

### STEP 7

### TIA Portal V16 中 S7-1200 Motion Control V6.0 升级为 V7.0

#### 功能手册

前言

简介

1

使用 S7-1200 运动控制的基础知识

2

运动控制使用指南

3

使用版本

4

定位轴工艺对象

5

工艺对象命令表

6

下载到 CPU

7

调试

8

编程

9

轴 - 诊断

10

指令

11

附录

12

## 法律资讯

### 警告提示系统

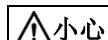
为了您的人身安全以及避免财产损失，必须注意本手册中的提示。人身安全的提示用一个警告三角表示，仅与财产损失有关的提示不带警告三角。警告提示根据危险等级由高到低如下表示。



表示如果不采取相应的小心措施，**将会导致死亡或者严重的人身伤害。**



表示如果不采取相应的小心措施，**可能导致死亡或者严重的人身伤害。**



表示如果不采取相应的小心措施，**可能导致轻微的人身伤害。**

#### 注意

表示如果不采取相应的小心措施，**可能导致财产损失。**

当出现多个危险等级的情况下，每次总是使用最高等级的警告提示。如果在某个警告提示中带有警告可能致人身伤害的警告三角，则可能在该警告提示中另外还附带有可能导致财产损失的警告。

### 合格的专业人员

本文件所属的产品/系统只允许由符合各项工作要求的**合格人员**进行操作。其操作必须遵照各自附带的文件说明，特别是其中的安全及警告提示。由于具备相关培训及经验，合格人员可以察觉本产品/系统的风险，并避免可能的危险。

### 按规定使用 Siemens 产品

请注意下列说明：



Siemens 产品只允许用于目录和相关技术文件中规定的使用情况。如果要使用其他公司的产品和组件，必须得到 Siemens 推荐和允许。正确的运输、储存、组装、装配、安装、调试、操作和维护是产品安全、正常运行的前提。必须保证允许的环境条件。必须注意相关文件中的提示。

### 商标

所有带有标记符号 ® 的都是 Siemens AG 的注册商标。本印刷品中的其他符号可能是一些其他商标。若第三方出于自身目的使用这些商标，将侵害其所有者的权利。

### 责任免除

我们已对印刷品中所述内容与硬件和软件的一致性作过检查。然而不排除存在偏差的可能性，因此我们不保证印刷品中所述内容与硬件和软件完全一致。印刷品中的数据都按规定经过检测，必要的修正值包含在下一个版本中。

# 前言

## 本手册用途

此文档可提供有关 S7-1200 运动控制的详细信息。此文档的内容与结构和 STEP 7 V16 在线帮助相同。能够与 STEP 7 进行交互是理解本文档中大部分内容的前提条件。

本文档适用于 STEP 7 程序员和运动控制应用自动化系统的组态、调试和维修领域的人员。

## 需要的基本知识

要理解本文档，需具备自动化工程与运动控制领域的常识。

也需要熟悉在 Windows 操作系统下使用计算机或编程设备的知识。

因为 S7-1200 运动控制是以 STEP 7 为基础的，所以您需要具备使用 STEP 7 的知识。

## 本手册适用范围

本手册适用于 STEP 7 V16。

## 约定

本文档中包含所述设备的相关图片。这些图片可能与实际提供的设备略有不同。

请同时遵循以下所标注的注意事项：

---

### 说明

这些注意事项包含有关本文档中所述产品、产品操作或应特别关注的文档部分的重要信息。

---

## 其它支持

如果您对本手册中所述的产品有任何疑问，而在本文档中未找到答案，请与我们当地办事处的 **Siemens** 合作伙伴联系。

可以在 Internet (<http://www.automation.siemens.com/mcms/aspa-db/en/automation-technology/Pages/default.aspx>) 上找到相应的联系人。

有关各种 SIMATIC 产品和系统的技术文档指南，敬请访问 Internet (<http://www.siemens.com/simatic-tech-doku-portal>)。

有关在线产品目录和在线订购系统，敬请访问 Internet (<https://mall.industry.siemens.com>)。

## 培训中心

西门子的系列课程，可帮助用户快速使用 S7 自动化系统。请联系您当地的培训中心或中央培训中心 (<http://support.automation.siemens.com/CN/view/zh/24486113>)。

## 安全性信息

**Siemens** 为其产品及解决方案提供了工业信息安全功能，以支持工厂、系统、机器和网络的安全运行。

为了防止工厂、系统、机器和网络受到网络攻击，需要实施并持续维护先进且全面的工业信息安全保护机制。**Siemens** 的产品和解决方案构成此类概念的其中一个要素。

客户负责防止其工厂、系统、机器和网络受到未经授权的访问。只有在有必要连接时并仅在采取适当安全措施（例如，防火墙和/或网络分段）的情况下，才能将该等系统、机器和组件连接到企业网络或 Internet。

关于可采取的工业信息安全措施的更多信息，请访问 (<https://www.siemens.com/industrialsecurity>)。

**Siemens** 不断对产品和解决方案进行开发和完善以提高安全性。**Siemens** 强烈建议您及时更新产品并始终使用最新产品版本。如果使用的产品版本不再受支持，或者未能应用最新的更新程序，客户遭受网络攻击的风险会增加。

要及时了解有关产品更新的信息，请订阅 **Siemens** 工业信息安全 RSS 源，网址为 (<https://www.siemens.com/industrialsecurity>)。

# 目录

前言 .....	.3
<b>1 简介 .....</b>	<b>12</b>
1.1 CPU S7-1200 的运动功能.....	12
1.2 用于运动控制的硬件组件 .....	13
<b>2 使用 S7-1200 运动控制的基础知识 .....</b>	<b>16</b>
2.1 PTO 上的步进电机 .....	16
2.1.1 运动控制相关的 CPU 输出 .....	16
2.1.2 脉冲接口的工作原理 .....	20
2.1.3 信号类型与行进方向之间的关系 .....	21
2.2 PROFIdrive 驱动装置 / 模拟量驱动装置接口 .....	25
2.2.1 驱动装置和编码器连接 .....	25
2.2.2 自动传送设备中的驱动装置与编码器参数 .....	26
2.2.3 PROFIdrive .....	29
2.2.4 闭环控制 .....	30
2.2.5 PROFIdrive 驱动装置/PROFIdrive 编码器的数据连接 .....	31
2.2.6 带模拟量驱动接口的数据连接驱动装置 .....	37
2.2.7 过程响应 .....	39
2.2.7.1 用于运动控制的组织块 .....	39
2.2.7.2 过程映像分区“OB 伺服 PIP” .....	42
2.2.7.3 操作顺序和超时 .....	42
2.2.7.4 操作模式 .....	43
2.3 硬件和软件限位开关 .....	45
2.4 冲击限制 .....	46
2.5 回原点 .....	47
<b>3 运动控制使用指南 .....</b>	<b>50</b>
<b>4 使用版本 .....</b>	<b>51</b>
4.1 版本概述 .....	51
4.2 更改工艺版本 .....	58
4.3 变量兼容表 V1...3 <-> V4...5 .....	58
4.4 变量兼容表 V4...5 <-> V6 .....	64
4.5 兼容性报文 V6 <-> V7 .....	65
4.6 限位开关状态 .....	71

---

5 定位轴工艺对象 .....	72
5.1 集成定位轴工艺对象 .....	72
5.2 定位轴工艺对象的工具 .....	76
5.3 添加一个定位轴工艺对象 .....	79
5.4 组态定位轴工艺对象 .....	80
5.4.1 使用组态对话框 .....	80
5.4.2 监视值 .....	81
5.4.3 基本参数 .....	82
5.4.3.1 组态 - 常规 .....	82
5.4.3.2 组态 - 驱动器 .....	84
5.4.3.3 组态 - 编码器 .....	91
5.4.4 扩展参数 .....	97
5.4.4.1 机械 .....	97
5.4.4.2 组态 - 模数 (仅限 PROFIdrive/模拟驱动器接口) .....	100
5.4.4.3 位置限制 .....	101
5.4.4.4 动态 .....	107
5.4.4.5 回原点 (自定位轴工艺对象 V2 起) .....	117
5.4.4.6 定位监控 .....	127
5.4.4.7 组态 - 控制环 (仅限 PROFIdrive 和模拟驱动器接口) .....	130
5.4.5 参数视图 .....	131
5.4.5.1 参数视图简介 .....	131
5.4.5.2 参数视图结构 .....	134
5.4.5.3 打开参数视图 .....	138
5.4.5.4 参数视图默认设置 .....	139
5.4.5.5 使用参数视图 .....	142
5.4.6 组态用于运动控制的工艺模块 .....	156
5.4.6.1 概述 .....	156
5.4.6.2 TM PosInput 1 / TM PosInput 2 .....	157
5.4.6.3 TM Count 1x24V / TM Count 2x24V .....	159

<b>6</b>	<b>工艺对象命令表</b>	<b>161</b>
6.1	使用作业表工艺对象	161
6.2	命令表工艺对象工具	161
6.3	添加工艺对象命令表	162
6.4	组态命令表工艺对象	163
6.4.1	使用组态对话框	163
6.4.2	监视值	164
6.4.3	基本参数	165
6.4.3.1	组态 - 常规	165
6.4.3.2	命令表组态	165
6.4.3.3	快捷菜单命令 - 命令表	171
6.4.3.4	使用趋势图	173
6.4.3.5	快捷菜单命令 - 曲线图	177
6.4.3.6	从“完成命令”过渡到“混合运动”	178
6.4.3.7	在用户程序中更改命令表组态	181
6.4.4	扩展参数	182
6.4.4.1	组态 - 扩展参数	182
6.4.4.2	组态 - 动态	183
6.4.4.3	组态 - 限值	184
<b>7</b>	<b>下载到 CPU</b>	<b>186</b>
<b>8</b>	<b>调试</b>	<b>188</b>
8.1	轴控制面板	188
8.2	调节	192
<b>9</b>	<b>编程</b>	<b>197</b>
9.1	运动控制语句概述	197
9.2	创建用户程序	198
9.3	编程注意事项	202
9.4	断电和重新启动后运动控制命令的行为	204
9.5	监视激活的命令	204
9.5.1	监视激活的命令	204
9.5.2	具有输出参数“Done”的运动控制指令	205
9.5.3	运动控制指令 MC_MoveVelocity	209
9.5.4	运动控制指令 MC_MoveJog	212
9.6	运动控制语句的错误显示	217
9.7	重新启动工艺对象	218
9.8	函数块的参数传递	219

---

<b>10</b>	<b>轴 - 诊断</b>	<b>221</b>
10.1	状态和错误位（自工艺对象 V4 起）	221
10.2	运动状态	225
10.3	动态设置	226
10.4	PROFIdrive 帧	227
<b>11</b>	<b>指令</b>	<b>228</b>
11.1	S7-1200 Motion Control（自 V6 起）	228
11.1.1	MC_Power	228
11.1.1.1	MC_Power: 启用、禁用轴 V6 及更高版本	228
11.1.1.2	MC_Power: 功能图（V6 及以上版本）	233
11.1.2	MC_Reset	235
11.1.2.1	MC_Reset: 确认故障，重新启动工艺对象（V6 及以上版本）	235
11.1.3	MC_Home	238
11.1.3.1	MC_Home: 使轴归位，设置参考点（V6 及以上版本）	238
11.1.4	MC_Halt	245
11.1.4.1	MC_Halt: 停止轴（V6 及以上版本）	245
11.1.4.2	MC_Halt: 功能图（V6 及以上版本）	247
11.1.5	MC_MoveAbsolute	249
11.1.5.1	MC_MoveAbsolute: 轴的绝对定位（V6 及以上版本）	249
11.1.5.2	MC_MoveAbsolute: 功能图（V6 及以上版本）	252
11.1.6	MC_MoveRelative	254
11.1.6.1	MC_MoveRelative: 轴的相对定位（V6 及以上版本）	254
11.1.6.2	MC_MoveRelative: 功能图（V6 及以上版本）	257
11.1.7	MC_MoveVelocity	259
11.1.7.1	MC_MoveVelocity: 以设定速度移动轴（V6 及以上版本）	259
11.1.7.2	MC_MoveVelocity: 功能图（V6 及以上版本）	263
11.1.8	MC_MoveJog	265
11.1.8.1	MC_MoveJog: 在点动模式下移动轴（V6 及以上版本）	265
11.1.8.2	MC_MoveJog: 功能图（V6 及以上版本）	268
11.1.9	MC_CommandTable	270
11.1.9.1	MC_CommandTable: 按照运动顺序运行轴命令（V6 及以上版本）	270
11.1.10	MC_ChangeDynamic	274
11.1.10.1	MC_ChangeDynamic: 更改轴的动态设置（V6 及以上版本）	274
11.1.11	MC_ReadParam	277
11.1.11.1	MC_ReadParam: 连续读取定位轴的运动数据（V6 及以上版本）	277
11.1.12	MC_WriteParam	280
11.1.12.1	MC_WriteParam: 写入定位轴的变量（V6 及以上版本）	280

12	附录 .....	283
12.1	将多个轴与同一个 PTO 一起使用 .....	283
12.2	将多个驱动器与同一个 PTO 一起使用 .....	286
12.3	在更高优先级等级（执行级别）中跟踪作业 .....	287
12.4	使用通过 PTO 的驱动器连接时的软限位开关时的特殊情况 .....	290
12.4.1	软件限位开关与回原点操作结合使用 .....	290
12.4.2	软件限位开关和软件限位开关的位置变化。 .....	294
12.4.3	软件限位开关与动态更改结合使用 .....	295
12.5	降低短时间定位的速度 .....	297
12.6	启动/停止速度的动态调整 .....	297
12.7	维修时，无需位置控制，即可移动轴 .....	298
12.8	ErrorID 和 ErrorInfo（工艺对象 V6...V7）的列表 .....	300
12.8.1	错误和 ErrorID 概览 .....	300
12.8.2	ErrorID 16#8000-16#8013 .....	301
12.8.3	ErrorID 16#8200-16#820C .....	307
12.8.4	ErrorID 16#8400-16#8412 .....	310
12.8.5	错误 ID 16#8600-16#864B .....	315
12.8.6	ErrorID 16#8700-16#8704 .....	333
12.8.7	ErrorID 16#8FF .....	334
12.9	定位轴工艺对象 V6...7 的变量 .....	335
12.9.1	图例 .....	335
12.9.2	位置值和速度值变量 V6 及以上版本 .....	337
12.9.3	仿真变量（V6 及以上版本） .....	338
12.9.4	执行器变量（V6 及以上版本） .....	339
12.9.5	Sensor[1] 变量（V6 及以上版本） .....	345
12.9.6	单位变量（V6 及以上版本） .....	354
12.9.7	机械变量（V6 及以上版本） .....	355
12.9.8	Modulo 变量（V6 及以上版本） .....	356
12.9.9	DynamicLimits 变量（V6 及以上版本） .....	357
12.9.10	DynamicDefaults（V6 及以上版本） .....	358
12.9.11	PositionLimits_SW 变量（V6 及以上版本） .....	359
12.9.12	PositionLimits_HW 变量（V6 及以上版本） .....	361
12.9.13	归位变量（V6 及以上版本） .....	364
12.9.14	PositionControl 变量（V6 及以上版本） .....	365
12.9.15	FollowingError 变量（V6 及以上版本） .....	366
12.9.16	PositionMonitoring 变量（V6 及以上版本） .....	367
12.9.17	StandstillSignal 变量（V6 及以上版本） .....	368
12.9.18	StatusPositioning 变量（V6 及以上版本） .....	369
12.9.19	StatusDrive 变量（V6 及以上版本） .....	371
12.9.20	StatusSensor 变量（V6 及以上版本） .....	372
12.9.21	StatusBits 变量（V6 及以上版本） .....	374

---

12.9.22	ErrorBits 变量 (V6 及更高版本) .....	380
12.9.23	ControlPanel 变量 (V6 及更高版本) .....	382
12.9.24	内部变量 (V6 及更高版本) .....	384
12.9.25	工艺对象变量的更新 .....	384
12.10	工艺对象命令表 V6 中的变量.....	385
12.11	版本 V1...6 .....	387
12.11.1	与运动控制相关的 CPU 输出 (工艺版本 V1...3) .....	387
12.11.2	组态对话框 .....	393
12.11.2.1	V1...3.....	393
12.11.2.2	V4.....	402
12.11.3	诊断 - 状态和错误位 (“轴”工艺对象 V1...3) .....	411
12.11.4	ErrorID 和 ErrorInfo.....	414
12.11.4.1	ErrorID 和 ErrorInfo 列表 (工艺对象 V4...5) .....	414
12.11.4.2	ErrorID 和 ErrorInfo 列表 (工艺对象 V2...3) .....	446
12.11.4.3	ErrorID 和 ErrorInfo 列表 (工艺对象 V1) .....	460
12.11.5	Legend V1...5.....	470
12.11.6	轴工艺对象 V1...3 的变量.....	472
12.11.6.1	Config 变量 V1...3.....	472
12.11.6.2	MotionStatus 变量 V1...3 .....	478
12.11.6.3	StatusBits 变量 V1...3 .....	479
12.11.6.4	ErrorBits 变量 V1...3 .....	483
12.11.6.5	内部变量 V1...3 .....	484
12.11.6.6	ControlPanel 变量 V1...3 .....	484
12.11.6.7	工艺对象变量的更新 .....	484
12.11.7	定位轴工艺对象 V4...5 的变量 .....	485
12.11.7.1	位置值和速度值变量 V4...5.....	485
12.11.7.2	执行器变量 V4...5 .....	486
12.11.7.3	Sensor[1] 变量 V4...5 .....	488
12.11.7.4	单位变量 V4...5 .....	493
12.11.7.5	机械变量 V4...5 .....	494
12.11.7.6	Modulo 变量 V4...5 .....	495
12.11.7.7	DynamicLimits 变量 V4...5 .....	496
12.11.7.8	DynamicDefaults 变量 V4...5 .....	497
12.11.7.9	PositionLimitsSW 变量 V4...5 .....	498

---

12.11.7.10 PositionLimitsHW 变量 V4...5 .....	500
12.11.7.11 Homing 变量 V4...5 .....	501
12.11.7.12 PositionControl 变量 V5 .....	502
12.11.7.13 FollowingError 变量 V5 .....	503
12.11.7.14 PositionMonitoring 变量 V5 .....	504
12.11.7.15 StandstillSignal 变量 V5 .....	505
12.11.7.16 StatusPositioning 变量 V4...5 .....	506
12.11.7.17 StatusDrive 变量 V5 .....	507
12.11.7.18 StatusSensor 变量 V5 .....	508
12.11.7.19 StatusBits 变量 V4...5 .....	509
12.11.7.20 ErrorBits 变量 V4...5 .....	514
12.11.7.21 ControlPanel 变量 V4...5 .....	515
12.11.7.22 内部变量 V4...5 .....	515
12.11.7.23 工艺对象变量的更新 .....	515
12.11.8 命令表工艺对象 V1...3 的变量 .....	516
12.11.9 命令表工艺对象变量 V4...5 .....	517
索引 .....	519

## 1.1 CPU S7-1200 的运动功能

TIA Portal 完美集成 CPU S7-1200 的运动控制功能，对步进电机和伺服电机进行高效控制：

- 在 TIA Portal 中对定位轴和命令表工艺对象进行组态。CPU S7-1200 使用这些工艺对象来控制用于控制驱动器的输出。
- 在用户程序中，可以通过运动控制指令来控制轴，也可以启动驱动器的运动命令。

### 参见

用于运动控制的硬件组件 (页 13)

集成定位轴工艺对象 (页 72)

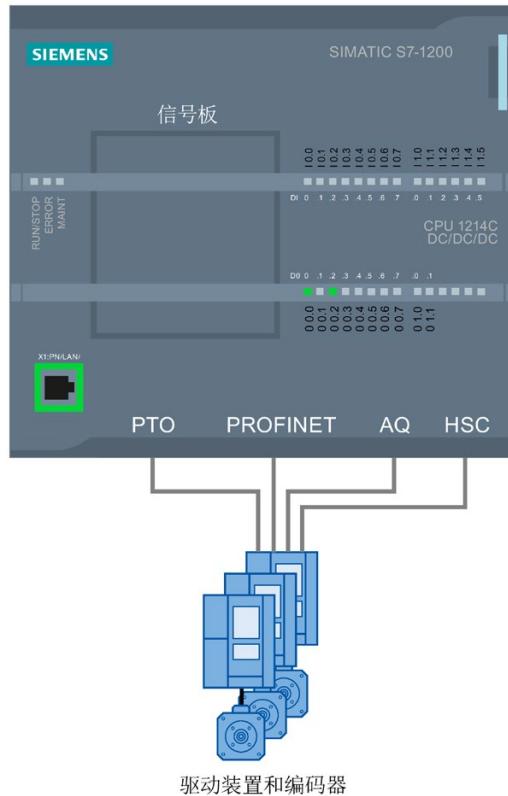
定位轴工艺对象的工具 (页 76)

使用作业表工艺对象 (页 161)

命令表工艺对象工具 (页 161)

## 1.2 用于运动控制的硬件组件

下图显示了使用 CPU S7-1200 进行运动控制应用的基本硬件配置。



### CPU S7-1200

CPU S7-1200 兼具可编程逻辑控制器的功能和用于控制驱动器运行的运动控制功能。运动控制功能负责对驱动器进行监控。

### 信号板

可以使用信号板为 CPU 添加其它输入和输出。

如果需要，还可将数字量输出用作控制驱动器的脉冲发生器输出。对于具有继电器输出的 CPU，由于继电器不支持所需的开关频率，因此无法通过板载输出来输出脉冲信号。如果要在这些 CPU 中使用 PTO (Pulse Train Output)，必须使用具有数字量输出的信号板。

如果需要，还可使用模拟量输出来控制所连接的模拟量驱动器。

## 1.2 用于运动控制的硬件组件

### PROFINET

PROFINET 接口用于在 CPU S7-1200 与编程设备之间建立在线连接。除了 CPU 的在线功能外，附加的调试和诊断功能也可用于运动控制。

PROFINET 仍支持用于连接 PROFIdrive 驱动器和编码器的 PROFIdrive 配置文件。

### 驱动装置和编码器

驱动器用于控制轴的运动。编码器提供轴的闭环位置控制的实际位置。

下表显示了驱动器和编码器可能的连接方式：

驱动器连接	轴的闭环/开环控制	编码器连接
PTO (Pulse Train Output) (带有脉冲接口的步进电机和伺服电机)	位置控制	-
模拟量输出 (AQ)	位置控制	<ul style="list-style-type: none"> <li>• 高速计数器 (HSC) 上的编码器</li> <li>• 工艺模块 (TM) 上的编码器</li> <li>• PROFINET 上的编码器</li> </ul>
PROFINET	位置控制	<ul style="list-style-type: none"> <li>• 驱动装置上的编码器</li> <li>• 高速计数器 (HSC) 上的编码器</li> <li>• 工艺模块 (TM) 上的编码器</li> <li>• PROFINET 上的编码器</li> </ul>

### CPU 固件 V4.4 的订购信息

下列订购信息适用于 TIA Portal 中当前已安装的产品系列（不含括所有已安装的硬件支持包）。

可以使用硬件支持包 (HSP) 安装新的硬件组件。安装后，硬件组件随即显示在硬件目录中。

名称	订货号
CPU 1211C DC/DC/DC	6ES7211-1AE40-0XB0
CPU 1211C AC/DC/RLY	6ES7211-1BE40-0XB0
CPU 1211C DC/DC/RLY	6ES7211-1HE40-0XB0
CPU 1212C DC/DC/DC	6ES7212-1AE40-0XB0

名称	订货号
CPU 1212C AC/DC/RLY	6ES7212-1BE40-0XB0
CPU 1212C DC/DC/RLY	6ES7212-1HE40-0XB0
CPU 1214C DC/DC/DC	6ES7214-1AG40-0XB0
CPU 1214C AC/DC/RLY	6ES7214-1BG40-0XB0
CPU 1214C DC/DC/RLY	6ES7214-1HG40-0XB0
CPU 1214FC DC/DC/DC	6ES7214-1AF40-0XB0
CPU 1214FC DC/DC/RLY	6ES7214-1HF40-0XB0
CPU 1215C DC/DC/DC	6ES7215-1AG40-0XB0
CPU 1215C AC/DC/RLY	6ES7215-1BG40-0XB0
CPU 1215C DC/DC/RLY	6ES7215-1HG40-0XB0
CPU 1215FC DC/DC/DC	6ES7215-1AF40-0XB0
CPU 1215FC DC/DC/RLY	6ES7215-1HF40-0XB0
CPU 1217C DC/DC/DC	6ES7217-1AG40-0XB0
信号板 DI4 x DC 24 V (200 kHz)	6ES7221-3BD30-0XB0
信号板 DI4 x DC 5 V (200 kHz)	6ES7221-3AD30-0XB0
信号板 DQ4 x DC 24 V (200 kHz)	6ES7222-1BD30-0XB0
信号板 DQ4 x DC 5 V (200 kHz)	6ES7222-1AD30-0XB0
信号板 DI2/DQ2 x DC 24 V (20 kHz)	6ES7223-0BD30-0XB0
信号板 DI2/DQ2 x DC 24 V (200 kHz)	6ES7223-3BD30-0XB0
信号板 DI2/DQ2 x DC 5 V (200 kHz)	6ES7223-3AD30-0XB0
信号板 AQ1 x 12 位 ( $\pm 10$ V, 0 到 20 mA)	6ES7232-4HA30-0XB0

## 参见

[CPU S7-1200 的运动功能 \(页 12\)](#)

[运动控制相关的 CPU 输出 \(页 16\)](#)

# 使用 S7-1200 运动控制的基础知识

2

## 2.1 PTO 上的步进电机

### 2.1.1 运动控制相关的 CPU 输出

可用驱动器的数目取决于 PTO（脉冲串输出）数目以及可用的脉冲发生器输出数目。

下表提供了有关相关依赖性的信息：

#### 最大 PTO 数

每个配备工艺版本 V4 的 CPU 都可使用 4 个 PTO。也就是说最多可以控制 4 个驱动器。

#### PTO 的信号类型

根据 PTO 的信号类型，每个 PTO（驱动器）需要 1 到 2 个脉冲发生器输出：

信号类型	脉冲发生器输出数目
脉冲 A 和方向 B（禁用方向输出） <sup>1</sup>	1
脉冲 A 和方向 B <sup>1</sup>	2
时钟增加 A 和时钟减少 B	2
A/B 相移	2
A/B 相移，四相位	2

<sup>1</sup> 方向输出必须为板载输出或位于信号板中。

#### 可用的脉冲发生器输出和频率范围

继电器型 CPU 仅可访问信号板的脉冲发生器输出。

根据 CPU, 脉冲信号发生器输出 Q0.0 到 Q1.1 可与以下频率范围配合使用:

CPU	Q0.0	Q0.1	Q0.2	Q0.3	Q0.4	Q0.5	Q0.6	Q0.7	Q1.0	Q1.1
1211 (DC/DC/DC)	100 kHz	100 kHz	100 kHz	100 kHz	-	-	-	-	-	-
1212 (DC/DC/DC)	100 kHz	100 kHz	100 kHz	100 kHz	20 kHz	20 kHz	-	-	-	-
1214(F) (DC/DC/DC)	100 kHz	100 kHz	100 kHz	100 kHz	20 kHz	20 kHz	20 kHz	20 kHz	20 kHz	20 kHz
1215(F) (DC/DC/DC)	100 kHz	100 kHz	100 kHz	100 kHz	20 kHz	20 kHz	20 kHz	20 kHz	20 kHz	20 kHz
1217 (DC/DC/DC)	1 MHz	1 MHz	1 MHz	1 MHz	100 kHz	100 kHz	100 kHz	100 kHz	100 kHz	100 kHz

根据信号板, 脉冲信号发生器输出 Qx.0 到 Qx.3 可与以下频率范围配合使用:

信号板	Qx.0	Qx.1	Qx.2	Qx.3	-	-	-	-	-	-
DI2/DQ2 x DC24V 20kHz	20 kHz	20 kHz	-	-	-	-	-	-	-	-
DI2/DQ2 x DC24V 200kHz	200 kHz	200 kHz	-	-	-	-	-	-	-	-
DQ4 x DC24V 200kHz	200 kHz	200 kHz	200 kHz	200 kHz	-	-	-	-	-	-
DI2/DQ2 x DC5V 200kHz	200 kHz	200 kHz	-	-	-	-	-	-	-	-
DQ4 x DC5V 200kHz	200 kHz	200 kHz	200 kHz	200 kHz	-	-	-	-	-	-

## 2.1 PTO 上的步进电机

最低频率始终为 1 Hz。

可将脉冲发生器输出自由分配给 PTO。

### 说明

如果按照信号类型使用具有不同频率范围的脉冲发生器输出，那么每种情况中都要使用最低频率。

但信号类型“脉冲 A 和方向 B”除外。采用此类信号类型时，始终使用脉冲发生器输出的频率范围。

### 说明

#### 通过过程映像访问脉冲发生器输出

如果已选择 PTO (Pulse Train Output) 并将其分配给某个轴，固件将通过相应的脉冲发生器和方向输出接管控制。

在实现上述控制功能接管后，将断开过程映像和 I/O 输出间的连接。虽然用户可通过用户程序或监视表写入脉冲发生器和方向输出的过程映像，但所写入的内容不会传送到 I/O 输出。因此通过用户程序或监视表格无法监视 I/O 输出。读取的信息反映过程映像中的值，与 I/O 输出的实际状态不一致。

对于 CPU 固件非永久使用的其它所有 CPU 输出，通常可以通过过程映像监控 I/O 输出的状态。

## 驱动器信号输出

对于运动控制，可以选择将驱动器接口设置为“驱动器启用”或“驱动器准备就绪”。

使用驱动器接口时，可以随意选择用于驱动器启用的数字量输出和用于“驱动器准备就绪”的数字量输入。

## 加速度/减速度限值

以下限值适用于加速度和减速度：

加速度/减速度	值
最小加速度/减速度	5.0E-3 个脉冲/秒 <sup>2</sup>
最大加速度/减速度	9.5E+9 个脉冲/秒 <sup>2</sup>

## 加加速度限值

以下限值适用于加加速度：

加加速度	值
最小加加速度	4.0E-3 个脉冲/秒 <sup>3</sup>
最大加加速度	1.0E+10 个脉冲/秒 <sup>3</sup>

## 参见

与运动控制相关的 CPU 输出（工艺版本 V1...3）(页 387)

脉冲接口的工作原理 (页 20)

信号类型与行进方向之间的关系 (页 21)

硬件和软件限位开关 (页 45)

冲击限制 (页 46)

回原点 (页 47)

用于运动控制的硬件组件 (页 13)

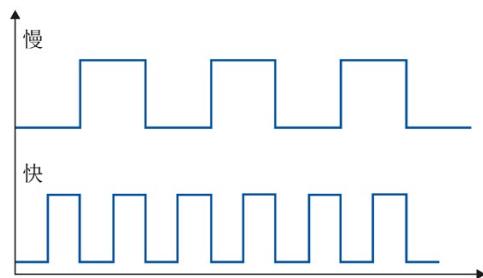
集成定位轴工艺对象 (页 72)

定位轴工艺对象的工具 (页 76)

### 2.1.2 脉冲接口的工作原理

根据步进电机的设置，每个脉冲会使步进电机移动特定角度。例如，如果将步进电机设置为每转 1000 个脉冲，则每个脉冲使电机移动  $0.36^\circ$ 。

步进电机的速度通过每单位时间的脉冲数来确定。



此处所做的说明同样适用于带有脉冲接口的伺服电机。

### 参见

运动控制相关的 CPU 输出 (页 16)

信号类型与行进方向之间的关系 (页 21)

硬件和软件限位开关 (页 45)

冲击限制 (页 46)

回原点 (页 47)

集成定位轴工艺对象 (页 72)

定位轴工艺对象的工具 (页 76)

### 2.1.3 信号类型与行进方向之间的关系

CPU 通过两个输出来输出速度和行进方向。

组态与行进方向之间的关系会因所选信号类型的不同而异。可在轴组态的“基本参数 > 常规”(Basic parameters > General) 下组态以下信号类型：

- “PTO - 脉冲 A 和方向 B”
- “PTO - 时钟增加 A 和时钟减少 B”（版本 V4 及以上版本）
- “PTO - A/B 相移”（版本 V4 及以上版本）
- “PTO - A/B 相移，四相位”（版本 V4 及以上版本）

可在轴组态的“扩展参数 > 机械”(Extended Parameters > Mechanics) 下组态方向。如果选择“反转方向”(Invert direction) 选项，则相应信号类型的以下方向逻辑会反转。

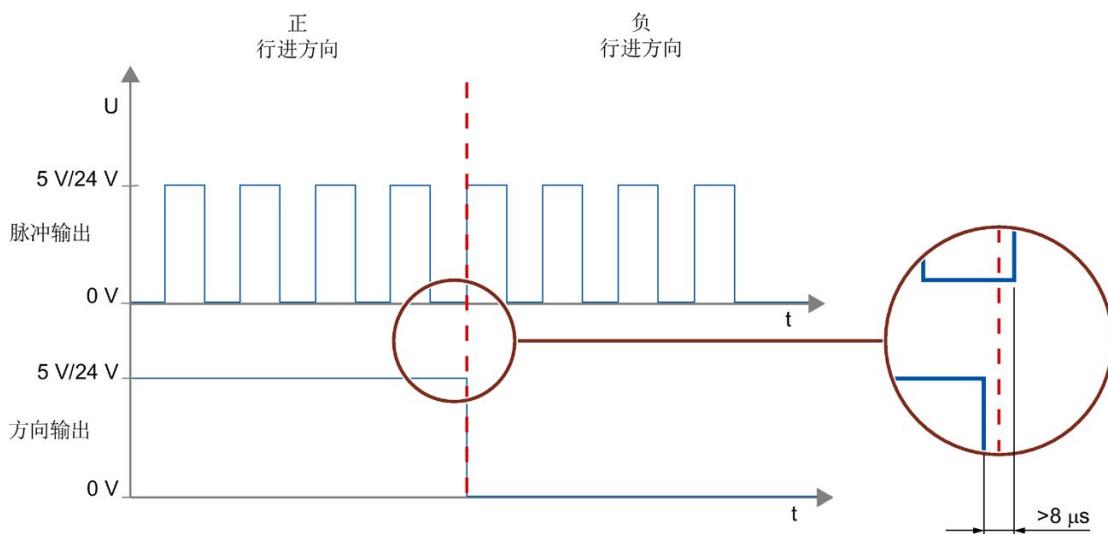
#### PTO - 脉冲 (A) 和方向 (B)

为该信号类型评估脉冲输出脉冲和方向输出电平。

脉冲通过 CPU 的脉冲输出进行输出。CPU 的方向输出指定驱动器的旋转方向：

- 方向输出上输出 5 V/24 V ⇒ 正向旋转
- 方向输出上输出 0 V ⇒ 反向旋转

指定的电压取决于所使用的硬件。各指示值并不适用于 CPU 1217 的差分输出。



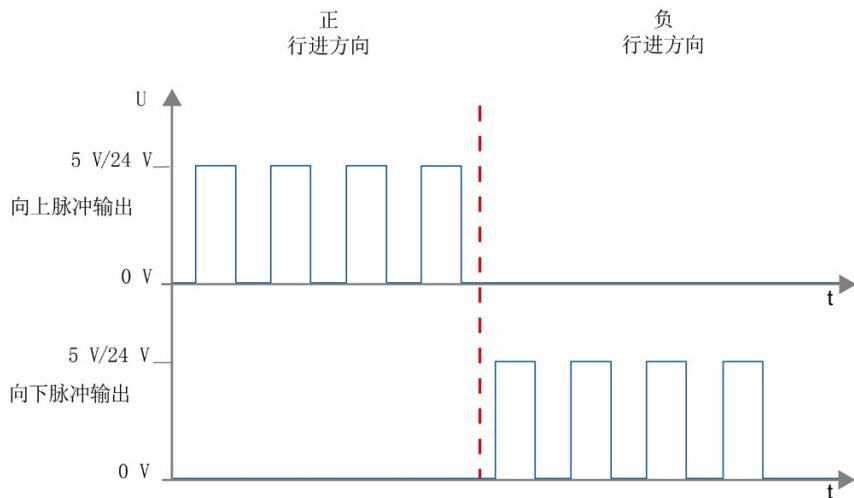
#### PTO - 时钟增加 A 和时钟减少 B (版本 V4 及以上版本)

针对该信号类型评估一路输出的脉冲。

## 2.1 PTO 上的步进电机

正转脉冲通过“脉冲输出加计数”输出；反转脉冲通过“脉冲输出减计数”输出。

指定的电压取决于所使用的硬件。各指示值并不适用于 CPU 1217 的差分输出。



### PTO - A/B 相移（版本 V4 及以上版本）

为该信号类型评估每种情况下一个输出的上升沿。

脉冲通过“信号 A”输出，相移通过“信号 B”输出。输出之间的相移定义了旋转方向：

- 信号 A 超前信号 B  $90^\circ \Rightarrow$  正转
- 信号 B 超前信号 A  $90^\circ \Rightarrow$  反转

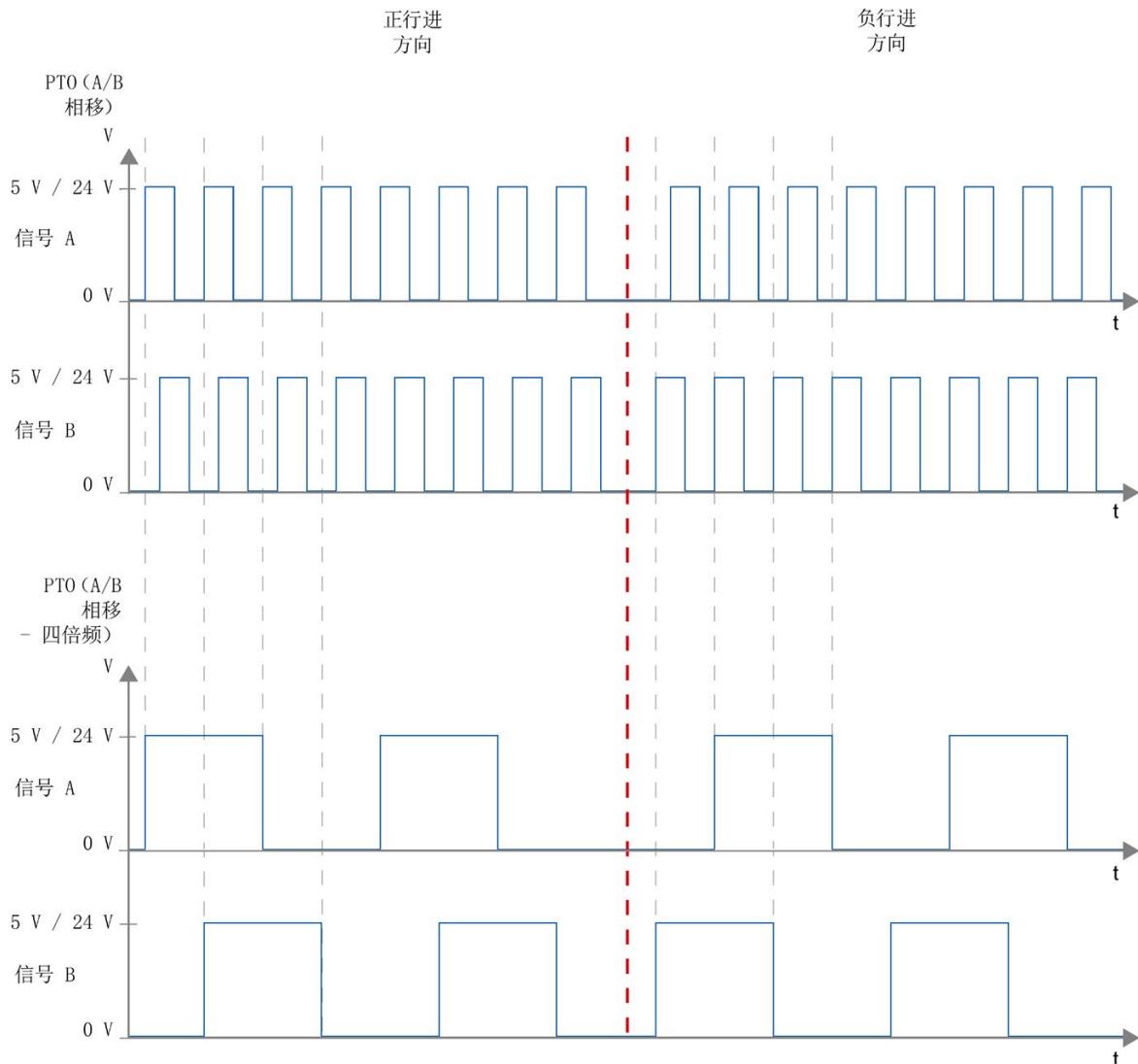
### PTO - A/B 相移, 四相位 (版本 V4 及以上版本)

为该信号类型评估两个输出的上升沿和下降沿。一个脉冲周期有四沿两相 (A 和 B)。因此, 输出中的脉冲频率会减小到四分之一。

脉冲通过“信号 A”输出, 相移通过“信号 B”输出。输出之间的相移定义了旋转方向:

- 信号 A 超前信号 B  $90^\circ \Rightarrow$  正转
- 信号 B 超前信号 A  $90^\circ \Rightarrow$  反转

指定的电压取决于所使用的硬件。各指示值并不适用于 CPU 1217 的差分输出。



## 反转方向

如果选择“反转旋转信号”(Invert rotation signal) 选项，则会反转方向逻辑：

- PTO - 脉冲 (A) 和方向 (B)
  - 方向输出上输出 0 V (低电平)  $\Rightarrow$  正向旋转
  - 方向输出上输出 5 V/24 V (高电平)  $\Rightarrow$  负向旋转

指定的电压取决于所使用的硬件。各指示电压并不适用于 CPU 1217 的差分输出。
- PTO - 时钟增加 A 和时钟减少 B

“向下脉冲输出”和“向上脉冲输出”这两个输出相互交换。
- PTO - A/B 相移

“信号 A”和“信号 B”输出相互交换。
- PTO - A/B 相位偏移量 - 四重

“信号 A”和“信号 B”输出相互交换。

## 参见

运动控制相关的 CPU 输出 (页 16)

脉冲接口的工作原理 (页 20)

硬件和软件限位开关 (页 45)

冲击限制 (页 46)

回原点 (页 47)

集成定位轴工艺对象 (页 72)

定位轴工艺对象的工具 (页 76)

## 2.2 PROFIdrive 驱动装置 / 模拟量驱动装置接口

### 2.2.1 驱动装置和编码器连接

驱动器和编辑器分配给通过 PROFIdrive/模拟驱动器接口与驱动器进行连接的定位轴。

具有 PROFIdrive 功能的驱动器可通过 PROFIdrive 报文进行连接。通过 PROFIdrive 报文，指定该设定值。

带模拟量设定值接口的驱动器可使用模拟量输出与可选启用信号连接。通过一个模拟量输出指定该设定值。

#### 连接选项

具有 PROFIdrive 功能的驱动器可通过 CPU 的 PROFINET 接口进行连接。

具有模拟量设定值接口的驱动器通过以下某种连接与 CPU 进行连接：

- 通过信号板的模拟量输出
- 板载模拟量输出
- 模拟量输出模块上的模拟量输出

可通过以下方式连接编码器：

- PROFIdrive 驱动装置上的编码器
- 工艺模块上的编码器
- PROFIdrive 编码器（直接连接 PROFINET IO）

（对于这些编码器，编码器值通常通过 PROFIdrive 报文经由 PROFIBUS 或 PROFINET 进行传送）

- 高速计数器 (HSC) 上的编码器

（通过这种连接方式，编码器信号将直接连接 HSC，并生成编码器值）根据所用的 CPU，最多可使用 6 个 HSC 编码器。

#### 最大轴数量

通过 PROFIdrive 或模拟量驱动接口，最多可控制 8 个驱动装置（具体数量与轴的仿真状态无关）。

## 2.2.2 自动传送设备中的驱动装置与编码器参数

运行时，控制器、驱动装置以及编码器中设置的驱动装置和编码器接口参考值必须相同。

速度设定值 NSET 和实际速度值 NACT 将以参考速度的百分比形式在 PROFIdrive 报文中进行传送。因此，在控制器和驱动装置中必须设置相同的速度参考值。

PROFIdrive 报文中实际值的精度也必须与在控制器和驱动器或编码器模块中设置的相同。

### 自动传送参数

对于 SINAMICS 驱动装置（V4.x 及以上版本）和 PROFIdrive 编码器（产品版本 A16 及以上版本），驱动装置和编码器参数可自动传送到 CPU 中。

对工艺对象进行（重新）初始化并（重新）启动驱动装置和 CPU 后，系统将传送相应的参数。重启驱动器或工艺对象后，将传送对驱动器组态进行的更改。

在控制器中，可通过 `<轴名称>.StatusDrive.AdaptionState = 2` 变量和 `<轴名称>.StatusSensor[1].AdaptionState = 2` 工艺对象确定参数是否传送成功。

## 参数

在 TIA Portal 中，可通过“工艺对象 > 组态 > 基本参数 > 驱动装置/编码器”(Technology object > Configuration > Basic parameters > Drive/encoder) 对控制器进行设置。

在组态或相应的硬件中进行驱动器和编码器设置。

下表对 TIA Portal 和控制器中的设置以及相应的驱动装置/编码器参数进行了比较：

TIA Portal 中的设置	工艺数据块中的 控制器变量	驱动器参数	自动传 送
<b>驱动装置</b>			
报文编号	报文的输入地址 <轴名称>.Actor.Interface.AddressIn	报文编号 P922	-
	报文的输出地址 <轴名 称>.Actor.Interface.AddressOut		
参考速度 [1/min]	<轴名 称>.Actor.DriveParameter.ReferenceSpeed	SINAMICS 变频 器：P2000	✓
电机的最大转速 [1/min]	<轴名 称>.Actor.DriveParameter.MaxSpeed	SINAMICS 变频 器：P1082	✓
驱动装置	<轴名称>.Actor.Type 0 = 模拟量驱动装置接口 1 = PROFdrive 2 = PTO (脉冲串输出)	-	-
<b>编码器</b>			
报文	<轴名 称>.Sensor[n].Interface.AddressIn	P922	-
	<轴名 称>.Sensor[n].Interface.Addressout		
编码器类型 • 线性增量式 • 线性绝对式	<轴名称>.Sensor[n].System 0: 旋转式 1: 线性	P979[1] Bit0 编码器 1 P979[11] Bit0 编码 器 2	✓

## 2.2 PROFIdrive 驱动装置 / 模拟量驱动装置接口

TIA Portal 中的设置	工艺数据块中的控制器变量	驱动器参数	自动传送
• 旋转增量式 • 旋转绝对式	<轴名称>.Sensor[n].Type 0: 增量式 1: 绝对式	P979[5] 编码器 1 P979[15] 编码器 2	-
精度, 线性编码器 在编码器铭牌上指定的网格间距, 是指线性测量系统上各个标记间的间隔距离。	<轴名称>.Sensor[n].Parameter.Resolution	P979[2] 编码器 1 P979[12] 编码器 2	√
每转增量数, 旋转编码器	<轴名称>.Sensor[n].Parameter.StepsPerRevolution	P979[2] 编码器 1 P979[12] 编码器 2	√
高精度的位数 XIST1 编码器实际周期值, 线性或旋转编码器	<轴名称>.Sensor[n].Parameter.FineResolutionXist1	P979[3] 编码器 1 P979[13] 编码器 2	√
高精度的位数 XIST2 编码器绝对值, 线性或旋转编码器	<轴名称>.Sensor[n].Parameter.FineResolutionXist2	P979[4] 编码器 1 P979[14] 编码器 2	√
差动编码器转数, 旋转式绝对值编码器	<轴名称>.Sensor[n].Parameter.DeterminableRevolutions	P979[5] 编码器 1 P979[15] 编码器 2	√

## 参见

- 组态 - 驱动器 - PTO (脉冲串输出) (页 84)
- 组态 - 驱动器 - 模拟驱动器接口 (页 86)
- 组态 - 驱动装置 - PROFIdrive (页 88)
- 组态 - 编码器 - PROFINET/PROFIBUS 上的编码器 (页 Error! Bookmark not defined.)
- 组态 - 编码器 - 高速计数器 (HSC) 上的编码器 (页 94)

### 2.2.3 PROFIdrive

PROFIdrive 是通过 PROFINET IO 连接驱动器和编码器的标准化驱动技术配置文件。支持 PROFIdrive 配置文件的驱动器都可根据 PROFIdrive 标准进行连接。

有关 PROFIdrive 的最新规范，敬请访问：

<https://www.profibus.com> (<http://www.profibus.com>)

控制器和驱动器/编码器之间通过各种 PROFIdrive 报文进行通信。每个报文均有一个标准化的结构。可根据具体应用，选择相应的报文。通过 PROFIdrive 报文，可传输控制字、状态字、设定值和实际值。

#### PROFIdrive 报文

定位轴的设定值可通过 PROFIdrive 消息帧 1、2、3 或 4 传送到驱动装置中。

编码器值既可与设定值（消息帧 3 和 4）一同经由消息帧传送，也可通过单独的编码器消息帧（消息帧 81 或消息帧 83）进行传送。

下表显示了支持用于分配驱动器和编码器的 PROFIdrive 报文：

标准报文	简要描述
1	<ul style="list-style-type: none"> <li>• 16 位速度设定值 (NSET)</li> <li>• 16 位实际速度 (NACT)</li> </ul>
2	<ul style="list-style-type: none"> <li>• 32 位速度设定值 (NSET)</li> <li>• 32 位实际速度 (NACT)</li> <li>• 设备状态</li> </ul>
3	<ul style="list-style-type: none"> <li>• 32 位速度设定值 (NSET)</li> <li>• 32 位实际速度 (NACT)</li> <li>• 1 个编码器</li> <li>• 设备状态</li> </ul>
4	<ul style="list-style-type: none"> <li>• 32 位速度设定值 (NSET)</li> <li>• 32 位实际速度 (NACT)</li> <li>• 2 个编码器</li> <li>• 设备状态</li> </ul>

编码器的标准报文	简要描述
81	<ul style="list-style-type: none"> <li>1 个编码器</li> <li>设备状态</li> </ul>
83	<ul style="list-style-type: none"> <li>32 位实际速度 (NACT)</li> <li>1 个编码器</li> <li>设备状态</li> </ul>

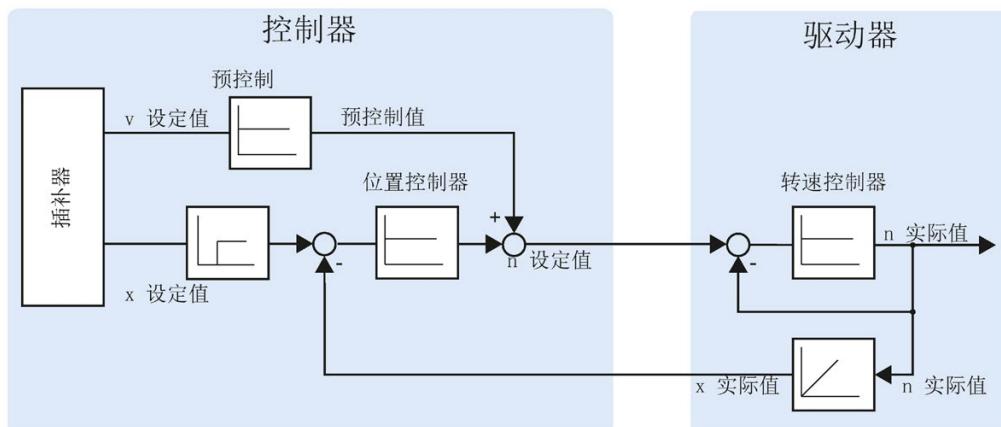
## 2.2.4 闭环控制

通过 PROFIdrive 或模拟量驱动装置接口连接的驱动装置，操作是通常基于位置控制功能。需要维修时，轴也可操作为不带位置控制功能。

位置控制器是使用预控制速度的 P 控制器。

### 控制器结构

下图显示了 S7-1200 运动控制的控制器结构：



MC-Interpolator [OB92] 计算轴的位置设定值。位置设定值与实际位置之间的差，乘以位置控制器的增益系数。计算值将添加到预控制值中，并作为驱动装置的速度设定值，通过 PROFIdrive 或模拟量输出进行输出。

编码器将记录轴的实际位置，并通过 PROFIdrive 消息帧或 HSC（高速计数器）返回控制器中。

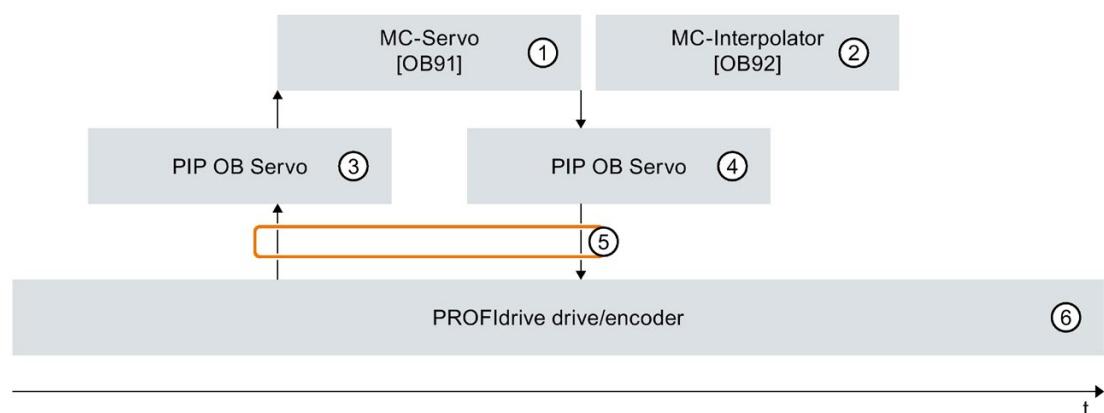
### 2.2.5 PROFIdrive 驱动装置/PROFIdrive 编码器的数据连接

通过 PROFIdrive 报文或数据块，PROFIdrive 驱动装置和 PROFIdrive 编码器可直接建立数据连接。

出于过程特定的原因，如果要影响或评估用户程序中的报文内容时，则需通过数据块建立连接。

#### 与驱动装置/编码器直接进行数据连接的工作原理

以下功能图简要说明了如何通过报文与 PROFIdrive 驱动装置和 PROFIdrive 编码器建立直接数据连接：



- ① 组织块“MC-Servo”用于计算位置控制器。  
在“MC-Servo”开始时，将读取驱动装置或编码器的输入报文 (⑥ -> ⑤ -> ③)。如果添加了组织块“MC-PreServo”，则在“MC-PreServo”开始时读取该报文。  
在“MC-Servo”结束时，将输出报文写入驱动装置或编码器中 (⑥ -> ⑤ -> ③)。如果添加了组织块“MC-PostServo”，则在“MC-PostServo”结束时写入该报文。
- ② 在每个运动应用循环中，都会在“MC-Servo”之后调用组织块“MC-Interpolator”。  
在“MC-Interpolator”中，将对运动控制指令进行评估、为下一个运动应用循环生成设定值并对工艺对象进行监视。
- ③ 输入过程映像分区“OB 伺服 PIP”将在运动应用循环中进行更新。
- ④ 输出过程映像分区“OB 伺服”将在运动应用循环中进行更新。
- ⑤ 通过控制器、驱动装置或编码器的 I/O 地址，进行报文交换。
- ⑥ PROFIdrive 驱动装置或 PROFIdrive 编码器

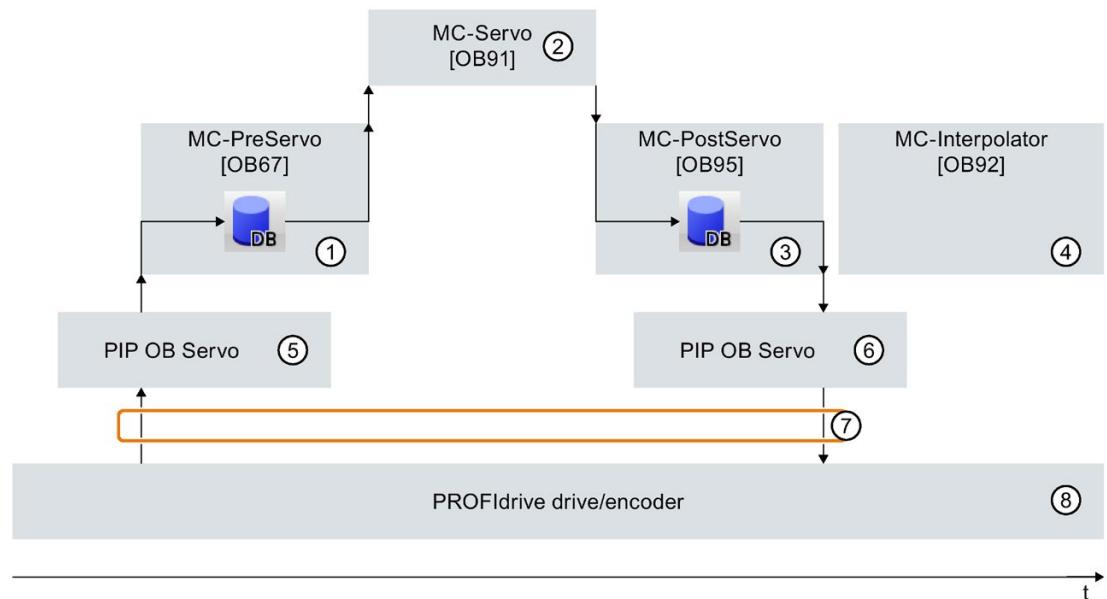
### 通过数据块建立数据连接的操作原理

以下功能图简要说明了如何通过数据块与 PROFIdrive 驱动装置和 PROFIdrive 编码器建立数据连接：在以下章节中，对这一过程进行了详细说明。

基于过程特定原因对报文内容进行影响或评估时，必须在驱动装置和编码器之间连接一个数据块并作为数据接口（参见 ① 和 ③）。

为此，可使用组织块 "MC-PreServo" 和 "MC-PostServo"，确保高品质的位置控制。

通过命令“新增块”(Add new block)，可在项目树中添加组织块 "MC-PreServo" 和 "MC-PostServo"。



- ① “MC-PreServo”在“MC-Servo”之前调用。  
在“MC-PreServo”的用户程序中，将输入报文中的内容从过程映像分区“OB 伺服 PIP”⑤ 传送到数据接口的数据块中。  
在 "MC-PreServo" 的其它用户程序中，可对该报文的输入区域进行处理或评估。
- ② 组织块“MC-Servo”用于计算位置控制器。  
在“MC-Servo”开始时，将从数据接口的数据块内读取驱动装置或编码器的输入报文 (① -> ②)。  
在“MC-Servo”结束时，将驱动装置或编码器的输出报文写入数据接口的数据块内 (② -> ③)。
- ③ “MC-PostServo”在“MC-Servo”之后调用。  
在 "MC-PostServo" 的用户程序中，可对该报文的输出区域进行处理或评估。  
在“MC-PostServo”的用户程序结束时，将输出报文的内容从数据块的数据接口传送到过程映像分区“OB 伺服 PIP”⑥ 中。
- ④ 在每个运动应用循环中，都会在“MC-PostServo”之后调用组织块“MC-Interpolator”。  
在“MC-Interpolator”中，将对运动控制指令进行评估、为下一个运动应用循环生成设定值并对工艺对象进行监视。
- ⑤ 输入过程映像分区“OB 伺服 PIP”将在运动应用循环中进行更新。
- ⑥ 输出过程映像分区“OB 伺服”将在运动应用循环中进行更新。
- ⑦ 通过控制器、驱动装置或编码器的 I/O 地址，进行报文交换。
- ⑧ PROFIdrive 驱动装置或 PROFIdrive 编码器

### 通过数据块进行数据连接的基本操作过程

要通过数据块进行数据连接，请按以下步骤操作：PROFIdrive 驱动装置和编码器的数据连接可分别组态。

## 创建用于数据连接的数据块

为此，用户需创建数据连接的数据块。该数据块中需包含数据类型为“PD\_TELx”的数据结构，以进行数据连接。其中，“x”为设备组态中，所组态的驱动装置或编码器的报文编号。

要创建数据块，请按以下步骤操作：

1. 创建一个类型为“Global DB”的新数据块。
2. 在项目树中选择该数据块，然后选择快捷菜单命令“属性”(Properties)。
3. 禁用“属性”(Attributes) 下的以下属性，并通过“确定”(OK) 接受更改：
  - “仅存储在装载存储器中”(Only store in load memory)
  - “数据块在设备中受写保护”(Data block write-protected in the device)
  - “优化块访问”(Optimized block access) (V7.0 以下版本的工艺对象)
4. 在块编辑器中打开该数据块。
5. 在块编辑器中，插入一个“PD\_TELx”类型的变量。
6. 在轴的组态中进行使用前，需先对该数据连接的数据块进行编译。

该变量中包含有报文输入区域的“Input”变量结构，以及报文输出区域的“Output”变量结构。

### 说明

“Input”和“Output”与闭环位置控制视图有关。例如，该输入区域中包含驱动装置的实际值，而输出区域中则包含驱动装置的设定值。

数据块可能包含多个轴与编码器的数据结构和其他内容。

## 通过数据块组态数据连接

要完成轴组态，请按以下步骤操作：

1. 打开组态窗口“硬件接口 > 驱动器”(Hardware interface > Drive) 或“硬件接口 > 编码器”(Hardware interface > Encoder)。
2. 在数据块下拉列表中，选择“数据块”(Data block)：
3. 在“数据块”(Data block) 字段中，选择先前创建的数据块。  
打开该数据块，选择为驱动装置或编码器定义的变量名称。

## 添加一个报文访问的 PLC 变量

要访问报文的输入和输出区域，需创建以下 PLC 变量：

对于输入区域的 PLC 变量，请按以下步骤操作：

1. 在项目树中打开“PLC 变量”(PLC tags) 文件夹并显示所有变量。
2. 添加一个新的变量并指定一个唯一名称。例如，可以是轴/编码器、报文类型和地址区名称的组合。
3. 在“数据类型”(Data type) 列中，以文本形式输入类型“PD\_TELx\_IN”。
4. 在“地址”(Address) 列中输入驱动设备/编码器的报文输入地址。  
可以在驱动设备或编码器组态中找到该地址。

对于输出区域的 PLC 变量，请遵循相同的操作步骤，选择“PD\_TELx\_OUT”数据类型，并将驱动装置/编码器的报文输出地址作为变量的地址。

## 编程 MC-PreServo 和 MC-PostServo

### MC-PreServo

"MC-PreServo" 的用户程序将读取该报文的输入区域，并传送到该数据连接的数据块中。

为 "MC-PreServo" 用户程序中该数据块的 "Input" 变量结构，指定输入区域先前定义的 PLC 变量。

将该数据块 "Input" 变量结构中的数据传送到 "MC-Servo" 中，在 "MC\_Servo" 内进行处理前，可对其进行编辑修改。

### MC-PostServo

处理完成后，"MC-Servo" 将报文的输出区域传送到该数据块的变量结构 "Output" 中。

数据块中变量结构 "Output" 的内容需写入 "MC-PostServo" 用户程序的报文输出地址内。

将数据块的变量结构 "Output" 指定给 "MC-PostServo" 用户程序中输出区域之前定义的 PLC 变量。

如果要修改输出区域，则必须在分配指令前完成此操作。

#### 注意

##### 机器损坏

对驱动装置和编码器报文操作不当，将导致驱动装置运动异常。

请检查用户程序中驱动装置和编码器连接的一致性问题。

有关使用 MC-PreServo 和 MC-PostServo 的应用示例，敬请访问：

<https://support.industry.siemens.com/cs/document/109741575>

(<https://support.industry.siemens.com/cs/document/109741575>)

## 参见

PROFIdrive 帧 (页 227)

组态 - 驱动装置 - PROFIdrive (页 88)

组态 - 编码器 - PROFINET/PROFIBUS 上的编码器 (页 Error! Bookmark not defined.)

带模拟量驱动接口的数据连接驱动装置 (页 37)

用于运动控制的组织块 (页 39)

## 2.2.6 带模拟量驱动接口的数据连接驱动装置

带模拟量驱动接口的数据连接驱动装置，也可通过数据块进行数据连接。

出于过程特定的原因，如果要调整用户程序中的模拟量设定值，则需通过数据块建立连接。

### 通过数据块建立数据连接的操作原理

在 MC-Servo [OB91] 位置控制结束处，模拟量驱动装置的设定值将写入所指定的模拟量输出中。

出于过程特定的原因，如果要调整模拟量设定值，则需通过数据块连接数据接口。

模拟量驱动装置的设定值可通过数据块中的组织块 MC-PostServo [OB95] 进行编辑，并写入该 I/O 地址中。

在 MC-Servo 后调用 MC-PostServo。用户可对 MC-PostServo 组织块进行编程，编程时需使用“添加新块”(Add new block) 指令添加组织块。

### 标准操作过程

要通过数据块进行数据连接，请按以下步骤操作：对于带有模拟量驱动接口的驱动装置和 PROFdrive 编码器，可分别组态数据连接。有关 PROFdrive 编码器数据连接的信息，请参见“PROFdrive 驱动装置/PROFdrive 编码器的数据连接 (页 31)”部分。

### 创建用于数据连接的数据块

必须在用户侧创建数据块。

要创建数据块，请按以下步骤操作：

1. 创建一个类型为“Global DB”的新数据块。
2. 在项目树中选择该数据块，然后在快捷菜单中选择“属性”(Properties)。
3. 禁用“属性”(Attributes) 下的以下属性，并通过“确定”(OK) 接受更改：
  - “仅存储在装载存储器中”(Only store in load memory)
  - “数据块在设备中受写保护”(Data block write-protected in the device)
  - “优化块访问”(Optimized block access) (V7.0 以下版本的工艺对象)
4. 在块编辑器中打开该数据块。
5. 在块编辑器中，插入一个“WORD”数据类型的变量。
6. 在轴的组态中进行使用前，需先对该数据连接的数据块进行编译。

### 通过数据块组态数据连接

进行以下组态设置（在“基本参数 > 常规组态窗口”(Basic parameters > General configuration window) 中需选择“模拟量驱动接口”(Analog drive connection)）：

1. 打开组态窗口“基本参数 > 驱动装置”(Basic parameters > Drive)。
2. 在“模拟量输出”(Analog output) 对话框中，选择预定义的数据块变量。

### 在过程映像 TPA OB Servo 中，设置模拟量输出地址

为了确保控制质量，模拟量输出的地址区必须位于过程映像“TPA OB Servo”中。

具体操作，如下所示：

1. 在设备组态中，打开模拟量输出模块。
2. 打开“常规”(General) 选项卡。
3. 选择“I/O 地址”(I/O addresses)。
4. 选择组织块“MC-Servo”。“TPA OB Servo”将自动选为过程映像。
5. 在“I/O 变量”(I/O variables) 选项卡中，未模拟量输出指定一个变量名。

### 编程 MC-PostServo

将数据块的这一变量指定给 MC-PostServo 用户程序中模拟量输出的变量。

在 MC-PostServo 结束时，“TPA OB Servo”的输出区域将写入 I/O 中。



警告

驱动装置设定值设置不正确，可能会导致人员和设备伤害。

为防对人员或机器造成危害，需采取相应的预防措施。

## 2.2.7 过程响应

### 2.2.7.1 用于运动控制的组织块

#### 说明

创建带有 PROFdrive 驱动装置或模拟量驱动接口的工艺对象“定位轴”(Positioning axis)时，系统将自动创建用于处理工艺对象的组织块。工艺对象的运动控制功能可创建自己的执行级别，并根据运动控制应用循环进行调用。

创建下列块：

- MC-Servo [OB91]  
位置控制器的计算
- MC-Interpolator [OB92]  
评估运动控制指令、生成设定值和监控功能

也可选择继续使用以下组织块：

- MC-PreServo [OB67]  
例如：从驱动系统准备报文内容。
- MC-PostServo [OB95]  
例如：为驱动系统准备设定值。

与 MC-PreServo [OB67] 和 MC-PostServo [OB95] 相比，组织块 MC-Servo [OB91] 和 MC-Interpolator [OB92] 受到保护（专有技术保护）。无法查看或更改程序代码。

2 个组织块彼此之间出现的频率关系始终为 1:1。MC-Servo [OB91] 总是在 MC-Interpolator [OB92] 之前执行。

根据控制质量要求与系统负载的不同，设定相应的运动控制应用循环和组织块的优先级。

## 运动控制应用循环

在组织块属性的“常规 > 循环时间”(General > Cycle Time) 中，可设置调用 MC-Servo [OB91] 的运动控制应用循环。

MC-Servo [OB91] 以安装指定的“应用循环”周期进行循环调用。

所选的运动控制应用循环时间必须足够长，以确保可在一个循环中完成所有工艺对象的处理操作。如果工艺对象的处理时间长于应用循环，则将发生溢出 (页 42)。

为了避免 CPU 上程序执行中断，可根据所使用的轴数量设置运动控制应用循环，如下所示：

$$\text{运动控制应用循环时间} = (\text{位置控制轴的数量} \times 2 \text{ ms}) + 2 \text{ ms}$$

在下表中，举例说明了不同位置控制轴数量对应的运动控制应用循环时间：

轴数量	运动控制应用循环
1	4 ms
2	6 ms
4	10 ms
8	18 ms

对于 SINAMICS，以下公示同样适用：

- 运动控制应用循环时间 (MC-Servo)  $\geq$  SINAMICS 驱动装置过程映像 (参数 P2048)  $\geq$  总线时钟周期

待选择的所有时间之间应相互为整数倍。

## 优先级

在组织块的属性中“常规 > 属性 > 优先级”(General > Properties > Priority) 中，可以按需设定组织块的优先级：

- MC-Servo [OB91]
  - 优先级 17 至 26 (默认值 25)
- MC-Interpolator [OB92]
  - 优先级 16 至 25 (默认值 24)

MC-Servo [OB91] 的优先级必须至少比 MC-Interpolator [OB92] 的优先级高 1 级。

### MC-PreServo [OB67] 和 MC-PostServo [OB95]

组织块 MC-PreServo [OB67] 和 MC-PostServo [OB95] 均可编程，并在组态的应用程序循环中调用。MC-PreServo [OB67] 在 MC-Servo [OB91] 之前直接调用。MC-PostServo [OB95] 在 MC-Servo [OB91] 之后直接调用。

### 参见

PROFdrive 驱动装置/PROFdrive 编码器的数据连接 (页 31)

### 2.2.7.2 过程映像分区“OB 伺服 PIP”

为了对控制进行优化时，需要将运动控制系统使用的所有 I/O 模块（如，驱动装置、工艺模块，数字量和模拟量输入/输出模块）指定给过程映像分区“OB 伺服 PIP”。这样，I/O 模块即可与工艺对象同时处理。

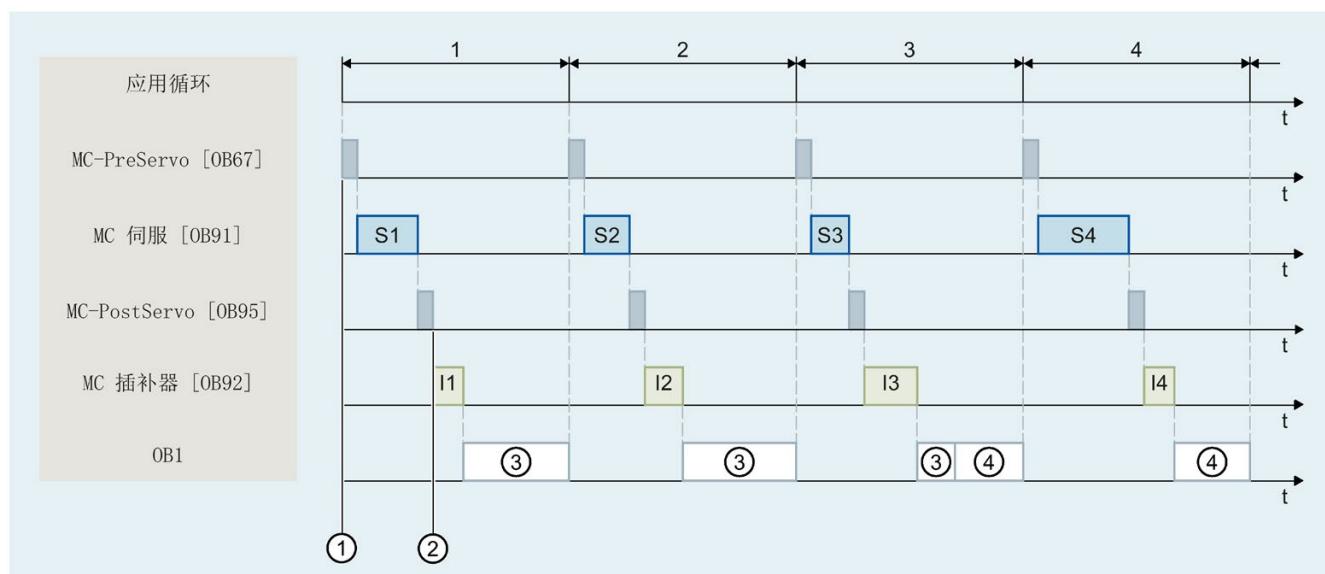
### 2.2.7.3 操作顺序和超时

执行运动控制功能时，将在每个应用循环中调用和执行组织块 MC-Servo [OB91] 和 MC-Interpolator [OB92]（CPU 的操作模式为 STOP，同样执行）。余下的循环时间用于处理用户程序。

要实现无错程序执行，应遵循下列规则：

- 在每个应用循环中，都必须启动并完全执行 MC-Servo [OB91]。
- 在每个应用循环中，都必须至少启动相关的 MC-Interpolator [OB92]。

下图示例描述了用于运行组织块 OB1 的无故障操作顺序：



- ① “TPA OB Servo”输入
- ② “TPA OB Servo”输出
- ③ 首次 OB1 循环
- ④ 第二次 OB1 循环

## 溢出

如果未遵循设置的应用循环（例如由于应用循环过短），则可能发生溢出。

MC-Servo [OB91]、MC-Interpolator [OB92]，MC\_PreServo [OB67] 和 MC\_PostServo [OB95] 的溢出将输入 CPU 的诊断缓冲区中，并导致 CPU 转入 STOP 模式。

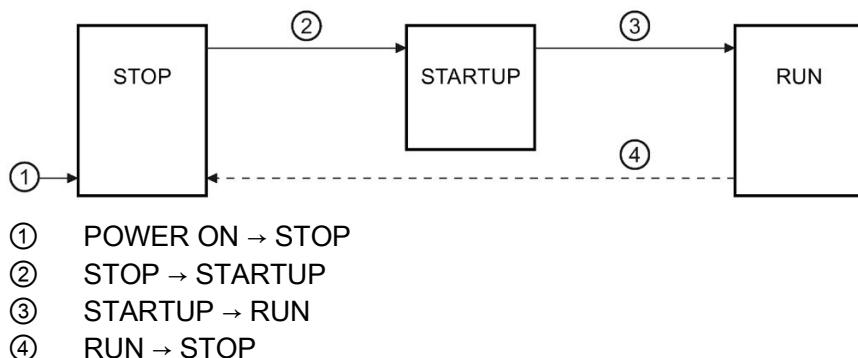
MC-PreServo、MC-Servo、MC-PostServo 和 MC-Interpolator 已停止。必要时，可通过时间错误 OB (OB80) 对诊断缓冲区的输入条目进行评估。

### 2.2.7.4 操作模式

本节描述各个操作模式下以及模式转换期间的运动控制特性。有关操作模式的一般性描述，请参见《系统手册 S7-1200》。

## 操作模式和转换

CPU 有 3 个操作模式：STOP、STARTUP 和 RUN。下图描述了这些操作模式和操作模式转换：



## STOP 模式

在 STOP 模式下，不执行用户程序，所有过程输出均被禁用。因此，不执行任何运动控制作业。

并更新位置控制轴的工艺对象数据块。

## STARTUP 模式

CPU 开始循环地执行用户程序之前，会运行启动 OB 一次。

在启动 (STARTUP) 模式下，过程输出被禁用。运动控制作业被拒绝。

并更新位置控制轴的工艺对象数据块。

## RUN 模式

用户程序在 RUN 模式执行。

在 RUN 模式下，将循环调用并处理 OB1 中编写的运动控制指令（也可在其它执行层运行）。

工艺对象数据块被更新。

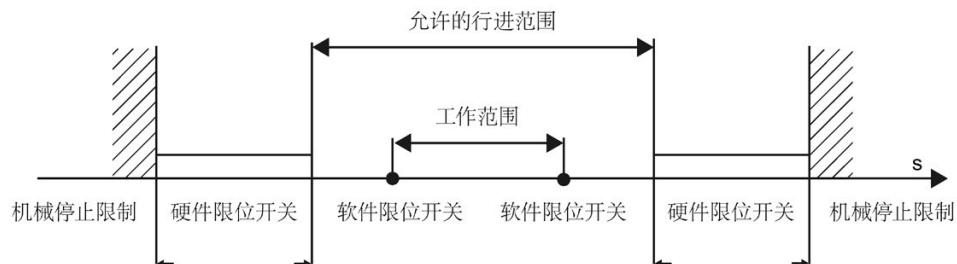
## 操作模式的转换

下表列出了操作模式转换过程中的运动控制特性。

操作模式转换	特性
POWER ON → STOP	CPU 重启工艺对象。工艺对象使用装载存储器中的值重新初始化。
STOP → STARTUP	工艺对象将初始化为 CPU 中的起始值。
STARTUP → RUN	过程输出被启用。
RUN → STOP	CPU 从 RUN 模式变为 STOP 模式时，将根据错误响应“取消启用”，禁用所有的工艺对象。运行中的运动控制作业被终止。

## 2.3 硬件和软件限位开关

硬限位开关和软限位开关用于限制定位轴工艺对象的“允许行进范围”和“工作范围”。这两者的相互关系如下图所示：



硬限位开关是限制轴的最大“允许行进范围”的限位开关。硬限位开关是物理开关元件，必须与 CPU 中具有中断功能的输入相连接。

软限位开关将限制轴的“工作范围”。它们应位于限制行进范围的相关硬限位开关的内侧。由于软限位开关的位置可以灵活设置，因此可根据当前的运行轨迹和具体要求调整轴的工作范围。与硬限位开关不同，软限位开关只通过软件来实现，而无需借助自身的开关元件。

在组态中或用户程序中使用硬件和软限位开关之前，必须先事先将其激活。只有在轴回原点之后，才可以激活软限位开关。

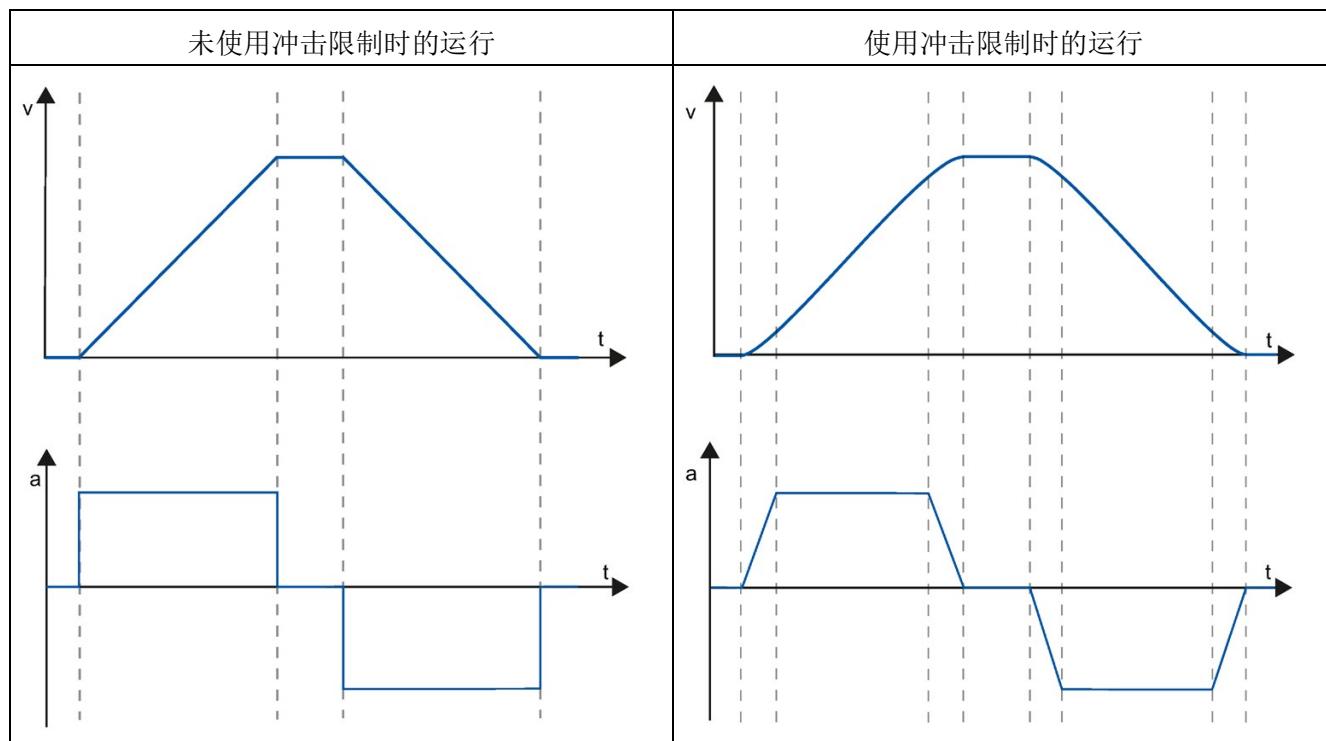
### 参见

- 运动控制相关的 CPU 输出 (页 16)
- 脉冲接口的工作原理 (页 20)
- 信号类型与行进方向之间的关系 (页 21)
- 冲击限制 (页 46)
- 回原点 (页 47)
- 集成定位轴工艺对象 (页 72)
- 定位轴工艺对象的工具 (页 76)
- 位置限制 (页 101)

## 2.4 冲击限制

### 2.4.1 冲击限制

利用冲击限制，可以降低在加速和减速斜坡运行期间施加到机械上的应力。当冲击限制器处于激活状态时，加速度和减速度的值不会突然改变，而是逐渐增大和减小的。下图显示了不使用和使用冲击限制时的速度和加速度曲线。



使用冲击限制可以产生“平滑”的轴运动速度轨迹。例如，这可以确保传送带的软启动和软制动。

### 参见

- 使用冲击限制时轴的行为 (页 113)
- 运动控制相关的 CPU 输出 (页 16)
- 脉冲接口的工作原理 (页 20)
- 信号类型与行进方向之间的关系 (页 21)
- 硬件和软件限位开关 (页 45)
- 回原点 (页 47)
- 集成定位轴工艺对象 (页 72)
- 定位轴工艺对象的工具 (页 76)

## 2.5 回原点

回原点是指使工艺对象的轴坐标与驱动器的实际物理位置相匹配。对于位置控制的轴，位置的输入与显示完全参考轴的坐标。因此，轴坐标必需与实际情形相一致。如果要确保通过驱动器也能准确到达轴的绝对目标位置，上述步骤必不可缺。

在 S7-1200 CPU 中，使用运动控制指令“MC\_Home”执行轴回原点。“已回原点”(Homed)状态将显示在工艺对象 <轴名称>.StatusBits.HomingDone 的变量中。回原点模式有：

### 回原点模式

- 主动回原点

在主动回原点模式下，运动控制指令“MC\_Home”将执行所需要的参考点逼近。检测到回原点开关时，将根据组态使轴回原点。同时终止当前的行进运动。

- 被动回原点

被动回原点期间，运动控制指令“MC\_Home”不会执行任何回原点运动。用户需通过其它运动控制指令，执行这一步骤中所需的行进移动。检测到回原点开关时，将根据组态使轴回原点。被动回原点启动时，不会中止当前的行进运动。

- 绝对式直接回原点

轴位置的设置与回原点开关无关。同时终止当前的行进运动。立即将运动控制指令“MC\_Home”中输入参数“Position”的值，设置为轴的参考点。

- 相对式直接回原点

轴位置的设置与回原点开关无关。同时终止当前的行进运动。以下语句适用于回到原点后轴的定位：

新的轴位置 = 当前轴位置 + 指令“MC\_Home”中“Position”参数的值。

## 复位“已回原点”状态

在以下条件下，将复位工艺对象（<轴名>.StatusBits.HomingDone）的“已回原点”状态：

- 通过 PTO (Pulse Train Output) 连接驱动装置：
  - 启动“MC\_Home”命令，进行主动回原点  
(成功完成回原点操作之后，“已回原点”状态将再次置位。)
  - 通过“MC\_Power”运动控制指令，禁用轴
  - 在自动模式和手动控制之间切换
  - CPU 关闭 -> 上电后
  - CPU 重新启动后 (RUN-STOP -> STOP-RUN)
- 带增量实际值的工艺对象：
  - 启动“MC\_Home”命令，进行主动回原点  
(成功完成回原点操作之后，“已回原点”状态将再次置位。)
  - 编码器系统出错或编码器故障
  - 重新启动工艺对象
  - CPU 关闭 → 上电后
  - 存储器复位
  - 修改编码器组态
- 带绝对实际值的工艺对象：
  - 传感器系统出错/编码器故障
  - 更换 CPU
  - 修改编码器组态
  - 将 CPU 恢复为出厂设置
  - 将其它项目传送到控制器

## 参见

- 运动控制相关的 CPU 输出 (页 16)
- 脉冲接口的工作原理 (页 20)
- 信号类型与行进方向之间的关系 (页 21)
- 硬件和软件限位开关 (页 45)
- 冲击限制 (页 46)
- 集成定位轴工艺对象 (页 72)
- 定位轴工艺对象的工具 (页 76)
- 回原点 (自定位轴工艺对象 V2 起) (页 117)

# 运动控制使用指南

# 3

该指南介绍了使用 CPU S7-1200 进行运动控制所需的基本步骤。

## 要求

要使用定位轴工艺对象，必须创建一个含有 CPU S7-1200 的项目。

## 步骤

要使用 CPU S7-1200 进行运动控制，并按指定的顺序执行以下步骤。为此，请点击以下链接：

1. 添加一个定位轴工艺对象 (页 79)
2. 使用组态对话框 (页 80)
3. 下载到 CPU (页 186)
4. 在调试窗口中对轴执行功能测试 (页 188)
5. 编程 (页 197)
6. 对轴控制执行诊断 (页 221)

# 使用版本

## 4.1 版本概述

S7-1200 运动控制相关版本之间的关系可在下表中找到：

### 工艺版本

在“指令”(Instructions) 任务卡中的“工艺 > 运动控制”(Technology > Motion Control) 文件夹和“工艺对象 > 添加新对象”(Technology object > Add new object) 对话框中，可检查当前选定的工艺版本。

在“指令”(Instructions) 任务卡的“工艺 > 运动控制”(Technology > Motion Control) 文件夹中，选择工艺版本。

如果将相关版本的工艺对象添加到“添加新对象”(Add new object) 对话框，则工艺版本也会更改。

---

### 说明

相关工艺版本的选择也会影响运动控制指令版本（任务卡）。

工艺对象和运动控制指令仅能在编译时或“下载到设备”时才会转换为选定版本。

---

### 工艺对象版本

可以在巡视窗口的“属性 > 常规 > 信息”(Properties > General > Information) 下的“版本”(Version) 框中检查工艺对象的版本。

### 运动控制指令版本

可以在巡视窗口的“属性 > 常规 > 信息”(Properties > General > Information) 下的“版本”(Version) 框中检查运动控制指令版本。

如果使用的运动控制指令版本与下面的兼容性列表不符，相关的运动控制指令将会突出显示在程序编辑器中。

## 4.1 版本概述

### 兼容性列表

工艺	CPU	工艺对象	运动控制指令
V7.0 创新: <ul style="list-style-type: none"><li>驱动器连接 PTO 的分段时间</li><li>离线数据调整驱动器和编码器值</li><li>Behaviour_Gx_XIST1</li></ul>	V4.4	定位轴 V7.0 命令表 V7.0	MC_Power V7.0 MC_Reset V7.0 MC_Home V7.0 MC_Stop V7.0 MC_MoveAbsolute V7.0 MC_MoveRelative V7.0 MC_MoveVelocity V7.0 MC_MoveJog V7.0 MC_CommandTable V7.0 MC_ChangeDynamic V7.0 MC_ReadParam V7.0 MC_WriteParam V7.0

工艺		CPU	工艺对象	运动控制指令
V6.0	<p>创新：</p> <ul style="list-style-type: none"> <li>• MC-PreServo</li> <li>• MC-PostServo</li> <li>• 直接连接 SINAMICS 驱动装置或通过数据块进行数据连接</li> <li>• 直接连接带有模拟量驱动接口的驱动装置 模拟量输出，或通过数据块进行数据连接</li> <li>• 传送设备中的驱动装置和编码器参数到 PROFIdrive 驱动装置和编码器中。</li> <li>• 维修时，无需位置控制功能，即可移动位置控制驱动装置</li> <li>• 仿真位置控制器驱动装置，而无需组态或使用现有硬件</li> <li>• 组态硬限位开关时的级别选择</li> <li>• 支持 PROFIdrive Telegram 4</li> </ul>	V4.2 V4.3	定位轴 V6.0 命令表 V6.0	MC_Power V6.0 MC_Reset V6.0 MC_Home V6.0 MC_Halt V6.0 MC_MoveAbsolute V6.0 MC_MoveRelative V6.0 MC_MoveVelocity V6.0 MC_MoveJog V6.0 MC_CommandTable V6.0 MC_ChangeDynamic V6.0 MC_ReadParam V6.0 MC_WriteParam V6.0

## 4.1 版本概述

工艺	CPU	工艺对象	运动控制指令
V5.0 创新: • 通过 PROFIdrive 连接驱动装置 • 模拟驱动装置接口 • PROFIdrive / 模拟量驱动装置接口的位置控制 • PROFIdrive / 模拟量驱动装置接口的定位监控 • MC-Servo [OB91] • MC-Interpolator [OB92]	V4.1	定位轴 V5.0 命令表 V5.0	MC_Power V5.0 MC_Reset V5.0 MC_Home V5.0 MC_Halt V5.0 MC_MoveAbsolute V5.0 MC_MoveRelative V5.0 MC_MoveVelocity V5.0 MC_MoveJog V5.0 MC_CommandTable V5.0 MC_ChangeDynamic V5.0 MC_ReadParam V5.0 MC_WriteParam V5.0

工艺		CPU	工艺对象	运动控制指令
V4.0	<p>创新：</p> <ul style="list-style-type: none"><li>• MC_ReadParam</li><li>• MC_WriteParam</li><li>• S7-1200 和 S7-1500 运动控制工艺数据块的标准化。</li></ul>	V4.0	定位轴 V4.0 命令表 V4.0	MC_Power V4.0 MC_Reset V4.0 MC_Home V4.0 MC_Halt V4.0 MC_MoveAbsolute V4.0 MC_MoveRelative V4.0 MC_MoveVelocity V4.0 MC_MoveJog V4.0 MC_CommandTable V4.0 MC_ChangeDynamic V4.0 MC_ReadParam V4.0 MC_WriteParam V4.0

