

SIEMENS

SIMATIC

STEP 7 S7-1200 运动控制 V11 SP2

功能手册

前言

使用 S7-1200 运动控制

1




S7-1200 运动控制

2

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。

 危险
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。
 警告
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。
 小心
带有警告三角，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。
小心
不带警告三角，表示如果不采取相应的小心措施，可能导致财产损失。
注意
表示如果不注意相应的提示，可能会出现不希望的结果或状态。


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导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。

Siemens 产品

请注意下列说明：

 警告
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 ® 的都是西门子股份有限公司的注册商标。标签中的其他符号可能是一些其他商标，这是出于保护所有权利的目的由第三方使用而特别标示的。

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版本中。

前言

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。此文档在内容和结构方面与 **STEP 7 V11 SP2** 在线帮助一致。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 V11 SP2**。

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，请与我们联系。请与我们的当地办事处的 **Siemens** 合作伙伴联系。

可在以下网站找到有关联系人的信息：

(<http://www.siemens.com/automation/partner>)

以下网站提供各种 **SIMATIC** 产品和系统的文档标记说明：

(<http://www.siemens.com/simatic-tech-doku-portal>)

可在以下网站找到在线产品目录和订购系统：

(<http://mall.automation.siemens.com>)

培训中心

我们提供了一系列课程来帮助您开始使用 S7 可编程控制器。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>)。

目录

前言	3
1 使用 S7-1200 运动控制	9
1.1 简介	9
1.1.1 CPU S7-1200 的运动功能	9
1.1.2 用于运动控制的硬件组件	10
1.2 使用 S7-1200 运动控制的基础知识	13
1.2.1 与运动控制相关的 CPU 输出	13
1.2.2 脉冲接口的工作原理	16
1.2.3 行进方向与方向输出处电压电平间的关系	17
1.2.4 硬件和软件限位开关	19
1.2.5 冲击限制	20
1.2.6 回原点	21
1.3 运动控制使用指南	22
1.4 版本概述	23
1.5 轴工艺对象	26
1.5.1 集成轴工艺对象	26
1.5.2 轴工艺对象的工具	30
1.5.3 添加工艺对象“轴”	32
1.5.4 组态轴工艺对象	33
1.5.4.1 使用组态对话框	33
1.5.4.2 基本参数	34
1.5.4.3 扩展参数	36
1.6 工艺对象命令表	57
1.6.1 使用命令表工艺对象	57
1.6.2 命令表工艺对象工具	57
1.6.3 添加工艺对象命令表	58
1.6.4 组态命令表工艺对象	59
1.6.4.1 使用组态对话框	59
1.6.4.2 基本参数	60
1.6.4.3 扩展参数	76
1.7 下载到 CPU	79
1.8 调试轴 - 轴控制面板	81
1.9 编程	84
1.9.1 运动控制语句概述	84
1.9.2 创建用户程序	85

1.9.3	编程注意事项.....	88
1.9.4	断电和重新启动后运动控制命令的行为	90
1.9.5	监视激活的命令	91
1.9.5.1	监视激活的命令	91
1.9.5.2	具有输出参数“Done”的运动控制指令	91
1.9.5.3	运动控制指令 MC_MoveVelocity	95
1.9.5.4	运动控制指令 MC_MoveJog.....	100
1.9.6	运动控制语句的错误显示.....	104
1.10	轴 - 诊断	106
1.10.1	状态和错误位.....	106
1.10.2	运动状态.....	109
1.10.3	动态设置.....	110
1.11	使用监视表格.....	111
1.12	附录.....	113
1.12.1	将多个轴与同一个 PTO 一起使用.....	113
1.12.2	将多个驱动器与同一个 PTO 一起使用.....	116
1.12.3	在更高优先级等级（执行级别）中跟踪作业	117
1.12.4	使用软件限位开关的特殊情况.....	120
1.12.4.1	软件限位开关与回原点操作结合使用	120
1.12.4.2	软件限位开关和软件限位开关的位置变化。	124
1.12.4.3	软件限位开关与动态更改结合使用	125
1.12.5	降低短时间定位的速度	127
1.12.6	启动/停止速度的动态调整.....	127
1.12.7	ErrorID 和 ErrorInfo 列表（从 V2.0 的工艺对象起）	128
1.12.8	轴工艺对象的变量	140
1.12.8.1	Config. 变量	140
1.12.8.2	MotionStatus. 变量.....	155
1.12.8.3	StatusBits. 变量.....	156
1.12.8.4	ErrorBits. 变量	160
1.12.8.5	Internal. 变量.....	162
1.12.8.6	ControlPanel. 变量	162
1.12.8.7	工艺对象变量的更新	163
1.12.9	命令表工艺对象变量.....	163
1.12.9.1	Config.Command.Command[1 ... 32] 变量.....	163
1.12.10	先前版本的功能的文档	165
1.12.10.1	组态 - 回原点（V1.0“轴”工艺对象）	165
1.12.10.2	ErrorID 和 ErrorInfo 的列表（工艺对象 V1.0）	168
2	S7-1200 运动控制.....	177
2.1	MC_Power	177
2.1.1	MC_Power: 启用、禁用轴	177
2.1.2	MC_Power: 功能图	181
2.2	MC_Reset	182

2.2.1	MC_Reset: 确认错误.....	182
2.3	MC_Home	185
2.3.1	MC_Home: 归位轴, 设置归位位置.....	185
2.4	MC_Halt	189
2.4.1	MC_Halt: 停止轴	189
2.4.2	MC_Halt: 功能图	191
2.5	MC_MoveAbsolute	193
2.5.1	MC_MoveAbsolute: 轴的绝对定位	193
2.5.2	MC_MoveAbsolute: 功能图.....	196
2.6	MC_MoveRelative	198
2.6.1	MC_MoveRelative: 轴的相对定位	198
2.6.2	MC_MoveRelative: 功能图.....	201
2.7	MC_MoveVelocity.....	203
2.7.1	MC_MoveVelocity: 以预设的旋转速度移动轴.....	203
2.7.2	MC_MoveVelocity: 功能图	206
2.8	MC_MoveJog	208
2.8.1	MC_MoveJog: 在点动模式下移动轴.....	208
2.8.2	MC_MoveJog: 功能图	211
2.9	MC_CommandTable	212
2.9.1	MC_CommandTable: 按移动顺序运行轴作业 (从 V2.0“轴”工艺对象起)	212
2.10	MC_ChangeDynamic	216
2.10.1	MC_ChangeDynamic: 更改轴的动态设置 (从 V2.0“轴”工艺对象起)	216
	索引	219

使用 S7-1200 运动控制

1.1 简介

1.1.1 CPU S7-1200 的运动功能

TIA Portal 结合 CPU S7-1200 的“运动控制”功能，可帮助用户实现通过脉冲接口控制步进电机和伺服电机：

- 在 TIA Portal 中，可以组态“轴”和“命令表”工艺对象。CPU S7-1200 可以使用这些工艺对象控制用于控制驱动器的脉冲和方向输出。
- 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作业。

参见

用于运动控制的硬件组件 (页 10)

集成轴工艺对象 (页 26)

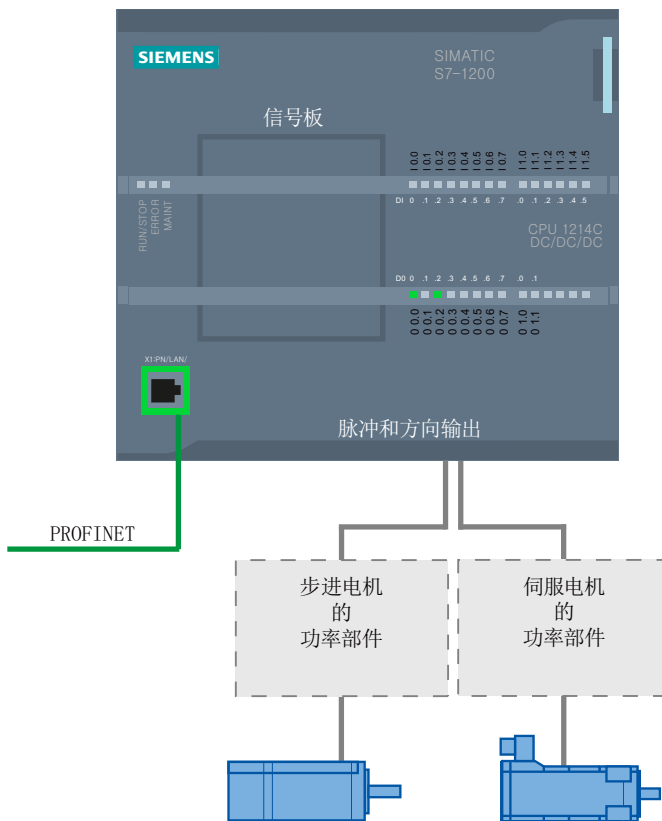
使用命令表工艺对象 (页 57)

命令表工艺对象工具 (页 57)

1.1 简介

1.1.2 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通过脉冲接口控制步进电机和伺服电机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。

DC/DC/DC 型 CPU S7-1200 上配备有用于直接控制驱动器的板载输出。继电器型 CPU 需要使用下述信号板之一来控制驱动器。

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和方向输出。

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。

同时使用 DC/DC/DC 型 CPU S7-1200 和信号板时，可控制驱动器的最大数目为“2”个。

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。

可控制驱动器的最大数目

下表列出了各种 CPU 可控制驱动器的最大数目：

CPU		信号板					
		无	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	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

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	V1.0“轴”工艺对象的限制频率	V2.0 及更高版本“轴”工艺对象的限制频率
板载	$2 \text{ Hz} \leq f \leq 100 \text{ kHz}$	$2 \text{ Hz} \leq f \leq 100 \text{ kHz}$
信号板 DI2/DO2 x DC24V 20kHz	$2 \text{ Hz} \leq f \leq 20 \text{ kHz}$	$2 \text{ Hz} \leq f \leq 20 \text{ kHz}$
信号板 DI2/DO2 x DC24V 200kHz	$2 \text{ Hz} \leq f \leq 100 \text{ kHz}$	$2 \text{ Hz} \leq f \leq 200 \text{ kHz}$
信号板 DO4 x DC24V 200kHz	$2 \text{ Hz} \leq f \leq 100 \text{ kHz}$	$2 \text{ Hz} \leq f \leq 200 \text{ kHz}$
信号板 DI2/DO2 x DC5V 200kHz	$2 \text{ Hz} \leq f \leq 100 \text{ kHz}$	$2 \text{ Hz} \leq f \leq 200 \text{ kHz}$
信号板 DO4 x DC5V 200kHz	$2 \text{ Hz} \leq f \leq 100 \text{ kHz}$	$2 \text{ Hz} \leq f \leq 200 \text{ kHz}$

1.1 简介

订购信息

下列订购信息适用于 TIA Portal 中的当前已安装产品系列（不包括所有已安装的硬件支持包）。

名称	MLFB (订货号)
CPU 1211C DC/DC/DC	6ES7211-1AD30-0XB0
CPU 1211C AC/DC/RLY	6ES7211-1BD30-0XB0
CPU 1211C DC/DC/RLY	6ES7211-1HD30-0XB0
CPU 1212C DC/DC/DC	6ES7212-1AD30-0XB0
CPU 1212C AC/DC/RLY	6ES7212-1BD30-0XB0
CPU 1212C DC/DC/RLY	6ES7212-1HD30-0XB0
CPU 1214C DC/DC/DC	6ES7214-1AE30-0XB0
CPU 1214C AC/DC/RLY	6ES7214-1BE30-0XB0
CPU 1214C DC/DC/RLY	6ES7214-1HE30-0XB0
信号板 DI2/DO2 x DC24V 20kHz	6ES7223-0BD30-0XB0
信号板 DI2/DO2 x DC24V 200kHz	6ES7223-3BD30-0XB0
信号板 DO4 x DC24V 200kHz	6ES7222-1BD30-0XB0
信号板 DI2/DO2 x DC5V 200kHz	6ES7223-3AD30-0XB0
信号板 DO4 x DC5V 200kHz	6ES7222-1AD30-0XB0

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。

参见

CPU S7-1200 的运动功能 (页 9)

与运动控制相关的 CPU 输出 (页 13)

1.2 使用 S7-1200 运动控制的基础知识

1.2.1 与运动控制相关的 CPU 输出

脉冲和方向输出

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。板载 CPU 输出或信号板输出可用作脉冲输出和方向输出。在设备组态期间，可以在“属性”(Properties) 选项卡的脉冲发生器 (PTO/PWM) 中，选择板载 CPU 输出或信号板输出。

下表列出了脉冲输出和方向输出的地址分配：

CPU S7-1200:	不使用信号板				信号板 DI2/DO2 *)				信号板 DO4 **)			
	输出 PTO1		输出 PTO2		输出 PTO1		输出 PTO2		输出 PTO1		输出 PTO2	
	脉冲	方向	脉冲	方向	脉冲	方向	脉冲	方向	脉冲	方向	脉冲	方向
CPU 1211C、CPU 1212C、CPU 1214C (DC/DC/DC)	Ax. .0	Ax.1	Ax. 2	Ax.3	Ax. 0	Ax.1	Ax. 2	Ax.3	Ax. 0	Ax.1	Ax.2	Ax.3
					Ay. 0	Ay.1			Ay. 0	Ay.1	Ay.2	Ay.3
CPU 1211C、CPU 1212C、CPU 1214C (AC/DC/RLY)	-	-	-	-	Ay. 0	Ay.1	-	-	Ay. 0	Ay.1	Ay.2	Ay.3
CPU 1211C、CPU 1212C、CPU 1214C (DC/DC/RLY)	-	-	-	-	Ay. 0	Ay.1	-	-	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）

* 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 的信号。

** 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 和 PTO2 的信号。

驱动器接口

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。

说明

如果已选择 PTO (Pulse Train Output) 并将其分配给一个轴，则固件将通过相应的脉冲和方向输出执行控制。

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格写入脉冲输出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。

参见

脉冲接口的工作原理 (页 16)

行进方向与方向输出处电压电平间的关系 (页 17)

硬件和软件限位开关 (页 19)

冲击限制 (页 20)

回原点 (页 21)

用于运动控制的硬件组件 (页 10)

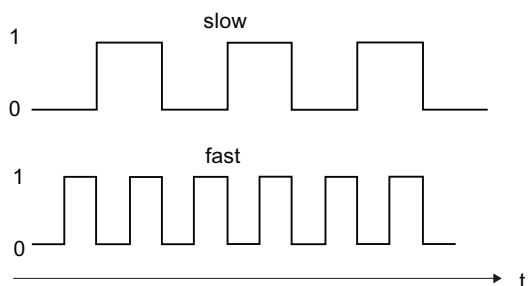
集成轴工艺对象 (页 26)

轴工艺对象的工具 (页 30)

1.2.2 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，则每个脉冲电机移动 0.36° 。

步进电机的速度通过每单位时间的脉冲数来确定。



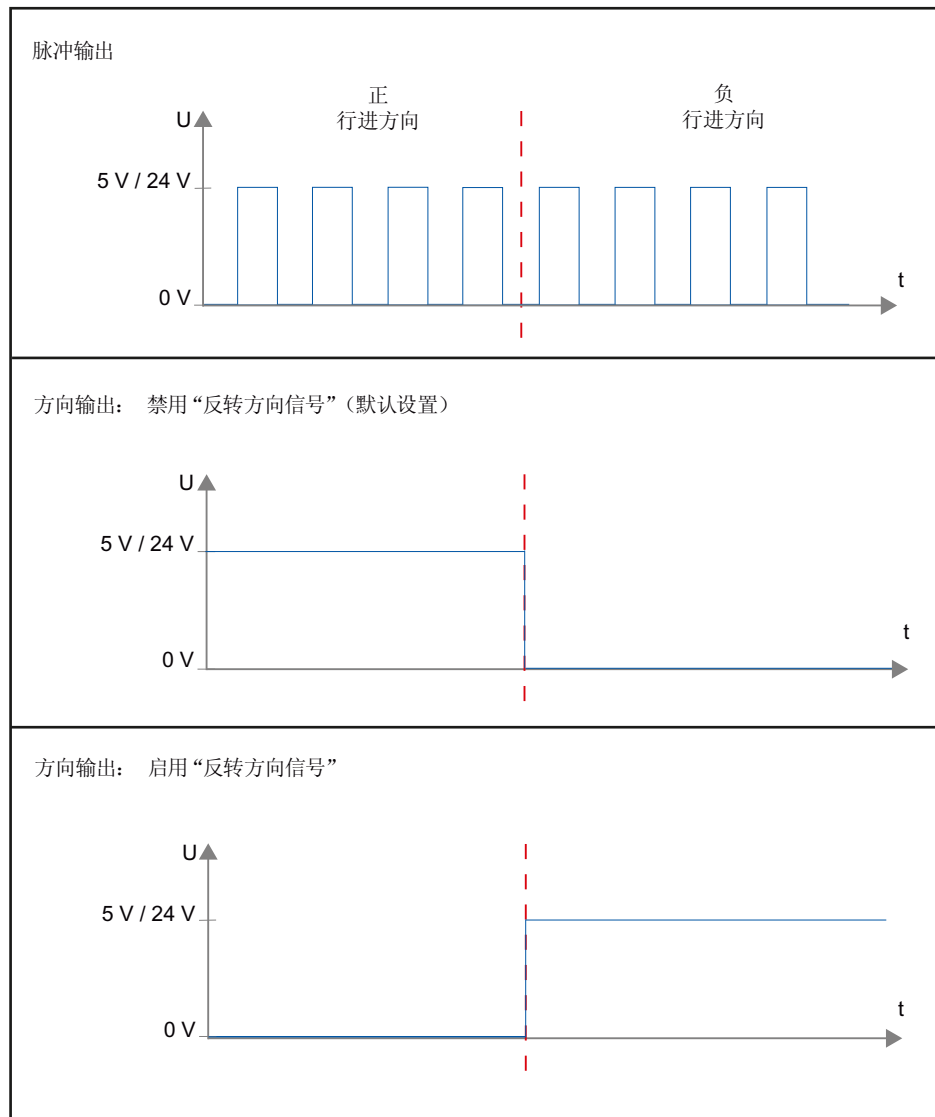
(此处所做的说明同样适用于带脉冲接口的伺服电机。)

参见

- 与运动控制相关的 CPU 输出 (页 13)
- 行进方向与方向输出处电压电平间的关系 (页 17)
- 硬件和软件限位开关 (页 19)
- 冲击限制 (页 20)
- 回原点 (页 21)
- 集成轴工艺对象 (页 26)
- 轴工艺对象的工具 (页 30)

1.2.3 行进方向与方向输出处电压电平间的关系

CPU 的方向输出指定的是驱动器的行进方向。在轴组态期间中，可在“机械”(Mechanics) 下组态方向信号。下图显示了组态、方向输出和行进方向间的关系：



如果在组态中禁用“反转方向信号”(Invert direction signal)，则对于正行进方向（电压取决于使用的硬件），将在方向输出上输出 5 V/24 V 的电平。如果在组态中激活“反转方向信号”(Invert direction signal)，则对于正行进方向，将在方向输出上输出 0 V 电平。

参见

与运动控制相关的 CPU 输出 (页 13)

脉冲接口的工作原理 (页 16)

硬件和软件限位开关 (页 19)

冲击限制 (页 20)

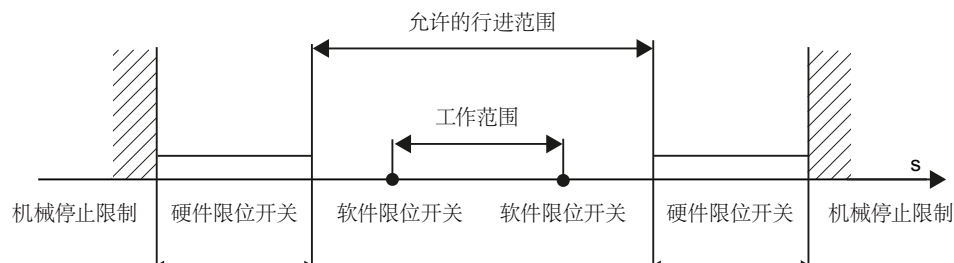
回原点 (页 21)

集成轴工艺对象 (页 26)

轴工艺对象的工具 (页 30)

1.2.4 硬件和软件限位开关

硬件和软件限位开关用于限制轴工艺对象的“允许行进范围”和“工作范围”。这两者的相互关系如下图所示：



硬件限位开关是限制轴的最大“允许行进范围”的限位开关。硬件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。

软件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件限位开关的内侧。由于软件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。与硬件限位开关不同，软件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件限位开关之前，必须先事先将其激活。只有在轴回原点之后，才可以激活软件限位开关。

参见

与运动控制相关的 CPU 输出 (页 13)

脉冲接口的工作原理 (页 16)

行进方向与方向输出处电压电平间的关系 (页 17)

冲击限制 (页 20)

回原点 (页 21)

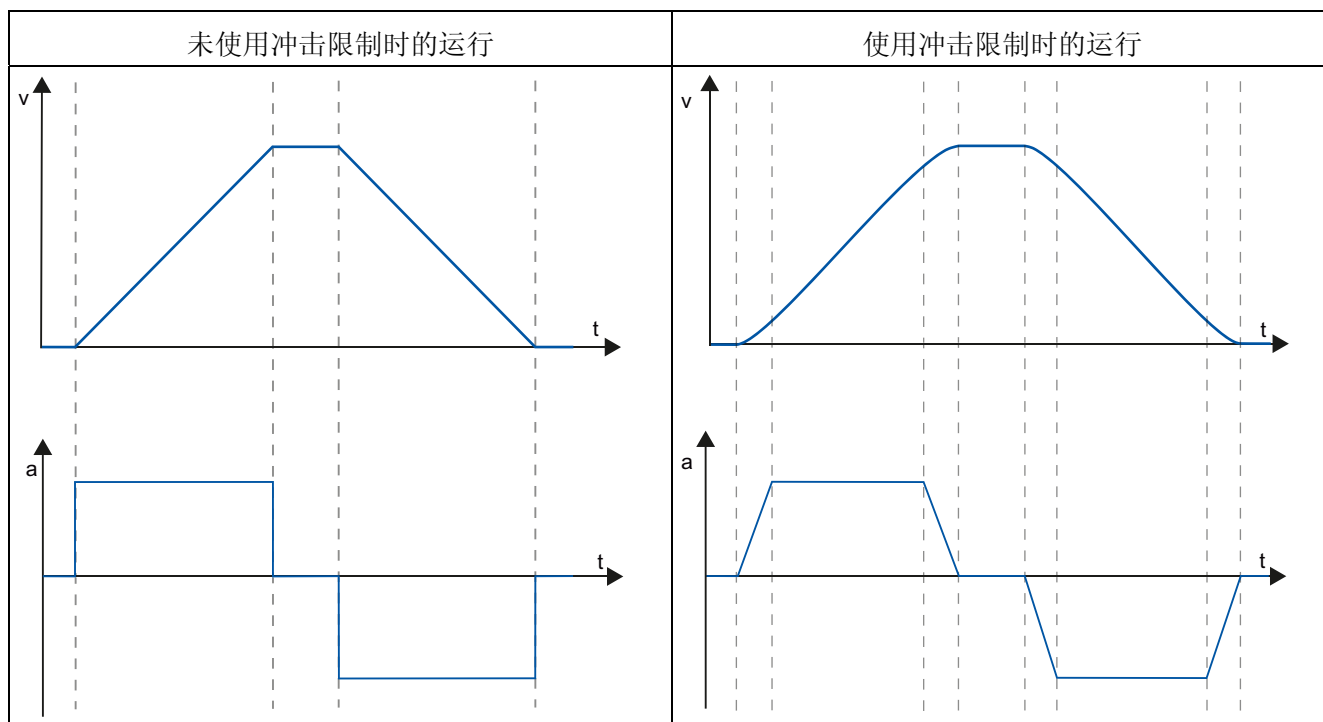
集成轴工艺对象 (页 26)

轴工艺对象的工具 (页 30)

位置限制 (页 38)

1.2.5 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 47)

与运动控制相关的 CPU 输出 (页 13)

脉冲接口的工作原理 (页 16)

行进方向与方向输出处电压电平间的关系 (页 17)

硬件和软件限位开关 (页 19)

回原点 (页 21)

集成轴工艺对象 (页 26)

轴工艺对象的工具 (页 30)

1.2.6 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，上述步骤必不可少。

在 S7-1200 CPU 中，使用运动控制指令“MC_Home”执行轴回原点。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**

在被动回原点期间，运动控制指令“MC_Home”不会执行任何回原点运动。用户必须通过其它运动控制指令，执行这一步骤中所需的往返运动。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时轴的定位：

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

参见

- 与运动控制相关的 CPU 输出 (页 13)
- 脉冲接口的工作原理 (页 16)
- 行进方向与方向输出处电压电平间的关系 (页 17)
- 硬件和软件限位开关 (页 19)
- 冲击限制 (页 20)
- 集成轴工艺对象 (页 26)
- 轴工艺对象的工具 (页 30)
- 回原点（从 V2.0“轴”工艺对象起） (页 50)

1.3 运动控制使用指南

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“轴”工艺对象，必须创建具有 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. 添加工艺对象“轴” (页 32)
2. 使用组态对话框 (页 33)
3. 下载到 CPU (页 79)
4. 在调试窗口中对轴执行功能测试 (页 81)
5. 编程 (页 84)
6. 对轴控制执行诊断 (页 106)

1.4 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

可以在任务卡“指令 > 工艺 > 运动控制 > S7-1200 运动控制”(Instructions > Technology > Motion Control > S7-1200 Motion Control) 和“添加新对象”(Add new object) 对话框中检查当前选定的工艺版本。在任务卡“指令 > 工艺 > 运动控制 > S7-1200 运动控制”(Instructions > Technology > Motion Control > S7-1200 Motion Control) 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“装载到设备”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) 字段中检查工艺对象的版本。

要更改版本，请在“指令 > 工艺”(Instructions > Technology) 任务卡下选择相关版本，然后选择菜单命令“编辑 > 编译”(Edit > Compile)。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，则工艺对象版本也会更改。

如果编译期间显示错误信息，请排除所有错误的原因。重复编译，直到编译可以无错完成。

执行完该操作后，检查工艺对象的组态。

运动控制指令版本

要检查运动控制指令的版本，请按以下步骤操作：

1. 在浏览器中打开“程序块 > 系统块 > 程序资源”文件夹并选择所需的运动控制指令。
2. 选择“编辑 > 属性”(Edit > Properties) 菜单命令。
3. 将会在“信息”(Information) 选项卡的“版本”(Version) 字段中找到运动控制指令版本。

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。

兼容性列表

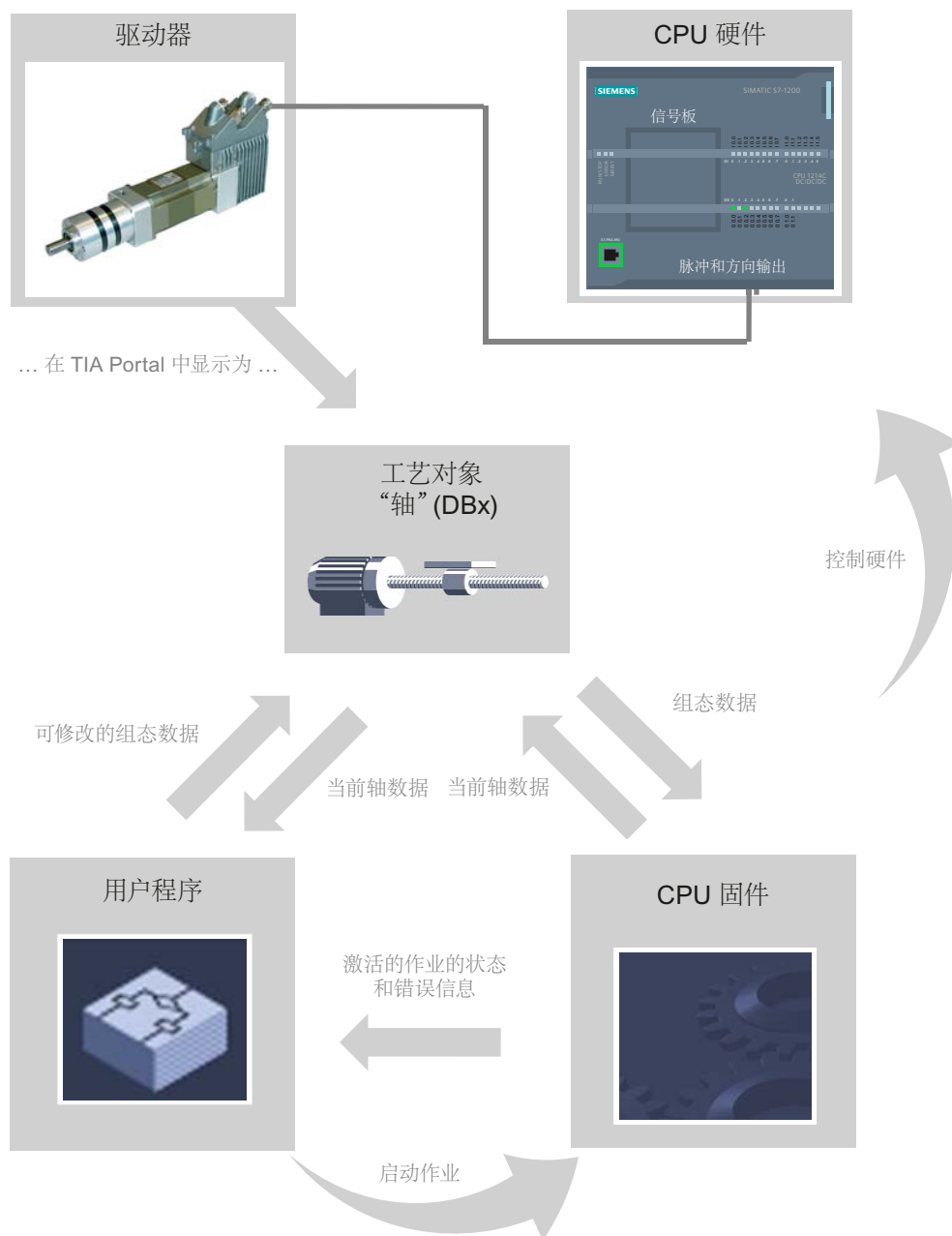
工艺		CPU	工艺对象	运动控制指令
V1.0		V1.0、V2.0、 V2.1、V2.2	轴 V1.0	MC_Power V1.0 MC_Reset V1.0 MC_Home V1.0 MC_Halt V1.0 MC_MoveAbsolute V1.0 MC_MoveRelative V1.0 MC_MoveVelocity V1.0 MC_MoveJog V1.0
V2.0	创新： <ul style="list-style-type: none"> • 步进控制 • 命令表 • MC_ChangeDynamic 	V2.1、V2.2	轴 V2.0、命令表 V2.0	MC_Power V2.0 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

工艺		CPU	工艺对象	运动控制指令
V3.0	创新： 在 RUN 操作模式下加载	V2.2	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_CommandTable V3.0 MC_ChangeDynamic V3.0

1.5 轴工艺对象

1.5.1 集成轴工艺对象

下图显示了使用“轴”工艺对象时所执行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。

驱动器

驱动器是由动力装置和电机组成的单元。可以使用具有脉冲接口的步进电机或伺服电机。

“轴”工艺对象

包含机械的物理驱动器在 TIA Portal 中映射为“轴”工艺对象。为此，需要使用以下参数组态“轴”工艺对象：

- 要使用的 PTO（Pulse Train Output，脉冲串输出）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
- 位置监视参数、动态参数和回原点参数

“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。

1.5 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：

- 绝对定位轴
- 相对定位轴
- 以设定的速度移动轴
- 按移动顺序运行轴作业（从 V2.0 工艺起）
- 在点动模式下移动轴
- 停止轴
- 参考轴；设置参考点
- 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。

可以使用工艺对象的变量读取组态数据和当前轴数据。可以通过用户程序更改工艺对象的单个可更改变量（例如，当前加速度）。

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。使用轴控制表时，可以通过操作轴控制表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
- 控制驱动器启用，以及脉冲和方向信号
- 监视驱动器，以及硬件和软件限位开关
- 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

与运动控制相关的 CPU 输出 (页 13)

行进方向与方向输出处电压电平间的关系 (页 17)

轴工艺对象的工具 (页 30)

硬件和软件限位开关 (页 19)

