

SIEMENS

SIMATIC

工程工具
S7-Technology

功能手册

前言	
S7-Technology 中的新增内容	1
应用和优势	2
工艺对象	3
组态	4
编程	5
工艺功能	6
工艺数据块	7
装载、测试和诊断	8
附录	9

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

仅允许安装和驱动与本文件相关的附属设备或系统。设备或系统的调试和运行仅允许由**合格的专业人员**进行。本文件安全技术提示中的合格专业人员是指根据安全技术标准具有从事进行设备、系统和电路的运行，接地和标识资格的人员。

按规定使用 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地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

本手册用途

本手册完整地概述了“S7-Technology”选件包。它根据 PLCopen 阐明了编程模型、各个工艺对象和各个功能块。

其目标读者为编写 STEP 7 程序的程序员以及利用运动控制应用程序对自动化系统进行组态、调试和维修的工作人员。

需要的基本知识

要理解本手册，您需具备自动化工程和运动控制领域的基本知识。

同时，还必须能够在 Windows 2000 或 XP PC 操作系统下熟练使用计算机或编程设备。作为选件包，

“S7-Technology”是以 STEP 7 基本软件为基础，您还应了解如何处理 STEP 7 基本软件。“使用 STEP 7 编程”手册中介绍了这方面内容。

本手册适用范围

本手册适用于版本为 V 4.1 或更高版本的“S7-Technology”选件包。

“S7-Technology V 4.1 中有哪些新增内容？”一节中介绍了

与 V 1.0 版本的“S7-Technology”选件包之间在功能和数量框架方面的差异。

其它支持

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

可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.de/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.ad.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。请与当地培训中心或位于德国纽伦堡 **D 90327** 的培训中心总部联系。

目录

前言	3
1 S7-Technology 中的新增内容.....	19
2 应用和优势.....	21
2.1 紧凑型和集成式	21
2.2 “SIMATIC”领域	21
2.3 STEP 7 中符合 PLCopen 的运动控制功能的集成	22
2.4 支持的组件和系统.....	22
2.5 组件及其用途.....	27
2.5.1 硬件组件.....	27
2.5.2 组态工具.....	28
2.5.3 工艺功能和工艺 DB	31
3 工艺对象.....	33
3.1 轴 — 基本知识.....	33
3.1.1 轴工艺	33
3.1.2 特定的操作模式	34
3.1.3 轴类型	34
3.1.4 液压轴	35
3.1.5 实轴和虚拟轴.....	38
3.1.6 “轴”和“驱动器”之间的区别.....	39
3.1.7 基准坐标系 — 叠加坐标系	40
3.1.8 回原点	42
3.1.9 绝对编码器调整	44
3.1.10 数据记录转换.....	45
3.2 “速度控制轴”工艺对象.....	46
3.2.1 “速度控制轴”工艺对象.....	46
3.3 “定位轴”工艺对象.....	47
3.3.1 “定位轴”工艺对象.....	47
3.3.2 位置控制.....	48
3.3.2.1 位置控制.....	48
3.3.2.2 设定值叠加	52
3.3.2.3 动态响应调整.....	53
3.3.2.4 准备电气轴的受控变量	54
3.3.2.5 操纵变量叠加.....	55
3.3.2.6 动态伺服控制 (DSC, Dynamic Servo Control)	56
3.3.2.7 位置控制轴的速度控制操作	57

3.4	“同步轴”工艺对象.....	57
3.4.1	“同步轴”工艺对象.....	57
3.4.2	“同步轴”工艺对象的结构.....	58
3.4.3	同步组	58
3.4.4	齿轮传动.....	62
3.4.5	凸轮传动.....	64
3.4.6	叠加同步.....	69
3.4.7	相对/绝对同步.....	71
3.5	“凸轮盘”工艺对象.....	75
3.5.1	“凸轮盘”工艺对象.....	75
3.5.2	缩放.....	76
3.5.3	凸轮盘应用	77
3.5.4	凸轮插补.....	78
3.5.5	插补类型.....	81
3.6	“输出凸轮”工艺对象	89
3.6.1	“输出凸轮”工艺对象	89
3.6.2	基于位置的凸轮	91
3.6.3	切换输出凸轮.....	92
3.6.4	基于时间的凸轮	93
3.6.5	有效方向和特性	94
3.6.6	滞后	96
3.6.7	凸轮切换点的基于时间的偏移.....	98
3.6.8	电子凸轮控制实例	99
3.7	“凸轮轨迹”工艺对象	100
3.7.1	“凸轮轨迹”工艺对象	100
3.7.2	基于位置的凸轮	103
3.7.3	基于时间的凸轮	104
3.7.4	具有最大 ON 长度的基于时间的输出凸轮	105
3.7.5	循环和非循环创建凸轮轨迹	105
3.8	“测量输入”工艺对象	107
3.8.1	“测量输入”工艺对象	107
3.8.2	测量输入的互连和连接	108
3.8.3	测量范围.....	109
3.9	“外部编码器”工艺对象	110
3.9.1	“外部编码器”工艺对象	110
3.9.2	外部编码器的互连和连接.....	110
3.9.3	外部编码器的同步	111

4	组态	113
4.1	组态基本步骤	113
4.2	在 HW Config 中组态 T-CPU 和驱动器	114
4.2.1	在 HW Config 中组态 T-CPU	114
4.2.2	在 HW Config 中组态驱动器	118
4.2.3	消息帧类型及其功能	122
4.3	工艺对象管理	125
4.3.1	启动工艺对象管理	125
4.3.2	工艺对象管理的用户界面	126
4.3.3	使用工艺对象管理	127
4.3.4	创建和管理工艺 DB	128
4.4	使用 S7T Config	130
4.4.1	启动 S7T Config	130
4.4.2	S7T Config 的用户界面	131
4.5	组态电气轴	134
4.5.1	组态电气轴	134
4.5.2	添加数据记录以进行数据记录转换	142
4.5.3	高精度	144
4.5.4	实际值记录	145
4.5.5	组态	147
4.5.5.1	组态 — 轴	147
4.5.6	机械系统	148
4.5.6.1	机械 — 电气轴	148
4.5.7	默认值	151
4.5.7.1	缺省 — “动态”(Dynamics) 标签	151
4.5.8	限制	153
4.5.8.1	限制 — “位置和速度”(Position and velocity) 标签	153
4.5.8.2	硬件限位开关	154
4.5.8.3	软件限位开关	156
4.5.8.4	限制 — “动态响应”(Dynamic response) 标签	157
4.5.8.5	限制 — “固定挡块”(Fixed end stop) 标签	159
4.5.9	实际值	161
4.5.9.1	实际值 (Actual value) — “实际值”(Actual value) 选项卡	161
4.5.9.2	实际值 (Actual value) — “推断”(Extrapolation) 选项卡	162
4.5.10	控制	166
4.5.10.1	控制 — “静态控制器数据”(Static controller data) 标签	166
4.5.10.2	控制 — “动态控制器数据”(Dynamic controller data) 标签	168
4.5.10.3	控制 — “摩擦补偿”(Friction compensation) 标签	169
4.5.11	回原点	170
4.5.11.1	引言	170
4.5.11.2	回原点 — “主动回原点”(Active homing) 标签	170
4.5.11.3	回原点 — “被动回原点”(Passive homing) 标签	178

4.5.11.4	被动回原点模式的定位特性	183
4.5.11.5	直接回原点	185
4.5.11.6	位置校正	185
4.5.11.7	未回原点的轴的运动	185
4.5.12	监视功能	186
4.5.12.1	监视功能 — 总览	186
4.5.12.2	监视功能 — 定位和停止监视	187
4.5.12.3	监视功能 — “跟踪误差监视”(Following error monitoring) 标签	188
4.5.12.4	监视功能 — “停止信号”(Standstill signal) 标签	189
4.5.12.5	监视功能 — “速度监视”(Velocity monitoring) 标签	190
4.5.12.6	操纵变量监视	190
4.6	组态液压轴	191
4.6.1	组态液压轴 — 插入轴	191
4.6.2	组态液压轴 — 通过 IM 174/ADI4 的 Q 输出	196
4.6.3	组态液压轴 — 通过模拟输出模块的 Q 输出	202
4.6.4	确定并添加一个阀轮廓	206
4.6.5	分配阀轮廓	211
4.6.6	添加数据记录以进行数据记录转换	212
4.6.7	实际值记录	215
4.6.8	组态	218
4.6.8.1	组态 — 轴	218
4.6.9	机械系统	219
4.6.9.1	机械 — 液压轴	219
4.6.10	默认值	220
4.6.10.1	缺省 — “动态”(Dynamics) 标签	220
4.6.11	限制	222
4.6.11.1	限制 — “位置和速度”(Position and velocity) 标签	222
4.6.11.2	硬件限位开关	223
4.6.11.3	软件限位开关	225
4.6.11.4	限制 — “动态响应”(Dynamic response) 标签	226
4.6.11.5	限制 — “固定挡块”(Fixed end stop) 标签	228
4.6.12	实际值	230
4.6.12.1	实际值 (Actual value) — “实际值”(Actual value) 选项卡	230
4.6.12.2	实际值 (Actual value) — “推断”(Extrapolation) 选项卡	231
4.6.13	控制	235
4.6.13.1	控制 — “静态控制器数据”(Static controller data) 标签	235
4.6.13.2	控制 — “动态控制器数据”(Dynamic controller data) 标签	236
4.6.13.3	控制 — “摩擦补偿”(Friction compensation) 标签	237
4.6.13.4	控制 — “附加补偿功能”(Additional compensation functions) 标签	238
4.6.14	回原点	239
4.6.14.1	引言	239
4.6.14.2	回原点 — “主动回原点”(Active homing) 标签	239
4.6.14.3	回原点 — “被动回原点”(Passive homing) 标签	248
4.6.14.4	被动回原点模式的定位特性	253

4.6.14.5	直接回原点	255
4.6.14.6	位置校正	255
4.6.14.7	未回原点的轴的运动	255
4.6.15	监视功能	256
4.6.15.1	监视功能 — 总览	256
4.6.15.2	监视功能 — 定位和停止监视	257
4.6.15.3	监视功能 — “跟踪误差监视”(Following error monitoring) 标签	258
4.6.15.4	监视功能 — “停止信号”(Standstill signal) 标签	259
4.6.15.5	监视功能 — “速度监视”(Velocity monitoring) 标签	260
4.7	组态同步轴	261
4.7.1	组态同步轴	261
4.7.2	分配引导轴和凸轮盘	263
4.7.3	组态叠加同步	264
4.7.4	同步	267
4.7.5	同步	274
4.7.6	去同步	278
4.7.7	同步状态	280
4.7.8	组态	281
4.7.8.1	组态 — 轴	281
4.7.9	机械系统	282
4.7.9.1	机械 — 电气轴	282
4.7.10	默认值	285
4.7.10.1	缺省 — “动态”(Dynamics) 标签	285
4.7.11	限制	287
4.7.11.1	限制 — “位置和速度”(Position and velocity) 标签	287
4.7.11.2	硬件限位开关	288
4.7.11.3	软件限位开关	290
4.7.11.4	限制 — “动态响应”(Dynamic response) 标签	291
4.7.11.5	限制 — “固定挡块”(Fixed end stop) 标签	293
4.7.12	实际值	295
4.7.12.1	实际值 (Actual value) — “实际值”(Actual value) 选项卡	295
4.7.12.2	实际值 (Actual value) — “推断”(Extrapolation) 选项卡	296
4.7.13	控制	300
4.7.13.1	控制 — “静态控制器数据”(Static controller data) 标签	300
4.7.13.2	控制 — “动态控制器数据”(Dynamic controller data) 标签	302
4.7.13.3	控制 — “摩擦补偿”(Friction compensation) 标签	303
4.7.14	回原点	304
4.7.14.1	引言	304
4.7.14.2	回原点 — “主动回原点”(Active homing) 标签	304
4.7.14.3	回原点 — “被动回原点”(Passive homing) 标签	313
4.7.14.4	被动回原点模式的定位特性	318
4.7.14.5	直接回原点	320
4.7.14.6	位置校正	320
4.7.14.7	未回原点的轴的运动	320

4.7.15	监视功能.....	321
4.7.15.1	监视功能.....	321
4.7.15.2	监视功能 — 总览.....	321
4.7.15.3	监视功能 — 定位和停止监视.....	323
4.7.15.4	监视功能 —“跟踪误差监视”(Following error monitoring) 标签.....	324
4.7.15.5	监视功能 —“停止信号”(Standstill signal) 标签.....	325
4.7.15.6	监视功能 —“同步运行监视”(Synchronous operation monitoring) 标签.....	325
4.7.15.7	监视功能 —“速度监视”(Velocity monitoring) 标签.....	327
4.7.16	跟随对象.....	328
4.7.16.1	组态.....	328
4.7.16.2	设置.....	329
4.7.16.3	默认设置.....	331
4.8	凸轮组态.....	339
4.8.1	凸轮组态.....	339
4.8.2	插入凸轮.....	339
4.8.3	定义凸轮.....	342
4.8.4	使用 CamEdit 创建凸轮.....	343
4.8.4.1	使用 CamEdit 创建凸轮.....	343
4.8.4.2	插补.....	344
4.8.4.3	插补(2).....	346
4.8.4.4	缩放和移位.....	347
4.8.5	使用 CamTool 创建凸轮.....	355
4.8.6	符合 VDI 的运动法则.....	356
4.8.6.1	工作范围和运动转换.....	356
4.8.6.2	用区段定义凸轮盘.....	358
4.9	组态输出凸轮.....	359
4.9.1	组态输出凸轮.....	359
4.9.2	插入输出凸轮.....	359
4.9.3	组态 — 输出凸轮.....	361
4.9.4	高速输出凸轮.....	363
4.10	组态凸轮轨迹.....	363
4.10.1	组态凸轮轨迹.....	363
4.10.2	插入凸轮轨迹.....	364
4.10.3	组态 — 凸轮轨迹.....	366
4.10.4	默认值 — 凸轮轨迹.....	368
4.11	组态测量输入.....	371
4.11.1	组态测量输入.....	371
4.11.2	插入测量输入.....	371
4.11.3	组态 — 测量输入.....	373
4.12	组态外部编码器.....	375
4.12.1	组态外部编码器.....	375
4.12.2	插入外部编码器.....	375

4.12.3	缺省值 — 外部编码器	383
4.12.4	外部编码器 — 与增量编码器同步	384
4.12.5	设置停止信号	386
4.12.6	组态	386
4.12.6.1	组态 — 外部编码器	386
4.12.7	机械	387
4.12.7.1	机械 — 外部编码器	387
4.12.8	实际值	388
4.12.8.1	实际值 (Actual value) —“实际值”(Actual value) 选项卡	388
4.12.8.2	实际值 (Actual value) —“推断”(Extrapolation) 选项卡	389
4.12.9	回原点	393
4.12.9.1	回原点 —“被动回原点”(Passive homing) 标签	393
4.13	从另一个站中复制组态数据	399
5	编程	401
5.1	CPU 31xT 的时间图	401
5.2	WinLC T (MICROBOX T) 的时序	406
5.3	T-CPU 的周期	412
5.4	分配系统时钟	414
5.5	顺序模型和编程模型	416
5.6	访问 CPU 31xT 的地址空间	418
5.7	访问 MICROBOX T 的地址空间	420
5.8	监视活动命令	422
5.9	在 IPO 同步模式下启动轴命令	424
5.10	工艺功能处的错误和警告	425
5.11	工艺 DB (MCDevice 和 Trace) 中的错误	426
5.12	工艺 DB (Axes 和 External Encoders) 中的错误和警告	427
5.13	工艺 DB (凸轮盘、测量输入和输出凸轮) 中的错误和警告	428
5.14	虚拟轴的响应	429
5.15	使用 FC400“DB2INT”进行符号编程	430
5.16	对轴特定的参数更改进行编程	431
6	工艺功能	433
6.1	总览	433
6.1.1	总览 — 编程	433
6.1.2	按编号排序的工艺功能列表	436
6.1.3	按字母顺序排序的工艺功能列表	438

6.2	工艺功能 — 单轴	439
6.2.1	FB401 MC_Power — 禁用/启用轴	439
6.2.2	FB403 MC_Home — 使轴回原点/设置轴	452
6.2.3	FB404 MC_Stop — 停止轴并阻止新的运动命令	464
6.2.4	FB405 MC_Halt — 正常停止	470
6.2.5	FB409 MC_ChangeDataset — 数据记录转换	475
6.2.6	FB410 MC_MoveAbsolute — 绝对定位	481
6.2.7	FB411 MC_MoveRelative — 相对定位	493
6.2.8	FB412 MC_MoveAdditive — 相对定位到当前目标位置	504
6.2.9	FB413 MC_MoveSuperImposed — 叠加定位	510
6.2.10	FB 414 MC_MoveVelocity — 预设了速度的运动	518
6.2.11	FB 415 MC_MoveToEndPos — 移动到固定挡块/紧固件	528
6.2.12	FB437 MC_SetTorqueLimit — 启用/禁用扭矩限制	536
6.2.13	FB439 MC_SetCharacteristic — 激活阀控曲线	541
6.3	工艺功能 — 齿轮传动/凸轮传动	545
6.3.1	FB420 MC_GearIn — 启动齿轮传动	545
6.3.2	FB422 MC_GearOut — 终止齿轮传动	556
6.3.3	FB440 MC_GearInSuperImposed — 启动叠加齿轮传动	561
6.3.4	FB442 MC_GearOutSuperImposed — 停止叠加齿轮传动	574
6.3.5	FB421“MC_CamIn”— 启动凸轮传动	578
6.3.6	FB423 MC_CamOut — 停止凸轮传动	594
6.3.7	FB 441 MC_CamInSuperImposed — 启动叠加凸轮传动	599
6.3.8	FB 443 MC_CamOutSuperImposed — 终止叠加凸轮传动	610
6.3.9	FB 424 MC_Phasing — 更改引导轴和跟随轴之间的相移	615
6.3.10	FB 444 MC_PhasingSuperImposed — 更改叠加相移	622
6.4	工艺功能 — 凸轮盘	631
6.4.1	FB 434 MC_CamClear — 删除凸轮	631
6.4.2	FB 435 MC_CamSectorAdd — 添加凸轮扇区	635
6.4.3	FB 436 MC_CamInterpolate — 插补凸轮	645
6.4.4	FB 438 MC_GetCamPoint — 从凸轮读取点	652
6.5	工艺功能 — 输出凸轮，凸轮轨迹	659
6.5.1	FB 430 MC_CamSwitch — 基于位置的凸轮	659
6.5.2	FB 431 MC_CamSwitchTime — 基于时间的凸轮	667
6.5.3	FB 461 MC_CamTrack — 凸轮轨迹	673
6.5.4	FB 462 MC_ReadCamTrackData — 读出凸轮轨迹	681
6.5.5	FB 463 MC_WriteCamTrackData — 写入凸轮轨迹	686
6.6	工艺功能 — 测量输入，外部编码器	691
6.6.1	FB433 MC_MeasuringInput — 测量输入	691
6.6.2	FB432 MC_ExternalEncoder — 外部编码器	697
6.7	工艺功能 — 基本功能	704
6.7.1	FB402 MC_Reset — 确认错误	704
6.7.2	FB 460 MC_ActivateTO — 取消激活/激活工艺对象	709

6.7.3	FB 457 MC_ActivateDP Slave — 取消激活/激活 DP 从站.....	714
6.7.4	FB406 MC_ReadSysParameter — 读取参数	719
6.7.5	FB407 MC_WriteParameter — 更改参数	723
6.7.6	FB450 MC_ReadPeriphery — 读取工艺 I/O	733
6.7.7	FB451 MC_WritePeriphery — 写入工艺 I/O	739
6.7.8	FB453 MC_ReadRecord — 读取数据记录	745
6.7.9	FB454 MC_WriteRecord — 写入数据记录	751
6.7.10	FB455 MC_ReadDriveParameter — 读取驱动参数.....	757
6.7.11	FB456 MC_WriteDriveParameter — 写入驱动参数.....	764
6.8	命令的相互作用	771
6.8.1	新命令 — 活动的单个命令 (1)	771
6.8.2	新命令 — 活动的单个命令 (2)	773
6.8.3	新命令 — 活动的单个命令 (3)	775
6.8.4	新命令 — 活动的多个命令 (1)	776
6.8.5	新命令 — 活动的多个命令 (2)	778
6.8.6	使用表格的实例	780
6.9	有关参数的信息	782
6.9.1	断电和重启后工艺功能的响应.....	782
6.9.2	生成 DoneFlag	782
6.9.3	值范围	783
6.9.4	模数轴的绝对位置.....	784
7	工艺数据块.....	785
7.1	“速度控制轴”工艺 DB.....	785
7.2	“定位轴”工艺 DB.....	791
7.3	“同步轴”工艺 DB.....	798
7.4	“外部编码器”工艺 DB.....	807
7.5	“凸轮盘”工艺 DB.....	811
7.6	“测量输入”工艺 DB	813
7.7	“输出凸轮”工艺 DB	816
7.8	“凸轮轨迹”工艺 DB	820
7.9	“跟踪”工艺 DB	825
7.10	“MCDevice”工艺 DB	826
7.11	更新工艺 DB.....	829
7.12	ErrorID — 工艺 DB.....	835
7.12.1	ErrorID — 轴工艺 DB	835
7.12.2	ErrorID — 同步工艺 DB.....	843
7.12.3	ErrorID — 外部编码器工艺 DB	857
7.12.4	ErrorID — 凸轮工艺 DB.....	864

7.12.5	ErrorID — 测量输入工艺 DB	869
7.12.6	ErrorID — 输出凸轮工艺 DB	873
7.12.7	ErrorID — 凸轮轨迹工艺 DB	878
7.12.8	ErrorID — MCDevice/Trace 工艺 DB	882
8	装载、测试和诊断	883
8.1	连接编程设备	883
8.2	下载命令	884
8.2.1	STEP 7 中的装载命令	884
8.2.2	S7T Config 中的装载命令 — 存储器组织	886
8.2.2.1	集成工艺的存储器组织	886
8.2.2.2	接通电源和 CPU 存储器复位	888
8.2.2.3	重启工艺对象	889
8.2.2.4	S7T Config 中的装载命令	890
8.2.2.5	从 Actual 复制到 ROM	893
8.2.2.6	S7T Config 中的在线更改	894
8.3	工艺系统时钟	897
8.3.1	设置工艺系统周期时钟	897
8.3.2	检查集成工艺上的负载	900
8.4	优化位置控制器	901
8.4.1	优化位置控制器 — 概述	901
8.4.2	准备调谐	901
8.4.3	用于控制器调谐的用户程序	905
8.4.4	优化位置控制器	908
8.4.5	调节平衡滤波器	913
8.4.6	计算位置控制环的等效时间常量	918
8.5	使用断点进行测试	920
8.5.1	通过断点测试	920
8.6	使用监视表进行测试	921
8.6.1	使用监视表监视系统变量	921
8.7	将 TraceTool 用于测试	923
8.7.1	总览 — TraceTool	923
8.7.2	跟踪 CPU 用户程序的值	923
8.7.3	跟踪控制器数据	924
8.8	使用控制面板进行测试	927
8.8.1	引言 — 控制面板	927
8.8.2	启动轴控制面板	928
8.8.3	轴控制面板的布局	929
8.8.4	使用控制面板监视轴值	930
8.8.5	采用控制优先级	932
8.8.6	控制轴	933

9	附录	937
9.1	应用实例	937
9.1.1	应用实例“使用目标传感器定位”	937
9.1.2	应用实例“飞剪”	939
9.1.3	应用实例“夹具进给”	941
9.2	问题、提示和技巧	943
9.2.1	列表 — 问题、提示和技巧	943
9.2.2	如何利用新性能特性?	944
9.2.3	如何将 T-CPU 替换为其它类型?	946
9.2.4	如何识别固件版本?	948
9.2.5	如何升级 CPU 31xT-2 DP 的固件?	949
9.2.6	如何升级集成工艺的固件?	950
9.2.7	如何转换工艺?	952
9.2.8	如何转换工艺以及如何升级工艺包?	953
9.2.9	如何重新创建工艺 DB?	957
9.2.10	升级后如何更新工艺功能?	959
9.2.11	升级后需要将哪些内容下载到 PLC?	960
9.2.12	如何使用 S7-Technology 分析早期版本中的项目?	961
9.2.13	如何使用 S7-Technology 更改早期版本中的项目?	963
9.2.14	使用“另存为”并进行重组时,要遵循什么规定?	965
9.2.15	为什么某些系统 DB 具有不同的在线/离线时间戳?	965
9.2.16	STEP 7 报告“存储空间不足”(Insufficient memory space)	966
9.2.17	超时后, CPU 偶尔会跳转到 STOP	966
9.2.18	对工艺数据执行“保存并编译全部”(Save and compile all) 功能时出错	967
9.3	S7T Config 中的专家列表	967
9.3.1	使用专家列表	967
9.3.2	轴设置	970
9.3.2.1	分配插补器周期 2	970
9.3.2.2	扭矩设置	971
9.3.3	监视功能	972
9.3.3.1	编码器监视功能	972
9.3.3.2	编码器限制频率监视	972
9.4	工艺参数	973
9.4.1	工艺参数 0001 至 0999 — 轴	973
9.4.2	工艺参数 1000 到 1999 — 轴	980
9.4.3	工艺参数 2000 至 2999 — 轴	1009
9.4.4	工艺参数 3000 至 3999 — 轴	1029
9.4.5	工艺参数 4000 至 4999 — 轴	1047
9.4.6	工艺参数 5000 至 9999 — 轴	1091
9.4.7	工艺参数 — 外部编码器	1103
9.4.8	工艺参数 — 输出凸轮	1132
9.4.9	工艺参数 — 凸轮轨迹	1135
9.4.10	工艺参数 — 凸轮盘	1142

9.4.11	工艺参数 — 测量输入	1146
9.4.12	DINT 值	1148
9.4.12.1	EnumActiveInactive	1148
9.4.12.2	EnumAxisApproachDirection	1148
9.4.12.3	EnumAxisEncoderAssemblyType	1149
9.4.12.4	EnumAxisExtrapolatedVelocitySwitch	1149
9.4.12.5	EnumAxisFilterMode	1149
9.4.12.6	EnumAxisFineInterpolatorMode	1150
9.4.12.7	EnumAxisHomingMode	1150
9.4.12.8	EnumAxisPassiveApproachDirection	1151
9.4.12.9	EnumAxisPassiveHomingMode	1151
9.4.12.10	EnumAxisTorqueForceReductionGranularity	1152
9.4.12.11	EnumBackLashDiff	1152
9.4.12.12	EnumBackLashType	1152
9.4.12.13	EnumBalanceFilterMode	1153
9.4.12.14	EnumCamInterpolationMode	1153
9.4.12.15	EnumCammingDirection	1154
9.4.12.16	EnumCammingMode	1154
9.4.12.17	EnumCamMode	1155
9.4.12.18	EnumCamPositionMode	1155
9.4.12.19	EnumCamTrackType	1156
9.4.12.20	EnumChangeMode	1156
9.4.12.21	EnumDirection	1157
9.4.12.22	EnumDirectionType	1157
9.4.12.23	EnumErrorReporting	1158
9.4.12.24	EnumFollowingObjectSynchronizeWithLookAhead	1158
9.4.12.25	EnumFollowingObjectSynchronizingDirection	1159
9.4.12.26	EnumForceDirection	1159
9.4.12.27	EnumGearingDirection	1160
9.4.12.28	EnumGearingMode	1160
9.4.12.29	EnumGearingPosToleranceCommandValue	1161
9.4.12.30	EnumGearingType	1161
9.4.12.31	EnumLogicOperation	1162
9.4.12.32	EnumLimitExceededOk	1162
9.4.12.33	EnumMasterMode	1162
9.4.12.34	EnumMeasuredEdge	1163
9.4.12.35	EnumMeasuringRangeMode	1163
9.4.12.36	EnumMountSwitch	1164
9.4.12.37	EnumOutputCamType	1164
9.4.12.38	EnumProfile	1165
9.4.12.39	EnumRecognitionMode	1165
9.4.12.40	EnumSensorState	1166
9.4.12.41	EnumSlaveMode	1166
9.4.12.42	EnumSyncModeCamming	1167

9.4.12.43 EnumSyncModeGearing.....	1168
9.4.12.44 EnumSyncOffModeCamming	1169
9.4.12.45 EnumSyncOffModeGearing	1169
9.4.12.46 EnumSyncOffPositionReference	1170
9.4.12.47 EnumSyncPositionReference	1170
9.4.12.48 EnumSyncProfileReference	1171
9.4.12.49 EnumToActivationModeSetConfigData.....	1171
9.4.12.50 EnumYesNo.....	1172
9.5 Internet 上的附加信息.....	1173
9.5.1 Internet 上的附加信息.....	1173
索引	1175

S7-Technology 中的新增内容

S7-Technology V 4.1 中有哪些新增内容？

灵活的机器概念

是否想从集成工艺的循环处理中临时或永久地删除已组态的工艺对象？是否想要临时或永久地取消激活和删除 DP(DRIVE) 上的 DP 从站？

新工艺功能“MC_ActivateTO”和“MC_ActivateDPSlave”用于激活/取消激活 DP(DRIVE) 上的工艺对象和 DP 从站。另一方面，您可以在 S7T Config 中组态激活/取消激活。

凸轮轨迹

如果想在轴上使用多个相同类型的输出凸轮，请使用新的“凸轮轨迹”工艺对象。

该凸轮轨迹在 S7T Config 中组态并通过调用“MC_CamTrack”工艺功能激活。通过调用“MC_ReadCamTrackData”工艺功能，可以读取专用输出凸轮的切换状态。通过调用“MC_WriteCamTrackData”工艺功能，可以在用户程序中编辑凸轮轨迹的参数。

T-CPU CPU 31xT 的新功能（固件 V2.6）

- PROFIBUS DP 上的等时模式（接口 X1 MPI/DP）
- 在线更新固件（通过网络）
- 将 CPU 复位为出厂状态

有关详细信息，请参考“CPU 31xT”设备手册。

较早的 CPU 可升级为固件版本 V2.6。

新 STARTER 功能

S7-Technology 中集成的 STARTER 功能对应于新 STARTER 版本 V4.1。

制动控制（垂直轴）

利用新版本的“MC_Power”工艺功能控制驱动器的集成制动控制。支持以下驱动器的制动控制：

- SIMODRIVE 611U 通用型
- SINAMICS S120
- MASTERDRIVES 运动控制 Plus

读取凸轮点的一阶导数和二阶导数

如果凸轮盘是在 T-CPU 中生成和插补的，则无法在 S7T Config 中检查插补结果。可在“MC_GetCamPoint”工艺功能的输入参数 *Mode* 中使用新模式值 **3** 和 **4** 来读取凸轮点的一阶导数和二阶导数。利用读取的值，可以确定凸轮点的速度和加速度。

设置用于实际值耦合的推断

设置实际值过滤器和推断以在 S7T Config 的新对话框中进行实际值耦合。

扭矩减小的高精度

SINAMICS 和 SIMODRIVE 驱动器现在支持扭矩减小的高精度。可以将扭矩减小的高精度与工艺功能“MC_SetTorqueLimit”和“MC_MoveToEndPos”一起使用。

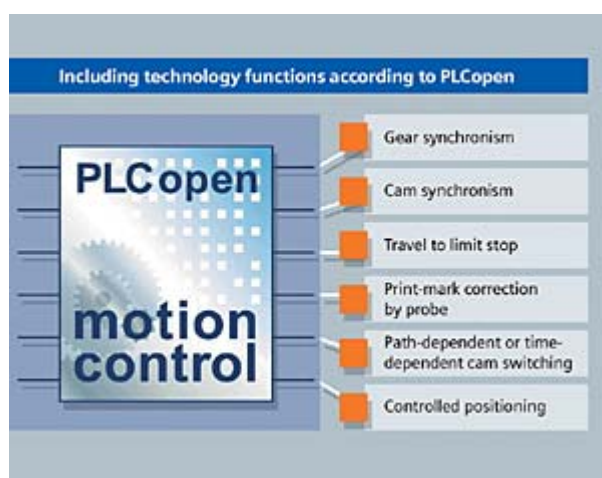
删除使能后扭矩限制保持不变

利用集成工艺 V4.1.x 的当前固件，即使在“MC-Power”工艺功能删除使能后，扭矩限制仍保持不变。

应用和优势

2.1 紧凑型 and 集成式

T-CPU 在 SIMATIC CPU 内集成了运动控制 (Motion Control) 功能，以将 SIMATIC CPU S7-300 的功能与符合 PLCopen 的运动控制功能结合在一起。



T-CPU 已完全集成到 SIMATIC 和 TIA 领域中，并且在用于耦合的运动顺序时展示了它的高性能。

2.2 “SIMATIC”领域

T-CPU 是一个具有集成运动控制功能的标准 SIMATIC CPU。该功能使用户可以轻松地将 S7-300 程序从现有项目复制到新项目中。

基于 SIMATIC 编程语言 LAD、FBD 或 STL 对 T-CPU 进行编程。在单个应用程序中对 PLC 和运动控制功能进行编辑。无须学习任何其它编程语言。可以始终享用您的 SIMATIC S7 知识。

可以在 STEP 7 中轻松组态 T-CPU。在此，可以设置所有必要的参数，例如机械数据、驱动器选择数据、控制器设置、缺省值、监视功能、输出凸轮、测量输入、凸轮盘等。

2.3 STEP 7 中符合 PLCopen 的运动控制功能的集成

T-CPU 符合 PLCopen 的运动控制功能使用户可以直接利用现有的运动控制知识。运动控制功能在接口、功能性和顺序方面符合 PLCopen 规范，方便了工程组态、调试和服务。

有了标准化接口，您几乎可以在应用程序中无缝地应用启动运动控制命令所需的方块。

2.4 支持的组件和系统

SIMATIC T-CPU/软件

功能	产品	订货号
SIMATIC T-CPU	CPU 315T-2DP (工作存储器 128 KB)	6ES7315-6TG10-0AB0
SIMATIC T-CPU	CPU 315T-2DP (工作存储器 256 KB)	6ES7315-6TH13-0AB0
SIMATIC T-CPU	CPU 317T-2DP (工作存储器 512 KB)	6ES7317-6TJ10-0AB0
SIMATIC T-CPU	CPU 317T-2DP (工作存储器 1,024 KB)	6ES7317-6TK13-0AB0
SIMATIC T-CPU	MICROBOX 420-T	6ES7675-3AG30-0AB0
MMC 卡 (适用于 CPU 31xT-2DP)	MMC 4 MB 或 8 MB (建议 8 MB)	6ES7953-8Lx20-0AA0
“S7-Technology”选项数据包	SIMATIC S7-Technology V4.1	6ES7864-1CC41-0YX0
STEP 7	STEP 7 V5.4 + SP2	6ES7810-4CC07-0Yxx

除“S7-Technology”选件包之外，还可以使用以下软件产品：

功能	产品	订货号
SCOUT CamTool	SCOUT CamTool V2.1	6AU1 810-0FA21-0XA0

PROFIBUS DP(DRIVE) 上的组件

T-CPU 支持下列组件在等时模式下在 DP(DRIVE) 上执行工艺任务（本手册印刷发行时的状态）：

产品	订货号
SIMODRIVE	
SIMODRIVE 611U 通用型	6SN1118-xNH00-0AAx
SIMODRIVE 611U 通用型 HR	6SN1114-0NB0x-0AAx
带有 PROFIBUS DP 的选件模块运动控制 (适用于 SIMODRIVE 611 U)	6SN1114-0NB01-0AA0
SIMODRIVE POSMO CA	6SN2703-3AAx
SIMODRIVE POSMO CD	6SN2703-2AAx
SIMODRIVE POSMO SI	6SN2 4x
SIMODRIVE 传感器单匝/同步法兰	6FX2001-5FP12
SIMODRIVE 传感器单匝/紧固法兰	6FX2001-5QP12
SIMODRIVE 传感器多匝/同步法兰	6FX2001-5FP24
SIMODRIVE 传感器多匝/紧固法兰	6FX2001-5QP24
MICROMASTER 4	
COMBIMASTER 411	6SE6401-0PB00-0AA0
MICROMASTER 420	6SE6400-1PB00-0AA0
MICROMASTER 430	6SE6400-1PB00-0AA0
MICROMASTER 440	6SE6400-1PB00-0AA0
具有通信模块 CBP2 的 MASTERDRIVES	
运动控制	6SE7090-0xx84-0FF5
运动控制补充	6SE7090-0xx84-0FF5
矢量控制 CUVC	6SE7090-0xx84-0FF5
矢量控制 Plus	6SE7090-0xx84-0FF5
当订购通信模块 CBP2 时要注意订货号后缀“Gxx”。	

2.4 支持的组件和系统

产品	订货号
SINAMICS	
SINAMICS S120 CU310 DP (V2.5 + SP1 以及 V2.5 + SP1 以下的固件)	6SL3040-0xA00-0xxx
SINAMICS S120 CU320 (V2.5 + SP1 以及 V2.5 + SP1 以下的固件)	6SL3040-0xA00-0xxx
SINAMICS G120 CU240S DP	6SL324x-0BAxx-xPAx
SINAMICS G120 CU240S DP F	6SL3244-0BA21-1PA0
SINAMICS G120D CU240D DP	6SL3544-0FA20-1PA0
SINAMICS G120D CU240D DP F	6SL3544-0FA21-1PA0
端子模块 TM15 *	6SL3055-0AA00-3FA0
端子模块 TM17 高性能型 *	6SL3055-0AA00-3HA0
SINUMERIK	
ADI4	6FC5211-0BA01-0AAx
SIMATIC IM	
PROFIBUS 模块 IM 174	6ES7174-0AA00-0AA0
SIMATIC ET 200M **	
IM 153-2	6ES7153-2AA01-0XB0
IM 153-2	6ES7153-2AA02-0XB0
IM 153-2	6ES7153-2BA00-0XB0
IM 153-2	6ES7153-2BA01-0XB0
SM 331 AI8x14Bit	6ES7331-7HF00-0AB0
SM 331 AI8x14Bit	6ES7331-7HF01-0AB0
SM 332 AO4x16Bit	6ES7332-7ND01-0AB0
SM 332 AO4x16Bit	6ES7332-7ND02-0AB0
SM 321 DI16xDC24V	6ES7321-1BH10-0AA0
SM 321 DI16xDC24V, 报警	6ES7321-7BH01-0AB0
SM 322 DO16xDC24V/0.5A	6ES7322-1BH10-0AA0

产品	订货号
SIMATIC ET 200S **	
IM 151-1 高性能型	6ES7151-1BA00-0AB0
IM 151-1 高性能型	6ES7151-1BA01-0AB0
IM 151-1 高性能型	6ES7 151-1BA02-0AB0
2AI I 2/4WIRE HS	6ES7134-4MB02-0AB0
2AI I 2WIRE HS	6ES7134-4GB51-0AB0
2AI I 2WIRE HS (仅带有 IM 151-1 高性能型 6ES7 151-1BA02-0AB0)	6ES7134-4GB52-0AB0
2AI I 4WIRE HS	6ES7134-4GB61-0AB0
2AI U HF	6ES7134-4LB02-0AB0
2AI U HS	6ES7134-4FB51-0AB0
2AO I HF	6ES7135-4MB01-0AB0
2AO I HF	6ES7135-4MB02-0AB0
2AO U HF	6ES7135-4LB01-0AB0
2AO U HF	6ES7135-4LB02-0AB0
2AO U HS (仅带有 IM 151-1 高性能型 6ES7 151-1BA02-0AB0)	6ES7135-4FB52-0AB0
2DI DC24V HF	6ES7131-4BB00-0AB0
2DI DC24V HF	6ES7131-4BB01-0AB0
4DI UC24..48V	6ES7131-4CD00-0AB0
4DI DC24V HF	6ES7131-4BD00-0AB0
4DI DC24V HF	6ES7131-4BD01-0AB0
8DI DC24V	6ES7131-4BF00-0AA0
2DO DC24V/0,5A HF	6ES7132-4BB00-0AB0
2DO DC24V/0,5A HF	6ES7132-4BB01-0AB0
2DO DC24V/2A HF	6ES7132-4BB30-0AB0
2DO DC24V/2A HF	6ES7132-4BB31-0AB0
4DO DC24V/0,5A ST	6ES7132-4BD00-0AA0
4DO DC24V/0,5A ST	6ES7132-4BD01-0AA0
4DO DC24V/2A ST	6ES7132-4BD30-0AA0
4DO DC24V/2A ST	6ES7132-4BD31-0AA0
4DO DC24V/2A ST	6ES7132-4BD32-0AA0
8DO DC24V 0.5A	6ES7132-4BF00-0AA0

2.4 支持的组件和系统

产品	订货号
PM-E DC24..48V	6ES7138-4CA50-0AB0
PM-E DC24-48V/AC24-230V	6ES7138-4CB00-0AB0
PM-E DC24-48V/AC24-230V	6ES7138-4CB10-0AB0
PM-E DC24-48V/AC24-230V	6ES7138-4CB11-0AB0
PM-E DC24..48V	6ES7138-4CA00-0AA0
PM-E DC24..48V	6ES7138-4CA01-0AA0
* 针对附加的高速输出凸轮、硬件限位开关和测量输入 ** 针对模拟绝对编码器、液压轴的模拟输出、附加输出凸轮和硬件限位开关。	

本部分列出的组件还可以在 HW Config 的硬件配置文件中找到。为此，请在 HW Config 中选择配置文件“SIMATIC T-CPU”。

必须安装最新版的 S7 Technology 才能在配置文件中获得完整选择列表。

DP(DRIVE) 上的 DP-V0 从站

除了 ET 200M 和 ET 200S，该系统还支持将附加 I/O 用作 DP(DRIVE) 上的 DP-V0 从站，但有一些限制：

- 无法中断
- DP(DRIVE) 上 DP-V0 从站无法在等时模式下一致操作
- 期望更长的响应时间

POSMO A 定位驱动器

POSMO A 定位驱动器可以在 CPU 317T-2 DP 的 DP/MPI 区段上操作。可以使用“Posmo A 库”的特殊功能块集成定位驱动器。“Posmo A 库”不支持 DP(DRIVE)。

产品	订货号
SIMODRIVE POSMO A	6SN21x

硬件和软件要求

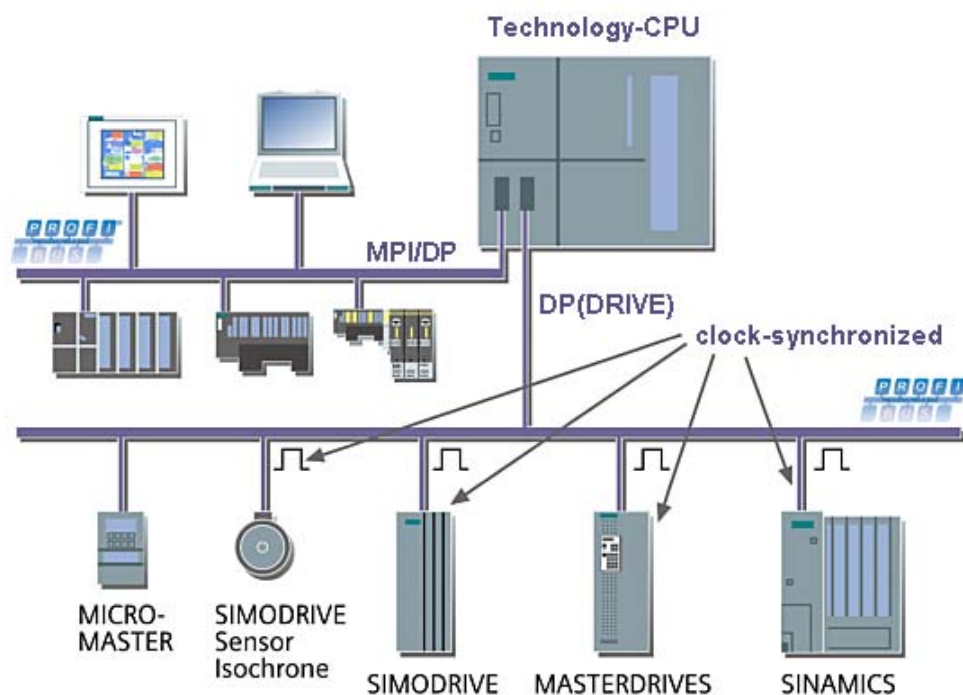
有关实施“S7-Technology”选件包

时对硬件和软件的要求，请参考产品光盘中的 readme.wri 文件。

2.5 组件及其用途

2.5.1 硬件组件

下图显示了使用 T-CPU 的运动控制解决方案的硬件组件：



T-CPU

T-CPU 控制单元执行的任务与 S7-300 系列的标准 CPU 不同。该集成工艺可以控制、评估和监视 DP(DRIVE) 上运动控制任务需要的所有硬件组件。CPU 31xT 具有 4 个集成数字输入和 8 个集成数字输出。MICROBOX T 具有 8 个集成数字输出。

MPI/DP

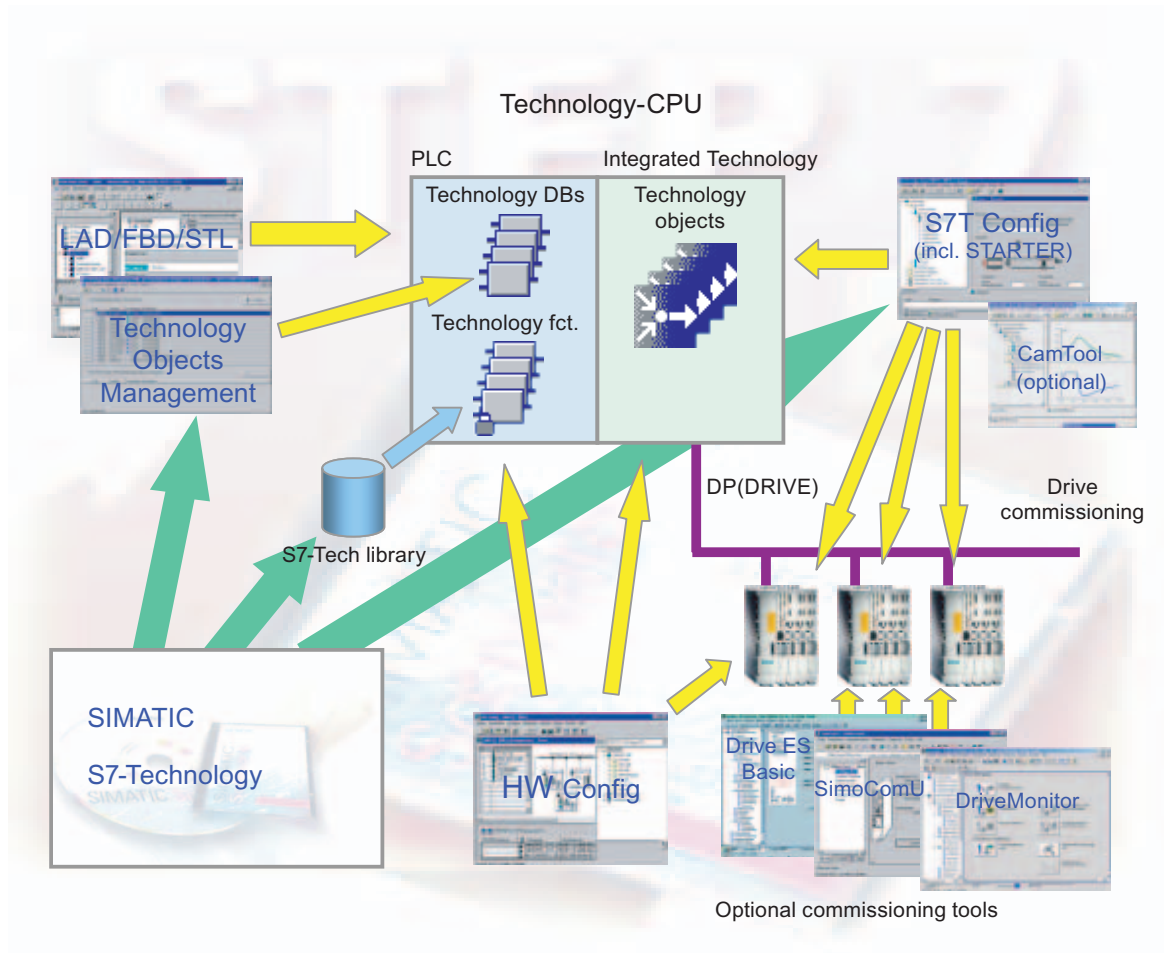
MPI/DP 接口用于连接附加 SIMATIC 组件，例如 PG、OP、S7 控制器和分布式 I/O。将其用作 DP 接口可以组态大型的扩展网络。

DP(DRIVE)

T-CPU 用于在等时模式下对 DP(DRIVE) PROFIBUS 接口进行操作。通过该集成工艺寻址的所有硬件组件必须在 DP(DRIVE) 可用。这些组件包括 MICROMASTER、SIMODRIVE、MASTERDRIVES、SINAMICS 和 SIMODRIVE 传感器系列。

2.5.2 组态工具

所有运动控制应用程序都在 **STEP 7** 中进行组态和编程。下图显示了用于组态 **MC** 应用程序的工具。



SIMATIC STEP 7

STEP 7 是用于对 T-CPU 进行组态和编程的全局平台。所需的所有组态工具都在 STEP 7 的 SIMATIC Manager 中调用。

SIMATIC S7-Technology

SIMATIC S7-Technology 是一个选件包，用于组态 T-CPU 的运动控制功能。

SIMATIC S7-Technology 在安装后将完全集成到 STEP 7 中。安装包括的工具具有：

- 工艺对象管理
- S7-Tech 库
- S7T Config, 包括 STARTER

LAD/FBD/STL

在 LAD/FBD/STL 块编辑器中对您的应用程序和运动控制作业进行编程。从用户程序中读取运动控制应用程序的实际值，以分析信息和错误。

工艺对象管理

“工艺对象管理”工具用于创建和删除工艺 DB，以及设置其参数。还可以调用“工艺对象管理”工具来重命名工艺 DB 或分配不同的块编号。

S7-Tech 库

S7-Tech 库与符合 PLCopen 标准的工艺功能兼容，在用户程序中作为功能块来调用。使用这些功能控制运动控制命令。

S7T Config

在 S7T Config 中组态执行运动控制任务所需的工艺对象。

MICROMASTER 和 SINAMICS 系列驱动器的 STARTER 功能已集成到 S7T Config 中。

S7T Config 中的 STARTER 功能

参数化具有已集成到 S7T Config 中的 STARTER 功能的 COMBIMASTER、MICROMASTER 和 SINAMICS 驱动器。可以在 SIMATIC Manager 和 S7T Config 的浏览器中调用对这些驱动器的参数化。

上述驱动器也可以作为独立驱动器进行参数化。为此，请将所需驱动器作为“单个驱动器设备”插入 S7T Config 中。参数化通过 PROFIBUS 或串行接口 COM1 进行，具体取决于可用的通信接口。

2.5 组件及其用途

CamTool（可选）

您可以购买可选的 SCOUT CamTool，它提供了一个可创建凸轮盘的易用图形界面。

HW Config

HW Config 用于配置 T-CPU 的硬件及 DP/MPI 和 DP(DRIVE) 接口的子网。

Drive ES Basic/SimoCom U/DriveMonitor

可选工具包 Drive ES Basic 和/或单机工具 SimoCom U (SIMODRIVE) 或 DriveMonitor (MASTERDRIVE) 可以支持驱动器调试。

工艺数据块

可在用户程序中调用工艺 DB 以读取工艺对象的实际数据，即轴的实际值和状态，或错误信息。

工艺功能

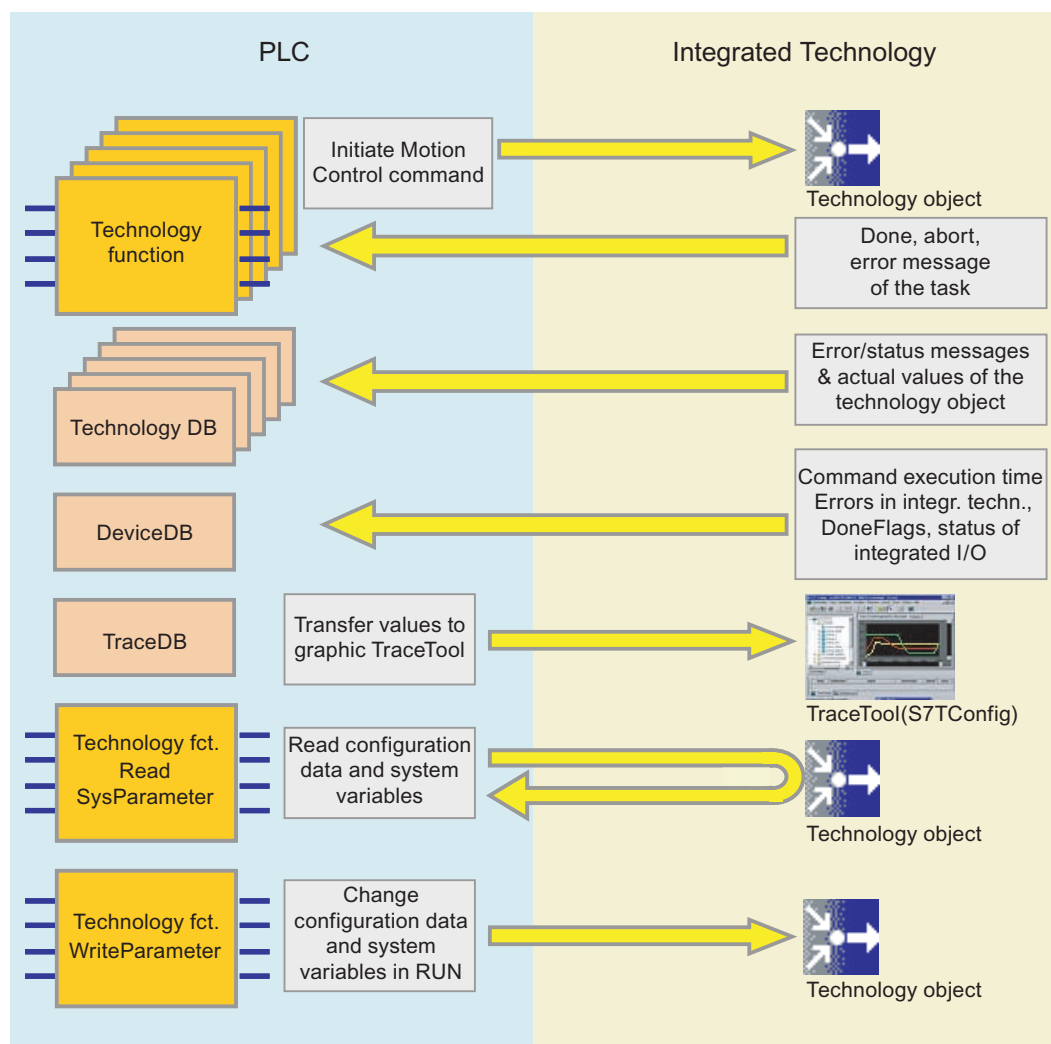
符合 PLCopen 的工艺功能在控制器的用户程序中调用。工艺功能构成了工艺对象的命令接口。用户程序可以通过判断工艺功能的输出参数来监视运动控制命令的状态。

工艺对象

物理驱动器被映射到各工艺对象及其在各工艺对象中参数化的属性。每个工艺对象都作为工艺 DB 映射到 STEP 7 用户程序，以指明状态。可以将工艺对象互连，并逻辑关联到硬件组件。所有工艺对象（例如轴、凸轮盘、输出凸轮、测量输入或外部编码器）都在 S7T Config 中组态。

2.5.3 工艺功能和工艺 DB

工艺功能和工艺 DB 构成了集成工艺的用户接口。下图显示了由这些组件处理的相应任务：



工艺功能

所有运动控制作业都是在 T-CPU 中通过工艺功能来启动。集成工艺根据控制系统启动的顺序来处理作业。

运动控制命令对您 S7T Config 中组态的工艺对象进行寻址。通过设置相应工艺 DB 的编号对这些工艺对象进行寻址。

命令通过输入参数 *Execute* 或 *Enable* 的信号跳转（上升沿）启动。工艺功能的输出参数将返回有关作业已完成和作业取消消息以及错误消息（如果作业启动失败）的信息。

工艺 DB

集成工艺将工艺对象的实际值写入分配的工艺 DB。在作业处理期间生成的状态和错误消息也将写入工艺 DB。

不管 DB 中设置了什么对象属性，工艺 DB 在自动化系统中始终**不保持**并且处于写保护状态。

DB — MCDevice

集成工艺的状态被映射到工艺 DB“MCDevice”。“MCDevice”包含有关运动控制命令的最长作业处理时间和平均作业处理时间的信息，以及有关集成工艺的所有错误的信息。

另外，通过“MCDevice”工艺 DB 还可以显示集成 I/O 的状态和某些工艺功能的 32 条完成的消息 (*DoneFlag*)。可以使用工艺功能的输入参数 *DoneFlag* 选择显示哪些完成的消息。

DB — 跟踪

S7T Config 拥有图形化的 TraceTool，可以用来分析系统参数和实际值。

除了集成工艺的系统参数，该工具还支持记录最多 8 个 S7 用户程序变量（两个 DINT 值、两个 DWORD 值和四个 REAL 值）。跟踪工艺 DB 作为接口，将控制系统和集成工艺联系在一起。

工艺功能 ReadSysParameter/WriteParameter

在运行时可能需要临时修改集成工艺的组态数据和系统变量。使用工艺功能“MC_ReadSysParameter”可以读取组态数据和系统变量，使用“MC_WriteParameter”可以重写。在参数中所做的更改**不保持**。

工艺对象

3.1 轴 — 基本知识

3.1.1 轴工艺

将“轴”工艺对象组态为“速度控制轴”、“定位轴”或“同步轴”。各个轴工艺的轴功能各不相同。

功能	速度控制轴	定位轴	同步轴
更改数据记录	X	X	X
速度或速率预设	X	X	X
扭矩减小的运动	X	X	X
定位	-	X	X
移至固定终点挡块	-	X	X
回原点	-	X	X
高级功能			
测量输入	-	X	X
输出凸轮	-	X	X
同步操作（齿轮，凸轮）	-	-	X
叠加同步（齿轮/凸轮）	-	-	X

3.1.2 特定的操作模式

禁用了电源单元的跟踪模式

位置和速度控制器在禁用了电源单元的跟踪模式下处于非活动状态。轴既不能启动运动，也不能启动动态制动。实际位置值和速度值将更新。如果外部功能处于活动状态，则还可以在禁用了电源单元的跟踪模式下监视轴位置。该模式仅支持真实轴。

启用了电源单元的跟踪模式

在启用了电源单元的跟踪模式下，位置控制处于非活动状态，而速度控制处于活动状态。实际位置值和速度值将更新。如果外部功能在轴上处于活动状态，则还可以在启用了电源单元的跟踪模式下监视轴位置。该模式仅支持真实轴。

仿真模式

仿真模式用于基于跟踪记录测试控制器中的编程顺序以及不同轴之间的相互作用，而无需移动轴。该模式仅支持真实轴。

在仿真模式期间，所有驱动器都必须正确连接，并且必须正常操作。通过将实际值设置为等于设定值来对轴进行内部模拟。仿真模式下的跟随误差始终为零。

3.1.3 轴类型

可以选择两种不同类型的轴进行组态。事实上，轴类型（线性或回转，每一种还作为模数轴）是由机械条件和用于计算轴特定的变量（例如位置或速度）的物理单位来决定的。

- **线性轴**

通常在轴的行程范围受到机械限制时组态线性轴。在行程范围内，位置曲线是线性的。运动的基本物理单位是长度单位（例如毫米）。

- **回转轴**

回转轴通常组态用于旋转运动范围。这些轴的运动范围在机械上不受限制。这些运动的基本单位是旋转单位（例如度）。通常，还将回转轴组态成作为模数轴操作。

可以将线性轴和回转轴组态为模数轴。在以模数起始值为下限、以模数起始值加模数长度为上限的范围内定义此类模数轴的位置（设定值和实际值）。

运动范围为 0 °到 360 °的旋转轴的模数起始值为 0 °，模数长度为 360 °。

当轴位置达到其上限时 (360 °) 复位为 0 °。当轴位置越过下限 (0 °) 时，它将设置为模数起始值加模数长度 (360 °)。在模数长度内，位置曲线是线性的。

说明

如果线性轴或旋转轴未组态为作为单向模数轴操作，则位置值将连续递增。

表示设置值和实际位置的变量是 **REAL** 数据类型，精度为 23 个二进制位（尾数）。因此，轴的定位精度随位置值的增加成比例地减小。因此，最好为无限轴使用模数轴，或在适当的时候将位置复位为零。

3.1.4 液压轴

适用于固件版本为 **V3.2.x** 或更高版本的集成工艺

V3.2.x 或更高固件版本的集成工艺支持对液压轴进行控制。T-CPU 支持通过 +/-10 V 模拟量信号控制液压驱动器/液压阀。轴将以百分比的形式返回一个操纵运动变量，即 -100% 相当于 -10 V，+100% 相当于 +10 V。

组态和编程与电轴的组态和编程类似。

液压轴的特性

- 在计算中包括阀控曲线，以补偿控制信号和液压驱动器之间的非线性。
- “轴”工艺可用于控制多个液压驱动器/阀。
- 液压驱动器中没有层叠速度或扭矩限制功能。
- 偏移叠加的附加摩擦补偿

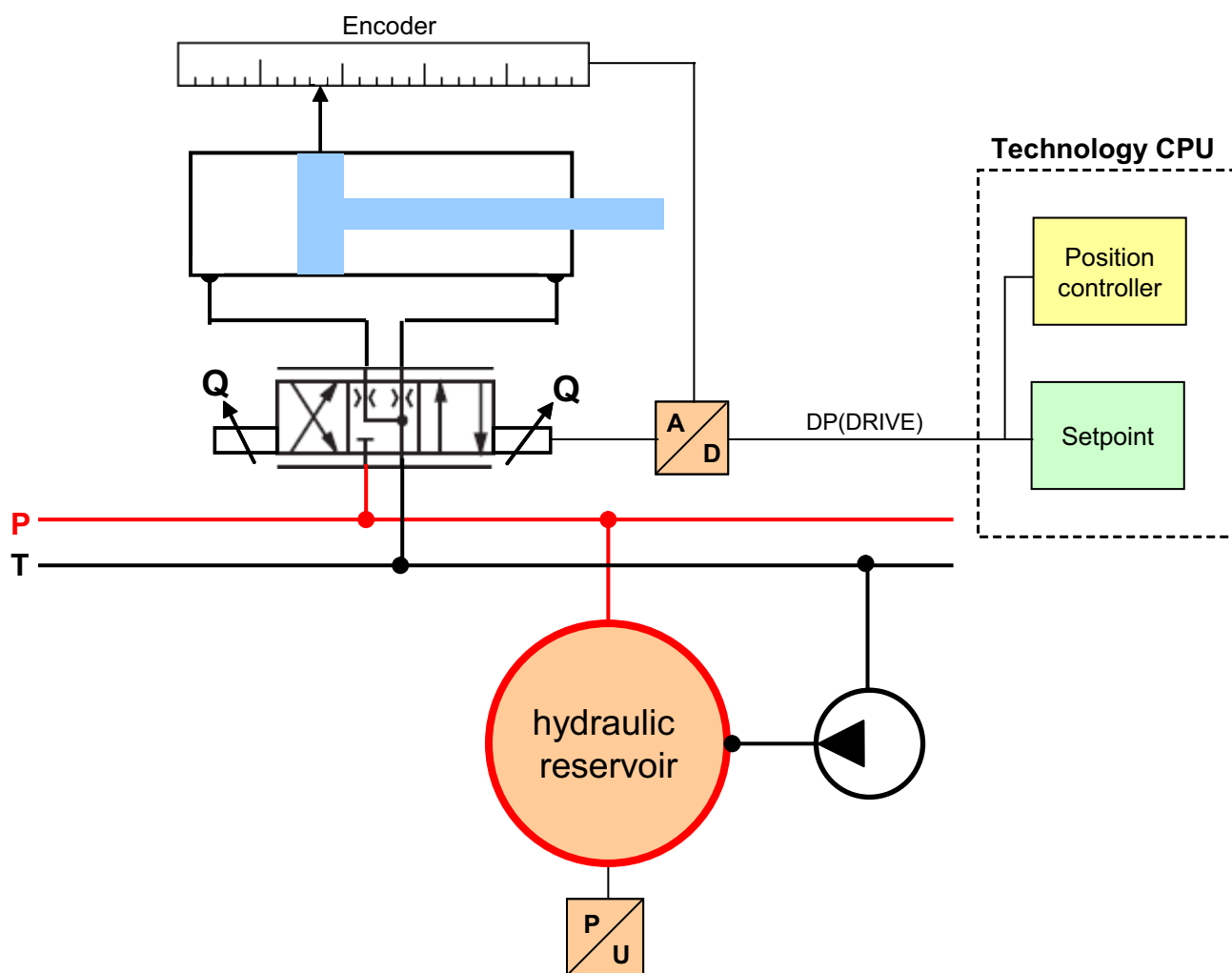
I/O 支持在 DP(DRIVE) 上操作

- 驱动器接口 IM 174/ADI4
- 用于控制液压驱动器的 ET 200M 或 ET 200S 的模拟输出模块
- 增量编码器/矩形 TTL
- 绝对编码器/SSI
- 模拟绝对编码器/传感器模拟

T-CPU 支持的控制变量：

通过比例方向阀的控制示例

使用 T-CPU 控制比例方向阀的示例



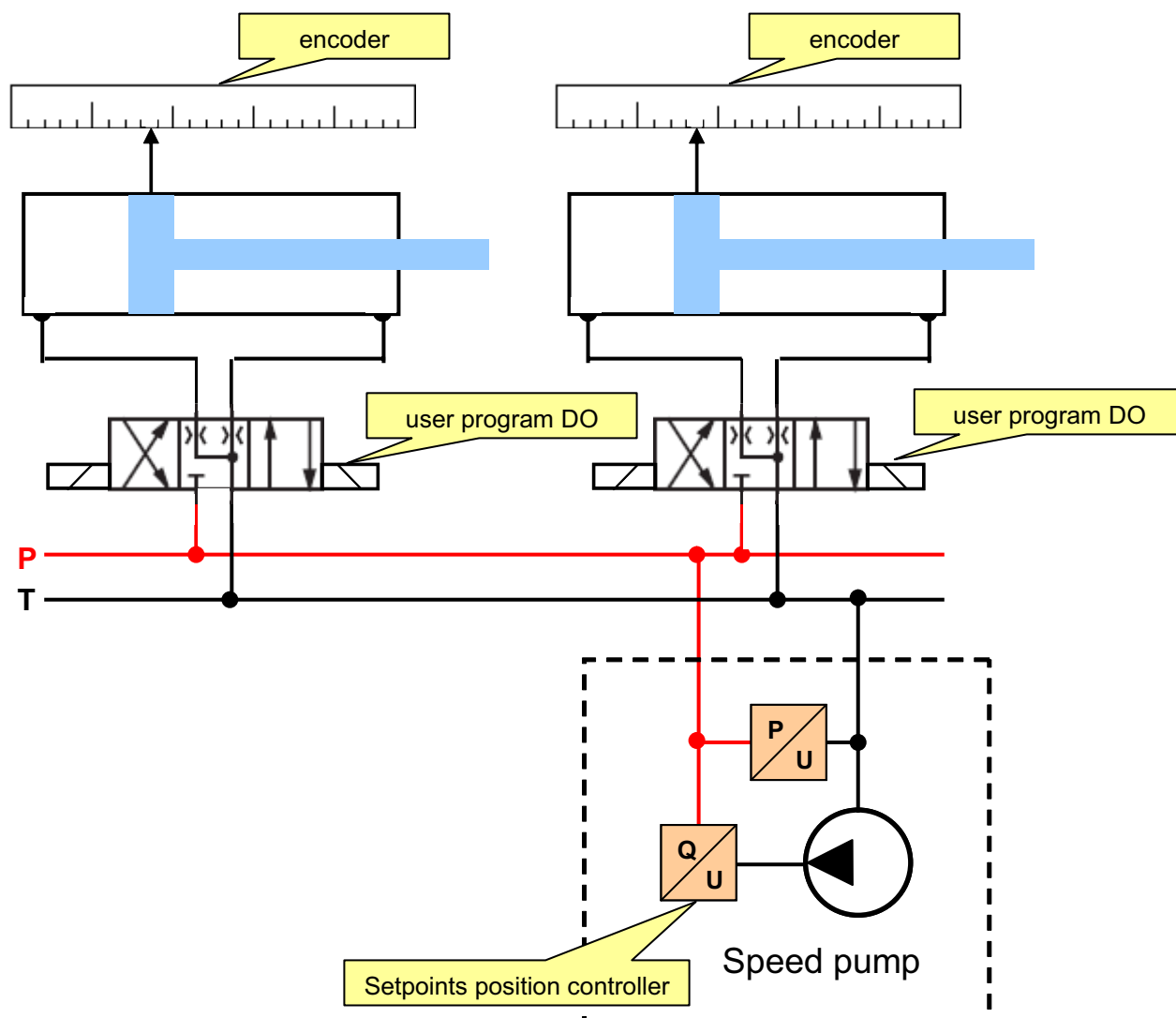
液压泵为调压液压油箱充油。液压泵必须由用户程序来控制。

在本例中，4/3 比例方向阀控制油缸或液压驱动器。T-CPU 通过在 DP(DRIVE) 上设置模拟输出来控制比例方向阀的流量 (Q)。编码器可用于记录油缸位置。

可以将液压轴组态成作为速度控制轴、定位轴或同步轴来操作。

使用切换方向阀控制变速泵的示例

控制切换方向阀和变速泵的示例



变速泵控制油缸运动的流量 (Q)。T-CPU 通过在 DP(DRIVE) 上设置模拟输出来处理变速泵的控制。必须在用户程序中通过设置数字输出来处理 4/3 切换方向阀的控制。编码器可用于记录油缸位置。

该变速泵控制仅允许通过 T-CPU 进行液压单向控制，即作为速度控制轴、定位控制轴或同步轴。必须在用户程序中通过设置 4/3 切换方向阀和激活单个阀控曲线来处理方向的反转。

该示例表明也可以使用单个液压轴（变速泵）来控制多个油缸或液压驱动器。在这种情况下，为共享轴的每个液压驱动器使用一对单独的阀控曲线。

3.1.5 实轴和虚拟轴

本文档中的“轴”一词同时指代真实轴和虚拟轴。

- 真实轴

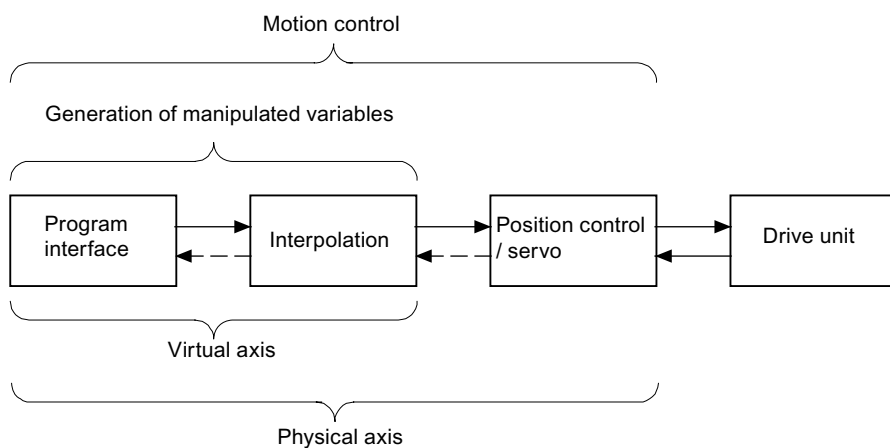
该轴的特点是具有运动控制功能以及一个驱动器和编码器接口

- 虚拟轴

该轴类型支持生成控制变量，但它没有闭环控制，也没有驱动器接口和编码器接口。

设定值和实际值始终相等。虚拟轴通常作为辅助轴，例如，以在作为同步组中的引导轴操作时为多个真实轴生成主设定值。

以一个位置轴为例，来说明真实轴和虚拟轴之间的区别：



说明

液压轴是具有阀控曲线的真实轴。

3.1.6 “轴”和“驱动器”之间的区别

“轴”工艺对象作为接口，将用户程序和实际驱动器联系起来。它可以接收、执行和监视 PLC 的用户程序所请求的运动控制命令。

轴通过 PROFIBUS 与包含速度和电流控制的驱动器进行通信。

从轴中对驱动器进行单独组态和调试。

到驱动器的功能接口

T-CPU 支持通过 DP(DRIVE) 接口操作数字驱动器（例如 SIMODRIVE 611 U，MASTERDRIVES MC）、通过 IM 174/ADI4 操作模拟驱动器或通过 IM 174 操作步进驱动器。定义的消息帧作为接口将工艺对象和驱动组件连接起来，并且必须根据所需的功能单独选择和组态每个组件。

这些消息帧用于在控制系统（工艺对象）和驱动组件（例如驱动器）之间传输数据（例如控制字、状态信号或编码器信息）。

说明

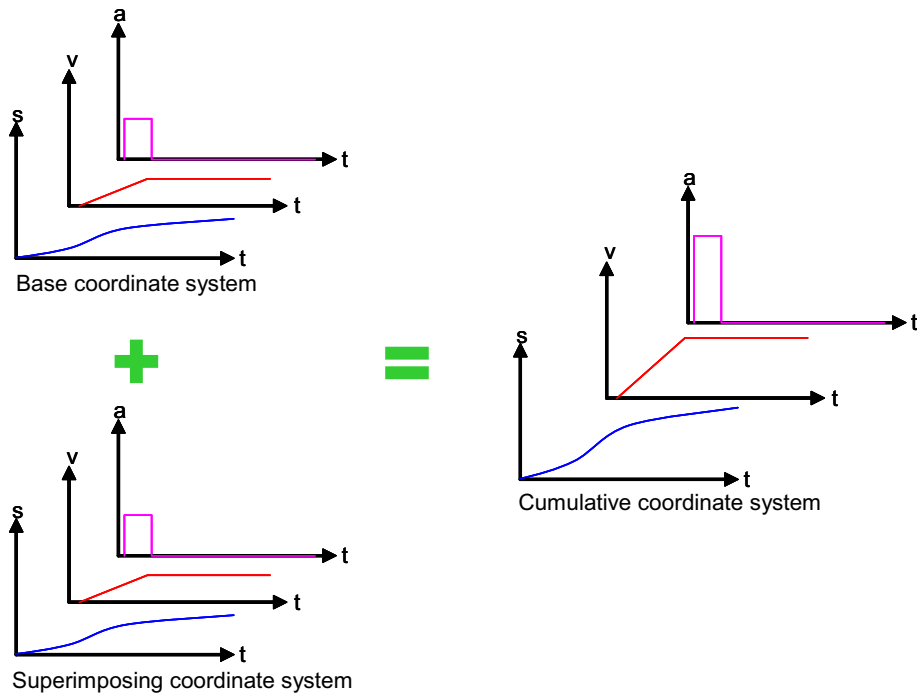
轴只能执行所连接的驱动器确实支持的功能。驱动器文档中介绍了所支持的功能，例如使用 SIMODRIVE 611U 或 MASTERDRIVES MC 的操作（有关详细信息，另请参考相应的产品说明）。

3.1.7 基准坐标系 — 叠加坐标系

我们根据它们的作用区分 T-CPU 中两种不同的运动类型：

- 基本运动
- 叠加运动

T-CPU 基于基准坐标系和叠加坐标系来计算运动。叠加运动将返回以下情况：



在给定时间添加所有坐标（时间/距离、时间/速度和时间/加速度）以返回累积坐标系。

基本运动命令：

- "MC_MoveAbsolute"
- "MC_MoveRelative"
- "MC_MoveAdditive"
- "MC_MoveVelocity"
- "MC_MoveToEndPos"
- "MC_GearIn"
- "MC_GearOut"
- "MC_CamIn"
- "MC_CamOut"
- "MC_Phasing"

这些命令会影响基准坐标系。

叠加运动命令:

- "MC_MoveSuperImposed"
- "MC_GearInSuperImposed"
- "MC_GearOutSuperImposed"
- "MC_CamInSuperImposed"
- "MC_CamOutSuperImposed"
- "MC_PhasingSuperImposed"

这些命令会影响叠加坐标系。

根据输入参数 *Mode* 的值, "MC_Home"命令可能对两种坐标系都有影响。

在轴上以系统变量来指示坐标系的矢量。

轴上基准坐标系的值:

- *basicmotion.positionbasicmotion.velocitybasicmotion.acceleration*

轴上叠加坐标系的值:

- *superimposedmotion.positionsuperimposedmotion.velocity*
- *superimposedmotion.acceleration*

累积坐标:

- *positioningstate.commandposition*: 位置设定值 (总和)
- *motionstatedata.commandvelocity*: 速度设定值 (总和)
- *motionstatedata.commandacceleration*: 加速度设定值 (总和)

在任何给定的时间, 轴上只能有一个基本运动和一个叠加运动处于活动状态。也可能在没有基本运动处于活动状态的情况下叠加运动处于活动状态。

3.1.8 回原点

位置控制轴的位置输入和显示均参考轴坐标系。轴坐标系必须与轴的实际物理位置同步。

说明

注意事项:

- 绝对编码器必须**仅进行一次**调整。
 - 当 CPU 转到 STOP 时，带增量的轴的回原点位置被保持，并在电源关后丢失。
 - 当使用增量测量系统对已回原点的轴进行操作时，该轴必须在初始化后再次回原点 (*Restart*)。
 - 使用相对位置定义（例如“MC_MoveRelative”）的运动命令在已回原点状态或未回原点状态下都可以执行。
 - 未回原点轴上，使用绝对定位的运动命令：
 - 如果从 S7T Config 中的**轴 (Axis) > 回原点 (Homing)** 对话框中“主动回原点”(Active homing) 选项卡的下拉列表中选择了“否”(no)，则可以执行
 - 如果从 S7T Config 中的**轴 (Axis) > 回原点 (Homing)** 对话框中“主动回原点”(Active homing) 选项卡的下拉列表中选择了“是”(yes)，则不可执行
-

使用 MC_Home 工艺功能同步增量编码器。在主动回原点中，将在所定义的轴机械位置设置参考点坐标或参考点坐标减去参考点偏移。

回原点模式

- 主动回原点

在主动回原点模式下，“MC_Home”工艺功能执行所需要的参考点逼近。该操作将取消所有活动运动。在 S7T Config 中可用的回原点模式：

- 参考凸轮和编码器零标记
- 仅编码器零标记
- 仅外部零标记

- 被动回原点

在被动回原点模式下，“MC_Home”工艺功能不执行参考点逼近。活动的运动命令不会受到影响。必须通过外部功能或在用户程序中实现所需要的“参考点逼近”。

在 S7T Config 中可用于组态的回原点模式：

- 参考凸轮和编码器零标记
- 仅编码器零标记
- 仅外部零标记
- 默认设置

- 直接回原点

设置轴位置时不考虑参考凸轮或零标记。轴必须处于停止状态才能将参考点准确分配到机械位置。

- 校正位置值

从当前轴位置中减去偏移值。当前运动和回原点操作不受影响。

- 校正内部轴坐标系

从基准坐标系或叠加坐标系的当前位置中减去偏移值。当前运动和回原点操作不受影响。

说明

设备专用属性

参考凸轮可以连接到 CPU 的集成输入，也可以连接到 DP(DRIVE) 上的从站，以使用参考凸轮和编码器零标记回原点。

在使用“仅外部零标记”的回原点模式下，外部零标记必须连接到同时还注册了编码器的输入设备上，例如，驱动器或 IM 174/ADI4。

有关设备专用条件和附加参数设置的详细信息，请参见产品光盘和手册上有关 SIMODRIVE 611 U、MASTERDRIVE-MC、SINAMICS S120 或 IM 174/ADI4 的补充信息。

3.1.9 绝对编码器调整

“MC_Home”(Mode = 5) 和“(MC_ExternalEncoder= Mode6MC_ExternalEncoder) 工艺功能用于在轴和外部编码器的计算中激活绝对编码器偏移。

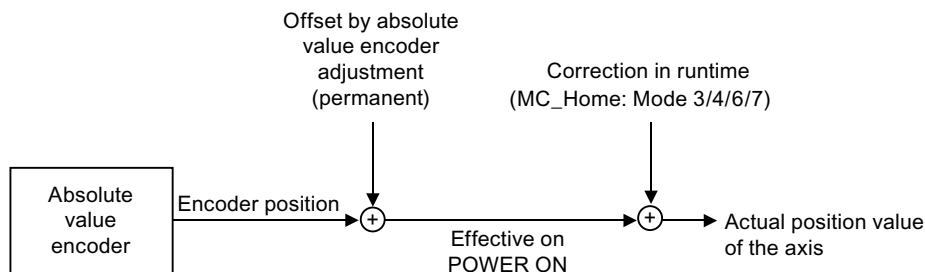
将使用绝对编码器的轴的实际位置设置为所需的值。这使绝对编码器的绝对相位移动了一个偏移值。该偏移值将永久存储并保持激活状态，直到下一次调整绝对编码器。该功能必须在控制器调试阶段执行一次。

如果 T-CPU 从 STOP 转到 RUN 且工艺对象无效，则删除该偏移。例如，如果 T-CPU 在没有 MMC 或紧凑型闪存卡的情况下启动，便是这种情况。

调整绝对编码器的步骤：

1. 禁用软件限位开关，因为这些开关处于活动状态时您无法调整绝对编码器。
2. 将轴移动到相应的参考位置，然后调整绝对编码器（“MC_Home”或“MC_ExternalEncoder”工艺功能）。
3. 重新启用软件限位开关（如果需要）

请注意，绝对编码器调整的结果仅会使编码器值发生偏移。绝对编码器调整的偏移和绝对编码器值将决定电源关或重新启动（“MC_Reset”、*Restart = TRUE*）后的位置。在操作期间，当前的实际位置还受轴的模数设置以及定位或位置校正命令的影响。



3.1.10 数据记录转换

引言

例如，您可以创建多个轴数据记录来更改控制器参数，或在系统操作时从电机编码器模式切换到机床编码器模式。

下列轴支持多数据记录：

- 速度控制轴
- 定位轴
- 同步轴

说明

虚拟轴仅有一个数据记录。

组态

在 **S7T Config** 的轴组态中创建数据记录。使用轴向导对数据记录进行编程。调用“MC_ChangeDataset”工艺功能以在 **RUN** 状态下更改数据记录。

3.2 “速度控制轴”工艺对象

3.2.1 “速度控制轴”工艺对象



如果您只想在位置不相关时预设、控制和监视轴的速度，则使用“速度控制轴”工艺对象。

速度控制轴支持的操作模式： 工艺对象 — 速度控制轴速度控制轴 — 工艺对象

- 速度控制（可使用工艺功能“MC_Power”设置）
- 跟随模式（可使用工艺功能“MC_Power”设置）
- 仿真模式（可使用工艺功能“MC_Power”设置）

支持与速度控制轴一起使用的功能：

- 预设的速度
- 扭矩减小的运动

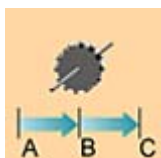
工艺功能

该工艺对象支持的工艺功能：

MC_Power	MC_Reset	MC_Stop
MC_Halt	MC_MoveVelocity	MC_ReadSysParameter
MC_WriteParameter	MC_SetTorqueLimit	MC_ChangeDataset

3.3 “定位轴”工艺对象

3.3.1 “定位轴”工艺对象



使用该工艺对象将轴移动到定义的位置并控制和监视该位置。

定位轴具有速度控制轴的所有功能。

定位轴支持的操作模式： 工艺对象 — 定位轴定位轴 — 工艺对象

- 速度控制（可使用工艺功能“MC_MoveVelocity”设置）
- 位置控制（可使用工艺功能“MC_Power”设置）
- 跟随模式（可使用工艺功能“MC_Power”设置）
- 仿真模式（可使用工艺功能“MC_Power”设置）

支持与定位轴一起使用的功能：

- 以预设的速度运行
- 扭矩减小的运动
- 定位
- 移至固定终点挡块
- 回原点

工艺功能

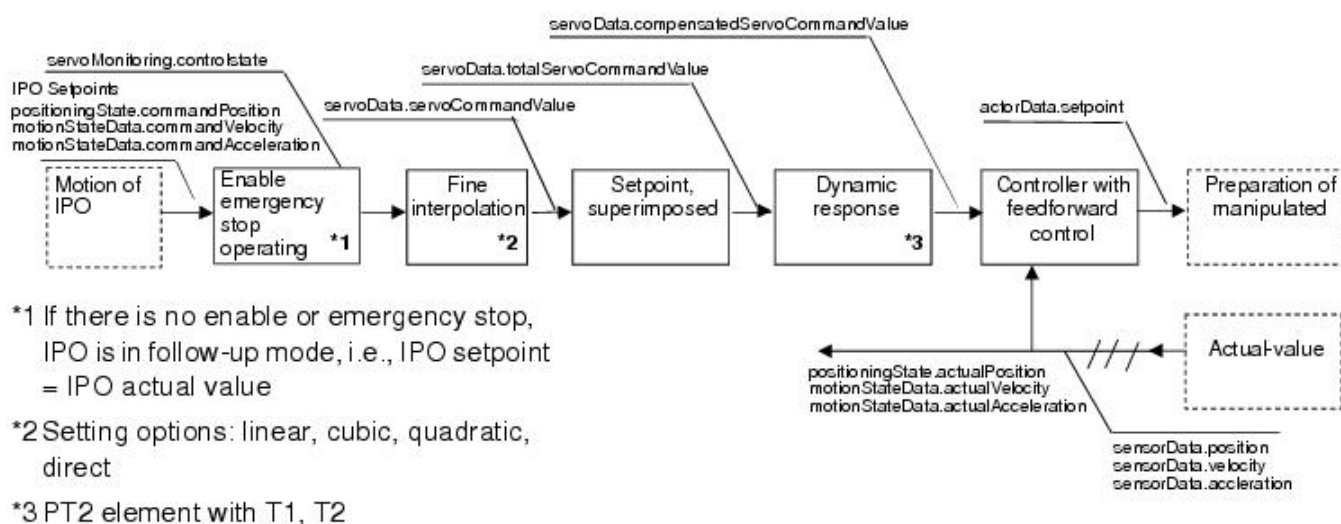
该工艺对象支持的工艺功能：

MC_Power	MC_Reset	MC_Home
MC_Stop	MC_Halt	MC_MoveAbsolute
MC_MoveRelative	MC_MoveAdditive	MC_MoveSuperImposed
MC_MoveVelocity	MC_MoveToEndPos	MC_ReadSysParameter
MC_WriteParameter	MC_SetTorqueLimit	MC_ChangeDataset

3.3.2 位置控制

3.3.2.1 位置控制

下图显示了具有位置控制的定位轴的方框图。



当位置控制处于**活动状态**时，控制器、监视和补偿都处于活动状态。监视功能在某些模式下被禁用，例如，用于扭矩或压力限制的位置相关的监视功能。

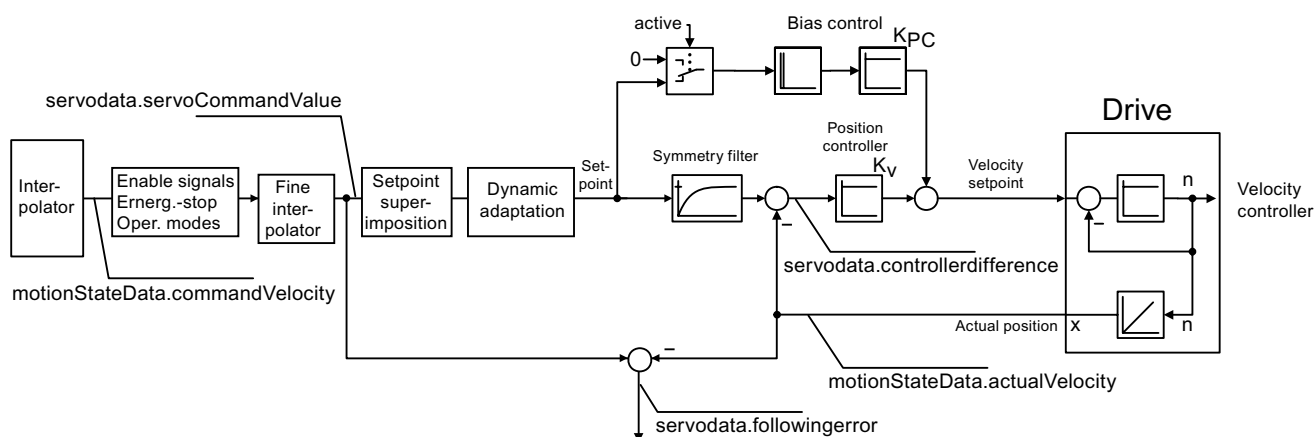
可启用/禁用所有补偿功能。

当位置控制未激活时，编码器系统、实际值计算和监视功能在实际值一侧处于活动状态。不考虑补偿功能。

servomonitoring.controlstate 系统变量指示位置控制器是否处于活动状态。

控制回路结构

S7T Config 提供了一个具有预控制功能或不具有预控制功能的比例控制器和一个 PID 调节器。



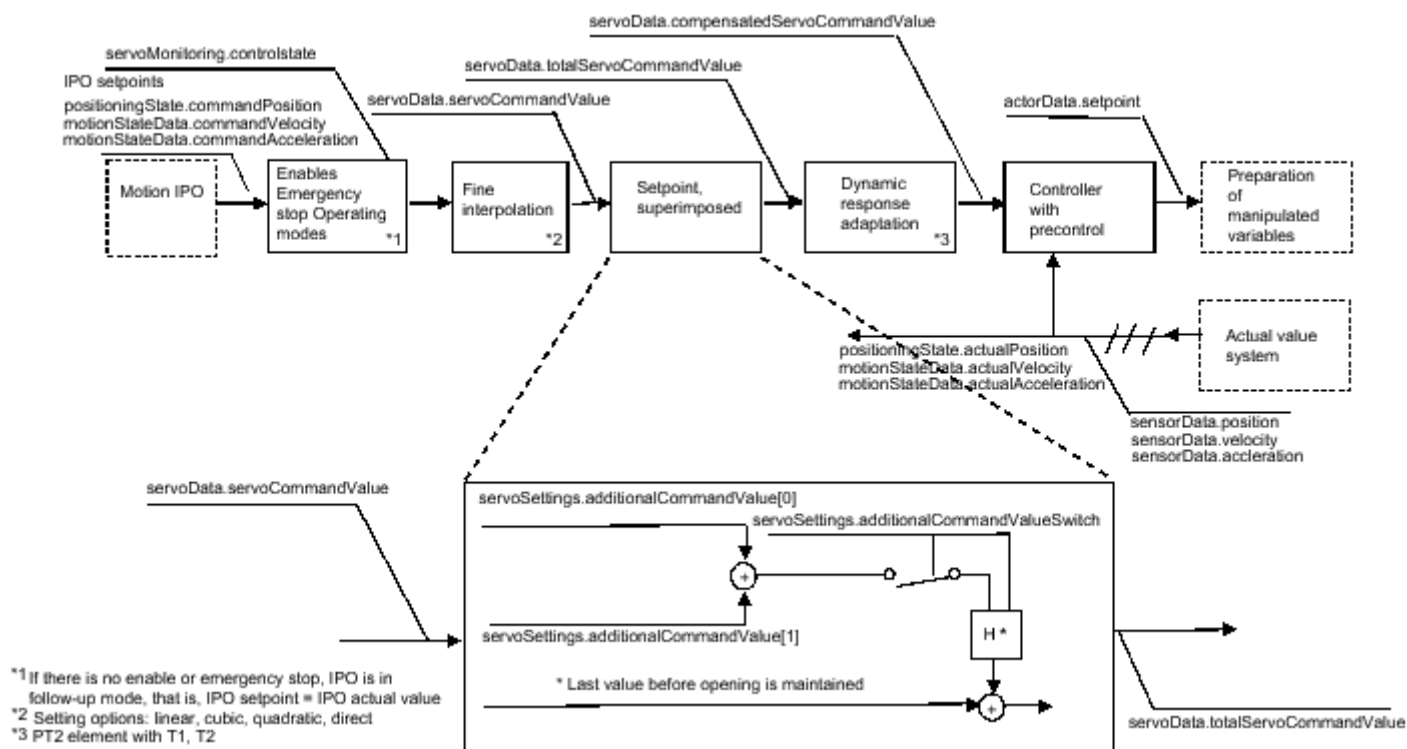
如果在轴运动时模式从速度控制更改为位置控制模式，需要位置调节器的等效时间来设置设定值。在 *dynamicData.positionTimeConstant*（电气轴）或 *dynamicQFData.positionTimeConstant*（液压功能）中进行组态期间设置该等效时间。

控制器错误的量化

使用 *commandValueQuantizationFilter* 组态数据元素量化控制器错误。

3.3 “定位轴”工艺对象

具有预控制的控制器



说明

具有预控制功能的控制器的使用方法如下：

- 带有电气轴预控制功能的比例控制器
- 用于提高数字耦合驱动器的控制质量（更高的 kv 值）的 DSC（仅对于具有预控制的比例作用控制器）
- 液压轴上的 PID 控制器。可以选择将实际值直接设置在微分作用元素中

平衡滤波器

平衡滤波器用于在将设定值与实际值比较并乘以位置控制增益之前通过处理转换时间的值来延迟设定值，然后在过程期间再次设置。

预控制处于活动状态时，可以在平衡滤波器中为形成位置设定值和实际位置的系统偏差之前的过程响应留出余量。

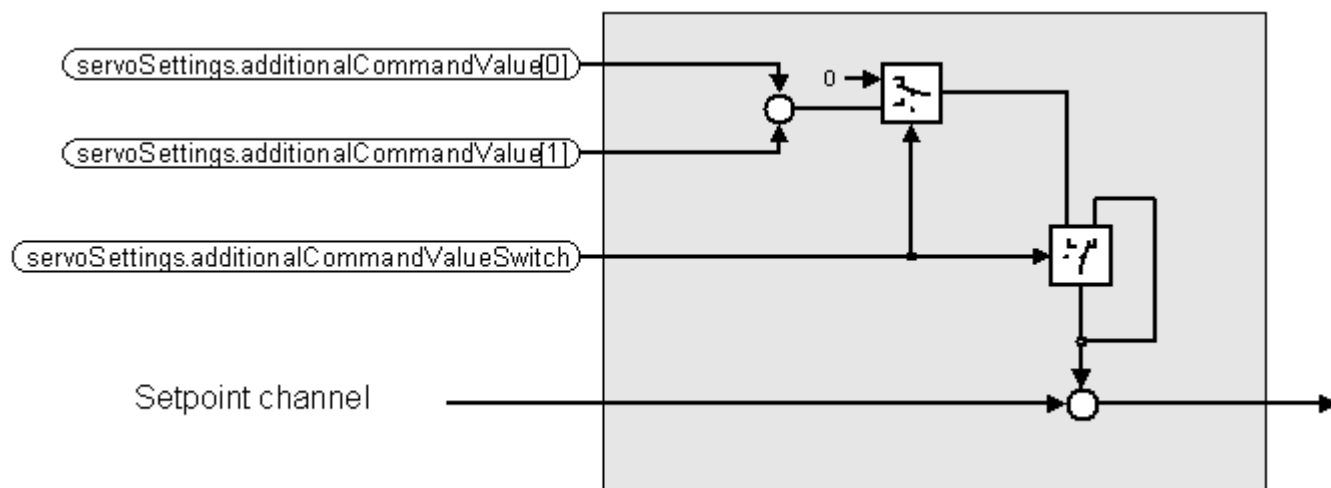
- PT1 过滤器作为具有组态数据元素设置 *balanceFilterMode = MODE_1* 的平衡滤波器使用。其时间常量在电气轴的 *dynamicData.velocityTimeConstant* 中和液压轴的 *dynamicQFdata.qOutputTimeConstant* 中设置。
- 在平衡滤波器中设置组态日期 *balanceFilterMode = MODE_2* (V3.1 和更高版本) 时，应考虑下列参数：
 - 速度控制回路的等效时间
 - 系统为驱动器确定的死区时间
 - 可由用户在附加增量中输入的死区时间
- 在具有组态数据元素 *balanceFilterMode = MODE_2* (V3.1 或更高版本) 的平衡滤波器中，要考虑速度控制回路的等效时间；包括系统根据驱动器确定的死区时间，加上可由用户输入的附加死区时间。
- 速度控制回路的等效时间包括在通过 DSC 所做的设置中。

建议您为平衡滤波器使用设置 *balanceFilterMode = MODE_2*。

3.3.2.2 设定值叠加

设定值叠加使您可以在设定值通道中设置附加设定值。

可以通过在该通道设置函数发生器的附加信号来调节控制器。

**说明**

使用设定值叠加操作的轴无法直接切换回常规位置控制模式。必须首先将设定值叠加复位为零。

叠加对定位轴的位置有影响。这种叠加还适用于跟随模式下的活动位置控制和插补器。

3.3.2.3 动态响应调整

控制回路的设定值电路包括可组态的动态 PT2 过滤器，它具有时间常量 T_1 和 T_2 。该过滤器可用于调节轴同步。该特性使您可以将较快轴的动态响应调整为适应最慢轴的响应。

选择具有最差动态响应的轴的等效时间常量 T_{LR} 作为生成的总时间常量 T_{Res} 。

从以下等式得出轴的动态响应：

$$T_{Res} = T_1 + T_2 + T_{LR}$$

或从以下等式得出两个时间常量的值：

$$T_1 + T_2 = T_{Res} - T_{LR}$$

T_1 附加时间常量 1

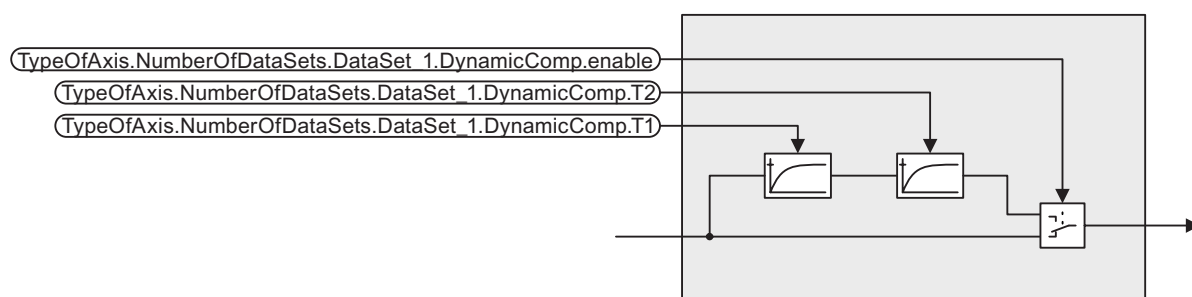
T_2 附加时间常量 2

T_{Res} （需要的）生成的轴的总时间常量

T_{LR} 闭环位置控制回路的等效时间常量

时间常量在 *NumberOfDataSets.DataSet_1.DynamicComp* 组态数据元素中设置。

可以设置 PT2 响应，从而可以通过串联两个 PT1 元件来防止发生超调。

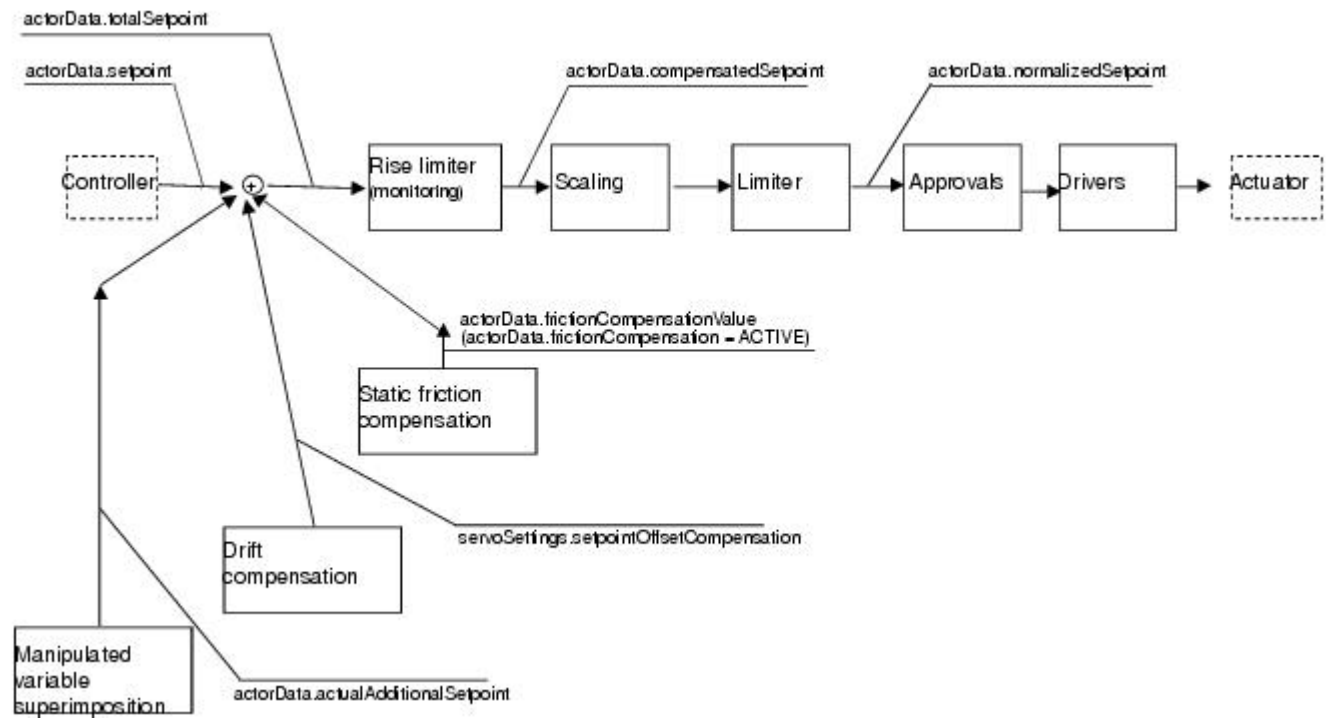


说明

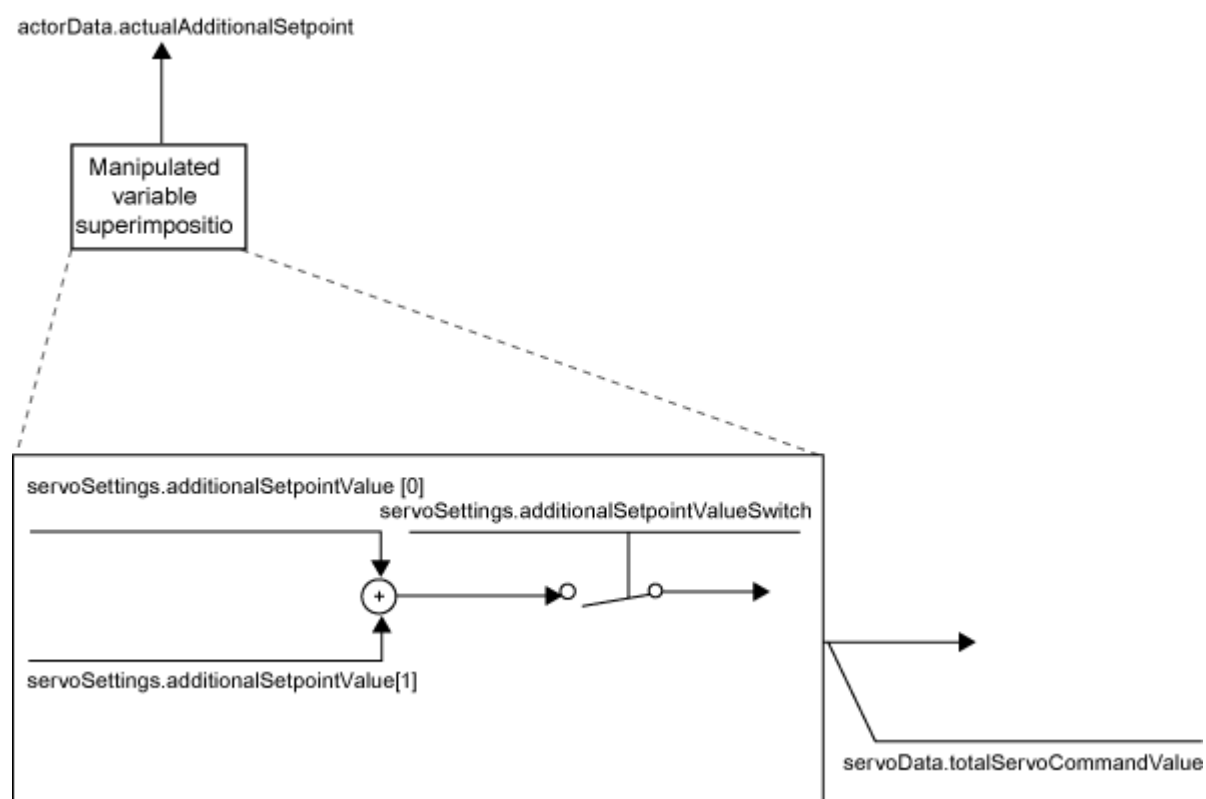
将 T_1 设置为不等于 T_2 。

3.3 “定位轴”工艺对象

3.3.2.4 准备电气轴的受控变量



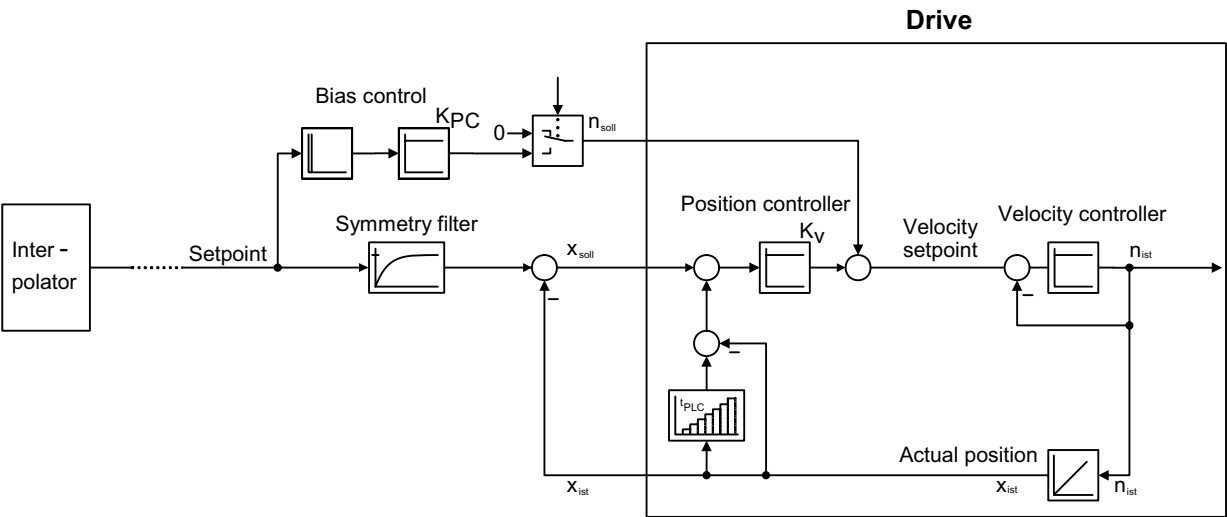
3.3.2.5 操纵变量叠加



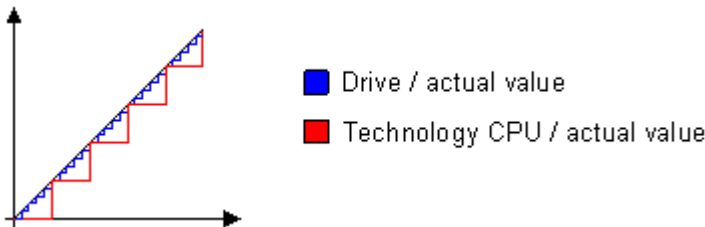
使用开关设置调节变量叠加。

3.3.2.6 动态伺服控制 (DSC, Dynamic Servo Control)

如果驱动器支持动态伺服控制 (DSC, Dynamic Servo Control)，则可在 CPU 或在驱动器中控制位置控制轴（定位轴和同步轴）的位置。



对于具有 DSC（Dynamic Servo Control，动态伺服控制）的位置控制，系统将计算驱动器中的位置控制器和速度控制器。这使您可以为位置控制器设置高出很多的增益系数 (K_v)。位置和速度控制参数在同一周期中计算。因此，位置控制及其次级控制回路（即速度控制器）将以完全相同的速度计算。



与时钟一样，实际值更改小于 T-CPU 中的实际值。通过设置更精确的实际值，可以更精确地校正位置调节器。

这将允许调节控制器的低信号响应，从而可以设置高出很多的增益系数。

DSC 使您可以为采样时间设置更高的 K_v 比率。这将增加参考变量序列中的动态过程，以及高动态驱动器上干扰响应的动态过程。支持消息帧 5、6 或 105 和 106 的所有驱动器都支持 DSC。

DSC 仅对比例位置控制器有用。

说明

位置控制器需要调节。有关详细信息，请参考“”。

3.3.2.7 位置控制轴的速度控制操作

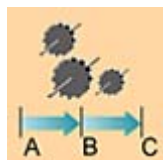
在轴处于停止状态和运动状态时，都可以设置位置控制运动和速度控制运动之间的跳转。

动态响应参数和速度规范的最大值从位置控制轴操作的设置中得到。

位置相关的监视功能将被取消激活。可以超出编码器限制频率。

3.4 “同步轴”工艺对象

3.4.1 “同步轴”工艺对象



同步轴是跟随在“引导轴”后面的“跟随轴”。使用“同步轴”工艺对象可以将引导轴的运动和位置值用作主设定值。

同步轴包括速度控制轴和定位轴的所有功能。

同步轴支持的操作模式：工艺对象 — 同步轴同步轴 — 工艺对象

- 速度控制（可使用工艺功能“MC_MoveVelocity”设置）
- 位置控制（可使用工艺功能“MC_Power”设置）
- 跟随模式（可使用工艺功能“MC_Power”设置）
- 仿真模式（可使用工艺功能“MC_Power”设置）

支持与同步轴一起使用的功能：

- 以预设的速度运行
- 扭矩减小的运动
- 定位
- 移至固定终点挡块
- 回原点
- 同步操作（齿轮传动/凸轮传动）
- 叠加同步（齿轮传动/凸轮传动）

3.4 “同步轴”工艺对象

工艺功能

该工艺对象支持的工艺功能：

MC_Power	MC_Reset	MC_Home
MC_Stop	MC_Halt	MC_MoveAbsolute
MC_MoveRelative	MC_MoveAdditive	MC_MoveSuperImposed
MC_MoveVelocity	MC_MoveToEndPos	MC_GearIn
MC_CamIn	MC_GearOut	MC_CamOut
MC_Phasing	MC_ReadSysParameter	MC_WriteParameter
MC_SetTorqueLimit	MC_ChangeDataset	MC_GearInSuperimposed
MC_GearOutSuperimposed	MC_CamInSuperimposed	MC_CamOutSuperimposed
MC_PhasingSuperimposed		

3.4.2 “同步轴”工艺对象的结构

轴对象和同步对象是相互独立的对象。两个对象共同形成了一个同步轴。

“轴”和同步对象根据它们的相关操作状态和命令的有效性相互作用。

如果激活错误仅在同步对象中未决，则跟随轴仍可定位，但是不能同步操作。可以通过确认所有错误而始终避免该影响。“轴”对象的错误对同步功能有直接影响。

当轴响应一个停止命令时，同步运动也停止。

3.4.3 同步组

“同步”工艺对象可用于互连轴，从而使它们构成同步组。

示例

引导轴（主）生成主设定值。同步对象根据定义的标准（包括齿轮速比、缩放、偏移和凸轮盘）处理该值，并且将其作为控制变量分配给跟随轴。

说明

主设定值和从值在没有物理转换的情况下，以相关的编程单位耦合在一起。例如，如果系统使用线性引导轴（以 mm 为单位）和从回转轴（以度为单位）操作，则 1 毫米对应于 1 度，转换比为 1:1。

机械同步示例

例如，当两个机械耦合的滚筒由同一电机驱动时，给定齿轮同步。

凸轮传动模型可理解为有机械凸轮盘和传感机制构成的凸轮盘。

用于同步系统的同步和结束同步组件的工艺功能对应于机械耦合的功能。

同步组

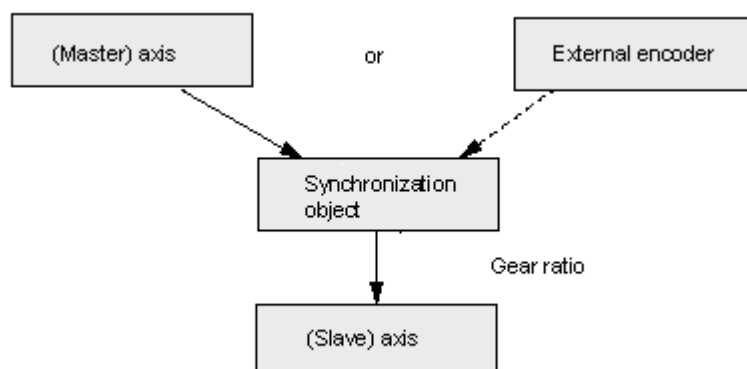
同步组至少包括这些元素：

- 返回主设定值的引导轴

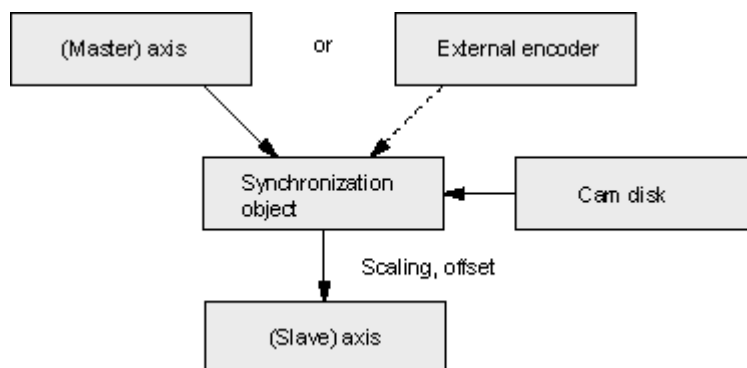
引导轴可由真实和虚拟的定位和同步轴，或由外部编码器构成。

- 同步对象
- 跟随轴

齿轮传动模式下的对象



凸轮传动模式下的对象

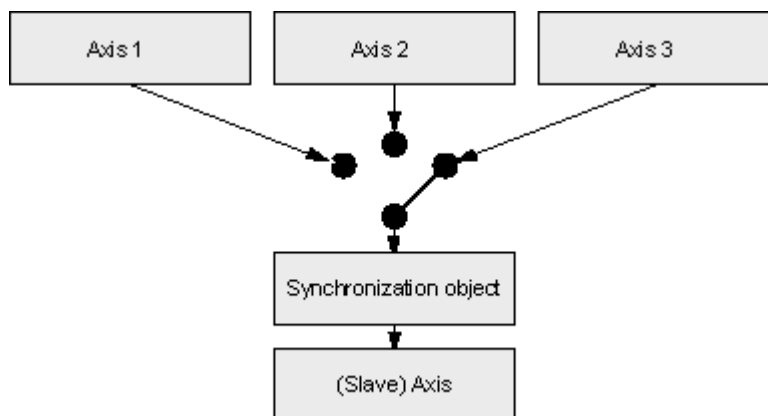


主设定值耦合

跟随轴可使用同步对象与多个主设定值互连。但是，任何时候都仅可以主动评估一个主设定值。主设定值可由定位轴、同步轴（真实轴或虚拟轴）或外部编码器返回。

为了切换为不同的主设定值，再次在用户程序中调用“MC_GearIn”（齿轮传动）或“MC_CamIn”（凸轮传动）工艺功能。使用叠加同步命令的“MC_GearInSuperimposed”和“MC_CamInSuperimposed”工艺功能。

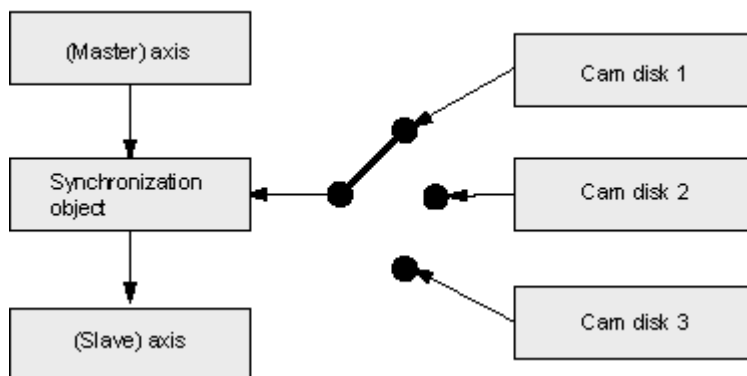
具有多个主设定值的同步对象示例



凸轮耦合

以相同方式，可以使用多个凸轮盘进行凸轮传动。可以在用户程序中使用“MC_CamIn”或“MC_CamInSuperimposed”工艺功能实现凸轮盘的动态切换。

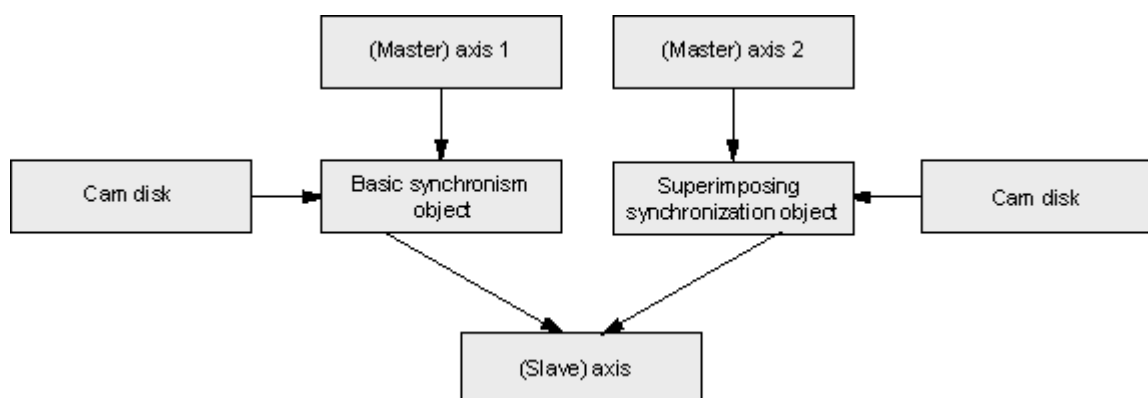
使用多个凸轮盘进行凸轮传动的示例：



叠加同步

叠加同步的响应类似于轴上基本运动的叠加定位运动（运动或同步）。

叠加同步和基本同步的互连条件一样。与基本同步对象相同，您可以将叠加同步对象和多个引导轴或凸轮盘逻辑链接在一起。



规则

以下规则适用于互连：

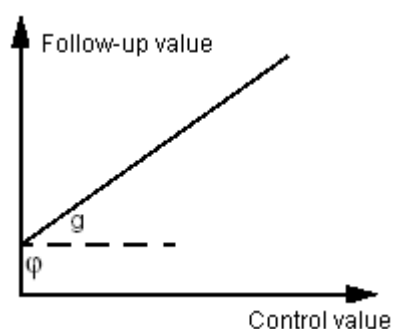
- 控制值可用于多个实例。引导轴可将主设定值输出给多个跟随轴。定位轴、同步轴、虚拟轴或外部编码器可决定主设定值。
- 同步对象可与多个主设定值和凸轮盘互连。必须在同步对象的组态中进行设置，使操作所需的所有组合都可以使用。
- 轴可能会有最多两个同步对象（基本同步和叠加同步）。

3.4.4 齿轮传动

在“MC_GearIn”和“MC_GearInSuperImposed”工艺功能中，将变换比和传动比定义为由两个整数之比与一个偏移量构成的函数。此后，可以使用“MC_Phasing”和“MC_PhasingSuperImposed”工艺功能修改该偏移。

响应特性

齿轮传动的响应特性可表示为主设定值和从设定值之间的线性相关。



跟随值 = g 主设定值 + φ

g = 传动比（变速器传动比）

φ = 偏移

方向

传动比可定义为正值或负值。产生的响应：

- 正传动比：
引导轴和跟随轴同向运动。
- 负传动比：
引导轴和跟随轴反向运动。

绝对齿轮传动或相对齿轮传动

可以在输入参数 *Absolute* 中设置绝对齿轮传动或相对齿轮传动。

- **绝对同步**确立了引导轴和跟随轴之间的绝对相关性。

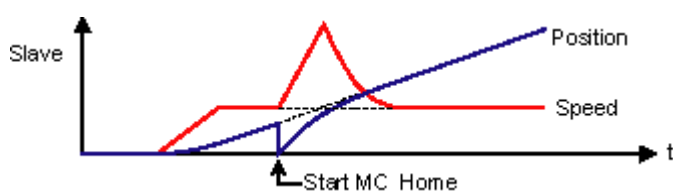
在同步期间，根据在该工艺功能中定义的偏移对引导轴与跟随轴之间的偏移进行补偿。

- **相对同步**确立了引导轴和跟随轴之间的相对相关性

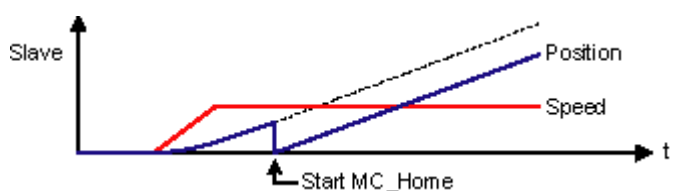
系统**不会**补偿在同步开始后轴之间可能产生的任何偏移。轴位置之间的偏移是随机设置的，由它们同步的时间决定。

在同步操作期间复位轴位置

绝对耦合将导致可以消除位置设定值的差异的补偿运动：



相对耦合不会导致补偿运动，这意味着偏移将更改：



在工艺功能“MC_Home”（*Mode = 3* 或 *4*）中复位轴位置。

3.4.5 凸轮传动

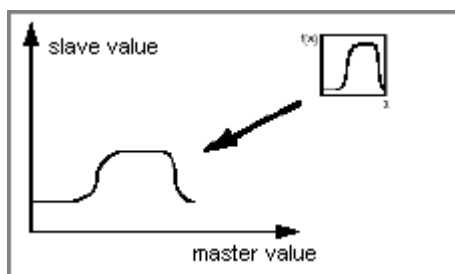
凸轮传动的特点是引导轴和跟随轴之间存在的**动态变速器传动比**以及存在偏移。变速器传动比由**凸轮盘**（传送功能）描述。

所用凸轮盘在引导轴和跟随轴中都可能发生缩放和偏移。这将根据凸轮盘的定义范围和取值范围分别对凸轮盘进行修改。

也可以在凸轮盘中将跟随轴与位置同步，就像当前活动的凸轮盘末端的凸轮盘转换。

响应特性

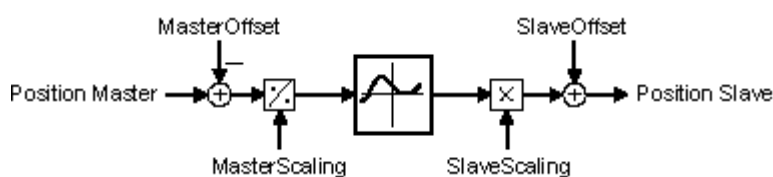
凸轮传动的响应特性由凸轮盘来定义 ($y = f[x]$)。



缩放和偏移

凸轮传动功能的缩放和偏移可以使用凸轮传动工艺功能在引导轴和跟随轴中设置。不通过调用“MC_CamIn”工艺功能修改组态的凸轮盘。

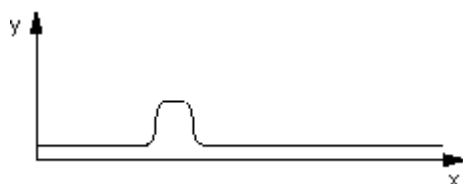
凸轮传动的传送步骤



非循环凸轮盘操作或循环凸轮盘操作

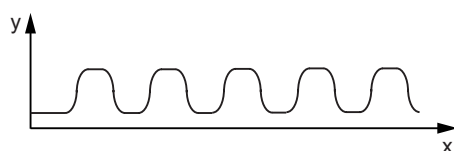
可以使用凸轮传动工艺功能的输入参数 *CyclicMode* 来设置非循环 (*CyclicMode* = 0) 或循环 (*CyclicMode* = 1) 凸轮传动模式。

- “非循环”模式意味着凸轮盘仅执行一次。当达到凸轮的终点或起点时终止凸轮传动，例如：



(x = 主设定值; y = 从设定值)

- 在循环凸轮盘模式下，引导轴计算为凸轮盘长度的模数功能。和齿轮传动一样，可以通过已编程的取消命令终止耦合。示例：



(x = 主设定值; y = 从设定值)

绝对凸轮传动和相对凸轮传动

可以在“MC_CamIn”工艺功能的 *MasterAbsolute* 和 *SlaveAbsolute* 输入参数中设置绝对凸轮传动和相对凸轮传动模式。

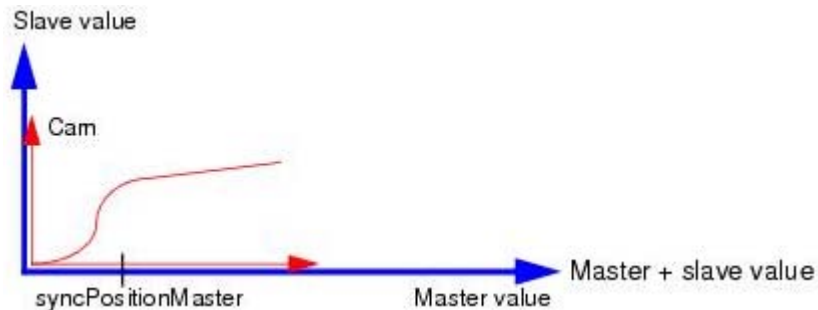
- 如果设置了**绝对凸轮传动**，则基于凸轮盘定义范围的主设定值和基于凸轮盘值范围的从设定值被解释为绝对值。

在绝对凸轮传动模式下，模数轴的起始值和结束值应当相同，因为如果不同的话，可能发生意外的步骤跳转。

- 在**相对凸轮传动**模式下操作时，凸轮盘从凸轮传动的起始值开始偏移。

绝对同步和相对同步的可能设置如下所示。

与引导轴和跟随轴绝对同步

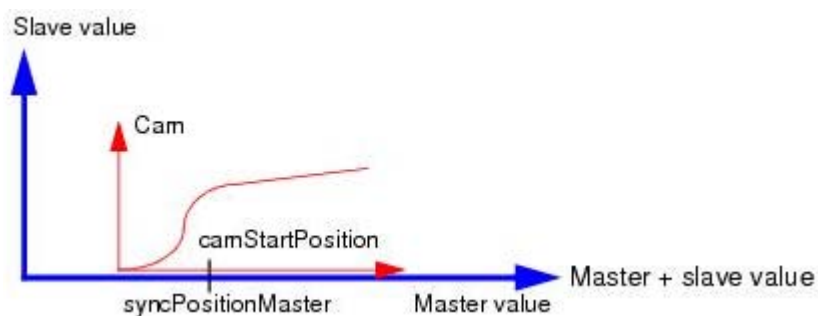


MasterAbsolute = 1; SlaveAbsolute = 1

在绝对同步操作中，有效的凸轮盘可以精确地表达已组态的凸轮盘。凸轮盘在坐标系“从值 = f（主设定值）”中不会移动。

在同步期间，系统将接近分配给实际主位置的凸轮盘点。如果跟随轴在凸轮传动开始时已定位在某个凸轮盘点上，则同步运动不会启动。

与引导轴的相对同步，和与跟随轴的绝对同步



MasterAbsolute = 0; SlaveAbsolute = 1

当正在运行相对于引导轴的凸轮传动时，系统将计算

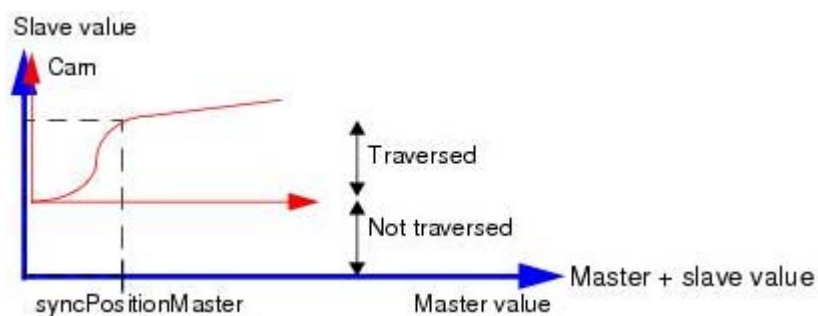
userdefault.cammingsettings.camstartpositionmaster 系统变量，并将该值设置为凸轮传动的起始位置。

通过在“跟随值 = f（主设定值）”坐标系中沿横坐标移动已组态的凸轮盘来获得有效的凸轮盘，从而使 *userdefault.cammingsettings.camstartpositionmaster* 与凸轮传动开始时跟随轴的实际位置一致。由于同步对跟随轴来说是绝对的，因此轴不会沿坐标系的纵坐标移动。

在同步期间，跟随轴将逼近分配给 *userdefault.cammingsettings.camstartpositionmaster* 的凸轮盘上的跟随值。

如果跟随轴已达到该位置，则不启动同步运动。

与引导轴的绝对同步，和与跟随轴的相对同步



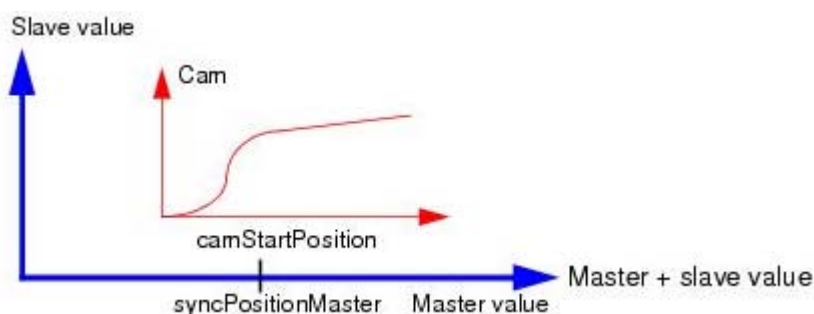
MasterAbsolute = 1; SlaveAbsolute = 0

通过在“从值 = f（主设定值）”坐标系中沿纵坐标移动已组态的凸轮盘来获得有效的凸轮盘，从而使其起始位置与跟随轴的实际位置一致。由于同步对引导轴来说是绝对的，因此轴不会沿坐标系的横坐标移动。

在同步期间，在凸轮盘定义中设置的引导轴的实际位置的从设定值，和凸轮盘的起始位置的从设定值之间的差值定义了跟随轴移动的距离。跟随轴定位到分配给引导轴的实际位置的有效凸轮盘的位置点上。

如果引导轴到达起始位置，或者到达其从值与起始位置的从值相一致的某个凸轮盘点，则轴将不会执行同步运动。

与引导轴和跟随轴的相对同步



$MasterAbsolute = 0, SlaveAbsolute = 0$

当正在运行相对于主轴的凸轮传动时，系统将计算系统变量

`userdefault.cammingsettings.camstartpositionmaster`，并将该值设置为凸轮传动的起始位置。

通过在“从值 = f（主值）”坐标系中沿横坐标移动已组态的凸轮盘来获得有效的凸轮盘，以移动 `userdefault.cammingsettings.camstartpositionmaster` 引导轴的实际位置。凸轮盘的起始位置必须与活动跟随轴的起始位置一致。

在同步期间，跟随轴将逼近分配给 `userdefault.cammingsettings.camstartpositionmaster` 的凸轮盘上的跟随值。

在同步期间，`userdefault.cammingsettings.camstartpositionmaster` 的跟随值和凸轮盘起始位置的跟随值之间的差值确定了跟随轴移动的距离。跟随轴定位到分配给引导轴的实际位置的有效凸轮盘的位置点上。

如果跟随轴已达到该位置，则不启动同步运动。

同步运动的校正

可通过切换主设定值和从设定值的缩放和偏移来校正同步运动。

其它选项包括：

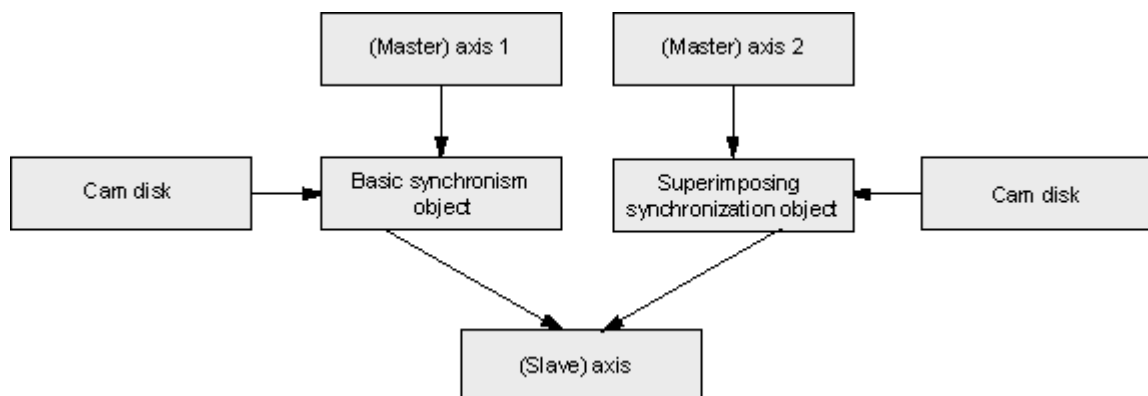
- 直接在凸轮盘上偏移和缩放
- 实时设置主值源和从轴上的参考点
- 叠加定位

说明

凸轮盘的任意点都可设置为凸轮传动的起点。可以是凸轮盘中的起点、终点或任意点。

3.4.6 叠加同步

可以为凸轮传动和齿轮传动组态叠加同步。该操作通过在同步轴上组态一个附加叠加同步对象来完成。不能组态交叉参考同步对象。



使用叠加同步来组态轴

叠加同步对象的组态方式与同步对象类似。

叠加运动

叠加同步的响应类似于轴上基本运动的叠加定位运动（运动或同步）。

无法在轴上进行多个同步叠加运动：

- 叠加定位运动
- 叠加同步

叠加同步可在当前没有活动基本运动或基本同步的情况下处于活动状态。

有关轴上叠加运动的更多信息，请参考工艺功能的相关说明。

坐标

当轴在绝对同步模式下操作并且定义了从位置时，基本同步对象将参考基准坐标。

当轴在绝对同步模式下操作并且定义了从位置时，叠加同步对象将参考叠加坐标。

与基准坐标的参考的同步和与叠加坐标的参考的叠加同步互不影响。

累积坐标从基准坐标和叠加坐标中获得。

3.4 “同步轴”工艺对象

绝对同步和相对同步

可以在同步功能（具有对叠加坐标的绝对参考）中编程和执行相对或绝对叠加同步！

监视

同步对象的输出值（和叠加轴同步的运动元素）可从同步对象中的系统变量 *currentslavedata* 中读取。

同步监视/状态

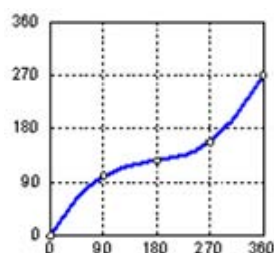
轴上的变量和监视功能将参考所产生的同步。

错误消息（同步轴中的同步错误）会报告给所有互连的同步对象。

3.4.7 相对/绝对同步

下面的示例显示了相对和绝对同步之间的差异。

在凸轮传动模式下，跟随轴位置 (XF) 对引导轴位置 (XL) 的依赖关系通过凸轮来说明。
下一个示例始终使用如下所示的基本凸轮：



这些示例涉及“立即同步”。参与轴的实际位置与该用途相关。对于位置依赖的同步，将计算“同步位置”而不是“实际位置”。

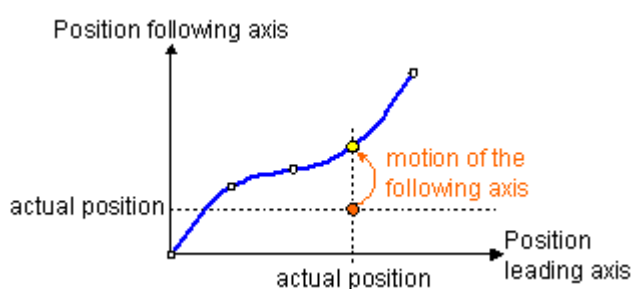
绝对同步

使用以下输入参数，在“MC_CamIn”工艺功能上组态“绝对同步”：

- *MasterAbsolute* = *TRUE*
- *SlaveAbsolute* = *TRUE*

绝对同步是最简单的使用例子，因为它使用的是基本凸轮。因此，分配永远由凸轮定义。跟随轴的位置从某个特定的引导轴位置中精确获得，因为可以从凸轮中读取该位置。

在同步操作激活期间，跟随轴从其当前位置移动到从凸轮中获得的位置。



相对同步

与基本凸轮有偏移的凸轮应用于相对同步中。我们区分以下情况：

- 凸轮沿横坐标移动 (*MasterAbsolute* = *FALSE*; *SlaveAbsolute* = *TRUE*)
- 凸轮沿纵坐标移动 (*MasterAbsolute* = *TRUE*; *SlaveAbsolute* = *FALSE*)
- 凸轮沿任意方向移动 (*MasterAbsolute* = *FALSE*; *SlaveAbsolute* = *FALSE*)

示例 1 — 凸轮沿横坐标移动

设置:

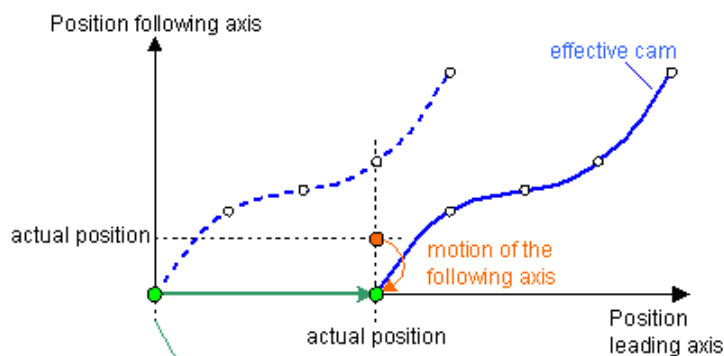
- *MasterAbsolute* = FALSE
- *SlaveAbsolute* = TRUE

使用这些设置, 可以通过沿横坐标移动基本凸轮来生成一个有效凸轮。

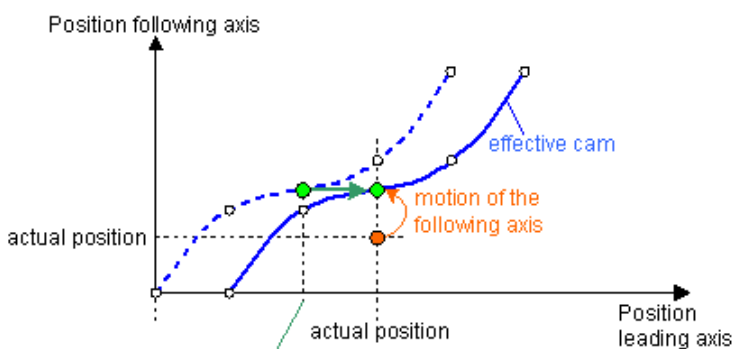
凸轮沿横坐标移动的距离取决于激活凸轮传动时引导轴的位置以及同步对象的 *userdefault.cammingsettings.camstartpositionmaster* 系统变量。移动凸轮从而使引导轴坐标 *userdefault.cammingsettings.camstartpositionmaster* 上基本凸轮的点与引导轴的实际位置一致。

第一个图适用于 *userdefault.cammingsettings.camstartpositionmaster* = 0, 第二个图适用于 *userdefault.cammingsettings.camstartpositionmaster* = 180。

跟随轴从其当前位置移动到已移动的凸轮的位置。



CamStartPositionMaster = 0



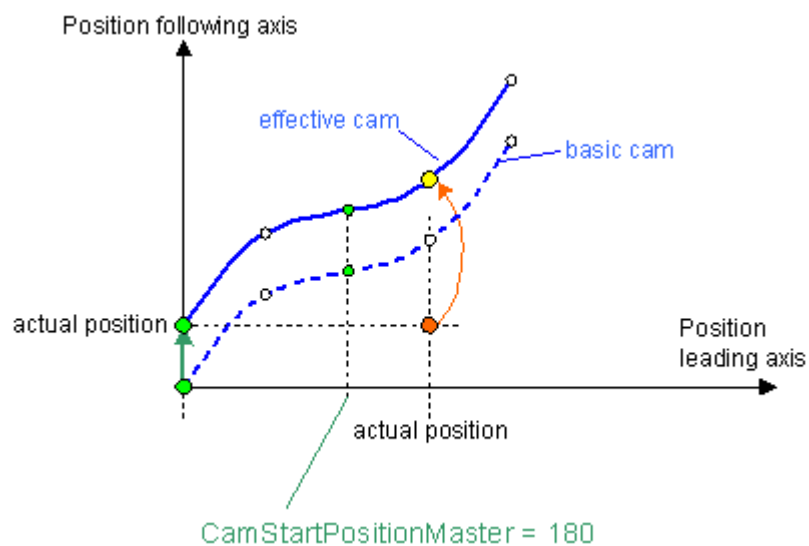
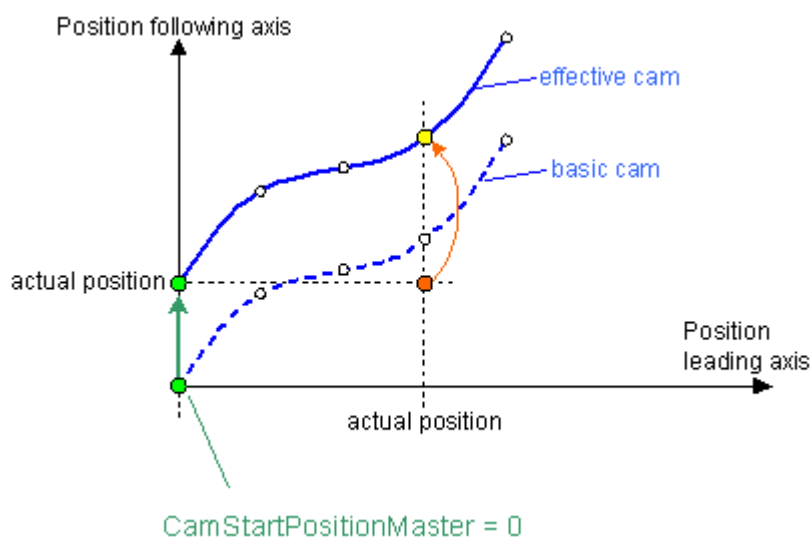
CamStartPositionMaster = 180

示例 2 — 凸轮沿纵坐标移动

设置：

- *MasterAbsolute = TRUE*
- *SlaveAbsolute = FALSE*

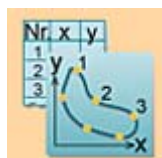
使用这些设置，可以通过沿纵坐标移动基本凸轮来生成一个有效凸轮。



在这种环境下，与此相关的是凸轮起始位置（最小的 X 值），而不是系统变量 *userdefault.cammingsettings.camstartpositionmaster*。移动凸轮，从而使其起点移动到跟随轴的实际位置。

3.5 “凸轮盘”工艺对象

3.5.1 “凸轮盘”工艺对象



使用“凸轮盘”工艺对象实现复杂的运动顺序。“凸轮盘”定义了跟随轴位置对引导轴位置的依靠关系。

凸轮盘还可用于定义液压轴的阀控曲线。

在 S7T Config 中组态“凸轮盘”工艺对象。

凸轮曲线可以在 S7T Config 中定义，也可以在用户程序中定义。

在 S7T Config 中定义凸轮

使用 CamEdit 或 Scout CamTool 程序，根据插补点或多项式定义凸轮。

在用户程序中定义凸轮

1. 在 S7T Config 中创建凸轮盘。
2. 使用“MC_CamClear”工艺功能删除和复位凸轮盘。
3. 使用工艺功能“MC_CamSectorAdd”，通过设置插补点或多项式定义凸轮。
4. 准备操作凸轮之前，使用“MC_CamInterpolate”工艺功能插补凸轮盘。

工艺功能

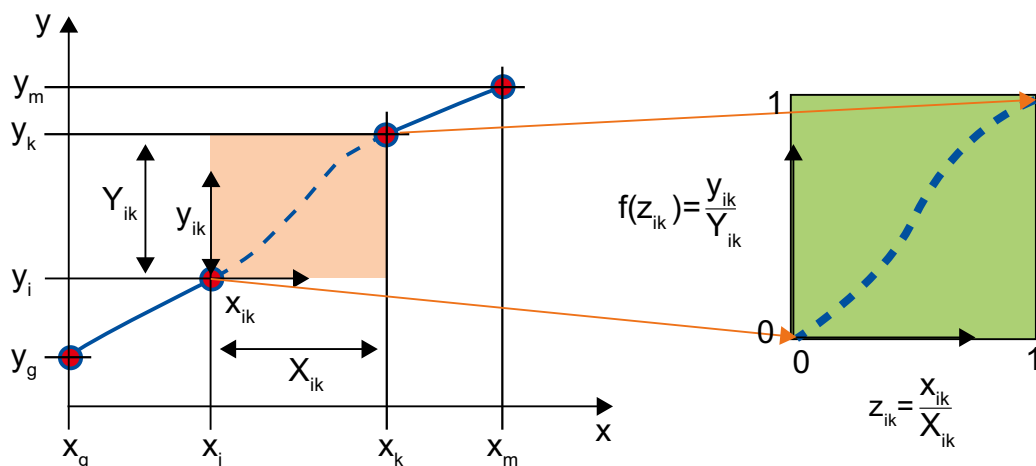
该工艺对象支持的工艺功能：

MC_Reset	MC_ReadSysParameter	MC_WriteParameter
MC_CamClear	MC_CamSectorAdd	MC_CamInterpolate
MC_GetCamPoint		

3.5.2 缩放

当在 S7T Config 中根据区段定义凸轮盘时，可以缩放到因子 1 的标准形式提供各个凸轮区段，这意味着取值范围和定义的范围对应闭区间 $[0,1]$ 。

将真实凸轮盘区段映射到缩放后的范围：



还可以在实数范围内输入这些区段。

缩放的优势

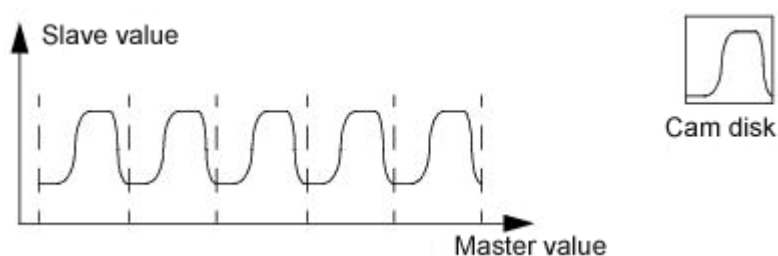
- 类似任务的运动的明确说明
- 独立于引导轴和跟随轴的实际单位和范围。

3.5.3 凸轮盘应用

可以组态凸轮盘，使其操作在非循环（单次操作）、相对循环（持续）或绝对循环（返回到插补点）模式下。

在绝对循环模式下操作的凸轮盘的示例

(x = 主设定值; y = 从设定值)

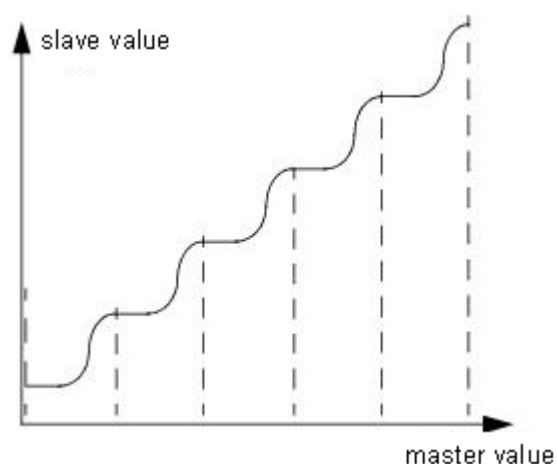


如果凸轮盘在循环模式下操作并且对于跟随轴来说是绝对的，则凸轮盘所有的连续起点和终点值都应当匹配。

任何不匹配的值都将导致在凸轮盘跳转时发生阶跃响应，从而触发同步监视。

在相对循环模式下操作的凸轮盘的示例

(x = 主设定值; y = 从设定值)

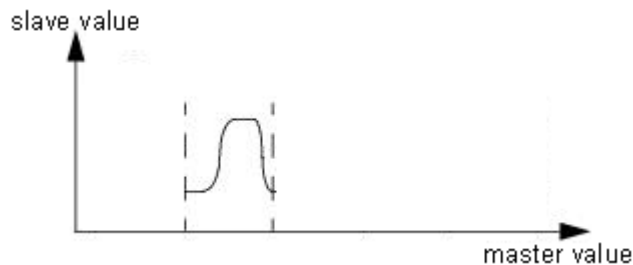


如果凸轮盘在循环模式下操作并且对于跟随轴来说是相对的，则凸轮盘的连续起点和终点值无需必须匹配。

在凸轮盘跳转时，系统将自动在终点设置起点值。这就使得在凸轮盘跳转中不会出现任何跳跃。

在非循环模式下操作的凸轮盘的示例

(x = 主设定值; y = 从设定值)



非循环模式意味着凸轮盘仅执行一次。当达到凸轮的终点或起点时终止凸轮传动

3.5.4 凸轮插补

凸轮盘的插补是在同步操作中使用凸轮盘或将其用作阀控曲线的基本要求。

凸轮盘的插补：

- 在 S7T Config 中 — 与凸轮盘一起下载到目标系统中。
- 通过调用“MC_CamInterpolate”工艺功能的用户程序。

在插补期间，系统执行跟随检查和校正：

- 检查定义范围（引导轴值范围）和（跟随轴的）取值范围中的连续性。此检查可防止对定义值进行多余的值分配。
- 填充了插补点和区段之间的间隔。根据插补模式对这些区域进行填充
- 填充缺失的便沿区域。根据插补模式和凸轮盘类型对这些区域进行填充。

当插补完成后，将为凸轮盘的定义范围分配一个明确的取值范围。

连续性检查

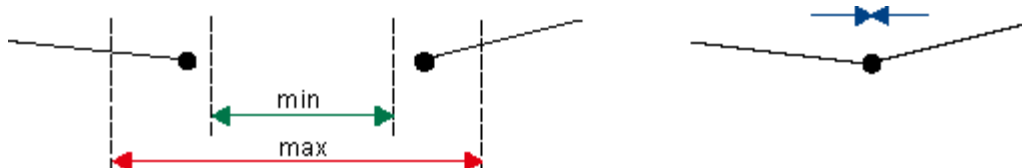
系统将检查定义范围和取值范围内凸轮盘的连续性，并校正发现的任何不连续性。

在该过程中，为定义范围/取值范围分别检查不连续点，并且为以下某一校正行为评估这些不连续点。

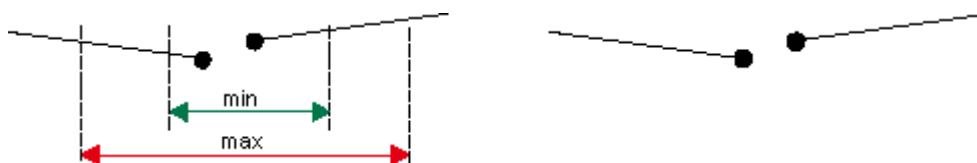
- 如果区段之间的间隔的绝对值超过最大值，则通过两个区段之间的插补进行校正。该操作将根据定义的插补模式插入一个新区段。如果在边沿区域插入一个区段，则凸轮盘类型将包括在该操作中。



- 如果区段之间的间隔的绝对值介于最小值和最大值之间，则通过将区段终点连接在一起进行校正。函数间隔的平均值用于进行校正。因此，区段的形状会受到影响。



- 如果区段或插补点之间的间隔的绝对值小于最小值，则不进行校正。保留不连续点。当访问该不连续点时，输出右边的边沿点。



3.5 “凸轮盘”工艺对象

当使用 CamEdit 进行插补时，可以在“插补”(Interpolation) 选项卡中设置最大值。“最小值”(minimum value) 对应于条目“忽略间隔”(ignore gaps)，而“最大值”(maximum) 对应于条目“连接点”(join points)。

使用“MC_CamInterpolate”工艺功能进行插补时，最小值和最大值都等于 *1e-004*。因此，只能通过使用 CamEdit 的插补连接区段的终点。

根据定义范围/取值范围的标定值来校正不连续点。

条件	结果
偏差 < 最小值	保留不连续性
最小值 < 偏差 < 最大值	连接区段点
偏差 > 最大值	插补（新区段）

通过最小和最大形状偏差的定义分别对定义范围和取值范围的校正进行控制。

- 可以通过线性插补来获得功能连续性
- 可通过样条插补获得偏差中的连续性

如果因为选定的插补方式或编程的几何图形而无法遵循连续性条件，则会输出一条消息来说明此影响。

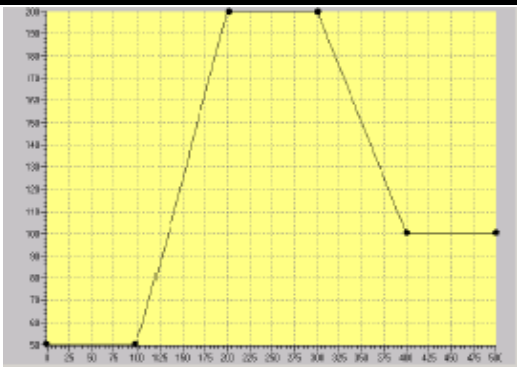
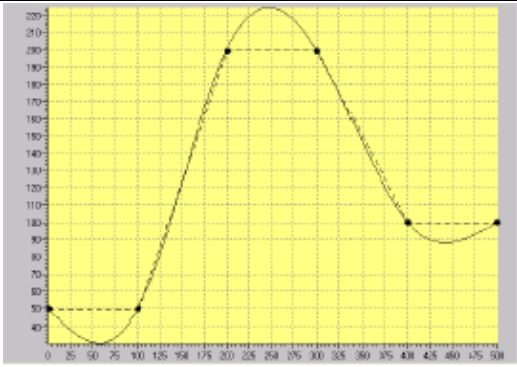
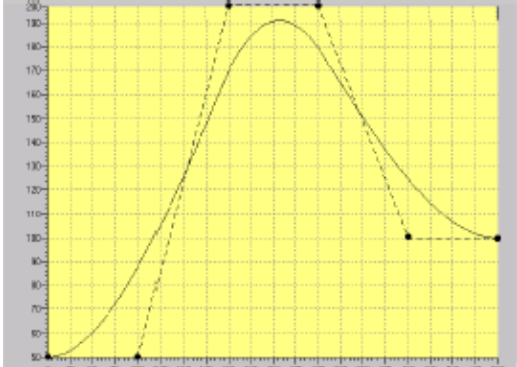
如果插补边沿点处于编程的几何图形内，则到达边沿点的所有几何图形元素都将被拒绝。如果插补边沿点处于编程的几何图形外，则根据使用的插补方式并考虑几何图形的特性来外插终点。

说明

在插补后，只能在使用（“MC_CamClear”）复位凸轮后插入新多项式或插补点。

3.5.5 插补类型

插补类型决定了间隔插补的规则。 可使用的插补类型：

插补类型	说明	示例
线性	线性插补： 凸轮曲线中的间隔通过“线性”操作来闭合，即使用直线。	
三次样条函数	使用三次样条函数的插补： 插补的凸轮曲线在凸轮插补点或定义的凸轮区段内运行。 当插补完成后，凸轮盘的取值范围（Y-min 到 Y-max）可能高于插补起点的值。	
贝塞尔样条函数	使用贝塞尔样条函数的插补： 插补的凸轮曲线沿凸轮插补点或凸轮区段运行。 插补不会更改凸轮盘的取值范围（Y-min 到 Y-max）。	

可使用“MC_CamInterpolate”工艺功能和 CamEdit 选择插补类型：

插补类型	"MC_CamInterpolate"	CamEdit —“插补”(Interpolation) 选项卡
线性	<i>Mode = 0</i>	“线性”(linear)
三次样条函数	<i>Mode = 1</i>	“三次样条函数”(cubic splines)
贝塞尔样条函数	<i>Mode = 2</i>	“贝塞尔样条函数”(Bezier splines)

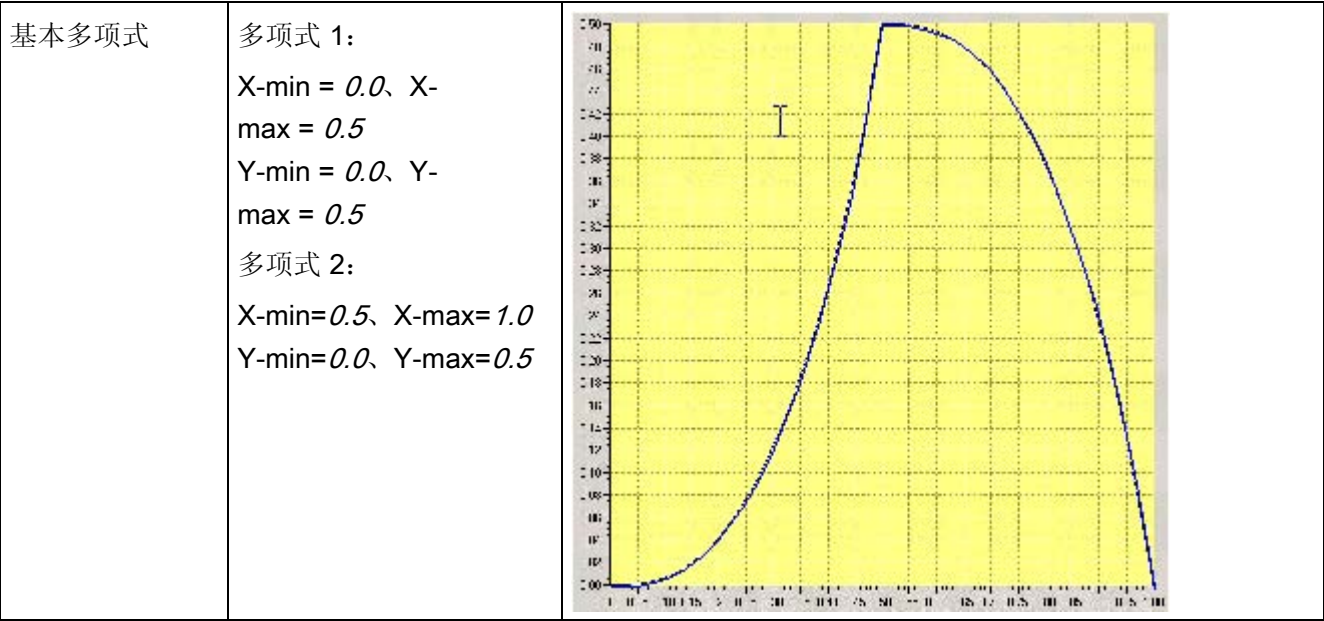
凸轮类型

凸轮盘类型说明了凸轮盘的应用，以及最终在凸轮盘的边沿所使用的不连续点插补。

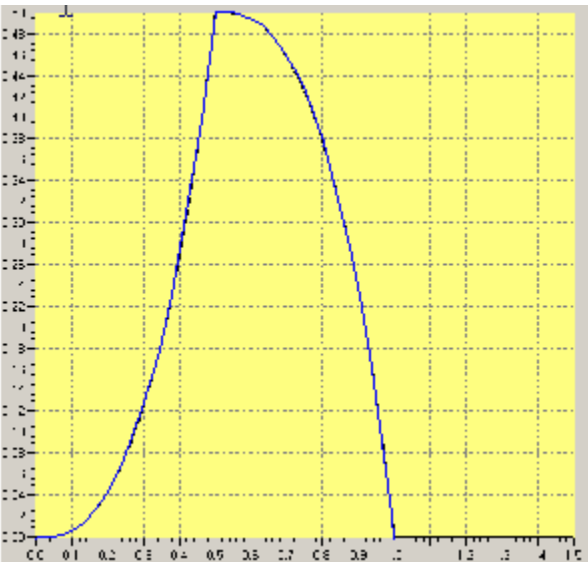
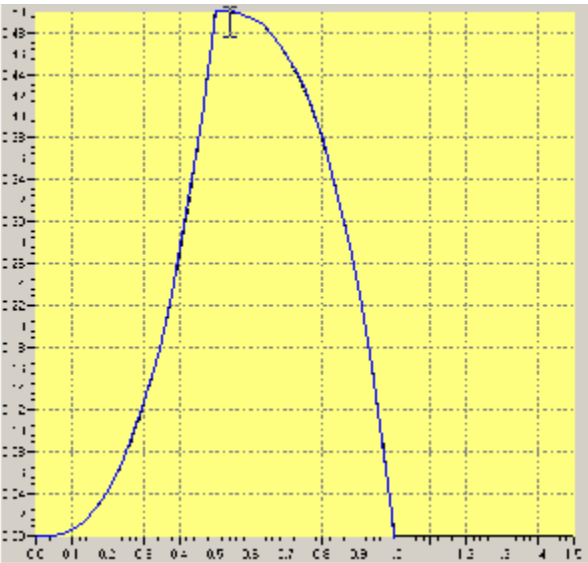
- **非循环**（边沿点不连续）
T-CPU 按照定义的情况使用凸轮盘，包括边沿上的所有不连续点。这也适用于凸轮盘操作于循环模式下时。但是，机械系统和驱动器的加速度限制及惯性起决定作用。
- **绝对循环**（边沿点位置和速度恒定）
插补凸轮盘，从而使其在循环模式下操作时边沿的位置和速度保持恒定。在凸轮边沿点，跟随轴的值和速度（凸轮斜度）完全相同。
- **相对循环**（边沿点速度恒定）
T-CPU 转换凸轮盘，从而使其边沿的速度在循环模式下保持恒定。凸轮沿的值和速度（凸轮斜度）完全相同。不对跟随轴的值进行校准。

在“绝对循环”或“相对循环”模式下，T-CPU 只能在凸轮盘的第一个定义的位置和定义范围的终点之间有足够的距离时才会计算凸轮盘。对于第一个定义的位置和定义范围的起点之间，该规则同样适用。

基于基本凸轮曲线的不同插补模式的示例：



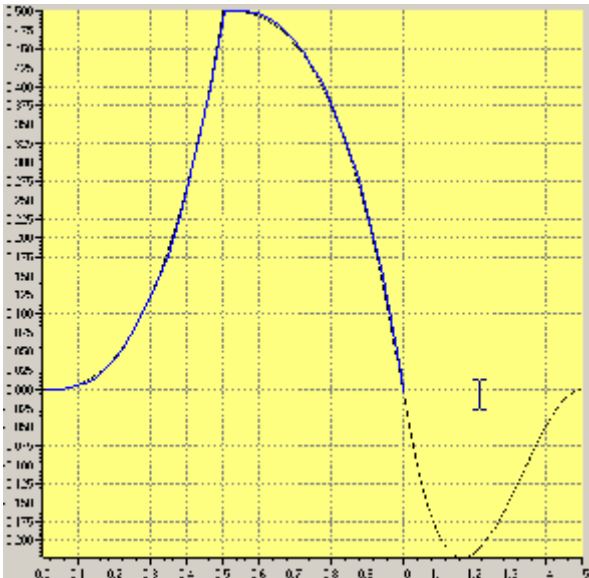
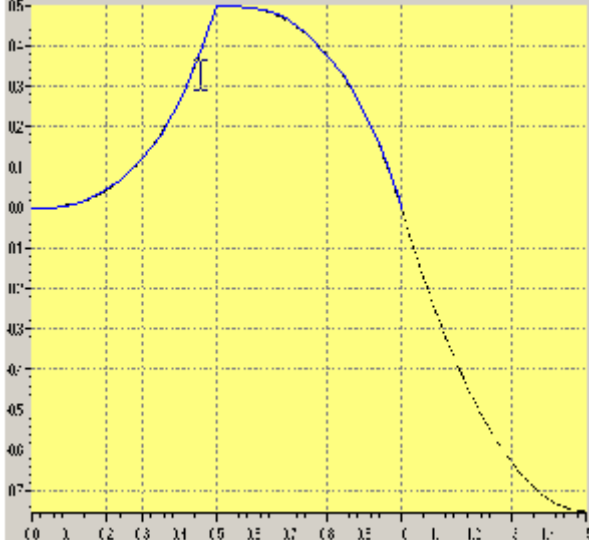
基本凸轮包括两个多项式。为了能够根据凸轮的给定插补模式（非循环模式、绝对循环模式和相对循环模式）插补凸轮，在后续内容中所显示的示例内，定义范围（引导轴的取值范围）的终点从 1.00 扩展到了 1.50。在该扩展的范围中，根据设置的插补模式来插补凸轮。

插补类型	说明	插补结果
线性	插补： 线性，非循环 输入参数 “MC_CamInterpolate”： $Mode = 0$ $CamMode = 2$ $StartPoint = 0.0$ $EndPoint = 1.5$ 结果： 基本凸轮保持不变。通过“线性”插补，沿横坐标扩展了其终点。	
	插补： 线性，绝对循环 输入参数 “MC_CamInterpolate”： $Mode = 0$ $CamMode = 1$ $StartPoint = 0.0$ $EndPoint = 1.5$ 结果： 基本凸轮保持不变；通过“线性”插补，相对于其起点，对其终点作了扩展	

3.5 “凸轮盘”工艺对象

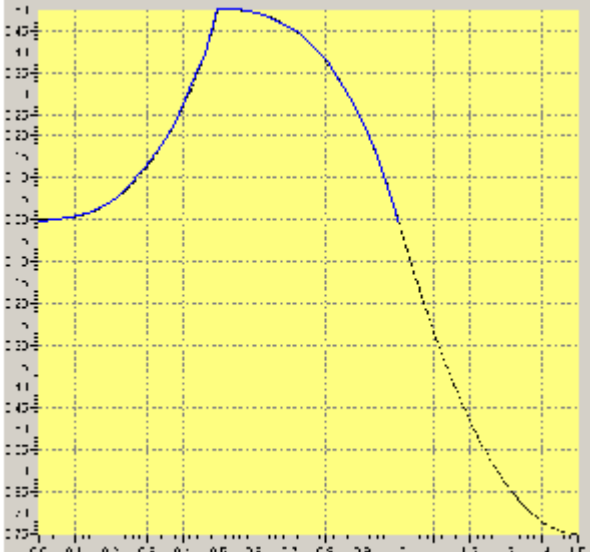
插补类型	说明	插补结果
	插补： 线性，相对循环 输入参数 “MC_CamInterpolate”： <i>Mode = 0</i> <i>CamMode = 0</i> <i>StartPoint = 0.0</i> <i>EndPoint = 1.5</i> 结果： 基本凸轮保持不变；根据其 初始斜度，沿线性坐标对凸 轮进行了扩展。	

插补类型	说明	插补结果
三次样条函数	插补： 三次样条函数， 非循环 输入参数 “MC_CamInterpolate”： <i>Mode = 1</i> <i>CamMode = 2</i> <i>StartPoint = 0.0</i> <i>EndPoint = 1.5</i> 结果： 基本凸轮保持不变。根据插补 模式，在扩展的定义范围内对 凸轮进行了扩展	

插补类型	说明	插补结果
	插补： 三次样条函数， 绝对循环 输入参数 “MC_CamInterpolate”： <i>Mode = 1CamMode = 1StartPoint = 0.0EndPoint = 1.5</i> 结果： 基本凸轮保持不变。根据插补模式，在扩展的定义范围内对凸轮进行了扩展；插补过的凸轮在恒定的位置，以恒定速度运行	
	插补： 三次样条函数， 相对循环 输入参数 “MC_CamInterpolate”： <i>Mode = 1CamMode = 0StartPoint = 0.0EndPoint = 1.5</i> 结果： 基本凸轮保持不变。根据插补模式，在定义范围内对凸轮进行了扩展；插补过的凸轮以恒定速度运行	

3.5 “凸轮盘”工艺对象

插补类型	说明	插补结果
贝塞尔样条函数	插补： 贝塞尔样条函数， 非循环 输入参数 “MC_CamInterpolate”： <i>Mode = 2</i><i>CamMode = 2</i><i>Start Point = 0.0</i><i>EndPoint = 1.5</i> 结果： 基本凸轮保持不变。根据插 补模式，在扩展的定义范围内 对凸轮进行了扩展	
	插补： 贝塞尔样条函数， 绝对循环 输入参数 “MC_CamInterpolate”： <i>Mode = 2</i><i>CamMode = 1</i><i>Start Point = 0.0</i><i>EndPoint = 1.5</i> 结果： 基本凸轮保持不变。根据插 补模式，在扩展的定义范围内 对凸轮进行了扩展；插补过的 凸轮在恒定的位置，以恒定速 度运行	

插补类型	说明	插补结果
	插补： 贝塞尔样条函数， 相对循环 输入参数 “MC_CamInterpolate”： <i>Mode = 2CamMode = 0Start Point = 0.0EndPoint = 1.5</i> 结果： 基本凸轮保持不变。根据插 补模式，在定义范围内对凸轮 进行了扩展；插补过的凸轮以 恒定速度运行	

条件：

- 在“MC_CamInterpolate”工艺功能中为 *StartPoint* 和 *EndPoint* 定义的任何不同值都可决定主值的范围。设置在定义的范围外的任何插补点或区段都将被截断。如果在 *StartPoint* 和 *EndPoint* 中设置了值 *0.0*，则不会保持插补凸轮的限制。
- 要在 *CamMode = 0* 或 *CamMode = 1* 的情况下插补凸轮盘，则在 *StartPoint* 或 *EndPoint* 中设置的值的范围应当始终大于插补凸轮的范围。如果不满足该条件，则无法根据默认参数进行插补。工艺功能“MC_CamInterpolate”将输出警告 *0027*。如果在 *CamEdit* 中插补了凸轮，则 *S7T Config* 将输出报警。

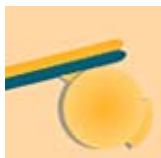
示例：

“MC_CamSectorAdd”工艺功能用于在 0 到 0.9 的定义范围内定义多项式。为了能够根据给定的插补规则在边沿区域中插补凸轮盘，应当在“MC_CamInterpolate”中将范围至少定义为 0 到 1.0。

- 如果要在“相对循环”(CamMode = 0) 或“绝对循环”(CamMode = 1) 模式下插补凸轮盘，则“线性插补”(Mode = 0) 没有任何意义。此时，应选择“三次样条函数”(Mode = 1) 或“贝塞尔样条函数”(Mode = 2)。
- 如果在工艺功能“MC_CamSectorAdd”中通过多项式定义多个区段，则应当始终确保多项式的起点与伙伴多项式的终点匹配，或者各个多项式之间存在足够的定义间隔。否则将无法插补。凸轮盘在插补后可能会显示阶跃响应。任何部分重叠区段的输入都会违背插补规则，并且将通过“MC_CamInterpolate”工艺功能输出警告 0027。凸轮盘可能在该位置生成一个阶跃
- 如果要在“非循环”模式 (CamMode = 2) 下插补凸轮盘，则“MC_CamSectorAdd”的定义范围 (X-min 到 X-max) 应当对应于“MC_CamInterpolate”工艺功能的定义范围 (StartPoint 到 EndPoint)。
- 相对循环插补的规则：
凸轮跳转时速度保持恒定
 $V(X-min) = V(X-max)$
- 绝对循环插补的规则：
凸轮跳转时速度和位置保持恒定
 $V(X-min) = V(X-max)$ 、
 $Y(X-min) = Y(X-max)$

3.6 “输出凸轮”工艺对象

3.6.1 “输出凸轮”工艺对象



使用“输出凸轮”工艺对象，基于轴位置生成控制信号。可以在用户程序中判断控制信号，也可以在数字输出中设置这些信号。支持的 I/O：

- 集成工艺的数字输出
- DP(DRIVE) 上标准从站的数字输出
- SINAMICS 端子模块 TM15 或 TM17 的数字输出

T-CPU 区分下面列出的一些输出凸轮：

- 基于位置的凸轮

基于位置的凸轮在起始位置和结束位置之间被激活。而在此范围外被取消激活。

- 切换输出凸轮

当到达起始位置时对切换凸轮进行设置，并且必须通过用户程序对其进行复位。

- 基于时间的凸轮

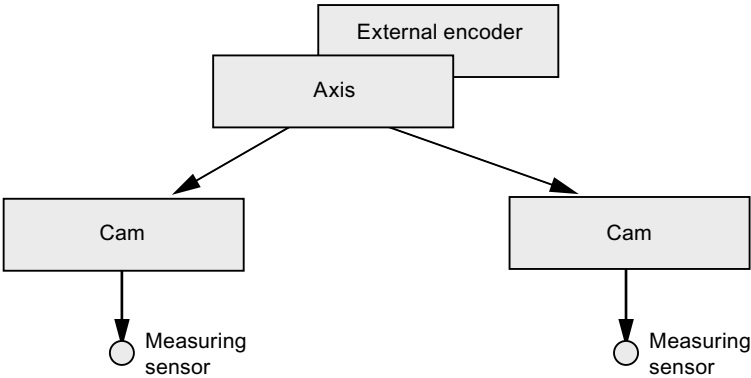
当到达起始位置时，设置基于时间的凸轮，持续时间为定义的时间段。

3.6 “输出凸轮”工艺对象

互连

“输出凸轮”工艺对象可与下列工艺对象互连：

- 定位轴
- 同步轴
- 外部编码器



实际位置或设定值位置的参考

输出凸轮的切换位置可以参考设定值或实际值（设定值输出凸轮、实际值输出凸轮）。
为了提高切换精度，可以在位置控制器周期内计算输出凸轮，或使用 T-CPU 的集成输出进行高速输出凸轮操作。

工艺功能

该工艺对象支持的工艺功能：

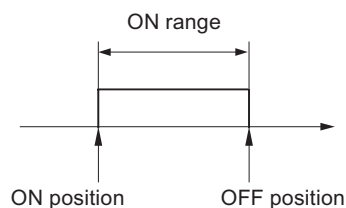
MC_Reset	MC_CamSwitch	MC_CamSwitchTime
MC_ReadSysParameter	MC_WriteParameter	

3.6.2 基于位置的凸轮

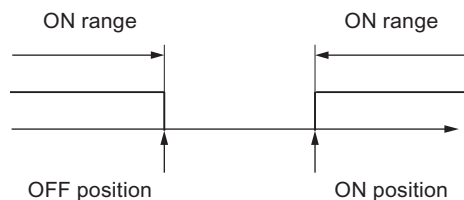
凸轮激活范围

基于位置的凸轮的凸轮激活范围由**正**计数方向上起始位置和结束位置之间的距离定义。

起始位置小于结束位置：



起始位置大于结束位置：



起始位置大于结束位置时，凸轮输出将反向。

切换特性

基于位置的凸轮的切换特性由起始位置和结束位置决定。

在“**输出凸轮**”工艺对象中，可以在工艺对象上另外定义“有效方向”。

基于以下各项的切换操作	活动的基于位置的凸轮	不活动的基于位置的凸轮
起始位置、结束位置	- 如果使用“MC_CamSwitch”启用输出凸轮，或使用“MC_CamTrack”启用凸轮轨迹 - 如果位置在凸轮激活范围内 - 如果位置值移到输出凸轮激活范围内	- 如果轴位置在起始区域或结束区域之外 - 当位置值移到凸轮激活范围外后 *
有效方向 (在凸轮轨迹未处于活动状态)	当位置在凸轮激活范围内并且有效方向 = 运动方向时	- 如果轴位置在起始区域或结束区域之外 - 如果运动方向与组态的有效方向不匹配 - 当位置值移到凸轮激活范围外后 *
滞后	当凸轮移动到滞后范围外时	当凸轮移动到滞后范围外时

* 互连对象的位置值可能突然变化，例如由于其参考发生变化或其坐标系移动。

3.6 “输出凸轮”工艺对象

反向输出

反向输出功能将凸轮激活范围和凸轮取消激活范围进行互换。

3.6.3 切换输出凸轮

切换特性

切换输出凸轮的切换操作由起始位置和运动方向决定。

操作:	条件:
切换输出凸轮激活	- 通过调用“MC_CamSwitch”工艺功能（输入参数 <i>Execute</i> 的上升沿）启用输出凸轮。 - 运动方向与有效方向（输入参数 <i>Direction</i>）一致 - 经过起始位置（输入参数 <i>OnPosition</i>）
切换输出凸轮取消激活	通过调用“MC_CamSwitch”工艺功能（输入参数 <i>Mode = 1</i> ）取消激活输出凸轮

说明

如果未通过凸轮起始位置（例如，由于实际值设置所限），则不会激活切换输出凸轮。

反向切换操作

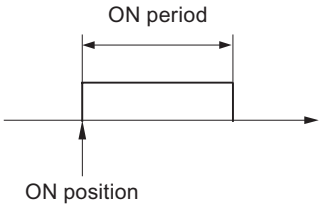
如果是通过“MC_CamSwitch”工艺功能禁用输出凸轮，则**激活**输出凸轮。

说明

如果运动方向 = 有效方向，或者输出凸轮通过“MC_CamSwitch”工艺功能激活，则输出凸轮将在凸轮起始位置**取消激活**。

3.6.4 基于时间的凸轮

切换特性



基于时间的凸轮的切换操作由起始位置和凸轮激活时间决定。

在“输出凸轮”工艺对象中，可以在工艺对象上另外定义有效方向。

基于以下各项的切换操作	基于时间的凸轮激活	基于时间的凸轮取消激活
起始位置	在起始位置	-
凸轮激活时间	-	超过编程时间时
有效方向 (在凸轮轨迹未处于活动状态)	在起始位置，如果运动方向 = 有效方向	超过编程时间时

说明

无法重新触发基于时间的凸轮。

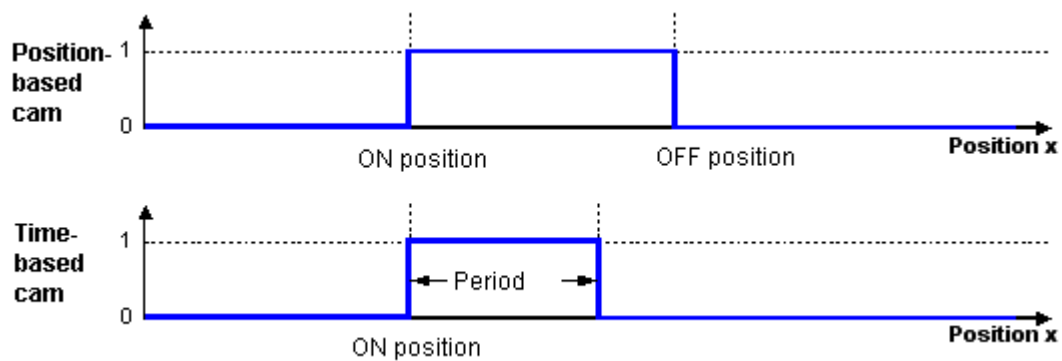
反向输出

为反向输出激活基于时间的凸轮，并且在指定的时间期间在凸轮取消激活位置将其取消激活。

3.6.5 有效方向和特性

响应

下图显示了输出凸轮的 ON 和 OFF 切换操作，没有滞后、微分时间，也没有凸轮取消激活时间。



切换操作仅取决于位置（位置设定值或实际位置）。

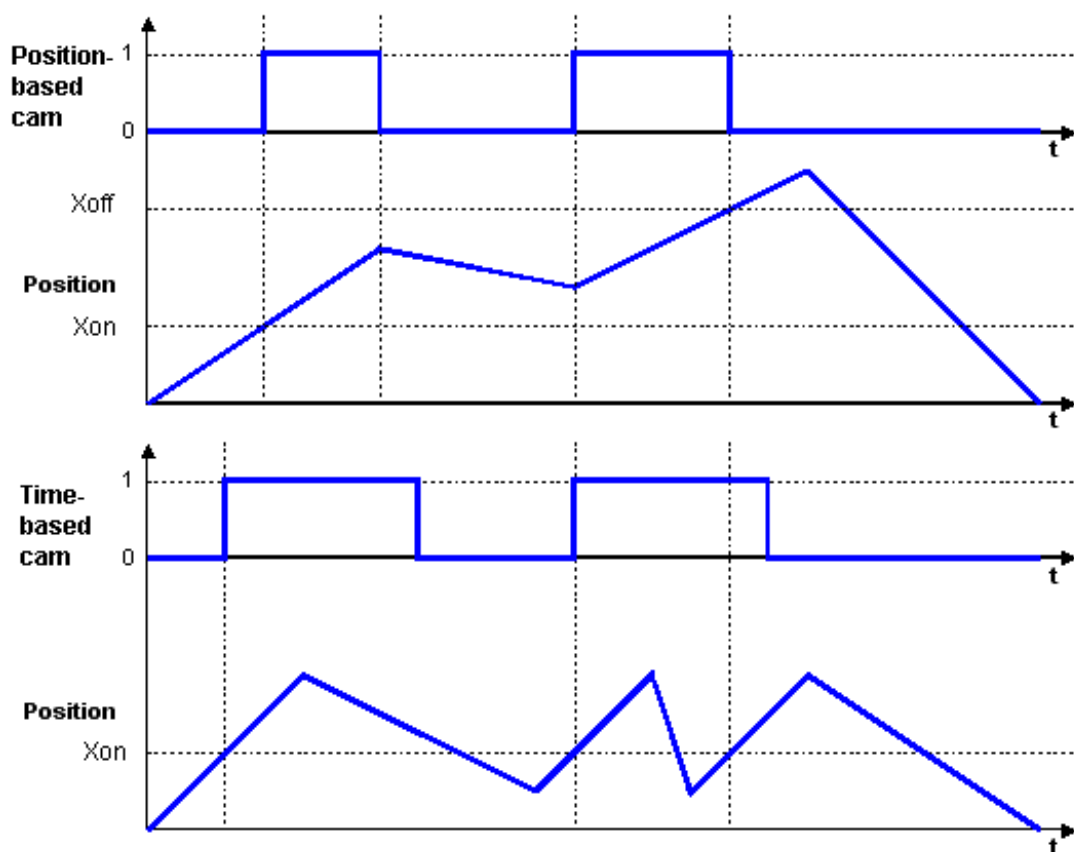
有效方向

当激活输出凸轮时可以定义一个默认的有效方向。仅当运动方向与设置的有效方向一致时，输出凸轮才会激活。

有两个选项：

有效方向	特性
正方向	输出凸轮仅在正运动方向上激活。
正方向和负方向	输出凸轮的激活与运动方向无关。
负方向	输出凸轮仅在负运动方向上激活。
当前旋转方向	使用该设置时，输出凸轮仅在当前设置的旋转方向上激活。停止状态为正旋转方向。

下图显示了有效方向 (= *Positiv*) 的结果：



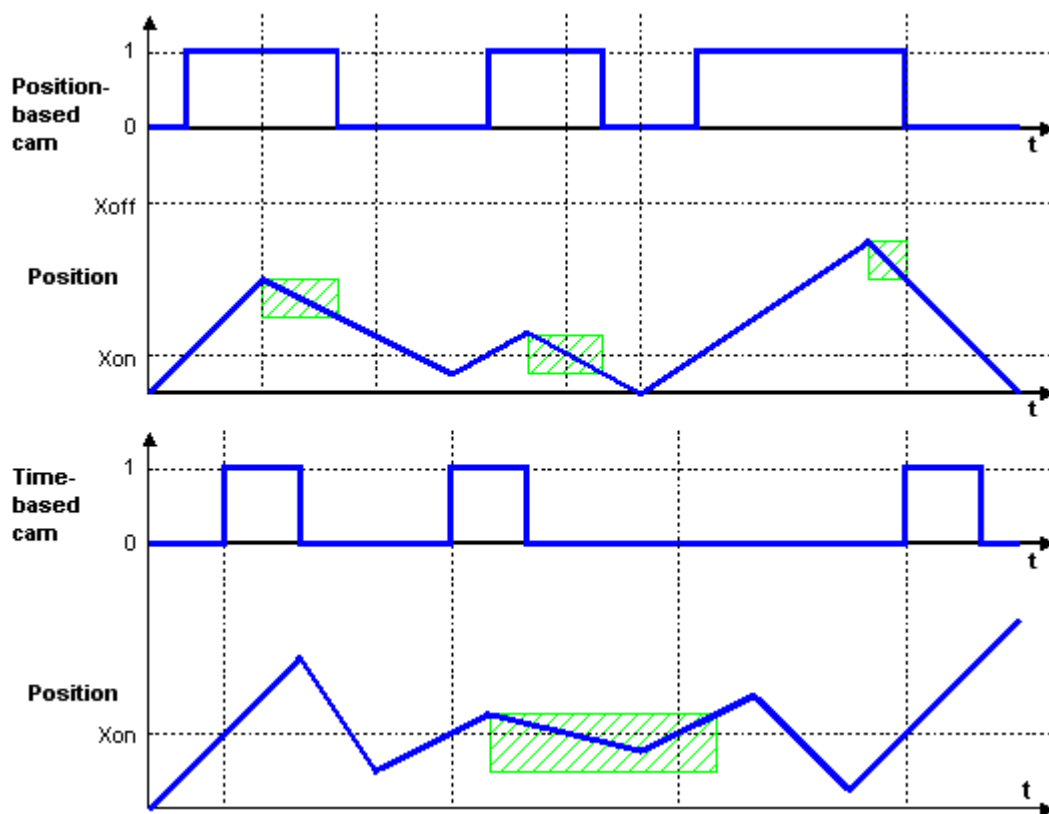
3.6.6 滞后

机械反向间隙可能导致实际位置值的波动，从而可能会导致输出凸轮的切换状态发生意外跳转。可以通过设置滞后来防止这种影响的产生。

滞后条件

- 当轴反转其方向时滞后始终处于活动状态。
- 在滞后范围内不会重新定义运动方向。
- 在滞后范围内，基于位置的凸轮的切换状态不会改变。
- 如果在凸轮移动到滞后窗口外后检测到输出凸轮切换条件被修改过，则将设置该当前切换状态。

下图显示了有效方向 (= positive) 的结果



绿色阴影区的高度对应滞后。

滞后范围

滞后的高度限制设置为模数轴范围的四分之一和非模数轴范围的四分之一。工作范围由软件限位开关定义。如果不存在软件限位开关，则无法限制工作范围。如果超出该最大设置，则会收到错误消息。

- **基于位置的凸轮的滞后**

在检测到换向后立即启用滞后。当基于位置的凸轮移动到滞后窗口外后将关闭这些凸轮。

- **基于时间的凸轮的滞后**

基于时间的凸轮的取消激活响应由凸轮激活时间决定，而不由滞后决定。

在“MC_CamSwitch”或“MC_CamSwitchTime”工艺功能的 *Hysteresis* 输入参数中设置该功能。

3.6.7 凸轮切换点的基于时间的偏移

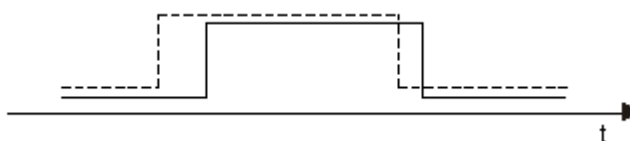
可以定义基于时间的偏移（指微分作用时间），以补偿机械响应时间、数字输出的上升时间、所连接的开关设备的响应时间和 DP(DRIVE) 上的通信时间。基于时间的偏移表示累积延迟时间。

可通过设置正值和负值来定义基于时间的偏移。如果要在输出凸轮到达已编程的起始位置之前切换输出凸轮，则必须输入负的凸轮激活时间。

微分作用时间在“MC_CamSwitch”或“MC_CamSwitchTime”工艺功能的输入参数 *Delay* 中设置。要最大化切换精度，请执行以下操作：

1. 根据恒定速度的多次测量，来决定系统延迟时间。
2. 在输入参数 *Delay* 中，将得出的平均值定义为基于时间的偏移。

对于与 T-CPU 集成输出互连的高速输出凸轮，如果使用的补偿时间适当，则获得的切换精度将为 $\pm 70 \mu\text{s}$ 。绝对切换精度（距离或角度）从设置的切换精度和当前速度中得出。



_____ 输入参数 *Delay* = 0（无基于时间的偏移）时的输出凸轮激活

- - - - - 输入参数 *Delay* < 0 时的输出凸轮激活

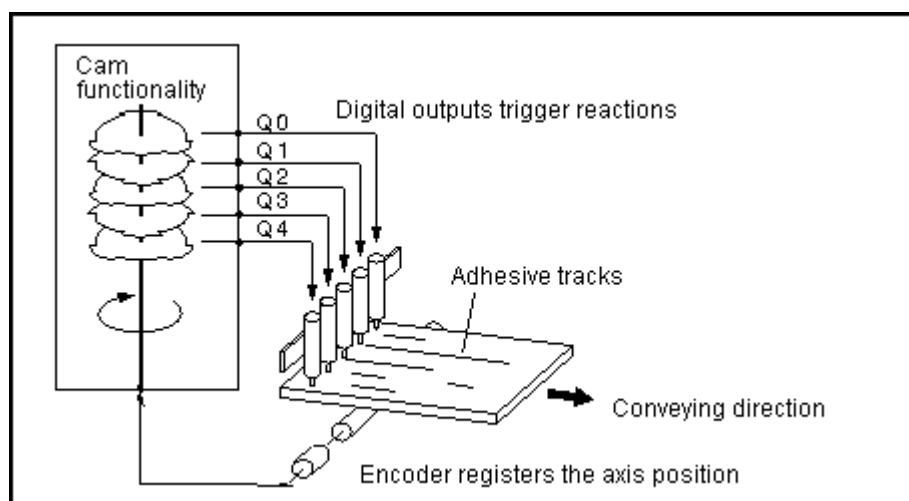
说明

注意：

- 该集成技术将动态切换位置计算为轴速度和定义的基于时间的偏移的函数。参考速度从输出凸轮激活时的轴速度中得出。在计算输出凸轮切换点时将忽略后续的速度变化。
- 计算的切换位置的偏移可能超出模数轴的长度。
- 超出一个模数周期的长微分作用时间可能导致实际值输出凸轮的切换位置（实际值曲线）的大幅波动。必要时，使用设定值输出凸轮。
- 模数轴的动态微分作用可大于一个模数长度。但是，系统不会收集切换操作。在任意给定的时间点，系统中有一个切换操作处于活动状态。当输出凸轮关闭后切换操作完成。

3.6.8 电子凸轮控制实例

粘贴线用于木板。将输出凸轮分配给外部编码器。分配给定义的输出的输出凸轮在定义的位置打开和关闭。



3.7 “凸轮轨迹”工艺对象

3.7.1 “凸轮轨迹”工艺对象



如果要在一个轴或外部编码器上使用多个相同类型的输出凸轮，请使用“凸轮轨迹”工艺对象。在一个凸轮轨迹中最多可使用 **32** 个专用输出凸轮。可在用户程序中评估专用输出凸轮的切换信号。可以将所有专用输出凸轮的切换状态最终切换为数字输出。可以使用下列 IO 设备：

- 集成工艺的数字输出
- DP(DRIVE) 上标准从站的数字输出
- SINAMICS 端子模块 TM15 或 TM17 的数字输出

可以为凸轮轨迹选择下列输出凸轮类型：

- 基于位置的凸轮

基于位置的凸轮在起始位置和结束位置之间被激活。而在此范围外被取消激活。

- 基于时间的凸轮

当到达起始位置时，设置基于时间的凸轮，持续时间为定义的时间段。

- 具有最大 ON 长度的基于时间的输出凸轮

对于具有最大 ON 长度的基于时间的凸轮，输出凸轮的 ON 时间段除了受持续时间限制外，还受最大 ON 长度限制。当超出 ON 时间段或最大 ON 长度时，输出凸轮将被关闭。

所选凸轮类型适用于凸轮轨迹的所有输出凸轮。基于位置的凸轮和基于时间的凸轮在凸轮轨迹中的切换特性与“输出凸轮”工艺对象的相同。

互连

“凸轮轨迹”工艺对象可与下列工艺对象互连：

- 定位轴
- 同步轴
- 外部编码器

实际位置或设定值位置的参考

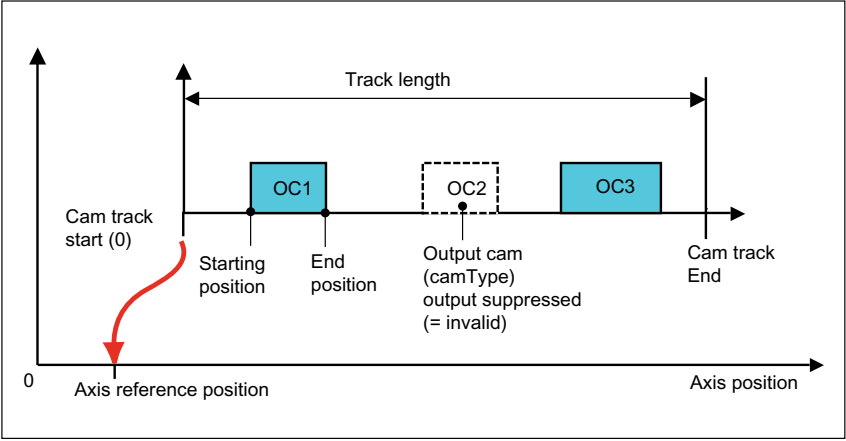
输出凸轮的切换位置可以参考设定值或实际值（设定值输出凸轮、实际值输出凸轮）。

为了提高切换精度，可以在位置控制器周期内计算输出凸轮，或使用 T-CPU 的集成输出进行高速输出凸轮操作。

表格 3-1 “输出凸轮”和“凸轮轨迹”工艺对象的比较

特性	输出凸轮	凸轮轨迹
支持的凸轮类型		
基于位置的凸轮	X	X
基于时间的凸轮	X	X
具有最大 ON 长度的基于时间的输出凸轮	-	X
单方向的输出凸轮	X	-
输出凸轮的互连		
多个相同凸轮类型的输出凸轮与一个数字输出互连	通过 S7T Config 中的 AND/OR	X
多个不同凸轮类型的输出凸轮与一个数字输出互连	通过 S7T Config 中的 AND/OR	-
输出凸轮属性		
滞后	X	X
有效方向	X	-
激活/取消激活的类型	立即	可调整
微分时间（开/关一次）	X	X
使用	循环	循环或非循环
输出凸轮或在工艺 DB 中映射的专用输出凸轮的状态	X	X
通过使用工艺功能启用输出凸轮	MC_CamSwitch MC_CamSwitchTime	MC_CamTrack
性能比较		
性能	性能取决于输出凸轮的数量。	专用输出凸轮数大于 5 时性能会提高

凸轮轨迹的参考量

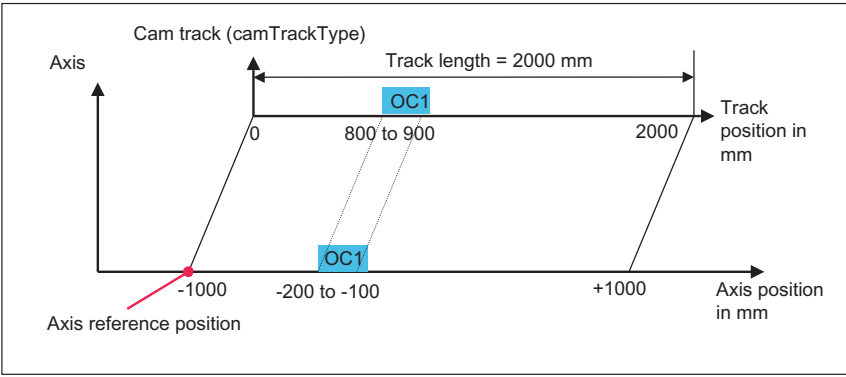


示例中的凸轮轨迹使用 3 个专用输出凸轮 (*N1 ... N3*) 进行定义。

专用输出凸轮的起始位置和结束位置在 **S7T Config** 的预分配中指定。各位置均参考相应的凸轮轨迹起点。

在示例中，输出凸轮 *N2* 被参数化为无效。无论作为硬件输出凸轮还是软件输出凸轮，均不考虑输出凸轮 *N2*。输出凸轮的有效性可以在 **S7T Config** 的预分配中参数化。

轨迹长度确定要使用的凸轮轨迹部分。凸轮轨迹在轴参考位置使用。调用“**MC_CamTrack**”工艺功能时确定要使用的凸轮轨迹。



凸轮轨迹始终根据轨迹长度在轴参考位置的正方向上创建。

工艺功能

该工艺对象支持的工艺功能：

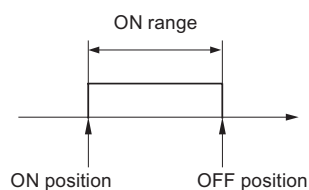
MC_Reset	MC_CamTrack	MC_ReadCamTrackData
MC_WriteCamTrackData	MC_ReadSysParameter	MC_WriteParameter

3.7.2 基于位置的凸轮

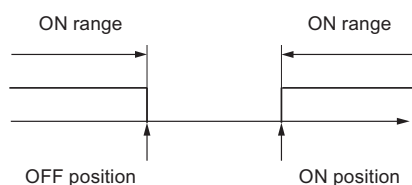
凸轮激活范围

基于位置的凸轮的凸轮激活范围由正计数方向上起始位置和结束位置之间的距离定义。

起始位置小于结束位置：



起始位置大于结束位置：



起始位置大于结束位置时，凸轮输出将反向。

切换特性

基于位置的凸轮的切换特性由起始位置和结束位置决定。

在“输出凸轮”工艺对象中，可以在工艺对象上另外定义“有效方向”。

基于以下各项的切换操作	活动的基于位置的凸轮	不活动的基于位置的凸轮
起始位置、结束位置	- 如果使用“MC_CamSwitch”启用输出凸轮，或使用“MC_CamTrack”启用凸轮轨迹 - 如果位置在凸轮激活范围内 - 如果位置值移到输出凸轮激活范围内	- 如果轴位置在起始区域或结束区域之外 - 当位置值移到凸轮激活范围外后 *
有效方向 (在凸轮轨迹未处于活动状态)	当位置在凸轮激活范围内并且有效方向 = 运动方向时	- 如果轴位置在起始区域或结束区域之外 - 如果运动方向与组态的有效方向不匹配 - 当位置值移到凸轮激活范围外后 *
滞后	当凸轮移动到滞后范围外时	当凸轮移动到滞后范围外时

* 互连对象的位置值可能突然变化，例如由于其参考发生变化或其坐标系移动。

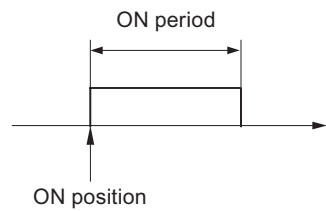
3.7 “凸轮轨迹”工艺对象

反向输出

反向输出功能将凸轮激活范围和凸轮取消激活范围进行互换。

3.7.3 基于时间的凸轮

切换特性



基于时间的凸轮的切换操作由起始位置和凸轮激活时间决定。

在“输出凸轮”工艺对象中，可以在工艺对象上另外定义有效方向。

基于以下各项的切换操作	基于时间的凸轮激活	基于时间的凸轮取消激活
起始位置	在起始位置	-
凸轮激活时间	-	超过编程时间时
有效方向 (在凸轮轨迹未处于活动状态)	在起始位置，如果运动方向 = 有效方向	超过编程时间时

说明

无法重新触发基于时间的凸轮。

反向输出

为反向输出激活基于时间的凸轮，并且在指定的时间期间在凸轮取消激活位置将其取消激活。

3.7.4 具有最大 ON 长度的基于时间的输出凸轮

切换特性

对于具有最大 ON 长度的基于时间的凸轮，除了考虑基于时间的凸轮功能外，还要考虑最大 ON 长度。

最大 ON 长度在 S7T Config 的凸轮轨迹预分配中定义。

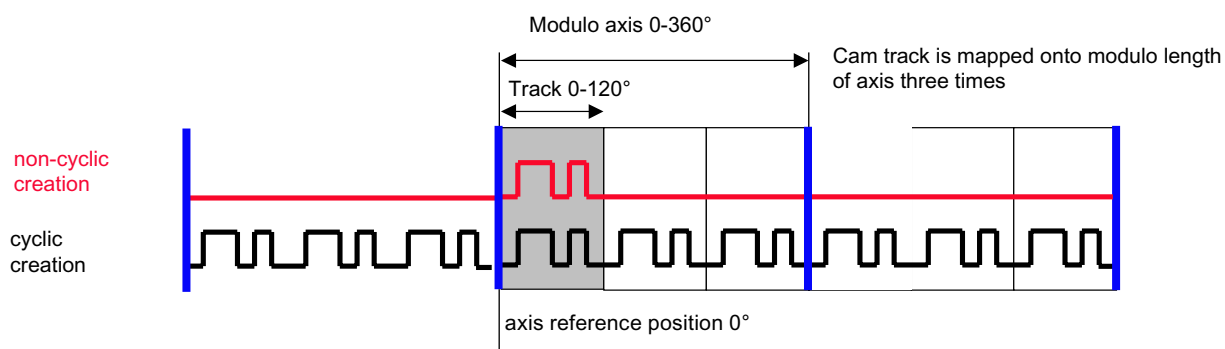
当过了组态时间或超出了最大 ON 长度时（取决于先发生的事件），具有最大 ON 长度的基于时间的凸轮将关闭。

最大 ON 长度在轴的两个行程方向上均有效，并且凸轮轨迹的打开位置为参考位置。

3.7.5 循环和非循环创建凸轮轨迹

可以在循环和非循环模式下，于轴参考位置创建凸轮轨迹。为此，使用工艺功能 "MC_CamTrack" 的输入参数 *CyclicMode*。

以下示例显示了以循环和非循环方式在轴参考位置上创建凸轮轨迹时的不同结果。本示例中使用具有凸轮轨迹（凸轮轨迹长度 120° ）的模数轴 ($0-360^\circ$)。凸轮轨迹在轴参考位置 0° 创建。



以非循环方式在轴参考位置创建凸轮轨迹

凸轮轨迹在轴参考位置创建一次。

首次沿正方向或负方向进入凸轮轨迹时，凸轮轨迹将变为活动状态。对于模数轴，可根据邻近模数周期中的轴位置和驱动方向进行创建和激活。

凸轮轨迹的取消激活取决于在 S7T Config 的凸轮轨迹组态的下拉列表框“在轨迹范围外激活非循环激活的凸轮轨迹”(Activate non-cyclic activated cam track outside the track range) 中的选择。

- 选择“是”(Yes) (默认设置)

即使在退出凸轮轨迹后（凸轮轨迹长度），凸轮轨迹也会保持活动状态。再次进入凸轮轨迹时，凸轮将重新切换。利用 *Mode = 1* 和工艺功能“MC_CamTrack”可以取消激活凸轮轨迹。

- 选择“否”(No)

退出凸轮轨迹后，凸轮轨迹将被取消激活。

以循环方式在轴参考位置创建凸轮轨迹

凸轮轨迹按照其轨迹长度在轴参考位置重复创建。凸轮轨迹在所有轴范围内为活动状态。利用 *Mode = 1* 和工艺功能“MC_CamTrack”可以取消激活凸轮轨迹。

说明

如果选择的凸轮轨迹长度和模数长度不适合，则会引起意外的非循环重叠。避免出现以下情况：

- 凸轮轨迹投影超出模数长度，或凸轮轨迹长度 > 模数长度
(对于循环和非循环创建凸轮轨迹)
- 模数长度与凸轮轨迹长度之间不是整数倍关系
(对于循环创建凸轮轨迹)

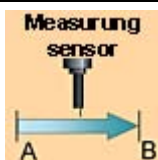


提示

对于循环创建凸轮轨迹，如果想仅在每隔一个模数周期中切换凸轮，则可选择长度为模数长度二倍的凸轮轨迹。

3.8 “测量输入”工艺对象

3.8.1 “测量输入”工艺对象



使用“测量输入”工艺对象进行精确、快速的实际位置值的记录。

可在 S7T Config 中将测量输入分配给轴或专用外部编码器。将传感器连接到驱动组件的数字输入或高性能型 TM15/TM17 模块。

通过在 CPU 的用户程序中调用运动控制命令“MC_MeasuringInput”启动实际位置值的测量。可通过设置起始值和结束值来相应地限制测量输入的操作范围。

实际位置值在测量输入信号的上升沿或下降沿记录和保存。倘若使用的组件支持该模式，则测量功能在两个边沿均可触发。

工艺功能

该工艺对象支持的工艺功能：

MC_Reset	MC_MeasuringInput	MC_ReadSysParameter
MC_WriteParameter		

3.8.2 测量输入的互连和连接

- 在 S7T Config 中，只能为“轴”和“外部编码器”工艺对象插入“测量输入”工艺对象。
- 每个“测量输入”工艺对象只能分配给单个轴或单个外部编码器
- 测量输入必须位于相应的驱动组件中，或高性能型 TM15/TM17 模块中。测量输入在相应的硬件中专门指出，或必须在那里组态为测量输入。可用测量输入数取决于使用的硬件。

DP(DRIVE) 的其它数字输入不可用作测量输入。

在 TM15/TM17 高性能型模块中用于“测量输入”工艺对象的数字测量输入在 S7T Config 中组态。

- 可以为定位轴、同步轴或外部编码器组态多个“测量输入”工艺对象。
“测量输入”工艺对象不可同时处于活动状态。
- 不能组态测量输入用于虚拟轴。
- 不能组态测量输入与速度控制轴一起运行。

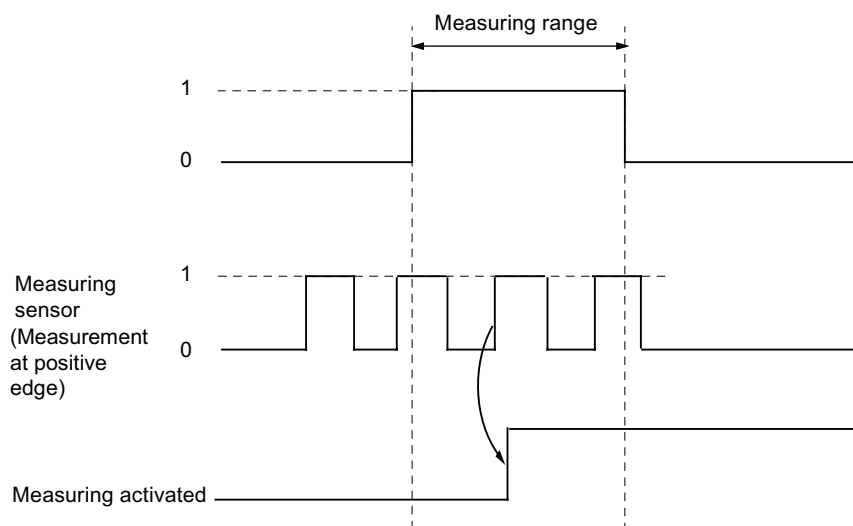
激活/取消激活

通过调用“MC_MeasuringInput”工艺功能激活/取消激活测量输入。

3.8.3 测量范围

测量命令可应用于整个范围，也可限制为位于起始位置和结束位置（活动范围）之内。

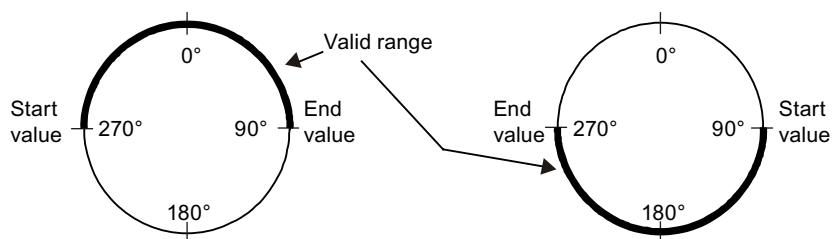
测量可限制到定义的范围中。仅当位置处于测量范围内时才触发测量。



如果未在测量范围内识别到触发信号，则取消测量命令，并在工艺功能和工艺 DB 中指示错误。

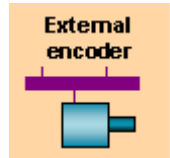
指定起始值和结束值的顺序与非模数轴无关。如果起始值大于结束值，则值互换。

如果模数轴的起始值大于结束值，则有效范围将从超过轴的模数跳转的起始值扩展到结束值。



3.9 “外部编码器”工艺对象

3.9.1 “外部编码器”工艺对象

The icon shows a blue rectangular block with a green line extending from its right side, representing a mechanical component or encoder. Above the block, the text "External encoder" is written in black.