#### 4.1 版本概述

工艺		CPU	工艺对象	运动控制指令
V3.0	创新： 在 RUN 模式下 加载	V2.2 V3.0 V4.0	轴 V3.0 命令表 V3.0	MC_Power V3.0 MC_Reset V3.0 MC_Home V3.0 MC_Stop V3.0 MC_MoveAbsolute V3.0 MC_MoveRelative V3.0 MC_MoveVelocity V3.0 MC_MoveJog V3.0 MC_CommandTabl e V3.0 MC_ChangeDynam ic V3.0

工艺		CPU	工艺对象	运动控制指令
V2.0	创新： • 加加速度限值 • 命令表 • MC_ChangeDynamic	V2.1 V2.2 V3.0	轴 V2.0 命令表 V2.0	MC_Power V2.0 MC_Reset V2.0 MC_Home V2.0 MC_Halt V2.0 MC_MoveAbsolute V2.0 MC_MoveRelative V2.0 MC_MoveVelocity V2.0 MC_MoveJog V2.0 MC_CommandTable V2.0 MC_ChangeDynamic V2.0
V1.0		V1.0 V2.0 V2.1 V2.2 V3.0	轴 V1.0	MC_Power V1.0 MC_Reset V1.0 MC_Home V1.0 MC_Halt V1.0 MC_MoveAbsolute V1.0 MC_MoveRelative V1.0 MC_MoveVelocity V1.0 MC_MoveJog V1.0

## 参见

[更改工艺版本 \(页 58\)](#)

[变量兼容表 V1...3 <-> V4...5 \(页 58\)](#)

[限位开关状态 \(页 71\)](#)

## 4.2 更改工艺版本

在获得新工艺版本的所有优势前，可能需要设置/修改现有项目的工艺版本。

### 说明

#### 工艺对象变量的兼容性

在 V1...3 以及 V4 及以上版本之间切换时，如果要在用户程序、监视表等中使用工艺对象的变量，则可参见“兼容性列表 (页 58)”。

### 设置/更改工艺版本

要设置或更改工艺版本，请执行以下步骤：

1. 打开程序编辑器（例如，通过打开 OB1）。
2. 在“指令”(Instructions) 任务卡的“工艺 > 运动控制”(Technology > Motion Control) 文件夹中，选择所需的工艺版本。
3. 保存并编译项目。请注意编译期间显示的任何错误信息。处理指示的错误原因。
4. 检查工艺对象的组态。
5. 如有必要，请调整以下对象中的变量名称以与兼容性列表一致。
  - 用户程序
  - 监控表
  - 强制表
  - HMI 组态
  - 跟踪组态

### 参见

[版本概述 \(页 51\)](#)

[限位开关状态 \(页 71\)](#)

## 4.3 变量兼容表 V1...3 <-> V4...5

在 V4 工艺框架中，S7-1200 运动控制和 S7-1500 运动控制的工艺数据块已经过标准化处理。对于 V4 及更高版本，这会生成新的定位轴和命令表工艺对象变量和变量名称。

如果用户程序中使用了工艺对象的变量，并且想要将项目从 V1...3 转换为 V4 或更高版本（或者反向转换），请注意下表中的信息。

编译项目时，“将 V1-3 自动转换到  $\geq$  V4”(Automatic conversion V1...3 to  $\geq$  V4) 列中所列的变量将进行自动转换。不会转换监视和强制表中的变量名称或者 HMI 或跟踪组态。

用户程序、监控表等中已新增以下变量或做出了调整，也可能还需要进行更正：

### Config 变量（定位轴）

V1.0 至 V3.0 的变量名称	V4.0 及更高版本的变量名称	将 V1 ... 3 自动转换为 $\geq$ V4
<轴名称>.Config.DynamicDefaults.Acceleration	<轴名称>.DynamicDefaults.Acceleration	√
<轴名称>.Config.DynamicDefaults.Deceleration	<轴名称>.DynamicDefaults.Deceleration	√
<轴名称>.Config.DynamicDefaults.EmergencyDeceleration	<轴名称>.DynamicDefaults.EmergencyDeceleration	√
<轴名称>.Config.DynamicDefaults.Jerk	<轴名称>.DynamicDefaults.Jerk	√
<轴名称>.Config.DynamicDefaults.JerkActive	不可用 在组态的加加速度 $> 0.004$ 个脉冲/ $s^3$ 时，将激活加加速度。	-
<轴名称>.Config.DynamicLimits.MaxVelocity	<轴名称>.DynamicLimits.MaxVelocity	√
<轴名称>.Config.DynamicLimits.MinVelocity	<轴名称>.DynamicLimits.MinVelocity	√
<轴名称>.Config.General.LengthUnit	<轴名称>.Units.LengthUnit	√
<轴名称>.Config.Homing.AutoReversal	<轴名称>.Homing.AutoReversal	√
<轴名称>.Config.Homing.Direction	<轴名称>.Homing.ApproachDirection	√

## 使用版本

### 4.3 变量兼容表 V1...3 <-> V4...5

V1.0 至 V3.0 的变量名称	V4.0 及更高版本的变量名称	将 V1 ... 3 自动转 换为 ≥ V4
<轴名称>.Config.Homing.FastVelocity	<轴名称>.Homing.ApproachVelocity	√
<轴名称>.Config.Homing.Offset	<轴名 称>.Sensor[1].ActiveHoming.HomePositionOffs et	√
<轴名 称>.Config.Homing.SideActiveHomi ng	<轴名称>.Sensor[1].ActiveHoming.SideInput	√
<轴名 称>.Config.Homing.SidePassiveHomi ng	<轴名称>.Sensor[1].PassiveHoming.SideInput	√
<轴名称>.Config.Homing.SlowVelocity	<轴名称>.Homing.ReferencingVelocity	√
<轴名 称>.Config.Homing.SwitchedLevel	<轴名称>.Sensor[1].ActiveHoming.SwitchLevel <轴名 称>.Sensor[1].PassiveHoming.SwitchLevel	-
<轴名 称>.Config.Mechanics.InverseDirec tion	<轴名称>.Actor.InverseDirection	√
<轴名 称>.Config.Mechanics.LeadScrew	<轴名称>.Mechanics.LeadScrew	√
<轴名 称>.Config.Mechanics.PulsesPerDrive Revolution	<轴名 称>.Actor.DriveParameter.PulsesPerDriveRevo lution	√
<轴名 称>.Config.PositionLimits_HW.Active	<轴名称>.PositionLimitsHW.Active	√
<轴名 称>.Config.PositionLimits_HW.MaxSw itchedLevel	<轴名称>.PositionLimitsHW.MaxSwitchLevel	√
<轴名 称>.Config.PositionLimits_HW.MinSwi tchedLevel	<轴名称>.PositionLimitsHW.MinSwitchLevel	√
<轴名 称>.Config.PositionLimits_SW.Active	<轴名称>.PositionLimitsSW.Active	√

V1.0 至 V3.0 的变量名称	V4.0 及更高版本的变量名称	将 V1 ... 3 自动转 换为 ≥ V4
<轴名 称>.Config.PositionLimits_SW.MaxPo sition	<轴名称>.PositionLimitsSW.MaxPosition	√
<轴名 称>.Config.PositionLimits_SW.MinPos ition	<轴名称>.PositionLimitsSW.MinPosition	√
不可用	<轴名称>.Actor.DirectionMode	-
不可用	<轴名称>.Actor.Type	-
不可用	<轴名称>.Sensor[1].ActiveHoming.Mode	-
不可用	<轴名称>.Sensor[1].PassiveHoming.Mode	-

**ErrorBits 变量 (定位轴)**

V1.0 至 V3.0 的变量名称	V4.0 及更高版本的变量名称	将 V1 ... 3 自动转 换为 ≥ V4
<轴名称>.ErrorBits.HwLimitMax	<轴名称>.ErrorBits.HWLimit	-
<轴名称>.ErrorBits.HwLimitMin	(注意新的状态位和限位开关状态 (页 71) 部 分。)	
<轴名 称>.ErrorBits.SwLimitMaxExceeded	<轴名称>.ErrorBits.SWLimits	-
<轴名 称>.ErrorBits.SwLimitMaxReached	(注意新的状态位和限位开关状态 (页 71) 部 分。)	
<轴名 称>.ErrorBits.SwLimitMinExceeded		
<轴名 称>.ErrorBits.SwLimitMinReached		
不可用	<轴名称>.ErrorBits.DirectionFault	-

## 使用版本

### 4.3 变量兼容表 V1...3 <-> V4...5

#### MotionStatus 变量（定位轴）

V1.0 至 V3.0 的变量名称	V4.0 及更高版本的变量名称	将 V1 ... 3 自动转 换为 ≥ V4
<轴名称>.MotionStatus.Distance	<轴名称>.StatusPositioning.Distance	√
<轴名称>.MotionStatus.Position	<轴名称>.Position	√
<轴名称>.MotionStatus.TargetPosition	<轴名称>.StatusPositioning.TargetPosition	√
<轴名称>.MotionStatus.Velocity	<轴名称>.Velocity	√

#### StatusBits 变量（定位轴）

V1.0 至 V3.0 的变量名称	V4.0 及更高版本的变量名称	将 V1 ... 3 自动转 换为 ≥ V4
<轴名称>.StatusBits.Homing	<轴名称>.StatusBits.HomingCommand	√
<轴名称>.StatusBits.SpeedCommand	<轴名称>.StatusBits.VelocityCommand	√
不可用	<轴名称>.StatusBits.HWLimitMaxActive	-
不可用	<轴名称>.StatusBits.HWLimitMinActive	-
不可用	<轴名称>.StatusBits.SWLimitMaxActive	-
不可用	<轴名称>.StatusBits.SWLimitMinActive	-

#### 变量（命令表）

V1.0 至 V3.0 的变量名称	V4.0 及更高版本的变量名称	将 V1 ... 3 自动转 换为 ≥ V4
<命令 表>.Config.Command[n].Position	<命令表>.Command[n].Position	√
<命令 表>.Config.Command[n].Velocity	<命令表>.Command[n].Velocity	√
<命令 表>.Config.Command[n].Duration	<命令表>.Command[n].Duration	√

V1.0 至 V3.0 的变量名称	V4.0 及更高版本的变量名称	将 V1 ... 3 自动转 换为 ≥ V4
<命令 表>.Config.Command[n].NextStep	<命令表>.Command[n].NextStep	√
<命令 表>.Config.Command[n].StepCode	<命令表>.Command[n].StepCode	√

## 参见

[版本概述 \(页 51\)](#)

[更改工艺版本 \(页 58\)](#)

[定位轴工艺对象 V4...5 的变量 \(页 485\)](#)

## 4.4 变量兼容表 V4...5 <-> V6

在工艺 V6 系统中，S7-1200 运动控制和 S7-1500 运动控制的工艺数据块仍将继续进行标准化处理。在 V6 及以上版本中，工艺对象的定位轴将生成新的变量名称。

如果在用户程序中使用该工艺对象的变量，而且要将项目从 V4-5 转换为 V6 或更高版本（或相反），请遵循下表中的信息。

编译项目时，“将 V4-5 自动转换到  $\geq$  V6”(Automatic conversion V4...5 to  $\geq$  V6) 列中所列的变量将进行自动转换。监视表、强制表、HMI 和跟踪组态中的变量名称不会转换。

在用户程序、监视表，将新增以下变量或进行相应调整和修正：

### 组态变量（定位轴）

变量名 V4.0 到 V5.0	变量名 V6.0 及以上版本	将 V4-5 自动转换为 $\geq$ V6
<轴名称>.PositionLimitsSW.Active	<轴名称>.PositionLimits_SW.Active	√
<轴名称>.PositionLimitsSW.MinPosition	<轴名称>.PositionLimits_SW.MinPosition	√
<轴名称>.PositionLimitsSW.MaxPosition	<轴名称>.PositionLimits_SW.MaxPosition	√
<轴名称>.PositionLimitsHW.Active	<轴名称>.PositionLimits_HW.Active	√
<轴名称>.PositionLimitsHW.MinSwitchLevel	<轴名称>.PositionLimits_HW.MinSwitchLevel	√
<轴名称>.PositionLimitsHW.MinSwitchAddress	<轴名称>.PositionLimits_HW.MinSwitchAddress	√

## 4.5 兼容性报文 V6 <-> V7

在 V7 工艺框架中，S7-1200 运动控制和 S7-1500 运动控制的驱动器连接 UDT 已经过标准化处理。在 V7 及以上版本中，驱动器连接的数据块将生成新的变量名称。

如果在用户程序中使用了报文变量，而且要将项目从 V6 转换为 V7 及以上版本，请遵循下表中的信息。编译项目时，“将 V6 自动转换为  $\geq$  V7”(Automatic conversion V6 to  $\geq$  V7) 列中所列的变量将进行自动转换。不会转换监视和强制表中的变量名称或者 HMI 或跟踪组态。

在用户程序、监视表，将新增以下变量或进行相应调整和修正：

### 报文 V6 <-> V7

报文	V6 中的变量名称	V7 中的变量名称	将 V6 自动转换为 $\geq$ V7
1	PD_TEL1_IN	Input	√
	ZSW1.SwitchingOnNotInhibited	ZSW1.SwitchingOnInhibited	√
	PD_TEL1_OUT	Output	√
2	PD_TEL2_IN	Input	√
	ZSW1.SwitchingOnNotInhibited	ZSW1.SwitchingOnInhibited	√
	PD_TEL2_OUT	Output	√
3	PD_TEL3_IN	Input	√

## 使用版本

### 4.5 兼容性报文 V6 <-> V7

报文	V6 中的变量名称	V7 中的变量名称		将 V6 自动转换为 ≥ V7
4	ZSW1.SwitchingOnNotInhibited	ZSW1.SwitchingOnInhibited		√
	Gx_ZSW	G1_ZSW		√
	Gx_ZSW.Reserved_Bit11	G1_ZSW.EncoderFaultAcknowledgeActive		√
	Gx_XIST1	G1_XIST1		√
	Gx_XIST2	G1_XIST2		√
	PD_TEL3_OUT	Output		√
	Gx_STW	G1_STW		√
	Gx_STW.RequestParkingSensor	G1_STW.RequestParkingEncoder		√
4	PD_TEL4_IN	Input		√
	ZSW1.SwitchingOnNotInhibited	ZSW1.SwitchingOnInhibited		√
	G1_ZSW.Reserved_Bit11	G1_ZSW.EncoderFaultAcknowledgeActive		√
	G2_ZSW.Reserved_Bit11	G2_ZSW.EncoderFaultAcknowledgeActive		√
	PD_TEL4_OUT	Output		√
	G1_STW.RequestParkingSensor	G1_STW.RequestParkingEncoder		√
	G2_STW.RequestParkingSensor	G2_STW.RequestParkingEncoder		√

报文	V6 中的变量名称	V7 中的变量名称	将 V6 自动转换为 ≥ V7
81	PD_TEL81_IN	Input	√
	ZSW2	ZSW2_ENC	√
	ZSW2.TravelToFixedEnd StopActive	ZSW2_ENC.Reserved_Bit08	√
	ZSW2.Reserved_Bit09	ZSW2_ENC.ControlRequested	√
	ZSW2.PulsesEnabled	ZSW2_ENC.Reserved_Bit10	√
	ZSW2.MotorDataSetChangeoverActive	ZSW2_ENC.Reserved_Bit11	√
	ZSW2.SlaveSignOfLifeBit0	ZSW2_ENC.SlaveLifeSignBit0	√
	ZSW2.SlaveSignOfLifeBit1	ZSW2_ENC.SlaveLifeSignBit1	√
	ZSW2.SlaveSignOfLifeBit2	ZSW2_ENC.SlaveLifeSignBit2	√
	ZSW2.SlaveSignOfLifeBit3	ZSW2_ENC.SlaveLifeSignBit3	√
	ZSW2.DriveDataSetEffectiveBit0	ZSW2_ENC.Reserved_Bit00	√
	ZSW2.DriveDataSetEffectiveBit1	ZSW2_ENC.Reserved_Bit01	√
	ZSW2.DriveDataSetEffectiveBit2	ZSW2_ENC.Reserved_Bit02	√
	ZSW2.DriveDataSetEffectiveBit3	ZSW2_ENC.FaultPresent	√
	ZSW2.DriveDataSetEffectiveBit4	ZSW2_ENC.Reserved_Bit04	√
	ZSW2.AlarmClassBit0	ZSW2_ENC.Reserved_Bit05	√
	ZSW2.AlarmClassBit1	ZSW2_ENC.Reserved_Bit06	√

## 使用版本

### 4.5 兼容性报文 V6 <-> V7

报文	V6 中的变量名称	V7 中的变量名称	将 V6 自动转换为 ≥ V7
	ZSW2.ParkingAxisActive	ZSW2_ENC.AlarmPresent	√
	Gx_ZSW	G1_ZSW	√
	Gx_ZSW.Reserved_Bit1	G1_ZSW.EncoderFaultAcknowledgeActive	√
	Gx_XIST1	G1_XIST1	√
	Gx_XIST2	G1_XIST2	√
	PD_TEL81_OUT	Output	√
	STW2	STW2_ENC	√
	STW2.TravelToFixedEndstop	STW2_ENC.Reserved_Bit08	√
	STW2.Reserved_Bit10	STW2_ENC.ControlByPic	√
	STW2.MotorSwitchoverFinished	STW2_ENC.Reserved_Bit11	√
	STW2.DriveDataSetSelectionBit0	STW2_ENC.Reserved_Bit00	√
	STW2.DriveDataSetSelectionBit1	STW2_ENC.Reserved_Bit01	√
	STW2.DriveDataSetSelectionBit2	STW2_ENC.Reserved_Bit02	√
	STW2.DriveDataSetSelectionBit3	STW2_ENC.Reserved_Bit03	√
	STW2.DriveDataSetSelectionBit4	STW2_ENC.Reserved_Bit04	√
	STW2.ParkingAxisSelection	STW2_ENC.FaultAcknowledge	√
	Gx_STW	G1_STW	√
	Gx_STW.RequestParkingSensor	G1_STW.RequestParkingEncoder	√

报文	V6 中的变量名称	V7 中的变量名称	将 V6 自动转换为 ≥ V7
83	PD_TEL83_IN	Input	√
	Gx_ZSW	G1_ZSW	√
	Gx_ZSW.Reserved_Bit1 1	G1_ZSW.EncoderFaultAcknowledgeActive	√
	Gx_XIST1	G1_XIST1	√
	Gx_XIST2	G1_XIST2	√
	ZSW2	ZSW2_ENC	√
	ZSW2.TravelToFixedEnd StopActive	ZSW2_ENC.Reserved_Bit08	√
	ZSW2.Reserved_Bit09	ZSW2_ENC.ControlRequested	√
	ZSW2.PulsesEnabled	ZSW2_ENC.Reserved_Bit10	√
	ZSW2.MotorDataSetChangeoverActive	ZSW2_ENC.Reserved_Bit11	√
	ZSW2.SlaveSignOfLifeBit0	ZSW2_ENC.SlaveLifeSignalBit0	√
	ZSW2.SlaveSignOfLifeBit1	ZSW2_ENC.SlaveLifeSignalBit1	√
	ZSW2.SlaveSignOfLifeBit2	ZSW2_ENC.SlaveLifeSignalBit2	√
	ZSW2.SlaveSignOfLifeBit3	ZSW2_ENC.SlaveLifeSignalBit3	√
	ZSW2.DriveDataSetEffectiveBit0	ZSW2_ENC.Reserved_Bit00	√
	ZSW2.DriveDataSetEffectiveBit1	ZSW2_ENC.Reserved_Bit01	√
	ZSW2.DriveDataSetEffectiveBit2	ZSW2_ENC.Reserved_Bit02	√
	ZSW2.DriveDataSetEffectiveBit3	ZSW2_ENC.FaultPresent	√

## 使用版本

### 4.5 兼容性报文 V6 <-> V7

报文	V6 中的变量名称	V7 中的变量名称	将 V6 自动转换为 ≥ V7
	ZSW2.DriveDataSetEffectiveBit4	ZSW2_ENC.Reserved_Bit04	√
	ZSW2.AlarmClassBit0	ZSW2_ENC.Reserved_Bit05	√
	ZSW2.AlarmClassBit1	ZSW2_ENC.Reserved_Bit06	√
	ZSW2.ParkingAxisActive	ZSW2_ENC.AlarmPresent	√
PD_TEL83_OUT		Output	√
	STW2	STW2_ENC	√
	STW2.TravelToFixedEndstop	STW2_ENC.Reserved_Bit08	√
	STW2.Reserved_Bit10	STW2_ENC.ControlByPic	√
	STW2.MotorSwitchoverFinished	STW2_ENC.Reserved_Bit11	√
	STW2.DriveDataSetSelectionBit0	STW2_ENC.Reserved_Bit00	√
	STW2.DriveDataSetSelectionBit1	STW2_ENC.Reserved_Bit01	√
	STW2.DriveDataSetSelectionBit2	STW2_ENC.Reserved_Bit02	√
	STW2.DriveDataSetSelectionBit3	STW2_ENC.Reserved_Bit03	√
	STW2.DriveDataSetSelectionBit4	STW2_ENC.Reserved_Bit04	√
	STW2.ParkingAxisSelection	STW2_ENC.FaultAcknowledge	√
	Gx_STW	G1_STW	√
	Gx_STW.RequestParkingSensor	G1_STW.RequestParkingEncoder	√

## 4.6 限位开关状态

在版本 V4 中，已对到达的限位开关显示的状态和错误位进行了调整。

为了复制版本 V1...3 的错误位行为，可使用以下逻辑运算符：

V1...3	V4 或更高版本
<轴名称>.ErrorBits.HwLimitMin	<轴名称>.ErrorBits.HWLimits AND <轴名称>.StatusBits.HWLimitsMinActive
<轴名称>.ErrorBits.HwLimitMax	<轴名称>.ErrorBits.HWLimits AND <轴名称>.StatusBits.HWLimitsMaxActive
<轴名称>.ErrorBits.SwLimitMinReached	<轴名称>.ErrorBits.SWLimits AND (<轴名称>.Position = <轴名称>.PositioningLimits_SW.MinPosition)
<轴名称>.ErrorBits.SwLimitMinExceeded	<轴名称>.ErrorBits.SWLimits AND (<轴名称>.Position < <轴名称>.PositioningLimits_SW.MinPosition)
<轴名称>.ErrorBits.SwLimitMaxReached	<轴名称>.ErrorBits.SWLimits AND (<轴名称>.Position = <轴名称>.PositioningLimits_SW.MaxPosition)
<轴名称>.ErrorBits.SwLimitMaxExceeded	<轴名称>.ErrorBits.SWLimits AND (<轴名称>.Position > <轴名称>.PositioningLimits_SW.MaxPosition)

### 参见

版本概述 (页 51)

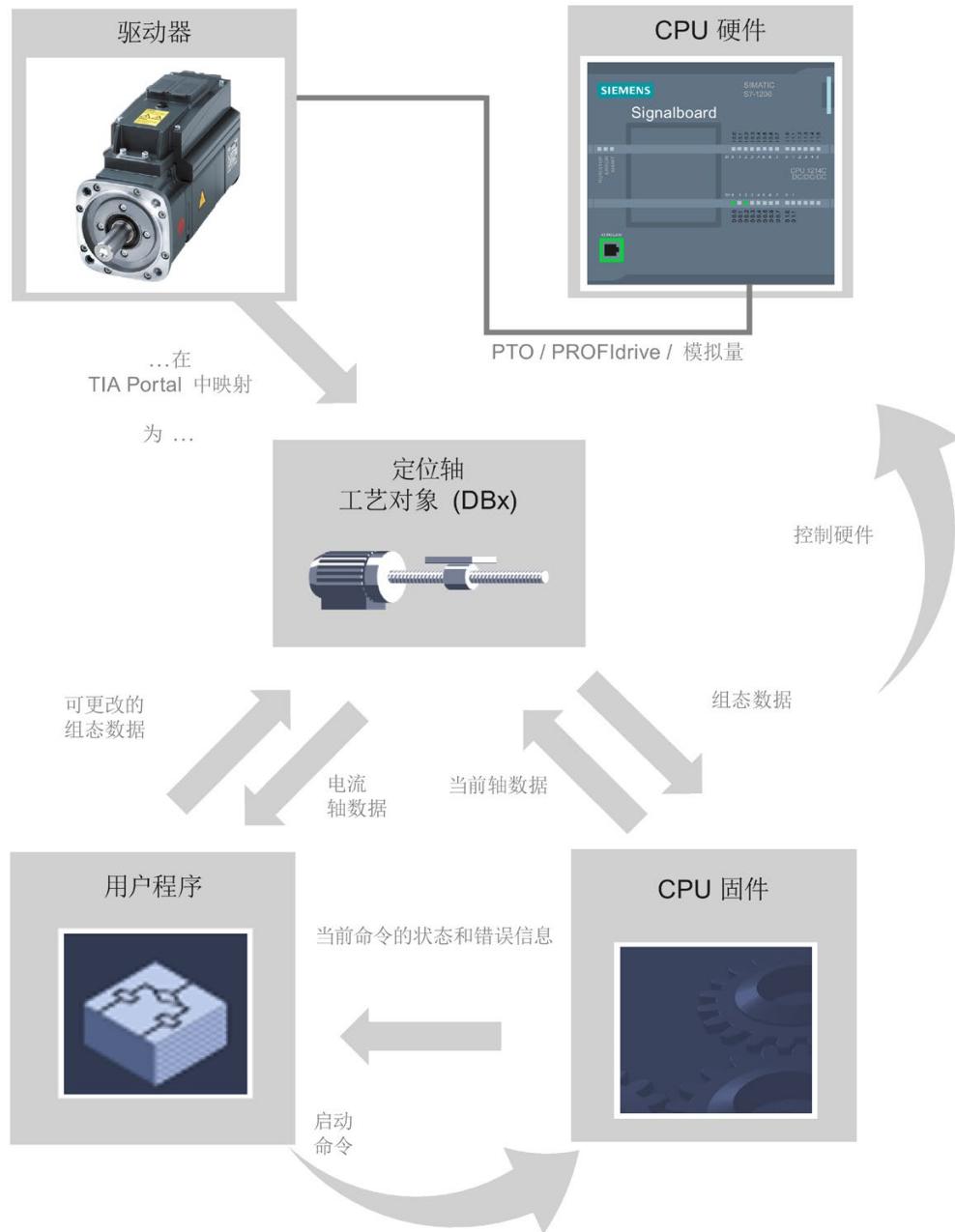
更改工艺版本 (页 58)

变量兼容表 V1...3 <-> V4...5 (页 58)

# 定位轴工艺对象

## 5.1 集成定位轴工艺对象

下图显示了使用定位轴工艺对象时所采用的硬件和软件组件间的关系：



## CPU 硬件

通过 CPU 硬件对物理驱动器进行监控。

## 驱动装置

驱动器是由动力装置和电机组成的单元。可以使用带有脉冲、PROFIdrive 或模拟接口的步进电机和伺服电机。

## 定位轴工艺对象

包含机械的物理驱动器在 TIA Portal 中映射为定位轴工艺对象。为此，使用以下参数组态定位轴工艺对象：

- 要使用的 PTO (Pulse Train Output)/PROFIdrive 驱动器/模拟量输出的选择选项和驱动器接口的组态
- 机械参数和驱动器（机器或系统）的传动比参数
- 位置限制和定位监控的参数
- 动态和回原点的参数
- 控制回路的参数

定位轴工艺对象的组态保存在该工艺对象（数据块）中。该数据块也将作为用户程序和 CPU 固件间的接口。用户程序运行期间，当前的轴数据保存在该工艺对象的数据块中。

## 5.1 集成定位轴工艺对象

### 用户程序

可以使用用户程序启动 CPU 固件中的运动控制指令作业。包括以下用于控制轴的作业：

- 启用和禁用轴
- 绝对定位轴
- 相对定位轴
- 以设定的速度移动轴
- 按运动顺序运行轴命令（自 V2 工艺版本起，仅限 PTO）
- 在点动模式下移动轴
- 停止轴
- 参考轴；设置参考点
- 更改轴的动态设置
- 连续读取轴的运动数据
- 读取和写入轴变量
- 确认错误

可以通过运动控制指令的输入参数和轴组态，确定命令参数。该指令的输出参数将提供有关状态和所有命令错误的最新信息。

启动轴命令之前，必须使用运动控制指令“MC\_Power”启用轴。

通过工艺对象的变量，可读取组态数据和当前的轴数据。通过用户程序，可更改工艺对象的单个可更改变量（如，当前的加速度）。

也可以通过运动控制指令“MC\_ChangeDynamic”更改轴的动态设置，通过“MC\_WriteParam”写入其它组态数据。通过运动控制指令“MC\_ReadParam”读取轴的当前运动状态。

## CPU 固件

用户程序中启动的运动控制作业在 CPU 固件中进行处理。使用轴控制面板时，可以通过操作轴控制面板来触发运动控制作业。CPU 固件执行以下作业，具体取决于组态：

- 计算运动作业的精确运动轨迹和紧急停止情况
- 通过 PROFIdrive/模拟驱动器接口的驱动器连接位置控制
- 通过 PTO 控制驱动器接口的脉冲和方向信号
- 控制驱动器启用
- 监视驱动器，以及硬限位开关和软限位开关
- 将最新状态和错误信息反馈给用户程序中的运动控制指令
- 将当前的轴数据写入到该工艺对象的数据块中

## 参见

[定位轴工艺对象 V6...7 的变量 \(页 335\)](#)

[运动控制相关的 CPU 输出 \(页 16\)](#)

[信号类型与行进方向之间的关系 \(页 21\)](#)

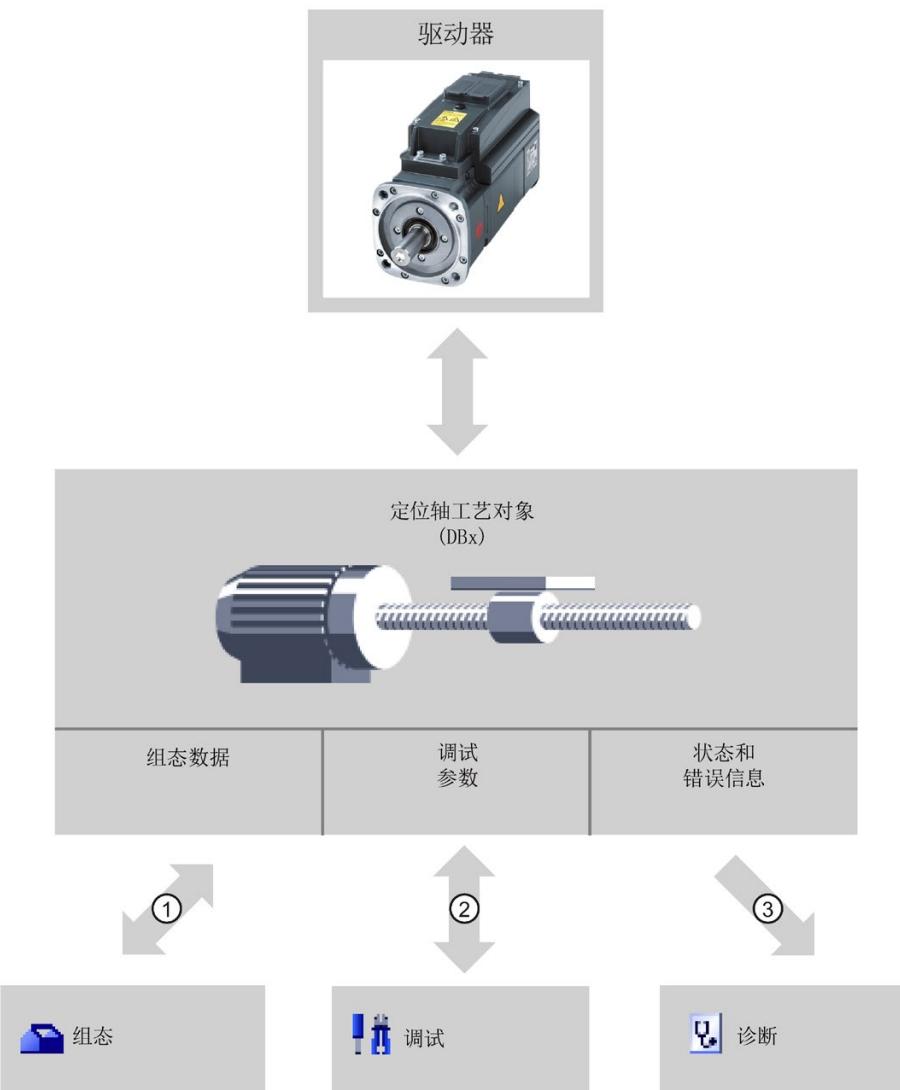
[定位轴工艺对象的工具 \(页 76\)](#)

[硬件和软件限位开关 \(页 45\)](#)

[回原点 \(页 47\)](#)

## 5.2 定位轴工艺对象的工具

TIA Portal 中将为定位轴工艺对象提供“组态”(Configuration)、“调试”(Commissioning) 和“诊断”(Diagnostics) 工具。下图显示了这三种工具与工艺对象和驱动器的相互关系：



①	读取和写入工艺对象的组态数据
②	通过工艺对象的驱动器控制。读取轴控制面板上显示的轴状态。优化位置控制
③	读取工艺对象的当前状态和错误信息 显示 PROFIdrive 驱动器的更多消息帧信息。

## 组态

使用“组态”(Configuration) 工具可以组态定位轴工艺对象的以下属性：

- 要使用的 PTO (Pulse Train Output)/PROFIdrive 驱动器/模拟量输出的选择选项和驱动器接口的组态
- 机械装置的属性和驱动器（机器/设备）的传动比参数
- 位置限制和定位监控的属性
- 动态和回原点的属性
- 控制回路的参数

在工艺对象的数据块中保存组态数据。

## 调试

使用“调试”工具即可测试轴的功能，无需创建用户程序。启动该工具时，将显示轴控制面板。轴控制面板提供了下列命令：

- 启用和禁用轴
- 在点动模式下移动轴
- 以绝对和相对方式定位轴
- 使轴回原点
- 确认错误信息

可以为运动命令相应地调整动态值。轴控制面板还显示当前的轴状态。

利用基于 PROFIdrive/模拟量输出的驱动器连接，通过自整定，可确定最佳的控制环增益。

## 诊断

使用“诊断”工具，可以跟踪轴和驱动器的当前状态和错误信息。

## 5.2 定位轴工艺对象的工具

### 参见

- 运动控制相关的 CPU 输出 (页 16)
- 信号类型与行进方向之间的关系 (页 21)
- 集成定位轴工艺对象 (页 72)
- 硬件和软件限位开关 (页 45)
- 回原点 (页 47)
- 组态定位轴工艺对象 (页 80)
- 轴控制面板 (页 188)
- 轴 - 诊断 (页 221)

## 5.3 添加一个定位轴工艺对象

### 要求

已创建具有 CPU S7-1200 的项目。

### 步骤

要在项目树中添加定位轴工艺对象，请按以下步骤操作：

1. 在项目树中打开“CPU > 工艺对象”(CPU > Technology objects) 文件夹。
2. 双击“添加新对象”(Add new object) 命令。  
将打开“添加新对象”(Add new object) 对话框。
3. 选择“运动控制”(Motion Control) 工艺。
4. 打开“运动控制”(Motion Control) 文件夹。
5. 在“版本”(Version) 列中选择所需的工艺版本。
6. 选择“TO\_PositioningAxis”对象。
7. 在“名称”(Name) 输入框中输入轴名称。
8. 要更改自动分配的数据块编号，请选择“手动”(Manual) 选项。
9. 要显示有关工艺对象的其它信息，请单击“其它信息”(Additional information)。
10. 单击“确定”(OK) 确认输入。

### 结果

创建了新工艺对象，并保存在项目树中的“工艺对象”(Technology objects) 文件夹中。

组织块 MC-Servo [OB91] 和 MC-Interpolator [OB92] 在“程序块”(Program blocks) 文件夹中自动创建。工艺对象在这些组织块中进行处理。MC-Servo [OB91] 中计算的位置控制器。MC-Interpolator [OB92] 对运动控制指令、设定值生成和监视功能进行评估。

### 参见

运动控制使用指南 (页 50)

## 5.4 组态定位轴工艺对象

### 5.4.1 使用组态对话框

在组态窗口中，组态工艺对象的属性。要打开工艺对象的组态窗口，请按以下步骤操作：

1. 在项目树中打开所需工艺对象组。
2. 双击“组态”(Configuration) 对象。

组态分为以下几类：

- **基本参数**

基本参数包括必须为工作轴组态的所有参数。

- **扩展参数**

高级参数包括适合特定驱动器或设备的参数。

### 组态窗口图标

组态的区域导航中的图标显示有关组态情况的详细信息：

	<b>组态包含默认值且已完成。</b> 组态仅包含默认值。使用这些默认值即可使用工艺对象，而无需进行更改。
	<b>组态中包含用户自定义或系统自动调整的值且已完成。</b> 组态的所有输入域均包含有效值，且至少一个预设值发生更改。
	<b>组态未完成或不正确。</b> 至少一个输入域或下拉列表包含无效值。相应域或下拉列表以红色背景显示。单击弹出错误消息可显示错误原因。
	<b>组态有效但包含警告。</b> 例如，仅组态一个硬限位开关。根据具体的应用，硬限位开关的组态缺失可能会引发危险。相应域或下拉列表以黄色背景显示。

### 参见

[运动控制使用指南 \(页 50\)](#)

[基本参数 \(页 82\)](#)

[扩展参数 \(页 97\)](#)

## 5.4.2 监视值

如果已在线连接 CPU，则“全部监视”(Monitor all) 图标  将显示在工艺对象的组态对话框中。

“监视全部”(Monitor all) 功能中包含以下选项：

- 将项目中组态的起始值与 CPU 中的起始值和实际值进行比较
- 直接编辑实际值和项目的起始值
- 立即检测并显示输入错误和建议的更正措施
- 备份项目中的实际值，并手动传送到项目的起始值中

### 图标和操作员控件

如果存在到 CPU 的在线连接，则参数中将显示实际值。

除参数的实际值外，还会显示下列符号：

图标	说明
	CPU 中的起始值与已组态的项目中起始值相匹配
	CPU 中的起始值与已组态的项目中的起始值不匹配
	由于所选 CPU 模块不支持这种比较方式，因此 CPU 中的起始值无法与所组态项目中的起始值进行比较。
	由于该值与组态无关，因此无法比较。
	使用按钮显示 CPU 的起始值和各个参数的项目起始值。

可以直接更改项目中的实际值和起始值，然后下载到 CPU。可直接修改的参数的实际值更改将直接传送到 CPU。

### 5.4.3 基本参数

#### 5.4.3.1 组态 - 常规

在“常规”(General) 组态窗口中，组态定位轴工艺对象的基本属性。

##### 轴名称

在该域中定义轴名称或定位轴工艺对象的名称。该工艺对象以该名称列出在项目树中。

##### 驱动装置

选择驱动器连接的类型：

- **PTO (Pulse Train Output)**

驱动器通过脉冲发生器输出、可选使能输出和可选准备就绪输入进行连接。

- **模拟驱动器接口**

驱动装置可通过模拟量输出、编码器、可选的使能输出和可选的就绪输入进行连接。

轴的所有运动均在位置控制下进行。

- **PROFIdrive**

驱动装置通过 PROFINET/PROFIBUS 进行连接。控制器和驱动器之间通过 PROFIdrive 报文进行通信。

轴的所有运动均在位置控制下进行。

如果选择“模拟驱动器连接”(Analog drive connection) 或“PROFIdrive”，其它元素将添加到组态导航中：

- 编码器
- 模数
- 位置监控（定位监控、跟随误差和静止信号）
- 控制回路

在其它组态窗口中，可以组态待连接的编码器，以及位置控制和定位监控的结果选项。

## 位置测量的单位

在下拉列表中，选择轴测量系统的测量单位。选择的测量单位将用于定位轴工艺对象的进一步组态中以及当前轴数据的显示中。

运动控制指令的输入参数（Position、Distance、Velocity 等）值也会使用该测量单位。

### 说明

在进行轴组态时，可选择该位置处的驱动装置接口和测量单位。

后续更改时，参数需复位或重新初始化，要求用户再次检查组态对话框的参数。

在用户程序中，可能需要根据新的测量单位对运动控制指令的输入参数值进行相应调整。

## 仿真

在下拉列表中，选择是否仿真驱动装置和编码器。模拟量驱动接口或 PROFIdrive 驱动装置均可仿真。在仿真模式下，不需要对驱动装置和编码器进行硬件配置（系统将忽略驱动装置和编码器组态中潜在的错误）。

应用：进行调试或使用组态的硬件时可对驱动器进行仿真。

用户程序在运行过程中可更改“仿真”操作模式，之后 MC\_Reset 的参数“Restart”为 TRUE。

在仿真模式下，设定值不会输出到驱动器，也不从驱动器/编码器读取实际值。硬限位开关和回原点开关不受影响。

下表列出了运动控制指令及仿真模式下的相应行为。

运动控制指令	仿真模式下的行为
MC_Power	该轴将立即启用，而不等待驱动装置的反馈。
MC_Home	回原点作业将立即执行，而不仿真轴运动。

PTO 驱动装置无需控制环。未连接 PTO 驱动装置时，仿真 PTO 驱动器无需使用单独的仿真功能。

## 参见

运动控制相关的 CPU 输出 (页 16)

信号类型与行进方向之间的关系 (页 21)

组态 - 常规 (“轴”工艺对象 V1...3) (页 393)

### 5.4.3.2 组态 - 驱动器

#### 组态 - 驱动器 - PTO (脉冲串输出)

在“驱动器”(Drive) 组态窗口中，组态脉冲发生器和驱动器的使能与反馈。

##### 硬件接口

脉冲通过固定分配的数字量输出输出到驱动器的动力装置。

对于具有继电器输出的 CPU，由于继电器不支持所需的开关频率，因此无法通过这些输出来输出脉冲信号。如果要在这些 CPU 中使用 PTO (Pulse Train Output)，必须使用具有数字量输出的信号板。 用于打开 CPU 的设备组态。

##### 说明

PTO 需要高速计数器 (HSC) 的功能。为此将使用内部 HSC，不能对此 HSC 的计数进行评估。

##### 脉冲发生器

在该下拉列表中，选择 PTO (Pulse Train Output)，通过脉冲接口来控制步进电机或伺服电机。如果没有在设备组态中的其它地方使用脉冲发生器和高速计数器，则系统会自动组态硬件接口。这种情况下，下拉列表中所选的 PTO 以白色背景显示。

如果选择 PTO (PulseTrain Output)，则按下“设备组态”(Device configuration) 按钮时将转至 CPU 设备组态中的脉冲选项参数分配。如果由于 PTO 正在另一侧使用或用户更改了参数而导致冲突时，这功能非要有用。

## 信号类型

从下拉列表中选择信号类型。可以使用以下信号类型：

- **PTO (脉冲 A 和方向 B)**

使用一个脉冲输出和一个方向输出控制步进电机。

- **PTO (时钟增加 A 和时钟减少 B)**

分别使用一个正向和负向运动的脉冲输出控制步进电机。

- **PTO (A/B 相移)**

A 相和 B 相的两个脉冲输出在同一频率下运行。

在驱动器步进结束时会评估这两个脉冲输出的周期。

A 相和 B 相之间的相位偏移量决定了运动方向。

- **PTO (A/B 相位偏移量 - 四重)**

A 相和 B 相的两个脉冲输出在同一频率下运行。

在驱动器步进结束时会评估 A 相和 B 相的所有上升沿和下降沿。

A 相和 B 相之间的相位偏移量决定了运动方向。

下表列出了要根据信号类型组态的参数：

信号类型/参数	说明
<b>PTO (脉冲 A 和方向 B)</b>	
脉冲输出	在该域中，选择正方向上运动的脉冲输出。 可以通过符号地址或将其分配给绝对地址来选择输出。
激活方向输出	使用此选项，可以启用或禁用方向输出。在禁用方向输出时，将会限制运动方向。
方向输出	在此域中选择用作方向输出的输出。 可以通过符号地址或将其分配给绝对地址来选择输出。

## 5.4 组态定位轴工艺对象

信号类型/参数	说明
<b>PTO (时钟增加 A 和时钟减少 B)</b>	
脉冲正向输出	在该域中，选择正方向上运动的脉冲输出。 可以通过符号地址或将其分配给绝对地址来选择输出。
脉冲反向输出	在该域中，选择负方向上运动的脉冲输出。 可以通过符号地址或将其分配给绝对地址来选择输出。
<b>PTO (A/B 相位偏移量) /PTO (A/B 相位偏移量 - 四重)</b>	
信号 A	在此域中为 A 相信号选择脉冲输出。 可以通过符号地址或将其分配给绝对地址来选择输出。
信号 B	在此域中为 B 相信号选择脉冲输出。 可以通过符号地址或将其分配给绝对地址来选择输出。

### 驱动器使能信号与反馈

在此区域中组态驱动器使能信号的输出以及驱动器的“驱动器准备就绪”(Drive ready) 反馈信号的输入。

- 使能输出

在此域中为驱动器使能信号选择使能输出。

- 就绪输入

在此域中为驱动器的“驱动器准备就绪”(Drive ready) 反馈选择准备就绪输入

驱动器使能信号由运动控制指令“MC\_Power”控制，可以启用对驱动器的供电。如果驱动器在接收到驱动器使能信号之后准备好开始执行运动，则驱动器会向 CPU 发送“驱动器准备就绪”(Drive ready) 信号。

如果驱动器不包含此类型的任何接口，则无需组态这些参数。这种情况下，为准备就绪输入选择值 TRUE。

### 组态 - 驱动器 - 模拟驱动器接口

在“驱动器”(Drive) 组态窗口中，组态模拟量输出以及驱动器的使能与反馈。

## 硬件接口

通过永久性分配的模拟量输出，可将速度设定值输出到驱动装置的电源装置内。

在以下区域中为驱动器的控制组态输入和输出：

- 模拟量输出

在此域中，选择用于控制驱动器的模拟量输出的 PLC 变量。

在打开自动填充功能时，所有输出地址都以 16 位显示（WORD、INT、UINT）。通过数据块建立数据连接时，可选择 WORD 数据类型的数据块变量。

也可以输入地址，例如 QW20。如果地址有效，将生成此地址的名称

“Axis\_1\_AnalogOutput”，并插入变量表中。为确保地址有效，需将其分配给相应的数据类型和 HW 模块。 用于打开模拟量输出的设备组态。

- 选择使能输出

在此域中，为驱动器使能信号选择一个可用的输出作为使能输出。

- 选择准备就绪输入

在此域中为驱动器的“驱动器准备就绪”(Drive ready) 反馈选择准备就绪输入

驱动器使能信号由运动控制指令“MC\_Power”控制，可以启用对驱动器的供电。如果驱动器在接收到驱动器使能信号之后准备好开始执行运动，则驱动器会向 CPU 发送“驱动器准备就绪”(Drive ready) 信号。如果驱动器不包含此类型的任何接口，则无需组态这些参数。这种情况下，为准备就绪输入选择值 TRUE。

## 5.4 组态定位轴工艺对象

### 与驱动器之间的数据交换

在该区域中，可以组态速度设定值的缩放：

- 参考速度

驱动器的参考速度，是指模拟量输出为 100% 输出时驱动器的旋转速度。驱动器中必须组态参考速度，并传输到工艺对象的组态中。

以 100% 输出的模拟值具体取决于模拟量输出的类型。例如，对于  $\pm 10 \text{ V}$  的模拟量输出，值  $10 \text{ V}$  是以 100% 的方式输出的。

- 最大速度

在该域中，将指定驱动器的最大速度。

最大速度受驱动装置和模拟量输出值范围的限制。为了便捷起见，可将参考速度设置为最大速度。

模拟量输出可过载约 17%。如果驱动装置允许过载，则模拟量输出的操作范围为 -117% 到 117%。

- 反转驱动装置方向

如果要反向驱动器的旋转方向，则可选择该复选框。

### 参见

带模拟量驱动接口的数据连接驱动装置 (页 37)

### 组态 - 驱动装置 - PROFIdrive

在“驱动装置”(Drive) 组态窗口中，组态 PROFIdrive 驱动装置的数据连接和参数。 用于打开驱动装置的设备组态。

## PROFIdrive 驱动装置 (V6 及以上版本)

- **数据连接**

在下拉列表中，选择数据接口直接连接驱动设备，或通过用户程序中的可编辑数据块连接。

- **驱动装置** (数据接口：“驱动装置”)

在“驱动装置”(Drive) 域中，选择一个已组态的 PROFIdrive 驱动装置。

- **数据块** (数据接口：“数据块”)

选择一个之前创建的数据块，并包含数据类型为“PD\_TELx”的变量结构（“x”为所用的报文编号）。

## 与驱动装置间进行数据交换

在该区域中，可组态驱动装置和控制器间的数据交换：

- **驱动装置报文** (数据接口：未选择“数据块”(Data block))

在下拉列表中，检查并选择该驱动装置的报文。该规范必须与驱动装置的设备组态相匹配。

- **输入/输出地址**

这些域显示报文的符号及绝对输入和输出地址。

- **反转驱动装置方向**

如果要反向驱动装置的旋转方向，则可选择该复选框。

## 5.4 组态定位轴工艺对象

- **组态时自动应用驱动值（离线）**

要将驱动装置“参考速度”和“最大速度”的离线值传输到项目中工艺对象的组态中，请选中此复选框。

- **运行时自动应用驱动值（在线）**

如果要将驱动装置参数“参考速度”(Reference speed) 和“最大速度”(Maximum speed) 作为驱动装置组态值传输到 CPU 中，可选择该复选框。对工艺对象进行（重新）初始化并（重新）启动驱动装置和 CPU 后，将通过总线传送驱动装置参数。

也可以选择手动同步以下参数：

- **参考速度**

组态参考速度，与驱动装置组态中的值相匹配。

举例来说，在总线上，待传送的值 16#4000 (16 位，针对报文 1) 与参考速度的 100% 相对应。

- **最大速度**

在该域中，将组态驱动装置的最大速度。

从驱动装置的组态中获取最大速度。通过总线可传送的参考速度范围为 -200% 到 +200%。因此，最大速度最多为参考速度的两倍。

---

### 说明

只有 SINAMICS 驱动装置 (V4.x 及以上版本) 支持自动传送驱动装置参数。为此，必须在组态窗口中为数据连接选择“驱动装置”(Drive)。

---

### 参见

[PROFIdrive 驱动装置/PROFIdrive 编码器的数据连接 \(页 31\)](#)

[自动传送设备中的驱动装置与编码器参数 \(页 26\)](#)

### 5.4.3.3 组态 - 编码器

#### 编码器连接

在“编码器”(Encoder) 组态窗口中，根据所选的编码器连接组态各个参数。可实现以下编码器连接：

- 高速计数器 (HSC) 上的编码器 (页 94)
- PROFINET/PROFIBUS 上的 PROFIdrive 编码器（驱动装置上的编码器、工艺模块上的编码器、PROFIdrive 编码器）(页 **Error! Bookmark not defined.**)

#### 组态 - 编码器 - PROFINET/PROFIBUS 上的编码器

#### 编码器选择

在“PROFIdrive 编码器”(PROFIdrive encoder) 框中，选择 PROFINET 上的 PROFIdrive 编码器。

- **数据连接**

在下拉列表中，选择数据接口直连连接编码器，或通过用户程序中的可编辑数据块进行连接。

- **PROFIdrive 编码器/数据块**

在该组态域中，选择一个事先组态的 PROFIdrive 编码器。

 用于打开编码器的设备组态。可选择以下编码器：

- **连接至驱动装置（不带模拟量驱动装置接口）**

编码器连接至驱动装置。编码器信号通过驱动装置进行评估，并作为驱动装置报文的一部分（报文 3 或 4）传送到控制器中（其它驱动装置的编码器报文无法使用）。

编码器通过 PROFIdrive 驱动装置的组态进行设置。

- **工艺模块 (TM) 上的编码器**

选择之前组态的工艺模块和要使用的通道。仅显示设置为“运动控制的位置输入”(Position input for Motion Control) 模式的工艺模块以供选择。

如果没有工艺模块可供选择，请切换到设备组态并添加工艺模块。

可以通过工艺模块文档和目录数据确定适合运动控制位置检测的工艺模块。

## 5.4 组态定位轴工艺对象

### – PROFINET/PROFIBUS 上的 PROFIdrive 编码器 (PROFIdrive)

在“PROFIdrive 编码器”(PROFIdrive encoder) 域中，选择之前在 PROFINET/PROFIBUS 上组态的某个编码器。如果没有可选编码器，则切换至网络视图中的设备组态，并添加一个编码器。

如果选择“数据块”(Data block) 进行数据连接，则在此必需选择之前创建的数据块（含数据类型“PD\_TELx”的变量结构）。其中，“x”为连接编码器时使用的报文编号。可使用所选驱动装置报文的编码器（报文 3 或 4），也可选择一个单独的编码器（报文 81 或 83）。 用于打开 DB 编辑器。

## 与编码器之间的数据交换

在该区域中，可以组态编码器和控制器之间的数据交换：

- **编码器报文** (用于数据连接：未选择“数据块”(Data block))

在下拉列表中，选择编码器的报文。其技术数据必须与设备组态相一致。

- **输入/输出地址**

这些域显示报文的符号及绝对输入和输出地址。

- **反转编码器方向**

要反转编码器的实际值，请选中此复选框。

- **组态时自动应用编码器值 (离线)**

如果要将编码器的离线值传送到项目中工艺对象的组态中，则可选择该复选框。

- **运行时自动应用编码器值 (在线)**

如果要从编码器组态传输编码器参数到 CPU，则选中该复选框。（重新）初始化工艺对象和（重新）启动编码器和 CPU 后，将从总线传送编码器参数。编码器的类型必须与轴组态以及驱动装置组态中的类型相同。

## 说明

只有 PROFIdrive 编码器 (A16 及以上版本) 支持自动传送编码器参数。为此，必须在组态窗口中为数据连接选择“编码器”(Encoder)。

在 SINAMICS 驱动装置上使用编码器，产品版本应为 V4.x 及以上版本。

如果编码器参数未自动传送，则需手动调整参数。有关需同步的参数，请参见“自动传送设备中的驱动装置与编码器参数 (页 26)”部分。

## 编码器类型

在“编码器类型”(Encoder type) 对话框中选择使用的编码器类型。可选择以下编码器类型：

- 线性增量式
- 线性绝对式
- 旋转增量式
- 旋转绝对式

根据所选编码器类型，组态各个参数。根据所选编码器类型，组态以下参数：

编码器类型/参数	说明
<b>线性增量式</b>	
两个增量之间的距离	在该域中，可以组态编码器两步之间的距离。
高精度 - 增量实际值中的位数(Gx_XIST1)	在该字段中，组态增量实际值中高精度的位数(Gx_XIST1)。
<b>线性绝对式</b>	
两个增量之间的距离	在该域中，可以组态编码器两步之间的距离。
高精度 - 增量实际值中的位数(Gx_XIST1)	在该字段中，组态增量实际值中高精度的位数(Gx_XIST1)。
高精度 - 绝对实际值中的位数 (Gx_XIST2)	在该字段中，组态高精度绝对值倍增系数的预留位数 (Gx_XIST2)。
<b>旋转增量式</b>	
单转步数	在该域中，组态编码器每一转可以解析出的步数。
高精度 - 增量实际值中的位数(Gx_XIST1)	在该字段中，组态增量实际值中高精度的位数(Gx_XIST1)。
<b>旋转绝对式</b>	
单转步数	在该域中，组态编码器每一转可以解析出的步数。
转数	在该域中，组态绝对值编码器可以检测出的转数。
高精度 - 增量实际值中的位数(Gx_XIST1)	在该字段中，组态增量实际值中高精度的位数(Gx_XIST1)。
高精度 - 绝对实际值中的位数 (Gx_XIST2)	在该字段中，组态高精度绝对值倍增系数的预留位数 (Gx_XIST2)。

## 5.4 组态定位轴工艺对象

### 参见

**PROFIdrive 驱动装置/PROFIdrive 编码器的数据连接 (页 31)**

**组态用于运动控制的工艺模块 (页 156)**

### 组态 - 编码器 - 高速计数器 (HSC) 上的编码器

#### 选择高速计数器 (HSC)

在“选择高速计数器”(Select high-speed counter) 对话框中，选择编码器将向其传输实际值的高速计数器。

检查所使用的两个高速计数器数字量输入的过滤时间。过滤时间应足够短，以确保可靠记录脉冲。

#### HSC 接口

在“操作模式”(Operating mode) 框中，选择高速计数器的操作模式。

根据操作模式组态各个输入：

操作模式/参数	说明
<b>双相</b>	
正向时钟脉冲发生器	<p>在此域中选择用于加计数的输入。</p> <p>可以通过符号地址或将其分配给绝对地址来选择输入。</p> <p>输入的频率和位置（板载，信号板）显示在地址框的旁边。</p>
反向时钟脉冲发生器	<p>在此域中选择用于减计数的输入。</p> <p>可以通过符号地址或将其分配给绝对地址来选择输入。</p> <p>输入的频率和位置（板载，信号板）显示在地址框的旁边。</p>

操作模式/参数	说明
<b>A/B 计数器 / A/B 四重计数器</b>	
时钟脉冲发生器 A	<p>在此域中选择用于脉冲 A 信号的输入。</p> <p>可以通过符号地址或将其分配给绝对地址来选择输入。</p> <p>输入的频率和位置（板载，信号板）显示在地址框的旁边。</p>
时钟脉冲发生器 B	<p>在此域中选择用于脉冲 B 信号的输入。</p> <p>可以通过符号地址或将其分配给绝对地址来选择输入。</p> <p>输入的频率和位置（板载，信号板）显示在地址框的旁边。</p>

### 反转编码器方向

要反转编码器的实际值，请选中此复选框。

### 自动传送设备中的编码器参数

在高速计数器 (HSC) 上使用编码器时，无法使用该选择。

## 5.4 组态定位轴工艺对象

### 编码器类型

在“编码器类型”(Encoder type) 框中选择编码器类型。可选择以下编码器类型：

- 线性增量式
- 旋转增量式

根据所选编码器类型，组态各个参数。根据所选编码器类型，组态以下参数：

编码器类型/参数	说明
<b>线性增量式</b>	
两个增量之间的距离	在该域中，可以组态编码器两步之间的距离。
高精度 - 增量实际值中的位数(Gx_XIST1)	在该字段中，组态增量实际值中高精度的位数(Gx_XIST1)。
<b>旋转增量式</b>	
单转步数	在该域中，组态编码器每一转可以解析出的步数。
高精度 - 增量实际值中的位数(Gx_XIST1)	在该字段中，组态增量实际值中高精度的位数(Gx_XIST1)。

## 5.4.4 扩展参数

### 5.4.4.1 机械

#### 组态 - 机械 - PTO (脉冲串输出)

在“机械”(Mechanics) 组态窗口中组态驱动器的机械属性。

##### 电机每转的脉冲数

在此框中组态电机每转所需的脉冲数。

限值（与所选测量单位无关）：

- $0 < \text{电机每转的脉冲数} \leq 2147483647$

##### 电机每转的负载运动

在此框中，组态电机每转带动单元的机械系统行进的负载距离。

限值（与所选测量单位无关）：

- $0.0 < \text{每转距离} \leq 1.0e12$

##### 允许的旋转方向（自 V4 工艺版本起）

通过组态此框可决定系统机械是同时朝两个方向运动，还是只朝正向或负向运动。

如果尚未在“PTO (脉冲 A 和方向 B)”模式下激活脉冲发生器的方向输出，则选择受限于正方向或负方向。

## 5.4 组态定位轴工艺对象

### 反转方向

可使用“反转方向”(Invert direction) 复选框根据驱动器的方向逻辑对控制系统进行调整。

方向逻辑将根据所选脉冲发生器的模式反转：

- **PTO (脉冲 A 和方向 B)**

- 方向输出上输出 0 V ⇒ 正向旋转
- 方向输出上输出 5 V/24 V ⇒ 负向旋转

指定的电压取决于所使用的硬件。各指示值并不适用于 CPU 1217 的差分输出。

- **PTO (时钟增加 A, 时钟减少 B)**

“向下脉冲输出”和“向上脉冲输出”这两个输出相互交换。

- **PTO (A/B 相移)**

“A 相”和“B 相”输出相互交换。

- **PTO (A/B 相位偏移量 - 四重)**

“A 相”和“B 相”输出相互交换。

### 组态 - 机械 - PROFIdrive/模拟驱动器接口

在“机械”(Mechanics) 组态窗口中组态驱动器的机械属性及其编码器。

### 编码器安装方式

在下拉列表中，选择如何将编码器安装在机械机构上。支持下列编码器安装类型：

- 在电机轴上
- 外部测量系统（仅适用于旋转式编码器）

## 位置参数

根据所选编码器安装类型，组态以下位置参数：

编码器安装类型/位置参数	说明
<b>在电机轴上</b>	
电机每转的负载运动	在该域中，组态电机每转的负载距离。
<b>外部测量系统</b>	
电机每转的负载运动	在该域中，组态电机每转的负载距离。
编码器每转的距离	在该域中，组态编码器每旋转一圈外部测量系统所记录的距离。

## 5.4 组态定位轴工艺对象

### 5.4.4.2 组态 - 模数 (仅限 PROFIdrive/模拟驱动器接口)

使用“模数”(modulo) 设置，可根据产品长度/产品循环，将行进范围限制为一个固定距离。模函数仅适用于轴的位置控制操作中。

启用“模数”(modulo) 时，将使用一个递归模数范围表示工艺对象的位置值。模数范围由起始值和长度定义。

例如，要将轴的位置值限制为一整圈，可将模数范围定义为起始值 = 0°、长度 = 360°。

编码器精度为 0.1°/编码器步时，位置值的模数范围是 0.0° 到 359.9°。在本示例中，如果轴移动到位置 400° 处，则到达的实际位置为 40° (400° 到 360°)。

“Modulo”激活时，运动控制命令“MC\_MoveAbsolute”将通过输入参数“Direction”指定行进方向。可使用以下参数值：

- 0: 速度的符号 (“Velocity”参数) 用于确定运动的方向。
- 1: 从正方向逼近目标位置。
- 2: 从负方向逼近目标位置。
- 3: 工艺将选择从当前位置开始，到目标位置的最短距离

#### 启用模数

选择“启用模数”(Enable modulo) 复选框，为轴使用递归参考系统（例如 0.0° 到 359.9°）。

#### 模数起始值

在该域中，定义模数运算范围的起始位置（例如 0°）。

#### 模数长度

在该域中，定义模数范围的长度（例如 360°）。

#### 参见

**MC\_MoveAbsolute:** 轴的绝对定位 (V6 及以上版本) (页 249)

### 5.4.4.3 位置限制

#### 对硬件限位开关的要求

仅使用逼近后始终保持切换的硬件限位开关。返回至允许的行进范围之后，才可撤消该开关状态。

#### 参见

组态 - 位置限制 (页 101)

触发轴限位时的轴响应 (页 104)

在用户程序中可以更改位置限位的组态 (页 106)

#### 组态 - 位置限制

在“位置限制”(Position limits) 组态窗口中组态轴的硬件和软限位开关。

#### 启用硬限位开关

使用此复选框可激活硬限位开关的下限和上限功能。硬限位开关可用于在回原点过程中反转行进方向。有关详细信息，请参见回原点的组态说明。

#### 启用软限位开关

使用此复选框可激活软限位开关的下限和上限功能。

---

#### 说明

已激活的软限位开关仅影响已回原点的轴。

---

## 5.4 组态定位轴工艺对象

### 输入开关硬限位开关下限 / 上限

在下拉列表中，选择硬限位开关下限或上限的数字量输入。

PTO 轴的输入必须具有中断功能。通过 PROFIdrive / 模拟量驱动装置进行连接时，带有驱动装置连接且具有中断功能的输入可达到最短响应时间。此外，也可将该输入指定给过程映像“TPA OB Servo”，之后即可在“TPA OB Servo”的循环时间中接收到一个响应时间。由于可能会导致响应时间最大，因此不建议指定组织块 OB1 的标准过程映像。

板载数字量 CPU 的输入和所插入信号板的数字量输入，均可选作为硬限位开关具备中断功能的输入。

#### 说明

默认情况下，数字量输入的滤波时间设置为 6.4 ms。如果将这种数字量输入用作硬限位开关的输入，则可能会发生意外减速情况。如果出现这种情况，则需降低相关数字量输入的滤波时间。

可以在数字量输入设备组态的“输入滤波器”(Input filter) 中设置滤波时间。

### 选择信号电平

在此下拉列表中，可选择逼近硬限位开关时 CPU 输入端的信号电平。

- 选择“低电平”(Low level)（常闭触点）

CPU 输入端电平为 0 V (FALSE) 时表示已逼近硬限位开关

- 选择“高电平”(High level)（常开触点）

CPU 输入端电平为 5 V/24 V (TRUE) = 已逼近硬限位开关（实际电压取决于使用的硬件）

## 软限位开关上限 / 下限

在这些输入框中，可输入软限位开关下限和上限的位置值。

限值（与所选测量单位无关）：

- $-1.0E12 \leq$  下限软限位开关  $\leq 1.0E12$
- $-1.0E12 \leq$  上限软限位开关  $\leq 1.0E12$

软限位开关的上限值必须大于或等于软限位开关的下限值。

## 参见

对硬件限位开关的要求 (页 101)

触发轴限位时的轴响应 (页 104)

在用户程序中可以更改位置限位的组态 (页 106)

组态 - 回原点 - 主动 (页 117)

## 5.4 组态定位轴工艺对象

### 触发轴限位时的轴响应

#### 逼近硬限位开关时的轴操作

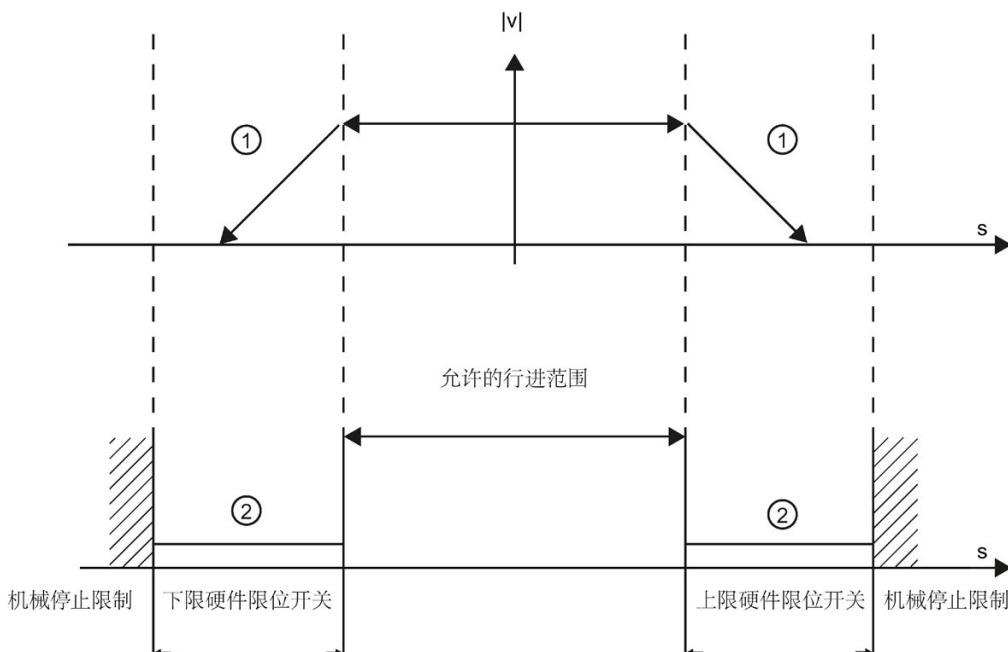
逼近硬限位开关时，根据驱动器连接，轴的动作会不同：

- 通过 PROFldrive / 模拟量输出连接驱动装置

逼近硬限位开关时，轴将禁用，并根据驱动器中的组态进行制动，并转入停止状态。为此，必须选择足够大的驱动器减速度，以使轴在机械挡块前可靠停止。

- 通过 PTO (Pulse Train Output) 连接驱动装置

逼近硬限位开关时，轴将以所组态的急停减速度制动直到停止。为此，必须选择足够大的急停减速度，以使轴在机械挡块前可靠停止。下图显示了轴逼近硬限位开关后的轴操作：



- ① PTO：轴以组态的紧急减速度制动。

PROFldrive 或类似驱动器接口：轴将锁定并以驱动器中组态的减速度制动直到停止。

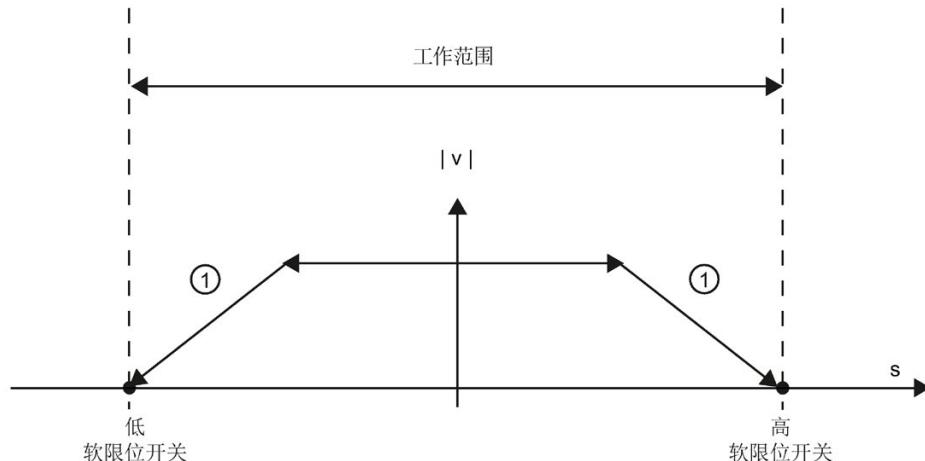
- ② 硬限位开关产生“已逼近”状态信号的范围。

“已逼近硬限位开关”错误将显示在正在启动的运动控制指令、“MC\_Power”和工艺对象变量中。有关用于清除错误的指令，请参见附录中的“ErrorID 和 ErrorInfo 列表”。

## 到达软限位开关时的轴操作

如果软限位开关激活，则在软限位开关所在的位置将停止当前的运动。轴将以所组态的减速度制动。

下图显示了轴到达软限位开关前的轴操作：



① 轴将以所组态的减速度制动直到停止。

“已逼近软限位开关”错误将显示在正在启动的运动控制指令、“MC\_Power”和工艺对象变量中。有关用于清除错误的指令，请参见附录中的“ErrorID 和 ErrorInfo 列表”。

超出软限位开关时，根据驱动器连接，轴的动作会不同：

- 通过 PROFIdrive / 模拟量输出连接驱动装置

超出软限位开关时，轴将禁用，并根据驱动器中的组态进行制动，并转入停止状态。

- 通过 PTO (Pulse Train Output) 连接驱动装置

有关超出软限位开关时的轴的动作，请参见“软件限位开关与回原点操作结合使用 (页 290)”和“软件限位开关与动态更改结合使用 (页 295)”部分。

如果机械停止块位于软限位开关的后面并且有发生机械损坏的风险，则需要使用附加的硬限位开关。

## 参见

对硬件限位开关的要求 (页 101)

组态 - 位置限制 (页 101)

在用户程序中可以更改位置限位的组态 (页 106)

## 5.4 组态定位轴工艺对象

### 在用户程序中可以更改位置限位的组态

可以在 CPU 中运行用户程序期间更改下列组态参数：

#### 硬限位开关

在用户程序运行期间，还可以激活和禁用硬限位开关。要执行该操作，可使用以下工艺对象变量：

- <轴名称>.PositionLimits\_HW.Active

有关组态参数的更改何时生效的信息，请参见附录中的工艺对象变量 (页 335)说明。

#### 软限位开关

在用户程序运行期间，还可以激活和禁用软限位开关并更改其位置值。要执行该操作，可使用以下工艺对象变量：

- <轴名称>.PositionLimits\_SW.Active

用于激活和取消激活软限位开关

- <轴名称>.PositionLimits\_SW.MinPosition

用于更改下限软限位开关的位置

- <轴名称>.PositionLimits\_SW.MaxPosition

用于更改上限软限位开关的位置

有关组态参数的更改何时生效的信息，请参见附录中的工艺对象变量 (页 335)说明。

### 参见

变量兼容表 V1...3 <-> V4...5 (页 58)

**MC\_ChangeDynamic:** 更改轴的动态设置 (V6 及以上版本) (页 274)

对硬件限位开关的要求 (页 101)

组态 - 位置限制 (页 101)

触发轴限位时的轴响应 (页 104)

#### 5.4.4.4 动态

##### 组态 - 动态 - 常规

可以在“常规动态”(General dynamics) 组态窗口中组态轴的最大速度、启动/停止速度、加速度和减速度以及加加速度限值（自定位轴工艺对象 V2 起）。

##### 速度限值的单位

从此下拉列表中可选择要用于设置速度限值的测量单位。此处单位的设置取决于“组态 > 基本参数 > 常规”(Configuration - General parameters > General) 中设置的测量单位。此处设置的单位仅用于简化输入。因此，可输入最大速度作为电机的速度值 (rpm)。

---

##### 说明

###### 圆整错误

如果在“速度限值的单位”(Unit of velocity limitation) 下拉列表中选择一个其它单位，而非“组态 > 基本参数 > 常规”(Configuration > Basic parameters > General)，则需注意可能会发生圆整错误。

---

##### 最大速度/启动/停止速度

在这些框中定义轴的最大允许速度和启动/停止速度。启动/停止速度是轴的最小允许速度，只能组态用于通过 PTO (Pulse Train Output) 的驱动器连接。

对于通过 PROFdrive 或模拟量输出的驱动器连接，启动/停止速度固定为 0。通过 PROFdrive 或模拟量输出进行连接时，最大速度可达所选测量单位的 1.0E12 倍（如，mm/s、°/s 等）。

## 5.4 组态定位轴工艺对象

### 定位轴工艺对象 (PTO) V4 及以上版本

信号板	速度 [脉冲/s]
20 kHz	$1 \leq \text{启动/停止速度} \leq 20000$ $1 \leq \text{最大速度} \leq 20000$
200 kHz	$1 \leq \text{启动/停止速度} \leq 200000$ $1 \leq \text{最大速度} \leq 200000$

板载 CPU 输出	速度 [脉冲/s]
100 kHz	$1 \leq \text{启动/停止速度} \leq 100000$ $1 \leq \text{最大速度} \leq 100000$
20 kHz	$1 \leq \text{启动/停止速度} \leq 20000$ $1 \leq \text{最大速度} \leq 20000$
1 MHz CPU 1217	$1 \leq \text{启动/停止速度} \leq 1000000$ $1 \leq \text{最大速度} \leq 1000000$

有关工艺对象定位轴 < V4 的限值, 请参见附录“与运动控制相关的 CPU 输出 (工艺版本 V1-3) (页 387)”。

最大速度值必须大于等于启动/停止速度值。

用户必须对其它测量单位所对应的限值进行相应地转换, 以便符合指定的机械系统。

### 加速度/减速度 - 加速时间/减速时间

在“加速时间”(Ramp-up time) 或“加速度”(Acceleration) 框中, 可在设置所需加速度。所需要的减速速度可以在“减速时间”(Ramp-down time) 或“减速度”(Deceleration) 框中设置。

加速时间与加速度、减速时间与减速度之间的关系如下面的方程所示:

$$\text{加速时间} = \frac{\text{最大速度} - \text{启动/停止速度}}{\text{加速度}}$$

$$\text{减速时间} = \frac{\text{最大速度} - \text{启动/停止速度}}{\text{减速度}}$$

用户程序中启动的运动作业将使用所选加速度/减速度执行。

有关通过基于 PTO (Pulse Train Output) 的驱动器连接时的加速度和减速度的限值，请参见“运动控制相关的 CPU 输出 (页 16)”部分。

---

#### 说明

更改速度限值（“启动/停止速度”和“最大速度”）将影响轴的加速度和减速度值。加速时间和减速时间保持不变。

---

### 启用工艺对象定位轴的加加速度限值、（V2 及以上版本）

使用该复选框，可启用加加速度限值。

如果激活了加加速度限值，则不会突然停止轴加速和轴减速，而是根据设置的步进或平滑时间逐渐调整。

---

#### 说明

在 V4 及以上版本中，该复选框将不再显示为工艺数据块中的参数。通过禁用该复选框，可将加加速度值设置为 0.0。

---

## 5.4 组态定位轴工艺对象

### 圆整时间/加加速度，定位轴工艺对象（V2 及以上版本）

在“滤波时间”(Smoothing time) 框或“加加速度”(Jerk) 框中，可以输入加加速度的参数。

- 在“加加速度”(Jerk) 框中，可以为加速和减速斜坡设置所需的加加速度。
- 在“滤波时间”(Smoothing time) 框中，可设置斜坡加速所需的滤波时间。

#### 说明

##### 滤波时间 V2-3

在组态中，设置的滤波时间仅适用于斜坡加速。

如果加速度值和减速度值不同，则根据斜坡加速的加加速度计算斜坡减速的滤波时间并使用该滤波时间。（另请参见“使用冲击限制时轴的行为（页 113）”）

调整减速度的滤波时间，如下所述：

- 加速度 > 减速度  
斜坡减速所用的滤波时间小于斜坡加速的滤波时间。
- 加速度 < 减速度  
斜坡减速所用的滤波时间大于斜坡加速的滤波时间。
- 加速度 = 减速度  
斜坡加速和减速的滤波时间相等。

滤波时间与加加速度之间的关系如下图中的公式所示：

$$\text{滤波时间 (斜坡加速)} = \frac{\text{加速度}}{\text{加加速度}}$$

$$\text{滤波时间 (斜坡减速)} = \frac{\text{减速度}}{\text{加加速度}}$$

用户程序中启动的运动作业将按照所选的加加速度进行。

有关通过基于 PTO (Pulse Train Output) 的驱动器连接时的加加速度限值，请参见“运动控制相关的 CPU 输出（页 16）”部分。

对于 PROFIdrive 驱动装置和带模拟器驱动接口的驱动装置，限值为 1E12。

## 参见

- 使用冲击限制时轴的行为 (页 113)
- 用于运动控制的硬件组件 (页 13)
- 运动控制相关的 CPU 输出 (页 16)
- 组态 - 动态 - 急停 (页 111)
- 在用户程序中可以更改动态组态 (页 114)

## 组态 - 动态 - 急停

在“动态急停”(Dynamics emergency stop) 组态窗口中，可以组态轴的急停减速度。出现错误或者禁用轴时，通过运动控制指令“MC\_Power”（输入参数 StopMode = 0 或 2）可以使用该减速度将轴制动至停止状态。

## 速度

“常规动态”(General dynamics) 组态窗口中组态的速度值，将再次显示在该信息区域中。

## 5.4 组态定位轴工艺对象

### 减速度

可在“急停减速速度”(Emergency stop deceleration) 或“急停减速时间”(Emergency stop ramp-down time) 字段中设置急停减速速度值。

急停减速时间和急停减速度之间的关系如下面的方程所示：

$$\text{急停减速时间} = \frac{\text{最大速度} - \text{启动/停止速度}}{\text{急停减速度}}$$

指定的急停减速速度必须足够大，保证可以在出现紧急情况时（例如，在到达机械挡块前逼近硬限位开关时）及时使轴停止。

必须基于组态的最大轴速度选择急停减速度。

限值：

下述限值采用“脉冲/秒<sup>2</sup>”作为测量单位。

- 自 CPU 固件 V3 起

0.005 ≤ 急停减速度 ≤ 9.5E9

- CPU 固件 V1...2

0.28 ≤ 急停减速度 ≤ 9.5E9

必须对其它测量单位所对应的限值进行相应地转换，以便符合指定的机械系统。

有关通过基于 PTO (Pulse Train Output) 的驱动器连接时的加加速度限值，请参见“运动控制相关的 CPU 输出 (页 16)”部分。

对于 PROFIdrive 驱动装置和带模拟器驱动接口的驱动装置，限值为 1.0E12。

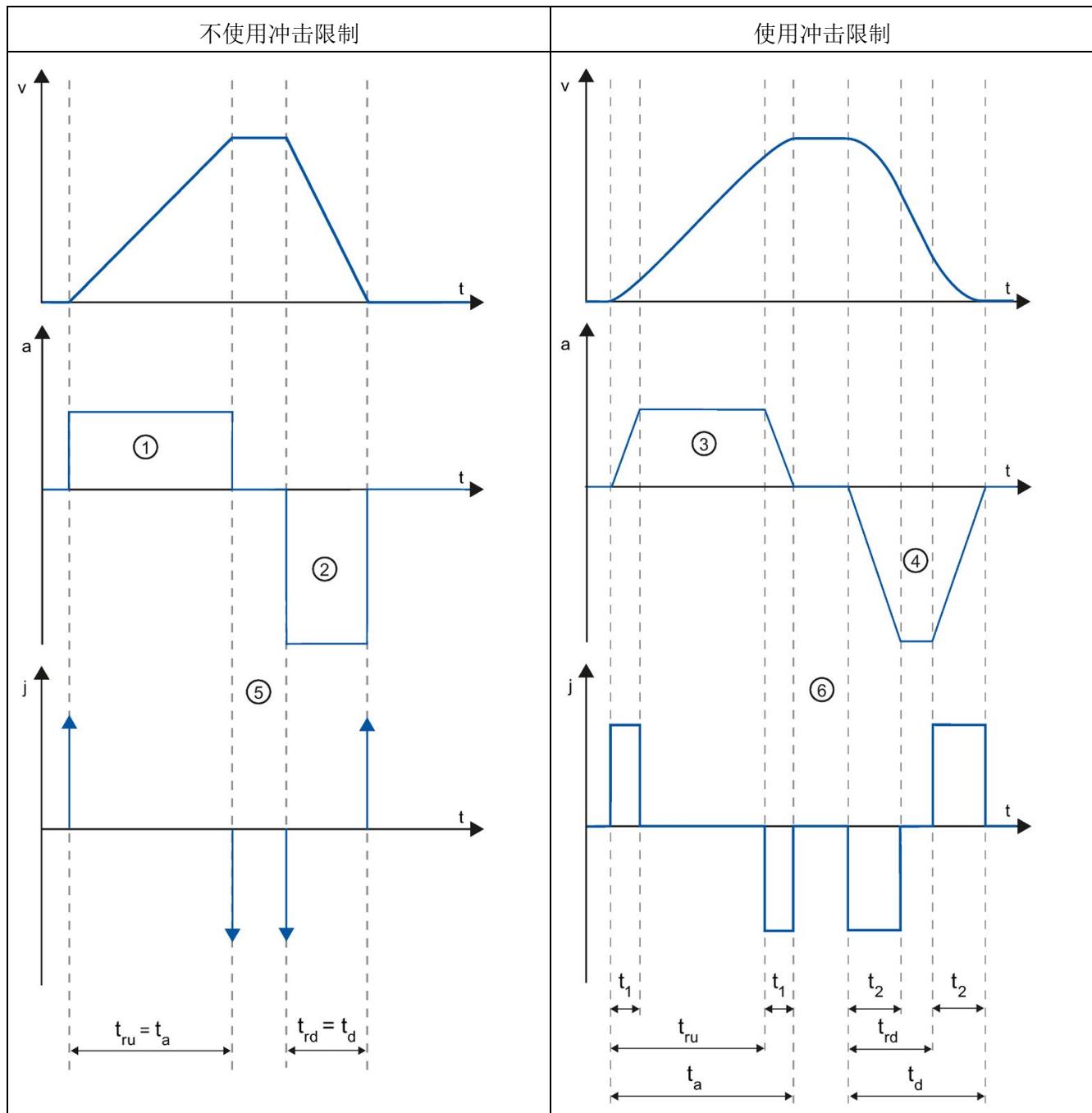
### 参见

组态 - 动态 - 常规 (页 107)

在用户程序中可以更改动态组态 (页 114)

### 使用冲击限制时轴的行为

如果激活了加速度限值，则不会突然停止轴加速和轴减速，而是根据设置的步进或平滑时间逐渐调整。下图详细显示了在激活和不激活冲击限制的情况下轴的行为：



## 5.4 组态定位轴工艺对象

<b>t</b>	时间轴
<b>v</b>	速度
<b>a</b>	加速度
<b>j</b>	加加速度
<b>t<sub>ru</sub></b>	加速时间
<b>t<sub>a</sub></b>	轴加速所用时间
<b>t<sub>rd</sub></b>	减速时间
<b>t<sub>d</sub></b>	轴减速所用时间
<b>t<sub>1</sub></b>	加速斜坡的平滑时间
<b>t<sub>2</sub></b>	减速斜坡的平滑时间

本示例说明减速度值 ② 是加速度值 ① 的二倍的行程。此时，得出的减速时间  $t_{rd}$  仅是加速时间  $t_{ru}$  的一半。

如果没有激活冲击限制，加速度 ① 和减速度 ② 将发生突变。如果激活了冲击限制器，则加速度 ① 和减速度 ② 将逐渐改变。由于冲击适用于整个运动，加速度的加速度率和减速度的减速度率相同。

在不使用冲击限制的情况下进行更改时，步进值 (j) 将变得无穷大 ⑥。如果激活了冲击限制，则步进将被限制为组态值 ⑥。

组态中给出的平滑时间  $t_1$  适用于加速斜坡。减速斜坡平滑时间  $t_2$  会根据组态的加加速度值以及组态的减速度进行计算。

## 参见

组态 - 动态 - 常规 (页 107)

## 在用户程序中可以更改动态组态

可以在 CPU 中运行用户程序期间更改下列组态参数：

## 加速度和减速度

在用户程序运行期间，还可以更改加速度和减速度值。要执行该操作，可使用以下工艺对象变量：

- <轴名称>.DynamicDefaults.Acceleration  
用于更改加速度
- <轴名称>.DynamicDefaults.Deceleration  
用于更改减速度

有关组态参数的更改何时生效的信息，请参见附录中的“工艺对象变量 (页 335)”说明。

## 急停减速度

在用户程序运行期间，还可更改急停减速度值。要执行该操作，可使用以下工艺对象变量：

- <轴名称>.DynamicDefaults.EmergencyDeceleration

有关组态参数的更改何时生效的信息，请参见附录中的工艺对象变量说明。

---

### 说明

更改该参数后，可能需要调整硬限位开关的位置以及其它安全相关的设置。

---

## 加加速度限值

还可在用户程序运行时激活和禁用加加速度限值，并更改加加速度值。为此，可使用工艺对象变量 <轴名称>.DynamicDefaults.Jerk。如果工艺对象的版本低于 V4，则变量 <轴名称>.Config.DynamicDefaults.JerkActive 必须设置为 TRUE，以便达到该加加速度的限值以及显示/用于该加加速度值的更改。

以下规则适用于 PTO 轴：

- 如果输入的加加速度值  $\geq 0.004$  脉冲/ $s^3$ ，则使用指定的值启用加加速度限值。
- 如果输入的加加速度值  $< 0.004$  脉冲/ $s^3$ ，则禁用加加速度限值。

对于位置控制轴，值为 0.0 时禁用加加速度限值；值  $> 0.0$  时，激活加加速度限值。

有关组态参数的更改何时生效的信息，请参见附录中的工艺对象变量说明。

## 5.4 组态定位轴工艺对象

### 参见

[在用户程序中更改动态值的组态（“轴”工艺对象 V1...3）\(页 401\)](#)

[变量兼容表 V1...3 <-> V4...5 \(页 58\)](#)

[MC\\_ChangeDynamic: 更改轴的动态设置 \(V6 及以上版本\) \(页 274\)](#)

[定位轴工艺对象 V4...5 的变量 \(页 485\)](#)

[组态 - 动态 - 常规 \(页 107\)](#)

[组态 - 动态 - 急停 \(页 111\)](#)

#### 5.4.4.5 回原点（自定位轴工艺对象 V2 起）

##### 组态 - 回原点 - 主动

在“主动回原点”(Active homing) 组态窗口中组态主动回原点所需的参数。运动控制指令“MC\_Home”的输入参数“Mode”= 3 时，会启动主动回原点。

##### 选择回原点模式（仅限通过 PROFIdrive V5 或更高版本的驱动器连接）

执行以下任一回原点模式：

- 使用基于 PROFIdrive 报文和接近开关的零位标记
- 使用基于 PROFIdrive 报文的零位标记
- 使用基于数字量输入的回原点标记

如果选择通过 PTO (Pulse Train Output) 或将 HSC 作为编码器的模拟量输出连接驱动装置，则唯一可用的回原点模式为“使用基于数字量输入的回原点标记”(Use homing mark via digital input)。

## 5.4 组态定位轴工艺对象

### 数字量输入

在此区域中，可组态回原点开关：

- **回原点开关输入**

在此域中为回原点开关选择数字量输入。

#### 说明

默认情况下，数字量输入的滤波时间设置为 6.4 ms。

数字量输入用作回原点开关的输入时，可能引起意外减速，从而导致出现误差。根据回原点速度和回原点开关的范围，可能检测不到回原点位置。可以在数字量输入设备组态的“输入滤波器”(Input filter) 中设置滤波时间。

选定的滤波时间必须小于待检测输入信号的脉冲持续时间，该时间可用作参考点开关。

通过 PTO (Pulse Train Output) 的驱动器连接：

该输入必须具有中断功能。板载 CPU 输入和所插入信号板输入都可选作回原点开关的输入。

- **选择信号电平**

在下拉列表中，选择回原点时使用的回原点开关电平。

- **允许在硬限位开关处自动反向**

激活该复选框可将硬限位开关用作回原点过程中的反向凸轮。只有启用硬限位开关才能实现反向控制（必须至少组态位于逼近方向上的硬限位开关）。

如果在主动回原点过程中到达硬限位开关，轴将以组态的减速度（不是以急停减速度）制动，然后反向。然后反向检测回原点开关。

如果未激活反向功能且在主动回原点过程中轴到达硬限位开关，则将因错误而中止回原点过程并以急停减速度对轴进行制动。

#### 说明

如果可能，采用以下措施之一以确保机器在发生反向时不会行进到机械挡块：

- 保持较低的行进速度。
- 增加组态的加速度/减速度。
- 增加硬限位开关和机械挡块之间的距离。

## 逼近/回原点方向

通过方向选择，可以决定主动回原点过程中搜索回原点开关的逼近方向以及回原点的方向。回原点方向指定执行回原点操作时轴用于逼近组态的回原点开关端的行进方向。

## 回原点开关侧

在此处可以选择轴是在回原点开关的上侧还是下侧进行回原点。

## 逼近速度

在该域中，可以指定回原点期间搜索回原点开关的速度。

限值（与所选测量单位无关）：

- 启动/停止速度  $\leq$  逼近速度  $\leq$  最大速度

## 回原点速度

在该域中，可以指定回原点期间逼近回原点开关的速度。

限值（与所选测量单位无关）：

- 启动/停止速度  $\leq$  回原点速度  $\leq$  最大速度

## 起始位置偏移值

如果指定的回原点位置与回原点开关的位置存在偏差，则可在此域中指定起始位置偏移量。

如果该值不等于 0，轴在回原点开关处回原点后将执行以下动作：

1. 以回原点速度使轴移动起始位置偏移值指定的一段距离
2. 达到“起始位置偏移值”时，轴处于运动控制指令“MC\_Home”的输入参数“Position”中指定的起始位置处。