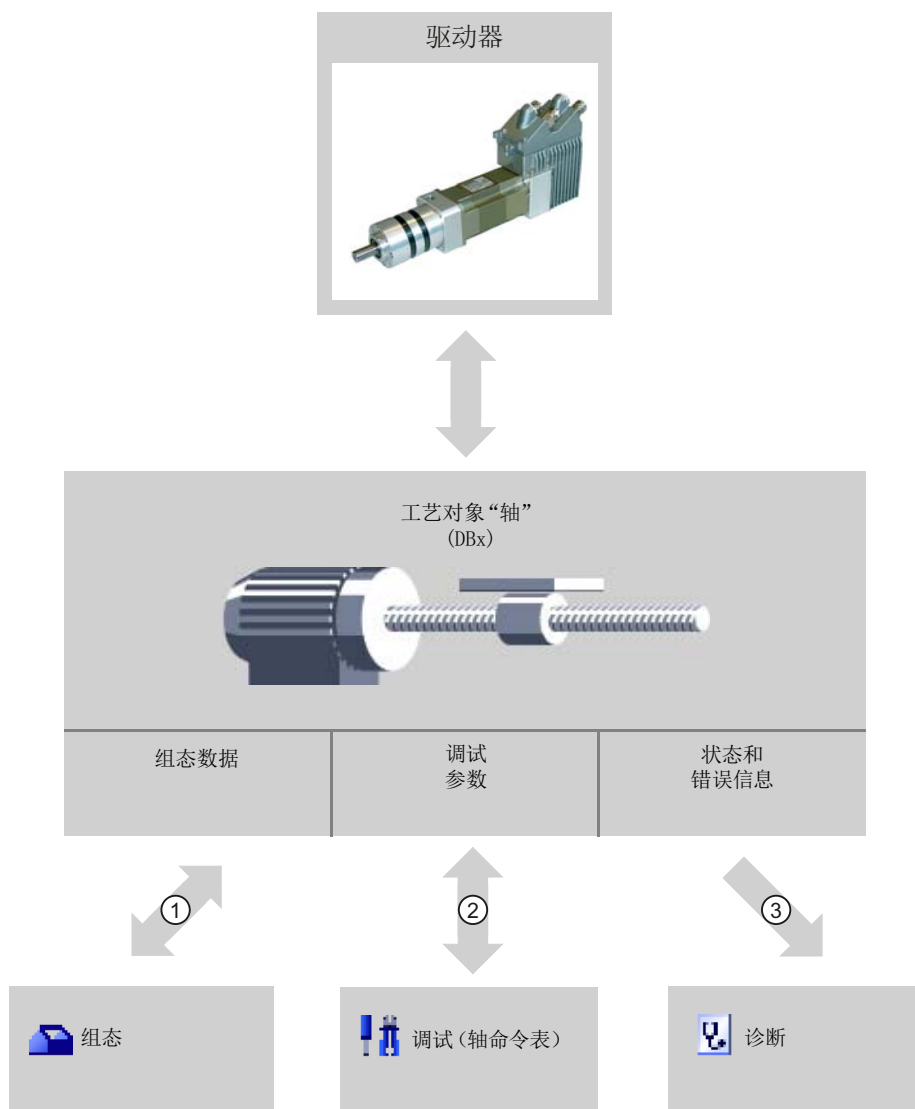
回原点 (页 21)

轴工艺对象的变量 (页 140)

1.5 轴工艺对象

1.5.2 轴工艺对象的工具

TIA Portal 中将为“轴”工艺对象提供“组态”、“调试”和“诊断”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	通过工艺对象对驱动器进行控制；读取轴状态以便在控制面板上显示
③	读取工艺对象的当前状态和错误信息。

组态

使用“组态”工具可以组态“轴”工艺对象的以下属性：

- 选择要使用的 PTO 和驱动器接口的组态
- 机械的属性和驱动器（机器或系统）的传动比参数
- 位置监视属性、动态属性和回原点属性

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，将显示轴控制表。轴控制表提供了下列命令：

- 释放轴和冻结轴
- 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。轴控制表还显示当前的轴状态。

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。

参见

与运动控制相关的 CPU 输出 (页 13)

行进方向与方向输出处电压电平间的关系 (页 17)

集成轴工艺对象 (页 26)

硬件和软件限位开关 (页 19)

回原点 (页 21)

组态轴工艺对象 (页 33)

调试轴 - 轴控制面板 (页 81)

轴 - 诊断 (页 106)

1.5 轴工艺对象

1.5.3 添加工艺对象“轴”

要在项目树中添加“轴”工艺对象，请按以下步骤操作：

要求

已创建具有 CPU S7-1200 的项目。

步骤

1. 在项目树中打开 CPU 文件夹。
2. 打开工艺对象文件夹。
3. 双击“添加新对象”(Add new object)。将打开“添加新对象”(Add new object) 对话框。
4. 选择“运动”(Motion) 工艺。
5. 打开“运动控制”(Motion Control) 文件夹。
6. 打开“S7-1200 运动控制”(S7-1200 Motion Control) 文件夹。
7. 如果要添加旧版本轴，则单击版本并选择相关工艺版本。
8. 选择“TO_Axis_PTO”对象。
9. 在“名称”(Name) 输入字段中更改轴的名称以符合您的需要。
10. 如果要更改推荐的数据块编号，则选择“手动”(Manual) 选项。
11. 如果要为该工艺对象补充用户信息，则单击“更多信息”(More information)。
12. 如果要添加该工艺对象，单击“确定”(OK)。如果要放弃输入，单击“取消”(Cancel)。

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。

参见

运动控制使用指南 (页 22)

1.5.4 组态轴工艺对象

1.5.4.1 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。

组态窗口的图标

组态的区域导航中的图标显示有关组态情况的详细信息：

	<p>组态包含默认值且已完成。</p> <p>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。</p>
	<p>组态包含用户设置的值且已完成。</p> <p>组态的所有输入字段均包含有效值，且至少一个预设值发生更改。</p>
	<p>组态未完成或不正确</p> <p>至少一个输入字段或下拉列表包含无效值。相应字段或下拉列表以红色背景显示。单击弹出错误消息可找出错误原因。</p>
	<p>组态有效但包含警告</p> <p>只组态了一个硬件限位开关。根据设备的不同，缺少硬件限位开关的组态可能引起危险。相应字段或下拉列表以黄色背景显示。</p>

1.5 轴工艺对象

参见

运动控制使用指南 (页 22)

基本参数 (页 34)

扩展参数 (页 36)

1.5.4.2 基本参数

组态 - 常规

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 digital 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 digital 量输出的信号板。

说明

PTO 在内部需要高速计数器 (HSC) 的功能。因此，无法在其它地方使用相应的高速计数器。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。

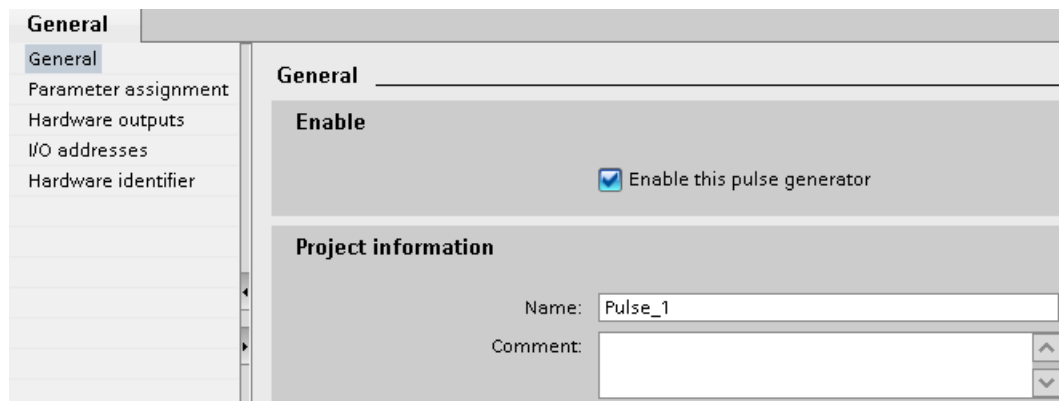
在“脉冲发生器选择”(Pulse generator selection) 下拉列表中，选择 PTO (Pulse Train Output)，它将提供用于控制具有脉冲接口的步进电机或伺服电机的脉冲。如果没有在设备组态中的其它地方使用脉冲发生器和高速计数器，则系统会自动组态硬件接口。这种情况下，下拉列表中所选的 PTO 以白色背景显示。将在“输出源”(Output source)、“脉冲输出”(Pulse output)、“方向输出”(Direction output) 和“已分配高速计数器”(Assigned fast counter) 输出字段中列出所用接口。

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。

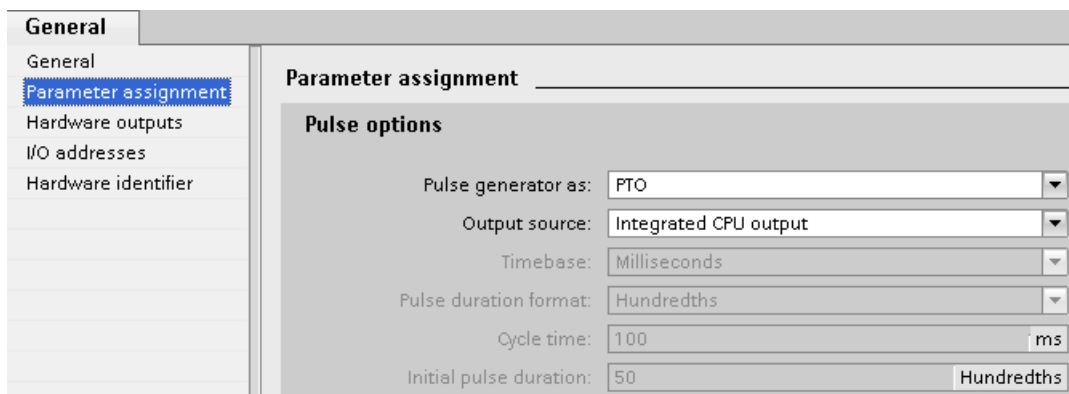
将打开脉冲发生器设备组态。

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. 在块浏览器中，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，具体取决于插入的信号板。更多详细信息，请参见章节：与运动控制相关的 CPU 输出 (页 13)
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。

1.5 轴工艺对象

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“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。

注意

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。

1.5.4.3 扩展参数

组态 - 驱动器接口

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送给 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 (页 37)

位置限制 (页 38)

动态 (页 43)

回原点（从 V2.0“轴”工艺对象起） (页 50)

组态 - 机械

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

在该字段中，可以组态电机每转所需的脉冲数。

限值（与所选测量单位无关）：

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

电机每转的加载距离

在该字段中，将组态电机每转带动单元的机械系统行进的加载距离。

限值（与所选测量单位无关）：

- $0.0 < \text{电机每转的负载距离} \leq 1.0e12$

反转方向信号

可以使用“反转方向信号”(Invert direction signal) 复选框，将方向输出调整为驱动器的方向逻辑。

- **反转方向信号：禁用**

0 V 电平 = 负行进方向

5 V/24 V 电平 = 正行进方向（实际电压取决于使用的硬件）

- **反转方向信号：激活**

0 V 电平 = 正行进方向

5 V/24 V 电平 = 负行进方向（实际电压取决于使用的硬件）

参见

组态 - 驱动器接口 (页 36)

位置限制 (页 38)

动态 (页 43)

回原点（从 V2.0“轴”工艺对象起） (页 50)

行进方向与方向输出处电压电平间的关系 (页 17)

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。只有在返回到有效行程范围后，才会取消这种切换状态。

参见

组态 - 位置限制 (页 38)

触发位置限制时的轴操作 (页 40)

在用户程序中更改位置限制的组态 (页 42)

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件限位开关。

启用硬件限位开关

使用此复选框可激活下限和上限硬件限位开关的功能。硬件限位开关可用于在逼近参考点时反转行进方向。有关详细信息，请参见回原点的组态说明。

下限/上限硬件限位开关输入

从下拉列表中为下限或上限硬件限位开关选择数字量输入。该输入必须具有中断功能。数字量板载 CPU 输入和所插入信号板的数字量输入可以选作硬件限位开关的输入。



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件限位开关的输入，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。

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

有效电平

在此下拉列表中，可以选择逼近硬件限位开关时 CPU 输入端存在的信号电平。

- 选择“低电平”

CPU 输入端电平为 0 V (FALSE) 时表示已逼近硬件限位开关

- 选择“高电平”

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

启用软件限位开关

使用此复选框可激活下限和上限软件限位开关的功能。

注意
启用的软件限位开关仅影响已回到原点的轴。

上限和下限软件限位开关

在这些框中可输入下限和上限软件限位开关的位置值。

限值（与所选测量单位无关）：

- $-1.0e12 \leq$ 下限软件限位开关 $\leq 1.0e12$
- $-1.0e12 \leq$ 上限软件限位开关 $\leq 1.0e12$

上限软件限位开关的值必须大于等于下限软件限位开关的值。

参见

对硬件限位开关的要求 (页 38)

触发位置限制时的轴操作 (页 40)

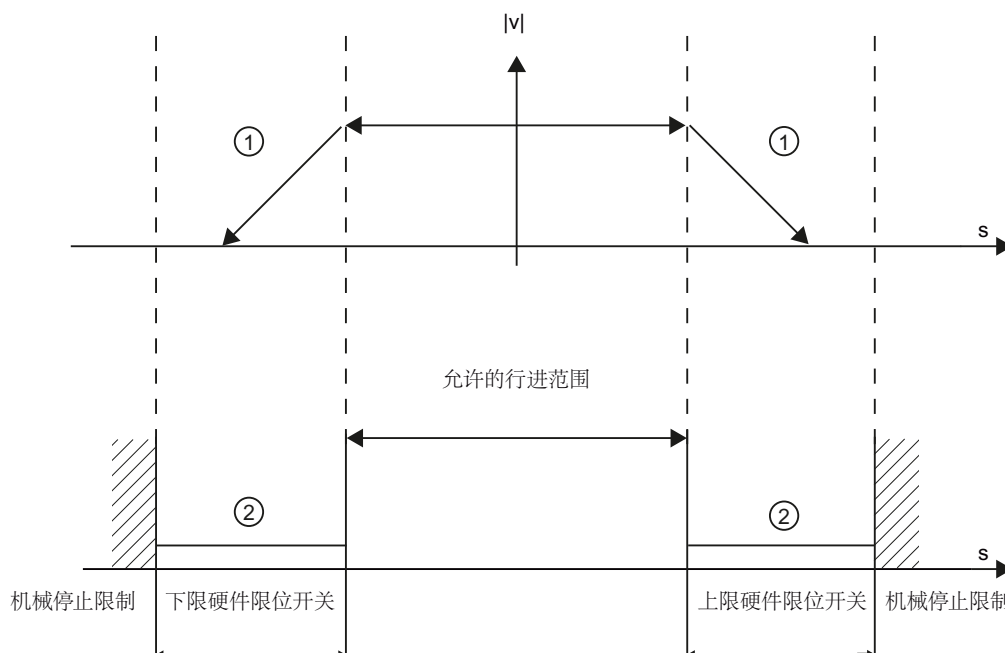
在用户程序中更改位置限制的组态 (页 42)

组态 - 回原点 - 主动 (页 51)

触发位置限制时的轴操作

逼近硬件限位开关时的轴操作

逼近硬件限位开关时，轴将以所组态的急停减速度制动直到停止。指定的急停减速度必须足够大，保证可以在到达机械挡块前可靠地停止轴。下图显示了轴逼近硬件限位开关后的轴操作：



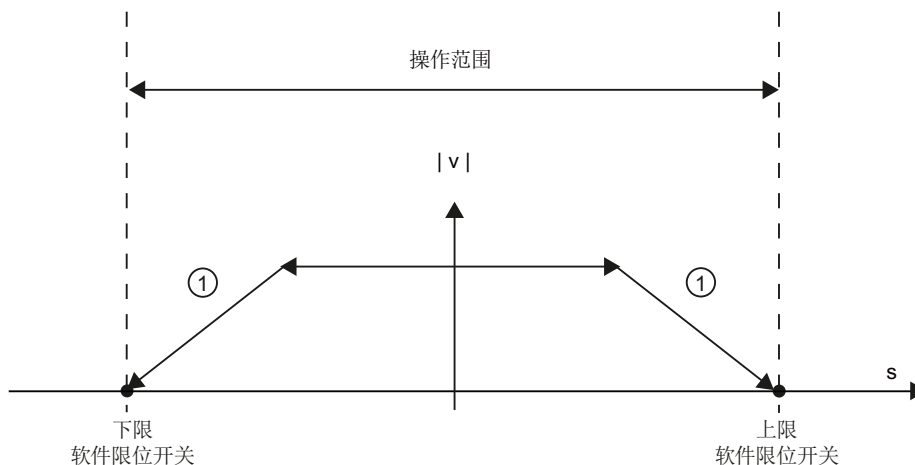
①	轴以所组态的急停减速度制动直到停止。
②	硬件限位开关产生“已逼近”状态信号的范围。

“已逼近硬件限位开关”错误将显示在待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”。

到达软件限位开关时的轴操作

如果软件限位开关激活，则在软件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件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。

“已到达软件限位开关”错误将显示在要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”。

有关显示“超出软件限位开关”错误的各种情况，请参见主题为“软件限位开关与回原点操作结合使用 (页 120)”和“软件限位开关与动态更改结合使用 (页 125)”的章节。

如果机械停止块位于软件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件限位开关。

参见

对硬件限位开关的要求 (页 38)

组态 - 位置限制 (页 38)

在用户程序中更改位置限制的组态 (页 42)

1.5 轴工艺对象

在用户程序中更改位置限制的组态

可以在用户程序在 CPU 中运行期间更改下列组态参数：

硬件限位开关

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

- `<轴名称>.Config.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说明。

软件限位开关

在用户程序运行期间，还可以激活和禁用软件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>.Config.PositionLimits_SW.Active`
用于激活和取消激活软件限位开关
- `<轴名称>.Config.PositionLimits_SW.MinPosition`
用于更改下限软件限位开关的位置
- `<轴名称>.Config.PositionLimits_SW.MaxPosition`
用于更改上限软件限位开关的位置

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说明。

参见

MC_ChangeDynamic: 更改轴的动态设置（从 V2.0“轴”工艺对象起）(页 216)

对硬件限位开关的要求 (页 38)

组态 - 位置限制 (页 38)

触发位置限制时的轴操作 (页 40)

动态

组态 - 常规动态

可以在“常规动态”(General dynamics) 组态窗口中组态最大速度、启动/停止速度、加速度和减速度以及冲击限制（从 V2.0“轴”工艺对象起）。

速度限值单位

从此下拉列表中可选择要用于设置速度限值的测量单位。这里设置的单位取决于“组态 - 常规”(Configuration - General) 下设置的测量单位，设置此单位仅仅是为了便于输入。

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。

限值：

下述限值采用“脉冲/秒”作为测量单位：

- **V2.0 轴工艺对象**
 - $2 \leq \text{启动/停止速度} \leq 20000$ （信号板 20kHz）
 - $2 \leq \text{启动/停止速度} \leq 200000$ （信号板 200kHz）
 - $2 \leq \text{启动/停止速度} \leq 100000$ （车载 CPU 输出）
 - $2 \leq \text{最大速度} \leq 20000$ （信号板 20kHz）
 - $2 \leq \text{最大速度} \leq 200000$ （信号板 200kHz）
 - $2 \leq \text{最大速度} \leq 100000$ （车载 CPU 输出）
- **V1.0 轴工艺对象**
 - $2 \leq \text{启动/停止速度} \leq 20000$ （信号板 20kHz）
 - $2 \leq \text{启动/停止速度} \leq 100000$ （信号板 200kHz）
 - $2 \leq \text{启动/停止速度} \leq 100000$ （车载 CPU 输出）
 - $2 \leq \text{最大速度} \leq 20000$ （信号板 20kHz）
 - $2 \leq \text{最大速度} \leq 100000$ （信号板 200kHz）
 - $2 \leq \text{最大速度} \leq 100000$ （车载 CPU 输出）

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。

1.5 轴工艺对象

加速度/延迟 - 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。在“减速时间”(Deceleration time) 或“减速度”(Deceleration) 框中，可以设置所需减速度。

加速时间和加速度与减速时间和减速度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。

限值：

下述限值采用“脉冲/秒²”作为测量单位：

- $0.28 \leq \text{加速度} \leq 9.5e9$
- $0.28 \leq \text{减速度} \leq 9.5e9$

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。

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。

激活冲击限制（从 V2.0 轴工艺对象起）

使用该复选框可激活冲击限制。

说明

如果发生错误，轴将以组态的急停减速度减速。此时不考虑激活的冲击限制。

平滑时间/冲击（从 V2.0“轴”工艺对象起）

在“平滑时间”(Smoothing time) 字段或“冲击”(Jerk) 字段中，可以输入冲击限制的参数。

- 在“冲击”(Jerk) 字段中，可以为加速和减速斜坡设置所需的冲击。
- 在“取整时间”(Rounding time) 字段中，可以为加速斜坡设置所需的平滑时间。

说明

在组态中显示的所设置的平滑时间仅适用于加速斜坡。

如果加速度值和减速度值不同，则根据加速斜坡的冲击计算减速斜坡的平滑时间并使用该平滑时间。（另请参见 使用冲击限制时轴的行为 (页 47)）

如下所述调整减速度的平滑时间：

- **加速度 > 减速度**
减速斜坡使用的平滑时间比加速斜坡使用的平滑时间短。
 - **加速度 < 减速度**
减速斜坡使用的平滑时间比加速斜坡使用的平滑时间短。
 - **加速度 = 减速度**
加速和减速斜坡的平滑时间相等。
-

平滑时间和冲击之间的关系如下面的方程所示：

$$\text{取整时间 (加速斜坡)} = \frac{\text{加速度}}{\text{步进}}$$

$$\text{取整时间 (减速斜坡)} = \frac{\text{减速度}}{\text{步进}}$$

用户程序中启动的运动作业将使用所选冲击执行。

限值：

下述限值采用“脉冲/秒³”作为测量单位：

- $0.04 \leq \text{冲击} \leq 1.5e8$

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。

参见

使用冲击限制时轴的行为 (页 47)

组态 - 动态急停 (页 46)

在用户程序中可以更改动态组态 (页 49)

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。发生错误或使用运动控制指令“MC_Power”（输入参数 StopMode = 0）禁用轴时，将以该减速度使轴制动直到停止。

速度限值

“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。

减速度

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

急停减速时间和急停减速度间的关系如下面的方程所示：

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

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件限位开关时）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²”作为测量单位：

- $0.28 \leq \text{急停减速度} \leq 9.5e9$

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。

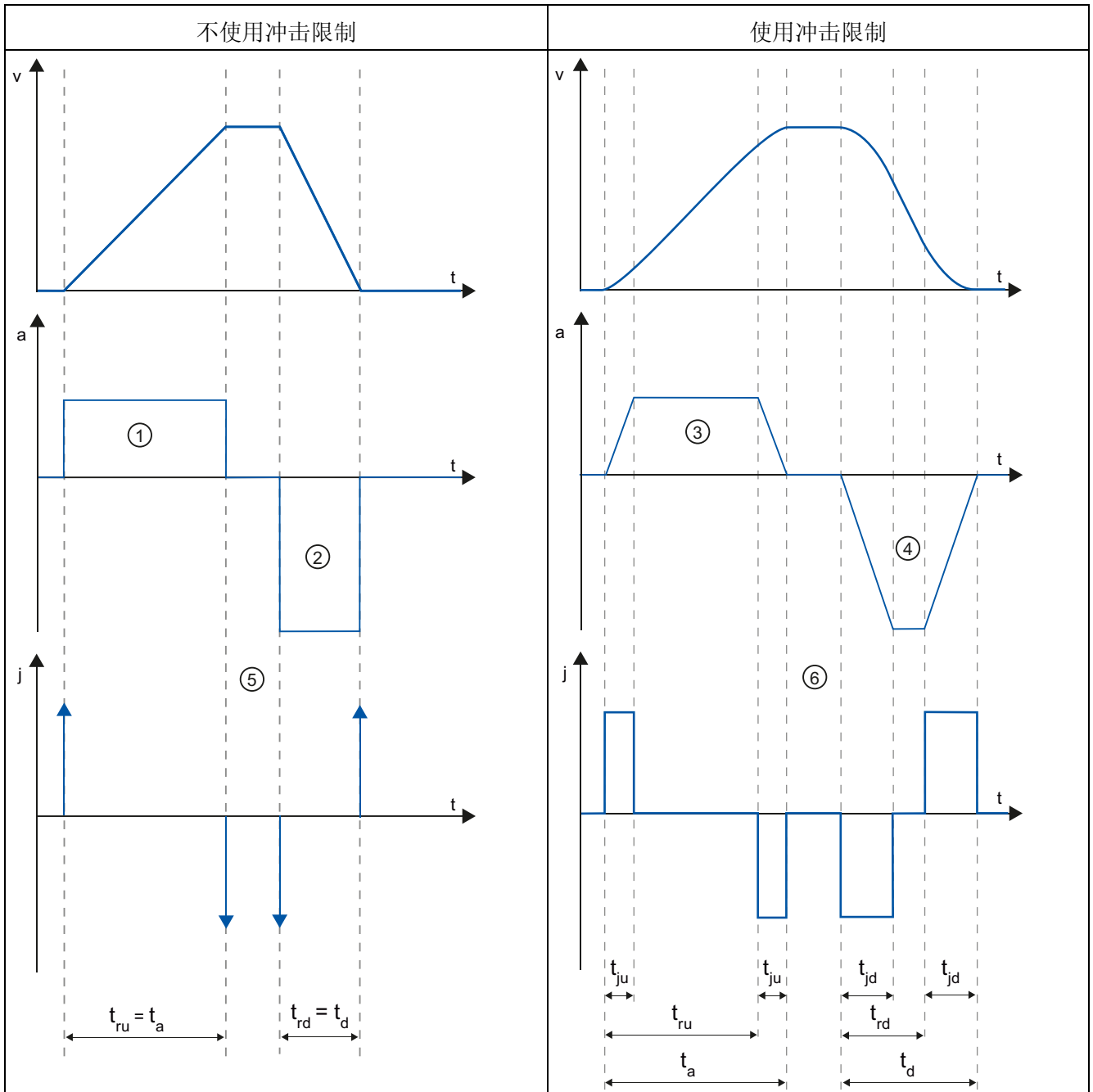
参见

组态 - 常规动态 (页 43)

在用户程序中可以更改动态组态 (页 49)

使用冲击限制时轴的行为

如果激活了冲击限制，则不会突然停止轴加速和轴减速，而是根据设置的步进或取整时间逐渐调整。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：



1.5 轴工艺对象

t	时间轴
v	速度
a	加速度
j	步进
t _{ru}	加速时间
t _a	轴加速所用时间
t _{rd}	减速时间
t _d	轴减速所用时间
t _{ju}	加速斜坡的平滑时间
t _{jd}	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，加速度的加速率和减速度的减速率相同。

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 ⑤。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_{ju} 适用于加速斜坡。会使用组态的冲击值以及组态的减速度计算减速斜坡平滑时间 t_{jd} 。

参见

组态 - 常规动态 (页 43)

在用户程序中可以更改动态组态

可以在用户程序在 CPU 中运行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。



警告

更改该参数后，可能需要调整硬件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。

1.5 轴工艺对象

参见

MC_ChangeDynamic: 更改轴的动态设置（从 V2.0“轴”工艺对象起）(页 216)

组态 - 常规动态 (页 43)

组态 - 动态急停 (页 46)

回原点（从 V2.0“轴”工艺对象起）

组态 - 回原点 - 常规

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

从下拉列表框中为参考点开关选择数字量输入。该输入必须能够生成中断。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。可以在数字量输入的设备组态的“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。

参见

顺序 - 主动回原点 (页 54)

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的必要参数。

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。

参考点开关侧

在此处可以选择轴会在参考点开关的下侧还是上侧回原点。

参考点位置

运动控制指令“MC_Home”中分配的参数指代的位置被用作参考点位置。

说明

如果被动回原点未通过轴运动命令执行（轴处于停止状态），将会在下一个参考点开关处的上升沿或下降沿上执行回原点。

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的必要参数。运动控制指令“MC_Home”的输入参数“Mode”= 3 时，会启动主动回原点。

允许在硬件限位开关处自动反向

激活该复选框，可将硬件限位开关用作指示参考点逼近的反向凸轮。只有启用硬件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件限位开关）。

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（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期间轴到达硬件限位开关，则将因错误中止参考点逼近并以急停减速度对轴进行制动。

注意

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：

- 保持较低的逼近速度
- 增大组态的加速度/减速度
- 增大硬件限位开关和机械停止块间的距离

逼近/回原点方向

通过方向选择，可以决定主动回原点期间用于搜索回原点开关的逼近方向以及回原点方向。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。

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回原点。

1.5 轴工艺对象

速度

在该字段中，可以指定参考点逼近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

在该字段中，可以指定回原点时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

回原点位置偏移

如果预期的参考位置与回原点开关的位置有偏差，则可在该字段中指定回原点位置偏移。

如果值不等于 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回原点位置偏移值
2. 达到“回原点位置偏移”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处。

限值（与所选测量单位无关）：

- $-1.0e12 \leq$ 回原点位置偏移 $\leq 1.0e12$

回原点位置

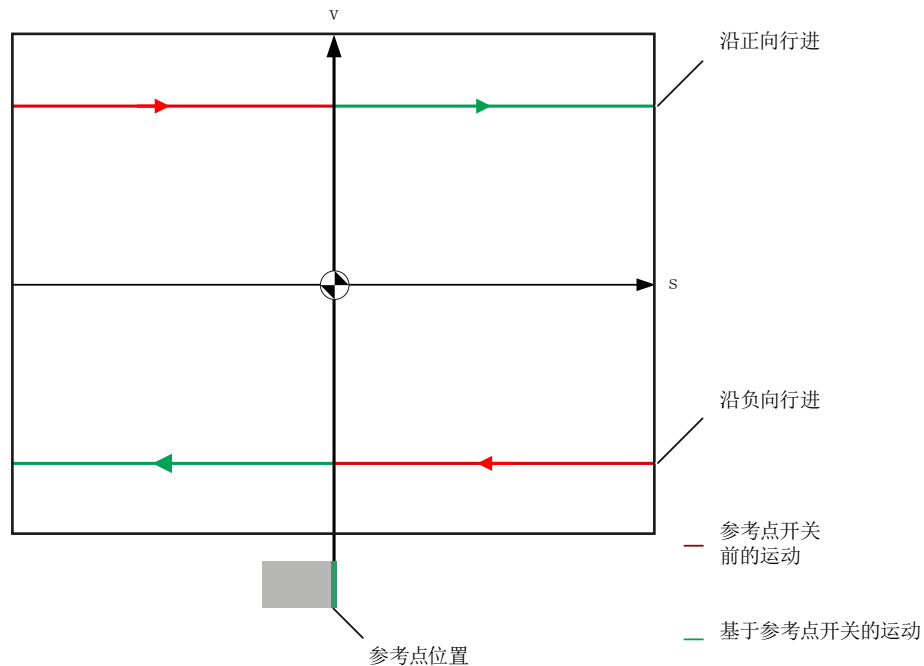
运动控制指令“MC_Home”中分配的参数指代的位置被用作参考点位置。

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：

- “参考点开关侧”=“上侧”



向参考点开关方向移动（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”）来执行到达参考点开关所需的行程。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）

到达参考点开关的组态侧时，轴会回原点。轴的当前位置被设置为参考点位置。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。

移动至超过参考点开关（曲线的绿色部分）

在参考点开关处回原点后，轴继续移动并以正确的轴位置完成之前启动的行程。

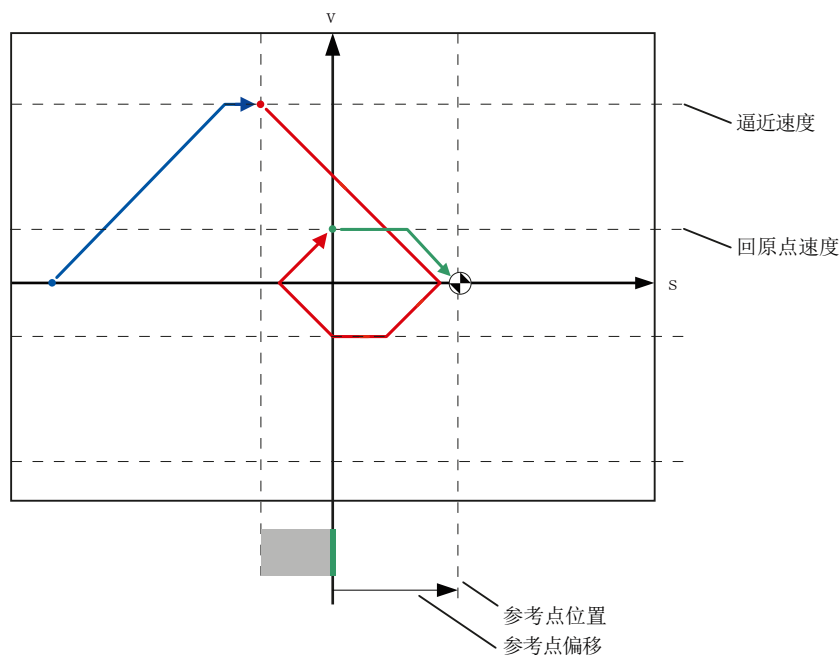
顺序 - 主动回原点

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，也可在轴命令表中启动主动回原点。

下图举例说明了使用以下组态参数时主动参考点逼近的特征曲线：

- “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 (页 50)

在用户程序中更改回原点组态

可以在用户程序运行期间更改 CPU 中的下列组态参数（从 V2.0“轴”工艺对象起）：

被动回原点

可以在用户程序运行期间更改用于被动回原点的回原点开关侧。要执行该操作，可使用以下工艺对象变量：

- `<轴名称>.Config.Homing.SidePassiveHoming`

用于更改回原点开关侧

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。

1.5 轴工艺对象

主动回原点

可以在用户程序运行期间更改主动回原点的逼近方向、回原点开关侧、逼近速度、回原点速度和回原点位置偏移。要执行该操作，可使用以下工艺对象变量：

- `<轴名称>.Config.Homing.AutoReversal`
用于更改“在硬件限位开关处自动反向”
- `<轴名称>.Config.Homing.Direction`
用于更改“逼近/回原点方向”
- `<轴名称>.Config.Homing.SideActiveHoming`
用于更改“回原点开关侧”
- `<轴名称>.Config.Homing.FastVelocity`
用于更改“速度”
- `<轴名称>.Config.Homing.SlowVelocity`
用于更改“回原点速度”
- `<轴名称>.Config.Homing.Offset`
用于更改“回原点位置偏移”

