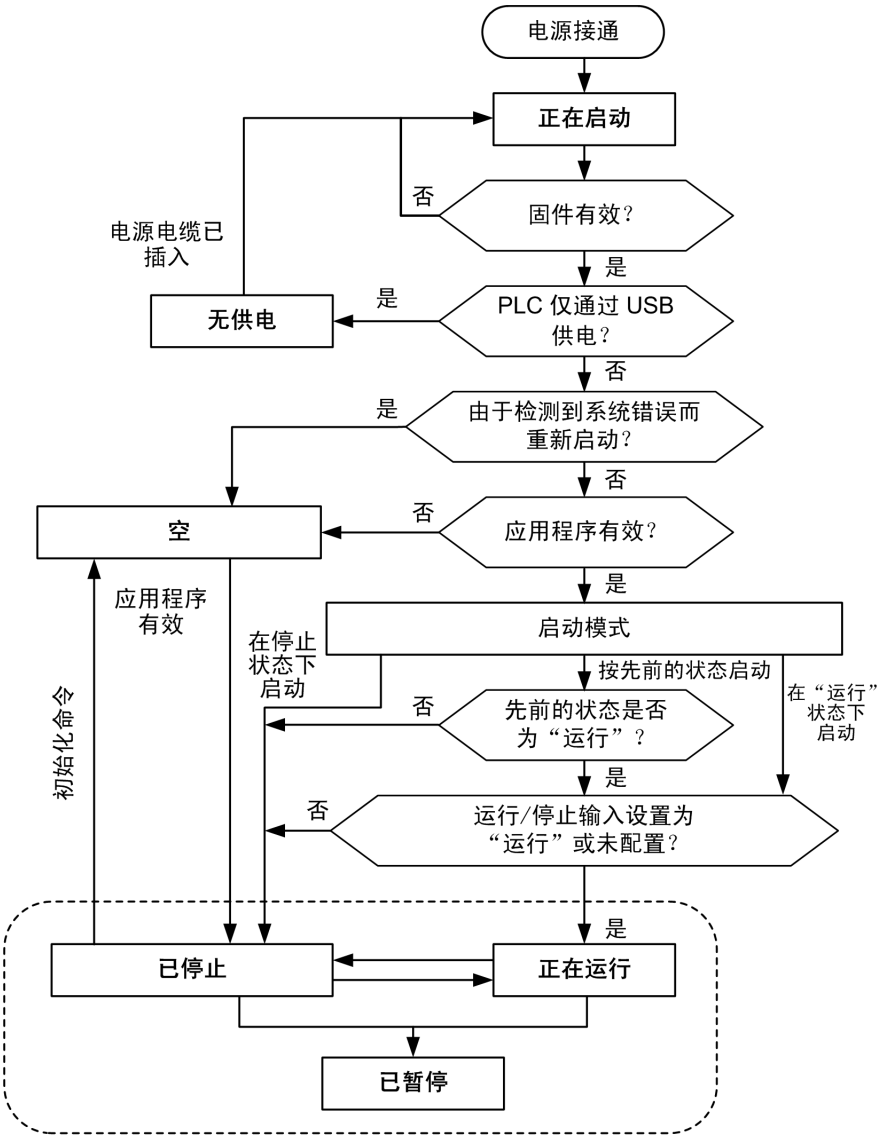
本节包含了以下主题：

主题	页
控制器状态示意图	40
控制器状态描述	41
控制器状态转换	44
持续性变量	47
输出行为	49

控制器状态示意图

控制器状态示意图

下图描述了控制器操作状态：



控制器状态描述

简介

本节详细介绍控制器状态。

⚠ 警告

意外的设备操作

- 在没有通过命令进行状态修改、配置控制器选项、上载程序或修改控制器及其所连接设备的物理配置之前，切勿认为控制器已经处于某种控制器状态下了。
- 在执行以上任何操作之前，请考虑对所有所连接设备的影响。
- 在对某个控制器进行操作之前，请务必主动确认控制器的状态，方法是查看其 LED、确认运行/停止输入的情况、检查是否存在输出强制，以及通过 SoMachine Basic 查看控制器状态信息。

不遵循上述说明可能导致人员伤亡或设备损坏。

使用“在‘运行’状态下启动”功能时，控制器将在设备通电时开始执行程序逻辑。预先了解输出的自动重新激活如何影响过程或所控制的机器，这一点至关重要。配置运行/停止输入可帮助控制“在‘运行’状态下启动”功能。此外，运行/停止输入旨在提供对远程运行命令的本地控制。如果在控制器由 SoMachine 本地停止后发出的远程运行命令可能具有意外后果，则必须配置和连接运行/停止输入，以帮助控制这种情况。

⚠ 警告

机器意外启动

- 在使用“在‘运行’状态下启动”功能之前，请确认自动重新激活输出不会产生意外的后果。
- 使用运行/停止输入可帮助控制“在‘运行’状态下启动”功能，并有助于防止从远程位置意外启动。
- 在对运行/停止输入加电之前或在从远程位置发出运行命令之前，请验证机器或过程环境的安全状态。

不遵循上述说明可能导致人员伤亡或设备损坏。

控制器状态表

下表详细介绍控制器操作状态：

控制器状态	描述	通讯	应用程序的执行	LED		
				PWR	RUN	ERR
BOOTING	逻辑控制器无任何有效固件。 启用通信通道以允许更新运行时固件。 无法通过 SoMachine Basic 进行登录。 输出设置为初始化值 (参见第 *49* 页)。	受限制	否	亮起	熄灭	亮起
EMPTY	此状态指示没有有效的应用程序。 无法通过 SoMachine Basic （*download/animation table*） 进行登录。 输入强制设置为 0。 输出设置为初始化值 (参见第 *49* 页)。	有	否	亮起	熄灭	闪烁 1 次
STOPPED	此状态指示逻辑控制器具有有效的应用程序，但已停止。 读取输入。 输出设置为 SoMachine Basic 中的故障预置值 (参见第 *50* 页)或强制值 (参见第 *51* 页)。 状态警报输出设置为 0。	有	否	亮起	闪烁	熄灭
RUNNING	此状态指示逻辑控制器正在执行应用程序。 应用程序任务读取输入。 应用程序任务所编写的输出，或者通过 SoMachine Basic 编写（在在线模式下） （动态数据表、强制输出 (参见第 *51* 页)）。 状态警报输出设置为 1。	是	是	亮起	亮起	熄灭
HALTED	此状态表示，由于检测到 应用程序或系统看门狗超时错误 (参见第 *50* 页)，因此应用程序已经停止。 对象将保留其值，从而用于分析检测到错误的原因。任务停止在最后一条指令上。 通讯功能同样处于 STOPPED 状态。 不读取输入，并且保持其最后值。 输出设置为故障预置值 (参见第 *50* 页)。 状态警报输出设置为 0。	有	否	亮起	闪烁	亮起

控制器状态	描述	通讯	应用程序的执行	LED		
				PWR	RUN	ERR
POWERLESS	此状态指示逻辑控制器仅由 USB 电缆供电。 此模式可以用于更新固件（通过 USB）或下载/上载用户应用程序（通过 USB）。 若要更改逻辑控制器的状态，应连接主电源使逻辑控制器启动并重启加载安装的组件。 无法通过 SoMachine Basic（*download/upload/animation table*）进行登录。 输入强制设置为 0。 输出设置为初始化值 (参见第 *49* 页)。	是 （仅 USB）	否	熄灭	闪烁	熄灭

注意： 系统字 %SW6 指示逻辑控制器状态（EMPTY、STOPPED、RUNNING、HALTED 和 POWERLESS）。

控制器状态转换

启动控制器

效果：命令控制器重新启动。关于通电顺序的详情，请参阅控制器状态示意图 (参见第 *40* 页)。

方法：

- 电源重置
- 利用脚本重新启动
  - micro SD 卡上的脚本可以发出 REBOOT 作为其最终命令。

应用程序下载

效果： 将应用程序下载到逻辑控制器存储器。

方法：

- SoMachine Basic 在线按钮：
  - 选择 **PC 到控制器（下载）**命令。
    效果： 消除逻辑控制器中的应用程序，并将逻辑控制器设置为 EMPTY 状态。将应用程序下载到逻辑控制器存储器。如果下载成功，则冷启动完成，并且逻辑控制器被设置为 STOPPED 状态。
- SD 卡进行的应用程序文件传输：
  - 效果：在下一次重新启动时，消除逻辑控制器中的应用程序，并将应用文件从 micro SD 卡下载至逻辑控制器存储器中。如果下载成功，则冷启动完成，并且逻辑控制器被设置为 STOPPED 状态。

初始化控制器

效果： 将控制器设置为 EMPTY 状态，接着在冷启动后设置为 STOPPED 状态。

方法：

- SoMachine Basic 在线按钮：
  - 选择**初始化控制器**命令。

运行控制器

效果： 命令转换为 RUNNING 状态。

方法：

- 前面板的运行/停止 *(参见 Modicon M100/M200 可编程控制器, 硬件指南)*开关：
  - 其命令在上升沿转换为 RUNNING 状态。
- 运行/停止输入: *(参见 Modicon M100/M200 可编程控制器, 硬件指南)*
  - 输入必须在应用程序中配置（配置数字量输入 (参见第 *73* 页)）。
  - 其命令在上升沿转换为 RUNNING 状态。
- SoMachine Basic 在线按钮：
  - 选择**运行控制器**命令。
- 应用程序启动模式 *(参见 SoMachine Basic, 操作指南)*设置：
  - **在“运行”状态下启动**或**在先前状态下启动**。

停止控制器

效果： 命令转换为 STOPPED 状态。

方法：

- 前面板的运行/停止 *(参见 Modicon M100/M200 可编程控制器, 硬件指南)*开关：
  - 其强制在低电平上转换为 STOPPED 状态。
- 运行/停止输入: *(参见 Modicon M100/M200 可编程控制器, 硬件指南)*
  - 输入必须在应用程序中配置（配置数字量输入 (参见第 *73* 页)）。
  - 其强制在低电平上转换为 STOPPED 状态。
- SoMachine Basic 在线按钮：
  - 选择**停止控制器**命令。
- 应用程序启动模式 *(参见 SoMachine Basic, 操作指南)*设置：
  - **上电停止**或**保持掉电前状态**。
- **下载**命令：
  - 需要控制器设置为 STOPPED 状态（在下载后，控制器处于 STOPPED 状态）。

检测到的错误（转换到 HALTED 状态）

效果： 命令转换为 HALTED 状态。

切换到 HALTED 状态的原因：

- 应用程序警戒时钟超时（由用户配置）
- 系统警戒时钟超时（系统溢出，超过 80% 的 CPU 处理能力已被使用)

冷启动

冷启动定义为：在所有数据初始化为其缺省值的情况下通电，并且程序从开头启动，同时所有程序变量都被清除。软件和硬件设置被初始化。

会由于下列原因而进行冷启动：

- 启动控制器，且不含经过验证的应用程序在线修改。
- 下载应用程序
- 初始化逻辑控制器

冷启动的效果：

- 初始化功能块。
- 清除用户存储器。
- 将系统位 %S 和系统字 %SW 置于它们的初始值。
- 配置后重新加载参数（应用配置后的更改）。
- 从闪存中恢复应用程序（未保存的在线更改将丢失）。
- 重新启动控制器的内部部件。

持续性变量

断电时自动保存

TM100C••R 和 TM200C••• 控制器在发生下列任何一种断电状况之后，会自动将最前面的 3000 个存储字（%MW0 至 %MW2999）保存在内部闪存中。

TM100C••RN 控制器在发生下列任何一种断电状况之后，会自动将最前面的 2000 个存储字（%MW0 至 %MW1999）保存在内部闪存中。

在初始化期间，即使控制器由于电池缺失或电量耗尽而执行了冷启动，数据也会被恢复至存储字区域。

在新下载、INIT 命令或 %S0 激活的情况下，将重新初始化这些 *自动保存*的持续性变量。请参阅系统位 (参见第 *454* 页)。

通过用户请求保存

您可以将存储器字保存在非易失性存储器或 SD 卡中。要执行保存操作：

1. 选择带 %S90 的目标（请参阅系统位 (参见第 *454* 页)）：
   - 设置为 0：非易失性存储器（缺省）
   - 设置为 1：SD 卡
2. 设置要在系统字 %SW148（请参阅系统字 (参见第 *463* 页)）中保存的存储字数量。
3. 将系统位 %S93 设置为 1（请参阅系统位 (参见第 *454* 页)）。

保存操作完成后：

- 系统位 %S93 复位为 0。
- 系统位 %S92 设置为 1，表示存储器字已成功保存在非易失性存储器中（%S90 设置为 0）。
- 系统字 %SW147 指示 SD 卡操作结果（%S90 设置为 1）。

根据用户请求恢复

可以恢复之前保存的存储器字。要执行恢复操作：

1. 将系统位 %S92 设置为 1。
   如果 %S92 为 0（之前未保存任何值），则非易失性存储器的操作没有效果。
2. 选择带 %S90 的源（请参阅系统位 (参见第 *454* 页)）：
   - 设置为 0：非易失性存储器（缺省）
   - 设置为 1：SD 卡
3. 如要通过非易失性存储器恢复，则设置系统字 %SW148 中存储器字的数量（请参阅系统字 (参见第 *463* 页)）。在通过 SD 卡恢复时，会处理整个 Memory Variables.csv 文件。
4. 将系统位 %S94 设置为 1（请参阅系统位 (参见第 *454* 页)）。

注意：

恢复操作完成后：

- 系统位 %S94 被系统复位为 0。
- 使用已恢复的对象数更新系统字 %SW148（比如，如果指定了 100 个要恢复的字，并且之前仅保存了 50 个，那么 %SW148 的值将为 50）。
- 系统字 %SW147 指示 SD 卡操作结果（%S90 设置为 1）。

根据用户请求删除

可以删除之前保存在非易失性存储器上的存储器字。

要执行删除操作：

1. 将系统位 %S91 设置为 1（请参阅系统位 (参见第 *454* 页)）。
2. 删除操作完成时，系统位 %S91 和 %S92 以及系统字 %SW148 由逻辑控制器重置为 0。

注意： 不可能仅删除选择的变量：已保存变量的整个组被删除（这意味着 %SW148 对擦除操作没有任何影响，擦除操作的实施与 %SW148 的值无关）。

输出行为

简介

控制器将输出行为定义为通过更加灵活的方式响应命令和系统事件。在讨论影响控制器状态的命令和事件之前，有必要先了解此行为。

可能应用这两个选项的输出行为和控制器状态包括：

- 由应用程序管理
- 初始值
- 故障预置行为 *(参见 SoMachine Basic, 操作指南)*
  - 保持当前值
  - 故障预置值
- 输出强制

由应用程序管理

正常情况下，您的应用程序会管理输出。这适用于 RUNNING 状态。

硬件初始化值

此输出状态适用于 BOOTING、EMPTY 和 POWERLESS 状态。

在初始化状态下，输出采用下列值：

- 对于内置输出：
  - 快速源晶体管输出：0 Vdc
  - 快速漏极晶体管输出：0 Vdc
  - 继电器输出：打开
- 对于扩展模块输出：
  - 常规源晶体管输出：0 Vdc
  - 常规漏极晶体管输出：24 Vdc
  - 继电器输出：打开

软件初始化值

此输出状态当下载时或复位应用程序时应用。它在下载结束或者冷复位结束时应用。

输入对象（%I 和 %IW）、网络对象 (%QWM) 以及 Modbus Serial IOScanner 输入对象（%IN 和 %IWN）设置为 0。输出对象（%Q 和 %QW）、网络对象 (%IWM) 以及 Modbus Serial IOScanner 输出对象（%QN 和 %QWN）根据所选择的故障预置行为设置。

故障预置管理

故障预置行为的目的是在控制器退出 RUNNING 状态时控制输出。

除下面所述的特殊情况外，当发生从 RUNNING 到 STOPPED 或 HALTED 状态的过渡时将应用故障预置值。

故障预置行为配置

故障预置行为在**任务→行为**窗口的**编程**选项卡上配置：

- 当选择了**故障预置值**时，如果发生故障预置，输出值将采用在**故障预置值**中配置的值。
- 当选择了**保持当前值**时，如果发生故障预置，输出会保持其值，但在脉冲发生器 (PWM、PLS、PTO、FREQGEN) 或反射功能中配置的输出除外。

故障预置执行

发生故障预置时：

- 如果选择了**故障预置值**，输出将采用在**故障预置值**中配置的值。
- 如果**维持值**被选中，输出将保持其值。

特殊情况：

- 警报输出、PTO 和 FREQGEN：从不应用故障预置。其故障预置值强制设置为 0。
- PLS、PWM 和反射输出：
  - 如果选择**故障预置值**，输出将采用在**故障预置值**中配置的值。
  - 如果选择**保持当前值**，输出将设置为 0。

注意：

- 在下载完成之后，输出将设置为故障预置值。
- 在 EMPTY 状态下，输出将设置为 0。
- 因为数据映像反映物理值，所以故障预置值也适用于数据映像。然而，使用系统位 %S9 应用故障预置值，不会修改数据映像的值。

故障预置值

此输出状态适用于 STOPPED 和 HALTED 状态。

在故障预置期间，输出采用下列值：

- 对于内置输出：
  - 快速晶体管输出：根据故障预置设置
  - 常规晶体管输出：根据故障预置设置
  - 继电器输出：根据故障预置设置
  - 专业 I/O 功能（HSC、PLS、PWM、PTO 和 FREQGEN）：
    - 源极输出：0 Vdc
    - 漏极输出：0 Vdc
- 对于扩展模块输出：
  - 常规晶体管输出：根据故障预置设置
  - 继电器输出：根据故障预置设置

注意： 故障预置值应用的例外情况为 I/O 扩展总线错误情况。有关更多信息，请参见嵌入式输入/输出配置 (参见第 *72* 页)章节。

输出强制

控制器允许您将所选输出的状态强制为定义值，以便于系统测试、试运行和维护。

当控制器连接到 SoMachine Basic 时，可以强制设置输出的值。

要如此操作，请使用动态数据表中的**强制**命令或使用梯形图编辑器中的 F0 或 F1 按钮强制值。

输出强制将覆盖对输出执行的所有其他命令（无论正在执行何种任务逻辑）。

任何在线更改或者注销 SoMachine Basic 都不会解除强制。

冷启动 (参见第 *46* 页)和下载应用程序 (参见第 *44* 页)命令可自动解除强制。

强制操作不适用于专用 I/O 功能（HSC、PLS、PWM、PTO 和 FREQGEN）。

⚠ 警告

意外的设备操作

- 您必须全面了解强制会对与执行中的任务相关的输出产生怎样的影响。
- 请勿尝试强制包含在您不确定是否会及时执行的任务中的 I/O，除非您打算让强制在下次执行该任务时生效（无论何时）。
- 如果您强制某个输出，但是对物理输出没有产生明显效果，请勿在没有撤销强制的情况下退出 SoMachine Basic。

不遵循上述说明可能导致人员伤亡或设备损坏。

输出重置

如果出现短路或电流过载，则公共输出组会自动进入热保护模式（该组中的所有输出都设置为 0），随后会定期重置（每秒）以测试连接状态。但是，您必须了解这种重置对所控制的机器或过程的影响。

注意： 输出重置不应用到漏极输出。

⚠ 警告

机器意外启动

如果不想对机器或过程执行输出的自动重置，请禁用此功能。

不遵循上述说明可能导致人员伤亡或设备损坏。

注意： 只检测到输出设置 TRUE 和 0 之间的短路。没有检测到输出设置 FALSE 和 24 V 之间的短路。如有必要，可以使用系统位和字检测是发生了短路还是过载，以及出现问题的输出群集。系统位 %S10 可用于在程序中检测到发生了输出错误。随后，您可以使用系统字 %SW139 以编程方式确定发生短路或过载的输出群集。
可以禁用自动重置功能，只需将系统位 %S49 设置为 0（默认情况下，%S49 设置为 0）。

第2.4节
后配置

简介

本节将介绍如何管理和配置 Modicon M100/M200 可编程控制器 的后配置文件。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
后配置	53
后配置文件管理	54

后配置

简介

使用后配置选项可以在不更改应用程序的情况下修改应用程序的某些参数。后配置参数在一个存储在控制器中的名为 **Machine.cfg** 的文件中定义。

默认情况下，所有通讯参数都将在应用程序的配置过程中进行设置。但是，在某些情况下，其中的部分或全部参数可以通过后配置机制进行自动修改。可以在后配置文件中指定一个或多个通讯参数，这些参数可以覆盖配置操作所指定的参数。例如，可以在后配置文件中存储一个参数，以便更改控制器的 EtherNet/IP 地址并使其他以太网参数保持不变（如网关地址）。

参数

后配置文件用于修改网络参数。

以太网参数：
- 地址配置模式
- IP 地址
- 子网掩码
- 网关地址
- 设备名称

串行线路参数，适用于应用程序中的各个串行线路（内置端口或TMCR2SL1/TMCR2SL1A 扩展板）：
- 物理介质
- 波特率
- 奇偶校验
- 数据位
- 停止位
- Modbus 地址
- 极化（适用于 RS-485）

操作模式

在以下情况中，将读取并应用后配置文件：
- 冷启动 (参见第 *46* 页)之后
- 在重新启动 (参见第 *44* 页)后
- 在下载应用程序 (参见第 *44* 页)后
- 在以太网线重新连接导致以太网重新配置后（仅限于后配置文件的以太网部分 (参见第 *44* 页)）

有关控制器状态和转换的进一步详细信息，请参阅控制器状态和行为 (参见第 *39* 页)。

后配置文件管理

简介

后配置文件可以通过 SD 卡来进行存储、传输、修改或删除。

注意： SoMachine Basic 的安装目录 Firmwares & PostConfiguration\PostConfiguration\add_change\usr\cfg 中有提供后配置文件示例。

后配置文件格式

有效配置必须使用以下格式：

- 字符“#”代表注释开始，从该符号开始至行末的所有内容都将被忽略。注释不会保存在 M100/M200 可编程控制器 的后配置区域中。
- 规则为 channel.parameter=value（“=”符号两侧没有空格）。
- Channel 和 parameter 区分大小写。
- 下表含有允许的通道、参数和值。

通道	参数	描述	值
ETH	IPMODE	地址配置模式	0 = 固定 1 = BOOTP 2 = DHCP
	IP	IP 地址	以点分隔的十进制字符串
	MASK	子网掩码	以点分隔的十进制字符串
	GATEWAY	网关地址	以点分隔的十进制字符串
	NETWORKNAME	网络上的设备名称	ASCII 字符串（最多 16 个字符）
SL1 SL2	HW	物理介质	0 = RS-232 1 = RS-485
	BAUDS	数据传输速率	1200、2400、4800、9600、19200、38400、57600 或 115200
	PARITY	用于错误检测的奇偶校验	0 = 无 1 = 奇 2 = 偶
	DATAFORMAT	数据格式	7 或 8
	STOPBIT	停止位	1 或 2
	MODBUSADDR	Modbus 地址	1...247
	POLARIZATION	极化	0 = 否 1 = 是

注意：
当对以太网配置使用后配置文件时，不强制要求指定所有参数：
- 如果控制器（被用户应用程序）配置为 DHCP 或 BOOTP 模式，则文件中不会配置网络参数 IP（IP 地址）、MASK（子网掩码）和 GATEWAY（网关地址）。
- 如果后配置文件中没有配置某个参数，控制器将使用用户应用程序 (参见第 *146* 页)中的配置值。
- 如果用户应用程序没有将控制器配置为 DHCP 或 BOOTP 模式，且如果后配置文件中配置的是固定 IP 模式 (IPMODE=0)，则应配置所有网络参数（IP（IP 地址）、MASK（子网掩码）和 GATEWAY（网关地址）），因为用户应用程序没有配置它们。否则控制器会使用默认的以太网配置来启动。

后配置文件的传输

在创建和修改过后配置文件之后，必须将其传输到逻辑控制器。可通过脚本进行传输以将后配置文件复制到微型 SD 卡。

请参阅添加或更改后配置 (参见第 *197* 页)。

修改后配置文件

请使用文本编辑器来修改 PC 上的后配置文件。

注意： 请勿更改文本文件编码。缺省编码为 ANSI。

注意： 后配置文件的以太网参数可以通过 SoMachine Basic 来修改。有关详细信息，请参阅连接到逻辑控制器 *(参见 SoMachine Basic, 操作指南)*。

删除后配置文件

请参阅删除后配置文件 (参见第 *200* 页)。

注意： 将使用应用程序中所定义的参数，而非后配置文件中所定义的相应参数。

第II部分
配置 M100/M200 可编程控制器

概述

本部分提供有关如何配置 M100/M200 可编程控制器 型号的信息。

本部分包含了哪些内容？

本部分包括以下各章：

章	章节标题	页
3	如何配置控制器	59
4	内置输入/输出配置	72
5	I/O 总线配置	101
6	扩展板配置	107
7	TM3R扩展模块配置	135
8	内置通讯配置	145
9	Micro SD 卡	181

第3章
如何配置控制器

概述

本章介绍如何在 SoMachine Basic 中建立配置以及如何配置 M100/M200 可编程控制器。

本章包含了哪些内容？

本章包含了以下主题：

主题	页
建立配置	60
可选 I/O 扩展模块	65
配置 M100/M200 可编程控制器	69
使用 Executive Loader Wizard 更新固件	70

建立配置

简介

通过在 SoMachine Basic 中建立配置来配置控制器。要构建配置，请首先创建一个新项目或打开现有项目。

请参阅《*SoMachine Basic 操作指南*》，以了解如何执行以下操作：

- 创建新项目或打开现有项目
- 替换缺省逻辑控制器
- 将扩展模块添加到逻辑控制器
- 向逻辑控制器中添加扩展板
- 保存项目。

以下提供了有关 SoMachine Basic 用户界面的一些常规信息。

起始页

当您启动 SoMachine Basic 时，始终会显示起始页窗口。使用此窗口可注册 SoMachine Basic 软件，管理与逻辑控制器的连接，并且可以创建或选择要处理的项目。

SoMachine Basic 窗口

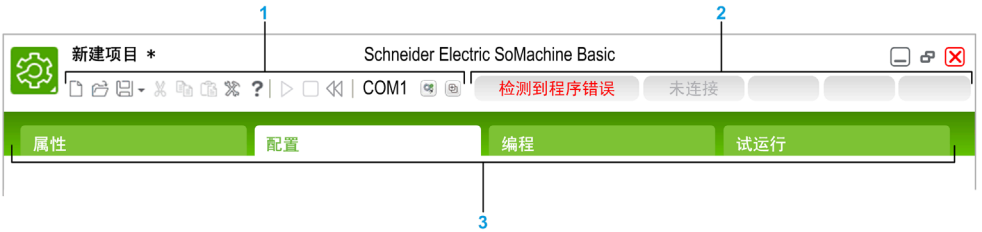
选择要处理的项目后，SoMachine Basic 将显示主窗口。

在主窗口顶部，工具栏 *(参见 SoMachine Basic, 操作指南)*包含一些图标，用于执行常见任务，包括返回起始页窗口。

在工具栏旁边是状态栏 *(参见 SoMachine Basic, 操作指南)*，它显示了关于逻辑控制器相连连接状态的参考性消息。

在工具栏和状态栏下方，主窗口划分为多个*模块*。每个模块控制开发周期的不同阶段，并且可通过单击模块选项卡来访问。

下图显示了主窗口中的工具栏、状态栏和模块选项卡：



1 工具栏
2 状态栏
3 选项卡

项	描述
工具栏	供轻松地访问常用功能。 有关详细信息，请参阅工具栏 *(参见 SoMachine Basic, 操作指南)*。
状态栏	显示有关系统状态的状态和参考性消息。 有关详细信息，请参阅状态栏 *(参见 SoMachine Basic, 操作指南)*。
选项卡	要开发应用程序，请使用如下从左到右显示的选项卡： ● **属性** 设置项目属性。 ● **配置** 复制和配置逻辑控制器和关联扩展模块的硬件配置。 ● **编程** 使用其中一种支持的编程语言开发程序。 ● **试运行** 管理 SoMachine Basic 与逻辑控制器之间的连接、上载/下载应用程序、测试和试运行应用程序。

硬件树

硬件树显示在**配置**窗口中的左侧。它显示了硬件配置的结构化视图。添加控制器时，扩展模块或者项目的扩展板以及若干节点会自动添加到硬件树。

注意： 硬件树中的节点特定于控制器和硬件配置。这些节点取决于控制器、扩展模块和扩展板提供的 I/O 功能。

下图显示了控制器配置的硬件树：

项	描述
数字量输入	用于配置逻辑控制器的内置数字量输入。
数字量输出	用于配置逻辑控制器的内置数字量输出。
高速计数器	用于配置内置高速计数功能 (HSC)。
脉冲发生器	用于配置内置脉冲发生器功能 (PLS/PWM/PTO/FREQGEN)。
IO 总线	用于配置连接到逻辑控制器的扩展模块和扩展板。
n　串行线路数（1 或 2，特定于控制器）。	

项	描述
ETH1	用于配置内置以太网通讯。
Modbus TCP	用于配置以太网通讯的 Modbus TCP 协议。
SLn（串行线路）	用来配置内置串行线路或使用扩展板添加的串行线路。 **注意：** 所有 M100/M200 型号只能支持一个串行线路扩展板。
n　串行线路数（1 或 2，特定于控制器）。	

编辑器

编辑器区域显示在**配置**窗口的中心。以图形方式显示设备的硬件配置。项目中的硬件配置可以是：

- 仅控制器
- 带有扩展板的控制器(s)
- 带有扩展模块的控制器
- 带有扩展板和扩展模块的控制器

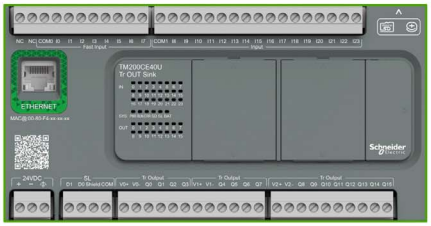
编辑器区域显示：

- 当您单击设备图像或单击硬件树中的设备节点时，显示有关设备的简短描述。
- 硬件树中选定项目的配置属性。

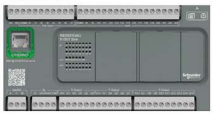
如果将扩展模块添加到配置中，该扩展模块将显示在控制器或以前添加的扩展模块的右侧。在控制器上，扩展板添加到扩展板插槽中。

在配置控制器、扩展板或扩展模块时，在硬件树中所选节点的配置属性将显示在图形配置下方。这些属性用于配置设备。

下图显示了具有扩展模块的控制器的配置（已选择控制器）：



设备信息



消息

设备描述

TM200CE40U 24 通道漏极/源极输入，带 2 个公共端，16 通道 0.5 A 晶体管漏极输出，带 3 个公共端，1 个串行线路端口，1 个以太网端口，带有可插拔端子块的 24 Vdc 一体型数字控制器。

目录

目录区域显示在**配置**窗口的右侧。显示可以使用 SoMachine Basic 配置的所有逻辑控制器、扩展模块和扩展板。它也提供所选设备的简短描述。

您可以将这些对象从目录区域拖放到编辑器区域。您也可以从目录中通过简单的拖放操作将现有控制器替换为不同控制器。

下图显示了逻辑控制器和扩展模块的目录。

可选 I/O 扩展模块

简介

I/O 扩展可以在配置中被标记为可选项目。 通过接受并非以物理方式连接至逻辑控制器上的模块的定义，**可选模块** 功能可提供更加灵活的配置。因此，单个应用程序可以支持 I/O 模块的多个物理配置，实现更高程度的可扩展性，而无需维护同一个应用程序的多个应用程序文件。

如果没有 **可选模块** 功能，当逻辑控制器启动 I/O 扩展总线（在供电循环、应用程序下载或初始化命令之后），它会将应用程序中所定义的配置与连接至 I/O 总线上的物理 I/O 模块进行比较。至于作出的其他诊断，如果逻辑控制器确定配置中定义的 I/O 模块并未实际上出现在 I/O 总线上，则会检出错误，且 I/O 总线不会启动。

如果有 **可选模块** 功能，则逻辑控制器会忽略您已标记为可选模块但并不存在的 I/O 扩展模块，后者随后会允许逻辑控制器启动 I/O 扩展总线。

即使可选模块在物理上并未与逻辑模块相连，逻辑控制器也会在配置期间启动 I/O 扩展总线（在供电循环、应用程序下载或初始化命令之后）。

下列类型的模块可标记为可选模块：

- TM3R I/O 扩展模块
- TM3 I/O 扩展模块
- TM2 I/O 扩展模块

对于被标记为逻辑控制器等认可的可选模块的模块，应用程序必须配置有至少达到**等级 3.2** 的 功能等级。

当运行机器或工艺时，必须充分认识到在 I/O 模块不存在和存在的情况下在应用程序中将它们标示为可选的后果和影响。在风险分析中务必考虑这一功能。

⚠ 警告

意外的设备操作

在风险分析中考虑到将 I/O 扩展模块标为可选，特别是将 TM3 安全模块 (TM3S...) 确定为可选 I/O 模块可以实现的每种 I/O 配置版本，并在它与您的应用程序相关时确定它是否可以接受。

不遵循上述说明可能导致人员伤亡或设备损坏。

在离线模式下将 I/O 扩展模块标记为可选模块

要添加模块并将其在配置中标记为可选模块：

步骤	动作
1	将 I/O 扩展模块从目录中拖放至编辑器中。
2	在 **设备信息** 区域中，选择 **可选模块** 复选框： 设备信息 ☑ 可选模块 消息 设备说明 TM3DI8（螺旋式）、TM3DI8G（弹簧式）8 通道 24 Vdc 数字输入扩展模块

要在配置中将现有 I/O 扩展模块标记为可选模块：

步骤	动作
1	在编辑器中选择 I/O 扩展模块。
2	在 **设备信息** 区域中，选择 **可选模块** 复选框：

在线模式下的可选 I/O 扩展模块

如果尚未与逻辑控制器建立物理连接，则 SoMachine Basic 将以在线模式运行。

当处于 SoMachine Basic 在线模式时，对 **可选模块** 功能的修改将被禁用。您可以在应用程序中将已下载的配置可视化：

- 以黄色表示的 I/O 扩展模块会被标记为可选模块，且不会在启动时与逻辑控制器建立物理连接。**设备信息** 区域中会显示一条关于该效果的参考性消息。
- 以红色表示的 I/O 扩展模块不会被标记为可选模块，且不会在启动时被检测。**设备信息** 区域中会显示一条关于该效果的参考性消息。

可选模块 功能的选择被逻辑控制器用于启动 I/O 总线。下列系统字会更新以指示物理 I/O 总线配置的状态：

系统字	注释
%SW118 逻辑控制器状态字	位 13 和 14 与有关于 I/O 总线的 I/O 模块状态相关。 如果状态为“假”，则位 13 表示：当逻辑控制器尝试启动 I/O 扩展总线时，按 I/O 总线配置的强制性模块不存在或者无法工作。此时，I/O 总线不会启动。 如果状态为“假”，则位 14 表示：在 I/O 扩展总线启动之后，一个或多个模块已停止与逻辑控制通信。这是一种无论 I/O 模块被定义为强制模块或可选模块都会在启动时出现的状况。
%SW119 I/O 扩展模块配置	从位 1 开始（位 0 已保留），每一个位都专用于一个已配置的 I/O 扩展模块，并指示该模块在控制器尝试启动 I/O 总线时是可选模块（真）还是强制模块（假）。
%SW120 I/O 扩展模块状态	从位 1 开始（位 0 已保留），每一个位都专用于一个已配置的 I/O 扩展模块，并指示该模块的状态。 当逻辑控制器尝试启动 I/O 模块时，如果 %SW120 值并非零（表示至少有一个模块被检出错误），则 I/O 扩展模块不会启动，除非 %SW119 中相应的位已设置为“真”（表示该模块已被标记为可选模块）。 当 I/O 模块启动时，如果系统修改了 %SW120 的值，这表明在一个或多个 I/O 扩展模块上检出了错误（与 **可选模块** 功能无关）。

有关更多信息，请参阅系统字 (参见第 *463* 页)。

共享的内部 ID 代码

逻辑控制器利用简单的内部 ID 代码来识别各扩展模块。该 ID 代码并不是每一个参考所特有的，但可以识别扩展模块的结构。因此，不同的参考可以共享同一 ID 代码。

如果在配置中声明具有相同内部 ID 代码的两个相邻模块，且两个模块均声明为可选模块，则在**配置**窗口底部将显示一条消息。两个可选模块之间必须至少有一个非可选模块。

下表对共享相同内部 ID 代码的模块参考进行了分组：

共享相同内部 ID 代码的模块
TM2DDI16DT、TM2DDI16DK
TM2DRA16RT、TM2DDO16UK、TM2DDO16TK
TM2DDI8DT、TM2DAI8DT
TM2DRA8RT、TM2DDO8UT、TM2DDO8TT
TM2DDO32TK、TM2DDO32UK
TM3DI16K、TM3DI16/G
TM3DQ16R/G、TM3DQ16T/G、TM3DQ16TK、TM3DQ16U、TM3DQ16UG、TM3DQ16UK
TM3DQ32TK、TM3DQ32UK
TM3DI8/G、TM3DI8A
TM3DQ8R/G、TM3DQ8T/G、TM3DQ8U、TM3DQ8UG

配置 M100/M200 可编程控制器

控制器配置

控制器配置取决于内置输入/输出、I/O 对象和通讯端口的数量和类型。

使用**配置**选项卡配置控制器和扩展模块的属性。在硬件树中选择一个节点以配置控制器的属性。

下表显示了 M100/M200 可编程控制器 的可用配置：

型号	数字量输入	数字量输出	高速计数器	脉冲发生器	以太网	串行线路
TM100C••R TM100C••RN TM200C••R	X	X	X	–	–	X
TM200C••U	X	X	X	X	–	X
TM200CE••R	X	X	X	–	X	X
TM200CE••U	X	X	X	X	X	X
TM200C••T	X	X	X	X	–	X
TM200CE••T	X	X	X	X	X	X

– 不适用于 SoMachine Basic 中的配置。

X 用于 SoMachine Basic 中的配置。有关如何配置的信息：
- 数字量输入，请参阅配置数字量输入 (参见第 *73* 页)
- 数字量输出，请参阅配置数字量输出 (参见第 *76* 页)
- 高速计数器，请参阅配置高速计数器 (参见第 *90* 页)
- 脉冲发生器，请参阅配置脉冲发生器 (参见第 *78* 页)
- 以太网，请参阅配置以太网 (参见第 *146* 页)
- 串行线路，请参阅配置串行线路 (参见第 *164* 页)。

使用 Executive Loader Wizard 更新固件

概述

您可以使用 Executive Loader 向导更新控制器的固件。

请参阅*控制器状态和行为*了解有关控制器固件状态的信息。 (参见第 *39* 页)

更新控制器的固件

要启动 Exec Loader Wizard，请遵循以下步骤：

步骤	动作
1	关闭所有 Windows 应用程序（包括虚拟计算机）。
2	单击**开始** → **程序** → **Schneider Electric** → **SoMachine Basic** → **SoMachine Basic 固件更新**，或者从 *SoMachine Basic 安装文件夹\Execloader* 文件夹中运行 *ExecLoaderWizard.exe*。
3	按照向导中的步骤完成固件更新。

第4章
内置输入/输出配置

概述

本章介绍如何配置 M100/M200 可编程控制器 的内置 I/O 对象。

内置输入/输出数目取决于控制器型号。有关详细信息，请参阅 M100/M200 可编程控制器 型号 *(参见 Modicon M100/M200 可编程控制器, 硬件指南)*的表。

本章包含了哪些内容？

本章包含了以下部分：

节	主题	页
4.1	配置数字量输入	73
4.2	配置数字量输出	76
4.3	配置脉冲发生器	78
4.4	配置频率发生器	87
4.5	配置高速计数器	89

第4.1节
配置数字量输入

配置数字量输入

简介

缺省情况下，所有数字量输入都用作常规输入。某些数字量输入是快速输入，可通过配置高速计数器 (参见第 *90* 页)使用，而其他输入可配置为事件输入。

数字量输入配置

下表介绍了如何配置数字量输入：

步骤	动作
1	单击硬件树中的**数字量输入**节点，显示数字量输入属性。 下图显示了编辑器区域中的数字量输入的属性：
2	编辑属性以配置数字量输入。 有关数字量输入配置参数的详细信息，请参阅下表。

下表介绍了数字量输入配置的每个参数：

参数	可编辑	值	缺省值	描述
已使用	否	True/False	False	表示输入点是否在程序中被使用。
地址	否	%I0.x	–	显示控制器上数字量输入的地址，其中 x 表示通道编号。 如果控制器具有 8 个数字量输入通道，则 x 在 0...7 的范围内变化。 如果控制器具有 16 个数字量输入通道，则 x 在 0...15 的范围内变化。 例如，%I0.2 是逻辑控制器的第三个数字量输入通道。
符号	是	–	–	用于将符号与数字量输入对象关联。 在**符号**列中双击，键入符号的名称，然后按 Enter 键。
使用者	否	*任何*	**滤波**	显示使用输入通道的组件的名称。 例如，如果子程序使用输入通道， 则此字段显示**用户逻辑**。此字段的可能值有： ● **用户逻辑** ● **滤波** ● **锁存** ● **运行/停止** ● **事件** ● **%HSCx** 其中 x 是控制器上的高速计数器实例 ● **%FCy** 其中 y 是控制器上的快速计数器实例 如果输入由多个操作使用，则在此字段中显示用逗号分隔的所有值。
滤波	是	**无滤波器** **3 毫秒** **12 毫秒**	**3 毫秒**	用于为输入通道选择噪声滤波器持续时间。 为数字量输入使用滤波器可减少控制器输入中的噪声。 如果为某个输入选择滤波器，则无法配置该输入用于以下项目： ● **锁存** ● **事件**
锁存	是	True/False	False	用于对配置为事件的输入（%I0.2 到 %I0.5）启用或禁用锁存。 缺省情况下，由于缺省值**过滤**，因此会禁用此选项。将**过滤**设置**无滤波器**即可启用**锁存**选项。 锁存允许记忆持续时间短于控制器扫描时间的脉冲。 如果脉冲持续时间短于一次扫描时间，且具有一个大于或等于 1 毫秒的值，则控制器将锁存该脉冲，然后在下一次扫描中进行更新。 如果为某个输入启用**锁存**， 则无法配置该输入用于以下项目： ● **滤波** ● **运行/停止** ● **事件**

参数	可编辑	值	缺省值	描述
运行/停止	是	True/False	False	用于将 1 个数字量输入配置为额外的运行/停止开关。 如果将某个数字量输入配置为“运行/停止”开关，则无法在任何其他功能块（例如，高速计数器功能块、快速计数器功能块等）中使用该输入。 如果为某个输入启用**运行/停止**，则无法配置该输入用于以下项目： ● **锁存** ● **事件**
事件	是	**未使用** **下降沿** **上升沿** **上升/下降沿**	**未使用**	用于选择触发输入%I0.2...%I0.5 的事件。 缺省情况下，由于缺省值为**过滤**，因此会禁用此选项。 将**过滤**设置为**无过滤器**即可启用**事件**选项。 从下拉列表中选择事件（**未使用**除外）时： ● 启用**优先级** 参数，让您能够设置事件的优先级。 ● 创建事件任务，并在**配置**选项卡中显示。
优先级	是	0...7	7	用于设置输入%I0.2...%I0.5 的触发事件的优先级。 可使用**优先级**参数（只有配置为事件的输入才能编辑）设置每个事件的优先级。 为配置的每个事件分配不同的优先级：如果 2 个事件具有相同优先级，则会在窗口中显示检测到的错误消息。
子程序	否	–	–	显示与配置为事件的输入关联的子程序编号。
注释	是	–	–	用于将注释与数字量输入对象关联。 在**注释**列中双击，键入可选注释，然后按 Enter 键。

在**编程**选项卡上显示了其他配置详细信息。有关详细信息，请参阅数字量输入 (%I) (参见第 *224* 页)。

第4.2节
配置数字量输出

配置数字量输出

简介

缺省情况下，所有数字量输出都用作常规输出。对于配有晶体管输出的控制器，两路输出为快速晶体管输出，可通过配置脉冲发生器 (参见第 *78* 页)使用。

数字量输出配置

下表介绍了如何配置数字量输出：

步骤	动作
1	单击硬件树中的**数字量输出**节点，显示数字量输出属性。 下图显示了编辑器区域中的数字量输出的属性：
2	编辑属性以配置数字量输出。 有关数字量输出配置参数的详细信息，请参阅下表。

下表介绍了数字量输出配置的每个参数：

参数	可编辑	值	缺省值	描述
已使用	否	True/False	False	表示输出点是否在程序中被使用。
地址	否	%Q0.x	–	显示控制器上数字量输出的地址，其中 x 表示通道编号。 如果控制器具有 8 个数字量输出通道， 则 x 在 0...7 的范围内变化。 如果控制器具有 16 个数字量输出通道，则 x 在 0...15 的范围内变化。 如果控制器具有 24 个数字量输出通道，则 x 在 0...23 的范围内变化。 例如：%Q0.2 是控制器上的第三个数字量输出通道。
符号	是	–	–	用于将符号与数字量输出对象关联。 在**符号**列中双击，键入符号的名称，然后按 Enter 键。
使用者	否	*任何*	*空*	显示使用输出通道的组件的名称。 例如，如果将输出通道用作状态警报，则会显示**警报**。
状态警报	是	True/False	False	用于为输出（%Q0.0 到 %Q0.7）启用或禁用状态警报。 只能为状态警报配置一个输出通道。 在程序中已使用的输出不能被配置为状态警报。 当控制器处于 RUNNING 状态时，状态警报的值为 1，在其他所有状态下，该值为 0。
故障预置值	是	1 或 0	0	指定该值，以便在逻辑控制器进入 STOPPED 或异常状态时应用到此输出（故障预置为 0 或故障预置为 1）。缺省值是 0。如果配置了**维护值**故障预置模式，则在逻辑控制器进入STOPPED 或异常状态时，输出仍保留其当前值。 针对配置为**状态警报**的输出禁用了此字段。
注释	是	–	–	用于将注释与数字量输出对象关联。 在**注释**列中双击，键入可选注释，然后按 Enter 键。

在**编程**选项卡上显示了其他配置详细信息。有关详细信息，请参阅数字量输出 (%Q) (参见第 *225* 页)。

第4.3节
配置脉冲发生器

配置脉冲发生器

简介

Pulse (PLS)、Pulse Width Modulation (PWM)、Pulse Train Output (PTO) 和 Frequency Generator (FREQGEN) 等脉冲发生器功能块用于在专用输出通道 %Q0.0 或 %Q0.1 上生成方波或调制波信号。

PWM 输出提供具有可变宽度和占空比的调制波信号，而 PTO 输出生成方波，用于在开环模式下控制线性单轴步进驱动器或伺服驱动器。PLS 可产生由程序指定数量的方波。

脉冲发生器配置

下表介绍了如何配置脉冲发生器：

步骤	动作
1	单击硬件树中的**脉冲发生器**节点，显示脉冲发生器属性。 下图展示了编辑器区域中的脉冲发生器的属性：
2	编辑属性以配置脉冲发生器输出。 有关脉冲发生器配置参数的详细信息，请参阅下表。

下表介绍了脉冲发生器配置的每个参数：

参数	可编辑	值	缺省值	描述
已使用	否	True/False	False	指示脉冲输出在程序中是否被使用。
地址	否	%PLSx %PWMx %PTOx %FREQGENx	%PLSx/ %PWMx/ %PTO/ %FREQGENx	显示 Pulse 输出、 Pulse Width Modulation输出、Pulse Train Output 或Frequency Generator 的地址， 其中 x 是输出数量。 如果控制器有 2 个脉冲发生器， 则 x 的值将是 0 和 1。例如，%PLS0 是控制器上第一个脉冲输出的地址。 如果您选择输出类型 **PLS**，则**地址**字段将仅显示 %PLSx，如果您选择 **PWM**，它将仅显示 %PWMx。
符号	是	–	–	用于指定与脉冲发生器对象关联的符号。 在**符号**列中双击，键入符号的名称，然后按 **Enter** 键。 **注意：**仅当配置脉冲发生器时，才启用此字段。
类型	否	**未配置** **PLS** **PWM** **PTO** **FREQGEN**	**未配置**	显示输出通道使用的脉冲发生器的类型。
配置	是	**[...]** (按钮)	已启用	用于选择脉冲发生器的类型。 单击此按钮可打开 **Pulse Generator Assistant** 窗口 (参见第 *80* 页)进行配置， 其中 x 是控制器上的脉冲发生器数量。
注释	是	–	–	用于指定与脉冲发生器对象关联的注释。 在**注释**列中双击，键入注释，然后按 **Enter** 键。

Pulse Generator Assistant 窗口

此图显示 Pulse Generator Assistant 窗口：

下表介绍了 Pulse Generator Assistant 窗口的参数：

参数	可编辑	值	缺省值	描述
常规				
Type of pulse generator	是	**未配置** PLS PWM PTO FREQGEN	**未配置**	用于选择脉冲发生器的类型，以及配置输出属性。 选择： ● **PLS**，可在 PLS 模式下配置输出通道。请参阅 PLS 配置 (参见第 *81* 页)。 ● **PWM**，可在 PWM 模式下配置输出通道。请参阅 PWM 配置 (参见第 *82* 页)。 ● **PTO**，可在 PTO 模式下配置输出通道。请参阅 PTO 配置 (参见第 *84* 页)。 ● **FREQGEN**，可在 FREQGEN 模式下配置输出通道。请参阅 FREQGEN 配置 (参见第 *87* 页)。

PLS 配置

下图显示了将**脉冲发生器的类型**设置为 PLS 时的**脉冲发生器助手**窗口：

下表介绍了在 PLS 模式下配置通道时可用的各个参数：

参数	可编辑	值	缺省值	描述
行为				
双字	是	False	True/False	用于在 Word（16 位）和 Double Word（32 位）的数据大小之间切换。 缺省情况下禁用此参数，这表示当前数据大小为 Word（16 位）。 启用此字段会将数据大小更改为 Double Word（32 位）。
周期				
时基	是	0.1 毫秒 1 毫秒 10 毫秒 1 秒	1 秒	用于选择频率测量的时基。
预设	是	有关 PLS 类型的脉冲发生器预设值的完整范围，请参阅下表。	0	用于指定脉冲输出的预设值。

下表显示了**预设**参数的值范围：

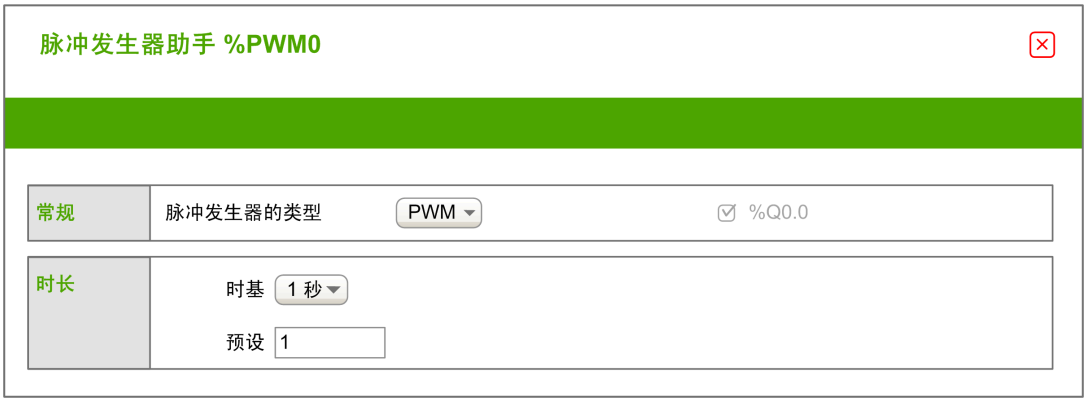
类型	时基	预设值范围
PLS	0.1 毫秒	1...20000
	1 毫秒	1...2000
	10 毫秒	1...200
	1 秒	1 或 2

在**编程**选项卡上显示了其他配置详细信息。

有关 Pulse 功能块的详细信息，请参阅脉冲 (%PLS) (参见第 *255* 页)。

PWM 配置

下图显示了将**脉冲发生器的类型**设置为 PWM 时的**脉冲发生器助手**窗口：



下表介绍了在 PWM 模式下配置通道时可用的各个参数：

参数	可编辑	值	缺省值	描述
周期				
时基	是	**0.1 毫秒** **1 毫秒** **10 毫秒** **1 秒**	**1 秒**	用于选择频率测量的时基。
预设	是	有关 PWM 类型的脉冲发生器预设值的完整范围，请参阅下表。	0	可用于指定 PWM 输出的预设值。

下表显示了**预设**参数的值范围：

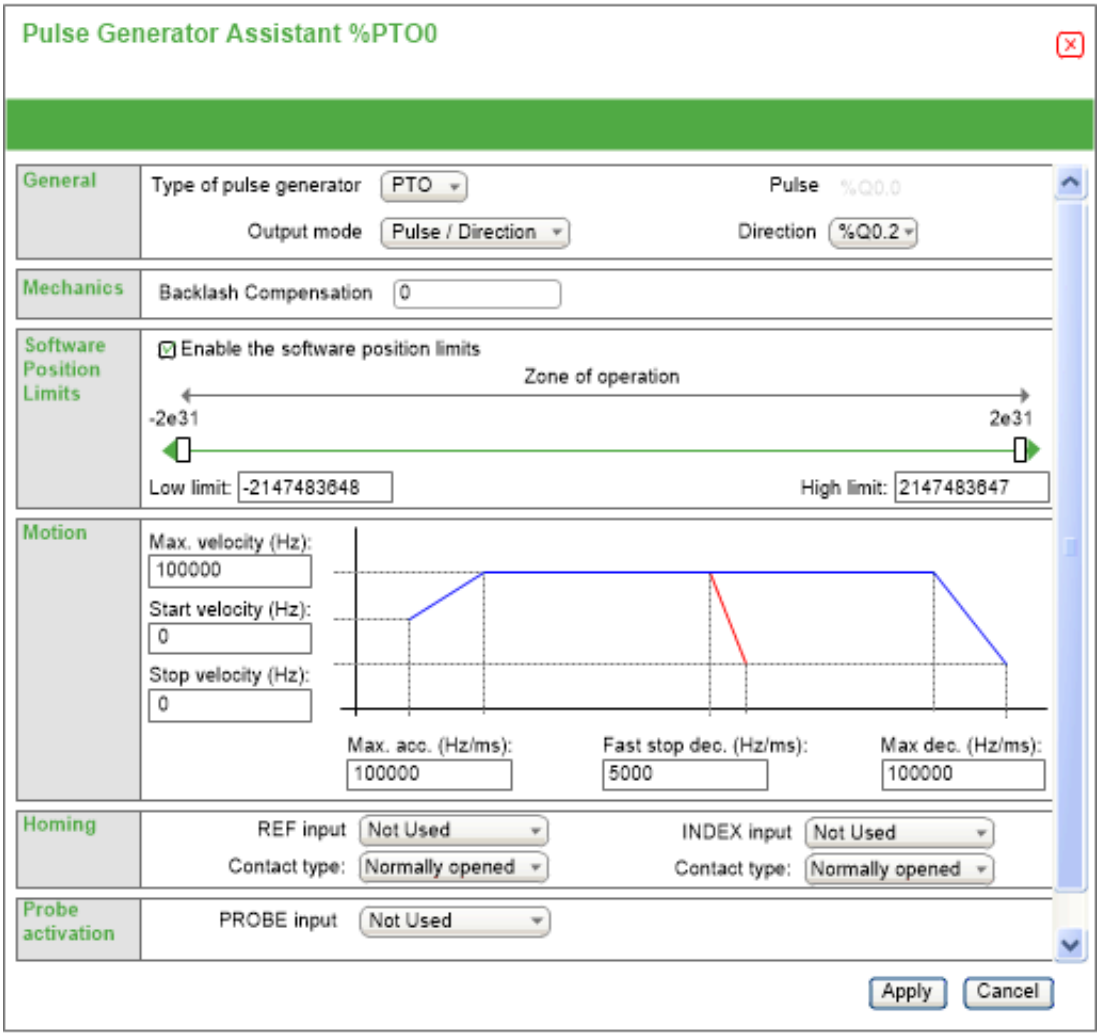
类型	时基	预设值范围
PWM	0.1 毫秒	1...10000
	1 毫秒	1...1000
	10 毫秒	1...100
	1 秒	1

在**编程**选项卡上显示了其他配置详细信息。

有关 Pulse Width Modulation 功能块的详细信息，请参阅脉冲宽度调制 (%PWM) (参见第 *291* 页)。

PTO 配置

下图显示了将**脉冲发生器的类型**设置为 **PTO** 时的**脉冲发生器助手**窗口：



下表描述了在 PTO 模式下配置通道时可用的各个参数：

参数	值	缺省	描述
常规			
输出模式 (参见第 *321* 页)	顺时针/逆时针输入 脉冲/方向	脉冲/方向	选择脉冲输出模式。 CW = 顺时针/CCW = 逆时针 **注意：** CW / CCW 输出模式仅对 PTO0 有效。此模式禁用了 PTO1。
脉冲	%Q0.0（对于 PTO0），%Q0.1（对于 PTO1）	%Q0.0（对于 PTO0），%Q0.1（对于 PTO1）	如果**输出模式**选择的是**脉冲/方向**，则选择提供电机工作速度的输出。
方向	**未使用** %Q0.0...15 (取决于控制器型号)	%Q0.2（对于 PTO0），%Q0.3（对于 PTO1）	如果**输出模式**选择的是**脉冲/方向**，则选择提供电机旋转方向的输出。 如果应用程序不需要方向输出，则设置为**未使用**（禁用）。 **注意：** 此应用程序必须配置至少 **5.0 级**的功能级别才能启用**未使用**选项。
顺时针	%Q0.0	%Q0.0	如果**输出模式**选择的是**顺时针/逆时针**，则选择针对正向电机工作速度和方向提供相应信号的输出。
逆时针	%Q0.1	%Q0.1	如果**输出模式**选择的是**顺时针/逆时针**，则选择针对反向电机工作速度和方向提供相应信号的输出。
力学			
反向间隙补偿	0...65535	0	设置反向间隙补偿值。不会将指定数量的间隙补偿脉冲增加到位置计数器。 请参阅反向间隙补偿 (参见第 *316* 页)。
软件位置限制			
启用软件位置限制	已启用 已禁用	已启用	选择是否使用软件位置限制 (参见第 *318* 页)。
下限	-2,147,483,648 ... 2,147,483,647	-2,147,483,648	设置要在反方向上检测的软件限制位置。
上限	-2,147,483,648 ... 2,147,483,647	2,147,483,647	设置要在正方向上检测的软件限制位置。
运动			
最大速度	0...100,000	100,000	设置脉冲输出最大速度（赫兹）。
启动速度	0...100,000	0	设置脉冲输出启动速度 (参见第 *312* 页)（赫兹）。未使用时设置为 0。
停止速度	0...100,000	0	设置脉冲输出停止速度 (参见第 *312* 页)（赫兹）。未使用时设置为 0。
最大加速度	1...100,000	100,000	设置加速度最大值（赫兹）。

参数	值	缺省	描述
快速停止减速度	1...100,000	5,000	如果检测到错误，设置减速度值（赫兹）
最大减速度	1...100,000	100,000	设置减速度最大值（赫兹）。
回归			
Enable the REF input	已启用 已禁用	已禁用	选择是否使用 REF 输入设置原点位置。
触点类型	常开 常闭	常开	选择开关触点缺省状态为开或为关。 **注意：** 仅当选择“Enable the REF input”时，此输入类型才可用。
启用 INDEX 输入 (%I0.x)	已启用 已禁用	已禁用	选择是否将 INDEX 输入用作正向限位信号。
触点类型	常开 常闭	常开	选择开关触点缺省状态为开或为关。 **注意：** 仅当选择“启用 INDEX 输入”时，此输入类型才可用。
探测器激活			
Enable the PROBE input	已启用 已禁用	已禁用	选择是否使用 PROBE 输入。 **注意：** 有关所选输入的物理特性，请参阅常规输入特性。

在**编程**选项卡上显示了其他配置详细信息。

有关 Pulse Train Output 功能块的详细信息，请参阅脉冲串输出 (%PTO) (参见第 *307* 页)。

第4.4节
配置频率发生器

配置频率发生器 (%FREQGEN)

FREQGEN 的脉冲发生器助手

下图显示在将**脉冲发生器的类型**设置为 **FREQGEN** 时的**脉冲发生器助手**窗口：

频率发生器 (FG) 功能会生成具有可编程频率且占空比为 50% 的方波信号。控制器使用一个内部时钟发生器，并在专用输出通道 (%Q0.0) 上提供输出信号。此输出信号可以直接命令轴的匀速运动。目标频率始终为正。

有关 FREQGEN 功能块的详细信息，请参阅频率发生器（%FREQGEN 章节）(参见第 *409* 页)。

第4.5节
配置高速计数器

本节包含了哪些内容？

本节包含了以下主题：

主题	页
配置高速计数器	90
配置单相和双相	94
配置频率计	99

配置高速计数器

简介

您可以将高速计数器配置为以下任一功能：

- 单相
- 双相[脉冲/方向]
- 双相[顺时针/逆时针输入]
- 双相 [积分 X1]
- 双相 [积分 X2]
- 双相 [积分 X4]
- 频率计

单相和**双相** HSC 类型计数器具有多种模式：

- 单相
  - 单次
  - 模数回路
  - 自由大型
- 双相
  - 模数回路
  - 自由大型

在单次模式中，应用到输入的每个脉冲都递减当前值。计数器在其当前值达到最大值时停止。然后溢出标记置位。更多详情，请参考单次计数模式 (参见第 *249* 页)。

在模数回路模式中，计数器反复从 0 计数到用户指定的模数值。然后归零，并重新开始计数。反过来，计数器从模数值减计数到 0，然后预设为模数值，再重新启动计数。更多详情，请参考模数回路计数模式 (参见第 *250* 页)。

在自由大型模式中，模块类似于标准的加减计数器。此模式可结合一个编码器使用。

高速计数器支持数字量输入最大频率为 100 kHz 的计数（在单字或双字计算模式下）。

专用 I/O 分配

High Speed Counter 功能块使用专用输入以及辅助输入和输出。这些输入和输出并非保留供 High Speed Counter 功能块专用。

如果程序将 %I0.0 用作常规数字量输入，则 %HSC0 不可用。

如果程序将 %I0.1 用作常规数字量输入，则 %HSC0 和 %HSC2 不可用。

如果程序将 %I0.6 用作常规数字量输入，则 %HSC1 不可用。

如果程序将 %I0.7 用作常规数字量输入，则 %HSC1 和 %HSC3 不可用。

下表概述了 %HSC0 和 %HSC1 的分配：

计数器类型	主输入		辅助输入		反射输出					
					TM200•••U TM200•••T		TM100C••R TM200•••R		TM100C••RN	
%HSC0	%I0.0	%I0.1	%I0.2	%I0.3	%Q0.4	%Q0.5	%Q0.0	%Q0.1	%Q0.0	%Q0.1
%HSC1	%I0.6	%I0.7	%I0.5	%I0.4	%Q0.6	%Q0.7	%Q0.2	%Q0.3	未使用	未使用
单相	脉冲输入	未使用	预设输入*	捕捉输入*	反射输出 R*	反射输出 S*	反射输出 R*	反射输出 S*	反射输出 R*	反射输出 S*
双相[脉冲/方向]	脉冲输入	方向输入	预设输入*	捕捉输入*	反射输出 R*	反射输出 S*	反射输出 R*	反射输出 S*	反射输出 R*	反射输出 S*
双相[顺时针/逆时针输入]	CW 输入相位 A	CCW 输入相位 B	预设输入*	捕捉输入*	反射输出 R*	反射输出 S*	反射输出 R*	反射输出 S*	反射输出 R*	反射输出 S*
双相[积分 X1]	脉冲输入相位 A	脉冲输入相位 B	预设输入*	捕捉输入*	反射输出 R*	反射输出 S*	反射输出 R*	反射输出 S*	反射输出 R*	反射输出 S*
双相[积分 X2]	脉冲输入相位 A	脉冲输入相位 B	预设输入*	捕捉输入*	反射输出 R*	反射输出 S*	反射输出 R*	反射输出 S*	反射输出 R*	反射输出 S*
双相[积分 X4]	脉冲输入相位 A	脉冲输入相位 B	预设输入*	捕捉输入*	反射输出 R*	反射输出 S*	反射输出 R*	反射输出 S*	反射输出 R*	反射输出 S*
频率计	脉冲输入	未使用	未使用	未使用	未使用	未使用	未使用	未使用	未使用	未使用
* 当不使用时，输入或输出用作可由应用程序在主任务循环中管理的正常数字量 I/O。										

下表概述了 %HSC2 和 %HSC3 的分配：

计数器类型	主输入		辅助输入		反射输出					
					TM200•••U TM200•••T		TM100C••R TM200•••R		TM100C••RN	
%HSC2	%I0.0	%I0.1	%I0.2	%I0.3	%Q0.4	%Q0.5	%Q0.0	%Q0.1	%Q0.0	%Q0.1
%HSC3	%I0.6	%I0.7	%I0.5	%I0.4	%Q0.6	%Q0.7	%Q0.2	%Q0.3	未使用	未使用
单相	未使用	脉冲输入	未使用							
* 当不使用时，输入或输出用作可由应用程序在主任务循环中管理的正常数字量 I/O。										

高速计数器配置

下表介绍了如何配置高速计数器：

步骤	动作
1	单击硬件树中的**高速计数器**节点，显示高速计数器属性。 **结果：**高速计数器的属性显示在编辑器区域： 高速计数器 已配置 | 地址 | 符号 | 类型 | 配置 | 注释 %HSC0 未配置 ... %HSC1 未配置 ... %HSC2 未配置 ... %HSC3 未配置 ... 有关高速计数器配置的详细信息，请参阅下表。
2	单击**[...]**按钮以配置高速计数器。 **结果：**高速计数器配置参数显示在助手窗口中。 有关使用助手配置单相和双相模式的更多详情，请参阅配置单相和双相 (参见第 *94* 页)。 有关使用助手配置频率计模式的更多详情，请参阅配置频率计 (参见第 *99* 页)。

下表描述了高速计数器配置的每个参数：

参数	可编辑	值	缺省值	描述
已使用	否	True/False	False	表示高速计数功能是否在程序中被使用。
地址	否	%HSCx		显示高速计数器的地址， 其中 x 是对象编号。例如，%HSC1 是控制器上第二个高速计数器的地址。
符号	是	–	–	用于将符号与高速计数器对象关联。 在**符号**列中双击，键入符号的名称， 然后按 Enter 键。
类型	是	**未配置** **单相** **双相** **频率计**	**未配置**	可从下拉列表中选择计数器操作模式。

参数	可编辑	值	缺省值	描述
配置	是	**[...]** (按钮)	已禁用	用于使用助手窗口配置高速计数器参数。只有从列表中选择高速计数器功能才会启用此按钮。当计数器类型是**未配置**时，将禁用助手配置按钮。 当您单击配置按钮时，将显示**高速计数器助手 (%HSCx)**窗口，其中 x 是控制器上的计数器编号。
注释	是	–	–	用于将注释与高速计数器对象关联。 在**注释**列中双击，键入可选注释，然后按 Enter 键。

配置单相和双相

带助手窗口的单相和双相

下图所示为将 %HSC0 配置为**双相[脉冲/方向]**的助手窗口的实例：

项	描述
1	显示助手对话框窗口的标题。 如果您正在配置计数器%HSC0，则窗口标题显示为 **高速计数器助手 (%HSC0)**，而若为计数器 %HSC1，则窗口标题显示为 **高速计数器助手 (%HSC1)**。
2	显示特定类型高速计数器的输入模式。

项	描述
3	显示专用输入、辅助输入和反射输出。 对于每个计数器类型和 %HSC 实例，助手窗口此区域中的属性都有所不同。有关更多详细信息，请参阅专用 I/O 分配 (参见第 *91* 页)一节。

下表介绍了 %HSC0 和 %HSC1 的高速计数器助手画面的每个参数：

参数	可编辑	值	缺省值	描述
预配置				
HSC 类型	是	**未配置** **单相** **双相** **频率计**	**未配置**	指示已选的计数器操作模式，您可以对其进行更改。
配置单相时的**计数模式**	是	**自由大型** **模数回路** **单次**	**自由大型**	指示已选的计数器操作模式，您可以对其进行更改。 选项取决于实例以及其他实例中的 HSC 类型。请参阅专用 I/O 分配 (参见第 *91* 页)。
配置双相时的**计数模式**	是	**自由大型** **模数回路**	**自由大型**	指示已选的计数器操作模式，您可以对其进行更改。 选项取决于实例以及其他实例中的 HSC 类型。请参阅专用 I/O 分配 (参见第 *91* 页)。
配置双相时的**输入模式**	是	**脉冲/方向** **顺时针/** **逆时针输入** **积分 X1** **积分 X2** **积分 X4**	**脉冲/方向**	指示已选的计数器操作模式，您可以对其进行更改。 选项取决于实例以及其他实例中的 HSC 类型。请参阅专用 I/O 分配 (参见第 *91* 页)。
常规				
双字	是	True/False	False	用于在Word（16 位）和Double Word（32 位）输入数据大小之间进行切换。缺省情况下禁用此参数，这表示当前数据大小为 Word（16 位）。 启用此字段会将数据大小更改为 Double Word（32 位）。
* 对于 **X1**，值限制为 2147483647				

参数	可编辑	值	缺省值	描述
预设	是	0...65535 (Word) 0...4294967295* (Double Word) 0...2147483647 (Double Word) 在配置**双相 [积分 X1]** 时	0 (Word) 0 (Double Word)	用于指定计数功能的预设值。
模数值 (%HSC.P/.PD)	是	0...65535 (Word) 0...4294967295* (Double Word) 0...2147483647（Double Word）在配置双相[积分 X1]时	0 (Word) 0 (Double Word)	可用于指定计数功能的模数值。
阈值 S0	是	0...65535 (Word) 0...4294967295* (Double Word) 0...2147483647（Double Word）在配置双相[积分 X1]时	65535 (Word) 4294967295* (Double Word)	用于指定包含阈值 TH0 的值的 HSC 标志 S0 的值。
阈值 S1	是	0...65535 (Word) 0...4294967295* (Double Word) 0...2147483647（Double Word）在配置双相[积分 X1]时	65535 (Word) 4294967295* (Double Word)	用于指定包含阈值 TH1 的值的 HSC 标志 S1 的值。
触发器	是	**未使用** **下降沿** **上升沿** **上升/下降沿**	**未使用**	用于从下拉列表中选择事件（对于阈值 TH0 和 TH1）的触发功能。 如果从下拉列表中选择某个触发功能（非**未使用**），将能够编辑**优先级**参数，以设置事件的优先级。
优先级	是	0...7	7	用于设置事件（对于阈值 TH0 和 TH1）的触发功能的优先级。 仅当您为事件选择**触发**功能时启用此字段。
* 对于 **X1**，值限制为 2147483647				

参数	可编辑	值	缺省值	描述
子程序	否	*任何*	*空*	显示与配置为事件（对于阈值 TH0 和 TH1）的输入关联的子程序。
输入配置				
配置**单相**时：				
脉冲输入	否	True/False	True	● 对于 %HSC0：%I0.0 用作脉冲输入。 ● 对于 %HSC1：%I0.6 用作脉冲输入。
在配置**双相[脉冲/方向]** 时：				
脉冲输入	否	True/False	True	● 对于 %HSC0：%I0.0 用作脉冲输入。 ● 对于 %HSC1：%I0.6 用作脉冲输入。
方向输入	否	True/False	True	● 对于 %HSC0：%I0.1 用作方向输入。 ● 对于 %HSC1：%I0.7 用作方向输入。
配置**双相时[顺时针/逆时针输入]**：				
顺时针输入相位 A	否	True/False	True	● 对于 %HSC0：%I0.0 用作相位 A 的脉冲输入。 ● 对于 %HSC1：%I0.6 用作相位 A 的脉冲输入。
逆时针输入相位 B	否	True/False	True	● 对于 %HSC0：%I0.1 用作相位 B 的脉冲输入。 ● 对于 %HSC1：%I0.7 用作相位 B 的脉冲输入。
配置**双相[积分 X1]**、**双相[积分 X2]**和**双相[积分 X4]**时：				
脉冲输入相位 A	否	True/False	True	● 对于 %HSC0：%I0.0 用作相位 A 的脉冲输入。 ● 对于 %HSC1：%I0.6 用作相位 A 的脉冲输入。
脉冲输入相位 B	否	True/False	True	● 对于 %HSC0：%I0.1 用作相位 B 的脉冲输入。 ● 对于 %HSC1：%I0.7 用作相位 B 的脉冲输入。
配置**单相**和**双相**时：				
正常输入	是	True/False	False	● 对于 %HSC0：%I0.2 用作正常输入。 ● 对于 %HSC1：%I0.5 用作正常输入。 单击**用作**复选框，将此输入用作**预设输入**。
正常输入	是	True/False	False	● 对于 %HSC0：%I0.3 用作正常输入。 ● 对于 %HSC1：%I0.4 用作正常输入。 单击**用作**复选框，将此输入用作**捕捉输入**。
* 对于 **X1**，值限制为 2147483647				

参数	可编辑	值	缺省值	描述
反射输出配置				
反射输出 0	否	True/False	False	用于启用或禁用以下地址处的反射输出： 适用于 TM200•••U/TM200•••T: ● %Q0.4（对于 %HSC0） ● %Q0.6（对于 %HSC1） 适用于 TM100•••R/TM200•••R: ● %Q0.0（对于 %HSC0） ● %Q0.2（对于 %HSC1）
反射输出 1	否	True/False	False	用于启用或禁用以下地址处的反射输出： 适用于 TM200•••U/TM200•••T: ● %Q0.5（对于 %HSC0） ● %Q0.7（对于 %HSC1） 适用于 TM100•••R/TM200•••R: ● %Q0.1（对于 %HSC0） ● %Q0.3（对于 %HSC1）
值 < S0	是	True/False	False	用于启用或禁用将计数器与输出值持续进行比较的条件，以便在输出值小于 HSC 标志 S0 的值时设置反射输出。
S0 <= 值 < S1	是	True/False	False	用于启用或禁用将计数器与输出值持续进行比较的条件，以便在输出值属于以下情况时设置反射输出： ● 大于或等于 HSC 标志 S0 的值，且 ● 小于 HSC 标志 S1 的值。 **注意：** 配置为模数回路模式时，S1 不是一个变量。
值 >= S1	是	True/False	False	用于启用或禁用将计数器与输出值持续进行比较的条件，以便在输出值大于或等于 HSC 标志 S1 的值时设置反射输出。 **注意：** 配置为模数回路模式时，S1 不是一个变量。
* 对于 **X1**，值限制为 2147483647				

配置频率计

频率计助手窗口

下图显示了计数器类型**频率计**的**高速计数器助手 (%HSC0)** 窗口：

下表介绍了计数器类型**频率计**的**高速计数器助手 (%HSCx)** 窗口的每个参数：

参数	可编辑	值	缺省值	描述
HSC 类型	是	**未配置** **单相** **双相** **频率计**	**未配置**	指示已选的计数器操作模式， 您可以对其进行更改。 Frequency Meter可在 %HSC0 和/或 %HSC1 上进行配置。请参阅频率计 I/O 分配。
常规				
双字	是	TRUE/FALSE	FALSE	可用于在字（16 位）和双字（32 位）输入数据大小之间切换。 启用此字段将会使数据大小从单字（16 位）转变为双字（32 位）。
时间窗口 100 毫秒 (1)	是	TRUE/FALSE	FALSE	可用于选择 100 毫秒的时基，以测量 100 Hz 和 100 kHz 之间的频率。
时间窗口 1 秒(1)	是	TRUE/FALSE	TRUE	可用于选择 1 秒的时基， 以测量 100 Hz 和 100 kHz 之间的频率。
输入				
脉冲输入	否	TRUE/FALSE	TRUE	指示用作脉冲输入的输入， 即 %I0.0（对于 %HSC0）或 %I0.6（对于 %HSC1）。
(1) 默认情况下，时基值设置为 1 秒。您只能为计数器选择 1 个时基值。				

在**编程**选项卡上显示了其他配置详细信息。

有关 High Speed Counter 功能块的更多详细信息，请参阅高速计数器功能块 (%HSC) (参见第 *236* 页)。

第5章
I/O 总线配置

概述

本章介绍如何配置 M100/M200 可编程控制器 的 I/O 总线（扩展模块）。

本章包含了哪些内容？

本章包含了以下主题：

主题	页
I/O 配置概述	102
配置扩展模块	105

I/O 配置概述

简介

在项目中，您可以将 I/O 扩展模块添加到 M100/M200 可编程控制器 以增加数字量和模拟量输入与输出的数量（相对于逻辑控制器本身（嵌入式 I/O）自有的数量）。

可以将 TM3、TM3R 或 TM2 I/O 扩展模块添加到逻辑控制器。当创建本地和远程 I/O 扩展时，以及当混合 TM2、TM3 和 TM3R I/O 扩展模块（请参阅最大硬件配置）时，特殊规则适用所有情况。

当您把 I/O 扩展模块组装到逻辑控制器上时，M100/M200 可编程控制器 的 I/O 扩展总线将会生成。I/O 扩展模块在逻辑控制器架构中被视为外部设备，这样它们就与逻辑控制器的嵌入式 I/O 区别对待。

I/O 扩展总线错误

如果逻辑控制器无法与程序配置中包含的一个或多个 I/O 扩展模块通讯，并且这些模块未配置为可选模块（请参阅可选 I/O 扩展模块），则逻辑控制器视其为 I/O 扩展总线错误。不成功通讯可在逻辑控制器启动期间或实时系统期间检测到，并且有可能存在任何数量的原因。I/O 扩展总线上通讯异常的原因包括但不限于 I/O 模块断开或物理上缺失，电磁辐射超出公布的环境标准，或模块以其他方式无法工作。

在实时系统期间，如果检测到 I/O 扩展总线错误，则诊断信息包含在 %SW118 和 %SW120 系统字 (参见第 *463* 页)中，并且红色 LED 指示灯（标记有 **ERR**）闪烁。

主动 I/O 扩展总线错误处理

系统位 %S106 缺省设置为 0，以指定对活动 I/O 错误处理的使用。应用程序可以将这个位设置为 1 以改为使用被动 I/O 错误处理。

缺省情况下（系统位 %S106 设置为 0），逻辑控制器在检测到 TM3 模块存在总线通讯错误时，会将总线设置为“总线关闭”状态，在这种情况下，TM3 扩展模块输出设置为 0。如果 I/O 在至少两个连续的总线任务循环内都未能与扩展模块成功进行数据交换，则 TM3 扩展模块被视为存在总线通讯错误。出现总线通讯错误时，%SW120 的位 n 设置为 1，其中 n 是扩展模块的数量，并且 %SW118 位 14 设置为 0。

只有在消除了错误源并且执行了以下其中一种操作之后，才能恢复 I/O 扩展总线的正常操作：

- 电源重置
- 下载新应用程序
- 通过位 %S107 上的上升沿发出应用程序请求
- 利用 SoMachine Basic，选择**初始化控制器**命令

被动 I/O 扩展总线错误处理

应用程序可以将系统位 %S106 设置为 1 以使用被动 I/O 错误处理。这种错误处理旨在实现与先前固件版本以及 M100/M200 可编程控制器 所替换的先前控制器兼容。

在使用 被动 I/O 错误处理的情况下，控制器会尝试以检测到总线通讯错误的模块继续进行数据总线交换。尽管仍存在扩展总线错误，逻辑控制器也会尝试在总线上重新建立与不通讯模块之间的通讯，具体取决于 I/O 扩展模块、TM3 或 TM2 的类型：

- 对于 TM3 I/O 扩展模块，I/O 通道的值保持（**保持当前值**）大约 10 秒，同时，逻辑控制器尝试重新建立通讯。如果逻辑控制器在该时间内无法重新建立通讯，则所有受影响的 TM3 I/O 扩展输出设置为 0。
- 对于有可能为配置组成部分的 TM2 I/O 扩展模块，I/O 通道的值保持不确定。这就是说，TM2 I/O 扩展模块的输出设置为**保持当前值**，直至逻辑控制器系统重置电源或者您使用 SoMachine Basic 发出**初始化控制器**命令。

在任一情况下，逻辑控制器继续解决逻辑问题，并且嵌入式 I/O 继续由应用程序（由应用程序管理 (参见第 *49* 页)）管理，同时，它尝试重新建立与不通讯 I/O 扩展模块之间的通讯。如果通讯成功，则 I/O 扩展模块将恢复由应用程序管理。如果与 I/O 扩展模块的通讯不成功，您必须解决通讯不成功的原因，然后循环逻辑控制器系统的电源或使用 SoMachine Basic 发出**初始化控制器**命令。

此外，如果不通讯的 I/O 模块干扰到不受影响的模块的通讯，则不受影响的模块也将被视为存在错误，它们在 %SW120 中的相应位也将设置为 1。但在以被动 I/O 扩展总线错误处理来继续进行数据交换的情况下，不受影响的模块不会应用所发送的数据，而且也不会像不通讯模块那样应用故障预置值。

因此，您必须在应用程序中监视总线状态以及总线上模块的错误状态，并采取应用程序中提供的相应必要措施。

⚠ 警告

意外的设备操作

- 您的风险评估中应包含逻辑控制器与任何 I/O 扩展模块之间发生通讯失败的可能性。
- 如果在 I/O 扩展总线错误期间部署的“保持当前值”与您的应用程序不兼容，应使用备用方案来控制应用程序以应对此类事件。
- 使用专用系统字监控 I/O 扩展总线的状态并采取风险评估确定的适当措施。

不遵循上述说明可能导致人员伤亡或设备损坏。

有关检测到 I/O 扩展总线错误的情况下启动逻辑控制器时所采取措施的更多信息，请参阅可选 I/O 扩展模块。

重启 I/O 扩展总线

如果正应用主动 I/O 错误处理，即，在检测到总线通讯错误时 TM3 输出设置为 0，则应用程序可以在逻辑控制器仍在运行的情况下请求重启 I/O 扩展总线（不需要执行冷启动、热启动、电源重置或应用程序下载）。

系统位 %S107 可用于请求重启 I/O 扩展总线。这个位的缺省值为 0。应用程序可以将 %S107 设置为 1 以请求重启 I/O 扩展总线。若检测到这个位的上升沿，逻辑控制器会重新配置并重启 I/O 扩展总线，但前提是满足以下全部条件：

- %S106 设置为 0（即，I/O 扩展总线的活动停止）
- %SW118 位 14 设置为 0（I/O 扩展总线存在错误）
- %SW120 的至少一个位设置为 1（至少一个扩展模块存在总线通讯错误）