限值（与所选测量单位无关）：

- $-1.0e12 \leq$  起始位置偏移值  $\leq 1.0e12$

## 起始位置

将运动控制指令“MC\_Home”中所组态的位置用作起始位置。

## 5.4 组态定位轴工艺对象

### 组态 - 回原点 - 被动

在“回原点 - 被动”(Homing - Passive) 组态窗口中，可以组态被动回原点所需的参数。

被动回原点的移动必须由用户触发（例如，使用轴运动命令）。运动控制指令“MC\_Home”的输入参数“Mode”= 2 时，会启动被动回原点。

### 选择回原点模式（仅限通过 PROFIdrive V5 或更高版本的驱动器连接）

执行以下任一回原点模式：

- 使用基于 PROFIdrive 报文和接近开关的零位标记

系统将检查到达接近开关的时间。在到达接近开关并置于指定的回原点方向后，可通过 PROFIdrive 报文启用零位标记检测。在预先选定的方向上到达零位标记后，会将工艺对象的实际位置设置为回原点标记位置。

- 使用基于 PROFIdrive 报文的零位标记

当工艺对象的实际值按照指定的回原点方向移动时，系统将立即启用零位标记检测。在指定的回原点方向上到达零位标记后，会将工艺对象的实际位置设置为回原点标记位置。

- 使用基于数字量输入的回原点标记

当轴或编码器的实际值在指定的回原点方向上移动时，系统将立即检查数字量输入的状态。在指定的回原点方向上到达回原点标记（数字量输入的设置）后，会将工艺对象的实际位置设置为回原点标记位置。

如果选择了通过 PTO (Pulse Train Output) 的驱动器连接，则默认使用基于数字量输入的回原点标记。

## 数字量输入

在此区域中，可组态回原点开关：

- 回原点开关输入

在此域中为回原点开关选择数字量输入。该输入必须具有中断功能。板载 CPU 输入和所插入信号板输入都可选作回原点开关的输入。

---

### 说明

默认情况下，数字量输入的滤波时间设置为 6.4 ms。

数字量输入用作回原点开关的输入时，可能引起意外减速，从而导致出现误差。根据回原点速度和回原点开关的范围，可能检测不到回原点位置。可以在数字量输入设备组态的“输入滤波器”(Input filter) 中设置滤波时间。

选定的滤波时间必须小于待检测输入信号的脉冲持续时间，该时间可用作参考点开关。

---

- 选择信号电平

在下拉列表中，选择回原点时使用的回原点开关电平。

## 回原点开关侧

在此处可以选择轴是在回原点开关的上侧还是下侧进行回原点。

## 起始位置

将运动控制指令“MC\_Home”中所组态的位置用作起始位置。

---

### 说明

如果未使用轴运动命令进行被动回原点（轴处于停止状态），则将在下一个回原点开关的上升沿或下降沿处执行回原点操作。

---

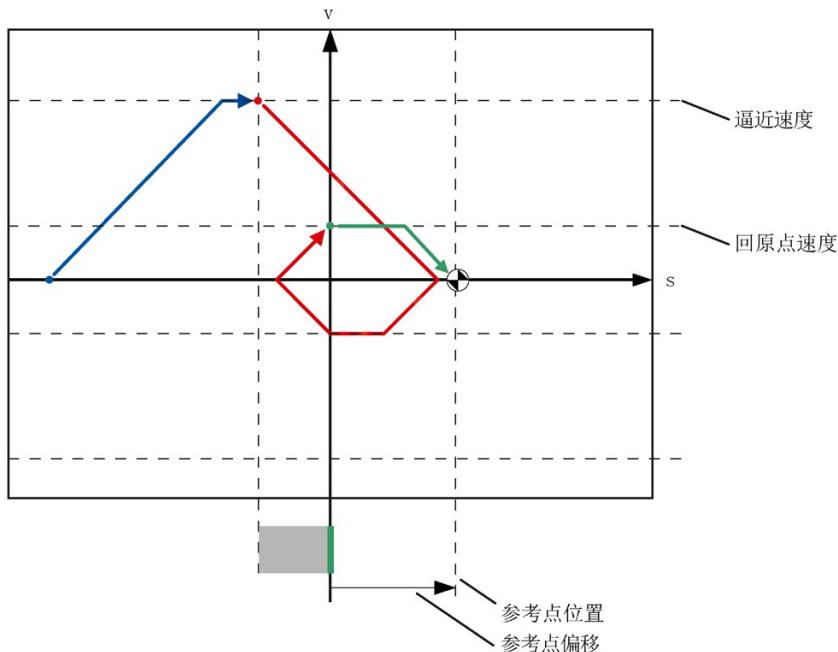
## 5.4 组态定位轴工艺对象

### 顺序 - 主动回原点

使用运动控制指令“MC\_Home”（输入参数 Mode = 3），可以启动主动回原点。“Position”输入参数指定绝对回原点位置。另外，若要进行测试，也可在轴控制面板中启动主动回原点。

下图举例说明了使用以下组态参数时主动起始位置逼近的特征曲线：

- “回原点模式”=“使用基于数字量输入的回原点标记”
- “逼近/回原点方向”=“正方向”
- “回原点开关侧”=“上侧”
- “回原点位置偏移”值 > 0



### 搜索回原点开关（蓝色曲线部分）

主动回原点开始后，轴加速到组态的“逼近速度”并以该速度搜索回原点开关。变量 <轴名称>.StatusBits.HomingDone 设置为 FALSE。

### 参考点逼近（红色曲线段）

检测到回原点开关时，本示例中的轴将制动并反向，以组态的回原点速度向组态的回原点开关侧回原点。回原点运动会使变量 <轴名称>.StatusBits.HomingDone 变为 TRUE。

## 行进到回原点位置偏移值（绿色曲线段）

回原点后，轴以回原点速度沿该路径移动到回原点位置偏移值。此时，轴处于运动控制指令“MC\_Home”的输入参数“Position”中指定的回原点位置。

## 参见

组态 - 回原点 - 常规（轴工艺对象 V2...3）(页 398)

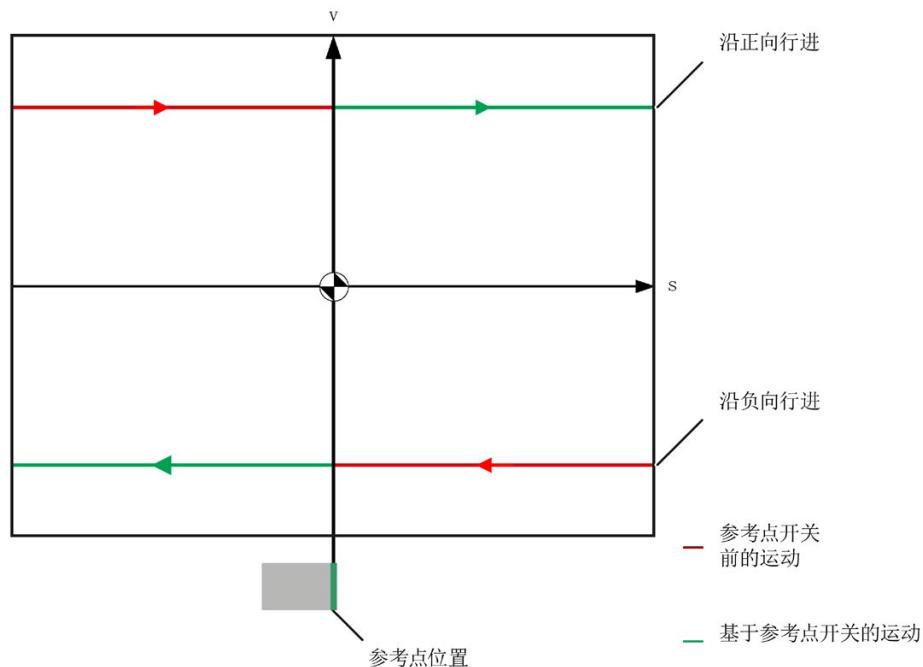
## 5.4 组态定位轴工艺对象

### 顺序 - 被动回原点

通过运动控制指令“MC\_Home”（输入参数 Mode = 2）启动被动回原点。输入参数“Position”指定绝对参考点位置。

下图举例说明了使用以下组态参数时被动回原点的特性曲线：

- “回原点开关侧”=“上侧”
- “回原点模式”=“使用基于数字量输入的回原点标记”



### 向回原点开关方向移动（曲线的红色部分）

启动被动回原点时，运动控制指令“MC\_Home”本身并不执行任何回原点运动。必须由用户通过其它运动控制指令（如“MC\_MoveRelative”）来执行到达回原点开关所需的行程。如果轴已回到原点，则被动回原点过程中，变量 <轴名称>.StatusBits.HomingDone 保持为 TRUE。

### 轴回原点（由曲线的红色部分过渡到绿色部分）

到达回原点开关的组态侧时，轴会回原点。将当前的轴位置设置为起始位置。这在运动控制指令“MC\_Home”的“Position”参数中指定。如果之前轴未回原点，则会将变量 <轴名称>.StatusBits.HomingDone 设置为“TRUE”。之前启动的行程不会取消。

## 移动至超过回原点开关（曲线的绿色部分）

在回原点开关处回原点后，轴继续移动并以正确的轴位置完成之前启动的行程。

## 在用户程序中更改回原点组态

利用版本 V2 及以上版本的定位轴工艺对象，您可以在 CPU 中运行用户程序期间更改下列组态参数：

### 被动归位

可以在用户程序运行期间更改用于被动归位的归位开关端。要执行该操作，可使用以下工艺对象变量：

- <轴名称>.Sensor[1].PassiveHoming.SideInput  
用于更改归位开关侧
- <轴名称>.Sensor[1].PassiveHoming.Mode  
用于更改归位速度

有关组态参数的更改何时生效的信息，请参见附录中的工艺对象变量 (页 485)说明。

## 5.4 组态定位轴工艺对象

### 主动归位

可以在用户程序运行期间更改主动归位的逼近方向、归位开关侧、逼近速度、归位速度和起始位置偏移量。要执行该操作，可使用以下工艺对象变量：

- <轴名称>.Homing.AutoReversal  
用于更改“在硬限位开关处自动反向”
- <轴名称>.Homing.ApproachDirection  
用于更改逼近/归位方向
- <轴名称>.Sensor[1].ActiveHoming.SideInput  
用于更改归位开关侧
- <轴名称>.Homing.ApproachVelocity  
用于更改逼近速度
- <轴名称>.Homing.ReferencingVelocity  
用于更改归位速度
- <轴名称>.Sensor[1].ActiveHoming.HomePositionOffset  
用于更改起始位置偏移
- <轴名称>.Sensor[1].ActiveHoming.Mode  
用于更改归位速度

有关组态参数的更改何时生效的信息，请参见附录中的工艺对象变量说明。

### 参见

变量兼容表 V1...3 <-> V4...5 (页 58)

MC\_ChangeDynamic: 更改轴的动态设置 (V6 及以上版本) (页 274)

#### 5.4.4.6 定位监控

##### 组态 - 定位监控（仅限 PROFIdrive 和模拟驱动器接口）

在“定位监控”(Positioning Monitoring) 组态窗口中，组态用于监控目标位置的标准。

定位监控功能将在设定值计算结束时对实际位置的状态进行监控。一旦速度设定值达到值 0，实际位置值就必须介于定位窗口的容差时间范围内。实际值在定位窗口内的停留时间必须超出最短停留时间。

当实际位置在容差时间内到达定位窗口，且在最短停留时间内停留在该窗口，则置位状态位 `<轴名称>.StatusBits.Done`。这样就完成了一个运动命令。

各种设定值插补操作方式中的定位监控都相同。例如，设定值插补完成的方式如下所示：

- 设定值到达目标位置
- 通过运动控制指令“MC\_Halt”，在运动期间用位置控制停止

在以下情况下，通过定位监控停止轴，并且定位错误 (ErrorID 16#800F) 显示在运动控制指令中：

- 在容差时间内，实际值未到达定位窗口。
- 在最短停留时间内，实际值离开定位窗口。

##### 定位窗口

在该域中，组态定位窗口的大小。

##### 容差时间

在该域中，组态容差时间。在该容差时间内，位置值必须到达定位窗口。

##### 在定位窗口停留的最短时间

在该域中，组态最短停留时间。在最短停留时间内，实际位置值必须位于定位窗口中。

## 5.4 组态定位轴工艺对象

### 组态 - 跟随错误（仅限 PROFIdrive 和模拟驱动器接口）

在“跟随误差”(Following error) 组态窗口中，组态轴的实际位置与位置设定值之间的容许偏差。

跟随误差是轴的位置设定值与实际位置值之间的差值。计算跟随误差时，会将设定值到驱动器的传输时间、实际位置值到控制器的传输时间计算在内。

根据一个与速度有关的跟随误差限值对跟随误差进行监视。允许跟随误差取决于速度设定值。

当速度小于一个可调整的速度下限时，将允许跟随误差指定为常数。而高于该速度下限值时，允许跟随误差则随速度设定值按比例增长。在最大速度下可达到最大允许跟随误差。

超出允许跟随误差时，轴停止，错误 (ErrorID 16#800D) 显示在运动控制指令中。

#### 启用跟随误差监控

如果想启用跟随误差监控功能，则选中该复选框。

在启用跟随误差监控时，轴在错误范围（橙色）内停止。

#### 最大跟随误差

在该域中，组态最大速度时容许的跟随误差。

#### 跟随误差

在该域中，组态低速度时的容许跟随误差（无动态调整）。

#### 启动动态调整

在该域中，组态一个速度；超过该速度时，将会动态调整跟随误差。超过该速度时，将会调整跟随误差，直至达到最大速度时的最大跟随误差。

#### 最大速度

该框显示在“动态 > 常规”(Dynamics > General) 下组态的最大速度。

### 组态 - 停止信号（仅限 PROFIdrive 和模拟驱动器接口）

在“停止信号”(Standstill signa) 组态窗口中，组态停止检测标准。

要显示停止 (<轴名称>.StatusBits.StandStill)，轴速度必须在停止窗口内保持最短停留时间。

### 停止窗口

在该域中，组态停止窗口的大小。

### 在停止窗口停留的最短时间

在该域中，组态停止窗口中的最短停留时间。

## 5.4 组态定位轴工艺对象

### 5.4.4.7 组态 - 控制环 (仅限 PROFIdrive 和模拟驱动器接口)

在“控制回路”(Control loop) 组态窗口中，组态位置控制回路的预控制和增益  $K_v$ 。

$K_v$  因数影响以下参数：

- 定位精度和停止控制
- 运动的一致性
- 定位时间

轴的机械状态（硬度越高）越好，可以组态的  $K_v$  因数越大。这样可以减小跟随误差，实现更快的动态响应。

通过“Tuning (页 192)”(自整定) 功能，可确定控制轴位置的最佳增益。

#### 预控制

在此框中，组态位置控制回路的百分比速度预控制。

#### 增益 ( $K_v$ 因子)

在该域中，组态位置控制回路的增益  $K_v$ 。

## 5.4.5 参数视图

### 5.4.5.1 参数视图简介

参数视图提供了工艺对象中所有相关参数的一般概述。可获得参数设置的概述，并可在离线和在线模式下轻松地对其进行更改。

The screenshot shows the 'Parameter view' (Parameter view) interface in TIA Portal. The interface consists of several main components:

- Top Bar:** Includes tabs for '功能视野' (Function View) and '参数视图' (Parameter View), along with other standard window controls.
- Toolbar:** Located above the navigation tree, it contains icons for search, filter, and export.
- Navigation Tree:** On the left, it shows a hierarchical tree of parameters under '所有参数' (All Parameters). Key categories include '组态参数' (Configurable parameters), '控制器类型' (Controller type), 'Input/Output 参数' (Input/Output parameters), '过程值设置' (Process value settings), '执行器设置' (Actuator settings), '高级设置' (Advanced settings), and '调试参数' (Debug parameters).
- Parameter Table:** The main content area displays a table of parameters. The columns are: 功能视图中的名称 (Name in Parameter View), 在 DB 中的名称 (Name in DB), 项目起始值 (Initial Value), 数据类型 (Data Type), and 注释 (Notes). The table includes rows for various parameters such as 替代输出值 (SavePosition), 输出值的上限 (OutputUpperLimit), and 输出值的下限 (OutputLowerLimit).

- ① “参数视图”(Parameter view) 选项卡
- ② 工具栏 (页 134)
- ③ 导航 (页 135)
- ④ 参数表 (页 136)

## 5.4 组态定位轴工艺对象

### 功能范围

提供以下可用于分析工艺对象参数和启用目标性监视与修改的功能。

显示功能:

- 在离线和在线模式下显示参数值
- 显示参数的状态信息
- 显示值偏差和直接连接选项
- 显示组态错误
- 显示由参数引起的值更改
- 显示某参数所有的存储值: PLC 起始值、项目起始值、监视值
- 显示参数存储值的参数比较

操作员控制功能:

- 为在参数之间和参数结构之间进行快速更改而导航。
- 用于更快搜索具体参数的文本过滤器。
- 用于按需自定义参数和参数组顺序的排序功能。
- 用于备份参数视图的结构设置的存储功能。
- 在线监视和修改参数值。
- 更改值的显示格式。
- 为捕获并响应瞬时情况而保存 CPU 参数值快照的功能。
- 用于将参数值快照应用为起始值的功能。
- 将已修改的起始值下载至 CPU。
- 用于比较两个参数值的比较功能。

## 有效性

此处所述的“参数视图”(Parameter view) 适用于以下工艺对象：

- PID\_Compact
- PID\_3Step
- PID\_Temp
- CONT\_C (仅适用于 S7-1500)
- CONT\_S (仅适用于 S7-1500)
- TCONT\_CP (仅适用于 S7-1500)
- TCONT\_S (仅适用于 S7-1500)
- TO\_Axis\_PTO (S7-1200 运动控制)
- TO\_Positioning\_Axis (S7-1200 运动控制)
- TO\_CommandTable\_PTO (S7-1200 运动控制)
- TO\_CommandTable (S7-1200 运动控制)

## 5.4 组态定位轴工艺对象

### 5.4.5.2 参数视图结构

#### 工具栏

可在参数视图的工具栏中选择以下功能。

图标	功能	说明
	监视全部	在活动的参数视图中启动可见参数监视（在线模式）。
	创建监视值的快照并 将该快照的设定值接 受为起始值	将当前的监视值应用到“快照”(Snapshot) 列，并更新项目中的起始值。 仅可在 PID_Compact、PID_3Step 和 PID_Temp 的在线模式下执行。
	加载设定值的起始值 作为实际值（初始化 设定值）	把在项目中更新过的起始值传送至 CPU。 仅可在 PID_Compact、PID_3Step 和 PID_Temp 的在线模式下执行。
	创建监视值的快照	将当前的监视值应用到“快照”(Snapshot) 列。 仅可在在线模式下执行。
	请立即一次性修改全 部选定参数	该命令尽快执行一次，而不参考用户程序中的任何特定点。 仅可在在线模式下执行。
	选择导航结构	在功能导航和数据导航之间进行切换。
	文本过滤器...	在输入字符串之后：显示某一当前可见列中所有包括指定字符串在内的参数。
	选择比较值	在线模式下，选择要与另一个参数值进行比较的参数值 (项目起始值、PLC 起始值、快照)。 仅可在在线模式下执行。
	保存窗口设置	为参数视图保存显示设置（例如选择的导航结构和激活的表中的列等）

## 导航

在“参数视图”(Parameter view) 选项卡中，有以下替代导航结构可供选择。

导航		说明
功能导航	<ul style="list-style-type: none"><li>▼ 所有参数<ul style="list-style-type: none"><li>▶ 组态参数</li><li>▶ 调试参数</li><li>其它参数</li></ul></li></ul>	在功能导航中，参数结构以组态对话框（“功能视图”(Functional view) 选项卡）、调试对话框和诊断对话框中的结构为基础。 最后一个组“其它参数”(Other parameters) 包括工艺对象的所有其它参数。
数据导航	<ul style="list-style-type: none"><li>▼ 所有参数<ul style="list-style-type: none"><li>Input</li><li>Output</li><li>InOut</li><li>▶ Static</li><li>其它参数</li></ul></li></ul>	在数据导航中，参数结构以背景数据块/工艺数据块中的结构为基础。 最后一个组“其它参数”(Other parameters) 包括背景数据块/工艺数据库中不包括的参数。

可以使用“选择导航结构”(Select navigation structure) 下拉列表来切换导航结构。

## 5.4 组态定位轴工艺对象

### 参数表

下表给出了参数表各列的含义。可以根据需要显示或隐藏列。

- “离线”列 = X：该列在离线模式下可见。
- “在线”列 = X：该列在在线模式下可见（在线连接到 CPU）。

列	说明	离线	在线
功能视图中的名称	功能视图中的参数名称。 如果参数未通过工艺对象组态，该显示字段留空。	X	X
DB 中的全称	背景数据块/工艺数据块中参数的完整路径。 如果参数未包含在背景数据块/工艺数据块中，该显示字段留空。	X	X
在 DB 中的名称	背景数据块/工艺数据块中的参数名称。 如果参数是某结构或 UDT 的一部分，则应添加前缀“. .”。 如果参数未包含在背景数据块/工艺数据块中，该显示字段留空。	X	X
组态的状态	使用状态符号显示组态完整性。 请参见组态的状态（离线）(页 147)	X	
比较结果	“比较值”功能的结果。 如果已在线连接并选择了“监视所有”(Monitor all) 按钮  ，则会显示该列。		X
项目起始值	在项目中组态起始值。 如果输入的值具有语法错误或过程相关错误，则会出现错误指示。	X	X
默认值	为该参数预分配的值。 如果参数未包含在背景数据块/工艺数据块中，该显示字段留空。	X	X
快照	CPU 中当前值的快照（监视值）。 如果值具有过程相关错误，则会出现错误指示。	X	X
PLC 起始值	CPU 中的起始值。 如果已在线连接并选择了“监视所有”(Monitor all) 按钮  ，则会显示该列。 如果值具有过程相关错误，则会出现错误指示。		X
监视值	CPU 中的当前值。 如果已在线连接并选择了“监视所有”(Monitor all) 按钮  ，则会显示该列。 如果值具有过程相关错误，则会出现错误指示。		X

列	说明	离线	在线
修改值	<p>用于更改监视值的值。</p> <p>如果已在线连接并选择了“监视所有”(Monitor all) 按钮 , 则会显示该列。</p> <p>如果输入的值具有语法错误或过程相关错误, 则会出现错误指示。</p>		X
选择用于传输 	<p>使用“请立即一次性修改全部选定参数”(Modify all selected parameters immediately and once) 按钮选择要传输的修改值。</p> <p>该列与“修改值”(Modify value) 列一起显示。</p>		X
最小值	<p>参数的最小过程相关值。</p> <p>如果最小值取决于其它参数, 则将其定义为:</p> <ul style="list-style-type: none"> <li>• 离线: 由项目起始值决定。</li> <li>• 在线: 由监视值决定。</li> </ul>	X	X
最大值	<p>参数的最大过程相关值。</p> <p>如果最大值取决于其它参数, 则将其定义为:</p> <ul style="list-style-type: none"> <li>• 离线: 由项目起始值决定。</li> <li>• 在线: 由监视值决定。</li> </ul>	X	X
设定值	将参数指定为设定值。可在线初始化这些参数。	X	X
数据类型	<p>变量的数据类型。</p> <p>如果参数未包含在背景数据块/工艺数据块中, 该显示字段留空。</p>	X	X
保持性	<p>将值指定为保持值。</p> <p>保持性参数的值将保留, 即使在电源关闭后也是如此。</p>	X	X
可从 HMI 访问	指示运行期间 HMI 是否可以访问此参数。	X	X
HMI 中可见	指示 HMI 选择列表中的参数是否默认可见。	X	X
注释	参数的简要描述。	X	X

## 5.4 组态定位轴工艺对象

### 5.4.5.3 打开参数视图

#### 要求

工艺对象已添加到项目树中，即已创建指令的相关背景数据块/工艺数据块。

#### 步骤

1. 在项目树中打开“工艺对象”(Technology objects) 文件夹。
2. 在项目树中打开该工艺对象。
3. 双击“组态”(Configuration) 对象。
4. 选择右上角的“参数视图”(Parameter view) 选项卡。

#### 结果

参数视图将打开。所显示的每个参数都会以参数表中的一行表示。

可显示的参数属性（表的列）根据所使用的参数视图模式（离线或在线）而有所不同。

此外，可以有选择地显示和隐藏表的各个列。

#### 参见

[参数视图默认设置 \(页 139\)](#)

#### 5.4.5.4 参数视图默认设置

##### 默认设置

为了让您高效使用参数视图，可以自定义参数显示并保存设置。

可以执行并保存下列自定义项：

- 显示和隐藏列
- 改变列宽
- 改变列的顺序
- 切换导航
- 在导航中选择参数组
- 选择比较值

##### 显示和隐藏列

要显示或隐藏参数表中的列，请按以下步骤操作：

1. 将光标放在参数表的标题上。
2. 在快捷菜单中选择“显示/隐藏”(Show/Hide) 命令。  
将显示可选的列。
3. 要显示列，请选中该列的复选框。
4. 要隐藏列，请清除该列的复选框。

或

1. 将光标放在参数表的标题上。
2. 如果要显示所有离线或在线模式的列，请在快捷菜单中选择“显示所有列”(Show all columns) 命令。

一些列只能在在线模式下显示：请参见参数表 (页 136)。

## 5.4 组态定位轴工艺对象

### 改变列宽

要自定义列宽以便阅读所有行中的文本，请按以下步骤操作：

1. 将光标放在参数表标题中要进行自定义的列的右侧，直到光标的形状变为十字。
2. 然后双击此位置。

或

1. 打开参数表标题上的快捷菜单。
2. 点击
  - “优化列宽”(Optimize column width) 或
  - “优化所有列的宽度”(Optimize width of all columns)。

如果列宽设置得过窄，可将光标短暂停在相关字段上，即可出现各个字段的完整内容。

### 改变列的顺序

参数表的列可按照任意方式排列。

要更改列的顺序，请按以下步骤操作：

1. 单击列标题并使用拖放操作将其移动至所需位置。

当松开鼠标时，对象会锚定在新位置上。

### 切换导航

要切换参数的显示格式，请执行以下步骤：

1. 在“选择导航结构”(Select navigation structure) 下拉列表中选择所需的导航。
  - 数据导航
  - 功能导航

另请参见导航 (页 135)。

### 在导航中选择参数组

在所选导航内，可在显示“所有参数”(All parameters) 或显示所选的下级参数组二者中选择其一。

1. 在导航中单击所需的参数组。

参数表仅显示该参数组的参数。

## 选择比较值（在线）

要为“比较值”(Compare values) 功能设置比较值，请按下列步骤操作：

1. 在“选择比较值”(Selection of compare values) 下拉列表中选择所需的比较值。
  - 项目起始值/PLC 起始值
  - 项目起始值/快照
  - PLC 起始值/快照

默认情况下，会设置“项目起始值/PLC 起始值”(Start value project / Start value PLC) 选项。

## 保存“参数视图”(Parameter view) 的默认设置

要保存参数视图的以上自定义项，请按下列步骤操作：

1. 根据需要自定义参数视图。
2. 单击参数视图右上角的“保存窗口设置”(Save window settings) 按钮。

## 5.4.5.5 使用参数视图

## 概述

如下文所述，下表提供了在线和离线状态下的参数视图功能总览。

- “离线”列 = X： 只能在离线模式下使用本功能。
- “在线”列 = X： 只能在在线模式下使用本功能。

功能/操作	离线	在线
过滤参数表 (页 142)	X	X
将参数表排序 (页 143)	X	X
将参数数据传送给其它编辑器 (页 144)	X	X
指示错误 (页 145)	X	X
在项目中编辑起始值 (页 145)	X	X
组态的状态 (离线) (页 147)	X	
参数视图中的在线监视值 (页 147)		X
创建监视值的快照 (页 150)		X
修改值 (页 151)		X
比较值 (页 152)		X
将来自在线程序的值应用为起始值 (页 154)		X
初始化在线程序中的设定值 (页 155)		X

## 过滤参数表

可通过下列方式过滤参数表中的参数：

- 使用文本过滤器
- 使用导航子组

两种过滤方法可同时使用。

## 使用文本过滤器

可过滤参数表中可见的文本。这表示只可以过滤显示的参数行和参数列中的文本。

1. 在“文本过滤器...”(Text filter...) 输入框中输入所需字符串以进行过滤。

参数表仅显示包含该字符串的参数。

文本过滤器会在下列情况下复位。

- 当选择了导航中的另一个参数组时。
- 当导航从数据导航更改为功能导航，或从功能导航更改为数据导航时。

## 使用导航子组

1. 在导航中单击所需的参数组，例如“静态”(Static)。

参数表仅会显示静态参数。可以为导航的一些组选择其它子组。

2. 如果要再次显示所有参数，请在导航中单击“所有参数”(All parameters)。

## 将参数表排序

参数值按行排列。参数表可以按照任意显示的列进行排序。

- 包含数字值的列会按照数字值的峰值进行排序。
- 文本列会按照字母顺序进行排序。

## 按列排序

1. 将光标放在所需列的标题单元格中。

该单元格的背景会变为蓝色。

2. 单击列标题。

## 5.4 组态定位轴工艺对象

### 结果

整个参数表会按照所选的列进行排序。列标题中会出现一个尖向上的三角形。

再次单击列标题会按以下情况更改排序方式：

- 符号“▲”：参数表按升序排序。
- 符号“▼”：参数表按降序排序。
- 无符号：再次移除排序。参数表采用默认显示。

排序时，忽略“在 DB 中的名称”列的“..”前缀。

### 将参数数据传送给其它编辑器

在选择了参数表的整个参数行之后，可以使用以下操作：

- 拖放
- <Ctrl+C>/<Ctrl+V>
- 通过快捷菜单中的复制/粘贴

将参数传送给 TIA Portal 的以下编辑器：

- 程序编辑器
- 监视表
- 用于跟踪功能的信号表

参数会以全称插入：请参见“DB 中的全称”(Full name in DB) 列中的信息。

## 指示错误

### 错误指示

导致编译错误（例如越限）的参数分配错误将在参数视图中指示。

每当在参数视图中输入值，就会检查是否有过程相关错误和语法错误，并将结果表示出来。

错误的值会由以下方法表示：

- “组态的状态”（离线模式）或“比较结果”（在线模式，取决于所选的比较类型）列中的红色错误符号

和/或

- 背景为红色的表字段

如果单击错误的字段，会出现弹出错误消息，其中包含有关允许的值范围或所需语法（格式）的信息

## 编译错误

如果参数未显示在组态对话框中，可以从编译器的错误消息处直接打开包含出错参数的参数视图（功能导航）。

## 在项目中编辑起始值

可使用参数视图在离线模式和在线模式下编辑项目中的起始值。

- 可在参数表的“项目起始值”(Start value project) 列中更改值。
- 在参数表的“组态状态”(Status of configuration) 列中，会通过与工艺对象组态对话框中相似的状态符号来表示组态进度。

## 约束条件

- 如果其它参数取决于那些起始值发生更改的参数，那么这些相关参数的起始值也会发生调整。
- 如果工艺对象的参数不可编辑，则也无法在参数视图中对其进行编辑。参数是否可以编辑还取决于其它参数值。

## 5.4 组态定位轴工艺对象

### 定义新起始值

要在参数视图中定义参数的起始值，请按下列步骤操作：

1. 打开工艺对象的参数视图。
2. 在“项目起始值”(Start value project) 列中输入所需的起始值。该值必须与参数的数据类型相匹配，不能超过参数的值范围。

“最大值”(Maximum value) 和“最小值”(Minimum value) 列中给出了值范围的限值。

“组态的状态”(Status of configuration) 列用彩色符号表示组态进度。

另请参见 [组态的状态（离线）（页 147）](#)

在调整了起始值并将工艺对象下载到 CPU 的情况下，如果未将参数声明为保持（“保留”(Retain) 列），则参数会在启动时采用定义值。

### 错误指示

当输入了起始值时，会检查是否有过程相关错误和语法错误，并将结果表示出来。

错误的起始值会由以下方法表示：

- “组态的状态”（离线模式）或“比较结果”（在线模式，取决于所选的比较类型）列中的红色错误符号
- 和/或
- “项目起始值”(Start value project) 字段中的红色背景  
如果单击错误的字段，会出现弹出错误消息，其中包含有关允许的值范围或必要的语法（格式）的信息。

### 更正错误起始值

1. 使用来自弹出错误消息的信息更正错误的起始值。

将不再显示红色错误信息、红色字段背景和弹出错误消息。

除非起始值无误，否则项目将无法成功编译。

## 组态的状态（离线）

表示组态状态的图标位于：

- 参数表中的“组态的状态”(Status of configuration) 列中
- 功能导航和数据导航的导航结构中

### “组态的状态”列中的符号

符号	含义
✓	参数的起始值对应于默认值且有效。用户尚未定义起始值。
✓	参数的起始值中包含用户定义或自动调整的值。起始值与默认值不同。该起始值无误且有效。
✗	参数的起始值无效（语法或过程相关错误）。 输入框的背景为红色。单击弹出错误消息，会指出错误原因。
⚠	仅限 S7-1200 运动控制： 参数的起始值有效但包含警告。 输入框的背景为黄色。
●	在当前组态中，该参数不相关。

## 导航中的符号

导航中的符号指示组态过程的方式与工艺对象组态对话框中的方式相同。

## 参数视图中的在线监视值

可直接在参数视图中监视 CPU 中工艺对象参数当前采用的值（监视值）。

## 要求

- 需要有在线连接。
- 工艺对象已下载到 CPU 中。
- 程序执行处于激活状态（CPU 处于“RUN”模式）。
- 工艺对象的参数视图已打开。

## 5.4 组态定位轴工艺对象

### 步骤

1. 单击  启动监视。

一旦参数视图在线，将额外显示以下各列：

- 比较结果
- PLC 起始值
- 监视值
- 修改值
- 选择用于传输

“监视值”(Monitor value) 列显示了 CPU 上的当前参数值。

各附加列的含义：请参见参数表 (页 136)

2. 再次单击  停止监视。

### 显示

所有仅在线时可用的列以橙色背景显示：

- 浅橙色单元格  中的值可以更改。
- 背景为深橙色  的单元格中的值无法更改。

## 更改值的显示格式

通过工艺对象的参数视图表行中的快捷菜单，可以选择值的显示格式。

无论是在线模式还是离线模式，均可更改下列值的显示格式：

- 项目起始值
- PLC 起始值
- 最大值
- 最小值
- 快照
- 监视值
- 默认值
- 修改值

设定的显示格式应用于表行的所有值。

可更改值的下列显示格式：

- 默认 (Default)
- 十六进制 (Hex)
- 八进制 (Octal)
- 二进制 (Bin)
- 十进制 (+/-) (Dec (+/-))
- DEC

根据参数视图中选择的参数，只能选择支持的显示格式。

## 5.4 组态定位轴工艺对象

### 要求

- 工艺对象的参数视图已打开。

### 步骤

要更改值的显示格式，请按照以下步骤操作：

- 选择一个或多个要更改显示格式的表行。
- 在快捷菜单中选择“显示格式”命令。
- 选择所需的显示格式。

### 说明

要更改多个表行中某一特定数据类型的显示格式，请将参数视图按该数据类型进行排序。然后按住 **<Shift>** 键选中属于该数据类型的第一个表行和最后一个表行，最后更改选定表行的显示格式。

### 创建监视值的快照

可在 CPU 上备份工艺对象的当前值（监视值）并将其显示在参数视图中。

### 要求

- 需要有在线连接。
- 工艺对象已下载到 CPU 中。
- 程序执行处于激活状态（CPU 处于“RUN”模式）。
- 工艺对象的参数视图已打开。
- “监视所有”(Monitor all) 按钮  已选择。

### 步骤

要显示当前参数值，请按以下步骤操作：

- 在参数视图中，单击“创建监视值的快照”(Create snapshot of monitor values) 图标 。

## 结果

会向参数表的“快照”(Snapshot) 列传送一次当前监视值。

监视值会在“监视值”(Monitor values) 列中继续更新，此时可分析以此方式“冻结”的值。

## 修改值

可通过参数视图修改 CPU 中工艺对象的值。

可以向参数分配一次值（修改值）并立即对其进行修改。修改请求会尽快执行，而不参考用户程序中的任何特定点。



### 修改时存在的危险：

在发生故障或程序错误的情况下，如果在设备运行时更改参数值，则可能会导致严重财产损失和人员重伤。

在使用“修改”功能之前，请确保不会发生危险。

## 要求

- 需要有在线连接。
- 工艺对象已下载到 CPU 中。
- 程序执行处于激活状态（CPU 处于“RUN”模式）。
- 工艺对象的参数视图已打开。
- “监视所有”(Monitor all) 按钮 已选择。
- 参数需可修改（“修改值”(Modify value) 列中的相关字段背景为浅橙色）。

## 步骤

要立即修改参数，请按以下步骤操作：

1. 在参数表的“修改值”(Modify value) 列中输入所需修改值。
2. 检查“选择用于传输”(Select for transmission) 列中用于修改的复选框是否已选中。  
修改值和相关参数的相关复选框会同时自动调整。
3. 单击“请立即一次性修改全部选定参数”(Modify all selected parameters immediately and once) 图标 。

选定参数会由指定值立即一次性修改，并可以在“修改值”(Modify values) 列中进行监视。“选择用于传输”(Selection for transmission) 列中用于修改的复选框会在修改请求完成之后自动清除。

## 错误指示

当输入了起始值时，会立即检查是否有过程相关错误和语法错误，并将结果表示出来。

错误的起始值会由以下方法表示：

- “修改值”(Modify value) 字段中的红色背景  
并且
- 如果单击错误的字段，会出现弹出错误消息，其中包含有关允许的值范围或必要的语法（格式）的信息

## 错误修改值

- 具有过程相关错误的修改值可以进行传输。
- 具有语法错误的修改值无法进行传输。

## 比较值

可以使用比较功能来比较参数的以下存储值：

- 项目起始值
- PLC 起始值
- 快照

## 要求

- 需要有在线连接。
- 工艺对象已下载到 CPU 中。
- 程序执行处于激活状态（CPU 处于“RUN”模式）。
- 工艺对象的参数视图已打开。
- “监视所有”(Monitor all) 按钮  已选择。

## 步骤

要比较不同目标系统中的起始值，请按下列步骤操作：

1. 单击“选择比较值”(Selection of compare values) 图标 。

将打开一个包含比较选项的选择列表：

- 项目起始值 - PLC 起始值（默认设置）
- 项目起始值 - 快照
- PLC 起始值 - 快照

2. 选择所需的比较选项。

所选比较选项的执行方式如下：

- 在选择进行比较的两个列的标题单元格中会出现刻度符号。
- “比较结果”(Compare result) 列中使用的符号用来指示所选列的比较结果。

## “比较结果”列中的符号

符号	含义
	比较值相等且无误。
	比较值不相等但无误。
	两个比较值中至少有一个具有过程相关错误或语法错误。
	无法进行比较。两个比较值中至少有一个不可用（例如快照）。
	由于该值与组态无关，不适用于进行比较。

## 5.4 组态定位轴工艺对象

### 导航中的符号

如果所显示的导航结构下方的参数中至少有一个应用了比较结果，则导航中的符号会以相同方式显示。

### 将来自在线程序的值应用为起始值

为了将来自 CPU 的优化值应用为项目的起始值，应创建监视值的快照。标记为设定值的快照值将应用为项目的起始值。

### 要求

- 工艺对象的类型为“PID\_Compact”、“PID\_3Step”或“PID\_Temp”。
- 需要有在线连接。
- 工艺对象已下载到 CPU 中。
- 程序执行处于激活状态（CPU 处于“RUN”模式）。
- 工艺对象的参数视图已打开。
- 已选择“监视所有”(Monitor all) 按钮 。

### 操作步骤

要应用来自 CPU 的优化值，请按下列步骤操作：

1. 单击“创建监视值的快照并将该快照的设定值接受为起始值”(Create snapshot of monitor values and accept setpoints of this snapshot as start values) 图标 。

### 结果

当前监视值会应用到“快照”(Snapshot) 列，其设定值将作为新的起始值复制到“项目起始值”(Start value in project) 列。

---

### 说明

#### 应用各个参数的值

还可以将未标记为设定值的各个参数值从“快照”(Snapshot) 列应用到“项目起始值”(Start value project) 列。为此，请使用快捷菜单中的“复制”(Copy) 和“粘贴”(Paste) 命令复制这些值并将其插入“项目起始值”(Start value in project) 列中。

## 初始化在线程序中的设定值

在 CPU 中，仅通过一个步骤即可将所有在参数视图中标记为“设定值”(Setpoint) 的参数初始化为新值。为此，请将起始值从项目下载到 CPU 中。CPU 将保持为“RUN”模式。

为避免在冷启动或暖启动期间丢失 CPU 上的数据，请务必把工艺对象也下载到 CPU 中。



### 更改参数值时存在的危险

在发生故障或程序错误的情况下，如果在设备运行时更改参数值，则可能会导致严重财产损失和人员重伤。

在重新初始化设定值之前，请确保不会发生危险。

## 要求

- 工艺对象的类型为“PID\_Compact”、“PID\_3Step”或“PID\_Temp”。
- 需要有在线连接。
- 工艺对象已下载到 CPU 中。
- 程序执行处于激活状态（CPU 处于“RUN”模式）。
- 工艺对象的参数视图已打开。
- 已选择“监视所有”(Monitor all) 按钮 。
- 标记为“设定值”(Setpoint) 的参数具有无过程相关错误和语法错误的“项目起始值”(Start value in project)。

## 操作步骤

要初始化所有设定值，请按以下步骤操作：

- 在“项目起始值”(Start value in project) 列中输入所需的值。

确保起始值没有过程相关错误和语法错误。

- 单击图标  “加载设定值的起始值作为实际值”。

## 结果

CPU 中的设定值初始化为项目的起始值。

## 5.4.6 组态用于运动控制的工艺模块

### 5.4.6.1 概述

下列工艺模块可用于 S7-1200 运动控制系统中的编码器连接。

ET 200MP	ET 200 SP
TM Count 2x24V (页 159)	TM Count 1x24V (页 159)
TM PosInput 2 (页 157)	TM PosInput 1 (页 157)

在系统中，工艺模块既可以用作集中式模块，也可以用作分布式模块。

以下部分介绍了如何组态工艺模块为编码器。

### 参见

[TM Count 1x24V / TM Count 2x24V \(页 159\)](#)

[TM PosInput 1 / TM PosInput 2 \(页 157\)](#)

### 5.4.6.2 TM PosInput 1 / TM PosInput 2

要与 S7-1200 运动控制系统结合使用，必须组态以下参数：

组态	
工艺模块 TM PosInput 1 / TM PosInput 2	工艺对象 轴
“基本参数 > 通道 0/1 > 工作模式”(Basic parameters > Channel 0/1 > Operating mode)	-
选择模式“工艺对象“运动控制”的位置输入”(Position input for technology object "Motion Control")	
“基本参数 > 通道 0/1 > 模块参数”(Basic parameters > Channel 0/1 > Module parameters)	基本参数 > 编码器
-	编码器连接 选择编码器至 PROFINET/PROFIBUS
-	编码器选择 选择“编码器”(Encoder) 数据连接以及激活并组态为工艺模块上的编码器的通道
-	与编码器之间的数据交换 选择编码器后，自动选择报文“DP_TEL83_STANDARD”。 取消选择复选框“运行时自动应用编码器值（在线）” 选择复选框“组态时自动应用编码器值（离线）” 如果该复选框未选择，则可手动匹配该表中所列的参数和确定的参数。
信号类型 <ul style="list-style-type: none"><li>• 增量式编码器</li><li>• 绝对值编码器</li></ul> 编码器类型 <ul style="list-style-type: none"><li>• 线性</li><li>两个增量之间的距离</li><li>• 旋转</li><li>输入每转增量数</li></ul>	编码器类型 选择与工艺模块的组态相对应的编码器类型 选择测量系统的版本 1): <ul style="list-style-type: none"><li>• 线性版本（增量或绝对） 输入增量间距离 1)<ul style="list-style-type: none"><li>- 增量值：输入与工艺模块组态相对应的每转的步数 (1:1)<sup>1)</sup></li><li>- 绝对值：输入与工艺模块组态相对应的每转步数和转数 (1:1)<sup>1)</sup></li></ul></li><li>• 旋转版本</li></ul>

## 5.4 组态定位轴工艺对象

组态	
工艺模块	工艺对象
TM PosInput 1 / TM PosInput 2	 轴
信号评估	高精度 选择与工艺模块组态相对应的高精度 <sup>1)</sup>
<ul style="list-style-type: none"> <li>• 单段</li> <li>• 双重</li> <li>• 四重:</li> </ul>	<ul style="list-style-type: none"> <li>• 增量式编码器:           <ul style="list-style-type: none"> <li>- 0 = 单段</li> <li>- 1 = 双重或</li> <li>- 2 = 四重</li> </ul> </li> <li>• 绝对值编码器:           <ul style="list-style-type: none"> <li>- 0 (= 单重)</li> </ul> </li> </ul>
-	基本参数驱动 > 驱动
<ul style="list-style-type: none"> <li>• 旋转类型: 输入与工艺对象的组态相对应的基准速度 (1:1)</li> <li>• 线性类型: 输入基准速度</li> </ul>	输入基准速度

<sup>1)</sup> 激活“组态时自动应用编码器值（离线）”(Automatically apply encoder values during configuration (offline)) 时，自动传送参数。

“-”表示对于这些参数，无需工艺模块/工艺对象的组态

## 参见

[概述 \(页 156\)](#)

[TM Count 1x24V / TM Count 2x24V \(页 159\)](#)

### 5.4.6.3 TM Count 1x24V / TM Count 2x24V

要与 S7-1200 运动控制系统结合使用，必须组态以下参数：

组态	
工艺模块	工艺对象
TM Count 1x24V / TM Count 2x24V	轴
“基本参数 > 通道 0/1 > 工作模式”(Basic parameters > Channel 0/1 > Operating mode)	-
选择模式“工艺对象“运动控制”的位置输入”(Position input for technology object "Motion Control")	
“基本参数 > 通道 0/1 > 模块参数”(Basic parameters > Channel 0/1 > Module parameters)	基本参数 > 编码器
-	编码器连接 选择编码器至 PROFINET/PROFIBUS
-	编码器选择 选择“编码器”(Encoder) 数据连接以及激活并组态为工艺模块上的编码器的通道
-	与编码器之间的数据交换 选择编码器后，自动选择报文“DP_TEL83_STANDARD”。 取消选择复选框“运行时自动应用编码器值（在线）” 选择复选框“组态时自动应用编码器值（离线）”(Automatically apply encoder values during configuration (offline))。 如果已清除复选框，您可以手动匹配此表中描述和确定的参数。

## 5.4 组态定位轴工艺对象

组态	
工艺模块	工艺对象
TM Count 1x24V / TM Count 2x24V	 轴
<b>信号类型</b> <ul style="list-style-type: none"> <li>增量式编码器</li> </ul> <b>编码器类型</b> <ul style="list-style-type: none"> <li>线性           <ul style="list-style-type: none"> <li>两个增量之间的距离</li> </ul> </li> <li>旋转           <ul style="list-style-type: none"> <li>输入每转增量数</li> </ul> </li> </ul>	<b>编码器类型</b> 选择与工艺模块组态相对应的编码器类型 选择测量系统类型 <sup>1)</sup> : <ul style="list-style-type: none"> <li>线性版本（增量或绝对）               <ul style="list-style-type: none"> <li>输入增量间距离<sup>1)</sup></li> </ul> </li> <li>旋转版本               <ul style="list-style-type: none"> <li>增量值: 输入与工艺模块组态相对应的每转的步数(1:1)<sup>1)</sup></li> <li>绝对值: 输入与工艺模块组态相对应的每转步数和转数(1:1)<sup>1)</sup></li> </ul> </li> </ul>
<b>信号评估</b> <ul style="list-style-type: none"> <li>单段</li> <li>双重</li> <li>四重:</li> </ul>	<b>高精度</b> 选择与工艺模块组态相对应的高精度 <sup>1)</sup> <ul style="list-style-type: none"> <li>0 = 单段</li> <li>1 = 双倍频</li> <li>2 = 四重</li> </ul>
-	<b>基本参数驱动 &gt; 驱动</b>
<ul style="list-style-type: none"> <li>旋转类型:           <ul style="list-style-type: none"> <li>输入与工艺对象的组态相对应的基准速度(1:1)</li> </ul> </li> <li>线性类型:           <ul style="list-style-type: none"> <li>输入基准速度</li> </ul> </li> </ul>	输入基准速度

<sup>1)</sup> 激活“组态时自动应用编码器值（离线）”(Automatically apply encoder values during configuration (offline))时，自动传送参数。

“-”表示对于这些参数，无需工艺模块/工艺对象的组态

## 参见

概述 (页 156)

TM PosInput 1 / TM PosInput 2 (页 157)

# 工艺对象命令表

## 6.1 使用作业表工艺对象

使用工艺对象“命令表”(Command table)，可将多个单独的轴控制命令合并到一个运动序列中。对于通过 PTO (Pulse Train Output) 连接驱动器的轴，此工艺对象可以用于工艺版本 V2 及更高版本。

用于在组态对话框中以表格形式组态移动序列。

将项目装载到 CPU 之前，可以在曲线图上检查移动序列的运动轨迹。然后，所创建的命令表将链接到某个轴并在用户程序中通过“MC\_CommandTable”运动控制指令进行使用。可以处理部分或全部命令表。

## 6.2 命令表工艺对象工具

在 TIA Portal 中，为“命令表”工艺对象提供了“组态”工具。下图显示了该工具与工艺对象的相互关系：



① 读取和写入工艺对象的组态

## 6.3 添加工艺对象命令表

### 组态

使用“组态”工具可以组态“命令表”工艺对象的以下属性：

- 可以通过组态单独的作业来创建一个或多个移动序列。
- 可以使用已组态的轴或可组态的默认轴来组态图形显示，以检查移动序列。

移动序列数据保存在工艺对象的数据块中。

## 6.3 添加工艺对象命令表

### 要求

- 已创建具有 CPU S7-1200 的项目。
- CPU 固件版本为 V2.1 或更高版本

### 步骤

要在项目树中添加“命令表”工艺对象，请按以下步骤操作：

1. 在项目树中打开“CPU > 工艺对象”(CPU > Technology objects) 文件夹。
2. 双击“添加新对象”(Add new object) 命令。  
将打开“添加新对象”(Add new object) 对话框。
3. 选择“运动控制”(Motion Control) 工艺。
4. 打开“运动控制”(Motion Control) 文件夹。
5. 在“版本”(Version) 列中选择所需的工艺版本。
6. 选择“TO\_CommandTable”对象。
7. 在“名称”(Name) 输入框中输入命令表名称。
8. 要更改自动分配的数据块编号，请选择“手动”(Manual) 选项。
9. 要显示有关工艺对象的其它信息，请单击“其它信息”(Additional information)。
10. 单击“确定”(OK) 确认输入。

### 结果

创建了新工艺对象，并保存在项目树中的“工艺对象”(Technology objects) 文件夹中。

## 6.4 组态命令表工艺对象

### 6.4.1 使用组态对话框

在组态窗口中，组态工艺对象的属性。要打开工艺对象的组态窗口，请按以下步骤操作：

1. 在项目树中打开所需工艺对象组。
2. 双击“组态”(Configuration) 对象。

组态分为以下几类：

- **基本参数**

基本参数包括必须为功能命令表组态的所有参数。

- **扩展参数**

扩展参数包含默认轴的参数或显示所选轴的参数值。

### 组态窗口图标

组态的区域导航中的图标显示有关组态情况的详细信息：

	<b>组态包含默认值且已完成。</b> 组态仅包含默认值。使用这些默认值即可使用工艺对象，而无需进行更改。
	<b>组态中包含用户自定义或系统自动调整的值且已完成。</b> 组态的所有输入域均包含有效值，且至少一个预设值发生更改。
	<b>组态未完成或不正确。</b> 至少一个输入域或下拉列表包含无效值。相应域或下拉列表以红色背景显示。单击弹出错误消息可显示错误原因。
	<b>组态包含彼此不兼容的参数值。</b> 组态包含在大小或逻辑上互相矛盾的参数值。相应域或下拉列表以黄色背景显示。

### 参见

[运动控制使用指南 \(页 50\)](#)

[基本参数 \(页 165\)](#)

[扩展参数 \(页 182\)](#)

## 6.4.2 监视值

如果已在线连接 CPU，则“全部监视”(Monitor all) 图标  将显示在工艺对象的组态对话框中。

“监视全部”(Monitor all) 功能中包含以下选项：

- 将项目中组态的起始值与 CPU 中的起始值和实际值进行比较
- 直接编辑实际值和项目的起始值
- 立即检测并显示输入错误和建议的更正措施
- 备份项目中的实际值，并手动传送到项目的起始值中

### 图标和操作员控件

如果存在到 CPU 的在线连接，则参数中将显示实际值。

除参数的实际值外，还会显示下列符号：

图标	说明
	CPU 中的起始值与已组态的项目中起始值相匹配
	CPU 中的起始值与已组态的项目中的起始值不匹配
	由于所选 CPU 模块不支持这种比较方式，因此 CPU 中的起始值无法与所组态项目中的起始值进行比较。
	由于该值与组态无关，因此无法比较。
	使用按钮显示 CPU 的起始值和各个参数的项目起始值。

可以直接更改项目中的实际值和起始值，然后下载到 CPU。可直接修改的参数的实际值更改将直接传送到 CPU。

## 6.4.3 基本参数

### 6.4.3.1 组态 - 常规

在“常规”(General) 组态窗口中组态工艺对象的名称。

#### 名称

在该域中定义命令表的名称或“命令表”工艺对象的名称。该工艺对象以该名称列出在项目树中。

#### 参见

命令表组态 (页 165)

快捷菜单命令 - 命令表 (页 171)

使用趋势图 (页 173)

快捷菜单命令 - 曲线图 (页 177)

从“完成命令”过渡到“混合运动” (页 178)

在用户程序中更改命令表组态 (页 181)

### 6.4.3.2 命令表组态

在“命令表”(Command Table) 组态窗口中创建所需移动序列，并根据趋势图中的图形视图检查结果。

---

#### 说明

可能在显示的时间行为和趋势位置与轴的实际运动之间存在较小偏差。未显示与达到软限位开关对应的移动。

---

#### 启用警告

使用该复选框，可在命令表中激活警报显示。

## 使用轴参数

从下拉列表中，选择使用哪些轴参数选择图形视图和检查移动顺序。如果尚未将轴添加到“工艺对象”(Technology objects)文件夹或想要使用未在任何可用轴中组态的值，应选择“默认轴”(Default axis)。在“高级参数”(Advanced parameters)下组态默认轴的属性。

“Axis”参数中选择的轴参数，将用于处理用户程序中的命令表。

## 列：步骤

显示命令步数。

## 列：命令类型

在该列中选择处理命令表所使用的命令类型。最多可输入 32 个命令。将会按顺序处理命令。可以在下列条目和命令类型之间选择：

- **Empty**

该条目用作所有要添加的命令的占位符。处理命令表时会忽略空条目。

- **Halt**

暂停轴

(只有在执行“Velocity setpoint”命令之后该命令才生效)

- **Positioning Relative**

相对定位轴

- **Positioning Absolute**

必须使轴回原点以便进行绝对定位。

- **Velocity setpoint**

以设定速度移动轴

- **Wait**

等到给定期结束。Wait 不会停止当前行进。

- **Separator**

将 Separator 行添加到所选行的上方。Separator 行用作趋势图图形显示的范围限制。

如想处理部分命令表，请使用 Separator 行。

## 列：位置/行进路径

在此列中可为所选命令输入位置或行进路径：

- 命令“Positioning Relative”

该命令将按给定的行进路径移动轴。

- 命令“Positioning Absolute”

该命令将按给定的位置移动轴。

必须使轴回原点。

- Separator

给定值指定了图形显示的起始位置。

限值（与所选的用户单位无关）：

- $-1.0e12 \leq \text{位置/距离} \leq -1.0e-12$
- $1.0e-12 \leq \text{位置/距离} \leq 1.0e12$
- 位置/行进路径 = 0.0

列：速度

在该列中，可以为所选命令输入速度：

- 命令“Positioning Relative”

该命令将以给定的速度移动轴。

如果选定行进路径不够大，则不会达到给定速度。

- 命令“Positioning Absolute”

该命令将以给定的速度移动轴。

如果目标位置距起始位置太近，则不会达到给定速度。

- 命令“Velocity setpoint”

该命令将以给定的速度移动轴。

如果选择的运行时间过短，则执行命令期间不会达到给定速度。

限值（与所选的用户单位无关）：

- 对于命令：“Positioning Relative”和“Positioning Absolute”
  - $1.0e-12 \leq$  速度  $\leq 1.0e12$
- 对于命令：“Velocity setpoint”
  - $-1.0e12 \leq$  速度  $\leq -1.0e-12$
  - $1.0e-12 \leq$  速度  $\leq 1.0e12$
  - 速度 = 0.0

#### 列：持续时间

在该列中可为所选命令输入持续时间：

- 命令“Velocity setpoint”

该命令将在指定的持续时间内移动轴。持续时间包括加速阶段和匀速行进阶段。持续时间结束后将会处理下一个命令。

- 命令“Wait”

等待，直到给定持续时间结束。

限值（与所选的用户单位无关）：

- $0.001s \leq$  持续时间  $\leq 64800s$

## 列：下一步

从下拉列表中选择过渡到下一步的模式：

- 完成命令

命令将被完成。会立即处理下一命令。

- 混合运动

当前命令的运动将会与下面命令的运动混合。利用“Positioning Relative”和“Positioning Absolute”命令类型可以进入“混合运动”转换模式。

运动将会与下列命令类型的运动混合：

- Positioning Relative
- Positioning Absolute
- Velocity setpoint

利用其他命令类型不会发生混合。

有关附加或重叠命令时轴响应的具体行为，请参见“从“完成命令”过渡到“混合运动”(页 178)”部分

## 列：步代码

在此列中可输入数值/位模式，处理命令时会在“MC\_CommandTable”运动控制指令的“StepCode”输出参数中输出该数值/位模式。

限值：

- $0 \leq$  代码编号  $\leq 65535$

## 参见

组态 - 常规 (页 165)

快捷菜单命令 - 命令表 (页 171)

使用趋势图 (页 173)

快捷菜单命令 - 曲线图 (页 177)

从“完成命令”过渡到“混合运动” (页 178)

在用户程序中更改命令表组态 (页 181)

### 6.4.3.3 快捷菜单命令 - 命令表

命令表中有以下可用快捷菜单命令：

#### 插入空行 (Insert empty line)

将空行添加到所选行的上方。

只有命令表的结尾有足够的空行时才会执行快捷菜单命令。

#### 添加空行 (Add empty line)

将空行添加到所选行的下方。

只有命令表的结尾有足够的空行时才会执行快捷菜单命令。

#### 插入分隔符行 (Insert separator line)

将分隔符行添加到所选行的上方。

无法连续添加两个分隔符行。

#### 添加分隔符行 (Add separator line)

将分隔符行添加到所选行的下方。

不可能有两条连续的分隔线；也无法在命令表的末尾添加分隔线。

#### 剪切 (Cut)

移除所选行或所选单元格的内容并将其保存到剪切板。

所选行将会删除并且命令表的后续行将会上移。

#### 复制 (Copy)

复制所选行或所选单元格的内容并将其保存到剪切板。

### 粘贴 (Paste)

- 所选行:

将剪切板中的行粘贴到所选行上方的表中。

- 所选单元格:

将剪切板的内容粘贴到所选行。

只有命令表的结尾有足够的空行时才会执行快捷菜单命令。

### 替换 (Replace)

用剪切板中的行替换所选行。

### 删除 (Delete)

删除所选行。命令表中下方的行将上移。

## 参见

[组态 - 常规 \(页 165\)](#)

[命令表组态 \(页 165\)](#)

[使用趋势图 \(页 173\)](#)

[快捷菜单命令 - 曲线图 \(页 177\)](#)

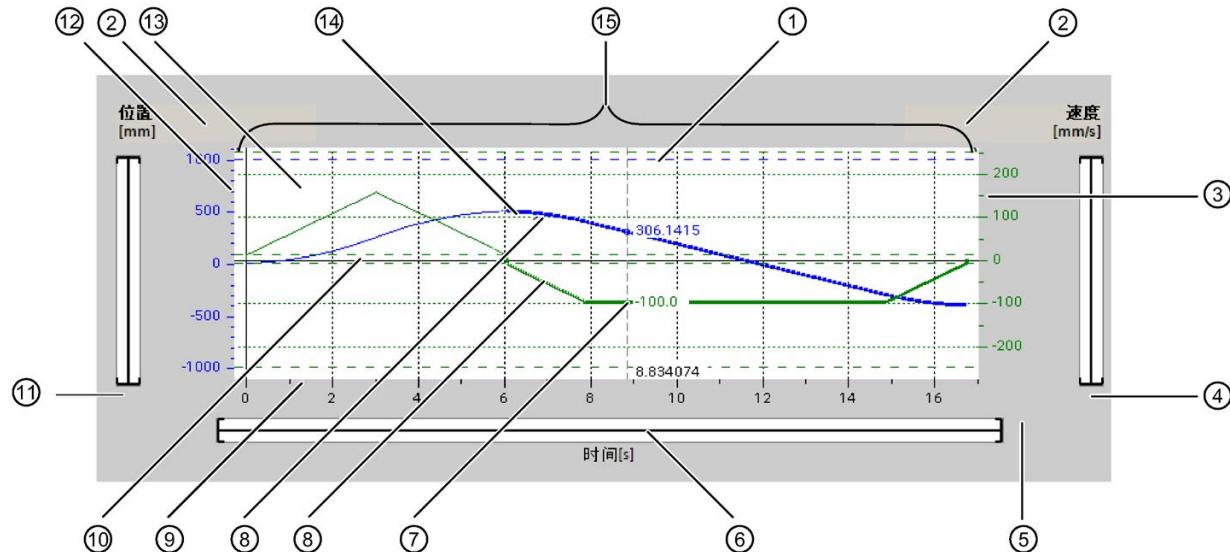
[从“完成命令”过渡到“混合运动” \(页 178\)](#)

[在用户程序中更改命令表组态 \(页 181\)](#)

#### 6.4.3.4 使用趋势图

趋势视图中提供下列工具和信息：

##### 趋势图和组件



(1)	比例尺
(2)	选择网格
(3)	速度轴刻度范围
(4)	滚动条, 速度轴
(5)	滚动条时间轴
(6)	比例尺位置标记
(7)	速度曲线
(8)	所选命令的曲线部分
(9)	时间轴刻度范围
(10)	启动/停止速度
(11)	滚动条, 位置轴
(12)	位置轴刻度范围
(13)	软限位开关位置
(14)	位置曲线
(15)	趋势视图

## 选择分隔符部分

如果命令表包含多个由分隔符分隔的部分，可以在趋势图中选择这些部分，方法是在相应部分中选择一个命令。

## 选择命令

可以在趋势图和命令表中选择命令：

- 单击速度上的某个点或趋势图中的位置曲线。相应的命令会在命令表中突出显示。
- 在命令表中选择一个命令。

曲线的相应部分将会突出显示。

## 选择趋势图的可见范围

按照下面的步骤可以调整要显示的趋势图部分：

在快捷菜单中选择缩放：

- 缩放到曲线：

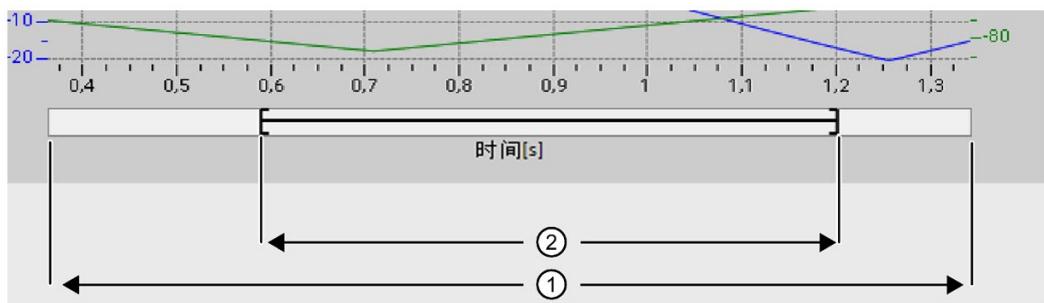
缩放轴，以便位置和速度曲线可见。

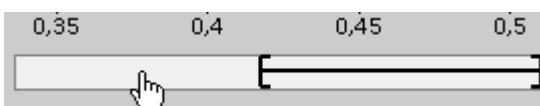
- 缩放到曲线和限值：

缩放轴，以便位置和速度曲线、激活软限位开关的位置和最小最大速度限值可见。

所选图在快捷菜单中将会用勾号标记。

选择要在范围内显示的部分：

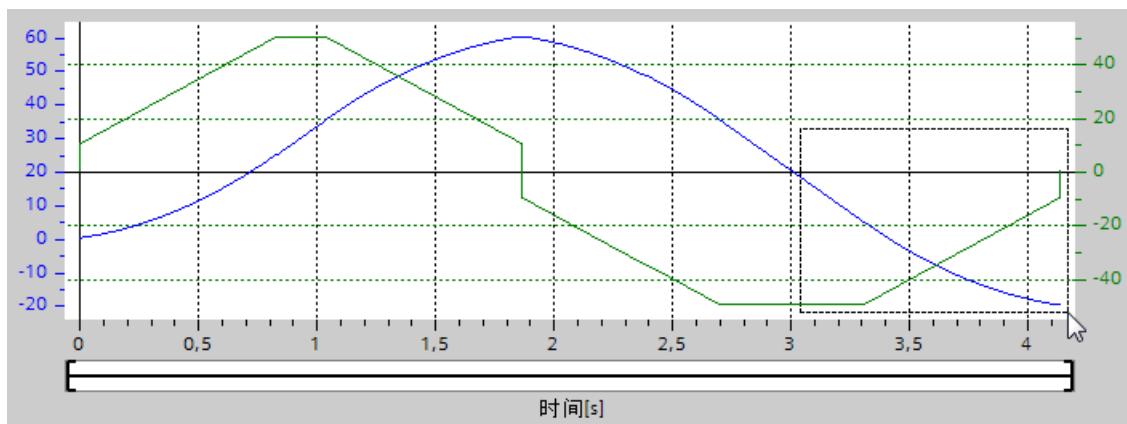


①	曲线值和/或限值所在的范围。（请参见快捷菜单中的选择）
②	<p>要在趋势窗口中显示的所选范围。 利用空白光标可以设置右边和左边的空白范围。</p>  <p>通过拖动光标在范围 ① 内设置位置。</p>  <p>还可以通过在范围 ① 内单击来定义位置。</p> 

利用鼠标选择要显示的部分：

通过单击和拖动鼠标拖选出曲线图的某一部分。所选曲线的该部分将会在释放鼠标后立即放大。

## 6.4 组态命令表工艺对象



撤消对该部分的最后一次更改:

选择快捷命令“撤消缩放”(Undo zoom)，撤消对该部分的最后一次更改。

## 同步网格

单击轴刻度，选择是否将网格与位置轴或速度轴同步。

## 从比例尺读取曲线值

使用快捷菜单命令“显示比例尺”(Show ruler) 激活比例尺。

使用比例尺光标可以将比例尺移动至曲线上的任何点。



## 参见

- 组态 - 常规 (页 165)
- 命令表组态 (页 165)
- 快捷菜单命令 - 命令表 (页 171)
- 快捷菜单命令 - 曲线图 (页 177)
- 从“完成命令”过渡到“混合运动” (页 178)
- 在用户程序中更改命令表组态 (页 181)

### 6.4.3.5 快捷菜单命令 - 曲线图

曲线窗口中有以下可用快捷菜单命令：

#### 缩放 100% (Zoom 100%)

选择一个可以显示 100% 曲线值和/或限值的缩放系数。

#### 撤消缩放 (Undo zoom)

撤消最后一次缩放更改。

#### 趋势缩放

缩放轴，以便位置和速度趋势可见。

#### 趋势和限值标度

缩放轴，以便位置和速度趋势、激活软限位开关的位置和最小最大速度限值可见。

#### 显示速度限值

显示速度限值行。

#### 显示软限位开关

显示软限位开关的行。

## 显示标尺

使标尺渐显/渐隐

在要查看趋势的各个值时可使用标尺。

## 参见

组态 - 常规 (页 165)

命令表组态 (页 165)

快捷菜单命令 - 命令表 (页 171)

使用趋势图 (页 173)

从“完成命令”过渡到“混合运动” (页 178)

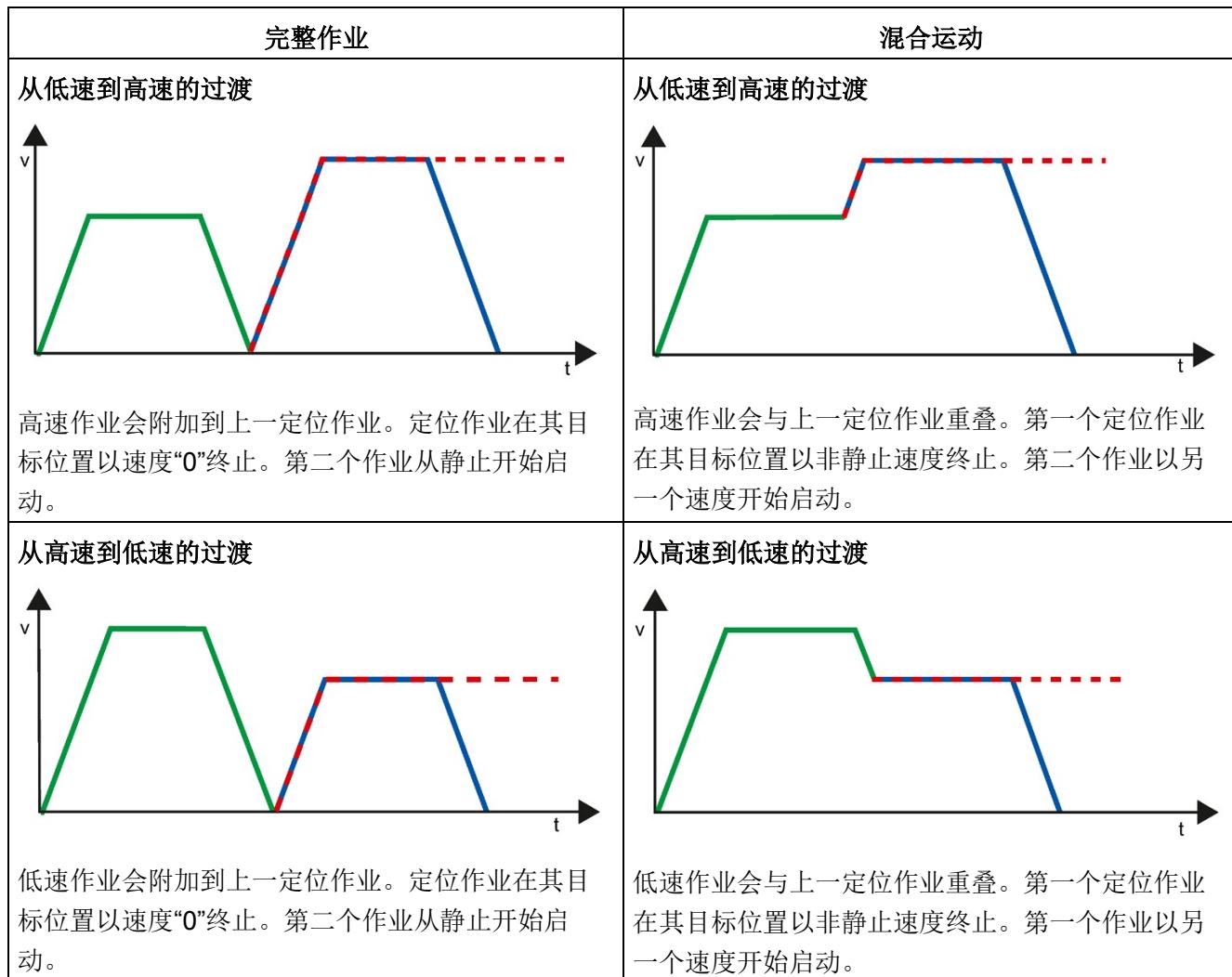
在用户程序中更改命令表组态 (页 181)

### 6.4.3.6 从“完成命令”过渡到“混合运动”

下图在“下一步”列中显示了不同转换模式移动之间的转换：

### 具有先前定位作业的运动过渡

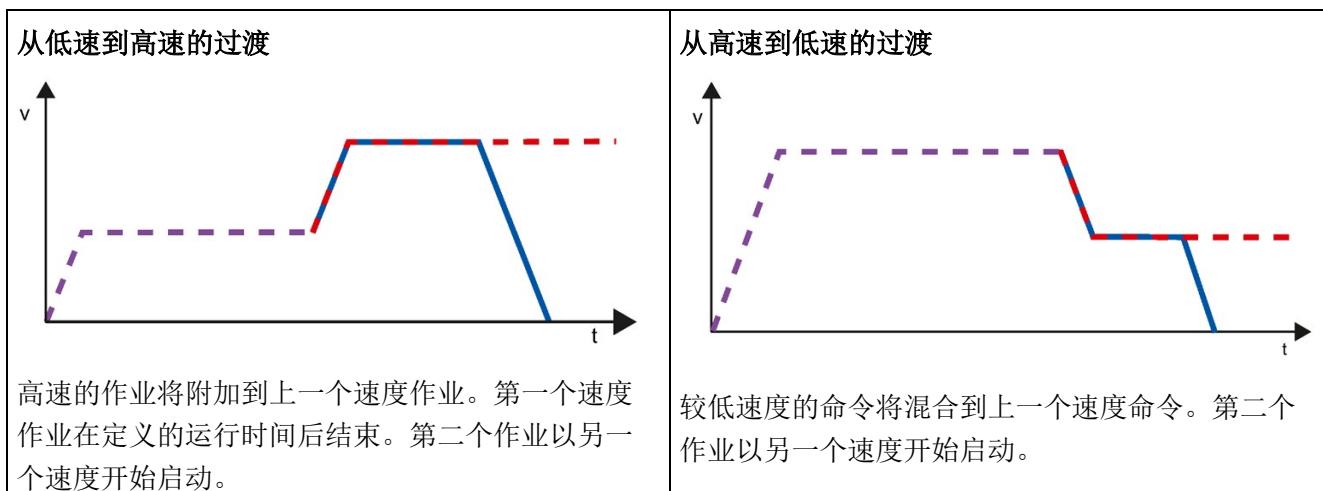
下图显示了两个运动任务的命令顺序。第一个命令用于定位（绿色）。第二个命令用于速度（红色）或定位（蓝色）：



	第一个作业“Positioning Relative”或“Positioning Absolute”
	第二个作业“Velocity setpoint”
	第二个作业“Positioning Relative”或“Positioning Absolute”

## 具有先前速度作业的运动过渡

下图显示了两个运动任务的命令顺序。第一个命令用于速度（紫色）。第二个命令用于速度（红色）或定位（蓝色）：



..	第一个作业“Velocity setpoint”
..	第二个作业“Velocity setpoint”
-	第二个作业“Positioning Relative”或“Positioning Absolute”

## 参见

组态 - 常规 (页 165)

命令表组态 (页 165)

快捷菜单命令 - 命令表 (页 171)

使用趋势图 (页 173)

快捷菜单命令 - 曲线图 (页 177)

在用户程序中更改命令表组态 (页 181)

### 6.4.3.7 在用户程序中更改命令表组态

可以在 CPU 中运行用户程序期间更改下列组态参数：

#### 命令和对应值

还可以在用户程序运行期间更改命令表的参数。要执行该操作，可使用以下工艺对象变量：

- <表名称>.Command[1..32].Type  
用于更改命令类型
- <表名称>.Command[1..32].Position  
用于更改位置/行进距离
- <表名称>.Command[1..32].Velocity  
用于更改速度
- <表名称>.Command[1..32].Duration  
用于更改持续时间
- <表名称>.Command[1..32].NextStep  
用于更改参数“下一步”
- <表名称>.Command[1..32].StepCode  
用于更改步代码

有关组态参数的更改何时生效的信息，请参见附录中的工艺对象变量 (页 517)说明。

#### 参见

变量兼容表 V1...3 <-> V4...5 (页 58)

组态 - 常规 (页 165)

命令表组态 (页 165)

快捷菜单命令 - 命令表 (页 171)

使用趋势图 (页 173)

快捷菜单命令 - 曲线图 (页 177)

从“完成命令”过渡到“混合运动” (页 178)

## 6.4.4 扩展参数

### 6.4.4.1 组态 - 扩展参数

在“扩展参数”(Extended parameters) 组态窗口中组态“命令表”工艺对象图表视图的基本属性。

#### 使用轴参数

从下拉列表中，选择使用哪些轴参数选择图形视图和检查移动顺序。如果尚未将轴添加到“工艺对象”(Technology objects) 文件夹中或需使用在任何可用轴中均未组态的值，则需在下拉列表中选择“默认轴”(Default axis)。

“轴”(Axis) 参数中选择的轴的轴参数将用于处理用户程序中的命令表。

#### 位置计量单位

如果在“使用轴参数”(Use axis parameter from) 下拉列表中选择一个默认轴，则可设置测量单位。

如果在下拉列表中选择一个已组态的轴，则将显示所组态轴的测量单位。

#### 复制轴参数

选择复制方向及要复制轴参数的轴。可将默认轴的轴参数复制到所选轴，或将所选轴的轴参数作为默认轴的轴参数。使用“应用组态”(Apply configuration) 按钮，可根据组态复制轴参数。

#### 6.4.4.2 组态 - 动态

在“动态”(Dynamics) 组态窗口中可以为默认轴组态加速度和减速度以及加加速度限制。

如果在下拉列表的“组态 > 扩展参数 > 扩展参数”(Configuration > Extended parameters > Extended parameters) 中选择一个已组态的轴，则将显示该已组态轴的值。

如果在下拉列表“使用轴参数”(Use axis parameters from) 的“组态 > 扩展参数 > 扩展参数”(Configuration > Extended parameters > Extended parameters) 中选择条目“默认轴”(Default axis)，则可对以下字段进行编辑。

##### 加速度/减速度

在“加速度”(Acceleration) 字段中设置默认轴的所需加速度。在“减速度”(Deceleration) 字段中可以设置所需减速度。

将会用所选加速度/减速度计算命令表中组态的运动作业。

限值：

- $1.0E-12 \leq$  加速度  $\leq 1.0E12$
- $1.0E-12 \leq$  减速度  $\leq 1.0E12$

##### 启用加加速度限值

使用该复选框，可启用加加速度限值。

##### 加加速度

在“加加速度”(Jerk) 字段中，可设置加速和减速的所需加加速度。

将会用所选加加速度计算命令表中组态的运动作业。

限值：

- $1.0E-12 \leq$  加加速度  $\leq 1.0E12$

#### 6.4.4.3 组态 - 限值

在“限值”(Limits) 组态窗口中组态最大速度、启动/停止速度和默认轴的软限位开关。

如果在下拉列表的“组态 > 扩展参数 > 扩展参数”(Configuration > Extended parameters > Extended parameters) 中选择一个已组态的轴，则将显示该已组态轴的值。

如果在下拉列表“使用轴参数”(Use axis parameters from) 的“组态 > 扩展参数 > 扩展参数”(Configuration > Extended parameters > Extended parameters) 中选择条目“默认轴”(Default axis)，则可对以下字段进行编辑。

#### 最大速度/启动/停止速度

在这些框中定义默认轴的最大允许速度和启动/停止速度。启动/停止速度是默认轴的最小允许速度。

速度	限值
启动/停止速度	0.0
	1.0E-12 到 1.0E12
最大速度	0.0
	1.0E-12 到 1.0E12

最大速度值必须大于等于启动/停止速度值。

#### 启用软限位开关

使用此复选框可激活下限和上限软限位开关的功能。未在趋势图中显示与达到软限位开关对应的移动。

## 软限位开关上限 / 下限

在这些框中可输入下限和上限软限位开关的位置值。

软限位开关	限值
软限位开关下限	-1.0E12 到 -1.0E-12
	0.0
	1.0E-12 到 1.0E12
软限位开关上限	-1.0E12 到 -1.0E-12
	0.0
	1.0E-12 到 1.0E12

软限位开关的上限值必须大于或等于软限位开关的下限值。

# 下载到 CPU

运动控制工艺对象的数据保存在数据块中。因此，加载新的或已修改的工艺对象时，下载“块”的条件适用。



小心

## 不使用硬件配置加载时轴的可能故障

对轴或编码器组态进行以下修改时，会修改硬件配置：

- 修改脉冲发生器 (PTO)
- 修改硬限位开关地址
- 修改回原点开关的地址
- 修改 PROFIdrive 报文的地址
- 修改模拟量输出的地址
- 修改使能输出或准备就绪输入的地址

如果使用右键快捷菜单命令“软件”(Software) 或“软件（所有块）”(Software (all blocks)) 加载轴的已修改组态而未下载硬件配置，则可能会导致轴发生故障。

确保在下列条件下将当前硬件配置下载到 CPU 中。

## 在 CPU S7-1200 RUN 模式下下载（固件版本 V2.2 及更高版本）

对于固件版本 V2.2 及更高版本的 CPU S7-1200，在 CPU RUN 模式下加载时，会检查是否可在不停止 CPU 的情况下进行加载。

在 RUN 模式下加载数据块时，以下条件适用：

	下载到装载存储器	下载到工作存储器
数据块修改的值	√	-
数据块修改的结构	√ (自固件 V4 起)	√ (自固件 V4 起) • 下载需重新初始化时 • 下载无需重新初始化的系统备用变量
	- (固件 V2.2...3)	- (固件 V2.2...3)
新数据块	√	√
删除的数据块	√	√

删除数据块以及下载需要重新初始化的数据块时还需要注意以下信息：

- 下载定位轴工艺对象时必须将轴禁用。
- 下载命令表工艺对象时，不得激活该命令表的 MC\_CommandTable 命令（参数“Busy”= FALSE）。
- 下载 MC\_Power 背景数据块时，不得激活 MC\_Power 指令（参数“Busy”= FALSE）。

从工艺版本 V3.0 起，运动控制工艺对象（数据块）也可以在 CPU RUN 模式下下载。

V3.0 之前的工艺对象无法通过 CPU RUN 模式下载。

选择下述操作之一，将运动控制工艺对象（版本 V3.0 或更高版本）的已修改版本下载到工作存储器：

- **工艺对象定位轴和命令表**  
将 CPU 操作模式从 STOP 切换到 RUN。
- **工艺对象定位轴**  
将禁用轴并使用运动控制指令“MC\_Reset”执行“Restart”。
- **工艺对象命令表**  
可确保命令表未被使用。使用扩展指令“READ\_DBL”将命令表的数据块下载到工作存储器。

## 参见

运动控制使用指南 (页 50)

## 8.1 轴控制面板

轴控制面板用于在手动模式下移动轴、优化轴设置和测试系统。

只有与 CPU 建立在线连接后，才能使用轴控制面板。为确保响应时间尽可能短而使用轴控制面板和优化时，建议显示其它在线通信。

轴控制面板中包含以下几个区域：

- 主控制
- 轴
- 命令
- 当前值
- 轴状态

---

### 说明

#### 轴控制面板的响应时间

轴控制面板操作期间的响应时间取决于 CPU 的通信负载。关闭 TIA Portal 的所有其它在线窗口可将响应时间降至最低。

可以启动对话框中调整超时时间。

---

## 主控制

在此区域中，用户可获取工艺对象的主控制权限，或将其返回给用户程序：

- “激活”(Activate) 按钮

通过“激活”(Activate) 按钮，将与 CPU 建立在线连接并获取对所选工艺对象的主控制权限。获取主控制权限时，请注意以下事项：

- 要获取主控制权限，必须在用户程序中禁用工艺对象。
- 在返回主控制权限之前，用户程序对该工艺对象的功能无任何影响。系统拒绝将运动控制作业从用户程序传送到工艺对象中，并报告错误。



### 自动模式下的其它轴

主控制权限仅适用于所选择工艺对象。如果自动模式下存在其它轴，则可能引发危险。

如果发生这种情况，请禁用所有其它轴。

- “禁用”(Deactivate) 按钮

通过“禁用”(Deactivate) 按钮，可将主控制权限返回给用户程序。

## 轴

在此区域中，可启用或禁用工艺对象使用轴控制面板/进行优化：

- “启用”(Enable) 按钮

使用“启用”(Enable) 按钮，可启用所选择的工艺对象。

- “禁用”(Disable) 按钮

使用“禁用”(Disable) 按钮，可禁用所选的工艺对象。

## 8.1 轴控制面板

### 命令

仅当轴启用后，才能执行“命令”(Command) 区域中的操作。可以选择以下命令输入之一：

- 点动

该命令相当于用户程序中的运动控制命令“MC\_MoveJog”。

- 定位

该命令相当于用户程序中的运动控制命令“MC\_MoveAbsolute”和“MC\_MoveRelative”。必须使轴回原点以便进行绝对定位。

- 回原点

该命令相当于用户程序中的运动控制命令“MC\_Home”。

- “设置参考点”(Set reference point) 按钮相当于 Mode = 0 (绝对式直接回原点)
- “主动回原点”(Active homing) 按钮相当于 Mode = 3 (主动回原点)

对于主动回原点，必须在轴组态中组态回原点开关。

逼近速度、回原点速度和参考位置偏移的值取自尚未更改的轴组态。

根据选择，将显示相关的设定值输入框和命令启动按钮。

选择“启用加加速度限值”(Enable jerk limitation) 复选框，将激活加加速度限值。默认情况下，加加速度为组态值的 10%。可根据需要更改该值。

### 当前值

在该区域中，将显示轴的以下实际值：

- 位置
- 速度

## 轴状态

“轴状态”(Axis status) 区域中，将显示当前轴状态和驱动装置的状态。

状态消息	说明
已启用	轴已启用且准备就绪，可通过运动控制命令进行控制。
已回原点	轴已回原点，可执行运动控制指令“MC_MoveAbsolute”的绝对定位命令。
就绪	驱动装置已就绪，可以运行。
轴错误	定位轴工艺对象出错。在“错误消息”(Error message) 框中，将显示有关该错误原因的详细信息。
编码器值有效	编码器值有效。
仿真激活	在 CPU 中对轴进行仿真。设定值未输出到驱动装置中。
驱动装置错误	驱动装置因“驱动装置就绪”信号丢失而报错。
需要重新启动	已在 CPU RUN 模式下将已修改的轴组态下载到装载存储器。要将已修改组态下载到工作存储器，需要重新启动轴。为此，可使用运动控制指令“MC_Reset”。

“信息消息”(Info message) 框会显示有关轴状态的高级信息。

“错误消息”(Error message) 框会显示当前错误。

单击“确认”(Acknowledge)，确认所有已清除的错误。

---

## 说明

### 速度、加速度/减速度和加加速度的初始值

出于安全考虑，激活轴控制面板时，仅使用 10% 的组态值对“速度”(Velocity)、“加速度/减速度”(Acceleration/deceleration) 和“加加速度”(Jerk) 参数进行初始化。“加加速度”(Jerk) 参数仅用于 V2.0 及更高版本的“轴”工艺对象。

在组态视图中选择“扩展参数 > 动态 > 常规”(Extended parameters > Dynamics > General) 后显示的值可用于初始化。

轴控制面板中的“速度”(Velocity) 参数基于组态中的“最大速度”(Maximum velocity)，“加速度/减速度”(Acceleration/deceleration) 参数则基于“加速度”(Acceleration)。

在轴控制面板中，“速度”(Velocity)、“加速度/减速度”(Acceleration/deceleration) 和“加加速度”(Jerk) 参数可以更改，而不会影响组态中的值。

---

## 参见

运动控制使用指南 (页 50)

## 8.2 调节

通过 PROFIdrive/模拟量输出连接驱动器的轴的移动是受位置控制的。

通过“自整定”(Tuning) 功能，可确定用于轴控制回路 (页 130) 的最佳增益 (Kv 因子)。为此，可通过“跟踪”(Trace) 功能记录所组态的定位移动过程中轴速度配置文件。然后，可对记录进行评估并对增益进行相应调整。为确保响应时间尽可能短而使用轴控制面板和优化时，建议显示其它在线通信。

定位轴工艺对象的“自整定”(Tuning) 功能位于项目树的“工艺对象 > 调试”(Technology object > Commissioning) 中。

“自整定”(Tuning) 对话框中包含有以下区域：

- 主控制
- 轴
- 轴状态
- 优化增益设置
- 运行测量
- 跟踪

---

### 说明

#### 不传输参数

返回主控制后放弃所组态的参数值。并根据需要将值传输到组态中。

#### 启动优化

启动优化的同时启动跟踪。为此可在启动对话框中调整超时时间。

---

## 主控制

在此区域中，用户可获取工艺对象的主控制权限，或将其返回给用户程序：

- “激活”(Activate) 按钮

通过“激活”(Activate) 按钮，将与 CPU 建立在线连接并获取对所选工艺对象的主控制权限。获取主控制权限时，请注意以下事项：

- 要获取主控制权限，必须在用户程序中禁用工艺对象。
- 在返回主控制权限之前，用户程序对该工艺对象的功能无任何影响。系统拒绝将运动控制作业从用户程序传送到工艺对象中，并报告错误。



### 自动模式下的其它轴

主控制权限仅适用于所选择工艺对象。如果自动模式下存在其它轴，则可能引发危险。

如果发生这种情况，请禁用所有其它轴。

- “禁用”(Deactivate) 按钮

通过“禁用”(Deactivate) 按钮，可将主控制权限返回给用户程序。

## 轴

在此区域中，可启用或禁用工艺对象使用轴控制面板/进行优化：

- “启用”(Enable) 按钮

使用“启用”(Enable) 按钮，可启用所选择的工艺对象。

- “禁用”(Disable) 按钮

使用“禁用”(Disable) 按钮，可禁用所选的工艺对象。

## 8.2 调节

### 轴状态

“轴状态”(Axis status) 区域中，将显示当前轴状态和驱动装置的状态。

状态消息	说明
已启用	轴已启用且准备就绪，可通过运动控制命令进行控制。
已回原点	轴已回原点，可执行运动控制指令“MC_MoveAbsolute”的绝对定位命令。
就绪	驱动装置已就绪，可以运行。
轴错误	定位轴工艺对象出错。在“错误消息”(Error message) 框中，将显示有关该错误原因的详细信息。
编码器值有效	编码器值有效。
仿真激活	在 CPU 中对轴进行仿真。设定值未输出到驱动装置中。
驱动装置错误	驱动装置因“驱动装置就绪”信号丢失而报错。
需要重新启动	已在 CPU RUN 模式下将已修改的轴组态下载到装载存储器。要将已修改组态下载到工作存储器，需要重新启动轴。使用运动控制指令 MC_Reset 执行此操作。