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

MC_ChangeDynamic: 更改轴的动态设置（从 V2.0“轴”工艺对象起）（页 216）

1.6 工艺对象命令表

1.6.1 使用命令表工艺对象

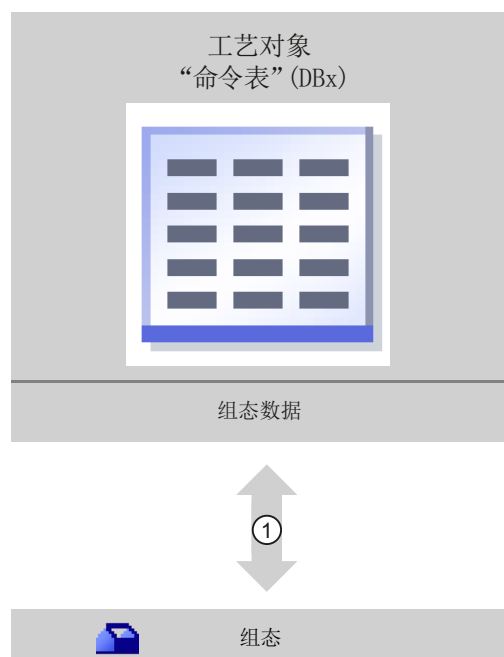
运动控制指令“Command table”允许您将多个单独的轴控制作业合并到一个移动序列中。此工艺对象可以用于自版本 V2.0 起的工艺。

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。

1.6.2 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

1.6 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。

1.6.3 添加工艺对象命令表

要在项目树中添加“命令表”工艺对象，请按以下步骤操作：

前提条件

- 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

1. 在项目树中打开 CPU 文件夹。
2. 打开工艺对象文件夹。
3. 双击“添加新对象”(Add new object)。
将打开“添加新对象”(Add new object) 对话框。
4. 选择“运动”(Motion) 工艺。
5. 打开“运动控制”(Motion Control) 文件夹。
6. 打开“S7-1200 运动控制”(S7-1200 Motion Control) 文件夹。
7. 选择“V2.0”版本的“S7-1200 运动控制”(S7-1200 Motion Control) 文件夹（单击版本对应的条目）。
8. 选择“TO_CommandTable”对象。
9. 在“名称”(Name) 输入字段中更改命令表的名称以符合您的需要。
10. 如果要更改推荐的数据块编号，则选择“手动”(Manual) 选项。
11. 如果要为该工艺对象补充用户信息，则单击“更多信息”(More information)。
12. 如果要添加该工艺对象，单击“确定”(OK)。
如果要放弃输入，单击“取消”(Cancel)。

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。

1.6.4 组态命令表工艺对象

1.6.4.1 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。

组态窗口的图标

组态的区域导航中的图标显示有关组态情况的详细信息：

	<p>组态包含默认值且已完成。</p> <p>组态仅包含默认值。使用这些默认值即可使用工艺对象，无需另做更改。</p>
	<p>组态包含用户设置的值且已完成。</p> <p>组态的所有输入字段均包含有效值，且至少一个预设值发生变更。</p>
	<p>组态未完成或不正确</p> <p>至少一个输入字段或下拉列表包含无效值。相应字段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。</p>
	<p>组态包含彼此不兼容的参数值</p> <p>组态包含在大小或逻辑上互相矛盾的参数值。相应字段或下拉列表以黄色背景显示。</p>

参见

运动控制使用指南 (页 22)

基本参数 (页 60)

扩展参数 (页 76)

1.6 工艺对象命令表

1.6.4.2 基本参数

组态 - 常规

在“常规”(General) 组态窗口中组态工艺对象的名称。

名称

在该字段中定义命令表的名称或“命令表”工艺对象的名称。工艺对象将以该名称在项目树中列出。

参见

组态 - 命令表 (页 60)

快捷菜单命令 - 命令表 (页 65)

使用趋势图 (页 67)

快捷菜单命令 - 曲线图 (页 71)

从“完成命令”过渡到“混合运动” (页 73)

在用户程序中更改命令表组态 (页 75)

组态 - 命令表

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。未显示与达到软件限位开关对应的移动。

激活警告

使用该复选框可在命令表中激活警告显示。

使用轴参数

从下拉列表中，选择使用哪些轴参数来选择图形视图以及检查移动序列。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。

“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。

列：步

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 个作业。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**

绝对定位轴

- **Velocity setpoint**

以设定速度移动轴

- **Wait**

等到给定期结束。Wait 不会停止激活的行进运动。

- **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。

1.6 工艺对象命令表

列：位置

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。
- **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）：

- 对于作业: "Positioning relative"和"Positioning absolute"
 - $1.0e-12 \leq \text{速度} \leq 1.0e12$
- 对于命令: "Velocity setpoint"
 - $-1.0e12 \leq \text{速度} \leq -1.0e-12$
 - $1.0e-12 \leq \text{速度} \leq 1.0e12$
 - 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):
- $0.001s \leq \text{持续时间} \leq 64800s$

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的具体行为,请参见: 从“完成命令”过渡到“混合运动”(页 73)

1.6 工艺对象命令表

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：

- $0 \leq \text{代码编号} \leq 65535$

参见

组态 - 常规 (页 60)

快捷菜单命令 - 命令表 (页 65)

使用趋势图 (页 67)

快捷菜单命令 - 曲线图 (页 71)

从“完成命令”过渡到“混合运动” (页 73)

在用户程序中更改命令表组态 (页 75)

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多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多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。

无法连续添加两个分隔符行，也无法在命令表的结尾添加分隔符行。

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。

1.6 工艺对象命令表

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多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)

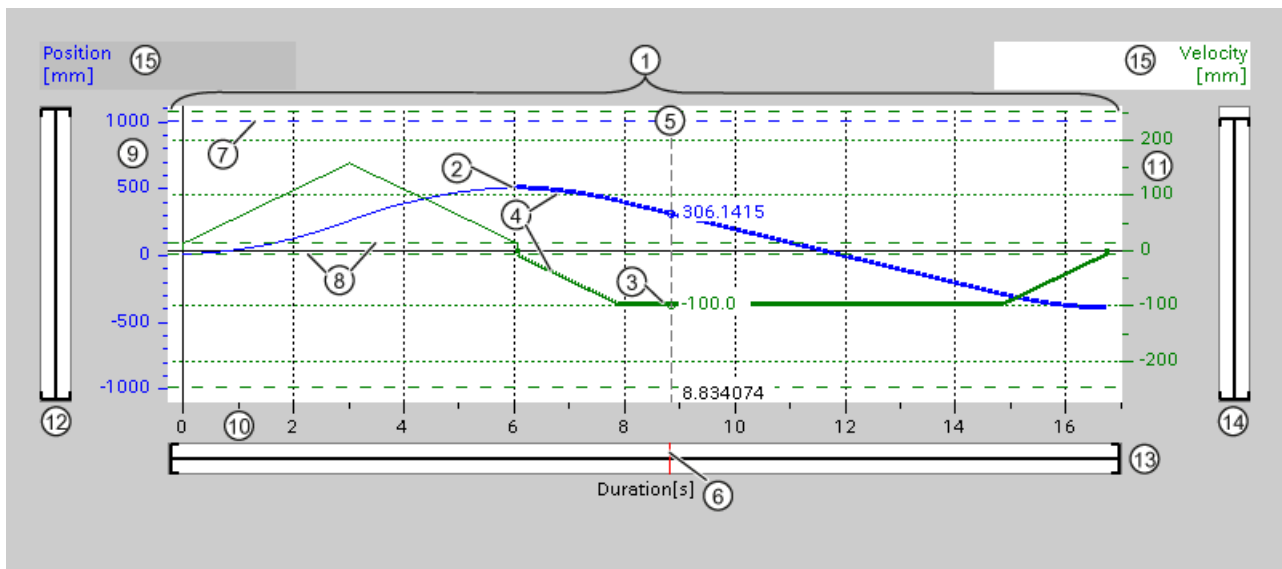
删除所选行。命令表中下方的行将上移。

参见

- 组态 - 常规 (页 60)
- 组态 - 命令表 (页 60)
- 使用趋势图 (页 67)
- 快捷菜单命令 - 曲线图 (页 71)
- 从“完成命令”过渡到“混合运动” (页 73)
- 在用户程序中更改命令表组态 (页 75)

使用趋势图

趋势图和组件



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

1.6 工艺对象命令表

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：

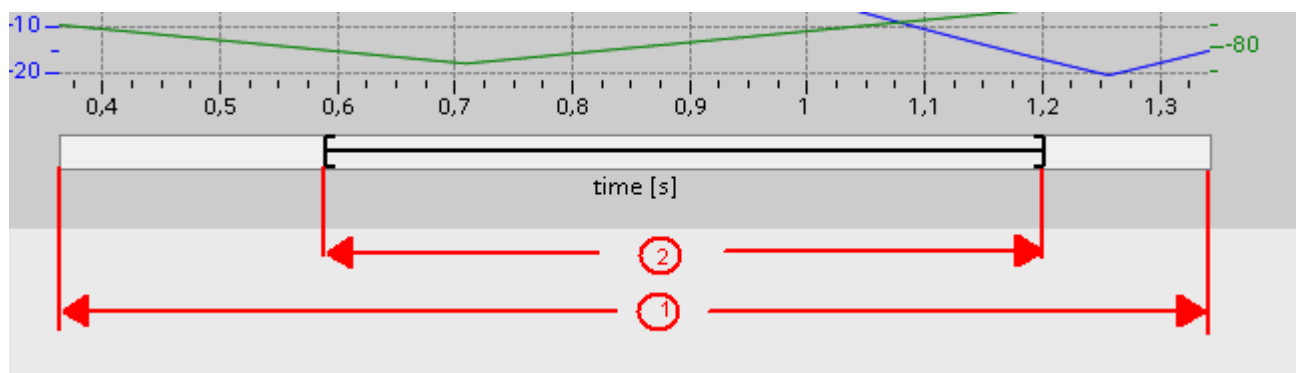
缩放轴，以便位置和速度曲线可见。

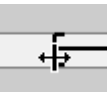
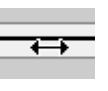
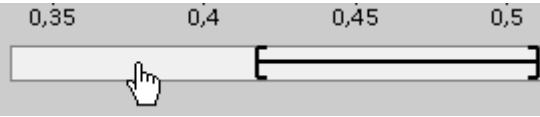
- 缩放到曲线和限值：

缩放轴，以便位置和速度曲线、激活软件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：

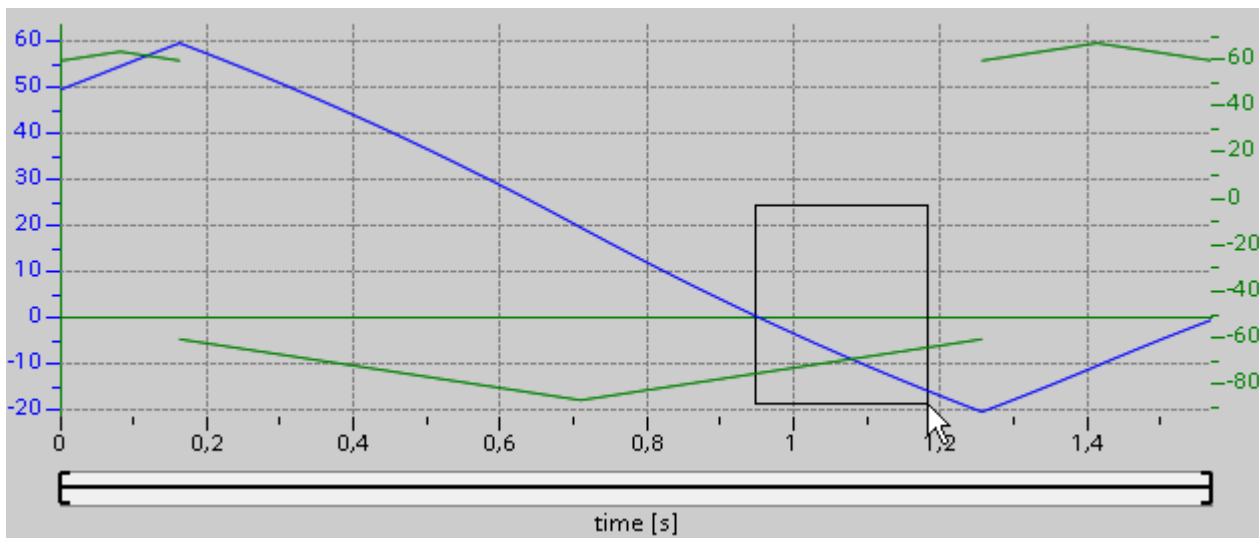


①	曲线值和/或限值所在的范围。（请参见“在快捷菜单中选择”）
②	<p>要在趋势窗口中显示的所选范围。</p> <p>利用空白光标可以设置右边和左边的空白范围。</p>  <p>通过拖动光标在范围 ① 内设置位置。</p>  <p>还可以通过范围 ① 内单击来定义位置。</p> 

1.6 工艺对象命令表

利用鼠标选择要显示的部分：

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



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 60)
- 组态 - 命令表 (页 60)
- 快捷菜单命令 - 命令表 (页 65)
- 快捷菜单命令 - 曲线图 (页 71)
- 从“完成命令”过渡到“混合运动” (页 73)
- 在用户程序中更改命令表组态 (页 75)

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。

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、激活软件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。

显示软件限位开关

显示软件限位开关的行。

1.6 工艺对象命令表

显示比例尺

使比例尺渐显/渐隐

在要查看曲线的各个值时可使用比例尺。

参见

组态 - 常规 (页 60)

组态 - 命令表 (页 60)

快捷菜单命令 - 命令表 (页 65)

使用趋势图 (页 67)

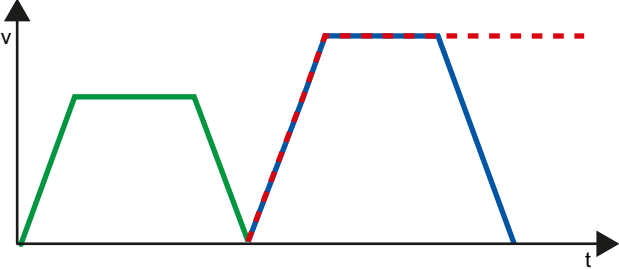
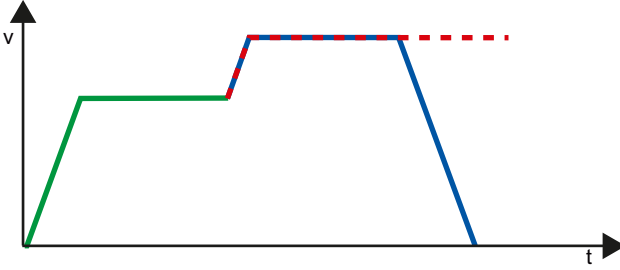
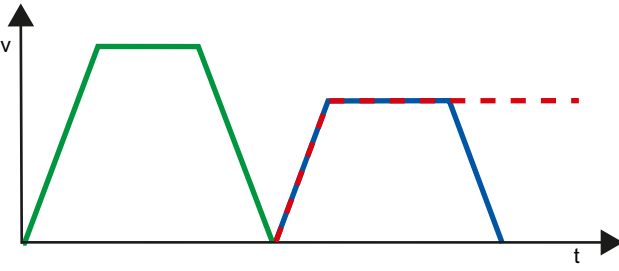
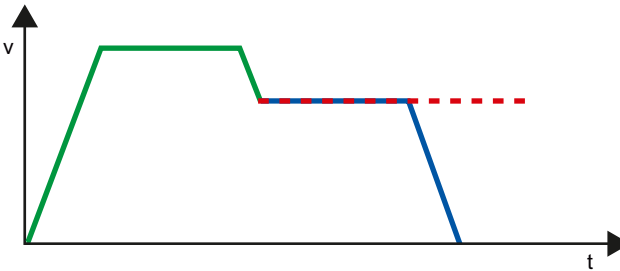
从“完成命令”过渡到“混合运动” (页 73)




在用户程序中更改命令表组态 (页 75)

从“完成命令”过渡到“混合运动”

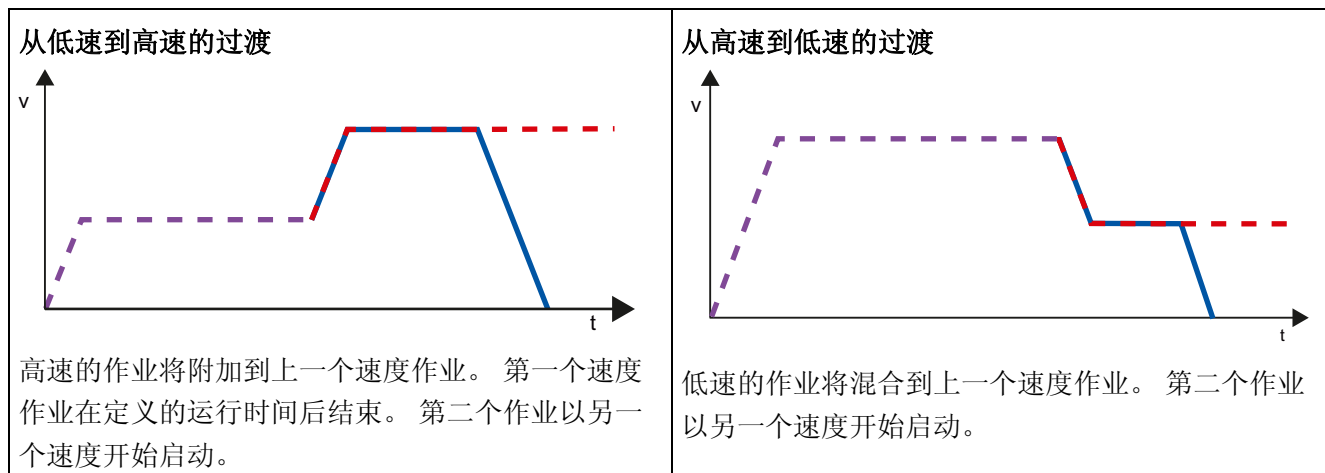
下图在“下一步”列中显示了不同转换模式移动之间的转换：

具有先前定位作业的运动过渡

完整作业	混合运动
<p data-bbox="161 645 416 678">从低速到高速的过渡</p>  <p data-bbox="161 981 783 1093">高速作业会附加到上一定位作业。定位作业在其目标位置以速度“0”终止。第二个作业从静止开始启动。</p>	<p data-bbox="825 645 1080 678">从低速到高速的过渡</p>  <p data-bbox="825 981 1447 1093">高速作业会与上一定位作业重叠。第一个定位作业在其目标位置以非静止速度终止。第二个作业以另一个速度开始启动。</p>
<p data-bbox="161 1120 416 1153">从高速到低速的过渡</p>  <p data-bbox="161 1451 783 1563">低速作业会附加到上一定位作业。定位作业在其目标位置以速度“0”终止。第二个作业从静止开始启动。</p>	<p data-bbox="825 1120 1080 1153">从高速到低速的过渡</p>  <p data-bbox="825 1451 1447 1563">低速作业会与上一定位作业重叠。第一个定位作业在其目标位置以非静止速度终止。第一个作业以另一个速度开始启动。</p>

	1. 作业“Positioning Relative”或“Positioning Absolute”
	2. 作业“Velocity set point”
	2. 作业“Positioning Relative”或“Positioning Absolute”

具有先前速度作业的运动过渡



- - -	1. 作业“Velocity set point”
- - -	2. 作业“Velocity set point”
—	2. 作业“Positioning Relative”或“Positioning Absolute”

参见

- 组态 - 常规 (页 60)
- 组态 - 命令表 (页 60)
- 快捷菜单命令 - 命令表 (页 65)
- 使用趋势图 (页 67)
- 快捷菜单命令 - 曲线图 (页 71)
- 在用户程序中更改命令表组态 (页 75)

在用户程序中更改命令表组态

可以在用户程序在 CPU 中运行期间更改下列组态参数：

作业和对应值

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>.Config.Commands[1..32].Command`
用于更改命令类型
- `<表名称>.Config.Commands[1..32].Position`
用于更改位置/行进路径
- `<表名称>.Config.Commands[1..32].Velocity`
用于更改速度
- `<表名称>.Config.Commands[1..32].Duration`
用于更改持续时间
- `<表名称>.Config.Commands[1..32].BufferMode`
用于更改参数“下一步”
- `<表名称>.Config.Commands[1..32].Code`
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说明。

参见

组态 - 常规 (页 60)

组态 - 命令表 (页 60)

快捷菜单命令 - 命令表 (页 65)

使用趋势图 (页 67)

快捷菜单命令 - 曲线图 (页 71)

从“完成命令”过渡到“混合运动” (页 73)

1.6 工艺对象命令表

1.6.4.3 扩展参数

图表参数

组态 - 常规

在“常规”(General) 组态窗口中组态“命令表”工艺对象图表视图的基本属性。

说明

如果已在“使用轴参数”(Use axis parameters of) 下选择默认轴，则可以编辑测量单位。
如果已选择组态轴，将会显示该轴的测量单位。

使用轴参数

从下拉列表中，选择使用哪些轴参数来选择图形视图以及检查移动序列。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)。

“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。

计量单位

在此字段中输入默认轴的计量单位。如果已在“使用轴参数”(Use axis parameters of) 中选择预组态的轴，将会显示这些参数中组态的计量单位。

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冲击限制。

说明

如果已在“使用轴参数”(Use axis parameters of) 下选择默认轴，则可以编辑下列字段。
如果已选择组态的轴，将会显示该轴的值。

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:

- $1.0e-12 \leq \text{加速度} \leq 1.0e12$
- $1.0e-12 \leq \text{减速度} \leq 1.0e12$

激活冲击限制

使用该复选框可激活冲击限制。

步进

在“步进”(Step) 字段中可设置加速和减速的所需步进。

将会用所选步进计算命令表中组态的运动作业。

限值:

- $1.0e-12 \leq \text{冲击} \leq 1.0e12$

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件限位开关。

说明

如果已在“使用轴参数”(Use axis parameters of) 下选择默认轴，则可以编辑下列字段。
如果已选择组态的轴，将会显示该轴的值。

最大速度/启动/停止速度

在这些字段中可以定义默认轴的最大允许速度以及启动/停止速度。启动/停止速度是默认轴的最小允许速度。

限值:

- $1.0e-12 \leq \text{启动/停止速度} \leq 1.0e12$

启动/停止速度 = 0.0

- $1.0e-12 \leq \text{最大速度} \leq 1.0e12$

最大速度 = 0.0

最大速度值必须大于等于启动/停止速度值。

启用软件限位开关

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

下限/上限软件限位开关

在这些字段中输入下限和上限软件限位开关的位置值。

限值:

- $-1.0e12 \leq \text{下限软件限位开关} \leq -1.0e-12$

$1.0e-12 \leq \text{下限软件限位开关} \leq 1.0e12$

下限软件限位开关 = 0.0

- $-1.0e12 \leq \text{上限软件限位开关} \leq -1.0e-12$

$1.0e-12 \leq \text{上限软件限位开关} \leq 1.0e12$

上限软件限位开关 = 0.0

上限软件限位开关的值必须大于等于下限软件限位开关的值。


1.7 下载到 CPU

加载到 CPU S7-1200 时，始终确保项目文件经过在线和离线加载后保持一致。无法加载单个块。选择单个块时，将始终加载所有新的和已修改的块。

可将以下对象组加载到 CPU 中：

上下文菜单命令 “下载到设备”	说明
全部	下载所有新的和已修改的块以及新的或已修改的硬件配置
硬件配置	下载新的或已修改的硬件配置
软件	下载所有新的和已修改的块
软件（所有块）	下载所有块

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
<p>不使用硬件配置加载时轴的可能故障</p> <p>对轴组态进行以下修改时，会修改硬件配置：</p> <ul style="list-style-type: none"> • 修改脉冲发生器 (PTO) • 修改硬件限位开关地址 • 修改参考点开关地址 <p>如果使用上下文菜单命令“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。</p> <p>确保在下列条件下将当前硬件配置下载到 CPU 中。</p>

在 CPU S7-1200 RUN 操作模式下下载（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 操作模式下加载时，会检查是否可在不停止 CPU 的情况下进行加载。

在 RUN 操作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	是	否
数据块修改的结构	否	否
新数据块	是	是
删除的数据块	是	是

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 操作模式下下载。

V3.0 之前的工艺对象无法在 CPU RUN 操作模式下下载。

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：

- **工艺对象轴和命令表**
将 CPU 操作模式从 STOP 切换到 RUN。
- **工艺对象轴**
禁用轴并使用运动控制指令“MC_Reset”执行“重新启动”。
- **工艺对象命令表**
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。

说明

与 STOP 操作模式下的下载相反，在 RUN 操作模式下不会覆盖实际参数。仅当操作模式下一次从 STOP 切换到 RUN 时，对实际参数进行的修改才会生效。

参见

运动控制使用指南 (页 22)

MC_Reset: 确认错误 (页 182)

1.8 调试轴 - 轴控制面板

轴命令表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，才能使用轴控制表。

注意

轴控制面板的响应时间

轴控制表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。

“手动控制”按钮

单击“手动控制”(Manual control) 可在手动控制模式下移动轴。首先在用户程序中使用运动控制指令“MC_Power”禁用轴。在“手动控制”模式下，轴控制表对轴功能具有优先控制权。在结束手动控制前，用户程序不能影响轴功能。

警告

只能为一根轴激活手动控制。可以在自动模式下移动第二根轴，但这可能引起危险。在这种情况下，将第二根轴设置为不运行。

“自动模式”按钮

单击“自动模式”(Automatic mode)，可结束“手动控制”模式。轴控制表交回优先控制权，然后轴可再次由用户程序控制。必须在用户程序中重新启用轴。需要时，可使其回原点。

请在切换到自动控制前完成所有激活的行进运动；否则，将以急停减速度对轴进行制动。

“启用”按钮

单击“启用”(Enable) 可在“手动控制”模式下启用轴。轴启用时，可以使用轴控制面板功能。

如果因为没有满足某些条件而无法启用轴，则请注意“错误消息”(Error message) 字段中的错误消息。有关清除错误的信息，请参见附录中的“ErrorID 和 ErrorInfo 列表”。纠正错误后，再次启用轴。

“禁用”按钮

如果要在“手动控制”模式下临时禁用轴，则单击“禁用”(Disable)。

“命令”区域

仅在轴启用后才可以执行“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作业“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字段和命令启动按钮。

“轴状态”区域

如果激活“手动控制”模式，则“轴状态”(Axis status) 区域中将显示当前的轴状态和驱动器状态。“过程值”(Process values) 中显示轴的当前轴位置和速度。

单击“确认”(Acknowledge) 可确认所有清除的错误。

“信息消息”(Info message) 字段会显示有关轴的状态的高级信息。

错误消息

“错误消息”(Error message) 字段显示当前错误。在“手动控制”模式下，清除错误后，可通过按下“确认”(Acknowledge) 按钮来删除错误条目。

说明

速度、加速度/减速度和冲击的初始值

出于安全考虑，轴命令表激活时，仅使用等于组态值 10% 的值来初始化“速度”(Velocity)、“加速度/减速度”(Acceleration/deceleration) 和“冲击”(Jerk) 参数。“冲击”(Jerk) 参数仅用于 V2.0 及更高版本的“轴”工艺对象。

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

控制面板中的“速度”(Velocity) 参数来自组态中的“最大速度”(Maximum velocity)，“加速度/减速度”(Acceleration/deceleration) 参数则来自组态中的“加速度”(Acceleration)。

在轴命令表中，可以更改“速度”(Velocity)、“加速度/减速度”(Acceleration/deceleration) 和“冲击”(Jerk) 参数；这不会影响组态中的值。

参见

运动控制使用指南 (页 22)

使用监视表格 (页 111)

1.9 编程

1.9.1 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作业。

可以从运动控制指令的输出参数中获取运动控制作业的状态及作业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 (页 177)
- MC_Reset: 确认错误 (页 182)
- MC_Home: 归位轴，设置归位位置 (页 185)
- MC_Halt: 停止轴 (页 189)
- MC_MoveAbsolute: 轴的绝对定位 (页 193)
- MC_MoveRelative: 轴的相对定位 (页 198)
- MC_MoveVelocity: 以预设的旋转速度移动轴 (页 203)
- MC_MoveJog: 在点动模式下移动轴 (页 208)
- MC_CommandTable: 按移动顺序运行轴作业（从 V2.0“轴”工艺对象起） (页 212)
- MC_ChangeDynamic: 更改轴的动态设置（从 V2.0“轴”工艺对象起） (页 216)

参见

创建用户程序 (页 85)

编程注意事项 (页 88)

断电和重新启动后运动控制命令的行为 (页 90)

监视激活的命令 (页 91)

运动控制语句的错误显示 (页 104)

1.9.2 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：

1. 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。

2. 打开“工艺”(Technology) 类别以及“运动控制”(Motion Control) 和“S7-1200 运动控制”(S7-1200 Motion Control) 文件夹。

3. 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。

4. 在下一个对话框中，选择其中一个选项：

单背景

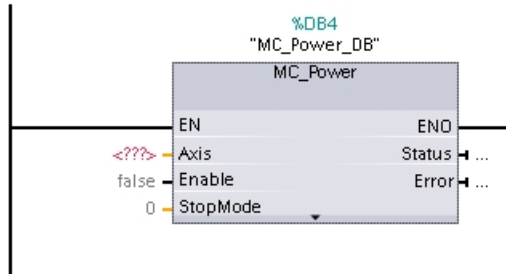
单击“单背景”(Single instance) 并选择是要自动还是手动定义背景数据块的名称和编号。

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)。

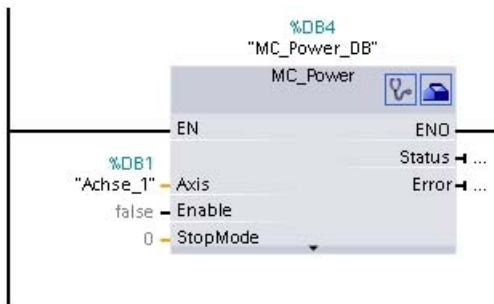
运动控制指令“MC_Power”将插入到该程序段中。



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。

6. 在项目树中选择工艺对象并将其拖放到 <???”上。



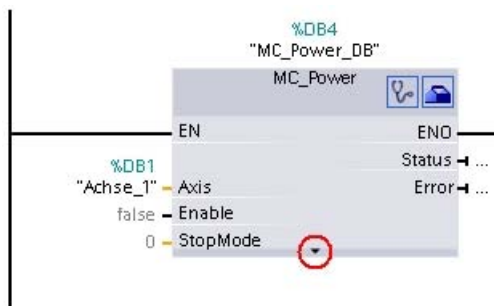
选择工艺对象数据块后，以下按钮将可用：



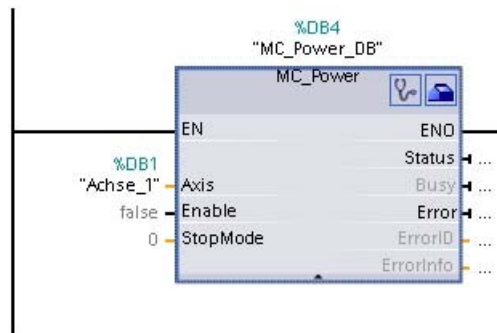
如果想要打开工艺对象的诊断对话框，则单击听诊器图标。



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详细信息，请参见详细说明。

参见

运动控制语句概述 (页 84)

编程注意事项 (页 88)

断电和重新启动后运动控制命令的行为 (页 90)

监视激活的命令 (页 91)

运动控制语句的错误显示 (页 104)

1.9.3 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。“Enable”= TRUE、“JogForward”和“JogBackward”时，也应用输入参数中的更改。

- 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 <轴名称>.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

使轴回原点后，才可启动 MC_MoveAbsolute 命令。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，将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。

- **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 117)”。

参见

运动控制语句概述 (页 84)

创建用户程序 (页 85)

断电和重新启动后运动控制命令的行为 (页 90)

监视激活的命令 (页 91)

运动控制语句的错误显示 (页 104)

在更高优先级等级（执行级别）中跟踪作业 (页 117)

1.9.4 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。

参见

运动控制语句概述 (页 84)

创建用户程序 (页 85)

编程注意事项 (页 88)

监视激活的命令 (页 91)

运动控制语句的错误显示 (页 104)

1.9.5 监视激活的命令

1.9.5.1 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作业：

- 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”

1.9.5.2 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）。作业完成，轴处于停止状态。

以下运动控制指令的作业具有明确的结论：

- MC_Reset
- MC_Home
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_CommandTable（从 V2.0“轴”工艺对象起）
- MC_ChangeDynamic（从 V2.0“轴”工艺对象起）

如果作业已成功完成，则输出参数“Done”的值为 TRUE。

输出参数“Busy”、“CommandAborted”和“Error”分别指示作业仍在处理、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。