使用“外部编码器”工艺对象，将机械组件的位置或角度作为主设定值返回到 T-CPU。

“外部编码器”工艺对象返回距离测量系统的实际位置值和速度值。该工艺对象不控制有效驱动组件。

工艺功能

该工艺对象支持的工艺功能：

MC_Reset	MC_ExternalEncoder	MC_ReadSysParameter
MC_WriteParameter		

3.9.2 外部编码器的互连和连接

互连

“外部编码器”技术对象可与以下技术对象互连：

- 作为引导轴的“同步轴”技术对象
- 作为实际位置值的“输出凸轮”技术对象
- 作为实际位置值的“测量输入”技术对象

接线

“外部编码器”技术对象支持的编码器有：

- 接线至 ET 200M 或 ET 200S 的模拟输入模块的模拟绝对编码器（传感器模拟）
- 接线至模拟驱动器接口 IM 174/ADI4 的增量编码器（长方形 TTL）
- 接线至模拟驱动器接口 IM 174/ADI4 的绝对编码器 (SSI)
- 接线至 DP 驱动器的编码器输入的增量编码器或绝对编码器
- 绝对编码器 SIMODRIVE 传感器等时线（消息帧 81）

示例

- 可以使用（可用于将外部编码器连接到 SIMODRIVE 611 U）双轴模块的第二个编码器接口。
- 可使用编码器模块将第二个编码器连接到 MASTERDRIVES MC
- 可以在 DP（驱动器）上直接操作等时 PROFIBUS 编码器
- SINAMICS S120 支持通过 SMC 30、SME 20、SME 25 连接第二个编码器

3.9.3 外部编码器的同步

T-CPU 支持多种回原点模式。可以在“MC_ExternalEncoder”工艺功能的输入参数 *Mode* 中设置外部编码器的参考位置。

使用增量编码器回原点

- 直接回原点（*Mode* = 2 和 4）
设置参考点。参考点的坐标值将被分配给当前编码器位置
- 被动回原点/实时回原点（*Mode* = 3 和 5）
当编码器以下列方式移动时，将参考点的坐标值分配给当前的编码器位置：
 - 到达编码器零标记后（默认设置）
 - 将要到达外部零标记时（在专家列表中进行设置）
 - 到达参考凸轮之后的下一个编码器零标记时（在专家列表中进行设置）可以使用编码器零标记监视功能来监视参考凸轮和编码器零标记之间的距离。
可以在专家列表中设置与增量编码器同步。

3.9 “外部编码器”工艺对象

组态

4.1 组态基本步骤

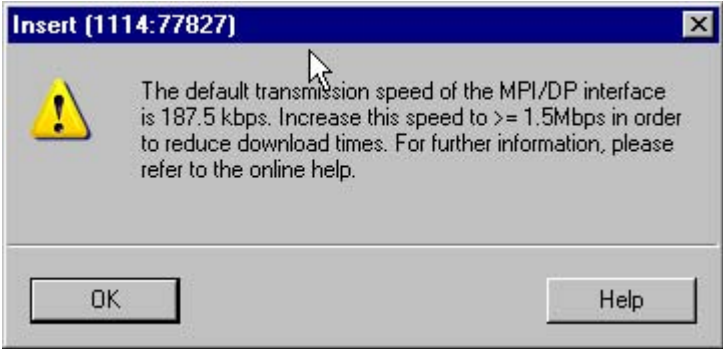
执行运动控制命令所需的步骤：

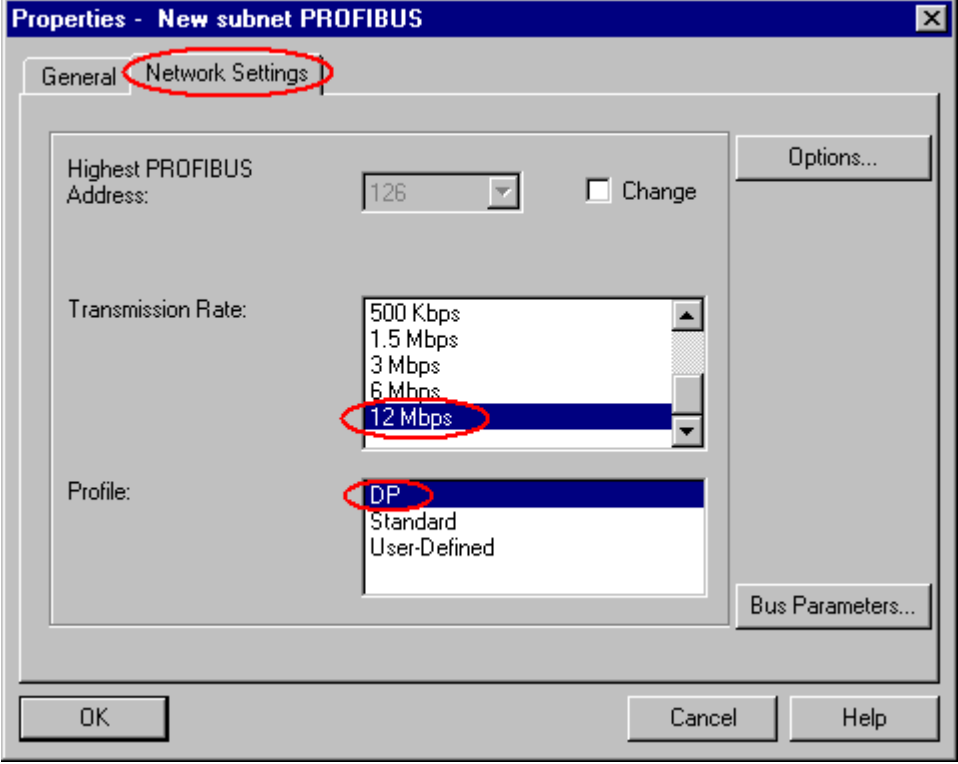
步骤	操作	工具
1.	创建项目	STEP 7 — SIMATIC Manager
2.	组态 T-CPU	HW Config
	组态驱动器	
	“站”(Station) > “保存并编译”(Save and compile) 菜单命令	
3.	组态轴	S7T Config
	组态工艺对象	
	“项目”(Project) > “保存并编译全部”(Save and compile all) 菜单命令	
4.	创建和管理工艺 DB	工艺对象管理
5.	对功能块进行编程	LAD/FBD/STL
6.	PLC > “将用户程序下载到存储卡” (Download user program to Memory Card) 菜单命令（包括系统数据）	STEP 7 — SIMATIC Manager

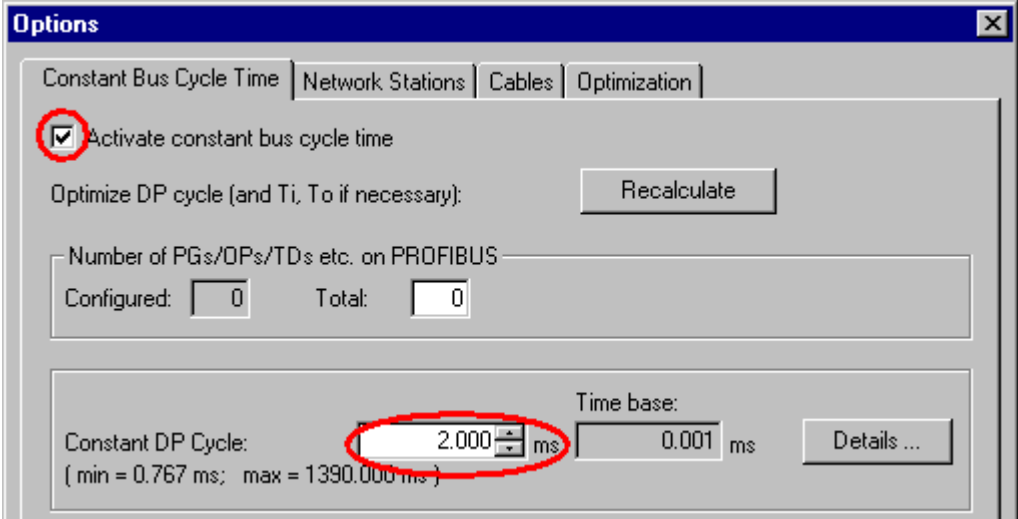
4.2 在 HW Config 中组态 T-CPU 和驱动器

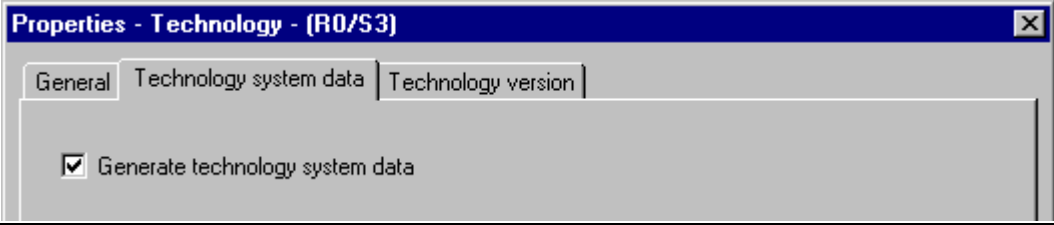
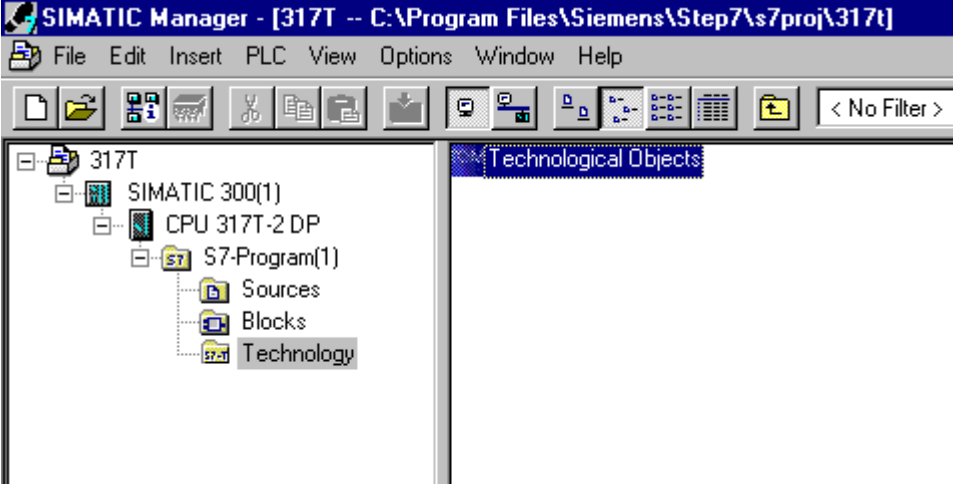
4.2.1 在 HW Config 中组态 T-CPU

要组态 T-CPU:

步骤	说明
1.	创建新的项目 SIMATIC Manager，并添加 SIMATIC 300 站。
2.	通过选择“SIMATIC 300”站并双击“硬件”(Hardware) 打开 HW Config。
3.	在“硬件目录”(Hardware catalog) 视图中，从“配置文件”(Profile) 下拉列表中选择“SIMATIC T-CPU”(SIMATIC Technology CPU) 硬件配置文件。 
4.	将安装导轨从“硬件目录”(Hardware catalog) 视图复制到 HW Config 的站窗口中。
5.	将电源模块拖放到该安装导轨（例如：“PS 307 5A”）中。
6.	将相关的 T-CPU 从 HW 目录中拖放到包含该导轨的选定行中。这将打开如下所示的消息框。  通过提高 MPI/DP 接口的传输率尽量减小传送时间。 有关详细信息，请参考消息框帮助。
7.	关闭消息框 在下一个对话框中设置 DP(DRIVE) 的 PROFIBUS 属性。

步骤	说明
8.	单击“新建”(New) 以创建新的“子网”(Subnet) (DP 主站系统)。 确保 T-CPU 是 DP(DRIVE) 上唯一的主站。 在 DP(DRIVE) 上设置地址 0，以提高操作性能。
9.	从 PROFIBUS 网络的属性对话框中选择“网络设置”(Network settings) 选项卡  选择 12 Mbps 的传输率。请勿更改 PROFIBUS 网络的“DP”配置文件。 要提高 DP(DRIVE) 系统的性能： - 在“最高 PROFIBUS 地址”(Highest PROFIBUS address) 中设置 0 值。 - 取消激活“启用总线参数的循环广播”(Enable cyclic broadcast of bus parameters) 选项。 单击“总线参数”(Bus parameters)，然后重设该复选框。

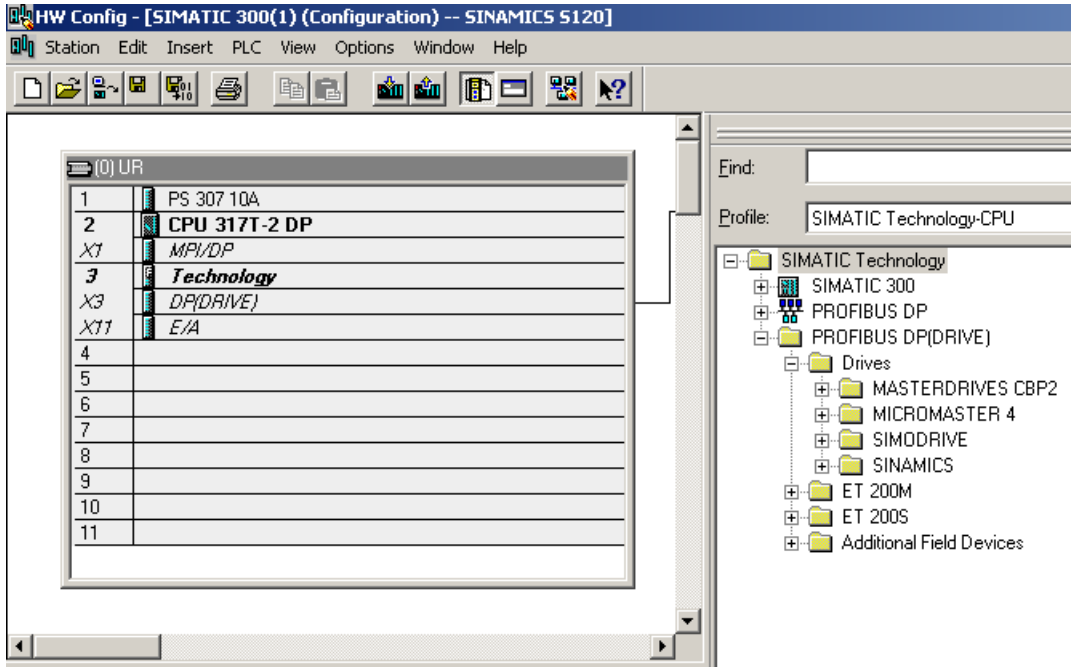
步骤	说明
10.	单击“选项”(Options)。 在下一个对话框中设置“激活恒定总线循环时间”(Activate constant bus cycle time) 复选框。  输入 DP(DRIVE) 上驱动组件等时操作所需的时间，作为恒定的 DP 总线周期时间的值。注意设备特定的属性和在 DP(DRIVE) 上操作所支持的组件数量框架。 保留该对话框中所有其它选项的默认设置。 注意： 也可以使用驱动器的属性对话框进行设置。随后，等距 DP 周期和周期时间将自动激活或调准。
11.	单击“确定”(OK) 关闭 HW Config 的所有对话框。
12.	提高 MPI/DP 接口的传输率（请参阅步骤 6）。保存并编译硬件配置数据，然后将这些数据下载至目标系统。

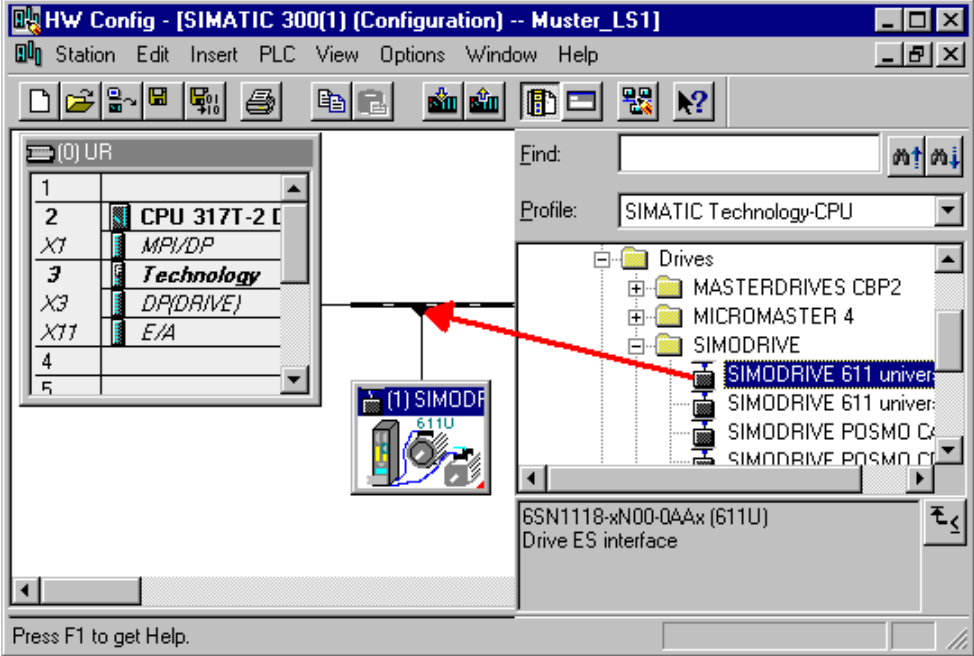
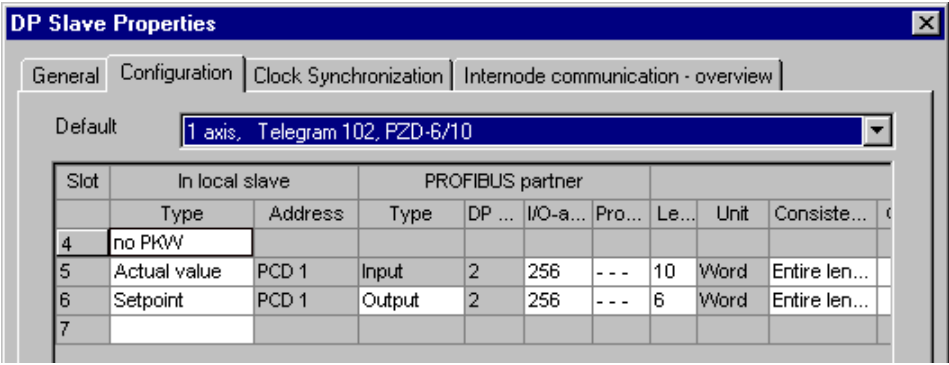
步骤	说明
13.	在 HW Config 中，双击“工艺”(Technology)（插槽 3），然后选择“工艺系统数据”(Technology System Data) 选项卡。设置“生成工艺系统数据”(Generate Technology System Data) 复选框。单击“确定”(OK)，确认设置。 
14.	保存并编译硬件配置数据，然后将这些数据下载至目标系统。调整 PG/PC 接口。 将集成工艺的硬件配置和固件数据下载至目标系统。 结果：编译数据后，SIMATIC Manager 中的“项目”(Project) 对话框将显示包含“工艺对象”(Technology Objects) 的“工艺”(Technology) 文件夹。 

4.2.2 在 HW Config 中组态驱动器

基本操作步骤

以下步骤以 SIMODRIVE 611 通用型为例介绍了组态驱动器的基本操作步骤。
有关驱动器的硬件组态的详细信息，请参考产品信息和相关的驱动器文档。

步骤	说明
1.	打开 HW 目录，然后打开文件夹 “SIMATIC 工艺”(SIMATIC Technology) > “PROFIBUS DP(DRIVE)”(PROFIBUS DP[DRIVE]) > “驱动器”(Drives) > “SIMODRIVE”。 

步骤	说明
2.	选择所需的驱动组件（例如：“SIMODRIVE 611U”）。将该组件拖放到 DP 主站系统。  “属性 – PROFIBUS 接口 SIMODRIVE 611U”(Properties – PROFIBUS interface SIMODRIVE 611U) 对话框随即打开。
3.	设置相关的 PROFIBUS 地址，例如地址 4。单击“确定”(OK)以确认。 “属性 — 驱动器”(Properties - Drive) 对话框随即打开。 单击“确定”(OK) 关闭该对话框。
4.	在“DP 从站属性”(DP Slave Properties) 对话框上选择“组态”(Configuration) 选项卡，然后选择所需的消息帧。  有关选择消息帧和支持的功能的详细信息，请参考“消息帧类型和功能”。 例如，如果 SIMODRIVE 611U 上某一轴用作不带具有扭矩减小的 DSC 的定位轴或同步轴，对于该轴的操作则选择消息帧 102。

步骤	说明
5.	选择“同步操作”(Synchronous operation) 选项卡，然后设置“将驱动器与等距离 DP 周期同步”(Synchronize drive with equidistant DP cycle) 复选框。 设置完 PROFIBUS 网络组态后，将激活并在该对话框上显示 DP 周期时间，DP 周期系数也将相应调整。 如果没有在网络属性中组态周期时间，则必须在该对话框中调整这些系数以获得相关的周期时间（例如：对于 2.00 ms 的 DP 周期，系数为 16）。 DP(DRIVE) 接口的 DP 周期始终为 0.5 ms 的整数倍，主要取决于该网络上操作的组件的数量和类型。 定义 DP 周期时间时，还要注意该对话框中可用的驱动器特定的帮助。
6.	在该例中设置“To”时间系数以获得 0.5 ms 的时间。请勿更改默认的“Ti”系数值。

步骤	说明
7.	单击“调整”(Alignment)。 该操作将触发以下设置： • 激活恒定的总线周期 • 将调整主站系统的 DP 周期时间，以适应驱动器属性 • 同一系列的所有驱动组件接受相同的参数组态
8.	单击“确定”(OK) 关闭该对话框。
9.	保存并编译硬件配置数据，然后将这些数据下载至目标系统。 如果已在工艺属性中设置了“生成工艺系统数据”(Generate technology system data)，则系统将为硬件组态和工艺固件生成系统数据。

说明

必须手动校准不同系列的驱动组件。要执行该操作，请记下设定值，然后将其传送至相应的对话框中。

4.2.3 消息帧类型及其功能

必须在 HW Config 中定义用于与驱动器进行通信的消息帧。要选择的消息帧由所需的轴功能（例如扭矩限制或 DSC）或所用驱动器的功能决定。

说明
在 HW Config 中更改驱动组件的消息帧类型时，还需要调整在 S7T Config 中的相应工艺对象的接口组态中选择的消息帧。

根据 PROFIdrive V3 规范，处理通过 PROFIBUS DP 与数字驱动器的通信。

支持的工艺和消息帧类型

表格 4- 1 消息帧类型和功能：

消息帧类型	简短说明/功能
1	n 设定值接口，16 位
2	n 设定值接口，32 位，不带编码器
3	n 设定值接口，32 位，带编码器 1
4	n 设定值接口，32 位，带编码器 1 和编码器 2
5	n 设定值接口，32 位，带 DSC 和编码器 1
6	n 设定值接口，32 位，带 DSC、编码器 1 和编码器 2
101	n 设定值接口，带扭矩减小
102	n 设定值接口，带编码器 1 和扭矩减小
103	n 设定值接口，带编码器 1、编码器 2 和扭矩减小
105	n 设定值接口，带 DSC、编码器 1 和扭矩减小
106	n 设定值接口，带 DSC、编码器 1、编码器 2 和扭矩减小

表格 4-2 驱动器和消息帧类型:

驱动组件	工艺对象	接口	可编程的标准消息帧类型
SIMODRIVE			
SIMODRIVE 611U 通用型	速度控制轴、定位轴、同步轴、外部编码器、测量输入、输出凸轮	数字	1 ... 6, 101, 102, 103, 105, 106 **
SIMODRIVE 611U 通用型 HR	速度控制轴、定位轴、同步轴、外部编码器、测量输入、输出凸轮	数字	1 ... 6, 101, 102, 103, 105, 106 **
SIMODRIVE POSMO CA/CD	速度控制轴、定位轴、同步轴、外部编码器、测量输入、输出凸轮	数字	1 ... 6, 101, 102, 103, 105, 106
SIMODRIVE POSMO SI	速度控制轴、定位轴、同步轴、测量输入、输出凸轮	数字	1, 2, 3, 5, 101, 102, 105
SIMODRIVE 传感器同步	外部编码器	数字	81
MICROMASTER 4			
COMBIMASTER 411	速度控制轴	数字 *	1
MICROMASTER 420	速度控制轴	数字 *	1
MICROMASTER 430	速度控制轴	数字 *	1
MICROMASTER 440	速度控制轴	数字 *	1
MASTERDRIVES			
运动控制	速度控制轴、定位轴、同步轴、外部编码器、测量输入、输出凸轮	数字	1 ... 6
运动控制补充	速度控制轴、定位轴、同步轴、外部编码器、测量输入、输出凸轮	数字	1 ... 6
矢量控制 CUVC	速度控制轴	数字 *	1, 2
矢量控制 Plus	速度控制轴	数字 *	1, 2

驱动组件	工艺对象	接口	可编程的标准消息帧类型
SINAMICS			
SINAMICS S120	速度控制轴、定位轴、同步轴、外部编码器、测量输入、输出凸轮	数字	1 ... 6, 102, 103, 105, 106
TM15 端子模块	测量输入 输出凸轮	数字	-
TM17 高性能型端子模块	测量输入 输出凸轮	数字	-
SINUMERIK			
ADI4	速度控制轴、定位轴、同步轴、外部编码器、测量输入、输出凸轮	数字	3
SIMATIC 分布式 I/O			
PROFIBUS 模块 IM 174	速度控制轴、定位轴、同步轴、外部编码器、测量输入、输出凸轮	数字	3
ET 200M、ET 200S	输出凸轮	数字	-
* 该驱动器不支持 PROFIBUS 上的等时操作。 ** 有关详细信息，请参考 611 U 的补充说明			

4.3 工艺对象管理

4.3.1 启动工艺对象管理

使用 S7T Config，为运动控制任务组态工艺对象。

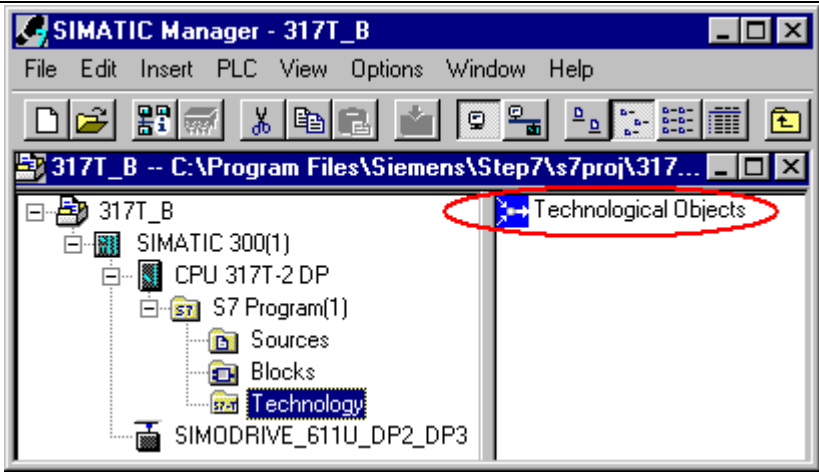
工艺 DB 构成了用户程序和工艺对象之间的接口。在“工艺对象管理”(TOM, Technology Objects Management) 中管理工艺 DB。

“工艺对象管理”(Technology Objects Management) 可与 SIMATIC Manager 同时使用。

先决条件

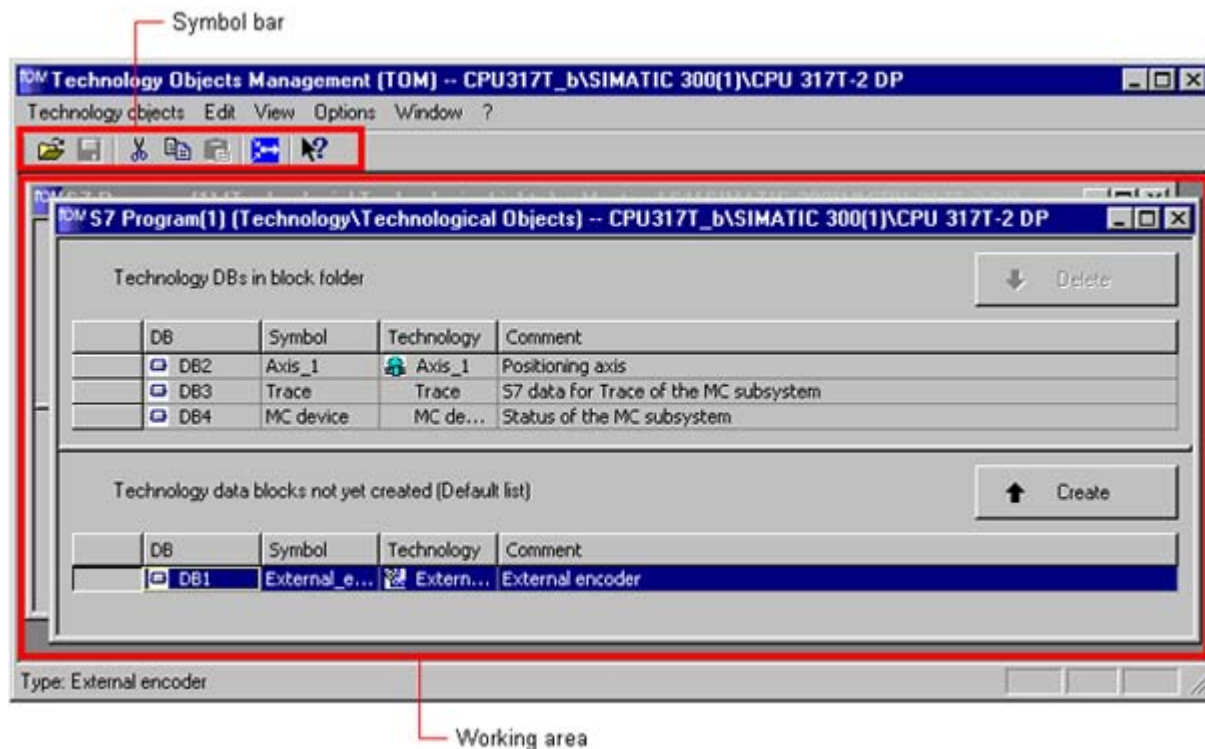
- SIMATIC Manager 处于打开状态
- 在 HW Config 中组态了 T-CPU，并保存了组态数据

启动工艺对象管理

步骤	含义
1.	从 SIMATIC Manager 的“工艺”(Technology) 文件夹中选择“工艺对象”(Technology objects)，以运行“工艺对象管理”(Technology Objects Management)。
2.	 选择“工艺对象”(Technology Objects)，然后在 SIMATIC Manager 中选择编辑 (Edit) > 打开对象 (Open object) 命令。 不运行“工艺对象管理”(Technology Objects Management) 也可以启动 S7T Config。
3.	从 SIMATIC Manager 的“工艺”(Technology) 文件夹中选择“工艺对象”(Technology objects)，然后选择选项 (Options) > 组态工艺 (Configure technology)。 如果不想创建任何新的工艺对象，那么使用 SIMATIC Manager 中的菜单命令启动 S7T Config 将非常有用，这样无需创建任何新的工艺 DB。

4.3.2 工艺对象管理的用户界面

“工艺对象管理”(Technology Objects Management) 的布局:



工具栏

工具栏包含基本的菜单命令。

工作区域

每个 STEP 7 项目都在工作区域的一个单独工作窗口中打开。每个窗口都显示现有的所有工艺 DB:

- 该窗口的上部显示已创建的工艺 DB。
- 工作区域的底部列出了尚未生成任何工艺数据块的工艺对象。

4.3.3 使用工艺对象管理

“工艺对象管理”(Technology Objects Management) 用于管理 STEP 7 项目的工艺 DB。

可以在“工艺对象管理”(Technology Objects Management) 的工作区域上，为包含 T-CPU 的每个已组态的站打开一个独立窗口。

在活动窗口上，您可以：

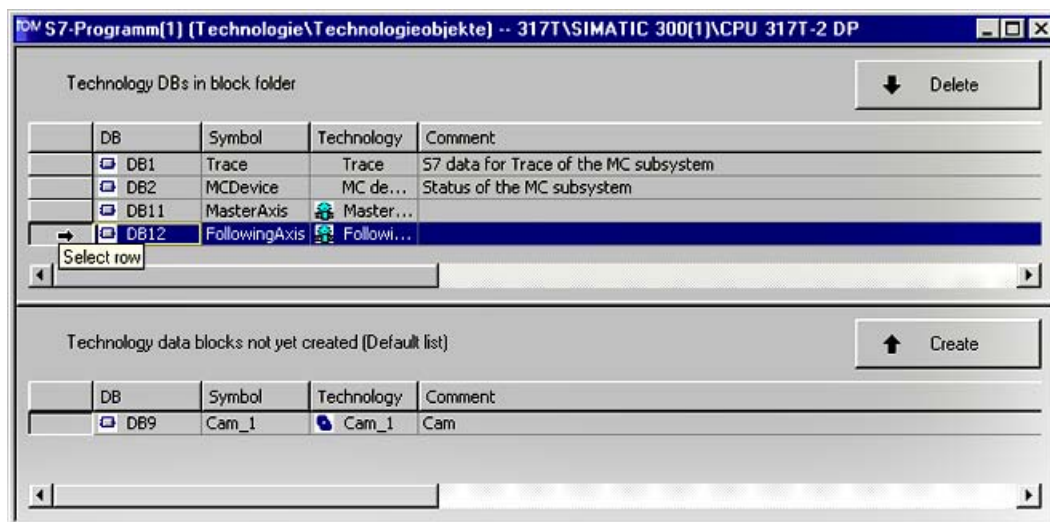
- 创建、删除工艺 DB，或对工艺 DB 进行编程
- 在轴中设置 IPO 同步模式（视图 [View] > 专家模式 [Expert mode] 菜单命令）
- 定义工艺 DB 的更新周期（专家模式）
- 组态轴命令的启动参数（专家模式中的“IPO 同步”属性）
- 启动 S7T Config

先决条件

- 已在 S7T Config 中完成、保存和编译了 T-CPU 的组态。

创建或删除工艺 DB

选择要创建或删除的一个或多个工艺 DB：



然后，单击“创建”(Create) 或“删除”(Delete)。

说明

有关使用“工艺对象管理”(Technology Objects Management) 的详细信息，请参考相应的“在线帮助”系统。

启动 S7T Config

通过选择选项 (Options) > 组态工艺 (Configure technology) 命令启动 S7T Config。

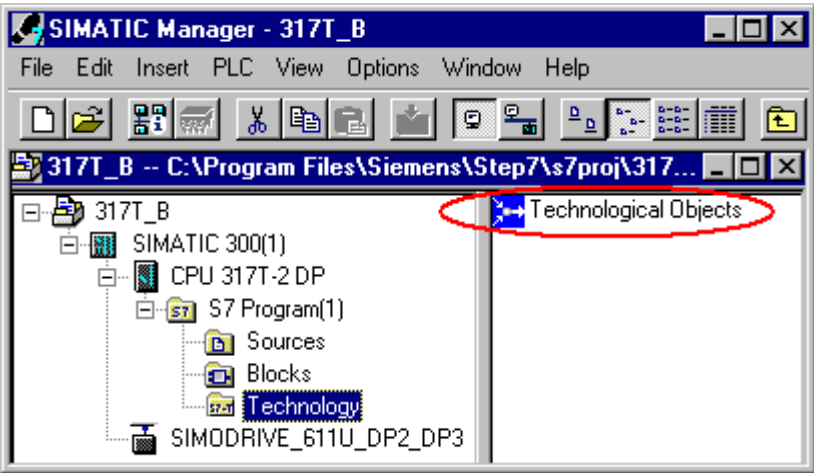
4.3.4 创建和管理工艺 DB

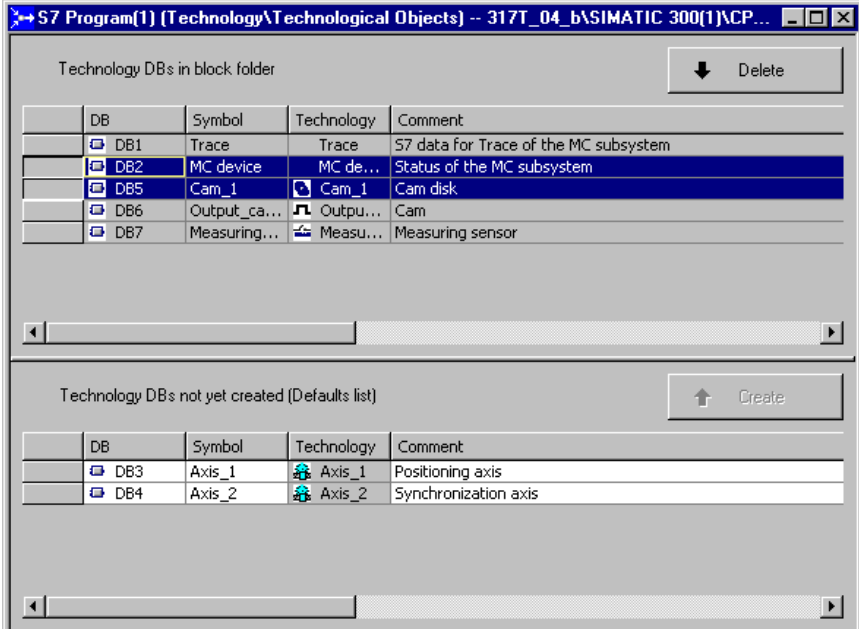
可通过工艺 DB 访问“工艺对象 (TO, Technology objects)”。因此，应在创建用户程序之前使用“工艺对象管理”(Technology Objects Management) 工具生成工艺 DB。

先决条件

- 已通过执行站 (Station) > 保存并编译 (Save and compile) 命令在 HW Config 中完成了 T-CPU 和驱动器的组态。
- 已在 S7T Config 中成功完成并保存了 T-CPU 的组态。

示例：重命名工艺 DB

步骤	说明
1.	在 SIMATIC Manager 中，双击“工艺对象”(Technological objects)。 The screenshot shows the SIMATIC Manager interface. The title bar reads 'SIMATIC Manager - 317T_B'. The menu bar includes File, Edit, Insert, PLC, View, Options, Window, and Help. Below the menu is a toolbar with various icons. The main window displays a project tree on the left. The tree structure is as follows: 317T_B (root) -> SIMATIC 300(1) -> CPU 317T-2 DP -> S7 Program(1) -> Sources, Blocks, Technology (highlighted with a red box). Other items in the tree include SIMODRIVE_611U_DP2_DP3. On the right side of the window, there is a pane titled 'Technological Objects' which is also highlighted with a red box. The status bar at the bottom shows the file path: C:\Program Files\Siemens\Step7\proj\317...

步骤	说明
	“工艺对象管理”(Technology Objects Management) 对话框打开。（示例）：  选择要创建的 DB。
2.	从“块文件夹中的工艺 DB”(Technology DBs in block folder) 区域中，选择相关的行以重命名其 DB。
3.	单击“删除”(Delete)
4.	在“尚未创建的工艺 DB”区域中编辑块编号。
5.	选择显示您要创建的 DB 的行。
6.	单击“创建”(Create)

说明

从 CPU 中删除所有先前下载的具有该名称的工艺 DB。然后，可以将重命名的具有新 DB 编号的工艺 DB 下载至 CPU 中。

说明

- 必须验证在 SIMATIC Manager 的块文件夹中复制的工艺 DB。在下列情况下，相应的 DB 不能用作有效的工艺 DB：
- 在上述对话框中标记为红色
在这种情况下，选择“块文件夹中的工艺 DB”(Technology DBs in block folder)，然后单击“删除”(Delete)。
 - 在上述对话框中没有列出
在这种情况下，从 SIMATIC Manager 的块文件夹中删除该块。

4.4 使用 S7T Config

4.4.1 启动 S7T Config

先决条件

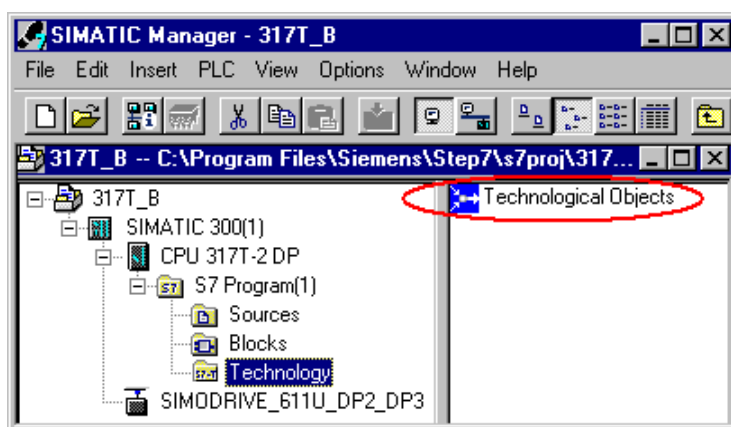
- SIMATIC Manager 处于打开状态
- 在 HW Config 中组态了 T-CPU，并保存了组态数据

启动 S7T Config

S7-Technology 选项数据包中随附的 S7T Config 提供了合适功能，可用于对运动控制任务的工艺对象进行编程。

通过从 SIMATIC Manager 的“工艺”(Technology) 文件夹中选择“工艺对象”

(Technological objects) 启动 S7T Config。只有在 HW Config 中组态了 T-CPU 并且编译了所有组态数据时，该对象才可见。如果驱动器和外部编码器的名称包含任何空格，将在 S7T Config 启动时对它们进行重命名。将用下划线替换这些空格。



选择“工艺对象”(Technology objects)，然后选择**编辑 (Edit) > 打开对象 (Open Object)** 以运行“工艺对象管理”(Technology Objects Management)。

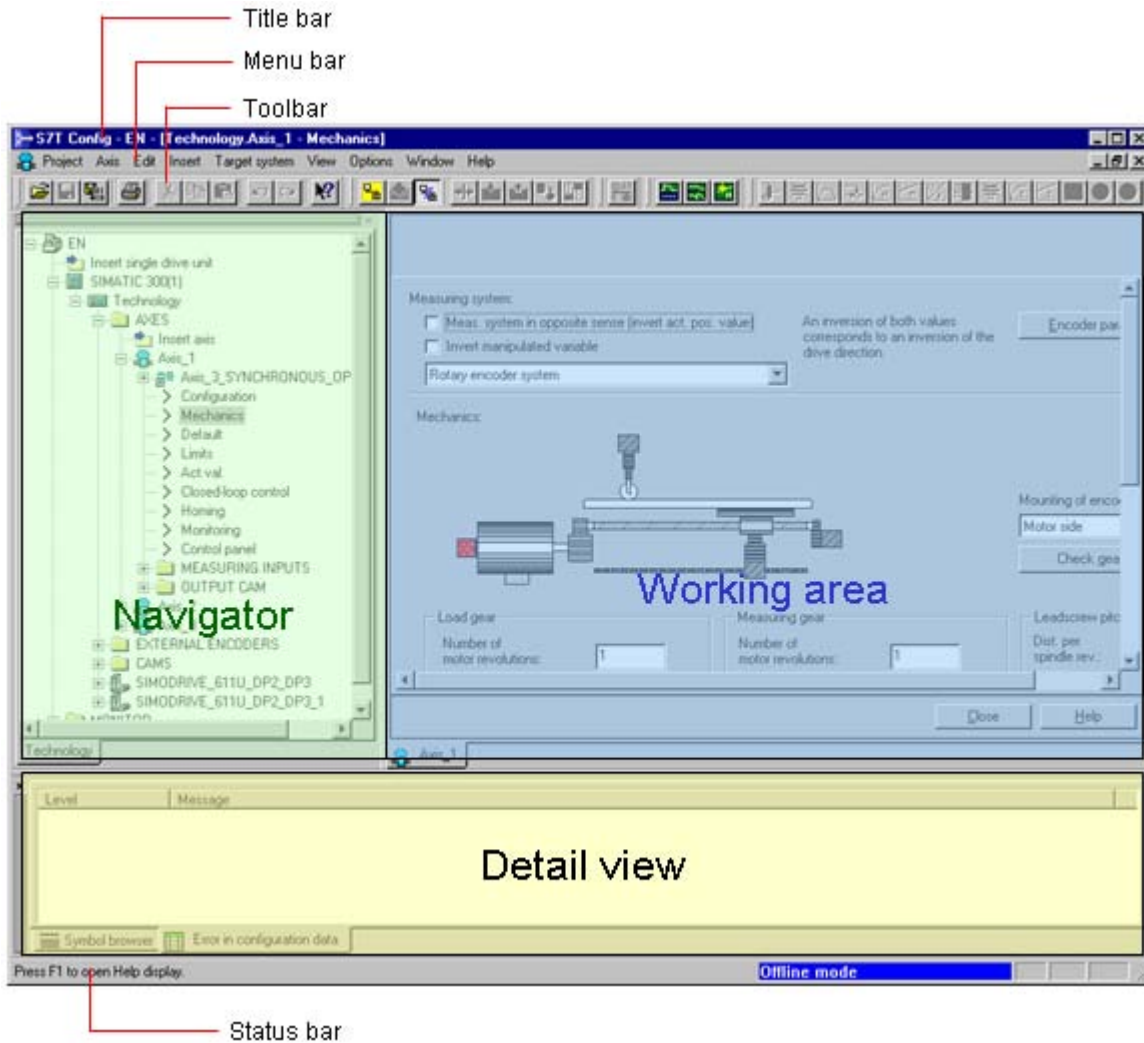
在“工艺对象管理”(Technology Objects Management) 中选择**选项 (Options) > 组态工艺 (Configure technology)** 以打开 S7T Config 并组态工艺对象。如果尚未创建任何工艺对象，S7T Config 将自动启动。

不运行“工艺对象管理”(Technology Objects Management) 也可以启动 S7T Config。从 SIMATIC Manager 的“工艺”(Technology) 文件夹中选择“工艺对象”(Technology objects)，然后选择**选项 (Options) > 组态工艺 (Configure technology)**。如果不想创建任何新的工艺对象，那么使用 SIMATIC Manager 中的菜单命令启动 S7T Config 将非常有用，这样无需创建任何新的工艺 DB。

4.4.2 S7T Config 的用户界面

S7T Config 用于为工艺对象组态运动控制任务。

S7T Config 打开时用户界面显示如下：



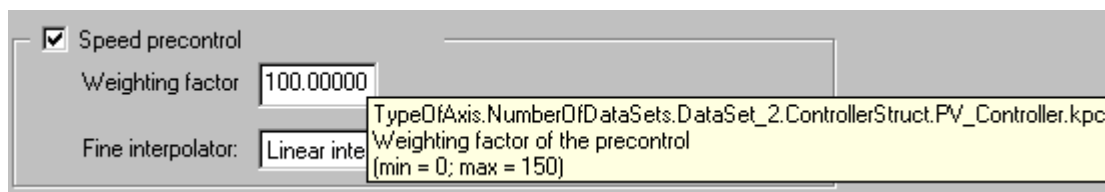
浏览器

浏览器用于可视化工艺的项目结构以及已组态的工艺对象的分配。相同的对象由相同的图标表示。系统通过相应背景色的高亮图标，或者表示 PG/PC 与 PLC 连接的其它图标来指示在线模式。有关详细说明，请参考 S7T Config 在线帮助。通过双击工作区域中的对象，可打开相关组态对话框。

工作区域

例如，在工作区域上显示可用于设置组态数据、默认值和限值的工艺对象组态对话框。这些选项卡指示您已为其打开相关对话框的工艺对象。可以为多个工艺对象打开对话框。

工作区域中的所有可修改参数都被指定了由 S7T Config 中的工具提示指明的上限和下限。



详细视图

详细视图可输出与在浏览器中选择的元素相关的信息。根据选定的对象，该视图将显示不同的选项卡。选项卡的数量取决于工作区域上的活动对话框以及项目状态（离线或在线）。

若要在详细视图中获得有关某对象的信息：

1. 在浏览器中选择相关的对象。详细视图将显示该对象可用的所有选项卡。
2. 在详细视图中单击选项卡，将显示选项卡的内容

标题栏

标题栏包含项目名称以及所选工艺对象的活动组态对话框名称。

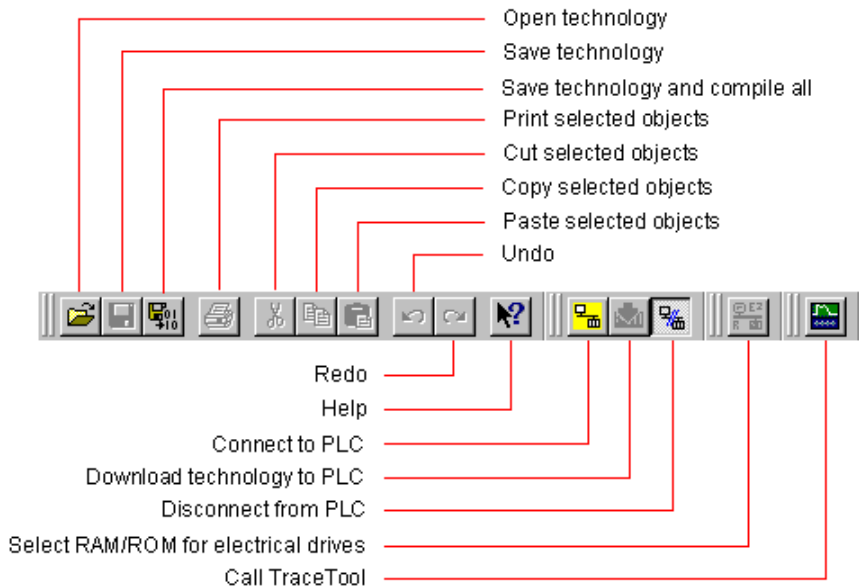
菜单栏

菜单栏包含 S7T Config 中当前支持的菜单。

工具栏

通过单击工具栏图标可以快速访问常用且当前可用的菜单命令。当将指针短暂放在某图标上时，将显示此功能的工具提示。状态栏显示其它信息。

灰显符号表示无法在当前工作状态下执行的功能。



状态栏

状态栏显示上下文相关的信息和项目状态（离线或在线模式）。

离线模式

S7T Config 在离线模式下操作时，无法与 T-CPU 进行通信。在该模式下，可以创建新的工艺对象，使用集成的 **STARTER** 工具组态驱动器以及编译工艺数据。

在线模式

S7T Config 在在线模式下操作时，可以与 T-CPU 进行通信。在该模式下，可以执行诊断操作，将新的项目数据下载至设备并编辑 CPU 中的参数。

更改为在线模式时，系统将检查工艺数据的一致性。该检查将验证在工艺中组态的设备、驱动器和工艺对象的可用性或连接。如果发现不一致，或出现错误（例如，目标系统中的现有组态与 S7T Config 中的离线组态之间有偏差），则在浏览器中以红色背景突出显示相应对象。

 小心

为了确保对驱动器的同步控制，请勿将编程设备 (PG/PC) 连接至 DP(DRIVE) 接口。

4.5 组态电气轴

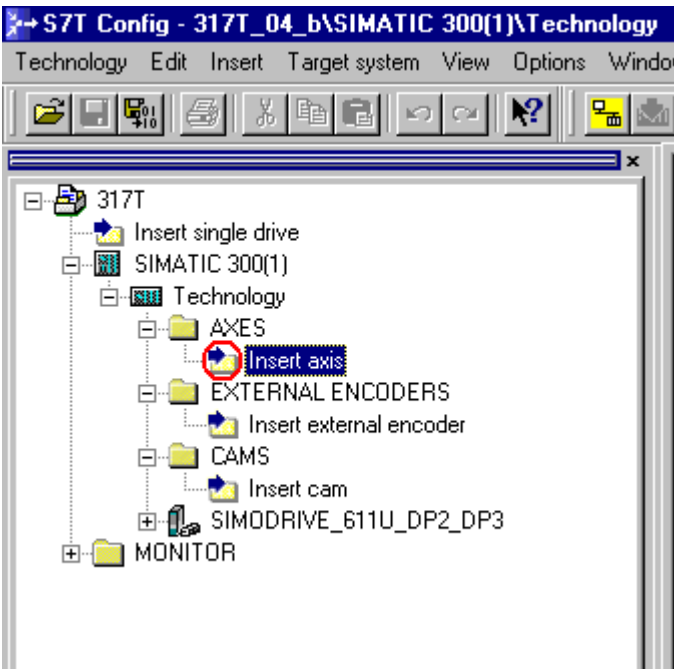
4.5.1 组态电气轴

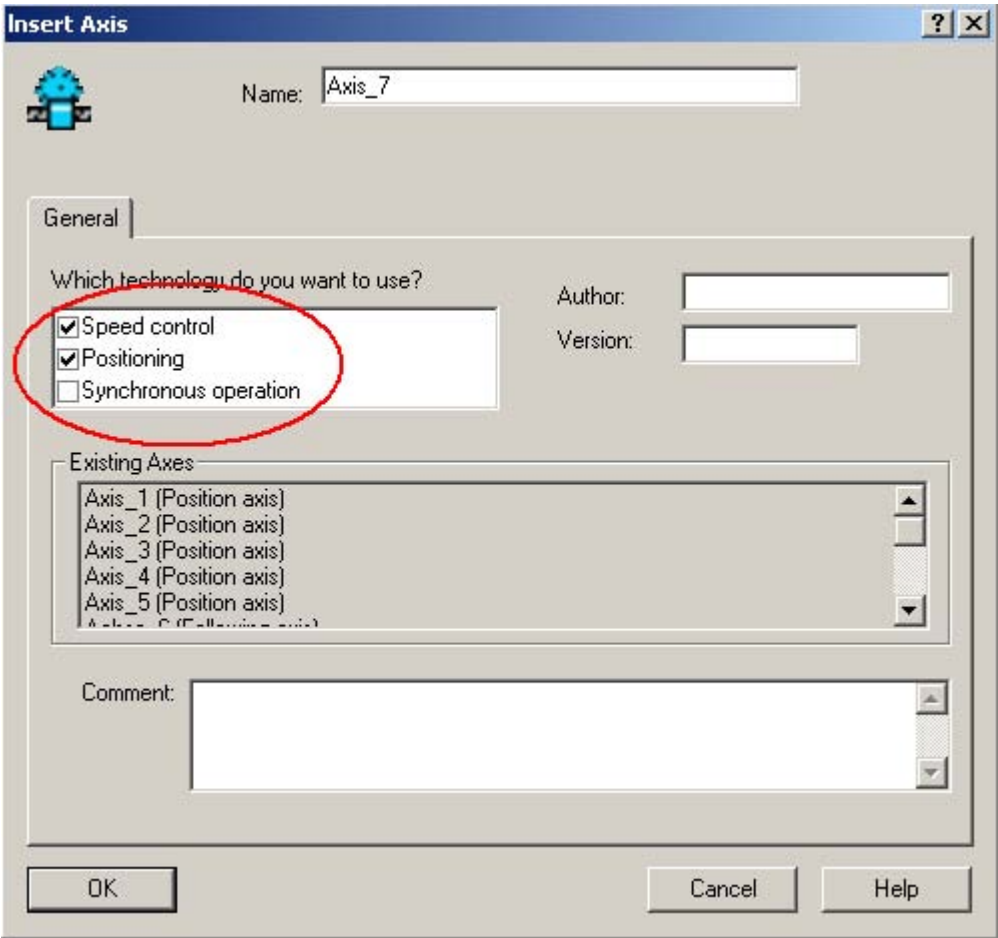
先决条件

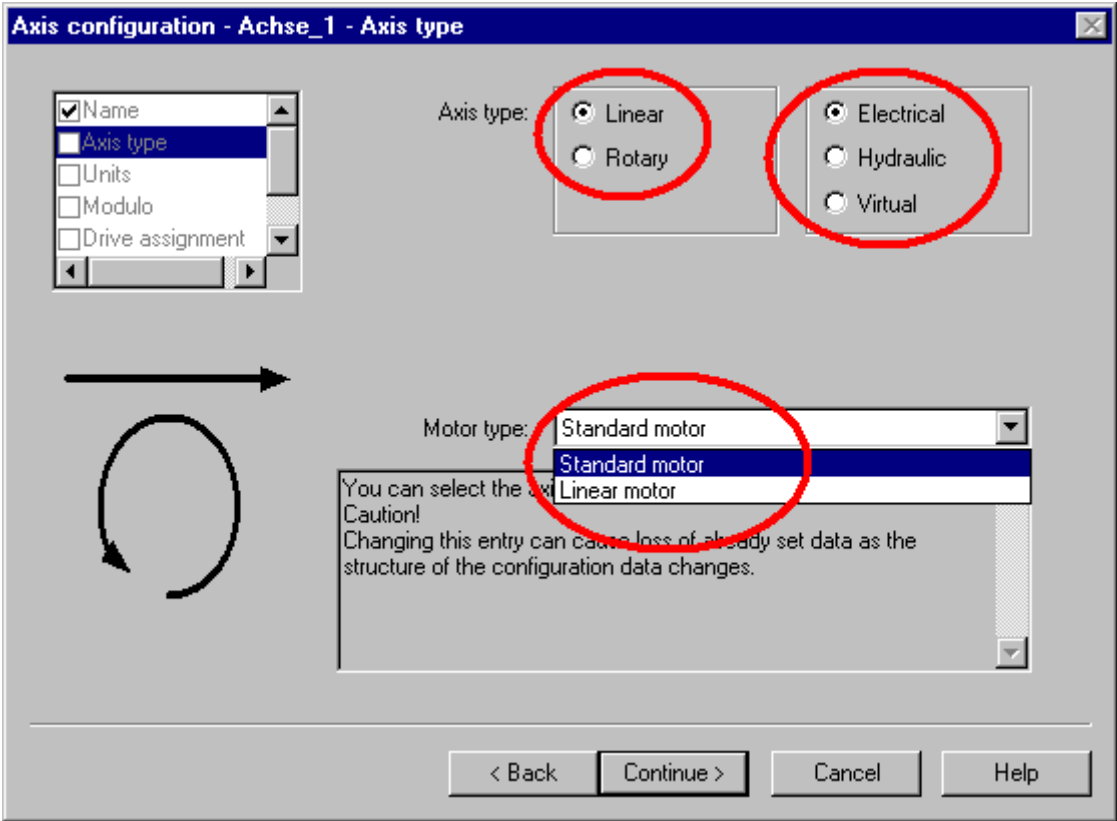
- 已在 HW Config 中组态了 T-CPU，并已编译了组态数据。

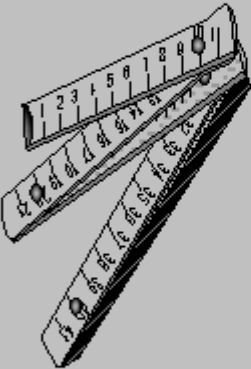
在 S7T Config 中插入新轴

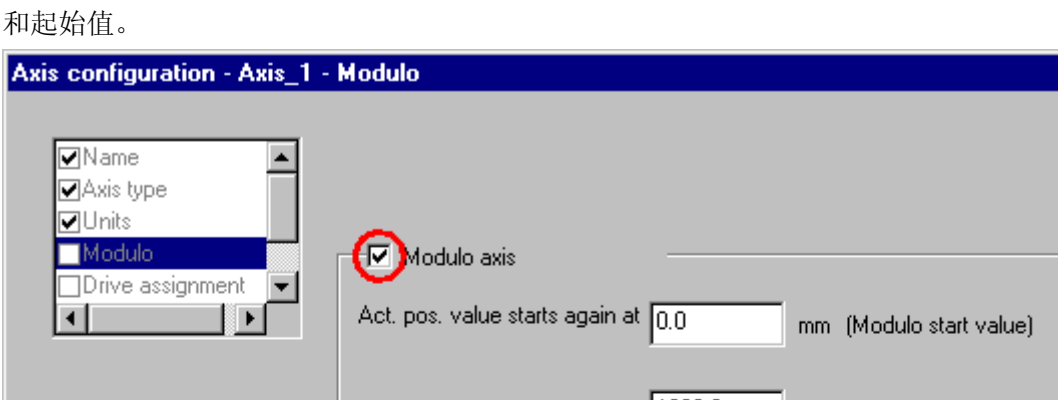
下面以定位轴为例介绍了插入和组态轴的基本操作步骤。

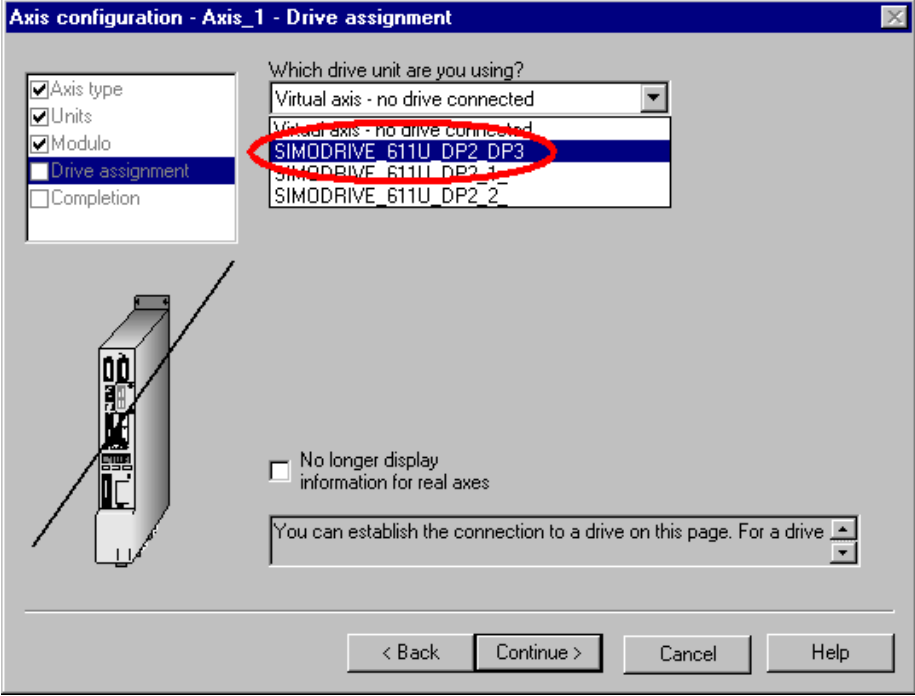
步骤	说明
1.	在 S7T Config 的浏览器中，双击“插入轴”(Insert axis)。 

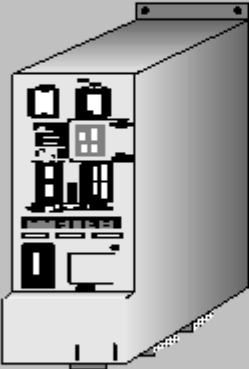
步骤	说明
2.	例如，对于定位轴的组态，从“常规”(General) 选项卡中选择“定位”(Positioning) 工艺。 组态速度控制轴时，请禁用“定位”(Positioning) 工艺。  可以编辑名称、输入作者、定义版本名称以及输入注释。“工艺对象管理”(Technology Objects Management) 将工艺对象的名称作为符号名称应用于 STEP 7 中对象的工艺 DB。
3.	单击“确定”(OK) 按钮。 结果： 轴向导出现。 注意 组态新轴后立即便可定义轴工艺（速度、定位、同步操作）。日后要更改轴工艺，必须删除并重新创建轴。

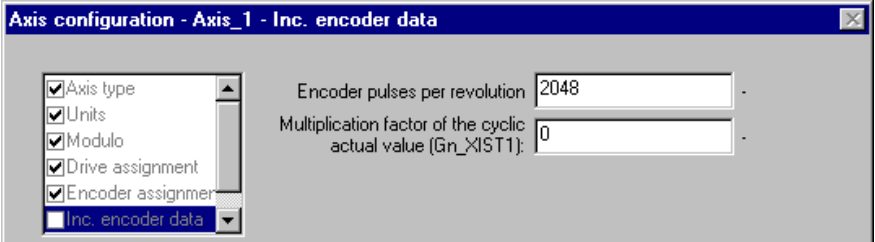
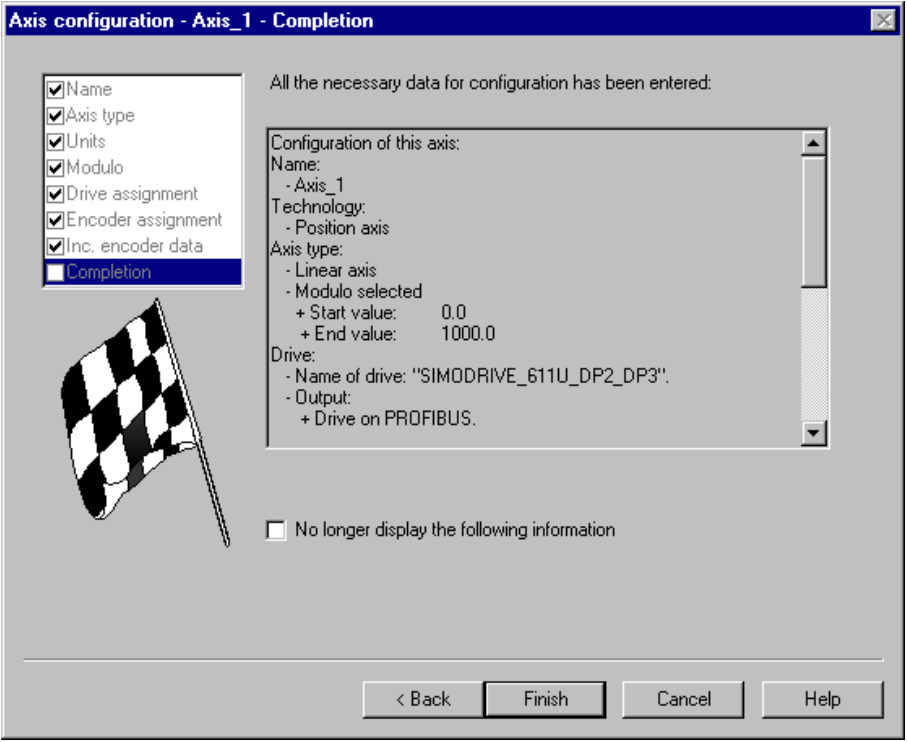
步骤	说明
4.	从下一个对话框中选择轴类型和电机类型。  单击“下一步”(Next)。

步骤	说明																						
5.	从下一个对话框的列表中为轴组态和用户程序选择物理单位。 Axis configuration - Axis_1 - Units ☒Name ☒Axis type ☐Units ☒Modulo ☒Drive assignment  Units: <table><thead><tr><th>Physical quantity</th><th>Unit</th></tr></thead><tbody><tr><td>Position</td><td>mm</td></tr><tr><td>Increments/position</td><td>1000/unit</td></tr><tr><td>Velocity</td><td>mm/s</td></tr><tr><td>Acceleration</td><td>mm/s²</td></tr><tr><td>Jerk</td><td>mm/s³</td></tr><tr><td>Ratio</td><td>%</td></tr><tr><td>Time</td><td>s</td></tr><tr><td>Speed</td><td>1/s</td></tr><tr><td>Leadscrew pitch</td><td>mm/rot</td></tr><tr><td>Angle</td><td>°</td></tr></tbody></table> Caution ! If you change the system of units, the configuration and system variables will be converted (rounding errors are possible) but the data in the programs will not be considered. < Back Continue > Cancel Help 注意： 物理单位设置仅适用于该轴。 根据轴组态，该列表还可能显示与该轴不相关的物理单位。 示例： 为“MC_MoveAbsolute”工艺功能的输入参数 <i>Position</i> 设置的值“1000.0”（位置单位：[mm]）即相当于位置值 1000.0 mm。 注意： 如果更改单位设置，在用户程序中设置的数字值（例如，为运动命令设置的数字值）将不会转换为新单位。 值始终参考已组态的单位！ 单击“下一步”(Next)。	Physical quantity	Unit	Position	mm	Increments/position	1000/unit	Velocity	mm/s	Acceleration	mm/s²	Jerk	mm/s³	Ratio	%	Time	s	Speed	1/s	Leadscrew pitch	mm/rot	Angle	°
Physical quantity	Unit																						
Position	mm																						
Increments/position	1000/unit																						
Velocity	mm/s																						
Acceleration	mm/s²																						
Jerk	mm/s³																						
Ratio	%																						
Time	s																						
Speed	1/s																						
Leadscrew pitch	mm/rot																						
Angle	°																						

步骤	说明
6.	如果想要以模数模式操作该轴，请选中“模数轴”(Modulo axis) 复选框，然后输入所需的模数长度和起始值。  单击“下一步”(Next)。

步骤	说明
7.	在下一个对话框上，选择要连接至该轴的驱动组件。  下拉列表显示在 HW Config 中组态的所有驱动器。已连接的驱动器将不再包括在该下拉列表中。如果在 HW Config 中组态 SINAMICS S120 驱动器，则程序仅显示“同步 Sinamics 设备”(Synchronize Sinamics devices) 按钮。 选择“虚拟轴”(virtual axis) 驱动组件，为同步组中的轴分配逻辑控制功能。如果选择了真实的驱动器单元，该对话框将显示其它选项。 如果选定的驱动器是多轴模块（例如，SIMODRIVE 611 通用型双轴模块、IM 174/ADI4、SINAMICS S120），请从“驱动器”(Drive) 框中选择相关的“轴”(Axis)。 “逻辑 HW 地址”(log. HW addresses)、“输入”(Input) 和“输出”(Output) 显示框显示在 HW Config 的驱动器组态中设置的驱动器的逻辑地址。 如果重新创建该组态，则 HW Config 中设置的默认消息帧类型将输入到下拉列表框中（有关详细信息，请参阅消息帧类型和功能）。如果在 HW Config 中设置的消息帧未知或已被用户编辑，从而导致该消息帧与任何已知的标准消息帧不匹配，则报文向导将帮助您从下拉列表中进行正确的选择。 在“额定速度”(Rated speed) 输入框中输入电机的额定速度。该值与速度评估有关，因此必须与驱动器组态一致。 单击“下一步”(Next)。 结果：“交互作用轴 — 驱动器”(Interaction axis - drive) 对话框打开。单击“确定”(OK) 确认该对话框。S7T Config 在线帮助 (S7T Config Online Help) 随即打开。关闭“在线帮助”，然后继续进行轴组态。

步骤	说明
	注意 - 在 S7T Config 中为驱动组件的工艺对象接口组态的消息帧类型必须与在 HW Config 中组态的组件的消息帧类型相匹配。 - 请确保在驱动器中组态的电机速度和在此对话框中输入的值一致。 如果使用 SINAMICS 驱动器，此对话框还会另外包含“从驱动器传输的数据”(Data transfer from drive) 按钮，利用此按钮可以使用驱动器参数化的值。
8.	从下一个对话框选择驱动器的编码器。 Axis configuration - Axis_1 - Encoder assignment ☒Axis type ☒Units ☒Modulo ☒Drive assignment ☐Encoder assignment  Where is the position encoder connected? SIMODRIVE_611U_DP2_DP3 - Encoder 1 Log. HW addresses: Input: 256 Output: 256 Which message frame type do you want to use for data transfer? Message frame 102 Encoder type: Incremental encoder Encoder mode: Rectangular TTL Measuring system: Rotary encoder system You can set the encoder used for this axis on this page. You can also select special connection types of the encoder. < Back Continue > Cancel Help 从在 HW Config 中组态的消息帧中选择要将其值返回至 CPU 的默认编码器（通常，编码器直接连接至驱动器）。 可以从已组态的编码器列表中为轴位置检测选择其它编码器。 打开下拉列表，选择用于交换编码器数据的消息帧类型。 默认消息帧是在 HW Config 中组态的消息帧。 在“编码器类型”(Encoder type)、“编码器模式”(Encoder mode) 和“测量系统”(Measuring system) 下拉列表中定义在编码器文档或驱动器组态数据中指定的设置。 单击“下一步”(Next)。

步骤	说明
	注意 请确保在驱动器中组态的编码器数据和在此对话框中输入的值一致。如果使用 SINAMICS 驱动器，此对话框还会另外包含“从驱动器传输的数据”(Data transfer from drive) 按钮，利用此按钮可以使用驱动器参数化的值。
9.	在下一个对话框中，定义周期性实际值的编码器精度和倍增因子。 在下一个对话框上，定义周期性实际值的编码器精度和倍增因子。 有关编码器精度的信息，请参考编码器文档。根据在驱动组件的组态中使用的值来校准周期性实际值的倍增因子。  单击“下一步”(Next) 继续进行轴组态。
10.	完成轴组态时，检查输入项的汇总。  单击“完成”(Finish) 退出轴组态对话框。 结果：“交互作用轴 — 驱动器”(Interaction axis - drive) 对话框打开。单击“确定”(OK) 确认该对话框。在轴向导的帮助下，您已成功完成了定位轴组态。

4.5.2 添加数据记录以进行数据记录转换

引言

“MC_ChangeDataset”工艺功能可用于在轴上执行数据记录转换。 可以为速度控制轴、定位轴和同步轴创建数据记录。 虚拟轴始终只有一个数据记录。

例如，数据记录转换功能可用于在电机编码器和机器编码器之间进行切换，或者编辑控制器参数。

说明

编辑数据记录参数时，请注意，数据记录中的某些参数必须相等：

- 更改结构的参数（例如，不同的控制器类型，如 PV 或 PID 控制器）
- 重要的激活参数（例如，跟踪监视开/关、DSC 开/关）

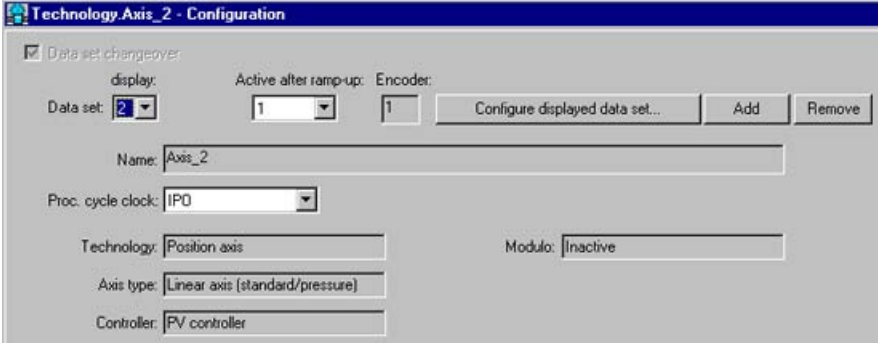
在 S7T Config 中组态数据记录时，一致性检查将报告所有非法的工艺参数设置。

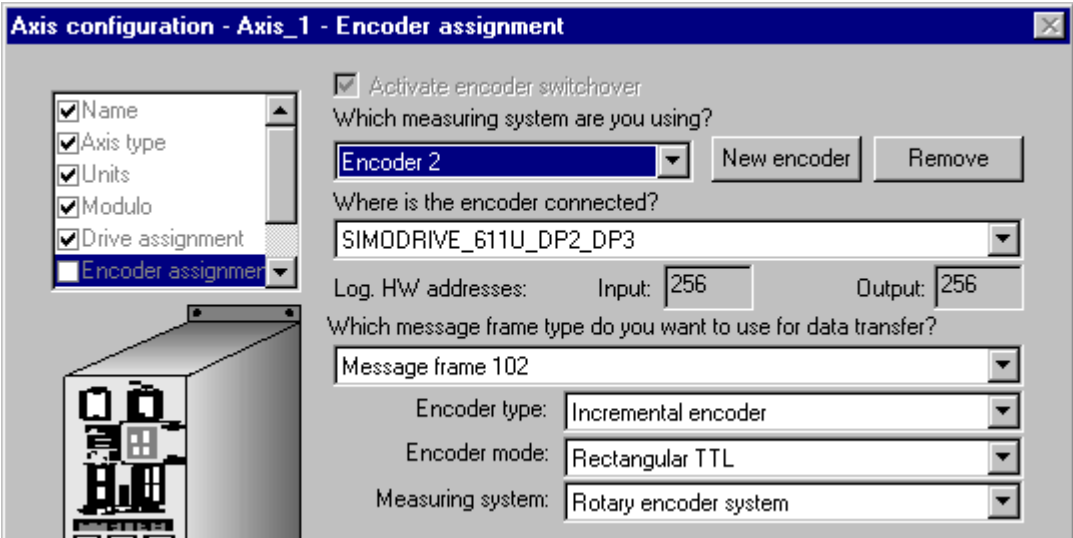
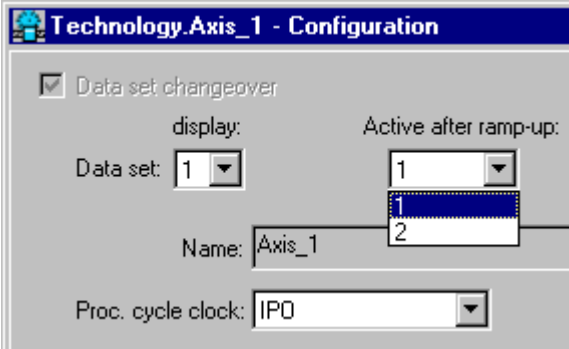
先决条件

- 轴已组态。

添加新数据记录

若要组态新数据记录：

步骤	说明
1.	在 S7T Config 的浏览器中，打开相应轴的数据记录添加组态对话框。
2.	设置“数据记录转换”(data record changeover) 复选框。 如果该轴已包含多个数据记录，则该复选框是隐藏的。
3.	单击“添加”(Add)。 结果： 新数据记录被添加到该轴。 

步骤	说明
4.	单击“组态活动数据集”(Configure active data set)。 结果：轴向导出现。
5.	使用该轴向导组态该数据记录。 注意： 一旦创建了轴，即设置了使用的工艺。不能使用另一个数据记录更改该轴工艺。
6.	也可以使用数据记录转换功能执行编码器转换。可通过设置“轴组态 — 编码器分配”(Axis configuration - Encoder assignment) 对话框上的“启用编码器转换”(enable encoder changeover) 复选框来实现该操作。否则，请忽略该步骤并执行下一步骤。 单击“新建编码器”(new encoder) 创建第二个编码器。 结果：打开一个下拉列表，您可以从中为该数据记录选择一个编码器。
7.	为第二个编码器选择连接和消息帧类型。在最后一步中，输入编码器信息（“编码器类型”[Encoder type]、“编码器模式”[Encoder mode] 和“测量系统”[Measuring system]）。对于编码器组态，选择编码器文档中的设置，或编码器所连接的驱动组件的组态设置。  单击“下一步”(Next)。继续编辑向导中的步骤，完成该组态。
8.	在轴组态对话框中，选择要在 CPU 启动后装载的数据记录（“启动后激活”）。  结果：“编码器”(Encoder) 表将显示轴的已组态编码器的总览。

删除某个轴的数据记录

通过从轴组态对话框中选择某个数据记录，开始删除该数据记录。然后，单击“删除”(Delete)。数据记录被删除。

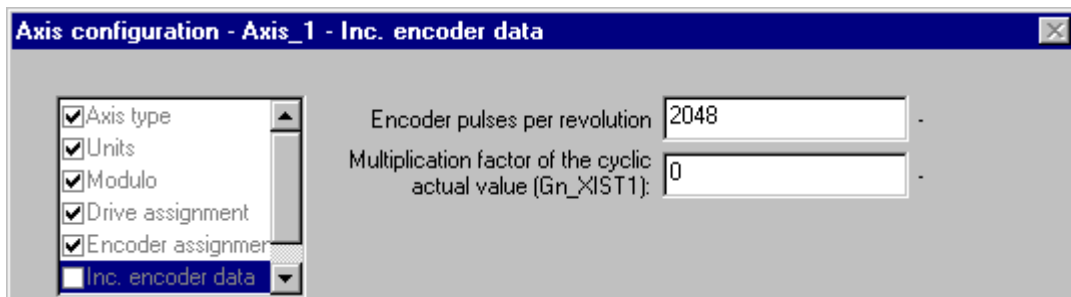
选择数据记录

从“显示数据记录”(Display data record) 下拉列表中选择要编辑的数据记录。在该轴的所有对话框中均可进行该选择。

从“启动后激活的数据记录”(Data record active after startup) 中选择启动后默认使用的数据记录。

4.5.3 高精度

根据编码器注册和类型，可在驱动器中以较高精度来评估编码器脉冲。在驱动器中组态信号计算时必须考虑 S7T Config 中轴组态的“周期性实际值的倍增因子”(Multiplication factor of the cyclic actual value) 参数。



示例

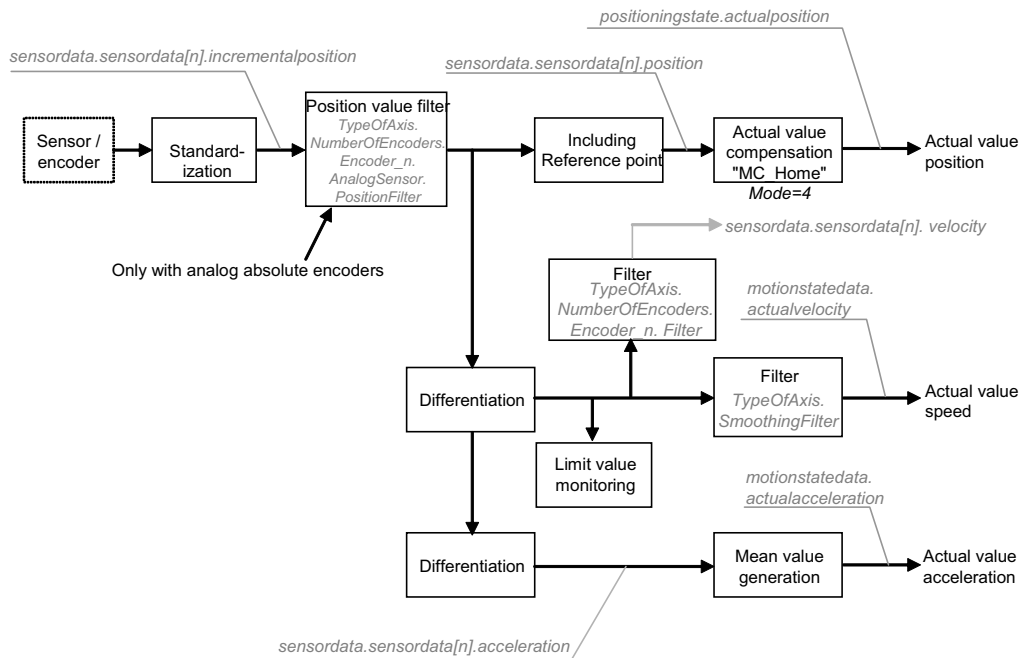
增量编码器连接至 SIMODRIVE 611U，并返回 2048 p/rev（编码器精度）。默认情况下，驱动器将此脉冲数与因子 2048（2 的 11 次幂）相乘。必须在“周期性实际值的倍增因子”(Multiplication factor of the cyclic actual value) 参数中输入该因子。

默认值

默认情况下，“周期性实际值的倍增因子”(Multiplication factor of the cyclic actual value) 设置为 0。系统将自动使用倍增因子 2048 ($2^{\exp 11}$)。

4.5.4 实际值记录

下图显示了如何得到 T-CPU 中轴或外部编码器实际值的过程。



在位置控制周期中进行数据记录。这些数据可用于计算更多变量，如速度和加速度。

系统变量 `sensordata.sensordata[n].incrementalposition` 和 `sensordata.sensordata[n].position` 在位置控制周期时钟中计算。所有其它系统变量均在轴或外部编码器的相应执行周期时钟中进行计算。对于外部编码器，系统变量 `sensordata.sensordata[n]....` 名为 `sensordata....`

可以使用 `TypeOfAxis.NumberOfEncoders.Encoder_n.AnalogSensor.PositionFilter` 组态数据元素激活或取消激活实际位置值。

该过滤器仅可用于模拟绝对编码器：

组态数据元素	含义	
<code>TypeOfAxis.NumberOfEncoders.Encoder_n.</code> <code>AnalogSensor.PositionFilter.enable</code>	YES	实际位置值已激活
	NO	实际位置值已取消激活
<code>TypeOfAxis.NumberOfEncoders.Encoder_n.</code> <code>AnalogSensor.PositionFilter.timeConstant</code>	用于 PT1 滤波的时间常量。	

可通过 *TypeOfAxis.NumberOfEncoders.Encoder_n.Filter* 组态数据元素激活/取消激活“传感器”速度过滤器：

组态数据元素	含义	
<i>TypeOfAxis.NumberOfEncoders.Encoder_n.Filter.enable</i>	<i>YES</i>	实际值过滤器已激活
	<i>NO</i>	实际值过滤器已取消激活
<i>TypeOfAxis.NumberOfEncoders.Encoder_n.Filter.timeConstant</i>	用于 PT1 滤波的时间常量。	

对于外部编码器，组态数据元素名为 *TypeOfAxis.Encoder_1.Filter...*

可通过 *TypeOfAxis.SmoothingFilter* 组态数据元素激活/取消激活实际速度值过滤器：

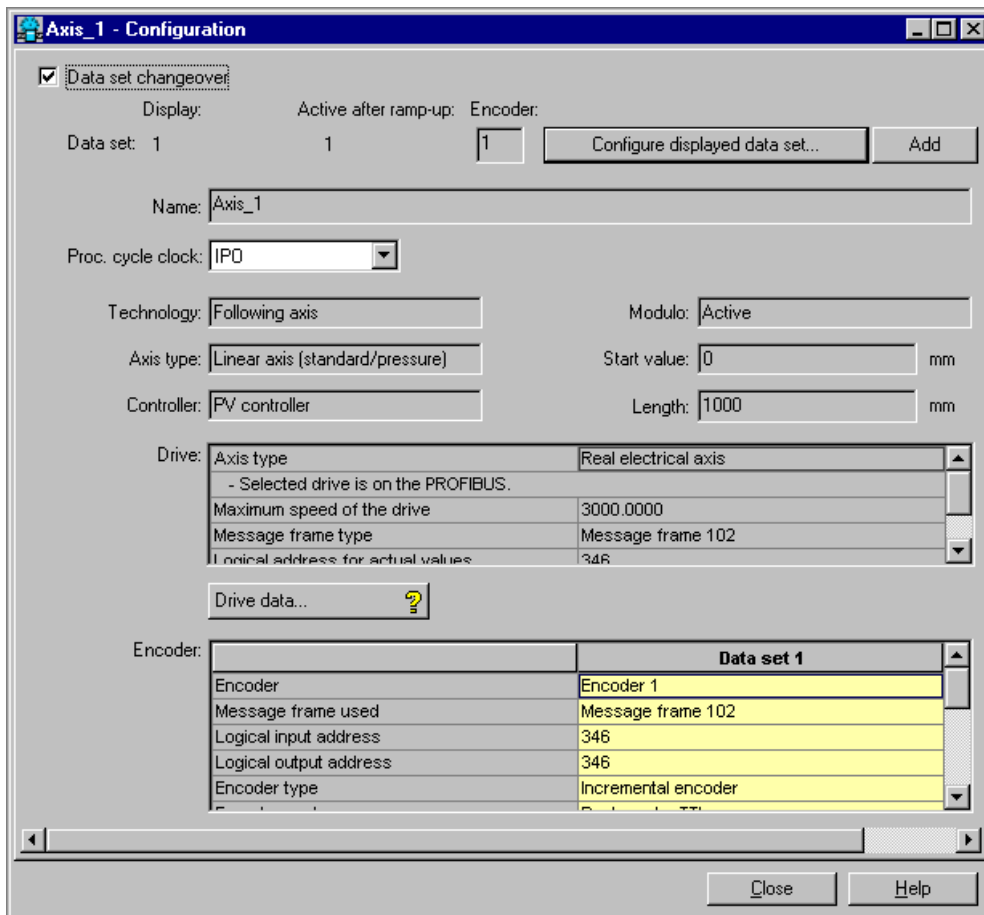
组态数据元素	含义	
<i>TypeOfAxis.SmoothingFilter.enable</i>	<i>YES</i>	实际值过滤器已激活
	<i>NO</i>	实际值过滤器已取消激活
<i>TypeOfAxis.SmoothingFilter.mode</i>	可以通过此组态参数设置过滤器计算方法：	
	<i>DEFAULT_MODE</i>	按比率函数求平均值：轴/外部编码器的执行周期与位置控制周期之比 例如：设置执行周期与位置控制周期之比为 3:1 。在这种情况下，平均值是由三个位置控制器周期得出的。
	<i>AVERAGING</i>	使用时间常量求平均值
	<i>PT</i>	使用时间常量进行 PT1 平滑
<i>TypeOfAxis.SmoothingFilter.timeConstant</i>	用于 PT1 滤波或“ <i>AVERAGING</i> ”的时间常量。	

4.5.5 组态

4.5.5.1 组态 — 轴

轴 (Axis) > 组态 (Configuration) 对话框显示了轴和驱动器设置

可以通过单击“组态活动数据集”(Configure active data set) 在对话框中编辑这些设置。



该对话框还包含用于管理数据记录转换的数据记录的功能。如果工艺对象支持多个数据记录，则您可以：

- 创建新数据记录
- 删除数据记录
- 定义在 CPU 启动期间为工艺对象装载哪个数据记录，以及
- 重新组态选定的数据记录。

4.5.6 机械系统

4.5.6.1 机械 — 电气轴

轴 (Axis) > 机械 (Mechanics) 对话框用于组态轴和编码器的机械参数。

Measuring system:

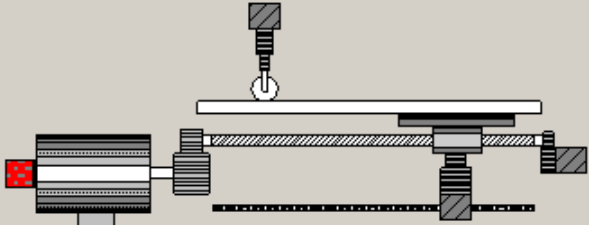
☐ Meas. system in opposite sense (invert act. pos. value) An inversion of both values corresponds to an inversion of the drive direction.

☐ Invert setpoint

Rotary encoder system

Encoder parameter...

Mechanics:



Mounting of encoder: Motor side

Check gear ratio? ?

Load gear

Number of motor revolutions: 1

Number of load revolutions: 1

Measuring gear

Number of motor revolutions: 1

Number of encoder revolutions: 1

Leadscrew pitch

Dist. per spindle rev.: 10.0 mm/rot

☐ Modulo axis

☐ Backlash on reversal compensation:

轴的设置:

- 驱动器方向设置
- 装载齿轮设置
- 线性轴的丝杠螺距
- 模数设置
- 间隙补偿设置

编码器的设置:

- 测量齿轮设置
- 间隙补偿
- 编码器安装类型

模数轴的机械设置的边界条件（长期稳定性）

使用模数轴，可以执行检查以确定是否保证长期稳定性。长期稳定性可确保编码器测量的位置与实际值的内部表示始终同步。这样能使位置精确逼近，甚至在出现多个模数溢出之后。如果不能保证长期稳定性，则在一致性检查期间工程系统中会输出以下错误消息之一：

- 无法表示组态的传动比 (Configured gear ratio cannot be represented)
- 无法表示组态的模数长度 (Configured modulo length cannot be represented)

运行时还会对目标设备执行长期稳定性测试。如果由于组态无法确保此测试，则 S7T Config 中会发出以下报警：“报警 20006 组态错误，原因 3041”(Alarm 20006 Configuration error, reason 3041)。

此错误的原因是组态数据中选择的值不当。这些值必须满足以下条件：

计算模数旋转轴的 f_1

$f_1 = \text{测量齿轮分子} \times 360 \times \text{内部精度} \times \text{负载旋转次数}^1)$

计算模数线性轴的 f_1

$f_0 = \text{TRUNC}(\text{丝杠螺距} \times \text{内部精度})^2)$

舍掉 f_0 的小数位。必须以基本单位 mm 指定丝杠螺距。

$f_1 = \text{测量齿轮分子} \times f_0 \times \text{负载旋转次数}^1)$

计算 f_2

$f_2 = \text{测量齿轮分母} \times \text{编码器精度} \times \text{实际值因子} \times \text{电机旋转次数}^1)$

计算 f_{11} 和 f_{22}

必须从 f_1 和 f_2 确定最大公约数 k ，并将其用于以下公式：

$f_{11} = (f_1 / k) < 2^{31}$

$f_{22} = (f_2 / k) < 2^{31}$

f_{11} 和 f_{22} 的结果必须小于 2^{31} 。否则，请检查相应修改 f_1 和 f_2 公式中的参数后 f_{11} 和 f_{22} 的结果是否会小于或等于允许的最大值。

计算 f_{31}

如果 f_{11} 和 f_{22} 满足上述要求，但下载操作仍中止，且显示 S7T Config 消息“组态错误 20006 原因 3041”(Configuration error 20006 reason 3041)，则可执行以下测试：

$f_3 = \text{TRUNC}(\text{模数长度} \cdot \text{内部精度})$

舍掉 f_3 的小数位。对于线性轴和旋转轴度数，模数长度必须以 mm 指定。

现在可以计算 f_3 和 f_{11} 的最大公约数 k_2 ，并将其插入以下公式中：

$f_{31} = (f_3 \times f_{22} \times f_{11}) / (k_2 \times k_2) < 2^{62}$

此时，必须检查 f_{31} 是否小于 2^{62} 。如果大于 2^{62} ，请检查是否可以减小模数长度。也可以更改 f_1 和 f_2 公式中的参数，只要确保仍然符合对 f_{11} 和 f_{22} 的要求即可。

1)	对于负荷端或外部编码器安装，在公式中，电机旋转次数和负载旋转次数都插入 1。
2)	对于外部编码器安装，在丝杠螺距位置插入组态的每转距离。

参数说明

参数	注释/轴上的组态数据元素
测量齿轮分子	测量齿轮比率的分子 <i>TypeOfAxis.NumberOfEncoders.Encoder_1.AdaptDrive.numFactor</i> <i>TypeOfAxis.NumberOfEncoders.Encoder_1.AdaptExtern.numFactor</i> <i>TypeOfAxis.NumberOfEncoders.Encoder_1.AdaptLoad.numFactor</i>
测量齿轮分母	测量齿轮比率的分母 <i>TypeOfAxis.NumberOfEncoders.Encoder_1.AdaptDrive.denFactor</i> <i>TypeOfAxis.NumberOfEncoders.Encoder_1.AdaptExtern.denFactor</i> <i>TypeOfAxis.NumberOfEncoders.Encoder_1.AdaptLoad.denFactor</i>
电机旋转次数	电机旋转次数的分子 <i>TypeOfAxis.NumberOfDataSets.DataSet_1.Gear.numFactor</i>
负载旋转次数	负载旋转次数的分子 <i>TypeOfAxis.NumberOfDataSets.DataSet_1.Gear.denFactor</i>
内部精度	内部增量/位置单位 已使用轴向导在组态中进行了定义。
编码器精度	每转编码器脉冲数（在编码器上指定） <i>TypeOfAxis.NumberOfEncoders.Encoder_1.IncEncoder.incResolution</i> <i>TypeOfAxis.NumberOfEncoders.Encoder_1.AbsEncoder.absResolution</i>
倍增因子，实际值（实际值因子）	= $X^{1)} \text{IncEncoder.incResolution MultiplierCyclic}$ <i>TypeOfAxis.NumberOfEncoders.Encoder_1.IncEncoder.incResolution MultiplierCyclic</i> <i>TypeOfAxis.NumberOfEncoders.Encoder_1.AbsEncoder.absResolution MultiplierCyclic</i>
模数长度	<i>Modulo.length</i>
轴每转的丝杠螺距	<i>LeadScrew.pitchVal</i>
每转路径	<i>TypeOfAxis.NumberOfEncoders.Encoder_1.pathPerResolution.length</i>

1)	X=0	对于高精度，根据“使用 PROFIdrive 消息帧的编码器接口”部分（PROFIdrive 轴消息帧/高精度中的编码器 (Gn_XIST1)）中介绍的 SIMATIC S7-Technology 中的高精度默认设置表来考虑默认值。（在这种情况下，仅对目标设备执行长期稳定性测试。在工程系统中的一致性测试期间不检查此项。）
	X<>0	将组态值用于高精度。

4.5.7 默认值

4.5.7.1 缺省 —“动态”(Dynamics) 标签

可在轴 (Axis) > 默认 (Default) 对话框的“动态”(Dynamics) 选项卡中设置轴动态的默认值。如果在工艺功能中设置了负的动态值，则会激活默认值。

Dynamic response

Direction: Positive

Velocity: 100.0 mm/s

Velocity profile: Last programmed velocity profile

Jerk: 1000000.0 mm/s³

Acceleration: 1000.0 mm/s²

Jerk: 1000000.0 mm/s³

Jerk: 1000000.0 mm/s³

Deceleration: 1000.0 mm/s²

Jerk: 1000000.0 mm/s³

Stopping time: 0.0 s

支持的默认值:

- 方向 (Direction)
- 速度 (Velocity)
- 加速度 (Acceleration)
- 减速度 (Deceleration)
- 冲击 (Jerk)
- 速度曲线 (Velocity profile)

停止时间

禁用移动轴且在“MC_Power”将 *Stopmode* 设置为 2 时，“停止时间”(Stop time) 中设置的时间便会生效。

速度曲线 (Velocity profile)

速度曲线定义启动期间和制动时的响应，以及对速度变化的响应。

工艺功能不使用默认值。如果要更改速度曲线，请使用 *Jerk* 输入参数。

可以在以下曲线之间进行选择：

- **梯形**

梯形曲线用于正方向和负方向运动的线性加速度（突变化 = 0）。

- **常量**

该曲线显示常量加速度，突变化曲线是可控制的（突变化 $\neq 0$ ）。

4.5.8 限制

4.5.8.1 限制 —“位置和速度”(Position and velocity) 标签

在轴 (Axis) > 限制 (Limits) 选项卡的“位置和速度”(Position and velocity) 选项卡中，您可以

- 组态和启用对硬件限位开关的监视
- 组态和启用对软件限位开关的监视
- 设置速度限制
- 设置急停减速度

Position and velocity

Dynamic response

Fixed endstop

Hardware limit switch

☒ Active

Negative log. address

Bit number:

Positive log. address

Software limit switch

☒ Active

mm

mm

There are two different limits:

- Limits which must never be exceeded because of mechanical reasons.
- Limits which can be modified by programs.

Max. velocity

mm/s

☒ Absolute values

Pos. prog. velocity

mm/s

Emergency stop delay:

mm/s²

急停减速度

禁用移动轴且通过“MC_Power”参数将 *Stopmode* 设置为 0 时，将激活为急停减速度设置的值。

4.5.8.2 硬件限位开关

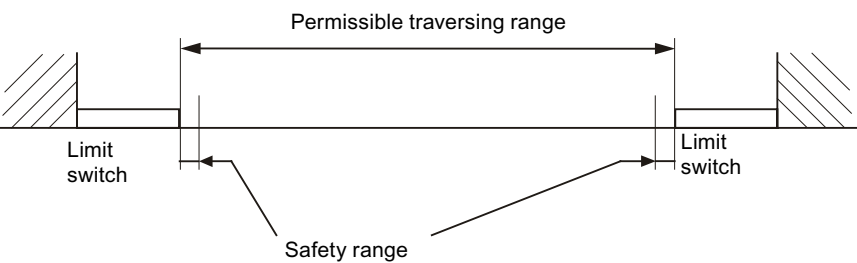
在轴 (Axis) > 限制 (Limits) 对话框的“位置和速度”(Position and Velocity) 选项卡中启用硬件限位开关监视。硬件限位开关监视用于限制轴的运动范围或保护机器。

接线

可将硬件限位开关连接至 T-CPU 的四个集成数字输入端，或者通过在 DP(DRIVE) 上操作的 TM15/TM17 连接至 I/O 模块（如 ET 200 或 SINAMICS S120）。

行程范围

通过硬件限位开关的数字输入监视允许的行程范围。



注意
硬件限位开关必须作为 NC 触点实施。 轴通过允许的行程范围后，硬件限位开关必须保持活动状态，直到达到机械终点位置。

退回

触发硬件限位开关的轴停止，并显示错误消息 **8013** 和 **804B**。该轴从硬件限位开关退回（释放运动），如下所述：

- 手动退回

手动使该轴返回允许的行程范围。只有该轴返回该范围后，**工艺 DB** 的错误才能被确认。

- 通过驱动器退回

该轴的 **工艺 DB** 的错误已被确认，但是错误消息和 **LimitSwitchActive** 位仍保持激活。现在，该轴可以返回到允许的行程范围。反转运动命令将再次触发轴错误。该轴移动到限位开关的范围之外后，便可确认错误消息和 **LimitSwitchActive** 状态。

保存到达硬件限位开关后，即可保存轴的当前位置。只有在通过此位置（加上一个安全范围）之后，该轴才被视为已离开限位开关。

注意
为避免硬件限位开关的极性监视和硬件限位开关的超程监视在有效范围方向上发生冲突，在该轴通过硬件限位开关后 不能 关闭控制器。在这种情况下，该轴将移至有效范围之内（无硬件限位开关监视），然后重新启用。 打开控制器时，该轴必须位于有效行程范围之内。

轴通过硬件限位开关时，内部状态会丢失并且将重新装载组态。只有在有效范围内重新装载时才不会丢失逼近信息。

例外情况： 发生极性反向错误后取消激活限位监视

安全范围

根据轴单位系统的已组态精度计算硬件限位开关的安全范围。

安全范围 = 1000/（增量/位置）

示例： 在“组态单位”(Configure units) 对话框中，将线性轴的位置单位指定为“mm”，将精度（增量/位置）指定为“1000/单位”，这意味着轴位置的计算精确到 0.001 mm。本示例中的安全范围是精度的 1000 倍： 1 mm。

4.5.8.3 软件限位开关

可以在**轴 (Axis) > 限制 (limits)** 对话框的“位置和速度”(Position and Velocity) 选项卡中组态软件限位开关并启用监视功能。如果激活软件限位开关，则通过软件限位开关限制轴的行程距离。

例如，软件限位开关应位于硬件限位开关的行程范围内，以限制轴的工作范围。

运动开始时监视软件限位开关：

- **复选框已激活**

如果位置控制运动命令的目标位置在软件限位开关之外，则运动开始时轴的工艺 DB 中就已显示警告 **0026**。在 **S7T Config** 中，报告中断 **40105**。

轴最多行进到软件限位开关位置，并在工艺 DB 中报告错误 **8014**。

- **复选框未激活**

如果位置控制运动命令的目标位置在软件限位开关之外，则轴来回移动，直到到达软件限位开关的 **p** 位置。在工艺 DB 中报告错误 **8014**（不输出警告 **0026**）。

行进到软件限位开关的特性：

- **对于位置受控的来回移动：**

轴来回移动，直到到达软件限位开关。到达软件限位开关后，保持各自活动的位置控制模式或速度控制模式

- **在所有操作模式中：**

轴来回移动，直到到达软件限位开关。到达软件限位开关后，保持活动的位置控制模式。在速度控制模式中，轴变为位置控制模式。

负位置/正终点位置：

在这些输入域中输入软件限位开关的负位置和正终点位置。

用于退回的公差窗口：

在此输入域中输入适合的值，以防止退回期间重新开始触发软件限位开关错误。

说明

软件限位开关的响应由轴组态根据回原点功能确定。如果轴尚未回原点时，则不监视组态中要求使用已回原点轴来执行绝对运动命令（轴 **[Axis] > 回原点 [homing]** 对话框中的“需要回原点: 是”[Homing required: Yes]）的限位开关。反之，如果在轴未回原点时允许执行绝对运动命令（轴 **[Axis] > 回原点 [homing]** 对话框中的“需要回原点: 否”[Homing required: No]），则无论轴的回原点状态如何，都将监视软件限位开关。

4.5.8.4 限制 —“动态响应”(Dynamic response) 标签

可以在**轴 (Axis) > 限制 (Limits)** 对话框的“动态响应”(Dynamic response) 选项卡中调整 HW 限制（机械）和 SW 限制。

禁用移动轴且在“MC_Power”参数将 *FastStop* 设置为 1 时，激活设定的减速度限制。

默认情况下，可以通过设置“绝对值”(Absolute values) 复选框启用**取决于状态的**加速度模型。

可以通过重设“绝对值”(Absolute values) 复选框启用**取决于方向的**加速度模型。 在这种情况下会显示其他一些附加参数。

The screenshot displays the 'Dynamic response' tab of the 'Limits' dialog box. It is divided into two main sections: 'Hardware limits' and 'Software limits'.

Hardware limits (limits that must not be exceeded for mechanical reasons):

- Acceleration: 10000.0 mm/s²
- Jerk: 200000.0 mm/s³

Software limits (limits that can be modified by programs):

The 'Absolute values' checkbox is checked. The 'Deceleration' section is visible, showing a value of 1000000000000.0 mm/s² for 'In positive dir. of travel'.

The 'Positive jerk' and 'Negative jerk' sections are also visible, both showing a value of 1000000000000.0 mm/s³ for 'In positive dir. of travel'.

Below the input fields, there are three graphs illustrating the acceleration (a) and jerk (j) profiles over time (t). The top graph shows the acceleration profile, which is a smooth curve. The middle graph shows the jerk profile, which is a step function. The bottom graph shows the jerk profile, which is a step function.

At the bottom, there is a checkbox labeled 'Local stop reaction with jerk' which is currently unchecked.

取决于状态的加速度

- **加速度 (Acceleration)**
轴加速度，与运动方向无关
- **减速度 (Deceleration)**
轴减速度，与运动方向无关

取决于方向的加速度

- **加速度 (Acceleration)**
正运动方向的加速度和负运动方向的减速度
 - **减速度 (Deceleration)**
负运动方向的加速度和正运动方向的减速度
- 例如，动态方向矢量的参数设置对于悬挂轴很有用。

带突变化的局部停止响应

设置“带突变化的局部停止响应”(Local stop reactions with jerk) 复选框，以执行由轴上的报警响应触发的停止操作，带有突变化限制和倒圆功能。

对于预编程减速度停止（仅当使用真实轴时输入框才可见）

在此输入框中输入斜坡设置的减速度。

用于平滑由控制器转换产生的受控变量变化的时间常量 （仅当使用真实轴时输入框才可见）

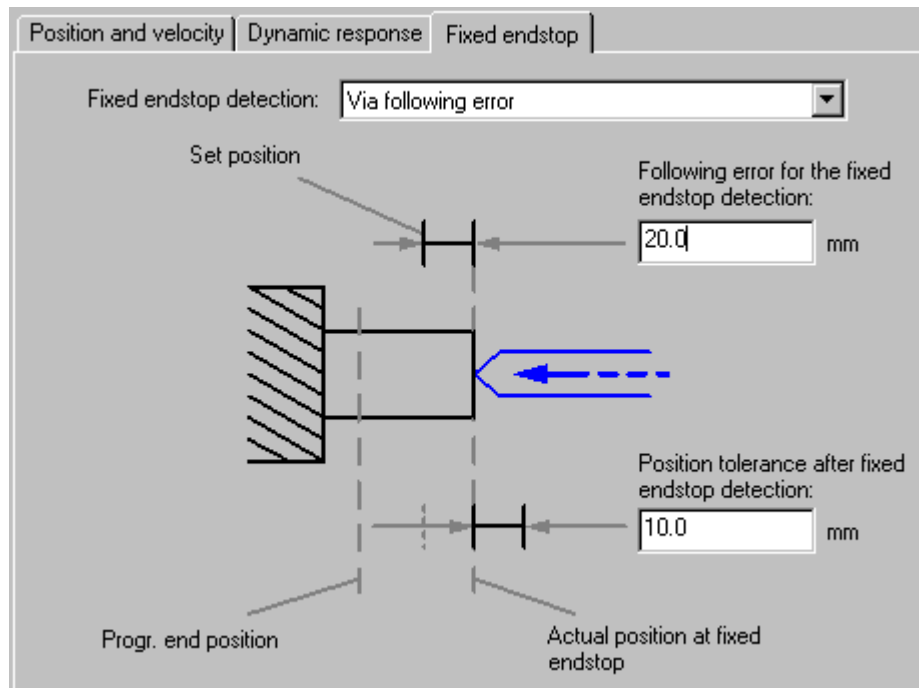
在此处输入用于平滑由控制器转换产生的受控变量变化的时间常量。对于会由于转换引起受控变量偏移的所有状态跳转/转换，此转换平滑滤波器都处于活动状态。

4.5.8.5 限制 —“固定挡块”(Fixed end stop) 标签

轴 (Axis) > 限制 (Limits) 的“固定终点挡板”(Fixed end stop) 选项卡可用于启用固定终点挡板检测以及设置相应的检测模式：

- 使用跟随误差
- 使用力/扭矩

满足选定的条件时，将达到“固定终点挡板”(fixed end stop) 状态。如果启用了“移至固定终点挡块”(Move to fixed end stop)，将禁用跟随误差监视。



使用跟随误差

说明

当轴移至固定终点挡块且将固定终点挡板检测设置为“使用跟随误差”(use following error) 时，在“检测到固定终点挡板之后的位置公差”(Position tolerance after fixed end stop detection) 中输入的值应明显小于“固定终点挡板检测的跟随误差”(Following error for fixed end stop detection) 中的值。

使用扭矩值

固定终点挡板检测功能“使用扭矩值”(use torque value) 要求分配给轴的数字驱动器支持扭矩限制，并且需要设置相应的消息帧，例如消息帧 102 或消息帧 105。

移至固定终点挡块

“MC_MoveToEndPos”功能可以在达到终点挡板后激活“移至终点挡板”(Move to end stop)功能并设置紧固扭矩。该操作也称为“夹紧”。

运动将在轴到达固定终点挡板时停止，而控制仍然保持活动状态。位置控制器输入的设定值保持为常量。钳位位置方向上的新运动命令将被取消；将在退回方向上执行新运动控制命令以减少扭矩。该轴的位置设定值用作新运动控制命令在退回方向上的起始位置。

轴位置设定值由以下其中一个等式产生，具体取决于固定终点挡板检测功能：

- “使用跟随误差”
在固定终点挡板的位置 + 跟随误差
- “使用扭矩值”
在固定终点挡板的位置 + 夹紧公差

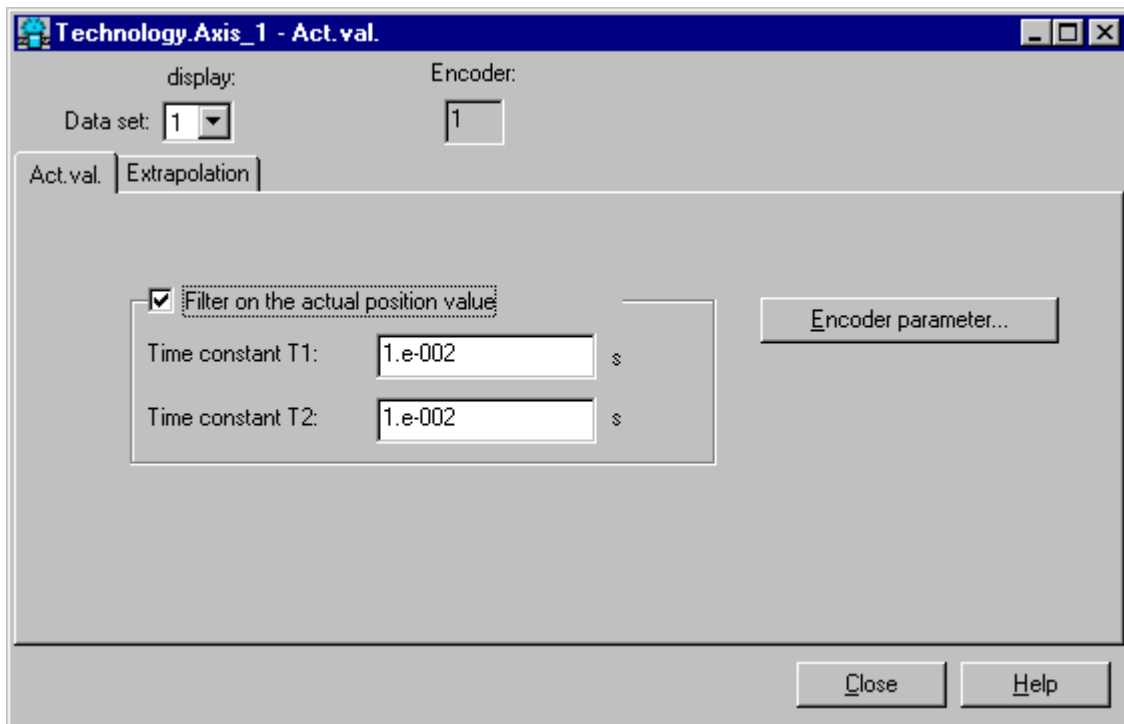
“固定终点挡板检测”功能的条件

- 当轴移至钳位公差窗口外时，将复位“移至固定终点挡块”(Move to fixed end stop)功能。
- 还可以输出新命令，以在钳位处于活动状态时切换扭矩的方向。
- 可在用户程序中，在已定义的时间段上执行非步进扭矩跳转和扭矩保持，就像扭矩曲线的定义一样。
- 可以通过设置反向定位命令禁用移至固定终点挡块（钳位）。
- 不允许反向命令 MC_MoveToEndPos，将忽略该命令。
- 可以通过轴的实际值监视终点挡板的机械制动（钳位公差窗口监视）。
- 通过 MC_MoveToEndPos 工艺功能的 *Torque* 参数，以 [N/m] 为单位设置驱动器的扭矩限制。
- 如果该命令处于活动状态且未检测到固定终点挡板，则系统会像对待活动的扭矩限制那样作出反应。

4.5.9 实际值

4.5.9.1 实际值 (Actual value) —“实际值”(Actual value) 选项卡

在“轴/外部编码器”(Axis/External encoder) > “实际值”(Actual value) 对话框中的“实际值”(Actual value) 选项卡上激活实际值过滤，并设置相应的时间常量。



过滤实际位置值 (Filter on the actual position value)

如果想要激活实际位置值过滤，则激活该复选框。

时间常量 T1 (Time constant T1)

此处设置实际值系统中 PT2 位置过滤器的时间常量 T1。

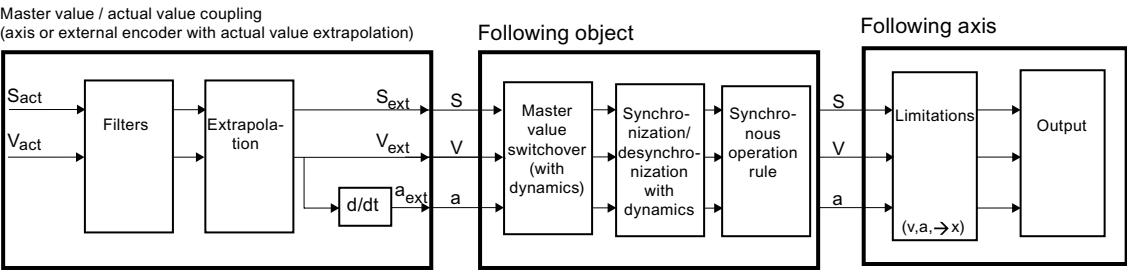
时间常量 T2 (Time constant T2)

此处设置实际值系统中 PT2 位置过滤器的时间常量 T2。

“编码器参数”(Encoder parameter) 按钮

打开一个对话框，其中显示编码器数据。

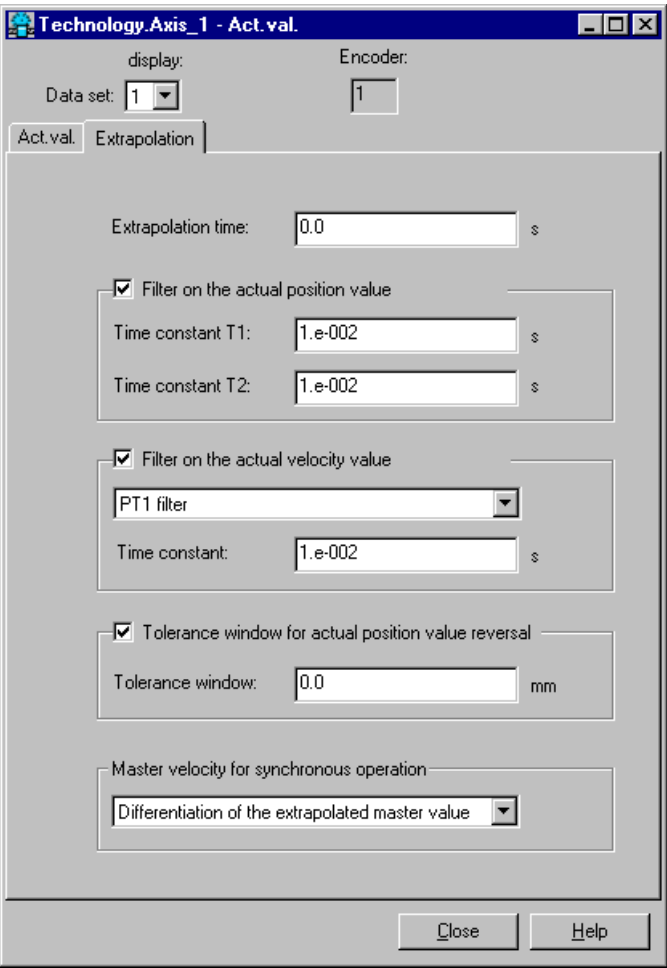
4.5.9.2 实际值 (Actual value) —“推断”(Extrapolation) 选项卡



如果控制内有同步操作互连，则同步操作会考虑主值位置的位置、速度和加速度。

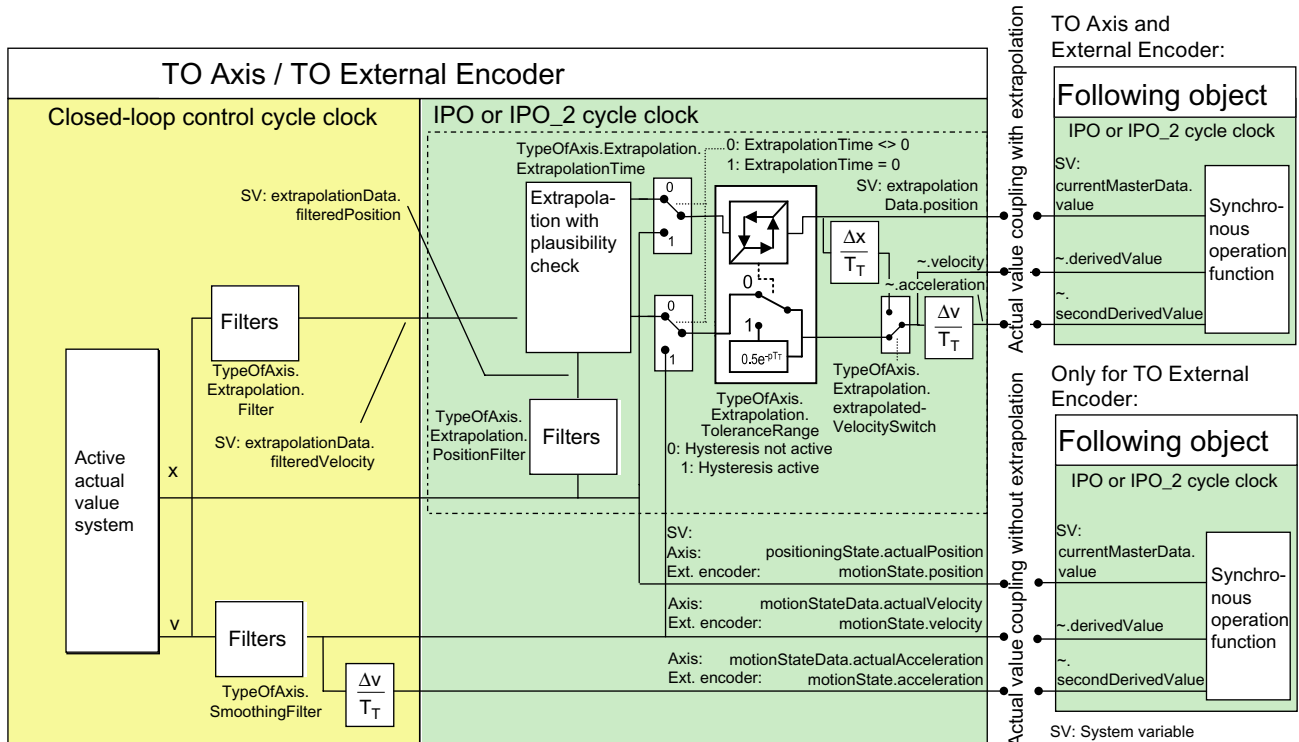
如果实际编码器值用作主值，则推断同步操作的测量实际值以补偿死区时间是很用的。死区时间是在测量实际值时在系统内产生，例如，由于总线通信和系统处理时间产生。

在 S7T Config 中，在引导轴或外部编码器上，在“轴/外部编码器”(Axis/External encoder) >“实际值”(Actual value) 对话框的“推断”(Extrapolation) 选项卡上设置推断。



也可以通过工艺参数从用户程序读取此对话框中的参数，或将此对话框中的参数写入用户程序。

实际值结合推断（轴和外部编码器）



推断时间 (Extrapolation time)

(参数 1110; 组态数据 *TypeOfAxis.Extrapolation.ExtrapolationTime*)

此处设置推断的时间。如果输入的值为 0.0, 则无推断。

过滤实际位置值 (Filter on the actual position value)

(参数 1130 组态数据 *TypeOfAxis.Extrapolation.ExtrapolationPositionFilter.enable*)

如果想要推断实际位置值, 则激活该复选框。

时间常量 T1 (Time constant T1)

(参数 1131 组态数据 *TypeOfAxis.Extrapolation.ExtrapolationPositionFilter.T1*)

此处设置实际值系统中用于推断的 PT2 过滤器的时间常量 T1。

时间常量 T2 (Time constant T2)

(参数 1132 组态数据 *TypeOfAxis.Extrapolation.ExtrapolationPositionFilter.T2*)

此处设置实际值系统中 PT2 位置过滤器的时间常量 T2。

该过滤器作用于推断的实际位置。推断的速度取自应用平滑滤波器 (*TypeOfAxis.smoothingFilter*) 之前轴或外部编码器的实际值。

过滤实际速度值 (Filter on the actual velocity value)

(参数 1112 组态数据 *TypeOfAxis.Extrapolation.Filter.enable*)

如果想要推断实际速度值，则激活该复选框。

在此处从下拉列表中选择推断速度的过滤器。

(参数 1111 组态数据 *TypeOfAxis.Extrapolation.Filter.Mode*)

时间常量 (Time constant)

(参数 1113 组态数据 *TypeOfAxis.Extrapolation.Filter.timeConstant*)

此处输入过滤器的时间常量。

根据过滤的速度值或平均速度值推断位置。通过“时间常量”计算平均值。

实际位置值倒置公差窗口 (Tolerance window for actual position value inversion)

(参数 1114 组态数据 *TypeOfAxis.Extrapolation.ToleranceRange.enable*)

在此处可以激活用于实际位置值倒置的公差窗口

公差窗口 (Tolerance window)

(参数 1115 组态数据 *TypeOfAxis.Extrapolation.ToleranceRange.Value*)

在此处输入公差窗口的大小。

如果主值与同步操作无法跟随的高频噪声信号叠加，则会造成超出动态响应限制或同步期间主值暂时改变方向。

在这种情况下，可定义公差窗口，以防止在跟随轴上超出动态响应限制，或防止同步期间方向改变。

同步操作的主速度 (Master velocity for synchronous operation)

(参数 1116 组态数据 *TypeOfAxis Extrapolation.extrapolatedVelocitySwitch*)

在下拉列表中，选择该速度是否应用于推断，或是否对推断的主位置值求微分。

检查推断值和过滤值 (Checking the extrapolated and filtered values)

可在以下系统变量中检查推断值和过滤值：

- *extrapolationdata.position*
- *extrapolationdata.velocity*
- *extrapolationdata.filteredposition*
- *extrapolationdata.filteredvelocity*
- *extrapolationdata.acceleration*

支持具有 NIST 评估的编码器 (Support of encoders with NIST evaluation)

使用具有 NIST 评估的编码器，速度由编码器确定，所得的速度可被编码器接受。在这种情况下，不需要由集成工艺计算速度和速率。有两种传送方法可用：

- 在 PROFIdrive 消息帧中传送

通过组态数据元素 *TypeOfAxis.NumberOfEncoders.Encoder_n.EncoderValueType = POSITION_AND_PROFIDRIVE_NIST_* 设置

- 在 I/O 区域中传送

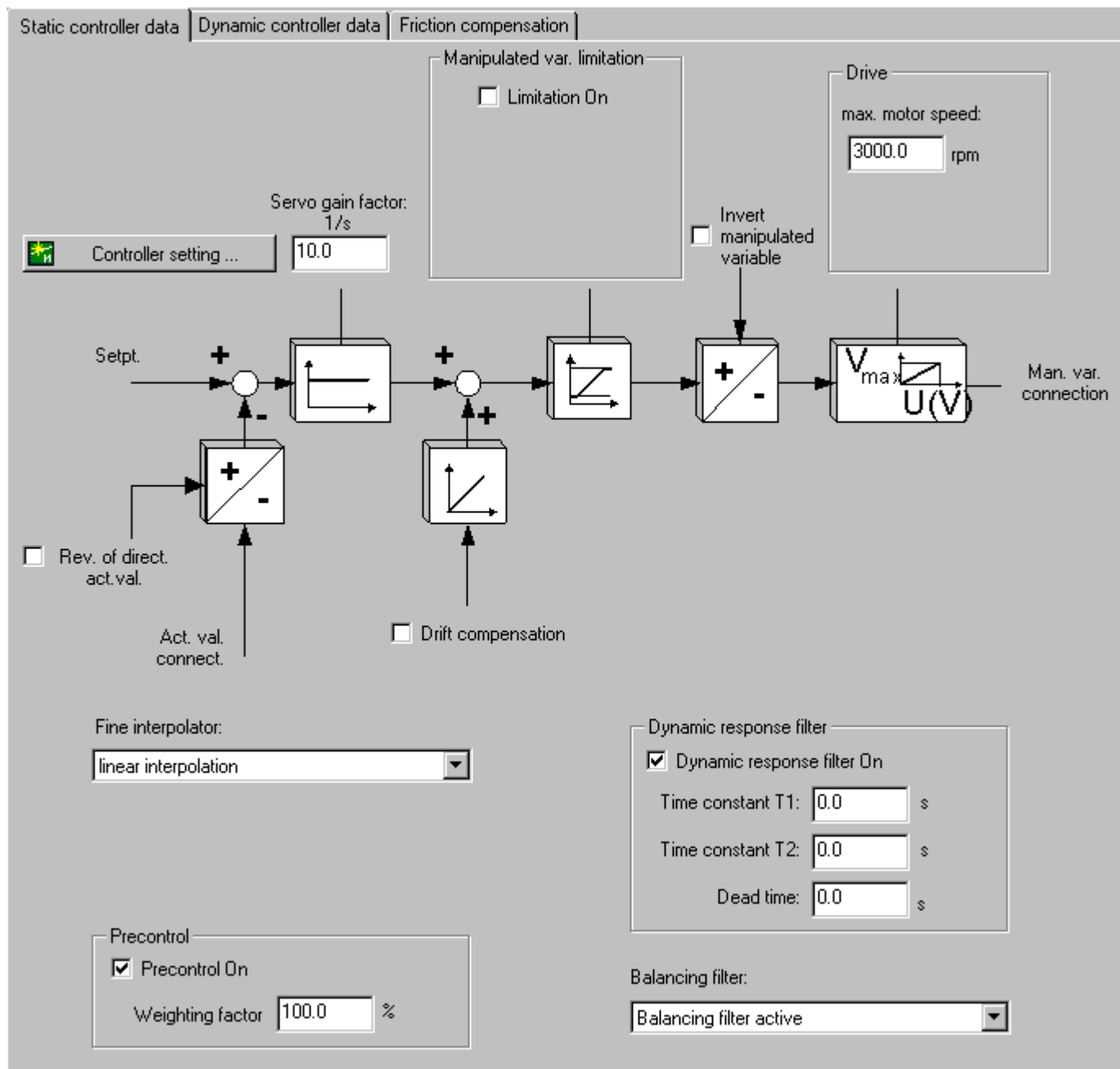
通过以下组态数据元素设置：

TypeOfAxis.NumberOfEncoders.Encoder_n.EncoderValueType = POSITION_AND_DIRECT_NIST。在这种情况下，4000H 对应于 100%。地址在组态数据元素 *TypeOfAxis.NumberOfEncoders.Encoder_n.nistConfig.logAdress* 中设置，参考值在组态数据元素 *TypeOfAxis.NumberOfEncoders.Encoder_n.nistConfig.referenceValue* 中设置。

4.5.10 控制

4.5.10.1 控制 —“静态控制器数据”(Static controller data) 标签

“轴”(Axis) > “控制”(Control) 对话框的“静态控制器数据”(Static controller data) 选项卡可用于组态轴的位置控制。



受控变量限制

受控变量限制表示控制范围的绝对上限和下限。该限制在反转之前应用。

说明

当动态伺服控制（驱动器中的位置控制器）功能处于活动状态时，间隙锁定（驱动器受控变量的限制）无效。因此，DSC 处于活动状态时，必须在驱动器中生成后退定程挡块。

驱动器

使用此输入框输入驱动器的最大速度。

“控制器设置”(Controller setting) 按钮

此按钮可用于实施简单的控制器优化，无需预控制和带有 SINAMICS 驱动器的平衡滤波器。

如果要想实现实施后的位置控制系统，请按照『优化位置控制器 — 总览』一章中的说明操作。

控制器参数

有关各控制器参数及其最佳设置的详细信息，请参考『优化位置控制器 — 总览』一章。

高精度插补器

高精度插补器功能用于在插补器和控制器具有不同的占空比时生成中间设定值。高精度插补器上的可选插补模式设置：

- 无插补
- 线性插补（定位轴的位置恒定）
- 以恒定加速度（定位轴上加速度恒定）插补
- 以恒定速度（定位轴上速度恒定）插补

当设置为定位轴时，将插补位置设定值。

当设置为速度控制轴时，将插补速度设定值。

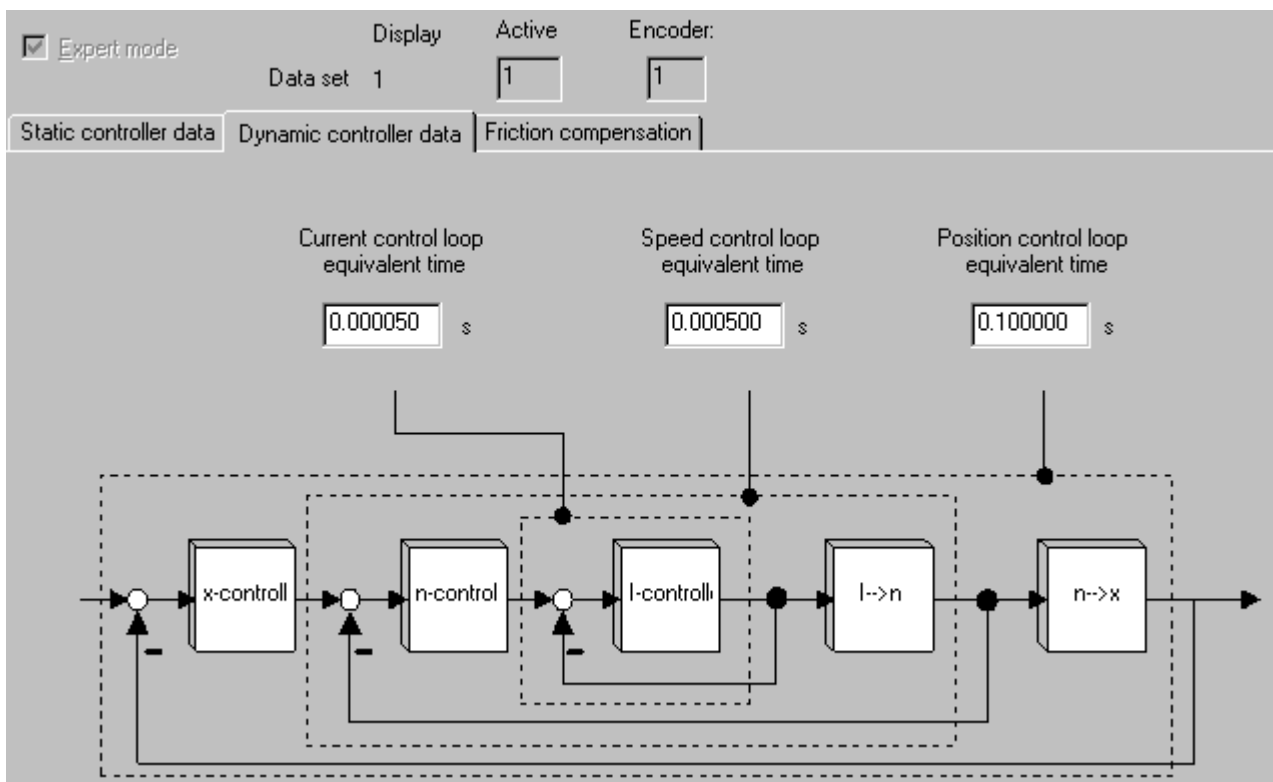
动态过滤器、预控制、平衡滤波器

有关这些设置的详细信息，请参考『优化位置控制器 — 总览』一章。

4.5.10.2 控制 —“动态控制器数据”(Dynamic controller data) 标签

例如，轴 (Axis) > 控制 (Control) 的“动态控制器数据”(Dynamic controller data) 选项卡可用于调整轴的动态响应，以匹配它们在同步操作中的响应。在“静态控制器数据”(Static controller data) 选项卡中设置了专家模式后，“动态控制器数据”(Dynamic controller data) 选项卡便可用。

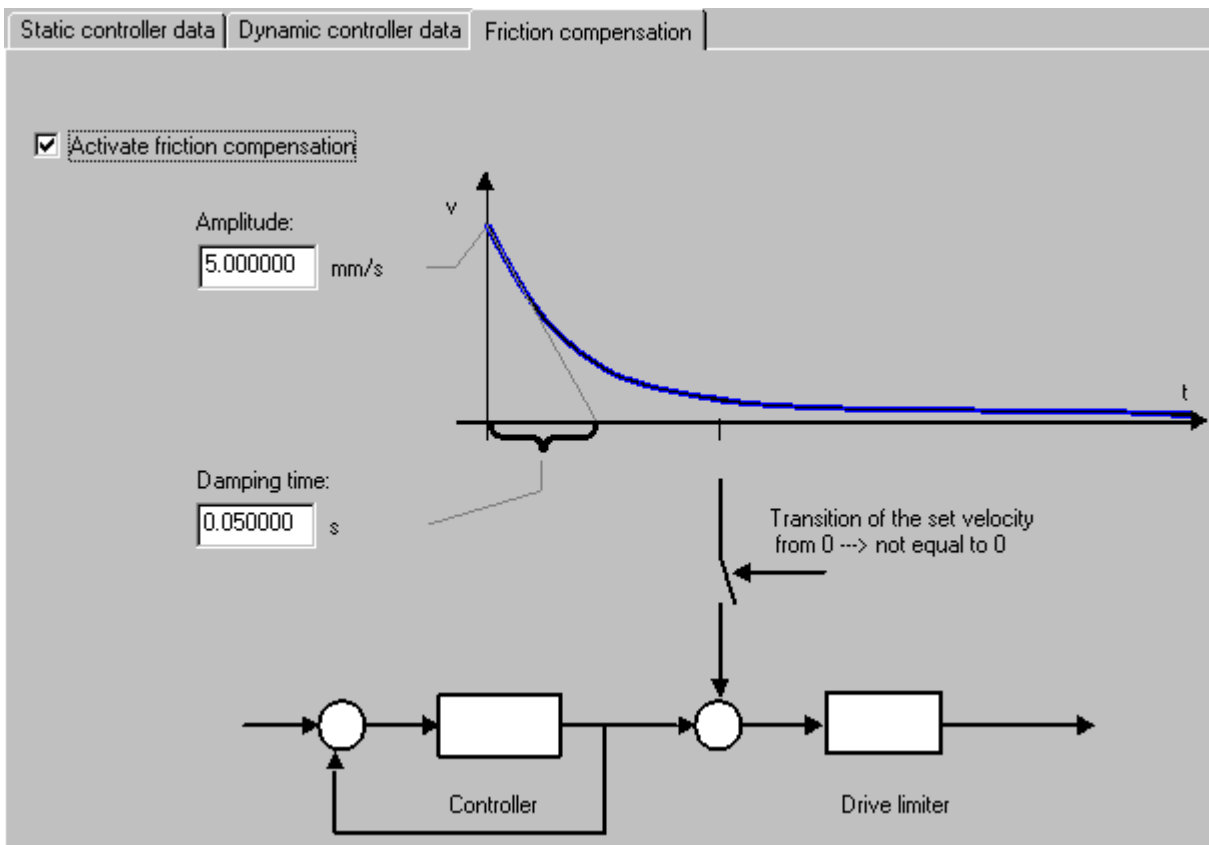
控制回路的设定值分支包含一个可组态的动态过滤器，可用于调整轴的动态响应。



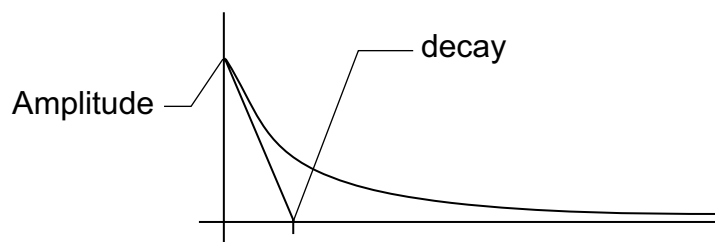
- **等效时间当前控制回路**
在本版本中不使用等效时间当前控制回路。
- **等时速度控制回路**
“等时速度控制回路”参数可用于设置平衡滤波器的时间常量 vTc （速度时间常量）。
- **等时位置控制回路**
需要位置控制回路的等时，以从速度控制模式切换到位置控制模式，并启动急停斜坡。

4.5.10.3 控制 —“摩擦补偿”(Friction compensation) 标签

轴 (Axis) > 控制 (Control) 选项卡的“摩擦补偿”(Friction compensation) 选项卡用于启用摩擦补偿。在“静态控制器数据”(Static controller data) 选项卡中设置了专家模式后，“摩擦补偿”(Friction compensation) 选项卡便可用。



系统提供了一种简单的方法来补偿静态摩擦力。从停止开始启动时，DT1 单元在受控变量上添加一个静态摩擦补偿信号。



摩擦补偿是相对于速度设定值而添加的。只有执行运动命令时摩擦补偿才会处于活动状态。

可以单独为静态摩擦补偿设置停止状态标识、振幅和衰减响应。可以在组态中设置振幅和衰减响应。

4.5.11 回原点

4.5.11.1 引言

每次接通电源时，装有增量电机测量系统的位置控制轴必须参考机器的机械系统的位置。根据轴机制已定义位置的激活特定位置值，在回原点期间同步轴。

轴可以在以下模式下回原点：

- 主动模式（参考点逼近）
- 被动模式（快速参考）
- 直接模式（使用位置设定值）

详细设置取决于参考点的测量系统以及轴的回原点运动。

4.5.11.2 回原点 —“主动回原点”(Active homing) 标签

主动回原点

可以在 S7T Config 中的轴 (Axis) > 回原点 (Homing) 对话框的“主动回原点”(Active homing) 选项卡中组态主动回原点。

“主动回原点”(Active homing) 有三种可用的回原点模式：

- 参考凸轮和编码器零标记
- 仅编码器零标记
- 仅外部零标记

启用回原点功能：

MC_Home	Mode = 0	根据轴组态确定参考点
	Mode = 1	根据轴组态确定参考点。给参数点分配的值是 <i>Position</i> 输入参数。
	Position = x	

成功回原点后，轴工艺 DB 将返回状态 *Statusword.HomingDone* = TRUE。

使用“参考凸轮和编码器零标记”(reference cam and encoder zero mark) 的主动回原点模式

使用“MC_Home”工艺功能开始回原点后，轴将根据轴组态移动至参考凸轮，然后逼近参考编码器零标记。该图的下半部分显示了最终的运动顺序。

各个参数的含义：

参数	值	说明
需要回原点	是	轴必须回原点才能执行绝对运动命令。
	否	轴不必回原点就可以执行绝对运动命令。
回原点模式	-	在这种情况下，“回原点输出凸轮和编码器零标记”
编码器零标记	参考凸轮之前	轴以参考点逼近方向返回编码器零标记，该标记位于参考凸轮前面。
	参考凸轮之后	轴以参考点逼近方向返回编码器零标记，该标记位于参考凸轮后面。
开始参考点逼近	正方向	正方向上的参考点逼近。
	负方向	负方向上的参考点逼近。

参数	值	说明
参考凸轮的逻辑地址	[字节地址]	参考凸轮的逻辑字节地址 可以将参考凸轮连接至 T-CPU 的集成输入或连接至 DP(DRIVE) 上的 IO。
位号	[位号]	用于参考凸轮的信号的位地址
逼近速度	-	轴逼近参考凸轮的速度
进入速度	-	检测到编码器零标记后，轴逼近（移位后的）参考点的速度
关闭速度	-	检测到参考凸轮后，轴逼近编码器零标记的速度。
参考点坐标 (参考凸轮和编码器零标记、仅外部零标记、仅编码器零标记)	-	在此处输入原点位置的实际位置值（相对于轴的坐标系）
回原点位置偏移	-	“原点位置偏移”(Home position offset) 将参考点移位所组态的距离。轴按照在“进入速度”(Entry velocity) 中定义的速度移动该组态距离，起点为具有编码器零标记的同步位置。 模数轴始终取最短距离。
到回原点输出凸轮的最大距离	取消激活	没有监视到参考凸轮检测的距离
	激活	监视参考点逼近的起点和参考凸轮检测之间的距离。如果该距离的差值超出了组态的值，则相应的工艺 DB 将返回错误 801D。参考点逼近取消。
到编码器零标记的最大距离	取消激活	没有监视到编码器零标记所需要移动的距离
	激活	监视轴在参考凸轮和编码器零标记之间移动的距离。如果此距离的差值超过组态的距离，则相应的轴工艺 DB 会指示错误 801D。参考点逼近取消。

参考点逼近的顺序

- 阶段 1: 参考凸轮逼近

轴以组态的逼近速度，沿在“参考点逼近的起点”(Start of reference point approach) 中设置的方向开始其参考点逼近。到达参考凸轮（阶段 1）后，参考凸轮逼近结束。

也可以通过设置“到回原点输出凸轮的最大距离”(Max. distance to homing output cam) 复选框，来监视轴从参考点逼近的起点到检测出参考凸轮时所移动的距离。如果在已组态距离的起点和终点之间没有检测到参考凸轮，则参考点逼近将因出错而取消。

- 阶段 2: 与编码器零标记同步

到达参考凸轮后，轴加速/减速到关闭速度，并逼近编码器零标记。编码器零标记从“编码器零标记”(Encoder zero mark)（参考凸轮之后或之前）和“参考点逼近的起点”(Start of reference point approach)（正方向或负方向）的组合设置得出。

检测到参考凸轮后，PLC 会根据组态将轴与检测到的第一个编码器零标记同步。将轴位置设置为默认值减去在“参考点坐标”(Reference point coordinate)(*Mode = 0*) 中或在输入参数 *Position (Mode = 1)* 中定义的参考点位移。

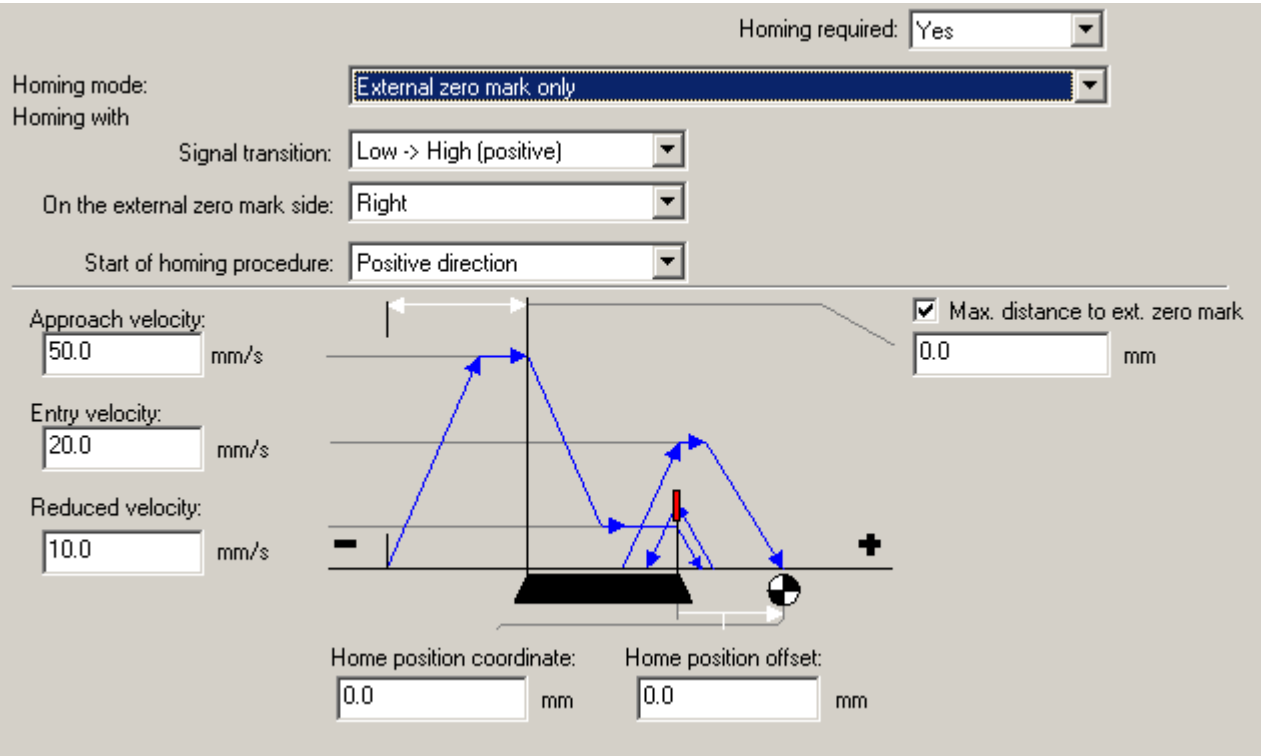
也可以通过设置“到编码器零标记的最大距离”(Max. distance to encoder zero mark) 复选框，来监视轴在参考凸轮和编码器零标记之间移动的距离。如果在检测到参考凸轮后，在指定的距离中未发现编码器零标记，则参考点逼近将因出错而取消。

- 阶段 3: 参考点逼近

检测到编码器零标记后，轴将加速/减速到逼近速度，以逼近参考点。

如果已组态了参考点位移，则轴以相应的距离（起点为同步位置）逼近该参考点。运动方向由参考点位移的符号以及检测到编码器零标记后的减速斜坡长度确定（前提是参考点位于减速斜坡内）。

使用“仅外部零标记”(External zero mark only) 的主动回原点模式



如果通过调用“MC_Home”工艺功能启动回原点过程，则轴将根据组态逼近参考外部零标记。该图的下半部分显示了最终的运动顺序。

各个参数的含义：

参数	值	说明
需要回原点	是	轴必须回原点才能执行绝对运动命令。
	否	轴不必回原点就可以执行绝对运动命令。
回原点模式	-	在此例中：“仅外部零标记”(External zero mark only)
信号跳转	低 -> 高（正）	运动参考外部零标记的上升沿（根据驱动组件中的沿评估进行设置）。
	高 -> 低（负）	运动参考外部零标记的下降沿（根据驱动组件中的沿评估进行设置）。
在外部零标记一侧	左	沿选定的参考点逼近的方向，从外部零标记的左侧进行观测，测量信号跳转。
	右	沿选定的参考点逼近的方向，从外部零标记的右侧观测，测量信号跳转。

参数	值	说明
开始参考点逼近	正方向	正方向上的参考点逼近。
	负方向	负方向上的参考点逼近。
逼近速度	-	轴逼近外部零标记的速度
进入速度	-	检测到外部零标记后，轴逼近（移位后的）回原点位置的速度。
关闭速度	-	检测到外部零标记后，轴逼近参考点坐标的速度
回原点位置偏移	-	“原点位置偏移”(Home position offset) 将参考点移位所组态的距离。轴在外部零标记沿处同步后，以“回原点速度”(Homing velocity) 移动所组态的距离。模数轴始终取最短距离。
到外部零标记的最大距离	取消激活	没有监视到外部零标记检测的距离。
	激活	监视从参考点逼近的起点到检测到外部零标记所移动的距离 如果该距离的差值超出了组态的值，则相应的工艺 DB 将返回错误 801D。参考点逼近取消。

说明

对于使用“外部零标记”(external zero mark) 的回原点，将外部零标记作为数字测量输入连接至驱动组件。

为了以“外部零标记”(external zero mark) 回原点模式执行参考点逼近，请选择与驱动器的组态或功能相对应的“信号跳转”(Signal transition) 和“在外部零标记一侧”(on the side of the external zero mark) 的值。“信号跳转”(Signal transition) 和“在外部零标记一侧”(on external zero mark side) 参数仅反映其功能，以根据组态控制轴运动，而不是用来组态驱动器中的测量功能。

有关外部零标记检测的组态的信息，请参考相关的产品信息或驱动器手册。

参考点逼近的顺序

- 阶段 1：与外部零标记同步

轴以组态的逼近速度，沿在“参考点逼近的起点”(Start of reference point approach) 中设置的方向开始其参考点逼近。在外部零标记处检测到组态的信号跳转（已在驱动组件中组态）后，同步在外部零标记处（阶段 1）结束。将轴位置设置为默认值减去在“参考点坐标”(Reference point coordinate)(*Mode = 0*) 中或在输入参数 *Position* (*Mode = 1*) 中定义的参考点位移。

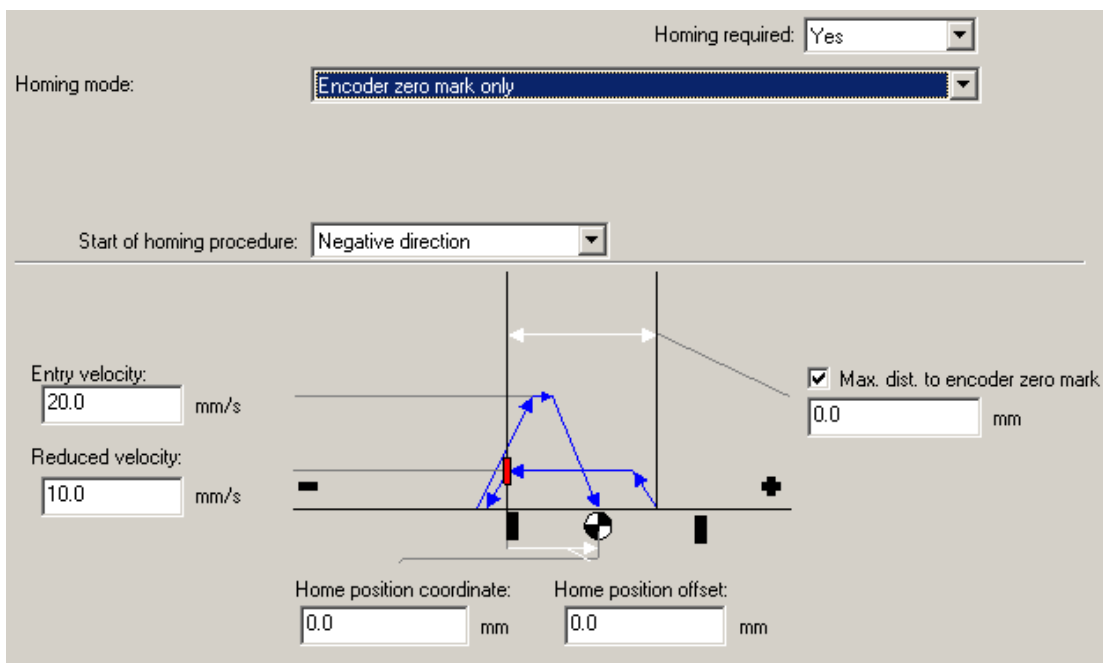
也可以通过设置“到外部零标记的最大距离”(Max. distance to external zero mark) 复选框来监视轴在参考点逼近的起点和信号沿检测之间移动的距离。如果在参考点逼近的起点和已组态距离的终点之间没有检测到沿，则回原点将因出错而取消。

- 阶段 2：参考点逼近

检测到组态的信号沿后，轴将加速/减速，从而以关闭速度逼近参考点坐标。

如果已组态了参考点位移，则轴以相应的距离（起点为同步位置）逼近该参考点。运动方向由参考点位移的符号以及检测到外部零标记后的减速斜坡长度确定（前提是参考点位于减速斜坡内）。

使用“仅编码器零标记”(Encoder zero mark only) 的主动回原点模式



例如，对于某些轴，编码器在轴的整个行程范围内只能设置一个编码器零标记，在这些轴上就会使用回原点到编码器零标记。回原点命令启动到编码器零标记的轴逼近。检测到编码器零标记后，轴以回原点速度逼近位移后的参考点。现在，轴位置具有在参考点坐标中定义的值。该图中的图形显示最终的运动顺序。

各个参数的含义：

参数	值	说明
需要回原点	是	轴必须回原点才能执行绝对运动命令。
	否	轴不必回原点就可以执行绝对运动命令。
回原点模式	-	在此例中：“仅编码器零标记” (Encoder zero mark only)
开始参考点逼近	正方向	正方向上的参考点逼近。
	负方向	负方向上的参考点逼近。
进入速度	-	检测到编码器零标记后，轴逼近（移位后的）参考点的速度
关闭速度	-	轴开始参考点逼近以及逼近编码器零标记的速度
回原点位置偏移	-	回原点位置偏移功能会将回原点位置移位所组态的距离，这意味着轴以“回原点速度”(Homing velocity) 在编码器零标记检测位置之后移动所组态的距离。 模数轴始终取最短距离。
到编码器零标记的最大距离	取消激活	没有监视到编码器零标记所需要移动的距离
	激活	监视从参考点逼近的起点到检测出编码器零标记时所移动的距离 如果该距离的差值超出了组态的值，则相应的工艺 DB 将返回错误 801D 。参考点逼近取消。

参考点逼近的顺序

- 阶段 1： 与编码器零标记同步
轴以组态的关闭速度，沿在“参考点逼近的起点”(Start of reference point approach) 中设置的方向开始其参考点逼近。 与编码器零标记的同步（阶段 1）在检测到编码器零标记后结束。 将轴位置设置为默认值减去在“参考点坐标”(Reference point coordinate)(*Mode = 0*) 中或在输入参数 *Position (Mode = 1)* 中定义的参考点位移。
也可以通过设置“到零标记的最大距离”(Max. distance to zero mark) 复选框，来监视轴在参考点逼近的起点和编码器零标记检测之间移动的距离。 如果在参考点逼近的起点和已组态距离的终点之间没有检测到编码器零标记，则回原点将因出错而取消。
- 阶段 2： 参考点逼近
检测到编码器零标记后，轴将加速/减速，从而以进入速度逼近参考点坐标。
如果已组态了参考点位移，则轴以相应的距离（起点为同步位置）逼近该参考点。 运动方向由参考点位移的符号以及检测到编码器零标记后的减速斜坡长度确定（前提是参考点位于减速斜坡内）。

4.5.11.3 回原点 —“被动回原点”(Passive homing) 标签

可以在 S7T Config 的轴 (Axis) > 回原点 (Homing) 对话框的“被动回原点”(Passive homing) 选项卡中组态被动回原点。与主动回原点相比，这二者的不同之处在于，被动回原点所需的回原点运动不是由回原点命令启动的。

“被动回原点”(Passive homing) 有三种可用的回原点模式：

- 参考凸轮和编码器零标记
- 仅外部零标记
- 仅编码器零标记

启用回原点功能：

MC_Home	<i>Mode = 2</i> <i>Position = x</i>	为当前位置分配了参考点输入参数 <i>Position</i> 的值。

成功回原点后，轴工艺 DB 将返回状态 *Statusword.HomingDone = TRUE*。

说明

请注意，并非所有驱动组件都支持所有的回原点模式或测量信号计算功能。 有关详细信息，请参考所用驱动组件的文档。
由工艺 DB 上相应的错误消息来表示在被动回原点操作期间对编码器零标记或参考凸轮监视的触发。 在这种情况下，当前轴运动终止。

使用“参考凸轮和编码器零标记”(Reference cam and encoder zero mark) 的被动回原点模式

Homing required: Yes

Homing mode: Homing output cam and encoder zero mark

Log. address of homing cam: 64

Bit number: 0

Direction of motion: Current direction

☐ Max. distance to homing output cam
0.0 mm

☐ Max. dist. to encoder zero mark
0.0 mm

Home position coordinate

轴经过参考凸轮后，下一个编码器零标记将触发轴回原点。将参考点的轴位置设置为通过“MC_Home”工艺功能的输入参数 *Position* 定义的值。

各个参数的含义：

参数	值	说明
需要回原点	是	轴必须回原点才能执行绝对运动命令。
	否	轴不必回原点就可以执行绝对运动命令。
回原点模式	-	在此例中：“参考凸轮和编码器零标记” (Reference cam and encoder zero mark)
运动方向	正方向	轴仅通过正逼近方向回原点到编码器零标记。
	负方向	轴仅通过负逼近方向回原点到编码器零标记。
	当前方向	轴在到达下一个编码器零标记时回原点。
参考凸轮的逻辑地址	[字节地址]	参考凸轮的逻辑字节地址 可以将参考凸轮连接至 T-CPU 的集成输入或连接至 DP(DRIVE) 上的 IO。
位号	[位号]	用于参考凸轮的信号的位地址

参数	值	说明
到回原点输出凸轮的最大距离	取消激活	没有监视到参考凸轮检测的距离
	激活	监视回原点功能的起点和参考凸轮检测之间的距离。如果该距离的差值超出了组态的值，则相应的工艺 DB 将返回错误 <i>801D</i> 。回原点功能取消。
到编码器零标记的最大距离	取消激活	没有监视到编码器零标记所需要移动的距离
	激活	监视轴在参考凸轮和编码器零标记之间移动的距离。如果该距离的差值超出了组态的值，则相应的工艺 DB 将返回错误 <i>801D</i> 。回原点功能取消。

使用“仅外部零标记”(External zero mark only) 的被动回原点模式

Homing required:

Homing mode:

On the external zero mark side:

Direction of motion:

☐ Max. distance to ext. zero mark

mm

Home position coordinate

在检测到外部零标记时，开始轴回原点。将轴设置为“MC_Home”工艺功能的 *Position* 输入参数的值。

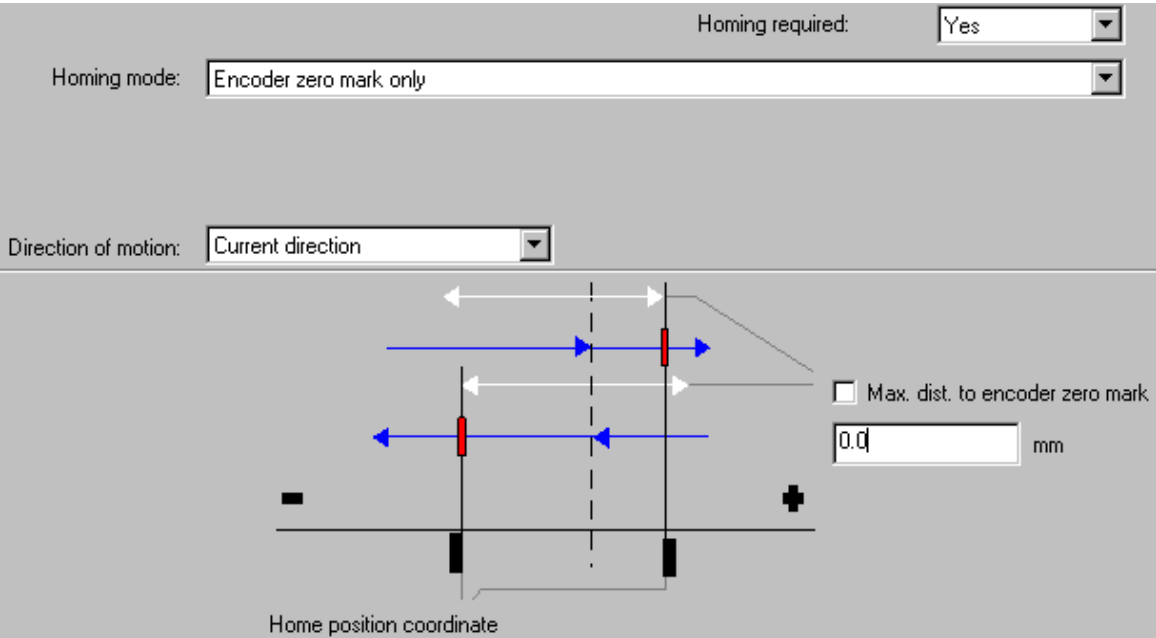
各个参数的含义：

参数	值	说明
需要回原点	是	轴必须回原点才能执行绝对运动命令。
	否	轴不必回原点就可以执行绝对运动命令。
回原点模式	-	在此例中：“仅外部零标记”(External zero mark only)
运动方向	正方向	轴仅通过正逼近方向回原点到外部零标记。
	负方向	轴仅通过负逼近方向回原点到外部零标记。
	当前方向	轴在到达下一个外部零标记时回原点。
在外部零标记一侧	左	移动方向： 正方向 在上升沿上触发轴回原点。 移动方向： 负方向 在下降沿上触发轴回原点。
	右	移动方向： 正方向 在下降沿上触发轴回原点。 移动方向： 负方向 在上升沿上触发轴回原点。
到外部零标记的最大距离	取消激活	没有监视到外部零标记所需要移动的距离
	激活	该功能将监视轴从其回原点功能的起点到检测出外部零标记所移动的距离 如果该距离的偏差超出了组态的距离，则相应的轴工艺 DB 将指示错误 801D。回原点功能取消。

说明

- 对于使用“外部零标记”(external zero mark) 的回原点，将外部零标记作为数字测量输入连接至驱动组件。
- 为了能够根据需要以“外部零标记”(external zero mark) 回原点模式执行参考点逼近，您应始终在“移动方向”(Direction of movement) 和“在外部零标记一侧”(on the side of the external zero mark) 上设置数值，使其与驱动器组态或功能兼容。“在外部零标记一侧”(on the side of the external zero mark) 和“移动方向”(Direction of movement) 参数不用于组态驱动器中的测量功能，仅反映其功能。
- 有关外部零标记检测的组态的信息，请参考相关的产品信息或驱动器手册。

使用“仅编码器零标记”(Encoder zero mark only) 的被动回原点模式



例如，使用“仅零标记”(Zero mark only) 的回原点模式用于装有一个在轴的整个行程范围内只能输出一个零标记信号的编码器的轴。

在检测到编码器零标记时，开始轴回原点。检测到编码器零标记后，将轴的位置值设置为“MC_Home”工艺功能的输入参数 *Position* 的值。

各个参数的含义：

参数	值	说明
需要回原点	是	轴必须回原点才能执行绝对运动命令。
	否	轴不必回原点就可以执行绝对运动命令。
回原点模式	-	在此例中：“仅编码器零标记”(Encoder zero mark only)
运动方向	正方向	轴仅通过正逼近方向回原点到编码器零标记。
	负方向	轴仅通过负逼近方向回原点到编码器零标记。
	当前方向	轴在到达下一个编码器零标记时回原点。
到编码器零标记的最大距离	取消激活	没有监视到编码器零标记所需要移动的距离
	激活	该功能将监视轴从其回原点功能的起点到检测到编码器零标记所移动的距离。 如果该距离的偏差超出了组态的距离，则相应的轴工艺 DB 将指示错误 801D。回原点功能取消。

使用“默认”(Default) 的被动回原点模式

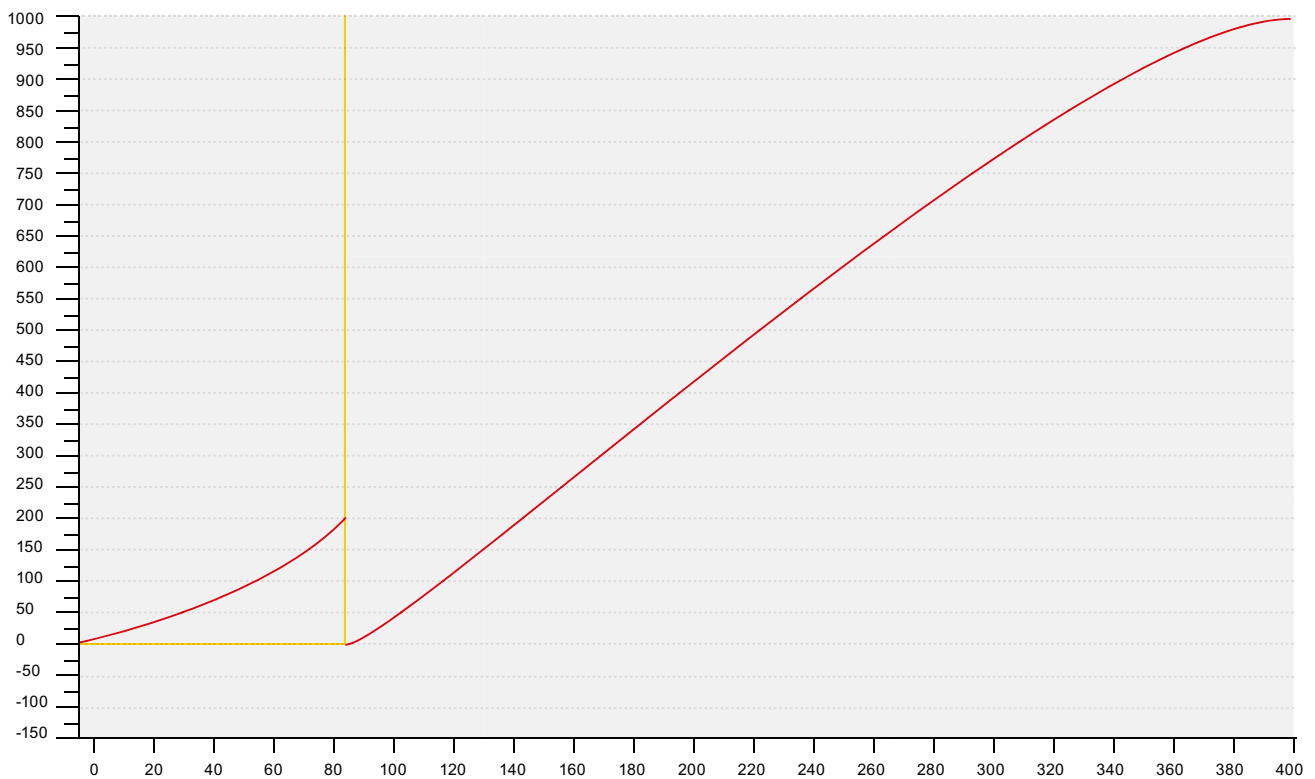
在 S7T Config 中创建新轴时，会预设“默认”(Default) 回原点模式。

如果组态的编码器提供零标记，则使用“仅编码器零标记”(encoder zero mark only) 回原点模式。如果编码器不提供零标记，则使用仅外部零标记回原点模式。

4.5.11.4 被动回原点模式的定位特性

以下几个示例表明了使用运动命令“MC_MoveRelative”和“MC_MoveAbsolute”时被动回原点模式的终点位置的响应：

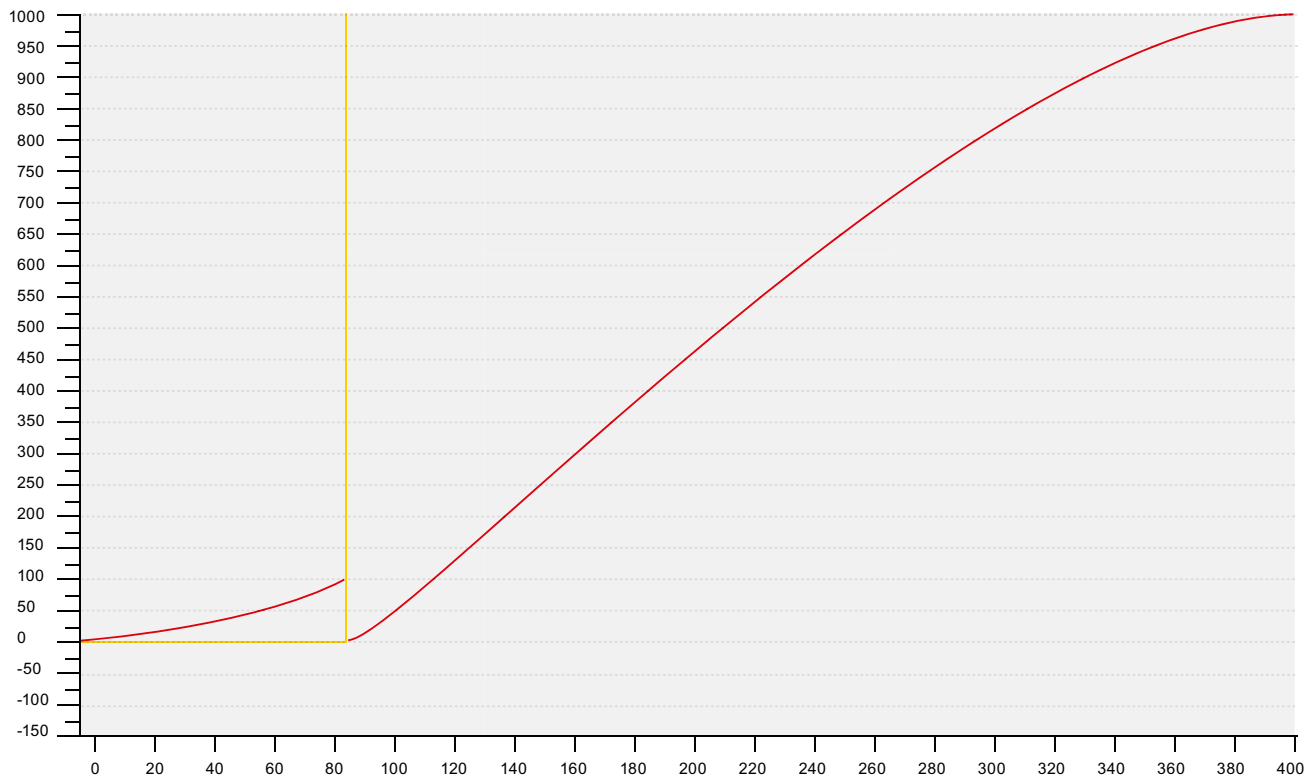
使用“MC_MoveRelative”的被动回原点模式的定位特性



针对被动回原点模式启动 MC_MoveRelative 命令 1000。

1. MC_MoveRelative 命令和被动回原点模式同时启动。
2. 到达黄色标记的回原点位置 200 时，轴回原点，轴的位置设置为位置 0。
3. 轴移动剩余的相对距离，移到位置 800 (1000-200)。

使用“MC_MoveAbsolute”的被动回原点模式的定位特性



针对被动回原点模式启动到位置 1000 的 MC_MoveAbsolute 命令。

1. MC_MoveAbsolute 命令和被动回原点模式同时启动。
2. 到达黄色标记的回原点位置 200 时，轴回原点，轴的位置设置为位置 0。
3. 根据新回原点位置，轴移动到位置 1000。

4.5.11.5 直接回原点

将实际轴位置设置为在 **MC_Home** 工艺功能中定义的值。没有激活的参考点位移设置。
该功能不执行运动。执行命令时使轴回原点。
启用回原点功能

MC_Home	<i>Mode = 3</i> <i>Position = x</i>	直接回原点： 为当前位置分配了输入参数 <i>Position</i> 的值。

成功回原点后，轴工艺 DB 将返回状态 *Statusword.HomingDone = TRUE*。

4.5.11.6 位置校正

从轴的实际位置值中扣除校正值。在这种情况下，与其它回原点模式不同，轴将保持其回原点状态（已回原点/未回原点）。
位置校正功能还可以用于操纵各个坐标系（基准坐标系、叠加坐标系）的设定值。这对于叠加凸轮传动能够在凸轮盘中生成参考非常重要。
启用回原点功能：

MC_Home	<i>Mode = 4</i> <i>Position = x</i>	实际值校正： 位置值 = (当前位置) - (参数位置)。
	<i>Mode = 6</i> <i>Position = x</i>	基坐标系中的设定值校正： 位置值 = (当前位置) - (参数位置)。
	<i>Mode = 7</i> <i>Position = x</i>	叠加坐标系中的设定值校正： 位置值 = (当前位置) - (参数位置)。

位置校正不会影响轴工艺 DB 中的 *Statusword.HomingDone* 状态。

4.5.11.7 未回原点的轴的运动

可以在**轴 (Axis) > 回原点 (Homing)** 对话框中确定绝对定位是否可用于未回原点的轴。
“需要回原点”(Homing required) 的可选设置：

- 否： 可以进行相对运动和绝对运动。 将监视软件限位开关。
- 是： 仅可以进行相对运动。 只要轴没有回原点便不监视软件限位开关。

4.5.12 监视功能

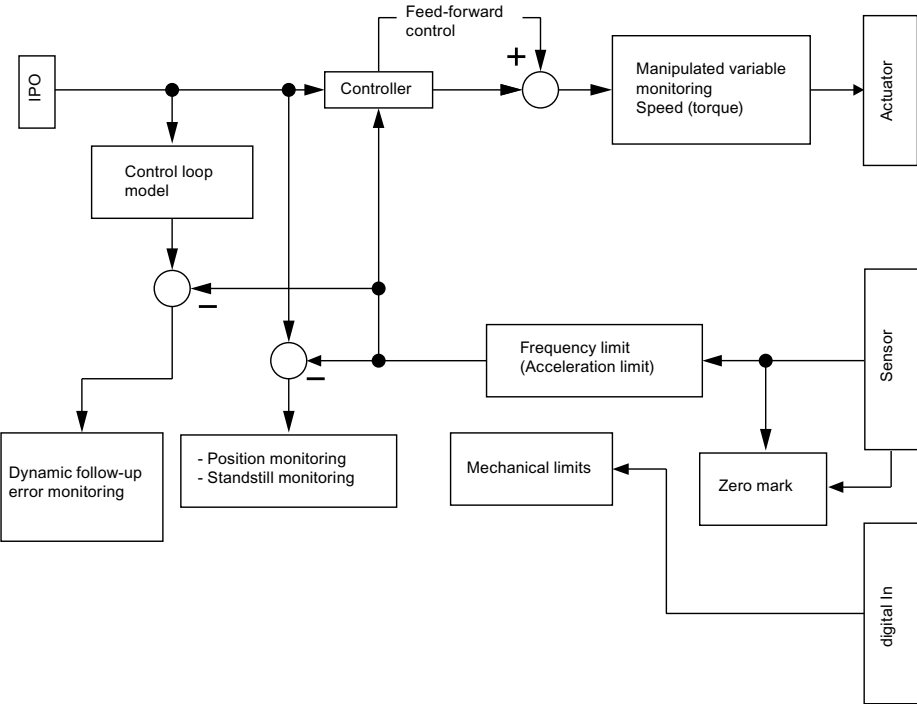
4.5.12.1 监视功能 — 总览

可以在 S7T Config 中组态的轴监视功能：

监视功能	速度控制轴	定位轴	同步轴
速度误差监视	X	-	-
定位监视	-	X	X
跟随误差监视	-	X	X
停止信号	X	X	X
软件限位开关	-	X	X
硬件限位开关	X	X	X
同步监视	-	-	X
受控变量监视（始终可用）	X	X	X
编码器限制频率监视	X（仅适用于编码器）	X	X

定位轴监视

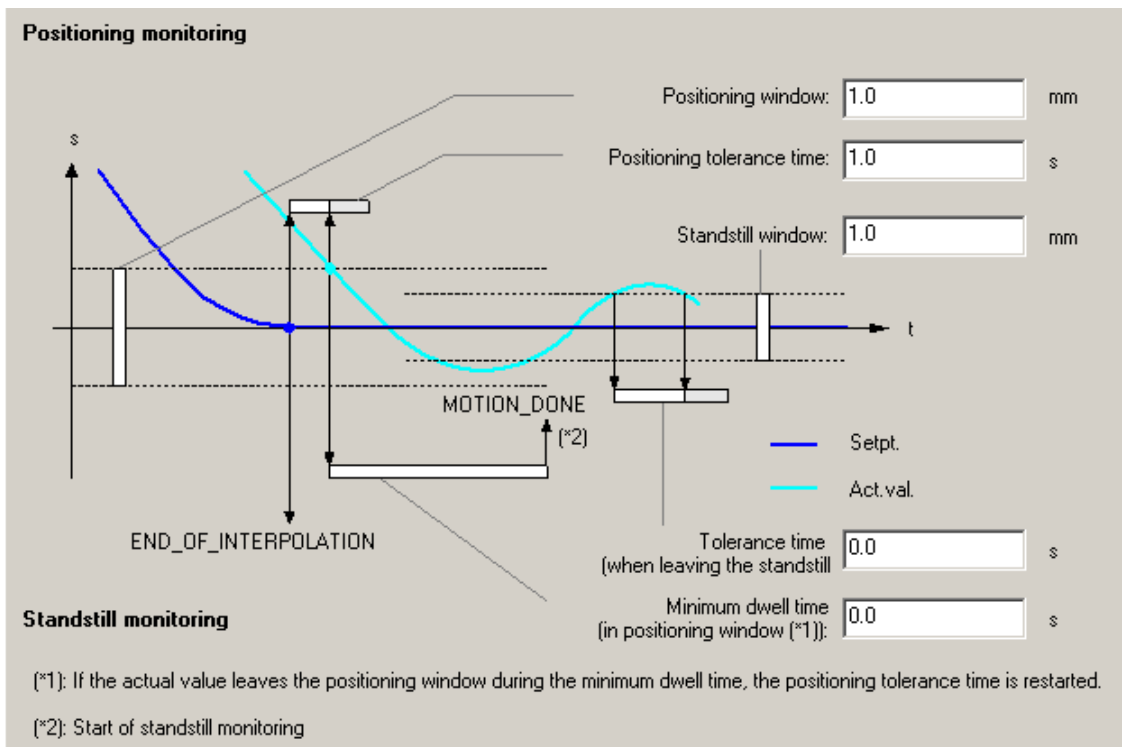
下图显示了一个定位轴监视的示例：



在工艺 DB 的 *ErrorStatus* 参数中报告错误。

4.5.12.2 监视功能 — 定位和停止监视

可以在轴 (Axis) > 监视功能 (Monitoring functions) 对话框的“定位和停止监视”(Positioning and standstill monitoring) 选项卡中设置用于监视位置控制轴和同步轴定位的限制值。



“定位和停止监视”(Positioning and standstill monitoring) 选项卡显示为轴设置了定位还是同步操作工艺。

定位监视

在定位即将结束时，该功能将监视轴进入目标位置。为此，指定一个定位窗口和一个时间公差，在该时间公差内必须到达终点位置。在设定值插补结束时激活监视。

也可以将定位窗口中的最小停留时间设置为在激活定位命令的正反馈之前结束。该时间可以用于振动过程和控制回路，为此公差窗口应小于超调振幅。

当实际值到达定位窗口时，将输出定位命令的正反馈。

如果触发了定位监视功能，则轴工艺 DB 将返回错误 8019。

停止（零速度）监视

当定位命令的位置设定值与目标位置的值相等并且停止监视激活的延迟已到期时，将启用停止监视。

如果轴移至已组态的停止公差窗口之外的时间超过已组态的时间段，则将触发停止监视。

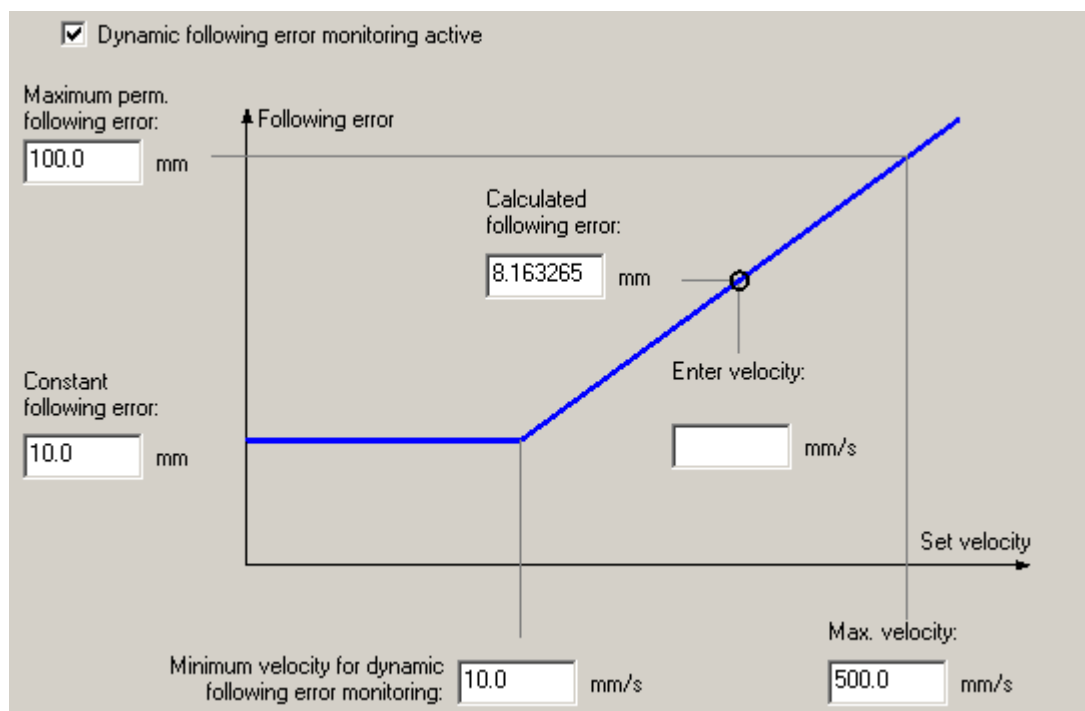
如果触发了停止监视，则轴工艺 DB 将返回错误 8018。

4.5.12.3 监视功能 —“跟踪误差监视”(Following error monitoring) 标签

可以在**轴 (Axis) > 监视功能 (Monitoring functions)** 对话框的“跟踪误差监视”(Following error monitoring) 选项卡中启用动态跟踪误差监视。

使用计算出的跟踪误差执行对位置控制轴的跟踪误差监视。如果实际位置值和位置设定值之间的偏移超出已编程的跟踪误差限制，轴工艺 DB 将返回错误 **8016**。允许的跟踪误差取决于轴的速度设定值。

当速度小于已组态的最小速度时允许的跟踪误差为常量，这是在“恒定跟踪误差”(constant following error) 参数中编程的。大于此限制时，跟踪误差将线性增加倒最大值，该最大值由最大速度时的参数“允许的最大跟踪误差”(maximum permitted following error) 定义。在最大速度时达到允许的最大跟踪误差。



如果在“输入速度”(Enter velocity) 输入框中指定一个值，则相应的功能值显示在“计算的跟踪误差”(Calculated following error) 框中。这允许您控制在输入的速度下跟踪误差可以有多大。

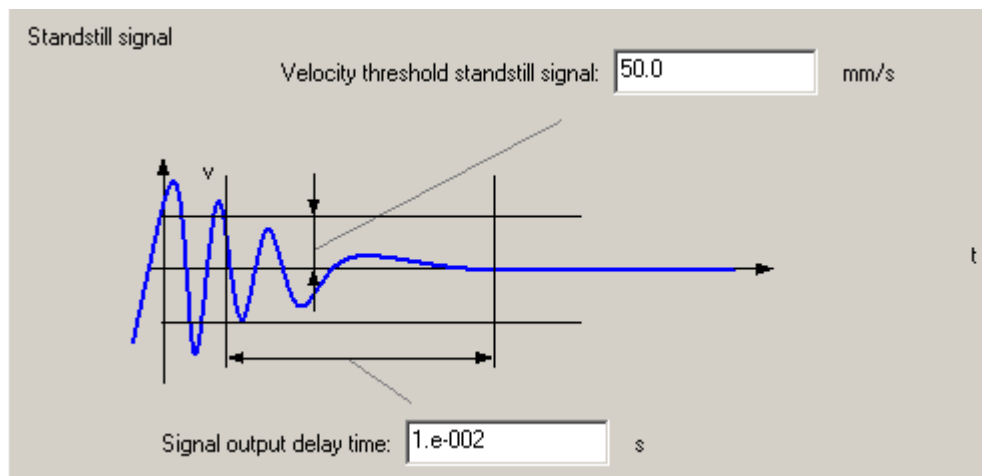
对于位置控制的实际轴，将显示“跟踪误差监视”(Following error monitoring) 选项卡。

4.5.12.4 监视功能 —“停止信号”(Standstill signal) 标签

可以在“轴”(Axis) > “监视功能”(Monitoring functions) 对话框的“停止信号”(Standstill signal) 选项卡中设置速度阈值和“信号输出延迟”(Signal output delay)。

如果当前速度至少在设定的延迟时间期间低于组态的速度阈值，则停止信号将在工艺 DB 的状态字中设置“停止”(standstill) 位。

在速度控制轴和定位轴上，将在速度控制模式下激活重写。在定位轴上，将在到达定位窗口时激活重写。

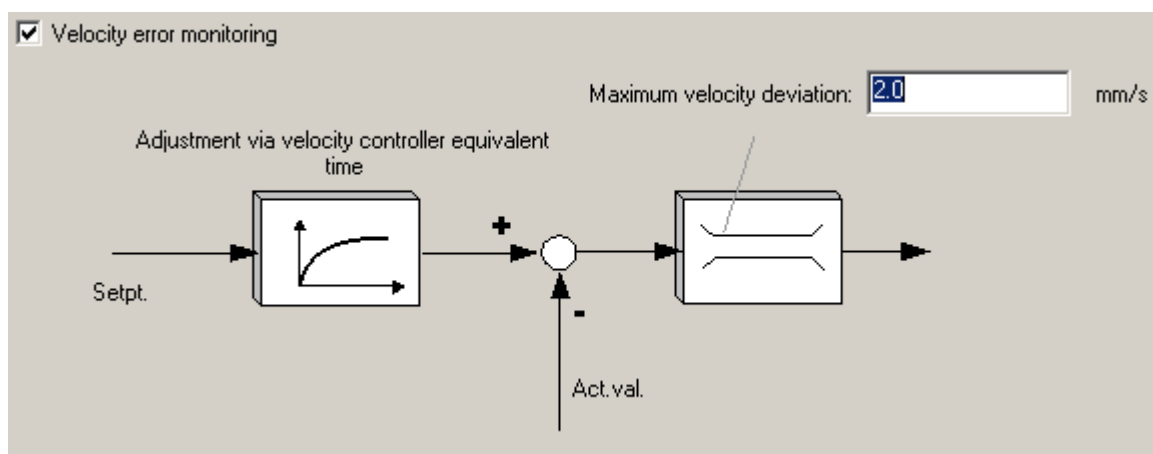


4.5.12.5 监视功能 —“速度监视”(Velocity monitoring) 标签

可以在**轴 (Axis) > 监视功能 (Monitoring functions)** 对话框的“速度误差监视”(Velocity error monitoring) 选项卡中启用速度误差监视。

速度误差监视仅与以下这些轴有关：

- 带有编码器的速度控制轴
- 操作在速度控制模式下的定位轴
(“MC_MoveVelocity”—输入参数 *PositionControl = FALSE*)



速度误差监视

在该对话框中启用速度误差监视。

最大速度误差：

仅当启用了速度误差监视时该输入框才可见。

在该对话框中输入最大速度误差。

4.5.12.6 操纵变量监视

对受控变量的最大值进行了限制，以便监视组态的速度限制。如果受控变量的值超出了可组态的最大值，则工艺 DB 将返回一个警告。

通过监视受控变量的斜率来监视可能的最大加速度和最大扭矩。

在“**轴 (Axis) > 限制 (Limits)**”对话框的“**位置和速度 (Position and velocity)**”选项卡中定义轴的最大值。

4.6 组态液压轴

4.6.1 组态液压轴 — 插入轴

HW Config 中的要求

- 已组态了 T-CPU
- 组态了模拟控制输出（模拟输出模块或模拟驱动器接口 IM 174/ADI4）
- 组态了编码器（模拟编码器、增量编码器或绝对编码器）
- 组态已保存在 HW Config 中

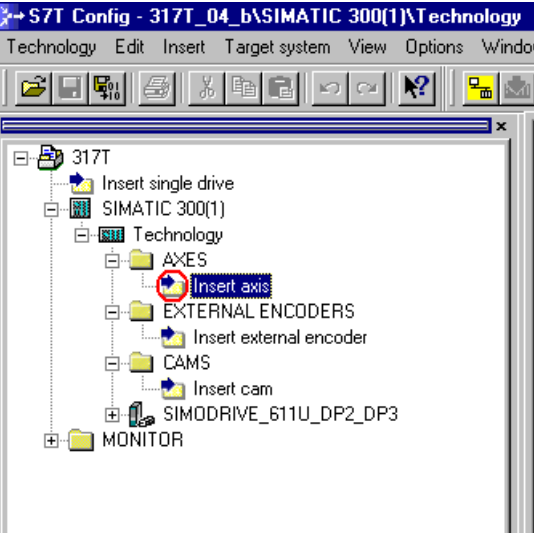
说明

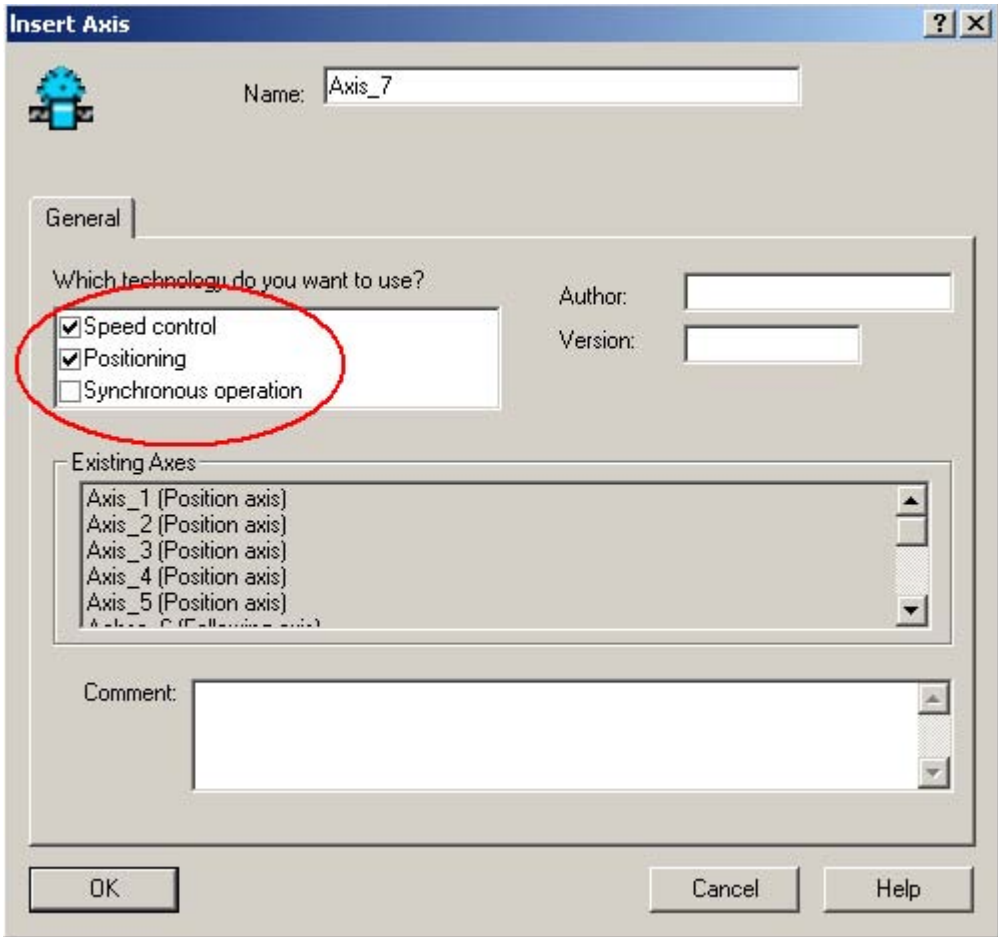
使用模拟驱动器接口 IM 174/ADI4 时，在 HW Config 中设置“标准报文 3”(Standard telegram 3)。

通过双击 IM 174/ADI4 对象在 HW Config 中调用“DP 从站属性”(DP Slave properties)。在“组态”(Configuration) 选项卡的“默认”(Default) 下拉列表中选择“标准报文 3”(Standard telegram 3)。

在 S7T Config 中插入新轴

下面以液压定位轴为例，介绍了插入和组态轴的基本操作步骤。液压轴仅可作为实际轴插入。

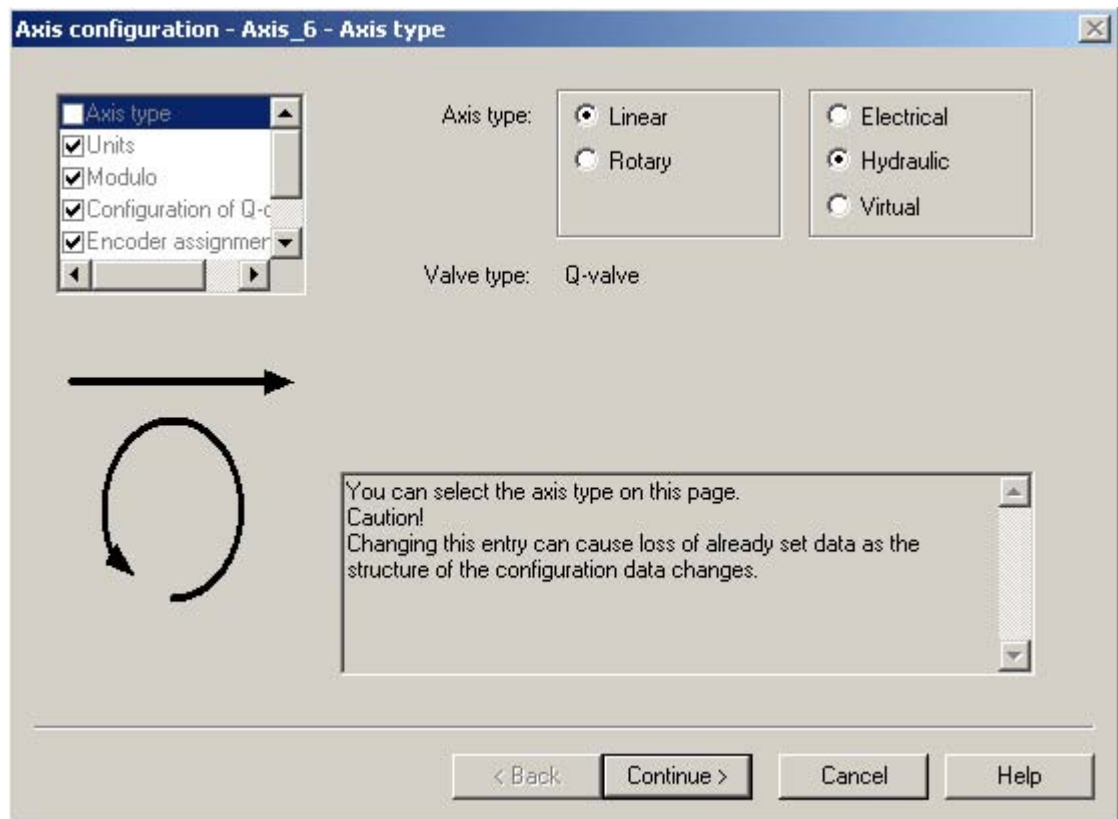
步骤	说明
1.	在 S7T Config 的浏览器中，双击“插入轴”(Insert axis)。 

步骤	说明
2.	在“常规”(General) 选项卡中，选择“速度控制”(Speed control) 和“定位”(Positioning) 工艺。  可以编辑名称、输入作者、定义版本名称以及输入注释。“工艺对象管理”(Technology Objects Management) 将工艺对象的名称作为符号名称应用于 STEP 7 中对象的工艺 DB。
3.	单击“确定”(OK) — 这可以打开轴向导。

说明

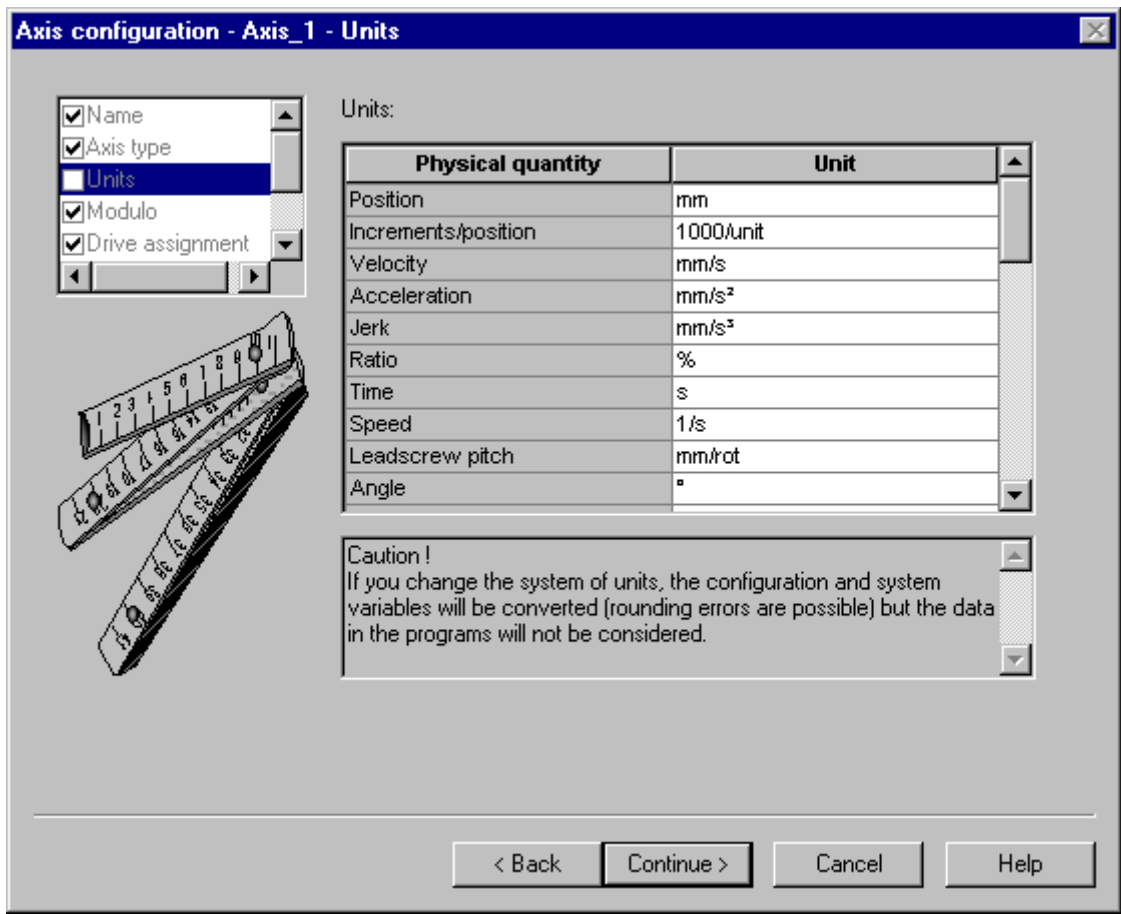
组态新轴后立即便可定义轴工艺（速度、定位、同步操作）。日后要更改轴工艺，必须删除并重新创建轴。

4. 在下一个对话框中，选择“线性”(linear) 或“旋转”(rotary) 以及“液压”(hydraulic) 轴类型。将阀类型定义为“Q 阀”(Q valve)。



单击“下一步”(Next)。

5. 从下一个对话框的列表中为轴组态和用户程序选择物理单位。



注意：

物理单位设置仅适用于该轴。 根据轴组态，该列表还可能显示与该轴不相关的物理单位。

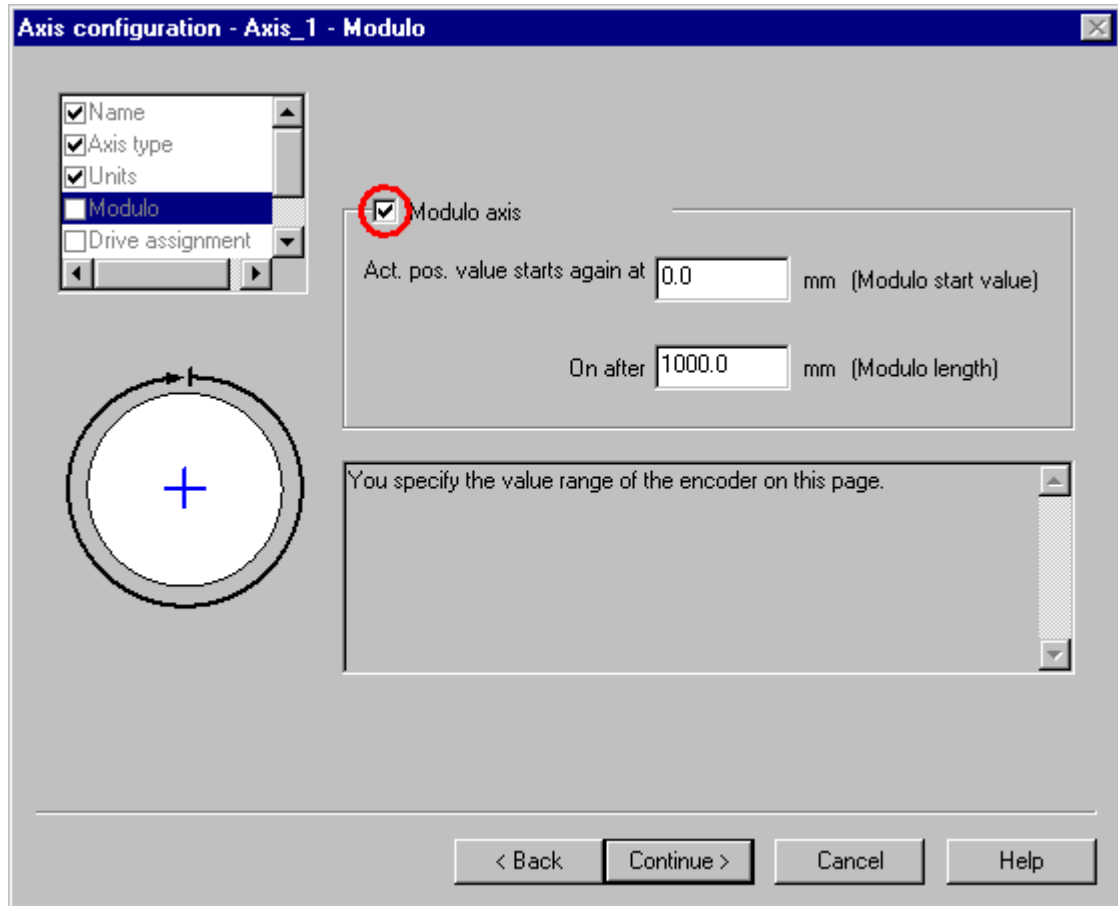
示例： 为“MC_MoveAbsolute”工艺功能的输入参数 *Position* 设置的值“1000.0”（位置单位：[mm]）即相当于位置值 1000.0 mm。

注意：

如果更改单位设置，在用户程序中设置的数字值（例如，为运动命令设置的数字值）将**不会**转换为新单位。 值始终参考已组态的单位！

单击“下一步”(Next)。

6. 如果您要以模数模式操作该轴，请设置“模数轴”(Modulo axis) 复选框，然后键入所需的模数长度和起始值。



单击“下一步”(Next)。

有关操作步骤的更多信息，请参考下面列出的章节：

- 『组态液压轴 — 通过 IM 174/ADI4 的 Q 输出』
- 『组态液压轴 — 通过模拟输出模块的 Q 输出』

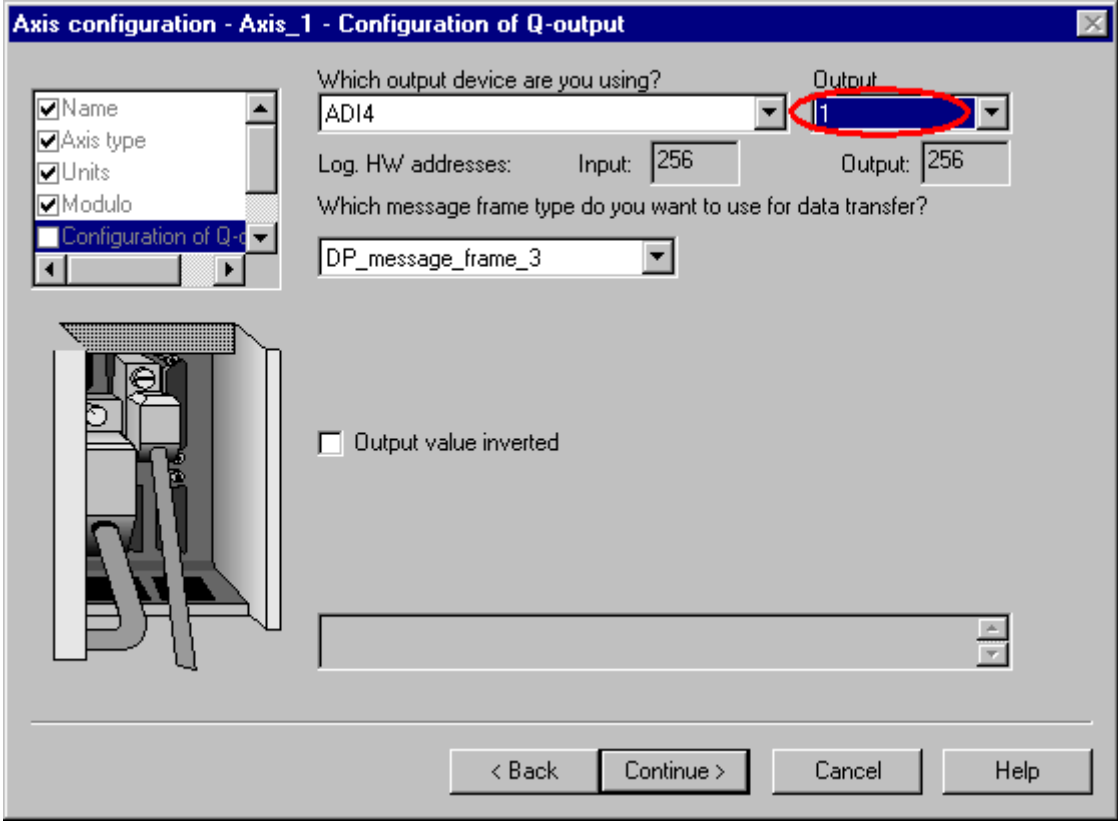
4.6.2 组态液压轴 — 通过 IM 174/ADI4 的 Q 输出

组态模拟驱动器接口 IM 174/ADI4 的 Q 输出

以下说明基于『组态液压轴 — 插入轴』一章。

1.