如果 %S107 设置为 1，且上述条件中有任一个条件未满足，逻辑控制器不会执行任何动作。

匹配硬件和软件配置

可在控制器中嵌入的 I/O 独立于采用 I/O 扩展的形式添加的 I/O。程序中的逻辑 I/O 配置应与安装的物理 I/O 配置匹配，这十分重要。如果对 I/O 扩展总线添加或删除任何物理 I/O，或根据控制器型号，对控制器进行添加或删除操作（以扩展板的形式），则必须更新应用程序配置。这也适用于安装中包含的任何现场总线设备。否则，扩展总线或现场总线可能不再正常工作，而控制器中可能存在的嵌入式 I/O 会继续操作。

⚠ 警告

意外的设备操作

每次添加或删除 I/O 总线上任何类型的 I/O 扩展，或添加或删除现场总线上的任何设备时，都需更新程序配置。

不遵循上述说明可能导致人员伤亡或设备损坏。

配置扩展模块

简介

在您的项目中，可以向控制器添加以下设备：

- TM3 数字量 I/O 模块
- TM3R 数字量混合 I/O 模块
- TM3 模拟量 I/O 模块
- TM2 数字量 I/O 模块
- TM2 模拟量 I/O 模块

TM3/TM3R扩展模块

有关模块配置的详细信息，请参阅下面各个扩展模块类型的编程指南和硬件指南：

扩展模块类型	硬件指南	编程指南
TM3 数字量 I/O 扩展模块	TM3 数字量 I/O 扩展模块硬件指南	TM3 扩展模块 - 编程指南
TM3R 数字量混合 I/O 扩展模块	Modicon M100/M200 可编程控制器硬件指南	TM3R 扩展模块配置 (参见第 *137* 页)
TM3 模拟量 I/O 扩展模块	TM3 模拟量模块硬件指南	TM3 扩展模块 - 编程指南

TM2 扩展模块

有关模块配置的详细信息，请参阅各个扩展模块类型的编程指南和硬件指南：

扩展模块类型	硬件指南	编程指南
TM2 数字量 I/O 模块	TM2 数字量 I/O 模块硬件指南	TM2 扩展模块 - 编程指南
TM2 模拟量 I/O 模块	TM2 模拟量 I/O 模块硬件指南	

第6章
扩展板配置

概述

本章介绍如何配置 M100/M200 可编程控制器 的扩展板。

本章包含了哪些内容？

本章包含了以下部分：

节	主题	页
6.1	扩展板配置一般信息	108
6.2	TMCR2••• 扩展板配置	113

第6.1节
扩展板配置一般信息

本节包含了哪些内容？

本节包含了以下主题：

主题	页
一般描述	109
在配置中使用扩展板	111
配置扩展板	112

一般描述

简介

TMCR2••• 扩展板连接至 Modicon M100/M200 可编程控制器，从而增加了控制器上的可用 I/O 或串行线路的数量。

扩展板可能是：
- 数字量扩展板
- 模拟量扩展板
- 串行线路扩展板

扩展板的功能

下表描述 TMCR2••• 扩展板的功能：

型号	描述
TMCR2DM4U	带有 2 路数字量输入和 2 路漏极晶体管输出的 TMCR2 扩展板
TMCR2AI2	带有 2 路模拟量电压或电流输入（0 到 10 V、0 到 20 mA 以及 到 20 mA）的 TMCR24 扩展板，12 位
TMCR2AQ2C	带有 路模拟量电流输出（4 到 20 mA）的 TMCR22 扩展板，12 位
TMCR2AQ2V	带有 路模拟量电压输出（0 到 10 V）的 TMCR22 扩展板，12 位
TMCR2AM3	带有 2 路模拟量电压输入和 1 路模拟量电压输出（0 到 10 V）的 TMCR2 扩展板，16 位
TMCR2TI2	带有 2 路模拟量温度输入（热电偶，RTD）的 TMCR2 扩展板，14 位
TMCR2SL1	带有 1 路串行线路（RS-232 或 RS-485）的 TMCR2 扩展板
TMCR2SL1A	带有 1 路隔离串行线路的 TMCR2 扩展板 (RS-485)

逻辑控制器的兼容性

注意： 有关兼容特定控制器的扩展板的详细信息，请参阅 Modicon M100/M200 可编程控制器 硬件指南 *(参见 Modicon M100/M200 可编程控制器, 硬件指南)*。

下表描述了可在 Modicon M100/M200 可编程控制器 中安装的 TMCR2••• 扩展板数量：

型号	扩展板插槽	兼容的扩展板组合	
		TMCR2DM4U TMCR2AI2 TMCR2AQ2V TMCR2AQ2C TMCR2AM3 TMCR2TI2	TMCR2SL1 TMCR2SL1A
TM200C•16• TM200C•24•	1	1	0
		0	1
TM200C•40• TM200C•60•	2 (1)	1	0
		0	1
		1	1
		2	0
(1) 只能将一个串行线路扩展板（TMCR2SL1 或 TMCR2SL1A）添加到逻辑控制器。			

注意

静电释放

- 在给控制器通电之前，请确认空扩展板已盖有护盖。
- 请勿触摸扩展板的触点。
- 仅通过外壳处理扩展板。
- 采取必要的防静电措施。

不遵循上述说明可能导致设备损坏。

在配置中使用扩展板

添加扩展板

TMCR2••• 扩展板可连接至带有 1 个或 2 个扩展板的 M100/M200 可编程控制器。

注意： 无法将 2 个串联线路扩展板添加至相同的逻辑控制器。有关扩展板与 M100/M200 可编程控制器 的兼容性的详细信息，请参阅逻辑控制器兼容性 (参见第 *110* 页)。

注意： 控制器必须至少有一个空闲的扩展板。

以下步骤阐述了如何将扩展板添加至 SoMachine Basic 配置的逻辑控制器：

步骤	描述
1	单击 SoMachine Basic 窗口中的**配置**选项卡。
2	在窗口的硬件目录区域中，选择 **M200 扩展板**。
3	选择扩展板的型号。 **结果**：所选扩展板的物理特性描述会显示在 SoMachine Basic 窗口的右下角。
4	将扩展板拖放到 M100/M200 可编程控制器 的空扩展板上。 **结果**：扩展板已添加至设备树的 **MyController → IO 总线**区域。 对于串行线路扩展板，显示 **SL2（串行线路）**节点。对于模拟量扩展板，**模拟量输入**或**模拟量输出**子节点会立即显示在扩展板型号下方。 以下有关所选扩展板的信息随即显示在 SoMachine Basic 窗口较低的中心区域： ● 扩展板电流状态的相关信息。 ● 对于应用扩展板，显示用于扩展板的工程模板列表。

替换现有扩展板

要用另一个型号替换现有扩展板，请将新扩展板拖放到要替换的扩展板上。

此时会显示一条消息，请您确认操作。单击**是**以继续。

卸下扩展板

要卸下控制器的扩展板，请单击扩展板并按下 **Delete** 键，或右键单击扩展板，然后单击随后显示的上下文菜单上的**删除**。

如果扩展板至少包含一个用于程序用户逻辑的地址，则会显示一条消息，要求您确认操作。单击**是**以继续。

配置扩展板

概述

您可以在以下选项卡上配置扩展板：

- **配置**选项卡
- **编程**选项卡

显示配置详细信息

以下步骤介绍如何在**配置**选项卡中查看数字量输入的配置：

步骤	描述
1	选择**配置**选项卡。
2	对于数字量扩展板，选择 SoMachine Basic 窗口左侧设备树中的**扩展板 x → 数字量输入**或**扩展板 x → 数字量输出**。 对于模拟量扩展板，选择 SoMachine Basic 窗口左侧设备树中的**扩展板 x → 模拟量输入**或**扩展板 x → 模拟量输出**。 对于串行线路扩展板，选择 SoMachine Basic 窗口左侧设备树中的**扩展板 x → SL2（串行线路）**。 随即会显示所选扩展板的属性。
3	有关配置详细信息，请参阅 TMCR2 标准扩展板配置 (参见第 *113* 页)。

显示编程属性

通过**编程**选项卡，您可以配置数字量或模拟量扩展板的编程相关的属性，如符号和注释。

要在**编程**选项卡中显示扩展板属性：

步骤	描述
1	选择**编程**选项卡。
2	对于数字量扩展板，单击**工具 → I/O 对象 → 数字量输入**或**工具 → I/O 对象 → 数字量输出** 对于模拟量扩展板，单击**工具 → I/O 对象 → 模拟量输入**或**工具 → I/O 对象 → 模拟量输出** I/O 地址列表会显示在 SoMachine Basic 窗口的中心区域下方（对于数字量输入为 %I，对于数字量输出为 %Q，对于模拟量输入为 %IW，对于模拟量输出为 %QW）。
3	向下滚动到与您配置的扩展板对应的地址范围：随即会显示以下属性： ● **已使用**。该地址是否正在您的程序中使用 ● **地址**。模拟量输入或模拟量输出地址。 ● **符号**。与地址相关的可选符号。 在**符号**列中双击并键入要与此输入关联的符号名称。 如果某个符号已经存在，则右键单击**符号**列，然后选择**搜索并替换**，在应用中查找并替换出现此符号的地方。 ● **注释**。与地址关联的可选注释。 在**注释**列中双击并键入与此地址关联的注释。

第6.2节
TMCR2••• 扩展板配置

本节包含了哪些内容？

本节包含了以下主题：

主题	页
TMCR2DM4U	114
TMCR2AI2	115
TMCR2AQ2C	117
TMCR2AQ2V	118
TMCR2AM3	119
TMCR2TI2	121
TMCR2SL1	124
TMCR2SL1A	129
TMCR2模拟量扩展板诊断	134

TMCR2DM4U

简介

TMCR2DM4U 是指具有 2 路数字量漏极/源极输入和 2 路数字量漏极晶体管输出的标准 M200 扩展板。

有关详细的硬件信息，请参阅 TMCR2DM4U *(参见 Modicon M100/M200 可编程控制器, 硬件指南)*。

配置扩展板模块

对于每个数字量输入，您可以定义：

参数	值	缺省值	描述
已使用	真/假	假	指示地址是否正在程序中使用。
地址	%I0. x0y	-	显示输入通道的地址，其中 *x* 为扩展板编号，*y* 为通道编号
符号		-	指定与程序中使用的相应数字量输入对象关联的可选符号。
注释		-	键入要与相应数字量输入对象关联的可选注释。

对于每个数字量输出，您可以定义：

参数	值	缺省值	描述
已使用	真/假	假	指示地址是否正在程序中使用。
地址	%Q0. x0y	-	显示输出通道的地址，其中 *x* 为扩展板编号，*y* 为通道编号
符号		-	指定与程序中使用的相应数字量输出对象关联的可选符号。
故障预置值	0...1	0	指定输出通道的故障预置值。
注释		-	键入要与相应数字量输出对象关联的可选注释。

TMCR2AI2

简介

TMCR2AI2 是指拥有 2 个 12 位精度模拟量电压或电流输入通道的标准扩展板。

通道输入类型为：

- 0 到 10 V
- 0...20 mA
- 4...20 mA

有关详细的硬件信息，请参阅 TMCR2AI2 *(参见 Modicon M100/M200 可编程控制器, 硬件指南)*。

例如，如果已物理连接用于电压信号的模拟量通道，并在 SoMachine Basic 中为电流信号配置了通道，则可能会损坏模拟电路。

注意
设备无法操作 确认模拟电路的物理线路与模拟量通道的软件配置兼容。 **不遵循上述说明可能导致设备损坏。**

配置模块

对于每个输入，您可以定义：

参数		值	缺省值	描述
已使用		真/假	假	指示地址是否正在程序中使用。
地址		%IW0.x0y	-	输入通道的地址，其中 *x* 为扩展板编号，*y* 为通道编号
符号		-	-	双击并键入可选的符号，与要在程序中使用的相应模拟量输入对象关联。
类型		**0 - 10 V** **0 - 20 mA** **4 - 20 mA**	**0 - 10 V**	选择通道模式。
范围		**正常**	**正常**	通道值的范围。
最小值	**0 - 10 V**	-32768...32767	0	指定测量下限。
	0 - 20 mA		0	
	4 - 20 mA		4000	
最大值	**0 - 10 V**	-32768...32767	10000	指定测量上限。
	0 - 20 mA		20000	
	4 - 20 mA		20000	
滤波器		0...100	0	指定滤波值。乘以**滤波单位**的值以获得滤波时间。

参数	值	缺省值	描述
滤波单位	100 毫秒	100 毫秒	设定滤波单位。
单位	-	-	-
注释	-	-	双击并键入要与通道关联的可选注释。

TMCR2AQ2C

简介

TMCR2AQ2C 是指拥有 2 个 12 位精度模拟量电流输出通道的标准扩展板。

通道输出类型为：

- 4...20 mA

有关详细的硬件信息，请参阅 TMCR2AQ2C *(参见 Modicon M100/M200 可编程控制器, 硬件指南)*。

例如，如果已物理连接用于电压信号的模拟量通道，并在 SoMachine Basic 中为电流信号配置了通道，则可能会损坏模拟电路。

注意
设备无法操作 确认模拟电路的物理线路与模拟量通道的软件配置兼容。 **不遵循上述说明可能导致设备损坏。**

配置扩展板模块

对于每个输出，您可以定义：

参数	值	缺省值	描述
已使用	真/假	假	指示地址是否正在程序中使用。
地址	%QW0.x0y	-	显示输出通道的地址， 其中 *x* 为扩展板编号，*y* 为通道编号
符号	-	-	双击并键入可选的符号，与要在程序中使用的相应模拟量输出对象关联。
类型	**4 - 20 mA**	**4 - 20 mA**	通道模式。
范围	**正常**	**正常**	通道值的范围。
最小值	-32768...32767	4000	指定测量下限。
最大值	-32768...32767	20000	指定测量上限。
故障预置值	**最小值**...**最大值**	0（**最小值**，如果范围之中不包含 0）	指定输出通道的故障预置值。
单位	-	-	-
注释	-	-	双击并键入要与通道关联的可选注释。

TMCR2AQ2V

简介

TMCR2AQ2V 是指 2 个具有 12 位精度的模拟量电压输出通道的标准扩展板。

通道输出类型为：

- 0...10 V

有关详细的硬件信息，请参阅 TMCR2AQ2V *(参见 Modicon M100/M200 可编程控制器, 硬件指南)*。

例如，如果已物理连接用于电压信号的模拟量通道，并在 SoMachine Basic 中为电流信号配置了通道，则可能会损坏模拟电路。

注意
设备无法操作 确认模拟电路的物理线路与模拟量通道的软件配置兼容。 **不遵循上述说明可能导致设备损坏。**

配置扩展板模块

对于每个输出，您可以定义：

参数	值	缺省值	描述
已使用	真/假	假	指示地址是否正在程序中使用。
地址	%QW0.x0y	-	显示输出通道的地址， 其中 *x* 为扩展板编号，*y* 为通道编号
符号	-	-	双击并键入可选的符号，与要在程序中使用的相应模拟量输出对象关联。
类型	**0 - 10 V**	**0 - 10 V**	通道模式。
范围	**正常**	**正常**	通道值的范围。
最小值	-32768...32767	0	指定测量下限。
最大值	-32768...32767	10000	指定测量上限。
故障预置值	**最小值**...**最大值**	0（**最小值**，如果范围之中不包含 0）	指定输出通道的故障预置值。
单位	-	-	-
注释	-	-	双击并键入要与通道关联的可选注释。

TMCR2AM3

简介

TMCR2AM3 是指拥有 2 个 16 位精度模拟量电流或电压输入通道和 1 个 16 位精度模拟量电流或电压输出通道的标准 M200 扩展板。

通道输入类型有：
- 0...5 V
- 0...10 V
- 0...20 mA
- 4...20 mA

通道输出类型有：
- 0...5 V
- 0...10 V
- 0...20 mA
- 4...20 mA

有关详细的硬件信息，请参阅 TMCR2AM3 *(参见 Modicon M100/M200 可编程控制器, 硬件指南)*。

例如，如果已物理连接用于电压信号的模拟量通道，并在 SoMachine Basic 中为电流信号配置了通道，则可能会损坏模拟电路。

注意
设备无法操作 确认模拟电路的物理线路与模拟量通道的软件配置兼容。 **不遵循上述说明可能导致设备损坏。**

配置扩展板模块

对于每个输入，您可以定义：

参数	值	缺省值	描述
已使用	真/假	假	指示地址是否正在程序中使用。
地址	%IW0.x0y	-	显示输入通道的地址， 其中 *x* 为扩展板编号，*y* 为通道编号
符号	-	-	双击并键入可选的符号，与要在程序中使用的相应模拟量输入对象关联。
类型	0 - 5 V 0 - 10 V 0 - 20 mA 4 - 20 mA	0 - 5 V	通道模式。

参数	值	缺省值	描述
范围	**自定义** **正常**	**自定义**	通道值的范围。
最小值	-32768...32767	4000	指定测量下限。
最大值	-32768...32767	20000	指定测量上限。
滤波器	0...6	0	指定要在此通道上应用的滤波水平： ● 0：未滤波 ● 1、2：平滑滤波 ● 3、4：均值滤波 ● 5、6：高滤波
滤波器单位		-	-
单位		-	-
注释			双击并键入要与通道关联的可选注释。

对于输出，您可以定义：

参数	值	缺省值	描述
已使用	真/假	假	指示地址是否正在程序中使用。
地址	%QW0.x0y	-	显示输出通道的地址， 其中 *x* 为扩展板编号，*y* 为通道编号
符号	-	-	双击并键入可选的符号，与要在程序中使用的相应模拟量输出对象关联。
类型	**0 - 5 V** **0 - 10 V** **0 - 20 mA** **4 - 20 mA**	**0 - 5 V**	通道模式。
范围	**自定义** **正常**	**自定义**	通道值的范围。
最小值	-32768...32767	4000	指定测量下限。
最大值	-32768...32767	20000	指定测量上限。
故障预置值	**最小值**...**最大值**	0（**最小值**，如果范围之中不包含 0）	指定输出通道的故障预置值。
单位		-	-
注释			双击并键入要与通道关联的可选注释。

TMCR2TI2

简介

TMCR2TI2 是指拥有 2 个 14 位精度的模拟量输入通道的标准扩展板。

通道输入类型为：

- 热电偶 K
- 热电偶 J
- 热电偶 R
- 热电偶 S
- 热电偶 B
- 热电偶 E
- 热电偶 T
- 热电偶 N
- 热电偶 C
- PT100
- PT1000
- NI100
- NI1000

有关详细的硬件信息，请参阅 TMCR2TI2 *(参见 Modicon M100/M200 可编程控制器, 硬件指南)*。

例如，如果已物理连接用于电压信号的模拟量通道，并在 SoMachine Basic 中为电流信号配置了通道，则可能会损坏模拟电路。

注意
设备无法操作 确认模拟电路的物理线路与模拟量通道的软件配置兼容。 **不遵循上述说明可能导致设备损坏。**

配置模块

对于每个输入，您可以定义：

参数	值	缺省值	描述
已使用	真/假	假	指示地址是否正在程序中使用。
地址	%IW0.x0y	-	输入通道的地址，其中 *x* 为模块编号，*y* 为通道编号
符号	-	-	双击并键入可选的符号，与要在程序中使用的相应模拟量输入对象关联。/P
类型	**热电偶 K** **热电偶 J** **热电偶 R** **热电偶 S** **热电偶 B** **热电偶 E** **热电偶 T** **热电偶 N** **热电偶 C** **PT100** **PT1000** **NI100** **NI1000**	**热电偶 K**	选择通道模式。
范围	**正常** **摄氏度 (0.1°C)** **华氏度 (0.1°F)** （热电偶 B 和 C 除外） **华氏度 (0.2°F)** （仅适用于热电偶 B 和 C）	**正常**	选择通道的温度单位。
最小值	请参见下表		指定测量下限。
最大值	请参见下表		指定测量上限。
滤波器	0...100	0	指定滤波值。乘以**滤波单位**的值以获得滤波时间。
滤波单位	100 毫秒	100 毫秒	设定滤波单位。
单位	请参见下表		显示配置的温度单位。
注释	-	-	双击并键入可选的符号，与要在程序中使用的相应模拟量输入对象关联。

类型	自定义		摄氏			华氏度		
	最小值	最大值.	最小值	最大值.	单位	最小值	最大值.	单位
热电偶 K	-32768	32767	-2000	13000	0.1 °C	-3280	23720	0.1 °F
热电偶 J	-32768	32767	-2000	10000	0.1 °C	-3280	18320	0.1 °F
热电偶 R	-32768	32767	0	17600	0.1 °C	320	32000	0.1 °F
热电偶 S	-32768	32767	0	17600	0.1 °C	320	32000	0.1 °F
热电偶 B	-32768	32767	0	18200	0.1 °C	160	16540	0.2 °F
热电偶 E	-32768	32767	-2000	8000	0.1 °C	-3280	14720	0.1 °F
热电偶 T	-32768	32767	-2000	4000	0.1 °C	-3280	7520	0.1 °F
热电偶 N	-32768	32767	-2000	13000	0.1 °C	-3280	23720	0.1 °F
热电偶 C	-32768	32767	0	23150	0.1 °C	160	20995	0.2 °F
PT100	-32768	32767	-2000	8500	0.1 °C	-3280	15620	0.1 °F
PT1000	-32768	32767	-2000	6000	0.1 °C	-3280	11120	0.1 °F
NI100	-32768	32767	-600	1800	0.1 °C	-760	3560	0.1 °F
NI1000	-32768	32767	-600	1800	0.1 °C	-760	3560	0.1 F

TMCR2SL1

简介

TMCR2SL1 是指具有 1 条串行线路的标准扩展板模块。

有关详细的硬件信息，请参阅 TMCR2SL1 *(参见 Modicon M100/M200 可编程控制器, 硬件指南)*。

串行线路可以配置为以下任一协议：

- Modbus RTU
- Modbus ASCII
- ASCII
- Modbus Serial IOScanner

您可以配置串行线路的物理设置和协议设置。缺省情况下，串行线路配置为 Modbus RTU 协议。

注意： 仅可添加一个串行线路扩展板到控制器。

串行线路配置

下表介绍了如何配置串行线路：

步骤	动作
1	单击硬件树中的 SL2（**串行线路**）节点，显示串行线路属性。 下图显示 **Modbus RTU** 和 **Modbus ASCII** 协议的串行线路属性：

步骤	动作
	下图显示了适用于 ASCII 协议的串行线路的属性： 串行线路配置 协议设置 协议 ASCII 串行线路设置 波特率 19200 校验位 偶数 数据位 8 停止位 1 物理介质 RS-485 RS-232 极化 否 应用 取消 ASCII 设备设置 设备 无 初始命令 协议设置 响应超时（x 100 毫秒） 10 停止条件 收到的帧长度 0 帧收到超时（毫秒） 0 帧结构 起始字符 0 第一个结束字符 10 <LF> 第二个结束字符 0 发送帧字符 应用 取消
2	编辑属性以配置串行线路。 有关串行线路配置参数的详细信息，请参阅下表。

下表介绍了串行线路的每个参数：

参数	可编辑	值	缺省值	描述
物理设置				
波特率	是	1200 2400 4800 9600 19200 38400 57600 115200	19200	用于从下拉列表中选择调制解调器的数据传输速率（每秒位数）。
奇偶校验	是	**无** **偶数** **奇数**	**偶数**	用于为错误检测选择对传输数据的奇偶校验。奇偶校验是传输中使用的错误检测方法。 对串行端口使用奇偶校验时，将随着每个数据字符发送额外的一个数据位，从而使每个字符中 1 的位数（包括奇偶校验位）始终为奇数或偶数。 如果字节以错误的 1 的位数接收， 则字节会损坏。但是，检测到的偶数错误数可以通过奇偶校验。
数据位	是 （仅针对 **ASCII** 协议）	**7** **8**	**7** 适用于 Modbus ASCII，**8** 适用于 Modbus RTU	用于从下拉列表中选择数据位数。 每个字符中的数据位个数可能是 7（对于真正的 ASCII）或 8（对于任何一种数据，这样也匹配字节大小）。几乎所有的应用程序中都普遍使用 8 个数据位。
停止位	是	**1** **2**	**1**	用于从下拉列表中选择停止位数。 停止位是指示数据字节结束的一个位。对于电子设备来说，通常使用 1 个停止位。对于像机电电传打字机这样较慢的设备来说， 将使用 2 个停止位。
物理介质	是	**RS-485** True/False **RS-232** True/False	**RS-485** True	用于选择通讯的物理介质。 您只能选择 **RS-485** 或 **RS-232** 介质。启用一个介质，禁用另一个。 数据通讯中的物理介质为传播信号使用的传输路径。它是设备与逻辑控制器互连的接口。
极化	是	**是** **否**	**否**	极化电阻器集成于扩展板模块中。规定是否打开或关闭极化。
协议设置				
协议	是	**Modbus RTU** **Modbus ASCII** **ASCII** **Modbus Serial IOScanner**	**Modbus RTU**	用于从下拉列表中选择通讯的协议传输模式。 根据选择的协议显示高级协议参数。 请参阅下图和下表。

参数	可编辑	值	缺省值	描述
Modbus RTU 和 Modbus ASCII 协议的协议设置：				
寻址	是	**从站** **主任务**	**从站**	用于选择寻址模式。您只能选择**从站**或**主站**寻址。启用一个寻址模式， 禁用另一个。
地址 [1...247]	是	1...247	1	用于指定从站的地址 ID。 **注意：** 仅为从站寻址显示此字段。 对于主站，不会在屏幕上显示此字段。
响应时间 **（× 100 毫秒）**	是	10...255 毫秒	10	用于为查询指定协议的响应时间。
帧间时间 **（毫秒）**	是	3...255 毫秒	10	用于指定协议的帧间时间。
ASCII 协议的协议设置：				
停止条件				
响应时间 **（× 100 毫秒）**	是	1...255	10	用于为查询指定协议的响应时间。
收到的帧长度	是	0...255	0	用于指定收到的帧长度。
帧收到超时 **（毫秒）**	是	0...255	10	用于指定帧收到的超时。
帧结构				
起始字符	是	0...255	58（如果已选中复选框）	用于指定帧的起始字符。
第一个结束字符	是	0...255	10（如果已选中复选框）	用于指定帧的第一个结束字符。
第二个结束字符	是	0...255	10（如果已选中复选框）	用于指定帧的第二个结束字符。
发送帧字符	是	True/False	False	用于对 ASCII 协议启用或禁用发送帧的第一个结束字符。

TMCR2SL1A

简介

TMCR2SL1A 是指具有 1 条隔离串行线路的标准扩展板模块。

有关详细的硬件信息，请参阅 TMCR2SL1A *(参见 Modicon M100/M200 可编程控制器, 硬件指南)*。

串行线路可以配置为以下任一协议：

- Modbus RTU
- Modbus ASCII
- ASCII
- Modbus Serial IOScanner

您可以配置串行线路的物理设置和协议设置。缺省情况下，串行线路配置为 Modbus RTU 协议。

注意： 仅可添加一个串行线路扩展板到控制器。

串行线路配置

下表介绍了如何配置串行线路：

步骤	动作
1	单击硬件树中的 **SL2（串行线路）**节点，显示串行线路属性。 下图显示 **Modbus RTU** 和 **Modbus ASCII** 协议的串行线路属性：

步骤	动作
	下图显示了适用于 ASCII 协议的串行线路的属性：
2	编辑属性以配置串行线路。 有关串行线路配置参数的详细信息，请参阅下表。

下表介绍了串行线路的每个参数：

参数	可编辑	值	缺省值	描述
物理设置				
波特率	是	1200 2400 4800 9600 19200 38400 57600 115200	19200	用于从下拉列表中选择调制解调器的数据传输速率（每秒位数）。
奇偶校验	是	**无** **偶数** **奇数**	**偶数**	用于为错误检测选择对传输数据的奇偶校验。奇偶校验是传输中使用的错误检测方法。 对串行端口使用奇偶校验时，将随着每个数据字符发送额外的一个数据位，从而使每个字符中 1 的位数（包括奇偶校验位）始终为奇数或偶数。 如果字节以错误的 1 的位数接收，则字节会损坏。但是，检测到的偶数错误数可以通过奇偶校验。
数据位	是 （仅针对 **ASCII** 协议）	7 8	7 适用于 Modbus ASCII，8 适用于 Modbus RTU	用于从下拉列表中选择数据位数。 每个字符中的数据位个数可能是 7（对于真正的 ASCII）或 8（对于任何一种数据，这样也匹配字节大小）。几乎所有的应用程序中都普遍使用 8 个数据位。
停止位	是	1 2	1	用于从下拉列表中选择停止位数。 停止位是指示数据字节结束的一个位。对于电子设备来说，通常使用 1 个停止位。对于像机电电传打字机这样较慢的设备来说，将使用 2 个停止位。
物理介质	是	**RS-485** True/False **RS-232** True/False	**RS-485** True	用于选择通讯的物理介质。 您只能选择 **RS-485** 或 **RS-232** 介质。启用一个介质，禁用另一个。 数据通讯中的物理介质为传播信号使用的传输路径。它是设备与逻辑控制器互连的接口。
极化	是	**是** **否**	**否**	极化电阻器集成于扩展板模块中。规定是否打开或关闭极化。
协议设置				
协议	是	**Modbus RTU** **Modbus ASCII** **ASCII** **Modbus Serial IOScanner**	**Modbus RTU**	用于从下拉列表中选择通讯的协议传输模式。根据选择的协议显示高级协议参数。请参阅下图和下表。

参数	可编辑	值	缺省值	描述
Modbus RTU 和 Modbus ASCII 协议的协议设置：				
寻址	是	**从站** **主任务**	**从站**	用于选择寻址模式。您只能选择**从站**或**主站**寻址。启用一个寻址模式，禁用另一个。
地址 [1...247]	是	1...247	1	用于指定从站的地址 ID。 **注意：** 仅为从站寻址显示此字段。 对于主站，不会在屏幕上显示此字段。
响应时间（× 100 毫秒）	是	10...255 毫秒	10	用于为查询指定协议的响应时间。
帧间时间（毫秒）	是	3...255 毫秒	10	用于指定协议的帧间时间。
ASCII 协议的协议设置：				
停止条件				
响应时间（× 100 毫秒）	是	1...255	10	用于为查询指定协议的响应时间。
收到的帧长度	是	0...255	0	用于指定收到的帧长度。
帧收到超时（毫秒）	是	0...255	10	用于指定帧收到的超时。
帧结构				
起始字符	是	0...255	58（如果已选中复选框）	用于指定帧的起始字符。
第一个结束字符	是	0...255	10（如果已选中复选框）	用于指定帧的第一个结束字符。
第二个结束字符	是	0...255	10（如果已选中复选框）	用于指定帧的第二个结束字符。
发送帧字符	是	True/False	False	用于对 ASCII 协议启用或禁用发送帧的第一个结束字符。

TMCR2模拟量扩展板诊断

简介

这些对象为模拟量扩展板提供了每个 I/O 通道的运行状态：

- 扩展板 x 上输入通道 y 的 %IWS0.x0y
- 扩展板 x 上输出通道 y 的 %QWS0.x0y

可使用动态数据表 *(参见 SoMachine Basic, 操作指南)*或应用程序在在线模式下读取这些对象的实时值。

输入通道状态说明

下表介绍 %IWS 输入通道状态字的可能值：

字节值	描述
0	正常
1	正在进行数据转换
2	初始化
3	输入操作设置错误或扩展板无输入
4	未定义
5	检测到接线错误（超过输入电压/电流上限）。
6	检测到接线错误（超过输入电压/电流下限）。
7	非易失性存储器错误
8...255	未定义

输出通道状态说明

下表介绍 %QWS 输出通道状态字的可能值：

字节值	描述
0	正常
1	未定义
2	初始化
3	输出操作设置错误或扩展板无输出
4	未定义
5	未定义
6	未定义
7	非易失性存储器错误
8...255	未定义

第7章
TM3R扩展模块配置

概述

本章介绍如何配置 M100/M200 可编程控制器 的 TM3R 扩展模块。

本章包含了哪些内容？

本章包含了以下主题：

主题	页
I/O 配置一般做法	136
配置 TM3R 数字量 I/O 模块	137
在配置中使用 I/O 模块	138
配置数字量 I/O	140
TM3 模拟量 I/O 模块诊断	143

I/O 配置一般做法

匹配硬件和软件配置

可在控制器中嵌入的 I/O 独立于采用 I/O 扩展的形式添加的 I/O。程序中的逻辑 I/O 配置应与安装的物理 I/O 配置匹配，这十分重要。如果对 I/O 扩展总线添加或删除任何物理 I/O，或根据控制器型号，对控制器进行添加或删除操作（以扩展板的形式），则必须更新应用程序配置。这也适用于安装中包含的任何现场总线设备。否则，I/O 扩展将无法再正常工作，而控制器中可能存在的内置 I/O 会继续运行。

⚠ 警告

意外的设备操作

每次添加或删除 I/O 总线上任何类型的 I/O 扩展，或添加或删除现场总线上的任何设备时，都需更新程序配置。

不遵循上述说明可能导致人员伤亡或设备损坏。

配置 TM3R 数字量 I/O 模块

简介

TM3R 数字量 I/O 扩展模块的范围包括：

- TM3R 数字量混合输入/输出模块 *(参见 Modicon TM3 (SoMachine Basic), 扩展模块配置, 编程指南)*

配置模块

配置选项卡：在“配置”选项卡中显示配置详细信息 (参见第 *141* 页)介绍如何查看这些模块的配置。

编程选项卡：在“编程”选项卡中显示配置详细信息 (参见第 *142* 页)介绍如何查看和更新这些模块的编程相关属性。

在配置中使用 I/O 模块

添加模块

以下步骤介绍如何将 TM3R 扩展模块添加到 SoMachine Basic 项目中的逻辑控制器：

步骤	动作
1	单击 SoMachine Basic 窗口中的**配置**选项卡。
2	在目录区域中，单击以下模块类型之一以展开扩展模块列表： ● **TM3 数字量 I/O 模块**
3	从列表中选择要添加的 TM3R 扩展模块。 **结果**：所选扩展模块的物理特性描述会出现在目录区域底部。
4	将所选扩展模块拖到编辑器区域，并将此模块放到配置中的控制器或最后一个扩展模块的右侧。 **结果**：此模块已添加到硬件树的**我的控制器 → I/O 总线**分支下，并且所选模块的物理特性描述会出现在编辑器区域底部。

在两个现有模块之间插入模块

在两个模块之间或控制器与第一个模块之间拖动模块，直到出现绿色垂直条，然后放下此模块。

注意： 在您通过插入新模块来更改模块位置时，地址会发生变化。例如，如果您将一个输入模块从位置 4 移动到位置 2，则地址会从 I4. x 变为 I2. x，且程序中的所有相应地址都会自动重命名。

可在控制器中嵌入的 I/O 独立于采用 I/O 扩展的形式添加的 I/O。程序中的逻辑 I/O 配置应与安装的物理 I/O 配置匹配，这十分重要。如果对 I/O 扩展总线添加或删除任何物理 I/O，或根据控制器型号，对控制器进行添加或删除操作（以扩展板的形式），则必须更新应用程序配置。这也适用于安装中包含的任何现场总线设备。否则，扩展总线或现场总线可能不再正常工作，而控制器中可能存在的嵌入式 I/O 会继续操作。

⚠ 警告

意外的设备操作

每次添加或删除 I/O 总线上任何类型的 I/O 扩展，或添加或删除现场总线上的任何设备时，都需更新程序配置。

不遵循上述说明可能导致人员伤亡或设备损坏。

替换现有扩展模块

通过拖动新模块并将其放到要更换的模块上，可将现有模块更换为新模块。

此时会显示一条消息，请您确认操作。单击**是**以继续。

删除模块

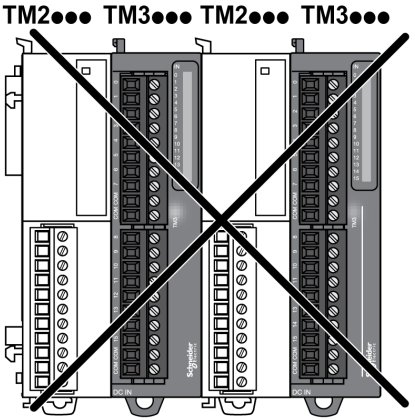
可以通过按下**删除**键或通过右键单击模块并在出现的上下文菜单中单击**删除**，从而删除扩展模块。

如果扩展模块至少包含一个程序中正在使用的地址，则会显示一条消息，请您确认操作。单击**是**以继续。

混合扩展模块类型

您可以在同一个逻辑控制器上混合不同的 I/O 模块类型（例如，TM3R、TM3 和 TM2 模块）。

将任何 TM2 模块放在您的配置的末尾（在任何 TM3 模块右面）：



在这种情况下，逻辑控制器的 I/O 总线会以较慢模块类型的速度运行。例如，同时使用 TM2 和 TM3 模块时，逻辑控制器的 I/O 总线会以 TM2 模块的速度运行。

最大硬件配置

SoMachine Basic 会在以下情况下显示一条消息：

- 超过逻辑控制器支持的最大模块数。
- 直接连接到逻辑控制器的所有扩展模块的总功耗超过逻辑控制器所提供的最大电流。

有关支持的最高配置的详细信息，请参阅控制器的硬件指南。

配置数字量 I/O

概述

您可以使用以下选项卡配置扩展模块的数字量 I/O：

- **配置**选项卡：
  - 数字量输入 *(参见 Modicon TM3 (SoMachine Basic), 扩展模块配置, 编程指南)*
  - 数字量输出 *(参见 Modicon TM3 (SoMachine Basic), 扩展模块配置, 编程指南)*
- **编程**选项卡 *(参见 Modicon TM3 (SoMachine Basic), 扩展模块配置, 编程指南)*。

在配置选项卡中配置数字量输入

按照以下步骤在**配置**选项卡中显示和配置数字量输入属性：

步骤	描述
1	单击 SoMachine Basic 窗口中的**配置**选项卡。
2	在硬件树中，单击**我的控制器 → IO 总线 → 模块 x → 数字量输入**，其中 x 是控制器上的扩展模块编号。 **结果**：所选模块的数字量输入属性会显示在编辑器区域中，例如： 数字量输入 已使用 | 地址 | 符号 | 注释 %I4.0 %I4.1 %I4.2
3	编辑属性以配置数字量输入： ● **已使用**：指示是否正在程序中使用相应地址。 ● **地址**：显示数字量输入在扩展模块上的地址。有关 I/O 对象寻址的详细信息，请参阅 I/O 寻址 *(参见 SoMachine Basic, 通用功能库指南)*。 ● **符号**：可用于指定程序中使用的与相应数字量输入对象关联的符号。在**符号**列中双击，键入相应对象的符号名称，然后按 Enter 键。 ● **注释**：可用于指定与相应数字量输入对象关联的注释。在**注释**列中双击，键入相应对象的注释，然后按 Enter 键。
4	单击**应用**保存更改。

在配置选项卡中配置数字量输出

按照以下步骤在**配置**选项卡中显示和配置数字量输出属性：

步骤	描述
1	单击 SoMachine Basic 窗口中的**配置**选项卡。
2	在硬件树中，单击**我的控制器 → IO 总线 → 模块 x → 数字量输出**，其中 x 是控制器上的扩展模块编号。 **结果**：所选模块的数字量输出属性会显示在编辑器区域中，例如： 数字量输出 已使用 | 地址 | 符号 | 故障预置值 | 注释 %Q3.0 | 0 %Q3.1 | 1 %Q3.2 | 0
3	编辑属性以配置数字量输出： ● **已使用**：指示是否正在程序中使用相应地址。 ● **地址**：显示数字量输出在扩展模块上的地址。有关 I/O 对象寻址的详细信息，请参阅 I/O 寻址 *(参见 SoMachine Basic, 通用功能库指南)*。 ● **符号**：可用于指定程序中使用的与相应数字量输出对象关联的符号。在**符号**列中双击，键入相应对象的符号名称，然后按 **Enter** 键。 ● **故障预置值**。可用于指定该值，以便在 Logic Controller 进入 STOPPED 或异常状态时应用到相应输出（故障预置到 0 或故障预置到 1）。缺省值是 0。如果配置了**维护值**故障预置模式，则在逻辑控制器进入STOPPED或异常状态时，输出仍保留其当前值。有关维护输出值的更多详细信息，请参阅故障预置行为。 ● **注释**：可用于指定与相应数字量输出对象关联的注释。在**注释**列中双击，键入相应对象的注释，然后按 **Enter** 键。
4	单击**应用**保存更改。

在编程选项卡中显示配置详细信息

编程选项卡会显示所有输入/输出的配置详细信息，并允许您更新符号和注释等与编程相关的属性。

按照以下步骤在**编程**选项卡中查看和更新 I/O 模块的详细信息：

步骤	描述
1	单击 SoMachine Basic 窗口中的**编程**选项卡。
2	在**编程**选项卡的左侧区域中，单击**工具**选项卡，然后从 **I/O 对象**分支中，选择以下 I/O 类型之一，以显示属性： ● **数字量输入** ● **数字量输出** ● **模拟量输入** ● **模拟量输出** **结果**：所有嵌入式和扩展模块 I/O 地址的列表会显示在 SoMachine Basic 窗口的中下方区域中，例如： 数字量输出属性 已使用 | 地址 | 符号 | 注释 %Q.6 %Q0.7 %Q1.0 CH1 控制方向 1 %Q1.1 CH1 控制方向 2 %Q1.2
3	向下滚动到与您配置的扩展模块对应的地址范围：随即会显示以下属性： ● **已使用**：指示是否正在程序中使用相应地址。 ● **地址**：显示数字量输出在扩展模块上的地址。有关 I/O 对象寻址的详细信息，请参阅 I/O 寻址 *(参见 SoMachine Basic, 通用功能库指南)*。 ● **符号**：可用于指定程序中使用的与相应 I/O 对象关联的符号。 在**符号**列中双击，键入相应对象的符号名称，然后按 **Enter** 键。 如果某个符号已经存在，请右键单击**符号**列，然后选择**搜索并替换**，在整个程序和/或程序注释中查找并替换出现此符号的地方。 ● **注释**：可用于指定与相应 I/O 对象关联的注释。 在**注释**列中双击，键入相应对象的注释，然后按 **Enter** 键。
4	单击**应用**保存更改。

TM3 模拟量 I/O 模块诊断

简介

这些对象提供了每个 I/O 通道的运行状态。

- 模块 x 输入通道 y 的 %IWSx.y
- 模块 x 输出通道 y 的 %QWSx.y

输入通道状态字节描述

下表介绍 %IWS 输入通道状态字节：

字节值	描述
0	正常
1	未定义
2	未定义
3	检测到配置错误。
4	检测到外部电源错误。
5	检测到接线错误（超过输入电压/电流上限）。
6	检测到接线错误（超过输入电压/电流下限）。
7	检测到硬件错误.
8...255	未定义

输出通道状态字节描述

下表介绍 %QWS 输出通道状态字节：

字节值	描述
0	正常
1	未定义
2	未定义
3	检测到配置错误
4	超过外部电源电压限制
5	未定义
6	未定义
7	检测到硬件错误
8...255	未定义

第8章
内置通讯配置

概述

本章介绍如何配置 M100/M200 可编程控制器 的通讯功能。

本章包含了哪些内容？

本章包含了以下部分：

节	主题	页
8.1	以太网配置	146
8.2	串行线路配置	163
8.3	支持的 Modbus 功能代码	178

第8.1节
以太网配置

本节包含了哪些内容？

本节包含了以下主题：

主题	页
配置以太网网络	147
配置 Modbus TCP	153

配置以太网网络

简介

可通过配置以太网网络来配置与 Logic Controller 的 TCP/IP 连接。以太网在逻辑控制器和其他设备之间建立局域网 (LAN)。以太网配置让您能够配置网络设备的 IP 地址。

注意： 控制器与 PC 的链接使用 TCP/IP 协议。必须在 PC 上安装此协议。

您可以通过以下协议获取 IP 地址：
- 动态主机配置协议 (DHCP)
- 引导程序协议 (BOOTP)

您也可以通过指定以下地址来指定 IP 地址：
- IP 地址
- 子网掩码
- 网关地址

注意： Schneider Electric 在控制系统的开发和实施过程中严格遵循行业最佳实践。这其中包括一种“深度防御”方法，旨在保护工业控制系统的安全。此方法将控制器置于一个或多个防火墙之后，将访问范围限制为仅经过授权的人员和协议。

⚠ 警告

未经授权访问及其导致的未经授权的机器操作

- 评估环境或机器是否已连接到关键基础结构，如果已连接，请在将自动化系统连接到任何网络之前，基于深度防护采取适当的预防措施。
- 将连接到网络的设备数限制为所需的最小数量。
- 将工业网络与公司内部的其他网络隔离。
- 使用防火墙、VPN 或其他经证实的安全措施，防止意外访问任何网络。
- 监控系统内的活动。
- 防止未经授权方或未经身份验证的操作直接访问或直接链接主体设备。
- 准备恢复计划，包括系统和过程信息的备份。

不遵循上述说明可能导致人员伤亡或设备损坏。

以太网服务

逻辑控制器支持以下服务：
- Modbus TCP 服务器 *(参见 Modicon M258 Logic Controller, 编程指南)*
- Modbus TCP 客户端 *(参见 Modicon M258 Logic Controller, 编程指南)*
- Modbus TCP 从站设备

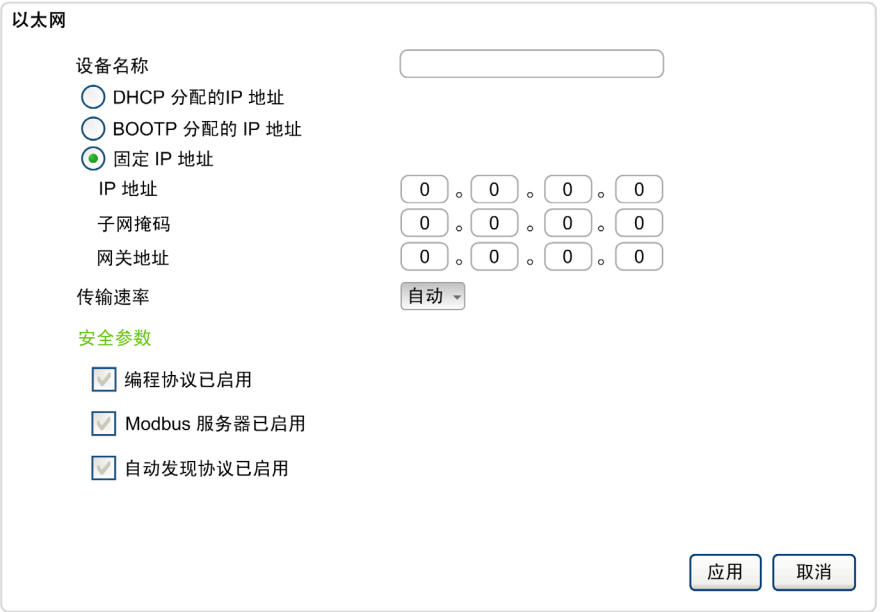
下表列出了 TCP 服务器的最大连接数：

连接类型	最大连接数
服务器	8
客户端	1

每个基于 TCP 的服务器都管理其各自的一组连接。当客户端尝试打开超出轮询大小的连接时，逻辑控制器会关闭最旧的连接，而不是 SoMachine Basic 的连接。只要逻辑控制器继续处于其当前工作状态（正在运行、已停止或已暂停），则服务器连接就会继续处于开启状态。如果逻辑控制器从当前的工作状态（正在运行、已停止或已暂停）发生转变，则服务器连接将会关闭，但断电时除外（因为控制器来不及关闭连接）。

以太网配置

下表介绍了如何配置以太网：

步骤	动作
1	单击硬件树中的 **ETH1** 节点，以在编辑器区域中显示以太网属性。 
2	编辑属性以配置以太网。 有关以太网配置参数的详细信息，请参阅下表。

注意： 显示的安全参数取决于为应用程序选择的功能等级 *(参见 SoMachine Basic, 操作指南)*。

下表介绍了以太网配置的每个参数：

参数	可编辑	值	缺省值	描述
以太网				
设备名称	否	*任何*	**M200**（如果配置中使用的控制器是 M100/M200 可编程控制器）	显示与以太网网络连接的设备的名称。
DHCP 协议分配 IP 地址	是(1)	True/False	False	用于从网络上的 DHCP 服务器获取 IP 地址。
BOOTP 协议分配 IP 地址	是(1)	True/False	False	用于从网络上的 Boot PROM 配置服务器中获取 IP 地址。
固定 IP 地址	是(1)	True/False	True	用于手动指定主机或网络接口标识的 IP 地址。
IP 地址	是(2)	*w.x.y.z*(3)	0.0.0.0	用于指定以太网网络中设备的 IP 地址。 请参见地址类别 (参见第 *152* 页) 分配 0.0.0.0（缺省值）作为 M100/M200 可编程控制器 的 IP 地址将强制固件从 MAC 地址生成 IP 地址。 生成的 IP 地址是 10.10.XXX.YYY，其中 XXX 和 YYY 是 MAC 地址 (AA.BB.CC.DD.EE.FF) 最后两个字节 (EE.FF) 的十进制值 示例： MAC 地址：00:80:78:19:19:73 EE（十六进制 19）= **25**（十进制） FF（十六进制 73）= **155**（十进制） 生成的 IP 地址：10.10.**25.155**。 固件也通过 MAC 地址生成 IP 地址，条件是指定的 IP 地址标识为网络上的重复地址。 当检测到重复的 IP 地址时，系统字 %SW118 的位 9 设置为 1（请参见系统字说明 (参见第 *464* 页)）并且系统字 %SW62 设置为 1（请参见系统字说明 (参见第 *464* 页)）。 逻辑控制器的 MAC 地址存储在 %SW107-%SW109 中（请参见系统字说明 (参见第 *464* 页)）。
子网掩码	是(2)	*w.x.y.z*(3)	0.0.0.0	使您可以指定子网络的地址以授权可进行数据交换的一组设备。它确定 IP 地址中的哪些位对应于网络地址，哪些位对应于地址的子网部分。请参见子网掩码 (参见第 *152* 页).

(1) 您可以选择任何 1 个选项进行 IP 寻址。选择任何 1 个选项，禁用其他选项。
(2) 只有选择**固定 IP 地址**选项进行 IP 寻址时才启用这些选项。
(3) *w*、*x*、*y* 和 *z* 是用于存储地址的字节，每个字节可以存储 0 到 255 范围内的值。

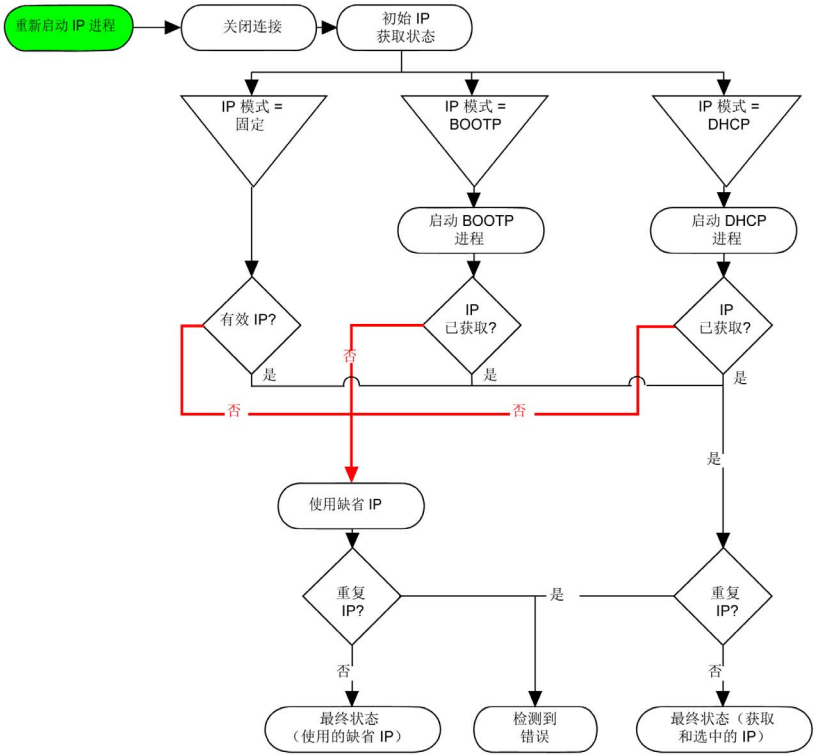
参数	可编辑	值	缺省值	描述
网关地址	是(2)	*w.x.y.z*(3)	0.0.0.0	用于指定 TCP/IP 网络上被用作另一网络访问点的节点（路由器）IP 地址。 请参见网关地址 (参见第 *152* 页).
传输速率	否	–	**自动**	显示选定的以太网速度模式。 自动代表“自动协调"。
安全参数				
编程协议已启用	是	True/False	True	使您可以启用或禁用编程协议 （用于与网络中的其他设备通讯）。
启用自动发现协议	是	True/False	True	使您可以启用或禁用自动发现协议 （用于自动检测网络中的设备）。
Modbus 服务器已启用	是	True/False	True	使您可以为串行设备连接启用或禁用 Modbus 服务器。

(1) 您可以选择任何 1 个选项进行 IP 寻址。选择任何 1 个选项，禁用其他选项。
(2) 只有选择**固定 IP 地址**选项进行 IP 寻址时才启用这些选项。
(3) *w*、*x*、*y* 和 *z* 是用于存储地址的字节，每个字节可以存储 0 到 255 范围内的值。

注意： 当列出在 **安全参数** 中的协议被禁用时，来自相应服务器类型的请求将被忽略。相应的配置屏幕仍可访问；但是，程序执行并不会受到影响。

地址管理

此示意图显示 M100/M200 可编程控制器 的不同类型的地址系统：



注意： 如果编程为使用 DHCP 或 BOOTP 寻址方法的设备无法联系其相应的服务器，则控制器使用缺省 IP 地址。但是，它将不断地重复此请求。

在以下情况中，IP 进程会重新启动：

- 控制器重新启动
- 以太网电缆重新连接
- 下载应用程序（如果 IP 参数发生更改）
- 在上一个寻址尝试不成败或 DHCP 地址租赁到期后检测到了 DHCP 或 BOOTP 服务器。

地址类别

IP 地址与以下两项相关联：

- 设备（主机）
- 该设备所连接的网络

IP 地址始终采用 4 字节进行编码。

这些字节在网络地址和设备地址之间的分配可能会发生变化。具体分配方法由地址类别定义。

下表定义了不同的 IP 地址类别：

<table>
<tr><th>地址类别</th><th colspan="5">字节 1</th><th>字节 2</th><th>字节 3</th><th>字节 4</th></tr>
<tr><td>A 类</td><td>0</td><td colspan="4">网络 ID</td><td colspan="3">主机 ID</td></tr>
<tr><td>B 类</td><td>1</td><td>0</td><td colspan="3">网络 ID</td><td colspan="2">主机 ID</td></tr>
<tr><td>C 类</td><td>1</td><td>1</td><td>0</td><td colspan="4">网络 ID</td><td>主机 ID</td></tr>
<tr><td>D 类</td><td>1</td><td>1</td><td>1</td><td>0</td><td colspan="4">多播地址</td></tr>
<tr><td>E 类</td><td>1</td><td>1</td><td>1</td><td>1</td><td>0</td><td colspan="3">保留地址，供以后使用</td></tr>
</table>

子网掩码

子网掩码的作用是在同一个网络地址下为多个物理网络编址。掩码用于划分子网地址和主机 ID 的设备地址。

获取子网地址的方法是：保留 IP 地址中与包含 1 的掩码的位置相对应的位，然后用 0 替换其他位。

反之，获取主机设备子网地址的方法是：保留 IP 地址中与包含 0 的掩码的位置相对应的位，然后用 1 替换其他位。

子网地址的示例：

IP 地址	192 (11000000)	1 (00000001)	17 (00010001)	11 (00001011)
子网掩码	255 (11111111)	255 (11111111)	240 (11110000)	0 (00000000)
子网地址	192 (11000000)	1 (00000001)	16 (00010000)	0 (00000000)

注意： 如果没有网关，设备不在其子网中进行通讯。

网关地址

网关可将消息路由到不在当前网络中的设备。

如果没有网关，则网关地址为 0.0.0.0。

配置 Modbus TCP

简介

您可以将 Modbus TCP 或 Modbus TCP IOScanner 的 Ethernet 端口配置为：

- Modbus 映射 (参见第 *153* 页)
- 客户端模式 (参见第 *155* 页)

只能定义 IOScanner 的一个实例：如果您将其配置在串行端口上，则无法将其配置在 Ethernet 端口上，反之亦然。请参阅配置 Modbus Serial IOScanner (参见第 *170* 页)。

TCP 和 Serial IOScanner 对象的最大数量为：

- 128，如果**功能级别 < 6.0**。
- 512，如果**功能级别 ≥ 6.0**。

配置 Modbus TCP：Modbus 映射

下表介绍了如何配置 Modbus 映射：

步骤	动作
1	在**配置**窗口中，单击 **ETH1→ Modbus TCP**，显示 Modbus TCP 属性。 下图显示了编辑器区域中所显示的属性： Modbus TCP Modbus 映射 ☑ 已启用　单元 ID 247　输出寄存器 (%IWM) 10　输入寄存器 (%QWM) 10
2	选择**启用**可编辑用于配置 **Modbus 映射**的属性。 **注意：** 如果**已启用**按钮变灰，核实应用程序的**功能级别**（**编程 → 任务 → 行为**选项卡）至少为 **3.2 级**。
3	单击**应用**。

下表介绍了 **Modbus 映射**配置的每个参数：

参数	可编辑[1]	值	缺省值	描述
已启用	是	TRUE/FALSE	FALSE	选择后，会启用 **Modbus 映射**。 **注意：** 如果您取消选中**启用** 复选框，并且您已在程序中使用了网络变量，则它们将不再有效，程序也无法再编译。如果您想要暂时禁用 Modbus TCP/IP 服务但同时又不使其网络变量的使用无效，您可以在以太网属性窗口 (参见第 *151* 页)中停用该协议的**安全参数**。

[1]只有当 **Modbus 服务器已启用**选项在 Ethernet 属性窗口 (参见第 *151* 页)的**安全参数**部分中被选中时。

参数	可编辑[1]	值	缺省值	描述
单元 ID	是	1...247	-	指定远程服务器的单元 ID。 来自某一个设备并具有相同单元 ID 的 Modbus TCP 请求会被发送至 Modbus 映射表中而非常规 Modbus 服务器中。
输出寄存器 (%IWM)	是	1...20	10	可用输出寄存器的数量。 输出寄存器用于存储 Modbus TCP (%IWM) 对象 的值。
输入寄存器 (%QWM)	是	1...20	10	可用输入寄存器的数量。 输入寄存器用于存储 Modbus TCP (%QWM) 对象 的值。
[1]只有当 **Modbus 服务器已启用**选项在 Ethernet 属性窗口 (参见第 *151* 页)的**安全参数**部分中被选中时。				

Modbus TCP 从站设备 I/O 映射表

在配置 Modbus TCP 从站设备后，发送到其单元 ID（Modbus 地址）的 Modbus 命令便会访问控制器的网络对象（%IWM 和 %QWM），而不是当单元 ID 为 255 时被访问的常规 Modbus 字。这将有利于 Modbus 主站 I/O 扫描器应用程序进行读/写操作。

如果在主站中选择的单元 ID 不是在 M100/M200 从站中配置的单元 ID（或者相反），数据读取或写入常规 Modbus 字 %MWx，而不是网络对象 %IWMx 和 %QWMx。无 Modbus 错误返回。

以与访问常规 Modbus 字 (%MW) 相同的优先级访问 Modbus TCP 从站 I/O 映射表 (%IWM/%QWM)。

Modbus TCP 从站设备响应 Modbus 功能代码的子集，但响应方式与 Modbus 标准不同，并且其目的是与外部 I/O 扫描器交换数据。Modbus TCP 从站设备支持以下 Modbus 功能代码：

功能代码十进制（十六进制）	功能	注释
3（十六进制 3）	读取输出寄存器	允许主站 I/O 扫描器读取设备的网络对象 %QWM
4（十六进制 4）	读取输入寄存器	允许主站 I/O 扫描器读取设备的网络对象 %IWM
6（十六进制 6）	写入单个寄存器	允许主站 I/O 扫描器写入设备的单个网络对象 %IWM
16（十六进制 10）	写入多个寄存器	允许主站 I/O 扫描器写入设备的多个网络对象 %IWM
23（十六进制 17）	读取/ 写入多个寄存器	允许主站 I/O 扫描器读取网络对象 %QWM 并写入设备的网络对象 %IWM

配置 Modbus TCP：客户端模式

下表介绍了如何配置客户端模式：

步骤	动作
1	在**配置**窗口中，单击 **ETH1→ Modbus TCP**，显示 Modbus TCP 属性。 下图显示了编辑器区域中所显示的属性： **Modbus TCP** 客户端模式：远程服务器表（最大 16 个） 地址 0 . 0 . 0 . 0 添加 单元 ID 255 连接超时（100 毫秒） 100
2	添加远程设备。请参阅添加远程设备 (参见第 *155* 页)。
3	如要配置 Modbus TCP IOScanner，请选择**启用 Modbus TCP IOScanner**。 **注意：** 如果**启用 Modbus TCP IOScanner** 按钮灰显，则确认应用程序的**功能级别**（**编程 → 任务 → 行为**选项卡）至少为**6.0 级**，并且**串行线路 → Modbus Serial IOScanner**中未配置实例。 即使启用了 Modbus TCP IOScanner，您也可以为 Modbus TCP 配置和添加远程设备。

添加远程设备

下表介绍用于添加设备的**客户端模式：远程设备表（最大 16）**的参数：

参数	可编辑(1)	值	缺省值	描述
IP 地址	是	*w.x.y.z*(2)	–	可用于指定要读取的设备的 IP 地址。
通用 **驱动器** **预定义**	是	选择	通用	让您能够选择要添加的设备的类型。 如果启用了 Modbus TCP IOScanner，则**驱动器**和**预定义**选项可用。

(1)只有当 **Modbus 服务器已启用**选项在以太网属性窗口 的**安全参数**部分中被选中时。
(2) *w*、*x*、*y* 和 *z* 是用于存储地址的字节，每个字节可以存储介于一定范围内的值。

下表介绍如何添加远程设备：

步骤	动作
1	在 **IP 地址**字段中输入 IP 地址。
2	选择**通用**、**驱动器**或**预定义**。 只有在选择了**启用 Modbus TCP IOScanner** 的情况下，**驱动器**和**预定义**才会启用。
3	单击**添加**按钮。 在以下情况下，**添加**按钮被禁用： ● 配置的设备数已达到上限（16 个设备）。 ● IP 地址格式不正确。 **结果**：屏幕上随即显示您已添加的远程设备列表。 索引 | 地址 | 单元 ID | 连接超时（100 毫秒） 1 | 192.165.110.156 | 255 | 100
4	单击**应用**。

下表对列出远程设备的表的每列加以说明：

参数	可编辑	值	缺省值	描述
ID	否	0...15	**0**	SoMachine Basic 分配的唯一设备标识符。
名称	是	1...32 个字符 设备名称必须具有唯一性。	**设备 x** (1)	设备的名称。
地址	否	– %DRVn(2)	– %DRVn	%DRVn 用于在使用 驱动器功能块 (参见第 *262* 页) 的应用程序中配置设备。
类型	否	设备的类型	–	如要修改设备类型，必须从列表中移除该设备 （具体方式为：右键单击并选择**删除**）， 然后添加正确的设备类型。
索引	否	1...16	–	远程连接的设备的索引号。
IP 地址	是	*w.x.y.z*(2)	–	用于识别网络中的设备的地址。 从站地址允许重复。

(1) *w*、*x*、*y* 和 *z* 是用于存储地址的字节，每个字节可以存储 0 到 255 范围内的值。
(2) *x* 和 *n* 是整数，每次添加了设备或驱动设备后，计数相应递增。
(3) 在以下情况下启用：在**串行线路** 节点 →**协议设置**中未配置 Modbus Serial IOScanner。

参数	可编辑	值	缺省值	描述
响应超时（x 100 毫秒）	是	0...65535	10	连接超时持续时间。 即一段时间（以 100 毫秒为单位），控制器在此时段内会尝试与远程设备建立 TCP 连接。在此时段结束时，如果 TCP 连接仍然没有建立，则控制器会停止连接尝试，直到下一个带有 EXCH 指令的连接请求出现。
复位变量	是	%Mn	–	指定要用来复位设备的存储器位的地址 （重新发送初始化请求）。 在指定的存储器位被应用程序设置为 1 时，设备复位。
已扫描	否	TRUE/FALSE	TRUE	用于查看为 Modbus TCP IOScanner 配置了哪个设备。
初始请求单元 ID	是	0...255	255	指定本地设备的单元 ID。 来自某一个设备并具有相同单元 ID 的 Modbus TCP 请求会被发送至 Modbus 映射表中而非常规 Modbus 服务器中。
初始化请求 (3)	是	...	–	单击以显示初始化请求助手窗口 (参见第 *158* 页)。
通道单元 ID	是	0...255	255	指定本地设备的单元 ID。 来自某一个设备并具有相同单元 ID 的 Modbus TCP 请求会被发送至 Modbus 映射表中而非常规 Modbus 服务器中。
通道 (3)	是	...	–	单击以显示通道助手窗口 (参见第 *160* 页)。

(1) *w*、*x*、*y* 和 *z* 是用于存储地址的字节，每个字节可以存储 0 到 255 范围内的值。
(2) *x* 和 *n* 是整数，每次添加了设备或驱动设备后，计数相应递增。
(3) 在以下情况下启用：在**串行线路** 节点 →**协议设置**中未配置 **Modbus Serial IOScanner**。

配置初始化请求

初始化请求是 Modbus TCP IOScanner 或 Modbus Serial IOScanner 为了初始化从站设备而发送的设备特定的命令。在设备确认完所有初始化请求之前，Modbus TCP IOScanner 或 Modbus Serial IOScanner 不启动与设备的循环数据交换。初始化期间，不更新网络对象。

最多可以为每个从站设备定义 20 个初始化请求。

初始化请求助手窗口会显示定义的初始化请求：

ID	消息类型	偏移	长度	初始化值	注释
0	Mbs 0x06 - 写入单个字（寄存器）	8501	1	0	将 ATV 切换到 NST 状态
1	Mbs 0x06 - 写入单个字（寄存器）	12701	1	3201	ETA 寄存器配置
2	Mbs 0x06 - 写入单个字（寄存器）	12702	1	8604	RFRD 寄存器配置 (RPM)
3	Mbs 0x06 - 写入单个字（寄存器）	12703	1	3206	ETI 寄存器配置
4	Mbs 0x06 - 写入单个字（寄存器）	12704	1	7200	DP0 寄存器配置
5	Mbs 0x06 - 写入单个字（寄存器）	12721	1	8501	CMD 寄存器配置
6	Mbs 0x06 - 写入单个字（寄存器）	12722	1	8602	LFRD 寄存器配置 (RPM)

预配置的初始化请求显示有锁符号 和灰色背景。预定义初始化请求的一些参数无法修改。

根据所选择的设备类型，可以配置某些初始化请求。

下表描述了初始化请求的属性：

参数	可编辑	值	缺省值	描述
ID	否	0...19	0	唯一的初始化请求标识符。
消息类型	如果未预定义初始化请求，则为是。	请参见支持的 Modbus 功能代码 (参见第 *178* 页)	Mbs 0x05 - 写入单个位（线圈）	针对此初始化请求要使用的交换类型选择 Modbus 功能代码。 **注意：** 如果配置不支持缺省 Mbs 0x05 - Write single bit (coil) 请求类型的一般设备，则必须将缺省值替换为所支持的请求类型。
偏移	如果未预定义初始化请求，则为是。	0...65535	0	要初始化的首个寄存器的偏移。
长度	如果未预定义初始化请求，则为是。	1（对于 Mbs 0x05 - 写入单个位（线圈）） 1（对于 Mbs 0x06 - 写入单个字（寄存器）） 128（对于 Mbs 0x0F - 写入多个位（线圈）） 123（对于 Mbs 0x10 - 写入多个字（寄存器））	1	要初始化的对象数量（存储器字或位）。例如，如果以**偏移** = 2 且**长度** = 3 来写入多个字，则会初始化%MW2、%MW3 和 %MW4。
初始化值	如果未预定义初始化请求，则为是。	0...65535（如果正初始化存储器字（寄存器）） 0...1（如果正初始化存储器位（线圈））	0	用于初始化目标寄存器的值。
注释	如果未预定义初始化请求，则为是。	-	空	可以选择输入与该请求关联的注释。

单击**添加**，创建新的初始化请求。

选择一个条目，然后使用上、下箭头按钮改变初始化请求发送至设备的顺序。

在定义了初始化请求后，单击**确定**，保存配置，然后关闭**初始化请求助手**。

通道助手

最多可以为每个从站设备定义 10 个通道。每个通道代表一个 Modbus 请求。

注意： 在单击属性窗口上的**应用**后，所定义的对象数（读取和写入的数据项）将生效。

通道助手窗口列出了已定义的通道：

预配置的通道显示有锁符号 和灰色背景。预定义通道的一些参数无法修改。

下表描述了通道的属性：

参数	可编辑	值	缺省值	描述
ID	否	0...19， 最多10个通道	0	唯一的初始化标识符。
名称	是	0...32 个字符	Device_channel0	双击以编辑通道名称。
配置	是	...	-	单击以显示通道助手窗口 (参见第 *160* 页)。
消息类型	否	-	-	通道助手窗口 (参见第 *160* 页)中选择的 Modbus 功能代码。
触发器	否	-	-	通道助手窗口 (参见第 *160* 页)上选择的触发类型和循环时间。
R Offset	否	-	-	通道助手窗口 (参见第 *160* 页)中选择的读取对象偏移。

参数	可编辑	值	缺省值	描述
R Length	否	-	-	通道助手窗口 (参见第 *160* 页)中选择的读取对象长度。
错误管理	否	-	-	通道助手窗口 (参见第 *160* 页)中选择的错误管理策略。
W Offset	否	-	-	通道助手窗口 (参见第 *160* 页)中选择的写入对象偏移。
W Length	否	-	-	通道助手窗口 (参见第 *160* 页)中选择的写入对象长度。
注释	是	-	空	可以选择输入与该通道关联的注释。

单击**添加**，创建新的通道。

在定义了通道后，单击**确定**，保存配置，然后关闭**通道助手**。

配置通道

使用**通道助手**窗口配置通道。

以下示例示出了为“读/写多个字”请求配置的通道（Modbus 功能代码 23）。它以偏移 16#0C21 从寄存器读取一个字，并且以偏移 16#0C20 向寄存器写入两个字。在定义的**触发器**存在上升沿时，执行该请求（参见下表）：

下表描述了通道的属性：

参数	可编辑	值	缺省值	描述
名称	是	0...32 个字符	Device 0_ Channel0	输入通道的名称。
消息类型	是	请参见支持的 Modbus 功能代码 (参见第 *178* 页)	Mbs 0x17 - 读取/写入多个字（寄存器）	针对此通道要使用的交换类型选择 Modbus 功能代码。
触发器	是	**循环** **上升沿**	**循环**	为数据交换选择触发类型： ● **循环**：以**循环时间（x 10 毫秒）**字段中定义的频率来触发请求 ● **上升沿**：检测到存储器位的上升沿时，触发请求。指定要使用的**存储器位**的地址。
循环时间（× 10 毫秒） （如果选择的是**循环**）	是	1...6000	20	指定循环触发循环时间，以 10 毫秒为单位。
存储器位 （如果选择的是**上升沿**）	是	%Mn	-	指定存储器位地址，比如 %M8。在检测到该存储器位的上升沿时，触发数据交换。
注释	是	-	空	可以选择输入描述通道用途的注释。
读取对象				
偏移	是	0...65535	0	要读取的首个存储器字（寄存器）或位（线圈）的地址。
长度	是	有关最大长度，请参见支持的 Modbus 功能代码 (参见第 *178* 页)	-	要读取的存储器字（寄存器）或位（线圈）的数量。
错误管理	是	**设置为零** **保留最后一个值**	**设置为零**	指定如何管理无法从设备读取数据的情形： ● 选择**设置为零**，将所接收的最后数据值设置为零。 ● 选择**保留最后一个值**，保留所接收的最后数据值。
写入对象				
偏移	是	0...65535	0	要写入的首个存储器字（寄存器）或位（线圈）的地址。
长度	是	有关最大长度，请参见支持的 Modbus 功能代码 (参见第 *178* 页)	-	要写入的存储器字（寄存器）或位（线圈）的数量。

单击**确定**，完成通道配置。

第8.2节
串行线路配置

本节包含了哪些内容？

本节包含了以下主题：

主题	页
配置串行线路	164
配置 Modbus Serial IOScanner	170
在 Modbus Serial IOScanner 上添加设备	171

配置串行线路

简介

M100/M200 可编程控制器 所有型号标配内置一路串行线路 (SL1)。

串行线路可以针对以下其中一种协议配置：

- Modbus（RTU 或 ASCII）(参见第 *164* 页)。缺省情况下，串行线路配置为 Modbus RTU 协议。
- ASCII (参见第 *164* 页)
- Modbus Serial IOScanner (参见第 *170* 页)。只能配置一个实例：如果在一条串行线路上配置，则无法用于其他串行线路。
  注意： 如果应用程序中既使用了 Modbus Serial IOScanner，又使用了消息 (%MSG) 功能块，则配置时必须小心，否则可能导致正在进行的 IOScanner 通讯被取消。

  此应用程序必须配置至少 **5.0 级**的功能级别才能支持 Modbus Serial IOScanner。

串行线路配置

下表介绍了如何配置串行线路：

步骤	动作
1	单击硬件树中的 **SL1（串行线路）**节点，显示串行线路属性。 下图显示 **Modbus RTU** 和 **Modbus ASCII** 协议的串行线路属性：

步骤	动作
	下图显示了适用于 ASCII 协议的串行线路的属性： 串行线路配置 协议设置 协议 ASCII 串行线路设置 波特率 19200 校验位 偶数 数据位 8 停止位 1 物理介质 RS-485 RS-232 极化 否 应用 取消 ASCII 设备设置 设备 无 初始命令 协议设置 响应超时（x 100 毫秒） 10 停止条件 收到的帧长度 0 帧收到超时（毫秒） 0 帧结构 起始字符 0 第一个结束字符 10 <LF> 第二个结束字符 0 发送帧字符 应用 取消
2	编辑属性以配置串行线路。 有关串行线路配置参数的详细信息，请参阅下表。

下表介绍了串行线路的每个参数：

参数	可编辑	值	缺省值	描述
物理设置				
波特率	是	1200 2400 4800 9600 19200 38400 57600 115200	19200	用于从下拉列表中选择调制解调器的数据传输速率（每秒位数）。
奇偶校验	是	**无** **偶数** **奇数**	**偶数**	用于为错误检测选择对传输数据的奇偶校验。奇偶校验是传输中使用的错误检测方法。 如果奇偶校验与串行端口搭配使用，则会随每个数据字符发送一个额外的数据位。此安排可确保每个字符中设置为 1 的位数（包括奇偶校验位）始终为奇或始终为偶。 如果收到的字节中包含的设置为 1 的位数错误，则字节会损坏。
数据位	是 （仅针对 ASCII 协议）	7 8	8	用于从下拉列表中选择数据位。 每个字符中的数据位数可能是 7（对于真正的 ASCII）或 8。
停止位	是	1 2	1	用于从下拉列表中选择停止位。 停止位是指示数据字节结束的一个位。 对于电子设备来说， 通常使用 1 个停止位。对于像机电电传打字机这样较慢的设备来说， 将使用 2 个停止位。
物理介质	是	RS-485 True	RS-485 True	用于选择通讯的物理介质。 数据通讯中的物理介质为传播信号使用的传输路径。它是设备与逻辑控制器互连的接口。
极化	否	**是** **否**	**否**	扩展板模块中集成了极化电阻器。 对于控制器，将禁用此参数， 对于扩展板，此参数允许您开启或关闭极化。
协议设置				
协议	是	Modbus RTU Modbus ASCII ASCII Modbus Serial IOScanner	Modbus RTU	用于从下拉列表中选择通讯的协议传输模式。 根据选择的协议显示高级协议参数。 请参阅下图和下表。

参数	可编辑	值	缺省值	描述
Modbus RTU 和 Modbus ASCII 协议的协议设置：				
寻址	是	**从站** True/False **主任务** True/False	**从站** True	用于选择寻址模式。您只能选择**从站**或**主站**寻址。选择任一寻址模式后，会清除现有的模式。
地址 [1...247]	是	1...247	1	用于指定从站的地址 ID。 **注意：** 仅为从站寻址显示此字段。对于主站，不会在屏幕上显示此字段。
响应时间（× 100 毫秒）	是	0...255 毫秒	10	用于为查询指定协议的响应时间。
帧间时间（毫秒）	是	2...255 毫秒	10	用于指定协议帧间的时间周期。（对应于其他产品中使用的帧间延迟）。此规定值会自动调整以符合 Modbus 标准 3.5 字符时间延时。
ASCII 协议的协议设置：				
响应时间（× 100 毫秒）	是	0...255 毫秒	10	用于为查询指定协议的响应时间。
停止条件				
收到的帧长度	是（仅当选中复选框时）	1...255	0（如果未选中复选框） 1（如果已选中复选框）	使您可以指定接收帧的长度。 **注意：** 对于**收到的帧长度**或**帧收到超时（毫秒）**类型的停止条件，只能配置一个参数。
帧收到超时（毫秒）	是（仅当选中复选框时）	1...255	0（如果未选中复选框） 10（如果已选中复选框）	可让您指定收到的帧的超时持续时间。
帧结构				
起始字符	是（仅当选中复选框时）	1...255	0（如果未选中复选框） 58（如果已选中复选框）	用于指定帧的起始字符。 与起始字符值对应的 ASCII 字符显示在值字段的右侧。
第一个结束字符	是	1...255	0（如果未选中复选框） 10（如果已选中复选框）	用于指定帧的第一个结束字符。 **注意：** 要能够启用或禁用**第一个结束字符**，请至少配置一个停止条件。 与第一个结束字符值对应的 ASCII 字符显示在值字段的右侧。

参数	可编辑	值	缺省值	描述
第二个结束字符	是 （仅当选中复选框时）	1...255	0（如果未选中复选框） 10（如果已选中复选框）	用于指定帧的第二个结束字符。 **注意：** 禁用**第一个结束字符**参数时，将禁用此字段。 与第二个结束字符值对应的 ASCII 字符显示在值字段的右侧。
发送帧字符	是	True/False	False	用于对 ASCII 协议启用或禁用发送帧的第一个结束字符。

配置 Modbus Serial IOScanner

描述

只能定义 IOScanner 的一个实例：如果您将其配置在以太网端口上，则无法将其配置在串行端口上。请参阅配置 Modbus TCP IOScanner (参见第 *170* 页)。

TCP 和 Serial IOScanner 寄存器的最大个数为：

- 128，如果**功能级别 < 6.0**。
- 512，如果**功能级别 ≥ 6.0**。

协议设置

下表描述了在选择了 **Modbus Serial IOScanner** 协议时的参数：

参数	可编辑	值	缺省值	描述
传输模式	是	RTU ASCII	RTU	从下拉列表中选择通讯的协议传输模式。
响应超时 **（x 100 毫秒）**	是	0...255	10	定义在错误中终止交换前控制器等待响应的最长时间。 输入 0 可禁用超时。
帧间时间（毫秒）	是	1...255	10	帧之间时间的周期（对应于其他产品中所用的帧间延时）。 **注意：** 此值可以调查以符合 Modbus 标准 3.5 字符延时。

在 Modbus Serial IOScanner 上添加设备

简介

本节介绍了如何添加要供 Modbus Serial IOScanner 扫描的设备。

您可以最多添加 16 个 Modbus 从站设备。

SoMachine Basic 具有多个预定义的设备类型。预定义的设备类型具有预定义的初始化请求和预配置通道，以简化设备在网络中的集成。

还可提供必须配置初始化请求和通道的一般从站设备。

在 Modbus Serial IOScanner 上添加设备

在 Modbus Serial IOScanner 上添加设备：

步骤	动作
1	以下选项任选其一： ● **驱动器**，然后从下拉列表中选择其中一种支持的设备类型。 ● **其他**，然后从下拉列表中选择设备类型。 如果在两个列表中都无法找到您的设备类型，则选择**一般设备**，然后对其进行配置。
2	单击**添加**。
3	根据设备设置 (参见第 *171* 页)的说明来配置设备。
4	单击**应用**。

设备设置

下表描述了在选择了 Modbus Serial IOScanner 协议时的参数：

参数	可编辑	值	缺省值	描述
ID	否	0...15	**0**	SoMachine Basic 分配的唯一设备标识符。
名称	是	1...32 个字符 设备名称必须具有唯一性。	**设备 x**(1)	为设备指定唯一的名称。
地址	否	– %DRVn (1) (2)	– %DRV0	%DRVn 用于在使用 驱动器功能块 (参见第 *262* 页)的应用程序中配置设备。
类型	否	设备的类型	–	设备类型不可编辑。 如要修改设备类型，必须从列表中移除该设备（具体方式为：右键单击并选择**删除**）， 然后添加正确的设备类型。

(1) *x* 和 *n* 是整数，每次添加了设备或驱动设备后，计数递增。
(2) 只有当选择的设备类型为**驱动器**时。

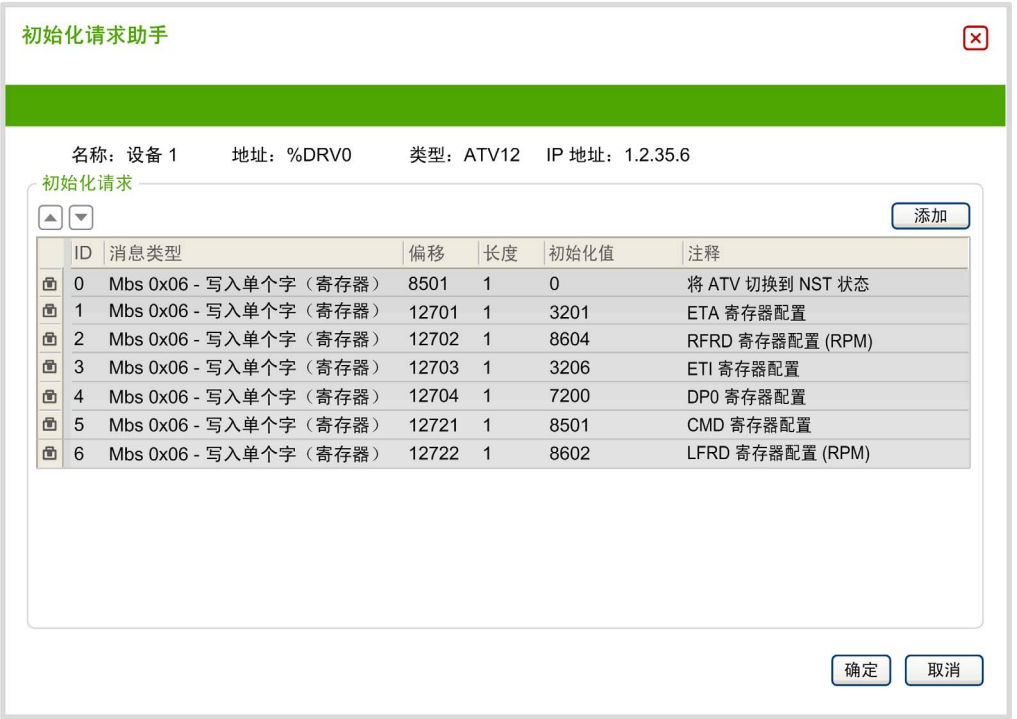
参数	可编辑	值	缺省值	描述
从站地址	是	1...247	1	用于识别网络中的设备的地址。 从站地址允许重复。
响应超时 （x 100 毫秒）	是	0...255	10	与设备进行数据交换时所使用的超时（毫秒）。这个值可以根据具体的设备单独调整，并且会覆盖**协议设置**中为主站设置的**响应超时**。
复位变量	是	%Mn	–	指定要用来复位设备的存储器位的地址（重新发送初始化请求）。在指定的存储器位被应用程序设置为 1 时，设备复位。
初始化	是	...	-	单击以显示初始化请求助手窗口 (参见第 *173* 页)。
通道	是	...	-	单击以显示通道助手窗口 (参见第 *175* 页)。
(1) *x* 和 *n* 是整数，每次添加了设备或驱动设备后，计数递增。 (2) 只有当选择的设备类型为**驱动器**时。				