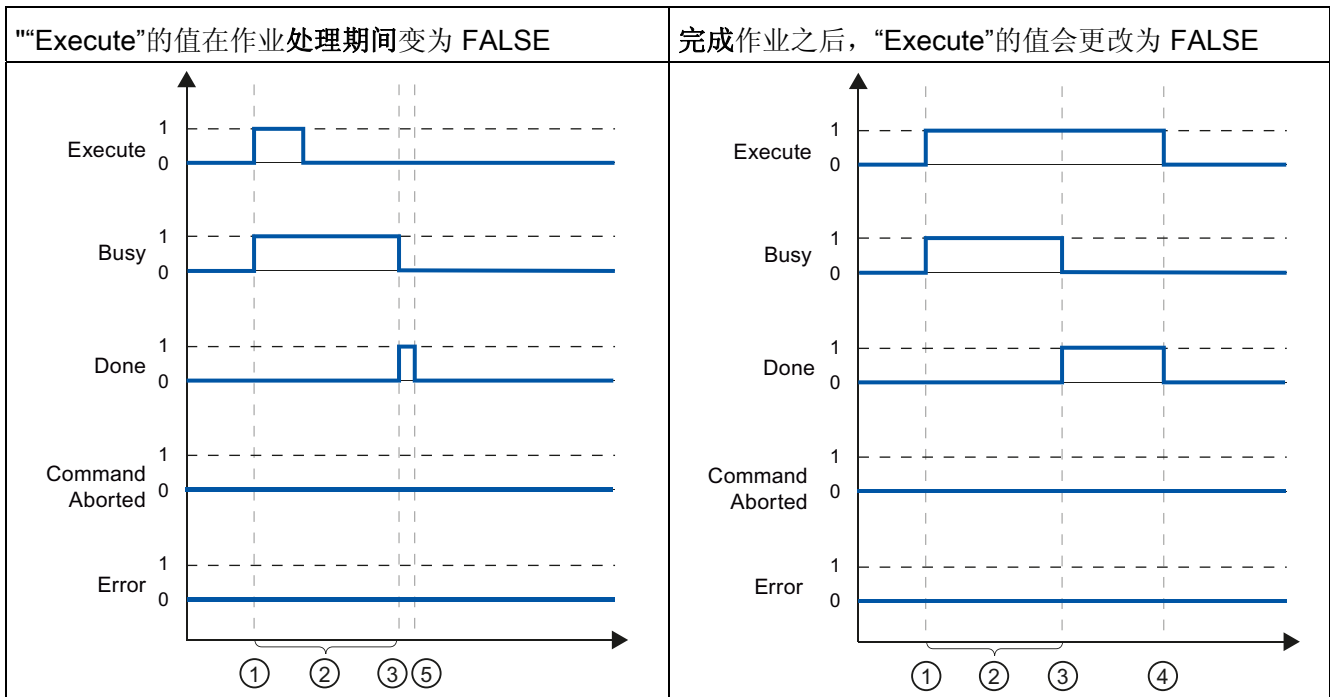
在运动控制作业处理期间，输出参数“Busy”的值为 TRUE。如果作业已完成、中止或因错误停止，则输出参数“Busy”的值将变为 FALSE。无论输入参数“Execute”的信号状态是什么，都会发生这种变化。

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：

作业完全执行

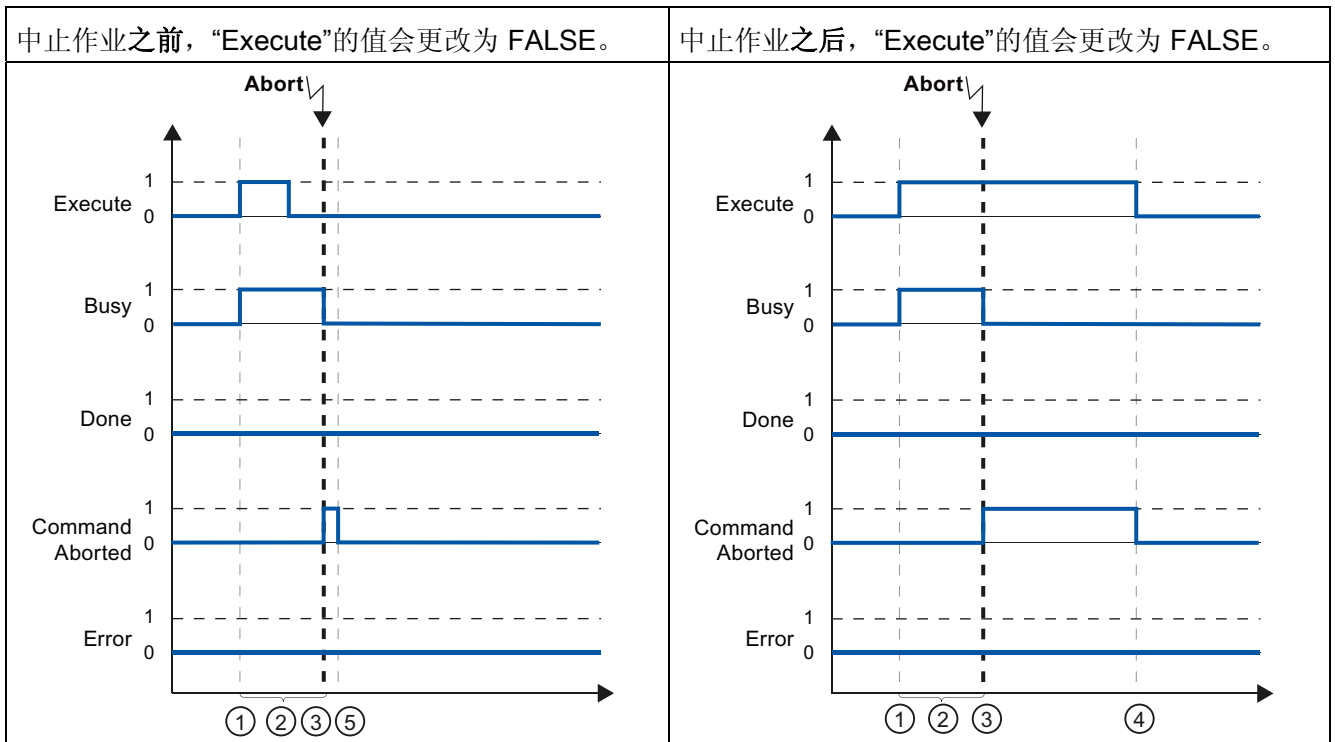
如果运动控制作业已在对其下结论前完全执行，则将通过输出参数“Done”的值为 TRUE 的形式对此进行指示。输入参数“Execute”的信号状态影响输出参数“Done”中的显示持续时间：



①	输入参数“Execute”的上升沿时启动作业。根据编程情况，“Execute”在作业执行期间仍然可能被重置为值 FALSE，或者保持为值 TRUE，直到作业完成为止。
②	作业激活期间，输出参数“Busy”的值为 TRUE。
③	作业结束后（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”的值在作业完成之前保持为 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”在作业完成之前设置为 FALSE，则“Done”的值仅在一个执行周期内为 TRUE。

中止作业

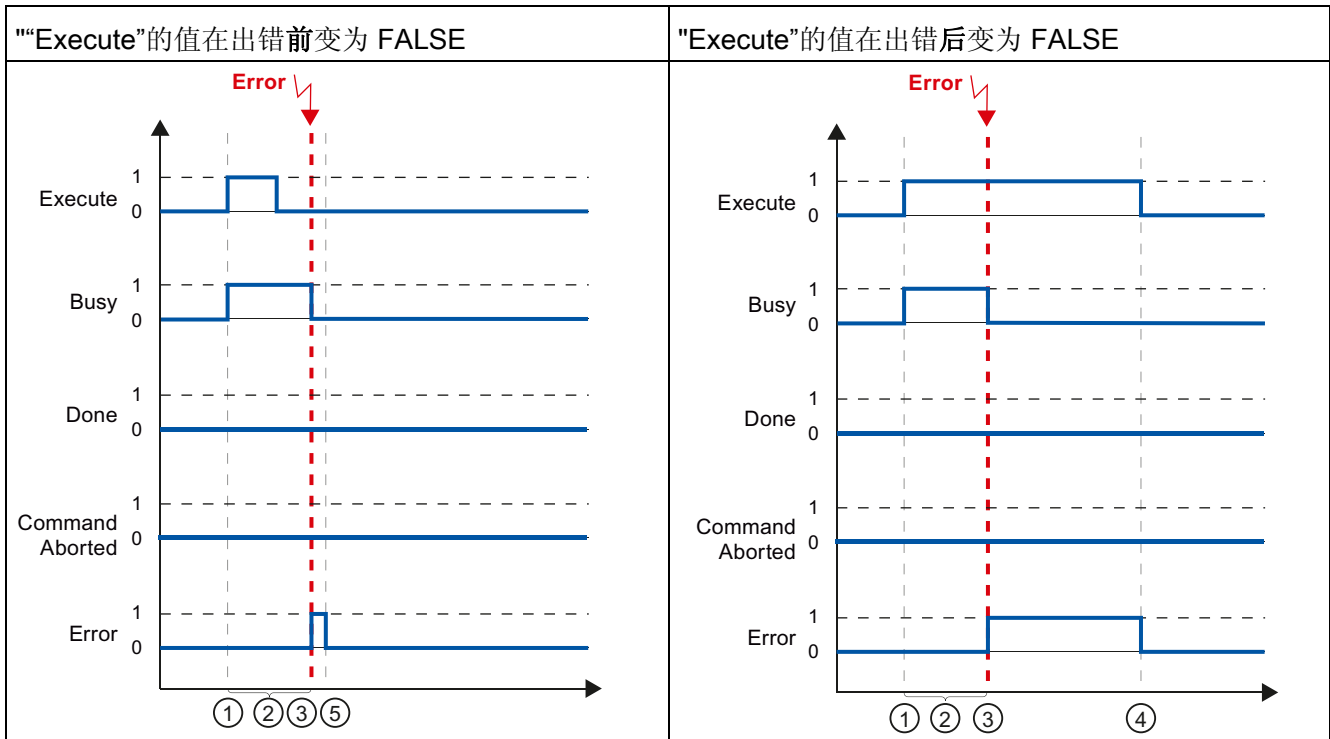
如果运动控制作业在执行期间中止，则将通过输出参数“CommandAborted”的值为 TRUE 的方式对此进行指示。输入参数“Execute”的信号状态影响输出参数“CommandAborted”中的显示持续时间：



①	输入参数“Execute”的上升沿时启动作业。根据编程情况，“Execute”在作业执行期间仍然可能被重置为值 FALSE，或者保持为值 TRUE，直到作业完成为止。
②	作业激活期间，输出参数“Busy”的值为 TRUE。
③	作业执行期间，作业被其它运动控制作业中止。如果作业中止，输出参数“Busy”的值将变为 FALSE，“CommandAborted”的值将变为 TRUE。
④	如果“Execute”的值在作业中止之前保持为 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”在作业中止之前设置为 FALSE，则“CommandAborted”的值仅在一个执行周期内为 TRUE。

作业执行期间出错

如果在运动控制作业执行期间出错，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”的上升沿时启动作业。根据编程情况，“Execute”在作业执行期间仍然可能被重置为值 FALSE，或者保持为值 TRUE，直到作业完成为止。
②	作业激活期间，输出参数“Busy”的值为 TRUE。
③	作业执行期间出错。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。

1.9.5.3 运动控制指令 MC_MoveVelocity

运动控制指令“MC_MoveVelocity”的作业的结束没有明确的定义。第一次达到设置的速度且轴恒速运转时，就实现了作业目标。如果达到设置的速度，则通过输出参数“InVelocity”的 TURE 值对此进行指示。

如果已达到设置的速度且输入参数“Execute”的值设置为 FALSE，则作业完成。但是，作业完成时轴运动尚未完成。例如，可以使用运动控制作业“MC_Halt”停止轴运动。

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

运动控制作业执行期间，输出参数“Busy”的值为 TRUE。如果作业已完成、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。

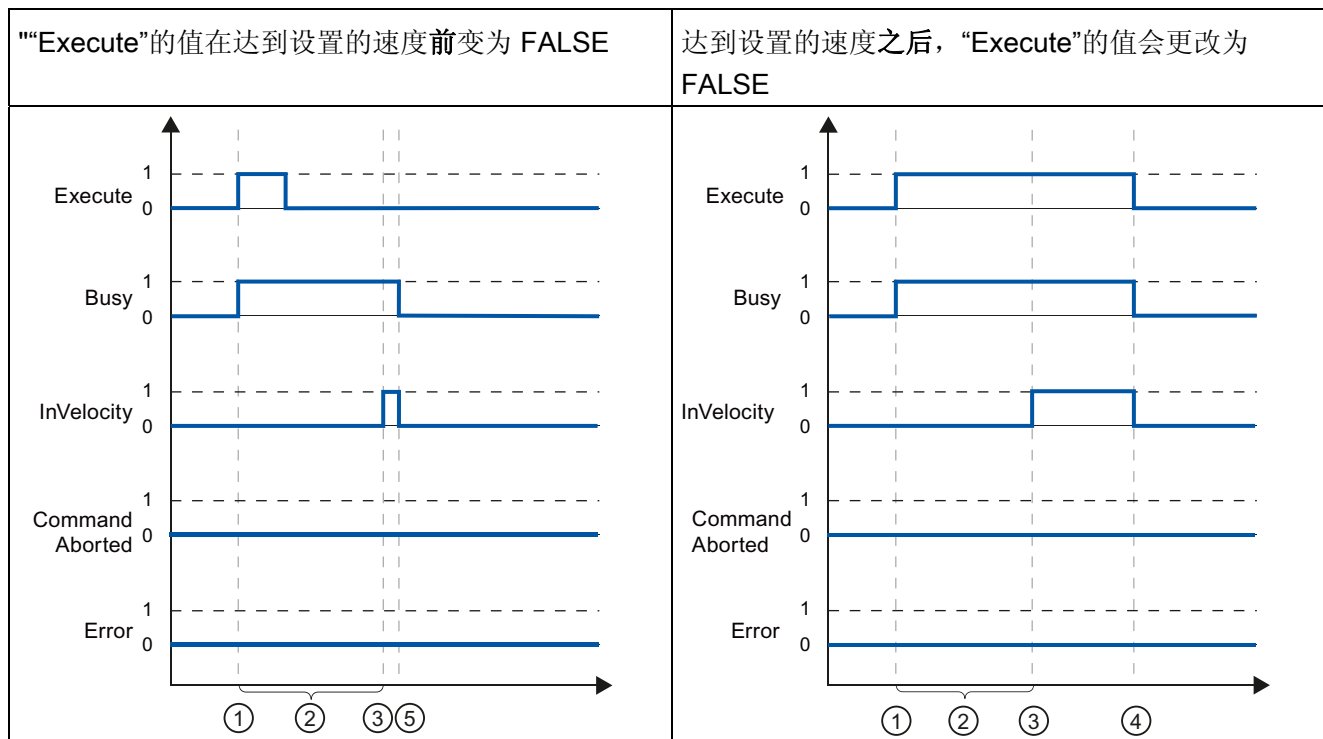
当输出参数“InVelocity”、“CommandAborted”和“Error”的条件满足时，它们的值至少在一个周期内都为 TRUE。当输入参数“Execute”设置为 TRUE 时，将锁存这些状态消息。

下图针对各种示例情况显示了状态位的操作：

达到设置的速度

如果运动控制作业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。

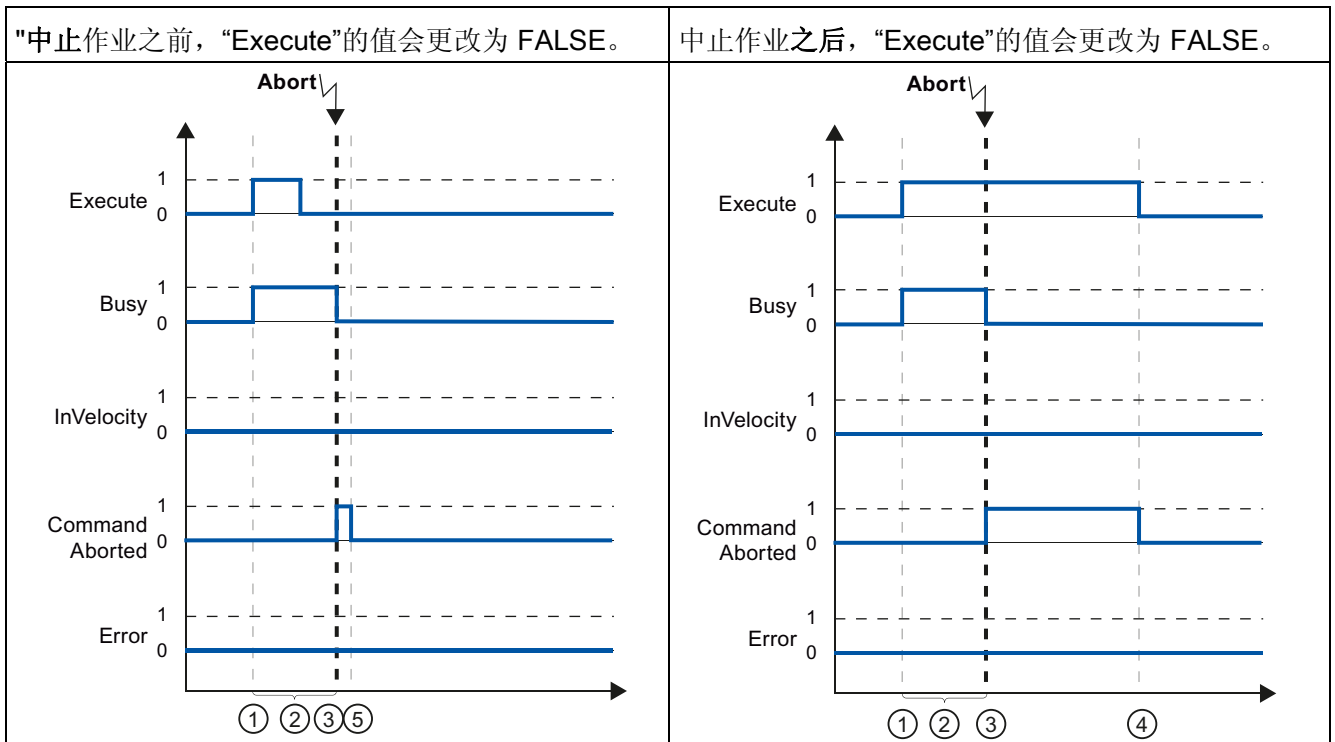
输入参数“**Execute**”的信号状态影响输出参数“**InVelocity**”中的显示持续时间：



①	输入参数“ Execute ”的上升沿时启动作业。根据编程情况，“ Execute ”的值可在达到设置的速度前重置为 FALSE ，也可在达到设置的速度后重置为 FALSE 。
②	作业激活期间，输出参数“ Busy ”的值为 TRUE 。
③	达到设置的速度时，输出参数“ InVelocity ”的值变为 TRUE 。
④	如果“ Execute ”的值在达到设置的速度后仍保持为 TRUE ，则作业将保持激活状态。“ InVelocity ”和“ Busy ”的值保持为 TRUE 并且其状态仅随“ Execute ”一起变为 FALSE 。
⑤	如果“ Execute ”在达到设置的速度前重置为 FALSE ，则作业将在达到设置的速度时完成。“ InVelocity ”的值仅在一个执行周期内为 TRUE ，并且随“ Busy ”一起变为 FALSE 。

作业在达到设置的速度前中止

如果运动控制作业在达到设置的速度前中止，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”的上升沿时启动作业。根据编程情况，“Execute”在作业执行期间仍然可能被重置为值 FALSE，或者保持为值 TRUE，直到作业中止为止。
②	作业激活期间，输出参数“Busy”的值为 TRUE。
③	作业执行期间，作业被其它运动控制作业中止。如果作业中止，输出参数“Busy”的值将变为 FALSE，“CommandAborted”的值将变为 TRUE。
④	如果“Execute”的值在作业中止之前保持为 TRUE，则“CommandAborted”的值也将保持为 TRUE 并且其状态随“Execute”一起变为 FALSE。
⑤	如果“Execute”在作业中止之前重置为 FALSE，则“CommandAborted”的值仅在一个执行周期内为 TRUE。

说明

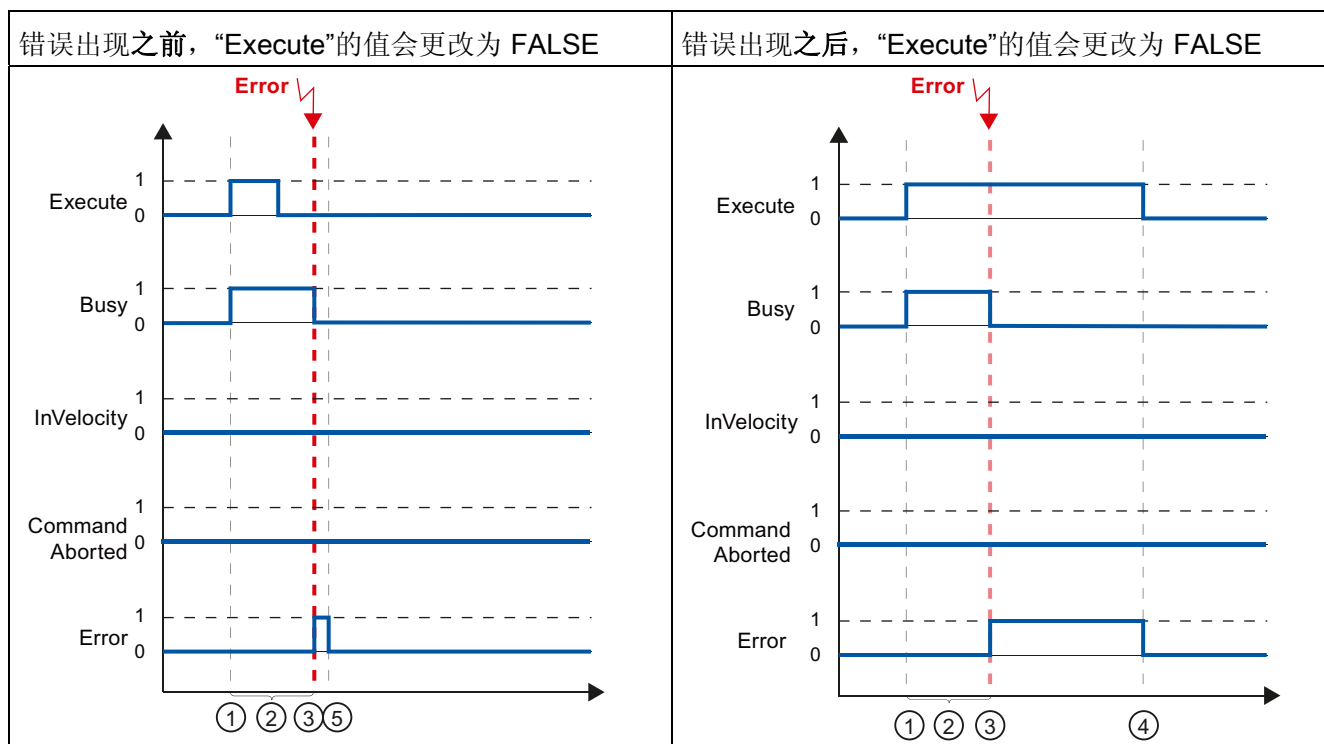
在以下条件下，输出参数“CommandAborted”不指示出现中止：

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作业。

达到设置的速度并且输入参数“Execute”的值为 FALSE 时，作业完成。因此，不会将新作业的启动指示为中止。

达到设置速度前出错

如果在运动控制作业执行期间，达到设置的速度之前出错，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”的上升沿时启动作业。根据编程情况，“Execute”在作业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	作业激活期间，输出参数“Busy”的值为 TRUE。
③	作业执行期间出错。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，逼近软件限位开关）。

达到设置的速度并且输入参数“Execute”的值为 FALSE 时，作业完成。作业完成后，仅在运动控制指令“MC_Power”中指示轴错误。

1.9.5.4 运动控制指令 MC_MoveJog

运动控制指令“MC_MoveJog”的作业实现点动操作。

运动控制作业“MC_MoveJog”的结束没有明确的定义。第一次达到设置的速度且轴恒速运转时，就实现了作业目标。如果达到设置的速度，则通过输出参数“InVelocity”的 TURE 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”分别指示作业仍在处理、已中止或有未决的错误。

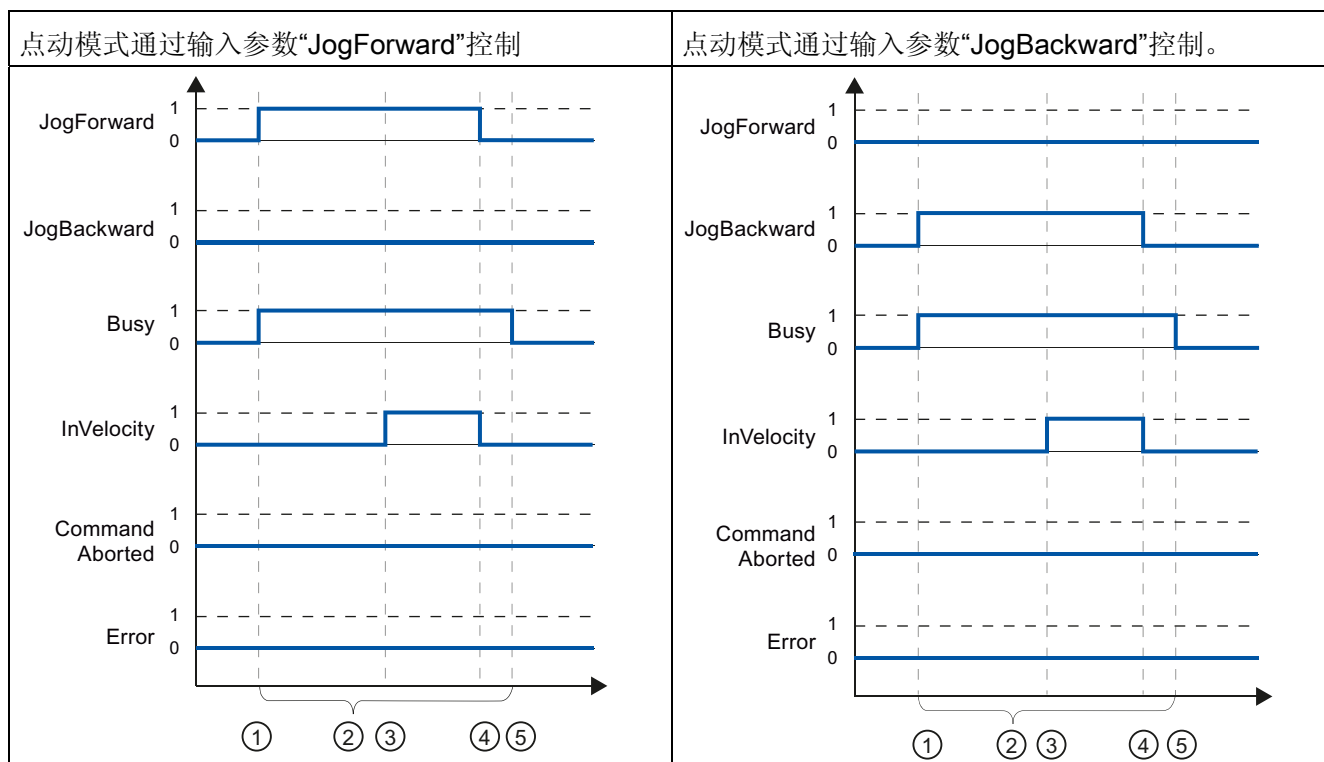
在运动控制作业处理期间，输出参数“Busy”的值为 TRUE。如果作业已完成、中止或因错误停止，则输出参数“Busy”的值将变为 FALSE。

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

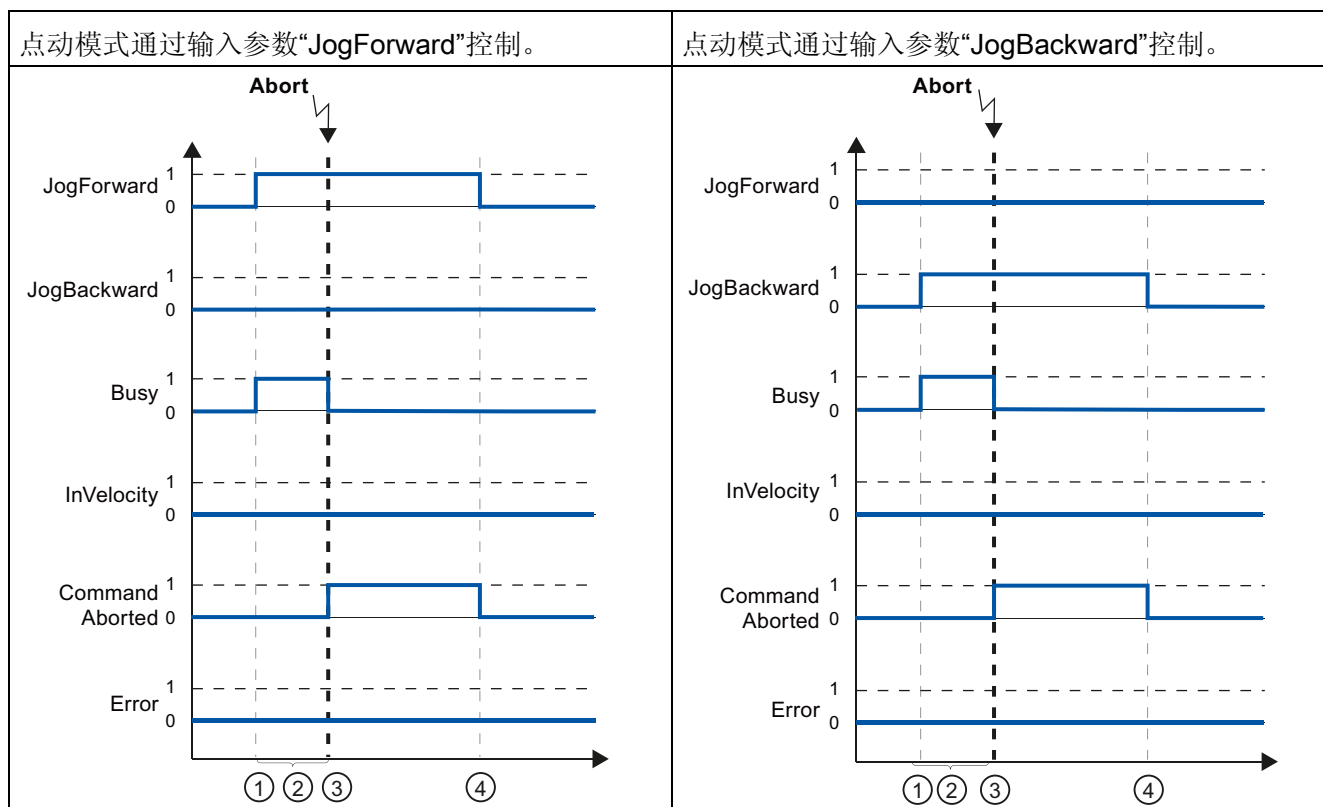
如果运动控制作业在达到设置的速度前执行完毕，则将通过输出参数“**InVelocity**”的值为 **TURE** 的方式对此进行指示。



①	输入参数“ JogForward ”或“ JogBackward ”的上升沿时启动作业。
②	作业激活期间，输出参数“ Busy ”的值为 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，则运动控制作业完成并且输出参数“ Busy ”的值将变为 FALSE 。

作业在执行期间中止

如果运动控制作业在执行期间中止，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”的上升沿时启动作业。
②	作业激活期间，输出参数“Busy”的值为 TRUE。
③	作业执行期间，作业被其它运动控制作业中止。如果作业中止，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。

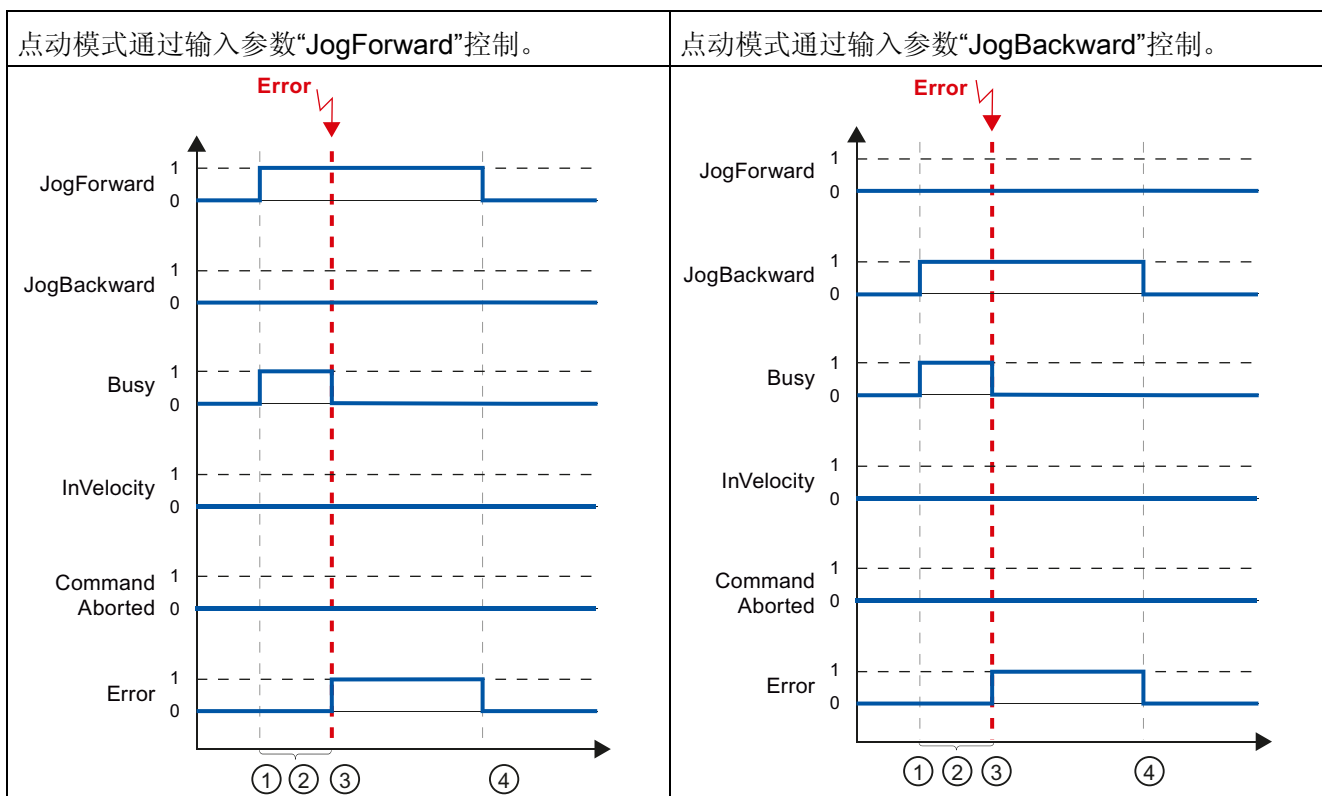
说明

如果满足以下所有条件，则将在输出参数“CommandAborted”中指示作业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），并且新的运动控制作业已启动。