在下一个对话框中定义模拟输出模块的属性。



您使用的是哪个输出设备？	选择模拟驱动器接口 IM 174/ADI4。
输出	选择驱动器接口的模拟输出。
逻辑 硬件地址： 输入， 输出	不能编辑逻辑地址， 因为逻辑地址是通过选择模拟输出定义的。
您想要将哪种消息帧类型用于数据传输？	只允许“标准报文 3”(Standard telegram 3)（此处为“DP_Tel3”）作为消息帧类型。 消息帧必须与 HW Config 中的选择相匹配。
倒置值输出	选择该复选框， 以设置将反向控制输出到液压驱动器。
完成选择后， 单击“下一步”(Next)。	

2a.

在随后出现的对话框中，将增量传感器分配给液压轴。该编码器必须支持“长方形 TTL”(rectangular TTL) 模式。

有关使用模拟绝对编码器的信息，请参考『组态液压轴 — 通过模拟输出模块的 Q 输出』一章。如果使用绝对编码器，请继续步骤 2b。

在何处连接位置编码器？	选择驱动器接口模块 IM 174/ADI4 的编码器接口。
逻辑 硬件地址： 输入/输出：	不能编辑逻辑地址，因为逻辑地址是由所选的编码器定义的。
您想要将哪种消息帧类型用于数据传输？	只允许“标准报文 3”(Standard telegram 3)（此处为“DP_Tel3”）作为消息帧类型。
编码器类型：	从下拉列表中选择“增量编码器”(Incremental encoder)。
测量系统：	根据所用的编码器，选择“线性编码器系统”(Linear encoder system) 或“旋转编码器系统”(Rotary encoder system)。
完成设置后，单击“下一步”(Next)。	

3a.

在下面的对话框中组态编码器数据。

Axis configuration - Axis_6 - Inc. encoder data

☒Units

☒Modulo

☒Configuration of Q-c

☒Encoder assignmer

☐Inc. encoder data

Grid line spacing

2048

mm

Multiplication factor of the cyclic act. val. (Gn_XIST1):

0

.

Enter the special data of the selected encoder on this page.

< Back

Continue >

Cancel

Help

编码器精度:	有关编码器精度的信息，请参考编码器文档。
周期性实际值的倍增因子	根据在驱动组件的组态中使用的值来校准周期性实际值的倍增因子。
完成设置后，单击“下一步”(Next)。	
您已成功完成了轴组态。	

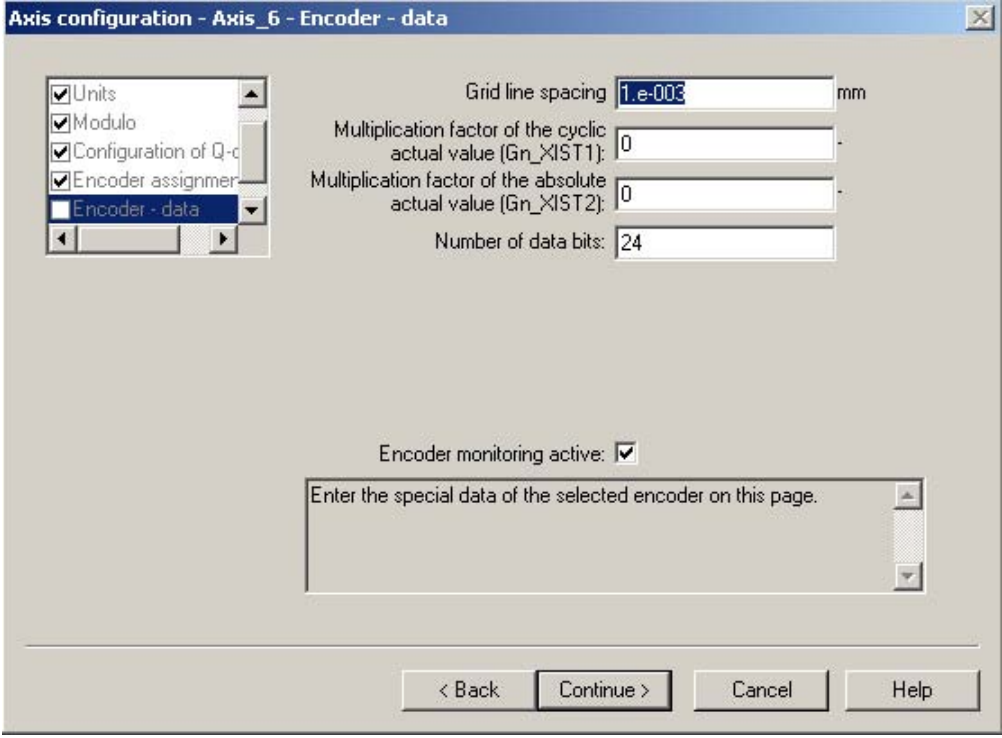
198

S7-Technology

功能手册, 03/2008, A5E01078448-06

- 2b. 在随后出现的对话框中，将绝对编码器分配给液压轴。该编码器必须支持“SSI”模式。

在何处连接位置编码器？	选择驱动器接口模块 IM 174/ADI4 的编码器接口。
逻辑 硬件地址：输入/输出：	不能编辑逻辑地址，因为逻辑地址是由所选的编码器定义的。
您想要将哪种消息帧类型用于数据传输？	只允许“标准报文 3”(Standard telegram 3)（此处为“DP_Tel3”）作为消息帧类型。
编码器类型：	从下拉列表中选择“绝对编码器”(Absolute encoder) 或“绝对编码器循环绝对”(Absolute encoder cyclic absolute)
测量系统：	根据所用的编码器，选择“线性编码器系统”(Linear encoder system) 或“旋转编码器系统”(Rotary encoder system)。
完成设置后，单击“下一步”(Next)。	

3b.	在下一个对话框中，定义所使用的编码器的网格划分和周期性实际值的倍增因子。根据在驱动组件的组态中使用的值来校准周期性实际值的倍增因子。 
网格间距	网格间距说明了编码器脉冲间的距离。
周期性实际值的倍增因子	连接至驱动器的编码器可以提供精度比实际编码器精度（高精度）更高的脉冲。 默认情况下，周期性实际值的倍增因子设置为 0。系统自动使用倍增因子 2048(2¹¹)。 偏差值的示例： 连接至驱动器的绝对编码器每转返回 4096(2¹²) 个脉冲（编码器精度）。在这种情况下，为周期性实际值输入乘数 12。
绝对实际值的倍增因子	将通电（启动）时传送的编码器实际值与绝对实际值的倍增因子相乘。 默认情况下，绝对实际值的倍增因子设置为 0。系统自动使用倍增因子 512(2⁹)。
还必须在驱动器上设置倍增因子！ - 在 SINAMICS S120 上：参数 p0418 和 p0419（对于编码器 1）； （可以在 r0979.3 和 r0979.4 中读取有效值） - 在 SIMODRIVE 611U 上：参数 1042 和 1043（对于编码器 1） - 在 MASTERDRIVE MC 上：参数 171	

	数据位数:	总脉冲数 示例: 4096 个脉冲 = 2^{12} 指数 12 = 12 个数据位
	完成设置后, 单击“下一步”(Next)。	

结束

您已成功完成了轴组态。

Axis configuration - Axis_6 - Completion

☒Axis type

☒Units

☒Modulo

☒Configuration of Q-output

☒Encoder assignment

☒Position value from I/O

☐Completion



All the necessary data for configuration has been entered:

Configuration of this axis:

Name:
- Axis_6

Technology:
- Position axis

Axis type:
- Drive axis
- Modulo selected
+ Start value: 0.0
+ End value: 1000.0

Hydraulic axis
Configuration of Q-output
- Analog output module
- Logical HW address:

☐ Do not display information anymore

< Back

Finish

Cancel

Help

单击“完成”(Finish) 退出轴组态对话框。

说明

使用工艺功能“MC_Power”启动带有位置控制的液压轴的条件:

- 必须在 S7T Config 中将液压轴作为阀控曲线分配给凸轮盘。
- 必须由“MC_SetCharacteristic”工艺功能激活该阀控曲线。

S7-Technology
功能手册, 03/2008, A5E01078448-06

201

4.6.3 组态液压轴 — 通过模拟输出模块的 Q 输出

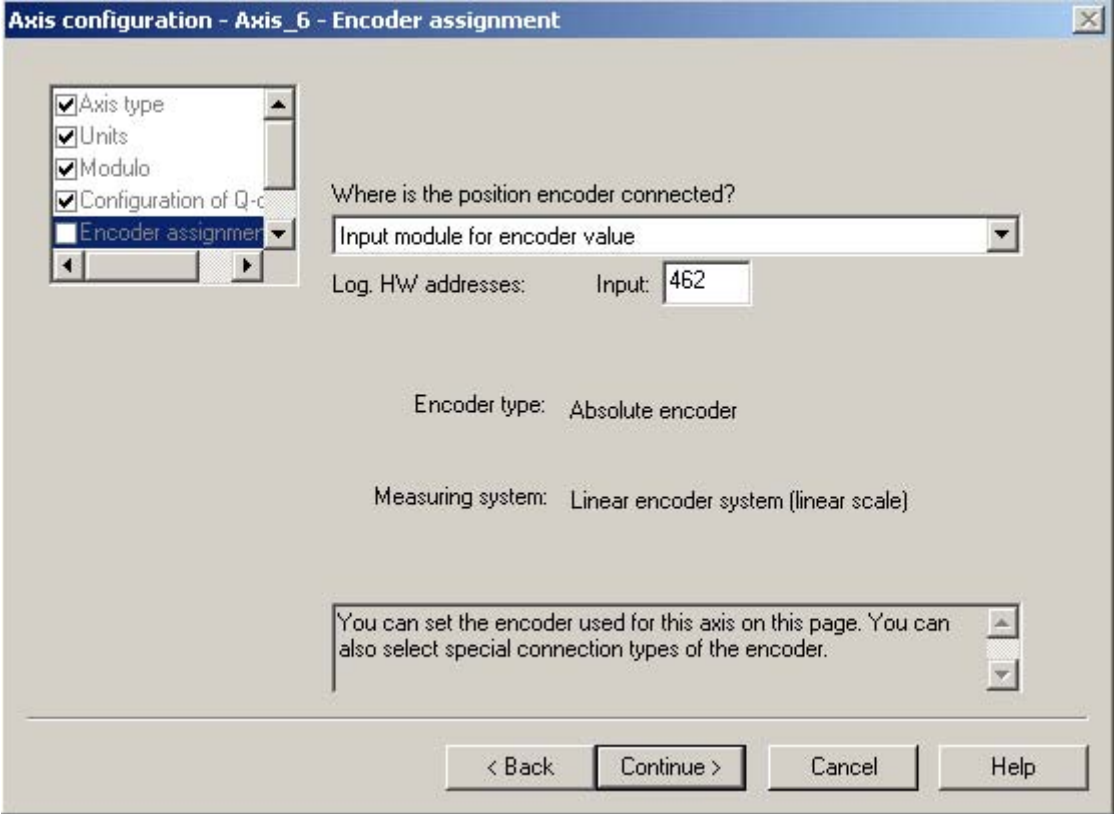
组态 ET 200M 或 ET 200S 的模拟输出模块的 Q 输出

以下说明基于『组态液压轴 — 插入轴』一章。

1.

在随后出现的对话框中，指定模拟输出模块的属性。

您使用的是哪个输出设备？	从下拉列表中选择“模拟输出模块”(analog output module)。
逻辑 硬件地址： 输出：	输入 ET 200M 或 ET 200S 的模拟输出模块的硬件地址。
格式：	为 ET 200M 和 ET 200S 的模拟输出模块选择“左对齐”(left aligned) 格式。
输出反向的值：	如果模拟值是流量的倒数，请选中该复选框。
精度：	有关精度的信息，请参考“HW Config”的硬件目录（无符号的精度）。
激活阀使能输出：	设置该复选框，以使用工艺功能“MC_Power”的使能状态作为控制信号。
逻辑地址：	如果激活“阀使能输出”(Valve enable output) 复选框，则可以在此处设置使能输出信号的逻辑地址。
位号：	输入使能输出信号的位号。
完成设置后，单击“下一步”(Next)。	

2.	在随后出现的对话框中，将液压轴的模拟绝对编码器参数化。有关使用增量编码器（矩形 TTL）和绝对编码器 (SSI) 的信息，请参考『组态液压轴 — 通过 IM 174/ADI4 的 Q 输出』一章。 
	在何处连接位置编码器？ 从下拉列表中选择“编码器值输入模块” (Encoder value input module)。
	逻辑 硬件地址： 输入： 输入 ET 200M 或 ET 200S 的模拟输入模块的硬件地址。
完成设置后，单击“下一步”(Next)。	

3.

在随后出现的对话框中，将位置值参数化。

Axis configuration - Axis_6 - Position value from I/O

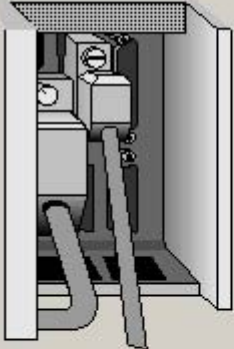
☒Units

☒Module

☒Configuration of Q-c

☒Encoder assignmen

☐Position value from



Position value = Raw value * +

Number of usable bits:

Format:

Minimum raw value:

Maximum raw value:

Error tolerance time: s

Activate filter ☐

< Back

Continue >

Cancel

Help

因子： /偏移：	设置因子 (<i>Factor</i>) 以定义位置值的斜率，设置 偏移 (<i>Offset</i>) 以定义位移。
可利用的位数：	输入模拟模块的可利用数据位数。
格式：	为 ET 200M 和 ET 200S 的模拟模块选择“左对齐” (left aligned)。
最小原始值：	输入模拟模块的最小原始值。
最大原始值：	输入该模拟模块的最大原始值。
误差公差时间：	输入误差公差时间。
激活过滤器：	设置该复选框，以对位置的模拟值应用 PT1 过滤器。
时间常量 PT1 过滤器	如果设置了“激活过滤器”(Activate filter) 复选框，则可以为 PT1 过滤器输入一个时间常量。
完成设置后，单击“下一步”(Next)。	

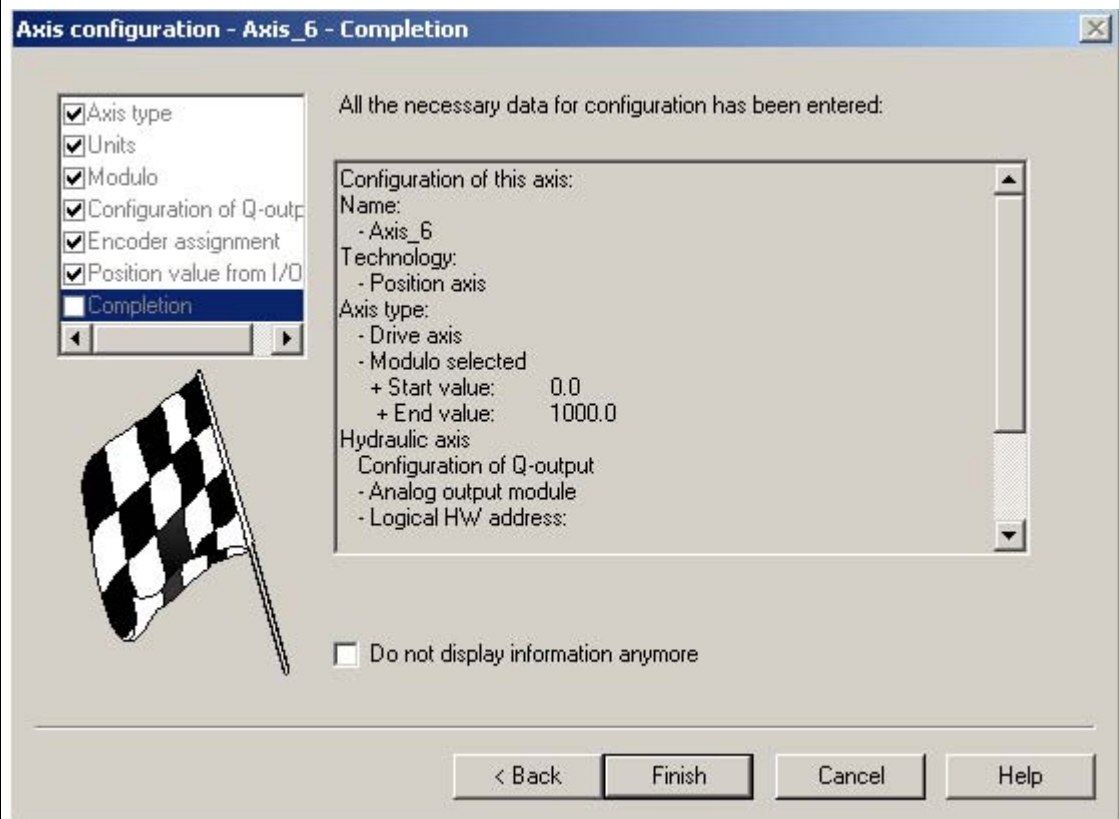
204

S7-Technology

功能手册, 03/2008, A5E01078448-06

4.

您已成功完成了轴组态。



单击“完成”(Finish) 退出轴组态对话框。

说明

使用工艺功能“MC_Power”启动带有位置控制的液压轴的条件:

- 必须在 S7T Config 中将液压轴作为阀控曲线分配给凸轮盘。
- 必须由“MC_SetCharacteristic”工艺功能激活该阀控曲线。

4.6.4 确定并添加一个阀轮廓

使用阀控曲线映射阀的受控变量（例如 -10 V 到 +10 V）和液压轴的速度之间的非线性。使用“凸轮盘”(cam disk) 工艺对象作为阀控曲线。阀控曲线由反映了相对达到速度的阀位置的数值对构成。

使用符号浏览器和跟踪工具确定阀控曲线

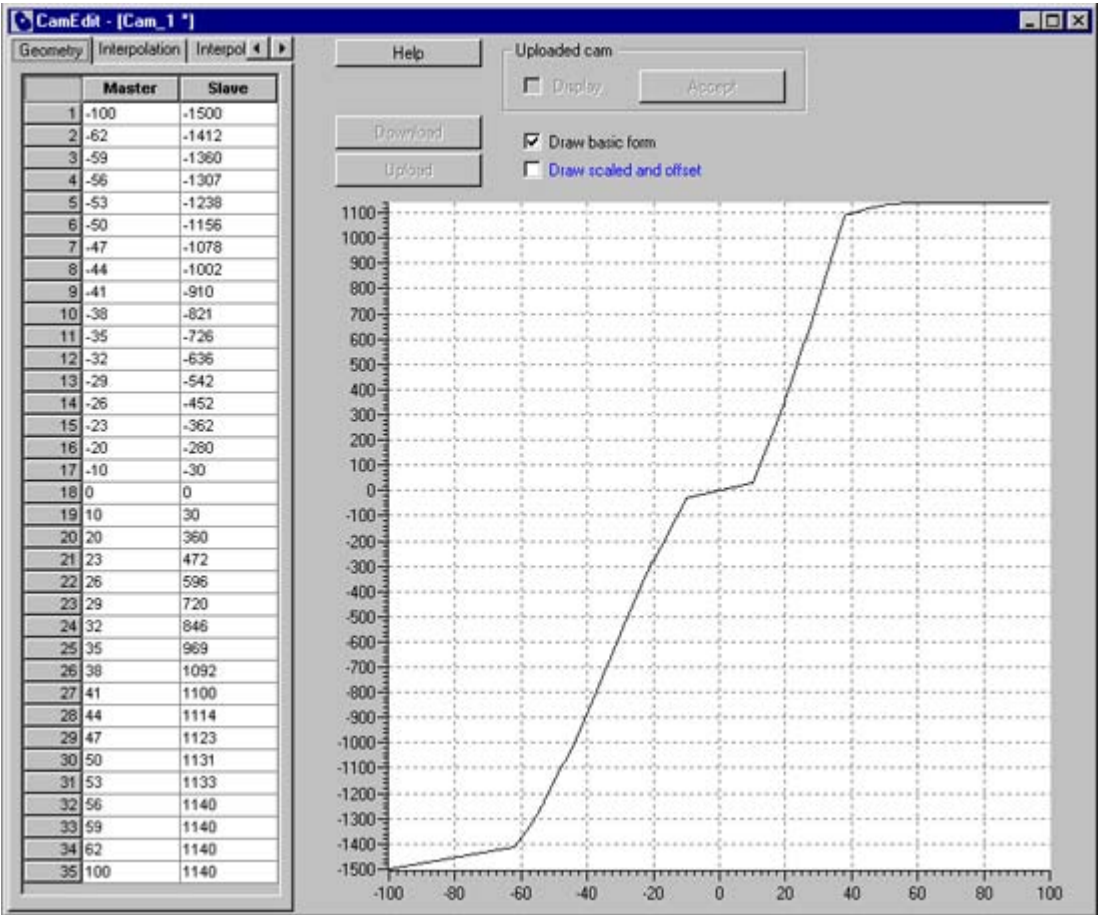
1.	验证是否可以移动液压轴（液压存在、阀准备就绪等）。
2.	验证 没有 为液压轴分配阀控曲线（S7T Config 的浏览器中的“曲线”[Profiles] 设置）。
3.	通过在 <i>Mode = 2</i> 时调用“MC_Power”工艺功能来启用液压轴。
4.	在 S7T Config 中编写以下变量，以在“跟踪”(Trace) 中进行记录： • <i>actordata.qoutputvalue</i> — 作为相对于 Q 输出的百分比的受控变量 • <i>sensordata.sensordata[n].velocity</i> — 液压轴的速度
5.	在 S7T Config 的符号浏览器中将系统变量 <i>servosettings.additionalqoutputvalueswitch</i> 设置为 YES。
6.	开始跟踪。
7.	在符号浏览器中的系统变量 <i>servosettings.additionalqoutputvalue</i> 中输入一个小的百分数，以移动轴。输入值 0 停止该轴。
8.	依次输入介于 0 % 和 100 % 之间的值，以便获得足够的阀控曲线插补点。
9.	在 S7T Config 中，使用 CamEdit 将已记录的值传送到阀控曲线的凸轮盘。将受控变量传送到“主站”(Master) 列，将速度值传送到“从站”(Slave) 列。

使用 STEP 7 程序和“跟踪”(Trace) 确定阀控曲线

1.	验证是否可以移动液压轴（液压存在、阀准备就绪等）。
2.	验证 没有 为液压轴分配阀控曲线（S7T Config 的浏览器中的“曲线”[Profiles] 设置）。
3.	通过在 <i>Mode = 0</i> 时调用“MC_Power”工艺功能来启用液压轴。
4.	在 S7T Config 中编写以下变量，以在“跟踪”(Trace) 中进行记录： • <i>actordata.qoutputvalue</i> — 作为相对于 Q 输出的百分比的受控变量 • <i>sensordata.sensordata[n].velocity</i> — 液压轴的速度
5.	开始跟踪。
6.	使用带有 <i>PositionControl = FALSE</i> 的“MC_MoveVelocity”工艺功能，以不同的速度移动该轴。 注意： 由于您尚未激活阀控曲线，因此 <i>sensordata.sensordata[n].velocity</i> 中显示的速度不必与通过“MC_MoveVelocity”设置的默认速度一致。 可以将轴停止在已定义的位置，以防止轴移至固定终点挡板。使用工艺功能“MC_Stop”设置该位置。
7.	依次设置若干速度值（直到最大值），以便为阀控曲线获得足够的插补点。
8.	在 S7T Config 中，使用 CamEdit 将已记录的值传送到阀控曲线的凸轮盘。将受控变量传送到“主站”(Master) 列，将速度值传送到“从站”(Slave) 列。

还可以使用“MC_CamClear”、“MC_CamSectorAdd”和“MC_CamInterpolate”工艺功能创建凸轮盘（而不使用 CamEdit）。

示例 — 比例方向阀的曲线

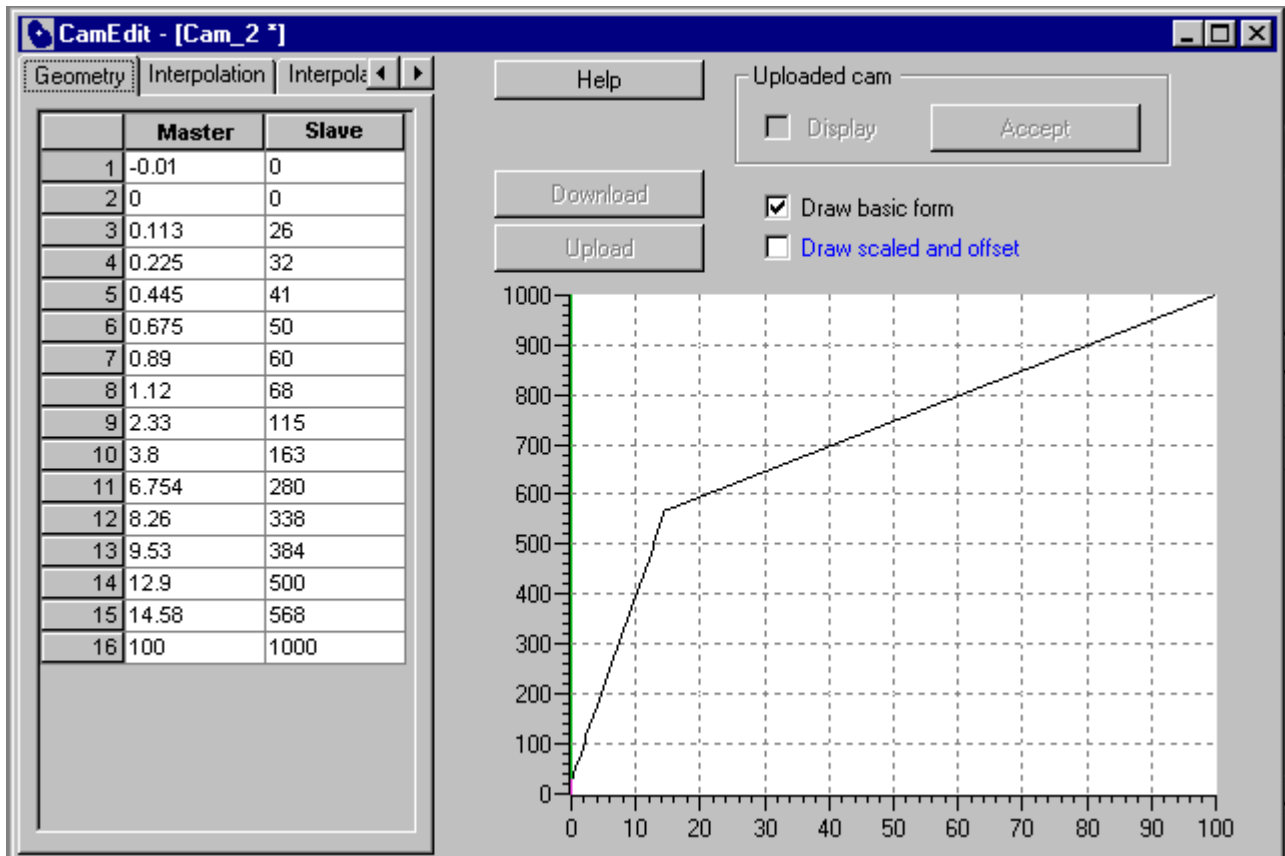


在“主站”(Master) 列中输入阀控制输出，作为“%”单位。“从站”(Slave) 列包含液压轴相应的速度值。

示例 — 带有切换方向阀的变速泵的阀控曲线

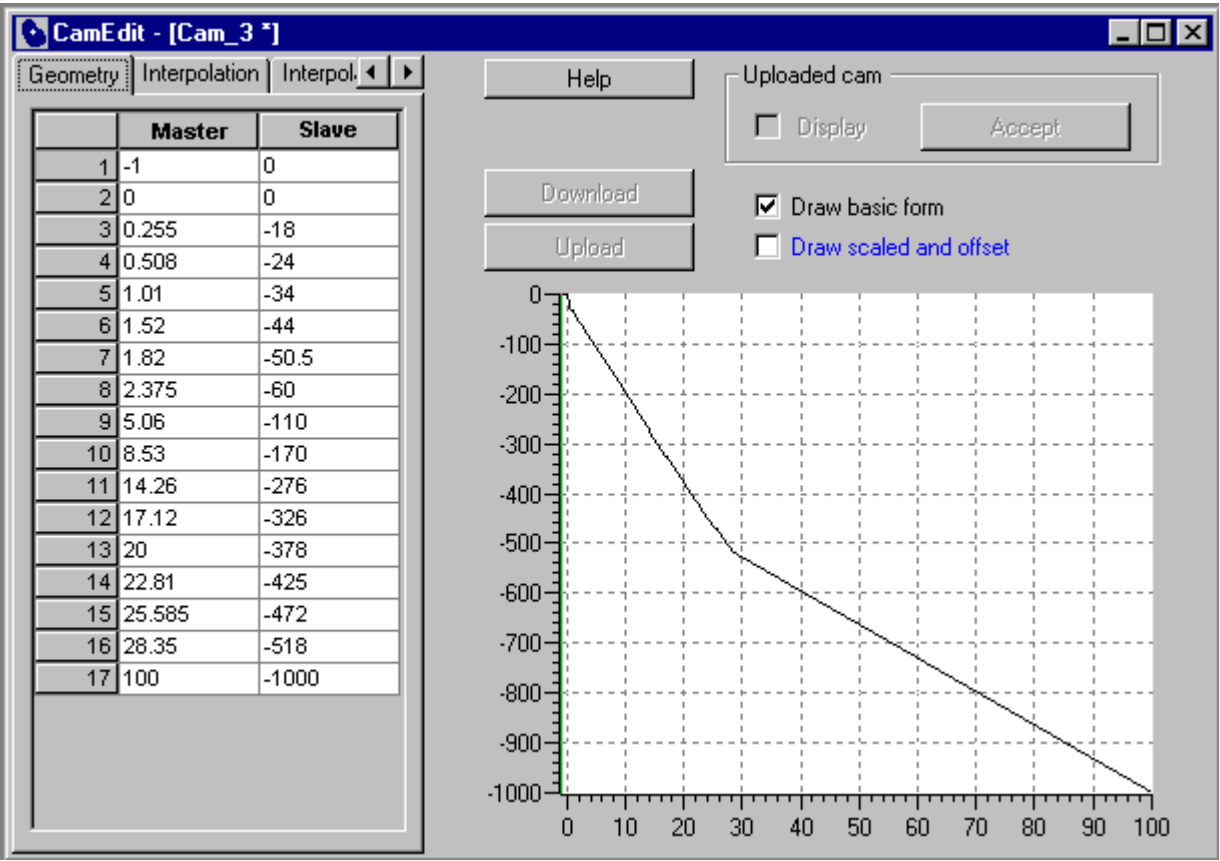
在包含带有切换方向阀的变速泵的系统，可以通过控制切换方向阀来实现负方向运动。使用单独的阀控曲线进行正向移动和反向移动。

用于正方向移动的阀控曲线的示例：



在“主站”(Master) 列中输入阀控制输出，作为“%”单位。“从站”(Slave) 列包含液压轴相应的速度值。任何由于控制功能而可能暂时出现的负控制值在行 1 中均限制为 0。

用于负方向移动的阀控曲线的示例：



用于负方向移动的阀控曲线的结构相当于用于正方向移动的阀控曲线的结构。第一个值对也将负受控变量限制为 0。

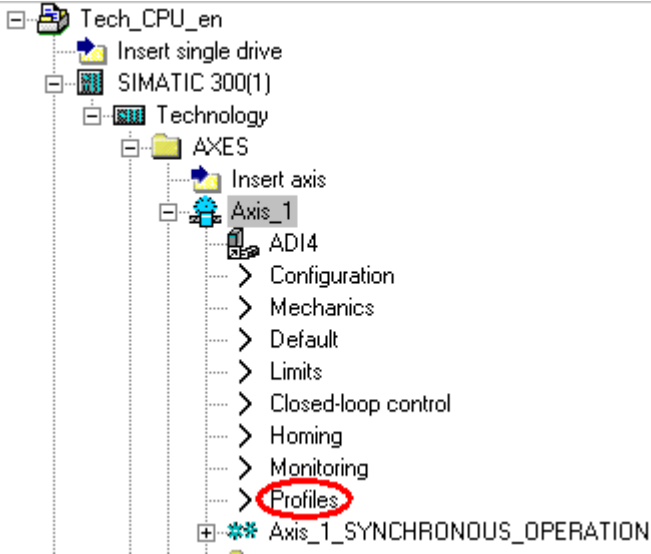
根据移动方向，通过调用“MC_SetCharacteristic”工艺功能在用户程序中激活相关的阀控曲线。

4.6.5 分配阀轮廓

将相应的凸轮盘分配给液压轴作为可能的阀控曲线，以便您可以通过调用“MC_SetCharacteristic”工艺功能来激活阀控曲线。

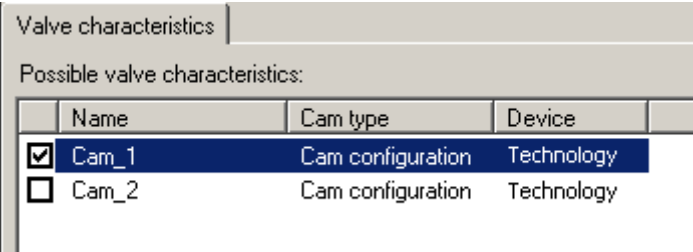
1.

要执行此操作，请双击 S7T Config 浏览器中的“曲线”(Profile)。



2.

在下一个对话框中，选择凸轮盘用作此液压轴的阀控曲线。



4.6.6 添加数据记录以进行数据记录转换

引言

“MC_ChangeDataset”工艺功能可用于在轴上执行数据记录转换。可以为速度控制轴、定位轴和同步轴创建数据记录。虚拟轴始终只有一个数据记录。

例如，数据记录转换功能可用于在电机编码器和机器编码器之间进行切换，或者编辑控制器参数。

说明

编辑数据记录参数时，请注意，数据记录中的某些参数必须相等：

- 更改结构的参数（例如，不同的控制器类型，如 PV 或 PID 控制器）
- 重要的激活参数（例如，跟踪监视开/关、DSC 开/关）

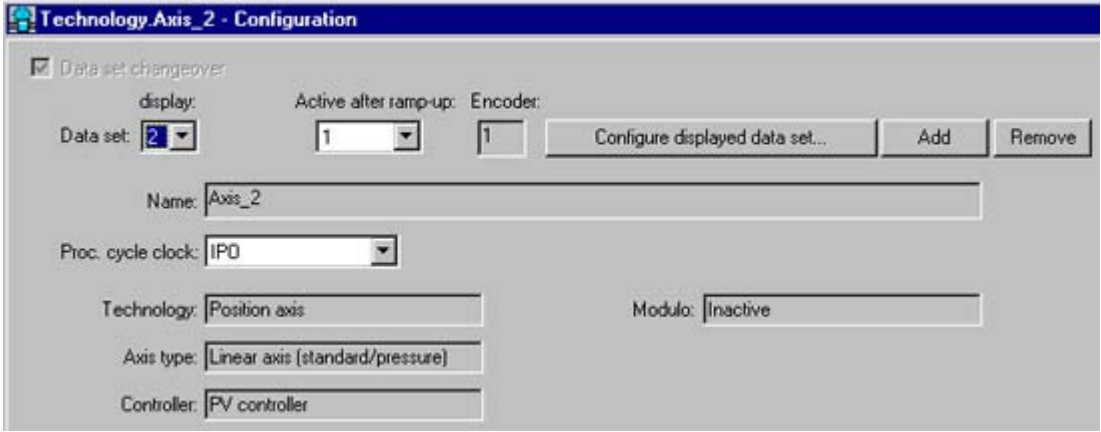
在 S7T Config 中组态数据记录时，一致性检查将报告所有非法的工艺参数设置。

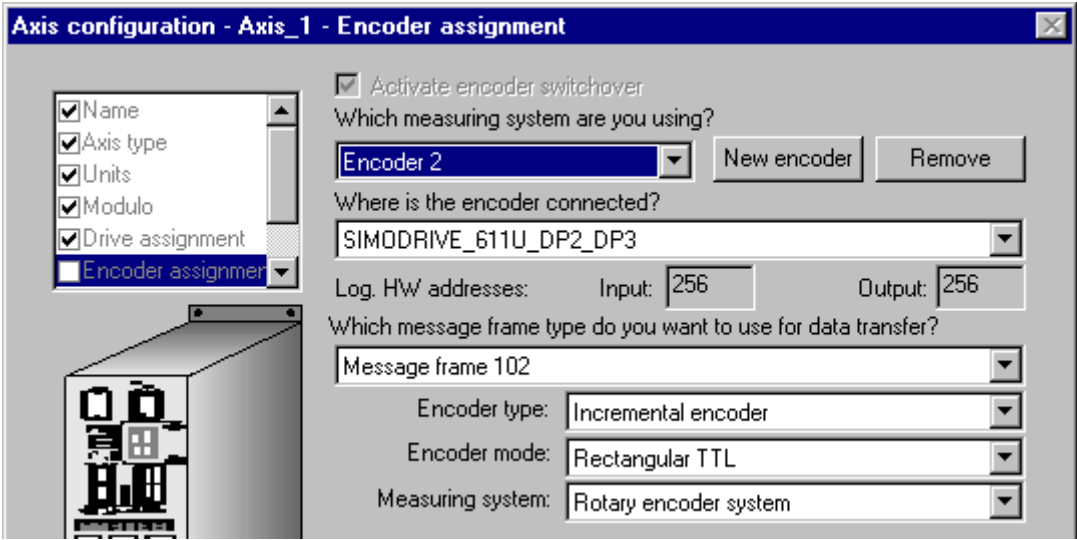
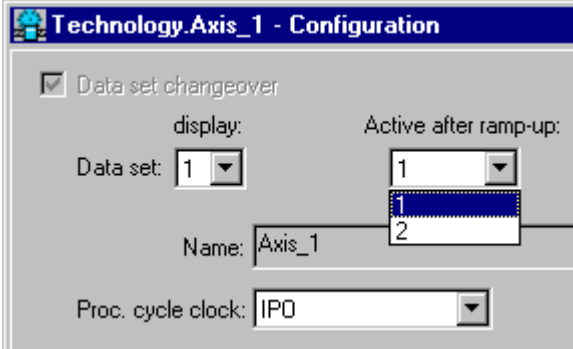
先决条件

- 轴已组态。

添加新数据记录

若要组态新数据记录：

步骤	说明
1.	在 S7T Config 的浏览器中，打开相应轴的数据记录添加组态对话框。
2.	设置“数据记录转换”(data record changeover) 复选框。 如果该轴已包含多个数据记录，则该复选框是隐藏的。
3.	单击“添加”(Add)。 结果： 新数据记录被添加到该轴。 
4.	单击“组态活动数据集”(Configure active data set)。 结果： 轴向导出现。
5.	使用该轴向导组态该数据记录。 注意： 一旦创建了轴，即设置了使用的工艺。不能使用另一个数据记录更改该轴工艺。
6.	也可以使用数据记录转换功能执行编码器转换。可通过设置“轴组态 — 编码器分配”(Axis configuration - Encoder assignment) 对话框上的“启用编码器转换”(enable encoder changeover) 复选框来实现该操作。 否则，请忽略该步骤并执行下一步骤。 单击“新建编码器”(new encoder) 创建第二个编码器。 结果： 打开一个下拉列表，您可以从中为该数据记录选择一个编码器。

步骤	说明
7.	为第二个编码器选择连接和消息帧类型。 在最后一步中，输入编码器信息（“编码器类型”[Encoder type]、“编码器模式”[Encoder mode] 和“测量系统”[Measuring system]）。 对于编码器组态，选择编码器文档中的设置，或编码器所连接的驱动组件的组态设置。  单击“下一步”(Next)。 继续编辑向导中的步骤，完成该组态。
8.	在轴组态对话框中，选择要在 CPU 启动后装载的数据记录（“启动后激活”）。  结果：“编码器”(Encoder) 表将显示轴的已组态编码器的总览。

删除某个轴的数据记录

通过从轴组态对话框中选择某个数据记录，开始删除该数据记录。 然后，单击“删除”(Delete)。 数据记录被删除。

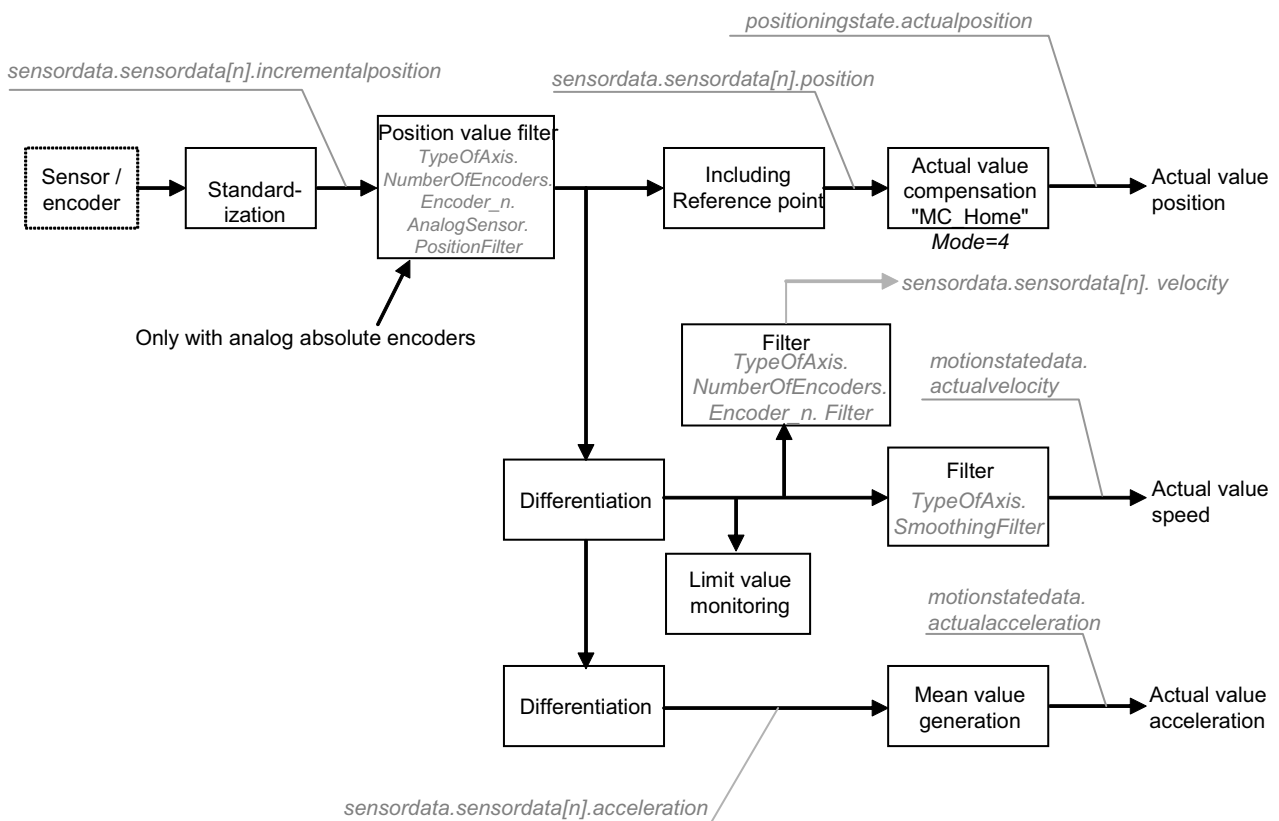
选择数据记录

从“显示数据记录”(Display data record) 下拉列表中选择要编辑的数据记录。在该轴的所有对话框中均可进行该选择。

从“启动后激活的数据记录”(Data record active after startup) 中选择启动后默认使用的数据记录。

4.6.7 实际值记录

下图显示了如何得到 T-CPU 中轴或外部编码器实际值的过程。



在位置控制周期中进行数据记录。这些数据可用于计算更多变量，如速度和加速度。

系统变量 *sensordata.sensordata[n].incrementalposition* 和 *sensordata.sensordata[n].position* 在位置控制周期时钟中计算。所有其它系统变量均在轴或外部编码器的相应执行周期时钟中进行计算。对于外部编码器，系统变量 *sensordata.sensordata[n]....* 名为 *sensordata...*。

可以使用 *TypeOfAxis.NumberOfEncoders.Encoder_n.AnalogSensor.PositionFilter* 组态数据元素激活或取消激活实际位置值。

该过滤器仅可用于模拟绝对编码器：

组态数据元素	含义	
<i>TypeOfAxis.NumberOfEncoders.Encoder_n.AnalogSensor.PositionFilter.enable</i>	<i>YES</i>	实际位置值已激活
	<i>NO</i>	实际位置值已取消激活
<i>TypeOfAxis.NumberOfEncoders.Encoder_n.AnalogSensor.PositionFilter.timeConstant</i>	用于 PT1 滤波的时间常量。	

可通过 *TypeOfAxis.NumberOfEncoders.Encoder_n.Filter* 组态数据元素激活/取消激活“传感器”速度过滤器：

组态数据元素	含义	
<i>TypeOfAxis.NumberOfEncoders.Encoder_n.Filter.enable</i>	<i>YES</i>	实际值过滤器已激活
	<i>NO</i>	实际值过滤器已取消激活
<i>TypeOfAxis.NumberOfEncoders.Encoder_n.Filter.timeConstant</i>	用于 PT1 滤波的时间常量。	

对于外部编码器，组态数据元素名为 *TypeOfAxis.Encoder_1.Filter...*

可通过 *TypeOfAxis.SmoothingFilter* 组态数据元素激活/取消激活实际速度值过滤器：

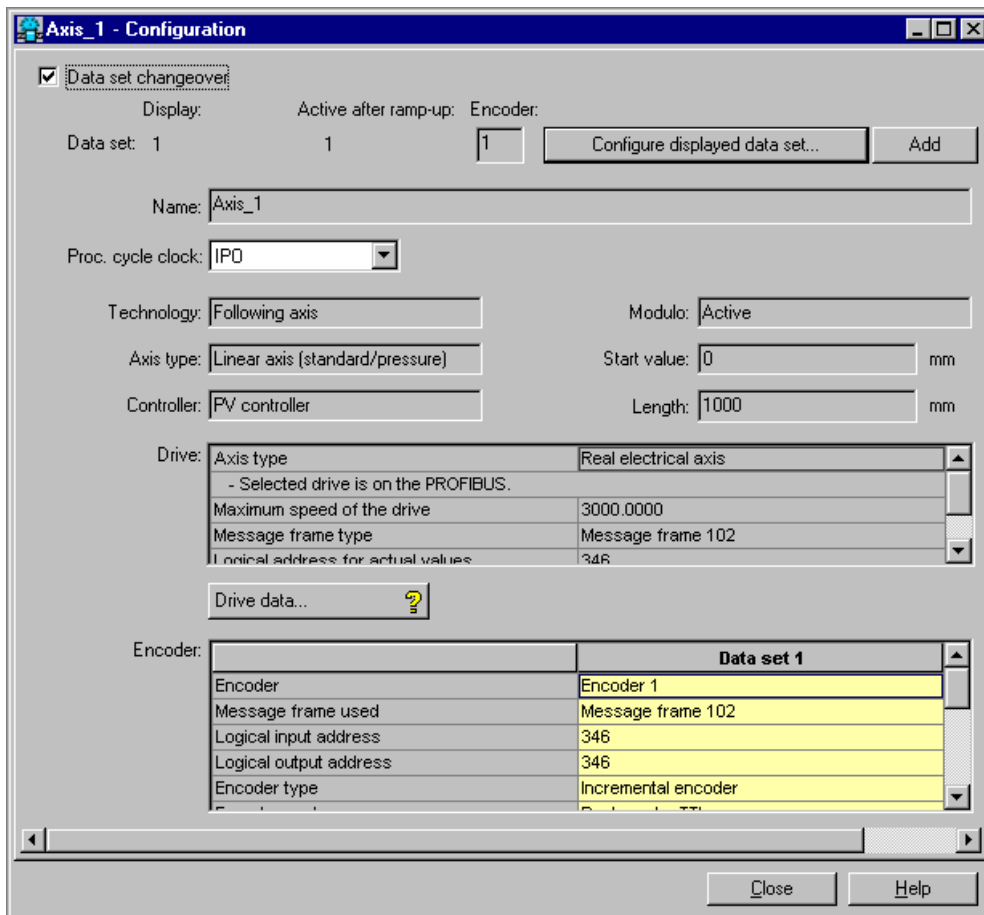
组态数据元素	含义	
<i>TypeOfAxis.SmoothingFilter.enable</i>	<i>YES</i>	实际值过滤器已激活
	<i>NO</i>	实际值过滤器已取消激活
<i>TypeOfAxis.SmoothingFilter.mode</i>	可以通过此组态参数设置过滤器计算方法：	
	<i>DEFAULT_MODE</i>	按比率函数求平均值： 轴/外部编码器的执行周期与位置控制周期之比例如： 设置执行周期与位置控制周期之比为 3:1 。在这种情况下，平均值是由三个位置控制器周期得出的。
	<i>AVERAGING</i>	使用时间常量求平均值
	<i>PT</i>	使用时间常量进行 PT1 平滑
<i>TypeOfAxis.SmoothingFilter.timeConstant</i>	用于 PT1 滤波或“ <i>AVERAGING</i> ”的时间常量。	

4.6.8 组态

4.6.8.1 组态 — 轴

轴 (Axis) > 组态 (Configuration) 对话框显示了轴和驱动器设置

可以通过单击“组态活动数据集”(Configure active data set) 在对话框中编辑这些设置。



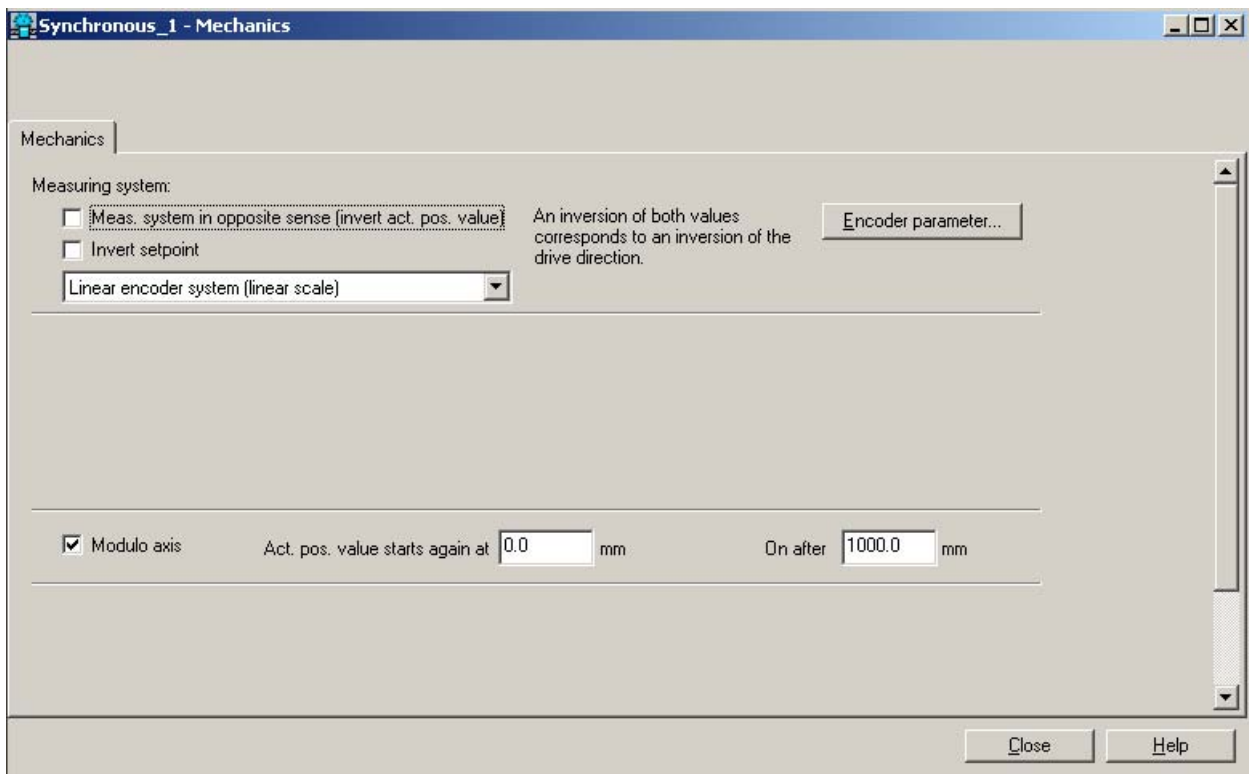
该对话框还包含用于管理数据记录转换的数据记录的功能。如果工艺对象支持多个数据记录，则您可以：

- 创建新数据记录
- 删除数据记录
- 定义在 CPU 启动期间为工艺对象装载哪个数据记录，以及
- 重新组态选定的数据记录。

4.6.9 机械系统

4.6.9.1 机械 — 液压轴

在轴 (Axis) > 机械 (Mechanics) 对话框中组态轴和编码器的机械参数。



轴的设置:

- 驱动器方向设置
- 线性轴的丝杠螺距
- 模数设置
- 间隙补偿设置

编码器的设置:

- 间隙补偿
- 编码器安装类型

4.6.10 默认值

4.6.10.1 缺省 —“动态”(Dynamics) 标签

可在**轴 (Axis) > 默认 (Default)** 对话框的“动态”(Dynamics) 选项卡中设置轴动态的默认值。如果在工艺功能中设置了负的动态值，则会激活默认值。

Dynamic response

Direction: Positive

Velocity: 100.0 mm/s

Velocity profile: Last programmed velocity profile

Jerk: 1000000.0 mm/s³

Acceleration: 1000.0 mm/s²

Jerk: 1000000.0 mm/s³

Deceleration: 1000.0 mm/s²

Jerk: 1000000.0 mm/s³

Stopping time: 0.0 s

支持的默认值:

- 方向 (Direction)
- 速度 (Velocity)
- 加速度 (Acceleration)
- 减速度 (Deceleration)
- 冲击 (Jerk)
- 速度曲线 (Velocity profile)

停止时间

禁用移动轴且在“MC_Power”将 *Stopmode* 设置为 2 时，“停止时间”(Stop time) 中设置的时间便会生效。

速度曲线 (Velocity profile)

速度曲线定义启动期间和制动时的响应，以及对速度变化的响应。

工艺功能不使用默认值。如果要更改速度曲线，请使用 *Jerk* 输入参数。

可以在以下曲线之间进行选择：

- **梯形**

梯形曲线用于正方向和负方向运动的线性加速度（突变化 = 0）。

- **常量**

该曲线显示常量加速度，突变化曲线是可控制的（突变化 $\neq 0$ ）。

4.6.11 限制

4.6.11.1 限制 —“位置和速度”(Position and velocity) 标签

在轴 (Axis) > 限制 (Limits) 选项卡的“位置和速度”(Position and velocity) 选项卡中，您可以

- 组态和启用对硬件限位开关的监视
- 组态和启用对软件限位开关的监视
- 设置速度限制
- 设置急停减速度

Position and velocity

Dynamic response

Fixed endstop

Hardware limit switch

☒ Active

Negative log. address

66

Bit number:

0



Positive log. address

66

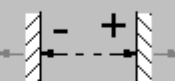
1

Software limit switch

☒ Active

-1000000000000.00

 mm



1000000000000.000

 mm

There are two different limits:

- Limits which must never be exceeded because of mechanical reasons.

- Limits which can be modified by programs.

Max. velocity

500.000000

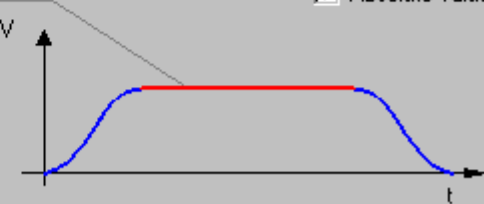
 mm/s

Pos. prog. velocity

> 1000000000000.

 mm/s

☒ Absolute values



Emergency stop delay:

10000.000000

 mm/s²

急停减速度

禁用移动轴且通过“MC_Power”参数将 *Stopmode* 设置为 0 时，将激活为急停减速度设置的值。

4.6.11.2 硬件限位开关

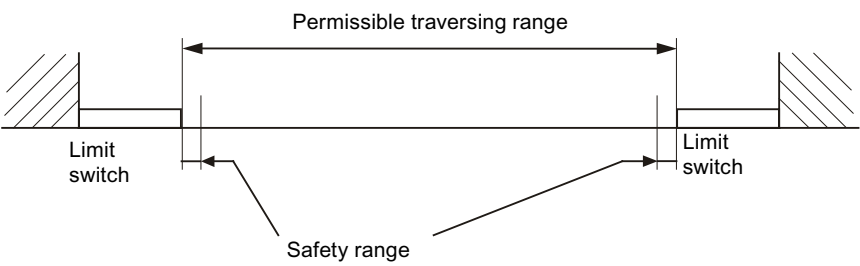
在轴 (Axis) > 限制 (Limits) 对话框的“位置和速度”(Position and Velocity) 选项卡中启用硬件限位开关监视。硬件限位开关监视用于限制轴的运动范围或保护机器。

接线

可将硬件限位开关连接至 T-CPU 的四个集成数字输入端，或者通过在 DP(DRIVE) 上操作的 TM15/TM17 连接至 I/O 模块（如 ET 200 或 SINAMICS S120）。

行程范围

通过硬件限位开关的数字输入监视允许的行程范围。



注意
硬件限位开关必须作为 NC 触点实施。 轴通过允许的行程范围后，硬件限位开关必须保持活动状态，直到达到机械终点位置。

退回

触发硬件限位开关的轴停止，并显示错误消息 **8013** 和 **804B**。该轴从硬件限位开关退回（释放运动），如下所述：

- 手动退回

手动使该轴返回允许的行程范围。只有该轴返回该范围后，**工艺 DB** 的错误才能被确认。

- 通过驱动器退回

该轴的 **工艺 DB** 的错误已被确认，但是错误消息和 *LimitSwitchActive* 位仍保持激活。现在，该轴可以返回到允许的行程范围。反转运动命令将再次触发轴错误。该轴移动到限位开关的范围之外后，便可确认错误消息和 *LimitSwitchActive* 状态。

保存到达硬件限位开关后，即可保存轴的当前位置。只有在通过此位置（加上一个安全范围）之后，该轴才被视为已离开限位开关。

注意
为避免硬件限位开关的极性监视和硬件限位开关的超程监视在有效范围方向上发生冲突，在该轴通过硬件限位开关后不能关闭控制器。在这种情况下，该轴将移至有效范围之内（无硬件限位开关监视），然后重新启用。 打开控制器时，该轴必须位于有效行程范围之内。

轴通过硬件限位开关时，内部状态会丢失并且将重新装载组态。只有在有效范围内重新装载时才不会丢失逼近信息。

例外情况： 发生极性反向错误后取消激活限位监视

安全范围

根据轴单位系统的已组态精度计算硬件限位开关的安全范围。

安全范围 = 1000/（增量/位置）

示例： 在“组态单位”(Configure units) 对话框中，将线性轴的位置单位指定为“mm”，将精度（增量/位置）指定为“1000/单位”，这意味着轴位置的计算精确到 0.001 mm。本示例中的安全范围是精度的 1000 倍： 1 mm。

4.6.11.3 软件限位开关

可以在**轴 (Axis) > 限制 (limits)** 对话框的“位置和速度”(Position and Velocity) 选项卡中组态软件限位开关并启用监视功能。如果激活软件限位开关，则通过软件限位开关限制轴的行程距离。

例如，软件限位开关应位于硬件限位开关的行程范围内，以限制轴的工作范围。

运动开始时监视软件限位开关：

- **复选框已激活**
如果位置控制运动命令的目标位置在软件限位开关之外，则运动开始时轴的工艺 DB 中就已显示警告 **0026**。在 **S7T Config** 中，报告中断 **40105**。
轴最多行进到软件限位开关位置，并在工艺 DB 中报告错误 **8014**。
- **复选框未激活**
如果位置控制运动命令的目标位置在软件限位开关之外，则轴来回移动，直到到达软件限位开关的 **p** 位置。在工艺 DB 中报告错误 **8014**（不输出警告 **0026**）。

行进到软件限位开关的特性：

- **对于位置受控的来回移动：**
轴来回移动，直到到达软件限位开关。到达软件限位开关后，保持各自活动的位置控制模式或速度控制模式
- **在所有操作模式中：**
轴来回移动，直到到达软件限位开关。到达软件限位开关后，保持活动的位置控制模式。在速度控制模式中，轴变为位置控制模式。

负位置/正终点位置：

在这些输入域中输入软件限位开关的负位置和正终点位置。

用于退回的公差窗口：

在此输入域中输入适合的值，以防止退回期间重新开始触发软件限位开关错误。

说明

软件限位开关的响应由轴组态根据回原点功能确定。如果轴尚未回原点时，则不监视组态中要求使用已回原点轴来执行绝对运动命令（轴 **[Axis] > 回原点 [homing]** 对话框中的“需要回原点: 是”**[Homing required: Yes]**）的限位开关。反之，如果在轴未回原点时允许执行绝对运动命令（轴 **[Axis] > 回原点 [homing]** 对话框中的“需要回原点: 否”**[Homing required: No]**），则无论轴的回原点状态如何，都将监视软件限位开关。

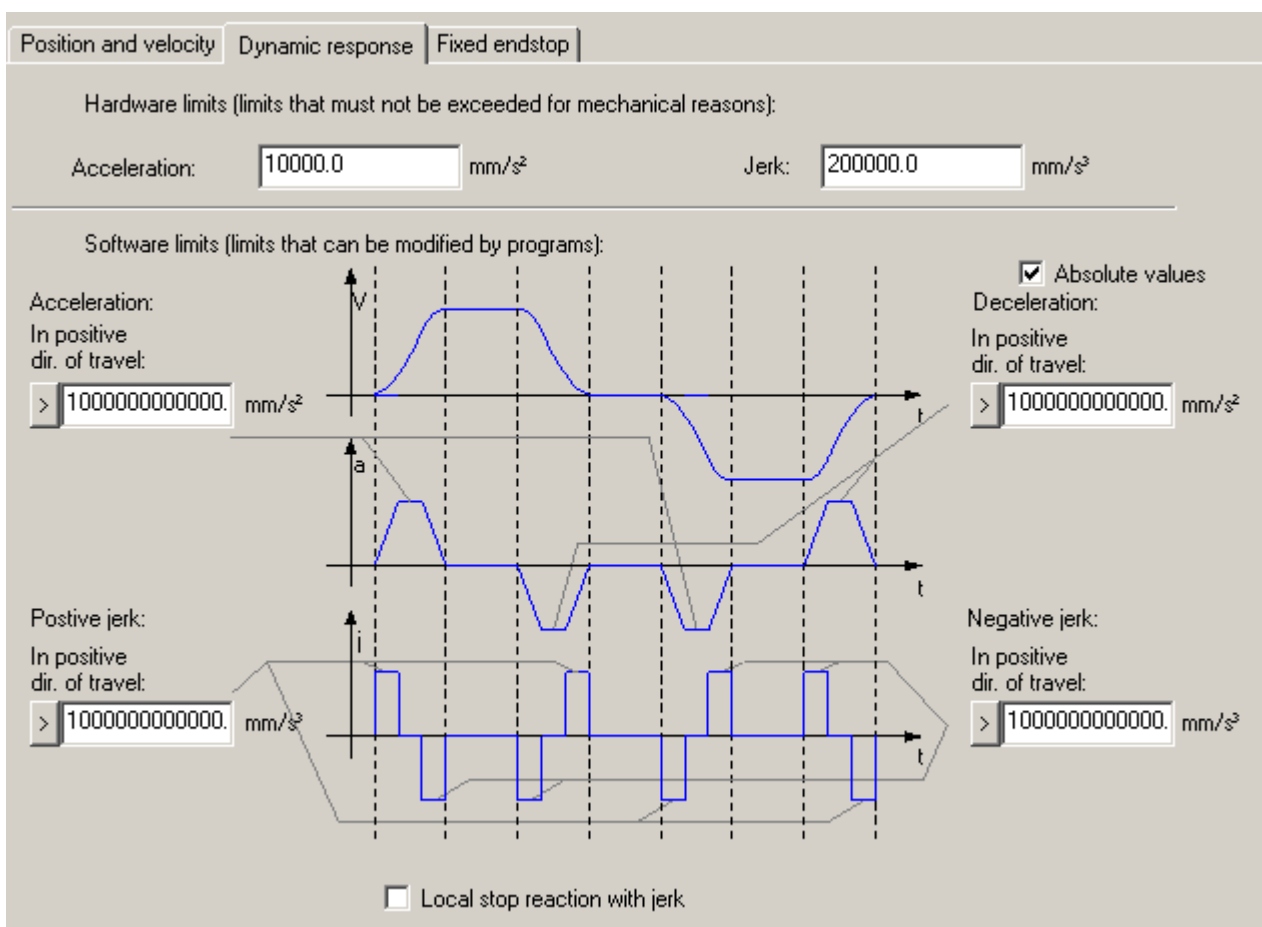
4.6.11.4 限制 —“动态响应”(Dynamic response) 标签

可以在轴 (Axis) > 限制 (Limits) 对话框的“动态响应”(Dynamic response) 选项卡中调整 HW 限制（机械）和 SW 限制。

禁用移动轴且在“MC_Power”参数将 *FastStop* 设置为 1 时，激活设定的减速度限制。

默认情况下，可以通过设置“绝对值”(Absolute values) 复选框启用取决于状态的加速度模型。

可以通过重设“绝对值”(Absolute values) 复选框启用取决于方向的加速度模型。在这种情况下会显示其他一些附加参数。



取决于状态的加速度

- **加速度 (Acceleration)**
轴加速度，与运动方向无关
- **减速度 (Deceleration)**
轴减速度，与运动方向无关

取决于方向的加速度

- **加速度 (Acceleration)**
正运动方向的加速度和负运动方向的减速度
 - **减速度 (Deceleration)**
负运动方向的加速度和正运动方向的减速度
- 例如，动态方向矢量的参数设置对于悬挂轴很有用。

带突变化的局部停止响应

设置“带突变化的局部停止响应”(Local stop reactions with jerk) 复选框，以执行由轴上的报警响应触发的停止操作，带有突变化限制和倒圆功能。

对于预编程减速度停止（仅当使用真实轴时输入框才可见）

在此输入框中输入斜坡设置的减速度。

用于平滑由控制器转换产生的受控变量变化的时间常量 （仅当使用真实轴时输入框才可见）

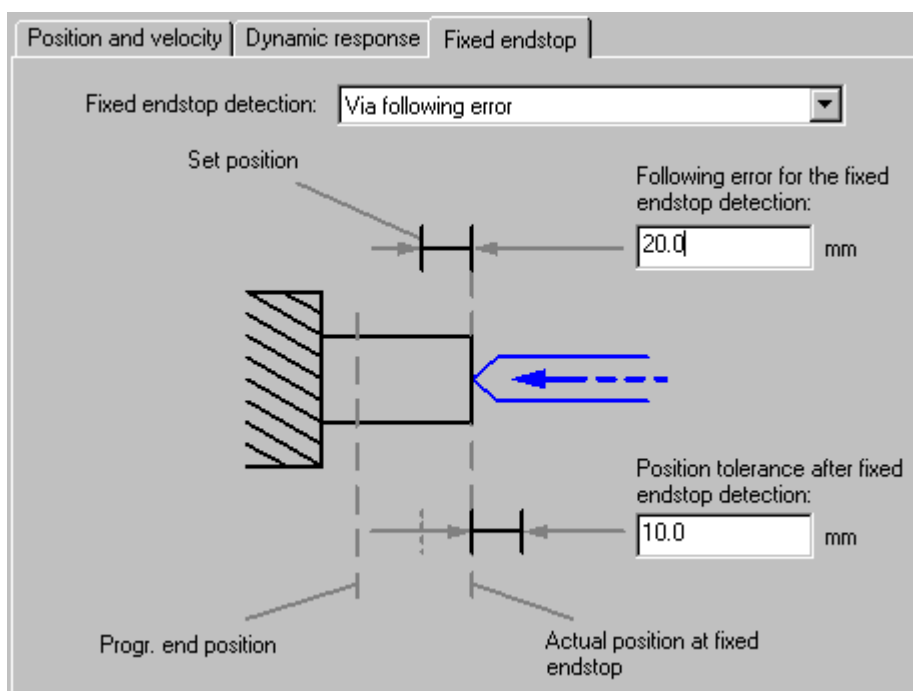
在此处输入用于平滑由控制器转换产生的受控变量变化的时间常量。对于会由于转换引起受控变量偏移的所有状态跳转/转换，此转换平滑滤波器都处于活动状态。

4.6.11.5 限制 —“固定挡块”(Fixed end stop) 标签

轴 (Axis) > 限制 (Limits) 的“固定终点挡板”(Fixed end stop) 选项卡可用于启用固定终点挡板检测以及设置相应的检测模式：

- 使用跟随误差
- 使用力/扭矩

满足选定的条件时，将达到“固定终点挡板”(fixed end stop) 状态。如果启用了“移至固定终点挡块”(Move to fixed end stop)，将禁用跟随误差监视。



使用跟随误差

说明

当轴移至固定终点挡块且将固定终点挡板检测设置为“使用跟随误差”(use following error) 时，在“检测到固定终点挡板之后的位置公差”(Position tolerance after fixed end stop detection) 中输入的值应明显小于“固定终点挡板检测的跟随误差”(Following error for fixed end stop detection) 中的值。

使用扭矩值

固定终点挡板检测功能“使用扭矩值”(use torque value) 要求分配给轴的数字驱动器支持扭矩限制，并且需要设置相应的消息帧，例如消息帧 102 或消息帧 105。

移至固定终点挡块

“MC_MoveToEndPos”功能可以在达到终点挡板后激活“移至终点挡板”(Move to end stop) 功能并设置紧固扭矩。该操作也称为“夹紧”。

运动将在轴到达固定终点挡板时停止，而控制仍然保持活动状态。位置控制器输入的设定值保持为常量。钳位位置方向上的新运动命令将被取消；将在退回方向上执行新运动控制命令以减少扭矩。该轴的位置设定值用作新运动控制命令在退回方向上的起始位置。

轴位置设定值由以下其中一个等式产生，具体取决于固定终点挡板检测功能：

- “使用跟随误差”
在固定终点挡板的位置 + 跟随误差
- “使用扭矩值”
在固定终点挡板的位置 + 夹紧公差

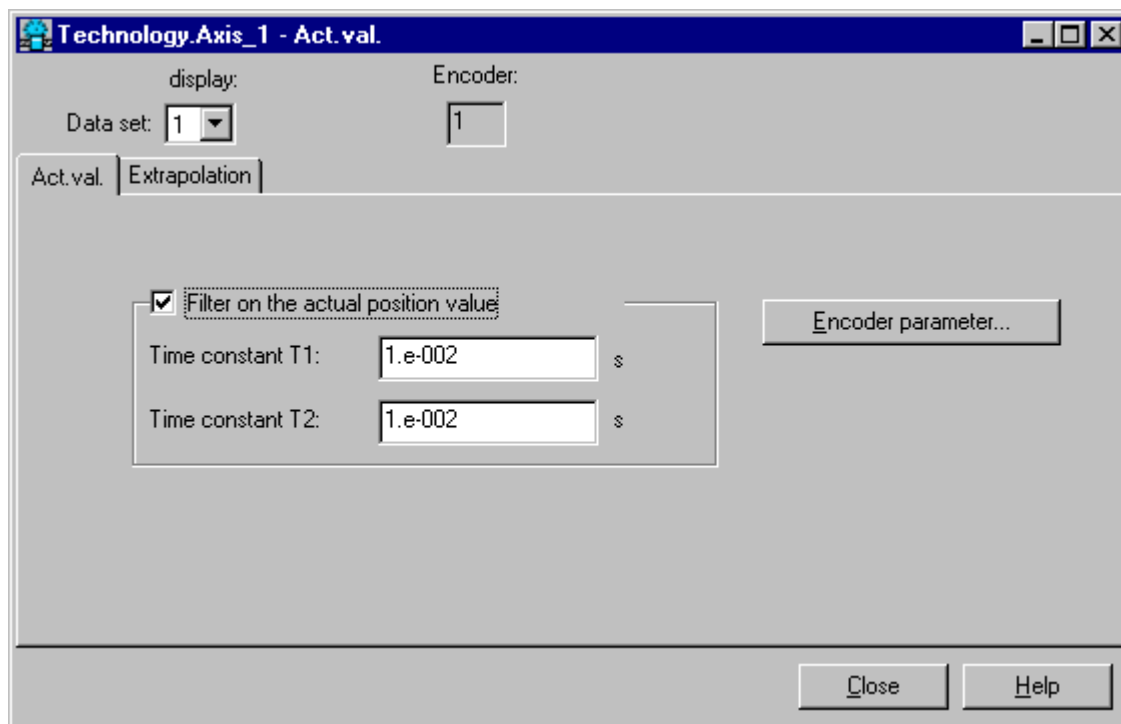
“固定终点挡板检测”功能的条件

- 当轴移至钳位公差窗口外时，将复位“移至固定终点挡块”(Move to fixed end stop) 功能。
- 还可以输出新命令，以在钳位处于活动状态时切换扭矩的方向。
- 可在用户程序中，在已定义的时间段上执行非步进扭矩跳转和扭矩保持，就像扭矩曲线的定义一样。
- 可以通过设置反向定位命令禁用移至固定终点挡块（钳位）。
- 不允许反向命令 MC_MoveToEndPos，将忽略该命令。
- 可以通过轴的实际值监视终点挡板的机械制动（钳位公差窗口监视）。
- 通过 MC_MoveToEndPos 工艺功能的 *Torque* 参数，以 [N/m] 为单位设置驱动器的扭矩限制。
- 如果该命令处于活动状态且未检测到固定终点挡板，则系统会像对待活动的扭矩限制那样作出反应。

4.6.12 实际值

4.6.12.1 实际值 (Actual value) —“实际值”(Actual value) 选项卡

在“轴/外部编码器”(Axis/External encoder) > “实际值”(Actual value) 对话框中的“实际值”(Actual value) 选项卡上激活实际值过滤，并设置相应的时间常量。



过滤实际位置值 (Filter on the actual position value)

如果想要激活实际位置值过滤，则激活该复选框。

时间常量 T1 (Time constant T1)

此处设置实际值系统中 PT2 位置过滤器的时间常量 T1。

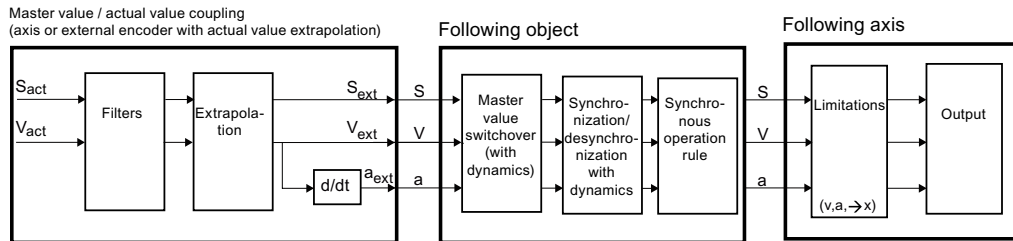
时间常量 T2 (Time constant T2)

此处设置实际值系统中 PT2 位置过滤器的时间常量 T2。

“编码器参数”(Encoder parameter) 按钮

打开一个对话框，其中显示编码器数据。

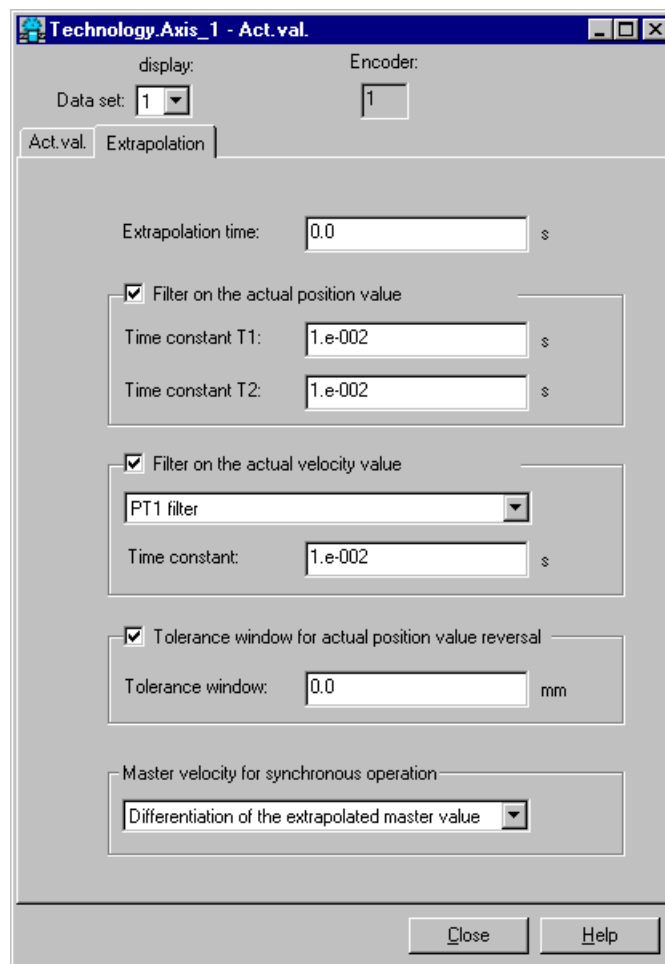
4.6.12.2 实际值 (Actual value) —“推断”(Extrapolation) 选项卡



如果控制内有同步操作互连，则同步操作会考虑主值位置的位置、速度和加速度。

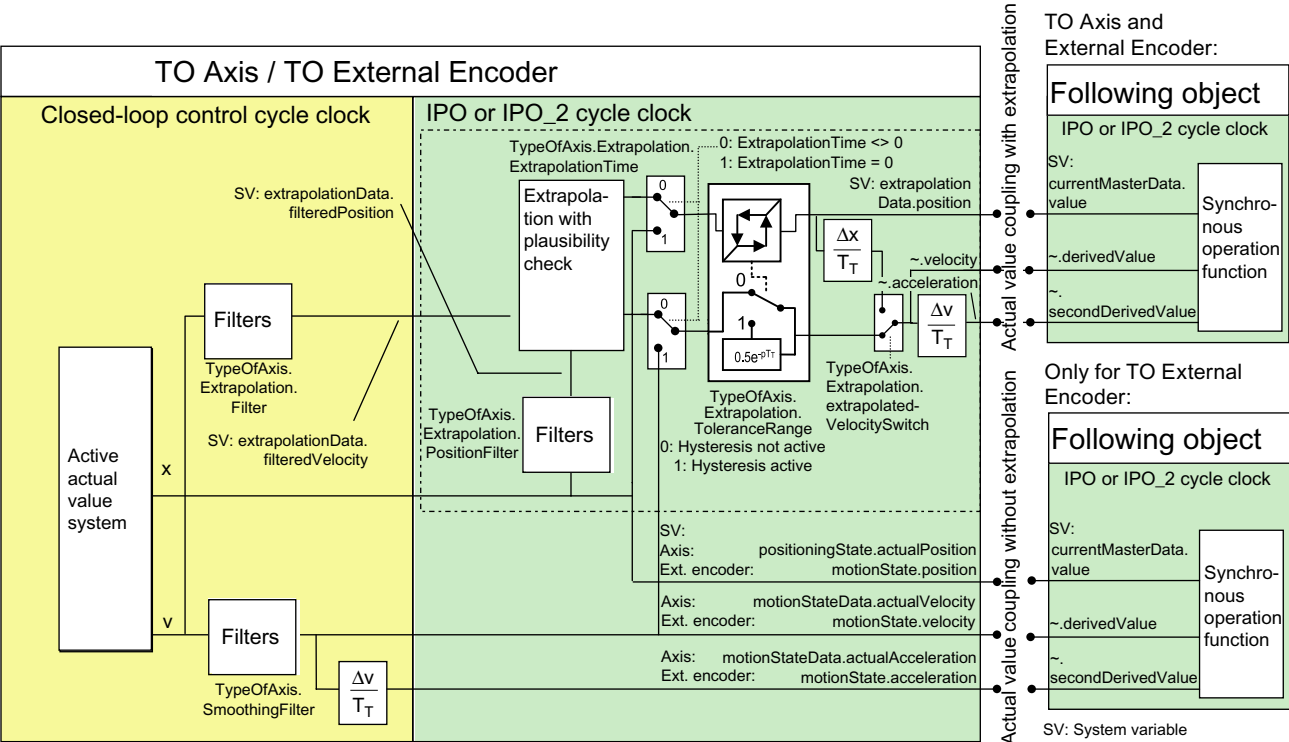
如果实际编码器值用作主值，则推断同步操作的测量实际值以补偿死区时间是很用的。死区时间是在测量实际值时在系统内产生，例如，由于总线通信和系统处理时间产生。

在 S7T Config 中，在引导轴或外部编码器上，在“轴/外部编码器”(Axis/External encoder) >“实际值”(Actual value) 对话框的“推断”(Extrapolation) 选项卡上设置推断。



也可以通过工艺参数从用户程序读取此对话框中的参数，或将此对话框中的参数写入用户程序。

实际值结合推断（轴和外部编码器）



推断时间 (Extrapolation time)

(参数 1110; 组态数据 *TypeOfAxis.Extrapolation.ExtrapolationTime*)

此处设置推断的时间。 如果输入的值为 0.0, 则无推断。

过滤实际位置值 (Filter on the actual position value)

(参数 1130 组态数据 *TypeOfAxis.Extrapolation.ExtrapolationPositionFilter.enable*)

如果想要推断实际位置值, 则激活该复选框。

时间常量 T1 (Time constant T1)

(参数 1131 组态数据 *TypeOfAxis.Extrapolation.ExtrapolationPositionFilter.T1*)

此处设置实际值系统中用于推断的 PT2 过滤器的时间常量 T1。

时间常量 T2 (Time constant T2)

(参数 1132 组态数据 *TypeOfAxis.Extrapolation.ExtrapolationPositionFilter.T2*)

此处设置实际值系统中 PT2 位置过滤器的时间常量 T2。

该过滤器作用于推断的实际位置。推断的速度取自应用平滑滤波器 (*TypeOfAxis.smoothingFilter*) 之前轴或外部编码器的实际值。

过滤实际速度值 (Filter on the actual velocity value)

(参数 1112 组态数据 *TypeOfAxis.Extrapolation.Filter.enable*)

如果想要推断实际速度值，则激活该复选框。

在此处从下拉列表中选择推断速度的过滤器。

(参数 1111 组态数据 *TypeOfAxis.Extrapolation.Filter.Mode*)

时间常量 (Time constant)

(参数 1113 组态数据 *TypeOfAxis.Extrapolation.Filter.timeConstant*)

此处输入过滤器的时间常量。

根据过滤的速度值或平均速度值推断位置。通过“时间常量”计算平均值。

实际位置值倒置公差窗口 (Tolerance window for actual position value inversion)

(参数 1114 组态数据 *TypeOfAxis.Extrapolation.ToleranceRange.enable*)

在此处可以激活用于实际位置值倒置的公差窗口

公差窗口 (Tolerance window)

(参数 1115 组态数据 *TypeOfAxis.Extrapolation.ToleranceRange.Value*)

在此处输入公差窗口的大小。

如果主值与同步操作无法跟随的高频噪声信号叠加，则会造成超出动态响应限制或同步期间主值暂时改变方向。

在这种情况下，可定义公差窗口，以防止在跟随轴上超出动态响应限制，或防止同步期间方向改变。

同步操作的主速度 (Master velocity for synchronous operation)

(参数 1116 组态数据 *TypeOfAxis.Extrapolation.extrapolatedVelocitySwitch*)

在下拉列表中，选择该速度是否应用于推断，或是否对推断的主位置值求微分。

检查推断值和过滤值 (Checking the extrapolated and filtered values)

可在以下系统变量中检查推断值和过滤值：

- *extrapolationdata.position*
- *extrapolationdata.velocity*
- *extrapolationdata.filteredposition*
- *extrapolationdata.filteredvelocity*
- *extrapolationdata.acceleration*

支持具有 NIST 评估的编码器 (Support of encoders with NIST evaluation)

使用具有 NIST 评估的编码器，速度由编码器确定，所得的速度可被编码器接受。在这种情况下，不需要由集成工艺计算速度和速率。有两种传送方法可用：

- 在 PROFIdrive 消息帧中传送

通过组态数据元素 *TypeOfAxis.NumberOfEncoders.Encoder_n.EncoderValueType = POSITION_AND_PROFIDRIVE_NIST_* 设置

- 在 I/O 区域中传送

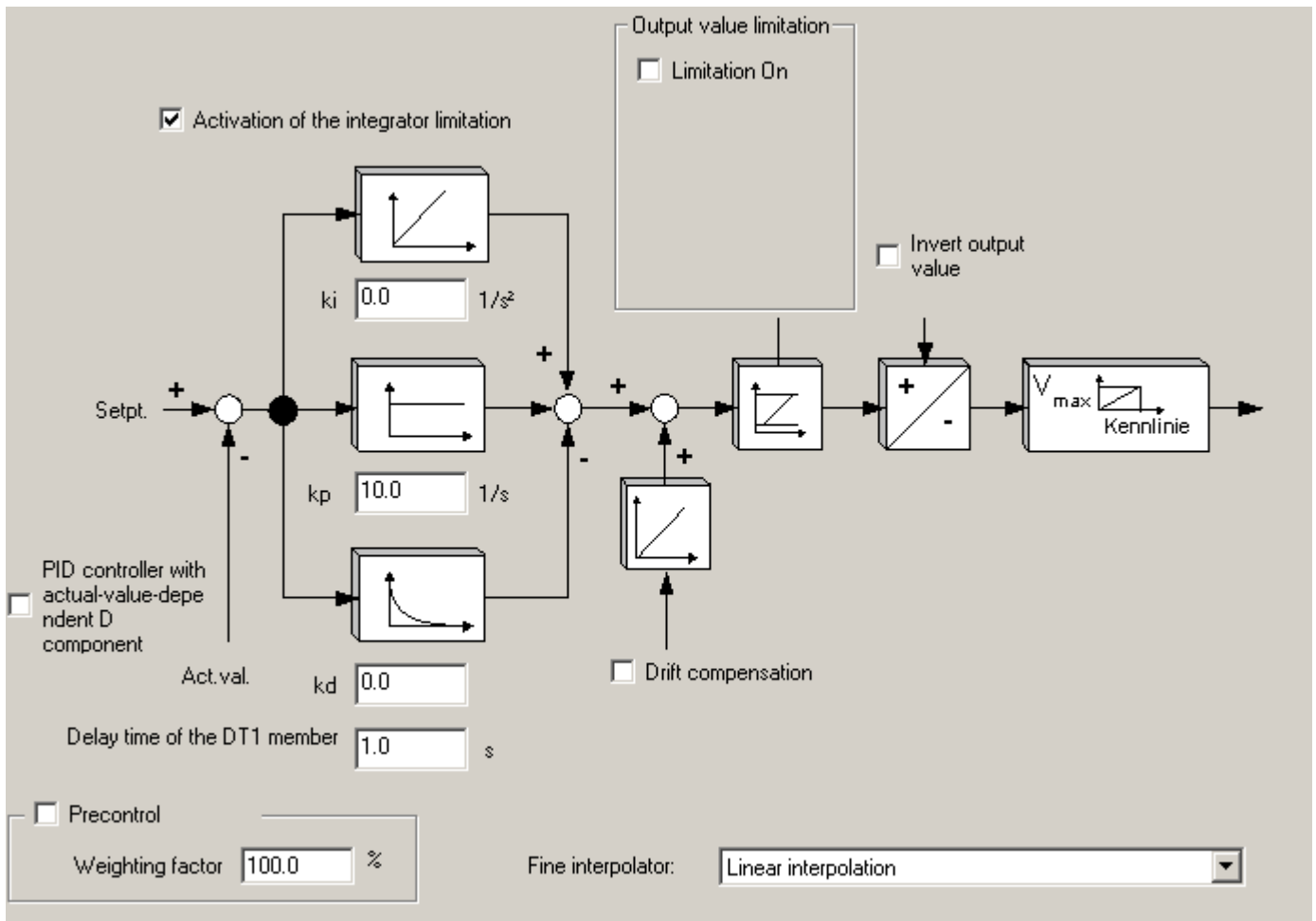
通过以下组态数据元素设置：

TypeOfAxis.NumberOfEncoders.Encoder_n.EncoderValueType = POSITION_AND_DIRECT_NIST。在这种情况下，4000H 对应于 100%。地址在组态数据元素 *TypeOfAxis.NumberOfEncoders.Encoder_n.nistConfig.logAdress* 中设置，参考值在组态数据元素 *TypeOfAxis.NumberOfEncoders.Encoder_n.nistConfig.referenceValue* 中设置。

4.6.13 控制

4.6.13.1 控制 —“静态控制器数据”(Static controller data) 标签

在轴 (Axis) > 控制 (Control) 对话框的“静态控制器数据”(Static controller data) 选项卡中设置轴的位置控制。



受控变量限制

受控变量限制用于设置控制范围的绝对上限和下限。该限制在反转之前应用。

高精度插补器

高精度插补器功能用于在插补器和控制器具有不同的占空比时生成中间设定值。高精度插补器上的可选插补模式设置：

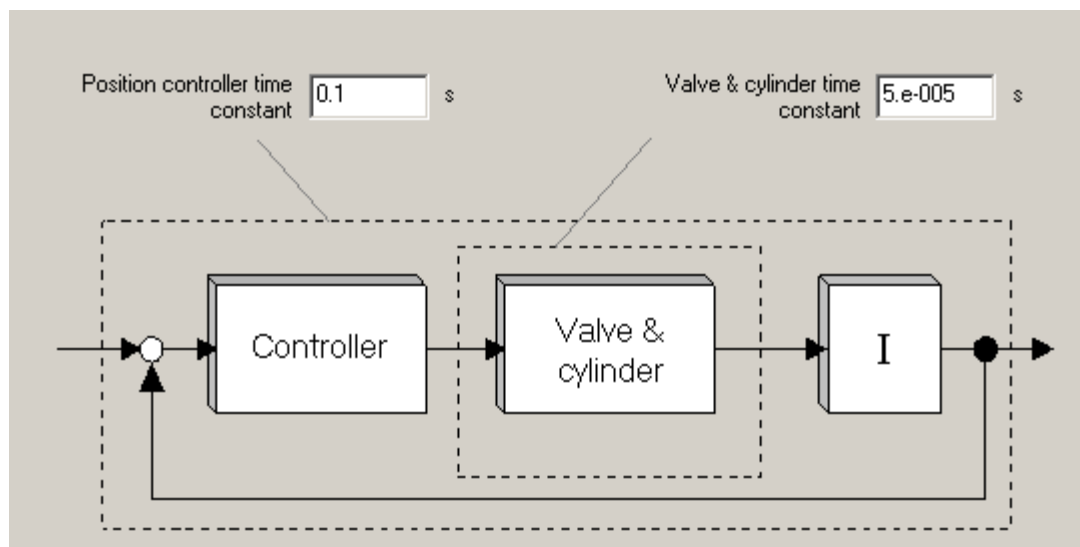
- 无插补
- 线性插补（定位轴的位置恒定）
- 以恒定加速度（定位轴上加速度恒定）插补
- 以恒定速度（定位轴上速度恒定）插补

当设置为定位轴时，将插补位置设定值。

当设置为速度控制轴时，将插补速度设定值。

4.6.13.2 控制 —“动态控制器数据”(Dynamic controller data) 标签

在“轴”(Axis) > “控制”(Control) 对话框的“动态控制器数据”(Dynamic controller data) 选项卡中设置轴的动态控制器数据。



位置控制器时间常量 (Position controller time constant)

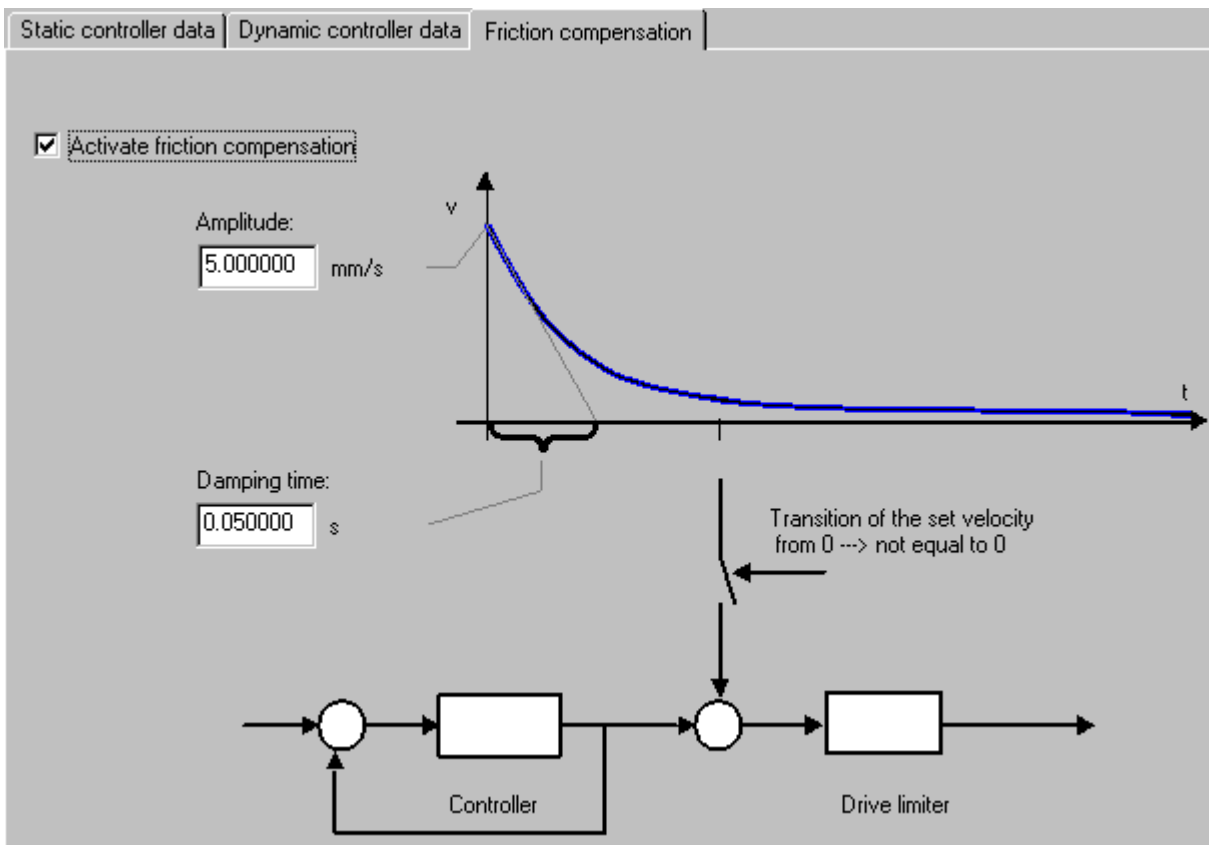
为轴的位置控制器输入相等的时间常量。

阀和气缸时间常量(Valve & cylinder time constant):

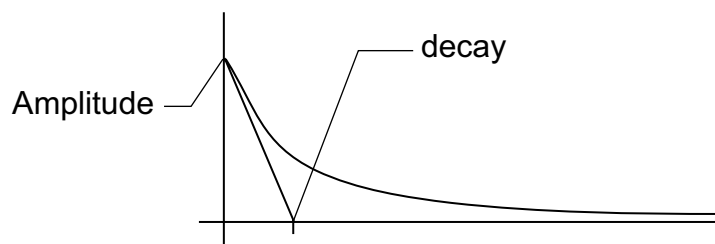
为此参数的液压控制回路输入相等的时间常量。

4.6.13.3 控制 —“摩擦补偿”(Friction compensation) 标签

轴 (Axis) > 控制 (Control) 选项卡的“摩擦补偿”(Friction compensation) 选项卡用于启用摩擦补偿。在“静态控制器数据”(Static controller data) 选项卡中设置了专家模式后,“摩擦补偿”(Friction compensation) 选项卡便可用。



系统提供了一种简单的方法来补偿静态摩擦力。从停止开始启动时, DT1 单元在受控变量上添加一个静态摩擦补偿信号。

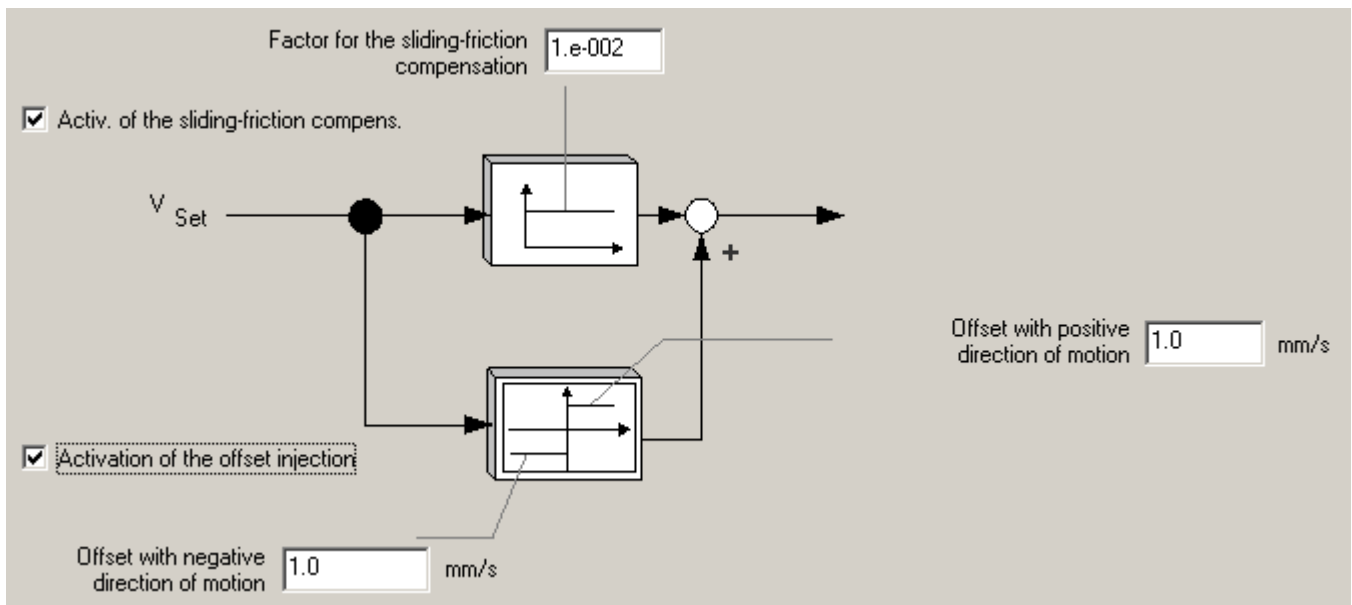


摩擦补偿是相对于速度设定值而添加的。只有执行运动命令时摩擦补偿才会处于活动状态。

可以单独为静态摩擦补偿设置停止状态标识、振幅和衰减响应。可以在组态中设置振幅和衰减响应。

4.6.13.4 控制 —“附加补偿功能”(Additional compensation functions) 标签

选择**轴 (Axis) > 控制 (Control)** 对话框的“附加补偿功能”(Additional compensation functions) 选项卡，以组态静态补偿元素（附加摩擦系数）和一个与液压轴的位置控制器的速度（摩擦）成比例的补偿元素。



- **摩擦补偿因子：**（摩擦补偿已激活）
在此参数中输入摩擦补偿因子。
- **激活摩擦补偿：**
设置此框以激活摩擦补偿。
- **正方向移动的偏移量：**（偏移量已激活）
输入正方向移动的偏移量。
- **激活偏移量：**
选中此框以激活偏移量。
- **负方向移动的偏移量：**（偏移量已激活）
输入负方向移动的偏移量。

4.6.14 回原点

4.6.14.1 引言

每次接通电源时，装有增量电机测量系统的位置控制轴必须参考机器的机械系统的位置。根据轴机制已定义位置的激活特定位置值，在回原点期间同步轴。

轴可以在以下模式下回原点：

- 主动模式（参考点逼近）
- 被动模式（快速参考）
- 直接模式（使用位置设定值）

详细设置取决于参考点的测量系统以及轴的回原点运动。

4.6.14.2 回原点 —“主动回原点”(Active homing) 标签

主动回原点

可以在 S7T Config 中的轴 (Axis) > 回原点 (Homing) 对话框的“主动回原点”(Active homing) 选项卡中组态主动回原点。

“主动回原点”(Active homing) 有三种可用的回原点模式：

- 参考凸轮和编码器零标记
- 仅编码器零标记
- 仅外部零标记

启用回原点功能：

MC_Home	Mode = 0	根据轴组态确定参考点
	Mode = 1 Position = x	根据轴组态确定参考点。给参数点分配的值是 Position 输入参数。

成功回原点后，轴工艺 DB 将返回状态 *Statusword.HomingDone* = TRUE。

Homing required: Yes

Homing mode: Homing output cam and encoder zero mark

Homing with Encoder zero mark: In front of homing output cam

Log. address of homing output cam: 65535

Start of homing procedure: Positive direction Bit number:

Approach velocity: 50.0 mm/s

Entry velocity: 20.0 mm/s

Reduced velocity: 10.0 mm/s

☒ Max. distance to homing output cam: 0.0 mm

☐ Max. dist. to encoder zero mark: 0.0 mm

Home position coordinate: 0.0 mm Home position offset: 0.0 mm

各个参数的含义:

参数	值	说明
需要回原点	是	轴必须回原点才能执行绝对运动命令。
	否	轴不必回原点就可以执行绝对运动命令。
回原点模式	-	在这种情况下，“回原点输出凸轮和编码器零标记”
编码器零标记	参考凸轮之前	轴以参考点逼近方向返回编码器零标记，该标记位于参考凸轮前面。
	参考凸轮之后	轴以参考点逼近方向返回编码器零标记，该标记位于参考凸轮后面。
开始参考点逼近	正方向	正方向上的参考点逼近。
	负方向	负方向上的参考点逼近。

参数	值	说明
参考凸轮的逻辑地址	[字节地址]	参考凸轮的逻辑字节地址 可以将参考凸轮连接至 T-CPU 的集成输入或连接至 DP(DRIVE) 上的 IO。
位号	[位号]	用于参考凸轮的信号的位地址
逼近速度	-	轴逼近参考凸轮的速度
进入速度	-	检测到编码器零标记后，轴逼近（移位后的）参考点的速度
关闭速度	-	检测到参考凸轮后，轴逼近编码器零标记的速度。
参考点坐标 (参考凸轮和编码器零标记、仅外部零标记、仅编码器零标记)	-	在此处输入原点位置的实际位置值（相对于轴的坐标系）
回原点位置偏移	-	“原点位置偏移”(Home position offset) 将参考点移位所组态的距离。轴按照在“进入速度”(Entry velocity) 中定义的速度移动该组态距离，起点为具有编码器零标记的同步位置。 模数轴始终取最短距离。
到回原点输出凸轮的最大距离	取消激活	没有监视到参考凸轮检测的距离
	激活	监视参考点逼近的起点和参考凸轮检测之间的距离。如果该距离的差值超出了组态的值，则相应的工艺 DB 将返回错误 801D。参考点逼近取消。
到编码器零标记的最大距离	取消激活	没有监视到编码器零标记所需要移动的距离
	激活	监视轴在参考凸轮和编码器零标记之间移动的距离。如果此距离的差值超过组态的距离，则相应的轴工艺 DB 会指示错误 801D。参考点逼近取消。

参考点逼近的顺序

● 阶段 1： 参考凸轮逼近

轴以组态的逼近速度，沿在“参考点逼近的起点”(Start of reference point approach) 中设置的方向开始其参考点逼近。到达参考凸轮（阶段 1）后，参考凸轮逼近结束。

也可以通过设置“到回原点输出凸轮的最大距离”(Max. distance to homing output cam) 复选框，来监视轴从参考点逼近的起点到检测出参考凸轮时所移动的距离。如果在已组态距离的起点和终点之间没有检测到参考凸轮，则参考点逼近将因出错而取消。

● 阶段 2： 与编码器零标记同步

到达参考凸轮后，轴加速/减速到关闭速度，并逼近编码器零标记。编码器零标记从“编码器零标记”(Encoder zero mark)（参考凸轮之后或之前）和“参考点逼近的起点”(Start of reference point approach)（正方向或负方向）的组合设置得出。

检测到参考凸轮后，PLC 会根据组态将轴与检测到的第一个编码器零标记同步。将轴位置设置为默认值减去在“参考点坐标”(Reference point coordinate)(*Mode = 0*) 中或在输入参数 *Position (Mode = 1)* 中定义的参考点位移。

也可以通过设置“到编码器零标记的最大距离”(Max. distance to encoder zero mark) 复选框，来监视轴在参考凸轮和编码器零标记之间移动的距离。如果在检测到参考凸轮后，在指定的距离中未发现编码器零标记，则参考点逼近将因出错而取消。

● 阶段 3： 参考点逼近

检测到编码器零标记后，轴将加速/减速到逼近速度，以逼近参考点。

如果已组态了参考点位移，则轴以相应的距离（起点为同步位置）逼近该参考点。运动方向由参考点位移的符号以及检测到编码器零标记后的减速斜坡长度确定（前提是参考点位于减速斜坡内）。

使用“仅外部零标记”(External zero mark only) 的主动回原点模式

Homing required: Yes

Homing mode: External zero mark only

Homing with

Signal transition: Low -> High (positive)

On the external zero mark side: Right

Start of homing procedure: Positive direction

Approach velocity: 50.0 mm/s

Entry velocity: 20.0 mm/s

Reduced velocity: 10.0 mm/s

Max. distance to ext. zero mark: 0.0 mm

Home position coordinate: 0.0 mm

Home position offset: 0.0 mm

如果通过调用“MC_Home”工艺功能启动回原点过程，则轴将根据组态逼近参考外部零标记。该图的下半部分显示了最终的运动顺序。

各个参数的含义：

参数	值	说明
需要回原点	是	轴必须回原点才能执行绝对运动命令。
	否	轴不必回原点就可以执行绝对运动命令。
回原点模式	-	在此例中：“仅外部零标记” (External zero mark only)
信号跳转	低 -> 高（正）	运动参考外部零标记的上升沿（根据驱动组件中的沿评估进行设置）。
	高 -> 低（负）	运动参考外部零标记的下降沿（根据驱动组件中的沿评估进行设置）。

参数	值	说明
在外部零标记一侧	左	沿选定的参考点逼近的方向，从外部零标记的左侧进行观测，测量信号跳转。
	右	沿选定的参考点逼近的方向，从外部零标记的右侧观测，测量信号跳转。
开始参考点逼近	正方向	正方向上的参考点逼近。
	负方向	负方向上的参考点逼近。
逼近速度	-	轴逼近外部零标记的速度
进入速度	-	检测到外部零标记后，轴逼近（移位后的）回原点位置的速度。
关闭速度	-	检测到外部零标记后，轴逼近参考点坐标的速度
回原点位置偏移	-	“原点位置偏移”(Home position offset) 将参考点移位所组态的距离。轴在外部零标记沿处同步后，以“回原点速度”(Homing velocity) 移动所组态的距离。模数轴始终取最短距离。
到外部零标记的最大距离	取消激活	没有监视到外部零标记检测的距离。
	激活	监视从参考点逼近的起点到检测到外部零标记所移动的距离 如果该距离的差值超出了组态的值，则相应的工艺 DB 将返回错误 801D。参考点逼近取消。

说明

对于使用“外部零标记”(external zero mark) 的回原点，将外部零标记作为数字测量输入连接至驱动组件。

为了以“外部零标记”(external zero mark) 回原点模式执行参考点逼近，请选择与驱动器的组态或功能相对应的“信号跳转”(Signal transition) 和“在外部零标记一侧”(on the side of the external zero mark) 的值。“信号跳转”(Signal transition) 和“在外部零标记一侧”(on external zero mark side) 参数仅反映其功能，以根据组态控制轴运动，而不是用来组态驱动器中的测量功能。

有关外部零标记检测的组态的信息，请参考相关的产品信息或驱动器手册。

参考点逼近的顺序

- 阶段 1：与外部零标记同步

轴以组态的逼近速度，沿在“参考点逼近的起点”(Start of reference point approach) 中设置的方向开始其参考点逼近。在外部零标记处检测到组态的信号跳转（已在驱动组件中组态）后，同步在外部零标记处（阶段 1）结束。将轴位置设置为默认值减去在“参考点坐标”(Reference point coordinate)(*Mode = 0*) 中或在输入参数 *Position* (*Mode = 1*) 中定义的参考点位移。

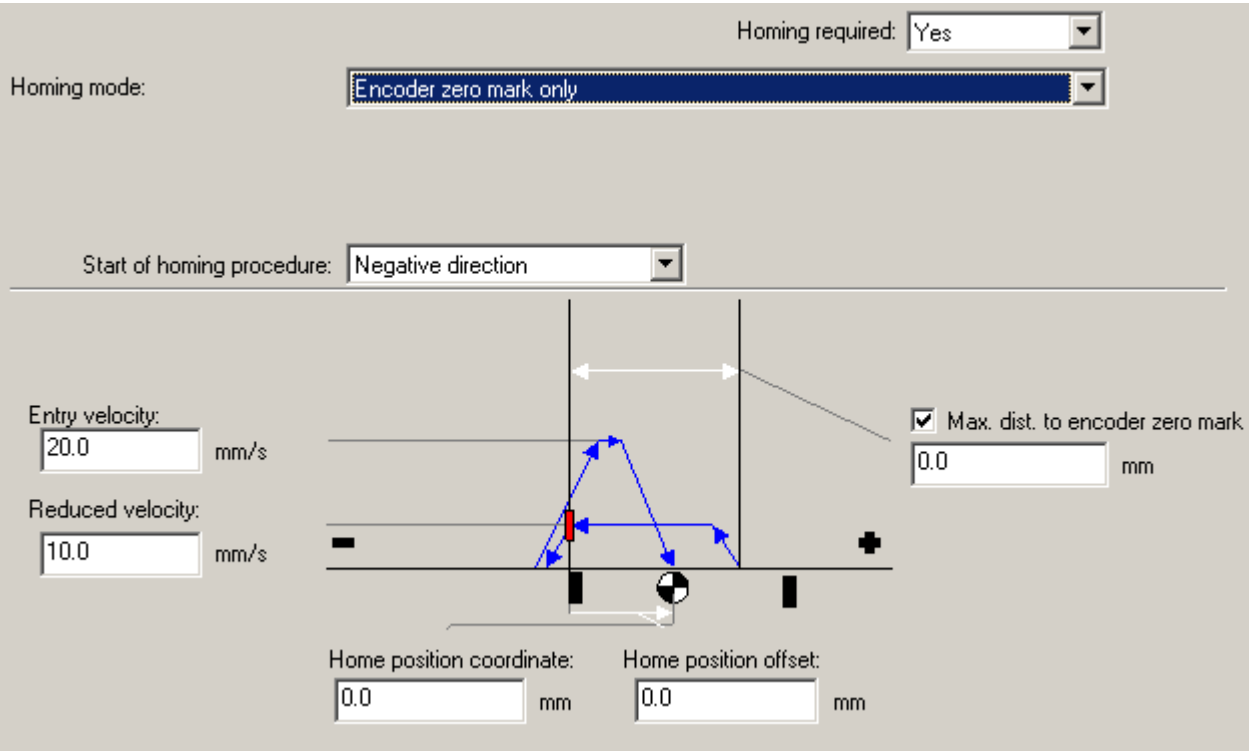
也可以通过设置“到外部零标记的最大距离”(Max. distance to external zero mark) 复选框来监视轴在参考点逼近的起点和信号沿检测之间移动的距离。如果在参考点逼近的起点和已组态距离的终点之间没有检测到沿，则回原点将因出错而取消。

- 阶段 2：参考点逼近

检测到组态的信号沿后，轴将加速/减速，从而以关闭速度逼近参考点坐标。

如果已组态了参考点位移，则轴以相应的距离（起点为同步位置）逼近该参考点。运动方向由参考点位移的符号以及检测到外部零标记后的减速斜坡长度确定（前提是参考点位于减速斜坡内）。

使用“仅编码器零标记”(Encoder zero mark only) 的主动回原点模式



例如，对于某些轴，编码器在轴的整个行程范围内只能设置一个编码器零标记，在这些轴上就会使用回原点到编码器零标记。回原点命令启动到编码器零标记的轴逼近。检测到编码器零标记后，轴以回原点速度逼近移位后的参考点。现在，轴位置具有在参考点坐标中定义的值。该图中的图形显示最终的运动顺序。各个参数的含义：

参数	值	说明
需要回原点	是	轴必须回原点才能执行绝对运动命令。
	否	轴不必回原点就可以执行绝对运动命令。
回原点模式	-	在此例中：“仅编码器零标记”(Encoder zero mark only)
开始参考点逼近	正方向	正方向上的参考点逼近。
	负方向	负方向上的参考点逼近。
进入速度	-	检测到编码器零标记后，轴逼近（移位后的）参考点的速度
关闭速度	-	轴开始参考点逼近以及逼近编码器零标记的速度

参数	值	说明
回原点位置偏移	-	回原点位置偏移功能会将回原点位置移位所组态的距离，这意味着轴以“回原点速度”(Homing velocity) 在编码器零标记检测位置之后移动所组态的距离。 模数轴始终取最短距离。
到编码器零标记的最大距离	取消激活	没有监视到编码器零标记所需要移动的距离
	激活	监视从参考点逼近的起点到检测出编码器零标记时所移动的距离 如果该距离的差值超出了组态的值，则相应的工艺 DB 将返回错误 801D。参考点逼近取消。

参考点逼近的顺序

- 阶段 1：与编码器零标记同步

轴以组态的关闭速度，沿在“参考点逼近的起点”(Start of reference point approach) 中设置的方向开始其参考点逼近。与编码器零标记的同步（阶段 1）在检测到编码器零标记后结束。将轴位置设置为默认值减去在“参考点坐标”(Reference point coordinate)(*Mode = 0*) 中或在输入参数 *Position* (*Mode = 1*) 中定义的参考点位移。

也可以通过设置“到零标记的最大距离”(Max. distance to zero mark) 复选框，来监视轴在参考点逼近的起点和编码器零标记检测之间移动的距离。如果在参考点逼近的起点和已组态距离的终点之间没有检测到编码器零标记，则回原点将因出错而取消。

- 阶段 2：参考点逼近

检测到编码器零标记后，轴将加速/减速，从而以进入速度逼近参考点坐标。

如果已组态了参考点位移，则轴以相应的距离（起点为同步位置）逼近该参考点。运动方向由参考点位移的符号以及检测到编码器零标记后的减速斜坡长度确定（前提是参考点位于减速斜坡内）。

4.6.14.3 回原点 —“被动回原点”(Passive homing) 标签

可以在 S7T Config 的轴 (Axis) > 回原点 (Homing) 对话框的“被动回原点”(Passive homing) 选项卡中组态被动回原点。与主动回原点相比，这二者的不同之处在于，被动回原点所需的回原点运动不是由回原点命令启动的。

“被动回原点”(Passive homing) 有三种可用的回原点模式：

- 参考凸轮和编码器零标记
- 仅外部零标记
- 仅编码器零标记

启用回原点功能：

MC_Home	<i>Mode = 2</i> <i>Position = x</i>	为当前位置分配了参考点输入参数 <i>Position</i> 的值。

成功回原点后，轴工艺 DB 将返回状态 *Statusword.HomingDone = TRUE*。

说明

请注意，并非所有驱动组件都支持所有的回原点模式或测量信号计算功能。有关详细信息，请参考所用驱动组件的文档。

由工艺 DB 上相应的错误消息来表示在被动回原点操作期间对编码器零标记或参考凸轮监视的触发。在这种情况下，当前轴运动终止。

使用“参考凸轮和编码器零标记”(Reference cam and encoder zero mark) 的被动回原点模式

Homing required: Yes

Homing mode: Homing output cam and encoder zero mark

Log. address of homing cam: 64

Bit number: 0

Direction of motion: Current direction

☐ Max. distance to homing output cam
0.0 mm

☐ Max. dist. to encoder zero mark
0.0 mm

Home position coordinate

轴经过参考凸轮后，下一个编码器零标记将触发轴回原点。将参考点的轴位置设置为通过“MC_Home”工艺功能的输入参数 *Position* 定义的值。

各个参数的含义：

参数	值	说明
需要回原点	是	轴必须回原点才能执行绝对运动命令。
	否	轴不必回原点就可以执行绝对运动命令。
回原点模式	-	在此例中：“参考凸轮和编码器零标记”(Reference cam and encoder zero mark)
运动方向	正方向	轴仅通过正逼近方向回原点到编码器零标记。
	负方向	轴仅通过负逼近方向回原点到编码器零标记。
	当前方向	轴在到达下一个编码器零标记时回原点。
参考凸轮的逻辑地址	[字节地址]	参考凸轮的逻辑字节地址 可以将参考凸轮连接至 T-CPU 的集成输入或连接至 DP(DRIVE) 上的 IO。
位号	[位号]	用于参考凸轮的信号的位地址

参数	值	说明
到回原点输出凸轮的最大距离	取消激活	没有监视到参考凸轮检测的距离
	激活	监视回原点功能的起点和参考凸轮检测之间的距离。如果该距离的差值超出了组态的值，则相应的工艺 DB 将返回错误 <i>801D</i> 。回原点功能取消。
到编码器零标记的最大距离	取消激活	没有监视到编码器零标记所需要移动的距离
	激活	监视轴在参考凸轮和编码器零标记之间移动的距离。如果该距离的差值超出了组态的值，则相应的工艺 DB 将返回错误 <i>801D</i> 。回原点功能取消。

使用“仅外部零标记”(External zero mark only) 的被动回原点模式

Homing required:

Homing mode:

On the external zero mark side:

Direction of motion:

☐ Max. distance to ext. zero mark

mm

Home position coordinate

在检测到外部零标记时，开始轴回原点。将轴设置为“MC_Home”工艺功能的 *Position* 输入参数的值。

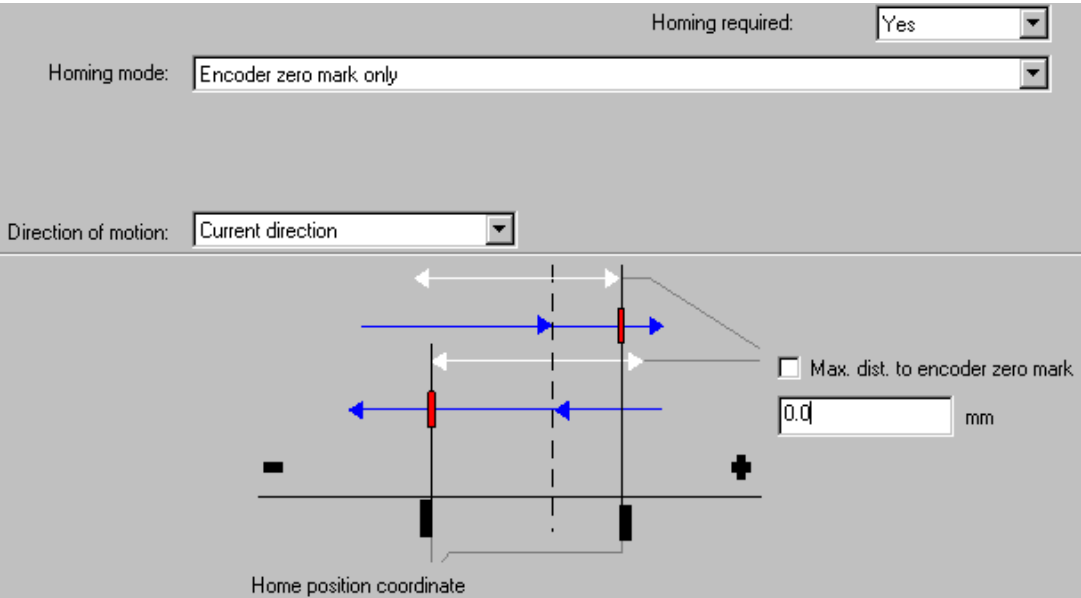
各个参数的含义：

参数	值	说明
需要回原点	是	轴必须回原点才能执行绝对运动命令。
	否	轴不必回原点就可以执行绝对运动命令。
回原点模式	-	在此例中：“仅外部零标记”(External zero mark only)
运动方向	正方向	轴仅通过正逼近方向回原点到外部零标记。
	负方向	轴仅通过负逼近方向回原点到外部零标记。
	当前方向	轴在到达下一个外部零标记时回原点。
在外部零标记一侧	左	移动方向： 正方向 在上升沿上触发轴回原点。 移动方向： 负方向 在下降沿上触发轴回原点。
	右	移动方向： 正方向 在下降沿上触发轴回原点。 移动方向： 负方向 在上升沿上触发轴回原点。
到外部零标记的最大距离	取消激活	没有监视到外部零标记所需要移动的距离
	激活	该功能将监视轴从其回原点功能的起点到检测出外部零标记所移动的距离 如果该距离的偏差超出了组态的距离，则相应的轴工艺 DB 将指示错误 801D。回原点功能取消。

说明

- 对于使用“外部零标记”(external zero mark) 的回原点，将外部零标记作为数字测量输入连接至驱动组件。
- 为了能够根据需要以“外部零标记”(external zero mark) 回原点模式执行参考点逼近，您应始终在“移动方向”(Direction of movement) 和“在外部零标记一侧”(on the side of the external zero mark) 上设置数值，使其与驱动器组态或功能兼容。“在外部零标记一侧”(on the side of the external zero mark) 和“移动方向”(Direction of movement) 参数不用于组态驱动器中的测量功能，仅反映其功能。
- 有关外部零标记检测的组态的信息，请参考相关的产品信息或驱动器手册。

使用“仅编码器零标记”(Encoder zero mark only) 的被动回原点模式



例如，使用“仅零标记”(Zero mark only) 的回原点模式用于装有一个在轴的整个行程范围内只能输出一个零标记信号的编码器的轴。

在检测到编码器零标记时，开始轴回原点。检测到编码器零标记后，将轴的位置值设置为“MC_Home”工艺功能的输入参数 *Position* 的值。

各个参数的含义：

参数	值	说明
需要回原点	是	轴必须回原点才能执行绝对运动命令。
	否	轴不必回原点就可以执行绝对运动命令。
回原点模式	-	在此例中：“仅编码器零标记”(Encoder zero mark only)
运动方向	正方向	轴仅通过正逼近方向回原点到编码器零标记。
	负方向	轴仅通过负逼近方向回原点到编码器零标记。
	当前方向	轴在到达下一个编码器零标记时回原点。
到编码器零标记的最大距离	取消激活	没有监视到编码器零标记所需要移动的距离
	激活	该功能将监视轴从其回原点功能的起点到检测到编码器零标记所移动的距离。 如果该距离的偏差超出了组态的距离，则相应的轴工艺 DB 将指示错误 801D。回原点功能取消。

使用“默认”(Default) 的被动回原点模式

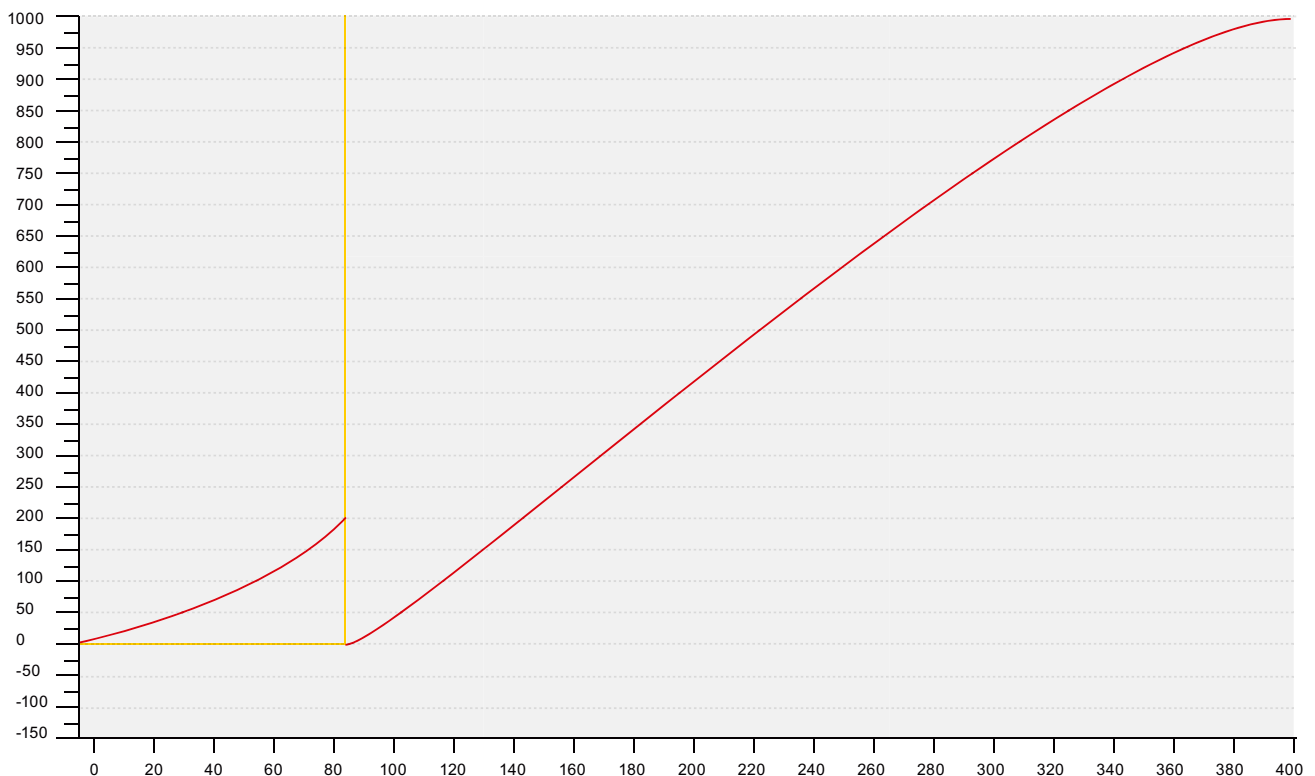
在 S7T Config 中创建新轴时，会预设“默认”(Default) 回原点模式。

如果组态的编码器提供零标记，则使用“仅编码器零标记”(encoder zero mark only) 回原点模式。如果编码器不提供零标记，则使用仅外部零标记回原点模式。

4.6.14.4 被动回原点模式的定位特性

以下几个示例表明了使用运动命令“MC_MoveRelative”和“MC_MoveAbsolute”时被动回原点模式的终点位置的响应：

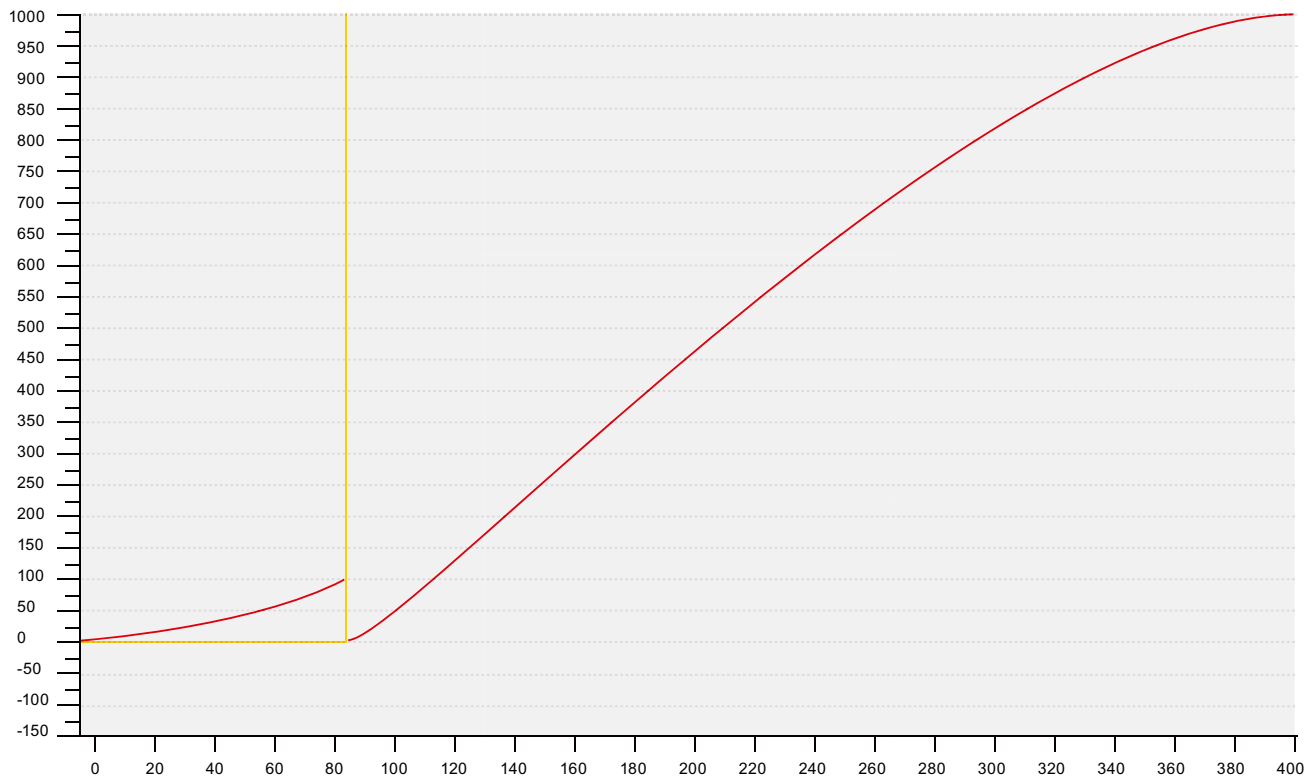
使用“MC_MoveRelative”的被动回原点模式的定位特性



针对被动回原点模式启动 MC_MoveRelative 命令 1000。

1. MC_MoveRelative 命令和被动回原点模式同时启动。
2. 到达黄色标记的回原点位置 200 时，轴回原点，轴的位置设置为位置 0。
3. 轴移动剩余的相对距离，移到位置 800 (1000-200)。

使用“MC_MoveAbsolute”的被动回原点模式的定位特性



针对被动回原点模式启动到位置 1000 的 MC_MoveAbsolute 命令。

1. MC_MoveAbsolute 命令和被动回原点模式同时启动。
2. 到达黄色标记的回原点位置 200 时，轴回原点，轴的位置设置为位置 0。
3. 根据新回原点位置，轴移动到位置 1000。

4.6.14.5 直接回原点

将实际轴位置设置为在 **MC_Home** 工艺功能中定义的值。没有激活的参考点位移设置。该功能不执行运动。执行命令时使轴回原点。

启用回原点功能

MC_Home	<i>Mode = 3</i> <i>Position = x</i>	直接回原点： 为当前位置分配了输入参数 <i>Position</i> 的值。

成功回原点后，轴工艺 DB 将返回状态 *Statusword.HomingDone = TRUE*。

4.6.14.6 位置校正

从轴的实际位置值中扣除校正值。在这种情况下，与其它回原点模式不同，轴将保持其回原点状态（已回原点/未回原点）。

位置校正功能还可以用于操纵各个坐标系（基准坐标系、叠加坐标系）的设定值。这对于叠加凸轮传动能够在凸轮盘中生成参考非常重要。

启用回原点功能：

MC_Home	<i>Mode = 4</i> <i>Position = x</i>	实际值校正： 位置值 = (当前位置) - (参数位置)。
	<i>Mode = 6</i> <i>Position = x</i>	基坐标系中的设定值校正： 位置值 = (当前位置) - (参数位置)。
	<i>Mode = 7</i> <i>Position = x</i>	叠加坐标系中的设定值校正： 位置值 = (当前位置) - (参数位置)。

位置校正不会影响轴工艺 DB 中的 *Statusword.HomingDone* 状态。

4.6.14.7 未回原点的轴的运动

可以在**轴 (Axis) > 回原点 (Homing)** 对话框中确定绝对定位是否可用于未回原点的轴。

“需要回原点”(Homing required) 的可选设置：

- 否： 可以进行相对运动和绝对运动。将监视软件限位开关。
- 是： 仅可以进行相对运动。只要轴没有回原点便不监视软件限位开关。

4.6.15 监视功能

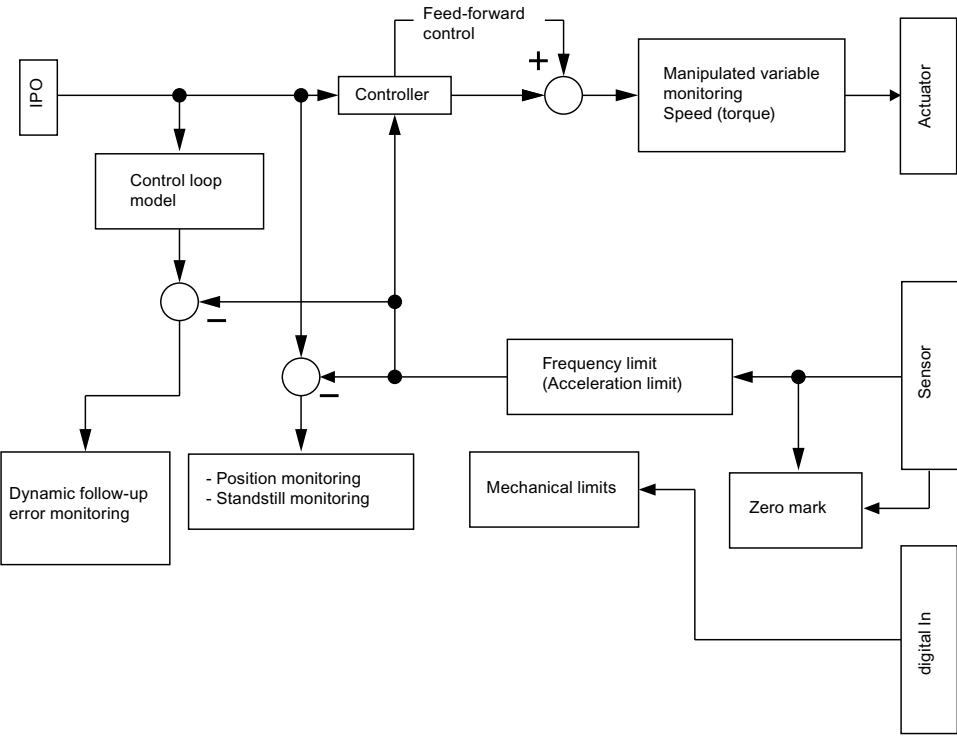
4.6.15.1 监视功能 — 总览

可以在 S7T Config 中组态的轴监视功能：

监视功能	速度控制轴	定位轴	同步轴
速度误差监视	X	-	-
定位监视	-	X	X
跟随误差监视	-	X	X
停止信号	X	X	X
软件限位开关	-	X	X
硬件限位开关	X	X	X
同步监视	-	-	X
受控变量监视（始终可用）	X	X	X
编码器限制频率监视	X（仅适用于编码器）		X

定位轴监视

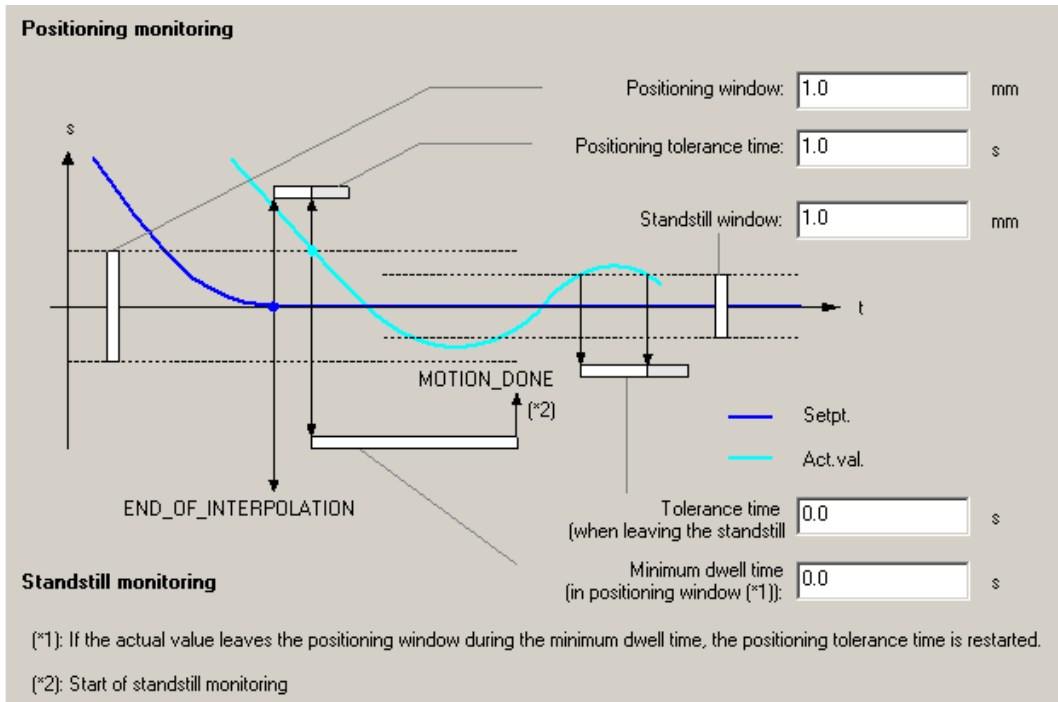
下图显示了一个定位轴监视的示例：



在工艺 DB 的 *ErrorStatus* 参数中报告错误。

4.6.15.2 监视功能 — 定位和停止监视

可以在轴 (Axis) > 监视功能 (Monitoring functions) 对话框的“定位和停止监视”(Positioning and standstill monitoring) 选项卡中设置用于监视位置控制轴和同步轴定位的限制值。



“定位和停止监视”(Positioning and standstill monitoring) 选项卡显示为轴设置了定位还是同步操作工艺。

定位监视

在定位即将结束时，该功能将监视轴进入目标位置。为此，指定一个定位窗口和一个时间公差，在该时间公差内必须到达终点位置。在设定值插补结束时激活监视。

也可以将定位窗口中的最小停留时间设置为在激活定位命令的正反馈之前结束。该时间可以用于振动过程和控制回路，为此公差窗口应小于超调振幅。

当实际值到达定位窗口时，将输出定位命令的正反馈。

如果触发了定位监视功能，则轴工艺 DB 将返回错误 8019。

停止（零速度）监视

当定位命令的位置设定值与目标位置的值相等并且停止监视激活的延迟已到期时，将启用停止监视。

如果轴移至已组态的停止公差窗口之外的时间超过已组态的时间段，则将触发停止监视。

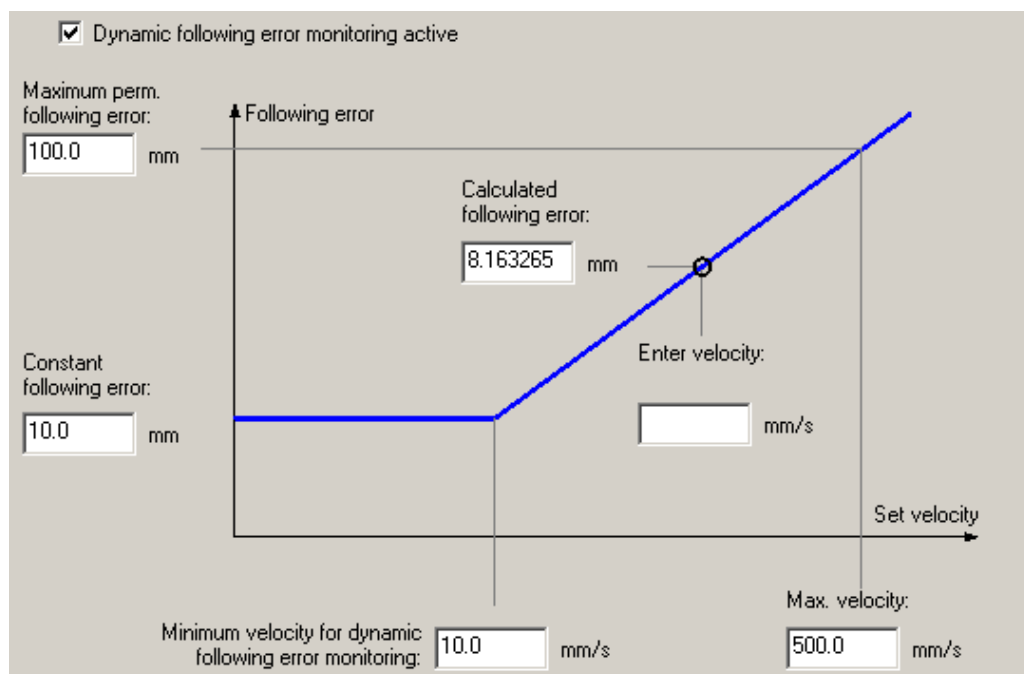
如果触发了停止监视，则轴工艺 DB 将返回错误 8018。

4.6.15.3 监视功能 —“跟踪误差监视”(Following error monitoring) 标签

可以在轴 (Axis) > 监视功能 (Monitoring functions) 对话框的“跟踪误差监视”(Following error monitoring) 选项卡中启用动态跟踪误差监视。

使用计算出的跟踪误差执行对位置控制轴的跟踪误差监视。如果实际位置值和位置设定值之间的偏移超出已编程的跟踪误差限制，轴工艺 DB 将返回错误 8016。允许的跟踪误差取决于轴的速度设定值。

当速度小于已组态的最小速度时允许的跟踪误差为常量，这是在“恒定跟踪误差”(constant following error) 参数中编程的。大于此限制时，跟踪误差将线性增加倒最大值，该最大值由最大速度时的参数“允许的最大跟踪误差”(maximum permitted following error) 定义。在最大速度时达到允许的最大跟踪误差。



如果在“输入速度”(Enter velocity) 输入框中指定一个值，则相应的功能值显示在“计算的跟踪误差”(Calculated following error) 框中。这允许您控制在输入的速度下跟踪误差可以有多大。

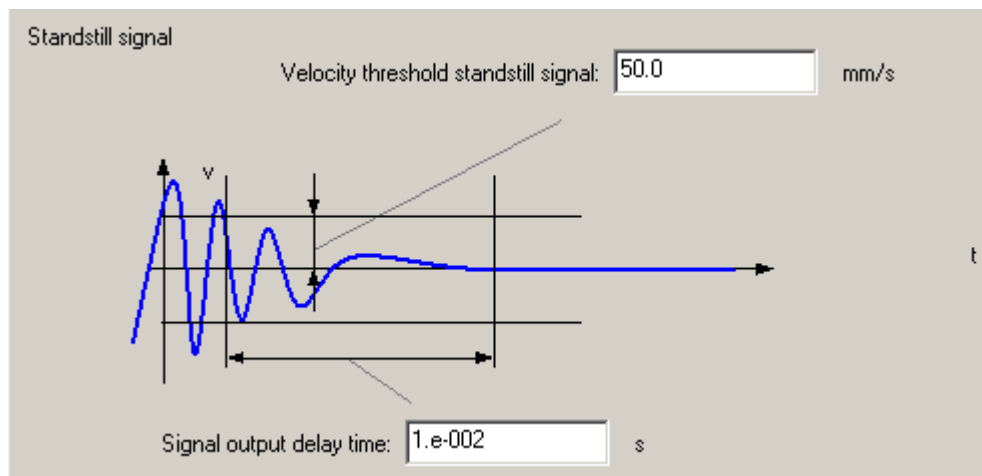
对于位置控制的实际轴，将显示“跟踪误差监视”(Following error monitoring) 选项卡。

4.6.15.4 监视功能 —“停止信号”(Standstill signal) 标签

可以在“轴”(Axis) > “监视功能”(Monitoring functions) 对话框的“停止信号”(Standstill signal) 选项卡中设置速度阈值和“信号输出延迟”(Signal output delay)。

如果当前速度至少在设定的延迟时间期间低于组态的速度阈值，则停止信号将在工艺 DB 的状态字中设置“停止”(standstill) 位。

在速度控制轴和定位轴上，将在速度控制模式下激活重写。在定位轴上，将在到达定位窗口时激活重写。

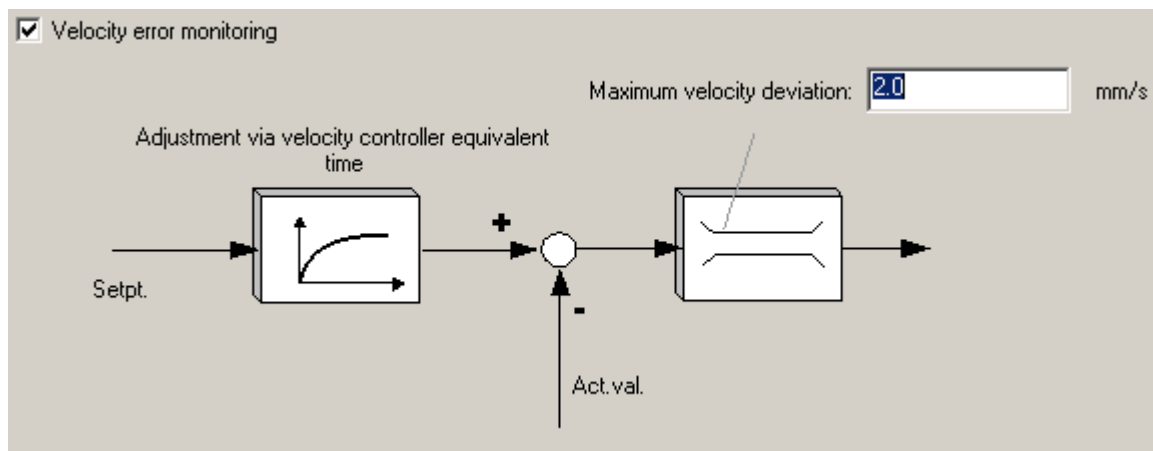


4.6.15.5 监视功能 —“速度监视”(Velocity monitoring) 标签

可以在轴 (Axis) > 监视功能 (Monitoring functions) 对话框的“速度误差监视”(Velocity error monitoring) 选项卡中启用速度误差监视。

速度误差监视仅与以下这些轴有关：

- 带有编码器的速度控制轴
- 操作在速度控制模式下的定位轴
(“MC_MoveVelocity”—输入参数 *PositionControl = FALSE*)



速度误差监视

在该对话框中启用速度误差监视。

最大速度误差：

仅当启用了速度误差监视时该输入框才可见。

在该对话框中输入最大速度误差。

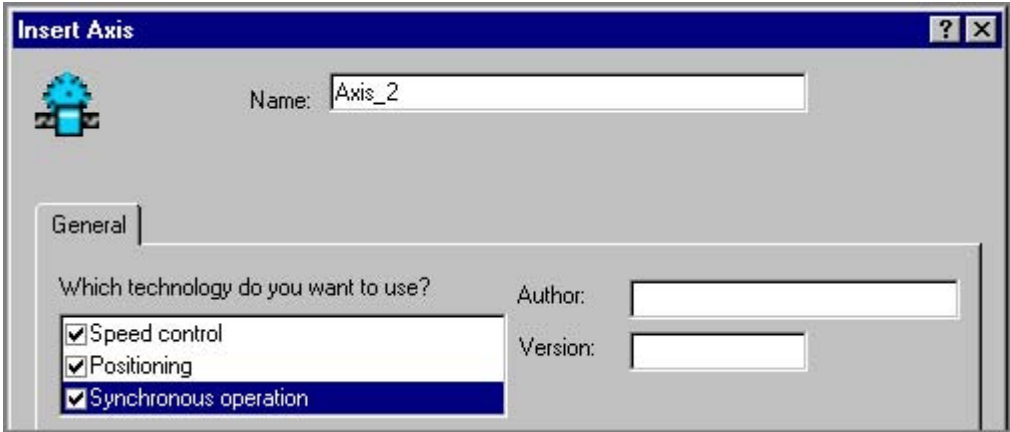
4.7 组态同步轴

4.7.1 组态同步轴

先决条件

- 已在 HW Config 中组态了 T-CPU，并已编译了组态数据。

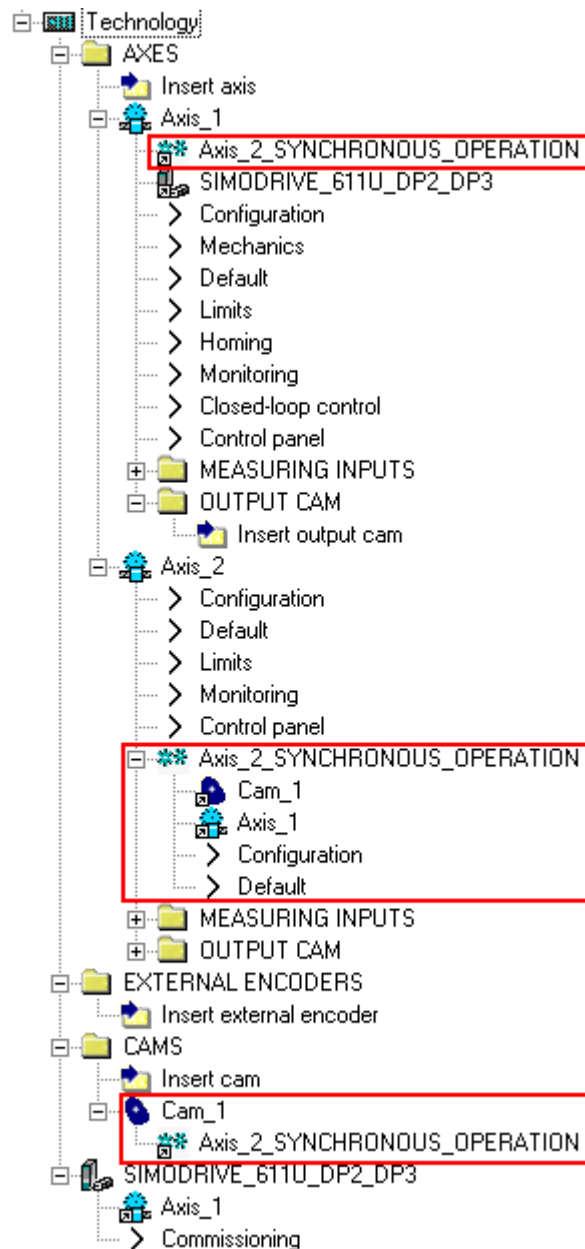
如何在 S7T Config 中插入同步轴

步骤	说明
1.	在 S7T Config 的浏览器中，双击“插入轴”(Insert axis)。
2.	在组态轴时激活同步操作工艺。  该“同步”工艺需要激活“定位”和“速度控制”工艺。为此，您不能禁用这些工艺。
3.	按照『组态轴』一章中的说明执行这些步骤。
	结果：同步轴显示在 S7T Config 的浏览器中，其相应的同步对象会自动创建。该对象的名称会自动指定为轴的名称再加上 _SYNCHRONOUS_OPERATION 结尾。

在浏览器中的表达方式

分配给同步轴的引导轴和凸轮盘通过逻辑链接表示：

- 在同步操作对象下面
- 在已使用的凸轮下面
- 在引导轴（轴、外部编码器）下面



其它步骤

对于同步操作，请为同步轴分配相应的引导轴和/或凸轮盘。

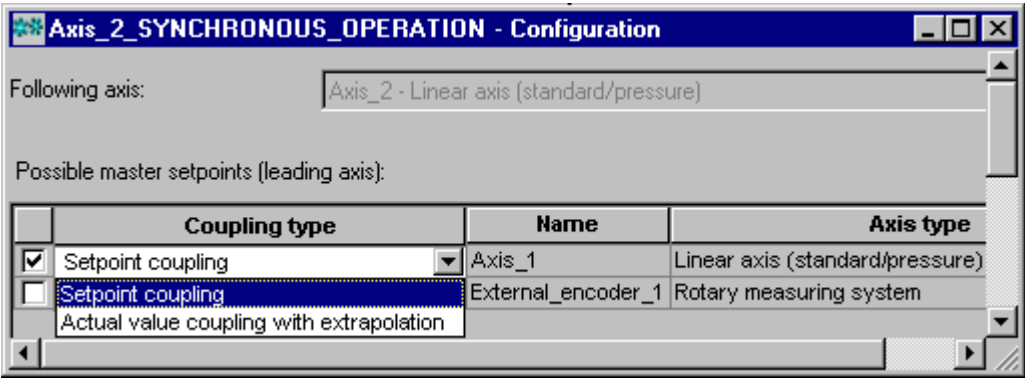
4.7.2 分配引导轴和凸轮盘

要执行传动，您必须为跟随轴分配一个可用的引导轴。要执行凸轮传动，您必须为跟随轴再分配一个可用的凸轮盘。错误！未定义书签。

先决条件

- 将具有“同步”工艺的跟随轴插入 S7T Config 中。
- 将引导轴或凸轮盘插入 S7T Config 中。

如何将引导轴和凸轮盘分配给跟随轴

步骤	说明
1.	在 S7T Config 的浏览器中，双击同步对象（同步轴的子对象）中的“组态”(Configuration)。
2.	在下一个对话框中，为跟随轴分配一个或多个可用的引导轴。要执行此操作，可以选中左侧列中的复选框，然后选择相关的耦合模式。可能的耦合模式： 引导轴为实际轴： • 设定值耦合 • 带有推断的实际值耦合 引导轴为虚拟轴： • 设定值耦合 引导轴为外部编码器： • 带有推断的实际值耦合 • 无推断的实际值耦合  由哪个轴为跟随轴提供主设定值取决于调用相关工艺功能（例如“MC_GearIn”或“MC_CamIn”）的运行时间。

步骤	说明									
3.	从对话框底部，选择可选的凸轮盘以执行凸轮传动。 Possible valve characteristics: <table><tr><th></th><th>Name</th><th>Cam type</th></tr><tr><td>☐</td><td>Cam_1</td><td>Cam configuration</td></tr><tr><td>☐</td><td>Cam_2</td><td>Cam configuration</td></tr></table>		Name	Cam type	☐	Cam_1	Cam configuration	☐	Cam_2	Cam configuration
	Name	Cam type								
☐	Cam_1	Cam configuration								
☐	Cam_2	Cam configuration								
4.	单击“关闭”(Close)									

4.7.3 组态叠加同步

同步轴包含一个轴对象和一个同步对象。两个对象均在您创建轴时生成。您可以通过为同步轴分配一个协调叠加运动的附加同步对象来组态叠加同步。

同步轴最多可以包含两个同步对象：

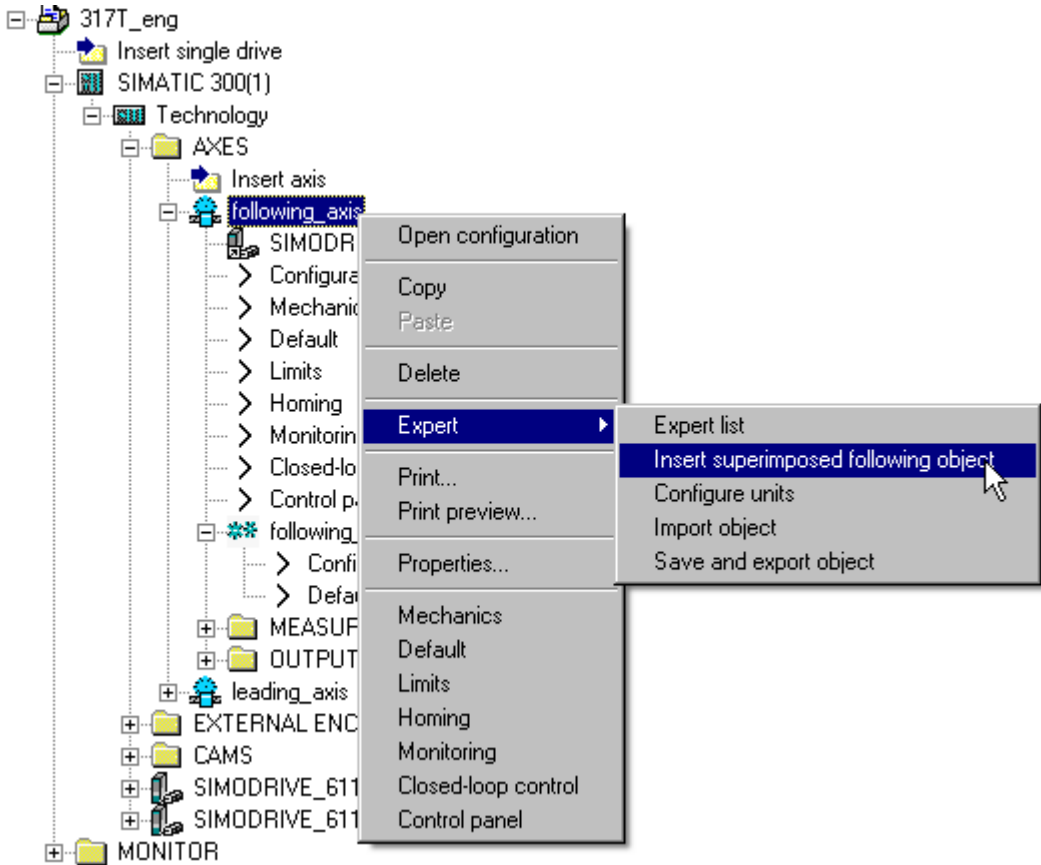
- 一个同步对象用于基本同步
- 一个同步对象用于叠加同步

先决条件

- 在 S7T Config 中组态同步轴。
- 没有为该轴组态叠加同步对象

组态一个叠加同步对象

接下来的步骤会向您说明组态叠加同步对象的基本步骤。

步骤	说明
1.	在 S7T Config 的浏览器中，选择要组态叠加同步的同步轴。从上下文菜单中选择“专家”(Expert) >“叠加同步对象”(Superimposed synchronous object)。  结果：浏览器在同步轴下面显示第二个同步对象。
2.	双击第二个同步对象的组态 (Configuration) 条目，打开组态对话框。

步骤	说明																							
3.	通过为叠加同步的同步组分配引导轴（如有需要可包括凸轮盘），可以对其进行确定。选中相应的复选框，然后选择引导轴的耦合类型。 FollowingAxis_SYNCHRONOUS_OPERATION - Configuration Following axis: FollowingAxis - Rotary axis (standard/pressure) Possible master setpoints (leading axis): <table><tr><th></th><th>Coupling type</th><th>Name</th><th>Axis type</th><th>Device</th></tr><tr><td>☒</td><td>Setpoint coupling</td><td>MasterAxis</td><td>Rotary axis (standard/pressure)</td><td>Technologie</td></tr><tr><td>☐</td><td>Actual value coupling with e</td><td></td><td></td><td></td></tr></table> Possible cams: <table><tr><th></th><th>Name</th><th>Axis type</th><th>Device</th></tr><tr><td>☐</td><td>Cam_1</td><td>Cam configuration</td><td>Technologie</td></tr></table> Close Help 叠加同步对象的同步特定参数组态和分配值与基本同步操作相同。		Coupling type	Name	Axis type	Device	☒	Setpoint coupling	MasterAxis	Rotary axis (standard/pressure)	Technologie	☐	Actual value coupling with e					Name	Axis type	Device	☐	Cam_1	Cam configuration	Technologie
	Coupling type	Name	Axis type	Device																				
☒	Setpoint coupling	MasterAxis	Rotary axis (standard/pressure)	Technologie																				
☐	Actual value coupling with e																							
	Name	Axis type	Device																					
☐	Cam_1	Cam configuration	Technologie																					

区分同步对象

在同步对象的专家列表中的 *SyncingMotion.motionImpact* 组态参数中，定义同步轴上的同步对象是代表基本同步还是叠加同步。

<i>SyncingMotion.motionImpact</i>	STANDARD (0)	基本同步操作
	SUPERIMPOSED_MOTION (1)	叠加同步

4.7.4 同步

可以根据应用分别定义是在同步期间还是在去同步期间耦合主设定值。

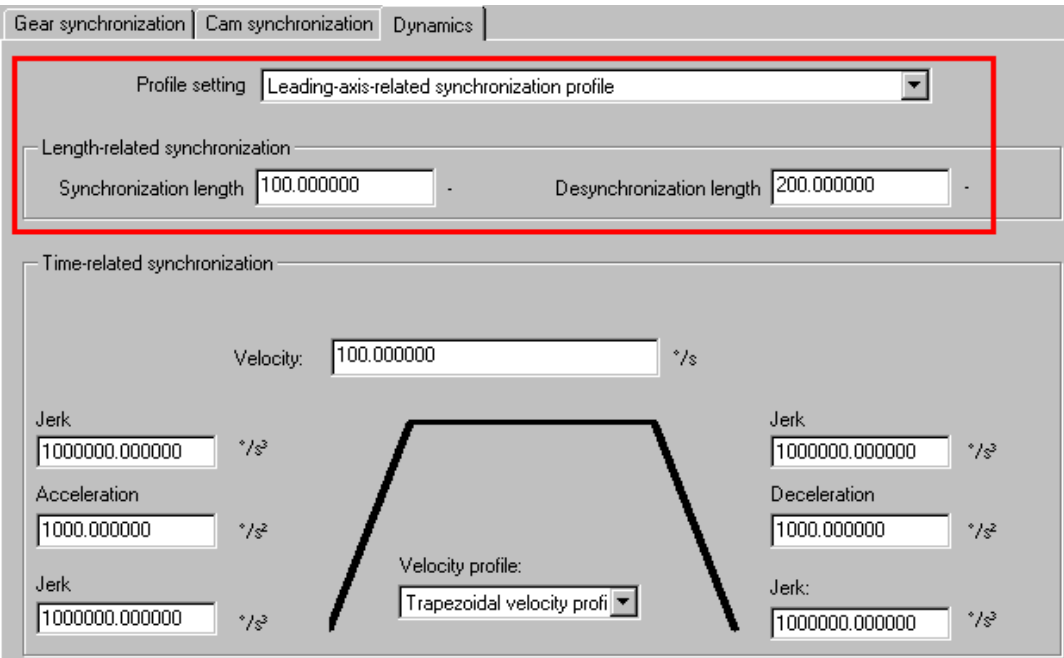
其取决于：

- 同步模式
- 同步点
- 同步点的位置
- 同步条件
- 同步的方向（仅针对模数轴）

您可以在**同步操作 (Synchronous operation) > 默认 (Default)** 对话框的“动态”(Dynamics) 选项卡中设置以下同步模式。

- **与引导轴相关的同步曲线：** 同步和去同步（与主值参考）
- **与时间相关的同步曲线：** 基于指定动态值的同步和去同步

基于与引导轴相关的同步曲线的同步



与引导轴相关的同步曲线在“同步长度”(Sync. length) 和“去同步长度”(Desync. length) 参数中进行定义。

Sync. Length	同步长度
Desync. Length	去同步长度

对于与引导轴相关的同步，为同步（包括同步长度 [“同步长度”或“去同步长度”]）定义了与主设定值相关的起始值或结束值。

同步曲线随即确定，即动态转换的计算与主设定值的动态曲线无关。

同步操作沿的速度为常量。

在参考主设定值的情况下，对话框中定义的同步长度可以成为定义同步范围的基础。使用“MC_GearIn”（传动）、“MC_CamIn”（凸轮传动）、“MC_GearInSuperImposed”（叠加传动）或“MC_CamInSuperImposed”（叠加凸轮传动）工艺功能来处理同步。

工艺功能中的输入参数 *Mode* 用于确定是否为同步曲线激活在 S7T Config (*Mode* = 0) 中设置的值，或者是否激活工艺功能 (*Mode* = 1) 中的动态默认值。

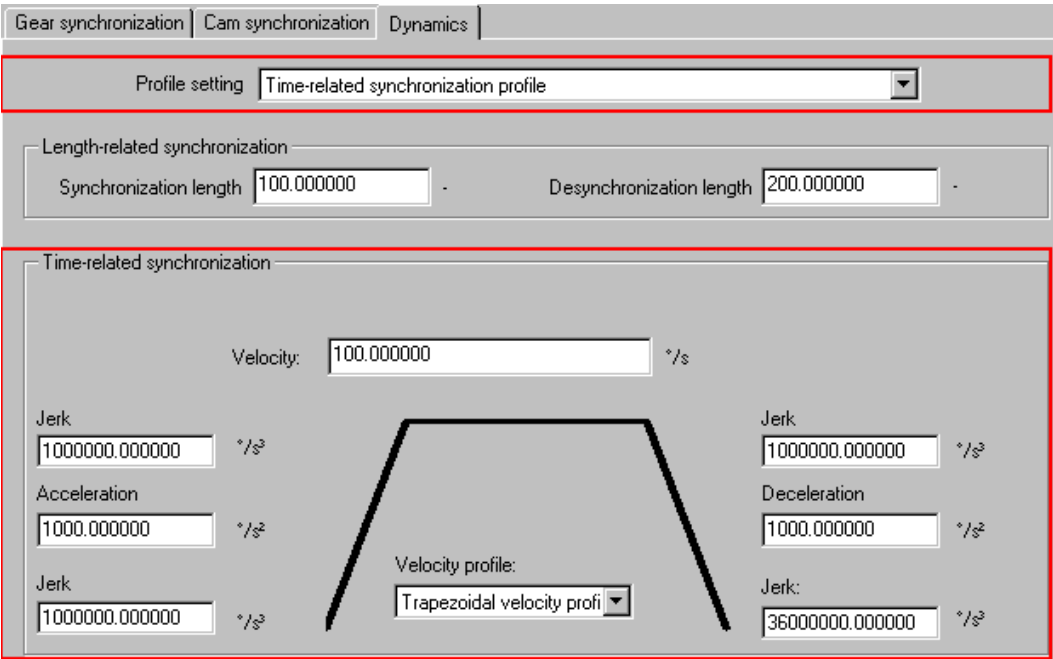
说明

同步曲线取决于引导轴的速度和设置的同步长度。同步对象中的所有动态定义都将被忽略。因此，动态响应仅限于从轴。

基于与时间相关的同步曲线的同步

根据在 S7T Config 或块输入中设置的动态参数，在启动同步功能之后执行基于时间的同步。

工艺功能中的输入参数 *Mode* 用于确定是否为同步曲线激活在 S7T Config (*Mode* = 0) 中设置的值，或者是否激活工艺功能 (*Mode* = 1) 中的动态默认值：



与时间相关的同步配置文件由“冲击”(Jerk)、“加速度”(Acceleration)、“减速度”(Deceleration)、“速度”(Velocity) 和“速度曲线”(Velocity profile) 参数定义。

冲击 (Jerk)	冲击设置支持下面列出的速度转换： - 加速开始时的冲击 - 加速结束时的冲击 - 减速开始时的冲击 - 减速结束时的冲击
加速度 (Acceleration)	同步期间轴的加速度
减速度 (Deceleration)	同步期间轴的减速度
速度 (Velocity)	同步期间轴的最大速度 是否可以达到该速度取决于同步条件
速度曲线 (Velocity profile)	可用的速度曲线如下： - 梯形速度曲线 - 恒定速度曲线

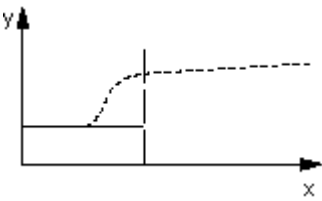
延迟同步和高级同步

在基于时间的同步模式中，可以区分高级同步和延迟同步。

	同步起点...	同步终点...
高级同步	由系统根据默认动态值和主设定值特性进行计算。	根据跟随轴位置模糊定义或直接定义
延迟同步	根据跟随轴位置模糊定义或直接定义	从动态值和主设定值特性中导出

说明
使用默认动态值同步定义了同步起点的同步轴时，跟随轴会沿双曲线斜坡加速至目标速度。

高级同步（在同步位置之前同步）



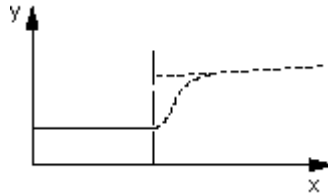
- 对于高级同步，同步运动开始方式如下。
- 如果该同步可以在最佳时间（动态主设定值）内执行，则根据默认动态响应参数和恒定主设定值，同步运动开始于同步的起点
 - 如果无法计算最佳同步时间，但可以到达同步位置（例如使用静态主设定值），则同步运动立即开始

说明
如果无法顺利执行同步，则可能生成反向运动。

对速度主设定值所做的任何更改也都将相应地应用于同步操作的动态值。 如果超出了组态的公差，则输出一条警告消息。

如果主设定值在同步处于活动状态时反转，则同步会因出错而被取消。

延迟同步（在与同步位置对称时开始）



在延迟同步模式下

- 同步运动在达到开始条件后开始
- 仅支持梯形速度曲线

如果设置了带有恒定加速度的速度曲线，则会生成警告“无法保持运动曲线.....的动态响应”(Dynamic response of motion profiles... cannot be maintained)，并且系统会自动切换为梯形速度曲线。

同步方向

您可以在以下对话框中设置同步方向：

- 齿轮同步
Axis_SYNCHRONOUS_OPERATION > 默认 (Default) 对话框中“齿轮同步”(Gear synchronization) 选项卡中的“同步方向”(Synchronization direction)
- 凸轮同步
Axis_SYNCHRONOUS_OPERATION > 默认 (Default) 对话框中“凸轮同步”(Cam synchronization) 选项卡中的“同步方向”(Synchronization direction)

下述设置适用于同步和去同步。

您还可以对 *userdefault.gearingsettings.synchronizingdirection* 或 *userdefault.cammingsettings.synchronizingdirection* 系统变量中的设置进行编程。

- 兼容性模式 (*SYSTEM_DEFINED*)
跟随轴在可能的最短距离内同步。如果移动跟随轴的同步在当前设置的方向实现，则保持此方向。如果更改当前运动方向来实现同步，则将此方向反向。系统会计算同步所需的运动顺序。
- 最短距离 (*SHORTEST_WAY*)
跟随轴在可能的最短距离内同步。

- 负同步方向 (*NEGATIVE_DIRECTION*)

仅允许在负方向进行同步，即必须避免在同步期间从负速度转为正速度。如果同步失败，则 S7T Config 会输出错误消息“当激活/取消激活同步操作时出现 50007 错误”(50007 Error when activating/deactivating synchronous operation)。

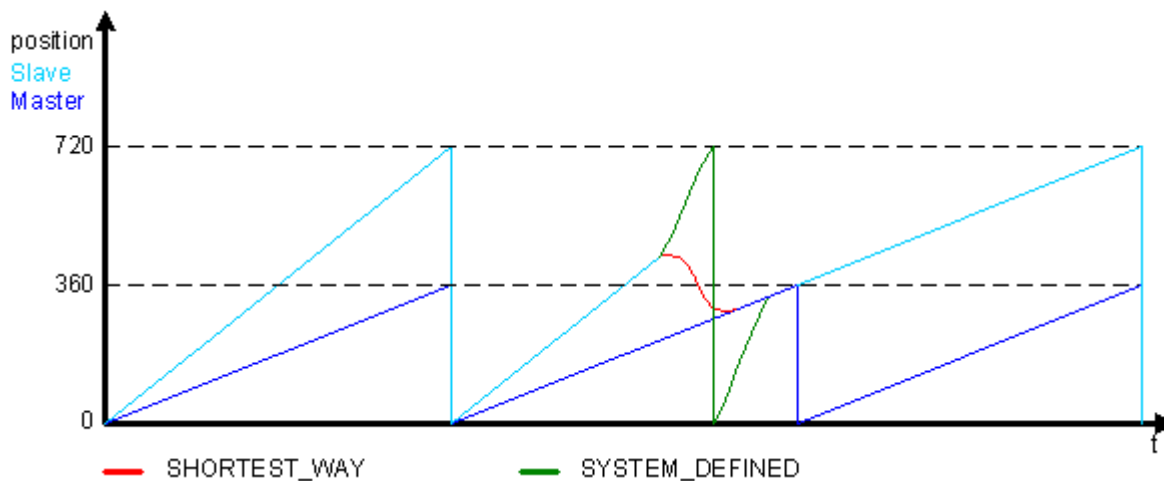
- 正同步方向 (*POSITIVE_DIRECTION*)

仅允许在正方向进行同步，即必须避免在同步期间从正速度转为负速度。如果同步失败，则 S7T Config 会输出错误消息“当激活/取消激活同步操作时出现 50007 错误”(50007 Error when activating/deactivating synchronous operation)。

如果“MC_CamIn”或“MC_GearIn”中的 *Mode* = 1，则始终使用兼容性模式。

“SHORTEST_WAY”和“SYSTEM_DEFINED”之间的区别

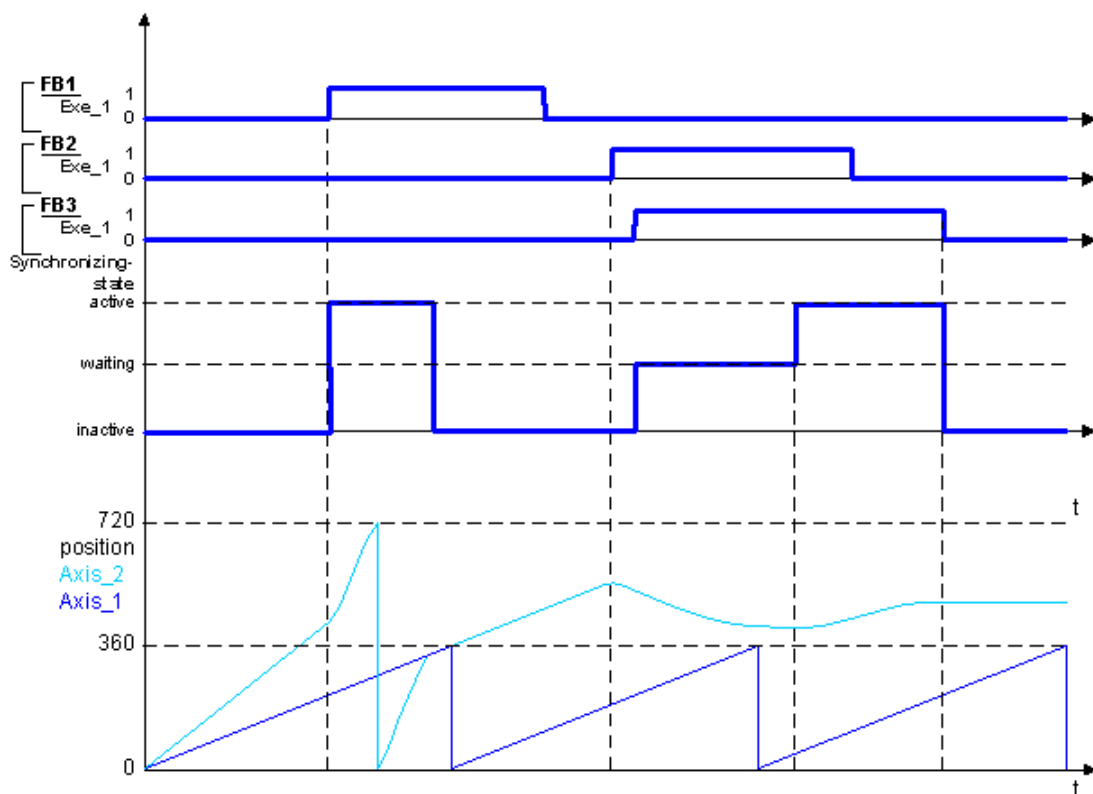
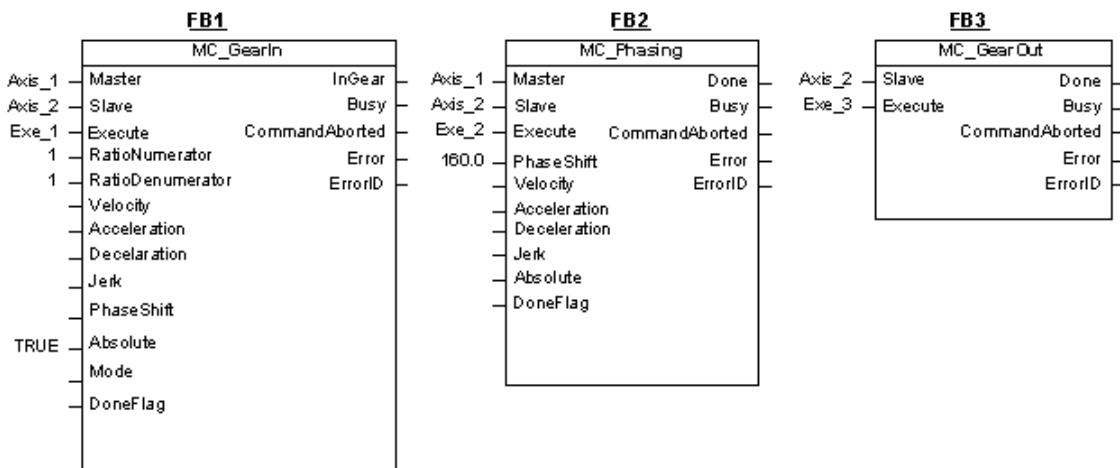
例如，两个轴都以正速度移动。启动 1:1 比率的绝对传动和立即同步。当保持跟随轴的当前运动方向时，绿色曲线显示其路线。另一方面，红色曲线显示反转方向的同步。



“正同步方向”的示例

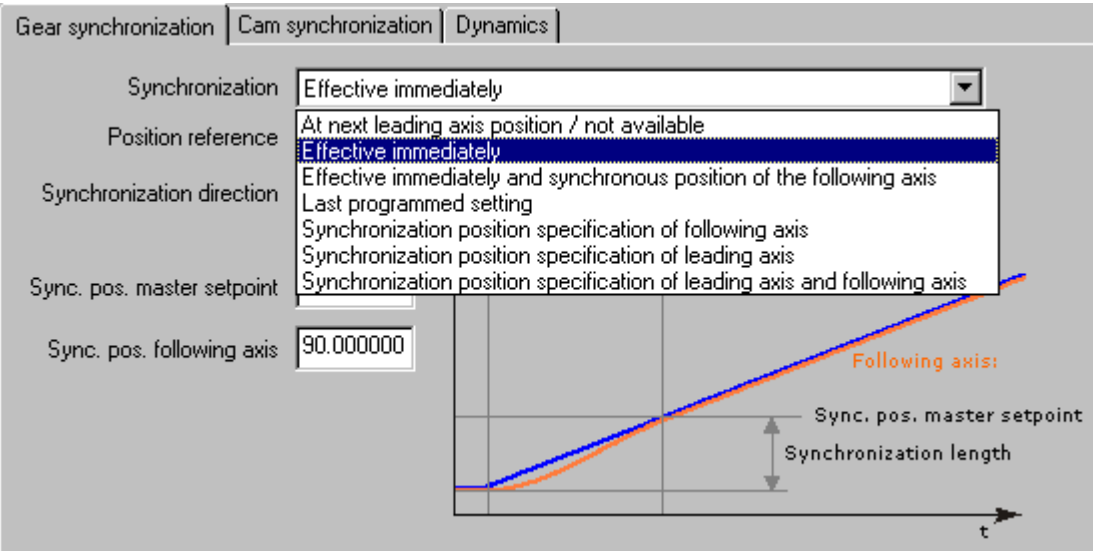
(Gearingsettings.synchronizingdirection = POSITIVE_DIRECTION)

下一个示例显示仅允许沿正方向移动的同步/去同步运动的默认设置。同步/去同步的默认设置为“立即生效”(Effective immediately)。使用 *Exe_1* 立即启动同步。现在，要通过 *Exe_2* 将从站移动一个相移。此运动在负方向上执行。如果是使用 *Exe_3* 在负方向启动“立即”激活的“MC_CamOut”，则在其去同步之前，轴会一直等到其沿正方向移动。



4.7.5 同步

您可以在同步操作 (Synchronous operation) > 默认 (Default) 对话框的“齿轮同步”(Gear synchronization) 和“凸轮同步”(Cam synchronization) 选项卡中设置同步条件。

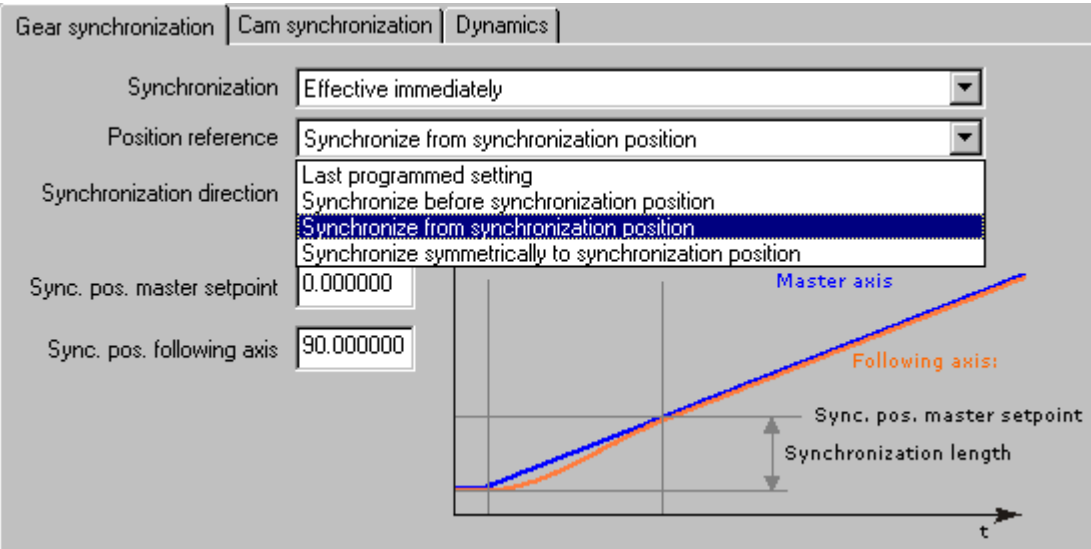


使用“同步”(Synchronization) 下拉列表框设置同步条件：

设置	含义
立即生效	启动此功能后立即执行同步。 “同步位置主设定值”(Sync. pos. master setpoint)、“同步位置跟随轴”(Sync. pos. following axis) 和“位置参考”(Position reference) 中的设置未进行评估。
引导轴的默认同步位置	同步条件取决于引导轴的位置。在“同步位置主设定值”(Sync. pos. master setpoint) 中定义同步位置。 “同步位置跟随轴”(Sync. pos. following axis) 中的设置被忽略。
跟随轴的默认同步位置 (仅具有传动)	同步条件取决于跟随轴的位置。在“同步位置跟随轴”(Sync. pos. following axis) 中定义同步位置。 “同步位置主设定值”(Sync. pos. master setpoint) 中的设置被忽略。
在活动凸轮盘结束时转换 (仅具有凸轮传动)	这只能用相关的引导轴参考来设置。 同步条件是主设定值位置在当前凸轮盘周期的末尾。 “同步位置跟随轴”(Sync. pos. following axis) 中的设置被忽略。

设置	含义
引导轴和跟随轴的默认同步位置	同步条件取决于引导轴的位置。在“同步位置主设定值”(Sync. pos. master setpoint) 中定义同步位置。 另外，作为“同步位置跟随轴”(Sync. pos. following axis) 中设置的结果，在跟随轴中会产生偏移，即不是使用编程的（例如通过凸轮盘）跟随位置进行同步，而是使用“同步位置跟随轴”(Sync. pos. following axis) 位置加上跟随轴离凸轮盘的绝对位置值来同步跟随轴。 示例： 跟随轴通过凸轮盘与主轴耦合。 已创建凸轮盘，以便当引导轴在位置 50 时将跟随轴设置为位置 70。 如果参数“同步位置跟随轴”(Sync. pos. following axis) = 80，则当引导轴在位置 50 时跟随轴在位置 150 (70 + 80) 同步。 使用凸轮盘的耦合仍处于活动状态（缩放比例保持不变），但是偏移量为 80。
立即生效和从轴的同步位置	启动此功能后立即执行同步。在此功能启动后，引导从轴和跟随从轴结果之间生成的相移根据跟随轴的位置得出。这可以补偿由跟随轴加速度斜坡导致的偏移量。同步运动基于动态响应默认值。 “同步位置主设定值”(Sync. pos. master setpoint) 中的设置被忽略。
上一个已编程的设置	不可用

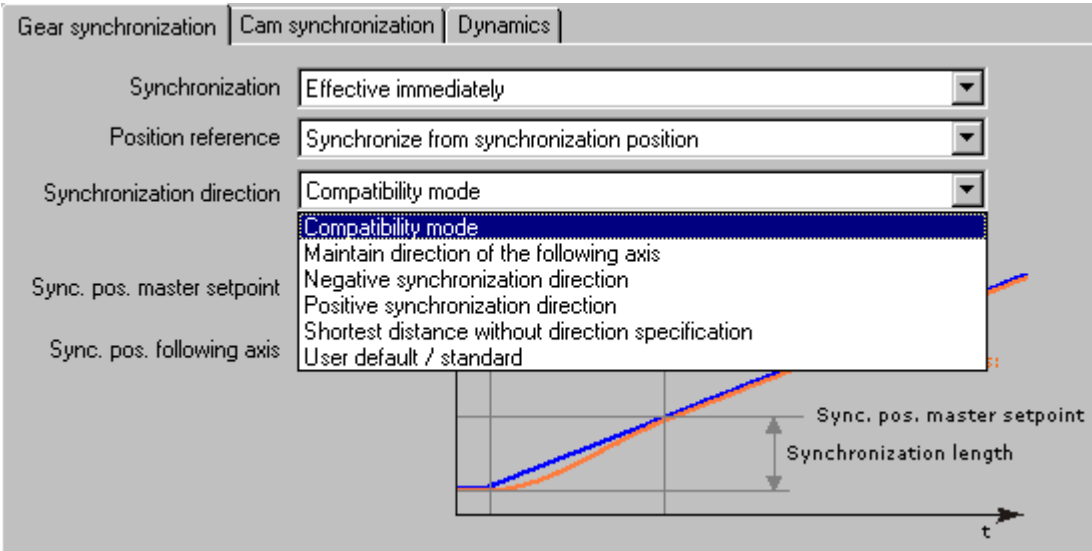
位置参考



在“位置参考”(Position reference) 下拉列表框中定义同步模式：

设置	含义
从同步位置同步	在同步位置开始同步。同步长度从动态数据中与时间相关的同步曲线中得出，适用于“同步长度”(Synchronization length) 参数中与引导轴相关的同步曲线。在超过此长度后会同步跟随轴。
在同步位置之前同步	在该同步模式下，跟随轴会在到达同步位置时与引导轴同步。 具有与时间相关的同步曲线的同步开始位置取决于动态数据，而具有与引导轴相关的同步曲线的同步开始位置取决于“同步长度”(Synchronization length) 参数。
在与同步位置对称时同步	在该模式下，在同步长度的中间达到同步。同步移动在达到同步位置之前开始，并在超过该位置之后停止。 根据与同步长度或动态默认值相关的曲线执行同步。
上一个已编程的设置	不可用

同步方向

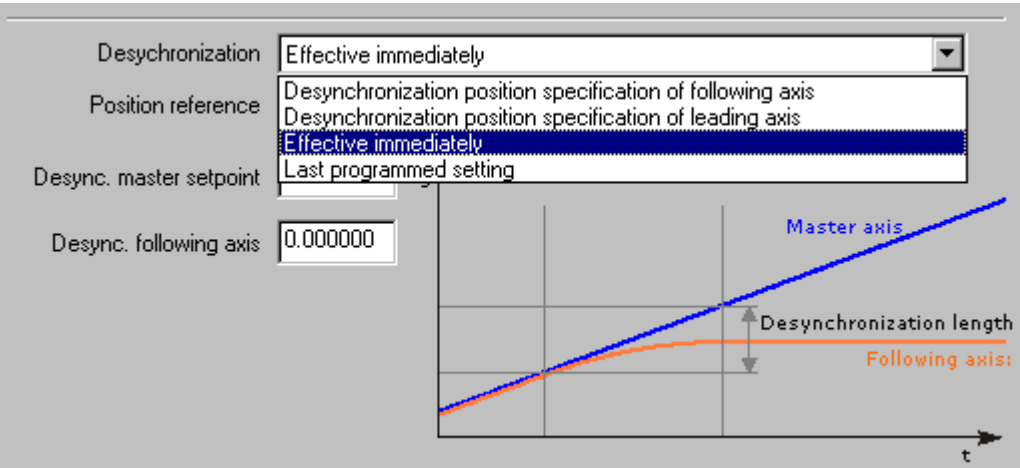


同步的方向取决于“同步方向”(Synchronization Direction) 下拉列表框中的设置。此设置仅适用于模数轴。

设置	含义
与主轴方向相同	同步始终发生在跟随轴的运动方向上。
兼容性模式	齿轮传动： 跟随轴在正向上同步 凸轮传动： 跟随轴将在最短的距离内同步。
没有预置方向的最短路径	从轴在最短路径上同步。
负同步方向	此轴始终在负方向同步。
正同步方向	此轴始终在正方向同步。

4.7.6 去同步

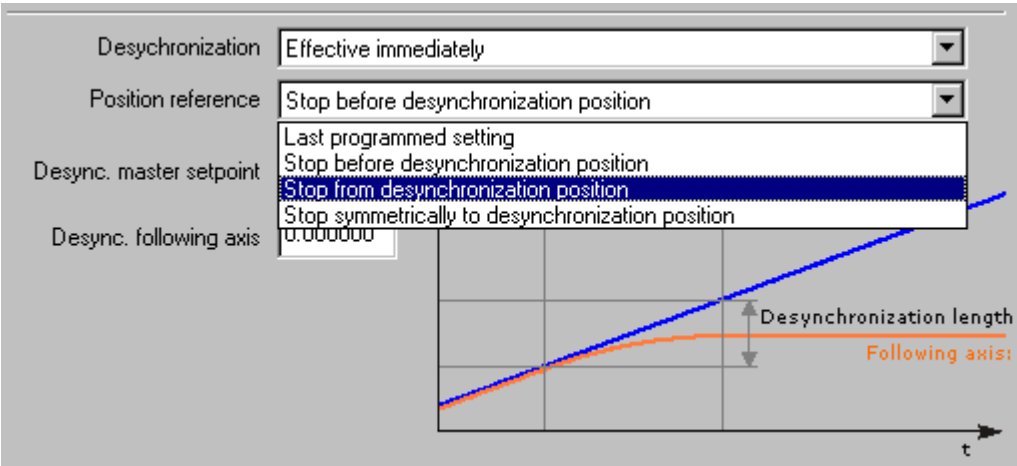
您可以在**同步操作 (Synchronous operation) > 默认 (Default)** 对话框的“齿轮同步”(Gear synchronization) 和“凸轮同步”(Cam synchronization) 选项卡中设置去同步条件。



在“去同步”(desynchronization) 下拉列表框中设置去同步条件。

设置	含义
立即生效	启动此功能后立即启动去同步。 “去同步主设定值”(DeSync. master setpoint)、“去同步跟随轴”(Desync. following axis) 和“位置参考”(Position reference) 中的设置被忽略。
跟随轴的默认去同步位置	根据在“去同步跟随轴”(Desync. following axis) 中设置的跟随轴位置来启动去同步。 “去同步引导轴”中的设置被忽略。
由引导轴的去同步位置指定	根据在“去同步主设定值”(DeSync master setpoint) 中设置的引导轴位置来启动去同步。 “去同步跟随轴”中的设置被忽略。
凸轮盘周期的末尾 (仅具有凸轮传动)	去同步在当前凸轮盘周期的末尾执行。
上一个已编程的设置	不可用

位置参考



在“位置参考”(Position reference) 下拉列表框中设置去同步模式:

设置	含义
在去同步位置之后停止	在去同步位置启动去同步。 去同步长度从基于时间的同步曲线的动态数据中得出，并从与引导轴相关的同步曲线的“去同步长度”(Desynchronization length)参数中得出。 在超过此长度后启动去同步。
在与去同步位置对称时停止	在该模式下，当轴到达去同步长度中间时启动去同步。 去同步运动在去同步位置之前启动，并在超过此位置后停止。 根据与去同步长度或动态默认值相关的曲线执行去同步。
在去同步位置之前停止	在该模式下，当轴到达去同步位置时启动去同步。 具有与时间相关的同步曲线的去同步开始位置取决于动态数据，而具有与引导轴相关的同步曲线的同步开始位置取决于“同步长度”(Synchronization length) 参数。
上一个已编程的设置	不可用

4.7.7 同步状态

以下部分说明了 S7T Config 可用于查看同步状态的系统变量：

同步对象中的系统变量：

- **enablecommand**
此系统变量用于指明在同步对象中是启动“MC_GearIn”命令还是“MC_CamIn”命令。指明了以下状态：
 - *ACTIVE* — 命令处于活动状态
 - *NONE* — 没有命令处于活动状态
 - *WAITING_TO_START* — 命令等待同步条件
- **disablecommand**
此系统变量用于指明在同步对象中是启动“MC_GearOut”命令还是“MC_CamOut”命令。指明了以下状态：
 - *ACTIVE* — 命令处于活动状态，跟随轴正在去同步
 - *NONE* — 没有命令处于活动状态
 - *WAITING_TO_START* — 命令等待去同步条件
- **syncstate**
此系统变量用于返回同步状态。指明了以下状态：
 - *NO* — 同步操作尚未同步。
 - *YES* — 同步操作已同步。
- **state**
此系统变量用于指明同步对象中的活动同步模式。指明了以下状态：
 - *CAMMING* — 曲线同步处于活动状态
 - *INACTIVE* — 没有同步操作处于活动状态

跟随轴上的系统变量：

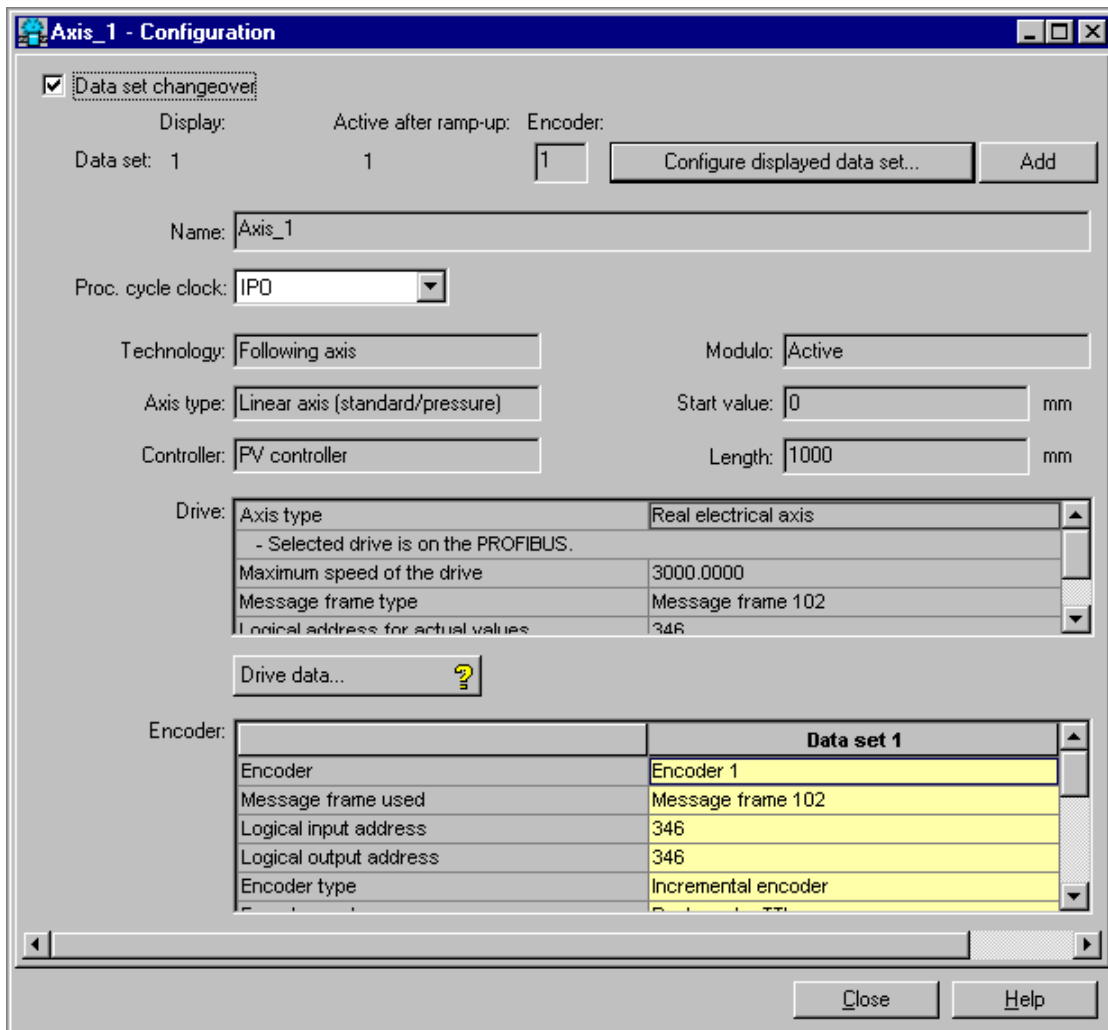
- **syncmonitoring.differencecommandvalue**
此系统变量用于指明引导轴和跟随轴之间的主设定值之差。

4.7.8 组态

4.7.8.1 组态 — 轴

轴 (Axis) > 组态 (Configuration) 对话框显示了轴和驱动器设置

可以通过单击“组态活动数据集”(Configure active data set) 在对话框中编辑这些设置。



该对话框还包含用于管理数据记录转换的数据记录的功能。如果工艺对象支持多个数据记录，则您可以：

- 创建新数据记录
- 删除数据记录
- 定义在 CPU 启动期间为工艺对象装载哪个数据记录，以及
- 重新组态选定的数据记录。

4.7.9 机械系统

4.7.9.1 机械 — 电气轴

轴 (Axis) > 机械 (Mechanics) 对话框用于组态轴和编码器的机械参数。

Measuring system:

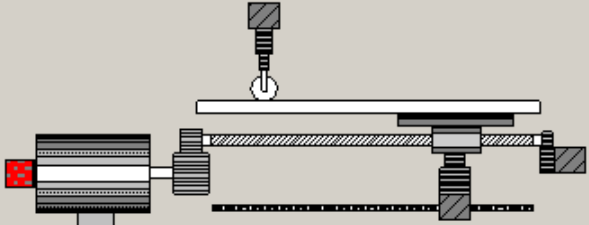
☐ Meas. system in opposite sense (invert act. pos. value) An inversion of both values corresponds to an inversion of the drive direction.