配置初始化请求

初始化请求是 Modbus TCP IOScanner 或 Modbus Serial IOScanner 为了初始化从站设备而发送的设备特定的命令。在设备确认完所有初始化请求之前，Modbus TCP IOScanner 或 Modbus Serial IOScanner 不启动与设备的循环数据交换。初始化期间，不更新网络对象。

最多可以为每个从站设备定义 20 个初始化请求。

初始化请求助手窗口会显示定义的初始化请求：



ID	消息类型	偏移	长度	初始化值	注释
0	Mbs 0x06 - 写入单个字（寄存器）	8501	1	0	将 ATV 切换到 NST 状态
1	Mbs 0x06 - 写入单个字（寄存器）	12701	1	3201	ETA 寄存器配置
2	Mbs 0x06 - 写入单个字（寄存器）	12702	1	8604	RFRD 寄存器配置 (RPM)
3	Mbs 0x06 - 写入单个字（寄存器）	12703	1	3206	ETI 寄存器配置
4	Mbs 0x06 - 写入单个字（寄存器）	12704	1	7200	DP0 寄存器配置
5	Mbs 0x06 - 写入单个字（寄存器）	12721	1	8501	CMD 寄存器配置
6	Mbs 0x06 - 写入单个字（寄存器）	12722	1	8602	LFRD 寄存器配置 (RPM)

预配置的初始化请求显示有锁符号 🔒 和灰色背景。预定义初始化请求的一些参数无法修改。

根据所选择的设备类型，可以配置某些初始化请求。

下表描述了初始化请求的属性：

参数	可编辑	值	缺省值	描述
ID	否	0...19	0	唯一的初始化请求标识符。
消息类型	如果未预定义初始化请求，则为是。	请参见支持的 Modbus 功能代码 (参见第 *178* 页)	Mbs 0x05 - 写入单个位（线圈）	针对此初始化请求要使用的交换类型选择 Modbus 功能代码。 **注意：** 如果配置不支持缺省 Mbs 0x05 - Write single bit (coil) 请求类型的一般设备，则必须将缺省值替换为所支持的请求类型。
偏移	如果未预定义初始化请求，则为是。	0...65535	0	要初始化的首个寄存器的偏移。
长度	如果未预定义初始化请求，则为是。	1（对于 Mbs 0x05 - 写入单个位（线圈）） 1（对于 Mbs 0x06 - 写入单个字（寄存器）） 128（对于 Mbs 0x0F - 写入多个位（线圈）） 123（对于 Mbs 0x10 - 写入多个字（寄存器））	1	要初始化的对象数量（存储器字或位）。例如，如果以**偏移** = 2 且**长度** = 3 来写入多个字，则会初始化%MW2、%MW3 和 %MW4。
初始化值	如果未预定义初始化请求，则为是。	0...65535（如果正初始化存储器字（寄存器）） 0...1（如果正初始化存储器位（线圈））	0	用于初始化目标寄存器的值。
注释	如果未预定义初始化请求，则为是。	-	空	可以选择输入与该请求关联的注释。

单击**添加**，创建新的初始化请求。

选择一个条目，然后使用上、下箭头按钮改变初始化请求发送至设备的顺序。

在定义了初始化请求后，单击**确定**，保存配置，然后关闭**初始化请求助手**。

通道助手

最多可以为每个从站设备定义 10 个通道。每个通道代表一个 Modbus 请求。

注意： 在单击属性窗口上的**应用**后，所定义的对象数（读取和写入的数据项）将生效。

通道助手窗口列出了已定义的通道：

预配置的通道显示有锁符号 和灰色背景。预定义通道的一些参数无法修改。

下表描述了通道的属性：

参数	可编辑	值	缺省值	描述
ID	否	0...19， 最多10个通道	0	唯一的初始化标识符。
名称	是	0...32 个字符	Device_channel0	双击以编辑通道名称。
配置	是	...	-	单击以显示通道助手窗口 (参见第 *160* 页)。
消息类型	否	-	-	通道助手窗口 (参见第 *160* 页)中选择的 Modbus 功能代码。
触发器	否	-	-	通道助手窗口 (参见第 *160* 页)上选择的触发类型和循环时间。
R Offset	否	-	-	通道助手窗口 (参见第 *160* 页)中选择的读取对象偏移。

参数	可编辑	值	缺省值	描述
R Length	否	-	-	通道助手窗口 (参见第 *160* 页)中选择的读取对象长度。
错误管理	否	-	-	通道助手窗口 (参见第 *160* 页)中选择的错误管理策略。
W Offset	否	-	-	通道助手窗口 (参见第 *160* 页)中选择的写入对象偏移。
W Length	否	-	-	通道助手窗口 (参见第 *160* 页)中选择的写入对象长度。
注释	是	-	空	可以选择输入与该通道关联的注释。

单击**添加**，创建新的通道。

在定义了通道后，单击**确定**，保存配置，然后关闭**通道助手**。

配置通道

使用**通道助手**窗口配置通道。

以下示例示出了为“读/写多个字”请求配置的通道（Modbus 功能代码 23）。它以偏移 16#0C21 从寄存器读取一个字，并且以偏移 16#0C20 向寄存器写入两个字。在定义的**触发器**存在上升沿时，执行该请求（参见下表）：

下表描述了通道的属性：

参数	可编辑	值	缺省值	描述
名称	是	0...32 个字符	Device 0_ Channel0	输入通道的名称。
消息类型	是	请参见支持的 Modbus 功能代码 (参见第 *178* 页)	**Mbs 0x17 - 读取/写入多个字（寄存器）**	针对此通道要使用的交换类型选择 Modbus 功能代码。
触发器	是	**循环** **上升沿**	**循环**	为数据交换选择触发类型： ● **循环**：以**循环时间（x 10 毫秒）**字段中定义的频率来触发请求 ● **上升沿**： 检测到存储器位的上升沿时，触发请求。 指定要使用的**存储器位**的地址。
循环时间（× 10 毫秒） （如果选择的是**循环**）	是	1...6000	20	指定循环触发循环时间，以 10 毫秒为单位。
存储器位 （如果选择的是**上升沿**）	是	%Mn	-	指定存储器位地址，比如 %M8。在检测到该存储器位的上升沿时，触发数据交换。
注释	是	-	空	可以选择输入描述通道用途的注释。
读取对象				
偏移	是	0...65535	0	要读取的首个存储器字（寄存器）或位（线圈）的地址。
长度	是	有关最大长度， 请参见支持的 Modbus 功能代码 (参见第 *178* 页)	-	要读取的存储器字（寄存器）或位（线圈）的数量。
错误管理	是	**设置为零** **保留最后一个值**	**设置为零**	指定如何管理无法从设备读取数据的情形： ● 选择**设置为零**，将所接收的最后数据值设置为零。 ● 选择**保留最后一个值**，保留所接收的最后数据值。
写入对象				
偏移	是	0...65535	0	要写入的首个存储器字（寄存器）或位（线圈）的地址。
长度	是	有关最大长度， 请参见支持的 Modbus 功能代码 (参见第 *178* 页)	-	要写入的存储器字（寄存器）或位（线圈）的数量。

单击**确定**，完成通道配置。

第8.3节
支持的 Modbus 功能代码

支持的 Modbus 功能代码

简介

本节列出了支持的 Modbus 功能代码及其对以下项的控制器内存变量的影响：
- Modbus Serial
- Modbus Serial IOScanner (参见第 *179* 页)
- Modbus TCP (参见第 *179* 页)
- Modbus TCP IOScanner (参见第 *179* 页)

Modbus Serial

以下是支持的 Modbus 请求：

支持的 Modbus 功能代码	支持的子功能代码	描述
1 (0x01)或 2 (0x02)	–	读取多个内部位 %M
3 (0x03)或 4 (0x04)	–	读取多个内部寄存器 %MW
5 (0x05)	–	写入单个内部位 %M
6 (0x06)	–	写入单一内部寄存器 %MW
8 (0x08)	0 (0x00), 10 (0x0A)...18 (0x12)	诊断
15 (0x0F)	–	写入多个内部位 %M
16 (0x10)	–	写入多个内部寄存器 %MW
23 (0x17)	–	读取/写入多个内部寄存器 %MW
43 (0x2B)	14 (0x0E)	读取设备标识（常规服务）

注意：
主 M100/M200 逻辑控制器所使用的 Modbus 功能代码的影响取决于从站设备类型。在从站设备的主要类型中：
- 内部位表示 %M
- 输入位表示 %I
- 内部寄存器表示 %MW
- 输入寄存器表示 %IW

视从站和从站地址的类型而定，内部位应为 %M 或 %Q；输入位应为 %I 或 %S；输入寄存器应为 %IW 或 %SW；内部位寄存器应为 %MW 或 %QW。有关更多详情，请参阅从站设备。

Modbus Serial IOScanner 和 Modbus TCP IOScanner

下表列出了 Modbus Serial IOScanner 和 Modbus TCP IOScanner 支持的 Modbus 功能代码：

功能代码十进制（十六进制）	描述	可用于配置	最大长度（位）
1 (1 hex)	读取多个位（线圈）	**通道**	128
2 (2 hex)	读取多个位（离散量输入）	**通道**	128
3 (3 hex)	读取多个位（保持寄存器）	**通道**	125
4 (4 hex)	读取多个位（输入寄存器）	**通道**	125
5 (5 hex)	写入单个位（线圈）	**通道** **初始化值**（用于初始化值的缺省消息类型）	1
6 (6 hex)	写入单个字（寄存器）	**通道** **初始化值**	1
15 (0F hex)	写入多个位（线圈）	**通道** **初始化值**	128
16 (10 hex)	写入多个字（寄存器）	**通道** **初始化值**	123
23 (17 hex)	读/写多个字（寄存器）	**通道**（用于通道配置的缺省消息类型）	125（读取） 121（写入）

Modbus TCP 的 Modbus 映射表

Modbus TCP 从站设备支持 Modbus 功能代码的子集。来自 Modbus 主站且带有匹配单元 ID 的功能代码会被引导至 Modbus 映射表并访问控制器的网络对象（%IWM 和 %QWM）。请参阅 Modbus TCP 从站设备 I/O 映射表 (参见第 *154* 页)。

第9章
Micro SD 卡

简介

Modicon M100/M200 可编程控制器 允许使用 Micro SD 卡传输文件。

本章介绍如何使用 Micro SD 卡管理 Modicon M100/M200 可编程控制器 文件。

本章包含了哪些内容？

本章包含了以下主题：

主题	页
文件管理操作	182
SD 卡支持的文件类型	184
克隆管理	186
固件管理	188
应用程序管理	193
后配置管理	197
错误日志管理	205
存储器管理：备份和恢复控制器存储器	211

文件管理操作

简介

Modicon M100/M200 可编程控制器 允许使用 micro SD 卡进行以下类型的文件管理：

- 克隆管理 (参见第 *186* 页)：备份逻辑控制器的应用程序、固件和后配置（如有）
- 固件管理 (参见第 *188* 页)：将固件更新直接下载至逻辑控制器
- 应用程序管理 (参见第 *193* 页)：备份并恢复逻辑控制器应用程序，或者将其复制到相同应用的另一个逻辑控制器
- 后配置管理 (参见第 *197* 页)：添加、更改或删除逻辑控制器的后配置文件
- 错误日志管理 (参见第 *205* 页)：备份或删除逻辑控制器的错误日志文件
- 存储器管理 (参见第 *211* 页)：备份和恢复逻辑控制器的存储对象

注意：

- 某些命令需要重启逻辑控制器。请参阅命令描述了解更多信息。
- Modicon M100/M200 可编程控制器 只接受格式化为 FAT 或 FAT32 的微型 SD 卡。应使用知名品牌的 SD 卡，而且该卡容量不得大于 64 GB。

通过使用 SD 卡，可以自动执行各种功能强大的操作，进而影响 Logic Controller 和常驻应用程序的行为。将 SD 卡插入控制器时，必须小心；您必须了解 SD 卡的内容将对 Logic Controller 造成的影响。

注意： 通过 SD 卡执行的文件管理使用脚本文件。可使用存储器管理任务 (参见第 *211* 页)手动创建这些脚本。

⚠ 警告

意外的设备操作

- 在将此设备连接到控制器之前，您必须具有机器或过程的操作知识。
- 确保防护措施已到位，以便任何可能的意外设备操作不会导致人身伤害或设备损坏。

不遵循上述说明可能导致人员伤亡或设备损坏。

如果移除设备电源，或者在应用程序的数据传输期间出现断电或通讯中断，则设备可能无法正常工作。如果出现断电或通讯中断，请再次尝试传输。如果在固件更新过程中出现断电或通讯中断，或者如果使用了无效固件，则设备可能无法正常工作。在这种情况下，使用有效的固件并重新尝试固件更新。

注意

设备无法操作

- 传输一旦开始，不要中断应用程序的传输或固件更改。
- 如传输因任何原因中断，则重新开始传输。
- 在文件传输成功完成之前，不得将设备（逻辑控制器、运动控制器或 HMI 控制器或驱动器）投入运行。

不遵循上述说明可能导致设备损坏。

注意： 在将微型 SD 卡插入控制器之前，如果微型 SD 卡不为空，请验证微型 SD 卡是否包含有效的脚本文件。否则，控制器将保持在 BOOTING 模式下，因为它在启动期间检测不到任何有效的脚本。在此情况下，拔出微型 SD 卡以使控制器正常启动。

SD 卡支持的文件类型

简介

下表列出了文件位置和克隆或脚本命令可管理的文件类型：

文件夹	描述	缺省文件名
/	脚本文件	Script.cmd
/	脚本日志	Script.log
/sys/os	逻辑控制器固件文件	M200M100.mfw
/TM3	TM3 模拟量扩展模块固件	TM3_Ana.mfw
/usr/app	应用程序文件	*.smbk
/usr/cfg	后配置文件	Machine.cfg
/usr/mem	存储器备份文件	Memories.csv
/sys/log	已检测到的错误日志文件	PlcLog.csv

脚本文件命令

脚本文件是存储在微型 SD 卡根目录中的文本文件，其中包含用于管理与控制器交换的命令。脚本文件必须编码成 ANSI 格式。

下表介绍了支持的脚本命令

命令	描述	源极	目标
下载	将微型 SD 卡中的文件下载到控制器。	选择要下载的文件。	选择控制器目标文件夹。
上载	将控制器存储器的文件夹中包含的文件上载至微型 SD 卡。	选择文件夹。	选择微型 SD 卡文件夹。
删除	删除控制器文件夹中包含的文件。	选择文件夹，然后输入特定文件名。 **重要信息：**缺省情况下，将选择所有文件夹文件。	-
重新启动	重新启动控制器（此命令必须是脚本中的最后一个命令）。	-	-

注意：

- 仅当控制器处于 STOP 或 BOOTING 状态时，才能执行所有命令。如果在控制器处于 SD 模式下时插入微型 RUN 卡，则不会执行任何命令。
- 在执行微型 SD 卡命令时，将忽略启动控制器命令。当微型 SD 卡命令执行完成后，必须手动启动控制器。为此，请将 RUN/STOP 按钮切换到 RUN 位置，或在 SoMachine Basic 中执行 RUN 命令。

脚本文件示例

下载命令：

```
Download "/usr/Cfg"
```

```
Download "/sys/os/M200M100.mfw"
```

上载命令：

```
Upload "/usr/app/*"
```

```
Upload "/usr/cfg/Machine.cfg"
```

删除命令：

```
Delete "/usr/app/*"
```

重新启动命令：

```
Reboot
```

注意： **上载**或**删除**命令中指定的后配置文件的扩展名必须为 .cfg 或 .CFG。
如未指定后配置文件，或者指定的文件名不存在，则使用缺省文件名 Machine.cfg。

脚本日志

脚本操作完成后，将在 SD 卡根目录中自动创建一个 script.log 文件。要验证脚本操作的状态，可阅读该文件。

克隆管理

克隆

功能 1：克隆用于将 Modicon M100/M200 可编程控制器 的应用程序、固件和后配置（如果存在）备份到微型 SD 卡。无需创建任何脚本。

功能 2：微型 SD 卡以后可用于将固件、应用程序和后配置（如果存在）恢复到逻辑控制器，或将其复制到相同型号的另一个逻辑控制器。

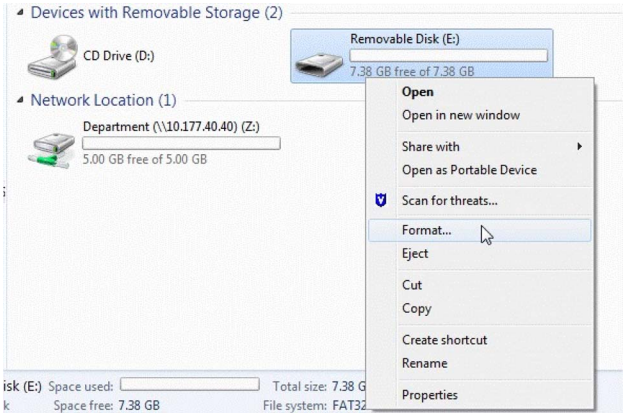
在克隆控制器之前，M100/M200 可编程控制器 会验证应用程序是否处于复制保护状态。关于详情，请参与 密码保护应用程序 *(参见 SoMachine Basic, 操作指南)*。

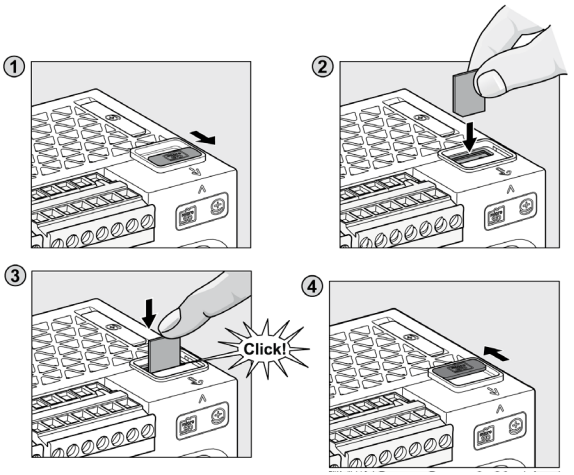
注意：

- 为实现功能 1，微型 SD 卡必须为空且正确格式化才能执行此过程。
- 不会克隆检测到的错误日志和数据存储器。
- 如果应用程序受密码保护，则克隆操作会完成但用户应用程序无法恢复，且 **ERR** LED 常亮。

创建克隆微型 SD 卡（功能 1）

以下过程介绍如何将应用程序、固件和后配置（如果存在）从控制器复制到微型 SD 卡：

步骤	动作
1	在 PC 上格式化一张 SD 卡（微型 SD 卡为空且正确格式化才能执行此过程）。 
2	拔下控制器的电源。

步骤	动作
3	将微型 SD 卡插入控制器。 
4	恢复对控制器供电。 **结果：** 克隆操作自动启动，在操作期间，**SD** LED 将亮起。 如果检测到错误，**SD** LED 会闪烁，检测到的错误将记录在 Script.log 文件中。
5	等到操作结束（直到 **SD** LED 熄灭或闪烁）。 **注意：** 克隆操作持续 2 至 3 分钟。克隆操作的优先级较低，以便最大限度地减少对逻辑控制器的程序和通讯性能的影响。
6	从控制器中移除微型 SD 卡。创建好克隆微型 SD 卡后，妥善保管该卡。

从克隆微型 SD 卡中恢复或复制（功能 2）

以下过程介绍如何将存储在微型 SD 卡中的应用程序、固件和后配置（如果存在）下载到控制器：

步骤	动作
1	恢复从控制器供电。
2	将克隆微型 SD 卡插入控制器。
3	恢复对控制器供电。 **结果：** 克隆操作正在进行。 **注意：** 在操作期间，**SD** LED 亮起。
4	等到操作结束（直到 **SD** LED 熄灭或闪烁）。 如果检测到错误，**SD** 和 **ERR** LED 会闪烁，检测到的错误将记录在 Script.log 文件中。
5	移除微型 SD 卡以重新启动控制器。

注意： 若将已克隆的应用程序下载到控制器，首先会从控制器存储器中删除现有的应用程序，不管在目标控制器中启用了哪个用户访问权限。

固件管理

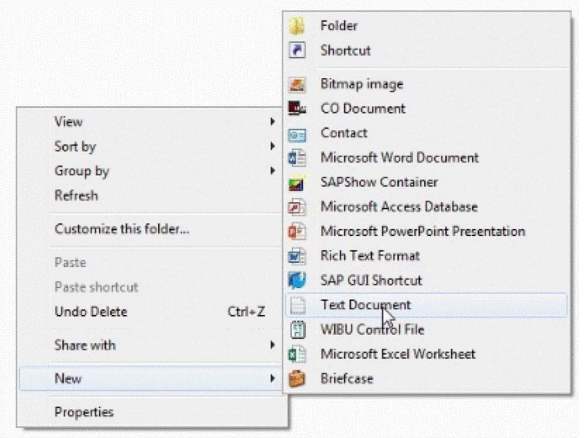
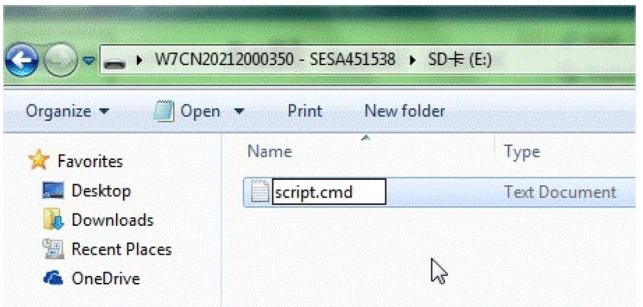
概述

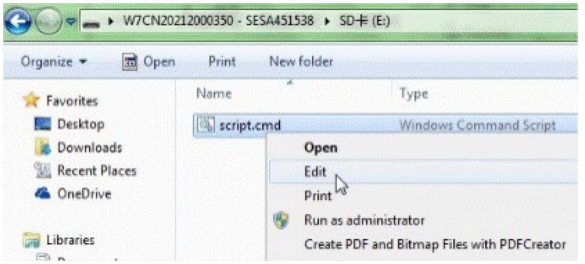
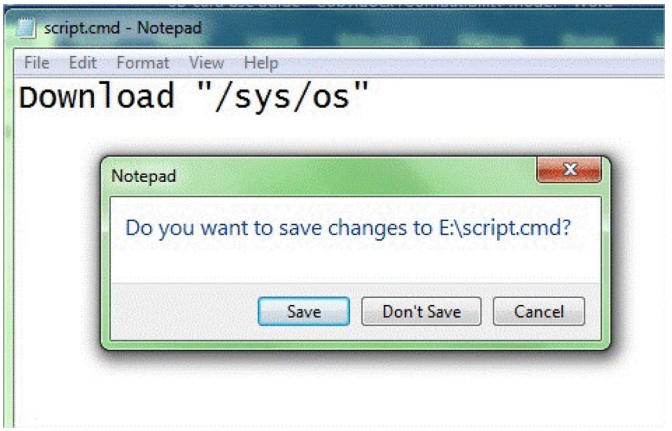
您可以使用微型 SD 卡将固件更新直接下载到逻辑控制器或 TM3 模拟量扩展模块。

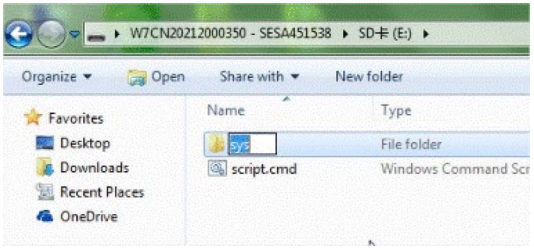
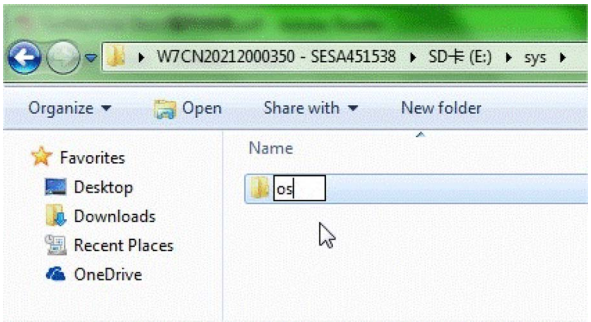
有关逻辑控制器操作状态和 LED 状态的信息，请参阅控制器状态和行为 (参见第 *39* 页)。

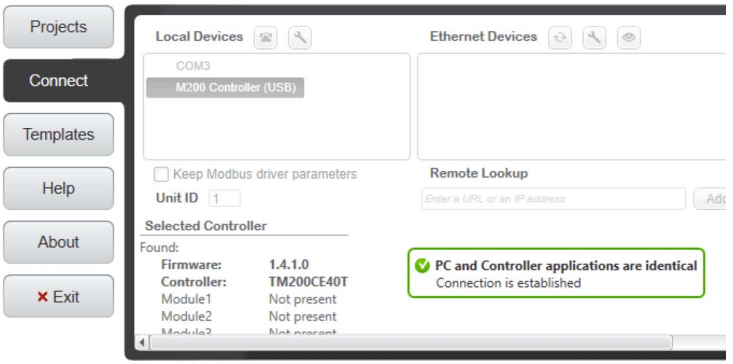
将固件下载到控制器

下表介绍了如何用 SD 卡将固件下载到逻辑控制器中：

步骤	动作
1	将空微型 SD 卡插入到运行 SoMachine Basic 的 PC。
2	在 SD 卡根目录中创建名为 script.cmd 的文件。  

步骤	动作
3	编辑 script.cmd 文件并插入以下行： Download "/sys/os"  

步骤	动作
4	在 SD 卡根目录中创建文件夹 sys，并在该文件夹下创建子文件夹 os。然后，将固件文件复制到 os 文件夹中：   **注意：** SoMachine Basic 安装目录的 Firmwares & PostConfiguration\M200M100\ 文件夹中提供了固件文件示例和脚本。 M100/M200 可编程控制器 的固件文件名为 M200M100.mfw。
5	从 PC 中移除微型 SD 卡。
6	拔下控制器的电源，并将微型 SD 卡插入逻辑控制器的 SD 卡槽中。
7	恢复对控制器供电。 **结果：**开始复制固件文件。在操作期间，逻辑控制器上的 **SD** 系统 LED 亮起。 **注意：** 操作进行期间，避免关闭逻辑控制器的电源。
8	等到操作结束（直到 **SD** 系统 LED 熄灭或闪烁）。
9	拔除 SD 卡。
10	将 USB 编程电缆重新连接到逻辑控制器并使用 SoMachine Basic 软件登录到逻辑控制器。 通过读取在 SD 卡根目录中自动创建的 script.log 文件，可以验证控制器固件更新的状态。

步骤	动作
11	确认固件更新状态。 

将固件下载到 TM3 模拟量扩展模块

可在固件版本不低于 26 的 TM3 模拟量扩展模块中更新固件。必要时，可使用 SoMachine Basic 确认固件版本。

使用 SD 卡上的脚本文件来执行固件更新。在 SD 卡已插入 M100/M200 可编程控制器 的 SD 卡插槽的情况下，逻辑控制器更新 I/O 总线上的以下 TM3 模拟量扩展模块的固件：

- 远程连接的那些，使用 TM3 发射器/接收器模块来更新
- 在混合了 TM3 和 TM2 扩展模块的配置中。

下表介绍了如何用 SD 卡将固件下载到一个或多个 TM3 模拟量扩展模块：

步骤	动作
1	为逻辑控制器上电。

2	删除逻辑控制器中的应用程序，确保逻辑控制器处于 EMPTY 状态。为此，可以使用以下其中一个脚本命令借助 SoMachine Basic 来执行：
	`Delete "usr/*"`
	`Delete "usr/app"`
	有关详细信息，请参阅文件管理操作 (参见第 *182* 页)。

3	将空的 SD 卡插入到 PC 中。
4	在 SD 卡根目录中创建名为 `script.cmd` 的文件。

步骤	动作
5	编辑该文件并插入以下命令： Download "/TM3/<filename>/*" **注意：** <filename> 是要更新的固件的文件名。星号表示所有模拟模块都将更新。 如要将固件下载到某个特定的 TM3 模拟量扩展模块，则用星号代替配置中的扩展模块位置。例如，如要指定位置 4 处的模块，则： Download "/TM3/<filename>/4"
6	在 SD 卡根目录中创建文件夹路径 /TM3/ 并在 TM3 文件夹中复制固件文件。 **注意：** 固件文件（固件文件在 SoMachine Basic 安装时生效）和示例脚本位于 SoMachine Basic 安装文件夹的文件夹 Firmwares & PostConfiguration\TM3\ 中。
7	从 PC 中拔出 SD 卡并将其插入 M100/M200 可编程控制器 的 SD 卡槽。 **结果：**逻辑控制器开始将固件文件从 SD 卡传输到可更新的 TM3 模拟量扩展模块或者步骤 5 中指定的那个模块。在该操作期间，M100/M200 可编程控制器 上的 **SD** 系统 LED 亮起。 **注意：** 每个扩展模块的固件更新时间为 10 到 15 分钟。在操作中请勿断开 M100/M200 可编程控制器 电源或移除 SD 卡。否则，可能无法成功更新固件，模块也可能不再正确工作。在这种情况下，请运行恢复程序，重新初始化模块上的固件。
8	等到操作结束（直到 **SD** LED 熄灭或闪烁）。 如果检测到错误，**SD** 和 **ERR** LED 会闪烁，检测到的错误将记录在Script.log 文件中。

如果移除设备电源，或者在应用程序的数据传输期间出现断电或通讯中断，则设备可能无法正常工作。如果出现断电或通讯中断，请再次尝试传输。如果在固件更新过程中出现断电或通讯中断，或者如果使用了无效固件，则设备可能无法正常工作。在这种情况下，使用有效的固件并重新尝试固件更新。

注意

设备无法操作

- 传输一旦开始，不要中断应用程序的传输或固件更改。
- 如传输因任何原因中断，则重新开始传输。
- 在文件传输成功完成之前，不得将设备（逻辑控制器、运动控制器或 HMI 控制器或驱动器）投入运行。

不遵循上述说明可能导致设备损坏。

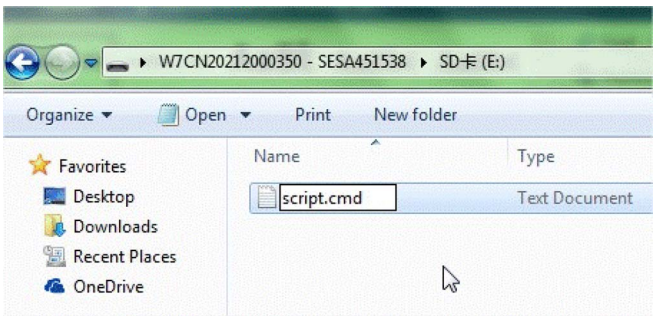
应用程序管理

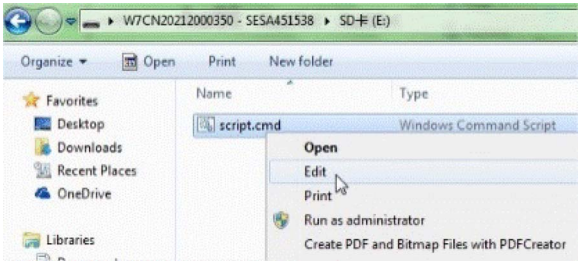
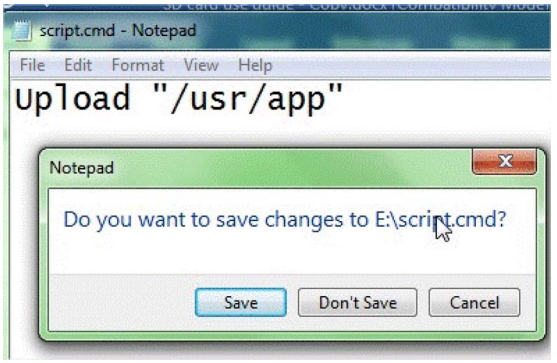
概述

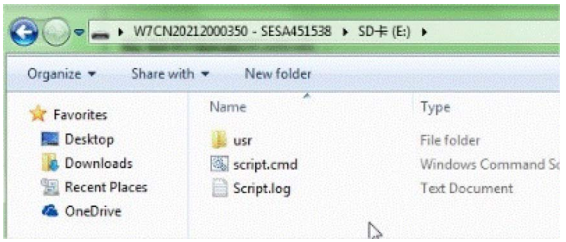
您可以使用微型 SD 卡备份和恢复控制器应用程序，或将其复制到相同型号的另一个控制器。

备份应用程序

下表介绍如何在微型 SD 卡上备份控制器应用程序：

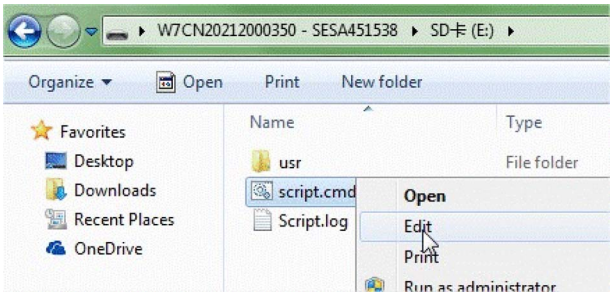
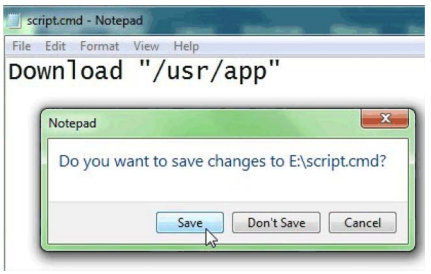
步骤	动作
1	使用 PC 上的文本编辑器在 SD 卡根存储器中创建 script.cmd 文件。 

步骤	动作
2	编辑 script.cmd 文件并插入以下行： Upload "/usr/app"  
3	拔下控制器的电源。
4	将准备好的微型 SD 卡插入控制器。
5	恢复对控制器供电。 **结果：**开始复制应用程序文件。在操作期间，逻辑控制器上的 **SD** 系统 LED 亮起。 **注意：** 操作进行期间，避免关闭逻辑控制器的电源。 **注意：** 应用程序备份过程的优先级较低，以便最大限度地减少对逻辑控制器程序和通讯性能的影响。

步骤	动作
6	等到操作结束（直到 **SD** LED 熄灭或闪烁）。 如果检测到错误，**SD** 和 **ERR** LED 会闪烁，检测到的错误将记录在 Script.log 文件中。 **结果**：应用程序文件 (*.smbk) 保存在微型 SD 卡上。 

将应用程序恢复或复制到其他控制器

下表介绍如何将控制器应用程序从微型 SD 卡传输到控制器：

步骤	动作
1	使用先前创建的 SD 卡，在文本编辑器中编辑 SD 卡根文件夹中的 script.cmd 文件。 
2	用下行内容替换脚本的内容： Download "/usr/app" 
3	断开控制器的电源。

步骤	动作
4	将准备好的微型 SD 卡插入控制器。
5	恢复对控制器供电。 **结果：**开始复制应用程序文件。在操作期间，逻辑控制器上的 **SD** 系统 LED 亮起。 **注意：** 操作进行期间，避免关闭逻辑控制器的电源。
6	等到操作结束（直到 **SD** LED 熄灭或闪烁）。 如果检测到错误，**SD** 和 **ERR** LED 会闪烁，检测到的错误将记录在 Script.log 文件中。
7	移除 SD 卡以重新启动控制器。

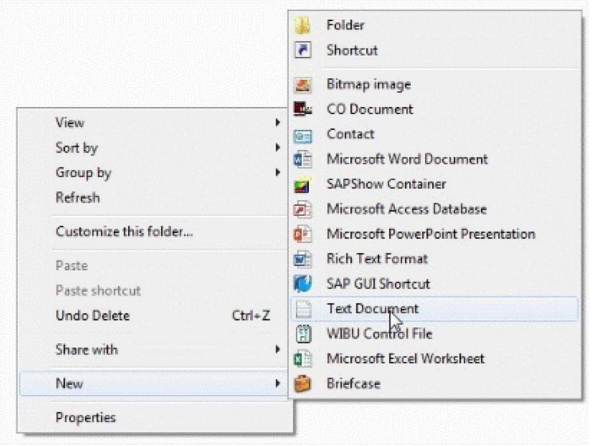
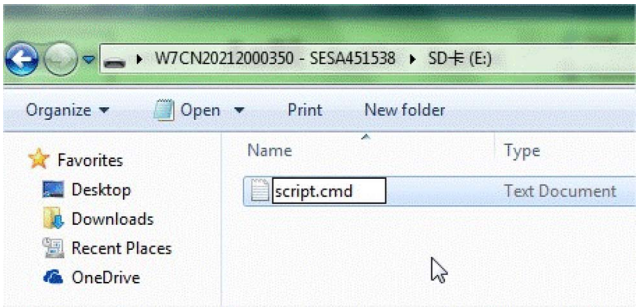
后配置管理

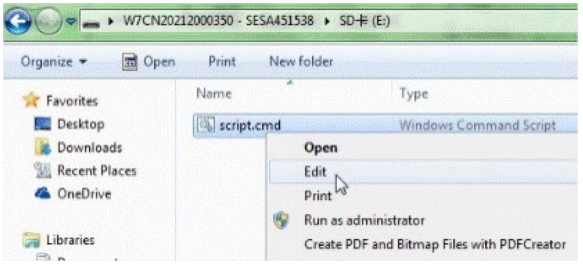
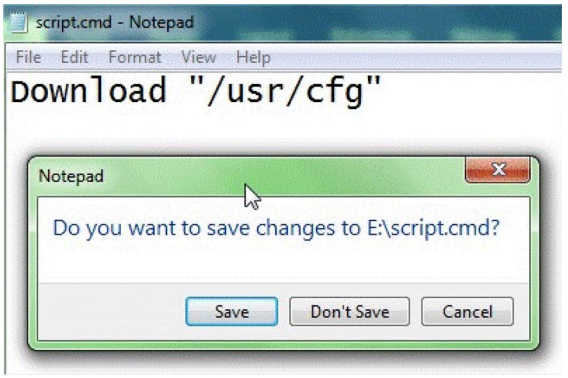
概述

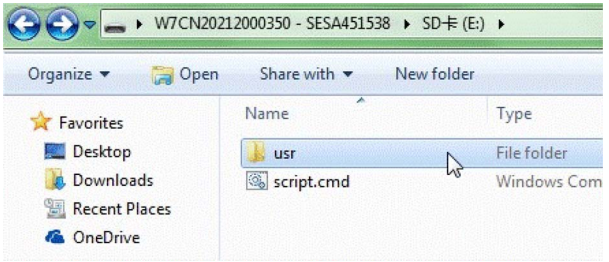
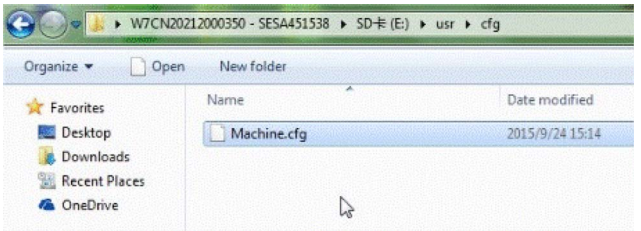
您可以使用 SD 卡添加、更改或删除控制器的后配置文件。

添加或更改后配置

下表介绍了如何添加或更改控制器后配置：

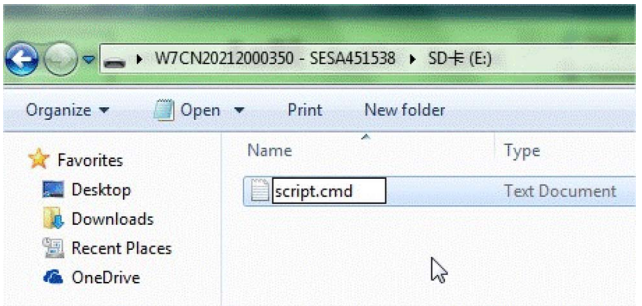
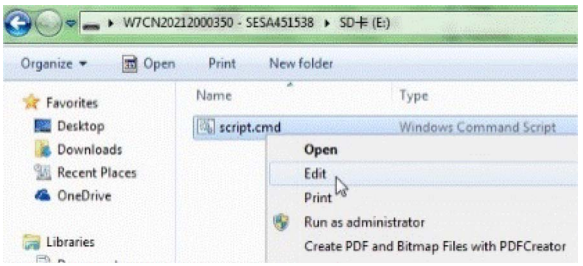
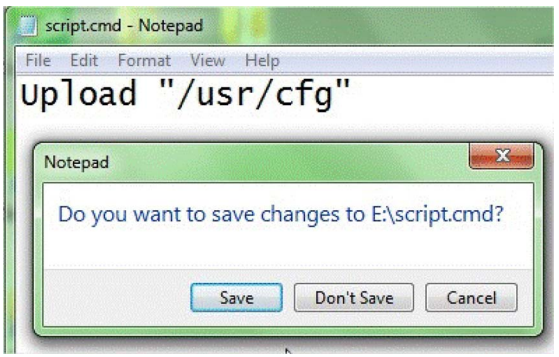
步骤	动作
1	在 SD 卡根目录中创建名为 script.cmd 的文件。  

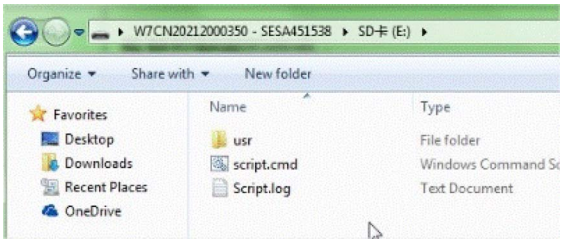
步骤	动作
2	编辑该文件并插入以下行： Download "/usr/cfg" ，然后保存该文件。  

步骤	动作
3	在 SD 卡根目录中创建文件夹 usr，并在该文件夹中创建子文件夹 cfg。 将后配置文件复制到 (Machine.cfg) 文件夹 \usr\cfg：   **注意：** SoMachine Basic 安装目录的 Firmwares & PostConfiguration\PostConfiguration\add_change\ 目录中提供了后配置文件示例和相关脚本。
4	如有必要，编辑 Machine.cfg 文件以配置后配置参数。
5	拔下控制器的电源。
6	将准备好的 SD 卡插入控制器。
7	恢复对控制器供电。 **结果**：开始下载后配置文件。在操作期间，逻辑控制器上的 **SD** 系统 LED 亮起。 **注意：** 操作进行期间，避免关闭逻辑控制器的电源。 **注意：** 在下载之前，将检查文件格式以及配置的所有通道、参数和值是否有效；如果检测到错误，则会中止下载。
8	等到操作结束（直到 **SD** LED 熄灭或闪烁）。 如果检测到错误，**SD** 和 **ERR** LED 会闪烁，检测到的错误将记录在 script.log 文件中。
9	重新通电或执行初始化命令以使新的后配置文件生效。

读取后配置文件

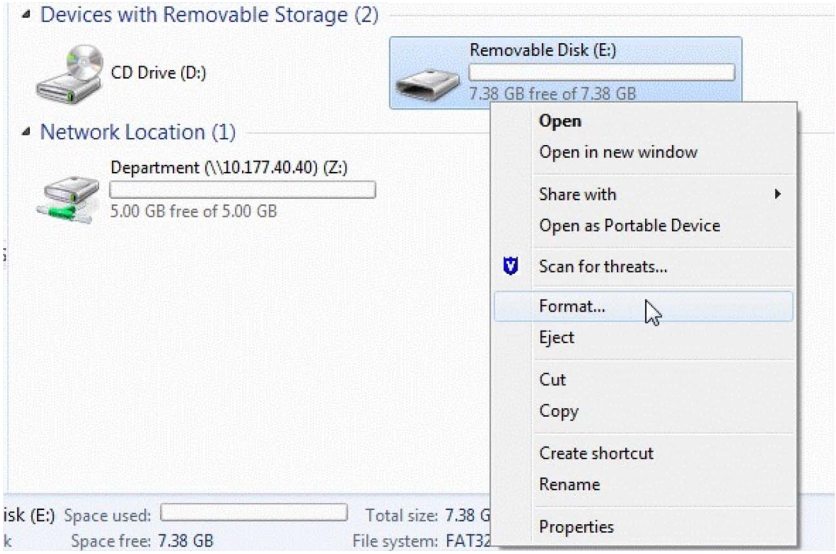
下表介绍如何读取控制器的后配置文件：

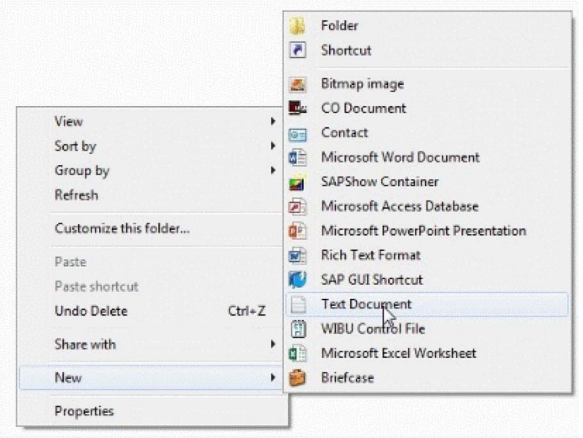
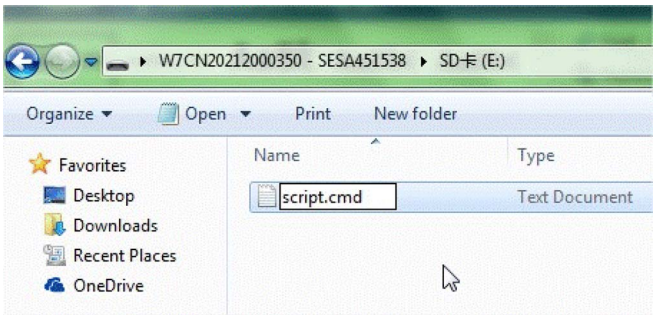
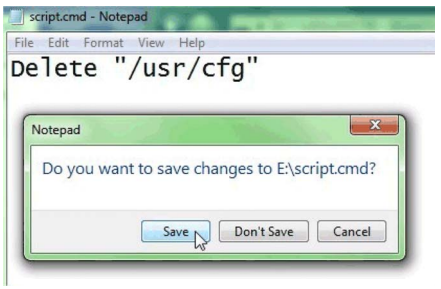
步骤	动作
1	使用 PC 上的文本编辑器在 SD 卡根目录中创建 script.cmd 文件。 
2	编辑该文件并插入以下行。然后，保存该文件： Upload "/usr/cfg"  
3	拔下控制器的电源。
4	将准备好的 SD 卡插入控制器。

步骤	动作
5	恢复对控制器供电。 **结果：** 开始复制后配置文件。在操作期间，逻辑控制器上的 **SD** 系统 LED 亮起。 **注意：** 操作进行期间，避免关闭逻辑控制器的电源。 **注意：** 应用程序备份过程的优先级较低，以便最大限度地减少对逻辑控制器程序和通讯性能的影响。
6	等到操作结束（直到 **SD** LED 熄灭或闪烁）。 如果检测到错误，**SD** 和 **ERR** LED 会闪烁，检测到的错误将记录在 script.log 文件中。 **结果：**结果：后配置文件保存在 SD 卡上。 

删除后配置文件

下表介绍如何删除控制器的后配置文件：

步骤	动作
1	将空的 SD 卡插入到 PC 中。 

步骤	动作
2	在 SD 卡根目录中创建名为 scr ipt. cmd 的文件。  
3	编辑该文件并插入以下行： Delete "/usr/cfg" 

步骤	动作
4	将 SoMachine Basic 安装目录的 Firmwares & PostConfiguration\PostConfiguration\remove\ 目录中可用的脚本文件复制到 SD 卡的根目录。
5	拔下控制器的电源。
6	将准备好的 SD 卡插入控制器。
7	恢复对控制器供电。 **结果：**删除后配置文件。在操作期间，逻辑控制器上的 **SD** 系统 LED 亮起。 **注意：** 操作进行期间，避免关闭逻辑控制器的电源。
8	等到操作结束（直到 **SD** LED 熄灭或闪烁）。 如果检测到错误，**SD** 和 **ERR** LED 会闪烁，检测到的错误将记录在 script.log 文件中。
9	使控制器重新通电或执行初始化命令以使应用参数生效。

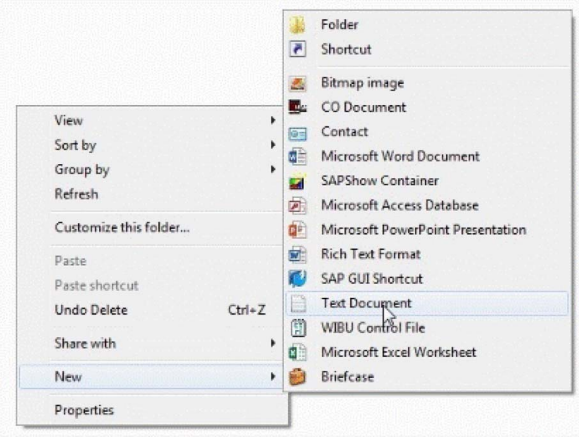
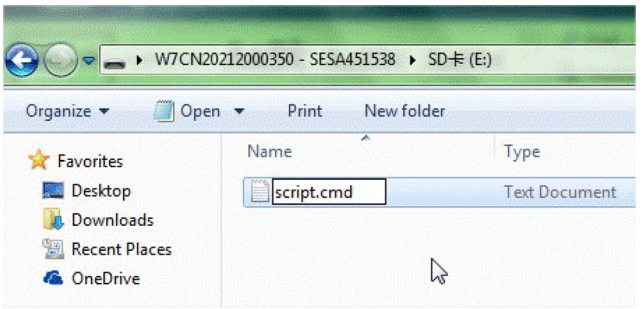
错误日志管理

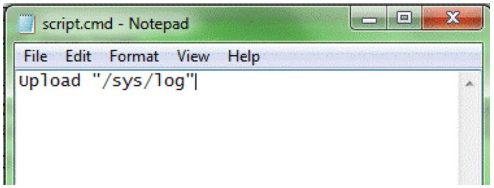
概述

您可以使用微型 SD 卡来备份或删除逻辑控制器的错误日志文件。

备份错误日志

下表介绍了如何在微型 SD 卡上备份逻辑控制器错误日志文件：

步骤	动作
1	使用 PC 上的文本编辑器在 SD 卡根目录中创建 script.cmd 文件。  

步骤	动作
2	编辑该文件并插入以下行： Upload "/sys/log" 
3	将准备好的微型 SD 卡插入逻辑控制器。
4	将逻辑控制器上的 RUN/STOP 开关切换到 STOP 位置。 **结果：**开始传输错误日志文件。在操作期间，逻辑控制器上的 **SD** 系统 LED 亮起。
5	等到操作结束（直到 **SD** LED 熄灭或闪烁）。 如果检测到错误，LED 会闪烁，检测到的错误将记录在 Script.log 文件中。 **结果：**错误日志文件 (PlcLog.csv) 保存在微型 SD 卡上。

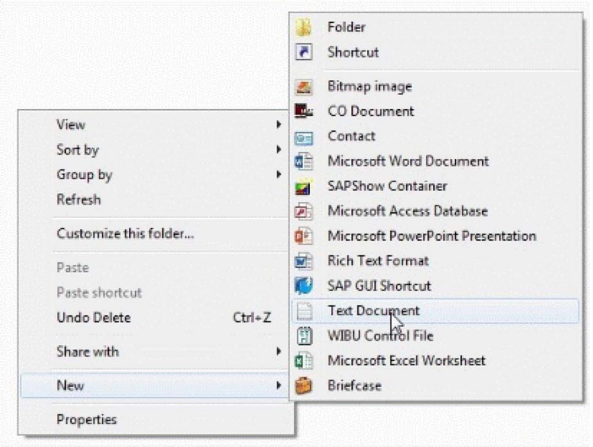
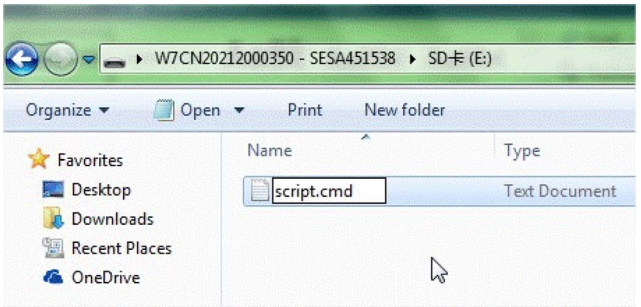
注意： 如果控制器中没有记录任何错误，SD LED 将保持关闭状态。RUN 和 ERR LED 将每秒闪烁一次。Script.log 文件中的信息如下所示：

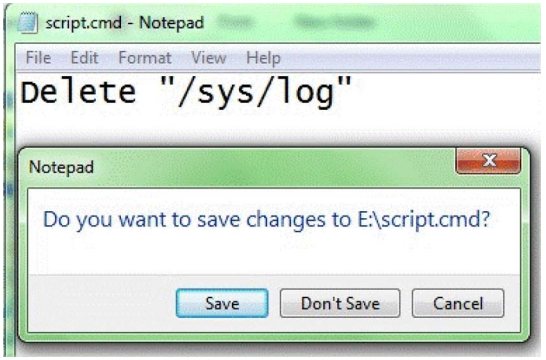
```
Ref = TM200CE24R   SN -20345592 01/01/00, 00:45:57 : # Start Script #
Ref = TM200CE24R   SN -20345592 01/01/00, 00:45:57 : Upload "/sys/log" - Log Area
is empty
Ref = TM200CE24R   SN -20345592 01/01/00, 00:45:57 : # End Script #
```

注意： 在该过程中，切勿拔下控制器的电源。将逻辑控制器的 RUN/STOP 开关切换到 STOP 位置。然后，插入 SD 卡。此时，SD LED 将开启。该过程完成后，SD LED 将关闭。

删除错误日志

下表介绍了如何删除逻辑控制器中的错误日志文件:

步骤	动作
1	使用 PC 上的文本编辑器在 SD 卡根目录中创建 script.cmd 文件。  

步骤	动作
2	编辑该文件并插入以下行： Delete "/sys/log" 
3	将准备好的微型 SD 卡插入逻辑控制器。
4	将逻辑控制器的 RUN/STOP 开关切换到 STOP 位置。 **结果：**开始删除错误日志文件。在操作期间，逻辑控制器上的 **SD** 系统 LED 亮起。
5	等到操作结束（直到 **SD** LED 熄灭或闪烁）。 如果检测到错误，LED 会闪烁，检测到的错误将记录在 Script.log 文件中。 **结果：**将错误日志文件 (PlcLog.csv) 从逻辑控制器中删除。

错误日志格式

逻辑控制器会提供一个错误列表，其中包含在日志存储器中最新检测到的 10 个错误。错误日志文件中的各个错误条目均由以下部分组成：

- 日期和时间
- 级别
- 上下文
- 错误代码

通过微型 SD 卡上载后，代码将如以下示例所示：

02/06/14, 12:04:01, 0x0111000100

下表介绍了十六进制错误所代表的意义：

分组	错误代码（十六进制）	错误描述	结果
常规	08000011xx	无效的硬件校准参数	以太网通道不工作 %SW118.bit10 设置为 0 **ERR** LED 闪烁
操作系统	0F01xxxxxx	检出操作系统错误	切换为已暂停状态
存储器管理	0F030009xx	检出内部存储器分配错误	切换为已暂停状态

分组	错误代码（十六进制）	错误描述	结果
SD 卡	010C001Bxx	访问 SD 卡时出错；操作时间超过内部超时时间（3000 毫秒）。	SD 卡操作被取消。
警戒时钟定时器	0104000Axx	逻辑控制器资源利用率超过 80% - 首次检测	信令显示看门狗超时： ● %s11 设置为 1ERR LED 闪烁
	0804000Bxx	逻辑控制器资源利用率超过 80% - 第二次连续检测	切换为已暂停状态
	0804000Cxx	任务警戒时钟定时器处于主任务状态	切换为已暂停状态
	0804000Dxx	任务警戒时钟定时器处于周期性任务状态	切换为已暂停状态
电池	0105000Exx	电池电量耗尽	信令显示电池电量耗尽： %S75 设置为 1 BAT LED 亮起
用户应用程序	0807000Fxx	应用程序与固件不兼容	切换为空状态
	08070010xx	检测到校验和错误	切换为空状态
以太网	010B0014xx	检出重复的 IP 地址	已指示 IP 重复 ● %SW62 设置为 1 ● %SW118.bit9 设置为 0。ERR LED 闪烁
嵌入式 I/O	010D0013xx	在受保护输出端上检出短路	信令显示过流： %SW139 设置为 1（视输出块而定） ERR LED 闪烁
读取非易失性存储器	01110000xx	检出读取错误 - 未找到文件	读取操作失败
	01110001xx	检出读取错误 - 逻辑控制器类型不正确	
	01110002xx	检出读取错误 - 页眉不正确	
	01110003xx	检出读取错误 - 区域描述符不正确	
	01110004xx	检出读取错误 - 区域描述符大小不正确	
写入非易失性存储器	01120002xx	检出写入错误 - 页眉不正确	写入操作不正确
	01120004xx	检出写入错误 - 区域描述符大小不正确	
	01120005xx	检出写入错误 - 消除失败	
	01120006xx	检出写入错误 - 页眉大小不正确	

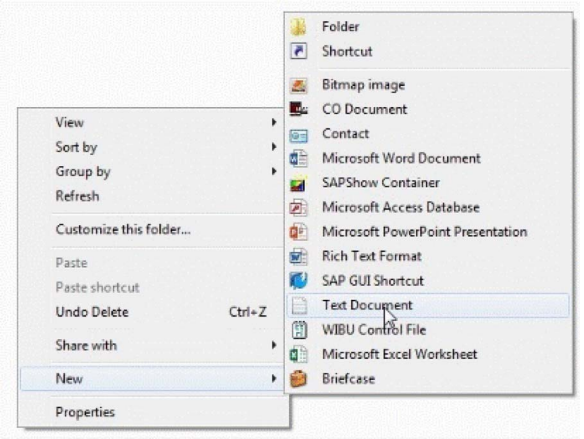
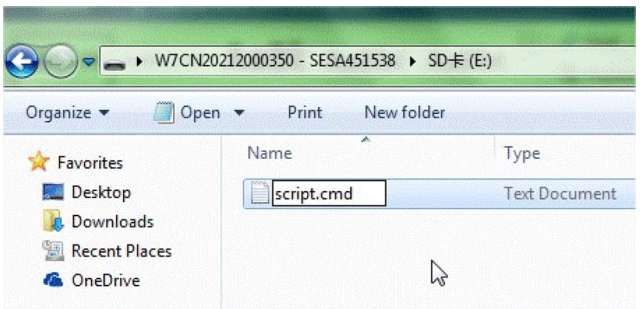
分组	错误代码（十六进制）	错误描述	结果
保持型变量	01130007xx	在保持型变量中检出校验和错误	无法恢复保持型变量
	01130008xx	在保持型变量中检出大小错误	
以太网 IP	01140012xx	以太网 IP 变量创建失败	无法创建变量，操作失败

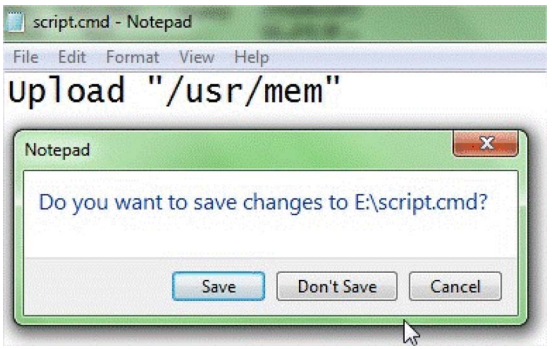
存储器管理：备份和恢复控制器存储器

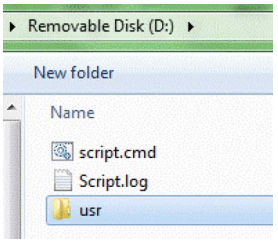
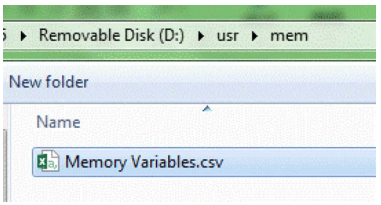
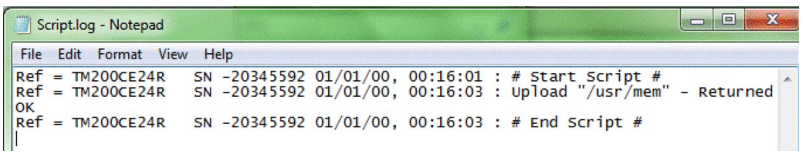
概述

您可以使用 SD 卡备份和恢复控制器存储器对象，或将存储器对象复制到另一控制器。

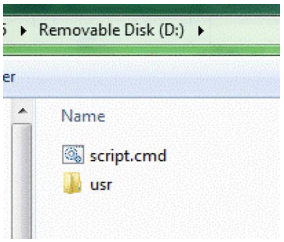
备份控制器存储器

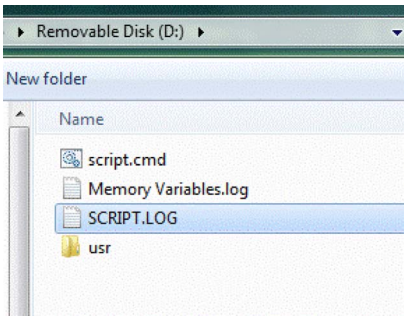
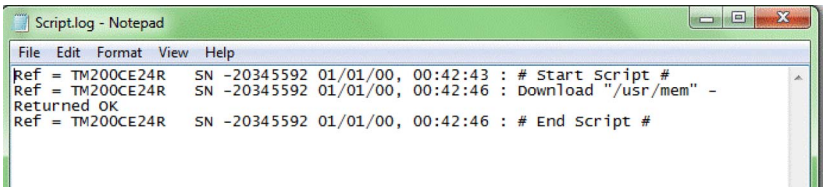
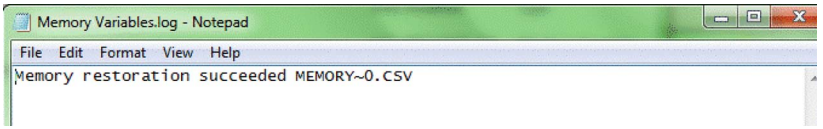
步骤	动作
1	使用 PC 上的文本编辑器在 SD 卡根目录中创建 script.cmd 文件。  

步骤	动作
2	编辑该文件并插入以下行： Upload "/usr/mem" 
3	将准备好的 SD 卡插入控制器。
4	将逻辑控制器的 RUN/STOP 开关切换到 STOP 位置。 **结果：** 开始复制存储器。在操作期间，逻辑控制器上的 **SD** 系统 LED 亮起。 **注意：** 操作进行期间，避免关闭逻辑控制器的电源。 **注意：** 存储器备份过程的优先级较低， 以便最大限度地减少对逻辑控制器程序和通讯性能的影响。

步骤	动作
5	等到操作结束（直到 **SD** LED 熄灭或闪烁）。 如果检测到错误，**SD** 和 **ERR** LED 会闪烁，检测到的错误将记录在 Script.log 文件中。 **结果：** 存储器文件 (*.csv) 保存在 SD 卡上。   

恢复控制器存储器或复制到另一个控制器

步骤	动作
1	使用文本编辑器编辑 SD 卡根文件夹中的 script.cmd 文件。 
2	编辑该文件并插入以下行： Download "/usr/mem"
3	将准备好的 SD 卡插入控制器。
4	将控制器的 RUN/STOP 开关切换到 STOP 位置。 **结果：**开始复制存储器文件。在操作期间，逻辑控制器上的 **SD** 系统 LED 亮起。 **注意：** 操作进行期间，避免关闭逻辑控制器的电源。

步骤	动作
5	等到操作结束（直到 **SD** LED 熄灭或闪烁）。 如果检测到错误，**SD** 和 **ERR** LED 会闪烁，检测到的错误将记录在 Script.log 文件中。   
6	删除 SD 卡，然后重新启动控制器。

第III部分
对 M100/M200 可编程控制器 进行编程

概述

本部分提供有关 M100/M200 可编程控制器 特定的系统对象和 I/O 对象的信息。这些对象显示在**编程**选项卡中。

有关所有其他对象的描述，请参阅《SoMachine Basic 通用功能库指南》。

本部分包含了哪些内容？

本部分包括以下各章：

章	章节标题	页
10	如何使用源代码示例	219
11	I/O 对象	223
12	功能块	229
13	网络对象	297
14	脉冲串输出 (%PTO)	307
15	频率发生器	409
16	PID 功能	413
17	系统对象	453

第10章
如何使用源代码示例

如何使用源代码示例

概述

除非明确提出，否则本手册中包含的源代码示例对梯形图和指令列表编程语言均有效。整个示例可能需要多个梯级。

可转换性过程

要获得对应的梯形图源代码，请执行以下操作：

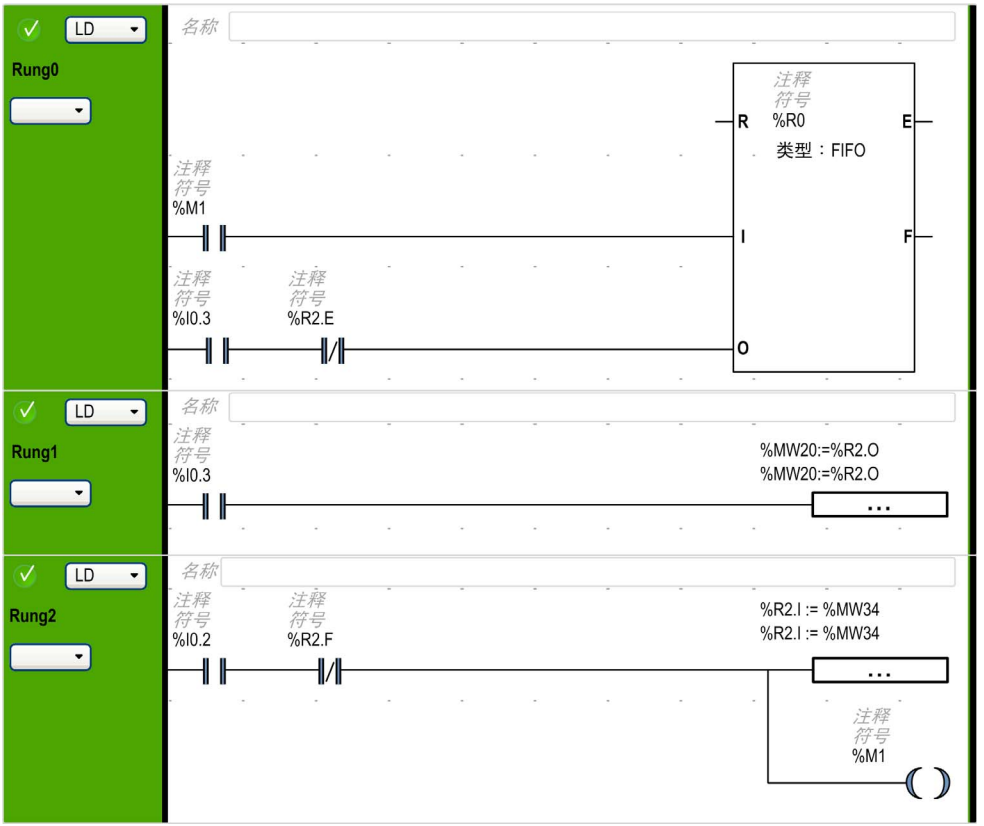
步骤	动作
1	选择并复制 (**Ctrl+C**) 本手册中所示示例程序的第一个梯级的源代码。
2	在 SoMachine Basic 中，单击工具栏上的 创建新的梯级。
3	在此梯级中，单击 **LD > IL** 按钮以显示指令列表源代码。
4	选择行号 **0000**，然后右键单击并选择**粘贴指令**以将源代码粘贴到梯级中： **注意：**如果通过在缺省LD操作符之前插入行来粘贴指令，那么务必从梯级的最后一行中删除 **LD** 指令。
5	单击 **IL > LD** 按钮以显示梯形图源代码。
6	对示例程序中的任何其他梯级重复上述步骤。

示例

指令列表程序：

梯级	源代码
0	BLK %R0 LD %M1 I LD %I0.3 ANDN %R2.E O END_BLK
1	LD %I0.3 [%MW20:=%R2.O]
2	LD %I0.2 ANDN %R2.F [%R2.I:=%MW34] ST %M1

相应的梯形图：


LD
Rung0
名称
注释
符号
%R0
R
E
类型：FIFO
注释
符号
%M1
I
F
注释
符号
%I0.3
注释
符号
%R2.E
O
LD
Rung1
名称
注释
符号
%I0.3
%MW20:=%R2.O
%MW20:=%R2.O
...
LD
Rung2
名称
注释
符号
%I0.2
注释
符号
%R2.F
%R2.I := %MW34
%R2.I := %MW34
...
注释
符号
%M1

第11章
I/O 对象

本章包含了哪些内容？

本章包含了以下主题：

主题	页
数字量输入 (%I)	224
数字量输出 (%Q)	225
模拟量输入 (%IW)	226
模拟量输出 (%QW)	227

数字量输入 (%I)

简介

数字量输入位对象是逻辑控制器上数字量输入的映射。

显示数字量输入属性

遵循以下步骤以显示数字量输入属性：

步骤	动作
1	选择**编程**窗口左侧区域的**工具**选项卡。
2	单击 **I/O 对象 → 数字量输入**。 **结果**：屏幕中显示数字量输入属性。

数字量输入属性

下表介绍了数字量输入的每个属性：

参数	可编辑	值	缺省值	描述
已使用	否	True/False	False	指示是否在程序中引用输入通道。
地址	否	%I0.i	–	显示控制器上数字量输入的地址，其中 i 表示通道编号。如果控制器具有 n 个数字量输入通道，则 i 的值以 0...n-1 形式提供。 例如，%I0. 2 是逻辑控制器的数字量输入通道号 2 中的数字量输入。
符号	是	–	–	与此地址关联的符号。 在**符号**列中双击， 然后键入要与此输入关联的符号的名称。 如果某个符号已经存在，则可以右键单击**符号**列，然后选择**搜索并替换**，在整个程序和/或程序注释中查找并替换出现此符号的地方。
注释	是	–	–	与此地址关联的注释。 双击**注释**列，然后键入要与此通道关联的可选注释。

数字量输出 (%Q)

简介

数字量输出位对象是逻辑控制器上数字量输出的映射。

显示数字量输出属性

遵循以下步骤以显示数字量输出属性：

步骤	动作
1	选择**编程**窗口左侧区域的**工具**选项卡。
2	单击 **I/O 对象 → 数字量输出**。 **结果**：屏幕中显示数字量输出属性。

数字量输出属性

下表介绍了数字量输出的每个属性：

参数	可编辑	值	缺省值	描述
已使用	否	True/False	False	指示是否在程序中引用输出通道。
地址	否	%Q0.i	–	显示控制器上数字量输出的地址，其中 i 表示通道编号。 如果控制器具有 n 个数字量输出通道，则 i 的值以 0...n-1 形式提供。 例如，%Q0. 3 是逻辑控制器的数字量输出通道号 3 中的数字量输出。
符号	是	–	–	与此地址关联的符号。 在**符号**列中双击，然后键入要与此输出关联的符号的名称。 如果某个符号已经存在，则可以右键单击**符号**列，然后选择**搜索并替换**，在整个程序和/或程序注释中查找并替换出现此符号的地方。
注释	是	–	–	与此地址关联的注释。 双击**注释**列，然后键入要与此通道关联的可选注释。

模拟量输入 (%IW)

简介

模拟量输入字对象是连接到逻辑控制器的模拟量信号的数字值。

在扩展板 TMCR2AI2、TMCR2TI2 和 TMCR2AM3 中内置了两个 0-10V 的模拟量输入。嵌入的模拟量输入使用一个 10 位精度转换器，因此每个增量大约为 10 mV ($10V/2^{10}-1$)。系统检测到值 1023 后，将认为通道已饱和。

有关详细信息，请参阅 M100/M200 硬件指南 *(参见 Modicon M100/M200 可编程控制器, 硬件指南)*。

显示模拟量输入属性

遵循以下步骤以显示模拟量输入属性：

步骤	动作
1	选择**编程**窗口左侧区域的**工具**选项卡。
2	单击 **I/O 对象 → 模拟量输入**。 **结果**：屏幕中显示模拟量输入属性。

模拟量输入属性

下表描述了模拟量输入的每个属性：

参数	可编辑	值	缺省值	描述
已使用	否	True/False	False	指示是否在程序中引用输入通道。
地址	否	%IW0.i	–	显示控制器上内置模拟量输入的地址， 其中 i 表示通道编号。 如果控制器具有 n 个模拟量输入通道， 则 i 的值以 0...n-1 形式提供。 例如，%IW0.1 是逻辑控制器的模拟量输入通道号 1 中的模拟量输入。
		%IW0.x0y	–	显示扩展板上模拟量输出通道的地址， 其中 x 是扩展板编号，y 是通道编号。
符号	是	–	–	与此地址关联的符号。 在**符号**列中双击，然后键入要与此输入关联的符号的名称。 如果某个符号已经存在，则可以右键单击**符号**列，然后选择**搜索并替换**， 在整个程序和/或程序注释中查找并替换出现此符号的地方。
注释	是	–	–	与此地址关联的注释。 双击**注释**列， 然后键入要与此地址关联的注释。

模拟量输出 (%QW)

简介

模拟量输出字对象是使用扩展板从逻辑控制器中接收的模拟量信号的数字值。

扩展板 TMCR2AQ2C 和 TMCR2AQ2V 中分别内置了两个 0-10 V 模拟量输出和两个 4-20 mA 模拟量输出。

扩展板 TMCR2AM3 中内置了一个 0-5 V/0-10 V 模拟量电压输出或 4-20 mA 模拟量电流输出。

有关详细信息，请参阅 Modicon M100/M200 可编程控制器 硬件指南 *(参见 Modicon M100/M200 可编程控制器, 硬件指南)*。

显示模拟量输出属性

遵循以下步骤以显示模拟量输出属性：

步骤	动作
1	选择**编程**窗口左侧区域的**工具**选项卡。
2	单击 **I/O 对象** → **模拟量输出**。 **结果**：屏幕中显示模拟量输出属性。

模拟量输出属性

下表描述了模拟量输出的每个属性：

参数	可编辑	值	缺省值	描述
已使用	否	True/False	False	指示是否在程序中引用输出通道。
地址	否	%QW0.x0y	–	显示扩展板上模拟量输出通道的地址， 其中 x 是扩展板编号，y 是通道编号。
符号	是	–	–	与此地址关联的符号。 在**符号**列中双击，然后键入要与此输出关联的符号的名称。 如果某个符号已经存在，则可以右键单击**符号**列，然后选择**搜索并替换**，在整个程序和/或程序注释中查找并替换出现此符号的地方。
注释	是	–	–	与此地址关联的注释。 双击**注释**列， 然后键入要与此地址关联的注释。

第12章
功能块

本章包含了哪些内容？

本章包含了以下部分：

节	主题	页
12.1	快速计数器 (%FC)	230
12.2	高速计数器 (%HSC)	236
12.3	脉冲 (%PLS)	255
12.4	驱动器功能块 (%DRV)	262
12.5	脉冲宽度调制 (%PWM)	291

第12.1节
快速计数器 (%FC)

使用快速计数器功能块

本节介绍如何使用Fast Counter 功能块.

本节包含了哪些内容？

本节包含了以下主题：

主题	页
描述	231
配置	233
编程示例	235

描述

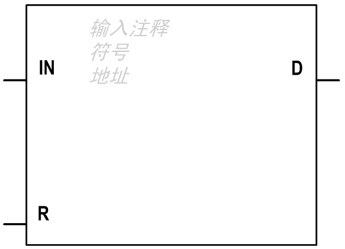
简介

Fast Counter 功能块 1123 用作加计数器或减计数器。在单字或双字计算模式下，它可以对最高频率为 5 kHz 的数字量输入的上升沿进行计数。由于 Fast Counter 功能块受特定硬件中断的管理，因此维持最大频率采样比例可能会根据特定的应用程序和硬件配置而变化。

Fast Counter 功能块 %FC0、%FC1、%FC2 和 %FC3 分别使用专用输入 %I0.2、%I0.3、%I0.4 和 %I0.5。这些位并非保留供其专用。其分配必须考虑到这些专用资源的其他功能块的使用。

示意图

本示意图为Fast Counter功能块：



输入

Fast Counter 功能块具有以下输入：

标签	描述	值
IN	启用	在状态 1，根据应用到物理输入的脉冲更新值。 在状态 0，值将保持为其上次的值。
R	复位（可选）	用于初始化功能块。 在状态 1： ● 会考虑 %FC.P 或 %FC.PD 值。 ● 如果配置为加计数器，则当前值复位为 0，如果配置为减计数器，则值设置为 %FC.P 或 %FC.PD。 ● “完成”位 %FC.D 将被设置回其缺省值。

输出

Fast Counter 功能块具有以下输出：

标签	描述	值
D	Done (%FCi.D)	在下列情况下，此位设置为 1： ● 配置为加计数器时，%FCi.V 或 %FCi.VD 达到预设值 %FCi.P 或 %FCi.PD。 ● 或者配置为减计数器时 %FCi.V 或 %FCi.VD 达到 0。 此只读位只能通过将 %FCi.R 设为 1 进行复位。

配置

参数

要配置参数，请执行配置功能块过程 *(参见 SoMachine Basic, 通用功能库指南)*，并阅读 SoMachine Basic 操作指南中存储器分配模式 *(参见 SoMachine Basic, 操作指南)*的描述。

Fast Counter 功能块具有以下参数：

参数	描述	值
已使用	已使用的地址	如果选择此参数，则当前在程序中使用此地址。
地址	%FCi Fast Counter 地址	实例标识符，其中 i 是从 0 到此逻辑控制器上可用的对象数。 有关 Fast Counters 的最大数量，请参阅“对象的最大数量”表 (参见第 *32* 页)。
输入	%I0.i	与此功能块实例对应的专用输入。 %I0.2...%I0.5
符号	符号	与此对象关联的符号。有关详细信息，请参阅 SoMachine Basic 操作指南（定义和使用符号）。
已配置	是加计数还是减计数	设置为以下值之一： ● **未使用** ● **加计数器** ● **减计数器**
预设	预设值 （%FCi.P 或 %FCi.PD）	初始值可能设置为： ● 使用关联对象 %FCi.P，从 0 到 65535（在单字模式下）， ● 在双字模式下使用关联对象 %FCi.PD，从 0 到 4294967295。
Double Word	双字模式	如果已选择，则使用双字模式。否则，使用单字模式。
注释	注释	可以将可选注释与此对象相关联。 双击**注释**列并键入注释。

对象

Fast Counter 功能块与以下对象相关联：

对象	描述	值
%FCi.V %FCi.VD	当前值	当前值根据所选的加或减计数功能递增或递减。对于加计数，当前计数值更新，在单字模式 (%FCi.V) 下可以达到 65535，在双字模式 (%FCi.VD) 下可以达到 4294967295。对于减计数，当前值为预设值 %FC.P 或 %FC.PD，并且可以减到零。
%FCi.P %FCi.PD	预设值	只有在 R 输入激活的情况下，才会考虑新的预设值。 请参见上表中的“参数”说明。
%FCi.D	完成	请参见上面“输出”表中的说明。

操作

下表介绍了 Fast Counter 功能块操作的主要过程：

操作	动作	结果
加计数	在加计数输入中出现上升沿。	当前值 %FCi.V 递增 1 个单位。
	在达到预设值 %FCi.P 或 %FCi.PD 时。	“完成”输出位 %FCi.D 设置为 1。
减计数	在减计数输入中出现上升沿。	当前值 %FCi.V 减少一个单位。
	当值为 0 时。	“完成”输出位 %FCi.D 设置为 1。

特殊情况

下表罗列了 Fast Counter 功能块的特殊操作情况：

特殊情况	描述
冷重启 (%S0=1) 的结果	使用配置值或用户应用重置 Fast Counter 属性。 请参阅系统位 (%S) (参见第 *454* 页)。
控制器停止的结果	Fast Counter 在控制器设置为已停止状态时停止计数，并在控制器恢复正在运行状态时重新计数。计数器从进入已停止状态前所使用的最后一个值开始重新计数。

编程示例

简介

在此示例中，当 %I0.1 设置为 1 时，应用程序实现加计数到预设值 5000。%FC1 的输入是专用输入 %I0.3。达到预设值时，%FC1.D 设置为 1 并一直保留该状态，直到 %FC1.R 得到 AND 结果（对 %I0.2 和 %M0）的命令。

编程

本示例为 Fast Counter 功能块的特殊操作情况的列表：

梯级	指令
0	BLK %FC1 LD %I0.1 IN LD %I0.2 AND %M0 R OUT_BLK LD D ST %Q0.0 END_BLK

注意： 请参阅可转换性过程 (参见第 *217* 页)以获取等效梯形图。

第12.2节
高速计数器 (%HSC)

使用高速计数器功能块

本节介绍如何使用High Speed Counter 功能块.