作业执行期间出错

如果在运动控制作业执行期间出错，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”的上升沿时启动作业。
②	作业激活期间，输出参数“Busy”的值为 TRUE。
③	作业执行期间出错。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，逼近软件限位开关）。

1.9.6 运动控制语句的错误显示

运动控制指令在运动控制指令的输出参数 "Error"、"ErrorID" 和 "ErrorInfo" 中指示任何运动控制作业和工艺对象错误。

输出参数 "Error", "ErrorID" 和 "ErrorInfo" 中的错误显示

如果输出参数 "Error" 的值为 TRUE，则无法执行整个作业或其中各个部分。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，“已逼近硬件限位开关”）**

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。

- **运动控制指令的参数化错误
（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。

参见

运动控制语句概述 (页 84)

创建用户程序 (页 85)

编程注意事项 (页 88)

断电和重新启动后运动控制命令的行为 (页 90)

监视激活的命令 (页 91)

1.10 轴 - 诊断

1.10.1 状态和错误位

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	轴已启用且准备好接受运动控制作业控制。 (工艺对象的变量: <轴名称>.StatusBits.Enable)
已回原点	轴已回原点且能够执行运动控制指令“MC_MoveAbsolute”的绝对定位作业。对于相对回原点而言，轴不必回原点。特殊情况： <ul style="list-style-type: none"> 主动回原点期间，该状态为 FALSE。 如果回原点的轴经受被动回原点，则在被动回原点期间该状态设置为 TURE。 (工艺对象的变量: <轴名称>.StatusBits.HomingDone)
轴错误	“轴”工艺对象发生错误。有关错误的更多信息，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，轴命令表的“错误消息”(Error message) 字段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)
轴命令表启用	在轴命令表中启用了“手动控制”模式。轴命令表对“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	驱动器准备好运行。 (工艺对象的变量: <轴名称>.StatusBits.DriveReady)
驱动器错误	驱动器在其“驱动器准备就绪”信号出现故障后报告了错误。 (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	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	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	轴在执行运动控制指令“MC_MoveAbsolute”或“MC_MoveRelative”或者轴命令表的定位作业。 (工艺对象的变量: <轴名称>.StatusBits.PositioningCommand)
以速度设定值行进	轴正在以运动控制指令“MC_MoveVelocity”或“MC_MoveJog”或者轴命令表的速度设定值执行定位作业。 (工艺对象的变量: <轴名称>.StatusBits.SpeedCommand)

1.10 轴 - 诊断

状态	说明
回原点	轴在执行运动控制指令“MC_Home”或轴命令表的回原点作业。 (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
已到达下限软件限位开关	已到达下限软件限位开关。 (工艺对象的变量: <轴名称>.ErrorBits.SwLimitMinReached)
已超出下限软件限位开关	已超出下限软件限位开关。 (工艺对象的变量: <轴名称>.ErrorBits.SwLimitMinExceeded)
已到达上限软件限位开关	已到达上限软件限位开关。 (工艺对象的变量: <轴名称>.ErrorBits.SwLimitMaxReached)
已超出上限软件限位开关	已超出上限软件限位开关。 (工艺对象的变量: <轴名称>.ErrorBits.SwLimitMaxExceeded)
已逼近下限硬件限位开关	已逼近下限硬件限位开关。 (工艺对象的变量: <轴名称>.ErrorBits.HwLimitMin)
已逼近上限硬件限位开关	已逼近上限硬件限位开关。 (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

运动状态 (页 109)

StatusBits. 变量 (页 156)

ErrorBits. 变量 (页 160)

1.10.2 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
位置	“位置”(Position) 字段指示轴的当前位置。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>.MotionStatus.Position)
速度	“速度”(Velocity) 字段指示轴的当前速度。 (工艺对象的变量: <轴名称>.MotionStatus.Velocity)
目标位置	“目标位置”(Target position) 字段指示活动定位作业或轴命令表的当前目标位置。 “目标位置”(Target position) 的值仅在定位作业执行期间有效。 (工艺对象的变量: <轴名称>.MotionStatus.TargetPosition)
剩余行进距离	“剩余行进距离”(Remaining traversing distance) 字段指示活动定位作业或轴命令表的当前剩余行进距离。“剩余行进距离”(Remaining travasing distance) 的值仅在定位作业执行期间有效。 (工艺对象的变量: <轴名称>.MotionStatus.Distance)

参见

状态和错误位 (页 106)

MotionStatus. 变量 (页 155)

1.10.3 动态设置

在 TIA Portal 中使用“动态设置”(Dynamics settings) 诊断功能来监视轴的动态限值。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) 字段指示当前组态的轴的加速度。 (工艺对象变量: <轴名称>.Config.DynamicDefaults.Acceleration)
减速度	“减速度”(Deceleration) 字段指示当前组态的轴的减速度。 (工艺对象变量: <轴名称>.Config.DynamicDefaults.Deceleration)
急停减速度	“急停减速度”(Emergency stop deceleration) 字段指示当前组态的轴的急停减速度。 (工艺对象变量: <轴名称>.Config.DynamicDefaults.EmergencyDeceleration)
步进 (从 V2.0 轴工艺对象起)	“速度”(Velocity) 字段指示组态的当前轴的步进速度。 (工艺对象变量: <轴名称>.Config.DynamicDefaults.Jerk)

1.11 使用监视表格

如果要在调试期间监视和修改运动控制指令或“轴”工艺对象的变量，请使用监视表格。

要监视和修改变量，必须在监视表格中指定完整的变量名，包括对象名称和所有结构名称。

示例： <轴名称>.Config.DynamicDefaults.Acceleration

提示：

如果变量名较长，则可以使用复制和粘贴操作简化输入。

步骤

要插入变量名，请按以下步骤操作：

1. 在项目树中，选择背景数据块或轴工艺对象。

2. 运动控制指令的参数

- 右键单击，然后在快捷菜单中选择“打开”(Open) 命令。

工艺对象的变量

- 右键单击，然后在快捷菜单中选择“在编辑器中打开”(Open in editor) 命令。

3. 运动控制指令的参数

- 在 Input 或 Output 区域中选择变量所在的行

工艺对象的变量

- 在 Static 区域中，打开相关结构并选择变量所在的行

4. 选择“编辑 > 复制”(Edit > Copy) 菜单命令。

5. 双击打开监视表格。

6. 选择执行变量插入的起始行

7. 选择“编辑 > 粘贴”(Edit > Paste) 菜单命令。

在监视表格中使用完整名称插入变量。

1.11 使用监视表格



通过监视表格，还可对出于安全原因而禁止在用户程序中使用的变量进行写访问。修改这些变量可能导致当前轴组态数据损坏和未定义的轴响应。只能修改工艺对象的变量列表中标有“RW”访问权限的变量。

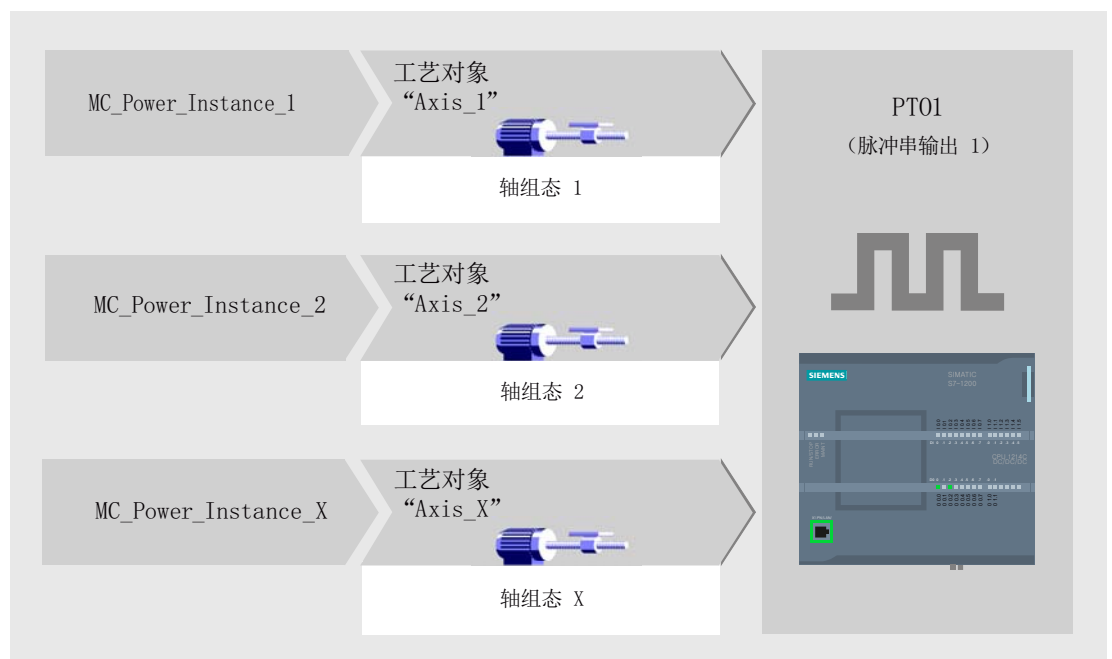
参见

调试轴 - 轴控制面板 (页 81)

1.12 附录

1.12.1 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一个 PTO (Pulse Train Output) 运行多个“轴”工艺对象，因而就可以使用一个 CPU 输出运行多个“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：

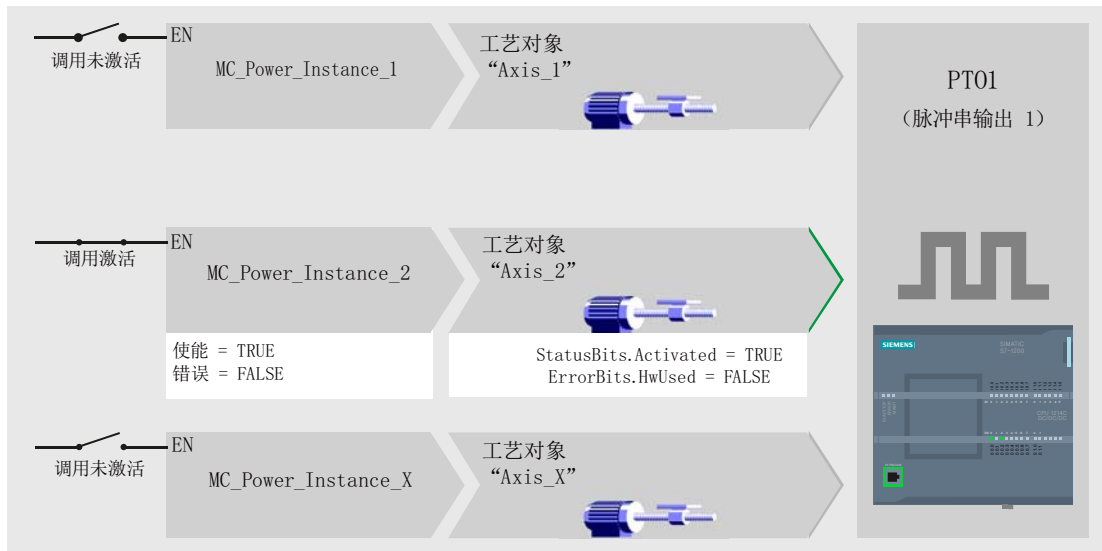


在本示例中，几个“轴”工艺对象（每个具有其自己的轴组态）使用同一个 PTO。必须在用户程序中通过单独调用使用独立背景数据块的运动控制指令“MC_Power”来调用各个“轴”。每次只能有一个“轴”使用 PTO。通过变量 <轴名称>.StatusBits.Activated = TRUE 指示当前使用 PTO 的轴。

切换“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（这里是“Axis_2”）的变量在用户程序中一般显示以下内容：

- <轴名称>.StatusBits.Activated = TRUE
- <轴名称>.ErrorBits.HwUsed = FALSE

要切换“轴”工艺对象，请按以下步骤操作。在本示例中，执行从“Axis_2”到“Axis_1”的切换：

1. 结束所激活“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“Axis_2”
3. 要检查“Axis_2”是否已禁用，请对运动控制指令“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“Axis_2”的运动控制指令的有条件调用。
5. 激活对“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“Axis_2”并激活“Axis_1”。
6. 使用运动控制指令“MC_Power”（输入参数 Enable = TRUE）启用“Axis_1”
7. 要检查“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（这里是“Axis_2”）时，该轴进入激活状态。

与有条件调用不同，禁用轴（这里是“Axis_1”和“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

将多个驱动器与同一个 PTO 一起使用 (页 116)

在更高优先级等级（执行级别）中跟踪作业 (页 117)

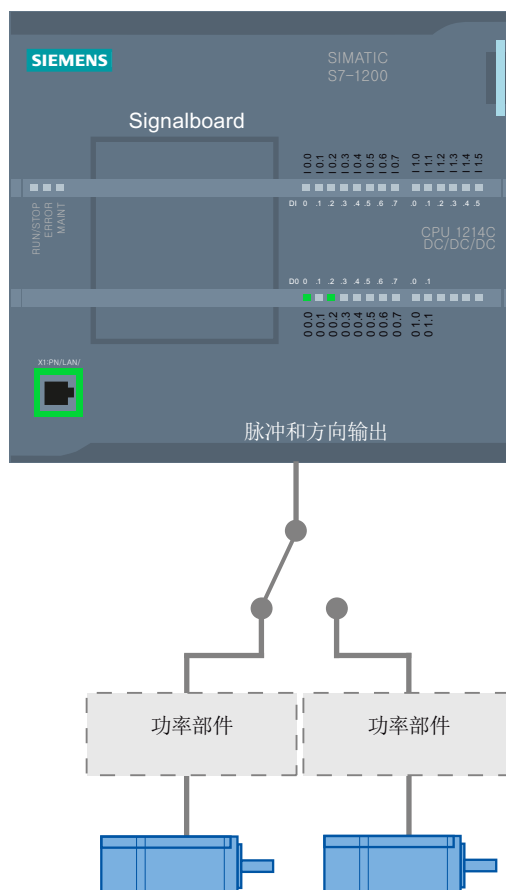
使用软件限位开关的特殊情况 (页 120)

ErrorID 和 ErrorInfo 列表（从 V2.0 的工艺对象起）(页 128)

轴工艺对象的变量 (页 140)

1.12.2 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，则可以通过切换的方式使用一个公共 PTO (Pulse Train Output) 来运行这些驱动器。下图显示了基本的电路设计：



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。有关该主题的详细信息，请参见“将多个轴与同一个 PTO 一起使用 (页 113)”。

参见

将多个轴与同一个 PTO 一起使用 (页 113)

在更高优先级等级（执行级别）中跟踪作业 (页 117)

使用软件限位开关的特殊情况 (页 120)

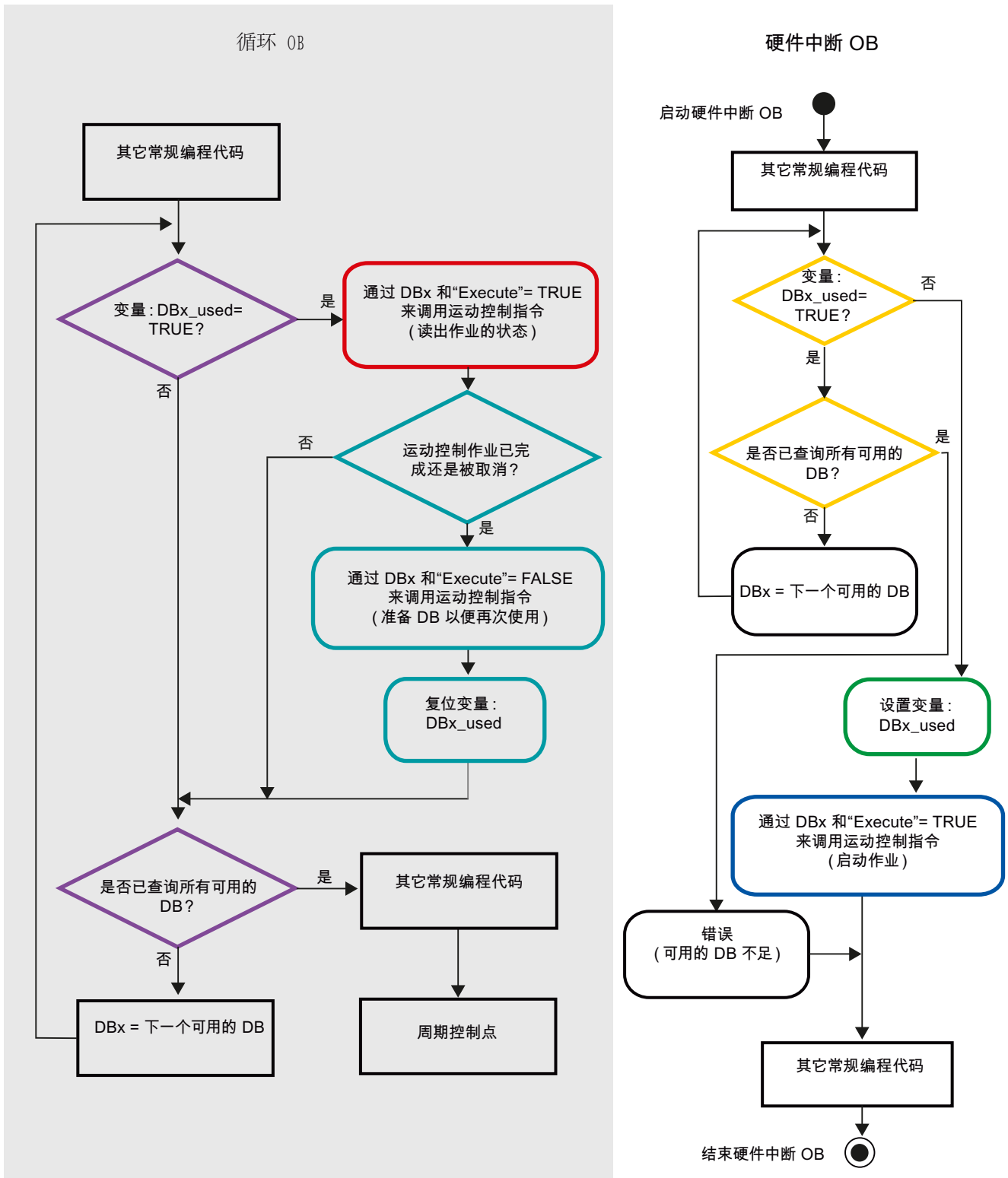
ErrorID 和 ErrorInfo 列表（从 V2.0 的工艺对象起） (页 128)

轴工艺对象的变量 (页 140)

1.12.3 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）中启动运动控制作业（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。在这些情况之下的跟踪可以在循环 OB 中执行。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）中启动运动控制作业和继续在程序循环 OB 中跟踪：



根据想要启动运动控制作业的频率，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 中启动运动控制作业

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启动运动控制作业。

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作业

检查循环 OB 中可用的所有背景数据块以确定“`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（正在处理运动控制作业），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）。

如果作业完成或已中止，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

这随即完成作业跟踪，并且该背景数据块此时可供再次使用。

参见

将多个轴与同一个 PTO 一起使用 (页 113)

将多个驱动器与同一个 PTO 一起使用 (页 116)

使用软件限位开关的特殊情况 (页 120)

`ErrorID` 和 `ErrorInfo` 列表（从 V2.0 的工艺对象起）(页 128)

轴工艺对象的变量 (页 140)

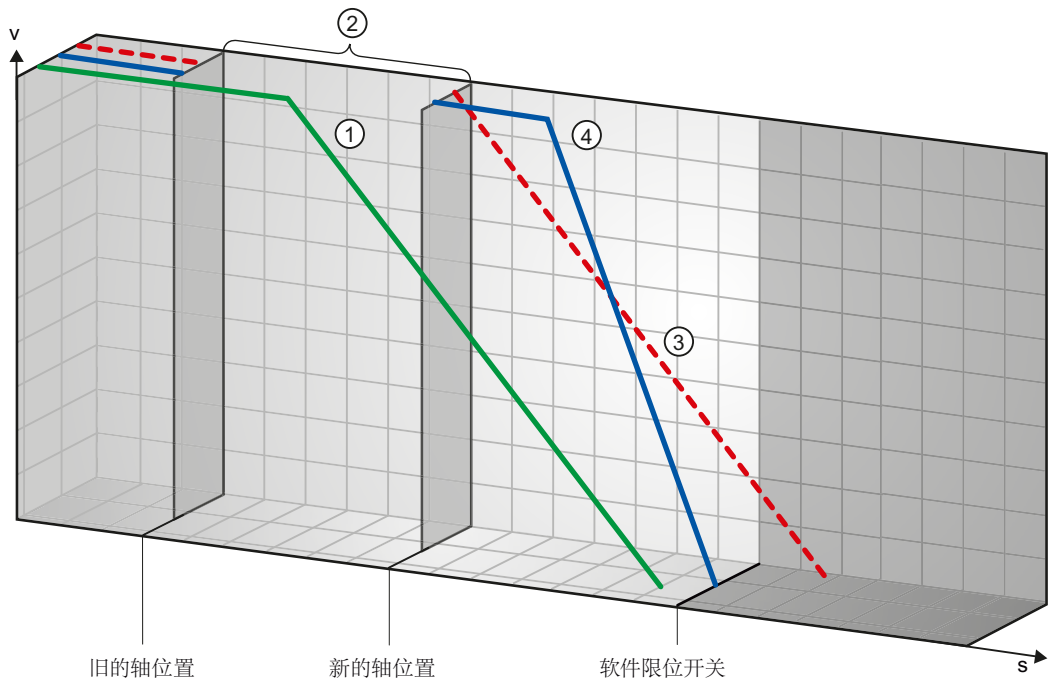
1.12.4 使用软件限位开关的特殊情况

1.12.4.1 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，在软件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:

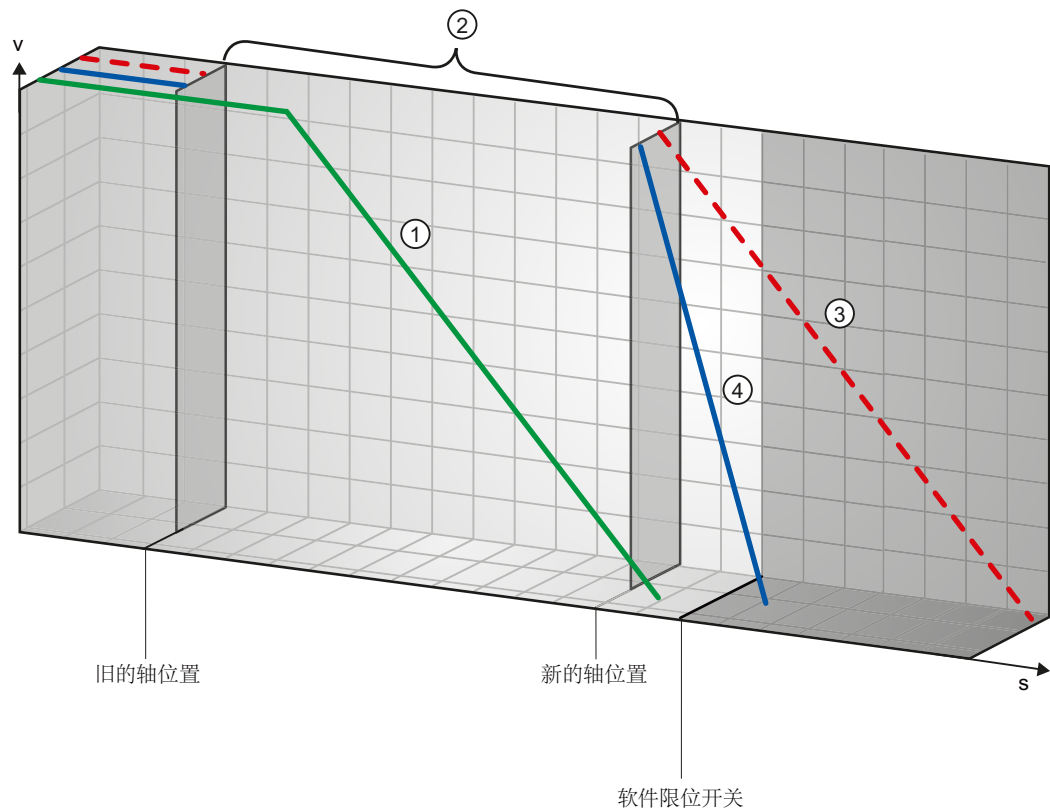
行进命令执行期间，回原点作业（例如，设置参考点）使当前轴位置在软件限位开关方向上偏移。仍可在到达软件限位开关前使轴停止：



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件限位开关前的某处。
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件限位开关后的某处（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，轴以急停减速度制动并停止在软件限位开关的位置处。

示例 2:

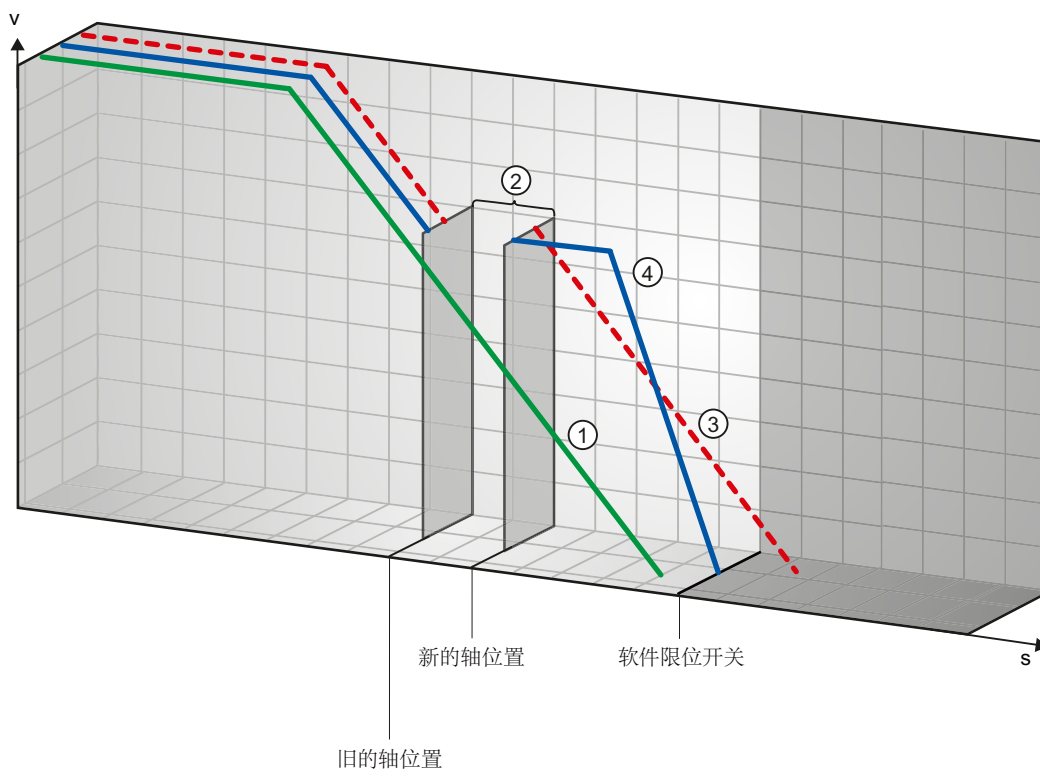
行进命令执行期间，回原点作业（例如，设置参考点）使当前轴位置在软件限位开关方向上偏移。与示例 1 不同，无法再在到达软件限位开关前使轴停止。轴超出软件限位开关的位置。



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件限位开关前的某处。
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件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。轴以急停减速度制动。但该急停减速度不足以使轴停止在软件限位开关的位置处。将超出软件限位开关的位置。

示例 3:

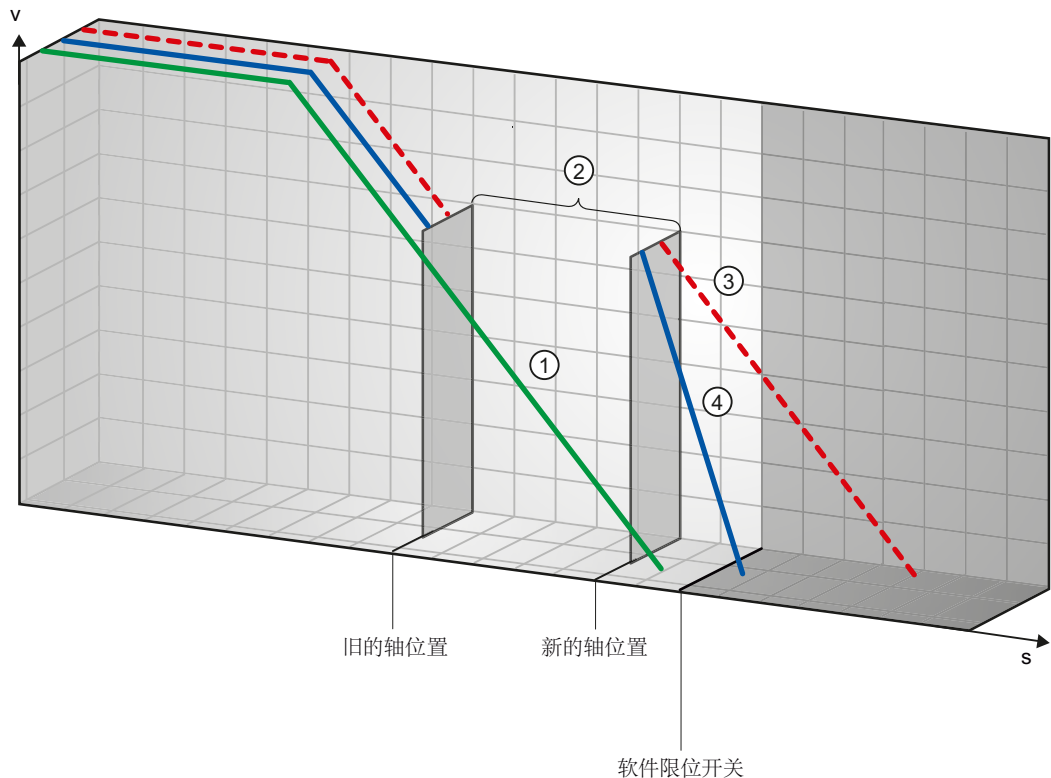
制动操作期间，回原点作业（例如，设置参考点）使当前轴位置在软件限位开关方向上偏移。 仍可在到达软件限位开关前使轴停止：



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件限位开关前的某处。
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件限位开关后的某处（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，轴以急停减速度制动并停止在软件限位开关的位置处。

示例 4:

制动操作期间，回原点作业（例如，设置参考点）使当前轴位置在软件限位开关方向上偏移。与示例 3 不同，无法再在到达软件限位开关前使轴停止。轴超出软件限位开关的位置。



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件限位开关前的某处。
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件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。轴以急停减速度制动。但该急停减速度不足以使轴停止在软件限位开关的位置处。将超出软件限位开关的位置。

参见

软件限位开关和软件限位开关的位置变化。(页 124)

软件限位开关与动态更改结合使用(页 125)

触发位置限制时的轴操作(页 40)

1.12.4.2 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 120)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 120)

软件限位开关与动态更改结合使用(页 125)

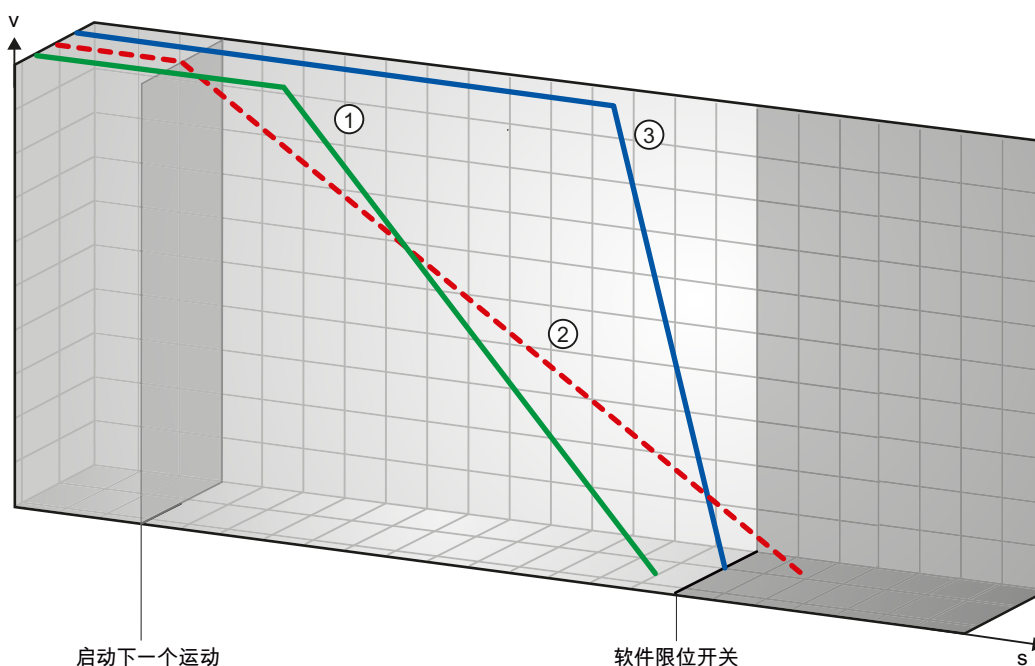
触发位置限制时的轴操作(页 40)