☐ Invert setpoint

Rotary encoder system

Encoder parameter...

Mechanics:



Mounting of encoder: Motor side

Check gear ratio? ?

Load gear

Number of motor revolutions: 1

Number of load revolutions: 1

Measuring gear

Number of motor revolutions: 1

Number of encoder revolutions: 1

Leadscrew pitch

Dist. per spindle rev.: 10.0 mm/rot

☐ Modulo axis

☐ Backlash on reversal compensation:

轴的设置:

- 驱动器方向设置
- 装载齿轮设置
- 线性轴的丝杠螺距
- 模数设置
- 间隙补偿设置

编码器的设置:

- 测量齿轮设置
- 间隙补偿
- 编码器安装类型

模数轴的机械设置的边界条件（长期稳定性）

使用模数轴，可以执行检查以确定是否保证长期稳定性。长期稳定性可确保编码器测量的位置与实际值的内部表示始终同步。这样能使位置精确逼近，甚至在出现多个模数溢出之后。如果不能保证长期稳定性，则在一致性检查期间工程系统中会输出以下错误消息之一：

- 无法表示组态的传动比 (Configured gear ratio cannot be represented)
- 无法表示组态的模数长度 (Configured modulo length cannot be represented)

运行时还会对目标设备执行长期稳定性测试。如果由于组态无法确保此测试，则 S7T Config 中会发出以下报警：“报警 20006 组态错误，原因 3041”(Alarm 20006 Configuration error, reason 3041)。

此错误的原因是组态数据中选择的值不当。这些值必须满足以下条件：

计算模数旋转轴的 f_1

$f_1 = \text{测量齿轮分子} \times 360 \times \text{内部精度} \times \text{负载旋转次数}^1)$

计算模数线性轴的 f_1

$f_0 = \text{TRUNC}(\text{丝杠螺距} \times \text{内部精度})^2)$

舍掉 f_0 的小数位。必须以基本单位 mm 指定丝杠螺距。

$f_1 = \text{测量齿轮分子} \times f_0 \times \text{负载旋转次数}^1)$

计算 f_2

$f_2 = \text{测量齿轮分母} \times \text{编码器精度} \times \text{实际值因子} \times \text{电机旋转次数}^1)$

计算 f_{11} 和 f_{22}

必须从 f_1 和 f_2 确定最大公约数 k ，并将其用于以下公式：

$f_{11} = (f_1 / k) < 2^{31}$

$f_{22} = (f_2 / k) < 2^{31}$

f_{11} 和 f_{22} 的结果必须小于 2^{31} 。否则，请检查相应修改 f_1 和 f_2 公式中的参数后 f_{11} 和 f_{22} 的结果是否会小于或等于允许的最大值。

计算 f_{31}

如果 f_{11} 和 f_{22} 满足上述要求，但下载操作仍中止，且显示 S7T Config 消息“组态错误 20006 原因 3041”(Configuration error 20006 reason 3041)，则可执行以下测试：

$f_3 = \text{TRUNC}(\text{模数长度} \cdot \text{内部精度})$

舍掉 f_3 的小数位。对于线性轴和旋转轴度数，模数长度必须以 mm 指定。

现在可以计算 f_3 和 f_{11} 的最大公约数 k_2 ，并将其插入以下公式中：

$f_{31} = (f_3 \times f_{22} \times f_{11}) / (k_2 \times k_2) < 2^{62}$

此时，必须检查 f_{31} 是否小于 2^{62} 。如果大于 2^{62} ，请检查是否可以减小模数长度。也可以更改 f_1 和 f_2 公式中的参数，只要确保仍然符合对 f_{11} 和 f_{22} 的要求即可。

1)	对于负荷端或外部编码器安装，在公式中，电机旋转次数和负载旋转次数都插入 1。
2)	对于外部编码器安装，在丝杠螺距位置插入组态的每转距离。

参数说明

参数	注释/轴上的组态数据元素
测量齿轮分子	测量齿轮比率的分子 <i>TypeOfAxis.NumberOfEncoders.Encoder_1.AdaptDrive.numFactor</i> <i>TypeOfAxis.NumberOfEncoders.Encoder_1.AdaptExtern.numFactor</i> <i>TypeOfAxis.NumberOfEncoders.Encoder_1.AdaptLoad.numFactor</i>
测量齿轮分母	测量齿轮比率的分母 <i>TypeOfAxis.NumberOfEncoders.Encoder_1.AdaptDrive.denFactor</i> <i>TypeOfAxis.NumberOfEncoders.Encoder_1.AdaptExtern.denFactor</i> <i>TypeOfAxis.NumberOfEncoders.Encoder_1.AdaptLoad.denFactor</i>
电机旋转次数	电机旋转次数的分子 <i>TypeOfAxis.NumberOfDataSets.DataSet_1.Gear.numFactor</i>
负载旋转次数	负载旋转次数的分子 <i>TypeOfAxis.NumberOfDataSets.DataSet_1.Gear.denFactor</i>
内部精度	内部增量/位置单位 已使用轴向导在组态中进行了定义。
编码器精度	每转编码器脉冲数（在编码器上指定） <i>TypeOfAxis.NumberOfEncoders.Encoder_1.IncEncoder.incResolution</i> <i>TypeOfAxis.NumberOfEncoders.Encoder_1.AbsEncoder.absResolution</i>
倍增因子，实际值 （实际值因子）	= X ¹⁾ <i>IncEncoder.incResolution MultiplierCyclic</i> <i>TypeOfAxis.NumberOfEncoders.Encoder_1.IncEncoder.incResolution MultiplierCyclic</i> <i>TypeOfAxis.NumberOfEncoders.Encoder_1.AbsEncoder.absResolution MultiplierCyclic</i>
模数长度	<i>Modulo.length</i>
轴每转的丝杠螺距	<i>LeadScrew.pitchVal</i>
每转路径	<i>TypeOfAxis.NumberOfEncoders.Encoder_1.pathPerResolution.length</i>

1)	X=0	对于高精度，根据“使用 PROFIdrive 消息帧的编码器接口”部分（PROFIdrive 轴消息帧/高精度中的编码器 (Gn_XIST1)）中介绍的 SIMATIC S7-Technology 中的高精度默认设置表来考虑默认值。（在这种情况下，仅对目标设备执行长期稳定性测试。在工程系统中的一致性测试期间不检查此项。）
	X<>0	将组态值用于高精度。

4.7.10 默认值

4.7.10.1 缺省 —“动态”(Dynamics) 标签

可在轴 (Axis) > 默认 (Default) 对话框的“动态”(Dynamics) 选项卡中设置轴动态的默认值。如果在工艺功能中设置了负的动态值，则会激活默认值。

Dynamic response

Direction: Positive

Velocity: 100.0 mm/s

Velocity profile: Last programmed velocity profile

Jerk: 1000000.0 mm/s³

Acceleration: 1000.0 mm/s²

Jerk: 1000000.0 mm/s³

Jerk: 1000000.0 mm/s³

Deceleration: 1000.0 mm/s²

Jerk: 1000000.0 mm/s³

Stopping time: 0.0 s

支持的默认值:

- 方向 (Direction)
- 速度 (Velocity)
- 加速度 (Acceleration)
- 减速度 (Deceleration)
- 冲击 (Jerk)
- 速度曲线 (Velocity profile)

停止时间

禁用移动轴且在“MC_Power”将 *Stopmode* 设置为 2 时，“停止时间”(Stop time) 中设置的时间便会生效。

速度曲线 (Velocity profile)

速度曲线定义启动期间和制动时的响应，以及对速度变化的响应。

工艺功能不使用默认值。如果要更改速度曲线，请使用 *Jerk* 输入参数。

可以在以下曲线之间进行选择：

- **梯形**

梯形曲线用于正方向和负方向运动的线性加速度（突变化 = 0）。

- **常量**

该曲线显示常量加速度，突变化曲线是可控制的（突变化 $\neq 0$ ）。

4.7.11 限制

4.7.11.1 限制 —“位置和速度”(Position and velocity) 标签

在轴 (Axis) > 限制 (Limits) 选项卡的“位置和速度”(Position and velocity) 选项卡中，您可以

- 组态和启用对硬件限位开关的监视
- 组态和启用对软件限位开关的监视
- 设置速度限制
- 设置急停减速度

Position and velocity
Dynamic response
Fixed endstop

Hardware limit switch
☒ Active

Negative log. address

Bit number:

Positive log. address

Software limit switch
☒ Active

mm

mm

There are two different limits:

- Limits which must never be exceeded because of mechanical reasons.
- Limits which can be modified by programs.

Max. velocity
 mm/s

Pos. prog. velocity
 mm/s

☒ Absolute values

Emergency stop delay: mm/s²

急停减速度

禁用移动轴且通过“MC_Power”参数将 *Stopmode* 设置为 0 时，将激活为急停减速度设置的值。

4.7.11.2 硬件限位开关

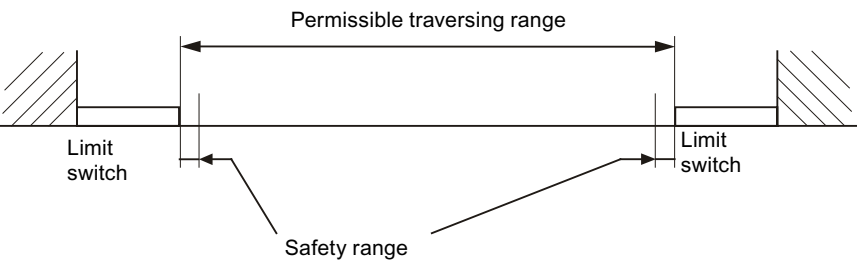
在**轴 (Axis) > 限制 (Limits)** 对话框的“位置和速度”(Position and Velocity) 选项卡中启用硬件限位开关监视。硬件限位开关监视用于限制轴的运动范围或保护机器。

接线

可将硬件限位开关连接至 T-CPU 的四个集成数字输入端，或者通过在 DP(DRIVE) 上操作的 TM15/TM17 连接至 I/O 模块（如 ET 200 或 SINAMICS S120）。

行程范围

通过硬件限位开关的数字输入监视允许的行程范围。



注意
硬件限位开关必须作为 NC 触点实施。 轴通过允许的行程范围后，硬件限位开关必须保持活动状态，直到达到机械终点位置。

退回

触发硬件限位开关的轴停止，并显示错误消息 **8013** 和 **804B**。该轴从硬件限位开关退回（释放运动），如下所述：

- **手动退回**

手动使该轴返回允许的行程范围。只有该轴返回该范围后，**工艺 DB** 的错误才能被确认。

- **通过驱动器退回**

该轴的 **工艺 DB** 的错误已被确认，但是错误消息和 **LimitSwitchActive** 位仍保持激活。现在，该轴可以返回到允许的行程范围。反转运动命令将再次触发轴错误。该轴移动到限位开关的范围之外后，便可确认错误消息和 **LimitSwitchActive** 状态。

保存到达硬件限位开关后，即可保存轴的当前位置。只有在通过此位置（加上一个安全范围）之后，该轴才被视为已离开限位开关。

注意
为避免硬件限位开关的极性监视和硬件限位开关的超程监视在有效范围方向上发生冲突，在该轴通过硬件限位开关后 不能 关闭控制器。在这种情况下，该轴将移至有效范围之内（无硬件限位开关监视），然后重新启用。 打开控制器时，该轴必须位于有效行程范围之内。

轴通过硬件限位开关时，内部状态会丢失并且将重新装载组态。只有在有效范围内重新装载时才不会丢失逼近信息。

例外情况： 发生极性反向错误后取消激活限位监视

安全范围

根据轴单位系统的已组态精度计算硬件限位开关的安全范围。

安全范围 = 1000/（增量/位置）

示例： 在“组态单位”(Configure units) 对话框中，将线性轴的位置单位指定为“mm”，将精度（增量/位置）指定为“1000/单位”，这意味着轴位置的计算精确到 0.001 mm。本示例中的安全范围是精度的 1000 倍： 1 mm。

4.7.11.3 软件限位开关

可以在**轴 (Axis) > 限制 (limits)** 对话框的“位置和速度”(Position and Velocity) 选项卡中组态软件限位开关并启用监视功能。如果激活软件限位开关，则通过软件限位开关限制轴的行程距离。

例如，软件限位开关应位于硬件限位开关的行程范围内，以限制轴的工作范围。

运动开始时监视软件限位开关：

- **复选框已激活**

如果位置控制运动命令的目标位置在软件限位开关之外，则运动开始时轴的工艺 DB 中就已显示警告 **0026**。在 **S7T Config** 中，报告中断 **40105**。
轴最多行进到软件限位开关位置，并在工艺 DB 中报告错误 **8014**。

- **复选框未激活**

如果位置控制运动命令的目标位置在软件限位开关之外，则轴来回移动，直到到达软件限位开关的 **p** 位置。在工艺 DB 中报告错误 **8014**（不输出警告 **0026**）。

行进到软件限位开关的特性：

- **对于位置受控的来回移动：**

轴来回移动，直到到达软件限位开关。到达软件限位开关后，保持各自活动的位置控制模式或速度控制模式

- **在所有操作模式中：**

轴来回移动，直到到达软件限位开关。到达软件限位开关后，保持活动的位置控制模式。在速度控制模式中，轴变为位置控制模式。

负位置/正终点位置：

在这些输入域中输入软件限位开关的负位置和正终点位置。

用于退回的公差窗口：

在此输入域中输入适合的值，以防止退回期间重新开始触发软件限位开关错误。

说明

软件限位开关的响应由轴组态根据回原点功能确定。如果轴尚未回原点时，则不监视组态中要求使用已回原点轴来执行绝对运动命令（轴 **[Axis] > 回原点 [homing]** 对话框中的“需要回原点: 是”[Homing required: Yes]）的限位开关。反之，如果在轴未回原点时允许执行绝对运动命令（轴 **[Axis] > 回原点 [homing]** 对话框中的“需要回原点: 否”[Homing required: No]），则无论轴的回原点状态如何，都将监视软件限位开关。

4.7.11.4 限制 —“动态响应”(Dynamic response) 标签

可以在**轴 (Axis) > 限制 (Limits)** 对话框的“动态响应”(Dynamic response) 选项卡中调整 HW 限制（机械）和 SW 限制。

禁用移动轴且在“MC_Power”参数将 *FastStop* 设置为 1 时，激活设定的减速度限制。

默认情况下，可以通过设置“绝对值”(Absolute values) 复选框启用**取决于状态的**加速度模型。

可以通过重设“绝对值”(Absolute values) 复选框启用**取决于方向的**加速度模型。 在这种情况下会显示其他一些附加参数。

The screenshot displays the 'Dynamic response' tab of the 'Limits' dialog box. It is divided into two main sections: 'Hardware limits' and 'Software limits'.

Hardware limits (limits that must not be exceeded for mechanical reasons):

- Acceleration: 10000.0 mm/s²
- Jerk: 200000.0 mm/s³

Software limits (limits that can be modified by programs):

☒ Absolute values

Acceleration:

- In positive dir. of travel: > 1000000000000.0 mm/s²
- In negative dir. of travel: > 1000000000000.0 mm/s²

Jerk:

- Positive jerk: In positive dir. of travel: > 1000000000000.0 mm/s³
- Negative jerk: In positive dir. of travel: > 1000000000000.0 mm/s³

The graphs show the acceleration (a) and jerk (j) profiles over time (t). The acceleration graph shows a smooth curve, and the jerk graph shows a step function. The 'Local stop reaction with jerk' checkbox is unchecked.

取决于状态的加速度

- **加速度 (Acceleration)**
轴加速度，与运动方向无关
- **减速度 (Deceleration)**
轴减速度，与运动方向无关

取决于方向的加速度

- **加速度 (Acceleration)**
正运动方向的加速度和负运动方向的减速度
 - **减速度 (Deceleration)**
负运动方向的加速度和正运动方向的减速度
- 例如，动态方向矢量的参数设置对于悬挂轴很有用。

带突变化的局部停止响应

设置“带突变化的局部停止响应”(Local stop reactions with jerk) 复选框，以执行由轴上的报警响应触发的停止操作，带有突变化限制和倒圆功能。

对于预编程减速度停止（仅当使用真实轴时输入框才可见）

在此输入框中输入斜坡设置的减速度。

用于平滑由控制器转换产生的受控变量变化的时间常量 （仅当使用真实轴时输入框才可见）

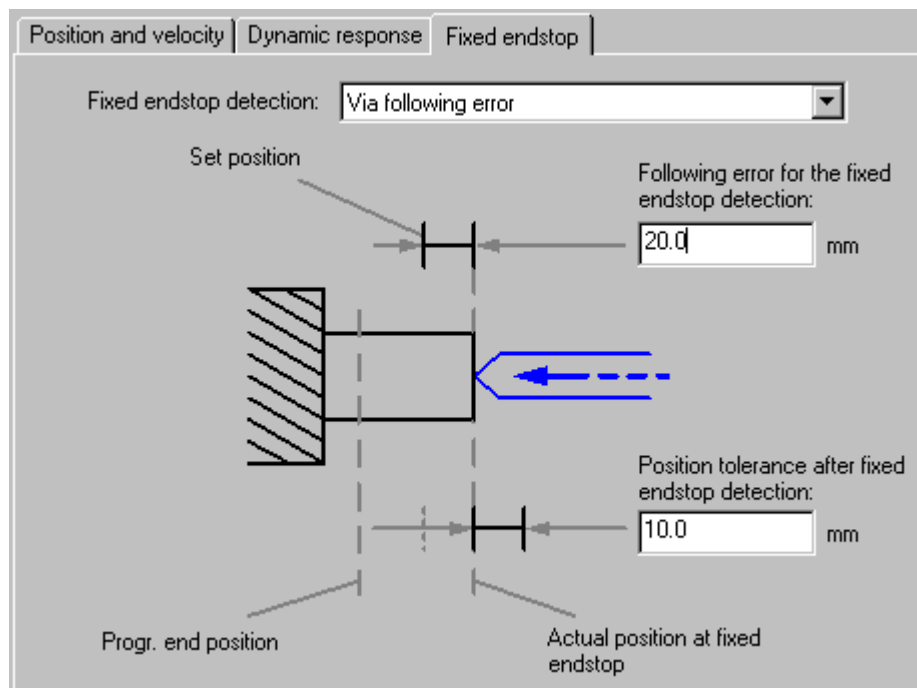
在此处输入用于平滑由控制器转换产生的受控变量变化的时间常量。对于会由于转换引起受控变量偏移的所有状态跳转/转换，此转换平滑滤波器都处于活动状态。

4.7.11.5 限制 —“固定挡块”(Fixed end stop) 标签

轴 (Axis) > 限制 (Limits) 的“固定终点挡板”(Fixed end stop) 选项卡可用于启用固定终点挡板检测以及设置相应的检测模式：

- 使用跟随误差
- 使用力/扭矩

满足选定的条件时，将达到“固定终点挡板”(fixed end stop) 状态。如果启用了“移至固定终点挡块”(Move to fixed end stop)，将禁用跟随误差监视。



使用跟随误差

说明

当轴移至固定终点挡块且将固定终点挡板检测设置为“使用跟随误差”(use following error) 时，在“检测到固定终点挡板之后的位置公差”(Position tolerance after fixed end stop detection) 中输入的值应明显小于“固定终点挡板检测的跟随误差”(Following error for fixed end stop detection) 中的值。

使用扭矩值

固定终点挡板检测功能“使用扭矩值”(use torque value) 要求分配给轴的数字驱动器支持扭矩限制，并且需要设置相应的消息帧，例如 *消息帧 102* 或 *消息帧 105*。

移至固定终点挡块

“MC_MoveToEndPos”功能可以在达到终点挡板后激活“移至终点挡板”(Move to end stop) 功能并设置紧固扭矩。该操作也称为“夹紧”。

运动将在轴到达固定终点挡板时停止，而控制仍然保持活动状态。位置控制器输入的设定值保持为常量。钳位位置方向上的新运动命令将被取消；将在退回方向上执行新运动控制命令以减少扭矩。该轴的位置设定值用作新运动控制命令在退回方向上的起始位置。

轴位置设定值由以下其中一个等式产生，具体取决于固定终点挡板检测功能：

- “使用跟随误差”
在固定终点挡板的位置 + 跟随误差
- “使用扭矩值”
在固定终点挡板的位置 + 夹紧公差

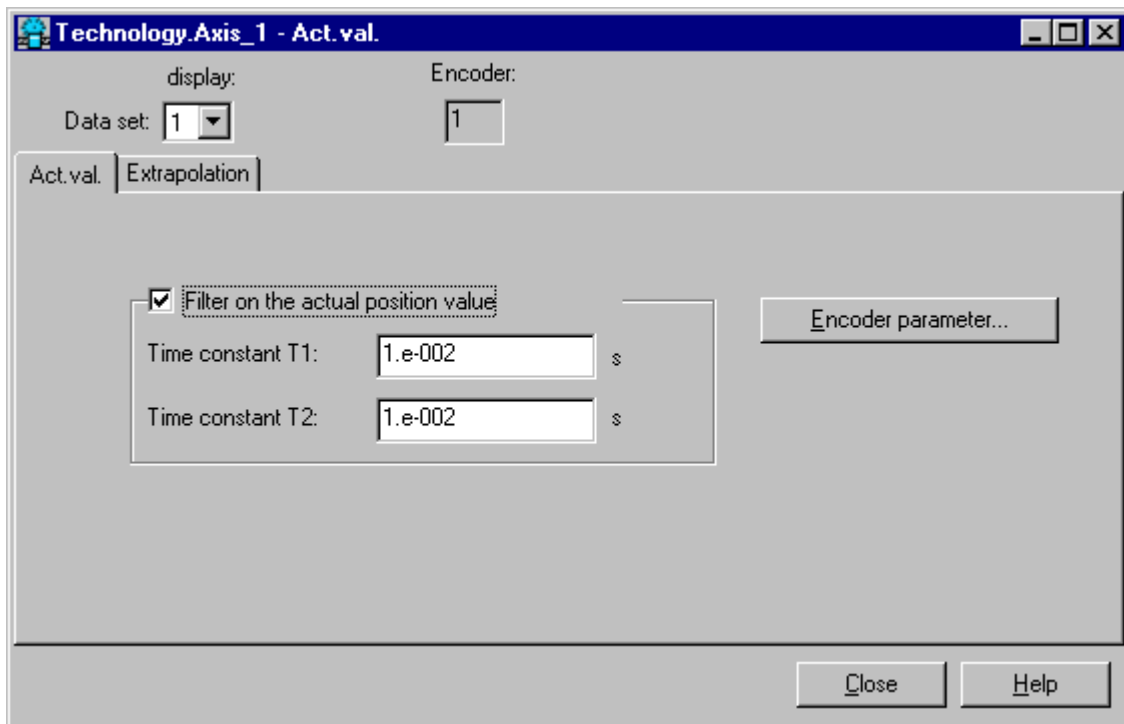
“固定终点挡板检测”功能的条件

- 当轴移至钳位公差窗口外时，将复位“移至固定终点挡块”(Move to fixed end stop) 功能。
- 还可以输出新命令，以在钳位处于活动状态时切换扭矩的方向。
- 可在用户程序中，在已定义的时间段上执行非步进扭矩跳转和扭矩保持，就像扭矩曲线的定义一样。
- 可以通过设置反向定位命令禁用移至固定终点挡块（钳位）。
- 不允许反向命令 MC_MoveToEndPos，将忽略该命令。
- 可以通过轴的实际值监视终点挡板的机械制动（钳位公差窗口监视）。
- 通过 MC_MoveToEndPos 工艺功能的 *Torque* 参数，以 [N/m] 为单位设置驱动器的扭矩限制。
- 如果该命令处于活动状态且未检测到固定终点挡板，则系统会像对待活动的扭矩限制那样作出反应。

4.7.12 实际值

4.7.12.1 实际值 (Actual value) —“实际值”(Actual value) 选项卡

在“轴/外部编码器”(Axis/External encoder) > “实际值”(Actual value) 对话框中的“实际值”(Actual value) 选项卡上激活实际值过滤，并设置相应的时间常量。



过滤实际位置值 (Filter on the actual position value)

如果想要激活实际位置值过滤，则激活该复选框。

时间常量 T1 (Time constant T1)

此处设置实际值系统中 PT2 位置过滤器的时间常量 T1。

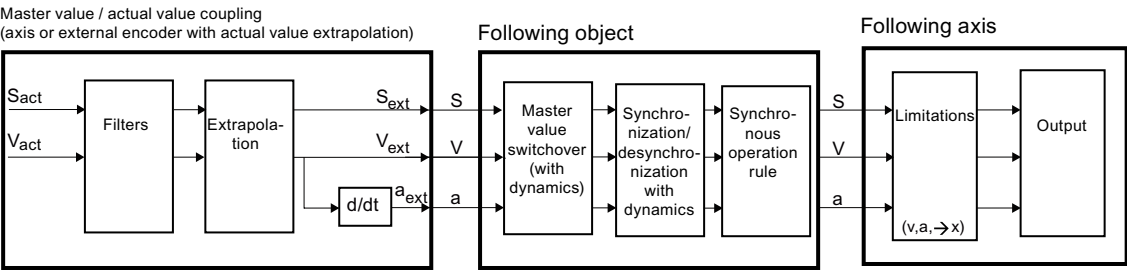
时间常量 T2 (Time constant T2)

此处设置实际值系统中 PT2 位置过滤器的时间常量 T2。

“编码器参数”(Encoder parameter) 按钮

打开一个对话框，其中显示编码器数据。

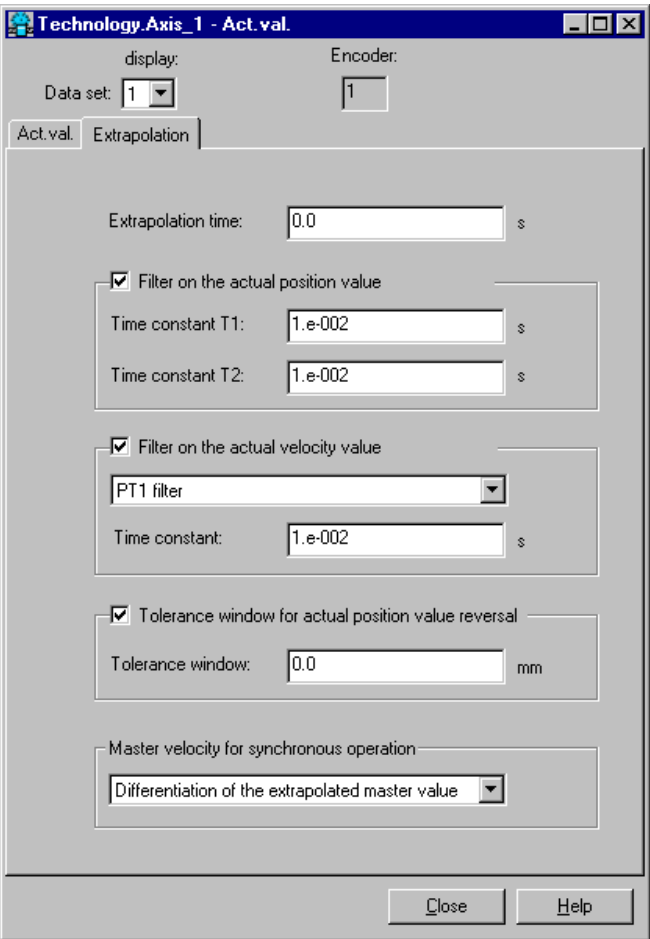
4.7.12.2 实际值 (Actual value) —“推断”(Extrapolation) 选项卡



如果控制内有同步操作互连，则同步操作会考虑主值位置的位置、速度和加速度。

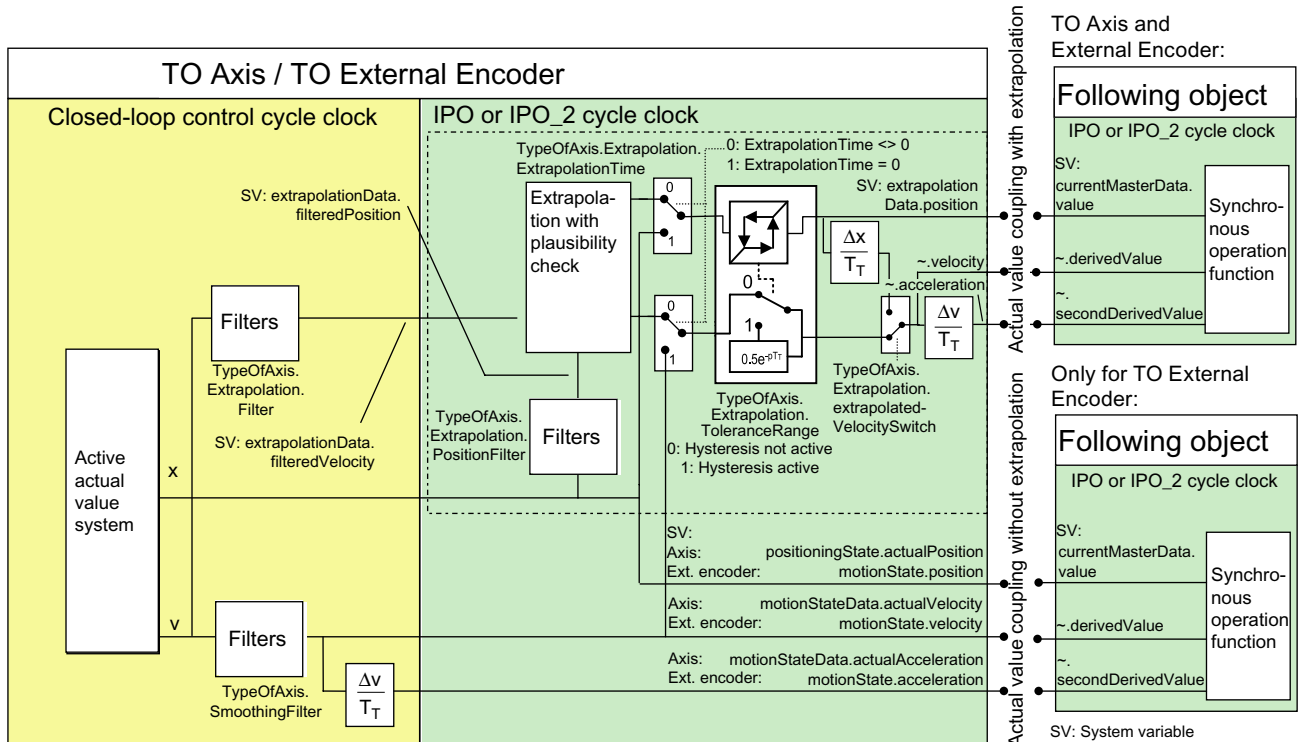
如果实际编码器值用作主值，则推断同步操作的测量实际值以补偿死区时间是很用的。死区时间是在测量实际值时在系统内产生，例如，由于总线通信和系统处理时间产生。

在 S7T Config 中，在引导轴或外部编码器上，在“轴/外部编码器”(Axis/External encoder) >“实际值”(Actual value) 对话框的“推断”(Extrapolation) 选项卡上设置推断。



也可以通过工艺参数从用户程序读取此对话框中的参数，或将此对话框中的参数写入用户程序。

实际值结合推断（轴和外部编码器）



推断时间 (Extrapolation time)

(参数 1110; 组态数据 *TypeOfAxis.Extrapolation.ExtrapolationTime*)

此处设置推断的时间。如果输入的值为 0.0, 则无推断。

过滤实际位置值 (Filter on the actual position value)

(参数 1130 组态数据 *TypeOfAxis.Extrapolation.ExtrapolationPositionFilter.enable*)

如果想要推断实际位置值, 则激活该复选框。

时间常量 T1 (Time constant T1)

(参数 1131 组态数据 *TypeOfAxis.Extrapolation.ExtrapolationPositionFilter.T1*)

此处设置实际值系统中用于推断的 PT2 过滤器的时间常量 T1。

时间常量 T2 (Time constant T2)

(参数 1132 组态数据 *TypeOfAxis.Extrapolation.ExtrapolationPositionFilter.T2*)

此处设置实际值系统中 PT2 位置过滤器的时间常量 T2。

该过滤器作用于推断的实际位置。推断的速度取自应用平滑滤波器 (*TypeOfAxis.smoothingFilter*) 之前轴或外部编码器的实际值。

过滤实际速度值 (Filter on the actual velocity value)

(参数 1112 组态数据 *TypeOfAxis.Extrapolation.Filter.enable*)

如果想要推断实际速度值，则激活该复选框。

在此处从下拉列表中选择推断速度的过滤器。

(参数 1111 组态数据 *TypeOfAxis.Extrapolation.Filter.Mode*)

时间常量 (Time constant)

(参数 1113 组态数据 *TypeOfAxis.Extrapolation.Filter.timeConstant*)

此处输入过滤器的时间常量。

根据过滤的速度值或平均速度值推断位置。通过“时间常量”计算平均值。

实际位置值倒置公差窗口 (Tolerance window for actual position value inversion)

(参数 1114 组态数据 *TypeOfAxis.Extrapolation.ToleranceRange.enable*)

在此处可以激活用于实际位置值倒置的公差窗口

公差窗口 (Tolerance window)

(参数 1115 组态数据 *TypeOfAxis.Extrapolation.ToleranceRange.Value*)

在此处输入公差窗口的大小。

如果主值与同步操作无法跟随的高频噪声信号叠加，则会造成超出动态响应限制或同步期间主值暂时改变方向。

在这种情况下，可定义公差窗口，以防止在跟随轴上超出动态响应限制，或防止同步期间方向改变。

同步操作的主速度 (Master velocity for synchronous operation)

(参数 1116 组态数据 *TypeOfAxis Extrapolation.extrapolatedVelocitySwitch*)

在下拉列表中，选择该速度是否应用于推断，或是否对推断的主位置值求微分。

检查推断值和过滤值 (Checking the extrapolated and filtered values)

可在以下系统变量中检查推断值和过滤值：

- *extrapolationdata.position*
- *extrapolationdata.velocity*
- *extrapolationdata.filteredposition*
- *extrapolationdata.filteredvelocity*
- *extrapolationdata.acceleration*

支持具有 NIST 评估的编码器 (Support of encoders with NIST evaluation)

使用具有 NIST 评估的编码器，速度由编码器确定，所得的速度可被编码器接受。在这种情况下，不需要由集成工艺计算速度和速率。有两种传送方法可用：

- 在 PROFIdrive 消息帧中传送

通过组态数据元素 *TypeOfAxis.NumberOfEncoders.Encoder_n.EncoderValueType = POSITION_AND_PROFIDRIVE_NIST_* 设置

- 在 I/O 区域中传送

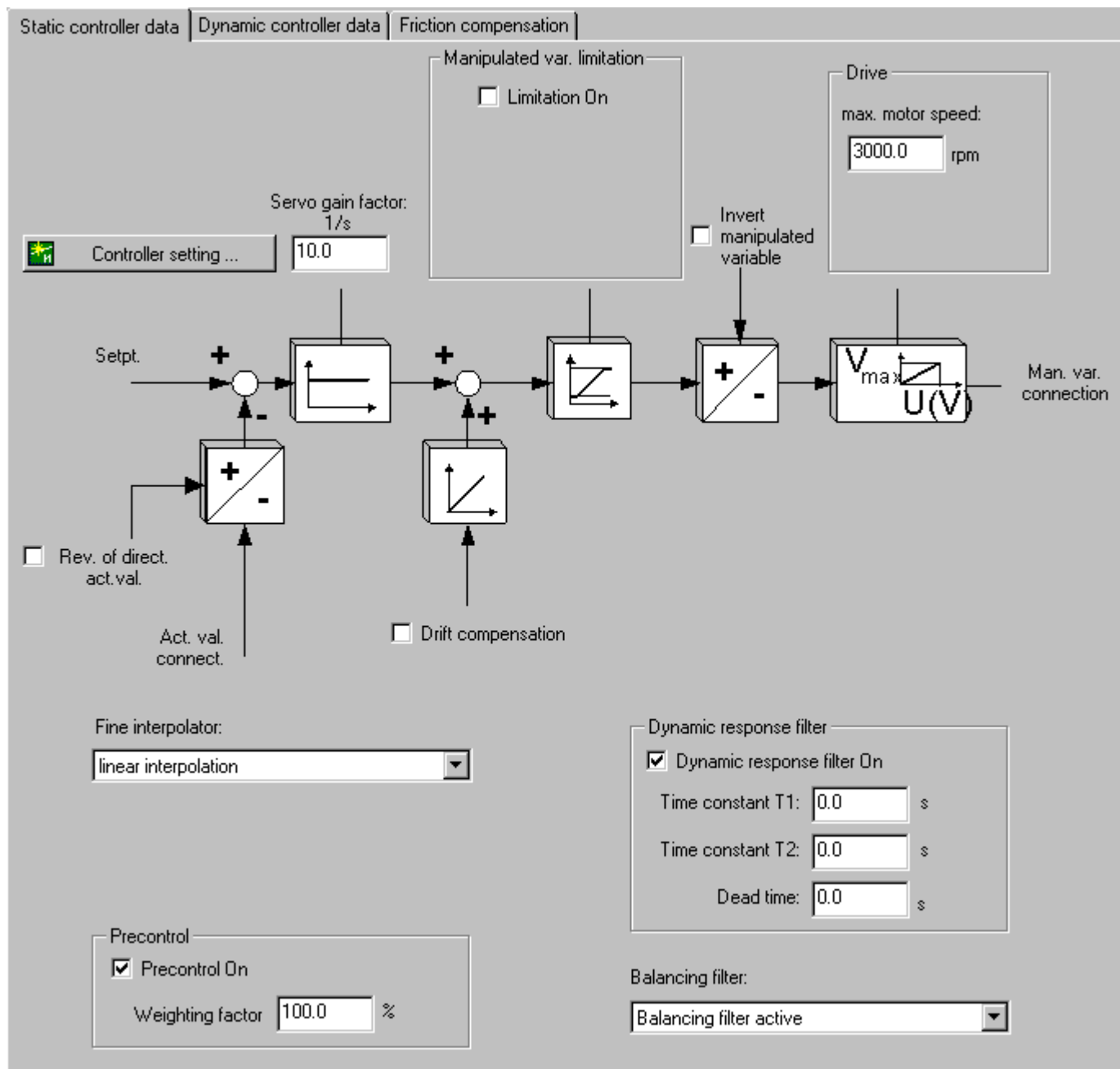
通过以下组态数据元素设置：

TypeOfAxis.NumberOfEncoders.Encoder_n.EncoderValueType = POSITION_AND_DIRECT_NIST。在这种情况下，4000H 对应于 100%。地址在组态数据元素 *TypeOfAxis.NumberOfEncoders.Encoder_n.nistConfig.logAdress* 中设置，参考值在组态数据元素 *TypeOfAxis.NumberOfEncoders.Encoder_n.nistConfig.referenceValue* 中设置。

4.7.13 控制

4.7.13.1 控制 —“静态控制器数据”(Static controller data) 标签

“轴”(Axis) > “控制”(Control) 对话框的“静态控制器数据”(Static controller data) 选项卡可用于组态轴的位置控制。



受控变量限制

受控变量限制表示控制范围的绝对上限和下限。该限制在反转之前应用。

说明

当动态伺服控制（驱动器中的位置控制器）功能处于活动状态时，间隙锁定（驱动器受控变量的限制）无效。因此，DSC 处于活动状态时，必须在驱动器中生成后退定程挡块。

驱动器

使用此输入框输入驱动器的最大速度。

“控制器设置”(Controller setting) 按钮

此按钮可用于实施简单的控制器优化，无需预控制和带有 SINAMICS 驱动器的平衡滤波器。

如果要想实现实施后的位置控制系统，请按照『优化位置控制器 — 总览』一章中的说明操作。

控制器参数

有关各控制器参数及其最佳设置的详细信息，请参考『优化位置控制器 — 总览』一章。

高精度插补器

高精度插补器功能用于在插补器和控制器具有不同的占空比时生成中间设定值。

高精度插补器上的可选插补模式设置：

- 无插补
- 线性插补（定位轴的位置恒定）
- 以恒定加速度（定位轴上加速度恒定）插补
- 以恒定速度（定位轴上速度恒定）插补

当设置为定位轴时，将插补位置设定值。

当设置为速度控制轴时，将插补速度设定值。

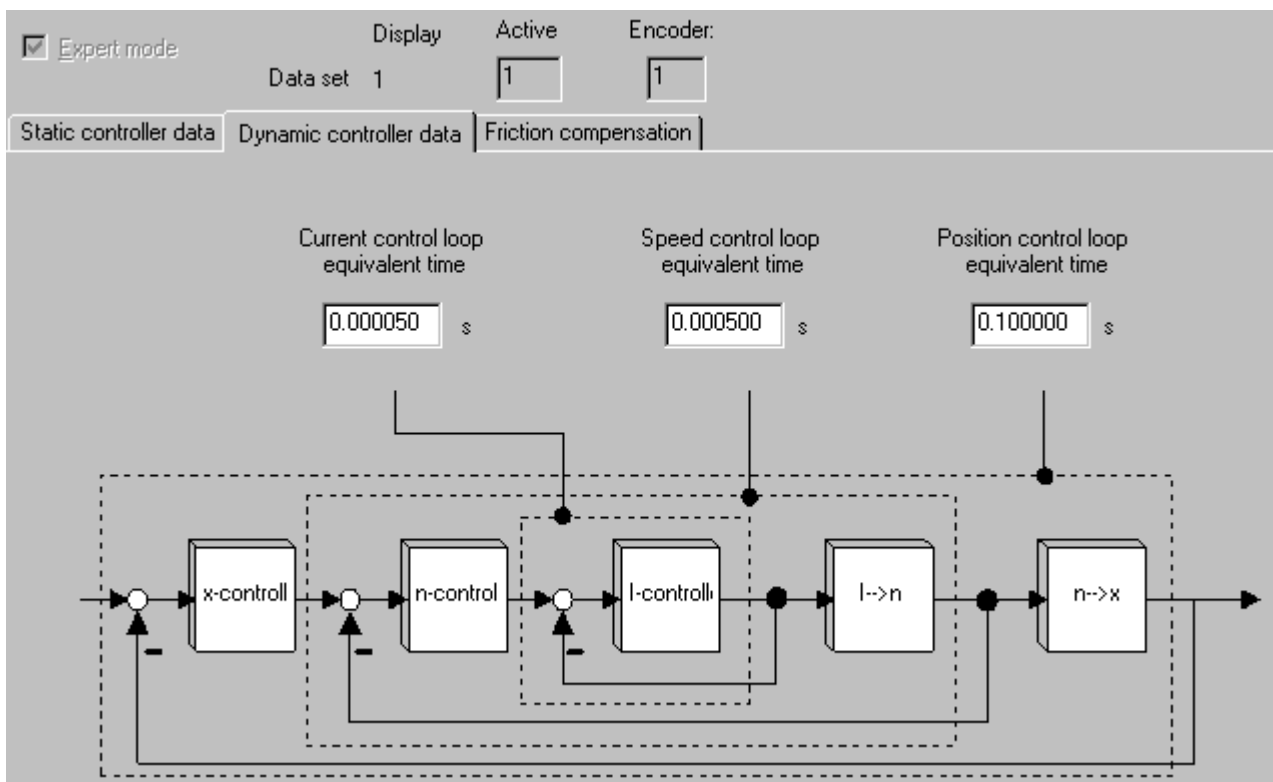
动态过滤器、预控制、平衡滤波器

有关这些设置的详细信息，请参考『优化位置控制器 — 总览』一章。

4.7.13.2 控制 —“动态控制器数据”(Dynamic controller data) 标签

例如，轴 (Axis) > 控制 (Control) 的“动态控制器数据”(Dynamic controller data) 选项卡可用于调整轴的动态响应，以匹配它们在同步操作中的响应。在“静态控制器数据”(Static controller data) 选项卡中设置了专家模式后，“动态控制器数据”(Dynamic controller data) 选项卡便可用。

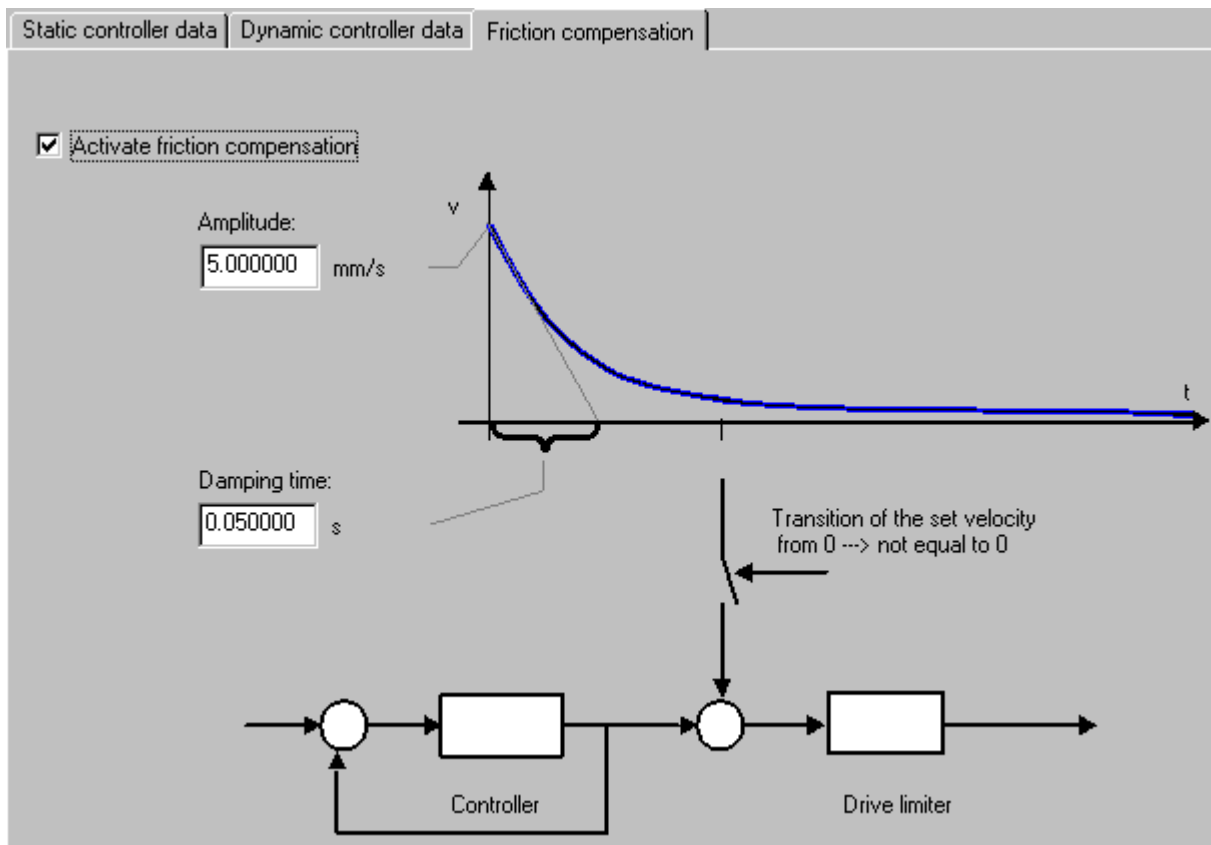
控制回路的设定值分支包含一个可组态的动态过滤器，可用于调整轴的动态响应。



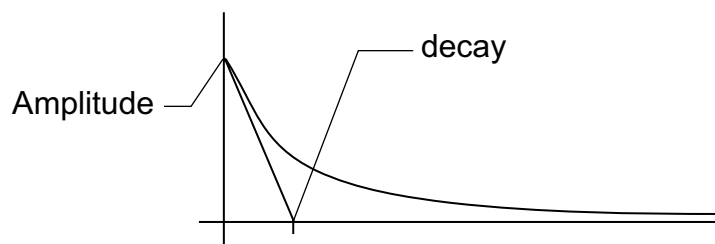
- **等效时间当前控制回路**
在本版本中不使用等效时间当前控制回路。
- **等时速度控制回路**
“等时速度控制回路”参数可用于设置平衡滤波器的时间常量 vTc （速度时间常量）。
- **等时位置控制回路**
需要位置控制回路的等时，以从速度控制模式切换到位置控制模式，并启动急停斜坡。

4.7.13.3 控制 —“摩擦补偿”(Friction compensation) 标签

轴 (Axis) > 控制 (Control) 选项卡的“摩擦补偿”(Friction compensation) 选项卡用于启用摩擦补偿。在“静态控制器数据”(Static controller data) 选项卡中设置了专家模式后,“摩擦补偿”(Friction compensation) 选项卡便可用。



系统提供了一种简单的方法来补偿静态摩擦力。从停止开始启动时, DT1 单元在受控变量上添加一个静态摩擦补偿信号。



摩擦补偿是相对于速度设定值而添加的。只有执行运动命令时摩擦补偿才会处于活动状态。

可以单独为静态摩擦补偿设置停止状态标识、振幅和衰减响应。可以在组态中设置振幅和衰减响应。

4.7.14 回原点

4.7.14.1 引言

每次接通电源时，装有增量电机测量系统的位置控制轴必须参考机器的机械系统的位置。根据轴机制已定义位置的激活特定位置值，在回原点期间同步轴。

轴可以在以下模式下回原点：

- 主动模式（参考点逼近）
- 被动模式（快速参考）
- 直接模式（使用位置设定值）

详细设置取决于参考点的测量系统以及轴的回原点运动。

4.7.14.2 回原点 —“主动回原点”(Active homing) 标签

主动回原点

可以在 S7T Config 中的轴 (Axis) > 回原点 (Homing) 对话框的“主动回原点”(Active homing) 选项卡中组态主动回原点。

“主动回原点”(Active homing) 有三种可用的回原点模式：

- 参考凸轮和编码器零标记
- 仅编码器零标记
- 仅外部零标记

启用回原点功能：

MC_Home	Mode = 0	根据轴组态确定参考点
	Mode = 1	根据轴组态确定参考点。给参数点分配的值是 <i>Position</i> 输入参数。
	Position = x	

成功回原点后，轴工艺 DB 将返回状态 *Statusword.HomingDone* = TRUE。

使用“参考凸轮和编码器零标记”(reference cam and encoder zero mark) 的主动回原点模式

使用“MC_Home”工艺功能开始回原点后，轴将根据轴组态移动至参考凸轮，然后逼近参考编码器零标记。该图的下半部分显示了最终的运动顺序。

各个参数的含义：

参数	值	说明
需要回原点	是	轴必须回原点才能执行绝对运动命令。
	否	轴不必回原点就可以执行绝对运动命令。
回原点模式	-	在这种情况下，“回原点输出凸轮和编码器零标记”
编码器零标记	参考凸轮之前	轴以参考点逼近方向返回编码器零标记，该标记位于参考凸轮前面。
	参考凸轮之后	轴以参考点逼近方向返回编码器零标记，该标记位于参考凸轮后面。
开始参考点逼近	正方向	正方向上的参考点逼近。
	负方向	负方向上的参考点逼近。

参数	值	说明
参考凸轮的逻辑地址	[字节地址]	参考凸轮的逻辑字节地址 可以将参考凸轮连接至 T-CPU 的集成输入或连接至 DP(DRIVE) 上的 IO。
位号	[位号]	用于参考凸轮的信号的位地址
逼近速度	-	轴逼近参考凸轮的速度
进入速度	-	检测到编码器零标记后，轴逼近（移位后的）参考点的速度
关闭速度	-	检测到参考凸轮后，轴逼近编码器零标记的速度。
参考点坐标 (参考凸轮和编码器零标记、仅外部零标记、仅编码器零标记)	-	在此处输入原点位置的实际位置值（相对于轴的坐标系）
回原点位置偏移	-	“原点位置偏移”(Home position offset) 将参考点移位所组态的距离。轴按照在“进入速度”(Entry velocity) 中定义的速度移动该组态距离，起点为具有编码器零标记的同步位置。 模数轴始终取最短距离。
到回原点输出凸轮的最大距离	取消激活	没有监视到参考凸轮检测的距离
	激活	监视参考点逼近的起点和参考凸轮检测之间的距离。如果该距离的差值超出了组态的值，则相应的工艺 DB 将返回错误 801D。参考点逼近取消。
到编码器零标记的最大距离	取消激活	没有监视到编码器零标记所需要移动的距离
	激活	监视轴在参考凸轮和编码器零标记之间移动的距离。如果此距离的差值超过组态的距离，则相应的轴工艺 DB 会指示错误 801D。参考点逼近取消。

参考点逼近的顺序

- 阶段 1: 参考凸轮逼近

轴以组态的逼近速度，沿在“参考点逼近的起点”(Start of reference point approach) 中设置的方向开始其参考点逼近。到达参考凸轮（阶段 1）后，参考凸轮逼近结束。

也可以通过设置“到回原点输出凸轮的最大距离”(Max. distance to homing output cam) 复选框，来监视轴从参考点逼近的起点到检测出参考凸轮时所移动的距离。如果在已组态距离的起点和终点之间没有检测到参考凸轮，则参考点逼近将因出错而取消。

- 阶段 2: 与编码器零标记同步

到达参考凸轮后，轴加速/减速到关闭速度，并逼近编码器零标记。编码器零标记从“编码器零标记”(Encoder zero mark)（参考凸轮之后或之前）和“参考点逼近的起点”(Start of reference point approach)（正方向或负方向）的组合设置得出。

检测到参考凸轮后，PLC 会根据组态将轴与检测到的第一个编码器零标记同步。将轴位置设置为默认值减去在“参考点坐标”(Reference point coordinate)(*Mode = 0*) 中或在输入参数 *Position (Mode = 1)* 中定义的参考点位移。

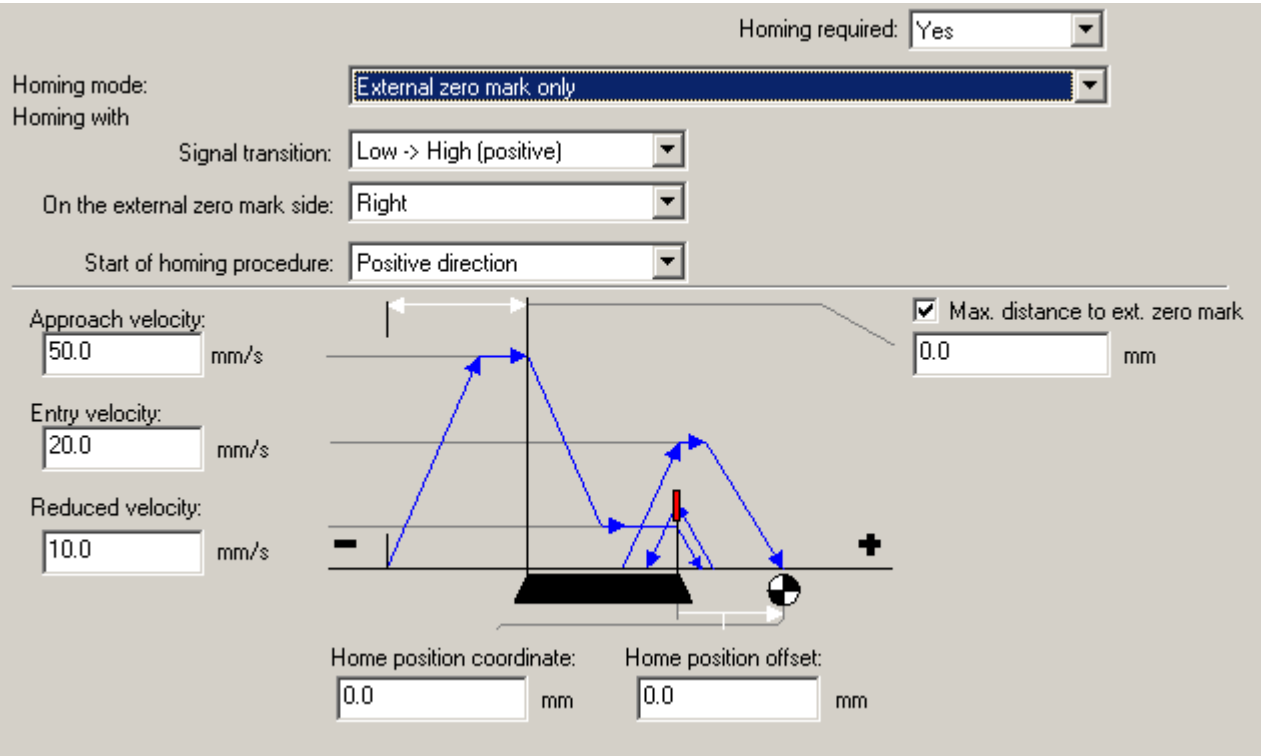
也可以通过设置“到编码器零标记的最大距离”(Max. distance to encoder zero mark) 复选框，来监视轴在参考凸轮和编码器零标记之间移动的距离。如果在检测到参考凸轮后，在指定的距离中未发现编码器零标记，则参考点逼近将因出错而取消。

- 阶段 3: 参考点逼近

检测到编码器零标记后，轴将加速/减速到逼近速度，以逼近参考点。

如果已组态了参考点位移，则轴以相应的距离（起点为同步位置）逼近该参考点。运动方向由参考点位移的符号以及检测到编码器零标记后的减速斜坡长度确定（前提是参考点位于减速斜坡内）。

使用“仅外部零标记”(External zero mark only) 的主动回原点模式



如果通过调用“MC_Home”工艺功能启动回原点过程，则轴将根据组态逼近参考外部零标记。该图的下半部分显示了最终的运动顺序。

各个参数的含义：

参数	值	说明
需要回原点	是	轴必须回原点才能执行绝对运动命令。
	否	轴不必回原点就可以执行绝对运动命令。
回原点模式	-	在此例中：“仅外部零标记”(External zero mark only)
信号跳转	低 -> 高（正）	运动参考外部零标记的上升沿（根据驱动组件中的沿评估进行设置）。
	高 -> 低（负）	运动参考外部零标记的下降沿（根据驱动组件中的沿评估进行设置）。

参数	值	说明
在外部零标记一侧	左	沿选定的参考点逼近的方向，从外部零标记的左侧进行观测，测量信号跳转。
	右	沿选定的参考点逼近的方向，从外部零标记的右侧观测，测量信号跳转。
开始参考点逼近	正方向	正方向上的参考点逼近。
	负方向	负方向上的参考点逼近。
逼近速度	-	轴逼近外部零标记的速度
进入速度	-	检测到外部零标记后，轴逼近（移位后的）回原点位置的速度。
关闭速度	-	检测到外部零标记后，轴逼近参考点坐标的速度
回原点位置偏移	-	“原点位置偏移”(Home position offset) 将参考点移位所组态的距离。轴在外部零标记沿处同步后，以“回原点速度”(Homing velocity) 移动所组态的距离。模数轴始终取最短距离。
到外部零标记的最大距离	取消激活	没有监视到外部零标记检测的距离。
	激活	监视从参考点逼近的起点到检测到外部零标记所移动的距离 如果该距离的差值超出了组态的值，则相应的工艺 DB 将返回错误 801D。参考点逼近取消。

说明

对于使用“外部零标记”(external zero mark) 的回原点，将外部零标记作为数字测量输入连接至驱动组件。

为了以“外部零标记”(external zero mark) 回原点模式执行参考点逼近，请选择与驱动器的组态或功能相对应的“信号跳转”(Signal transition) 和“在外部零标记一侧”(on the side of the external zero mark) 的值。“信号跳转”(Signal transition) 和“在外部零标记一侧”(on external zero mark side) 参数仅反映其功能，以根据组态控制轴运动，而不是用来组态驱动器中的测量功能。

有关外部零标记检测的组态的信息，请参考相关的产品信息或驱动器手册。

参考点逼近的顺序

- 阶段 1：与外部零标记同步

轴以组态的逼近速度，沿在“参考点逼近的起点”(Start of reference point approach) 中设置的方向开始其参考点逼近。在外部零标记处检测到组态的信号跳转（已在驱动组件中组态）后，同步在外部零标记处（阶段 1）结束。将轴位置设置为默认值减去在“参考点坐标”(Reference point coordinate)(*Mode = 0*) 中或在输入参数 *Position* (*Mode = 1*) 中定义的参考点位移。

也可以通过设置“到外部零标记的最大距离”(Max. distance to external zero mark) 复选框来监视轴在参考点逼近的起点和信号沿检测之间移动的距离。如果在参考点逼近的起点和已组态距离的终点之间没有检测到沿，则回原点将因出错而取消。

- 阶段 2：参考点逼近

检测到组态的信号沿后，轴将加速/减速，从而以关闭速度逼近参考点坐标。

如果已组态了参考点位移，则轴以相应的距离（起点为同步位置）逼近该参考点。运动方向由参考点位移的符号以及检测到外部零标记后的减速斜坡长度确定（前提是参考点位于减速斜坡内）。

使用“仅编码器零标记”(Encoder zero mark only) 的主动回原点模式

Homing required:

Homing mode:

Start of homing procedure:

Entry velocity: mm/s

Reduced velocity: mm/s

☒ Max. dist. to encoder zero mark mm

Home position coordinate: mm

Home position offset: mm

例如，对于某些轴，编码器在轴的整个行程范围内只能设置一个编码器零标记，在这些轴上就会使用回原点到编码器零标记。回原点命令启动到编码器零标记的轴逼近。检测到编码器零标记后，轴以回原点速度逼近位移后的参考点。现在，轴位置具有在参考点坐标中定义的值。该图中的图形显示最终的运动顺序。各个参数的含义：

参数	值	说明
需要回原点	是	轴必须回原点才能执行绝对运动命令。
	否	轴不必回原点就可以执行绝对运动命令。
回原点模式	-	在此例中：“仅编码器零标记”(Encoder zero mark only)
开始参考点逼近	正方向	正方向上的参考点逼近。
	负方向	负方向上的参考点逼近。
进入速度	-	检测到编码器零标记后，轴逼近（移位后的）参考点的速度
关闭速度	-	轴开始参考点逼近以及逼近编码器零标记的速度

参数	值	说明
回原点位置偏移	-	回原点位置偏移功能会将回原点位置移位所组态的距离，这意味着轴以“回原点速度”(Homing velocity) 在编码器零标记检测位置之后移动所组态的距离。 模数轴始终取最短距离。
到编码器零标记的最大距离	取消激活	没有监视到编码器零标记所需要移动的距离
	激活	监视从参考点逼近的起点到检测出编码器零标记时所移动的距离 如果该距离的差值超出了组态的值，则相应的工艺 DB 将返回错误 801D。参考点逼近取消。

参考点逼近的顺序

- 阶段 1：与编码器零标记同步

轴以组态的关闭速度，沿在“参考点逼近的起点”(Start of reference point approach) 中设置的方向开始其参考点逼近。与编码器零标记的同步（阶段 1）在检测到编码器零标记后结束。将轴位置设置为默认值减去在“参考点坐标”(Reference point coordinate)(*Mode = 0*) 中或在输入参数 *Position (Mode = 1)* 中定义的参考点位移。

也可以通过设置“到零标记的最大距离”(Max. distance to zero mark) 复选框，来监视轴在参考点逼近的起点和编码器零标记检测之间移动的距离。如果在参考点逼近的起点和已组态距离的终点之间没有检测到编码器零标记，则回原点将因出错而取消。

- 阶段 2：参考点逼近

检测到编码器零标记后，轴将加速/减速，从而以进入速度逼近参考点坐标。

如果已组态了参考点位移，则轴以相应的距离（起点为同步位置）逼近该参考点。运动方向由参考点位移的符号以及检测到编码器零标记后的减速斜坡长度确定（前提是参考点位于减速斜坡内）。

4.7.14.3 回原点 —“被动回原点”(Passive homing) 标签

可以在 S7T Config 的轴 (Axis) > 回原点 (Homing) 对话框的“被动回原点”(Passive homing) 选项卡中组态被动回原点。与主动回原点相比，这二者的不同之处在于，被动回原点所需的回原点运动不是由回原点命令启动的。

“被动回原点”(Passive homing) 有三种可用的回原点模式：

- 参考凸轮和编码器零标记
- 仅外部零标记
- 仅编码器零标记

启用回原点功能：

MC_Home	<i>Mode = 2</i> <i>Position = x</i>	为当前位置分配了参考点输入参数 <i>Position</i> 的值。

成功回原点后，轴工艺 DB 将返回状态 *Statusword.HomingDone = TRUE*。

说明

请注意，并非所有驱动组件都支持所有的回原点模式或测量信号计算功能。有关详细信息，请参考所用驱动组件的文档。

由工艺 DB 上相应的错误消息来表示在被动回原点操作期间对编码器零标记或参考凸轮监视的触发。在这种情况下，当前轴运动终止。

使用“参考凸轮和编码器零标记”(Reference cam and encoder zero mark) 的被动回原点模式

Homing required:

Yes

Homing mode:

Homing output cam and encoder zero mark

Log. address of homing cam:

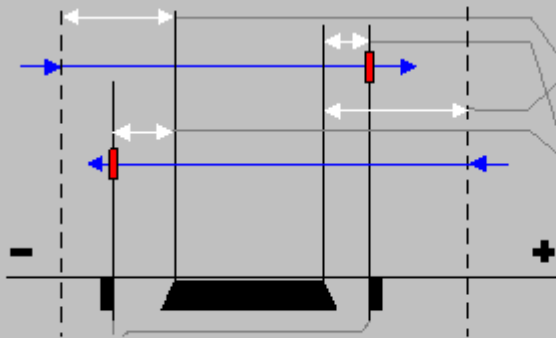
64

Direction of motion:

Current direction

Bit number:

0



☐ Max. distance to homing output cam

0.0 mm

☐ Max. dist. to encoder zero mark

0.0 mm

Home position coordinate

轴经过参考凸轮后，下一个编码器零标记将触发轴回原点。 将参考点的轴位置设置为通过“MC_Home”工艺功能的输入参数 *Position* 定义的值。

各个参数的含义：

参数	值	说明
需要回原点	是	轴必须回原点才能执行绝对运动命令。
	否	轴不必回原点就可以执行绝对运动命令。
回原点模式	-	在此例中：“参考凸轮和编码器零标记”(Reference cam and encoder zero mark)
运动方向	正方向	轴仅通过正逼近方向回原点到编码器零标记。
	负方向	轴仅通过负逼近方向回原点到编码器零标记。
	当前方向	轴在到达下一个编码器零标记时回原点。
参考凸轮的逻辑地址	[字节地址]	参考凸轮的逻辑字节地址 可以将参考凸轮连接至 T-CPU 的集成输入或连接至 DP(DRIVE) 上的 IO。
位号	[位号]	用于参考凸轮的信号的位地址

314

S7-Technology
功能手册, 03/2008, A5E01078448-06

参数	值	说明
到回原点输出凸轮的最大距离	取消激活	没有监视到参考凸轮检测的距离
	激活	监视回原点功能的起点和参考凸轮检测之间的距离。如果该距离的差值超出了组态的值，则相应的工艺 DB 将返回错误 <i>801D</i> 。回原点功能取消。
到编码器零标记的最大距离	取消激活	没有监视到编码器零标记所需要移动的距离
	激活	监视轴在参考凸轮和编码器零标记之间移动的距离。如果该距离的差值超出了组态的值，则相应的工艺 DB 将返回错误 <i>801D</i> 。回原点功能取消。

使用“仅外部零标记”(External zero mark only) 的被动回原点模式

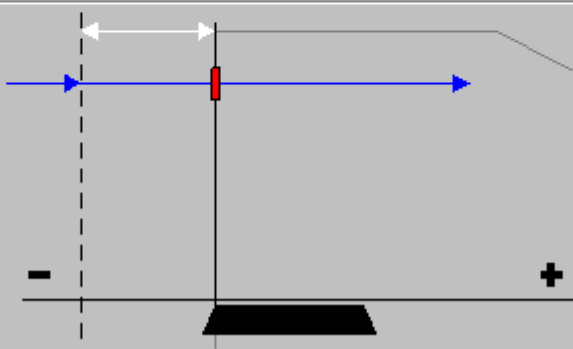
Homing required:

Homing mode:

On the external zero mark side:

Direction of motion:

☐ Max. distance to ext. zero mark
 mm



Home position coordinate

在检测到外部零标记时，开始轴回原点。将轴设置为“MC_Home”工艺功能的 *Position* 输入参数的值。

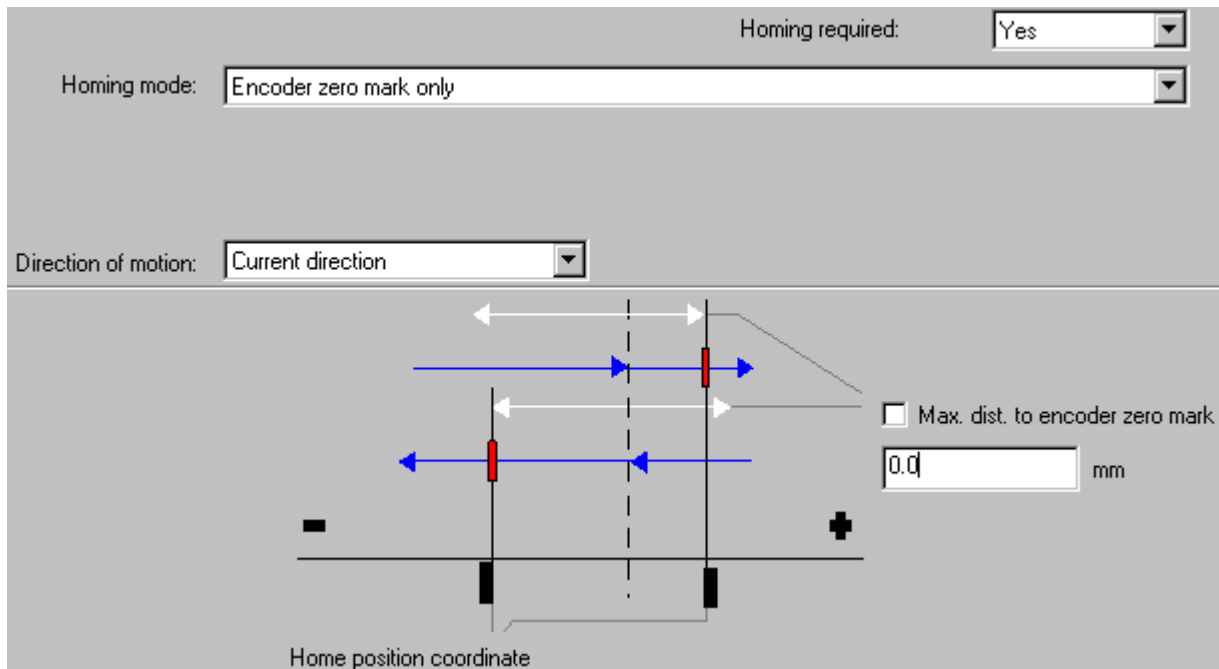
各个参数的含义：

参数	值	说明
需要回原点	是	轴必须回原点才能执行绝对运动命令。
	否	轴不必回原点就可以执行绝对运动命令。
回原点模式	-	在此例中：“仅外部零标记”(External zero mark only)
运动方向	正方向	轴仅通过正逼近方向回原点到外部零标记。
	负方向	轴仅通过负逼近方向回原点到外部零标记。
	当前方向	轴在到达下一个外部零标记时回原点。
在外部零标记一侧	左	移动方向： 正方向 在上升沿上触发轴回原点。 移动方向： 负方向 在下降沿上触发轴回原点。
	右	移动方向： 正方向 在下降沿上触发轴回原点。 移动方向： 负方向 在上升沿上触发轴回原点。
到外部零标记的最大距离	取消激活	没有监视到外部零标记所需要移动的距离
	激活	该功能将监视轴从其回原点功能的起点到检测出外部零标记所移动的距离 如果该距离的偏差超出了组态的距离，则相应的轴工艺 DB 将指示错误 801D。回原点功能取消。

说明

- 对于使用“外部零标记”(external zero mark) 的回原点，将外部零标记作为数字测量输入连接至驱动组件。
- 为了能够根据需要以“外部零标记”(external zero mark) 回原点模式执行参考点逼近，您应始终在“移动方向”(Direction of movement) 和“在外部零标记一侧”(on the side of the external zero mark) 上设置数值，使其与驱动器组态或功能兼容。“在外部零标记一侧”(on the side of the external zero mark) 和“移动方向”(Direction of movement) 参数不用于组态驱动器中的测量功能，仅反映其功能。
- 有关外部零标记检测的组态的信息，请参考相关的产品信息或驱动器手册。

使用“仅编码器零标记”(Encoder zero mark only) 的被动回原点模式



例如，使用“仅零标记”(Zero mark only) 的回原点模式用于装有一个在轴的整个行程范围内只能输出一个零标记信号的编码器的轴。

在检测到编码器零标记时，开始轴回原点。检测到编码器零标记后，将轴的位置值设置为“MC_Home”工艺功能的输入参数 *Position* 的值。

各个参数的含义：

参数	值	说明
需要回原点	是	轴必须回原点才能执行绝对运动命令。
	否	轴不必回原点就可以执行绝对运动命令。
回原点模式	-	在此例中：“仅编码器零标记”(Encoder zero mark only)
运动方向	正方向	轴仅通过正逼近方向回原点到编码器零标记。
	负方向	轴仅通过负逼近方向回原点到编码器零标记。
	当前方向	轴在到达下一个编码器零标记时回原点。
到编码器零标记的最大距离	取消激活	没有监视到编码器零标记所需要移动的距离
	激活	该功能将监视轴从其回原点功能的起点到检测到编码器零标记所移动的距离。 如果该距离的偏差超出了组态的距离，则相应的轴工艺 DB 将指示错误 <i>801D</i> 。回原点功能取消。

使用“默认”(Default) 的被动回原点模式

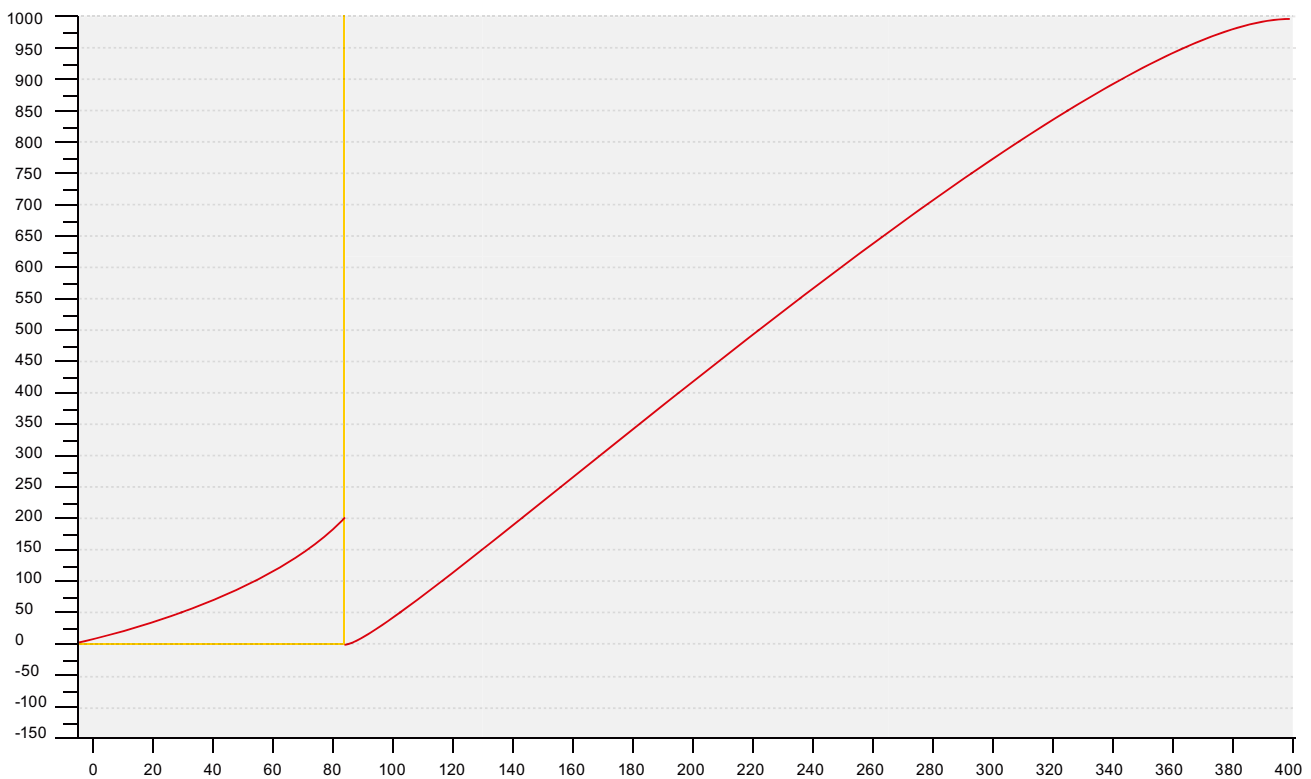
在 S7T Config 中创建新轴时，会预设“默认”(Default) 回原点模式。

如果组态的编码器提供零标记，则使用“仅编码器零标记”(encoder zero mark only) 回原点模式。如果编码器不提供零标记，则使用仅外部零标记回原点模式。

4.7.14.4 被动回原点模式的定位特性

以下几个示例表明了使用运动命令“MC_MoveRelative”和“MC_MoveAbsolute”时被动回原点模式的终点位置的响应：

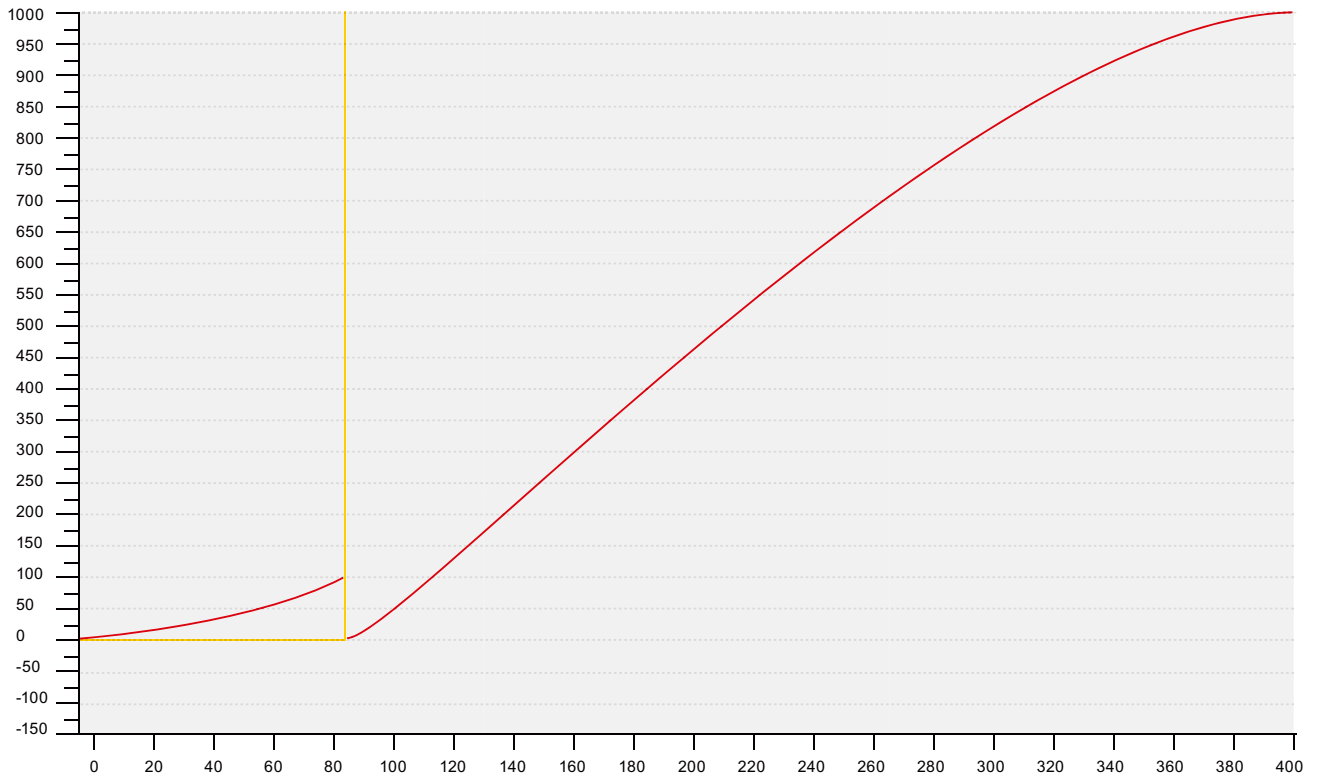
使用“MC_MoveRelative”的被动回原点模式的定位特性



针对被动回原点模式启动 MC_MoveRelative 命令 1000。

1. MC_MoveRelative 命令和被动回原点模式同时启动。
2. 到达黄色标记的回原点位置 200 时，轴回原点，轴的位置设置为位置 0。
3. 轴移动剩余的相对距离，移到位置 800 (1000-200)。

使用“MC_MoveAbsolute”的被动回原点模式的定位特性



针对被动回原点模式启动到位置 1000 的 MC_MoveAbsolute 命令。

1. MC_MoveAbsolute 命令和被动回原点模式同时启动。
2. 到达黄色标记的回原点位置 200 时，轴回原点，轴的位置设置为位置 0。
3. 根据新回原点位置，轴移动到位置 1000。

4.7.14.5 直接回原点

将实际轴位置设置为在 MC_Home 工艺功能中定义的值。 没有激活的参考点位移设置。
该功能不执行运动。 执行命令时使轴回原点。

启用回原点功能

MC_Home	<i>Mode = 3</i> <i>Position = x</i>	直接回原点： 为当前位置分配了输入参数 <i>Position</i> 的值。

成功回原点后，轴工艺 DB 将返回状态 *Statusword.HomingDone = TRUE* 。

4.7.14.6 位置校正

从轴的实际位置值中扣除校正值。 在这种情况下，与其它回原点模式不同，轴将保持其回原点状态（已回原点/未回原点）。

位置校正功能还可以用于操纵各个坐标系（基准坐标系、叠加坐标系）的设定值。 这对于叠加凸轮传动能够在凸轮盘中生成参考非常重要。

启用回原点功能：

MC_Home	<i>Mode = 4</i> <i>Position = x</i>	实际值校正： 位置值 = (当前位置) - (参数位置)。
	<i>Mode = 6</i> <i>Position = x</i>	基坐标系中的设定值校正： 位置值 = (当前位置) - (参数位置)。
	<i>Mode = 7</i> <i>Position = x</i>	叠加坐标系中的设定值校正： 位置值 = (当前位置) - (参数位置)。

位置校正不会影响轴工艺 DB 中的 *Statusword.HomingDone* 状态。

4.7.14.7 未回原点的轴的运动

可以在轴 (Axis) > 回原点 (Homing) 对话框中确定绝对定位是否可用于未回原点的轴。

“需要回原点”(Homing required) 的可选设置：

- 否： 可以进行相对运动和绝对运动。 将监视软件限位开关。
- 是： 仅可以进行相对运动。 只要轴没有回原点便不监视软件限位开关。

4.7.15 监视功能

4.7.15.1 监视功能

根据通过跟随轴上的同步对象和补偿功能计算的从站值在跟随轴上的动态激活，对其进行监视。 如果无法进行动态激活，则生成相应的调整移动。

目前设置的速度和加速度（包括冲击）的上限在此轴上处于活动状态。

如果无法执行运动，则尝试通过启动补偿运动（取决于此轴的最大动态值）尽量接近这些值。 结果为设定值错误。

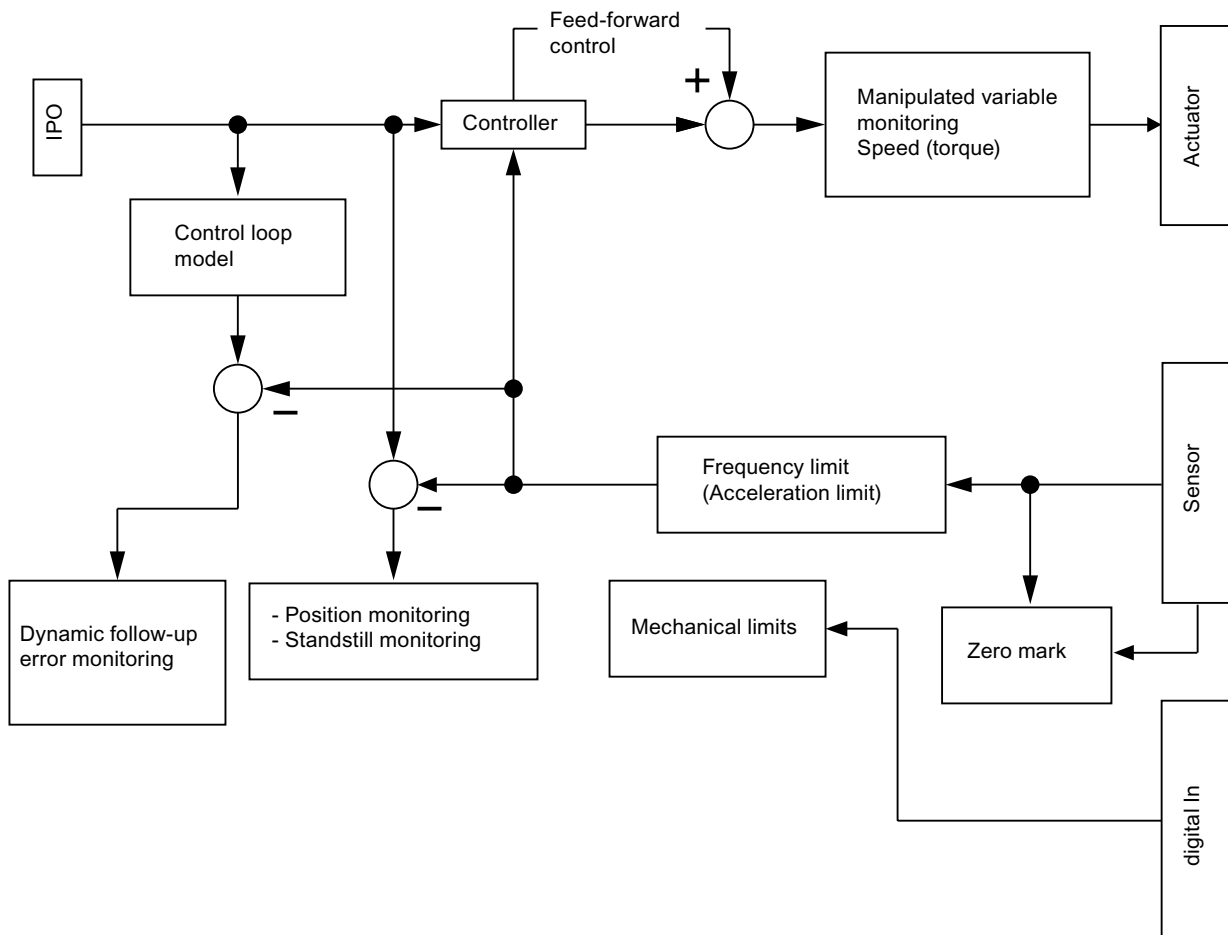
4.7.15.2 监视功能 — 总览

可以在 S7T Config 中组态的轴监视功能：

监视功能	速度控制轴	定位轴	同步轴
速度误差监视	X	-	-
定位监视	-	X	X
跟随误差监视	-	X	X
停止信号	X	X	X
软件限位开关	-	X	X
硬件限位开关	X	X	X
同步监视	-	-	X
受控变量监视（始终可用）	X	X	X
编码器限制频率监视	X（仅适用于编码器）	X	X

定位轴监视

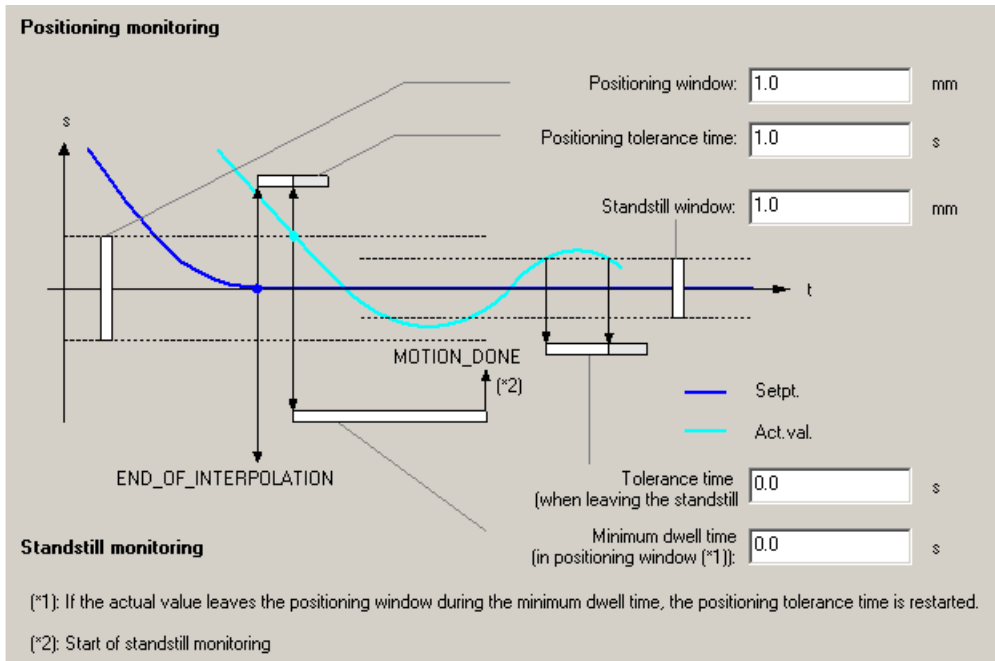
下图显示了一个定位轴监视的示例：



在工艺 DB 的 *ErrorStatus* 参数中报告错误。

4.7.15.3 监视功能 — 定位和停止监视

可以在轴 (Axis) > 监视功能 (Monitoring functions) 对话框的“定位和停止监视”(Positioning and standstill monitoring) 选项卡中设置用于监视位置控制轴和同步轴定位的限制值。



“定位和停止监视”(Positioning and standstill monitoring) 选项卡显示为轴设置了定位还是同步操作工艺。

定位监视

在定位即将结束时，该功能将监视轴进入目标位置。为此，指定一个定位窗口和一个时间公差，在该时间公差内必须到达终点位置。在设定值插补结束时激活监视。

也可以将定位窗口中的最小停留时间设置为在激活定位命令的正反馈之前结束。该时间可以用于振动过程和控制回路，为此公差窗口应小于超调振幅。

当实际值到达定位窗口时，将输出定位命令的正反馈。

如果触发了定位监视功能，则轴工艺 DB 将返回错误 8019。

停止（零速度）监视

当定位命令的位置设定值与目标位置的值相等并且停止监视激活的延迟已到期时，将启用停止监视。

如果轴移至已组态的停止公差窗口之外的时间超过已组态的时间段，则将触发停止监视。

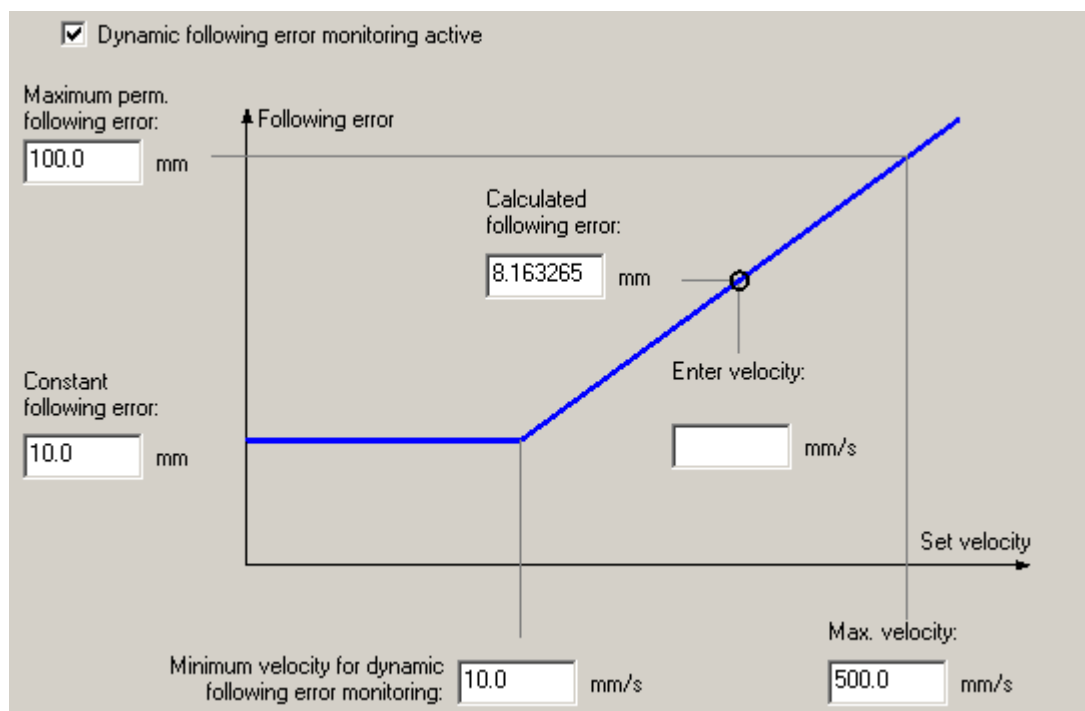
如果触发了停止监视，则轴工艺 DB 将返回错误 8018。

4.7.15.4 监视功能 —“跟踪误差监视”(Following error monitoring) 标签

可以在轴 (Axis) > 监视功能 (Monitoring functions) 对话框的“跟随误差监视”(Following error monitoring) 选项卡中启用动态跟随误差监视。

使用计算出的跟随误差执行对位置控制轴的跟随误差监视。如果实际位置值和位置设定值之间的偏移超出已编程的跟随误差限制，轴工艺 DB 将返回错误 8016。允许的跟随误差取决于轴的速度设定值。

当速度小于已组态的最小速度时允许的跟随误差为常量，这是在“恒定跟随误差”(constant following error) 参数中编程的。大于此限制时，跟随误差将线性增加倒最大值，该最大值由最大速度时的参数“允许的最大跟随误差”(maximum permitted following error) 定义。在最大速度时达到允许的最大跟随误差。



如果在“输入速度”(Enter velocity) 输入框中指定一个值，则相应的功能值显示在“计算的跟随误差”(Calculated following error) 框中。这允许您控制在输入的速度下跟随误差可以有多大。

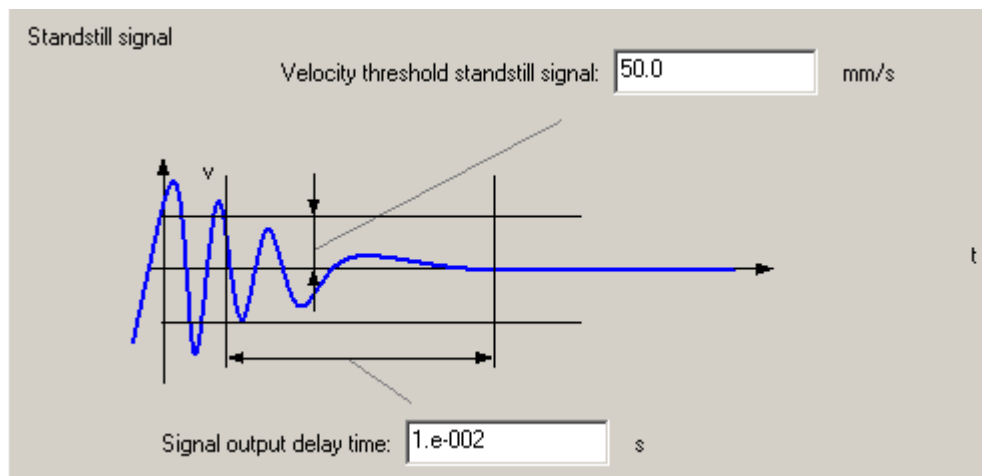
对于位置控制的实际轴，将显示“跟随误差监视”(Following error monitoring) 选项卡。

4.7.15.5 监视功能 —“停止信号”(Standstill signal) 标签

可以在“轴”(Axis) > “监视功能”(Monitoring functions) 对话框的“停止信号”(Standstill signal) 选项卡中设置速度阈值和“信号输出延迟”(Signal output delay)。

如果当前速度至少在设定的延迟时间期间低于组态的速度阈值，则停止信号将在工艺 DB 的状态字中设置“停止”(standstill) 位。

在速度控制轴和定位轴上，将在速度控制模式下激活重写。在定位轴上，将在到达定位窗口时激活重写。



4.7.15.6 监视功能 —“同步运行监视”(Synchronous operation monitoring) 标签

轴 (Axis) > 监视功能 (Monitoring functions) 的“同步操作监视”(Synchronous operation monitoring) 选项卡可用于在跟随轴上启用同步误差监视。如果为轴设置了同步工艺，则显示“同步操作监视”(Synchronous operation monitoring) 选项卡。

以下设置可用：

Standstill signal Synchronous operation monitoring

Activate setpoint monitoring: With jerk

Setpoint tolerance: 10.000000 mm

Activate actual value monitoring: Yes

Actual value tolerance: 0.000000 mm

Report error of master axis: Setpoints

激活设定值监视:

同步关联的设定值与受动态值限制的跟随轴设定值之间可能出现偏差。您可以通过采取以下措施对设定值偏差做出反应:

- “无”(no)
如果您要放弃监视, 请选择“无”(no)。
- “无冲击”(without jerk)/“有冲击”(with jerk)
选择其中一个设置可以激活监视 (此版本不允许为冲击留出余量)。

跟随轴上的设定值错误使轴停止: 跟随轴工艺 DB 返回错误 **801A**。

设定值公差:

输入设定值的最大公差。

激活实际值监视:

实际值监视功能用于监视跟随轴的设定值和实际值之差。可以启用和禁用实际值监视。

跟随轴上的实际值错误使轴停止: 跟随轴工艺 DB 返回错误 **801A**。

实际值公差:

输入实际值的最大公差。

报告主轴的错误:

同步关联的设定值与受动态值限制的设定值之间可能出现偏差。您可以通过采取以下措施对设定值偏差做出反应:

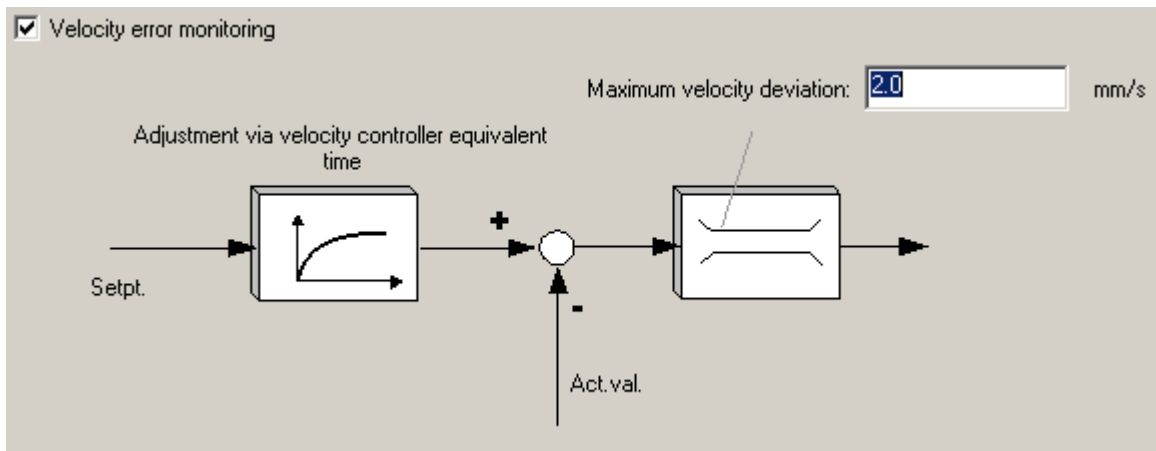
- “无”(no)
在跟随轴上检测到设定值或实际值错误时, 引导轴不会停止。
- “设定值”(Setpoints)
跟随轴上的设定值错误还可以停止引导轴; 引导轴工艺 DB 返回错误 **801A**。
- “实际值”(Actual values)
跟随轴上的实际值错误还可以停止引导轴; 引导轴工艺 DB 返回错误 **801A**。

4.7.15.7 监视功能 —“速度监视”(Velocity monitoring) 标签

可以在轴 (Axis) > 监视功能 (Monitoring functions) 对话框的“速度误差监视”(Velocity error monitoring) 选项卡中启用速度误差监视。

速度误差监视仅与以下这些轴有关：

- 带有编码器的速度控制轴
- 操作在速度控制模式下的定位轴
(“MC_MoveVelocity”— 输入参数 *PositionControl* = *FALSE*)



速度误差监视

在该对话框中启用速度误差监视。

最大速度误差：

仅当启用了速度误差监视时该输入框才可见。

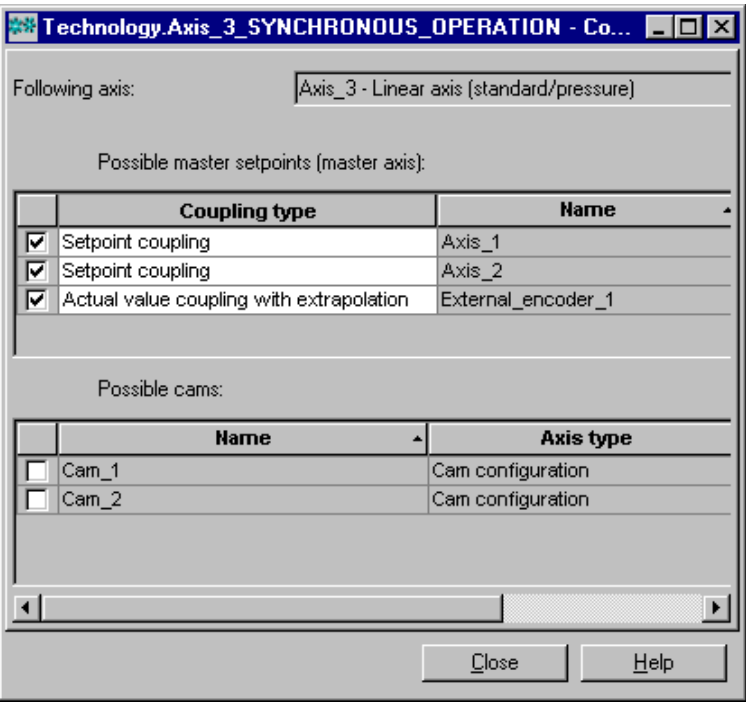
在该对话框中输入最大速度误差。

4.7.16 跟随对象

4.7.16.1 组态

组态 — 同步对象

在对话框“同步对象”(Synchronous object) >“组态”(Configuration) 中将主值和凸轮分配到同步轴（跟随轴）。



跟随轴：

此域中显示跟随轴（同步轴）的名称。

可能的主值（引导轴）：

此列表中列出了项目中可用的主值，可将该值分配给跟随轴。 可通过下列工艺对象指定主值：

- 轴（实际轴和虚拟轴）
- 外部编码器

在此对话框中，考虑将在包含各工艺功能（如“MC_GearIn”或“MC_CamIn”）的用户程序中使用的引导轴/跟随轴的所有组合。

可能的凸轮：

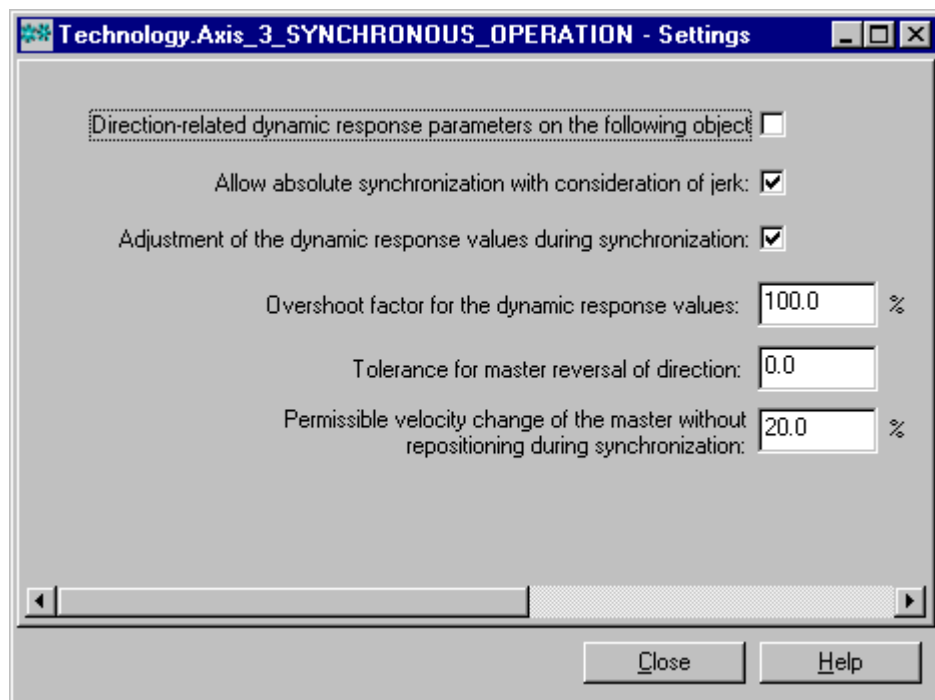
该列表包括所有已组态的凸轮。可将凸轮分配给用于凸轮传动的同步对象。

在此对话框中，考虑将在包含相关同步轴的用户程序中使用的所有凸轮。

4.7.16.2 设置

设置 — 同步对象

在对话框“**设置 — 同步对象**”(Settings - Synchronous object) 中，定义用于同步的一些参数。



同步对象上与方向相关的动态参数 (Direction-related dynamic parameters on the synchronous object)

如果希望已编程动态响应值达到与方向相关的效果，则激活该复选框。

允许绝对同步考虑冲击 (Permit absolute synchronization with provision for jerk)

如果在绝对同步时考虑冲击，则激活此复选框。此设置仅对于跟踪同步有效。

修改同步的动态响应值 (Adapt the dynamic response values for the synchronization)

如果要在同步位置修改动态响应值，则激活该复选框。

动态响应值的放大因子 (Magnification factor for the dynamic response values)

（仅当激活复选框“修改同步的动态响应值”[Adapt the dynamic response values for the synchronization] 时，该域才可见）

在此处为已修改的动态响应值指定超调因子，以补偿剩余路径差异。输入与用于同步启动的当前主值速度相关的百分数 (%) 值。

主方向反转的公差 (Tolerance for master direction reversal)

指定用于取消主值方向反转的同步的公差值。

同步时无需重启即允许改变主轴速度 (Permitted velocity change of the master without restarting for synchronization)

在此域中输入同步期间允许对主值速度进行的最大更改。该值指的是用于同步启动的当前主值速度

如果在同步期间引导轴加速大于指定值，则跟随轴无法同步。在这种情况下，增加允许主轴速度更改的值。

4.7.16.3 默认设置

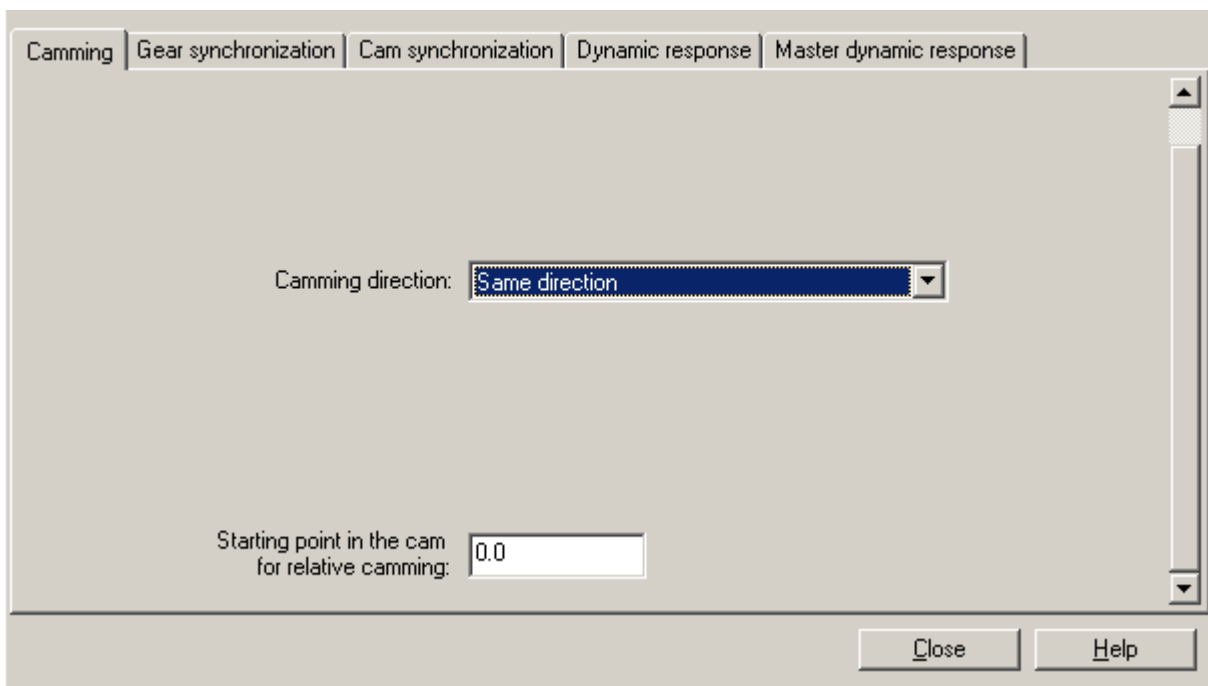
同步对象缺省值 —“凸轮同步”(Cam synchronization) 标签

在 **Axis_SYNCHRONOUS_OPERATION > 默认 (Default)** 对话框的“凸轮同步”(Cam synchronization) 选项卡中，设置同步对象的属性以激活凸轮传动。

仅当在“MC_CamIn”或“MC_CamInSuperImposed”工艺功能中设置 *Mode = 2* 时，此对话框中的设置才能被激活。

主设定值源和跟随轴之间的变量耦合是凸轮传动的特性。凸轮盘（传送功能）说明了耦合。

凸轮传动的缩放和移位（偏移）在主设定值源和跟随轴中均可能发生。这将根据凸轮盘的定义范围和取值范围分别对凸轮盘进行修改。



凸轮传动中的方向感：

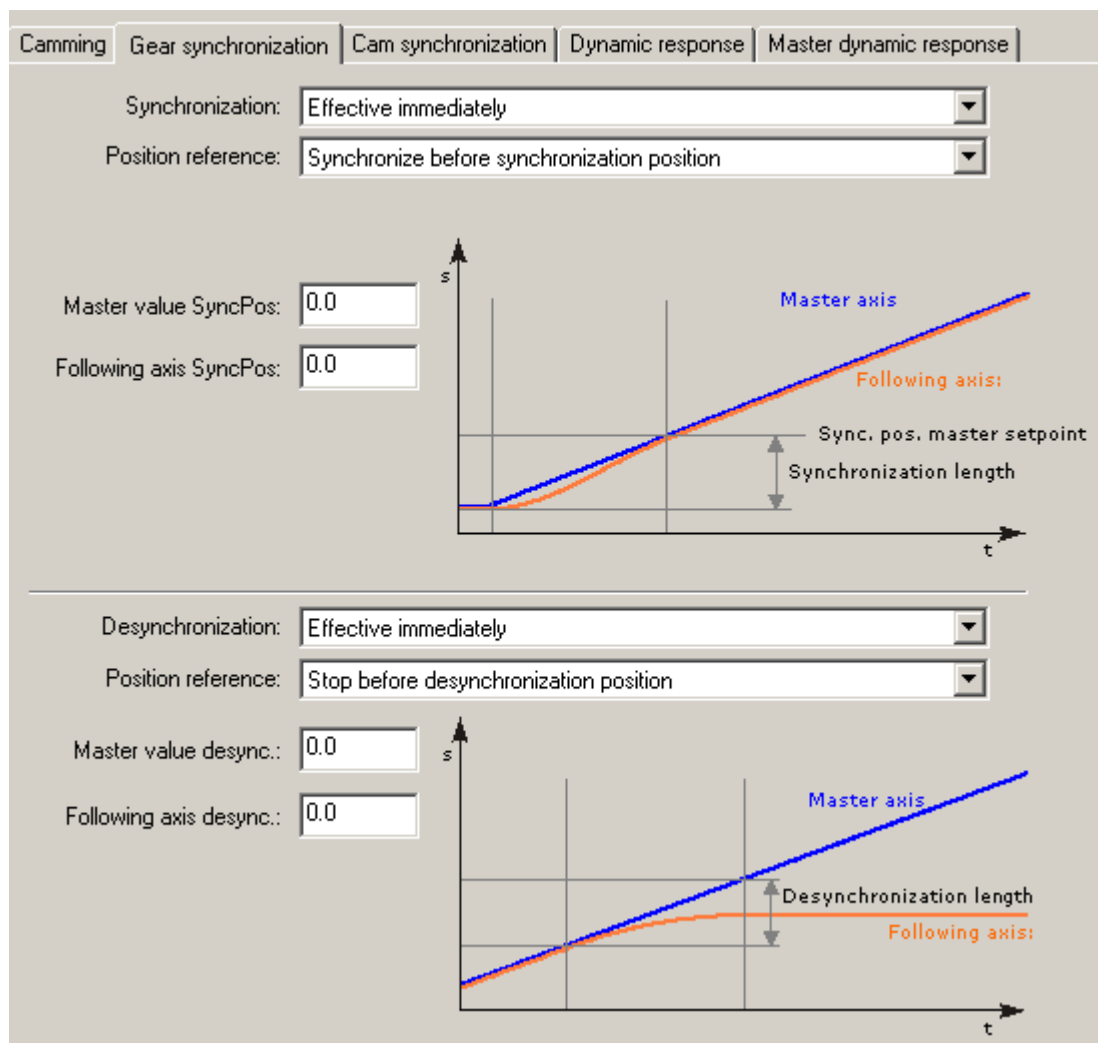
指定凸轮盘执行的方向。

相关凸轮传动在凸轮盘中的起点：

在凸轮盘中定义相关凸轮传动的起点，作为引导轴超过凸轮盘的起点。

同步对象缺省值 —“齿轮同步”(Gear synchronization) 标签

在 **Axis_SYNCHRONOUS_OPERATION > 默认 (Default)** 对话框的“齿轮同步”(Gear synchronization) 选项卡中设置同步/去同步参数。这些设置仅在传动模式下相关。您可以设置以下参数：



同步

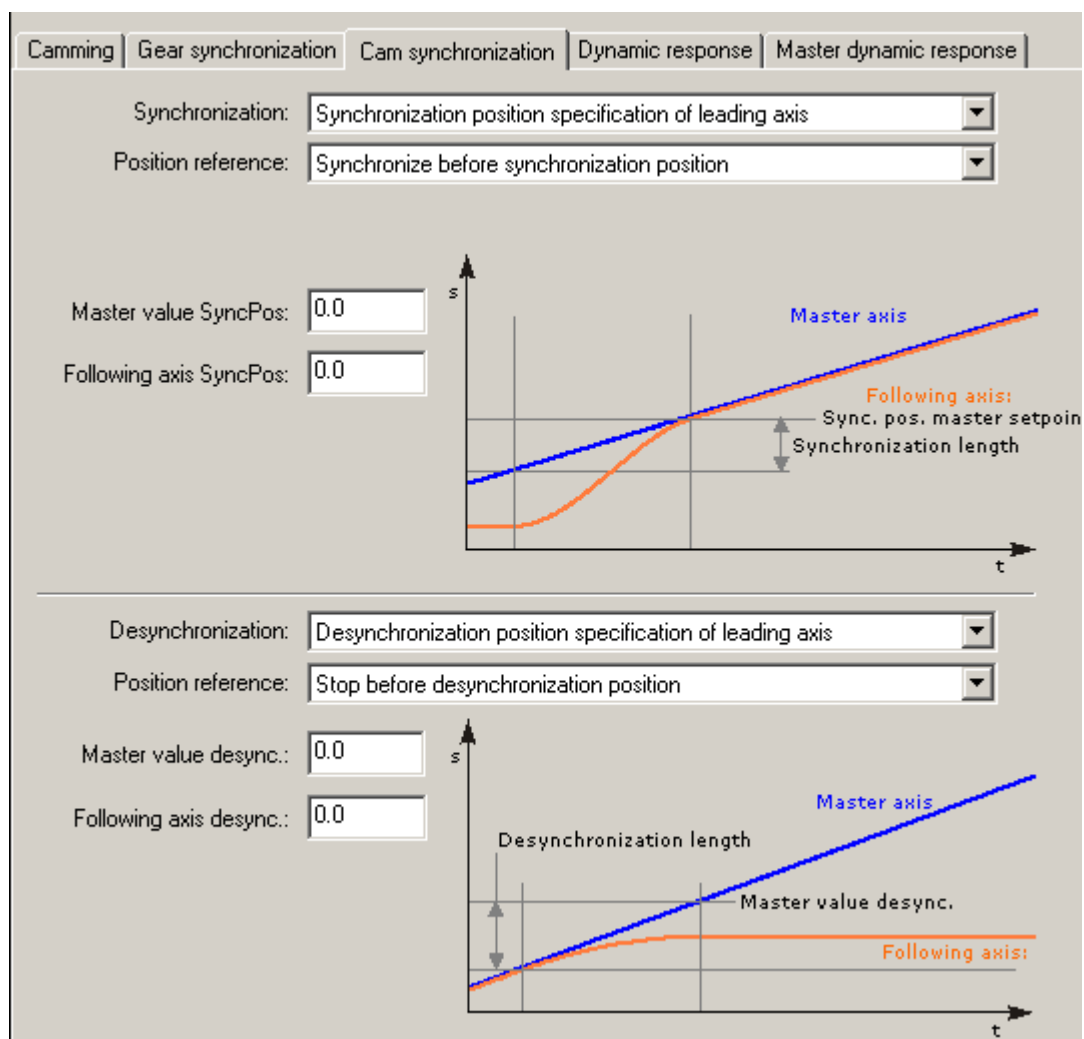
- **同步：**
定义跟随轴与引导轴同步的时间。
- **位置参考：**
定义与同步点位置相关的同步曲线的位置。
- **同步方向：**（仅针对模数轴）
定义同步方向。
- **主值同步位置：**
输入引导轴同步点的位置。
- **跟随轴同步位置：**
输入跟随轴同步点的位置。

去同步：

- **去同步：**
定义取消激活跟随轴与引导轴同步的时间
- **位置参考：**
定义与去同步点位置相关的去同步曲线的位置。
- **Desync. desync.:**
输入引导轴去同步点的位置。
- **Desync. desync.:**
输入跟随轴去同步点的位置。

同步对象缺省值 —“凸轮同步”(Cam synchronization) 标签

在 **Axis_SYNCHRONOUS_OPERATION** 对话框的“凸轮同步”(Cam synchronization) 选项卡中设置同步/去同步参数。这些设置仅与凸轮传动模式相关。您可以设置以下参数：



同步

- **同步：**
定义跟随轴与引导轴同步的时间。
- **位置参考：**
定义与同步点位置相关的同步曲线的位置。
- **同步方向：**（仅针对模数轴）
定义同步方向。
- **主值同步位置：**
输入引导轴同步点的位置。
- **跟随轴同步位置：**
输入跟随轴同步点的位置。

去同步

- **去同步：**
定义取消激活跟随轴与引导轴同步的时间
- **位置参考：**
定义与去同步点位置相关的去同步曲线的位置。
- **主值去同步：**
输入引导轴去同步点的位置。
- **跟随轴去同步：**
输入跟随轴去同步点的位置。

同步对象缺省值 —“动态”(Dynamics) 标签

在 **Axis_SYNCHRONOUS_OPERATION** 对话框的“动态”(Dynamics) 选项卡中，设置同步/去同步的基本默认值。

Camming | Gear synchronization | Cam synchronization | **Dynamic response** | Master dynamic response

Profile specification: Time-related synchronization profile

Master-axis-related synchronization

Sync. length: 2.0 - Desync. length: 3.0 -

Time-related synchronization

Velocity: 100.0 mm/s

Velocity profile: Last programmed velocity profile

Jerk: 1000000.0 mm/s³

Acceleration: 1000.0 mm/s²

Jerk: 1000000.0 mm/s³

Jerk: 1000000.0 mm/s³

Deceleration: 1000.0 mm/s²

Jerk: 1000000.0 mm/s³

曲线设置:

您可以从中定义同步曲线的参考。

- 与引导轴相关的同步
- 与时间相关的同步曲线
- 上一个已编程的设置

在此版本中，默认值“上一个已编程的设置”(last programmed setting) 曲线不可用。

与引导轴相关的同步

- **同步 长度 (Sync. Length):**
您可以从中输入同步距离。
- **去同步 长度 (Desync. Length):**
您可以从中输入去同步距离。

对于与引导轴相关的同步曲线，只能判断同步/去同步长度。

与时间相关的同步

- **速度 (Velocity):**
您可以从中输入最大速度。
- **速度曲线 (Velocity profile):**
您可以从中选择速度曲线。
- **加速度 (Acceleration):**
在此参数中输入最大加速度。
- **减速度 (Deceleration):**
在此参数中输入最大减速度。
- **冲击 (Jerk):**
在此参数中输入最大冲击。

对于与时间相关的同步曲线，只能判断参数速度曲线 (Velocity profile)、速度 (Velocity)、加速度 (Acceleration)、减速度 (Deceleration) 和冲击 (Jerk)。

同步对象缺省值 —“主动态响应”(Master dynamic response) 标签

在 **Axis_SYNCHRONOUS_OPERATION > 缺省 (Default)** 对话框的“主动态响应”(Master dynamic response) 选项卡中，设置主设定值转换的动态响应值。

Camming | Gear synchronization | Cam synchronization | Dynamic response | **Master dynamic response**

Master switchover with dynamic response values

Velocity: 100.0

Velocity profile: Last programmed velocity profile

Jerk: 1000000.0

Acceleration: 1000.0

Jerk: 1000000.0

Jerk: 1000000.0

Deceleration: 1000.0

Jerk: 1000000.0

The mass units of the respective master axis are used.

您可以设置以下参数：

- **速度 (Velocity):**
您可以从中输入最大速度。
- **速度曲线 (Velocity profile):**
您可以从中选择速度曲线。
- **冲击 (Jerk):**
在此参数中输入最大冲击。
- **加速度 (Acceleration):**
在此参数中输入最大加速度。
- **减速度 (Deceleration):**
在此参数中输入最大减速度。

4.8 凸轮组态

4.8.1 凸轮组态

在用户程序中应用的任何凸轮盘必须已插入 S7T Config 作为“凸轮盘”工艺对象。

用户程序中的凸轮盘通过“MC_CamClear”、“MC_CamSectorAdd”和“MC_CamInterpolate”工艺功能进行定义。可以调用“MC_GetCamPoint”工艺功能读入插补凸轮盘的位置值。

通过使用 *CamTable* 输入参数来定义在 S7T Config 中创建的凸轮盘的引用。必须在 *CamTable* 中输入为凸轮盘创建的工艺数据块的数量。

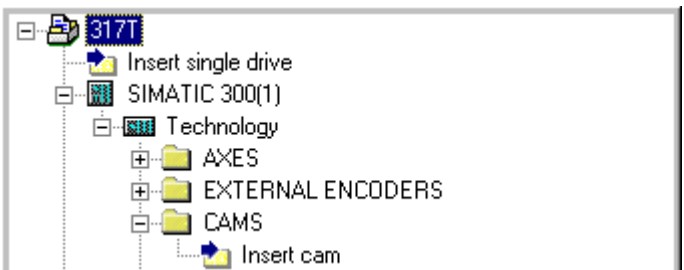
而凸轮盘应用是在用户程序中通过对“MC_CamIn”和“MC_CamInSuperimposed”工艺功能进行编程定义的。凸轮盘通过 *CamTable* 输入参数进行选择。使用的先决条件是分配到相应的同步组态。

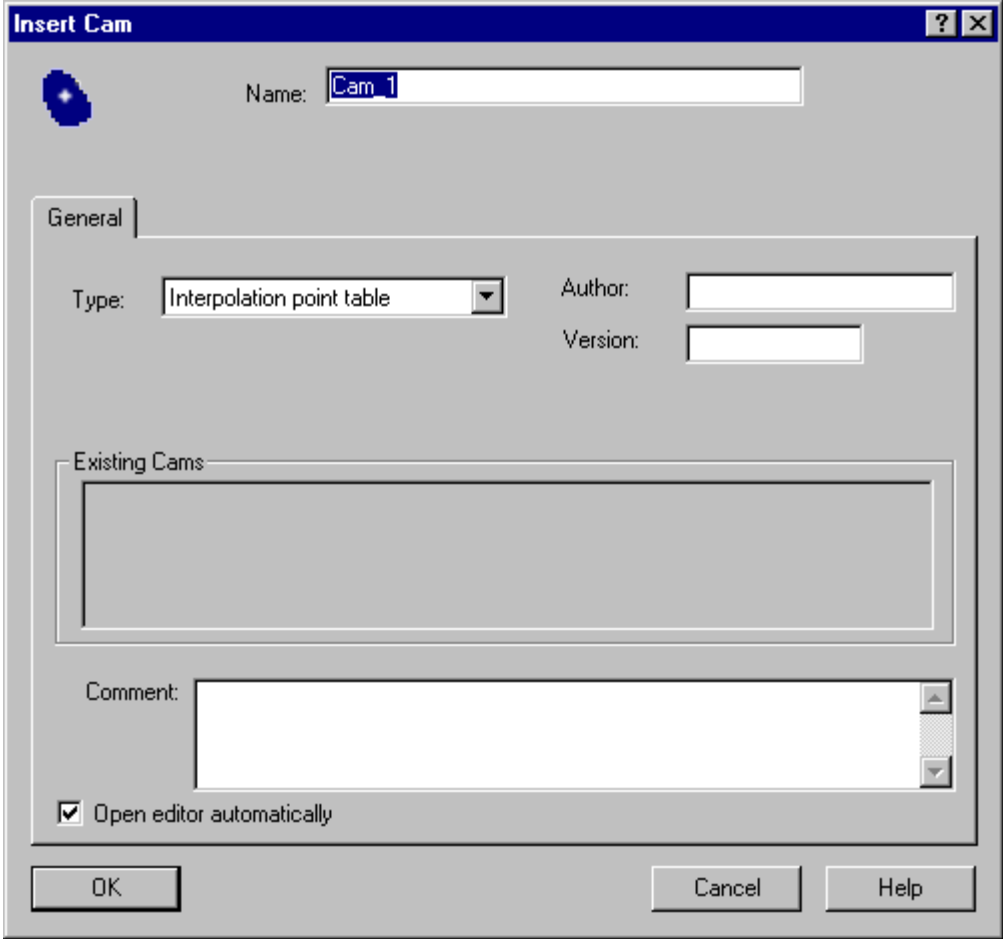
4.8.2 插入凸轮

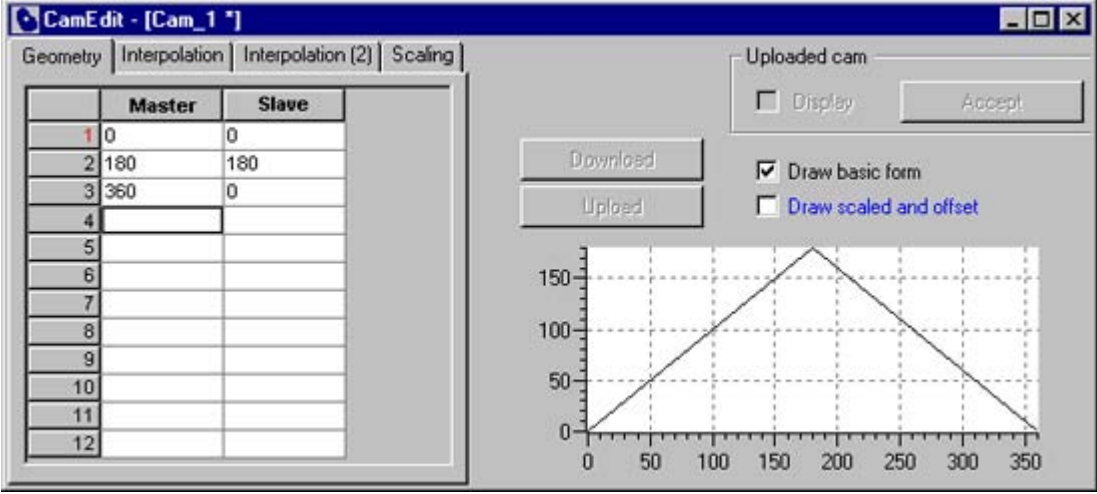
要求

- T-CPU 的数据已在 HW Config 中进行组态并编译。

如何将凸轮盘插入 S7T Config 中

步骤	说明
1.	要使用 CamEdit 工具在 S7T Config 中创建新的凸轮盘，请双击 S7T Config 的浏览器中的“插入凸轮盘”(Insert cam disk).  要使用可选软件包 SCOUT CamTool 创建凸轮盘，请双击“使用 CamTool 插入凸轮盘”(Insert cam disk with CamTool)。该条目仅在安装可选软件包后才会显示。 在该示例的下一个阶段，我们将仅涉及 CamEdit。有关 SCOUT CamTool 的功能和操作的详细说明，请参考《SIMOTION CamTool》手册。

步骤	说明
2.	在“插入凸轮盘”(Insert cam disk) 对话框中输入工艺对象的名称。 将另外提供用于输入作者、版本号和注释的域。 选择是使用插补点表定义凸轮，还是使用多项式进行定义。 选中“自动打开编辑器”(Open editor automatically) 复选框，以便自动打开凸轮盘组态对话框：  单击“确定”(OK)。 结果： CamEdit 对话框随即打开。

步骤	说明
3.	通过插补表或多项式定义凸轮盘，具体取决于类型。必要时更改凸轮盘的缩放比例和插补模式。 
4.	关闭 CamEdit。
	结果：工艺对象随即插入。

说明

如果希望在插补后插入新的多项式或插补点，则必须预先删除凸轮盘。
在插补和连续性检查期间会忽略引导轴和跟随轴的属性。

4.8.3 定义凸轮

定义凸轮盘

在**插入凸轮 (Insert Cam)** 对话框中，您可以选择定义基于插补点的凸轮或基于多项式的凸轮。

- **基于插补点的定义**
插补点在插补点表中以 **P = P(x,y)** 的形式表示。与数值对的输入顺序无关。它们在域中按升序自动排序。
S7T Config 根据组态的插补类型插补凸轮。
- **基于多项式/区段的定义**
各种多项式按照“凸轮机械运动法则，VDI 指令 2143”予以说明。多项式的最大阶数为 6。多项式还可以包含三角函数。

凸轮的定义方法

两种定义方法，即基于插补点或多项式，各有其自己的优点和缺点。您的应用决定了哪种方法可以满足要求。
一个凸轮中结合两种方法的情况只能在使用 MC_CamSectorAdd 工艺功能的运行时间中出现。

	基于插补点的定义	基于多项式的定义
优点	● 定义简单 ● 任何算法均可以通过插补点进行映射 ● 通过 Teach-In 绘图 ● HMI 的界面简单	● 用于定义的数据量少 ● 按照 VDI 的标准转换..... ● 外形非常精确，转换连续
缺点	● 需要大量插补点才能获得平滑的外形	● 需要进行复杂的系数计算

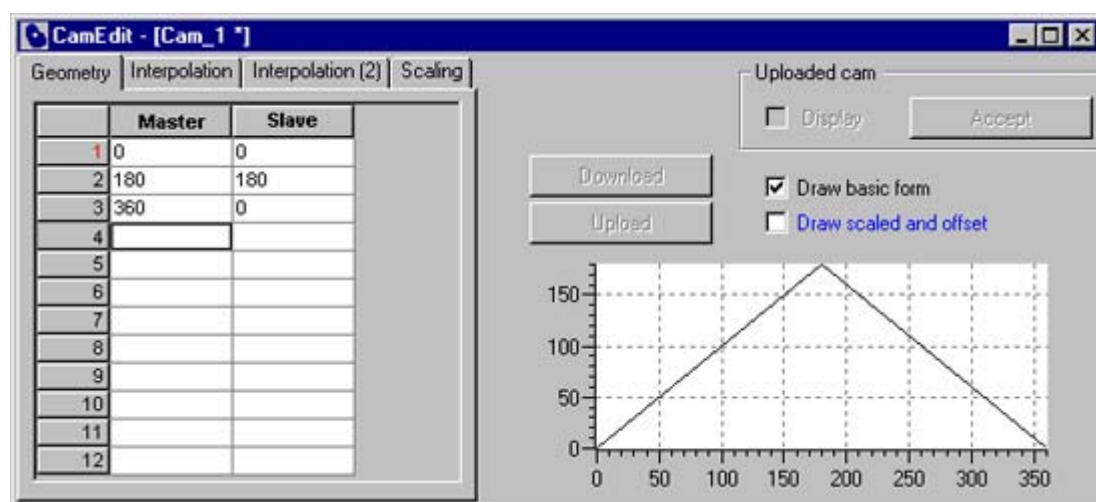
4.8.4 使用 CamEdit 创建凸轮

4.8.4.1 使用 CamEdit 创建凸轮

CamEdit 对话框可用于执行以下任务：

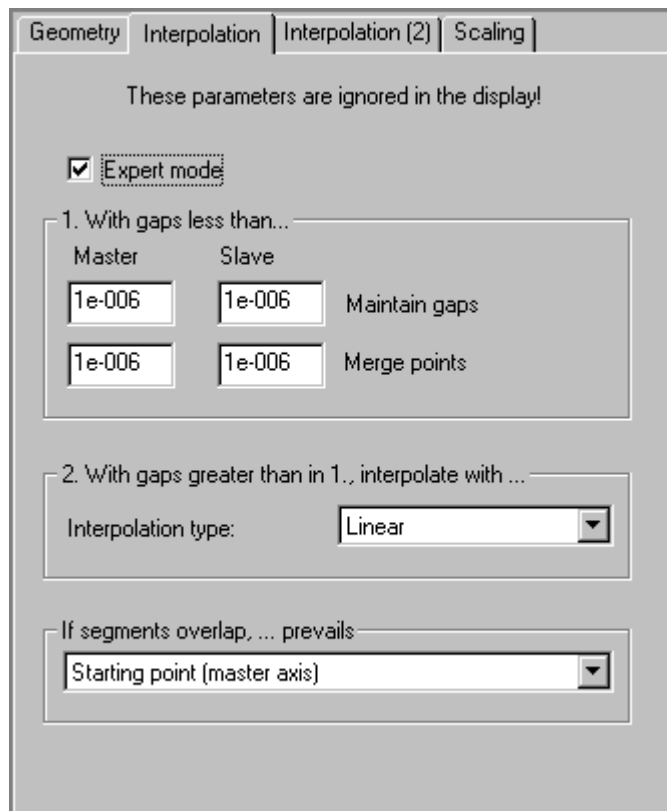
- 通过输入凸轮盘的区段或插补点来定义凸轮盘
- 使用 VDI 向导来创建带有已定义运动转换的凸轮盘。
- 编辑凸轮盘的凸轮
- 将凸轮盘的凸轮下载到 PLC（仅当凸轮盘未使用时可以进行）
- 从 PLC 中读取凸轮
- 查看凸轮的几何图形。将编程设备/PC 上的凸轮与目标设备（已缩放/未缩放的显示）中的凸轮进行比较。

如果在“MC_CamInterpolate”工艺功能中进行比较，则忽略 **CamEdit** 中的插补设置。凸轮不能在用户程序中进行缩放和移位。



4.8.4.2 插补

选择 **CamEdit** 对话框的“插补”(Interpolation) 选项卡，可以定义是否保留凸轮盘区段之间的间隔，或者是否在特定区段的端点之间插补凸轮。间隔在曲线特性中是不一样大的。只要凸轮区段不重叠便会出现间隔。如果个别区段重叠，您可以选择哪个凸轮区段具有优先级。



显示的设置可以通过选中“专家模式”(Expert mode) 复选框来设置：

间隔最多可长达.....

该功能仅在专家模式下可用。

- 保留间隔

选择“主”(Master) 或“从”(Slave) 字段的“保留间隔”(Retain gaps) 行，可以定义要在“主轴”和“从轴”的凸轮盘区段之间保留的间隔长度。

- 连接点

选择“主”(Master) 或“从”(Slave) 字段的“连接点”(Join points) 行，可以定义要在其端点上连接的“主轴”(master axis) 和“从轴”(slave axis) 区段之间的最大间隔长度。

如果间隔长度大于在 1 中定义的长度，则使用下列方式进行插补.....

在“插补模式”下，您可以选择间隔之间的长度大于以前定义的最大值的插补模式。

- **线性**（线性插补）
通过在插补点之间插入线性区段来持续连接间隔。
- **C 样条**（使用 C 样条的插补）
连续可区分地连接间隔。曲线贯穿指定的插补点。
- **B 样条**（使用 B 样条的插补）
逼近趋势沿定义的插补点延伸。

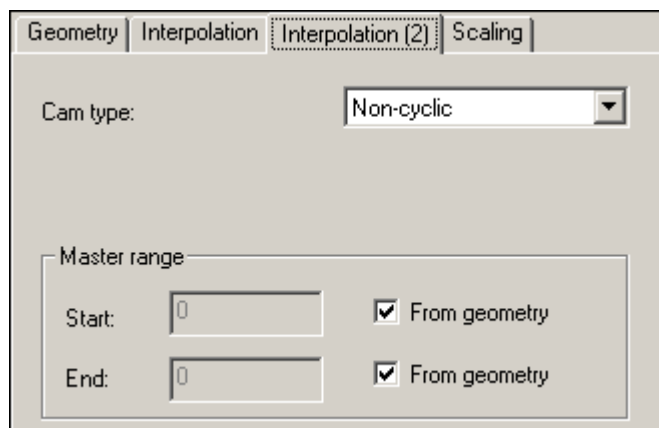
如果区段重叠，则优先级属于.....

选择哪个重叠凸轮盘区段应该具有优先级。该功能仅在专家模式下可用。

- **起点（主轴）**
凸轮曲线在下一个区段的起点继续。
- **终点（主轴）**
下一个区段的凸轮曲线在当前区段的终点开始。
- **时间顺序**
凸轮曲线取决于区段插入的时间顺序。最后插入的区段具有最高优先级，而首先插入的区段则具有最低优先级。

4.8.4.3 插补(2)

在 CamEdit 的“插补(2)”(Interpolation[2]) 选项卡中设置凸轮盘引导轴值的类型和范围。这些设置构成了插补条件。



凸轮类型

在下拉列表框中设置凸轮盘的操作模式：

- **非周期性**
在周期模式下不使用凸轮盘
- **周期性（绝对）**
凸轮盘在周期性凸轮传动模式下操作，与跟随轴无关
- **周期性（相对）**
凸轮盘在周期性凸轮传动模式下相对于跟随轴操作。

主范围（引导轴值的范围）

您可从中设置引导轴值的范围：

- **起点**
通过使用数字或设置复选标记来定义。通过该设置，起点从凸轮盘的现有几何图形中得出。
- **终点**
通过使用数字或设置复选标记来定义。通过该设置，终点从凸轮盘的现有几何图形中得出。

4.8.4.4 缩放和移位

在 **CamEdit** 中，您可以将凸轮盘曲线缩放（拉伸或压缩）和移位。可将缩放和移位应用于引导轴和跟随轴的值。

您可以缩放整个凸轮盘，也可以缩放通过起点和终点定义的范围：

- 您可以为整个凸轮盘定义基本缩放和移位
- 范围缩放使您可以将选定的凸轮区段缩放和移位。通过其起点和终点定义范围。

不能使用用户程序修改凸轮盘的缩放比例和偏移量。

在 **CamEdit** 对话框中选择“缩放”(Scaling) 选项卡，以定义引导轴和跟随轴的缩放和移位。

The screenshot shows the 'Scaling' tab of the CamEdit dialog box. It contains two sections: 'Master axis' and 'Slave axis'. Each section has a 'Basic scaling' field, a 'Scaling' table with 'From', 'To', and 'Factor' columns, and an 'Offset' field.

Master axis			
Basic scaling:	1		
Scaling:	From	To	Factor
	1) 0	90	2
	2) 90	180	2
Offset:	0		

Slave axis			
Basic scaling:	1		
Scaling:	From	To	Factor
	1) 0	0	1
	2) 0	0	1
Offset:	0		

引导轴值（主轴）范围内的缩放和移位设置：

- **基本缩放 (Basic scaling)**

在“主轴的基本缩放”(Basic scaling of the master axis) 对话框中输入引导轴的缩放系数。坐标原点始终作为缩放的中心。

不要使用大于 5 的缩放比例值。过大的值会导致凸轮曲线粗糙。

- **缩放 1) 和缩放 2)**

使用缩放对话框可以定义“主轴”的最多两个用户特定范围（引导轴值的范围）的缩放。使用“从”(From) 和“至”(To) 指定要缩放的范围。在“因子”(Factor) 下面输入缩放因子。不能使两个缩放范围重叠。缩放中心是缩放范围的起点。

- **移位 (Shift)**

在“移位”(Shift) 下，可以输入主轴（引导轴值）的移位系数。移位可以应用于已缩放的凸轮或基本形式（如果尚未缩放范围）。

跟随轴值（从轴）范围内的缩放和移位设置：

- **基本缩放 (Basic scaling)**

在“从轴的基本缩放”(Basic scaling of the slave axis) 对话框中输入跟随轴的缩放系数。坐标原点始终形成缩放的中心。不要使用大于 5 的缩放比例值。过大的值会导致凸轮曲线粗糙。

- **缩放 1) 和缩放 2)**

使用缩放对话框可以定义“从轴”的最多两个用户特定范围（跟随轴值的范围）的缩放。使用“从”(From) 和“至”(To) 指定要缩放的范围。在“因子”(Factor) 下面输入缩放因子。不能使两个缩放范围重叠。缩放中心是缩放范围的起点。

- **移位 (Shift)**

在“移位”(Shift) 下，可以输入从轴（跟随轴值）的移位系数。移位可以应用于已缩放的凸轮或基本形式（如果尚未缩放范围）。

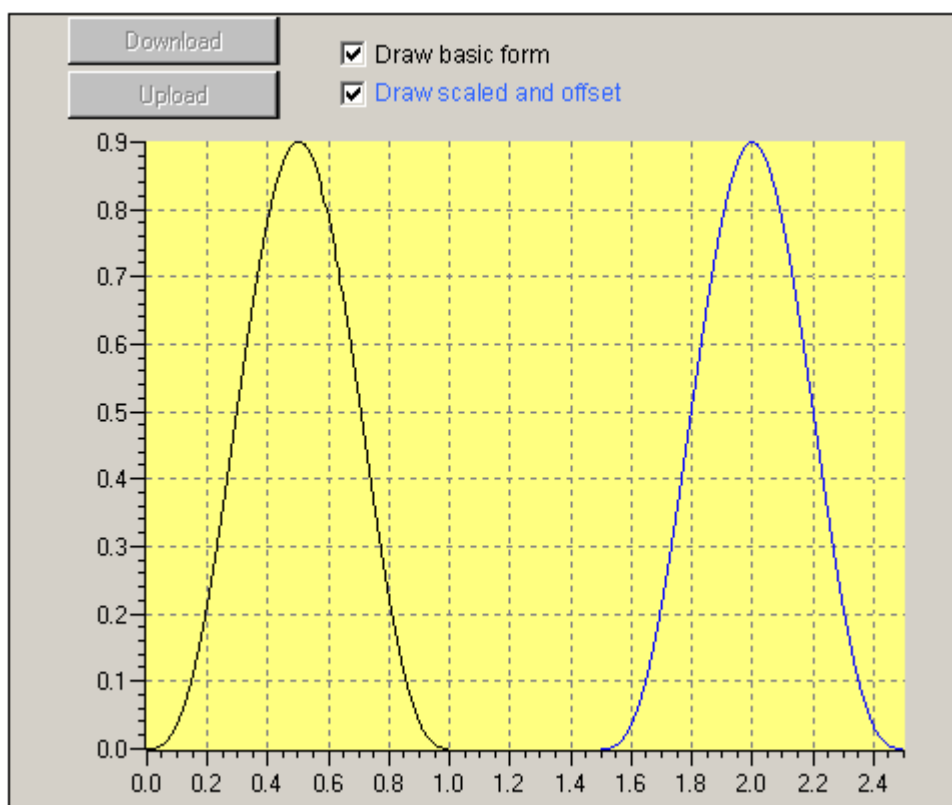
说明

如果在多个对象中使用了缩放和/或移位的凸轮盘，则缩放和移位设置会影响所有相关的对象。

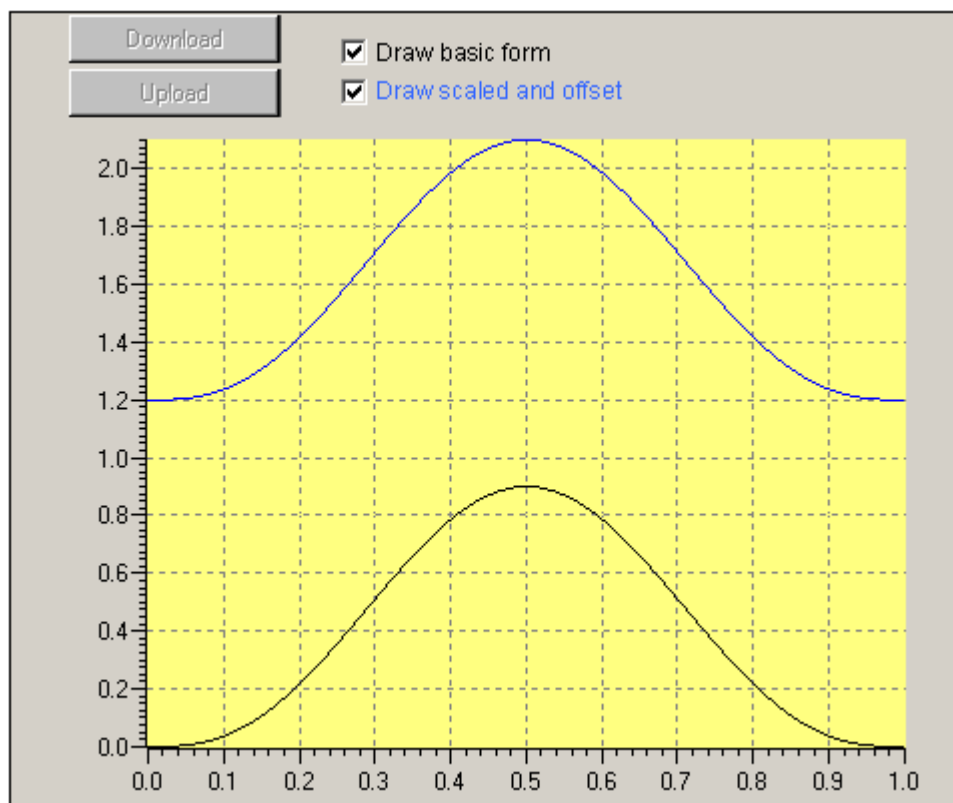
凸轮缩放和移位示例

以下示例以图示说明了对凸轮进行缩放和移位的影响。黑线代表原凸轮。蓝线代表已缩放或移位的凸轮。

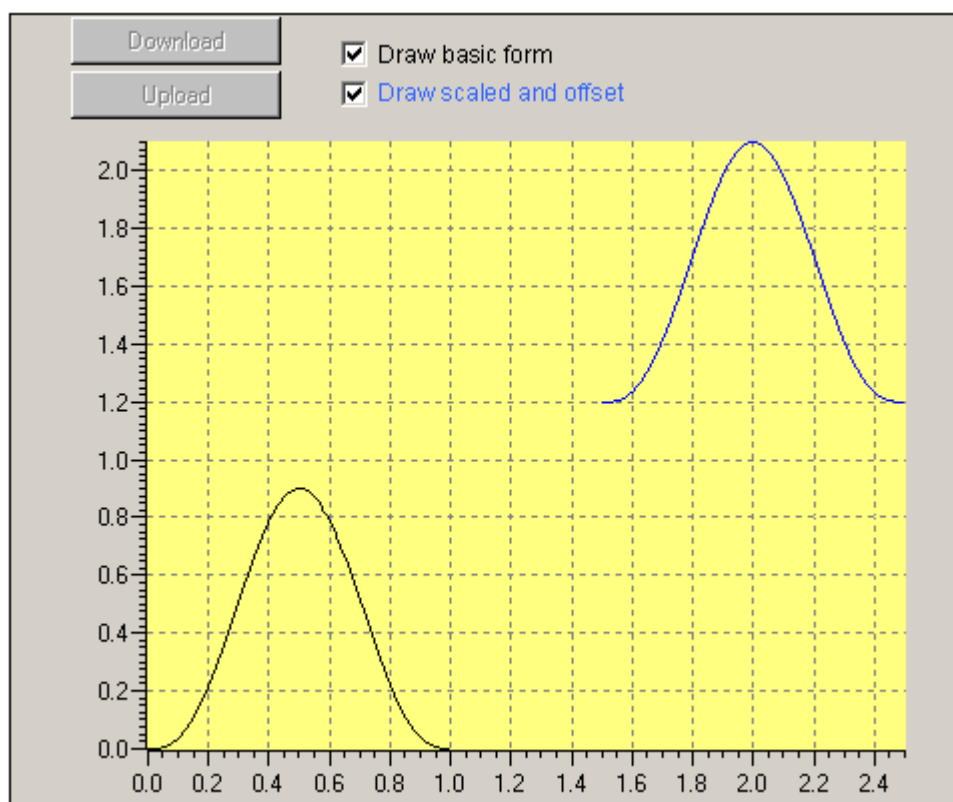
引导轴值的移位：



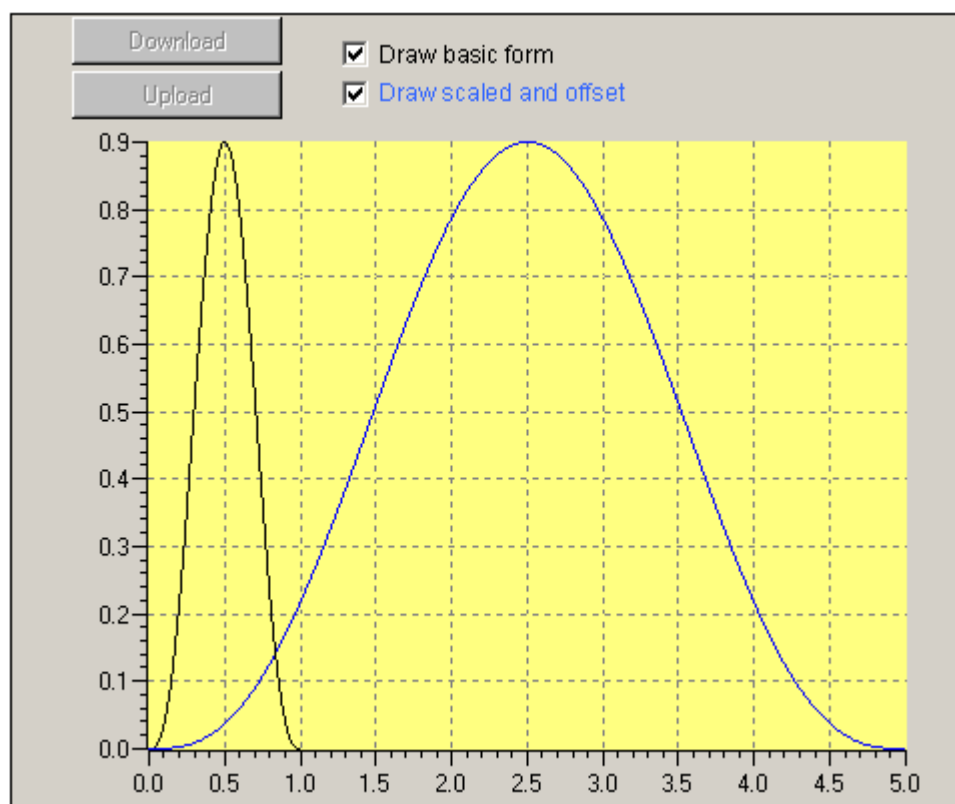
跟随轴值的移位:



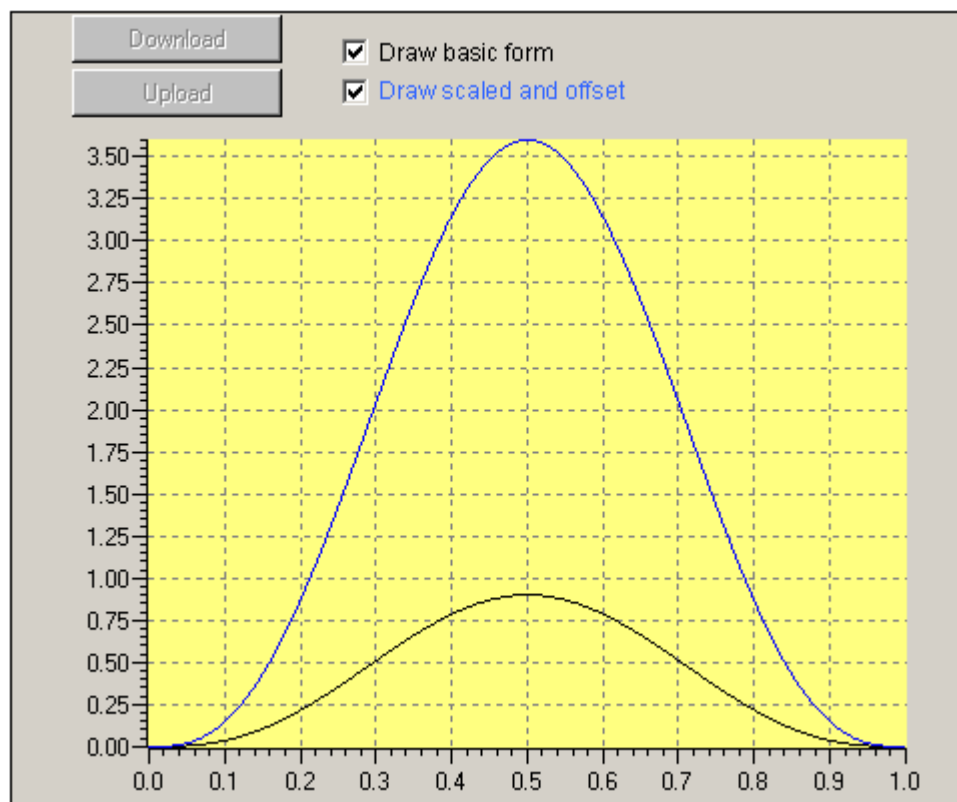
引导轴值和跟随轴值的移位:



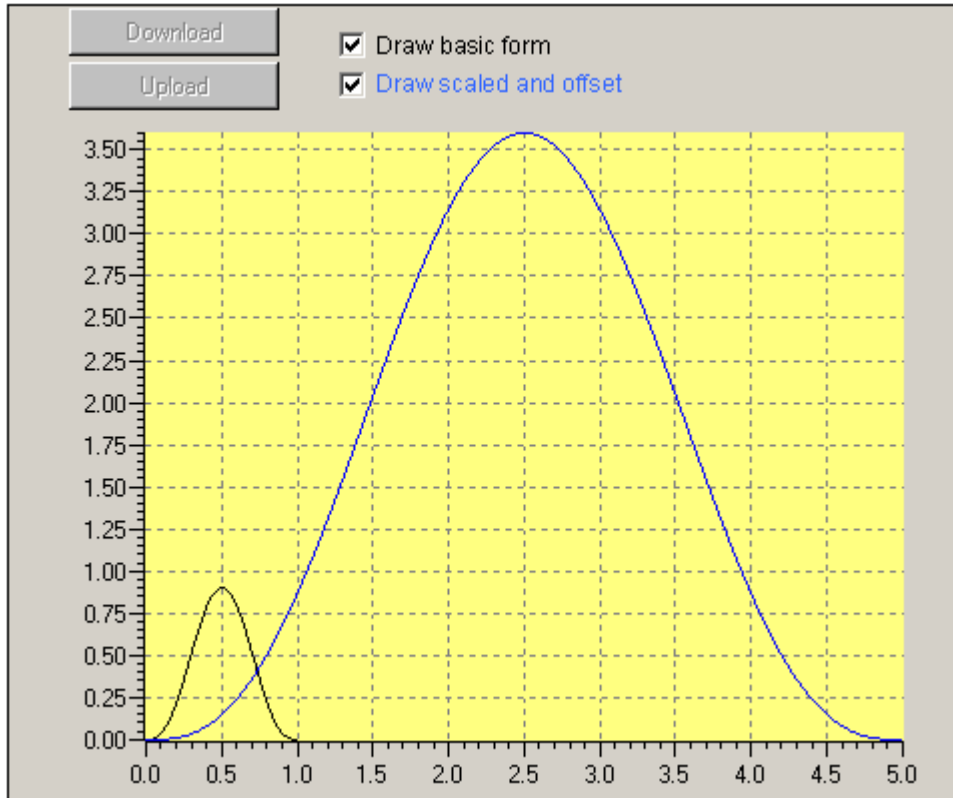
引导轴值的缩放:



跟随轴值的缩放:

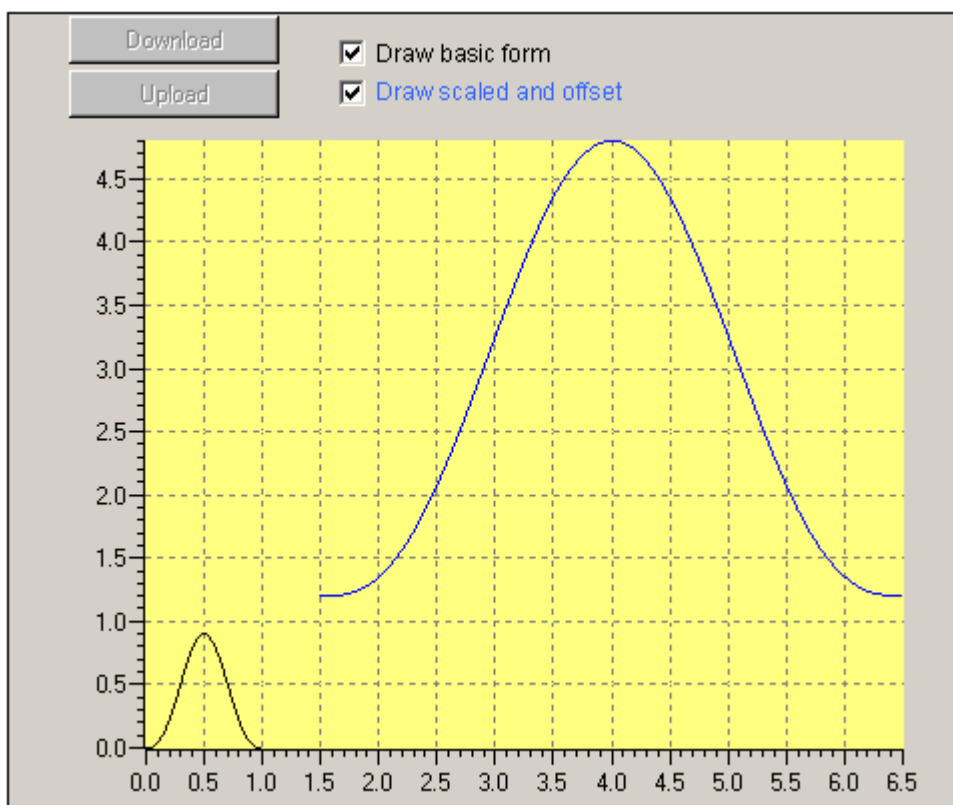


引导轴值和跟随轴值的缩放:



引导轴值和跟随轴值的移位

引导轴值和跟随轴值的缩放:



4.8.5 使用 CamTool 创建凸轮

还可以使用 SCOUTCamTool 创建和编辑凸轮盘。此工具必须单独购买。

SCOUTCamTool 具有以下优点:

- 能够以图形精确直观地反映凸轮
- 通过拖放凸轮元素快速、简单地定义凸轮
- 通过“在曲线上拖动”快捷、简单地调节凸轮
- 同时直观地反映位置、速度、加速度和冲击曲线对最大速度、所需的电机力矩和机械负载产生的直接影响。
- 调节凸轮的速度、加速度或冲击参数

SCOUTCamTool 可以完全集成在 S7T Config 的用户界面中。

有关此软件的功能和操作的详细信息, 请参考《SIMOTIONCamTool》手册。

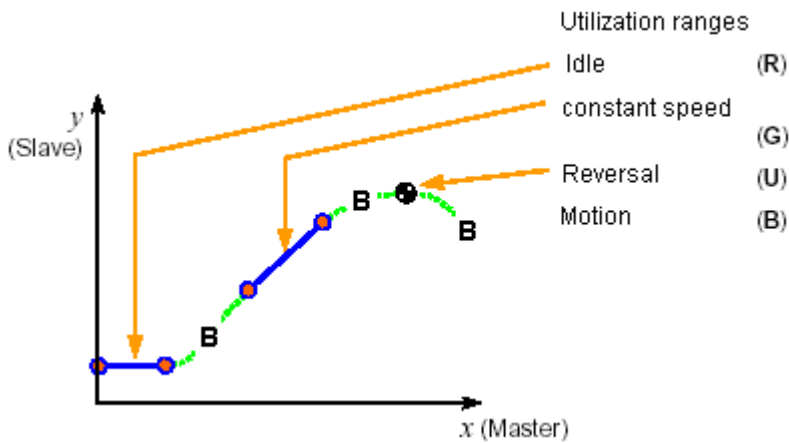
4.8.6 符合 VDI 的运动法则

4.8.6.1 工作范围和运动转换

VDI 概念用于区分工作范围和运动转换：

- 与过程中的顺序相对应的工作范围。VDI 概念可以区分以下四个工作范围：
- 运动转换代表工作范围间的转换。虽然这些转换与过程并不直接相关，但它们必须满足某些条件，例如恒定速度和加速度。

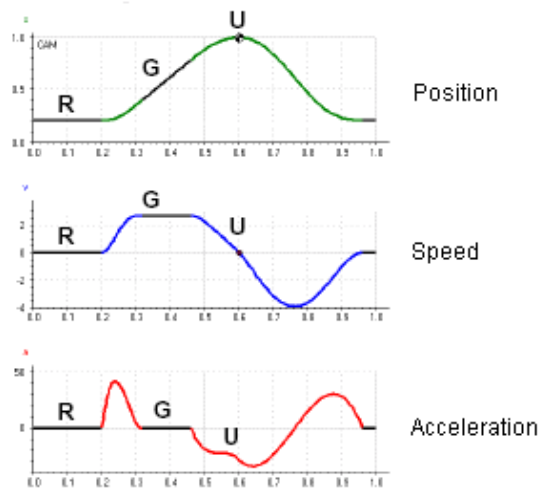
工作范围



VDI 概念可以区分下列工作范围：

	工作范围	速度 (v)	加速度 (a)
R	空闲	= 0	= 0
G	恒定速度	≠ 0	= 0
U	反向	= 0	≠ 0
B	运动	≠ 0	≠ 0

具有三个工作范围的凸轮盘示例



运动转换

处于各种工作范围的可能的运动转换：

	R	G	U	B
R				
G				
U				
B				

说明

VDI 向导支持在 S7T Config 中创建凸轮盘。您可以在 **CamEdit** 对话框中打开此向导。

4.8.6.2 用区段定义凸轮盘

工作范围定义

运动任务的**工作范围**通常由过程指定。

示例：

1. 工具在生产线上等待零件经过（空闲状态）。
2. 此工具与零件同步并在加工件上执行任务（恒定速度）。
3. 然后，此工具返回等待位置（反向）。

此过程从头重新开始。

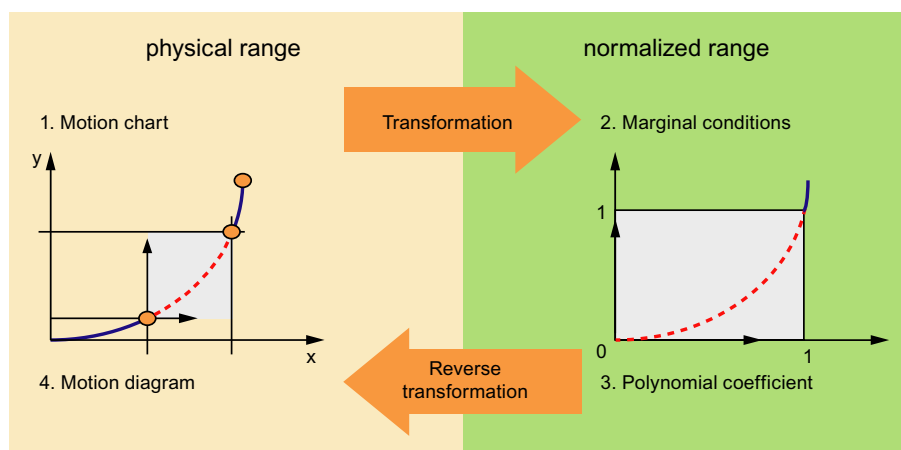
为了实现此顺序，凸轮盘首先必须根据工作范围进行分段。

创建运动转换

在下一步中，定义符合某些条件（例如无冲击运动）的运动转换。

- 通过将运动转换转换到标准范围开始。
- 请顾及在区段限制中的位置、速度和加速度等条件。
- 为了应用以此方法定义的多项式，它必须转换回实际范围。

创建符合 VDI 的运动转换：



参考资料

- VDI 准则 2143, Bl. 1: 《凸轮机械的运动法则 — 杜塞尔多夫基本理论》：VDI 出版社，1980 年出版
- Volmer, J. (编辑): 《机械设计 — 凸轮机械》第 2 次于柏林印刷，技术出版社，1989 年出版

4.9 组态输出凸轮

4.9.1 组态输出凸轮

任何要在用户程序中使用的输出凸轮必须已插入 **S7T Config** 作为“输出凸轮”工艺对象。

您可以使用 **MC_CamSwitch**（基于位置的输出凸轮/切换输出凸轮）或 **MC_CamSwitchTime** 工艺功能（基于时间的输出凸轮）在用户程序中对此对象进行编程。通过在输入 **CamSwitch** 中设置相应的工艺 DB 号，来定义在 **S7T Config** 中创建的输出凸轮的分配。

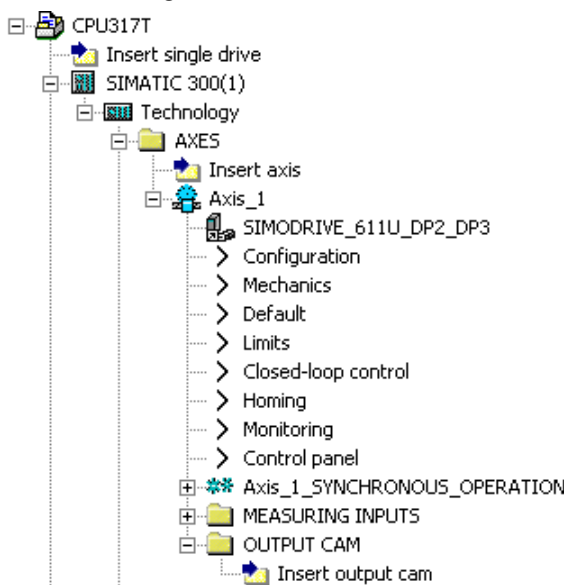
ON/OFF 条件直接在块中组态。

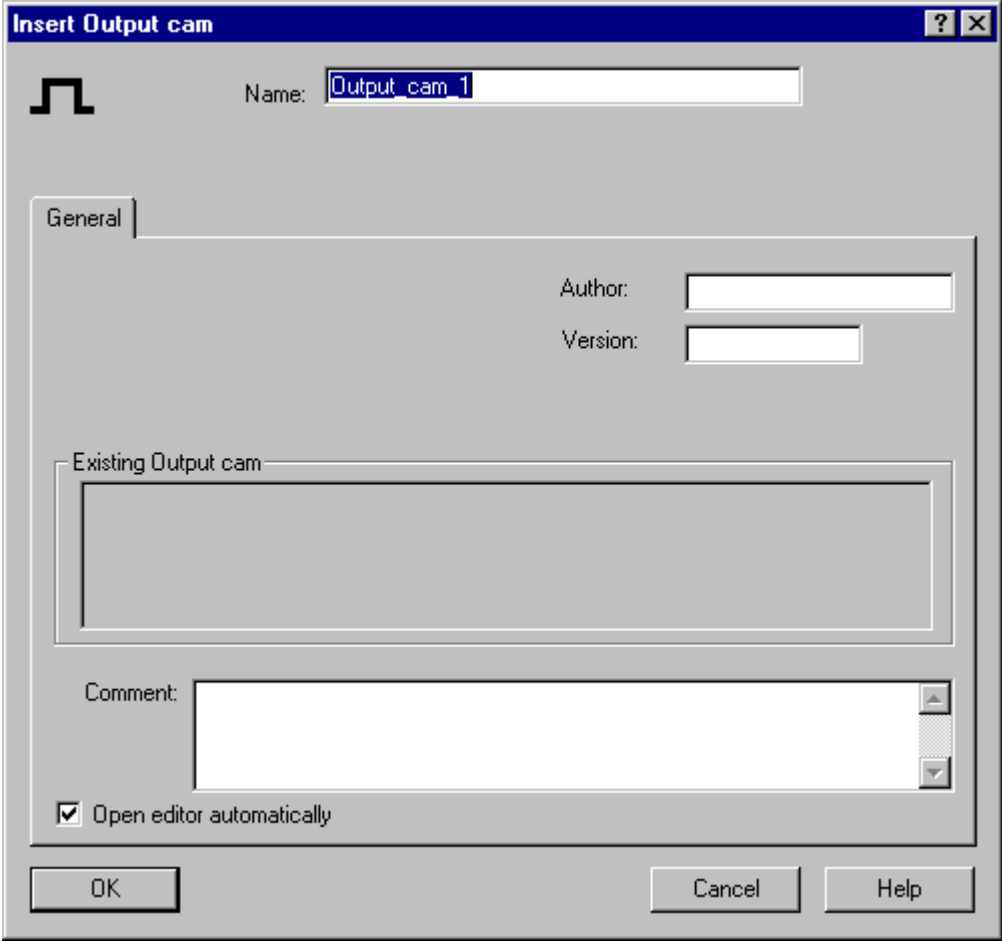
4.9.2 插入输出凸轮

先决条件

- T-CPU 的数据已在 **HW Config** 中进行组态并编译。
- 已在 **S7T Config** 中创建轴或外部编码器。

如何将输出凸轮插入 **S7T Config** 中

步骤	说明
1.	在 S7T Config 的浏览器中，双击“插入输出凸轮”(Insert output cam)。  The screenshot shows a hierarchical tree structure in the S7T Config browser. The path is: CPU317T > SIMATIC 300(1) > Technology > AXES > Insert axis > Axis_1 > SIMODRIVE_611U_DP2_DP3 > OUTPUT CAM > Insert output cam. The 'Insert output cam' icon is highlighted at the bottom of the tree.

步骤	说明
2.	在“插入输出凸轮”(Insert output cam) 对话框中输入工艺对象的名称。 将另外提供用于输入作者、版本号和注释的域。 激活“自动打开编辑器”(Open editor automatically) 复选框，以便自动打开输出凸轮组态对话框： 
3.	单击“确定”(OK) 确认输入。 结果： 工艺对象随即插入。

其它步骤

在**输出凸轮 (Output cam) > 组态 (Configuration)** 中组态输出凸轮。

4.9.3 组态 — 输出凸轮

在**输出凸轮 (Output cam) > 组态 (Configuration)** 对话框中，您可以为“输出凸轮”工艺对象设置以下条目：

- 输出凸轮类型 (Output cam type): 基于位置的凸轮、基于时间的凸轮或切换凸轮
- 计算凸轮的系统周期
- 输出凸轮的参考值: 设定值或实际值

设定值参考的规则：

如果凸轮在位置控制周期内没有得到处理，则其必须与轴在同一个周期内运行。

- 将输出凸轮作为高速输出凸轮操作

Output_cam_1 - Configuration

Name: Output_cam_1

Output cam type: Position-based cam

Processing cycle clock: IPO

Type of output cam values: Setpoints

High-speed output cam onboard ☐ High-speed output cam on the TM15/TM17 module ☐

Output

☒ Activate output

Logical operation: Logical OR

HW address: 66 Bit number: 0

Close Help

说明

输出凸轮在非主轴上同样有效。

输出

选中“启用输出”(Enable output) 复选框可以为“输出凸轮”工艺对象分配硬件地址和逻辑运算。如果禁用输出，则输出凸轮只能在软件中进行诊断。

硬件地址

输出凸轮只能分配给一个输出。这可以是 T-CPU 的集成数字输出、DP(DRIVE) 上的数字输出（例如在 ET200 站中）或者 TM15 或 TM17 高性能型模块的输出。

可以将多个输出凸轮连接到同一个输出。

输出凸轮的切换精度取决于：

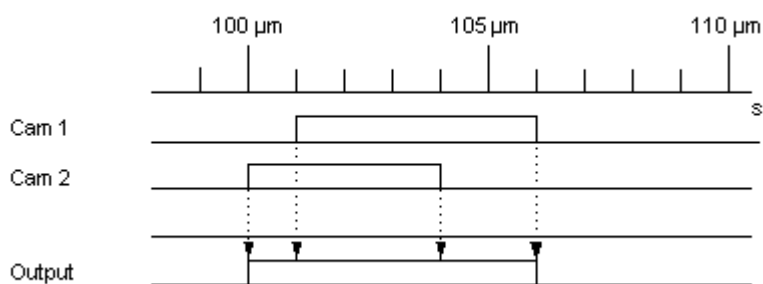
- I/O 的输出精度
- 输出凸轮在系统周期中的优先级
- 恒定延迟时间的补偿

逻辑运算

在**输出凸轮 > 组态 (Output cam > Configuration)** 对话框中，您可以确定是通过逻辑 AND 运算还是通过逻辑 OR 运算将输出凸轮与输出互联。

即所有使用 OR 的输出凸轮将被编组，然后在输出中与通过 AND 逻辑链接的输出凸轮进行逻辑链接。

OR 逻辑具有两个输出凸轮



4.9.4 高速输出凸轮

在**输出凸轮 > 组态 (Output cam > Configuration)** 对话框中，您可以激活“高速输出凸轮”功能。

只有 T-CPU 的集成输出或者 TM15 或 TM17 高性能型模块的输出支持高速输出凸轮。内部硬件功能仅支持这些凸轮。

说明

高速输出凸轮和标准输出凸轮不能共享相同的硬件地址和位号。务必使用不同的地址

通过中断执行输出凸轮输出，而与处理周期无关。

根据 IPO 周期或位置控制器周期计算输出凸轮的状态。使用集成输出时，如果需要最新的状态显示，则在对话框**“输出凸轮”(Output cam) > “组态”(Configuration)** 中选择位置控制器周期作为处理周期。使用 TM15 和 TM17 输出凸轮时，处理周期不会影响状态计算。

4.10 组态凸轮轨迹

4.10.1 组态凸轮轨迹

将凸轮轨迹用于用户程序之前，必须首先将凸轮轨迹作为轴或外部编码器的“凸轮轨迹”工艺对象插入 S7T Config。

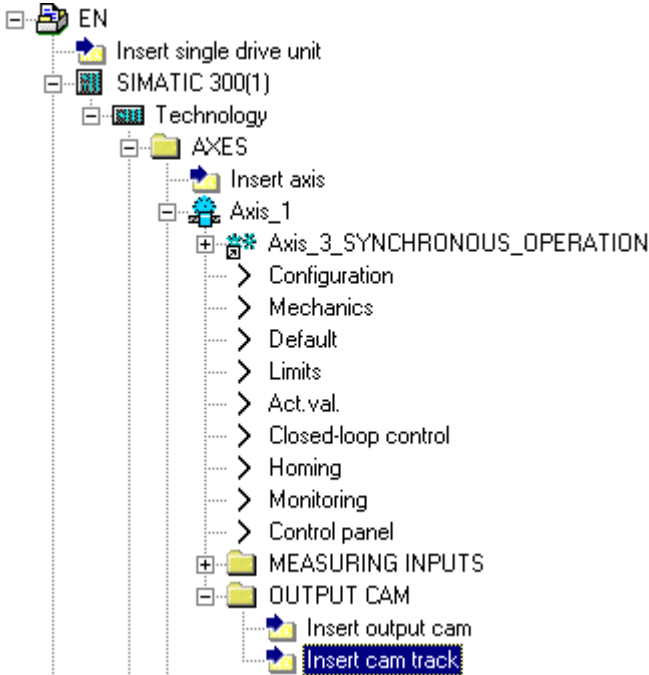
通过为用户程序中调用工艺功能“MC_CamTrack”，可使用在 S7T Config 中组态的凸轮轨迹。通过在输入参数 *CamTrack* 中输入工艺 DB 的编号，可定义在 S7T Config 中创建的凸轮轨迹的分配。

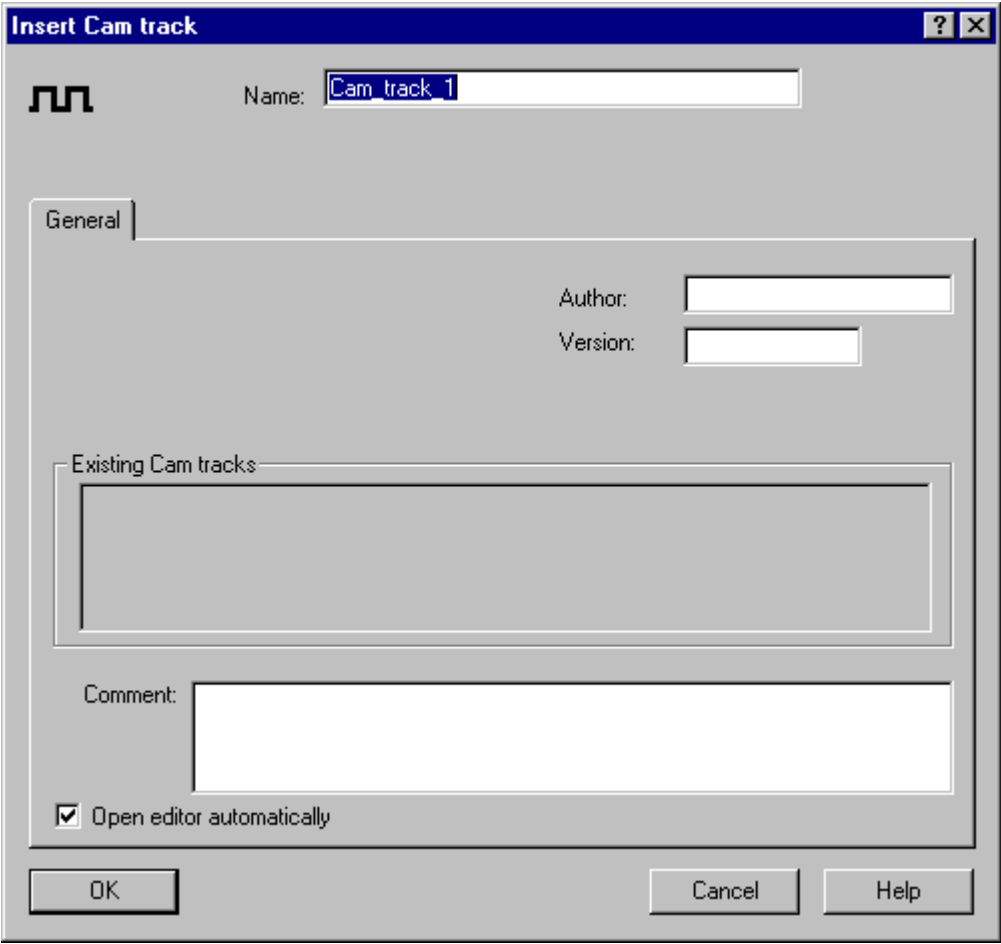
4.10.2 插入凸轮轨迹

要求

- T-CPU 的数据已在 HW Config 中进行组态并编译。
- 已在 S7T Config 中插入轴或外部编码器。

如何在 S7T Config 中插入凸轮轨迹

步骤	说明
1.	在 S7T Config 的浏览器中，双击“插入凸轮轨迹”(Insert cam track)。  The screenshot shows a hierarchical tree structure in the S7T Config browser. The path is as follows: EN (Expanded) -> Insert single drive unit (Expanded) -> SIMATIC 300(1) (Expanded) -> Technology (Expanded) -> AXES (Expanded) -> Insert axis (Expanded) -> Axis_1 (Expanded) -> Axis_3_SYNCHRONOUS_OPERATION (Expanded) -> OUTPUT CAM (Expanded) -> Insert output cam (Expanded) -> Insert cam track (Selected). The 'Insert cam track' item is highlighted with a blue selection box.

步骤	说明
2.	在“插入输出凸轮”(Insert output cam) 对话框中输入工艺对象的名称。 将另外提供用于输入作者、版本号和注释的域。  激活“自动打开编辑器”(Open editor automatically) 复选框，以便自动打开凸轮轨迹组态对话框：
3.	单击“确定”(OK) 确认输入。 结果：工艺对象随即插入。

其它步骤

在“凸轮轨迹”(Cam track) > “组态”(Configuration) 对话框中组态凸轮轨迹。

4.10.3 组态 — 凸轮轨迹

“凸轮轨迹”(Cam track) > “组态”(Configuration) 对话框可用于设置“凸轮轨迹”工艺对象的以下参数：

The screenshot shows the 'S7-Technology.Cam_track_1 - Configuration' dialog box. The fields are as follows:

- Name: Cam_track_1
- Output cam type: Position-based cam
- Processing cycle clock: IPO
- Type of output cam values: Setpoints
- Fast output cams onboard: ☐
- Fast output cams on TM15/TM17 module: ☐
- Output section:
 - ☒ Assign output
 - ☐ Output output cams inverted
 - HW address: 64
 - Bit number: 0
- Leave non-cyclic activated cam track outside of the track range activated: Yes

- **名称 (Name):**

显示在 S7T Config 浏览器中定义的工艺对象的名称。

- **输出凸轮类型 (Output cam type):**

在此处指定专用输出凸轮的类型（基于位置的凸轮、基于时间的凸轮或具有最大 ON 长度的基于时间的凸轮）。设置的凸轮类型适用于凸轮轨迹的所有专用输出凸轮。

- **执行周期 (Execution cycle):**

执行周期用于指定执行凸轮轨迹的周期（IPO、IPO2 或伺服时钟/位置控制器周期）。

- **输出凸轮值的类型 (Type of output cam value):**

指定是将轴或外部编码器的设定值还是实际值用于当前输出凸轮切换点。

设定值参考的规则： 如果凸轮轨迹在位置控制周期内没有得到处理，则必须与轴在同一个周期内运行。

- **集成高速输出凸轮 (Integrated high-speed output cams)/TM15/TM17 模块上的高速输出凸轮 (high-speed output cams on TM15/TM17-Modul)**

如果想将凸轮轨迹的输出切换为高速输出凸轮，可选中两个复选框之一。 可以在 T-CPU 集成 I/O 的高速输出凸轮与 TM15/TM17-Modul. 的高速输出凸轮之间进行选择。

输出

如果激活了“分配输出”(Assign output) 复选框，则可以进一步进行选择：

- **输出凸轮反向输出(Output output cam inverted)**

如果要将凸轮轨迹的切换状态反向输出到硬件输出，则选中此复选框。 此设置不会影响专用输出凸轮的切换状态。

- **硬件地址 (HW address)/位号 (bit number)**

在这两个域中，输入凸轮轨迹的硬件输出的硬件地址和位号。 为凸轮轨迹分配专用输出。这可以是 T-CPU 的集成数字输出、DP(DRIVE) 上的数字输出（例如在 ET200 站中）或者 TM15 或 TM17 高性能型模块的输出。

- **在轨迹范围外激活非循环激活的凸轮轨迹。(Non-cyclic activated cam track can be activated outside the track range.)**

- 下拉列表框“是”(Yes)

如果轴或外部编码器移动到凸轮轨迹之外，则凸轮轨迹保持活动状态。 如果轴或外部编码器随后又移回到凸轮轨迹范围内，则已组态的凸轮将重新切换。 在 *Mode = 1* 时，利用工艺功能“MC_CamTrack”可以锁定和/或取消激活凸轮轨迹。

- 下拉列表框“否”(No)

如果轴或外部编码器移动到凸轮轨迹之外，则凸轮轨迹被取消激活。 重新进入凸轮轨迹的范围不会触发凸轮轨迹的任何切换过程。

说明

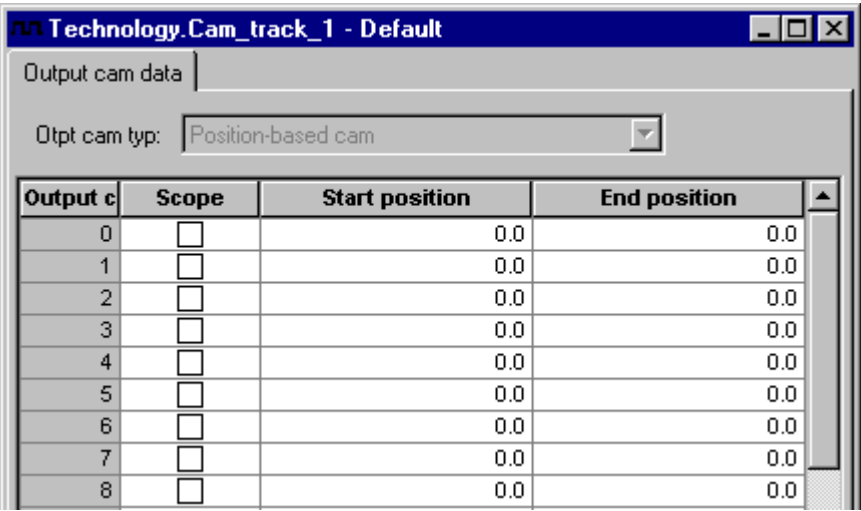
凸轮轨迹在非主轴上同样有效。

4.10.4 默认值 — 凸轮轨迹

在“凸轮轨迹”(Cam track) > “默认值”(Default) 对话框中指定凸轮轨迹的专用输出凸轮的输出凸轮数据。

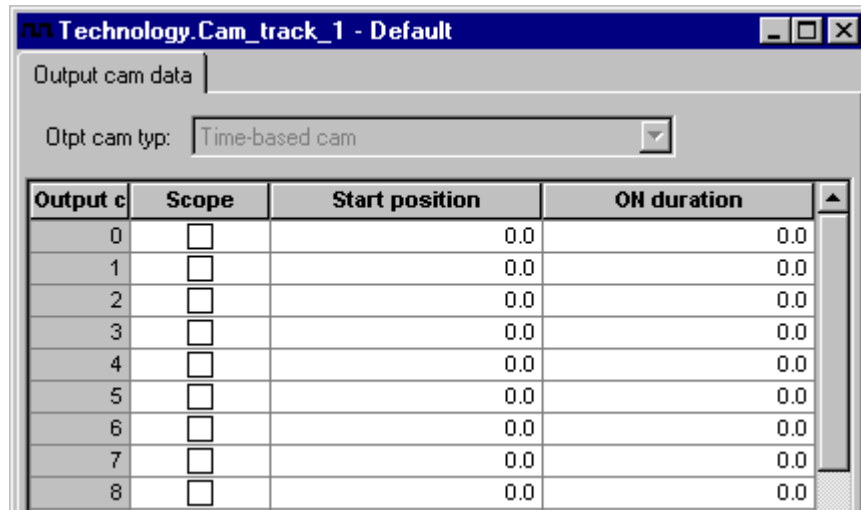
根据凸轮类型，专用输出凸轮 0 至 31 的可用输入域有所不同：

基于位置的凸轮



- **有效性 (Validity)**
如果激活了该复选框，则在编辑凸轮轨迹时会考虑专用输出凸轮。 如果不需要专用输出凸轮，或者专用输出凸轮位于轨迹长度之外，则将其设置为无效。
- **起始位置 (Start position)**
在此输入框中定义专用输出凸轮的起始位置（称为凸轮轨迹起点）。
- **结束位置 (End position)**
在此输入框中定义专用输出凸轮的结束位置（称为凸轮轨迹终点）。 如果运动方向为正，则专用输出凸轮在到达起始位置时打开，而在结束位置再次关闭。 如果运动方向为负，则专用输出凸轮在到达结束位置时打开，而在起始位置再次关闭。

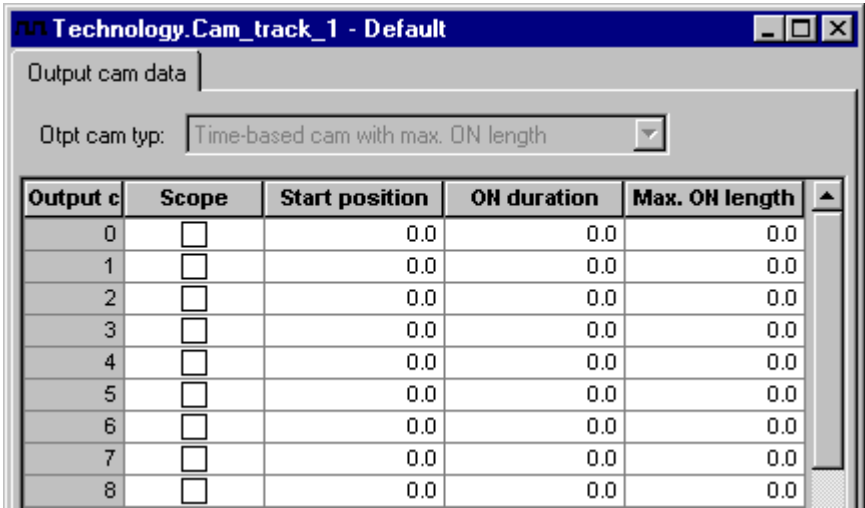
基于时间的凸轮



Output c	Scope	Start position	ON duration
0	☐	0.0	0.0
1	☐	0.0	0.0
2	☐	0.0	0.0
3	☐	0.0	0.0
4	☐	0.0	0.0
5	☐	0.0	0.0
6	☐	0.0	0.0
7	☐	0.0	0.0
8	☐	0.0	0.0

- **有效性 (Validity)**
如果激活了该复选框，则在编辑凸轮轨迹时会考虑专用输出凸轮。 如果不需要专用输出凸轮，或者专用输出凸轮位于轨迹长度之外，则将其设置为无效。
- **起始位置 (Start position)**
在此输入框中定义专用输出凸轮的起始位置（称为凸轮轨迹起点）。
- **ON 时间 (ON time)**
在此输入框中定义专用输出凸轮的 ON 时间。 专用输出凸轮在到达起始位置时打开，而在经过了 ON 时间后再次关闭。

具有最大 ON 长度的基于时间的输出凸轮



- **有效性 (Validity)**
如果激活了该复选框，则在编辑凸轮轨迹时会考虑专用输出凸轮。 如果不需要专用输出凸轮，或者专用输出凸轮位于轨迹长度之外，则将其设置为无效。
- **起始位置 (Start position)**
在此输入框中定义专用输出凸轮的起始位置（称为凸轮轨迹起点）。
- **ON 时间 (ON time)**
在此输入框中定义专用输出凸轮的 ON 时间。
- **最大 ON 长度(Max. ON length)**
在此输入框中指定专用输出凸轮的最大 ON 长度。 专用输出凸轮在到达起始位置时打开。 经过了 ON 时间或超出了 ON 长度后，专用输出凸轮会再次关闭。

说明

要读出凸轮轨迹的凸轮数据，可在用户程序中使用工艺功能“MC_ReadCamTrackData”。
工艺功能“MC_WriteCamTrackData”可用于将已修改的凸轮数据重新写入凸轮轨迹。

4.11 组态测量输入

4.11.1 组态测量输入

任何在用户程序中使用的测量输入都必须插入 **S7T Config** 作为“测量输入”工艺对象。

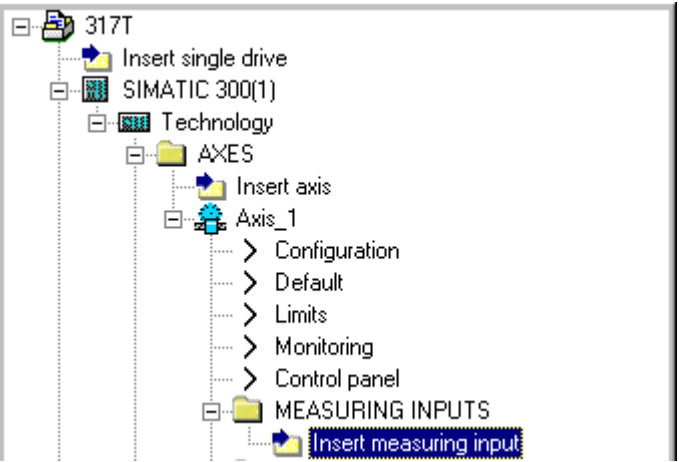
可以使用“**MC_MeasuringInput**”工艺功能在用户程序中对测量输入进行编程。这些功能用于根据各种操作模式设置启用和禁用测量输入，并定义其测量范围。通过在输入参数 *MeasuringInput* 中输入工艺 DB 号，定义在 **S7T Config** 中创建的测量输入的分配。

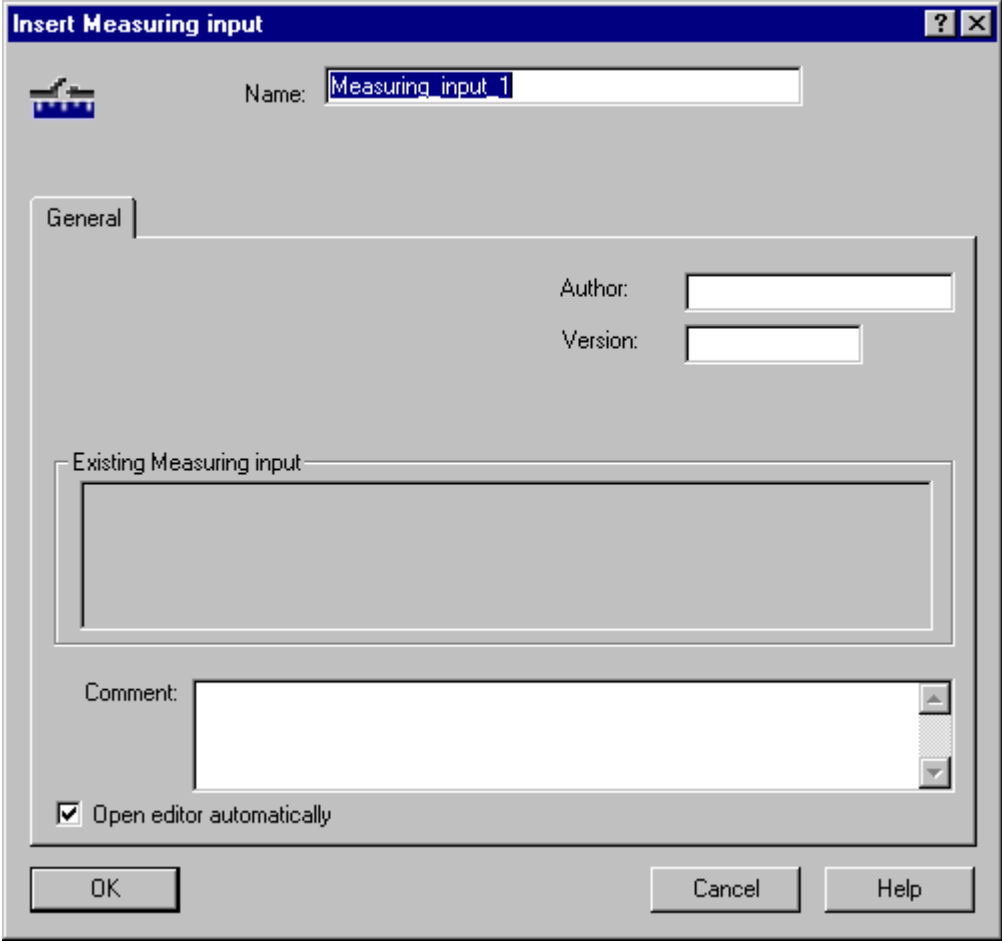
4.11.2 插入测量输入

先决条件

- T-CPU 的数据已在 **HW Config** 中进行组态并编译。
- 已在 **S7T Config** 中插入轴或外部编码器。

如何将测量输入插入 **S7T Config** 中

步骤	说明
1.	在 S7T Config 的浏览器中，双击“插入测量输入”(Insert measuring input)。 

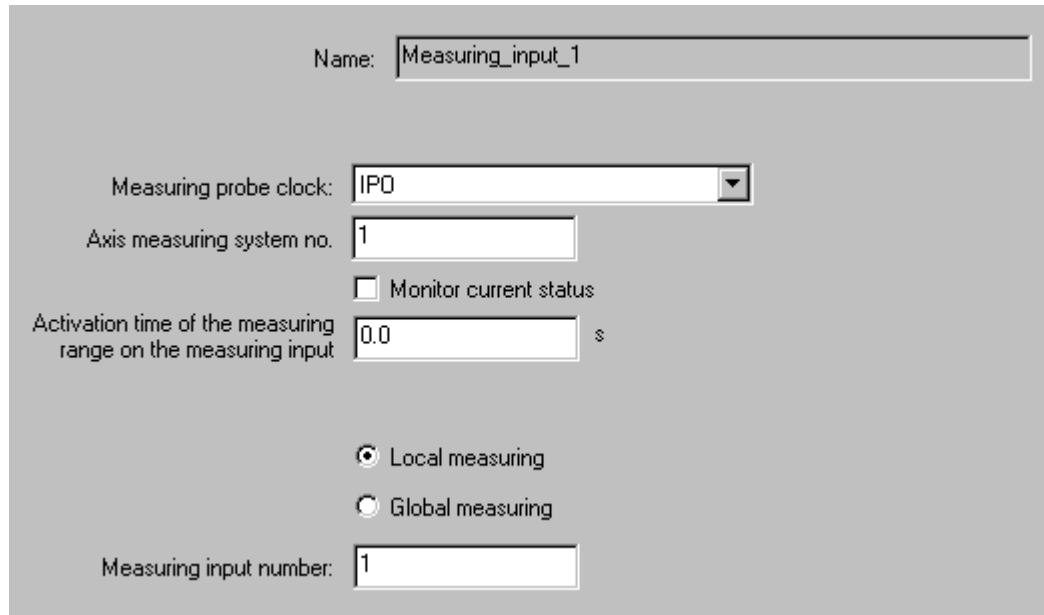
步骤	说明
2.	在“插入凸轮测量输入”(Insert cam measuring input) 对话框中输入工艺对象的名称。 将另外提供用于输入作者、版本号和注释的域。 选中“自动打开编辑器”(Open editor automatically) 复选框，以便自动打开测量输入组态对话框： 
3.	单击“确定”(OK) 确认输入。 结果： 工艺对象随即插入。

其它步骤

在测量输入 > 组态 (Measuring Input> Configuration) 对话框中组态测量输入。

4.11.3 组态 — 测量输入

在“测量输入”(Measuring input) > “组态”(Configuration) 对话框中设置“测量输入”工艺对象的以下属性：



Name: Measuring_input_1

Measuring probe clock: IPO

Axis measuring system no. 1

☐ Monitor current status

Activation time of the measuring range on the measuring input 0.0 s

☒ Local measuring
☐ Global measuring

Measuring input number: 1

测量输入时钟 (Measuring input clock):

使用下拉列表框选择用来插补和处理测量结果的系统周期。

轴测量系统编号 (Axis measuring system no.):

在此输入框中输入所用编码器系统的编号。默认情况下使用“编码器系统 1”。可以将一个编码器系统分配给多个测量输入。

监视当前状态 (Monitor current status):

如果激活了该复选框，则将抑制测量输入的短脉冲（比位置控制周期短）。

如果在工艺功能“MC_MeasuringInput”中选择 *Mode = 1*（通过上升沿测量），则只有在测量输入的输入信号状态 0 至少持续了一个位置控制周期之后，才会激活测量输入。

测量输入的测量范围的激活时间 (Activation time of the measuring range at the measuring input):

在此输入用于激活和取消激活活动范围的激活时间（以秒为单位）。例如，该时间可用于在 DP(DRIVE) 和驱动器的激活期间补偿运行时间。激活的精度取决于位置控制周期时钟。

本地测量 (Local measuring):

如果编码器系统和测量输入的输入位于同一驱动器或系统中，则选择“本地测量”(Local measuring) 选项。

- **测量输入号 (Measuring input number):** （在使用“全局测量”(Global measuring) 时该输入框可见）
在此输入驱动器或系统上的测量输入号。

全局测量 (Global measuring):

如果测量输入的输入位于端子模块 TM15/TM17 上，则选择“全局测量”(Global measuring) 选项。

- **硬件地址 (HW address):** （在使用“全局测量”(Global measuring) 时该输入框可见）
在此输入端子模块 TM15/TM17 的硬件地址。
- **位号 (Bit number):** （在使用“全局测量”(Global measuring) 时下拉列表框可见）
在此选择所用测量输入的输入位号。

说明

测量输入必须位于所用的驱动组件或 TM15/TM17 高性能型模块上。其它数字输入不能用作测量输入。测量输入只能连接包含编码器输入的驱动组件。例如，当编码器输入连接至 SIMODRIVE 后，如果 IM 174/ADI4 也连接至 SIMODRIVE，则您只能将测量输入连接至 IM 174/ADI4。

4.12 组态外部编码器

4.12.1 组态外部编码器

“外部编码器”工艺对象支持的编码器有：

- 接线至 ET 200M 或 ET 200S 的模拟输入模块的模拟绝对编码器（传感器模拟）
- 接线至模拟驱动器接口 IM 174/ADI4 的增量编码器（长方形 TTL）
- 接线至模拟驱动器接口 IM 174/ADI4 的绝对编码器 (SSI)
- 接线至 DP 驱动器的编码器输入的增量编码器或绝对编码器
- 绝对编码器 SIMODRIVE 传感器等时线（消息帧 81）

在用户程序中应用的任何外部编码器的硬件必须在 HW Config 中进行组态，并且外部编码器必须作为“外部编码器”工艺对象插入 S7T Config 中。有关在 HW Config 中组态编码器系统的信息，请参考驱动器或 SIMODRIVE 传感器的说明。

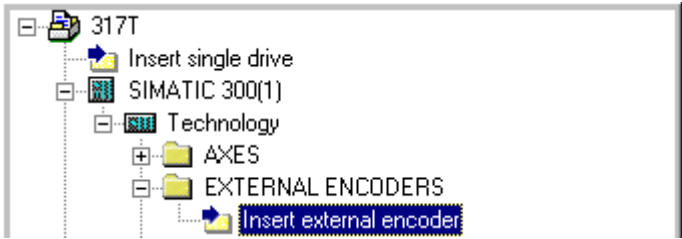
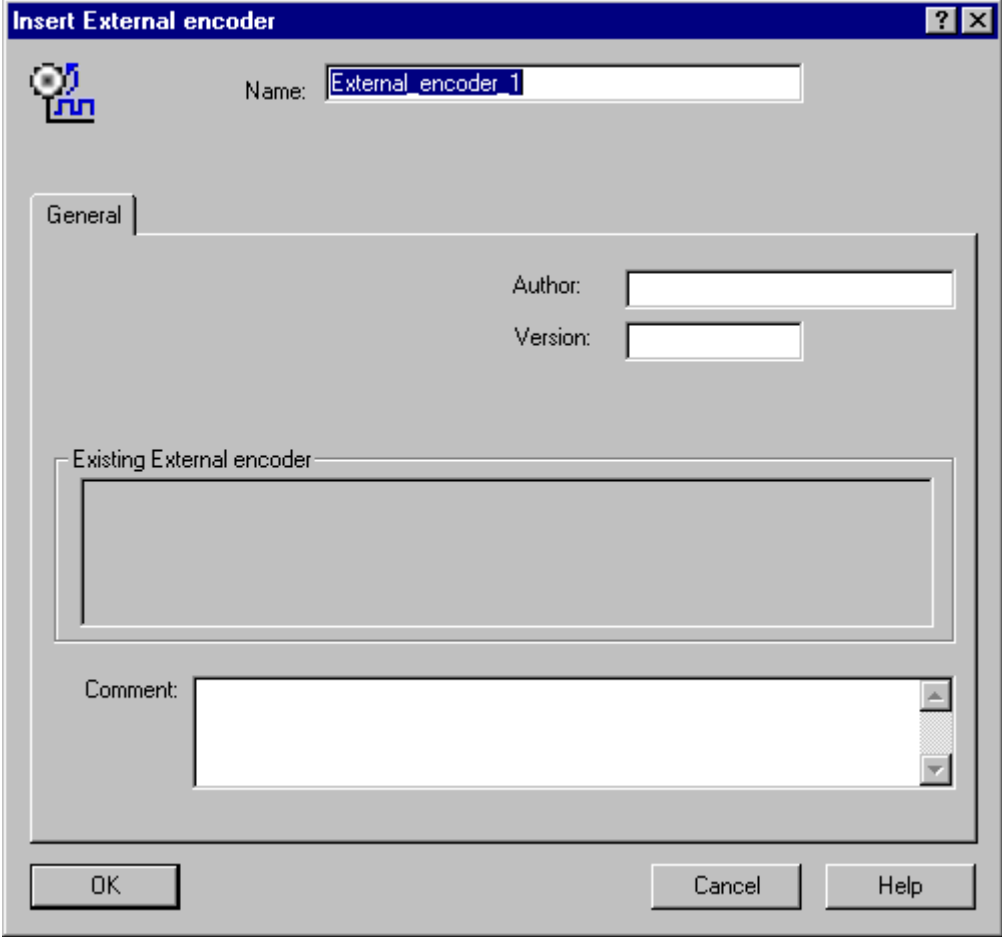
可以使用“MC_ExternalEncoder”工艺功能在用户程序中对测量输入进行编程。此功能可用于启用和禁用外部编码器和回原点设置。通过在 *Axis* 输入中设置工艺 DB 号来定义在 S7T Config 中创建的外部编码器的参考。

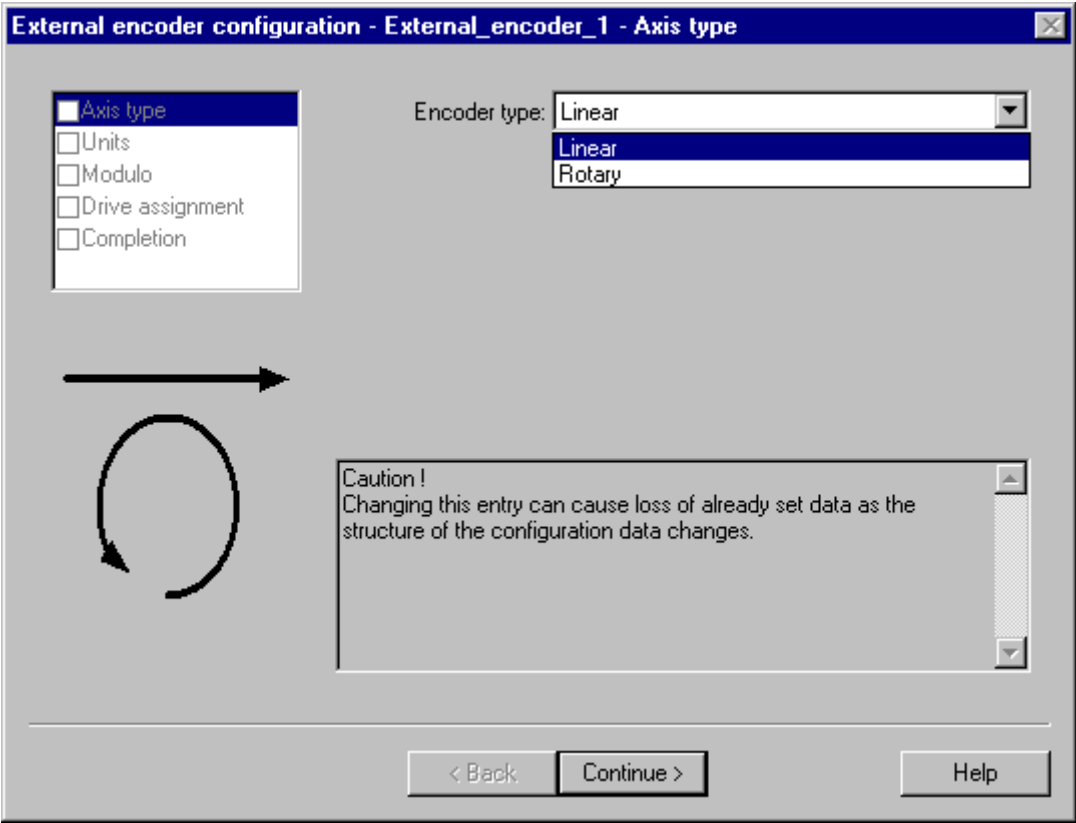
4.12.2 插入外部编码器

要求

- T-CPU 在 HW Config 中进行的组态
- 在 HW Config 中，已组态具有空闲编码器系统的驱动组件或 PROFIBUS 编码器 SIMODRIVE 传感器。PROFIBUS 编码器 SIMODRIVE 传感器必须在“同步模式”下操作。有关详细信息，请参考产品信息或编码器文档。
- 已在 HW Config 中编译并保存硬件组态。

如何将外部编码器插入 S7T Config 中

步骤	说明
1.	打开 S7T Config 的浏览器，然后双击“插入外部编码器”(Insert external encoder)。 
2.	在下一个对话框中，键入工艺对象名称，以及其他可选的作者名字、版本号和注释。  单击“确定”(OK) 确认输入。

步骤	说明
3.	“外部编码器组态”向导随即打开。 选择编码器类型“线性”(Linear) 或“旋转”(Rotary):  单击“继续”(Continue)确认输入。

步骤

说明

4.