本节包含了哪些内容？

本节包含了以下主题：

主题	页
描述	237
计数模式下的高速计数器	241
One-shot 计数模式	249
Modulo-loop 计数模式	250
频率计模式下的高速计数器	252

描述

简介

High Speed Counter 功能块 *11123* 可由 SoMachine Basic 配置以执行以下任一功能：

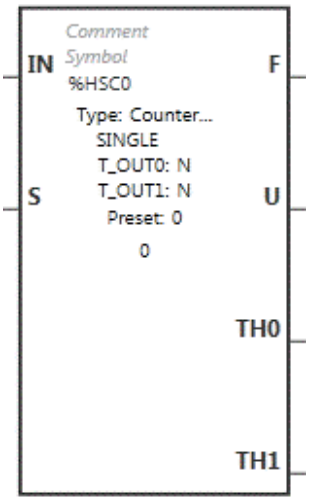
- Single Phase
- Dual Phase [Pulse / Direction]
- Dual Phase [Clock Wise / Counter Clock Wise]
- Dual Phase [Quadrature X1]
- Dual Phase [Quadrature X2]
- Dual Phase [Quadrature X4]
- Frequency Meter

对于所有的计数模式，High Speed Counter 功能块最高可以采用 100 kHz 的频率进行工作，单字模式下其范围为 65535 以内，双字模式下其范围为 4294967295 以内。

High Speed Counter 功能块使用专用输入以及辅助输入和输出。有关输入和输出的详细信息，请参阅 M100/M200 可编程控制器 - 硬件指南。

必须先使用**高速计数器助手**在**配置**选项卡中初始化High Speed Counter功能，之后才能使用该功能块的实例。请参阅配置高速计数器 (参见第 *90* 页)。

图形表示形式



输入

High Speed Counter 功能块具有以下输入：

标签	描述	值
IN	启用（必需） 处于状态 1 时，将启用计数功能或频率测量。 处于状态 0 时，当前值将保持为其上次的值。	0 或 1
S	预设输入。 在上升沿时，将使用预设值初始化预设值。： 另外，还会初始化阈值输出的操作，并考虑对属性窗口或程序中设置的阈值所做的任何用户修改。	0 或 1

High Speed Counter功能块与以下输入对象相关联：

对象	类型	描述	值
%HSCi.P %HSCi.PD	WORD DOUBLE WORD	预设值 **注意：** 当配置为模数回路模式时， 为模数值。	请参见辅助输入 (参见第 *242* 页)。
%HSCi.S0 %HSCi.S0D	WORD DOUBLE WORD	阈值 0	请参见计数模式下的输出阈值 (参见第 *241* 页)。
%HSCi.S1 %HSCi.S1D	WORD DOUBLE WORD	阈值 1	请参见计数模式下的输出阈值 (参见第 *241* 页)。
%HSCi.T	WORD	时基	请参见频率计模式下的高速计数器 (参见第 *99* 页)。
%HSCi.R	BOOL	启用反射输出 0	处于状态 1 时，将启用反射输出 0。
%HSCi.S	BOOL	启用反射输出 1	处于状态 1 时，将启用反射输出 1。

输出

High Speed Counter 功能块具有下列输出：

标签	描述	值
F	溢出 如果发生算术溢出，则设置为 1。	0 或 1
U	计数方向 由系统设置，Dual Phase计数功能将使用该位来指示计数方向。	0：减计数 1：加计数
TH0	阈值位 0 当前值大于或等于阈值 S0 (%HSCi.S0) 时设置为 1。 请仅在程序中对该位进行一次测试，因为该位将实时更新。 用户应用程序负责验证该值在使用时的有效性。	0 或 1

标签	描述	值
TH1	阈值位 1 当前值大于或等于阈值 S1 (%HSCi.S1) 时设置为 1。 请仅在程序中对该位进行一次测试，因为该位将实时更新。	0 或 1

High Speed Counter功能块与以下输出对象相关联：

对象	类型	描述	值
%HSCi.V %HSCi.VD	WORD DOUBLE WORD	当前值	请参见计数模式下的高速计数器 (参见第 *241* 页)。
%HSCi.C %HSCi.CD	WORD DOUBLE WORD	捕捉值	请参见辅助输入 (参见第 *242* 页)。
%HSCi.U	BOOL	计数方向	0：减计数 1：加计数
%HSCi.F	BOOL	溢出	0：未溢出 1：计数器溢出

属性

High Speed Counter功能块具有以下属性：

属性	值	描述
已使用	已激活/已停用的复选框	指示地址是否正在使用。
地址	%HSCi，其中 i 为 0 至 3（取决于已配置的计数器的类型）。	i 是实例标示符。 有关可用 High Speed Counter 对象的数量，请参阅对象的最大数量 (参见第 *32* 页)。
符号	用户定义的文本	可唯一标识该对象的符号。有关详细信息，请参阅《SoMachine Basic 操作指南（定义和使用符号）》。
预设	● 对于 %HSCi.P，为 0 到 65535 ● 对于 %HSCi.PD，为 0 到 4294967295	预设值，用于初始化 HSC 当前值（%HSCi.P、%HSCi.PD）。 对Frequency Meter无效。
S0	● 对于 %HSCi.S0，为 1 到 65535 ● 对于 %HSCi.S0D，为 1 到 4294967295	阈值 0 用作当前值的比较值。 S0 的值必须小于 S1 (%HSCi.S1)。
S1	● 对于 %HSCi.S1，为 2 到 65535 ● 对于 %HSCi.S1D，为 2 到 4294967295	阈值 1 用作当前值的比较值。 S1 的值必须大于 S0 (%HSCi.S0)。
时基	对于 %HSCi.T，为 100 毫秒或 1 s	频率测量时基
注释	用户定义的文本	与该对象关联的注释。

特殊情况

下表介绍了对 High Speed Counter 功能块进行编程时的特殊情况列表：

特殊情况	描述
冷重启 (%S0=1) 的结果	使用程序配置的值复位High Speed Counter属性。
控制器停止的结果	High Speed Counter 停止其功能，且输出保持在其当前状态。 **注意：** 在控制器停止后，仅当任务的故障预置行为配置为保持输出的值时，反射输出才会保持其当前状态。有关配置故障预置行为的更多信息，请参阅故障预置行为。

计数模式下的高速计数器

简介

High Speed Counter 功能块的最大工作频率为 100 kHz，单字模式下的范围是 0 到 65535，双字模式下的范围是 0 到 4294967295。

脉冲按照以下方式进行计数：

功能	描述	输入类型	%HSC0	%HSC1	%HSC2	%HSC3
Single Phase	脉冲将应用到与**脉冲输入**关联的物理输入。	**脉冲输入**	%I0.0	%I0.6	–	%I0.7
Dual Phase [Pulse / Direction]	脉冲将应用到与**脉冲输入**关联的物理输入，当前操作（加计数/减计数）将由**方向输入**的状态来指定。	**脉冲输入**	%I0.0	%I0.6	–	–
		方向输入	%I0.1	%I0.7	–	–
Dual Phase [Clock Wise / Counter Clock Wise]	这些脉冲应用于关联到**顺时针输入**和**逆时针输入**的物理输入。	**顺时针输入**	%I0.0	%I0.6	–	–
		逆时针输入	%I0.1	%I0.7	–	–
Dual Phase [Quadrature X1]	编码器的 2 个相位将应用到与**脉冲输入相位 A** 和**脉冲输入相位 B** 关联的物理输入。	**脉冲输入相位 A**	%I0.0	%I0.6	–	–
		脉冲输入相位 B	%I0.1	%I0.7	–	–
Dual Phase [Quadrature X2]	编码器的 2 个相位将应用到与**脉冲输入相位 A** 和**脉冲输入相位 B** 关联的物理输入。	**脉冲输入相位 A**	%I0.0	%I0.6	–	–
		脉冲输入相位 B	%I0.1	%I0.7	–	–
Dual Phase [Quadrature X4]	编码器的 2 个相位将应用到与**脉冲输入相位 A** 和**脉冲输入相位 B** 关联的物理输入。	**脉冲输入相位 A**	%I0.0	%I0.6	–	–
		脉冲输入相位 B	%I0.1	%I0.7	–	–

输出阈值

计数期间，会将当前值与两个阈值进行比较：%HSCi.S0 或 %HSCi.S0D 以及 %HSCi.S1 或 %HSCi.S1D。

在单字模式中，无论**预设**输入的值为何，都会考虑这些阈值的修改。

在双字模式中，不考虑安全动态数据表中的阈值修改。但无论**预设**输入的值为何，都会考虑应用程序中的修改。

阈值修改保存在逻辑控制器中（%HSCi.S0、%HSCi.S1、%HSCi.S0D 和 %HSCi.S1D 对象），但不在 SoMachine Basic 的**配置**窗口中保存。

根据比较结果，位对象（%HSCi.TH0 和 %HSCi.TH1）将进行如下设置：

- 如果当前值大于或等于对应阈值，则设置为 1。
- 如果当前值小于对应阈值，则复位为 0。

可以对物理反射输出进行配置，以便根据阈值和计数器当前值的比较结果进行不同的响应。

注意： 可以配置 0、1 或 2 个反射输出。

有关反射输出的详细配置信息，请参阅配置单相和双相 (参见第 *94* 页)。

%HSCi.U 是功能块的输出；它给出关联计数器变化的方向（1 为加计数，0 为减计数）。

辅助输入

计数操作会在脉冲的上升沿上进行，并且只有在启用了计数功能块的情况下才能进行。

在计数模式下有两个可选输入：**捕捉输入**和**预设输入**：

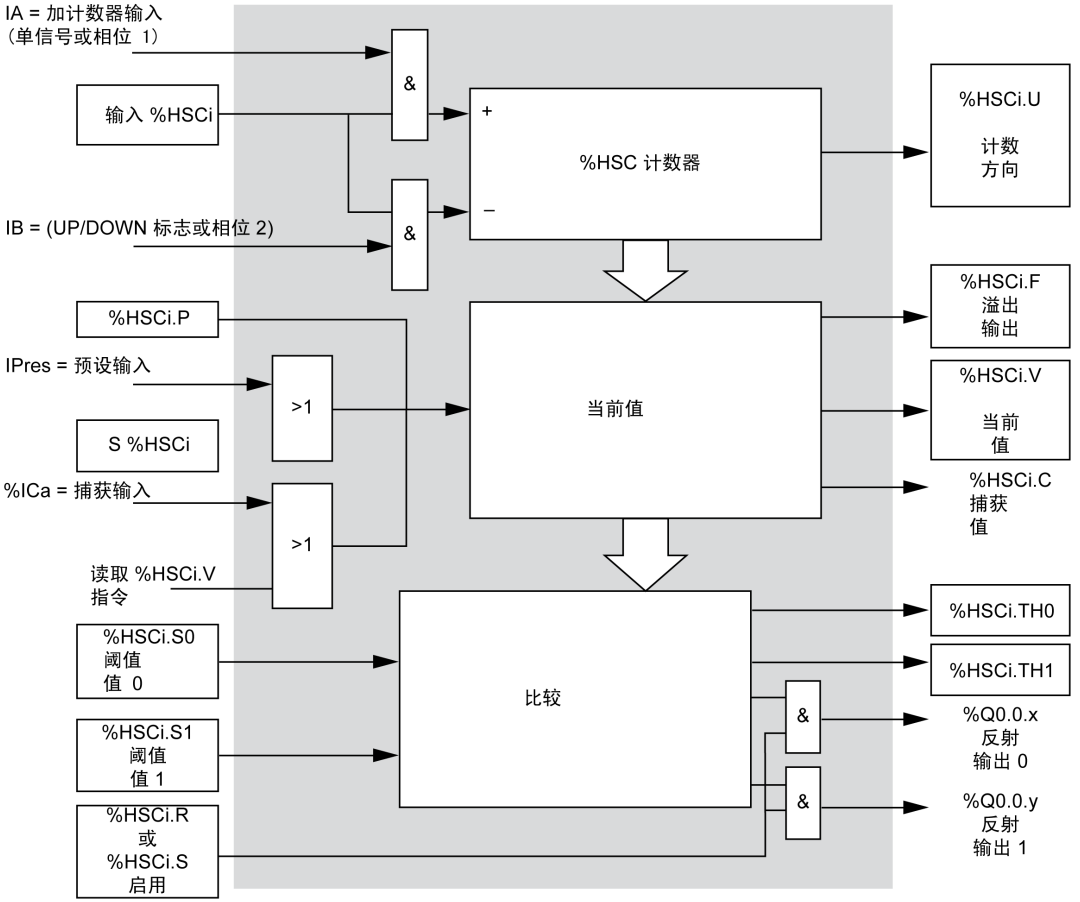
- **捕捉输入**用于捕捉当前值（%HSCi.V 或 %HSCi.VD），并将其存储在 %HSCi.C 或 %HSCi.CD 中。在适用的情况下，针对 %HSC0 将捕捉输入指定为 %I0.3，针对 %HSC1 将其指定为 %I0.4。
- **预设输入**使用下列的当前值对 %HSCi.V或 %HSCi.VD 值初始化：
  - 双相 [积分 X1]
  - 双相 [积分 X2.
  - 双相 [积分 X4]
  - 正在执行减计数的双相 [脉冲/方向]

  预设输入将下面的值重置为 0：
  - 单相
  - 正在执行减计数的双相 [脉冲/方向]

注意： %HSCi.F 也设置为 0。**预设输入**会被指定为 %I0.2（对于 %HSC0）和/或 %I0.5（对于 %HSC1）。

操作

以下示意图为单字模式下的计数模式操作图（在双字模式下，请使用双字功能变量）：



注意： 反射输出的管理独立于控制器循环时间。

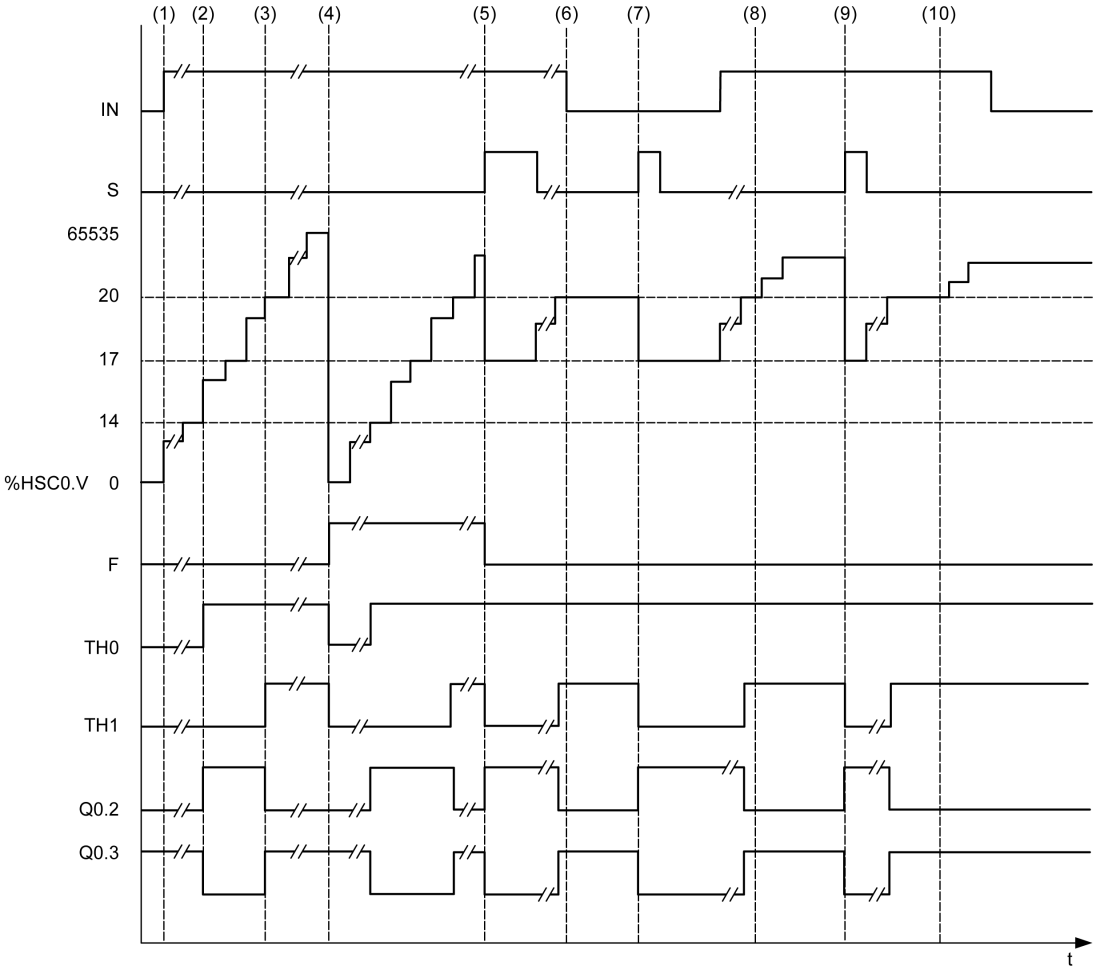
Single Phase 时序图

反射输出配置示例：

反射输出	值 < %HSC0.S0	%HSC0.S0 <= 值 < %HSC0.S1	值 >= %HSC0.S1
%Q0. 4	–	X	–
%Q0. 5	X	–	X

时序图:

%HSC0.P = 17
%HSC0.S0 = 14
%HSC0.S1 = 20



(1) IN 设置为 1：激活计数功能（%HSC0.U = 1，因为 %HSC0 为加计数器）
(2) %Q0.4（反射输出）和 TH0 设置为 1
(3) TH1 设置为 1
(4) 达到最大值，因此在下一个计数时，%HSC0.V 复位为 0，F 设置为 1
(5) S 设置为 1，当前值 %HSC0.V 设置为预设值.
(6) 禁止当前功能，同时将 IN 设置为 0
(7) 禁止功能，同时将 S 设置为 1，因此当前值设置为预设值 17
(8) 更改阈值 S1 为 17
(9) S 设置为 1，因此将在下一次计数时赋予 S1 的新值
(10) 捕捉输入设置为 1，因此 %HSC0.C = 20

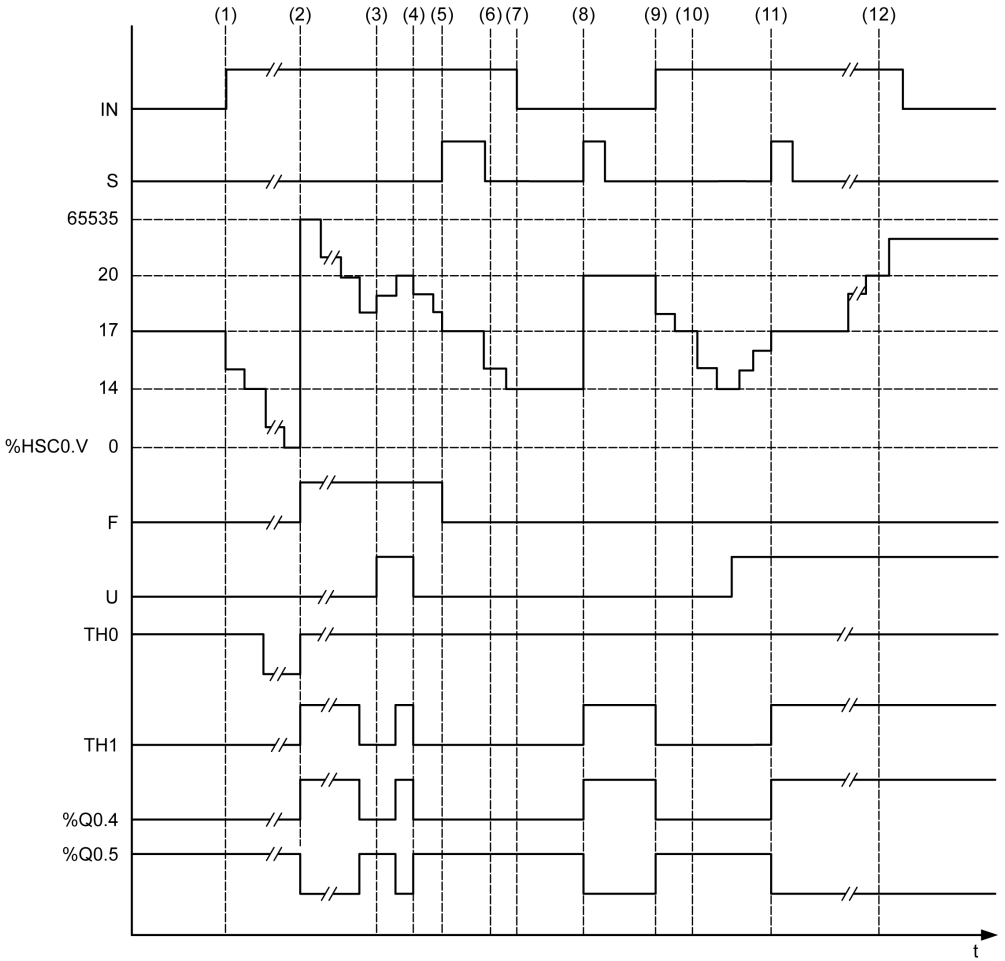
Dual Phase [Pulse / Direction] 时序图

反射输出配置示例：

反射输出	值 < %HSC0.S0	%HSC0.S0 <= 值 < %HSC0.S1	值 >= %HSC0.S1
%Q0.4	–	–	X
%Q0.5	X	X	–

时序图:

%HSC0.P = 17
%HSC0.S0 = 14
%HSC0.S1 = 20

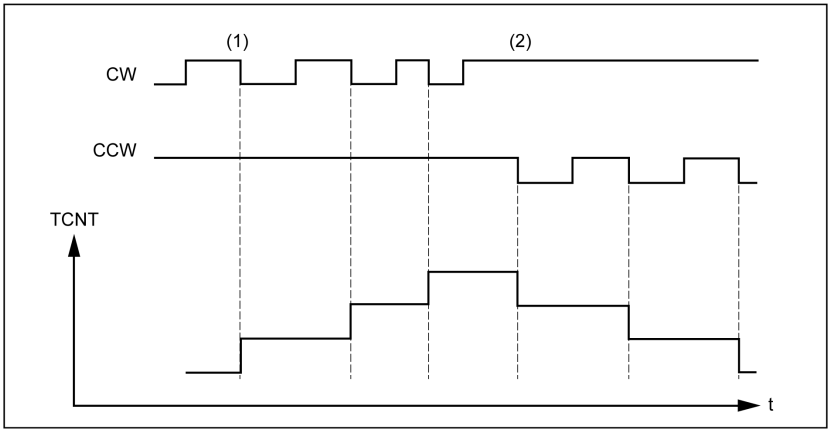


(1) 输入 IN 设置为 1，因此减计数模式启动（%HSC0.U = 0，即 IB = 0）
(2) 当前值达到 0，因此在下一次计数时，F 输出标记设置为 1，%HSC0.V 设置为 65535
(3) 在 IB 输入时更改，计数器现在处于加计数模式下，且 %HSC0.U = 1
(4) IB 输入设置为 0，因此计数器处于减计数模式下，且 %HSC0.U 设置为 0
(5) 输入 S 设置为 1，同时正在进行减计数，因此 %HSC0.V 初始化为预设值 %HSC0.P = 17
(6) S 复位为 0，预设值 %HSC0.P 更改为 20
(7) 输入 IN 设置为 0，因此禁止该功能，保留 %HSC0.V
(8) S 设置为 1，因此考虑使用新预设值 (%HSC0.P = 20)，并更新反射输出
(9) IN 输入设置为 1，然后以减计数模式重新启动该功能

(10) 阈值 %HSC0.S1 设置为 17
(11) S 输入处于活动状态，使下一次计数时赋予阈值 S1 新值，%HSC0.V 复位为预设值 17
(12) 捕获当前值 %HSC0.V，因此 %HSC0.C = 20

Dual Phase [Clock Wise / Counter Clock Wise] 时序图

时序图:



(1) 加计数开始
(2) 减计数开始

Dual Phase [Quadrature X1]、Dual Phase [Quadrature X2]、Dual Phase [Quadrature X4] 时序图

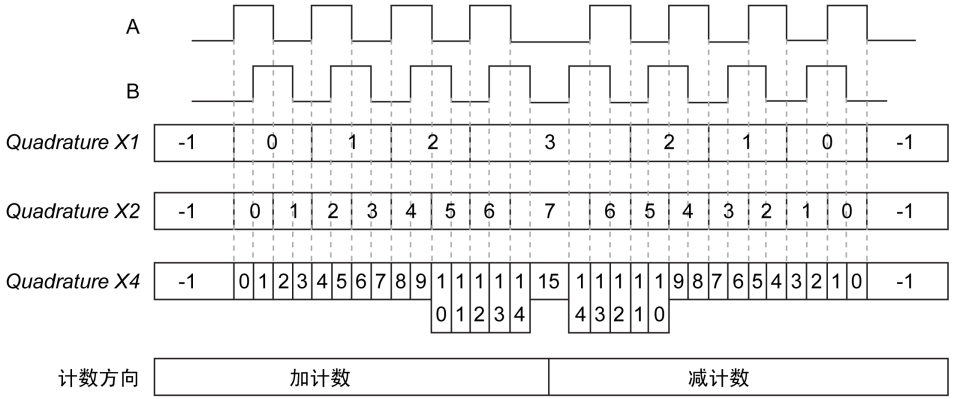
编码器信号根据选定的输入模式进行计数，如下所示：

X1 每个编码器周期计数 1 次

X2 每个编码器周期计数 2 次

X4 每个编码器周期计数 4 次

时序图:



Quadrature X1 当通道 A 领先通道 B 时，计数器会在通道 A 的上升沿上递增；当通道 B 领先通道 A 时，计数器会在通道 A 的下降沿上递减。

Quadrature X2 视两个通道之间的领先和落后关系而定，计数器会在通道 A 的每一个边沿上递增或递减。每个周期都会引起两次递增或递减。

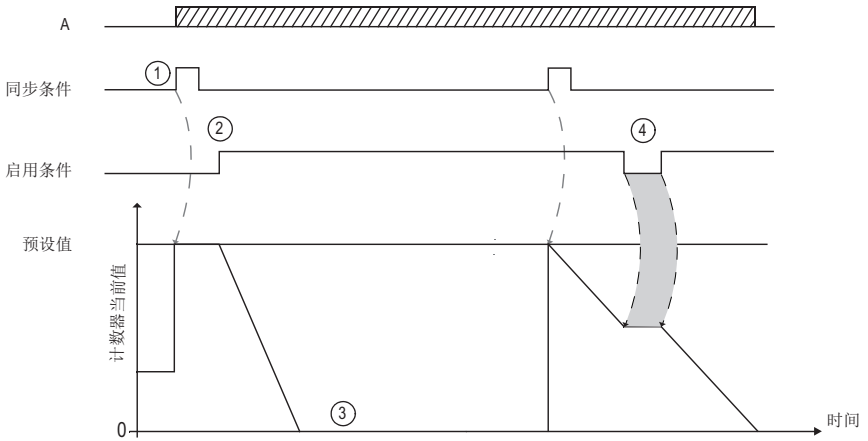
Quadrature X4 计数器在通道 A 和 B 的每一个边沿上递增或递减。计数器是递增还是递减取决于两个通道之间的领先和落后关系。每个周期都会引起四次递增或递减。

One-shot 计数模式

概述

- 脉冲只需要一个输入。
- 当出现预设条件的上升沿时，将启用计数器，并将当前值设为预设值。
- 当计数器被启用时，应用到输入的每个脉冲都递增当前值。计数器在其当前值达到最大值时停止。然后溢出标记置位。
- 即使新的脉冲应用到输入，计数器值仍然保持为最大值。
- 要再次激活计数器，需要新的预设。

One-shot 模式时序图



下表介绍上图的各个阶段：

阶段	动作
1	当出现预设条件的上升沿时，预设值将被加载到计数器（不考虑当前值），并且启用计数器。
2	当启用条件 = 1 时，当前的计数器值依据输入 A 上的各个脉冲递增，直至达到 0。
3	计数器等待预设条件的下一个上升沿出现。 **注：**此时，输入 A 上的脉冲对计数器不起作用。
4	当启用条件 = 0 时，计数器将忽略来自输入 A 的脉冲并保留其当前值，直到启用条件再次 = 1。计数器将从保留的值开始，在 Enable 输入的上升沿恢复对输入 A 的脉冲进行计数。

Modulo-loop 计数模式

概述

模数回路模式可用于对一系列移动对象执行的重复操作，如包装和贴标签等应用。

原理

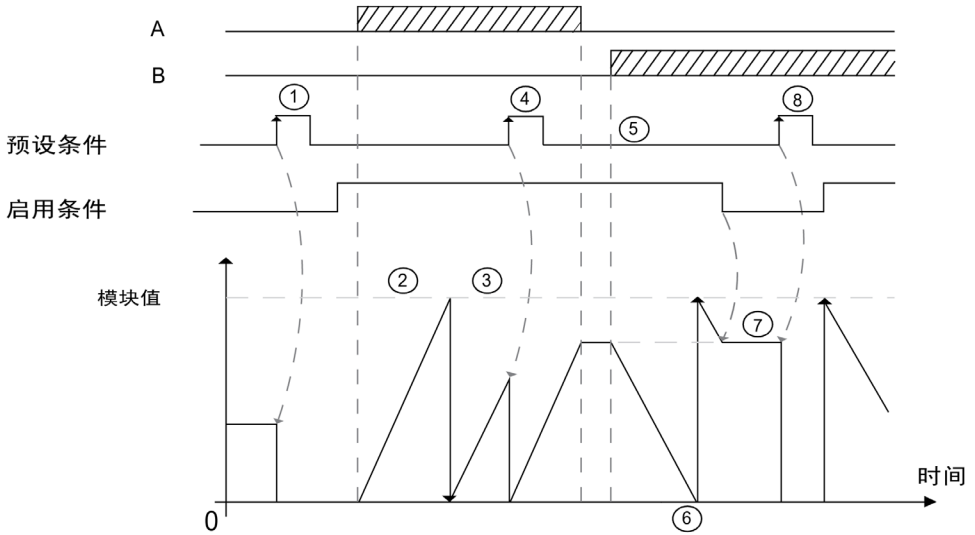
当出现预设条件的上升沿时，启用计数器，并将当前值复位为 0。

当计数器被启用时：

递增方向： 计数器在达到模数值之前一直递增。在下一个脉冲，计数器复位为 0，模数标志设置为 1 并继续计数。

递减方向： 计数器在达到 0 之前一直递减。在下一个脉冲，计数器设置为模数值，模数标志设置为 1 并继续计数。

Modulo-loop 模式时序图



阶段	动作
1	当出现预设条件的上升沿时，当前值复位为 0 并启用计数器。
2	当启用条件 = 1 时，A 上的每个脉冲会递增计数器值。
3	计数器达到模数值减 1 时，计数器在下一个脉冲返回 0，并继续计数。Modulo_Flag 设置为 1。
4	当出现预设条件的上升沿时，当前计数器值复位为 0。
5	当启用条件 = 1 时，B 上的每个脉冲会递减计数器。

阶段	动作
6	计数器达到 0 时，计数器在下一个脉冲返回模数值减 1，并继续计数。
7	启用条件 = 0 时，忽略输入上的脉冲。
8	当出现预设条件的上升沿时，当前计数器值复位为 0。

频率计模式下的高速计数器

简介

频率计模式下的 High Speed Counter 用于测量输入 IA（脉冲输入相位 A）上周期性信号的频率 (Hz)。

可以测量的频率范围是 1 Hz 到 100 kHz。

可以在 2 个时基之间进行选择，由对象 %HSC.T（时基）做出选择：

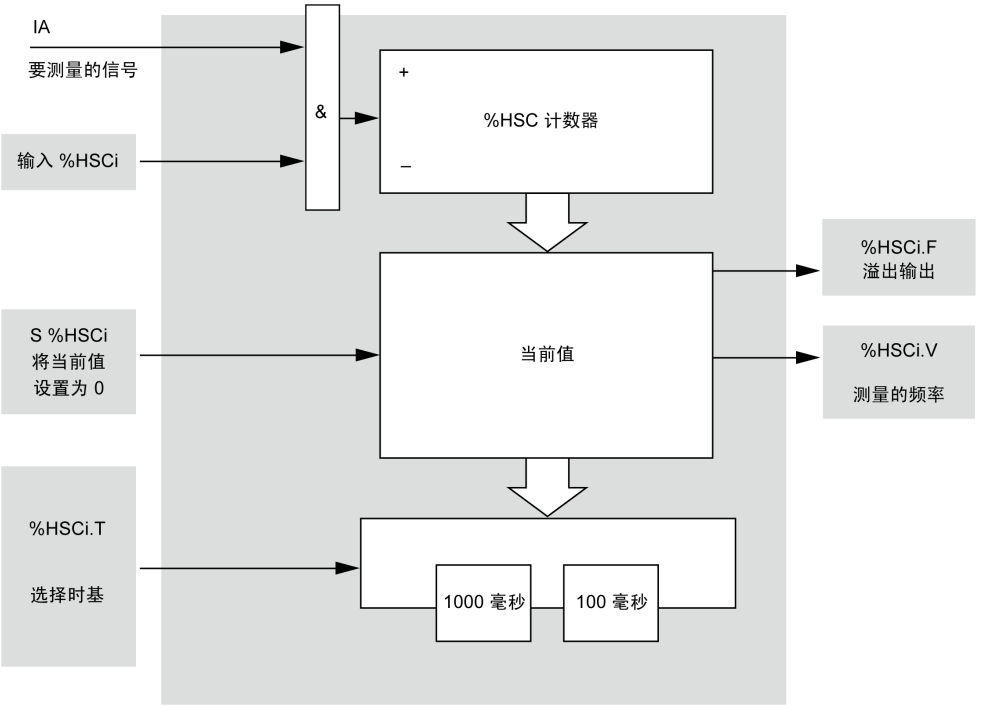
时基	准确度	更新
100 毫秒	100 kHz 为 0.05% 100 Hz 为 10%	每秒 10 次
1 秒	100 kHz 为 0.005% 10 Hz 为 10%	每秒一次

准确度测量

$$\text{准确度}(\%) = \frac{1}{f[Hz]} \times \frac{1}{TB[s]} \times 100$$

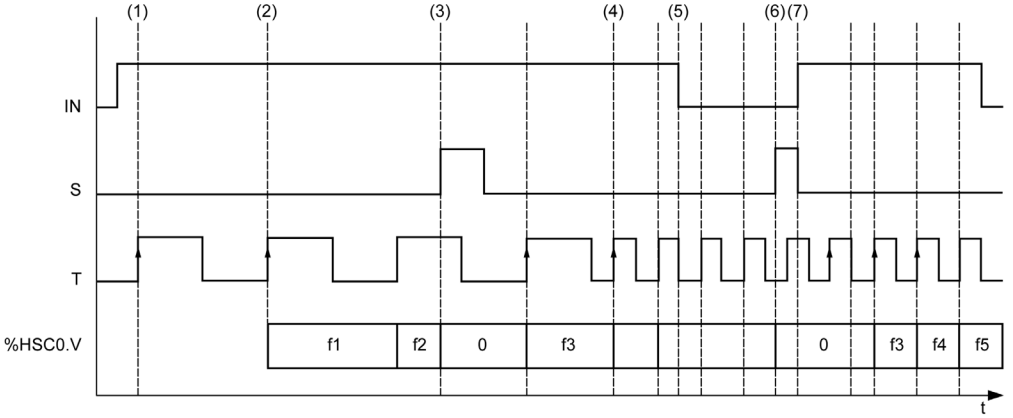
操作

以下示意图是频率计模式的操作图：



时序图

以下时序图是在频率计模式下使用 High Speed Counter 的示例：



(1) 第一个频率测量开始于 TB 信号的下一个上升沿开始测量
(2) %HSC0.V 在一个周期的 TB
(3) 在输入 S 上升沿上，当前值 %HSC0.V 将设置为 0
(4) %HSC0.T 设置为 100 毫秒，因此取消当前测量并开始一个新测量
(5) 输入 IN 设置为 0，因此禁止频率测量功能并保留 %HSC0.V。
(6) 在输入 S 上升沿上，当前值 %HSC0.V 将设置为 0
(7) S 设置为 0 且 IN 设置为 1，因此测量将在 TB 信号的下一个上升沿上开始

f*x* 对应于当前频率值。

第12.3节
脉冲 (%PLS)

使用脉冲功能块

本节介绍如何使用Pulse 功能块.

本节包含了哪些内容？

本节包含了以下主题：

主题	页
描述	256
配置	258
编程示例	261

描述

简介

Pulse 功能块 用于生成方波信号。

两个 Pulse 功能块用于专用输出通道 %Q0.0 或 %Q0.1 上。对这两个通道使用继电器输出的逻辑控制器不支持 Pulse 功能块。有关输入和输出的详细信息，请参阅 M100/M200 可编程控制器 - 硬件指南。

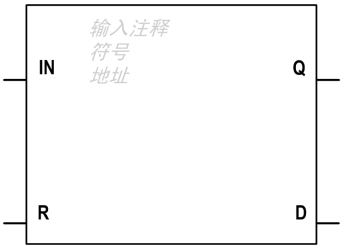
Pulse 功能块生成占空比为 50% 的脉冲。

执行脉冲输出时，可以选择限制脉冲数或周期。这些因素可在配置时确定，并/或通过程序进行更新。

在使用功能块的实例之前，您必须首先在**配置 → 脉冲发生器**中配置 Pulse 功能块。请参阅配置脉冲发生器 (参见第 *78* 页)。

示意图

此示意图为 Pulse 功能快：



输入

Pulse 功能块具有以下输入：

标签	对象	描述	值
IN	%PLSi.IN	启用	处于状态 1 时，在专用输出通道上生成脉冲。 处于状态 0 时，输出通道设置为 0。
R	%PLSi.R	复位为 0（可选）	处于状态 1 时，输出 %PLSi.Q 和 %PLSi.D 设置为 0。 在周期 T 中生成的脉冲数设置为 0。

输出

Pulse 功能块具有下列输出：

标签	对象	描述	值
Q	%PLSi.Q	正在生成	状态为 1 时，表示在配置的专用输出通道上生成Pulse信号。
D	%PLSi.D	生成完成（可选）	处于状态 1 时，表示信号生成已完成。已达到所需的脉冲数。

配置

参数

要配置参数，请执行配置功能块过程 *(参见 SoMachine Basic, 通用功能库指南)*，并阅读 SoMachine Basic 操作指南中存储器分配模式 *(参见 SoMachine Basic, 操作指南)*的描述。

Pulse 功能块具有以下参数：

参数	描述	值
已使用	已使用的地址	如果选择此参数，则当前在程序中使用此地址。
地址	%PLSi Pulse 地址	实例标识符，其中 i 是从 0 到此逻辑控制器上可用的对象数。有关 Pulse 对象的最大数量，请参阅“对象的最大数量”表 (参见第 *32* 页)。
符号	符号	与此对象关联的符号。有关详细信息，请参阅 SoMachine Basic 操作指南（定义和使用符号）。
预设	周期预选 (%PLSi.P)	● 时基 = 1 秒，%PLSi.P=1 或 2 ● 时基 = 10 毫秒，1<=%PLSi.P<=200 ● 时基 = 1 毫秒，1<=%PLSi.P<=2000 ● 时基 = 0.1 毫秒，1<=%PLSi.P<=20000
脉冲数	脉冲数 (%PLSi.N, %PLSi.ND)	要生成无限数量的脉冲，请将 %PLS.N 或 %PLS.ND 设置为 0。
当前	当前输出 (%PLSi.Q)	0 或 1。
完成	完成脉冲 (%PLSi.D)	处于状态 1 时，表示信号生成已完成。已达到所需的脉冲数。该值通过将 IN 输入或 R 输入设置为 1 进行复位。
占空比	%PLSi.R	该值提供了周期中处于状态 1 的信号百分比。因此宽度 Tp 等于：TP = T x (%PLSi.R:100)。用户应用程序将会写入 %PLSi.R 的值。正是该字控制周期的占空比。 缺省值为 0，大于 100 的值被视为 100。
注释	注释	可以将可选注释与此对象相关联。 双击**注释**列并键入注释。

对象

Pulse 功能块与以下对象相关联：

对象	描述	大小（位）	缺省值	范围	
%PLSi.P	预设值	16	预设值（在**配置 → 脉冲发生器**中设置）	**预设** %PLSi.P	**时基**
				1...20000	0.1 毫秒
				1...2000	1 毫秒
				1...200	10 毫秒
				1 或 2	1 s（缺省值）
%PLSi.N	脉冲数	16	0	0...32767	
%PLSi.ND		32	0	0 到 2147483647	

使用规则

输出信号周期 T 可通过**预设**和**时基**参数（如 T = %PLSi.P x. **时基**）来设置。

下表显示了可用周期的范围：

时基	频率
0.1 毫秒	0.5 Hz...10000 Hz
1 毫秒	0.5 Hz...1000 Hz
10 毫秒	0.5 Hz...100 Hz
1 秒	0.5 Hz...1 Hz

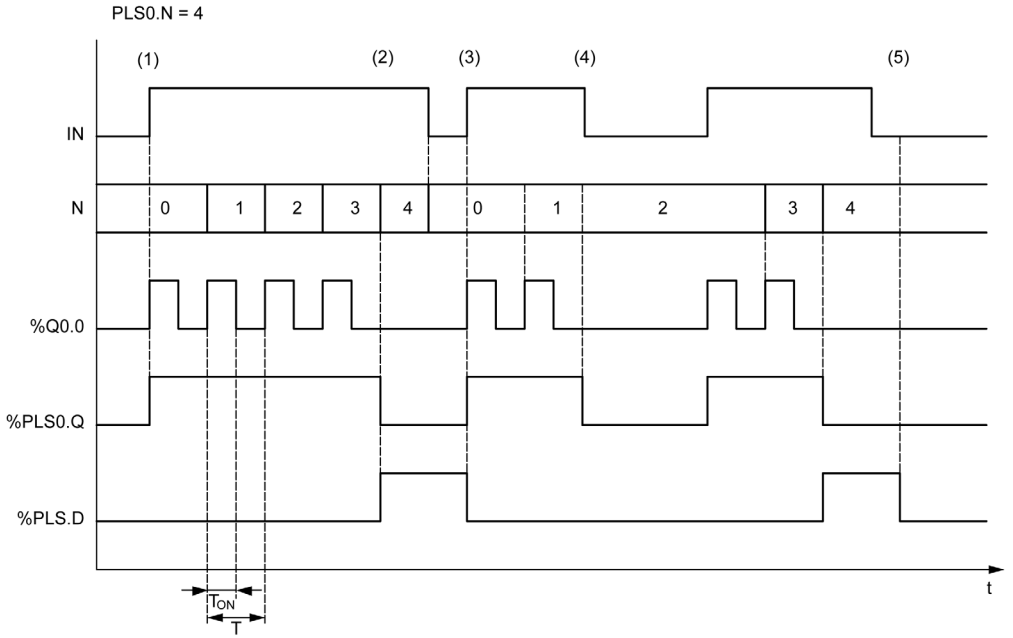
时基在**配置 → 脉冲发生器**中设置且无法修改。请参阅配置脉冲发生器 (参见第 *78* 页)。

如果 %PLSi.P：
- 已更改，则输出信号周期在当前周期结束时更改。
- 设置为 0，则脉冲发生功能停止。
- 超出范围，则参数强制为 0 并且脉冲发生功能停止。

如果 %PLSi.N（或**双字**模式下的 %PLSi.ND）：
- 已更改，则在下次执行脉冲发生功能时（%PLSi.D = 1 或者在 %PLSi.R = 1 之后）使用要生成的脉冲数。
- 设置为 0，则生成无限数量的脉冲。
- 超出范围，则参数强制为 0。

时序图

下图显示了 Pulse 功能块的时序：



(1) IN 输入设置为 1，在专用输出 (%Q0.0) 处生成脉冲信号，因此 %PLSi.Q 设置为 1
(2) 脉冲数达到 %PLS0.N (=4)，因此“完成”标志输出 (%PLS0.D) 设置为 1，并且脉冲生成停止 (%PLS0.Q = 0)
(3) IN 输入设置为 1，因此 %PLS0.D 复位为 0
(4) IN 输入设置为 0，因此输出通道设置为 0 并且 %PLS0.Q = 0 指示信号生成不处于活动状态
(5) 通过将 R 输入设置为 1，从而将 %PLS0.D 设置为 0。

特殊情况

特殊情况	描述
冷重启 (%S0=1) 的结果	● 脉冲发生停止。 ● 在控制器初始化期间，输出复位为 0。 ● 如果控制器初始化后： ❍ 控制器进入 STOPPED 状态，则会将故障预置行为应用于输出。 ❍ 控制器进入运行状态，则会恢复配置参数。
控制器停止时的效果	● 脉冲发生停止 ● 将**故障预置**行为应用于输出。
在线修改的结果	无

编程示例

简介

Pulse 功能块可按此编程示例进行配置。

编程

本示例为 Pulse 功能块的特殊操作情况的列表：

梯级	指令
0	BLK %PLS0 LD %M1 IN LD %M0 R OUT_BLK LD Q ST %Q0.5 LD D ST %M10 END_BLK

注意： 请参阅可转换性过程 (参见第 *217* 页)以获取等效梯形图。

第12.4节
驱动器功能块 (%DRV)

使用驱动器功能块

本节介绍了驱动器功能块的使用并提供了驱动器功能块使用的编程指南。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
描述	263
驱动器和逻辑控制器状态	265
添加驱动器功能块	267
功能块配置	269
MC_Jog_ATV：启动点动模式	270
MC_Power_ATV：启用/禁用电源级	273
MC_MoveVel_ATV：以指定速度移动	275
MC_Stop_ATV：停止移动	278
MC_ReadStatus_ATV：读取设备状态	280
MC_ReadMotionState_ATV：读取运动状态	282
MC_Reset_ATV：确认和复位错误	285
错误代码	287

描述

简介

驱动器功能块 DRV 让驱动装置，如 Altivar 高速驱动器，能够受到 Modicon M100/M200 可编程控制器 的控制。例如：

- 控制 ATV 驱动器管理的电机的速度，并持续更新这个速度
- 监视 ATV 驱动器和电机的状态
- 管理 ATV 驱动器中检测到的错误。

通过以下其中一种方法实现通讯：

- 使用 Modbus RTU 协议，将逻辑控制器的其中一条串行线路配置作为 Modbus Serial IOScanner (参见第 *170* 页)。
- 将以太网端口配置作为 Modbus TCP IOScanner (参见第 *153* 页)。

在 SoMachine Basic 中，首先将目标 ATV 驱动器类型添加到 Modbus Serial IOScanner 或 Modbus TCP IOScanner。这就能够建立预定义通道和初始化请求，从而允许对 ATV 驱动器上的指定寄存器执行数据读写操作，这些数据包括例如：

- **ETA** 状态字
- **ETI** 扩展状态字
- **RFRD** 输出速度 (RPM)
- **DP0** 上个错误的错误代码
- **CMD** 控制字

使用 Modbus 请求类型 **FC23 - 读/写多个寄存器**来执行数据传输。这就让程序能够以单个 Modbus 请求，例如，对 **ETA**、**ETI** 和 **DP0** 寄存器执行读取操作，并且对 **CMD** 寄存器执行写操作。

SoMachine Basic **编程**选项卡中的以下单轴驱动器功能块可用：

功能块	描述
MC_Power_ATV (参见第 *273* 页)	启用或禁用设备的电源级。
MC_Jog_ATV (参见第 *270* 页)	启动设备的点动操作模式。
MC_MoveVel_ATV (参见第 *275* 页)	指定设备的目标速度。
MC_Stop_ATV (参见第 *278* 页)	停止当前的设备移动。
MC_ReadStatus_ATV (参见第 *280* 页)	返回与设备相关的状态信息。
MC_ReadMotionState_ATV (参见第 *282* 页)	返回与当前设备移动相关的状态信息。
MC_Reset_ATV (参见第 *285* 页)	复位与驱动器状态 (参见第 *265* 页)相关的设备错误，并确认 MC_Power_ATV (参见第 *273* 页) 错误。

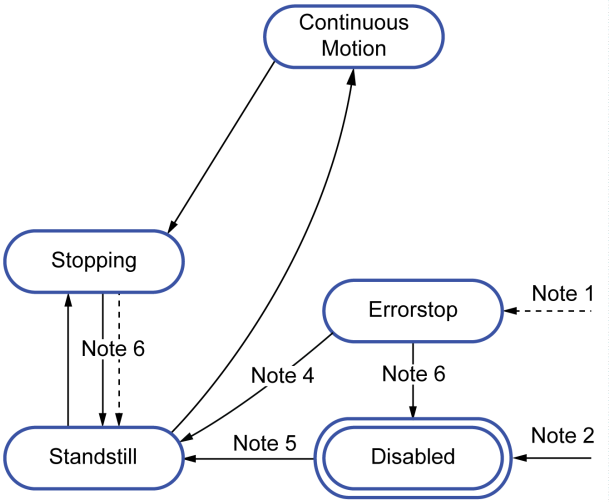
对于每个驱动器功能块，在程序中一次最多可以使用 16 个实例。

在将设备添加到 Modbus Serial IOScanner 或 Modbus TCP IOScanner 时，SoMachine Basic 使用 %DRV*n* 对象为设备分配轴，其中 *n* 是 ATV 驱动器的数量。每次将驱动器功能块添加到程序时，必须将其与轴关联，从而在功能块、轴以及 Modbus Serial IOScanner 或 Modbus TCP IOScanner 中定义的目标设备之间创建链接。

驱动器和逻辑控制器状态

驱动器状态图

驱动器始终处于下图所定义的其中一种状态下。在执行了驱动器功能块或出现了错误的情况下，可能会引起状态转换：



注 1 如果出现错误，则可能从任何状态发生转换。
注 2 %MC_Power_ATV.status 为 0 时，可能从任何状态发生转换（如果没有 ErrorAxis）。
注 3 如果 %MC_Reset_ATV.Done = 1 且 %MC_Power_ATV.status = 0，则只会从 ErrorStop 状态转换到 Disabled 状态。
注 4 如果 %MC_Reset_ATV.Done = 1 且 %MC_Power_ATV.Enable = 1 且 %MC_Power_ATV.Status = 1，则只会从 ErrorStop 状态转换到 Standstill 状态。
注 5 如果 %MC_Power_ATV.Enable = 1 且 %MC_Power_ATV.Status = 1，则只会从 DISABLED 状态转换到 Standstill 状态。
注 6 如果 %MC_Stop_ATV.Done = 1 且 %MC_Stop._ATV.Execute = 0，则只会从 Stopping 状态转换到 Standstill 状态。

下表描述驱动器状态：

状态	描述
Disabled	初始状态。驱动器不处于运行状态或错误状态。
Standstill	驱动器处于运行状态 (ETA = 16#xx37) 且 Velocity = 0 (RFRD = 0)。
ErrorStop	驱动器处于错误状态 (ETA = 16#xxx8)。
Continuous motion	驱动器处于运行状态 (ETA = 16#xx37) 且 Velocity ≠ 0 (RFRD ≠ 0)。
Stopping	正在执行 MC_Stop_ATV 功能块。

功能块 MC_ReadStatus_ATV (参见第 *280* 页) 可用于读取 ATV 驱动器的状态。

逻辑控制器状态转换

下表描述驱动器功能块如何受到逻辑控制器状态变化的影响：

逻辑控制器状态	对驱动器功能块的影响
RUNNING	驱动器功能块通常根据用户逻辑来执行。
STOPPED	控制器进入 STOPPED 状态时，除非**故障预置行为**选项设置为**保持当前值**，否则会停止所配置的驱动器轴。 如果**故障预置行为**选项设置为**故障预置值**，则会向 ATV 驱动器发送 0x00 命令，从而导致进入 Switch on Disabled (NST) 状态。相反，如果**故障预置行为**设置为**保持当前值**，则不会采取任何动作（命令未修改）。
HALTED	控制器进入 HALTED 状态时，除非**故障预置行为**选项设置为**保持当前值**，否则会停止所配置的驱动器轴。 如果**故障预置行为**选项设置为**故障预置值**，则会向 ATV 驱动器发送 0x00 命令，从而导致进入 Switch on Disabled (NST) 状态。相反，如果**故障预置行为**设置为**保持当前值**，则不会采取任何动作（命令未修改）。
POWERLESS, EMPTY	不执行驱动器功能块（Modbus Serial IOScanner 或 Modbus TCP IOScanner 停止）。 在更新了控制器中的应用程序时，也会出现这种情况。

注意： 如果控制器状态为 HALTED 或 STOPPED，并且已经选择了**保持当前值**，则控制器不会向驱动器发出任何其他命令。 因此，驱动器必须确定要进入的合适状态。如果为驱动器选择了**保持当前值**，则危害和风险分析中必须包含这一信息，以避免由此可能发生的危害性事件。

⚠ 警告

意外的设备操作

确保在设计机器的过程中，根据 EN/ISO 12100 进行风险评估。

不遵循上述说明可能导致人员伤亡或设备损坏。

添加驱动器功能块

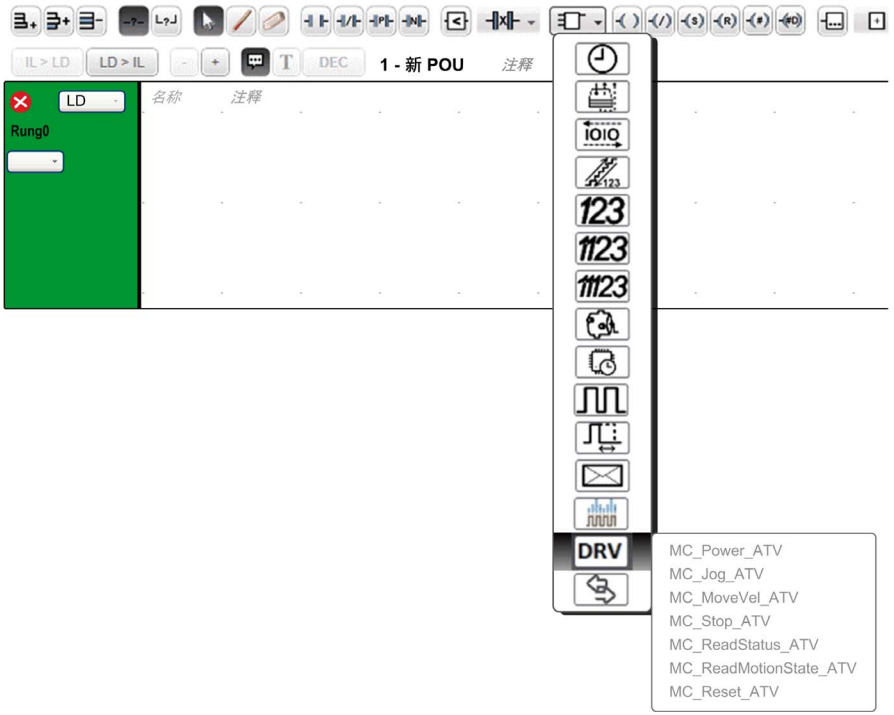
前提条件

添加驱动器功能块的前提条件：

- 必须在串行线路或以太网上对 Modbus Serial IOScanner 或 Modbus TCP IOScanner 进行了配置。
- 必须在 Modbus Serial IOScanner 或 Modbus TCP IOScanner 上添加并配置 (参见第 *170* 页)要控制的 ATV 驱动器。

添加驱动器功能块

遵照以下步骤添加驱动器功能块的实例：

步骤	动作
1	选择**编程**选项卡。
2	选择**功能块** → **驱动器**，如下图所示： 
3	单击梯级以放置所选功能块。
4	关联功能块的输入/输出。

删除功能块

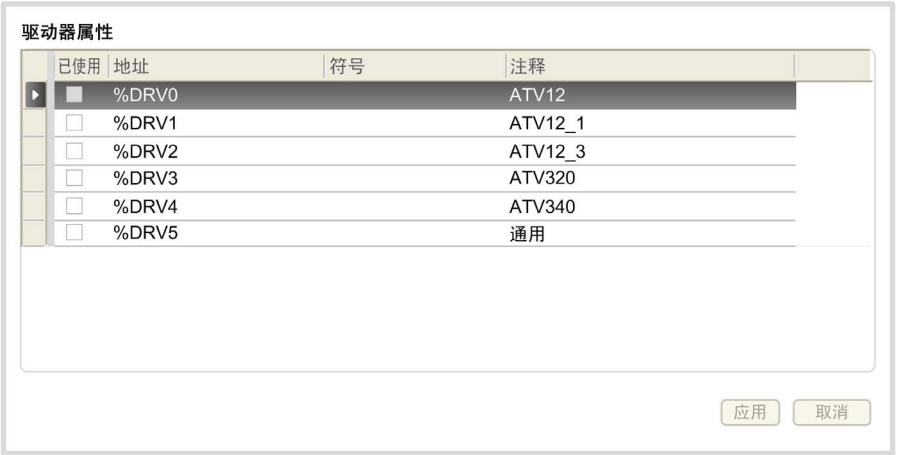
遵照以下步骤移除驱动器功能块的实例：

步骤	动作
1	在**编程**选项卡中，单击此功能块的实例。
2	按**删除**以删除所选功能块。

功能块配置

配置驱动器对象

每个驱动器功能块均与驱动器 (%DRV) 对象关联。如要显示所配置的驱动器对象的列表，则执行以下操作：

步骤	动作
1	选择**编程** → **工具**选项卡，然后单击**驱动器对象** → **驱动器**以显示驱动器对象的属性。 
2	根据需要更新属性，然后单击**应用**

驱动器功能块具有以下属性：

参数	可编辑	值	缺省值	描述
已使用	否	True/False	False	指示程序中是否在使用该驱动器对象。
地址	否	%DRVn	%DRVn	驱动器对象的地址，其中 n 是对象数量。
符号	是	–	–	可用于指定与驱动器对象关联的符号。 双击单元格以定义或编辑符号。
注释	是	–	–	可用于指定与驱动器对象关联的注释。 双击单元格以定义或编辑注释。

MC_Jog_ATV：启动点动模式

描述

此功能块可启动 Jog 操作模式。点动操作让设备以指定速度向前或向后移动。

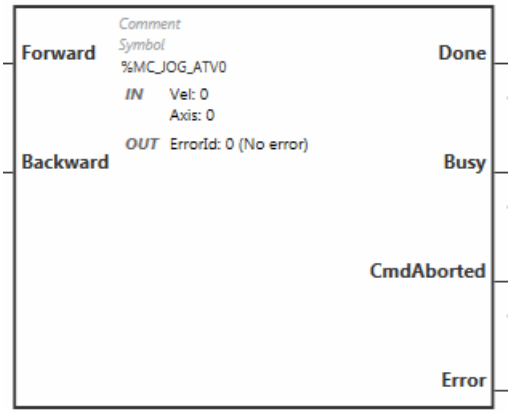
如果在执行该功能块（Busy 输出设置为 1）时，启用了功能块 MC_MoveVel_ATV (参见第 *275* 页) 或 MC_Stop_ATV (参见第 *278* 页)，则 MC_Jog_ATV 功能块会命令执行这种移动。将 Busy 输出复位为 0，将 CmdAborted 输出设置为 1。

在执行点动操作时，速度值 (Vel) 的变化仅在检测到 Forward 或 Backward 输入的下降/上升沿时才应用。

如果 Error 或 CmdAborted 输出设置为 1，则必须先将 Forward 和 Backward 输入复位为 0，然后再将新的上升沿应用至 Forward 和/或 Backward 输入，以重启移动。

如果在正执行 MC_Stop_ATV (参见第 *278* 页) 功能块时启动点动操作，则会导致 Stop Active Error。如果在驱动器不处于运行状态 (ETA ≠ 16#xx37) 时启动点动操作，则会导致 Not Run Error。

图形表示形式



输入

下表描述功能块的输入：

输入	对象	初始值	描述
Forward	-	0	将 Forward 输入或 Backward 输入设置为 1，从而启动点动。 如果 Forward 和 Backward 输入都设置为 1，则该操作模式保持活动状态，点动停止，并且 Busy 输出保持设置为 1。 如果 Forward 和 Backward 输入都设置为 0，则该操作模式终止，并且 Done 输出针对一个循环设置为 1。
Backward	-	0	
Vel	%MC_JOG_ATVi.VEL 其中 i 为 0...15	0	Jog 操作模式的目标速度，单位为每分钟转数 (rpm)。 在点动期间，速度值 (Vel) 的变化仅在检测到 Forward 或 Backward 输入的下降/上升沿时才应用。 范围：-32768...32767
Axis	%MC_JOG_ATVi.AXIS 其中 i 为 0...15	-	将执行功能块的轴的标识符 (%DRV0...%DRV15)。 必须首先在**配置**选项卡中声明该轴。

输出

下表描述功能块的输出：

输出	输出对象	初始值	描述
Done	%MC_JOG_ATVi.DONE	0	在 Forward 和 Backward 输入都设置为 0 时，输出针对一个循环设置为 1。 设置为 1 时，指示 Jog 操作模式已终止。
Busy	%MC_JOG_ATVi.BUSY	0	在以下情况下设置为 1： ● Jog 正在进行（Forward = 1 或 Backward = 1） ● Forward 和 Backward 输入都设置为 1，指示点动操作模式保持活动状态，点动已停止。
CmdAborted	%MC_JOG_ATVi.CMDABORTED	0	如果因为正执行另一命令而导致功能块执行终止，则设置为 1。
Error	%MC_JOG_ATVi.ERROR	0	当未检测到错误时，设置为 0。 如果在执行期间出现错误，则设置为 1。 功能块执行结束。 ErrorId 输出对象指示导致错误的原因。
ErrorId	%MC_JOG_ATVi.ERRORID	0（没有错误）	在 Error 输出设置为 1 时，功能块返回错误代码。 有关错误的详细信息，请参见错误代码 (参见第 *287* 页)。 范围：0...65535

参数

双击功能块，显示功能块参数。

MC_Jog_ATV 功能块具有以下参数：

参数	值	描述
已使用	已使用的地址	如果选择此参数，则当前在程序中使用此地址。
地址	%MC_Jog_ATVi	实例标识符，其中 i 是从 0 到此逻辑控制器上可用的对象数。关于驱动器对象的数量上限，请参阅对象的数量上限 (参见第 *32* 页)。
符号	符号	与此对象关联的符号。关于详细信息，请参阅 定义和使用符号。
轴	**%DRV***n*， 其中 n 为 0...15 **无**	选择要执行功能块的轴（驱动器对象实例）。 必须事先在 Modbus TCP IOScanner 或 Modbus Serial IOScanner (参见第 *171* 页) 中配置了驱动器对象。
Vel	目标速度	输入 Jog 操作模式的目标速度 ，然后按下 Enter。 缺省值：0 范围：-32768...32767
注释	注释	可以将可选注释与此对象相关联。 双击**注释**列并键入注释。

根据需要更新参数，然后单击**应用**。

MC_Power_ATV：启用/禁用电源级

描述

此功能块可启用或禁用驱动电源级。

输入 Enable 的上升沿启用电源级。在启用了电源级的情况下，输出 Status 设置为 1。

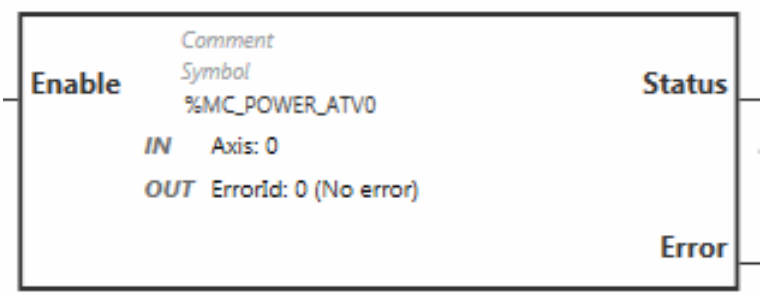
输入 Enable 的下降沿禁用电源级（Shutdown 命令，无 Error）。在禁用了电源级的情况下，输出 Status 复位为 0。

如果 ATV 驱动器的内部状态寄存器在超时值结束前尚未达到运行状态，则会生成 Timeout Error。超时计算公式为：通道循环时间乘以 4，或为 10 秒（以结果大的为准）。至少需要 10 毫秒的时间，以便驱动器能够响应。

如果在功能块执行期间检测到错误，则输出 Error 设置为 1。这会导致产生 Shutdown 命令 (CMD = 16#0006)，从而禁用 ATV 驱动器（Ready to switch on 状态，ETA = 16#xx21）。

如果出现错误，只有在成功执行了 MC_Reset_ATV (参见第 285 页) 功能块之后，才会恢复电源级。

图形表示形式



输入

下表描述功能块的输入：

标签	对象	初始值	描述
Enable	-	0	设置为 1 时，会启动该功能块的执行并启用电源级。 设置为 0 时，会停止该功能块的执行并禁用电源级。
Axis	%MC_POWER_ATVi.AXIS 其中 i 为 0...15	-	将执行功能块的轴的标识符 (%DRV0...%DRV15)。

输出

下表描述功能块的输出：

标签	对象	初始值	值
Status	%MC_POWER_ATVi.STATUS 其中 i 为 0...15	0	缺省值：0 ● 0：电源级已禁用。 ● 1：电源级已启用。 ATV 驱动器达到运行状态 (ETA = 16#xx37) 时设置为 1
Error	%MC_POWER_ATVi.ERROR 其中 i 为 0...15	0	当未检测到错误时，设置为 0。如果在执行期间出现错误，则设置为 1。功能块执行结束。 ErrorId 输出对象指示导致错误的原因。
ErrorId	%MC_POWER_ATVi.ERRORID 其中 i 为 0...15	0（没有错误）	在 Error 输出设置为 1 时，功能块返回错误代码。有关错误的详细信息，请参见错误代码 (参见第 *287* 页)。 范围：0...65535

参数

双击功能块，显示功能块参数。

MC_Power_ATV 功能块具有以下参数：

参数	值	描述
已使用	已使用的地址	如果选择此参数，则当前在程序中使用此地址。
地址	%MC_Power_ATVi	实例标识符，其中 i 是从 0 到此逻辑控制器上可用的对象数。 关于驱动器对象的数量上限，请参阅对象的数量上限 (参见第 *32* 页)。
符号	符号	与此对象关联的符号。关于详细信息，请参阅 定义和使用符号。
轴	**%DRV***n*，其中 n 为 0...15 **无**	选择要执行功能块的轴（驱动器对象实例）。 必须事先在 Modbus TCP IOScanner 或 Modbus Serial IOScanner (参见第 *171* 页) 中配置了驱动器对象。
注释	注释	可以将可选注释与此对象相关联。 双击**注释**列并键入注释。

根据需要更新参数，然后单击**应用**。

MC_MoveVel_ATV：以指定速度移动

描述

此功能块以指定的速度来启动 Profile Velocity 操作模式。在达到目标速度后，InVel 输出设置为 1。

如果在执行该功能块（Busy 输出设置为 1）时，启用了 MC_Jog_ATV (参见第 *270* 页) 或 MC_Stop_ATV (参见第 *278* 页) 功能块，则 MC_MoveVel_ATV 会命令执行这种移动。在这种情况下，将 Busy 输出复位为 0，将 CmdAborted 输出设置为 1。

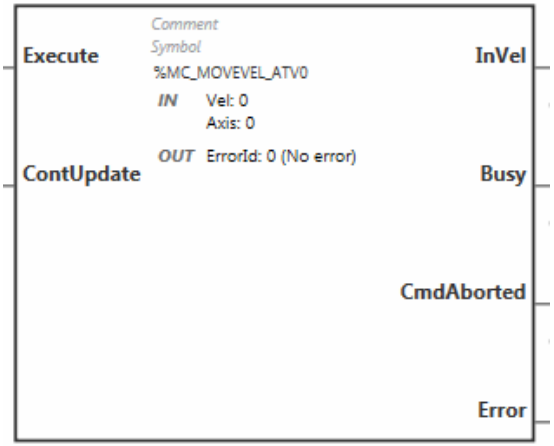
将 ContUpdate 和 Vel 输入值应用到 Execute 输入的上升沿。

如果 MC_MoveVel_ATV 的 Error 或 CmdAborted 输出设置为 1，则必须使用 Execute 的新上升沿来重新执行这种移动。

如果在正执行 MC_Stop_ATV (参见第 *278* 页) 功能块时启动该功能块，则会导致 Stop Active Error。

如果在驱动器不处于运行状态 (ETA ≠ 16#xx37) 时启动该功能块，则会导致 Not Run Error。

图形表示形式



输入

下表描述功能块的输入：

输入	对象	初始值	描述
Execute	-	0	设置为 1 时，会启动该功能块的执行。
ContUpdate	-	0	在执行功能块之前，设置为 1，以启用 Vel 参数值的连续更新。
Vel	%MC_MOVEVEL_ATVi.VEL 其中 i 为 0...15	0	操作模式的目标速度，单位为每分钟转数 (rpm)。 范围：-32 768...32 767。 若值为负，则强制在相反方向上移动。
Axis	%MC_MOVEVEL_ATVi.AXIS 其中 i 为 0...15	-	将执行功能块的轴的标识符 (%DRV0...%DRV15)。 必须首先在**配置**选项卡中声明该轴。

输出

下表描述功能块的输出：

输出	对象	初始值	描述
InVel	%MC_MOVEVEL_ATVi.INVEL	0	0 指示尚未达到目标速度 (Vel)。 在达到目标速度 (Vel) 后设置为 1。
Busy	%MC_MOVEVEL_ATVi.BUSY	0	在执行功能块时设置为 1。 即使在达到目标速度后仍保持为 1。在功能块停止或中止时复位为 0。
CmdAborted	%MC_MOVEVEL_ATVi.CMDABORTED	0	如果因为正执行另一命令而导致功能块执行终止，则设置为 1。
Error	%MC_MOVEVEL_ATVi.ERROR	0	当未检测到错误时，设置为 0。如果在执行期间出现错误，则设置为 1。功能块执行结束。ErrorId 输出对象指示导致错误的原因。
ErrorId	%MC_MOVEVEL_ATVi.ERRORID	0（没有错误）	在 Error 输出设置为 1 时，功能块返回错误代码。有关错误的详细信息，请参见错误代码 (参见第 *287* 页)。 范围：0...65535

注意： 在 ATV 驱动器的速度命令为低 (< 10) 时，InVel 和 ConstantVel 参数可能无效，因为 ATV 驱动器本身的速度范围可能不准确。

参数

双击功能块，显示功能块参数。

MC_MoveIVel_ATV 功能块具有以下参数：

参数	值	描述
已使用	已使用的地址	如果选择此参数，则当前在程序中使用此地址。
地址	%MC_MoveIVel_ATVi	实例标识符，其中 i 是从 0 到此逻辑控制器上可用的对象数。关于驱动器对象的数量上限，请参阅对象的数量上限 (参见第 *32* 页)。
符号	符号	与此对象关联的符号。关于详细信息，请参阅 定义和使用符号。
轴	**%DRV***n*，其中 n 为 0...15 **无**	选择要执行功能块的轴（驱动器对象实例）。必须事先在 Modbus TCP IOScanner 或 Modbus Serial IOScanner (参见第 *171* 页) 中配置了驱动器对象。
Vel	目标速度	输入该操作模式的目标速度 ，然后按下 Enter。 缺省值：0 范围：-32768...32767。若值为负，则强制在相反方向上移动。
注释	注释	可以将可选注释与此对象相关联。双击**注释**列并键入注释。

根据需要更新参数，然后单击**应用**。

MC_Stop_ATV：停止移动

描述

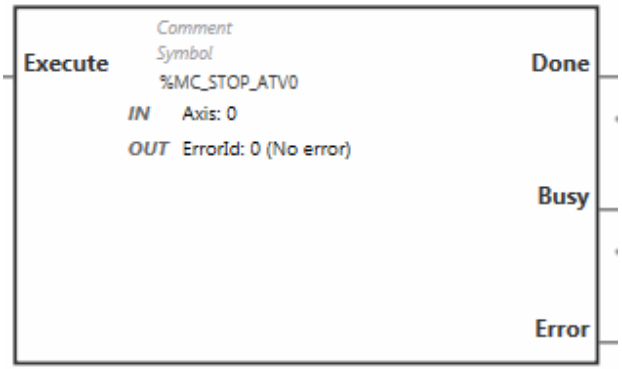
此功能块停止指定驱动器正在进行的移动。

驱动器特定的停止参数，比如减速度，通过驱动器配置来提供。

一旦通过 Execute 输入的上升沿启动，在 Done 设置为 TRUE 之前，对 Execute 输入执行的任何其他操作都将被忽略。若在 MC_Stop_ATV 为忙状态时执行另一个驱动器功能块，不会中止停止持续——功能块 MC_Stop_ATV 保持为忙，其他功能块则以错误告终。

只有在禁用电源级或者出现错误（例如，ATV Not Run 错误或 Modbus TCP IOScanner 或 Modbus Serial IOScanner 错误）的情况下，才会中断停止程序。

图形表示形式



输入

下表描述功能块的输入：

输入	对象	初始值	描述
Execute	-	0	设置为 1 时，会启动该功能块的执行。 Busy 输出设置为 1 时，无法执行其他运动功能块。 在这种情况下，这个“其他功能块”会返回错误。
Axis	%MC_STOP_ATVi.AXIS 其中 i 为 0...15	-	将执行功能块的轴的标识符 (%DRV0...%DRV15)。

输出

下表描述功能块的输出：

输出	输出对象	初始值	描述
Done	%MC_STOP_ATV*i*.DONE	0	设置为 1 时，指示功能块执行已完成。
Busy	%MC_STOP_ATV*i*.BUSY	0	在功能块执行开始时设置为 1。
Error	%MC_STOP_ATV*i*.ERROR	0	当未检测到错误时，设置为 0。如果在执行期间出现错误，则设置为 1。功能块执行结束。 ErrorId 输出对象指示导致错误的原因。
ErrorId	%MC_STOP_ATV*i*.ERRORID	0（没有错误）	在 Error 输出设置为 1 时， 功能块返回错误代码。 有关错误的详细信息，请参见错误代码 (参见第 *287* 页)。 范围：0...65535

参数

双击功能块，显示功能块参数。

MC_Stop_ATV 功能块具有以下参数：

参数	值	描述
已使用	已使用的地址	如果选择此参数，则当前在程序中使用此地址。
地址	%MC_Stop_ATVi	实例标识符，其中 i 是从 0 到此逻辑控制器上可用的对象数。 关于驱动器对象的数量上限，请参阅对象的数量上限 (参见第 *32* 页)。
符号	符号	与此对象关联的符号。关于详细信息， 请参阅 定义和使用符号。
轴	**%DRV***n*， 其中 n 为 0...15 **无**	选择要执行功能块的轴（驱动器对象实例）。 必须事先在 Modbus TCP IOScanner 或 Modbus Serial IOScanner (参见第 *171* 页) 中配置了驱动器对象。
注释	注释	可以将可选注释与此对象相关联。 双击**注释**列并键入注释。

根据需要更新参数，然后单击**应用**。

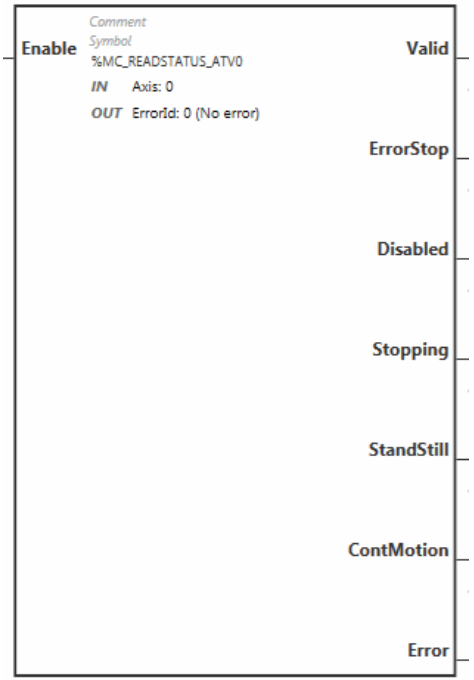
MC_ReadStatus_ATV：读取设备状态

描述

此功能块读取 ATV 驱动器的状态。

有关状态的详细信息，请参见驱动器状态图 (参见第 *265* 页)。

图形表示形式



输入

下表描述功能块的输入：

标签	对象	初始值	描述
Enable	-	0	设置为 1 时，会启用功能块。
Axis	%MC_READSTATUS_ATVi.AXIS 其中 i 为 0...15	-	将执行功能块的轴的标识符 (%DRV0...%DRV15)。

输出

下表描述功能块的输出：

标签	对象	初始值	描述
Valid	%MC_ READSATUS _ATVi.VALID	0	在功能块正无错误地运行时设置为 1。
ErrorStop	%MC_ READSTATUS _ATVi.ERRORSTOP	0	如果 ATV 驱动器处于错误状态 (ETA = 16#xxx8)，则设置为 1。
Disabled	%MC_ READSTATUS _ATVi.DISABLED	0	如果 ATV 驱动器既不处于运行状态，也不处于错误状态，则设置为 1。
Stopping	%MC_ READSTATUS _ATVi.STOPPING	0	如果正在执行 MC_Stop_ATV 功能块或正在停止移动，则设置为 1。
Standstill	%MC_ READSTATUS _ATVi.STANDSTILL	0	如果 ATV 驱动器既处于运行状态且速度为 0（ETA = 16#xx37 且 RFRD = 0），则设置为 1。
ContMotion	%MC_ READSTATUS _ATVi.CONTMOTION	0	如果 ATV 驱动器既处于运行状态且速度不等于 0（ETA = 16#xx37 且 RFRD ≠ 0），则设置为 1。
Error	%MC_ READSTATUS _ATVi.ERROR	0	当未检测到错误时，设置为 0。如果在执行期间出现错误，则设置为 1。功能块执行结束。ErrorId 输出对象指示导致错误的原因。
ErrorId	%MC_READSTATUS_ATV*i*. ERRORID	0（没有错误）	在 Error 输出设置为 1 时，功能块返回错误代码。有关错误的详细信息，请参见错误代码 (参见第 *287* 页)。 范围：0...65535

参数

双击功能块，显示功能块参数。

MC_ReadStatus_ATV 功能块具有以下参数：

参数	值	描述
已使用	已使用的地址	如果选择此参数，则当前在程序中使用此地址。
地址	%MC_ReadStatus_ATVi	实例标识符，其中 i 是从 0 到此逻辑控制器上可用的对象数。关于驱动器对象的数量上限，请参阅对象的数量上限 (参见第 *32* 页)。
符号	符号	与此对象关联的符号。关于详细信息，请参阅 定义和使用符号。
轴	**%DRV***n*， 其中 n 为 0...15 **无**	选择要执行功能块的轴（驱动器对象实例）。 必须事先在 Modbus TCP IOScanner 或 Modbus Serial IOScanner (参见第 *171* 页) 中配置了驱动器对象。
注释	注释	可以将可选注释与此对象相关联。 双击**注释**列并键入注释。

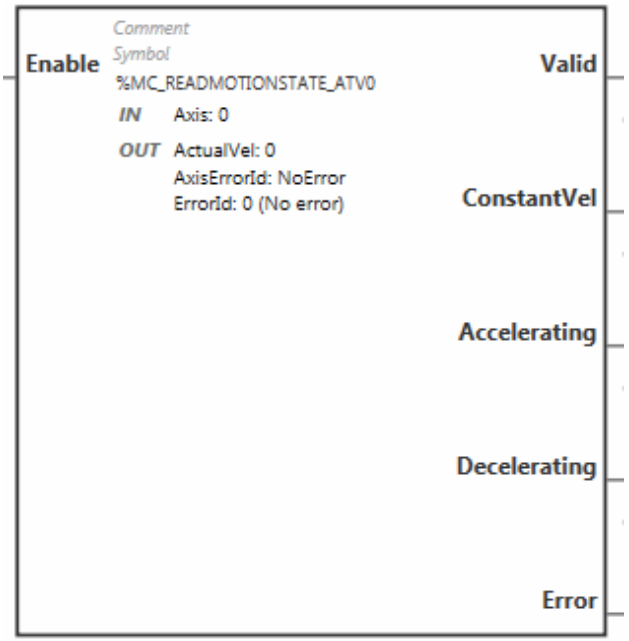
根据需要更新参数，然后单击**应用**。

MC_ReadMotionState_ATV：读取运动状态

描述

该功能块输出从 ATV 驱动器读取的与移动相关的状态信息。

图形表示形式



输入

下表描述功能块的输入：

输入	对象	初始值	描述
Enable	-	0	设置为 1 时，会启动该功能块的执行。
Axis	%MC_READMOTIONSTATE_ATVi.AXIS 其中 i 为 0...15	-	将执行功能块的轴的标识符 (%DRV0...%DRV15)。

输出

下表描述功能块的输出：

输出	对象	初始值	描述
Valid	%MC_ READMOTIONSTATE _ATVi.VALID	0	在功能块正无错误地运行时设置为 1。
ConstantVel	%MC_ READMOTIONSTATE_ATVi.CONSTANTVEL	0	在执行恒速移动（ETA 寄存器）时设置为 1。
Accelerating	%MC_ READMOTIONSTATE_ATVi.ACCELERATING	0	在电机正加速（ETI 寄存器）时设置为 1。
Decelerating	%MC_ READMOTIONSTATE_ATVi.DECELERATING	0	在电机正减速（ETI 寄存器）时设置为 1。
Error	%MC_ READMOTIONSTATE _ATVi.ERROR	0	当未检测到错误时，设置为 0。如果在执行期间出现错误，则设置为 1。功能块执行结束。ErrorId 输出对象指示导致错误的原因。
ActualVel	%MC_READMOTIONSTATE_ATVi.ACTUALVEL	0	ATV 驱动器返回的速度（RFRD 寄存器）。 范围：-32768...32767
AxisErrorId	%MC_READMOTIONSTATE_ATVi.AXISERRORID	0	ATV 驱动器返回的轴错误标识符（DP0 寄存器）。 驱动器处于错误状态时，会出现轴错误。如果驱动器不处于错误状态（ETA 寄存器 ≠ 16#xxx8），则设置为 0 有关错误的详细信息，请参见 AxisErrorId 错误代码 (参见第 *287* 页)。 范围：-32768...32767
ErrorId	%MC_READMOTIONSTATE_ATVi.ERRORID	未检测到错误 (nOF)	在 Error 输出设置为 1 时，功能块返回错误代码。 有关错误的详细信息，请参见错误代码 (参见第 *287* 页)。 范围：0...65535

注意： 在 ATV 驱动器的速度命令为低 (< 10) 时，InVel 和 ConstantVel 参数可能无效，因为 ATV 驱动器本身的速度范围可能不准确。

参数

双击功能块，显示功能块参数。

MC_ReadMotionState_ATV 功能块具有以下参数：

参数	值	描述
已使用	已使用的地址	如果选择此参数，则当前在程序中使用此地址。
地址	%MC_ReadMotionState_ATVi	实例标识符，其中 i 是从 0 到此逻辑控制器上可用的对象数。关于驱动器对象的数量上限，请参阅对象的数量上限 (参见第 *32* 页)。
符号	符号	与此对象关联的符号。关于详细信息，请参阅 定义和使用符号。
轴	**%DRV***n*，其中 n 为 0...15 **无**	选择要执行功能块的轴（驱动器对象实例）。 必须事先在 Modbus TCP IOScanner 或 Modbus Serial IOScanner (参见第 *171* 页) 中配置了驱动器对象。
注释	注释	可以将可选注释与此对象相关联。 双击**注释**列并键入注释。

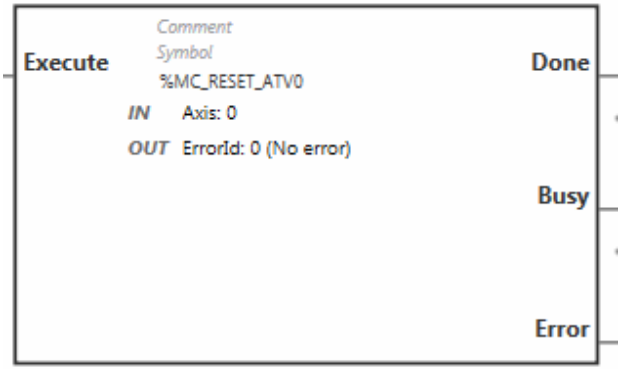
根据需要更新参数，然后单击**应用**。

MC_Reset_ATV：确认和复位错误

描述

此功能块用于确认错误，并对驱动器上的错误状态初始化。有关详细信息，请参见驱动器状态图 (参见第 *265* 页)。

图形表示形式



输入

下表描述功能块的输入：

标签	对象	初始值	描述
Execute	-	0	设置为 1 时，会启动该功能块的执行。
Axis	%MC_RESET_ATVi.AXIS 其中 i 为 0...15	-	将执行功能块的轴的标识符 (%DRV0...%DRV15)。

输出

下表描述功能块的输出：

输出	输出对象	初始值	描述
Done	%MC_RESET_ATV*i*.DONE	0	在 Reset 没有错误地结束时，设置为 1。
Busy	%MC_RESET_ATV*i*.BUSY	0	在功能块开始执行时设置为 1。
Error	%MC_RESET_ATV*i*.ERROR	0	如果设备在超时时间结束后依然保持在错误状态，则设置为 1。超时计算公式为： 通道循环时间乘以 4 或 200 毫秒 （以结果大的为准）。 至少需要 200 毫秒的时间，以便驱动器能够响应。 有关通道循环时间配置的信息，请参阅配置通道 (参见第 *175* 页)。
ErrorId	%MC_RESET_ATV*i*.ERRORID	0（没有错误）	在 Error 输出设置为 1 时，功能块返回错误代码。 有关错误的详细信息，请参见错误代码 (参见第 *287* 页)。 范围：0...65535

参数

双击功能块，显示功能块参数。

MC_Reset_ATV 功能块具有以下参数：

参数	值	描述
已使用	已使用的地址	如果选择此参数，则当前在程序中使用此地址。
地址	%MC_Reset_ATVi	实例标识符，其中 i 是从 0 到此逻辑控制器上可用的对象数。 关于驱动器对象的数量上限，请参阅对象的数量上限 (参见第 *32* 页)。
符号	符号	与此对象关联的符号。关于详细信息， 请参阅 定义和使用符号。
轴	**%DRV***n*， 其中 n 为 0...15 **无**	选择要执行功能块的轴（驱动器对象实例）。 必须事先在 Modbus TCP IOScanner 或 Modbus Serial IOScanner (参见第 *171* 页) 中配置了驱动器对象。
注释	注释	可以将可选注释与此对象相关联。 双击**注释**列并键入注释。

根据需要更新参数，然后单击**应用**。

错误代码

ErrorId 错误代码

此表列出了可能的功能块错误代码：

值	名称	描述
0	没有错误	未检测到错误。
1	IOScanner 错误	在 IOScanner 上检测到的错误 (1)。
2	ATV 处于错误状态	ATV 驱动器处于错误状态 (ETA = 16#xxx8)。
3	超时错误	在 MC_Power_ATV 功能块从驱动器接收到正确的状态之前，超时时间已结束。
4	无效的 ATV 状态	ATV 驱动器的 ETA 值无效。
5	复位错误	在 ATV 驱动器处于错误状态时，请求了 MC_Reset_ATV 功能块。
6	停止处于激活状态错误	在 MC_Stop 处于激活状态时，请求了 MC_Jog_ATV 或 MV_MoveVelocity_ATV 功能块。
7	ATV 不运行错误	ATV 驱动器不运行时，请求了 MC_Jog_ATV 或 MV_MoveVelocity_ATV 功能块。
8	无效的 AxisRef 错误	功能块的 AxisRef 输入 %DRV 无效（在 Modbus TCP IOScanner 或 Modbus Serial IOScanner 配置 (参见第 *170* 页)中不存在）。
9	内部错误	出现固件错误。

(1) 仅适用于 Modbus TCP IOScanner。
如果在设备扫描时 %MC_Power_ATV 功能块引起 IOScanner 错误，则原因可能是以太网网络过载。 如要确定错误的原因，可以：
- 检查 IOScanner 状态：%SW212 (参见第 *464* 页)。
- 检查驱动器状态：%IWNS (300+x) (参见第 *482* 页)。
- 检查通道状态：%IWNS (300+x).y (参见第 *482* 页)。
- 增大驱动器 (参见第 *155* 页)的**响应超时**值。

AxisErrorId 错误代码

下表列出了 MC_ReadMotionStatus 功能块可能返回的功能块轴错误代码：

值	名称
0	未检测到错误 (nOF)
2	EEPROM 控制 (EEF1)
3	配置不正确 (CFF)
4	配置无效 (CFI)
5	Modbus 通讯中断 (SLF1)
6	内部链接错误 (ILF)

值	名称
7	现场总线通讯中断 (CnF)
8	外部错误 (EPF1)
9	过流 (OCF)
10	预充电电容器 (CrF)
13	AI2 4-20 mA 损耗 (LFF2)
15	输入过热 (IHF)
16	驱动器过热 (OHF)
17	电机过载 (OLF)
18	DC 总线过压 (ObF)
19	主电源过压 (OSF)
20	单输出相损耗 (OPF1)
21	输入相损耗 (PHF)
22	主电源欠压 (USF)
23	电机短路 (SCF1)
24	电机过速 (SOF)
25	自调节错误
26	内部错误 1 (InF1)
27	内部错误 2 (InF2)
28	内部错误 3 (InF3)
29	内部错误 4 (InF4)
30	EEPROM ROM 功率 (EEF2)
32	接地短路 (SCF3)
33	输出相损耗 (OPF2)
37	内部错误 (InF7)
38	现场总线错误 (EPF2)
40	内部错误 8 (InF8)
42	PC 通讯中断 (SLF2)
45	HMI 通讯中断 (SLF3)
51	内部错误 9 (InF9)
52	内部错误 10 (InFA)
53	内部错误 11 (InFb)
54	IGBT 过热 (tJF)
55	IGBT 短路 (SCF4)
56	电机短路 (SCF5)

值	名称
60	内部错误 12 (InFC)
64	输入接触器 (LCF)
68	内部错误 6 (InF6)
69	内部错误 14 (InFE)
71	AI3 4-20mA 损耗 (LFF3)
72	AI4 4-20mA 损耗 (LFF4)
73	板兼容性 (HCF)
77	Conf 传输错误 (CFI2)
79	AI5 4-20mA 损耗 (LFF5)
99	通道开关错误 (CSF)
100	过程欠载 (ULF)
101	过程过载 (OLC)
105	角度错误 (ASF)
106	AI1 4-20mA 损耗 (LFF1)
107	安全功能错误 (SAFF)
110	AI2 温度检测到错误 (tH2F)
111	AI2 温度传感器错误 (t2CF)
112	AI3 温度检测到错误 (tH3F)
113	AI3 温度传感器错误 (t3CF)
114	泵循环启动错误 (PCPF)
119	泵低流量错误 (PLFF)
120	AI4 温度检测到错误 (tH4F)
121	AI4 温度传感器错误 (t4CF)
122	AI5 温度检测到错误 (tH5F)
123	AI5 温度传感器错误 (t5CF)
126	干运行错误 (drYF)
127	PID 反馈错误 (PFMF)
128	程序加载错误 (PGLF)
129	程序运行错误 (PGrF)
130	先导泵错误 (MPLF)
131	低位错误 (LCLF)
132	高位错误 (LCHF)
142	内部错误 16 (InFG)
143	内部错误 17 (InFH)

值	名称
144	内部错误 0 (InF0)
146	内部错误 13 (InFd)
149	内部错误 21 (InFL)
151	内部错误 15 (InFF)
152	固件更新错误 (FEr)
153	内部错误 22 (InFM)
154	内部错误 25 (InFP)
155	内部错误 20 (InF)
157	内部错误 27 (InFr)

第12.5节
脉冲宽度调制 (%PWM)

使用脉冲宽度调制功能块

本节介绍如何使用Pulse Width Modulation 功能块.