1.12.4.3 软件限位开关与动态更改结合使用

可以结合覆盖运动作业来改变轴在软件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>.Config.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:

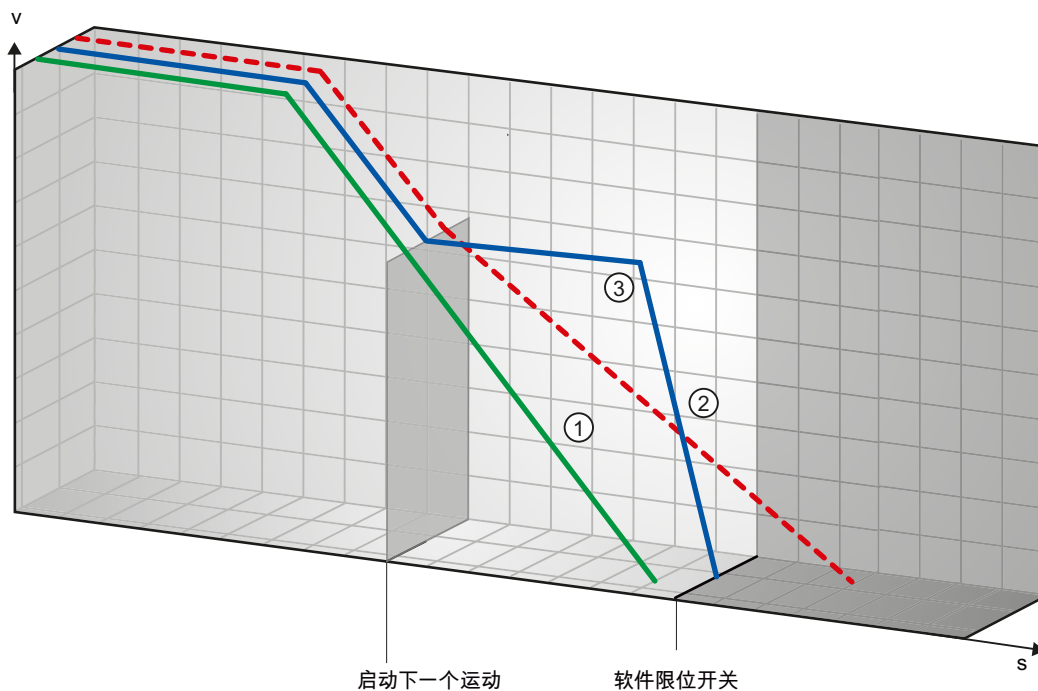
轴运动期间，一个具有较低减速度的运动作业覆盖另一个激活的运动作业：



①	绿色曲线显示了激活的作业 未被覆盖 时该作业的运动。轴以组态的减速度制动并停止在软件限位开关前的某处。
②	基于具有较低减速度的覆盖运动作业，轴理论上会以组态的减速度停止在软件限位开关后的某处（红色曲线）。
③	由于以覆盖运动作业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件限位开关的位置处。

示例 2:

轴制动期间，一个具有较低减速度的运动作业覆盖另一个激活的运动作业：



①	绿色曲线显示了激活的作业未被覆盖时该作业的运动。轴以组态的减速度制动并停止在软件限位开关前的某处。
②	基于具有较低减速度的覆盖运动作业，轴理论上会停止在软件限位开关后较远的某处（红色曲线）。
③	由于以覆盖运动作业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件限位开关的位置处。

参见

软件限位开关与回原点操作结合使用 (页 120)

软件限位开关和软件限位开关的位置变化。(页 124)

触发位置限制时的轴操作 (页 40)

1.12.5 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 * 500Hz

如果计划的定位时间大于 2 ms，则速度不降低。

1.12.6 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、减速度和冲击）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。

例如，由于组态的启动/停止速度较低，第一个脉冲所需的时间将长于整个加速过程可能需要的时间，就属于这种情况。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。

确保在脉冲丢失的情况下，所用的硬件（驱动器）会通过调整适应这种情况，或更改轴的动态设置以避免启动/停止速度的动态调整。

1.12.7 ErrorID 和 ErrorInfo 列表（从 V2.0 的工艺对象起）

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
16#8000		驱动器错误，“驱动器准备就绪”故障	
	16#0001	-	使用指令“MC_Reset”确认错误；提供驱动器信号；可能会重新启动命令
16#8001		已触发下限软件限位开关	
	16#000E	已经以当前组态的减速度到达下限软件限位开关的位置	使用“MC_Reset”指令确认错误；使用运动命令使轴沿正方向移动，脱离软件限位开关的范围
	16#000F	已经以急停减速度到达下限软件限位开关的位置	
	16#0010	已经以急停减速度超出下限软件限位开关的位置	
16#8002		已触发上限软件限位开关	
	16#000E	已经以当前组态的减速度到达上限软件限位开关的位置	使用“MC_Reset”指令确认错误；使用运动命令使轴沿负方向移动，脱离软件限位开关的范围
	16#000F	已经以急停减速度到达上限软件限位开关的位置	
	16#0010	已经以急停减速度超出上限软件限位开关的位置	
16#8003		已逼近下限硬件限位开关	
	16#000E	已逼近下限硬件限位开关。轴以急停减速度停止。 (主动参考点逼近期间，未找到回原点开关)	使用指令“MC_Reset”确认已释放的轴的错误；使用运动命令使轴沿正方向移动，脱离硬件限位开关的范围。

ErrorID	ErrorInfo	说明	解决方法
16#8004		已逼近上限硬件限位开关	
	16#000E	已逼近上限硬件限位开关。轴以急停减速停止。 (主动参考点逼近期间, 未找到回原点开关)	使用指令“MC_Reset”确认已释放的轴的错误; 使用运动命令使轴沿负方向移动, 脱离硬件限位开关的范围。
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 将多个轴与同一个 PTO 一起使用 (页 113))</p>
16#8006		控制面板中发生通信错误	
	16#0012	已超时	检查电缆连接, 然后再次按下“手动控制”按钮。
16#8007		无法释放轴	
	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。

不伴随轴停止的运行错误

ErrorID	ErrorInfo	说明	解决方法
16#8200		轴未释放	
	16#0001	-	释放轴；重新启动命令
16#8201		轴已由另一个“MC_Power”实例释放	
	16#0001	-	仅使用一个“MC_Power”实例释放轴
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 个命令)	
	16#0001	-	减小并行激活的作业数；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”（轴命令表）下运行	
	16#0001	-	退出“手动控制”；重新启动命令
16#8204		轴未回原点	
	16#0001	-	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（该错误仅显示在轴命令表中）	
	16#0013	轴已在用户程序中释放。	使用指令 "MC_Power" 禁用轴或者在轴命令表中再次选择“手动控制”
16#8206		工艺对象尚未激活	
	16#0001	-	使用指令“MC_Power”Enable = TRUE 释放轴或者在轴命令表中释放轴。
16#8207		命令被拒绝	
	16#0016	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#0018	当轴正在直接地或被动地回原点时，不能使用命令表进行移动。	一直等到直接地或被动地回原点完成。
	16#0019	当正在处理命令表时，轴不能直接地或被动地回原点。	等到命令表完成或通过运动命令（例如，“MC_Halt”）中止命令表。

ErrorID	ErrorInfo	说明	解决方法
16#8208		最大速度和启动/停止速度的差值无效	
	16#0002	值的数字格式无效	更正该值；重新启动命令
	16#000A	值小于或等于 0	
16#8209		对工艺对象“轴”的加速度无效	
	16#0002	值的数字格式无效	更正该值；重新启动命令
	16#000A	值小于或等于 0	
16#820A		无法重新启动轴	
	16#0013	轴已在用户程序中释放。	使用“MC_Power”指令阻止轴；再次重新启动
	16#0027	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#820B		无法执行命令表	
	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。

块参数错误

ErrorID	ErrorInfo	说明	解决方法
16#8400		运动控制指令的“Position”参数值无效	
	16#0002	值的数字格式无效	更正该值；重新启动命令
	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#8401		运动控制指令的“Distance”参数值无效	
	16#0002	值的数字格式无效	更正该值；重新启动命令
	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#8402		运动控制指令的“Velocity”参数值无效	
	16#0002	值的数字格式无效	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#8403		运动控制指令的“Direction”参数值无效	
	16#0011	选取值无效	更正选取值；重新启动命令
16#8404		运动控制指令的“Mode”参数值无效	
	16#0011	选取值无效	更正选取值；重新启动命令
	16#0015	未组态主动/被动回原点	更正组态并将其下载到控制器；释放轴并重新启动命令
	16#0017	虽然硬件限位开关已禁用，但还是激活了在硬件限位开关处反向	<ul style="list-style-type: none"> 通过使用变量 <轴>.Config.PositionLimits_HW.Active = TRUE 激活硬件限位开关，重新启动命令 更正组态并将其下载到控制器；释放轴并重新启动命令
16#8405		运动控制指令的“StopMode”参数值无效	
	16#0011	选取值无效	更正选取值；再次释放轴

ErrorID	ErrorInfo	说明	解决方法
16#8406		不允许同时正向和反向点动	
	16#0001	-	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#0001	-	禁用激活的轴；然后可以切换到其它轴并释放该轴。
16#8408		运动控制指令的“Axis”参数值无效	
	16#001A	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#001B	指定的值与所需的工艺对象类型不匹配	
	16#001C	指定的值不是运动控制工艺数据块	
16#8409		运动控制指令的“CommandTable”参数值无效	
	16#001A	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#001B	指定的值与所需的工艺对象类型不匹配	
	16#001C	指定的值不是运动控制工艺数据块	
16#840A		运动控制指令的“StartStep”参数值无效	
	16#000A	值小于或等于 0	更正该值；重新启动命令
	16#001D	开始步进大于结束步进	
	16#001E	值大于 32	
16#840B		运动控制指令的“EndStep”参数值无效	
	16#000A	值小于或等于 0	更正该值；重新启动命令
	16#001E	值大于 32	
16#840C		运动控制指令的“RampUpTime”参数值无效	
	16#0002	值的数字格式无效	更正该值；重新启动命令
	16#000A	值小于或等于 0	
16#840D		运动控制指令的“RampDownTime”参数值无效	
	16#0002	值的数字格式无效	更正该值；重新启动命令
	16#000A	值小于或等于 0	

ErrorID	ErrorInfo	说明	解决方法
16#840E		运动控制指令的“EmergencyRampTime”参数值无效	
	16#0002	值的数字格式无效	更正该值；重新启动命令
	16#000A	值小于或等于 0	
16#840F		运动控制指令的“JerkTime”参数值无效	
	16#0002	值的数字格式无效	更正该值；重新启动命令
	16#000A	值小于或等于 0	

轴的组态错误

ErrorID	ErrorInfo	说明	解决方法
16#8600		脉冲发生器 (PTO) 的参数化无效	
	16#000B	地址无效	<ul style="list-style-type: none"> 将无错误组态下载到控制器；使用指令“MC_Power”再次释放轴
	16#0014	所选硬件由另一应用程序使用	
16#8601		高速计数器 (HSC) 的参数化无效	
	16#000B	地址无效	<ul style="list-style-type: none"> 将无错误组态下载到控制器；使用指令“MC_Power”再次释放轴
	16#0014	所选硬件由另一应用程序使用	
16#8602		“使能输出”(Enable-Output) 的参数分配无效	
	16#000B	地址无效	<ul style="list-style-type: none"> 将无错误组态下载到控制器；使用指令“MC_Power”再次释放轴
16#8603		“输入就绪”(Ready-Input) 的参数分配无效	
	16#000B	地址无效	<ul style="list-style-type: none"> 将无错误组态下载到控制器；使用指令“MC_Power”再次释放轴
16#8604		“电机每转的脉冲数”(Pulses per motor revolution) 值无效	
	16#000A	值小于或等于零	<ul style="list-style-type: none"> 将无错误组态下载到控制器；使用指令“MC_Power”再次释放轴
16#8605		“电机每转的负载距离”(Load distance per motor revolution) 值无效	
	16#0002	值的数字格式无效	<ul style="list-style-type: none"> 将无错误组态下载到控制器；使用指令“MC_Power”再次释放轴
	16#0005	该值超出取值范围（大于 $1e^{12}$ ）	
	16#000A	值小于或等于零	

ErrorID	ErrorInfo	说明	解决方法
16#8606	“启动/停止速度”(Start / stop velocity) 值无效		
	16#0002	值的数字格式无效	<ul style="list-style-type: none"> 将无错误组态下载到控制器；使用指令“MC_Power”再次释放轴
	16#0003	值超出硬件上限值	
	16#0004	值小于硬件下限值	
	16#0007	启动/停止速度大于最大速度	
16#8607	“最大速度”(maximum velocity) 值无效		
	16#0002	值的数字格式无效	<ul style="list-style-type: none"> 将无错误组态下载到控制器；使用指令“MC_Power”再次释放轴
	16#0003	值超出硬件上限值	
	16#0004	值小于硬件下限值	
16#8608	“加速度”(Acceleration) 值无效		
	16#0002	值的数字格式无效	<ul style="list-style-type: none"> 将无错误组态下载到控制器；使用指令“MC_Power”再次释放轴 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
	16#0003	值超出硬件上限值	
	16#0004	值小于硬件下限值	
16#8609	“减速度”(Deceleration) 值无效		
	16#0002	值的数字格式无效	<ul style="list-style-type: none"> 将无错误组态下载到控制器；使用指令“MC_Power”再次释放轴 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
	16#0003	值超出硬件上限值	
	16#0004	值小于硬件下限值	
16#860A	“急停减速度”(Emergency stop deceleration) 值无效		
	16#0002	值的数字格式无效	<ul style="list-style-type: none"> 将无错误组态下载到控制器；使用指令“MC_Power”再次释放轴 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
	16#0003	值超出硬件上限值	
	16#0004	值小于硬件下限值	

ErrorID	ErrorInfo	说明	解决方法
16#860B		下限软件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"> 将无错误组态下载到控制器；使用指令“MC_Power”再次释放轴 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
	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	下限软件限位开关的位置值大于上限软件限位开关的位置值	
16#860C		上限软件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"> 将无错误组态下载到控制器；使用指令“MC_Power”再次释放轴 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
	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#860D		下限硬件限位开关的地址无效	
	16#000C	下降沿的地址无效	<ul style="list-style-type: none"> 将无错误组态下载到控制器；使用指令“MC_Power”再次释放轴
	16#000D	上升沿的地址无效	
16#860E		上限硬件限位开关的地址无效	
	16#000C	下降沿的地址无效	<ul style="list-style-type: none"> 将无错误组态下载到控制器；使用指令“MC_Power”再次释放轴
	16#000D	上升沿的地址无效	
16#860F		“回原点位置偏移”(home position offset) 值无效	
	16#0002	值的数字格式无效	<ul style="list-style-type: none"> 将无错误组态下载到控制器；使用指令“MC_Power”再次释放轴 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
	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#8610		“逼近速度”(approach velocity) 值无效	
	16#0002	值的数字格式无效	<ul style="list-style-type: none"> 将无错误组态下载到控制器；使用指令“MC_Power”再次释放轴 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
	16#0008	速度大于最大速度	
	16#0009	速度小于启动/停止速度	

ErrorID	ErrorInfo	说明	解决方法
16#8611	“回原点速度”(Homing velocity) 值无效		
	16#0002	值的数字格式无效	<ul style="list-style-type: none"> 将无错误组态下载到控制器；使用指令“MC_Power”再次释放轴 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
	16#0008	速度大于最大速度	
	16#0009	速度小于启动/停止速度	
16#8612	回原点开关的地址无效		
	16#000C	下降沿的地址无效	<ul style="list-style-type: none"> 将无错误组态下载到控制器；使用指令“MC_Power”再次释放轴
	16#000D	上升沿的地址无效	
16#8613	虽然未组态硬件限位开关，但还是在主动回原点期间，激活了在该硬件限位开关处反向		
	16#0001	-	<ul style="list-style-type: none"> 将无错误组态下载到控制器；使用指令“MC_Power”再次释放轴 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
16#8614	“冲击”(Jerk) 值无效		
	16#0002	值的数字格式无效	<ul style="list-style-type: none"> 将无错误组态下载到控制器；使用指令“MC_Power”再次释放轴 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
	16#001F	值大于最大冲击值	
	16#0020	值小于最小冲击值	
16#8615	“测量单位”的值无效		
	16#0011	选取值无效	将无错误组态下载到控制器；使用指令“MC_Power”再次释放轴

命令表的组态错误

ErrorID	ErrorInfo	说明	解决方法
16#8700		命令表中“命令类型”(Command type) 的值无效	
	16#0001	-	<ul style="list-style-type: none"> 将无错误组态下载到控制器；使用指令“MC_Power”再次释放轴 在线更正不正确的值，并视需要重新启动命令
16#8701		命令表中“位置/行进路径”(Position/travel path) 的值无效	
	16#0002	值的数字格式无效	<ul style="list-style-type: none"> 将无错误组态下载到控制器；使用指令“MC_Power”再次释放轴 在线更正不正确的值，并视需要重新启动命令
	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#8702		命令表中“速度”(Velocity) 的值无效	
	16#0002	值的数字格式无效	<ul style="list-style-type: none"> 将无错误组态下载到控制器；使用指令“MC_Power”再次释放轴 在线更正不正确的值，并视需要重新启动命令
	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#8703		命令表中“持续时间”(Duration) 的值无效	
	16#0002	值的数字格式无效	<ul style="list-style-type: none"> 将无错误组态下载到控制器；使用指令“MC_Power”再次释放轴 在线更正不正确的值，并视需要重新启动命令
	16#0021	值大于 64800 s	
	16#0022	值小于 0.001 s	
16#8704		命令表中“下一步”(Next step) 的值无效	
	16#0011	选取值无效	<ul style="list-style-type: none"> 将无错误组态下载到控制器；使用指令“MC_Power”再次释放轴 在线更正不正确的值，并视需要重新启动命令
	16#0023	该命令不允许命令转换	

内部错误

ErrorID	ErrorInfo	说明	解决方法
16#8FFF		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"> • ErrorID • ErrorInfo • 诊断缓冲区条目

参见

将多个轴与同一个 PTO 一起使用 (页 113)

将多个驱动器与同一个 PTO 一起使用 (页 116)

在更高优先级等级（执行级别）中跟踪作业 (页 117)

使用软件限位开关的特殊情况 (页 120)

轴工艺对象的变量 (页 140)

1.12.8 轴工艺对象的变量

1.12.8.1 Config. 变量

Config.General. 变量

图例

数据类型	变量的数据类型		
起始值	变量的起始值 轴组态可以改写初始值。		
访问	在用户程序中访问变量的方式：		
	RW	可以在用户程序中读写该变量。	
	R	可以在用户程序中读取该变量。	
	-	不能在用户程序中使用该变量。	
有效	指定变量的更改何时生效。		
HMI	可以在 HMI 系统中使用该变量。		

<轴名称>.Config.General.PTO				
不能在用户程序中评估该变量。				
数据类型	起始值	访问	有效	HMI
DWORD	DW#16#00000000	-	-	-

<轴名称>.Config.General.HSC				
不能在用户程序中评估该变量。				
数据类型	起始值	访问	有效	HMI
DWORD	DW#16#00000000	-	-	-

<轴名称>.Config.General.LengthUnit (从 V2.0“轴”工艺对象起)				
在组态中选择的参数的测量单位:				
<ul style="list-style-type: none"> • 1013 =“毫米” • 1010 =: “米” • 1019 =“英寸” • 1018 =“英尺” • 1005 =“°” (度) • -1 =“脉冲” 				
数据类型	起始值	访问	有效	HMI
Int	1013	R	-	X

Config.DriveInterface. 变量

图例

数据类型	变量的数据类型		
起始值	变量的起始值 轴组态可以改写初始值。		
访问	在用户程序中访问变量的方式:		
	RW	可以在用户程序中读写该变量。	
	R	可以在用户程序中读取该变量。	
	-	不能在用户程序中使用该变量。	
有效	指定变量的更改何时生效。		
HMI	可以在 HMI 系统中使用该变量。		

<轴名称>.Config.DriveInterface.EnableOutput...				
不能在用户程序中评估该变量。				
数据类型	起始值	访问	有效	HMI
-	-	-	-	-

<轴名称>.Config.DriveInterface.ReadyInput...				
不能在用户程序中评估该变量。				
数据类型	起始值	访问	有效	HMI
-	-	-	-	-

Config.Mechanics. 变量**图例**

数据类型	变量的数据类型		
起始值	变量的起始值 轴组态可以改写初始值。		
访问	在用户程序中访问变量的方式：		
	RW	可以在用户程序中读写该变量。	
	R	可以在用户程序中读取该变量。	
	-	不能在用户程序中使用该变量。	
有效	指定变量的更改何时生效。		
HMI	可以在 HMI 系统中使用该变量。		

<轴名称>.Config.Mechanics.PulsesPerDriveRevolution				
电机每转的增量				
数据类型	起始值	访问	有效	HMI
DInt	L#1000	R	-	X

<轴名称>.Config.Mechanics.LeadScrew				
电机每转的负载距离（单位为组态的测量单位）				
数据类型	起始值	访问	有效	HMI
Real	1.0E+001	R	-	X

<轴名称>.Config.Mechanics.InverseDirection				
反转方向信号				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

Config.DynamicLimits. 变量

图例

数据类型	变量的数据类型	
起始值	变量的起始值 轴组态可以改写初始值。	
访问	在用户程序中访问变量的方式：	
	RW	可以在用户程序中读写该变量。
	R	可以在用户程序中读取该变量。
	-	不能在用户程序中使用该变量。
有效	指定变量的更改何时生效。	
HMI	可以在 HMI 系统中使用该变量。	

<轴名称>.Config.DynamicLimits.MinVelocity				
轴的启动/停止速度（单位为组态的测量单位）				
数据类型	起始值	访问	有效	HMI
Real	1.0E+001	R	-	X

<轴名称>.Config.DynamicLimits.MaxVelocity				
轴的最大速度（单位为组态的测量单位）				
数据类型	起始值	访问	有效	HMI
Real	2.5E+002	R	-	X

Config.DynamicDefaults. 变量

图例

数据类型	变量的数据类型	
起始值	变量的起始值 轴组态可以改写初始值。	
访问	在用户程序中访问变量的方式：	
	RW	可以在用户程序中读写该变量。
	R	可以在用户程序中读取该变量。
	-	不能在用户程序中使用该变量。
有效	指定变量的更改何时生效。	
	1	在轴激活（变量 <轴名称>.StatusBits.Activated 从 FALSE 更改为 TRUE）、锁定或释放时
	2	轴已启用时
	5	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	MC_MoveJog 命令停止时
HMI	可以在 HMI 系统中使用该变量。	

<轴名称>.Config.DynamicDefaults.Acceleration					
轴的加速度（单位为组态的量纲单位）					
数据类型	起始值	访问	有效		HMI
Real	4.8E+001	RW	5	CPU 固件 V1.0	X
			1,	自 V2.0 版起的 CPU 固件	
			5, 6		

<轴名称>.Config.DynamicDefaults.Deceleration					
轴的减速度（单位为组态的量纲单位）					
数据类型	起始值	访问	有效		HMI
Real	4.8E+001	RW	5, 6	CPU 固件 V1.0	X
			1, 5, 6	自 V2.0 版起的 CPU 固件	

<轴名称>.Config.DynamicDefaults.EmergencyDeceleration					
轴的急停减速度（单位为组态的量纲单位）					
数据类型	起始值	访问	有效		HMI
Real	1.2E+002	RW	2, 5, 6	CPU 固件 V1.0	X
			1, 5, 6	自 V2.0 版起的 CPU 固件	

<轴名称>.Config.DynamicDefaults.JerkActive （从 V2.0“轴”工艺对象起）					
TRUE = 冲击限制已激活					
数据类型	起始值	访问	有效		HMI
Bool	FALSE	RW	1, 5		X

<轴名称>.Config.DynamicDefaults.Jerk （从 V2.0“轴”工艺对象起）					
轴加速和轴减速斜坡期间的冲击（使用组态的测量单位指定）					
数据类型	起始值	访问	有效		HMI
Real	1.92E+002	RW	1, 5		X

Config.PositionLimits_SW. 变量

图例

数据类型	变量的数据类型	
起始值	变量的起始值 轴组态可以改写初始值。	
访问	在用户程序中访问变量的方式:	
	RW	可以在用户程序中读写该变量。
	R	可以在用户程序中读取该变量。
	-	不能在用户程序中使用该变量。
有效	指定变量的更改何时生效。	
	1	在轴激活（变量 <轴名称>.StatusBits.Activated 从 FALSE 更改为 TRUE）、锁定或释放时
	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	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)。
HMI	可以在 HMI 系统中使用该变量。	

<轴名称>.Config.PositionLimits_SW.Active					
TRUE = 软件限位开关激活					
数据类型	起始值	访问	有效		HMI
Bool	FALSE	RW	4	CPU 固件 V1.0	X
			1, 5, 6	自 V2.0 版起的 CPU 固件	

<轴名称>.Config.PositionLimits_SW.MinPosition					
下限软件限位开关的位置（单位为组态的测量单位）					
数据类型	起始值	访问	有效		HMI
Real	-1.0E+004	RW	4	CPU 固件 V1.0	X
			1, 5, 6	自 V2.0 版起的 CPU 固件	

<轴名称>.Config.PositionLimits_SW.MaxPosition					
上限软件限位开关的位置（单位为组态的测量单位）					
数据类型	起始值	访问	有效		HMI
Real	1.0E+004	RW	4	CPU 固件 V1.0	X
			1, 5, 6	自 V2.0 版起的 CPU 固件	

Config.PositionLimits_HW. 变量

图例

数据类型	变量的数据类型	
起始值	变量的起始值 轴组态可以改写初始值。	
访问	在用户程序中访问变量的方式:	
	RW	可以在用户程序中读写该变量。
	R	可以在用户程序中读取该变量。
	-	不能在用户程序中使用该变量。
有效	指定变量的更改何时生效。	
	1	在轴激活（变量 <轴名称>.StatusBits.Activated 从 FALSE 更改为 TRUE）、锁定或释放时
	3	启用轴后（轴之前必须处于停止状态）。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	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)。
HMI	可以在 HMI 系统中使用该变量。	

<轴名称>.Config.PositionLimits_HW.Active					
TRUE = 硬件限位开关激活。					
数据类型	起始值	访问	有效		HMI
Bool	FALSE	RW	3,	CPU 固件 V1.0	X
			4		
			1,	自 V2.0 版起的	
			5,	CPU 固件	
			6		

<轴名称>.Config.PositionLimits_HW.MinSwitchedLevel				
TRUE = CPU 输入端电平为 24 V，表示已逼近下限硬件限位开关 FALSE = CPU 输入端电平为 0 V，表示已逼近下限硬件限位开关				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

<轴名称>.Config.PositionLimits_HW.MinFallingEvent				
不能在用户程序中评估该变量。				
数据类型	起始值	访问	有效	HMI
DWord	DW#16#00000000	-	-	-

<轴名称>.Config.PositionLimits_HW.MinRisingEvent				
不能在用户程序中评估该变量。				
数据类型	起始值	访问	有效	HMI
DWord	DW#16#00000000	-	-	-

<轴名称>.Config.PositionLimits_HW.MaxSwitchedLevel				
TRUE = CPU 输入端电平为 24 V，表示已逼近上限硬件限位开关 FALSE = CPU 输入端电平为 0 V，表示已逼近上限硬件限位开关				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

<轴名称>.Config.PositionLimits_HW.MaxFallingEvent				
不能在用户程序中评估该变量。				
数据类型	起始值	访问	有效	HMI
DWord	DW#16#00000000	-	-	-

<轴名称>.Config.PositionLimits_HW.MaxRisingEvent				
不能在用户程序中评估该变量。				
数据类型	起始值	访问	有效	HMI
DWord	DW#16#00000000	-	-	-

Config.Homing. 变量

图例

数据类型	变量的数据类型	
起始值	变量的起始值 轴组态可以改写初始值。	
访问	在用户程序中访问变量的方式:	
	RW	可以在用户程序中读写该变量。
	R	可以在用户程序中读取该变量。
	-	不能在用户程序中使用该变量。
有效	指定变量的更改何时生效。	
	1	在轴激活（变量 <轴名称>.StatusBits.Activated 从 FALSE 更改为 TRUE）、锁定或释放时
	7	启动被动回原点命令时
	8	启动主动回原点命令时
HMI	可以在 HMI 系统中使用该变量。	

<轴名称>.Config.Homing.AutoReversal					
TRUE = 在启用硬件限位开关时反向（主动回原点）					
FALSE = 在禁用硬件限位开关时反向（主动回原点）					
数据类型	起始值	访问	有效		HMI
Bool	TRUE	R	-	V1.0“轴”工艺对象	X
		RW	1, 8	V2.0“轴”工艺对象	

<轴名称>.Config.Homing.Direction					
TRUE = 正逼近方向（用于搜索回原点开关）和正回原点方向（主动回原点）					
FALSE = 负逼近方向（用于搜索回原点开关）和正回原点方向（主动回原点）					
数据类型	起始值	访问	有效	HMI	
Bool	TRUE	R	-	V1.0“轴”工艺对象	X
		RW	1, 8	V2.0“轴”工艺对象	

<轴名称>.Config.Homing.SideActiveHoming（从 V2.0“轴”工艺对象起）				
TRUE = 在回原点开关的上方回原点（主动回原点）				
TRUE = 在回原点开关的下方回原点（主动回原点）				
数据类型	起始值	访问	有效	HMI
Bool	TRUE	RW	1, 8	X

<轴名称>.Config.Homing.SidePassiveHoming（从 V2.0“轴”工艺对象起）				
TRUE = 在回原点开关的上方回原点（被动回原点）				
TRUE = 在回原点开关的下方回原点（被动回原点）				
数据类型	起始值	访问	有效	HMI
Bool	TRUE	RW	1, 7	X

<轴名称>.Config.Homing.RisingEdge（从 V1.0“轴”工艺对象起）				
TRUE = 利用回原点开关的下降信号沿回原点（主动回原点）				
FALSE = 利用回原点开关的上升信号沿回原点（主动回原点）				
有关该变量对被动回原点的影响的信息，请参见“组态 - 回原点”中的说明。				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

<轴名称>.Config.Homing.Offset					
回原点位置偏移/单位为组态的测量单位（主动回原点）					
数据类型	起始值	访问	有效		HMI
Real	0.0	R	-	V1.0“轴”工艺对象	X
		RW	1, 8	V2.0“轴”工艺对象	

<轴名称>.Config.Homing.FastVelocity					
逼近速度/单位为组态的测量单位（主动回原点）					
数据类型	起始值	访问	有效		HMI
Real	2.0E+002	R	-	V1.0“轴”工艺对象	X
		RW	1, 8	V2.0“轴”工艺对象	

<轴名称>.Config.Homing.SlowVelocity					
回原点速度/单位为组态的测量单位（主动回原点）					
数据类型	起始值	访问	有效		HMI
Real	4.0E+001	R	-	V1.0“轴”工艺对象	X
		RW	1, 8	V2.0“轴”工艺对象	

<轴名称>.Config.Homing.FallingEvent					
不能在用户程序中评估该变量。					
数据类型	起始值	访问	有效		HMI
DWord	DW#16#00000000	-	-		-

<轴名称>.Config.Homing.RisingEvent					
不能在用户程序中评估该变量。					
数据类型	起始值	访问	有效		HMI
DWord	DW#16#00000000	-	-		

1.12.8.2 MotionStatus. 变量

图例

数据类型	变量的数据类型
起始值	变量的起始值
访问	在用户程序中访问变量的方式:
	RW 可以在用户程序中读写该变量。
	R 可以在用户程序中读取该变量。
-	不能在用户程序中使用该变量。
有效	指定变量的更改何时生效。
HMI	可以在 HMI 系统中使用该变量。

<轴名称>.MotionStatus.Position				
轴的当前位置（单位为组态的测量单位）。如果轴未回原点，则该变量指示相对于轴启用位置的位置值。				
数据类型	起始值	访问	有效	HMI
Real	0.0	R	-	X

<轴名称>..MotionStatus.Velocity				
轴的当前速度（单位为组态的测量单位）				
数据类型	起始值	访问	有效	HMI
Real	0.0	R	-	X

<轴名称>.MotionStatus.Distance				
当前与轴的目标位置的距离（单位为组态的测量单位）。变量值仅在利用“MC_MoveAbsolute”或“MC_MoveRelative”执行定位命令时或执行轴命令表时有效。				
数据类型	起始值	访问	有效	HMI
Real	0.0	R	-	X

<轴名称>.MotionStatus.TargetPosition				
轴的目标位置（单位为组态的测量单位）。变量值仅在使用“MC_MoveAbsolute”或“MC_MoveRelative”执行定位命令时或执行轴命令表时有效。				
数据类型	起始值	访问	有效	HMI
Real	0.0	R	-	X

参见

运动状态 (页 109)

1.12.8.3 StatusBits. 变量

图例

数据类型	变量的数据类型			
起始值	变量的起始值			
访问	在用户程序中访问变量的方式:			
	RW	可以在用户程序中读写该变量。		
	R	可以在用户程序中读取该变量。		
	-	不能在用户程序中使用该变量。		
有效	指定变量的更改何时生效。			
HMI	可以在 HMI 系统中使用该变量。			

<轴名称>.StatusBits.Activated				
TRUE = 轴激活。轴连接到分配的 PTO (Pulse Train Output)。将循环更新工艺数据块中的数据。				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

<轴名称>.StatusBits.Enable				
TRUE = 轴已启用且准备好执行运动控制作业。				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