“外部编码器组态 — 单位”(External encoder configuration - Units)对话框随即打开。
选择单位，例如“度”(degrees)。

External encoder configuration - External_encoder_1 - Units

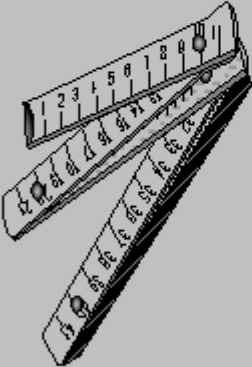
☒Axis type

☐Units

☐Modulo

☐Drive assignment

☐Completion



Units:

Physical quantity	Unit
Position	mm
Increments/position	1000/unit
Velocity	mm/s
Acceleration	mm/s ²
Jerk	mm/s ³
Ratio	%
Time	s
Angle	°
Angular velocity	°/s
Angular acceleration	°/s ²

Caution !

If you change the system of units, the configuration and system variables will be converted (rounding errors are possible) but the data in the programs will not be considered.

< Back

Continue >

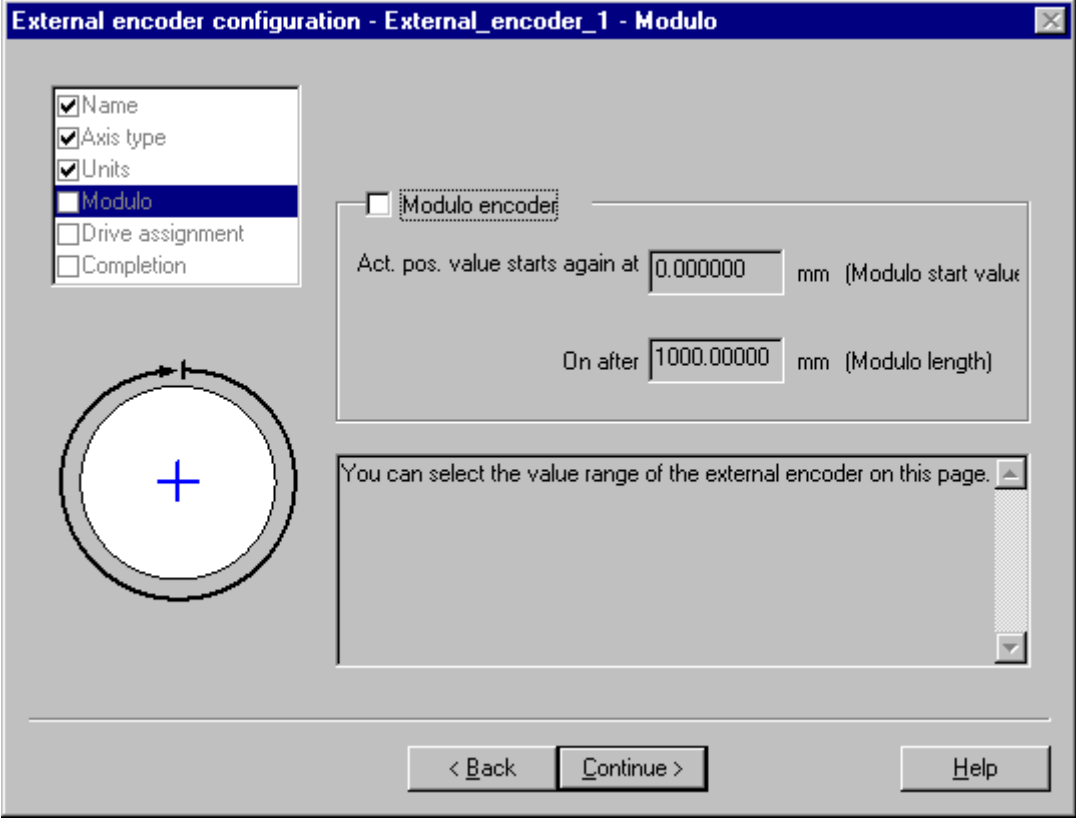
Help

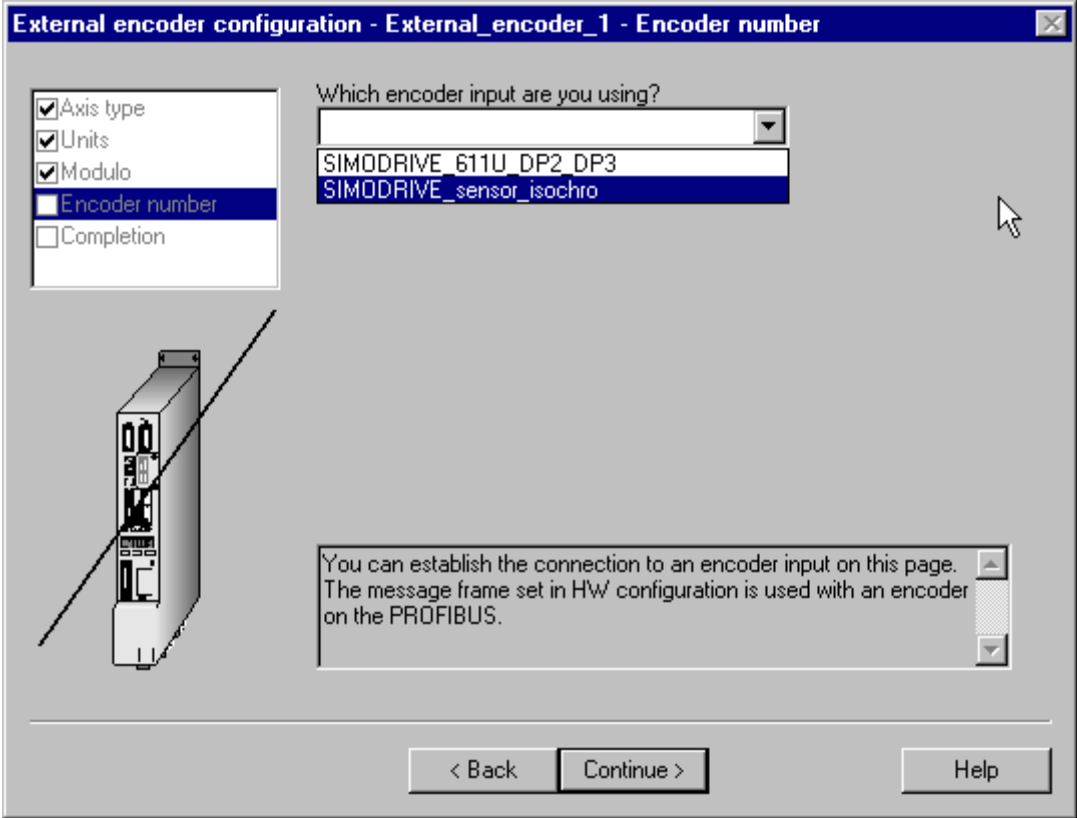
单击“继续”(Continue)确认输入。

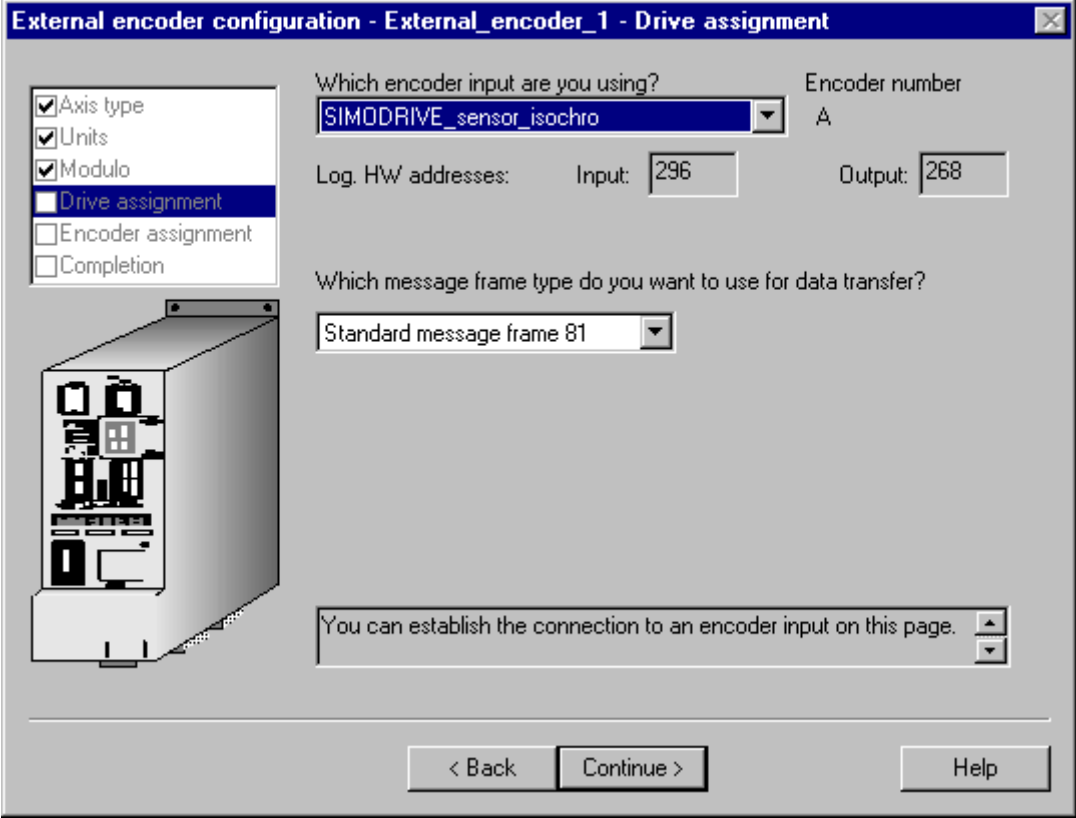
378

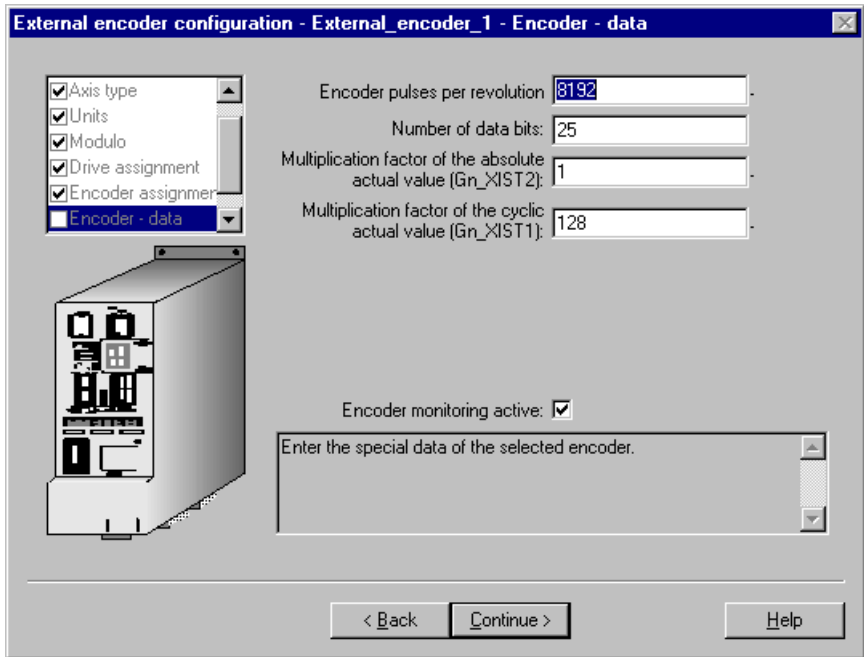
S7-Technology

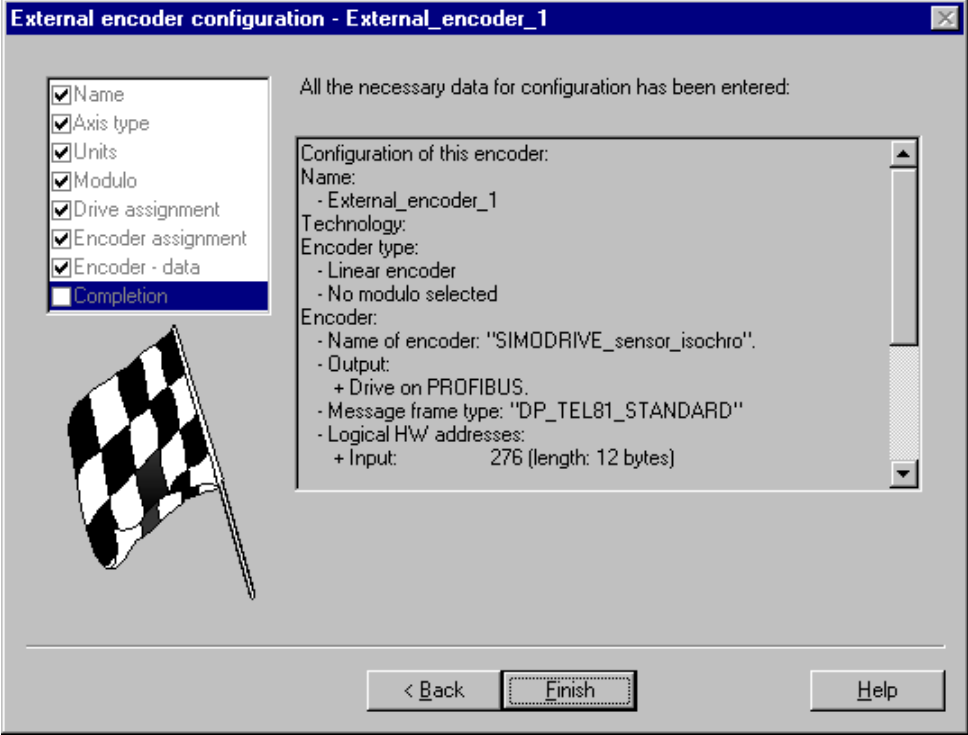
功能手册, 03/2008, A5E01078448-06

步骤	说明
5.	“外部编码器组态 — 模”(External encoder configuration - Modulo)对话框随即打开。  如果想将外部编码器用作模数编码器，即其位置值参考由模数起始值和模数长度定义的行程范围，则激活相应的复选框。输入模数起始值和模数长度的值。 地址信息将从 HW Config 中读取。

步骤	说明
6.	如果有多个编码器系统可用，则“外部编码器组态 — 编码器号”(External encoder configuration - Encoder number) 对话框将打开。 如果在 HW Config 中仅组态了一个驱动器或一个编码器系统，则步骤 7 中显示的对话框将立即打开。  从下拉列表中，选择编码器系统，或编码器系统连接的、要用作外部编码器的驱动组件。

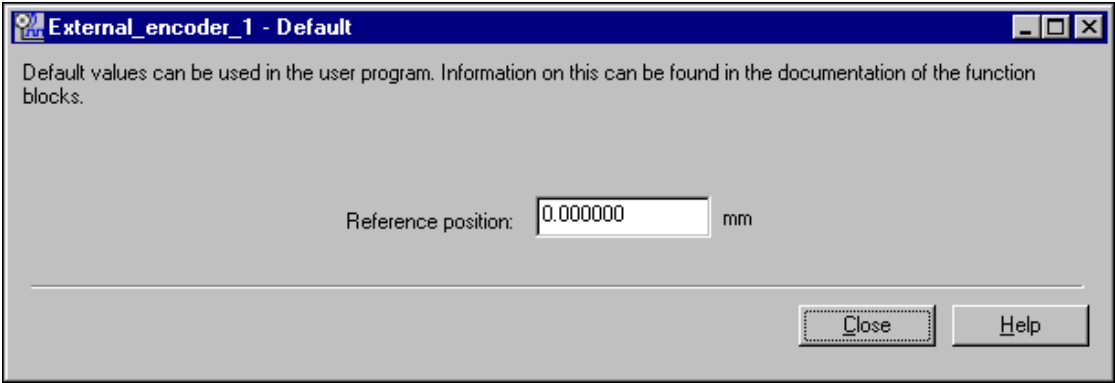
步骤	说明
7.	选择编码器之后，对话框会输出其它信息和一个下拉列表以供您选择消息帧类型。  消息帧类型： 为驱动器的编码器系统选择的消息帧类型必须与 HW Config 中的设置一致（请参阅“选择消息帧类型”）。 SIMODRIVE 传感器设置 当使用 PROFIBUS 编码器 SIMODRIVE 传感器时，请始终选择“标准报文 81”(Standard telegram 81)作为消息帧类型。 IM 174/ADI4 设置 始终为 IM 174/ADI4 选择类型“标准消息帧 3”(Standard message frame 3)。 “编码器号”(Encoder number) 参数显示在 IM 174/ADI4 的选项对话框中的下拉列表框中。从此下拉列表框中选择在 IM 174/ADI4 中为此编码器编写的编码器号。 编码器类型/模式和测量系统： 从“编码器类型”(Encoder type)、“编码器模式”(Encoder mode)和“测量系统”(Measuring system) 下拉列表框中，选择用于组态驱动组件或编码器的设置。有关详细信息，请参考 SIMODRIVE 611 通用型、MASTERDRIVES MC、IM 174/ADI4 和 SIMODRIVE 传感器的补充说明，或者参考设备手册。

步骤	说明																	
	SIMODRIVE 传感器上的设置示例： - 编码器类型：“绝对编码器周期性绝对”(Absolute encoder cyclic absolute) - 编码器模式：“SSI” - 测量系统：“旋转编码器系统”(Rotary encoder system) 单击“继续”(Continue)确认输入。																	
8.	“外部编码器组态 — 编码器数据”(External encoder configuration - Encoder data)对话框随即打开。该对话框的内容取决于所选的编码器类型。例如，下图显示了 SIMODRIVE 传感器编码器的对话框。  IM 174/ADI4 设置 在您在 HW Config 中组态的 IM 174/ADI4 中输入编码器数据。 SIMODRIVE、MASTERDRIVES 等设备上编码器系统的设置 输入在驱动器组态中设置的编码器数据。 SIMODRIVE 传感器设置 为 SIMODRIVE 传感器输入以下值： <table><tr><th colspan="2"></th><th>单匝编码器</th><th>多匝编码器</th></tr><tr><td rowspan="4">参数</td><td>编码器系列</td><td>8192</td><td>8192</td></tr><tr><td>绝对实际值的倍增因子</td><td>1</td><td>1</td></tr><tr><td>周期性实际值的倍增因子</td><td>524288</td><td>128</td></tr><tr><td>数据位数</td><td>13</td><td>25</td></tr></table> 单击“继续”(Continue)确认输入。			单匝编码器	多匝编码器	参数	编码器系列	8192	8192	绝对实际值的倍增因子	1	1	周期性实际值的倍增因子	524288	128	数据位数	13	25
		单匝编码器	多匝编码器															
参数	编码器系列	8192	8192															
	绝对实际值的倍增因子	1	1															
	周期性实际值的倍增因子	524288	128															
	数据位数	13	25															

步骤	说明
9.	下一个对话框显示了组态的汇总内容，如下所示。  单击“应用”(Apply)关闭向导。 结果： 工艺对象随即插入。

4.12.3 缺省值 — 外部编码器

打开外部编码器 (External encoder) > 默认值 (Default) 对话框可以输入外部编码器的参考位置 (Reference position)。使用参考位置可以定义“外部编码器”工艺对象在坐标系中的位置。在 4 和 5 模式下使用“MC_ExternalEncoder”工艺功能，该位置设置可用于同步。



4.12.4 外部编码器 — 与增量编码器同步

T-CPU 支持外部增量编码器的各种同步模式。您可以在 S7T Config 的“专家列表”(Expert List) 中设置外部编码器的参考位置。

同步编码器期间，参考点坐标的值会分配到当前编码器位置：

- 到达编码器零标记后（默认设置）
- 将要到达外部零标记时（在专家列表中进行设置）

外部编码器的回原点模式可以在专家列表中的

TypeOfAxis.Encoder_1.IncHomingEncoder.passiveHomingMode 参数中进行编程。

选项：

- *CAM_AND_ZM_PASSIVE(1)*
带有参考凸轮和下一个编码器零标记的被动回原点模式
- *ZM_PASSIVE(2)*
带有下一个编码器零标记的被动回原点模式
- *CAM_PASSIVE(3)*
带有外部零标记的被动回原点模式
- *DEFAULT_PASSIVE(4)*
使用 *ZM_PASSIVE(2)*（带有编码器零标记）还是 *CAM_PASSIVE(3)*（没有编码器标记）回原点模式，具体取决于编码器类型。

另外，在 *TypeOfAxis.Encoder_1.IncHomingEncoder.passiveApproachDirection* 组态参数中设置所需的参考点逼近方向。

可用的选项：

- *APPROACH_NEGATIVE_PASSIVE(0)*
回原点（沿负方向运动逼近编码器零标记）。
- *APPROACH_POSITIVE_PASSIVE(1)*
回原点（沿正方向运动逼近编码器零标记）。
- *EDGE_POS_SIDE_NEG_PASSIVE(2)*
回原点（沿正方向运动逼近外部零标记）。
- *EDGE_POS_SIDE_POS_PASSIVE(3)*
回原点（沿负方向运动逼近外部零标记）。
- *EDGE_NEG_SIDE_NEG_PASSIVE(4)*
沿正方向运动超过外部零标记后开始回原点。
- *EDGE_NEG_SIDE_POS_PASSIVE(5)*
沿负方向运动超过外部零标记后开始回原点。
- *ACTUAL_DIRECTION_PASSIVE(6)*
在外部零标记或编码器零标记的下一个沿上回原点

所选逼近方向取决于回原点模式。下表显示了逼近方向与回原点模式之间的关系：

逼近方向	回原点模式			
	<i>CAM_AND_ZM_PASSIVE(1)</i>	<i>ZM_PASSIVE(2)</i>	<i>CAM_PASSIVE(3)</i>	<i>DEFAULT_PASSIVE(4)</i>
<i>APPROACH_NEGATIVE_PASSIVE(0)</i>	X	X	F	Gx
<i>APPROACH_POSITIVE_PASSIVE(1)</i>	X	X	F	Gx
<i>EDGE_POS_SIDE_NEG_PASSIVE(2)</i>	F	F	X	Gy
<i>EDGE_POS_SIDE_POS_PASSIVE(3)</i>	F	F	X	Gy
<i>EDGE_NEG_SIDE_NEG_PASSIVE(4)</i>	F	F	X	Gy
<i>EDGE_NEG_SIDE_POS_PASSIVE(5)</i>	F	F	X	Gy
<i>ACTUAL_DIRECTION_PASSIVE(6)</i>	X	X	N	Gx
X — 有效组态 F — 无效组态 N — 不支持的功能 Gx — 带有零标记的编码器（TTL，正弦/余弦） Gy — 带有零标记的编码器（Endat）				

说明

请注意，某些驱动组件不支持上面列出的参考点模式。有关详细信息，请参考相应文档。

根据组态数据中的编码器模式设置，回原点模式 *DEFAULT_PASSIVE(4)* 支持选择逼近方向。如果在组态中定义的是带有零标记的编码器，则只能选择逼近方向 *0*、*1* 和 *6*。如果定义的是没有零标记的编码器，则只能选择逼近方向 *2*、*3*、*4* 和 *5*。任何其它组合都将导致组态错误。

如果选择的是 *CAM_AND_ZM_PASSIVE(1)* 回原点模式，则参考凸轮可以直接接线至 T-CPU 的硬件输入（例如地址 *66*）或 DP(DRIVE) 中分布式 I/O 的输入。在以下组态数据中设置地址：

- 位地址

TypeOfAxis.Encoder_1.IncHomingEncoder.passiveBitNumberBero

- 字节地址

TypeOfAxis.Encoder_1.IncHomingEncoder.passiveLogAdressBero

如果选择的是 *CAM_PASSIVE(3)* 回原点模式，则外部零标记必须接线至相应轴的相关驱动组件。

4.12.5 设置停止信号

外部编码器的停止信号可以在 S7T Config 的“专家列表”(Expert List)中的组态数据中设置:

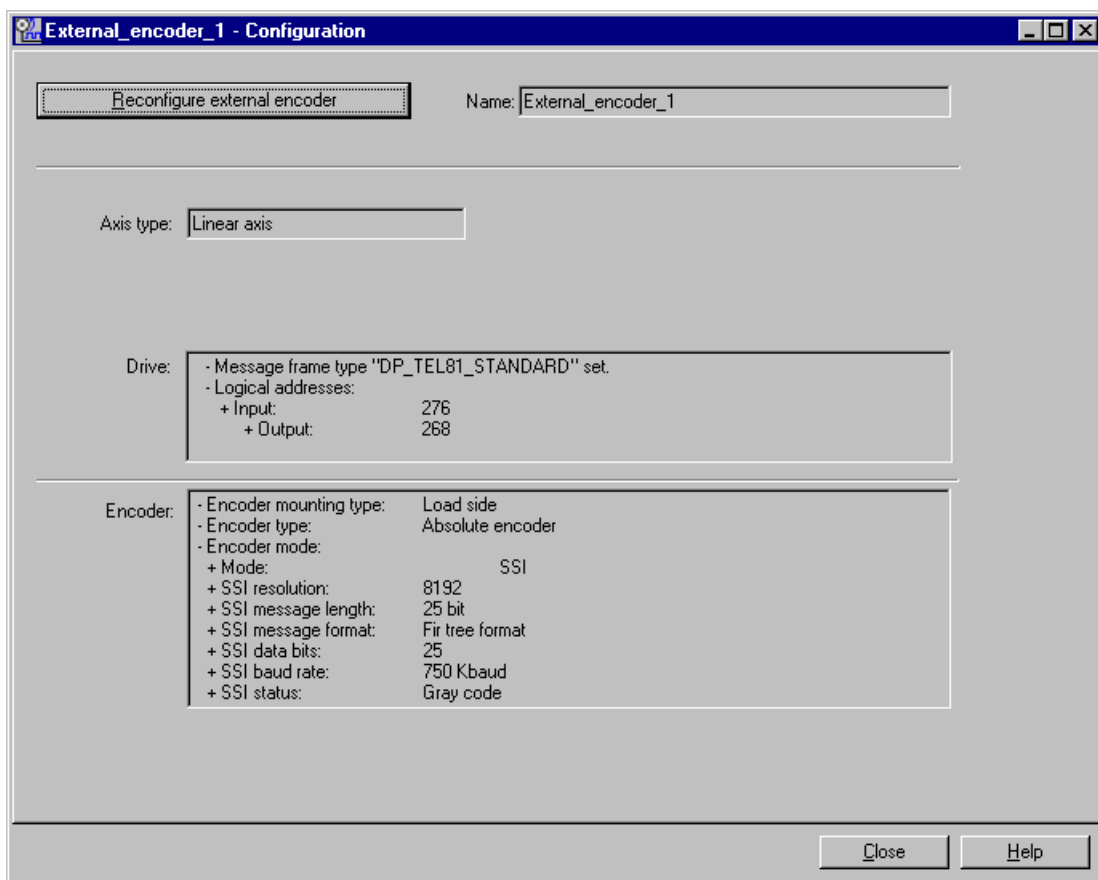
- “速度阈值停止信号”在组态参数中定义
TypeOfAxis.StandStillSignal.maxVeloStandStill
- “信号输出延迟”在组态参数中定义
TypeOfAxis.StandStillSignal.delayTimeToActivate

4.12.6 组态

4.12.6.1 组态 — 外部编码器

外部编码器 > 组态 (External encoder > Configuration) 对话框显示了驱动器设置和编码器设置。

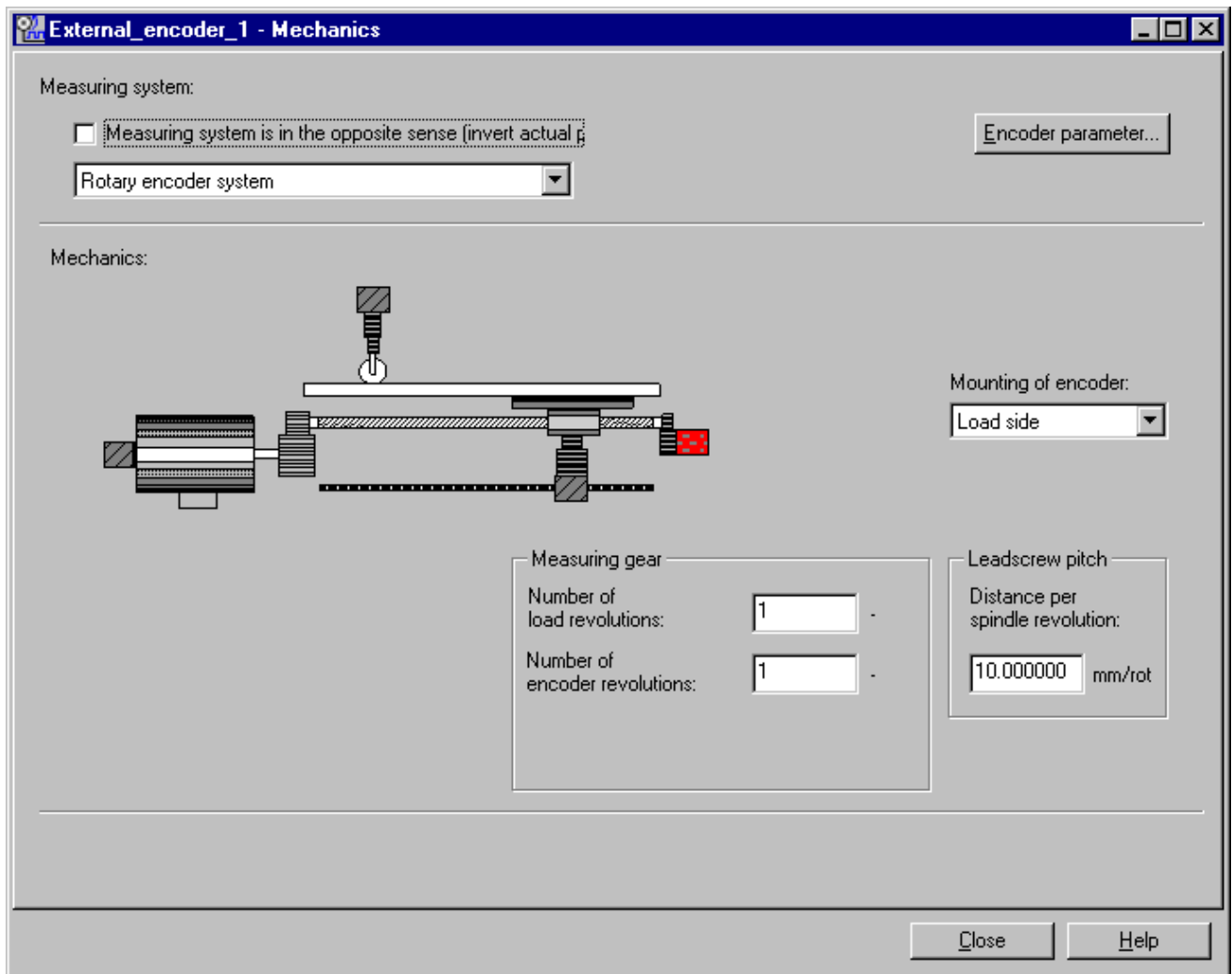
单击“重新组态外部编码器”(Reconfigure external encoder)，以打开编码器向导并编辑设置。



4.12.7 机械

4.12.7.1 机械 — 外部编码器

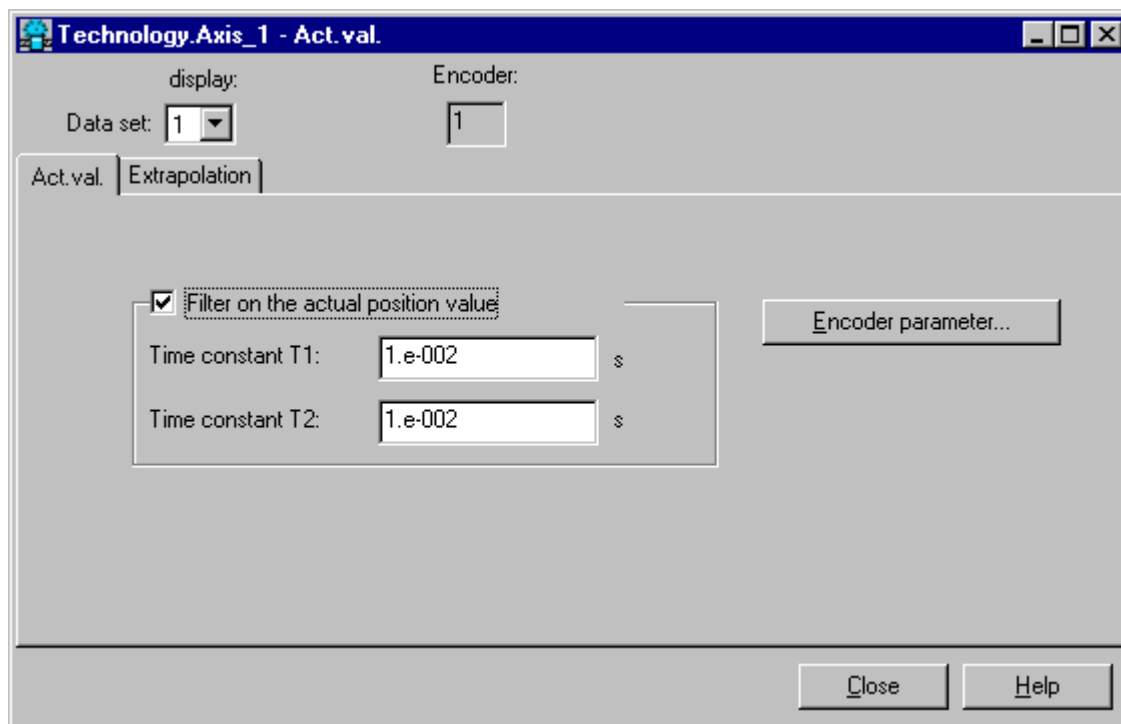
在**外部编码器 (External encoder) > 机械 (Mechanics)** 对话框中定义外部编码器的机械参数。



4.12.8 实际值

4.12.8.1 实际值 (Actual value) —“实际值”(Actual value) 选项卡

在“轴/外部编码器”(Axis/External encoder) > “实际值”(Actual value) 对话框中的“实际值”(Actual value) 选项卡上激活实际值过滤，并设置相应的时间常量。



过滤实际位置值 (Filter on the actual position value)

如果想要激活实际位置值过滤，则激活该复选框。

时间常量 T1 (Time constant T1)

此处设置实际值系统中 PT2 位置过滤器的时间常量 T1。

时间常量 T2 (Time constant T2)

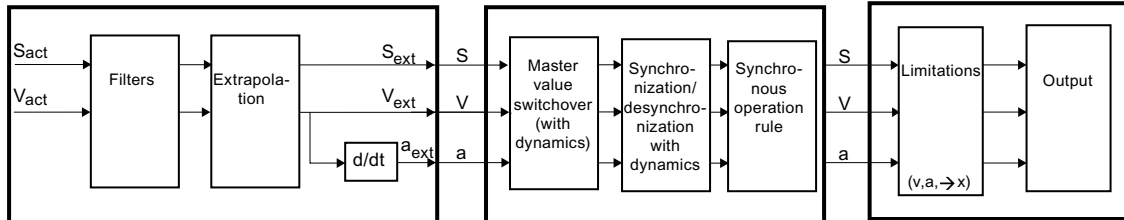
此处设置实际值系统中 PT2 位置过滤器的时间常量 T2。

“编码器参数”(Encoder parameter) 按钮

打开一个对话框，其中显示编码器数据。

4.12.8.2 实际值 (Actual value) —“推断”(Extrapolation) 选项卡

Master value / actual value coupling
(axis or external encoder with actual value extrapolation)

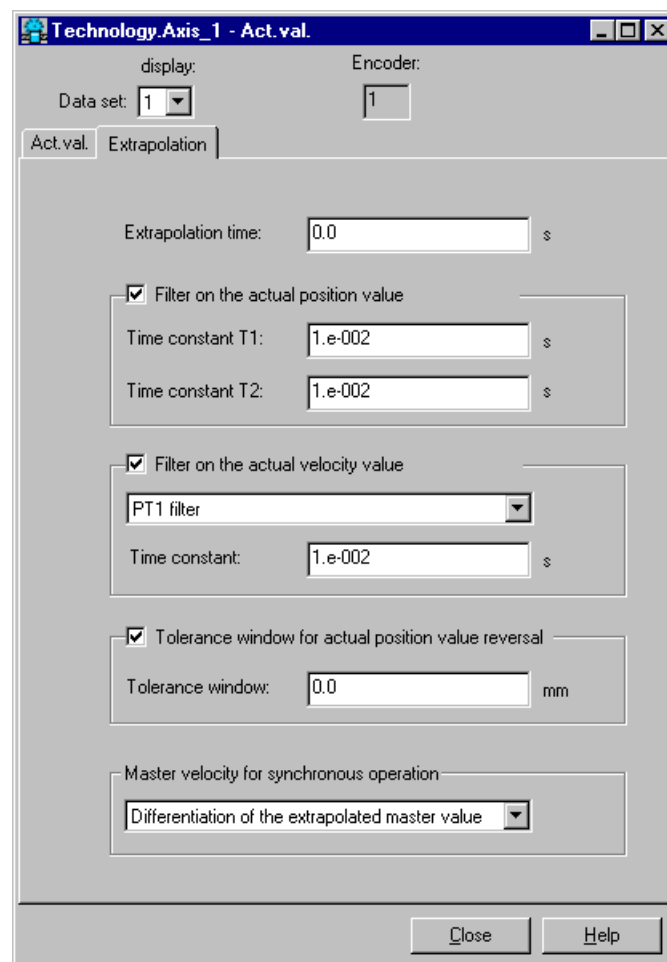


如果控制内有同步操作互连，则同步操作会考虑主值位置的位置、速度和加速度。

如果实际编码器值用作主值，则推断同步操作的测量实际值以补偿死区时间是很用的。

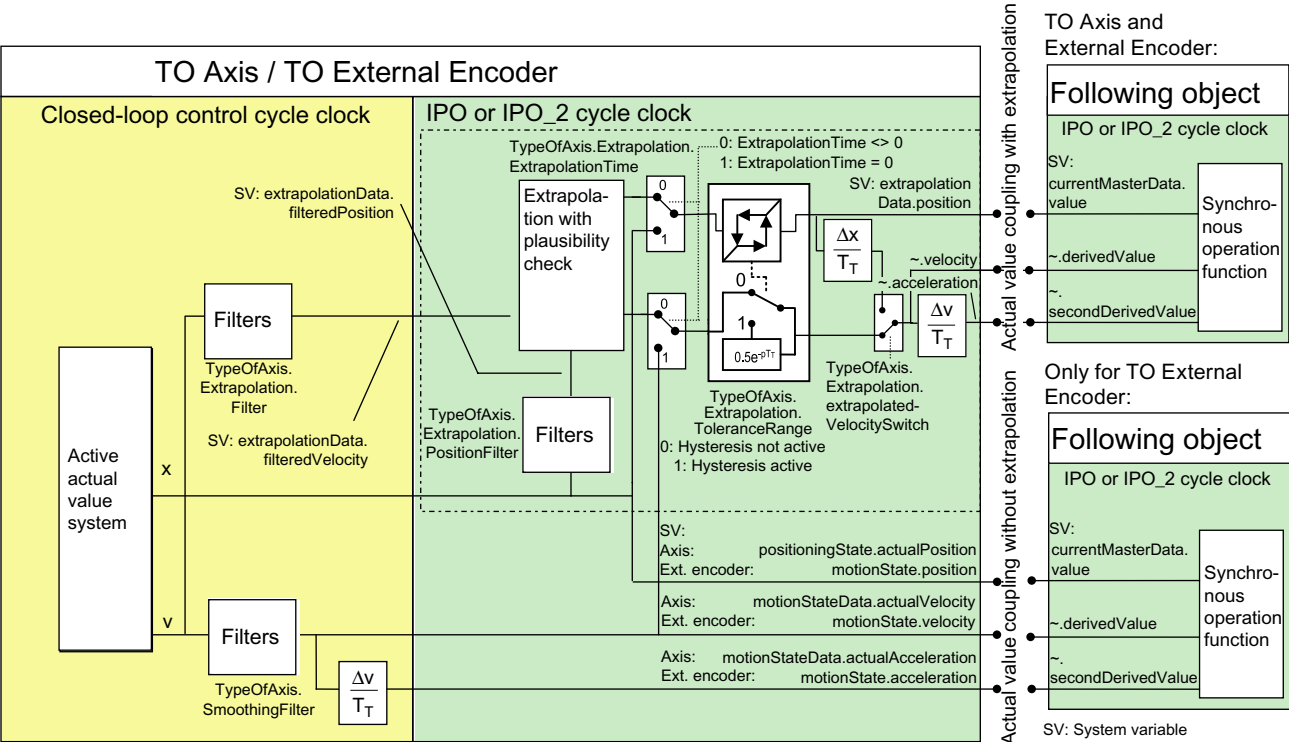
死区时间是在测量实际值时在系统内产生，例如，由于总线通信和系统处理时间产生。

在 S7T Config 中，在引导轴或外部编码器上，在“轴/外部编码器”(Axis/External encoder) >“实际值”(Actual value) 对话框的“推断”(Extrapolation) 选项卡上设置推断。



也可以通过工艺参数从用户程序读取此对话框中的参数，或将此对话框中的参数写入用户程序。

实际值结合推断（轴和外部编码器）



推断时间 (Extrapolation time)

(参数 1110; 组态数据 *TypeOfAxis.Extrapolation.ExtrapolationTime*)

此处设置推断的时间。 如果输入的值为 0.0, 则无推断。

过滤实际位置值 (Filter on the actual position value)

(参数 1130 组态数据 *TypeOfAxis.Extrapolation.ExtrapolationPositionFilter.enable*)

如果想要推断实际位置值, 则激活该复选框。

时间常量 T1 (Time constant T1)

(参数 1131 组态数据 *TypeOfAxis.Extrapolation.ExtrapolationPositionFilter.T1*)

此处设置实际值系统中用于推断的 PT2 过滤器的时间常量 T1。

时间常量 T2 (Time constant T2)

(参数 1132 组态数据 *TypeOfAxis.Extrapolation.ExtrapolationPositionFilter.T2*)

此处设置实际值系统中 PT2 位置过滤器的时间常量 T2。

该过滤器作用于推断的实际位置。推断的速度取自应用平滑滤波器 (*TypeOfAxis.smoothingFilter*) 之前轴或外部编码器的实际值。

过滤实际速度值 (Filter on the actual velocity value)

(参数 1112 组态数据 *TypeOfAxis.Extrapolation.Filter.enable*)

如果想要推断实际速度值，则激活该复选框。

在此处从下拉列表中选择推断速度的过滤器。

(参数 1111 组态数据 *TypeOfAxis.Extrapolation.Filter.Mode*)

时间常量 (Time constant)

(参数 1113 组态数据 *TypeOfAxis.Extrapolation.Filter.timeConstant*)

此处输入过滤器的时间常量。

根据过滤的速度值或平均速度值推断位置。通过“时间常量”计算平均值。

实际位置值倒置公差窗口 (Tolerance window for actual position value inversion)

(参数 1114 组态数据 *TypeOfAxis.Extrapolation.ToleranceRange.enable*)

在此处可以激活用于实际位置值倒置的公差窗口

公差窗口 (Tolerance window)

(参数 1115 组态数据 *TypeOfAxis.Extrapolation.ToleranceRange.Value*)

在此处输入公差窗口的大小。

如果主值与同步操作无法跟随的高频噪声信号叠加，则会造成超出动态响应限制或同步期间主值暂时改变方向。

在这种情况下，可定义公差窗口，以防止在跟随轴上超出动态响应限制，或防止同步期间方向改变。

同步操作的主速度 (Master velocity for synchronous operation)

(参数 1116 组态数据 *TypeOfAxis.Extrapolation.extrapolatedVelocitySwitch*)

在下拉列表中，选择该速度是否应用于推断，或是否对推断的主位置值求微分。

检查推断值和过滤值 (Checking the extrapolated and filtered values)

可在以下系统变量中检查推断值和过滤值：

- *extrapolationdata.position*
- *extrapolationdata.velocity*
- *extrapolationdata.filteredposition*
- *extrapolationdata.filteredvelocity*
- *extrapolationdata.acceleration*

支持具有 NIST 评估的编码器 (Support of encoders with NIST evaluation)

使用具有 NIST 评估的编码器，速度由编码器确定，所得的速度可被编码器接受。在这种情况下，不需要由集成工艺计算速度和速率。有两种传送方法可用：

- 在 PROFIdrive 消息帧中传送

通过组态数据元素 *TypeOfAxis.NumberOfEncoders.Encoder_n.EncoderValueType = POSITION_AND_PROFIDRIVE_NIST_* 设置

- 在 I/O 区域中传送

通过以下组态数据元素设置：

TypeOfAxis.NumberOfEncoders.Encoder_n.EncoderValueType = POSITION_AND_DIRECT_NIST。在这种情况下，4000H 对应于 100%。地址在组态数据元素 *TypeOfAxis.NumberOfEncoders.Encoder_n.nistConfig.logAdress* 中设置，参考值在组态数据元素 *TypeOfAxis.NumberOfEncoders.Encoder_n.nistConfig.referenceValue* 中设置。

4.12.9 回原点

4.12.9.1 回原点 —“被动回原点”(Passive homing) 标签

可以在 S7T Config 的轴 (Axis) > 回原点 (Homing) 对话框的“被动回原点”(Passive homing) 选项卡中组态被动回原点。与主动回原点相比，这二者的不同之处在于，被动回原点所需的回原点运动不是由回原点命令启动的。

“被动回原点”(Passive homing) 有三种可用的回原点模式：

- 参考凸轮和编码器零标记
- 仅外部零标记
- 仅编码器零标记

启用回原点功能：

MC_Home	<i>Mode = 2</i> <i>Position = x</i>	为当前位置分配了参考点输入参数 <i>Position</i> 的值。

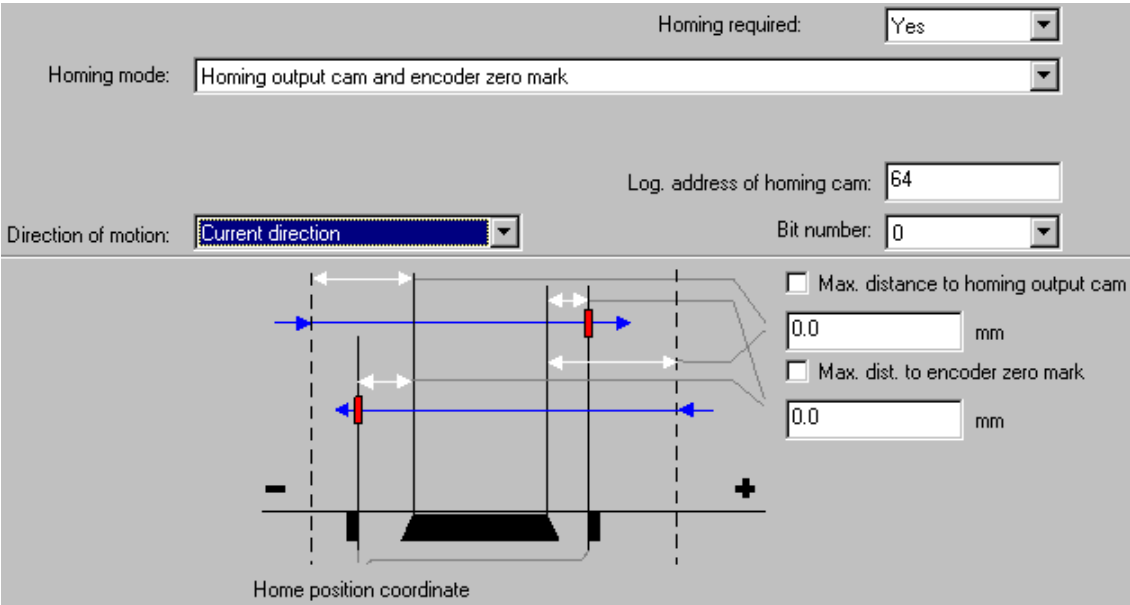
成功回原点后，轴工艺 DB 将返回状态 *Statusword.HomingDone = TRUE*。

说明

请注意，并非所有驱动组件都支持所有的回原点模式或测量信号计算功能。有关详细信息，请参考所用驱动组件的文档。

由工艺 DB 上相应的错误消息来表示在被动回原点操作期间对编码器零标记或参考凸轮监视的触发。在这种情况下，当前轴运动终止。

使用“参考凸轮和编码器零标记”(Reference cam and encoder zero mark) 的被动回原点模式



轴经过参考凸轮后，下一个编码器零标记将触发轴回原点。将参考点的轴位置设置为通过“MC_Home”工艺功能的输入参数 *Position* 定义的值。

各个参数的含义：

参数	值	说明
需要回原点	是	轴必须回原点才能执行绝对运动命令。
	否	轴不必回原点就可以执行绝对运动命令。
回原点模式	-	在此例中：“参考凸轮和编码器零标记”(Reference cam and encoder zero mark)
运动方向	正方向	轴仅通过正逼近方向回原点到编码器零标记。
	负方向	轴仅通过负逼近方向回原点到编码器零标记。
	当前方向	轴在到达下一个编码器零标记时回原点。
参考凸轮的逻辑地址	[字节地址]	参考凸轮的逻辑字节地址 可以将参考凸轮连接至 T-CPU 的集成输入或连接至 DP(DRIVE) 上的 IO。
位号	[位号]	用于参考凸轮的信号的位地址
到回原点输出凸轮的最大距离	取消激活	没有监视到参考凸轮检测的距离
	激活	监视回原点功能的起点和参考凸轮检测之间的距离。 如果该距离的差值超出了组态的值，则相应的工艺 DB 将返回错误 801D。回原点功能取消。

参数	值	说明
到编码器零标记的最大距离	取消激活	没有监视到编码器零标记所需要移动的距离
	激活	监视轴在参考凸轮和编码器零标记之间移动的距离 如果该距离的差值超出了组态的值，则相应的工艺 DB 将返回错误 801D。回原点功能取消。

使用“仅外部零标记”(External zero mark only) 的被动回原点模式

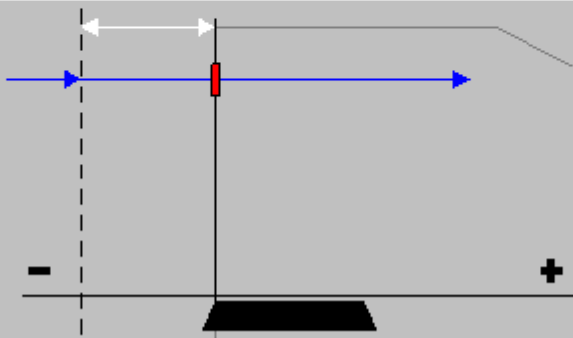
Homing required:

Homing mode:

On the external zero mark side:

Direction of motion:

☐ Max. distance to ext. zero mark
0.0 mm



Home position coordinate

在检测到外部零标记时，开始轴回原点。将轴设置为“MC_Home”工艺功能的 *Position* 输入参数的值。

各个参数的含义：

参数	值	说明
需要回原点	是	轴必须回原点才能执行绝对运动命令。
	否	轴不必回原点就可以执行绝对运动命令。
回原点模式	-	在此例中：“仅外部零标记”(External zero mark only)
运动方向	正方向	轴仅通过正逼近方向回原点到外部零标记。
	负方向	轴仅通过负逼近方向回原点到外部零标记。
	当前方向	轴在到达下一个外部零标记时回原点。

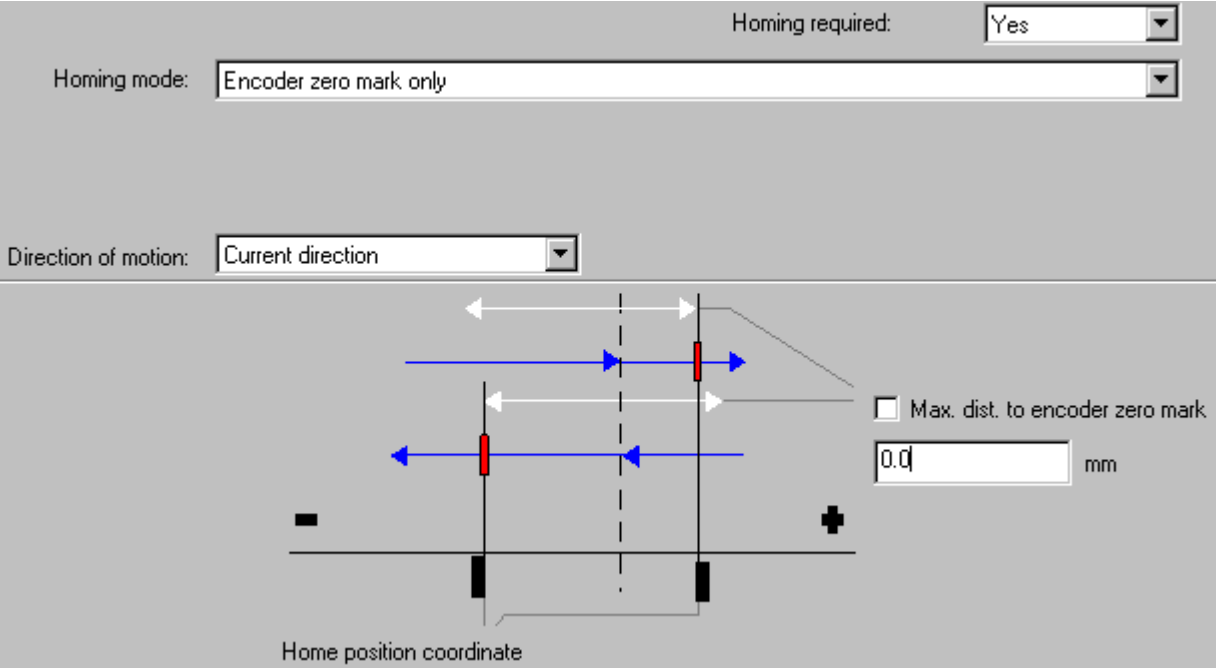
4.12 组态外部编码器

参数	值	说明
在外部零标记一侧	左	移动方向： 正方向 在上升沿上触发轴回原点。 移动方向： 负方向 在下降沿上触发轴回原点。
	右	移动方向： 正方向 在下降沿上触发轴回原点。 移动方向： 负方向 在上升沿上触发轴回原点。
到外部零标记的最大距离	取消激活	没有监视到外部零标记所需要移动的距离
	激活	该功能将监视轴从其回原点功能的起点到检测出外部零标记所移动的距离 如果该距离的偏差超出了组态的距离，则相应的轴工艺 DB 将指示错误 801D 。回原点功能取消。

说明

- 对于使用“外部零标记”(external zero mark) 的回原点，将外部零标记作为数字测量输入连接至驱动组件。
- 为了能够根据需要以“外部零标记”(external zero mark) 回原点模式执行参考点逼近，您应始终在“移动方向”(Direction of movement) 和“在外部零标记一侧”(on the side of the external zero mark) 上设置数值，使其与驱动器组态或功能兼容。“在外部零标记一侧”(on the side of the external zero mark) 和“移动方向”(Direction of movement) 参数不用于组态驱动器中的测量功能，仅反映其功能。
- 有关外部零标记检测的组态的信息，请参考相关的产品信息或驱动器手册。

使用“仅编码器零标记”(Encoder zero mark only) 的被动回原点模式



例如，使用“仅零标记”(Zero mark only) 的回原点模式用于装有一个在轴的整个行程范围内只能输出一个零标记信号的编码器的轴。

在检测到编码器零标记时，开始轴回原点。检测到编码器零标记后，将轴的位置值设置为“MC_Home”工艺功能的输入参数 *Position* 的值。

各个参数的含义：

参数	值	说明
需要回原点	是	轴必须回原点才能执行绝对运动命令。
	否	轴不必回原点就可以执行绝对运动命令。
回原点模式	-	在此例中：“仅编码器零标记”(Encoder zero mark only)
运动方向	正方向	轴仅通过正逼近方向回原点到编码器零标记。
	负方向	轴仅通过负逼近方向回原点到编码器零标记。
	当前方向	轴在到达下一个编码器零标记时回原点。

4.12 组态外部编码器

参数	值	说明
到编码器零标记的最大距离	取消激活	没有监视到编码器零标记所需要移动的距离
	激活	该功能将监视轴从其回原点功能的起点到检测到编码器零标记所移动的距离。 如果该距离的偏差超出了组态的距离，则相应的轴工艺 DB 将指示错误 <i>801D</i> 。回原点功能取消。

使用“默认”(Default) 的被动回原点模式

在 S7T Config 中创建新轴时，会预设“默认”(Default) 回原点模式。

如果组态的编码器提供零标记，则使用“仅编码器零标记”(encoder zero mark only) 回原点模式。如果编码器不提供零标记，则使用仅外部零标记回原点模式。

4.13 从另一个站中复制组态数据

下面的步骤说明了在同一项目的第二个站中重复使用硬件和工艺组态数据以及用户程序的方法。

进行以下操作的条件：

- 源站的硬件组态已成功编译并保存
- S7T Config 中工艺的组态数据已在源项目中编译并保存
- SIMATIC Manager 已打开（S7T Config、HW Config 等已关闭）

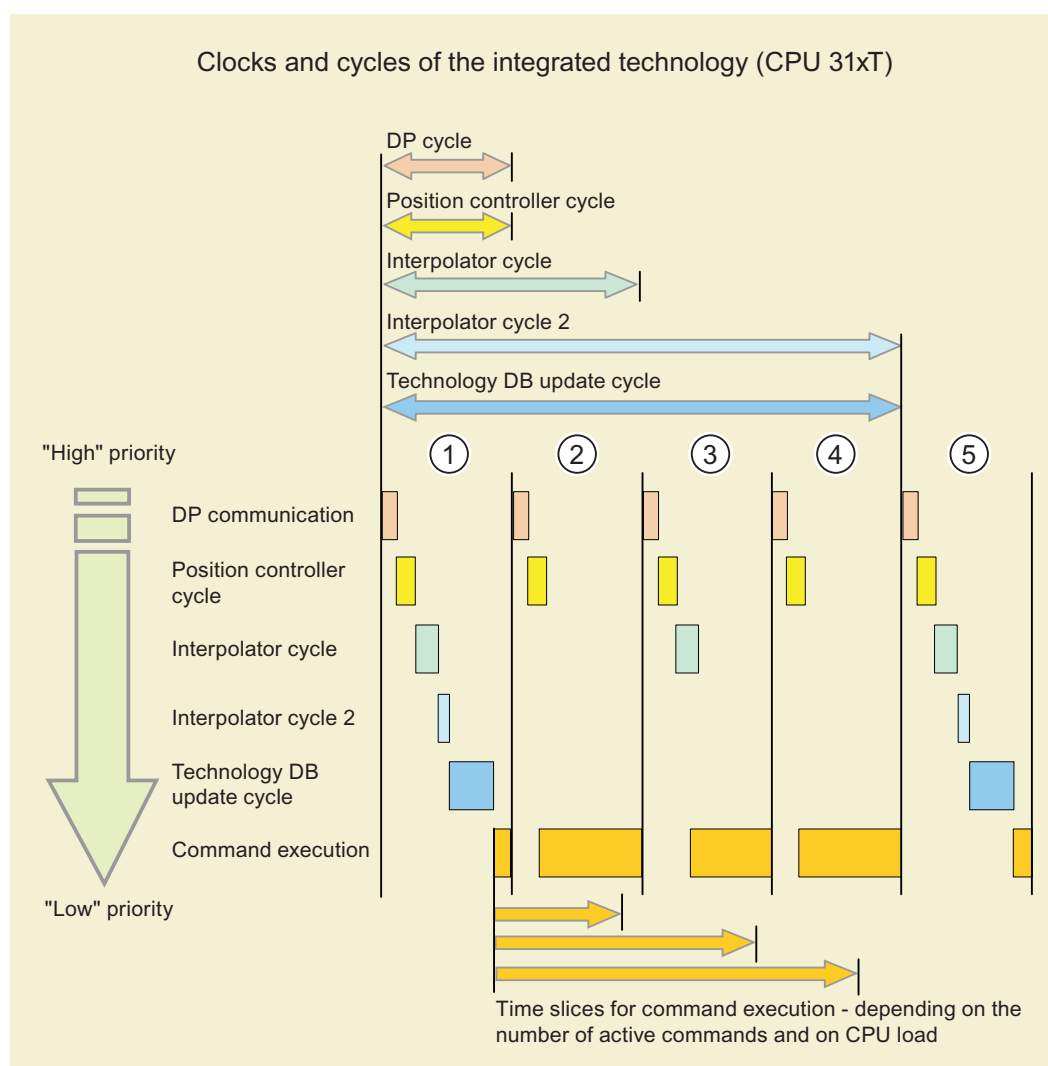
步骤	说明
1.	在 SIMATIC Manager 中选择源站，然后选择 编辑 > 复制 (Edit > Copy) 。
2.	选择目标项目，然后选择 编辑 > 粘贴 (Edit > Paste) 命令。 插入后，一个带有明确名称的新站随即显示在项目中。所有的 STEP 7 组件都被复制到该新站中。
3.	打开新站的硬件组态。根据需要将其添加到 DP(DRIVE) 接口和 MPI/DP 接口的组态中，因为此复制操作不包括逻辑网络链接和网络。因此，您需要创建新的网络对象，然后修改组态的驱动组件的同步操作的设置。
4.	保存并编译硬件组态。关闭 HW Config。
5.	在 SIMATIC Manager 中选择“工艺对象”(Technology objects)，然后通过选择 编辑 > 打开对象 (Edit > Open object) 命令运行“工艺对象管理”。 根据需要修改生成的工艺 DB。

至此，您已创建了新站，其中包含了源站的数据（保存为不同的名称）。

4.13 从另一个站中复制组态数据

5.1 CPU 31xT 的时间图

下图显示了集成工艺的周期，以及该周期对 CPU 31xT 的工艺对象的作业处理的影响。该实践示例显示了多个位置控制周期的顺序。



该图显示了各个周期在各时间片单元内的时间要求。时间片的要求根据组态的工艺对象的数量不同而有所不同。插补器周期的时间片、插补器周期 2 的时间片和工艺 DB 的更新周期的时间片不在任何位置控制周期中进行处理。

DP 通信、位置控制周期、插补器周期、插补器周期 2 和工艺 DB 更新周期按依次降低的优先级执行。

可以利用位置控制周期期间的间隔执行命令。间隔越小，执行命令所需的时间越长。

示例中选定的周期如下所示：

- 位置控制周期 = 1
与 DP 周期和位置控制周期之间的比率 1:1 相对应
- 插补器周期 = 2
与 DP 周期和位置控制周期之间的比率 1:2 相对应
- 插补器周期 2 = 2
与插补器周期和插补器周期 2 之间的比率 1:2 相对应
- 工艺 DB 的更新周期 = 插补器周期的倍数
示例中给出的是系数为 2

所示周期与 DP 周期同步，或与 DP 通信同步。

- **DP 周期**

DP 周期决定了数据与 DP(DRIVE) 上 DP I/O 交换的间隔。数据在 DP 通信时间片内交换。DP(DRIVE) 的恒定总线循环时间必须在 HW Config 中的相应参数中激活。集成工艺连同 DP(DRIVE) 上的同步组件共同构成了同步操作的基础。

- **位置控制周期**

该系统还可以在位置控制周期时间片内计算轴位置控制。当 T-CPU 运行时，该时间片的时间要求仍接近恒定。

位置控制周期时钟决定了位置控制周期期间的间隔。

在 S7T Config 中，位置控制周期时钟的值可以设置为 DP 周期的倍数。建议采用 1:1 的时钟比率，以便提高工艺对象处理方面的性能。

- **插补器周期**

插补器周期的时间片还可以用于计算控制变量。插补器周期时钟决定了两个插补器周期期间的间隔。在一个或多个命令开始时，时间片要求可能会暂时提高。

- 插补器周期 2

插补器周期 2 与插补器周期处理的任务相同，但可用于低优先级的工艺对象。

- 更新工艺 DB

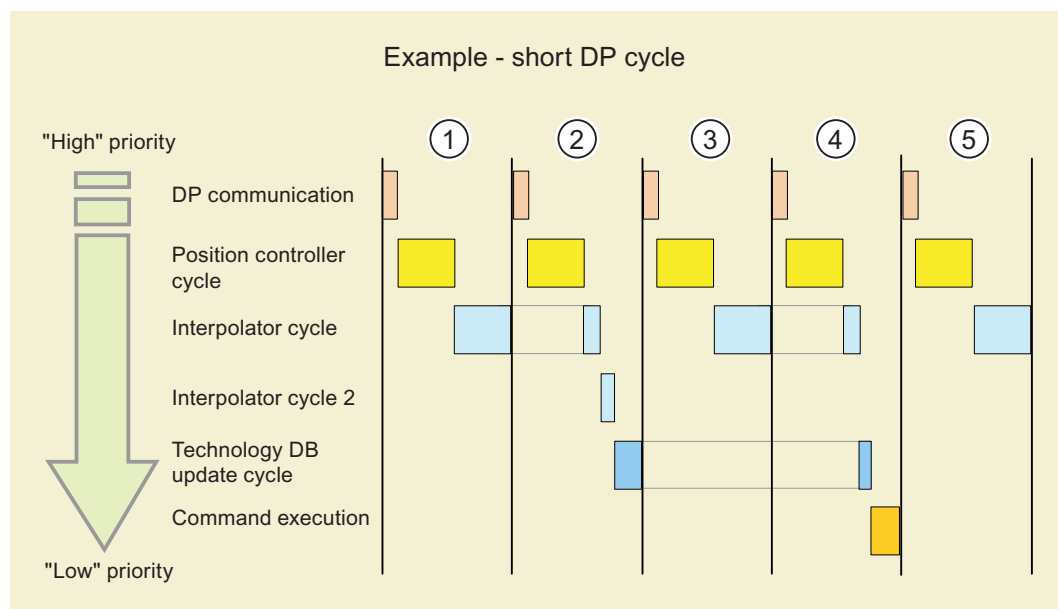
工艺 DB 在此时间片内更新。固件版本为 V3.2.x 或更高版本的集成工艺支持将工艺 DB 更新分布到不同的周期。有关详细信息，请参考“更新工艺数据块”。在固件版本 V3.2.x 之前的版本中，所有工艺 DB 都在同一时间片内更新。

在 S7T Config 中，工艺 DB 的更新周期可以设置为插补器周期的倍数。每次更新工艺 DB 后都会调用工艺同步中断 OB。

命令处理

集成工艺的命令在不确定的周期中进行处理和监视。处理当前命令所需的时间取决于活动命令的数量和 CPU 负载。命令处理的平均时间和最大时间都可以从 MCDevice 工艺 DB 的变量 *CmdLoopDuration* 和 *MaxLoopDuration* 中读取。固件版本为 V3.2.x 或更高版本的集成工艺支持在插补器周期中启动轴命令。后续的命令监视操作再次通过集成工艺的命令处理功能进行处理。

以上示例显示周期 1 仅为命令处理提供一个小时间片。除了 DP 通信和位置控制周期之外，系统还可以处理插补器周期和插补器周期 2 以及工艺 DB 的更新。该方法可以为在后面的周期中进行命令处理提供额外时间。



该示例显示了在 DP 周期长度不足的情况下的时间曲线。上一个示例中显示的时钟比率适用于：

- 周期 1

插补器周期所需的时间超过了周期 1 中可用的时间。插补器周期 2（包括工艺 DB 的更新）不能在此周期中开始。开始时间将推迟至下一个周期。

- 周期 2

DP 通信和位置控制周期会中断插补器周期。中断清除后，插补器周期的时间片将继续。接下来，系统会执行插补器周期时钟 2 并启动工艺 DB 更新。

- 周期 3

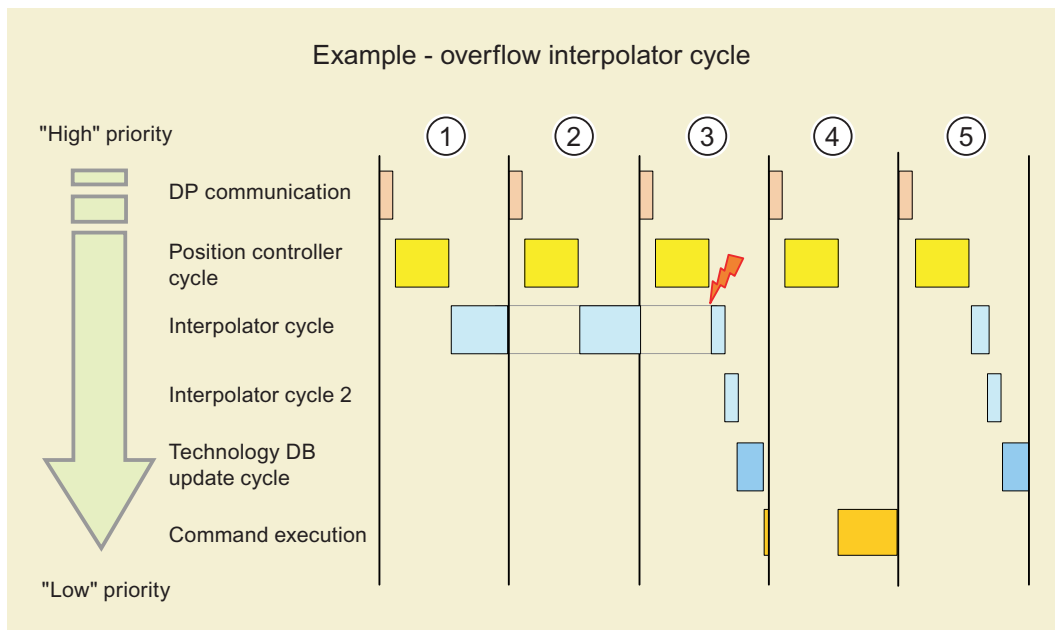
中断工艺 DB 更新，以便执行更高优先级的任务（DP 通信、位置控制周期/插补器周期）。与周期 1 相同，插补器周期无法在此周期内中止。

- 周期 4

插补器周期的时间片将继续，并在位置控制周期结束时终止。接下来，系统将执行插补器周期 2 的时间片并完成工艺 DB 的更新。这只为命令处理留了很短的时间片。

- 周期 5

在周期 5 中，按照在周期 1 到 4 中说明的顺序重新开始。



该命令开始时可能会暂时提高插补器周期的时间需求。该图显示了可能会导致插补器周期“溢出”的状态。应用与上一个示例相同的时钟比率：

- **周期 1**

插补器周期所需的时间超过了周期 1 中可用的时间。插补器周期 2（包括工艺 DB 的更新）不能在此周期中开始。开始时间将推迟至下一个周期。

- **周期 2**

DP 通信和位置控制周期会中断插补器周期。中断清除后，插补器周期的时间片将继续。

- **周期 3**

再次中断插补器周期的时间片，以便执行 DP 通信和位置控制周期。系统仍忙于执行当前的插补器周期，并调用新的插补器周期。该操作会导致插补器周期“溢出”。T-CPU 是停止还是对此溢出进行容错，具体取决于 S7T Config 中的系统时钟设置。在此示例中至少对一个溢出进行容错，然后继续处理，意味着插补器周期将结束。接下来，系统会执行插补器周期 2 并更新工艺 DB。这只为命令处理留了很短的时间片。

- **周期 4**

本应在周期 3 中开始的插补器周期在此周期中跳过，即系统在位置控制周期结束后，执行用于处理命令的时间片。

- **周期 5**

同样，插补器周期的时间要求很少。插补器周期、插补器周期 2 和工艺 DB 更新可以在周期 5 中处理。

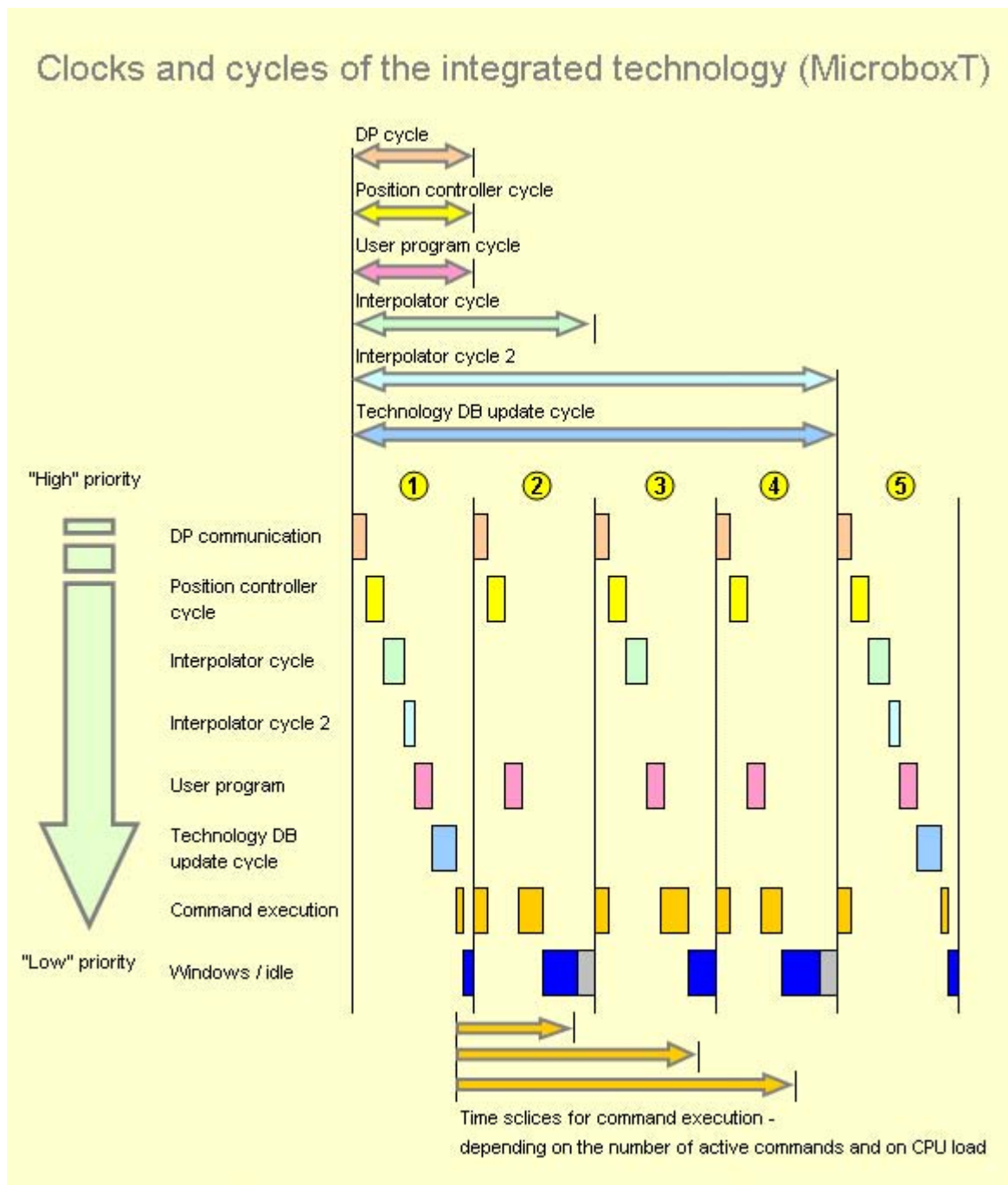
说明

周期时间设置中的窄边可能会导致插补器周期 2 溢出。

在 S7T Config 的“系统时钟周期”(System clock cycles) 对话框中，可以为插补器周期和插补器周期 2 设置容错溢出的数量（“目标系统”[Target system] > “设置系统时钟周期”[Set System Clock Cycles] 命令）。如果超出了溢出事件的容错数量，则 T-CPU 将转为 STOP 模式。

5.2 WinLC T (MICROBOX T) 的时序

下图显示了集成工艺的周期，以及它们对执行 MICROBOX T 的工艺对象命令的影响。该示例显示了包含多个位置控制周期的顺序。



该图显示了各个周期在各时间片单元内的时间要求。时间片的要求根据组态的工艺对象的数量不同而有所不同。插补器周期的时间片、插补器周期 2 的时间片和工艺 DB 的更新周期的时间片不在任何位置控制周期中进行处理。

DP 通信、位置控制周期、插补器周期、插补器周期 2 和工艺 DB 更新周期按依次降低的优先级执行。在 MICROBOX T 中，DP 通信的时间还可用于集成工艺周期，但位置控制周期除外。

可以利用位置控制周期期间的间隔执行命令。间隔越小，执行命令所需的时间越长。

示例中选定的周期如下所示：

- 位置控制周期 = 1
与 DP 周期和位置控制周期之间的比率 1:1 相对应
- 插补器周期 = 2
与 DP 周期和位置控制周期之间的比率 1:2 相对应
- 插补器周期 2 = 2
与插补器周期和插补器周期 2 之间的比率 1:2 相对应
- 工艺 DB 的更新周期 = 插补器周期的倍数
示例中给出的是插补器周期 2

所示周期与 DP 周期同步，或与 DP 通信同步。

- **DP 周期**

DP 周期决定了数据与 DP(DRIVE) 上 DP I/O 交换的间隔。数据在 DP 通信时间片内交换。DP(DRIVE) 的恒定总线循环时间必须在 HW Config 中的相应参数中激活。集成工艺连同 DP(DRIVE) 上的同步组件共同构成了同步操作的基础。

- **位置控制周期**

该系统还可以在位置控制周期时间片内计算轴位置控制。当 T-CPU 运行时，该时间片的时间要求仍接近恒定。

位置控制周期时钟决定了位置控制周期期间的间隔。

在 S7T Config 中，位置控制周期时钟的值可以设置为 DP 周期的倍数。建议采用 1:1 的时钟比率，以便提高工艺对象处理方面的性能。

- **插补器周期**

插补器周期的时间片还可以用于计算控制变量。插补器周期时钟决定了两个插补器周期期间的间隔。在一个或多个命令开始时，时间片要求可能会暂时提高。

- **插补器周期 2**

插补器周期 2 与插补器周期处理的任务相同，但可用于低优先级的工艺对象。

- **用户程序**

用户程序和 WinLC T (MICROBOX T) 中的所有中断 OB 均在所示时间片内执行。在 WinLC T 面板上可以将时间片的长度设置为 DP 周期时间的百分比。可以设置的值范围为 10 % 到 50 %。

OB 1 周期在此时间片内执行。但是，它并不与 DP 周期同时开始。当用户程序的时间片结束后，当前程序执行将被中断，并在下一个时间片开始时继续。

用户程序的时间片在 DP 周期的每个周期时钟中调用。如果时间片连续三次无法执行，则 WinLC T 状态会变为 STOP。

- **更新工艺 DB**

工艺 DB 在此时间片内更新。固件版本为 V3.2.x 或更高版本的集成工艺支持将工艺 DB 更新分布到不同的周期（另请参见“更新工艺 DB”）。在固件版本 V3.2.x 之前的版本中，所有工艺 DB 均在同一时间片内更新。

在 S7T Config 中，工艺 DB 的更新周期可以设置为插补器周期的倍数。每次更新工艺 DB 后都会调用工艺同步中断 OB。

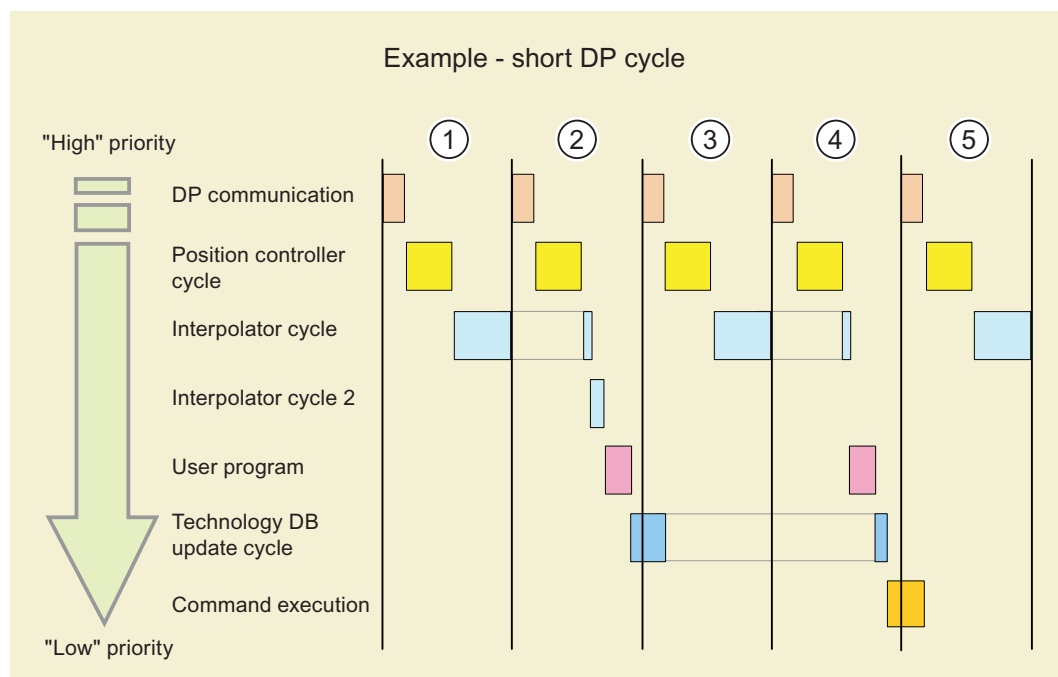
- **命令处理**

集成工艺的命令在不确定的周期中进行处理和监视。处理当前命令所需的时间取决于活动命令的数量和 CPU 负载。命令处理的平均时间和最大时间都可以从 MCDevice 工艺 DB 的变量 *CmdLoopDuration* 和 *MaxLoopDuration* 中读取。固件版本为 V3.2.x 或更高版本的集成工艺支持在插补器周期中启动轴命令。后续的命令监视操作再次通过集成工艺的命令处理功能进行处理。

- **Windows/空闲周期**

所示时间片可用于 **Windows** 应用程序。如果 **Windows** 应用程序不再使用此时间片，则插入空闲时间片，直到下一个 **DP** 通信。

以上示例显示周期 1 仅为命令处理提供一小时间片。除了 **DP** 通信和位置控制周期之外，系统还可以处理插补器周期和插补器周期 2 以及工艺 **DB** 的更新。该方法可以为在后面的周期中进行命令处理提供额外时间。



该示例显示了在 **DP** 周期长度不足的情况下的时间曲线。上一个示例中显示的时钟比率适用于：

- **周期 1**

插补器周期所需的时间超过了周期 1 中可用的时间。插补器周期 2（包括工艺 **DB** 的更新）不能在此周期中开始。开始时间将推迟至下一个周期。

- **周期 2**

DP 通信和位置控制周期会中断插补器周期。中断清除后，插补器周期的时间片将继续。接下来，系统会执行插补器周期时钟 2 并启动工艺 **DB** 更新。

- **周期 3**

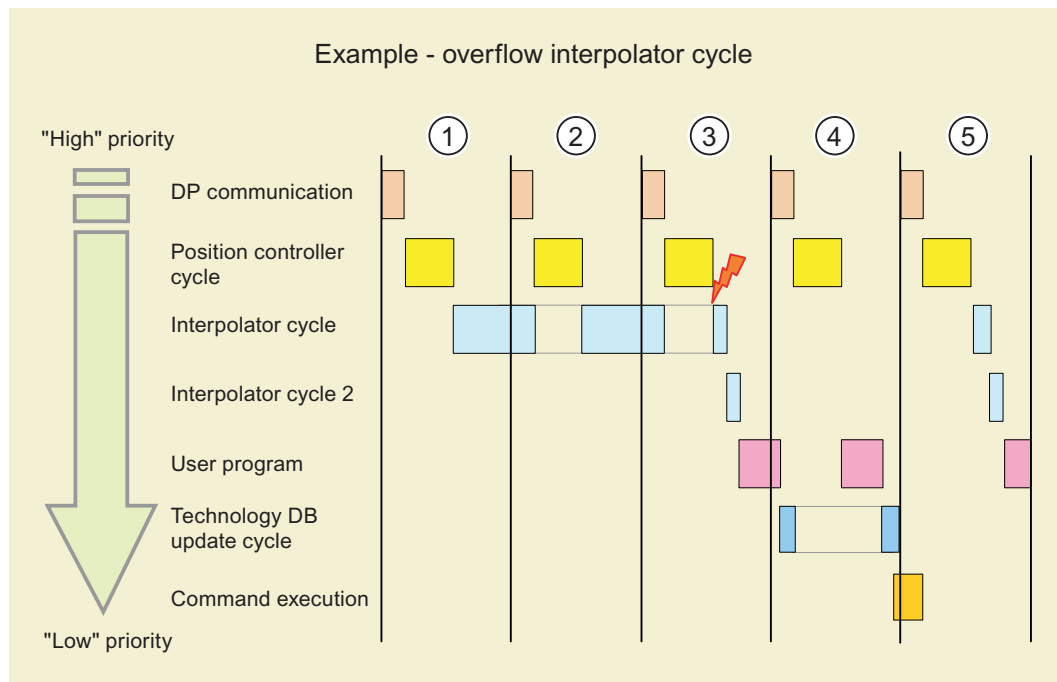
中断工艺 **DB** 更新，以便执行更高优先级的任务（**DP** 通信、位置控制周期/插补器周期）。与周期 1 相同，插补器周期无法在此周期内中止。

- 周期 4

插补器周期的时间片将继续，并在位置控制周期结束时终止。接下来，系统将执行插补器周期 2 的时间片并完成工艺 DB 的更新。这只为命令处理留了很短的时间片。

- 周期 5

在周期 5 中，按照在周期 1 到 4 中说明的顺序重新开始。



该命令开始时可能会暂时提高插补器周期的时间需求。该图显示了可能会导致插补器周期“溢出”的状态。应用与上一个示例相同的时钟比率：

- 周期 1

插补器周期所需的时间超过了周期 1 中可用的时间。插补器周期 2（包括工艺 DB 的更新）不能在此周期中开始。开始时间将推迟至下一个周期。

- 周期 2

DP 通信和位置控制周期会中断插补器周期。中断清除后，插补器周期的时间片将继续。

- 周期 3

再次中断插补器周期的时间片，以便执行 DP 通信和位置控制周期。系统仍忙于执行当前的插补器周期，并调用新的插补器周期。该操作会导致插补器周期“溢出”。T-CPU 是停止还是对此溢出进行容错，具体取决于 S7T Config 中的系统时钟设置。在此示例中至少对一个溢出进行容错，然后继续处理，意味着插补器周期将结束。接下来，系统会执行插补器周期 2 并更新工艺 DB。这只为命令处理留了很短的时间片。

- **周期 4**

本应在周期 3 中开始的插补器周期不适用于此周期，即系统在位置控制周期结束后，执行用于处理命令的时间片。

- **周期 5**

同样，插补器周期的时间要求很少。插补器周期、插补器周期 2 和工艺 DB 更新可以在周期 5 中处理。

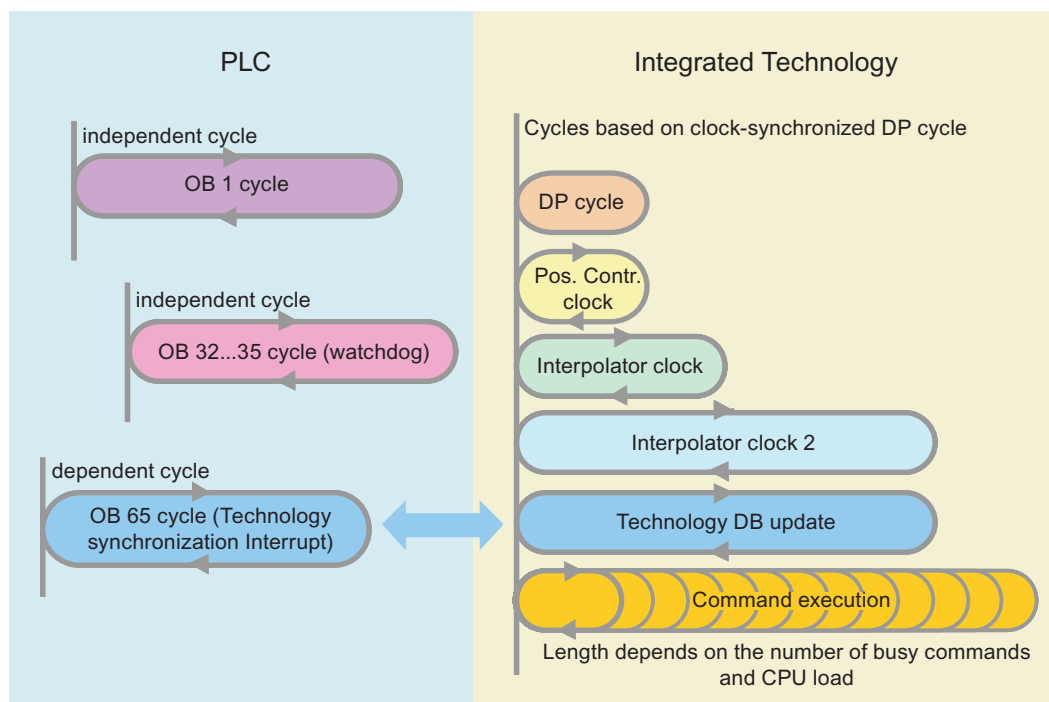
说明

周期时间设置中的窄边可能会导致插补器周期 2 溢出。

在 S7T Config 的“系统时钟周期”(System clock cycles) 对话框中，可以为插补器周期和插补器周期 2 设置容错溢出的数量（“目标系统”[Target system] > “设置系统时钟周期”[Set System Clock Cycles] 命令）。如果超出了溢出事件的容错数量，则 T-CPU 将转为 STOP 模式。

5.3 T-CPU 的周期

下图显示了 T-CPU 的各种周期和时钟间的关系：



控制器周期

- **OB1 周期**

OB1 周期的长度取决于执行的指令数量。OB1 周期的运行与集成工艺的周期无关。

- **OB32 到 OB35 周期**

在 1 到 60 秒的时序内调用定时中断 OB。定时中断的调用与集成工艺的时钟和周期无关。

- **OB65 周期**

在每次更新工艺 DB 后都会调用工艺同步中断 OB。工艺同步中断与集成工艺耦合。

集成工艺的周期

集成工艺的周期与 DP 周期和 DP 通信同步。

- **DP 周期**

在 DP 周期内与 DP(DRIVE) 上的 DP I/O 交换数据。

- **位置控制周期**

该系统还可以在位置控制周期时间片内计算轴位置控制。

- **插补器周期**

插补器周期主要用于计算控制变量。

- **插补器周期 2**

插补器周期 2 与插补器周期处理的任务相同，但可用于低优先级的工艺对象。

- **工艺 DB 更新周期**

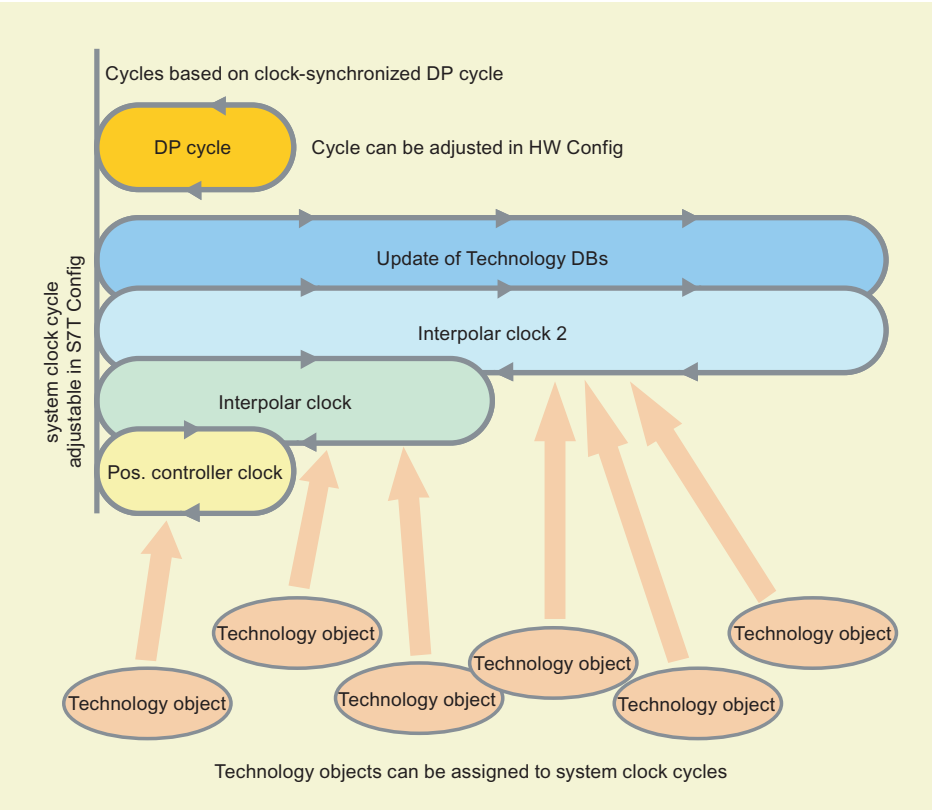
工艺 DB 在此周期内更新。

命令处理

集成工艺的命令在不确定的周期中进行处理和监视。处理当前命令所需的时间取决于活动命令的数量和 CPU 负载。

5.4 分配系统时钟

通过向工艺对象分配系统时钟，可以提高对系统资源的利用。



您可以向工艺对象分配以下时钟：

运动控制任务	高优先级	...	低优先级
工艺对象	位置控制周期	插补器周期	插补器周期 2
速度控制轴	-	标准	X
定位轴	-	标准	X
同步轴	-	标准	X
外部编码器	-	标准	X
输出凸轮	X	X	X
测量输入	X	X	X
"X"= 允许			
"- "= 不允许			

您可以将工艺 DB 更新的基本时钟设置为插补器周期的倍数。

说明

如果为输出凸轮及关联轴选择不同的执行周期，则允许以下组合：

- 轴和输出凸轮在同一周期时钟内（插补器周期时钟或插补器周期时钟 2）
 - 轴在插补器周期时钟或插补器周期时钟 2 内，输出凸轮在位置控制周期内。
 - 轴在插补器周期时钟内，实际值输出凸轮在插补器周期时钟 2 内。
-

在 S7T Config 的组态 (Configuration) 对话框中设置轴的工艺系统时钟、外部编码器、输出凸轮和测量输入。

同步轴和相应的同步对象必须在同一执行周期内操作。对同步轴的处理周期所做的任何更改也必须应用于同步对象：

1. 在 S7T Config 的浏览器中，选择相应同步轴的同步对象。
2. 选择快捷菜单命令 **专家 > 专家列表 (Expert > Expert list)**
3. 您可以在 *Execution.executionlevel* 组态数据元素中更改处理周期时钟。

更改默认设置：

- 如果您检测到命令处理时间过长
执行时间可从 MCDevice 工艺 DB 的 *MaxLoopDuration* 和 *CmdLoopDuration* 变量中获取。
- 如果您在 T-CPU 上检测到过量负载
通过选择“**目标系统 (Target system) > 设备诊断 (Device diagnostics) > 系统负载 (System load)**”菜单命令，可以在在线模式下确定 T-CPU 上的负载。

将具有低优先级任务的“轴”和“外部编码器”工艺对象分配给“插补器周期 2”，将“输出凸轮”和“测量输入”工艺对象分配给“插补器周期”或“插补器周期 2”。将“插补器周期”或“位置控制周期”分配给具有高优先级任务的“输出凸轮”和“测量输入”工艺对象。

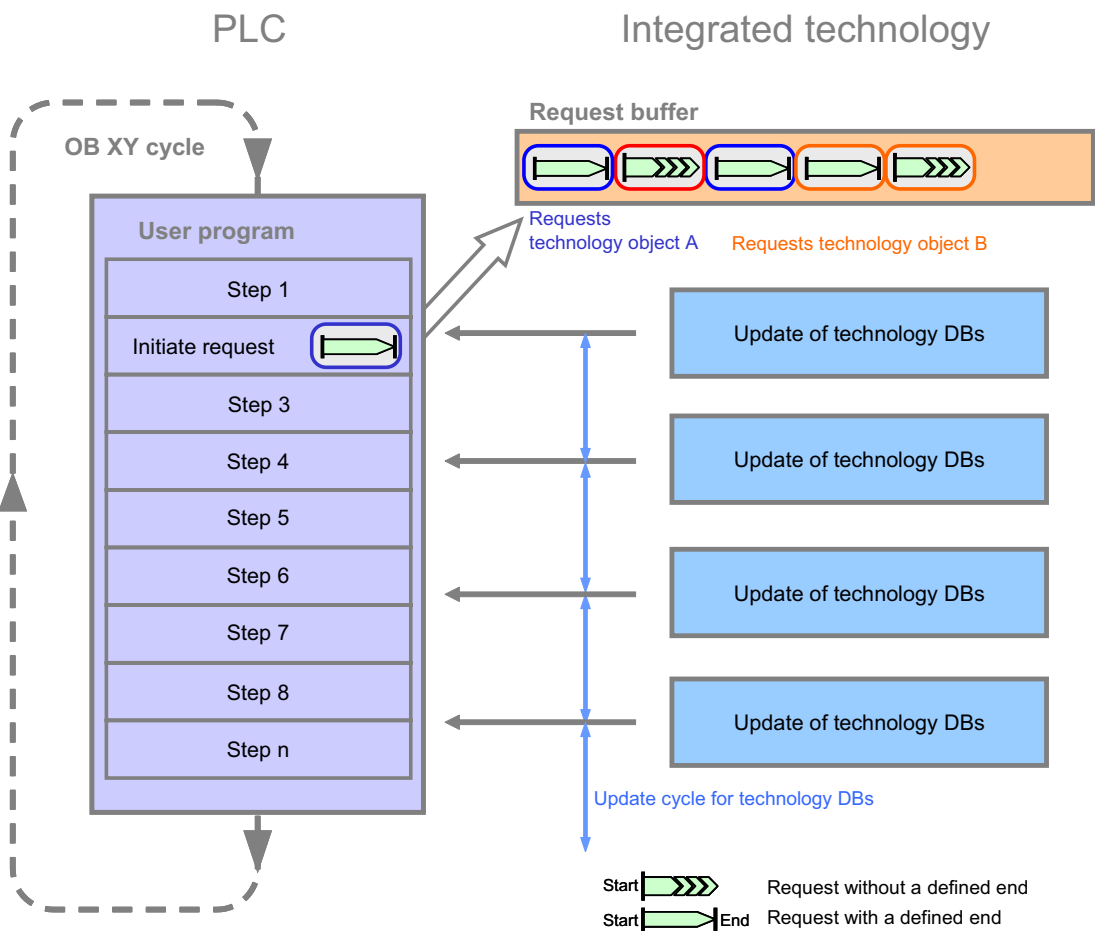
如果超过了命令监视时间，则您可以通过采取以下措施减少命令执行时间：

- 设置较长的 DP 周期或位置控制周期。
- 延长工艺 DB 更新周期或将各个工艺 DB 的更新分布到不同的周期（有关详细信息，请参考“更新工艺 DB”）。
- 将插补器周期 2 分配给低优先级的工艺对象。这会减少插补器周期上的负载，并增加命令执行的时间片。
- 在工艺对象管理中选择专家模式，并为特定工艺对象启动“**IPO 同步**”命令。此设置会强制该命令在插补器周期的时间片内开始，从而减少命令执行时间片上的负载。

5.5 顺序模型和编程模型

顺序模型

下图显示了 PLC 和集成工艺的命令界面。



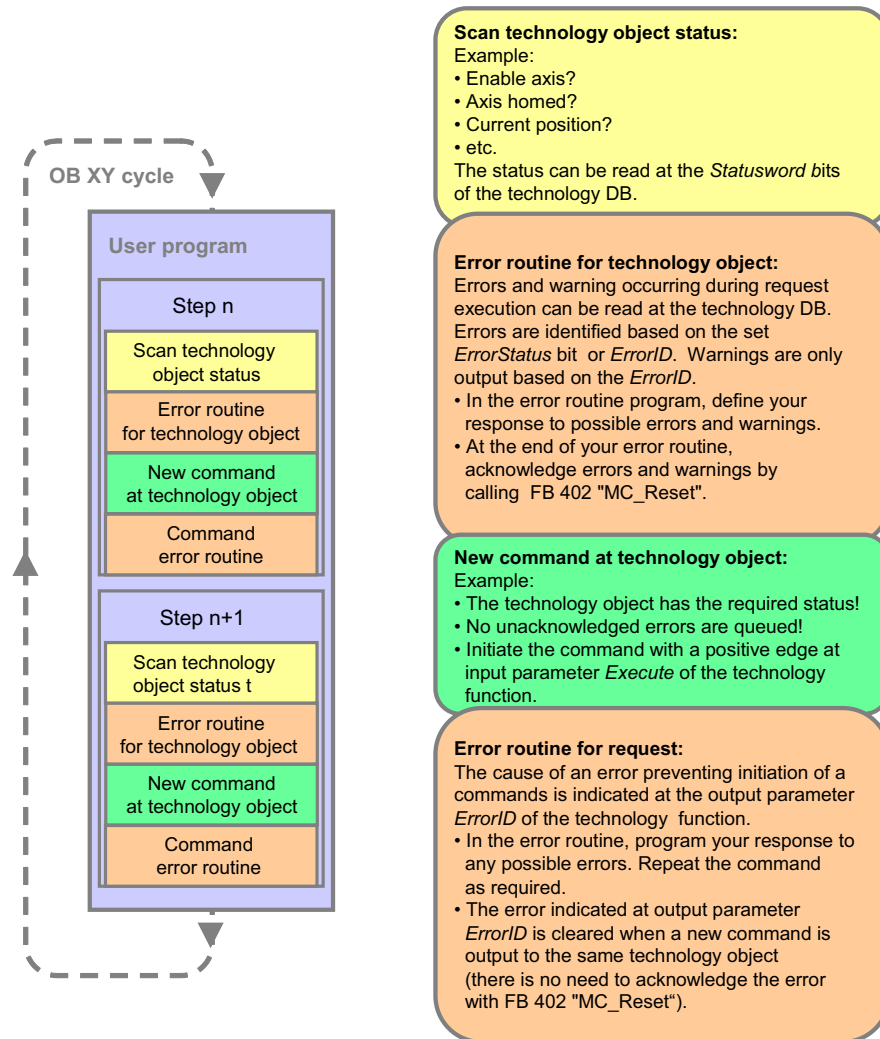
顺序用户程序（OB1 或定时中断 OB）启动了一个命令，并将其输入到集成工艺的命令缓冲区中。“IPO 同步”和“非 IPO 同步”命令分别保存在不同的输入缓冲区中。每个工艺对象均能够同时处理多个命令，而无需等待处理离散命令。

状态和错误消息（包括工艺对象的实际值）均随 PLC 中的工艺 DB 一起更新。更新与 OB XY 周期不同步运行，这表示新的实际值会在用户程序的每个扫描周期中返回。

注意
不要在重启 OB 中调用工艺功能。在重启阶段，集成工艺既不接受命令也不执行命令。

编程模型

下图显示了用户程序的结构。此步骤顺序与在 OB 1 或定时中断周期（OB 32 到 OB 35）中的处理无关。



每个命令步骤均包含这些子程序：

- 请求工艺对象状态
- 判断工艺对象错误
- 启动用于工艺对象的新命令
- 判断命令错误

所示的编程模型用作指导。在考虑前面所示的时序和顺序模型的情况下，可以自定义编程模型以满足用户要求。

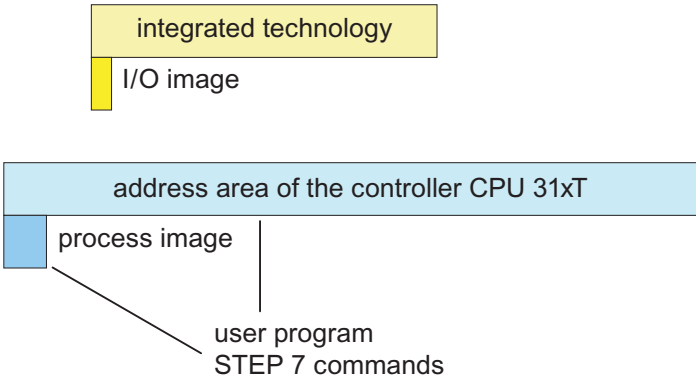
5.6 访问 CPU 31xT 的地址空间

CPU 31xT 特性可以区分 PLC 的地址区和集成工艺的地址区。



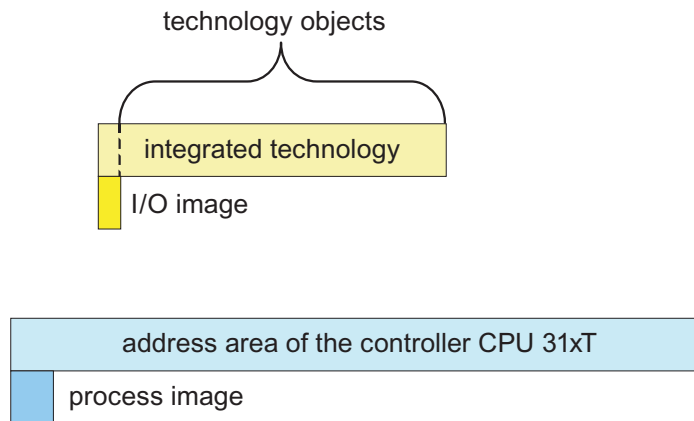
地址区	地址	
	CPU 315T-2DP	CPU 317T-2 DP
CPU 31xT PLC	0 到 2047	0 到 8191
过程映像区	0 到 127	0 到 255
集成工艺	0 到 1023	0 到 1023
I/O 映像区	0 到 63	0 到 63

通过 PLC 访问



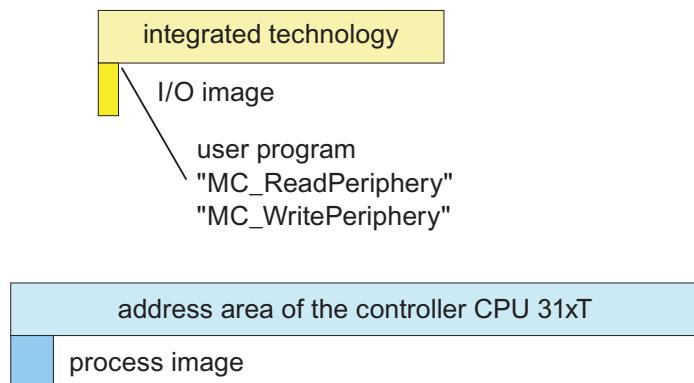
使用 STEP 7 命令，该用户程序可用于访问 PLC 的整个地址空间。

通过工艺对象进行访问



将通过工艺对象访问的地址分配到地址空间 **64** 到 **1023**。工艺对象不能访问此空间以外的地址。

通过“MC_ReadPeriphery”和“MC_WritePeriphery”工艺功能进行访问。



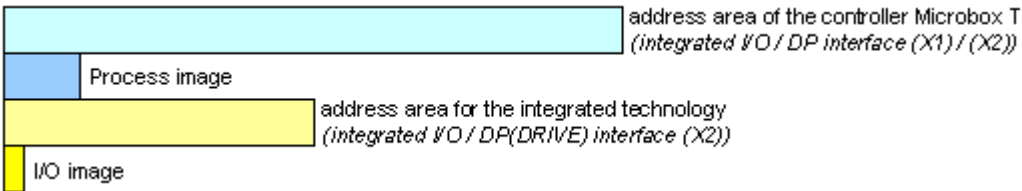
您可以使用“MC_ReadPeriphery”和“MC_WritePeriphery”工艺功能来访问集成工艺的 I/O 映像区（地址区 **0** 到 **63**）。

将要通过“MC_ReadPeriphery”和“MC_WritePeriphery”工艺功能访问的地址全部放入该地址区中。这些工艺功能不能访问此空间以外的地址。

可为该 I/O 映像区分配 DP(DRIVE) 接口 (X3) 的地址或集成 I/O 的地址。

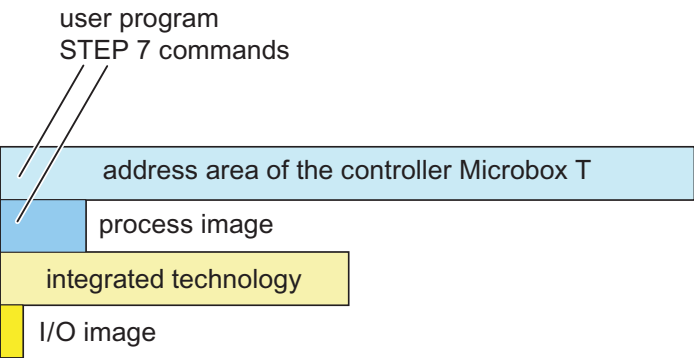
5.7 访问 MICROBOX T 的地址空间

MICROBOX T 提供了一个由控制器和集成工艺共享的地址空间。 可以为此地址空间分配 DP 接口 (X1) 和 DP(DRIVE) 接口 (X2) 的地址以及集成 I/O 的地址。



地址区	MICROBOX 420-T 地址
MICROBOX T PLC	0 到 2047
过程映像区（默认设置 *）	0 到 511
集成工艺	0 到 2047
I/O 映像区	0 到 63
*) 可以在 HW Config 中设置过程映像区的大小。	

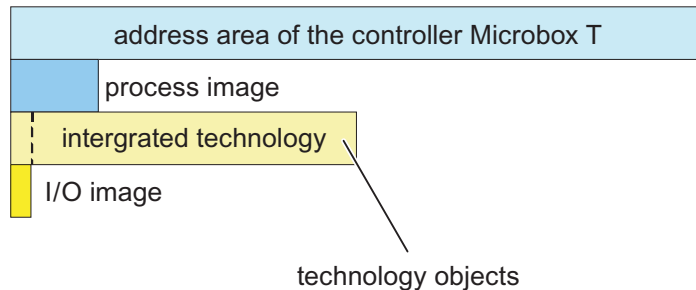
通过 PLC 访问



用户程序始终可以通过 STEP 7 命令访问 MICROBOX T 的整个地址区。 但是，这些地址必须在 HW Config 中分配给 DP 接口 (X1)。

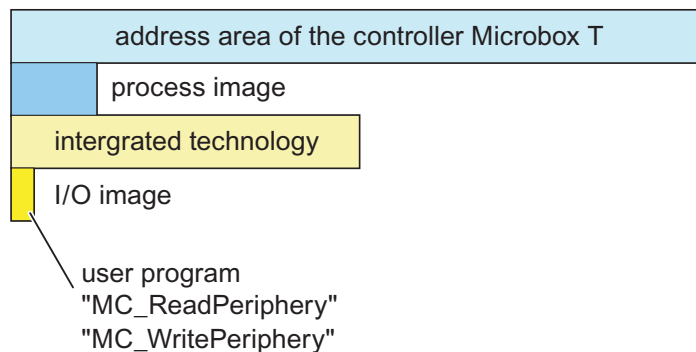
不能直接访问集成 I/O 的地址或已分配给 DP(DRIVE) 接口 (X2) 的地址。 如果在用户程序中访问这些地址，则控制器会像这些地址未被实际使用一样进行响应。

通过工艺对象进行访问



将通过工艺对象访问的地址分配到地址空间 **64** 到 **1023**。工艺对象不能访问此空间以外的地址。工艺对象只能访问在 HW Config 中分配给 DP(DRIVE) 接口 (X2) 或集成 I/O 的地址。

通过“MC_ReadPeriphery”和“MC_WritePeriphery”工艺功能进行访问。



您可以使用“MC_ReadPeriphery”和“MC_WritePeriphery”工艺功能来访问集成工艺的 I/O 映像区（地址区 **0** 到 **63**）。

将要通过“MC_ReadPeriphery”和“MC_WritePeriphery”工艺功能访问的地址全部放入该地址区中。这些工艺功能不能访问此空间以外的地址。

可为 I/O 映像区指定 DP(DRIVE) 接口 (X2) 的地址或集成 I/O 的地址。

说明

- “MC_ReadPeriphery”
如果“MC_ReadPeriphery”工艺功能用于访问已分配给 DP 接口 (X1) 的地址区或子区，则为这些地址读取的值无效。
- “MC_WritePeriphery”
如果“MC_WritePeriphery”工艺功能用于访问已分配给 DP 接口 (X1) 的地址区或子区，则不会对这些地址进行写访问。

5.8 监视活动命令

可以通过读取工艺功能的输出参数来监视活动命令。工艺功能的输出参数（状态输出）可以表明命令状态。

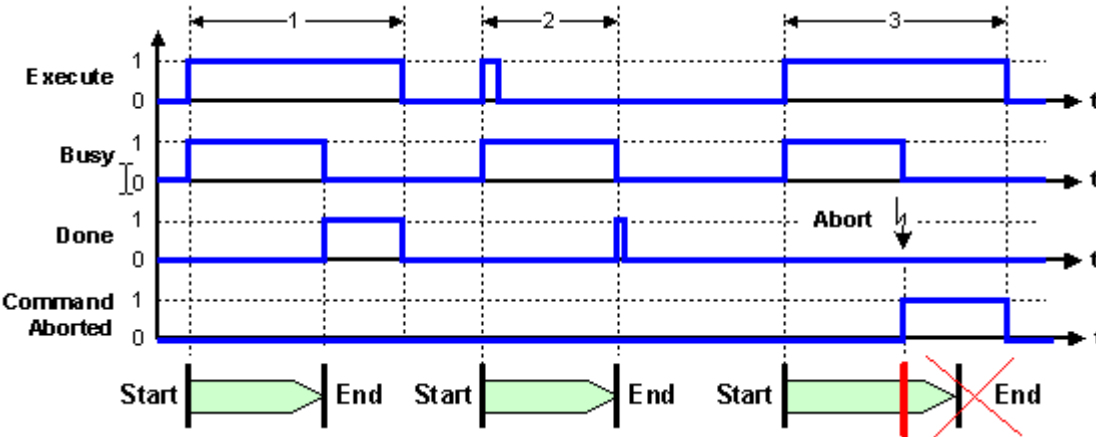
在命令处于活动状态期间，输出参数 *Busy* 的值为 *TRUE*；在命令执行完后，参数 *Busy* 返回值 *FALSE*。

其余输出参数表明至少一个周期的状态。只要输入参数 *Execute* = *TRUE*，就会显示并保存这些状态消息（另请参见下图中的应用 1 和 3）。

具有输出参数 *Done* 的工艺功能

分配了输出参数 *Done* 的命令具有已定义的终止操作。输出参数 *Done* 返回值 *TRUE*，以表明成功完成该命令（参阅应用 1）。

如果命令被过早取消，则置位输出参数 *CommandAborted*，而不置位输出参数 *Done*（另请参考应用 3）。如果向同一工艺对象输出一个新命令，或者发生错误事件，则会触发取消操作。



不具有输出参数 *Done* 的工艺功能

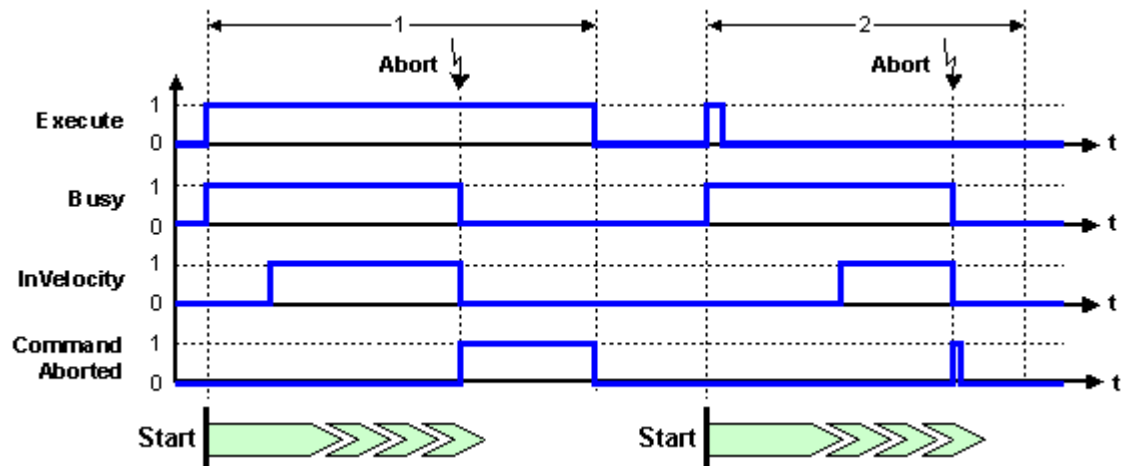
不具有输出参数 *Done* 的命令没有已定义的终止操作。操作或运动将会继续进行，直到收到取消命令或检测到错误。

这些工艺功能会返回一条状态消息，表明初次达到所需的操作状态（*InVelocity*, *InSync*, *InGear*, *DataValid* 等）。

示例 — 工艺功能“MC_MoveVelocity”（另请参考下图）

每个命令均以输入参数 *Execute* = 1 开始。输出参数 *Busy* 表明该命令正处于活动状态。在首次达到所需的速度后，输出参数 *InVelocity* 的状态将更改为 *TRUE*。如果命令未被取消，将始终保持活动状态 (*Busy* = *TRUE*)

如果在同一工艺对象启动新命令或检测到错误，则活动命令将被取消 (*Busy* = *FALSE*, *InVelocity* = *FALSE* 和 *CommandAborted* = *TRUE*)。



通过同一工艺功能取消命令

根据应用，可能有必要用同一类型的新命令来超驰活动命令（例如，通过逼近位置 200 来替代逼近位置 100）。

说明

使用本实例中的其它背景数据块。如果使用同一个背景数据块，则无法再监视工艺功能输出参数中的活动命令。例如，由于出错而无法启动新命令。通过使用其它背景数据块可以避免此问题。

这些背景数据块中的每一个都分配了一个单独的命令缓冲区。同一背景数据块的一个快速的命令顺序可能导致命令缓冲区发生一致性问题。因此，在处理命令缓冲区期间，会使用 *ErrorID* = 804C 拒绝新命令。这也会阻止对工艺功能的输出参数进行命令监视。

5.9 在 IPO 同步模式下启动轴命令

在默认情况下，集成工艺在插补器周期之后与下一个位置控制周期之前的时间间隔内执行命令。周期时间短可能导致将命令处理及由此而来的命令启动分发至多个位置控制周期。

集成工艺 V3.2.x 或更高版本支持在“IPO 同步”下启动单轴命令。可以在“工艺对象管理”中为相应的“轴”工艺对象设置“IPO 同步”属性。通过选择“视图”(View) > “专家模式”(Expert mode) 命令启动。

Technology DBs not yet created (Defaults list)							
	IPO synchronous	Reduction ratio	[ms]	Offset	[ms]	DB	Symbol
	☐	1	18.000	0	0.000	DB3	Axis_1
	☐	1	18.000	0	0.000	DB4	Axis_2
	☐	1	18.000	0	0.000	DB5	Axis_3
	☐	1	18.000	0	0.000	DB6	Axis_4
		1	18.000	0	0.000	DB7	Cam_1
		1	18.000	0	0.000	DB8	Cam_2
		1	18.000	0	0.000	DB9	Cam_3

“IPO 同步”属性允许轴命令以决定性的、快于平均启动速度的方式启动。

可在“IPO 同步”情况下启动的轴命令有：

MC_Power	MC_Home (无绝对值编码器调整)	MC_Stop
MC_Halt	MC_ChangeDataset	MC_MoveAbsolute
MC_MoveRelative	MC_MoveAdditive	MC_MoveSuperImposed
MC_MoveVelocity	MC_MoveToEndPos (同时也会在“IP 同步”模式下监视这些命令，直到到达挡块)	MC_GearIn
MC_CamIn	MC_GearOut	MC_CamOut
MC_Phasing	MC_SetTorqueLimit	MC_SetCharacteristic
MC_GearInSuperImposed	MC_CamInSuperImposed	MC_GearOutSuperImposed
MC_CamOutSuperImposed	MC_PhasingSuperImposed	

除非在表中另外指定，否则在“IPO 同步”模式下仅执行命令启动。命令在标准命令处理期间将继续被跟踪。

说明

每次启动“IPO 同步”命令时，都会将插补器周期的处理时间延长约 300 µs。避免同时使用大量的“IPO 同步”命令，以防 IPO 溢出。

“IPO 同步”和“非 IPO 同步”命令的顺序

并非“轴”工艺对象的所有命令都可以在“IPO 同步”模式下执行。“IPO 同步”和“非 IPO 同步”命令分别保存在不同的输入缓冲区中。混合使用“IPO 同步”和“非 IPO 同步”命令的条件：

- 一条轴的所有“IPO 同步”命令都按其启动的顺序进行处理。
- 一条轴的所有“非 IPO 同步”命令都按其启动的顺序进行处理。

请注意，混合使用“IPO 同步”和“非 IPO 同步”命令时，启动顺序和处理顺序可能会有所不同。

说明

要定义命令处理的混合顺序，请执行以下操作：

在向 TO 传送新命令之前，分析命令的 *Done*, *InGear*, *InSync* 或 *InClamping* 输出参数。

5.10 工艺功能处的错误和警告

工艺功能会指示输出参数 *Error*、*ErrorID* 或 *CommandAborted* 的所有错误。如果由于背景数据块长度不够而导致程序无法正确地更新所有输出参数，则可通过读取 BIE-Bit 位来评估此错误。

输出参数 *Error* 或 *ErrorID* 的警告和错误

输出参数 *Error* = *TRUE* 表示工艺功能无法启动命令。错误原因通过输出参数 *ErrorID* 的值 (*ErrorID* = 8xxx) 指示。

警告也与命令启动相关。输出参数 *ErrorID* 返回 *Wert 00xx* 以指示警告的原因。输出参数 *Error* 显示 *FALSE*。

如果工艺功能报告了错误，则您需要使用正确的参数或者在不同的时间调用它（假设该功能允许的话）。

既不需要，又不可能确认错误或警告。在参数 *Enable* 和 *Execute* 复位之前，错误保持活动状态。

输出参数 *CommandAborted*

输出参数 *CommandAborted* 报告在命令执行期间由新命令或错误触发的取消活动命令的操作。如果工艺 DB 的 *ErrorStatus* 未报告错误，则该命令由后续命令取消。

ErrorStatus 报告的错误表明在命令执行期间，集成工艺中的错误取消了该命令。

通过 BIE-Bit 的错误分析

当存在活动错误，或工艺功能的输出参数无法正确更新时，BIE-Bit（二进制结果位）的值将为 0。

控制器中背景数据块的长度不够会引起输出参数更新失败。

以下示例说明了如何在用户程序中判断 BRB 位：

```
CALL "MC_Power" , DB401
Axis      :=1
Enable    :=E5.0
Mode      :=0
StopMode:=0
Status    :=M100.0
Busy      :=M100.1
Error     :=M100.2
ErrorID   :=MW102
```

UN	BIE
=	A 16.0

如果检测到错误，则 BIE-Bit 将从 1 切换到 0。为输出 Q16.0 分配的值为 TRUE。

说明

为了验证该输出参数，应始终在调用工艺功能之后立即评估 BIE-Bit。
在下一步中，检查参数 Error 和 ErrorID。

5.11 工艺 DB (MCDevice 和 Trace) 中的错误

无效的“MCDevice”或“Trace”工艺 DB 会通过 ErrorID 8008 在相应的工艺 DB 中报告。如果工艺 DB 已被替换为新实例，则还会输出一条错误消息 (ErrorID 8009)。

在工艺 DB “MCDevice”和“Trace”中定义的错误指示有：

- ErrorID 变量
ErrorID 变量会返回最新检测到的错误的 ErrorID (ErrorID = 8xxx)。
通过用“MC_Reset”确认此变量可以清除其值 (Restart = FALSE)。
- ErrorBuffer[0..2] 变量（不适用于“Trace”工艺 DB）
前三个错误保存在变量数组 ErrorBuffer[0..2] 中。第一个错误写到数组元素 1，第二个错误写到数组元素 2，依次类推。
通过用“MC_Reset”确认此变量可以清除其值 (Restart = FALSE)。

确认错误

排除在 ErrorID 变量中检测到的所有错误的原因 (ErrorID = 8xxx)。无法确认在“MCDevice”和“Trace”工艺 DB 中表明的错误。

5.12 工艺 DB (Axes 和 External Encoders) 中的错误和警告

命令执行期间在工艺对象中出现的错误和警告都会在工艺 DB 中进行报告。这也适用于传送不一致或矛盾的动态值。

在工艺 DB (轴和外部编码器) 中定义的错误指示有：

- *Statusword.Error* 变量

如果出现至少一个错误，则此变量的值为 *TRUE*。可以在 *ErrorID* 或 *ErrorBuffer[0..2]* 变量通过 *ErrorID* 来分析错误。

- *Statusword.Errorstop* 变量

此轴由于错误事件已停止；工艺对象可能被禁用。

如果 *Statusword.Errorstop* 和 *Statusword.Error* 变量的值为 *TRUE*，则您可以通过读取 *ErrorID* 或 *ErrorBuffer[0..2]* 来分析错误。

- *ErrorID* 变量

ErrorID 变量显示由工艺对象返回的、最新检测到的错误 (*ErrorID = 8xxx*) 或警告 (*ErrorID = 00xx*) 的 *ErrorID*。

通过用“MC_Reset”确认此变量可以清除其值 (*Restart = FALSE*)。

- *ErrorBuffer[0..2]* 变量

前三个错误或警告事件（自上次确认后）保存在 *ErrorBuffer[0..2]* 变量数组中。第一个错误写到数组元素 1，第二个错误写到数组元素 2，依次类推。

通过用“MC_Reset”确认此变量可以清除其值 (*Restart = FALSE*)。

- *ErrorStatus.xxx* 变量

ErrorStatus.xxx 变量用于以位格式表明具体的轴错误信息（注意工艺 DB 中的注释）。

确认警告

(*ErrorID = 00xx*) 工艺 DB 中的警告可能需要，但不是必需通过调用“MC_Reset”进行确认。

但是，看一下 *ErrorBuffer[0..2]* 变量中的数据量。您可以通过适时地确认警告来清除 *ErrorBuffer[0..2]* 变量以便其它错误消息进入。

确认错误

为了确认所有的错误，您先要消除其原因，然后通过执行“MC_Reset”来确认这些错误 (*Restart = FALSE*)。然后，您可以通过调用“MC_Power”工艺功能来重新启用工艺对象。

 小心

几个未确认的错误可能导致 CPU 转为 STOP 模式。

5.13 工艺 DB（凸轮盘、测量输入和输出凸轮）中的错误和警告

命令执行期间在工艺对象中出现的错误和警告都会在工艺 DB 中进行报告。这也适用于传送不一致或矛盾的数据。

在工艺 DB（凸轮盘、测量输入和输出凸轮）中定义的错误指示有：

- *ErrorID* 变量
ErrorID 变量显示工艺对象的最新检测到的错误 (*ErrorID = 8xxx*) 或警告 (*ErrorID = 00xx*) 的 *ErrorID*。
通过用“MC_Reset”确认此变量可以清除其值 (*Restart = FALSE*)。
- *ErrorBuffer[0..2]* 变量
前三个错误或警告事件（自上次确认后）保存在 *ErrorBuffer[0..2]* 变量数组中。
第一个错误写到数组元素 1，第二个错误写到数组元素 2，依次类推。
通过用“MC_Reset”确认此变量可以清除其值 (*Restart = FALSE*)。

确认警告

工艺 DB (*ErrorID = 00xx*) 中的警告可能需要，但不是必需通过调用“MC_Reset”进行确认。

但是，注意 *ErrorBuffer[0..2]* 变量中的数据量。您可以通过适时地确认警告来清除 *ErrorBuffer[0..2]* 变量以便其它错误消息进入

确认错误

为了确认所有的错误，您先要消除其原因，然后通过执行“MC_Reset”来确认这些错误 (*Restart = FALSE*)。

 小心

几个未确认的错误可能导致 CPU 转为 STOP 模式。

5.14 虚拟轴的响应

尽管虚拟轴用作控制变量，但其既不是用作位置控制系统，也不是用作驱动器，更不是用作编码器接口。虚拟轴的实际值总是设置为等于设定值。缺少位置控制系统以及缺少驱动器和编码器接口，会导致在命令处理期间发生某些特殊情况，如下所述。

变量 *Statusword.DriveEnabled* 的特殊响应

如果虚拟轴的启用状态在“MC_Power”的输入参数 *Enable* 中被复位，则相应工艺 DB 的 *Statusword.DriveEnabled* 变量的状态保持为 *TRUE*。

虚拟轴的 *Statusword.DriveEnabled* 变量始终为 *TRUE*。

在 T-CPU 处于 RUN 状态时编辑参数

- **重启**

由于虚拟轴不用作位置控制、驱动器和编码器接口，因此要激活参数更改，通常不需要执行重启（与对话框的帮助系统中的技术规范或工艺参数的列表中的技术规范无关）。

相应工艺 DB 的 *Statusword.RequestRestart* 变量表明是否需要重启。

- **参数更改**

所有轴参数均在工艺参数列表和 DB 参数的对话框中列出。

请注意不可能读取或编辑虚拟轴的所有参数。

5.15 使用 FC400“DB2INT”进行符号编程

如果您要使用工艺块的符号名称来调用工艺功能，请使用“S7-Tech”库的 FC400“DB2INT”。

以下示例说明了使用 FC400“DB2INT”的一个应用：

```
CALL  "DB2Int"
  DBref  := "Achse_1_"
  RET_VAL:= #TECH_DB

CALL  "MC_Power" , DB401
  Axis   := #TECH_DB
  Enable :=
  Mode   :=
  StopMode:=
  Status :=
  Busy   :=
  Error  :=
  ErrorID :=
```

1. 定义一个类型为 INT 的变量（在此实例中为变量“TECH_DB”）。
2. 通过执行“CALL”命令来调用 FC400“DB2INT”。
3. 在输入参数 *Dbref* 中设置工艺 DB 的符号名称。
4. 在输出参数 *RET_VAL* 中设置定义的变量值。
5. 接下来，通过执行“CALL”命令来调用工艺功能（此处，FB401 工艺功能为“MC_Power”）。
6. 在输入参数 *Axis* 中设置定义的变量（对于其它工艺功能，在输入参数 *Master*, *Slave*, *CamSwitch*, *MeasureInput* 或 *CamTable* 中设置定义的变量）。

5.16 对轴特定的参数更改进行编程

通过调用工艺功能“MC_WriteParameter”，可以在 T-CPU 处于 RUN 状态时在用户程序中更改在 S7T Config 中设置的工艺对象的参数。这些更改在下一次电源关之前或 CPU 存储器复位之前将一直处于活动状态。

某些参数不会被激活，除非已启动（重启）寻址的工艺对象。这种情况仅在工艺对象被禁用的情况下才可能发生。在 T-CPU 中的电源断开、接通之后，在 S7T Config 中设置的参数再次生效。

编辑参数主要有两个选项：

- 通过编辑相应的输入参数数据。
- 在工艺功能的背景数据块中对参数进行了更改。您可以选择不同的背景数据块来编辑不同的参数。

为工艺功能创建一个新背景数据块“MC_WriteParameter”

1. 在 SIMATIC Manager 中选择块文件夹。
2. 选择**插入 > S7 块 > 数据块** (Insert > S7 block > Data block)，以插入一个新数据块。
3. 指定所需的名称，然后选择类型“背景数据块”和相应的 FB 407。
4. 您还可以定义符号名称和输入符号注释。
5. 单击“确定”(OK)，关闭 DB 创建 (DB creation)。

通过编辑输入参数数据来更改参数

为了更改工艺对象参数，您需要其参数编号以及关于类型、访问模式和允许的取值范围的信息。此信息可从参数列表中获取。但是，建议使用背景数据块的组态视图，因为您也可以从中搜索特定参数。

1. 在 **SIMATIC Manager** 中打开背景数据块。
“DB 参数”工具随即打开并在组态视图中显示背景数据块的内容。
2. 从下拉列表中选择工艺对象以编辑其参数。
3. 从树形视图中选择相关的组和参数。
4. 将鼠标指针放在此参数上。工具提示会向您显示参数编号、类型和访问模式（“读取”、“写入”）。记下这些值，以便设置工艺功能的输入参数。您也可以将工艺参数列表从在线帮助输出到打印机。
5. 您可以在下拉列表的“值”(Value) 区域中或工具提示中查看允许的参数值。
6. 关闭组态视图。在随后出现的对话框中确认或拒绝所进行的更改。在工艺功能的输入参数中设置数据时，与背景数据块的已组态内容无关。
7. 在用户程序中使用背景数据块来调用“MC_WriteParameter”工艺功能，然后设置您记下的输入参数值。参数赋值命令由 *Execute* 输入中的上升沿启动。

此操作步骤的优点是仅需要一个背景数据块。但是，回缩是一次仅可以编辑一个参数，而且您必须等待 **MC_WriteParameter** 工艺功能的响应，然后才能编辑下一个参数。

使用其它背景数据块来编辑参数

每个参数更改需要创建并组态新背景数据块。

1. 在 **SIMATIC Manager** 中打开背景数据块。
“DB 参数”工具随即打开并在组态视图中显示背景数据块的内容。
2. 从下拉列表中，选择包含您要使用此背景数据块编辑的参数的工艺对象。
3. 从树形视图中选择相关的组和参数。
4. 从“值”下拉列表中选择所需的值，或键入值。
5. 关闭对话框，然后在下一个对话框中确认所做的更改。
6. 使用相关的背景数据块在用户程序中调用“MC_WriteParameter”工艺功能。
7. 仅设置 *Execute* 输入参数。输入参数 *Execute* 中的正值会启动命令来激活具有背景数据块的值的参数。

对所有的参数更改重复此操作步骤。

此操作步骤的优点是您可以使用 **DB** 参数随意设置参数。还有一个优点是您可以接连编辑多个参数。但是，缺点是您总是需要多个 **DB** 或背景数据块。

工艺功能

6.1 总览

6.1.1 总览 — 编程

要求

如果存在下列情况，则运动控制命令可以借助于功能块进行编程：

- 在 HW Config 中的站组态已保存并编译
- 在 S7T Config 中已插入并配置工艺对象
- 在 S7T Config 的“工艺对象管理”(Technology Objects Management) 对话框中已生成工艺 DB

通过相关工艺 DB 编号来定义对工艺对象的参考。

功能块

可编程功能块：

功能	编号	名称	说明
单轴	FB 401	MC_Power (页 439)	启用/禁用轴
	FB 403	MC_Home (页 452)	使轴回原点/设置轴
	FB 404	MC_Stop (页 464)	停止轴并阻止新运动命令
	FB 405	MC_Halt (页 470)	正常停止
	FB 409	MC_ChangeDataset (页 475)	更改数据集
	FB 410	MC_MoveAbsolute (页 481)	绝对定位
	FB 411	MC_MoveRelative (页 493)	相对定位
	FB 412	MC_MoveAdditive (页 504)	相对当前目标位置的相对定位
	FB 413	MC_MoveSuperImposed (页 510)	叠加定位
	FB 414	MC_MoveVelocity (页 518)	以预置的速度运动
	FB 415	MC_MoveToEndPos (页 528)	移动到固定挡块/终端
	FB 437	MC_SetTorqueLimit (页 536)	激活/取消激活扭矩限制
	FB 439	MC_SetCharacteristic (页 541)	激活阀特性

功能	编号	名称	说明
同步操作	FB 420	MC_GearIn (页 545)	启动齿轮传动
	FB 440	MC_GearInSuperImposed (页 561)	启动叠加的齿轮传动
	FB 422	MC_GearOut (页 556)	结束齿轮传动
	FB 442	MC_GearOutSuperImposed (页 574)	结束叠加的齿轮传动
	FB 421	MC_CamIn (页 578)	启动凸轮转动
	FB 441	MC_CamInSuperImposed (页 599)	启动叠加的凸轮转动
	FB 423	MC_CamOut (页 594)	结束凸轮转动
	FB 443	MC_CamOutSuperImposed (页 610)	结束叠加的凸轮转动
	FB 424	MC_Phasing (页 615)	更改引导轴和跟随轴之间的相移
	FB 444	MC_PhasingSuperImposed (页 622)	更改叠加的相移
高级功能	FB 430	MC_CamSwitch (页 659)	基于位置的凸轮/单向输出凸轮
	FB 431	MC_CamSwitchTime (页 667)	基于时间的凸轮
	FB 461	MC_CamTrack (页 673)	激活凸轮轨迹
	FB 462	MC_ReadCamTrackData (页 681)	读出凸轮轨迹
	FB 463	MC_WriteCamTrackData (页 686)	写入凸轮轨迹
	FB 432	MC_ExternalEncoder (页 697)	外部编码器
	FB 433	MC_MeasuringInput (页 691)	测量输入
凸轮盘	FB 434	MC_CamClear (页 631)	清除凸轮
	FB 435	MC_CamSectorAdd	添加凸轮扇区
	FB 436	MC_CamInterpolate (页 645)	插补凸轮
	FB 438	MC_GetCamPoint	从凸轮读取点

功能	编号	名称	说明
基本功能	FB 402	MC_Reset (页 704)	确认错误/中断
	FB 460	MC_ActivateTO (页 709)	激活/取消激活工艺对象
	FB 457	MC_ActivateDP Slave (页 714)	激活/取消激活 DP 从站
	FB 406	MC_ReadSysParameter (页 719)	读取参数
	FB 407	MC_WriteParameter (页 723)	更改参数
	FB 450	MC_ReadPeriphery (页 733)	读取工艺 I/O
	FB 451	MC_WritePeriphery (页 739)	写入工艺 I/O
	FB 453	MC_ReadRecord (页 745)	读取数据记录
	FB 454	MC_WriteRecord (页 751)	写入数据记录
	FB 455	MC_ReadDriveParameter (页 757)	读取驱动参数
	FB 456	MC_WriteDriveParameter (页 764)	写入驱动参数

6.1.2 按编号排序的工艺功能列表

编号	名称	功能	说明
FB 401	MC_Power (页 439)	单轴	启用/禁用轴
FB 402	MC_Reset (页 704)	基本功能	确认错误/中断
FB 403	MC_Home (页 452)	单轴	使轴回原点/设置轴
FB 404	MC_Stop (页 464)	单轴	停止轴并阻止新运动命令
FB 405	MC_Halt (页 470)	单轴	正常停止
FB 406	MC_ReadSysParameter (页 719)	基本功能	读取参数
FB 407	MC_WriteParameter (页 723)	基本功能	更改参数
FB 409	MC_ChangeDataset (页 475)	单轴	更改数据集
FB 410	MC_MoveAbsolute (页 481)	单轴	绝对定位
FB 411	MC_MoveRelative (页 493)	单轴	相对定位
FB 412	MC_MoveAdditive (页 504)	单轴	相对当前目标位置的相对定位
FB 413	MC_MoveSuperImposed (页 510)	单轴	叠加定位
FB 414	MC_MoveVelocity (页 518)	单轴	以预置的速度运动
FB 415	MC_MoveToEndPos (页 528)	单轴	移动到固定挡块/终端
FB 420	MC_GearIn (页 545)	同步操作	启动齿轮传动
FB 421	MC_CamIn (页 578)	同步操作	启动凸轮转动
FB 422	MC_GearOut (页 556)	同步操作	结束齿轮传动
FB 423	MC_CamOut (页 594)	同步操作	结束凸轮转动
FB 424	MC_Phasing (页 615)	同步操作	更改引导轴和跟随轴之间的相移
FB 430	MC_CamSwitch (页 659)	扩展功能	基于位置的凸轮/单向输出凸轮
FB 431	MC_CamSwitchTime (页 667)	扩展功能	基于时间的凸轮
FB 432	MC_ExternalEncoder (页 697)	扩展功能	外部编码器
FB 433	MC_MeasuringInput (页 691)	扩展功能	测量输入
FB 434	MC_CamClear (页 631)	凸轮盘	清除凸轮
FB 435	MC_CamSectorAdd	凸轮盘	添加凸轮扇区
FB 436	MC_CamInterpolate (页 645)	凸轮盘	插补凸轮
FB 437	MC_SetTorqueLimit (页 536)	单轴	激活/取消激活扭矩限制

编号	名称	功能	说明
FB 438	MC_GetCamPoint	凸轮盘	从凸轮读取点
FB 439	MC_SetCharacteristic (页 541)	单轴	激活阀特性
FB 440	MC_GearInSuperImposed (页 561)	同步操作	启动叠加的齿轮传动
FB 441	MC_CamInSuperImposed (页 599)	同步操作	启动叠加的凸轮转动
FB 442	MC_GearOutSuperImposed (页 574)	同步操作	结束叠加的齿轮传动
FB 443	MC_CamOutSuperImposed (页 610)	同步操作	结束叠加的凸轮转动
FB 444	MC_PhasingSuperImposed (页 622)	同步操作	更改叠加的相移
FB 450	MC_ReadPeriphery (页 733)	基本功能	读取工艺 I/O
FB 451	MC_WritePeriphery (页 739)	基本功能	写入工艺 I/O
FB 453	MC_ReadRecord (页 745)	基本功能	读取数据记录
FB 454	MC_WriteRecord (页 751)	基本功能	写入数据记录
FB 455	MC_ReadDriveParameter (页 757)	基本功能	读取驱动参数
FB 456	MC_WriteDriveParameter (页 764)	基本功能	写入驱动参数
FB 457	MC_ActivateDP Slave (页 714)	基本功能	激活/取消激活工艺对象
FB 460	MC_ActivateTO (页 709)	基本功能	激活/取消激活 DP 从站
FB 461	MC_CamTrack (页 673)	扩展功能	激活凸轮轨迹
FB 462	MC_ReadCamTrackData (页 681)	扩展功能	读出凸轮轨迹
FB 463	MC_WriteCamTrackData (页 686)	扩展功能	写入凸轮轨迹

说明

如果要从其它项目中使用现有 FB，则可在 SIMATIC Manager 中重命名 FB。请注意，文档引用默认 FB 编号。

6.1.3 按字母顺序排序的工艺功能列表

名称	编号	功能	说明
MC_ActivateDPSlave (页 714)	FB 457	基本功能	激活/取消激活 DP 从站
MC_ActivateTO (页 709)	FB 460	基本功能	激活/取消激活工艺对象
MC_CamClear (页 631)	FB 434	凸轮盘	清除凸轮
MC_CamIn (页 578)	FB 421	同步操作	启动凸轮转动
MC_CamInSuperImposed (页 599)	FB 441	同步操作	启动叠加的凸轮转动
MC_CamInterpolate (页 645)	FB 436	凸轮盘	插补凸轮
MC_CamOut (页 594)	FB 423	同步操作	结束凸轮转动
MC_CamOutSuperImposed (页 610)	FB 443	同步操作	结束叠加的凸轮转动
MC_CamSectorAdd	FB 435	凸轮盘	添加凸轮扇区
MC_CamSwitch (页 659)	FB 430	扩展功能	基于位置的凸轮/单向输出凸轮
MC_CamSwitchTime (页 667)	FB 431	扩展功能	基于时间的凸轮
MC_CamTrack (页 673)	FB 460	扩展功能	激活凸轮轨迹
MC_ChangeDataset (页 475)	FB 409	单轴	更改数据集
MC_ExternalEncoder (页 697)	FB 432	扩展功能	外部编码器
MC_GearIn (页 545)	FB 420	同步操作	启动齿轮传动
MC_GearInSuperImposed (页 561)	FB 440	同步操作	启动叠加的齿轮传动
MC_GearOut (页 556)	FB 422	同步操作	结束齿轮传动
MC_GearOutSuperImposed (页 574)	FB 442	同步操作	结束叠加的齿轮传动
MC_GetCamPoint	FB 438	凸轮盘	从凸轮读取点
MC_Halt (页 470)	FB 405	单轴	正常停止
MC_Home (页 452)	FB 403	单轴	使轴回原点/设置轴
MC_MeasuringInput (页 691)	FB 433	扩展功能	测量输入
MC_MoveAbsolute (页 481)	FB 410	单轴	绝对定位
MC_MoveAdditive (页 504)	FB 412	单轴	相对当前目标位置的相对定位
MC_MoveRelative (页 493)	FB 411	单轴	相对定位
MC_MoveSuperImposed (页 510)	FB 413	单轴	叠加定位
MC_MoveToEndPos (页 528)	FB 415	单轴	移动到固定挡块/终端

名称	编号	功能	说明
MC_MoveVelocity (页 518)	FB 414	单轴	以预置的速度运动
MC_Phasing (页 615)	FB 424	同步操作	更改引导轴和跟随轴之间的相移
MC_PhasingSuperImposed (页 622)	FB 444	同步操作	更改叠加的相移
MC_Power (页 439)	FB 401	单轴	启用/禁用轴
MC_ReadCamTrackData (页 681)	FB 462	扩展功能	读出凸轮轨迹
MC_ReadDriveParameter (页 757)	FB 455	基本功能	读取驱动参数
MC_ReadPeriphery (页 733)	FB 450	基本功能	读取工艺 I/O
MC_ReadRecord (页 745)	FB 453	基本功能	读取数据记录
MC_ReadSysParameter (页 719)	FB 406	基本功能	读取参数
MC_Reset (页 704)	FB 402	基本功能	确认错误
MC_SetCharacteristic (页 541)	FB 439	单轴	激活阀特性
MC_SetTorqueLimit (页 536)	FB 437	单轴	激活/取消激活扭矩限制
MC_Stop (页 464)	FB 404	单轴	停止轴并阻止新运动命令
MC_WriteCamTrackData (页 686)	FB 463	扩展功能	写入凸轮轨迹
MC_WriteDriveParameter (页 764)	FB 456	基本功能	写入驱动参数
MC_WriteParameter (页 723)	FB 407	基本功能	更改工艺对象参数
MC_WritePeriphery (页 739)	FB 451	基本功能	写入工艺 I/O
MC_WriteRecord (页 751)	FB 454	基本功能	写入数据记录

工艺功能 — 单轴

6.2.1 FB401 MC_Power — 禁用/启用轴

用 FB401“MC_Power”来启用/禁用轴

用途

- 使用“MC_Power”工艺功能启用或禁用轴。
- 使用该工艺功能指定轴的操作模式。

6.2 工艺功能 — 单轴

- 使用该工艺功能指定用于处理轴（当轴被禁用或 CPU 转换为 STOP 时）的紧急程序。
- 使用该工艺功能控制驱动器的集成制动控制（例如，当操作“吊轴”时）。支持以下驱动器的制动控制：
 - SIMODRIVE 611U 通用型
 - SINAMICS S120
 - MASTERDRIVES 运动控制 Plus

适用对象

- 速度控制轴
- 定位轴
- 同步轴

先决条件

- 在阻止启用轴的工艺 DB 中，可能没有激活错误。
- 每个轴可能只能激活工艺功能一次。

超驰命令

MC_Power 命令不能被其它任何命令取消。

MC_Power 命令不能通过 *Enable = TRUE* 来取消其它任何命令。

MC_Power 命令通过 *Enable = FALSE* 来取消此工艺对象的其它所有命令。

新命令 — 活动的单个命令 (3)

输入参数

参数	数据类型	初始值	说明
<i>Axis</i>	INT	0	工艺 DB 的编号
<i>Enable</i>	BOOL	FALSE	<i>Enable</i> = TRUE 时，系统会尝试启用该轴。如果没有错误阻止启用轴，则轴会被启用。 如果在相应的工艺 DB 中，<i>Statusword.Error</i> 和 <i>Statusword.Errorstop</i> 变量均为 TRUE，则意味着出现了一个特定错误阻止了该轴启用。 如果 <i>Enable</i> = FALSE，则取消所有活动命令。删除使能后通过“MC_SetTorqueLimit”工艺功能设置的扭矩限制保持不变。
<i>Mode</i>	INT	0	轴模式：
			值 = 0: 默认（取决于轴组态） 速度控制轴： 速度控制 （没有编码器的速度控制轴） 定位轴和同步轴： 位置控制或速度控制 （“MC_MoveVelocity” <i>PositionControl</i> = FALSE）
			值 = 1: 通过速度控制启用轴： 速度控制轴： 速度控制 （没有编码器的速度控制轴） 定位轴和同步轴： 轴的启用是速度控制的
			值 = 2: 启用了功率部件的跟踪模式
			值 = 3: 禁用了功率部件的跟踪模式
			值 = 4: 仿真模式
			值 = 5: 启用轴进行位置控制操作 （仅限于定位轴和同步轴）

参数	数据类型	初始值	说明	
<i>StopMode</i>	INT	0	值 = 0:	<i>DefaultStop</i> (急停减速度)
			值 = 1:	<i>FastStop</i> (加速度的硬件限制)
			值 = 2:	<i>TimeStop</i> 轴在组态的急停时间 (STOP 时间) 内停止
			值 = 3:	<i>OFF2Stop</i> (仅在液压轴) 该轴的控制信号通过已组态的斜坡更改为已编程的替换控制信号值 (输入参数 <i>QOutputValue</i>)。出现“缺少启用信号”错误, 运动命令被取消。 已组态的任何启用输出被复位。您可以在 S7T Config 中的系统变量 <i>userdefaultqfaxis.maxderivative.qoutput</i> 中设置斜坡。您也可以使用工艺功能 “MC_WriteParameter” (参数 5009) 更改斜坡值。 该值定义为百分比/秒。默认值为 100%/秒。
			值 = 4:	<i>GearStop</i> 同步轴会保持同步操作, 直到引导轴变为静止。
			值 = 5	<i>Rampenhalt</i> (仅限于电气实轴) 通过使用斜坡函数发生器停止轴
			值 = 6	<i>Schnellhalt</i> (仅限于电气实轴) 通过使用快速停止斜坡停止轴

参数	数据类型	初始值	说明
<i>QOutputValue</i>	REAL	0.0	Q 阀的控制信号替代值。 控制信号的替换值定义为该阀的最大设定值电压输出的百分比（与阀打开的百分比一致）。 这些值的设置范围为 <i>-100.0 %</i> 到 <i>+100.0 %</i>。 如果通过设置输入参数 <i>Enable = FALSE</i> 和选中 <i>StopMode = 3</i> 禁用该轴，则会激活替换值。 设置控制信号的替换值以阻止液压传动器在轴禁用时移动。 <i>QOutputValue</i> 的已修改值在上升沿和下降沿的 <i>Enable</i> 输入参数中被激活。如果在输入参数 <i>QOutputValue</i> 的 <i>Enable</i> 的下降沿上检测到无效的值，则“MC_Power”工艺功能会针对一个周期的持续时间输出 ErrorID 808B。在本实例中，“MC_Power”为 <i>QOutputValue</i> 使用了一个值，其在上升沿在 <i>Enable</i> 时输出时有效。
<i>FOutputValue</i>	REAL	0.0	此输入参数尚未受集成工艺（固件 V3.2.x）的支持。

输出参数（状态输出）

参数	数据类型	初始值	说明	
<i>Status</i>	BOOL	<i>FALSE</i>	轴启用状态：	
			<i>FALSE</i>	没有启用信号。轴被禁用 轴不存在或不接受新的运动命令。禁用驱动器功率部件。 如果轴已锁定， <i>Status</i> 可通过延迟更改为 <i>FALSE</i> 。
			<i>TRUE</i>	启用 仅当控制和驱动器之间的循环通信处于活动状态且激活的编码器的实际位置值有效时才启用轴。检查工艺 DB 上的 <i>Statusword.CyclicInterface</i> 和 <i>Statusword.EncoderValid</i> 。 如果出现了阻止启用的错误（在相关工艺 DB 中，由 <i>Statusword.Error = TRUE</i> 和 <i>Statusword.Errorstop = TRUE</i> 指示），则使能信号被复位。
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i>	正在执行命令
<i>Error</i>	BOOL	<i>FALSE</i>	<i>FALSE</i>	命令启动时没有错误。
			<i>TRUE</i>	命令启动时有错误。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。	

注意
一条由于错误而被切断的轴，在消除并确认错误之后通过 <i>Enable = TRUE</i> 重新启用。

说明

“MC_Power”工艺功能影响工艺 DB 中的 *Statusword* 变量的下列状态位：

- *DriveEnable*
- *FollowUpControl*
- *Simulation*

推荐的轴禁用操作步骤

要禁用轴，请执行以下操作：

1. 停止驱动器
2. 在驱动器停止后禁用轴 (*Enable* = *FALSE*)。

参数 *Mode* — 操作模式

通过输入 *Enable* 的上升沿来启用驱动器。

轴工艺	支持的操作模式	
	实轴	虚拟轴
速度控制轴	<i>Mode</i> = 0: 速度控制 <i>Mode</i> = 3: 跟随模式 <i>Mode</i> = 4: 仿真	<i>Mode</i> = 0: 速度控制
定位轴	<i>Mode</i> = 0: 位置控制 <i>Mode</i> = 3: 跟随模式 <i>Mode</i> = 4: 仿真	<i>Mode</i> = 0: 位置控制
同步轴	<i>Mode</i> = 0: 位置控制 <i>Mode</i> = 3: 跟随模式 <i>Mode</i> = 4: 仿真	<i>Mode</i> = 0: 位置控制

参数 *StopMode* — 急停程序

如果驱动器在处于 RUN (MC_Power 中的下降沿) 时或 CPU 转为 STOP 时被禁用，则启动急停程序。轴的响应不能在输入 *StopMode* 中进行定义。在输入参数 *Enable* 中的下降沿或上升沿处确定 *StopMode*。

如果扭矩减少处于活动状态，则禁用参数 *StopMode*。在这种情况下，轴立即被禁用。驱动器根据其内部减速度斜坡设置转为静止。

在 CPU 转为 STOP 后启动的急停程序必须在“最大关机时间”内执行。在“最大关机时间”结束时，CPU 转为 STOP 模式。驱动器将按照在驱动器组态中定义的方式停止，而不是根据所选的 *StopMode* 停止。

可以在 S7T Config 的“系统时钟”(System clocks)对话框中组态“最大关机时间”（目标系统 > 设置系统时钟 [Target system > Set System Clocks] 菜单命令）。

在重启之前，必须通过调用“MC_Reset”工艺功能来确认所有错误。

StopMode	轴的响应
值 = 0: DefaultStop	轴通过激活“急停减速度”功能逐渐停止。可以在 S7T Config 中（轴 > 限制 [Axis > Limits] 对话框中“动态响应”[Dynamic response] 选项卡的“通过预编程的减速斜坡停止”[Stop with preprogrammed deceleration ramp] 输入框）定义“急停减速度”功能。此输入框不适用于虚拟轴。在这种情况下，速度设定值立即设置为零。
值 = 1: FastStop	轴会以最大减速度（S7T Config > 限制 > “动态响应”选项卡 > 硬件限制 [S7T Config > Limits > "Dynamic response" tab > Hardware limits]）逐渐停止。
值 = 2: TimeStop	轴在为紧急停止组态的“停止时间”（S7T Config > 默认 > “动态”选项卡 > 停止时间 [S7T Config > Default > "Dynamics" tab > Stop time]）内逐渐停止。
值 = 3: OFF2Stop	驱动器由脉冲禁止信号禁用，并转为静止 (OFF2)。在轴的工艺 DB 中，由 8040 指示缺少启用信号。
值 = 4: GearStop	同步轴会保持其状态，直到引导轴变为静止。该轴随即被禁用。 在其它任何情况中，该轴都会根据默认轴组态逐渐停止（例如，速度控制的轴）。
值 = 5: 斜坡停止	通过斜坡函数发生器停止轴
值 = 6: 快速停止	通过快速停止斜坡来停止轴

说明

如果轴在跟随模式下运行，则选定的 *StopMode* 是无关的，因为该轴无法动态减速！因此将有一条旋转轴在不确定的时间内变为静止。

注意

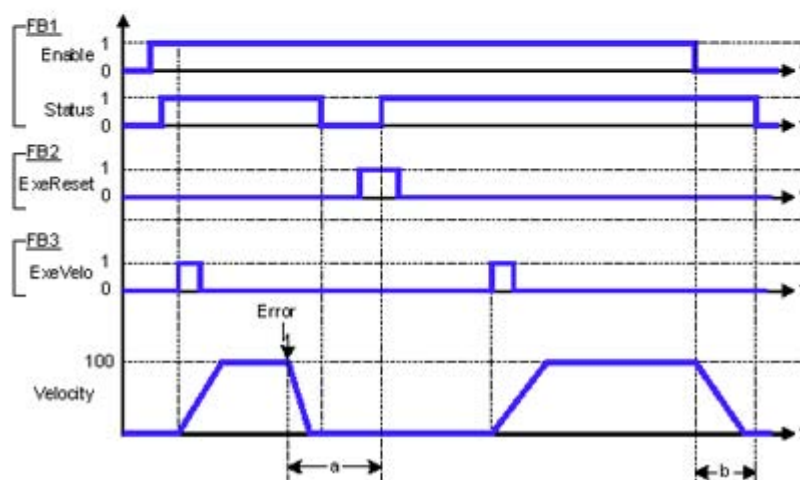
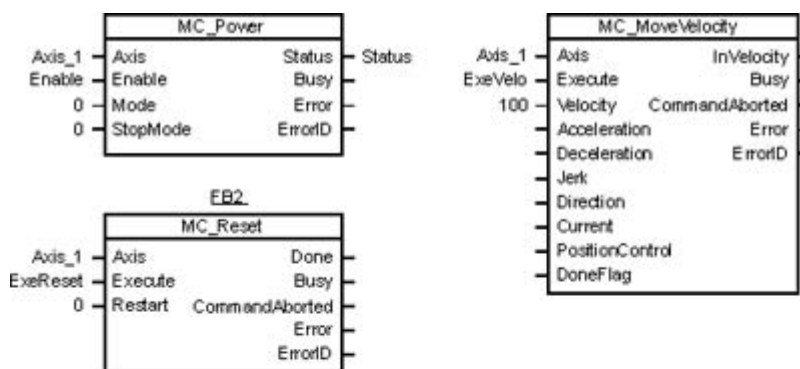
在 *StopMode* = 4 且已设置凸轮传动时，跟随轴可能在引导轴变为静止前重启！

MC_Power、MC_Reset — 示例

该示例说明了工艺功能“MC_Power”（轴启用）、“MC_Reset”（错误确认）与轴运动工艺功能（此处为“MC_MoveVelocity”）之间的相互作用。

a) 该轴已启用并加速到 100。出现了一个错误使轴启用信号复位。该错误已消除并通过“MC_Reset”确认。该轴再次启用。

b) 该轴已加速到 100。*Enable* 信号已从工艺功能“MC_Power”中清除。该轴现在按定义的减速度逐渐停止，随即被禁用。

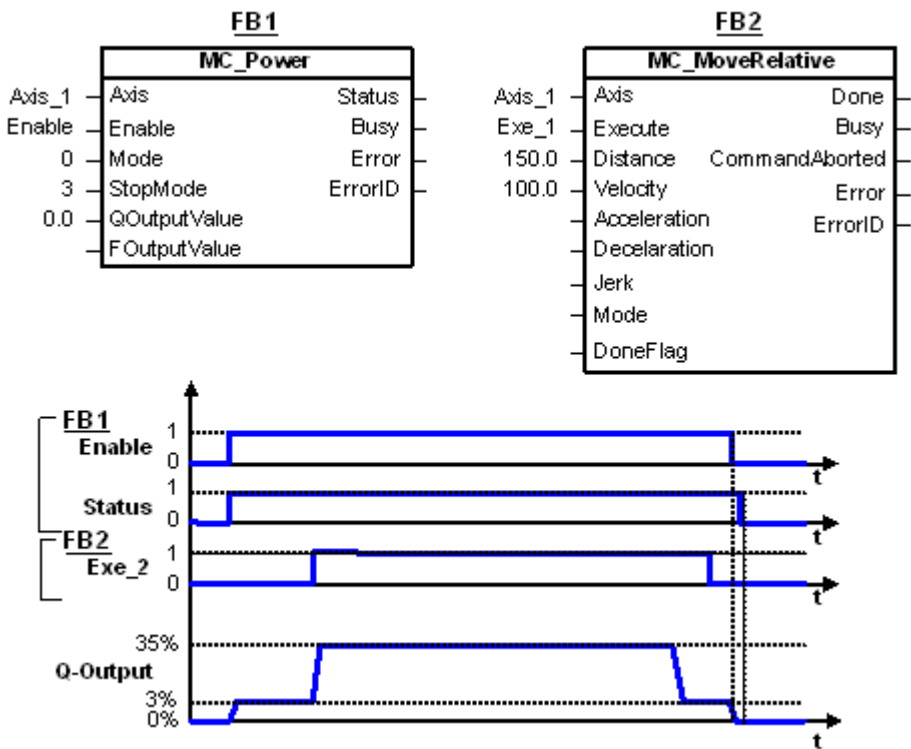


MC_Power — 示例 — “液压轴”

比例阀可用于运行两条轴。这两个轴具有不同的阀控曲线。“轴 1”的速度 $v = 0$ 为阀位置的 3 %。“轴 2”的速度 $v = 0$ 为阀位置的 0 %。

在此示例中，启用“轴 1”用于位置控制。因此，输出 3 % 的主信号以使轴保持在其原位。“轴 1”相对于较小的值定位。在定位结束时，该轴再次“保持”在其原位，速度为 0。当“轴 1”禁用时，受控变量的值逐渐减小为已编程的替换值 $QOutputValue = 0\%$ 。

通过调用 MC_Power 启用“轴 2”后，受控变量将具有适当的值。

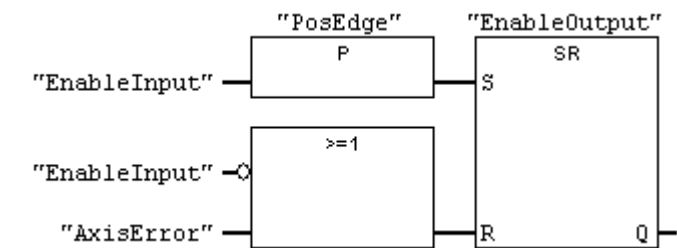


沿触发的驱动器启用功能

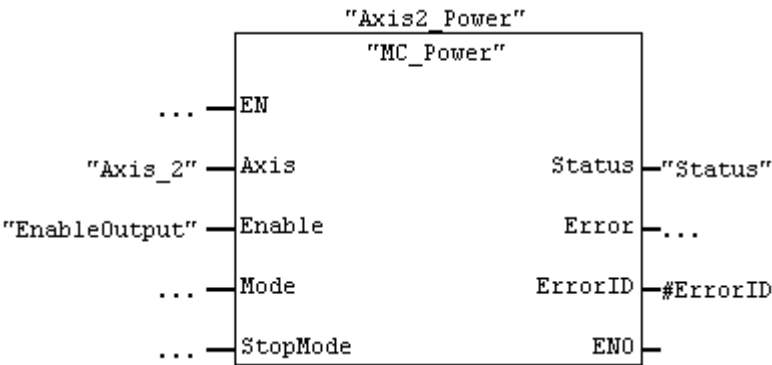
工艺功能“MC_Power”是否启用该驱动器，取决于输入 *Enable* 的逻辑状态。您可用以仅在控制信号的上升沿启用驱动器的逻辑运算有：

网络 1 — 控制沿计算的逻辑

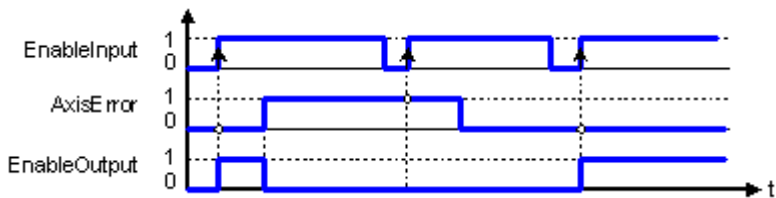
<i>EnableInput</i>	控制轴启用信号的输入
<i>AxisError</i>	0 没有错误 1 轴上的错误激活
<i>EnableOutput</i>	输出信号（到“MC_Power”；输入 <i>Enable</i> ）



网络 2 — 轴启用（用“MC_Power”）



相应的信号图



MC_Power — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致
8005	命令已取消，因为命令存储器正由其它过程使用。	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 下列工艺功能上的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一个周期内调用这些工艺功能，直到输出参数 <i>Done</i>, <i>CommandAborted</i> 或 <i>Error</i> 中有一个为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8043	参数值非法	涉及数据类型为 <i>REAL</i> 的所有输入参数，以及输入参数 <i>Mode</i> 或 <i>StopMode</i>。 示例： <i>Mode</i> 或 <i>Stopmode</i> 中输入的值无效，例如， <i>Mode = 129</i>。
8044	命令不受工艺对象支持。	例如，在输出凸轮中使用"MC_Power"。
8045	在当前状态下不允许使用该命令	可能的原因： 在 <i>S7T Config</i> 中或通过使用"MC_ActivateTO"工艺功能取消激活轴。
804C	命令输出率过高。	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是可能无法通过状态输出对其进行监视。

ErrorID	错误消息	说明/纠正或避免错误的方法
804E	每个轴仅允许一个实例	每个轴可能只能在一个实例上激活工艺功能。 另一个“MC_Power”(Busy = TRUE) 命令已在指定的轴上激活。 只使用一个实例，或中止活动命令。
804F	不允许更改“MC_Power”中的轴。	在输入参数 <i>Axis</i> 中设置的轴与在启用参数中设置的轴不同时，将禁用此轴 (Enable 1->0)。您只能禁用实际是由“MC_Power”的此实例启用的轴。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8063	未插补凸轮	未插补液压轴的阀控曲线。
8083	DB 不是工艺 DB	通过输入参数 <i>Axis</i> 指定的 DB 未找到或不是工艺 DB。

ErrorID	错误消息	说明/纠正或避免错误的方法
8084	工艺 DB 无效	- 在通过输入参数 <i>Axis</i> 定义的工艺 DB 的控制器中不存在某个工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Axis</i> 更改 DB 号。 - 用户在输入参数 <i>Axis</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management)中删除工艺 DB，然后创建一个新的工艺 DB。
808B	参数值的 REAL 格式无效	数据类型为 REAL 的输入参数的值不符合有效的浮点数格式。 检查输入参数值或背景数据块数据。不能以浮点格式表示格式无效的值。它们以十六进制格式 (DW16# ...) 显示。

6.2.2 FB403 MC_Home — 使轴回原点/设置轴

使用 FB403“MC_Home”来使轴回原点/设置轴

用途

- “MC_Home”工艺功能通过测量系统在控制和机械系统之间建立了基于位置的相关关系。
- 轴的主动回原点和被动回原点
回原点模式总是在工艺功能中进行定义。必须在 S7T Config 中设置回原点模式和方向参数
- 设置位置值
分配的位置值是绝对值，这允许绝对运动和相对运动。
- 相对于实际值的校正/偏移量
- 在基坐标系和叠加坐标系中校正设定点。仅支持相对偏移量
- 绝对编码器调整。

适用对象

- 定位轴
- 同步轴

说明

虚拟轴仅支持 *Mode* = 3、4、6 和 7。

具有增量编码器的轴仅支持 *Mode* = 0、1、2、3、4、6 和 7。

具有绝对编码器的轴仅支持 *Mode* = 3、4、5、6 和 7。

如果在轴的回原点组态中选择了“无模式”(no mode)，则您只能设置 *Mode* = 3、4、6 和 7。

在 *Mode* = 4、6 和 7 时，轴状态保持不变（已回原点/未回原点）。

仅当轴不是在“速度控制模式”（“MC_MoveVelocity” — *PositionControl* = TRUE）下运行时，才能使用 *Mode* = 2、3 和 5。

先决条件

- 在 *Mode* = 0、1 和 2 时，必须启用该轴以进行位置控制操作。
- 在 *Mode* = 0 和 1 下操作轴时，不能有 MC_Stop 命令处于活动状态。
- 驱动器接口 IM 174/ADI4 不支持同步执行 MC_MeasuringInput 和 MC_Home 命令。

命令的相互作用

新命令 — 活动的单个命令 (2)

新命令 — 活动的命令 (2)

输入参数

参数	数据类型	初始值	说明
<i>Axis</i>	INT	0	工艺 DB 的编号
<i>Execute</i>	BOOL	FALSE	在上升沿处启动命令
<i>Position</i>	REAL	0.0	到达原点位置时的绝对位置或者位置设定值或位置校正值。
<i>Mode</i>	INT	0	回原点模式：
			值 = 0: 主动回原点： 按照轴组态进行参考点逼近
			值 = 1: 主动回原点： 按照轴组态进行参考点逼近 输入参数 <i>Position</i> 的值被用作参考点坐标。
			值 = 2: 被动回原点： 根据 S7T Config 中的组态回原点。输入参数 <i>Position</i> 的值被用作参考点坐标。
			值 = 3: 直接回原点： 为当前位置分配了输入参数 <i>Position</i> 的值。
			值 = 4: 校正实际位置： (新的实际位置 = 当前实际位置 - <i>Position</i> 参数)。
			值 = 5: 绝对编码器调整： 为当前位置分配了输入参数 <i>Position</i> 的值。 如果出现电源故障，则根据绝对编码器调整得出的位移将保留。该轴必须处于静止状态，以便进行精确调整。绝对编码器调整后将保留在 <i>Mode</i> = 3、4、6 或 7 下执行的所有位置校正。 在绝对编码器调整期间，必须在位置控制模式下操作该轴。有关详细信息，请参考“MC_MoveVelocity”的输入参数 <i>PositionControl</i> 的说明。

参数	数据类型	初始值	说明
			值 = 6: 基坐标系中的设定值校正: 任何活动的叠加运动都基于校正后的位置设定值。(新位置设定值 = 当前基点位置 - 输入参数 <i>Position</i>)
			值 = 7: 叠加坐标系中的设定值校正: (新位置设定值 = 当前叠加位置 - <i>Position</i> 输入参数) 请注意,除了绝对编码器调整之外,还要激活所有位置校正 (<i>Mode</i> = 3、4、6 或 7)。在断电或重启 (“MC_Reset”, <i>Restart</i> = <i>TRUE</i>) 之后,只激活绝对编码器调整。
			值 = 8 取消被动回原点 取消已启动的被动回原点命令 (<i>Mode</i> = 2)。
			没有启用的轴时,也可以在 <i>Mode</i> = 2 到 7 中执行 “MC_Home”。如果使用绝对编码器操作轴,则工艺 DB 的 <i>Statusword.EncoderValid</i> 参数中的值在 MC_Home 命令启动之前必须为 <i>TRUE</i> (在轴重启后延迟设置该信号)。
<i>DoneFlag</i>	INT	0	在 MCDevice DB 中生成 DoneFlag

Mode = 7 的说明。

可以通过叠加运动或回原点命令 (“MC_Home” *Mode* = 7) 来修改叠加坐标系。叠加坐标系的坐标根据组态参数 *TypeOfAxis.DecodingConfig.transferSuperimposed* 的设置传送到基本坐标系。叠加坐标系会在此操作中复位。

设置此组态参数可以确定应在何时在基坐标系中激活叠加坐标系的坐标。

组态参数 *TypeOfAxis.DecodingConfig.transferSuperimposed* 的设置：

- *TRANSFER_STANDSTILL* (默认) :
 - 在轴处于静止状态后激活 — 轴必须去同步
 - 在轴切换为跟随模式时激活
(“MC_Power” *Mode* = 2 或 3)
 - 在轴重启动时激活
(*Restart* = *TRUE* 时“复位”)。
 - 在命令输出以超驰基本运动时激活
- *TRANSFER_MERGE*:
 - 在轴切换为跟随模式时激活
(“MC_Power” *Mode* = 2 或 3)
 - 在轴重启动时激活
(*Restart* = *TRUE* 时“复位”)。
 - 在命令输出以超驰基本运动时激活
- *TRANSFER_RESET*:
 - 在轴切换为跟随模式时激活
(“MC_Power” *Mode* = 2 或 3)
 - 在轴重启动时激活
(*Restart* = *TRUE* 时“复位”)。

注意
注意： 在 <i>Mode</i> = 7 下，各模块的超驰响应将有所不同。叠加运动不能再由基本运动取消（例外）。 <i>Mode</i> = 7 总是在基坐标系中执行，以便在跟随模式中操作轴。

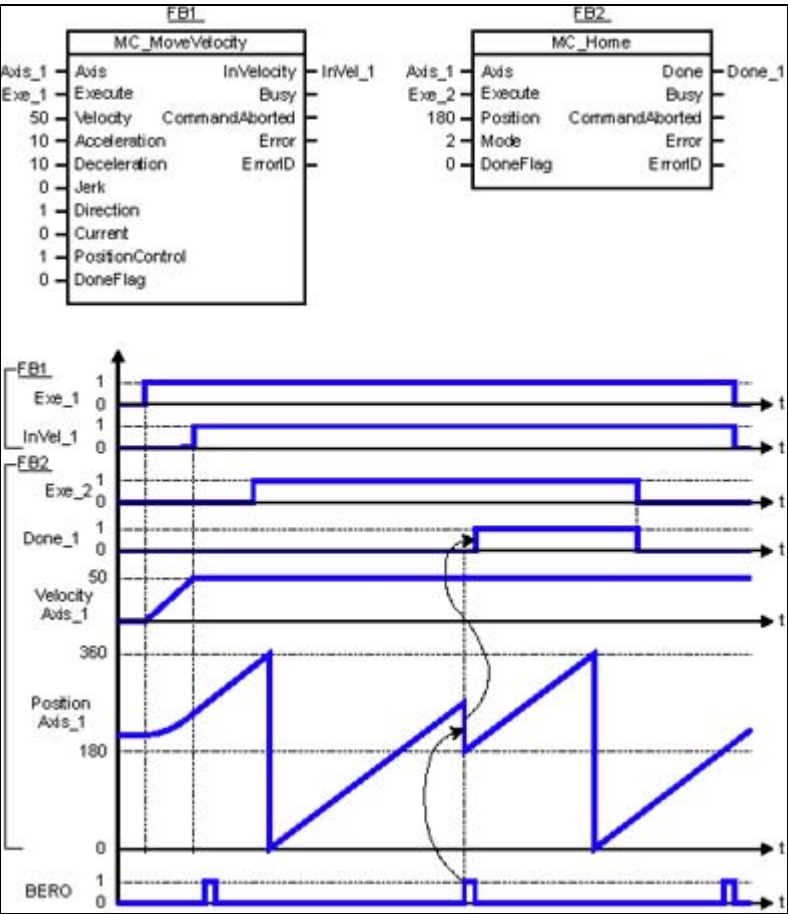
输出参数（状态输出）

参数	数据类型	初始值	说明	
<i>Done</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令已完成	
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令	
<i>CommandAborted</i>	BOOL	<i>FALSE</i>	<i>TRUE</i>: 命令由其它命令或因其执行过程中的错误而取消。 如果工艺 DB 的 <i>ErrorStatus</i> 中未显示错误, 则该命令由后续命令取消。 如果工艺 DB 的变量 <i>ErrorStatus</i> 中指示有错误, 则命令执行期间影响该工艺对象的错误将导致该命令中止。	
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> :	命令启动时 有错误 。未执行该命令。有关原因的信息, 请参考 <i>ErrorID</i> 。
			<i>FALSE</i> :	命令启动时 没有错误 。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。	

MC_Home — 示例 — “被动回原点”

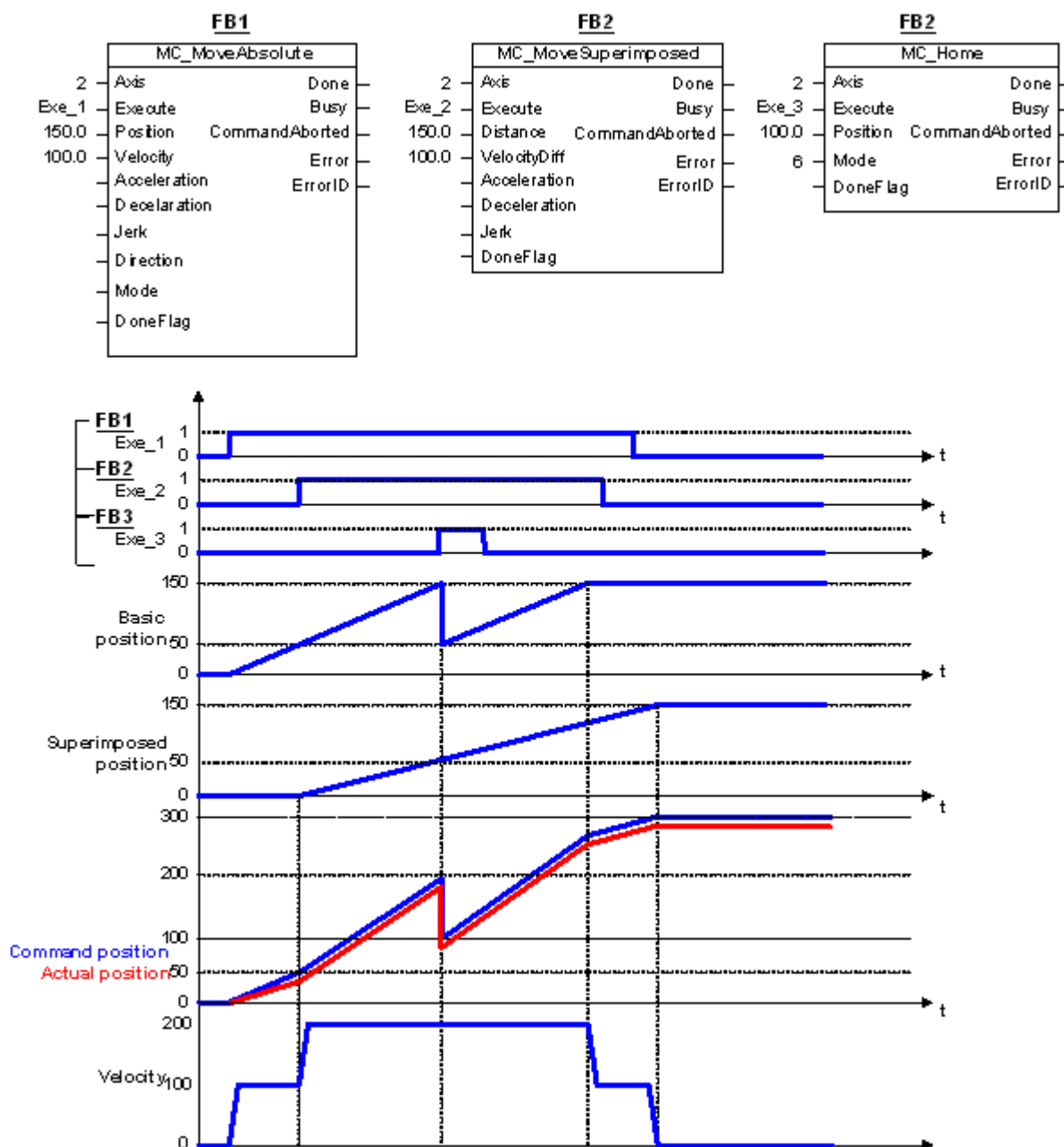
该示例说明了被动（实时）回原点的响应。

“Axis_1”通过调用“MC_MoveVelocity”开始。被动回原点已启用（“MC_Home”； *Mode* = 2）。在下一个同步事件中，（在本实例中： 外部零标记的边沿处），实际轴位置设置为 180°。



MC_Home — “模式 6”示例

位置设定值校正仅影响基坐标系，也就是说此更改不会影响叠加同步。本示例说明了“MC_MoveAbsolute”启动以及后续“MC_MoveSuperImposed”短暂延迟的启动。在轴运动时，对基坐标系中的位置用 *100* 进行校正。

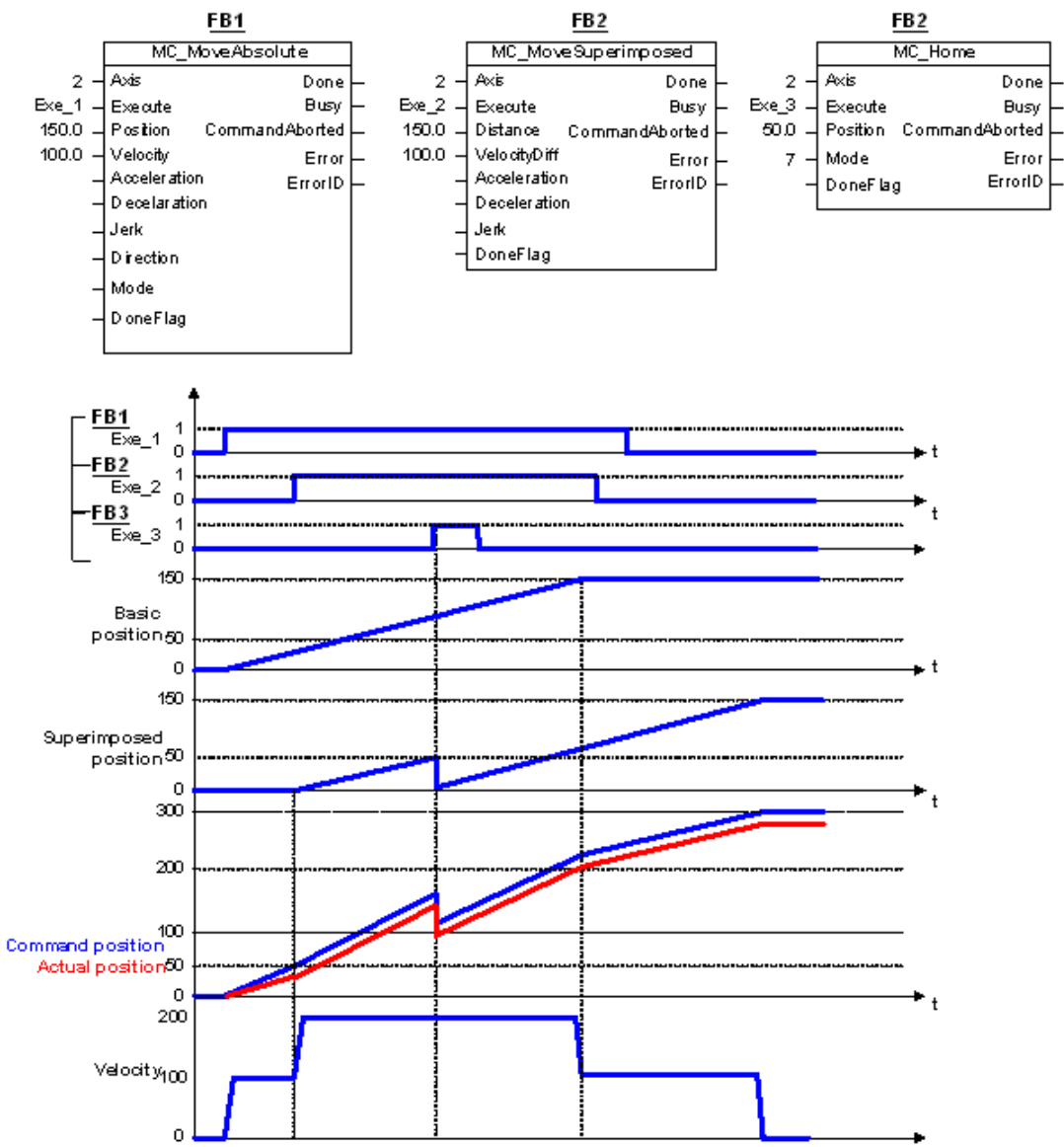


MC_Home —“模式 7”示例

默认情况下，在轴停止后，由叠加命令引起的位置更改会从叠加坐标系转换到基坐标系。
可以在轴组态参数 *TypeOfAxis.DecodingConfig.transferSuperimposed* 中组态此响应。
在接下来的示例中设置的是 *Transfer_Reset* 而不是默认设置。因此仅当取消轴使能信号后，叠加坐标系才会转换到基坐标系。

本示例说明了“MC_MoveAbsolute”启动以及后续“MC_MoveSuperImposed”短暂延迟的启动。将对叠加定位用 50 进行校正。

模式 6 或 7 校正用于调整相关坐标系的位置设定值。实际位置的跟踪建立在实际跟随误差计算的基础之上。



MC_Home — ErrorID

适用于固件版本为 V4.1.x 的集成工艺

本部分将介绍 V3.2.x 及更低版本固件的应用。

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8022	轴/外部编码器中没有实际值	例如，编码器或数据总线未准备就绪
8040	轴/外部编码器被禁用，或者设置了错误的模式	缺少运动命令所需的使能信号。 消除并确认所有排队的错误，然后启用相关的轴模式（例如，位置控制）。
8043	参数值非法	涉及数据类型为 <i>REAL</i> 的所有输入参数，或者输入参数 <i>Mode</i> 或 <i>DoneFlag</i> 。
8044	命令不受工艺对象支持	向速度控制轴或输出凸轮发送回原点命令。

ErrorID	错误消息	说明/纠正或避免错误的方法
8045	在当前状态下不允许使用该命令	可能的原因： • MC_Stop 命令处于活动状态 • 在绝对编码器调整期间，该轴不在位置控制模式下操作。有关详细信息，请参考“MC_MoveVelocity”的输入参数 <i>PositionControl</i> 的说明。 • 请求主动回原点 (<i>Mode = 0、1</i>) 且 MC_MoveToEndPos 命令处于活动状态。 • 选定的轴处于速度控制模式 (“MC_MoveVelocity”，<i>PositionControl = FALSE</i>) • 电源被禁用，同时伺服互锁（例如，由于急停）。 • 工艺对象的参数已改变。对参数所做的此更改需要重新启动，但是尚未执行重新启动。 • 在工艺对象的专家列表中，“集中更改”(Collect changes) 按钮处于活动状态（系统变量 <i>activationmodechangedconfigdata</i> 的值为 <i>collect_changed_config_data</i>）。 • 尽管没有被动回原点命令处于活动状态，但还是启动了用于取消被动回原点的命令 (<i>Mode = 8</i>)；或者被动回原点命令已经完成。
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是可能无法通过状态输出对其进行监视。
804D	轴类型无效	这是一个虚拟轴。虚拟轴不支持模式 <i>0、1、2</i> 和 <i>5</i> 。
8050	工艺未准备就绪	• 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 • 在重启 OB 中输出了该命令。

ErrorID	错误消息	说明/纠正或避免错误的方法
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效。	工艺功能的背景数据块存在故障（例如，长度错误）。
8083	DB 不是工艺 DB	通过输入参数 <i>Axis</i> 指定的 DB 未找到或不是工艺 DB。
8084	工艺 DB 无效	- 在通过输入参数 <i>Axis</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Axis</i> 更改 DB 号。 - 用户在输入参数 <i>Axis</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management) 中删除工艺 DB，然后创建一个新的工艺 DB。
808B	参数值的 REAL 格式无效	数据类型为 REAL 的输入参数的值不符合有效的浮点数格式。 检查输入参数值或背景数据块数据。 不能以浮点格式表示格式无效的值。 它们以十六进制格式 (DW16# ...) 显示。

6.2.3 FB404 MC_Stop — 停止轴并阻止新的运动命令

用 FB404“MC_Stop”来停止轴

适用于固件版本为 V3.2.x 或更高版本的集成工艺

本部分介绍了 V3.1.x 及更低版本固件的应用

用途

- “MC_Stop”工艺功能将停止所有运动，并使轴停止。未定义停止位置。
- 在轴已停止并且已设置 *Execute = FALSE* 之后，结束命令 (*Done = TRUE*)。即，只要 *Execute = TRUE*，该轴就无法启动。这也适用于在轴被禁用然后再使用工艺功能“MC_Power”重新启用的情况。
- 在输入参数 *Jerk* 和 *Deceleration* 中定义轴对停止命令的动态响应。

适用对象

- 速度控制轴
- 定位轴
- 同步轴

先决条件

动态停止需要启用该轴以进行位置控制操作/速度控制操作。如果该轴在下列模式下运行或者处于禁用状态，则其无法动态逐渐停止！

命令的相互作用

新命令 — 活动的单个命令 (2)

新命令 — 活动的命令 (2)

说明

在“MC_STOP”命令处于活动状态时，您不应通过“MC_Power”禁用该轴。但是如果确实发生这种情况，则“MC_Power”的 STOP 条件会取消您在“MC_Stop”中编程的 STOP 运动。在“MC_Stop”中的 *Execute = TRUE* 时，无论是否使用“MC_Power”激活，动态轴运动均会被锁定。

输入参数

参数	数据类型	初始值	说明
<i>Axis</i>	INT	0	工艺 DB 的编号
<i>Execute</i>	BOOL	FALSE	在上升沿处启动命令
<i>Deceleration</i>	REAL	-1.0	减速（降低电机功率）：
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值
<i>Jerk</i>	REAL	-1.0	冲击：
			值 > 0: 使用定义的值
			值 = 0: 使用梯形运动曲线
			值 < 0: 使用默认值
<i>DoneFlag</i>	INT	0	在 MCDevice DB 中生成 DoneFlag

如果设置的值 *Deceleration* > +1E+12，则轴将以最大减速度停止。

如果设置的值 *Jerk* > +1E+12，则轴将以最大冲击停止。

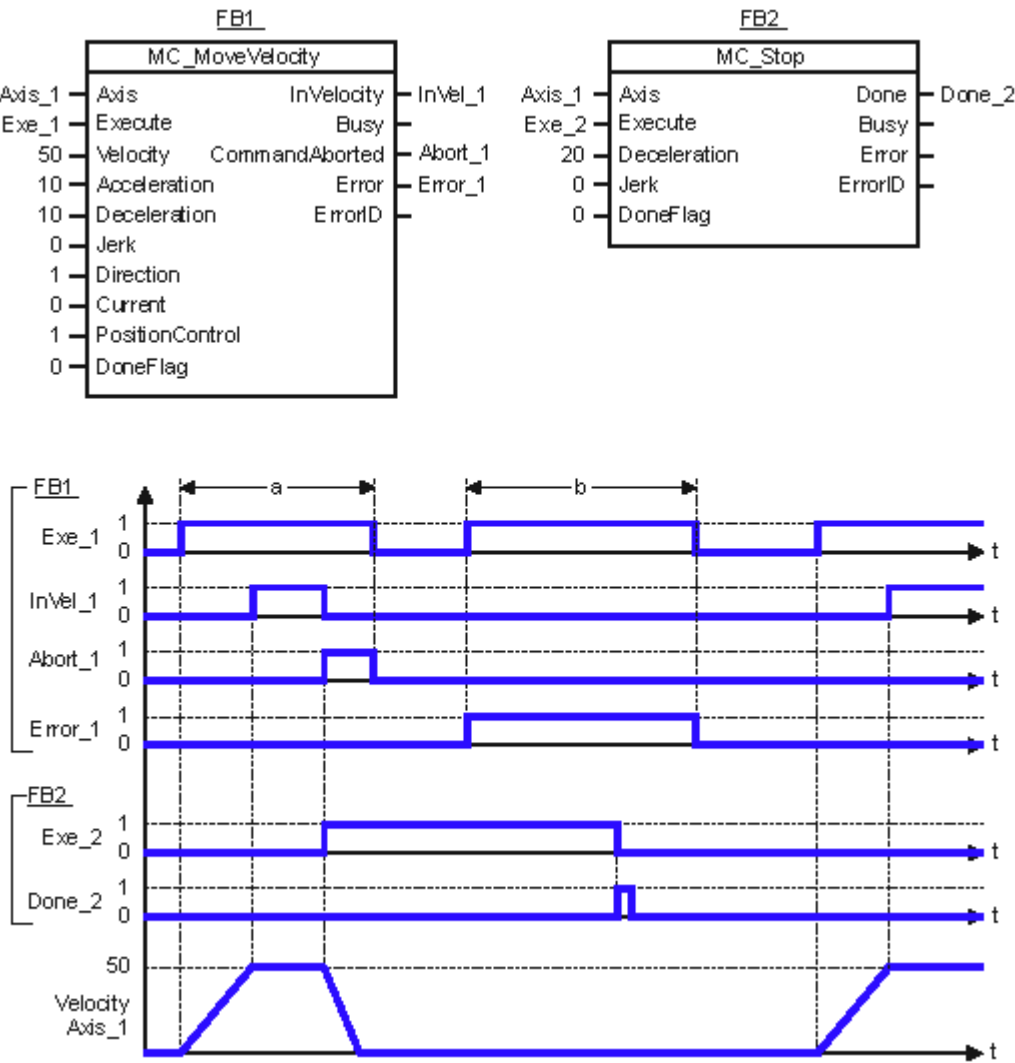
 警告
如果在输入参数 <i>Deceleration</i> 或 <i>Jerk</i> 中设置了不能表示为 REAL 值的数值，则“MC_Stop”将不执行。

输出参数（状态输出）

参数	数据类型	初始值	说明
<i>Done</i>	BOOL	FALSE	TRUE: 速度达到零，且输入 <i>Execute</i> = FALSE（针对一个周期的时间间隔的设置）
<i>Busy</i>	BOOL	FALSE	TRUE: 正在执行命令
<i>CommandAborted</i>	BOOL	FALSE	TRUE: 该命令由其它 MC_Stop 命令取消。
<i>Error</i>	BOOL	FALSE	TRUE: 命令启动时有错误。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
			FALSE: 命令启动时没有错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 ErrorID。

MC_Stop — 示例

- a) 通过调用工艺功能“MC_Stop”逐渐停止旋转轴。
- b) 只要在“MC_Stop”工艺功能中设置了参数 *Execute = TRUE*，该轴便拒绝运动命令。“MC_MoveVelocity”工艺功能报告了一个错误，以表明活动的 MC_Stop 命令。



MC_Stop — ErrorID

适用于固件版本为 V3.2.x 的集成工艺

本部分将介绍 V3.1.x 及更低版本固件的应用。

ErrorID	警告消息	说明/纠正或避免错误的方法
0000	没有错误	-
0021	动态响应值受限制	命令（ <i>Deceleration</i> 或 <i>Jerk</i> ）的动态值受限制，因为它们超出组态的限制
0028	忽略非法的参数值	无效的输入参数值。该值被忽略，取而代之的是默认值。检查输入参数值，并纠正无效的值。

ErrorID	错误消息	说明/纠正或避免错误的方法
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8043	参数值非法	涉及数据类型为 REAL 的所有输入参数，或者输入参数 <i>DoneFlag</i> 。
8044	命令不受工艺对象支持	"例如，向输出凸轮发送"MC_Stop"命令。

ErrorID	错误消息	说明/纠正或避免错误的方法
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是可能无法通过状态输出对其进行监视。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8083	DB 不是工艺 DB	通过输入参数 <i>Axis</i> 指定的 DB 未找到或不是工艺 DB。

ErrorID	错误消息	说明/纠正或避免错误的方法
8084	工艺 DB 无效	- 在通过输入参数 <i>Axis</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Axis</i> 更改 DB 号。 - 用户在输入参数 <i>Axis</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management)中删除工艺 DB，然后创建一个新的工艺 DB。
808B	参数值的 REAL 格式无效	数据类型为 REAL 的输入参数的值不符合有效的浮点数格式。 检查输入参数值或背景数据块数据。 不能以浮点格式表示格式无效的值。它们以十六进制格式 (DW16# ...) 显示。

6.2.4 FB405 MC_Halt — 正常停止

使用 FB 405“MC_Halt”进行正常停止

用途

- “MC_Halt”工艺功能将停止所有运动，并使轴停止。未定义停止位置。
- 在输入参数 *Jerk* 和 *Deceleration* 中定义运动轴的动态响应。
- 该命令在轴停止后结束，或者在被新运动命令取消后结束。

适用对象

- 速度控制轴
- 定位轴
- 同步轴

先决条件

- 启用轴进行速度控制操作或位置控制操作
- 没有处于活动状态的 MC_Stop 命令

命令的相互作用

新命令 — 活动的单个命令 (2)

新命令 — 活动的命令 (2)