“信息消息”(Info message) 框会显示有关轴状态的高级信息。

“错误消息”(Error message) 框会显示当前错误。

单击“确认”(Acknowledge)，确认所有已清除的错误。

## 优化增益设置

可以在此区域中设置增益优化：

- “**预控制”(Precontrol)**

在此字段中，可以百分比形式组态位置控制器的速度预控制值。

- “**距离”(Distance)**

在此字段中，可组态一个测试步的负载距离。

- “**自定义动态”(Customize dynamics)** 复选框

选择此选项可调整优化的加速度和最大加速度。

- “**速度”(Velocity)**

在此字段中，可组态一个测试步的最大速度。

- “**加速度”(Acceleration)**

在此域中，组态一个测试步的加速度。

- “**测量持续时间”(Measurement duration)**

根据所选择的加速度、速度和距离，对测量持续时间进行重新计算和输入。

在此之后，可调整测量的持续时间。

- “**增益”(Gain)**

在此域中，组态位置控制器的实际增益 ( $K_v$ )。

输入增益时即生效。如果位置控制器的增益 ( $K_v$ ) 太大，会导致驱动器故障。

## 运行测量

在此区域中，执行测试步：

- “**向前”(Forward)** 按钮

使用“向前”(Forward) 按钮，以正方向启动优化测试步。

- “**向后”(Backward)** 按钮

使用“向后”(Backward) 按钮，以负方向启动优化测试步。

- “**停止”(Stop)** 按钮

使用“停止”(Stop) 按钮，将结束当前的优化运动，并结束跟踪记录。

## 8.2 调节

### 跟踪

对于每个测试步，都会自动启动所需参数的跟踪记录，并在完成该测试步后显示该记录。返回主控制后，将删除跟踪记录。

有关跟踪功能的完整说明，请参见 TIA Portal 帮助中有关使用跟踪和逻辑分析器功能的部分。

## 9.1 运动控制语句概述

在用户程序中，可以使用运动控制指令控制轴。这些指令会启动执行所需功能的运动控制命令。

可以从运动控制指令的输出参数中获取运动控制命令的状态及命令执行期间发生的任何错误。适用的运动控制指令有：

- **MC\_Power:** 启用、禁用轴 (V6 及更高版本) (页 228)
- **MC\_Reset:** 确认故障，重新启动工艺对象 (V6 及以上版本) (页 235)
- **MC\_Home:** 使轴归位，设置参考点 (V6 及以上版本) (页 238)
- **MC\_Halt:** 停止轴 (V6 及以上版本) (页 245)
- **MC\_MoveAbsolute:** 轴的绝对定位 (V6 及以上版本) (页 249)
- **MC\_MoveRelative:** 轴的相对定位 (V6 及以上版本) (页 254)
- **MC\_MoveVelocity:** 以设定速度移动轴 (V6 及以上版本) (页 259)
- **MC\_MoveJog:** 在点动模式下移动轴 (V6 及以上版本) (页 265)
- **MC\_CommandTable:** 按照运动顺序运行轴命令 (V6 及以上版本) (页 270)
- **MC\_ChangeDynamic:** 更改轴的动态设置 (V6 及以上版本) (页 274)
- **MC\_ReadParam:** 连续读取定位轴的运动数据 (V6 及以上版本) (页 277)
- **MC\_WriteParam:** 写入定位轴的变量 (V6 及以上版本) (页 280)

### 参见

[创建用户程序 \(页 198\)](#)

[编程注意事项 \(页 202\)](#)

[断电和重新启动后运动控制命令的行为 \(页 204\)](#)

[监视激活的命令 \(页 204\)](#)

[运动控制语句的错误显示 \(页 217\)](#)

## 9.2 创建用户程序

以下部分介绍了如何创建具有用于轴控制的基本组态的用户程序。使用待插入的运动控制指令，可以控制所有可用的轴功能。

### 要求

- 已正确创建并组态工艺对象。

在创建和测试用户程序之前，建议使用轴命令表测试轴功能和系统的相应部分。

### 步骤

要创建符合下述原则的用户程序，请按以下步骤操作：

- 在项目树中，双击代码块（代码块必须在循环程序中调用）。  
代码块在程序编辑器中打开并显示所有可用指令。
- 打开“工艺”(Technology) 类别和“运动控制”(Motion Control) 文件夹。
- 将“MC\_Power”指令拖放到代码块中相应的程序段。  
将打开用于定义背景数据块的对话框。
- 在下一个对话框中，选择其中一个选项：

#### 单背景

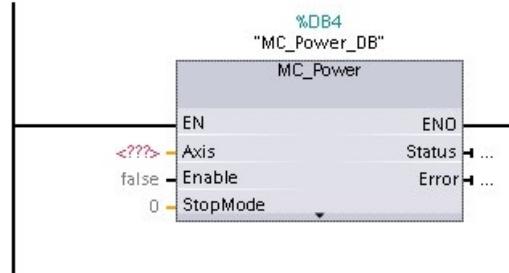
单击“单背景”(Single instance) 并选择是要自动还是手动定义背景数据块的名称和编号。

#### 多重背景

单击“多重背景”(Multi-instance) 并选择是自动还是手动定义多重背景名称。

5. 单击“确定”(OK)。

运动控制指令“MC\_Power”将插入到该程序段中。

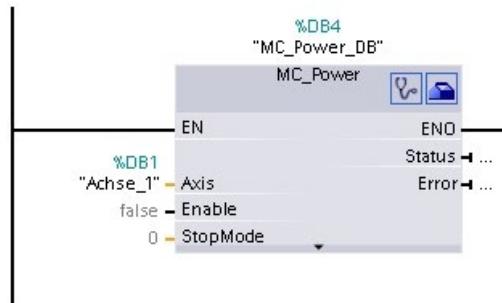


必须初始化标有“<??>”的参数；给所有其它参数分配默认值。

黑体显示的参数是使用运动控制指令时所必需的参数。

## 9.2 创建用户程序

6. 在项目树中选择工艺对象并将其拖放到 <???> 上。



选择工艺对象数据块后，以下按钮将可用：



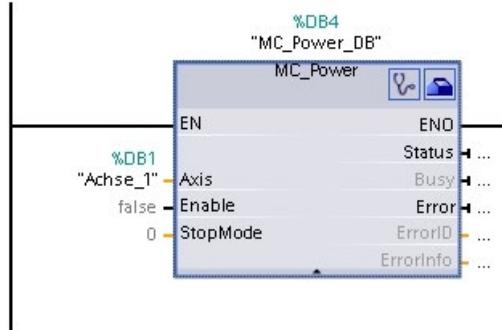
如果想要打开工艺对象的诊断对话框，则单击听诊器图标。



如果想要打开工艺对象的组态视图，则单击工具箱图标。



单击向下箭头图标，可查看运动控制指令的其它参数。



也可使用此时可见的灰显参数。

7. 按照步骤 3 至 6 添加您所选择的运动控制指令。

## 结果

已在用户程序中创建用于轴控制的基本组态。

在用户程序的其它部分中初始化运动控制指令的输入参数，启动需要对“轴”工艺对象执行的作业。

评估运动控制指令的输出参数和数据块中的变量，跟踪启动的作业和轴状态。

有关运动控制指令的参数的详细信息，请参见详细说明。

## 参见

[运动控制语句概述 \(页 197\)](#)

[编程注意事项 \(页 202\)](#)

[断电和重新启动后运动控制命令的行为 \(页 204\)](#)

[监视激活的命令 \(页 204\)](#)

[运动控制语句的错误显示 \(页 217\)](#)

## 9.3 编程注意事项

创建用户程序时，请注意以下信息：

- **循环调用所使用的运动控制指令**

命令的当前执行状态通过运动控制指令的输出参数提供。每次调用运动控制指令时都会更新状态。因此，确保循环调用所使用的运动控制指令。

- **传送运动控制指令的参数值**

如果调用了该块，则在输入参数“Execute”的上升沿时将传送输入参数的未决参数值。

运动控制命令将以这些参数值启动。在下次启动运动控制命令前，不会传送运动控制指令随后发生更改的参数值。

运动控制指令“MC\_Power”的 StopMode 输入参数和运动控制指令“MC\_MoveJog”的 Velocity 输入参数是个例外。输入参数中的更改同样适用于“Enable” = TRUE 或“JogForward”和“JogBackward”。

- **编程时考虑状态信息**

在逐步执行运动控制命令时，确保等到激活的命令执行完成后再启动新命令。使用运动控制指令的状态消息和工艺对象的“StatusBits”变量，可以检查激活的命令是否完成。

在下例中，请按照所示顺序执行。如果未遵守此顺序，将显示轴或命令错误。

- 使用运动控制指令“MC\_Power”启用轴

只有启用轴后，轴才可以执行运动作业。通过对变量 <轴名>.StatusBits.Enable = TRUE 与运动控制指令“MC\_Power”的输出参数 Status = TRUE 进行与运算，将检查轴是否已启用。

- 使用运动控制指令“MC\_Reset”确认错误

启动运动控制命令前，必须使用“MC\_Reset”确认需要确认的错误。消除错误原因并使用运动控制指令“MC\_Reset”确认错误。启动新命令前，需检查错误是否已成功确认。为此，需要对变量 <轴名称>.StatusBits.Error = FALSE 和运动控制指令“MC\_Reset”的输出参数 Done = TRUE 进行与运算。

- 使用运动控制指令“MC\_Home”使轴回原点

在可以启动 MC\_MoveAbsolute 命令之前，轴必须回原点。通过对变量 <轴名称>.StatusBits.HomingDone = TRUE 与运动控制指令“MC\_Home”的输出参数 Done = TRUE 进行与运算，可检查轴是否已回原点。

- 执行运动控制命令覆盖

用于移动轴的运动控制作业还可作为覆盖作业执行。

如果在一个运动控制命令处于激活状态时对轴另启动一个新的运动控制命令，则在完全执行现有命令前，新命令将覆盖激活的命令。被覆盖的命令将通过在运动控制指令中使 `CommandAborted = TRUE` 发出信号。可以使用 `MC_MoveAbsolute` 命令覆盖激活的 `MC_MoveRelative` 命令。

- 避免多次使用相同的背景

运动控制命令的所有相关信息都存储在其背景中。

如果要跟踪当前命令的状态，请勿使用该背景启动新命令。如果要单独跟踪命令，则需使用不同的背景。如果将同一个背景用于多个运动控制命令，则各命令的状态和错误信息将会相互覆盖。

在用户程序中，各个轴由运动控制指令“`MC_Power`”搭配不同的背景数据块进行分别调用。

- 以不同优先级（运行级别）调用运动控制指令

在未互锁的情况下，可能无法以不同优先级调用具有相同背景的运动控制指令。要了解如何调用已锁定的运动控制指令，请参见“跟踪较高优先级（运行级别）的命令”(页 287)。

## 参见

[运动控制语句概述 \(页 197\)](#)

[创建用户程序 \(页 198\)](#)

[断电和重新启动后运动控制命令的行为 \(页 204\)](#)

[监视激活的命令 \(页 204\)](#)

[运动控制语句的错误显示 \(页 217\)](#)

[在更高优先级等级（执行级别）中跟踪作业 \(页 287\)](#)

## 9.4 断电和重新启动后运动控制命令的行为

断电或 CPU STOP 将中止所有激活的运动控制作业。包括脉冲和方向输出在内的所有 CPU 输出都将被重置。

随后上电或 CPU 重新启动 (CPU RUN) 后，将重新初始化工艺对象和运动控制作业。

工艺对象的所有实际数据以及之前激活的运动控制作业的所有状态和错误信息，都将重置为各自的初始值。

只有再次使用运动控制指令“MC\_Power”启用轴后，才可重新使用该轴。如果需要回原点，则必须再次使用运动控制指令“MC\_Home”使轴回原点。使用绝对值编码器时，断电后将保持回原点状态。

### 参见

[运动控制语句概述 \(页 197\)](#)

[创建用户程序 \(页 198\)](#)

[编程注意事项 \(页 202\)](#)

[监视激活的命令 \(页 204\)](#)

[运动控制语句的错误显示 \(页 217\)](#)

## 9.5 监视激活的命令

### 9.5.1 监视激活的命令

有三个典型组可用于跟踪激活的运动控制命令：

- 具有输出参数“Done”的运动控制指令
- 运动控制指令“MC\_MoveVelocity”
- 运动控制指令“MC\_MoveJog”

### 9.5.2 具有输出参数“Done”的运动控制指令

具有输出参数“Done”的运动控制指令通过输入参数“Execute”启动，并且具有明确的结论（例如，对于运动控制指令“MC\_Home”：回原点已成功）。该命令已完成，轴处于停止状态。

以下运动控制指令的命令具有一个指定功能：

- MC\_Reset
- MC\_Home
- MC\_Halt
- MC\_MoveAbsolute
- MC\_MoveRelative
- MC\_CommandTable（自工艺对象 V2 起）
- MC\_ChangeDynamic（自工艺对象 V2 起）
- MC\_WriteParam（自工艺对象 V4 起）
- MC\_ReadParam（自工艺对象 V4 起）

如果命令已成功完成，则输出参数“Done”的值为 TRUE。

输出参数“Busy”、“CommandAborted”和“Error”分别指示命令仍在处理、已中止或有未决的错误。运动控制指令“MC\_Reset”不会被中止，因此没有“CommandAborted”输出参数。运动控制指令“MC\_ChangeDynamic”立即完成，因此没有“Busy”或“CommandAborted”输出参数。

在运动控制命令的执行过程中，输出参数“Busy”的值将为 TRUE。如果命令已完成、中止或因错误停止，则输出参数“Busy”的值将变为 FALSE。无论输入参数“Execute”的信号状态是什么，都会发生这种变化。

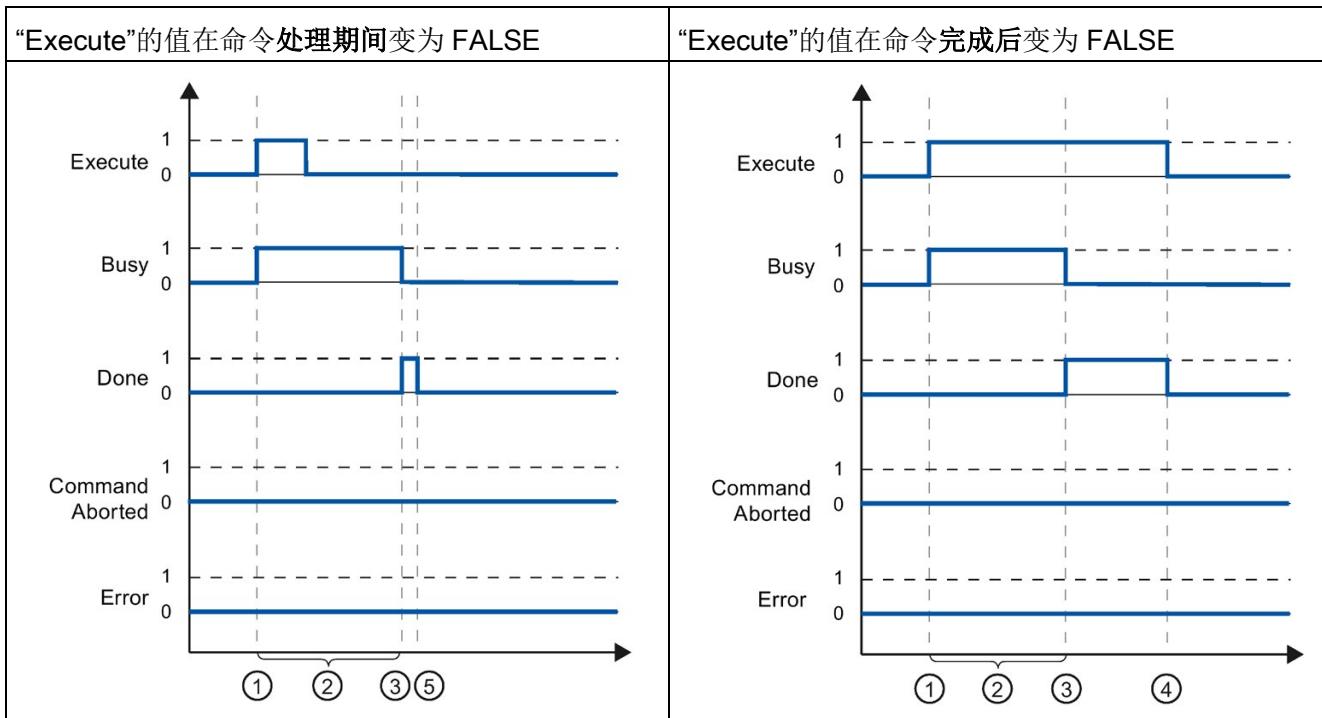
输出参数“Done”、“CommandAborted”和“Error”的值至少在一个周期内都为 TRUE。当输入参数“Execute”设置为 TRUE 时，将锁存这些状态消息。

下图针对各种示例情况显示了状态位的操作：

## 9.5 监视激活的命令

## 命令执行完成

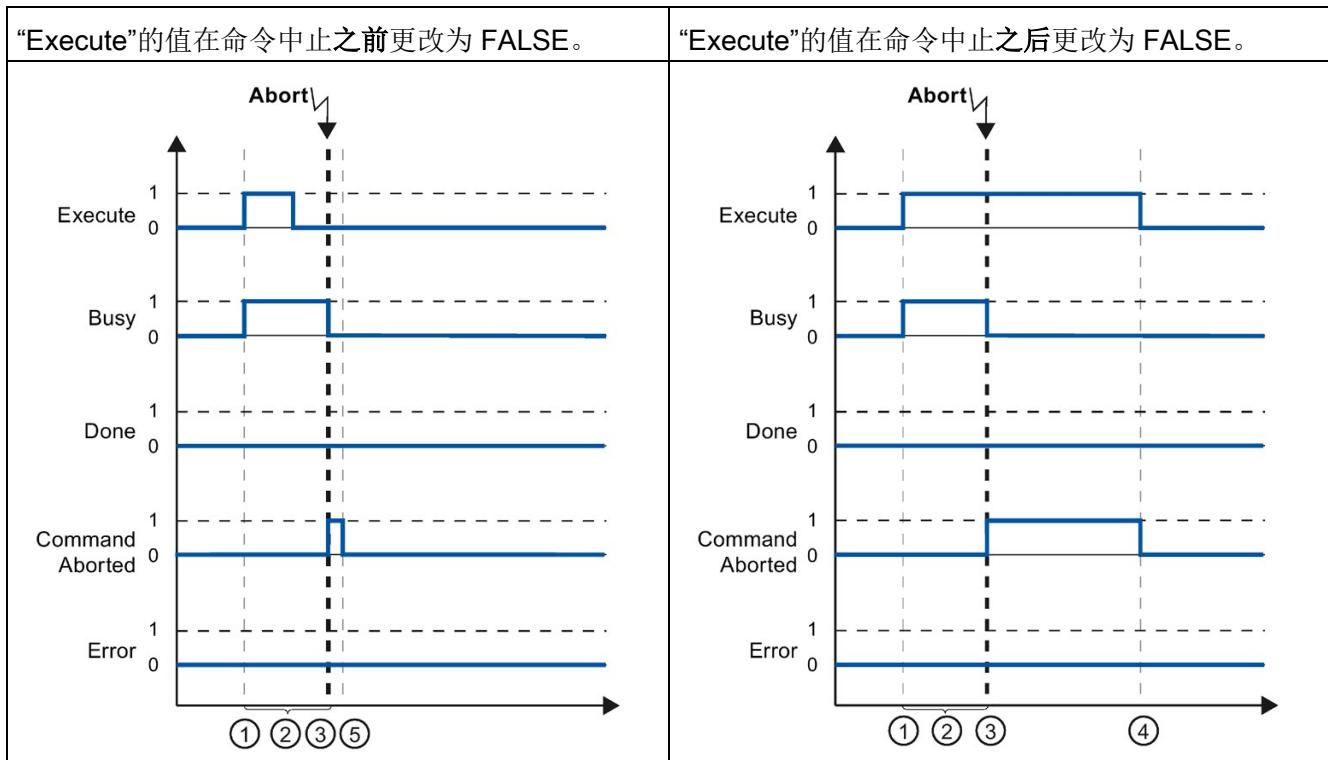
如果运动控制命令已在对其下结论前完全执行，则将通过输出参数“Done”的值为 TRUE 的形式对此进行指示。输入参数“Execute”的信号状态影响输出参数“Done”中的显示持续时间：



- ① 输入参数“Execute”出现上升沿时启动命令。根据编程情况，“Execute”在命令的执行过程中仍然可能复位为值 FALSE，或者保持为值 TRUE，直到命令执行完成为止。
- ② 激活命令时，输出参数“Busy”的值将为 TRUE。
- ③ 命令执行结束后（例如，对于运动控制指令“MC\_Home”：回原点已成功），输出参数“Busy”变为 FALSE，“Done”变为 TRUE。
- ④ 如果“Execute”的值在命令完成之前保持为 TRUE，则“Done”的值也将保持为 TRUE 并且其值随“Execute”一起变为 FALSE。
- ⑤ 如果“Execute”在命令执行完成之前设置为 FALSE，则“Done”的值仅在一个执行周期内为 TRUE。

## 中止命令

如果运动控制命令在执行期间中止，则将通过输出参数“CommandAborted”的值为 TRUE 的方式对此进行指示。输入参数“Execute”的信号状态影响输出参数“CommandAborted”中的显示持续时间：

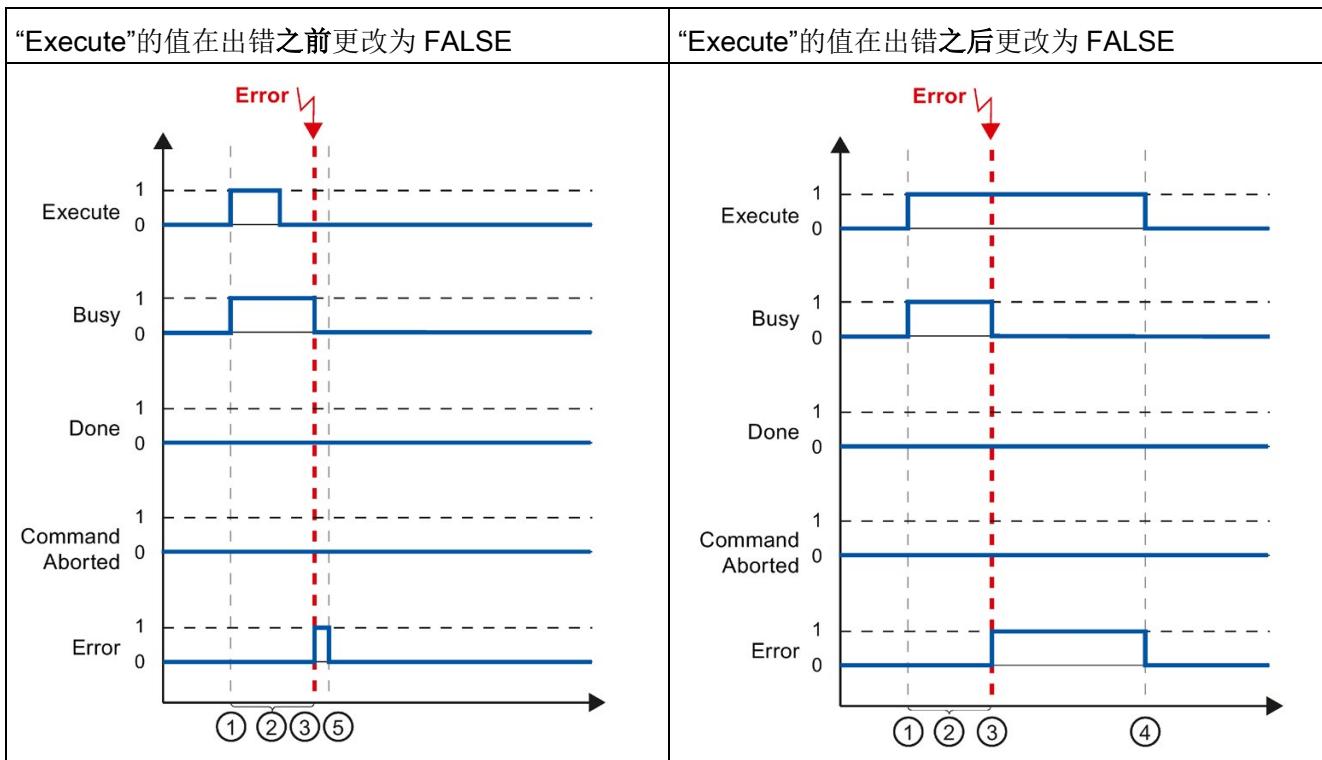


- ① 输入参数“Execute”出现上升沿时启动命令。根据编程情况，“Execute”在命令的执行过程中仍然可能复位为值 FALSE，或者保持为值 TRUE，直到命令执行完成为止。
- ② 激活命令时，输出参数“Busy”的值将为 TRUE。
- ③ 在执行命令的过程中，该命令可由其它运动控制命令中止。如果中止命令，则输出参数“Busy”的值将变为 FALSE，“CommandAborted”的值将变为 TRUE。
- ④ 如果“Execute”的值在中止命令之前保持为 TRUE，则“CommandAborted”的值也将保持为 TRUE 并且其值随“Execute”一起变为 FALSE。
- ⑤ 如果“Execute”在中止命令执行之前设置为 FALSE，则“CommandAborted”的值仅在一个执行周期内为 TRUE。

## 9.5 监视激活的命令

## 命令执行过程中出错

如果在执行运动控制命令过程中出错，则将通过输出参数“Error”的值为 TRUE 的方式对此进行指示。输入参数“Execute”的信号状态影响输出参数“Error”中的显示持续时间：



- ① 输入参数“Execute”出现上升沿时启动命令。根据编程情况，“Execute”在命令的执行过程中仍然可能复位为值 FALSE，或者保持为值 TRUE，直到命令执行完成为止。
- ② 激活命令时，输出参数“Busy”的值将为 TRUE。
- ③ 执行命令过程中出错。出错时，输出参数“Busy”的值将变为 FALSE 且“Error”的值将变为 TRUE。
- ④ 如果“Execute”的值在出错之前保持为 TRUE，则“Error”的值也将保持为 TRUE 并且其值仅随“Execute”一起变为 FALSE。
- ⑤ 如果“Execute”已在出错之前设置为 FALSE，则“Error”的值仅在一个执行周期内为 TRUE。

### 9.5.3 运动控制指令 MC\_MoveVelocity

在参数“Execute”的上升沿启动“MC\_MoveVelocity”命令。当达到指定的速度且轴以恒定速度移动时，就达到了命令目标。在达到并保持指定的速度时，将在参数“InVelocity”中用值为 TRUE 进行指示。

例如，可通过“MC\_Halt”命令停止轴运动。

输出参数“Busy”、“CommandAborted”和“Error”分别指示命令仍在处理、已中止或有未决的错误。

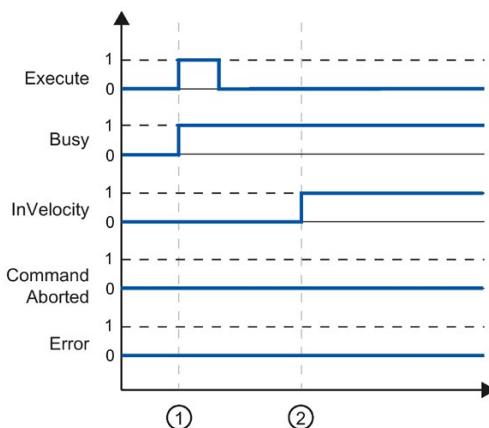
在运动控制命令的执行过程中，输出参数“Busy”的值将为 TRUE。如果该命令因其它命令或错误而停止，则输出参数“Busy”的值将更改为 FALSE。无论输入参数“Execute”的信号状态是什么，都会发生这种变化。

输出参数“CommandAborted”和“Error”的值至少在一个周期内保持为 TRUE。当输入参数“Execute”设置为 TRUE 时，将锁存这些状态消息。

下图针对各种示例情况显示了状态位的操作：

#### 达到设置的速度

如果运动控制命令在达到设置的速度前执行完毕，则将通过输出参数“InVelocity”的值为 TRUE 的方式对此进行指示。参数“Execute”不会影响“InVelocity”参数中的指示持续时间。

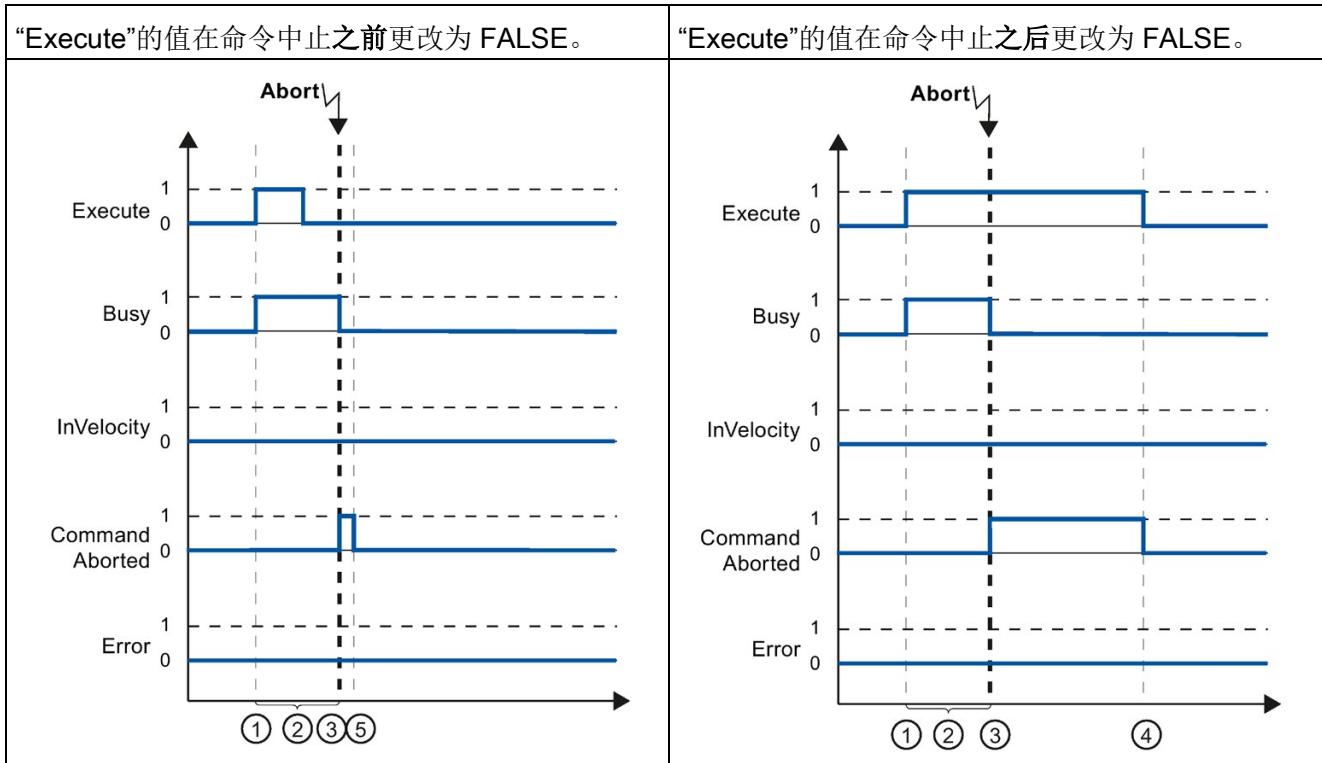


①	将在参数“Execute”的上升沿启动作业。根据编程，在达到设置的速度之前或之后，可以将“Execute”复位为值 FALSE。在该作业处于激活状态期间，参数“Busy”的值为 TRUE。
②	在达到指定的速度时，参数“InVelocity”的值将变为 TRUE。在“MC_MoveVelocity”命令由另一运动控制命令覆盖或因错误停止前，“Busy”和“InVelocity”参数的值将一直为 TRUE。

## 9.5 监视激活的命令

## 命令在达到设置的速度前中止

如果运动控制命令在达到设置的速度前中止，则通过输出参数“CommandAborted”的 TRUE 值对此进行指示。输入参数“Execute”的信号状态影响输出参数“CommandAborted”中的显示持续时间。



- ① 输入参数“Execute”出现上升沿时启动命令。根据编程情况，“Execute”在命令执行期间仍然可能被重置为值 FALSE，或者保持为值 TRUE，直到命令中止为止。
- ② 激活命令时，输出参数“Busy”的值将为 TRUE。
- ③ 在执行命令的过程中，该命令可由其它运动控制命令中止。如果中止命令，则输出参数“Busy”的值将变为 FALSE，“CommandAborted”的值将变为 TRUE。
- ④ 如果“Execute”的值在命令中止之前保持为 TRUE，则“CommandAborted”的值也将保持为 TRUE 并且其状态会随“Execute”一起变为 FALSE。
- ⑤ 如果“Execute”在命令中止之前重置为 FALSE，则“CommandAborted”的值仅在一个执行周期内为 TRUE。

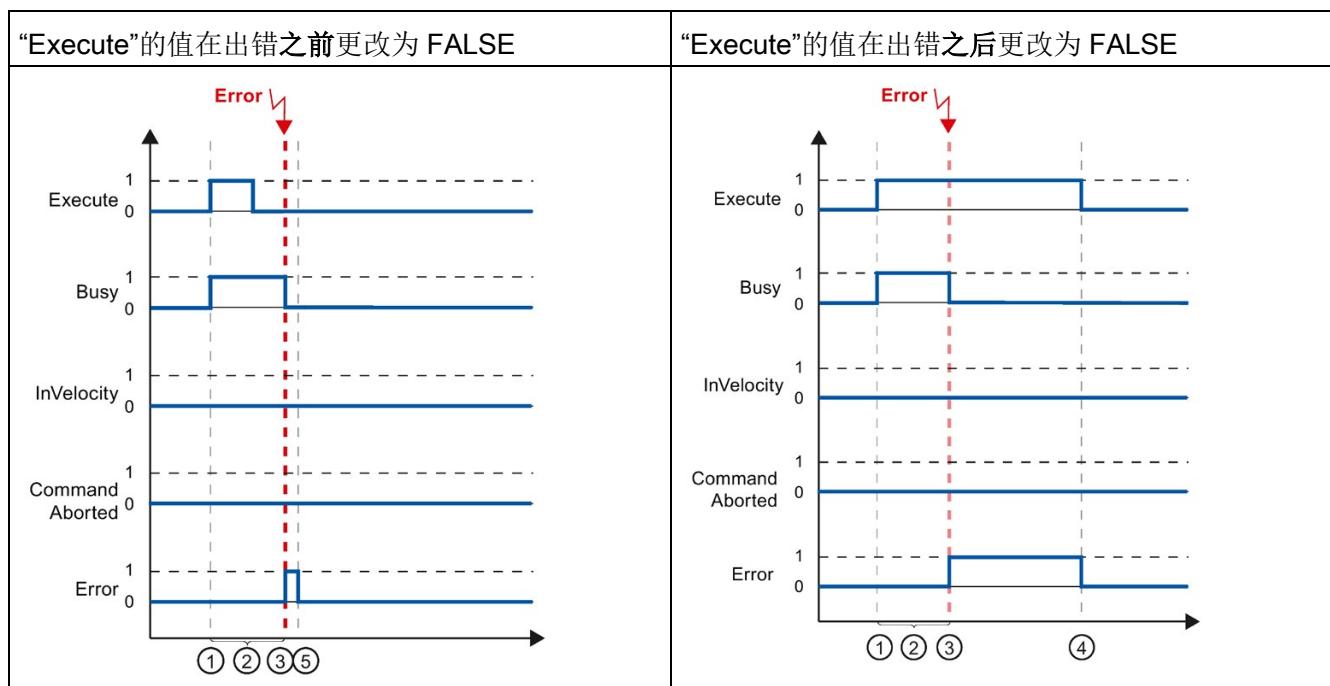
### 说明

在以下条件下，输出参数“CommandAborted”不指示出现中止：

已达到设置的速度，输入参数“Execute”的值为 FALSE，并且已启动一个新的运动控制命令。

### 达到设置速度前出错

如果在运动控制命令执行期间，达到设置速度之前出错，则通过输出参数“Error”的 TRUE 值对此进行指示。输入参数“Execute”的信号状态影响输出参数“Error”中的显示持续时间：



- ① 输入参数“Execute”出现上升沿时启动命令。根据编程情况，“Execute”在命令执行期间仍然可能重置为值 FALSE，或者保持为值 TRUE，直到出现错误为止。
- ② 激活命令时，输出参数“Busy”的值将为 TRUE。
- ③ 执行命令过程中出错。出错时，输出参数“Busy”的值将变为 FALSE 且“Error”的值将变为 TRUE。
- ④ 如果“Execute”的值在出错之前保持为 TRUE，则“Error”的值也将保持为 TRUE 并且其状态仅随“Execute”一起变为 FALSE。
- ⑤ 如果“Execute”已在出错之前重置为 FALSE，则“Error”的值仅在一个执行周期内为 TRUE。

---

### 说明

在以下条件下，输出参数“Error”不指示出现错误：

已达到设置的速度，输入参数“Execute”的值为 FALSE，并且发生轴错误（例如，逼近软限位开关）。

轴错误仅在运动控制指令“MC\_Power”中指示。

---

## 9.5.4 运动控制指令 MC\_MoveJog

运动控制指令“MC\_MoveJog”的命令将实现点动操作。

运动控制命令“MC\_MoveJog”无需定义明确的结束标记。第一次达到设置的速度且轴恒速运转时，就达到了该命令的目标。如果达到设置的速度，则通过输出参数“InVelocity”的 TRUE 值对此进行指示。

输入参数“JogForward”或“JogBackward”的值已设置为 FALSE 并且轴已停止时，命令完成。

输出参数“Busy”、“CommandAborted”和“Error”分别指示命令仍在处理、已中止或有未决的错误。

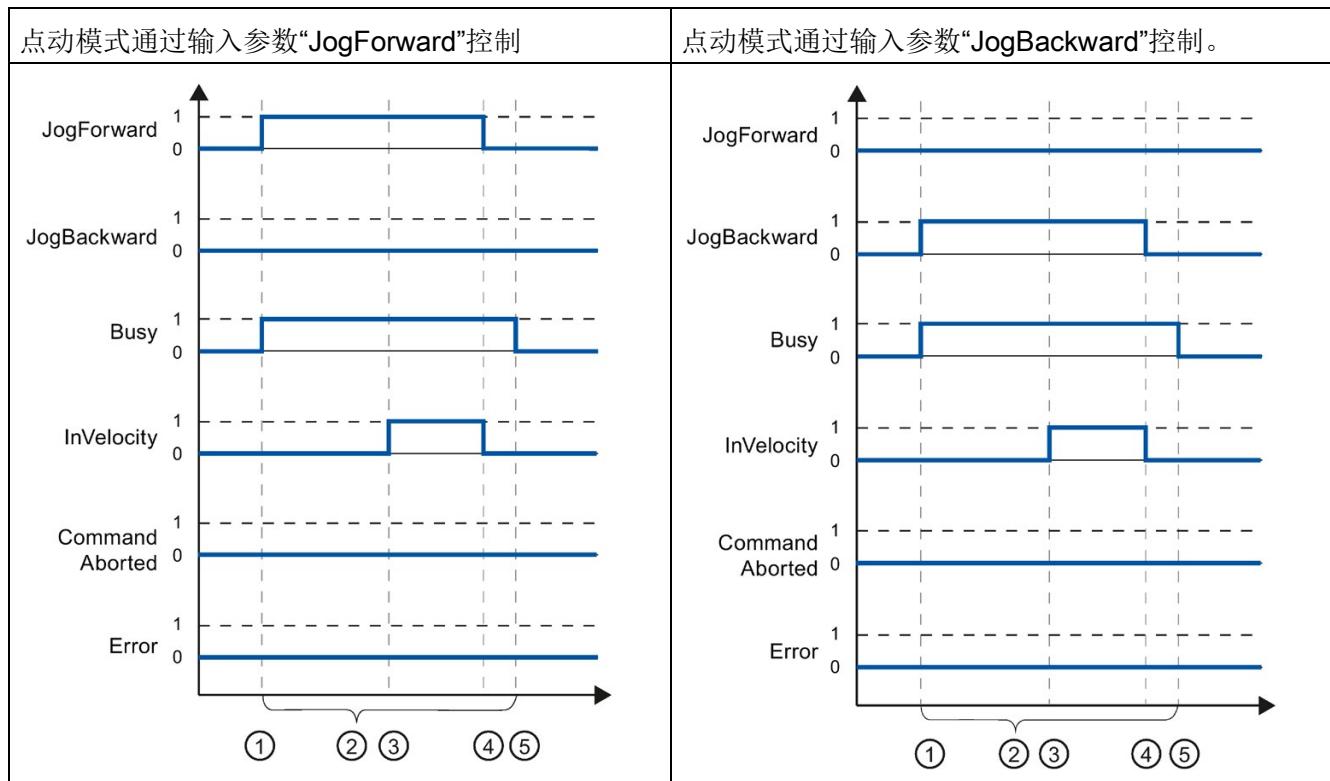
在运动控制命令的处理过程中，输出参数“Busy”的值将为 TRUE。如果命令已完成、中止或因错误停止，则输出参数“Busy”的值将变为 FALSE。

只要轴在以设置的速度运转，输出参数“InVelocity”的值就为 TRUE。输出参数“CommandAborted”和“Error”保持该状态至少一个周期。只要输入参数“JogForward”或“JogBackward”设置为 TRUE，就锁存这些状态消息。

下图针对各种示例情况显示了状态位的操作：

## 达到并维持设置的速度

如果运动控制命令在达到设置的速度前执行完毕，则将通过输出参数“InVelocity”的值为 TRUE 的方式对此进行指示。

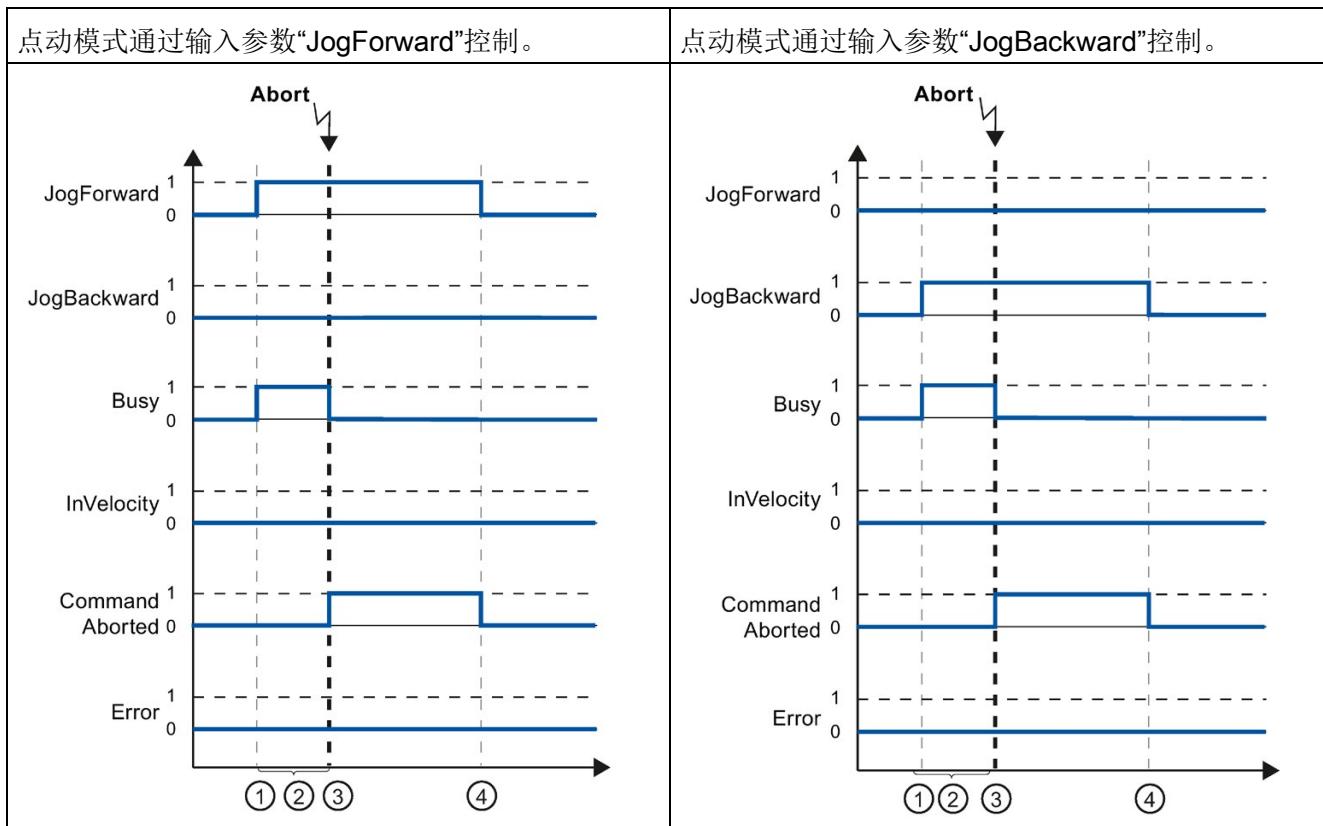


①	输入参数“JogForward”或“JogBackward”的上升沿时，将启动该命令。
②	激活命令时，输出参数“Busy”的值将为 TRUE。
③	达到设置的速度时，输出参数“InVelocity”的值变为 TRUE。
④	输入参数“JogForward”或“JogBackward”的值重置为 FALSE 时，轴运动结束。轴开始减速。结果，轴不再恒速运转并且输出参数“InVelocity”的状态变为 FALSE。
⑤	如果轴已停止，则运动控制命令执行完成。同时将输出参数“Busy”的值变为 FALSE。

## 9.5 监视激活的命令

## 在执行过程中止命令

如果运动控制命令在执行期间中止，则将通过输出参数“CommandAborted”的值为 TRUE 的方式对此进行指示。该行为与是否达到设置的速度无关。



- ① 输入参数“JogForward”或“JogBackward”的上升沿时，将启动该命令。
- ② 激活命令时，输出参数“Busy”的值将为 TRUE。
- ③ 在执行命令的过程中，该命令可由其它运动控制命令中止。如果中止命令，则输出参数“Busy”的值将变为 FALSE，“CommandAborted”的值将变为 TRUE。
- ④ 输入参数“JogForward”或“JogBackward”的值重置为 FALSE 时，输出参数“CommandAborted”的值将变为 FALSE。

---

### 说明

如果满足以下所有条件，则将在输出参数“CommandAborted”中指示命令中止并且仅持续一个执行周期：

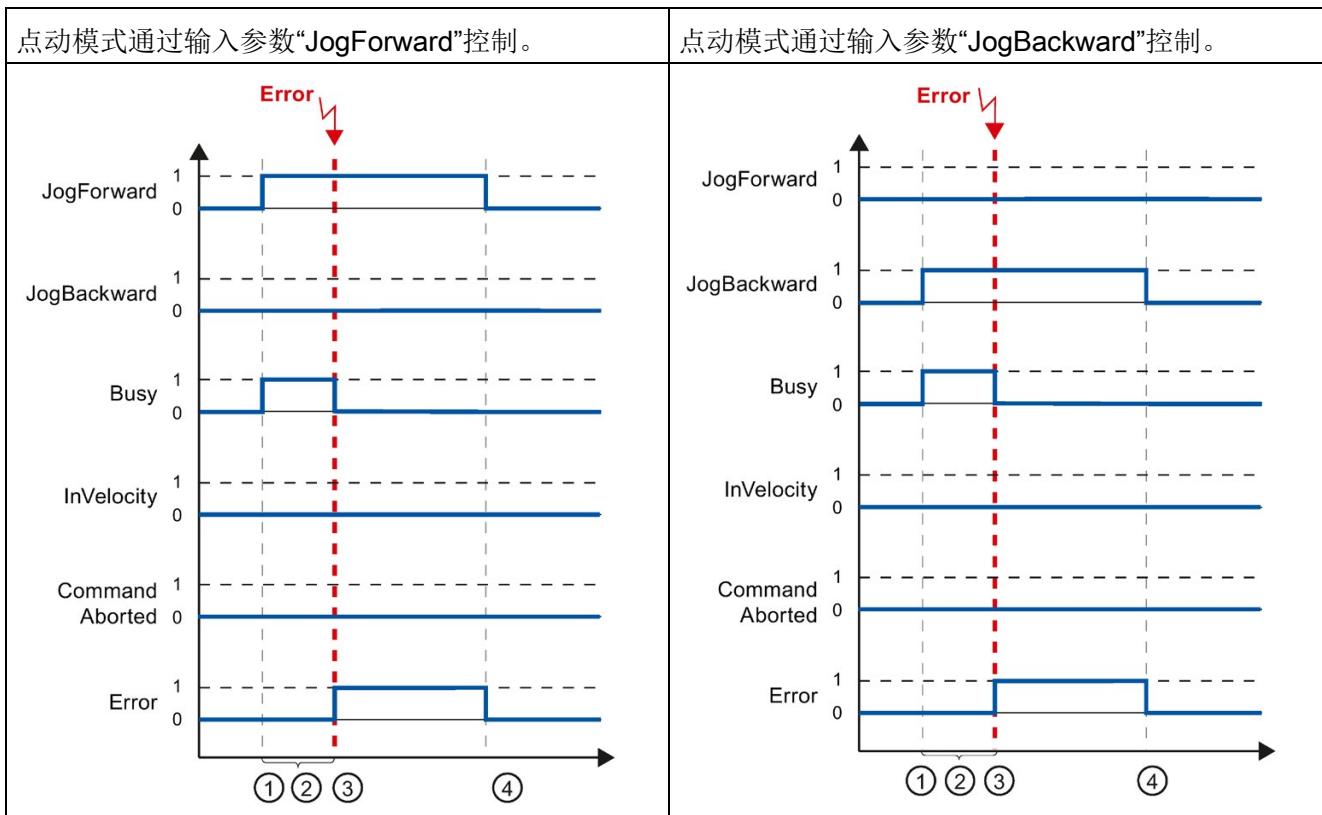
输入参数“JogForward”和“JogBackward”的值为 FALSE（但轴仍在减速），并且新的运动控制命令已启动。

---

## 9.5 监视激活的命令

## 在执行命令过程中出错

如果在执行运动控制命令中出错，则将通过输出参数“Error”的值为 TRUE 的方式对此进行指示。该行为与是否达到设置的速度无关。



① 输入参数“JogForward”或“JogBackward”的上升沿时，将启动该命令。
② 激活命令时，输出参数“Busy”的值将为 TRUE。
③ 执行命令过程中出错。出错时，输出参数“Busy”的值将变为 FALSE 且“Error”的值将变为 TRUE。
④ 输入参数“JogForward”或“JogBackward”的值重置为 FALSE 时，输出参数“Error”的值将变为 FALSE。

## 说明

如果满足以下所有条件，则在输出参数“Error”中指示出错并且仅持续一个执行周期：

输入参数“JogForward”和“JogBackward”的值为 FALSE（但轴仍在减速），并且发生新错误（例如，逼近软限位开关）。

## 9.6 运动控制语句的错误显示

运动控制指令将在运动控制指令的输出参数“Error”、“ErrorID”和“ErrorInfo”中指示运动控制命令和工艺对象中的所有错误。

### 在输出参数“Error”、“ErrorID”和“ErrorInfo”中显示错误信息

如果输出参数“Error”的值为 TRUE，则无法执行整个命令或其中各个部分。错误原因通过输出参数“ErrorID”的值进行指示。输出参数 “ErrorInfo”的值返回关于错误原因的详细信息。我们区分以下用于错误指示的错误类别：

- **伴随轴停止的运行错误（例如，“已逼近硬限位开关”）**

伴随轴停止的运行错误发生在用户程序运行期间。如果轴正在运转，则将以组态的减速度或急停减速度使轴停止，具体视错误情况而定。此类错误在错误触发运动控制指令和运动控制指令“MC\_Power”中指示。

- **不伴随轴停止的运行错误（例如，“轴未回原点”）**

不伴随轴停止的运行错误发生在用户程序运行期间。如果轴正在运转，则将继续运转。此类错误仅在触发错误的运动控制指令中指示。

- **运动控制指令中组态出错  
(例如，参数“Velocity”的值错误)**

如果在运动控制指令的输入参数中指定的信息不正确，则将发生参数化错误。如果轴正在运转，则将继续运转。此类错误仅在触发错误的运动控制指令中指示。

- **“轴”工艺对象的组态错误（例如，“加速度”的值无效）**

如果在轴组态中错误组态一个或多个参数，或者在程序运行期间错误修改可编辑的组态数据，则将发生组态错误。以组态的急停减速度使正在运转的轴停止。此类错误在错误触发运动控制指令和运动控制指令“MC\_Power”中指示。

- **“命令表”工艺对象的组态错误（例如，“速度”的值无效）**

如果在轴命令表中错误设置一个或多个参数，或者在程序运行期间错误修改可编程的组态数据，则将发生组态错误。如果轴正在运转，则将继续运转。此类错误仅在运动控制指令“MC\_CommandTable”中指示。

- **内部错误**

发生内部错误时，将使轴停止。此类错误在错误触发运动控制指令中指示，有时也在运动控制指令“MC\_Power”中指示。

有关 ErrorID 和 ErrorInfo 的详细说明及其解决方法，请参见附录 (页 414)。

## 参见

- [运动控制语句概述 \(页 197\)](#)
- [创建用户程序 \(页 198\)](#)
- [编程注意事项 \(页 202\)](#)
- [断电和重新启动后运动控制命令的行为 \(页 204\)](#)
- [ErrorID 和 ErrorInfo 列表 \(工艺对象 V4...5\) \(页 414\)](#)
- [监视激活的命令 \(页 204\)](#)

## 9.7 重新启动工艺对象

### 说明

在接通 CPU 之后或将工艺对象下载到 CPU 之后，系统将使用工艺对象数据块的起始值自动初始化工艺对象。如果在向 CPU 重新进行加载时检测到会引起重新启动的相关变化，则将自动重新启动工艺对象。

如果在 RUN 模式下用户程序更改了与重新启动相关的数据，则用户必须对工艺对象进行重新初始化，以使所做的更改生效。

如果要在重新启动工艺对象之后保留工艺对象数据块中的更改，则必须使用扩展指令“WRIT\_DB”将更改写入装载存储器中的起始值中。

### 必须重新启动

如果需要重新启动工艺对象，则可在“工艺对象 > 诊断 > 状态和错误位 > 状态消息 > 轴 > 需要重新启动”(Technology object > Diagnostics > Status and error bits > Status messages > Axis > Restart required) 下和工艺对象的“<轴名称>.StatusBits.RestartRequired”变量中指出。

### 重新启动工艺对象

可通过带有参数“Restart”= TRUE 的“MC\_Reset”运动控制指令，触发工艺对象的重新启动。

重新启动将使用增量实际值（<轴名称>.StatusBits.HomingDone）重置工艺对象的“已回原点”(Homed) 状态。

## 9.8 函数块的参数传递

如果要在不同工艺对象的运动控制指令中重复使用一个函数块，则需在函数块的块接口中创建相应工艺对象的数据类型输入参数。可通过直接输入，在块接口中分配数据类型。之后，该参数将作为工艺对象的引用传送到运动控制指令的参数“axis”中。工艺对象的数据类型与相关工艺数据块的结构相对应。

通过指定数据类型，可在函数块中寻址该工艺对象的变量（<块接口的参数>.<工艺对象的变量>）。

如果无需访问该工艺对象的变量，则可使用数据类型“DB\_ANY”。数据类型“DB\_Any”可用于获取更多变量的编程。

下表列出了引用工艺对象的数据类型：

工艺对象	引用工艺对象的数据类型
定位轴	TO_PositioningAxis
命令表	TO_CommandTable

### 示例 1

下表列出了所用变量的定义：

操作数	声明	数据类型	说明
axis	Input	TO_PositioningAxis	工艺对象的引用
on	Input	BOOL	启用该轴的信号
actPosition	Output	Real	查询工艺数据块的实际位置
instMC_POWER	Static	MC_POWER	运动控制指令 MC_Power 的多重实例

在以下 STL 程序中，说明了该任务的具体执行过程：

SCL	说明
#instMC_POWER(Axis := #axis, Enable := #on); #actPosition := #axis.ActualPosition;	//Call of the Motion Control instruction MC_Power //with enable of the axis //查询工艺数据块的实际位置

## 9.8 函数块的参数传递

### 示例 2

数据类型“DB\_Any”为工艺对象的数据类型传输提供了另一种选项。数据类型“DB\_Any”可以在运行时间内分配到程序。

示例显示用于传输工艺指定数据类型到相关指令的示例，如“MC\_CommandTable”，创建为多重背景。第一个选项显示了数据类型“TO\_PositioningAxis”的使用。第二个选项显示了命令表工艺对象作为“cmdTablToUse”输入的函数的简单传输。根据输入的值，三个“cmdTabl”输入中的一个通过“tempCmdTableSel”传输到“MC\_CommandTable”指令中。

下表列出了所用变量的声明：

变量	声明	数据类型	说明
axis	Input	TO_PositioningAxis	定位轴
cmdTabl1	Input	DB_ANY	第一命令表
cmdTabl2	Input	DB_ANY	第二命令表
cmdTabl3	Input	DB_ANY	第三命令表
cmdTablToUse	Input	Int	选择，命令表 1 至 3
instMC_CommandTable	Static	MC_CommandTable	MC_CommandTable 的多重背景
tempCmdTableSel	Temp	DB_ANY	当前命令表

以下示例为基础过程：

SCL	说明
<pre>CASE #cmdTablToUse OF   1: #tempCmdTableSel := #cmdTabl1;           //Program for scenario 1   2: #tempCmdTableSel := #cmdTabl2;           //Program for scenario 2   3: #tempCmdTableSel := #cmdTabl3;           //Program for scenario 3   ELSE     #tempCmdTableSel := #cmdTabl1;             //Program for all other values   //--&gt;Default setting 1st command table END CASE;  #instMC_CommandTable(Axis:=#axis,                       CommandTable:=#tempCmdTableSel);</pre>	

### 更多信息

有关程序示例更多信息，请在以下常见问题解答中使用数据类型“DB\_Any”：

<https://support.industry.siemens.com/cs/ww/en/view/109750880>

(<https://support.industry.siemens.com/cs/ww/en/view/109750880>)

## 10.1 状态和错误位（自工艺对象 V4 起）

在 TIA Portal 中使用诊断功能“状态和错误位”(Status and error bits) 可监视轴的最重要状态和错误消息。当轴激活时，可以在“手动控制”模式和“自动控制”模式下在线显示诊断功能。所显示的状态错误消息具有下列以下含义：

### 状态消息

状态消息 - 轴	说明
已启用	轴已启用且准备就绪，可通过运动控制命令进行控制。 (工艺对象的变量：<轴名称>.StatusBits.Enable)
已回原点	轴已回原点，可执行运动控制指令“MC_MoveAbsolute”的绝对定位命令。对于相对定位而言，轴不必回原点。特殊情况： <ul style="list-style-type: none"> <li>• 主动回原点过程中，该状态为 FALSE。</li> <li>• 如果回原点的轴经受被动回原点，则在被动回原点过程中该状态设置为 TRUE。</li> </ul> (工艺对象的变量：<轴名称>.StatusBits.HomingDone)
轴错误	“轴”工艺对象发生错误。有关错误的详细信息，请参见自动控制模式下的运动控制指令的 ErrorCode 和 ErrorInfo 参数。在手动模式下，轴控制面板的“错误消息”(Error message) 框可显示有关错误原因的详细信息。 (工艺对象的变量：<轴名称>.StatusBits.Error)
控制面板激活	在轴控制面板中启用了“手动控制”(Manual control) 模式。轴控制面板对“轴”工艺对象具有优先控制权。不能通过用户程序来控制轴。 (工艺对象的变量：<轴名称>.StatusBits.ControlPanelActive)
需要重新启动	已在 CPU RUN 模式下将已修改的轴组态下载到装载存储器。要将已修改组态下载到工作存储器，需要重新启动轴。使用运动控制指令 MC_Reset 执行此操作。 (工艺对象的变量：<轴名称>.StatusBits.RestartRequired)

## 10.1 状态和错误位 (自工艺对象 V4 起)

状态消息 - 驱动器	说明
就绪	驱动装置已就绪，可以运行。 (工艺对象的变量: <轴名称>.StatusBits.DriveReady)
驱动装置错误	驱动装置因“驱动装置就绪”信号丢失而报错。 (工艺对象的变量: <轴名称>.ErrorBits.DriveFault)

状态消息 - 运动	说明
停止	轴处于停止状态。 (工艺对象的变量: <轴名称>.StatusBits.StandStill)
加速度	轴在加速。 (工艺对象的变量: <轴名称>.StatusBits.Accelerating)
恒速	轴在恒速运转。 (工艺对象的变量: <轴名称>.StatusBits.ConstantVelocity)
减速度	轴在减速（速度下降）。 (工艺对象的变量: <轴名称>.StatusBits.Decelerating)

状态消息 - 运动类型	说明
定位	轴执行运动控制指令“MC_MoveAbsolute”、“MC_MoveRelative”或者轴控制面板的定位命令。 (工艺对象的变量: <轴名称>.StatusBits.PositioningCommand)
以速度设定值行进	轴以运动控制指令“MC_MoveVelocity”、“MC_MoveJog”或者轴控制面板的速度设定值执行命令。 (工艺对象的变量: <轴名称>.StatusBits.VelocityCommand)
回原点	轴将执行运动控制指令“MC_Home”或者轴控制面板的回原点命令。 (工艺对象的变量: <轴名称>.StatusBits.HomingCommand)
命令表已激活	该轴由运动控制指令“MC_CommandTable”控制。 (工艺对象的变量: <轴名称>.StatusBits.CommandTableActive)

## 限位开关状态消息

限位开关的状态消息	说明
已到达软限位开关的下限	已到达或超出软限位开关。 (工艺对象的变量: <轴名称>.StatusBits.SWLimitMinActive)
已到达软限位开关的上限	已到达或超出硬限位开关。 (工艺对象的变量: <轴名称>.StatusBits.SWLimitMaxActive)
已到达硬限位开关的下限	已到达或超出硬限位开关的下限。 (工艺对象的变量: <轴名称>.StatusBits.HWLimitMinActive)
已到达硬限位开关的上限	已到达或超出硬限位开关的上限。 (工艺对象的变量: <轴名称>.StatusBits.HWLimitMaxActive)

## 错误消息

错误消息	说明
已到达软限位开关	已到达或超出软限位开关。 (工艺对象的变量: <轴名称>.ErrorBits.SWLimit)
已到达硬限位开关	已到达或超出硬限位开关。 (工艺对象的变量: <轴名称>.ErrorBits.HWLimit)
无效的运动方向	命令的运动方向与组态的运动方向不符。 (工艺对象的变量: <轴名称>.ErrorBits.DirectionFault)
PTO 已在使用。	另一个轴正在使用此 PTO (Pulse Train Output) 和 HSC (High Speed Counter) 并且该轴已通过“MC_Power”启用。 (工艺对象的变量: <轴名称>.ErrorBits.HWUsed)
编码器	编码器系统中发生错误。 (工艺对象的变量: <轴名称>.ErrorBits.SensorFault)
数据交换	与所连接设备进行通信时发生错误。 (工艺对象的变量: <轴名称>.ErrorBits.CommunicationFault)
定位	在定位运动的末端，轴定位错误。 (工艺对象的变量: <轴名称>.ErrorBits.PositioningFault)
跟随误差	超出了允许的最大跟随误差。 (工艺对象的变量: <轴名称>.ErrorBits.FollowingErrorFault)

## 10.1 状态和错误位 (自工艺对象 V4 起)

错误消息	说明
编码器值无效	编码器值无效。 (工艺对象的变量: <轴名称>.StatusSensor.State)
组态错误	错误地组态了“轴”工艺对象，或者在用户程序运行期间错误地修改了可编辑的组态数据。 (工艺对象的变量: <轴名称>.ErrorBits.ConfigFault)
内部错误	发生内部错误。 (工艺对象的变量: <轴名称>.ErrorBits.SystemFault)

下方的输出窗口显示了首次报告且仍未确认的错误。

## 参见

[StatusBits 变量 V4...5 \(页 509\)](#)

[ErrorBits 变量 V4...5 \(页 514\)](#)

[诊断 - 状态和错误位 \(“轴”工艺对象 V1...3\) \(页 411\)](#)

[变量兼容表 V1...3 <-> V4...5 \(页 58\)](#)

[运动状态 \(页 225\)](#)

## 10.2 运动状态

在 TIA Portal 中使用诊断功能“运动状态”(Motion status) 可监视轴的运动状态。当轴激活时，可以在“手动控制”模式和“自动控制”模式下在线显示诊断功能。显示的状态信息的含义如下：

状态	说明
实际位置	在“实际位置”(Actual position) 框中，将指示轴的测量位置。如果轴未回原点，则该值是相对于轴启用位置的位置值。 (工艺对象的变量: <轴名称>.ActualPosition)
实际速度	在“实际速度”(Actual velocity) 框中，将指示轴的测量速度。 (工艺对象的变量: <轴名称>.ActualVelocity)
位置设定值	在“位置设定值”(Position setpoint) 框中，将指示轴的测量位置设定值。如果轴未回原点，则该值是相对于轴启用位置的位置值。 (工艺对象的变量: <轴名称>.Position)
速度设定值	在“速度设定值”(Velocity setpoint) 框中，将指示轴的测量速度设定值。 (工艺对象的变量: <轴名称>.Velocity)
目标位置	“目标位置”(Target position) 框指示活动定位命令或轴命令表的当前目标位置。“目标位置”(Target position) 的值仅在定位命令执行期间有效。 (工艺对象的变量: <轴名称>.StatusPositioning.TargetPosition)
剩余行进距离	“剩余行进距离”(Remaining travel distance) 框指示活动定位命令或轴命令表的当前剩余行进距离。“剩余行进距离”(Remaining travel distance) 值仅在定位命令执行期间有效。 (工艺对象的变量: <轴名称>.StatusPositioning.Distance)

### 参见

位置值和速度值变量 V4...5 (页 485)

StatusPositioning 变量 V4...5 (页 506)

变量兼容表 V1...3 <-> V4...5 (页 58)

位置值和速度值变量 V6 及以上版本 (页 337)

MotionStatus 变量 V1...3 (页 478)

状态和错误位 (自工艺对象 V4 起) (页 221)

## 10.3 动态设置

在 TIA Portal 中使用“动态设置”(Dynamics settings) 诊断功能可监视轴的动态限值。当轴激活时，可以在“手动控制”模式和“自动控制”模式下在线显示诊断功能。显示的状态信息的含义如下：

动态限制	说明
加速度	“加速度”(Acceleration) 框指示当前组态的轴的加速度。 (工艺对象的变量: <轴名称>.DynamicDefaults.Acceleration)
减速度	“减速度”(Deceleration) 框指示当前组态的轴的减速度。 (工艺对象的变量: <轴名称>.DynamicDefaults.Deceleration)
急停减速度	“急停减速度”(Emergency deceleration) 框指示当前组态的轴的急停减速度。 (工艺对象的变量: <轴名称>.DynamicDefaults.EmergencyDeceleration)
加加速度 (自轴工艺对象 V2 起)	“速度”(Velocity) 框指示组态的当前轴步进速度。 (工艺对象的变量: <轴名称>.DynamicDefaults.Jerk)

### 参见

DynamicDefaults 变量 V4...5 (页 497)

变量兼容表 V1...3 <-> V4...5 (页 58)

## 10.4 PROFIdrive 帧

在 TIA Portal 中，“工艺对象 > 诊断 > PROFIdrive 报文”(Technology object > Diagnostics > PROFIdrive telegram) 诊断功能用于监视驱动器和编码器返回的 PROFIdrive 报文。在线模式下将显示该诊断功能。

### “驱动器”(Drive) 区域

在此区域中，将显示驱动器返回给控制器的 PROFIdrive 报文中所包含的以下参数：

- 状态字“SW1”和“SW2”
- 输出到驱动器的速度设定值 (NSET)
- 从驱动器发送信号的实际速度 (NACT)

### “编码器”(Encoder) 区域

在此区域中，将显示编码器返回给控制器的 PROFIdrive 报文中所包含的以下参数：

- 状态字“G1\_ZSW”
- 实际位置值“G1\_XIST1”（周期性实际编码器值）
- 实际位置值“G1\_XIST2”（编码器的绝对值）

## 11.1 S7-1200 Motion Control (自 V6 起)

### 11.1.1 MC\_Power

#### 11.1.1.1 MC\_Power: 启用、禁用轴 V6 及更高版本

##### 说明

“MC\_Power”运动控制指令可启用或禁用轴。

##### 要求

- 工艺对象已正确组态。
- 没有待决的启用/禁止错误。

对于 PROFIdrive 驱动装置或模拟量驱动装置接口：

- 控制器与编码器之间已建立周期性总线通信（“<轴名称>.StatusSensor[1].CommunicationOK”= TRUE）。
- 控制器与驱动装置之间已建立周期性总线通信（“<轴名称>.StatusDrive.CommunicationOK”= TRUE）。

##### 超驰响应

运动控制命令无法中止“MC\_Power”的执行。

禁用轴（输入参数“Enable”= FALSE）之后，将根据所选“StopMode”中止相关工艺对象的所有运动控制命令。

## 参数

参数	声明	数据类型	默认值	说明	
Axis	INPUT	TO_Axis	-	轴工艺对象	
Enable	INPUT	BOOL	FALSE	TRUE	轴已启用。
				FALS E	根据组态的“StopMode”中断当前所有作业。停止并禁用轴。
StartMode	INPUT	INT	1	0	启用位置不受控的定位轴 *)
				1	启用位置受控的定位轴 *)
<p>*) 使用带 PTO (Pulse Train Output) 驱动器的定位轴时忽略该参数。</p> <p>此参数在启用定位轴时 (Enable 从 FALSE 变为 TRUE) 以及在成功确认导致轴被禁用的中断后再次启用轴时执行一次。</p>					
StopMode	INPUT	INT	0	0	紧急停止  如果禁用轴的请求处于待决状态，则轴将以组态的急停减速度进行制动。轴在变为静止状态后被禁用。
				1	立即停止  如果禁用轴的请求处于待决状态，则会输出该设定值 0，并禁用轴。轴将根据驱动器中的组态进行制动，并转入停止状态。  对于通过 PTO (Pulse Train Output) 的驱动器连接：  禁用轴时，将根据基于频率的减速度，停止脉冲输出： <ul style="list-style-type: none"><li>• 输出频率 <math>\geq 100 \text{ Hz}</math> 减速度：最长 30 ms</li><li>• 输出频率 <math>&lt; 100 \text{ Hz}</math> 减速度：30 ms；2 Hz 时，最长 1.5 s</li></ul>

参数	声明	数据类型	默认值	说明	
				2	带有加速度变化率控制的紧急停止 如果禁用轴的请求处于待决状态，则轴将以组态的急停减速度进行制动。如果激活了加速度变化率控制，会将已组态的加速度变化率考虑在内。轴在变为静止状态后被禁用。
Status	OUTPUT	BOOL	FALSE	轴的使能状态	
				FALSE	禁用轴。 轴不会执行运动控制命令也不会接受任何新命令（例外：MC_Reset 命令） 通过 PTO (Pulse Train Output) 的驱动器连接： 轴未回原点。 在禁用轴时，只有在轴停止之后，才会将状态更改为 FALSE。
				TRUE	轴已启用。 轴已就绪，可以执行运动控制命令。 在启用轴时，直到信号“驱动器准备就绪”处于待决状态之后，才会将状态更改为 TRUE。在轴组态中，如果未组态“驱动器准备就绪”驱动器接口，那么状态将会立即更改为 TRUE。
Busy	OUTPUT	BOOL	FALSE	TRUE	“MC_Power”处于活动状态。
Error	OUTPUT	BOOL	FALSE	TRUE	运动控制指令“MC_Power”或相关工艺对象发生错误。错误原因，请参见“ErrorID”和“ErrorInfo”的参数说明。
ErrorID	OUTPUT	WORD	16#0000	参数“Error”的错误 ID (页 300)	
ErrorInfo	OUTPUT	WORD	16#0000	参数“ErrorID”的错误信息 ID (页 300)	

---

#### 说明

如果因出错而停止轴，那么在清除并确认错误之后，会再次自动启用轴。这要求输入参数“Enable”的值在该过程中保持为 TRUE。

---

## 启用带有已组态驱动器接口的轴

要启用轴，请按下列步骤操作：

1. 首先检查是否满足上述要求。
2. 使用所需值初始化输入参数“StartMode”和“StopMode”。将输入参数“Enable”设置为 TRUE。

将“启用驱动器”的使能输出更改为 TRUE，以接通驱动器的电源。CPU 将等待驱动器的“驱动器就绪”信号。

当 CPU 组态完成且输入端出现“驱动器就绪”信号时，将启用轴。输出参数“Status”和工艺对象变量 <轴名称>.StatusBits.Enable 的值为 TRUE。

## 启用不带已组态驱动器接口的轴

要启用轴，请按下列步骤操作：

1. 首先检查是否满足上述要求。
2. 使用所需值初始化输入参数“StartMode”和“StopMode”。将输入参数“Enable”设置为 TRUE。轴已启用。输出参数“Status”和工艺对象变量 <轴名称>.StatusBits.Enable 的值为 TRUE。

## 禁用轴

要禁用轴，可以按照下列步骤操作：

1. 停止轴。

可以通过工艺对象变量 <轴名称>.StatusBits.StandStill 识别轴何时处于停止状态。

2. 在轴停止后，将输入参数“Enable”设置为 FALSE。
3. 如果输出参数“Busy”和“Status”以及工艺对象变量 <轴名称>.StatusBits.Enable 的值均为 FALSE，则说明禁用轴已完成。

参见

**MC\_Reset:** 确认故障，重新启动工艺对象（V6 及以上版本）(页 235)

**MC\_Home:** 使轴归位，设置参考点（V6 及以上版本）(页 238)

**MC\_MoveAbsolute:** 轴的绝对定位（V6 及以上版本）(页 249)

**MC\_MoveRelative:** 轴的相对定位（V6 及以上版本）(页 254)

**MC\_MoveVelocity:** 以设定速度移动轴（V6 及以上版本）(页 259)

**MC\_MoveJog:** 在点动模式下移动轴（V6 及以上版本）(页 265)

**MC\_CommandTable:** 按照运动顺序运行轴命令（V6 及以上版本）(页 270)

**MC\_ChangeDynamic:** 更改轴的动态设置（V6 及以上版本）(页 274)

**MC\_ReadParam:** 连续读取定位轴的运动数据（V6 及以上版本）(页 277)

**MC\_WriteParam:** 写入定位轴的变量（V6 及以上版本）(页 280)

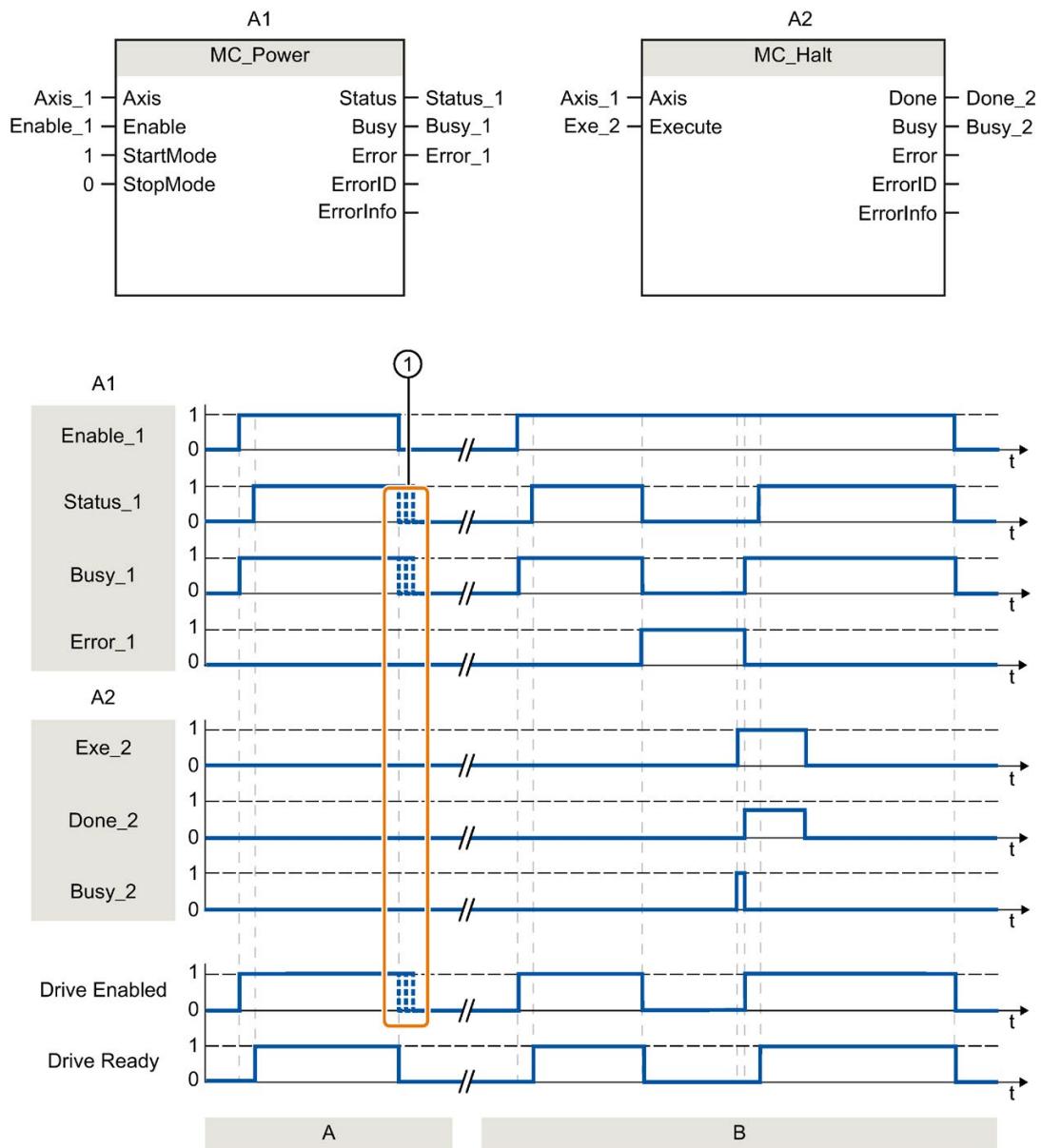
**MC\_Halt:** 停止轴（V6 及以上版本）(页 245)

组态 - 停止信号（仅限 PROFIdrive 和模拟驱动器接口）(页 129)

维修时，无需位置控制，即可移动轴(页 298)

### 11.1.1.2 MC\_Power: 功能图 (V6 及以上版本)

#### 功能图



区域 A	启用轴之后再次禁用。驱动器将“驱动器准备就绪”信号发送回 CPU 后，可以通过“Status_1”标识已成功启用轴。
区域 B	在启用轴之后，如果发生错误，则会禁用轴。在消除错误并通过“MC_Reset”确认后，然后该轴再次被启用。
①	准确的结束信号取决于所选的驱动装置和 STOP 模式。

## 11.1.2 MC\_Reset

### 11.1.2.1 MC\_Reset: 确认故障，重新启动工艺对象（V6 及以上版本）

#### 说明

运动控制指令“MC\_Reset”可用于确认“伴随轴停止出现的运行错误”和“组态错误”。关于这些需要进行确认的错误，请参见“解决方法”(Remedy) 下的“ErrorID 和 ErrorInfo 列表”。

在 RUN 模式下完成下载后，可将轴组态下载到工作存储器。

#### 要求

- 定位轴工艺对象已正确组态。
- 已经清除了引起这些需确认的待决组态错误的原因（例如，已将定位轴工艺对象中的加速度更改为有效值）。

#### 超驰响应

任何其它运动控制命令均无法中止 MC\_Reset 命令。

新的 MC\_Reset 命令不会中止任何其它激活的运动控制命令。

## 参数

参数	声明	数据类型	默认值	说明	
Axis	INPUT	TO_Axis	-	轴工艺对象	
Execute	INPUT	BOOL	FALSE	上升沿时启动命令	
Restart	INPUT	BOOL	FALSE	TRUE	将轴组态从装载存储器下载到工作存储器。 仅可在禁用轴后，才能执行该命令。 请参见有关下载到 CPU (页 186) 的说明。
				FALSE	确认待决的错误
Done	OUTPUT	BOOL	FALSE	TRUE	错误已确认。
Busy	OUTPUT	BOOL	FALSE	TRUE	命令正在执行
Error	OUTPUT	BOOL	FALSE	TRUE	执行命令期间出错。错误原因，请参见“ErrorID”和“ErrorInfo”的参数说明。
ErrorID	OUTPUT	WORD	16#0000	参数“Error”的错误 ID (页 300)	
ErrorInfo	OUTPUT	WORD	16#0000	参数“ErrorID”的错误信息 ID (页 300)	

可通过 MC\_Reset 参数，对需要进行确认的错误进行确认

要确认错误，请按照下列步骤操作：

- 首先检查是否满足上述要求。
- 从输入参数“Execute”的上升沿开始确认错误。使用 V7.0 或更高版本的 MC\_Reset，可以在启用轴之前确认已组态的编码器或驱动器上的防止接通错误。
- 如果输出参数“Done”的值为 TRUE，同时工艺对象变量 <轴名称>.StatusBits.Error 的值为 FALSE，则说明错误已被确认。

---

#### 说明

##### 使用“重新启动”= FALSE 进行确认

要仅确认错误，请设置“重新启动”= FALSE。重启期间，将不能使用该工艺对象。即使未启用或未激活轴和编码器，也会确认其中的所有错误。

---

## 参见

- MC\_Power:** 启用、禁用轴 V6 及更高版本 (页 228)
  - MC\_Home:** 使轴归位, 设置参考点 (V6 及以上版本) (页 238)
  - MC\_Halt:** 停止轴 (V6 及以上版本) (页 245)
  - MC\_MoveAbsolute:** 轴的绝对定位 (V6 及以上版本) (页 249)
  - MC\_MoveRelative:** 轴的相对定位 (V6 及以上版本) (页 254)
  - MC\_MoveVelocity:** 以设定速度移动轴 (V6 及以上版本) (页 259)
  - MC\_MoveJog:** 在点动模式下移动轴 (V6 及以上版本) (页 265)
  - MC\_CommandTable:** 按照运动顺序运行轴命令 (V6 及以上版本) (页 270)
  - MC\_ChangeDynamic:** 更改轴的动态设置 (V6 及以上版本) (页 274)
  - MC\_ReadParam:** 连续读取定位轴的运动数据 (V6 及以上版本) (页 277)
  - MC\_WriteParam:** 写入定位轴的变量 (V6 及以上版本) (页 280)
- 下载到 CPU (页 186)

## 11.1.3 MC\_Home

### 11.1.3.1 MC\_Home: 使轴归位, 设置参考点 (V6 及以上版本)

#### 说明

使用“MC\_Home”运动控制指令可将轴坐标与实际物理驱动器位置匹配。轴的绝对定位需要回原点。可执行以下类型的回原点:

- 主动回原点 (Mode = 3)

自动执行回原点步骤。

- 被动回原点 (Mode = 2)

被动回原点期间, 运动控制指令“MC\_Home”不会执行任何回原点运动。用户需通过其它运动控制指令, 执行这一步骤中所需的行进移动。检测到回原点开关时, 轴即回原点。

- 直接绝对回原点 (Mode = 0)

将当前的轴位置设置为参数“Position”的值。

- 直接相对回原点 (Mode = 1)

将当前轴位置的偏移值设置为参数“Position”的值。

- 绝对编码器相对调节 (Mode = 6)

将当前轴位置的偏移值设置为参数“Position”的值。

- 绝对编码器绝对调节 (Mode = 7)

将当前的轴位置设置为参数“Position”的值。

Mode 6 和 7 仅用于带模拟驱动接口的驱动器和 PROFIdrive 驱动器。

#### 要求

- 定位轴工艺对象已正确组态。
- 轴已启用。 (PROFIdrive 驱动装置 / 模拟量驱动装置连接 Mode = 0 或 1 时无效)
- 以 Mode = 0、1 或 2 启动时不会任何激活 MC\_CommandTable 命令。

#### 超驰响应

超驰响应取决于所选的模式:

**Mode = 0, 1, 6, 7**

任何其它运动控制命令均无法中止 MC\_Home 命令。

MC\_Home 命令不会中止任何激活的运动控制命令。按照新的回原点位置（输入参数“Position”的值）进行回原点操作后，将继续执行与位置相关的运动命令。

**Mode = 2**

可通过下列运动控制命令中止 MC\_Home 命令：

- MC\_Home 命令 Mode = 2、3

新的 MC\_Home 命令可中止下列激活的运动控制命令：

- MC\_Home 命令 Mode = 2

按照新的回原点位置（输入参数“Position”的值）进行回原点操作后，将继续执行与位置相关的运动命令。

**Mode = 3**

可通过下列运动控制命令中止 MC\_Home 命令：

- MC\_Home 命令 Mode = 3
- MC\_Halt 命令
- MC\_MoveAbsolute 命令
- MC\_MoveRelative 命令
- MC\_MoveVelocity 命令
- MC\_MoveJog 命令
- MC\_CommandTable 命令

新的 MC\_Home 命令可中止下列激活的运动控制命令：

- MC\_Home 命令 Mode = 2、3
- MC\_Halt 命令
- MC\_MoveAbsolute 命令
- MC\_MoveRelative 命令
- MC\_MoveVelocity 命令
- MC\_MoveJog 命令
- MC\_CommandTable 命令

## 参数

参数	声明	数据类型	默认值	说明
Axis	INPUT	TO_Axis	-	轴工艺对象
Execute	INPUT	BOOL	FALSE	上升沿时启动命令
Position	INPUT	REAL	0.0	<ul style="list-style-type: none"><li>• Mode = 0、2 和 3 完成回原点操作之后，轴的绝对位置</li><li>• Mode = 1 对当前轴位置的修正值 限值： <math>-1.0E12 \leq \text{Position} \leq 1.0E12</math></li></ul>

参数	声明	数据类型	默认值	说明
Mode	INPUT	INT	0	回原点模式 0 绝对式直接回原点 新的轴位置为参数“Position”位置的值。 1 相对式直接回原点 新的轴位置等于当前轴位置 + 参数“Position”位置的值。 2 被动回原点 将根据轴组态进行回原点。回原点后，将新的轴位置设置为参数“Position”的值。 如果已引用轴 <轴名>.StatusBits.HomingDone = TRUE，此状态位在附加被动回原点操作期间保持置位。 3 主动回原点 按照轴组态进行回原点操作。回原点后，将新的轴位置设置为参数“Position”的值。 6 绝对编码器调节（相对） 将当前轴位置的偏移值设置为参数“Position”的值。计算出的绝对值偏移值保持性地保存在 CPU 内。(<轴名>.StatusSensor.AbsEncoderOffset) 7 绝对编码器调节（绝对） 将当前的轴位置设置为参数“Position”的值。计算出的绝对值偏移值保持性地保存在 CPU 内。(<轴名>.StatusSensor.AbsEncoderOffset)
Done	OUTPUT	BOOL	FALSE	TRUE 命令已完成

参数	声明	数据类型	默认值	说明	
Busy	OUTPUT	BOOL	FALSE	TRUE	命令正在执行
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	命令在执行过程中被另一命令中止。
Error	OUTPUT	BOOL	FALSE	TRUE	执行命令期间出错。错误原因, 请参见“ErrorID”和“ErrorInfo”的参数说明。
ErrorID	OUTPUT	WORD	16#0000	参数“Error”的错误 ID (页 300)	
ErrorInfo	OUTPUT	WORD	16#0000	参数“ErrorID”的错误信息 ID (页 300)	
ReferenceMarkPosition	OUTPUT	REAL	0.0	显示工艺对象回原点位置 (“Done” = TRUE 时有效)	

## 复位“已回原点”状态

在以下条件下，将复位工艺对象（<轴名>.StatusBits.HomingDone）的“已回原点”状态：

- 通过 PTO (Pulse Train Output) 的驱动装置连接：

- 启动“MC\_Home”命令，进行主动回原点  
(成功完成回原点操作之后，“已回原点”状态将再次置位。)
- 通过“MC\_Power”运动控制指令，禁用轴
- 在自动模式和手动控制之间切换
- CPU 关闭 → 上电后
- CPU 重新启动后 (RUN-STOP → STOP-RUN)

- 带增量式实际值的工艺对象：

- 启动“MC\_Home”命令，进行主动回原点  
(成功完成回原点操作之后，“已回原点”状态将再次置位。)
- 编码器系统出错或编码器故障
- 重新启动工艺对象
- CPU 关闭 → 上电后
- 存储器复位
- 修改编码器组态

- 带绝对实际值的工艺对象：

- 传感器系统出错/编码器故障
- 更换 CPU
- 修改编码器组态
- 将 CPU 恢复为出厂设置
- 将其它项目传送到控制器

## 使轴回原点

要使轴回原点, 请按下列步骤操作:

1. 首先检查是否满足上述要求。
2. 使用这些值提供所需的输入参数, 然后在输入参数“Execute”的上升沿开始回原点。
3. 如果输出参数“Done”和工艺对象变量 <轴名称>.StatusBits.HomingDone 的值为 TRUE, 则说明回原点已完成。参考位置可根据变量 <轴名称>.ReferenceMarkPosition。

## 参见

**MC\_Power:** 启用、禁用轴 V6 及更高版本 (页 228)

**MC\_Reset:** 确认故障, 重新启动工艺对象 (V6 及以上版本) (页 235)

**MC\_Halt:** 停止轴 (V6 及以上版本) (页 245)

**MC\_MoveAbsolute:** 轴的绝对定位 (V6 及以上版本) (页 249)

**MC\_MoveRelative:** 轴的相对定位 (V6 及以上版本) (页 254)

**MC\_MoveVelocity:** 以设定速度移动轴 (V6 及以上版本) (页 259)

**MC\_MoveJog:** 在点动模式下移动轴 (V6 及以上版本) (页 265)

**MC\_CommandTable:** 按照运动顺序运行轴命令 (V6 及以上版本) (页 270)

**MC\_ChangeDynamic:** 更改轴的动态设置 (V6 及以上版本) (页 274)

**MC\_ReadParam:** 连续读取定位轴的运动数据 (V6 及以上版本) (页 277)

**MC\_WriteParam:** 写入定位轴的变量 (V6 及以上版本) (页 280)

## 11.1.4 MC\_Halt

### 11.1.4.1 MC\_Halt: 停止轴 (V6 及以上版本)

#### 说明

通过运动控制指令“MC\_Halt”，可停止所有运动并以组态的减速度停止轴。未定义停止位置。

#### 要求

- 定位轴工艺对象已正确组态。
- 轴已启用。

#### 超驰响应

可通过下列运动控制命令中止 MC\_Halt 命令：

- MC\_Home 命令 Mode = 3
- MC\_Halt 命令
- MC\_MoveAbsolute 命令
- MC\_MoveRelative 命令
- MC\_MoveVelocity 命令
- MC\_MoveJog 命令
- MC\_CommandTable 命令