本节包含了哪些内容？

本节包含了以下主题：

主题	页
描述	292
功能块配置	293
编程示例	296

描述

简介

Pulse Width Modulation功能块 可以在专用输出通道 %Q0.0 或 %Q0.1 上生成具有可变宽度和占空比的可变波形信号。

带有用于这两个通道的继电器输出的控制器不支持该功能。

%PWM0 使用专用输出 %Q0.0，而 %PMW1 使用专用输出 %Q0.1。也可以将脉冲功能块 %PLS 配置为使用这些相同的专用输出。您可以为任意给定的输出配置其中的一个功能，但不能同时配置这两个功能。

在使用功能块的实例之前，您必须首先在**配置 → 脉冲发生器**中配置 Pulse Width Modulation 功能块。请参阅配置脉冲发生器 (参见第 *78* 页)。

示意图

本示意图为 Pulse Width Modulation 功能块：

输入

Pulse Width Modulation 功能块具有以下输入：

标签	对象	描述	值
IN	%PWMi.IN	启用	状态为 1 时，在输出通道生成 Pulse Width Modulation 信号。 处于状态 0 时，输出通道设置为 0。

功能块配置

概述

要配置 Pulse Generator 资源，请参阅配置脉冲发生器 (参见第 *78* 页)。

要配置 Pulse Generator 资源作为 PWM，请参阅 PWM 配置 (参见第 *82* 页)。

参数

要配置参数，请执行配置功能块过程 *(参见 SoMachine Basic, 通用功能库指南)*，并阅读 SoMachine Basic 操作指南中存储器分配模式 *(参见 SoMachine Basic, 操作指南)*的描述。

Pulse Width Modulation 功能块具有以下属性：

属性	值	描述
已使用	已激活/已停用的复选框	指示地址是否正在使用。
地址	%PWMi，其中 i 为 0 或 1	i 是实例标识符。
符号	用户定义的文本	可唯一标识该对象的符号。有关详细信息，请参阅《SoMachineBasic操作指南》（定义和使用符号）。
预设	● 如果**时基** = 1 秒，那么 %PWMi.P=1 ● 如果**时基** = 10 毫秒，那么 1<=%PWMi.P<=100 ● 如果**时基** = 1 毫秒，那么 1<=%PWMi.P<=1000 ● 如果**时基** = 0.1 毫秒，那么 1<=%PWMi.P<=10000	周期预选。
占空比	0 至 100。 **注意：** 大于 100 的值将被视为等于 100。	**占空比**由对象 %PWMi.R 控制，是这一期间处于状态 1 的信号的百分比。因此，状态 1 的宽度 (Tp) 等于：TP = T x (%PWMi.R/100)。用户应用程序将写入 %PWMi.R 的值。
注释	用户定义的文本	与该对象关联的注释。

注意：编程选项卡下的**脉冲发生器属性**表中出现的**脉冲数**、**电流**和**完成**属性不适用于 PWM 功能。

对象

Pulse Width Modulation 功能块与以下对象相关联：

对象	描述	大小（位）	缺省值	范围	
%PWMi.P	预设值	16	预设值（在**配置** → **脉冲发生器**中设置）	**预设** %PWMi.P	**时基**
				1...10000	0.1 毫秒
				1...1000	1 毫秒
				1...100	10 毫秒
				1	1 秒（缺省值）
%PWMi.R	占空比（比率）	16	0	0 到 100	

如果 %PWMi.P：
- 已更改，则输出信号周期在当前周期结束时会受影响。
- 设置为 0，则脉冲发生功能停止。
- 超出范围，则参数强制为 0 并且脉冲发生功能停止。

如果 %PWMi.R：
- 设置为 0，则脉冲发生功能停止（输出设置为 0）。
- 设置为 100，则输出设置设置为 1
- 已更改，则输出信号比在电流周期结束时更改
- 超出范围，则参数强制为 0。

时基

时基可通过**配置** → **脉冲发生器**进行设置，并且只能在**配置**选项卡中进行修改。请参阅配置脉冲发生器 (参见第 *78* 页)。

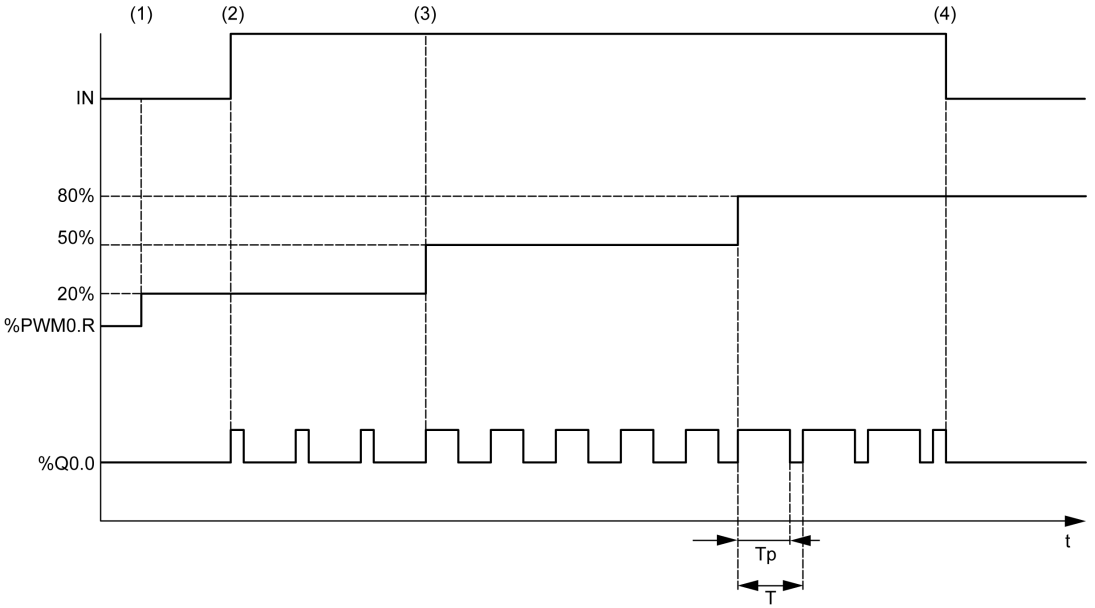
输出信号周期 T 可通过**预设**和**时基**参数（如 T = %PWMi.P x **时基**）来设置。

下表显示了可用周期的范围：

时基	频率
0.1 毫秒	1 Hz...10000 Hz
1 毫秒	1 Hz...1000 Hz
10 毫秒	1 Hz...100 Hz
1 秒	1 Hz...1 Hz

时序图

本图显示 Pulse Width Modulation 功能块的时序：



(1) PWM 比率 (%PWMi.R) 设置为 20%IN = 0，因此脉冲发生不处于活动状态
(2) IN 设置为 1，因此激活 PWM 输出
(3) 可编程宽度 (Tp) 随 %PWM.R 而变化
(4) IN 设置为 0，因此禁止 PWM 功能

特殊情况

特殊情况	描述
冷重启 (%S0=TRUE) 的结果	● 脉冲发生停止。 ● 在控制器初始化期间，输出复位为 0。 ● 如果控制器初始化后： ❍ 控制器进入已停止状态，则会将配置的故障预置策略应用于输出。 ❍ 控制器进入运行状态，则会恢复配置参数。
控制器停止时的效果	● 脉冲发生停止。 ● 将故障预置行为应用于输出。
在线修改的结果	无

编程示例

简介

Pulse Width Modulation 功能块可按此编程示例进行配置。

编程示例

在本示例中：

- 程序根据控制器输入 %I0.0 和 %I0.1 的状态修改信号宽度。
- 时基设置为 10 毫秒。
- 预设值 %PWM0.P 设置为 50，因此时基周期比为 2%。
- 可配置周期 T 等于 500 毫秒。

结果是：

- 如果 %I0.0 和 %I0.1 设置为 0，则 %PWM0.R 比率设置为 20%，状态 1 时信号的持续时间则为：20% x 500 毫秒 = 100 毫秒。
- 如果 %I0.0 设置为 1，%I0.1 设置为 0，则 %PWM0.R 比率设置为 50%（持续时间为 250 毫秒）。
- 如果 %I0.0 和 %I0.1 都设置为 1，则 %PWM0.R 比率设置为 80%（持续时间为 400 毫秒）。

Pulse Width Modulation 指令示例：

梯级	指令
0	LDN %I0.0 ANDN %I0.1 [%PWM0.R:=20]
1	LD %I0.0 ANDN %I0.1 [%PWM0.R:=50]
2	LD %I0.0 AND %I0.1 [%PWM0.R:=80]
3	BLK %PWM0 LD %I0.2 IN END_BLK

注意： 请参阅可转换性过程 (参见第 *217* 页)以获取等效梯形图。

第13章
网络对象

本章包含了哪些内容？

本章包含了以下主题：

主题	页
输入寄存器对象 (%QWM)	298
输出寄存器 (Modbus TCP) 对象 (%IWM)	299
数字量输入 (IOScanner) 对象 (%IN)	300
数字量输出 (IOScanner) 对象 (%QN)	301
输入寄存器 (IOScanner) 对象 (%IWN)	303
输出寄存器 (IOScanner) 对象 (%QWN)	304
Modbus IOScanner 网络诊断代码 (%IWNS)	306

输入寄存器对象 (%QWM)

简介

输入出寄存器对象是逻辑控制器上所接收的 Modbus 映射表输入寄存器的数字值。

显示输入寄存器属性

遵循下列步骤可显示输入寄存器对象的属性：

步骤	动作
1	选择**编程**窗口左侧区域的**工具**选项卡。
2	单击 **网络对象 → 输入寄存器**。 **结果：** 属性窗口随即出现。

输入寄存器属性

此表介绍了输入寄存器对象的每一个属性：

参数	可编辑	值	缺省值	描述
已使用	否	TRUE/FALSE	FALSE	指示是否正在程序中引用对象。
地址	否	%QWMi	–	输入寄存器对象的地址， i 在其中充当实例标识符。 关于实例的数量上限，请参阅 对象的数量上限 (参见第 *32* 页)。
符号	是	–	–	与此地址关联的符号。 在 **符号** 列中双击，然后键入要与此对象关联的符号的名称。 如果某个符号已经存在，则可以右键单击**符号**列，然后选择**搜索并替换**，在整个程序和/或程序注释中查找并替换出现此符号的地方。
故障预置值	是	-32768...32767	0	指定当逻辑控制器进入“停止”或异常状态时将适用于此对象的值。 **注意：** 如果配置了**维护值**故障预置模式，则在逻辑控制器进入“”或异常状态时，该对象会保持其当前值。值 0 随即出现，且无法编辑。关于更多详情，请参阅 故障预置行为 (参见第 *50* 页)。
注释	是	–	–	注释与该对象关联。 双击 **注释** 列， 然后键入要与此对象关联的可选注释。

输出寄存器 (Modbus TCP) 对象 (%IWM)

简介

输出寄存器对象是逻辑控制器上所接收的 Modbus TCP 映射表输出寄存器的数字值。

显示输出寄存器属性

遵循下列步骤可显示输出寄存器对象的属性：

步骤	动作
1	选择**编程**窗口左侧区域的**工具**选项卡。
2	单击 **网络对象 → 输出寄存器 (Modbus TCP)**。 **结果：** 属性窗口随即出现。

输出寄存器属性

此表介绍了输出寄存器对象的每一个属性：

参数	可编辑	值	缺省值	描述
已使用	否	TRUE/FALSE	FALSE	指示是否正在程序中引用对象。
地址	否	%IWMi	–	输出寄存器对象的地址，i 在其中充当实例标识符。 关于实例的数量上限， 请参阅 对象的数量上限 (参见第 *32* 页)。
符号	是	–	–	与此地址关联的符号。 在 **符号** 列中双击，然后键入要与此对象关联的符号的名称。 如果某个符号已经存在，则可以右键单击**符号**列，然后选择**搜索并替换**， 在整个程序和/或程序注释中查找并替换出现此符号的地方。
注释	是	–	–	注释与该对象关联。 双击 **注释** 列，然后键入要与此对象关联的可选注释。

数字量输入 (IOScanner) 对象 (%IN)

简介

数字量输入 (IOScanner) 对象是从 Modbus Serial IOScanner 或 Modbus TCP IOScanner 设备接收的数字值。

显示 Digital inputs (IOScanner) 属性

遵循下列步骤可显示 Digital inputs (IOScanner) 对象的属性：

步骤	动作
1	选择**编程**窗口左侧区域的**工具**选项卡。
2	单击**网络对象** → **数字量输入 (IOScanner)**。 **结果：** 属性窗口随即出现。

Digital inputs (IOScanner) 属性

此表介绍了 Digital inputs (IOScanner) 对象的每一个属性：

参数	可编辑	值	缺省值	描述
已使用	否	TRUE/FALSE	FALSE	指示是否正在程序中引用对象。
地址	否	%IN(i+x).y.z)	–	对象的地址，其中： ● i：索引： ❍ 100（对于 SL1） ❍ 300（对于 ETH1(Modbus TCP IOScanner)） ● x：设备 ID ● y：通道 ID ● z：对象实例标识符 关于实例的数量上限，请参阅 对象的数量上限 (参见第 *32* 页)。
通道	否	已配置通道的名称。	-	正用于从设备接收数据的通道的名称。
符号	是	–	–	与此地址关联的符号。 在 **符号** 列中双击，然后键入要与此对象关联的符号的名称。 如果某个符号已经存在，则可以右键单击**符号**列，然后选择**搜索并替换**，在整个程序和/或程序注释中查找并替换出现此符号的地方。
注释	是	–	–	注释与该对象关联。 双击 **注释** 列， 然后键入要与此对象关联的可选注释。

数字量输出 (IOScanner) 对象 (%QN)

简介

数字量输出 (IOScanner) 对象是发送到 Modbus Serial IOScanner 或 Modbus TCP IOScanner 设备的数字值。

显示 Digital ouputs (IOScanner) 属性

遵循下列步骤可显示 Digital ouputs (IOScanner) 对象的属性：

步骤	动作
1	选择**编程**窗口左侧区域的**工具**选项卡。
2	单击**网络对象 → 数字量输出 (IOScanner)**。 **结果：** 属性窗口随即出现。

Digital ouputs (IOScanner) 对象属性

下表介绍了 Digital ouputs (IOScanner) 对象的每一个属性：

参数	可编辑	值	缺省值	描述
已使用	否	TRUE/FALSE	FALSE	指示是否正在程序中引用对象。
地址	否	%QN(i+x).y.z	–	对象的地址，其中： ● i：索引： ❍ 100（对于 SL1） ❍ 300（对于 ETH1(Modbus TCP IOScanner)） ● x：设备 ID ● y：通道 ID ● z：对象实例标识符 关于实例的数量上限， 请参阅 对象的数量上限 (参见第 *32* 页)。
通道	是	已配置通道的名称。	-	正用于向设备发送数据的通道的名称。
故障预置值	是	0 或 1	0	指定当逻辑控制器进入 STOPPED 或例外状态时将适用于此对象的值。 **注意：** 如果配置了**保持当前值**故障预置模式，则在逻辑控制器进入 STOPPED 或例外状态时，该对象会保持其当前值。值 0 随即出现，且无法编辑。关于更多详情，请参阅 故障预置行为 (参见第 *50* 页)。

参数	可编辑	值	缺省值	描述
符号	是	–	–	与此地址关联的符号。 在 **符号** 列中双击，然后键入要与此对象关联的符号的名称。 如果某个符号已经存在， 则可以右键单击**符号**列，然后选择**搜索并替换**，在整个程序和/或程序注释中查找并替换出现此符号的地方。
注释	是	–	–	注释与该对象关联。 双击 **注释** 列， 然后键入要与此对象关联的可选注释。

输入寄存器 (IOScanner) 对象 (%IWN)

简介

输入寄存器 (IOScanner) 对象是从 Modbus Serial IOScanner 或 Modbus TCP IOScanner 设备接收的寄存器值。

显示 Input registers (IOScanner) 属性

遵循下列步骤可显示 Input registers (IOScanner) 对象的属性：

步骤	动作
1	选择**编程**窗口左侧区域的**工具**选项卡。
2	单击 **网络对象 → 输入寄存器 (IOScanner)**。 **结果：** 属性窗口随即出现。

Input registers (IOScanner) 属性

此表介绍了 Input registers (IOScanner) 对象的每一个属性：

参数	可编辑	值	缺省值	描述
已使用	否	TRUE/FALSE	FALSE	指示是否正在程序中引用对象。
地址	否	%IWN(i+x).y.z	–	对象的地址，其中： ● i：索引： ❍ 200（对于 SL2） ❍ 300（对于 ETH1(Modbus TCP IOScanner)） ● x：设备 ID ● y：通道 ID ● z：对象实例标识符 关于实例的数量上限，请参阅 对象的数量上限 (参见第 *32* 页)。
通道	否	已配置通道的名称。	-	正用于从设备接收数据的通道的名称。
符号	是	–	–	与此地址关联的符号。 在 **符号** 列中双击，然后键入要与此对象关联的符号的名称。 如果某个符号已经存在，则可以右键单击**符号**列，然后选择**搜索并替换**，在整个程序和/或程序注释中查找并替换出现此符号的地方。
注释	是	–	–	注释与该对象关联。 双击 **注释** 列，然后键入要与此对象关联的可选注释。

输出寄存器 (IOScanner) 对象 (%QWN)

简介

输出寄存器 (IOScanner) 对象是发送到 Modbus Serial IOScanner 或 Modbus TCP IOScanner 设备的寄存器值。

显示 Output registers (IOScanner) 属性

遵循下列步骤可显示 Output registers (IOScanner) 对象的属性：

步骤	动作
1	选择**编程**窗口左侧区域的**工具**选项卡。
2	单击 **网络对象 → 输出寄存器 (IOScanner)**。 **结果：** 属性窗口随即出现。

Output registers (IOScanner) 对象属性

下表介绍了 Output registers (IOScanner) 对象的每一个属性：

参数	可编辑	值	缺省值	描述
已使用	否	TRUE/FALSE	FALSE	指示是否正在程序中引用对象。
地址	否	%QWN(i+x).y.z	–	对象的地址，其中： ● i：索引： ❍ 100（对于 SL1） ❍ 300（对于 ETH1(Modbus TCP IOScanner)） ● x：设备 ID ● y：通道 ID ● z：对象实例标识符 关于对象的数量上限，请参阅 对象的数量上限 (参见第 *32* 页)。
通道	是	已配置通道的名称。	-	正用于向设备发送数据的通道的名称。
故障预置值	是	-32768...32767	0	指定当逻辑控制器进入 STOPPED 或例外状态时将适用于此对象的值。 **注意：** 如果配置了**保持当前值**故障预置模式，则在逻辑控制器进入 STOPPED 或例外状态时，该对象会保持其当前值。值 0 随即出现，且无法编辑。关于更多详情，请参阅 故障预置行为 (参见第 *50* 页)。

参数	可编辑	值	缺省值	描述
符号	是	–	–	与此地址关联的符号。 在 **符号** 列中双击，然后键入要与此对象关联的符号的名称。 如果某个符号已经存在，则可以右键单击**符号**列，然后选择**搜索并替换**，在整个程序和/或程序注释中查找并替换出现此符号的地方。
注释	是	–	–	注释与该对象关联。 双击 **注释** 列， 然后键入要与此对象关联的可选注释。

Modbus IOScanner 网络诊断代码 (%IWNS)

设备诊断代码

下表显示了相应 Modbus IOScanner 网络诊断对象（对于 SL1，为 %IWNS(100+x)，对于 ETH1，为 %IWNS(300+x)）中设备 x 返回的诊断代码的可能值：

值	描述
0	设备未被扫描。
1	设备正被 Modbus IOScanner 初始化（设备正发送初始化请求）。
2	设备存在并已准备好被扫描（已发送初始化请求（如有））。
3	在设备的通道上检测到通讯错误，因此未能正确扫描设备。
4	在设备的初始化请求期间检测到通讯错误，因此未能正确初始化设备。
5	设备返回的供应商名称或产品代码与预期值不匹配，因此未能正确识别设备。
6	在识别和初始化期间发生通讯错误。可能的原因有：设备无法通讯或不存在，通讯参数不正确，或不支持 Modbus 功能。

通道诊断代码

下表显示了相应 Modbus IOScanner 网络诊断对象（对于 SL1，为 %IWNS(100+x).y，对于 ETH1，为 %IWNS(300+x)）中设备 x 和通道 y 返回的诊断代码的可能值：

值	描述
0	通道处于活动状态
-1	通道处于不活动状态
其他	通讯错误代码 (CommError) 的值

第14章
脉冲串输出 (%PTO)

使用脉冲串输出功能块

本章介绍如何使用 Pulse Train Output 功能块并提供其编程指南。

本章包含了哪些内容？

本章包含了以下部分：

节	主题	页
14.1	描述	308
14.2	配置	319
14.3	编程	329
14.4	原点模式	335
14.5	数据参数	347
14.6	操作模式	351
14.7	**运动**功能块	356
14.8	**管理**功能块	387

第14.1节
描述

概述

本节介绍 Pulse Train Output 功能。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
脉冲串输出 (PTO)	309
加速度/减速度斜坡	312
探测器事件	314
反向间隙补偿	316
定位限制	317

脉冲串输出 (PTO)

简介

M200 PTO 功能通过专用输出通道输出指定数量和速度（频率）的脉冲。PTO 功能用于在开环模式下控制一个或两个独立的线性单轴步进驱动器或伺服驱动器的定位或速度。PTO 功能没有过程的任何位置反馈信息。因此，在驱动器中必须集成位置信息。

PLS（脉冲）、PWM（脉宽调制）、PTO（脉冲序列输出）和 FREQGEN（频率发生器）功能使用相同的专用输出。在同一个通道上只能使用这四个功能的其中一个。通道 0 和通道 1 上可以使用不同的功能。

PTO 通道可对回归（参考）、事件（检测器）、限制（LimP、LimN）或驱动接口（DriveReady、DriveEnable）使用可选的接口信号。

自动原点补偿也得到管理，以改善定位精度。可以对状态监视进行诊断，以便全面快速地进行排除故障。

支持的功能

PTO 通道支持以下功能：

- 两种输出模式（两个分别用于脉冲和方向的通道或一个用于 CW/CCW 的通道）
- 单轴移动（速度和位置）
- 相对和绝对定位，采用自动方向管理
- 梯形和 S 曲线加速和减速
- 原点（四个模式，带有偏移补偿）
- 动态加速、减速、速度和位置修改
- 从速度模式切换到位置模式
- 移动排队（一个移动的缓冲区）
- 事件触发位置捕捉和移动（使用探测器输入）
- 反向间隙补偿
- 限位（硬件和软件）
- 诊断

注意： 运动功能块 (参见第 *356* 页)和 管理功能块 (参见第 *387* 页)帮助您设置这些功能。

PTO 特性

PTO 通道最多可以有六个物理输入：

- 其中两路输入通过配置分配给 PTO 功能，并在出现输入的上升沿时纳入考虑范畴：
  - 参考输入（%I0.8 用于 %PT00 并且 %I0.10 用于 %PT01）
  - 索引输入（%I0.9 用于 %PT00 并且 %I0.11 用于 %PT01）
  - 探测器输入（%I0.12 用于 %PT00 并且 %I0.13 用于 %PT01）
- 另外三路输入分配给 MC_Power_PTO 功能块。它们都没有固定分配（不在屏幕画面中进行配置），并通过任何其他输入读取：
  - DriveReady 输入
  - 正向限位输入
  - 反向限位输入

注意： 这些输入像任何其他常规输入一样进行管理，但是在分配给 MC_Power_PTO 功能块时由 PTO 功能使用。

注意： 帮助防止越程需要正向和反向限位输入。

⚠ 警告

意外的设备操作

- 确保控制器硬件限位开关在应用程序的设计和逻辑中集成。
- 在允许充足制动距离的位置安装控制器硬件控制开关。

不遵循上述说明可能导致人员伤亡或设备损坏。

PTO 通道最多可以有三个物理输出：

- 其中两路输出是强制输出，用于管理 PTO 功能的输出模式。它们都有固定分配，且必须通过配置启用：
  - 顺时针/逆时针
  - 脉冲
- 另一路输出 DriveEnable 已与 MC_Power_PTO 功能块关联。它没有固定的分配，在 MAST 周期结束时作为常规输出写入。

PTO 功能具有以下特性：

特性	值
通道数	2
轴数	每通道 1 个
位置范围	-2,147,483,648...2,147,483,647（32 位）
最小速度	0 Hz
最大速度	100 kHz（针对 40/60 占空比和最大 200 mA）
最小步长	1 Hz
速度精度	1 %

特性	值
加速度/减速度（最小值）	1 Hz/ms
加速度/减速度（最大值）	100 kHz/ms
原点补偿	-2,147,483,648...2,147,483,647（32 位）
软件限位范围	-2,147,483,648...2,147,483,647（32 位）

加速度/减速度斜坡

启动速度

启动速度是步进电机在施加负载且没有步进损失的情况下可以开始移动的最小频率。

当从速度 0 开始运动时使用启动速度参数。

启动速度必须在 0...MaxVelocityAppl *（参见 Modicon M241 Logic Controller，PTOPWM，库指南）*的范围内。

值为 0 表示未使用**启动速度**参数。在这种情况下，将以“速度 = 加速度 x 1 毫秒”的速度开始运动。

停止速度

停止速度是步进电机在施加负载且没有步进损失的情况下停止移动的最大频率。

仅当从高于**停止速度**的速度向速度 0 移动时，使用**停止速度**。

停止速度必须在 0...MaxVelocityAppl *（参见 Modicon M241 Logic Controller，PTOPWM，库指南）*的范围内。

值为 0 表示未使用**停止速度**参数。在这种情况下，将以“速度 = 减速度 x 1 毫秒”的速度停止运动。

加速度/减速度

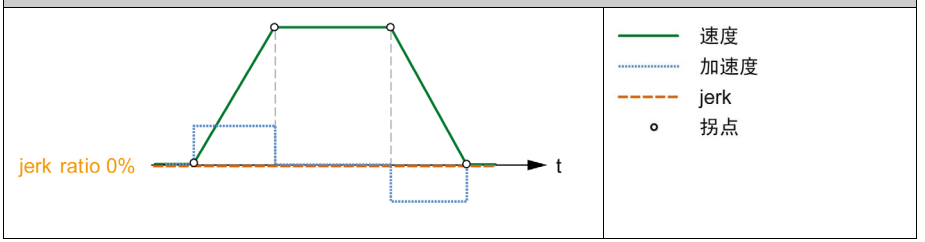
加速度是指从**启动速度**开始到目标速度为止，速度变化的比率。减速度是从目标速度开始到**停止速度**为止的速度变化比率。这些速度变换是由 PTO 功能根据 Acceleration、Deceleration 和 JerkRatio 参数（后跟**梯形**或 **S 曲线**配置文件）来隐式管理。

呈梯形的加速度/减速度斜坡

将 jerk 率参数设置为 0 时，加速度/减速度斜坡将呈梯形。

以赫兹/毫秒表示，acceleration 和 deceleration 参数代表速度变化比率。

JerkRatio 0%：恒定加速度/减速度

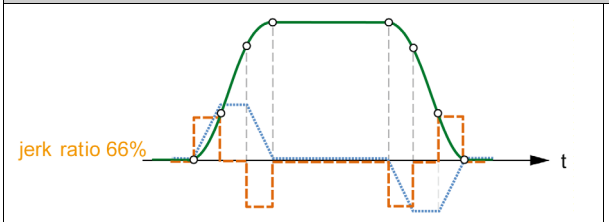


呈 S 形曲线的加速度/减速度斜坡

当 jerk 率参数大于 0 时，加速度/减速度斜坡将呈S 形曲线。

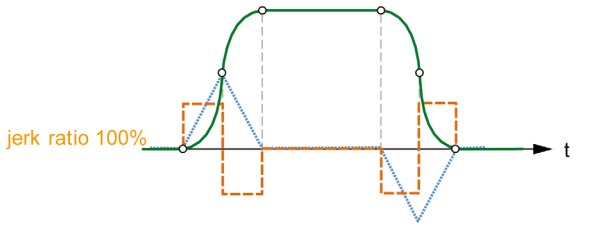
S 曲线斜坡用于可控制高惯量的应用中，或是用于需要操作易碎物品或液体的应用中。如以下各图所示，S 曲线斜坡能够以更加平稳的方式逐步进行加速/减速：

JerkRatio 66%：有 2/3 的加速和减速时间用在增加和减少加速度和减速度值上。



速度
加速度
jerk
拐点

JerkRatio 100%：时间全都用在增加和减少加速度和减速度值上。



速度
加速度
jerk
拐点

注意： JerkRatio 是加速度和减速度的通用参数值，因此凹凸时间均等。

S 曲线斜坡对于加速度/减速度的影响

无论 JerkRatio 参数为何值，加速度/减速度的持续时间都维持不变。要维持这一持续时间，请不要使用功能块（Acceleration 或 Deceleration 参数）中配置的加速度或减速度。

应用 JerkRatio 时，加速度/减速度会受影响。

应用 100% JerkRatio 时，加速度/减速度将是 Acceleration/Deceleration 参数所配置值的两倍。

注意： 如果 JerkRatio 参数值无效，则重新计算该值，以反映 MaxAccelerationAppl 和 MaxDecelerationAppl 参数。
在以下情况下 JerkRatio 无效：

- 其值大于 100。在这种情况下，应用的 JerkRatio 值为 100。
- 其值小于 0。在这种情况下，应用的 JerkRatio 值为 0。

探测器事件

描述

探测器输入通过配置启用，使用 MC_TouchProbe_PTO 功能块激活。

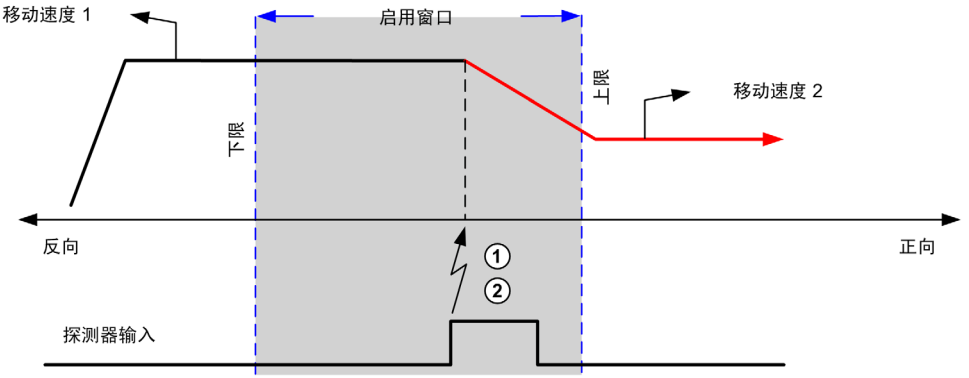
探测器输入用作事件，以便：

- 捕捉位置，
- 启动一个独立于任务的运动。

运动触发器

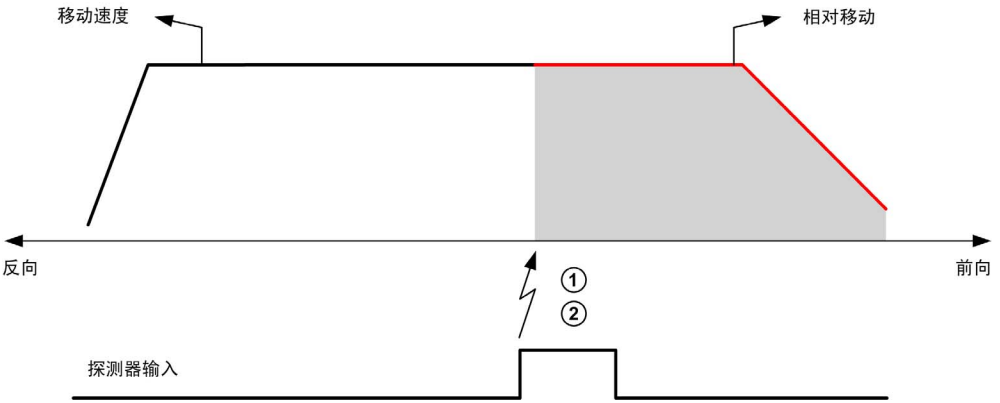
运动功能块的 BufferMode 输入必须设置为 seTrigger。

下例显示了使用启用窗口更改目标速度的时序：



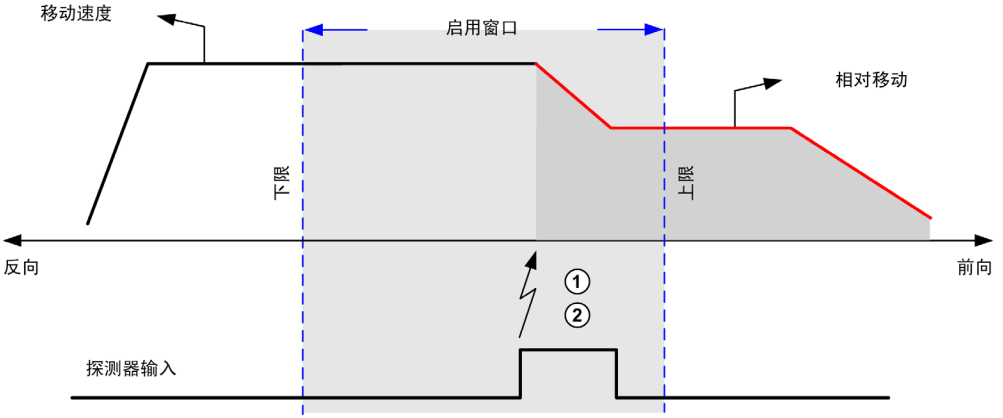
1 捕捉位置计数器值
2 触发 Move Velocity 功能块

下图示例为一个预定义行程的简单运动，无启用窗口：



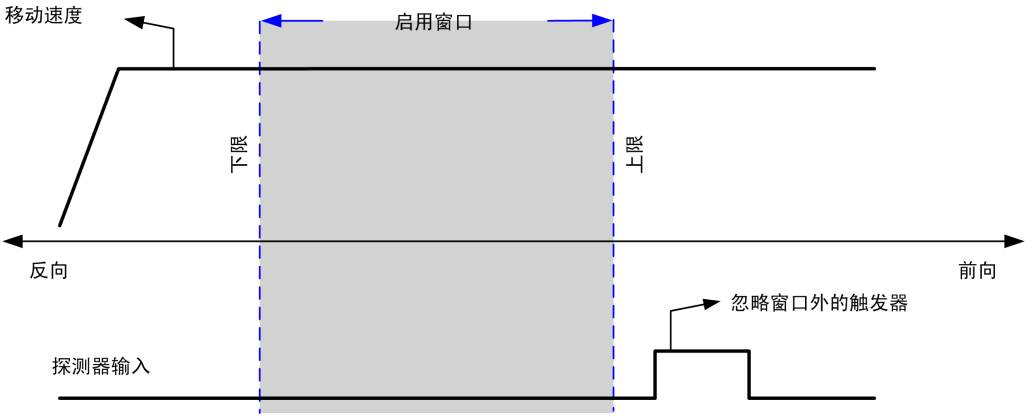
1 捕捉位置计数器值
2 触发 Move Relative 功能块

下图示例为一个预定义行程的复杂运动以及启用窗口：



1 捕捉位置计数器值
2 触发 Move Relative 功能块

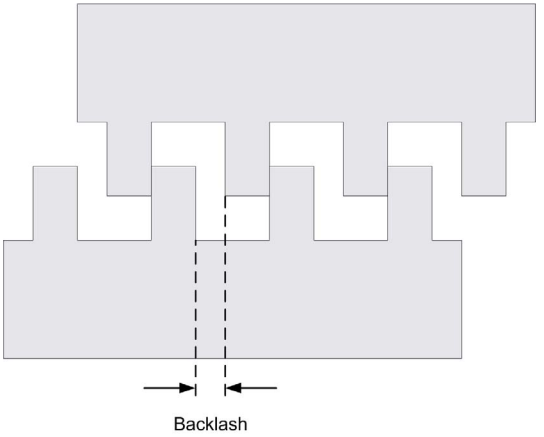
下例显示了启用窗口之外的触发事件：



反向间隙补偿

描述

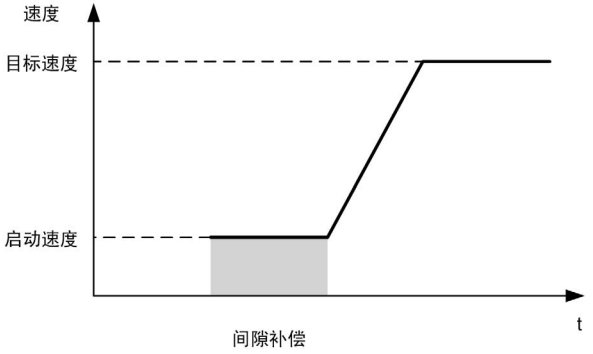
反向间隙补偿参数定义为，当运动反转时为了补偿齿轮间的机械间隙（反向间隙）而需要的运动量：



注意： 该功能不考虑外部运动源，如惰性运动或其他形式的诱导运动。

以脉冲数设置间隙补偿（0...65535，缺省值为 0）。设置后，将在每次方向反转时首先以启动速度发射指定的脉冲数，然后再执行编程的运动。不会将间隙补偿脉冲增加到位置计数器。

下图显示了间隙补偿：



注意：

- 在最初开始运动之前，该功能无法知道要补偿的间隙量。因此反向间隙补偿只有在执行第一次移动并且在第一次反转方向时进行补偿后才会激活。
- 如果接收到中止命令或者在反向间隙补偿以前检测到错误，则绝对位置仍保持不变。
- 在中止命令后，反向间隙在新的移动启动时从当前反向间隙位置开始继续。

有关更多详情，请参考配置脉冲序列输出 (参见第 *320* 页)。

定位限制

简介

可以设置正向和反向限位以控制两个方向的运动边界。硬件和软件限位都由控制器进行管理。

硬件和软件限位开关仅用于管理控制器应用程序中的边界。它们不会取代接线到驱动器的任何功能安全性限位开关。将功能安全性限位开关接线到驱动器之前，必须激活控制器应用程序限制。在任何情况下，超出现有文档范围、基于安全性分析部署的功能安全性架构的类型包括但不限于：

- 根据 EN/ISO 12100 进行的风险评估
- 根据 EN 60812 进行的失效模式与后果分析

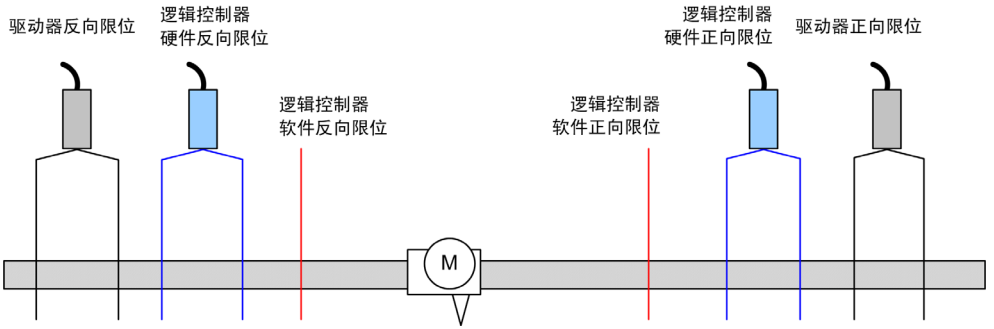
⚠ 警告

意外的设备操作

确保在设计机器的过程中，根据 EN/ISO 12100 进行风险评估。

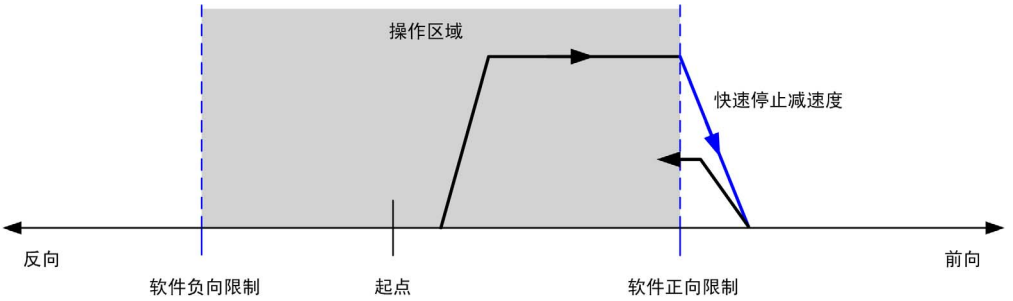
不遵循上述说明可能导致人员伤亡或设备损坏。

下图显示了硬件和软件限位开关：



控制器硬件或软件限位交叉之后，应用程序将检测到错误并执行快速停止减速：
- 正在执行的功能块检测到错误状态。

要清除轴错误状态，并返回 **Standstill** 状态，需要执行 MC_Reset_PTO，因为当轴一直处于限制（功能块终止时 ErrorId=InvalidDirectionValue）之外时，所有运动命令都将被拒（请参阅 PTO 参数 (参见第 *348* 页) EnableDirPos 或 EnableDirNeg））。在这些情况下，只能反向执行运动命令。



软件限位

可以设置软件限位以控制两个方向的运动边界。

在配置屏幕中启用和设置限位值，例如：
- 正向限位 > 反向限位
- 值在 -2,147,483,648 到 2,147,483,647 范围内

注意： 启用时，在成功建立原点（即轴已建立过原点，MC_Home_PTO（参见第 *376* 页））后软件限位才有效。

硬件限位

注意： 当限位输入为 FALSE 时不管传感方向为何，这些移动限制都有效。当它们返回 TRUE 时，会删除移动限制并对硬件限位进行功能性重置。因此，先使用复位输出指令的下降沿触点，然后再使用此功能块。然后使用这些位控制这些功能块输入。在完成操作后，设置这些位以恢复正常操作。

⚠ 警告

意外的设备操作

- 确保控制器硬件限位开关在应用程序的设计和逻辑中集成。
- 在允许充足制动距离的位置安装控制器硬件控制开关。

不遵循上述说明可能导致人员伤亡或设备损坏。

注意： 充足制动距离取决于正在运动的设备的最大速度、最大负载（质量）和快速停止减速度参数的值。

第14.2节
配置

概述

本节介绍如何配置 PTO 通道和关联参数。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
PTO 配置	320
脉冲输出模式	321
Motion Task Table	322

PTO 配置

概述

要配置 Pulse Generator 资源，请参阅配置脉冲发生器 (参见第 *78* 页)。

要配置 Pulse Generator 资源作为 PTO，请参阅 PTO 配置 (参见第 *84* 页)。

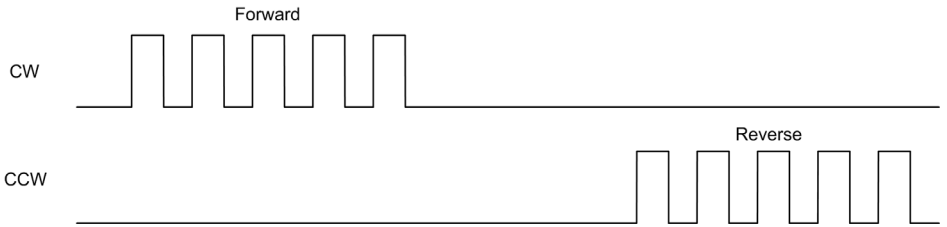
脉冲输出模式

概述

有两种可能的输出模式：
- 顺时针/逆时针
- 脉冲/方向

顺时针 (CW)/逆时针 (CCW) 模式

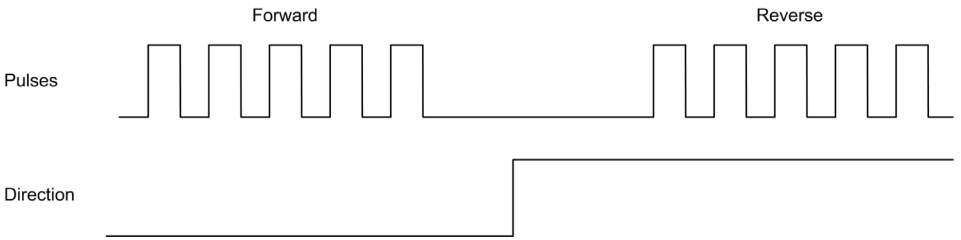
此模式生成一个定义电机运转速度和方向的脉冲串信号。此信号在第一个 PTO 通道（仅 PT00）上实现。



注意： PT01 不适用此模式。

脉冲/方向模式

此模式在 PTO 通道上生成两个信号：
- 脉冲信号提供电机运转速度 (Pulses)。
- 方向信号提供电机旋转方向 (Direction)。



特殊情况

特殊情况	描述
冷重启 (%S0=TRUE) 的影响	● 轴的状态设置为 Disabled。 ● 此 PTO 功能块已初始化。
控制器停止时的效果	轴的状态设置为 ErrorStop。
在线修改的结果	无

Motion Task Table

概述

Motion Task Table 是运动功能块的编程方案，专门用于重复性运动序列。在配置时对轴的运动序列进行定义（序列可以比作混合了多种运动的工艺程序）

Motion Task Table 可专用于多个轴，并提供配置的运动序列的图形概览。

使用 MC_MotionTask_PTO 功能块执行 Motion Task Table。 当此表由 MC_MotionTask_PTO 功能块调用时，它需要与特定轴关联。Motion Task Table 用于 MC_MotionTask_PTO 功能块使用的轴。多个 MC_MotionTask_PTO 功能块可以同时执行相同的 %MTMotion Task Table 实例。

特性

Motion Task Table(%MT) 实例的最大数目为 4 个。

Motion Task Table 包含单轴运动的序列：
- 序列是一连串的连续步骤。
- 每个步骤定义运动的参数。
- 每个步骤使用专门的运动功能块实例。

可在 Motion Task Table 中使用的移动：
- 绝对移动
- 相对移动
- 停止
- 设置位置
- 移动速度

配置 Motion Task Table

运动任务表助手使您可以按有序序列配置每个移动，并展示预计的全局移动轮廓线。

要显示**运动任务表助手**，请执行以下操作：

步骤	动作
1	选择**编程** → **工具**模块选项卡，然后单击硬盘树中的 **PTO 对象** → **运动任务表**以显示 Motion Task Table 属性。 工具提示运动任务表属性
2	单击 **[...]** 以配置 Motion Task Table。

Motion Task Table 属性窗口说明：

参数	可编辑	值	缺省值	描述
已配置	否	True/False	False	指示 Motion Task Table 是否包含配置的步骤。
地址	否	%MTx	%MTx	显示 Motion Task Table 的地址， 其中 x 是表编号。
符号	是	–	–	可用于指定与 Motion Task Table 关联的符号。 双击单元格以编辑此字段。
配置	是	**[...]** (按钮)	已启用	可使用**运动任务表助手**配置移动的序列。
注释	是	–	–	可用于指定与 Motion Task Table 关联的备注。 双击单元格以编辑此字段。

运动任务表助手：

运动任务表助手
步骤
步骤 类型 位置 距离 速度 加速度 十进制 抖动率 下一步 事件 延迟 软件对象
1 MC_MoveAbs_P 2000 5000 20 50 0 完成 10 %MC_MOVEA
2 MC_MoveRel_P* 5000 7500 20 100 0 完成 0 %MC_MOVER
3 MC_MoveRel_P* 5000 4000 20 200 0 软件事件 %M1 1000 %MC_MOVER
4 MC_Halt_PTO 1 0 完成 0 %MC_HAL_PT
5 无
6 无
使用检测器事件范围
第一位置 -2147483648
最后位置 2147483647
运动概述
下面所示图表可能不代事实际事件。有关更多信息，请参见产品文档。
速度 (Hz)
8000
7000
6000
5000
4000
3000
2000
1000
0
1
2
3
4
0
1000
2000
3000
4000
5000
6000
时间（毫秒）
应用
取消

运动任务表助手主区域：

- **步骤** ：列出单轴移动的序列和每个移动的输入参数。
- **运动概述** ：单击刷新按钮或按 **F5** 以生成步骤序列所形成的移动的图形视图。
  该曲线提供了运动的一般概述。曲线以下列假设为前提：
  - 初始位置为 0。
  - 未启用位置限制。
  - 使用了轴默认运动配置参数。
  - 步骤完成后延迟 100 毫秒发生事件（检测器输入，POU）。
  - %MWx 延迟用图形表示为 100 毫秒延迟。

步骤窗口说明：

参数	值	缺省值	描述
步骤	1...16	–	序列中的单轴移动数目。
类型	无 绝对移动 相对移动 停止 设置位置 移动速度	无	运动命令。 运动命令使用**软件对象**参数中指示的一个运动功能块实例。
Pos	请参见每个软件对象功能块参数值。	*空*	移动参数是分配给步骤的软件对象的参数。 参数描述： ● Pos：位置 ● Distance：距离 ● Vel：速度 ● Acc：加速度 ● Dec：减速度 ● Jerk ratio：反射比 **注意：** 移动速度运动命令的 Vel 参数是速度和方向的组合。在表中，MC_MoveVel_PTO 运动命令的速度范围是： - 最大速度...+ 最大速度。负速度指反方向， 正速度指正方向。
Distance			
Vel			
Acc			
Dec			
Jerk ratio			

参数	值	缺省值	描述
Next step	Done / In velocity、Blending previous、Probe input event、SW event、Delay	*empty*	执行表序列中的下一步骤需要满足的条件。 条件说明： ● Done / In velocity： ❍ Done：完成当前步骤时执行下一步骤。此参数适用于除移动速度之外的不同运动命令。 ❍ In velocity：达到请求的速度时执行下一步骤。此参数仅可用于移动速度运动命令。 ● Blending previous：下一步骤的速度在此步骤的终点位置处与此速度相混合。 ● Probe input event：在检测器输入上检测到定义的事件时执行下一步骤。 在**事件**参数中定义沿。 输入字段在**步骤**窗口的底部打开，**使用检测器事件范围**，如下一表格中所述。 **注意：** 对每个 Motion Task Table 可使用一次**检测器输入事件**。 ● SW event：当**事件**参数中设置的存储器字位地址 (%Mx) 设置为 1 时，执行下一步骤。 ● Delay：当延迟（从步骤起始时开始）结束时执行下一步骤。在**延迟**参数中定义延迟。 **注意：** 当 Probe input event、SW event 或 Delay 事件发生时，执行下一步，即使当前步骤尚未完成。
Event	– 0/1 %Mx	*empty*	**事件**值完成 Next step 参数中所述的一些条件。 Next step 选项和相应的 Event 选择： ● Probe input event: ❍ 0：下降沿 ❍ 1：上升沿 **注意：** 检测器输入事件独立于应用任务循环和运动任务循环。 ● SW event: 存储器位 %Mx。 **注意：** %Mx 每 4 毫秒评估一次。

参数	值	缺省值	描述
Delay	0...65535 %MWx	*empty*	Delay 值代表执行下一步骤之前的时间量。取决于 。 Next step 参数值，从步骤开始或结束开始评估 Delay。 ● Done / In velocity：延迟从当前步骤 Done 或 In Velocity 时开始。 ● Blending previous: 不可用. ● Probe input event 和 SW event：延迟从步骤开始时开始。 ❍ 如果事件没事发生，延迟结束时将发生超时，然后再继续下一步骤。 ❍ 如果在延迟结束前发生事件，将执行下一步骤，并取消延迟超时。 **注意：** 如果 Delay 保持其默认值 (0)，运动命令等待检测器输入或软件事件发生，并且不会发生超时。 ● Delay：延迟从步骤开始时开始。 延迟结束时执行下一步骤。 **注意：** 在应用程序 POU 中不可更改当前值，而 %MWx 值必须通过应用程序 POU 设置。如果 MC_ReadPar_PTO 或 MC_WritePar_PTO 设定为 ParNumber = 1000（延迟），Motion Task Table Delay 参数不可更改。
Software Objects	%MC_MOVEABS_PTOx %MC_MOVEREL_PTOx %MC_HALT_PTOx %MC_SETPOS_PTOx %MC_MOVEVEL_PTOx	*empty*	显示分配给步骤的软件对象。它由系统分配，并且为只读参数。这些软件对象是功能块实例。
Symbol	–	*空*	用于指定与软件对象步骤相联的符号。 双击单元格以编辑此字段。

步骤窗口中的使用 PROBE 事件范围参数：

参数		值	缺省值	描述
使用 PROBE 事件范围		True/False	False	值为 TRUE 时，仅在第一位置和最后一个位置之间所定义的位置范围内识别触发事件。 如果下一步骤设置为 Motion Task Table 中的 Probe input event，则可以更改此参数。
	第一位置	– 2147483648... 2147483647 %MDx	– 2147483648	**注意：** 第一位置必须小于最后一个位置。
	最后一个位置	– 2147483648... 2147483647 %MDx	2147483647	
有关位置范围对触发的影响的图示，请参见检测器事件 *(参见 Modicon M241 Logic Controller, PTOPWM, 库指南)*一节。 **注意：** 检测到触发事件的位置没有记录。				

管理步骤参数和事件

步骤中定义的参数和事件只有在开始执行步骤时有效，因此：

- 通过应用程序修改的步骤参数值只有在步骤激活前修改才有效。在 POU 中使用系统分配的软件对象参数可以修改参数。
- 存储对象值（%MW 或 %MWx）只有在步骤激活前更新才有效。
- 事件只在步骤激活期间评估。在 Probe input event 的情况下，无法检测到步骤激活后发生的事件。

管理 Motion Task Table 中使用的功能块实例

系统分配的软件对象实例：

- 不可用于应用程序 POU 中控制轴运动。
- 在 Motion Task Table 的执行期间，系统不会更新输出参数。即，输入位和输出参数无效。
- 输入参数：
  - 不可在软件对象实例编辑器或**编程**选项卡中修改。
  - 可用于在应用程序 POU 中动态修改 Motion Task Table。要动态地修改系统分配的软件对象实例输入参数，使用参数地址或其关联符号。

  注意： 可以修改执行步骤，但要到下一次执行时才会应用修改的内容。

  Motion Task Table 中所述移动的示例：
  - 步骤：2
  - 移动类型：相对移动
  - 软件对象: %MC_MOVEREL_PTO1
  - 符号：Move_Relative_Label2

  在上一示例中，使用下列一种语法可通过程序修改速度输入参数：
  - %MC_MOVEREL_PTO1.Vel
  - Move_Relative_Label2.Vel

Motion Task Table 中所用功能块实例的管理：

- 当配置 Motion Task Table 时，预留的功能块实例设置为**已使用**。
- 如果保留特定功能块的所有实例，则不可再使用关联的移动类型。

第14.3节
编程

概述

本节列出了用于编程 PTO 功能的功能块，并描述了如何添加或移除这些功能块。

本节包含了哪些内容？

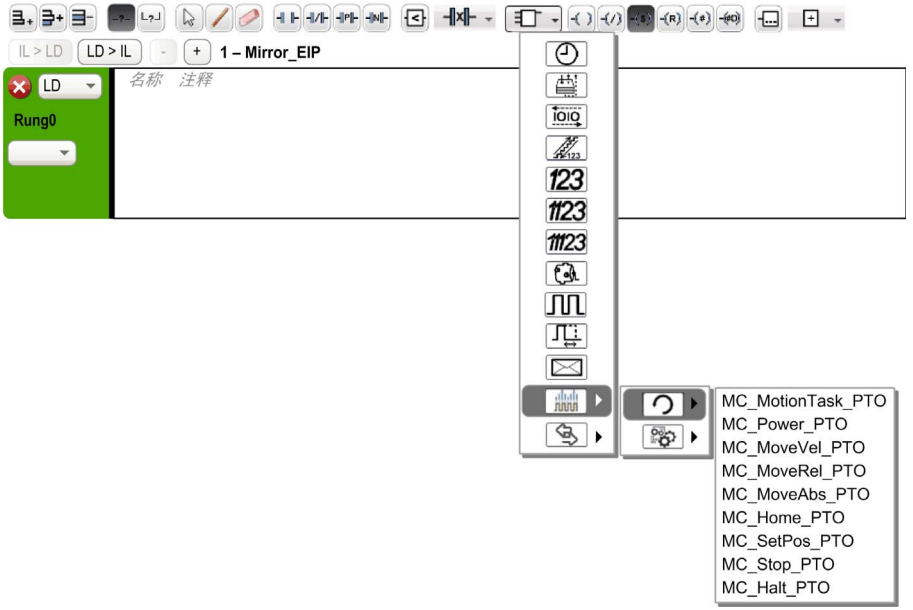
本节包含了以下主题：

主题	页
添加/删除功能块	330
PTO 功能块	331

添加/删除功能块

添加功能块

遵照以下步骤添加 PTO 功能块的实例：

步骤	动作
1	选择**编程**选项卡。
2	选择**功能块 → PTO → 管理**或**功能块 → PTO → 运动**，如下图所示： 
3	单击梯级以放置所选功能块。
4	关联功能块的输入/输出变量。

注意： 在**配置**选项卡中设置参数。
有关详细信息，请参阅 PTO 配置 (参见第 *84* 页)。

删除功能块

遵照以下步骤删除 PTO 功能块的实例：

步骤	动作
1	在**编程**选项卡中，单击此功能块的实例。
2	按**删除**以删除所选功能块。

PTO 功能块

功能块

在 SoMachine Basic 中使用以下功能块对 PTO 功能进行编程：

类别	功能块	描述
运动（单轴）	MC_MotionTask_PTO (参见第 *357* 页)	调用运动任务表。
	MC_Power_PTO (参见第 *361* 页)	向轴供电，将轴状态从 Disabled 切换为 Standstill。当 %MC_Power_PTO.Status 位为 FALSE 时，不能对该轴运行任何运动功能块。
	MC_MoveVel_PTO (参见第 *364* 页)	使指定轴以指定速度移动，并且将轴的状态转换为 Continuous。在达到软件限位、中止移动触发或检测到转换为 ErrorStop 状态之前，将保持此连续运动。
	MC_MoveRel_PTO (参见第 *368* 页)	使指定轴以指定速度运动一段相对距离，并且将轴的状态转换为 Discrete。 执行此功能块时，以当前位置作为移动的起始位置，增加设定的位置。
	MC_MoveAbs_PTO (参见第 *372* 页)	使指定轴以指定速度运动至设定位置，并且将轴的状态转换为 Discrete。 如果轴无原点（未定义绝对定位），功能块将终止并将 Error 设置为 TRUE。在此情况下，ErrorId 设置为 InvalidAbsolute。
	MC_Home_PTO (参见第 *376* 页)	设定目标轴执行绝对定位，并且将轴的状态转换为 Homing（参见第 *335* 页)。此序列的详细信息取决于 Homing 配置参数设置。
	MC_SetPos_PTO (参见第 *379* 页)	修改轴的坐标，而不产生任何物理运动。
	MC_Stop_PTO (参见第 *381* 页)	运动任务停止，并将轴的状态转换为 Stopping。它将中止正在进行的任何运动控制任务。
	MC_Halt_PTO (参见第 *384* 页)	停止运动控制任务，直到速度达到零，并将轴的状态转换为 Discrete。将 Done 输出设置为 TRUE 后，状态转换为 Standstill。

类别	功能块	描述
管理	MC_ReadActVel_PTO (参见第 *388* 页)	返回轴的速度值。
	MC_ReadActPos_PTO (参见第 *390* 页)	返回轴的位置值。
	MC_ReadSts_PTO (参见第 *392* 页)	返回轴的状态 (参见第 *352* 页)。
	MC_ReadMotionState_PTO (参见第 *394* 页)	返回轴的运动状态。
	MC_ReadAxisError_PTO (参见第 *396* 页)	返回轴的控制错误 (如果有) 。
	MC_Reset_PTO (参见第 *398* 页)	复位所有与轴相关的错误，如有可能，允许从状态 ErrorStop 转换为 Standstill。它不影响功能块实例的输出。
	MC_TouchProbe_PTO (参见第 *400* 页)	在探测器输入上激活触发事件。此触发事件用于记录轴位置和 /或启动缓冲移动。
	MC_AbortTrigger_PTO (参见第 *402* 页)	中止连接到触发事件的功能块 (例如 MC_TouchProbe_PTO) 。
	MC_ReadPar_PTO (参见第 *404* 页)	从 PTO 获取参数。
	MC_WritePar_PTO (参见第 *406* 页)	将参数写入 PTO。

注意： 根据运动状态图 (参见第 *352* 页)，运动功能块按照轴的位置进行操作。管理功能块不影响运动状态。

注意： MC_Power_PTO 是必不可少的，只有执行过该功能块之后，才能发出移动命令。

⚠ 警告

意外的设备操作

- 请勿在不同的程序任务中使用同一个功能块实例。
- 请勿在功能块正在执行时更改功能块参考 (AXIS)。

不遵循上述说明可能导致人员伤亡或设备损坏。

PTO 特性

PTO 通道最多可以有六个物理输入：

- 其中两路输入通过配置分配给 PTO 功能，并在出现输入的上升沿时纳入考虑范畴：
  - 参考输入（%I0.8 用于 %PT00 并且 %I0.10 用于 %PT01）
  - 索引输入（%I0.9 用于 %PT00 并且 %I0.11 用于 %PT01）
  - 探测器输入（%I0.12 用于 %PT00 并且 %I0.13 用于 %PT01）
- 另外三路输入分配给 MC_Power_PTO 功能块。它们都没有固定分配（不在屏幕画面中进行配置），并通过任何其他输入读取：
  - DriveReady 输入
  - 正向限位输入
  - 反向限位输入

注意： 这些输入像任何其他常规输入一样进行管理，但是在分配给 MC_Power_PTO 功能块时由 PTO 功能使用。

注意： 帮助防止越程需要正向和反向限位输入。

⚠ 警告

意外的设备操作

- 确保控制器硬件限位开关在应用程序的设计和逻辑中集成。
- 在允许充足制动距离的位置安装控制器硬件控制开关。

不遵循上述说明可能导致人员伤亡或设备损坏。

PTO 通道最多可以有三个物理输出：

- 其中两路输出是强制输出，用于管理 PTO 功能的输出模式。它们都有固定分配，且必须通过配置启用：
  - CW / CCW（%Q0.0 和 %Q0.1 分别仅用于 %PT00）
  - 脉冲（%Q0.0 用于 %PT00 并且 %Q0.1 用于 %PT01）
- 另一路输出 DriveEnable 已与 MC_Power_PTO 功能块关联。它没有固定分配，并通过任何其他输出写入。

PTO 功能具有以下特性：

特性	值
通道数	2
轴数	每通道 1 个
位置范围	-2,147,483,648...2,147,483,647（32 位）
最小速度	0 Hz
最大速度	100 kHz（针对 40/60 占空比和最大 200 mA）
最小步长	1 Hz
速度精度	1 %

特性	值
加速度/减速度（最小值）	1 Hz/ms
加速度/减速度（最大值）	100 kHz/ms
原点补偿	-2,147,483,648...2,147,483,647（32 位）
软件限位范围	-2,147,483,648...2,147,483,647（32 位）

第14.4节
原点模式

概述

本节介绍 PTO 原点模式。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
原点模式	336
位置设置	338
长参考	339
短参考无反向	341
短参考反向	343
INDEX 短参考	345
原点补偿	346

原点模式

描述

原点是用于为绝对移动确定参考点或起点的方法。

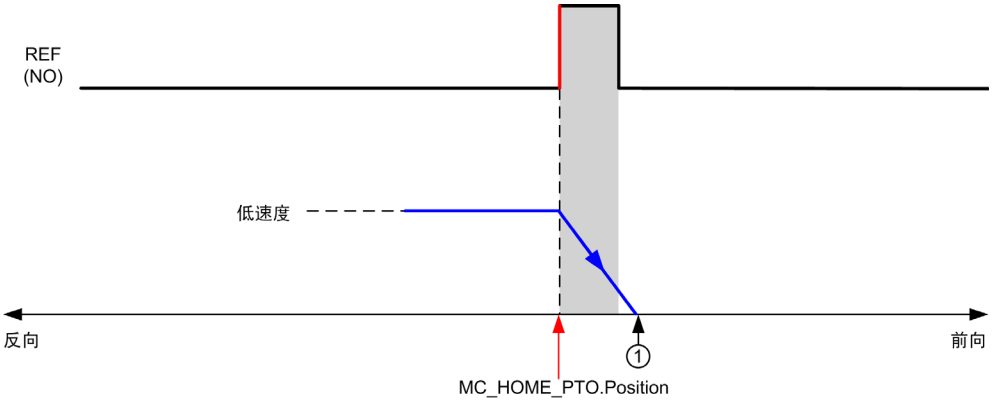
可以使用各种方法执行原点建立。M200 PTO 通道提供多种标准原点运动类型：

- 位置设置 (参见第 *338* 页)，
- 长参考 (参见第 *339* 页)，
- 短参考无反向 (参见第 *341* 页)，
- 短参考反向 (参见第 *343* 页)，
- INDEX 的短参考 (参见第 *343* 页)。

原点位置

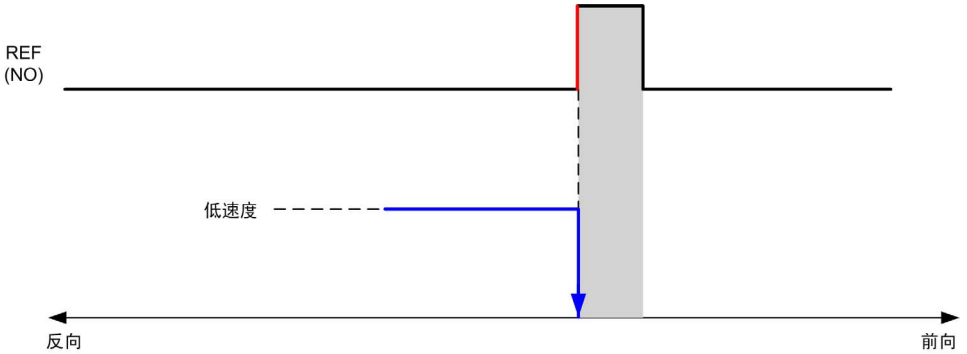
使用外部开关作为原点，并在开关跳变沿上定义原点位置。然后，减速直到停止。

因此，轴在运动序列结束时的实际位置可能不同于在功能块上设置的位置参数。



REF (NO) 参考点（常开）

为简化原点模式图中某次停止的表述，用以下图示表示轴的实际位置：



REF (NO) 参考点（常开）

限位

硬件限位是 MC_Home_PTO 功能块正确运行所必需的（定位限制 (参见第 *317* 页)和 MC_Power_PTO (参见第 *361* 页））。根据使用原点模式请求的运动类型，硬件限位可帮助确保功能块接受行程终端。