输入参数

参数	数据类型	初始值	说明
<i>Axis</i>	INT	0	工艺 DB 的编号
<i>Execute</i>	BOOL	FALSE	在上升沿处启动命令
<i>Deceleration</i>	REAL	-1.0	减速（降低电机功率）：
			值 > 0: 使用定义的值
			值 = 0: 不允许（轴在停止状态时除外）
			值 < 0: 使用默认值
<i>Jerk</i>	REAL	-1.0	冲击：
			值 > 0: 使用定义的值
			值 = 0: 使用梯形运动曲线
			值 < 0: 使用默认值
<i>DoneFlag</i>	INT	0	在 MCDevice DB 中生成 DoneFlag

输出参数（状态输出）

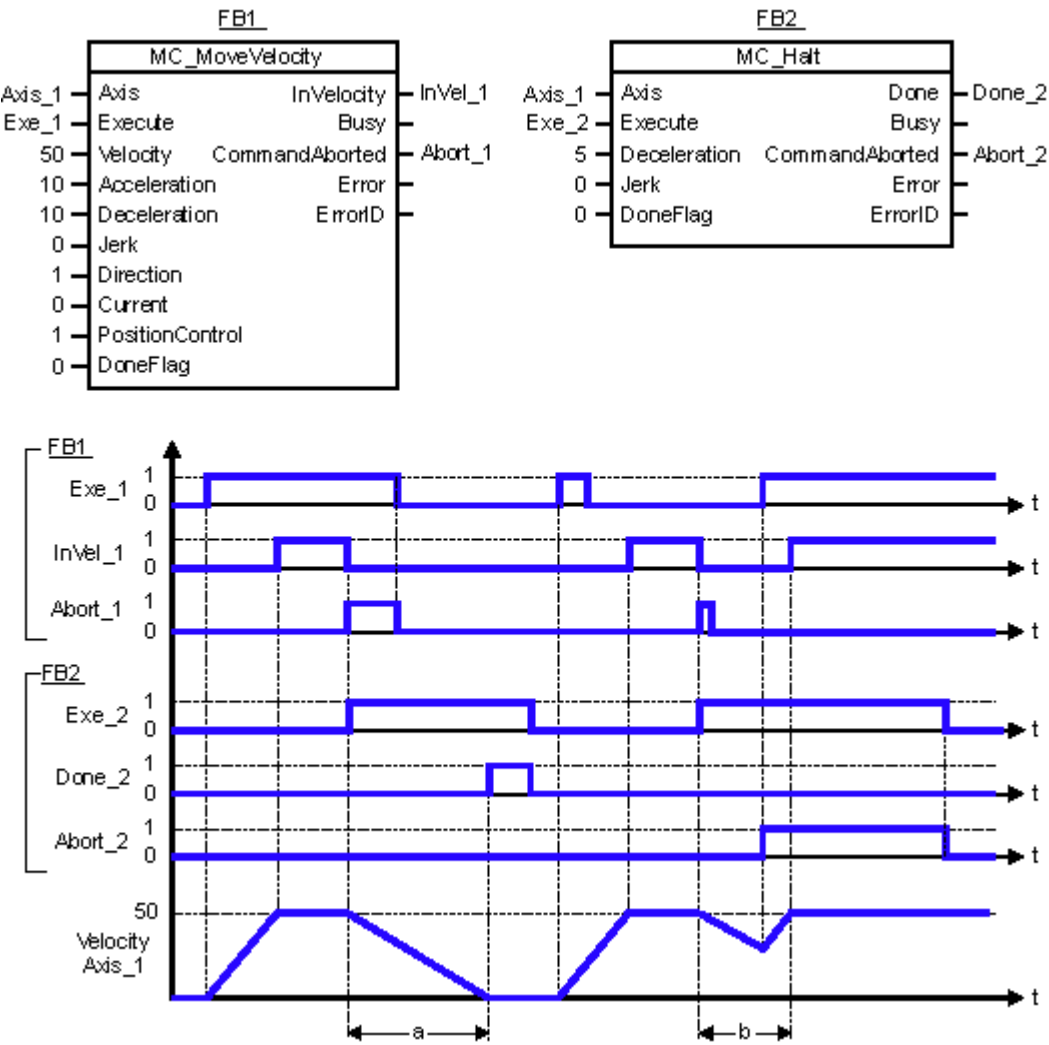
参数	数据类型	初始值	说明
<i>Done</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 速度已达到零 如果一个新叠加命令在 <i>MC_Halt</i> 命令执行期间启动，则即使叠加运动仍处于活动状态，活动的 <i>MC_Halt</i> 命令也可以通过 <i>Done = TRUE</i> 终止。 在『新命令 — 活动的单个命令 (2)』中，可以获取更多信息。
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令
<i>CommandAborted</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令由其它命令或因其执行过程中的错误而取消。 如果工艺 DB 的 <i>ErrorStatus</i> 中未显示错误，则该命令由后续命令取消。 如果工艺 DB 的变量 <i>ErrorStatus</i> 中指示有错误，则命令执行期间影响该工艺对象的错误将导致该命令中止。
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令启动时有错误。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
			<i>FALSE</i> : 命令启动时没有错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。

说明

请注意，T-CPU 可以减小任何活动加速度，具体取决于动态参数设置 (*Jerk*)。因此，在启动 *MC_Halt* 命令之后，轴速度仍可能会增加。

MC_Halt — 示例

- a) 通过调用工艺功能“MC_Halt”暂停旋转轴。
- b) 另一个运动命令超驰了活动的 MC_Halt 命令。与“MC_Stop”相反，“MC_Halt”命令支持此动作。



MC_Halt — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8040	轴/外部编码器被禁用，或者设置了错误的模式	缺少运动命令所需的使能信号。 消除并确认所有排队的错误，然后启用相关的轴模式（例如，位置控制）。
8043	参数值非法	涉及数据类型为 <i>REAL</i> 的所有输入参数，或者输入参数 <i>DoneFlag</i> 。
8044	命令不受工艺对象支持	例如，向外部编码器发送"MC_Halt"。
8045	在当前状态下不允许使用该命令	MC_Stop 命令处于活动状态。
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是可能无法通过状态输出对其进行监视。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 <i>HOLD</i> 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 <i>OB</i> 中输出了该命令。

ErrorID	错误消息	说明/纠正或避免错误的方法
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。工艺功能一开始在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： • 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 • 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8056	由于正在向固定挡块运动而取消	轴已运动到固定挡块 (<i>InClamping = TRUE</i>)。只有新命令从固定挡块释放轴，它们才被接受。 已启动的命令不符合该条件。
8083	DB 不是工艺 DB	通过输入参数 <i>Axis</i> 指定的 DB 未找到或不是工艺 DB。
8084	工艺 DB 无效	• 在通过输入参数 <i>Axis</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Axis</i> 更改 DB 号。 • 用户在输入参数 <i>Axis</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management)中删除工艺 DB，然后创建一个新的工艺 DB。
808B	参数值的 REAL 格式无效	数据类型为 REAL 的输入参数的值不符合有效的浮点数格式。 检查输入参数值或背景数据块数据。不能以浮点格式表示格式无效的值。它们以十六进制格式 (DW16# ...) 显示。

6.2.5 FB409 MC_ChangeDataset — 数据记录转换

使用 FB409“MC_ChangeDataset”进行数据记录转换

用途

- 该工艺功能可用于在轴的数据记录之间进行转换。
- 有几个数据集可用于进行下列操作：
 - 在系统运行时转换控制器数据。
 - 在系统运行时转换使用的编码器（电机编码器、机器编码器等）。

适用对象

- 速度控制轴
- 定位轴
- 同步轴

先决条件

- 相应的数据记录必须已在相关工艺对象中组态。
- 编码器必须与工艺对象互连，才可以进行转换。编码器必须在转换时返回有效值。您可以通过使用参数号 *4050* 调用“MC_ReadSysParameter”工艺功能来验证这一点。

命令的相互作用

新命令 — 活动的单个命令 (3)

输入参数

参数	数据类型	初始值	说明
<i>Axis</i>	INT	0	轴工艺 DB 的编号
<i>Execute</i>	BOOL	FALSE	在上升沿处启动命令
<i>SyncEncoder</i>	BOOL	FALSE	FALSE: 不进行编码器同步
			TRUE: 进行编码器同步
			仅当在数据记录中定义了不同的编码器时，输入参数 <i>SyncEncoder</i> 才会被激活。
<i>Dataset</i>	INT	1	要启用的数据记录的编号
<i>DoneFlag</i>	INT	0	在 MCDevice DB 中生成 DoneFlag

输出参数（状态输出）

参数	数据类型	初始值	说明
<i>Done</i>	BOOL	FALSE	TRUE: 数据记录已更改 如果定义的数据记录与活动的数据记录没有区别，则工艺功能会立即返回 <i>Done</i> = TRUE。
<i>Busy</i>	BOOL	FALSE	TRUE: 正在执行命令
<i>CommandAborted</i>	BOOL	FALSE	TRUE: 命令由其它命令或因其执行过程中的错误而取消。 如果工艺 DB 的 <i>ErrorStatus</i> 中未显示错误，则该命令由后续命令取消。 如果工艺 DB 的变量 <i>ErrorStatus</i> 中指示有错误，则命令执行期间影响该工艺对象的错误将导致该命令中止。
<i>Error</i>	BOOL	FALSE	TRUE: 命令启动时有错误。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
			FALSE: 命令启动时没有错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 ErrorID。

数据记录转换的条件

数据记录转换的时间由组态变量 *TypeOfAxis.NumberOfDataSets.changeMode* 定义。

此变量可以具有以下值：

- **NEVER:**
不允许进行数据记录转换。“MC_ChangeDataset”命令由于错误 (*ErrorID = 8045*) 已终止。
- **IN_POSITION:** （在插入轴后，S7T Config 中的默认设置）
在活动运动到达已编程的定位窗口后，数据记录发生更改。必须启用该轴以进行位置控制操作。数据记录在轴重新启用后立即更改。
数据记录在同步跟随轴 (*InSync = TRUE* 或 *InGear = TRUE*) 中不更改。为跟随轴设置 *IN_STANDSTILL* 或 *IMMEDIATELY* 值。
- **IN_STANDSTILL:**
数据记录在相关轴返回停止状态后更改。数据记录在轴停止后立即更改
- **IMMEDIATELY:**
数据记录立即更改

您可以在 S7T Config 的专家列表中修改此响应。您也可以通过“MC_WriteParameter”工艺功能修改组态变量。

说明

根据对控制变量的生成有影响的参数，数据记录中的不同还可能导致在停止状态的轴发生步进响应。此状态对传动比参数或所有控制器参数有影响。

编码器同步

如果当前编码器与新数据记录的编码器不匹配，则根据输入参数 *SyncEncoder* 中的设置，使编码器同步。编码器同步会在新编码器中设置当前编码器的实际位置值。数据记录将在同步成功完成后进行转换。

两个编码器必须以相同的速度旋转，以确保精确同步。编码器间的滑动或其中一个编码器处于空闲状态都将妨碍编码器与活动进程的精确互连。编码器的同步将因此而失败，由于在编码器同步与数据记录转换期间编码器值之间出现的差异。

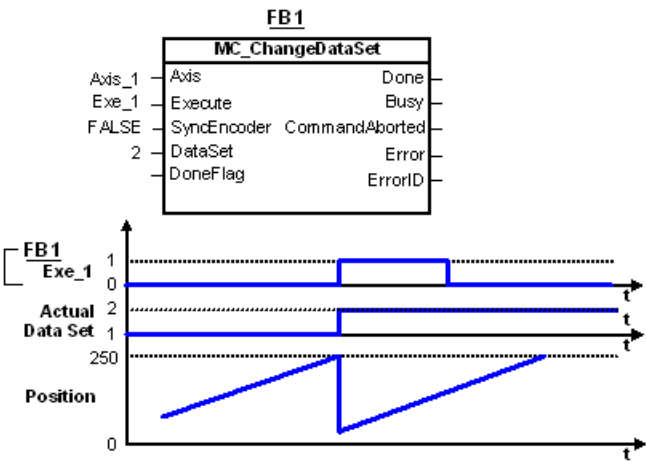
如果数据记录不能立即转换，则编码器同步将周期性重复。

说明

如果您将编码器连同数据记录一起转换，而不同步编码器，则由于编码器位置不同，轴可能会执行补偿运动。

MC_ChangeDataSet — 示例 — “编码器转换”

在下面的示例中，我们将在轴上转换数据记录。随后在编码器不同步的情况下，使用第二个测量系统的位置值。

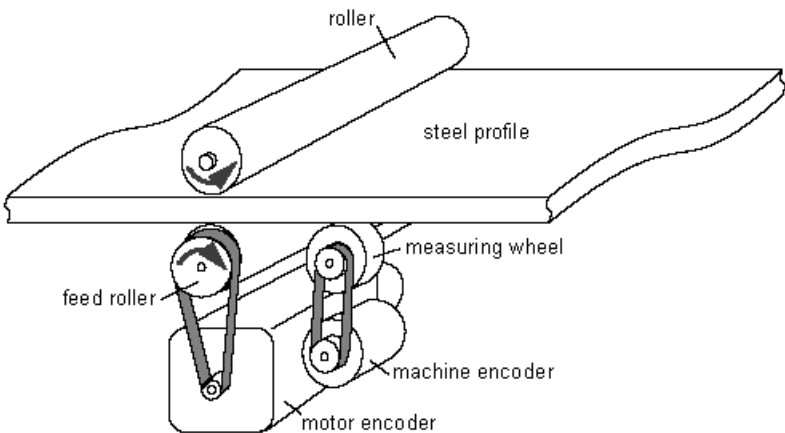


MC_ChangeDataSet — 示例 — “应用”

下图说明了作为工艺功能“MC_ChangeDataSet”典型应用的编码器转换。

要使用加工机械对连续带钢进行等长切割。额外集成了一个带有测量轮的机器编码器，以消除由进给辊滑动引起的所有测量误差。

在新带钢进给到机器中时，测量轮不会返回位置值，因此定位必须依靠电机编码器。当测量轮检测到带钢的位置后，会有一个传感器记录该位置。检测到传感器信号后，“MC_ChangeDataSet”便从电机编码器切换到机器编码器模式（此时系统正在编码器同步的情况下运行）。在编码器转换后，带钢可以精确进给，以在位置控制模式下进行进一步加工。



MC_ChangeDataset — ErrorID

受固件版本为 V3.1.x 或更高版本的集成工艺的支持

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8043	参数值非法	涉及输入参数 <i>Dataset</i> 或 <i>DoneFlag</i> 。
8044	命令不受工艺对象支持	例如，将命令发送至输出凸轮
8045	在当前状态下不允许使用该命令	示例： - 组态变量 <i>TypeOfAxis.NumberOfDataSets.changeMode</i> 的值为 <i>NEVER</i>。无法进行数据记录转换。 - 当前，工艺对象正在执行由"MC_Reset"启动的重新启动。 - 新数据记录的编码器未返回有效值。此错误可能在 STOP 转为 RUN、轴重新启动或硬件发生故障后立即出现。

ErrorID	错误消息	说明/纠正或避免错误的方法
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是可能无法通过状态输出对其进行监视。
804D	轴类型无效	这是一个虚拟轴。虚拟轴不支持此功能。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。

ErrorID	错误消息	说明/纠正或避免错误的方法
8083	DB 不是工艺 DB	通过输入参数 <i>Axis</i> 指定的 DB 未找到或不是工艺 DB。
8084	工艺 DB 无效	- 在通过输入参数 <i>Axis</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Axis</i> 更改 DB 号。 - 用户在输入参数 <i>Axis</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management)中删除工艺 DB，然后创建一个新的工艺 DB。

6.2.6 FB410 MC_MoveAbsolute — 绝对定位

使用 FB410“MC_MoveAbsolute”进行绝对定位

适用于固件版本为 V3.1.x 或更高版本的集成工艺

本部分介绍了固件版本为 V3.0.x 的应用

用途

- “MC_MoveAbsolute”工艺功能将开始轴到绝对位置的定位运动。
- 使用输入参数 *Velocity*、*Jerk*、*Acceleration* 和 *Deceleration* 来定义轴运动的动态响应。
- 该功能在到达目标位置时终止。
- 您可以预置模数轴的旋转方向。
- 在输入参数 *Mode* 中定义是否要触发活动的运动，或者定义是添加运动还是覆盖运动。

适用对象

- 定位轴
- 同步轴

先决条件

- 已启用轴进行位置控制操作
- 如果在组态中选择了“需要回原点”(Homing required)，则轴将回原点
- 没有处于活动状态的 MC_Stop 命令
- 适用于添加运动和叠加运动 (*Mode = 1、2*) 镀金：
 - “MC_CamIn”和“MC_GearIn”都未启动。
 - 没有处于活动状态的基本同步

命令的相互作用

新命令 — 活动的单个命令 (2)

新命令 — 活动的命令 (2)

输入参数

参数	数据类型	初始值	说明
<i>Axis</i>	INT	0	工艺 DB 的编号
<i>Execute</i>	BOOL	FALSE	在上升沿处启动命令
<i>Position</i>	REAL	0.0	目标位置（负值或正值）
<i>Velocity</i>	REAL	-1.0	最大速度（不是总能达到）：
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值
<i>Acceleration</i>	REAL	-1.0	加速（增加电机功率）：
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值
<i>Deceleration</i>	REAL	-1.0	减速（降低电机功率）：
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值

参数	数据类型	初始值	说明
<i>Jerk</i>	REAL	<i>-1.0</i>	冲击:
			值 > 0: 使用定义的值
			值 = 0: 使用梯形运动曲线
			值 < 0: 使用默认值
<i>Direction</i>	INT	<i>0</i>	预置模数轴的方向:
			值 = 0: S7T Config 中的默认设置 *
			值 = 1: 正向旋转
			值 = 2: 最短距离
			值 = 3: 反向旋转
			值 = 4: 当前旋转方向 (上次使用的旋转方向)
<i>Mode</i>	INT	<i>0</i>	超驰模式:
			值 = 0: 超驰运动: 当前运动被取消
			值 = 1: 添加运动: 运动命令被写入命令缓冲区。轴将在运动跳转时停止
			值 = 2: 覆盖运动: 运动命令被写入命令缓冲区。该运动跳转被覆盖。
			如果满足下列条件, 则“覆盖运动”超驰模式不适用于模数轴: 输入参数 <i>Direction</i> = 4 或者 输入参数 <i>Direction</i> = 0 并且 如果在 S7T Config 的轴 (Axis) > 动态响应 (Dynamic response) 对话框中设置了“方向 = 上次编程的目标方向”(Direction = last programmed target direction)。
<i>DoneFlag</i>	INT	<i>0</i>	在 MCDevice DB 中生成 DoneFlag
* 建议不要通过 S7T Config 默认值中的“基于速度设定值的符号来确定”设置来选择方向, 因为输入参数 <i>Velocity</i> 不支持负的速度设定值。			

输出参数（状态输出）

参数	数据类型	初始值	说明	
<i>Done</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 已到达目标位置	
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令	
<i>CommandAborted</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令由其它命令或因其执行过程中的错误而取消。 如果工艺 DB 的 <i>ErrorStatus</i> 中未显示错误，则该命令由后续命令取消。 如果工艺 DB 的变量 <i>ErrorStatus</i> 中指示有错误，则命令执行期间影响该工艺对象的错误将导致该命令中止。	
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> :	命令启动时有错误。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
			<i>FALSE</i> :	命令启动时没有错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。	

MC_MoveAbsolute — 示例 — “超驰运动”

以下信号图显示了 (*Mode* = 0) 工艺功能“MC_MoveAbsolute”的超驰特性。

阶段“a”

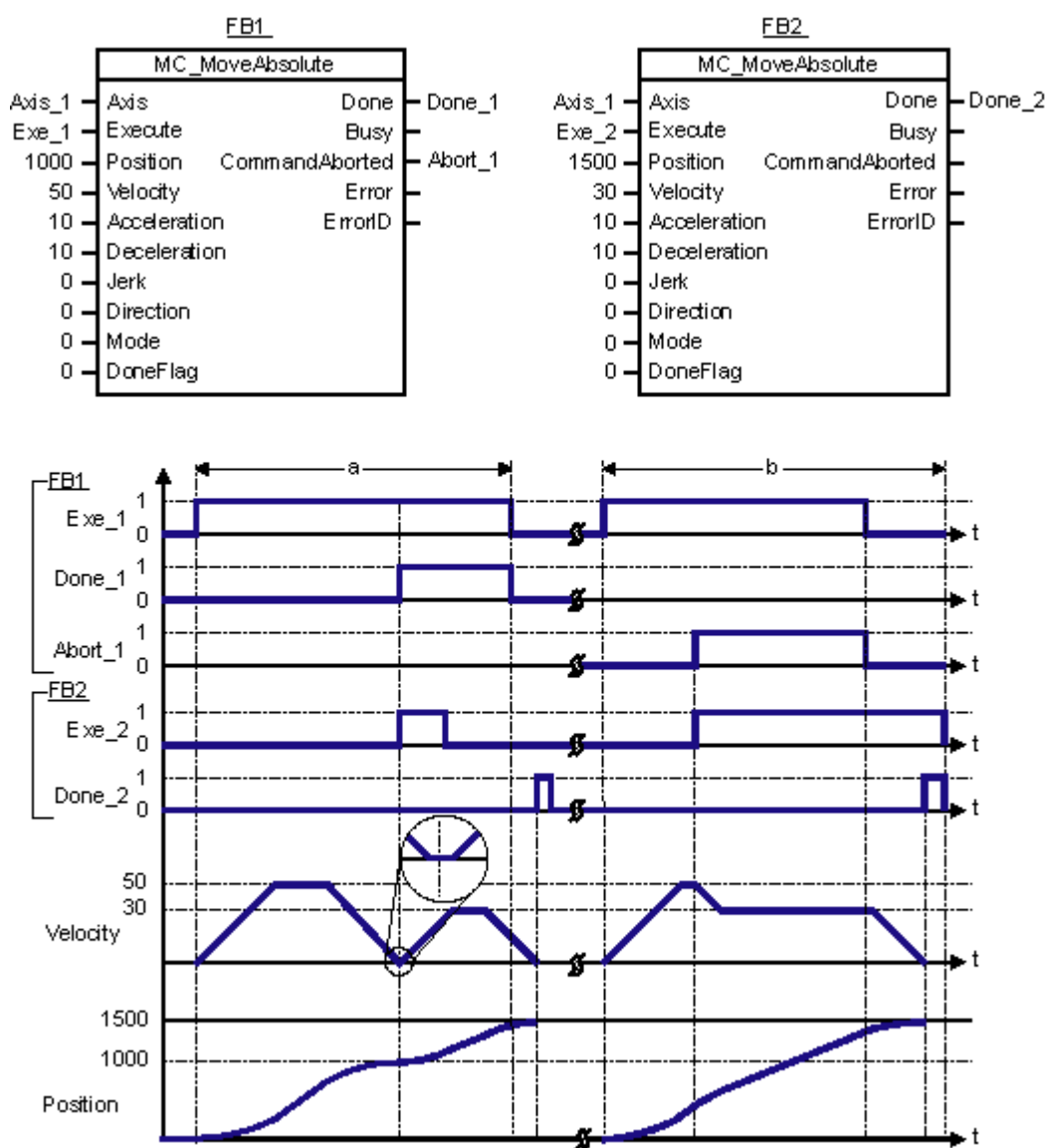
FB1 输入参数 *Execute* (*Exe_1*) 的第一个上升沿用于启动轴 (*Axis_1*) 的绝对定位。
Done_1 将发送信号，指示已到达绝对目标位置 1000。

在 *Done_1* = *TRUE* 的同时，FB2 上将启动另一个定位命令（输入参数 *Execute* (*Exe_2*)). ）。与命令执行有关的响应时间将导致轴短暂停止。请参见缩放视图。在此短暂停止后，轴 (*Axis_1*) 将移到绝对位置 1500。 *Done_2* 会报告该轴已到达目标位置

阶段“b”

FB1 输入参数 *Execute* (*Exe_1*) 的第二个上升沿用于启动轴 (*Axis_1*) 的绝对定位。

在轴到达目标位置 1000 之前，另一个定位命令在 FB2 中启动（输入参数 *Execute* (*Exe_2*)）。*Abort_1* 会报告 FB1 中活动定位命令的取消。轴 (*Axis_1*) 以定义的 *Deceleration* = 10 移动，直到速度 *Velocity* = 30。FB2 的 *Done_2* 会报告轴到达目标位置 1500。



MC_MoveAbsolute — 示例 — “添加运动”

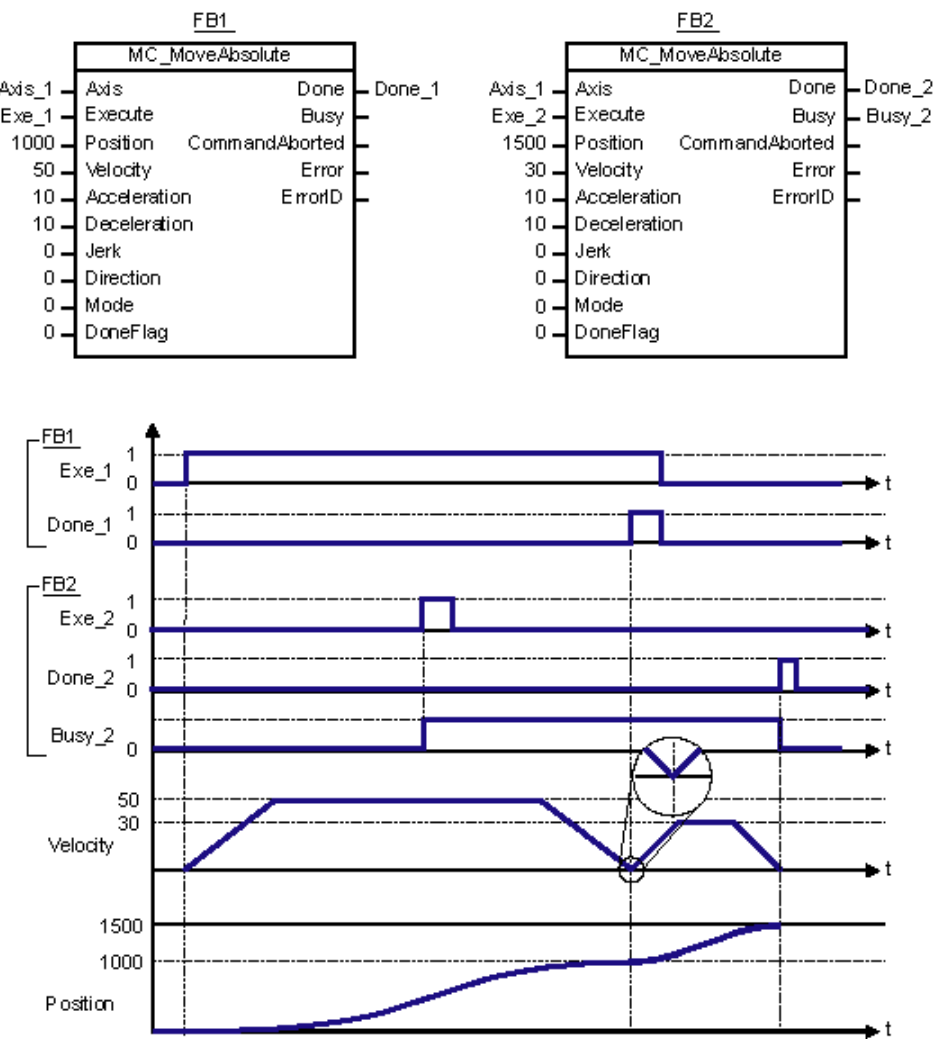
以下信号图显示了工艺功能“MC_MoveAbsolute”(Mode = 1) 的“添加”特性。

相对定位通过 FB1 的输入参数 *Execute (Exe_1)* 的上升沿启动。轴 (*Axis_1*) 将以定义的 *Acceleration = 10* 加速到其最终速度 *50*。在轴到达其目标位置 (*Position = 1000*) 之前，将输出另一个 MC_MoveAbsolute 命令 (FB2 的 *Exe_2* 上升沿)。 *Busy_2* 将报告新命令处于活动状态。

通过设置 FB2 的 *Mode = 1* 来“添加”当前定位命令而不是将其取消。

“MC_MoveAbsolute”将通过 *Deceleration = 10* 来为该轴减速，然后根据启动的命令到达目标位置 *1000*。通过 *Done_1* 报告命令完成。

在到达目标位置 *1000* 后，系统将添加第二个 MC_MoveAbsolute 命令，该命令将立即执行，而没有任何时间间隔 (请参阅缩放视图)。轴 (*Axis_1*) 将沿设置的加速斜坡加速到最终速度 (目标位置 = *1500*)。 *Done_2* 会报告该轴已到达目标位置； *Busy_2* 将更改为 *FALSE*。



MC_MoveAbsolute — 示例 —“覆盖运动 1”

以下显示的信号图显示了当前速度大于新速度的情况下工艺功能“MC_MoveAbsolute”(Mode = 2) 的“覆盖”特性。

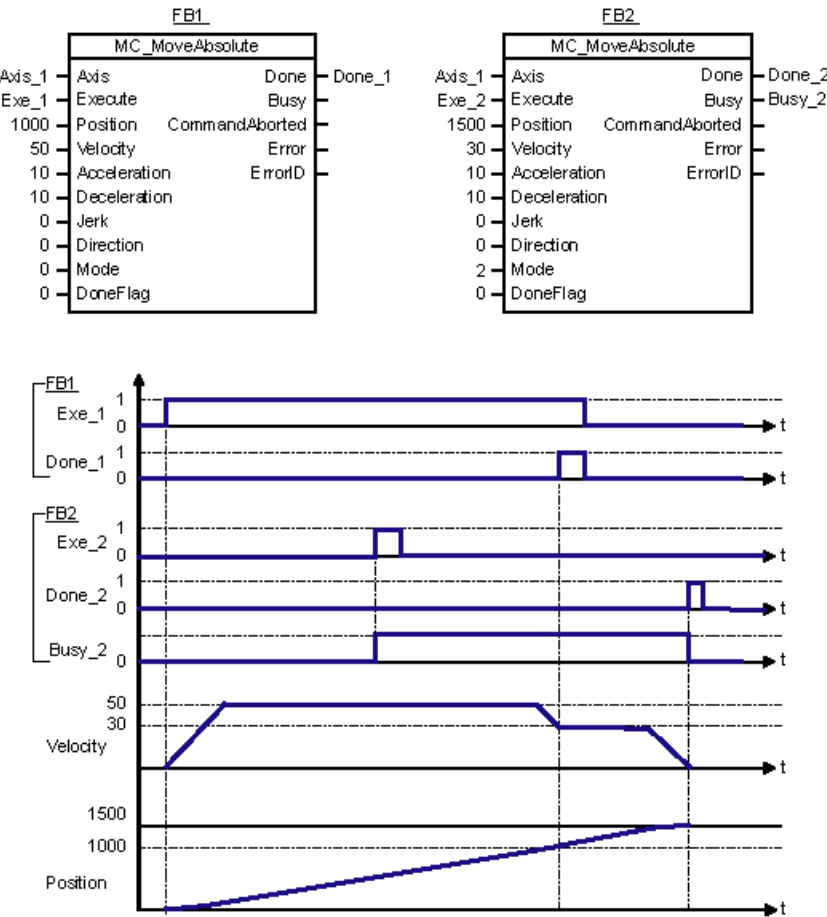
当前速度 > 新速度

相对定位通过 FB1 的输入参数 *Execute* (Exe_1) 的上升沿启动。轴 (Axis_1) 将以定义的 *Acceleration* = 10 加速到其最终速度 50。在轴到达其目标位置 (*Position* = 1000) 之前，将输出另一个 MC_MoveAbsolute 命令 (FB2 的 Exe_2 上升沿)。Busy_2 将报告新命令处于活动状态。

通过设置 FB2 的 *Mode* = 2 来“覆盖”当前定位命令而不是将其取消。

“MC_MoveAbsolute”将通过 *Deceleration* = 10 使轴减速，从而使轴在到达目标位置 1000 时达到覆盖 MC_MoveAbsolute 命令为它设置的最终速度 30。通过 Done_1 来报告 FB1 中命令完成。

在到达目标位置 1000 后，该轴 (Axis_1) 继续以速度 30 和 *Deceleration* = 10 逼近目标位置 1500。Done_2 会在 FB2 报告该轴已到达目标位置；Busy_2 将更改为 FALSE。



MC_MoveAbsolute — 示例 — “覆盖运动 2”

以下显示的信号图显示了当前速度小于新速度的情况下工艺功能“MC_MoveAbsolute”(Mode = 2) 的“覆盖”特性。

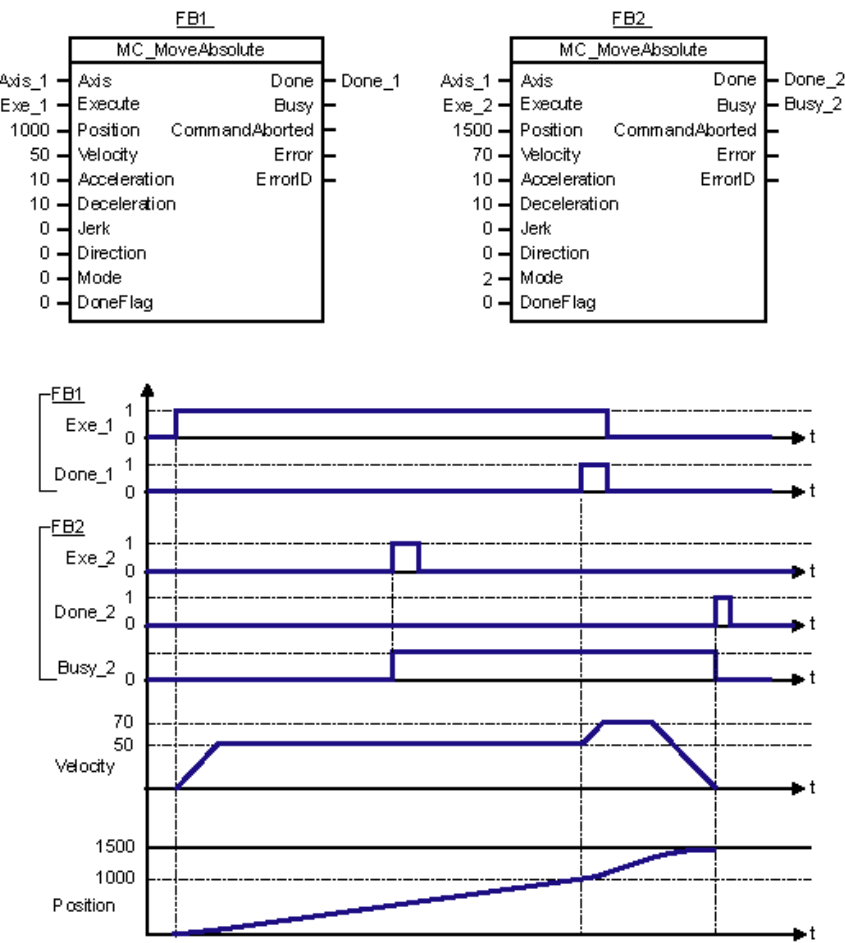
当前速度 < 新速度

相对定位通过 FB1 的输入参数 *Execute* (*Exe_1*) 的上升沿启动。轴 (*Axis_1*) 将以定义的 *Acceleration* = 10 加速到其最终速度 50。在轴到达其目标位置 (*Position* = 1000) 之前，将输出另一个 MC_MoveAbsolute 命令 (FB2 的 *Exe_2* 上升沿)。*Busy_2* 将报告新命令处于活动状态。

通过设置 FB2 的 *Mode* = 2 来“覆盖”当前定位命令而不是将其取消。

“MC_MoveAbsolute”将按第一个命令设置的速度 50 逼近目标位置 1000。通过 *Done_1* 来报告 FB1 中命令完成。

在到达目标位置 1000 后，该轴 (*Axis_1*) 将以 *Acceleration* = 10 加速到由第二个命令设置的最终速度 70。该轴将以 *Deceleration* = 10 逼近目标位置 1500。*Done_2* 会在 FB2 报告该轴已到达目标位置；*Busy_2* 将更改为 *FALSE*。



MC_MoveAbsolute — 示例 —“覆盖运动 3”

以下信号图显示了反转方向情况下工艺功能“MC_MoveAbsolute”(Mode = 2) 的“覆盖”特性。

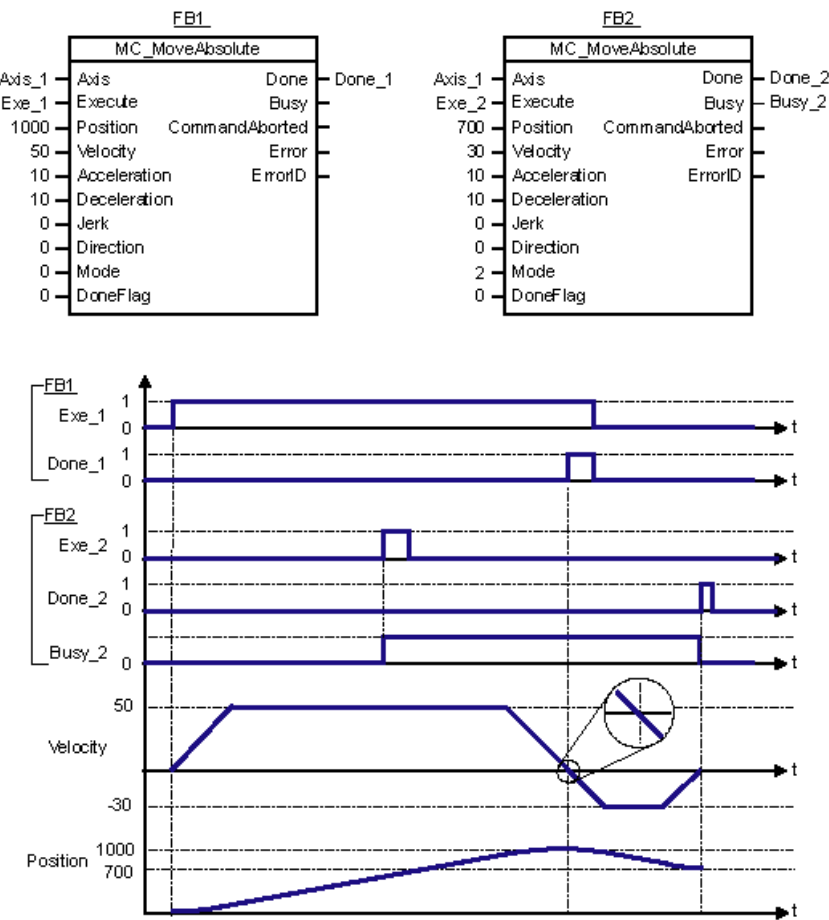
反转方向

相对定位通过 FB1 的输入参数 *Execute* (Exe_1) 的上升沿启动。轴 (Axis_1) 将以定义的 *Acceleration* = 10 加速到其最终速度 50。在轴到达其目标位置 (*Position* = 1000) 之前，将输出另一个 MC_MoveAbsolute 命令 (FB2 的 Exe_2 上升沿)。Busy_2 将报告新命令处于活动状态。

通过设置 FB2 的 *Mode* = 2 来“覆盖”当前定位命令而不是将其取消。当前处于活动状态的 MC_MoveAbsolute 命令的目标位置值 1000 大于下一个 MC_MoveAbsolute 命令的目标位置 700。因此，您必须反转方向。“MC_MoveAbsolute”将通过 *Deceleration* = 10 使轴减速，从而使轴在目标位置 1000 达到速度 0。通过 Done_1 来报告 FB1 中命令完成。

在到达目标位置 1000 后，轴 (Axis_1) 将以 *Acceleration* = 10 减速至速度 -30。以下缩放视图表明在跳转时没有间隔。该轴将以 *Deceleration* = 10 逼近目标位置 700。

Done_2 会在 FB2 报告该轴已到达目标位置；Busy_2 将更改为 FALSE。



MC_MoveAbsolute — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8040	轴/外部编码器被禁用，或者设置了错误的模式	缺少运动命令所需的使能信号。 消除并确认所有排队的错误，然后启用相关的轴模式（例如，位置控制）。
8043	参数值非法	涉及数据类型为 REAL 的所有输入参数，或者输入参数 <i>Mode</i>、<i>Direction</i> 或者 <i>DoneFlag</i>。 示例： <i>Mode</i> = 2 和 <i>Direction</i> = 4。
8044	命令不受工艺对象支持	例如，将命令发送至输出凸轮。
8045	在当前状态下不允许使用该命令	"MC_Stop"处于活动状态
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是可能无法通过状态输出对其进行监视。

ErrorID	错误消息	说明/纠正或避免错误的方法
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8055	当前状态下不允许锁存运动	当前轴状态下无法接受锁存命令。原因： - 正在该轴上执行基础同步或叠加同步命令 "MC_Halt"处于活动状态
8056	由于正在向固定挡块运动而取消	轴已运动到固定挡块 (<i>InClamping = TRUE</i>)。只有新命令从固定挡块释放轴，它们才被接受。 已启动的命令不符合该条件。
8083	DB 不是工艺 DB	通过输入参数 <i>Axis</i> 指定的 DB 未找到或不是工艺 DB。

ErrorID	错误消息	说明/纠正或避免错误的方法
8084	工艺 DB 无效	- 在通过输入参数 <i>Axis</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Axis</i> 更改 DB 号。 - 用户在输入参数 <i>Axis</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management)中删除工艺 DB，然后创建一个新的工艺 DB。
808B	参数值的 REAL 格式无效	数据类型为 REAL 的输入参数的值不符合有效的浮点数格式。 检查输入参数值或背景数据块数据。不能以浮点格式表示格式无效的值。它们以十六进制格式 (DW16# ...) 显示。

6.2.7 FB411 MC_MoveRelative — 相对定位

使用 FB 411“MC_MoveRelative”进行相对定位

适用于固件版本为 V3.1.x 或更高版本的集成工艺

本部分介绍了固件版本为 V3.0.x 的应用

用途

- “MC_MoveRelative”工艺功能通过指定起始位置的相对位置来对轴进行定位操作。如果在开始执行命令时该轴已处于运动状态并且通过 *Mode = 0* 设置了“运动超驰”，则系统将使用开始执行命令时内部可用的起始位置。必须基于装载和周期时间为工艺功能和命令执行之间的响应时间留有余量。
- 通过输入参数 *Velocity*、*Jerk*、*Acceleration* 和 *Deceleration* 来定义轴运动的动态响应。
- 在输入参数 *Mode* 中定义是否要触发活动的运动，或者定义是添加运动还是覆盖运动。

适用对象

- 定位轴
- 同步轴

先决条件

- 已启用轴进行位置控制操作
- 没有处于活动状态的 MC_Stop 命令
- 保存和覆盖运动的规则 (*Mode = 1、2*) :
 - “MC_CamIn”和“MC_GearIn”都未启动。
 - 没有处于活动状态的基本同步

命令的相互作用

新命令 — 活动的单个命令 (2)

新命令 — 活动的命令 (2)

输入参数

参数	数据类型	初始值	说明
<i>Axis</i>	INT	0	工艺 DB 的编号
<i>Execute</i>	BOOL	FALSE	在上升沿处启动命令
<i>Distance</i>	REAL	0.0	定位距离 (负或正)
<i>Velocity</i>	REAL	-1.0	最大速度 (不是总能达到) :
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值
<i>Acceleration</i>	REAL	-1.0	加速 (增加电机功率) :
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值
<i>Deceleration</i>	REAL	-1.0	减速 (降低电机功率) :
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值
<i>Jerk</i>	REAL	-1.0	冲击:
			值 > 0: 使用定义的值
			值 = 0: 使用梯形运动曲线
			值 < 0: 使用默认值
<i>Mode</i>	INT	0	超驰模式:
			值 = 0: 超驰运动: 当前运动被取消
			值 = 1: 添加运动: 运动命令被写入命令缓冲区。轴将在运动跳转时停止。
			值 = 2: 覆盖运动: 运动命令被写入命令缓冲区。该运动跳转被覆盖。
<i>DoneFlag</i>	INT	0	在 MCDevice DB 中生成 DoneFlag

输出参数（状态输出）

参数	数据类型	初始值	说明	
<i>Done</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 已到达目标位置	
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令	
<i>CommandAborted</i>	BOOL	<i>FALSE</i>	<i>TRUE</i>: 命令由其它命令或因其执行过程中的错误而取消。 如果工艺 DB 的 <i>ErrorStatus</i> 中未显示错误, 则该命令由后续命令取消。 如果工艺 DB 的变量 <i>ErrorStatus</i> 中指示有错误, 则命令执行期间影响该工艺对象的错误将导致该命令中止。	
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> :	命令启动时 有 错误。未执行该命令。有关原因的信息, 请参考 <i>ErrorID</i> 。
			<i>FALSE</i> :	命令启动时 没有 错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。	

MC_MoveRelative — 示例 —“超驰运动”

以下信号图显示了 (*Mode* = 0) 工艺功能“MC_MoveRelative”的超驰特性。

阶段“a”

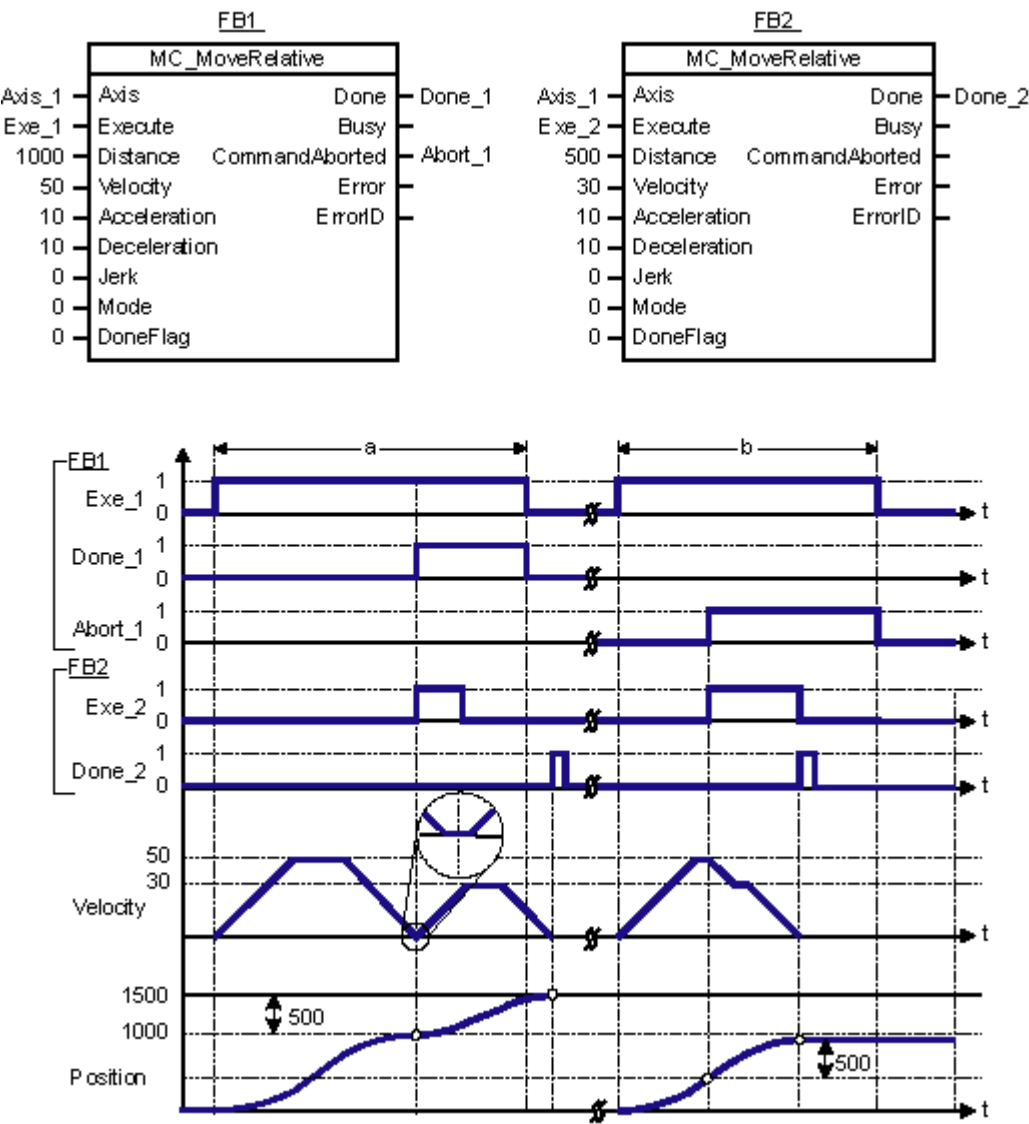
FB1 的输入参数 *Execute* (*Exe_1*) 的第一个上升沿用于启动轴 (*Axis_1*) 的相对定位。轴到达 *Distance* = 1000 之后, *Done_1* 将更改为 *TRUE*。

另一个相对定位命令在 *Done_1* = *TRUE* 时在 FB2 中启动 (输入参数 *Execute* (*Exe_2*))。命令执行期间的响应时间将导致轴短暂停止。请参见缩放视图。短暂停止之后, 轴 (*Axis_1*) 将继续移动 500 的距离。 *Done_2* 将报告命令完成。

阶段“b”

FB1 输入参数 *Execute (Exe_1)* 的另一个上升沿用于启动轴 (*Axis_1*) 的相对定位。

在轴到达目标位置 *1000* 之前，另一个定位命令将在 FB2 中启动（输入参数 *Execute (Exe_2)*）。*Abort_1* 将报告 FB1 中活动定位命令的取消。轴 (*Axis_1*) 将以定义的 *Deceleration = 10* 移动，直到速度 *Velocity = 30*。FB2 的 *Done_2* 将报告轴已运行了 *Distance = 500*。



MC_MoveRelative — 示例 — “添加运动”

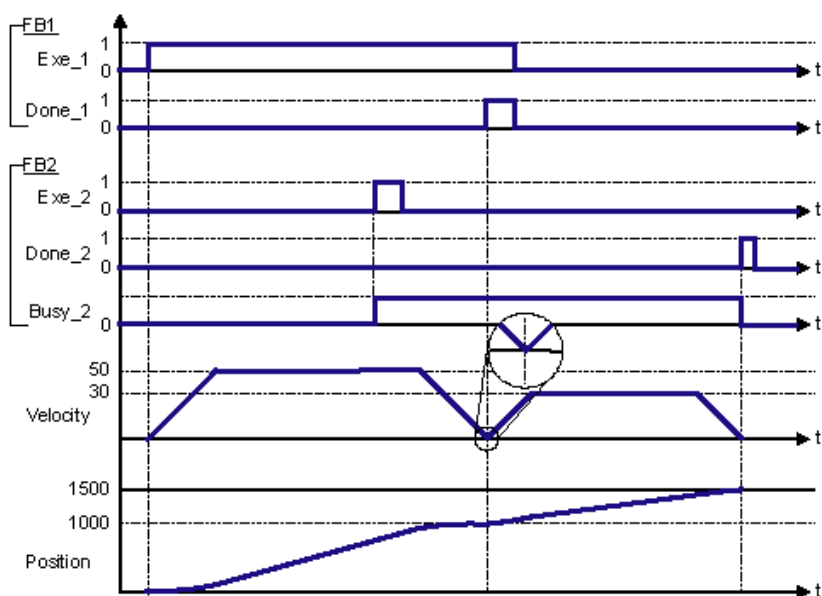
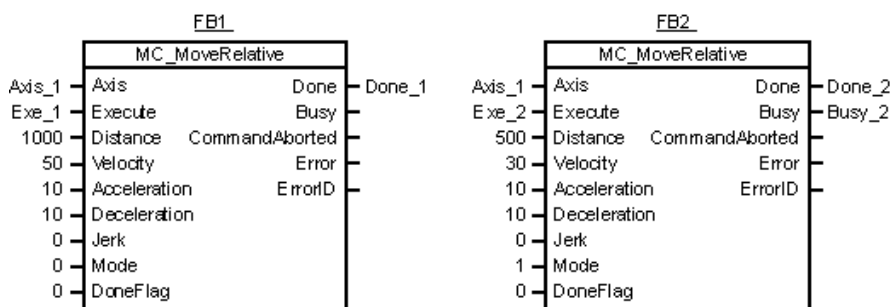
以下信号图显示了工艺功能“MC_MoveRelative”(Mode = 1) 的“添加”特性。

相对定位通过 FB1 的输入参数 *Execute* (Exe_1) 的上升沿启动。轴 (Axis_1) 将以定义的 *Acceleration* = 10 加速到其最终速度 50。在轴运行定义的距离 (*Distance* = 1000) 之前，另一个 MC_MoveRelative 命令将启动（在 FB2 的 *Exe_2* 上升沿）。*Busy_2* 将报告新命令处于活动状态。

通过设置 FB2 的 *Mode* = 1 来“添加”当前定位命令而不是将其取消。

“MC_MoveRelative”将通过 *Deceleration* = 10 来为该轴减速，然后根据启动的命令运行 1000 的距离。通过 *Done_1* 报告命令完成。

轴行进 1000 的距离后，将立即添加第二个 MC_MoveRelative 命令的运动，期间没有任何时间上的间隔。请参见缩放视图。轴 (Axis_1) 将沿设置的加速和减速斜坡移动 500 的距离。轴运行了 *Distance* = 500 之后，*Done_2* 将更改为 TRUE，*Busy_2* 将更改为 FALSE。



MC_MoveRelative — 示例 — “覆盖运动 1”

以下显示的信号图显示了当前速度大于新速度的情况下工艺功能“MC_MoveRelative”(Mode = 2) 的“覆盖”特性。

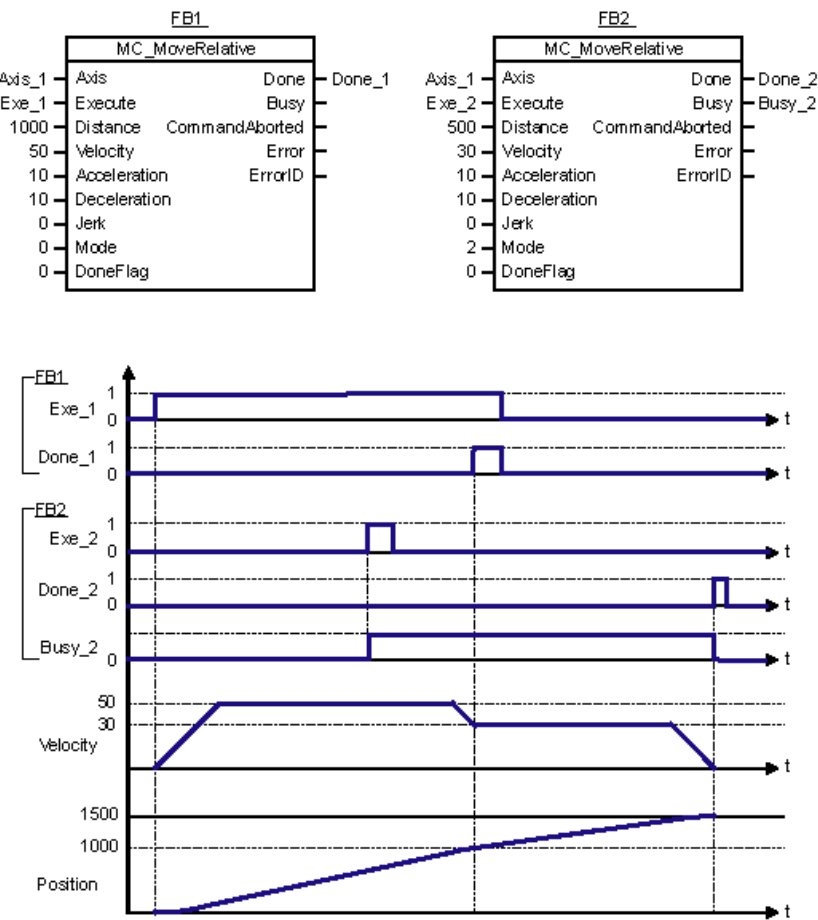
当前速度 > 新速度

相对定位通过 FB1 的输入参数 *Execute (Exe_1)* 的上升沿启动。轴 (*Axis_1*) 将以定义的 *Acceleration = 10* 加速到其最终速度 *50*。在轴运行定义的距离 (*Distance = 1000*) 之前，另一个 MC_MoveRelative 命令将启动（在 FB2 的 *Exe_2* 上升沿）。*Busy_2* 将报告新命令处于活动状态。

通过设置 FB2 的 *Mode = 2* 来“覆盖”当前定位命令而不是将其取消。

“MC_MoveRelative”将通过 *Deceleration = 10* 使轴减速，从而使轴在运行了 *1000* 的距离后达到覆盖 MC_MoveRelative 命令为它设置的最终速度 *30*。通过 *Done_1* 来报告 FB1 中命令完成。

运行了 *1000* 的距离后，轴 (*Axis_1*) 将继续以速度 *30* 移动。轴 (*Axis_1*) 将沿设置的速度和减速斜坡移动 *500* 的距离。轴运行了 *Distance = 500* 之后，*Done_2* 将更改为 *TRUE*，*Busy_2* 将更改为 *FALSE*。



MC_MoveRelative — 示例 — “覆盖运动 2”

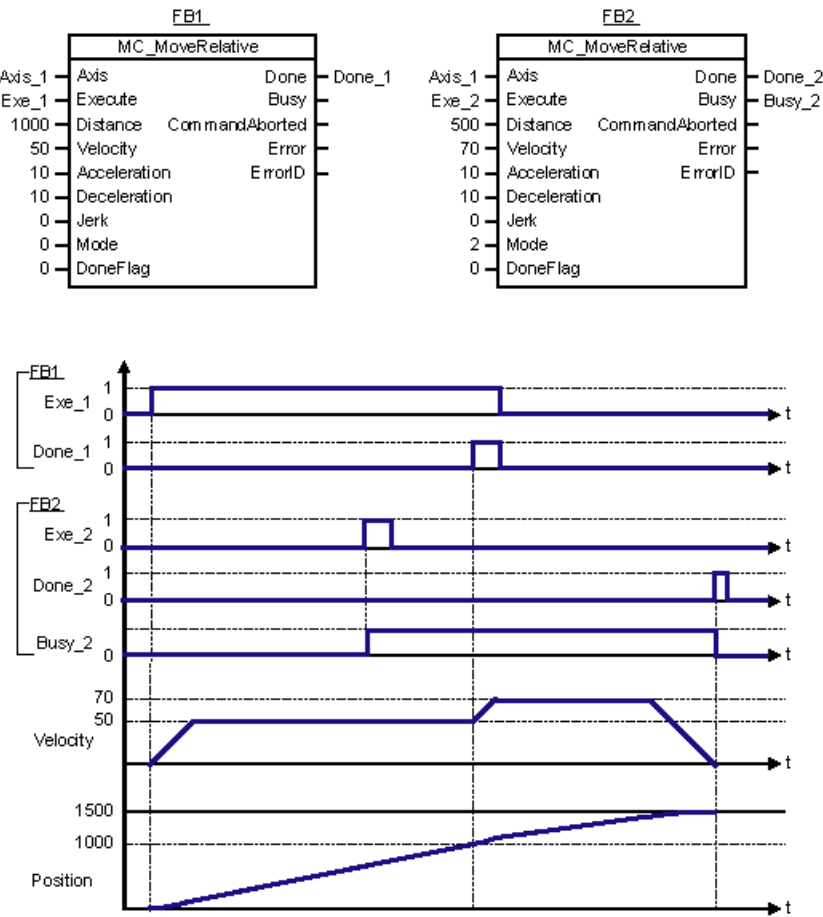
以下显示的信号图显示了当前速度小于新速度的情况下工艺功能“MC_MoveRelative”(Mode = 2) 的“合并”特性。

当前速度 < 新速度

相对定位在 FB1 输入参数 *Execute (Exe_1)* 的上升沿启动。轴 (*Axis_1*) 将以指定的速率 (*Acceleration = 10*) 加速到其最终速度 *50*。在轴运行了定义的距离 (*Distance = 1000*) 之前，另一个 MC_MoveRelative 命令将启动（在 FB2 的 *Exe_2* 上升沿）。*Busy_2* 将发送信号，指示新命令处于活动状态。

通过设置 FB2 的 *Mode = 2* 来“合并”当前定位命令而不是将其取消。
“MC_MoveRelative”将使轴以最终速度 *50* 运行剩余的距离。通过 *Done_1* 报告 FB1 中命令完成。

运行距离达到 *1000* 后，轴 (*Axis_1*) 将以 *Acceleration = 10* 加速到第二个命令设置的最终速度 *70*。轴 (*Axis_1*) 将以设置的速度和减速度行进剩余的距离。当运行距离达到 *Distance = 500* 后，*Done_2* 将切换为 *TRUE*，同时 *Busy_2* 将切换为 *FALSE*。



MC_MoveRelative — 示例 —“覆盖运动 3”

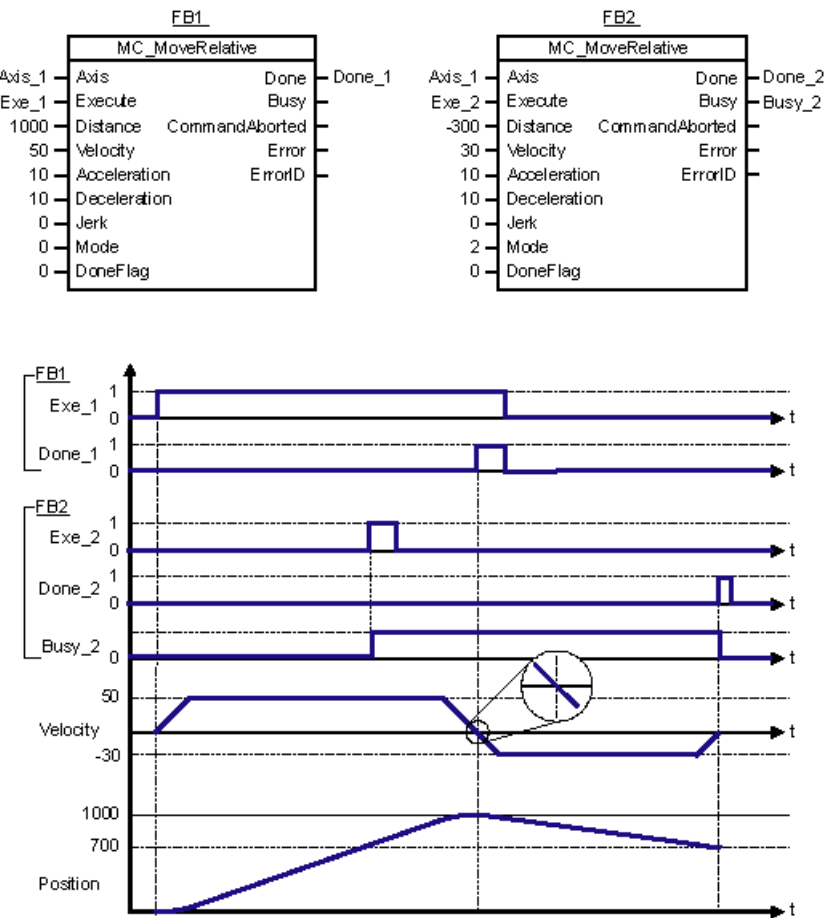
以下信号图显示了反转方向情况下工艺功能“MC_MoveRelative”(Mode = 2) 的“覆盖”特性。

反转方向

相对定位通过 FB1 的输入参数 *Execute (Exe_1)* 的上升沿启动。轴 (*Axis_1*) 将以定义的 *Acceleration = 10* 加速到其最终速度 *50*。在轴运行定义的距离 (*Distance = 1000*) 之前，另一个 MC_MoveRelative 命令将启动（在 FB2 的 *Exe_2* 上升沿）。*Busy_2* 将报告新命令处于活动状态。

通过设置 FB2 的 *Mode = 2* 来“覆盖”当前定位命令而不是将其取消。方向已反转，因为在 FB2 中设置了反向距离 (*Distance = -300*)。 “MC_MoveRelative”将通过 *Deceleration = 10* 使轴减速，从而使轴在运行了 *1000* 的距离后达到 *0* 速度。通过 *Done_1* 来报告 FB1 中命令完成。

运行了 *1000* 的距离后，轴 (*Axis_1*) 将通过 *Acceleration = 10* 加速到速度 *-30*。以下缩放视图表明在跳转时没有间隔。轴 (*Axis_1*) 将以设置的速度和减速度运行剩余的距离。轴运行了 *Distance = -300* 之后，*Done_2* 将更改为 *TRUE*，*Busy_2* 将更改为 *FALSE*。



MC_MoveRelative — ErrorID

适用于固件版本为 **V3.1.x** 或更高版本的集成工艺

本部分介绍了固件版本为 **V3.0.x** 的应用

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8040	轴/外部编码器被禁用，或者设置了错误的模式	缺少运动命令所需的使能信号。 消除并确认所有排队的错误，然后启用相关的轴模式（例如，位置控制）。
8043	参数值非法	涉及数据类型为 <i>REAL</i> 的所有输入参数，或者输入参数 <i>DoneFlag</i> 或 <i>Mode</i> 。
8044	命令不受工艺对象支持	例如，将命令发送至输出凸轮
8045	在当前状态下不允许使用该命令	例如 <i>MC_Stop</i> 命令处于活动状态

ErrorID	错误消息	说明/纠正或避免错误的方法
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是可能无法通过状态输出对其进行监视。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8055	当前状态下不允许锁存运动	当前轴状态下无法接受锁存命令。原因： - 正在该轴上执行基础同步或叠加同步命令 "MC_Halt"处于活动状态
8056	由于正在向固定挡块运动而取消	轴已运动到固定挡块 (<i>InClamping = TRUE</i>)。只有新命令从固定挡块释放轴，它们才被接受。 已启动的命令不符合该条件。
8083	DB 不是工艺 DB	通过输入参数 <i>Axis</i> 指定的 DB 未找到或不是工艺 DB。

ErrorID	错误消息	说明/纠正或避免错误的方法
8084	工艺 DB 无效	- 在通过输入参数 <i>Axis</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Axis</i> 更改 DB 号。 - 用户在输入参数 <i>Axis</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management)中删除工艺 DB，然后创建一个新的工艺 DB。
808B	参数值的 REAL 格式无效	数据类型为 REAL 的输入参数的值不符合有效的浮点数格式。 检查输入参数值或背景数据块数据。 不能以浮点格式表示格式无效的值。它们以十六进制格式 (DW16# ...) 显示。

6.2.8 FB412 MC_MoveAdditive — 相对定位到当前目标位置

使用 FB 412“MC_MoveAdditive”相对定位到当前目标位置

用途

- “MC_MoveAdditive”工艺功能通过用户自定义的动态值将轴定位到当前定位命令目的地的相对位置。该方法可以通过定义的距离校正先前得到的目标位置。
- 在输入参数 *Velocity*、*Jerk*、*Acceleration*、*Deceleration* 中定义运动的动态响应。
- 轴将在目标位置停止
- MC_MoveAdditive 命令将超驰活动的命令

适用对象

- 定位轴
- 同步轴

先决条件

- 必须启用该轴以进行位置控制操作。
- 如果在组态中设置了“需要回原点”并且出现以下情况时，轴必须回原点
 - 轴处于运动状态，
 - 或者“MC_MoveAdditive”超驰活动的运动命令（“MC_MoveVelocity”除外）。
- 出现以下情况时，轴不必回原点
 - 轴处于停止状态
 - 活动的 MC_MoveVelocity 命令被超驰
- 未在执行 MC_Stop 命令。

说明

在达到以下某个要求时，“MC_MoveAdditive”将与“MC_MoveRelative”实现相同的功能：

- 轴在命令启动时处于停止状态
 - 连续功能被超驰，即未定义目标位置时。在这种情况下，目标位置取决于超驰位置，并根据 *Distance* 和操作开始时的位置来获得。
-

对模数轴的限制

对模数轴应用该功能是受限制的，仅适用于较短距离（根据模数长度，不能超出当前模数长度）：

新目标位置 = (旧目标位置 + 距离) 模数长度

- *Distance* 必须小于一个模数长度。
- 剩余距离必须小于一个模数长度。
- *Distance* 加上超驰命令的剩余距离必须小于一个模数长度。

命令的相互作用

新命令 — 活动的单个命令 (2)

新命令 — 活动的命令 (2)

输入参数

参数	数据类型	初始值	说明
<i>Axis</i>	INT	0	工艺 DB 的编号
<i>Execute</i>	BOOL	FALSE	在上升沿处启动命令
<i>Distance</i>	REAL	0.0	定位距离（负或正）
<i>Velocity</i>	REAL	-1.0	最大速度（不是总能达到）：
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值
<i>Acceleration</i>	REAL	-1.0	加速（增加电机功率）：
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值
<i>Deceleration</i>	REAL	-1.0	减速（降低电机功率）：
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值

参数	数据类型	初始值	说明
<i>Jerk</i>	REAL	<i>-1.0</i>	冲击:
			值 > 0: 使用定义的值
			值 = 0: 使用梯形运动曲线
			值 < 0: 使用默认值
<i>DoneFlag</i>	INT	<i>0</i>	在 MCDevice DB 中生成 DoneFlag

输出参数（状态输出）

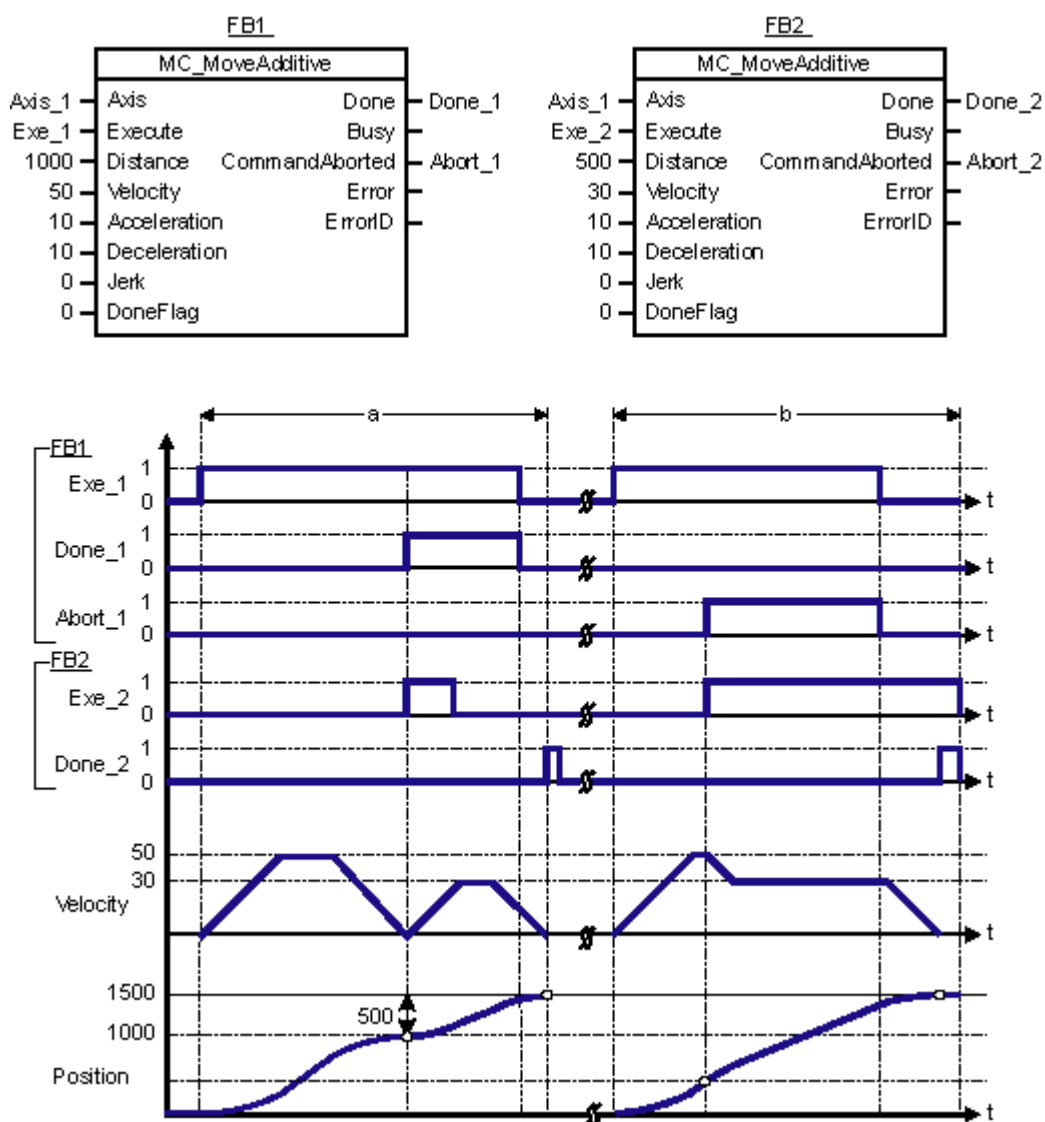
参数	数据类型	初始值	说明
<i>Done</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 已到达目标位置
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令
<i>CommandAborted</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令由其它命令或因其执行过程中的错误而取消。 如果工艺 DB 的 <i>ErrorStatus</i> 中未显示错误, 则该命令由后续命令取消。 如果工艺 DB 的变量 <i>ErrorStatus</i> 中指示有错误, 则命令执行期间影响该工艺对象的错误将导致该命令中止。
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令启动时有错误。未执行该命令。有关原因的信息, 请参考 <i>ErrorID</i> 。
			<i>FALSE</i> : 命令启动时没有错误。
<i>ErrorID</i>	WORD	<i>0</i>	输出参数 <i>Error</i> 的 ErrorID。

MC_MoveAdditive — 示例

信号图

示例 a：两个 MC_MoveAdditive 命令相继启动。

示例 b：第二个 MC_MoveAdditive (FB 2) 在第一个命令完成之前启动。该操作将取消第一个命令 (FB 1)。目标位置从第一个命令的目标位置获得，并通过第二个命令的距离来校正。



MC_MoveAdditive — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8040	轴/外部编码器被禁用，或者设置了错误的模式	缺少运动命令所需的使能信号。 消除并确认所有排队的错误，然后启用相关的轴模式（例如，位置控制）。
8043	参数值非法	涉及数据类型为 REAL 的所有输入参数，或者输入参数 <i>DoneFlag</i> 。
8044	命令不受工艺对象支持	例如，将命令发送至输出凸轮。
8045	在当前状态下不允许使用该命令	MC_Stop - 命令处于活动状态。
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是可能无法通过状态输出对其进行监视。

ErrorID	错误消息	说明/纠正或避免错误的方法
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8056	由于正在向固定挡块运动而取消	轴已运动到固定挡块 (<i>InClamping = TRUE</i>)。只有新命令从固定挡块释放轴，它们才被接受。 已启动的命令不符合该条件。
8083	DB 不是工艺 DB	通过输入参数 <i>Axis</i> 指定的 DB 未找到或不是工艺 DB。
8084	工艺 DB 无效	- 在通过输入参数 <i>Axis</i> 定义的工艺 DB 的控制器中不存在工艺对象。 - 将当前工艺下载到目标系统，或通过输入参数 <i>Axis</i> 更改 DB 号。 - 用户在输入参数 <i>Axis</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management) 中删除工艺 DB，然后创建一个新的工艺 DB。
808B	参数值的 REAL 格式无效	数据类型为 REAL 的输入参数的值不符合有效的浮点数格式。 检查输入参数值或背景数据块数据。不能以浮点格式表示格式无效的值。它们以十六进制格式 (DW16# ...) 显示。

6.2.9 FB413 MC_MoveSuperImposed — 叠加定位

使用 FB413“MC_MoveSuperImposed”进行叠加定位

用途

- “MC_MoveSuperImposed”工艺功能允许轴的叠加定位，例如进行压力标记校正。
- 在输入参数 *VelocityDiff*、*Jerk*、*Acceleration*、*Deceleration* 中定义运动的动态响应。
- 将工艺功能“MC_MoveSuperImposed”的动态值加起来形成基本运动的动态值。如果从该相加操作中获得的动态值大于在 S7T Config（默认值 [Default] > 动态响应 [Dynamic response] 对话框）中组态的动态值，则实际动态值将限制为在 S7T Config 中组态的值。在这种情况下，轴工艺 DB 将返回警告 0021。
- 活动的定位或同步操作不会取消。
- 活动的 MC_MoveSuperImposed 命令被新的 MC_MoveSuperImposed 命令超驰。所取消 MC_MoveSuperImposed 的剩余距离值会丢失！

适用对象

- 定位轴
- 同步轴

先决条件

- 已启用轴进行位置控制操作
- 没有处于活动状态的 MC_Stop 命令
- 通常需要增加轴速度以进行叠加定位。因此，当设置了该定位功能时，轴的基本速度可能尚未达到其最大值。

命令的相互作用

新命令 — 活动的单个命令 (2)

新命令 — 活动的命令 (2)

输入参数

参数	数据类型	初始值	说明
<i>Axis</i>	INT	0	工艺 DB 的编号
<i>Execute</i>	BOOL	FALSE	在上升沿处启动命令
<i>Distance</i>	REAL	0.0	进行叠加定位的其它距离
<i>VelocityDiff</i>	REAL	-1.0	相对于当前运动的最大速度偏差：
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值
<i>Acceleration</i>	REAL	-1.0	加速（增加电机功率）：
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值
<i>Deceleration</i>	REAL	-1.0	减速（降低电机功率）：
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值
<i>Jerk</i>	REAL	-1.0	冲击：
			值 > 0: 使用定义的值
			值 = 0: 使用梯形运动曲线
			值 < 0: 使用默认值
<i>DoneFlag</i>	INT	0	在 MCDevice DB 中生成 DoneFlag

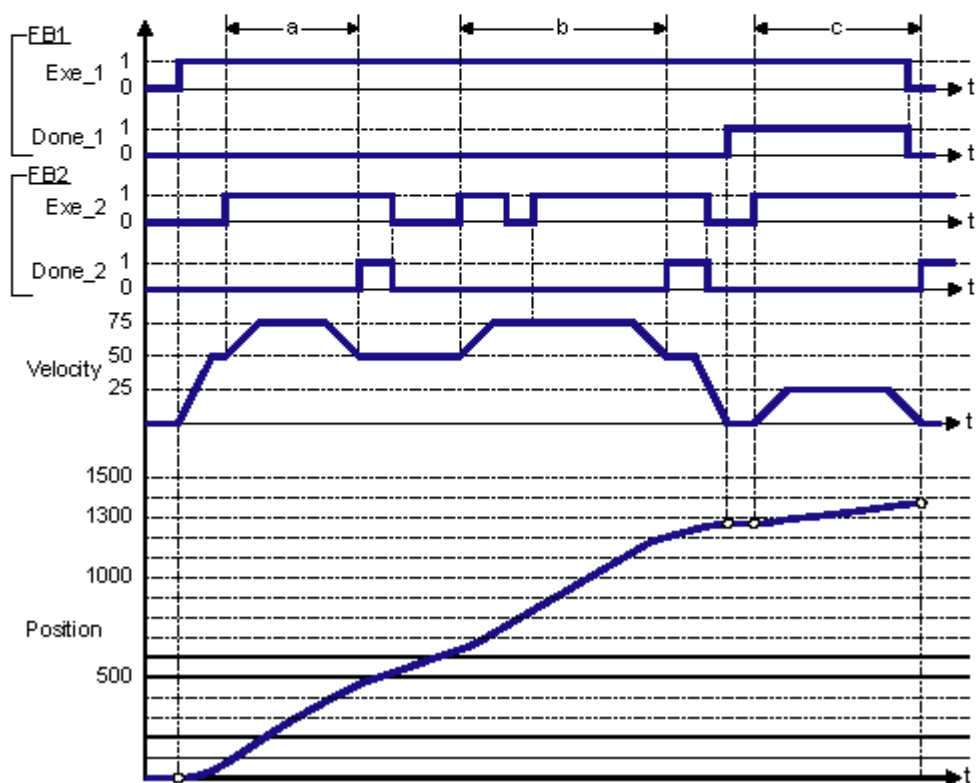
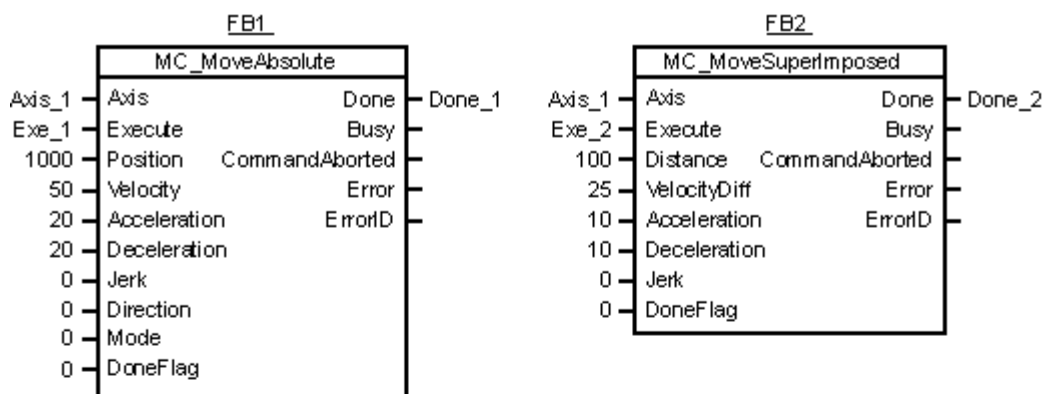
输出参数（状态输出）

参数	数据类型	初始值	说明	
<i>Done</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 已完成叠加定位	
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在进行叠加定位	
<i>CommandAborted</i>	BOOL	<i>FALSE</i>	<i>TRUE</i>: 命令由其它命令或因其执行过程中的错误而取消。 如果工艺 DB 的 <i>ErrorStatus</i> 中未显示错误, 则该命令由后续命令取消。 如果工艺 DB 的变量 <i>ErrorStatus</i> 中指示有错误, 则命令执行期间影响该工艺对象的错误将导致该命令中止。	
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> :	命令启动时 有错误 。未执行该命令。有关原因的信息, 请参考 <i>ErrorID</i> 。
			<i>FALSE</i> :	命令启动时 没有错误 。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 ErrorID。	

MC_MoveSuperImposed — 示例 — “绝对定位”

信号图： 叠加定位的作用

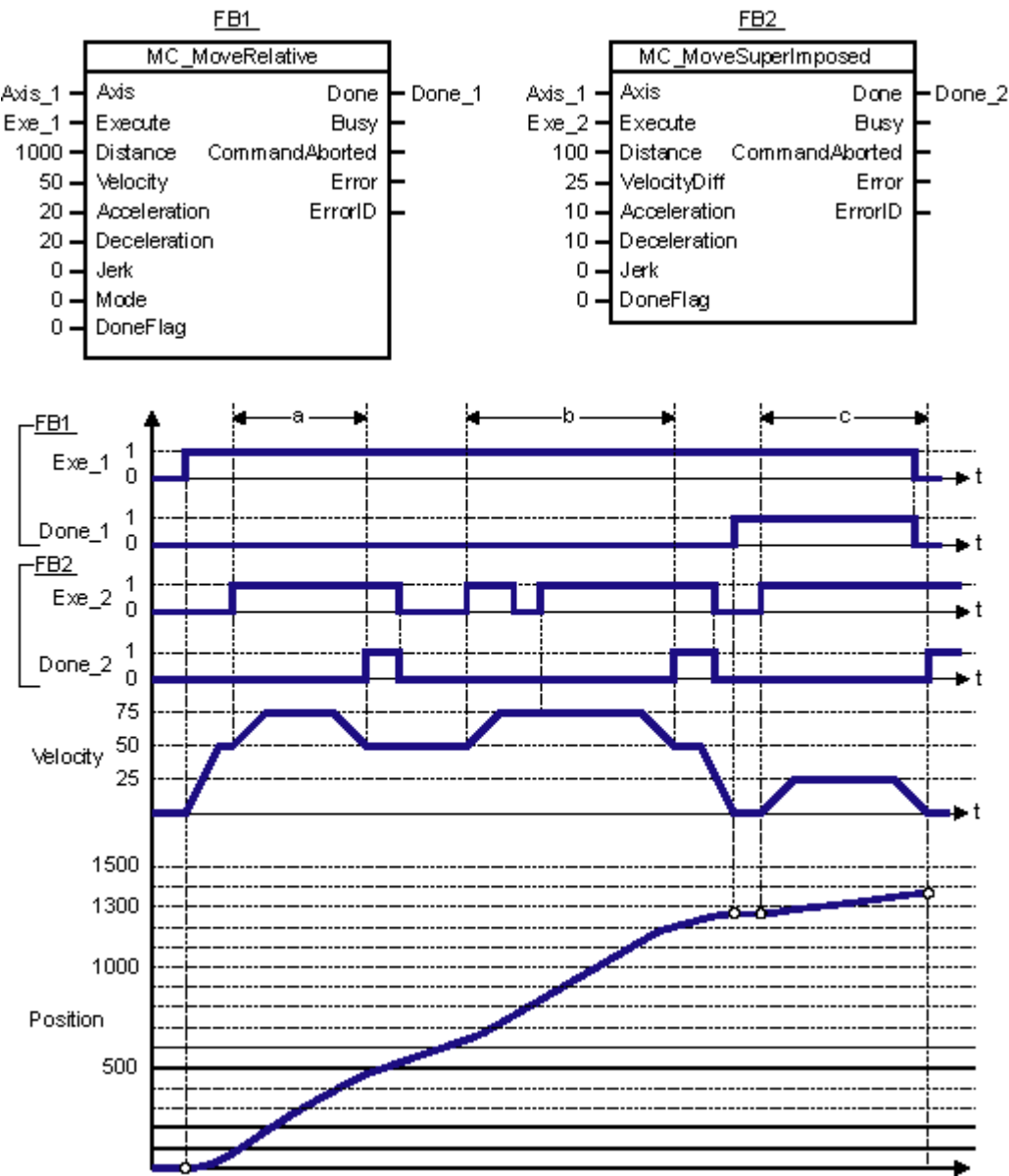
- 在绝对定位期间启动“MC_MoveSuperImposed”。
- 在“MC_MoveSuperImposed”完成之前重新启动“MC_MoveSuperImposed”。
- 当轴处于停止状态时启动“MC_MoveSuperImposed”。



MC_MoveSuperImposed — 示例 — “相对定位”

信号图： 叠加定位的作用

- a) 在相对定位期间启动“MC_MoveSuperImposed”。
- (b) 在“MC_MoveSuperImposed”完成之前重新启动“MC_MoveSuperImposed”。
- (c) 当轴处于停止状态时启动“MC_MoveSuperImposed”。



MC_MoveSuperImposed — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8040	轴/外部编码器被禁用，或者设置了错误的模式	缺少运动命令所需的使能信号。 消除并确认所有排队的错误，然后启用相关的轴模式（例如，位置控制）。
8043	参数值非法	涉及数据类型为 <i>REAL</i> 的所有输入参数，或者输入参数 <i>DoneFlag</i> 。
8044	命令不受工艺对象支持	例如对速度控制轴的命令请求。
8045	在当前状态下不允许使用该命令	<i>MC_Stop</i> 命令处于活动状态。
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是可能无法通过状态输出对其进行监视。

ErrorID	错误消息	说明/纠正或避免错误的方法
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8056	由于正在向固定挡块运动而取消	轴已运动到固定挡块 (<i>InClamping = TRUE</i>)。只有新命令从固定挡块释放轴，它们才被接受。 已启动的命令不符合该条件。
8083	DB 不是工艺 DB	通过输入参数 <i>Axis</i> 指定的 DB 未找到或不是工艺 DB。

ErrorID	错误消息	说明/纠正或避免错误的方法
8084	工艺 DB 无效	- 在通过输入参数 <i>Axis</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Axis</i> 更改 DB 号。 - 用户在输入参数 <i>Axis</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management)中删除工艺 DB，然后创建一个新的工艺 DB。
808B	参数值的 REAL 格式无效	数据类型为 REAL 的输入参数的值不符合有效的浮点数格式。 检查输入参数值或背景数据块数据。 不能以浮点格式表示格式无效的值。它们以十六进制格式 (DW16# ...) 显示。

6.2.10 FB 414 MC_MoveVelocity — 预设了速度的运动

使用 FB 414“MC_MoveVelocity”移动预设了速度的轴

适用于固件版本为 V3.1.x 或更高版本的集成工艺

本部分介绍了固件版本为 V3.0.x 的应用

用途

- “MC_MoveVelocity”工艺功能用于将轴加速到预设速度。
- 在输入参数 *Jerk*、*Acceleration*、*Deceleration* 中定义运动的动态响应。
- 在计算最终速度时，为任何活动的速度超驰功能留出余量（输出参数 *InVelocity*）。在用户程序中为这种响应留出余量。
- 您可以使用输入参数 *PositionControl* 设置位置控制轴的位置控制模式或速度控制模式。
- 使用输入参数 *Mode* 指定要触发活动的运动、超驰当前运动还是添加运动。

适用对象

- 速度控制轴
- 定位轴
- 同步轴

先决条件

- 启用轴进行速度控制操作或位置控制操作
- 没有处于活动状态的 MC_Stop 命令
- 添加运动的规则 (*Mode = 1*):
 - “MC_CamIn”和“MC_GearIn”都未启动。
 - 没有处于活动状态的基本同步
 - “MC_MoveVelocity”无法保存并添加到已经处于活动状态的“MC_MoveVelocity”

命令的相互作用

新命令 — 活动的单个命令 (2)

新命令 — 活动的命令 (2)

输入参数

参数	数据类型	初始值	说明
<i>Axis</i>	INT	0	工艺 DB 的编号
<i>Execute</i>	BOOL	FALSE	在上升沿处启动命令
<i>Velocity</i>	REAL	-1.0	最终速度:
			值 < 0: 使用默认值
			值 = 0: 支持
			值 > 0: 最终速度
<i>Acceleration</i>	REAL	-1.0	加速 (增加电机功率):
			值 > 0: 使用定义的值
			值 = 0: 仅当轴不必加速就能达到最终速度时才允许。
			值 < 0: 使用默认值
<i>Deceleration</i>	REAL	-1.0	减速 (降低电机功率):
			值 > 0: 使用定义的值
			值 = 0: 仅当轴不必减速就能达到最终速度时才允许。
			值 < 0: 使用默认值
<i>Jerk</i>	REAL	-1.0	冲击:
			值 > 0: 使用定义的值
			值 = 0: 使用梯形运动曲线
			值 < 0: 使用默认值
<i>Direction</i>	INT	0	预置方向:
			值 = 0: S7T Config 中的默认设置 *
			值 = 1: 正向旋转
			值 = 2: 不允许
			值 = 3: 反向旋转
			值 = 4: 当前旋转方向 (上次使用的旋转方向)

参数	数据类型	初始值	说明	
<i>Current</i>	BOOL	<i>FALSE</i>	保持当前速度:	
			<i>FALSE</i> :	禁用“保持当前速度”
			<i>TRUE</i> :	保持当前速度和方向。 例如，用于终止同步操作。 输入参数 <i>Velocity</i> 被忽略。 当驱动器以当前速度恢复运行时，输出 <i>InVelocity</i> 将返回值 <i>TRUE</i> 。
<i>PositionControl</i>	BOOL	<i>TRUE</i>	<i>TRUE</i> :	位置控制模式
			<i>FALSE</i> :	速度控制模式
			使用输入参数 <i>PositionControl</i> 在轴中切换位置控制模式/速度控制模式。 输入参数 <i>PositionControl</i> 在速度控制轴中被忽略。 要允许该轴用作引导轴，必须将其设置为在位置控制模式下操作。 您可以从工艺 DB 的 <i>Statusword.SpeedMode</i> 读取当前活动的模式。 将保持到速度控制模式的转换。 可以在 <i>PositionControl = TRUE</i> 时通过新的 <i>MC_MoveVelocity</i> 命令和其它任何运动命令（“MC_Halt”和“MC_Stop”除外）来终止速度控制模式。 通过设置输入参数 <i>Distance = 0.0</i> ，可以在没有运动处于活动状态的情况下，使用 <i>MC_MoveRelative</i> 命令来停止速度控制模式。	
<i>Mode</i>	INT	<i>0</i>	超驰模式:	
			值 = <i>0</i> :	超驰运动: 当前运动被取消
			值 = <i>1</i> :	添加运动: 运动命令被写入命令缓冲区。 轴将在运动跳转时停止
<i>DoneFlag</i>	INT	<i>0</i>	在 MCDevice DB 中生成 DoneFlag 当达到最终速度时，将对 DoneFlag 值取反。	
* 建议不要通过 S7T Config 默认值中的“基于速度设定值的符号来确定”设置来选择方向，因为输入参数 <i>Velocity</i> 不支持负的速度设定值。 如果 S7T Config 中的组态出现故障，则轴将始终沿正向移动。				

输出参数（状态输出）

参数	数据类型	初始值	说明	
<i>InVelocity</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 已达到在输入参数 <i>Velocity</i> 中定义的速度，或该速度保持不变。	
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令	
<i>CommandAborted</i>	BOOL	<i>FALSE</i>	<i>TRUE</i>: 命令由其它命令或因其执行过程中的错误而取消。如果工艺 DB 的 <i>ErrorStatus</i> 中未显示错误，则该命令由后续命令取消。 如果工艺 DB 的变量 <i>ErrorStatus</i> 中指示有错误，则命令执行期间影响该工艺对象的错误将导致该命令中止。 请遵守下列信息。	
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> :	命令启动时 有错误 。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
			<i>FALSE</i> :	命令启动时 没有错误 。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。	

说明

在速度设定值为零 (*Velocity* = 0.0) 时:

InVelocity 在轴达到停止状态时进行设置，并且只要 *Execute* = 1 就保留这种设置。设置了 *InVelocity* 后，命令执行完成，即 *Busy* = *FALSE*，并且该工艺功能既不能报告 *CommandAborted*，也不能报告 *Error*。

位置控制轴的速度控制操作

无论轴处于停止状态还是运动状态，都可以设置“位置控制运动”和“速度控制运动”之间的跳转。

您可以使用输入参数 *PositionControl* 设置位置控制轴的位置控制模式或速度控制模式。

在速度预设值为 0 的速度控制模式下，轴可以立即减速，而不管是否出现了更大的跟随误差（例如因为扭矩限制）。

软件限位开关监视仍处于活动状态。

说明

将保持速度控制操作，直至以下某个定位命令启动：

- "MC_MoveAbsolute"
- "MC_MoveRelative"
- "MC_MoveAdditive"
- ""MC_MoveVelocity"（当 *PositionControl* = *TRUE* 时）
- "MC_MoveToEndPos"
- "MC_GearIn"
- "MC_CamIn"

速度控制处于活动状态时的限制：

- 无法回原点（“MC_Home”）
 - 轴必须在位置控制模式下才能用作采用设定值耦合的引导轴！ 当 *PositionControl* = *FALSE* 时，轴的设定值位置保持不变。
-

说明

速度控制模式下软件限位开关的作用取决于 S7T Config 的 轴 (Axis) > 限制 (Limitations) 中“位置和速度”(Position and velocity) 选项卡的“软件限位开关的作用”(Effect of the software limit switches) 下拉列表中的设置。

- 设置“仅当在位置控制模式下运动时才在软件限位开关处停止”(Stop at software limit switch only when moving in position-controlled mode)

通过该设置，用户程序可以控制轴在速度控制模式下忽略软件限位开关运行。仅当将软件限位开关取消激活时才能进行后续退回。

如果用户程序不允许进行该取消激活操作，则您只能通过循环断电/通电来使轴返回位置控制模式。然后您便可以退回轴。

- 设置“所有运行模式下在软件限位开关处停止并强制使用位置控制模式”(Stop at software limit switch in all modes of operation and force position-controlled mode)

通过该设置，即使在速度控制模式下轴也会在软件限位开关处停止。轴将在第一步中更改为位置控制模式。无需进行退回。

MC_MoveVelocity — 示例 — “超驰运动”

适用于固件版本为 V3.1.x 或更高版本的集成工艺

以下信号图显示了 (*Mode* = 0) 工艺功能“MC_MoveVelocity”的超驰特性。

阶段“a”

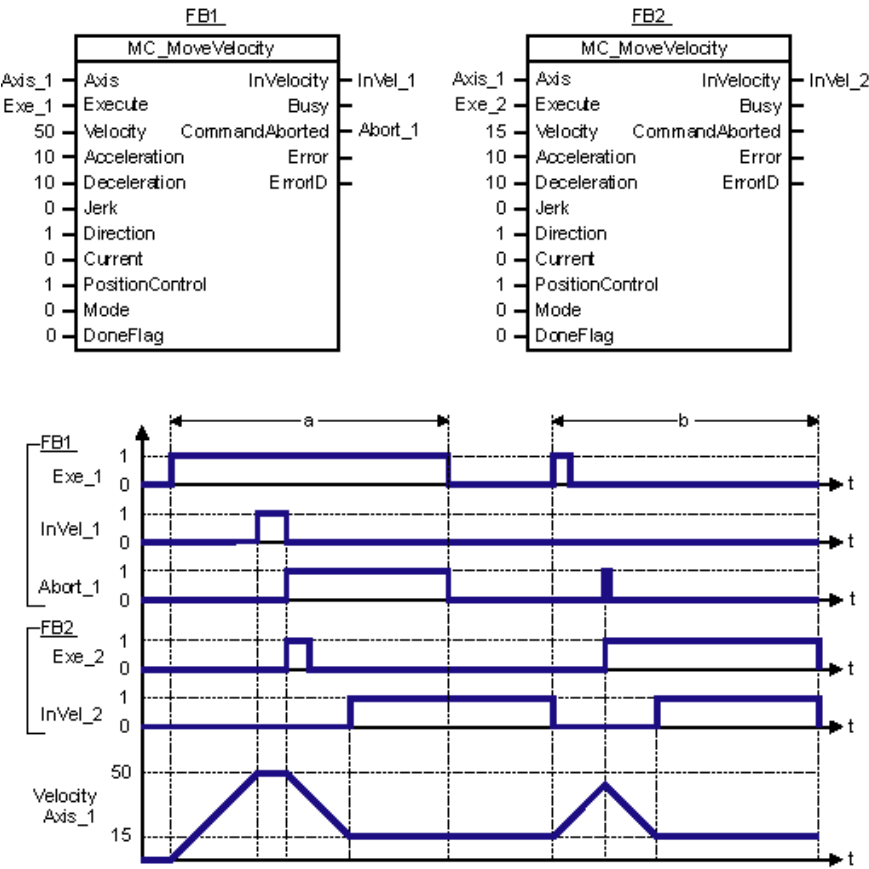
FB1 输入参数 *Execute* (*Exe_1*) 的第一个上升沿用于以速度 50 启动轴运动 (*Axis_1*)。在 *InVel_1* 中报告最终速度。

FB2 输入参数 *Execute* (*Exe_2*) 的上升沿将超驰当前运动。超驰操作报告给 *Abort_1*。只要在 FB1 中设置了 *Exe_1*，*Abort_1* 就将保留设置。超驰后，轴将以速度 15 运动 (*Axis_1*)。

阶段“b”

输入参数 *Execute* (*Exe_1*) 的另一个上升沿将超驰当前运动并将速度重设为 50。在轴的速度达到 50 之前，FB2 中的另一个命令 (*Exe_2* 的上升沿) 将超驰该运动。超驰后，轴将以速度 15 运动 (*Axis_1*)。在 *InVel_2* 中报告最终速度。

由于仅在阶段“b”简单地设置参数 *Exe_1*；因此仅在一个周期内返回 *Abort_1*。



MC_MoveVelocity — 示例 — “添加运动”

适用于固件版本为 V3.1.x 或更高版本的集成工艺

以下信号图显示了工艺功能“MC_MoveVelocity”(Mode = 1) 的“添加”特性。

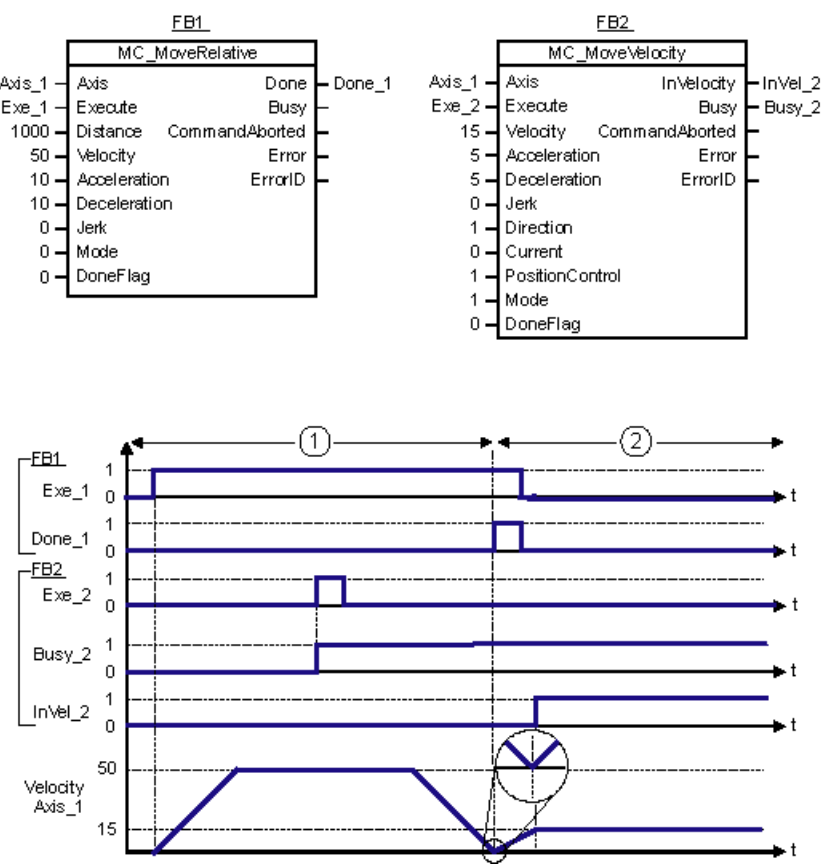
阶段 — 相对定位

相对定位通过 FB1 的输入参数 *Execute (Exe_1)* 的上升沿启动。轴 (*Axis_1*) 将按照设置的加速斜坡 (*Acceleration = 10*) 加速到最终速度。在轴达到目标位置 (*Distance = 1000*) 之前，输出 MC_MoveVelocity 命令 (FB2 的上升沿 *Exe_2*)。

如果在“MC_MoveVelocity”中设置了 *Mode = 1*，则当前定位命令不会取消。
“MC_MoveRelative”通过 *Deceleration = 10* 让轴执行减速运动，然后根据启动的命令逼近目标位置。通过 *Done_1* 报告命令完成。

阶段 — 速度已预设的运动

轴达到 MC_MoveRelative 命令定义的目标位置后，将立即添加 MC_MoveVelocity 运动。参数 *InVel_2* 将报告最终速度（存在短暂延时）；轴 (*Axis_1*) 将以设置的速度 15 运动。



MC_MoveVelocity — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8040	轴/外部编码器被禁用，或者设置了错误的模式	缺少运动命令所需的使能信号。 消除并确认所有排队的错误，然后启用相关的轴模式（例如，位置控制）。
8043	参数值非法	涉及数据类型为 <i>REAL</i> 的所有输入参数，或者输入参数 <i>Direction</i> 、 <i>Mode</i> 或 <i>DoneFlag</i> 。
8044	命令不受工艺对象支持	例如，将命令发送至输出凸轮
8045	在当前状态下不允许使用该命令	MC_Stop 命令处于活动状态
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是可能无法通过状态输出对其进行监视。

ErrorID	错误消息	说明/纠正或避免错误的方法
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8055	当前状态下不允许锁存运动	当前轴状态下无法接受锁存命令。原因： - 正在该轴上执行基础同步或叠加同步命令 "MC_Halt"处于活动状态
8056	由于正在向固定挡块运动而取消	轴已运动到固定挡块 (<i>InClamping = TRUE</i>)。 - 新的位置控制 MC_MoveVelocity 命令 (<i>PositionControl = TRUE</i>) 正控制朝固定挡块方向的运动 - 新的 MC_MoveVelocity 命令启用了速度控制 (<i>PositionControl = TRUE</i>) 只有新命令启用以位置控制方式从固定挡块释放轴，它们才会被接受。
8083	DB 不是工艺 DB	通过输入参数 <i>Axis</i> 指定的 DB 未找到或不是工艺 DB。

ErrorID	错误消息	说明/纠正或避免错误的方法
8084	工艺 DB 无效	- 在通过输入参数 <i>Axis</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Axis</i> 更改 DB 号。 - 用户在输入参数 <i>Axis</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management)中删除工艺 DB，然后创建一个新的工艺 DB。
808B	参数值的 REAL 格式无效	数据类型为 REAL 的输入参数的值不符合有效的浮点数格式。 检查输入参数值或背景数据块数据。 不能以浮点格式表示格式无效的值。 它们以十六进制格式 (DW16# ...) 显示。

6.2.11 FB 415 MC_MoveToEndPos — 移动到固定挡块/紧固件

通过调用 FB 415“MC_MoveToEndPos”移动到固定挡块/紧固件

用途

- “MC_MoveToEndPos”工艺功能可以将轴移动到与障碍物接触的位置，并使其保持在该位置，例如行程终点。
- 在输入参数 *Velocity*、*Jerk*、*Acceleration*、*Deceleration* 中定义运动的动态响应。
- 可以在 S7T Config 的轴组态中定义用于检测末端位置的方法和标准（例如跟随误差或扭矩）。
- 如果可以对驱动扭矩进行限制，则该工艺功能的扭矩限制还会影响固定挡块检测设置“使用跟随误差”（在 S7T Config 中）

适用对象

- 定位轴（仅电气实轴）
- 同步轴（仅电气实轴）

先决条件

- 已启用轴进行位置控制操作
- 没有处于活动状态的 MC_Stop 命令
- 必须为轴启用固定挡块检测。

在 S7T Config 的“轴”(Axis) >“限制”(Limiting) >“固定挡块”(Fixed stop) 对话框中，设置固定挡块检测标准“使用跟随误差”或“使用扭矩”。

命令的相互作用

- 在到达固定挡块之前，其它运动命令可以超驰当前命令。
- 如果轴被固定在固定挡块处 (*InClamping* = TRUE)，则系统仅接受将轴从该位置移开的位置控制运动命令。MC_MoveToEndPos 仍处于活动状态，直到驱动器离开夹紧公差窗口。现在，可以输入朝固定挡块方向运动的新运动命令。在夹紧公差窗口内执行的任何用来启动与固定挡块反向的同步命令都会被取消。
- MC_MoveToEndPos 命令只能超驰其它活动的 MC_MoveToEndPos 命令（如果两个命令都设置了相同的有效方向）。

新命令 — 活动的单个命令 (2)

新命令 — 活动的命令 (2)

输入参数

参数	数据类型	初始值	说明
<i>Axis</i>	INT	0	工艺 DB 的编号
<i>Execute</i>	BOOL	FALSE	在上升沿处启动命令
<i>Direction</i>	INT	0	预置方向：
			值 = 0: 默认设置
			值 = 1: 正向旋转
			值 = 2: 不允许
			值 = 3: 反向旋转
			值 = 4: 当前旋转方向 (上次使用的旋转方向)
<i>Torque</i>	REAL	1.0	驱动扭矩限制 (以 [N/m] 为单位)。 条件： 驱动器支持并使用报文 101 到 106，或输入驱动器的最大扭矩。
			值 > 0: 使用定义的值
			值 <= 0: 不允许
<i>Velocity</i>	REAL	0.0	最大速度 (不是总能达到)：
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值
<i>Acceleration</i>	REAL	-1.0	加速 (增加电机功率)：
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值
<i>Deceleration</i>	REAL	-1.0	减速 (降低电机功率)：
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值

参数	数据类型	初始值	说明
<i>Jerk</i>	REAL	<i>-1.0</i>	冲击:
			值 > 0: 使用定义的值
			值 = 0: 使用梯形运动曲线
			值 < 0: 使用默认值
<i>DoneFlag</i>	INT	<i>0</i>	在 MCDevice DB 中生成 DoneFlag

输出参数（状态输出）

参数	数据类型	初始值	说明
<i>InClamping</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 轴已到达固定挡块，现位于“固定挡块检测后的位置公差”内。 位置公差在 S7T Config 中的“限制”(Limits) > “固定挡块”(Fixed stop) 选项卡 > “固定挡块检测后的位置公差”(Position tolerance after fixed stop detection) 参数中设置。
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令
<i>CommandAborted</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令由其它命令或因其执行过程中的错误而取消。 如果工艺 DB 的 <i>ErrorStatus</i> 中未显示错误，则该命令由后续命令取消。 如果工艺 DB 的变量 <i>ErrorStatus</i> 中指示有错误，则命令执行期间影响该工艺对象的错误将导致该命令中止。
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令启动时有错误。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
			<i>FALSE</i> : 命令启动时没有错误。
<i>ErrorID</i>	WORD	<i>0</i>	输出参数 <i>Error</i> 的 ErrorID。

扭矩减小的高精度

对于集成工艺的固件版本 V4.1.x 或更高版本，扭矩减小的高精度可用于 SINAMICS 和 SIMODRIVE 驱动器。可以使用工艺功能“MC_WriteParameter”，通过参数 4502 或

S7T Config 中的组态参数

TypeOfAxis.SetPointDriverInfo.DriveData.torqueReductionGranularity 设置扭矩减小的高精度。必须根据下表调整驱动器参数：精细粒度的扭矩减小

精度	ENUM 值（工艺参数 4502）	SIMODRIVE	SINAMICS
1/100	基本	<i>P0881</i> = 4000h (16384) (默认)	<i>P1544</i> = 4000h (16384)
1/16384	标准 (默认)	<i>P0881</i> = 64h (100)	<i>P1544</i> = 64h (100) (默认)

说明

在 S7T Config 轴组态中，如果在“轴”(Axis) > “限制”(Limits) 对话框的“固定挡块”(Fixed stop) 选项卡上设置了“使用跟随误差”(use following error)，则当使用该工艺功能时将禁用跟随误差监视。

如果固定挡块断开，跟随误差将根据在驱动器上组态的限制而减小。在输入参数中定义的动态值和在该工艺对象中设置的限制都不会影响该操作。

在使用“MC_Stop”或“MC_Halt”停止轴之前，请务必降低跟随误差。

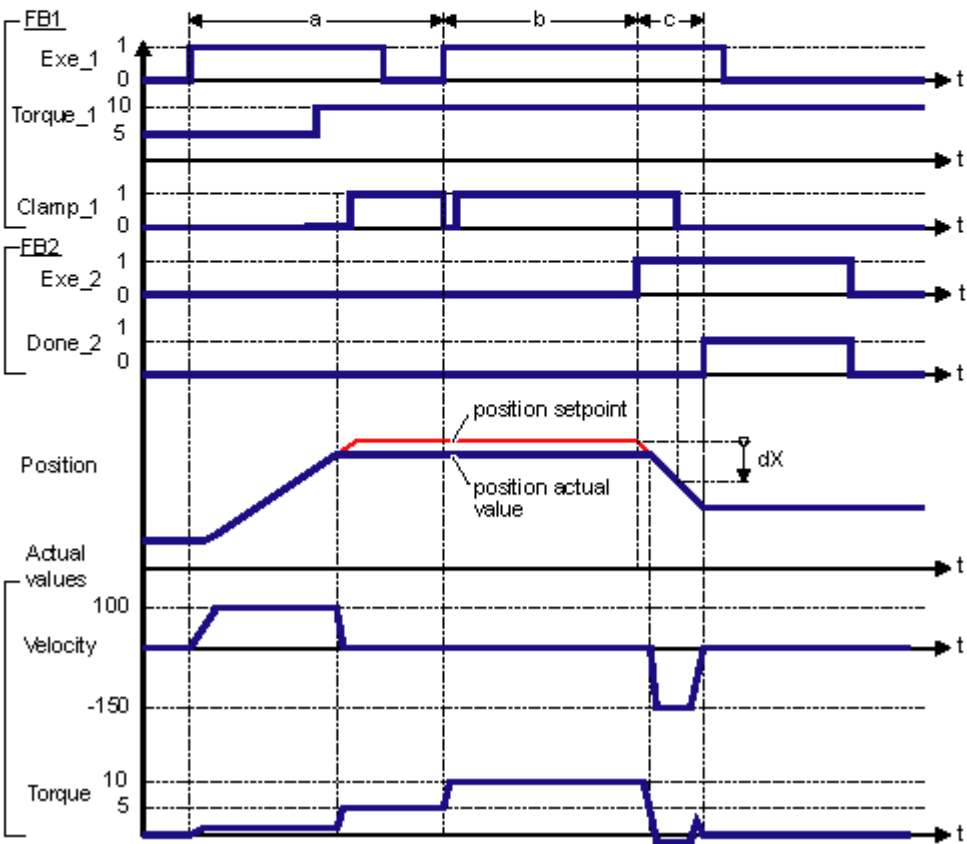
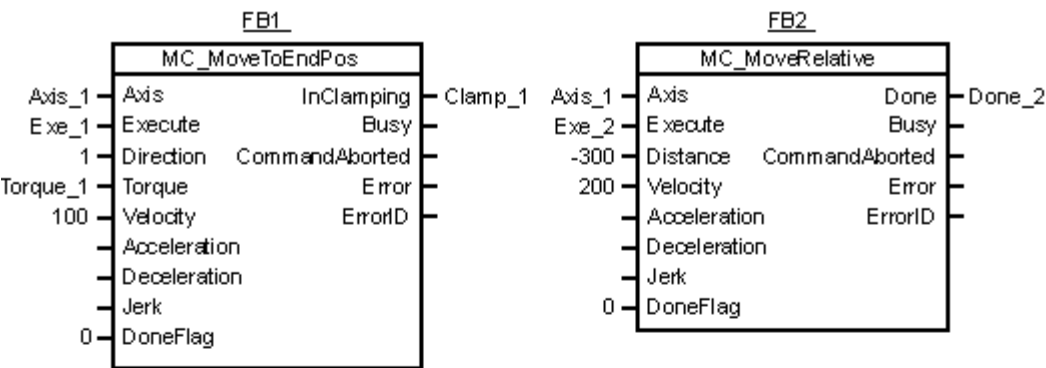
动态参数 *Velocity*

说明

出于安全考虑，初始速度值设置为 0.0。在实际情况中该值是无效的，并将导致错误。请定义一个适合的值。

MC_MoveToEndPos — 示例

- a) 以减小的扭矩将轴移动到固定终端挡块。根据位置设定值 > 实际位置值来产生跟随误差以保持扭矩。
- b) 当新命令将轴朝同一方向移动时，紧固扭矩将加倍。
- c) 通过启动一个可以从紧固位置释放轴的“MC_MoveRelative”命令来终止紧固。一旦位置设定值超出夹紧公差窗口将立即终止紧固件。



MC_MoveToEndPos — ErrorID

ErrorID	警告消息	说明/纠正或避免错误的方法
0000	没有错误	-
0029	“MC_MoveToEndPos”在相反方向上被忽略	启动了一个起反向作用的新 MC_MoveToEndPos 命令以超越活动的 MC_MoveToEndPos 命令。这是不允许的。 有关终止活动的 MC_MoveToEndPos 命令的有效命令和条件的信息，请参考“MC_MoveToEndPos”文档。

ErrorID	错误消息	说明/纠正或避免错误的方法
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： “MC_CamSectorAdd” “MC_ReadPeriphery” “MC_WritePeriphery” “MC_ReadRecord” “MC_WriteRecord” “MC_ReadDriveParameter” “MC_WriteDriveParameter” “MC_ReadCamTrackData” “MC_WriteCamTrackData” 在同一个周期内调用这些工艺功能，直到输出参数 <i>Done</i> , <i>CommandAborted</i> 或 <i>Error</i> 中有一个为 <i>TRUE</i> 。验证程序是否包含不必要的（多余的）命令。
8040	轴/外部编码器被禁用，或者设置了错误的模式	缺少运动命令所需的使能信号。 消除并确认所有排队的错误，然后启用相关的轴模式（例如，位置控制）。

ErrorID	错误消息	说明/纠正或避免错误的方法
8043	参数值非法	- 涉及数据类型为 REAL 的所有输入参数，或者 <i>Direction</i> 或 <i>DoneFlag</i> 输入参数。 - 在 S7T Config 中没有为轴启用固定挡块检测 - 要么没有使用介于 101 到 106 之间的消息帧，要么输入参数 <i>Torque</i> 的值与最大驱动扭矩不一致。
8044	命令不受工艺对象支持	例如，将命令发送至输出凸轮。
8045	在当前状态下不允许使用该命令	- MC_Stop 命令处于活动状态 - MC_MoveToEndPos 命令和紧固处于活动状态，又启动“MC_MoveToEndPos”的一个新实例来反转运动方向。这是不允许的。
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是可能无法通过状态输出对其进行监视。
804D	轴类型无效	该轴类型不支持“MC_MoveToEndPos”工艺功能。 - 该轴是一个虚拟轴 - 该轴是一个液压轴
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。

ErrorID	错误消息	说明/纠正或避免错误的方法
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： • 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 • 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8056	由于正在向固定挡块运动而取消	轴已运动到固定挡块 (<i>InClamping = TRUE</i>)。只有新命令从固定挡块释放轴，它们才被接受。 已启动的命令不符合该条件。
8083	DB 不是工艺 DB	通过输入参数 <i>Axis</i> 指定的 DB 未找到或不是工艺 DB。
8084	工艺 DB 无效	• 在通过输入参数 <i>Axis</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Axis</i> 更改 DB 号。 • 用户在工艺 DB（通过输入参数 <i>Axis</i> 定义）中输入的数据无效。 在“工艺 <i>Objects</i> 管理”中删除该工艺 DB，然后重新创建一个工艺 DB。
808B	参数值的 REAL 格式无效	数据类型为 REAL 的输入参数的值不符合有效的浮点数格式。 检查输入参数值或背景数据块数据。不能以浮点格式表示格式无效的值。它们以十六进制格式 (DW16# ...) 显示。

6.2.12 FB437 MC_SetTorqueLimit — 启用/禁用扭矩限制

使用 FB 437“MC_SetTorqueLimit”进行扭矩限制

用途

- “MC_SetTorqueLimit”工艺功能用于启用/禁用扭矩限制。
- 对于集成工艺固件版本 V4.1.x 和更高版本，即使通过“MC-Power”工艺功能删除使能后，仍会保持扭矩限制。

适用对象

- 速度控制轴
- 定位轴
- 同步轴

先决条件

- 没有处于活动状态的 MC_Stop 命令
- 启用轴进行速度控制操作或位置控制操作
- 必须正确设置轴的参考扭矩才能保持指定的扭矩。参考扭矩的默认值是 100 N/m。
- 驱动器必须支持扭矩减小，并且必须通过报文 101 到 106 中的一个进行操作。

命令的相互作用

新命令 — 活动的单个命令 (3)

输入参数

参数	数据类型	初始值	说明
<i>Axis</i>	INT	0	工艺 DB 的编号
<i>Execute</i>	BOOL	<i>FALSE</i>	在上升沿处启动命令
<i>EnableLimit</i>	BOOL	<i>TRUE</i>	启用/禁用限制
<i>MaxTorque</i>	REAL	3.2	最大有效驱动扭矩。*
			定义扭矩（以 [N/m] 为单位）。
			值 = 0: 使用定义的值
			值 < 0: 不允许
* 在轴组态中务必始终设置“最大驱动扭矩”参数（在专家列表： <i>TypeOfAxis.SetPointDriverInfo.DriveData.maxTorque</i> 中）。			

输出参数（状态输出）

参数	数据类型	初始值	说明
<i>Done</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 已成功执行命令
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令
<i>CommandAborted</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令由其它命令或因其执行过程中的错误而取消。 如果工艺 DB 的 <i>ErrorStatus</i> 中未显示错误，则该命令由后续命令取消。 如果工艺 DB 的变量 <i>ErrorStatus</i> 中指示有错误，则命令执行期间影响该工艺对象的错误将导致该命令中止。
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令启动时有错误。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
			<i>FALSE</i> : 命令启动时没有错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。

扭矩减小的高精度

对于集成工艺的固件版本 V4.1.x 或更高版本，扭矩减小的高精度可用于 SINAMICS 和 SIMODRIVE 驱动器。可以使用工艺功能“MC_WriteParameter”，通过参数 4502 或 S7T Config 中的组态参数 *TypeOfAxis.SetPointDriverInfo.DriveData.torqueReductionGranularity* 设置扭矩减小的高精度。必须根据下表调整驱动器参数：精细粒度的扭矩减小

精度	ENUM 值（工艺参数 4502）	SIMODRIVE	SINAMICS
1/100	基本	<i>P0881</i> = 4000h (16384) (默认)	<i>P1544</i> = 4000h (16384)
1/16384	标准 (默认)	<i>P0881</i> = 64h (100)	<i>P1544</i> = 64h (100) (默认)

说明

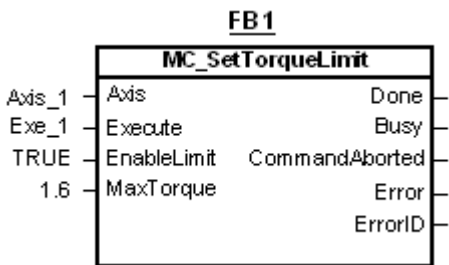
该工艺功能正在使用时，将禁用跟随误差监视。当扭矩增大或反向扭矩减小时，将根据组态的驱动限制消除产生的所有跟随误差。在输入参数中定义的动态值和在该工艺对象中设置的限制都不会影响该操作。

在使用“MC_Stop”或“MC_Halt”停止轴之前，请务必降低跟随误差。最好使用工艺功能“MC_MoveVelocity”（具有设置：*PositionControl* = FALSE 和 *Velocity* = 0.0 ）停止无位置控制的轴。

MC_SetTorqueLimit — 示例

默认情况下，扭矩限制设置为值 3.2 N/m（组态参数 *TypeOfAxis.SetPointDriverInfo.maxTorque* ）。

在以下示例中，扭矩限制减小为 1.6 N/m。



MC_SetTorqueLimit — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8043	参数值非法	涉及 <i>MaxTorque</i> 输入参数（值必须大于或等于零）。
8044	命令不受工艺对象支持	例如，将命令发送至输出凸轮
8045	在当前状态下不允许使用该命令	MC_Stop 命令处于活动状态
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是可能无法通过状态输出对其进行监视。
804D	轴类型无效	该轴类型不支持"MC_SetTorqueLimit"工艺功能。 - 该轴是一个虚拟轴 - 该轴是一个液压轴

ErrorID	错误消息	说明/纠正或避免错误的方法
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8056	由于正在向固定挡块运动而取消	MC_MoveToEndPos 命令处于活动状态。
8083	DB 不是工艺 DB	通过输入参数 <i>Axis</i> 指定的 DB 未找到或不是工艺 DB。
8084	工艺 DB 无效	- 在通过输入参数 <i>Axis</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Axis</i> 更改 DB 号。 - 用户在输入参数 <i>Axis</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management) 中删除工艺 DB，然后创建一个新的工艺 DB。
808B	参数值的 REAL 格式无效	数据类型为 REAL 的输入参数的值不符合有效的浮点数格式。 检查输入参数值或背景数据块数据。不能以浮点格式表示格式无效的值。它们以十六进制格式 (DW16# ...) 显示。

6.2.13 FB439 MC_SetCharacteristic — 激活阀控曲线

通过调用 FB 439“MC_SetCharacteristic”激活阀控曲线

固件版本为 V3.2.x 或更高版本的集成工艺支持该功能

用途

- 激活 Q 阀的阀控曲线（速度阀）
- 也可以在液压轴运动时更改阀控曲线。可选的控制信号补偿将受到限制
- 通过输入参数 *Execute* 的上升沿重复调用该工艺功能将取消上一个命令。

适用对象

- 液压速度控制轴
- 液压定位轴
- 液压同步轴

先决条件

- 必须将选定的轴组态为液压轴
- 曲线已映射到一个凸轮盘，并在 S7T Config 中作为可选阀控曲线进行组态。

命令的相互作用

新命令 — 活动的单个命令 (3)

输入参数

参数	数据类型	初始值	说明
<i>Axis</i>	INT	0	轴工艺 DB 的编号
<i>CamTable</i>	INT	0	凸轮盘工艺 DB 的编号
<i>Execute</i>	BOOL	<i>FALSE</i>	在上升沿处激活曲线
<i>Mode</i>	INT	0	曲线类型 值 = 0 Q 阀控曲线
<i>DoneFlag</i>	INT	0	在 MCDevice DB 中生成 DoneFlag

输出参数（状态输出）

参数	数据类型	初始值	含义
<i>Done</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 曲线已激活
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令启动时有错误。未执行该命令。有关原因的信息, 请参考 <i>ErrorID</i> 。
			<i>FALSE</i> : 命令启动时没有错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 ErrorID。

控制信号补偿的限制

转换时, 输出信号补偿在默认情况下限制为 100%/s, 这意味着 100% 的输出信号跳转在 1 秒内得到补偿。

可以在液压轴的专家列表中组态控制信号补偿:

“系统变量”变量 *userdefaultqfaxis.maxderivate.qoutput*

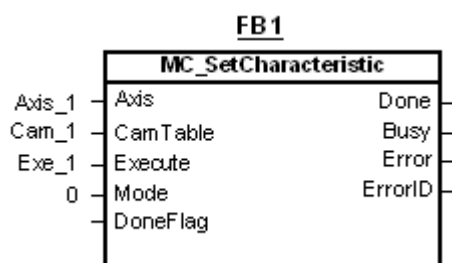
说明

只有满足以下条件才能通过调用工艺功能“MC_Power”启用液压轴:

- 必须为该液压轴分配一个凸轮盘以作为阀控曲线。
- 必须由“MC_SetCharacteristic”工艺功能激活该阀控曲线。

MC_SetCharacteristic — 示例

特性阀控曲线存储在凸轮盘 (Cam_1) 中。调用时，用于在液压轴确定运动命令的控制信号。



MC_SetCharacteristic — ErrorID

受固件版本为 V3.2.x 的集成工艺的支持

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8043	参数值无效	涉及输入参数 <i>Mode</i> 和 <i>DoneFlag</i> 。

ErrorID	错误消息	说明/纠正或避免错误的方法
8044	命令不受工艺对象支持	涉及输入参数 <i>Axis</i> 和 <i>CamTable</i> 。 <i>Axis</i> 不是轴或不是液压轴 <i>CamTable</i> 不是凸轮盘
804A	缺少所需的对象连接	未在 S7T Config 的轴 (Axis) > 曲线 (Profile) 对话框中将组态的凸轮盘作为阀控曲线分配给轴。
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是可能无法通过状态输出对其进行监视。
804D	轴类型无效	该工艺功能只能应用于液压轴。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8063	未插补凸轮	未插补指定的阀控曲线。

ErrorID	错误消息	说明/纠正或避免错误的方法
8083	DB 不是工艺 DB	通过输入参数 <i>Axis</i> 指定的 DB 未找到或不是工艺 DB。
8084	工艺 DB 无效	- 在通过输入参数 <i>Axis</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Axis</i> 更改 DB 号。 - 用户在输入参数 <i>Axis</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management) 中删除工艺 DB，然后创建一个新的工艺 DB。

6.3 工艺功能 — 齿轮传动/凸轮传动

6.3.1 FB420 MC_GearIn — 启动齿轮传动

使用 FB 420“MC_GearIn”启动齿轮传动

适用于固件版本为 V3.2.x 或更高版本的集成工艺

本部分介绍了固件版本为 V3.0.x 和 V3.1.x 的应用。

用途

- “MC_GearIn”工艺功能用于在引导轴（主轴）和跟随轴（从轴）之间启动基本齿轮传动
- 可以在动态参数 *Velocity*, *Jerk*, *Acceleration*, *Deceleration* 中定义跟随轴同步期间的动态响应。
- 传动比定义为输入参数中的两个整数的比率函数（分子/分母）
- 同步与起始位置（达到同步时的随机位置值）可以是相对的，也可以是绝对的
- 可以通过请求一个新的 MC_GearIn 命令在系统处于 RUN 时更改传动比。该操作无需停止引导轴或跟随轴。将根据指定的加速或减速值执行跳转
- 引导轴处于停止状态或运动状态时，均可以启动该功能。

适用对象

- 同步轴

先决条件

- 将引导轴组态为外部编码器、定位轴或同步轴
- 将跟随轴组态为同步轴
- 设置引导轴以用作跟随轴的基础同步对象的主值
- 启用跟随轴以进行位置控制操作
- 跟随轴上没有正在执行的 **MC_Stop** 命令。

命令的相互作用

新命令 — 活动的单个命令 (1)

新命令 — 活动的命令 (1)

输入参数

参数	数据类型	初始值	说明
<i>Master</i>	INT	0	引导轴工艺 DB 的编号
<i>Slave</i>	INT	0	跟随轴工艺 DB 的编号
<i>Execute</i>	BOOL	FALSE	在上升沿处启动命令
<i>RatioNumerator</i>	DINT	1	传动比分子（不允许值 0）
<i>RatioDenominator</i>	DINT	1	传动比分母（不允许值 0）
<i>Velocity</i>	REAL	-1.0	跳转到齿轮传动时的最大速度（仅当 <i>Mode</i> = 1 时有效）：
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值
<i>Acceleration</i>	REAL	-1.0	跳转到齿轮传动时的加速度（仅当 <i>Mode</i> = 1 时有效）：
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值
<i>Deceleration</i>	REAL	-1.0	切换为齿轮传动时的减速度（仅当 <i>Mode</i> = 1 时有效）：
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值

参数	数据类型	初始值	说明
<i>Jerk</i>	REAL	-1.0	跳转到齿轮传动时的冲击（仅当 <i>Mode</i> = 1 时有效）：
			值 > 0: 使用定义的值
			值 = 0: 使用梯形运动曲线
			值 < 0: 使用默认值
<i>PhaseShift</i>	REAL	0.0	达到同步后的相位偏移 如果 <i>Absolute</i> = <i>TRUE</i> ，则当达到同步操作时，相位偏移是绝对的。 如果 <i>Absolute</i> = <i>FALSE</i> ，则指定的相位偏移将添加到由相对关系确定的相位偏移。
			值 > 0: 使用定义的值
			值 = 0: 无相位偏移
			值 < 0: 不允许
<i>Absolute</i>	BOOL	<i>TRUE</i>	<i>TRUE</i> : 绝对齿轮传动 适用于固件版本不高于 V3.2.x 的集成工艺： 在 <i>Jerk</i> 中预设的参数被忽略；始终使用梯形运动曲线（不受限制的冲击）
			<i>FALSE</i> : 相对齿轮传动
<i>Mode</i>	INT	0	同步模式/耦合模式：
			值 = 0: 使用相应同步对象的默认值
			值 = 1: 基于时间的同步： 根据 <i>Velocity</i> , <i>Jerk</i> , <i>Acceleration</i> , <i>Deceleration</i> 输入参数立即同步（仅出于可能的兼容性原因而使用 — 不用于新项目）。
			值 = 2: 基于时间的同步： 根据输入参数 <i>Velocity</i> , <i>Jerk</i> , <i>Acceleration</i> , <i>Deceleration</i> 立即同步。 程序还将评估 <i>userdefault.gearingsetting.synchronizingdirection</i> 系统变量。 可以在 S7T Config 的专家列表或通过“MC_WriteParameter”工艺功能来设置该系统变量。

6.3 工艺功能 — 齿轮传动/凸轮传动

参数	数据类型	初始值	说明
<i>DoneFlag</i>	INT	0	在 MCDevice DB 中生成 DoneFlag 当到达齿轮传动时，将对 DoneFlag 值取反。

输出参数（状态输出）

参数	数据类型	初始值	说明
<i>InGear</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 实现基本齿轮传动
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令
<i>CommandAborted</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令由其它命令或因其执行过程中的错误而取消。 如果工艺 DB 的 <i>ErrorStatus</i> 中未显示错误，则该命令由后续命令取消。 如果工艺 DB 的变量 <i>ErrorStatus</i> 中指示有错误，则命令执行期间影响该工艺对象的错误将导致该命令中止。
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令启动时有错误。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
			<i>FALSE</i> : 命令启动时没有错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 ErrorID。

输入参数 *Mode* = 1

当使用**相对**同步操作时，同步（补偿）将立即启动。达到同步时，将在引导轴和跟随轴之间设置一个随机偏移。

当使用**绝对**同步操作时，同步启动时可能会发生延迟。达到同步时，引导轴的实际位置值等于跟随轴的实际位置值（偏移 = 0）。

在传动比为 1:1 且模数长度相等或者使用无数轴时，偏移值在同步操作期间保持为零。

模数轴的同步方向：如果 *Mode* = 1，

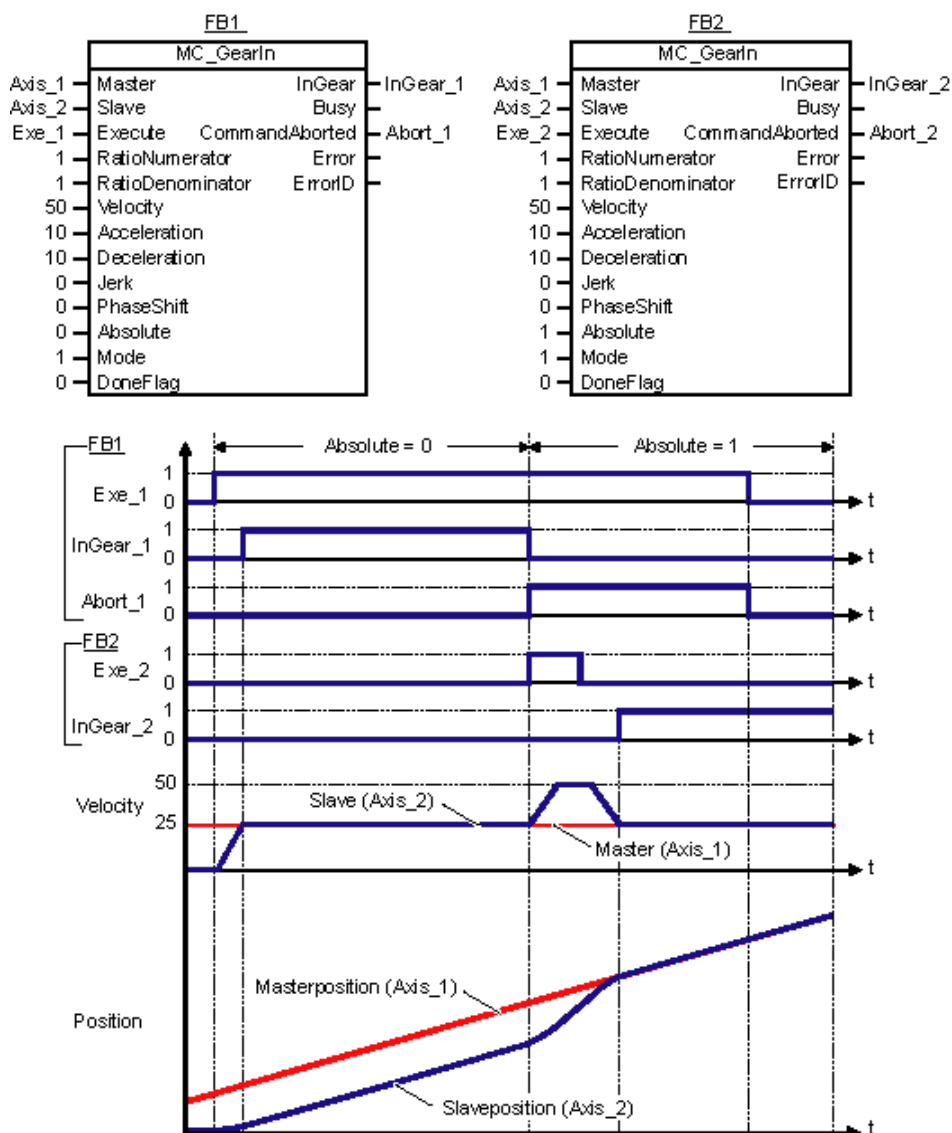
则“兼容性模式”处于活动状态，而不管 S7T Config 中的设置如何：

- 齿轮传动：
跟随轴在正向上同步

MC_GearIn — 示例 — “相对/绝对同步”

同步后基于预设动态参数的绝对/相对基本同步

在第一步中，当相对同步处于活动状态时 (*Absolute = FALSE*)，*Axis_2* 与 *Axis_1* 同步。两个轴之间将产生一个随机偏移。*Absolute = TRUE* 时重启同步。这将消除轴之间的所有偏移。



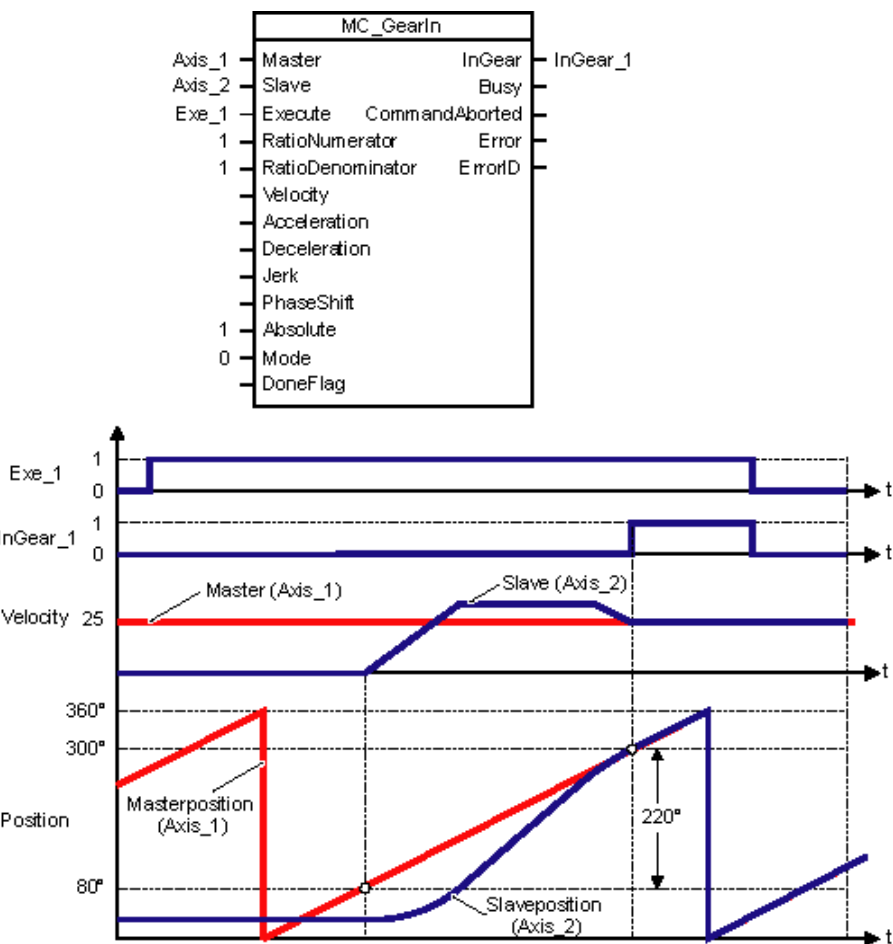
MC_GearIn — 示例 — “基于组态数据的同步”

如果 *Mode* = 0, 则基于轴组态数据使轴同步。 在本示例中, 我们已组态了以下内容:

- 同步: "引导轴的默认同步位置"
- 位置参考: "从同步位置进行同步"
- 同步位置主设定值: 80.0
- 曲线设置: "特定于引导轴的同步曲线"
- 同步 长度: 220.0

当引导轴在 80 °位置通过同步位置后, 同步启动。 当引导轴到达 220 °时, 该操作结束。

动态参数 *Velocity*, *Acceleration*, *Deceleration*, *Jerk* 是不相关的。



MC_GearIn — 示例 — “相移”

以下示例显示了使用相移和不使用相移的绝对同步之间的差别（基于信号图）。

不使用相移的绝对同步

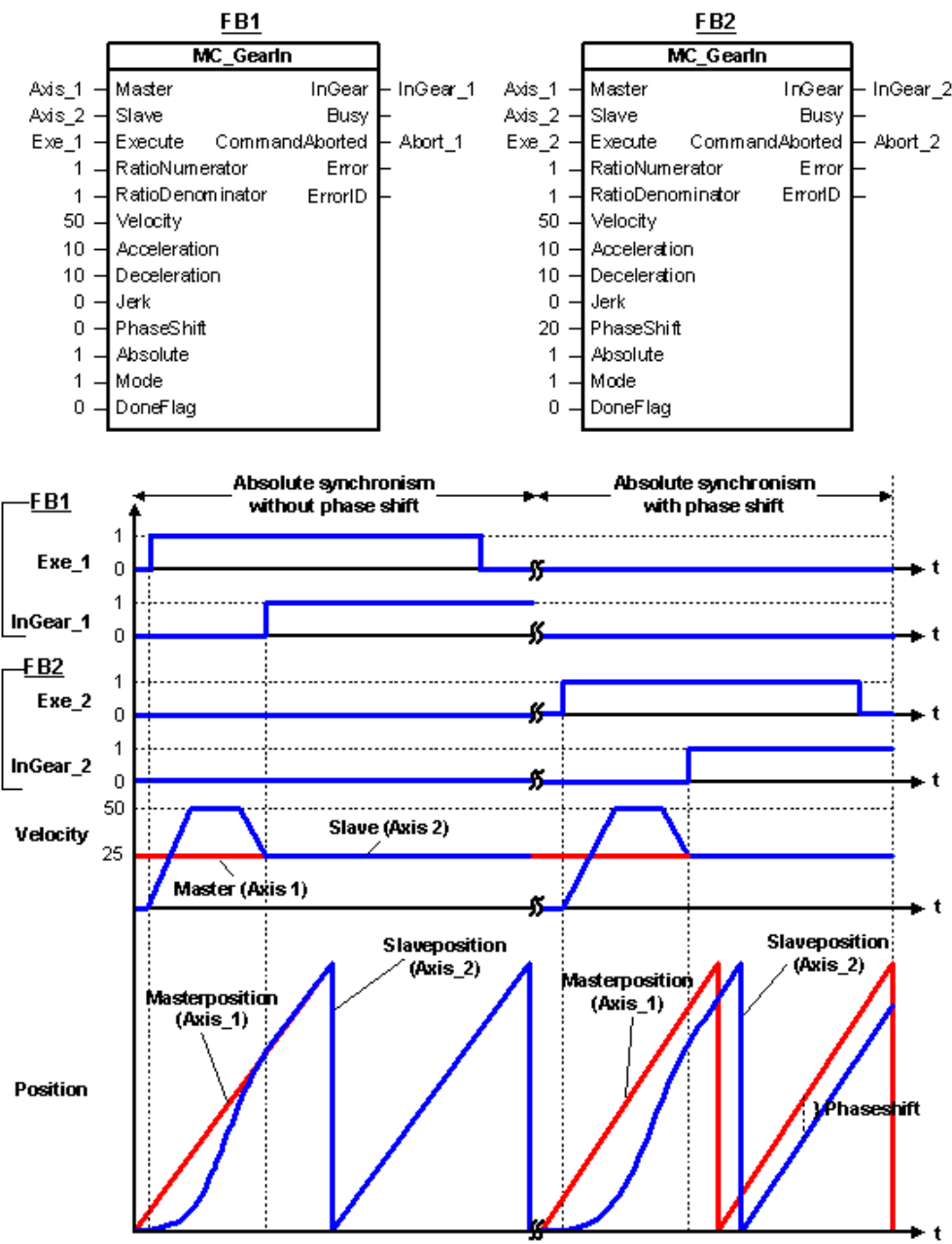
信号图的起始位置：

- 主位置 ($Axis\_1$) = 0
- 从位置 ($Axis\_2$) = 0

Exe_1 的上升沿可以启动不使用相移的绝对同步。短暂延迟后， $InGear\_1$ 将报告已达到绝对同步。引导轴位置 ($Axis\_1$) 和跟随轴位置 ($Axis\_2$) 之间达到了绝对同步，即它们的位置相同。

使用相移的绝对同步

显示的信号配置文件适用于与不使用相移的信号图相同的起始位置。
Exe_2 的上升沿可以启动使用相移的绝对同步 (*PhaseShift* = 20)。 短暂延迟后，*InGear_2* 将报告已达到绝对同步。 应用引导轴位置和跟随轴位置之间的指定相移。



MC_GearIn — ErrorID

适用于固件版本为 V3.2.x 的集成工艺

本部分介绍了固件版本为 V3.0.x 和 V3.1.x 的应用。

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8022	轴/外部编码器中没有实际值	例如，编码器或数据总线未准备就绪
8040	轴/外部编码器被禁用，或者设置了错误的模式	缺少用于启用跟随轴的命令所需的信号。
8043	参数值或默认值无效	涉及数据类型为 <i>REAL</i> 的所有输入参数，或者输入参数 <i>RatioNumerator</i>、<i>RatioDenominator</i>、<i>Mode</i> 或 <i>DoneFlag</i>。 S7T Config 中可能存在组态错误，通常是因为存在同步长度为 0 的组态。

ErrorID	错误消息	说明/纠正或避免错误的方法
8044	命令不受工艺对象支持	通过输入参数 <i>Master</i> 定义的工艺 DB 必须属于“定位轴”、“同步轴”或“外部编码器”类型。 通过输入参数 <i>Slave</i> 定义的工艺 DB 必须属于“同步轴”类型。
8045	在当前状态下不允许使用该命令	MC_Stop 命令处于活动状态
804A	缺少所需的对象连接	在组态基础同步对象时未选择输入参数 <i>Master</i> 中定义的引导轴。
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是您可能无法通过状态输出对其进行监视。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。

ErrorID	错误消息	说明/纠正或避免错误的方法
8056	由于正在向固定挡块运动而取消	轴已运动到固定挡块 (<i>InClamping = TRUE</i>)。只有新命令从固定挡块释放轴，它们才被接受。 已启动的命令不符合该条件。
807A	主设定值无效	引导轴的实际值无效。
807B	工艺对象的递归互连	同步操作中，在该输入参数中定义的引导轴已作为跟随轴处于活动状态，并且将从输入参数中定义的跟随轴直接或间接获得主设定值。
8083	DB 不是工艺 DB	通过输入参数 <i>Master</i> 或 <i>Slave</i> 指定的 DB 未找到或不是工艺 DB。
8084	工艺 DB 无效	- 在通过输入参数 <i>Master</i> 定义的工艺 DB 的控制器中不存在工艺对象。<i>Slave</i> 将当前工艺下载到目标系统，或通过输入参数更改 DB 号。 - 用户在工艺 DB（通过输入参数 <i>Master</i> 或 <i>Slave</i> 定义）中输入的数据无效。 在“工艺对象管理”(Technology Objects Management) 中删除该工艺 DB，然后创建一个新的工艺 DB。
808B	参数值的 REAL 格式无效	数据类型为 REAL 的输入参数的值不符合有效的浮点数格式。 检查输入参数值或背景数据块数据。不能以浮点格式表示格式无效的值。它们以十六进制格式 (DW16# ...) 显示。

6.3.2 FB422 MC_GearOut — 终止齿轮传动

使用 FB 422“MC_GearOut”停止齿轮传动

用途

- “MC_GearOut”工艺功能用于终止基本齿轮传动。该操作不会影响叠加运动或叠加同步。
- 将根据 S7T Config 中同步对象的默认设置停止同步操作。
- 工艺功能在跟随轴的基本齿轮传动停止后终止。

说明

如果关闭过程取决于引导轴和/或跟随轴的位置，请使用该工艺功能。

也可以使用“MC_Halt”、“MC_Stop”、“MC_MoveRelative”、“MC_MoveAdditive”、“MC_MoveAbsolute”、“MC_MoveVelocity”或“MC_MoveToEndPos”工艺功能终止跟随轴的同步操作。

适用对象

- 同步轴

先决条件

- 对于基本齿轮传动，轴必须作为跟随轴（从轴）进行操作。如果没有基本齿轮传动处于活动状态，则该命令将被忽略，并且该工艺功能将报告 *Done = TRUE*。
- 跟随轴上没有 MC_Stop 命令处于执行状态。

命令的相互作用

新命令 — 活动的单个命令 (1)

新命令 — 活动的命令 (1)

输入参数

参数	数据类型	初始值	说明
<i>Slave</i>	INT	0	跟随轴工艺 DB 的编号
<i>Execute</i>	BOOL	<i>FALSE</i>	在上升沿处启动命令

输出参数（状态输出）

参数	数据类型	初始值	说明
<i>Done</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 基本齿轮传动已终止 如果没有基本齿轮传动命令处于活动状态，则输出参数 <i>Done</i> 还将报告 <i>TRUE</i> 。
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令
<i>CommandAborted</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令由其它命令或因其执行过程中的错误而取消。 如果工艺 DB 的 <i>ErrorStatus</i> 中未显示错误，则该命令由后续命令取消。 如果工艺 DB 的变量 <i>ErrorStatus</i> 中指示有错误，则命令执行期间影响该工艺对象的错误将导致该命令中止。
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令启动时有错误。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
			<i>FALSE</i> : 命令启动时没有错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。

说明

将 MC_GearIn 和 MC_GearOut 同步输出到同一工艺对象时可能会取消 MC_GearIn 命令 (*CommandAborted* = *TRUE*)。因此，“MC_GearOut”工艺功能将报告 *Done* = *TRUE*。
请勿启动 MC_GearOut 命令，除非 MC_GearIn 已报告了同步状态 (*InGear* = *TRUE*)。

MC_GearOut — 示例

可以通过调用“MC_GearOut”工艺功能并根据组态的模式停止基本齿轮传动和轴。 在本示例中存在一个相对的基本齿轮传动，它在引导轴和跟随轴之间存在偏移（同步期间就已产生）。传动比为 1:1。跟随轴应在 180 °的位置停止。相应的设置：

去同步：“跟随轴的去同步位置的规范”

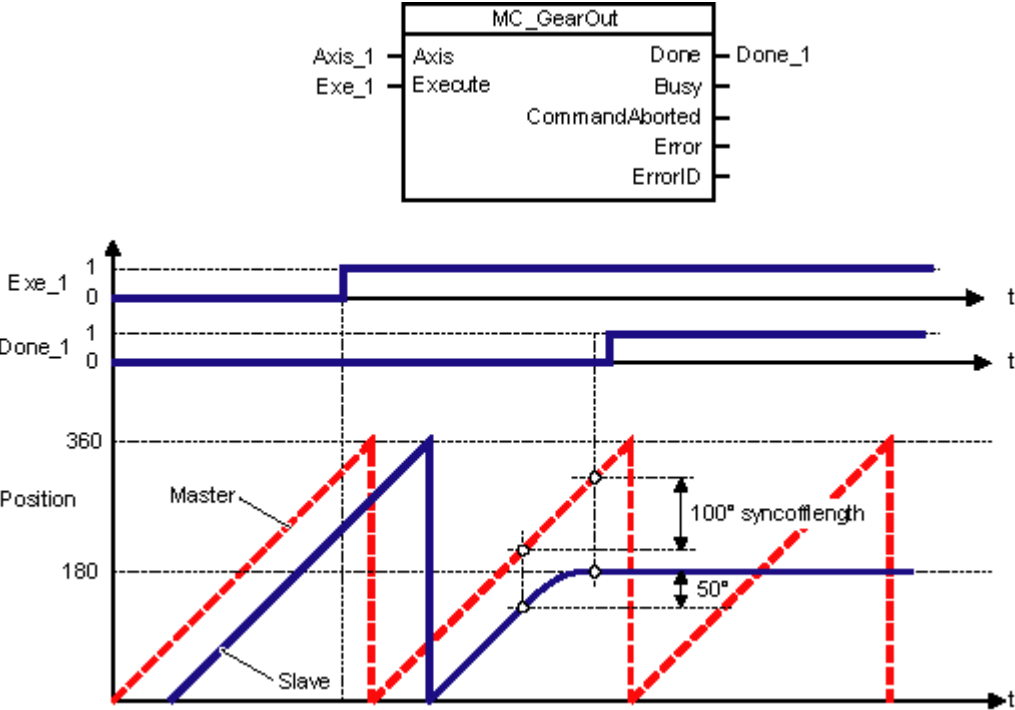
位置参考：“在去同步位置之前停止”

去同步跟随轴： 180 °

该轴基于引导轴位置执行去同步。跟随轴在引导轴覆盖的 100 °去同步长度内去同步。跟随轴覆盖引导轴的一半距离 (50 °)。

曲线设置：“引导轴相关的同步曲线”

去同步长度： 100 °



MC_GearOut — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8040	轴/外部编码器被禁用，或者设置了错误的模式	缺少用于启用跟随轴的命令所需的信号。
8043	参数值或默认值无效	S7T Config 中可能存在组态错误，通常是因为组态了去同步长度 <i>0</i> 。
8044	命令不受工艺对象支持	在输入参数 <i>Slave</i> 中定义的工艺 DB 至少必须属于类型“同步轴”。
8045	在当前状态下不允许使用该命令	"例如，“MC_Stop”仍处于活动状态
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是您可能无法通过状态输出对其进行监视。

ErrorID	错误消息	说明/纠正或避免错误的方法
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i>）的上升沿或下降沿）未传送到集成工艺 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8083	DB 不是工艺 DB	通过输入参数 <i>Slave</i> 指定的 DB 未找到或不是工艺 DB。
8084	工艺 DB 无效	- 在通过输入参数 <i>Slave</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Slave</i> 更改 DB 号。 - 用户在输入参数 <i>Slave</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management) 中删除工艺 DB，然后创建一个新的工艺 DB。

6.3.3 FB440 MC_GearInSuperImposed — 启动叠加齿轮传动

使用 FB440“MC_GearInSuperImposed”启动叠加齿轮传动

用途

- “MC_GearInSuperImposed”工艺功能用于在引导轴和跟随轴之间启动叠加齿轮传动。
- 可以在动态参数 *Velocity, Jerk, Acceleration, Deceleration* 中定义跟随轴同步期间的动态响应。
- 传动比定义为输入参数中的两个整数的比率函数（分子/分母）
- 同步与起始位置（达到同步时的随机位置值）可以是相对的，也可以是绝对的
- 可以通过请求一个新的 MC_GearInSuperImposed 命令在系统处于 RUN 时更改传动比。该操作无需停止引导轴或跟随轴。将根据指定的加速或减速值执行跳转
- 引导轴处于停止状态或运动状态时，均可以启动该功能。

适用对象

- 具有叠加同步对象的同步轴

先决条件

- 将引导轴组态为外部编码器、定位轴或同步轴
- 将跟随轴组态为具有叠加同步对象的同步轴
- 将引导轴设置为跟随轴的可能叠加同步对象
- 启用跟随轴以进行位置控制操作
- 跟随轴上没有正在执行的 MC_Stop 命令。

命令的相互作用

新命令 — 活动的单个命令 (1)

新命令 — 活动的命令 (1)

输入参数

参数	数据类型	初始值	说明
<i>Master</i>	INT	0	引导轴工艺 DB 的编号
<i>Slave</i>	INT	0	跟随轴工艺 DB 的编号
<i>Execute</i>	BOOL	FALSE	在上升沿处启动命令
<i>RatioNumerator</i>	DINT	1	传动比分子
<i>RatioDenominator</i>	DINT	1	传动比分母
<i>Velocity</i>	REAL	-1.0	跳转到齿轮传动时的最大速度（仅当 <i>Mode</i> = 1 时有效）：
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值
<i>Acceleration</i>	REAL	-1.0	跳转到齿轮传动时的加速度（仅当 <i>Mode</i> = 1 时有效）：
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值
<i>Deceleration</i>	REAL	-1.0	跳转到齿轮传动时的减速度（仅当 <i>Mode</i> = 1 时有效）：
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值
<i>Jerk</i>	REAL	-1.0	跳转到齿轮传动时的冲击（仅当 <i>Mode</i> = 1 时有效）：
			值 > 0: 使用定义的值
			值 = 0: 使用梯形运动曲线
			值 < 0: 使用默认值

参数	数据类型	初始值	说明
<i>PhaseShift</i>	REAL	0.0	达到同步后的相位偏移。 如果 <i>Absolute</i> = <i>TRUE</i> ，则当达到同步操作时，相位偏移是绝对的。 如果 <i>Absolute</i> = <i>FALSE</i> ，则指定的相位偏移将添加到由相对关系确定的相位偏移。
			值 > 0: 使用定义的值
			值 = 0: 无相位偏移
			值 < 0: 不允许
<i>Absolute</i>	BOOL	<i>TRUE</i>	<i>TRUE</i> : 绝对叠加齿轮传动 适用于固件版本不高于 V3.2.x 的集成工艺： 忽略 <i>Jerk</i> 中预设的参数。始终使用梯形运动曲线（不受限制的冲击）。
			<i>FALSE</i> : 相对叠加齿轮传动
<i>Mode</i>	INT	0	同步模式/耦合模式：
			值 = 0: 使用相应同步对象的默认值
			值 = 1: 基于时间的同步： 根据输入参数 <i>Velocity</i> , <i>Jerk</i> , <i>Acceleration</i> , <i>Deceleration</i> 立即同步（与在 S7T Config 中设置的同步方向的“兼容性模式”一致）
			值 = 2: 基于时间的同步： 根据输入参数 <i>Velocity</i> , <i>Jerk</i> , <i>Acceleration</i> , <i>Deceleration</i> 立即同步。程序还将评估 <i>userdefault.gearingsetting.synchronizingdirection</i> 系统变量。 可以在 S7T Config 的专家列表或通过“MC_WriteParameter”工艺功能来设置该系统变量。
<i>DoneFlag</i>	INT	0	在 MCDevice DB 中生成 DoneFlag 当达到齿轮传动状态时，将对 DoneFlag 值取反。

输出参数（状态输出）

参数	数据类型	初始值	说明	
<i>InGear</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 实现叠加齿轮传动	
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令	
<i>CommandAborted</i>	BOOL	<i>FALSE</i>	<i>TRUE</i>: 命令由其它命令或因其执行过程中的错误而取消。 如果工艺 DB 的 <i>ErrorStatus</i> 中未显示错误，则该命令由后续命令取消。 如果工艺 DB 的变量 <i>ErrorStatus</i> 中指示有错误，则命令执行期间影响该工艺对象的错误将导致该命令中止。	
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> :	命令启动时 有错误 。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
			<i>FALSE</i> :	命令启动时 没有错误 。
<i>ErrorID</i>	WORD	<i>0</i>	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。	

输入参数 *Mode* = 1

模数轴的同步方向：如果 *Mode* = 1，
则“兼容性模式”处于活动状态，而不管 S7T Config 中的设置如何：

- 齿轮传动：
跟随轴在正向上同步

MC_GearInSuperImposed — 示例 — “叠加”

以下示例中的信号图显示了绝对基本同步和绝对叠加同步之间的区别。

第一阶段 — 基本同步

Exe_1 用于启动绝对基本同步。短暂延迟后，跟随轴 (*Axis_3*) 将更改为同步操作。该工艺功能将报告 *InGear_1*。

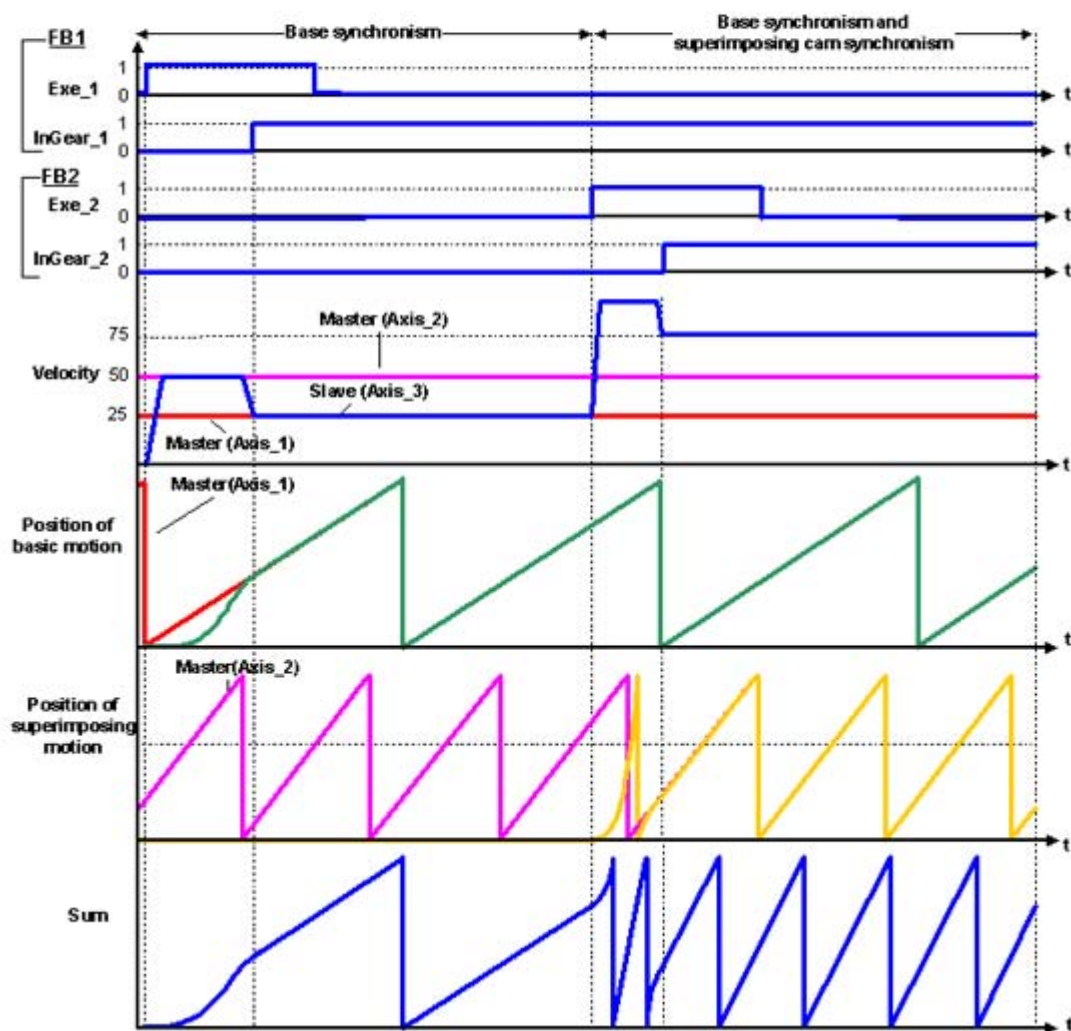
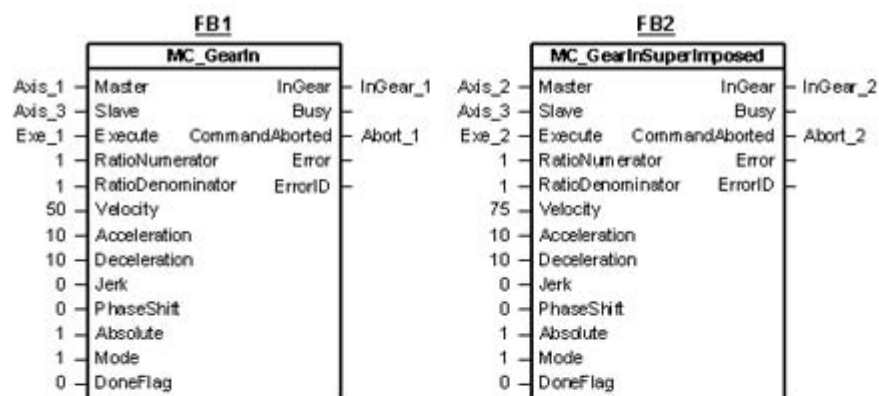
默认情况下，叠加运动的起始位置在 0 处，因此在基本同步阶段内，基本运动的位置与附加运动的位置是一致的。

绝对参考作为引导轴位置 (*Axis_1*) 的函数提供给基本运动位置（绿色信号图）。

第二阶段 — 基本同步和叠加同步

Exe_2 用于启动绝对叠加同步（橙色信号图）。绝对参考作为引导轴位置 (*Axis_2*) 的函数提供给跟随轴的叠加运动。引导轴位置 (*Axis_1*) 和基本运动位置（绿色信号图）之间的绝对参考保持不变。

跟随轴位置 (*Axis_3*) 来自于“基本运动的位置”值与“叠加运动的位置”值之和。当达到叠加同步时，该工艺功能将报告 *InGear_2*。



MC_GearInSuperImposed — 示例 — “相对/绝对叠加同步”

以下示例中的信号图显示了相对叠加同步和绝对叠加同步之间的区别。为了获得更好的观察效果，在所显示的信号图中未执行基本同步。

第一阶段 — 相对叠加同步

在显示的信号图中，引导轴 (*Axis_1*) 和跟随轴 (*Axis_2*) 在同一位置启动。相对叠加同步通过 *Exe_1* 的上升沿启动。短暂延迟后，*InGear_1* 将报告已达到相对叠加同步。

橙色信号图（第一阶段）显示了叠加跟随轴位置 (*Axis_2*)。默认情况下，叠加跟随轴的起始位置在 0 处。

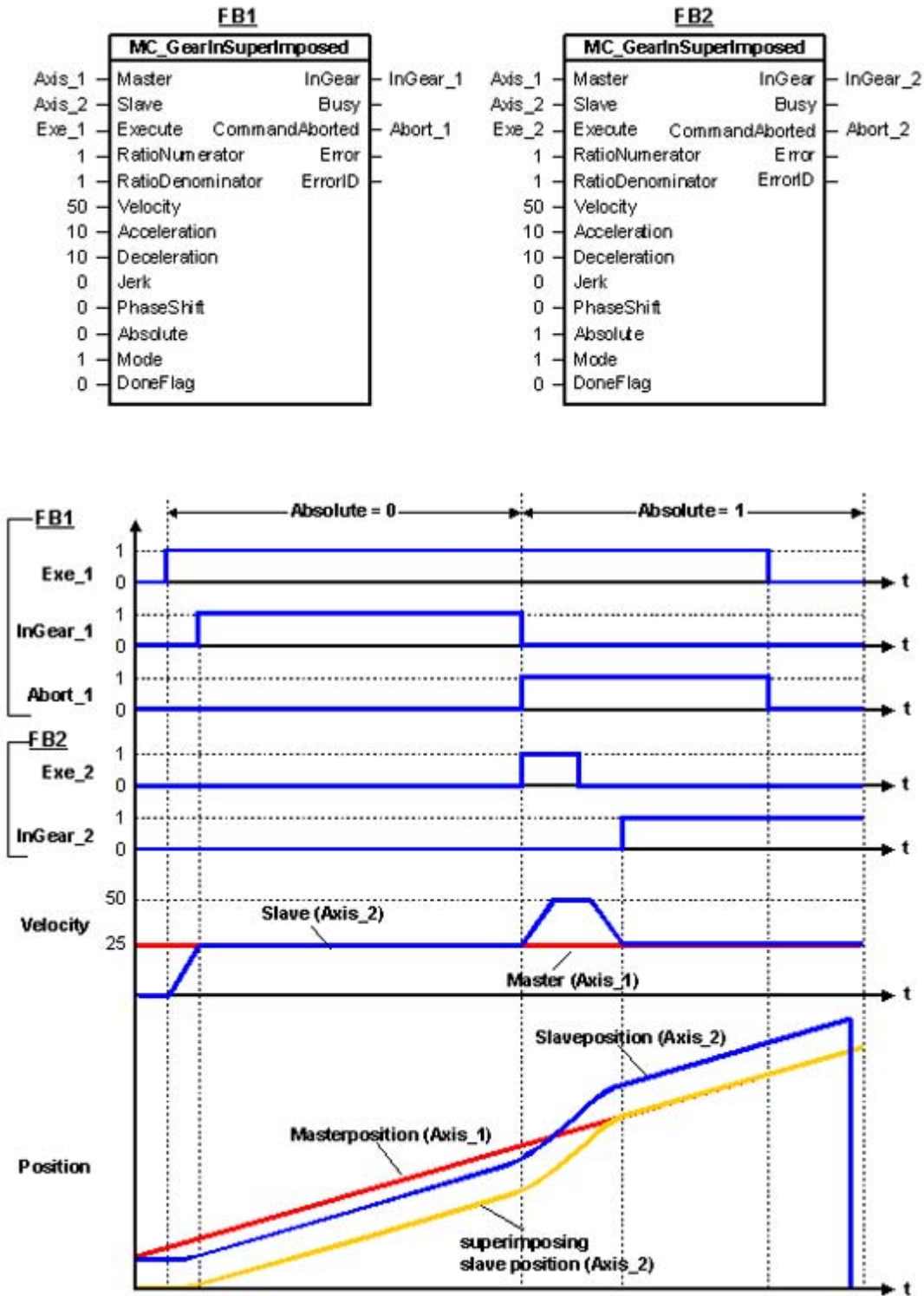
在引导轴位置 (*Axis_1*) 和叠加跟随轴位置 (*Axis_2*) 之间已建立相对同步。

第二阶段 — 绝对叠加同步

Exe_2 用于取消当前相对叠加同步并启动绝对叠加同步。短暂延迟后，*InGear_2* 将报告达到了绝对叠加同步。

橙色信号图（第二阶段）显示了叠加跟随轴位置 (*Axis_2*)。在引导轴位置 (*Axis_1*) 和叠加跟随轴位置 (*Axis_2*) 之间已建立绝对同步。原始叠加跟随轴的参考位置保持不变，包括叠加跟随轴位置和跟随轴位置之间的偏移。

6.3 工艺功能 — 齿轮传动/凸轮传动



MC_GearInSuperImposed — 示例 — “相移”

以下示例中的信号图显示了使用相移和不使用相移的绝对叠加同步之间的区别。为了更好的观察效果，在所显示的信号图中未执行基本同步。

不使用相移的绝对叠加同步

信号图的起始位置：

- 引导轴位置 ($Axis\_1$) = 0
- 跟随轴位置 ($Axis\_2$) = X
- 叠加跟随轴位置 = 0

Exe_1 的上升沿可以启动不使用相移的绝对叠加同步。短暂延迟后，*InGear_1* 将报告达到了绝对叠加同步。

橙色信号图指示了叠加从位置。默认情况下，叠加从位置的起始位置在 0 处。

在引导轴位置 ($Axis\_1$) 和叠加跟随轴位置 ($Axis\_2$) 之间已建立绝对同步。

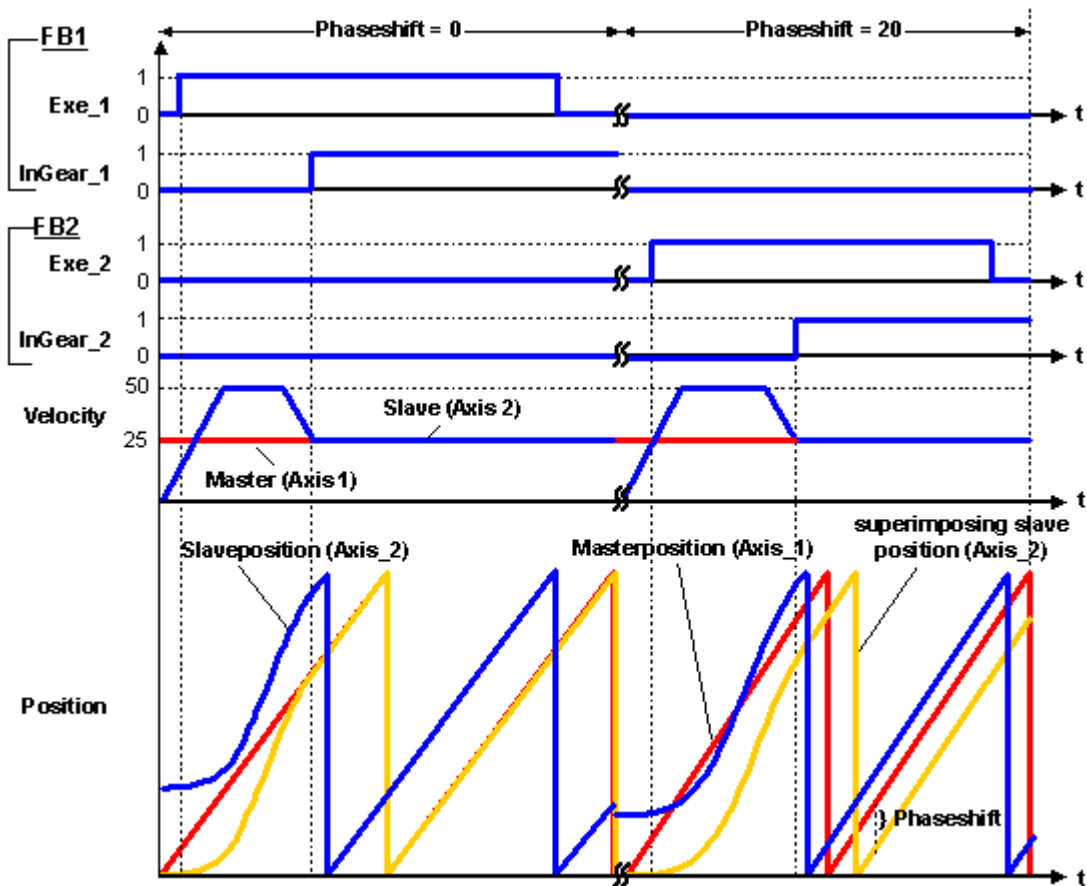
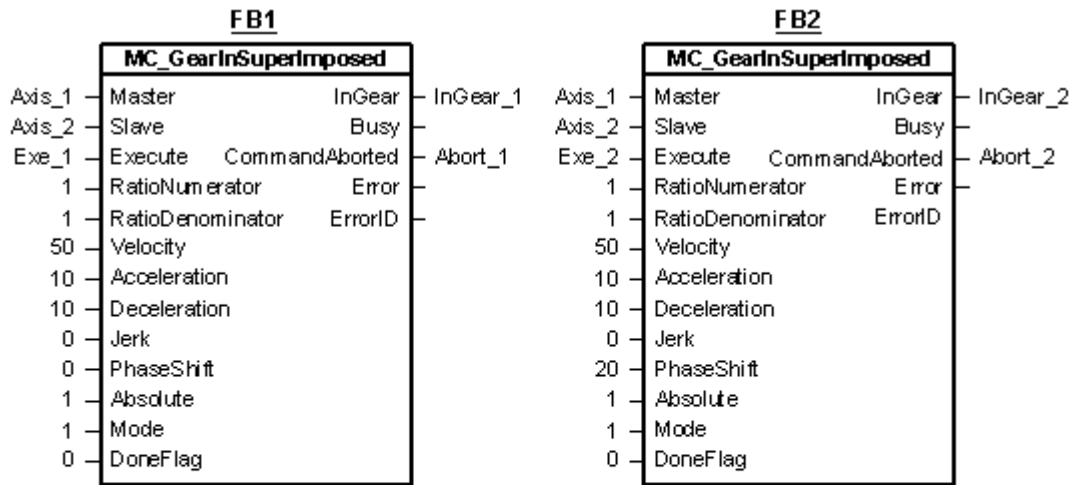
使用相移的绝对叠加同步

该信号图适用于与不使用相移的信号图类似的起始位置。

- 引导轴位置 ($Axis\_1$) = 0
- 跟随轴位置 ($Axis\_2$) = Y
- 叠加跟随轴位置 = 0

Exe_2 的上升沿可以启动使用相移的绝对叠加同步。短暂延迟后，*InGear_2* 将报告达到了绝对叠加同步。

该橙色信号图显示了叠加跟随轴位置 ($Axis\_2$)，但是它使用指定的相移。默认情况下，叠加跟随轴的起始位置仍位于 0 处。



MC_GearInSuperImposed — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8022	轴/外部编码器中没有实际值	例如，编码器或数据总线未准备就绪
8040	轴/外部编码器被禁用，或者设置了错误的模式	缺少运动命令所需的使能信号。 消除并确认所有排队的错误，然后启用相关的轴模式（例如，位置控制）。
8043	参数值非法	涉及数据类型为 <i>REAL</i> 的所有输入参数，或者输入参数 <i>RatioNumerator</i>、<i>RatioDenominator</i>、<i>Mode</i> 或 <i>DoneFlag</i>。 这可能是由组态引起的，因为当在基于该组态形成默认值（例如，同步长度 = 0）的块内发现错误的命令参数时也会显示该错误。
8044	命令不受工艺对象支持	在输入参数 <i>Master</i> 中定义的工艺 DB 至少必须属于类型“定位轴”。 在输入参数 <i>Slave</i> 中定义的工艺 DB 至少必须属于类型“同步轴”。

ErrorID	错误消息	说明/纠正或避免错误的方法
8045	在当前状态下不允许使用该命令	示例: - MC_Stop 命令处于活动状态 - 如果输入参数 <i>PositionControl</i> = <i>TRUE</i>, 则通过工艺功能“MC_MoveVelocity”对从轴的运动进行速度控制。从轴必须在位置控制模式下运行, 才能进行叠加同步操作。
804A	缺少所需的对象连接	在组态叠加同步对象时未选择输入参数 <i>Master</i> 中定义的引导轴。
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝, 以保持第一个命令的一致性。 如果命令输出率较高, 请务必使用单独的背景数据块或再次请求命令。请注意, 尽管接受的第一个命令可能处于活动状态, 但是您可能无法通过状态输出对其进行监视。
8050	工艺未准备就绪	- 在使用断点进行测试期间 (CPU 处于 <i>HOLD</i> 状态), 工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 <i>OB</i> 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块, 在不同运行级别调用了此工艺功能。 示例: 在 <i>OB 1</i> 和 <i>OB 35</i> 中都使用背景数据块 <i>DBx</i> 调用工艺功能 <i>x</i>。开始工艺功能在 <i>OB 1</i> 中执行, 然后由于在 <i>OB 35</i> 中调用它而中断其在 <i>OB 1</i> 中的执行。由于背景数据块是共享使用的, 因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应: - 新命令 (输入参数 <i>Execute / Enable</i> 的上升沿或下降沿) 未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是, 该命令在集成工艺中可能仍处于活动状态 注意: 在不同的运行级别使用不同的背景数据块, 或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障 (例如, 长度错误)。

ErrorID	错误消息	说明/纠正或避免错误的方法
8056	由于正在向固定挡块运动而取消	轴已运动到固定挡块 (<i>InClamping = TRUE</i>)。只有新命令从固定挡块释放轴，它们才被接受。 已启动的命令不符合该条件。
807A	主设定值无效	引导轴的实际值无效。
807B	工艺对象的递归互连	同步操作中，在该输入参数中定义的引导轴已作为跟随轴处于活动状态，并且将从输入参数中定义的跟随轴直接或间接获得主设定值。
8083	DB 不是工艺 DB	在输入参数 <i>Master</i> 或 <i>Slave</i> 中定义的某个 DB 未找到或不是工艺 DB。
8084	工艺 DB 无效	- 在通过输入参数 <i>Master</i> 或 <i>Slave</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Master</i> 或 <i>Slave</i> 更改 DB 号。 - 用户在工艺 DB（通过输入参数 <i>Master</i> 或 <i>Slave</i> 定义）中输入的数据无效。 在“工艺对象管理”(Technology Objects Management) 中删除该工艺 DB，然后创建一个新的工艺 DB。
808B	参数值的 REAL 格式无效	数据类型为 REAL 的输入参数的值不符合有效的浮点数格式。 检查输入参数值或背景数据块数据。不能以浮点格式表示格式无效的值。它们以十六进制格式 (DW16# ...) 显示。

6.3.4 FB442 MC_GearOutSuperImposed — 停止叠加齿轮传动

使用 FB442“MC_GearOutSuperImposed”停止叠加齿轮传动

用途

- “MC_GearOutSuperImposed”工艺功能用于停止叠加齿轮传动。这不会影响基本运动或基本同步操作。
- 使用 *Deceleration* 和 *Jerk* 输入参数可以在 *Mode = 1* 时立即停止叠加齿轮传动。根据 S7T Config 中叠加同步对象的默认设置，还可以在 *Mode = 0* 时使轴去同步。
- 工艺功能在跟随轴的叠加同步终止后终止。

适用对象

- 具有叠加同步对象的同步轴

先决条件

- 必须将该轴组态为跟随轴（从轴），以用于叠加同步系统。如果没有叠加齿轮传动处于活动状态，将忽略该命令，并且该工艺功能将报告 *Done = TRUE*。
- 未在执行 MC_Stop 命令。

命令的相互作用

- 新命令 — 活动的单个命令 (1)
- 新命令 — 活动的命令 (1)

输入参数

参数	数据类型	初始值	说明
<i>Slave</i>	INT	<i>0</i>	跟随轴工艺 DB 的编号
<i>Execute</i>	BOOL	<i>FALSE</i>	在上升沿处启动命令
<i>Deceleration</i>	REAL	<i>-1.0</i>	叠加齿轮传动结束时的减速度（仅当 <i>Mode = 1</i> 时有效）
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值

参数	数据类型	初始值	说明
<i>Jerk</i>	REAL	-1.0	叠加齿轮传动结束时的冲击（仅当 <i>Mode</i> = 1 时有效）
			值 > 0: 使用定义的值
			值 = 0: 使用梯形运动曲线
			值 < 0: 使用默认值
<i>Mode</i>	INT	0	去同步模式:
			值 = 0: 使用相应同步对象的默认值
			值 = 1: 根据输入参数 <i>Deceleration</i> 和 <i>Jerk</i> 立即去同步
<i>DoneFlag</i>	INT	0	在 MCDevice DB 中生成 DoneFlag

输出参数（状态输出）

参数	数据类型	初始值	说明
<i>Done</i>	BOOL	FALSE	TRUE: 停止叠加齿轮传动。
<i>Busy</i>	BOOL	FALSE	TRUE: 正在执行命令
<i>CommandAborted</i>	BOOL	FALSE	TRUE: 命令由其它命令或因其执行过程中的错误而取消。 如果工艺 DB 的 <i>ErrorStatus</i> 中未显示错误, 则该命令由后续命令取消。 如果工艺 DB 的变量 <i>ErrorStatus</i> 中指示有错误, 则命令执行期间影响该工艺对象的错误将导致该命令中止。
<i>Error</i>	BOOL	FALSE	TRUE: 命令启动时有错误。未执行该命令。有关原因的信息, 请参考 <i>ErrorID</i> 。
			FALSE: 命令启动时没有错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 ErrorID。

说明

将 MC_GearInSuperImposed 和 MC_GearOutSuperImposed 同步输出到同一工艺对象时可能会取消 MC_GearInSuperImposed 命令 (*CommandAborted* = TRUE)。因此, “MC_GearOutSuperImposed”工艺功能将报告 *Done* = TRUE。

请勿启动 MC_GearOutSuperImposed 命令, 除非 MC_GearInSuperImposed 已报告了同步的状态 (*InGear* = TRUE)。

MC_GearOutSuperImposed — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8040	轴/外部编码器被禁用，或者设置了错误的模式	缺少运动命令所需的使能信号。 消除并确认所有排队的错误，然后启用相关的轴模式（例如，位置控制）。
8043	参数值非法	涉及数据类型为 REAL 的所有输入参数，或者输入参数 <i>Mode</i> 或 <i>DoneFlag</i> 。
8044	命令不受工艺对象支持	例如，将命令发送至测量输入或输出凸轮
8045	在当前状态下不允许使用该命令	例如 MC_Stop 命令处于活动状态
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是您可能无法通过状态输出对其进行监视。

ErrorID	错误消息	说明/纠正或避免错误的方法
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8083	DB 不是工艺 DB	通过输入参数 <i>Slave</i> 指定的 DB 未找到或不是工艺 DB。
8084	工艺 DB 无效	- 在通过输入参数 <i>Slave</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Slave</i> 更改 DB 号。 - 用户在输入参数 <i>Slave</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management) 中删除工艺 DB，然后创建一个新的工艺 DB。
808B	参数值的 REAL 格式无效	数据类型为 REAL 的输入参数的值不符合有效的浮点数格式。 检查输入参数值或背景数据块数据。不能以浮点格式表示格式无效的值。它们以十六进制格式 (DW16# ...) 显示。

6.3.5 FB421“MC_CamIn”— 启动凸轮传动

使用 FB421“MC_CamIn”启动凸轮传动

适用于固件版本为 V3.2.x 或更高版本的集成工艺

本部分将介绍 V3.1.x 及更低版本固件的应用。

用途

- “MC_CamIn”工艺功能用于在引导轴和跟随轴（主轴和从轴）之间启动凸轮传动。
- 可以在动态参数 *Velocity* 和 *Jerk, Acceleration* 中定义跟随轴同步期间的动态响应。
- 可以对指定的凸轮进行缩放和/或平移操作。
- 可以选择让指定的凸轮运行一次或周期性运行。
- 同步可以是绝对的，也可以是相对的。

适用对象

- 同步轴

先决条件

- 将引导轴组态为外部编码器、定位轴或同步轴
- 将跟随轴组态为同步轴
- 组态跟随轴的同步对象时，必须已选择所需的凸轮盘和引导轴。
- 启用跟随轴以进行位置控制操作
- 跟随轴上没有正在执行的 MC_Stop 命令。

命令的相互作用

新命令 — 活动的单个命令 (1)

新命令 — 活动的命令 (1)

输入参数

参数	数据类型	初始值	说明
<i>Master</i>	INT	0	引导轴工艺 DB 的编号
<i>Slave</i>	INT	0	跟随轴工艺 DB 的编号
<i>CamTable</i>	INT	0	凸轮盘工艺 DB 的编号
<i>Execute</i>	BOOL	FALSE	在上升沿处启动命令
<i>MasterOffset</i>	REAL	0.0	在引导轴坐标系中平移凸轮（仅当 <i>MasterAbsolute</i> = TRUE 时生效）
<i>SlaveOffset</i>	REAL	0.0	在跟随轴坐标系中平移凸轮（仅当 <i>SlaveAbsolute</i> = TRUE 时生效）
<i>MasterScaling</i>	REAL	1.0	引导轴坐标系中的凸轮缩放因子（不允许值 0）
<i>SlaveScaling</i>	REAL	1.0	跟随轴坐标系中的凸轮缩放因子（不允许值 0）
<i>MasterAbsolute</i>	BOOL	TRUE	表明凸轮盘坐标对于引导轴是绝对的还是相对的：
			TRUE: 绝对坐标
			FALSE: 相对坐标
<i>SlaveAbsolute</i>	BOOL	TRUE	表明凸轮盘坐标对于跟随轴是绝对的还是相对的
			TRUE: 绝对坐标
			FALSE: 相对坐标
<i>CyclicMode</i>	BOOL	TRUE	处理凸轮盘
			TRUE: 循环执行
			FALSE: 非循环执行
<i>Velocity</i>	REAL	-1.0	最大耦合速度 （仅当 <i>Mode</i> = 1 时有效）：
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值
<i>Acceleration</i>	REAL	-1.0	耦合时的加速度： （仅当 <i>Mode</i> = 1 时有效）：
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值

参数	数据类型	初始值	说明
<i>Jerk</i>	REAL	<i>-1.0</i>	耦合跟随轴时冲击 (仅当 <i>Mode = 1</i> 时有效) :
			值 > 0: 使用定义的值
			值 = 0: 使用梯形运动曲线
			值 < 0: 使用默认值
<i>Mode</i>	INT	<i>0</i>	同步模式/耦合模式:
			值 = 0: 使用相应同步对象的默认值
			值 = 1: 基于时间的同步: 根据输入参数 <i>Velocity, Jerk, Acceleration</i> 立即同步 (与在 <i>S7T Config</i> 中设置的同步方向的“兼容性模式”一致)
			值 = 2: 基于时间的同步: 根据输入参数 <i>Velocity, Jerk, Acceleration, Deceleration</i> 立即同步。 程序还将考虑到同步对象的 Axis_SYNCHRONISM > “默认值”(Default) 对话框的“齿轮传动”选项卡中的设置。 它还将评估系统变量 <i>userdefault.cammingsetting</i> . <i>synchronizingdirection</i> 。
<i>DoneFlag</i>	INT	<i>0</i>	在 MCDevice DB 中生成 DoneFlag 命令 (<i>InSync</i>) 完成时, 将对 DoneFlag 的值取反

输出参数（状态输出）

参数	数据类型	初始值	说明	
<i>InSync</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 实现基本凸轮传动	
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令	
<i>CommandAborted</i>	BOOL	<i>FALSE</i>	<i>TRUE</i>: 命令由其它命令或因其执行过程中的错误而取消。 如果工艺 DB 的 <i>ErrorStatus</i> 中未显示错误，则该命令由后续命令取消。 如果工艺 DB 的变量 <i>ErrorStatus</i> 中指示有错误，则命令执行期间影响该工艺对象的错误将导致该命令中止。	
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> :	命令启动时 有 错误。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
			<i>FALSE</i> :	命令启动时 没有 错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。	

输入参数 *Mode* = 1

模数轴的同步方向：如果 *Mode* = 1，
则“兼容性模式”处于活动状态，而不管 S7T Config 中的设置如何：

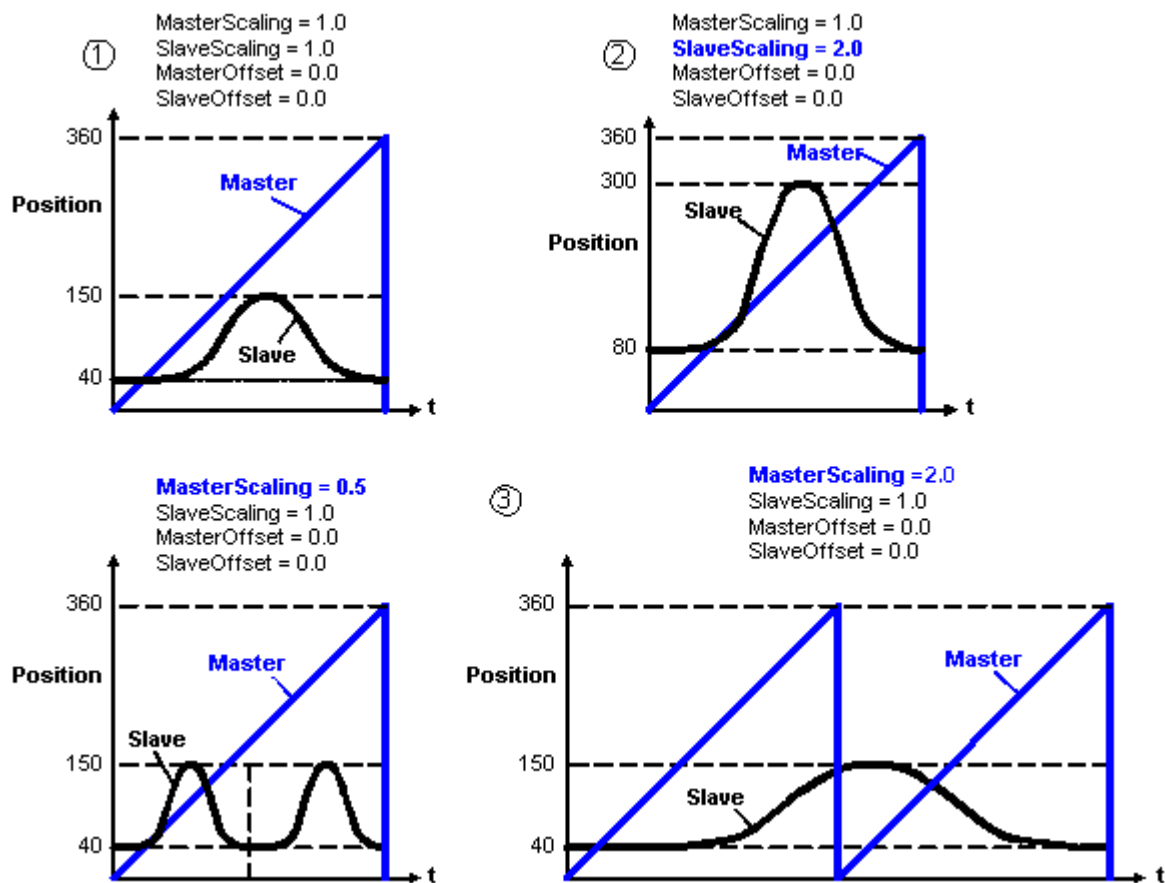
- 凸轮传动：
跟随轴将在最短的距离内同步。

启动凸轮传动的建议

1. 将跟随轴 (*Slave*) 移动到位置 X_S 。
2. 将引导轴 (*Master*) 移动到位置 X_m ，其中 X_S = 凸轮值 (X_m)
3. 启动“MC_CamIn”工艺功能。

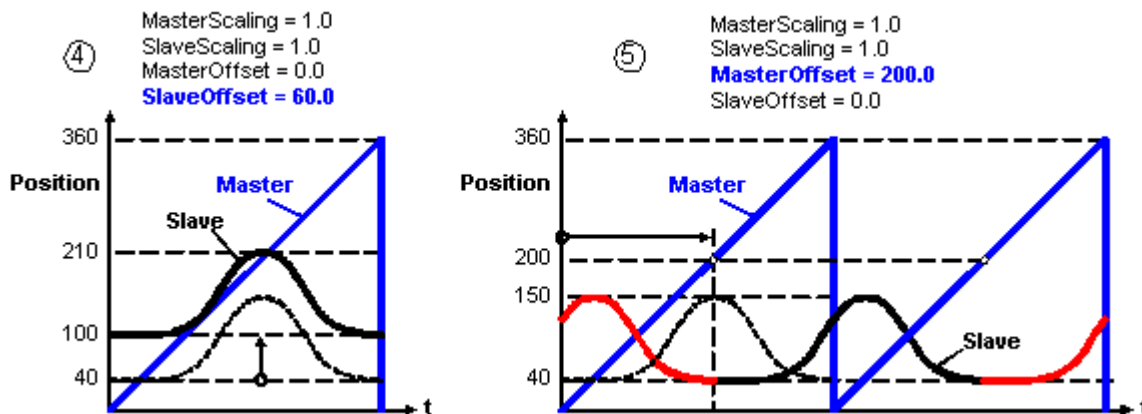
MC_CamIn — 示例 — “缩放因子和偏移值的影响”

缩放因子的影响



- ① 非缩放的输出凸轮（缩放因子为 1.0）。
- ② 可以使用参数 *SlaveScaling* 沿 Y 轴放大或缩小凸轮。因此跟随轴在引导轴周期内多少会有些错位。
- ③ 可以使用参数 *MasterScaling* 沿 X 轴放大或缩小凸轮。

偏移值的效果



- ④ *SlaveOffset* 参数将沿 Y 轴平移整个凸轮。
- ⑤ *MasterOffset* 参数将沿 X 轴平移凸轮，以生成带有相移的凸轮。

MC_CamIn — 示例 — “根据引导轴位置的同步（循环相对同步）”

同步模式：

相对凸轮传动，循环模式

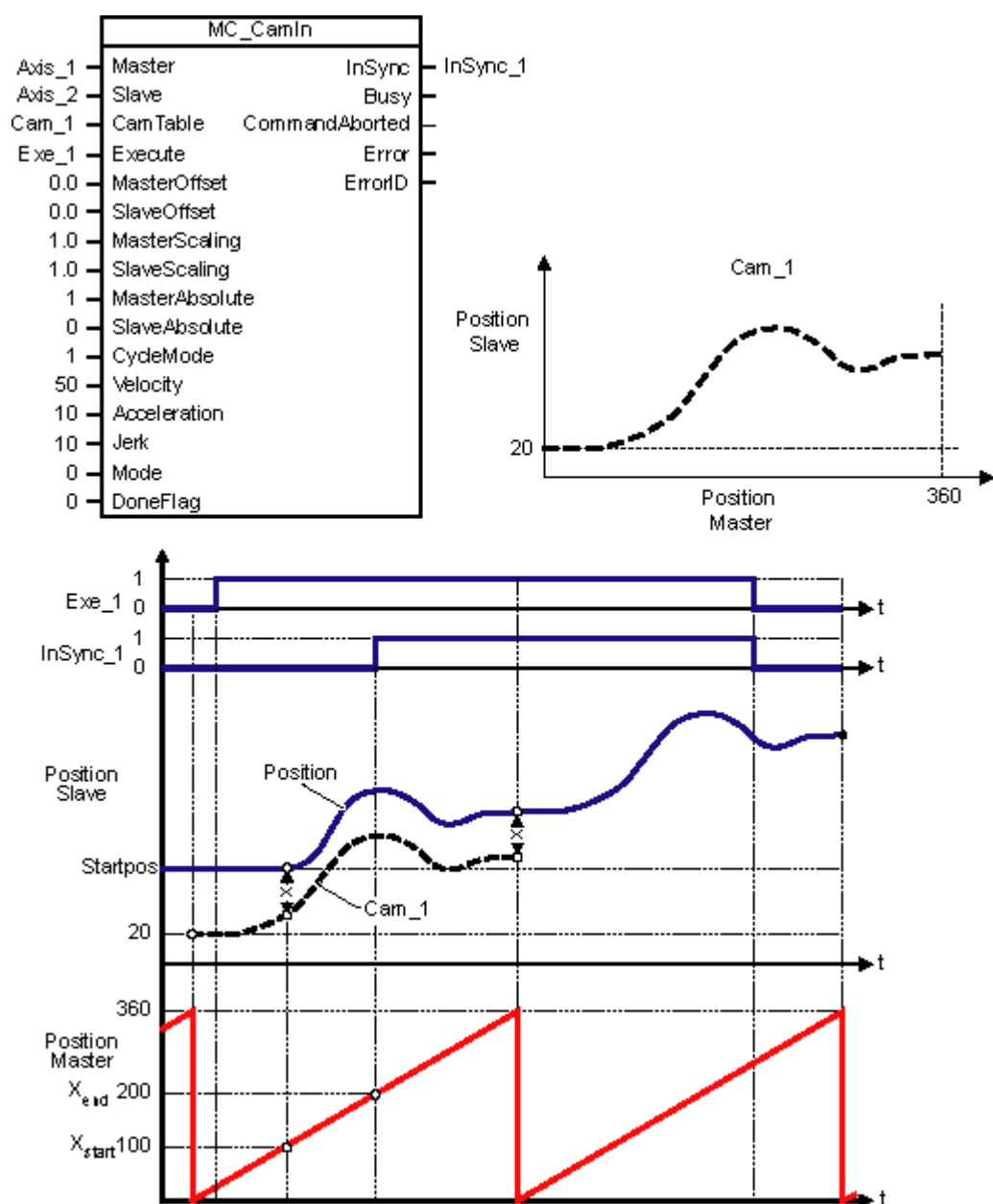
在相对凸轮传动模式下，凸轮起始值和结束值可能采用 Y 轴上的任意值。在当前凸轮周期结束时，凸轮将添加到当前 Y 值。

同步条件：

如果 *Mode* = 0，则轴组态中定义或者通过在相应参数中输入来定义的同步条件将生效：

同步：	“引导轴的默认同步位置”
位置参考：	“从同步位置进行同步”
同步位置主设定值：	100.0
曲线设置：	“特定于引导轴的同步曲线”
同步长度：	100.0

当引导轴通过位置 100° 后同步启动。引导轴在角度范围 100° 到 200° 内移动时跟随轴同步（同步长度 = 100°）。



MC_CamIn — 示例 — “根据引导轴位置的同步（循环绝对同步）”**同步模式：**

绝对凸轮传动，循环模式

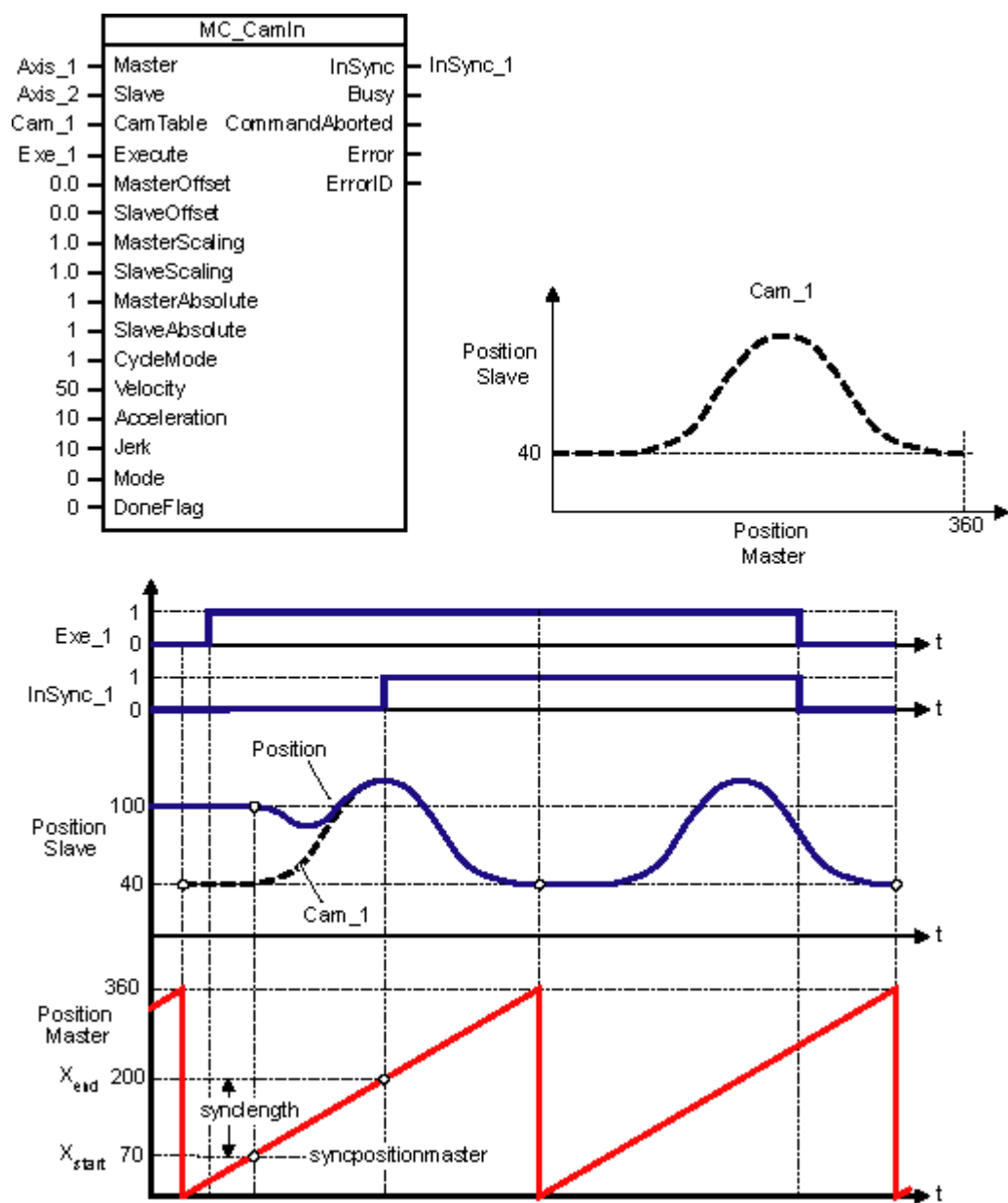
凸轮起始位置和结束位置的 Y 值和斜率是相同的。这使您可以在循环模式下联结凸轮，而不会导致中断。

同步条件：

如果 *Mode* = 0，则轴组态中定义或者通过在相应参数中输入来定义的同步条件将生效：

同步：	“引导轴的默认同步位置”
位置参考：	“从同步位置进行同步”
同步位置主设定值：	70.0
曲线设置：	“特定于引导轴的同步曲线”
同步长度：	130.0

当引导轴通过位置 70 °后同步启动。引导轴在角度范围 70 °到 200 °内移动时跟随轴同步（同步长度 = 130 °）。



MC_CamIn —“带有立即同步的信号图”实例

同步模式:

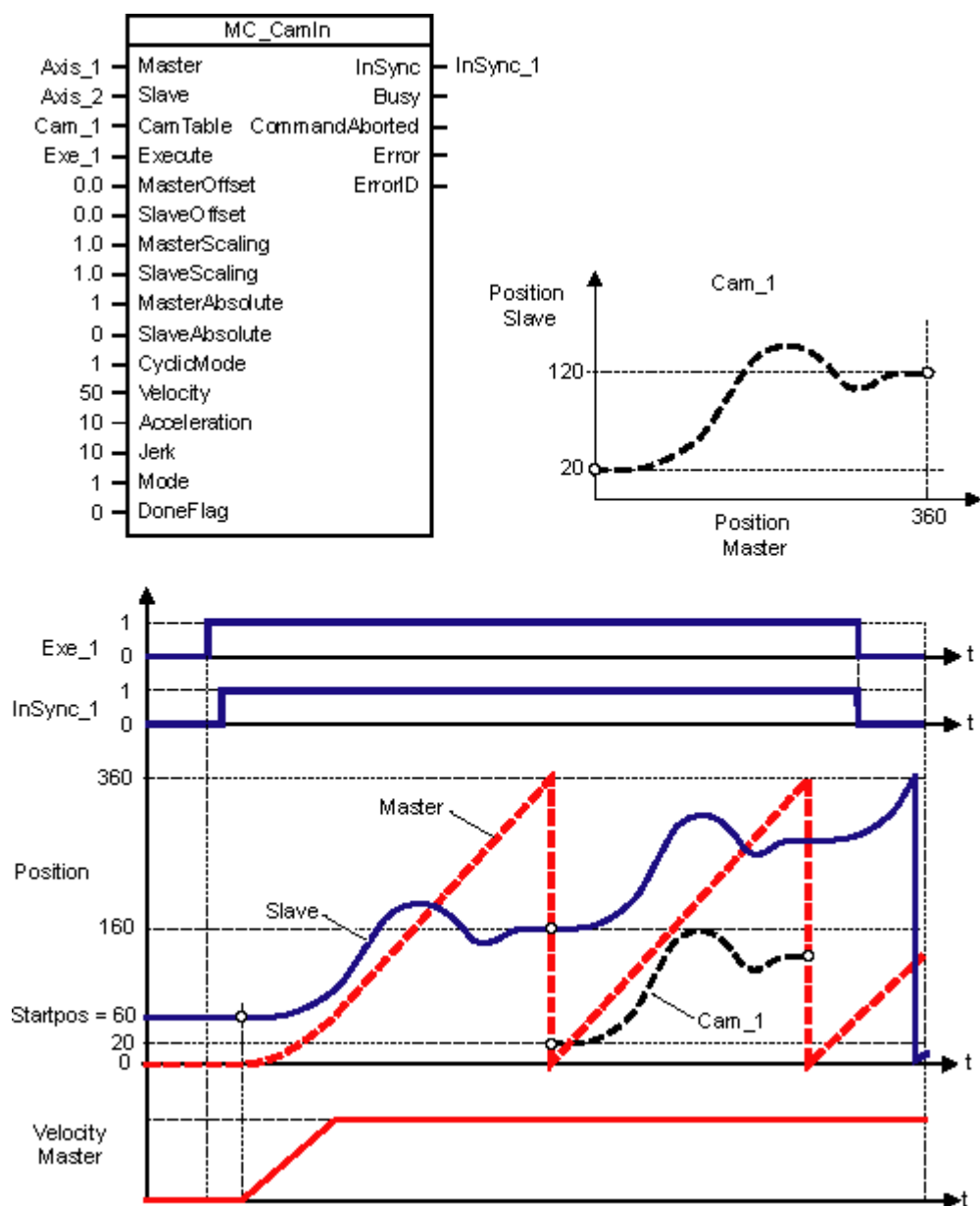
相对凸轮传动，循环模式

在相对凸轮传动模式下，凸轮起始值和结束值可能采用 Y 轴上的任意值。在当前凸轮周期结束时，凸轮将添加到当前 Y 值。

同步条件:

如果 *Mode* = 1，则凸轮将根据在“MC_CamIn”工艺功能中设置的动态参数立即同步。

在本示例中，最初通过定位命令将引导轴和跟随轴移动到其初始位置。“MC_CamIn”工艺功能在下一步中启动。两个轴都处于停止状态并请求相对同步，因此同步立即生效。要获得绝对同步，从轴首先必须移动到分配给引导轴的绝对凸轮插补点。



MC_CamIn — 示例 — “周期结束时的凸轮转换”

同步模式:

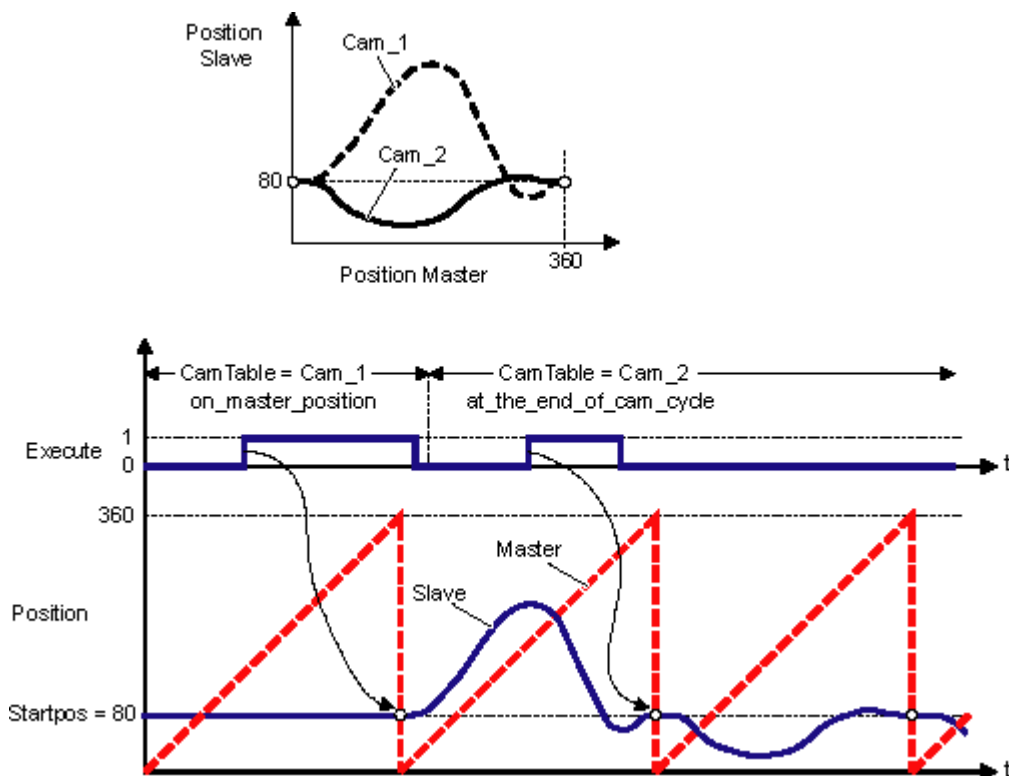
在绝对凸轮传动模式下对跟随轴进行操作。要在“活动周期结束时”获得同步，必须在相对同步模式下对引导轴进行操作。

同步条件:

输入参数 *Mode* = 0 将激活轴组态中定义的或者通过相应参数条目定义的同步条件。在本示例中，同步条件在系统处于 RUN 时更改。

顺序

1. “引导轴的预设同步位置”在 S7T Config 中作为同步轴的默认同步模式设置。位置参考设置为“从同步位置同步”，其中同步位置主值为 0.0。
Execute = 1 将启动与“Cam_1”的同步（在引导轴的值为 0.0 时启动）。
2. 使用“MC_WriteParameter”工艺功能可更改同步位置。为此，将跟随轴的参数 4266 设置为值“AT_THE_END_OF_CAM_CYCLE”。
3. 在输入参数 *CamTable* 中设置“Cam_2”的凸轮实例。
4. 下次 *Execute* = 1 时，在活动凸轮结束时启动与“Cam_2”的同步。



MC_CamIn — 示例 — “同步条件 AT_MASTER_AND_SLAVE_POSITION”

同步模式:

引导轴和跟随轴在绝对凸轮传动模式下进行操作。

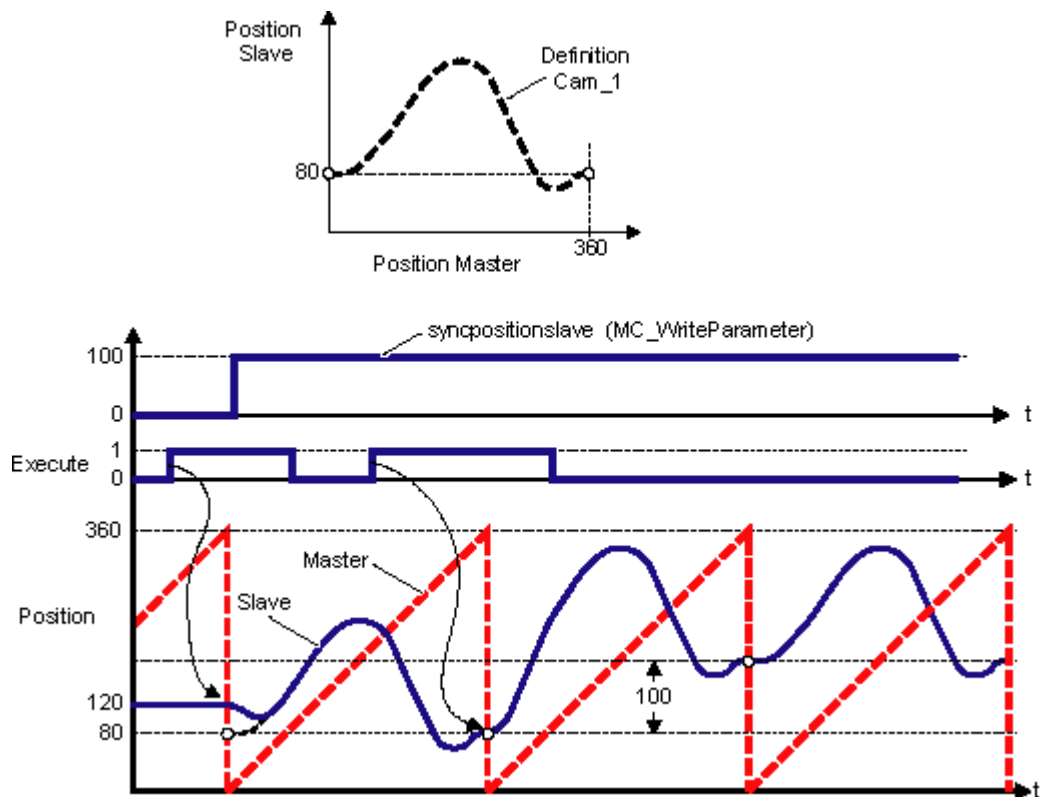
同步条件:

输入参数 *Mode* = 0 将激活轴组态中定义的或者通过相应参数条目定义的同步条件。

在本示例中，跟随轴的同步位置（参数 4269）在系统处于 RUN 时更改。

顺序

1. 同步条件: “引导轴和跟随轴的默认同步位置”
位置参考: “从同步位置同步”
同步位置主轴: 0.0
同步位置跟随轴: 0.0
2. 通过调用“MC_WriteParameter”工艺功能（参数 4269）更改同步位置
3. 同步条件: “引导轴和跟随轴的默认同步位置”
位置参考: “从同步位置同步”
同步位置主轴: 0.0
同步位置跟随轴: 100.0



MC_CamIn — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8022	轴/外部编码器中没有实际值	例如，编码器或数据总线未准备就绪
8040	轴/外部编码器被禁用，或者设置了错误的模式	缺少用于启用跟随轴的命令所需的信号。
8043	参数值或默认值无效	涉及数据类型为 <i>REAL</i> 的所有输入参数，或者输入参数 <i>Mode</i> 或 <i>DoneFlag</i>。 <i>S7T Config</i> 中也可能存在错误组态。通常是组态的同步长度为 0。
8044	命令不受工艺对象支持	在输入参数 <i>Master</i> 中定义的工艺 DB 至少必须属于类型“定位轴”。 在输入参数 <i>Slave</i> 中定义的工艺 DB 至少必须属于类型“同步轴”。
8045	在当前状态下不允许使用该命令	MC_Stop 命令处于活动状态

ErrorID	错误消息	说明/纠正或避免错误的方法
804A	缺少所需的对象连接	在基本同步对象的组态中，既没有选择在输入参数 <i>Master</i> 中定义的引导轴，也没有选择在输入参数 <i>CamTable</i> 中定义的凸轮盘
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是您可能无法通过状态输出对其进行监视。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8056	由于正在向固定挡块运动而取消	轴已运动到固定挡块 (<i>InClamping = TRUE</i>)。只有新命令从固定挡块释放轴，它们才被接受。 已启动的命令不符合该条件。
807A	主设定值无效	引导轴的实际值无效。

ErrorID	错误消息	说明/纠正或避免错误的方法
807B	工艺对象的递归互连	同步操作中，在该输入参数中定义的引导轴已作为跟随轴处于活动状态，并且将从输入参数中定义的跟随轴直接或间接获得主设定值。
8083	DB 不是工艺 DB	通过输入参数 <i>Master</i> , <i>Slave</i> 或 <i>CamTable</i> 指定的 DB 未找到或不是工艺 DB。
8084	工艺 DB 无效	- 在通过输入参数 <i>Master</i>, <i>Slave</i> 或 <i>CamTable</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Master</i>, <i>Slave</i> 或 <i>CamTable</i> 更改 DB 号。 - 用户在工艺 DB（通过输入参数 <i>Master</i>, <i>Slave</i> 或 <i>CamTable</i> 定义）中输入的数据无效。 在“工艺对象管理”(Technology Objects Management) 中删除该工艺 DB，然后创建一个新的工艺 DB。
808B	参数值的 REAL 格式无效	数据类型为 REAL 的输入参数的值不符合有效的浮点数格式。 检查输入参数值或背景数据块数据。 不能以浮点格式表示格式无效的值。它们以十六进制格式 (DW16# ...) 显示。

6.3.6 FB423 MC_CamOut — 停止凸轮传动

使用 FB 423“MC_CamOut”停止凸轮传动

用途

- “MC_CamOut”用于停止凸轮传动和跟随轴。可以在轴组态中定义相应的条件

说明

如果关闭过程取决于引导轴和/或跟随轴的位置，请使用该工艺功能。

也可以使用“MC_Halt”、“MC_Stop”、“MC_MoveRelative”、“MC_MoveAdditive”、“MC_MoveAbsolute”、“MC_MoveVelocity”或“MC_MoveToEndPos”工艺功能终止跟随轴的同步操作。

适用对象

- 同步轴

先决条件

- 必须将该轴组态为跟随轴 (Slave) 以进行凸轮传动操作。如果没有凸轮传动处于活动状态，则该命令将被忽略，并且该工艺功能将报告 *Done = TRUE*。
- 没有处于活动状态的 MC_Stop 命令

命令的相互作用

新命令 — 活动的单个命令 (1)

新命令 — 活动的命令 (1)

输入参数

参数	数据类型	初始值	说明
<i>Slave</i>	INT	0	跟随轴工艺 DB 的编号
<i>Execute</i>	BOOL	<i>FALSE</i>	在上升沿处启动命令

输出参数（状态输出）

参数	数据类型	初始值	说明
<i>Done</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 凸轮传动已停止 如果没有基本凸轮传动命令处于活动状态，则输出参数 <i>Done</i> 还将报告 <i>TRUE</i> 。
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令
<i>CommandAborted</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令由其它命令或因其执行过程中的错误而取消。 如果工艺 DB 的 <i>ErrorStatus</i> 中未显示错误，则该命令由后续命令取消。 如果工艺 DB 的变量 <i>ErrorStatus</i> 中指示有错误，则命令执行期间影响该工艺对象的错误将导致该命令中止。
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令启动时有错误。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
			<i>FALSE</i> : 命令启动时没有错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。

说明

将 MC_CamIn 和 MC_CamOut 同步输出到同一工艺对象时可能会取消 MC_CamIn 命令 (*CommandAborted* = *TRUE*)。因此，“MC_CamOut”工艺功能将报告 *Done* = *TRUE*。

请勿启动 MC_CamOut 命令，除非 MC_CamIn 已报告了凸轮传动状态 (*InSync* = *TRUE*)。

MC_CamOut — 示例

“MC_CamOut”工艺功能用于停止凸轮传动以及根据组态的模式停止轴。 在本示例中，跟随轴应在位置 160° 处停止。 相应的设置：

去同步：“在跟随轴位置处”

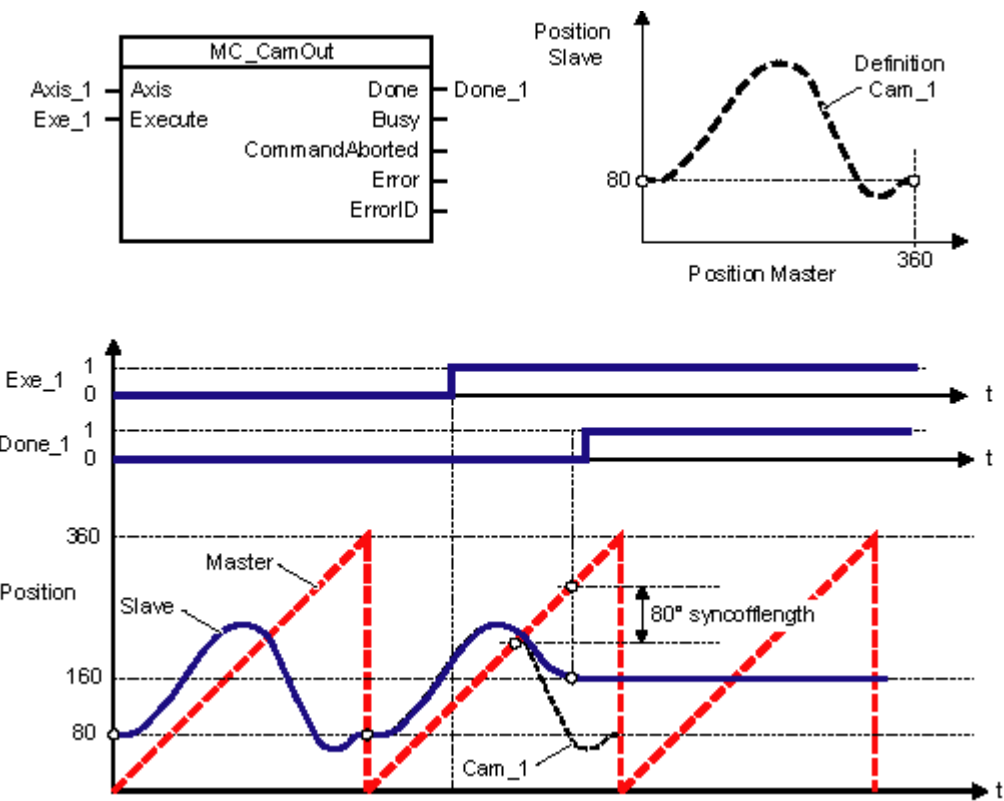
位置参考：“在去同步位置之前停止”

去同步跟随轴： 160°

该轴基于引导轴位置执行去同步。 跟随轴在引导轴覆盖的 80° 的去同步长度内去同步。
去同步的开始时间在内部进行计算。

曲线设置：“引导轴相关的同步曲线”

去同步长度： 80°



MC_CamOut — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8040	轴/外部编码器被禁用，或者设置了错误的模式	缺少用于启用跟随轴的命令所需的信号。
8043	参数值或默认值无效	S7T Config 中也可能存在组态错误。通常是组态的去同步长度为 0。
8044	命令不受工艺对象支持	在输入参数 <i>Slave</i> 中定义的工艺 DB 至少必须属于类型“同步轴”。
8045	在当前状态下不允许使用该命令	"例如，"MC_Stop"仍处于活动状态"
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是您可能无法通过状态输出对其进行监视。

ErrorID	错误消息	说明/纠正或避免错误的方法
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8083	DB 不是工艺 DB	通过输入参数 <i>Slave</i> 指定的 DB 未找到或不是工艺 DB。
8084	工艺 DB 无效	- 在通过输入参数 <i>Slave</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Slave</i> 更改 DB 号。 - 用户在输入参数 <i>Slave</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management) 中删除工艺 DB，然后创建一个新的工艺 DB。

6.3.7 FB 441 MC_CamInSuperImposed — 启动叠加凸轮传动

使用 FB 441“MC_CamInSuperImposed”启动叠加凸轮传动

用途

- “MC_CamInSuperImposed”工艺功能用于在引导轴和跟随轴（主轴和从轴）之间启动叠加凸轮传动。
- 可以在动态参数 *Velocity, Jerk, Acceleration* 中定义跟随轴同步期间的动态响应。
- 可以对指定的凸轮进行缩放和/或平移操作。
- 同步可以是绝对的，也可以是相对的。
- 指定的凸轮盘可以执行一次，也可以循环执行。

适用对象

- 具有叠加同步对象的同步轴

先决条件

- 将引导轴组态为外部编码器、定位轴或同步轴
- 将跟随轴组态为具有叠加同步对象的同步轴
- 将引导轴设置为跟随轴的可能叠加同步对象
- 凸轮盘标记为在叠加同步对象中可用。
- 启用跟随轴以进行位置控制操作
- 跟随轴中没有活动的 MC_Stop 命令。

命令的相互作用

新命令 — 活动的单个命令 (1)

新命令 — 活动的命令 (1)

输入参数

参数	数据类型	初始值	说明
<i>Master</i>	INT	0	引导轴工艺 DB 的编号
<i>Slave</i>	INT	0	跟随轴工艺 DB 的编号
<i>CamTable</i>	INT	0	凸轮盘工艺 DB 的编号
<i>Execute</i>	BOOL	FALSE	在上升沿处启动命令
<i>MasterOffset</i>	REAL	0.0	沿引导轴坐标移动凸轮
<i>SlaveOffset</i>	REAL	0.0	沿跟随轴坐标移动凸轮
<i>MasterScaling</i>	REAL	1.0	引导轴坐标系中的凸轮缩放因子（不允许值 0）
<i>SlaveScaling</i>	REAL	1.0	跟随轴坐标系中的凸轮缩放因子（不允许值 0）
<i>MasterAbsolute</i>	BOOL	TRUE	表明凸轮盘坐标对于引导轴是绝对的还是相对的：
			TRUE: 绝对坐标
			FALSE: 相对坐标
<i>SlaveAbsolute</i>	BOOL	TRUE	表明凸轮盘坐标对于跟随轴是绝对的还是相对的
			TRUE: 绝对坐标
			FALSE: 相对坐标
<i>CyclicMode</i>	BOOL	TRUE	处理凸轮盘
			TRUE: 循环执行
			FALSE: 非循环执行
<i>Velocity</i>	REAL	-1.0	耦合从轴时的最大速度（仅当 <i>Mode</i> = 1 时有效）
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值
<i>Acceleration</i>	REAL	-1.0	耦合从轴时的最大加速度（仅当 <i>Mode</i> = 1 时有效）
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值

参数	数据类型	初始值	说明
<i>Jerk</i>	REAL	-1.0	耦合从轴时的冲击（仅当 <i>Mode</i> = 1 时有效）
			值 > 0: 使用定义的值
			值 = 0: 使用梯形运动曲线
			值 < 0: 使用默认值
<i>Mode</i>	INT	0	同步模式/耦合模式:
			值 = 0: 使用相应同步对象的默认值
			值 = 1: 时间相关的同步: 根据输入参数 <i>Velocity</i> , <i>Jerk</i> , <i>Acceleration</i> 立即同步（与在 S7T Config 中设置的同步方向的“兼容性模式”一致）
			值 = 2: 基于时间的同步: 根据输入参数 <i>Velocity</i> , <i>Jerk</i> , <i>Acceleration</i> , <i>Deceleration</i> . 立即同步。程序还将考虑同步对象的 Axis_SYNCHRONISM >“默认值”(Default) 对话框的“齿轮传动”选项卡中的设置。 它还会评估系统变量 <i>userdefault.cammingsetting.synchronizingdirection</i> 。
<i>DoneFlag</i>	INT	0	在 MCDevice DB 中生成 DoneFlag 命令 (<i>InSync</i>) 完成时, 将对 DoneFlag 的值取反

输出参数（状态输出）

参数	数据类型	初始值	说明
<i>InSync</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 到达附加凸轮传动
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令
<i>CommandAborted</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令由其它命令或因其执行过程中的错误而取消。 如果工艺 DB 的 <i>ErrorStatus</i> 中未显示错误，则该命令由后续命令取消。 如果工艺 DB 的变量 <i>ErrorStatus</i> 中指示有错误，则命令执行期间影响该工艺对象的错误将导致该命令中止。
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令启动时有错误。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
			<i>FALSE</i> : 命令启动时没有错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。

输入参数 *Mode* = 1

模数轴的同步方向： 如果 *Mode* = 1，
则“兼容性模式”处于活动状态，而不管 *S7T Config* 中的设置如何：

- 凸轮传动：
跟随轴将在最短的距离内同步。

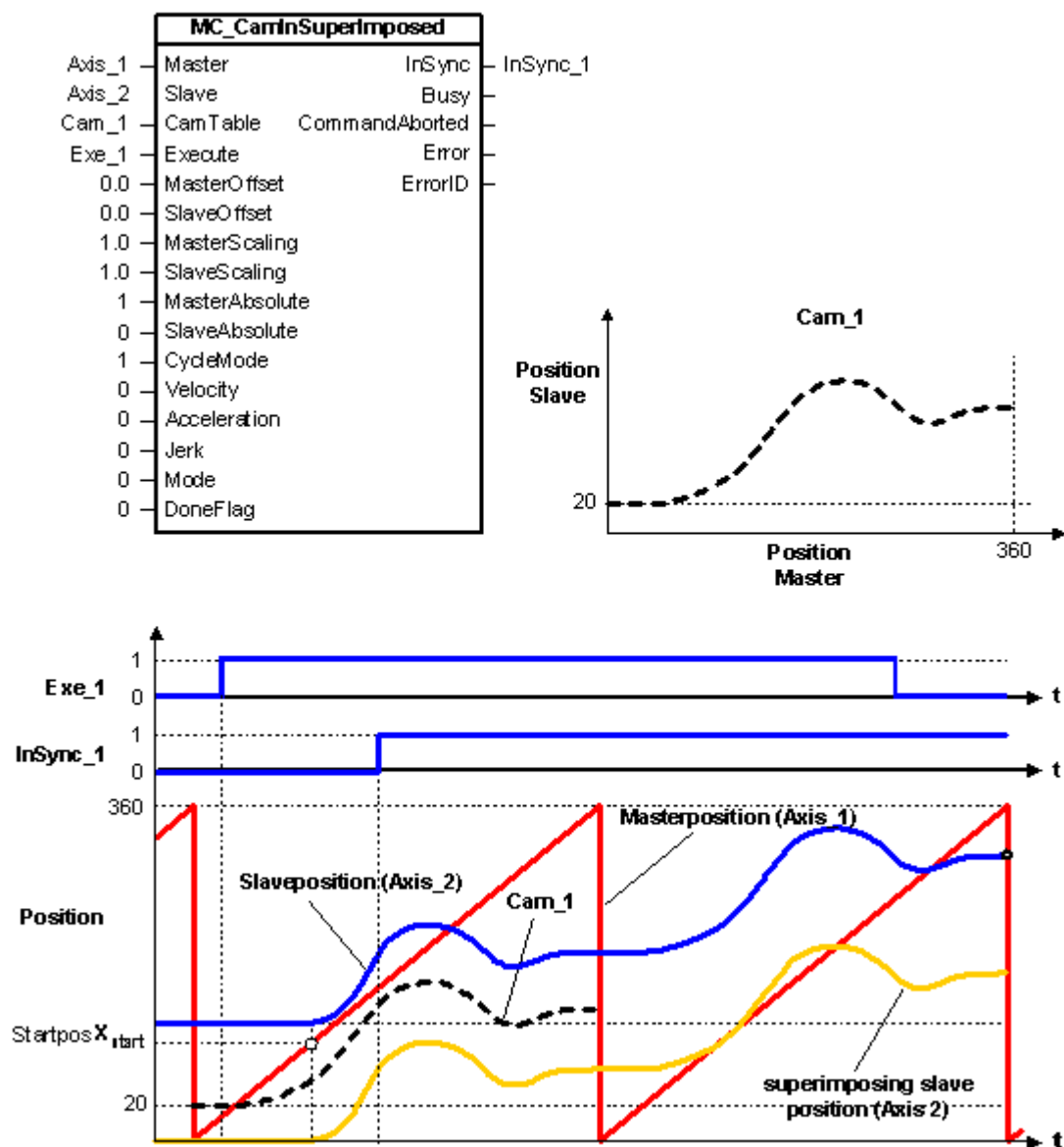
MC_CamInSuperImposed — 示例：“叠加”

以下示例基于叠加凸轮的信号图显示了叠加凸轮的工作原理。 为了获得更好的观察效果，在所显示的信号图中未执行基本同步。

在所显示信号图的开始阶段，引导轴 (*Axis_1*) 和跟随轴 (*Axis_2*) 的起始位置都大于 (>) 0。默认情况下，叠加跟随轴的起始位置在 0 处。

叠加凸轮传动通过 *Exe_1* 启动。 根据 *Mode* = 0，同步将遵循 *S7T Config* 中的默认设置进行启动（同步使用动态响应规范在引导轴位置 100 处启动）。 当引导轴 (*Axis_1*) 达到位置 100 时同步启动。 短暂延迟后，*InSync_1* 将发送信号，指示已到达凸轮传动。

橙色信号图曲线表示叠加跟随轴位置。 跟随轴位置从其起始位置开始，并与叠加跟随轴位置保持固定的偏移。



MC_CamInSuperImposed — 示例：“使用叠加凸轮传动实现基本同步操作”

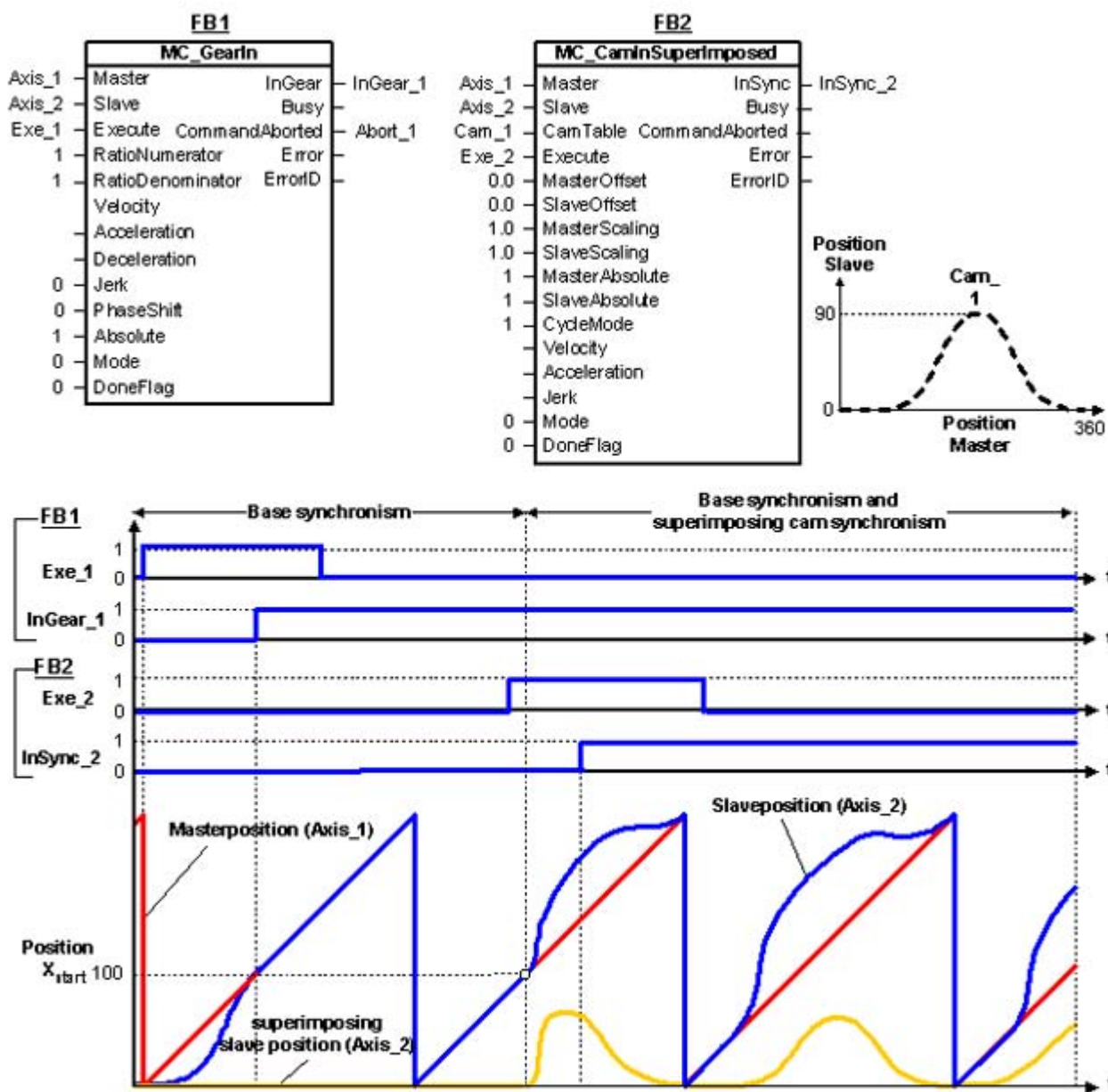
以下示例中的信号图显示了绝对基本同步操作和叠加凸轮的影响。

第一阶段 — 基本同步操作

绝对基本同步操作通过 *Exe_1* 来启动。根据“MC_GearIn”中的 *Mode = 0*，同步将遵循 S7T Config 中的默认设置进行启动（同步使用动态响应规范在引导轴位置 100 处启动）。当引导轴 (*Axis_1*) 达到位置 100 时同步启动。短暂延迟后，*InGear_1* 将发送信号，指示已到达绝对基本同步操作。

第二阶段 — 基本同步操作和叠加凸轮传动

叠加凸轮传动通过 *Exe_2* 启动。默认情况下，叠加从轴位置的起始位置在 0 处。短暂延迟后，*InSync_2* 将发送信号，指示已到达叠加凸轮传动。通过“基本从轴位置”（与主轴位置 *Axis_1* 相同）与“叠加从轴位置”相加获得从轴的位置。



MC_CamInSuperImposed — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8022	轴/外部编码器中没有实际值	例如，编码器或数据总线未准备就绪
8040	轴/外部编码器被禁用，或者设置了错误的模式	缺少运动命令所需的使能信号。 消除并确认所有排队的错误，然后启用相关的轴模式（例如，位置控制）。
8043	参数值非法	涉及数据类型为 REAL 的所有输入参数，或者输入参数 <i>DoneFlag</i> 或 <i>Mode</i>。 错误原因可能还存在于组态中，因为当块中包含错误的命令参数（已被接受为组态的默认设置）时，也会出现该错误。示例：同步长度 = 0。 例如，如果同步对象的组态中未选择指定的凸轮，则也可能发生该组态错误。
8044	命令不受工艺对象支持	例如，将命令发送至测量输入或输出凸轮。

ErrorID	错误消息	说明/纠正或避免错误的方法
8045	在当前状态下不允许使用该命令	示例: - MC_Stop 命令处于活动状态 - 指定的凸轮当前正在执行重启。等到“MC_Reset”完成 (<i>Restart = TRUE</i>)。然后再次调用“MC_CamInSuperImposed”。 - 如果输入参数 <i>PositionControl = TRUE</i>, 则通过工艺功能“MC_MoveVelocity”对从轴的运动进行速度控制。从轴必须在位置控制模式下运行, 才能进行叠加同步操作。
804A	缺少所需的对象连接	在叠加同步对象的组态中, 既没有选择在输入参数 <i>Master</i> 中定义的引导轴, 也没有选择在输入参数 <i>CamTable</i> 中定义的凸轮。
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝, 以保持第一个命令的一致性。 如果命令输出率较高, 请务必使用单独的背景数据块或再次请求命令。请注意, 尽管接受的第一个命令可能处于活动状态, 但是您可能无法通过状态输出对其进行监视。
8050	工艺未准备就绪	- 在使用断点进行测试期间 (CPU 处于 HOLD 状态), 工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。

ErrorID	错误消息	说明/纠正或避免错误的方法
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8056	由于正在向固定挡块运动而取消	轴已运动到固定挡块 (<i>InClamping = TRUE</i>)。只有新命令从固定挡块释放轴，它们才被接受。 已启动的命令不符合该条件。
807A	主设定值无效	引导轴的实际值无效。
807B	工艺对象的递归互连	同步操作中，在该输入参数中定义的引导轴已作为跟随轴处于活动状态，并且将从输入参数中定义的跟随轴直接或间接获得主设定值。
8083	DB 不是工艺 DB	通过输入参数 <i>Master</i> 、 <i>Slave</i> 或 <i>CamTable</i> 指定的 DB 未找到或不是工艺 DB。

ErrorID	错误消息	说明/纠正或避免错误的方法
8084	工艺 DB 无效	- 在通过输入参数 <i>Master</i>、<i>Slave</i> 或 <i>CamTable</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Master</i>、<i>Slave</i> 或 <i>CamTable</i> 更改 DB 号。 - 用户在工艺 DB（通过输入参数 <i>Master</i>、<i>Slave</i> 或 <i>CamTable</i> 定义）中输入的数据无效。 在“工艺对象管理”中删除该工艺 DB，然后创建一个新的工艺 DB。
808B	参数值的 REAL 格式无效	数据类型为 REAL 的输入参数的值不符合有效的浮点数格式。 检查输入参数值或背景数据块数据。 不能以浮点格式表示格式无效的值。它们以十六进制格式 (DW16# ...) 显示。

6.3.8 FB 443 MC_CamOutSuperImposed — 终止叠加凸轮传动

通过 FB 443“MC_CamOutSuperImposed”停止叠加凸轮传动

用途

- “MC_CamOutSuperImposed”工艺功能用于停止叠加凸轮传动。这不会影响基本运动或基本同步操作。
- 当根据轴组态或该工艺功能的动态响应参数执行从轴去同步时，该命令完成。

适用对象

- 具有叠加同步对象的同步轴

先决条件

- 对于叠加凸轮传动过程，该轴必须作为跟随轴（从轴）进行操作。如果没有同步操作处于活动状态，则该命令将被忽略，并且该工艺功能将报告 *Done = TRUE*。
- 未在执行 MC_Stop 命令。

命令的相互作用

新命令 — 活动的单个命令 (1)

新命令 — 活动的命令 (1)

输入参数

参数	数据类型	初始值	说明
<i>Slave</i>	INT	0	跟随轴工艺 DB 的编号
<i>Execute</i>	BOOL	FALSE	在上升沿处启动命令
<i>Deceleration</i>	REAL	-1.0	凸轮传动结束时的减速度（仅当 <i>Mode</i> = 1 时有效）：
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值
<i>Jerk</i>	REAL	-1.0	凸轮传动结束时的冲击（仅当 <i>Mode</i> = 1 时有效）：
			值 > 0: 使用定义的值
			值 = 0: 使用梯形运动曲线
			值 < 0: 使用默认值
<i>Mode</i>	INT	0	解耦模式
			值 = 0: 默认设置
			值 = 1: 根据动态响应规范
<i>DoneFlag</i>	INT	0	在 MCDevice-DB 中生成 DoneFlag。 当达到最终速度时，DoneFlag 值将取反。

输出参数（状态输出）

参数	数据类型	初始值	说明	
<i>Done</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 叠加凸轮传动终止	
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令	
<i>CommandAborted</i>	BOOL	<i>FALSE</i>	<i>TRUE</i>: 命令由其它命令或因其执行过程中的错误而取消。 如果工艺 DB 的 <i>ErrorStatus</i> 中未显示错误, 则该命令由后续命令取消。 如果工艺 DB 的变量 <i>ErrorStatus</i> 中指示有错误, 则命令执行期间影响该工艺对象的错误将导致该命令中止。	
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> :	命令启动时有错误。未执行该命令。有关原因的信息, 请参考 <i>ErrorID</i> 。
			<i>FALSE</i> :	命令启动时没有错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。	

说明

将 MC_CamInSuperImposed 和 MC_CamOutSuperImposed 同时输出到同一工艺对象可能会取消 MC_CamInSuperImposed 命令 (*CommandAborted* = *TRUE*)。因此, “MC_CamOutSuperImposed”工艺功能将报告 *Done* = *TRUE*。

请勿启动 MC_CamOutSuperImposed 命令, 除非 MC_CamInSuperImposed 已报告了同步状态 (*InGear* = *TRUE*)。

MC_CamOutSuperImposed — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8040	轴/外部编码器被禁用，或者设置了错误的模式	缺少运动命令所需的使能信号。 消除并确认所有排队的错误，然后启用相关的轴模式（例如，位置控制）。
8043	参数值非法	涉及数据类型为 <i>REAL</i> 的所有输入参数，或者输入参数 <i>Mode</i> 或 <i>DoneFlag</i>。 这也可能是 <i>S7T Config</i> 中的一个错误组态。
8044	命令不受工艺对象支持	例如，将命令发送至测量输入或输出凸轮
8045	在当前状态下不允许使用该命令	例如 <i>MC_Stop</i> 命令处于活动状态
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是您可能无法通过状态输出对其进行监视。

ErrorID	错误消息	说明/纠正或避免错误的方法
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8083	DB 不是工艺 DB	通过输入参数 <i>Slave</i> 指定的 DB 未找到或不是工艺 DB。
8084	工艺 DB 无效	- 在通过输入参数 <i>Slave</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Slave</i> 更改 DB 号。 - 用户在输入参数 <i>Slave</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management) 中删除工艺 DB，然后创建一个新的工艺 DB。
808B	参数值的 REAL 格式无效	数据类型为 REAL 的输入参数的值不符合有效的浮点数格式。 检查输入参数值或背景数据块数据。不能以浮点格式表示格式无效的值。它们以十六进制格式 (DW16# ...) 显示。

6.3.9 FB 424 MC_Phasing — 更改引导轴和跟随轴之间的相移

通过调用 FB424“MC_Phasing”来更改引导轴和跟随轴之间的相移

用途

- “MC_Phasing”工艺功能用于产生引导轴相对于跟随轴的相移。引导轴的实际位置不受影响。
- 对凸轮传动模式下跟随轴操作的影响与水平凸轮位移相同。
- 相参考可以是相对的，也可以是绝对的。
- 在引导轴处于停止状态或运动状态时，均可以启动该功能。
- 通过输入参数 *Velocity*, *Jerk*, *Acceleration*, *Deceleration* 定义运动的动态响应。

适用对象

- 在凸轮传动或齿轮传动模式下运行的同步轴

先决条件

- 跟随轴中没有活动的 MC_Stop 命令。
- 该轴必须作为跟随轴在凸轮传动或齿轮传动模式下进行操作。

命令的相互作用

新命令 — 活动的单个命令 (1)

新命令 — 活动的命令 (1)

输入参数

参数	数据类型	初始值	说明
<i>Master</i>	INT	0	引导轴工艺 DB 的编号
<i>Slave</i>	INT	0	跟随轴工艺 DB 的编号
<i>Execute</i>	BOOL	<i>FALSE</i>	在上升沿处启动命令
<i>PhaseShift</i>	REAL	0.0	相移
<i>Velocity</i>	REAL	-1.0	用于设置相位置的最大速度:
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值
<i>Acceleration</i>	REAL	-1.0	用于设置相位置的加速度:
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值
<i>Deceleration</i>	REAL	-1.0	用于设置相位置的减速度:
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值
<i>Jerk</i>	REAL	-1.0	用于设置相位置的冲击:
			值 > 0: 使用定义的值
			值 = 0: 使用梯形运动曲线
			值 < 0: 使用默认值
<i>Absolute</i>	BOOL	<i>TRUE</i>	相移:
			值 = 0: 相对于当前相:
			值 = 1: 对于当前相是绝对的 (请参阅 MC_Phasing — 示例)
<i>DoneFlag</i>	INT	0	在 MCDevice DB 中生成 DoneFlag

输出参数（状态输出）

参数	数据类型	初始值	说明	
<i>Done</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 到达了所需的相位置	
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令	
<i>CommandAborted</i>	BOOL	<i>FALSE</i>	<i>TRUE</i>: 命令由其它命令或因其执行过程中的错误而取消。 如果工艺 DB 的 <i>ErrorStatus</i> 中未显示错误, 则该命令由后续命令取消。 如果工艺 DB 的变量 <i>ErrorStatus</i> 中指示有错误, 则命令执行期间影响该工艺对象的错误将导致该命令中止。	
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> :	命令启动时 有错误 。未执行该命令。有关原因的信息, 请参考 <i>ErrorID</i> 。
			<i>FALSE</i> :	命令启动时 没有错误 。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。	

说明

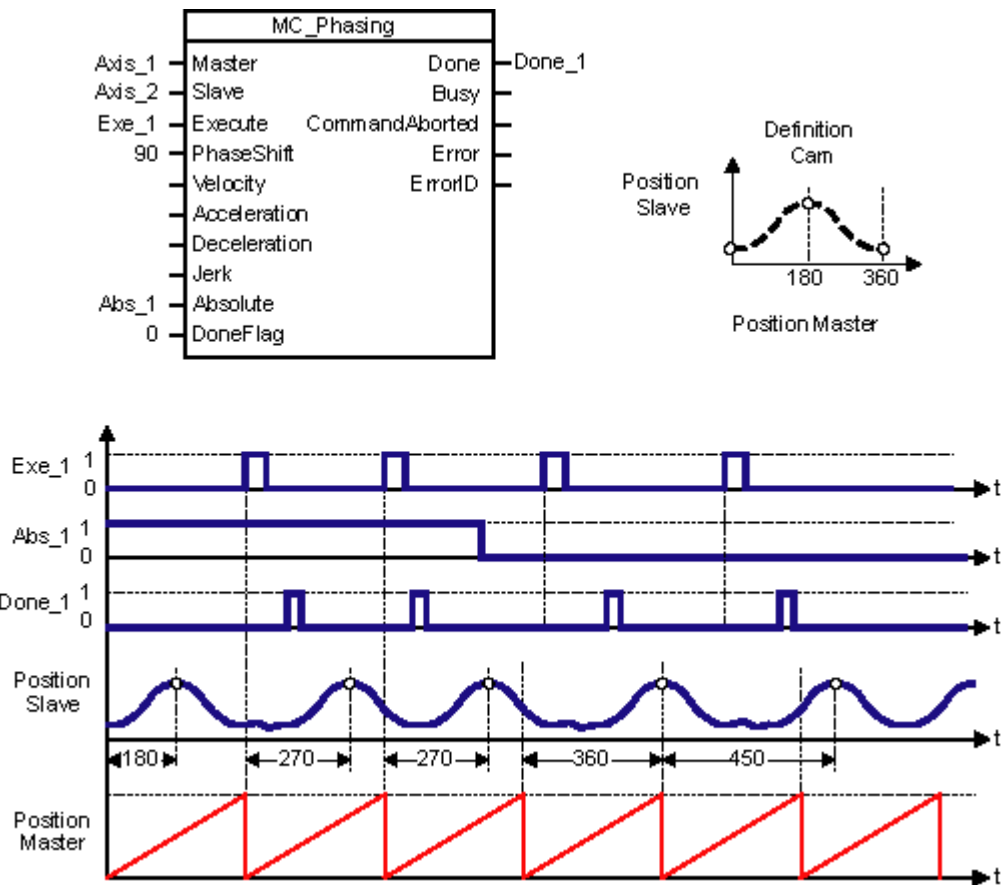
动态响应值 *Jerk*, *Acceleration* 和 *Velocity* 是绝对值。为了能够消除正向偏移, 跟随轴的驱动速度必须比当前速度快。因此, 应确保动态响应值足够大。

MC_Phasing — 示例

使用“MC_Phasing”工艺功能在使用凸轮传动期间对凸轮的相位置进行平移。

Absolute = 1 设置将凸轮移位定义为绝对值（与凸轮定义相比）。使用相同的“PhaseShift”值重复移位不会产生任何影响。

当 *Absolute* = 0 时，移位更改是相对的，即每个命令都将相位置移位 *PhaseShift* 中定义的值。



MC_Phasing — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一个周期内调用这些工艺功能，直到输出参数 <i>Done</i>, <i>CommandAborted</i> 或 <i>Error</i> 中有一个为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8040	轴/外部编码器被禁用，或者设置了错误的模式	缺少用于启用跟随轴的命令所需的信号。
8043	参数值非法	涉及数据类型为 <i>REAL</i> 的所有输入参数，或者输入参数 <i>DoneFlag</i> 。
8044	命令不受工艺对象支持	例如，将命令发送至输出凸轮
8045	在当前状态下不允许使用该命令	没有同步操作处于活动状态
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是您可能无法通过状态输出对其进行监视。

ErrorID	错误消息	说明/纠正或避免错误的方法
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 无法通过工艺功能的输出参数监视最初启动的命令。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8056	由于正在向固定挡块运动而取消	轴已运动到固定挡块 (<i>InClamping = TRUE</i>)。只有新命令从固定挡块释放轴，它们才被接受。 已启动的命令不符合该条件。
8079	没有带有指定主轴的同步操作处于活动状态	指定的主轴和从轴都未在同步操作中。 调用该功能前使相关的轴同步。
8083	DB 不是工艺 DB	通过输入参数 <i>Master</i> 或 <i>Slave</i> 指定的 DB 未找到或不是工艺 DB。

ErrorID	错误消息	说明/纠正或避免错误的方法
8084	工艺 DB 无效	- 在通过输入参数 <i>Master</i> 或 <i>Slave</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Master</i> 或 <i>Slave</i> 更改 DB 号。 - 用户在工艺 DB（通过输入参数 <i>Master</i> 或 <i>Slave</i> 定义）中输入的数据无效。 在“工艺对象管理”中删除该工艺 DB，然后创建一个新的工艺 DB。
808B	参数值的 REAL 格式无效	数据类型为 REAL 的输入参数的值不符合有效的浮点数格式。 检查输入参数值或背景数据块数据。 不能以浮点格式表示格式无效的值。它们以十六进制格式 (DW16# ...) 显示。

6.3.10 FB 444 MC_PhasingSuperImposed — 更改叠加相移

通过 FB 444“MC_PhasingSuperImposed”更改叠加相移

用途

- MC_PhasingSuperImposed 工艺功能用于在引导轴位置和跟随轴位置之间设置一个叠加相移。根据跟随轴位置定义相移值。叠加相移将影响跟随轴。它不影响引导轴的位置。
- 对凸轮传动模式下跟随轴操作的影响与水平凸轮位移相同。
- 相参考可以是相对的，也可以是绝对的。
- 在引导轴处于停止状态或运动状态时，均可以启动该功能。
- 在输入参数 *Velocity*、*Jerk*、*Acceleration*、*Deceleration* 中定义运动的动态响应。

适用对象

- 带有在凸轮传动和齿轮传动模式下操作的叠加同步对象的同步轴

先决条件

- 跟随轴中没有活动的 MC_Stop- 命令
- 该轴必须作为跟随轴（从轴）在叠加凸轮传动或齿轮传动模式下进行操作。

命令的相互作用

新命令 — 活动的单个命令 (1)

新命令 — 活动的命令 (1)

输入参数

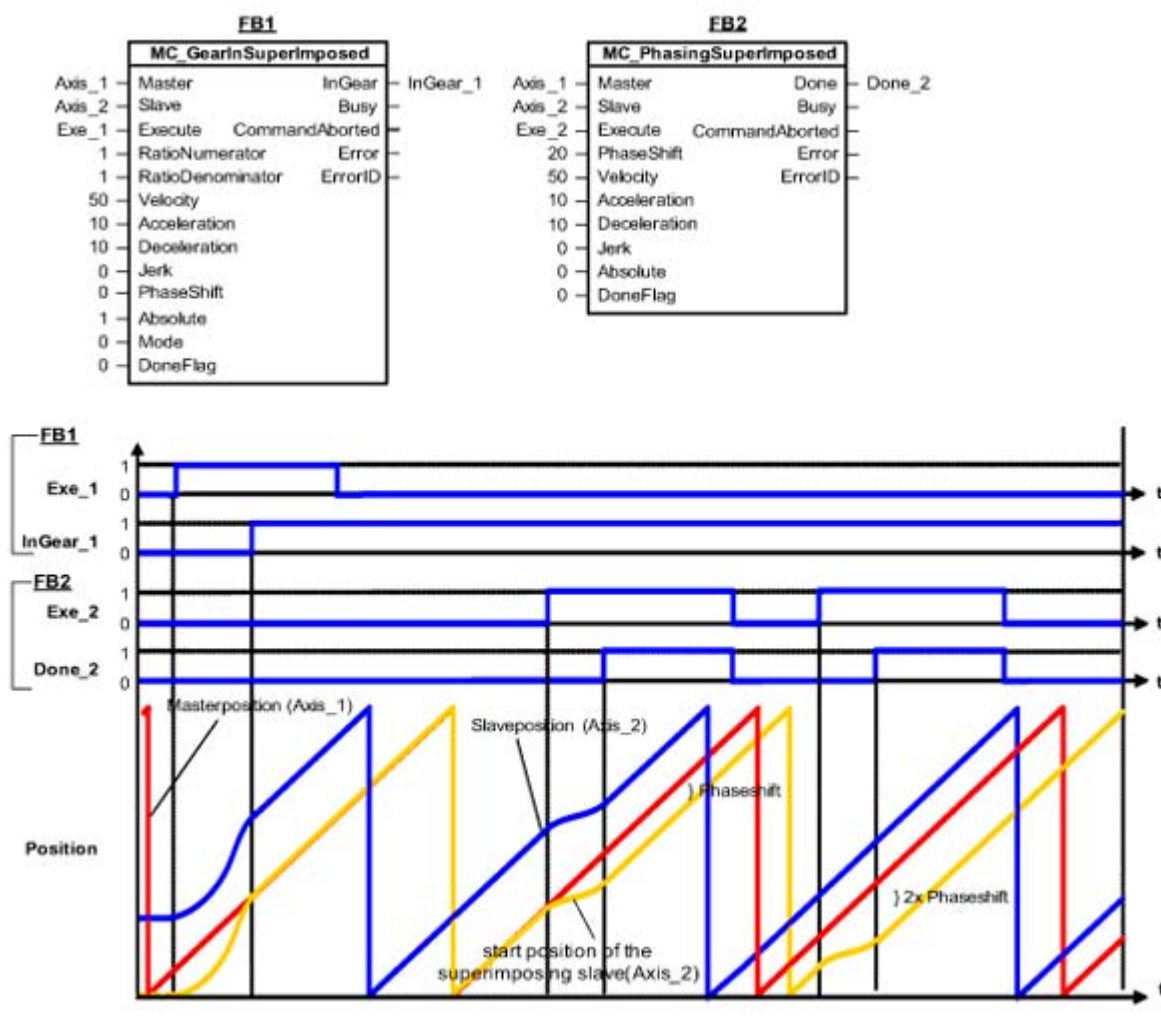
参数	数据类型	初始值	说明
<i>Master</i>	INT	0	引导轴工艺 DB 的编号
<i>Slave</i>	INT	0	跟随轴工艺 DB 的编号
<i>Execute</i>	BOOL	FALSE	在上升沿处启动命令
<i>PhaseShift</i>	REAL	0.0	相移
<i>Velocity</i>	REAL	-1.0	用于设置相位置的最大速度
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值
<i>Acceleration</i>	REAL	-1.0	用于设置相位置的最大加速度
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值
<i>Deceleration</i>	REAL	-1.0	用于设置相位置的最大减速度
			值 > 0: 使用定义的值
			值 = 0: 不允许
			值 < 0: 使用默认值
<i>Jerk</i>	REAL	-1.0	用于设置相位置的最大冲击
			值 > 0: 使用定义的值
			值 = 0: 使用梯形运动曲线
			值 < 0: 使用默认值
<i>Absolute</i>	BOOL	TRUE	相移
			值 = 0: 相对
			值 = 1: 对于当前相是绝对的
<i>DoneFlag</i>	INT	0	在 MCDevice DB 中生成 DoneFlag

输出参数（状态输出）

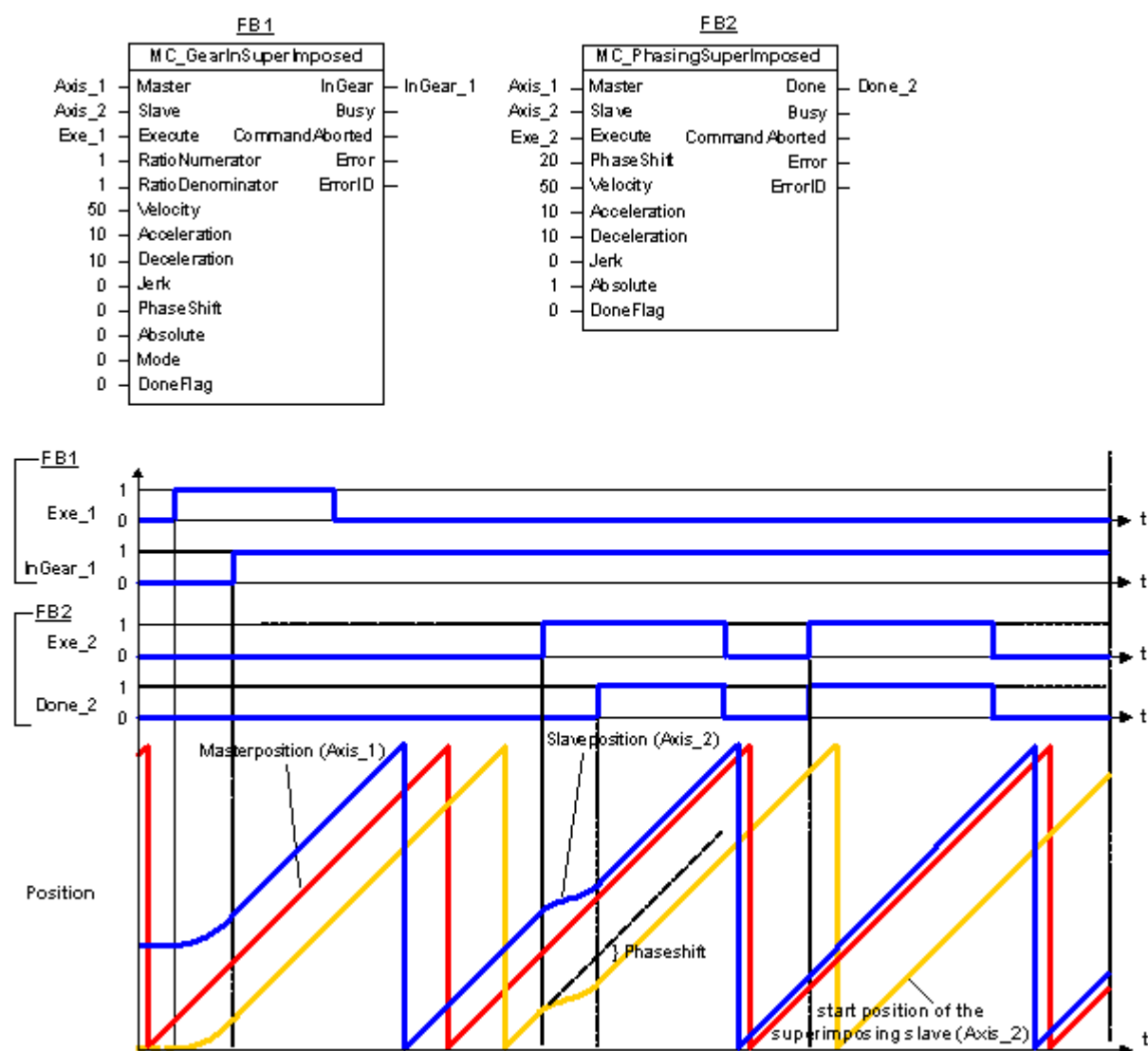
参数	数据类型	初始值	说明	
<i>Done</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 到达了所需的相位置	
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令	
<i>CommandAborted</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令由其它命令或因其执行过程中的错误而取消。 如果工艺 DB 的 <i>ErrorStatus</i> 中未显示错误，则该命令由后续命令取消。 如果工艺 DB 的变量 <i>ErrorStatus</i> 中指示有错误，则命令执行期间影响该工艺对象的错误将导致该命令中止。	
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> :	命令启动时 有错误 。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
			<i>FALSE</i> :	命令启动时 没有错误 。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。	

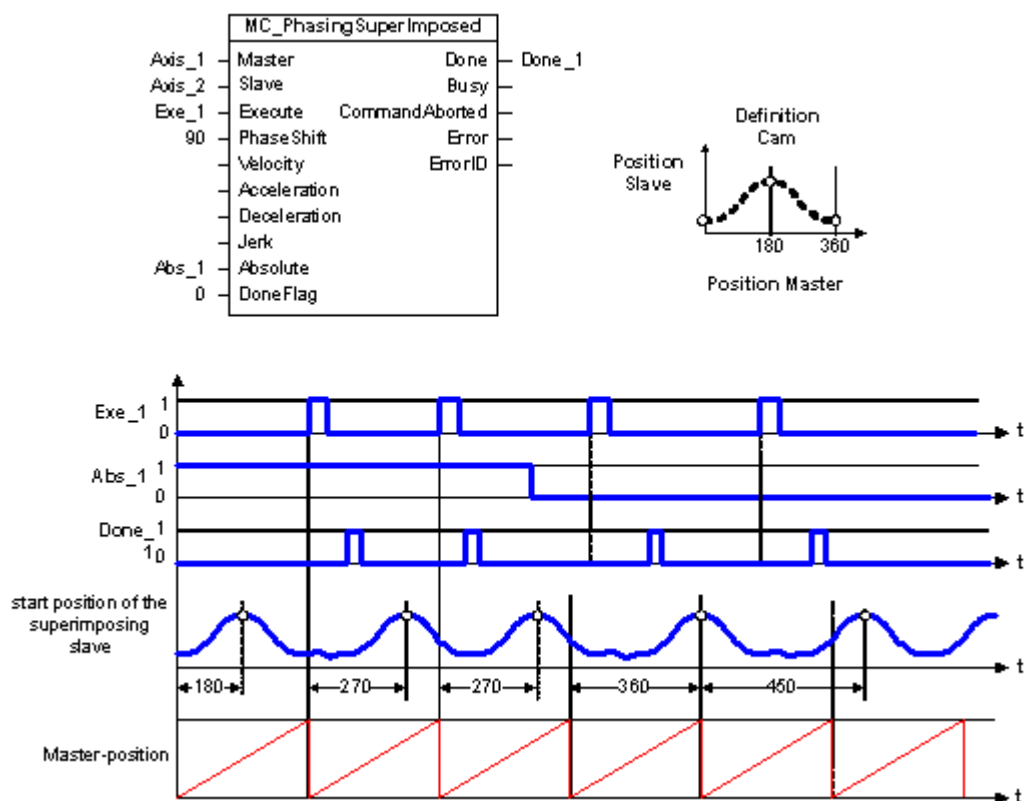
说明
动态响应值 *Jerk*, *Acceleration* 和 *Velocity* 是绝对值。为了能够消除正向偏移，跟随轴的驱动速度必须比当前速度快。因此，应确保动态响应值足够大。

MC_PhasingSuperImposed — 示例



6.3 工艺功能 — 齿轮传动/凸轮传动





MC_PhasingSuperImposed — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8040	轴/外部编码器被禁用，或者设置了错误的模式	缺少运动命令所需的使能信号。 消除并确认所有排队的错误，然后启用相关的轴模式（例如，位置控制模式）。
8043	参数值非法	涉及数据类型为 REAL 的所有输入参数，或者输入参数 <i>DoneFlag</i> 。
8044	命令不受工艺对象支持	例如，将命令发送给测量传感器或凸轮
8045	在当前状态下不允许使用该命令	例如 MC_Stop 命令处于活动状态
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是您可能无法通过状态输出对其进行监视。

ErrorID	错误消息	说明/纠正或避免错误的方法
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8056	由于正在向固定挡块运动而取消	轴已运动到固定挡块 (<i>InClamping = TRUE</i>)。只有新命令从固定挡块释放轴，它们才被接受。 已启动的命令不符合该条件。
8079	没有带有指定主轴的同步操作处于活动状态	指定的主轴和从轴都未在同步操作中。 调用该功能前使相关的轴同步。
8083	DB 不是工艺 DB	通过输入参数 <i>Master bzw. Slave</i> 指定的 DB 未找到或不是工艺 DB。

ErrorID	错误消息	说明/纠正或避免错误的方法
8084	工艺 DB 无效	- 在通过输入参数 <i>Master</i> 或 <i>Slave</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Master</i> 或 <i>Slave</i> 更改 DB 号。 - 用户在工艺 DB（通过输入参数 <i>Master</i> 或 <i>Slave</i> 定义）中输入的数据无效。 在“工艺对象管理”中删除该工艺 DB，然后创建一个新的工艺 DB。
808B	参数值的 REAL 格式无效	数据类型为 REAL 的输入参数的值不符合有效的浮点数格式。 检查输入参数值或背景数据块数据。不能以浮点格式表示格式无效的值。它们以十六进制格式 (DW16# ...) 显示。

6.4 工艺功能 — 凸轮盘

6.4.1 FB 434 MC_CamClear — 删除凸轮

使用 FB 434“MC_CamClear”删除凸轮

用途

- “MC_CamClear”工艺功能将删除凸轮所有定义的插补点或区段，并将凸轮设置为编辑模式。

适用对象

- 凸轮

先决条件

- 删除凸轮时，该凸轮一定不能处于使用状态。
- 不必激活 MC_CamSectorAdd 命令（限制仅适用于集成工艺的固件版本 V3.0.x）。

超驰命令

MC_CamClear 命令不能由任何其它命令取消。

MC_CamClear 命令不取消任何其它命令。

输入参数

参数	数据类型	初始值	说明
<i>CamTable</i>	INT	0	工艺 DB 的编号
<i>Execute</i>	BOOL	<i>FALSE</i>	在上升沿处启动删除操作

输出参数（状态输出）

参数	数据类型	初始值	说明
<i>Done</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 凸轮扇区已被删除
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令启动时有错误。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
			<i>FALSE</i> : 命令启动时没有错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。

说明

删除具有大量插补点或多项式的凸轮可能要花费一些时间。在此期间，其它工艺对象无法接受任何新命令（甚至包括 **MC_Stop** 命令）。请注意，在此期间，如果未定义终止（例如，“**MC_MoveVelocity**”）就不能取消命令。

如果 **S7T Config** 中命令监视时间的设置过短，则删除操作可能会导致超时，从而导致 **T-CPU** 进入 **STOP** 模式。

MC_CamClear — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8044	命令不受工艺对象支持	例如，命令要求有一个轴
8045	在当前状态下不允许使用该命令	- 凸轮处于使用状态 "MC_CamSectorAdd"命令正在将插补点插入凸轮。 （此原因仅适用于固件版本为 V3.0.x 的集成工艺）。
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是您可能无法通过状态输出对其进行监视。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。

ErrorID	错误消息	说明/纠正或避免错误的方法
8051	读/写访问被拒绝。	对相关工艺对象的读/写访问失败。 例如，如果在同步操作中主动跟踪凸轮时启动“MC_CamClear”，则可能会出现此错误。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： • 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 • 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8083	DB 不是工艺 DB	通过输入参数 <i>Axis</i> 指定的 DB 未找到或不是工艺 DB。
8084	工艺 DB 无效	• 在通过输入参数 <i>Axis</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Axis</i> 更改 DB 号。 • 用户在输入参数 <i>Axis</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management) 中删除工艺 DB，然后创建一个新的工艺 DB。

6.4.2 FB 435 MC_CamSectorAdd — 添加凸轮扇区

使用 FB 435“MC_CamSectorAdd”添加凸轮扇区

用途

- “MC_CamSectorAdd”工艺功能可以将新插补点或多项式添加到凸轮（每条命令不限制数量）。

适用对象

- 凸轮

先决条件

- 凸轮处于编辑模式下
通过调用“MC_CamClear”工艺功能启动编辑模式，通过调用“MC_CamInterpolate”工艺功能结束凸轮插补。凸轮便可重新用于凸轮传动。

超驰命令

MC_CamSectorAdd 命令不能由任何其它命令取消。

MC_CamSectorAdd 命令不取消任何其它命令。

输入参数

参数	数据类型	初始值	说明
<i>CamTable</i>	INT	0	凸轮盘工艺 DB 的编号。
<i>Execute</i>	BOOL	FALSE	在上升沿处开始
<i>Data</i>	ANY	-	包含要插入的凸轮区段（插补点或多项式）的数据区域。
<i>Table</i>	INT	0	数据区域的规范：
			值 = 0: 多项式（数学函数）
			值 = 1: 插补点表

输出参数（状态输出）

参数	数据类型	初始值	说明
<i>Done</i>	BOOL	FALSE	TRUE: 凸轮扇区的更改完成
<i>Busy</i>	BOOL	FALSE	TRUE: 正在执行命令
<i>Error</i>	BOOL	FALSE	TRUE: 命令启动时有错误。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
			FALSE: 命令启动时没有错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。

输入参数 **Data:** 定义多项式（数学函数）

说明

注意事项:

- 添加区段时，只能使用最多三阶的数学函数。由于浮点值的精度所限，较高阶的任何系数都可能导致不精确。
- 在 T-CPU 中对定义为多项式的区段重新计算。随后读出 S7T Config 的值时，可能会返回偏离系数。起点和终点的 Y 坐标也可能同样被交换。
但是，凸轮形状将被保留。

数学函数由以下公式导出：

$$Y = A_0 + A_1 \cdot X + A_2 \cdot X^2 + A_3 \cdot X^3 + A_4 \cdot X^4 + A_5 \cdot X^5 + A_6 \cdot X^6 + B_0 \cdot (\sin(B_1 \cdot X) + B_2)$$

相对 字节地址	数据类型	变量	含义
0	INT	类型	凸轮类型（0 = 多项式）
2	INT	备用	-
4	REAL	X — 最小	X 坐标（最小值）
8	REAL	Y — 最小	Y 坐标（最小值）
12	REAL	X — 最大	X 坐标（最大值）
16	REAL	Y — 最大	Y 坐标（最大值）
20	REAL	A0	0 阶系数
24	REAL	A1	1 阶系数
28	REAL	A2	2 阶系数
32	REAL	A3	3 阶系数
36	REAL	A4	4 阶系数
40	REAL	A5	5 阶系数
44	REAL	A6	6 阶系数
48	REAL	B0	三角元素的系数 0
52	REAL	B1	三角元素的系数 1
56	REAL	B2	三角元素的系数 2

示例:

```
CALL  "MC_CamSectorAdd" , DB435
CamTable:=
Execute :=
Data    :=P#DB20.DBX 16.0 BYTE 60
Table   :=0
Done    :=
Error   :=
ErrorID :=
```

在本示例中，ANY 指针指向从 DB20 中的地址 16 处开始的范围。多项式的数据结构是固定的，且具有 60 字节的既定长度。

说明

可以通过使用“S7-Tech”库中的 UDT 20 来保存您的编辑工作。UDT 20 包含多项式的数据结构。通过重复此结构，在一次调用中最多可以添加 4 个多项式。

输入参数 Data: 插补点表

使用插补点表创建凸轮时，将从用户程序的数据区域获取 X 值和 Y 值。插补点表至少包含一个插补点。每个插补点均由两个 REAL 值组成。每次调用最多可以传送 30 个插补点。

示例:

```
CALL "MC_CamSectorAdd" , DB435
  CamTable:=
  Execute :=
  Data     :=P#DB10.DBX 6.0 BYTE 64
  Table    :=1
  Done     :=
  Busy     :=
  Error    :=
  ErrorID  :=
```

在我们的示例中，ANY 指针指向从 DB10 中的地址 6 开始的插补点的范围，并包括八个插补点（八对值）。

相对 字节地址	数据类型	变量	含义
0	REAL	X1	插补点 1 的 X 坐标
4	REAL	Y1	插补点 1 的 Y 坐标
8	REAL	X2	插补点 2 的 X 坐标
12	REAL	Y2	插补点 2 的 Y 坐标
...	...	...	...
	REAL	Xn	最后一个插补点的 X 坐标
	REAL	Yn	最后一个插补点的 Y 坐标

缺点: 与多项式定义相比，凸轮描述所需的数据量较大，因此基于插补点表创建凸轮花费的时间较长。

说明**固件版本为 V3.1.x 或更高版本的集成工艺**

通过该固件版本，可以在一个命令执行周期内添加 MC_CamSectorAdd 命令的所有插补点和多项式。通过运行若干 MC_CamSectorAdd 命令，可以将 30 个以上的插补点或 4 个以上的多项式添加到凸轮。

固件版本为 V3.0.x 的集成工艺

通过此固件版本，插补点和多项式的插入被分配到若干命令执行周期。

如果正在运行一个 MC_CamSectorAdd 命令，则不允许启动其它 MC_CamSectorAdd、MC_CamClear 或 MC_CamInterpolate 命令。

凸轮区段系数的作用

系数 $A0$ 到 $A6$ 和 $B0$ 到 $B2$ 定义凸轮区段的标准曲线特性。通过插补将间隔 $0.0 \leq X \leq 1.0$ 内的凸轮的范围转换为 $\{X1, Y1\}$ 和 $\{X2, Y2\}$ 定义的范围。

$$Y(X) = 1.0 + 2.0 \cdot \sin(2\pi \cdot X)$$

$$A0 = 1.0$$

$$A1 = A2 = A3 = A4 = A5 = A6 = 0.0$$

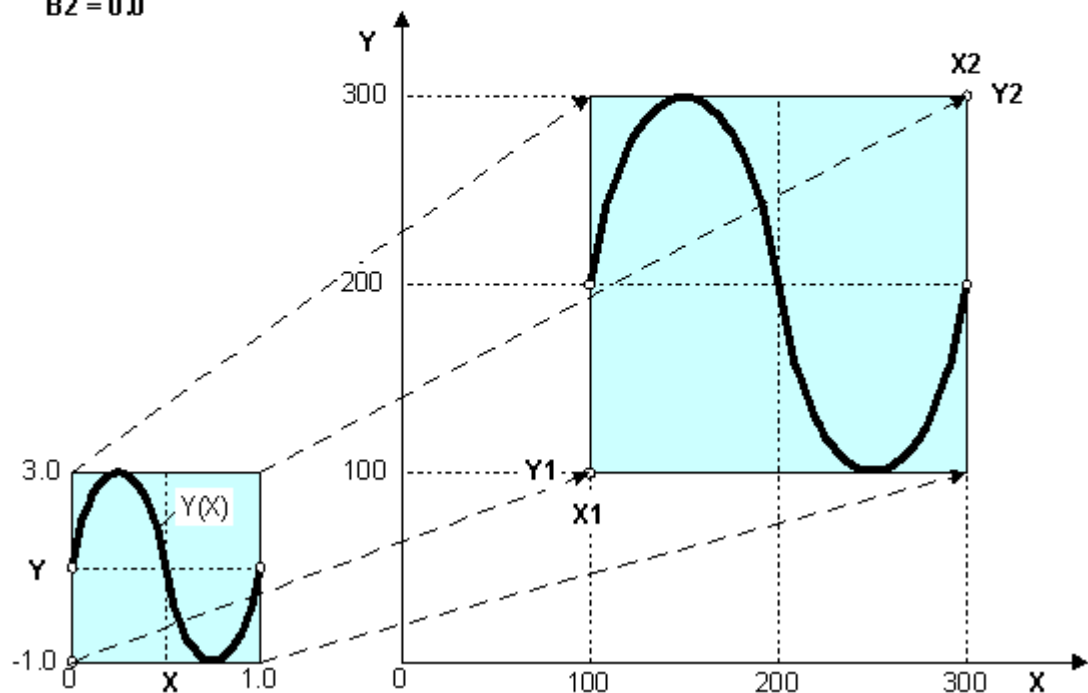
$$B0 = 2.0$$

$$B1 = 6.2831$$

$$B2 = 0.0$$

$$X1 = 100 \quad Y1 = 100$$

$$X2 = 300 \quad Y2 = 300$$



定义凸轮盘

在**插入凸轮 (Insert Cam)** 对话框中，您可以选择定义基于插补点的凸轮或基于多项式的凸轮。

- 基于插补点的定义

插补点在插补点表中以 $P = P(x,y)$ 的形式表示。与数值对的输入顺序无关。它们在域中按升序自动排序。

S7T Config 根据组态的插补类型插补凸轮。

- 基于多项式/区段的定义

各种多项式按照“凸轮机械运动法则，VDI 指令 2143”予以说明。多项式的最大阶数为 6。多项式还可以包含三角函数。

凸轮的定义方法

两种定义方法，即基于插补点或多项式，各有其自己的优点和缺点。您的应用决定了哪种方法可以满足要求。

一个凸轮中结合两种方法的情况只能在使用 MC_CamSectorAdd 工艺功能的运行时间中出现。

	基于插补点的定义	基于多项式的定义
优点	- 定义简单 - 任何算法均可以通过插补点进行映射 - 通过 Teach-In 绘图 - HMI 的界面简单	- 用于定义的数据量少 - 按照 VDI 的标准转换..... - 外形非常精确，转换连续
缺点	需要大量插补点才能获得平滑的外形	需要进行复杂的系数计算

MC_CamSectorAdd — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正由其它过程使用。	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。

ErrorID	错误消息	说明/纠正或避免错误的方法
8006	存储器不足	集成工艺的存储器不足。 减少要插入的插补点或多项式的数量。
800B	无法保留传输缓冲区	命令的传输缓冲区不足。可能的补救措施： - 传输缓冲区已被其它的活动工艺功能占用。防止同时执行以下工艺功能： “MC_ReadCamTrackData” “MC_WriteCamTrackData” “MC_CamSectorAdd” “MC_ReadPeriphery” “MC_WritePeriphery” “MC_ReadRecord” “MC_WriteRecord” “MC_ReadDriveParameter” “MC_WriteDriveParameter” - 减少插补点/多项式的数量。
8043	参数值非法	例如， <i>Table = 167</i> 或多项式值非法（ <i>Xmin</i> 和 <i>Xmax</i> 相等）。
8044	命令不受工艺对象支持	例如，将命令发送至输出凸轮。
8045	在当前状态下不允许使用该命令	“MC_CamSectorAdd”处于活动状态，且正在将插补点插入凸轮。等待其它“MC_CamSectorAdd”的“完成”位，然后重新触发此命令。 （只有使用固件版本为 V3.0.x. 的集成工艺才会出现此错误。）
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是您可能无法通过状态输出对其进行监视。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8051	读/写访问被拒绝。	对相关工艺对象的读/写访问失败。 例如，如果在同步操作中主动跟踪凸轮，则可能会出现错误。

ErrorID	错误消息	说明/纠正或避免错误的方法
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8062	不允许访问已插补的凸轮	该凸轮已被插补。因此，您无法在此凸轮上插入任何新插补点或多项式。 补救措施：使用“MC_CamClear”删除凸轮的内容，然后对其重新创建。
8083	DB 不是工艺 DB	通过输入参数 <i>Axis</i> 指定的 DB 未找到或不是工艺 DB。
8084	工艺 DB 无效	- 在通过输入参数 <i>Axis</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Axis</i> 更改 DB 号。 - 用户在输入参数 <i>Axis</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management) 中删除工艺 DB，然后创建一个新的工艺 DB。
8088	DB 无效 (ANY 指针)	ANY 指针指向无效的数据区域。 - 找不到数据块 - 找不到指定的起始地址 - 数据区域过短

ErrorID	错误消息	说明/纠正或避免错误的方法
8089	数据记录长度无效 (ANY 指针)	对于数学函数, 必须传送 60 个字节; 对于插补点表, 必须传送 8 个字节的倍数。插补点由数据类型为 REAL (4 个字节) 的 X 值和 Y 值组成。
808A	DB 中的 REAL 值无效	传送到工艺的 DB 包含至少一个与允许的 REAL 格式规范不兼容的值。 检查所有 DB 值并根据需要进行调整。
808C	指针格式无效或不受支持	ANY 指针包含无效的定义。 接受的区域: I、O、M、DB、DI 接受的数据类型: BYTE、CHAR、WORD、INT、DWORD、DINT、REAL

6.4.3 FB 436 MC_CamInterpolate — 插补凸轮

使用 FB 436“MC_CamInterpolate”插补凸轮

用途

- “MC_CamInterpolate”工艺功能可以插补凸轮。对凸轮插补后，该凸轮可用于凸轮传动。
- 根据指定的插补类型，插补将填补凸轮插补点和多项式之间的间隙。

适用对象

- 凸轮

先决条件

- 凸轮处于编辑模式下

通过调用“MC_CamClear”工艺功能启动编辑模式，通过调用“MC_CamInterpolate”工艺功能结束凸轮插补。凸轮便可重新用于凸轮传动。

条件

- 适用于插补点的规则：
 - 创建的最后一个叠加插补点优先级较高。
 - 值 $1E-4$ 用于在系统中进行连续性检查。
- 适用于区段的规则：
 - 通过插补凸轮来填补区段之间的间隙。
 - 如果区段重叠，将使用区段起点；
从该位置开始截取上一区段。
- 适用于混合模式凸轮的规则（区段和插补点）：
 - 起点与纯区段相比优先级较高。
- 根据输入参数 *CamMode* 中的设置计算凸轮。系统可能会相应地更改凸轮边界。

超驰命令

MC_CamInterpolate 命令不能由任何其它命令取消。

MC_CamInterpolate 命令不取消任何其它命令。

输入参数

参数	数据类型	初始值	说明
<i>CamTable</i>	INT	0	工艺 DB 的编号
<i>Execute</i>	BOOL	<i>FALSE</i>	在上升沿处启动
<i>Mode</i>	INT	0	插补类型:
			值 = 0: 线性插补
			值 = 1: 三次样条函数 (通过插补点或凸轮区段的曲线特性)
			值 = 2: B 样条 (通过插补点或凸轮区段的曲线特性)
<i>CamMode</i>	INT	0	凸轮类型: 可以通过输入参数 <i>CamMode</i> 定义插补处理在凸轮沿的不规则性的方式。
			值 = 0: 相对循环 — 连续速度: 凸轮的起点和终点被插补, 以便在循环模式下操作凸轮时可以在跟随轴上防止任何速度跳跃。凸轮在其边沿区域以恒定速度运行。
			值 = 1: 绝对循环 — 连续位置和速度: 凸轮的起点和终点相同。同时, 将对凸轮进行插补, 以便在循环模式下操作凸轮时可以防止任何速度跳跃。
			值 = 2: 非循环 — 边沿点不恒定 凸轮的起点和终点没有改变。凸轮的运行范围不受 <i>StartPoint</i> 和 <i>EndPoint</i> 输入参数的限制。 运行系统使用指定的凸轮 (包括沿上的所有中断), 即使在循环模式下操作该凸轮。 但是, 机械系统和驱动器的加速度限制及惯性起决定作用。
			将 <i>CamMode</i> 设置为 = 0 或 1 以在循环模式下对凸轮进行操作。如果凸轮不在循环模式下操作, 则设置 <i>CamMode</i> = 2。

参数	数据类型	初始值	说明
<i>StartPoint</i>	REAL	<i>0.0</i>	起点（凸轮的最小 X 值） （最大值： 3.402 823E+38）
<i>EndPoint</i>	REAL	<i>0.0</i>	终点（凸轮的最大 X 值） （最大值： 3.402 823E+38）

对于“绝对循环”或“相对循环”设置，只有当凸轮的最后一个定义位置与域结束之间，或者域开始与第一个定义值之间有充足的间隔时，才能在运行系统中对凸轮进行计算。

输出参数（状态输出）

参数	数据类型	初始值	说明
<i>Done</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 凸轮仅可用于同步操作（如果插补了一个空凸轮，也将指示 <i>TRUE</i> ）
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令启动时有错误。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
			<i>FALSE</i> : 命令启动时没有错误。
<i>ErrorID</i>	WORD	<i>0</i>	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。

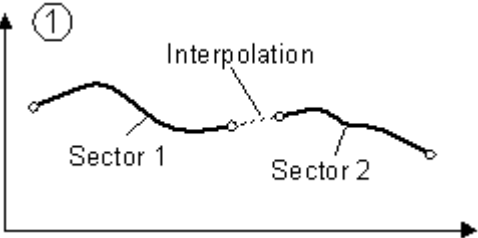
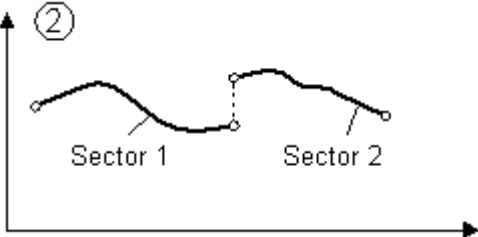
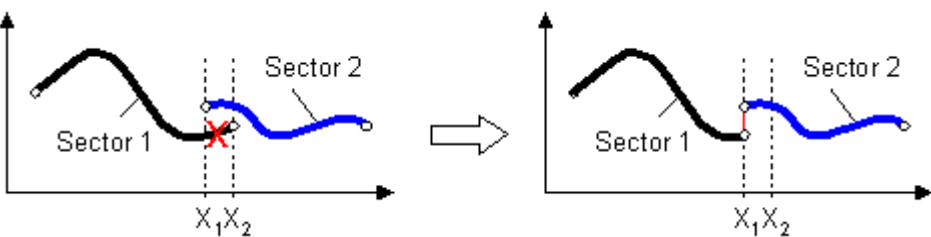
说明

插补具有大量插补点或多项式的凸轮可能要花费一些时间。在此期间，其它工艺对象无法接受任何新命令（甚至包括 **MC_Stop** 命令）。请注意，在此期间，如果未定义终止（例如，“**MC_MoveVelocity**”）就不能取消命令。

如果 **S7T Config** 中命令监视时间设置得过短，则插补可能会导致超时进而导致 T-CPU 进入 **STOP** 状态。

两个凸轮区段之间的插补

如果凸轮包含若干区段，则可以在区段边界产生各种中断：

1.	插补期间将自动填补两个区段之间的间隙。通过联接插补点填补小于特定限制的间隙。 如果凸轮是在 S7T Config 中创建的，则可在专家模式下定义间隙限制（转至 CamEdit >“插补”[Interpolation] 选项卡，然后选中“专家模式”[Expert mode] 复选框）。 如果凸轮盘是使用“MC_CamSectorAdd”工艺功能创建的，则间隙限制将固定为 $1E-04$。在此限制之上，间隙由插补进行填补。在此限制之下，通过连接点填补间隙。 
2.	一个区段的终点和下一个区段的起点具有相同的 X 值和不同的 Y 值，不管是否插补，该中断将一直持续。 
3.	后续区段（扇区 2）的起始位置应用于两个后续区段重叠的位置。上一区段（扇区 1）的重叠终点位置被截短，因此丢失。由于截取操作，在区段 (y(x1)) 的末端随机为扇区 1 分配一个 Y 值，因此在区段限制中，重叠区段特别易于中断。 

MC_CamInterpolate — ErrorID

ErrorID	警告	说明/纠正或避免错误的方法
0000	无警告	-
0027	无法保持插补属性	即使凸轮被插补，也无法保持凸轮的插补属性。 示例： 凸轮仅包含一个在其起点和终点具有不同梯度的多项式。

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i> 、 <i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i> 。验证程序是否包含不必要的（多余的）命令。
8006	存储器不足	集成工艺的存储器不足。 凸轮插补失败。 减少要插入的插补点或多项式，然后再次对凸轮进行插补。

ErrorID	错误消息	说明/纠正或避免错误的方法
8043	参数值非法	涉及数据类型为 REAL 的所有输入参数，或者输入参数 Mode 或 CamMode 。 输入参数 StartPoint 和 EndPoint 的值是相同的（两个输入参数都允许值 0.0 ）。
8044	命令不受工艺对象支持	例如，将命令发送至输出凸轮。
8045	在当前状态下不允许使用该命令	- 该凸轮已被插补 “MC_CamSectorAdd”命令正在将插补点插入凸轮。 此原因仅适用于固件版本为 V3.0.x 的集成工艺。
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是您可能无法通过状态输出对其进行监视。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8051	读/写访问被拒绝。	对相关工艺对象的读/写访问失败。 例如，如果在同步操作中主动跟踪凸轮时启动 MC_CamInterpolate ，则可能会出现此错误。

ErrorID	错误消息	说明/纠正或避免错误的方法
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8062	凸轮插补失败	凸轮正在使用或已被插补。
8066	凸轮盘为空	凸轮盘未包含任何插补点或凸轮区段。
8083	DB 不是工艺 DB	通过输入参数 <i>Axis</i> 指定的 DB 未找到或不是工艺 DB。
8084	工艺 DB 无效	- 在通过输入参数 <i>Axis</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Axis</i> 更改 DB 号。 - 用户在输入参数 <i>Axis</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management) 中删除工艺 DB，然后创建一个新的工艺 DB。
808B	参数值的 REAL 格式无效	数据类型为 REAL 的输入参数的值不符合有效的浮点数格式。 检查输入参数值或背景数据块数据。不能以浮点格式表示格式无效的值。它们以十六进制格式 (DW16# ...) 显示。

6.4.4 FB 438 MC_GetCamPoint — 从凸轮读取点

使用 FB 438“MC_GetCamPoint”从凸轮读取点

用途

- “MC_GetCamPoint”工艺功能用于确定跟随轴相对于引导轴的位置，以及现有凸轮中凸轮点的一阶导数和二阶导数。
- “MC_GetCamPoint”工艺功能用于确定引导轴相对于现有凸轮中的跟随轴的位置。
 - 由于定义凸轮时，可为不同的主轴位置输入相同的从轴位置，因此必须更精确地定义主轴位置。可以通过输入参数 *ApproachPosition* 执行该操作。在此输入参数中输入假定引导轴位置的近似值。
- 如果在 S7T Config 中将凸轮组态为使用 CamEdit 或 CamTool，则计算位置时将包括定义的缩放和移位。运行系统中动态创建的凸轮始终不会进行缩放和移位。确定位置后，MC_CamIn 工艺功能的输入参数所导致的缩放和移位操作都会被忽略。

适用对象

- 凸轮

先决条件

- 凸轮被无错误插补（工艺数据块 2 的状态 = 已插补）。

超驰命令

MC_GetCamPoint 命令不能由任何其它命令取消。

MC_GetCamPoint 命令不取消任何其它命令。

输入参数

参数	数据类型	初始值	说明
<i>CamTable</i>	INT	0	凸轮盘工艺 DB 的编号
<i>Execute</i>	BOOL	FALSE	在上升沿处启动命令
<i>Mode</i>	INT	0	值 = 0: 跟随轴的位置取决于与引导轴的相对位置
			值 = 1: 引导轴的位置取决于与跟随轴的相对位置
			值 = 2: 凸轮点的一阶导数取决于与引导轴的相对位置
			值 = 3: 凸轮点的二阶导数取决于与引导轴的相对位置
			这两种模式都将考虑在 S7T Config 中组态的任何缩放和偏移。
<i>Position</i>	REAL	0.0	要确定的相应值的引导轴或跟随轴位置： • <i>Mode</i> = 0, 2, 3: 引导轴的位置 • <i>Mode</i> = 1: 跟随轴的位置 参数值的有效范围是从域或凸轮的值范围导出的。
<i>ApproachPosition</i>	REAL	0.0	假定的引导轴位置 凸轮的域中的近似值，用于确定引导轴位置（仅 <i>Mode</i> = 1 时有效）。 • 该功能仅确定距离 <i>ApproachPosition</i> 最近的引导轴的位置。 • 当两个引导轴所在位置与 <i>ApproachPosition</i> 的距离相同时，将确定位置值较低的引导轴位置。
<i>DoneFlag</i>	INT	0	在 MCDevice DB 中生成 DoneFlag

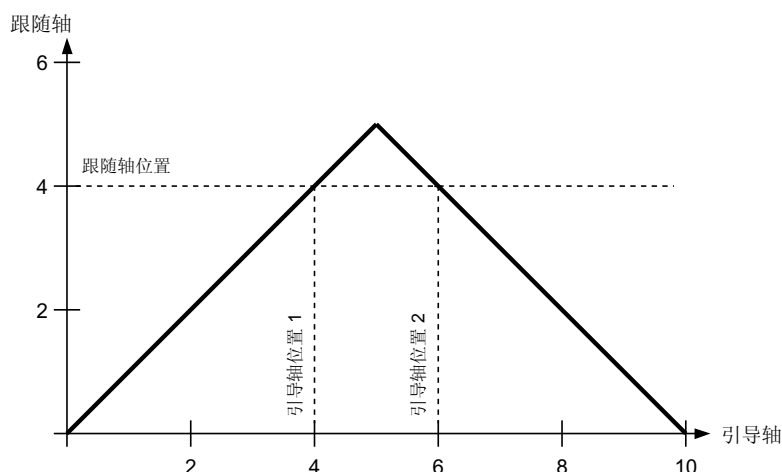
输出参数（状态输出）

参数	数据类型	初始值	说明
<i>Done</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令已成功完成
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令
<i>CommandAborted</i>	BOOL	<i>FALSE</i>	<i>TRUE</i>: 命令由其它命令或因其执行过程中的错误而取消。 如果工艺 DB 的 <i>ErrorStatus</i> 中未显示错误, 则该命令由后续命令取消。 如果工艺 DB 的变量 <i>ErrorStatus</i> 中指示有错误, 则命令执行期间影响该工艺对象的错误将导致该命令中止。
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i>: 命令启动时有错误。未执行该命令。有关原因的信息, 请参考 <i>ErrorID</i>。
			<i>FALSE</i>: 命令启动时没有错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。
<i>Value</i>	REAL	0.0	读取位置值 <i>Mode</i> = 0: 跟随轴的位置 <i>Mode</i> = 1: 引导轴的位置 <i>Mode</i> = 2: 1. 凸轮点的一阶导数 <i>Mode</i> = 3: 凸轮点的二阶导数

模式 1

一个跟随轴位置可能对应于任意数量的引导轴位置。通过输入参数 *ApproachPosition* 可以指定假定的引导轴位置。

示例：



要确定跟随轴位置为 4 时相应的引导轴位置。在以上示例中，跟随轴位置有两个对应的引导轴位置。通过输入参数 *ApproachPosition* 对要确定的引导轴位置进行定义：

- *ApproachPosition* < 5
确定引导轴位置 1 的位置值。
- *ApproachPosition* > 5
确定引导轴位置 2 的位置值。
- *ApproachPosition* = 5
ApproachPosition 的值正好在引导轴位置 1 和引导轴位置 2 之间。根据定义，在这种情况下，应确定位置值较小的引导轴位置。为避免舍入结果，应使用 *ApproachPosition* 的定值。

ApproachPosition 的值也可能在凸轮定义范围之外。在本例中，*ApproachPosition* = -10 时，应确定引导轴位置 1 的值。

模式 2 和 3

如果凸轮盘是在 T-CPU 中生成和插补的，则无法在 S7T Config 中检查插补结果。以下公式用于求解速度和加速度：

跟随轴的速度

$$v_{跟随轴} = v_{引导轴} * (凸轮点的一阶导数)$$

跟随轴的加速度

$$a_{跟随轴} = a_{引导轴} * (凸轮点的一阶导数) + v_{引导轴}^2 * (凸轮点的二阶导数)$$

例如，想要检查凸轮连续性时，可以使用“MC_GetCamPoint”工艺功能。按适当的间隔在临界点读取凸轮点的导数，并应用上述公式。

MC_GetCamPoint — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： • 活动命令的数量已超出限制。 • 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。

ErrorID	错误消息	说明/纠正或避免错误的方法
8043	参数值非法	涉及数据类型为 REAL 的所有输入参数，或者输入参数 <i>Mode</i> 或 <i>DoneFlag</i> 。
8044	命令不受工艺对象支持	例如，命令要求有一个轴
8045	在当前状态下不允许使用该命令。	该凸轮未被插补
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是您可能无法通过状态输出对其进行监视。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 无法通过工艺功能的输出参数监视最初启动的命令。但是，该命令在集成工艺中可能仍处于活动状态。 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8061	凸轮点超出域/范围	“MC_GetCamPoint”中定义的引导/跟随轴的位置不存在于凸轮的域/范围中。 在“MC_GetCamPoint”中设置有效位置。

ErrorID	错误消息	说明/纠正或避免错误的方法
8083	DB 不是工艺 DB	通过输入参数 <i>CamTable</i> 指定的 DB 未找到或不是功能 DB。
8084	工艺 DB 无效	- 在通过输入参数 <i>CamTable</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>CamTable</i> 更改 DB 号。 - 用户在输入参数 <i>CamTable</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management) 中删除工艺 DB，然后创建一个新的工艺 DB。
808B	参数值的 REAL 格式无效	数据类型为 REAL 的输入参数的值不符合有效的浮点数格式。 检查输入参数值或背景数据块数据。不能以浮点格式表示格式无效的值。它们以十六进制格式 (DW16# ...) 显示。

6.5 工艺功能 — 输出凸轮，凸轮轨迹

6.5.1 FB 430 MC_CamSwitch — 基于位置的凸轮

使用 FB 430“MC_CamSwitch”的基于位置的凸轮或单方向的输出凸轮

用途

- “MC_CamSwitch”工艺功能用于启用/禁用基于位置的凸轮或单方向的输出凸轮以及定义它们的切换性能。
- 在凸轮组态中，您可以定义切换位置基于设定值还是实际值。
- 切换点可以超前也可以延迟（比率时间和取消激活时间）。在 S7T Config 中设置的时间单位适用于轴。
- 输出凸轮的当前切换状态被保存到工艺 DB。

适用对象

- 基于位置的凸轮
- 单方向的输出凸轮

要求

- 必须在 S7T Config 中组态了输出凸轮工艺对象，而且必须已将当前组态装载到 T-CPU 中。
- 输出凸轮的工艺 DB 必须已生成并且已装载到控制系统。
- 仅当轴处于位置控制模式时，组态为设定值输出凸轮的输出凸轮才会激活。

超驰命令

MC_CamSwitch 命令仅可被其它输出到相同 TO 的 MC_CamSwitch 命令取消。
MC_CamSwitch 命令无法取消其它命令。

输入参数

参数	数据类型	初始值	说明
<i>CamSwitch</i>	INT	0	工艺 DB 的编号
<i>Execute</i>	BOOL	FALSE	上升沿的操作状态转换
<i>OnPosition</i>	REAL	0.0	起始位置
<i>OffPosition</i>	REAL	0.0	结束位置（适用于基于位置的凸轮）
<i>Hysteresis</i>	REAL	0.0	滞后
<i>Delay</i>	REAL	0.0	输出凸轮切换点的基于时间的偏移
			值 < 0: 提前激活
			值 > 0: 延缓激活
			将应用在 S7T Config 中为输出凸轮设置的时间单位。 要在 S7T Config 中验证或设置时间单位： 在浏览器中选择输出凸轮 TO (Output Cam TO)，然后 选择快捷菜单“专家 > 组态单位”(Expert > Configure Units)。
<i>Mode</i>	INT	2	操作模式：
			值 = 1: 取消激活输出凸轮 输出凸轮永久取消激活，工艺 DB: <i>state</i> = 0 (OFF)
			值 = 2: 输出凸轮已启用（输出未取反） 输出凸轮输出在激活范围内激活
			值 = 3: 输出凸轮已启用（输出已取反） 输出凸轮输出在激活范围外激活
			值 = 4: 输出凸轮被永久激活 （输出凸轮输出永久激活，工艺 DB: <i>state</i> = 1 (ON)）

参数	数据类型	初始值	说明
<i>Direction</i>	INT	1	输出凸轮的有效方向：
			值 = 1: 正有效方向
			值 = 2: 正有效方向和负有效方向 (即，有效方向不相关)
			值 = 3: 负有效方向
			值 = 4: 使用上一次激活的有效输出凸轮方向
<i>DoneFlagPos</i>	INT	0	激活输出凸轮时，在 MCDevice-DB 中生成 DoneFlag。
<i>DoneFlagNeg</i>	INT	0	取消激活输出凸轮时，在 MCDevice-DB 中生成 DoneFlag。

说明**驱动实际值输出凸轮失败时进行响应：**

在 S7T Config 中组态为实际值输出凸轮的输出凸轮需要有效的编码器值。因此，当驱动失败时，将禁用关联的输出凸轮。该命令由输出凸轮工艺 DB 中的 *ErrorID = 8021* 取消。

驱动设定值输出凸轮失败时进行响应：

在 S7T Config 中组态为设定值输出凸轮的输出凸轮需要关联的编码器值。驱动失败后输出凸轮保持活动状态。在确认错误并启用驱动后，输出凸轮将在组态的位置重新开始切换操作。

轴重启时响应

当与输出凸轮关联的轴执行重启时，取消激活该输出凸轮（“MC_Reset”、*Restart = TRUE*）。不管在 S7T Config 中将输出凸轮组态为设定值还是实际值，这始终适用。

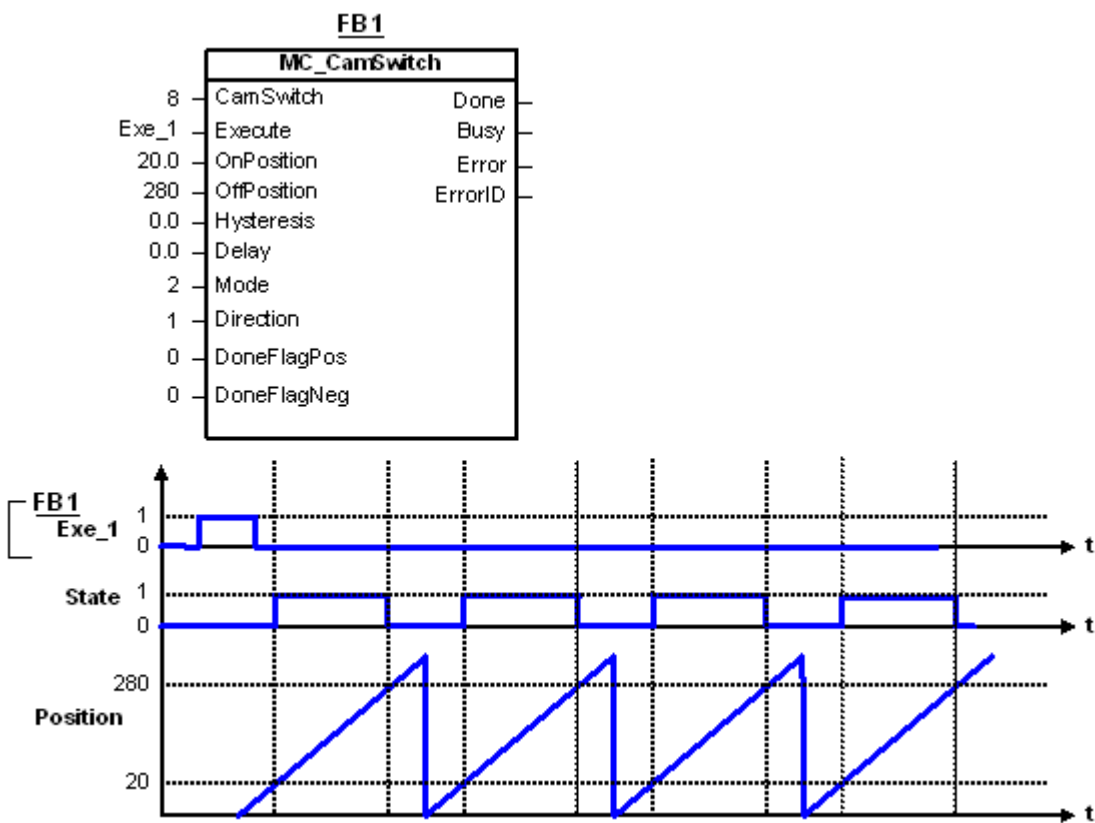
输出参数（状态输出）

参数	数据类型	初始值	说明
<i>Done</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> ：更改已接受
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> ：正在执行命令
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> ：命令启动时有错误。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
			<i>FALSE</i> ：命令启动时没有错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。

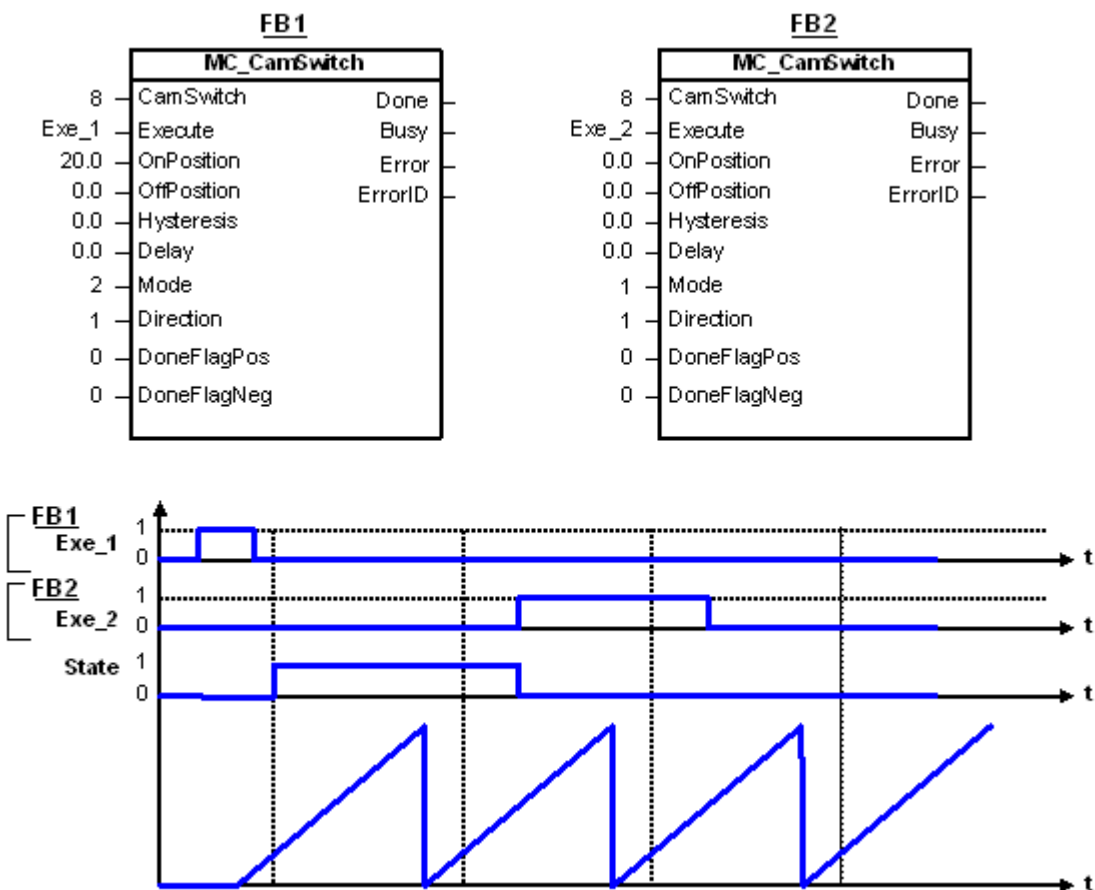
输入参数 *OffPosition*

没有为单方向的输出凸轮定义结束位置。 *OffPosition* 输入没有任何影响。 可以通过用户程序重设（例如，通过重复调用“MC_CamSwitch”工艺功能）单方向的输出凸轮。

MC_CamSwitch — 示例 — “基于位置的凸轮”



MC_CamSwitch — 示例 — “单方向的输出凸轮”



调用 FB1:

通过设置输入参数 *Mode* = 2 选择“启用输出凸轮”操作模式。通过输入参数 *Execute* 的上升沿启用单方向的输出凸轮。

运动的方向对应于“正有效方向”（输入参数 *Direction* = 1）。通过起始位置（输入参数 *OnPosition* = 20）时，单方向的输出凸轮将被激活并保持此状态。

调用 FB 2:

通过设置输入参数 *Mode* = 1 选择“禁用输出凸轮”操作模式。通过输入参数 *Execute* 的上升沿禁用单方向的输出凸轮。

MC_CamSwitch — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8043	参数值非法	涉及数据类型为 <i>REAL</i> 的所有输入参数，或者输入参数 <i>Mode</i>、<i>Direction</i>、<i>DoneFlagPos</i> 或 <i>DoneFlagNeg</i>。 适用于输入参数 <i>Hysteresis</i>： - 无限轴 — 值 > 操作范围的 ¼ 通过软件限位开关的位置定义操作范围。 - 模数轴 — 值 > 模数长度的 ¼
8044	命令不受工艺对象支持	- 例如，命令被发送到测量输入。 - 例如，命令被输出到组态为基于时间的凸轮的凸轮。
8045	在当前状态下不允许使用该命令	可能的原因： - 输出凸轮在硬件终端（例如，ET 200M）输出失败 - 工艺对象在 <i>S7T Config</i> 中或通过使用“MC_ActivateTO”工艺功能被取消激活。

ErrorID	错误消息	说明/纠正或避免错误的方法
804A	缺少所需的对象连接	可能的原因： 输出凸轮的轴在 S7T Config 中或通过使用“MC_ActivateTO”工艺功能被取消激活。
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是您可能无法通过状态输出对其进行监视。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8083	DB 不是工艺 DB	通过输入参数 <i>Axis</i> 指定的 DB 未找到或不是工艺 DB。

ErrorID	错误消息	说明/纠正或避免错误的方法
8084	工艺 DB 无效	- 在通过输入参数 <i>Axis</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Axis</i> 更改 DB 号。 - 用户在输入参数 <i>Axis</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management)中删除工艺 DB，然后创建一个新的工艺 DB。
808B	参数值的 REAL 格式无效	数据类型为 REAL 的输入参数的值不符合有效的浮点数格式。 检查输入参数值或背景数据块数据。不能以浮点格式表示格式无效的值。它们以十六进制格式 (DW16# ...) 显示。

6.5.2 FB 431 MC_CamSwitchTime — 基于时间的凸轮

使用 FB 431“MC_CamSwitchTime”的基于时间的凸轮

用途

- “MC_CamSwitchTime”工艺功能用于激活/取消激活基于时间的凸轮以及定义它们的切换性能。到达起始位置时，激活基于时间的凸轮；超出脉冲周期时，复位基于时间的凸轮。
- 在组态中，您可以定义切换位置基于设定值还是实际值。
- 所有时间定义都参考在 S7T Config 中组态的时基。
- 输出凸轮的当前激活状态被保存到工艺 DB。

适用对象

- 基于时间的凸轮

要求

- 必须在 S7T Config 中组态了输出凸轮工艺对象，而且必须已将当前组态装载到 T-CPU 中。
- 输出凸轮的工艺 DB 必须已生成并且已装载到控制系统。
- 仅当轴处于位置控制模式时，组态为设定值输出凸轮的输出凸轮才会激活。

超驰命令

MC_CamSwitchTime 命令仅可被其它输出到相同 TO 的 MC_CamSwitchTime 命令取消。MC_CamSwitchTime 命令无法取消其它命令。

输入参数

参数	数据类型	初始值	说明
<i>CamSwitch</i>	INT	0	工艺 DB 的编号
<i>Execute</i>	BOOL	FALSE	上升沿的操作状态转换
<i>OnPosition</i>	REAL	0.0	起始位置
<i>Duration</i>	REAL	0.0	脉冲宽度（根据相应轴中设置的“时间”单位的物理单位）
<i>Hysteresis</i>	REAL	0.0	滞后

参数	数据类型	初始值	说明
<i>Delay</i>	REAL	0.0	输出凸轮切换点的基于时间的偏移
			值 = < 0: 提前激活
			值 = > 0: 延缓激活
			将应用在 S7T Config 中为输出凸轮设置的时间单位。 要在 S7T Config 中验证或设置时间单位： 在浏览器中选择输出凸轮 TO (Output Cam TO)，然后 选择快捷菜单“专家 > 组态单位”。
<i>Mode</i>	INT	2	操作模式：
			值 = 1: 输出凸轮被封锁
			值 = 2: 输出凸轮已启用（输出未取反）
			值 = 3: 输出凸轮已启用（输出已取反）
			值 = 4: 输出凸轮被永久激活
<i>Direction</i>	INT	1	输出凸轮的有效方向：
			值 = 1: 正有效方向
			值 = 2: 正有效方向和负有效方向 （表明与有效方向无关）
			值 = 3: 负有效方向
			值 = 4: 使用上一次激活的有效输出凸轮方向
<i>DoneFlagPos</i>	INT	0	激活输出凸轮时，在 MCDevice DB 中生成 DoneFlag。
<i>DoneFlagNeg</i>	INT	0	取消激活输出凸轮时，在 MCDevice DB 中生成 DoneFlag。

说明**驱动实际值输出凸轮失败时进行响应：**

在 S7T Config 中组态为实际值输出凸轮的输出凸轮需要有效的编码器值。因此，当驱动失败时，将禁用关联的输出凸轮。该命令由输出凸轮工艺 DB 中的 *ErrorID = 8021* 取消。

驱动设定值输出凸轮失败时进行响应：

在 S7T Config 中组态为设定值输出凸轮的输出凸轮需要关联的编码器值。驱动失败后输出凸轮保持活动状态。在确认错误并启用驱动后，输出凸轮将在组态的位置重新开始切换操作。

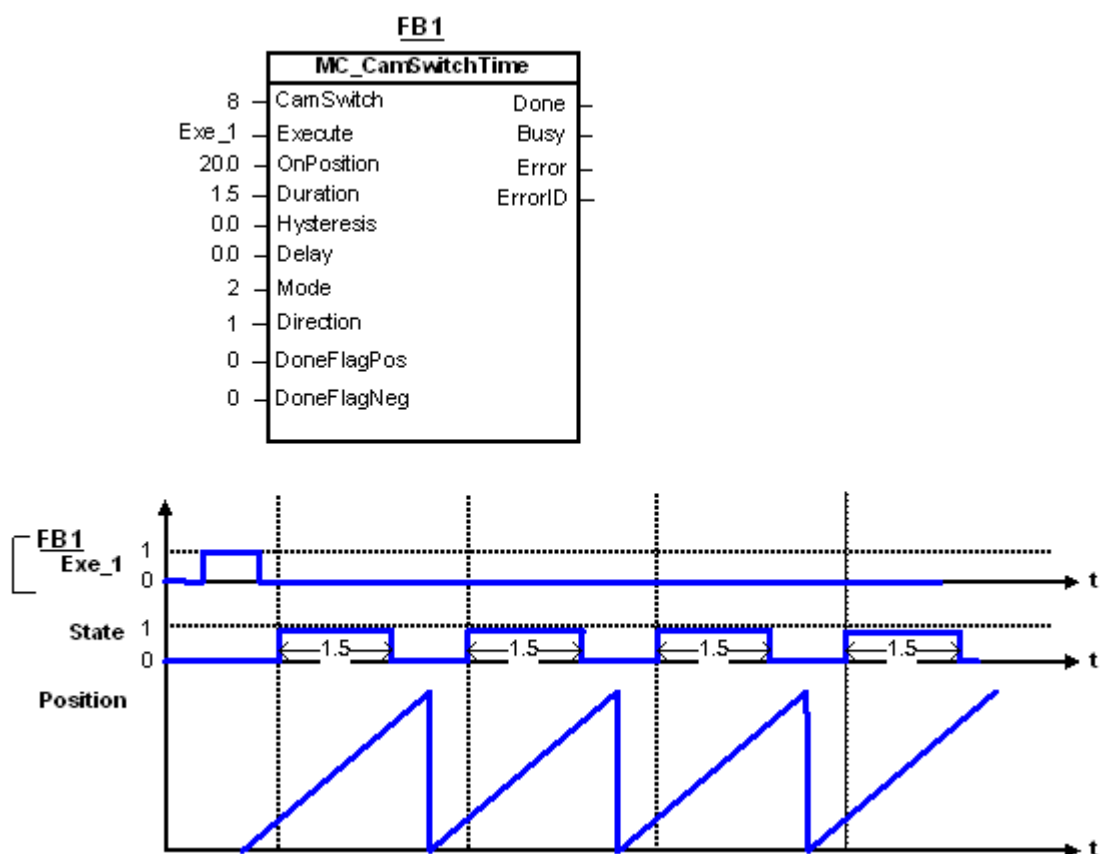
轴重启时响应

当与输出凸轮关联的轴执行重启时，取消激活该输出凸轮（“MC_Reset”、*Restart = TRUE*）。不管在 S7T Config 中将输出凸轮组态为设定值还是实际值，这始终适用。

输出参数（状态输出）

参数	数据类型	初始值	说明
<i>Done</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> ：更改已接受
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> ：正在执行命令
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> ：命令启动时 有 错误。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
			<i>FALSE</i> ：命令启动时 没有 错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。

MC_CamSwitchTime — 示例



MC_CamSwitchTime — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8043	参数值非法	涉及数据类型为 <i>REAL</i> 的所有输入参数，或者输入参数 <i>Mode</i>、<i>Direction</i>、<i>DoneFlagPos</i> 或 <i>DoneFlagNeg</i>。 适用于输入参数 <i>Hysteresis</i>： - 无限轴 — 值 > 操作范围的 ¼ 通过软件限位开关的位置定义操作范围。 - 模数轴 — 值 > 模数长度的 ¼
8044	命令不受工艺对象支持	- 例如，命令被发送到测量输入。 - 例如，命令被发送到基于位置的凸轮或切换凸轮。
8045	在当前状态下不允许使用该命令	可能的原因： - 输出凸轮在硬件终端（例如，ET 200M）输出失败 - 工艺对象在 <i>S7T Config</i> 中或通过使用“MC_ActivateTO”工艺功能被取消激活。

ErrorID	错误消息	说明/纠正或避免错误的方法
804A	缺少所需的对象连接	可能的原因： 输出凸轮的轴在 S7T Config 中或通过使用“MC_ActivateTO”工艺功能被取消激活。
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是可能无法通过状态输出对其进行监视。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8083	DB 不是工艺 DB	通过输入参数 <i>Axis</i> 指定的 DB 未找到或不是工艺 DB。

ErrorID	错误消息	说明/纠正或避免错误的方法
8084	工艺 DB 无效	- 在通过输入参数 <i>Axis</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Axis</i> 更改 DB 号。 - 用户在输入参数 <i>Axis</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management)中删除工艺 DB，然后创建一个新的工艺 DB。
808B	参数值的 REAL 格式无效	数据类型为 REAL 的输入参数的值不符合有效的浮点数格式。 检查输入参数值或背景数据块数据。不能以浮点格式表示格式无效的值。它们以十六进制格式 (DW16# ...) 显示。

6.5.3 FB 461 MC_CamTrack — 凸轮轨迹

使用 FB 461“MC_CamTrack”激活凸轮轨迹

适用于固件版本为 V4.1.x 的集成工艺

用途

- 使用工艺功能“MC_CamTrack”可以定义凸轮轨迹的当前操作模式。
- 可根据软件凸轮中的变量 *SingleCamState[0 ... 31]* 以软件凸轮的形式评估凸轮轨迹专用输出凸轮的切换状态。
- 根据 S7T Config 中的组态，可以为所有激活的专用输出凸轮切换公共硬件输出凸轮输出。

适用对象

- 凸轮轨迹

先决条件

- 必须在 S7T Config 中组态了凸轮轨迹工艺对象，而且必须已将当前组态装载到 T-CPU 中。
- 凸轮轨迹的工艺 DB 必须已生成并且已装载到控制系统。
- 仅当轴处于位置控制模式时，组态为设定值输出凸轮的凸轮轨迹才会激活。

超驰命令

命令仅能由另一个输出到同一工艺对象的命令取消。它不取消任何其它命令。

6.5 工艺功能 — 输出凸轮，凸轮轨迹

输入参数

参数	数据类型	初始值	说明
<i>CamTrack</i>	INT	0	工艺 DB 的编号
<i>Execute</i>	BOOL	FALSE	在上升沿处执行命令
<i>CyclicMode</i>	BOOL	TRUE	在轴参考位置创建凸轮轨迹。
			TRUE 循环创建凸轮轨迹。 在轴参考位置 (输入参数 <i>ReferencePosition</i>) 创建凸轮轨迹。创建的凸轮轨迹长度对应于轨迹长度 (输入参数 <i>CamTrackLength</i>)。按照凸轮轨迹长度以循环方式在轴参考位置重复创建凸轮轨迹。凸轮轨迹在所有轴范围内为活动状态。
			FALSE 非循环创建凸轮轨迹。 在轴参考位置 (输入参数 <i>ReferencePosition</i>) 创建凸轮轨迹。创建的凸轮轨迹长度对应于轨迹长度 (输入参数 <i>CamTrackLength</i>)。
<i>Mode</i>	INT	2	凸轮轨迹的操作模式
			值 = 1 禁用凸轮轨迹
			值 = 2 启用凸轮轨迹
			值 = 3 值非法
			值 = 4 永久激活专用输出凸轮和输出凸轮输出

参数	数据类型	初始值	说明
<i>CommandMode</i>	INT	1	输入参数仅在 <i>Mode</i> = 1 和 2 时生效。
			值 = 1 • <i>Mode</i> = 1 (锁定) 立即取消激活/锁定凸轮轨迹 • <i>Mode</i> = 2 (释放) 如果轴位置在凸轮轨迹长度范围内，则立即激活凸轮轨迹。 如果轴位置在凸轮轨迹长度范围之外，则第一次沿正方向或负方向进入凸轮轨迹时凸轮轨迹将变为活动状态。 – 非循环创建凸轮轨迹： 凸轮轨迹在轴参考位置创建一次。 对于模数轴，可根据邻近模数周期中的轴位置和驱动方向进行创建和激活。 凸轮轨迹的取消激活依赖于 S7T Config 中的组态。 – 循环创建凸轮轨迹： 凸轮轨迹将保持活动状态，直到凸轮轨迹被锁定。

参数	数据类型	初始值	说明	
			值 = 2	• Mode = 1（锁定） – 非循环创建凸轮轨迹： 当退出凸轮轨迹后，凸轮轨迹将被取消激活/锁定 – 循环创建凸轮轨迹： 当切换到下一个凸轮轨迹周期时，凸轮轨迹将被取消激活/锁定 • Mode = 2（释放） 如果轴位置在凸轮轨迹长度范围内，则不激活凸轮轨迹。 只有当再次进入凸轮轨迹时，才会激活凸轮轨迹。 如果希望在操作期间更改凸轮轨迹数据，请使用 CommandMode。在这种情况下，可在激活新的凸轮轨迹之前处理活动的凸轮轨迹。 – 非循环创建凸轮轨迹： 凸轮轨迹在轴参考位置创建一次。 对于模数轴，可根据邻近模数周期中的轴位置和驱动方向进行创建和激活。 凸轮轨迹的取消激活依赖于 S7T Config 中的组态。 – 循环创建凸轮轨迹： 凸轮轨迹将保持活动状态，直到凸轮轨迹被锁定。
			值 = 3	活动的凸轮轨迹命令将继续执行，直到凸轮轨迹输出或所有专用输出凸轮被取消激活。之后，将通过新命令激活/取消激活凸轮轨迹。

参数	数据类型	初始值	说明
<i>CamTrackLength</i>	REAL	0.0	凸轮轨迹长度 通过凸轮轨迹长度来定义要激活的凸轮轨迹的长度。 如果要循环使用凸轮轨迹，则由凸轮轨迹长度决定凸轮轨迹周期。
			值 < 0: 不允许
			值 = 0: - 对于模数轴，轨迹长度被设置为模数长度 - 对于非模数轴，不允许使用该值
			值 > 0: 凸轮轨迹的轨迹长度
<i>ReferencePosition</i>	REAL	0.0	轴参考位置 凸轮轨迹从该轴位置创建。
<i>Hysteresis</i>	REAL	0.0	凸轮轨迹专用输出凸轮的滞后 切换位置对应的范围，在该范围内，尽管有最小位置波动，但输出凸轮不会更改其切换操作。
<i>ActivationDelay</i>	REAL	0.0	基于时间的激活偏移 (以便补偿数字输出和切换元素的切换时间)
			值 < 0: 提前激活
			值 = 0: 激活无延迟
			值 > 0: 延迟激活
<i>DeactivationDelay</i>	REAL	0.0	基于时间的取消激活偏移 (以便补偿数字输出和切换元素的切换时间)
			值 < 0: 提前取消激活
			值 = 0: 取消激活无延迟
			值 > 0: 延迟取消激活

注意

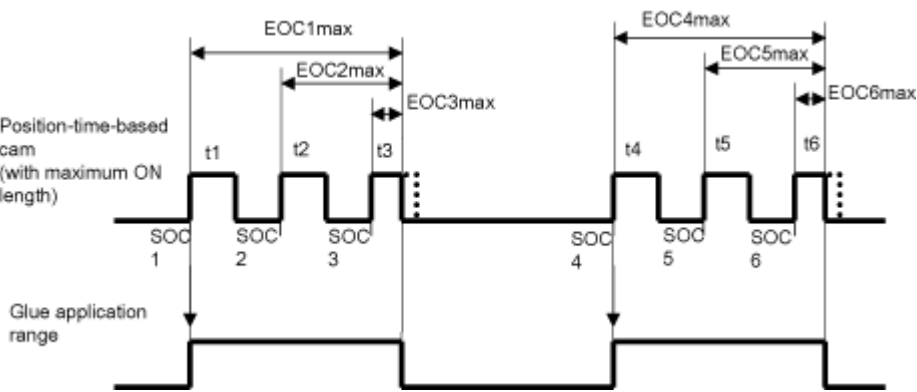
选择凸轮轨迹长度时，请确保所有有效的专用输出凸轮全部位于凸轮轨迹长度范围之内。如有必要，可延长凸轮轨迹长度或取消相应的专用输出凸轮的有效性。
部分或完全位于凸轮轨迹长度之外的有效专用凸轮会被投影到凸轮轨迹长度的范围内，并会引起意外的切换状态。

输出参数（状态输出）

参数	数据类型	初始值	说明	
<i>Done</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 已执行命令	
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令	
<i>CommandAborted</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令由其它命令或因其执行过程中的错误而取消。 如果工艺 DB 的 <i>ErrorStatus</i> 中未显示错误，则该命令由后续命令取消。 如果工艺 DB 的变量 <i>ErrorStatus</i> 中指示有错误，则命令执行期间影响该工艺对象的错误将导致该命令中止。	
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> :	命令启动时有错误。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
			<i>FALSE</i> :	命令启动时没有错误。
<i>ErrorID</i>	BOOL	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。	

用于控制粘胶剂涂抹的凸轮轨迹示例

在以下示例中，使用具有三个输出凸轮的凸轮轨迹来控制在工件上涂抹粘胶剂。 不能将任何粘胶剂涂抹到预定义区域之外。



粘胶剂的用量通过基于时间的凸轮 *t1* 到 *t3* 以及 *t4* 到 *t6* 进行控制。粘胶剂涂抹的起点分别取决于凸轮起点 *SOC1* 或 *SOC4*。

为了确保当轴快速移动时不会将粘胶剂涂抹到工件长度外部，应限制最大开启长度。

在本例中，开启长度 *EOC1max*、*EOC2max* 和 *EOC3max* 可确保粘胶剂不会涂抹到粘胶剂涂抹范围之外。这同样适用于第二个粘胶剂涂抹范围，而与轴速度无关。

MC_CamSwitch — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8043	参数值非法	涉及数据类型为 <i>REAL</i> 的所有输入参数，或者输入参数 <i>Mode</i> 或 <i>CommandMode</i>。 适用于输入参数 <i>Hysteresis</i>： - 无限轴 — 值 > 操作范围的 $\frac{1}{4}$ 通过软件限位开关的位置定义操作范围。 - 模数轴 — 值 > 模数长度的 $\frac{1}{4}$
8044	命令不受工艺对象支持	例如，将命令应用到轴工艺对象。
8045	在当前状态下不允许使用该命令	示例： - 输出凸轮在 ET 200M 硬件终端输出失败 - 分配给凸轮轨迹的轴处于速度控制模式。
804A	缺少所需的对象连接	可能的原因： 凸轮轨迹的轴在 S7T Config 中或通过使用 "MC_ActivateTO" 工艺功能被取消激活。

ErrorID	错误消息	说明/纠正或避免错误的方法
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是您可能无法通过状态输出对其进行监视。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8083	DB 不是工艺 DB	通过输入参数 <i>CamTrack</i> 指定的 DB 未找到或不是工艺 DB。

ErrorID	错误消息	说明/纠正或避免错误的方法
8084	工艺 DB 无效	- 在通过输入参数 <i>CamTrack</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>CamTrack</i> 更改 DB 号。 - 用户在输入参数 <i>CamTrack</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management)中删除工艺 DB，然后创建一个新的工艺 DB。
808B	参数值的 REAL 格式无效	数据类型为 REAL 的输入参数的值不符合有效的浮点数格式。 检查输入参数值或背景数据块数据。不能以浮点格式表示格式无效的值。它们以十六进制格式 (DW16# ...) 显示。

6.5.4 FB 462 MC_ReadCamTrackData — 读出凸轮轨迹

使用 FB 462“MC_ReadCamTrackData”读出凸轮轨迹

适用于固件版本为 V4.1.x 的集成工艺

用途

- 该工艺功能用于从集成工艺中读出凸轮轨迹的专用输出凸轮数据。
- Mode* 输入参数用于确定是读出凸轮轨迹专用输出凸轮数据的“实际值”还是当前有效的专用凸轮数据。

适用对象

- 凸轮轨迹

超驰命令

命令仅能由另一个在同一工艺对象上输出的命令取消。命令不会取消其它任何命令。

输入参数

参数	数据类型	初始值	说明
<i>CamTrack</i>	INT	0	工艺 DB 的编号
<i>Execute</i>	BOOL	FALSE	在上升沿处执行命令
<i>Mode</i>	INT	0	确定要读取的范围。
			值 = 0: 读取凸轮轨迹专用输出凸轮数据的“实际值”。 这些值可在 S7T Config 的在线模式下找到：凸轮轨迹的专家列表 >“系统变量”(System variables) 选项卡 >“实际值”(Actual value) 列 > 系统变量 <i>userdefault.singlecamsettings.cam.cam[x]</i>。
			值 = 1: 读取凸轮轨迹实际有效的专用输出凸轮数据。 这些值可在 S7T Config 的在线模式下找到：凸轮轨迹的专家列表 >“系统变量”(System variables) 选项卡 >“实际值”(Actual value) 列 > 系统变量 <i>effectivedata.singlecamsettings.cam.cam[x]</i>。
<i>StartCam</i>	INT	0	专用输出凸轮的编号（0 到 31），从该编号开始读取专用凸轮数据。读出专用输出凸轮数据时，将一直读取到专用输出凸轮 31。
<i>Data</i>	ANY	-	读取数据的目的地。 要读取的数据的最大长度取决于数据类型和 ANY 指针的重复因子。

说明

可以通过使用“S7-Tech”库中的 UDT 100 来保存您的编辑工作。UDT 100 包含用于读取凸轮轨迹数据的数据结构。

输出参数（状态输出）

参数	数据类型	初始值	说明
<i>Done</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 已执行命令
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令启动时 有错误 。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
			<i>FALSE</i> : 命令启动时 没有错误 。
<i>ErrorID</i>	BOOL	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。

MC_ReadCamTrackData — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。

ErrorID	错误消息	说明/纠正或避免错误的方法
800B	无法保留传输缓冲区	命令的传输缓冲区不足。可能的补救措施： - 传输缓冲区已被其它的活动工艺功能占用。防止同时执行以下工艺功能： “MC_ReadCamTrackData” “MC_WriteCamTrackData” “MC_CamSectorAdd” “MC_ReadPeriphery” “MC_WritePeriphery” “MC_ReadRecord” “MC_WriteRecord” “MC_ReadDriveParameter” “MC_WriteDriveParameter” - 减少凸轮数据量。
8043	参数值非法	涉及数据类型为 REAL 的所有输入参数，或者输入参数 <i>StartCam</i>。 适用于输入参数 <i>Hysteresis</i>： - 无限轴 — 值 > 操作范围的 ¼ 通过软件限位开关的位置定义操作范围。 - 模数轴 — 值 > 模数长度的 ¼
8044	命令不受工艺对象支持	例如，将命令应用到轴工艺对象
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是您可能无法通过状态输出对其进行监视。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。

ErrorID	错误消息	说明/纠正或避免错误的方法
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8083	DB 不是工艺 DB	通过输入参数 <i>CamTrack</i> 指定的 DB 未找到或不是工艺 DB。
8084	工艺 DB 无效	- 在通过输入参数 <i>CamTrack</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>CamTrack</i> 更改 DB 号。 - 用户在输入参数 <i>CamTrack</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management) 中删除工艺 DB，然后创建一个新的工艺 DB。
8086	参数不可用	当前无法执行读取/写入命令；重复该命令。
8088	DB 无效（ANY 指针）	ANY 指针指向无效的数据区域。 - 找不到数据块 - 找不到指定的起始地址 - 数据区域过短

6.5 工艺功能 — 输出凸轮，凸轮轨迹

ErrorID	错误消息	说明/纠正或避免错误的方法
8089	数据记录长度无效（ANY 指针）	指定的数据长度无效。可能的原因： • 长度超过 576 个字节。 • 指定的长度不能被 18 整除。
808C	指针格式无效或不受支持	ANY 指针包含无效的定义。接受的区域： I、O、M、DB、DI 接受的数据类型： BYTE、CHAR、WORD、INT、DWORD、DINT、REAL

6.5.5 FB 463 MC_WriteCamTrackData — 写入凸轮轨迹

使用 FB463“MC_WriteCamTrackData”写入凸轮轨迹

用途

- 该工艺功能用于将凸轮轨迹的专用输出凸轮数据写入集成工艺的“实际值”。更改的专用输出凸轮数据可在 S7T Config 的在线模式下找到：凸轮轨迹的专家列表 >“系统变量”(System variables) 选项卡 >“实际值”(Actual value) 列 > 系统变量 *userdefault.singlecamsettings.cam.cam[x]*。

适用对象

- 凸轮轨迹

超驰命令

命令仅能由另一个在同一工艺对象上输出的命令取消。命令不会取消其它任何命令。

输入参数

参数	数据类型	初始值	说明
<i>CamTrack</i>	INT	0	工艺 DB 的编号
<i>Execute</i>	BOOL	FALSE	在上升沿处执行命令
<i>StartCam</i>	INT	0	专用输出凸轮的编号（0 到 31），从该编号开始写入专用凸轮数据。写入专用输出凸轮数据时，将一直写入到专用输出凸轮 31。
<i>Data</i>	ANY	-	要写入数据的源区域。 要写入数据的长度取决于数据类型和 ANY 指针中的重复因子。

说明

可以通过使用“S7-Tech”库中的 UDT 100 来保存您的编辑工作。UDT 100 包含用于写入凸轮轨迹数据的数据结构。

输出参数（状态输出）

参数	数据类型	初始值	说明
<i>Done</i>	BOOL	FALSE	TRUE: 已执行命令
<i>Busy</i>	BOOL	FALSE	TRUE: 正在执行命令
<i>Error</i>	BOOL	FALSE	TRUE: 命令启动时有错误。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
			FALSE: 命令启动时没有错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。

MC_WriteCamTrackData — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
800B	无法保留传输缓冲区	命令的传输缓冲区不足。可能的补救措施： - 传输缓冲区已被其它的活动工艺功能占用。防止同时执行以下工艺功能： "MC_ReadCamTrackData" "MC_WriteCamTrackData" "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" - 减少凸轮数据量。

ErrorID	错误消息	说明/纠正或避免错误的方法
8043	参数值非法	涉及数据类型为 REAL 的所有输入参数，或者输入参数 <i>StartCam</i>。 适用于输入参数 <i>Hysteresis</i>: - 无限轴 — 值 > 操作范围的 $\frac{1}{4}$ 通过软件限位开关的位置定义操作范围。 - 模数轴 — 值 > 模数长度的 $\frac{1}{4}$
8044	命令不受工艺对象支持	可能的原因: 命令被发送到测量输入。
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是您可能无法通过状态输出对其进行监视。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例: 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应: - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意: 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。

ErrorID	错误消息	说明/纠正或避免错误的方法
8083	DB 不是工艺 DB	通过输入参数 <i>Axis</i> 指定的 DB 未找到或不是工艺 DB。
8084	工艺 DB 无效	- 在通过输入参数 <i>CamTrack</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>CamTrack</i> 更改 DB 号。 - 用户在输入参数 <i>CamTrack</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management) 中删除工艺 DB，然后创建一个新的工艺 DB。
8086	参数不可用	当前无法执行读取/写入命令；重复该命令。
8088	DB 无效 (ANY 指针)	ANY 指针指向无效的数据区域。 - 找不到数据块 - 找不到指定的起始地址 - 数据区域过短
8089	数据记录长度无效 (ANY 指针)	指定的数据长度无效。可能的原因： - 长度超过 576 个字节。 - 指定的长度不能被 18 整除。
808A	DB 中的 REAL 值无效	传送到工艺的 DB 包含至少一个与允许的 REAL 格式规范不兼容的值。 检查所有 DB 值并根据需要进行调整。
808C	指针格式无效或不受支持	ANY 指针包含无效的定义。接受的区域： I、O、M、DB、DI 接受的数据类型： BYTE、CHAR、WORD、INT、DWORD、DINT、REAL

6.6 工艺功能 — 测量输入，外部编码器

6.6.1 FB433 MC_MeasuringInput — 测量输入

使用 FB 433“MC_MeasuringInput”的测量输入

用途

- “MC_MeasuringInput”工艺功能用于通过测量输入启用单个测量的采集。
- 测量输入基于发生的外部事件（例如，压力标记或参考凸轮）记录外部编码器或者组态轴的位置。
- 可以将测量限制为特定的测量范围。将接受由 *StartPosition* 和 *EndPosition* 定义的范围内的第一个测量值。
- 当 *StartPosition = EndPosition* 时，将忽略测量范围。
- 记录和输出测量值后或者取消测量时（例如被其它命令取消）功能将被终止。
- 可以通过输入 *Mode* 基于驱动和使用的测量元素选择多种操作模式。

适用对象

- 测量输入

先决条件

- 启用轴以进行位置控制操作
- 连接的驱动或测量元素支持选定模式。有关详细信息，请参考相关的驱动文档！
- 驱动器接口 IM 174/ADI4 不支持同步执行 MC_MeasuringInput 和 MC_Home 命令。

说明

考虑驱动响应时间。有关详细信息，请参考相关的驱动文档（例如，在“S7-Technology”CD 上）。

超驰命令

MC_MeasuringInput 命令仅可被其它输出到相同 TO 的 MC_MeasuringInput 命令取消。
MC_MeasuringInput 命令无法取消其它命令。

输入参数

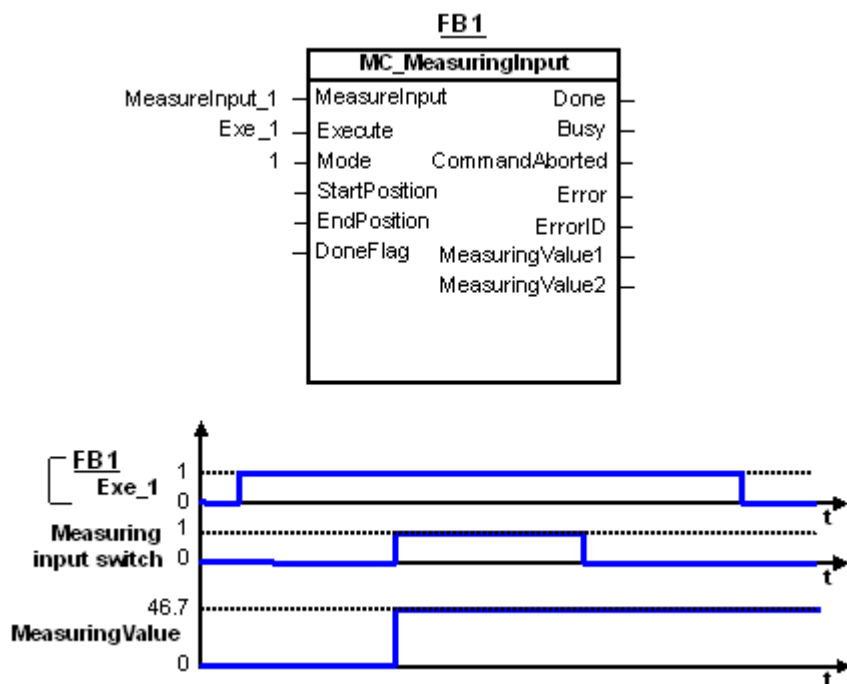
参数	数据类型	初始值	说明
<i>MeasureInput</i>	INT	0	工艺 DB 的编号
<i>Execute</i>	BOOL	<i>FALSE</i>	在上升沿处启动操作模式
<i>Mode</i>	INT	1	操作模式:
			值 = 1: 上升沿处测量
			值 = 2: 下降沿处测量
			值 = 3: 在两个沿上均进行测量，从上升沿开始（两个单独的测量值）
			值 = 4: 在两个沿上均进行测量，从下降沿开始（两个单独的测量值）
			值 = 5: 取消当前测量
<i>StartPosition</i>	REAL	0.0	测量范围的起点
<i>EndPosition</i>	REAL	0.0	测量范围的终点
<i>DoneFlag</i>	INT	0	在 MCDevice DB 中生成 DoneFlag 当返回新测量值后，将转换 DoneFlags 值

输出参数（状态输出）

参数	数据类型	初始值	说明
<i>Done</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 测量已完成；测量值有效
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令
<i>CommandAborted</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 测量命令被取消。 (测量范围内未检测到沿)
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令启动时有错误。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
			<i>FALSE</i> : 命令启动时没有错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 ErrorID。
<i>MeasuringValue1</i>	REAL	0.0	<i>Messwert1</i>
<i>MeasuringValue2</i>	REAL	0.0	<i>Messwert2</i> （仅对两个沿处的测量有效）

MC_MeasuringInput — 示例

测量输入被激活。当检测到测量输入时，实际位置值（例如，46.7）将被保存到 *MeasuringValue1*。



MC_MeasuringInput — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8034	无法记录测量范围内的测量值	可能的原因： - 在指定的测量范围内没有测量事件 - 未对驱动器上的测量输入进行编程
8043	参数值非法	涉及数据类型为 REAL 的所有输入参数，或者输入参数 <i>Mode</i> 或 <i>DoneFlag</i> 。
8044	命令不受工艺对象支持	例如，将命令发送至输出凸轮。
8045	在当前状态下不允许使用该命令	可能的原因： 工艺对象在 S7T Config 中或通过使用“MC_ActivateTO”工艺功能被取消激活。
804A	缺少所需的对象连接	可能的原因： 测量输入的轴在 S7T Config 中或通过使用“MC_ActivateTO”工艺功能被取消激活。

ErrorID	错误消息	说明/纠正或避免错误的方法
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是您可能无法通过状态输出对其进行监视。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8083	DB 不是工艺 DB	通过输入参数 <i>Axis</i> 指定的 DB 未找到或不是工艺 DB。

ErrorID	错误消息	说明/纠正或避免错误的方法
8084	工艺 DB 无效	- 在通过输入参数 <i>Axis</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Axis</i> 更改 DB 号。 - 用户在输入参数 <i>Axis</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management)中删除工艺 DB，然后创建一个新的工艺 DB。
808B	参数值的 REAL 格式无效	数据类型为 REAL 的输入参数的值不符合有效的浮点数格式。 检查输入参数值或背景数据块数据。不能以浮点格式表示格式无效的值。它们以十六进制格式 (DW16# ...) 显示。

6.6.2 FB432 MC_ExternalEncoder — 外部编码器

使用 FB 432“MC_ExternalEncoder”的外部编码器

适用于固件版本为 V3.2.x 的集成工艺

本部分介绍了 V3.1.x 及更低版本固件的应用

用途

- “MC_ExternalEncoder”工艺功能用于控制可以使用的外部编码器，例如，作为用于同步操作的实际引导轴（主轴）。
- 可以启用/禁用外部编码器
- 支持绝对编码器调整
- 同步： 增量编码器的实际位置值受相应设置或被动回原点的影响。

适用对象

- 外部编码器

先决条件

- 编码器值必须有效。检查工艺数据块中的变量 *Statusword.CyclicInterface* 和 *Statusword.EncoderValid*。
- 仅可以修改已启用的外部编码器的实际位置值。

超驰命令

MC_ExternalEncoder 命令仅可被其它输出到相同 TO 的 MC_ExternalEncoder 命令取消。MC_ExternalEncoder 命令无法取消其它命令。

输入参数

参数	数据类型	初始值	说明
<i>Axis</i>	INT	0	工艺 DB 的编号
<i>Execute</i>	BOOL	<i>FALSE</i>	在上升沿处开始
<i>Position</i>	REAL	0.0	位置设定值
<i>Mode</i>	INT	1	操作模式：
			值 = 0: 取消激活外部编码器 位置实际值不再更新。
			值 = 1: 启用外部编码器 (位置实际值被更新)
			值 = 2: 直接回原点： 为当前位置分配了输入参数 <i>Position</i> 的值。
			值 = 3: 通过以下方式被动回原点：“仅编码器零标记”或“仅外部零标记”或“参考凸轮和编码器零标记”（默认为：“仅编码器零标记”）： 已将参考点中输入参数 <i>Position</i> 的值分配给当前位置。
			值 = 4: 直接回原点： 将 S7T Config 中预组态的默认参考位置分配给当前位置。
			值 = 5: 通过以下方式被动回原点：“仅编码器零标记”或“仅外部零标记”或“参考凸轮和编码器零标记”（默认为：“仅编码器零标记”）： 在 S7T Config 中编程的默认参考位置已在参考点分配给当前位置。

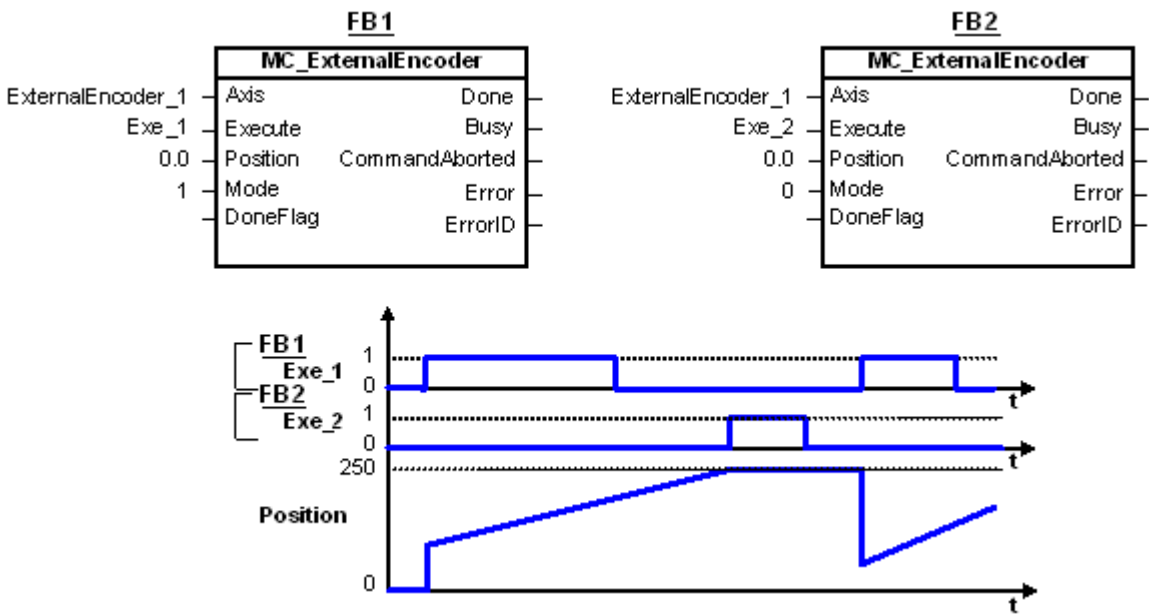
			值 = 6:	绝对编码器调整 将输入参数 <i>Position</i> 的值分配给当前位置。 电源发生故障后，将保留从绝对编码器调整获取的位置偏移。该轴必须处于静止状态，以便进行精确调整。
			值 = 7:	实际值校正 位置值 = 实际位置 - 输入参数 <i>Position</i>
			请注意，除了绝对编码器调整以外，任何位置调整（ <i>Mode</i> = 2, 4 或 7）也将对轴产生影响。在关闭电源或重启后，仅可以激活绝对编码器调整（“MC_Reset”、 <i>Restart</i> = TRUE）。	
<i>DoneFlag</i>	INT	0	在 MCDevice DB 中生成 DoneFlag	

输出参数（状态输出）

参数	数据类型	初始值	说明	
<i>Done</i>	BOOL	FALSE	TRUE: 命令已完成	
<i>Busy</i>	BOOL	FALSE	TRUE: 正在执行命令	
<i>CommandAborted</i>	BOOL	FALSE	TRUE: 命令由其它命令或因其执行过程中的错误而取消。 如果工艺 DB 的 <i>ErrorStatus</i> 中未显示错误，则该命令由后续命令取消。 如果工艺 DB 的变量 <i>ErrorStatus</i> 中指示有错误，则命令执行期间影响该工艺对象的错误将导致该命令中止。	
<i>Error</i>	BOOL	FALSE	TRUE:	命令启动时有错误。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
			FALSE:	命令启动时没有错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 ErrorID。	

MC_ExternalEncoder — 示例

启用外部编码器。在启用之前，位置值为零。直到完成激活后，位置值才会更新。如果随后禁用编码器，则不管该编码器是否可以移动，位置值将不再更新。再次启用编码器时，位置将更改为实际值。



MC_ExternalEncoder — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8022	轴/外部编码器中没有实际值	例如，编码器或数据总线未准备就绪
8040	轴/外部编码器被禁用，或者设置了错误的模式	禁用外部编码器。必须启用外部编码器才能使用 <i>Mode > 1</i> 调用“MC_ExternalEncoder”。 补救措施： - 使用 <i>Mode = 1</i> 通过工艺功能“MC_ExternalEncoder”启用外部编码器。 - 消除并确认可能禁用了外部编码器的所有错误，然后通过“MC_ExternalEncoder”工艺功能中设置 <i>Mode = 1</i> 启用外部编码器。
8043	参数值非法	涉及数据类型为 <i>REAL</i> 的所有输入参数，或者输入参数 <i>Mode</i> 或 <i>DoneFlag</i>。 例如， <i>Mode = 167</i>

ErrorID	错误消息	说明/纠正或避免错误的方法
8044	命令不受工艺对象支持	例如，命令要求有一个轴
8045	在当前状态下不允许使用该命令	示例：启动命令（例如，关闭硬盘）时，编码器值无效。
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是您可能无法通过状态输出对其进行监视。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8083	DB 不是工艺 DB	通过输入参数 <i>Axis</i> 指定的 DB 未找到或不是工艺 DB。

ErrorID	错误消息	说明/纠正或避免错误的方法
8084	工艺 DB 无效	- 在通过输入参数 <i>Axis</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Axis</i> 更改 DB 号。 - 用户在输入参数 <i>Axis</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management)中删除工艺 DB，然后创建一个新的工艺 DB。
808B	参数值的 REAL 格式无效	数据类型为 REAL 的输入参数的值不符合有效的浮点数格式。 检查输入参数值或背景数据块数据。 不能以浮点格式表示格式无效的值。 它们以十六进制格式 (DW16# ...) 显示。

6.7 工艺功能 — 基本功能

6.7.1 FB402 MC_Reset — 确认错误

通过调用 FB 402“MC_Reset”确认错误

用途

- 调用“MC_Reset”工艺功能以确认接受用户程序中的确认的所有错误。只能通过断开、接通电源或将项目数据再次下载到模块来确认致命错误。确认还将清除 *ErrorID* 和 *ErrorBuffer* 参数中的条目，并重设工艺 DB 中的 *ErrorStatus* 位。
- 调用工艺功能“MC_Reset” *Restart = TRUE* 可以启动轴、输出凸轮、凸轮轨迹、凸轮盘、外部编码器和测量输入的初始化（重启）。轴初始化后，具有增量编码器的轴将处于“非回原点”状态。最初在 S7T Config 中组态的凸轮将在凸轮盘上恢复。
- 如果将工艺功能“MC_Reset”（*Restart = TRUE* 或 *FALSE*）应用到工艺数据块“MCDevice”（*Axis = MCDevice* DB 的 DB 编号），则命令执行的最大运行时间将被复位为 0（MCDevice DB 中的 *MaxLoopDuration*）。当使用固件版本为 V3.2x 或更高版本的集成工艺时，系统也会将 *TODBTaskOverflows*（工艺 DB 更新中的溢出数）变量设置为 0。
- 重启过程中，系统将继续执行命令。但是，执行重启的工艺对象在重启过程中不能接受任何命令。

适用对象

- 速度控制轴
- 定位轴
- 同步轴
- 输出凸轮
- 凸轮轨迹
- 测量输入
- 凸轮
- 外部编码器
- MCDevice-DB

重启条件

- “轴”工艺对象
仅当禁用轴时才能进行重启 (“MC_Power"输出参数 *Status = FALSE* 且 *Busy = FALSE*)
- “外部编码器”工艺对象
仅当禁用外部编码器时才能进行重启 (工艺 DB, 变量 *Statusword.DriveEnabled = FALSE*)
- “输出凸轮”工艺对象
仅当禁用输出凸轮时才能进行重启 (工艺 DB 变量 *Mode = 1*)
- “凸轮轨迹”工艺对象
仅当禁用输出凸轮时才能进行重启 (工艺 DB 变量 *Control = 2 / INACTIVE*)
- “测量输入”工艺对象
仅当测量输入处于非激活状态时才能进行重启 (工艺 DB 变量 *Status = 0, 2 或 3*)
- “凸轮”工艺对象
仅当凸轮未处于使用状态时才能进行重启 (工艺 DB, 变量 *UserCount = 0*)

说明

如果您仅希望清除工艺对象的错误, 请置位 *Restart = FALSE*。

工艺对象 (*Restart = TRUE*) 的重新初始化可能会花费很长时间。在此过程中, 无法使用该工艺对象。

命令的相互作用

新命令 — 活动的单个命令 (3)

输入参数

参数	数据类型	初始值	说明
<i>Axis</i>	INT	0	工艺 DB 的编号
<i>Execute</i>	BOOL	<i>FALSE</i>	在上升沿处启动命令
<i>Restart</i>	BOOL	<i>FALSE</i>	<i>FALSE</i> : 确认错误
			<i>TRUE</i> : 重启 — 初始化以及激活已修改的组态参数

输出参数（状态输出）

参数	数据类型	初始值	说明
<i>Done</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 已确认错误或者执行重启
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令
<i>CommandAborted</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令由其它命令或因其执行过程中的错误而取消。 如果工艺 DB 的 <i>ErrorStatus</i> 未报告错误, 则该命令由后续命令取消。 如果 <i>ErrorStatus</i> 中指明存在错误, 则命令执行期间影响该工艺对象的错误将导致该命令中止。
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令启动时有错误。未执行该命令。有关原因的信息, 请参考 <i>ErrorID</i> 。
			<i>FALSE</i> : 命令启动时没有错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。

MC_Reset — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8045	在当前状态下不允许	"MC_Reset"(<i>Restart = TRUE</i>) 已激活。命令被取消
8047	无法确认该错误	不能通过调用工艺功能"MC_Reset"确认错误。示例： - 轴已停止，无法确认导致轴停止的错误。 - 在确认期间或重启阶段发生了更多错误。 - 叠加限制动态值的命令（请参阅同步工艺 DB 中的警告 0021） 消除导致错误的所有原因，然后重新尝试确认。请注意，您必须通过断开然后再接通电源来确认特定的错误。
8048	无法重启	- 仅在禁用轴时才允许重启。 - 其它应用程序正在使用凸轮盘，无法复位。

ErrorID	错误消息	说明/纠正或避免错误的方法
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是您可能无法通过状态输出对其进行监视。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8083	DB 不是工艺 DB	通过输入参数 <i>Axis</i> 指定的 DB 未找到或不是工艺 DB。
8084	工艺 DB 无效	- 在通过输入参数 <i>Axis</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Axis</i> 更改 DB 号。 - 用户在输入参数 <i>Axis</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management)中删除工艺 DB，然后创建一个新的工艺 DB。

6.7.2 FB 460 MC_ActivateTO — 取消激活/激活工艺对象

使用 FB 460“MC_ActivateTO”取消激活/激活工艺对象

用途

- 如果要从集成工艺的循环处理中临时或永久删除已组态的工艺功能，可使用“MC_ActivateTO”工艺功能。对于标准操作，无需激活预先未取消激活的工艺对象。
- 更好地利用集成工艺的计算性能。使用“MC_ActivateTO”工艺功能可从集成工艺的循环处理中删除不需要的工艺对象。
- 工艺对象被取消激活后，将不再在集成工艺中对其进行处理。已取消激活的工艺对象不能再接受任何命令，并且不能再更新相应的工艺 DB，直到它再次被激活。
- 激活工艺对象会使先前的取消激活无效。
- 通过使用 *Mode = 0* 可以读出工艺对象的激活状态。如果要检查当前激活状态或取消激活状态，可在另一个实例中调用“MC_ActivateTO”工艺功能。

说明

也可在 S7T Config 中永久取消激活工艺对象。

在浏览器中选择“工艺”(Technology) 对象，然后选择“编辑”(Edit) > “对象状态”(Object states) 命令。

工艺对象的取消激活存储在项目中，并且在断电、通电和重启后仍会保留。

请务必遵守相应在线帮助中的相关说明。

适用对象

- 轴（实际轴和虚拟轴）
对于同步轴，相应的同步对象也会被取消激活/激活。
- 外部编码器
- 输出凸轮
- 凸轮轨迹
- 测量输入
- 凸轮

要求

- T-CPU 必须处于循环操作状态（“RUN”状态）。
- 工艺对象不能处于激活或使用状态。

超驰命令

MC_ActivateTO 命令不能被其它任何命令取消。

MC_ActivateTO 命令不会取消其它任何命令。

输入参数

参数	数据类型	初始值	说明	
轴	INT	0	工艺 DB（属于相应工艺对象）的编号	
执行	BOOL	FALSE	在上升沿处启动命令	
Mode	INT	0	值 = 0:	在 <i>Status</i> 输出参数中提供工艺对象的当前激活状态。
			值 = 1	激活工艺对象
			值 = 2	取消激活工艺对象

输出参数（状态输出）

参数	数据类型	初始值	说明	
Done	BOOL	FALSE	命令已完成： - 工艺对象已激活 - 工艺对象已取消激活 - 激活状态已读出	
Busy	BOOL	FALSE	TRUE	正在执行命令
Error	BOOL	FALSE	FALSE	命令启动时没有错误。
			TRUE	命令启动时有错误。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
ErrorID	WORD	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。	
Status	INT	0	<i>Done</i> = TRUE 时，显示工艺对象的当前激活状态。仅当 <i>Mode</i> = 0 时，才会显示状态。	
			值 = 0	激活状态未生效，或尚未更新。
			值 = 1	工艺对象已激活。
			值 = 2	工艺对象已取消激活。
			值 = 3	工艺对象正在激活。激活尚未完成。
			值 = 4	工艺对象正在取消激活。取消激活尚未完成。

取消激活工艺对象时，需在程序中考考虑以下条件：

- 确保先取消激活从属工艺对象（输出凸轮、凸轮轨迹和测量输入）。
- 确保在激活凸轮盘时未使用这些工艺对象。
- 如果想要取消激活用作阀控曲线的凸轮盘，则禁用已分配的液压轴。

在用户程序中取消激活工艺对象时，请采用以下步骤：

1. 在要取消激活的工艺对象上终止所有命令。一直等到所有运动命令均已终止。
2. 取消激活或禁用工艺对象。
3. 使用“MC_ActivateTO”工艺功能取消激活工艺对象。

驱动器或 DP 从站的所有工艺对象都已取消激活后，即可使用“MC_ActivateDP Slave”工艺功能取消激活相应的 DP 从站。为此，请遵守『使用 FB 457 MC_ActivateDP Slave 取消激活/激活 DP 从站』一章中的说明。

说明

取消激活和更新激活后的组态数据

操作期间更改的组态数据即使在更新激活后也会保持不变。如果在取消激活前没有执行必要的重启，则必须在更新激活后执行重启。

取消激活工艺对象时，无法更改工艺对象的组态数据。

说明

取消激活和更新激活后的系统变量

将在更新激活后重新初始化系统变量，如同重启了工艺对象一样。重新初始化将产生以下影响：

- 具有增量编码器的参考轴将丢失其回原点。
 - 就测量输入而言，先前的测量值将被删除。
 - 基本坐标系或高级坐标系中的位置偏移将被删除。
-

说明

如果通过使用“MC_ActivateTO”工艺功能或在 S7T Config 中取消激活轴工艺对象，则相应的驱动器可指示 S7T Config 中的错误，因为与该驱动器的循环通信已停止。

MC_ActivateTO — ErrorID

适用于固件版本为 V4.1.x 或更高版本的集成工艺

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8043	参数值非法	输入参数 <i>Mode</i> 中的值无效。
8045	在当前状态下不允许使用该命令	可能的原因： - 取消激活期间，使用"MC_Power"工艺功能启用轴。 - 工艺对象的命令尚未完成，例如，仍在执行重启。
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是您可能无法通过状态输出对其进行监视。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。

ErrorID	错误消息	说明/纠正或避免错误的方法
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8054	不允许取消激活工艺对象	不能取消激活工艺对象，因为该工艺对象仍在使用。
8057	取消激活/激活命令已激活	一直等到取消激活/激活命令完成后再启动新命令。
8083	DB 不是工艺 DB	通过输入参数 <i>Axis</i> 指定的 DB 未找到或不是工艺 DB。
8084	工艺 DB 无效	- 在通过输入参数 <i>Axis</i> 指定的工艺 DB 的 PLC 中不存在工艺对象。 - 将当前工艺下载到目标系统，或通过输入参数 <i>Axis</i> 更改 DB 号。 - 用户在工艺 DB（通过输入参数 <i>Axis</i> 指定）写入的数据无效。 - 在“工艺对象管理”(Technology Objects Management) 中，删除该工艺 DB，然后重新创建一个工艺 DB

6.7.3 FB 457 MC_ActivateDP Slave — 取消激活/激活 DP 从站

使用 FB 457“MC_ActivateDP Slave”取消激活/激活 DP 从站

适用于固件版本为 V4.1.x 或更高版本的集成工艺

用途

- 如果想临时或永久取消激活 DP(DRIVE) 的已组态 DP 从站，可使用“MC_ActivateDP Slave”工艺功能。这在某些情况下很有用，例如，要将工厂单元临时从 DP(DRIVE) 分离时。对于标准操作，无需激活预先未取消激活的 DP 从站。
- 取消激活的 DP 从站不会指示 T-CPU 的总线故障。
- 取消激活的 DP 从站不再参与 DP(DRIVE) 的循环数据交换。可在 HW Config 中设置的 DP 周期的最小时间段仍取决于组态 DP 从站数。
- 如果取消激活的 DP 从站仍保留在 DP(DRIVE) 中，则可通过 S7T Config 的 STARTER 功能继续对其组态。
- 通过使用 *Mode = 0* 可以读出 DP 从站的激活状态。如果要检查当前激活状态或取消激活状态，可在另一个实例中调用“MC_ActivateDP Slave”工艺功能。

适用对象

- DP(DRIVE) 上的任何 DP 从站

要求

- T-CPU 必须处于循环操作状态（“RUN”状态）。
- 必须取消激活了引用 DP 从站地址的工艺对象。

超驰命令

MC_ActivateDP Slave 命令不能被其它任何命令取消。

MC_ActivateDP Slave 命令不会取消其它任何命令。

输入参数

参数	数据类型	初始值	说明	
Diagnostic address	INT	0	DP 从站的诊断地址（请参见 HW Config 中的条目）	
Execute	BOOL	FALSE	在上升沿处启动命令	
Mode	INT	0	值 = 0:	在 <i>Status</i> 输出参数中提供 DP 从站的当前激活状态。
			值 = 1	激活 DP 从站 建立与 DP 从站的循环通信。
			值 = 2	取消激活 DP 从站 断开与 DP 从站的循环通信。

输出参数（状态输出）

参数	数据类型	初始值	说明	
Done	BOOL	FALSE	命令已启动： - 激活命令已激活 如果 DP(DRIVE) 没有要激活的 DP 从站，则激活 DP(DRIVE) 上具有连接的 DP 从站。 - 取消激活命令已启动 如果已从 DP(DRIVE) 删除要取消激活的 DP 从站，则总线故障 LED 也会熄灭。条目“站返回”(Station return) 将被输入 CPU 和集成工艺的诊断缓冲区。 - 激活状态已读出 <i>Mode = 0</i> 时调用“MC_ActivateDPSlave”工艺功能可检查 DP 从站的激活/取消激活状态。	
Busy	BOOL	FALSE	TRUE	正在执行命令
Error	BOOL	FALSE	FALSE	命令启动时没有错误。
			TRUE	命令启动时有错误。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。

参数	数据类型	初始值	说明
ErrorID	WORD	0	输出参数 <i>Error</i> 的 ErrorID。
Status	INT	0	<i>Done = TRUE</i> 时，显示 DP 从站的当前激活状态。 仅当 <i>Mode = 0</i> 时，才会显示状态。
			值 = 0 激活状态未生效，或尚未更新
			值 = 1 DP 从站已激活
			值 = 2 DP 从站已取消激活或不在 DP(DRIVE) 上
			值 = 3 DP 从站正在激活。激活尚未完成。 如果持续显示状态值 3，则说明 DP 从站有故障。
			值 = 4 不显示状态值 4。
			值 = 5 DP 从站未连接到 DP(DRIVE)

说明

取消激活 DP 从站前，请确保所连接的所有工艺对象均已取消激活。为此，请遵守『MC_ActivateTO — 取消激活/激活工艺对象』一章中的说明。

取消激活了所连接的所有工艺对象后，即可取消激活 DP 从站，然后将其从总线删除。

MC_ActivateDPSlave — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一个周期内调用这些工艺功能，直到输出参数 <i>Done</i>, <i>CommandAborted</i> 或 <i>Error</i> 中有一个为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8043	参数值非法	<i>Mode</i> . 输入参数中的值无效
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。

ErrorID	错误消息	说明/纠正或避免错误的方法
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8057	取消激活/激活命令已激活	一直等到取消激活/激活命令完成后再启动新命令。
8058	无法组态 DP 从站	在 DP 从站激活期间，无法从 DP 主站装载 DP 从站的组态。检查从站的硬件。
8090	逻辑地址无效	指定的诊断地址未被分配给任何 DP 从站。

6.7.4 FB406 MC_ReadSysParameter — 读取参数

使用 FB 406“MC_ReadSysParameter”读取参数

用途

- MC_ReadSysParameter 工艺功能返回 TO 最重要的系统变量值和组态数据，例如，轴或输出凸轮数据。通过设置参数编号对系统变量和组态数据进行寻址
- 根据读取的参数的数据类型，将结果传送至相应的输出参数。其它输出返回 0 值。
- 可以使用输入参数 *Index* 访问参数的多个实例，以下区域存在多个参数：
 - 数据集 (*Dataset_1* 到 *Dataset_16*)
 - 编码器 (*Encoder_1* 到 *Encoder_8*)
 - 同步对象 (1 代表同步对象，2 代表叠加的同步对象)

适用对象

- 系统变量和组态数据

要求

- 由参数编号和索引定义的参数必须在 S7T Config 中组态
- 在读取编码器参数之前，先检查编码器值的有效性。通过读取轴或外部编码器的工艺 DB 中的变量 *Statusword.EncoderValid* 和 *Statusword.CyclicInterface*，可以验证当前活动编码器的值。如果为轴分配了多个数据记录，则可以通过使用参数编号 4050 调用“MC_ReadSysParameter”工艺功能来请求其它编码器的状态。

超驰命令

MC_ReadSysParameter 命令不能由任何其它命令取消。

MC_ReadSysParameter 命令不取消任何其它命令。

输入参数

参数	数据类型	初始值	说明
<i>Axis</i>	INT	0	工艺 DB 的编号
<i>Execute</i>	BOOL	FALSE	在上升沿处请求参数值
<i>ParameterNumber</i>	INT	0	要读取的参数的编号
<i>Index</i>	INT	1	多个参数编号（例如，2 代表 <i>Datensatz_2</i> ）

输出参数（状态输出）

参数	数据类型	初始值	说明
<i>Done</i>	BOOL	FALSE	TRUE: 根据参数数据类型，可以通过输出 <i>Value</i> 、 <i>ValueDInt</i> 、 <i>ValueDword1</i> 、 <i>ValueDword2</i> 或 <i>ValueBool</i> 使用请求的参数值
<i>Busy</i>	BOOL	FALSE	TRUE: 正在执行命令
<i>Error</i>	BOOL	FALSE	TRUE: 命令启动时有错误。该命令被拒绝。
			FALSE: 命令启动时没有错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。
<i>Value</i>	REAL	0.0	类型为 REAL 时指定参数的值（其它类型为 0.0）。
<i>ValueDInt</i>	DINT	0	类型为 DINT 时，为指定参数的值（否则为 0）。ENUM 值也通过该参数输出。
<i>ValueDword1</i>	DWORD	0	类型为 DWORD 且长度为 1 DWORD（其它类型为 0）时指定参数的值。
<i>ValueDword2</i>	DWORD	0	类型为 DWORD 且长度为 2 DWORD（其它类型为 0）时指定参数的值。
<i>ValueBool</i>	BOOL	FALSE	类型为 BOOL 时指定参数的值（其它类型为 FALSE）。

MC_ReadSysParameter — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
8043	参数值非法	涉及输入参数 <i>ParameterNumber</i> 或 <i>Index</i> 。
8044	命令不受工艺对象支持	例如，将命令发送到 MCDevice DB
8045	在当前状态下不允许使用该命令	可能的原因： 使用“MC_ActivateTO”工艺功能取消激活了同步轴
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是您可能无法通过状态输出对其进行监视。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。

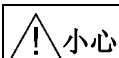
ErrorID	错误消息	说明/纠正或避免错误的方法
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： • 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 • 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8080	参数编号或参数索引无效	工艺对象不支持此参数或参数索引。
8083	DB 不是工艺 DB	通过输入参数 <i>Axis</i> 指定的 DB 未找到或不是工艺 DB。
8084	工艺 DB 无效	• 在通过输入参数 <i>Axis</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Axis</i> 更改 DB 号。 • 用户在输入参数 <i>Axis</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management) 中删除工艺 DB，然后创建一个新的工艺 DB。
8086	参数不可用	可能的原因： • 在具有绝对值编码器的系统中访问增量编码器参数 • 访问以下类型编码器的参数：“绝对编码器周期性绝对” • 索引中定义的数据记录未处于活动状态。必须通过调用“MC_ChangeDataset”来更改数据记录。 • 当前，工艺对象正在执行重启。因此，无法读取参数。

6.7.5 FB407 MC_WriteParameter — 更改参数

使用 FB 407“MC_WriteParameter”更改参数

用途

- 当 T-CPU 处于 RUN 模式时，可以使用“MC_WriteParameter”工艺功能编辑 TO 最重要的系统变量和组态数据。通过设置参数编号对系统变量和组态数据进行寻址。在再次断电/通电或 CPU 执行存储器复位前，更改将一直有效。例外情况：在任何情况下，绝对值编码器调整期间设置的值都将保留。
- 在输入参数 *Value*、*ValueDInt*、*ValueDword1*、*ValueDword2* 和 *ValueBool* 中，系统将仅评估与相关工艺参数相符的参数（有关详细信息，请参考工艺参数列表）。
- 工艺功能仅可以修改由访问模式“W”（“写入”）标识的工艺参数列表中的参数。
- 可以使用输入参数 *Index* 访问参数的多个实例，以下区域存在多个参数：
 - 数据集（Dataset_1 到 Dataset_16）
 - 编码器（Encoder_1 到 Encoder_8）
 - 同步对象（1 代表同步对象，2 代表叠加的同步对象）

**小心**

此工艺功能的使用专门针对有经验用户。错误的输入可能导致不受控制的驱动器响应！

系统不对您的输入进行似然性检查！

错误的输入可能会导致工艺对象不一致。

适用对象

- 在 S7T Config 中组态的工艺对象的参数。
请注意，某些参数更改要求重启系统（有关信息，请参考工艺参数列表的“激活”列。）

先决条件

- 由参数编号和索引定义的参数必须在系统中可用并且允许写访问。必须使用工艺功能“MC_ChangeDataset”在 T-CPU 中启用由索引定义的数据记录。
- 在写入编码器参数（例如，绝对编码器调整）之前验证编码器值。可以查询当前激活的编码器的编码器值，并通过轴或外部编码器的工艺 DB 中的 *Statusword.EncoderValid* 变量检查其有效性。如果为轴分配了多个数据记录，则可以通过使用参数编号 4050 调用“MC_ReadSysParameter”工艺功能来请求其它编码器的状态。

超驰命令

MC_WriteParameter 命令不能由任何其它命令取消。

MC_WriteParameter 命令不取消任何其它命令。

输入参数

参数	数据类型	初始值	说明
<i>Axis</i>	INT	<i>0</i>	工艺 DB 的编号
<i>Execute</i>	BOOL	<i>FALSE</i>	在 <i>Execute</i> 的上升沿处写入参数
<i>ParameterNumber</i>	INT	<i>0</i>	工艺参数的编号
<i>Index</i>	INT	<i>1</i>	例如，参数字段编号， <i>2</i> 代表 dataset_2 （如果仅存在一个数据集编号，将忽略该值）。
<i>Value</i>	REAL	<i>0.0</i>	如果类型为 REAL ，则为指定参数的新值（否则忽略该值）。
<i>ValueDInt</i>	DINT	<i>0</i>	类型为 DINT 指定参数的新值（否则忽略该值）。
<i>ValueDword1</i>	DWORD	<i>0</i>	类型为 DWORD 且长度为 <i>1</i> DWORD 时定义参数的新值（否则忽略该值）。
<i>ValueDword2</i>	DWORD	<i>0</i>	类型为 DWORD 且长度为 <i>2</i> DWORD 时定义参数的新值（否则忽略该值）。
<i>ValueBool</i>	BOOL	<i>0</i>	类型为 BOOL 指定参数的新值（否则忽略该值）。

输出参数（状态输出）

参数	数据类型	初始值	说明
<i>Done</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 工艺参数已成功更改
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令启动时有错误。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
			<i>FALSE</i> : 命令启动时没有错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。

注意

观察参数的有效范围限制。工艺参数列表中列出了有效范围限制。

编辑数据记录参数时，请注意，数据记录中的某些参数必须相等：

- 更改结构的参数（例如，不同的控制器类型，如 PV 或 PID 控制器）
- 重要的激活参数（例如，跟踪监视开/关、DSC 开/关）

如果通过工艺功能“MC_WriteParameter”在运行期间为数据记录参数分配了无效的值，则系统将报告组态错误。因此，您应该仅编辑在 S7T Config 中对其进行组态时可以编译为一致项目的数据记录参数。

说明

要更改不需要重启或更改后立即生效的几个参数，您应该搜集所有参数更改，然后将其激活。

要执行此操作，请按以下步骤进行操作：

1. 使用 "MC_WriteParameter" 将工艺参数 *ParameterNumber* = 4001 (*activationmodechangedconfigdata*) 设置为 *COLLECT_CHANGED_CONFIG_DATA* (82)。

这就形成了对相关工艺对象的所有后续更改的集合。

2. 使用“MC_WriteParameter”更改相关工艺参数。
3. 使用“MC_WriteParameter”将 *activationmodechangedconfigdata* 设置为 *ACTIVATE_CHANGED_CONFIG_DATA* (81) 以应用所有更改。

当编辑需要所有数据记录中的值均相同的工艺参数时，将调用此步骤。

MC_WriteParameter — 示例 —“倍率”

使用“倍率”更改参数的示例

因子可以根据当前运行速度或加速度/减速度进行在线叠加。速度倍率适用于速度，而加速倍率适用于加速度和减速度。

设置以下输入参数值，以设置**速度**倍率：

ParameterNumber = 4142

Value = 0.0 到 200.0（为百分比值）

为**加速/减速**倍率设置以下输入参数值：

ParameterNumber = 4141

Value = 1.0 到 1000.0（为百分比值）

MC_WriteParameter — 示例 —“驱动器控制字”

通过 PROFIBUS 耦合的驱动器使用标准消息帧进行通信。这些消息帧包含最多两个驱动器控制字。以下字的内容在 PROFIdrive 标准中指定。

控制字 STW1 的结构			控制字 STW2 的结构		
位	为工艺对象保留	PROFIdrive V3.1 定义	位	为工艺对象保留	PROFIdrive V3.1 定义
0	X	驱动器开（不是 AUS1）	0		任意
1	X	操作条件（不是 AUS2）	1		任意
2	X	操作条件（不是 AUS3）	2		任意
3	X	整流器启用	3		任意
4	X	斜坡加速生成器已启用	4		任意
5	X	启动斜坡函数生成器	5		任意
6	X	设定值启用	6		任意
7	X	复位数据块 DBx 的存储器	7		任意
8		步长距离 1（可选）	8		任意
9		步长距离 2（可选）	9		任意
10	X	所需的控制	10		任意
11		任意	11		任意
12		任意	12	X	主站设备状态
13		任意	13	X	主站设备状态
14		任意	14	X	主站设备状态
15		任意	15	X	主站设备状态

可以通过“MC_WriteParameter”设置参数编号 *4500* 写入驱动器控制字，或者通过调用“MC_ReadSysParameter”读取驱动器控制字。两种控制字均可以通过输出参数 *ValueDword1*（未经掩码）读取。

使用相应的值和掩码写入控制字。掩码确定通过写操作可以访问控制字的哪些位。此掩码可用于操纵单个或多个位。未经掩码的位（值 *0*）保留其当前状态。选择掩码的标准是为此工艺对象保留的位保持不变。

通过输入参数 *ValueDword1* 定义控制字 *STW1* 和 *STW2* 的值。控制字 *STW1* 和 *STW2* 对应的掩码值通过输入参数 *ValueDword2* 设置。

ValueDword1 的示例

<i>DW#16#</i>	<i>0001</i>	<i>0300</i>
	<i>STW2</i>	<i>STW1</i>

ValueDword2 的示例

<i>DW#16#</i>	<i>0FFF</i>	<i>FB00</i>
	<i>STW2</i> 掩码	<i>STW1</i> 掩码

如果要改写控制字的所有自由位，请为控制字 *STW1* 设置掩码值 *FB00*。十六进制代码的含义：

F = *1111* 写入位 12 到 15

B = *1011* 写入位 8、9 和 11；保护位 10

0 = *0000* 保护位 4 到 7

0 = *0000* 保护位 0 到 3

如果要改写控制字的所有自由位，请为控制字 *STW1* 设置掩码值 *0FFF*。十六进制代码的含义：

0 = *0000* 保护位 12 到 15

F = *1111* 写入位 8 到 11

F = *1111* 写入位 4 到 7

F = *1111* 写入位 0 到 3

MC_WriteParameter — 示例 — “绝对编码器调整数据的备份”

基于示例“绝对编码器调整数据的备份”编辑参数

绝对编码器调整的数据被保存在 T-CPU 的非易失性存储器中。因此，更换 CPU 时这些数据将丢失。

“MC_ReadSysParameter”和“MC_WriteParameter”工艺功能可用于将绝对编码器调整的数据备份到 MMC 卡，以便在更换 CPU 后再将其恢复到 T-CPU 的非易失性存储器中。先决条件是，您已在 S7T Config 中设置了“绝对编码器”类型。

使用“MC_Home”工艺功能完成绝对编码器调整后，请根据下述内容继续操作：

备份绝对编码器调整数据

步骤	说明
1.	通过调用“MC_ReadSysParameter”工艺功能以及读取下述输入参数来读取绝对编码器调整的当前值，例如： • <i>Axis</i> = 相关轴的工艺 DB • <i>Execute</i> = 用于启动读取操作的上升沿 • <i>ParameterNumber</i> = 4010 绝对编码器调整的参数编号 • <i>ValueDword1</i> = DBx.DBD0 （临时存储位置） • <i>ValueDword2</i> = DBx.DBD4 （临时存储位置）
2.	调用 SFC82“CREA_DBL”以在 T-CPU 的加载存储器中生成 DBy。通过输入参数 <i>ATTRIB</i> 重设为 0 时，也会在工作存储器中生成 DB。
3.	调用 SFC84“WRIT_DBL”以将先前备份的绝对编码器调整的数据写入新 DB。

更换 CPU 并恢复绝对编码器调整数据

步骤	说明
1.	更换 T-CPU，然后插入包含组态数据的 MMC。 请注意，启动 T-CPU 后，可以在 RAM 中获得绝对编码器调整的备份 DB。
2.	调用“MC_WriteParameter”工艺功能以将绝对编码器调整的数据重新存储到 T-CPU 的非易失性存储器中。使用以下输入参数： • <i>Axis</i> = 相关轴的工艺 DB • <i>Execute</i> = 用于启动写入操作的上升沿 • <i>ParameterNumber</i> = 4010 = 绝对编码器调整的参数编号 • <i>ValueDword1</i> = DBx.DBD0 (x = 由 SFC82 生成的 DB 编号) • <i>ValueDword2</i> = DBx.DBD4 (x = 由 SFC82 生成的 DB 编号)

MC_WriteParameter — ErrorID

ErrorID	警告	说明/纠正或避免错误的方法
0000	没有错误	-
0020	重启以激活参数更改	更改了一个参数后，通常需要重启工艺对象。如果新的参数值与旧参数值一致（例如，参数值“5”被修改为“5”），则无需重启。 是否需要重启的信息包含在工艺数据块中。 - 对于轴和外部编码器，<i>RequestRestart</i> 位是在状态字中设置的。 - 技术数据块的 ErrorID 0020 指示输出凸轮和测量输入是否需要重启。

ErrorID	错误消息	说明/纠正或避免错误的方法
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。

ErrorID	错误消息	说明/纠正或避免错误的方法
8043	参数值非法	涉及数据类型为 REAL 的所有输入参数，或者输入参数 <i>Value</i>、<i>ValueDInt</i>、<i>ValueDword1</i> 或 <i>ValueDword2</i>。 如果与其它参数存在任何交互作用，也可能会出现该错误。 示例： - 参数 1100 如果减少丝杠螺距（参数 1100），则轴无法再达到最大速度。 也要调整最大速度（参数 8）。 - 参数 2020 和 2021 如果更改传动系数（参数 2020 或 2021），则轴可能无法再达到最大速度。 也要调整最大速度（参数 8）。 - 参数 8 如果最大速度（参数 8）小于用于动态跟随误差监视的最小速度，则增加用于动态跟随误差监视的最小速度（参数 2072）。 - 选定 <i>ENUM</i> 的值无效
8044	命令不受工艺对象支持	例如，将命令发送到 MCDevice DB
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是您可能无法通过状态输出对其进行监视。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。

ErrorID	错误消息	说明/纠正或避免错误的方法
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8080	参数编号或参数索引无效	工艺对象不支持此参数或参数索引。
8081	无法更改参数	所选的参数是只读参数。 请检查所选参数的编号以及工艺对象的 ID。
8083	DB 不是工艺 DB	通过输入参数 <i>Axis</i> 指定的 DB 未找到或不是工艺 DB。
8084	工艺 DB 无效	- 在通过输入参数 <i>Axis</i> 定义的工艺 DB 的控制器中不存在工艺对象。 将当前工艺下载到目标系统，或通过输入参数 <i>Axis</i> 更改 DB 号。 - 用户在输入参数 <i>Axis</i> 中定义的工艺 DB 中输入了无效数据。 在“工艺对象管理”(Technology Objects Management)中删除工艺 DB，然后创建一个新的工艺 DB。

ErrorID	错误消息	说明/纠正或避免错误的方法
8086	参数不可用	可能的原因： • 在具有绝对值编码器的系统中访问增量编码器参数 • 访问以下类型编码器的参数：“绝对编码器周期性绝对” • 索引中定义的数据记录未处于活动状态。必须通过调用“MC_ChangeDataset”来更改数据记录。 • 当前，工艺对象正在执行重启。因此，无法读取参数。
808B	参数值的 REAL 格式无效	数据类型为 REAL 的输入参数的值不符合有效的浮点数格式。 检查输入参数值或背景数据块数据。不能以浮点格式表示格式无效的值。它们以十六进制格式 (DW16# ...) 显示。

6.7.6 FB450 MC_ReadPeriphery — 读取工艺 I/O

使用 FB 450“MC_ReadPeriphery”读取工艺 I/O

受固件版本为 V3.1.x 或更高版本的集成工艺的支持

用途

- “MC_ReadPeriphery”工艺功能用于读取集成工艺的 I/O 映像区。示例中显示了更新的时间从属性。

适用对象

- 地址区 0 到 63 中 DP(DRIVE) 上的 DP I/O。
- 位于地址区 0 到 63 的 T-CPU 的集成 I/O

先决条件

- 寻址的 I/O 必须存在于集成工艺的 I/O 映像区 DP(DRIVE) 中。

超驰命令

无法使用任何其它命令取消 MC_ReadPeriphery 命令。

MC_ReadPeriphery 命令不会取消任何其它命令。

输入参数

参数	数据类型	初始值	说明
<i>Enable</i>	BOOL	<i>FALSE</i>	只要 <i>Enable</i> = <i>TRUE</i> 就会读取 I/O 映像区的一个区域。输入参数 <i>InputOutput</i> 、 <i>ByteAddress</i> 和 <i>Data</i> 的更改只能通过 <i>Enable</i> 的上升沿激活。
<i>InputOutput</i>	BOOL	<i>FALSE</i>	要读取的 I/O 区域的定义：
			<i>FALSE</i> : 输入的 I/O 映像区
			<i>TRUE</i> : 输出的 I/O 映像区
<i>ByteAddress</i>	INT	0	开始进行读取的字节起始地址（可能的值：0 ... 63）。
<i>Data</i>	ANY	--	读取数据的目的地址。 要读取的数据长度取决于数据类型和 ANY 指针中的重复因子。最大长度是 64 个字节。

输出参数（状态输出）

参数	数据类型	初始值	说明
<i>DataValid</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 可以读取有效数据 （第一次成功完成读取操作时置位）
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令启动时有错误。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
			<i>FALSE</i> : 命令启动时没有错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。

说明

由于集成工艺中命令执行的运行时间和调用“MC_ReadPeriphery”的 OBx 循环的运行时间，响应时间可能会很长。因此，应将需要快速响应的 I/O 连接到 T-CPU 的 MPI/DP 接口。

MC_ReadPeriphery — 示例

下图中的示例显示了读取或装载 I/O 映像区 DP(DRIVE) 时的时间一致性。

在每个命令执行周期开始时，将从位置调节器中装载输入的信号状态并将其写入输入的 I/O 映像区。此装载操作与“MC_ReadPeriphery”工艺功能无关。“MC_ReadPeriphery”工艺功能访问该 I/O 映像区并向用户程序提供数据。

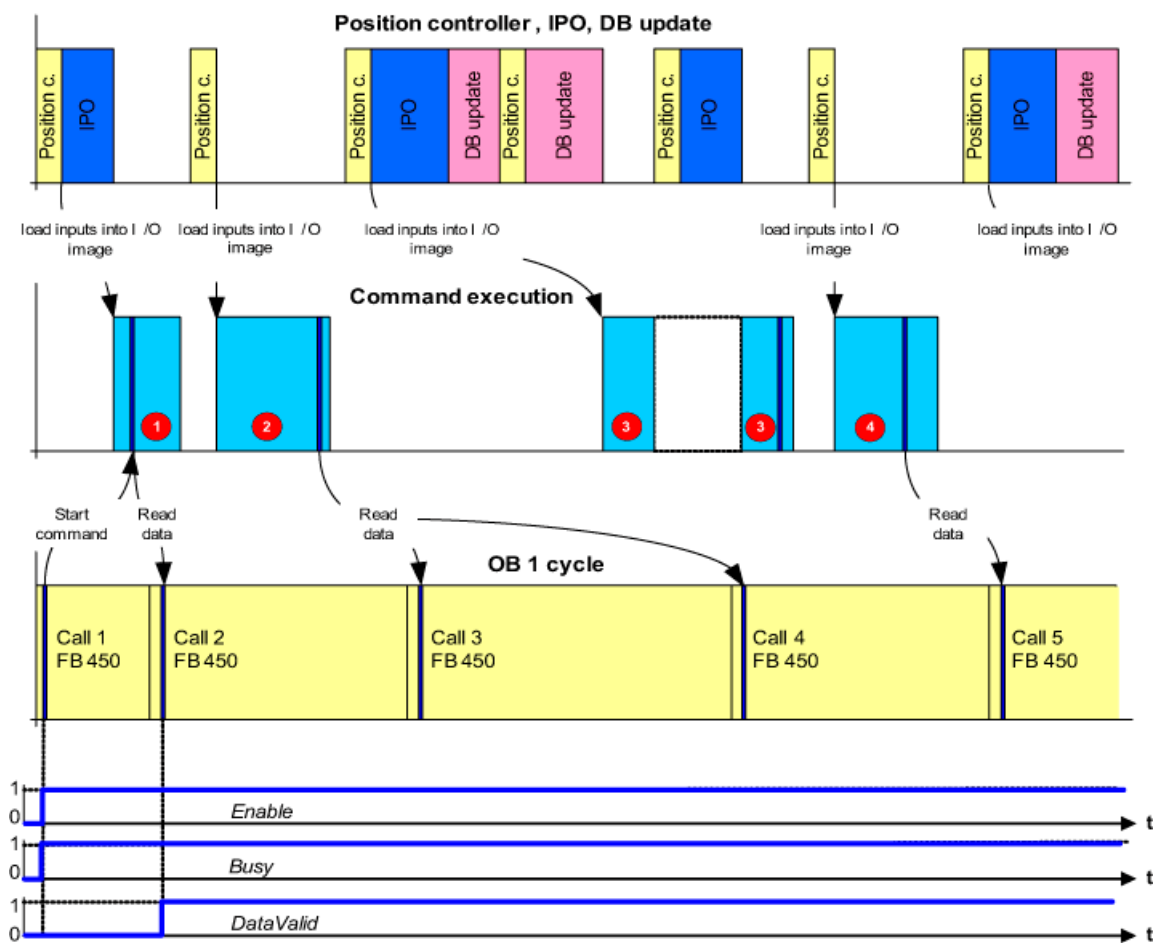
该 I/O 映像区将被传输到控制器，当通过 *Enable = TRUE* 第一次调用 MC_ReadPeriphery 时开始。同时，输出参数 *Busy* 被设置为 *TRUE*。

MC_ReadPeriphery 命令作为第一个命令执行周期时间片中的新命令执行。新命令在执行该命令时开始执行。再次调用 MC_ReadPeriphery 时，I/O 输入映像区被读入控制器输入。输出参数 *DataValid* 报告 *TRUE* 状态。

第二个命令执行周期开始时监视当前的 MC_ReadPeriphery 命令。命令执行结束时处理当前命令，其中最后执行最新的命令。第二个命令执行周期的 I/O 输入映像区在第三次调用 MC_ReadPeriphery 时被传输到控制器。

DB 更新由随后的 IPO 周期触发。此更新直到下一个位置控制周期完成时结束，并由位置控制中断。第三个命令执行周期的时间片在 DB 更新后可用。由于命令执行的延迟开始，该更新由下一个位置控制和随后的 IPO 中断。因此，更新的 I/O 映像区在第四次调用 MC_ReadPeriphery 时不可用。于是，与第三次调用中使用的相同的 I/O 输入映像区被传输到控制器。

第四次调用的 I/O 映像区在第五次调用 MC_ReadPeriphery 时可用，并被传输到控制器。控制器并不读取第三个命令执行周期的 I/O 映像区。



MC_ReadPeriphery — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
800B	无法保留传输缓冲区	命令的传输缓冲区不足。可能的补救措施： - 传输缓冲区已被其它的活动工艺功能占用。防止同时执行以下工艺功能： "MC_ReadCamTrackData" "MC_WriteCamTrackData" "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" - 减小要读取的 I/O 范围的大小。
8043	参数值非法	与输入参数 <i>ByteAddress</i> 有关

ErrorID	错误消息	说明/纠正或避免错误的方法
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是您可能无法通过状态输出对其进行监视。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效。	工艺功能的背景数据块存在故障（例如，长度错误）。
8086	参数不可用	当前无法执行读取/写入命令；重复该命令。
8088	DB 无效（ANY 指针）	ANY 指针指向无效的数据区域。 - 找不到数据块 - 找不到指定的起始地址 - 数据区域过短

ErrorID	错误消息	说明/纠正或避免错误的方法
8089	数据长度无效 (ANY 指针)	指定的数据长度无效。可能的原因： • 长度超过 64 个字节。 • 指定的长度加上起始地址定义的地址在 I/O 映像区 DP(DRIVE) 之外。
808C	指针格式无效或不受支持	ANY 指针包含无效的定义。接受的区域： I、O、M、DB、DI 接受的数据类型： BYTE、CHAR、WORD、INT、DWORD、DINT、REAL
808D	数据记录长度超出范围	已尝试传输长度超过 240 个字节的数据记录。减小 ANY 指针中的长度定义。

6.7.7 FB451 MC_WritePeriphery — 写入工艺 I/O

使用 FB 451“MC_WritePeriphery”写入工艺 I/O

受固件版本为 V3.1.x 或更高版本的集成工艺的支持

用途

- “MC_WritePeriphery”工艺功能用于写入集成工艺的 I/O 映像区。示例中显示了更新的时间从属性。

适用对象

- 处于地址区 0 到 63 的 DP(DRIVE) 上的 DP I/O
- 位于地址区 0 到 63 的 T-CPU 的集成 I/O

先决条件

- 寻址的 I/O 必须存在于集成工艺的 I/O 映像区 DP(DRIVE) 中。

超驰命令

MC_WritePeriphery 命令不能被任何其它命令取消。

MC_WritePeriphery 命令不会取消任何其它命令。

输入参数

参数	数据类型	初始值	说明
<i>Execute</i>	BOOL	<i>FALSE</i>	在上升沿处，开始执行单独的写入命令
<i>ByteAddress</i>	INT	<i>0</i>	开始进行读取的字节起始地址（可能的值： <i>0 ... 63</i> ）
<i>Data</i>	ANY	--	要写入数据的源区域。 要写入数据的长度取决于数据类型和 ANY 指针中的重复因子。最大长度是 64 个字节。
<i>DoneFlag</i>	INT	<i>0</i>	在 MCDevice DB 中生成 DoneFlag

输出参数（状态输出）

参数	数据类型	初始值	说明
<i>Done</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> ：命令已成功完成
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> ：正在执行命令
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> ：命令启动时有错误。未执行该命令。有关原因的信息，请参考 <i>ErrorID</i> 。
			<i>FALSE</i> ：命令启动时没有错误。
<i>ErrorID</i>	WORD	<i>0</i>	输出参数 <i>Error</i> 的 ErrorID。

说明

由于集成工艺中命令执行的运行时间和调用“MC_WritePeriphery”的 OBx 循环的运行时间，响应时间可能会很长。因此，应将需要快速写访问的所有 I/O 连接到 T-CPU 的 MPI/DP 接口。

要周期性写入 I/O 映像区 DP(DRIVE)，请务必在每次写入操作完成后重新触发输入参数 *Execute*。

要执行此操作，请将输入参数“*Execute*”与输出参数 *Busy* 和 *Error* 的取反信号相链接。

MC_WritePeriphery — 示例

下图中的示例显示了写入 I/O 映像区 DP(DRIVE) 时的时间一致性。

在命令执行的每个时间片结束时，输出的 I/O 映像区通过位置控制器写入到 I/O。不管“MC_WritePeriphery”工艺功能的运行状态如何，都会执行该写入操作。

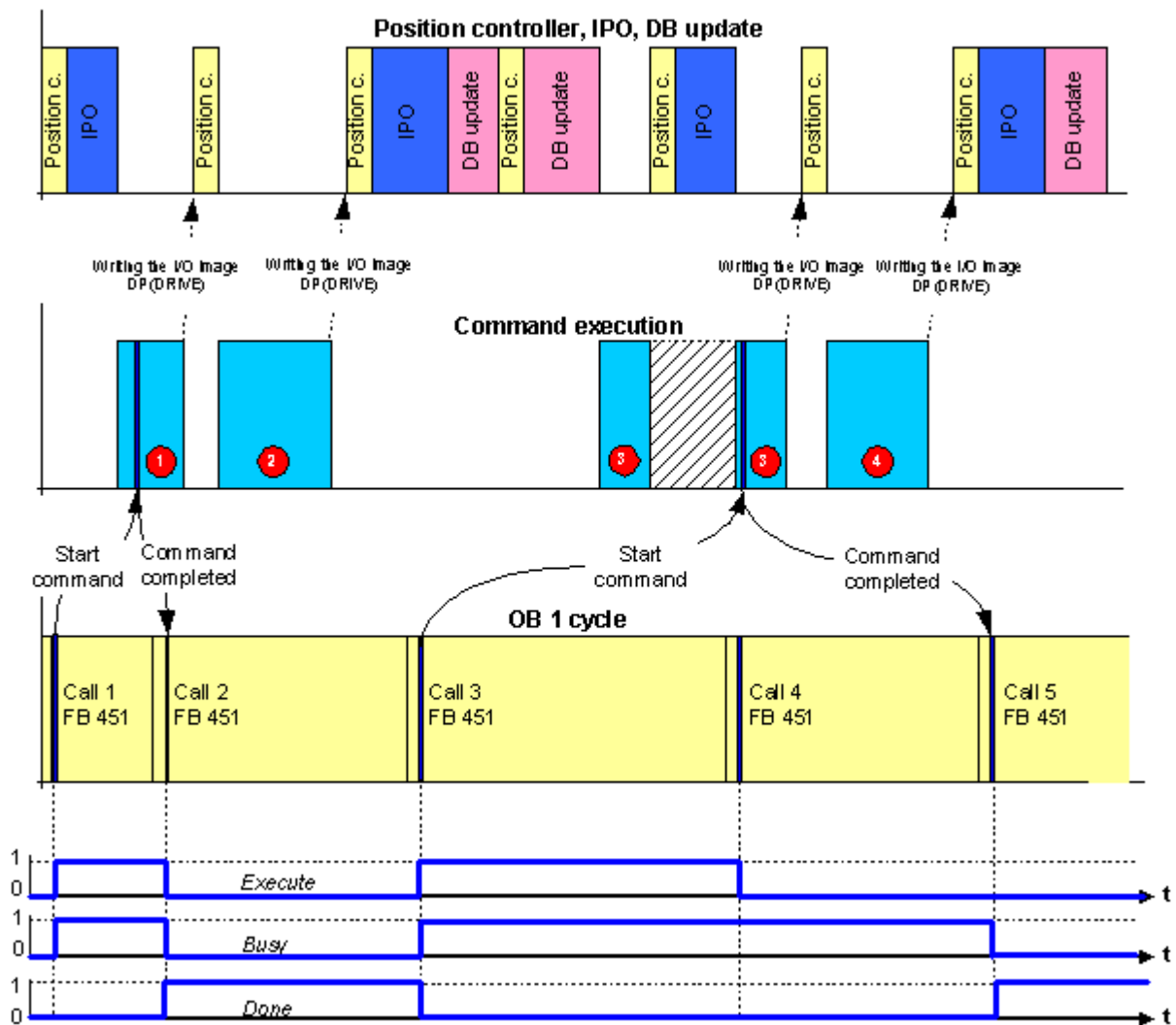
“MC_WritePeriphery”用于将输入参数 *Data* 上指定的控制器数据写入 I/O 输出映像区中。

输入参数 *Execute* 的上升沿触发到 I/O 输出映像区的写入访问，同时，*Busy* 输出参数被设置为 *TRUE*。MC_WritePeriphery 命令作为新命令添加到第一个命令执行周期的时间片。在第一个命令执行时间片内开始执行命令。第二次 MC_WritePeriphery 调用表明该命令已完成（将 *Busy* 设置为 *FALSE* 并将 *Done* 设置为 *TRUE*）。

第二个命令执行时间片结束时，输出的最新写入 I/O 映像区将再次通过位置控制器被写入 I/O。

在下一个位置控制器周期中，DB 更新被触发。DB 更新在位置控制器的时间片内被中断，但在该时间片过后恢复。命令执行因缺少时间间隔而延迟开始。延迟的命令执行周期被下一个位置控制器时间片和随后的 IPO 时间片中断。

第三次调用 MC_WritePeriphery 加上输入参数 *Execute* 的上升沿启动一个新写入命令。由于后面的命令已开始，因此新的写入命令排列在命令执行时间片的后半部分。因此，命令的完成不会在第四次调用 MC_WritePeriphery 时立即报告，而是在第五次调用 MC_WritePeriphery 时报告（现在，*Busy* 更改为 *FALSE* 且 *Done* 更改为 *TRUE*）。



MC_WritePeriphery — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
800B	无法保留传输缓冲区	命令的传输缓冲区不足。可能的补救措施： - 传输缓冲区已被其它的活动工艺功能占用。防止同时执行以下工艺功能： "MC_ReadCamTrackData" "MC_WriteCamTrackData" "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" - 减小要写入的 I/O 范围的大小。
8043	参数值非法	涉及输入参数 <i>ByteAddress</i> 或 <i>DoneFlag</i> 。

ErrorID	错误消息	说明/纠正或避免错误的方法
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是您可能无法通过状态输出对其进行监视。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8086	参数不可用	当前无法执行读取/写入命令；重复该命令。
8088	DB 无效（ANY 指针）	ANY 指针指向无效的数据区域。 - 找不到数据块 - 找不到指定的起始地址 - 数据区域过短

ErrorID	错误消息	说明/纠正或避免错误的方法
8089	数据长度无效 (ANY 指针)	指定的数据长度无效。可能的原因： - 长度超出 64 个字节 - 指定的长度加上起始地址定义的地址在 I/O 映像区 DP(DRIVE) 之外。
808C	指针格式无效或不受支持	ANY 指针包含无效的定义。接受的区域： I、O、M、DB、DI 接受的数据类型： BYTE、CHAR、WORD、INT、DWORD、DINT、REAL
808D	数据记录长度超出范围	已尝试传输长度超过 240 个字节的数据记录。减小 ANY 指针中的长度定义。

6.7.8 FB453 MC_ReadRecord — 读取数据记录

使用 FB 453“MC_ReadRecord”读取数据记录

用途

- “MC_ReadRecord”工艺功能可用于从 DP(DRIVE). 上的 I/O 模块中读取数据记录。
- 该工艺功能可与用于读取 S7 CPU 的 DP 接口的数据记录的系统功能 SFC59“RD_REC”相比较。

适用对象

- DP(DRIVE) 上的 DP I/O

先决条件

- I/O 必须支持数据记录通信。
- I/O 模块必须支持相关数据记录。

超驰命令

MC_ReadRecord 命令不能被任何其它命令取消。

MC_ReadRecord 命令不会取消任何其它命令。

输入参数

参数	数据类型	初始值	说明
<i>Execute</i>	BOOL	<i>FALSE</i>	在上升沿处启动命令
<i>InOut</i>	BOOL	<i>FALSE</i>	I/O 模块的逻辑基址的 I/O 分配
			<i>FALSE</i> : 输入地址
			<i>TRUE</i> : 输出地址
<i>Address</i>	INT	0	I/O 模块的逻辑基址。
<i>RecordNumber</i>	INT	0	I/O 模块的数据记录数（可能的值 0 到 255）
<i>Data</i>	ANY	-	读取数据的目的地址。 要读取的数据的最大长度取决于数据类型和 ANY 指针的重复因子。
<i>DoneFlag</i>	INT	0	在 MCDevice DB 中生成 DoneFlag

输出参数（状态输出）

参数	数据类型	初始值	说明
<i>Done</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令已成功完成
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 执行命令时有错误。命令执行失败；有关原因的信息，请参考 <i>ErrorID</i> 。
			<i>FALSE</i> : 执行命令时无错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 ErrorID。
<i>DataLength</i>	INT	0	读取的数据记录信息的长度（以字节为单位）

MC_ReadRecord — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正由其它过程使用。	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。
800B	无法保留传输缓冲区	命令的传输缓冲区不足。可能的补救措施： - 传输缓冲区已被其它的活动工艺功能占用。防止同时执行以下工艺功能： "MC_ReadCamTrackData" "MC_WriteCamTrackData" "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" - 减小要读取的数据区域的大小。
8043	参数值非法	涉及输入参数 <i>RecordNumber</i> 或 <i>DoneFlag</i> 。