新的 MC\_Halt 命令可中止下列激活的运动控制命令：

- MC\_Home 命令 Mode = 3
- MC\_Halt 命令
- MC\_MoveAbsolute 命令
- MC\_MoveRelative 命令
- MC\_MoveVelocity 命令
- MC\_MoveJog 命令
- MC\_CommandTable 命令

## 参数

参数	声明	数据类型	默认值	说明	
Axis	INPUT	TO_SpeedAxi s	-	轴工艺对象	
Execute	INPUT	BOOL	FALSE	上升沿时启动命令	
Done	OUTPUT	BOOL	FALSE	TRUE	速度达到零
Busy	OUTPUT	BOOL	FALSE	TRUE	正在执行命令。
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	命令在执行过程中被另一命令中止。
Error	OUTPUT	BOOL	FALSE	TRUE	执行命令期间出错。错误原因, 请参见“ErrorID”和“ErrorInfo”的参数说明。
ErrorID	OUTPUT	WORD	16#0000	参数“Error”的错误 ID (页 300)	
ErrorInfo	OUTPUT	WORD	16#0000	参数“ErrorID”的错误信息 ID (页 300)	

## 参见

[MC\\_Power](#): 启用、禁用轴 V6 及更高版本 (页 228)

[MC\\_Reset](#): 确认故障, 重新启动工艺对象 (V6 及以上版本) (页 235)

[MC\\_Home](#): 使轴归位, 设置参考点 (V6 及以上版本) (页 238)

[MC\\_MoveAbsolute](#): 轴的绝对定位 (V6 及以上版本) (页 249)

[MC\\_MoveRelative](#): 轴的相对定位 (V6 及以上版本) (页 254)

[MC\\_MoveVelocity](#): 以设定速度移动轴 (V6 及以上版本) (页 259)

[MC\\_MoveJog](#): 在点动模式下移动轴 (V6 及以上版本) (页 265)

[MC\\_CommandTable](#): 按照运动顺序运行轴命令 (V6 及以上版本) (页 270)

[MC\\_ChangeDynamic](#): 更改轴的动态设置 (V6 及以上版本) (页 274)

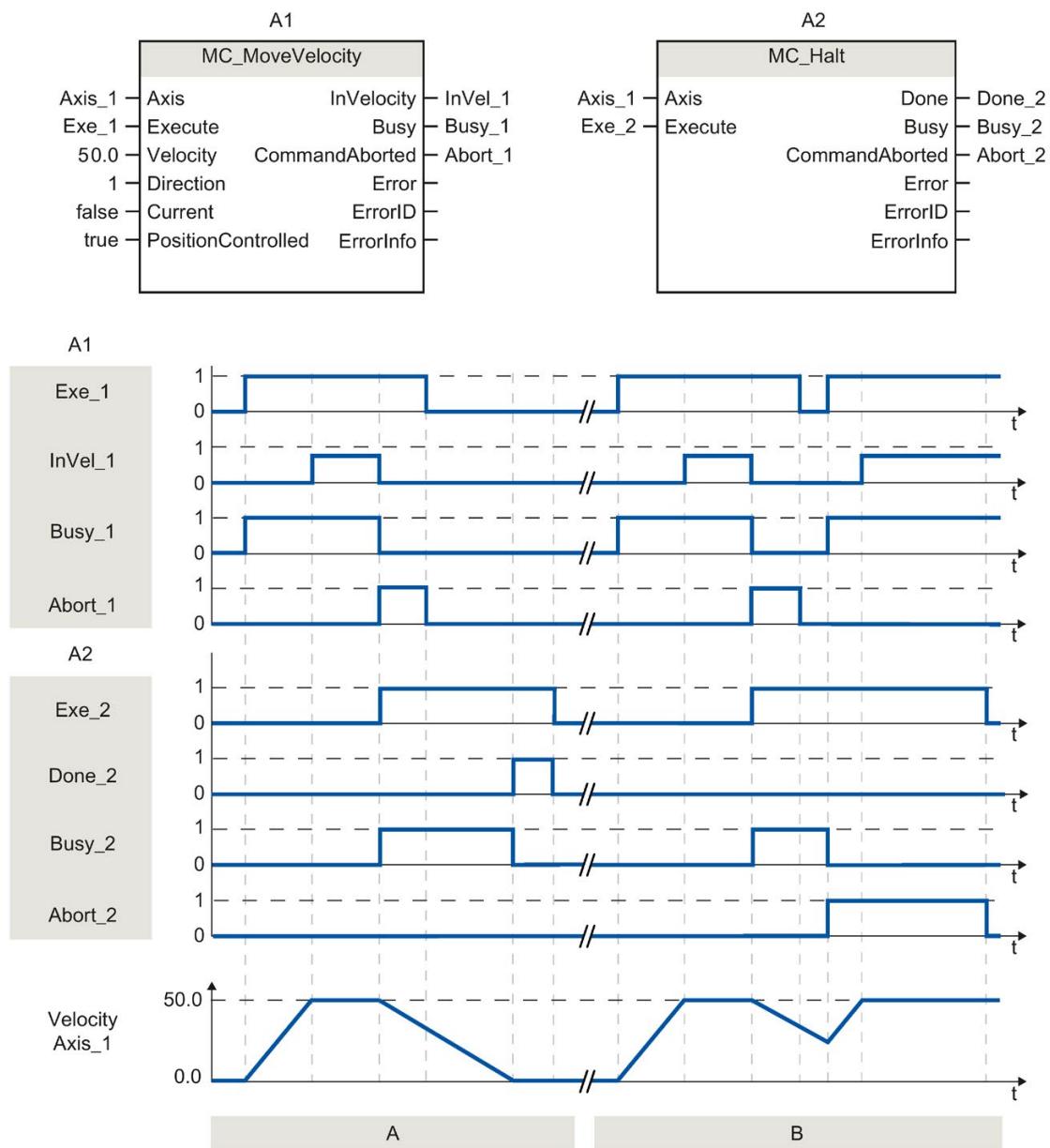
[MC\\_ReadParam](#): 连续读取定位轴的运动数据 (V6 及以上版本) (页 277)

[MC\\_WriteParam](#): 写入定位轴的变量 (V6 及以上版本) (页 280)

维修时, 无需位置控制, 即可移动轴 (页 298)

### 11.1.4.2 MC\_Halt: 功能图 (V6 及以上版本)

#### 功能图



在组态窗口“动态 > 常规”(Dynamics > General) 中，对下列值进行组态：

- 加速度：10.0
- 减速度：5.0

区域 A	轴将由 MC_Halt 命令制动，直至停止为止。通过“Done_2”发出轴停止的信号。
区域 B	当 MC_Halt 命令对轴进行制动处理时，另一个运动命令会中止该命令。将通过“Abort_2”发出中止信号。

## 11.1.5 MC\_MoveAbsolute

### 11.1.5.1 MC\_MoveAbsolute: 轴的绝对定位 (V6 及以上版本)

#### 说明

运动控制指令“MC\_MoveAbsolute”启动轴定位运动，以将轴移动到某个绝对位置。

#### 要求

- 定位轴工艺对象已正确组态。
- 轴已启用。
- 轴已回原点。

#### 超驰响应

可通过下列运动控制命令中止 MC\_MoveAbsolute 命令：

- MC\_Home 命令 Mode = 3
- MC\_Halt 命令
- MC\_MoveAbsolute 命令
- MC\_MoveRelative 命令
- MC\_MoveVelocity 命令
- MC\_MoveJog 命令
- MC\_CommandTable 命令

新的 MC\_MoveAbsolute 命令可中止下列激活的运动控制命令：

- MC\_Home 命令 Mode = 3
- MC\_Halt 命令
- MC\_MoveAbsolute 命令
- MC\_MoveRelative 命令
- MC\_MoveVelocity 命令
- MC\_MoveJog 命令
- MC\_CommandTable 命令

## 参数

参数	声明	数据类型	默认值	说明
Axis	INPUT	TO_PositioningAxis	-	轴工艺对象
Execute	INPUT	BOOL	FALSE	上升沿时启动命令
Position	INPUT	REAL	0.0	绝对目标位置 限值: -1.0E12 ≤ Position ≤ 1.0E12
Velocity	INPUT	REAL	10.0	轴的速度 由于所组态的加速度和减速度以及待接近的目标位置等原因，不会始保持这一速度。 限值： 启动/停止速度 ≤ Velocity ≤ 最大速度
Direction	INPUT	INT	1	轴的运动方向 仅在“模数”已启用的情况下才评估。“工艺对象 > 组态 > 扩展参数 > 模数 > 启用模数 (Technology object > Configuration > Extended parameters > Modulo > Enable Modulo) 对于 PTO 轴忽略该参数。
			0	速度的符号 (“Velocity”参数) 用于确定运动的方向。
			1	正方向 (从正方向逼近目标位置)
			2	负方向 (从负方向逼近目标位置)
			3	最短距离 (工艺将选择从当前位置开始，到目标位置的最短距离)
Done	OUTPUT	BOOL	FALSE	TRUE 达到绝对目标位置
Busy	OUTPUT	BOOL	FALSE	TRUE 命令正在执行

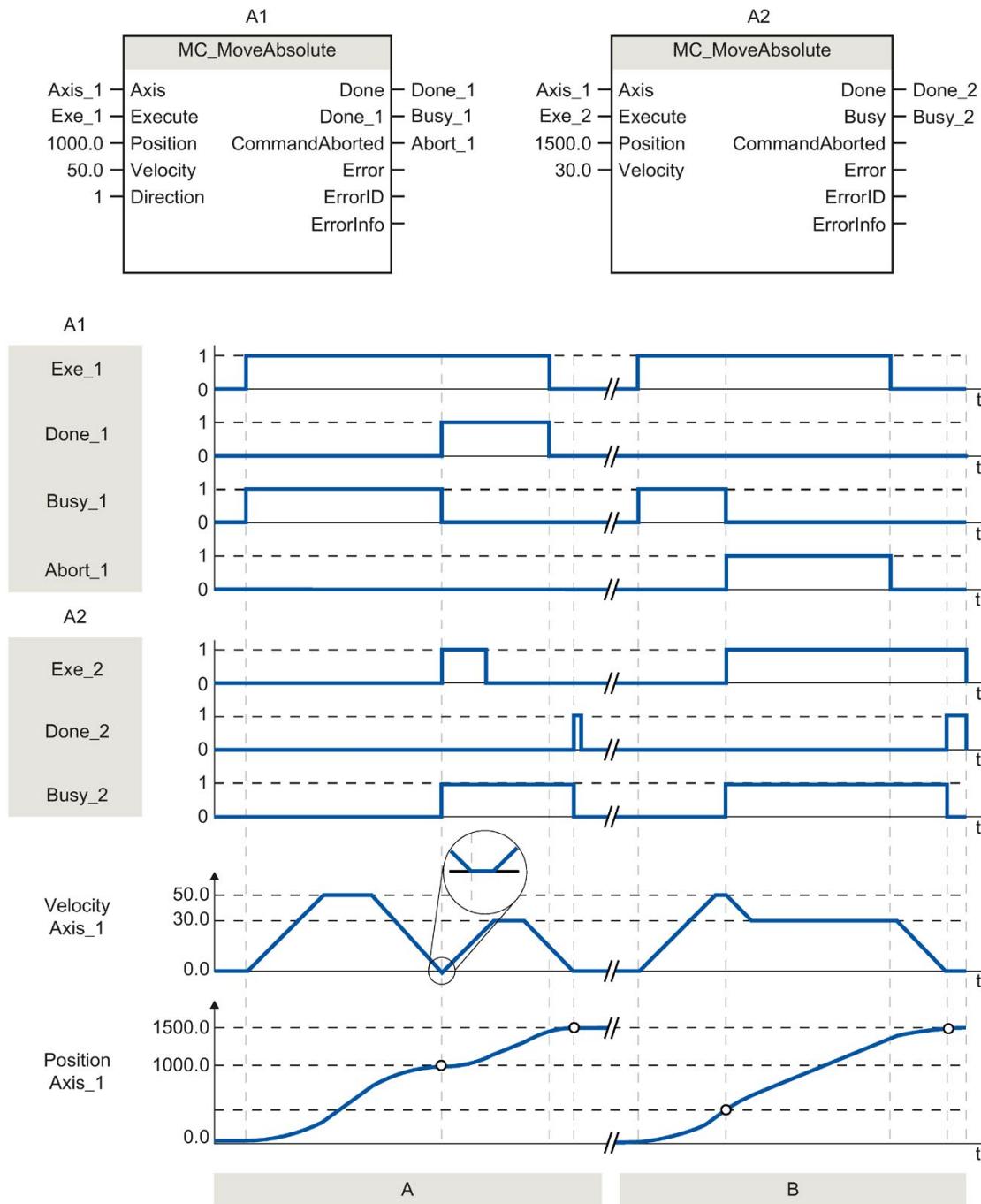
参数	声明	数据类型	默认值	说明	
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	命令在执行过程中被另一命令中止。
Error	OUTPUT	BOOL	FALSE	TRUE	执行命令期间出错。错误原因, 请参见“ErrorID”和“ErrorInfo”的参数说明。
ErrorID	OUTPUT	WORD	16#0000	参数“Error”的错误 ID (页 300)	
ErrorInfo	OUTPUT	WORD	16#0000	参数“ErrorID”的错误信息 ID (页 300)	

## 参见

- MC\_Power:** 启用、禁用轴 V6 及更高版本 (页 228)
- MC\_Reset:** 确认故障, 重新启动工艺对象 (V6 及以上版本) (页 235)
- MC\_Home:** 使轴归位, 设置参考点 (V6 及以上版本) (页 238)
- MC\_Halt:** 停止轴 (V6 及以上版本) (页 245)
- MC\_MoveRelative:** 轴的相对定位 (V6 及以上版本) (页 254)
- MC\_MoveVelocity:** 以设定速度移动轴 (V6 及以上版本) (页 259)
- MC\_MoveJog:** 在点动模式下移动轴 (V6 及以上版本) (页 265)
- MC\_CommandTable:** 按照运动顺序运行轴命令 (V6 及以上版本) (页 270)
- MC\_ChangeDynamic:** 更改轴的动态设置 (V6 及以上版本) (页 274)
- MC\_ReadParam:** 连续读取定位轴的运动数据 (V6 及以上版本) (页 277)
- MC\_WriteParam:** 写入定位轴的变量 (V6 及以上版本) (页 280)
- 组态 - 模数 (仅限 PROFIdrive/模拟驱动器接口) (页 100)

## 11.1.5.2 MC\_MoveAbsolute: 功能图 (V6 及以上版本)

## 功能图



在组态窗口“**动态 > 常规**(Dynamics > General) 中，对下列值进行组态：

- 加速度：10.0
- 减速度：10.0

区域 A	通过 MC_MoveAbsolute 命令，可将轴移动到绝对位置 1000.0。当轴达到目标位置后，将通过“Done_1”发出信号。当“Done_1”= TRUE 时，将启动另一个目标位置为 1500.0 的 MC_MoveAbsolute 命令。由于响应时间的原因（如用户程序的循环时间等），轴将暂时停止（参见放大后的详细信息）。当轴达到新的目标位置后，将通过“Done_2”发出信号。
区域 B	可通过另一个 MC_MoveAbsolute 命令中止激活的 MC_MoveAbsolute 命令。将通过“Abort_1”发出中止信号。之后，轴将会以新的速度移动到新的目标位置 1500.0。到达新的目标位置之后，将通过“Done_2”发出信号。

## 11.1.6 MC\_MoveRelative

### 11.1.6.1 MC\_MoveRelative: 轴的相对定位 (V6 及以上版本)

#### 说明

通过运动控制指令“MC\_MoveRelative”，启动相对于起始位置的定位运动。

#### 要求

- 定位轴工艺对象已正确组态。
- 轴已启用。

#### 超驰响应

可通过下列运动控制命令中止 MC\_MoveRelative 命令：

- MC\_Home 命令 Mode = 3
- MC\_Halt 命令
- MC\_MoveAbsolute 命令
- MC\_MoveRelative 命令
- MC\_MoveVelocity 命令
- MC\_MoveJog 命令
- MC\_CommandTable 命令

新的 MC\_MoveRelative 命令可中止下列激活的运动控制命令：

- MC\_Home 命令 Mode = 3
- MC\_Halt 命令
- MC\_MoveAbsolute 命令
- MC\_MoveRelative 命令
- MC\_MoveVelocity 命令
- MC\_MoveJog 命令
- MC\_CommandTable 命令

## 参数

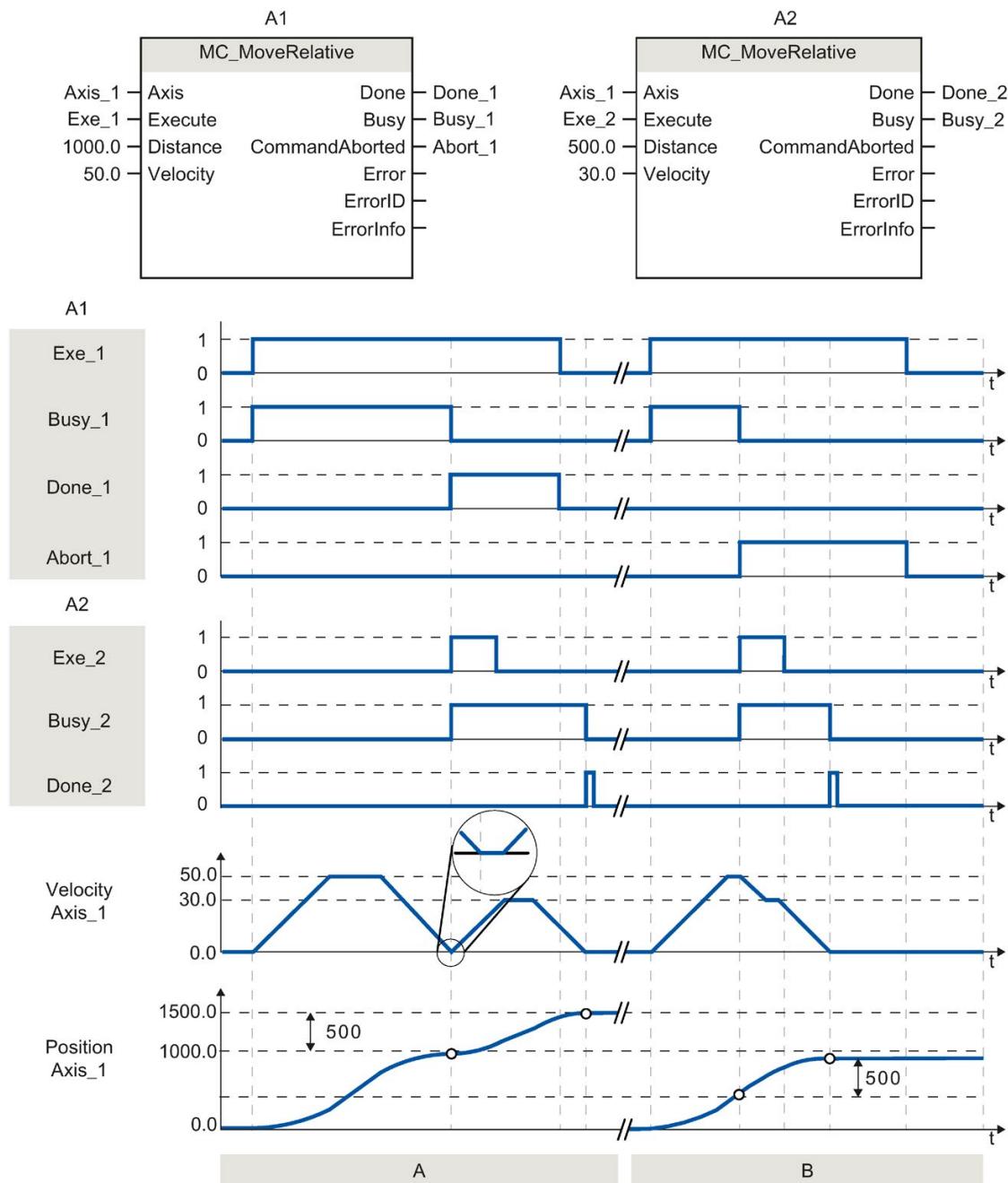
参数	声明	数据类型	默认值	说明	
Axis	INPUT	TO_PositioningAxis	-	轴工艺对象	
Execute	INPUT	BOOL	FALSE	上升沿时启动命令	
Distance	INPUT	REAL	0.0	定位操作的移动距离 限值： $-1.0E12 \leq Distance \leq 1.0E12$	
Velocity	INPUT	REAL	10.0	轴的速度 由于所组态的加速度和减速度以及要途经的距离等原因，不会始终保持这一速度。 限值： 启动/停止速度 $\leq Velocity \leq$ 最大速度	
Done	OUTPUT	BOOL	FALSE	TRUE	目标位置已到达
Busy	OUTPUT	BOOL	FALSE	TRUE	命令正在执行
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	命令在执行过程中被另一命令中止。
Error	OUTPUT	BOOL	FALSE	TRUE	执行命令期间出错。错误原因，请参见“ErrorID”和“ErrorInfo”的参数说明。
ErrorID	OUTPUT	WORD	16#0000	参数“Error”的错误 ID (页 300)	
ErrorInfo	OUTPUT	WORD	16#0000	参数“ErrorID”的错误信息 ID (页 300)	

参见

- MC\_Power:** 启用、禁用轴 V6 及更高版本 (页 228)
- MC\_Reset:** 确认故障, 重新启动工艺对象 (V6 及以上版本) (页 235)
- MC\_Home:** 使轴归位, 设置参考点 (V6 及以上版本) (页 238)
- MC\_Halt:** 停止轴 (V6 及以上版本) (页 245)
- MC\_MoveAbsolute:** 轴的绝对定位 (V6 及以上版本) (页 249)
- MC\_MoveVelocity:** 以设定速度移动轴 (V6 及以上版本) (页 259)
- MC\_MoveJog:** 在点动模式下移动轴 (V6 及以上版本) (页 265)
- MC\_CommandTable:** 按照运动顺序运行轴命令 (V6 及以上版本) (页 270)
- MC\_ChangeDynamic:** 更改轴的动态设置 (V6 及以上版本) (页 274)
- MC\_ReadParam:** 连续读取定位轴的运动数据 (V6 及以上版本) (页 277)
- MC\_WriteParam:** 写入定位轴的变量 (V6 及以上版本) (页 280)

### 11.1.6.2 MC\_MoveRelative: 功能图 (V6 及以上版本)

#### 功能图



在组态窗口“动态 > 常规”(Dynamics > General) 中，对下列值进行组态：

- 加速度：10.0
- 减速度：10.0

区域 A	轴将通过 MC_MoveRelative 命令移动 1000.0 的距离 (“Distance”)。当轴达到目标位置后，将通过“Done_1”发出信号。“Done_1”= TRUE 时，将启动另一个行进距离为 500.0 的 MC_MoveRelative 命令。由于响应时间的原因（如用户程序的循环时间等），轴将暂时停止（参见放大后的详细信息）。当轴达到新的目标位置后，将通过“Done_2”发出信号。
区域 B	可通过另一个 MC_MoveRelative 命令中止激活的 MC_MoveRelative 命令。将通过“Abort_1”发出中止信号。之后，轴将以新的速度移动 500.0 的距离 (“Distance”)。到达新的目标位置之后，将通过“Done_2”发出信号。

## 11.1.7 MC\_MoveVelocity

### 11.1.7.1 MC\_MoveVelocity: 以设定速度移动轴 (V6 及以上版本)

#### 说明

通过运动控制指令“MC\_MoveVelocity”，根据指定的速度连续移动轴。

#### 要求

- 定位轴工艺对象已正确组态。
- 轴已启用。

#### 超驰响应

MC\_MoveVelocity 可由下列运动控制命令中止：

- MC\_Home 命令 Mode = 3
- MC\_Halt 命令
- MC\_MoveAbsolute 命令
- MC\_MoveRelative 命令
- MC\_MoveVelocity 命令
- MC\_MoveJog 命令
- MC\_CommandTable 命令

新的 MC\_MoveVelocity 命令可中止下列激活的运动控制命令：

- MC\_Home 命令 Mode = 3
- MC\_Halt 命令
- MC\_MoveAbsolute 命令
- MC\_MoveRelative 命令
- MC\_MoveVelocity 命令
- MC\_MoveJog 命令
- MC\_CommandTable 命令

## 参数

参数	声明	数据类型	默认值	说明
Axis	INPUT	TO_SpeedAxi s	-	轴工艺对象
Execute	INPUT	BOOL	FALSE	上升沿时启动命令
Velocity	INPUT	REAL	10.0	轴运动的指定速度 限值： 启动/停止速度 $\leq  Velocity  \leq$ 最大速度 (允许“Velocity”= 0.0)
Direction	INPUT	INT	0	指定方向 0 旋转方向取决于参数“Velocity”值的符号 1 正旋转方向 (将忽略参数“Velocity”值的符号) 2 负旋转方向 (将忽略参数“Velocity”值的符号)
Current	INPUT	BOOL	FALSE	保持当前速度 FALSE “保持当前速度”已禁用。将使用参数“Velocity”和“Direction”的值。 TRUE “保持当前速度”已启用。而不考虑参数“Velocity”和“Direction”的值。 当轴继续以当前速度运动时，参数“InVelocity”返回值 TRUE。
PositionControlled	INPUT	BOOL	TRUE	FALSE 非位置控制操作 TRUE 位置控制操作 只要执行指令 MC_MoveVelocity 即应用该参数。之后，MC_Power 的设置再次适用。 使用 PTO 轴时忽略该参数。

参数	声明	数据类型	默认值	说明	
InVelocity	OUTPUT	BOOL	FALSE	TRUE	<ul style="list-style-type: none"> <li>"Current" = FALSE: 达到参数"Velocity"中指定的速度。</li> <li>"Current" = TRUE: 轴在启动时, 以当前速度进行移动。</li> </ul>
Busy	OUTPUT	BOOL	FALSE	TRUE	命令正在执行
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	命令在执行过程中被另一命令中止。
Error	OUTPUT	BOOL	FALSE	TRUE	执行命令期间出错。错误原因, 请参见"ErrorID"和"ErrorInfo"的参数说明。
ErrorID	OUTPUT	WORD	16#0000	参数"Error"的错误 ID (页 300)	
ErrorInfo	OUTPUT	WORD	16#0000	参数"ErrorID"的错误信息 ID (页 300)	

## 说明

### PLCopen 版本 2.0

自 V4 起, 运动控制指令“MC\_MoveVelocity”兼容 PLCopen 版本 2.0。

在命令被覆盖或因错误停止前, 无论“Execute”参数如何, “InVelocity”和“Busy”参数都会显示其各自的状态。有关更多信息, 请参见“跟踪激活命令 (页 204)”部分。

## 速度设定值为零 (Velocity = 0.0) 时的行为

“Velocity”= 0.0 的 MC\_MoveVelocity 命令 (如 MC\_Halt 命令) 可中止激活的运动命令, 并使用所组态的减速度停止轴运动。

当轴停止时, 至少在一个程序循环内将输出参数“InVelocity”的值指定为 TRUE。

“Busy”的值在减速过程中为 TRUE, 并且随“InVelocity”一起变为 FALSE。如果将参数“Execute”设置为 TRUE, 则“InVelocity”和“Busy”处于锁定状态。

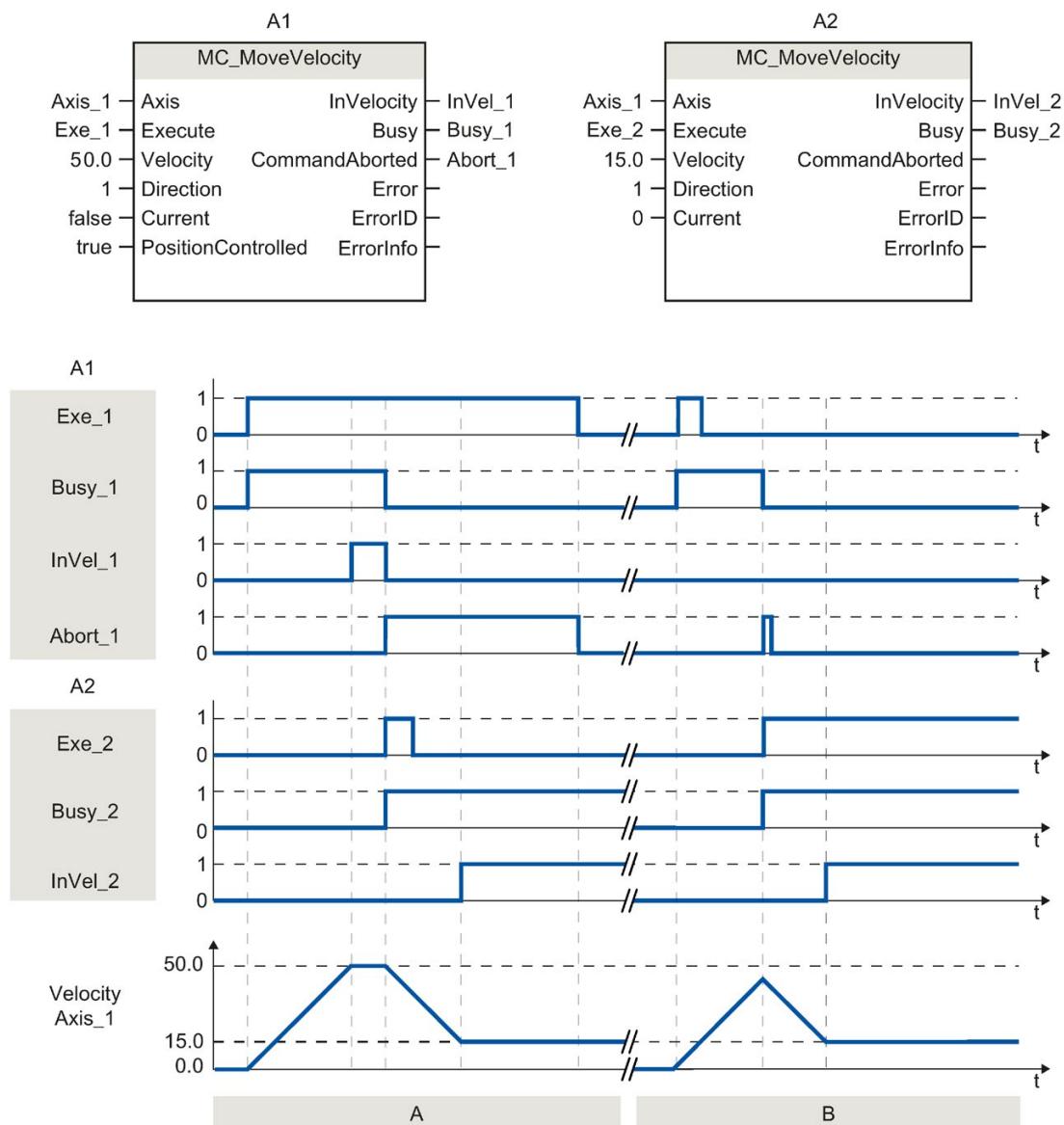
启动“MC\_MoveVelocity”命令时, 将置位工艺对象的状态位“SpeedCommand”。在轴停止时, 设置状态位“ConstantVelocity”。在启动新的运动命令后, 将对这两个位进行调整, 以适应新的情况。

参见

- MC\_Power:** 启用、禁用轴 V6 及更高版本 (页 228)
- MC\_Reset:** 确认故障, 重新启动工艺对象 (V6 及以上版本) (页 235)
- MC\_Home:** 使轴归位, 设置参考点 (V6 及以上版本) (页 238)
- MC\_Halt:** 停止轴 (V6 及以上版本) (页 245)
- MC\_MoveAbsolute:** 轴的绝对定位 (V6 及以上版本) (页 249)
- MC\_MoveRelative:** 轴的相对定位 (V6 及以上版本) (页 254)
- MC\_MoveJog:** 在点动模式下移动轴 (V6 及以上版本) (页 265)
- MC\_CommandTable:** 按照运动顺序运行轴命令 (V6 及以上版本) (页 270)
- MC\_ChangeDynamic:** 更改轴的动态设置 (V6 及以上版本) (页 274)
- MC\_ReadParam:** 连续读取定位轴的运动数据 (V6 及以上版本) (页 277)
- MC\_WriteParam:** 写入定位轴的变量 (V6 及以上版本) (页 280)

### 11.1.7.2 MC\_MoveVelocity: 功能图 (V6 及以上版本)

#### 功能图



在组态窗口“动态 > 常规”(Dynamics > General) 中，对下列值进行组态：

- 加速度：10.0
- 减速度：10.0

区域 A	激活的 MC_MoveVelocity 命令通过“InVel_1”发出信号，说明已达到目标速度。该命令随后将由另一个 MC_MoveVelocity 命令中止。将通过“Abort_1”发出中止信号。达到新的目标速度 15.0 之后，将通过“InVel_2”发出信号。之后，轴将以新的恒定速度继续移动。
区域 B	在达到目标速度之前，当前 MC_MoveVelocity 命令将由另一个 MC_MoveVelocity 命令中止。将通过“Abort_1”发出中止信号。达到新的目标速度 15.0 之后，将通过“InVel_2”发出信号。之后，轴将以新的恒定速度继续移动。

## 11.1.8 MC\_MoveJog

### 11.1.8.1 MC\_MoveJog: 在点动模式下移动轴 (V6 及以上版本)

#### 说明

通过运动控制指令“MC\_MoveJog”，在点动模式下以指定的速度连续移动轴。例如，可以使用该运动控制指令进行测试和调试。

#### 要求

- 定位轴工艺对象已正确组态。
- 轴已启用。

#### 超驰响应

可通过下列运动控制命令中止 MC\_MoveJog 命令：

- MC\_Home 命令 Mode = 3
- MC\_Halt 命令
- MC\_MoveAbsolute 命令
- MC\_MoveRelative 命令
- MC\_MoveVelocity 命令
- MC\_MoveJog 命令
- MC\_CommandTable 命令

新的 MC\_MoveJog 命令可中止下列激活的运动控制命令：

- MC\_Home 命令 Mode = 3
- MC\_Halt 命令
- MC\_MoveAbsolute 命令
- MC\_MoveRelative 命令
- MC\_MoveVelocity 命令
- MC\_MoveJog 命令
- MC\_CommandTable 命令

## 参数

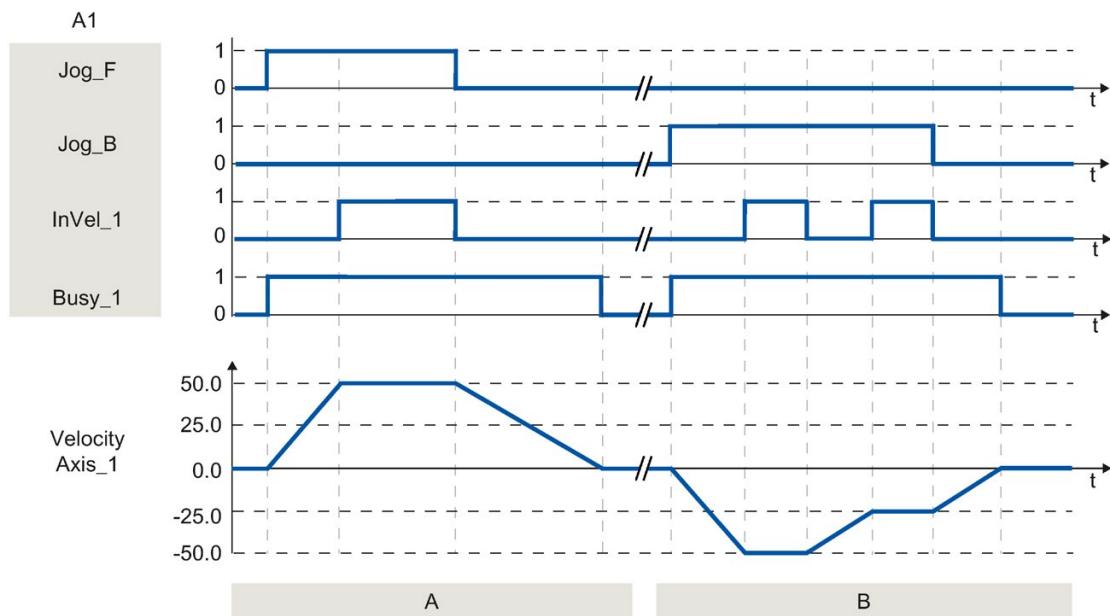
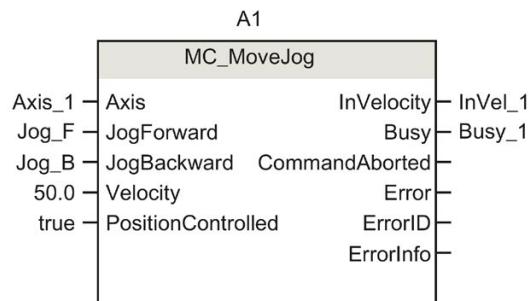
参数	声明	数据类型	默认值	说明	
Axis	INPUT	TO_SpeedAxi s	-	轴工艺对象	
JogForward	INPUT	BOOL	FALSE	如果参数值为 TRUE，则轴都将按参数“Velocity”中所指定的速度，正向移动。	
JogBackward	INPUT	BOOL	FALSE	如果参数值为 TRUE，则轴都将按参数“Velocity”中指定的速度，反向移动。	
如果两个参数同时为 TRUE，轴将根据所组态的减速度直至停止。通过参数“Error”、“ErrorID”和“ErrorInfo”，指出了错误。					
Velocity	INPUT	REAL	10.0	点动模式的预设速度 限值： 启动/停止速度 ≤ 速度 ≤ 最大速度	
PositionControlled	INPUT	BOOL	TRUE	FALSE	非位置控制操作
				TRUE	位置控制操作
只要执行指令 MC_MoveJog 即应用该参数。 之后，MC_Power 的设置再次适用。 使用 PTO 轴时忽略该参数。					
InVelocity	OUTPUT	BOOL	FALSE	TRUE	达到参数“Velocity”中指定的速度。
Busy	OUTPUT	BOOL	FALSE	TRUE	命令正在执行
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	命令在执行过程中被另一命令中止。
Error	OUTPUT	BOOL	FALSE	TRUE	执行命令期间出错。错误原因，请参见“ErrorID”和“ErrorInfo”的参数说明。
ErrorID	OUTPUT	WORD	16#0000	参数“Error”的错误 ID (页 300)	
ErrorInfo	OUTPUT	WORD	16#0000	参数“ErrorID”的错误信息 ID (页 300)	

## 参见

- MC\_Power:** 启用、禁用轴 V6 及更高版本 (页 228)
- MC\_Reset:** 确认故障，重新启动工艺对象 (V6 及以上版本) (页 235)
- MC\_Home:** 使轴归位，设置参考点 (V6 及以上版本) (页 238)
- MC\_Halt:** 停止轴 (V6 及以上版本) (页 245)
- MC\_MoveAbsolute:** 轴的绝对定位 (V6 及以上版本) (页 249)
- MC\_MoveRelative:** 轴的相对定位 (V6 及以上版本) (页 254)
- MC\_MoveVelocity:** 以设定速度移动轴 (V6 及以上版本) (页 259)
- MC\_CommandTable:** 按照运动顺序运行轴命令 (V6 及以上版本) (页 270)
- MC\_ChangeDynamic:** 更改轴的动态设置 (V6 及以上版本) (页 274)
- MC\_ReadParam:** 连续读取定位轴的运动数据 (V6 及以上版本) (页 277)
- MC\_WriteParam:** 写入定位轴的变量 (V6 及以上版本) (页 280)

## 11.1.8.2 MC\_MoveJog: 功能图 (V6 及以上版本)

## 功能图



在组态窗口“动态 > 常规”(Dynamics > General) 中，对下列值进行组态：

- 加速度：10.0
- 减速度：5.0

区域 A	在点动模式下，轴通过“Jog_F”正向移动。达到目标速度 50.0 之后，将通过“InVel_1”发出信号。复位“Jog_F”之后，将制动轴并逐步停止。
区域 B	在点动模式下，轴通过“Jog_B”反向移动。达到目标速度 -50.0 之后，将通过“InVel_1”发出信号。 “Jog_B”置位后，参数“Velocity”的值会更改为 25.0。“InVel_1”复位，轴制动。达到新的目标速度 -25.0 之后，将通过“InVel_1”发出信号。复位“Jog_B”之后，将制动轴并逐步停止。

## 11.1.9 MC\_CommandTable

### 11.1.9.1 MC\_CommandTable: 按照运动顺序运行轴命令 (V6 及以上版本)

#### 说明

运动控制指令“MC\_CommandTable”可将多个单独的轴控制命令组合到一个运动顺序中。“MC\_CommandTable”适用于通过 PTO (Pulse Train Output) 连接驱动器的轴。

#### 要求

- 定位轴工艺对象已插入且已正确组态。
- 驱动器通过 PTO (Pulse Train Output) 进行连接。
- 命令表工艺对象已插入并且组态正确。
- 轴已启用。

## 超驰响应

可通过下列运动控制命令中止 MC\_CommandTable 命令：

- MC\_Home 命令 Mode = 3
- MC\_Halt 命令
- MC\_MoveAbsolute 命令
- MC\_MoveRelative 命令
- MC\_MoveVelocity 命令
- MC\_MoveJog 命令
- MC\_CommandTable 命令

新的 MC\_CommandTable 命令可中止下列激活的运动控制命令：

- MC\_Home 命令 Mode = 3
- MC\_Halt 命令
- MC\_MoveAbsolute 命令
- MC\_MoveRelative 命令
- MC\_MoveVelocity 命令
- MC\_MoveJog 命令
- MC\_CommandTable 命令

当启动第一个“Positioning Relative”、“Positioning Absolute”、“Velocity set point”或“Halt”命令时，将取消激活的运动控制命令。

## 参数

参数	声明	数据类型	默认值	说明	
Axis	INPUT	TO_SpeedAxi s	-	轴工艺对象	
CommandT able	INPUT	TO_CommandTable	-	命令表工艺对象	
Execute	INPUT	BOOL	FALSE	命令表在上升沿时启动	
StartStep	INPUT	INT	1	定义命令表应开始执行的步 限值: $1 \leq \text{StartStep} \leq \text{EndStep}$	
EndStep	INPUT	INT	32	定义命令表应结束执行的步 限值: $\text{StartStep} \leq \text{EndStep} \leq 32$	
Done	OUTPUT	BOOL	FALSE	TRUE	已成功执行命令表
Busy	OUTPUT	BOOL	FALSE	TRUE	正在执行命令表
CommandAb orted	OUTPUT	BOOL	FALSE	TRUE	已通过另一个命令取消命令表。
Error	OUTPUT	BOOL	FALSE	TRUE	执行命令表期间出错。错误原因, 请参见 “ErrorID”和“ErrorInfo”的参数说明。
ErrorID	OUTPUT	WORD	16#0000	参数“Error”的错误 ID (页 300)	
ErrorInfo	OUTPUT	WORD	16#0000	参数“ErrorID”的错误信息 ID (页 300)	
CurrentStep	OUTPUT	INT	0	当前正在执行的命令表中的步	
StepCode	OUTPUT	WORD	16#0000	当前正在执行的步的用户定义的数值/位模式	

## 参见

**MC\_Power:** 启用、禁用轴 V6 及更高版本 (页 228)

**MC\_Reset:** 确认故障，重新启动工艺对象 (V6 及以上版本) (页 235)

**MC\_Home:** 使轴归位，设置参考点 (V6 及以上版本) (页 238)

**MC\_Halt:** 停止轴 (V6 及以上版本) (页 245)

**MC\_MoveAbsolute:** 轴的绝对定位 (V6 及以上版本) (页 249)

**MC\_MoveRelative:** 轴的相对定位 (V6 及以上版本) (页 254)

**MC\_MoveVelocity:** 以设定速度移动轴 (V6 及以上版本) (页 259)

**MC\_MoveJog:** 在点动模式下移动轴 (V6 及以上版本) (页 265)

**MC\_ChangeDynamic:** 更改轴的动态设置 (V6 及以上版本) (页 274)

**MC\_ReadParam:** 连续读取定位轴的运动数据 (V6 及以上版本) (页 277)

**MC\_WriteParam:** 写入定位轴的变量 (V6 及以上版本) (页 280)

## 11.1.10 MC\_ChangeDynamic

### 11.1.10.1 MC\_ChangeDynamic: 更改轴的动态设置 (V6 及以上版本)

#### 说明

使用运动控制指令“MC\_ChangeDynamic”可以更改轴的下列设置：

- 更改加速时间（加速度）值
- 更改减速时间（减速度）值
- 更改急停减速时间（急停减速度）值
- 更改平滑时间（冲击）值

有关上述更改产生的影响，请参见变量 (页 335) 的说明。

#### 要求

定位轴工艺对象已正确组态。

#### 超驰响应

任何其它运动控制命令均无法中止 MC\_ChangeDynamic 命令。

新的 MC\_ChangeDynamic 命令不会中止任何激活的运动控制命令。

## 参数

参数	声明	数据类型	默认值	说明	
Axis	INPUT	TO_SpeedAxes	-	轴工艺对象	
Execute	INPUT	BOOL	FALSE	上升沿时启动命令	
ChangeRampUp	INPUT	BOOL	FALSE	TRUE	按照输入参数“RampUpTime”更改加速时间
RampUpTime	INPUT	REAL	5.00	不使用冲击限制时，将轴从停止状态加速到组态的最大速度所需的时间（以秒为单位）。 更改将会影响变量 <轴名称>. Config.DynamicDefaults.Acceleration。有关上述更改产生的影响，请参见变量的说明。	
ChangeRampDown	INPUT	BOOL	FALSE	TRUE	更改减速时间以与输入参数“RampDownTime”相对应
RampDownTime	INPUT	REAL	5.00	不使用冲击限制器时，将轴从组态的最大速度减速到停止状态所需的时间（以秒为单位）。 更改将会影响变量 <轴名称>. Config.DynamicDefaults.Deceleration。有关上述更改产生的影响，请参见变量的说明。	
ChangeEmergency	INPUT	BOOL	FALSE	TRUE	按照输入参数“EmergencyRampTime”更改急停减速时间
EmergencyRampTime	INPUT	REAL	2.00	在急停模式下不使用冲击限制器时，将轴从组态的最大速度减速到停止状态所需的时间（以秒为单位）。 更改将会影响变量 <轴名称>. Config.DynamicDefaults.EmergencyDeceleration。有关上述更改产生的影响，请参见变量的说明。	
ChangeJerkTime	INPUT	BOOL	FALSE	TRUE	按照输入参数“JerkTime”更改平滑时间
JerkTime	INPUT	REAL	0.25	用于轴加速斜坡和轴减速斜坡的平滑时间（以秒为单位） 更改将会影响变量 <轴名称>. Config.DynamicDefaults.Jerk。有关上述更改产生的影响，请参见变量的说明。	

参数	声明	数据类型	默认值	说明	
Done	OUTPUT	BOOL	FALSE	TRUE	更改的值已写入工艺数据块。变量的说明将显示更改生效的时间。
Error	OUTPUT	BOOL	FALSE	TRUE	执行命令期间出错。错误原因, 请参见“ErrorID”和“ErrorInfo”的参数说明。
ErrorID	OUTPUT	WORD	16#0000	参数“Error”的错误 ID (页 300)	
ErrorInfo	OUTPUT	WORD	16#0000	参数“ErrorID”的错误信息 ID (页 300)	

### 说明

在输入参数“RampUpTime”、“RampDownTime”、“EmergencyRampTime”和“JerkTime”中, 输入值可能会导致以下生成的参数超出允许的限值范围: “加速度”(Acceleration)、 “减速度”(Deceleration) 、“急停减速度”(Emergency stop deceleration) 和“冲击”(Jerk)。请确保输入处于有效范围内, 同时注意“动态 (页 107)”部分给出的等式和限值。

### 参见

- [MC\\_Power:](#) 启用、禁用轴 V6 及更高版本 (页 228)
- [MC\\_Reset:](#) 确认故障, 重新启动工艺对象 (V6 及以上版本) (页 235)
- [MC\\_Home:](#) 使轴归位, 设置参考点 (V6 及以上版本) (页 238)
- [MC\\_Halt:](#) 停止轴 (V6 及以上版本) (页 245)
- [MC\\_MoveAbsolute:](#) 轴的绝对定位 (V6 及以上版本) (页 249)
- [MC\\_MoveRelative:](#) 轴的相对定位 (V6 及以上版本) (页 254)
- [MC\\_MoveVelocity:](#) 以设定速度移动轴 (V6 及以上版本) (页 259)
- [MC\\_MoveJog:](#) 在点动模式下移动轴 (V6 及以上版本) (页 265)
- [MC\\_CommandTable:](#) 按照运动顺序运行轴命令 (V6 及以上版本) (页 270)
- [MC\\_ReadParam:](#) 连续读取定位轴的运动数据 (V6 及以上版本) (页 277)
- [MC\\_WriteParam:](#) 写入定位轴的变量 (V6 及以上版本) (页 280)
- [定位轴工艺对象 V6...7 的变量 \(页 335\)](#)

## 11.1.11 MC\_ReadParam

### 11.1.11.1 MC\_ReadParam: 连续读取定位轴的运动数据 (V6 及以上版本)

#### 说明

“MC\_ReadParam”运动控制指令可连续读取轴的运动数据和状态消息。相应变量的当前值在命令的起始处决定。

可以读取以下运动数据和状态消息：

- 工艺版本 V4 及更高版本：
  - 轴的位置设定值
  - 轴的速度设定值和实际值
  - 轴距目标位置的当前距离
  - 轴的目标位置
- 工艺版本 V5 及更高版本的其它信息：
  - 轴的实际位置
  - 轴的实际速度
  - 当前的跟随误差
  - 驱动器状态
  - 编码器状态
  - 状态位
  - 错误位

#### 要求

定位轴工艺对象已正确组态。

#### 超驰响应

任何其它运动控制命令均无法中止 MC\_ReadParam 命令。

新的 MC\_ReadParam 命令不会中止任何激活的运动控制命令。

## 参数

参数	声明	数据类型	默认值	说明	
Enable	INPUT	BOOL	FALSE	TRUE	读取通过“Parameter”指定的变量并将值存储在通过“Value”指定的目标地址中。
				FALSE	不会更新已分配的运动数据
Parameter	INPUT	VARIANT (REAL)	-	指向要读取的值的 VARIANT 指针。允许使用下列变量： <ul style="list-style-type: none"> <li>• &lt;轴名称&gt;.Position</li> <li>• &lt;轴名称&gt;.Velocity</li> <li>• &lt;轴名称&gt;.ActualPosition</li> <li>• &lt;轴名称&gt;.ActualVelocity</li> <li>• &lt;轴名称&gt;.StatusPositioning.&lt;变量名称&gt;</li> <li>• &lt;轴名称&gt;.StatusDrive.&lt;变量名称&gt;</li> <li>• &lt;轴名称&gt;.StatusSensor.&lt;变量名称&gt;</li> <li>• &lt;轴名称&gt;.StatusBits.&lt;变量名称&gt;</li> <li>• &lt;轴名称&gt;.ErrorBits.&lt;变量名称&gt;</li> </ul> 有关指定变量的说明和变量结构，请参见附录定位轴工艺对象 V6...7 的变量 (页 335)。	
Value	INOUT	VARIANT (REAL)	-	指向写入所读取值的目标变量或目标地址的 VARIANT 指针。	
Valid	OUTPUT	BOOL	FALSE	TRUE	读取的值有效。
				FALSE	读取的值无效。
Busy	OUTPUT	BOOL	FALSE	TRUE	命令正在执行
Error	OUTPUT	BOOL	FALSE	TRUE	执行命令期间出错。错误原因，请参见“ErrorID”和“ErrorInfo”的参数说明。
ErrorID	OUTPUT	WORD	16#0000	参数“Error”的错误 ID (页 300)	
ErrorInfo	OUTPUT	WORD	16#0000	参数“ErrorID”的错误信息 ID (页 300)	

## 参见

**MC\_Power:** 启用、禁用轴 V6 及更高版本 (页 228)

**MC\_Reset:** 确认故障，重新启动工艺对象 (V6 及以上版本) (页 235)

**MC\_Home:** 使轴归位，设置参考点 (V6 及以上版本) (页 238)

**MC\_Halt:** 停止轴 (V6 及以上版本) (页 245)

**MC\_MoveAbsolute:** 轴的绝对定位 (V6 及以上版本) (页 249)

**MC\_MoveRelative:** 轴的相对定位 (V6 及以上版本) (页 254)

**MC\_MoveVelocity:** 以设定速度移动轴 (V6 及以上版本) (页 259)

**MC\_MoveJog:** 在点动模式下移动轴 (V6 及以上版本) (页 265)

**MC\_CommandTable:** 按照运动顺序运行轴命令 (V6 及以上版本) (页 270)

**MC\_ChangeDynamic:** 更改轴的动态设置 (V6 及以上版本) (页 274)

**MC\_WriteParam:** 写入定位轴的变量 (V6 及以上版本) (页 280)

## 11.1.12 MC\_WriteParam

### 11.1.12.1 MC\_WriteParam: 写入定位轴的变量 (V6 及以上版本)

#### 说明

运动控制指令“MC\_WriteParam”可在用户程序中写入定位轴工艺对象的变量。与用户程序中变量的赋值不同的是，“MC\_WriteParam”还可以更改只读变量的值。

有关变量、写入变量的条件以及变量生效的时间，请参见工艺对象变量 (页 335) 的说明。

#### 要求

- 定位轴工艺对象已正确组态。
- 要在用户程序中写入只读变量，必须禁用轴。
- 更改需要重新启动的变量不能使用“MC\_WriteParam”写入。

#### 超驰响应

任何其它运动控制命令均无法中止 MC\_WriteParam 命令。

新的 MC\_WriteParam 命令不会中止任何激活的运动控制命令。

## 参数

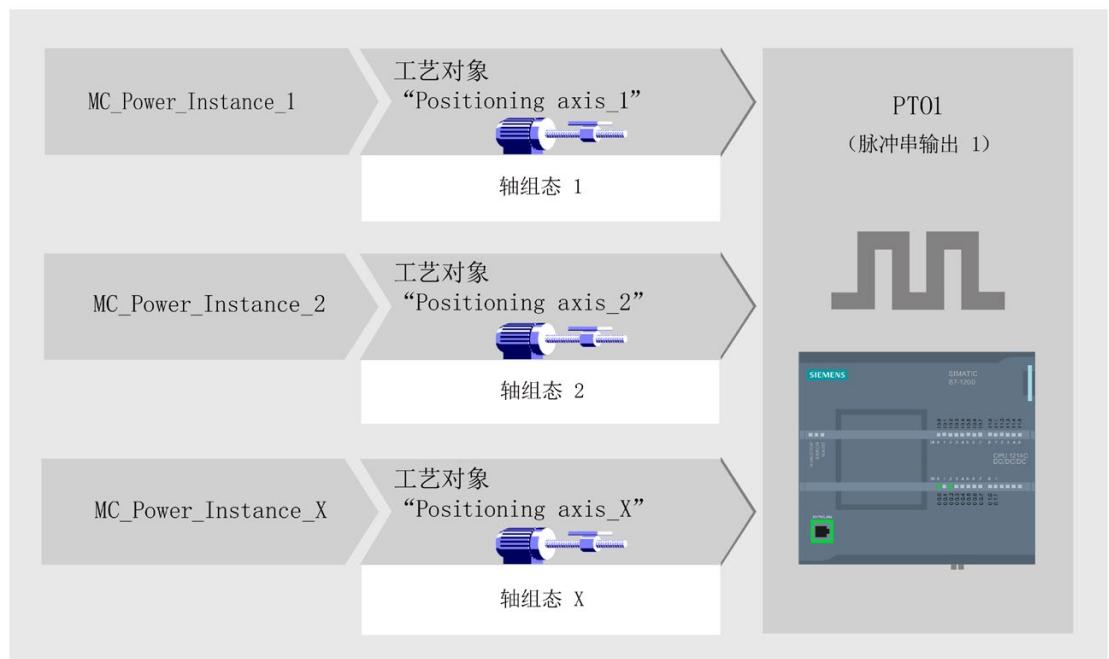
参数	声明	数据类型	默认值	说明	
Parameter	INPUT	VARIANT (BOOL, INT, DINT, UDINT, REAL)	-	指向要写入的工艺对象变量 (页 335) 定位轴 (目标地址) 的 VARIANT 指针	
Value	INPUT	VARIANT (BOOL, INT, DINT, UDINT, REAL)	-	指向要写入的值 (源地址) 的 VARIANT 指针	
Execute	INPUT	BOOL	FALSE	上升沿时启动命令	
Done	OUTPUT	BOOL	FALSE	TRUE	值已写入
Busy	OUTPUT	BOOL	FALSE	TRUE	命令正在执行
Error	OUTPUT	BOOL	FALSE	TRUE	执行命令期间出错。错误原因, 请参见“ErrorID”和“ErrorInfo”的参数说明。
ErrorID	OUTPUT	WORD	16#0000	参数“Error”的错误 ID (页 300)	
ErrorInfo	OUTPUT	WORD	16#0000	参数“ErrorID”的错误信息 ID (页 300)	

参见

- MC\_Power:** 启用、禁用轴 V6 及更高版本 (页 228)
- MC\_Reset:** 确认故障，重新启动工艺对象 (V6 及以上版本) (页 235)
- MC\_Home:** 使轴归位，设置参考点 (V6 及以上版本) (页 238)
- MC\_Halt:** 停止轴 (V6 及以上版本) (页 245)
- MC\_MoveAbsolute:** 轴的绝对定位 (V6 及以上版本) (页 249)
- MC\_MoveRelative:** 轴的相对定位 (V6 及以上版本) (页 254)
- MC\_MoveVelocity:** 以设定速度移动轴 (V6 及以上版本) (页 259)
- MC\_MoveJog:** 在点动模式下移动轴 (V6 及以上版本) (页 265)
- MC\_CommandTable:** 按照运动顺序运行轴命令 (V6 及以上版本) (页 270)
- MC\_ChangeDynamic:** 更改轴的动态设置 (V6 及以上版本) (页 274)
- MC\_ReadParam:** 连续读取定位轴的运动数据 (V6 及以上版本) (页 277)
- 定位轴工艺对象 V6...7 的变量 (页 335)

## 12.1 将多个轴与同一个 PTO 一起使用

使用 CPU S7-1200 的运动控制功能可通过同一 PTO (Pulse Train Output) 运行多个定位轴工艺对象，因而也可以使用相同的 CPU 输出运行多个定位轴工艺对象。例如，这适用于通过一个 PTO 将不同的轴组态用于不同的生产工序。如下所述，可以根据需要频繁地切换这些轴组态。下图显示了基本的功能关系：



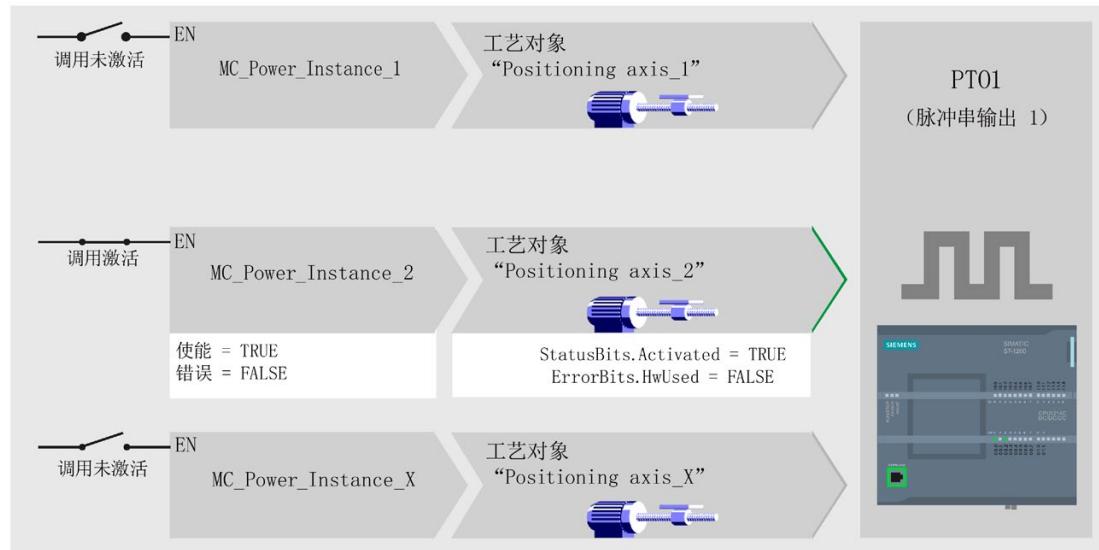
在本例中，多个具有不同轴组态的定位轴工艺对象使用同一 PTO。必须在用户程序中通过单独调用使用独立背景数据块的运动控制指令“MC\_Power”来调用各个轴。每次只能有一个轴使用 PTO。通过变量 <轴名称>.StatusBits.Activated = TRUE 指示当前使用 PTO 的轴。

## 12.1 将多个轴与同一个 PTO 一起使用

### 切换定位轴工艺对象

下列程序图式说明了如何切换不同的工艺对象，从而切换不同的轴组态。要将一个 PTO 与多个轴一起使用而不会出现错误指示，仅可调用当前所用轴的运动控制指令。

下图使用运动控制指令“MC\_Power”作为示例显示了该原则：



激活轴（此处为“Positioning axis\_2”）的变量在用户程序中通常显示以下内容：

- <轴名称>.StatusBits.Activated = TRUE
- <轴名称>.ErrorBits.HWUsed = FALSE

要从一个定位轴工艺对象切换至另一个，请按以下步骤操作。在本例中，执行从“Positioning axis\_2”到“Positioning axis\_1”的切换：

1. 结束所激活“Positioning axis\_2”的所有激活行进运动
2. 使用关联的运动控制指令“MC\_Power”（输入参数 Enable = FALSE）禁用“Positioning axis\_2”
3. 要检查“Positioning axis\_2”是否已禁用，需对运动控制指令“MC\_Power”的输出参数 Status = FALSE 与工艺对象变量 <轴名称>.StatusBits.Enable = FALSE 进行“与”运算。
4. 禁用对“Positioning axis\_2”的运动控制指令的有条件调用。
5. 激活对“Positioning axis\_1”的运动控制指令的有条件调用。第一次调用相应的运动控制指令“MC\_Power”时，将禁用“Positioning axis\_2”并激活“Positioning axis\_1”。

6. 使用关联的运动控制指令“MC\_Power”（输入参数 `Enable = TRUE`）启用“Positioning axis\_1”。
7. 要检查“Positioning axis\_1”是否已启用，请对运动控制指令“MC\_Power”的输出参数 `Status = TRUE` 与工艺对象变量 `<轴名称>.StatusBits.Enable = TRUE` 进行“与”运算。  
也可以循环调用使用单个 PTO 的所有轴的所有运动控制指令。



轴启用（此处为“Positioning axis\_2”）时，该轴进入激活状态。

与有条件调用不同，禁用轴（此处为“Positioning axis\_1”和“Positioning axis\_x”）的运动控制指令将指示错误。这些轴的变量指示状态 `<轴名称>.StatusBits.Activated = FALSE` 和 `<轴名称>.ErrorBits.HwUsed = TRUE`。

如果要在执行用户程序的过程中不出现错误提示，则需要对运动控制指令进行有条件调用。

## 参见

定位轴工艺对象 V4...5 的变量 (页 485)

将多个驱动器与同一个 PTO 一起使用 (页 286)

在更高优先级等级（执行级别）中跟踪作业 (页 287)

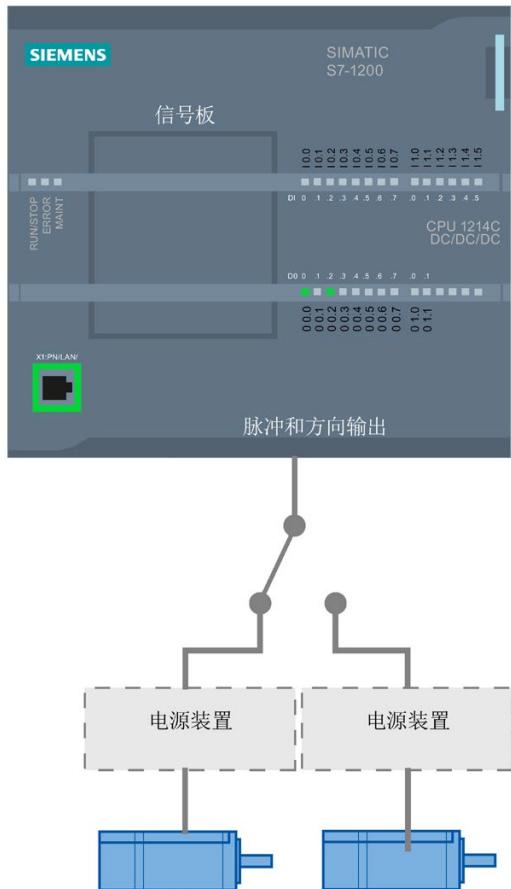
使用通过 PTO 的驱动器连接时的软限位开关时的特殊情况 (页 290)

ErrorID 和 ErrorInfo 列表（工艺对象 V4...5）(页 414)

## 12.2 将多个驱动器与同一个 PTO 一起使用

### 12.2 将多个驱动器与同一个 PTO 一起使用

如果要使用多个驱动器，则可以使用切换功能基于通用 PTO (Pulse Train Output) 运行这些驱动器。下表展示了基本电路设计：



如果需要，可由用户程序通过数字量输出来控制驱动器之间的切换。如果各个驱动器需要不同的轴组态，则需要为 PTO 切换这些组态。更多相关信息，请参见“使用具有相同 PTO 的多个轴”(页 283)。

### 参见

[定位轴工艺对象 V4...5 的变量 \(页 485\)](#)

[将多个轴与同一个 PTO 一起使用 \(页 283\)](#)

[在更高优先级等级（执行级别）中跟踪作业 \(页 287\)](#)

[使用通过 PTO 的驱动器连接时的软限位开关时的特殊情况 \(页 290\)](#)

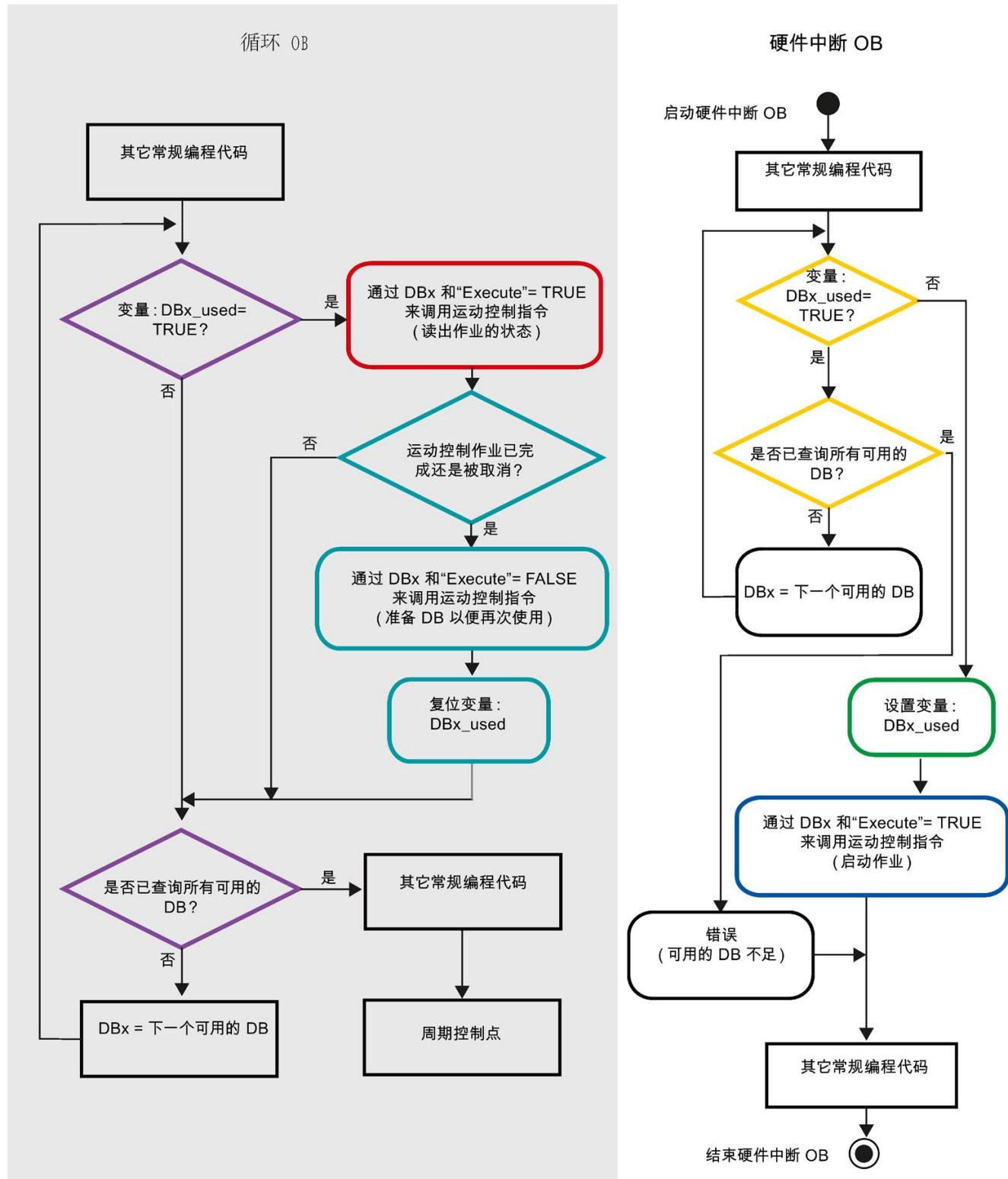
[ErrorID 和 ErrorInfo 列表（工艺对象 V4...5）\(页 414\)](#)

## 12.3 在更高优先级等级（执行级别）中跟踪作业

根据应用情况，可能需要在更高优先级等级（执行级别）中启动运动控制命令（例如，中断控制）。

用于状态监控的运动控制指令必须以短周期调用。如果优先级更高的运动控制命令仅调用一次或调用间隔过大，就不能保证运动控制命令监控的充分性和紧密性。在这些情况下，可以在循环 OB 中执行跟踪。当前没有使用的背景数据块对于每个优先级别更高的运动控制命令都必须是可用的。下面的流程图显示了如何在更高优先级等级（例如，硬件中断 OB）中启动运动控制命令并继续在程序循环 OB 中跟踪：

## 12.3 在更高优先级等级（执行级别）中跟踪作业



根据想要启动运动控制命令的频率，必须生成足够数量的背景数据块。用户在 **DBx\_used** 变量中确定当前使用的背景数据块。

## 在硬件中断 OB 中启动运动控制命令

对 **DBx\_used** 变量（橙色）进行二进制查询以查找当前未使用的背景数据块。如果找到未被使用的背景数据块，则使用的背景数据块标记为“已使用”（绿色），并且将使用该背景数据块（蓝色）启动运动控制命令。

然后执行硬件中断 OB 的所有其它程序部分，接着返回到程序循环 OB。

## 在程序循环 OB 中跟踪启动的运动控制命令

检查循环 OB 中可用的所有背景数据块以确定确定“**DBx\_used**”变量（紫罗兰色）当前是否正在使用它们。

如果背景数据块正在使用（正在处理运动控制命令），则调用使用该背景数据块且输入参数 **Execute = TRUE** 的运动控制指令来读出状态消息（红色）。

如果命令完成或已中止，将接着执行以下操作（青绿色）：

- 调用输入参数 **Execute = FALSE** 的运动控制指令
- 重置 **DBx\_used** 变量

这随即完成命令跟踪，并且该背景数据块此时可供再次使用。

## 参见

定位轴工艺对象 V4...5 的变量 (页 485)

将多个轴与同一个 PTO 一起使用 (页 283)

将多个驱动器与同一个 PTO 一起使用 (页 286)

使用通过 PTO 的驱动器连接时的软限位开关时的特殊情况 (页 290)

**ErrorID** 和 **ErrorInfo** 列表（工艺对象 V4...5）(页 414)

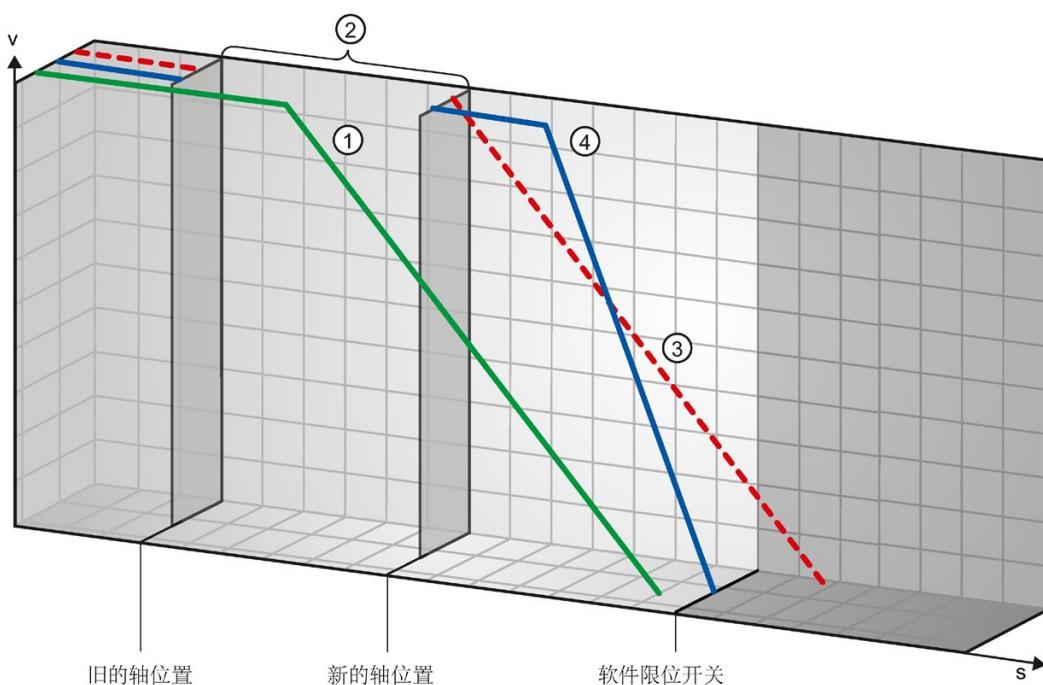
## 12.4 使用通过 PTO 的驱动器连接时的软限位开关时的特殊情况

### 12.4.1 软件限位开关与回原点操作结合使用

如果回原点作业参数化不当，在软限位开关处可能会影响轴的制动操作。开发程序时考虑以下示例。

#### 示例 1：

行进命令执行期间，回原点作业（例如，设置参考点）使当前轴位置在软限位开关方向上偏移。仍可在到达软限位开关前使轴停止：

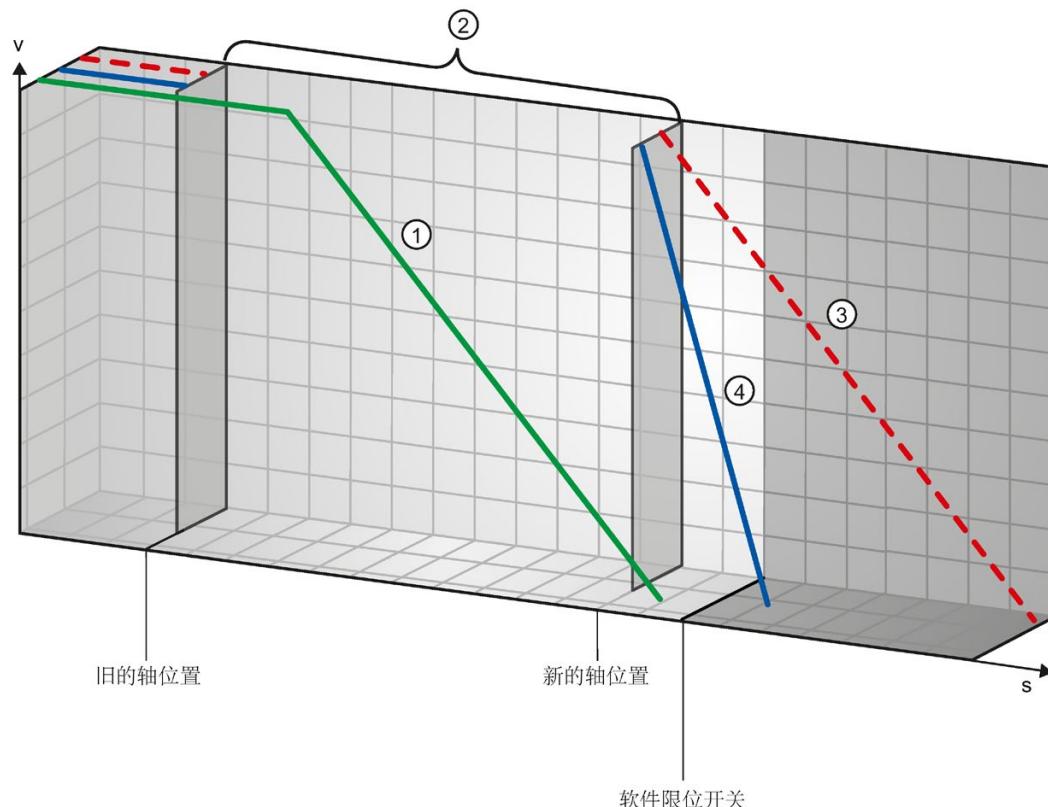


①	绿色曲线显示了无回原点作业时的运动。轴以组态的减速度制动并停止在软限位开关前的某处。
②	由于回原点作业而设置新的轴位置。所以会“跳过”新旧轴位置间的区域。
③	基于新的轴位置，轴理论上可以按组态的减速度制动并停止在软限位开关后的某处（红色曲线）。
④	由于以组态的减速度无法再进行足够的制动，所以轴实际遵从蓝色曲线。在一段恒速运动后，轴以组态的急停减速度制动并停止在软限位开关的位置处。

## 12.4 使用通过 PTO 的驱动器连接时的软限位开关时的特殊情况

## 示例 2：

行进命令执行期间，回原点作业（例如，设置参考点）使当前轴位置在软限位开关方向上偏移。与示例 1 不同，无法再在到达软限位开关前使轴停止。轴超出软限位开关的位置。

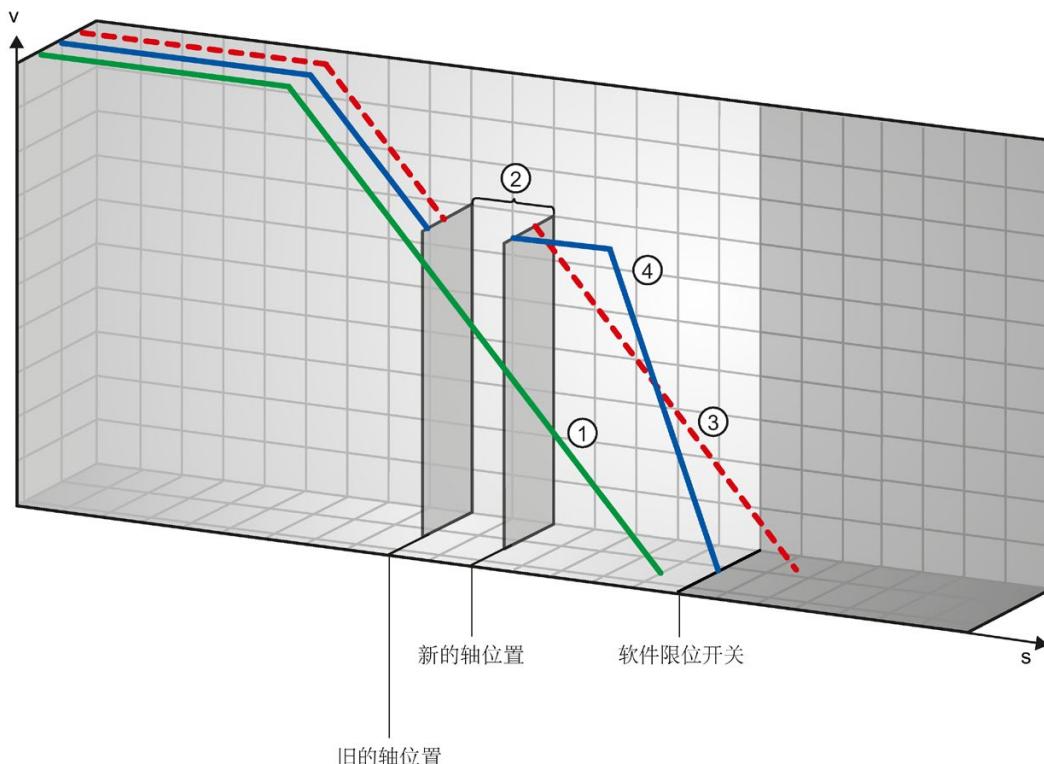


①	绿色曲线显示了无回原点作业时的运动。轴以组态的减速度制动并停止在软限位开关前的某处。
②	由于回原点作业而设置新的轴位置。所以会“跳过”新旧轴位置间的区域。
③	基于新的轴位置，轴理论上可以按组态的减速度制动并停止在软限位开关后较远的某处（红色曲线）。
④	由于以组态的减速度无法再进行足够的制动，所以轴实际遵从蓝色曲线。轴以组态的急停减速速度制动。但该急停减速速度不足以使轴停止在软限位开关的位置处。将超出软限位开关的位置。

## 12.4 使用通过 PTO 的驱动器连接时的软限位开关时的特殊情况

### 示例 3：

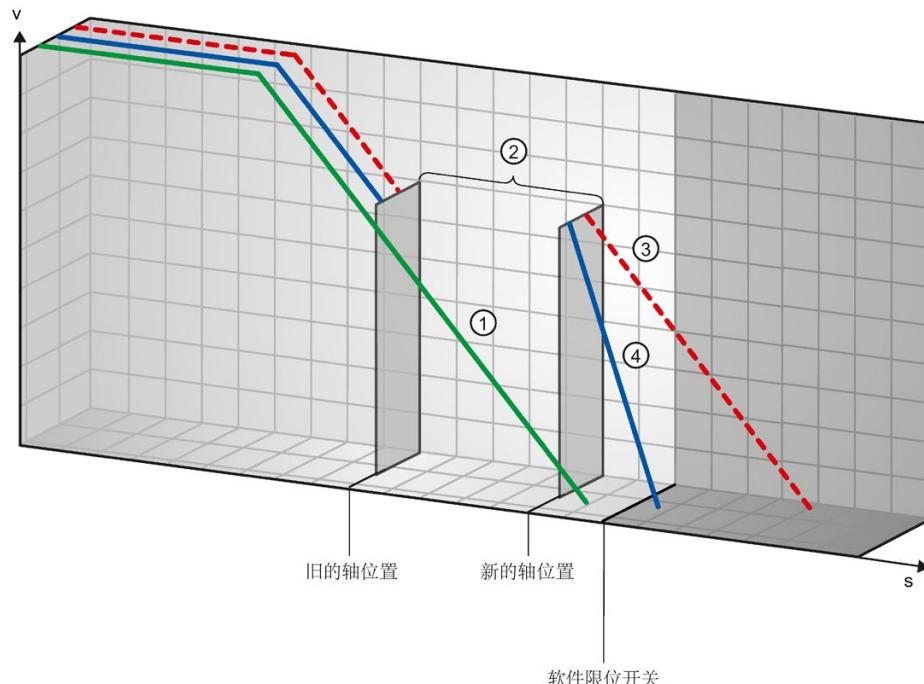
制动操作期间，回原点作业（例如，设置参考点）使当前轴位置在软限位开关方向上偏移。仍可在到达软限位开关前使轴停止：



①	绿色曲线显示了无回原点作业时的运动。轴以组态的减速度制动并停止在软限位开关前的某处。
②	由于回原点作业而设置新的轴位置。所以会“跳过”新旧轴位置间的区域。
③	基于新的轴位置，轴理论上可以按组态的减速度制动并停止在软限位开关后的某处（红色曲线）。
④	由于以组态的减速度无法再进行足够的制动，所以轴实际遵从蓝色曲线。在一段恒速运动后，轴以组态的急停减速度制动并停止在软限位开关的位置处。

**示例 4：**

制动操作期间，回原点作业（例如，设置参考点）使当前轴位置在软限位开关方向上偏移。与示例 3 不同，无法再在到达软限位开关前使轴停止。轴超出软限位开关的位置。



①	绿色曲线显示了无回原点作业时的运动。轴以组态的减速度制动并停止在软限位开关前的某处。
②	由于回原点作业而设置新的轴位置。所以会“跳过”新旧轴位置间的区域。
③	基于新的轴位置，轴理论上可以按组态的减速度制动并停止在软限位开关后较远的某处（红色曲线）。
④	由于以组态的减速度无法再进行足够的制动，所以轴实际遵从蓝色曲线。轴以组态的急停减速度制动。但该急停减速度不足以使轴停止在软限位开关的位置处。将超出软限位开关的位置。

**参见**

软件限位开关和软件限位开关的位置变化。(页 294)

软件限位开关与动态更改结合使用(页 295)

触发轴限位时的轴响应(页 104)

---

## 12.4 使用通过 PTO 的驱动器连接时的软限位开关时的特殊情况

### 12.4.2 软件限位开关和软件限位开关的位置变化。

在用户程序运行期间，如果软件限位开关的位置发生异常变化，当前轴位置和软件限位开关的位置之间的距离将会突然减少。

轴的响应与软件限位开关与回原点操作结合使用 (页 290) 中所述的内容相似。

### 参见

[软件限位开关与回原点操作结合使用 \(页 290\)](#)

[软件限位开关与动态更改结合使用 \(页 295\)](#)

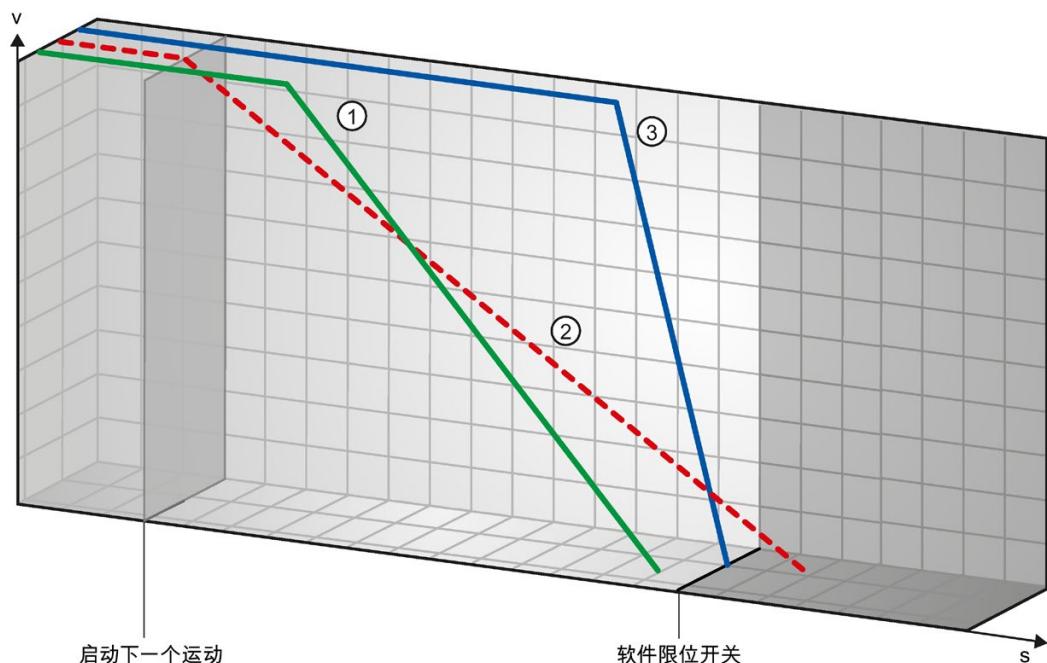
[触发轴限位时的轴响应 \(页 104\)](#)

### 12.4.3 软件限位开关与动态更改结合使用

可以结合覆盖运动命令来改变轴在软限位开关区域中的减速度。这适用于启动具有较低减速度（变量“<轴名称>.DynamicDefaults.Deceleration”）的覆盖运动命令的情况。开发程序时考虑以下示例。

#### 示例 1：

轴运动期间，一个具有较低减速度的运动命令覆盖另一个激活的运动命令：

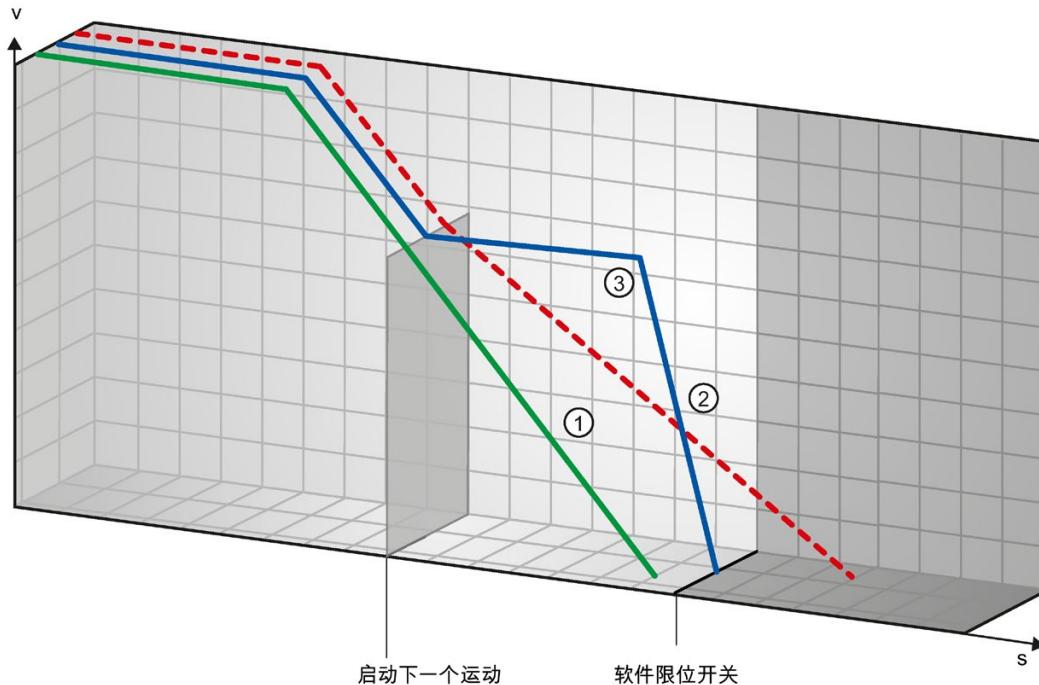


①	绿色曲线显示了当前命令未超驰时的运动。轴以组态的减速度制动并停止在软限位开关前的某处。
②	基于具有较低减速度的覆盖运动命令，轴理论上会以组态的减速度停止在软限位开关后的某处（红色曲线）。
③	由于以覆盖运动命令的组态减速度无法再进行足够的制动，因此轴实际上遵从蓝色曲线。在一段恒速运动后，轴以急停减速度制动并停止在软限位开关的位置处。