当以远离参考开关方向启动原点操作时，硬件限位用于以下两方面：

- 指示需要反向以朝着参考开关移动轴，或
- 指示因在到达行程末端之前未找到参考开关而检测到错误。

对于允许反向的原点运动类型，当运动达到硬件限位时，轴将使用配置的减速度停止，然后向相反方向恢复运动。

在不允许反向的原点运动类型中，当运动达到硬件限位时，原点步骤中止，轴将以快速停止减速度停止。

⚠ 警告

意外的设备操作

- 确保控制器硬件限位开关在应用程序的设计和逻辑中集成。
- 在允许充足制动距离的位置安装控制器硬件控制开关。

不遵循上述说明可能导致人员伤亡或设备损坏。

注意： 充足制动距离取决于正在运动的设备的最大速度、最大负载（质量）和快速停止减速度参数的值。

位置设置

描述

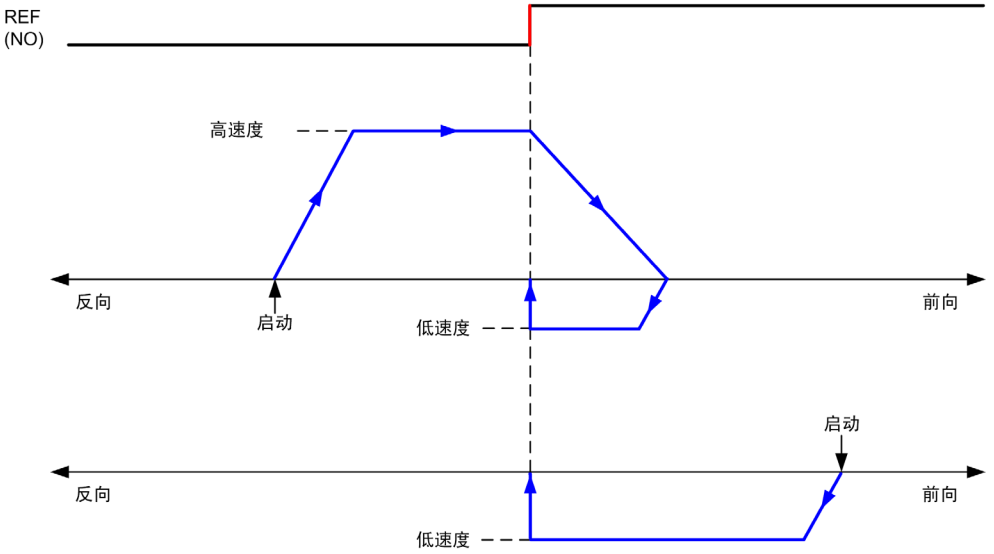
在位置设置情况下，当前位置设置为指定的位置值。不执行任何移动。

长参考

长参考：正向

反向位置上参考开关下降沿处为原点。

初始运动方向取决于参考开关的状态：

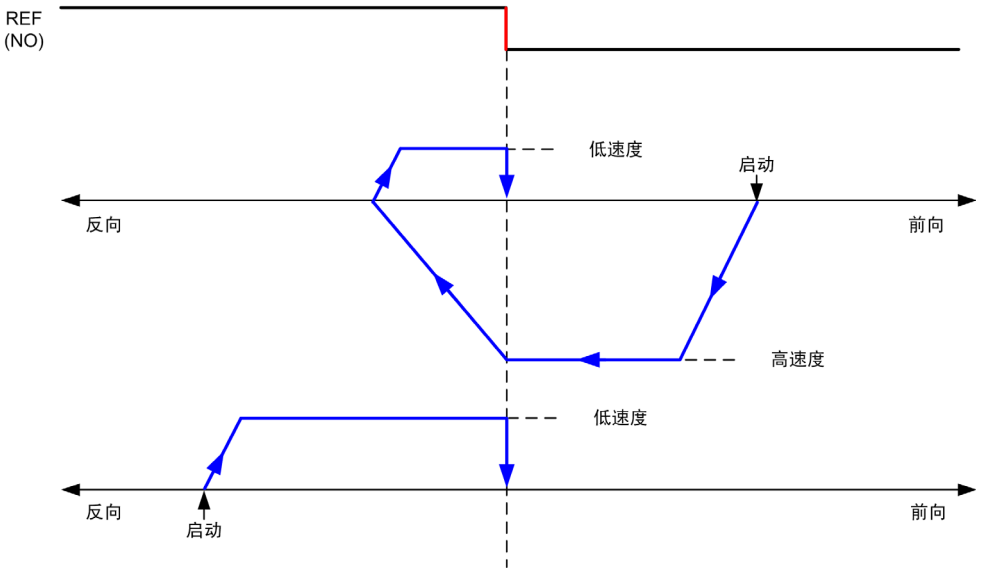


REF (NO) 参考点（常开）

长参考：反向

正向位置上参考开关下降沿处为原点。

初始运动方向取决于参考开关的状态：

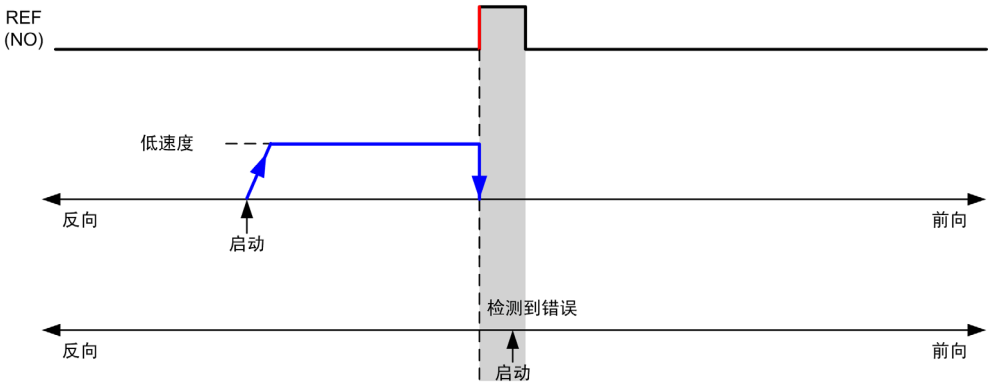


REF (NO) 参考点（常开）

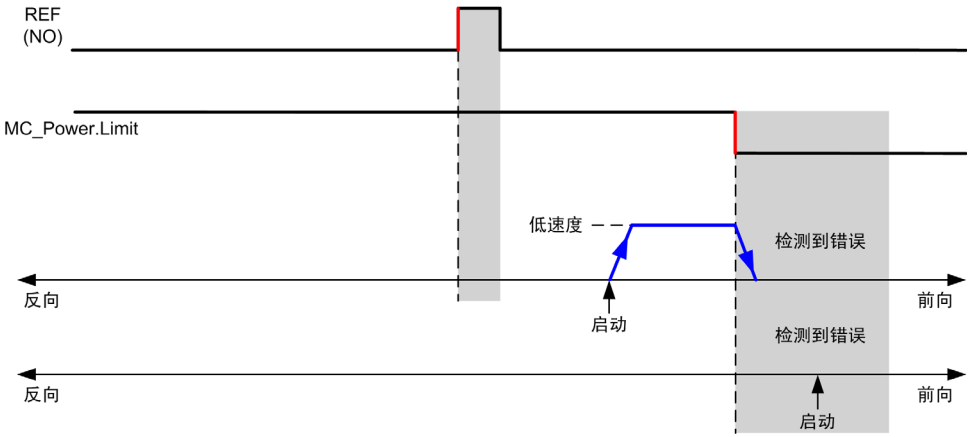
短参考无反向

短参考无反向：正向

低速正向运行至参考开关上升沿处，无反向，此上升沿处即为原点：



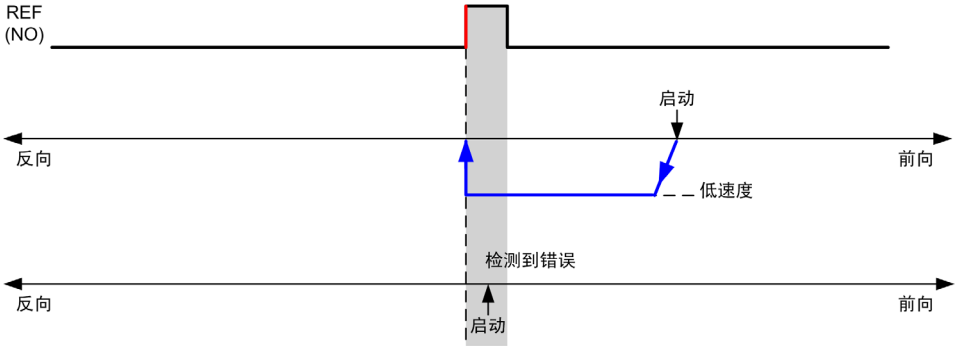
REF (NO) 参考点（常开）



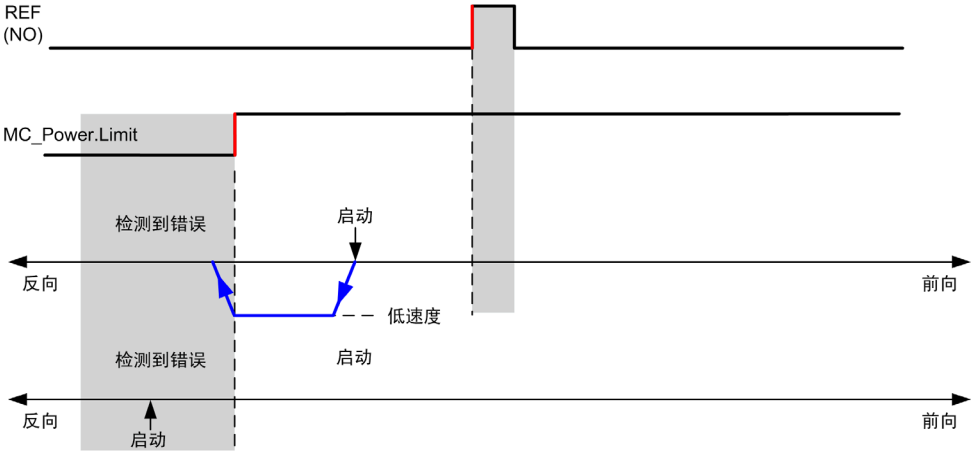
REF (NO) 参考点（常开）

短参考无反向：反向

低速反向运行至参考开关上升沿处，无反向，此上升沿处为原点：



REF (NO) 参考点（常开）



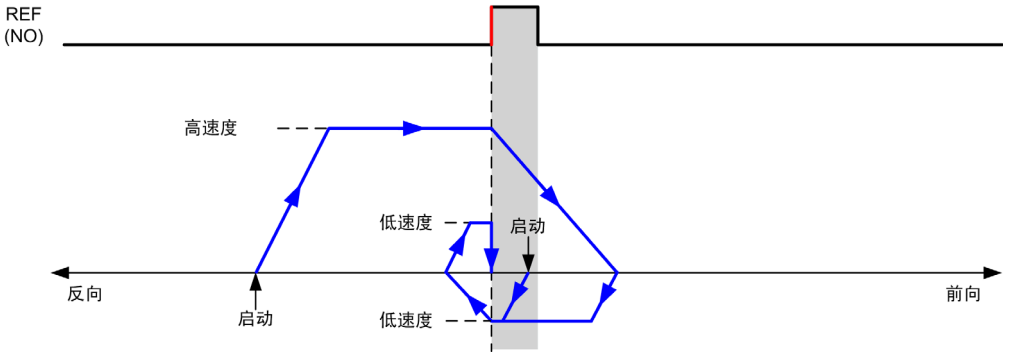
REF (NO) 参考点（常开）

短参考反向

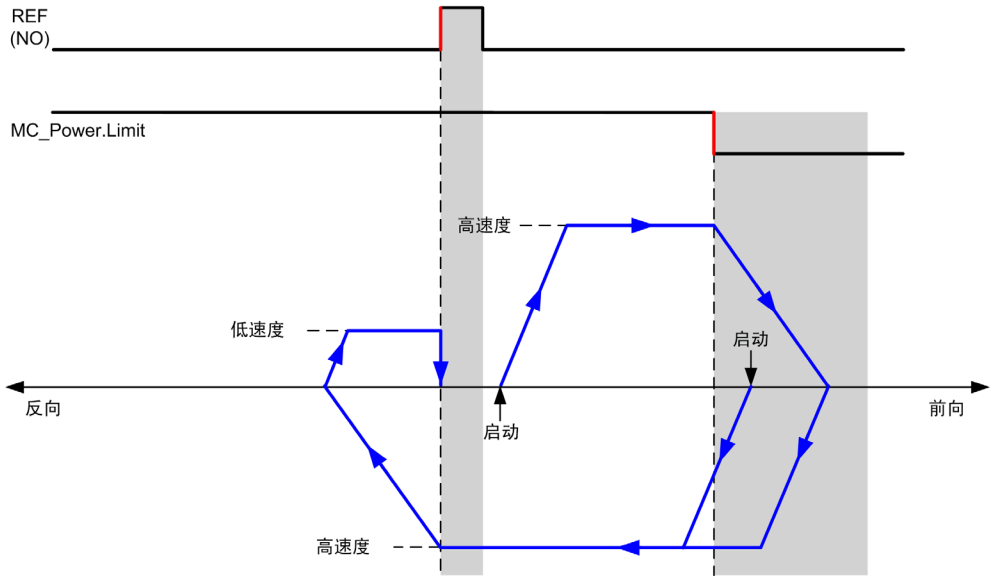
短参考反向：正向

正向位置上参考开关上升沿处为原点。

初始运动方向取决于参考开关的状态：



REF (NO) 参考点（常开）

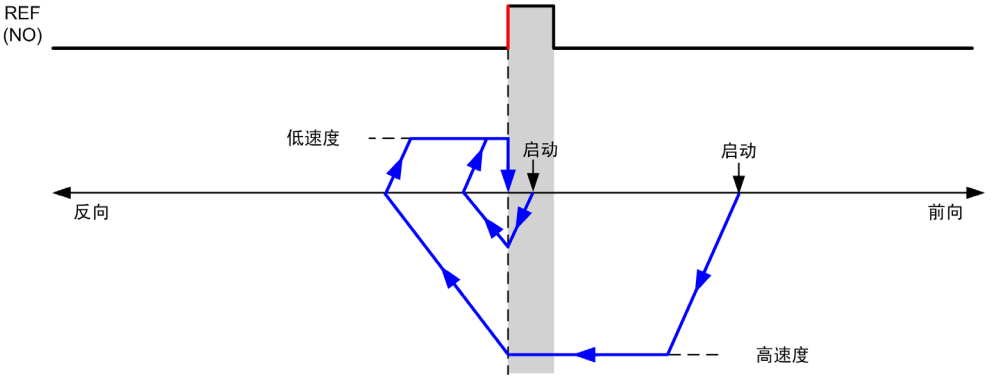


REF (NO) 参考点（常开）

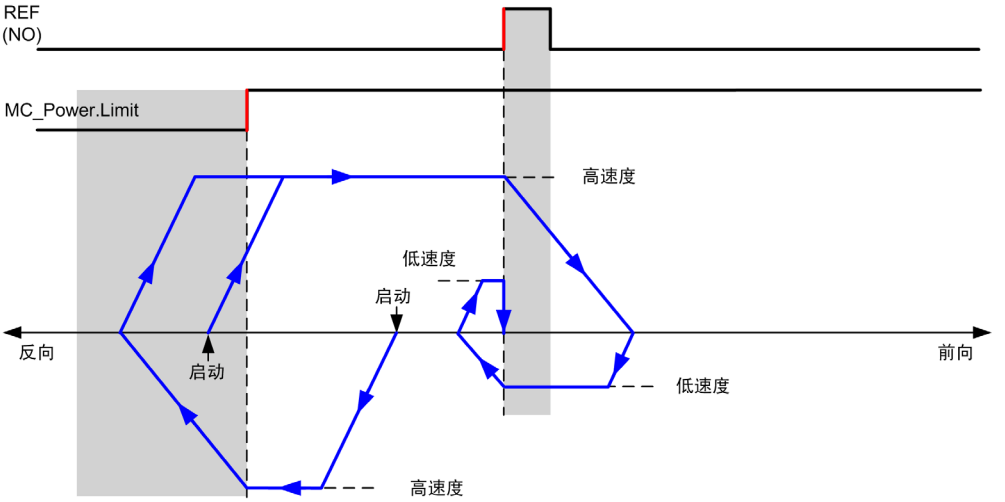
短参考反向：反向

正向位置上参考开关上升沿处为原点。

初始运动方向取决于参考开关的状态：



REF (NO) 参考点（常开）



REF (NO) 参考点（常开）

INDEX 短参考

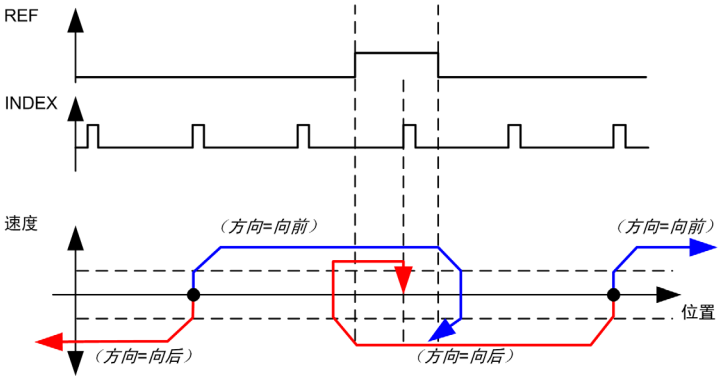
描述

PTOHome 和 PTOEnhancedHome 功能块用于将轴设置到参考位置。

INDEX 短参考回归方法使用两个原点相关输入：

- REF 输入，用作正向限位信号：在信号的上升沿（负端），轴必须更改方向。
- INDEX 输入，用作零标记信号 (Z)

下图描述 **INDEX 短参考**回归方法：



错误检测

当启用 REF 输入时，会在超出限制时在 LIMIT_FLT 状态对象中报告错误。

原点补偿

描述

如果无法通过具有足够精度的开关定义原点，则可以通过让轴移动至原点开关附近的一个特定位置来进行补偿。原点补偿允许机械原点和电气原点之前存在差别。

以脉冲数设置原点补偿（-2,147,483,648...2,147,483,647，缺省值 0）。通过配置进行设置时，将首先执行 MC_Home_PTO（参见第 *376* 页）命令，然后在指定的方向以低速输出指定的脉冲数。

注意： MC_Home_PTO 命令忙标志仅在原点补偿完成后才会释放。

第14.5节
数据参数

功能块对象代码

Direction

下表列出了方向功能块对象代码的值：

名称	值	描述
mcPositiveDirection	1	CW、向前、正向（根据**输出模式**配置设置）。
mcNegativeDirection	-1	CCW、向后、反向（根据**输出模式**配置设置）。

Buffer Modes

下表列出了缓冲模式功能块对象代码的值：

名称	值	描述
mcAborting	0	立即启动 FB（缺省模式）。 将中止正在进行的任何运动。清除移动队列。
mcBuffered	1	在当前运动完成（Done 或 InVel 位设置为 TRUE）后启动 FB。没有任何混合。
mcBlendingPrevious	3	速度与第一个 FB 的速度混合（与 FB1 的速度在 FB1 的末端位置混合）。
seTrigger	10	在探测器输入上检测到事件时立即启动 FB。 将中止正在进行的任何运动。清除移动队列。
seBufferedDelay	11	在当前运动完成（Done 或 InVel 输出设置为 TRUE）并且经过时间延迟后启动 FB。没有任何混合。 使用 MC_WritePar_PTO (参见第 *406* 页) 设置 Delay 参数，ParameterNumber 为 1000。

原点搜索

下表列出了原点搜索功能块对象代码的值：

名称	值	描述
PositionSetting	0	位置。
LongReference	1	长参考。
ShortReference_Reversal	20	短参考。
ShortReference_NoReversal	21	短参考无反向。
ShortReference_with_INDEX	22	INDEX 的短参考。

PTO 参数

下表列出了 PTO 参数功能块对象代码的值：

名称	参数编号	读/写	描述
CommandedPosition	1	读	命令的位置。
SWLimitPos (**上限**)	2	读/写	正向软件限位。
SWLimitNeg (**下限**)	3	读/写	反向软件限位。
EnableLimitPos (**启用软件限位**)	4	读/写	启用正向软件限位开关 (0...1)。
EnableLimitNeg (**启用软件限位**)	5	读/写	启用负向软件限位开关 (0...1)。
MaxVelocityAppl (**最大速度**)	9	读/写	应用程序中的最大允许轴速度 (0...100,000)。
ActualVelocity	10	读	轴的速度。
CommandedVelocity	11	读	命令的速度。
MaxAccelerationAppl (**最大加速度**)	13	读/写	应用程序中的最大允许轴加速度 (0...100,000)。
MaxDecelerationAppl (**最大减速度**)	15	读/写	应用程序中的最大允许轴减速度 (0...100,000)。
保留	16 到 999	-	为 PLCopen 标准所保留。
Delay	1000	读/写	以毫秒为单位的时间 (0...65,535) 缺省值：0
EnableDirPos	1004	读/写	启用正方向。 当值 = 0，轴上不允许正方向。本应生成正方向运动的运动功能块将因检测到 InvalidDirectionValue 错误而结束 (3006)。 如有负方向的运动进行中，并被正方向的新运动命令中断，则在进行中的负方向运动减速结束时才会检测到错误。 缺省值：1 **注意：** 在下一个移动命令或当速度再次等于 0 时才考虑更改值。

名称	参数编号	读/写	描述
EnableDirNeg	1005	读/写	启用负方向。 当值 = 0，轴上不允许负方向。本应生成负方向运动的运动功能块将因检测到 InvalidDirectionValue 错误而结束 (3006)。 如有正方向的运动进行中，并被负方向的新运动命令中断，则在进行中的正方向运动减速结束时才会检测到错误。 缺省值：1 **注意：** 在下一个移动命令或当速度再次等于 0 时才考虑更改值。

PTO 轴错误代码

下表列出了 PTO 轴错误代码的值：

名称	值	描述
NoError	0	未检测到错误。
轴控制警报		
InternalError	1000	检测到运动控制器内部错误。
DisabledAxis	1001	由于轴未准备就绪而无法开始移动或已被中止。
HwPositionLimitP	1002	超出硬件正向限位 limP。
HwPositionLimitN	1003	超出硬件反向限位 limN。
SwPositionLimitP	1004	超出软件正向限位。
SwPositionLimitN	1005	超出软件反向限位。
ApplicationStopped	1006	已停止执行应用程序（控制器处于 STOPPED 或 HALT 状态）。
OutputProtection	1007	短路输出保护在 PTO 通道上处于活动状态。请参阅系统位 (参见第 *454* 页)和系统字 (参见第 *463* 页)中 %S10 和 %SW139 的说明。
轴控制警告		
WarningVelocityValue	1100	命令的速度参数超出范围，因此速度受限于配置的速度最大值。
WarningAccelerationValue	1101	命令的加速度参数超出范围，因此加速度受限于配置的加速度最大值。
WarningDecelerationValue	1102	命令的减速度参数超出范围，因此减速度受限于配置的减速度最大值。
WarningJerkRatioValue	1103	命令的 jerk ratio 参数受限于配置的加速度或减速度最大值。在此情况下，需重新计算 jerk ratio 以遵循这些最大值。

轴控制警报将轴切换为 **ErrorStop** 状态（必须用 MC_Reset_PTO 来脱离 **ErrorStop** 状态）。轴的状态结果可通过 MC_ReadSts_PTO 和 MC_ReadAxisError_PTO 获得。

PTO 运动命令错误代码

下表列出了 PTO 运动命令错误代码的值：

名称	值	描述
NoError	0	未检测到错误。
运动状态建议警报		
ErrorStopActive	2000	由于运动被 ErrorStop 条件禁止而无法开始移动或者已被中止。
StoppingActive	2001	由于运动被控制轴的 MC_Stop_PTO 禁止（轴处于停止状态或 MC_Stop_PTO.Execute 输入保持为 TRUE）而无法开始移动。
InvalidTransition	2002	不允许转换，请参阅运动状态图 (参见第 *352* 页)。
InvalidSetPosition	2003	在轴移动时无法执行 MC_SetPos_PTO。
HomingError	2004	此模式下无法在参考凸轮上启动原点序列。
InvalidProbeConf	2005	必须配置探测器输入。
InvalidHomingConf	2006	必须为原点模式配置 Ref 输入。
InvalidAbsolute	2007	在轴未定义原点的情形下无法执行绝对移动。必须先执行原点搜索 (MC_Home_PTO（参见第 *376* 页))。
MotionQueueFull	2008	因为运动队列已满而无法缓冲移动。
InvalidTransitionMotionTask	2009	运动任务和与同一轴关联的其他运动功能块不可同时执行。
范围建议警报		
InvalidAxis	3000	功能块不适用于指定轴。
InvalidPositionValue	3001	位置参数超出限制，或者距离参数给出了一个超过限制的位置。
InvalidVelocityValue	3002	速度参数超出范围。
InvalidAccelerationValue	3003	加速度参数超出范围。
InvalidDecelerationValue	3004	减速度参数超出范围。
InvalidBufferModeValue	3005	缓冲模式未对应于有效值。
InvalidDirectionValue	3006	方向未对应于有效值，或者由于超出软件限位而使方向无效。
InvalidHomeMode	3007	原点模式不适用。
InvalidParameter	3008	对于指定的轴，参数编号不存在。
InvalidParameterValue	3009	参数值超出范围。
ReadOnlyParameter	3010	参数为只读。
InvalidStepMotionTask	3011	运动任务步进类型未定义。

运动状态警报或**范围警报**不影响轴状态，也不影响当前正在执行的任何移动或移动队列。在这种情况下，错误是局部的，仅针对适用的功能块：Error 输出设置为 TRUE，并且 ErrorId 对象输出设置为相应的 PTO 运动命令错误代码。

第14.6节
操作模式

概述

本节介绍操作模式。

本节包含了哪些内容？

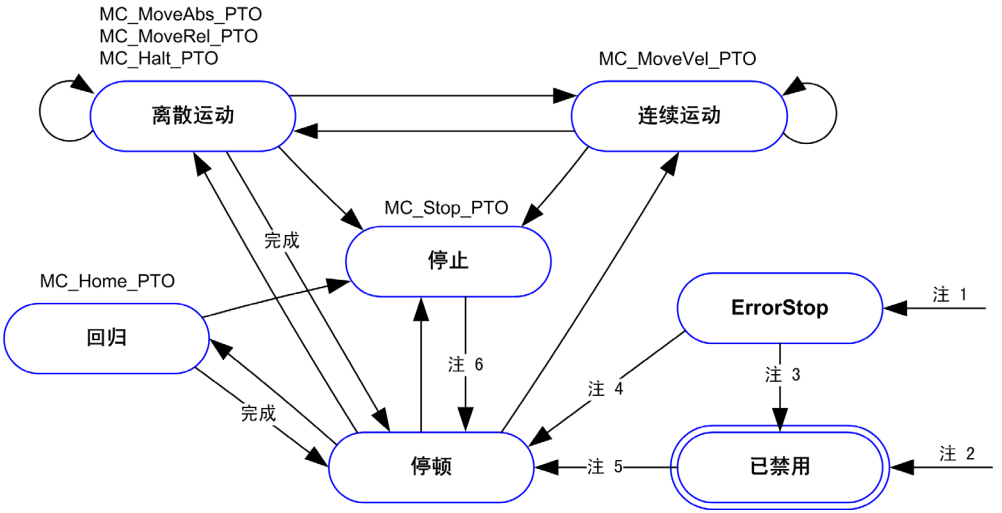
本节包含了以下主题：

主题	页
运动状态图	352
缓冲模式	354

运动状态图

状态图

轴始终处于下图中定义的状态之一；



注 1 任何状态，当检测到错误时。
注 2 除 ErrorStop 以外的任何状态，当 %MC_Power_PTO.Status = FALSE 时。
注 3 %MC_Reset_PTO.Done = TRUE 和 %MC_Power_PTO.Status = FALSE。
注 4 %MC_Reset_PTO.Done = TRUE 和 %MC_Power_PTO.Status = TRUE。
注 5 %MC_Power_PTO.Status = TRUE。
注 6 %MC_Stop_PTO.Done = TRUE 和 %MC_Stop_PTO.Execute = FALSE。

下表描述轴状态：

状态	描述
Disabled	初始轴状态，不允许执行任何运动命令。轴未建立原点。
Standstill	在接通电源后，未检测到任何错误，并且在轴上没有任何运动命令处于活动状态。允许执行运动命令。
ErrorStop	最高优先级，适用于在轴上或在控制器中检测到错误的情况。通过**快速停止减速**中止任何正在进行的移动。在适用的功能块上，Error 输出设置为 TRUE，且 ErrorId 用于设置错误代码。只要错误处于未解决状态，便会保持 ErrorStop 状态。在使用 MC_Reset_PTO 完成复位之前，不接受任何其他运动命令。
Homing	当 MC_Home_PTO 控制轴时适用。
Discrete	当 MC_MoveRel_PTO、MC_MoveAbs_PTO 或 MC_Halt_PTO 控制轴时适用。
Continuous	当 MC_MoveVel_PTO 控制轴时适用。
Stopping	当 MC_Stop_PTO 控制轴时适用。

注意： 在状态图中未列出的功能块不会影响轴的状态。
包括加速和减速斜坡在内的整个运动命令都不能超过 4,294,967,295 个脉冲。在最大频率为 100 kHz 时，将加速和减速斜坡限制为 80 秒。

运动转换表

执行当前命令时（在完成之前），PTO 通道可以按照下表响应新命令：

命令		下一个					
		Home	MoveVel	MoveRel	MoveAbs	Halt	Stop
当前	Standstill	允许	允许 (1)	允许 (1)	允许 (1)	允许	允许
	Home	拒绝	拒绝	拒绝	拒绝	拒绝	允许
	MoveVel	拒绝	允许	允许	允许	允许	允许
	MoveRel	拒绝	允许	允许	允许	允许	允许
	MoveAbs	拒绝	允许	允许	允许	允许	允许
	Halt	拒绝	允许	允许	允许	允许	允许
	Stop	拒绝	拒绝	拒绝	拒绝	拒绝	拒绝

(1) 当轴处于静止状态时，对于缓冲模式 mcAborting/mcBuffered/mcBlendingPrevious，移动立即开始。
允许 新命令开始执行，即使上一个命令尚未执行完成。
拒绝 新命令被忽略，减速停止并报错。

注意： 在运动转换过程中检测到错误时，轴将进入 ErrorStop 状态。ErrorId 设置为 InvalidTransition。

缓冲模式

描述

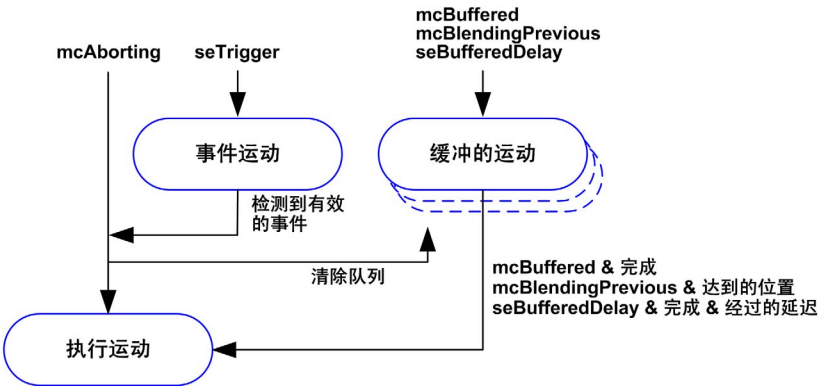
某些运动功能块有一个称为 BufferMode 的输入对象。使用此输入对象，在启动探测事件时，功能块可以立即启动，也可以进入缓冲区。

可用选项在缓冲模式功能块的对象代码 (参见第 *347* 页)中进行定义：

- 某个中止的运动 (mcAborting) 将立即启动，中止正在进行的任何移动，并清除运动队列。
- 事件运动 (seTrigger) 是中止移动，在发生探测事件 (参见第 *314* 页)时启动。
- 缓冲的运动（mcBuffered、mcBlendingPreviousseBufferedDelay）排入队列，即附加到当前正在执行或等待执行的任何移动后面，在前一个运动完成时启动。

运动队列图

下图显示了运动队列图：



缓冲区只能包含一个运动功能块。

缓冲区中存在的运动功能块的执行条件是：

- mcBuffered：在当前的连续运动处于 InVel 或当前的离散运动停止时。
- seBufferedDelay：当指定的延迟过后，从当前的连续运动处于 InVel，或从当前的离散运动停止。
- mcBlendingPrevious：达到当前功能块的位置和速度目标时。

在以下情况下清除运动队列（删除所有缓冲的运动）：

- 触发中止移动时（mcAborting 或 seTrigger）：在缓冲功能块上将 CmdAborted 输出设置为 TRUE。
- 执行 MC_Stop_PTO 功能时：在清除的缓冲功能块上将 Error 输出设置为 TRUE，ErrorId=StoppingActive（参见第 *350* 页）。
- 检测到转换为 **ErrorStop** 状态时：在缓冲功能块上将 Error 输出设置为 TRUE，ErrorId=ErrorStopActive（参见第 *350* 页）。

注意：

- 只有有效运动才能排入队列。如果功能块的执行由于 Error 输出设置为 TRUE 而终止，则不会将移动排入队列，不影响当前正在执行的任何移动，并且不会清除队列。
- 当队列已满时，将在适用的功能块上将 Error 输出设置为 TRUE，并且 ErrorId 输出返回错误 MotionQueueFull（参见第 *350* 页）。

第14.7节
运动功能块

概述

本节介绍**运动**功能块。

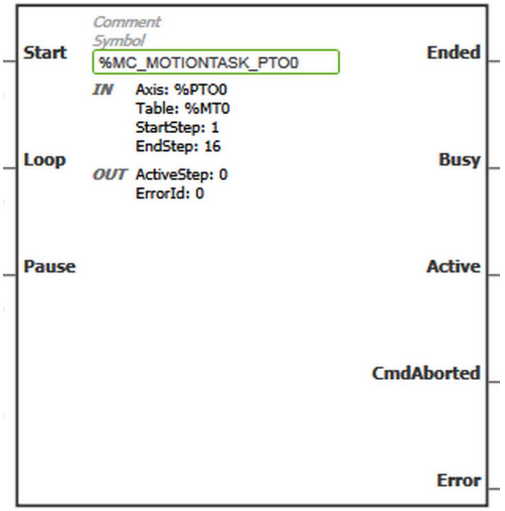
本节包含了哪些内容？

本节包含了以下主题：

主题	页
MC_MotionTask_PTO 功能块	357
MC_Power_PTO 功能块	361
MC_MoveVel_PTO 功能块	364
MC_MoveRel_PTO 功能块	368
MC_MoveAbs_PTO 功能块	372
MC_Home_PTO 功能块	376
MC_SetPos_PTO 功能块	379
MC_Stop_PTO 功能块	381
MC_Halt_PTO 功能块	384

MC_MotionTask_PTO 功能块

图形表示形式



注： 在首次进入功能块时，必须将其配置为使用预期轴和 motion task table。在功能块上双击以显示功能块属性，选择相关轴和表，然后单击 Apply。

输入

下表描述功能块的输入：

输入	初始值	描述
Start	FALSE	在上升沿启动功能块的执行。 Loop 和 Pause 输入在功能块执行期间是可以更改的， 不过这会影响到当前的执行。 Axis、Table、 StartStep 和 EndStep 输入对象值定义当发生上升沿时的动作顺序。对这些输入对象的后续更改不会影响到当前的执行。 这些输出在功能块执行结束时置位。 当下列情况为 FALSE 时： ● 当执行正在进行（移动操作Busy和Active）时，输出刷新。 ● 当执行终止时，稍后输出复位一个周期。
Loop	FALSE	当情况为 TRUE 时，一旦功能块执行终止并且没有检测到错误，运动任务顺序再次从 StartStep 开始。Ended 输出置位一个循环。 当功能块执行终止并且没有检测到错误时测试输入（Ended 输出为正）。

输入	初始值	描述
Pause	FALSE	当结果为 TRUE 时： ● Active = 1 和 Busy = 1 ● 强制轴进入 **Halt** 状态。 为了达到 **Halt** 状态，轴在 **Discrete motion** 状态中减速，然后当速度 = 0 时转入 **Standstill** 状态。 ● 只要 Pause 输入为 TRUE，便会保持 **Halt** 状态。 ● 即使速度等于 0，则保持 Active 输出置位。 当设置为 TRUE 后再复位为 FALSE 时，运动任务在以下情况下恢复执行： ● 无论当前速度值的大小，运动任务恢复执行。 ● 使用当前活动的步骤参数。 ● 绝对目标位置未变。如果运动任务是相对移动类型，则无需增加距离。 ● 在此步骤中，**下一步骤**条件复位（例如：延迟从 0 重新开始，启用 Probe input event 并等待配置的脉冲沿）。

下表描述功能块的输入对象：

输入对象	类型	初始值	描述
Axis	%PTOx	–	要执行的功能块的 PTO 轴实例。此参数在**编程** → **工具**模块选项卡中达到的功能块实例中设置。在 **PTO 对象** → **运动** → **MC_MotionTask_PTO** → **MC_MotionTask_PTO_properties** 对话框中选择**轴**参数。
Table	%MT	–	要执行的功能块的表实例。此参数在**编程** → **工具**模块选项卡中达到的功能块实例中设置。在 **PTO 对象** → **运动** → **MC_MotionTask_PTO** → **MC_MotionTask_PTO_properties** 对话框中选择**表**参数。
StartStep	Byte	1	定义在 Motion Task Table 执行的第一步的步骤编号。 执行顺序是从 StartStep 到 EndStep。 限制: StartStep ≤ EndStep.
EndStep	Byte	16	定义在 Motion Task Table 执行的最后一步的步骤编号。 执行顺序是从 StartStep 到 EndStep。 限制: StartStep ≤ EndStep. **注意：** 如果 EndStep 大于 Motion Task Table 中定义的最大步数，则使用表中的最后一步。

输出

下表描述功能块的输出：

输出	初始值	描述
Ended	0	值为 TRUE 时，表示功能块执行完成，且未检测到任何错误。 Ended 输出行为： ● 如果运动序列的最后一步是**离散**移动，则输出的行为类似于 Done 输出，其他输出 (Busy, Active, CmdAborted, Error) 复位为 0。 ● 如果动作顺序的最后一步是**持续**运动（移动速度），则输出的行为类似于 InVel 输出。 其他输出行为： ❍ Busy 和 Active 为 TRUE (1)。 ❍ CmdAborted 和 Error 为 FALSE (0)。 如果请求循环（Loop 输入），Ended 输出在一个任务周期内为 TRUE。
Busy	-	值为 TRUE 时，表示正在执行功能块。 值为 FALSE 时，表示功能块的执行终止。
Active	-	值为 TRUE 时，功能块实例控制着轴。对于同一轴，一次只能有一个功能块将 Active 设置为 TRUE。
CmdAborted	-	当为 TRUE 时，功能块因其他运动命令 (MC_Stop_PTO) 或检测到轴错误时终止执行。
Error	FALSE	如果为 TRUE，表示检测到错误。功能块执行结束。

下表描述功能块的输出对象：

输出对象	类型	初始值	描述
ActiveStep	Byte	0	步数已在 Motion Task Table 中执行。
ErrorId	Word	NoError	运动命令错误代码，在 Error 输出为 TRUE 时有效。请参阅 PTO 运动命令错误代码表 (参见第 *350* 页)。

操作模式

MC_MotionTask_PTO 启动：功能块只能从 **Standstill** 状态启动。

MC_MotionTask_PTO 停止：功能块可被以下其中一种操作停止：

- 将 Pause 输入设置为 TRUE。
- 执行 MC_Stop_PTO

动作任务中的步骤执行遵守与每个单轴功能块相同的规则和限制。在检测到错误时，功能块通常运行如下：

- 如果在功能块执行期间检测到动作状态或范围错误：
  - 使用当前步骤加速度参数值将动作停止命令应用于动作任务。如果步骤加速度参数无效，则应用快速停止加速度。
  - 在受控动作停止期间，功能块输出 Active，Busy 保持为 TRUE，输出对象 ActiveStep = 0。
  - 动作一旦被停止，功能块执行结束时 Error = 1，并且 ErrorId 输出对象设置为对应用被检出错误类型的值。
- 如果检测到轴控制错误，轴切换到 **Stopping** 状态。功能块执行结束时 Error = 1，并且 ErrorId 输出对象设置为对应用被检出错误类型的值。

MC_Power_PTO 功能块

行为

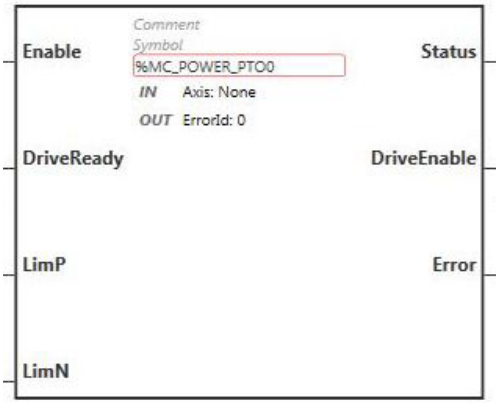
在以下情况下，禁用轴：

- %MC_Power_PTO.Enable = FALSE，或
- %MC_Power_PTO.DriveReady = FALSE，或
- 检测到硬件限位错误 (HwPositionLimitP / HwPositionLimitN)

如果禁用轴，则：

- 轴的状态从 Standstill 切换为 Disabled，或
  在任何当前移动中，状态先切换为 ErrorStop，然后切换为 Disabled（在复位错误时）。
- %MC_ReadSts_PTO.IsHomed 复位为 0（需要新的原点搜索过程）。

图形表示形式



注： 在首次进入功能块时，必须将其配置为使用预期轴。在功能块上双击以显示功能块属性，选择相关轴，然后单击 Apply。

输入

下表描述功能块的输入：

输入	初始值	描述
Enable	FALSE	当值为 TRUE 时，执行功能块。可连续修改其他功能块输入的值，并且连续更新功能块输出。 当值为 FALSE 时，终止功能块执行并复位其输出。
DriveReady	FALSE	来自驱动器的信号，指示其准备就绪。 当驱动器准备好开始执行运动时，设置为 TRUE。 如果已将驱动器信号连接到控制器，请使用相应的控制器输入。如果驱动器未提供此信号，则可以使用任何 TRUE 布尔值强制此输入设置值 TRUE。

输入	初始值	描述
LimP	TRUE	正向的硬件限位开关信息。 达到硬件限位开关时，设置为 FALSE。 如果已将硬件限位开关信号连接到控制器，请使用相应的控制器输入。如果此信号无法使用，则可以使用任何 TRUE 布尔值强制此输入设置值 TRUE。
LimN	TRUE	反向的硬件限位开关信息。 达到硬件限位开关时，设置为 FALSE。 如果已将硬件限位开关信号连接到控制器，请使用相应的控制器输入。如果此信号无法使用，则可以使用任何 TRUE 布尔值强制此输入设置值 TRUE。

下表描述功能块的输入对象：

输入对象	类型	初始值	描述
Axis	PTOx	-	将执行功能块的实例。在控制器配置中声明名称。

输出

下表描述功能块的输出：

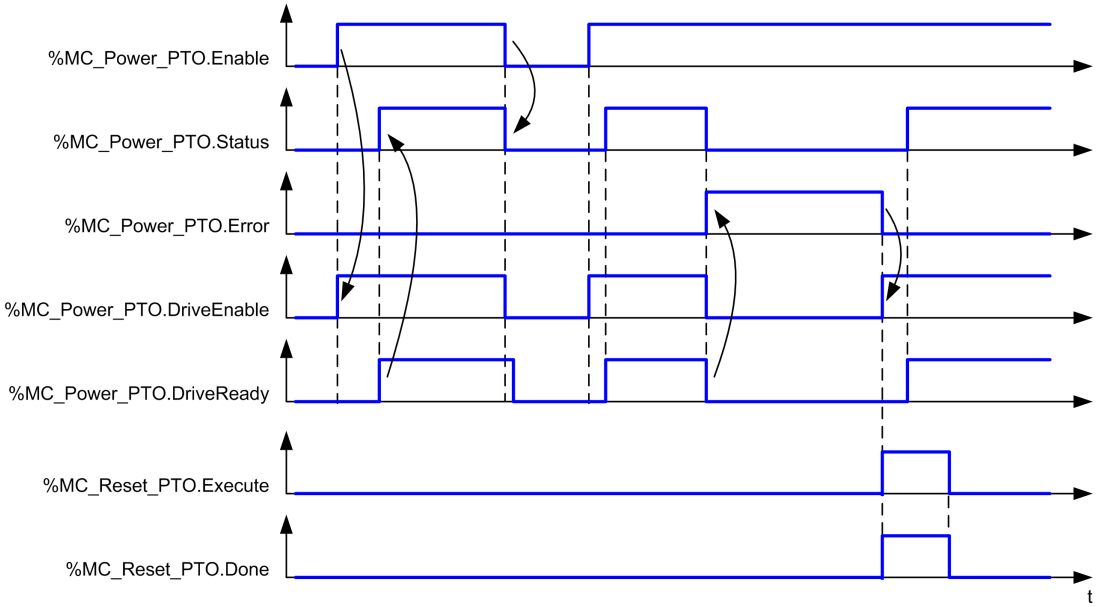
输出	初始值	描述
Status	FALSE	如果值为 TRUE，在准备好接受运动命令时对驱动器进行报告。
DriveEnable	FALSE	值为 TRUE 时，指示驱动器可以接受运动命令，并且应启用电源。 如果已将驱动器输入连接到控制器，请使用相应的控制器输出。如果驱动器没有此信号的输入，则可以保留此功能块输出不用。
Error	FALSE	如果为 TRUE，表示检测到错误。功能块执行结束。

下表描述功能块的输出对象：

输出对象	类型	初始值	描述
ErrorId	Word	NoError	运动命令错误代码，在 Error 输出为 TRUE 时有效。请参阅 PTO 运动命令错误代码表 (参见第 *350* 页)。

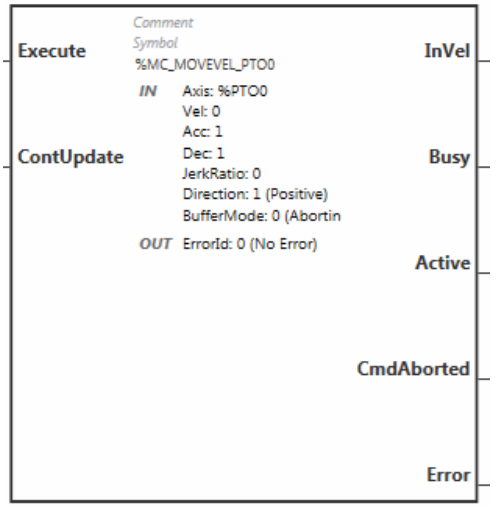
时序图示例

下图说明 MC_Power_PTO 功能块的操作：



MC_MoveVel_PTO 功能块

图形表示形式



注： 在首次进入功能块时，必须将其配置为使用预期轴。在功能块上双击以显示功能块属性，选择相关轴，然后单击 Apply。

输入

下表描述功能块的输入：

输入	初始值	描述
Execute	FALSE	在上升沿启动功能块的执行。其他功能块输入的值可在 Execute 上升沿控制此功能块的执行。这些输入参数的后续更改不会影响当前执行，除非 ContUpdate 输入为 TRUE。 这些输出在功能块终止时设置。 如果在执行功能块期间检测到第二个上升沿，将中止当前执行并再次执行该功能块。
ContUpdate	FALSE	值为 TRUE 时，将使功能块使用输入对象（Vel、Acc、Dec 和 Direction）的所有修改值，并将其应用于当前命令。 在要考虑的 Execute 输入出现上升沿之前，此输入必须为 TRUE。 **注意：** 不考虑对 Axis 参数值的修改。必须将 Execute 设置为 0，然后再设置为 1，从而更改 Axis。

下表描述功能块的输入对象：

输入对象	类型	初始值	描述
Axis	PTOx	-	将执行功能块的实例。在控制器配置中声明名称。
Vel	DINT	0	目标速度。 范围（赫兹）：0...MaxVelocityAppl（参见第 *348* 页）
Acc	DINT	0	加速度（赫兹/毫秒） 范围（赫兹/毫秒）：1...MaxAccelerationAppl （参见第 *348* 页）
Dec	DINT	0	减速度（赫兹/毫秒） 范围（赫兹/毫秒）：1...MaxDecelerationAppl （参见第 *348* 页）
JerkRatio	INT	0	用于创建 S 曲线轮廓 (参见第 *313* 页)的加速/减速调整百分比。 范围：0...100
Direction	INT	mcPositiveDirection	PTO 类型 CW/CCW 的运动方向 正向 (CW) = 1 (mcPositiveDirection) 反向 (CCW) = -1 (mcNegativeDirection)
BufferMode	INT	mcAborting	正在进行的移动的转换模式。请参阅缓冲模式表 (参见第 *347* 页)。

输出

下表描述功能块的输出：

输出	初始值	描述
InVel	FALSE	值为 TRUE 时，已达到目标速度。
Busy	-	值为 TRUE 时，表示正在执行功能块。 值为 FALSE 时，表示功能块的执行终止。 至少在 Busy 为 TRUE 时，功能块必须保留在应用程序的活动任务中。
Active	-	值为 TRUE 时，功能块实例控制着轴。对于同一轴，一次只能有一个功能块将 Active 设置为 TRUE。
CmdAborted	-	值为 TRUE 时，功能块因其他运动命令终止执行。
Error	FALSE	如果为 TRUE，表示检测到错误。功能块执行结束。

下表描述功能块的输出对象：

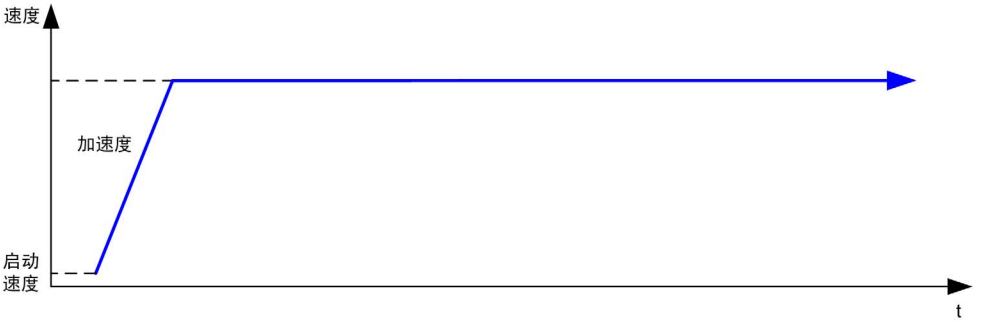
输出对象	类型	初始值	描述
ErrorId	Word	NoError	运动命令错误代码，在 Error 输出为 TRUE 时有效。请参阅 PTO 运动命令错误代码表 (参见第 *350* 页)。

注意：

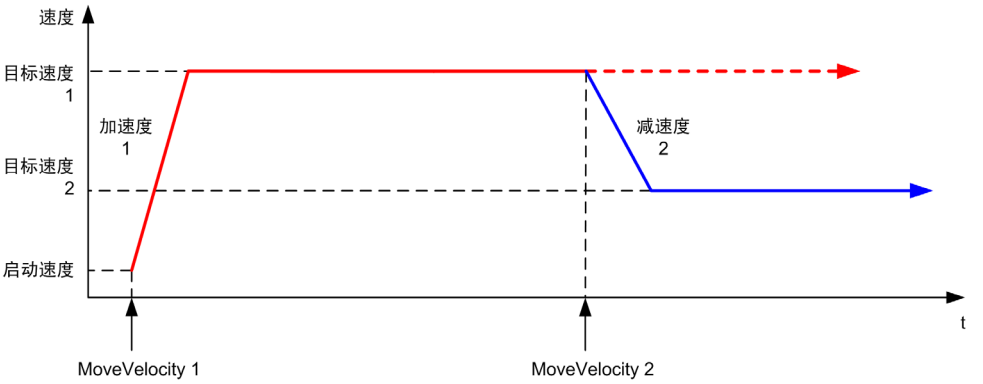
- 要停止运动，必须通过另一个功能块发出新命令来中断功能块。
- 如果运动正在进行，并且方向为反向，则首先使用 MC_MoveVel_PTO 功能块的减速度来暂停运动，然后朝反向继续运动。
- 加速/减速的持续时间不得超过 80 秒。

时序图示例

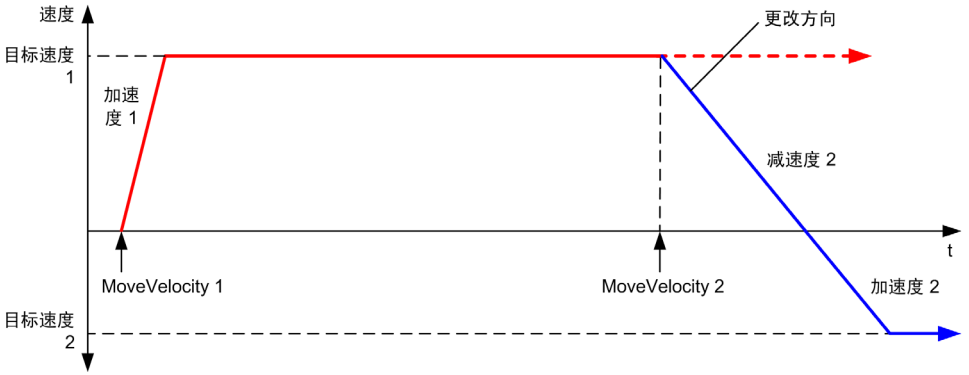
下图显示了 **Standstill** 状态的简单轮廓：



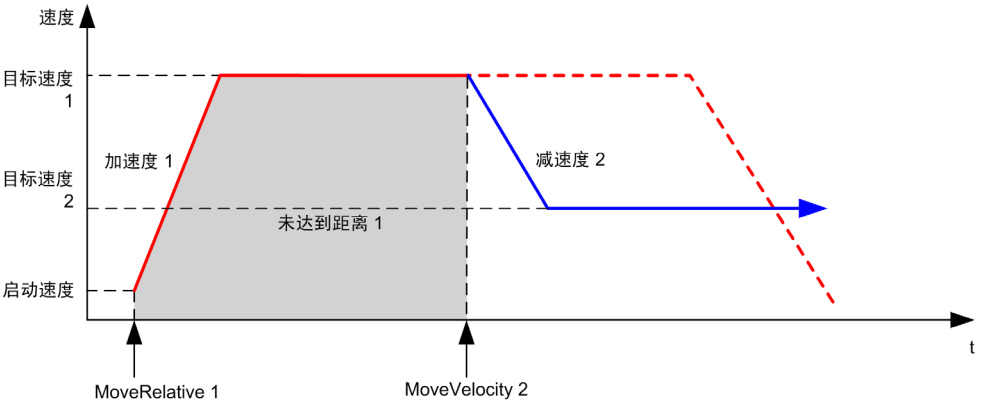
下图显示了 **Continuous** 状态的复杂轮廓：



下图显示了 Continuous 状态的复杂轮廓，有方向变化：

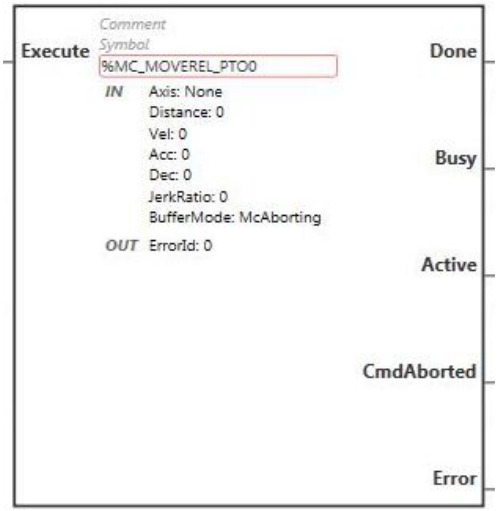


下图显示了 Discrete 状态的复杂轮廓：



MC_MoveRel_PTO 功能块

图形表示形式



注： 在首次进入功能块时，必须将其配置为使用预期轴。在功能块上双击以显示功能块属性，选择相关轴，然后单击 Apply。

输入

下表描述功能块的输入：

输入	初始值	描述
Execute	FALSE	在上升沿启动功能块的执行。其他功能块输入的值可在 Execute 上升沿控制此功能块的执行。后面更改这些输入参数不会影响到正在进行中的执行。 这些输出在功能块终止时设置。

下表描述功能块的输入对象：

输入对象	类型	初始值	描述
Axis	PTOx	-	将执行功能块的实例。在控制器配置中声明名称。
Distance	DINT	0	以脉冲数表示的相对运动距离。符号指定方向。
Vel	DINT	0	目标速度。 范围（赫兹）：0...MaxVelocityAppl（参见第 *348* 页）
Acc	DINT	0	加速度（赫兹/毫秒） 范围（赫兹/毫秒）：1...MaxAccelerationAppl （参见第 *348* 页）

输入对象	类型	初始值	描述
Dec	DINT	0	减速度（赫兹/毫秒） 范围（赫兹/毫秒）：1...MaxDecelerationAppl (参见第 *348* 页)
JerkRatio	INT	0	用于创建 S 曲线轮廓 (参见第 *313* 页)的加速/减速调整百分比。 范围：0...100
BufferMode	INT	mcAborting	正在进行的移动的转换模式。请参阅缓冲模式表 (参见第 *347* 页)。

输出

下表描述功能块的输出：

输出	初始值	描述
Done	FALSE	值为 TRUE 时，表示功能块执行完成，且未检测到任何错误。 如果在完成命令动作之前使用某轴上的一种运动中断同轴上的另一种运动，则 CmdAborted 设置为 TRUE 且 Done 设置为 FALSE。
Busy	-	值为 TRUE 时，表示正在执行功能块。 值为 FALSE 时，表示功能块的执行终止。 至少在 Busy 为 TRUE 时，功能块必须保留在应用程序的活动任务中。
Active	-	值为 TRUE 时，功能块实例控制着轴。对于同一轴，一次只能有一个功能块将 Active 设置为 TRUE。
CmdAborted	-	值为 TRUE 时，功能块因其他运动命令终止执行。
Error	FALSE	如果为 TRUE，表示检测到错误。功能块执行结束。

下表描述功能块的输出对象：

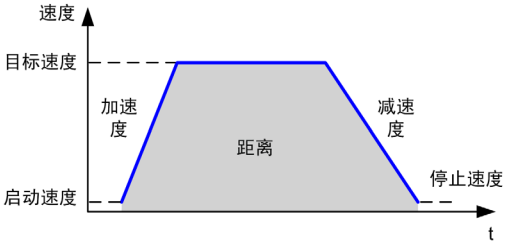
输出对象	类型	初始值	描述
ErrorId	Word	NoError	运动命令错误代码，在 Error 输出为 TRUE 时有效。请参阅 PTO 运动命令错误代码表 (参见第 *350* 页)。

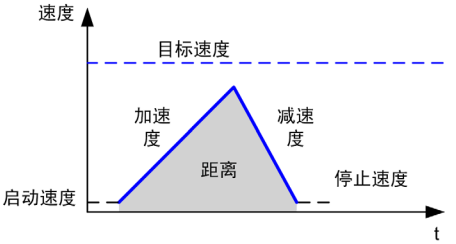
注意：

- 如果没有其他功能块处于未完成状态，功能块将以速度为零完成。
- 如果距离太短，无法达到目标速度，运动轮廓将呈三角形，而不是梯形。
- 如果运动正在进行，并且由于当前的运动参数而超出了命令的距离，则自动管理方向反转：首先使用 MC_MoveRel_PTO 功能块的减速度来暂停运动，然后朝反向继续运动。
- 加速/减速的持续时间不得超过 80 秒。

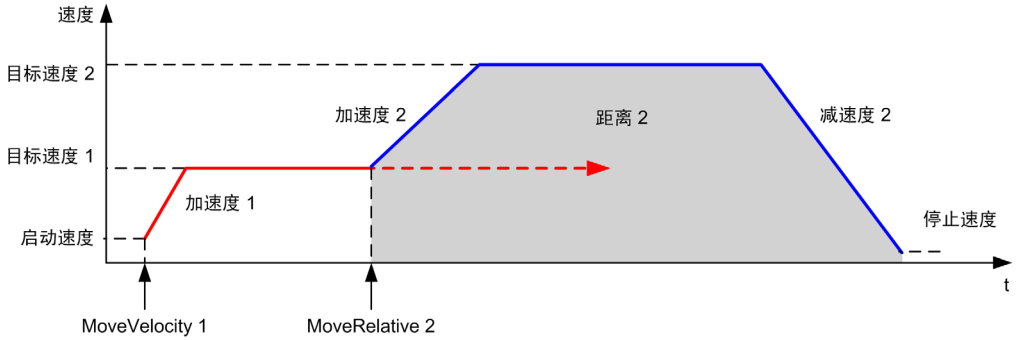
时序图示例

下图显示了 **Standstill** 状态的简单轮廓：

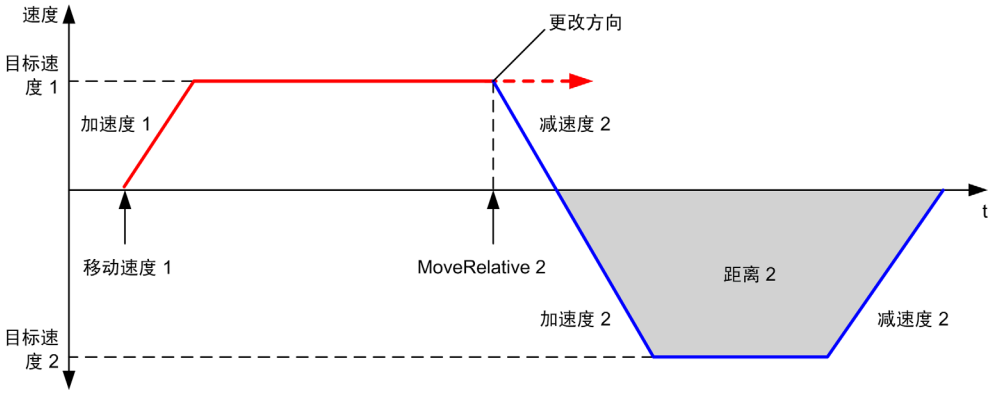




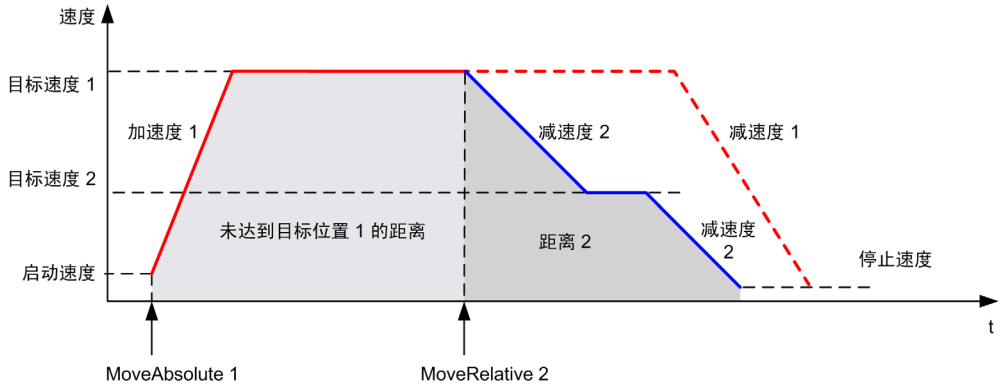
下图显示了 **Continuous** 状态的复杂轮廓：



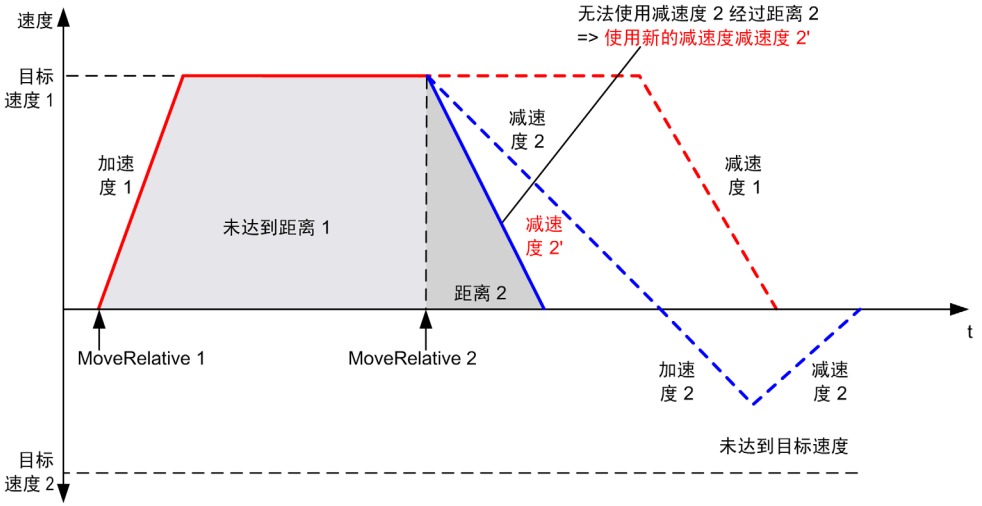
下图显示了 **Continuous** 状态的复杂轮廓，有方向变化：



下图显示了 Discrete 状态的复杂轮廓：

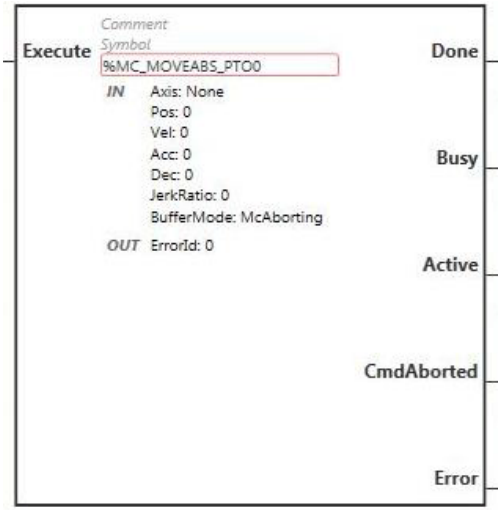


下图显示了 Discrete 状态的复杂轮廓，有方向变化：



MC_MoveAbs_PTO 功能块

图形表示形式



注： 在首次进入功能块时，必须将其配置为使用预期轴。在功能块上双击以显示功能块属性，选择相关轴，然后单击 Apply。

输入

下表描述功能块的输入：

输入	初始值	描述
Execute	FALSE	在上升沿启动功能块的执行。其他功能块输入的值可在 Execute 上升沿控制此功能块的执行。后面更改这些输入参数不会影响到正在进行中的执行。 这些输出在功能块终止时设置。

下表描述功能块的输入对象：

输入对象	类型	初始值	描述
Axis	PTOx	-	将执行功能块的实例。在控制器配置中声明名称。
Pos	DINT	0	轴的位置。
Vel	DINT	0	目标速度。 范围（赫兹）：0...MaxVelocityAppl（参见第 *348* 页）
Acc	DINT	0	加速度（赫兹/毫秒） 范围（赫兹/毫秒）：1...MaxAccelerationAppl （参见第 *348* 页）

输入对象	类型	初始值	描述
Dec	DINT	0	减速度（赫兹/毫秒） 范围（赫兹/毫秒）：1...MaxDecelerationAppl (参见第 *348* 页)
JerkRatio	INT	0	用于创建 S 曲线轮廓 (参见第 *313* 页) 的加速/减速调整百分比。 范围：0...100
BufferMode	INT	mcAborting	正在进行的移动的转换模式。 请参阅缓冲模式表 (参见第 *347* 页)。

输出

下表描述功能块的输出：

输出	初始值	描述
Done	FALSE	值为 TRUE 时，表示功能块执行完成，且未检测到任何错误。 如果在完成命令动作之前使用某轴上的一种运动中断同轴上的另一种运动， 则 CmdAborted 设置为 TRUE 且 Done 设置为 FALSE。
Busy	-	值为 TRUE 时，表示正在执行功能块。 值为 FALSE 时，表示功能块的执行终止。 至少在 Busy 为 TRUE 时，功能块必须保留在应用程序的活动任务中。
Active	-	值为 TRUE 时，功能块实例控制着轴。对于同一轴，一次只能有一个功能块将 Active 设置为 TRUE。
CmdAborted	-	值为 TRUE 时，功能块因其他运动命令终止执行。
Error	FALSE	如果为 TRUE，表示检测到错误。功能块执行结束。

下表描述功能块的输出对象：

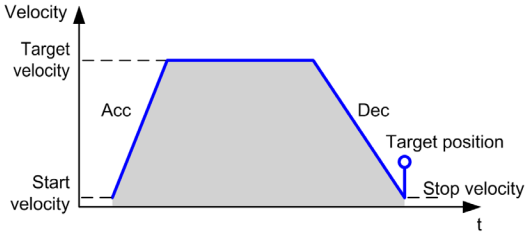
输出对象	类型	初始值	描述
ErrorId	Word	NoError	运动命令错误代码，在 Error 输出为 TRUE 时有效。请参阅 PTO 运动命令错误代码表 (参见第 *350* 页)。

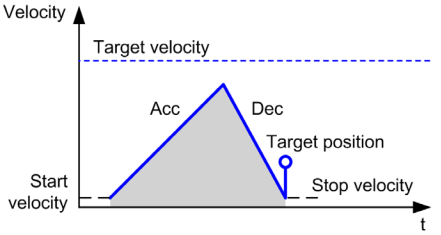
注意：

- 如果没有其他功能块处于未完成状态，功能块将以速度为零完成。
- 根据当前位置和目标位置自动设置运动方向。
- 如果距离太短，无法达到目标速度，运动轮廓将呈三角形，而不是梯形。
- 如果使用当前的方向无法达到该位置，则自动管理方向反转。如果运动正在进行，则首先使用 MC_MoveAbsolute_PTO 功能块的减速度来暂停运动，然后朝反向继续运动。
- 加速/减速的持续时间不得超过 80 秒。

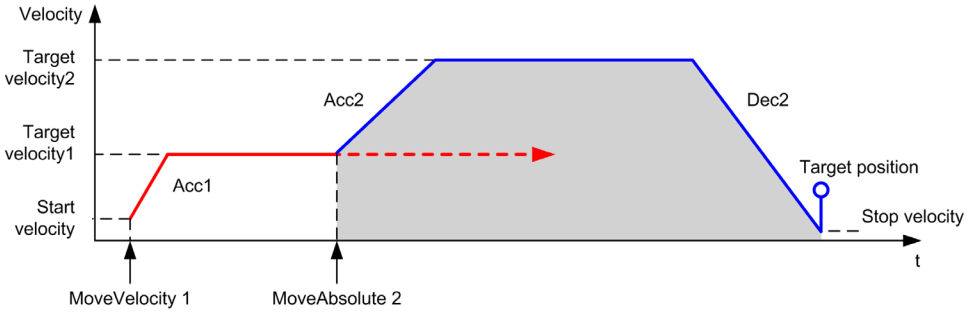
时序图示例

下图显示了 **Standstill** 状态的简单轮廓：

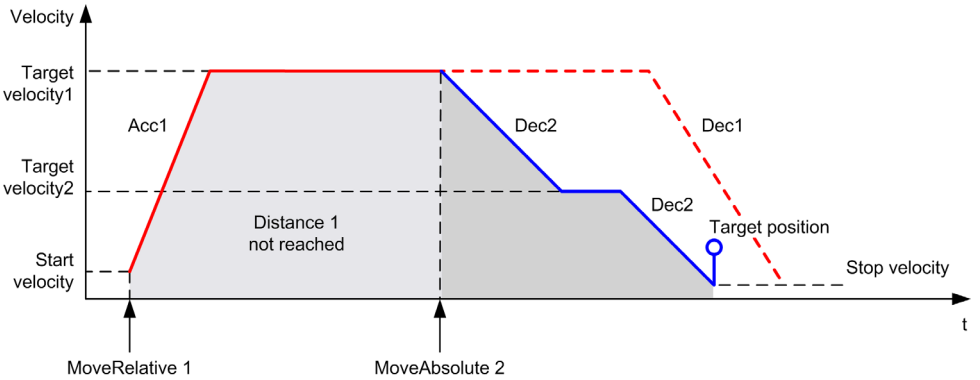




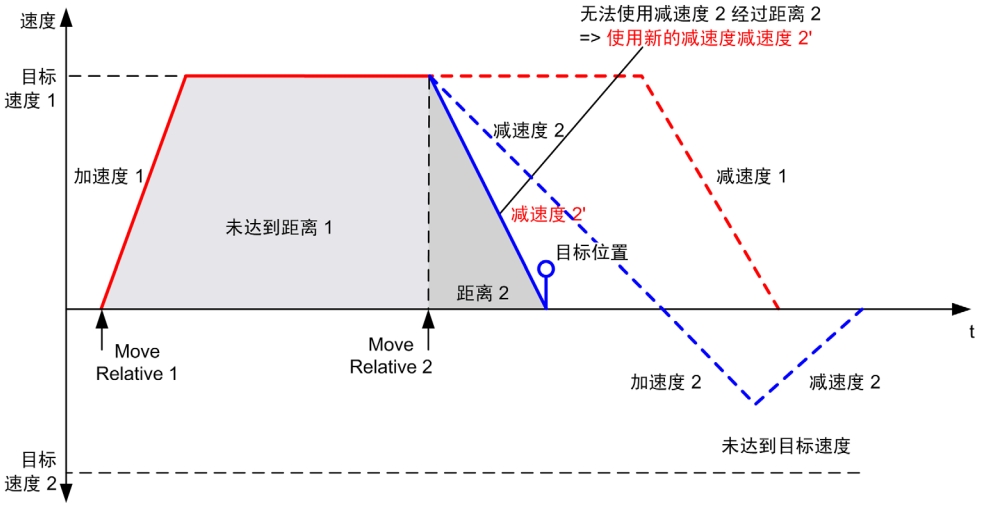
下图显示了 **Continuous** 状态的复杂轮廓：



下图显示了 **Discrete** 状态的复杂轮廓：

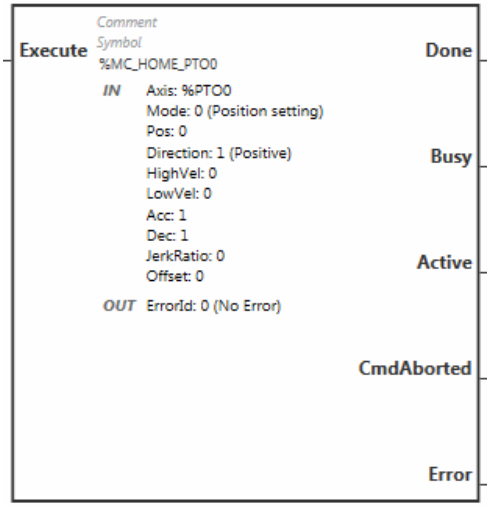


下图显示了 **Discrete** 状态的复杂轮廓，有方向变化：



MC_Home_PTO 功能块

图形表示形式



注： 在首次进入功能块时，必须将其配置为使用预期轴。在功能块上双击以显示功能块属性，选择相关轴，然后单击 Apply。

输入

下表描述功能块的输入：

输入	初始值	描述
Execute	FALSE	在上升沿启动功能块的执行。其他功能块输入的值可在 Execute 上升沿控制此功能块的执行。后面更改这些输入参数不会影响到正在进行中的执行。 这些输出在功能块终止时设置。

下表描述功能块的输入对象：

输入对象	类型	初始值	描述
Axis	PTOx	-	将执行功能块的实例。在控制器配置中声明名称。
Mode	BYTE	0	预定义原点顺序类型。
Pos	DINT	0	轴的位置。
HighVel	DINT	0	用于搜索限制或参考开关的目标原点速度。 范围（赫兹）：1...MaxVelocityAppl（参见第 *348* 页）

输入对象	类型	初始值	描述
LowVel	DINT	0	用于搜索参考开关信号的目标原点速度。在检测到限位开关或参考开关时，停止运动。 范围（赫兹）：1...HighVelocity
Acc	DINT	0	加速度（赫兹/毫秒） 范围（赫兹/毫秒）：1...MaxAccelerationAppl （参见第 *348* 页）
Dec	DINT	0	减速度（赫兹/毫秒） 范围（赫兹/毫秒）：1...MaxDecelerationAppl （参见第 *348* 页）
JerkRatio	INT	0	用于创建 S 曲线轮廓 (参见第 *313* 页) 的加速/减速调整百分比。 范围：0...100
Direction	INT	mcPosi- tiveDi- rection	PTO 类型 CW/CCW 的运动方向 正向 (CW) = 1 (mcPositiveDirection) 反向 (CCW) = -1 (mcNegativeDirection)
Offset	DINT	0	从起点开始的距离。达到起点时，运动将继续移动这段距离。方向取决于符号（原点补偿）。 范围：-2,147,483,648...2,147,483,647

输出

下表描述功能块的输出：

输出	初始值	描述
Done	FALSE	值为 TRUE 时，表示功能块执行完成，且未检测到任何错误。 如果在完成命令动作之前使用某轴上的一种运动中断同轴上的另一种运动，则 CmdAborted 设置为 TRUE 且 Done 设置为 FALSE。
Busy	-	值为 TRUE 时，表示正在执行功能块。 值为 FALSE 时，表示功能块的执行终止。 至少在 Busy 为 TRUE 时，功能块必须保留在应用程序的活动任务中。
Active	-	值为 TRUE 时，功能块实例控制着轴。对于同一轴，一次只能有一个功能块将 Active 设置为 TRUE。
CmdAborted	-	值为 TRUE 时，功能块因其他运动命令终止执行。
Error	FALSE	如果为 TRUE，表示检测到错误。功能块执行结束。

下表描述功能块的输出对象：

输出对象	类型	初始值	描述
ErrorId	Word	NoError	运动命令错误代码，在 Error 输出为 TRUE 时有效。请参阅 PTO 运动命令错误代码表 (参见第 *350* 页)。

注意： 加速/减速的持续时间不得超过 80 秒。

时序图示例

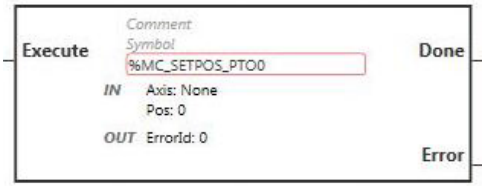
原点模式 (参见第 *336* 页)

MC_SetPos_PTO 功能块

行为

此功能块可修改轴的实际位置坐标，而不需要任何物理移动。仅当轴处于 Standstill 状态时，才能使用此功能块。

图形表示形式



注： 在首次进入功能块时，必须将其配置为使用预期轴。在功能块上双击以显示功能块属性，选择相关轴，然后单击 Apply。

输入

下表描述功能块的输入：

输入	初始值	描述
Execute	FALSE	在上升沿启动功能块的执行。其他功能块输入的值可在 Execute 上升沿控制此功能块的执行。后面更改这些输入参数不会影响到正在进行中的执行。 这些输出在功能块终止时设置。

下表描述功能块的输入对象：

输入对象	类型	初始值	描述
Axis	PTOx	-	将执行功能块的实例。在控制器配置中声明名称。
Pos	DINT	0	轴的位置。

输出

下表描述功能块的输出：

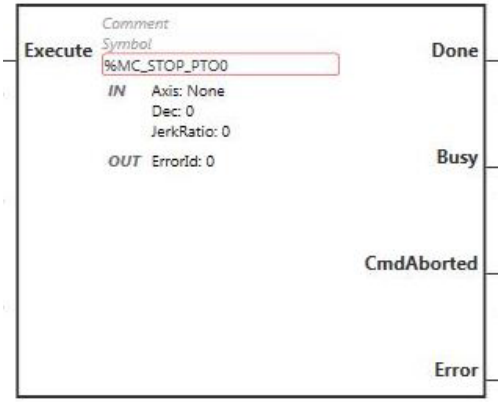
输出	初始值	描述
Done	FALSE	值为 TRUE 时，表示功能块执行完成，且未检测到任何错误。
Error	FALSE	如果为 TRUE，表示检测到错误。功能块执行结束。

下表描述功能块的输出对象：

输出对象	类型	初始值	描述
ErrorId	Word	NoError	运动命令错误代码，在 Error 输出为 TRUE 时有效。请参阅 PTO 运动命令错误代码表 (参见第 *350* 页)。

MC_Stop_PTO 功能块

图形表示形式



注： 在首次进入功能块时，必须将其配置为使用预期轴。在功能块上双击以显示功能块属性，选择相关轴，然后单击 Apply。

输入

下表描述功能块的输入：

输入	初始值	描述
Execute	FALSE	在上升沿启动功能块的执行。其他功能块输入的值可在 Execute 上升沿控制此功能块的执行。后面更改这些输入参数不会影响到正在进行中的执行。 这些输出在功能块终止时设置。

下表描述功能块的输入对象：

输入对象	类型	初始值	描述
Axis	PTOx	-	将执行功能块的实例。在控制器配置中声明名称。
Dec	DINT	0	减速度（赫兹/毫秒） 范围（赫兹/毫秒）：1...MaxDecelerationAppl (参见第 *348* 页)
JerkRatio	INT	0	用于创建 S 曲线轮廓 (参见第 *313* 页)的加速/减速调整百分比。 范围：0...100

输出

下表描述功能块的输出：

输出	初始值	描述
Done	FALSE	值为 TRUE 时，表示功能块执行完成，且未检测到任何错误。 如果在完成命令动作之前使用某轴上的一种运动中断同轴上的另一种运动，则 CmdAborted 设置为 TRUE 且 Done 设置为 FALSE。
Busy	-	值为 TRUE 时，表示正在执行功能块。 值为 FALSE 时，表示功能块的执行终止。 至少在 Busy 为 TRUE 时，功能块必须保留在应用程序的活动任务中。
CmdAborted	-	值为 TRUE 时，功能块因其他运动命令终止执行。
Error	FALSE	如果为 TRUE，表示检测到错误。功能块执行结束。

下表描述功能块的输出对象：

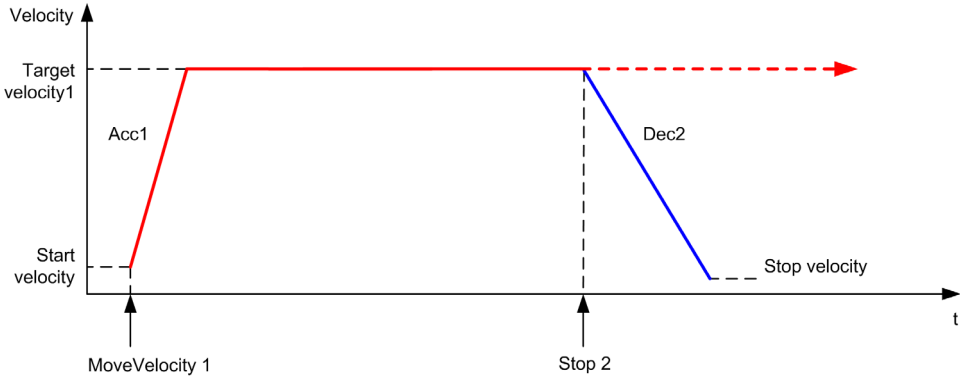
输出对象	类型	初始值	描述
ErrorId	Word	NoError	运动命令错误代码，在 Error 输出为 TRUE 时有效。 请参阅 PTO 运动命令错误代码表 (参见第 *350* 页)。

注意：

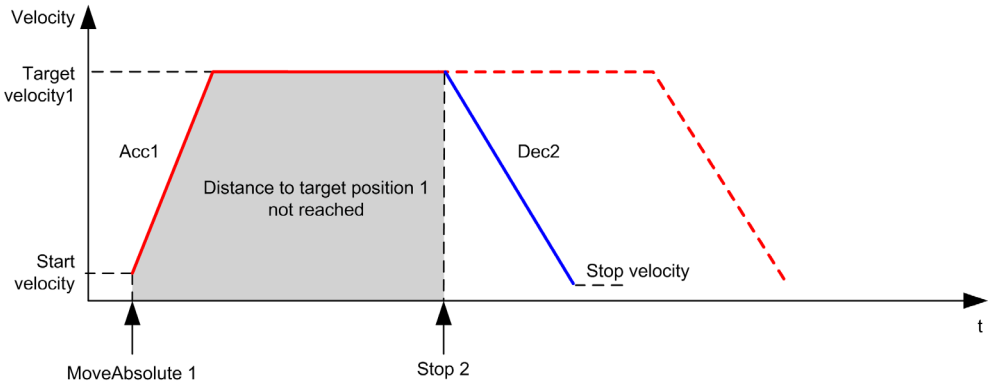
- 在状态 **Standstill** 下调用此功能块会将状态更改为 **Stopping**，当 Execute 为 FALSE 时恢复为 **Standstill**。
- 只要输入 **为 TRUE，便会保持 Stopping**Execute 状态。
- 当停止斜坡完成时设置 Done 输出。
- 如果 Deceleration = 0，则使用快速停止减速度。
- 功能块完成时速度为零。
- 减速持续时间不得超过 80 秒。

时序图示例

下图显示了 **Continuous** 状态的简单轮廓：



下图显示了 **Discrete** 状态的简单轮廓：



MC_Halt_PTO 功能块

图形表示形式

注： 在首次进入功能块时，必须将其配置为使用预期轴。在功能块上双击以显示功能块属性，选择相关轴，然后单击 Apply。

输入

下表描述功能块的输入：

输入	初始值	描述
Execute	FALSE	在上升沿启动功能块的执行。其他功能块输入的值可在 Execute 上升沿控制此功能块的执行。后面更改这些输入参数不会影响到正在进行中的执行。 这些输出在功能块终止时设置。

下表描述功能块的输入对象：

输入对象	类型	初始值	描述
Axis	PTOx	-	将执行功能块的实例。在控制器配置中声明名称。
Dec	DINT	0	减速度（赫兹/毫秒） 范围（赫兹/毫秒）：1...MaxDecelerationAppl （参见第 *348* 页）
JerkRatio	INT	0	用于创建 S 曲线轮廓 (参见第 *313* 页) 的加速/减速调整百分比。 范围：0...100

输入对象	类型	初始值	描述
BufferMode	INT	mcAborting	正在进行的移动的转换模式。请参阅缓冲模式表 (参见第 *347* 页)。

输出

下表描述功能块的输出：

输出	初始值	描述
Done	FALSE	值为 TRUE 时，表示功能块执行完成，且未检测到任何错误。 如果在完成命令动作之前使用某轴上的一种运动中断同轴上的另一种运动，则 CmdAborted 设置为 TRUE 且 Done 设置为 FALSE。
Busy	-	值为 TRUE 时，表示正在执行功能块。 值为 FALSE 时，表示功能块的执行终止。 至少在 Busy 为 TRUE 时，功能块必须保留在应用程序的活动任务中。
Active	-	值为 TRUE 时，功能块实例控制着轴。对于同一轴，一次只能有一个功能块将 Active 设置为 TRUE。
CmdAborted	-	值为 TRUE 时，功能块因其他运动命令终止执行。
Error	FALSE	如果为 TRUE，表示检测到错误。功能块执行结束。

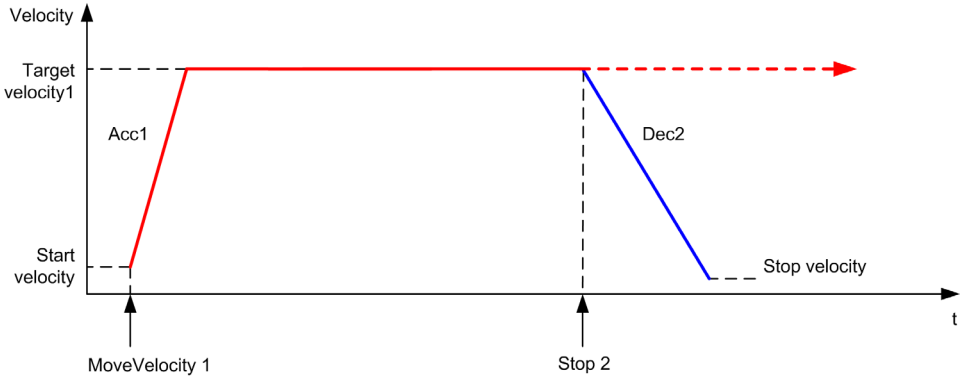
下表描述功能块的输出对象：

输出对象	类型	初始值	描述
ErrorId	Word	NoError	运动命令错误代码，在 Error 输出为 TRUE 时有效。请参阅 PTO 运动命令错误代码表 (参见第 *350* 页)。

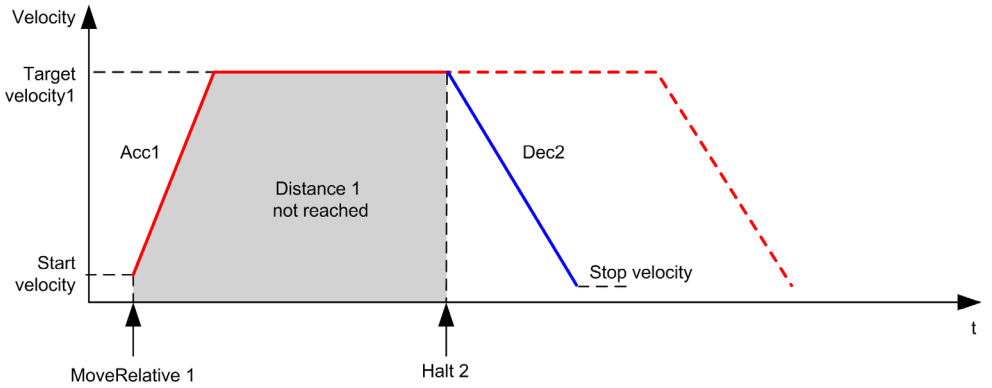
注意： 功能块完成时速度为零。

时序图示例

下图显示了 **Continuous** 状态的简单轮廓：



下图显示了 **Discrete** 状态的简单轮廓：



第14.8节
管理功能块

概述

本节介绍**管理**功能块。

本节包含了哪些内容？

本节包含了以下主题：

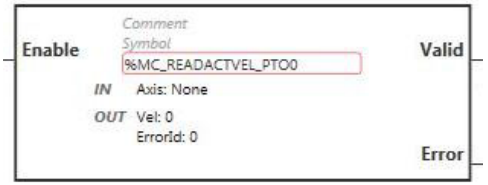
主题	页
MC_ReadActVel_PTO 功能块	388
MC_ReadActPos_PTO 功能块	390
MC_ReadSts_PTO 功能块	392
MC_ReadMotionState_PTO 功能块	394
MC_ReadAxisError_PTO 功能块	396
MC_Reset_PTO 功能块	398
MC_TouchProbe_PTO 功能块	400
MC_AbortTrigger_PTO 功能块	402
MC_ReadPar_PTO 功能块	404
MC_WritePar_PTO 功能块	406

MC_ReadActVel_PTO 功能块

功能描述

此功能块返回轴的实际速度值。

图形表示形式



注： 在首次进入功能块时，必须将其配置为使用预期轴。在功能块上双击以显示功能块属性，选择相关轴，然后单击 Apply。

输入

下表描述功能块的输入：

输入	初始值	描述
Enable	FALSE	当值为 TRUE 时，执行功能块。可连续修改其他功能块输入的值，并且连续更新功能块输出。 当值为 FALSE 时，终止功能块执行并复位其输出。