<轴名称>.StatusBits.HomingDone				
TRUE = 轴已回原点且能够执行绝对定位作业。对于相对回原点而言，轴不必回原点。 在主动回原点期间该状态是 FALSE。如果轴已经回原点，则在被动回原点期间该状态将保持为 TRUE。				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

<轴名称>.StatusBits.Done				
TRUE = 该轴没有已激活的运动控制命令。				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

<轴名称>.StatusBits.Error				
TRUE = 轴工艺对象出错。有关错误的详细信息，请参见自动模式下的运动控制指令的 "ErrorID" 和 "ErrorInfo" 参数。在手动模式下，轴命令表的“错误消息”(Error message) 字段可显示有关错误原因的详细信息。				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

<轴名称>.StatusBits.StandStill				
TRUE = 轴处于停止状态。				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

<轴名称>.StatusBits.PositioningCommand				
TRUE = 该轴正在执行定位命令。				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

<轴名称>.StatusBits.SpeedCommand				
TRUE = 该轴正在以预定义速度执行行进命令。				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

<轴名称>.StatusBits.Homing				
TRUE = 轴正在执行“MC_Home”运动控制指令或轴命令表的回原点命令。				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

<轴名称>.StatusBits.CommandTableActive				
TRUE = 该轴由运动控制指令“MC_CommandTable”控制。				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

<轴名称>.StatusBits.ConstantVelocity				
TRUE = 轴在恒速运转。				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

<轴名称>.StatusBits.Acceleration				
TRUE = 轴在加速。				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

<轴名称>.StatusBits.Deceleration				
TRUE = 轴在减速（速度下降）。				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

<轴名称>.StatusBits.ControlPanelActive				
TRUE = 在轴命令表中启用了“手动控制”模式。轴命令表对“轴”工艺对象具有优先控制权。不能通过用户程序来控制轴。				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

<轴名称>.StatusBits.DriveReady				
TRUE = 驱动器准备就绪。				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

<轴名称>.StatusBits.RestartRequired				
TRUE = 值已在装载存储器中修改。				
要在 CPU RUN 操作模式下将值下载到工作存储器，需要重新启动轴。使用运动控制指令 MC_Reset 执行此操作。				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

参见

状态和错误位 (页 106)

1.12.8.4 ErrorBits. 变量

图例

数据类型	变量的数据类型		
起始值	变量的起始值		
访问	在用户程序中访问变量的方式:		
	RW	可以在用户程序中读写该变量。	
	R	可以在用户程序中读取该变量。	
	-	不能在用户程序中使用该变量。	
有效	指定变量的更改何时生效。		
HMI	可以在 HMI 系统中使用该变量。		

<轴名称>.ErrorBits.SystemFault				
TRUE = 内部系统错误。				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

<轴名称>.ErrorBits.ConfigFault				
TRUE = 轴组态不正确。				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

<轴名称>.ErrorBits.DriveFault				
TRUE = 驱动器在其“驱动器准备就绪”信号出现故障后报告了错误。				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

<轴名称>.ErrorBits.SwLimitMinReached				
TRUE = 已到达下限软件限位开关。				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

<轴名称>.ErrorBits.SwLimitMinExceeded				
TRUE = 已超出下限软件限位开关。				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

<轴名称>.ErrorBits.SwLimitMaxReached				
TRUE = 已到达上限软件限位开关。				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

<轴名称>.ErrorBits.SwLimitMaxExceeded				
TRUE = 已超出上限软件限位开关。				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

<轴名称>.ErrorBits.HwLimitMin				
TRUE = 已逼近下限硬件限位开关。				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

<轴名称>.ErrorBits.HwLimitMax				
TRUE = 已逼近上限硬件限位开关。				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

<轴名称>.ErrorBits.HwUsed				
TRUE = 另一个轴正在使用相同的 PTO (Pulse Train Output) 并且该轴已使用“MC_Power”启用。				
数据类型	起始值	访问	有效	HMI
Bool	FALSE	R	-	X

参见

状态和错误位 (页 106)

1.12.8.5 Internal. 变量

“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。

1.12.8.6 ControlPanel. 变量

“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。

1.12.8.7 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。

1.12.9 命令表工艺对象变量**1.12.9.1 Config.Command.Command[1 ... 32] 变量**

图例

数据类型	变量的数据类型	
起始值	变量的起始值 起始值可由命令表的组态覆盖。	
访问	在用户程序中访问变量的方式：	
	RW	可以在用户程序中读写该变量。
	R	可以在用户程序中读取该变量。
	-	不能在用户程序中使用该变量。
有效	指定变量的更改何时生效。	
HMI	可以在 HMI 系统中使用该变量。	

<轴名称>.Config.Command.Command[x].Type				
命令类型				
<ul style="list-style-type: none"> • 0 =“空”命令 • 2 =“保持”命令 • 5 =“相对定位”命令 • 6 =“绝对定位”命令 • 7 =“速度设定值”命令 • 151 =“等待”命令 				
数据类型	起始值	访问	有效	HMI
Int	0	RW	-	X

<轴名称>. Config.Command.Command[x].Position				
目标位置/行进路径命令				
数据类型	起始值	访问	有效	HMI
Real		RW	-	X

<轴名称>. Config.Command.Command[x].Velocity				
速度命令				
数据类型	起始值	访问	有效	HMI
Real	0.0	RW	-	X

<轴名称>. Config.Command.Command[x].Duration				
持续时间命令				
数据类型	起始值	访问	有效	HMI
Real	0.0	RW	-	X

<轴名称>. Config.Command.Command[x].BufferMode				
“下一步”命令的值				
<ul style="list-style-type: none"> • 0 =“完成命令” • 1 =“混合运动” 				
数据类型	起始值	访问	有效	HMI
Int	0	RW	-	X

<轴名称>. Config.Command.Command[x].StepCode				
命令步代码				
数据类型	起始值	访问	有效	HMI
Word	0	RW	-	X

1.12.10 先前版本的功能的文档

1.12.10.1 组态 - 回原点 (V1.0“轴”工艺对象)

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“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。

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, 可能检测不到参考点。可以在数字量输入的设备组态的“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的持续时间。

允许在到达硬件限位开关后反向 (仅限主动回原点)

激活该复选框, 可将硬件限位开关用作指示参考点逼近的反向凸轮。只有激活硬件限位开关才能实现反向控制。如果使用 CPU 固件 V1.0, 必须组态两个硬件限位开关。如果使用 CPU 固件 V2.0 和更高版本, 仅必须组态逼近方向上的硬件限位开关。

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 (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期间轴到达硬件限位开关, 则将因错误中止参考点逼近并以急停减速度对轴进行制动。

注意

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
- 增大硬件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回原点开关的“逼近方向”以及回原点方向。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“回原点开关”下的表格。

回原点开关侧（主动和被动回原点）

- **主动回原点**

在此处可以选择轴是在回原点开关的下侧还是上侧回原点。

说明

根据轴的起始位置和回原点参数的组态，参考点逼近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。回原点发生在回原点开关的哪一侧取决于以下因素：

- “逼近方向”组态
- “回原点开关”组态
- 被动回原点期间的当前行进方向

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 回原点开关	当前的行进方向	回原点发生在 回原点开关
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）

在该字段中，可以指定参考点逼近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（仅限主动回原点）

在该字段中，可以指定回原点时轴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

回原点位置偏移（仅限主动回原点）

如果预期的参考位置与回原点开关的位置有偏差，则可在此字段中指定回原点位置偏移。

如果值不等于 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回原点位置偏移值
2. 达到“回原点位置偏移”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处。

限值（与所选测量单位无关）：

- $-1.0e12 \leq$ 回原点位置偏移 $\leq 1.0e12$

回原点位置

运动控制指令“MC_Home”中分配的参数指代的位置被用作回原点位置。

1.12.10.2 ErrorID 和 ErrorInfo 的列表（工艺对象 V1.0）

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
16#8000		驱动器错误，“驱动器准备就绪”故障	
	16#0001	-	使用指令“MC_Reset”确认错误；提供驱动器信号；可能会重新启动命令
16#8001		已触发下限软件限位开关	
	16#000E	已经以当前组态的减速度到达下限软件限位开关的位置	使用“MC_Reset”指令确认错误；使用运动命令使轴沿正方向移动，脱离软件限位开关的范围
	16#000F	已经以急停减速度到达下限软件限位开关的位置	
	16#0010	已经以急停减速度超出下限软件限位开关的位置	
16#8002		已触发上限软件限位开关	
	16#000E	已经以当前组态的减速度到达上限软件限位开关的位置	使用“MC_Reset”指令确认错误；使用运动命令使轴沿负方向移动，脱离软件限位开关的范围
	16#000F	已经以急停减速度到达上限软件限位开关的位置	
	16#0010	已经以急停减速度超出上限软件限位开关的位置	
16#8003		已逼近下限硬件限位开关	
	16#000E	已逼近下限硬件限位开关。轴以急停减速度停止。 (主动参考点逼近期间，未找到回原点开关)	使用指令“MC_Reset”确认已启用的轴的错误；使用运动命令使轴沿正方向移动，脱离硬件限位开关的范围。

ErrorID	ErrorInfo	说明	解决方法
16#8004		已逼近上限硬件限位开关	
	16#000E	已逼近上限硬件限位开关。轴以急停减速停止。 (主动参考点逼近期间, 未找到回原点开关)	使用指令“MC_Reset”确认已启用的轴的错误; 使用运动命令使轴沿负方向移动, 脱离硬件限位开关的范围。
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 将多个轴与同一个 PTO 一起使用 (页 113))</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#0001	-	启用轴；重新启动命令
16#8201		轴已被另一个“MC_Power”实例启用	
	16#0001	-	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作业数的最大值 (所有运动控制工艺对象最多可具有 200 个作业)	
	16#0001	-	减小并行激活的作业数；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”（轴命令表）下运行	
	16#0001	-	退出“手动控制”；重新启动命令
16#8204		轴未回原点	
	16#0001	-	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（该错误仅显示在轴命令表中）	
	16#0001	-	使用指令 "MC_Power" 禁用轴或者在轴命令表中再次选择“手动控制”
16#8206		工艺对象“轴”尚未启用	
	16#0001	-	使用指令“MC_Power”Enable = TRUE 启用轴或者在轴命令表中启用轴。
16#8207		命令被拒绝	
	16#0016	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
16#8400		运动控制指令的“Position”参数值无效	
	16#0002	值的数字格式无效	更正“位置”值；重新启动命令
	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#8401		运动控制指令的“Distance”参数值无效	
	16#0002	值的数字格式无效	更正“Distance”值；重新启动命令
	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#8402		运动控制指令的“Velocity”参数值无效	
	16#0002	值的数字格式无效	更正“Velocity”值；重新启动命令
	16#0008	速度大于最大速度	
	16#0009	速度小于启动/停止速度	
16#8403		运动控制指令的“Direction”参数值无效	
	16#0011	选取值无效	更正选取值；重新启动命令
16#8404		运动控制指令的“Mode”参数值无效	
	16#0011	选取值无效	更正选取值；重新启动命令
	16#0015	未组态主动/被动回原点	
	16#0017	虽然已禁用硬件限位开关，但还是激活了在该硬件限位开关处轴反向	
			<ul style="list-style-type: none"> • 通过使用变量 <轴>.Config.PositionLimits_HW.Active = TRUE 激活硬件限位开关，重新启动命令 • 更正组态并将其下载到控制器；启用轴并重新启动命令
16#8405		运动控制指令的“StopMode”参数值无效	
	16#0011	选取值无效	更正选取值；再次启用轴

ErrorID	ErrorInfo	说明	解决方法
16#8406		不允许同时正向和反向点动	
	16#0001	-	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		只有禁用轴后，才允许使用运动控制指令“MC_Power”开关轴。	
	16#0001	-	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
16#8600		脉冲发生器 (PTO) 的参数化无效	
	16#000B	地址无效	更正 PTO (Pulse Train Output) 的组态并将其下载到控制器
16#8601		高速计数器 (HSC) 的参数化无效	
	16#000B	地址无效	更正 HSC (High Speed Counter) 的组态并将其下载到控制器
16#8602		“使能输出”(Enable output) 的参数分配无效	
	16#000D	地址无效	更正组态并将其下载到控制器
16#8603		“准备就绪输入”(Ready input) 的参数分配无效	
	16#000D	地址无效	更正组态并将其下载到控制器
16#8604		“电机每转的脉冲数”(Pulses per motor revolution) 值无效	
	16#000A	该值小于或等于零	更正组态并将其下载到控制器
16#8605		“电机每转的负载距离”(Load distance per motor revolution) 值无效	
	16#0002	值的数字格式无效	更正组态并将其下载到控制器
	16#000A	该值小于或等于零	

ErrorID	ErrorInfo	说明	解决方法
16#8606	“启动/停止速度”(Start / stop velocity) 值无效		
	16#0002	值的数字格式无效	更正组态并将其下载到控制器
	16#0003	值超出硬件限值	
	16#0004	值小于硬件限值	
	16#0007	启动/停止速度大于最大速度	
16#8607	“最大速度”(Maximum velocity) 值无效		
	16#0002	值的数字格式无效	更正组态并将其下载到控制器
	16#0003	值超出硬件限值	
	16#0004	值小于硬件限值	
16#8608	“加速度”(Acceleration) 值无效		
	16#0002	值的数字格式无效	<ul style="list-style-type: none"> 将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴 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
	16#0003	值超出硬件限值	
	16#0004	值小于硬件限值	
16#8609	“减速度”(Deceleration) 值无效		
	16#0002	值的数字格式无效	<ul style="list-style-type: none"> 将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴 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
	16#0003	值超出硬件限值	
	16#0004	值小于硬件限值	
16#860A	“急停减速度”(Emergency stop deceleration) 值无效		
	16#0002	值的数字格式无效	<ul style="list-style-type: none"> 将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴 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
	16#0003	值超出硬件限值	
	16#0004	值小于硬件限值	

ErrorID	ErrorInfo	说明	解决方法
16#860B		下限软件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"> 将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴 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
	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	下限软件限位开关的位置值大于上限软件限位开关的位置值	
16#860C		上限软件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"> 将无错误组态下载到控制器；使用指令“MC_Power”再次启用轴 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
	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#860D		下限硬件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	
16#860E		上限硬件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	
16#860F		“回原点位置偏移”(home position offset) 值无效	
	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 $-1e^{12}$ ）	
16#8610		“逼近速度”(approach velocity) 值无效	
	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	
16#8611		“回原点速度”(Homing velocity) 值无效	
	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	

ErrorID	ErrorInfo	说明	解决方法
16#8612		回原点开关的地址无效	
	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	
16#8613		虽然未组态硬件限位开关，但还是在主动回原点期间，激活了在该硬件限位开关处反向	
	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
16#8FFF		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"> • ErrorID • ErrorInfo • 诊断缓冲区条目

参见

将多个轴与同一个 PTO 一起使用 (页 113)

S7-1200 运动控制

2.1 MC_Power

2.1.1 MC_Power: 启用、禁用轴

说明

运动控制指令“MC_Power”释放或锁定轴。

要求

- 已正确组态工艺对象“轴”。
- 没有待决的启用/禁止错误。

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作业。

参数

参数	声明	数据类型	默认值	说明	
Axis	INPUT	TO_Axis_1	-	轴工艺对象	
Enable	INPUT	BOOL	FALSE	TRUE	运动控制尝试启用轴。
				FALSE	根据组态的“StopMode”，中止所有已激活的作业，同时停止轴。
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，则会在不减速的情况下禁用轴。脉冲输出立即停止。
				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作业，也不会接受任何新作业（MC_Reset 命令除外）。 轴未归位。 在禁用轴时，只有在轴停止之后，才会将状态更改为 FALSE。

参数	声明	数据类型	默认值	说明
				TRUE 轴已启用。 轴已就绪，可以执行运动控制作业。 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 电源处于通电状态
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 128)
ErrorInfo	OUTPUT	WORD	16#0000	参数“ErrorID”的错误信息 ID (页 128)

注意

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。这就要求输入参数“Enable”在此过程中，保留值为 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. 使用所需值对输入参数“StopMode”进行初始化。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“驱动器准备就绪”信号。

当 CPU 组态完成且输入端出现“驱动器准备就绪”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. 使用所需值对输入参数“StopMode”进行初始化。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

ErrorID 和 ErrorInfo 列表（从 V2.0 的工艺对象起）（页 128）

MC_Power: 功能图（页 181）

MC_Reset: 确认错误（页 182）

MC_Home: 归位轴，设置归位位置（页 185）

MC_Halt: 停止轴（页 189）

MC_MoveAbsolute: 轴的绝对定位（页 193）

MC_MoveRelative: 轴的相对定位（页 198）

MC_MoveVelocity: 以预设的旋转速度移动轴（页 203）

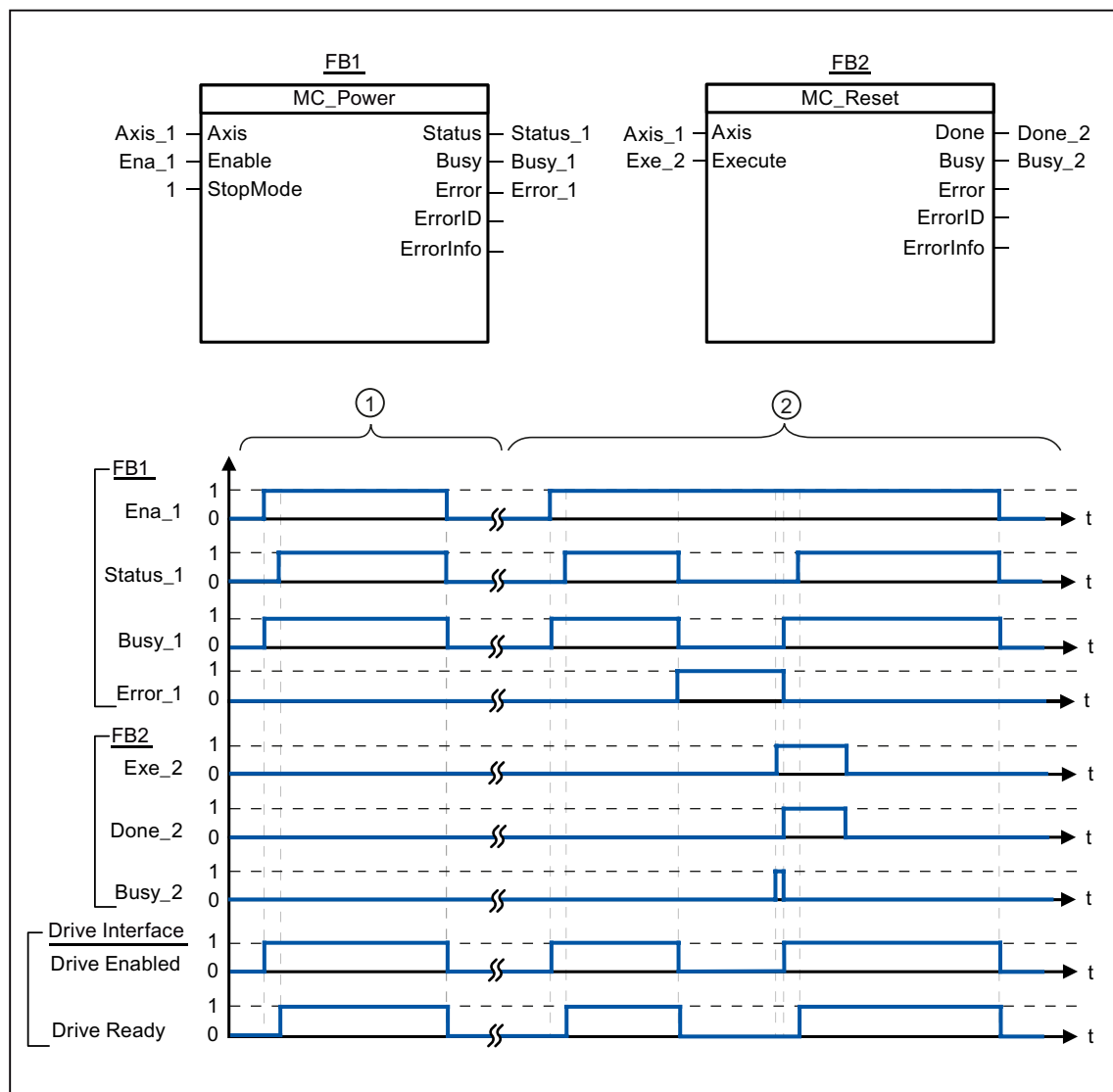
MC_MoveJog: 在点动模式下移动轴（页 208）

MC_ChangeDynamic: 更改轴的动态设置（从 V2.0“轴”工艺对象起）（页 216）

MC_CommandTable: 按移动顺序运行轴作业（从 V2.0“轴”工艺对象起）（页 212）

2.1.2 MC_Power: 功能图

功能图



①	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。
②	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。

参见

MC_Power: 启用、禁用轴 (页 177)

2.2 MC_Reset

2.2.1 MC_Reset: 确认错误

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”下的“ErrorID 和 ErrorInfo 列表”。

从版本 V3.0 开始，可在 RUN 运行模式下将轴组态下载到工作存储器。

要求

- 已正确组态工艺对象“轴”。
- 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“轴”工艺对象中的加速度更正为有效值）。

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作业。

参数

参数	声明	数据类型	默认值	说明	
Axis	INPUT	TO_Axis_1	-	轴工艺对象	
Execute	INPUT	BOOL	FALSE	上升沿时启动命令	
Restart	INPUT	BOOL	FALSE	(从版本 V3.0 开始)	
				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。请参阅有关下载到 CPU (页 79) 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 正在执行命令。	
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 128)	
ErrorInfo	OUTPUT	WORD	16#0000	参数“ErrorID”的错误信息 ID (页 128)	

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：

1. 首先检查是否满足上述要求。
2. 在输入参数“Execute”的上升沿开始确认错误。
3. 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。

参见

下载到 CPU (页 79)

ErrorID 和 ErrorInfo 列表 (从 V2.0 的工艺对象起) (页 128)

MC_Power: 启用、禁用轴 (页 177)

MC_Home: 归位轴, 设置归位位置 (页 185)

MC_Halt: 停止轴 (页 189)

MC_MoveAbsolute: 轴的绝对定位 (页 193)

MC_MoveRelative: 轴的相对定位 (页 198)

MC_MoveVelocity: 以预设的旋转速度移动轴 (页 203)

MC_MoveJog: 在点动模式下移动轴 (页 208)

MC_CommandTable: 按移动顺序运行轴作业 (从 V2.0“轴”工艺对象起) (页 212)

MC_ChangeDynamic: 更改轴的动态设置 (从 V2.0“轴”工艺对象起) (页 216)

2.3 MC_Home

2.3.1 MC_Home: 归位轴, 设置归位位置

说明

使用运动控制指令“MC_Home”，将轴坐标与实际的物理驱动器位置进行匹配。轴的绝对定位需要归位。可执行以下类型的归位：

- 主动归位 (Mode = 3)

自动执行归位步骤。

- 被动归位 (Mode = 2)

在被动回原点期间，运动控制指令“MC_Home”不会执行任何回原点运动。用户必须通过其它运动控制指令，执行这一步骤中所需的往返运动。检测到回原点开关时，轴即回原点。

- 直接绝对归位 (Mode = 0)

将当前的轴位置设置为参数“Position”的值。

- 直接相对归位 (Mode = 1)

将当前轴位置的偏移量设置为参数“Position”的值。

要求

- 已正确组态工艺对象“轴”。
- 轴已启用。
- 没有任何 MC_CommandTable 命令可以在以 Mode = 0、1 或 2 启动时激活。

超驰响应

超驰响应取决于所选的模式：

Mode = 0, 1

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作业。按照新的回原点位置（输入参数“Position”的值）回原点后，将继续执行与位置相关的运动作业。“Position”）。

Mode = 2

可通过下列运动控制作业中止 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）回原点后，将继续执行与位置相关的运动作业。“Position”）。

Mode = 3

可通过下列运动控制作业中止 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

新的 MC_Home 命令可中止下列激活的运动控制作业：

- 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

参数

参数	声明	数据类型	默认值	说明	
Axis	INPUT	TO_Axis_1	-	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"> Mode = 0、2 和 3 完成归位操作之后，轴的绝对位置 Mode = 1 对当前轴位置的修正值 限值： $-1.0e^{12} \leq \text{Position} \leq 1.0e^{12}$	
Mode	INPUT	INT	0	归位模式	
				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 将根据轴组态进行归位。在归位后，将新的轴位置设置为参数“Position”的值。
				3	主动回原点 按照轴组态，逼近参考点。在归位后，将新的轴位置设置为参数“Position”的值。
Done	OUTPUT	BOOL	FALSE	TRUE	命令已完成
Busy	OUTPUT	BOOL	FALSE	TRUE	正在执行命令。
Command Aborted	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。

参数	声明	数据类型	默认值	说明
ErrorID	OUTPUT	WORD	16#0000	参数“Error”的错误 ID
ErrorInfo	OUTPUT	WORD	16#0000	参数“ErrorID”的错误信息 ID

说明

在下列情况下轴归位会丢失：

- 通过运动控制指令“MC_Power”，禁用轴。
- 在自动模式和手动控制之间切换
- 在启动主动归位时。在成功完成归位操作之后，轴归位将再次可用。
- 关闭之后 -> 启动 CPU
- CPU 重新启动后 (RUN-STOP -> STOP-RUN)

归位轴

要归位轴，请按下列步骤操作：

1. 首先检查是否满足上述要求。
2. 使用这些值初始化所需的输入参数，然后在输入参数“Execute”的上升沿，开始归位。
3. 如果输出参数“Done”和工艺对象变量 <轴名称>.StatusBits.HomingDone 的值为 TRUE，则说明归位已完成。

参见

ErrorID 和 ErrorInfo 列表（从 V2.0 的工艺对象起）（页 128）

MC_Power: 启用、禁用轴（页 177）

MC_Reset: 确认错误（页 182）

MC_Halt: 停止轴（页 189）

MC_MoveAbsolute: 轴的绝对定位（页 193）

MC_MoveRelative: 轴的相对定位（页 198）

MC_MoveVelocity: 以预设的旋转速度移动轴（页 203）

MC_MoveJog: 在点动模式下移动轴（页 208）

MC_CommandTable: 按移动顺序运行轴作业（从 V2.0“轴”工艺对象起）（页 212）

MC_ChangeDynamic: 更改轴的动态设置（从 V2.0“轴”工艺对象起）（页 216）

2.4 MC_Halt

2.4.1 MC_Halt: 停止轴

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

- 已正确组态工艺对象“轴”。
- 轴已启用。

超驰响应

可通过下列运动控制作业中止 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

新的 MC_Halt 命令可中止下列激活的运动控制作业：

- 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

参数

参数	声明	数据类型	默认值	说明
Axis	INPUT	TO_Axis_1	-	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。
Command Aborted	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 128)
ErrorInfo	OUTPUT	WORD	16#0000	参数“ErrorID”的错误信息 ID (页 128)

参见

MC_Halt: 功能图 (页 191)

ErrorID 和 ErrorInfo 列表 (从 V2.0 的工艺对象起) (页 128)

MC_Power: 启用、禁用轴 (页 177)

MC_Reset: 确认错误 (页 182)

MC_Home: 归位轴, 设置归位位置 (页 185)

MC_MoveAbsolute: 轴的绝对定位 (页 193)

MC_MoveRelative: 轴的相对定位 (页 198)

MC_MoveVelocity: 以预设的旋转速度移动轴 (页 203)

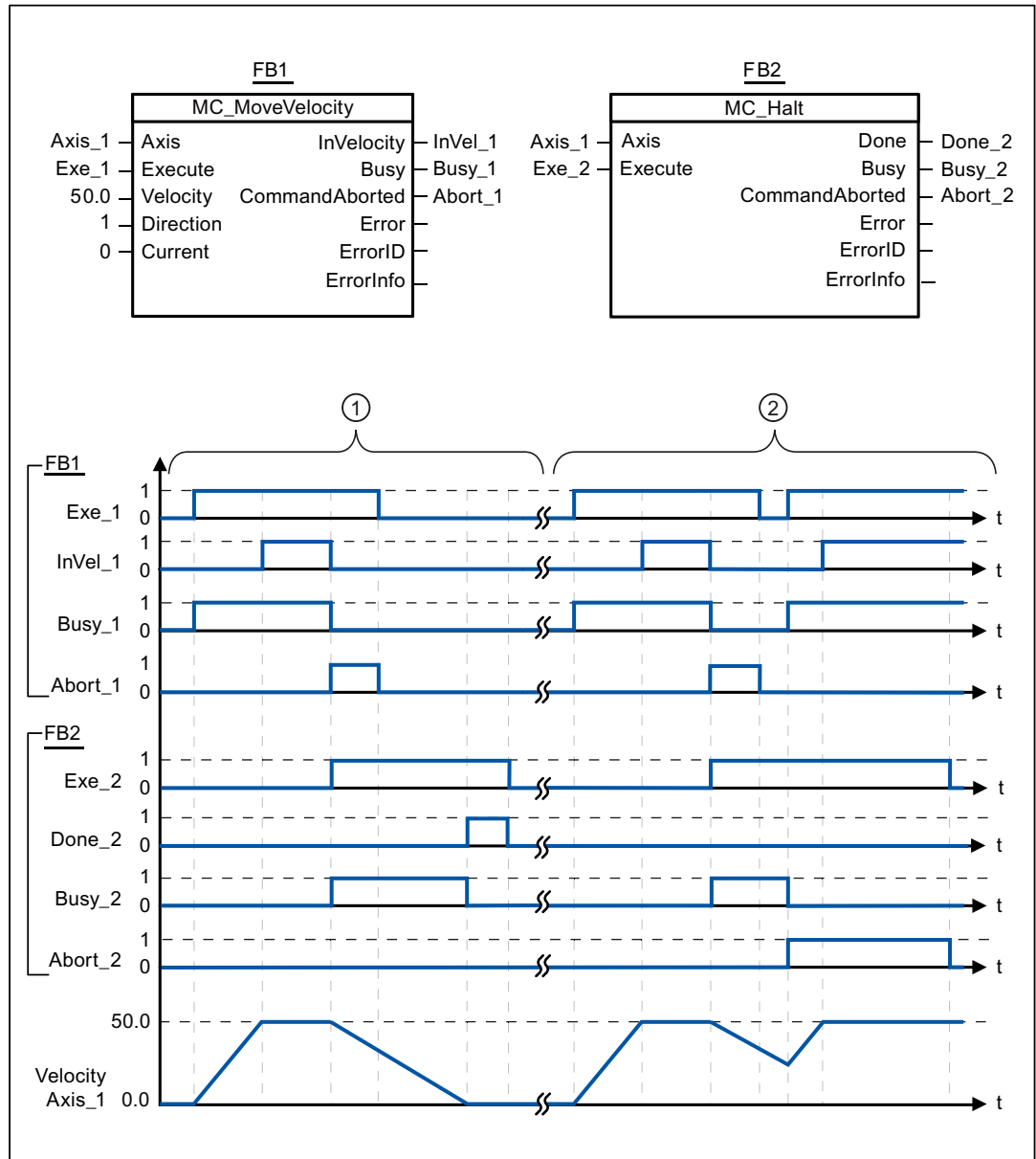
MC_MoveJog: 在点动模式下移动轴 (页 208)

MC_CommandTable: 按移动顺序运行轴作业 (从 V2.0“轴”工艺对象起) (页 212)

MC_ChangeDynamic: 更改轴的动态设置 (从 V2.0“轴”工艺对象起) (页 216)

2.4.2 MC_Halt: 功能图

功能图



2.4 MC_Halt

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

①	轴将由 MC_Halt 作业制动，直至停止为止。通过“Done_2”发出轴停止的信号。
②	当 MC_Halt 作业正在制动轴时，该作业被另一运动作业中止。将通过“Abort_2”发出中止信号。

参见

MC_Halt: 停止轴 (页 189)

2.5 MC_MoveAbsolute

2.5.1 MC_MoveAbsolute: 轴的绝对定位

说明

运动控制指令“MC_MoveAbsolute”启动轴定位运动以将其移动到某个绝对位置。

要求

- 已正确组态工艺对象“轴”。
- 轴已启用。
- 轴已归位。

超驰响应

可通过下列运动控制作业中止 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

新的 MC_MoveAbsolute 命令可中止下列激活的运动控制作业:

- MC_Home 命令 Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute 命令
- MC_MoveRelative 命令
- MC_MoveVelocity 命令
- MC_MoveJog 命令