## 12.4 使用通过 PTO 的驱动器连接时的软限位开关时的特殊情况

### 示例 2:

轴制动期间，一个具有较低减速度的运动命令覆盖另一个激活的运动命令：



①	绿色曲线显示了当前命令未超驰时的运动。轴以组态的减速度制动并停止在软限位开关前的某处。
②	基于具有较低减速度的覆盖运动命令，轴理论上会停止在软限位开关后较远的某处（红色曲线）。
③	由于以覆盖运动命令的组态减速度无法再进行足够的制动，因此轴实际上遵从蓝色曲线。在一段恒速运动后，轴以急停减速度制动并停止在软限位开关的位置处。

### 参见

软件限位开关与回原点操作结合使用 (页 290)

软件限位开关和软件限位开关的位置变化。 (页 294)

触发轴限位时的轴响应 (页 104)

## 12.5 降低短时间定位的速度

当计划的定位时间小于 2 ms 时，CPU 可以降低定位命令的速度。

然后，命令执行的速度将会在整个持续时间内降低。降低的速度（以脉冲/秒为单位）计算如下：

- 降低的速度 = 要输出的脉冲数 \* 500 Hz

如果计划的定位时间大于 2 ms，则速度不降低。

## 12.6 启动/停止速度的动态调整

如果组态速度限值（启动/停止速度和最大速度）、动态值（加速度、减速度和加加速度）以及运行命令的目标速度，在某些情况下可能导致 CPU 动态调整启动/停止速度。

如果因组态的启动/停止速度较低而导致第一个脉冲所需的时间大于整个加速度时间，则将执行以上操作。在这些情况下的第一个脉冲输出速度比组态的启动/停止速度大很多。后续的脉冲也会动态地调整以确保加速过程可以在指定的时间内完成。

如果发生脉冲丢失，则需确保将当前使用的硬件设备（驱动装置）调整到该状态或更改了轴的动态设置，从而避免启动/停止速度发生动态调整。

## 12.7 维修时，无需位置控制，即可移动轴

### 12.7 维修时，无需位置控制，即可移动轴

需要修时，无需位置控制，即可移动 PROFIdrive 驱动器或带模拟驱动接口的驱动器。

编码器值无效或不正确时即是这种情况。

以下运动控制指令会影响位置控制的状态：

#### 无需位置控制，使用 MC\_Power 启用位置控制驱动器

在非位置控制模式下，利用运动控制指令 MC\_Power StartMode = 0 启动轴。

在另一运动控制指令变为位置控制状态之前，非位置控制模式一直有效。

#### MC\_MoveVelocity

MC\_MoveVelocity (PositionControlled = FALSE) 强制执行非位置控制模式。

MC\_MoveVelocity (PositionControlled = TRUE) 强制执行位置控制模式。

即使 MC\_MoveVelocity 终止，也执行所选的位置控制模式。

#### MC\_MoveJog

MC\_MoveJog (PositionControlled = FALSE) 强制执行非位置控制模式。

MC\_MoveJog (PositionControlled = TRUE) 强制执行位置控制模式。

即使 MC\_MoveJog 终止，也执行所选的位置控制模式。

### MC\_Home, MC\_MoveRelative, MC\_MoveAbsolute

运动控制指令 MC\_Home、MC\_MoveRelative 和 MC\_MoveAbsolute 强制执行位置控制模式。

即使指令结束，位置控制也有效。

### MC\_Halt

运动控制指令“MC\_Halt”可在位置控制模式和非位置控制模式中执行。

位置控制的状态不会变更。

### 参见

[MC\\_Power: 启用、禁用轴 V6 及更高版本 \(页 228\)](#)

[MC\\_Halt: 停止轴 \(V6 及以上版本\) \(页 245\)](#)

## 12.8 ErrorID 和 ErrorInfo (工艺对象 V6...V7) 的列表

### 12.8.1 错误和 ErrorID 概览

下表列出了运动控制指令中指示的所有 ErrorID 和 ErrorInfo。除了错误原因，还列出了消除错误的补救措施。

根据错误响应，轴在停止时如果发生操作错误，将会停止轴。可能会出现以下错误响应：

- 取消启用

输出设定点 0，并取消启用。轴将根据驱动装置中的组态制动至停止状态。

- 通过急停斜坡功能进行停止

将中止处于激活状态的运动命令。轴将通过在“工艺对象 > 扩展参数 > 动态 > 急停斜坡功能”(Technology object > Extended parameters > Dynamics > Emergency stop ramp) 中组态的紧急减速速度功能进行制动（没有任何加加速度的限制），并转入停止状态。

### 错误和 ErrorID 概览

错误	ErrorID
伴随轴停止的运行错误 (页 301)	16#8000-16#8013
不伴随轴停止的运行错误 (页 307)	16#8200-16#820C
块参数错误 (页 310)	16#8400-16#8412
轴的组态错误 (页 315)	16#8600-16#864B
命令表的组态错误 (页 333)	16#8700-16#8704
内部错误 (页 334)	16#8FF

## 12.8.2 ErrorID 16#8000-16#8013

伴随轴停止的运行错误

ErrorID	ErrorInfo	说明	解决方法	错误反应
<b>16#8000</b>		驱动器错误，丢失“驱动器就绪”信号		
	16#0001	-	使用指令“MC_Reset”确认错误；提供驱动器信号；根据需要重新启动命令	
<b>16#8001</b>		已触发下限软限位开关		
	16#000E	已经以当前组态的减速度到达下限软限位开关的位置	使用“MC_Reset”指令确认错误；使用运动命令使轴沿正向移动，超出软限位开关的范围	取消启用
	16#000F	已经以紧急减速度到达下限软限位开关的位置		
	16#0010	已经以紧急减速度超出下限软限位开关的位置		
<b>16#8002</b>		已触发上限软限位开关		
	16#000E	已经以当前组态的减速度到达上限软限位开关的位置	使用“MC_Reset”指令确认错误；使用运动命令使轴沿负向移动，超出软限位开关的范围	取消启用
	16#000F	已经以紧急减速度到达上限软限位开关的位置		
	16#0010	已经以紧急减速度超出上限软限位开关的位置		

## 12.8 ErrorID 和 ErrorInfo (工艺对象 V6...V7) 的列表

ErrorID	ErrorInfo	说明	解决方法	错误反应
16#8003		已接近下限硬限位开关		通过 PTO (Pulse Train Output) 的驱动器连接: 通过急停斜坡功能进行停止 通过 PROFIdrive/模拟量输出进行驱动装置连接: 取消启用
	16#000 E	已接近下限硬限位开关。轴以紧急减速速度停止。 (主动回原点期间, 未找到回原点开关)	使用指令“MC_Reset”确认已启用轴的错误; 使用运动命令使轴沿正向移动, 超出硬限位开关的范围。	
16#8004		已接近上限硬限位开关		通过 PTO (Pulse Train Output) 的驱动器连接: 通过急停斜坡功能进行停止 通过 PROFIdrive/模拟量输出进行驱动装置连接: 取消启用
	16#000 E	已接近上限硬限位开关。轴以紧急减速速度停止。 (主动回原点期间, 未找到回原点开关)	使用指令“MC_Reset”确认已启用轴的错误; 使用运动命令使轴沿负向移动, 超出硬限位开关的范围。	

## 12.8 ErrorID 和 ErrorInfo (工艺对象 V6...V7) 的列表

ErrorID	ErrorInfo	说明	解决方法	错误反应
16#8005		PTO/HSC 已被另一个轴占用		-
	16#0001	-	<p>轴未正确组态： 更正 PTO (Pulse Train Output)/HSC (High Speed Counter) 的组态并将其下载到控制器</p> <p>多个轴将通过一个 PTO 运行： 另一个轴正在使用 PTO / HSC。如果要控制当前轴，则必须使用“MC_Power”Enable = FALSE 禁用另一个轴。 (另请参见“使用具有相同 PTO 的多个轴”(页 283))</p>	
16#8006		轴控制面板中发生通信错误		取消启用
	16#0012	已超时	检查电缆连接，然后再次按下“手动控制”(Manual control) 按钮。	
16#8007		无法启用轴		-
	16#0025	正在重新启动	请等待，直到轴重新启动完成。	
	16#0026	正在 RUN 模式下执行加载过程	请等待，直到加载过程完成。	
16#8008		无效的移动方向		-
	16#002E	不允许所选的运动方向。	<ul style="list-style-type: none"> <li>调整运动方向，然后重新启动该命令。</li> </ul>	
	16#002F	不能将所选的运动方向反转。	<ul style="list-style-type: none"> <li>在工艺对象组态的“扩展参数 &gt; 机械”(Extended parameters &gt; Mechanics) 下可调整允许的旋转方向。重新启动作业。</li> </ul>	

## 12.8 ErrorID 和 ErrorInfo (工艺对象 V6...V7) 的列表

ErrorID	ErrorInfo	说明	解决方法	错误反应
16#8009		未找到参考开关/编码器零位标记		
	16#003 3	编码器的组态、硬件或安装或者回原点开关中出现错误。	<ul style="list-style-type: none"> <li>连接合适的设备。</li> <li>检查设备 (I/O)。</li> <li>比较 HW Config 中的组态和工艺对象。</li> </ul>	通过急停斜坡功能进行停止
16#800A		编码器发出报警消息		
	16#000 1	-	检查设备的功能、连接和 I/O。	取消启用
	16#003 4	编码器中的硬件错误		
	16#003 5	编码器变脏		
	16#003 6	读取编码器绝对值时出错	将驱动装置中的编码器类型或编码器参数 P979 与工艺对象的组态数据进行比较。	
	16#003 7	监视编码器的零位标记	编码器报告零位标记监视错误 (Gx_XIST2 中的错误代码 0x0002, 请参见 PROFIdrive 配置文件)。 检查设备的电磁兼容性 (EMC)。	
	16#003 8	编码器已进入停止状态	<ul style="list-style-type: none"> <li>有关错误原因, 请查看所连接的驱动装置或编码器。</li> <li>检查错误消息是否可能由驱动器或编码器的调试操作触发。</li> </ul>	
	16#004 0	PROFIdrive: 总线上的编码器故障 (站故障)。	检查设备的功能、连接和 I/O。	

## 12.8 ErrorID 和 ErrorInfo (工艺对象 V6...V7) 的列表

ErrorID	ErrorInfo	说明	解决方法	错误反应	
<b>16#800B</b>		<b>位置超出范围</b>		取消启用	
	16#003 9	正方向超出范围	使轴回到有效的实际值范围。		
	16#003 A	负方向超出范围			
	16#003 B	位置控制时钟周期中实际位置的改变大于周期长度。	调整所用编码器的模数长度。		
<b>16#800C</b>		<b>驱动器发出报警消息</b>		取消启用	
	16#000 1	-	检查设备的功能、连接和 I/O。 在“调整”(Tuning) 对话框中，选择较小的增益 (Kv)。		
	16#003 C	PROFIdrive：驱动器信号“请求的控制”故障。			
	16#003 D	PROFIdrive/模拟驱动器接口：驱动器已关闭。			
	16#003 E	PROFIdrive：总线上的驱动器故障（站故障）。			
<b>16#800D</b>		<b>超出了允许的跟随误差</b>		取消启用	
	16#000 1	-	<ul style="list-style-type: none"> <li>检查控制回路的组态。</li> <li>检查编码器的信号方向。</li> <li>检查跟随误差监控的组态。</li> </ul>		
<b>16#800E</b>		<b>硬限位开关错误</b>		取消启用	
	16#004 2	激活硬限位开关的任意行进方向非法	由于激活了硬限位开关，将禁用编程的运动方向。 向相反方向缩回轴。		
	16#004 3	硬限位开关极性反转，无法任意行进。	检查硬限位开关的机械组态。		
	16#004 4	两个硬限位开关均已启用，无法任意行进			

## 12.8 ErrorID 和 ErrorInfo (工艺对象 V6...V7) 的列表

ErrorID	ErrorInfo	说明	解决方法	错误反应
<b>16#800F</b>		目标范围发生错误		
	16#004 5	未达到目标范围	在定位容差时间内未达到目标范围。 <ul style="list-style-type: none"><li>• 检查定位监控的组态。</li><li>• 检查控制回路的组态。</li></ul>	取消启用
	16#004 6	再次离开目标范围	在最短停留时间内已离开目标范围。 <ul style="list-style-type: none"><li>• 检查定位监控的组态。</li><li>• 检查控制回路的组态。</li></ul>	
<b>16#8010</b>		不是模数轴时，下限软限位开关的位置值大于上限软限位开关的位置值。		取消启用
	16#000 1	-	更改软限位开关的位置。	
<b>16#8011</b>		速度逼近回原点开关/零标记等于 0。		取消启用
	16#000 A	值小于或等于 0。	逼近速度 > 选择 0	
<b>16#8012</b>		用于设置回原点位置的回原点速度等于 0。		取消启用
	16#000 A	值小于或等于 0。	回原点速度 > 选择 0	
<b>16#8013</b>		由于“CTRL_PTO”正在使用，轴不能占用 PTO。		取消启用
	16#000 1	-	在组态中选择另一个 PTO。	

### 12.8.3 ErrorID 16#8200-16#820C

不伴随轴停止的运行错误

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8200</b>		轴未启用		
	16#000 1	-	启用轴; 重新启动命令	
<b>16#8201</b>		轴已由另一个“MC_Power”实例启用		
	16#000 1	-	仅通过一个“MC_Power”实例启用轴	
<b>16#8202</b>		已超出可同时执行的运动控制命令最大数量 (通过 PTO (Pulse Train Output) 进行驱动装置连接时, 最多 200 条命令, 通过 PROFIdrive/模拟量输出进行驱动装置连接时, 最多 100 条命令)		
	16#000 1	-	减少并行激活命令数; 重新启动命令 如果运动控制指令的参数“Busy”= TRUE, 则说明命令已激活。	
<b>16#8203</b>		轴当前在“手动控制”模式 (轴控制面板) 下运行		
	16#000 1	-	退出“手动控制”; 重新启动命令	
<b>16#8204</b>		轴未回原点		
	16#000 1	-	使用指令“MC_Home”使轴回原点; 重新启动命令	
<b>16#8205</b>		轴当前由用户程序控制 (该错误仅显示在轴控制面板中)		
	16#001 3	轴已在用户程序中启用	使用指令“MC_Power”禁用轴或者在轴控制面板中再次选择“手动控制”(Manual control)	
<b>16#8206</b>		工艺对象尚未启用		
	16#000 1	-	使用指令“MC_Power”Enable = TRUE 启用轴或者在轴控制面板中启用轴。	
<b>16#8207</b>		作业拒绝执行		
	16#000 1	-		

## 12.8 ErrorID 和 ErrorInfo (工艺对象 V6...V7) 的列表

ErrorID	ErrorInfo	说明	解决方法
	16#001 6	主动回原点操作正在进行中；无法启动其它类型的回原点操作。	等到主动回原点完成或通过运动命令（例如，“MC_Halt”）中止主动回原点。
	16#001 8	当轴正在主动或被动回原点时，不能使用命令表进行移动。	一直等到直接地或被动回原点完成。
	16#001 9	正在处理命令表时，轴不能主动或被动回原点。	等到命令表完成或通过运动命令（例如，“MC_Halt”）中止命令表。
	16#005 2	指定的位置超出了数值限制。	在运动控制指令中输入有效的位置值。
	16#005 3	轴处于斜坡上升状态。	等到轴准备就绪，可运行。
	16#005 4	实际值无效	要执行“MC_Home”命令，实际值必须有效。 请检查实际值的状态。工艺对象 <轴名称>.“StatusSensor.State”变量的值必须为 2（有效值）。
	16#005 8	命令已在其它执行层使用。	只能通过“MC_Power”实例调用轴。
	16#006 B	在非位置控制模式下不允许调用。	通过“MC_Power”(StartMode = 1) 在位置控制模式下启用轴。
<b>16#8208</b>		<b>最大速度和启动/停止速度的速度差值无效</b>	
	16#000 2	值不是有效数字	更正该值；重新启动命令
	16#000 A	值小于或等于 0。	
<b>16#8209</b>		<b>“轴”(Axis) 工艺对象的加速度值无效</b>	
	16#000 2	值不是有效数字	更正该值；重新启动命令
	16#000 A	值小于或等于 0。	

## 12.8 ErrorID 和 ErrorInfo (工艺对象 V6...V7) 的列表

ErrorID	ErrorInfo	说明	解决方法	
<b>16#820A</b>		无法重新启动轴		
	16#001 3	轴已在用户程序中启用	使用“MC_Power”指令阻止轴；再次重新启动	
	16#002 7	轴当前在“手动控制”模式（轴控制面板）下运行	退出“手动控制”；再次重新启动	
	16#002 C	轴未禁用。	禁用轴；重新启动命令	
	16#004 7	工艺对象未准备好重新启动。	重新下载项目。	
	16#004 8	不满足工艺对象的重新启动条件。	禁用工艺对象。	
<b>16#820B</b>		无法执行命令表		
	16#002 6	正在 RUN 模式下执行加载过程	请等待，直到加载过程完成。	
<b>16#820C</b>		无组态可用		
	16#000 1	-	内部错误 联系服务热线。	
	16#001 4	所选硬件由另一应用程序使用。		

## 12.8 ErrorID 和 ErrorInfo (工艺对象 V6...V7) 的列表

## 12.8.4 ErrorID 16#8400-16#8412

## 块参数错误

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8400</b>		运动控制指令的“Position”参数值无效		
	16#0002	值不是有效数字	更正该值；重新启动命令	
	16#0005	该值超出值范围（大于 1.0E12）		
	16#0006	该值超出值范围（小于 -1.0E12）		
	16#0030	值的数字格式不正确，或者超出有效的数字范围。		
<b>16#8401</b>		运动控制指令的“Distance”参数值无效		
	16#0002	值不是有效数字	更正该值；重新启动命令	
	16#0005	该值超出值范围（大于 1.0E12）		
	16#0006	该值超出值范围（小于 -1.0E12）		
<b>16#8402</b>		运动控制指令的“Velocity”参数值无效		
	16#0002	值不是有效数字	更正该值；重新启动命令	
	16#0008	值大于组态的最大速度		
	16#0009	值小于组态的启动/停止速度		
	16#0024	值小于 0		
	16#0030	值的数字格式不正确，或者超出有效的数字范围。		

ErrorID	ErrorInfo	说明	解决方法	
16#8403		运动控制指令的“Direction”参数值无效		
	16#001 1	所选值无效	更正所选值；重新启动命令	
16#8404		运动控制指令的“Mode”参数值无效		
	16#001 1	所选值无效	更正所选值；重新启动命令	
	16#001 5	未组态主动/被动回原点	更正组态并将其下载到控制器；启用轴并重新启动命令	
	16#001 7	即使已禁用硬限位开关，也会启用到达硬限位开关后自动反向功能。	<ul style="list-style-type: none"> <li>使用变量 &lt;轴名称&gt;.PositionLimits_HW.Active = TRUE 激活硬限位开关，并重新启动该命令。</li> <li>更正组态并将其下载到控制器；启用轴并重新启动命令</li> </ul>	
	16#005 5	增量编码器的模式无效	需使用参数“Mode”= 0、1、2、3 启动增量编码器的回原点操作。	
	16#005 6	绝对值编码器的模式无效	<p>绝对值编码器无法执行被动和主动回原点操作（“Mode”= 2、3）。</p> <p>需使用参数“Mode”= 0、1 启动绝对值编码器的回原点操作。</p>	
16#8405		运动控制指令的“StopMode”参数值无效		
	16#001 1	所选值无效	更正所选值；再次启用轴	
16#8406		不允许同时向前和向后点动。		
	16#000 1	-	采取措施确保参数“JogForward”和“JogBackward”的信号状态不会同时为 TRUE；重新启动命令。	
16#8407		只有禁用激活的轴后，才允许使用指令“MC_Power”切换到另一个轴。		
	16#000 1	-	禁用激活的轴；然后可以切换到其它轴并启用该轴。	

## 12.8 ErrorID 和 ErrorInfo (工艺对象 V6...V7) 的列表

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8408</b>		运动控制指令的“Axis”参数值无效		
	16#001 A	指定的值与所需的工艺对象版本不匹配	更正该值；重新启动命令	
	16#001 B	指定的值与所需的工艺对象类型不匹配		
	16#001 C	指定的值不是运动控制工艺数据块		
<b>16#8409</b>		运动控制指令的“CommandTable”参数值无效		
	16#001 A	指定的值与所需的工艺对象版本不匹配	更正该值；重新启动命令	
	16#001 B	指定的值与所需的工艺对象类型不匹配		
	16#001 C	指定的值不是运动控制工艺数据块		
<b>16#840A</b>		运动控制指令的“StartStep”参数值无效		
	16#000 A	值小于或等于 0。	更正该值；重新启动命令	
	16#001 D	开始步进大于结束步进		
	16#001 E	值大于 32		
<b>16#840B</b>		运动控制指令的“EndStep”参数值无效		
	16#000 A	值小于或等于 0。	更正该值；重新启动命令	
	16#001 E	值大于 32		

## 12.8 ErrorID 和 ErrorInfo (工艺对象 V6...V7) 的列表

ErrorID	ErrorInfo	说明	解决方法	
<b>16#840C</b>		运动控制指令的“RampUpTime”参数值无效		
	16#0002	值不是有效数字	更正该值；重新启动命令	
	16#0005	该值超出值范围（大于 1.0E12）		
	16#000A	值小于或等于 0。		
<b>16#840D</b>		运动控制指令的“RampDownTime”参数值无效		
	16#0002	值不是有效数字	更正该值；重新启动命令	
	16#0005	该值超出值范围（大于 1.0E12）		
	16#000A	值小于或等于 0。		
<b>16#840E</b>		运动控制指令的“EmergencyRampTime”参数值无效		
	16#0002	值不是有效数字	更正该值；重新启动命令	
	16#0005	该值超出值范围（大于 1.0E12）		
	16#000A	值小于或等于 0。		
<b>16#840F</b>		运动控制指令的“JerkTime”参数值无效		
	16#0002	值不是有效数字	更正该值；重新启动命令	
	16#0005	该值超出值范围（大于 1.0E12）		
	16#000A	值小于或等于 0。		

## 12.8 ErrorID 和 ErrorInfo (工艺对象 V6...V7) 的列表

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8410</b>		运动控制指令的“Parameter”参数值无效		
	16#000 2	值不是有效数字	更正该值；重新启动命令	
	16#000 B	地址无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
	16#002 8	VARIANT 指针“参数”和“值”的数据类型不匹配。	使用适当的数据类型；重新启动命令。	
	16#002 9	VARIANT 指针“参数”并未指向工艺对象的数据块。	更正 VARIANT 指针；重新启动命令	
	16#002 A	无法读取 VARIANT 指针“参数”值。	更正 VARIANT 指针；重新启动命令	
	16#002 B	无法写入 VARIANT 指针“参数”值。	更正 VARIANT 指针或值；重新启动命令	
	16#002 C	轴未禁用。	禁用轴；重新启动命令	
<b>16#8411</b>		运动控制指令的“Value”参数值无效		
	16#000 2	值不是有效数字	更正该值；重新启动命令	
	16#000 5	该值超出值范围（大于 1.0E12）		
	16#000 6	该值超出值范围（小于 -1.0E12）		
<b>16#8412</b>		运动控制指令的“启动模式”(Start Mode) 参数值无效		
	16#001 1	所选值无效	更正所选值；再次启用轴	

### 12.8.5 错误 ID 16#8600-16#864B

轴的组态错误

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8600</b>		脉冲发生器 (PTO) 的参数分配无效		
	16#000 B	地址无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
	16#001 4	所选硬件由另一应用程序使用。		
<b>16#8601</b>		高速计数器 (HSC) 的参数分配无效		
	16#000 B	地址无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
	16#001 4	所选硬件由另一应用程序使用。		
<b>16#8602</b>		“启用输出”的参数设置无效。		
	16#000 B	地址无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
<b>16#8603</b>		“输入就绪”的参数设置无效。		
	16#000 B	地址无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
<b>16#8604</b>		驱动装置每转脉冲数的值无效		
	16#000 A	值小于或等于 0	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	

## 12.8 ErrorID 和 ErrorInfo (工艺对象 V6...V7) 的列表

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8605</b>		“电机每转的距离”无效		
	16#0002	值不是有效数字	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
	16#0005	该值超出值范围（大于 1.0E12）		
	16#000A	值小于或等于 0		
	16#0030	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#8606</b>		最小速度无效		
	16#0002	值不是有效数字	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
	16#0003	值大于硬件上限值		
	16#0004	值小于硬件下限值		
	16#0007	启动/停止速度大于最大速度		
<b>16#8607</b>		“最大速度”(maximum velocity) 值无效		
	16#0002	值不是有效数字	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
	16#0003	值大于硬件上限值		
	16#0004	值小于硬件下限值		
	16#0030	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	

## 12.8 ErrorID 和 ErrorInfo (工艺对象 V6...V7) 的列表

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8608</b>		“加速度”的值无效		
	16#000 2	值不是有效数字	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
	16#000 3	值大于硬件上限值		
	16#000 4	值小于硬件下限值		
	16#003 0	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#8609</b>		减速度无效		
	16#000 2	值不是有效数字	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
	16#000 3	值大于硬件上限值		
	16#000 4	值小于硬件下限值		
	16#003 0	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	

## 12.8 ErrorID 和 ErrorInfo (工艺对象 V6...V7) 的列表

ErrorID	ErrorInfo	说明	解决方法
<b>16#860A</b>		<b>紧急减速速度无效</b>	
	16#000 2	值不是有效数字	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
	16#000 3	值大于硬件上限值	
	16#000 4	值小于硬件下限值	
	16#003 0	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
<b>16#860B</b>		<b>下限限位开关的位置无效</b>	
	16#000 2	值不是有效数字	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
	16#000 5	该值超出值范围（大于 1.0E12）	
	16#000 6	该值超出值范围（小于 -1.0E12）	
	16#003 0	软限位开关的下限位置值大于该软限位开关的上限位置值	
<b>16#860C</b>		<b>上限软限位开关的位置无效</b>	
	16#000 2	值不是有效数字	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
	16#000 5	该值超出值范围（大于 1.0E12）	
	16#000 6	该值超出值范围（小于 -1.0E12）	
	16#003 0	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>

## 12.8 ErrorID 和 ErrorInfo (工艺对象 V6...V7) 的列表

ErrorID	ErrorInfo	说明	解决方法	
<b>16#860D</b>		下限硬限位开关的地址无效。		
	16#000 B	地址无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
	16#000 C	下降沿的地址无效		
	16#000 D	上升沿的地址无效		
<b>16#860E</b>		上限硬限位开关的地址无效。		
	16#000 B	地址无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
	16#000 C	下降沿的地址无效		
	16#000 D	上升沿的地址无效		
<b>16#860F</b>		“回原点位置偏移”的值无效		
	16#000 2	值不是有效数字	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
	16#000 5	该值超出值范围（大于 1.0E12）		
	16#000 6	该值超出值范围（小于 -1.0E12）		
	16#003 0	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	

## 12.8 ErrorID 和 ErrorInfo (工艺对象 V6...V7) 的列表

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8610</b>		<b>接近速度无效</b>		
	16#0002	值不是有效数字	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
	16#0008	速度大于最大速度		
	16#0009	速度小于启动/停止速度		
	16#0030	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#8611</b>		<b>回原点速度无效</b>		
	16#0002	值不是有效数字	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
	16#0008	速度大于最大速度		
	16#0009	速度小于启动/停止速度		
	16#0030	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#8612</b>		<b>回原点开关的地址无效</b>		
	16#000B	地址无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
	16#000C	下降沿的地址无效		
	16#000D	上升沿的地址无效		

ErrorID	ErrorInfo	说明	解决方法
<b>16#8613</b>		即使未组态硬限位开关，也会启用到达硬限位开关后自动反向主动回原点。	
	16#000 1	-	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
<b>16#8614</b>		加加速度无效	
	16#000 2	值不是有效数字	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
	16#001 F	值大于最大加加速度值	
	16#002 0	值小于最小加加速度值	
	16#003 0	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
<b>16#8615</b>		“测量单位”的值无效	
	16#001 1	所选值无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴
<b>16#8616</b>		回原点开关的地址无效（自版本 V4 起为被动回原点）	
	16#001 1	所选值无效	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
	16#003 0	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>

## 12.8 ErrorID 和 ErrorInfo (工艺对象 V6...V7) 的列表

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8617</b>		变量 <轴名称>.Sensor.Sensor[1].ActiveHoming.Mode 的值无效		
	16#001 1	所选值无效 (有效值: 2 = 通过数字量输入回原点)	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器; 使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值; 使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#8618</b>		变量 <轴名称>.Sensor.Sensor[1].PassiveHoming.Mode 的值无效		
	16#001 1	所选值无效 (有效值: 2 = 通过数字量输入回原点)	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器; 使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值; 使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#8619</b>		变量 <轴名称>.Actor.Type 的值无效		
	16#001 1	所选值无效 (有效值: 2 = 通过脉冲接口连接)	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器; 使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值; 使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#861A</b>		“允许的旋转方向”(Permitted direction of rotation) 值无效		
	16#001 1	所选值无效	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器; 使用指令“MC_Power”再次启用轴</li> </ul>	
	16#002 D	方向输出关断时, 不允许“双向”操作。	<ul style="list-style-type: none"> <li>在线更正不正确的值; 使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#861B</b>		负载齿轮因数错误		
	16#003 1	数值无效。	将无错误组态下载到控制器; 使用指令“MC_Power”再次启用轴	
<b>16#861C</b>		通过增量编码器回原点的数据组合非法		
	16#003 1	数值无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器; 使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值; 使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	

ErrorID	ErrorInfo	说明	解决方法
<b>16#861D</b>		设置的编码器安装方式无效。<轴名称>.Sensor.Sensor[1].MountingMode 中的值无效	
	16#001 1	所选值无效	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
<b>16#861E</b>		测量编码器轮周长的组态无效。<轴名称>.Sensor.Sensor[1].Parameter.DistancePerRevolution 中的值无效	
	16#003 0	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
<b>16#861F</b>		组态的线性编码器精度错误。<轴名称>.Sensor.Sensor[1].Parameter.Resolution 中的值无效	
	16#003 0	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
<b>16#8620</b>		组态的 Gx_XIST1 高分辨率无效。<轴名称>.Sensor.Sensor[1].Parameter.FineResolutionXist1 中的值无效	
	16#003 0	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
<b>16#8621</b>		在 <轴名称>.Sensor.Sensor[1].Parameter.FineResolutionXist1 中设置的 Gx_XIST1 高精度值与 PROFIdrive 参数 P979 中的设置不一致	
	16#003 0	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>

## 12.8 ErrorID 和 ErrorInfo (工艺对象 V6...V7) 的列表

ErrorID	ErrorInfo	说明	解决方法	
16#8622		<轴名称>.Actor.Interface.AddressIn 或 <轴名称>.Actor.Interface.AddressOut 中的值无效		
	16#001 1	所选值无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
16#8623		变量 <轴名称>.Sensor.Sensor[1].Type 中设置的值无效。		
	16#001 1	所选值无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
16#8624		设置的编码器系统无效。<轴名称>.Sensor.Sensor[1].System 中的值无效		
	16#001 1	所选值无效	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
16#8625		定位监控的参数错误。<轴名称>.PositioningMonitoring.MinDwellTime 中的值无效		
	16#003 0	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
16#8626		定位监控的参数错误。<轴名称>.PositioningMonitoring.Window 中的值无效		
	16#003 0	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
16#8627		PROFIdrive 接口组态的实际值错误。<轴名称>.Sensor.Sensor[1].Interface.AddressIn 或 <轴名称>.Sensor.Sensor[1].Interface.AddressOut 中的值无效		
	16#001 1	所选值无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8628</b>		<b>控制器数错误</b>		
	16#0030	值的数字格式不正确，或者超出有效的数字范围。	<p>控制回路的增益或预控制值错误。</p> <ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令(&lt;轴名称&gt;.PositionControl.Kv、&lt;轴名称&gt;.PositionControl.Kpc)</li> </ul>	
<b>16#8629</b>		停止信号的限值错误。<轴名称>.StandStillSignal.VelocityThreshold 中的值无效		
	16#0030	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#862A</b>		定位监控的参数错误。<轴名称>.PositioningMonitoring.ToleranceTime 中的值无效		
	16#0030	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#862B</b>		PROFIBUS 参数化不一致；Ti 和 To 的总和大于一个 DP 周期		
	16#0030	值的数字格式不正确，或者超出有效的数字范围。	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
<b>16#862C</b>		停止监视的参数错误。<轴名称>.StandStillSignal.MinDwellTime 中的值无效		
	16#0030	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#862D</b>		跟随误差监控的参数错误。<轴名称>.FollowingError.MinValue 中的值无效		
	16#0030	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	

## 12.8 ErrorID 和 ErrorInfo (工艺对象 V6...V7) 的列表

ErrorID	ErrorInfo	说明	解决方法	
<b>16#862E</b>		<轴名称>.Modulo.Length 中的值无效		
	16#0030	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#862F</b>		<轴名称>.Modulo.StartValue 中的值无效		
	16#0030	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#8630</b>		<轴名称>.Actor.DriveParameter.ReferenceSpeed 中的值无效		
	16#0030	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#8631</b>		设置的 Gx_XIST2 高分辨率无效。<轴名称>.Sensor.Sensor[1].Parameter.FineResolutionXist2 中的值无效		
	16#0030	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#8632</b>		可确定的编码器精度值无效。<轴名称>.Sensor.Sensor[1].Parameter.DeterminableRevolutions 中的值无效		
	16#0030	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#8634</b>		跟随误差监控的参数错误。<轴名称>.FollowingError.MaxValue 中的值无效		
	16#0030	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8635</b>		跟随误差监控的参数错误。<轴名称>.FollowingError.MinVelocity 中的值无效		
	16#0030	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#8636</b>		控制器因数不正确。<轴名称>.PositionControl.Kpc 中预控制因数的值无效		
	16#0030	值的数字格式不正确，或者超出有效的数字范围。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#8637</b>		<轴名称>.Sensor.Sensor[1].Interface.Type 中的值无效		
	16#0011	所选值无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
<b>16#8638</b>		<轴名称>.Sensor.Sensor[1].Interface.HSC 中的值无效		
	16#0011	所选值无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
<b>16#8639</b>		驱动装置错误		
	16#0049	设备组态错误	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
	16#004A	工艺需要一个更小的伺服周期	内部系统错误。 检查项目的一致性并将项目重新加载到控制器中。	
	16#004B	斜坡上升过程中未初始化设备驱动程序。	要启用工艺对象，必须完成执行器驱动程序的初始化。 请稍后再执行该命令。	
	16#006F	不允许多次使用不同轴的地址。	检查该 CPU 中所有的轴，查看相同驱动装置是否多次使用、驱动装置地址与 IO 地址是否相同，或者 IO 地址是否重叠。减少轴的使用次数。	

## 12.8 ErrorID 和 ErrorInfo (工艺对象 V6...V7) 的列表

ErrorID	ErrorInfo	说明	解决方法	
<b>16#863A</b>		与驱动装置的通信错误		
	16#004 C	设备组态错误	<ul style="list-style-type: none"> <li>连接合适的设备。</li> <li>检查设备 (I/O)。</li> <li>比较 HW Config 中的组态和工艺对象。</li> </ul>	
	16#004 D	设备驱动程序需要一个更小的伺服时钟。	<ul style="list-style-type: none"> <li>连接合适的设备。</li> <li>检查设备 (I/O)。</li> <li>比较 HW Config 中的组态和工艺对象。</li> </ul>	
	16#004 E	与设备进行内部通信时发生错误	检查项目的一致性并将项目重新加载到控制器中。	
<b>16#863B</b>		编码器中的错误		
	16#004 9	设备组态错误	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
	16#004 A	工艺需要一个更小的伺服周期	<p>内部系统错误。 检查项目的一致性并将项目重新加载到控制器中。</p>	
	16#004 B	斜坡上升过程中未初始化设备驱动程序。	<p>要启用工艺对象，必须完成执行器驱动程序的初始化。 请稍后再执行该命令。</p>	
	16#006 F	不允许多次使用不同轴的地址。	检查该 CPU 中所有的轴，查看相同驱动装置是否多次使用、驱动装置地址与 IO 地址是否相同，或者 IO 地址是否重叠。减少轴的使用次数。	
<b>16#863C</b>		与编码器的通信错误		
	16#004 C	设备组态错误	<ul style="list-style-type: none"> <li>连接合适的设备。</li> <li>检查设备 (I/O)。</li> <li>比较 HW Config 中的组态和工艺对象。</li> </ul>	
	16#004 D	设备驱动程序需要一个更小的伺服时钟。	<ul style="list-style-type: none"> <li>连接合适的设备。</li> <li>检查设备 (I/O)。</li> <li>比较 HW Config 中的组态和工艺对象。</li> </ul>	
	16#004 E	与设备进行内部通信时发生错误	检查项目的一致性并将项目重新加载到控制器中。	

ErrorID	ErrorInfo	说明	解决方法	
<b>16#863D</b>		与设备（驱动装置或编码器）的通信错误		
	16#004 F	请求的逻辑地址无效。	<ul style="list-style-type: none"> <li>• 连接合适的设备。</li> <li>• 检查设备 (I/O)。</li> <li>• 检查 HW Config 中的拓扑组态。</li> <li>• 比较 HW Config 中的组态和工艺对象。</li> </ul>	
	16#005 0	请求的逻辑输入地址无效。		
	16#005 1	请求的逻辑输出地址无效。		
<b>16#863E</b>		变量“ControlPanel.Input.TimeOut”的值 无效（轴控制面板）		
	16#003 0	值的数字格式不正确，或者超出有效的数字范围。	更正工艺对象 <轴名称>.ControlPanel.Input.TimeOut 的变量值。 以毫秒为单位指定该值。	
<b>16#863F</b>		<轴名称>.Actor.DriveParameter.MaxSpeed 中的值无效		
	16#003 0	值的数字格式不正确，或者超出有效的数字范围。	将驱动装置和工艺对象组态中的参考值更正为 Actuator.MaxSpeed/2。 对于模拟量驱动装置接口，将驱动装置和工艺对象组态中的参考值更正为 Actuator.MaxSpeed/1.17。	

## 12.8 ErrorID 和 ErrorInfo (工艺对象 V6...V7) 的列表

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8640</b>		自动传送设备中的驱动装置参数时出错		
	16#003 0	值的数字格式不正确，或者超出有效的数字范围。	更正值。	
	16#005 9	该设备未分配给任何 SINAMICS 驱动装置或不支持调整所需的服务。	禁用自动传送设备中的编码器参数。 完成轴组态中的所需参数，并重装加载到设备中的轴组态。	
	16#005 A	由于资源不足，取消自动传送参数。		
	16#005 B	只有设备直接连接到 I/O 区域时，才自动传送参数。		
	16#005 C	最大速度 (p1082): 参数不存在，或者参数值无法读取或超出允许的限值范围。硬件发出了错误信号，因此参数的读取过程已中止。	检查原因。如果用户不能删除原因，请禁用自动传送设备中的参数。完成轴组态中的所需参数，并重装加载到设备中的轴组态。	
	16#005 D	最大力矩/力 (p1520): 参数不存在，或者参数值无法读取或超出允许的限值范围。硬件发出了错误信号，因此参数的读取过程已中止。		
	16#005 E	最大力矩/力 (p1521): 参数不存在，或者参数值无法读取或超出允许的限值范围。硬件发出了错误信号，因此参数的读取过程已中止。		
	16#005 F	高精度扭矩/力矩限值 (p1544) 参数不存在，或者参数值无法读取或超出允许的限值范围。硬件发出了错误信号，因此参数的读取过程已中止。		

## 12.8 ErrorID 和 ErrorInfo (工艺对象 V6...V7) 的列表

ErrorID	ErrorInfo	说明	解决方法
	16#0060	额定速度 (p2000): 参数不存在, 或者参数值无法读取或超出允许的限值范围。硬件发出了错误信号, 因此参数的读取过程已中止。	
	16#0061	额定力矩/力 (p2003): 参数不存在, 或者参数值无法读取或超出允许的限值范围。硬件发出了错误信号, 因此参数的读取过程已中止。	
	16#0070	该参数从此设备中无法读取:	检查与该设备的通信连接。
<b>16#8641</b>		自动传送设备中的编码器参数出错	
	16#0030	值的数字格式不正确, 或者超出有效的数字范围。	更正值。
	16#0059	该设备未分配给任何 SINAMICS 驱动装置或不支持调整所需的服务。	禁用自动传送设备中的编码器参数。完成轴组态中的所需参数, 并重装加载到设备中的轴组态。
	16#005A	由于资源不足, 取消自动传送参数。	
	16#005B	只有设备直接连接到 I/O 区域时, 才自动传送参数。	检查原因。如果用户不能删除原因, 请禁用自动传送设备中的参数。完成轴组态中的所需参数, 并重装加载到设备中的轴组态。
	16#0062	编码器系统 (r0979[1/11].0): 参数不存在, 或者参数值无法读取或超出允许的限值范围。硬件发出了错误信号, 因此参数的读取过程已中止。	
	16#0063	编码器精度 (r0979[2/12]): 参数不存在, 或者参数值无法读取或超出允许的限值范围。硬件发出了错误信号, 因此参数的读取过程已中止。	

12.8 *ErrorID* 和 *ErrorInfo* (工艺对象 V6...V7) 的列表

<b>ErrorID</b>	<b>ErrorInfo</b>	<b>说明</b>	<b>解决方法</b>
	16#006 4	编码器高精度 Gx_XIST1 (r0979[3/13]): 参数不存在, 或者参数值无法读取或超出允许的限值范围。硬件发出了错误信号,因此参数的读取过程已中止。	
	16#006 5	编码器高精度 Gx_XIST2 (r0979[4/14]): 参数不存在, 或者参数值无法读取或超出允许的限值范围。硬件发出了错误信号,因此参数的读取过程已中止。	
	16#006 6	可确定的编码器转数 (r0979[5/15]): 参数不存在, 或者参数值无法读取或超出允许的限值范围。硬件发出了错误信号,因此参数的读取过程已中止。	
	16#007 0	该参数从此设备中无法读取:	检查与该设备的通信连接。
<b>16#8646</b>		<轴名称>.Sensor.Interface.Number 中的值无效。	
	16#003 0	值的数字格式不正确, 或者超出有效的数字范围。	修正值, 并将组态装载到设备中。
<b>16#8647</b>		<b>PTO 轴不支持仿真</b>	
	16#000 1	-	禁用仿真
<b>16#864A</b>		<b>Actor.PTOSliceTime</b> 中的值无效 (V7 及以上版本)	
	16#006 D	值大于 20 ms	修正值, 并将组态装载到设备中。
	16#006 E	值小于 2 ms	
<b>16#864B</b>		<b>BehaviourGx_XIST1</b> (V7 或更高版本) 的值无效	
	16#000 1	-	修正值, 并将组态装载到设备中。

### 12.8.6 ErrorID 16#8700-16#8704

命令表的组态错误

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8700</b>		命令表中“命令类型”(command type) 的值无效。		
	16#000 1	-	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值，并视需要重新启动命令</li> </ul>	
<b>16#8701</b>		命令表中“位置/行进路径”(Position/travel path) 的值无效		
	16#000 2	值不是有效数字	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值，并视需要重新启动命令</li> </ul>	
	16#000 5	该值超出值范围（大于 1.0E12）		
	16#000 6	该值超出值范围（小于 -1.0E12）		
<b>16#8702</b>		命令表中的速度无效		
	16#000 2	值不是有效数字	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值，并视需要重新启动命令</li> </ul>	
	16#000 8	值大于组态的最大速度		
	16#000 9	值小于组态的启动/停止速度		
<b>16#8703</b>		命令表中的持续时间无效		
	16#000 2	值不是有效数字	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值，并视需要重新启动命令</li> </ul>	
	16#002 1	值大于 64800 s		
	16#002 2	值小于 0.001 s		

## 12.8 ErrorID 和 ErrorInfo (工艺对象 V6...V7) 的列表

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8704</b>		命令表中“下一步”(Next step) 的值无效。		
	16#001 1	所选值无效	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> </ul>	
	16#002 3	该命令不允许命令转换	<ul style="list-style-type: none"> <li>在线更正不正确的值，并视需要重新启动命令</li> </ul>	

## 12.8.7 ErrorID 16#8FF

## 内部错误

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8FFF</b>		内部错误		
	16#F0**	-	<p>CPU 的断电和上电</p> <p>如果这样做无效，则联系客户支持。准备好以下信息：</p> <ul style="list-style-type: none"> <li>ErrorID</li> <li>ErrorInfo</li> <li>诊断缓冲区条目</li> </ul>	

## 12.9 定位轴工艺对象 V6...7 的变量

### 12.9.1 图例

变量	变量的名称																				
数据类型	变量的数据类型																				
值	起始值 (变量的值范围 - 最小值到最大值) 如果未显示特定值，则应用相应数据类型的值范围限制或“说明”下面的信息。																				
访问	<p>在用户程序中访问该变量：</p> <table> <tr> <td>OPR</td><td>Openness 应用程序可对该变量进行读访问。</td></tr> <tr> <td>OPRW</td><td>Openness 应用程序可对该变量进行读写访问。</td></tr> <tr> <td>R</td><td>在用户程序和 HMI 中，可读取该变量。</td></tr> <tr> <td>RCCP</td><td>该变量可在用户程序和 HMI 中读取，并在每个周期控制点处进行更新。</td></tr> <tr> <td>RP</td><td>该变量可通过运动控制指令“MC_ReadParam”进行读取。相应变量的当前值将在命令启动时确定。</td></tr> <tr> <td>RW</td><td>在用户程序和 HMI 中，可读写该变量。该变量可通过运动控制指令“MC_WriteParam”进行写入。</td></tr> <tr> <td>WP</td><td>与驱动装置连接无关：如果该轴已禁用 (MC_Power.Status = FALSE)，则可使用运动控制指令“MC_WriteParam”对该变量进行写入操作。</td></tr> <tr> <td>WP_PD</td><td>通过 PROFIdrive/模拟量输出进行驱动装置连接：如果该轴已禁用 (MC_Power.Status = FALSE)，则可使用运动控制指令“MC_WriteParam”对该变量进行写入操作。</td></tr> <tr> <td>WP_PT_O</td><td>通过 PTO 进行驱动装置连接：如果该轴已禁用 (MC_Power.Status = FALSE)，则可使用运动控制指令“MC_WriteParam”对该变量进行写入操作。</td></tr> <tr> <td>-</td><td>该变量不能在用户程序中使用。</td></tr> </table>	OPR	Openness 应用程序可对该变量进行读访问。	OPRW	Openness 应用程序可对该变量进行读写访问。	R	在用户程序和 HMI 中，可读取该变量。	RCCP	该变量可在用户程序和 HMI 中读取，并在每个周期控制点处进行更新。	RP	该变量可通过运动控制指令“MC_ReadParam”进行读取。相应变量的当前值将在命令启动时确定。	RW	在用户程序和 HMI 中，可读写该变量。该变量可通过运动控制指令“MC_WriteParam”进行写入。	WP	与驱动装置连接无关：如果该轴已禁用 (MC_Power.Status = FALSE)，则可使用运动控制指令“MC_WriteParam”对该变量进行写入操作。	WP_PD	通过 PROFIdrive/模拟量输出进行驱动装置连接：如果该轴已禁用 (MC_Power.Status = FALSE)，则可使用运动控制指令“MC_WriteParam”对该变量进行写入操作。	WP_PT_O	通过 PTO 进行驱动装置连接：如果该轴已禁用 (MC_Power.Status = FALSE)，则可使用运动控制指令“MC_WriteParam”对该变量进行写入操作。	-	该变量不能在用户程序中使用。
OPR	Openness 应用程序可对该变量进行读访问。																				
OPRW	Openness 应用程序可对该变量进行读写访问。																				
R	在用户程序和 HMI 中，可读取该变量。																				
RCCP	该变量可在用户程序和 HMI 中读取，并在每个周期控制点处进行更新。																				
RP	该变量可通过运动控制指令“MC_ReadParam”进行读取。相应变量的当前值将在命令启动时确定。																				
RW	在用户程序和 HMI 中，可读写该变量。该变量可通过运动控制指令“MC_WriteParam”进行写入。																				
WP	与驱动装置连接无关：如果该轴已禁用 (MC_Power.Status = FALSE)，则可使用运动控制指令“MC_WriteParam”对该变量进行写入操作。																				
WP_PD	通过 PROFIdrive/模拟量输出进行驱动装置连接：如果该轴已禁用 (MC_Power.Status = FALSE)，则可使用运动控制指令“MC_WriteParam”对该变量进行写入操作。																				
WP_PT_O	通过 PTO 进行驱动装置连接：如果该轴已禁用 (MC_Power.Status = FALSE)，则可使用运动控制指令“MC_WriteParam”对该变量进行写入操作。																				
-	该变量不能在用户程序中使用。																				

## 12.9 定位轴工艺对象 V6...7 的变量

W	工艺数据块中更改的有效性	
	1	通过 PTO 进行驱动装置连接：在轴激活、禁用或已启用时
	2	通过 PTO 进行驱动装置连接：轴启用时
	5	通过 PTO 进行驱动装置连接：下一次启动 MC_MoveAbsolute、MC_MoveRelative、MC_MoveVelocity、MC_MoveJog、MC_Halt、MC_CommandTable 或激活的 MC_Home 命令时 (Mode = 3)
	6	通过 PTO 进行驱动装置连接：MC_MoveJog 命令停止时
	7	通过 PTO 进行驱动装置连接：启动被动回原地命令时
	8	通过 PTO 进行驱动装置连接：启动主动回原地命令时
	9	重新启动工艺对象时
	10	通过 PROFIdrive/模拟量输出进行驱动装置连接：下一次调用 MC-Interpolator [OB92] 时
	说明	变量的说明

通过“<TO>.<变量名称>”的形式访问变量。占位符 <TO> 代表工艺对象的名称。

#### 说明

##### 使用“WRIT\_DB”保存更改

CPU 掉电、启停或工艺对象重启时，在运行期间对工艺对象组态变量所做的更改将会丢失。

如果想要在 CPU 掉电、启停或者工艺对象重启之后，对工艺数据块所做的更改仍然保留，则必须使用扩展指令“WRIT\_DB”将这些更改写入装载存储器的起始值中。

#### 说明

##### 无法在 RAM 中更改的带“WRIT\_DB”的变量更改

工艺对象的某些组态变量无法在运行期间更改。这些变量不能直接写入（数据块访问），也不能使用 MC\_WriteParam 进行更改。

要更改这些组态变量，请使用扩展指令“WRIT\_DB”更改装载存储器中的值。

使用 MC\_Reset（重新启动 = TRUE）重启工艺对象以启用更改。

#### 说明

##### 使用“READ\_DB”访问

只能通过工艺数据块中的指令 READ\_DB 将标识为 RW 的参数从装载存储器应用到工作存储器。

## 12.9.2 位置值和速度值变量 V6 及以上版本

该变量结构中包含有轴的位置与速度的设定值和实际值。

### 变量

图例 (页 335)

变量	数据类型	值	访问	W	说明
Position	REAL	0.0 (-9.0E15 到 9.0E15)	RCCP、 PR、 ORP	-	轴的位置设定值 (用组态的测量单位表示) 如果轴未归位，则该变量指示的是相对于轴启用位置的位置值。 Openness 中的名称: Position
Velocity	REAL	0.0	RCCP、 PR、 ORP	-	轴的速度设定值 (用组态的测量单位表示) Openness 中的名称: Velocity
ActualPosition	REAL	0.0 (-9.0E15 到 9.0E15)	RCCP、 PR、 ORP	-	轴的实际位置 (用组态的测量单位表示) 如果轴未回原点，则该变量将指示相对于轴启用位置的位置值。 Openness 中的名称: ActualPosition
ActualVelocity	REAL	0.0 (-9.0E15 到 9.0E15)	RCCP、 PR、 ORP	-	轴的实际速度 (用组态的测量单位表示) Openness 中的名称: ActualVelocity

### 参见

运动状态 (页 225)

定位轴工艺对象 V4...5 的变量 (页 485)

轴工艺对象 V1...3 的变量 (页 472)

## 12.9 定位轴工艺对象 V6...7 的变量

### 12.9.3 仿真变量 (V6 及以上版本)

变量结构 <轴名称>.Simulation.Mode 中包含有仿真模式。

#### 变量

图例 (页 335)

变量	数据类型	值	访问	W	说明
Simulation.					
Mode	UDINT	0 (0 到 1)	R OPRW	- 2 、 9	仿真模式  Openness 中的名称: Simulation.Mode
				0	无仿真, 正常运行
				1	仿真模式  在仿真模式下, 无需实际驱动器即可在 CPU 中仿真轴。

#### 参见

定位轴工艺对象 V4...5 的变量 (页 485)

轴工艺对象 V1...3 的变量 (页 472)

## 12.9.4 执行器变量 (V6 及以上版本)

变量结构 <轴名称>.Actor.<变量名称> 中包含有驱动装置参数。

### 变量

图例 (页 335)

变量	数据类型	值	访问	W	说明
Actor.	STRUCT				TO_Struct_Actor
Type	DINT	2 (0 到 2)	R OPRW	-	Openness 中的名称: <b>Actor.Type</b> 定位轴工艺对象 V5 及更高版本: 0      0 = 驱动装置通过一个模拟量输出进行连接。轴的所有运动均在位置控制下进行。 1      驱动装置通过 PROFIdrive 报文进行连接。轴的所有运动均在位置控制下进行。 2      驱动装置通过一个脉冲接口进行连接。 定位轴工艺对象 V4: 2      驱动装置通过一个脉冲接口进行连接。
InverseDirection	BOOL	FALSE	R WP_PT O OPRW	- 2	Openness 中的名称: <b>Actor.InverseDirection</b> FALSE    方向未反转。 TRUE     方向已反转。
DirectionMode	INT	0 (0 到 2)	R WP_PT O OPRW	- 2	允许的旋转方向 Openness 中的名称: <b>Actor.DirectionMode</b> 0      双向 1      正方向 2      负方向

## 12.9 定位轴工艺对象 V6...7 的变量

变量	数据类型	值	访问	W	说明
DataAdaption	DINT	0 (0 到 1)	R OPRW	-	Openness 中的名称: Actor.DataAdaption
					0 清除“运行时自动应用驱动值（在线）”(Automatically apply drive values at runtime (online)) 复选框。
					1 选中“运行时自动应用驱动值（在线）”(Automatically apply drive values at runtime (online)) 复选框。
Actor.DataAdaptionOffline*	BOOL	-	OPRW	-	(V7 或更高版本) Opennes 中的名称: Actor.DataAdaptionOffline
					FALS E 清除选项“组态时自动应用驱动值（离线）”(Automatically apply drive values during configuration (offline))。
					TRUE 选中选项“组态时自动应用驱动值（离线）”(Automatically apply drive values during configuration (offline))。
PTOSliceTime	INT	2...20	R WP_PT O OPRW	2	PTO（脉冲串输出）的段时间，单位为 ms (V7 或更高版本) 运动控制命令分为多段，不能在段末尾更改。 段时间是段的持续时间，因此会影响运动控制指令的超驰响应。 Openness 中的名称: Actor.PTOSliceTime
Interface.	STRUCT				TO_Struct_ActorInterface
	AddressIn.	VREF	-	-	PROFIdrive 报文的输入地址（内部参数）
	AREA	BYTE	-	OPR	- Openness 中的名称: Actor.Interface.AddressIn.AREA
	DB_NUMBR	UINT	-	OPR	- Openness 中的名称: Actor.Interface.AddressIn.DB_NUMBER

变量	数据类型	值	访问	W	说明
OFFSET	UDINT	-	OPR	-	Openness 中的名称: Actor.Interface.AddressIn.OFFSET
RID	DWORD	-	OPR	-	Openness 中的名称: Actor.Interface.AddressIn.RID
ProfiDriveIn*	STRING	-	OPRW	-	Openness 中的名称: _Actor.Interface.ProfiDriveIn 有效输入地址（包含在报文中） 有效变量名
AddressOut.	VREF	-	-	-	PROFIdrive 报文的输出地址（内部参数）
AREA	BYTE	-	OPR	-	Openness 中的名称: Actor.Interface.AddressOut.AREA
DB_NUMBR	UINT	-	OPR	-	Openness 中的名称: Actor.Interface.AddressOut.DB_NUMBER
OFFSET	UDINT	-	OPR	-	Openness 中的名称: Actor.Interface.AddressOut.OFFSET
RID	DWORD	-	OPR	-	Openness 中的名称: Actor.Interface.AddressOut.RID
ProfiDriveOut*	STRING	-	OPRW	-	Openness 中的名称: _Actor.Interface.ProfiDriveOut 有效输出地址（包含在报文中） 有效变量名
DataBlock*	STRING	-	OPRW	-	Openness 中的名称: _Actor.Interface.DataBlock 有效数据块地址:
Analog*	STRING	-	OPRW	-	Openness 中的名称: _Actor.Interface.Analog 有效模拟量输出、有效数据块地址、有效变量名
DataConnection*	INT	0 (0 到 1)	OPRW	-	Openness 中的名称: _Actor.Interface.DataConnection
		0		驱动装置	
		1		数据块	

## 12.9 定位轴工艺对象 V6...7 的变量

变量	数据类型	值	访问	W	说明
EnableDriveOutput	VREF	-	-	-	使能输出（内部参数）
	AREA	BYTE	-	OPR	Openness 中的名称： Actor.Interface.DriveReadyOutput.AREA
	DB_NUMBR	UINT	-	OPR	Openness 中的名称： Actor.Interface.DriveReadyOutput.DB_NUMBE R
	OFFSET	UDINT	-	OPR	Openness 中的名称： Actor.Interface.DriveReadyOutput.OFFSET
	RID	DWORD	-	OPR	Openness 中的名称： Actor.Interface.DriveReadyOutput.RID
	EnableDriveOutput*	STRIN G	-	OPRW	Openness 中的名称： _Actor.Interface.EnableDriveOutput 有效输入、有效输出、有效存储器地址、有效 变量名
	DriveReadyIn put	VREF	-	-	输入就绪（内部参数）
	AREA	BYTE	-	OPR	Openness 中的名称： Actor.Interface.DriveReadyInput.AREA
	DB_NUMBR	UINT	-	OPR	Openness 中的名称： Actor.Interface.DriveReadyInput.DB_NUMBER
	OFFSET	UDINT	-	OPR	Openness 中的名称： Actor.Interface.DriveReadyInput.OFFSET
	RID	DWORD	-	OPR	Openness 中的名称： Actor.Interface.DriveReadyInput.RID
	DriveReadyIn put*	STRIN G	-	OPRW	Openness 中的名称： _Actor.Interface.DriveReadyInput 有效输入、有效输出、有效存储器地址、有效 变量名
	PTO	DWORD	0	OPR	脉冲输出（内部参数）

变量	数据类型	值	访问	W	说明
PTO*	STRIN G	-	OPRW	-	Openness 中的名称: <code>_Actor.Interface.PTO</code>
					0   Pulse_1
					1   Pulse_2
					2   Pulse_3
					3   Pulse_4
PTO_OutputA *	STRIN G	-	OPRW	-	Openness 中的名称: <code>_Actor.Interface.PTO_OutputA</code> 有效输入地址、有效变量名 仅支持板载 CPU 或信号板地址。
PTO_OutputB Enable*	BOOL		OPRW	-	Openness 中的名称: <code>_Actor.Interface.PTO_OutputBEnable</code> 仅 <code>PTO_SignalType = 2</code> 时可用
					FALS E   禁用输出 B。
					TRUE   启用输出 B。
PTO_OutputB *	STRIN G	-	OPRW	-	Openness 中的名称: <code>_Actor.Interface.PTO_OutputB</code> 有效输入地址、有效变量名 仅支持板载 CPU 或信号板地址。
PTO_SignalT ype*	INT	(2 到 5)	OPRW	-	Openness 中的名称: <code>_Actor.Interface.PTO_SignalType</code>
					2   脉冲 A 和方向 B
					3   时钟增加 A 和时钟减少 B
					4   A/B 相移
					5   A/B 相移, 四相位
DriveParameter.	STRUCT				TO_Struct_ActorDriveParameter

## 12.9 定位轴工艺对象 V6...7 的变量

变量	数据类型	值	访问	W	说明
ReferenceSpeed	REAL	3000.0	R OPRW	-	<p>驱动装置速度设定值 (NSET) 的参考值 (100 %)            速度设定值在 PROFIdrive 报文以从 -200% 到 200% “ReferenceSpeed”的标准值的形式来传输。</p> <p>通过模拟量输出指定设定值时，只要驱动装置允许，模拟量输出的操作范围可为 -117 % 到 117 %。</p> <p>Openness 中的名称：  <b>Actor.DriveParameter.ReferenceSpeed</b></p>
MaxSpeed	REAL	3000.0	R OPRW	-	<p>驱动装置速度设定值 (NSET) 的最大值            (PROFIdrive: MaxSpeed <math>\leq 2 \times</math> ReferenceSpeed)</p> <p>模拟量设定值： MaxSpeed <math>\leq 1.17 \times</math> ReferenceSpeed)</p> <p>Openness 中的名称：  <b>Actor.DriveParameter.MaxSpeed</b></p>
PulsesPerDriveRevolution	DINT	1000 (1 到 214748364 8)	R WP_PT O OPRW	2	<p>电机每转的脉冲数</p> <p>Openness 中的名称：  <b>Actor.DriveParameter.PulsesPerDriveRevolution</b></p>

\*) 适用于 Openness

## 参见

定位轴工艺对象 V4...5 的变量 (页 485)

轴工艺对象 V1...3 的变量 (页 472)

### 12.9.5 Sensor[1] 变量 (V6 及以上版本)

变量结构 <轴名称>.Sensor[1].<变量名称> 中包含有编码器参数。

#### 变量

图例 (页 335)

变量	数据类型	值	访问	W	说明
Sensor[1].	STRUCT				ARRAY[1..1] TO_Struct_Sensor
Type	DINT	0 (0 到 1)	R、 OPRW	-	编码器类型 (内部参数) Openness 中的名称: Sensor[1].Type
					0 增量式
					1 绝对式
InverseDirection	BOOL	FALSE	R、 OPRW	-	对实际值取反 Openness 中的名称: Sensor[1].InverseDirection
					FALSE 实际值不取反
					TRUE 实际值已取反
System	DINT	1 (0 到 1)	R、 OPRW	-	编码器系统 Openness 中的名称: Sensor[1].System
					0 线性编码器
					1 旋转编码器
MountingMode	DINT	0 (0 到 2)	R、 OPRW	-	编码器的安装方式 Openness 中的名称: Sensor[1].MountingMode
					0 在电机轴上
					2 外部测量系统

## 12.9 定位轴工艺对象 V6...7 的变量

变量	数据类型	值	访问	W	说明
DataAdaption	DINT	0 (0 到 1)	R、 OPRW	-	Openness 中的名称: <code>Sensor[1].DataAdaption</code>
					0 禁用复选框“运行时自动应用编码器值（在线）”(Automatically apply drive values at runtime (online))。
					1 启用复选框“运行时自动应用编码器值（在线）”(Automatically apply drive values at runtime (online))。
Sensor.DataAdaptionOffline*	BOOL	-	OPRW	-	(V7 或更高版本) Openness 中的名称: <code>_Sensor.DataAdaptionOffline</code>
					FALSE 清除“组态时自动应用编码器值（离线）”(Automatically apply encoder values during configuration (offline)) 选项。
					TRUE 选择“组态时自动应用编码器值（离线）”(Automatically apply encoder values during configuration (offline)) 选项。
Interface.	STRUCT				TO_Struct_SensorInterface
Type	DINT	4 (0 到 4)	OPR	-	编码器连接（内部参数） Openness 中的名称: <code>Sensor[1].Interface.Type</code>
					0 PROFINET 上的 PROFIdrive 编码器
					1 工艺模块 (TM) 上的编码器
					2 驱动装置上的编码器
					4 高速计数器上的编码器
AddressIn.	VREF	-	-	-	PROFINET 报文的输入地址（内部参数）
	AREA	BYTE	-	OPR	内部参数 Openness 中的名称: <code>Sensor[1].Interface.AddressIn.AREA</code>

变量	数据类型	值	访问	W	说明
	DB_NUMBER	UINT	-	OPR	- 内部参数 Openess 中的名称: Sensor[1].Interface.AddressIn.DB_NUMBER
	OFFSET	UDINT	-	OPR	- 内部参数 Openess 中的名称: Sensor[1].Interface.AddressIn.OFFSET
	RID	DWORD	-	OPR	- 内部参数 Openess 中的名称: Sensor[1].Interface.AddressIn.RID
ProfiDriveIn*	STRING	-	OPRW	-	Openess 中的名称: _Sensor[1].Interface.ProfiDriveIn 有效输入地址（包含在报文中） 有效变量名
AddressOut.	VREF	-	-	-	PROFIdrive 报文的输出地址（内部参数）
AREA	BYTE	-	OPR	-	内部参数 Openess 中的名称: Sensor[1].Interface.AddressOut.AREA
DB_NUMBER	UINT	-	OPR	-	内部参数 Openess 中的名称: Sensor[1].Interface.AddressOut.DB_NUMBER
OFFSET	UDINT	-	OPR	-	内部参数 Openess 中的名称: Sensor[1].Interface.AddressOut.OFFSET
RID	DWORD	-	OPR	-	内部参数 Openess 中的名称: Sensor[1].Interface.AddressOut.RID
ProfiDriveOut*	STRING	-	OPRW	-	Openess 中的名称: _Sensor[1].Interface.ProfiDriveOut 有效输出地址（包含在报文中） 有效变量名

## 12.9 定位轴工艺对象 V6...7 的变量

变量	数据类型	值	访问	W	说明
DataBlock*	STRING	-	OPRW	-	Openness 中的名称: _Sensor[1].Interface.DataBlock 有效数据块地址:
DataConnection*	UDINT	(0 到 1)	OPRW	-	Openness 中的名称: _Sensor[1].Interface.DataConnection
					0 编码器
					1 数据块
EncoderConnection*	INT	(4 到 7)	OPRW	-	Openness 中的名称: _Sensor[1].Interface.EncoderConnection
					4 高速计数器 (HSC) 上的编码器
					7 PROFIBUS/PROFINET 上的编码器
Number	UDINT	1	OPRW	-	编码器编号 Openness 中的名称: Sensor[1].Interface.Number
HSC	DWORD	0	OPR	-	编码器传送实际值的高速计数器 (内部参数)
HSC*	STRING	-	OPRW	-	Openness 中的名称: _Sensor[1].Interface.HSC 硬件配置中快速计数器的名称。
HSC_OperatingMode*	INT	(1 到 3)	OPRW	-	Openness 中的名称: _Sensor[1].Interface.HSC_OperatingMode
					1 双相
					2 A/B 计数器
					3 A/B 计数器, 四相位
HSC_InputA*	STRING	-	OPRW	-	Openness 中的名称: _Sensor[1].Interface.HSC_InputA 有效输入地址、有效变量名
HSC.InputB*	STRING	-	OPRW	-	Openness 中的名称: _Sensor[1].Interface.HSC.InputB 有效输入地址、有效变量名

变量	数据类型	值	访问	W	说明
Parameter.	STRUCT				TO_Struct_SensorParameter
Resolution	REAL	0.001 (-1.0E12 到 1.0E12)	R、 OPRW	-	线性编码器的精度（两个编码器脉冲之间的偏移值） Openness 中的名称: Sensor[1].Parameter.Resolution
StepsPerRevolution	UDINT	2048 (1 到 8388608)	R、 OPRW	-	编码器每转的增量数 Openness 中的名称: Sensor[1].Parameter.StepsPerRevolution
FineResolutionXist1	UDINT	11 (0 到 31)	R、 OPRW	-	高精度的位数 Gx_XIST1 (周期性实际编码器值) Openness 中的名称: Sensor[1].Parameter.FineResolutionXist1
FineResolutionXist2	UDINT	9 (0 到 31)	R、 OPRW	-	高精度的位数 Gx_XIST2 (编码器的绝对值) Openness 中的名称: Sensor[1].Parameter.FineResolutionXist2
DeterminableRevolutions	UDINT	1 (0 到 8388608)	R、 OPRW	-	多圈绝对值编码器的差动转数 Openness 中的名称: Sensor[1].Parameter.DeterminableRevolutions
DistancePerRevolution	REAL	100.0 (0.0 到 1.0E12)	R、 OPRW	-	0   增量式编码器
					1   单回绝对值编码器
BehaviorGx_XIST1	DINT	1 (0 到 1)	R、 OPRW	-	估算位 Gx_XIST1 (V7 或更高版本) 0   基于编码器分辨率占用的位数。 1   估算编码器值的所有 32 位
ActiveHoming.	STRUCT				TO_Struct_SensorActiveHoming

## 12.9 定位轴工艺对象 V6...7 的变量

变量	数据类型	值	访问	W	说明
Mode	DINT	2 (0 到 2)	R、 WP_PT O、 OPRW	- 2	主动回原点模式  Openness 中的名称: Sensor[1].ActiveHoming.Mode  定位轴工艺对象 V5 及更高版本:  0 基于 PROFIdrive 报文的零位标记 (非 PTO) 1 基于 PROFIdrive 报文和接近开关的零位标记 (非 PTO) 2 通过数字量输入回原点  定位轴工艺对象 V4:  2 通过数字量输入回原点
SideInput	BOOL	FALSE	RW、 WP、 OPRW	1 、 8 、 10	主动回原点过程中, 轴回原点时所到达的回原点开关端  Openness 中的名称: Sensor[1].ActiveHoming.SideInput  FALSE 下侧 TRUE 上侧
DigitalInputAddress.	VREF	-	-	-	回原点开关的符号化输入地址 (内部参数)
AREA	BYTE	-	OPR	-	内部参数  Openness 中的名称: Sensor[1].ActiveHoming.DigitalInputAddress.AREA
DB_NUMBER	UINT	-	OPR	-	内部参数  Openness 中的名称: Sensor[1].ActiveHoming.DigitalInputAddress.DB_NUMBER
OFFSET	UDINT	-	OPR	-	内部参数  Openness 中的名称: Sensor[1].ActiveHoming.DigitalInputAddress.OFFSET

变量	数据类型	值	访问	W	说明
	RID	DWORD	-	OPR	- 内部参数 Openness 中的名称: Sensor[1].ActiveHoming.DigitalInputAddress.RID
	DigitalInput*	STRING	-	OPRW	- Openness 中的名称: _Sensor[1].ActiveHoming.DigitalInput 有效输入地址、有效变量名
	HomePositionOffset	REAL	0.0 (-1.0E12 到 1.0E12)	RW、WP、OPRW	1 、 8 、 10 原地位置偏移值（主动回原点） (用组态的测量单位表示) Openness 中的名称: Sensor[1].ActiveHoming.HomePositionOffset
	SwitchLevel	BOOL	TRUE	RW、WP、OPRW	1 、 8 、 10 选择到达回原点开关时 CPU 输入端的信号电平 Openness 中的名称: Sensor[1].ActiveHoming.SwitchLevel
				FALSE	低电平（低电平有效）
				TRUE	高电平（高启用）
PassiveHoming.	STRUCT				TO_Struct_SensorPassiveHoming
DigitalInputAddress.	VREF	-	-	-	回原点开关的符号化输入地址（内部参数）
	AREA	BYTE	-	OPR	- 内部参数 Openness 中的名称: Sensor[1].PassiveHoming.DigitalInputAddress.AREA
	DB_NUMBER	UINT	-	OPR	- 内部参数 Openness 中的名称: Sensor[1].PassiveHoming.DigitalInputAddress.DB_NUMBER

## 12.9 定位轴工艺对象 V6...7 的变量

变量	数据类型	值	访问	W	说明
	OFFSET	UDINT	-	OPR	- 内部参数 Openness 中的名称: Sensor[1].PassiveHoming.DigitalInputAddress. OFFSET
	RID	DWORD	-	OPR	- 内部参数 Openness 中的名称: Sensor[1].PassiveHoming.DigitalInputAddress. RID
Mode	DINT	2 (0 到 2)	R WP_PT O OPRW	- 2	被动回原点模式 Openness 中的名称: Sensor[1].PassiveHoming.Mode  定位轴工艺对象 V5 及更高版本:  0 基于 PROFIdrive 报文的零位标记 (非 PTO) 1 基于 PROFIdrive 报文和接近开关的零位标记 (非 PTO) 2 通过数字量输入回原点  定位轴工艺对象 V4:  2 通过数字量输入回原点
SideInput	BOOL	FALSE	RW、 WP、 OPRW	1 、 7 、 10	被动回原点过程中, 轴回原点时所到达的回原点开关端 Openness 中的名称: Sensor[1].PassiveHoming.SideInput  FALSE 下侧 TRUE 上侧
DigitalInputAddress.	VREF	-	-	-	回原点开关的符号化输入地址 (内部参数)
AREA	BYTE	-	OPR	-	内部参数 Openness 中的名称: Sensor[1].PassiveHoming.DigitalInputAddress. AREA

变量	数据类型	值	访问	W	说明
	DB_NUMBER	UINT	-	OPR	内部参数 Openness 中的名称: Sensor[1].PassiveHoming.DigitalInputAddress. DB_NUMBER
	OFFSET	UDINT	-	OPR	内部参数 Openness 中的名称: Sensor[1].PassiveHoming.DigitalInputAddress. OFFSET
	RID	DWORD	-	OPR	内部参数 Openness 中的名称: Sensor[1].PassiveHoming.DigitalInputAddress. RID
DigitalInput*	STRING		OPRW	-	Openness 中的名称: _Sensor[1].PassiveHoming.DigitalInput 有效输入地址、有效变量名
SwitchLevel	BOOL	TRUE	RW、 WP、 OPRW	1 、 7 、 10	选择到达回原点开关时 CPU 输入端的电平 Openness 中的名称: Sensor[1].PassiveHoming.SwitchLevel
				FALSE	低电平（低电平有效）
				TRUE	高电平（高启用）

\*) 适用于 Openness

## 参见

定位轴工艺对象 V4...5 的变量 (页 485)

轴工艺对象 V1...3 的变量 (页 472)

## 12.9 定位轴工艺对象 V6...7 的变量

### 12.9.6 单位变量 (V6 及以上版本)

变量结构 <轴名称>.Units.LengthUnit 中包含有为参数组态的测量单位。

#### 变量

图例 (页 335)

变量	数据类型	值	访问	W	说明
Units.	STRUCT				TO_Struct_Units
LengthUnit	INT	1013 (-32768 到 32767)	R WP_PT O OPRW	-2	为参数组态的测量单位 Openness 中的名称: Units.LengthUnit
					-1 脉冲
					1005 °
					1010 m
					1013 mm
					1018 ft
					1019 in

#### 参见

定位轴工艺对象 V4...5 的变量 (页 485)

轴工艺对象 V1...3 的变量 (页 472)

## 12.9.7 机械变量 (V6 及以上版本)

变量结构 <轴名称>.Mechanics.LeadScrew 中包含有电机每转的行进距离。

### 变量

图例 (页 335)

变量	数据类型	值	访问	W	说明
Mechanics.	STRUCT				TO_Struct_Mechanics
LeadScrew	REAL	10.0 (-1.0E12 到 1.0E12)	R、 WP_PT O、 OPRW	-	每转的距离 (用组态的测量单位表示) Openness 中的名称: Mechanics.LeadScrew

### 参见

定位轴工艺对象 V4...5 的变量 (页 485)

轴工艺对象 V1...3 的变量 (页 472)

## 12.9 定位轴工艺对象 V6...7 的变量

### 12.9.8 Modulo 变量 (V6 及以上版本)

变量结构 <轴名称>.Modulo.<变量名称> 中包含有模数设置。

#### 变量

图例 (页 335)

变量	数据类型	值	访问	W	说明
Modulo.	STRUCT				TO_Struct_Modulo
Enable	BOOL	FALSE	R、OPRW	-	Openness 中的名称: Modulo.Enable
					FALSE 模数转换已禁用
					TRUE 模数转换已启用
					启用模数转换时，应对模数长度 > 0.0 进行检查
Length	REAL	360.0 (0.001 到 1.0E12)	R、 OPRW	-	模数长度 Openness 中的名称: Modulo.Length
StartValue	REAL	0.0 (-1.0E12 到 1.0E12)	R、 OPRW	-	模数起始值 Openness 中的名称: Modulo.StartValue

#### 参见

定位轴工艺对象 V4...5 的变量 (页 485)

轴工艺对象 V1...3 的变量 (页 472)

### 12.9.9 DynamicLimits 变量 (V6 及以上版本)

变量结构 <轴名称>.DynamicLimits.<变量名称> 中包含有动态限值组态。

#### 变量

图例 (页 335)

变量	数据类型	值	访问	W	说明
DynamicLimits.	STRUCT				TO_Struct_DynamicLimits
MaxVelocity	REAL	250.0	R WP_PT O OPRW	- 2	轴的最大速度 (用组态的测量单位表示) Openness 中的名称: DynamicLimits.MaxVelocity
MinVelocity	REAL	10.0	R WP_PT O OPRW	- 2	轴的启动/停止速度 (用组态的测量单位表示) Openness 中的名称: DynamicLimits.MinVelocity

#### 参见

定位轴工艺对象 V4...5 的变量 (页 485)

轴工艺对象 V1...3 的变量 (页 472)

## 12.9 定位轴工艺对象 V6...7 的变量

### 12.9.10 DynamicDefaults (V6 及以上版本)

变量结构 <轴名称>.DynamicDefaults.<变量名称> 中包含有动态默认值组态。

#### 变量

图例 (页 335)

变量	数据类型	值	访问	W	说明
DynamicDefaults.	STRUCT				TO_Struct_DynamicDefaults
Acceleration	REAL	48.0 (0.0 到 1.0E12)	RW、 WP、 OPRW	5 、 6 、 10	轴加速度的默认设置 (用组态的测量单位表示)  Openness 中的名称: DynamicDefaults.Acceleration
Deceleration	REAL	48.0 (0.0 到 1.0E12)	RW、 WP、 OPRW	5 、 6 、 10	轴的默认减速速度 (用组态的测量单位表示)  Openness 中的名称: DynamicDefaults.Deceleration
Jerk	REAL	192.0 (0.0 到 1.0E12)	RW、 WP、 OPRW	5 、 10	轴斜坡加速和减速过程中的加加速度默认设置 (用组态的测量单位表示)  所组态的加加速度大于 0.00004 mm/s <sup>2</sup> 时, 激活加加速度。  Openness 中的名称: DynamicDefaults.Jerk
EmergencyDeceleration	REAL	120.0 (0.0 到 1.0E12)	RW、 WP、 OPRW	1 、 5 、 6 、 10	轴的急停减速速度 (用组态的测量单位表示)  Openness 中的名称: DynamicDefaults.EmergencyDeceleration

## 参见

定位轴工艺对象 V4...5 的变量 (页 485)

轴工艺对象 V1...3 的变量 (页 472)

### 12.9.11 PositionLimits\_SW 变量 (V6 及以上版本)

变量结构 <轴名称>.PositionLimits\_SW.<变量名称> 中包含有软限位开关的定位监视组态。软限位开关用于限制定位轴的操作范围。

## 变量

图例 (页 335)

变量	数据类型	值	访问	W	说明	
PositionLimits_SW	STRUCT				TO_Struct_PositionLimitsSW	
Active	BOOL	FALSE	RW、WP、OPRW	1 、 5 、 6 、 10	Openness 中的名称: PositionLimits_SW.Active	
					FALSE	软限位开关取消激活。
					TRUE	软限位开关激活。
MinPosition	REAL	-10000.0 (-1.0E12 到 1.0E12)	RW、WP、OPRW	1 、 5 、 6 、 10	软限位开关下限的位置 (用组态的测量单位表示) Openness 中的名称: PositionLimits_SW.MinPosition	

---

## 12.9 定位轴工艺对象 V6...7 的变量

变量	数据类型	值	访问	W	说明
MaxPosition	REAL	10000.0 (-1.0E12 到 1.0E12)	RW、 WP、 OPRW	1 、 5 、 6 、 10	软限位开关的上限位置 (用组态的测量单位表示) Openness 中的名称: PositionLimits_SW.MaxPosition

### 参见

[定位轴工艺对象 V4...5 的变量 \(页 485\)](#)

[轴工艺对象 V1...3 的变量 \(页 472\)](#)

### 12.9.12 PositionLimits\_HW 变量 (V6 及以上版本)

变量结构 <轴名称>.PositionLimits\_HW.<变量名称> 中包含有硬限位开关的定位监视组态。硬限位开关用于限制定位轴的行进范围。

#### 变量

图例 (页 335)

变量	数据类型	值	访问	W	说明
PositionLimits_HW.	STRUCT				TO_Struct_PositionLimitsHW
Active	BOOL	FALSE	RW、WP、OPRW	1 、 5 、 6 、 10	Openness 中的名称: PositionLimits_HW.Active FALS E 硬限位开关取消激活。 TRUE 硬限位开关激活。
MinSwitchLevel	BOOL	FALSE	RW WP_PT O	- 2	选择到达硬限位开关下限时 CPU 输入端的信号电平 Openness 中的名称: PositionLimits_HW.MinSwitchLevel FALS E 低电平 (低电平有效) TRUE 高电平 (高启用)
MinSwitchAddress.	VREF	-	-	-	硬限位开关下限的符号化输入地址 (内部参数)
AREA	BYTE	-	OPR	-	Openness 中的名称: PositionLimits_HW.MinSwitchAddress.AREA
DB_NUMBER	USHORT	-	OPR	-	Openness 中的名称: PositionLimits_HW.MinSwitchAddress.DB_NUMBER
OFFSET	UINT	-	OPR	-	Openness 中的名称: PositionLimits_HW.MinSwitchAddress.OFFSET

## 12.9 定位轴工艺对象 V6...7 的变量

变量	数据类型	值	访问	W	说明
RID	UINT	-	OPR	-	Openness 中的名称: PositionLimits_HW.MinSwitchAddress.RID
MinSwitch*	STRING	-	OPRW	-	Openness 中的名称: _PositionLimits_HW.MinSwitch 有效输入地址、有效变量名
MaxSwitchLevel	BOOL	FALSE	RW WP_PT O	- 2	选择到达硬限位开关上限时 CPU 输入端的信号电平 Openness 中的名称: PositionLimits_HW.MaxSwitchLevel
				FALSE	低电平（低电平有效）
				TRUE	高电平（高启用）
MaxSwitchAddress.	VREF	-	-	-	硬限位开关上限的输入地址（内部参数）
AREA	BYTE	-	OPR	-	Openness 中的名称: PositionLimits_HW.MaxSwitchAddress.AREA
DB_NUMBER	USHORT	-	OPR	-	Openness 中的名称: PositionLimits_HW.MaxSwitchAddress.DB_NUMBER
OFFSET	UINT	-	OPR	-	Openness 中的名称: PositionLimits_HW.MaxSwitchAddress.OFFSET
RID	UINT	-	OPR	-	Openness 中的名称: PositionLimits_HW.MaxSwitchAddress.RID
MaxSwitch*	STRING	-	OPRW	-	Openness 中的名称: _PositionLimits_HW.MaxSwitch 有效输入地址、有效变量名

\*) 适用于 Openness

**参见**

定位轴工艺对象 V4...5 的变量 (页 485)

轴工艺对象 V1...3 的变量 (页 472)

## 12.9 定位轴工艺对象 V6...7 的变量

## 12.9.13 归位变量 (V6 及以上版本)

变量结构 <轴名称>.Homing.<变量名称> 中包含有轴回原点的组态。

## 变量

图例 (页 335)

变量	数据类型	值	访问	W	说明	
Homing.	STRUCT				TO_Struct_Homing	
AutoReversal	BOOL	FALSE	RW、WP、OPRW	1 、 8 、 10	Openness 中的名称: Homing.AutoReversal	
					FALSE	取消激活硬限位开关处的自动反向。
					TRUE	激活硬限位开关处的自动反向。
ApproachDirection	BOOL	TRUE	RW、WP、OPRW	1 、 8 、 10	Openness 中的名称: Homing.ApproachDirection	
					FALSE	负逼近方向（用于查找回原点开关）和 负回原点方向
					TRUE	正逼近方向（用于搜索参考点开关）和 正回原点方向
ApproachVelocity	REAL	200.0 (0.0 到 1.0E12)	RW、 WP、 OPRW	1 、 8 、 10	主动回原点期间轴的逼近速度 (用组态的测量单位表示) Openness 中的名称: Homing.ApproachVelocity	
ReferencingVelocity	REAL	40.0 (0.0 到 1.0E12)	RW、 WP、 OPRW	1 、 8 、 10	主动回原点期间轴的回原点速度 (用组态的测量单位表示) Openness 中的名称: Homing.ReferencingVelocity	

## 参见

[定位轴工艺对象 V4...5 的变量 \(页 485\)](#)

[轴工艺对象 V1...3 的变量 \(页 472\)](#)

### 12.9.14 PositionControl 变量 (V6 及以上版本)

变量结构 <轴名称>.PositionControl.<变量名称> 中包含有位置控制设置。

## 变量

[图例 \(页 335\)](#)

变量	数据类型	值	访问	W	说明
PositionControl.	STRUCT				TO_Struct_PositionControl
Kv	REAL	10.0 (0.0 到 2147480.0 )	R WP_P D OPRW	- 10	闭环位置控制的比例增益 ("Kv" > 0.0) Openness 中的名称: PositionControl.Kv
Kpc	REAL	100.0 (0.0 到 150.0)	R WP_P D OPRW	- 10	位置控制的百分比速度预控制 Openness 中的名称: PositionControl.Kpc

## 参见

[定位轴工艺对象 V4...5 的变量 \(页 485\)](#)

[轴工艺对象 V1...3 的变量 \(页 472\)](#)

## 12.9 定位轴工艺对象 V6...7 的变量

### 12.9.15 FollowingError 变量 (V6 及以上版本)

变量结构 <轴名称>.FollowingError.<变量名称> 中包含有动态跟随误差监视的组态。

#### 变量

图例 (页 335)

变量	数据类型	值	访问	W	说明
FollowingError.	STRUCT				TO_Struct_FollowingError
EnableMonitoring	BOOL	TRUE	R OPRW	-	Openness 中的名称: FollowingError.EnableMonitoring
					FALSE 跟随误差监视已取消激活
					TRUE 跟随误差监视已启用
MinValue	REAL	10.0 (0.0 到 1.0E12)	R WP_PD OPRW	- 10	速度低于“MinVelocity”的值时允许跟随误差。 Openness 中的名称: FollowingError.MinValue
.MaxValue	REAL	100.0 (0.0 到 1.0E12)	R WP_PD OPRW	- 10	速度达到最大速度时允许的最大跟随误差。 Openness 中的名称: FollowingError.MaxValue
MinVelocity	REAL	10.0 (0.0 到 1.0E12)	R WP_PD OPRW	- 10	“MinValue”允许低于该速度，并保持恒定。 Openness 中的名称: FollowingError.MinVelocity

#### 参见

定位轴工艺对象 V4...5 的变量 (页 485)

轴工艺对象 V1...3 的变量 (页 472)

### 12.9.16 PositionMonitoring 变量 (V6 及以上版本)

变量结构 <轴名称>.PositionMonitoring.<变量名称> 中包含有定位运动结束时的定位监控组态。

#### 变量

图例 (页 335)

变量	数据类型	值	访问	W	说明
PositionMonitoring	STRUCT				TO_Struct_PositionMonitoring
ToleranceTime	REAL	1.0 (0.0 到 1.0E12)	R WP_PD OPRW	- 10	容差时间 从达到速度设定值 0 开始直至进入到定位窗口中的最大允许持续时间 Openness 中的名称: PositionMonitoring.ToleranceTime
MinDwellTime	REAL	0.1 (0.0 到 1.0E12)	R WP_PD OPRW	- 10	在定位窗口停留的最短时间 Openness 中的名称: PositionMonitoring.MinDwellTime
Window	REAL	1.0 (0.001 到 1.0E12)	R WP_PD OPRW	- 10	定位窗口 Openness 中的名称: PositionMonitoring.Window

#### 参见

定位轴工艺对象 V4...5 的变量 (页 485)

轴工艺对象 V1...3 的变量 (页 472)

## 12.9 定位轴工艺对象 V6...7 的变量

### 12.9.17 StandstillSignal 变量 (V6 及以上版本)

变量结构 <轴名称>.StandstillSignal.<变量名称> 中包含有停止信号的组态。

#### 变量

图例 (页 335)

变量	数据类型	值	访问	W	说明
StandstillSignal.	STRUCT				TO_Struct_StandstillSignal
VelocityThreshold	REAL	5.0 (0.0 到 1.0E12)	R WP_PD OPRW	- 10	速度阈值  如果速度低于此阈值，则最短停留时间开始。  Openness 中的名称： StandStillSignal.VelocityThreshold
MinDwellTime	REAL	0.01 (0.0 到 1.0E12)	R WP_PD OPRW	- 10	最短停留时间  Openness 中的名称： StandStillSignal.MinDwellTime

#### 参见

定位轴工艺对象 V4...5 的变量 (页 485)

轴工艺对象 V1...3 的变量 (页 472)

### 12.9.18 StatusPositioning 变量 (V6 及以上版本)

变量结构 <轴名称>.StatusPositioning.<变量名称> 指示定位运动的状态。

#### 变量

图例 (页 335)

变量	数据类型	值	访问	W	说明
StatusPositioning.	STRUCT				TO_Struct_StatusPositioning
Distance	REAL	0.0 (-9.0E15 到 9.0E15)	RCCP、 PR、 ORP	-	轴距目标位置的当前距离 (用组态的测量单位表示) 该变量值仅在使用“MC_MoveAbsolute”、“MC_MoveRelative”或轴控制面板执行定位命令期间有效。 Openness 中的名称: StatusPositioning.Distance
TargetPosition	REAL	0.0 (-9.0E15 到 9.0E15)	RCCP、 PR、 ORP	-	轴的目标位置 (用组态的测量单位表示) 该变量值仅在使用“MC_MoveAbsolute”、“MC_MoveRelative”或轴控制面板执行定位命令期间有效。 Openness 中的名称: StatusPositioning.TargetPosition
FollowingError	REAL	0.0 (-9.0E15 到 9.0E15)	RCCP、 PR、 ORP	-	轴的当前跟随误差 (用组态的测量单位表示) 对于通过 PTO 进行驱动装置连接, FollowingError = 0.0 (Pulse Train Output)。 Openness 中的名称: StatusPositioning.FollowingError