下表描述功能块的输入对象：

输入对象	类型	初始值	描述
Axis	PTOx	-	将执行功能块的实例。在控制器配置中声明名称。

输出

下表描述功能块的输出：

输出	初始值	描述
Valid	-	如果值为 TRUE，则功能块对象数据有效。
Error	FALSE	如果为 TRUE，表示检测到错误。功能块执行结束。

下表描述功能块的输出对象：

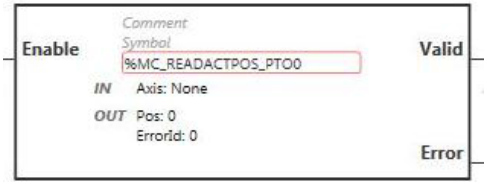
输出对象	类型	初始值	描述
Vel	DINT	-	轴的速度。
ErrorId	Word	NoError	运动命令错误代码，在 Error 输出为 TRUE 时有效。请参阅 PTO 运动命令错误代码表 (参见第 *350* 页)。

MC_ReadActPos_PTO 功能块

功能描述

此功能块返回轴的实际位置值。

图形表示形式



注： 在首次进入功能块时，必须将其配置为使用预期轴。在功能块上双击以显示功能块属性，选择相关轴，然后单击 Apply。

输入

下表描述功能块的输入：

输入	初始值	描述
Enable	FALSE	当值为 TRUE 时，执行功能块。可连续修改其他功能块输入的值，并且连续更新功能块输出。 当值为 FALSE 时，终止功能块执行并复位其输出。

下表描述功能块的输入对象：

输入对象	类型	初始值	描述
Axis	PTOx	-	将执行功能块的实例。在控制器配置中声明名称。

输出

下表描述功能块的输出：

输出	初始值	描述
Valid	-	如果值为 TRUE，则功能块对象数据有效。
Error	FALSE	如果为 TRUE，表示检测到错误。功能块执行结束。

下表描述功能块的输出对象：

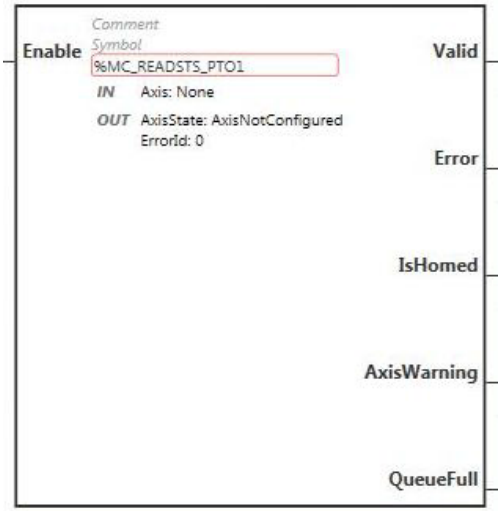
输出对象	类型	初始值	描述
Pos	DINT	-	轴的位置。
ErrorId	Word	NoError	运动命令错误代码，在 Error 输出为 TRUE 时有效。请参阅 PTO 运动命令错误代码表 (参见第 *350* 页)。

MC_ReadSts_PTO 功能块

功能描述

此功能块返回轴的状态图 (参见第 *352* 页)状态。

图形表示形式



注： 在首次进入功能块时，必须将其配置为使用预期轴。在功能块上双击以显示功能块属性，选择相关轴，然后单击 Apply。

输入

下表描述功能块的输入：

输入	初始值	描述
Enable	FALSE	当值为 TRUE 时，执行功能块。可连续修改其他功能块输入的值，并且连续更新功能块输出。 当值为 FALSE 时，终止功能块执行并复位其输出。

下表描述功能块的输入对象：

输入对象	类型	初始值	描述
Axis	PTOx	-	将执行功能块的实例。在控制器配置中声明名称。

输出

下表描述功能块的输出：

输出	初始值	描述
Valid	-	如果值为 TRUE，则功能块对象数据有效。
Error	FALSE	如果为 TRUE，表示检测到错误。功能块执行结束。
IsHomed	FALSE	值为 TRUE 时，表示轴已原点，因此绝对参考点有效，并且允许使用绝对运动命令。
AxisWarning	FALSE	值为 TRUE 时，表示某运动命令已调用警报或提供建议。使用 MC_ReadAxisError_PTO 功能块可获取详细信息。 (参见第 *396* 页)
QueueFull	FALSE	值为 TRUE 时，表示运动队列已满，不允许使用其他缓冲运动命令。

下表描述功能块的输出对象：

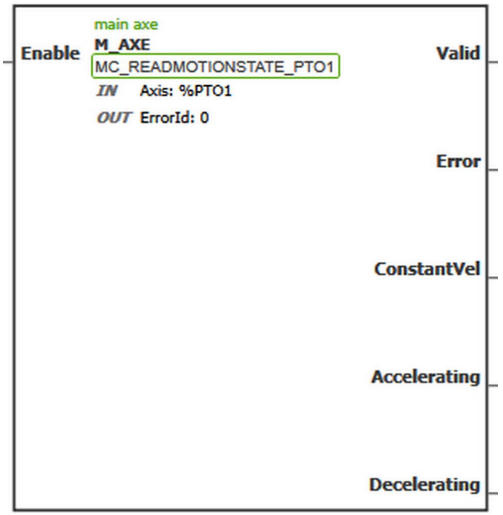
输出对象	类型	初始值	描述
AxisState	-	-	轴的状态代码： 0 = 未配置轴 1 = ErrorStop 2 = Disabled 4 = Stopping 8 = Homing 16 = 停顿 32 = 离散运动 64 = 连续运动 有关详细信息，请参阅状态描述表 (参见第 *352* 页)。
ErrorId	Word	NoError	运动命令错误代码，在 Error 输出为 TRUE 时有效。请参阅 PTO 运动命令错误代码表 (参见第 *350* 页)。

MC_ReadMotionState_PTO 功能块

功能描述

此功能块返回轴的实际运动状态。

图形表示形式



注： 在首次进入功能块时，必须将其配置为使用预期轴。在功能块上双击以显示功能块属性，选择相关轴，然后单击 Apply。

输入

下表描述功能块的输入：

输入	初始值	描述
Enable	FALSE	当值为 TRUE 时，执行功能块。可连续修改其他功能块输入的值，并且连续更新功能块输出。 当值为 FALSE 时，终止功能块执行并复位其输出。

下表描述功能块的输入对象：

输入对象	类型	初始值	描述
Axis	PTOx	-	将执行功能块的实例。在控制器配置中声明名称。

输出

下表描述功能块的输出：

输出	初始值	描述
Valid	-	如果值为 TRUE，则功能块对象数据有效。
Error	FALSE	如果为 TRUE，表示检测到错误。功能块执行结束。
ConstantVel	-	值为 TRUE 时，表示轴的速度是常量。
Accelerating	-	值为 TRUE 时，表示轴的速度在增加。
Decelerating	-	值为 TRUE 时，表示轴的速度在减少。

下表描述功能块的输出对象：

输出对象	类型	初始值	描述
ErrorId	Word	NoError	运动命令错误代码，在 Error 输出为 TRUE 时有效。请参阅 PTO 运动命令错误代码表 (参见第 *350* 页)。

MC_ReadAxisError_PTO 功能块

功能描述

此功能块检索轴控制错误。如果没有任何轴控制错误未解决，则功能块返回 AxisErrorId = 0。

图形表示形式

注： 在首次进入功能块时，必须将其配置为使用预期轴。在功能块上双击以显示功能块属性，选择相关轴，然后单击 Apply。

输入

下表描述功能块的输入：

输入	初始值	描述
Enable	FALSE	当值为 TRUE 时，执行功能块。可连续修改其他功能块输入的值，并且连续更新功能块输出。 当值为 FALSE 时，终止功能块执行并复位其输出。

下表描述功能块的输入对象：

输入对象	类型	初始值	描述
Axis	PTOx	-	将执行功能块的实例。在控制器配置中声明名称。

输出

下表描述功能块的输出：

输出	初始值	描述
Valid	-	如果值为 TRUE，则功能块对象数据有效。
Error	FALSE	如果为 TRUE，表示检测到错误。功能块执行结束。

下表描述功能块的输出对象：

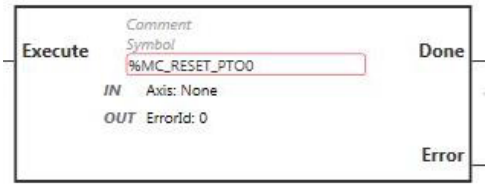
输出对象	类型	初始值	描述
AxisErrorId	-	-	轴错误代码，在 AxisWarning 输出为 TRUE 时有效。请参阅 PTO 轴错误代码表 (参见第 *349* 页)。
ErrorId	Word	NoError	运动命令错误代码，在 Error 输出为 TRUE 时有效。请参阅 PTO 运动命令错误代码表 (参见第 *350* 页)。

MC_Reset_PTO 功能块

行为

此功能块复位所有与轴相关的错误，如有可能，允许从状态 ErrorStop 转换为 Standstill。它不影响功能块实例的输出。

图形表示形式



注： 在首次进入功能块时，必须将其配置为使用预期轴。在功能块上双击以显示功能块属性，选择相关轴，然后单击 Apply。

输入

下表描述功能块的输入：

输入	初始值	描述
Execute	FALSE	在上升沿启动功能块的执行。其他功能块输入的值可在 Execute 上升沿控制此功能块的执行。后面更改这些输入参数不会影响到正在进行中的执行。 这些输出在功能块终止时设置。

下表描述功能块的输入对象：

输入对象	类型	初始值	描述
Axis	PTOx	-	将执行功能块的实例。在控制器配置中声明名称。

输出

下表描述功能块的输出：

输出	初始值	描述
Done	FALSE	值为 TRUE 时，表示功能块执行完成，且未检测到任何错误。
Error	FALSE	如果为 TRUE，表示检测到错误。功能块执行结束。

下表描述功能块的输出对象：

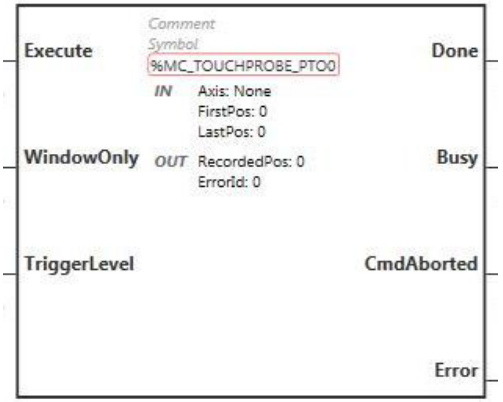
输出对象	类型	初始值	描述
ErrorId	Word	NoError	运动命令错误代码，在 Error 输出为 TRUE 时有效。请参阅 PTO 运动命令错误代码表 (参见第 *350* 页)。

MC_TouchProbe_PTO 功能块

功能描述

此功能块用于在探测器输入上激活触发事件。此触发事件用于记录轴位置和/或启动缓冲的移动。

图形表示形式



注： 在首次进入功能块时，必须将其配置为使用预期轴。在功能块上双击以显示功能块属性，选择相关轴，然后单击 Apply。

输入

下表描述功能块的输入：

输入	初始值	描述
Execute	FALSE	在上升沿启动功能块的执行。其他功能块输入的值可在 Execute 上升沿控制此功能块的执行。后面更改这些输入参数不会影响到正在进行中的执行。 这些输出在功能块终止时设置。 如果在执行功能块期间检测到第二个上升沿， 将中止当前执行并再次执行该功能块。 如果 Execute 输入随后设置为 0，则记录轴位置，并且 Done 输出针对一个 MAST 循环设置为 1。然后复位轴位置，并将 Done 输出设置为 0。
WindowOnly	FALSE	值为 TRUE 时，仅在 FirstPosition 和 LastPosition 所定义位置范围（窗口）内识别触发事件。
TriggerLevel	FALSE	值为 TRUE 时，则在上升沿进行位置捕捉或事件触发。 值为 FALSE 时，则在下降沿进行位置捕捉或事件触发。

下表描述功能块的输入对象：

输入对象	类型	初始值	描述
Axis	PTOx	-	将执行功能块的实例。在控制器配置中声明名称。
FirstPos	DINT	0	接受触发事件的绝对起始位置（启用窗口中包含的值）。
LastPos	DINT	0	接受触发事件的绝对结束位置（启用窗口中包含的值）。

输出

下表描述功能块的输出：

输出	初始值	描述
Done	FALSE	值为 TRUE 时，表示功能块执行完成，且未检测到任何错误。 如果在完成命令动作之前使用某轴上的一种运动中断同轴上的另一种运动，则 CmdAborted 设置为 TRUE 且 Done 设置为 FALSE。
Busy	-	值为 TRUE 时，表示正在执行功能块。 值为 FALSE 时，表示功能块的执行终止。 至少在 Busy 为 TRUE 时，功能块必须保留在应用程序的活动任务中。
CmdAborted	-	值为 TRUE 时，功能块因其他运动命令终止执行。
Error	FALSE	如果为 TRUE，表示检测到错误。功能块执行结束。

下表描述功能块的输出对象：

输出对象	类型	初始值	描述
RecordedPos	-	-	检测到触发事件的位置。
ErrorId	Word	NoError	运动命令错误代码，在 Error 输出为 TRUE 时有效。 请参阅 PTO 运动命令错误代码表 (参见第 *350* 页)。

注意：

- 在同轴上只允许使用此功能块的一个实例。
- 仅 MC_TouchProbe_PTO 功能块 Busy 输出上升沿后的第一个事件有效。在 Done 输出设置为 TRUE 后，将忽略后面的事件。该功能块需要重新激活才能响应其他事件。

MC_AbortTrigger_PTO 功能块

功能描述

此功能块用于中止连接到触发事件的功能块（例如 MC_TouchProbe_PTO）。

图形表示形式

注： 在首次进入功能块时，必须将其配置为使用预期轴。在功能块上双击以显示功能块属性，选择相关轴，然后单击 Apply。

输入

下表描述功能块的输入：

输入	初始值	描述
Execute	FALSE	在上升沿启动功能块的执行。其他功能块输入的值可在 Execute 上升沿控制此功能块的执行。后面更改这些输入参数不会影响到正在进行中的执行。 这些输出在功能块终止时设置。

下表描述功能块的输入对象：

输入对象	类型	初始值	描述
Axis	PTOx	-	将执行功能块的实例。在控制器配置中声明名称。

输出

下表描述功能块的输出：

输出	初始值	描述
Done	FALSE	值为 TRUE 时，表示功能块执行完成，且未检测到任何错误。
Error	FALSE	如果为 TRUE，表示检测到错误。功能块执行结束。

下表描述功能块的输出对象：

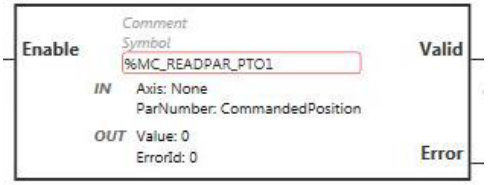
输出对象	类型	初始值	描述
ErrorId	Word	NoError	运动命令错误代码，在 Error 输出为 TRUE 时有效。请参阅 PTO 运动命令错误代码表 (参见第 *350* 页)。

MC_ReadPar_PTO 功能块

功能描述

此功能块用于从 PTO 获取参数。

图形表示形式



注： 在首次进入功能块时，必须将其配置为使用预期轴。在功能块上双击以显示功能块属性，选择相关轴，然后单击 Apply。

输入

下表描述功能块的输入：

输入	初始值	描述
Enable	FALSE	当值为 TRUE 时，执行功能块。可连续修改其他功能块输入的值，并且连续更新功能块输出。 当值为 FALSE 时，终止功能块执行并复位其输出。

下表描述功能块的输入对象：

输入对象	类型	初始值	描述
Axis	PTOx	-	将执行功能块的实例。在控制器配置中声明名称。
ParNumber	DINT	0	希望读取或写入的参数的代码。有关详细信息，请参阅 PTO 参数表。

输出

下表描述功能块的输出：

输出	初始值	描述
Valid	-	如果值为 TRUE，则功能块对象数据有效。
Error	FALSE	如果为 TRUE，表示检测到错误。功能块执行结束。

下表描述功能块的输出对象：

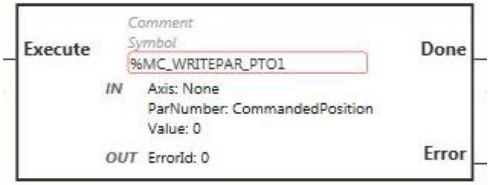
输出对象	类型	初始值	描述
Value	DINT	0	所请求参数的值。
ErrorId	Word	NoError	运动命令错误代码，在 Error 输出为 TRUE 时有效。请参阅 PTO 运动命令错误代码表 (参见第 *350* 页)。

MC_WritePar_PTO 功能块

功能描述

此功能块用于将参数写入到 PTO。

图形表示形式



注： 在首次进入功能块时，必须将其配置为使用预期轴。在功能块上双击以显示功能块属性，选择相关轴，然后单击 Apply。

输入

下表描述功能块的输入：

输入	初始值	描述
Execute	FALSE	在上升沿启动功能块的执行。其他功能块输入的值可在 Execute 上升沿控制此功能块的执行。后面更改这些输入参数不会影响到正在进行中的执行。 这些输出在功能块终止时设置。

下表描述功能块的输入对象：

输入对象	类型	初始值	描述
Axis	PTOx	-	将执行功能块的实例。在控制器配置中声明名称。
ParNumber	DINT	0	希望读取或写入的参数的代码。有关详细信息，请参阅 PTO 参数表。
Value	DINT	0	要写入使用 ParNumber 输入对象选择的参数的值。

输出

下表描述功能块的输出：

输出	初始值	描述
Done	FALSE	值为 TRUE 时，表示功能块执行完成，且未检测到任何错误。
Error	FALSE	如果为 TRUE，表示检测到错误。功能块执行结束。

下表描述功能块的输出对象：

输出对象	类型	初始值	描述
ErrorId	Word	NoError	运动命令错误代码，在 Error 输出为 TRUE 时有效。请参阅 PTO 运动命令错误代码表 (参见第 *350* 页)。

第15章
频率发生器

本章包含了哪些内容？

本章包含了以下主题：

主题	页
描述	410
配置	412

描述

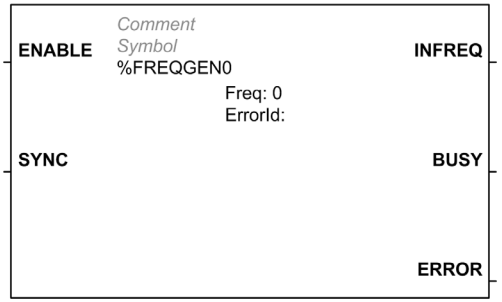
简介

频率发生器 FREQGEN 功能块 命令按指定频率输出方波信号。

频率可配置范围为 0 Hz 到 100 kHz，步长为 1 Hz。

示意图

此示意图为 FREQGEN 功能快：



输入

下表描述功能块的输入：

输入	初始值	描述
ENABLE	FALSE	若为 TRUE，则执行该功能块。可连续修改其他功能块输入的值，并且连续更新功能块输出。 若为 FALSE，则终止功能块执行并复位其输出。
SYNC	FALSE	检测到上升的脉冲沿时，会发射目标频率，无需等待进行中的周期输出结束。

下表描述功能块的输入对象：

输入对象	类型	初始值	描述
Freq	DWORD	-	Frequency Generator 输出信号的频率（以 Hz 为单位）。 在脉冲发生器属性 *(参见 Modicon M221 Logic Controller, 高级功能库指南)*表中指定频率 （范围：最小值 0 (0 Hz)...最大值 100000 (100 kHz)

输出

下表描述功能块的输出：

输出	初始值	描述
INFREQ	-	如果为 TRUE，则频率发生器信号将以 Freq 输入对象中指定的频率输出。
BUSY	-	如果为 TRUE，则表示正在执行功能块。 如果为 FALSE，则表示功能块的执行已终止。 至少在 BUSY 为 TRUE 时，该功能块必须保留在该应用程序的活动任务中。
ERROR	FALSE	如果为 TRUE，表示检测到错误。功能块执行结束。

下表描述功能块的输出对象：

输出对象	类型	初始值	描述
ErrorId	Word	NoError	错误代码，在 ERROR 输出为 TRUE 时有效。请参阅下面的 ErrorId 错误代码表。

ErrorId 错误代码

此表列出了功能块错误代码的值

名称	值	描述
NoError	0	未检测到错误。
OutputProtection	1007	一个或多个 PTO 对象激活了数字输出保护。有关更多详情，请参阅系统对象 %S10 和 %SW139 *(参见 Modicon M221, Logic Controller, 编程指南)*。
InvalidFrequencyValue	3002	频率 Freq 输入对象不在允许的范围内。

配置

概述

要配置 Pulse Generator 资源，请参阅配置脉冲发生器 (参见第 *78* 页)。

要将 Pulse Generator 资源配置为 FREQGEN，请参阅 配置频率发生器 (参见第 *87* 页)。

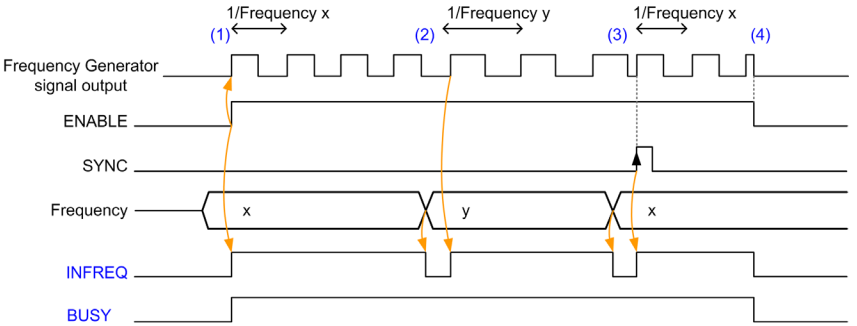
属性

FREQGEN功能块具有以下属性：

属性	描述	值
已使用	已使用的地址	如果选择，则在程序中使用此地址。
地址	%FREQGENi 频率发生器地址	实例标识符，其中 i 是从 0 到此逻辑控制器上可用的对象数。关于 FREQGEN 对象的数量上限，请参阅 对象的数量上限 (参见第 *32* 页)。
符号	符号	与此对象关联的符号。关于详细信息，请参阅 定义和使用符号 *(参见 SoMachine Basic, 操作指南)*。
Freq	频率	频率发生器输出信号的频率，以 Hz 为单位。 最小值：0 (0 Hz)。 最大值：100000 (100 kHz) 缺省值是 0。
注释	注释	可以将可选注释与此对象相关联。 双击**注释**列并键入注释。

时序图

本图显示 FREQGEN 功能块的时序：



(1) ENABLE 输入设置为 1。频率发生器信号在专用输出处发生。INFREQ 输出设置为 1。BUSY 输出设置为 1。
(2) 频率值已改变。INFREQ 输出设置为 0，直至专用输出处产生新的频率。BUSY 输出设置保持为 1。
(3) SYNC 输入设置为 1。当前频率发生器周期停止，开始新的周期。INFREQ 输出设置为 1。BUSY 输出设置保持为 1。
(4) ENABLE 输入设置为 0。频率发生器停止。INFREQ 输出设置为 0。BUSY 输出设置为 0。

当应用程序停止时，频率发生器停止，无需等待脉冲发生器周期结束。Error 输出保持在 FALSE。

如果检测到错误，在退出错误条件时将自动确认。

第16章
PID 功能

本章包含了哪些内容？

本章包含了以下部分：

节	主题	页
16.1	PID 操作模式	414
16.2	PID 自整定配置	416
16.3	PID 标准配置	419
16.4	PID 助手	430
16.5	PID 编程	441
16.6	PID 参数	447

第16.1节
PID 操作模式

PID 操作模式

简介

SoMachine Basic PID 控制器提供四种不同的操作模式，用户可在 SoMachine Basic 中 **PID 助手**的**常规**选项卡 (参见第 *432* 页)上对其进行配置。

PID 操作模式有：

- PID 模式
- AT + PID 模式
- AT 模式
- 字地址

PID 模式

当PID 控制器启动时，缺省情况下简单 PID 控制器模式处于活动状态。要成功控制此流程，须提前获取将在 **PID** 选项卡 (参见第 *436* 页)上指定的增益值 Kp、Ti 和 Td。您可以在 **PID 助手**画面 (参见第 *430* 页)的 **PID** 选项卡上选择校正器类型的控制器 (PID 或 PI) 。若选择 PI 校正器类型，则会禁用微分时间 **Td** 字段。

若使用PID 模式，则会禁用自整定功能且无法使用**助手配置** (参见第 *437* 页)画面上的 **AT 选项卡**。

AT + PID 模式

在此模式下，自整定功能将在PID 控制器启动时激活。随后，自整定功能将计算增益值 Kp、Ti 和 Td (参见第 *436* 页) 以及 PID 操作 (参见第 *439* 页)的类型。在自整定流程结束后，控制器会切换到PID 模式并使用自整定计算得到的参数用来调整设定值。

如果自整定算法检测到错误 (参见第 *446* 页)：

- 不对任何 PID 参数进行计算。
- 自整定输出将设置为在启动自整定前应用到流程的输出。
- **PID 状态列表**下拉列表中显示错误消息。
- 取消 PID 控制。

当处于 AT + PID 模式时，从自整定转换为 PID 模式是自动无缝进行的。

AT 模式

在此模式下，自整定功能将在PID 控制器启动并自动计算增益值 Kp、Ti 和 Td (参见第 *436* 页) 以及 PID 操作 (参见第 *439* 页)的类型时激活。在完成自整定流程并成功确定 Kp、Ti 和 Td 参数以及 PID 操作 (参见第 *439* 页)的类型后（或在自整定算法中检测到错误后），自整定数字输出将设置为 0，并在 PID 状态列表 (参见第 *445* 页)下拉列表中显示**自整定完成**的消息。然后，PID 控制器停止并等待。计算得出的 Kp、Ti 和 Td PID 系数可在其各自的存储器字 (%MWx) 中使用。

字地址

选择该 PID 模式是通过将所需值赋给与该选择关联的字地址实现的：

- %MWxx = 0：控制器已禁用。
- %MWxx = 1：控制器运行于简易PID 模式下。
- %MWxx = 2：控制器在 AT+ PID 模式下运行。
- %MWxx = 3：控制器仅在 AT 模式下运行。
- %MWxx = 4：控制器在简单 PID 模式下运行，带有 PI 修正器类型。

此 word address 模式让您能够通过应用程序管理 PID 控制器操作模式，从而能够适应您的要求。

第16.2节
PID 自整定配置

PID 自整定配置

简介

本节将指导您完成使用自整定 (AT) 配置 SoMachine Basic PID 控制器所需的所有步骤。

本节包含以下步骤：

步骤	主题
1	配置模拟量通道 (参见第 *416* 页)
2	PID 配置的先决条件 (参见第 *416* 页)
3	配置 PID (参见第 *417* 页)
4	控制设置 (参见第 *418* 页)

步骤 1：配置模拟量通道

PID 控制器使用模拟量反馈信号（称为过程值）来计算用于控制过程的算法。逻辑控制器具有用于获取该过程值的内置模拟量输入。

如果模拟量输出正用于驱动要控制的系统，请确保正确配置该模拟量输出。请参阅逻辑控制器的模拟量输出扩展模块。

步骤 2：PID 配置的先决条件

在配置 PID 控制器前，请确保以下阶段已执行：

阶段	描述
1	已在程序中启用 PID (参见第 *442* 页)。
2	**扫描模式**已设置为定期 (参见第 *444* 页)。

步骤 3：配置 PID

将固态输出与 PID 功能结合使用。使用输出继电器可能导致快速超过其使用寿命周期极限，导致触点冻开或者焊接关闭，进而造成继电器无法运行。

⚠ 警告

意外的设备操作或设备无法操作

- 请勿将继电器输出与 PID 功能搭配使用。
- 如果需要数字量输出以驱动要控制的系统，则仅使用固态输出。

不遵循上述说明可能导致人员伤亡或设备损坏。

要实施能够自整定的 PID 控制器，请执行以下步骤：

步骤	动作
1	在 **PID 助手**画面（离线模式）的**General** tab (参见第 *432* 页)中，选择 **AT+PID**（或 **AT**），或从Operating Modes (参见第 *414* 页)中选择可将关联的**字地址**设置为 2 或 3。
2	激活 **PID 状态**复选框，并在字段中输入存储器字的地址。
3	在**Input** tab (参见第 *435* 页)中，输入用作测量值的模拟量输入的地址。
4	如果需要**转换**或**报警**，请参阅 **PID 助手**画面的**输入**选项卡 (参见第 *435* 页)。
5	在 **PID** 选项卡 (参见第 *436* 页)中，输入设置点的值。通常，该值为存储器地址或模拟量输入。
6	**PID** 选项卡中的**修正器类型**必须设置为 **PID** 或 **PI**。
7	设置 **PID** 选项卡中的**参数**：**Kp (x0,01)**、**Ti (x0,1s)** 和 **Td (x0,1s)**。如果 **AT+PID** 或 **AT** 处于运行模式 (参见第 *414* 页)，这些参数应为**存储器字地址 (%MWxx)**，以便自整定算法填入这些参数的计算值。
8	输入 PID**采样周期** (Ts (参见第 *428* 页))（在 **PID** 选项卡中）。**采样周期**是关键参数，必须仔细确定。
9	在 **AT** 选项卡中，**AT 模式**在缺省情况下必须设置为**授权**。如果已激活**测量范围**（**授权**复选框），则输入**最小值**和**最大值**。从列表中选择**动态 AT 修正器**，该列表包含**快速**、**中速**、**慢速**或**字地址**修正器类型。有关详细信息，请参阅 PID 助手 (参见第 *437* 页)中的 **AT** 选项卡。
10	在 **AT** 选项卡中，输入 **AT 触发器**存储器位，以存储自整定期间步进变化的值。有关详细信息，请参阅 PID 助手 (参见第 *437* 页)中的 **AT** 选项卡。
11	在**Output** tab (参见第 *439* 页)中，将**动作**设置为列表中的**位地址**。在**位**字段中输入存储器位地址。如有需要，**限制**可在“输出”选项卡 (参见第 *439* 页)中进行配置。 在**模拟量输出**字段中，设置字地址：模拟量输出或存储器字。将**Output PWM** (参见第 *439* 页) 设置为**授权**。在手动模式中，在**周期（0.1 秒）**字段中输入值，或在**输出**字段中输入输出的存储器字地址。有关手动模式操作的详细信息，请参阅“输出”选项卡 (参见第 *439* 页)。
12	单击**确定**以确认 PID 控制器配置。

步骤 4：控制设置

将固态输出与 PID 功能结合使用。使用输出继电器可能导致快速超过其使用寿命周期极限，导致触点冻开或者焊接关闭，进而造成继电器无法运行。

⚠ 警告

意外的设备操作或设备无法操作

- 请勿将继电器输出与 PID 功能搭配使用。
- 如果需要数字量输出以驱动要控制的系统，则仅使用固态输出。

不遵循上述说明可能导致人员伤亡或设备损坏。

要以 AT+PID 操作模式 (参见第 *414* 页)开始操作，请执行以下步骤：

步骤	动作
1	将 PC 连接到控制器并传输应用程序。
2	将控制器切换到“运行”模式。

注意： 在将控制器切换到“运行”模式之前，请检查机器的操作条件是否允许应用程序的其余部分使用“运行”模式。

步骤	动作
1	创建动态数据表，该表包含配置期间所定义的对象。有关动态数据表创建的详细信息，请参阅 *SoMachine Basic 操作指南*。
2	验证过程值和应用程序值之间的一致性。该测试非常重要，因为 PID 控制器的操作是否成功取决于测量值的准确性。如果对测量值的准确性存在任何疑问， 请将逻辑控制器设置为“停止”状态并验证模拟量通道的接线。 如果执行器不受控制： ● 对于模拟量输出，请验证模拟量通道的输出电压或电流。 ● 对于 PWM 输出，请验证： ❍ 专用输出的 LED 是否亮起 ❍ 电源和 0 伏电路的接线 ❍ 执行器电源是否正在供电
3	在动态数据表中，请验证： ● 输出模式是否已设置为自动。 ● 应用程序所需的所有参数是否都已设置为适当的值。
4	设置逻辑控制器扫描周期，以使 PID 控制器的**采样周期** (Ts) 值正好为扫描周期的倍数。 有关如何确定采样周期的详细信息，请参阅调节 PID (参见第 *423* 页)。
5	自整定序列完成后，参数 Kp、Ti 和 Td 将写入逻辑控制器的 RAM 存储器。只要应用程序有效（电源关闭不满 30 天）且未执行冷启动，这些值将一直保存。

每当在 AT **触发器**存储器位检测到上升沿，自整定过程就会重复。

第16.3节
PID 标准配置

本节包含了哪些内容？

本节包含了以下主题：

主题	页
PID 字地址配置	420
使用自整定 (AT) 进行 PID 调节	423
手动模式	426
确定采样周期 (Ts)	428

PID 字地址配置

简介

本节将指导您完成所有所需步骤，以便使用字地址PID操作模式配置 SoMachine Basic (参见第 *414* 页) 控制器。该模式的使用灵活性要优于其他 PID 模式。

本节包含以下步骤：

步骤	主题
1	PID 配置的先决条件 (参见第 *420* 页)
2	配置 PID (参见第 *420* 页)
3	控制设置 (参见第 *421* 页)

步骤 1：PID 配置的先决条件

在配置PID 前，请确保以下阶段已执行：

阶段	描述
1	已配置模拟量输入和模拟量输出（如有需要）。
2	已在程序中启用 PID (参见第 *442* 页)。
3	扫描模式已设置为定期 (参见第 *444* 页)。

步骤 2：配置 PID

将固态输出与 PID 功能结合使用。使用输出继电器可能导致快速超过其使用寿命周期极限，导致触点冻开或者焊接关闭，进而造成继电器无法运行。

⚠ 警告

意外的设备操作或设备无法操作

- 请勿将继电器输出与 PID 功能搭配使用。
- 如果需要数字量输出以驱动要控制的系统，则仅使用固态输出。

不遵循上述说明可能导致人员伤亡或设备损坏。

以下步骤解释了如何在**字地址**模式下实现 PID 控制器。有关如何配置 PID 的详细信息，请参阅 PID 助手 (参见第 *430* 页)。

对于 PID 参数的动态修改（离线和在线模式下），请在关联字段中输入存储器地址，这样就无需切换至离线模式进行值的实时更改。

步骤	动作
1	在 **PID Assistant**画面的**常规**选项卡中（离线模式下），选择**Operating Modes**下拉列表中的**字地址**。选中与 PID 状态关联的框，并在字段中输入存储器字的地址。
2	在**输入** (参见第 *435* 页)选项卡中，输入用作测量值的模拟量输入的地址。如果需要**转换**或**报警**，请参阅 PID 助手 (参见第 *430* 页)的**输入**选项卡 (参见第 *435* 页)。
3	在 **PID** 选项卡中，输入**设置点**的值。通常，该值为存储器地址或模拟量输入。 **参数**（Kp、Ti 和 Td）应为**存储器字地址 (%MWxx)**。 在 PID 选项卡 (参见第 *436* 页)中输入 PID **采样周期** (Ts (参见第 *428* 页))。该参数也可以是存储器字（然后，值可以使用动态数据表来设置）。 在**字地址**操作模式中，**修正器类型**被设置为**自动**且呈灰色（不能对其进行手动修改）。
4	在 **AT** 选项卡中， AT 模式应被选中，以便进行**授权**。输入**动态修正器**和 **AT 触发器**。有关详细信息，请参阅 **PID 助手**画面中的 **AT**选项卡 (参见第 *437* 页)。
5	在**输出**选项卡中，**操作**应设置为**位地址**。输入**存储器位地址**。如有需要，可在“输出”选项卡 (参见第 *439* 页)中配置**限制**。在**模拟量输出**字段中，设置字地址：模拟量输出或存储器字。 如有需要，请设置**输出 PWM**，请参阅 PID 助手 (参见第 *430* 页)中的**输出**选项卡 (参见第 *439* 页)。
6	单击**确定**以确认 PID 控制器配置。

步骤 3：验证设置

步骤	动作
1	将 PC 连接到逻辑控制器并传输应用程序。
2	将逻辑控制器切换到“运行”模式。

注意： 在将逻辑控制器切换到“运行”模式之前，请验证机器的操作情况是否允许为应用程序的其余部分切换到“运行”模式。该过程与 AT 和 AT+PID 操作模式下使用的过程相同。字地址配置使您可以通过软件来修改 PID 操作模式。对于 PID 模式，该过程非常简单，如果参数（Kp、Ti、Td 和 Ts）已知，则无需执行自整定。

下表提供了设置 PID 控制器的一般步骤

步骤	动作
1	创建动态数据表，该表包含配置期间所定义的对象。有关详细信息，请参阅 *SoMachine Basic 操作指南*。
2	验证过程值和动态数据表所定义的其他值是否一致。 如果在测量值的准确度方面存在任何疑问，请将逻辑控制器设置为“停止”并验证模拟量通道的接线。 如果发现执行器不受控制，请执行以下操作： ● 对于模拟量输出，请验证模拟量通道的输出电压或电流。 ● 对于 PWM 输出，请验证： ❍ 专用输出的 LED 是否亮起 ❍ 电源和 0 伏电路的接线是否正确 ❍ 执行器电源是否正在供电
3	设置逻辑控制器扫描周期，以使 PID 控制器的**采样周期** (Ts) 正好为扫描周期的倍数。 有关采样周期的详细信息，请参阅确定采样周期 (参见第 *428* 页)。
4	如果计划使用自整定 (参见第 *423* 页)功能，您可能需要运行手动模式 (参见第 *426* 页)以确定 **PID 助手**的 **AT** 选项卡 (参见第 *437* 页)所定义的**动态修正器**和 **AT 触发器**。
5	在回路控制上电后使用动态数据表： ● 设置操作模式 (参见第 *414* 页)。 ● 启用 PID 控制器 (参见第 *442* 页)。 ● 根据所选操作模式，将配置 (参见第 *420* 页)阶段所定义的值设置为合适的值。

使用自整定 (AT) 进行 PID 调节

简介

自整定模式允许 Kp、Ti、Td 和操作参数进行自动调节，以实现 PID 功能的优化整合。SoMachine Basic 所提供的自整定功能尤其适合热过程的自动调节。

本节包含以下主题：

- 自整定要求
- 自整定过程的描述
- 存储计算所得的系数
- 调整 PID 参数
- 启动自整定
- 自整定和 PID 控制的使用限制

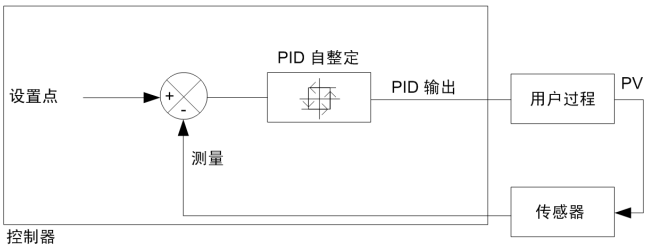
自整定要求

使用自整定功能时，请确保控制过程和逻辑控制器满足以下要求：

- 过程要求：
  - 过程必须是一个稳定的开放回路系统。
  - 过程在整个操作范围内必须大部分都是线性的。
  - 对模拟量输出进行级别更改的过程响应遵循瞬变渐近模式。
  - 过程处于稳定状态，且在自整定流程开始时的输入为空。
  - 整个过程都必须没有干扰。否则，计算得出的参数将是错误的，也可能会导致自整定过程无法正确运行。
- 配置要求：
  - 将逻辑控制器配置为定期扫描模式，以确保自整定功能的正确运行。
  - 只在没有其他 PID 控制器正在运行的情况下使用自整定功能。
  - 将 Kp、Ti 和 Td 系数配置为存储器字地址 (%MWxx)。
  - 将**输出**选项卡中的动作类型设置为存储器位地址 (%Mxx)。

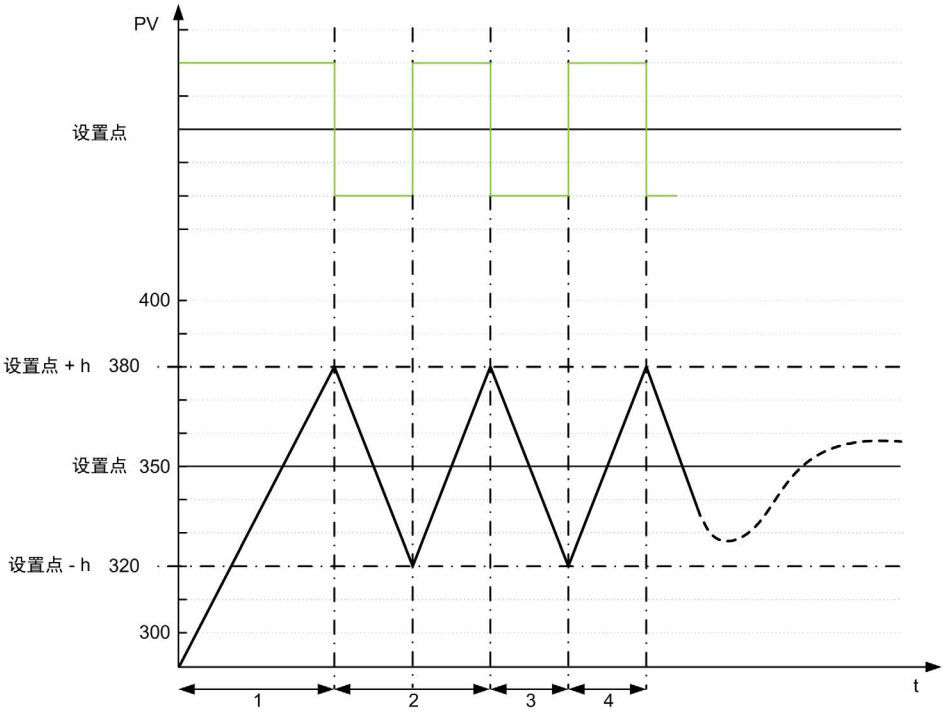
自整定过程的描述

下图显示了控制器和应用程序中的自整定：



自整定校准过程的描述

自整定校准过程分为 4 个连续的阶段。要成功完成自整定，必须执行该过程的所有阶段。下面的过程响应曲线和表描述了 SoMachine Basic PID 自调节功能的 4 个阶段：



PV 过程值
PID 输出
h = 1% (**最大值** - **最大值**) (**AT** 选项卡中的**测量范围**字段
---- PID 激活
1...4 自整定阶段 (参见下表)

下表介绍了自整定阶段：

自整定阶段	描述
1	在过程值达到设置点 + h 之前，PID 输出被强制为**输出**选项卡中**限制**字段的**最大值**。
2	自整定阶段 2 包含 2 个步骤： **1.** 在过程值达到设置点 - h 之前，PID 输出被强制为**输出**选项卡中**限制**字段的**最小值**。 **2.** 在过程值达到设置点 + h 之前，PID 输出被强制为**输出**选项卡中**限制**字段的**最大值**。
3	在过程值达到设置点 - h 之前，PID 输出被强制为**输出**选项卡中**限制**字段的**最小值**。
(1) 自整定开始之前最后应用于过程的输出被用作自整定过程的开始点和松弛点。	

自整定阶段	描述
4	自整定阶段 4 包含 2 个步骤： **1.** 在过程值达到设置点 + h 之前，PID 输出被强制为**输出**选项卡中**限制**字段的**最大值**。 **2.** PID 输出被强制为**输出**选项卡中**限制**字段的**最小值**，计算 PID 参数，且 PID 激活。
(1) 自整定开始之前最后应用于过程的输出被用作自整定过程的开始点和松弛点。	

注意： 当手动输出控制在自整定校准过程中被激活时，Kp、Ti 和 Td 参数无法计算。一旦手动输出控制结束，便再次启动自整定校准。

存储计算所得的系数

自整定流程完成后，将使用计算所得的值设置分配给 Kp、Ti 和 Td 系数的存储器字及动作类型。这些值将写入 RAM 存储器，只要应用程序有效且不执行冷启动 (%S0)，这些值将一直保存在逻辑控制器中。

如果系统未受到外部干扰，计算所得的值可能会写入 PID 控制器的设置中（请参阅 PID **助手的** PID (参见第 *436* 页) 选项卡）。这样，PID 控制器操作模式可以设置为 PID 模式。

调整 PID 参数

自整定方法可能会提供一个非常动态的命令，从而导致设置点步进变化期间发生有害过冲。要优化从自整定获取的 PID 参数（Kp、Ti、Td）所提供的过程调节，您还可以直接从 **PID 助手**画面的 **PID** 选项卡中或通过相应的存储器字 (%MW) 对这些参数值进行手动调整。有关手动参数调整的更多详细信息，请参阅附录 (参见第 *447* 页)。

启动自整定

在 **AT** 选项卡中，**AT 触发器**可以实现自整定序列的重复性。自整定过程将在已链接至 **AT 触发器**的信号的各个上升沿上启动。

如要配置自整定，请参阅 AT 选项卡。

自整定使用限制

加热过程常被整合到具有纯滞后的一阶模型。此类模型可用两个关键参数来描述：

- 时间常量 τ
- 延迟时间 θ

自整定最适合时间常量 (τ) 和延迟时间 (θ) 满足以下条件的过程：

- 10 秒 < (τ + θ) < 2700 秒（例如，45 分钟）
- 2 < τ / θ < 20

手动模式

简介

可通过 **PID 助手**画面 (**Output** tab (参见第 *439* 页)) 访问手动模式。此模式可让您从 PID 绕过命令。使用手动模式可以实现两个主要目标：

- 初始化设置
- 确定采样周期。

描述

手动模式可以让您指定 **Output** value (参见第 *439* 页)。此操作特别适合用于测试系统响应能力。

在**输出**选项卡 (参见第 *439* 页)上，将**位地址**设置为 1 以激活手动模式。如果设置为“允许”，则手动模式是唯一可访问的模式。

应用程序

当手动模式处于活动状态时，输出将分配有您设置的固定值。该输出值介于 0 至 10,000 之间（对于 PWM 输出，该值为 0 至 100%）。

此外，您也可以使用手动模式进行试用，以确定最小/最大输出限制。

手动模式也需要使用过程响应曲线方法 (参见第 *428* 页)，以帮助找到正确的采样时间 (Ts)。

启动手动模式

在启动手动模式之前，请务必确保逻辑控制器“运行/停止”开关处于“运行”位置。

使用动态数据表启动手动模式：

步骤	描述
1	通过将专用存储器位设置为 1 启用手动模式。有关详细信息，请参阅**输出**选项卡 (参见第 *439* 页)。
2	如果使用 PWM，请将 PWM 期间设置为所需的值。
3	在 **PID 助手**画面的**常规**选项卡 (参见第 *432* 页)上，将与操作模式关联的存储器字设置为 1（PID 模式）。有关使用字地址的操作模式的详细信息，请参阅操作模式说明 (参见第 *414* 页)。
4	在**输出**选项卡 (参见第 *439* 页)上，将与手动输出关联的存储器字设置为所需的值。在系统处于其初始状态的条件下，可以多次选择该手动设置点值。
5	启用回路控制器 (参见第 *420* 页)。

停止手动模式

使用动态数据表停止手动模式：

步骤	描述
1	禁用回路控制器 (参见第 *420* 页)。
2	通过将专用存储器位设置为 0 禁用手动模式。有关详细信息，请参阅**输出**选项卡 (参见第 *439* 页)。
3	在 PID 控制器的**常规**选项卡 (参见第 *432* 页)上，将与操作模式关联的存储器字设置为 0。有关使用字地址的操作模式的详细信息，请参阅操作模式说明 (参见第 *414* 页)。
4	在**输出**选项卡 (参见第 *439* 页)上，将与手动输出关联的存储器字设置为 0。

确定采样周期 (Ts)

简介

采样周期 (Ts) 是PID 调节的关键参数。应在 **PID 助手**画面的 **PID** 选项卡 (参见第 *436* 页)中小心地设置采样周期 (Ts)。该参数与要控制的过程的时间常量 (τ) 密切相关。

本节将介绍如何使用在线模式，还将介绍采用周期 (Ts) 的两种确定方式：

- 过程响应曲线方式，
- 试错法。

过程响应曲线方式

这种方式是一个开放回路过程，旨在确定要控制的过程的时间常量。首先，需要确保过程可以通过一阶时间延迟模型来描述。其原理非常简单：在记录过程输出曲线时，将步进变化应用于过程的输入。然后，使用图形方式确定过程的时间延迟。

要使用过程响应曲线方式来确定采样周期 (Ts)，请执行以下操作：

步骤	动作
1	假定您已配置 PID 的**常规**、**输入**、**PID**、**AT** 和 **输出**选项卡中的各项设置。
2	选择 **PID 助手**画面中的**输出**选项卡 (参见第 *439* 页)。
3	选择**手动模式**下拉列表中的**允许**或**地址位**，以对手动输出进行授权。
4	将“输出”字段设置为高级别（在 [5,000...10,000] 范围内）。
5	将应用程序下载到逻辑控制器。有关如何下载应用程序的详细信息，请参阅 *SoMachine Basic 操作指南*。
6	运行 PID 并检查响应曲线是否上升。
7	响应曲线达到稳定状态时，停止 PID 测量。
8	使用以下图形方式来确定控制过程的时间常量 (T)： **1.** 使用以下公式计算上升到 63% 时 ($S_{[63\%]}$) 的过程值输出： $S_{[63\%]} = S_{[初始]} + (S_{[结束]}-S_{[初始]})x63\%$ **2.** 利用图形计算对应于 S(63%) 的时间横坐标 ($t_{[63\%]}$)。 **3.** 利用图形计算对应于过程响应上升起始位置的初始时间 ($t_{[初始]}$)。 **4.** 使用以下关系计算控制过程的时间常量 (τ)：$\tau = t_{[63\%]}-t_{[初始]}$
9	根据上一步所确定的 (τ) 值，使用以下规则计算采样周期 (Ts)(1)：Ts = τ/75
10	设置定期扫描模式的扫描周期，以使采样周期 (Ts) 为扫描周期的整数倍：扫描周期 = Ts / n ，其中 n 为正整数 (2)
(1) 采样周期的基本单位为 10 毫秒。因此，您应对 Ts 的值进行舍入操作，以使其最接近于 10 毫秒。 **(2)** 您必须选择“n”，以使得出的扫描周期为位于 [2...150] 毫秒范围内的正整数。	

试错法

试错法需要向自整定功能提供连续的采样周期猜测值，直到算法成功整合出令人满意的 Kp、Ti 和 Td 值。

注意： 与过程响应曲线方式不同，试错法并非基于过程响应的任何近似法则。但是，它的优点在于：它能整合出与实际值位于同一数量级的采样周期值。

要执行自整定的反复试验估算，请执行以下操作：

步骤	动作
1	选择 PID 配置窗口中的 **AT** 选项卡。
2	将自整定的**输出限制**设置为 **10,000**
3	将应用程序下载到逻辑控制器。有关如何下载应用程序的详细信息，请参阅 SoMachine Basic 操作指南。
4	选择 **PID 助手**画面中的 **PID**选项卡。
5	在**采样周期**(1)字段中提供第一个或第 n 个猜测值。
6	启动自整定 (参见第 *416* 页)。
7	稍侯，直到自整定过程结束。
8	可能会出现两种情况： ● **自整定成功完成：**继续执行步骤 10。 ● **自整定失败：**请参阅自整定检测到的错误代码 (参见第 *446* 页)。这意味着采样周期 (Ts) 的当前猜测值不正确。根据需要，多次尝试新的 Ts 猜测值并重复步骤 3 至步骤 8，直到自整定过程最终整合。
9	请根据以下原则提供新的 Ts 猜测值： ● 自整定结束，其检测到的错误代码为 800C（十六进制）。这意味着采样周期 Ts 太大。减小 Ts 值，以便提供新的猜测值。 ● 自整定结束，其检测到的错误代码为 800A（十六进制）。这意味着采样周期 Ts 太小。增大 Ts 值，以便提供新的猜测值。
10	根据需要，在 **PID 助手**画面的 PID 选项卡 (参见第 *436* 页)中调节 PID 控制参数(2)（Kp、Ti 和 Td）。

(1) 如果您未掌握采样周期可能范围的初步迹象，请将此值设置为可能的最小值：1（10 毫秒为 1 个单位）。
(2) 如果该组控制参数提供的 PID 调节不能提供完全令人满意的结果，您可以不断进行采样周期反复试验求值优化，直到得出一组正确的 Kp、Ti 和 Td 控制参数。

在线模式

处于在线模式时，如果逻辑控制器有定期任务，Ts 字段中所显示的值（在 **PID 助手**画面 (参见第 *430* 页)中）可能不同于所输入的参数 (%MW)。Ts 值是定期任务的整数倍，而 %MW 值是逻辑控制器所读取的值。

第16.4节
PID 助手

本节包含了哪些内容？

本节包含了以下主题：

主题	页
访问 PID 助手	431
常规”选项卡	432
输入选项卡	435
PID 选项卡	436
AT 选项卡	437
输出选项卡	439

访问 PID 助手

简介

通过使用 SoMachine Basic 的 **PID 助手**窗口，用户便可配置PID 控制器。

配置助手

在 PID 属性表中，单击**配置 [...]** 按钮。**PID 助手**画面随即出现。

PID 助手画面如下图所示：

根据您是处于离线或在线模式，**PID 助手**画面可显示多个不同的选项卡。

选项卡	访问模式	链接
常规	离线	“常规”选项卡 (参见第 *432* 页)
输入	离线	“输入”选项卡 (参见第 *435* 页)
PID	离线	“PID”选项卡 (参见第 *436* 页)
AT	离线	“AT”选项卡 (参见第 *437* 页)
输出	离线	“输出”选项卡 (参见第 *439* 页)

选择某一操作模式后，包含空字段的选项卡便会如 中所示进行显示，且字段的边缘将填充为红色。

常规”选项卡

简介

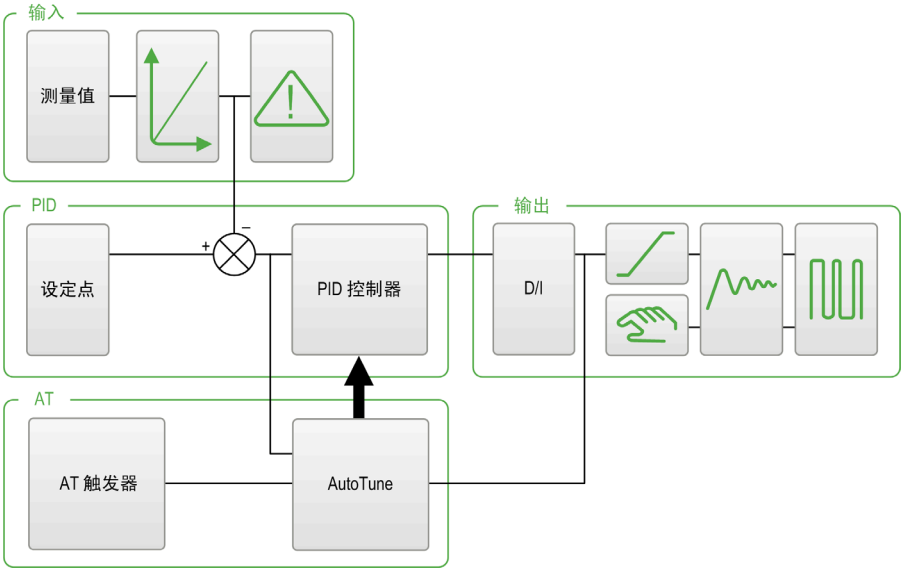
本节介绍 PID 的**常规** 选项卡。当您以离线模式访问“PID 助手”时，缺省情况下显示**常规**选项卡。

描述

下表介绍了**常规**选项卡上的设置。

参数	描述
操作模式	代表要使用的PID 模式： ● 未配置 ● PID ● AT + PID ● AT ● 字地址 有关操作模式的详细信息，请参阅 PID 操作模式 (参见第 *414* 页)。
字地址	您可在此文本框中提供内存字 (%MWxx)，以便通过编程来设置操作模式。根据要设置的操作模式，内存字可采用下列四个可能的值： ● %MWx = 0（已禁用 PID） ● %MWx = 1（仅设置 PID） ● %MWx = 2（设置 AT + PID） ● %MWx = 3（仅设置 AT） ● %MWx = 4（仅设置 PI）
PID 状态	如果选中此框以启用该选项，则可在PID 控制器使用的关联字段 (%MWxx) 中提供存储器字，以在运行 PID 控制器和/或自整定功能时存储当前 PID 状态。 有关详细信息，请参阅 PID 状态和检测到的错误码 (参见第 *445* 页)。

图形助手



图形助手有助于对PID 功能的构建方式进行形象化。该动态图像随配置进行更新。

下列图标阐述了何时可对该图标进行访问，以及单击该图标时所出现的现象：

显示	描述
设定点	单击此按钮以显示 **PID** 选项卡 (参见第 *436* 页)的“设置点”字段。
PID 控制器	单击此按钮以显示 **PID** 选项卡 (参见第 *436* 页)。
D/I	单击此按钮以显示**输出**选项卡 (参见第 *439* 页)。
测量值	单击此按钮以显示**输入**选项卡 (参见第 *435* 页)。

显示	描述
	单击此按钮以显示 AT 选项卡 (参见第 *437* 页)。
	单击此按钮以显示 AT 选项卡 (参见第 *437* 页)。
	在**输入**选项卡 (参见第 *435* 页)的**转换**区域中选中**授权**选项时，将出现该按钮。
	在**输入**选项卡 (参见第 *435* 页)的**报警**区域中选中**授权**选项时，将出现该按钮。
	若**限制**与**输出**选项卡 (参见第 *439* 页)中限制区域的约束条件不一致，则会出现该按钮。
	若手动模式与**输出**选项卡 (参见第 *439* 页)中手动模式区域的约束条件不一致，则会出现该按钮。
	单击此按钮以显示**输出**选项卡 (参见第 *439* 页)。
	在**输出**选项卡 (参见第 *439* 页)的“输出 PWM”区域中选中**授权**选项时，将出现该按钮。

输入选项卡

简介

本节介绍 PID 的**输入**选项卡。**输入**选项卡用于输入 PID 输入参数。

仅当处于离线模式以及从**常规**选项卡中选择某一操作模式后，方可访问此选项卡。

注意： 要在冷重启后保留输入值，使用字存储器 (%MW)，勿使用模拟量输入 (%IW)。

描述

下表描述您可以定义的设置。

参数	描述	
测量	指定含有待控制的过程值的变量。 默认量程为 0 到 10000。您可以输入存储器字 (%MWxx) 或模拟量输入。	
转换	**授权**	激活此框以将过程值 [**0...10000**] 转换为线性范围 [**最小值...最大值**]。 此转换还适用于设置值。
	最小值 **最大值**	指定转换量程的最小值和最大值。随后，过程值便会在 [**最小值...最大值**] 区间内自动重新转换。 **最小值**或**最大值**可为存储器字 (%MWxx)、 常量字 (%KWxx) 或介于 -32768 和 +32767 之间的某个值。 **注意**：**最小值**必须小于**最大值**。
滤波器	**授权**	激活此框以将滤波器应用于测量的输入。
	(100 毫秒)	指定介于 0 和 10000 之间的滤波器值或存储器字地址 (%MWxx))。滤波器时基单位为 100 毫秒。
报警	**授权**	激活此框以在输入变量中激活报警。 报警值应根据转换阶段完成后得到的过程变量进行确定。转换激活时，报警值必须介于**最小值**到**最大值**之间。否则，报警值将介于 0 到 10000 之间。
	低 **输出**	在**低**字段中指定低位报警值。 该值可为存储器字 (%MWxx)、常量 (%KWxx) 或直接值。 **输出**必须包含位的地址（当达到下限时该位将设为 1）。**输出**可为存储器位 (%Mxx) 或输出。
	高 **输出**	在**高**字段中指定高位报警值。 该值可为存储器字 (%MWxx)、常量 (%KWxx)) 或直接值。 **输出**必须包含位的地址（当达到上限时该位将设为 1）。**输出**可为存储器位 (%Mxx) 或输出。

PID 选项卡

简介

PID 选项卡用于输入内部PID 参数。

仅当处于离线模式以及已从**常规**选项卡中选择某一操作模式后，方可访问该选项卡。

描述

下表描述您可以定义的设置：

参数	描述	
设置点	指定PID设置点值。该值可为内存字 (%MWxx)、常量字 (%KWxx) 或直接值。 因此，当禁止转换时，该值必须介于 0 和 10000 之间。否则，它必须介于用于转换的**最小值**和**最大值**之间。	
修正器类型	若先前已在 PID 属性表中选择 **PID** 或 **AT + PID** 操作模式，PID下拉列表中选择所需的校正器类型（**PID** 或 **PI**）。若已选择其他模式（**AT** 或 **字地址**），则**校正器类型**将设为**自动**并灰化（无法手动对其进行修改）。 若已从下拉列表中选择 **PI**，Td 参数将强制设为 0 并禁用该字段。	
参数 (1)	Kp (x0,01s)	指定PID比例增益，并乘以 100。 该值可为内存字 (%MWxx)、常量字 (%KWxx) 或直接值。 Kp 参数的有效范围为：0 < Kp < 10000。 **注意：**若误将 Kp 设置为 0（Kp ≤ 0 时无效），则 PID 功能将自动赋以缺省值 Kp=100。
	Ti (x0,1s)	为 0.1 秒时基指定积分时间。 该值可为内存字 (%MWxx)、常量字 (%KWxx) 或直接值。 该值必须介于 0 到 36000。 **注意：**要禁用 PID 的积分操作，请将该系数设置为 0。
	Td (x0,1s)	指定 0.1 秒时基的微分时间。 该值可为内存字 (%MWxx)、常量字 (%KWxx) 或直接值。 该值必须介于 0 到 10000 之间。 **注意：**要禁用 PID 的微分操作，请将该系数设置为 0。
采样周期	在此处指定 10PID-2 秒（10 毫秒）时基的 采样周期。 该值可为内存字 (%MWxx)、常量字 (%KWxx) 或直接值。 该值必须介于 1（0.01 秒）到 10000（100 秒）。	

(1) 启用自整定后，您便无需再设置 Kp、Ti 和 Td 参数，因为自整定算法将以自动执行和编程方式自动对这些参数进行设置。在此情况下，您必须在这些字段中仅输入**内部字地址** (%MWxx)。启用自整定功能后请勿输入常量或直接值。

AT 选项卡

简介

AT 选项卡与自整定功能有关。有关更多详细信息，请参阅使用自整定进行 PID 调节 (参见第 *423* 页)。

仅当处于离线模式以及已从**常规**选项卡中选择某一操作模式后，方可访问该选项卡。

描述

PID 自整定是一个开放回路过程，它直接作用于控制过程，并只接受过程值 (PV) 限制和输出设置点提供的调节或任何限制。因此，必须在过程指定的允许范围内仔细选择这两个值，以避免潜在的过程过载。

通过自整定实现 PID 时，**动态 AT 修正器**参数将影响比例增益 (Kp) 值。自整定过程的比例增益计算取决于所选择的动态修正器速度。您可以选择以下其中一个选项：

- **快速**
- **中速**
- **慢速**
- **字地址**

请参见下表中的选项描述。

⚠ 警告

PID 运行不稳定

- 必须在充分了解过程值 (PV) 极限和输出设定点值对于机器或者过程影响的条件下对其进行设置。
- 切勿超出过程值和输出设定点值的允许范围。

不遵循上述说明可能导致人员伤亡或设备损坏。

⚠ 警告

意外的设备操作

切勿将继电器输出与 PID 功能结合使用。

不遵循上述说明可能导致人员伤亡或设备损坏。

下表描述您可以定义的设置：

字段	描述	
AT 模式	授权	激活此框以启用自整定操作。 该复选框有两种使用方式，具体取决于您在 PID 功能的**常规**选项卡中是手动设置操作模式还是通过字地址： ● 如果从**常规**选项卡 (参见第 *432* 页)中将**操作模式**设置为 **PID + AT** 或 **AT**，则将激活**授权**选项且不可编辑。 ● 如果通过字地址 %MWx（%MWx = 2: **PID + AT**；%MWx = 3: **AT**）设置操作模式，则必须手动激活**授权**选项，以便能够配置自整定参数。
测量范围	授权	激活此框以启用范围测量。 **注意：** 如果禁用范围测量，则**最小值**将设置为 0，**最大值**将设置为 10000。
	最小值 最大值	根据比设置点高或低 1% 的测量范围，设置**最小值**和**最大值**。 该值可以是介于 1 和 10000 之间的立即值或者可以是存储器字 %MWx。 **注意：** **最小值**必须小于**最大值**。 **示例**：如果过程值必须大约为 35°C ± 3°C： ● 则设置点为 350。 ● ± 3°C 为 h (参见第 *424* 页) 并且应为 30。 ● 因此，1% x (**最大值 - 最小值**) = 30 ● 因此，**最大值** = 3100 并且**最小值** = 100
动态 AT 修正器	快速 中速 慢速 字地址	此参数影响 AT 过程计算的比例增益 (Kp) 值。 ● **快速**提供快速响应时间，其过冲大于中速。 ● **中速**提供中速响应时间，其过冲为中等水平。 ● **慢速**提供慢速响应时间，其过冲小于中速。 ● **字地址**提供指定字对象 %MW 中配置的响应时间。
AT 触发器	AT 触发器	此参数允许您每次在专用位（存储器位或数字量输入位）上检测到上升沿时启动 AT 过程。

计算的 Kp、Ti、Td 系数

一旦完成自整定过程，计算的 Kp、Ti 和 TdPID 系数将存储到其各自的存储器字 (%MWx) 中。

输出选项卡

简介

该选项卡用于输入 PID 输出参数。

仅当处于离线模式以及已从**常规**选项卡中选择某一操作模式后，方可访问该选项卡。

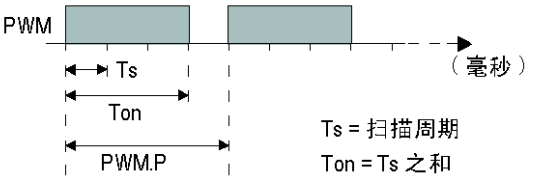
注意： 要在冷重启后保留输入值，使用字存储器 (%MW)，勿使用模拟量输出 (%QW)。

描述

下表描述您可以定义的设置：

字段	描述
动作	在此指定在过程上执行的 PID 操作的类型。可用选项有三个：**反向**、**直接**和**位地址**。 如果增大输出导致过程值测量值增大，请定义反向操作（“反向”）；反之，如果这导致过程值下降，请将 PID 设置为直接（“直接”）。 如果选择**位地址**(1)，则可以通过修改关联位（存储器位 (%Mxx) 或输入地址 (%Ix.y)）来修改操作类型。 如果操作为**直接**，则将内存位设置为 1；如果操作为**反向**，则将内存位设置为 0。
限制	指定是否要对 PID 输出施加限制。可用选项有 3 个：**启用**、**禁用**和**位地址**。 选择**启用**可将**位**设置为 1，选择**禁用**可将**位**设置为 0。 选择**位地址**则可通过修改关联位（存储器位 (%Mxx) 或输入地址 (%Ix.y)）来进行位限制管理。 设置 PID 输出的上限和下限。 **最小值**或**最大值**可为内存字 (%MWxx)、常量字 (%KWxx) 或介于 1 和 10000 之间的某个值。 **注意**：**最小值**必须小于**最大值**。
手动模式	指定是否要将 PID 更改为手动模式。可用选项有 3 个：**启用**、**禁用**和**位地址**。 如果选择**位地址**，则可以修改关联位（内存位 (%Mxx) 或输入），以便使用程序切换到手动模式（位设为 1）或自动模式（位设为 0）。 手动模式的**输出**必须包含您希望在 PID 处于手动模式 (参见第 *426* 页)时分配给模拟量输出的值。此**输出**可以是字 (%MWxx) 或 [0...10,000] 格式的直接值。
模拟量输出	指定在自整定模式时要使用的 PID 输出。 此**模拟量输出**(2) 可以是字存储器地址或模拟输出地址。

(1) 启用自整定时，自整定算法自动为控制过程确定正确的动作类型（“直接”或“反向”）。然后，必须在相关的**位地址**文本框中仅输入一个存储器位 (%Mxx)。
(2) 输入存储器地址 (%MWxx) 或模拟量输出地址 (%QWx.y)。

字段	描述
输出 PWM	选中此框可使用 PID 的 PWM 功能。 请在**周期（0.1 秒）**文本框中指定调制周期。此周期必须是介于 1 到 500 之间的值，可以是内存字 (%MWxx) 或常量字 (%KWxx)。PWM 精确度取决于 PWM 周期和扫描周期。当 PWM 比率 (%PWM.R) 有最多值时，该精确度会有所提升。例如，当扫描周围为 20 微秒且 PWM 周期为 200 微秒时，PWM.R 的值可为 0%、10%、20%、30%、40%、50%、60%、70%、80%、90%、100%。当扫描周围为 50 微秒且 PWM 周期为 200 微秒时，PWM.R 的值可为周期 PWM.P 的 0%、25%、50%、75% 和 100%。 示例： PWM.R = 75%  指定 PWM 输出位作为**输出**中的值。该值可以是内存位 (%Mxx) 或输出地址。有关 PWM 功能的详细信息，请参阅逻辑控制器的*功能库指南*。
(1) 启用自整定时，自整定算法自动为控制过程确定正确的动作类型（“直接”或“反向”）。然后，必须在相关的**位地址**文本框中仅输入一个存储器位 (%Mxx)。 **(2)** 输入存储器地址 (%MWxx) 或模拟量输出地址 (%QWx.y)。	

第16.5节
PID 编程

使用 PID 功能

本节介绍如何使用 **PID** 功能并提供其编程指南。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
描述	442
编程和配置	444
PID 状态和检测到的错误代码	445

描述

简介

比例-积分-微分 (PID) 是一个广泛用于工业控制系统的常规控制回路反馈机制（控制器）。PID 控制器使用一个涉及 3 个独立常量参数（分别以 P、I 和 D 表示的比例、积分和微分值）的算法。

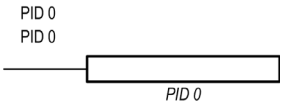
主要功能

SoMachine Basic PID 功能的主要功能如下所示：

- 模拟量输入
- 可配置测量值的线性转换
- 高位或低位可配置输入报警
- 模拟量或 PWM 输出
- 可配置输出的切断
- 可配置的直接或反向操作
- 自整定功能

示意图

这是 PID 的梯形图编辑器中的 SoMachine Basic 功能：



注意： PID 和 PID 编号之间必须有空格（例如，PID<空格>0）。

参数

与Timer 或 Counter 功能块不同，PID 中没有 SoMachine Basic 功能块。指令 [PID x] 仅支持 PID 控制回路功能，其中 x 是 PID 编号。

要配置 PID 功能，请转至**编程**窗口，单击**工具 → PID**，然后编辑 PID 属性（请参阅下表了解配置参数）。

PID 功能具有以下参数：

参数	描述	值
已使用	如果在项目的任意位置使用了 I/O，则选中该参数	True/False False（缺省值）
PID	当前PID 对象的名称	程序只能包含数量有限的 PID 功能。有关逻辑控制器中提供的 PID 对象的最大数量，请参阅“对象的最大数量”表。
符号	当前PID 对象的符号	与此 PID 对象关联的符号。有关详细信息，请参阅定义符号 *(参见 SoMachine Basic, 操作指南)*。

参数	描述	值
[...]	用于启动助手的按钮	单击可显示 **PID 助手**画面。有关详细信息，请参阅 PID 助手 (参见第 *430* 页)。
注释	注释	可以将注释与此对象进行关联。

编程和配置

简介

本节介绍了如何对 SoMachine BasicPID 控制器。

启用 PID 控制器

以下示例在位 %M 设置为 1 的情况下启用 PID 0 0 控制器循环：

梯级	指令
0	LD %M0 [PID 0]

注意： 请参阅可转换性过程 (参见第 *217* 页)以获取等效梯形图。

PID 模拟测量

该PID 功能使用模拟测量和设置点来完成 PID 纠正，并以相同的格式生成模拟指令或在数字量输出中生成 PWM。

要以全量程（最高分辨率）使用 PID，请以 [0...10,000] 格式配置 PID 控制器测量专用的模拟量输入。但是，如果使用缺省配置 [0...4095]，PID 控制器仍将正常运行。

配置扫描周期

在使用 SoMachine Basic PID 控制器时，您必须将逻辑控制器的扫描模式配置为**周期**扫描模式（**程序**选项卡，**任务 → 主任务**）。在周期扫描模式下，逻辑控制器均会以定期时间间隔开始每次扫描，以便使整个测量期间的采样率都一致。有关配置扫描模式的详细信息，请参阅 *SoMachine 基本操作指南*。

在周期扫描模式下，如果逻辑控制器的扫描时间超过用户程序定义的周期，则系统会将系统位 %S19 设置为 1。

PID 状态和检测到的错误代码

简介

SoMachine Basic PID 控制器能够将 PID 控制器和自整定过程的当前状态写入到用户定义的存储器字中。有关如何启用和配置 PID 状态存储器字的详细信息，请参阅 PID 助手 (参见第 *432* 页)的**常规**选项卡 (参见第 *430* 页)。

PID 状态存储器字可以记录以下类型的 PID 信息：

- PID 控制器的当前状态
- 自整定过程的当前状态
- PID 检测到的错误代码
- 自整定检测到的错误代码

注意： PID 状态存储器字处于只读状态。

PID 状态存储器字

PID 状态	描述
0000（十六进制）	PID 控制未处于活动状态
2000（十六进制）	PID 控制正在进行
4000（十六进制）	PID 设定点已到达

自整定状态存储器字

自整定状态	描述
0100（十六进制）	自整定阶段 1 (参见第 *424* 页) 正在进行
0200（十六进制）	自整定阶段 2 (参见第 *424* 页) 正在进行
0400（十六进制）	自整定阶段 3 (参见第 *424* 页) 正在进行
0800（十六进制）	自整定阶段 4 (参见第 *424* 页) 正在进行
1000（十六进制）	自整定阶段完成

PID 检测到的错误代码

下表描述 PID 控制期间可能会检测到的潜在错误：

检测到的错误代码	描述
8001（十六进制）	操作模式值超出范围
8002（十六进制）	线性转换的最大最小值相等
8003（十六进制）	离散量输出的上限低于下限
8004（十六进制）	设置点超出线性转换范围
8005（十六进制）	设置点小于 0 或大于 10000

检测到的错误代码	描述
8006（十六进制）	设置点超出线性转换范围
8007（十六进制）	设置点小于 0 或大于 10000
8008（十六进制）	控制动作与自整定开始时确定的动作不一致

自整定检测到的错误代码

下表记录了自整定检测到的错误消息，并介绍了可能的原因以及故障排除操作：

检测到的错误代码	描述
8009（十六进制）	已达到过程值 (PV) 限制。由于自整定是一个开放回路过程，因此过程值 (PV) 限制将用作最大允许值。
800A（十六进制）	采样周期过短，或输出设置点过低。拉长采样周期或增大自整定输出设置点值。
800B（十六进制）	Kp 为零。
800C（十六进制）	时间常量为负，因此采样周期可能过长。有关更多详细信息，请参阅自整定使用限制 (参见第 *425* 页)。
800D（十六进制）	延迟为负。
800E（十六进制）	计算 Kp 时检测到错误。自整定算法不稳定（未整合）。 这可能是由于： ● 自整定期间的过程干扰使得过程静态增益评估失真。 ● 过程值瞬时响应不够大，无法进行自整定来确定静态增益。 ● 综上所述。 检查 PID 和自整定参数并进行调整，以改善整合。同时检查有无可能影响过程值的干扰。尝试修改： ● 输出设置点 ● 采样周期 请确保自整定过程中无过程干扰。
800F 十六进制	时间常量超过延迟比率，τ/θ > 20。PID 调节可能不再稳定。有关更多详细信息，请参阅自整定使用限制 (参见第 *425* 页)。
8010（十六进制）	时间常量超过延迟比率，τ/θ < 2。PID 调节可能不再稳定。有关更多详细信息，请参阅自整定使用限制 (参见第 *425* 页)。
8011（十六进制）	已超过静态增益 Kp 的限值，Kp>10000。某些应用程序变量的测量灵敏度可能太低。范围必须重新调整在 [0...10000] 区间之内。
8012（十六进制）	已超过积分时间常量 Ti 的计算值，Ti > 20000。
8013（十六进制）	已超过微分时间常量 Td 的计算值，Td > 10000。
8014（十六进制）	无效输入变量值（超出低输出和高输出报警 (参见第 *435* 页)限定的范围）。
8015（十六进制）	过滤处理错误： ● 周期时间超出范围。 ● 过滤时间 < 10 x 周期时间。

第16.6节
PID 参数

本节包含了哪些内容？

本节包含了以下主题：

主题	页
PID 参数的作用和影响	448
PID 参数调整方法	450

PID 参数的作用和影响

简介

本节对 PID 参数的作用和影响进行了描述。

PID 控制器模型

SoMachine Basic PID 控制器实现了混合（串行-并行）PID 修正。积分和微分操作将以独立和并行方式执行。比例操作会对积分和微分操作的组合输出的产生作用。

算法

根据积分时间常量 (Ti) 的值，可以使用两种不同的算法：

- 如果 Ti ≠ 0，使用增量算法，
- 如果 Ti = 0，使用位置算法及应用于 PID 输出的 +5000 偏移。

操作的影响

比例动作用于影响过程响应速度。比例动作增加表示：

- 响应更快
- 静态误差更少
- 稳定性降低

积分动作用于消除静态误差。积分动作增加（即积分时间 Ti 缩短）表示：

- 响应更快
- 稳定性降低

微分动作是可预见的。实际上，它有考虑到偏差中的变化速度（这样就能通过在偏差增大时加快过程响应时间并在偏差减小时减慢过程响应时间来参与更改操作）。微分动作增加（即积分时间增加）代表：

- 响应更慢
- 过冲减少

注意： 从微分时间的角度考虑，Td 是用于参与微分变化的时间。Td 值太低或太高都会导致有害振荡。

必须针对各个动作在速度和稳定性之间找到适当的折中方案。

PID 控制回路的限制

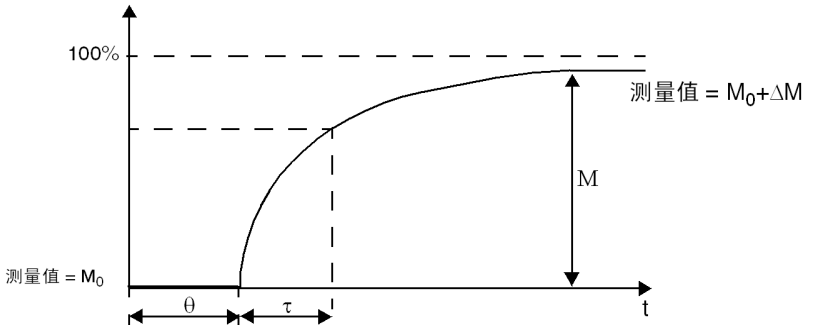
过程被吸收到具有转移功能的纯一阶延迟：

$$H(p) = K \times \frac{e^{-\theta p}}{1+\tau p}$$

其中：

τ：模型时间常量

θ：模型延迟



过程控制性能取决于比率 $\frac{\tau}{\theta}$

在以下域中可以实现适当的 PID 过程控制：$2 < \frac{\tau}{\theta} < 20$

PID 过程控制最适用于满足以下条件的过程调节：

- 对于 $\frac{\tau}{\theta} < 2$，也就是说，对于快速控制回路（低 θ）或对于具有大延迟（高 t）的过程而言，PID 过程控制并不适用。在此类情况下，应使用更复杂的算法。
- 对于 $\frac{\tau}{\theta} > 20$，使用阈值加滞后值的过程控制就足够了。

PID 参数调整方法

简介

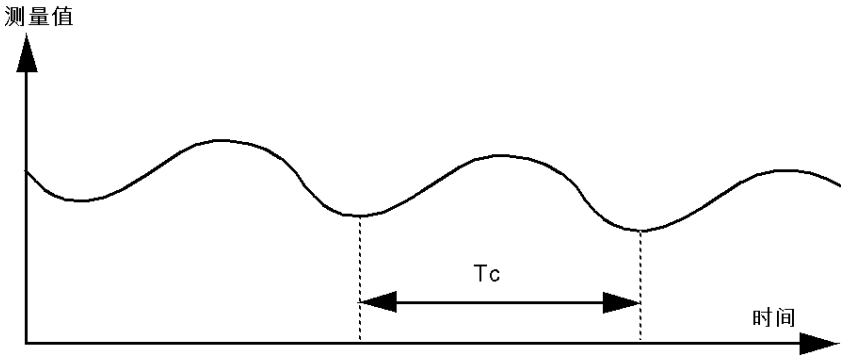
调整 PID 参数的方法有很多种。首选方法是 Ziegler 和 Nichols，它们具有 2 个变体：

- 闭合回路调整
- 开放回路调整

在实现其中任一方法之前，必须设置 PID 动作 (参见第 *439* 页)。

闭合回路调整

该原理是使用比例命令（Ti = 0、Td = 0），从而通过增大比例系数来启动过程，直到过程在将级别应用于 PID 修正器设定点后再次开始振荡为止。需要提高导致非衰减振荡的关键比例增益 (Kpc) 和振荡周期 (Tc)，以降低实现最佳调节的值。

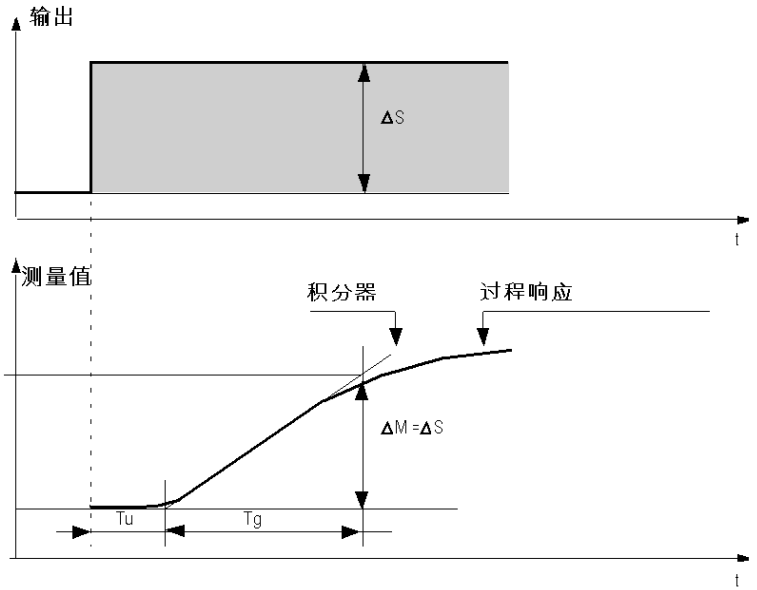


基于所使用的修正器类型（PID 或 PI），系数调整将通过以下值来执行：

修正器	Kp：比例增益	Ti：积分时间	Td：微分
PID	Kpc/1.7	Tc/2	Tc/8
PI	Kpc/2.22	0.83 x Tc	–

开放回路调整

由于调节器处于手动模式 (参见第 *426* 页)，因此您将电平施加于输出，并使过程响应的启动方式与具有纯延迟时间的积分器相同。



右手边的交点代表具有时间轴的积分器，该交点确定时间 Tu。接下来，Tg 时间定义为所控制变量（测量值）获得与调节器输出相同的变化幅度（量程的百分比）所需的时间。

基于所使用的修正器类型（PID 或 PI），系数调整将通过以下值来执行：

修正器	Kp：比例增益	Ti：积分时间	Td：微分
PID	-1.2 Tg/Tu	2 x Tu	0.5 x Tu
PI	-0.9 Tg/Tu	3.3 x Tu	–