参数

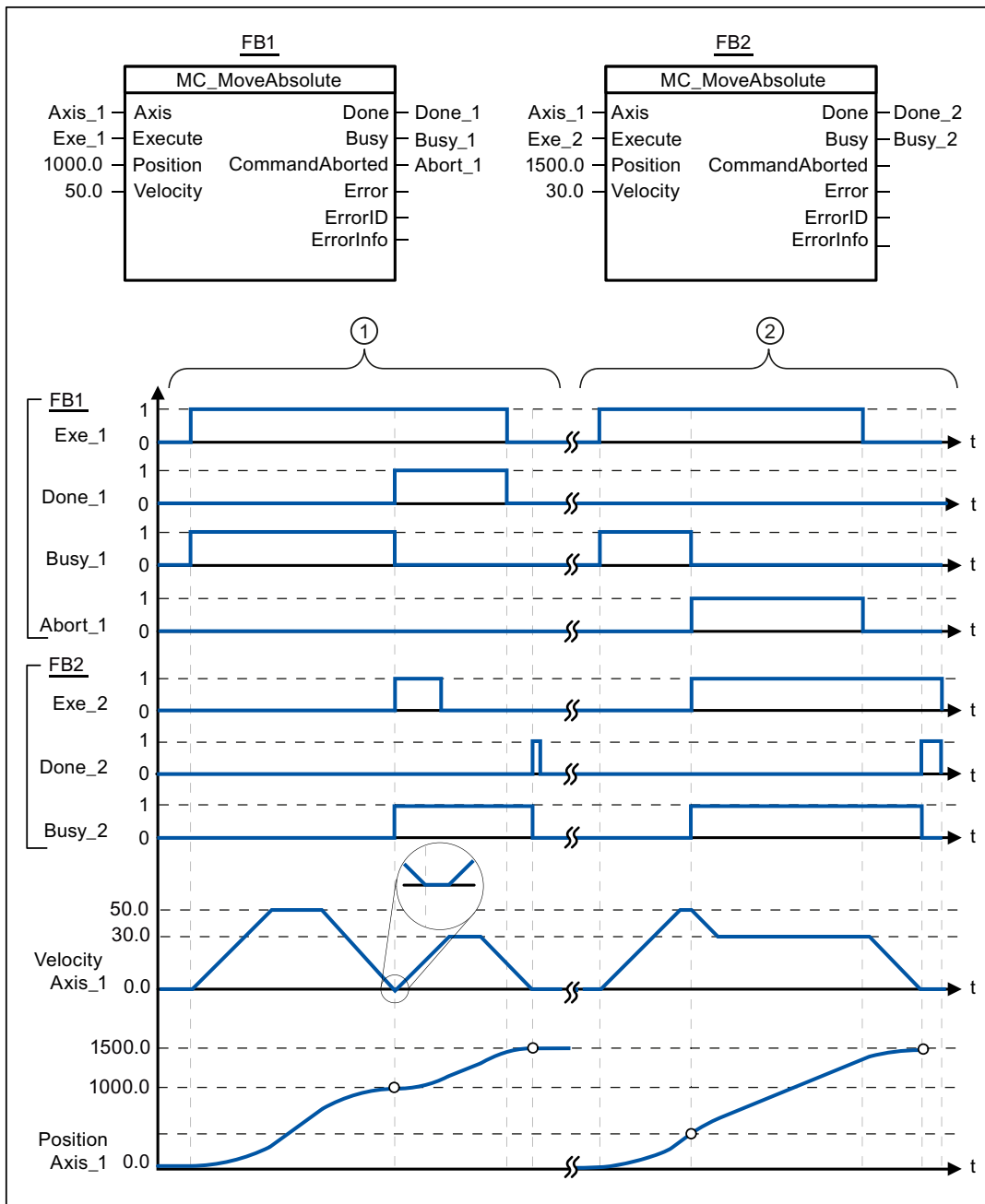
参数	声明	数据类型	默认值	说明
Axis	INPUT	TO_Axis_1	-	轴工艺对象
Execute	INPUT	BOOL	FALSE	上升沿时启动命令
Position	INPUT	REAL	0.0	绝对目标位置 限值： $-1.0e^{12} \leq \text{Position} \leq 1.0e^{12}$
Velocity	INPUT	REAL	10.0	轴的速度 由于所组态的加速度和减速度以及待接近的目标位置等原因，不会始终保持这一速度。 限值： 启动/停止速度 $\leq \text{Velocity} \leq$ 最大速度
Done	OUTPUT	BOOL	FALSE	TRUE 达到绝对目标位置
Busy	OUTPUT	BOOL	FALSE	TRUE 正在执行命令。
Command Aborted	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 128)
ErrorInfo	OUTPUT	WORD	16#0000	参数“ErrorID”的错误信息 ID (页 128)

参见

- MC_MoveAbsolute: 功能图 (页 196)
- ErrorID 和 ErrorInfo 列表 (从 V2.0 的工艺对象起) (页 128)
- MC_Power: 启用、禁用轴 (页 177)
- MC_Reset: 确认错误 (页 182)
- MC_Home: 归位轴, 设置归位位置 (页 185)
- MC_Halt: 停止轴 (页 189)
- MC_MoveRelative: 轴的相对定位 (页 198)
- MC_MoveVelocity: 以预设的旋转速度移动轴 (页 203)
- MC_MoveJog: 在点动模式下移动轴 (页 208)
- MC_CommandTable: 按移动顺序运行轴作业 (从 V2.0“轴”工艺对象起) (页 212)
- MC_ChangeDynamic: 更改轴的动态设置 (从 V2.0“轴”工艺对象起) (页 216)

2.5.2 MC_MoveAbsolute: 功能图

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

①	通过 MC_MoveAbsolute 作业，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 作业。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。
②	激活的 MC_MoveAbsolute 作业被另一个 MC_MoveAbsolute 作业中止。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。

参见

MC_MoveAbsolute: 轴的绝对定位 (页 193)

2.6 MC_MoveRelative

2.6.1 MC_MoveRelative: 轴的相对定位

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

- 已正确组态工艺对象“轴”。
- 轴已启用。

超驰响应

可通过下列运动控制作业中止 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

新的 MC_MoveRelative 命令可中止下列激活的运动控制作业：

- 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

参数

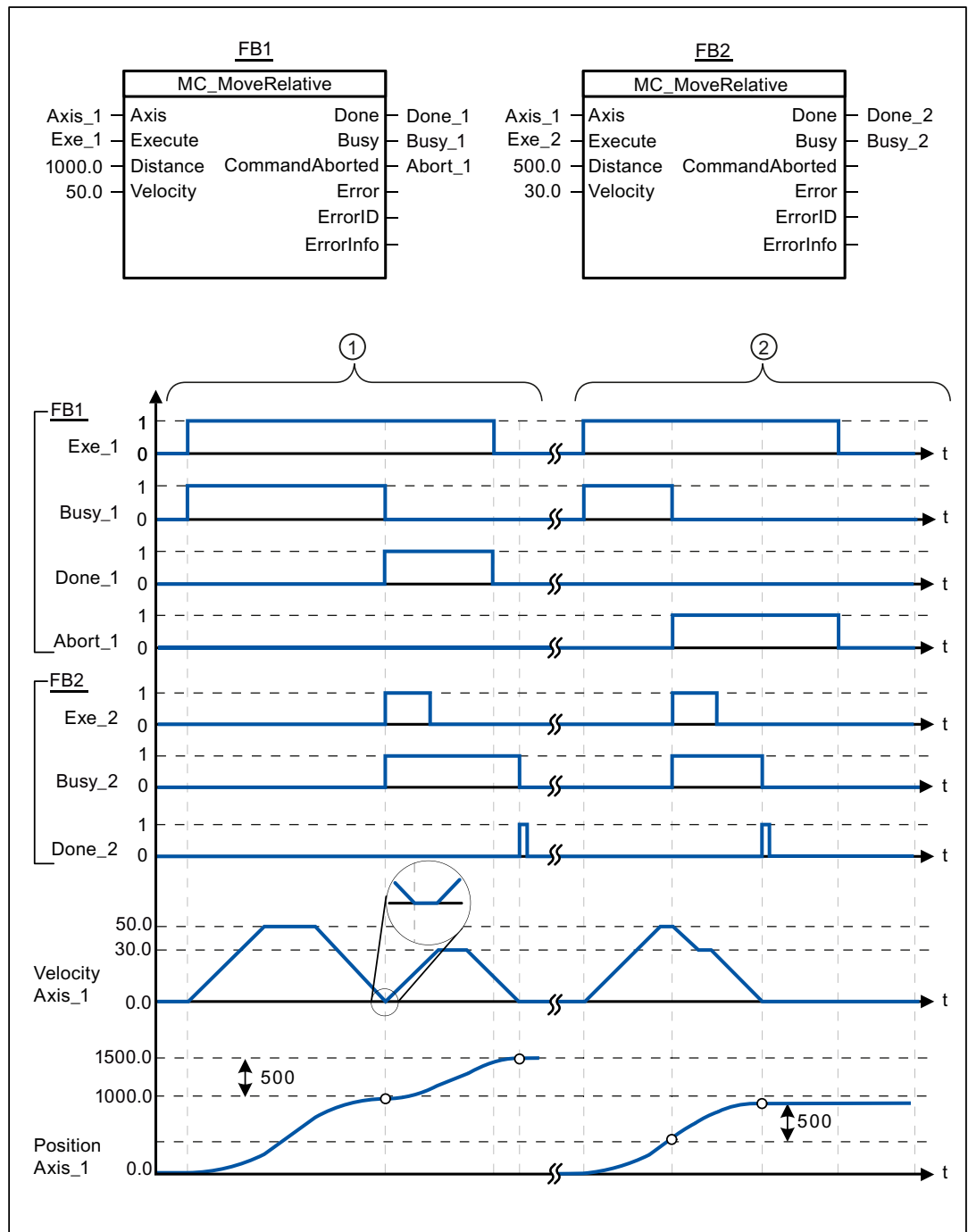
参数	声明	数据类型	默认值	说明
Axis	INPUT	TO_Axis_1	-	轴工艺对象
Execute	INPUT	BOOL	FALSE	上升沿时启动命令
Distance	INPUT	REAL	0.0	定位操作的移动距离 限值： $-1.0e^{12} \leq \text{Distance} \leq 1.0e^{12}$
Velocity	INPUT	REAL	10.0	轴的速度 由于所组态的加速度和减速度以及要途经的距离等原因，不会始终保持这一速度。 限值： 启动/停止速度 \leq Velocity \leq 最大速度
Done	OUTPUT	BOOL	FALSE	TRUE 目标位置已到达
Busy	OUTPUT	BOOL	FALSE	TRUE 正在执行命令。
Command Aborted	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 128)
ErrorInfo	OUTPUT	WORD	16#0000	参数“ErrorID”的错误信息 ID (页 128)

参见

- MC_MoveRelative: 功能图 (页 201)
- ErrorID 和 ErrorInfo 列表 (从 V2.0 的工艺对象起) (页 128)
- MC_Power: 启用、禁用轴 (页 177)
- MC_Reset: 确认错误 (页 182)
- MC_Home: 归位轴, 设置归位位置 (页 185)
- MC_Halt: 停止轴 (页 189)
- MC_MoveAbsolute: 轴的绝对定位 (页 193)
- MC_MoveVelocity: 以预设的旋转速度移动轴 (页 203)
- MC_MoveJog: 在点动模式下移动轴 (页 208)
- MC_CommandTable: 按移动顺序运行轴作业 (从 V2.0“轴”工艺对象起) (页 212)
- MC_ChangeDynamic: 更改轴的动态设置 (从 V2.0“轴”工艺对象起) (页 216)

2.6.2 MC_MoveRelative: 功能图

功能图



2.6 MC_MoveRelative

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

①	通过 MC_MoveRelative 作业，将轴移动了 1000.0 的距离 (“Distance”)。当轴达到目标位置后，将通过“Done_1”发出信号。当“Done_1”= TRUE 时，将启动另一个移动距离为 500.0 的 MC_MoveRelative 作业。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。
②	激活的 MC_MoveRelative 作业被另一个 MC_MoveRelative 作业中止。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。

参见

MC_MoveRelative: 轴的相对定位 (页 198)

2.7 MC_MoveVelocity

2.7.1 MC_MoveVelocity: 以预设的旋转速度移动轴

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

- 已正确组态工艺对象“轴”。
- 轴已启用。

超驰响应

MC_MoveVelocity 可由下列运动控制作业中止：

- 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

新的 MC_MoveVelocity 命令可中止下列激活的运动控制作业：

- 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

参数

参数	声明	数据类型	默认值	说明	
Axis	INPUT	TO_Axis_1	-	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 $\leq \text{Velocity} \leq$ 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。
InVelocity	OUTPUT	BOOL	FALSE	TRUE <ul style="list-style-type: none"> “Current” = FALSE: 达到参数“Velocity”中指定的速度。 “Current” = TRUE: 轴在启动时，以当前速度进行移动。 	
Busy	OUTPUT	BOOL	FALSE	TRUE 正在执行命令。	
Command Aborted	OUTPUT	BOOL	FALSE	TRUE 命令在执行过程中被另一命令中止。	

参数	声明	数据类型	默认值	说明
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 128)
ErrorInfo	OUTPUT	WORD	16#0000	参数“ErrorID”的错误信息 ID (页 128)

以下操作会将速度设置为零 (Velocity = 0.0)

“Velocity”= 0.0 的 MC_MoveVelocity 命令 (如 MC_Halt 命令) 将中止激活的运动作业，并使用组态的减速度来停止轴。

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。

“在减速操作时，Busy”的值为 TRUE，现在与“InVelocity”一起更改为 FALSE。如果将参数“Execute”设置为 TRUE，则“InVelocity”和“Busy”将处于锁定状态。

启动 MC_MoveVelocity 命令后，将在工艺对象中置位状态位“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_MoveVelocity: 功能图 (页 206)

ErrorID 和 ErrorInfo 列表 (从 V2.0 的工艺对象起) (页 128)

MC_Power: 启用、禁用轴 (页 177)

MC_Reset: 确认错误 (页 182)

MC_Home: 归位轴，设置归位位置 (页 185)

MC_Halt: 停止轴 (页 189)

MC_MoveAbsolute: 轴的绝对定位 (页 193)

MC_MoveRelative: 轴的相对定位 (页 198)

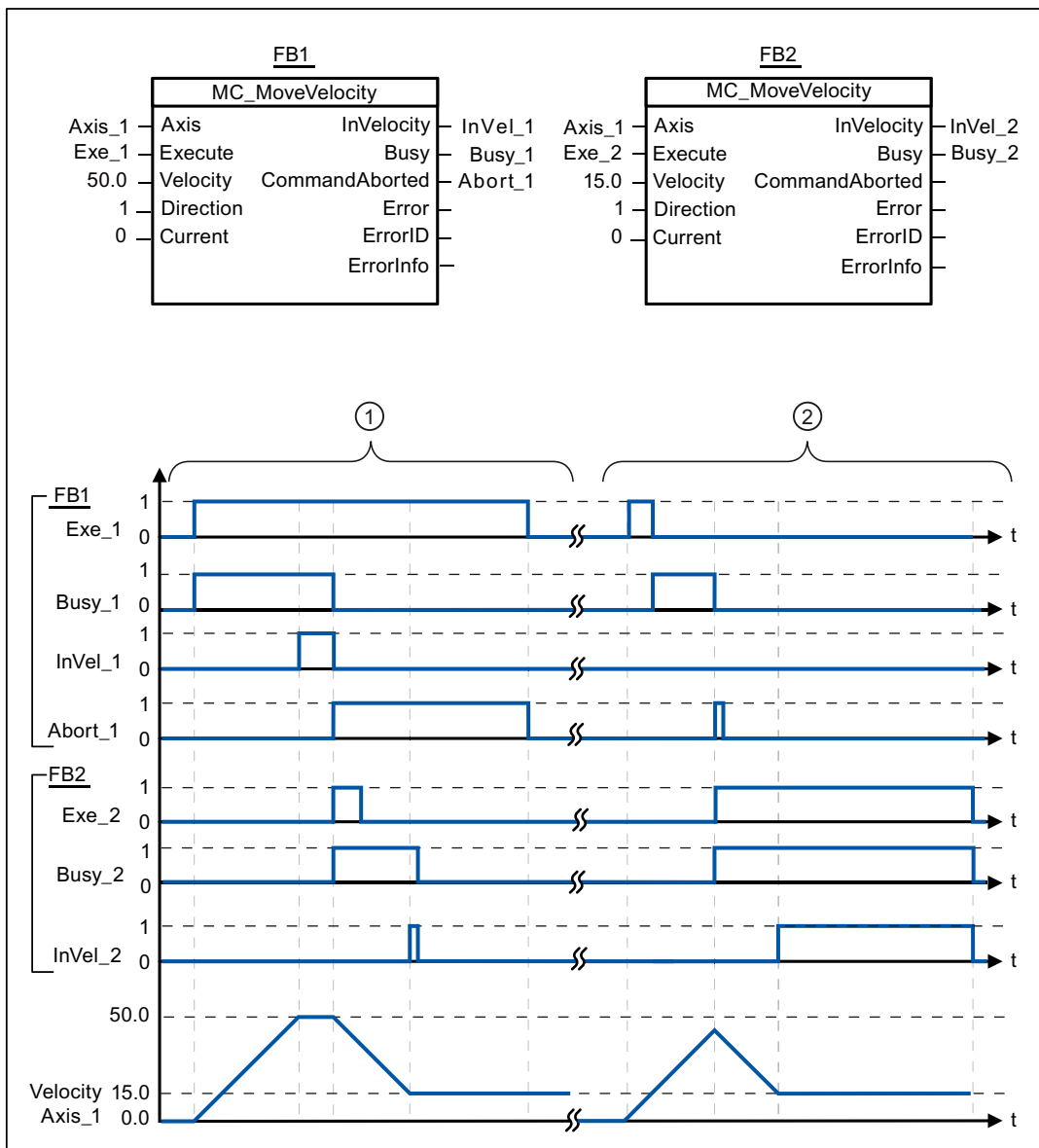
MC_MoveJog: 在点动模式下移动轴 (页 208)

MC_CommandTable: 按移动顺序运行轴作业 (从 V2.0“轴”工艺对象起) (页 212)

MC_ChangeDynamic: 更改轴的动态设置 (从 V2.0“轴”工艺对象起) (页 216)

2.7.2 MC_MoveVelocity: 功能图

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

①	激活的 MC_MoveVelocity 作业通过“InVel_1”发出信号，说明已达到目标速度。该作业之后被另一个 MC_MoveVelocity 作业中止。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。
②	在达到目标速度之前，激活的 MC_MoveVelocity 作业被另一个 MC_MoveVelocity 作业中止。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。

参见

MC_MoveVelocity: 以预设的旋转速度移动轴 (页 203)

2.8 MC_MoveJog

2.8.1 MC_MoveJog: 在点动模式下移动轴

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

- 已正确组态工艺对象“轴”。
- 轴已启用。

超驰响应

可通过下列运动控制作业中止 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

新的 MC_MoveJog 命令可中止下列激活的运动控制作业：

- 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

参数

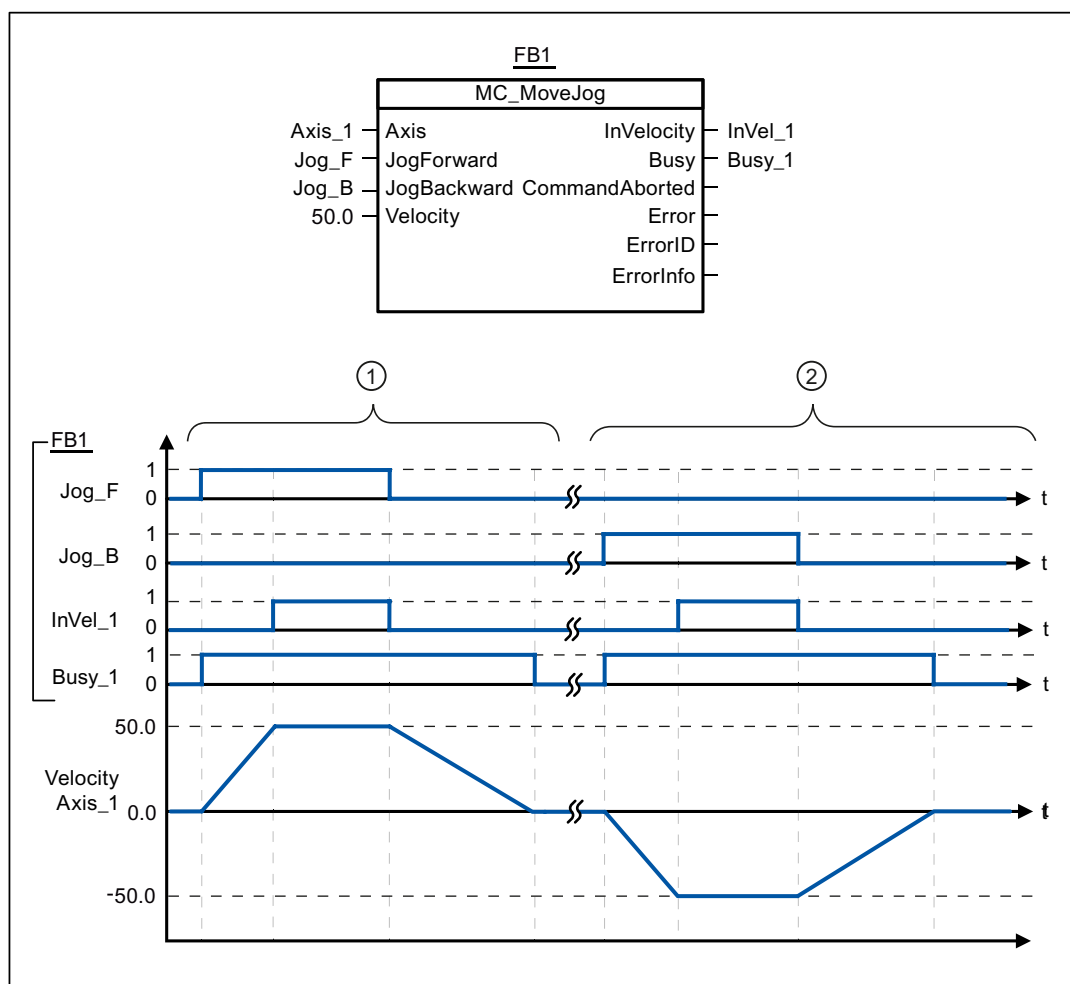
参数	声明	数据类型	默认值	说明
Axis	INPUT	TO_Axis_1	-	轴工艺对象
JogForward	INPUT	BOOL	FALSE	如果参数值为 TRUE，则轴都将按参数“Velocity”中所指定的速度，正向移动。
JogBackward	INPUT	BOOL	FALSE	如果参数值为 TRUE，则轴都将按参数“Velocity”中指定的速度，反向移动。
如果两个参数同时为 TRUE，轴将根据所组态的减速度直至停止。通过参数“Error”、“ErrorID”和“ErrorInfo”，指出了错误。				
Velocity	INPUT	REAL	10.0	点动模式的预设速度
				限值，指令版本 V1.0: 启动/停止速度 ≤ Velocity ≤ 最大速度
				限值，指令版本 V2.0: 启动/停止速度 ≤ 速度 ≤ 最大速度
InVelocity	OUTPUT	BOOL	FALSE	TRUE 达到参数“Velocity”中指定的速度。
Busy	OUTPUT	BOOL	FALSE	TRUE 正在执行命令。
Command Aborted	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 128)
ErrorInfo	OUTPUT	WORD	16#0000	参数“ErrorID”的错误信息 ID (页 128)

参见

- MC_MoveJog: 功能图 (页 211)
- ErrorID 和 ErrorInfo 列表 (从 V2.0 的工艺对象起) (页 128)
- MC_Power: 启用、禁用轴 (页 177)
- MC_Reset: 确认错误 (页 182)
- MC_Home: 归位轴, 设置归位位置 (页 185)
- MC_Halt: 停止轴 (页 189)
- MC_MoveAbsolute: 轴的绝对定位 (页 193)
- MC_MoveRelative: 轴的相对定位 (页 198)
- MC_MoveVelocity: 以预设的旋转速度移动轴 (页 203)
- MC_CommandTable: 按移动顺序运行轴作业 (从 V2.0“轴”工艺对象起) (页 212)
- MC_ChangeDynamic: 更改轴的动态设置 (从 V2.0“轴”工艺对象起) (页 216)

2.8.2 MC_MoveJog: 功能图

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

①	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“InVelo_1”发出信号。在重置了 Jog_F 之后，轴便经过制动达到停止状态。
②	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“InVelo_1”发出信号。在重置了 Jog_B 之后，轴便经过制动达到停止状态。

参见

MC_MoveJog: 在点动模式下移动轴 (页 208)

2.9 MC_CommandTable

2.9.1 MC_CommandTable: 按移动顺序运行轴作业 (从 V2.0“轴”工艺对象起)

说明

运动控制指令“MC_CommandTable”可将多个单独的轴控制作业组合到一个运动顺序中。

要求

- 已添加 V2.0 版“轴”工艺对象并正确组态。
- 已添加工艺对象“命令表”并正确组态。
- 轴已启用。

超驰响应

可通过下列运动控制作业中止 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作业：

- 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_Axis_1	-	轴工艺对象
Command Table	INPUT	TO_Command Table_1	-	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。
Command Aborted	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 128)
ErrorInfo	OUTPUT	WORD	16#0000	参数“ErrorID”的错误信息 ID (页 128)
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

- ErrorID 和 ErrorInfo 列表（从 V2.0 的工艺对象起）(页 128)
- 运动控制语句概述 (页 84)
- MC_Power: 启用、禁用轴 (页 177)
- MC_Reset: 确认错误 (页 182)
- MC_Home: 归位轴，设置归位位置 (页 185)
- MC_Halt: 停止轴 (页 189)
- MC_MoveAbsolute: 轴的绝对定位 (页 193)
- MC_MoveRelative: 轴的相对定位 (页 198)
- MC_MoveVelocity: 以预设的旋转速度移动轴 (页 203)
- MC_MoveJog: 在点动模式下移动轴 (页 208)
- MC_ChangeDynamic: 更改轴的动态设置（从 V2.0“轴”工艺对象起）(页 216)

2.10 MC_ChangeDynamic

2.10.1 MC_ChangeDynamic: 更改轴的动态设置（从 V2.0“轴”工艺对象起）

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

该变量 (页 140)的说明中显示了更改所产生的影响。

要求

- 已添加 V2.0 版“轴”工艺对象。
- 已正确组态工艺对象“轴”。

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作业。

参数

参数	声明	数据类型	默认值	说明
Axis	INPUT	TO_Axis_1	-	轴工艺对象
Execute	INPUT	BOOL	FALSE	上升沿时启动命令
Change RampUp	INPUT	BOOL	FALSE	TRUE 按照输入参数“RampUpTime”更改加速时间
RampUp Time	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）。 更改将影响变量 <轴名称>。 Config.DynamicDefaults.Acceleration。该变量的说明中显示了更改所产生的影响。
Change RampDown	INPUT	BOOL	FALSE	TRUE 按照输入参数“RampDownTime”更改减速时间
Ramp DownTime	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）。 更改将影响变量 <轴名称>。 Config.DynamicDefaults.Deceleration。该变量的说明中显示了更改所产生的影响。
Change Emergency	INPUT	BOOL	FALSE	TRUE 按照输入参数“EmergencyRampTime”更改急停减速时间
Emergency RampTime	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）。 更改将影响变量 <轴名称>。 Config.DynamicDefaults.EmergencyDeceleration。该变量的说明中显示了更改所产生的影响。
ChangeJerk Time	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） 更改将影响变量 <轴名称>。 Config.DynamicDefaults.Jerk。该变量的说明中显示了更改所产生的影响。

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 128)
ErrorInfo	OUTPUT	WORD	16#0000	参数“ErrorID”的错误信息 ID (页 128)

说明

在输入参数“RampUpTime”、“RampDownTime”、“EmergencyRampTime”和“JerkTime”中输入的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)。请注意“轴工艺对象”(Axis technology object) ->“组态工艺对象”(Configuring the technology object) ->“动态”(Dynamics) 中的方程和限值并确保输入的值处于有效范围内。

参见

ErrorID 和 ErrorInfo 列表（从 V2.0 的工艺对象起）(页 128)

运动控制语句概述 (页 84)

在用户程序中可以更改动态组态 (页 49)

在用户程序中更改回原点组态 (页 55)

MC_Power: 启用、禁用轴 (页 177)

MC_Reset: 确认错误 (页 182)

MC_Home: 归位轴，设置归位位置 (页 185)

MC_Halt: 停止轴 (页 189)

MC_MoveAbsolute: 轴的绝对定位 (页 193)

MC_MoveRelative: 轴的相对定位 (页 198)

MC_MoveVelocity: 以预设的旋转速度移动轴 (页 203)

MC_MoveJog: 在点动模式下移动轴 (页 208)

MC_CommandTable: 按移动顺序运行轴作业（从 V2.0“轴”工艺对象起）(页 212)

轴工艺对象的变量 (页 140)

索引

M

MC_ChangeDynamic: 参数, 217
MC_ChangeDynamic: 指令, 216
MC_CommandTable: 参数, 214
MC_CommandTable: 指令, 212
MC_Halt: 功能图, 191
MC_Halt: 参数, 190
MC_Halt: 指令, 189
MC_Home: 参数, 187
MC_Home: 指令, 185
MC_MoveAbsolute: 功能图, 196
MC_MoveAbsolute: 参数, 194
MC_MoveAbsolute: 指令, 193
MC_MoveJog: 功能图, 211
MC_MoveJog: 参数, 209
MC_MoveJog: 指令, 208
MC_MoveRelative: 功能图, 201
MC_MoveRelative: 参数, 199
MC_MoveRelative: 指令, 198
MC_MoveVelocity: 功能图, 206
MC_MoveVelocity: 参数, 204
MC_MoveVelocity: 指令, 203
MC_Power: 参数, 178
MC_Power: 指令, 177
MC_Power: 功能图, 181
MC_Reset: 指令, 182
MC_Reset: 参数, 183

T

TO_Axis_PTO, 32

三划

工艺对象、轴和命令表: ErrorID 和 ErrorInfo 列表, 128
工艺对象命令表: 快捷菜单命令, 65
工艺对象命令表: 命令表组态, 60
工艺对象命令表: 组态下一步, 63
工艺对象命令表: 组态位置/行进路径, 62
工艺对象命令表: 组态步代码, 64
工艺对象命令表: 组态使用轴参数, 61
工艺对象命令表: 组态命令类型, 61
工艺对象命令表: 组态持续时间, 63
工艺对象命令表: 组态速度, 62
工艺对象命令表: 组态窗口图标, 59
工艺对象命令表: 组态激活警告, 60
工艺对象命令表: 常规组态, 60
工艺对象命令表: 添加新对象, 58
工艺对象命令表: 工具, 57
工艺对象命令表: 用途, 57
工艺对象命令表: 扩展参数, 59
工艺对象命令表: 基本参数, 59

四划

手册
 用途, 3
 适用范围, 3
方向输出和行进方向: 关系, 17

五划

可控制驱动器的最大数目, 11
用途
 手册, 3

六划

- 冲击限制器：功能, 20
- 回原点：回原点模式, 21

七划

- 伺服电机, 10
- 步进电机, 10
- 运动控制 CPU S7-1200
 - 准则, 22
- 运动控制 S7-1200 的硬件配置, 10
- 驱动器信号, 15

九划

- 脉冲和方向输出, 13
- 脉冲接口：原理, 16
- 脉冲输出的频率范围, 11
- 轴工艺对象： Config.DriveInterface. 变量, 142
- 轴工艺对象： Config.DynamicDefaults 变量, 145
- 轴工艺对象： Config.DynamicLimits 变量, 144
- 轴工艺对象： Config.General. 变量, 140
- 轴工艺对象： Config.Homing 变量, 152
- 轴工艺对象： Config.Mechanics 变量, 143
- 轴工艺对象： Config.PositionLimits_HW 变量, 149
- 轴工艺对象： Config.PositionLimits_SW 变量, 147
- 轴工艺对象： ErrorBits 变量, 160
- 轴工艺对象： MotionStatus 变量, 155
- 轴工艺对象： PTO 和 HSC 组态, 34
- 轴工艺对象： StatusBits 变量, 156
- 轴工艺对象： 用户单位组态, 36
- 轴工艺对象： 在用户程序中可以更改动态组态参数, 49
- 轴工艺对象： 在用户程序中可以更改回原点组态参数, 55
- 轴工艺对象： 机械组态, 37
- 轴工艺对象： 启用冲击限制器时的响应, 47

- 轴工艺对象： 更新变量, 163
- 轴工艺对象： 诊断概述, 31
- 轴工艺对象： 驱动器使能组态, 36
- 轴工艺对象： 驱动器信号组态, 36
- 轴工艺对象： 驱动器准备就绪组态, 36
- 轴工艺对象： 组态允许在硬件限位开关处自动反向, 51
- 轴工艺对象： 组态反转方向信号, 37
- 轴工艺对象： 组态主动回原点, 51
- 轴工艺对象： 组态加速时间, 44
- 轴工艺对象： 组态加速度, 44
- 轴工艺对象： 组态平滑时间, 45
- 轴工艺对象： 组态电机每转的脉冲数, 37
- 轴工艺对象： 组态电机每转的距离, 37
- 轴工艺对象： 组态冲击限制器, 44
- 轴工艺对象： 组态回原点位置偏移, 52
- 轴工艺对象： 组态回原点速度, 52
- 轴工艺对象： 组态参考点开关侧, 50, 51
- 轴工艺对象： 组态参考点开关输入, 50
- 轴工艺对象： 组态参考点位置, 51, 52
- 轴工艺对象： 组态被动回原点, 50
- 轴工艺对象： 组态速度限值单位, 43
- 轴工艺对象： 组态减速时间, 44
- 轴工艺对象： 组态减速度, 44
- 轴工艺对象： 组态最大速度/启动/停止速度, 43
- 轴工艺对象： 组态窗口图标, 33
- 轴工艺对象： 组态逼近/回原点方向, 51
- 轴工艺对象： 组态概述, 31
- 轴工艺对象： 急停减速度组态, 46
- 轴工艺对象： 轴名称组态, 34
- 轴工艺对象： 调试概述, 31
- 轴工艺对象： 常规动态组态, 43
- 轴工艺对象： 添加新对象, 32
- 轴工艺对象： 硬件和软件组件, 26
- 轴工艺对象： 硬件接口组态, 34
- 轴工艺对象： 工具, 30
- 轴工艺对象： 主动回原点, 54

轴工艺对象: 扩展参数, 33
轴工艺对象: 被动回原点, 53
轴工艺对象: 基本参数, 33
适用范围
 手册, 3

十一划

基本知识
 需要的, 3
添加分隔符行 (Add separator line), 65
添加空行 (Add empty line), 65

十二划

插入分隔符行 (Insert separator line), 65
插入空行 (Insert empty line), 65
硬件和软件限位开关: 功能, 19

十四划

需要的基本知识, 3