---

## 12.9 定位轴工艺对象 V6...7 的变量

### 参见

[定位轴工艺对象 V4...5 的变量 \(页 485\)](#)

[轴工艺对象 V1...3 的变量 \(页 472\)](#)

### 12.9.19 StatusDrive 变量 (V6 及以上版本)

变量结构 <轴名称>.StatusDrive.<变量名称> 指示驱动装置的状态。

#### 变量

图例 (页 335)

变量	数据类型	值	访问	W	说明
StatusDrive.	STRUCT				TO_Struct_StatusDrive
InOperation	BOOL	FALSE	RCCP、 PR、 ORP	-	驱动器的运行状态  Openness 中的名称: StatusDrive.InOperation
					FALSE 驱动装置未就绪。将不执行设定值。
					TRUE 驱动装置就绪。可以执行设定值。
CommunicationOK	BOOL	FALSE	RCCP、 PR、 ORP	-	控制器与驱动器之间的周期性总线通信  Openness 中的名称: StatusDrive.CommunicationOK
					FALSE 未建立通信
					TRUE 通信已建立
AdaptionState	DINT	0 (0 到 4)	R、OPR	10	驱动装置的传送状态  Openness 中的名称: StatusDrive.AdaptionState
					0 数据未传送
					1 数据正在传送
					2 数据已传送
					3 无法传送或未选择
					4 数据传送时出错

## 12.9 定位轴工艺对象 V6...7 的变量

### 参见

[定位轴工艺对象 V4...5 的变量 \(页 485\)](#)

[轴工艺对象 V1...3 的变量 \(页 472\)](#)

### 12.9.20 StatusSensor 变量 (V6 及以上版本)

变量结构 <轴名称>.StatusSensor[1].<变量名称> 指示测量系统的状态。

### 变量

[图例 \(页 335\)](#)

变量	数据类型	值	访问	W	说明
StatusSensor[1].	STRUCT				TO_Struct_StatusSensor
State	DINT	0 (0 到 2)	RCCP、 PR、 ORP	-	编码器值的状态  Openness 中的名称: StatusSensor.State
					0 无效
					1 等待有效状态
					2 有效
Communication OK	BOOL	FALSE	RCCP、 PR、 ORP	-	控制器与编码器之间的周期性总线通信  Openness 中的名称: StatusSensor.CommunicationOK
					FALSE 未建立通信
					TRUE 通信已建立
AbsEncoderOffset	REAL	0.0 (-9.0E15 到 9.0E15)	RCCP、 PR、 ORP	-	到绝对值编码器的值的起始点偏移。  此值将永久性存储在 CPU 中。  Openness 中的名称: StatusSensor.AbsEncoderOffset

变量	数据类型	值	访问	W	说明
AdaptionState	DINT	0 (0 到 1)	R、OPR	10	编码器的传送状态  Openness 中的名称: StatusSensor.AdaptionState
		0			数据未传送
		1			数据正在传送
		2			数据已传送
		3			无法传送或未选择
		4			数据传送时出错

## 参见

定位轴工艺对象 V4...5 的变量 (页 485)

轴工艺对象 V1...3 的变量 (页 472)

## 12.9 定位轴工艺对象 V6...7 的变量

## 12.9.21 StatusBits 变量 (V6 及以上版本)

变量 <轴名称>.StatusBits.<变量名称> 中包含有工艺对象的状态信息。

## 变量

图例 (页 335)

变量	数据类型	值	访问	W	说明
StatusBits.	STRUCT				TO_Struct_StatusBits
Activated	BOOL	FALSE	RCCP、 PR、 ORP	-	轴的激活状态  Openness 中的名称: StatusBits.Activated
					FALSE 轴未激活。
					TRUE 轴已激活。轴将连接所指定的 PTO (Pulse Train Output)。将循环更新工艺 数据块中的数据。
Enable	BOOL	FALSE	RCCP、 PR、 ORP	-	轴的启用状态  Openness 中的名称: StatusBits.Enable
					FALSE 轴未启用。
					TRUE 轴已启用且已就绪可接受运动控制命 令。
AxisSimulation	BOOL	FALSE	RCCP、 PR、 ORP	-	Openness 中的名称: StatusBits.AxisSimulation
					FALSE 仿真已禁用。
					TRUE 仿真已启用。
NonPositionCon trolled	BOOL	FALSE	RCCP、 PR、 ORP	-	Openness 中的名称: StatusBits.NonPositionControlled
					FALSE 轴处于位置控制的模式。

变量	数据类型	值	访问	W	说明
				TRUE	轴处于非位置控制操作状态。
HomingDone	BOOL	FALSE	RCCP、 PR、 ORP	-	<p>轴的回原点状态 Openness 中的名称: StatusBits.HomingDone</p> <p>FALS E 轴未回原点。</p> <p>TRUE 轴已回原点且能够执行绝对定位命令。</p> <p>对于相对定位而言，轴不必回原点。 在主动回原点过程中，该状态为 FALSE。 如果轴事先已经回原点，则在被动回原点期间该状态会保持为 TRUE。</p>
Done	BOOL	FALSE	RCCP、 PR、 ORP	-	<p>轴上的命令执行 Openness 中的名称: StatusBits.Done</p> <p>FALS E 轴上的运动控制命令已激活。</p> <p>TRUE 轴上的运动控制命令未激活。</p>
Error	BOOL	FALSE	RCCP、 PR、 ORP	-	<p>轴上的错误状态 Openness 中的名称: StatusBits.Error</p> <p>FALS E 轴上当前无错误。</p> <p>TRUE 轴上发生错误。</p> <p>有关错误的详细信息，请参见自动模式下运动控制指令的“ErrorID”和“ErrorInfo”参数。 在手动模式下，轴控制面板的“错误消息”(Error message) 框可显示有关错误原因的详细信息。</p>

## 12.9 定位轴工艺对象 V6...7 的变量

变量	数据类型	值	访问	W	说明
Standstill	BOOL	FALSE	RCCP、PR、ORP	-	轴的停止状态  Openness 中的名称: StatusBits.Standstill
					FALS E 轴正在运动。
					TRUE 轴处于停止状态。
PositioningCom mand	BOOL	FALSE	RCCP、PR、ORP	-	执行定位命令  Openness 中的名称: StatusBits.PositioningCommand
					FALS E 轴上的定位命令未激活。
					TRUE 轴正在执行“MC_MoveRelative”或 “MC_MoveAbsolute”运动控制指令的 定位命令。
VelocityComma nd	BOOL	FALSE	RCCP、PR、ORP	-	以速度设定值执行命令  Openness 中的名称: StatusBits.VelocityCommand
					FALS E 轴上以速度设定值执行的命令未激活。
					TRUE 轴正在以“MC_MoveVelocity”或 “MC_MoveJog”运动控制指令的速度设 定值执行运动命令。
HomingComma nd	BOOL	FALSE	RCCP、PR、ORP	-	执行回原点命令  Openness 中的名称: StatusBits.HomingCommand
					FALS E 轴上的回原点命令未激活。
					TRUE 轴正在执行“MC_Home”运动控制指令 的回原点命令。

变量	数据类型	值	访问	W	说明
CommandTableActive	BOOL	FALSE	RCCP、PR、ORP	-	执行命令表 Openness 中的名称: StatusBits.CommandTableActive
					FALSE 轴上的命令表未激活。
					TRUE 该轴由运动控制指令 “MC_CommandTable”进行控制。
ConstantVelocity	BOOL	FALSE	RCCP、PR、ORP	-	恒速 Openness 中的名称: StatusBits.ConstantVelocity
					FALSE 轴处于加速、减速或停止状态。
					TRUE 已达到速度设定值。轴正以恒速运动。
Accelerating	BOOL	FALSE	RCCP、PR、ORP	-	加速过程 Openness 中的名称: StatusBits.Accelerating
					FALSE 轴正在减速、以恒速运动或处于停止状态。
					TRUE 轴正在加速。
Decelerating	BOOL	FALSE	RCCP、PR、ORP	-	减速过程 Openness 中的名称: StatusBits.Decelerating
					FALSE 轴正在加速、以恒速运动或处于停止状态。
					TRUE 轴正在减速。

## 12.9 定位轴工艺对象 V6...7 的变量

变量	数据类型	值	访问	W	说明
ControlPanelActive	BOOL	FALSE	RCCP、PR、ORP	-	轴控制面板的激活状态  Openness 中的名称: StatusBits.ControlPanelActive
					FALSE “自动模式”已激活。用户程序对轴具有优先控制权。
					TRUE 在轴控制面板中启用了“手动控制”(Manual control) 模式。轴控制面板对轴具有优先控制权。不能通过用户程序来控制轴。
DriveReady	BOOL	FALSE	RCCP、PR、ORP	-	驱动器的运行状态  Openness 中的名称: StatusBits.DriveReady
					FALSE 驱动装置未准备就绪。将不执行设定值。
					TRUE 驱动装置已准备就绪。可以执行设定值。
RestartRequired	BOOL	FALSE	RCCP、PR、ORP	-	需要重启轴  Openness 中的名称: StatusBits.RestartRequired
					FALSE 不需要重新启动轴。
					TRUE 值已在装载存储器中修改。  要在 CPU 处于 RUN 模式时将值下载到工作存储器中，必须重新启动轴。使用运动控制指令 MC_Reset 执行此操作。
SWLimitMinActive	BOOL	FALSE	RCCP、PR、ORP	-	软限位开关下限的状态  Openness 中的名称: StatusBits.SWLimitMinActive
					FALSE 轴保持在所组态的工作区域内。
					TRUE 已到达或超出软限位开关的下限。

变量	数据类型	值	访问	W	说明
SWLimitMaxActive	BOOL	FALSE	RCCP、PR、ORP	-	软限位开关上限的状态 Openness 中的名称: StatusBits.SWLimitMaxActive
					FALSE 轴保持在所组态的工作区域内。
					TRUE 已到达或超出软限位开关的上限。
HWLimitMinActive	BOOL	FALSE	RCCP、PR、ORP	-	硬限位开关下限的状态 Openness 中的名称: StatusBits.HWLimitMinActive
					FALSE 轴保持在所组态的允许行进范围内。
					TRUE 已到达或超出硬限位开关的下限。
HWLimitMaxActive	BOOL	FALSE	RCCP、PR、ORP	-	硬限位开关上限的状态 Openness 中的名称: StatusBits.HWLimitMaxActive
					FALSE 轴保持在所组态的允许行进范围内。
					TRUE 已到达或超出硬限位开关的上限。

## 参见

定位轴工艺对象 V4...5 的变量 (页 485)

轴工艺对象 V1...3 的变量 (页 472)

## 12.9 定位轴工艺对象 V6...7 的变量

## 12.9.22 ErrorBits 变量 (V6 及更高版本)

变量 <轴名称>.ErrorBits.<变量名称> 指示工艺对象处出错。

## 变量

图例 (页 335)

变量	数据类型	值	访问	W	说明
ErrorBits.	STRUCT				TO_Struct_ErrorBits
SystemFault	BOOL	FALSE	RCCP、 PR、 ORP	-	内部系统错误  Openness 中的名称: ErrorBits.SystemFault
ConfigFault	BOOL	FALSE	RCCP、 PR、 ORP	-	轴组态错误  Openness 中的名称: ErrorBits.ConfigFault
DriveFault	BOOL	FALSE	RCCP、 PR、 ORP	-	驱动器中发生错误。“驱动器准备就绪”(Drive ready) 信号丢失。  Openness 中的名称: ErrorBits.DriveFault
SWLimit	BOOL	FALSE	RCCP、 PR、 ORP	-	到达或超出软限位开关  Openness 中的名称: ErrorBits.SWLimit
HWLimit	BOOL	FALSE	RCCP、 PR、 ORP	-	到达或超出硬限位开关  Openness 中的名称: ErrorBits.HWLimit
DirectionFault	BOOL	FALSE	RCCP、 PR、 ORP	-	不允许的运动方向  Openness 中的名称: ErrorBits.DirectionFault
HWUsed	BOOL	FALSE	RCCP、 PR、 ORP	-	另一个轴正在使用相同的 PTO (Pulse Train Output) 并且该轴已通过“MC_Power”启用。  Openness 中的名称: ErrorBits.HWUsed

变量	数据类型	值	访问	W	说明
SensorFault	BOOL	FALSE	RCCP、 PR、 ORP	-	编码器系统中发生错误  Openness 中的名称: ErrorBits.SensorFault
Communication Fault	BOOL	FALSE	RCCP、 PR、 ORP	-	通信错误  与所连接设备进行通信时发生错误。  Openness 中的名称: ErrorBits.CommunicationFault
FollowingError	BOOL	FALSE	RCCP、 PR、 ORP	-	超出了最大的允许跟随误差  Openness 中的名称: ErrorBits.FollowingError
PositioningFault	BOOL	FALSE	RCCP、 PR、 ORP	-	定位误差  在定位运动的末端，轴定位错误。  Openness 中的名称: ErrorBits.PositioningFault
AdaptionError	BOOL	FALSE	RCCP、 PR、 ORP	-	数据传送时出错  Openness 中的名称: ErrorBits.AdaptionError

## 参见

[定位轴工艺对象 V4...5 的变量 \(页 485\)](#)

[轴工艺对象 V1...3 的变量 \(页 472\)](#)

## 12.9 定位轴工艺对象 V6...7 的变量

### 12.9.23 ControlPanel 变量 (V6 及更高版本)

“ControlPanel”变量中不含任何与用户相关的数据。不能在用户程序中访问这些变量。

#### 变量

在 Openness 中可读取以下“ControlPanel”变量：

图例 (页 335)

变量	数据类型	值	访问	W	说明
ControlPanel.	STRUCT				TO_Struct_ControlPanel
Input	STRUCT				TO_Struct_Input
TimeOut	DINT	-	OPR	-	(内部参数) Openness 中的名称: ControlPanel.Input.TimeOut
EsLifeSign	DINT	-	OPR	-	(内部参数) Openness 中的名称: ControlPanel.Input.EsLifeSign
Command[1].	STRUCT				ARRAY[1..1] TO_Struct_Command
ReqCounter	DINT	-	OPR	-	(内部参数) Openness 中的名称: ControlPanel.Input.Command[1].ReqCounter
Type	DINT	-	OPR	-	(内部参数) Openness 中的名称: ControlPanel.Input.Command[1].Type
Position	REAL	-	OPR	-	(内部参数) Openness 中的名称: ControlPanel.Input.Command[1].Position
Velocity	REAL	-	OPR	-	(内部参数) Openness 中的名称: ControlPanel.Input.Command[1].Velocity

变量	数据类型	值	访问	W	说明
	Acceleration	REAL	-	OPR	(内部参数) Openness 中的名称: ControlPanel.Input.Command[1].Acceleration
	Jerk	REAL	-	OPR	(内部参数) Openness 中的名称: ControlPanel.Input.Command[1].Jerk
	Param	INT	-	OPR	(内部参数) Openness 中的名称: ControlPanel.Input.Command[1].Param
Output.		STRUCT	-		TO_Struct_Output
	RTLifeSign	INT	-	OPR	(内部参数) Openness 中的名称: ControlPanel.Output.RTLifeSign
	Command[1].	STRUCT	-		ARRAY[1..1] TO_Struct_Command
	AckCounter	INT	-	OPR	(内部参数) Openness 中的名称: ControlPanel.Output.Command[1].AckCounter
	ErrorID	USHORT	-	OPR	(内部参数) Openness 中的名称: ControlPanel.Output.Command[1].ErrorID
	ErrorInfo	USHORT	-	OPR	(内部参数) Openness 中的名称: ControlPanel.Output.Command[1].ErrorInfo
	Done	BOOL	-	OPR	(内部参数) Openness 中的名称: ControlPanel.Output.Command[1].Done
	Aborted	BOOL	-	OPR	(内部参数) Openness 中的名称: ControlPanel.Output.Command[1].Aborted

## 12.9 定位轴工艺对象 V6...7 的变量

### 参见

[定位轴工艺对象 V4...5 的变量 \(页 485\)](#)

[轴工艺对象 V1...3 的变量 \(页 472\)](#)

### 12.9.24 内部变量 (V6 及更高版本)

“Internal”变量中不含任何与用户相关的数据。不能在用户程序中访问这些变量。

### 变量

[图例 \(页 335\)](#)

变量	数据类型	值	访问	W	说明
Internal[1..4].	STRUCT				ARRAY [1..4] TO_Struct_Internal
Id	INT	0 (-32768 到 32767)	OPRW	-	(内部参数) Openness 中的名称: Internal[1..4].Id
Value	REAL	0 (-9.0E15 到 9.0E15)	OPRW	-	(内部参数) Openness 中的名称: Internal[1..4].Value

### 参见

[定位轴工艺对象 V4...5 的变量 \(页 485\)](#)

[轴工艺对象 V1...3 的变量 \(页 472\)](#)

### 12.9.25 工艺对象变量的更新

工艺对象变量中指示的轴的状态和错误信息在各周期控制点处更新。

组态变量值的更改不会立即生效。有关更改生效条件的信息，请参见相关变量的详细说明。

## 12.10 工艺对象命令表 V6 中的变量

变量结构 <命令表>.Command[n]<变量名称> 中包含有所组态的命令参数。

### 变量

图例 (页 335)

变量	数据类型	值	访问	W	说明
Command[n].	STRUCT				ARRAY[1..32] TO_Struct_Command
Type	INT	0 (0 到 151)	RW、 OPRW	-	命令类型  Openness 中的名称: Command[n].Type
					0 “清空”命令
					2 “停止”命令
					5 “相对定位”命令
					6 “绝对定位”命令
					7 命令“Velocity setpoint”
					151 命令“Wait”
Position	REAL	0.0	RW、 OPRW	-	命令的目标位置/行进距离  Openness 中的名称: Command[n].Position
Velocity	REAL	0.0	RW、 OPRW	-	速度命令  Openness 中的名称: Command[n].Velocity
Duration	REAL	0.0	RW、 OPRW	-	持续时间命令  Openness 中的名称: Command[n].Duration
NextStep	INT	0 (0 到 1)	RW、 OPRW	-	用于切换到下一条命令的模式  Openness 中的名称: Command[n].NextStep
					0 “完成命令”

## 12.10 工艺对象命令表 V6 中的变量

变量	数据类型	值	访问	W	说明	
					1	“混合运动”
StepCode	WORD	0	RW、OPRW	-	命令步代码 Openness 中的名称： Command[n].StepCode	
WarningEnabled*	BOOL	FALSE	OPRW	-	Openness 中的名称： _WarningEnabled	
					FALS E	禁止警告。
					TRUE	启用警告。
UseAxisParametersFrom*	INT/STRING	-	OPRW	-	Openness 中的名称： _UseAxisParametersFrom 下拉列表“使用的轴参数来自”的枚举或 轴名称  不要将轴命名为“采样轴”，因为此名称 是保留的。	

\*) 适用于 Openness

## 参见

命令表工艺对象变量 V4...5 (页 517)

命令表工艺对象 V1...3 的变量 (页 516)

## 12.11 版本 V1...6

### 12.11.1 与运动控制相关的 CPU 输出（工艺版本 V1...3）

可用驱动器的数目取决于 CPU、PTO（脉冲串输出）的数目以及可用的脉冲发生器输出的数目。

下表提供了有关相关依赖性的信息：

#### 最大 PTO 数

可控制 PTO（驱动器）的最大数目取决于 CPU 的产品编号：

CPU 产品编号	PTO 数
xxxxxxxx-1xx <b>30</b> -xxxx	2
xxxxxxxx-1xx <b>31</b> -xxxx	4

无论是否使用信号板，可控制 PTO（驱动器）的最大数目都适用。

## 可用的脉冲发生器输出

CPU 提供一个脉冲输出和一个方向输出，通过脉冲接口对步进电机驱动器或伺服电机驱动器进行控制。脉冲输出为驱动器提供电机运动所需的脉冲。方向输出则用于控制驱动器的行进方向。

脉冲输出和方向输出彼此互相分配的关系保持不变并形成脉冲发生器输出。板载 CPU 输出或信号板输出可用作脉冲发生器输出。在设备组态期间，可以在“属性”(Properties) 选项卡上的脉冲发生器 (PTO/PWM) 下，选择板载 CPU 输出或信号板输出。

下表列出了每个 CPU 或信号板可用的驱动器的数目：

CPU	板载	信号板					
		DI2/DO2 x DC24V 20kHz	DI2/DO2 x DC24V 200kHz	DO4 x DC24V 200kHz	DI2/DO2 x DC5V 200kHz	DO4 x DC5V 200kHz	
CPU 1211C、CPU 1212C、CPU 1214C (MLFB - 产品编号 xxxxxxxx-1xx <b>30</b> -xxxx)	DC/DC /DC	2	2	2	2	2	2
	AC/DC /RLY	-	1	1	2	1	2
	DC/DC /RLY	-	1	1	2	1	2
CPU 1211C (MLFB - 产品编号 xxxxxxxx-1xx <b>31</b> -xxxx)	DC/DC /DC	2	3	3	4	3	4
	AC/DC /RLY	-	1	1	2	1	2
	DC/DC /RLY	-	1	1	2	1	2
CPU 1212C (MLFB - 产品编号 xxxxxxxx-1xx <b>31</b> -xxxx)	DC/DC /DC	3	4	4	4	4	4
	AC/DC /RLY	-	1	1	2	1	2
	DC/DC /RLY	-	1	1	2	1	2

CPU		板载	信号板				
CPU 1214C (MLFB - 产品编号 xxxxxxxx-1xx <b>31</b> - xxxx)	DC/DC /DC	4	4	4	4	4	4
	AC/DC /RLY	-	1	1	2	1	2
	DC/DC /RLY	-	1	1	2	1	2
CPU 1215C	DC/DC /DC	4	4	4	4	4	4
	AC/DC /RLY	-	1	1	2	1	2
	DC/DC /RLY	-	1	1	2	1	2

下表列出了脉冲输出和方向输出的地址分配：

CPU S7-1200		PTO1 输出 1		PTO2 输出 2		PTO3 输出 1		PTO4 输出 2	
		脉冲	方向	脉冲	方向	脉冲	方向	脉冲	方向
CPU 1211 C、 CPU 1212 C、 CPU 1214 C、 CPU 1215C (DC/DC/D C)	CPU	Ax.0	Ax.1	Ax.2	Ax.3	Ax.4	Ax.5	Ax.6	Ax.7
	信号板	Ay.0	Ay.1	Ay.2	Ay.3	Ay.0	Ay.1	Ay.2	Ay.3

CPU S7-1200		PTO1 输出 <sup>1</sup>		PTO2 输出 <sup>2</sup>		PTO3 输出 <sup>1</sup>		PTO4 输出 <sup>2</sup>	
		脉冲	方向	脉冲	方向	脉冲	方向	脉冲	方向
CPU 1211 C、 CPU 1212 C、 CPU 1214 C、 CPU 1215C (AC/DC/R LY)	CPU	-	-	-	-	-	-	-	-
	信号板	Ay.0	Ay.1	Ay.2	Ay.3	Ay.0	Ay.1	Ay.2	Ay.3
CPU 1211 C、 CPU 1212 C、 CPU 1214 C、 CPU 1215C (DC/DC/R LY)	CPU	-	-	-	-	-	-	-	-
	信号板	Ay.0	Ay.1	Ay.2	Ay.3	Ay.0	Ay.1	Ay.2	Ay.3

x = 板载 CPU 输出的初始字节地址（默认值为 0）

y = 信号板输出的初始字节地址（默认值为 4）

<sup>1</sup> 如果同时使用 DC/DC/DC 型 CPU 和 DI2/DO2 信号板，则可通过板载 CPU 输出或通过信号板生成 PTO1/3 的信号。

<sup>2</sup> 如果同时使用 DC/DC/DC 型 CPU 和 DO4 信号板，则可通过板载 CPU 输出或通过信号板生成 PTO1/3 或 PTO2/4 的信号。

PTO3 和 PTO4 仅适用于产品编号为 xxxxxxxx-1xx**31**-xxxx 的 CPU。

---

## 说明

### 通过过程映像访问脉冲发生器输出

如果已选择 PTO (Pulse Train Output) 并将其分配给某个轴，固件将通过相应的脉冲发生器输出接管控制。

在实现上述控制功能接管后，将断开过程映像和 I/O 输出间的连接。虽然用户可通过用户程序或监视表写入脉冲发生器输出的过程映像，但所写入的内容不会传送到 I/O 输出。因此通过用户程序或监视表格无法监视 I/O 输出。读取的信息反映过程映像中的值，与 I/O 输出的实际状态不一致。

对于 CPU 固件非永久使用的其它所有 CPU 输出，通常可以通过过程映像监控 I/O 输出的状态。

---

## 脉冲输出的频率范围

脉冲输出的频率范围如下表所示：

脉冲输出	工艺对象定位轴 V1 的频率范围	工艺对象定位轴 V2/V3 的频率范围 (CPU < V3.0)	工艺对象定位轴 V2/V3 的频率范围 (CPU V3.0)
板载 (MLFB - 产品编号 xxxxxxxx-1xx30-xxxx)	2 Hz ≤ f ≤ 100 kHz	2 Hz ≤ f ≤ 100 kHz	1 Hz ≤ f ≤ 100 kHz
板载 (MLFB - 产品编号 xxxxxxxx-1xx31xxxx)	2 Hz ≤ f ≤ 100 kHz (PTO 1+2) 2 Hz ≤ f ≤ 20 kHz (PTO 3+4)	2 Hz ≤ f ≤ 100 kHz (PTO 1+2) 2 Hz ≤ f ≤ 20 kHz (PTO 3+4)	1 Hz ≤ f ≤ 100 kHz (PTO 1+2) 1 Hz ≤ f ≤ 20 kHz (PTO 3+4)
信号板 DI2/DO2 x DC24V 20kHz	2 Hz ≤ f ≤ 20 kHz	2 Hz ≤ f ≤ 20 kHz	1 Hz ≤ f ≤ 20 kHz
信号板 DI2/DO2 x DC24V 200kHz	2 Hz ≤ f ≤ 100 kHz	2 Hz ≤ f ≤ 200 kHz	1 Hz ≤ f ≤ 200 kHz
信号板 DO4 x DC24V 200kHz	2 Hz ≤ f ≤ 100 kHz	2 Hz ≤ f ≤ 200 kHz	1 Hz ≤ f ≤ 200 kHz

脉冲输出	工艺对象定位轴 V1 的频率范围	工艺对象定位轴 V2/V3 的频率范围 (CPU < V3.0)	工艺对象定位轴 V2/V3 的频率范围 (CPU V3.0)
信号板 DI2/DO2 x DC5V 200kHz	2 Hz ≤ f ≤ 100 kHz	2 Hz ≤ f ≤ 200 kHz	1 Hz ≤ f ≤ 200 kHz
信号板 DO4 x DC5V 200kHz	2 Hz ≤ f ≤ 100 kHz	2 Hz ≤ f ≤ 200 kHz	1 Hz ≤ f ≤ 200 kHz

## 驱动器信号

对于运动控制，可以选择将驱动器接口设置为“驱动器启用”或“驱动器准备就绪”。使用驱动器接口时，可以随意选择用于驱动器启用的数字量输出和用于“驱动器准备就绪”的数字量输入。

## 加速度/减速度限值

以下限值适用于加速度和减速度：

加速度/减速度	值 (CPU < V3.0)	值 (CPU V3.0)
最小加速度/减速度	2.8E-1 个脉冲/秒 <sup>2</sup>	5.0E-3 个脉冲/秒 <sup>2</sup>
最大加速度/减速度	9.5E+9 个脉冲/秒 <sup>2</sup>	9.5E+9 个脉冲/秒 <sup>2</sup>

## 加加速度限值

以下限值适用于加加速度：

加加速度	值 (CPU < V3.0)	值 (CPU V3.0)
最小加加速度	4.0E-2 个脉冲/秒 <sup>3</sup>	4.0E-3 个脉冲/秒 <sup>3</sup>
最大加加速度	1.5E+8 个脉冲/秒 <sup>3</sup>	1.0E+10 个脉冲/秒 <sup>3</sup>

## 参见

运动控制相关的 CPU 输出 (页 16)

## 12.11.2 组态对话框

### 12.11.2.1 V1...3

#### 组态 - 常规 (“轴”工艺对象 V1...3)

在“常规”(General) 组态窗口中，组态“轴”工艺对象的基本属性。

##### 轴名称:

在该框中定义轴名称，或“轴”工艺对象的名称。该工艺对象以该名称列出在项目导航区中。

##### 硬件接口

脉冲通过固定分配的数字量输出输出到驱动器的动力装置。

对于具有继电器输出的 CPU，由于继电器不支持所需的开关频率，因此无法通过这些输出来输出脉冲信号。如果要在这些 CPU 中使用 PTO (Pulse Train Output)，必须使用具有数字量输出的信号板。

---

##### 说明

PTO 需要高速计数器 (HSC) 功能。CPU 版本小于 V3.0 时，可将 HSC 用于此目的。但随后用户不能再使用该 HSC。CPU 版本大于等于 V3.0 时，将使用内部 HSC。

无法通过其输入地址来评估计数。

PTO 和 HSC 间的分配是固定的。当用户激活 PTO1 时，其连接到 HSC1。激活 PTO2 时，其与 HSC2 相连。

---

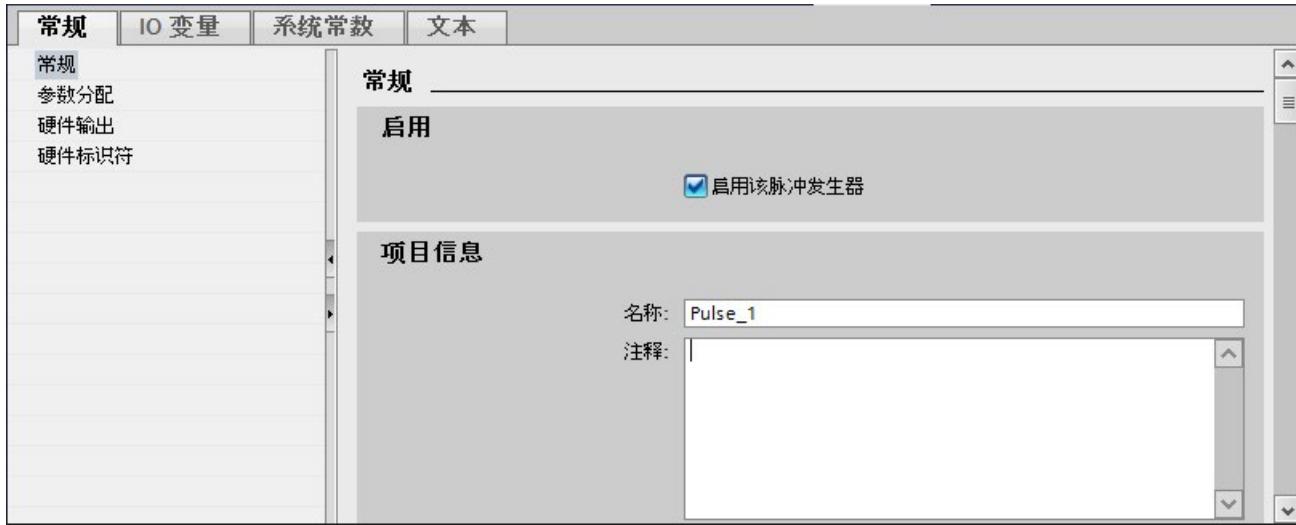
在“脉冲发生器选择”(Pulse generator selection) 下拉列表中，选择 PTO (Pulse Train Output)，它将提供用于控制采用脉冲接口的步进电机或伺服电机的脉冲。如果没有在设备组态中的其它地方使用脉冲发生器和高速计数器，则系统会自动组态硬件接口。这种情况下，下拉列表中所选的 PTO 以白色背景显示。将在“输出源”(Output source)、“脉冲输出”(Pulse output)、“方向输出”(Direction output) 和“已分配高速计数器”(Assigned fast counter) 输出框中列出所用接口。

如果要更换接口或者无法自动组态 PTO (“脉冲发生器选择”(Pulse generator selection) 下拉列表中的条目突出显示为红色)，请按照以下步骤操作：

1. 单击“设备组态”(Device configuration) 按钮。

将打开脉冲发生器设备组态。

如果看不到脉冲发生器的组态，则放大设备组态的属性窗口。



2. 选中“启用该脉冲发生器”(Enable this pulse generator) 复选框。

3. 在区域导航中选择“参数分配”(Parameter assignment) 条目。

将打开“参数分配”(Parameter assignment) 窗口。



4. 在“脉冲发生器用作：”(Pulse generator as:) 下拉列表中选择“PTO”条目。

5. 在“输出源:”(Output source:) 下拉列表中, 选择“集成 CPU 输出”(Integrated CPU output) 或“信号板输出”(Signal board output) 条目。只能为 PTO1 或为 PTO1 和 PTO2 选择“信号板输出”(Signal board output) 条目, 具体取决于安装的信号板。更多详细信息, 请参见以下部分: 运动控制相关的 CPU 输出 (页 16)
6. 返回到轴组态。

如果其它任何地方都未使用相应的高速计数器, 则“常规”(General) 轴组态中的 PTO 框的背景不会呈现为红色。否则, 根据错误消息更正组态。

## 用户单位

从下拉列表中为轴量纲系统选择所需的单位。所选单位用于进一步组态“轴”工艺对象和显示当前轴数据。

运动控制指令的输入参数 (例如, Position、Distance、Velocity 等) 的值也会采用该单位。

---

### 说明

如果后来更改了该量纲系统, 则在该工艺对象的所有组态窗口中可能无法相应地进行正确转换。这种情况下, 请检查所有轴参数的组态。

可能需要在用户程序中根据新测量单位调整运动控制指令的输入参数的值。

---

## 参见

组态 - 常规 (页 82)

## 组态 - 回原点 (“轴”工艺对象 V1)

在“回原点”(Homing) 组态窗口中, 组态主动和被动回原点参数。可以使用运动控制指令中的“模式”(Mode) 输入参数设置回原点方法。其中, Mode = 2 表示被动回原点, Mode = 3 表示主动回原点。

## 参考点开关输入

从下拉列表中为参考点开关选择数字量输入。该输入必须具有中断功能。板载 CPU 输入和所插入信号板输入都可以选作参考点开关的输入。

### 说明

默认情况下，数字量输入的滤波时间设置为 6.4 ms。

采用默认时间的数字量输入用作参考点开关的输入时，可能引起意外减速，从而导致出现误差。根据回原点速度和参考点开关的范围，可能检测不到参考点。可以在数字量输入设备组态的“输入滤波器”(Input filter) 中设置滤波时间。

指定的滤波时间必须小于参考点开关的输入信号的持续时间。

## 允许在到达硬限位开关后反向（仅限主动回原点）

激活该复选框可将硬限位开关用作回原点过程中的反向凸轮。只有激活硬限位开关才能实现反向控制。如果使用 CPU 固件 V1.0，必须组态两个硬限位开关。如果使用 CPU 固件 V2.0 及更高版本，则仅需组态逼近方向上的硬限位开关。

如果在主动回原点期间到达硬件限位开关，轴将以组态的减速度减速（不是以紧急减速度），然后反向。然后反向检测参考点开关。

如果未激活反向功能且在主动回原点过程中轴到达硬限位开关，则将因错误而中止回原点过程并以急停减速度对轴进行制动。

### 说明

采用以下措施之一可确保机器在发生反向时不会行进到机械停止块：

- 保持较低的逼近速度
- 增大组态的加速度/减速度
- 增大硬限位开关和机械停止块间的距离

## 逼近/回原点方向（主动和被动回原点）

通过方向选择，可以决定主动回原点期间用于搜索参考点开关的“逼近方向”以及回原点的方向。回原点方向指定执行回原点操作时轴用于逼近组态的回原点开关侧的行进方向。

有关逼近方向设置对被动回原点的影响，请参见“参考点开关”下的表格。

## 参考点开关侧（主动和被动回原点）

### • 主动回原点

在此处可以选择轴在参考点开关的下端还是上端进行回原点。

### 说明

根据轴的起始位置和组态的回原点参数，回原点过程顺序可能与组态窗口中图示的顺序不同。

### • 被动回原点

对于被动回原点，必须由用户通过运动命令来执行回原点的行进运动。回原点发生在参考点开关的哪一端取决于以下因素：

- “逼近方向”组态
- “参考点开关”组态
- 被动回原点过程中当前的行进方向

下表列出了受这些因素影响的详细信息：

影响因素:			结果:
组态 逼近方向	组态 参考点开关	当前的行进方向	回原点发生在 参考点开关
正方向	“下侧”	正方向	上侧
		负方向	下侧
正方向	“上侧”	正方向	下侧
		负方向	上侧
负方向	“下侧”	正方向	下侧
		负方向	上侧
负方向	“上侧”	正方向	上侧
		负方向	下侧

## 速度（仅限主动回原点）

在该框中，可以指定回原点期间搜索参考点开关的速度。

限值（与所选测量单位无关）：

- 启动/停止速度  $\leq$  逼近速度  $\leq$  最大速度

## 回原点速度（仅限主动回原点）

在该框中，可以指定回原点期间轴逼近参考点开关的速度。

限值（与所选测量单位无关）：

- 启动/停止速度  $\leq$  回原点速度  $\leq$  最大速度

## 起始位置偏移值（仅限主动回原点）

如果指定的起始位置与参考点开关的位置存在偏差，则可在此框中指定起始位置偏移值。

如果值不等于 0，轴回到参考点开关位置后将执行以下动作：

1. 以回原点速度使轴移动起始位置偏移值指定的一段距离
2. 达到“起始位置偏移值”时，轴处于运动控制指令“MC\_Home”的输入参数“Position”中指定的起始位置处。

限值（与所选测量单位无关）：

- $-1.0e12 \leq$  起始位置偏移值  $\leq 1.0e12$

## 起始位置

将运动控制指令“MC\_Home”中所组态的位置用作起始位置。

## 组态 - 回原点（“轴”工艺对象 V2...3）

### 组态 - 回原点 - 常规（轴工艺对象 V2...3）

在“回原点 - 常规”(Homing - General) 组态窗口中，可以组态主动和被动回原点的参考点开关输入。

## 参考点开关输入

从该下拉列表中可为回原点开关选择数字量输入。该输入必须具有中断功能。板载 CPU 输入和所插入信号板输入都可选作回原点开关的输入。

### 说明

默认情况下，数字量输入的滤波时间设置为 6.4 ms。

数字量输入用作回原点开关的输入时，可能引起意外减速，从而导致出现误差。根据回原点速度和回原点开关的范围，可能检测不到回原点位置。可以在数字量输入设备组态的“输入滤波器”(Input filter) 中设置滤波时间。

指定的滤波时间必须小于回原点开关的输入信号持续时间。

## 有效电平

在下拉列表中，选择回原点时使用的回原点开关电平。

### 参见

顺序 - 主动回原点 (页 122)

## 组态 - 回原点 - 被动 (轴工艺对象 V2...3)

在“归位 - 被动”(Homing - Passive) 组态窗口中，可以组态被动归位所需的参数。

被动归位的移动必须由用户触发（例如，使用轴运动命令）。运动控制指令“MC\_Home”的输入参数“Mode”= 2 时，会启动被动归位。

## 回原点开关侧

在此处可以选择轴是在回原点开关的下端还是上端进行回原点。

## 起始位置

将运动控制指令“MC\_Home”中所组态的位置用作起始位置。

### 说明

如果未使用轴运动命令进行被动归位（轴处于停止状态），则将在下一个归位开关的上升沿或下降沿处执行归位操作。

## 组态 - 回原点 - 主动 (轴工艺对象 V2...3)

在“主动回原点”(Active homing) 组态窗口中组态主动回原点所需的参数。运动控制指令“MC\_Home”的输入参数“Mode”= 3 时，会启动主动回原点。

### 允许在硬限位开关处自动反向

激活该复选框可将硬限位开关用作回原点过程中的反向凸轮。只有启用硬限位开关才能实现反向控制（必须至少组态位于逼近方向上的硬限位开关）。

如果在主动回原点过程中到达硬限位开关，轴将以组态的减速度（不是以急停减速度）制动，然后反向。然后反向检测回原点开关。

如果未激活反向功能且在主动回原点过程中轴到达硬限位开关，则将因错误而中止回原点过程并以急停减速度对轴进行制动。

---

### 说明

如果可能，采用以下措施之一以确保机器在发生反向时不会行进到机械挡块：

- 保持较低的行进速度。
  - 增加组态的加速度/减速度。
  - 增加硬限位开关和机械挡块之间的距离。
- 

### 逼近/回原点方向

通过方向选择，可以决定主动回原点过程中搜索回原点开关的逼近方向以及回原点的方向。回原点方向指定执行回原点操作时轴用于逼近组态的回原点开关端的行进方向。

### 回原点开关侧

在此处可以选择轴是在回原点开关的下端还是上端进行回原点。

### 速度

在该框中，可以指定回原点期间搜索回原点开关的速度。

限值（与所选测量单位无关）：

- 启动/停止速度  $\leq$  逼近速度  $\leq$  最大速度

## 回原点速度

在该框中，可以指定回原点时逼近参考点开关的速度。

限值（与所选测量单位无关）：

- 启动/停止速度  $\leq$  回原点速度  $\leq$  最大速度

## 起始位置偏移值

如果所需的回原点位置与回原点开关的位置存在偏差，则可在此框中指定回原点位置偏移量。

如果该值不等于 0，轴在回原点开关处回原点后将执行以下动作：

1. 以回原点速度使轴移动起始位置偏移值指定的一段距离
2. 达到“起始位置偏移值”时，轴处于运动控制指令“MC\_Home”的输入参数“Position”中指定的起始位置处。

限值（与所选测量单位无关）：

- $-1.0e12 \leq$  起始位置偏移值  $\leq 1.0e12$

## 起始位置

将运动控制指令“MC\_Home”中所组态的位置用作起始位置。

## 在用户程序中更改动态值的组态（“轴”工艺对象 V1...3）

可以在 CPU 中运行用户程序期间更改下列组态参数：

## 加速度和减速度

在用户程序运行期间，还可以更改加速度和减速度值。要执行该操作，可使用以下工艺对象变量：

- <轴名称>.Config.DynamicDefaults.Acceleration  
用于更改加速度
- <轴名称>.Config.DynamicDefaults.Deceleration  
用于更改减速度

有关组态参数的更改何时生效的信息，请参见附录中的工艺对象变量 (页 472)说明。

## 急停减速度

在用户程序运行期间，还可更改急停减速度值。要执行该操作，可使用以下工艺对象变量：

- <轴名称>.Config.DynamicDefaults.EmergencyDeceleration

有关组态参数的更改何时生效的信息，请参见附录中的工艺对象变量说明。

### 注意

更改该参数后，可能需要调整硬限位开关的位置以及其它安全相关的设置。

## 冲击限制（从 V2.0“轴”工艺对象起）

还可在用户程序运行时激活和禁用冲击限制和更改冲击值。要执行该操作，可使用以下工艺对象变量：

- <轴名称>.Config.DynamicDefaults.JerkActive

用于激活和禁用冲击限制

- <轴名称>.Config.DynamicDefaults.Jerk

用于更改冲击

有关组态参数的更改何时生效的信息，请参见附录中的工艺对象变量说明。

## 参见

在用户程序中可以更改动态组态 (页 114)

## 12.11.2.2 V4

### 组态 - 常规（自 V4 起的定位轴工艺对象）

在“常规”(General) 组态窗口中，组态定位轴工艺对象的基本属性。

#### 轴名称

在该域中定义轴名称或定位轴工艺对象的名称。该工艺对象以该名称列出在项目树中。

## 硬件接口

脉冲通过固定分配的数字量输出输出到驱动器的动力装置。

对于具有继电器输出的 CPU，由于继电器不支持所需的开关频率，因此无法通过这些输出来输出脉冲信号。如果要在这些 CPU 中使用 PTO (Pulse Train Output)，必须使用具有数字量输出的信号板。

---

### 说明

PTO 需要高速计数器 (HSC) 的功能。为此将使用内部 HSC，不能对此 HSC 的计数进行评估。

---

## 选择脉冲发生器

在该下拉列表中，选择 PTO (Pulse Train Output)，它将提供用于控制采用脉冲接口的步进电机或伺服电机的脉冲。如果没有在设备组态中的其它地方使用脉冲发生器和高速计数器，则系统会自动组态硬件接口。这种情况下，下拉列表中所选的 PTO 以白色背景显示。

## “设备组态”(Device configuration) 按钮

该按钮可转到 CPU 设备组态中的脉冲选项设置。

## 信号类型

从下拉列表中选择所需的信号类型。可以使用以下信号类型：

- **PTO (脉冲 A 和方向 B)**

使用一个脉冲输出和一个方向输出控制步进电机。

- **PTO (向上计数 A, 向下计数 B)**

分别使用一个正向和负向运动的脉冲输出控制步进电机。

- **PTO (A/B 相移)**

A 相和 B 相的两个脉冲输出在同一频率下运行。

在驱动器步进结束时会评估这两个脉冲输出的周期。

A 相和 B 相之间的相位偏移量决定了运动方向。

- **PTO (A/B 相位偏移量 - 四重)**

A 相和 B 相的两个脉冲输出在同一频率下运行。

在驱动器步进结束时会评估 A 相和 B 相的所有上升沿和下降沿。

A 相和 B 相之间的相位偏移量决定了运动方向。

### 脉冲输出（信号类型“PTO (脉冲 A 和方向 B) ”）

在此域中选择需要用作脉冲输出的输出。

可以通过符号地址或将其分配给绝对地址来选择输出。

### 激活方向输出（信号类型“PTO (脉冲 A 和方向 B) ”）

在“脉冲和方向”模式下，可以取消激活或激活方向输出。可以使用此选项限制移动方向，并将方向输出用于其它目的。

### 方向输出（信号类型“PTO (脉冲 A 和方向 B) ”）

在此域中选择需要用作方向输出的输出。

可以通过符号地址或将其分配给绝对地址来选择输出。

### 向上脉冲输出（信号类型“PTO (向上计数 A, 向下计数 B) ”）

在此域中选择用于实现正向运动所需的脉冲输出。

可以通过符号地址或将其分配给绝对地址来选择输出。

### 向下脉冲输出（信号类型“PTO（向上计数 A，向下计数 B）”）

在此域中选择用于实现负向运动所需的脉冲输出。

可以通过符号地址或将其分配给绝对地址来选择输出。

### A 相（信号类型“PTO（A/B 相位偏移量）”和“PTO（A/B 相位偏移量 - 四重）”）

在此域中为 A 相信号选择所需的脉冲输出。

可以通过符号地址或将其分配给绝对地址来选择输出。

### B 相（信号类型“PTO（A/B 相位偏移量）”和“PTO（A/B 相位偏移量 - 四重）”）

在此域中为 B 相信号选择所需的脉冲输出。

可以通过符号地址或将其分配给绝对地址来选择输出。

## 用户单位

从下拉列表中为轴量纲系统选择所需的单位。所选单位将用于进一步组态定位轴工艺对象和显示当前轴数据。

运动控制指令的输入参数（例如，Position、Distance、Velocity 等）的值也会采用该单位。

---

### 说明

如果后来更改了该量纲系统，则在该工艺对象的所有组态窗口中可能无法相应地进行正确转换。这种情况下，请检查所有轴参数的组态。

可能需要在用户程序中根据新测量单位调整运动控制指令的输入参数的值。

---

## 组态 - 驱动器信号 (定位轴工艺对象 V4)

在“驱动器信号”(Drive signals) 组态窗口中组态驱动器使能信号的输出以及驱动器的“驱动器准备就绪”(Drive ready) 反馈信号的输入。

驱动器使能信号由运动控制指令“MC\_Power”控制，可以启用对驱动器的供电。信号通过组态的输出提供给驱动器。

如果驱动器在接收到驱动器使能信号之后准备好开始进行行进，则驱动器会向 CPU 发送“驱动器准备就绪”(Drive ready) 信号。“驱动器准备就绪”信号通过组态的输入传回 CPU。

如果驱动器不包含任何这一类型的接口，则无需组态这些参数。这种情况下，为准备就绪输入选择值 TRUE。

## 参见

组态 - 机械 (定位轴工艺对象 V4) (页 406)

位置限制 (页 101)

动态 (页 107)

回原点 (自定位轴工艺对象 V2 起) (页 117)

## 组态 - 机械 (定位轴工艺对象 V4)

在“机械”(Mechanics) 组态窗口中组态驱动器的机械属性。

### 电机每转的增量

在此框中组态电机每转所需的脉冲数。

限值 (与所选测量单位无关) :

- $0 < \text{电机每转的脉冲数} \leq 2147483647$

### 每转的距离

在此框中，组态电机每转带动单元的机械系统行进的负载距离。

限值 (与所选测量单位无关) :

- $0.0 < \text{每转距离} \leq 1.0e12$

## 允许的旋转方向（自 V4 工艺版本起）

通过组态此框可决定系统机械是同时朝两个方向运动，还是只朝正向或负向运动。

如果尚未在“PTO（脉冲 A 和方向 B）”模式下激活脉冲发生器的方向输出，则选择受限于正方向或负方向。

## 反转方向

可使用“反转方向”(Invert direction) 复选框根据驱动器的方向逻辑对控制系统进行调整。

方向逻辑将根据所选脉冲发生器的模式反转：

- **PTO（脉冲 A 和方向 B）**

- 方向输出上输出 0 V ⇒ 正向旋转
- 方向输出上输出 5 V/24 V ⇒ 负向旋转

指定的电压取决于所使用的硬件。各指示值并不适用于 CPU 1217 的差分输出。

- **PTO（向上计数 A，向下计数 B）**

“向下脉冲输出”和“向上脉冲输出”这两个输出相互交换。

- **PTO（A/B 相位偏移量）**

“A 相”和“B 相”输出相互交换。

- **PTO（A/B 相位偏移量 - 四重）**

“A 相”和“B 相”输出相互交换。

## 参见

组态 - 驱动器信号（定位轴工艺对象 V4）(页 406)

信号类型与行进方向之间的关系(页 21)

位置限制(页 101)

动态(页 107)

回原点（自定位轴工艺对象 V2 起）(页 117)

## 组态 - 归位 - 被动 (定位轴工艺对象 V4)

在“归位 - 被动”(Homing - Passive) 组态窗口中，可以组态被动归位所需的参数。

被动归位的移动必须由用户触发（例如，使用轴运动命令）。运动控制指令“MC\_Home”的输入参数“Mode”= 2 时，会启动被动归位。

### 回原点开关输入

从该下拉列表中可为回原点开关选择数字量输入。该输入必须具有中断功能。板载 CPU 输入和所插入信号板输入都可选作回原点开关的输入。

#### 说明

默认情况下，数字量输入的滤波时间设置为 6.4 ms。

数字量输入用作回原点开关的输入时，可能引起意外减速，从而导致出现误差。根据回原点速度和回原点开关的范围，可能检测不到回原点位置。可以在数字量输入设备组态的“输入滤波器”(Input filter) 中设置滤波时间。

指定的滤波时间必须小于回原点开关的输入信号持续时间。

### 有效电平

在下拉列表中，选择回原点时使用的回原点开关电平。

### 回原点开关侧

在此处可以选择轴是在回原点开关的下端还是上端进行回原点。

### 起始位置

将运动控制指令“MC\_Home”中所组态的位置用作起始位置。

#### 说明

如果未使用轴运动命令进行被动归位（轴处于停止状态），则将在下一个归位开关的上升沿或下降沿处执行归位操作。

## 组态 - 回原点 - 主动 (定位轴工艺对象 V4)

在“主动回原点”(Active homing) 组态窗口中组态主动回原点所需的参数。运动控制指令“MC\_Home”的输入参数“Mode”= 3 时，会启动主动回原点。

## 回原点开关输入

从该下拉列表中可为回原点开关选择数字量输入。该输入必须具有中断功能。板载 CPU 输入和所插入信号板输入都可选作回原点开关的输入。

### 说明

默认情况下，数字量输入的滤波时间设置为 6.4 ms。

数字量输入用作回原点开关的输入时，可能引起意外减速，从而导致出现误差。根据回原点速度和回原点开关的范围，可能检测不到回原点位置。可以在数字量输入设备组态的“输入滤波器”(Input filter) 中设置滤波时间。

指定的滤波时间必须小于回原点开关的输入信号持续时间。

## 选择信号电平

在下拉列表中，选择回原点时使用的回原点开关电平。

## 允许在硬限位开关处自动反向

激活该复选框可将硬限位开关用作回原点过程中的反向凸轮。只有启用硬限位开关才能实现反向控制（必须至少组态位于逼近方向上的硬限位开关）。

如果在主动回原点过程中到达硬限位开关，轴将以组态的减速度（不是以急停减速度）制动，然后反向。然后反向检测回原点开关。

如果未激活反向功能且在主动回原点过程中轴到达硬限位开关，则将因错误而中止回原点过程并以急停减速度对轴进行制动。

### 说明

如果可能，采用以下措施之一以确保机器在发生反向时不会行进到机械挡块：

- 保持较低的行进速度。
- 增加组态的加速度/减速度。
- 增加硬限位开关和机械挡块之间的距离。

## 逼近/回原点方向

通过方向选择，可以决定主动回原点过程中搜索回原点开关的逼近方向以及回原点的方向。回原点方向指定执行回原点操作时轴用于逼近组态的回原点开关端的行进方向。

## 回原点开关侧

在此处可以选择轴是在回原点开关的下侧还是上侧进行回原点。

## 逼近速度

在该域中，可以指定回原点期间搜索回原点开关的速度。

限值（与所选测量单位无关）：

- 启动/停止速度  $\leq$  逼近速度  $\leq$  最大速度

## 回原点速度

在该域中，可以指定回原点期间逼近回原点开关的速度。

限值（与所选测量单位无关）：

- 启动/停止速度  $\leq$  回原点速度  $\leq$  最大速度

## 起始位置偏移值

如果指定的回原点位置与回原点开关的位置存在偏差，则可在此域中指定起始位置偏移量。

如果该值不等于 0，轴在回原点开关处回原点后将执行以下动作：

1. 以回原点速度使轴移动起始位置偏移值指定的一段距离
2. 达到“起始位置偏移值”时，轴处于运动控制指令“MC\_Home”的输入参数“Position”中指定的起始位置处。

限值（与所选测量单位无关）：

- $-1.0e12 \leq$  起始位置偏移值  $\leq 1.0e12$

## 起始位置

将运动控制指令“MC\_Home”中所组态的位置用作起始位置。

### 12.11.3 诊断 - 状态和错误位 (“轴”工艺对象 V1...3)

在 TIA Portal 中使用诊断功能“状态和错误位”(Status and error bits) 可监视轴的最重要状态和错误消息。当轴激活时，可以在“手动控制”模式和“自动控制”模式下在线显示诊断功能。状态错误消息具有下列含义：

#### 轴的状态

状态	说明
已启用	轴已启用且准备就绪，可通过运动控制命令进行控制。 (工艺对象的变量: <轴名称>.StatusBits.Enable)
已回原点	轴已回原点，可执行运动控制指令“MC_MoveAbsolute”的绝对定位命令。对于相对定位而言，轴不必回原点。特殊情况： <ul style="list-style-type: none"> <li>• 主动回原点过程中，该状态为 FALSE。</li> <li>• 如果回原点的轴经受被动回原点，则在被动回原点过程中该状态设置为 TRUE。</li> </ul> (工艺对象的变量: <轴名称>.StatusBits.HomingDone)
轴错误	“轴”工艺对象发生错误。有关错误的详细信息，请参见自动控制模式下的运动控制指令的 ErrorID 和 ErrorInfo 参数。在手动模式下，轴控制面板的“错误消息”(Error message) 框可显示有关错误原因的详细信息。 (工艺对象的变量: <轴名称>.StatusBits.Error)
轴控制面板已启用	在轴控制面板中启用了“手动控制”(Manual control) 模式。轴控制面板对“轴”工艺对象具有优先控制权。不能通过用户程序来控制轴。 (工艺对象的变量: <轴名称>.StatusBits.ControlPanelActive)
必须重新启动	已在 CPU RUN 操作模式下将已修改的轴组态下载到装载存储器。要将已修改组态下载到工作存储器，需要重新启动轴。使用运动控制指令 MC_Reset 执行此操作。

#### 驱动器状态

状态	说明
驱动器准备就绪	驱动装置已就绪，可以运行。 (工艺对象的变量: <轴名称>.StatusBits.DriveReady)
驱动装置错误	驱动装置因“驱动装置就绪”信号丢失而报错。 (工艺对象的变量: <轴名称>.ErrorBits.DriveFault)

## 轴运动的状态

状态	说明
停止	轴处于停止状态。 (工艺对象的变量: <轴名称>.StatusBits.StandStill)
加速度	轴在加速。 (工艺对象的变量: <轴名称>.StatusBits.Acceleration)
恒速	轴在恒速运转。 (工艺对象的变量: <轴名称>.StatusBits.ConstantVelocity)
减速度	轴在减速(速度下降)。 (工艺对象的变量: <轴名称>.StatusBits.Deceleration)

## 运动模式的状态

状态	说明
定位	轴执行运动控制指令“MC_MoveAbsolute”、“MC_MoveRelative”或者轴控制面板的定位命令。 (工艺对象的变量: <轴名称>.StatusBits.PositioningCommand)
以速度设定值行进	轴以运动控制指令“MC_MoveVelocity”、“MC_MoveJog”或者轴控制面板的速度设定值执行命令。 (工艺对象的变量: <轴名称>.StatusBits.SpeedCommand)
回原点	轴将执行运动控制指令“MC_Home”或者轴控制面板的回原点命令。 (工艺对象的变量: <轴名称>.StatusBits.Homing)
命令表已激活 (从 V2.0 轴工艺对象起)	该轴由运动控制指令“MC_CommandTable”控制。 (工艺对象的变量: <轴名称>.StatusBits.CommandTableActive)

## 错误消息

错误	说明
已到达下限软限位开关	已到达下限软限位开关。 (工艺对象的变量: <轴名称>.ErrorBits.SwLimitMinReached)
已超出下限软限位开关	已超出下限软限位开关。 (工艺对象的变量: <轴名称>.ErrorBits.SwLimitMinExceeded)
已到达上限软限位开关	已到达上限软限位开关。 (工艺对象的变量: <轴名称>.ErrorBits.SwLimitMaxReached)
已超出上限软限位开关	已超出上限软限位开关。 (工艺对象的变量: <轴名称>.ErrorBits.SwLimitMaxExceeded)
已到达下限硬限位开关	已到达下限硬限位开关。 (工艺对象的变量: <轴名称>.ErrorBits.HwLimitMin)
已到达上限硬限位开关	已到达上限硬限位开关。 (工艺对象的变量: <轴名称>.ErrorBits.HwLimitMax)
PTO 和 HSC 在使用中	另一个轴正在使用此 PTO (Pulse Train Output) 和 HSC (High Speed Counter) 并且该轴已通过“MC_Power”启用。 (工艺对象的变量: <轴名称>.ErrorBits.HwUsed)
组态错误	错误地组态了“轴”工艺对象，或者在用户程序运行期间错误地修改了可编辑的组态数据。 (工艺对象的变量: <轴名称>.ErrorBits.ConfigFault)
内部错误	发生内部错误。 (工艺对象的变量: <轴名称>.ErrorBits.SystemFault)

## 参见

状态和错误位（自工艺对象 V4 起） (页 221)

## 12.11.4 ErrorID 和 ErrorInfo

### 12.11.4.1 ErrorID 和 ErrorInfo 列表 (工艺对象 V4...5)

下表列出了运动控制指令中指示的所有 ErrorID 和 ErrorInfo。除了错误原因，还列出了消除错误的补救措施。

根据错误响应，轴在停止时如果发生操作错误，将会停止轴。可能会出现以下错误响应：

- 取消启用

输出设定点 0，并取消启用。轴将根据驱动器中的组态进行制动，并转入停止状态。

- 通过急停斜坡功能进行停止

将中止处于激活状态的运动命令。轴将通过在“工艺对象 > 扩展参数 > 动态 > 急停斜坡功能”(Technology object > Extended parameters > Dynamics > Emergency stop ramp) 中所组态的急停减速度功能进行制动（没有任何加加速度的限制），并转入停止状态。

#### 伴随轴停止的运行错误

ErrorID	ErrorInfo	说明	解决方法	错误反应
16#8000		驱动器错误，丢失“驱动器就绪”信号	-	
	16#0001	-	使用指令“MC_Reset”确认错误；提供驱动器信号；根据需要重新启动命令	
16#8001		已触发软限位开关的下限		
	16#000E	以当前所组态的减速度到达软限位开关的下限位置	使用“MC_Reset”指令确认错误；使用运动命令使轴沿正向移动，超出软限位开关的范围	
	16#000F	以急停减速度到达软限位开关的下限位置		
	16#0010	以急停减速度超出软限位开关的下限位置		取消启用

ErrorID	ErrorInfo	说明	解决方法	错误反应
16#8002		已触发软限位开关的上限		-
	16#00 0E	以当前所组态的减速度到达软限位开关的上限位置	使用“MC_Reset”指令确认错误；使用运动命令使轴沿负向移动，超出软限位开关的范围	取消启用
	16#00 0F	以急停减速度到达软限位开关的上限位置		
	16#00 10	以急停减速度超出软限位开关的上限位置		
16#8003		已到达硬限位开关的下限		通过 PTO (Pulse Train Output) 的驱动器连接： 通过急停斜坡功能进行停止 通过 PROFIdrive/模拟量输出进行驱动装置连接： 取消启用
	16#00 0E	已到达该硬限位开关的下限。轴以急停减速度停止。  (在主动回原点过程中，未找到回原点开关)	使用指令“MC_Reset”确认已启用轴的错误；使用运动命令使轴沿正向移动，超出硬限位开关的范围。	
16#8004		已到达硬限位开关的上限		通过 PTO (Pulse Train Output) 的驱动器连接： 通过急停斜坡功能进行停止 通过 PROFIdrive/模拟量输出进行驱动装置连接： 取消启用
	16#00 0E	已到达该硬限位开关的上限。轴以急停减速度停止。  (在主动回原点过程中，未找到回原点开关)	使用指令“MC_Reset”确认已启用轴的错误；使用运动命令使轴沿负向移动，超出硬限位开关的范围。	

ErrorID	ErrorInfo	说明	解决方法	错误反应
16#8005		PTO/HSC 已被另一个轴占用		-
	16#00 01	-	<p>轴未正确组态: 更正 PTO (Pulse Train Output)/HSC (High Speed Counter) 的组态并将其下载到控制器</p> <p>多个轴将通过一个 PTO 运行: 另一个轴正在使用 PTO / HSC。如果要控制当前轴，则必须使用“MC_Power”(Enable = FALSE) 禁用另一个轴。 (另请参见将多个轴与同一个 PTO 一起使用 (页 283))</p>	
16#8006		轴控制面板中发生通信错误		取消启用
	16#00 12	已超时	检查电缆连接，然后再次按下“手动控制”(Manual control) 按钮。	
16#8007		该轴无法启用		-
	16#00 25	正在重新启动	请等待，直到轴重新启动完成。	
	16#00 26	正在 RUN 模式下执行加载过程	请等待，直到加载过程完成。	
16#8008		无效的运动方向		-
	16#00 2E	不允许所选的运动方向。	<ul style="list-style-type: none"> <li>调整运动方向，然后重新启动该命令。</li> </ul>	
	16#00 2F	不能将所选的运动方向反转。	<ul style="list-style-type: none"> <li>在工艺对象组态的“扩展参数 &gt; 机械”(Extended parameters &gt; Mechanics) 下可调整允许的旋转方向。重新启动命令。</li> </ul>	

ErrorID	ErrorInfo	说明	解决方法	错误反应
<b>16#8009</b>		未找到参考开关/编码器零位标记		
	16#00 33	组态、编码器的硬件或安装、回原点开关中出现错误。	<ul style="list-style-type: none"> <li>连接合适的设备。</li> <li>检查设备 (I/O)。</li> <li>比较 HW Config 中的组态和工艺对象。</li> </ul>	通过急停斜坡功能进行停止
<b>16#800A</b>		编码器发出报警消息		
	16#00 01	-	检查设备的功能、连接和 I/O。	取消启用
	16#00 34	编码器中的硬件错误		
	16#00 35	编码器变脏		
	16#00 36	读取编码器绝对值期间出错	将驱动装置中的编码器类型或编码器参数 P979 与工艺对象的组态数据进行比较。	
	16#00 37	监视编码器的零位标记	<p>编码器报告零位标记监视错误 (Gx_XIST2 中的错误代码 0x0002, 请参见 PROFIdrive 配置文件)。</p> <p>检查设备的电磁兼容性 (EMC)。</p>	
	16#00 38	编码器处于“停止”(Parking) 状态	<ul style="list-style-type: none"> <li>有关错误原因, 请查看所连接的驱动装置或编码器。</li> <li>检查错误消息是否可能由驱动器或编码器的调试操作触发。</li> </ul>	
	16#00 40	PROFIdrive: 总线上的编码器故障 (站故障)。	检查设备的功能、连接和 I/O。	

ErrorID	ErrorInfo	说明	解决方法	错误反应	
<b>16#800B</b>		<b>位置超出范围</b>			
	16#00 39	正方向超出范围	使轴回到有效实际值范围。	取消启用	
	16#00 3A	负方向超出范围			
	16#00 3B	位置控制时钟周期中实际位置的改变大于周期长度。			
<b>16#800C</b>		<b>驱动器发出报警消息</b>			
	16#00 01	-	检查设备的功能、连接和 I/O。 在“调整”(Tuning) 对话框中，选择较小的增益 (Kv)。	取消启用	
	16#00 3C	PROFIdrive：驱动器信号“请求的控制”故障。			
	16#00 3D	PROFIdrive/模拟驱动器接口：驱动器已关闭。			
	16#00 3E	PROFIdrive：总线上的驱动器故障（站故障）。			
<b>16#800D</b>		<b>超出了允许跟随误差。</b>			
	16#00 01	-	<ul style="list-style-type: none"> <li>• 检查控制回路的组态。</li> <li>• 检查编码器的信号方向。</li> <li>• 检查跟随误差监控的组态。</li> </ul>	取消启用	
<b>16#800E</b>		<b>硬限位开关处发生错误</b>			
	16#00 42	激活硬限位开关的任意行进方向非法	由于激活了硬限位开关，将禁用编程的运动方向。 向相反方向缩回轴。	取消启用	
	16#00 43	硬限位开关极性反转，轴无法释放	检查硬限位开关的机械组态。		
	16#00 44	两个硬限位开关均激活，轴无法释放			

ErrorID	ErrorInfo	说明	解决方法	错误反应
<b>16#800F</b>		目标范围发生错误		
	16#00 45	未达到目标范围	在定位容差时间内未达到目标范围。 <ul style="list-style-type: none"><li>• 检查定位监控的组态。</li><li>• 检查控制回路的组态。</li></ul>	取消启用
	16#00 46	再次离开目标范围	在最短停留时间内已离开目标范围。 <ul style="list-style-type: none"><li>• 检查定位监控的组态。</li><li>• 检查控制回路的组态。</li></ul>	
<b>16#8010</b>		如果该轴不是一个模数轴，则软限位开关的下限位置将大于该软限位开关的上限位置		
	16#00 01	-	更改软限位开关的位置。	取消启用

### 不伴随轴停止的运行错误

ErrorID	ErrorInfo	说明	解决方法
<b>16#8200</b>		轴未启用	
	16#000 1	-	启用轴：重新启动命令
<b>16#8201</b>		轴已由另一个“MC_Power”实例启用	
	16#000 1	-	仅通过一个“MC_Power”实例启用轴
<b>16#8202</b>		同时执行的运动控制命令最大数量（对于通过 PTO (Pulse Train Output) 的驱动器连接，最多同时运行 200 条命令，对于通过 PROFIdrive/模拟量输出的驱动器连接，最多同时运行 100 条命令）	
	16#000 1	-	减少并行激活命令数；重新启动命令 如果运动控制指令的参数“Busy”= TRUE，则说明命令已激活。

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8203</b>		轴当前在“手动控制”(Manual control) 模式 (轴控制面板) 下运行		
	16#000 1	-	退出“手动控制”；重新启动命令	
<b>16#8204</b>		轴未回原点		
	16#000 1	-	使用指令“MC_Home”使轴回原点；重新启动命令	
<b>16#8205</b>		轴当前由用户程序控制 (该错误仅显示在轴控制面板中)		
	16#001 3	轴已在用户程序中启用	使用指令“MC_Power”禁用轴或者在轴控制面板中再次选择“手动控制”(Manual control)	
<b>16#8206</b>		工艺对象尚未激活		
	16#000 1	-	使用指令“MC_Power”Enable = TRUE 启用轴或者在轴控制面板中启用轴。	
<b>16#8207</b>		命令被拒绝		
	16#001 6	正在执行主动回原点；无法启动另一种回原点方法。	等到主动回原点完成或通过运动命令（例如，“MC_Halt”）中止主动回原点。	
	16#001 8	当轴正在主动或被动回原点时，不能使用命令表进行移动。	一直等到直接地或被动回原点完成。	
	16#001 9	正在处理命令表时，轴不能主动或被动回原点。	等到命令表完成或通过运动命令（例如，“MC_Halt”）中止命令表。	
	16#005 2	指定的位置超出了数值限制。	在运动控制指令中输入有效的位置值。	
	16#005 3	轴正在沿斜坡上升。	等到轴准备就绪，可运行。	
	16#005 4	实际值无效	要执行“MC_Home”命令，实际值必须有效。 请检查实际值的状态。工艺对象 <轴名称>.“StatusSensor.State”变量的值必须为 2 (有效值)。	

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8208</b>		<b>最大速度和启动/停止速度的差值无效</b>		
	16#000 2	值的数字格式无效。	更正该值；重新启动命令	
	16#000 A	值小于或等于 0。		
<b>16#8209</b>		<b>对工艺对象“轴”的加速度无效</b>		
	16#000 2	值的数字格式无效。	更正该值；重新启动命令	
	16#000 A	值小于或等于 0。		
<b>16#820A</b>		<b>无法重新启动轴</b>		
	16#001 3	轴已在用户程序中启用	使用“MC_Power”指令阻止轴；再次重新启动	
	16#002 7	轴当前在“手动控制”模式（轴控制面板）下运行	退出“手动控制”；再次重新启动	
	16#002 C	轴未禁用。	禁用轴；重新启动命令	
	16#004 7	工艺对象未准备好重新启动。	重新下载项目。	
	16#004 8	不满足工艺对象的重新启动条件。	禁用工艺对象。	
<b>16#820B</b>		<b>无法执行命令表</b>		
	16#002 6	正在 RUN 模式下执行加载过程	请等待，直到加载过程完成。	
<b>16#820C</b>		<b>无组态可用</b>		
	16#000 1	-	内部错误 联系服务热线。	

## 块参数错误

ErrorID	ErrorInfo	说明	解决方法
<b>16#8400</b>		运动控制指令的“Position”参数值无效	
	16#000 2	值的数字格式无效。	更正该值；重新启动命令
	16#000 5	该值超出值范围（大于 1E+12）	
	16#000 6	该值超出值范围（小于 -1E+12）	
<b>16#8401</b>		运动控制指令的“Distance”参数值无效	
	16#000 2	值的数字格式无效。	更正该值；重新启动命令
	16#000 5	该值超出值范围（大于 1E+12）	
	16#000 6	该值超出值范围（小于 -1E+12）	
<b>16#8402</b>		运动控制指令的“Velocity”参数值无效	
	16#000 2	值的数字格式无效。	更正该值；重新启动命令
	16#000 8	值大于组态的最大速度	
	16#000 9	值小于组态的启动/停止速度	
	16#002 4	值小于 0	
<b>16#8403</b>		运动控制指令的“Direction”参数值无效	
	16#001 1	所选值无效	更正所选值；重新启动命令

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8404</b>		运动控制指令的“Mode”参数值无效		
	16#001 1	所选值无效	更正所选值；重新启动命令	
	16#001 5	未组态主动/被动回原点	更正组态并将其下载到控制器；启用轴并重新启动命令	
	16#001 7	虽然硬限位开关已禁用，但还是激活了在硬限位开关处反向	<ul style="list-style-type: none"> <li>使用变量 &lt;轴名&gt;.PositionLimitsHW.Active = TRUE 激活硬限位开关，并重新启动该命令。</li> <li>更正组态并将其下载到控制器；启用轴并重新启动命令</li> </ul>	
	16#005 5	增量编码器的模式无效	需使用参数“Mode”= 0、1、2、3 启动增量编码器的回原点操作。	
	16#005 6	绝对值编码器的模式无效	<p>绝对值编码器无法执行被动和主动回原点操作（“Mode”= 2、3）。</p> <p>需使用参数“Mode”= 0、1 启动绝对值编码器的回原点操作。</p>	
<b>16#8405</b>		运动控制指令的“StopMode”参数值无效		
	16#001 1	所选值无效	更正所选值；再次启用轴	
<b>16#8406</b>		不允许同时正向和反向点动		
	16#000 1	-	采取措施确保参数“JogForward”和“JogBackward”的信号状态不会同时为 TRUE；重新启动命令。	
<b>16#8407</b>		只有禁用激活的轴后，才允许使用指令“MC_Power”切换到另一个轴。		
	16#000 1	-	禁用激活的轴；然后可以切换到其它轴并启用该轴。	

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8408</b>		运动控制指令的“Axis”参数值无效		
	16#001 A	指定的值与所需的工艺对象版本不匹配	更正该值；重新启动命令	
	16#001 B	指定的值与所需的工艺对象类型不匹配		
	16#001 C	指定的值不是运动控制工艺数据块		
<b>16#8409</b>		运动控制指令的“CommandTable”参数值无效		
	16#001 A	指定的值与所需的工艺对象版本不匹配	更正该值；重新启动命令	
	16#001 B	指定的值与所需的工艺对象类型不匹配		
	16#001 C	指定的值不是运动控制工艺数据块		
<b>16#840A</b>		运动控制指令的“StartStep”参数值无效		
	16#000 A	值小于或等于 0。	更正该值；重新启动命令	
	16#001 D	开始步进大于结束步进		
	16#001 E	值大于 32		
<b>16#840B</b>		运动控制指令的“EndStep”参数值无效		
	16#000 A	值小于或等于 0。	更正该值；重新启动命令	
	16#001 E	值大于 32		

ErrorID	ErrorInfo	说明	解决方法	
<b>16#840C</b>		运动控制指令的“RampUpTime”参数值无效		
	16#000 2	值的数字格式无效。	更正该值；重新启动命令	
	16#000 5	该值超出值范围（大于 1.0E12）		
	16#000 A	值小于或等于 0。		
<b>16#840D</b>		运动控制指令的“RampDownTime”参数值无效		
	16#000 2	值的数字格式无效。	更正该值；重新启动命令	
	16#000 A	值小于或等于 0。		
<b>16#840E</b>		运动控制指令的“EmergencyRampTime”参数值无效		
	16#000 2	值的数字格式无效。	更正该值；重新启动命令	
	16#000 A	值小于或等于 0。		
<b>16#840F</b>		运动控制指令的“JerkTime”参数值无效		
	16#000 2	值的数字格式无效。	更正该值；重新启动命令	
	16#000 A	值小于或等于 0。		

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8410</b>		<b>运动控制指令的“Parameter”参数值无效</b>		
	16#000 2	值的数字格式无效。	更正该值；重新启动命令	
	16#000 B	地址无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
	16#002 8	VARIANT 指针“参数”和“值”的数据类型不匹配。	使用适当的数据类型；重新启动命令。	
	16#002 9	VARIANT 指针“参数”并未指向工艺对象的数据块。	更正 VARIANT 指针；重新启动命令	
	16#002 A	无法读取 VARIANT 指针“参数”值。	更正 VARIANT 指针；重新启动命令	
	16#002 B	无法写入 VARIANT 指针“参数”值。	更正 VARIANT 指针或值；重新启动命令	
	16#002 C	轴未禁用。	禁用轴；重新启动命令	
<b>16#8411</b>		<b>运动控制指令的“Value”参数值无效</b>		
	16#000 2	值的数字格式无效。	更正该值；重新启动命令	

## 轴的组态错误

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8600</b>		脉冲发生器 (PTO) 的参数分配无效		
	16#000 B	地址无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
	16#001 4	所选硬件由另一应用程序使用		
<b>16#8601</b>		高速计数器 (HSC) 的参数分配无效		
	16#000 B	地址无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
	16#001 4	所选硬件由另一应用程序使用		
<b>16#8602</b>		“使能输出”(Enable output) 的参数分配无效		
	16#000 B	地址无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
<b>16#8603</b>		“输入就绪”(Ready input) 的参数分配无效		
	16#000 B	地址无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
<b>16#8604</b>		“电机每转的脉冲数”(Pulses per motor revolution) 值无效		
	16#000 A	值小于或等于 0	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	

ErrorID	ErrorInfo	说明	解决方法
<b>16#8605</b>		<b>“每转的距离”(Distance per revolution) 值无效</b>	
	16#0002	值的数字格式无效。	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴
	16#0005	该值超出值范围（大于 1E+12）	
	16#000A	值小于或等于 0	
	16#0030	值的数字格式不正确，或者超出有效的数字范围	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
<b>16#8606</b>		<b>“启动/停止速度”(Start / stop velocity) 值无效</b>	
	16#0002	值的数字格式无效。	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴
	16#0003	值大于硬件上限值	
	16#0004	值小于硬件下限值	
	16#0007	启动/停止速度大于最大速度	

ErrorID	ErrorInfo	说明	解决方法
<b>16#8607</b>		<b>“最大速度”(maximum velocity) 值无效</b>	
	16#000 2	值的数字格式无效。	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴
	16#000 3	值大于硬件上限值	
	16#000 4	值小于硬件下限值	
	16#003 0	值的数字格式不正确，或者超出有效的数字范围	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
<b>16#8608</b>		<b>“加速度”(Acceleration) 值无效</b>	
	16#000 2	值的数字格式无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
	16#000 3	值大于硬件上限值	
	16#000 4	值小于硬件下限值	
	16#003 0	值的数字格式不正确，或者超出有效的数字范围	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>

ErrorID	ErrorInfo	说明	解决方法
<b>16#8609</b>		<b>“减速度”(Deceleration) 值无效</b>	
	16#000 2	值的数字格式无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
	16#000 3	值大于硬件上限值	
	16#000 4	值小于硬件下限值	
	16#003 0	值的数字格式不正确，或者超出有效的数字范围	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
<b>16#860A</b>		<b>“急停减速度”(Emergency stop deceleration) 值无效</b>	
	16#000 2	值的数字格式无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
	16#000 3	值大于硬件上限值	
	16#000 4	值小于硬件下限值	
	16#003 0	值的数字格式不正确，或者超出有效的数字范围	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>

ErrorID	ErrorInfo	说明	解决方法	
<b>16#860B</b>		软限位开关的下限位置值无效		
	16#000 2	值的数字格式无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
	16#000 5	该值超出值范围（大于 1E+12）		
	16#000 6	该值超出值范围（小于 -1E+12）		
	16#003 0	软限位开关的下限位置值大于该软限位开关的上限位置值		
<b>16#860C</b>		软限位开关的上限位置值无效		
	16#000 2	值的数字格式无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
	16#000 5	该值超出值范围（大于 1E+12）		
	16#000 6	该值超出值范围（小于 -1E+12）		
	16#003 0	值的数字格式不正确，或者超出有效的数字范围		
<b>16#860D</b>		硬限位开关的下限地址无效		
	16#000 B	地址无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
	16#000 C	下降沿的地址无效		
	16#000 D	上升沿的地址无效		

ErrorID	ErrorInfo	说明	解决方法	
<b>16#860E</b>		<b>硬限位开关的上限地址无效</b>		
	16#000 B	地址无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
	16#000 C	下降沿的地址无效		
	16#000 D	上升沿的地址无效		
<b>16#860F</b>		<b>“起始位置偏移值”(home position offset) 无效</b>		
	16#000 2	值的数字格式无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
	16#000 5	该值超出值范围（大于 1E+12）		
	16#000 6	该值超出值范围（小于 -1E+12）		
	16#003 0	值的数字格式不正确，或者超出有效的数字范围	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#8610</b>		<b>“逼近速度”(approach velocity) 值无效</b>		
	16#000 2	值的数字格式无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
	16#000 8	速度大于最大速度		
	16#000 9	速度小于启动/停止速度		
	16#003 0	值的数字格式不正确，或者超出有效的数字范围	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8611</b>		“回原点速度”(Homing velocity) 值无效		
	16#000 2	值的数字格式无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
	16#000 8	速度大于最大速度		
	16#000 9	速度小于启动/停止速度	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
	16#003 0	值的数字格式不正确，或者超出有效的数字范围		
<b>16#8612</b>		回原点开关的地址无效		
	16#000 B	地址无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
	16#000 C	下降沿的地址无效		
	16#000 D	上升沿的地址无效		
<b>16#8613</b>		虽然未组态硬限位开关，但还是在主动回原点过程中，激活了在该硬限位开关处反向		
	16#000 1	-	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	

ErrorID	ErrorInfo	说明	解决方法
<b>16#8614</b>		<b>“加加速度”(Jerk) 值无效</b>	
	16#0002	值的数字格式无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
	16#001F	值大于最大加加速度值	
	16#0020	值小于最小加加速度值	
	16#0030	值的数字格式不正确，或者超出有效的数字范围	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
<b>16#8615</b>		<b>“测量单位”的值无效</b>	
	16#0011	所选值无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴
<b>16#8616</b>		<b>回原点开关的地址无效（自版本 V4 起的被动回原点）</b>	
	16#0011	所选值无效	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
	16#0030	值的数字格式不正确，或者超出有效的数字范围	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
<b>16#8617</b>		<b>变量 &lt;轴名称&gt;.Sensor.Sensor[1].ActiveHoming.Mode 的值无效</b>	
	16#0011	所选值无效 (有效值：2 = 通过数字量输入回原点)	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8618</b>		变量 <轴名称>.Sensor.Sensor[1].PassiveHoming.Mode 的值无效		
	16#001 1	所选值无效 (有效值: 2 = 通过数字量输入回原点)	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器; 使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值; 使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#8619</b>		变量 <轴名称>.Actor.Type 的值无效		
	16#001 1	所选值无效 (有效值: 2 = 通过脉冲接口连接)	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器; 使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值; 使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#861A</b>		“允许的旋转方向”(Permitted direction of rotation) 值无效		
	16#001 1	所选值无效	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器; 使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值; 使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
	16#002 D	方向输出关断时, 不允许“双向”操作。		
<b>16#861B</b>		负载齿轮因数错误		
	16#003 1	值无效。	将无错误组态下载到控制器; 使用指令“MC_Power”再次启用轴	
<b>16#861C</b>		通过增量编码器回原点的数据组合非法		
	16#003 1	值无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器; 使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值; 使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	

ErrorID	ErrorInfo	说明	解决方法	
<b>16#861D</b>		设置的编码器安装方式无效。<轴名称>.Sensor.Sensor[1].MountingMode 中的值无效		
	16#001 1	所选值无效	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#861E</b>		测量编码器轮周长的组态无效。<轴名称>.Sensor.Sensor[1].Parameter.DistancePerRevolution 中的值无效		
	16#003 0	值的数字格式不正确，或者超出有效的数字范围	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#861F</b>		组态的线性编码器精度错误。<轴名称>.Sensor.Sensor[1].Parameter.Resolution 中的值无效		
	16#003 0	值的数字格式不正确，或者超出有效的数字范围	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#8620</b>		组态的 Gx_XIST1 高分辨率无效。<轴名称>.Sensor.Sensor[1].Parameter.FineResolutionXist1 中的值无效		
	16#003 0	值的数字格式不正确，或者超出有效的数字范围	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	

ErrorID	ErrorInfo	说明	解决方法
16#8621		在 <轴名称>.Sensor.Sensor[1].Parameter.FineResolutionXist1 中设置的 Gx_XIST1 高精度值与 PROFIdrive 参数 P979 中的设置不一致	
	16#0030	值的数字格式不正确，或者超出有效的数字范围	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
16#8622		组态日期 <轴名称>.Actor.Interface.AddressIn 或 <轴名称>.Actor.Interface.AddressOut 的值无效	
	16#0011	所选值无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴
16#8623		变量 <轴名称>.Sensor.Sensor[1].Type 中设置的值无效。	
	16#0011	所选值无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴
16#8624		设置的编码器系统无效。<轴名称>.Sensor.Sensor[1].System 中的值无效	
	16#0011	所选值无效	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
16#8625		定位监控的参数错误。<轴名称>.PositioningMonitoring.MinDwellTime 中的值无效	
	16#0030	值的数字格式不正确，或者超出有效的数字范围	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
16#8626		定位监控的参数错误。<轴名称>.PositioningMonitoring.Window 中的值无效	
	16#0030	值的数字格式不正确，或者超出有效的数字范围	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>

ErrorID	ErrorInfo	说明	解决方法
16#8627		PROFIdrive 接口组态的实际值错误。<轴名称>.Sensor.Sensor[1].Interface.AddressIn 或 <轴名称>.Sensor.Sensor[1].Interface.AddressOut 中的值无效	
	16#001 1	所选值无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴
16#8628		控制器数错误	
	16#003 0	值的数字格式不正确，或者超出有效的数字范围	<p>控制回路的增益或预控制值错误。</p> <ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正错误值；使用指令“MC_Reset”确认错误并在必要时重新启动该命令（PositionControl.Kv、&lt;轴名称&gt;.PositionControl.Kpc、）</li> </ul>
16#8629		停止信号的限值错误。<轴名称>.StandStillSignal.VelocityThreshold 中的值无效	
	16#003 0	值的数字格式不正确，或者超出有效的数字范围	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
16#862A		定位监控的参数错误。<轴名称>.PositioningMonitoring.ToleranceTime 中的值无效	
	16#003 0	值的数字格式不正确，或者超出有效的数字范围	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
16#862B		PROFIBUS 参数化不一致；Ti 和 To 的总和大于一个 DP 周期	
	16#003 0	值的数字格式不正确，或者超出有效的数字范围	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴

ErrorID	ErrorInfo	说明	解决方法	
<b>16#862C</b>		停止监视的参数错误。<轴名称>.StandStillSignal.MinDwellTime 中的值无效		
	16#0030	值的数字格式不正确，或者超出有效的数字范围	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#862D</b>		跟随误差监控的参数错误。<轴名称>.FollowingError.MinValue 中的值无效		
	16#0030	值的数字格式不正确，或者超出有效的数字范围	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#862E</b>		<轴名称>.Modulo.Length 中组态日期无效		
	16#0030	值的数字格式不正确，或者超出有效的数字范围	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#862F</b>		<轴名称>.Modulo.StartValue 中组态日期无效		
	16#0030	值的数字格式不正确，或者超出有效的数字范围	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#8630</b>		<轴名称>.Actor.DriveParameter.ReferenceSpeed 中组态日期无效		
	16#0030	值的数字格式不正确，或者超出有效的数字范围	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	

ErrorID	ErrorInfo	说明	解决方法
16#8631		设置的 Gx_XIST2 高分辨率无效。<轴名称>.Sensor.Sensor[1].Parameter.FineResolutionXist2 中的值无效	
	16#0030	值的数字格式不正确，或者超出有效的数字范围	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
16#8632		可确定的编码器精度值无效。<轴名称>.Sensor.Sensor[1].Parameter.DeterminableRevolutions 中的值无效	
	16#0030	值的数字格式不正确，或者超出有效的数字范围	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
16#8633		指定的被动回原点的回原点开关逼近方向无效。<轴名称>.Sensor.Sensor[1].PassiveHoming.Direction 中的值无效	
16#8634		跟随误差监控的参数错误。<轴名称>.FollowingError.MaxValue 中的值无效	
	16#0030	值的数字格式不正确，或者超出有效的数字范围	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
16#8635		跟随误差监控的参数错误。<轴名称>.FollowingError.MinVelocity 中的值无效	
	16#0030	值的数字格式不正确，或者超出有效的数字范围	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8636</b>		控制器因数不正确。<轴名称>.PositionControl.Kpc 中预控制因数的值无效		
	16#0030	值的数字格式不正确，或者超出有效的数字范围	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#8637</b>		<轴名称>. 中组态日期无效 Sensor.Sensor[1].Interface.Type		
	16#0011	所选值无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
<b>16#8638</b>		<轴名称>. 中组态日期无效 Sensor.Sensor[1].Interface.HSC		
	16#0011	所选值无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
<b>16#8639</b>		驱动器上有错误		
	16#0049	设备的组态错误	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
	16#004A	工艺需要一个更小的伺服时钟。	内部系统错误。 检查项目的一致性并将项目重新加载到控制器中。	
	16#004B	斜坡上升过程中未初始化设备驱动程序。	要启用工艺对象，必须完成执行器驱动程序的初始化。 请稍后再执行该命令。	
<b>16#863A</b>		与驱动器的通信错误		
	16#004C	设备的组态错误	<ul style="list-style-type: none"> <li>连接合适的设备。</li> <li>检查设备 (I/O)。</li> <li>比较 HW Config 中的组态和工艺对象。</li> </ul>	
	16#004D	设备驱动程序需要一个更小的伺服时钟。	<ul style="list-style-type: none"> <li>连接合适的设备。</li> <li>检查设备 (I/O)。</li> <li>比较 HW Config 中的组态和工艺对象。</li> </ul>	
	16#004E	与设备进行内部通信时发生错误	检查项目的一致性并将项目重新加载到控制器中。	

ErrorID	ErrorInfo	说明	解决方法
<b>16#863B</b>		<b>编码器中的错误</b>	
	16#004 9	设备的组态错误	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴
	16#004 A	工艺需要一个更小的伺服时钟。	内部系统错误。 检查项目的一致性并将项目重新加载到控制器中。
	16#004 B	斜坡上升过程中未初始化设备驱动程序。	要启用工艺对象，必须完成执行器驱动程序的初始化。 请稍后再执行该命令。
<b>16#863C</b>		<b>与编码器的通信错误</b>	
	16#004 C	设备的组态错误	<ul style="list-style-type: none"> <li>连接合适的设备。</li> <li>检查设备 (I/O)。</li> <li>比较 HW Config 中的组态和工艺对象。</li> </ul>
	16#004 D	设备驱动程序需要一个更小的伺服时钟。	<ul style="list-style-type: none"> <li>连接合适的设备。</li> <li>检查设备 (I/O)。</li> <li>比较 HW Config 中的组态和工艺对象。</li> </ul>
	16#004 E	与设备进行内部通信时发生错误	检查项目的一致性并将项目重新加载到控制器中。
<b>16#863D</b>		<b>与设备（驱动器或编码器）的通信错误</b>	
	16#003 0	值的数字格式不正确，或者超出有效的数字范围	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
	16#005 5	请求的逻辑地址无效。	<ul style="list-style-type: none"> <li>连接合适的设备。</li> <li>检查设备 (I/O)。</li> </ul>
	16#005 6	请求的逻辑输出地址无效。	<ul style="list-style-type: none"> <li>检查 HW Config 中的拓扑组态。</li> <li>比较 HW Config 中的组态和工艺对象。</li> </ul>
	16#005 7	请求的逻辑输出地址无效。	