注意： 有关参数单位的详细信息，请参阅 **PID** 选项卡 (参见第 *436* 页)。

该调整方法会提供一个非常动态的命令，这体现在设置点脉冲变化期间会发生有害过冲。在这种情况下，请降低比例增益，直到获得所需行为为止。此方法不需要任何关于过程的本质和顺序的假设。您可以将其应用于稳定的过程，就像将其应用到真实的积分过程一样。对于慢速过程（例如，玻璃行业），用户只需要响应的开端就能调节系数 Kp、Ti 和 Td。

第17章
系统对象

本章包含了哪些内容？

本章包含了以下主题：

主题	页
系统位 (%S)	454
系统字 (%SW)	463
输入通道状态 (%IWS)	482
输出通道状态 (%QWS)	484

系统位 (%S)

简介

本节介绍有关系统位功能的信息。

显示系统位属性

遵循以下步骤以显示系统位属性：

步骤	动作
1	选择**编程**窗口左侧区域的**工具**选项卡。
2	单击**系统对象** → **系统位**。 **结果**：画面中显示系统位属性。

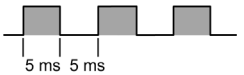
系统位属性

下表介绍了系统位的每个属性：

参数	可编辑	值	缺省值	描述
已使用	否	True/False	False	指示是否正在程序中引用系统位。
地址	否	%Si	–	显示系统位地址，其中 i 是位编号，表示系统位在存储器中的顺序位置。 如果控制器具有最多 n 个系统位， 则 i 的值以 0...n-1 形式提供。 例如，%S4 是系统位 4。
符号	是	–	–	与系统位关联的符号。 在**符号**列中双击，然后键入要与此系统位关联的符号的名称。 如果某个符号已经存在，则可以右键单击**符号**列，然后选择**搜索并替换**，在整个程序和/或程序注释中查找并替换出现此符号的地方。
注释	是	–	–	与系统位关联的注释。 双击**注释**列， 然后键入要与系统位关联的可选注释。

系统位介绍

下表介绍系统位并说明如何对它们进行控制：

系统位	功能	描述	初始状态	控制
%S0	冷启动	通常设置为 0，通过以下方式设置为 1： ● 电源恢复而数据丢失（电池故障）， ● 程序或动态数据表。 该位在第一次完全扫描的过程中设置为 1。系统在下次扫描前将其复位为 0。	0	S 或 U→S, SIM
%S4 %S5 %S6 %S7	时基：10 毫秒 时基：100 毫秒 时基：1 秒 时基：1 分钟	状态变化的速率由内部时钟测量。状态变化的速率与控制器扫描不同步。 示例：%S4 	–	S、SIM（%S4 除外）
%S9	故障预置输出	%S9 设置为 1 时： ● 对于配置为状态报警、PTO 或 FREQGEN 的输出，输出设置为 0。 ● 将故障预置值应用到物理数字量和模拟量输出（嵌入式输出、TM2/TM3 扩展模块输出和 扩展板输出）。数据映像不受 %S9 影响。数据映像会反映应用程序所应用的逻辑。只有物理输出才受影响。 ● 故障预置值的应用不考虑为具体输出配置的故障预置行为。 %S9 设置为 0 时， 数据映像值会重新应用到物理输出。 **注意：** 若控制器处于 STOPPED 状态，且配置的故障预置行为是**保持当前值**，则 %S9 的上升沿会将故障预置值应用至物理输出和数据映像值。	0	U
%S10	I/O 通讯状态	通常设置为 1（在控制面板上为 TRUE）。当检测到 I/O 通讯中断时，此位可由系统设置为 0（在控制面板上为 FALSE）。	1	S
%S11	警戒时钟溢出	通常设置为 0。如果程序执行时间（扫描时间）超过最大扫描时间（看门狗应用程序）或处理器过载（%SW75 超过 80%）， 则系统可以将该位设置为 1。 警戒时钟溢出将导致控制器状态更改为“暂停”。	0	S

S 由系统控制
U 由用户控制
U→S 由用户设置为 1，由系统复位为 0
S→U 由系统设置为 1，由用户复位为 0
SIM 在模拟器中应用

系统位	功能	描述	初始状态	控制
%S12	逻辑控制器处于正在运行状态	此位指示控制器处于正在运行状态。 系统按照如下方式设置该位： ● 当控制器状态为正在运行时，此位设置为 1； ● 当处于已停止、正在启动或任何其他状态时，此位设置为 0。	0	S, SIM
%S13	首个循环处于正在运行状态中	通常设置为 0，在控制器状态更改为“正在运行”后的第一次扫描过程中，系统将该位设置为 1。	1	S, SIM
%S15	强制输入	通常设置为 0。如果强制至少一个输入，则将被系统设为 1。	0	S, SIM
%S16	强制输出	通常设置为 0。如果强制至少一个输出，则将被系统设为 1。	0	S, SIM
%S17	最后弹出的位	通常设置为 0。它由系统根据最后弹出的位的值进行设置。 它指示最后弹出的位的值。	0	S→U, SIM
%S18	算术溢出或错误	通常设置为 0。在执行 16 位操作时发生溢出的情况下，设置为 1，即： ● 结果大于 + 32767 或小于 - 32768（单字长）， ● 结果大于 + 2147483647 或小于 - 2147483648（双字长）， ● 结果大于 + 3.402824E+38 或小于 - 3.402824E+38（浮点数）， ● 除零， ● 负数的平方根， ● BTI 或 ITB 转换无效：BCD 值超出限制。 在每个有溢出风险的操作之后必须由程序对其进行测试；如果发生溢出，程序将其复位为 0。	0	S→U, SIM
%S19	扫描周期溢出（perGrafcetiodic 扫描）	通常为 0，如果扫描周期溢出（扫描时间大于程序在配置时定义的周期或 %SW0 中编写的周期），则系统将该位设置为 1。 该位由程序复位为 0。	0	S→U
%S20	索引溢出	通常为 0，当索引对象的地址小于 0 或大于对象的最大大小时，该位设置为 1。 在每个有溢出风险的操作之后必须由程序对其进行测试；如果发生溢出，则复位为 0。	0	S→U, SIM

S 由系统控制
U 由用户控制
U→S 由用户设置为 1，由系统复位为 0
S→U 由系统设置为 1，由用户复位为 0
SIM 在模拟器中应用

系统位	功能	描述	初始状态	控制
%S21	Grafcet 初始化	通常设置为 0，通过以下方式设置为 1： ● 冷重启，%S0 = 1， ● 程序（仅在预处理程序部分），使用设置指令 (S %S21) 或设置线圈 –(S)–%S21， ● 终端 在状态 1，它导致 Grafcet 初始化。禁止活动步并激活初始步。 在 Grafcet 初始化之后，系统将其复位为 0。	0	U→S, SIM
%S22	Grafcet 复位	通常设为 0， 仅在预处理过程中可以由程序设置为 1。 在状态 1，它导致整个 Grafcet 的活动步骤被禁止。 在开始执行顺序处理时系统会将其复位为 0。	0	U→S, SIM
%S23	预设并冻结 Grafcet（列表）	通常设置为 0，仅在预处理程序模块中可以由程序设置为 1。 设置为 1 时，该位确认 Grafcet（列表）的预置。 若将该位保持为 1，则会冻结 Grafcet（列表）的执行。在开始执行顺序处理时系统会将其复位为 0，以确保 Grafcet 图形从冻结位置继续移动。	0	U→S, SIM
%S28	字符串溢出	设置为 1，表示在管理字符串时存储器对象中存在溢出。	0	S→U, SIM
%S33	以太网服务器配置的读取或写入选择（读取/更改）	通常设置为 0。 ● 设置为 0，%SW33 至 %SW38 包含使用中的以太网参数（声明的 IP 或由 BOOTP 分配的 IP 或自动分配 IP）。这些参数是在应用程序中配置的参数或者 SD 卡中后配置的参数（在后一种情况下，%SW98、%SW99 或 %SW100 不是 0）。 ● 设置为 1（如果未在使用后配置），然后由 %SW33 到 %SW38 提供新配置。 程序和系统（在冷重启时）可以将该位设置为其初始状态 0。然后，复位以太网以应用应用程序配置，不管当前配置如何。 如果在使用后配置，则此位不能设置为 1。	0	U→S
%S34	以太网自动协商	设置为 0 可允许自动协商速度和半双工或全双工模式。 设置为 1 可强制使用 %S35 和 %S36 中设置的部分特定配置。 **注意：** 在 %S34、%S35 或 %S36 状态下的修改导致以太网通道重新初始化。结果，以太网通道有可能在修改后的数秒钟不可用。	0	U
S 由系统控制 **U** 由用户控制 **U→S** 由用户设置为 1，由系统复位为 0 **S→U** 由系统设置为 1，由用户复位为 0 **SIM** 在模拟器中应用				

系统位	功能	描述	初始状态	控制
%S35	以太网半双工/全双工模式	如果 %S34 = 0（自动协商），则由系统设置此位，对用户来说是只读的。但如果 %S34 = 1，则根据用户设置的此位的值来强制使用模式。 ● 如果为半工，则设置为 0， ● 如果为双工，则设置为 1。 **注意：** 在 %S34、%S35 或 %S36 状态下的修改导致以太网通道重新初始化。结果，以太网通道有可能在修改后的数秒钟不可用。	–	U 或 S
%S36	以太网速度	如果 %S34 = 0（自动协商），则由系统设置此位，对用户来说是只读的。但如果 %S34 = 1，则根据用户设置的此位的值来强制使用模式。 ● 如果为 10 Mbps，则设置为 0 ● 如果为 100 Mbps，则设置为 1 **注意：** %S34、%S35 或 %S36 状态的修改将导致重新初始化以太网通道，因此，在修改后，以太网通道将有几分钟不可用。	–	U 或 S
%S38	允许将事件置于事件队列中	通常为 1。 ● 设置为 0，则不能将事件置于事件队列中。 ● 设置为 1， 则一检测到事件就将其置于事件队列中。 程序和系统（在冷重启时）可以将该位设置为其初始状态 1。	1	U→S
%S39	事件队列的饱和度	通常为 0。 ● 设置为 0，则报告所有事件。 ● 设置为 1，则至少丢失一个事件。 程序和系统（在冷重启时）可以将该位设置为 0。	0	U→S
%S49	输出重置	通常设置为 0，可以通过程序将该位设置为 1 或 0。 ● 设置为 0，禁用短路后自动重置输出。 ● 设置为 1，启用短路后自动重置输出。 **注意：** 冷启动后，该位复位为 0；否则，保留该位的值 系统位 %S10 用于检测程序是否有发生输出错误。随后，您可以使用系统字 %SW139 以编程方式确定发生短路或过载的群集输出。 **注意：** 将 %S49 设置为 1 时，%S10 和 %S1W139 将复位为初始值。	0	U→S

S 由系统控制
U 由用户控制
U→S 由用户设置为 1，由系统复位为 0
S→U 由系统设置为 1，由用户复位为 0
SIM 在模拟器中应用

系统位	功能	描述	初始状态	控制
%S50	使用字 %SW49 到 %SW53 更新日期和时间	通常设置为 0，可以通过程序将该位设置为 1 或 0。 ● 设置为 0，则可以读取日期和时间。 ● 设置为 1，则可以更新日期和时间但不可读取。 在 %S50 设置为 1 的同时，控制器的日期和时间不再由系统更新，并且无法由用户程序读取。 内部 RTC 控制器在 %S50 的下降沿上更新。 过程详细信息： ● 如果 %S50=0，控制器从其内部时钟定期更新系统字 %SW49-53。于是，在读取 %SW49-53 时，会向控制器提供内部日期与时间。 ● 将 %S50 设置为 1 后会停止此更新，并允许写入至 %SW49-53，无需通过上述进程覆盖。 ● 当控制器检测到 %S50 的下降沿（从 1 至 0）时，它将 %SW49-53 的值应用到其内部时钟，并重新开始更新 %SW49-53。 此 %S50 进程也是 SoMachine Basic 用于从 RTC 管理视图更新控制器时间的机制。因此，如果 SoMachine Basic 检测到 %S50 已设置为 1，则发出消息，通知 SoMachine Basic 无法读取控制器内部时钟的准确值。但是，此情况不会阻止从 RTC 管理视图更新控制器时间（如使用），%S50 将由 SoMachine Basic 复位。	0	U→S
%S51	实时时钟状态	通常设置为 0，可以通过程序将该位设置为 1 或 0。 ● 设置为 0，则日期和时间一致。 ● 设置为 1，则程序必须初始化日期和时间。 该位设置为 1 时，实时时钟数据无效。可能从未配置日期和时间、 电池电量低或者控制器修正常量无效 （从未配置过、修正的时钟值和保存的值之间存在差异或者值溢出）。 从状态 1 转换到状态 0 会强制将修正常量写入 RTC。	0	U→S, SIM
%S52	检测到 RTC 写入错误	此位由系统管理，设置为 1，指示 RTC 写入（由 %S50 请求）未执行，因为 %SW49 至 %SW53 (参见第 *455* 页) 中的值无效。 如果请求的 RTC 更改已正确应用， 则此位设置为 0。	0	S, SIM

S 由系统控制
U 由用户控制
U→S 由用户设置为 1，由系统复位为 0
S→U 由系统设置为 1，由用户复位为 0
SIM 在模拟器中应用

系统位	功能	描述	初始状态	控制
%S59	更新字 %SW59 中设置的日期和时间	通常设置为 0，可以通过程序将该位设置为 1 或 0： ● 设置为 0，则系统字 %SW59 不受管理。 ● 在此位的上升沿，日期和时间将根据在 %SW59 中设置的控制位递增或递减。	0	U
%S75	电池状态	该系统位由系统设置，用户可以读取。其指示电池状态： ● 设置为 0，则外部电池正常运转。 ● 设置为 1，则外部电池电量低，或未检测到外部电池。	0	S
%S90	备份/恢复/擦除目标	该系统位选择存储器字备份/恢复/擦除操作的目标： ● 设置为 0：非易失性存储器（缺省）。 ● 设置为 1：SD 卡。	0	U
%S91	擦除已备份的变量	将此位设置为 1 时，会擦除存储在非易失性存储器或 SD 卡中的备份变量，具体取决于 %S90。	–	U→S
%S92	备份在非易失性存储器中的 %MW 变量	如果非易失性存储器中有可用的存储器字 (%MW) 变量，则系统将此系统位设置为 1。	–	S
%S93	备份 %MW	将此位设置为 1 时，会将 %MW 变量备份在非易失性存储器或 SD 卡中，具体取决于 %S90。	–	U→S
%S94	恢复 %MW	将此位设置为 1 时，会恢复备份在非易失性存储器或 SD 卡中的数据，具体取决于 %S90。	–	U→S
%S96	备份程序正确	可以随时读取该位（通过程序或在调整时），尤其是在冷启动之后。 ● 设置为 0，则备份程序无效。 ● 设置为 1，则备份程序有效。	0	S, SIM
%S101	更改端口地址（Modbus 协议）	用于使用系统字 %SW101 (SL1) 和 %SW102 (SL2) 更改串行线路端口地址。要执行此操作，必须将 %S101 设置为 1。 ● 设置为 0，则不能更改端口地址。%SW101 和 %SW102 的值与当前端口地址匹配。 ● 设置为 1，该地址可通过更改 %SW101 (SL1) 和 %SW102 (SL2) 的值进行更改。 **注意：** 如果在 SL1 或 SL2 上定义了后配置文件，则 %S101 无法设置为 1。	0	U

S 由系统控制
U 由用户控制
U→S 由用户设置为 1，由系统复位为 0
S→U 由系统设置为 1，由用户复位为 0
SIM 在模拟器中应用

系统位	功能	描述	初始状态	控制
%S103 %S104	使用 ASCII 协议	能够在 SL1 (%S103) 或 SL2 (%S104) 上使用 ASCII 协议。ASCII 协议通过使用系统字 %SW103 和 %SW105（对于 SL1）以及系统字 %SW104 和 %SW106（对于 SL2）进行配置。 ● 设置为 0，使用的协议为 SoMachine Basic 中配置的协议或者在后配置 (参见第 *53* 页)中指定的协议。 ● 设置为 1，则在 SL1 (%S103) 或 SL2 (%S104) 上使用 ASCII 协议。在此情况下，必须事先为 SL1 配置系统字 %SW103、%SW105 和 %SW121，必须事先为 SL2 配置 %SW104、%SW106 和 %SW122。 在 %S103 或 %S104 的上升沿后，将会考虑这些 %SW 的每次更改。 **注意：** %S103 或 %S104 的上升或下降沿会取消正在进行的交换（EXCH 指令） **注意：** 如将 %S103 或 %S104 设置为 0，就会使用 SoMachine Basic 参数重新配置串行线路。 **注意：** 如果相应的串行线路上配置有 Modbus Serial Line IOScanner，则忽略 %S103 和 %S104。	0	U
%S106	I/O 总线行为	缺省值为 0，意味着扩展模块上的总线通讯错误使 I/O 扩展总线交换停止。 将这个位设置为 1，以指定控制器继续执行 I/O 扩展总线交换。 **注意：** 出现总线通讯错误时， %SW120 的位 n 设置为 1， 其中 n 是扩展模块的数量，并且 %SW118 位 14 设置为 0。 有关总线错误处理的更多信息，请参阅 I/O 配置一般说明。	0	U/S
S 由系统控制 **U** 由用户控制 **U→S** 由用户设置为 1，由系统复位为 0 **S→U** 由系统设置为 1，由用户复位为 0 **SIM** 在模拟器中应用				

系统位	功能	描述	初始状态	控制
%S107	I/O 总线重启	缺省值是 0。由系统复位为 0。 将这个位设置为 1 以强制重启 I/O 扩展总线。若检测到这个位的上升沿，逻辑控制器会重新配置并重启 I/O 扩展总线，但前提是满足以下条件： ● %S106 设置为 0（即，I/O 交换停止） ● %SW118 位 14 设置为 0（I/O 总线存在错误） ● %SW120 的至少一个位设置为 1（识别存在总线通讯错误的模块） 有关总线错误处理的更多信息，请参阅 I/O 配置一般说明。	0	U/S
%S110	IOScanner 复位 SL1	设置为 1 时，复位串行线路 1 上的 Modbus Serial IOScanner。	0	U/S
%S111	IOScanner 复位 SL2	设置为 1 时，复位串行线路 2 上的 Modbus Serial IOScanner。	0	U/S
%S112	IOScanner 复位 ETH1	设置为 1 时，复位以太网上的 Modbus TCP IOScanner。	0	U/S
%S113	IOScanner 暂停 SL1	设置为 1 时，暂停串行线路 1 上的 Modbus Serial IOScanner。	0	U/S
%S114	IOScanner 暂停 SL2	设置为 1 时，暂停串行线路 2 上的 Modbus Serial IOScanner。	0	U/S
%S115	IOScanner 暂停 ETH1	设置为 1 时，暂停以太网上的 Modbus TCP IOScanner。	0	U/S
%S119	检出本地 I/O 错误	通常设置为 1。在逻辑控制器上检测到 I/O 通讯中断时可以将该位设置为 0。%SW118 确定通讯中断的性质。通讯中断消失时复位为 1。	1	S
S 由系统控制 **U** 由用户控制 **U→S** 由用户设置为 1，由系统复位为 0 **S→U** 由系统设置为 1，由用户复位为 0 **SIM** 在模拟器中应用				

系统字 (%SW)

简介

本节介绍有关系统字功能的信息。

显示系统字属性

遵循以下步骤以显示系统字属性：

步骤	动作
1	选择**编程**窗口左侧区域的**工具**选项卡。
2	单击**系统对象 → 系统字**。 **结果**：画面中显示系统字属性。

系统字属性

下表介绍了系统字的每个属性：

参数	可编辑	值	缺省值	描述
已使用	否	True/False	False	指示是否正在程序中引用系统字。
地址	否	%SWi	–	显示系统字地址，其中 i 是字编号，表示系统字在存储器中的顺序位置。 如果控制器具有最多 n 个系统字，则 i 的值以 0...n-1 形式提供。 例如，%SW50 是系统字 50。
符号	是	–	–	与系统字关联的符号。 在**符号**列中双击，然后键入要与此系统字关联的符号的名称。 如果某个符号已经存在，则可以右键单击**符号**列，然后选择**搜索并替换**，在整个程序和/或程序注释中查找并替换出现此符号的地方。
注释	是	–	–	与系统字关联的注释。 双击**注释**列，然后键入要与系统字关联的可选注释。

系统字介绍

下表介绍系统字并说明如何对它们进行控制：

系统字	功能	描述	控制
%SW0	控制器扫描周期（主任务设置为定期扫描模式）	修改主任务属性 或动态数据表中定义的控制器扫描周期（1...150 毫秒）。	U, SIM
%SW1	周期性任务周期	修改周期性任务的循环时间 [1...255 毫秒]，但不会丢失在周期性任务属性窗口中指定的**周期**值。 允许您在下列情况下恢复在定期任务属性窗口中保存的**周期**值： ● 在冷启动的情况下，或者 ● 如果写入 %SW1 中的值超出 [1...255] 的范围。 可以在每个循环结束时在程序或动态数据表中修改 %SW1 值，而无需停止程序。在程序运行时可以正确观察循环时间。	U, SIM
%SW6	控制器状态 %MW60012	控制器状态： 0 = 空 2 = 已停止 3 = 正在运行 4 = 已暂停 5 = POWERLESS	S, SIM
S 由系统控制 **U** 由用户控制 **SIM** 在模拟器中应用			

系统字	功能	描述	控制
%SW7	控制器状态	● 位 [0]：正在进行备份/恢复： ❍ 如果程序的备份/恢复正在进行中，则设置为 1， ❍ 如果程序的备份/恢复已完成或者已禁用，则设置为 0。 ● 位 [1]：控制器的配置正确： ❍ 如果配置正确，则设置为 1。 ● 位 [2]：SD 卡的状态位： ❍ 如果存在 SD 卡，则设置为 1。 ● 位 [3]：SD 卡的状态位： ❍ 如果正在访问 SD 卡，则设置为 1。 ● 位 [4]：应用程序内存状态： ❍ 如果 RAM 内存中的应用程序与闪存中的应用程序不同，则设置为 1。 ● 位 [5]：SD 卡的状态位： ❍ 如果 SD 卡存在错误，则设置为 1。 ● 位 [6]：未使用（状态 0） ● 位 [7]：保留的控制器： ❍ 当控制器与 SoMachine Basic 之间处于已连接模式时，设置为 1。 ● 位 [8]：应用程序处于写入模式： ❍ 如果应用程序受保护，则设置为 1。在这种情况下，克隆操作不会复制应用程序（请参见克隆管理 (参见第 *186* 页)） ● 位 [9]：未使用（状态 0） ● 位 [10]：将第二个串行端口安装为扩展板（仅一体型）： ❍ 0 = 无串行扩展板 ❍ 1 = 已安装串行扩展板 ● 位 [11]：第二个串行端口的类型： ❍ 设置为 1 = EIA RS-485 ● 位 [12]：内部内存中应用程序的有效性： ❍ 如果该应用程序有效，则设置为 1。 ● 位 [14]：RAM 内存中应用程序的有效性： ❍ 如果该应用程序有效，则设置为 1。 ● 位 [15]：准备执行： ❍ 如果准备执行，则设置为 1。	S, SIM
%SW11	软件警戒时钟值	包含警戒时钟的最大值。该值（10...500 毫秒）由配置定义。	U, SIM
%SW13	引导版本 Vxx.yy	例如，如果 %SW13 = 000E hex： ● 8 MSB=00（十六进制），则 xx=0（十进制） ● 8 LSB=0E（十六进制），那么 yy=14（十进制） 因此，引载加载程序版本是 0.14，以十进制显示为 14。	U, SIM
S 由系统控制 U 由用户控制 SIM 在模拟器中应用			

系统字	功能	描述	控制
%SW14	商用版本，Vxx.yy	例如，如果 %SW14 = 0232： ● 8 MSB=02（十六进制），则 xx=2（十进制） ● 8 LSB=32（十六进制），则 yy=50（十进制） 因此，商业版本是 2.50，以十进制显示为 250。	S, SIM
%SW15 - %SW16	固件版本，aa.bb.cc.dd	例如，如果 %SW15 = 0003 hex： ● 8 MSB=00（十六进制），则 aa=00（十进制） ● 8 LSB=03（十六进制），则 bb=03（十进制） 例如，如果 %SW16 = 0B16 hex： ● 8 MSB=0B（十六进制），则 cc=11（十进制） ● 8 LSB=16（十六进制），则 dd=22（十进制） 因此，固件版本是 0.3.11.22，以十进制显示为 00031122。	S, SIM
%SW17	浮点运算的缺省状态	在浮点算术运算中检测到错误时，会将位 %S18 设置为 1，并根据以下编码更新 %SW17 的缺省状态： ● 位 [0]：运算无效，结果不是数字 (NaN) ● 位 [1]：保留 ● 位 [2]：以 0 为除数，结果无效（负无穷大或正无穷大） ● 位[3]：结果的绝对值大于 +3.402824e+38，结果无效（负无穷大或正无穷大）	S 与 U、SIM
%SW18- %SW19	100 毫秒绝对定时器计数器	计数器使用以下 2 个字进行工作： ● %SW18 表示最低有效字， ● %SW19 表示最高有效字。 双字 (%SW18 - %SW19) 每 100 毫秒从 0 增至 2^31，作为计数器模数 2^31。 在 INIT 阶段和 %S0 复位时，此双字也会复位。	S 与 U、SIM
%SW30	上次扫描时间（主任务）	指示上次控制器扫描循环的执行时间（单位：毫秒）。 **注意：** 此时间对应于主任务扫描循环的开始（获得输入）和结束（更新输出）之间的耗时。如果扫描时间为 2.250 毫秒，则 %SW30 为 2 并且 %SW70 为 250。	S
%SW31	最大扫描时间（主任务）	指示自上次冷启动以来最长的控制器扫描循环的执行时间（单位：毫秒）。 **注意：** ● 此时间对应于扫描循环的开始（获得输入）和结束（更新输出）之间的耗时。如果最大扫描时间为 2.25 毫秒，则 %SW31 为 2 并且 %SW71 为 250。 ● 要确保在选择锁定输入选项时正确地检测脉冲信号，脉宽 (T_{ON}) 和周期 (P) 必须满足以下 2 个要求： ❍ $T_{ON} \geq 1$ 毫秒 ❍ 输入信号周期 (P) 必须遵守 Nyquist-Shannon 采样规则，即输入信号周期 (P) 必须至少为最大程序扫描时间 (%SW31) 的两倍： $P \geq 2 \times$ %SW31。 **注意：**如果不满足此条件，则可能检测不到某些脉冲。	S
S 由系统控制 **U** 由用户控制 **SIM** 在模拟器中应用			

系统字	功能	描述	控制
%SW32	最小扫描时间（主任务）	指示自上次冷启动以来最短的控制器扫描循环的执行时间（单位：毫秒）。 **注意：** 此时间对应于扫描循环的开始（获得输入）和结束（更新输出）之间的耗时。如果最小扫描时间为 2.250 毫秒，则 %SW32 将为 2，并且 %SW72 将为 250。	S
%SW33 %SW34 %SW35 %SW36 %SW37 %SW38	用于读取/写入以太网服务器配置的 IP 地址	IP 设置可以修改。读取或写入选择将通过系统位 %S33 来执行。 系统字 %SW34...%SW38 包含以下以太网参数： ● IP 地址：%SW33 和 %SW34 对于 IP 地址 AA.BB.CC.DD：%SW33 = CC.DD 且 %SW34 = AA.BB ● 子网掩码：%SW35 和 %SW36 对于子网掩码 AA.BB.CC.DD：%SW35 = CC.DD 且 %SW36 = AA.BB ● 网关地址：%SW37 和 %SW38 对于网关地址 AA.BB.CC.DD：%SW37 = CC.DD 且 %SW38 = AA.BB	U
%SW39	周期性平均时间	以微秒为单位指示定期任务的平均执行时间（最后 5 次）	–
%SW40	事件 0 平均时间	以微秒为单位指示输入 %I0.2 的相关事件任务的平均执行时间（最后 5 次）	–
%SW41	事件 1 平均时间	以微秒为单位指示输入 %I0.3 的相关事件任务的平均执行时间（最后 5 次）	–
%SW42	事件 2 平均时间	以微秒为单位指示输入 %I0.4 的相关事件任务的平均执行时间（最后 5 次）	–
%SW43	事件 3 平均时间	以微秒为单位指示输入 %I0.5 的相关事件任务的平均执行时间（最后 5 次）	–
%SW44	事件 4 平均时间	以微秒为单位指示 HSC0 或 HSC2 的阈值 0 相关事件任务的平均执行时间（最后 5 次）	–
%SW45	事件 5 平均时间	以微秒为单位指示 HSC0 或 HSC2 的阈值 1 相关事件任务的平均执行时间（最后 5 次）	–
%SW46	事件 6 平均时间	以微秒为单位指示 HSC1 或 HSC3 的阈值 0 相关事件任务的平均执行时间（最后 5 次）	–
%SW47	事件 7 平均时间	以微秒为单位指示 HSC1 或 HSC3 的阈值 1 相关事件任务的平均执行时间（最后 5 次）	–
%SW48	事件数	指示自上次冷启动以来执行的事件数量。 （计算除循环事件以外的所有事件。） **注意：** 设置为 0（在应用程序加载和冷启动之后）， 并在每次执行事件时递增。	S, SIM
S 由系统控制 **U** 由用户控制 **SIM** 在模拟器中应用			

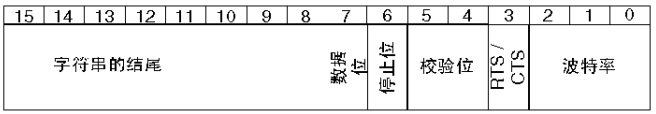
系统字	功能	描述		控制
%SW49 %SW50 %SW51 %SW52 %SW53	实时时钟 (RTC)	RTC 功能：包含当前日期和时间值的字（BCD 格式）：		S 和 U, SIM
		%SW49	星期的第 xN 天（N=1 为星期一）	
		%SW50	00SS 秒	
		%SW51	HHMM：小时和分钟	
		%SW52	MMDD：月份和日期	
		%SW53	CCYY：世纪和年	
		将系统位 %S50 设置为 1，以便能够使用系统字 %SW49 将 RTC 值更新为 %SW53。在 %S50 的下降沿，利用这些字中写入的值更新内部 RTC 控制器。更多详细信息，请参见系统位 %S50 (参见第 *455* 页)。		
%SW54 %SW55 %SW56 %SW57	上次停止的日期和时间	包含上次断电或控制器停止时的日期和时间的系统字（BCD 格式）：		S, SIM
		%SW54	SS 秒	
		%SW55	HHMM：小时和分钟	
		%SW56	MMDD：月份和日期	
		%SW57	CCYY：世纪和年	
%SW58	上次停止的代码	显示表明上停止原因的代码：		S, SIM
		0	未知原因或初始值（下载应用程序之后）	
		1	运行/停止输入沿	
		2	在检测到的软件错误时停止（控制器进入已暂停状态）	
		3	停止命令（SoMachine Basic 在线按钮）	
		4	断电	
		5	在检测到硬件错误处停止	
		6	在冷启动时初始化	
		7	在“停止”状态下启动	
		8	电池电量低	
		9	控制器无法运行	
		上次停止的原因按如下顺序划分优先级（即，当电源重置后控制器处于 STOPPED 状态时）： 1、7、4、8、2		
S 由系统控制 **U** 由用户控制 **SIM** 在模拟器中应用				

系统字	功能	描述			控制
%SW59	调整当前日期和时间	包含用于递增或递减当前日期和时间的 2 个 8 位组。 先设置 %SW59 的适当位，然后将 %S59 从 0 设置为 1。在 %S59 的上升沿设置该位时执行此操作。			U
		递增	**递减**	**参数**	
		位 0	位 8	星期几	未使用
		位 1	位 9	秒	
		位 2	位 10	分	
		位 3	位 11	时	
		位 4	位 12	日	
		位 5	位 13	月	
		位 6	位 14	年	
		位 7	位 15	世纪	未使用
%SW60	RTC 校正	RTC 校正值。 仅适用于 M200 逻辑控制器。缺省值为 0。 位 [7]：校正方向，0 为正，1 为负 位 [0] 到 [6]：校正值，在 0 到 30 的范围内 如果校正值不为 0，RTC 将在每周日 00:00:30 自动校正至该值。 **注意：** 可以在 PLC 联网的情况下，通过用户应用或动态数据表修改 RTC 校正值。在断电恢复的情况下，这个值会保存在闪存中。			U
%SW62	以太网错误检测	指示错误代码： **0** 未检测到错误 **1** IP 重复： M200 逻辑控制器配有其缺省 IP 地址（通过 MAC 地址生成） **2** DHCP 处于进程中 **3** BOOTP 处于进程中 **4** 无效参数：端口已禁用 **5** 固定 IP 地址：正在初始化 **6** 以太网链接已断开			S
S 由系统控制 **U** 由用户控制 **SIM** 在模拟器中应用					

系统字	功能	描述	控制
%SW63	EXCH1 功能块错误代码	EXCH1 错误代码： 0 - 操作已成功 1 - 待传输的字节数量超过限制 (> 255) 2 - 传输表不足 3 - 字表不足 4 - 接收表溢出 5 - 已超时 6 - 传输 7 - 表中存在错误命令 8 - 所选端口未配置/不可用 9 - 接收错误：该错误代码表示接收帧错误或受损。这可能是由于物理参数配置错误（例如校验位、数据位、波特率等），或者是由于不可靠的物理连接导致信号减弱而引起的。 10 - 接收时不能使用 %KW 11 - 传输偏移大于传输表 12 - 接收偏移大于接收表 13 - 控制器停止了 EXCH 处理	S
%SW64	EXCH2 功能块错误代码	EXCH2 错误代码：请参见 %SW63。	S
%SW65	EXCH3 功能块错误代码	1-4、6-13：请参见 %SW63。（请注意，错误代码 5 无效，由下文介绍的以太网特定的错误代码 109 和 122 取代。） 以下为以太网特定的错误代码： 101 - IP 地址错误 102 - 无 TCP 连接 103 - 无可用的套接字（所有连接通道均忙碌） 104 - 网络已关闭 105 - 无法访问网络 106 - 复位时网络丢失连接 107 - 对等设备中止连接 108 - 对等设备复位连接 109 - 连接已超时 110 - 连接尝试被拒绝 111 - 主机已关闭 120 - 索引不正确（配置表中未包含远程设备的索引） 121 - 系统错误（MAC、芯片） 122 - 发送数据后接收进程超时 123 - 正在初始化以太网	S
%SW67	控制器的功能和类型	包含逻辑控制器代码 ID。有关详细信息，请参阅M100/M200 可编程控制器代码 ID 表 (参见第 *481* 页)。	S, SIM
S 由系统控制 **U** 由用户控制 **SIM** 在模拟器中应用			

系统字	功能	描述	控制
%SW70	扫描时间微秒精度	指示上次控制器扫描循环的执行时间（单位：微秒）。 **注意：** 此时间对应于主任务扫描循环的开始（获得输入）和结束（更新输出）之间的耗时。如果扫描时间为 2.250 毫秒，则 %SW30 将为 2，并且 %SW70 将为 250。	–
%SW71	最大扫描时间微秒精度	指示自上次冷启动以来最长的控制器扫描循环的执行时间（单位：毫秒）。 **注意：** 此时间对应于扫描循环的开始（获得输入）和结束（更新输出）之间的耗时。如果扫描时间为 2.250 毫秒，则 %SW31 将为 2，并且 %SW71 将为 250。	–
%SW72	最小扫描时间微秒精度	指示自上次冷启动以来最短的控制器扫描循环的执行时间（单位：毫秒）。 **注意：** 此时间对应于扫描循环的开始（获得输入）和结束（更新输出）之间的耗时。如果扫描时间为 2.250 毫秒，则 %SW32 将为 2，并且 %SW72 将为 250。	–
%SW75	处理器负载	指示处理负载百分比。 处理负载是按照用于处理程序任务的总处理时间的百分比来定义的（该值为平均值，每秒计算一次）。如果处理负载在两个连续时间段内均大于 80%，控制器就会进入“HALTED”状态。	S
%SW76 至 %SW79	减计数器 1-4	这 4 个字用作 1 毫秒定时器。如果这 4 个字的值为正，则它们的值每毫秒均由系统递减。这提供了 4 个以毫秒递减计数的减计数器，相当于运算范围 1 毫秒到 32767 毫秒。将位 15 设置为 1 可以停止递减。	S 与 U、SIM
%SW89	扩展板插槽 1 版本	指示插槽 1 中扩展板的固件版本。	S
%SW90	扩展板插槽 2 版本	指示插槽 2 中扩展板的固件版本。	S
%SW91	TM3 插槽 1 版本	指示插槽 1 中 TM3 扩展模块的固件版本。	S
%SW92	TM3 插槽 2 版本	指示插槽 2 中 TM3 扩展模块的固件版本。	S
%SW93	TM3 插槽 3 版本	指示插槽 3 中 TM3 扩展模块的固件版本。	S
%SW94 %SW95	应用程序签名 %MW60028- %MW60034	如果应用程序改变（配置或编程数据方面），则签名（所有校验码的总和）也改变。 如果 %SW94 = 91F3（十六进制），则应用程序签名为 91F3（十六进制）。	S, SIM
S 由系统控制 **U** 由用户控制 **SIM** 在模拟器中应用			

系统字	功能	描述	控制
%SW96	程序和 %MW 保存/恢复功能的诊断	● 位 [1]：此位由固件设置，表明保存完成的时间： ❍ 如果备份已完成，则设置为 1。 ❍ 如果提出了新的备份请求，则设置为 0。 ● 位 [2]：检测到备份错误，请参阅位 8、9、10、12 和 14 获取详细信息： ❍ 如果检测到错误，则设置为 1。 ❍ 如果提出了新的备份请求，则设置为 0。 ● 位 [6]：如果控制器的 RAM 内存中包含有效的应用程序，则设置为 1。 ● 位 [10]：内部 RAM 与闪存之间存在差异（1 = 是）。 ❍ 如果存在差异，则设置为 1 ● 位 [12]：指示是否发生恢复错误： ❍ 如果检测到错误，则设置为 1。 ● 位 [14]：表示是否发生了闪存写错误： ❍ 如果检测到错误，则设置为 1。	S, SIM
%SW97	TM3 插槽 4 版本	指示插槽 4 中 TM3 扩展模块的固件版本。	S
%SW98	后配置状态（串行线路 1）	对参数应用后配置后，以下位设置为 1： ● 位 [0]：硬件选项（RS-485 或 RS-232） ● 位 [1]：波特率 ● 位 [2]：奇偶校验 ● 位 [3]：数据大小 ● 位 [4]：停止位数 ● 位 [5]：Modbus 地址 ● 位 [6]：极化（如果在端口中可用）	S
%SW99	后配置状态（串行线路 2）	对参数应用后配置后，以下位设置为 1： ● 位 [0]：硬件选项 (RS-485) ● 位 [1]：波特率 ● 位 [2]：奇偶校验 ● 位 [3]：数据大小 ● 位 [4]：停止位数 ● 位 [5]：Modbus 地址 ● 位 [6]：极化（如果在端口中可用）	S
%SW100	后配置状态（以太网）	对参数应用后配置后，以下位设置为 1： ● 位 [0]：IP 模式（固定、DHCP 或 BOOTP） ● 位 [1]：IP 地址 ● 位 [2]：子网掩码 ● 位 [3]：缺省网关 ● 位 [4]：设备名称 **注意：** 后配置的优先级高于应用程序所提供的配置。如果 M100/M200 逻辑控制器具有后配置，则不考虑应用程序配置。	S
S 由系统控制 **U** 由用户控制 **SIM** 在模拟器中应用			

系统字	功能	描述	控制
%SW101 %SW102	Modbus 地址端口的值	当位 %S101 设置为 1 时，可以更改 SL1 或 SL2 的 Modbus 地址。 SL1 的地址为 %SW101。SL2 的地址为 %SW102。 **注意：** 将新地址写入 %SW101 或 %SW102 之后，系统会立即应用更新。	S
%SW103 %SW104	使用 ASCII 协议的配置	当位 %S103 (SL1) 或 %S104 (SL2) 设置为 1 时，使用 ASCII 协议。必须根据以下元素设置系统字 %SW103 (SL1) 或 %SW104 (SL2)：  ● 波特率： ❍ 000：1200 波特， ❍ 001：2400 波特， ❍ 010：4800 波特， ❍ 011：9600 波特， ❍ 100：19200 波特， ❍ 101：38400 波特。 ❍ 110：57600 波特。 ❍ 111：115200 波特。 ● RTS/CTS： ❍ 0：禁用， ❍ 1：启用。 ● 奇偶校验： ❍ 00：无， ❍ 10：奇， ❍ 11：偶。 ● 停止位： ❍ 0：1 个停止位， ❍ 1：2 个停止位。 ● 数据位： ❍ 0：7 个数据位， ❍ 1：8 个数据位。	S, U
S 由系统控制 **U** 由用户控制 **SIM** 在模拟器中应用			

系统字	功能	描述	控制
%SW105 %SW106	使用 ASCII 协议的配置	当位 %S103 (SL1) 或 %S104 (SL2) 设置为 1 时，使用 ASCII 协议。必须根据以下元素设置系统字 %SW105 (SL1) 或 %SW106 (SL2)： 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 超时帧 （以毫秒计） 超时响应 以 100 毫秒的倍数计	S, U
%SW107 %SW108 %SW109	MAC 地址	指示控制器 MAC 地址（仅以太网通道中会显示）。 对于 MAC 地址 AA:BB:CC:DD:EE:FF: ● %SW107=AA:BB ● %SW108=CC:DD ● %SW107=EE:FF	S
%SW114	启用计划块	由程序启用或禁用计划块的操作： ● 位 [0]：启用/禁用计划块编号 0 ❍ 设置为 0： 已禁用 ❍ 设置为 1： 已启用 ● ... ● 位 [15]：启用/禁用计划块编号 15 ❍ 设置为 0： 已禁用 ❍ 设置为 1： 已启用 最初启用所有的计划块。 缺省值为 FFFF（十六进制）。	S 与 U、SIM
%SW115 %SW116 %SW117	分别为控制器序列号部分 1、2 和 3（BCD 格式）	允许获取控制器的序列号。 包含序列号 8A160400008 的示例： ● %SW115 ： 16#0008 ● %SW116 ： 16#6040 ● %SW117 ： 16#0001	S
S 由系统控制 **U** 由用户控制 **SIM** 在模拟器中应用			

系统字	功能	描述	控制
%SW118	逻辑控制器状态字	指示逻辑控制器的状况。 对于工作正常的控制器，此字的值为 FFFF（十六进制）。 ● 位 [9]： ❍ 设置为 0： 检出外部错误，或通信中断，例如重复 IP 地址. ❍ 设置为 1：未检测到错误。 ● 位 [10]： ❍ 设置为 0： 内部配置无效；联系 Schneider Electric 客户服务。 ❍ 设置为 1：未检测到错误。 ● 位 [13]： ❍ 设置为 0： 检出配置错误（当逻辑控制器尝试启动 I/O 扩展总线时，按 I/O 扩展总线配置所定义的强制性模块不存在或者无法工作）。此时，I/O 总线不会启动。 ❍ 设置为 1：未检测到错误。 ● 位 [14]： ❍ 设置为 0：在 I/O这是一种无论 I/O 模块被定义为强制模块或可选模块都会在启动时出现的状况。 ❍ 设置为 1：未检测到错误。 ● 位 [15]: ❍ 设置为 0： 检测到扩展板错误（配置或运行时操作）。 ❍ 设置为 1：未检测到错误。 有关总线错误处理的更多信息，请参阅 I/O 配置一般说明。 **注意：** 此字的其他位均设置为 1 并且是保留位。	S, SIM
%SW119	可选模块功能配置	在此配置下，每个扩展模块可使用一个位： ● 位 [0]：为逻辑控制器预留 ● 位 n： 模块 n ❍ 设置为 1： 模块在配置中被标记为可选模块。 ❍ 设置为 0： 模块在配置中未被标记为可选模块。	S, SIM
%SW120	扩展 I/O 模块状态	在此配置下，每个扩展模块可使用一个位。 位 0 = 为逻辑控制器保留 当逻辑控制器尝试启动 I/O 总线时，位 n： ● 0 = 未检测到错误 ● 1 = 检出错误，或模块不存在。除非 %SW119 中对应的为设置为 TRUE（表明该模块被标记为可选模块），否则 I/O 扩展总线不会启动。 当 I/O 总线启动时，位 n： ● 0 = 未检测到错误 ● 1 = 检出 I/O 扩展模块存在错误（无论此模块是否被标记为可选模块）。	S, SIM
%SW121 %SW122	使用 ASCII 协议的配置	当位 %S103 (SL1) 或 %S104 (SL2) 设置为 1 时，使用 ASCII 协议。您可以更改 SL1 或 SL2 的 ASCII 帧大小。SL1 的 ASCII 帧大小为 %SW121，SL2 的 ASCII 帧大小为 %SW122。	U
S 由系统控制 **U** 由用户控制 **SIM** 在模拟器中应用			

系统字	功能	描述	控制
%SW128	扩展板 1 状态	指示扩展板的状态代码： ● LSB：表明 I/O 通道 1 的状态 ● MSB：表明 I/O 通道 2 的状态 一般状态： ● 0x80：扩展板不存在，且未在 SoMachine Basic 中进行配置。 ● 0x81：模块存在，但未配置。 ● 0x82：扩展板存在内部通讯错误。 ● 0x83：扩展板存在内部通讯错误。 ● 0x84：检测到的扩展板与配置不同。 ● 0x85：未检测到已配置的扩展板。 输入通道运行状态： ● 0x00：正常。 ● 0x01：正在转换。 ● 0x02：初始化。 ● 0x03：检测到输入操作设置错误，或是模块无输入。 ● 0x04：保留。 ● 0x05：检测到接线错误（超出上限范围）。 ● 0x06：检测到接线错误（超出下限范围）。 ● 0x07：检测到闪存错误。 ● 其他：保留。 输出通道运行状态： ● 0x00：正常。 ● 0x01：保留。 ● 0x02：初始化。 ● 0x03：检测到输出操作设置错误，或是模块无输出。 ● 0x04：保留。 ● 0x05：保留。 ● 0x06：保留。 ● 0x07：检测到闪存错误。 ● 其他：保留。	S, SIM
%SW129	扩展板 2 状态		
%SW130	事件执行时间	以微秒为单位指示事件输入 %I0.2 的最后一次执行时间。	S
%SW131	事件执行时间	以微秒为单位指示事件输入 %I0.3 的最后一次执行时间。	S
%SW132	事件执行时间	以微秒为单位指示事件输入 %I0.4 的最后一次执行时间。	S
%SW133	事件执行时间	以微秒为单位指示事件输入 %I0.5 的最后一次执行时间。	S
%SW134	事件执行时间	以微秒为单位指示 HSC0 或 HSC2 的阈值 0 相关事件任务的最后一次执行时间。	S
%SW135	事件执行时间	以微秒为单位指示 HSC0 或 HSC2 的阈值 1 相关事件任务的最后一次执行时间。	S
S 由系统控制 **U** 由用户控制 **SIM** 在模拟器中应用			

系统字	功能	描述	控制
%SW136	事件执行时间	以微秒为单位指示 HSC1 或 HSC3 的阈值 0 相关事件任务的最后一次执行时间。	S
%SW137	事件执行时间	以微秒为单位指示 HSC1 或 HSC3 的阈值 1 相关事件任务的最后一次执行时间。	S
%SW138	周期性任务执行时间	以微秒为单位指示周期性任务的最后一次执行时间。	S
%SW139	内置数字量输出保护	指示输出块的保护错误状态： 位 0 = 1 - Q0 - Q3 保护错误 - 块 0 位 1 = 1 - Q4 - Q7 保护错误 - 块 1 位 2 = 1 - Q8 - Q11 保护错误 - 块 2 位 3 = 1 - Q12 - Q15 保护错误 - 块 3 **注意：** 漏极输出不使用 %SW139。	S
%SW140	上一个控制器错误代码 1	最近写入到 PlcLog.csv 的错误：AABBCCCCDD： %SW142 = AABB 十六进制 %SW141 = CCCC 十六进制 %SW140 = 00DD 十六进制 其中： ● AA = 错误级别 ● BB = 错误的发生环境 ● CCCC = 错误代码 ● DD = 错误优先级（仅内部使用）	S
%SW141	上一个控制器错误代码 2		
%SW142	上一个控制器错误代码 3		
%SW143	PlcLog.csv 中的条目数	PlcLog.csv 中包含的错误代码数。	S
%SW147	SD 卡操作结果	如果 %SW90 设置为 1，指示的是在保存了存储器字之后的 SD 卡操作结果。 错误代码有： ● 0：没有错误。 ● 1：正在运行。 ● 10：弹出 SD 卡。 ● 11：未检测到 SD 卡。 ● 12：SD 卡受到写保护 ● 13：SD 卡已满。 ● 21：无效的存储器字的数量。 ● 22：没有要保存的存储器字。 ● 30：CSV 文件中的某行无效。 ● 31：CSV 文件中的某行过长。 ● 32：CSV 文件的格式无效。 ● 40：创建 CSV 文件时出错。 ● 50：内部系统错误。 ● 51：打开 CSV 文件时出错。	S
S 由系统控制 U 由用户控制 SIM 在模拟器中应用			

系统字	功能	描述	控制
%SW148	保持型变量的数量	● 如果 %S90 设置为 0，最多能够保存 2000 个存储器字（%MW50 至 %MW2049）。 ● 如果 %S90 设置为 1，能够保存 %MW0 中的所有存储器字。 有关详细信息，请参阅根据用户请求保存的保持型变量 (参见第 *47* 页)。	U
%SW149	事件执行时间	以毫秒为单位指出与输入 %I0.2 相关的事件任务的最后一次执行时间。	S
%SW150	事件执行时间	以毫秒为单位指出与输入 %I0.3 相关的事件任务的最后一次执行时间。	S
%SW151	事件执行时间	以毫秒为单位指出与输入 %I0.4 相关的事件任务的最后一次执行时间。	S
%SW152	事件执行时间	以毫秒为单位指出与输入 %I0.5 相关的事件任务的最后一次执行时间。	S
%SW153	事件执行时间	以毫秒为单位指示 HSC0 或 HSC2 的阈值 0 相关事件任务的最后一次执行时间。	S
%SW154	事件执行时间	以毫秒为单位指示 HSC0 或 HSC2 的阈值 1 相关事件任务的最后一次执行时间。	S
%SW155	事件执行时间	以毫秒为单位指示 HSC1 或 HSC3 的阈值 0 相关事件任务的最后一次执行时间。	S
%SW156	事件执行时间	以毫秒为单位指示 HSC1 或 HSC3 的阈值 1 相关事件任务的最后一次执行时间。	S
%SW157	定期执行时间	以毫秒为单位指示定期任务的上次执行时间。	S
%SW158	周期性平均时间	以毫秒为单位指示定期任务的平均执行时间（最后 5 次）。	S
%SW159	事件 0 平均时间	以毫秒为单位指示输入 %I0.2 的相关事件任务的平均执行时间（最后 5 次）。	S
%SW160	事件 1 平均时间	以毫秒为单位指示输入 %I0.3 的相关事件任务的平均执行时间（最后 5 次）。	S
%SW161	事件 2 平均时间	以毫秒为单位指示输入 %I0.4 的相关事件任务的平均执行时间（最后 5 次）。	S
%SW162	事件 3 平均时间	以毫秒为单位指示输入 %I0.5 的相关事件任务的平均执行时间（最后 5 次）。	S
%SW163	事件 4 平均时间	以毫秒为单位指示 HSC0 的阈值 0 相关事件任务的平均执行时间（最后 5 次）。	S
%SW164	事件 5 平均时间	以毫秒为单位指示 HSC0 的阈值 1 相关事件任务的平均执行时间（最后 5 次）。	S
%SW165	事件 6 平均时间	以毫秒为单位指示 HSC1 的阈值 0 相关事件任务的平均执行时间（最后 5 次）。	S
%SW166	事件 7 平均时间	以毫秒为单位指示 HSC1 的阈值 1 相关事件任务的平均执行时间（最后 5 次）。	S
S 由系统控制 **U** 由用户控制 **SIM** 在模拟器中应用			

系统字	功能	描述	控制
%SW168	Modbus TCP - 正在使用的连接	指示正在使用的以太网 Modbus TCP 服务器连接的数量 **注意：**如果您断开电缆，连接不会立即关闭。每当电缆重新与网络连接时，它会请求一个新连接和由 %SW168 增加所表明的在用连接的数量。	S
%SW170	传输的帧 - 串行线路 1	指示通过串行线路 1 传输的帧数	S
%SW171	传输的帧 – 串行线路 2	指示通过串行线路 2 传输的帧数	S
%SW172	传输的帧 – USB	指示通过 USB 通道传输的帧数	S
%SW173	传输的帧 – Modbus TCP	指示通过以太网上的 Modbus TCP 传输的帧数	S
%SW174	成功接收的帧 – 串行线路 1	指示通过串行线路 1 正确接收的帧数	S
%SW175	成功接收的帧 – 串行线路 2	指示通过串行线路 2 正确接收的帧数	S
%SW176	成功接收的帧 – USB	指示通过 USB 通道正确接收的帧数	S
%SW177	成功接收的帧 – Modbus TCP	指示通过以太网上的 Modbus TCP 正确接收的帧数	S
%SW178	错误接收的帧 – 串行线路 1	指示在串行线路 1 中检测到的错误接收的帧数	S
%SW179	错误接收的帧 – 串行线路 2	指示在串行线路 2 中检测到的错误接收的帧数	S
%SW180	错误接收的帧 – USB	指示在 USB 通道中检测到的错误接收的帧数	S
%SW181	错误接收的帧 – Modbus TCP	指示在以太网上的 Modbus TCP 中检测到的错误接收的帧数	S
%SW188	传输帧 - Modbus 映射表	通过 Modbus 映射表所传输的帧的总数	S
%SW189	接收帧 - Modbus 映射表	通过 Modbus 映射表所接收的无错误帧的总数	S
%SW190, %SW191	1 类发送的传出数据包	为隐式（1 类）连接而发送的传出数据包总数。	S
%SW192, %SW193	1 类接收的传入数据包	为隐式（1 类）连接而接收的传入数据包总数。	S
%SW194, %SW195	接收的无连接传入数据包	传入的无连接数据包总数，包括在检出错误时会被返回的数据包。	S
S 由系统控制 U 由用户控制 SIM 在模拟器中应用			

系统字	功能	描述	控制
%SW196, %SW197	无效的无连接传入数据包	具有无效格式或定位了非支持服务、类别、实例、属性或成分的无连接传入数据包的总数。	S
%SW198, %SW199	为显式（3 类）连接而接收的传入数据包	显式（3 类）连接的传入数据包的总数， 包括在检出错误时会被返回的数据包。	S
%SW200, %SW201	无效的 3 类传入数据包	具有无效格式或定位了非支持服务、类别、实例、 属性或成分的显式（3 类）数据包的总数。	S
%SW202	实例输入	SoMachine Basic 中配置的实例输入。缺省值：0	S
%SW203	输入大小	SoMachine Basic 中配置的输入大小。缺省值：0	S
%SW204	实例输出	SoMachine Basic 中配置的实例输出。缺省值：0	S
%SW205	输出大小	SoMachine Basic 中配置的输出大小。缺省值：0	S
%SW206	超时	在连接中已发生的连接超时的总数。缺省值：0	S, U
%SW210	IOScanner SL1 状态	包含串行线路 1 上的 Modbus Serial IOScanner 的状态： ● 0：IOScanner 停止 ● 1：IOScanner 正向设备发送初始化请求 ● 2：IOScanner 正在运行 ● 3：IOScanner 正部分运行（一些设备未被扫描） ● 4：IOScanner 已暂停	S
%SW212	Modbus TCP IOScanner 的状态	包含以太网上 Modbus TCP IOScanner 的状态： ● 0：IOScanner 停止 ● 1：IOScanner 正向设备发送初始化请求 ● 2：IOScanner 正在运行 ● 3：IOScanner 正部分运行（一些设备未被扫描） ● 4：IOScanner 已暂停 **注意：** 对于此需要支持的系统字，应用程序必须配置有至少为**等级 6.0** 的功能等级。	S
S 由系统控制 **U** 由用户控制 **SIM** 在模拟器中应用			

M100/M200 可编程控制器 代码 ID

下表显示了 M100/M200 可编程控制器 引用的代码 ID：

型号	代码 ID
TM100C16R	0x07C1
TM100C24R	0x07C2
TM100C40R	0x07C4
TM100C16RN	0x07C6
TM100C24RN	0x07C7
TM100C32RN	0x07C8
TM100C40RN	0x07C9
TM200C16R	0x07A1
TM200C16T	0x07A3
TM200C16U	0x07A2
TM200C24R	0x07A7
TM200CE24R	0x07A8
TM200C24T	0x07AB
TM200CE24T	0x07AC
TM200C24U	0x07A9
TM200CE24U	0x07AA
TM200C32R	0x07B9
TM200CE32R	0x07BA
TM200C40R	0x07AD
TM200CE40R	0x07AE
TM200C40T	0x07B1
TM200CE40T	0x07B2
TM200C40U	0x07AF
TM200CE40U	0x07B0
TM200C60R	0x07B3

输入通道状态 (%IWS)

简介

下面介绍有关输入通道状态字属性的信息。存在专用输入通道状态字，针对使用 I/O 扩展模块或 TMCR2 卡盒添加的每个模拟量输入通道。

显示输入通道状态字属性

遵循以下步骤以显示输入通道状态字属性：

步骤	动作
1	选择**编程**窗口左侧区域的**工具**选项卡。
2	单击**系统对象 → 输入状态字**。 **结果**：显示输入通道状态字属性。

输入通道状态字属性

下表介绍了输入通道状态字的每个属性：

参数	可编辑	值	缺省值	描述
已使用	否	TRUE/FALSE	FALSE	指示是否在程序中引用输入通道状态字。
地址	否	%IWSx.y或 %IWS0.x0y	–	输入通道状态字的地址。 对于 I/O 扩展模块： ● x 表示模块编号 ● y 表示通道编号 对于模拟量扩展板： ● x 表示扩展板编号 ● y 表示通道编号 例如，%IWS0.101 表示逻辑控制器第一个插槽中扩展板的第二个通道的地址。
符号	是	–	–	与输入通道状态字关联的符号。 在**符号**列中双击，然后键入要与此输入通道状态字关联的符号的名称。 如果某个符号已经存在，请右键单击**符号**列，然后选择**搜索并替换**， 在整个程序和/或程序注释中查找并替换出现此符号的地方。
注释	是	–	–	与输入通道状态字关联的注释。 双击**注释**列，然后键入要与输入通道状态字关联的可选注释。

有关详情

要查看输入通道状态字可能的值：

有关以下内容的信息：	请参阅...
TM3 扩展模块	TM3 模拟量 I/O 模块诊断 *(参见 Modicon TM3 (SoMachine Basic), 扩展模块配置, 编程指南)*
TM3R 扩展模块	TM3R 模拟量 I/O 模块诊断 (参见第 *143* 页)
TM2 扩展模块	TM2 模拟量 I/O 模块诊断
TMCR2 扩展板	TMCR2 模拟量扩展板诊断 (参见第 *134* 页)

输出通道状态 (%QWS)

简介

下面介绍有关输出状态字属性的信息。存在专用输出通道状态字，针对使用 I/O 扩展模块或 TMCR2 卡盒添加的每个模拟量输出通道。

显示输出通道状态字属性

遵循以下步骤以显示输出通道状态字属性：

步骤	动作
1	选择**编程**窗口左侧区域的**工具**选项卡。
2	单击**系统对象 → 输出状态字**。 **结果**：在属性窗口中显示输出通道状态字属性。

输出通道状态字属性

下表介绍了输出通道状态字的每个属性：

参数	可编辑	值	缺省值	描述
已使用	否	TRUE/FALSE	FALSE	指示是否在程序中引用输出通道状态字。
地址	否	%QWSx.y或 %QWS0.x0y	–	输出通道状态字的地址。 对于 I/O 扩展模块： ● x 表示模块编号 ● y 表示通道编号 对于扩展板： ● x 表示扩展板编号 ● y 表示通道编号 例如，%QWS3.0 表示连接到逻辑控制器的第三个 I/O 扩展模块中的第一个输出通道的地址。
符号	是	–	–	与输出通道状态字关联的符号。 在**符号**列中双击，然后键入要与此输出通道状态字关联的符号的名称。 如果某个符号已经存在， 请右键单击**符号**列，然后选择**搜索并替换**，在整个程序和/或程序注释中查找并替换出现此符号的地方。
注释	是	–	–	与输出通道状态字关联的注释。 双击**注释**列，然后键入要与输出通道状态字关联的可选注释。

有关详情

要查看输出通道状态字可能的值：

有关以下内容的信息：	请参阅…
TM3 扩展模块	TM3 模拟量 I/O 模块诊断 *(参见 Modicon TM3 (SoMachine Basic), 扩展模块配置, 编程指南)*
TM3R 扩展模块	TM3R 模拟量 I/O 模块诊断 (参见第 *143* 页)
TM2 扩展模块	TM2 模拟量 I/O 模块诊断 *(参见 Modicon TM2 (SoMachine Basic), 扩展模块配置, 编程指南)*
TMCR2 扩展板	TMCR2 模拟量扩展板诊断 (参见第 *134* 页)

术语

停止速度

步进点击在施加负载且没有步进损失的情况下停止移动的最大频率。

加速度/减速度

加速度是指从**启动速度**开始到目标速度为止，速度变化的比率。减速度是从目标速度开始到**停止速度**为止，速度变化的比率。这些速度变化由 PTO 功能根据梯形或 S 型曲线后的加速度、减速度和抖动率隐式管理。

协议

一种用于控制和启用两个计算端点和设备之间的连接、通讯和数据传输的惯例或标准。

后配置

（*后配置*）使用该选项可以在不更改应用程序的情况下修改应用程序的某些参数。后配置参数由存储在控制器上的某个文件定义。它们可能会使应用程序的配置参数过载。

启动速度

步进点击在施加负载且没有步进损失的情况下可以开始移动的最小频率。

周期执行

循环执行或周期性执行的任务。在周期模式下，可以确定执行任务的特定时间（周期）。如果执行时间短于这个时间，则在下一个循环之前将生成等待时间。如果执行时间超过这个时间，则控制系统将指示溢出。如果溢出过高，控制器将停止。

回归

用于为绝对移动确定参考点的方法。

应用程序

包括配置数据、符号和文档的程序。

扩展总线

扩展 I/O 模块和控制器之间的电子通讯总线。

指令列表语言

以指令列表语言编写的程序，包括由控制器按顺序执行的一系列基于文本的指令。每个指令均包括一个行号、一个指令代码和一个操作数（请参阅 IEC 61131-3）。

控制器

自动化工业流程（也称为可编程逻辑控制器或可编程控制器）。

控制网络

此网络中包含逻辑控制器、SCADA 系统、PC、HMI、交换机……

支持以下两种拓扑：

- 扁平：此网络中的所有模块和设备都属于同一个子网。
- 2 层：网络分为操作网络和控制器间网络。

这两个网络可以在物理上独立，但通常通过路由设备链接。

数字量 I/O

（*数字量输入/输出*）电子模块上与数据表位直接对应的单独电路连接。数据表位用于存储 I/O 电路上的信号值。它可以对 I/O 值进行控制逻辑数字访问。

梯形图语言

控制器程序指令的图形表示，其中包括控制器按顺序执行的一系列梯级中的触点、线圈和块符号（请参阅 IEC 61131-3）。

梯形斜坡

加速度/减速度斜坡，包含设置为 0% 的 JerkRatio 参数。

模拟量输入

用于将收到的电压或电流电平转换为数值。可以在逻辑控制器中存储和处理这些值。

模拟量输出

在逻辑控制器内转换数值，并按比例发送电压或电流电平。

程序

应用程序的组成部分，其中包括可以在逻辑控制器的存储器中安装的经过编译的源代码。

绝对运动

通过参考点定义的位置的移动。

配置

一个系统内硬件组件的布局和互连以及硬件和软件的参数，可决定系统的运行特性。

ASCII

（*美国信息交换标准码*）用于表示字母数字字符（如字母、数字以及某些图形和控制字符）的通讯协议。

BOOTP

（*引导程序协议*）可由网络客户端用于从服务器自动获取 IP 地址（可能还包括其他数据）的 UDP 网络协议。客户端使用客户端 MAC 地址向服务器标识自己。服务器会维护预先配置的客户端设备 MAC 地址及关联 IP 地址表，从而向客户端发送其预先配置的 IP 地址。BOOTP 最初用于使无盘主机能够通过网络远程启动。BOOTP 进程分配一个无限租期的 IP 地址。BOOTP 服务利用 UDP 端口 67 和 68。

DHCP

（*动态主机配置协议*）BOOTP 的高级扩展。DHCP 虽然较为高级，但是 DHCP 和 BOOTP 可以通用。（DHCP 可以处理 BOOTP 客户端请求。）

DWORD

（*双字*）以 32 位格式进行编码的类型。

EtherNet/IP

（*Ethernet工业协议*）用于工业系统中自动化解决方案制造的开放式通讯协议。EtherNet/IP 是在其上层执行公共工业协议的网络家庭成员。支持组织 (ODVA) 规定 EtherNet/IP 是为了实现全球适应性和介质独立性。

FreqGen

（*频率发生器*）使用可编程频率生成方波信号的功能。

GRAFCET

以结构和图形格式表示顺序操作的运行。

这是一种将任何顺序控制系统划分为一系列步骤的分析方法，动作、转换和条件均与这些步骤关联。

HSC

（*高速计数器*）一种对控制器或扩展模块输入上的脉冲进行计数的功能。

I/O

（*输入/输出*）

IEC 61131-3

工业自动化设备的 3 部分标准的第 3 部分。IEC 61131-3 针对控制器编程语言，并定义了两个图形编程语言和两个文本编程语言标准。图形编程语言既是梯形图语言又是功能块图语言。文本编程语言包括结构化文本和指令列表。

IL

（*指令列表*）以某种语言编写的程序，包括由控制器按顺序执行的一系列基于文本的指令。每个指令均包括一个行号、一个指令代码和一个操作数（请参阅 IEC 61131-3）。

jerk ratio

基于时间更改加速度和减速度的比例。

LAN

（*局域网*）在家庭、办公室或机构环境中实施的一种短距离通讯网络。

LD

（*梯形图*）控制器程序指令的图形表示，其中包括控制器按顺序执行的一系列梯级中的触点、线圈和块符号（请参阅 IEC 61131-3）。

LSB

（*最低有效位/字节*）在传统的十六进制或二进制表示法中，它是数字、地址或字段的一部分，作为最右侧的单值写入。

MAST

通过其编程软件运行的处理器任务。MAST 任务有两个段：

- IN：在 MAST 任务执行之前，将输入复制到 IN 段。
- OUT：在 MAST 任务执行完后，将输出复制到 OUT 段。

Modbus

允许在连接到同一网络的多个设备之间进行通讯的协议。

MSB

（*最高有效位/字节*）在传统的十六进制或二进制表示法中，是数字、地址或字段的一部分，作为最左侧的单值写入。

PID

（*比例*、*积分*、*微分*）广泛用于工业控制系统的通用控制回路反馈机制（控制器）。

POU

（*程序组织单元*）源代码的变量声明和相应的指令集。POUs 有助于简化软件程序、功能和功能块的模块化重用。经过声明后，POUs 便可相互使用。

PTO

（*脉冲串输出*）采用50-50的固定占空比在开启和关闭之间振荡以产生方波形式的快速输出。PTO 特别适用于如步进电机、频率转换器和伺服电机控制等应用。

PWM

（*脉冲宽度调制*）以可调占空比在关闭和开启之间振荡以产生矩形波形式的快速输出（尽管可以调整它来产生方形波）。

RS-232

基于三条电线的串行通讯总线的标准类型（也称为 EIA RS-232C 或 V.24）。

RS-485

基于两条电线的串行通讯总线的标准类型（也称为 EIA RS-485）。

RTC

（*实时时钟*）由电池供电可连续运转以显示当天时间和日历的时钟，即使在为延长电池使用寿命而未对控制器通电时也一样。

S 曲线 斜坡

加速度/减速度斜坡，包含大于 0% 的 JerkRatio 参数。

SFC

（*顺序功能图*）一种包括具有关联操作的步骤、具有相关联逻辑条件的转换，以及步骤和转换之间的定向链接的语言。（SFC 标准已在 IEC 848 中定义。符合 IEC 61131-3。）

索引

“已暂停”状态, *45*
%C, *25*
%DR, *25*
%FC, *25*, *231*
%HSC, *25*, *237*
%I, *25*, *224*
%IN, *300*
%IW, *25*, *226*
%IWM, *299*
%IWM/%QWM, *154*
%IWN, *303*
%IWNS（IOScanner 网络诊断代码）, *306*
%IWS 输入通道状态, *134*
%IWS 输入通道状态字节, *143*
%IWS（输入通道状态）, *482*
%KD, *25*
%KF, *25*
%KW, *25*
%M, *25*
%MD, *25*
%MF, *25*
%MSG, *25*
%MW, *25*
%PLS, *25*, *256*
%PTO, *25*
%PWM, *25*, *292*
%Q, *25*, *225*
%QN, *301*
%QW, *25*, *227*
%QWM, *298*
%QWN, *304*
%QWS 输出通道状态, *134*
%QWS 输出通道状态字节, *143*
%QWS（输出通道状态）, *484*
%R, *25*
%READ_VAR, *25*
%S, *25*, *454*
%SBR, *25*
%SC, *25*
%SEND_RECV_MSG, *25*
%SW, *25*, *463*
%SW118, *67*, *68*
%SW119, *67*, *68*
%SW120, *67*, *68*
%SW6, *43*
%TM, *25*
%WRITE_READ_VAR, *25*
%WRITE_VAR, *25*
Altivar 设备
    添加到 Modbus Serial IOScanner, *171*
configuring
    驱动器功能块, *269*
Exec Loader, *70*
    更新固件, *70*
FREQGEN
    功能块, *410*
    功能块配置, *412*
HOMING_MODE, *348*
I/O 对象
    数字量输入, *224*
    数字量输出, *225*
    模拟量输入, *226*
    模拟量输出, *227*
I/O 总线
    配置, *101*
I/O 总线速度, *139*
I/O 总线错误处理
    主动, *102*, *103*
I/O 扩展总线
    重启, *104*
I/O 配置一般信息
    一般做法, *136*
    概述, *102*
INDEX 短参考
    PTOHome, *345*
Input registers (IOScanner)
    属性, *303*
IOScanner 的
    数字量输入，属性, *300*
    数字量输出，属性, *301*
IOScanner，Modbus Serial, *170*
Machine.cfg 后配置文件, *53*, *54*

MC_Jog_ATV
启动点动模式, *270*
MC_MoveVel_ATV
以指定速度移动, *275*
MC_Power_ATV
启用/禁用电源级, *273*
MC_ReadMotionState_ATV
读取运动状态, *282*
MC_ReadStatus_ATV
读取设备状态, *280*
MC_Reset_ATV
确认和复位错误, *285*
MC_Stop_ATV
停止移动, *278*
Modbus Serial IOScanner
初始化请求助手, *173*
将设备添加到, *171*
设备和通道诊断位, *306*
通道助手, *175*
配置, *170*
配置通道, *176*
Modbus TCP, *153*
modbus TCP
映射表, *298*
Modbus TCP
映射表, *179*
远程设备, *155*
配置, *153*
配置 Modbus 映射, *153*
配置客户端模式, *155*
Modbus TCP IOScanner
初始化请求助手, *158*
设备和通道诊断位, *306*
通道助手, *160*
配置 Modbus 映射, *153*
配置客户端模式, *155*
配置通道, *161*
Modbus 映射表, *153*, *154*
motion task table
PTO, *322*
Output registers (IOScanner)
属性, *304*
PID, *25*
AT 选项卡, *437*
PID 选项卡, *436*
参数, *448*
常规选项卡, *432*
开放回路调整, *451*
描述, *442*
操作模式, *414*
标准配置, *419*
状态和检测到的错误代码, *445*
编程和配置, *444*
自整定, *416*
输入选项卡, *435*
输出选项卡, *439*
配置助手, *431*
闭合回路调整, *450*
PTO
motion task table, *322*
功能, *309*
配置, *320*
PTO_ERROR, *349*, *350*
PTO_PARAMETER, *348*
PTOHome
INDEX 短参考, *345*
SCH, *25*
SD 卡, *188*
克隆, *186*
后配置管理, *197*
应用程序管理, *193*
更新固件, *188*
SoMachine Basic
工程, *111*
硬件树, *138*
设备树, *111*
项目, *138*
TM3R
配置, *135*
TM3R 数字量 I/O 扩展模块
TM3R, *137*
TM3R 模拟量 I/O 模块
诊断, *143*

TMCR2 模拟量 I/O 模块
TMCR2AI2, *115*
TMCR2AM3 , *119*
TMCR2AQ2C , *117*
TMCR2AQ2V , *118*
TMCR2DM4U , *114*
TMCR2TI2 , *121*
TMCR2 模拟量扩展板
诊断, *134*
TMCR2••• 扩展板
添加至配置, *111*
TMCR2AI2, *115*
TMCR2AM3, *119*
TMCR2AQ2C, *117*
TMCR2AQ2V, *118*
TMCR2DM4U, *114*
TMCR2SL1, *124*
TMCR2SL1A, *129*
TMCR2TI2, *121*
一般从站设备, *171*
上载应用程序, *43*
下载应用程序, *43*
串行线路, *164*
简介, *124*, *129*
配置, *125*, *130*, *164*
配置 Modbus Serial IOScanner, *170*
配置参数, *165*
串行线路扩展板, *124*, *129*
主动 I/O 总线错误处理, *102*
以太网, *147*
设备和通道诊断位, *306*
配置, *147*
配置参数, *148*
以指定速度移动
MC_MoveVel_ATV, *275*
任务, *36*
任务和扫描模式, *36*
停止控制器, *45*
停止移动
MC_Stop_ATV, *278*
克隆 (备份到 SD 卡, *186*
兼容性
扩展板, *110*
内置输入/输出
概述, *16*
配置, *72*
内置通讯
功能, *16*
配置, *145*
冷启动, *46*
减速度斜坡, *312*
初始化控制器, *44*
初始化请求助手
Modbus Serial IOScanner, *173*
Modbus TCP IOScanner, *158*
删除模块, *139*
功能
PTO, *309*
关键功能, *15*
内置通讯, *15*
扩展板, *109*
功能块
MC_Jog_ATV, *270*
MC_MoveVel_ATV, *275*
MC_Power_ATV, *273*
MC_ReadMotionState_ATV, *282*
MC_ReadStatus_ATV, *280*
MC_Reset_ATV, *285*
MC_Stop_ATV, *278*
快速计数器, *231*
脉冲, *256*
脉冲宽度调制, *292*
频率发生器 (%FREQGEN), *410*
高速计数器, *237*
功能块对象代码
HOMING_MODE, *348*
PTO_PARAMETER, *348*
方向, *347*
缓冲模式, *347*
加速度斜坡, *312*
单位 ID, *154*
卸下扩展板, *111*
参数, 后配置, *53*
反向, *316*
受支持设备, *105*

变量
断电时自动保存, *47*
根据用户请求删除, *48*
根据用户请求恢复, *47*
用户请求时保存, *47*
可移除存储器, *16*
后配置
参数, *53*
图示, *53*
应用时, *53*
文件 Machine.cfg, *53*, *54*
后配置, 备份到 SD 卡, *186*
启动控制器, *44*
启动点动模式 ， MC_Jog_ATV, *270*
启用/禁用电源级
MC_Power_ATV, *273*
固件, *70*, *188*
使用 Exec Loader 更新, *70*
使用 SD 卡更新, *188*
固件, 备份到 SD 卡, *186*
固件更新, *43*
备份控制器存储器, *211*
存储器, *16*
存储器 ，可移除, *16*
存储器对象
备份和恢复, *211*
存储字
断电时自动保存, *47*
根据用户请求删除, *48*
根据用户请求恢复, *47*
用户请求时保存, *47*
实时时钟 (RTC), *15*
对象, *24*
允许的最大数量, *32*
定义, *24*
对象类型, *25*
寻址, *29*
寻址示例, *29*
网络, *297*
寻址
移动模块时, *138*
嵌入式 HSC 的 HSC 模式
模数回路, *250*
应用程序, 备份到 SD 卡, *186*
应用程序下载, *44*
微型 SD 卡
参见 SD 卡, *182*
快速计数器
描述, *231*
编程示例, *235*
配置, *233*
总线速度, *139*
恢复控制器存储器, *211*
扩展板
兼容性, *110*
功能, *109*
卸下, *111*
描述, *109*
替换, *111*
添加至 SoMachine Basic 配置, *111*
配置, *107*, *112*
扩展模块
TM2, *105*
TM3, *105*, *105*
配置, *105*, *140*
扫描模式, *36*
持续性变量, *47*
控制器
功能, *15*
参考, *17*, *18*
持续性变量, *47*
状态, *41*
状态转换, *44*
编程语言, *15*
输出行为, *49*
配置, *59*, *69*
配置功能, *22*
控制器状态, *40*, *41*
已停止, *42*
已暂停, *42*
引导, *42*
断电, *43*
正在运行, *42*
空, *42*

控制器状态转换, *44*
停止, *45*
冷启动, *46*
初始化, *44*
启动, *44*
已暂停, *45*
应用程序下载, *44*
运行, *45*
控制器输出行为, *49*
故障预置值, *50*
故障预置执行, *50*
故障预置管理, *49*
故障预置行为配置, *50*
硬件初始化值, *49*
软件初始化值, *49*
输出强制, *51*
输出重置, *51*
描述
扩展板, *109*
插入模块, *138*
故障预置
值, 配置, *76*
值,配置, *141*
故障预置值, *50*, *298*
故障预置执行, *50*
故障预置管理, *49*
故障预置行为配置, *50*
数字量输入, *73*
属性, *224*
配置, *73*, *140*
配置参数, *73*
数字量输出, *76*
属性, *225*
配置, *76*
配置参数, *76*
配置故障预置值, *76*, *141*
断电，保存变量, *47*
方向, *347*
映射表，Modbus TCP, *154*, *179*, *299*
映射表、modbus TCP, *298*
显示
编程详细信息, *112*, *142*
更新固件, *70*, *188*
替换
扩展板, *111*
扩展模块, *138*
最大模块数, *139*
模块
删除, *139*
插入, *138*
替换, *138*
最大数, *139*
混合不同类型, *139*
添加, *138*
模拟量输入
属性, *226*
模拟量输出
属性, *227*
模数回路
嵌入式 HSC 的 HSC 模式, *250*
注释
显示, *112*, *142*
混合模块类型, *139*
电源, *15*
硬件初始化值, *49*
硬件树, *138*
确认和复位错误
MC_Reset_ATV, *285*
符号, 显示, *112*, *142*
系统 LED, *70*
系统位
%S106, *103*
%S107, *104*
系统位 (%S), *454*
系统字
%SW118, *67*, *68*
%SW119, *67*, *68*
%SW120, *67*, *68*
系统字 (%SW), *463*
系统对象
系统位 (%S), *454*
系统字 (%SW), *463*
维持值故障预置值, *298*
缓冲模式, *347*
编程详细信息
显示, *112*, *142*
编程语言
支持, *15*

网络对象, *154*, *297*
%IN, *300*
%QN, *301*
Input registers (IOScanner), *303*
Output registers (IOScanner), *304*
输入寄存器, *298*
输出寄存器 (Modbus TCP), *299*
网络诊断代码 (%IWNS), *306*
脉冲
描述, *256*
编程示例, *261*
配置, *258*
脉冲发生器, *78*
FREQGEN 配置, *87*
PLS 配置, *81*
PTO 配置, *84*
PWM 配置, *82*
简介, *78*
配置, *78*
脉冲宽度调制
描述, *292*
编程示例, *296*
配置, *293*
被动 I/O 总线错误处理, *103*
警戒时钟定时器, *37*
设备
添加到 Modbus Serial IOScanner, *171*
设备树, *111*
诊断字节 (%IWS, %QWS)), *134*, *143*
读取设备状态
%MC_ReadStatus_ATV, *280*
读取运动状态
MC_ReadMotionState_ATV, *282*
跃度率, *312*
软件初始化值, *49*
输入寄存器
属性, *298*
输入通道状态 (%IWS), *134*, *143*
输入通道状态 (%IWS)), *482*
输出寄存器
属性, *299*
输出强制, *51*
输出行为, *49*
输出通道状态 (%QWS), *134*, *143*
输出通道状态 (%QWS)), *484*
输出重置, *51*
运行/停止, *15*, *75*
将数字量输入配置为, *75*
运行控制器, *45*
远程设备
添加到 Modbus TCP, *155*
通道
Modbus Serial IOScanner, *176*
Modbus TCP IOScanner, *161*
通道助手
Modbus Serial IOScanner, *175*
Modbus TCP IOScanner, *160*
逻辑控制器
内置 I/O, *138*
添加到 SoMachine Basic 配置, *138*
配置, *137*
Modbus Serial IOScanner, *170*
Modbus TCP, *153*
串行线路, *164*
建立配置, *60*
扩展板, *112*
扩展模块, *105*, *105*
控制器, *69*
故障预置值, *141*
数字量 I/O, *140*
数字量输入, *73*, *140*
数字量输出, *76*
高速计数器, *90*
配置
以太网, *147*
重启 I/O 扩展总线, *104*
错误代码
驱动器功能块, *287*
频率发生器
功能块, *410*
配置, *87*
驱动器功能块
配置, *269*
驱动器功能块：错误代码, *287*

高速计数器, *90*, *94*, *99*
专用 I/O 分配, *91*
双相 , 脉冲/方向, *94*
描述, *237*
计数模式, *241*
配置, *90*, *94*, *99*
配置参数, *92*
频率计, *99*
频率计模式, *252*