ErrorID	ErrorInfo	说明	解决方法
<b>16#863E</b>		变量“ControlPanel.Input.TimeOut”的值无效（轴控制面板）	
	16#0030	值的数字格式不正确，或者超出有效的数字范围	更正工艺对象 <轴名称>.ControlPanel.Input.TimeOut 的变量值。 以毫秒为单位指定该值。
<b>16#863F</b>		<轴名称>. 中组态日期无效 Actor.DriveParameter.MaxSpeed	
	16#0030	值的数字格式不正确，或者超出有效的数字范围	将驱动器和工艺对象组态中的参考值更正为 Actuator.MaxSpeed/2。 对于模拟驱动器接口，将驱动器和工艺对象组态中的参考值更正为 Actuator.MaxSpeed/1.17。

## 命令表的组态错误

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8700</b>		命令表中“命令类型”(Command type) 的值无效		
	16#000 1	-	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值，并视需要重新启动命令</li> </ul>	
<b>16#8701</b>		命令表中“位置/行进路径”(Position/travel path) 的值无效		
	16#000 2	值的数字格式无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值，并视需要重新启动命令</li> </ul>	
	16#000 5	该值超出值范围（大于 1E+12）		
	16#000 6	该值超出值范围（小于 -1E+12）		
<b>16#8702</b>		命令表中“速度”(Velocity) 的值无效		
	16#000 2	值的数字格式无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值，并视需要重新启动命令</li> </ul>	
	16#000 8	值大于组态的最大速度		
	16#000 9	值小于组态的启动/停止速度		
<b>16#8703</b>		命令表中“持续时间”(Duration) 的值无效		
	16#000 2	值的数字格式无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值，并视需要重新启动命令</li> </ul>	
	16#002 1	值大于 64800 s		
	16#002 2	值小于 0.001 s		

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8704</b>		命令表中“下一步”(Next step) 的值无效		
	16#001 1	所选值无效	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值，并视需要重新启动命令</li> </ul>	
	16#002 3	该命令不允许命令转换		

## 内部错误

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8FFF</b>		内部错误		
	16#F0**	-	<p>CPU 的断电和上电</p> <p>如果这样做无效，则联系客户支持。准备好以下信息：</p> <ul style="list-style-type: none"> <li>ErrorID</li> <li>ErrorInfo</li> <li>诊断缓冲区条目</li> </ul>	

## 参见

[运动控制语句的错误显示 \(页 217\)](#)

[定位轴工艺对象 V4...5 的变量 \(页 485\)](#)

[ErrorID 和 ErrorInfo \(页 414\)](#)

[将多个轴与同一个 PTO 一起使用 \(页 283\)](#)

[将多个驱动器与同一个 PTO 一起使用 \(页 286\)](#)

[在更高优先级等级（执行级别）中跟踪作业 \(页 287\)](#)

[使用通过 PTO 的驱动器连接时的软限位开关时的特殊情况 \(页 290\)](#)

### 12.11.4.2 ErrorID 和 ErrorInfo 列表 (工艺对象 V2...3)

下表列出了运动控制指令中指示的所有 ErrorID 和 ErrorInfo。除错误原因外，还列出了清除错误的解决方法：

#### 伴随轴停止的运行错误

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8000</b>		驱动器错误，丢失“驱动器就绪”信号		
	16#0001	-	使用指令“MC_Reset”确认错误；提供驱动器信号；根据需要重新启动命令	
<b>16#8001</b>		已触发下限软限位开关		
	16#000E	已经以当前组态的减速度到达下限软限位开关的位置	使用“MC_Reset”指令确认错误；使用运动命令使轴沿正向移动，超出软限位开关的范围	
	16#000F	已经以急停减速度到达下限软限位开关的位置		
	16#0010	已经以急停减速度超出下限软限位开关的位置		
<b>16#8002</b>		已触发上限软限位开关		
	16#000E	已经以当前组态的减速度到达上限软限位开关的位置	使用“MC_Reset”指令确认错误；使用运动命令使轴沿负向移动，超出软限位开关的范围	
	16#000F	已经以急停减速度到达上限软限位开关的位置		
	16#0010	已经以急停减速度超出上限软限位开关的位置		
<b>16#8003</b>		已到达下限硬限位开关		
	16#000E	已到达下限硬限位开关。轴以急停减速度停止。 (主动回原点期间，未找到回原点开关)	使用指令“MC_Reset”确认已启用轴的错误；使用运动命令使轴沿正向移动，超出硬限位开关的范围。	

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8004</b>		<b>已到达上限硬限位开关</b>		
	16#000 E	已到达上限硬限位开关。轴以急停减速度停止。 (主动回原点期间，未找到回原点开关)	使用指令“MC_Reset”确认已启用轴的错误；使用运动命令使轴沿负向移动，超出硬限位开关的范围。	
<b>16#8005</b>		<b>PTO/HSC 已被另一个轴占用</b>		
	16#000 1	-	<p><b>轴未正确组态：</b> 更正 PTO (Pulse Train Output)/HSC (High Speed Counter) 的组态并将其下载到控制器</p> <p><b>多个轴将通过一个 PTO 运行：</b> 另一个轴正在使用 PTO / HSC。如果要控制当前轴，则必须使用 “MC_Power”(Enable = FALSE) 禁用另一个轴。 (另请参见将多个轴与同一个 PTO 一起使用 (页 283))</p>	
<b>16#8006</b>		<b>控制面板中发生通信错误</b>		
	16#001 2	已超时	检查电缆连接，然后再次按下“手动控制”(Manual control) 按钮。	
<b>16#8007</b>		<b>该轴无法启用</b>		
	16#002 5	正在重新启动	请等待，直到轴重新启动完成。	
	16#002 6	正在 RUN 模式下执行加载过程	请等待，直到加载过程完成。	

## 不伴随轴停止的运行错误

ErrorID	ErrorInfo	说明	解决方法	
16#8200		轴未启用		
	16#000 1	-	启用轴；重新启动命令	
16#8201		轴已由另一个“MC_Power”实例启用		
	16#000 1	-	仅通过一个“MC_Power”实例启用轴	
16#8202		已超出并行激活的运动控制命令数的最大值（所有运动控制工艺对象最多可具有 200 条命令）		
	16#000 1	-	减少并行激活命令数；重新启动命令 如果运动控制指令的参数“Busy”= TRUE，则说明命令已激活。	
16#8203		轴当前在“手动控制”(Manual control) 模式（轴控制面板）下运行		
	16#000 1	-	退出“手动控制”；重新启动命令	
16#8204		轴未回原点		
	16#000 1	-	使用指令“MC_Home”使轴回原点；重新启动命令	
16#8205		轴当前由用户程序控制（该错误仅显示在轴控制面板中）		
	16#001 3	轴已在用户程序中启用	使用指令“MC_Power”禁用轴或者在轴控制面板中再次选择“手动控制”(Manual control)	
16#8206		工艺对象尚未激活		
	16#000 1	-	使用指令“MC_Power”Enable = TRUE 启用轴或者在轴控制面板中启用轴。	

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8207</b>		<b>命令被拒绝</b>		
	16#001 6	正在执行主动回原点；无法启动另一种回原点方法。	等到主动回原点完成或通过运动命令（例如，“MC_Halt”）中止主动回原点。	
	16#001 8	当轴正在主动或被动回原点时，不能使用命令表进行移动。	一直等到直接地或被动回原点完成。	
	16#001 9	正在处理命令表时，轴不能主动或被动回原点。	等到命令表完成或通过运动命令（例如，“MC_Halt”）中止命令表。	
<b>16#8208</b>		<b>最大速度和启动/停止速度的差值无效</b>		
	16#000 2	值的数字格式无效。	更正该值；重新启动命令	
	16#000 A	值小于或等于 0。		
<b>16#8209</b>		<b>对工艺对象“轴”的加速度无效</b>		
	16#000 2	值的数字格式无效。	更正该值；重新启动命令	
	16#000 A	值小于或等于 0。		
<b>16#820A</b>		<b>无法重新启动轴</b>		
	16#001 3	轴已在用户程序中启用	使用“MC_Power”指令阻止轴；再次重新启动	
	16#002 7	轴当前在“手动控制”模式（轴控制面板）下运行	退出“手动控制”；再次重新启动	
<b>16#820B</b>		<b>无法执行命令表</b>		
	16#002 6	正在 RUN 模式下执行加载过程	请等待，直到加载过程完成。	

## 块参数错误

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8400</b>		运动控制指令的“Position”参数值无效		
	16#000 2	值的数字格式无效。	更正该值；重新启动命令	
	16#000 5	该值超出值范围（大于 1.0E+12）		
	16#000 6	该值超出值范围（小于 -1.0E+12）		
<b>16#8401</b>		运动控制指令的“Distance”参数值无效		
	16#000 2	值的数字格式无效。	更正该值；重新启动命令	
	16#000 5	该值超出值范围（大于 1.0E+12）		
	16#000 6	该值超出值范围（小于 -1.0E+12）		
<b>16#8402</b>		运动控制指令的“Velocity”参数值无效		
	16#000 2	值的数字格式无效。	更正该值；重新启动命令	
	16#000 8	值大于组态的最大速度		
	16#000 9	值小于组态的启动/停止速度		
	16#002 4	值小于 0		
<b>16#8403</b>		运动控制指令的“Direction”参数值无效		
	16#001 1	所选值无效	更正所选值；重新启动命令	

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8404</b>		运动控制指令的“Mode”参数值无效		
	16#001 1	所选值无效	更正所选值；重新启动命令	
	16#001 5	未组态主动/被动回原点	更正组态并将其下载到控制器；启用轴并重新启动命令	
	16#001 7	虽然硬限位开关已禁用，但还是激活了在硬限位开关处反向	<ul style="list-style-type: none"> <li>使用变量 &lt;轴&gt;.Config.PositionLimits_HW.Active=TRUE，激活硬限位开关，重新启动命令</li> <li>更正组态并将其下载到控制器；启用轴并重新启动命令</li> </ul>	
<b>16#8405</b>		运动控制指令的“StopMode”参数值无效		
	16#001 1	所选值无效	更正所选值；再次启用轴	
<b>16#8406</b>		不允许同时正向和反向点动		
	16#000 1	-	采取措施确保参数“JogForward”和“JogBackward”的信号状态不会同时为 TRUE；重新启动命令。	
<b>16#8407</b>		只有禁用激活的轴后，才允许使用指令“MC_Power”切换到另一个轴。		
	16#000 1	-	禁用激活的轴；然后可以切换到其它轴并启用该轴。	
<b>16#8408</b>		运动控制指令的“Axis”参数值无效		
	16#001 A	指定的值与所需的工艺对象版本不匹配	更正该值；重新启动命令	
	16#001 B	指定的值与所需的工艺对象类型不匹配		
	16#001 C	指定的值不是运动控制工艺数据块		

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8409</b>		运动控制指令的“CommandTable”参数值无效		
	16#001 A	指定的值与所需的工艺对象版本不匹配	更正该值；重新启动命令	
	16#001 B	指定的值与所需的工艺对象类型不匹配		
	16#001 C	指定的值不是运动控制工艺数据块		
<b>16#840A</b>		运动控制指令的“StartStep”参数值无效		
	16#000 A	值小于或等于 0。	更正该值；重新启动命令	
	16#001 D	开始步进大于结束步进		
	16#001 E	值大于 32		
<b>16#840B</b>		运动控制指令的“EndStep”参数值无效		
	16#000 A	值小于或等于 0。	更正该值；重新启动命令	
	16#001 E	值大于 32		
<b>16#840C</b>		运动控制指令的“RampUpTime”参数值无效		
	16#000 2	值的数字格式无效。	更正该值；重新启动命令	
	16#000 5	该值超出值范围（大于 1.0E12）		
	16#000 A	值小于或等于 0。		
<b>16#840D</b>		运动控制指令的“RampDownTime”参数值无效		
	16#000 2	值的数字格式无效。	更正该值；重新启动命令	
	16#000 A	值小于或等于 0。		

ErrorID	ErrorInfo	说明	解决方法	
<b>16#840E</b>		运动控制指令的“EmergencyRampTime”参数值无效		
	16#000 2	值的数字格式无效。	更正该值；重新启动命令	
	16#000 A	值小于或等于 0。		
<b>16#840F</b>		运动控制指令的“JerkTime”参数值无效		
	16#000 2	值的数字格式无效。	更正该值；重新启动命令	
	16#000 A	值小于或等于 0。		

## 轴的组态错误

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8600</b>		脉冲发生器 (PTO) 的参数分配无效		
	16#000 B	地址无效	• 将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
	16#001 4	所选硬件由另一应用程序使用		
<b>16#8601</b>		高速计数器 (HSC) 的参数分配无效		
	16#000 B	地址无效	• 将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
	16#001 4	所选硬件由另一应用程序使用		
<b>16#8602</b>		“使能输出”(Enable output) 的参数分配无效		
	16#000 B	地址无效	• 将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	
<b>16#8603</b>		“输入就绪”(Ready input) 的参数分配无效		
	16#000 B	地址无效	• 将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8604</b>		<b>“电机每转的脉冲数”(Pulses per motor revolution) 值无效</b>		
	16#000 A	值小于或等于 0	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> </ul>	
<b>16#8605</b>		<b>“每转的距离”(Distance per revolution) 值无效</b>		
	16#000 2	值的数字格式无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> </ul>	
	16#000 5	该值超出值范围（大于 1.0E+12）		
	16#000 A	值小于或等于 0		
<b>16#8606</b>		<b>“启动/停止速度”(Start / stop velocity) 值无效</b>		
	16#000 2	值的数字格式无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> </ul>	
	16#000 3	值高于硬件上限值		
	16#000 4	值小于硬件下限值		
	16#000 7	启动/停止速度大于最大速度		
<b>16#8607</b>		<b>“最大速度”(maximum velocity) 值无效</b>		
	16#000 2	值的数字格式无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> </ul>	
	16#000 3	值高于硬件上限值		
	16#000 4	值小于硬件下限值		

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8608</b>		<b>“加速度”(Acceleration) 值无效</b>		
	16#000 2	值的数字格式无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
	16#000 3	值高于硬件上限值		
	16#000 4	值小于硬件下限值		
<b>16#8609</b>		<b>“减速度”(Deceleration) 值无效</b>		
	16#000 2	值的数字格式无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
	16#000 3	值高于硬件上限值		
	16#000 4	值小于硬件下限值		
<b>16#860A</b>		<b>“急停减速度”(Emergency stop deceleration) 值无效</b>		
	16#000 2	值的数字格式无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
	16#000 3	值高于硬件上限值		
	16#000 4	值小于硬件下限值		
<b>16#860B</b>		<b>下限软限位开关的位置值无效</b>		
	16#000 2	值的数字格式无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
	16#000 5	该值超出值范围（大于 1.0E+12）		
	16#000 6	该值超出值范围（小于 -1.0E+12）		
	16#000 7	下限软限位开关的位置值大于上限软限位开关的位置值		

ErrorID	ErrorInfo	说明	解决方法
<b>16#860C</b>		上限软限位开关的位置值无效	
	16#000 2	值的数字格式无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
	16#000 5	该值超出值范围（大于 1.0E+12）	
	16#000 6	该值超出值范围（小于 -1.0E+12）	
<b>16#860D</b>		下限硬限位开关的地址无效	
	16#000 C	下降沿的地址无效	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> </ul>
	16#000 D	上升沿的地址无效	
<b>16#860E</b>		上限硬限位开关的地址无效	
	16#000 C	下降沿的地址无效	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> </ul>
	16#000 D	上升沿的地址无效	
<b>16#860F</b>		“起始位置偏移值”(home position offset) 无效	
	16#000 2	值的数字格式无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
	16#000 5	该值超出值范围（大于 1.0E+12）	
	16#000 6	该值超出值范围（小于 -1.0E+12）	
<b>16#8610</b>		“逼近速度”(approach velocity) 值无效	
	16#000 2	值的数字格式无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
	16#000 8	速度大于最大速度	
	16#000 9	速度小于启动/停止速度	

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8611</b>		“回原点速度”(Homing velocity) 值无效		
	16#000 2	值的数字格式无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
	16#000 8	速度大于最大速度		
	16#000 9	速度小于启动/停止速度		
<b>16#8612</b>		回原点开关的地址无效		
	16#000 C	下降沿的地址无效	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> </ul>	
	16#000 D	上升沿的地址无效		
<b>16#8613</b>		虽然未组态硬限位开关，但还是在主动回原点过程中，激活了在该硬限位开关处反向		
	16#000 1	-	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
<b>16#8614</b>		“加加速度”(Jerk) 值无效		
	16#000 2	值的数字格式无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
	16#001 F	值大于最大加加速度值		
	16#002 0	值小于最小加加速度值		
<b>16#8615</b>		“测量单位”的值无效		
	16#001 1	所选值无效	将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴	

## 命令表的组态错误

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8700</b>		命令表中“命令类型”(Command type) 的值无效		
	16#000 1	-	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值，并视需要重新启动命令</li> </ul>	
<b>16#8701</b>		命令表中“位置/行进路径”(Position/travel path) 的值无效		
	16#000 2	值的数字格式无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值，并视需要重新启动命令</li> </ul>	
	16#000 5	该值超出值范围（大于 1.0E+12）		
	16#000 6	该值超出值范围（小于 -1.0E+12）		
<b>16#8702</b>		命令表中“速度”(Velocity) 的值无效		
	16#000 2	值的数字格式无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值，并视需要重新启动命令</li> </ul>	
	16#000 8	值大于组态的最大速度		
	16#000 9	值小于组态的启动/停止速度		
<b>16#8703</b>		命令表中“持续时间”(Duration) 的值无效		
	16#000 2	值的数字格式无效。	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值，并视需要重新启动命令</li> </ul>	
	16#002 1	值大于 64800 s		
	16#002 2	值小于 0.001 s		

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8704</b>		命令表中“下一步”(Next step) 的值无效		
	16#001 1	所选值无效	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值，并视需要重新启动命令</li> </ul>	
	16#002 3	该命令不允许命令转换		

## 内部错误

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8FFF</b>		内部错误		
	16#F0**	-	<p>CPU 的断电和上电</p> <p>如果这样做无效，则联系客户支持。准备好以下信息：</p> <ul style="list-style-type: none"> <li>ErrorID</li> <li>ErrorInfo</li> <li>诊断缓冲区条目</li> </ul>	

## 参见

[ErrorID 和 ErrorInfo 列表（工艺对象 V4...5）\(页 414\)](#)

[将多个轴与同一个 PTO 一起使用 \(页 283\)](#)

### 12.11.4.3 ErrorID 和 ErrorInfo 列表 (工艺对象 V1)

下表列出了运动控制指令中指示的所有 ErrorID 和 ErrorInfo。除错误原因外，还列出了清除错误的解决方法：

#### 伴随轴停止的运行错误

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8000</b>		驱动器错误，丢失“驱动器就绪”信号		
	16#0001	-	使用指令“MC_Reset”确认错误；提供驱动器信号；根据需要重新启动命令	
<b>16#8001</b>		已触发下限软限位开关		
	16#000E	已经以当前组态的减速度到达下限软限位开关的位置	使用“MC_Reset”指令确认错误；使用运动命令使轴沿正向移动，超出软限位开关的范围	
	16#000F	已经以急停减速度到达下限软限位开关的位置		
	16#0010	已经以急停减速度超出下限软限位开关的位置		
<b>16#8002</b>		已触发上限软限位开关		
	16#000E	已经以当前组态的减速度到达上限软限位开关的位置	使用“MC_Reset”指令确认错误；使用运动命令使轴沿负向移动，超出软限位开关的范围	
	16#000F	已经以急停减速度到达上限软限位开关的位置		
	16#0010	已经以急停减速度超出上限软限位开关的位置		
<b>16#8003</b>		已到达下限硬限位开关		
	16#000E	已到达下限硬限位开关。轴以急停减速度停止。 (主动回原点期间，未找到参考点开关)	使用指令“MC_Reset”确认已启用轴的错误；使用运动命令使轴沿正向移动，超出硬限位开关的范围。	

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8004</b>		<b>已到达上限硬限位开关</b>		
	16#000 E	已到达上限硬限位开关。轴以急停减速度停止。 (主动回原点期间, 未找到参考点开关)	使用指令“MC_Reset”确认已启用轴的错误; 使用运动命令使轴沿负向移动, 超出硬限位开关的范围。	
<b>16#8005</b>		<b>PTO/HSC 已被另一个轴占用</b>		
	16#000 1	-	<p><b>轴未正确组态:</b> 更正 PTO (Pulse Train Output)/HSC (High Speed Counter) 的组态并将其下载到控制器</p> <p><b>多个轴将通过一个 PTO 运行:</b> 另一个轴正在使用 PTO/HSC。如果要控制当前轴, 则必须使用 “MC_Power”(Enable = FALSE) 禁用另一个轴。 (另请参见将多个轴与同一个 PTO 一起使用 (页 283))</p>	

## 不伴随轴停止的运行错误

ErrorID	ErrorInfo	说明	解决方法	
16#8200		轴未启用		
	16#000 1	-	启用轴；重新启动命令	
16#8201		轴已由另一个“MC_Power”实例启用		
	16#000 1	-	仅使用一个“MC_Power”指令启用轴	
16#8202		超出并行激活的运动控制制作命令的最大数量（所有运动控制工艺对象最多可执行 200 个命令）		
	16#000 1	-	减少并行激活命令数；重新启动命令 如果运动控制指令的参数“Busy”= TRUE，则说明命令已激活。	
16#8203		轴当前在“手动控制”(Manual control) 模式（轴控制面板）下运行		
	16#000 1	-	退出“手动控制”；重新启动命令	
16#8204		轴未回原点		
	16#000 1	-	使用指令“MC_Home”使轴回原点；重新启动命令	
16#8205		轴当前由用户程序控制（该错误仅显示在轴控制面板中）		
	16#000 1	-	使用指令“MC_Power”禁用轴或者在轴控制面板中再次选择“手动控制”(Manual control)	
16#8206		工艺对象“轴”尚未启用		
	16#000 1	-	使用指令“MC_Power”Enable = TRUE 启用轴或者在轴控制面板中启用轴。	
16#8207		命令被拒绝		
	16#001 6	正在执行主动回原点；无法启动另一种回原点方法。	等到主动回原点完成或通过运动命令（例如，“MC_Halt”）中止主动回原点。然后可以启动其它回原点类型。	

## 块参数错误

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8400</b>		运动控制指令的“Position”参数值无效		
	16#000 2	值的数字格式无效	更正“位置”值；重新启动命令	
	16#000 5	该值超出取值范围（大于 $1e^{12}$ ）		
	16#000 6	该值超出取值范围（小于 $-1e^{12}$ ）		
<b>16#8401</b>		运动控制指令的“Distance”参数值无效		
	16#000 2	值的数字格式无效	更正“Distance”值；重新启动命令	
	16#000 5	该值超出取值范围（大于 $1e^{12}$ ）		
	16#000 6	该值超出取值范围（小于 $-1e^{12}$ ）		
<b>16#8402</b>		运动控制指令的“Velocity”参数值无效		
	16#000 2	值的数字格式无效	更正“Velocity”值；重新启动命令	
	16#000 8	速度大于最大速度		
	16#000 9	速度小于启动/停止速度		
<b>16#8403</b>		运动控制指令的“Direction”参数值无效		
	16#001 1	所选值无效	更正所选值；重新启动命令	

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8404</b>		<b>运动控制指令的“Mode”参数值无效</b>		
	16#001 1	所选值无效	更正所选值；重新启动命令	
	16#001 5	未组态主动/被动回原点	更正组态并将其下载到控制器；启用轴并重新启动命令	
	16#001 7	虽然已禁用硬限位开关，但还是激活了在该硬限位开关处轴反向	<ul style="list-style-type: none"> <li>使用变量 &lt;轴&gt;.Config.PositionLimits_HW.Active=TRUE，激活硬限位开关，重新启动命令</li> <li>更正组态并将其下载到控制器；启用轴并重新启动命令</li> </ul>	
<b>16#8405</b>		<b>运动控制指令的“StopMode”参数值无效</b>		
	16#001 1	所选值无效	更正所选值；再次启用轴	
<b>16#8406</b>		<b>不允许同时正向和反向点动</b>		
	16#000 1	-	采取措施确保参数“JogForward”和“JogBackward”的信号状态不会同时为 TRUE；重新启动命令。	
<b>16#8407</b>		<b>只有禁用轴后，才允许使用运动控制指令“MC_Power”开关轴。</b>		
	16#000 1	-	禁用激活的轴；然后可以切换到其它轴并启用该轴。	

## 组态错误

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8600</b>		脉冲发生器 (PTO) 的参数分配无效		
	16#000 B	地址无效	更正 PTO (Pulse Train Output) 的组态并将其下载到控制器	
<b>16#8601</b>		高速计数器 (HSC) 的参数分配无效		
	16#000 B	地址无效	更正 HSC (High Speed Counter) 的组态并将其下载到控制器	
<b>16#8602</b>		“使能输出”(Enable output) 的参数分配无效		
	16#000 D	地址无效	更正组态并将其下载到控制器	
<b>16#8603</b>		“输入就绪”(Ready input) 的参数分配无效		
	16#000 D	地址无效	更正组态并将其下载到控制器	
<b>16#8604</b>		“电机每转的脉冲数”(Pulses per motor revolution) 值无效		
	16#000 A	值小于或等于 0	更正组态并将其下载到控制器	
<b>16#8605</b>		“每转的距离”(Distance per revolution) 值无效		
	16#000 2	值的数字格式无效	更正组态并将其下载到控制器	
	16#000 A	值小于或等于 0		
<b>16#8606</b>		“启动/停止速度”(Start / stop velocity) 值无效		
	16#000 2	值的数字格式无效	更正组态并将其下载到控制器	
	16#000 3	值超出硬件限值		
	16#000 4	值小于硬件限值		
	16#000 7	启动/停止速度大于最大速度		

ErrorID	ErrorInfo	说明	解决方法
<b>16#8607</b>		<b>“最大速度”(maximum velocity) 值无效</b>	
	16#000 2	值的数字格式无效	更正组态并将其下载到控制器
	16#000 3	值超出硬件限值	
	16#000 4	值小于硬件限值	
<b>16#8608</b>		<b>“加速度”(Acceleration) 值无效</b>	
	16#000 2	值的数字格式无效	<ul style="list-style-type: none"> <li>• 将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>• 在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
	16#000 3	值超出硬件限值	
	16#000 4	值小于硬件限值	
<b>16#8609</b>		<b>“减速度”(Deceleration) 值无效</b>	
	16#000 2	值的数字格式无效	<ul style="list-style-type: none"> <li>• 将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>• 在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
	16#000 3	值超出硬件限值	
	16#000 4	值小于硬件限值	
<b>16#860A</b>		<b>“急停减速度”(Emergency stop deceleration) 值无效</b>	
	16#000 2	值的数字格式无效	<ul style="list-style-type: none"> <li>• 将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>• 在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>
	16#000 3	值超出硬件限值	
	16#000 4	值小于硬件限值	

ErrorID	ErrorInfo	说明	解决方法	
<b>16#860B</b>		下限软限位开关的位置值无效		
	16#0002	值的数字格式无效	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
	16#0005	该值超出取值范围（大于 $1e^{12}$ ）		
	16#0006	该值超出取值范围（小于 $-1e^{12}$ ）		
	16#0007	下限软限位开关的位置值大于上限软限位开关的位置值		
<b>16#860C</b>		上限软限位开关的位置值无效		
	16#0002	值的数字格式无效	<ul style="list-style-type: none"> <li>将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴</li> <li>在线更正不正确的值；使用指令“MC_Reset”确认错误并视需要重新启动命令</li> </ul>	
	16#0005	该值超出取值范围（大于 $1e^{12}$ ）		
	16#0006	该值超出取值范围（小于 $-1e^{12}$ ）		
<b>16#860D</b>		下限硬限位开关的地址无效		
	16#000C	下降沿的地址无效	更正组态并将其下载到控制器	
	16#000D	上升沿的地址无效		
<b>16#860E</b>		上限硬限位开关的地址无效		
	16#000C	下降沿的地址无效	更正组态并将其下载到控制器	
	16#000D	上升沿的地址无效		

ErrorID	ErrorInfo	说明	解决方法	
<b>16#860F</b>		<b>“起始位置偏移值”(home position offset) 无效</b>		
	16#0002	值的数字格式无效	更正组态并将其下载到控制器	
	16#0005	该值超出取值范围（大于 $1e^{12}$ ）		
	16#0006	该值超出取值范围（小于 $-1e^{12}$ ）		
<b>16#8610</b>		<b>“逼近速度”(approach velocity) 值无效</b>		
	16#0002	值的数字格式无效	更正组态并将其下载到控制器	
	16#0008	速度大于最大速度		
	16#0009	速度小于启动/停止速度		
<b>16#8611</b>		<b>“回原点速度”(Homing velocity) 值无效</b>		
	16#0002	值的数字格式无效	更正组态并将其下载到控制器	
	16#0008	速度大于最大速度		
	16#0009	速度小于启动/停止速度		
<b>16#8612</b>		<b>参考点开关的地址无效</b>		
	16#000C	下降沿的地址无效	更正组态并将其下载到控制器	
	16#000D	上升沿的地址无效		
<b>16#8613</b>		虽然未组态硬限位开关，但还是在主动回原点过程中，激活了在该硬限位开关处反向		
	16#0001	-	更正组态并将其下载到控制器	

## 内部错误

ErrorID	ErrorInfo	说明	解决方法	
<b>16#8FFF</b>		内部错误		
	16#F0**	-	CPU 的断电和上电 如果这样做无效，则联系客户支持。准备好以下信息： <ul style="list-style-type: none"><li>• ErrorID</li><li>• ErrorInfo</li><li>• 诊断缓冲区条目</li></ul>	

## 参见

[ErrorID 和 ErrorInfo 列表（工艺对象 V4...5）\(页 414\)](#)

[将多个轴与同一个 PTO 一起使用 \(页 283\)](#)

## 12.11.5 Legend V1...5

变量	变量的名称																		
数据类型	变量的数据类型																		
值	<p>起始值 (变量的值范围 - 最小值到最大值) 如果未显示特定值，则应用相应数据类型的值范围限制或“说明”下面的信息。</p>																		
访问	<p>在用户程序中访问该变量：</p> <table> <tr> <td>R</td><td>在用户程序和 HMI 中，可读取该变量。</td></tr> <tr> <td>RP</td><td>该变量可通过运动控制指令“MC_ReadParam”进行读取。相应变量的当前值将在命令启动时确定。</td></tr> <tr> <td>RW</td><td>在用户程序和 HMI 中，可读写该变量。该变量可通过运动控制指令“MC_WriteParam”进行写入。</td></tr> <tr> <td>RCCP</td><td>该变量可在用户程序和 HMI 中读取，并在每个周期控制点处进行更新。</td></tr> <tr> <td>WP</td><td>与驱动装置连接无关：如果该轴已禁用 (MC_Power.Status = FALSE)，则可使用运动控制指令“MC_WriteParam”对该变量进行写入操作。</td></tr> <tr> <td>WP_PD</td><td>通过 PROFIdrive/模拟量输出进行驱动装置连接：如果该轴已禁用 (MC_Power.Status = FALSE)，则可使用运动控制指令“MC_WriteParam”对该变量进行写入操作。</td></tr> <tr> <td>WP_PT O</td><td>通过 PTO 进行驱动装置连接：如果该轴已禁用 (MC_Power.Status = FALSE)，则可使用运动控制指令“MC_WriteParam”对该变量进行写入操作。</td></tr> <tr> <td>WD BL</td><td>通过 PROFIdrive/模拟量输出进行驱动装置连接：使用扩展指令“WRIT_DB”，可将该变量写入装载存储器中的起始值内。</td></tr> <tr> <td>-</td><td>该变量不能在用户程序中使用。</td></tr> </table>	R	在用户程序和 HMI 中，可读取该变量。	RP	该变量可通过运动控制指令“MC_ReadParam”进行读取。相应变量的当前值将在命令启动时确定。	RW	在用户程序和 HMI 中，可读写该变量。该变量可通过运动控制指令“MC_WriteParam”进行写入。	RCCP	该变量可在用户程序和 HMI 中读取，并在每个周期控制点处进行更新。	WP	与驱动装置连接无关：如果该轴已禁用 (MC_Power.Status = FALSE)，则可使用运动控制指令“MC_WriteParam”对该变量进行写入操作。	WP_PD	通过 PROFIdrive/模拟量输出进行驱动装置连接：如果该轴已禁用 (MC_Power.Status = FALSE)，则可使用运动控制指令“MC_WriteParam”对该变量进行写入操作。	WP_PT O	通过 PTO 进行驱动装置连接：如果该轴已禁用 (MC_Power.Status = FALSE)，则可使用运动控制指令“MC_WriteParam”对该变量进行写入操作。	WD BL	通过 PROFIdrive/模拟量输出进行驱动装置连接：使用扩展指令“WRIT_DB”，可将该变量写入装载存储器中的起始值内。	-	该变量不能在用户程序中使用。
R	在用户程序和 HMI 中，可读取该变量。																		
RP	该变量可通过运动控制指令“MC_ReadParam”进行读取。相应变量的当前值将在命令启动时确定。																		
RW	在用户程序和 HMI 中，可读写该变量。该变量可通过运动控制指令“MC_WriteParam”进行写入。																		
RCCP	该变量可在用户程序和 HMI 中读取，并在每个周期控制点处进行更新。																		
WP	与驱动装置连接无关：如果该轴已禁用 (MC_Power.Status = FALSE)，则可使用运动控制指令“MC_WriteParam”对该变量进行写入操作。																		
WP_PD	通过 PROFIdrive/模拟量输出进行驱动装置连接：如果该轴已禁用 (MC_Power.Status = FALSE)，则可使用运动控制指令“MC_WriteParam”对该变量进行写入操作。																		
WP_PT O	通过 PTO 进行驱动装置连接：如果该轴已禁用 (MC_Power.Status = FALSE)，则可使用运动控制指令“MC_WriteParam”对该变量进行写入操作。																		
WD BL	通过 PROFIdrive/模拟量输出进行驱动装置连接：使用扩展指令“WRIT_DB”，可将该变量写入装载存储器中的起始值内。																		
-	该变量不能在用户程序中使用。																		

W	工艺数据块中更改的有效性	
	1	通过 PTO 进行驱动装置连接：在轴激活、禁用或已启用时
	2	通过 PTO 进行驱动装置连接：轴启用时
	3	启用轴后（之前，轴必须处于停止状态）。可以使用变量 <轴名称>.StatusBits.Standstill 检查轴停止状态。
	4	在轴停止后的下一次运动控制命令启动时。可以使用变量 <轴名称>.StatusBits.Standstill 检查轴停止状态。
	5	通过 PTO 进行驱动装置连接：下一次启动 MC_MoveAbsolute、MC_MoveRelative、MC_MoveVelocity、MC_MoveJog、MC_Halt、MC_CommandTable 或激活的 MC_Home 命令时 (Mode = 3)
	6	通过 PTO 进行驱动装置连接：MC_MoveJog 命令停止时
	7	通过 PTO 进行驱动装置连接：启动被动回原地命令时
	8	通过 PTO 进行驱动装置连接：启动主动回原地命令时
	9	重新启动工艺对象时
	10	通过 PROFIdrive/模拟量输出进行驱动装置连接：下一次调用 MC-Servo [OB91] 时
说明	变量的说明	

通过“<TO>.<变量名称>”的形式访问变量。占位符 <TO> 代表工艺对象的名称。

## 12.11.6 轴工艺对象 V1...3 的变量

### 12.11.6.1 Config 变量 V1...3

变量结构 <轴名称>.Config.<变量名称> 中包含有动态限值组态。

#### 变量

图例 (页 470)

变量	数据类型	值	访问	W	说明
Config.	STRUCT				TO_Struct_Config
General.	STRUCT				TO_Struct_General
PTO	DWORD	DW#16#00000000	-	-	该变量不能在用户程序中评估。
HSC	DWORD	DW#16#00000000	-	-	该变量不能在用户程序中评估。
LengthUnit <sup>2)</sup>	INT	1013 (-32768 到 32767)	R WP	- 2	为参数组态的测量单位 -1 脉冲 1005 ° 1010 m 1013 mm 1018 ft 1019 in
DriveInterface.	STRUCT				TO_Struct_DriveInterface
EnableOutput	-	-	-	-	该变量不能在用户程序中评估。
ReadyInput	-	-	-	-	该变量不能在用户程序中评估。
Mechanics.	STRUCT				TO_Struct_Mechanics
PulsesPerDriveRevolution	DINT	L#1000	R	-	电机每转的脉冲数

变量	数据类型	值	访问	W	说明	
LeadScrew	REAL	10.0	R	-	每转的距离 (用组态的测量单位表示)	
InverseDirection	BOOL	FALSE	R	-	反转方向	
DynamicLimits.	STRUCT				TO_Struct_DynamicLimits	
MinVelocity	REAL	10.0	R	-	轴的启动/停止速度 (用组态的测量单位表示)	
MaxVelocity	REAL	250.0	R	-	轴的最大速度 (用组态的测量单位表示)	
DynamicDefaults.	STRUCT				TO_Struct_DynamicDefaults	
Acceleration	REAL	48.0	RW	1 <sup>2)</sup> 、 5 、 6 <sup>2)</sup>	轴的加速度 (用组态的测量单位表示)	
Deceleration	REAL	48.0	RW	1 <sup>2)</sup> 、 5 、 6	轴的减速度 (用组态的测量单位表示)	
EmergencyDeceleration	REAL	120.0	RW	1 <sup>2)</sup> 、 2 <sup>1)</sup> 、 5 、 6	轴的急停减速度 (用组态的测量单位表示)	
JerkActive <sup>2)</sup>	BOOL	FALSE	RW	1 、 5	FALSE	加急速度限值禁用。
					TRUE	加急速度限值启用。

变量	数据类型	值	访问	W	说明
Jerk <sup>2)</sup>	REAL	192.0	RW	1 、 5	轴斜坡加速和斜坡减速过程中的加速度 (用组态的测量单位表示)
PositionLimits_SW.	STRUCT				TO_Struct_PositionLimits_SW
Active	BOOL	FALSE	RW	1 <sup>2)</sup> 、 4 <sup>1)</sup> 、 5 <sup>2)</sup> 、 6 <sup>2)</sup>	FALSE 软限位开关取消激活。 TRUE 软限位开关激活。
MinPosition	REAL	-1.0E4	RW	1 <sup>2)</sup> 、 4 <sup>1)</sup> 、 5 <sup>2)</sup> 、 6 <sup>2)</sup>	软限位开关下限的位置 (用组态的测量单位表示)
MaxPosition	REAL	1.0E4	RW	1 <sup>2)</sup> 、 4 <sup>1)</sup> 、 5 <sup>2)</sup> 、 6 <sup>2)</sup>	软限位开关的上限位置 (用组态的测量单位表示)

变量	数据类型	值	访问	W	说明	
PositionLimits_HW.					TO_Struct_PositionLimits_HW	
Active	BOOL	FALSE	RW	1 <sup>2)</sup> 、 3 <sup>1)</sup> 、 4 <sup>1)</sup> 、 5 <sup>2)</sup> 、 6 <sup>2)</sup>	FALS E	硬限位开关取消激活。
					TRUE	硬限位开关激活。
MinSwitchedLevel	BOOL	FALSE	R	-	FALS E	CPU 输入端电平为 0 V 时，表示已逼近硬限位开关下限
					TRUE	CPU 输入端电平为 24 V 时，表示已逼近硬限位开关下限
MinFallingEvent	DWORD	DW#16#00000000	-	-	该变量不能在用户程序中评估。	
MinRisingEvent	DWORD	DW#16#00000000	-	-	该变量不能在用户程序中评估。	
MaxSwitchedLevel	BOOL	FALSE	R	-	FALS E	CPU 输入端电平为 0 V 时，表示已逼近硬限位开关上限
					TRUE	CPU 输入端电平为 24 V 时，表示已逼近硬限位开关上限
MaxFallingEvent	DWORD	DW#16#00000000	-	-	该变量不能在用户程序中评估。	
MaxRisingEvent	DWORD	DW#16#00000000	-	-	该变量不能在用户程序中评估。	
Homing.	STRUCT				TO_Struct_Homing	
AutoReversal	BOOL	TRUE	R <sup>1)</sup> RW <sup>2)</sup>	-1) 12) 、 8 <sup>2)</sup>	FALS E	禁用硬限位开关时反向（主动回原点）
					TRUE	启用硬限位开关时反向（主动回原点）

变量	数据类型	值	访问	W	说明	
Direction	BOOL	TRUE	R <sup>1)</sup> RW <sup>2)</sup>	-1) 12) 、 8 <sup>2)</sup>	FALSE	负逼近方向（用于搜索参考点开关）和正回原点方向（主动回原点）
					TRUE	正逼近方向（用于搜索参考点开关）和正回原点方向（主动回原点）
SideActiveHoming <sup>2)</sup>	BOOL	TRUE	RW	1 、 8	FALSE	在参考点开关的下端回原点（主动回原点）
					TRUE	在参考点开关的上端回原点（主动回原点）
SidePassiveHoming <sup>2)</sup>	BOOL	TRUE	RW	1 、 7	FALSE	在参考点开关的下端回原点（被动回原点）
					TRUE	在参考点开关的上端回原点（被动回原点）
RisingEdge <sup>1)</sup>	BOOL	FALSE	R	-	FALSE	通过参考点开关的信号上升沿回原点（主动回原点）
					TRUE	通过参考点开关的信号下降沿回原点（主动回原点）有关该变量对被动回原点的影响，请参见“组态 - 回原点”中的说明。
Offset	REAL	0.0	R RW	-1) 12) 、 8 <sup>2)</sup>	回原点位置偏移 / 单位为组态的测量单位（主动回原点）	
FastVelocity	REAL	200.0	R RW	-1) 12) 、 8 <sup>2)</sup>	逼近速度 / 单位为组态的测量单位（主动回原点）	

变量	数据类型	值	访问	W	说明
SlowVelocity	REAL	40.0	R RW	- 1) , 12) , 8 <sup>2)</sup>	回原点速度 / 单位为组态的测量单位 (主动回原点)
FallingEvent	DWORD	DW#16#00000000	-	-	该变量不能在用户程序中评估。
RisingEvent	DWORD	DW#16#00000000	-	-	该变量不能在用户程序中评估。

1) 仅适用于工艺版本 V1.0

2) 适用于工艺版本 V2.0

## 参见

[定位轴工艺对象 V6...7 的变量 \(页 335\)](#)

[定位轴工艺对象 V4...5 的变量 \(页 485\)](#)

## 12.11.6.2 MotionStatus 变量 V1...3

变量 <轴名称>.MotionStatus.<变量名称> 中包含有运动的状态。

## 变量

图例 (页 470)

变量	数据类型	值	访问	W	说明
MotionStatus.	STRUCT				TO_Struct_MotionStatus
Position	REAL	0.0 (-9.0E15 到 9.0E15)	R	-	轴的实际位置 (用组态的测量单位表示) 如果轴未回原点，则该变量将指示相对于轴启用位置的位置值。
Velocity	REAL	0.0 (-9.0E15 到 9.0E15)	R	-	轴的实际速度 (用组态的测量单位表示)
Distance	REAL	0.0 (-9.0E15 到 9.0E15)	R	-	当前到轴目标位置的距离 (用组态的测量单位表示) 仅当使用“MC_MoveAbsolute”、“MC_MoveRelative”或轴命令表执行定位命令时，该变量值才有效。
TargetPosition	REAL	0.0 (-9.0E15 到 9.0E15)	R	-	轴的目标位置 (用组态的测量单位表示) 仅当使用“MC_MoveAbsolute”、“MC_MoveRelative”或轴命令表执行定位命令时，该变量值才有效。

## 参见

运动状态 (页 225)

定位轴工艺对象 V6...7 的变量 (页 335)

定位轴工艺对象 V4...5 的变量 (页 485)

### 12.11.6.3 StatusBits 变量 V1...3

变量 <轴名称>.StatusBits.<变量名称> 中包含有工艺对象的状态信息。

#### 变量

图例 (页 470)

变量	数据类型	值	访问	W	说明
StatusBits.	STRUCT				TO_Struct_StatusBits
Activated	BOOL	FALSE	R	-	轴的激活状态
					FALSE 轴未激活。
					TRUE 轴已激活。轴将连接所指定的 PTO (Pulse Train Output)。将循环更新工艺数据块中的数据。
Enable	BOOL	FALSE	R	-	轴的启用状态
					FALSE 轴未启用。
					TRUE 轴已启用且已就绪可接受运动控制命令。
HomingDone	BOOL	FALSE	R	-	轴的回原点状态
					FALSE 轴未回原点。
					TRUE 轴已回原点且能够执行绝对定位命令。
					对于相对定位而言，轴不必回原点。在主动回原点过程中，该状态为 FALSE。
					如果轴事先已回原点，则在被动回原点过程中该状态将保持为 TRUE。

变量	数据类型	值	访问	W	说明
Done	BOOL	FALSE	R	-	轴上的命令执行
					FALS E 轴上的运动控制命令已激活。
					TRUE 轴上的运动控制命令未激活。
Error	BOOL	FALSE	R	-	轴上的错误状态
					FALS E 轴上当前无错误。
					TRUE 轴上发生错误。
Standstill	BOOL	FALSE	R	-	有关错误的详细信息，请参见自动模式下运动控制指令的“ErrorID”和“ErrorInfo”参数。 在手动模式下，轴控制面板的“错误消息”(Error message) 框可显示有关错误原因的详细信息。
					FALS E 轴正在运动。
					TRUE 轴处于停止状态。
PositioningCommand	BOOL	FALSE	R	-	执行定位命令
					FALS E 轴上的定位命令未激活。
					TRUE 轴将执行一个定位命令。
SpeedCommand	BOOL	FALSE	R	-	以速度设定值执行命令
					FALS E 轴上以速度设定值执行的命令未激活。
					TRUE 该轴将以预定义的速度执行行进命令。

变量	数据类型	值	访问	W	说明
Homing	BOOL	FALSE	R	-	执行回原点命令
					FALS E 轴上的回原点命令未激活。
					TRUE 轴将执行运动控制指令“MC_Home”或者轴控制面板的回原点命令。
CommandTableActive	BOOL	FALSE	R	-	执行命令表
					FALS E 轴上的命令表未激活。
					TRUE 该轴由运动控制指令“MC_CommandTable”进行控制。
ConstantVelocity	BOOL	FALSE	R	-	恒速
					FALS E 轴处于加速、减速或停止状态。
					TRUE 已达到速度设定值。轴正以恒速运动。
Acceleration	BOOL	FALSE	R	-	加速过程
					FALS E 轴正在减速、以恒速运动或处于停止状态。
					TRUE 轴正在加速。
Deceleration	BOOL	FALSE	R	-	减速过程
					FALS E 轴正在加速、以恒速运动或处于停止状态。
					TRUE 轴正在减速。
ControlPanelActive	BOOL	FALSE	R	-	轴控制面板的激活状态
					FALS E “自动模式”已激活。用户程序对轴具有优先控制权。
					TRUE 在轴控制面板中启用了“手动控制”模式。轴控制面板对轴具有优先控制权。不能通过用户程序来控制轴。

变量	数据类型	值	访问	W	说明
DriveReady	BOOL	FALSE	R	-	驱动装置的运行状态
					FALS E 驱动装置未准备就绪。将不执行设定值。
					TRUE 驱动装置已准备就绪。可以执行设定值。
RestartRequired	BOOL	FALSE	R	-	需要重启轴
					FALS E 不需要重新启动轴。
					TRUE 值已在装载存储器中修改。
					要在 CPU 处于 RUN 模式时将值下载到工作存储器中，必须重新启动轴。使用运动控制指令 MC_Reset 执行此操作。

## 参见

[状态和错误位（自工艺对象 V4 起） \(页 221\)](#)

[定位轴工艺对象 V6...7 的变量 \(页 335\)](#)

[定位轴工艺对象 V4...5 的变量 \(页 485\)](#)

#### 12.11.6.4 ErrorBits 变量 V1...3

变量 <轴名称>.ErrorBits.<变量名称> 指示工艺对象处出错。

#### 变量

图例 (页 470)

变量	数据类型	值	访问	W	说明
ErrorBits.	STRUCT				TO_Struct_ErrorBits
SystemFault	BOOL	FALSE	R	-	内部系统错误
ConfigFault	BOOL	FALSE	R	-	轴组态错误
DriveFault	BOOL	FALSE	R	-	驱动装置中发生错误。“驱动装置准备就绪”(Drive ready) 信号丢失。
SWLimitMinReached	BOOL	FALSE	R	-	已到达下限软限位开关。
SWLimitMinExceeded	BOOL	FALSE	R	-	已超出下限软限位开关。
SWLimitMaxReached	BOOL	FALSE	R	-	已到达上限软限位开关。
SWLimitMaxExceeded	BOOL	FALSE	R	-	已超出上限软限位开关。
HWLimitMin	BOOL	FALSE	R	-	已到达下限硬限位开关。
HWLimitMax	BOOL	FALSE	R	-	已到达上限硬限位开关。
HWUsed	BOOL	FALSE	R	-	另一个轴正在使用相同的 PTO (Pulse Train Output) 并且该轴已通过“MC_Power”启用。

#### 参见

状态和错误位（自工艺对象 V4 起） (页 221)

定位轴工艺对象 V6...7 的变量 (页 335)

定位轴工艺对象 V4...5 的变量 (页 485)

#### 12.11.6.5 内部变量 V1...3

“Internal”变量不包含用户相关数据；无法在用户程序中访问这些变量。

#### 参见

定位轴工艺对象 V6...7 的变量 (页 335)

定位轴工艺对象 V4...5 的变量 (页 485)

#### 12.11.6.6 ControlPanel 变量 V1...3

“ControlPanel”变量不包含用户相关数据；无法在用户程序中访问这些变量。

#### 参见

定位轴工艺对象 V6...7 的变量 (页 335)

定位轴工艺对象 V4...5 的变量 (页 485)

#### 12.11.6.7 工艺对象变量的更新

工艺对象变量中指示的轴的状态和错误信息在各周期控制点处更新。

可编辑组态变量值的变化不立即生效。有关更改生效条件的信息，请参见相关变量的详细说明。

## 12.11.7 定位轴工艺对象 V4...5 的变量

### 12.11.7.1 位置值和速度值变量 V4...5

该变量结构中包含有轴的位置与速度的设定值和实际值。

#### 变量

图例 (页 470)

变量	数据类型	值	访问	W	说明
Position	REAL	0.0 (-9.0E15 到 9.0E15)	RCCP 、RP	-	轴的位置设定值 (用组态的测量单位表示) 如果轴未归位，则该变量指示的是相对于轴启用位置的位置值。
Velocity	REAL	0.0	RCCP 、RP	-	轴的速度设定值 (用组态的测量单位表示)
ActualPosition	REAL	0.0 (-9.0E15 到 9.0E15)	RCCP 、RP	-	轴的实际位置 (用组态的测量单位表示) 如果轴未回原点，则该变量将指示相对于轴启用位置的位置值。
ActualVelocity	REAL	0.0 (-9.0E15 到 9.0E15)	RCCP 、RP	-	轴的实际速度 (用组态的测量单位表示)

#### 参见

[运动状态 \(页 225\)](#)

[定位轴工艺对象 V6...7 的变量 \(页 335\)](#)

[轴工艺对象 V1...3 的变量 \(页 472\)](#)

## 12.11.7.2 执行器变量 V4...5

变量结构 <轴名称>.Actor.<变量名称> 中包含有驱动装置参数。

## 变量

图例 (页 470)

变量	数据类型	值	访问	W	说明
Actor.	STRUCT				TO_Struct_Actor
Type	DINT	1 (0 到 2)	R	-	定位轴工艺对象 V5 及更高版本: 0 = 驱动装置通过一个模拟量输出进行连接。轴的所有运动均在位置控制下进行。 1 通过 PROFIdrive 报文连接驱动装置。轴的所有运动均在位置控制下进行。 2 驱动装置通过一个脉冲接口进行连接。 定位轴工艺对象 V4: 2 驱动装置通过一个脉冲接口进行连接。
InverseDirection	BOOL	FALSE	R WP_P TO	- 2	FALSE 方向未反转。 TRUE 方向已反转。
DirectionMode	INT	0 (0 到 2)	R WP_P TO	- 2	允许的旋转方向 0 双向 1 正方向 2 负方向
Interface.	STRUCT				TO_Struct_ActorInterface
AddressIn <sup>3)</sup>	VREF	-	-	-	PROFIdrive 报文的输入地址 (内部参数)

变量	数据类型	值	访问	W	说明
AddressOut <sup>3)</sup>	VREF	-	-	-	PROFIdrive 报文的输出地址（内部参数）
EnableDriveOutput	VREF	-	-	-	使能输出（内部参数）
DriveReadyInput	VREF	-	-	-	输入就绪（内部参数）
PTO	DWORD	0	-	-	脉冲输出（内部参数）
DriveParameter.	STRUCT				TO_Struct_ActorDriveParameter
ReferenceSpeed <sup>3)</sup>	REAL	3000.0	R	-	<p>驱动装置速度设定值 (N-set) 的参考值 (100%)</p> <p>速度设定值在 PROFIdrive 报文中以 -200% 到 200% “ReferenceSpeed” 范围内的标称值形式进行传送。</p> <p>通过模拟量输出指定设定值时，只要驱动装置允许，模拟量输出就可在从 -117% 到 117% 的范围内运行。</p>
MaxSpeed <sup>3)</sup>	REAL	3000.0	R	-	<p>驱动装置速度设定值 (N-set) 的最大值 (PROFIdrive: MaxSpeed ≤ 2 × ReferenceSpeed)</p> <p>模拟量设定值: MaxSpeed ≤ 1.17 × ReferenceSpeed)</p>
PulsesPerDriveRevolution	DINT	1000 (1 到 21474836 48)	R WP_P TO	2	电机每转的脉冲数

3) 适用于工艺版本 V5.0

## 参见

轴工艺对象 V1...3 的变量 (页 472)

定位轴工艺对象 V6...7 的变量 (页 335)

## 12.11.7.3 Sensor[1] 变量 V4...5

变量结构 <轴名称>.Sensor.<变量名称> 中包含有编码器参数。

## 变量

图例 (页 470)

变量	数据类型	值	访问	W	说明
Sensor.	STRUCT				TO_Struct_Sensor
Sensor[1].	STRUCT				ARRAY[1..1] TO_Struct_Sensor
Type	DINT	0 (0 到 1)	R	-	编码器类型 (内部参数) 0 增量式 1 绝对式
InverseDirection	BOOL	FALSE	R	-	对实际值取反 FALSE 实际值不取反 TRUE 实际值已取反
System	DINT	1 (0 到 1)	R	-	编码器系统 0 线性编码器 1 旋转编码器
MountingMode	DINT	0 (0 到 2)	R	-	编码器的安装方式 0 在电机轴上 2 外部测量系统
Interface.	STRUCT				TO_Struct_SensorInterface
AddressIn.	VREF	-	-	-	PROFIdrive 报文的输入地址 (内部参数)
AREA	BYTE	-	-	-	内部参数
DB_NUMBER	UINT	-	-	-	内部参数
OFFSET	UDINT	-	-	-	内部参数

变量		数据类型	值	访问	W	说明
	RID	DWORD	-	-	-	内部参数
	AddressOut.	VREF	-	-	-	PROFIdrive 报文的输出地址（内部参数）
	AREA	BYTE	-	-	-	内部参数
	DB_NUMBER	UINT	-	-	-	内部参数
	OFFSET	UDINT	-	-	-	内部参数
	RID	DWORD	-	-	-	内部参数
	Parameter.	STRUCT				TO_Struct_SensorParameter
	Resolution	REAL	0.001 (-1.0E12 到 1.0E12)	R	-	线性编码器的精度（两个编码器脉冲之间的偏移值）
	StepsPerRevolution	UDINT	2048 (1 到 8388608)	R	-	旋转编码器的每转增量数
	FineResolutionXist1	UDINT	11 (0 到 31)	R	-	高精度的位数 Gx_XIST1 (周期性实际编码器值)
DeterminableRevolutions	FineResolutionXist2	UDINT	9 (0 到 31)	R	-	高精度的位数 Gx_XIST2 (编码器的绝对值)
	DeterminableRevolutions		1 (0 到 8388608)	R	-	多圈绝对值编码器的差动转数
	0					增量式编码器
1						单回绝对值编码器
DistancePerRevolution		REAL	100.0 (0.0 到 1.0E12)	R	-	外部安装的编码器每转的加载距离

变量	数据类型	值	访问	W	说明
ActiveHoming.	STRUCT				TO_Struct_SensorActiveHoming
Mode	DINT	2 (0 到 2)	R WP_P TO	- 2	主动回原点模式  定位轴工艺对象 V5 及更高版本:  0 基于 PROFIdrive 报文的零位标记 (非 PTO) 1 基于 PROFIdrive 报文和接近开关的零位标记 (非 PTO) 2 通过数字量输入回原点  定位轴工艺对象 V4: 2 通过数字量输入回原点
SideInput	BOOL	FALSE	RW	1 、 8 、 10	主动回原点过程中, 轴回原点时所到达的回原点开关端  FALSE 下侧 TRUE 上侧
DigitalInputAddress.	VREF	-	-	-	回原点开关的符号化输入地址 (内部参数)
AREA	BYTE	-	-	-	内部参数
DB_NUMBER	UINT	-	-	-	内部参数
OFFSET	UDINT	-	-	-	内部参数
RID	DWORD	-	-	-	内部参数
HomePositionOffset	REAL	0.0 (-1.0E12 到 1.0E12)	RW	1 、 8 、 10	原地位置偏移值 (主动回原点) (用组态的测量单位表示)

变量	数据类型	值	访问	W	说明
SwitchLevel	BOOL	TRUE	RW	1 、 8 、 10	选择到达回原点开关时 CPU 输入端的信号电平
					FALS E 低电平（低电平有效）
					TRUE 高电平（高启用）
PassiveHoming.	STRUCT				TO_Struct_SensorPassiveHoming
Mode	DINT	2 (0 到 2)	R WP_P TO	- 2	被动回原点模式
					定位轴工艺对象 V5 及更高版本：
					0 基于 PROFIdrive 报文的零位标记（非 PTO）
					1 基于 PROFIdrive 报文和接近开关的零位标记（非 PTO）
					2 通过数字量输入回原点
					定位轴工艺对象 V4：
					2 通过数字量输入回原点
SideInput	BOOL	FALSE	RW	1 、 7 、 10	被动回原点过程中，轴回原点时所到达的回原点开关端
					FALS E 下侧
					TRUE 上侧
DigitalInputAddress.	VREF	-	-	-	回原点开关的符号化输入地址（内部参数）
AREA	BYTE	-	-	-	内部参数
DB_NUMBER	UINT	-	-	-	内部参数
OFFSET	UDINT	-	-	-	内部参数
RID	DWORD	-	-	-	内部参数

变量	数据类型	值	访问	W	说明
SwitchLevel	BOOL	TRUE	RW	1	选择到达回原点开关时 CPU 输入端的电平
				7	FALS 低电平（低电平有效）
				10	E TRUE 高电平（高启用）

## 参见

[定位轴工艺对象 V6...7 的变量 \(页 335\)](#)

[轴工艺对象 V1...3 的变量 \(页 472\)](#)

### 12.11.7.4 单位变量 V4...5

变量结构 <轴名称>.Units.LengthUnit 中包含有为参数组态的测量单位。

#### 变量

图例 (页 470)

变量	数据类型	值	访问	W	说明
Units.	STRUCT				TO_Struct_Units
LengthUnit	INT	1013 (-32768 到 32767)	R WP_P TO	-2	为参数组态的测量单位
				-1	脉冲
				1005	°
				1010	m
				1013	mm
				1018	ft
				1019	in

#### 参见

定位轴工艺对象 V6...7 的变量 (页 335)

轴工艺对象 V1...3 的变量 (页 472)

### 12.11.7.5 机械变量 V4...5

变量结构 <轴名称>.Mechanics.LeadScrew 中包含有电机每转的行进距离。

#### 变量

图例 (页 470)

变量	数据类型	值	访问	W	说明
Mechanics.	STRUCT				TO_Struct_Mechanics
LeadScrew	REAL	10.0 (-1.0E12 到 1.0E12)	R、 WP_P TO	-	每转的距离 (用组态的测量单位表示)

#### 参见

定位轴工艺对象 V6...7 的变量 (页 335)

轴工艺对象 V1...3 的变量 (页 472)

### 12.11.7.6 Modulo 变量 V4...5

变量结构 <轴名称>.Modulo.<变量名称> 中包含有模数设置。

#### 变量

图例 (页 470)

变量	数据类型	值	访问	W	说明	
Modulo.	STRUCT				TO_Struct_Modulo	
Enable	BOOL	FALSE	R	-	FALSE	模数转换已禁用
					TRUE	模数转换已启用 启用模数转换时，应对模数长度 > 0.0 进行检查
Length	REAL	360.0 (0.001 到 1.0E12)	R	-	模数长度	
StartValue	REAL	0.0 (-1.0E12 到 1.0E12)	R	-	模数起始值	

#### 参见

定位轴工艺对象 V6...7 的变量 (页 335)

轴工艺对象 V1...3 的变量 (页 472)

## 12.11.7.7 DynamicLimits 变量 V4...5

变量结构 <轴名称>.DynamicLimits.<变量名称> 中包含有动态限值组态。

## 变量

图例 (页 470)

变量	数据类型	值	访问	W	说明
DynamicLimits.	STRUCT				TO_Struct_DynamicLimits
MaxVelocity	REAL	250.0	R WP_P TO	- 2	轴的最大速度 (用组态的测量单位表示)
MinVelocity	REAL	10.0	R WP_P TO	- 2	轴的启动/停止速度 (用组态的测量单位表示)

## 参见

定位轴工艺对象 V6...7 的变量 (页 335)

轴工艺对象 V1...3 的变量 (页 472)

### 12.11.7.8 DynamicDefaults 变量 V4...5

变量结构 <轴名称>.DynamicDefaults.<变量名称> 中包含有动态默认值组态。

#### 变量

图例 (页 470)

变量	数据类型	值	访问	W	说明
DynamicDefaults.	STRUCT				TO_Struct_DynamicDefaults
Acceleration	REAL	48.0 (0.0 到 1.0E12)	RW	1 ' 5 ' 6 ' 10	轴加速度的默认设置 (用组态的测量单位表示)
Deceleration	REAL	48.0 (0.0 到 1.0E12)	RW	1 ' 5 ' 6 ' 10	轴的默认减速度 (用组态的测量单位表示)
Jerk	REAL	192.0 (0.0 到 1.0E12)	RW	1 ' 5 ' 10	轴斜坡加速和减速过程中的加加速度 默认设置 (用组态的测量单位表示) 所组态的加加速度大于 0.00004 mm/s <sup>2</sup> 时，激活加加速度。
EmergencyDeceleration	REAL	120.0 (0.0 到 1.0E12)	RW	1 ' 5 ' 6 ' 10	轴的急停减速度 (用组态的测量单位表示)

## 参见

[定位轴工艺对象 V6...7 的变量 \(页 335\)](#)

[轴工艺对象 V1...3 的变量 \(页 472\)](#)

### 12.11.7.9 PositionLimitsSW 变量 V4...5

变量结构 <轴名称>.PositionLimits\_SW.<变量名称> 中包含有软限位开关的定位监视组态。软限位开关用于限制定位轴的操作范围。

## 变量

[图例 \(页 470\)](#)

变量	数据类型	值	访问	W	说明	
PositionLimits_SW.	STRUCT				TO_Struct_PositionLimitsSW	
Active	BOOL	FALSE	RW	1 、 5 、 6 、 10	FALSE	软限位开关取消激活。
					TRUE	软限位开关激活。
MinPosition	REAL	-10000.0 (-1.0E12 到 1.0E12)	RW	1 、 5 、 6 、 10	软限位开关下限的位置 (用组态的测量单位表示)	

变量	数据类型	值	访问	W	说明
MaxPosition	REAL	10000.0 (-1.0E12 到 1.0E12)	RW	1 、 5 、 6 、 10	软限位开关的上限位置 (用组态的测量单位表示)

## 参见

[定位轴工艺对象 V6...7 的变量 \(页 335\)](#)

[轴工艺对象 V1...3 的变量 \(页 472\)](#)

### 12.11.7.10 PositionLimitsHW 变量 V4...5

变量结构 <轴名称>.PositionLimits\_HW.<变量名称> 中包含有硬限位开关的定位监视组态。硬限位开关用于限制定位轴的行进范围。

#### 变量

图例 (页 470)

变量	数据类型	值	访问	W	说明	
PositionLimits_HW.	STRUCT				TO_Struct_PositionLimitsHW	
Active	BOOL	FALSE	RW	1 、 5 、 6 、 10	FALSE	硬限位开关取消激活。
					TRUE	硬限位开关激活。
MinSwitchLevel	BOOL	FALSE	R WP_P TO	- 2	选择到达硬限位开关下限时 CPU 输入端的信号电平	
					FALSE	低电平（低电平有效）
					TRUE	高电平（高启用）
MinSwitchAddress	VREF	-	-	-	硬限位开关下限的符号化输入地址（内部参数）	
MaxSwitchLevel	BOOL	FALSE	R WP_P TO	- 2	选择到达硬限位开关上限时 CPU 输入端的信号电平	
					FALSE	低电平（低电平有效）
					TRUE	高电平（高启用）
MaxSwitchAddress	VREF	-	-	-	硬限位开关上限的输入地址（内部参数）	

## 参见

定位轴工艺对象 V6...7 的变量 (页 335)

轴工艺对象 V1...3 的变量 (页 472)

### 12.11.7.11 Homing 变量 V4...5

变量结构 <轴名称>.Homing.<变量名称> 中包含有轴回原点的组态。

## 变量

图例 (页 470)

变量	数据类型	值	访问	W	说明	
Homing.	STRUCT				TO_Struct_Homing	
AutoReversal	BOOL	FALSE	RW	1 、 8 、 10	FALSE	取消激活硬限位开关处的自动反向。
					TRUE	激活硬限位开关处的自动反向。
ApproachDirection	BOOL	TRUE	RW	1 、 8 、 10	FALSE	负逼近方向（用于查找回原点开关）和负回原点方向
					TRUE	正逼近方向（用于搜索参考点开关）和正回原点方向
ApproachVelocity	REAL	200.0 (0.0 到 1.0E12)	RW	1 、 8 、 10	主动归位期间轴的逼近速度 (用组态的测量单位表示)	
ReferencingVelocity	REAL	40.0 (0.0 到 1.0E12)	RW	1 、 8 、 10	主动归位期间轴的归位速度 (用组态的测量单位表示)	

## 参见

[定位轴工艺对象 V6...7 的变量 \(页 335\)](#)

[轴工艺对象 V1...3 的变量 \(页 472\)](#)

**12.11.7.12 PositionControl 变量 V5**

变量结构 <轴名称>.PositionControl.<变量名称> 中包含有位置控制设置。

## 变量

[图例 \(页 470\)](#)

变量	数据类型	值	访问	W	说明
PositionControl.	STRUCT				TO_Struct_PositionControl
Kv	REAL	10.0 (0.0 到 2147480.0 )	R WP	- 10	闭环位置控制的比例增益 ("Kv">> 0.0)
Kpc	REAL	100.0 (0.0 到 150.0)	R WP	- 10	位置控制的百分比速度预控制

## 参见

[定位轴工艺对象 V6...7 的变量 \(页 335\)](#)

[轴工艺对象 V1...3 的变量 \(页 472\)](#)

### 12.11.7.13 FollowingError 变量 V5

变量结构 <轴名称>.FollowingError.<变量名称> 中包含有动态跟随误差监视的组态。

#### 变量

图例 (页 470)

变量	数据类型	值	访问	W	说明	
FollowingError.	STRUCT				TO_Struct_FollowingError	
EnableMonitoring	BOOL	TRUE	R	-	FALSE	跟随误差监视已取消激活
					TRUE	跟随误差监视已启用
MinValue	REAL (0.0 到 1.0E12)	10.0 (0.0 到 1.0E12)	R WP_P D	- 10	速度低于“MinVelocity”的值时允许跟随误差。	
MaxValue	REAL (0.0 到 1.0E12)	100.0 (0.0 到 1.0E12)	R WP_P D	- 10	速度达到最大速度时允许的最大跟随误差。	
MinVelocity	REAL (0.0 到 1.0E12)	10.0 (0.0 到 1.0E12)	R WP_P D	- 10	“MinValue”允许低于该速度，并保持恒定。	

#### 参见

定位轴工艺对象 V6...7 的变量 (页 335)

轴工艺对象 V1...3 的变量 (页 472)

### 12.11.7.14 PositionMonitoring 变量 V5

变量结构 <轴名称>.PositionMonitoring.<变量名称> 中包含有定位运动结束时的定位监控组态。

#### 变量

图例 (页 470)

变量	数据类型	值	访问	W	说明
PositionMonitoring.	STRUCT				TO_Struct_PositionMonitoring
ToleranceTime	REAL	1.0 (0.0 到 1.0E12)	R WP_P D	- 10	容差时间 从达到速度设定值 0 开始直至进入到定位窗口中的最大允许持续时间
MinDwellTime	REAL	0.1 (0.0 到 1.0E12)	R WP_P D	- 10	在定位窗口停留的最短时间
Window	REAL	1.0 (0.001 到 1.0E12)	R WP_P D	- 10	定位窗口

#### 参见

定位轴工艺对象 V6...7 的变量 (页 335)

轴工艺对象 V1...3 的变量 (页 472)

### 12.11.7.15 StandstillSignal 变量 V5

变量结构 <轴名称>.StandstillSignal.<变量名称> 中包含有停止信号的组态。

#### 变量

图例 (页 470)

变量	数据类型	值	访问	W	说明
StandstillSignal.	STRUCT				TO_Struct_StandstillSignal
VelocityThreshold	REAL (0.0 到 1.0E12)	5.0 (0.0 到 1.0E12)	R WP_P D	- 10	速度阈值 如果速度低于此阈值，则最短停留时间开始。
MinDwellTime	REAL (0.0 到 1.0E12)	0.01 (0.0 到 1.0E12)	R WP_P D	- 10	最短停留时间

#### 参见

定位轴工艺对象 V6...7 的变量 (页 335)

轴工艺对象 V1...3 的变量 (页 472)

### 12.11.7.16 StatusPositioning 变量 V4...5

变量结构 <轴名称>.StatusPositioning.<变量名称> 指示定位运动的状态。

#### 变量

图例 (页 470)

变量	数据类型	值	访问	W	说明
StatusPositioning.	STRUCT				TO_Struct_StatusPositioning
Distance	REAL	0.0 (-9.0E15 到 9.0E15)	RCCP 、 RP	-	轴距目标位置的当前距离 (用组态的测量单位表示) 该变量值仅在使用 “MC_MoveAbsolute”、 “MC_MoveRelative”或轴控制面板执行 定位命令期间有效。
TargetPosition	REAL	0.0 (-9.0E15 到 9.0E15)	RCCP 、 RP	-	轴的目标位置 (用组态的测量单位表示) 该变量值仅在使用 “MC_MoveAbsolute”、 “MC_MoveRelative”或轴控制面板执行 定位命令期间有效。
FollowingError	REAL	0.0 (-9.0E15 到 9.0E15)	RCCP 、 RP	-	轴的当前跟随误差 (用组态的测量单位表示) 对于通过 PTO 进行驱动装置连接， FollowingError = 0.0 (Pulse Train Output)。

#### 参见

运动状态 (页 225)

定位轴工艺对象 V6...7 的变量 (页 335)

轴工艺对象 V1...3 的变量 (页 472)

### 12.11.7.17 StatusDrive 变量 V5

变量结构 <轴名称>.StatusDrive.<变量名称> 指示驱动装置的状态。

#### 变量

图例 (页 470)

变量	数据类型	值	访问	W	说明
StatusDrive.	STRUCT				TO_Struct_StatusDrive
InOperation	BOOL	FALSE	RCCP 、 RP	-	驱动器的运行状态
				FALSE	驱动装置未就绪。将不执行设定值。
				TRUE	驱动装置就绪。可以执行设定值。
CommunicationOK	BOOL	FALSE	RCCP 、 RP	-	控制器与驱动器之间的周期性总线通信
				FALSE	未建立通信
				TRUE	通信已建立

#### 参见

定位轴工艺对象 V6...7 的变量 (页 335)

轴工艺对象 V1...3 的变量 (页 472)

### 12.11.7.18 StatusSensor 变量 V5

变量结构 <轴名称>.StatusSensor.<变量名称> 指示测量系统的状态。

#### 变量

图例 (页 470)

变量	数据类型	值	访问	W	说明
StatusSensor.	STRUCT				TO_Struct_StatusSensor
State	DINT	0 (0 到 2)	RCCP 、 RP	-	编码器值的状态
					0 无效
					1 等待有效状态
					2 有效
					控制器与编码器之间的周期性总线通信
CommunicationOK	BOOL	FALSE	RCCP 、 RP	-	FALSE 未建立通信
					TRUE 通信已建立
AbsEncoderOffset	REAL	0.0 (-9.0E15 到 9.0E15)	RCCP 、 RP	-	到绝对值编码器的值的起始点偏移。 此值将永久性存储在 CPU 中。

#### 参见

定位轴工艺对象 V6...7 的变量 (页 335)

轴工艺对象 V1...3 的变量 (页 472)

### 12.11.7.19 StatusBits 变量 V4...5

变量 <轴名称>.StatusBits.<变量名称> 中包含有工艺对象的状态信息。

#### 变量

图例 (页 470)

变量	数据类型	值	访问	W	说明	
StatusBits.	STRUCT				TO_Struct_StatusBits	
Activated	BOOL	FALSE	RCCP 、 RP	-	轴的激活状态	
					FALSE	轴未激活。
					TRUE	轴已激活。轴将连接所指定的 PTO (Pulse Train Output)。将循环更新工艺数据块中的数据。
Enable	BOOL	FALSE	RCCP 、 RP	-	轴的启用状态	
					FALSE	轴未启用。
					TRUE	轴已启用且已就绪可接受运动控制命令。
AxisSimulation	BOOL	FALSE	RCCP 、 RP	-	FALSE	仿真已禁用。
					TRUE	仿真已启用。
NonPositionControlled	BOOL	FALSE	RCCP 、 RP	-	FALSE	轴处于位置控制的模式。
					TRUE	轴未处于位置控制的模式。

变量	数据类型	值	访问	W	说明				
HomingDone	BOOL	FALSE	RCCP 、RP	-	<p>轴的回原点状态</p> <table border="1"> <tr> <td>FALS E</td><td>轴未回原点。</td></tr> <tr> <td>TRUE</td><td>轴已回原点且能够执行绝对定位命令。</td></tr> </table> <p>对于相对定位而言，轴不必回原点。 在主动回原点过程中，该状态为 FALSE。 如果轴事先已经回原点，则在被动回原点期间该状态会保持为 TRUE。</p>	FALS E	轴未回原点。	TRUE	轴已回原点且能够执行绝对定位命令。
FALS E	轴未回原点。								
TRUE	轴已回原点且能够执行绝对定位命令。								
Done	BOOL	FALSE	RCCP 、RP	-	<p>轴上的命令执行</p> <table border="1"> <tr> <td>FALS E</td><td>轴上的运动控制命令已激活。</td></tr> <tr> <td>TRUE</td><td>轴上的运动控制命令未激活。</td></tr> </table>	FALS E	轴上的运动控制命令已激活。	TRUE	轴上的运动控制命令未激活。
FALS E	轴上的运动控制命令已激活。								
TRUE	轴上的运动控制命令未激活。								
Error	BOOL	FALSE	RCCP 、RP	-	<p>轴上的错误状态</p> <table border="1"> <tr> <td>FALS E</td><td>轴上当前无错误。</td></tr> <tr> <td>TRUE</td><td>轴上发生错误。</td></tr> </table> <p>有关错误的详细信息，请参见自动模式下运动控制指令的“ErrorID”和“ErrorInfo”参数。 在手动模式下，轴控制面板的“错误消息”(Error message) 框可显示有关错误原因的详细信息。</p>	FALS E	轴上当前无错误。	TRUE	轴上发生错误。
FALS E	轴上当前无错误。								
TRUE	轴上发生错误。								
Standstill	BOOL	FALSE	RCCP 、RP	-	<p>轴的停止状态</p> <table border="1"> <tr> <td>FALS E</td><td>轴正在运动。</td></tr> <tr> <td>TRUE</td><td>轴处于停止状态。</td></tr> </table>	FALS E	轴正在运动。	TRUE	轴处于停止状态。
FALS E	轴正在运动。								
TRUE	轴处于停止状态。								

变量	数据类型	值	访问	W	说明
PositioningCommand	BOOL	FALSE	RCCP 、 RP	-	执行定位命令
					FALS E 轴上的定位命令未激活。
					TRUE 轴正在执行“MC_MoveRelative”或“MC_MoveAbsolute”运动控制指令的定位命令。
VelocityCommand	BOOL	FALSE	RCCP 、 RP	-	以速度设定值执行命令
					FALS E 轴上以速度设定值执行的命令未激活。
					TRUE 轴正在以“MC_MoveVelocity”或“MC_MoveJog”运动控制指令的速度设定值执行运动命令。
HomingCommand	BOOL	FALSE	RCCP 、 RP	-	执行回原点命令
					FALS E 轴上的回原点命令未激活。
					TRUE 轴正在执行“MC_Home”运动控制指令的回原点命令。
CommandTableActive	BOOL	FALSE	RCCP 、 RP	-	执行命令表
					FALS E 轴上的命令表未激活。
					TRUE 该轴由运动控制指令“MC_CommandTable”进行控制。
ConstantVelocity	BOOL	FALSE	RCCP 、 RP	-	恒速
					FALS E 轴处于加速、减速或停止状态。
					TRUE 已达到速度设定值。轴正以恒速运动。

变量	数据类型	值	访问	W	说明
Accelerating	BOOL	FALSE	RCCP 、 RP	-	加速过程
					FALS E 轴正在减速、以恒速运动或处于停止状态。
					TRUE 轴正在加速。
Decelerating	BOOL	FALSE	RCCP 、 RP	-	减速过程
					FALS E 轴正在加速、以恒速运动或处于停止状态。
					TRUE 轴正在减速。
ControlPanelActive	BOOL	FALSE	RCCP 、 RP	-	轴控制面板的激活状态
					FALS E “自动模式”已激活。用户程序对轴具有优先控制权。
					TRUE 在轴控制面板中启用了“手动控制”(Manual control) 模式。轴控制面板对轴具有优先控制权。不能通过用户程序来控制轴。
DriveReady	BOOL	FALSE	RCCP 、 RP	-	驱动器的运行状态
					FALS E 驱动装置未准备就绪。将不执行设定值。
					TRUE 驱动装置已准备就绪。可以执行设定值。
RestartRequired	BOOL	FALSE	RCCP 、 RP	-	需要重启轴
					FALS E 不需要重新启动轴。
					TRUE 值已在装载存储器中修改。  要在 CPU 处于 RUN 模式时将值下载到工作存储器中，必须重新启动轴。 使用运动控制指令 MC_Reset 执行此操作。

变量	数据类型	值	访问	W	说明
SWLimitMinActive	BOOL	FALSE	RCCP 、 RP	-	软限位开关下限的状态
					FALSE 轴保持在所组态的工作区域内。
					TRUE 已到达或超出软限位开关的下限。
SWLimitMaxActive	BOOL	FALSE	RCCP 、 RP	-	软限位开关上限的状态
					FALSE 轴保持在所组态的工作区域内。
					TRUE 已到达或超出软限位开关的上限。
HWLimitMinActive	BOOL	FALSE	RCCP 、 RP	-	硬限位开关下限的状态
					FALSE 轴保持在所组态的允许行进范围内。
					TRUE 已到达或超出硬限位开关的下限。
HWLimitMaxActive	BOOL	FALSE	RCCP 、 RP	-	硬限位开关上限的状态
					FALSE 轴保持在所组态的允许行进范围内。
					TRUE 已到达或超出硬限位开关的上限。

## 参见

状态和错误位（自工艺对象 V4 起）(页 221)

定位轴工艺对象 V6...7 的变量(页 335)

轴工艺对象 V1...3 的变量(页 472)

## 12.11.7.20 ErrorBits 变量 V4...5

变量 <轴名称>.ErrorBits.<变量名称> 指示工艺对象处出错。

## 变量

图例 (页 470)

变量	数据类型	值	访问	W	说明
ErrorBits.	STRUCT				TO_Struct_ErrorBits
SystemFault	BOOL	FALSE	RCCP 、RP	-	内部系统错误
ConfigFault	BOOL	FALSE	RCCP 、RP	-	轴组态错误
DriveFault	BOOL	FALSE	RCCP 、RP	-	驱动器中发生错误。“驱动器准备就绪”(Drive ready) 信号丢失。
SWLimit	BOOL	FALSE	RCCP 、RP	-	到达或超出软限位开关
HWLimit	BOOL	FALSE	RCCP 、RP	-	到达或超出硬限位开关
DirectionFault	BOOL	FALSE	RCCP 、RP	-	不允许的运动方向
HWUsed	BOOL	FALSE	RCCP 、RP	-	另一个轴正在使用相同的 PTO (Pulse Train Output) 并且该轴已通过“MC_Power”启用。
SensorFault	BOOL	FALSE	RCCP 、RP	-	编码器系统中发生错误
CommunicationFault	BOOL	FALSE	RCCP 、RP	-	通信错误 与所连接设备进行通信时发生错误。
FollowingErrorFault	BOOL	FALSE	RCCP 、RP	-	超出了最大的允许跟随误差
PositioningFault	BOOL	FALSE	RCCP 、RP	-	定位误差 在定位运动的末端，轴定位错误。

## 参见

状态和错误位（自工艺对象 V4 起） (页 221)

定位轴工艺对象 V6...7 的变量 (页 335)

轴工艺对象 V1...3 的变量 (页 472)

### 12.11.7.21 ControlPanel 变量 V4...5

“ControlPanel”变量中不含任何与用户相关的数据。不能在用户程序中访问这些变量。

## 参见

定位轴工艺对象 V6...7 的变量 (页 335)

轴工艺对象 V1...3 的变量 (页 472)

### 12.11.7.22 内部变量 V4...5

“Internal”变量中不含任何与用户相关的数据。不能在用户程序中访问这些变量。

## 参见

定位轴工艺对象 V6...7 的变量 (页 335)

轴工艺对象 V1...3 的变量 (页 472)

### 12.11.7.23 工艺对象变量的更新

工艺对象变量中指示的轴的状态和错误信息在各周期控制点处更新。

组态变量值的更改不会立即生效。有关更改生效条件的信息，请参见相关变量的详细说明。

### 12.11.8 命令表工艺对象 V1...3 的变量

变量结构 <命令表>.Config.Command.Command[x]<变量名称> 中包含有所组态的命令参数。

#### 变量

图例 (页 470)

变量	数据类型	值	访问	W	说明
Config.	STRUCT				TO_Struct_Config
Command.	STRUCT				TO_Struct_Command
Command[x].	STRUCT				ARRAY[1..32] TO_Struct_Command[x]
Type	INT	0 (0 到 151)	RW	-	命令类型 0 “清空”命令 2 “停止”命令 5 “相对定位”命令 6 “绝对定位”命令 7 “Velocity setpoint”命令 151 “Wait”命令
Position	REAL	0.0	RW	-	命令的目标位置/行进距离
Velocity	REAL	0.0	RW	-	速度命令
Duration	REAL	0.0	RW	-	持续时间命令
BufferMode	INT	0 (0 到 1)	RW	-	“下一步”命令的值 0 “完成命令” 1 “混合运动”
StepCode	WORD	0	RW	-	命令步代码

## 参见

命令表工艺对象变量 V4...5 (页 517)

工艺对象命令表 V6 中的变量 (页 385)

### 12.11.9 命令表工艺对象变量 V4...5

变量结构 <命令表>.Command[n]<变量名称> 中包含有所组态的命令参数。

## 变量

图例 (页 470)

变量	数据类型	值	访问	W	说明
Command[n].	STRUCT				ARRAY[1..32] TO_Struct_Command[n]
Type	INT	0 (0 到 151)	RW	-	命令类型
					0 “清空”命令
					2 “停止”命令
					5 “相对定位”命令
					6 “绝对定位”命令
					7 “Velocity setpoint”命令
					151 “Wait”命令
Position	REAL	0.0	RW	-	命令的目标位置/行进距离
Velocity	REAL	0.0	RW	-	速度命令
Duration	REAL	0.0	RW	-	持续时间命令
NextStep	INT	0 (0 到 1)	RW	-	用于切换到下一条命令的模式
					0 “完成命令”
					1 “混合运动”
StepCode	WORD	0	RW	-	命令步代码

## 参见

命令表工艺对象 V1...3 的变量 (页 516)

工艺对象命令表 V6 中的变量 (页 385)

# 索引

- MC\_MoveVelocity  
    功能图, 263  
    指令, 259
- MC\_Power  
    功能图, 233  
    参数, 229  
    指令, 228
- MC\_ReadParam  
    参数, 278  
    指令, 277
- MC\_Reset, 235
- MC\_WriteParam  
    参数, 281  
    指令, 280
- MoveVelocity  
    参数, 260
- O
- OB 伺服, 42
- P
- PROFIdrive, 26
- S
- S7-1200 Motion Control 的报文, 26  
S7-1200 Motion Control 的驱动装置接口, 26  
S7-1200 Motion Control 的编码器接口, 26
- T
- TO\_PositioningAxis, 79
- D
- DB\_ANY, 219
- M
- MC 伺服 OB, 39, 42  
MC 插补器 OB, 39, 42  
MC\_ChangeDynamic  
    参数, 275  
    指令, 274  
MC\_CommandTable  
    参数, 272  
    指令, 270  
MC\_Halt  
    功能图, 247  
    参数, 246  
    指令, 245  
MC\_Home  
    参数, 240  
    指令, 238  
MC\_MoveAbsolute  
    功能图, 252  
    参数, 250  
    指令, 249  
MC\_MoveJog  
    功能图, 268  
    参数, 266  
    指令, 265  
MC\_MoveRelative  
    功能图, 257  
    参数, 255  
    指令, 254

**G**

工艺对象  
    数据类型, 219  
工艺对象轴  
    ErrorBits 变量, 483  
    MotionStatus 变量, 478  
    StatusBits 变量, 479  
    变量 config., 472

**S H**

手册  
    用途, 3  
    适用范围, 3

**F**

方向输出和行进方向  
    关系, 21

**Y**

用途  
    手册, 3

**J**

加加速度限值  
    功能, 46

**H**

回原点  
    回原点模式, 47

**Y**

优化, 192  
优化位置控制器, 192

**B**

闭环控制, 130  
**Y**  
运动控制 CPU S7-1200  
    准则, 50  
运动控制 S7-1200  
    PROFIdrive, 26  
    报文, 26  
驱动器和编码器连接, 26  
运动控制 S7-1200 的硬件配置, 13

**B**

步进电机, 13

**W**

位置控制, 130

**C**

伺服 OB, 39, 42  
伺服电机, 13

**M**

命令表工艺对象  
    工具, 161  
    用途, 161  
扩展参数, 182  
快捷菜单命令, 171  
命令表组态, 165  
变量 command[1...32], 385, 517  
变量 Config.Command.Command[1...32], 516  
组态下一步, 170

- 组态步代码, 170  
 组态位置/行进路径, 167  
 组态使用轴参数, 166  
 组态命令类型, 166  
 组态持续时间, 169  
 组态速度, 168  
 组态窗口图标, 163  
 组态激活警告, 165  
 基本参数, 165  
 常规组态, 165  
 添加新对象, 162
- D**
- 定位轴  
 变量, 336, 471  
 定位轴工艺对象  
 Actor 变量, 339, 486  
 DynamicDefaults 变量, 358, 497  
 DynamicLimits 变量, 357, 496  
 ErrorBits 变量, 380, 514  
 FollowingError 变量, 366, 503  
 Homing 变量, 364, 501  
 Modulo 变量, 356, 495  
 PositionControl 变量, 365, 502  
 PositionLimitsHW 变量, 361, 500  
 PositionLimitsSW 变量, 359, 498  
 PositionMonitoring 变量, 367, 504  
 PTO 和 HSC 组态, 84, 403  
 StandstillSignal 变量, 368, 505  
 StatusBits 变量, 374, 509  
 StatusDrive 变量, 371, 507  
 StatusPositioning 变量, 369, 506  
 StatusSensor 变量, 372, 508  
 工具, 76  
 计量单位设置, 83
- 用户单位组态, 405  
 主动回原点, 122  
 扩展参数, 406  
 机械变量, 355, 494  
 机械组态, 97, 98, 406  
 在用户程序中可以更改归位组态参数, 125  
 在用户程序中可以更改动态组态参数, 114  
 传感器变量, 345, 488  
 仿真变量, 338  
 更新变量, 384, 515  
 位置变量, 337, 485  
 启用冲击限制器时的响应, 113  
 诊断概览, 77  
 驱动器使能组态, 406  
 驱动器信号组态, 86, 406  
 驱动器准备就绪组态, 406  
 变量 ActualPosition, 337  
 变量 ActualVelocity, 337  
 单位变量, 354, 493  
 组态反转方向信号, 98, 407  
 组态允许在硬限位开关处自动反向, 409  
 组态电机每转的脉冲数, 97, 406  
 组态电机每转的距离, 97, 406  
 组态主动回原点, 117, 408  
 组态加加速度限值, 109  
 组态加速时间, 108  
 组态加速度, 108  
 组态回原点开关输入, 118, 121, 409  
 组态回原点开关端, 119, 121, 410  
 组态回原点开关端, 119, 121, 410  
 组态回原点开关端, 119, 121, 410  
 组态回原点速度, 119, 410  
 组态参考点开关输入, 408  
 组态参考点开关端, 408  
 组态参考点位置, 119, 121, 408, 410  
 组态起始位置偏移值, 119, 410

组态速度限值单位, 107

组态被动归位, 408

组态被动回原点, 120

组态减速时间, 108

组态减速度, 108

组态逼近/回原点方向, 119, 409

组态最大速度/启动/停止速度, 107

组态窗口图标, 80

组态概述, 77

组态滤波时间, 110

轴名称组态, 82, 402

急停减速度组态, 111

速度变量, 337

被动回原点, 124

调试概述, 77

基本参数, 82, 402

常规动态组态, 107

添加新对象, 79

硬件和软件组件, 72

硬件接口组态, 84, 403

定位轴工艺对象的变量, 336, 471

定位监控, 127

## Z H

轴和命令表工艺对象

ErrorID 和 ErrorInfo 列表, 300, 414

## S H

适用范围

手册, 3

## Z H

重新启动工艺对象, 218

重新初始化工艺对象, 218

## M

脉冲接口

原理, 20

## J

基本知识

必需, 3

## T

停止信号, 129

添加分隔符行 (Add separator line), 171

添加空行 (Add empty line), 171

## C H

插入分隔符行 (Insert separator line), 171

插入空行 (Insert empty line), 171

插补器 OB, 39, 42

## Y

硬和软限位开关

功能, 45

## G

跟随误差监控, 128

## X

需要的基本知识, 3