ErrorID	错误消息	说明/纠正或避免错误的方法
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是您可能无法通过状态输出对其进行监视。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8088	DB 无效（ANY 指针）	ANY 指针指向无效的数据区域。 - 找不到数据块 - 找不到指定的起始地址 - 数据区域过短
8089	数据长度无效（ANY 指针）	在 ANY 指针上指定的数据长度短于数据记录长度。

ErrorID	错误消息	说明/纠正或避免错误的方法
808C	指针格式无效或不受支持	ANY 指针包含无效的定义。接受的区域： I、O、M、DB、DI 接受的数据类型： BYTE、CHAR、WORD、INT、DWORD、DINT、REAL
808D	数据记录长度超出范围	已尝试传输长度超过 240 个字节的数据记录。减小 ANY 指针中的长度定义。
8090	逻辑基址无效	- 数据集传送时出错 - 作业中止 - 指定的逻辑基址无效： SDB1/SDB2x 中不存在分配，或分配不是基址。
8091	逻辑基址不可用	数据记录传输过程中出错，命令被取消。 该功能无法访问指定的逻辑基址。 尝试与 T-CPU 的集成 I/O 建立数据记录通信时出错。
8092	故障响应 ID	数据记录传输过程中出错，命令被取消。响应符号中存在错误。
80A0	读取模块数据时出错	数据记录传输过程中出错，命令被取消。读取模块数据时进行否定确认： - 读取操作期间模块被删除 - 故障模块
80A2	第 2 层出现 DP 协议错误	数据记录传输过程中出错，可以在下一个程序周期中重试。 第 2 层出现 PROFIBUS DP 错误： - 站故障 - 超时 - 协议错误 - 总线错误
80A3	用户界面中出现 DP 协议错误	数据记录传输过程中出错，命令被取消。用户界面/用户中出现 PROFIBUS DP 协议错误。
80B0	不支持数据记录	数据记录传输过程中出错，命令被取消。 - 该模块类型不支持系统功能。 - 模块不识别数据记录 - 数据记录数量超出范围 0 到 255

ErrorID	错误消息	说明/纠正或避免错误的方法
80B1	长度定义错误	数据记录传输过程中出错，命令被取消。输入参数 Data 中的长度定义错误。示例： - 数据记录读取了 240 个字节，但是 ANY 指针仅指向长度为 80 个字节的字段。 ANY 指针中的长度定义（例如 260 字节）不受支持。
80C0	无读取的数据	数据记录传输过程中出错，可以在下一个程序周期中重试。该模块将维持数据记录，但是读取数据仍不存在。
80C2	达到最大命令数	数据记录传输时出错。可能进行了即时命令重复。模块当前执行的 CPU 命令数达到了容许的最大数量。
80C3	模块处于容量限制	数据记录传输时出错。可能进行了即时命令重复。所需资源当前正在被另一个应用程序使用： 在模块中
80C4	通信错误	数据记录传输时出错。可能进行了即时命令重复。通信错误： - 奇偶校验错误 - 未设置 SW 准备 - 块长度管理过程中出错 CPU 一侧的校验和错误 - 模块一侧的校验和错误
80C5	访问分布式 I/O 失败	数据记录传输过程中出错，可以在下一个程序周期中重试。分布式 I/O 当前不可用。
80C6	优先级错误	数据记录传输过程中出错，可以在下一个程序周期中重试。 由于优先级被取消（重启或后台），因此数据记录传输被取消。

说明

DP 从站可以根据 PROFIBUS DP V1 报告错误 **A0** 到 **CF**。错误 **A0** 到 **CF** 被作为错误 **80A0** 到 **80CF** 映射到工艺功能的 **ErrorID**。

如果输出参数 **ErrorID** 返回的 **80A0** 到 **80CF** 范围内的错误未上面的列表中定义，则请查阅 DP 从站文档。

6.7.9 FB454 MC_WriteRecord — 写入数据记录

使用 FB 454“MC_WriteRecord”写入数据记录

用途

- “MC_WriteRecord”工艺功能可用于将数据记录写入 DP(DRIVE) 上的 I/O 模块。
- 该工艺功能可与用于写入 S7 CPU 的 DP 接口的数据记录的系统功能 SFC58“WR_REC”相比。

适用对象

- DP(DRIVE) 上的 DP I/O

先决条件

- I/O 必须支持数据记录通信。
- I/O 模块必须支持相关数据记录。

超驰命令

MC_WriteRecord 命令不能被任何其它命令取消。

MC_WriteRecord 命令不会取消任何其它命令。

输入参数

参数	数据类型	初始值	说明
<i>Execute</i>	BOOL	<i>FALSE</i>	在上升沿处启动命令
<i>InOut</i>	BOOL	<i>FALSE</i>	I/O 模块的逻辑基址的 I/O 分配
			<i>FALSE:</i> 输入地址
			<i>TRUE:</i> 输出地址
<i>Address</i>	INT	<i>0</i>	I/O 模块的逻辑基址。
<i>RecordNumber</i>	INT	<i>0</i>	I/O 模块的数据记录数（可能的值 <i>0</i> 到 <i>255</i> ）
<i>Data</i>	ANY	-	要写入数据的源区域。 要写入数据的长度取决于数据类型和 ANY 指针中的重复因子。
<i>DoneFlag</i>	INT	<i>0</i>	在 MCDevice DB 中生成 DoneFlag

输出参数（状态输出）

参数	数据类型	初始值	说明
<i>Done</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令已成功完成
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 执行命令时有错误。命令执行失败；有关原因的信息，请参考 <i>ErrorID</i> 。
			<i>FALSE</i> : 执行命令时无错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。

MC_WriteRecord — ErrorID

受固件版本为 V3.1.x 或更高版本的集成工艺的支持

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。

ErrorID	错误消息	说明/纠正或避免错误的方法
800B	无法保留传输缓冲区	命令的传输缓冲区不足。可能的补救措施： - 传输缓冲区已被其它的活动工艺功能占用。防止同时执行以下工艺功能： “MC_ReadCamTrackData” “MC_WriteCamTrackData” “MC_CamSectorAdd” “MC_ReadPeriphery” “MC_WritePeriphery” “MC_ReadRecord” “MC_WriteRecord” “MC_ReadDriveParameter” “MC_WriteDriveParameter” - 减小要写入的数据区域的大小。
8043	参数值非法	涉及输入参数 <i>RecordNumber</i> 或 <i>DoneFlag</i> 。
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是您可能无法通过状态输出对其进行监视。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。

ErrorID	错误消息	说明/纠正或避免错误的方法
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： • 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 • 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8088	DB 无效（ANY 指针）	ANY 指针指向无效的数据区域。 • 找不到数据块 • 找不到指定的起始地址 • 数据区域过短
8089	数据长度无效（ANY 指针）	在 ANY 指针上指定的数据长度短于数据记录长度。
808C	指针格式无效或不受支持	ANY 指针包含无效的定义。接受的区域： I、O、M、DB、DI 接受的数据类型： BYTE、CHAR、WORD、INT、DWORD、DINT、REAL
808D	数据记录长度超出范围	已尝试传输长度超过 240 个字节的数据记录。修正 ANY 指针的长度定义。
8090	逻辑基址无效	数据记录传输过程中出错，命令被取消。指定的逻辑基址无效：SDB1/SDB2x 中不存在分配，或分配不是基址。

ErrorID	错误消息	说明/纠正或避免错误的方法
8091	逻辑基址不可用	数据记录传输过程中出错，命令被取消。 该功能无法访问指定的逻辑基址。 尝试与 T-CPU 的集成 I/O 建立数据记录通信时出错。
8092	故障响应 ID	数据记录传输过程中出错，命令被取消。响应符号中存在错误。
80A1	向模块写入时出错	数据记录传输过程中出错，命令被取消。向模块写入时进行否定确认： • 执行写入操作期间模块被删除 • 故障模块
80A2	第 2 层出现 DP 协议错误	数据记录传输过程中出错，可以在下一个程序周期中重试。 第 2 层出现 PROFIBUS DP 错误： • 站故障 • 超时 • 协议错误 • 总线错误
80A3	用户界面中出现 DP 协议错误	数据记录传输过程中出错，命令被取消。用户界面/用户中出现 PROFIBUS DP 协议错误。
80B0	不支持数据记录	数据记录传输过程中出错，命令被取消。 • 该模块类型不支持系统功能。 • 模块不识别数据记录 • 数据记录数量超出范围 0 到 255
80B1	长度定义错误	数据记录传输过程中出错，命令被取消。输入参数 <i>Data</i> 中的长度定义错误。 • 数据记录读取了 240 个字节，但是 ANY 指针仅指向长度为 80 个字节的字段。 • ANY 指针中的长度定义（例如 260 字节）不受支持。
80C0	前一写入命令仍在运行	数据记录传输时出错。可能进行了即时命令重复。模块还未处理同一数据记录的前一写入命令的数据。
80C2	达到最大命令数	数据记录传输时出错。可能进行了即时命令重复。模块当前执行的 CPU 命令数达到了容许的最大数量。

ErrorID	错误消息	说明/纠正或避免错误的方法
80C3	模块处于容量限制	数据记录传输时出错。可能进行了即时命令重复。所需资源当前正在被另一个应用程序使用： 在模块中
80C4	通信错误	数据记录传输时出错。可能进行了即时命令重复。通信错误： - 奇偶校验错误 - 未设置 SW 准备 - 块长度管理过程中出错 - CPU 一侧的校验和错误 - 模块一侧的校验和错误
80C5	访问分布式 I/O 失败	数据记录传输过程中出错，可以在下一个程序周期中重试。分布式 I/O 当前不可用。
80C6	优先级错误	数据记录传输过程中出错，可以在下一个程序周期中重试。 由于优先级被取消（重启或后台），因此数据记录传输被取消。

说明

DP 从站可以根据 PROFIBUS DP V1 报告错误 *A0* 到 *CF*。错误 *A0* 到 *CF* 被作为错误 *80A0* 到 *80CF* 映射到工艺功能的 *ErrorID*。

如果输出参数 *ErrorID* 返回的 *80A0* 到 *80CF* 范围内的错误未上面的列表中定义，则请查阅 DP 从站文档。

6.7.10 FB455 MC_ReadDriveParameter — 读取驱动参数

使用 FB 455“MC_ReadDriveParameter”读取驱动参数

用途

- “MC_ReadDriveParameter” 工艺功能允许用户程序读取连接到 DP(DRIVE) 的驱动参数。

适用对象

- DP(DRIVE) 上的符合 PROFIdrive 驱动

先决条件

- 组态该驱动器以在 DP(DRIVE) 上操作。
- 该驱动器必须支持数据记录通信。
- 该驱动器必须支持参数编号和索引。

超驰命令

MC_ReadDriveParameter 命令不能被任何其它命令取消。

MC_ReadDriveParameter 命令不会取消任何其它命令。

输入参数

参数	数据类型	初始值	说明
<i>Execute</i>	BOOL	<i>FALSE</i>	在上升沿处启动命令
<i>InOut</i>	BOOL	<i>FALSE</i>	驱动的逻辑基址的 I/O 分配
			<i>FALSE</i> : 输入地址
			<i>TRUE</i> : 输出地址
<i>Address</i>	INT	0	模块的逻辑基址的定义
<i>ParameterNumber</i>	DINT	0	指定从中读取数据的参数号 — 取值范围: 0 ... 65535
<i>SubIndex</i>	DINT	0	指定要读取的第一个参数索引 (该值在内部被设置为零, 即 <i>NumberOfElements</i> =0) — 范围: 0 ... 65535
<i>NumberOfElements</i>	DINT	0	要读取的子参数的数量:
			值 > 0: 具有子索引的参数, 范围 0 到 234 (取决于输入参数 <i>Data</i> 中的数据区域的最大长度)
			值 = 0: 无子索引的参数
			值 < 0: 不允许
<i>Data</i>	ANY	-	应将参数值保存到其中的目标地址区。 要读取的数据的最大长度取决于数据类型和 ANY 指针的重复因子。 该数据区域最大为 240 个字节。
<i>DoneFlag</i>	INT	0	在 MCDevice DB 中生成 DoneFlag

输出参数（状态输出）

参数	数据类型	初始值	说明
<i>Done</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 命令已成功完成
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 正在执行命令
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> : 执行命令时有错误。命令执行失败；有关原因的信息，请参考 <i>ErrorID</i> 。
			<i>FALSE</i> : 执行命令时无错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。
<i>DataLength</i>	INT	0	读取的数据记录信息的长度（以字节为单位）

MC_ReadDriveParameter — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。

ErrorID	错误消息	说明/纠正或避免错误的方法
800B	无法保留传输缓冲区	命令的传输缓冲区不足。可能的补救措施： - 传输缓冲区已被其它的活动工艺功能占用。防止同时执行以下工艺功能： “MC_ReadCamTrackData” “MC_WriteCamTrackData” “MC_CamSectorAdd” “MC_ReadPeriphery” “MC_WritePeriphery” “MC_ReadRecord” “MC_WriteRecord” “MC_ReadDriveParameter” “MC_WriteDriveParameter” - 减少要读取的子参数的数量。
8043	参数值非法	与输入参数 <i>ParameterNumber</i> , <i>SubIndex</i> , <i>NumberOfElements</i> 或 <i>DoneFlag</i> 有关。
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是您可能无法通过状态输出对其进行监视。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。

ErrorID	错误消息	说明/纠正或避免错误的方法
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： - 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 - 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8088	DB 无效（ANY 指针）	ANY 指针指向无效的数据区域。 - 找不到数据块 - 找不到指定的起始地址 - 数据区域过短
8089	数据长度无效（ANY 指针）	在 ANY 指针上指定的数据长度短于数据记录长度。 数据记录的长度取决于要读取的参数值的数量和类型。
808C	指针格式无效或不受支持	ANY 指针包含无效的定义。接受的区域： I、O、M、DB、DI 接受的数据类型： BYTE、CHAR、WORD、INT、DWORD、DINT、REAL
808D	数据记录长度超出范围	已尝试传输长度超过 240 个字节的数据记录。减小 ANY 指针中的长度定义。
8090	逻辑基址无效	数据记录传输过程中出错，命令被取消。指定的逻辑基址无效：SDB1/SDB2x 中不存在分配，或分配不是基址。

ErrorID	错误消息	说明/纠正或避免错误的方法
8091	逻辑基址不可用	数据记录传输过程中出错，命令被取消。 该功能无法访问指定的逻辑基址。 尝试与 T-CPU 的集成 I/O 建立数据记录通信时出错。
8092	故障响应 ID	数据记录传输过程中出错，命令被取消。响应符号中存在错误。
80A0	读取模块数据时出错	数据记录传输过程中出错，命令被取消。读取模块数据时进行否定确认： • 读取操作期间模块被删除 • 故障模块
80A2	第 2 层出现 DP 协议错误	数据记录传输过程中出错，可以在下一个程序周期中重试。 第 2 层出现 PROFIBUS DP 错误： • 站故障 • 超时 • 协议错误 • 总线错误
80B0	不支持数据记录	数据记录传输过程中出错，命令被取消。 • 该模块类型不支持系统功能。 • 模块不识别数据记录 • 数据记录数量超出范围 0 到 255
80B5	无法执行系统功能	数据记录传输过程中出错，可以在下一个程序周期中重试。驱动的内部处理状态导致无法执行系统功能。
80B7	故障命令	数据记录传输过程中出错，PROFIBUS DP 出错，命令被取消。无法输出故障命令。
80C3	模块处于容量限制	数据记录传输过程中出错，可以在下一个程序周期中重试。所需的资源当前正在被其它应用程序使用： • 在工艺功能“MC_ReadDriveParameter”中 • 在模块中
80C5	访问分布式 I/O 失败	数据记录传输过程中出错，可以在下一个程序周期中重试。分布式 I/O 当前不可用。
80C7	已将一个命令输出到驱动	数据记录传输过程中出错，可以在下一个程序周期中重试。已将另一个命令输出到驱动

ErrorID	错误消息	说明/纠正或避免错误的方法
8100	找不到参数	参数错误，命令被取消。访问的参数不存在。
8103	找不到子索引	参数错误，命令被取消。访问的子索引不存在。
8104	参数未编入索引	参数错误，命令被取消。使用子索引访问未编入索引的参数。
8111	当前状态拒绝访问	参数错误，命令被取消。操作状态导致无法执行请求。
8115	超出了最大可传输长度	参数错误，命令被取消。使用单个命令无法传输所请求数量的子参数。 请注意，参数数据类型的字节数乘以 <i>NumberOfElements</i> 不能超过 234。
8116	值无效或不受支持	参数错误，命令被取消。属性值、元素数量、参数编号、子索引或上述几项的组合无效或不受支持。
8119	轴不存在	参数错误，命令被取消。访问的轴不存在。
8120 ... 8164	备用	PROFIdrive 规范预留的错误代码
8165 ... 81FF	制造商特定的错误	符合 PROFIdrive 规范的制造商特定的错误代码。错误由驱动形成。

说明

DP 从站可以根据 PROFIBUS DP V1 报告错误 *A0* 到 *CF*。错误 *A0* 到 *CF* 被作为错误 *80A0* 到 *80CF* 映射到工艺功能的 *ErrorID*。

如果输出参数 *ErrorID* 返回的 *80A0* 到 *80CF* 范围内的错误未上面的列表中定义，则请查阅 DP 从站文档。

6.7.11 FB456 MC_WriteDriveParameter — 写入驱动参数

使用 FB 456“MC_WriteDriveParameter”写入驱动参数

用途

- “MC_WriteDriveParameter”工艺功能允许用户程序写入连接到 DP(DRIVE) 的驱动参数。

适用对象

- DP(DRIVE) 上的符合 PROFIdrive 驱动

先决条件

- 组态该驱动器以在 DP(DRIVE) 上操作。
- 该驱动器必须支持数据记录通信。
- 该驱动器支持参数编号和索引。

超驰命令

MC_WriteDriveParameter- 命令不能被任何其它命令取消。

MC_WriteDriveParameter- 命令不会取消任何其它命令。

输入参数

参数	数据类型	初始值	说明
<i>Execute</i>	BOOL	<i>FALSE</i>	在上升沿处启动命令
<i>InOut</i>	BOOL	0	驱动的逻辑基址的 I/O 分配
			<i>FALSE</i> : 输入地址
			<i>TRUE</i> : 输出地址
<i>Address</i>	INT	0	模块的逻辑基址的定义
<i>ParameterNumber</i>	DINT	0	数据应写入其中的参数编号的定义。 值范围: 0 ... 65535
<i>SubIndex</i>	DINT	0	指定要写入的第一个参数索引（该值在内部被设置为零，即 <i>NumberOfElements</i> = 0）— 范围: 0 ... 65535
<i>NumberOfElements</i>	DINT	0	要写入的子参数的数量:
			值 > 0: 具有子索引的参数, 范围 1 到 228 (取决于输入参数 <i>Data</i> 中的数据区域的最大长度)
			值 = 0: 无子索引的参数
			值 < 0: 不允许
<i>Data</i>	ANY	--	包含参数值的数据区域。 数据长度取决于参数数据类型和要写入的参数数量。 数据类型和 ANY 指针的重复因子的组合一定会产生相同的数据长度。 该数据区最大为 228 个字节。
<i>DoneFlag</i>	INT	0	在 MCDevice-DB 中生成 DoneFlag

输出参数（状态输出）

参数	数据类型	初始值	说明
<i>Done</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> ：命令已成功完成
<i>Busy</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> ：正在执行命令
<i>Error</i>	BOOL	<i>FALSE</i>	<i>TRUE</i> ：执行命令时有错误。命令执行失败；有关原因的信息，请参考 <i>ErrorID</i> 。
			<i>FALSE</i> ：执行命令时无错误。
<i>ErrorID</i>	WORD	0	输出参数 <i>Error</i> 的 <i>ErrorID</i> 。

MC_WriteDriveParameter — ErrorID

ErrorID	错误消息	说明/纠正或避免错误的方法
0000	没有错误	-
8001	内部错误	项目/软件发生故障或不一致。
8005	命令已取消，因为命令存储器正被其它进程使用	命令容量不足，无法执行该命令。 可能的原因： - 活动命令的数量已超出限制。 - 以下工艺功能中的活动命令过多： "MC_CamSectorAdd" "MC_ReadPeriphery" "MC_WritePeriphery" "MC_ReadRecord" "MC_WriteRecord" "MC_ReadDriveParameter" "MC_WriteDriveParameter" "MC_ReadCamTrackData" "MC_WriteCamTrackData" 在同一周期内调用这些工艺功能，直到输出参数 <i>Done</i>、<i>CommandAborted</i> 或 <i>Error</i> 中有一个变为 <i>TRUE</i>。验证程序是否包含不必要的（多余的）命令。

ErrorID	错误消息	说明/纠正或避免错误的方法
800B	无法保留传输缓冲区	命令的传输缓冲区不足。可能的补救措施： - 传输缓冲区已被其它的活动工艺功能占用。防止同时执行以下工艺功能： “MC_ReadCamTrackData” “MC_WriteCamTrackData” “MC_CamSectorAdd” “MC_ReadPeriphery” “MC_WritePeriphery” “MC_ReadRecord” “MC_WriteRecord” “MC_ReadDriveParameter” “MC_WriteDriveParameter” - 减少要写入的子参数的数量。
8043	参数值非法	与输入参数 <i>ParameterNumber</i> 、 <i>SubIndex</i> 、 <i>NumberOfElements</i> 或 <i>DoneFlag</i> 有关
804C	命令输出率过高	具有相同背景数据块的命令的输出率超过了命令接口的容量。第二个命令被拒绝，以保持第一个命令的一致性。 如果命令输出率较高，请务必使用单独的背景数据块或再次请求命令。请注意，尽管接受的第一个命令可能处于活动状态，但是您可能无法通过状态输出对其进行监视。
8050	工艺未准备就绪	- 在使用断点进行测试期间（CPU 处于 HOLD 状态），工艺功能向工艺输出了一条新命令。该命令被拒绝。 - 在重启 OB 中输出了该命令。

ErrorID	错误消息	说明/纠正或避免错误的方法
8052	在不同运行级别进行块调用	使用同一背景数据块，在不同运行级别调用了此工艺功能。 示例： 在 OB 1 和 OB 35 中都使用背景数据块 DBx 调用工艺功能 x。开始工艺功能在 OB 1 中执行，然后由于在 OB 35 中调用它而中断其在 OB 1 中的执行。由于背景数据块是共享使用的，因此将在这两个调用的输出参数 <i>ErrorID</i> 中都指示该错误。 预期的错误响应： • 新命令（输入参数 <i>Execute / Enable</i> 的上升沿或下降沿）未传送到集成工艺。 • 最初启动的命令不能在工艺功能的输出参数中进行监视。但是，该命令在集成工艺中可能仍处于活动状态 注意： 在不同的运行级别使用不同的背景数据块，或者互锁工艺功能的调用。
8053	背景数据块无效	工艺功能的背景数据块存在故障（例如，长度错误）。
8088	DB 无效（ANY 指针）	ANY 指针指向无效的数据区域。 • 找不到数据块 • 找不到指定的起始地址 • 数据区域过短
8089	数据长度无效（ANY 指针）	在 ANY 指针上指定的数据长度短于数据记录长度。 数据记录的长度取决于要写入的参数值的数量和类型。
808C	指针格式无效或不受支持	ANY 指针包含无效的定义。接受的区域： I、O、M、DB、DI 接受的数据类型： BYTE、CHAR、WORD、INT、DWORD、DINT、REAL
808D	数据记录长度超出范围	已尝试传输长度超过 228 个字节的数据记录。减小 ANY 指针中的长度定义。
808E	数据格式无效	数据记录传输过程中出错，命令被取消。找不到有效的数据格式。
8090	逻辑基址无效	数据记录传输过程中出错，命令被取消。指定的逻辑基址无效：SDB1/SDB2x 中不存在分配，或分配不是基址。

ErrorID	错误消息	说明/纠正或避免错误的方法
8091	逻辑基址不可用	数据记录传输过程中出错，命令被取消。 该功能无法访问指定的逻辑基址。 尝试与 T-CPU 的集成 I/O 建立数据记录通信时出错。
8092	故障响应 ID	数据记录传输过程中出错，命令被取消。响应符号中存在错误。
80A1	向模块写入时出错	数据记录传输过程中出错，命令被取消。向模块写入时进行否定确认： • 执行写入操作期间模块被删除 • 故障模块
80A2	第 2 层出现 DP 协议错误	数据记录传输过程中出错，可以在下一个程序周期中重试。 第 2 层出现 PROFIBUS DP 错误： • 站故障 • 超时 • 协议错误 • 总线错误
80B0	不支持数据记录	数据记录传输过程中出错，命令被取消。 • 该模块类型不支持系统功能。 • 模块不识别数据记录 • 数据记录数量超出范围 0 到 255
80B5	无法执行系统功能	数据记录传输过程中出错，可以在下一个程序周期中重试。驱动的内部处理状态导致无法执行系统功能。
80B7	故障命令	PROFIBUS DP 错误，命令被取消。无法输出故障命令。
80C3	模块处于容量限制	数据记录传输过程中出错，可以在下一个程序周期中重试。所需的资源当前正在被其它应用程序使用： • 在工艺功能 "MC_WriteDriveParameter" 中 • 在模块中
80C5	访问分布式 I/O 失败	数据记录传输过程中出错，可以在下一个程序周期中重试。分布式 I/O 当前不可用。
80C7	已将一个命令输出到驱动	数据记录传输过程中出错，可以在下一个程序周期中重试。已将另一个命令输出到驱动
8100	找不到参数	参数错误，命令被取消。访问的参数不存在。

ErrorID	错误消息	说明/纠正或避免错误的方法
8101	无法更改参数	参数错误，命令被取消。尝试修改只读参数。
8102	超出了参数限制。	参数错误，命令被取消。对值限制之外的值进行修改访问。
8103	找不到子索引	参数错误，命令被取消。访问的子索引不存在。
8104	参数未编入索引	参数错误，命令被取消。使用子索引访问未编入索引的参数。
8105	数据类型无效	参数错误，命令被取消。试图将当前值修改为新值，但参数数据类型不支持该新值。
8106	参数值不等于 0	参数错误，命令被取消。尝试修改不等于 0 的值。
810B	无专有访问权限	参数错误，命令被取消。尝试在没有专有访问权限的情况下进行修改。
8111	当前状态拒绝访问	参数错误，命令被取消 操作状态导致无法执行请求
8114	参数值非法	参数错误，命令被取消。尝试修改可能未超出值限制，但却因永久兼容性（具有定义的单独值的现有参数）而不允许使用的值。
8116	值无效或不受支持	参数错误，命令被取消。属性值、元素数量、参数编号、子索引或上述几项的组合无效或不受支持。
8117	格式无效	参数错误，命令被取消。格式无效
8118	参数数据数量无效	参数错误，命令被取消。参数数据中的值数量与参数地址中的地址数量不匹配。
8119	轴不存在	参数错误，命令被取消。访问的轴不存在。
8120 ... 8164	备用	PROFIdrive 规范预留的错误代码
8165 ... 81FF	制造商特定的错误	符合 PROFIdrive 规范的制造商特定的错误代码 错误由驱动形成。

说明

DP 从站可以根据 PROFIBUS DP V1 报告错误 *A0* 到 *CF*。错误 *A0* 到 *CF* 被作为错误 *80A0* 到 *80CF* 映射到工艺功能的 *ErrorID*。

如果输出参数 *ErrorID* 返回的 *80A0* 到 *80CF* 范围内的错误未上面的列表中定义，则请查阅 DP 从站文档。

6.8 命令的相互作用

6.8.1 新命令 — 活动的单个命令 (1)

下表显示了新命令对活动的单个命令的相应作用。

		busy single command → new command ↓																							
		d	w	d	w	d	d	d	w	d	w	d	d	MC_Home (Mode=0, 1)	MC_Home (Mode=2-7), Achse freigegeben	MC_Halt	MC_Stop	MC_MoveAbsolute	MC_MoveAdditive	MC_MoveRelative	MC_MoveVelocity	MC_MoveToEndPos (pClamping=True)	MC_MoveToEndPos (pClamping=False)	MC_MoveSuperImposed	
MC_CamIn	w	Wa	A1	Wa	A1	Wa	Wa	Wa	A1	Wa	A1	Wa	Wa	Wa	S	Wa	E	Wa	Wa	Wa	Wa	Ad	Wa	Wa	
MC_CamIn	d	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	S	A1	E	A1	A1	A1	A1	Ad	A1	A1	
MC_GearIn	w	Wa	A1	Wa	A1	Wa	Wa	Wa	A1	Wa	A1	Wa	Wa	Wa	S	Wa	E	Wa	Wa	Wa	Wa	Ad	Wa	Wa	
MC_GearIn	d	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	S	A1	E	A1	A1	A1	A1	Ad	A1	A1	
MC_CamOut	w	Wa	Wa	Ae	Ae	Ae	Ae	N	N	N	N	N	N	N	N	N	E	N	N	N	N	N	N	N	
MC_CamOut	d	A1	A1	Ae	Ae	Ae	Ae	N	N	N	N	N	N	N	N	N	E	N	N	N	N	N	N	N	
MC_GearOut	w	Ae	Ae	Wa	Wa	Ae	Ae	N	N	N	N	N	N	N	N	N	E	N	N	N	N	N	N	N	
MC_GearOut	d	Ae	Ae	A1	A1	Ae	Ae	N	N	N	N	N	N	N	N	N	E	N	N	N	N	N	N	N	
MC_Phasing		S	E	S	E	Ae	Ae	Ae	Ae	Ae	Ae	Ae	Ae	E	E	E	E	E	E	E	E	E	E	E	
MC_CamInSuperImposed	w	Sw	Sa	Sw	Sa	S	S	Wa	A1	Wa	A1	Wa	Wa	E	E	Sw	E	Sw	Sw	Sw	Sw	Ad	Sw	Sw	
MC_CamInSuperImposed	d	S	Sa	S	Sa	S	S	A1	A1	A1	A1	A1	A1	E	E	S	E	S	S	S	S	Ad	S	A1	
MC_GearInSuperImposed	w	Sw	Sa	Sw	Sa	S	S	Wa	A1	Wa	A1	Wa	Wa	E	E	W	E	Sw	Sw	Sw	Sw	Ad	Sw	Sw	
MC_GearInSuperImposed	d	S	Sa	S	Sa	S	S	A1	A1	A1	A1	A1	A1	E	E	S	E	S	S	S	S	Ad	S	A1	
MC_CamOutSuperImposed	w	N	N	N	N	N	N	Wa	Wa	Ae	Ae	Ae	Ae	N	N	N	E	N	N	N	N	N	N	N	
MC_CamOutSuperImposed	d	N	N	N	N	N	N	A1	A1	Ae	Ae	Ae	Ae	N	N	N	E	N	N	N	N	N	N	N	
MC_GearOutSuperImposed	w	N	N	N	N	N	N	Ae	Ae	Wa	Wa	Ae	Ae	N	N	N	E	N	N	N	N	N	N	N	
MC_GearOutSuperImposed	d	N	N	N	N	N	N	Ae	Ae	A1	A1	Ae	Ae	N	N	N	E	N	N	N	N	N	N	N	
MC_PhasingSuperImposed		Ae	Ae	Ae	Ae	Ae	Ae	S	Ae	S	Ae	Ae	Ae	E	E	E	E	E	E	E	E	E	E	E	

6.8 命令的相互作用

d	驱动 轴正在移动，即它正在同步模式中运行，或者当前正在被同步化或去同步化。 - 在同步操作中设置 <i>InGear</i> 或 <i>InSync</i> 工艺功能的输出参数。（同步对象 <i>enablecommand=active</i>）的系统变量）。 - 当同步运动处于活动状态时，该命令处于活动状态。将设置同步轴的工艺 DB 的 <i>Statusword.SynchrCommand</i> 变量（同步对象 <i>enablecommand=active</i>）的系统变量）。 - 当去同步运动处于活动状态时，工艺功能的输出参数 <i>InGear</i> 和 <i>InSync</i> 为 <i>FALSE</i>。仍将设置同步轴的工艺 DB 的 <i>Statusword.SynchrCommand</i> 变量（同步对象 <i>disablecommand=active</i>）的系统变量）。	w	等待 轴正在等待进行同步（去同步）的条件。 - 在同步等待状态中，该命令处于活动状态。尚未设置工艺 DB 的 <i>Statusword.SynchrCommand</i> 变量（同步对象 <i>enablecommand=waiting_to_start</i> 的系统变量）。 - 在去同步等待状态中，仍将设置工艺功能的 <i>InSync</i> 和 <i>InGear</i> 输出参数。另外，将设置同步轴的工艺 DB 的 <i>Statusword.SynchrCommand</i> 变量（同步对象 <i>disablecommand=waiting_to_start</i> 的系统变量）。
A1	中止 1 执行新命令。活动的命令被立即取消。	Ac	中止方向 MC_MoveToEndPos- 命令被中止还是新命令因错误被取消，取决于新命令的方向。
Ae	中止错误 新命令被取消，在工艺 DB 上输入 <i>ErrorID</i>，并继续执行活动的命令。	E	错误 新命令在工艺功能上报告一个错误，且不执行新命令。将继续执行活动的命令。
N	无影响 对活动的功能没有任何直接影响。命令的执行取决于工艺对象状态。	S	叠加 新运动命令叠加在当前运动命令上。未取消活动的命令。
Sa	叠加中止 新运动命令叠加在当前运动命令上。未取消活动的命令。如果既不满足活动的命令的同步条件也不满足活动的命令的去同步条件，则将在满足条件时取消新命令。	Sw	叠加等待 执行新命令并等待符合去同步条件。由该新命令启动的运动叠加在活动命令的运动上，而不取消活动命令。
W	等待 新命令在存储器中排队，并以等待状态附加到活动命令后。活动命令完成后，将执行等待的命令。	Wa	等待中止 执行新命令并等待符合同步或去同步的条件。当满足新命令的同步或去同步条件时，活动命令被中止。

对于轴，上表中的内容适用于标准轴组态（组态数据元素

TypeOfAxis.DecodingConfig.transferSuperimposedPosition = TRANSFER_STANDSTILL (0))。

6.8.2 新命令 — 活动的单个命令 (2)

下表显示了新命令对活动的单个命令的相应作用。

	busy single command																new command													
	MC_CamIn	MC_CamIn	MC_GearIn	MC_GearIn	MC_CamOut	MC_GearOut	MC_CamInSuperImposed	MC_CamInSuperImposed	MC_GearInSuperImposed	MC_GearInSuperImposed	MC_CamOutSuperImposed	MC_GearOutSuperImposed	MC_Home (Mode=0, 1)	MC_Home (Mode=2-7), Achse freigegeben	MC_Halt	MC_Stop	MC_MoveAbsolute	MC_MoveAdditive	MC_MoveRelative	MC_MoveVelocity	MC_MoveToEndPos (Clamping=True)	MC_MoveToEndPos (Clamping=False)	MC_MoveSuperImposed							
MC_Home (Mode=0, 1)	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	S	A1	E	A1	A1	A1	A1	E	E	A1							
MC_Home (Mode=2-7)	S	S	S	S	S	S	S	S	S	S	S	S	A1	S	S	S	S	S	S	S	S	S	S							
MC_Halt	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	S	A1	E	A1	A1	A1	A1	E	A1	A1							
MC_Stop	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	S	A1	A1	A1	A1	A1	A1	A1	A1	A1							
MC_MoveAbsolute (Mode=0)	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	S	A1	E	A1	A1	A1	A1	Ad	A1	A1							
MC_MoveAdditive	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	S	A1	E	A1	A1	A1	A1	Ad	A1	A1							
MC_MoveRelative (Mode=0)	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	S	A1	E	A1	A1	A1	A1	Ad	A1	A1							
MC_MoveVelocity (Mode=0)	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	S	A1	E	A1	A1	A1	A1	Ad	A1	A1							
MC_MoveAbsolute (Mode=1, 2)	E	E	E	E	E	E	E	E	E	E	E	E	W	S	E	E	W	W	W	W	Ad	W	S							
MC_MoveRelative (Mode=1, 2)	E	E	E	E	E	E	E	E	E	E	E	E	W	S	E	E	W	W	W	W	Ad	W	S							
MC_MoveVelocity (Mode=1)	E	E	E	E	E	E	E	E	E	E	E	E	W	S	E	E	W	W	W	W	Ad	W	S							
MC_MoveSuperImposed	S	Sa	S	Sa	S	S	A1	A1	A1	A1	A1	A1	A1	S	S	E	S	S	S	S	Ad	S	A1							
MC_MoveToEndPos	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	E	S	A1	E	A1	A1	A1	A1	Ad	Ad	A1							

d	驱动 轴正在移动，即它正在同步模式中运行，或者当前正在被同步化或去同步化。 - 在同步操作中设置 <i>InGear</i> 或 <i>InSync</i> 工艺功能的输出参数。（同步对象 <i>enablecommand=active</i>）的系统变量）。 - 当同步运动处于活动状态时，该命令处于活动状态。将设置同步轴的工艺 DB 的 <i>Statusword.SynchrCommand</i> 变量（同步对象 <i>enablecommand=active</i>）的系统变量）。 当去同步运动处于活动状态时，工艺功能的输出参数 <i>InGear</i> 和 <i>InSync</i> 为 <i>FALSE</i>。仍将设置同步轴的工艺 DB 的 <i>Statusword.SynchrCommand</i> 变量（同步对象 <i>disablecommand=active</i>）的系统变量）。	W	等待 轴正在等待进行同步（去同步）的条件。 在同步等待状态中，该命令处于活动状态。尚未设置工艺 DB 的 <i>Statusword.SynchrCommand</i> 变量（同步对象 <i>enablecommand=waiting_to_start</i> 的系统变量）。 在去同步等待状态中，仍将设置工艺功能的 <i>InSync</i> 和 <i>InGear</i> 输出参数。另外，将设置同步轴的工艺 DB 的 <i>Statusword.SynchrCommand</i> 变量（同步对象 <i>disablecommand=waiting_to_start</i> 的系统变量）。
A1	中止 1 执行新命令。活动的命令被立即取消。	Ad	中止方向 <i>MC_MoveToEndPos</i>- 命令被中止还是新命令因错误被取消，取决于新命令的方向。
Ae	中止错误 新命令被取消，在工艺 DB 上输入 <i>ErrorID</i>，并继续执行活动的命令。	E	错误 新命令在工艺功能上报告一个错误，且不执行新命令。将继续执行活动的命令。
N	无影响 对活动的功能没有任何直接影响。命令的执行取决于工艺对象状态。	S	叠加 新运动命令叠加在当前运动命令上。未取消活动的命令。
Sa	叠加中止 新运动命令叠加在活动的运动命令上。未取消活动的命令。如果既不满足活动的命令的同步条件也不满足活动的命令的去同步条件，则将在满足条件时取消新命令。	W	等待 新命令在存储器中排队，并以等待状态附加到活动命令后。活动命令完成后，将执行等待的命令。

对于轴，上表中的内容适用于标准轴组态（组态数据元素

TypeOfAxis.DecodingConfig.transferSuperimposedPosition = TRANSFER_STANDSTILL (0)）。

6.8.3 新命令 — 活动的单个命令 (3)

下表显示了新命令对活动的单个命令的相应作用。

busy single command new command							
	MC_Power (Enable=False)						
	MC_Power (Enable=True)						
	MC_Reset (Restart=False)						
	MC_Reset (Restart=True)						
	MC_SetCharacteristic						
	MC_SetTorqueLimit						
	MC_ChangeDataset						
MC_Power (Enable=False)		A1	E			S	S
MC_Power (Enable=True)	S	E	S	S		S	S
MC_Reset (Restart=False)	S	S	A1	E	S	S	S
MC_Reset (Restart=True)	S	E	A1	E	A1	A1	A1
MC_SetCharacteristic					A1		
MC_SetTorqueLimit	S	S	S	A1		A1	S
MC_ChangeDataset				E			A1

	无相应作用，或状态不存在。	A1	中止 1 执行新命令。活动的命令被立即取消。
E	错误 新命令在工艺功能上报告一个错误，且不执行新命令。将继续执行活动的命令。	S	叠加 新运动命令叠加在活动的运动命令上。未取消活动的命令。
W	等待 新命令在存储器中排队，并以等待状态附加到活动命令后。活动命令完成后，将执行等待的命令。		

对于轴，上表中的内容适用于标准轴组态（组态数据元素

TypeOfAxis.DecodingConfig.transferSuperimposedPosition = TRANSFER_STANDSTILL (0)）。

6.8.4 新命令 — 活动的多个命令 (1)

下表显示了新命令对活动的多个命令的相应作用。

		busy single command →																							
		MC_CamIn	MC_CamIn	MC_CamIn	MC_CamIn	MC_CamOut	MC_CamOut	MC_GearIn	MC_GearOut	MC_CamIn	MC_CamIn	MC_GearIn	MC_GearOut	MC_CamIn	MC_CamIn	MC_GearIn	MC_GearOut	MC_CamIn	MC_CamIn	MC_GearIn	MC_GearOut	MC_CamIn	MC_CamIn	MC_GearIn	MC_GearOut
		w	d	d	d	d	d	w	d	d	d	d	d	d	d	d	d	w	d	d	d	w	d	d	d
		A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
MC_CamIn	w	A1	A2	A2	A2	A2	A2	A1	A2	A2	A2	A2	A2	A2	A2	A2	A2	A1	A2	A2	A2	A2	A2	A2	A2
MC_CamIn	d	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
MC_GearIn	w	A1	A2	A2	A2	A2	A2	A1	A2	A2	A2	A2	A2	A2	A2	A2	A2	A1	A2	A2	A2	A2	A2	A2	A2
MC_GearIn	d	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
MC_CamOut	w	A2	A3	A2	A2	Ae	Ae	Ae	Ae	Ae	Ae	Ae	Ae	Ae	Ae	Ae	Ae	Ae	Ae	Ae	Ae	Ae	Ae	Ae	Ae
MC_CamOut	d	A1	A1	A1	A1	A3	Ae	Ae	A3	Ae	Ae	A3	Ae	A1	Ae	N	N	N	N	N	N	N	N	N	N
MC_GearOut	w	Ae	Ae	Ae	Ae	Ae	Ae	A2	A2	A3	A2	Ae	Ae	Ae	Ae	N	N	N	N	N	N	N	N	N	N
MC_GearOut	d	Ae	Ae	Ae	A3	Ae	A3	A1	A1	A1	A1	Ae	A3	Ae	A1	N	N	N	N	N	N	N	N	N	N
MC_Phasing		Ae	S	S	S	Ae	Ae	Ae	S	S	S	Ae	Ae	A3	A3	N	N	N	N	N	N	N	N	N	N
MC_CamInSuperImposed	w	Sa	Sa	Sa	Sa	Sa	Sa	Sa	Sa	Sa	Sa	Sa	Sa	S	S	A1	A2	A2	A2	A2	A2	A	A2	A2	A2
MC_CamInSuperImposed	d	Sa	Sa	Sa	Sa	Sa	Sa	Sa	Sa	Sa	Sa	Sa	Sa	S	S	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
MC_GearInSuperImposed	w	Sa	Sa	Sa	Sa	Sa	Sa	Sa	Sa	Sa	Sa	Sa	Sa	S	S	A1	A2	A2	A2	A2	A2	A	A2	A2	A2
MC_GearInSuperImposed	d	Sa	Sa	Sa	Sa	Sa	Sa	Sa	Sa	Sa	Sa	Sa	Sa	S	S	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
MC_CamOutSuperImposed	w	N	N	N	N	N	N	N	N	N	N	N	N	N	N	A2	A3	A2	A2	Ae	Ae	Ae	Ae	Ae	Ae
MC_CamOutSuperImposed	d	N	N	N	N	N	N	N	N	N	N	N	N	N	N	A1	A1	A1	A1	A3	Ae	Ae	A3	Ae	Ae
MC_GearOutSuperImposed	w	N	N	N	N	N	N	N	N	N	N	N	N	N	N	Ae	Ae	Ae	Ae	Ae	Ae	A2	A2	A3	A2
MC_GearOutSuperImposed	d	N	N	N	N	N	N	N	N	N	N	N	N	N	N	Ae	Ae	Ae	A3	Ae	A3	A1	A1	A1	Ae
MC_PhasingSuperImposed		N	N	N	N	N	N	N	N	N	N	N	N	N	N	Ae	S	S	S	Ae	Ae	Ae	S	S	S

d	驱动 轴正在移动，即它正在同步模式中运行，或者当前正在被同步化或去同步化。 - 在同步操作中设置 <i>InGear</i> 或 <i>InSync</i> 工艺功能的输出参数。（同步对象 <i>enablecommand=active</i>）的系统变量）。 - 当同步运动处于活动状态时，该命令处于活动状态。将设置同步轴的工艺 DB 的 <i>Statusword.SynchrCommand</i> 变量（同步对象 <i>enablecommand=active</i>）的系统变量）。 - 当去同步运动处于活动状态时，工艺功能的输出参数 <i>InGear</i> 和 <i>InSync</i> 为 <i>FALSE</i>。仍将设置同步轴的工艺 DB 的 <i>Statusword.SynchrCommand</i> 变量（同步对象 <i>disablecommand=active</i>）的系统变量）。	w	等待 轴正在等待进行同步（去同步）的条件。 - 在同步等待状态中，该命令处于活动状态。尚未设置工艺 DB 的 <i>Statusword.SynchrCommand</i> 变量（同步对象 <i>enablecommand=waiting_to_start</i> 的系统变量）。 - 在去同步等待状态中，仍将设置工艺功能的 <i>InSync</i> 和 <i>InGear</i> 输出参数。另外，将设置同步轴的工艺 DB 的 <i>Statusword.SynchrCommand</i> 变量（同步对象 <i>disablecommand=waiting_to_start</i> 的系统变量）。
A1	中止 1 执行新命令。活动的多个命令将被立即取消。	A2	中止 2 执行新命令。等待的命令将被立即取消。当满足新命令的同步或去同步条件时，活动命令被取消。
A3	中止 3 执行新命令并立即取消第二个活动的命令。继续执行（而不取消）第一个活动的命令。	Ad	中止方向 <i>MC_MoveToEndPos</i> 命令被中止还是新命令因错误被取消，取决于新命令的方向。
Ae	中止错误 新命令被取消，在工艺 DB 上输入 <i>ErrorID</i>，并继续执行活动的命令。	E	错误 新命令在工艺功能上报告一个错误，且不执行新命令。将继续执行活动的命令。
N	无影响 对活动的功能没有任何直接影响。命令的执行取决于工艺对象状态。	S	叠加 新运动命令叠加在活动的运动命令上。未取消活动的多个命令。
Sa	叠加中止 新运动命令叠加在活动的运动命令上。未取消活动的多个命令。如果既不满足活动的命令的同步条件也不满足活动的命令的去同步条件，则将在满足条件时取消新命令。	Wa	等待中止 执行新命令并等待符合同步或去同步的条件。当满足新命令的同步或去同步条件时，活动命令被中止。

对于轴，上表中的内容适用于标准轴组态（组态数据元素

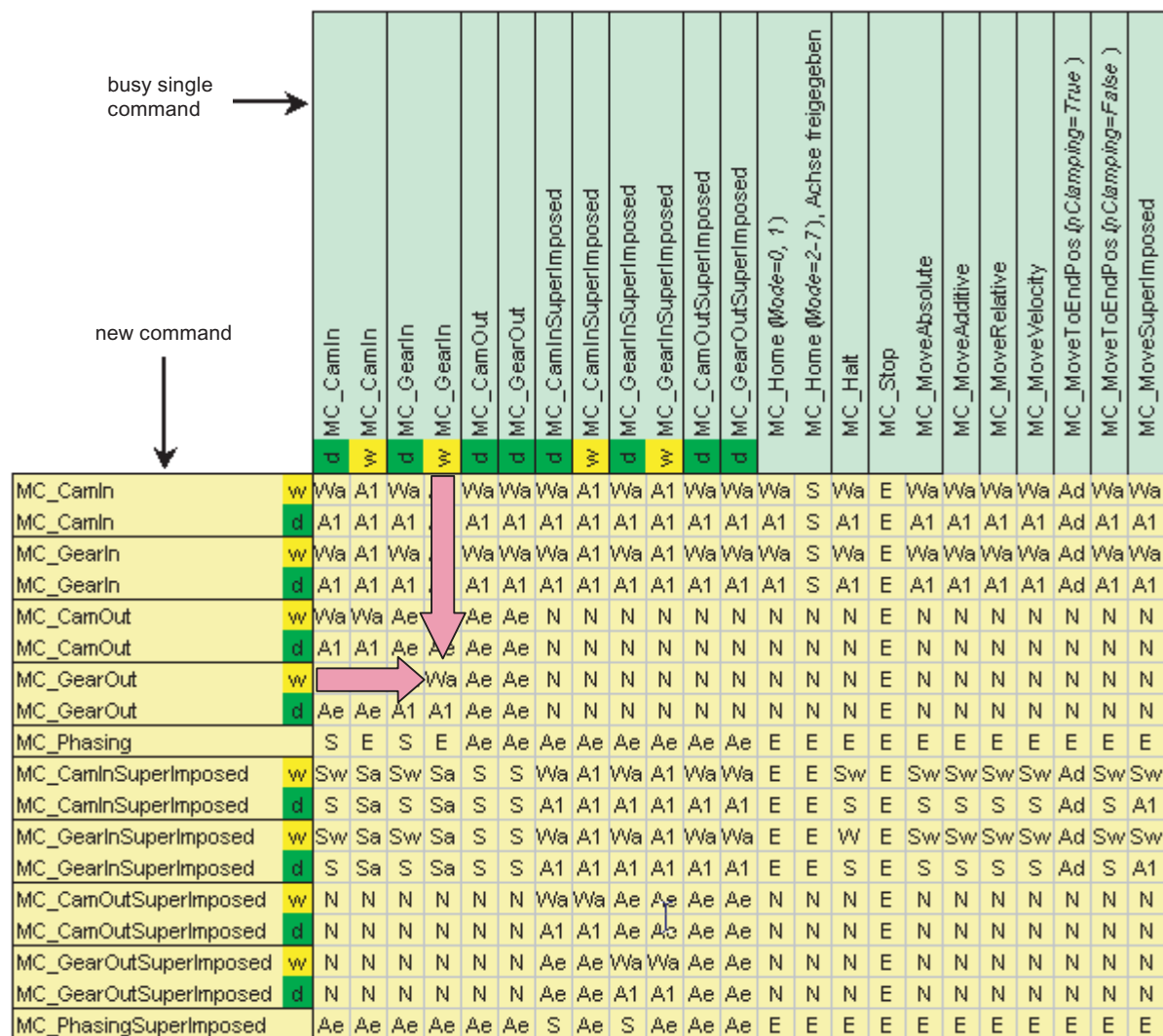
TypeOfAxis.DecodingConfig.transferSuperimposedPosition = TRANSFER_STANDSTILL (0)）。

d	驱动 轴正在移动，即它正在同步模式中运行，或者当前正在被同步化或去同步化。 - 在同步操作中设置 <i>InGear</i> 或 <i>InSync</i> 工艺功能的输出参数。（同步对象 <i>enablecommand=active</i>）的系统变量）。 - 当同步运动处于活动状态时，该命令处于活动状态。将设置同步轴的工艺 DB 的 <i>Statusword.SynchrCommand</i> 变量（同步对象 <i>enablecommand=active</i>）的系统变量）。 - 当去同步运动处于活动状态时，工艺功能的输出参数 <i>InGear</i> 和 <i>InSync</i> 为 <i>FALSE</i>。仍将设置同步轴的工艺 DB 的 <i>Statusword.SynchrCommand</i> 变量（同步对象 <i>disablecommand=active</i>）的系统变量）。	ww	等待 轴正在等待进行同步（去同步）的条件。 - 在同步等待状态中，该命令处于活动状态。尚未设置工艺 DB 的 <i>Statusword.SynchrCommand</i> 变量（同步对象 <i>enablecommand=waiting_to_start</i> 的系统变量）。 - 在去同步等待状态中，仍将设置工艺功能的 <i>InSync</i> 和 <i>InGear</i> 输出参数。另外，将设置同步轴的工艺 DB 的 <i>Statusword.SynchrCommand</i> 变量（同步对象 <i>disablecommand=waiting_to_start</i> 的系统变量）。
A1	中止 1 执行新命令。活动的多个命令将被立即取消。	S	叠加 新运动命令叠加在活动的运动命令上。未取消活动的多个命令。
Sa	叠加中止 新运动命令叠加在活动的运动命令上。未取消活动的多个命令。如果既不满足活动的命令的同步条件也不满足活动的命令的去同步条件，则将在满足条件时取消新命令。		

对于轴，上表中的内容适用于标准轴组态（组态数据元素

TypeOfAxis.DecodingConfig.transferSuperimposedPosition = TRANSFER_STANDSTILL (0)）。

示例 1



T-CPU 通过设置“Wa”进行响应。响应“Wa”的含义在相应的图例中进行了定义。

示例 2

[illegible]

一个工艺对象处于同步状态。MC_GearIn 命令处于活动状态且输出参数 *InGear* = TRUE (“d”)。MC_GearOut 命令正在等待满足其去同步条件 (“w”)。两个命令引用同一工艺对象。

通过同一工艺对象启动一个新 MC_Phasing 命令。

T-CPU 通过设置“S”进行响应。响应“S”的含义在相应的图例中进行了定义。

6.9 有关参数的信息

6.9.1 断电和重启后工艺功能的响应

工艺功能在每次断电（断开电源 -> 接通电源）和重新启动（运行 — 停止 -> 停止 — 运行）后都将进行初始化。

说明
如果断电或重启后在输入参数 *Execute* 中显示值 *TRUE*，则系统会检测到一个沿并启动命令。
为了防止命令异常启动，请打开 **SIMATIC Manager**，然后设置背景数据块的块属性中的“不保留”(non-retain) 复选框。
如果设置了“不保留”(non-retain)，则相关块在通电或重启后将再次进行初始化。

6.9.2 生成 DoneFlag

DoneFlag 可用于显示和判断本地 MCDevice DB 中工艺对象命令的完成情况。
通过工艺功能的输入参数 *DoneFlag*，您可以定义是否在 MCDevice DB 中生成 *DoneFlag*，以及如果要生成，生成哪个 *DoneFlag*。

参数值	含义
0	不在 MCDevice DB 中生成 <i>DoneFlag</i>
1	在 MCDevice DB 中生成 <i>DoneFlag</i> [1]
...	...
32	在 MCDevice DB 中生成 <i>DoneFlag</i> [32]

每次完成一个命令时，*DoneFlag* 中的值就会在 MCDevice DB 中取反（*FALSE* 变为 *TRUE* 而 *TRUE* 变为 *FALSE*）。只有可以监视初始命令后才能设置 *DoneFlag*。如果后续命令使用同一实例，则无法继续监视此命令。

6.9.3 值范围

REAL 值的范围

以下范围适用于工艺功能的输入参数的所有 REAL 值：

$$-1E+12 < \text{REAL 值} < +1E+12$$

输入参数说明中指定的限制也适用。

在开始执行运动控制命令时验证集成工艺中的这些值。如果超出这些值范围，将在工艺 DB 和工艺功能中输出错误消息。

工艺功能中动态参数的值范围

您可以使用输入参数 *Velocity*、*Acceleration*、*Deceleration* 和 *Jerk* 定义轴运动的动态响应限制。

如果运动的动态值位于轴组态的默认限制之内，轴响应如下：

动态参数	< 0 时响应	= 0 时响应	> 0 时响应
<i>Velocity</i>	使用在轴组态中设置的默认速度	错误：命令被忽略 例外情况： "MC_MoveVelocity"	使用速度
<i>Acceleration</i>	使用轴组态中设置的默认加速度	错误：命令被忽略 (不允许)	使用加速度
<i>Deceleration</i>	使用轴组态中设置的默认减速度	错误：命令被忽略	使用减速度
<i>Jerk</i>	使用轴组态中设置的默认冲击值	使用梯形曲线	使用冲击

如果运动的动态值在轴组态中的设置默认限制之外，则将在工艺 DB 中输出警告。

说明

您可以通过调用“MC_WriteParameter”工艺功能编辑轴组态的默认限制。

说明

最初，不会在开始执行运动控制命令时验证工艺功能中的动态值。该命令将输出到系统。系统验证动态值并在相应的工艺 DB 中报告所有错误。它还将在工艺功能中设置输出参数 *CommandAborted*。

工艺 DB — 值范围

工艺 DB 的值范围与用于数据块的 T-CPU 中所执行的数量范围相对应。有关值范围的信息，请参考《T-CPU 手册》。

工艺功能的输入参数 *Axis*, *Master* 等的数据类型为 INT。此数据类型可使用从 0 到 +32767 的正值对工艺 DB 进行定址。建议将工艺数据块存储在地址区 0 到 +32767 中。

如果不可以，您可以在范围 32768 到 65535 内对工艺 DB 进行定址，如下所示：

- 解决方案 1
使用功能 FC400“DB2INT”。
- 解决方案 2
在 *Axis* 等输入参数中为工艺 DB 编号输入以下值：
值 = 工艺 DB 的编号 — 65536
- 解决方案 3
使用变量，例如：

– 在块的减速区中定义变量：

```
VAR_TEMP  
TechDB: INT;  
END_VAR
```

– 在调用工艺功能时使用变量：

```
L L#56000 //工艺 DB 的编号  
T #TechDBNR  
CALL "MC_Power" , DB401  
Axis := #TechDBNR  
T #TechDBNR
```

6.9.4 模数轴的绝对位置

工艺将所有绝对位置参数转换为模数轴周期。

模数起始值为 0°，模数长度为 360° 的轴的示例：

- 绝对位置 400° 会将轴定位到 40° (400° 取模 360° = 40°)。
- -20° 的回原点位置设置等于回原点位置 340°。
- 输出凸轮的起始位置设置为 730°。输出凸轮 ON 位置为 10° (730° 取模 360° = 10°)

工艺数据块

7.1 “速度控制轴”工艺 DB

适用于固件版本为 V4.1.x 的集成工艺

本部分介绍了固件版本为 V3.0.x、V3.1.x 和 V3.2.x 的应用。

集成工艺将数据写入 DB。该 DB 在 STEP 7 用户程序中为只读。

无论 CPU 中的数据保持设置如何，该 DB 都不具有保持性。

UpdateFlag / UpdateCounter 变量

请注意以下几点：更新工艺 DB (页 829)

ErrorID 变量

返回速度控制轴上最新检测到的错误或警告的 *ErrorID*。

错误条目可以通过调用工艺功能“MC_Reset”进行确认（其中，*Axis* = 工艺 DB 的编号）。

请注意以下几点：可能出现的错误消息和警告 (页 835)

ErrorBuffer[0..2] 变量

收到的前三条错误和警告的存储器。第一个错误写到 ARRAY 元素 0，第二个错误写到 ARRAY 元素 1，依次类推。

此存储器中的内容可以通过调用工艺功能“MC_Reset”进行清除（其中，*Axis* = 工艺 DB 的编号）。

TO_Adaption 变量

此变量是为内部功能保留的。它不包含任何与用户相关的信息。

ErrorStatus.xxx 变量

ErrorStatus.xxx 变量中的位将返回有关轴错误的信息。

位号	变量	<i>TRUE</i> 状态的意义
0	<i>ErrorStatus.SystemFault</i>	内部系统错误
1	<i>ErrorStatus.ConfigFault</i>	工艺对象组态错误
2	<i>ErrorStatus.UserFault</i>	用户程序由于输出无效的命令而引发错误
3	<i>ErrorStatus.FaultDrive</i>	驱动器或工艺对象报告错误
4	<i>ErrorStatus.Reserve4</i>	未使用
5	<i>ErrorStatus.FollowingWarning</i>	未使用
6	<i>ErrorStatus.FollowingError</i>	未使用
7	<i>ErrorStatus.StandstillFault</i>	轴已移出停止窗口，或者无法在指定的时间内到达停止窗口。
8	<i>ErrorStatus.PositioningError</i>	未使用
9	<i>ErrorStatus.SynchronOpError</i>	未使用
10	<i>ErrorStatus.DynamicError</i>	已超出动态响应限制
11	<i>ErrorStatus.ClampingError</i>	未使用
12	<i>ErrorStatus.SoftwareLimitPos</i>	未使用
13	<i>ErrorStatus.SoftwareLimitNeg</i>	未使用
14	<i>ErrorStatus.LimitSwitchActive</i>	硬件限位开关处于活动状态
15	<i>ErrorStatus.SensorFreqViolation</i>	已超出编码器限制频率。
16	<i>ErrorStatus.ReferenceNotFound</i>	未使用
17	<i>ErrorStatus.ZeroMonitoring</i>	零标记脉冲监视已检测到错误（不是回原点错误）
18	<i>ErrorStatus.Overspeed</i>	未使用
19	<i>ErrorStatus.FollowObjectError</i>	未使用
20	<i>ErrorStatus.SupImpFollowObjectError</i>	未使用

Statusword.xxx 变量

变量 *Statusword* 指示速度控制轴的当前状态。

位号	变量 (S7T Config 中的系统变量)	TRUE 状态的意义
0	<i>Statusword.DriveEnabled</i>	脉冲使能和驱动器使能处于活动状态。（脉冲使能和驱动器使能与轴的启用不相同）。 对于虚拟轴，此值始终为 <i>TRUE</i> 。 脉冲使能在跟进模式中不处于活动状态（“MC_Power”，其中 <i>Mode = 3</i> ）。
	<i>(actormonitoring.power + actormonitoring.drivestate)</i>	
1	<i>Statusword.HomingDone</i>	与速度控制轴无关。
	<i>(positioningstate.homed)</i>	
2	<i>Statusword.Done</i>	未执行任何运动命令（在这种情况下，“MC_Power”不代表运动命令）。
3	<i>Statusword.SuperImposedCommand</i>	与速度控制轴无关。
4	<i>Statusword.Error</i>	已出现了至少一个错误（变量 <i>ErrorID = 8xxx</i> ）。出现警告时，将保持 <i>Statusword.Error = FALSE</i> 状态。
5	<i>Statusword.Errorstop</i>	此轴由于错误事件已停止；工艺对象可能被禁用。 消除错误原因并确认错误。
	<i>(errorreaction)</i>	
6	<i>Statusword.Stopping</i>	轴上活动的 MC_Stop 命令。新的运动命令被拒绝。 运动命令的状态显示在变量 <i>Statusword.Standstill</i> , <i>Statusword.ConstantVelocity</i> , <i>Statusword.Accelerating</i> 或 <i>Statusword.Decelerating</i> 中。响应时可能会有延迟。
7	<i>Statusword.Standstill</i>	轴速度（绝对值）小于组态的停止限制。
	<i>(motionstatedata.stillstandvelocity)</i>	
8	<i>Statusword.PositioningCommand</i>	与速度控制轴无关。
	<i>(poscommand.state)</i>	
9	<i>Statusword.SpeedCommand</i>	轴上活动的速度命令。示例： • "MC_MoveVelocity" • "MC_MoveToEndPos"（在检测到固定挡块之前）
	<i>(movecommand.state)</i>	
10	<i>Statusword.SynchrCommand</i>	与速度控制轴无关。

7.1 “速度控制轴”工艺 DB

位号	变量 (S7T Config 中的系统变量)	TRUE 状态的意义
11	<i>Statusword.Homing</i>	与速度控制轴无关。
	<i>(homingcommand.state)</i>	
12	<i>Statusword.FollowUpControl</i>	在跟随模式处于活动状态时进行设置。
	<i>(control</i> (取反)	这是指使用“MC_Power”禁用轴或者使用“ <i>Mode = 3</i> ”启用轴的情况。
13	<i>Statusword.ConstantVelocity</i>	在速度设定值仍为常量时进行设置。
	<i>(motionstatedata.motionstate)</i>	
14	<i>Statusword.Accelerating</i>	轴正在加速（增加驱动功率）。
	<i>(motionstatedata.motionstate)</i>	
15	<i>Statusword.Decelerating</i>	轴正在减速（降低驱动功率）。
	<i>(motionstatedata.motionstate)</i>	
16	<i>Statusword.RequestRestart</i>	轴参数已被更改，直到完成下一次重启后才会被激活（“MC_Reset”，其中 Restart = TRUE）。
17	<i>Statusword.Simulation</i>	轴处于仿真模式。
	<i>(simulation)</i>	您可以通过设置 <i>Mode = 4</i> 使用“MC_Power”来实现此目的。
18	<i>Statusword.CyclicInterface</i>	在虚拟轴上： 此变量的值始终为 <i>TRUE</i>。 在电气轴上： 控制器和驱动器之间的循环通信处于活动状态。没有活动的驱动器错误。 在液压轴上： 已在 DP(DRIVE) 上检测到用于控制信号输出的 I/O 和编码器信号，并且这些信号可用于轴工艺对象。 如果多个轴工艺对象是在同一 I/O 地址上组态的，则 <i>Statusword.CyclicInterface</i> 将在所有相关的工艺 DB 上返回 <i>TRUE</i>。所指定的状态与任何工艺对象对 I/O 的访问无关。
	<i>(actormonitoring.cyclicinterface)</i>	
19	<i>Statusword.EncoderValid</i>	与速度控制轴无关。
	<i>(sensordata.sensordata[当前编码器].state = VALID)</i>	

位号	变量 (S7T Config 中的系统变量)	TRUE 状态的意义
20	<i>Statusword.SpeedMode</i> (<i>speedmode</i>)	在速度控制轴上，该值始终为 TRUE。
21	<i>Statusword.TorqueLimiting</i> (<i>torquelimitingcommand.torquelimitingstate</i>)	轴的扭矩受“MC_SetTorqueLimit”或“MC_MoveToEndPos”命令的限制，而驱动器将运行到此限制。驱动器必须支持扭矩限制（消息帧 10x）。
22	<i>Statusword.SuplmpSynchrCommand</i>	与速度控制轴无关。
23	<i>Statusword.TorqueLimitingCommand</i> (<i>torquelimitingcommand.state</i>)	轴上的减小扭矩命令（“MC_SetTorqueLimit”或“MC_MoveToEndPos”）处于活动状态。
24	<i>Statusword.RequestStartUp</i>	“减小”、“偏移”或“IPO 同步”条目已在 "Technology Objects Management" 中发生了更改。直到再次启动控制器，更改的值才会被激活。
25	<i>Statusword.ToDeactivated</i>	工艺对象已取消激活
26	<i>Statusword.HWLimitSwitchMinus</i>	硬件限位开关在运动的负方向上逼近
27	<i>Statusword.HWLimitSwitchPlus</i>	硬件限位开关在运动的正方向上逼近

CommandVelocity 变量

(S7T Config 系统变量 *motionstatedata.commandvelocity*)

显示轴速度的当前设定值规范（已为当前 *SpeedOverride* 值留出余量）。

CommandAcceleration 变量

(S7T Config 系统变量 *motionstatedata.commandacceleration*)

显示轴加速度的当前设定值规范（已为当前 *AccelerationOverride* 值留出余量）。

ActualVelocity 变量

(S7T Config 系统变量 *motionstatedata.actualvelocity*)

该变量将返回轴的当前速度。

如果速度控制轴缺少编码器，则该值始终为 0。

7.1 “速度控制轴”工艺 DB

ActualAcceleration 变量

(S7T Config 系统变量 *motionstatedata.actualacceleration*)

返回轴的当前加速度。

如果速度控制轴缺少编码器，则该值始终为 0。

SpeedOverride 变量

(S7T Config 系统变量 *override.velocity*)

该变量将返回当前有效的 *SpeedOverride* 值的百分比。该百分比与轴的速度设定值相对应。默认值为 100%。

可以使用 MC_WriteParameter 工艺功能在范围 0% 到 200% 之间设置 *SpeedOverride* 值。

AccelerationOverride 变量

(S7T Config 系统变量 *override.acceleration*)

该变量将返回当前有效的 *AccelerationOverride* 值的百分比。该百分比与轴的加速度设定值相对应。默认值 = 100%。

可以使用 MC_WriteParameter 工艺功能在范围 1% 到 1,000% 之间设置 *AccelerationOverride* 值。

7.2 “定位轴”工艺 DB

适用于固件版本为 V4.1.x 的集成工艺

本部分介绍了固件版本为 V3.0.x、V3.1.x 和 V3.2.x 的应用。

集成工艺将数据写入 DB。该 DB 在 STEP 7 用户程序中为只读。

无论 CPU 中的数据保持设置如何，该 DB 都不具有保持性。

UpdateFlag / UpdateCounter 变量

请注意以下几点：更新工艺 DB (页 829)

ErrorID 变量

该变量将返回在定位轴上最近检测到的错误或警告的 *ErrorID*。

错误条目可以通过调用工艺功能“MC_Reset”进行确认（其中，*Axis* = 工艺 DB 的编号）。

请注意以下几点：可能出现的错误消息和警告 (页 835)

ErrorBuffer[0..2] 变量

收到的前三条错误和警告的存储器。第一个错误写到 ARRAY 元素 0，第二个错误写到 ARRAY 元素 1，依次类推。

此存储器中的内容可以通过调用工艺功能“MC_Reset”进行清除（其中，*Axis* = 工艺 DB 的编号）。

TO_Adaption 变量

此变量是为内部功能保留的。它不包含任何与用户相关的信息。

ErrorStatus.xxx 变量

ErrorStatus.xxx 变量中的位将返回有关定位轴错误的信息。

位号	变量	<i>TRUE</i> 状态的意义
0	<i>ErrorStatus.SystemFault</i>	内部系统错误
1	<i>ErrorStatus.ConfigFault</i>	工艺对象组态错误
2	<i>ErrorStatus.UserFault</i>	用户程序由于输出无效的命令而引发错误
3	<i>ErrorStatus.FaultDrive</i>	驱动器或工艺对象报告错误
4	<i>ErrorStatus.Reserve4</i>	未使用
5	<i>ErrorStatus.FollowingWarning</i>	动态跟随误差监视功能报告超出警告限制
6	<i>ErrorStatus.FollowingError</i>	动态跟随误差监视功能报告超出窗口限制
7	<i>ErrorStatus.StandstillFault</i>	轴已移出停止窗口，或者无法在指定的时间内到达停止窗口。
8	<i>ErrorStatus.PositioningError</i>	轴无法在指定的时间内到达定位窗口。
9	<i>ErrorStatus.SynchronOpError</i>	已超出同步公差
10	<i>ErrorStatus.DynamicError</i>	已超出动态响应限制
11	<i>ErrorStatus.ClampingError</i>	紧固时出错 未收到新的运动命令，轴就已超出“固定挡块检测后的位置公差”。 位置公差在 S7T Config 中的“限制”(Limits) > “固定挡块”(Fixed end stop) 选项卡 > “固定挡块检测后的位置公差”(Position tolerance after fixed end stop detection) 参数中设置。
12	<i>ErrorStatus.SoftwareLimitPos</i>	已达到或超出软件限位开关上限
13	<i>ErrorStatus.SoftwareLimitNeg</i>	已达到或超出软件限位开关下限
14	<i>ErrorStatus.LimitSwitchActive</i>	硬件限位开关处于活动状态
15	<i>ErrorStatus.SensorFreqViolation</i>	已超出编码器限制频率。
16	<i>ErrorStatus.ReferenceNotFound</i>	回原点时没找到参考凸轮或零标记脉冲
17	<i>ErrorStatus.ZeroMonitoring</i>	零标记脉冲监视已检测到错误（不是回原点错误）
18	<i>ErrorStatus.Overspeed</i>	未使用
19	<i>ErrorStatus.FollowObjectError</i>	未使用
20	<i>ErrorStatus.SupImpFollowObjectError</i>	未使用

Statusword.xxx 变量

变量 *Statusword* 指明该定位轴的当前状态。

位号	变量 (S7T Config 中的系统变量)	TRUE 状态的意义
0	<i>Statusword.DriveEnabled</i>	脉冲使能和驱动器使能处于活动状态。（脉冲使能和驱动器使能与轴的启用不相同）。 对于虚拟轴，此值始终为 <i>TRUE</i> 。 脉冲使能在跟随模式中不处于活动状态（“MC_Power”，其中 <i>Mode = 3</i> ）。
	<i>(actormonitoring.power + actormonitoring.drivestate)</i>	
1	<i>Statusword.HomingDone</i>	轴已回原点。满足了回原点条件。 在活动的回原点期间，将在检测到编码器零标记时设置工艺 DB 中的 <i>Statusword.HomingDone</i> 。但是，轴仍按参考点偏移的值减速或移动。 仅当轴停止后工艺功能“MC_Home”才报告 <i>Done=TRUE</i> 。该轴现在已到达参考点坐标的位置。
	<i>(positioningstate.homed)</i>	
2	<i>Statusword.Done</i>	未执行任何运动命令（在这种情况下，“MC_Power”不代表运动命令）。
3	<i>Statusword.SuperImposedCommand</i>	叠加的运动处于活动状态（例如，“MC_MoveSuperImposed”
4	<i>Statusword.Error</i>	已出现了至少一个错误（变量 <i>ErrorID = 8xxx</i> ）。出现警告时，将保持 <i>Statusword.Error = FALSE</i> 状态。
5	<i>Statusword.Errorstop</i>	此轴由于错误事件已停止；工艺对象可能被禁用。 消除错误原因并确认错误。
	<i>(errorreaction <> NONE)</i>	
6	<i>Statusword.Stopping</i>	轴上活动的 MC_Stop 命令。新的运动命令被拒绝。 运动命令的状态显示在变量 <i>Statusword.Standstill</i> , <i>Statusword.ConstantVelocity</i> 、 <i>Statusword.Accelerating</i> 或 <i>Statusword.Decelerating</i> 中。响应时可能会有延迟。
7	<i>Statusword.Standstill</i>	轴速度（绝对值）小于组态的停止限制。
	<i>(motionstatedata.stillstandvelocity)</i>	
8	<i>Statusword.PositioningCommand</i>	轴上的定位命令处于活动状态（也可能被叠加）。
	<i>(poscommand.state)</i>	

位号	变量 (S7T Config 中的系统变量)	TRUE 状态的意义
9	<i>Statusword.SpeedCommand</i>	轴上活动的速度命令。示例： "MC_MoveVelocity" "MC_MoveToEndPos"（在检测到固定挡块之前）
	<i>(movecommand.state)</i>	
10	<i>Statusword.SynchrCommand</i>	与定位轴无关。
11	<i>Statusword.Homing</i>	在 "MC_Home" 开始时置位该位，在结束时复位该位。
	<i>(homingcommand.state)</i>	
12	<i>Statusword.FollowUpControl</i>	在跟进模式处于活动状态时进行设置。 这是指使用“MC_Power”禁用轴或者使用“ <i>Mode = 3</i> ”启用轴的情况。
	<i>(control (取反))</i>	
13	<i>Statusword.ConstantVelocity</i>	在速度设定值仍为常量时进行设置。
	<i>(motionstatedata.motionstate)</i>	
14	<i>Statusword.Accelerating</i>	轴正在加速（增加驱动功率）。
	<i>(motionstatedata.motionstate)</i>	
15	<i>Statusword.Decelerating</i>	轴正在减速（降低驱动功率）。
	<i>(motionstatedata.motionstate)</i>	
16	<i>Statusword.RequestRestart</i>	轴参数已被更改，直到完成下一次重启后才会被激活（“MC_Reset”，其中 <i>Restart = TRUE</i> ）。
17	<i>Statusword.Simulation</i>	轴处于仿真模式。 您可以通过设置 <i>Mode = 4</i> 使用“MC_Power”来实现此目的。
	<i>(simulation)</i>	
18	<i>Statusword.CyclicInterface</i>	在虚拟轴上： 此变量的值始终为 <i>TRUE</i>。 在电气轴上： 控制器和驱动器之间的循环通信处于活动状态。没有活动的驱动器错误。 在液压轴上： 已在 DP(DRIVE) 上检测到用于控制信号输出的 I/O 和编码器信号，并且这些信号可用于轴工艺对象。 如果多个轴工艺对象是在同一 I/O 地址上组态的，则 <i>Statusword.CyclicInterface</i> 将在所有相关的工艺 DB 上返回 <i>TRUE</i>。所指定的状态与任何工艺对象对 I/O 的访问无关。
	<i>(actormonitoring.cyclicinterface)</i>	

位号	变量 (S7T Config 中的系统变量)	TRUE 状态的意义
19	<i>Statusword.EncoderValid</i> (<i>sensordata.sensordata[当前编码器].state = VALID</i>)	与轴关联的编码器的实际位置值有效（在绝对值编码器的启动阶段尤为重要）。
20	<i>Statusword.SpeedMode</i> (<i>speedmode</i>)	<i>TRUE</i> = 速度控制模式 <i>FALSE</i> = 位置控制模式 通过设置输入参数 <i>PositionControl = FALSE</i> 在工艺功能“MC_MoveVelocity”上激活速度控制模式。
21	<i>Statusword.TorqueLimiting</i> (<i>torquelimitecommand.torqueLimitingstate</i>)	轴的扭矩受“MC_SetTorqueLimit”或“MC_MoveToEndPos”命令的限制，而驱动器将运行到此限制。驱动器必须支持扭矩限制（消息帧 10x）。
22	<i>Statusword.SupImpSynchrCommand</i>	与定位轴无关。
23	<i>Statusword.TorqueLimitingCommand</i> (<i>torquelimitecommand.state</i>)	轴上的减小扭矩命令（“MC_SetTorqueLimit”或“MC_MoveToEndPos”）处于活动状态。
24	<i>Statusword.RequestStartUp</i>	“减小”、“偏移”或“IPO 同步”条目已在“Technology Objects Management”中发生了更改。直到再次启动控制器，更改的值才会被激活。
25	<i>Statusword.ToDeactivated</i>	工艺对象已取消激活
26	<i>Statusword.HWLimitSwitchMinus</i>	硬件限位开关在运动的负方向上逼近
27	<i>Statusword.HWLimitSwitchPlus</i>	硬件限位开关在运动的正方向上逼近

CommandVelocity 变量

(S7T Config 系统变量 *motionstatedata.commandvelocity*)

显示轴速度的当前设定值规范（已为当前 *SpeedOverride* 值留出余量）。

CommandAcceleration 变量

(S7T Config 系统变量 *motionstatedata.commandacceleration*)

显示轴加速度的当前设定值规范（已为当前 *AccelerationOverride* 值留出余量）。

ActualVelocity 变量

(S7T Config 系统变量 *motionstatedata.actualvelocity*)

该变量将返回轴的当前速度。

ActualAcceleration 变量

(S7T Config 系统变量 *motionstatedata.actualacceleration*)

返回轴的当前加速度。

SpeedOverride 变量

(S7T Config 系统变量 *override.velocity*)

该变量将返回当前有效的 *SpeedOverride* 值的百分比。该百分比与轴的速度设定值相对应。默认值为 100%。

可以使用“MC_WriteParameter”工艺功能在范围 0% 到 200% 之间设置 *SpeedOverride* 值。

AccelerationOverride 变量

(S7T Config 系统变量 *override.acceleration*)

返回当前活动的 *AccelerationOverride* 的百分比。该百分比与轴的加速度设定值相对应。默认值 = 100%。

可以使用 *MC_WriteParameter* 工艺功能在范围 1% 到 1,000% 之间设置 *AccelerationOverride* 值。

ActualPosition 变量

(S7T Config 系统变量 *positioningstate.actualposition*)

显示当前轴位置。

CommandPosition 变量

(S7T Config 系统变量 *positioningstate.commandposition*)

表示定位操作的当前位置设定值。

TargetPosition 变量

指明当前命令的目标位置。所指定的值仅适用于活动的定位命令。

Distance 变量

(S7T Config 系统变量 *poscommand.distance*to go)

显示相对/绝对定位的当前距离。所指定的值仅适用于活动的定位命令。

DecelerationDistance 变量

(S7T Config 系统变量 *poscommand.deceleration*distance)

该变量将返回轴的当前停止距离。所指定的值仅适用于活动的定位命令。

FollowingError 变量

(S7T Config 系统变量 *servodata.following*error)

指明轴的当前跟随误差值。所指定的值仅适用于活动的定位命令。

7.3 “同步轴”工艺 DB

适用于固件版本为 V4.1.x 的集成工艺

本部分介绍了固件版本为 V3.0.x、V3.1.x、V3.2.x 的应用。

集成工艺将数据写入 DB。该 DB 在 STEP 7 用户程序中为只读。

无论 CPU 中的数据保持设置如何，该 DB 都不具有保持性。

UpdateFlag / UpdateCounter 变量

请注意：更新工艺 DB (页 829)

ErrorID 变量

显示在同步轴上最近检测到的错误或警告的 ErrorID。

错误条目可以通过调用工艺功能“MC_Reset”进行确认（其中，*Axis* = 工艺 DB 的编号）。

请注意：可能出现的错误消息和警告 (页 843)

ErrorBuffer[0..2] 变量

收到的前三条错误和警告的存储器。第一个错误写到 ARRAY 元素 0，第二个错误写到 ARRAY 元素 1，依次类推。

此存储器中的内容可以通过调用工艺功能“MC_Reset”进行清除（其中，*Axis* = 工艺 DB 的编号）。

TO_Adaption 变量

此变量是为内部功能保留的。它不包含任何与用户相关的信息。

ErrorStatus.xxx 变量

ErrorStatus.xxx 变量中的位将返回有关轴错误的信息。

位号	变量	<i>TRUE</i> 状态的意义
0	<i>ErrorStatus.SystemFault</i>	内部系统错误
1	<i>ErrorStatus.ConfigFault</i>	工艺对象组态错误
2	<i>ErrorStatus.UserFault</i>	用户程序由于输出无效的命令而引发错误
3	<i>ErrorStatus.FaultDrive</i>	驱动器或工艺对象报告错误
4	<i>ErrorStatus.Reserve4</i>	未使用
5	<i>ErrorStatus.FollowingWarning</i>	动态跟随误差监视功能报告超出警告限制
6	<i>ErrorStatus.FollowingError</i>	动态跟随误差监视功能报告超出窗口限制
7	<i>ErrorStatus.StandstillFault</i>	轴已移出停止窗口，或者无法在指定的时间内到达停止窗口。
8	<i>ErrorStatus.PositioningError</i>	轴无法在指定的时间内到达定位窗口。
9	<i>ErrorStatus.SynchronOpError</i>	已超出同步公差
10	<i>ErrorStatus.DynamicError</i>	已超出动态响应限制
11	<i>ErrorStatus.ClampingError</i>	紧固时出错 未收到新的运动命令，轴就已超出“固定挡块检测后的位置公差”。 位置公差在 S7T Config 中的“限制”(Limits) > “固定挡块”(Fixed end stop) 选项卡 > “固定挡块检测后的位置公差”(Position tolerance after fixed end stop detection) 参数中设置。
12	<i>ErrorStatus.SoftwareLimitPos</i>	已达到或超出软件限位开关上限
13	<i>ErrorStatus.SoftwareLimitNeg</i>	已达到或超出软件限位开关下限
14	<i>ErrorStatus.LimitSwitchActive</i>	硬件限位开关处于活动状态
15	<i>ErrorStatus.SensorFreqViolation</i>	已超出编码器限制频率。
16	<i>ErrorStatus.ReferenceNotFound</i>	回原点时没找到参考凸轮或零标记脉冲
17	<i>ErrorStatus.ZeroMonitoring</i>	零标记脉冲监视已检测到错误（不是回原点错误）
18	<i>ErrorStatus.Overspeed</i>	未使用
19	<i>ErrorStatus.FollowObjectError</i>	同步对象出错。
20	<i>ErrorStatus.SupImpFollowObjectError</i>	叠加同步对象出错。

Statusword.xxx 变量

变量 *Statusword* 指明同步轴的当前状态。

位号	变量 (S7T Config 中的系统变量)	<i>TRUE</i> 状态的意义
0	<i>Statusword.DriveEnabled</i>	脉冲使能和驱动器使能处于活动状态。（脉冲使能和驱动器使能与轴的启用不相同）。 对于虚拟轴，此值始终为 <i>TRUE</i> 。 脉冲使能在跟进模式中不处于活动状态（“MC_Power”，其中 <i>Mode</i> = 3）。
	(<i>actormonitoring.power + actormonitoring.drivestate</i>)	
1	<i>Statusword.HomingDone</i>	轴已回原点。满足了回原点条件。 在活动的回原点操作中，当检测到编码器零标记时会设置参考点坐标和工艺 DB 中的 <i>Statusword.HomingDone</i> 。但是，轴仍按参考点偏移的值减速或移动。但轴已回原点。仅当轴停止后工艺功能“MC_Home”才报告 <i>Done=TRUE</i> 。
	(<i>positioningstate.homed</i>)	
2	<i>Statusword.Done</i>	未执行任何运动命令（在这种情况下，“MC_Power”不代表运动命令）。
3	<i>Statusword.SuperImposedCommand</i>	叠加的运动处于活动状态（例如，“MC_MoveSuperImposed”）
4	<i>Statusword.Error</i>	已出现了至少一个错误（变量 <i>ErrorID</i> = 8xxx）。出现警告时，将保持 <i>Statusword.Error</i> = <i>FALSE</i> 状态。
5	<i>Statusword.Errorstop</i>	此轴由于错误事件已停止；工艺对象可能被禁用。 消除错误原因并确认错误。
	(<i>errorreaction <> NONE</i>)	
6	<i>Statusword.Stopping</i>	轴上活动的 MC_Stop 命令。新的运动命令被拒绝。 运动命令的状态显示在变量 <i>Statusword.Standstill</i> , <i>Statusword.ConstantVelocity</i> , <i>Statusword.Accelerating</i> 或 <i>Statusword.Decelerating</i> 中。响应时可能会有延迟。
7	<i>Statusword.Standstill</i>	轴速度（绝对值）小于组态的停止限制。
	(<i>motionstatedata.stillstandvelocity</i>)	
8	<i>Statusword.PositioningCommand</i>	轴上的定位命令处于活动状态（也可能被叠加）。
	(<i>poscommand.state</i>)	

位号	变量 (S7T Config 中的系统变量)	TRUE 状态的意义
9	<i>Statusword.SpeedCommand</i>	轴上活动的速度命令。示例： “MC_MoveVelocity” “MC_MoveToEndPos”（在检测到固定挡块之前）
	<i>(movecommand.state)</i>	
10	<i>Statusword.SynchrCommand</i>	同步操作中的轴（活动的“MC_GearIn”或“MC_CamIn”）
11	<i>Statusword.Homing</i>	在“MC_Home”开始时置位该位，在结束时复位该位。
	<i>(homimingcommand.state)</i>	
12	<i>Statusword.FollowUpControl</i>	在跟进模式处于活动状态时进行设置。 这是指使用“MC_Power”禁用轴或者使用“ <i>Mode = 3</i> ”启用轴的情况。
	<i>(control (取反))</i>	
13	<i>Statusword.ConstantVelocity</i>	在速度设定值仍为常量时进行设置。
	<i>(motionstatedata.motionstate)</i>	
14	<i>Statusword.Accelerating</i>	轴正在加速（增加驱动功率）。
	<i>(motionstatedata.motionstate)</i>	
15	<i>Statusword.Decelerating</i>	轴正在减速（降低驱动功率）。
	<i>(motionstatedata.motionstate)</i>	
16	<i>Statusword.RequestRestart</i>	轴参数已被更改，直到完成下一次重启后才会被激活（“MC_Reset”，其中 <i>Restart = TRUE</i> ）。
17	<i>Statusword.Simulation</i>	轴处于仿真模式。 您可以通过设置 <i>Mode = 4</i> 使用“MC_Power”来实现此目的。
	<i>(simulation)</i>	
18	<i>Statusword.CyclicInterface</i>	在虚拟轴上： 此变量的值始终为 <i>TRUE</i> 。 在电气轴上： 控制器和驱动器之间的循环通信处于活动状态。没有活动的驱动器错误。 在液压轴上： 已在 DP(DRIVE) 上检测到用于控制信号输出的 I/O 和编码器信号，并且这些信号可用于轴工艺对象。 如果多个轴工艺对象是在同一 I/O 地址上组态的，则 <i>Statusword.CyclicInterface</i> 将在所有相关的工艺 DB 上返回 <i>TRUE</i> 。所指定的状态与任何工艺对象对 I/O 的访问无关。
	<i>(actormonitoring.cyclicinterface)</i>	

7.3 “同步轴”工艺 DB

位号	变量 (S7T Config 中的系统变量)	TRUE 状态的意义
19	<i>Statusword.EncoderValid</i>	与轴关联的编码器的实际位置值有效（在绝对值编码器的启动阶段尤为重要）。
	(<i>sensordata.sensordata</i> [当前编码器]. <i>state</i> = <i>VALID</i>)	
20	<i>Statusword.SpeedMode</i>	<i>TRUE</i> = 速度控制模式 <i>FALSE</i> = 位置控制模式 通过设置输入参数 <i>PositionControl</i> = <i>FALSE</i> 在工艺功能“MC_MoveVelocity”上激活速度控制模式。
	(<i>speedmode</i>)	
21	<i>Statusword.TorqueLimiting</i>	轴的扭矩受“MC_SetTorqueLimit”或“MC_MoveToEndPos”命令的限制，而驱动器将运行到此限制。驱动器必须支持扭矩限制（消息帧 10x）。
	(<i>torquelimitecommand.torqueLimitingstate</i>)	
22	<i>Statusword.SuperImpSynchrCommand</i>	叠加同步操作中的轴（活动的“MC_GearInSuperImposed”或“MC_CamInSuperImposed”）
23	<i>Statusword.TorqueLimitingCommand</i>	轴上的减小扭矩命令（“MC_SetTorqueLimit”或“MC_MoveToEndPos”）处于活动状态。
	(<i>torquelimitecommand.state</i>)	
24	<i>Statusword.RequestStartUp</i>	“减小”、“偏移”或“IPO 同步”条目已在 "Technology Objects Management" 中发生了更改。直到再次启动控制器，更改的值才会被激活。
25	<i>Statusword.Todeactivated</i>	工艺对象已取消激活
26	<i>Statusword.HWLimitSwitchMinus</i>	硬件限位开关在运动的负方向上逼近
27	<i>Statusword.HWLimitSwitchPlus</i>	硬件限位开关在运动的正方向上逼近

CommandVelocity 变量

(S7T Config 系统变量 *motionstatedata.commandvelocity*)

显示轴速度的当前设定值规范（已为当前 *SpeedOverride* 值留出余量）。

CommandAcceleration 变量

(S7T Config 系统变量 *motionstatedata.commandacceleration*)

显示轴加速度的当前设定值规范（已为当前 *AccelerationOverride* 值留出余量）。

ActualVelocity 变量

(S7T Config 系统变量 *motionstatedata.actualvelocity*)

该变量将返回轴的当前速度。

ActualAcceleration 变量

(S7T Config 系统变量 *motionstatedata.actualacceleration*)

返回轴的当前加速度。

SpeedOverride 变量

(S7T Config 系统变量 *override.velocity*)

返回当前有效的 **SpeedOverride** 的百分比。该百分比与轴的速度设定值相对应。默认值为 100%。

可以使用 **MC_WriteParameter** 工艺功能在范围 0% 到 200% 之间设置 **SpeedOverride** 值。

AccelerationOverride 变量

(S7T Config 系统变量 *override.acceleration*)

该变量将返回当前有效的 **AccelerationOverride** 值的百分比。该百分比与轴的加速度设定值相对应。默认值 = 100%。

可以使用 **MC_WriteParameter** 工艺功能在范围 1% 到 1,000% 之间设置 **AccelerationOverride** 值。

ActualPosition 变量

(S7T Config 系统变量 *positioningstate.actualposition*)

显示当前轴位置。

CommandPosition 变量

(S7T Config 系统变量 *positioningstate.commandposition*)

表示定位操作的当前位置设定值。

TargetPosition 变量

指明当前命令的目标位置。所指定的值仅适用于活动的定位命令。

Distance 变量

(S7T Config 系统变量 *poscommand.distancetogo*)

显示相对/绝对定位的当前距离。所指定的值仅适用于活动的定位命令。

DecelerationDistance 变量

(S7T Config 系统变量 *poscommand.decelerationdistance*)

该变量将返回轴的当前停止距离。所指定的值仅适用于活动的定位命令。

FollowingError 变量

(S7T Config 系统变量 *servodata.followingerror*)

指明轴的当前跟随误差值。该值仅在执行定位命令时或在同步操作中才有效。

SyncStatus 变量

(S7T Config 系统变量 *<同步对象>.state*)

值	含义	注释
0	没有同步操作。	活动的同步模式为“没有同步操作”。
1	齿轮传动的同步运动	活动的同步模式为“齿轮传动”。
2	曲线同步	活动的同步模式为“凸轮传动”。

Direction 变量

(S7T Config 系统变量 <同步对象>.effectivedata.gearingsettings.direction)

T-CPU 使用 *RatioNumerator* 和 *RatioDenominator* 输入参数计算齿轮传动模式的传动比和传送方向。

齿轮的传送方向由 *Direction* 变量定义。

值	含义	注释
5	正方向	引导轴和跟随轴沿相同的方向移动。
4	负方向	引导轴和跟随轴沿相反的方向移动。

凸轮传动中，不能沿相反方向移动，即在凸轮传动模式下，该值将被设置为 5。

NumGear / DenomGear 变量

(S7T Config 系统变量 <同步对象>.effectivedata.gearingsettings.numerator / <同步对象>.effectivedata.gearingsettings.denominator)

这些变量将返回基础同步的齿轮传动比。*NumGear* 变量将返回引导轴系数，而 *DenomGear* 变量将返回跟随轴系数（与基础同步操作有关）。所指定的值仅适用于基础同步。

示例：

当引导轴和跟随轴之间的齿轮传动比为 17:25 时，*NumGear* 变量将显示值 17，而 *DenomGear* 变量将显示值 25。

PhaseShift 变量

显示引导轴的位置设定值与在基础同步中工作的跟随轴之间的相位偏移。所指定的值仅在基础同步中有效。

7.3 “同步轴”工艺 DB

SupImpSyncStatus 变量

(S7T Config 系统变量 <叠加的同步对象>.state)

值	含义	注释
0	没有同步操作。	叠加的同步对象的活动模式为“不同步”
1	齿轮传动的同步运动	叠加的同步对象的活动模式为“齿轮传动”
2	曲线同步	叠加的同步对象的活动模式为“凸轮传动”

SupImpDirection 变量

(S7T Config 系统变量 <叠加的同步对象>.effectivedata.gearingsettings.direction)

T-CPU 使用 *RatioNumerator* 和 *RatioDenominator* 输入参数计算叠加的齿轮传动的传动比和传送方向。

齿轮的传送方向由 *Direction* 变量定义。

值	含义	注释
5	正方向	引导轴和跟随轴沿相同的方向移动。
4	负方向	引导轴和跟随轴沿相反的方向移动。

叠加的凸轮传动中，不能沿相反方向移动，即，在叠加的凸轮传动模式下，该值将被设置为 5。

SupImpNumGear / SupImpDenomGear 变量

(S7T Config 系统变量 <叠加的同步对象>.effectivedata.gearingsettings.numerator / <叠加的同步对象>.effectivedata.gearingsettings.denominator)

这些变量将返回叠加同步的传动比。*SupImpNumGear* 变量将显示引导轴的系数，而 *SupImpDenomGear* 将显示跟随轴的系数（与跟随轴叠加的坐标系有关）。所指定的值仅适用于活动的叠加同步。

SupImpPhaseShift 变量

该变量表示引导轴的位置设定值和跟随轴的叠加坐标系之间的偏移。该值仅适用于活动的叠加同步。

7.4 “外部编码器”工艺 DB

适用于固件版本为 V4.1.x 的集成工艺

本部分介绍了固件版本为 V3.0.x、V3.1.x 和 V3.2.x 的应用。

集成工艺将数据写入 DB。该 DB 在 STEP 7 用户程序中为只读。

无论 CPU 中的数据保持设置如何，该 DB 都不具有保持性。

UpdateFlag/UpdateCounter 变量

请注意：更新工艺 DB (页 829)

ErrorID 变量

显示外部编码器上最近检测到的错误或警告的 ErrorID。

错误条目可以通过调用工艺功能“MC_Reset”进行确认（其中，*Axis* = 工艺 DB 的编号）。

请注意：可能出现的错误消息和警告 (页 857)

ErrorBuffer[0..2] 变量

收到的前三条错误和警告的存储器。第一个错误写到 ARRAY 元素 0，第二个错误写到 ARRAY 元素 1，依次类推。

此存储器中的内容可以通过调用工艺功能“MC_Reset”进行清除（其中，*Axis* = 工艺 DB 的编号）。

TO_Adaption 变量

此变量是为内部功能保留的。它不包含任何与用户相关的信息。

ErrorStatus.xxx 变量

变量 *ErrorStatus.xxx* 中的位将返回有关外部编码器上的各种错误的信息。

位号	变量	<i>TRUE</i> 状态的意义
0	<i>ErrorStatus.SystemFault</i>	内部系统错误
1	<i>ErrorStatus.ConfigFault</i>	工艺对象组态错误
2	<i>ErrorStatus.UserFault</i>	用户程序由于输出无效的命令而引发错误
3	<i>ErrorStatus.FaultDrive</i>	外部编码器或工艺对象报告错误
4	<i>ErrorStatus.Reserve4</i>	未使用
5	<i>ErrorStatus.FollowingWarning</i>	动态跟随误差监视功能报告超出警告限制
6	<i>ErrorStatus.FollowingError</i>	动态跟随误差监视功能报告超出窗口限制
7	<i>ErrorStatus.StandstillFault</i>	外部编码器已超出停止窗口，或者在指定的时间内没有到达停止窗口。
8	<i>ErrorStatus.PositioningError</i>	轴无法在指定的时间内到达定位窗口。
9	<i>ErrorStatus.SynchronOpError</i>	未使用
10	<i>ErrorStatus.DynamicError</i>	已超出动态响应限制
11	<i>ErrorStatus.ClampingError</i>	紧固时出错 未收到新的运动命令，轴就已超出“固定挡块检测后的位置公差”。 位置公差在 S7T Config 中的“限制”(Limits) >“固定挡块”(Fixed end stop) 选项卡 >“固定挡块检测后的位置公差”(Position tolerance after fixed end stop detection) 参数中设置。
12	<i>ErrorStatus.SoftwareLimitPos</i>	未使用
13	<i>ErrorStatus.SoftwareLimitNeg</i>	未使用
14	<i>ErrorStatus.LimitSwitchActive</i>	未使用
15	<i>ErrorStatus.SensorFreqViolation</i>	已超出编码器限制频率。
16	<i>ErrorStatus.ReferenceNotFound</i>	回原点时没找到参考凸轮或零标记脉冲
17	<i>ErrorStatus.ZeroMonitoring</i>	零标记脉冲监视已检测到错误（不是回原点错误）
18	<i>ErrorStatus.Overspeed</i>	未使用
19	<i>ErrorStatus.FollowObjectError</i>	未使用
20	<i>ErrorStatus.SupImpFollowObjectError</i>	未使用

Statusword.xxx 变量

Statusword 变量表示外部编码器的当前状态。

如果禁用外部编码器，则这些值将不再更新。所显示的值对应复位使能信号时注册的最终状态。

位号	变量 (S7T Config 中的系统变量)	TRUE 状态的意义
0	<i>Statusword.DriveEnabled</i>	外部编码器已启用（输入参数“MC_ExternalEncoder”中 <i>Mode = 1</i> ）
	<i>(control)</i>	
1	<i>Statusword.HomingDone</i>	轴已回原点，即，“MC_ExternalEncoder”已成功地在 <i>Mode= 2</i> 到 <i>6</i> 中完成
	<i>(syncstate)</i>	
2	<i>Statusword.Done</i>	没有任何作业在执行中。
3	<i>Statusword.SuperImposedCommand</i>	与外部编码器无关。
4	<i>Statusword.Error</i>	已出现了至少一个错误（变量 <i>ErrorID = 8xxx</i> ）。出现警告时，将保持 <i>Statusword.Error = FALSE</i> 状态。
5	<i>Statusword.Errorstop</i>	由于出现错误，外部编码器将被/已被锁定。 消除错误原因并确认错误。
	<i>(errorreaction <> NONE)</i>	
6	<i>Statusword.Stopping</i>	与外部编码器无关。
7	<i>Statusword.Standstill</i>	外部编码器上的速度（绝对值）小于设定的停止限制。
	<i>(motionstate.stillstandvelocity)</i>	
8	<i>Statusword.PositioningCommand</i>	与外部编码器无关。
9	<i>Statusword.SpeedCommand</i>	与外部编码器无关。
10	<i>Statusword.SynchrCommand</i>	与外部编码器无关。
11	<i>Statusword.Homing</i>	在回原点开始时置位该位，在结束时复位该位 （“MC_ExternalEncoder” <i>Mode = 2</i> 到 <i>6</i> ）
	<i>(synccommand.state)</i>	
12	<i>Statusword.FollowUpControl</i>	与外部编码器无关。
13	<i>Statusword.ConstantVelocity</i>	与外部编码器无关。
14	<i>Statusword.Accelerating</i>	与外部编码器无关。
15	<i>Statusword.Decelerating</i>	与外部编码器无关。

7.4 “外部编码器”工艺 DB

位号	变量 (S7T Config 中的系统变量)	TRUE 状态的意义
16	<i>Statusword.RequestRestart</i>	外部编码器参数已被更改，直到下一次重启完成后才会被激活（“MC_Reset”，其中 <i>Restart</i> = <i>TRUE</i> ）。
17	<i>Statusword.Simulation</i>	与外部编码器无关。
18	<i>Statusword.CyclicInterface</i> (<i>sensormonitoring.cyclicInterface</i>)	控制器和编码器之间的循环通信处于活动状态。
19	<i>Statusword.EncoderValid</i> (<i>ExternalEncoder.sensordata.state</i> = <i>VALID</i>)	外部编码器的实际位置值有效（在绝对值编码器的启动阶段尤为重要）。
20	<i>Statusword.SpeedMode</i>	对于外部编码器始终为 <i>FALSE</i> 。
21	<i>Statusword.TorqueLimiting</i>	与外部编码器无关。
22	<i>Statusword.SupImpSynchrCommand</i>	与外部编码器无关。
23	<i>Statusword.TorqueLimitingCommand</i>	轴上的减小扭矩命令（“MC_SetTorqueLimit”或“MC_MoveToEndPos”）处于活动状态。
24	<i>Statusword.RequestStartUp</i>	“减小”、“偏移”或“IPO 同步”条目已在 "Technology Objects Management" 中发生了更改。直到再次启动控制器，更改的值才会被激活。
25	<i>Statusword.ToDeactivated</i>	工艺对象已取消激活

ActualPosition 变量(S7T Config 系统变量 *motionstate.position*)

显示外部编码器的当前位置。

如果禁用外部编码器，则该位置将不再更新。所指定的位置对应复位使能信号前的最终位置。

ActualVelocity 变量(S7T Config 系统变量 *motionstate.velocity*)

显示外部编码器的当前速度。

如果禁用外部编码器，则该速度值将不再更新。所指定的速度对应复位使能信号前的最终速度。

7.5 “凸轮盘”工艺 DB

适用于固件版本为 V4.1.x 的集成工艺

本部分介绍了固件版本为 V3.0.x、V3.1.x 和 V3.2.x 的应用。

集成工艺将数据写入 DB。该 DB 在 STEP 7 用户程序中为只读。

无论 CPU 中的数据保持设置如何，该 DB 都不具有保持性。

ErrorID 变量

显示凸轮上最近检测到的错误或警告的 ErrorID。

错误条目可以通过调用工艺功能“MC_Reset”进行确认（其中，*Axis* = 工艺 DB 的编号）。

请注意：可能出现的错误消息和警告 (页 864)

ErrorBuffer[0..2] 变量

收到的前三条错误和警告的存储器。第一个错误写到 ARRAY 元素 0，第二个错误写到 ARRAY 元素 1，依次类推。

此存储器中的内容可以通过调用工艺功能“MC_Reset”进行清除（其中，*Axis* = 工艺 DB 的编号）。

Status 变量

值	含义	注释
0	空白	该凸轮不包含任何插补点或区段。
1	插入的插补点/区段	该凸轮包含插补点/区段。
2	已插补	已对该凸轮进行了插补，这表明它已准备好进行同步操作。
3	错误	出现错误。

UserCount 变量

该变量将返回当前正在使用该凸轮的工艺对象的数目。

Statusword.xxx 变量

变量 *Statusword* 指明凸轮盘的当前状态。

位号	变量	<i>TRUE</i> 状态的意义
0	<i>Statusword.Error</i>	已出现了至少一个错误（变量 <i>ErrorID</i> = 8xxx）。出现警告时，将保持 <i>Statusword.Error</i> = <i>FALSE</i> 状态。
1	<i>Statusword.RequestRestart</i>	凸轮参数已被更改，直到下一次重启后才会被激活（工艺功能“MC_Reset”，其中 <i>Restart</i> = <i>TRUE</i> ）。
2	<i>Statusword.RequestStartUp</i>	“减小”、“偏移”或“IPO 同步”条目已在 "Technology Objects Management" 中发生了更改。直到再次启动控制器，更改的值才会被激活。
3	<i>Statusword.ToDeactivated</i>	工艺对象已取消激活

TO_Adaption 变量

此变量是为内部功能保留的。它不包含任何与用户相关的信息。

7.6 “测量输入”工艺 DB

适用于固件版本为 V4.1.x 的集成工艺

本部分介绍了固件版本为 V3.0.x、V3.1.x 和 V3.2.x 的应用。

“测量输入”工艺 DB 包含与“测量输入”工艺对象和命令执行相关的信息。集成工艺将数据写入 DB。该 DB 在 STEP 7 用户程序中为只读。

无论 CPU 中的数据保持设置如何，该 DB 都不具有保持性。

UpdateFlag / UpdateCounter 变量

请注意：更新工艺 DB (页 829)

ErrorID 变量

该变量将返回在测量输入上最近检测到的错误或警告的 ErrorID。

错误条目可以通过调用工艺功能“MC_Reset”进行确认（其中，*Axis* = 工艺 DB 的编号）。

请注意：可能出现的错误消息和警告 (页 428)

ErrorBuffer[0..2] 变量

收到的前三条错误和警告的存储器。第一个错误写到 ARRAY 元素 0，第二个错误写到 ARRAY 元素 1，依次类推。

此存储器中的内容可以通过调用工艺功能“MC_Reset”进行清除（其中，*Axis* = 工艺 DB 的编号）。

Status 变量

表示测量输入的操作模式。可以显示以下操作模式：

值	含义	注释
0	初始	空闲状态下的测量输入
1	等待触发	等待测量输入的沿（尚未记录测量值）
2	数据有效	测量完成，已记录值并且值有效
3	错误	检测到错误；值无效

7.6 “测量输入”工艺 DB

MeasureValue1 变量

(S7T Config 系统变量 *measuredvalue1*)

返回测量输入的“测量值 1”。如果状态 = 2，则该值无效（操作状态）。

MeasureValue2 变量

(S7T Config 系统变量 *measuredvalue2*)

返回测量输入的“测量值 2”。如果状态 = 2，则该值无效（操作状态）。

“测量值 2”仅记录在工艺功能“MC_MeasuringInput”的模式（*Mode 3* 和 *4*）中。

MeasureStart / MeasureEnd 变量

(S7T Config 系统变量 *effectivedata.measuringrangestartposition*
/effectivedata.measuringrangeendposition)

MeasureStart 变量将返回测量区域的下限值，而 *MeasureEnd* 变量将测量其上限值。将仅记录在这些限制值之内的测量值。当 *MeasureStart* 值等于 *MeasureEnd* 值时，测量区域限制值将被忽略。

DB 上显示的 *MeasureStart / MeasureEnd* 变量对应相关的工艺功能“MC_MeasuringInput”的输入参数的值。

Statusword.xxx 变量

变量 *Statusword* 表示测量输入的当前状态。

位号	变量	TRUE 状态的意义
0	<i>Statusword.Error</i>	已出现了至少一个错误（变量 <i>ErrorID</i> = 8xxx）。出现警告时，将保持 <i>Statusword.Error</i> = FALSE 状态。
1	<i>Statusword.RequestRestart</i>	测量输入参数已被更改，直到下一次重启后才会被激活（工艺功能“MC_Reset”，其中 <i>Restart</i> = TRUE）。
2	<i>Statusword.RequestStartUp</i>	“减小”、“偏移”或“IPO 同步”条目已在 "Technology Objects Management" 中发生了更改。直到再次启动控制器，更改的值才会被激活。
3	<i>Statusword.ToDeactivated</i>	工艺对象已取消激活

Reservexx 变量

变量 *Reserve02* 到 *Reserve10* 是为内部功能保留的。它们不包含任何与用户相关的信息。

TO_Adaption 变量

此变量是为内部功能保留的。它不包含任何与用户相关的信息。

7.7 “输出凸轮”工艺 DB

适用于固件版本为 V4.1.x 的集成工艺

本部分介绍了固件版本为 V3.0.x、V3.1.x 和 V3.2.x 的应用。

集成工艺将数据写入 DB。该 DB 在 STEP 7 用户程序中为只读。

无论 CPU 中的数据保持设置如何，该 DB 都不具有保持性。

ErrorID 变量

该变量将返回在输出凸轮上最近检测到的错误或警告的 ErrorID。

错误条目可以通过调用工艺功能“MC_Reset”进行确认（其中，Axis = 工艺 DB 的编号）。

请注意：可能出现的错误消息和警告 (页 873)

ErrorBuffer[0 到 2] 变量

收到的前三条错误和警告的存储器。第一个错误写到 ARRAY 元素 0，第二个错误写到 ARRAY 元素 1，依次类推。

此存储器中的内容可以通过调用工艺功能“MC_Reset”进行清除（其中，Axis = 工艺 DB 的编号）。

State 变量

(S7T Config 系统变量 state)

值	含义	注释
0	关	输出凸轮的当前活动状态为“关”。
1	开	凸轮的当前活动状态为“开”。

CamType 变量

值	含义	注释
0	基于位置的凸轮	输出凸轮作为基于位置的凸轮运行。
1	基于时间的凸轮	输出凸轮作为基于时间的凸轮运行。
2	单方向的输出凸轮	输出凸轮作为单方向的输出凸轮运行。

Mode 变量

值	含义	注释
1	禁用	输出凸轮的当前运行状态为“禁用凸轮”(Cam disabled)
2	标准	输出凸轮的当前运行状态为“启用凸轮（输出尚未反向）”(Cam enabled [output not inverted])
3	反向	输出凸轮的当前运行状态为“启用凸轮（输出已反向）”(Cam enabled [output inverted])
4	常开	输出凸轮的当前运行状态为“始终使用凸轮”(Cam always on)。

Direction 变量

（S7T Config 系统变量 *effectivedata.forcedirection*）

该变量将返回最近的 MC_CamSwitch 或 MC_CamSwitchTime 命令的有效输出凸轮方向。该变量在以下“MC_CamSwitch”和“MC_CamSwitchTime”模式中不会发生更改：

- *Mode = 1*（禁用输出凸轮）
- *Mode = 4*（始终使用输出凸轮）

值	含义
1	正有效方向
2	正负有效方向，意味着有效方向是不相关的
3	负有效方向
4	轴的当前旋转方向为有效方向

7.7 “输出凸轮”工艺 DB

OnPosition 变量

(S7T Config 系统变量 *effectivedata.switchonposition*)

该变量将返回当前起始位置。

OffPosition 变量

(S7T Config 系统变量 *effectivedata.switchoffposition*)

该变量将返回当前结束位置。

Duration 变量

(S7T Config 系统变量 *effectivedata.onduration*)

该变量将返回基于时间的凸轮的脉冲宽度。该物理单元对应相应轴上的“时间”物理单元。

Delay 变量

(S7T Config 系统变量 *effectivedata.activationtime / effectivedata.deactivationtime*)

表示凸轮激活/取消激活时间的当前偏移（微分作用时间）。

值 < 0: 提前激活

值 > 0: 延缓激活

Hysteresis 变量

(S7T Config 系统变量 *effectivedata.noswitchingrange*)

该变量将返回滞后的实际值。

Statusword.xxx 变量

变量 *Statusword* 表示输出凸轮的当前状态。

位号	变量	<i>TRUE</i> 状态的意义
0	<i>Statusword.Error</i>	已出现了至少一个错误（变量 <i>ErrorID</i> = 8xxx）。出现警告时，将保持 <i>Statusword.Error</i> = <i>FALSE</i> 状态。
1	<i>Statusword.RequestRestart</i>	输出凸轮参数已被更改，直到下一次重启后才会被激活（工艺功能“MC_Reset”，其中 <i>Restart</i> = <i>TRUE</i> ）。
2	<i>Statusword.RequestStartUp</i>	“减小”、“偏移”或“IPO 同步”条目已在 "Technology Objects Management" 中发生了更改。直到再次启动控制器，更改的值才会被激活。
3	<i>Statusword.ToDeactivated</i>	工艺对象已取消激活

Reservexx 变量

变量 *Reserve02* 到 *Reserve10* 是为内部功能保留的。它们不包含任何与用户相关的信息。

TO_Adaption 变量

此变量是为内部功能保留的。它不包含任何与用户相关的信息。

7.8 “凸轮轨迹”工艺 DB

适用于固件版本为 V4.1.x 的集成工艺

集成工艺将数据写入 DB。该 DB 在 STEP 7 用户程序中为只读。

无论 CPU 中的数据保持设置如何，该 DB 都不具有保持性。

UpdateFlag / UpdateCounter 变量

请注意：更新工艺 DB (页 829)

ErrorID 变量

返回凸轮轨迹上最新检测到的错误或警告的 ErrorID。

错误条目可以通过调用工艺功能“MC_Reset”进行确认（其中，*Axis* = 工艺 DB 的编号）。

请注意：可能出现的错误消息和警告 (页 878)

ErrorBuffer[0 到 2] 变量

收到的前三条错误和警告的存储器。第一个错误写到 ARRAY 元素 0，第二个错误写到 ARRAY 元素 1，依次类推。

此存储器中的内容可以通过调用工艺功能“MC_Reset”进行清除（其中，*Axis* = 工艺 DB 的编号）。

TO_Adaption 变量

此变量是为内部功能保留的。它不包含任何与用户相关的信息。

Statusword.xxx 变量

变量 *Statusword* 指明凸轮轨迹的当前状态。

位号	变量	TRUE 状态的意义
0	<i>Statusword.Error</i>	已出现了至少一个错误（变量 <i>ErrorID</i> = 8xxx）。出现警告时，将保持 <i>Statusword.Error</i> = FALSE 状态。
1	<i>Statusword.RequestRestart</i>	凸轮轨迹参数已被更改，直到下一次重启后才会被激活（工艺功能“MC_Reset”，其中 <i>Restart</i> = TRUE）。
2	<i>Statusword.RequestStartUp</i>	“减小”、“偏移”或“IPO 同步”条目已在 "Technology Objects Management" 中发生了更改。直到再次启动控制器，更改的值才会被激活。
3	<i>Statusword.TOdeactivated</i>	工艺对象已取消激活

State 变量

（S7T Config 系统变量 *effectivedata.state*）

该变量表示输出凸轮输出的当前切换状态。

值	含义	注释
0	关	输出凸轮输出的当前活动状态为“关”。
1	开	输出凸轮输出的当前活动状态为“开”。

CamTrackType 变量

（S7T Config 组态数据 *OctType.Camtracktype*）

该变量指明凸轮轨迹的凸轮类型。

值	含义	注释
0	基于位置的凸轮	凸轮轨迹的专用输出凸轮作为基于位置的凸轮运行。
1	基于时间的凸轮	凸轮轨迹的专用输出凸轮作为基于时间的凸轮运行。
2	具有最大 ON 时间的基于时间的输出凸轮	凸轮轨迹的专用输出凸轮作为基于时间的凸轮运行。如果超出了设定的最大 ON 长度，则会过早中止专用输出凸轮。

Control 变量(S7T Config 系统变量 *control*)

该变量指明凸轮轨迹的内部状态。

值	含义	注释
1	ACTIVE	凸轮轨迹处于活动状态
2	INACTIVE	凸轮轨迹处于非活动状态
3	ACTIVE_AND_WAITING _FOR_CAM_TRACK_ OUTPUT_INACTIVE	应用到“MC_CamTrack”工艺功能的 <i>CommandMode = 3</i> 输入参数。 - 凸轮轨迹已启用： 凸轮轨迹命令仍处于活动状态。当输出凸轮输出或所有专用输出凸轮处于非活动状态时，新的凸轮轨迹命令变为活动状态。 - 禁用凸轮轨迹： 凸轮轨迹命令将保持活动状态，直到输出凸轮输出或所有专用输出凸轮处于非活动状态。
4	ACTIVE_AND_WAITING _FOR_NEXT_CYCLE	应用到“MC_CamTrack”工艺功能的 <i>CommandMode = 2</i> 输入参数。 取消激活凸轮轨迹： 在进入凸轮轨迹周期及其循环期间，将取消激活凸轮轨迹。
5	INACTIVE_AND_ WAITING_FOR_NEXT_ CYCLE	应用到“MC_CamTrack”工艺功能的 <i>CommandMode = 2</i> 输入参数。 启用凸轮轨迹： 在进入凸轮轨迹周期及其循环期间，将激活凸轮轨迹。

CyclicMode 变量

该变量指明凸轮轨迹是否处于循环模式。

值	含义	注释
1	循环	循环使用凸轮轨迹。
2	非循环	不循环使用凸轮轨迹。

InvertOutput 变量

(S7T Config 组态数据 *OctTechnologicalCfg.invertOutput*)

该变量指明凸轮轨迹是否处于取反模式。

值	含义	注释
1	未取反	输出凸轮输出和专用输出凸轮未取反
2	取反	对输出凸轮输出和专用输出凸轮取反

ReferencePosition 变量

(S7T Config 系统变量 *effectivedata.axisreferenceposition*)

该变量显示轴上凸轮轨迹位置 0 的参考位置。

CamTrackLength 变量

(S7T Config 系统变量 *effectivedata.camtracklength*)

该变量返回活动的凸轮轨迹长度。

ActivationTime 变量

(S7T Config 系统变量 *effectivedata.activationtime*)

该变量显示激活期间的有效延迟或动作时间。

DeactivationTime 变量

(S7T Config 系统变量 *effectivedata.deactivationtime*)

该变量显示取消激活期间的有效延迟或动作时间。

7.8 “凸轮轨迹”工艺 DB

Hysteresis 变量

(S7T Config 系统变量 *effectivedata.hysteresisrange*)

该变量将返回滞后的实际值。

CamTrackPosition 变量

(S7T Config 系统变量 *effectiveData.camtrackposition*)

该变量指明凸轮轨迹的相对轴位置。

SingleCamState[0 ... 31] 变量

(S7T Config 系统变量 *singlecamstate*)

该变量指明专用输出凸轮 0 到 31 的激活状态。

值	含义	注释
<i>FALSE</i>	专用输出凸轮未处于活动状态	专用输出凸轮未处于活动状态。切换为无效的专用输出凸轮还将显示 <i>FALSE</i> 。
<i>TRUE</i>	专用输出凸轮处于活动状态	专用输出凸轮处于活动状态。

7.9 “跟踪”工艺 DB

适用于固件版本为 V3.1.x 或更高版本的集成工艺

本部分介绍了固件版本为 V3.0.x 的应用。

您可以使用 S7T Config 中的 TraceTool 以图形实时格式来记录集成工艺的变量。“跟踪”DB 连接了 TraceTool 和用户程序。

可将 2 个数据类型为 DINT 的值、2 个数据类型为 DWORD 的值以及 4 个数据类型为 REAL 的值从应用程序中写入跟踪 DB 的各个变量中，以便进行跟踪。用户程序不能覆盖 *ErrorID* 变量。

集成工艺在工艺 DB 更新周期内读取 DB 数据。可以使用目标系统 (Target system) > 设置系统时钟 (Set system clocks) 命令在 S7T Config 中设置工艺 DB 更新周期。

无论 CPU 中的数据保持设置如何，该 DB 都不具有保持性。

ErrorID 变量

返回最近检测到的错误的 *ErrorID*。

必须消除错误的原因。无法确认错误。

请注意：可能出现的错误消息和警告 (页 882)

S7_TraceDINT[0..1] 变量

S7_TraceDINT 变量包含具有两个数据类型为 DINT 的元素的 ARRAY。

当在 S7T Config 的 TraceTool 中的工艺 (*Technology*) > *userdata1* > *user1* 和 *user2* 中选择信号时，可以选择这些值来进行跟踪。

S7_TraceDWORD[0..1] 变量

S7_TraceDWORD 变量包含具有两个数据类型为 DWORD 的元素的 ARRAY。如果您要使用 S7T 跟踪工具跟踪位信息（例如，状态字或错误字），则可以使用这些变量。您可以使用跟踪工具选择以常规 SIMATIC 编号表示的位。

当在 S7T Config 的 TraceTool 中的工艺 (*Technology*) > *userdata1* > *user3* 和 *user4* 中选择信号时，可以选择这些值来进行跟踪。

S7_TraceREAL[0..3] 变量

S7_TraceREAL 变量包含具有四个数据类型为 REAL 的元素的 ARRAY。

当在 S7T Config 的 TraceTool 中的工艺 (*Technology*) > *userdata1* > *user5*, *user6*, *user7* 和 *user8* 中选择信号时，可以选择这些值来进行跟踪。

7.10 “MCDevice”工艺 DB

适用于固件版本为 V3.2.x 或更高版本的集成工艺

本部分介绍了固件版本为 V3.0.x 和 V3.1.x 的应用。

“MCDevice”工艺 DB 包含集成工艺的常规信息。可在相应的 DB 中找到有关各工艺对象的信息。

集成工艺将数据写入 DB。该 DB 在 STEP 7 用户程序中为只读。

无论 CPU 中的数据保持设置如何，该 DB 都不具有保持性。

UpdateFlag / UpdateCounter 变量

请注意：更新工艺 DB

ErrorID 变量

返回最近检测到的错误的 *ErrorID*。必须消除错误的原因。无法确认错误。

请注意：可能出现的错误消息和警告 (页 882)

ErrorBuffer[0..2] 变量

收到的前三条错误和警告的存储器。第一个错误写到 *ARRAY* 元素 0，第二个错误写到 *ARRAY* 元素 1，依次类推。

此存储器中的内容可以通过调用工艺功能“MC_Reset”进行清除（其中，*Axis* = 工艺 DB 的编号）。

Reserve34 变量

此变量是为内部功能保留的。它不包含任何与用户相关的信息。

MaxLoopDuration 变量

将返回集成工艺的最大命令执行时间。

将在 T-CPU 启动期间或调用 FB“MC_Reset”（其中 *Axis* = *MCDevice-DB*; *Restart* = *TRUE* 或 *FALSE*）时删除该条目。

CmdLoopDuration 变量

该变量将返回集成工艺的命令执行时间的平均值（以秒为单位）。

将在 T-CPU 启动期间或调用 FB“MC_Reset”（其中 *Axis = MCDevice-DB; Restart = TRUE* 或 *FALSE*）时删除该条目。

StationLifeList[0..127] 变量

该变量的 ARRAY 将列出在 PROFIBUS DP(DRIVE) 上可用的所有伙伴站。ARRAY 元素的编号对应 PROFIBUS 地址。元素中的值 *1* 表示某伙伴站可用，而值 *0* 表示该站不可用。

DoneFlag[1..32] 变量

可将 ARRAY 元素 1 到 32 分配给功能块的 *DoneFlag* 输入参数。

FB 上 <i>DoneFlag</i> 的值	所用的 ARRAY 元素	
0		没有生成 <i>DoneFlag</i>
1	1	生成了 <i>DoneFlag</i> [1]
...	...	...
32	32	生成了 <i>DoneFlag</i> [32]

每次完成相关的工艺功能后，*DoneFlag* 中的值都会变为相反值（*FALSE* 变成 *TRUE*，而 *TRUE* 变成 *FALSE*）。只有可以监视初始命令后才能设置 *DoneFlag*。如果后续命令使用同一实例，则无法继续监视此命令。

DIStatus[0..3] 变量

ARRAY 元素 0 到 3 返回 CPU 31xT 的集成数字输入的状态。

DOStatus[0..7] 变量

ARRAY 元素 0 到 7 返回 T-CPU 的集成数字输出的状态。例如，可将集成工艺的输出生配给 S7T Config 中的各个输出凸轮。

Statusword.xxx 变量

变量 *Statusword* 指明同步轴的当前状态。

位号	变量	TRUE 状态的意义
0	<i>Reserve01</i>	此变量是为内部功能保留的。它不包含任何与用户相关的信息。
1	<i>Reserve02</i>	此变量是为内部功能保留的。它不包含任何与用户相关的信息。
2	<i>Statusword.RequestStartUp</i>	“减小”、“偏移”或“IPO 同步”条目已在 "Technology Objects Management" 中发生了更改。直到再次启动控制器，更改的值才会被激活。

TODBTaskOverflows 变量

显示工艺 DB 更新溢出的次数。返回的最大值为 *FFFFFFFF*。

将在 T-CPU 启动期间或调用工艺功能“MC_Reset”（其中 *Axis = MCDevice-DB; Restart = TRUE* 或 *FALSE*）时删除该条目。

说明

您也可以通过选择**目标系统 (Target system) > 设置系统时钟 (Set system clocks)** 命令来为 S7T Config 中的工艺 DB 更新设置基础时钟。

时间设置过短可导致在工艺 DB 更新期间溢出。在运行固件版本为 V3.2x 或更高版本的集成工艺时，CPU 将不再进入 STOP 状态。

如果活动的工艺 DB 更新尚未完成，就会发生“溢出”，从而防止新更新周期开始。新的更新命令将被放弃。

可以从 *TODBTaskOverflows* 变量中读取溢出的次数，从而可以了解工艺 DB 更新失败的次数。

Reservexx 变量

变量 *Reserve02* 到 *Reserve10* 是为内部功能保留的。它们不包含任何与用户相关的信息。

TO_Adaption 变量

此变量是为内部功能保留的。它不包含任何与用户相关的信息。

7.11 更新工艺 DB

适用于固件版本为 V3.2.x 或更高版本的集成工艺

本部分介绍了固件版本为 V3.0.x 和 V3.1.x 的应用。

集成工艺将工艺 DB 相对循环用户程序进行异步更新。可以通过选择**目标系统 (Target system) > 设置系统时钟 (Set system clocks)** 命令在 S7T Config 中设置工艺 DB 更新周期。可以将更新周期设为插补器周期的倍数。

固件版本为 V3.2.x 的集成工艺支持将工艺 DB 更新周期分配给不同的系统时钟（除工艺 DB“跟踪”之外）。工艺 DB 更新周期代表基本的更新周期。

说明

您可以通过选择**目标系统 (Target system) > 设置系统时钟 (Set system clocks)** 命令来在 S7T Config 中定义更新的基础时钟。

如果在上一更新周期未完成时即启动一个新的工艺 DB 更新，就会发生“溢出”。当使用固件版本为 V3.1.x 或更低版本的集成工艺运行时，T-CPU 将在检测到工艺 DB 更新溢出时跳转到 STOP。固件版本为 V3.2.x 或更高版本的集成工艺容许此类溢出，这意味着将放弃不可执行的更新，并且不会调用工艺同步中断 OB 65。

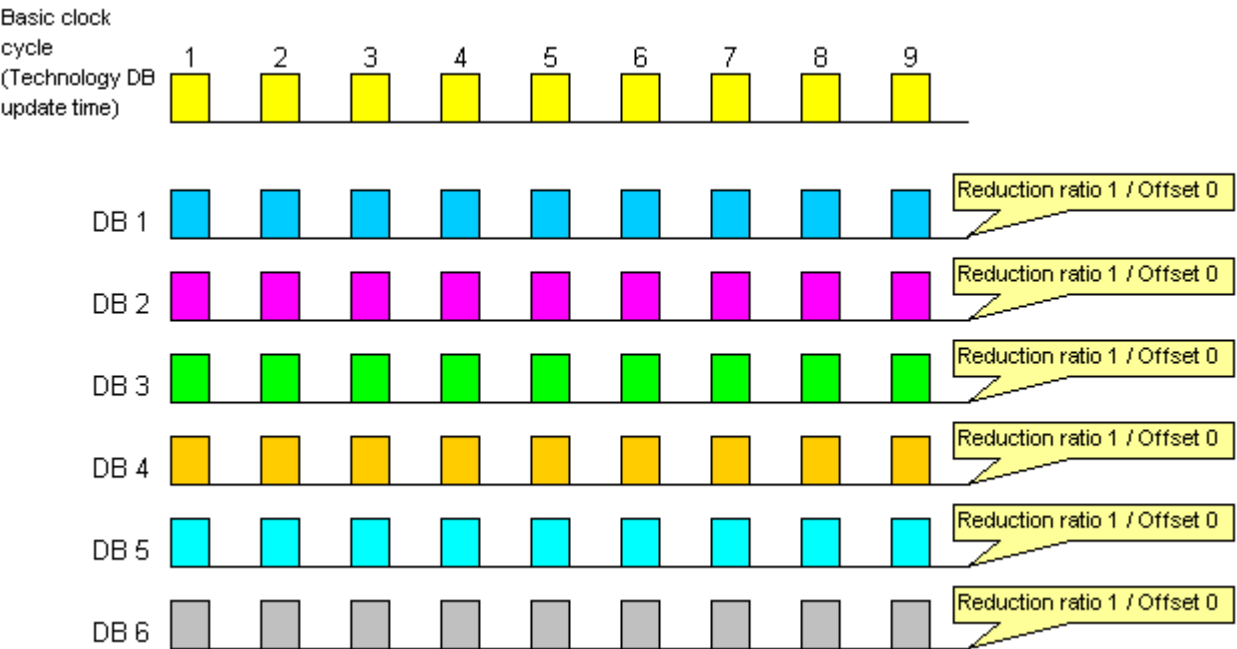
工艺 DB“MCDevice”上的 *TODBTaskOverflows* 变量中会显示溢出的次数，从而了解工艺 DB 更新的失败率。

关于该基础时钟，您可以对“工艺对象管理”中的减速比和偏移进行编程，从而将各工艺 DB 的更新分配给多个基础时钟。通过选择**查看 (View) > 专家模式 (Expert mode)** 命令激活专家模式。

Technology DBs not yet created (Defaults list)

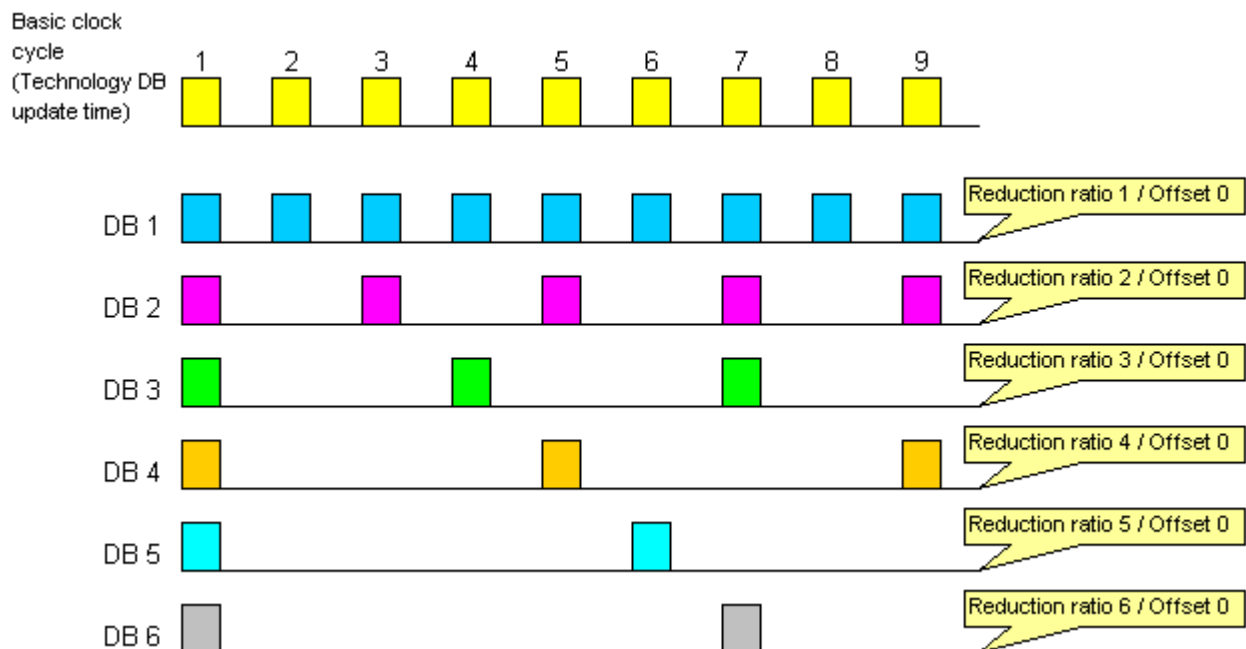
	IPO synchronous	Reduction ratio	[ms]	Offset	[ms]	DB	Symbol
	☐	1	18.000	0	0.000	DB3	Axis_1
	☐	1	18.000	0	0.000	DB4	Axis_2
	☐	1	18.000	0	0.000	DB5	Axis_3
						DB1	Trace
		1	18.000	0	0.000	DB2	MC device

使用默认设置更新工艺 DB



默认情况下，将工艺 DB 的减少因子设为 1，偏移设为 0。所有工艺 DB 都在基础时钟周期（工艺 DB 更新周期）内更新。所选的工艺 DB 更新的基础时钟周期必须足够长，以便可以更新所有工艺 DB。

使用分级的减速比和统一偏移更新工艺 DB

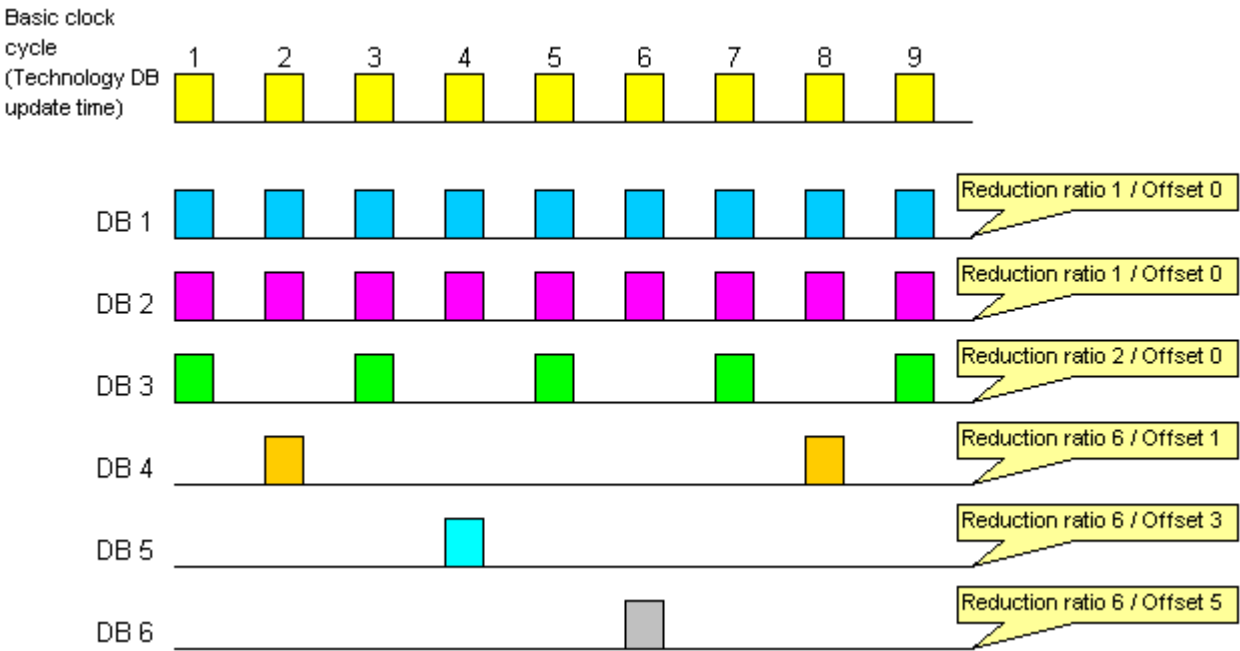


减速比决定要从中更新工艺 DB 的基础时钟。

- 减速比为 1，意味着在每个基础周期时钟上更新工艺 DB。
- 减速比为 2，意味着每隔一个基础周期时钟更新工艺 DB，依此类推。

与默认设置相比，该设置可以提高工艺 DB 更新的分配。但是，基础周期时钟 1、5、7 和 9 仍将产生较高的系统负载。

使用分级的减速比和偏移进行更新的示例



该示例表明分级的减速比和偏移可用来将更新负载限制为每个基础时钟 3 个工艺 DB。与先前给出的示例相比，该示例允许将基础时钟加倍。并将 DB1 和 DB2 的更新时间减半。DB3 的更新时间保持不变。将 DB3 到 DB6 的更新时间增至三倍。

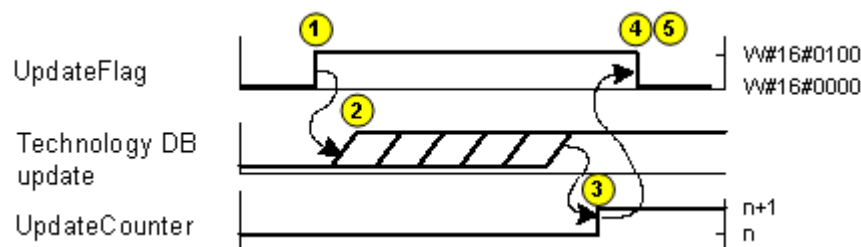
说明

使用下面的信息以优化减速比和偏移：

- 将周期宽度的大小设为至少可在用户程序中进行评估。
- 为您要统一判断的工艺 DB 选择相同的减速比和偏移
- 创建一个更新计划图来显示更新的周期分配
- 工艺 DB 更新所需的时间取决于 DB 长度。与长度较短的工艺 DB 的周期分配相比，长度较长的工艺 DB（例如，轴工艺 DB）的周期的负载分配可能更为平均。

"UpdateFlag"、"UpdateCounter"和工艺同步中断 OB (OB65)

下图显示了工艺 DB 的"UpdateFlag"和"UpdateCounter"的顺序。



1	通过将"UpdateFlag"从 <i>W#16#0000</i> 过渡到 <i>W#16#0100</i> . 来启动工艺 DB 更新。
2	将更新所有受该更新周期影响的工艺 DB。集成工艺始终写入整个工艺 DB。
3	"UpdateCounter"会将其值增加计数 1。所有与该更新周期相关的"UpdateCounter"值都将增加。
4	对于所有相关的工艺 DB, "UpdateFlag"都将从 <i>W#16#0100</i> 更改为 <i>W#16#0000</i> 。
5	调用工艺同步中断 OB65。有关局部数据的详细信息 (OB65 中的启动信息), 请参考"STEP 7 在线帮助"。可以将 OB 65 从"Standard Library"的"Organization Blocks"文件夹复制到项目中。

说明

注意事项:

- 无论是否在某基础时钟内更新了工艺 DB, 都会在每个基础时钟内调用工艺同步中断
- 工艺 DB 更新和工艺功能的反馈信号的生成是异步处理的, 因此数据不一致
调用了 *Done* 输出参数后, 请等到"UpdateCounter"增加后, 再分析工艺数据块 (例如, 用于确定位置)。

使用工艺同步中断 OB 65

统一评估工艺 DB

由于工艺同步中断 OB 是在每次更新了工艺 DB 之后调用的, 因此在调用该 OB 时, 相关的实际值将可用。该功能使您可以与集成工艺同步评估工艺 DB。因此建议使用工艺同步中断 OB 以确保对工艺 DB 数据进行统一评估。

如果用户程序包含较长的工艺同步中断 OB, 而工艺 DB 更新周期较短, 则可以在工艺同步中断 OB 终止前开始新的工艺 DB 更新周期。刚读取的数据将立刻被再次覆盖。

使用工艺 DB 的"UpdateCounter"和"UpdateFlag"检查对工艺 DB 的统一评估。上图显示了"UpdateFlag"和"UpdateCounter"的响应:

检查工艺 DB 的统一评估

根据上述步骤，检查当与一致性相关的操作在 OB 65 上处于活动状态时，新工艺 DB 更新周期是否已开始。工艺 DB 更新在“MCDevice”DB 上开始。但是，为确保一致性，必须检查要评估的每个工艺 DB 的“UpdateCounter”和“UpdateFlag”。步骤：

1. 启动 OB 65 时，保存要评估的工艺 DB 的“UpdateCounter”功能的各个值。
2. 评估一致性数据，或将要评估的工艺 DB 传送到其它工作区域
3. 检查“UpdateFlag”。如果“UpdateFlag”有一个值为 *W#16#0100*，则对该工艺数据块的统一评估将无法得到保证。
4. 将“UpdateCounter”与先前保存的值进行比较。如果“UpdateCounter”增加了计数 1，则对该工艺 DB 的统一评估将无法得到保证
5. 对要评估的所有工艺 DB，请重复步骤 3 和 4

如果工艺 DB 具有相同的“UpdateCounter”值，则工艺 DB 中的各值将来自同一更新周期。

采取以下措施解决任何一致性问题：

- 延长工艺 DB 更新时间。
- 在 OB 65 启动时评估工艺 DB。
- 减少要评估的指令的数目。

说明

异步更新 *ErrorID*、*ErrorStatus* 和 *ErrorBuffer* 条目，即，不考虑“UpdateFlag”的状态。

7.12 ErrorID — 工艺 DB

7.12.1 ErrorID — 轴工艺 DB

警告

ErrorID	警告消息	可能的原因	补救措施	S7T Config 中的中断 *
0010	非位置控制编码器存在故障	非活动数据集的编码器存在故障	可在 S7T 中通过报警编号 20015 识别故障编码器。	20015
0020	重启以激活参数更改	参数已更改，但没有立即应用	通过禁用工艺对象然后再初始化工艺对象来激活参数更改（“MC_Reset”，其中 <i>Restart</i> = <i>TRUE</i> ）	20010
0021	动态响应值受限制	命令的动态响应值（速度、加速度、减速度或冲击）受限制，因为这些值超出了设定限制。 动态值还可能受工艺对象上处于活动状态的任何叠加命令的限制。仅当只有一个运动命令保持活动状态时才能确认警告。	检查超出设定值的原因：组态的最大值；速度设定值界面；编码器连接；机械配置。	40002, 40003, 40004, 50003, 50005
0022	内部警告	临时内部问题	有关详细信息，请参考 S7T Config 的报警视图	20009, 40109, 50010
0023	最大加速度限制了保持时间	无法维持定义的保持时间。由于有效的最大加速度导致无法达到该保持时间，因此保持时间被延长。	在组态中增加保持时间或最大加速度设置。	40013
0026	位置受限于软件限位开关	定位命令的目标位置在软件限位开关的范围之外。	检查计算过的目标位置。	40105
* S7T Config 中，可能与所指示的警告有关的中断。				

错误

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
8001	致命的内部错误	项目/软件故障/不一致；用户无法校正。	检查项目的一致性。重新编译所有数据，然后将它们下载到模块。从 S7T Config 的报警视图中读取错误代码，然后拨打西门子热线。	20001, 50015
8002	内部错误	• 版本冲突 • 固件错误 • 软件不一致	检查项目的一致性。重新编译所有数据，然后将它们下载到模块。评估 S7T Config 的报警视图，然后拨打西门子热线。	30004, 30005, 30010, 30011, 30015
8003	致命的组态错误	轴组态错误。硬件驱动程序或其寻址的设备可能不可用	• 检查已更改参数的似然性。示例：丝杠螺距为 0；如果没有编码器零标记，则绝对编码器无法回原点。） • 在 HW Config 中检查连接的设备以及它们的组态。 • 验证所用的消息帧类型是否与工艺对象兼容。 • 将工艺再次下载到目标系统。	20004, 20006
8004	组态错误	工艺对象的组态不完整或存在错误。	检查已更改参数的似然性。评估 S7T Config 的报警视图中的详细信息。	20003, 20011

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
8005	命令存储器已满	由于命令容量不足，无法执行命令 — 原因： 工艺对象无法保存附加命令	在 S7T Config 中，增加“活动命令 ID 的最大个数”(maximum number of active command IDs)。使用 <i>numberOfMaxBuffered CommandId</i> 组态数据元素设置“专家列表”中的值。 该设置也可以增加工艺对象的命令处理容量。	30003
8006	存储器不足	集成工艺的存储器不足。	减少工艺对象数目，或者凸轮盘中的插补点数目或多项式数目。	20002
8008	工艺 DB 无效	工艺将拒绝该 DB，原因是该 DB 没有有效的内部 ID。该 DB 可能是在复制操作中生成的。	在“工艺对象管理”(Technology Objects Management) 对话框中重新创建 DB，并再次下载组态数据（建立在线/离线一致性）。	-
8009	由于该工艺 DB 已被更新的实例所代替，因此被禁用	由于已下载了具有另一 DB 号的更新的实例，因此不再将该 DB 分配给工艺对象。	删除 DB	-

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
800A	硬件驱动程序不可用	物理设备的驱动程序不可用或者正被其它工艺对象使用。	检查逻辑设备在输出请求时是否可用，或者是否正被其它工艺对象 (TO, technology object) 使用。 连接适当的设备。检查设备的功能、连接和接线情况。 检查 HW Config 中的拓扑结构。 比较 HW Config 中的组态与工艺对象的组态。	20014
8010	驱动器故障	由于组态或参数数据有误，而导致驱动器或外部设备出现故障，或通信被中断。	检查设备功能、连接和接线情况。根据当前的消息帧设置，检查 HW Config 中的组态。	20005, 50001
8012	回原点错误	不支持请求的回原点模式。 示例：使用绝对编码器回原点；回原点速度 = 0。	检查轴回原点的设置是否一致以及是否存在无效值。	40103
8013	硬件限位开关处于活动状态	- 硬件限位开关已响应。 - 开关的极性已被反向。 - 退回方向错误	将驱动器返回至其操作范围。检查限位开关连接。	50007
8014	软件限位开关已启动	运动已在到达软件限位开关后停止	检查运动程序。	40106, 40111
8015	软件限位开关超调	在运动执行期间软件限位开关过调。	检查运动程序。	40107, 40112
8016	跟随误差超出限制	位置设定值和轴的实际位置之间的偏移已超出组态的限制。	检查编码器的传输方向以及位置控制回路和跟随误差监视的设置。	50102
8017	跟随误差已超出了警告限制	位置设定值和轴的实际位置之间的偏移已超出组态的警告限制。	检查编码器的传输方向以及位置控制回路和跟随误差监视的设置。	50103

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
8018	停止监视错误	轴已移出停止窗口，或者无法在指定的时间内到达停止窗口。	检查速度和位置控制回路的功能以及停止监视的设置。	50107, 50008
8019	定位监视错误	轴无法在指定的时间内到达定位窗口。	检查速度和位置控制回路的功能以及定位监视的设置。	50106
801A	同步操作监视错误	同步分组的跟随轴不能在组态的公差限制内进行操作，这表明位置设定值和实际值之间的偏移过高。	检查同步参数和去同步参数、传动比和组态的单元。如果可能，增加设定公差。	40110, 40201
801B	紧固监视错误	紧固时出错。 未收到新的运动命令，轴就已超出“固定挡块检测后的位置公差”。 位置公差在 S7T Config 中的“限制”(Limits) > “固定挡块”(Fixed end stop) 选项卡 > “固定挡块检测后的位置公差”(Position tolerance after fixed end stop detection) 参数中设置。	检查： • 机械挡块 • 位置控制回路的功能 • S7T Config 中的紧固监视设置	50108
801C	超出动态限制	由于用户输入超出了限制值或导致轴发生不允许的反向或超调，因此内部监视系统已介入动态响应过程。	尤其在设置超驰运动的值时，请确保不要选择过低的加速度值、减速度值和冲击值。进行叠加运动时，始终为动态响应参数的叠加值留出余量。	40012, 40202
801D	找不到参考凸轮或编码器零标记	• 参考凸轮或编码器零标记在允许的范围之外。 • 限位监视功能已响应。 • 速度过高。	检查有效范围（例如，参考凸轮和编码器零标记之间的最大距离）以及限位监视系统，并降低关闭速度。	40101, 40102

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
801F	动态控制回路超出范围	- 闭环位置或速度控制系统未优化。 - 编码器的传输方向不正确。 - 跟随误差监视对这些参数设置过于敏感 - 速度设定值和实际速度值之间的偏差过大。	- 检查编码器的传输方向。 - 检查控制回路参数。 - 修改跟随误差监视的参数。	50101
8020	设定值限制处于活动状态	速度设定值受限于受控变量限制。	- 在“轴”(Axis) > “控制”(Control) 对话框的“静态控制器数据”(Static controller data) 选项卡的“受控变量限制”(Manipulated variable limiting) 域中检查受控变量限制的最大值。 - 检查速度设定值接口的组态 - 检查机械配置。 - 检查编码器连接	50016
8030	已超过了测量系统的限制频率	编码器信号的脉冲速率超出了允许的范围。	检查编码器的电气连接。检查组态的最大速度是否过低。	50002
8031	实际值监视错误	- 增量编码器超出范围 - 位置调节器每脉冲的移动超出了轴的模数长度	检查并调整组态数据以使实际值符合机械要求。	50001, 50013
8032	无法切换坐标系	编程的坐标系切换（设置或纠正实际位置值）将导致超出内部行程限制。未切换坐标系。	检查坐标系切换的编程。	30014
8033	已触发编码器零标记监视	编码器零标记之间的增量数目不正确。	检查电气连接和设定的编码器精度。	50006

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
8040	轴/外部编码器被禁用，或者设置了错误的模式	缺少运动命令所需的使能信号。	消除并确认所有排队的错误，然后启用相关的轴模式（例如，位置控制）。	40005
8041	轴未回原点	活动的运动命令需要已回原点的轴。	使轴回原点（“MC_Home”）。如果要在未回原点的轴上执行该命令，则将轴回原点设置为“需要回原点 = 否”	40108
8042	轴状态的更改无效	更改轴状态的命令被拒绝，因为：	根据原因执行下列测量：	40001
		当前操作阶段尚未结束。	重复该命令	
		无法进行模式转换。	首先执行复位。	
8043	命令参数值或默认值无效	命令的工艺功能中的输入参数中存在一个或多个无效的值，或者默认参数无效。	检查命令并调整无效的参数。您还应该注意当前使用的默认值。	30001
8045	在当前状态下不允许使用该命令	- 具有更高优先级的命令处于活动状态（例如“MC_Stop”） - 工艺对象被禁用或未准备就绪。 - 回原点时收到的测量命令	确认工艺对象的所有排队的错误。 等待工艺对象的初始化（在 Restart 之后）。 启用工艺对象。 取消激活“MC_Stop”。 回原点期间请勿启动任何测量命令。	30006, 30009
8046	驱动器不支持该命令	- 连接的驱动器不支持此类命令，例如，特定的回原点方法。 - 轴上的测量输入命令处于活动状态。	检查并调整驱动器设置和组态数据。	50012

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
8047	无法确认该错误	无法使用“MC_Reset”确认该错误。示例： - 轴已停止，无法确认导致轴停止的错误。 - 在确认期间或重启阶段发生了更多错误。	消除导致错误的所有原因，然后重新尝试确认。请注意，您必须通过断开然后再接通电源来确认特定的错误。	30007
8048	无法重启	工艺对象处于活动状态，或排队的错误阻止系统重启。	消除并确认所有排队的错误，然后禁用该工艺对象。	20012
8049	非法的对象组合	工艺对象的组合有误。示例：	根据原因执行下列测量：	20007
		同步操作中将 Axis_2 设置为 Axis_1 的引导轴；同时，将 Axis_1 设置为 Axis_2 的引导轴。	设置一个已定义的引导轴/跟随轴组合。	
		在不同的运行级别处理同步操作中涉及的轴。	将引导轴和跟随轴分配到相同的运行级别（通常为插补器周期时钟）。	
804A	缺少所需的对象连接	- 没有在组态中考虑引导轴与跟随轴和凸轮盘的所需组合。 - 组态的凸轮盘未分配给在 S7T Config 中作为阀控曲线的轴。	从跟随轴的同步对象的“组态”部分中选择所需的引导轴、凸轮盘或阀控曲线。	30008
804B	限位开关处于活动状态；命令无效	硬件限位开关处于活动状态。只允许在操作范围方向上的运动命令。	检查机械配置和电气连接。 纠正所有程序错误或使用软件限位开关。	50009

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
8082	参数中存在无效的零值	所要求的命令包含带有零值的参数。但是，该值必须始终不为零。	检查参数和默认值。尤其需要注意限制和动态值（例如，定位操作的最大速度或减速度）。	40006, 40007, 40008, 40009, 40010, 40011
8085	限位开关参数无效	SW 限位开关中设置的位置值不合理。其中一个值超出允许范围，或者起始值大于结束值。重新分配参数时如果不想按照正确的顺序，则起始值可以大于结束值。	调整软件限位开关的设置。第一次更改位置后，正向开关的位置值必须大于负向开关的位置值。	40104
* S7T Config 中，可能与所指的错误有关的中断				

7.12.2 ErrorID — 同步工艺 DB

警告

ErrorID	警告消息	可能的原因	补救措施	S7T Config 中的中断 * 同步对象	S7T Config 中的中断 * 同步轴
0010	非位置控制编码器存在故障	非活动数据集的编码器存在故障	可在 S7T 中通过报警编号 20015 识别故障编码器。	-	20015
0020	重启以激活参数更改	参数已更改，但没有立即应用	通过禁用工艺对象然后再初始化工艺对象（“MC_Reset”，其中 <i>Restart</i> = <i>TRUE</i> ）来应用参数更改	20010	20010

ErrorID	警告消息	可能的原因	补救措施	S7T Config 中的中断 * 同步对象	S7T Config 中的中断 * 同步轴
0021	动态响应值受限制	- 命令的动态响应值（速度、加速度、减速度或冲击）受限制，因为这些值超出了设定限制。 - 动态值还可能受工艺对象上处于活动状态的任何叠加命令的限制。直到叠加的命令的结果动态响应不再受限，才能确认警告。	- 检查超出设定值的原因：组态的最大值；速度设定值界面；编码器连接；机械配置。 - 在叠加运动结束后确认警告。	40002, 40003, 40004	40002, 40003, 40004, 50003, 50005
0022	内部警告	临时内部问题	有关详细信息，请参考 S7T Config 的报警视图	50103, 50104, 50105, 50106, 50108, 50109	20009, 40109, 50010
0023	最大加速度限制了保持时间	无法维持定义的保持时间。由于有效的最大加速度导致无法达到该保持时间，因此保持时间被延长。	在组态中增加保持时间或最大加速度设置。	40013	0023
0024	齿轮传动不能长时间保持稳定	传动比的分子和分母值过高。在长时间操作中这可能导致数学舍入错误。	通过运行新的同步修改传动比或纠正任何不精确的数据。	50008	-
0026	位置受限于软件限位开关	定位命令的目标位置在软件限位开关的范围之外。	检查计算过的目标位置。	-	40105
* S7T Config 中，可能与所指示的警告有关的中断。					

错误

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 * 同步对象	S7T Config 中的中断 * 同步轴
8001	致命的内部错误	项目/软件故障/不一致；用户无法校正。	检查项目的一致性。重新编译所有数据，然后将它们下载到模块。从 S7T Config 的报警视图中读取错误代码，然后拨打西门子热线。	20001	20001, 50015
8002	内部错误	- 版本冲突 - 固件错误 - 软件不一致	检查项目的一致性。重新编译所有数据，然后将它们下载到模块。评估 S7T Config 的报警视图，然后拨打西门子热线。	30004, 30005, 30010, 30011, 30015	30004, 30005, 30010, 30011, 30015
8003	致命的组态错误	轴组态错误。硬件驱动程序或其寻址的设备可能不可用	- 检查已更改参数的似然性。示例：丝杠螺距为 0；如果没有编码器零标记，则绝对编码器无法回原点。） - 在 HW Config 中检查连接的设备以及它们的组态。 - 验证所用的消息帧类型是否与工艺对象兼容。 - 将工艺再次下载到目标系统。	20004, 20006	20004, 20006
8004	组态错误	工艺对象的组态不完整或存在错误。	检查已更改参数的似然性。评估 S7T Config 的报警视图中的详细信息。	20003, 20011	20003, 20011

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 * 同步对象	S7T Config 中的中断 * 同步轴
8005	命令存储器已满	由于命令容量不足，无法执行命令 — 原因： 工艺对象无法保存附加命令	在 S7T Config 中，增加“活动命令 ID 的最大个数”(maximum number of active command IDs)。使用 <i>numberOfMaxBuffered CommandId</i> 组态数据元素设置“专家列表”中的值。 该设置也可以增加工艺对象的命令处理容量。	30003	30003
8006	存储器不足	集成工艺的存储器不足。	减少工艺对象数目，或者凸轮盘中的插补点数目或多项式数目。	20002	20002
8008	工艺 DB 无效	工艺将拒绝该 DB，原因是该 DB 没有有效的内部 ID。该 DB 可能是在复制操作中生成的。	在“工艺对象管理”(Technology Objects Management)对话框中重新创建 DB，并再次下载组态数据（建立在线/离线一致性）。	-	-
8009	由于该工艺 DB 已被更新的实例所代替，因此被禁用	由于已下载了具有另一 DB 号的更新的实例，因此不再将该 DB 分配给工艺对象。	删除 DB	-	-

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 * 同步对象	S7T Config 中的中断 * 同步轴
800A	硬件驱动程序不可用	物理设备的驱动程序不可用或者正被其它工艺对象使用。	- 检查逻辑设备在输出请求时是否可用，或者是否正被其它工艺对象 (TO, technology object) 使用。 - 连接适当的设备。检查设备的功能、连接和接线情况。 - 检查 HW Config 中的拓扑结构。 - 比较 HW Config 中的组态与工艺对象的组态。	20014	20014
8010	驱动器故障	由于组态或参数数据有误，而导致驱动器或外部设备出现故障，或通信被中断。	检查设备功能、连接和接线情况。根据当前的消息帧设置，检查 HW Config 中的组态。	20005	20005, 50001
8012	回原点错误	不支持请求的回原点模式。 示例：使用绝对编码器回原点；回原点速度 = 0。	检查轴回原点的设置是否一致以及是否存在无效值。	-	40103
8013	硬件限位开关处于活动状态	- 硬件限位开关已响应。 - 开关的极性已被反向。 - 退回运动的方向错误	将驱动器返回至其操作范围。检查限位开关连接。	-	50007
8014	软件限位开关已启动	运动已在到达软件限位开关后停止。	检查运动程序。	-	40106, 40111

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 * 同步对象	S7T Config 中 的中断 * 同步轴
8015	软件限位开关超调	在运动执行期间软件限位开关过调。	检查运动程序。	-	40107, 40112
8016	跟随误差超出限制	位置设定值和轴的实际位置之间的偏移已超出组态的限制。	检查编码器的传输方向以及位置控制回路和跟随误差监视的设置。	-	50102
8017	跟随误差已超出了警告限制	位置设定值和轴的实际位置之间的偏移已超出组态的警告限制。	检查编码器的传输方向以及位置控制回路和跟随误差监视的设置。	-	50103
8018	停止监视错误	轴已移出停止窗口，或者无法在指定的时间内到达停止窗口。	检查速度和位置控制回路的功能以及停止监视的设置。		50107, 50008
8019	定位监视错误	轴无法在指定的时间内到达定位窗口。	检查速度和位置控制回路的功能以及定位监视的设置。	-	50106
801A	同步监视错误	同步分组的跟随轴不能在组态的公差限制内进行操作，这表明位置设定值和实际值之间的偏移过高。	检查同步参数和去同步参数、传动比和组态的单元。如果可能，增加设定公差。	-	40110, 40201

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 * 同步对象	S7T Config 中 的中断 * 同步轴
801B	紧固监视错误	紧固时出错。 未收到新的运动命令，轴就已超出“固定挡块检测后的位置公差” 。 位置公差在 S7T Config 中的“限制”(Limits) > “固定挡块”(Fixed end stop) 选项卡 > “固定挡块检测后的位置公差”(Position tolerance after fixed end stop detection) 参数中设置。	检查： • 机械挡块 • 位置控制回路的功能 • S7T Config 中的紧固监视设置	-	50108

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 * 同步对象	S7T Config 中 的中断 * 同步轴
801C	超出动态限制	由于用户输入超出了限制值或导致轴发生不允许的反向或超调，因此内部监视系统已介入动态响应过程。	尤其在设置释放运动的值时，请确保不要选择过低的加速度值、减速度值和冲击值。进行叠加运动时，始终为动态响应参数的叠加值留出余量。	-	40012
		同步过程中由引导轴返回的动态值高于在跟随轴上定义的值。跟随轴上为同步设置的动态值将被忽略，跟随轴因此便与引导轴同步。	如果要根据跟随轴的设定动态值使跟随轴同步，则请将同步对象的组态参数 <i>SyncingMotion.synchronizing adaptation</i> 的值更改为 <i>NO</i>。您也可以使用“MC_WriteParameter”工艺功能上的参数号 4337 来更改组态数据元素。		
801D	找不到参考凸轮或编码器零标记	- 参考凸轮或编码器零标记在允许的范围之外。 - 限位监视已响应。 - 速度过高。	检查有效范围（例如，参考凸轮和编码器零标记之间的最大距离）以及限位监视系统，并降低关闭速度。	-	40101, 40102

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 * 同步对象	S7T Config 中的中断 * 同步轴
801F	动态控制回路超出范围	- 闭环位置或速度控制系统未优化。 - 编码器的传输方向不正确。 - 跟随误差监视对这些参数设置过于敏感 - 速度设定值和实际速度值之间的偏差过大。	- 检查编码器的传输方向。 - 检查控制回路参数。 - 编辑跟随误差监视的参数。	-	50101
8030	已超过了测量系统的限制频率	编码器信号的脉冲速率超出了允许的范围。	检查编码器的电气连接。检查组态的最大速度是否过低。	-	50002
8031	实际值监视错误	- 超出了增量编码器的范围。 - 位置调节器每脉冲的移动超出了轴的模数长度	检查并调整组态数据以使实际值符合机械要求。	-	50001, 50013
8032	无法切换坐标系	编程的坐标系切换（设置或纠正实际位置值）将导致超出内部行程限制。未切换坐标系。	检查编程的坐标系切换。	-	30014
8033	已触发编码器零标记监视	编码器零标记之间的增量数目不正确。	检查电气连接和设定的编码器精度。	-	50006
8040	轴/外部编码器被禁用，或者设置了错误的模式	缺少运动命令所需的使能信号。	消除并确认所有排队的错误，然后启用相关的轴模式（例如，位置控制）。	-	40005

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 * 同步对象	S7T Config 中 的中断 * 同步轴
8041	轴未回原点	活动的运动命令需要已回原点的轴。	使轴回原点 (“MC_Home”)。如果要在未回原点的轴上执行该命令，则将轴回原点设置为“需要回原点 = 否”	-	40108
8042	轴状态的更改无效	更改轴状态的命令被拒绝，因为：	根据原因执行下列测量：	-	40001
		• 运行状态的这一阶段尚未结束。	• 重复该命令		
		• 无法进行模式转换。	• 首先执行复位。		
8043	命令参数值或默认值无效	请求的工艺功能中的输入参数中存在一个或多个无效的值，或者默认参数无效。	检查命令并调整无效的参数。您还应该注意当前使用的默认值。	30001, 50107	30001
8045	在当前状态下不允许使用该命令	- 具有更高优先级的命令处于活动状态（例如 “MC_Stop”）。 - 工艺对象被禁用或未准备就绪。 - 回原点时收到的测量命令	确认工艺对象的所有排队错误。 等待工艺对象的初始化（在 Restart 之后）。 启用工艺对象。 取消激活“MC_Stop”。 回原点期间请勿启动任何测量命令。	30006	30006, 30009
8046	驱动器不支持该命令	- 连接的驱动器不支持此类命令，例如，特定的回原点方法。 - 轴上的测量输入命令处于活动状态。	检查并调整驱动器设置和组态数据。	-	50012

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 * 同步对象	S7T Config 中的中断 * 同步轴
8047	无法确认该错误	不能通过调用工艺功能“MC_Reset”确认错误。示例： - 轴已停止，无法确认导致轴停止的错误。 - 在确认期间或重启阶段发生了更多错误。	消除导致错误的所有原因，然后重新尝试确认。请注意，您必须通过断开然后再接通电源来确认特定的错误。	30007	30007
8048	无法重启	工艺对象处于活动状态，或排队的错误阻止系统重启。	消除并确认所有排队的错误，然后禁用该工艺对象。	20012	20012
8049	非法的对象组合	工艺对象的组合有误。示例：	根据原因执行下列测量：	20007	20007
		同步操作中将 Axis_2 设置为 Axis_1 的引导轴；同时，将 Axis_1 设置为 Axis_2 的引导轴。	设置一个已定义的引导轴/跟随轴组合。		
		在不同的运行级别处理同步操作中涉及的轴。	将引导轴和跟随轴分配到相同的运行级别（通常为插补器周期时钟）。		
804A	缺少所需的对象连接	- 没有在组态中考虑引导轴与跟随轴和凸轮盘的所需组合。 - 组态的凸轮盘未分配给在 S7T Config 中作为阀控曲线的轴。	从跟随轴的同步对象的“组态”部分中选择所需的引导轴、凸轮盘或阀控曲线。	30008	30008

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 * 同步对象	S7T Config 中的中断 * 同步轴
804B	限位开关处于活动状态；命令无效	硬件限位开关处于活动状态。只允许在操作范围方向上的运动命令。	检查机械配置和电气连接。 纠正所有程序错误或使用软件限位开关。	-	50009
8063	未插补凸轮	- 凸轮传动开始时，检测到尚未对该凸轮进行插补。 - 正在使用非插补阀控曲线。	检查工艺 DB 上的当前凸轮状态。添加缺少的点或区段，然后对凸轮进行插补。	50002	-
8064	无法访问凸轮	指定的凸轮盘不存在。	检查凸轮盘以及轴所有可能的互连（同步的组态）。	50001, 50004	-
		凸轮盘未与同步操作互连。	为了能够编辑凸轮盘，凸轮盘在凸轮同步中不能处于活动状态，也不能用作液压轴的阀控曲线。		
		凸轮盘已处于活动状态。	如果相对于引导轴启动凸轮传动，则同步对象中的系统变量 <i>userdefault.cammingsettings.camstartposition master</i> 的值必须位于凸轮盘的定义范围内。		

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 * 同步对象	S7T Config 中 的中断 * 同步轴
		引导轴的位置在凸轮定义范围之外	为了执行凸轮与引导轴绝对参考的非循环同步，引导轴的当前位置必须在凸轮的定义范围内。为由输入参数 <i>MasterOffset</i> 引起的凸轮切换留出余量。		
8065	凸轮已超出范围	指定的凸轮插补点超出了定义的凸轮范围。	检查凸轮同步和去同步的起始和结束位置。	50003	-
8072	激活/取消激活同步操作时出错	该错误是由于同步和去同步期间引导轴的无效状态引起的。示例：引导轴在基于位置的同步期间反向。	检查引导轴的运动，或选择其它同步策略。	50007	-
8073	同步操作期间引导轴动态值超出范围	同步和去同步期间，引导轴上的动态值过高（例如，加速度值过高）	检查引导轴的运动，或选择其它同步策略。	50009	-
8074	未为同步操作组态引导轴	- 没有在跟随轴组态中将连接到“引导”终端的轴定义为引导轴。 - 未启用引导轴（仅适用于外部编码器）。 - 在速度控制模式下对引导轴进行操作。	调整跟随轴的组态或使用已组态的引导轴。在位置控制模式下对引导轴进行操作或启用它。	50101, 50102	-

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 * 同步对象	S7T Config 中的中断 * 同步轴
8077	取消激活已中止的同步操作	由于当前的同步模式（齿轮/凸轮传动）与取消激活类型不一致，因此取消激活同步已被拒绝。	使用 MC_CamOut 取消激活凸轮传动，使用 MC_GearOut 取消激活齿轮传动，或使用全局指令（“MC_Halt”、“MC_Stop”等）。	50005	-
8078	无法进行叠加同步操作	在专家列表中为轴组态叠加同步操作。	纠正专家列表中的设置。	-	-
8082	参数中存在无效的零值	所要求的命令包含带有零值的参数。但是，该值必须始终不为零。	检查参数和默认值。尤其需要注意限制和动态值（例如，定位操作的最大速度或减速度）。	40006, 40007, 40008, 40009, 40010, 40011	40006, 40007, 40008, 40009, 40010, 40011
8085	限位开关参数无效	SW 限位开关中设置的位置值不合理。其中一个值超出允许范围，或者起始值大于结束值。如果忽略更改参数的顺序，则会出现后一种情况。	调整软件限位开关的设置。第一次更改位置后，正向开关的位置值必须大于负向开关的位置值。	-	40104
* S7T Config 中，可能与所指的错误有关的中断					

7.12.3 ErrorID — 外部编码器工艺 DB

适用于固件版本为 V4.1.x 的集成工艺

本部分介绍了固件版本为 V3.0.x、V3.1.x 和 V3.2.x 的应用。

警告

ErrorID	警告消息	可能的原因	补救措施	S7T Config 中的中断 *
0020	重启以激活参数更改	参数已更改，但没有立即应用	通过禁用工艺对象然后再初始化工艺对象 （“MC_Reset”，其中 <i>Restart = TRUE</i> ）来应用参数更改	20010
0021	动态响应值受限制	命令的动态响应值（速度、加速度、减速度或冲击）受限制，因为这些值超出了设定限制。	检查超出设定值的原因： 组态的最大值；速度设定值界面；编码器连接；机械配置。	50003, 50005
* S7T Config 中，可能与所指示的警告有关的中断。				

错误

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
8001	致命的内部错误	项目/软件故障/不一致；用户无法校正。	检查项目的一致性。重新编译所有数据，然后将它们下载到模块。从 S7T Config 的报警视图中读取错误代码，然后拨打西门子热线。	20001, 50015
8002	内部错误	• 版本冲突 • 固件错误 • 软件不一致	检查项目的一致性。重新编译所有数据，然后将它们下载到模块。评估 S7T Config 的报警视图，然后拨打西门子热线。	30004, 30005, 30010, 30011, 30015
8003	致命的组态错误	轴组态错误。硬件驱动程序或其寻址的设备可能不可用	• 检查已更改参数的似然性。示例：丝杠螺距为 0；如果没有编码器零标记，则绝对编码器无法回原点。） • 在 HW Config 中检查连接的设备以及它们的组态。 • 验证所用的消息帧类型是否与工艺对象兼容。 • 将工艺再次下载到目标系统。	20004, 20006
8004	组态错误	工艺对象的组态不完整或存在错误。	检查已更改参数的似然性。评估 S7T Config 的报警视图中的详细信息。	20003, 20011

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
8005	命令存储器已满	由于命令容量不足，无法执行命令 — 原因： 工艺对象无法保存附加命令	在 S7T Config 中，增加“活动命令 ID 的最大个数”(maximum number of active command IDs)。使用 <i>numberOfMaxBuffered CommandId</i> 组态数据元素设置“专家列表”中的值。 该设置也可以增加工艺对象的命令处理容量。	30003
8006	存储器不足	集成工艺的存储器不足。	减少工艺对象数目，或者凸轮盘中的插补点数目或多项式数目。	20002
8008	工艺 DB 无效	工艺将拒绝该 DB，原因是该 DB 没有有效的内部 ID。该 DB 可能是在复制操作中生成的。	在“工艺对象管理”(Technology Objects Management) 对话框中重新创建 DB，并再次下载组态数据（建立在线/离线一致性）。	-
8009	由于该工艺 DB 已被更新的实例所代替，因此被禁用	由于已下载了具有另一 DB 号的更新的实例，因此不再将该 DB 分配给工艺对象。	删除 DB	-

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
800A	硬件驱动程序不可用	物理设备的驱动程序不可用或者正被其它工艺对象使用。	- 检查逻辑设备在输出请求时是否可用，或者是否正被其它工艺对象 (TO, technology object) 使用。 - 连接适当的设备。检查设备的功能、连接和接线情况。 - 检查 HW Config 中的拓扑结构。 - 比较 HW Config 中的组态与工艺对象的组态。	20014
8010	驱动器故障	由于组态或参数数据有误，而导致驱动器或外部设备出现故障，或通信被中断。	检查设备功能、连接和接线情况。根据当前的消息帧设置，检查 HW Config 中的组态。	20005, 50001
8012	回原点错误	不支持请求的回原点模式。 示例：使用绝对编码器回原点；回原点速度 = 0。	检查轴回原点的设置是否一致以及是否存在无效值。	40103
8013	硬件限位开关处于活动状态	- 硬件限位开关已响应。 - 开关的极性已被反向。 - 退回方向错误	将驱动器返回至其操作范围。检查限位开关连接。	50007
8016	跟随误差超出限制	位置设定值和轴的实际位置之间的偏移已超出组态的限制。	检查编码器的传输方向以及位置控制回路和跟随误差监视的设置。	50102
8017	跟随误差已超出了警告限制	位置设定值和轴的实际位置之间的偏移已超出组态的警告限制。	检查编码器的传输方向以及位置控制回路和跟随误差监视的设置。	50103
8018	停止监视错误	轴已移出停止窗口，或者无法在指定的时间内到达停止窗口。	检查速度和位置控制回路的功能以及停止监视的设置。	50107, 50008

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
8019	定位监视错误	轴无法在指定的时间内到达定位窗口。	检查速度和位置控制回路的功能以及定位监视的设置。	50106
801B	紧固监视错误	紧固时出错。 未收到新的运动命令，轴就已超出“固定挡块检测后的位置公差”。 位置公差在 S7T Config 中的“限制”(Limits) >“固定挡块”(Fixed end stop) 选项卡 >“固定挡块检测后的位置公差”(Position tolerance after fixed end stop detection) 参数中设置。	检查： • 机械挡块 • 位置控制回路的功能 • S7T Config 中的紧固监视设置	50108
801D	找不到参考凸轮或编码器零标记	• 参考凸轮或编码器零标记在允许的范围之外。 • 限位监视功能已响应。 • 速度过高。	检查有效范围（例如，参考凸轮和编码器零标记之间的最大距离）以及限位监视系统，并降低关闭速度。	40102
801F	动态控制回路超出范围	• 闭环位置或速度控制系统未优化。 • 编码器的传输方向不正确。 • 为跟随误差监视分配的参数过于敏感 • 速度设定值和实际速度值之间的偏差过大。	• 检查编码器的传输方向。 • 检查控制回路参数。 • 修改跟随误差监视的参数。	50101
8030	已超过了测量系统的限制频率	编码器信号的脉冲速率超出了允许的范围。	检查编码器的电气连接。检查组态的最大速度是否过低。	50002
8031	实际值监视错误	• 增量编码器超出范围 • 位置调节器每脉冲的移动超出了轴的模数长度。	检查并调整组态数据以使实际值符合机械要求	50001, 50013

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
8032	无法切换坐标系	编程的坐标系切换（设置或纠正实际位置值）将导致超出内部行程限制。未切换坐标系。	检查编程的坐标系切换。	30014
8033	已触发编码器零标记监视	编码器零标记之间的增量数目不正确。	检查电气连接和设定的编码器精度。	50006
8040	轴/外部编码器被禁用，或者设置了错误的模式	缺少所需的命令使能信号。	消除并确认所有排队的错误，然后为相关模式启用轴。	40005
8043	命令参数值或默认值无效	命令的工艺功能中的输入参数中存在一个或多个无效的值，或者默认参数无效。	检查命令并调整无效的参数。您还应该注意当前使用的默认值。	30001
8045	在当前状态下不允许使用该命令	- 具有更高优先级的命令处于活动状态（例如“MC_Stop”）。 - 工艺对象被禁用或未准备就绪。 - 回原点时收到的测量命令	确认工艺对象的所有排队的错误。 等待工艺对象的初始化（在 Restart 之后）。 启用工艺对象。 取消激活“MC_Stop”。 请勿在回原点阶段启用任何测量命令。	30006, 30009
8046	驱动器不支持该命令	- 连接的驱动器不支持此类命令，例如，特定的回原点方法。 - 外部编码器上的测量输入命令处于活动状态。	检查并调整驱动器设置和组态数据。	50012
8047	无法确认该错误	无法使用“MC_Reset”确认该错误。示例： - 轴已停止，无法确认导致轴停止的错误。 - 在确认期间或重启阶段发生了更多错误。	消除导致错误的所有原因，然后重新尝试确认。 请注意，您必须通过断开然后再接通电源来确认特定的错误。	30007

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
8048	无法重启	工艺对象处于活动状态，或排队的错误阻止系统重启。	消除并确认所有排队的错误，然后禁用该工艺对象。	20012
8049	非法的对象组合	工艺对象的组合有误。示例：	根据原因执行下列测量：	20007
		同步操作中将 Axis_2 设置为 Axis_1 的引导轴；同时，将 Axis_1 设置为 Axis_2 的引导轴。	设置一个已定义的引导轴/跟随轴组合。	
		在不同的运行级别处理同步操作中涉及的轴。	将引导轴和跟随轴分配到相同的运行级别（通常为插补器周期时钟）。	
804A	缺少所需的对象连接	没有在组态中考虑引导轴与跟随轴和凸轮盘的所需组合。	在跟随轴的同步对象的“组态”(Configuration) 部分中，设置所需的引导轴和凸轮。	30008
804B	限位开关处于活动状态；命令无效	硬件限位开关处于活动状态。只允许在操作范围方向上的运动命令。	检查机械配置和电气连接。 纠正所有程序错误或使用软件限位开关。	50009
* S7T Config 中，可能与所指的错误有关的中断				

7.12.4 ErrorID — 凸轮工艺 DB

适用于固件版本为 V4.1.x 的集成工艺

警告

ErrorID	警告消息	可能的原因	补救措施	S7T Config 中的中断 *
0020	重启以激活参数更改	参数已更改，但没有立即应用	通过禁用工艺对象然后再初始化工艺对象（“MC_Reset”，其中 <i>Restart = TRUE</i> ）来应用参数更改	20010
0022	内部警告	临时内部问题	有关详细信息，请参考 S7T Config 的报警视图	40003
0027	无法维护插补属性。	在凸轮区段之间的转换处检测到位置或梯度中断	移动插补限值。 检查凸轮盘配置文件。 更改区段插补的标准。	40008
* S7T Config 中，可能与所指示的警告有关的中断。				

错误

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
8001	致命的内部错误	项目/软件故障/不一致； 用户无法校正。	检查项目的一致性。重新编译所有数据，然后将它们下载到模块。从 S7T Config 的报警视图中读取错误代码，然后拨打西门子热线。	20001
8002	内部错误	- 版本冲突 - 固件错误 - 软件不一致	检查项目的一致性。重新编译所有数据，然后将它们下载到模块。评估 S7T Config 的报警视图，然后拨打西门子热线。	30004, 30005, 30010, 30011, 30015

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
8003	致命的组态错误	轴组态错误。硬件驱动程序或其寻址的设备可能不可用	- 检查已更改参数的似然性。示例：丝杠螺距为 0；如果没有编码器零标记，则绝对编码器无法回原点。） - 在 HW Config 中检查连接的设备以及它们的组态。 - 验证所用的消息帧类型是否与工艺对象兼容。 - 将工艺再次下载到目标系统。	20004, 20006
8004	参数分配错误	工艺对象的组态不完整或存在错误。	检查已更改参数的似然性。评估 S7T Config 的报警视图中的详细信息。	20003, 20011
8005	命令存储器已满	由于命令容量不足，无法执行命令 — 原因： 工艺对象无法保存附加命令	在 S7T Config 中，增加“活动命令 ID 的最大个数”(maximum number of active command IDs)。在组态参数 <i>numberOfMaxBuffered CommandId</i> 中设置“专家列表”中的值。 该设置也可以增加工艺对象的命令处理容量。	30003
8006	存储器不足	集成工艺的存储器不足。	减少工艺对象数目，或者凸轮盘中的插补点数目或多项式数目。	20002
8008	工艺 DB 无效	工艺将拒绝该 DB，原因是该 DB 没有有效的内部 ID。该 DB 可能是在复制操作中生成的。	在“工艺对象管理”(Technology Objects Management) 对话框中重新创建 DB，并再次下载组态数据（建立在线/离线一致性）。	-
8009	由于该工艺 DB 已被更新的实例所代替，因此被禁用	由于已下载了具有另一 DB 号的更新的实例，因此不再将该 DB 分配给工艺对象。	删除 DB	-

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
800A	硬件驱动程序不可用	物理设备的驱动程序不可用或者正被其它工艺对象使用。	- 检查逻辑设备在输出请求时是否可用，或者是否正被其它工艺对象 (TO, technology object) 使用。 - 连接适当的设备。检查设备的功能、连接和接线情况。 - 检查 HW Config 中的拓扑结构。 - 比较 HW Config 中的组态与工艺对象的组态。	20014
8010	驱动器故障	由于组态或参数数据有误，而导致驱动器或外部设备出现故障，或通信被中断。	检查设备功能、连接和接线情况。根据当前的消息帧设置，检查 HW Config 中的组态。	20005
8043	命令参数值或默认值无效	命令的工艺功能中的输入参数中存在一个或多个无效的值，或者默认参数无效。	检查命令并调整无效的参数。您还应该注意当前使用的默认值。	30001
8045	在当前状态下不允许使用该命令	- 具有更高优先级的命令处于活动状态（例如“MC_Stop”）。 - 工艺对象被禁用或未准备就绪。 - 回原点时收到了测量命令。	- 确认工艺对象的所有排队的错误。 - 等待工艺对象的初始化（在 Restart 之后）。 - 启用工艺对象。 - 取消激活“MC_Stop”。 - 回原点期间请勿启动任何测量命令。	30006, 30009
8047	无法确认该错误	不能通过调用工艺功能“MC_Reset”确认错误。 示例： - 轴已停止，无法确认导致轴停止的错误。 - 在确认期间或重启阶段发生了更多错误。	消除导致错误的所有原因，然后重新尝试确认。请注意，您必须通过断开然后再接通电源来确认特定的错误。	30007

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
8048	无法重启	工艺对象处于活动状态，或排队的错误阻止系统重启。	消除并确认所有排队的错误，然后禁用该工艺对象。	20012
8049	非法的对象组合	工艺对象的组合有误。 示例：	根据原因执行下列测量：	20007
		同步操作中将 Axis_2 设置为 Axis_1 的引导轴；同时，将 Axis_1 设置为 Axis_2 的引导轴。	设置一个已定义的引导轴/跟随轴组合。	
		以不同的优先级处理同步操作中涉及的轴。	将引导轴和跟随轴分配到相同的优先级（通常为插补器周期时钟）。	
804A	缺少所需的对象连接	没有在组态中考虑引导轴与跟随轴和凸轮盘的所需组合。	在跟随轴的同步对象的“组态”(Configuration) 部分中，设置所需的引导轴和凸轮。	30008
8051	读/写访问被拒绝。	试图在凸轮正被使用期间将其删除或对其进行调整。	必需的步骤： - 同步运行中的凸轮终止凸轮的同步运行，然后对其进行编辑。 - 阀配置文件使用“MC_SetCharacteristic”为液压轴分配不同的凸轮。	40001, 40002
8060	添加凸轮区段时出错	新的凸轮区段有故障，例如，它的起点和终点相同。	检查定义新凸轮区段使用的值。	40004
8061	凸轮点超出域/范围	“MC_GetCamPoint”中定义的引导/跟随轴的位置不存在于凸轮的域/范围中。	在“MC_GetCamPoint”中设置有效位置。	40005, 40006

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
8062	不允许访问插补的凸轮	向已插补的凸轮输出了无效的命令。 示例： "MC_CamSectorAdd"	当凸轮正被使用时，无法对其进行调整。您在调整凸轮前，请使用“MC_CamClear”将其删除。	40007
* S7T Config 中，可能与所指的错误有关的中断				

7.12.5 ErrorID — 测量输入工艺 DB

警告

ErrorID	警告消息	可能的原因	补救措施	S7T Config 中的中断 *
0020	重启以激活参数更改	对参数作出的更改不能直接生效	通过禁用工艺对象然后再初始化工艺对象 (“MC_Reset”, 其中 <i>Restart = TRUE</i>) 来应用参数更改	20010
* S7T Config 中, 可能与所指示的警告有关的中断。				

错误

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
8001	致命的内部错误	项目/软件故障/不一致; 用户无法校正。	检查项目的一致性。重新编译所有数据, 然后将它们下载到模块。从 S7T Config 的报警视图中读取错误代码, 然后拨打西门子热线。	20001
8002	内部错误	• 版本冲突 • 固件错误 • 软件不一致	检查项目的一致性。重新编译所有数据, 然后将它们下载到模块。评估 S7T Config 的报警视图, 然后拨打西门子热线。	30004, 30005, 30010, 30011, 30015

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
8003	致命的组态错误	轴组态错误。硬件驱动程序或其寻址的设备可能不可用	- 检查已更改参数的似然性。示例：丝杠螺距为 0；如果没有编码器零标记，则绝对编码器无法回原点。） - 在 HW Config 中检查连接的设备以及它们的组态。 - 验证所用的消息帧类型是否与工艺对象兼容。 - 将工艺再次下载到目标系统。	20004, 20006, 40006
8004	参数分配错误	工艺对象的组态不完整或存在错误。	检查已更改参数的似然性。评估 S7T Config 的报警视图中的详细信息。	20003, 20011
8005	命令存储器已满	由于命令容量不足，无法执行命令 — 原因： 工艺对象无法保存附加命令	在 S7T Config 中，增加“活动命令 ID 的最大个数”(maximum number of active command IDs)。在组态参数 <i>numberOfMaxBuffered CommandId</i> 中设置“专家列表”中的值。 该设置也可以增加工艺对象的命令处理容量。	30003
8006	存储器不足	集成工艺的存储器不足。	减少工艺对象数目，或者凸轮盘中的插补点数目或多项式数目。	20002
8008	工艺 DB 无效	工艺将拒绝该 DB，原因是该 DB 没有有效的内部 ID。该 DB 可能是在复制操作中生成的。	在“工艺对象管理”(Technology Objects Management) 对话框中重新创建 DB，并再次下载组态数据（建立在线/离线一致性）。	-
8009	由于该工艺 DB 已被更新的实例所代替，因此被禁用	由于已下载了具有另一 DB 号的更新的实例，因此不再将该 DB 分配给工艺对象。	删除 DB	-

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
800A	硬件驱动程序不可用	物理设备的驱动程序不可用或者正被其它工艺对象使用。	- 检查逻辑设备在输出请求时是否可用，或者是否正被其它工艺对象 (TO, technology object) 使用。 - 连接适当的设备。检查设备的功能、连接和接线情况。 - 检查 HW Config 中的拓扑结构。 - 比较 HW Config 中的组态与工艺对象的组态。	20014
8010	驱动器故障	由于组态或参数数据有误，而导致驱动器或外部设备出现故障，或通信被中断。	检查设备功能、连接和接线情况。根据当前的消息帧设置，检查 HW Config 中的组态。	20005
8034	无法在该范围内记录测量值	未在指定的测量范围内检测到期望的传感器信号的跳变沿。	检查测量输入的测量范围和电气功能。	40003, 40004
8035	测量命令已被取消	原因： - 驱动器或编码器错误 - 操作被记录设备所中止 - 多个测量输入访问同一个编码器	清除编码器/驱动器错误。请勿与多个测量输入同时访问同一个编码器。	40005, 40007, 40008
8043	命令参数值或默认值无效	命令的工艺功能中的输入参数中存在一个或多个无效的值，或者默认参数无效。	检查命令并调整无效的参数。您还应该注意当前使用的默认值。	30001
8044	命令不受工艺对象支持	在工艺功能上参数化的工艺 DB 属于不支持该类型命令的工艺对象。 示例： 无法在虚拟轴上进行测量。	使用正确的对象类型。调整轴组态。	40001

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
8045	在当前状态下不允许使用该命令	- 具有更高优先级的命令处于活动状态（例如“MC_Stop”）。 - 工艺对象被禁用或未准备就绪。 - 回原点时收到了测量命令。	- 确认工艺对象的所有排队的错误。 - 等待工艺对象的初始化（在 Restart 之后）。 - 启用工艺对象。 - 取消激活“MC_Stop”。 - 请勿在回原点阶段启用任何测量命令。	30006, 30009, 40002
8047	无法确认该错误	不能通过调用工艺功能“MC_Reset”确认错误。 示例： - 轴已停止，无法确认导致轴停止的错误。 - 在确认期间或重启阶段发生了更多错误。	消除导致错误的所有原因，然后重新尝试确认。请注意，您必须通过断开然后再接通电源来确认特定的错误。	30007
8048	无法重启	工艺对象处于活动状态，或排队的错误阻止系统重启。	消除并确认所有排队的错误，然后禁用该工艺对象。	20012
8049	非法的对象组合	工艺对象的组合有误。 示例：	根据原因执行下列测量：	20007
		同步操作中将 Axis_2 设置为 Axis_1 的引导轴；同时，将 Axis_1 设置为 Axis_2 的引导轴。	设置一个已定义的引导轴/跟随轴组合。	
		以不同的优先级处理同步操作中涉及的轴。	将引导轴和跟随轴分配到相同的优先级（通常为插补器周期时钟）。	
804A	缺少所需的对象连接	没有在组态中考虑引导轴与跟随轴和凸轮盘的所需组合。	在跟随轴的同步对象的“组态”(Configuration) 部分中，设置所需的引导轴和凸轮。	30008
* S7T Config 中，可能与所指的错误有关的中断				

7.12.6 ErrorID — 输出凸轮工艺 DB

警告

ErrorID	警告消息	可能的原因	补救措施	S7T Config 中的中断 *
0020	重启以激活参数更改	参数已更改，但没有立即应用	通过禁用工艺对象然后再初始化工艺对象 (“MC_Reset”，其中 <i>Restart = TRUE</i>) 来应用参数更改	20010
* S7T Config 中，可能与所指示的警告有关的中断。				

错误

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
8001	致命的内部错误	项目/软件故障/不一致；用户无法校正。	检查项目的一致性。重新编译所有数据，然后将它们下载到模块。从 S7T Config 的报警视图中读取错误代码，然后拨打西门子热线。	20001
8002	内部错误	- 版本冲突 - 固件错误 - 软件不一致	检查项目的一致性。重新编译所有数据，然后将它们下载到模块。评估 S7T Config 的报警视图，然后拨打西门子热线。	30004, 30005, 30010, 30011, 30015

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
8003	致命的组态错误	轴组态错误。硬件驱动程序或其寻址的设备可能不可用	- 检查已更改参数的似然性。示例：丝杠螺距为 0；如果没有编码器零标记，则绝对编码器无法回原点。） - 在 HW Config 中检查连接的设备以及它们的组态。 - 验证所用的消息帧类型是否与工艺对象兼容。 - 将工艺再次下载到目标系统。	20004, 20006
8004	参数分配错误	工艺对象的组态不完整或存在错误。	检查已更改参数的似然性。评估 S7T Config 的报警视图中的详细信息。	20003, 20011
8005	命令存储器已满	由于命令容量不足，无法执行命令 — 原因： 工艺对象无法保存附加命令	在 S7T Config 中，增加“活动命令 ID 的最大个数”(maximum number of active command IDs)。在组态参数 <i>numberOfMaxBuffered CommandId</i> 中设置“专家列表”中的值。 该设置也可以增加工艺对象的命令处理容量。	30003
8006	存储器不足	集成工艺的存储器不足。	减少工艺对象数目，或者凸轮盘中的插补点数目或多项式数目。	20002
8008	工艺 DB 无效	工艺将拒绝该 DB，原因是该 DB 没有有效的内部 ID。该 DB 可能是在复制操作中生成的。	在“工艺对象管理”(Technology Objects Management) 对话框中重新创建 DB，并再次下载组态数据（建立在线/离线一致性）。	-

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
8009	由于该工艺 DB 已被更新的实例所代替，因此被禁用	由于已下载了具有另一 DB 号的更新的实例，因此不再将该 DB 分配给工艺对象。	删除 DB	-
800A	硬件驱动程序不可用	物理设备的驱动程序不可用或者正被其它工艺对象使用。	- 检查逻辑设备在输出请求时是否可用，或者是否正被其它工艺对象 (TO, technology object) 使用。 - 连接适当的设备。检查设备的功能、连接和接线情况。 - 检查 HW Config 中的拓扑结构。 - 比较 HW Config 中的组态与工艺对象的组态。	20014
8010	驱动器故障	由于组态或参数数据有误，而导致驱动器或外部设备出现故障，或通信被中断。	检查设备功能、连接和接线情况。根据当前的消息帧设置，检查 HW Config 中的组态。	20005
8021	凸轮位置无效	凸轮上的实际值无效。	- 重启相应的轴或外部编码器。 - 编码器系统上的实际值输出存在故障（例如，由于驱动器故障）。	40005
8043	命令参数值或默认值无效	命令的工艺功能中的输入参数中存在一个或多个无效的值，或者默认参数无效。示例： 输出凸轮的切换位置在轴的操作范围之外。	检查命令并调整无效的参数。您还应该注意当前使用的默认值。	30001, 40003

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
8045	在当前状态下不允许使用该命令	- 具有更高优先级的命令处于活动状态（例如“MC_Stop”）。 - 工艺对象被禁用或未准备就绪。 - 回原点时收到了测量命令。	- 确认工艺对象的所有排队错误。 - 等待工艺对象的初始化（在 Restart 之后）。 - 启用工艺对象。 - 取消激活“MC_Stop”。 - 请勿在回原点阶段启用任何测量命令。	30006, 30009
8047	无法确认该错误	不能通过调用工艺功能“MC_Reset”确认错误。示例： - 轴已停止，无法确认导致轴停止的错误。 - 在确认期间或重启阶段发生了更多错误。	消除导致错误的所有原因，然后重新尝试确认。请注意，您必须通过断开然后再接通电源来确认特定的错误。	30007
8048	无法重启	工艺对象处于活动状态，或排队的错误阻止系统重启。	消除并确认所有排队的错误，然后禁用该工艺对象。	20012
8049	非法的对象组合	工艺对象的组合有误。示例：	根据原因执行下列测量：	20007
		同步操作中将 Axis_2 设置为 Axis_1 的引导轴；同时，将 Axis_1 设置为 Axis_2 的引导轴。	设置一个已定义的引导轴/跟随轴组合。	
		在不同的运行级别处理同步操作中涉及的轴。	将引导轴和跟随轴分配到相同的运行级别（通常为插补器周期时钟）。	
804A	缺少所需的对象连接	没有在组态中考虑引导轴与跟随轴和凸轮盘的所需组合。	在跟随轴的同步对象的“组态”(Configuration) 部分中，设置所需的引导轴和凸轮。	30008

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
8087	输出凸轮位置非法	至少一个输出凸轮位置非法。输出凸轮已移到操作范围限制之外。	检查输出凸轮位置。缩短当前延迟/微分时间。	40001
* S7T Config 中，可能与所指的错误有关的中断				

7.12.7 ErrorID — 凸轮轨迹工艺 DB

适用于固件版本为 V4.1.x 的集成工艺

警告

ErrorID	警告消息	可能的原因	补救措施	S7T Config 中的中断 *
0020	重启以激活参数更改	参数已更改，但没有立即应用	通过禁用工艺对象然后再初始化工艺对象 （“MC_Reset”，其中 <i>Restart = TRUE</i> ）来应用参数更改	20010
* S7T Config 中，可能与所指示的警告有关的中断。				

错误

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
8001	致命的内部错误	项目/软件故障/不一致；用户无法校正。	检查项目的一致性。重新编译所有数据，然后将它们下载到模块。从 S7T Config 的报警视图中读取错误代码，然后拨打西门子热线。	20001
8002	内部错误	- 版本冲突 - 固件错误 - 软件不一致	检查项目的一致性。重新编译所有数据，然后将它们下载到模块。评估 S7T Config 的报警视图，然后拨打西门子热线。	30004, 30005, 30010, 30011, 30015

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
8003	致命的组态错误	轴组态错误。硬件驱动程序或其寻址的设备可能不可用	- 检查已更改参数的似然性。示例：丝杠螺距为 0；如果没有编码器零标记，则绝对编码器无法回原点。） - 在 HW Config 中检查连接的设备以及它们的组态。 - 验证所用的消息帧类型是否与工艺对象兼容。 - 将工艺再次下载到目标系统。	20004, 20006
8004	参数分配错误	工艺对象的组态不完整或存在错误。	检查已更改参数的似然性。评估 S7T Config 的报警视图中的详细信息。	20003, 20011
8005	命令存储器已满	由于命令容量不足，无法执行命令 — 原因： 工艺对象无法保存附加命令	在 S7T Config 中，增加“活动命令 ID 的最大个数”(maximum number of active command IDs)。在组态参数 <i>numberOfMaxBuffered CommandId</i> 中设置“专家列表”中的值。 该设置也可以增加工艺对象的命令处理容量。	30003
8006	存储器不足	集成工艺的存储器不足。	减少工艺对象数目，或者凸轮盘中的插补点数目或多项式数目。	20002

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
800A	硬件驱动程序不可用	物理设备的驱动程序不可用或者正被其它工艺对象使用。	- 检查逻辑设备在输出请求时是否可用，或者是否正被其它工艺对象 (TO, technology object) 使用。 - 连接适当的设备。检查设备的功能、连接和接线情况。 - 检查 HW Config 中的拓扑结构。 - 比较 HW Config 中的组态与工艺对象的组态。	20014
8010	驱动器故障	由于组态或参数数据有误，而导致驱动器或外部设备出现故障，或通信被中断。	检查设备功能、连接和接线情况。根据当前的消息帧设置，检查 HW Config 中的组态。	20005
8021	凸轮位置无效	凸轮上的实际值无效。	- 重启相应的轴或外部编码器。 - 编码器系统上的实际值输出存在故障（例如，由于驱动器故障）。	40005
8043	命令参数值或默认值无效	命令的工艺功能中的输入参数中存在一个或多个无效的值，或者默认参数无效。示例： - 单个输出凸轮的切换位置在轴的操作范围之外。 - 凸轮轨迹长度 <i>CamTrackLength</i> 具有值 0 或 <0。	检查命令并调整无效的参数。您还应该注意当前使用的默认值。	30001, 40003

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
8045	在当前状态下不允许使用该命令	- 具有更高优先级的命令处于活动状态（例如“MC_Stop”）。 - 工艺对象被禁用或未准备就绪。 - 回原点时收到了测量命令。	- 确认工艺对象的所有排队错误。 - 等待工艺对象的初始化（在 Restart 之后）。 - 启用工艺对象。 - 取消激活“MC_Stop”。 - 请勿在回原点阶段启用任何测量命令。	30006, 30009
8047	无法确认该错误	不能通过调用工艺功能“MC_Reset”确认错误。示例： - 轴已停止，无法确认导致轴停止的错误。 - 在确认期间或重启阶段发生了更多错误。	消除导致错误的所有原因，然后重新尝试确认。请注意，您必须通过断开然后再接通电源来确认特定的错误。	30007
8048	无法重启	工艺对象处于活动状态，或排队的错误阻止系统重启。	消除并确认所有排队的错误，然后禁用该工艺对象。	20012
8049	非法的对象组合	工艺对象的组合有误。示例：	根据原因执行下列测量：	20007
		同步操作中将 Axis_2 设置为 Axis_1 的引导轴；同时，将 Axis_1 设置为 Axis_2 的引导轴。	设置一个已定义的引导轴/跟随轴组合。	
		以不同的优先级处理同步操作中涉及的轴。	将引导轴和跟随轴分配到相同的运行级别（通常为插补器周期时钟）。	
804A	缺少所需的对象连接	没有在组态中考虑引导轴与跟随轴和凸轮盘的所需组合。	在跟随轴的同步对象的“组态”(Configuration) 部分中，设置所需的引导轴和凸轮。	30008
8087	输出凸轮位置非法	至少一个单个输出凸轮位置非法。	检查单个输出凸轮位置。缩短当前延迟/微分时间。	40001
* S7T Config 中，可能与所指的错误有关的中断				

7.12.8 ErrorID — MCDevice/Trace 工艺 DB

适用于固件版本为 V3.1.x 或更高版本的集成工艺

本部分介绍了固件版本为 V3.0.x 的应用。

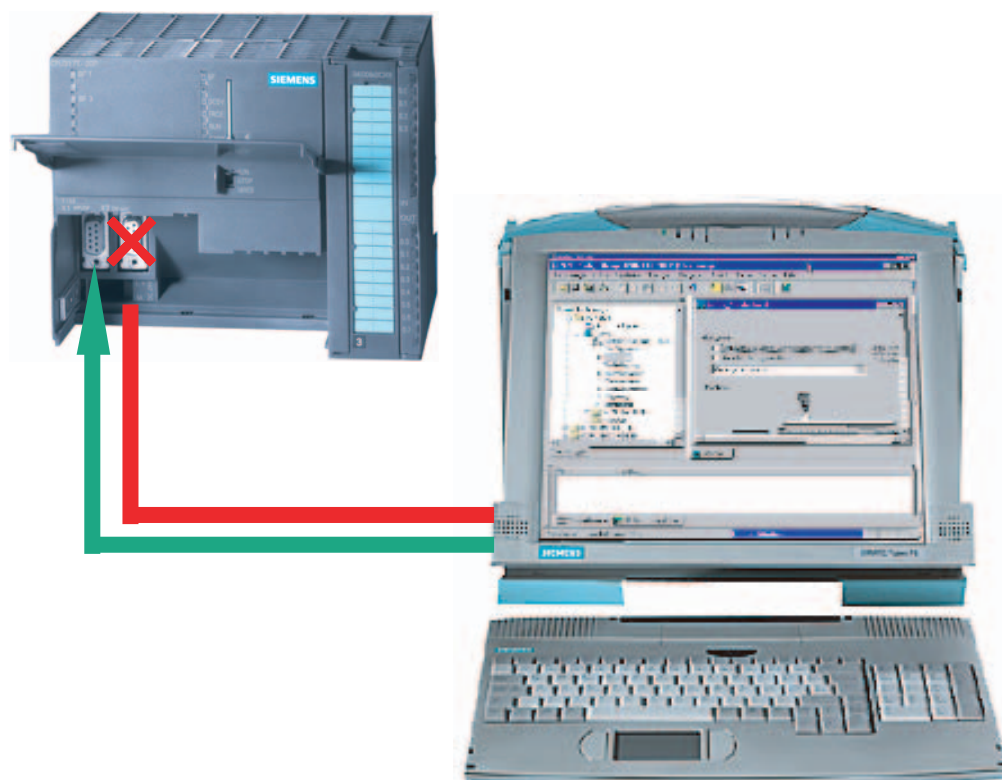
错误

ErrorID	错误消息	可能的原因	补救措施	S7T Config 中的中断 *
8008	工艺 DB 无效	工艺将拒绝该 DB，原因是该 DB 没有有效的内部 ID。该 DB 可能是在复制操作中生成的。	在“工艺对象管理”(Technology Objects Management) 对话框中重新创建 DB，并再次下载组态数据（建立在线/离线一致性）。	
8009	由于该工艺 DB 已被更新的实例所代替，因此被禁用	由于已下载了具有另一 DB 号的更新的实例，因此不再将该 DB 分配给工艺对象。	删除 DB	
* S7T Config 中，可能与所指的错误有关的中断				

装载、测试和诊断

8.1 连接编程设备

所有组态数据和程序都在 PG 或 PC 上创建。



始终使用联机电缆互连编程设备和 T-CPU 的 X1 接口。

编程设备（例如基于文本的显示器）和 OP 有可能不能连接到 X3 接口（DP(DRIVE)）。这些设备的任何操作都将危及 DP(DRIVE) 上的同步操作。

您可以选择将编程设备连接至您系统中的 CP。连接至 CP 时，或在 X1 接口处设置恒定的总线周期时，须考虑延长装载时间。

说明

有关到驱动器路由的详细信息，请参考相应的设备手册。

有关路由的更多信息，请参考 FAQ 中有关驱动器工艺的条目 ID: 26710833。可在 Siemens AG, Automation and Drives, Service & Support 的 Internet 页面上检索该条目 (<http://www.siemens.com/automation/service&support>)。

8.2 下载命令

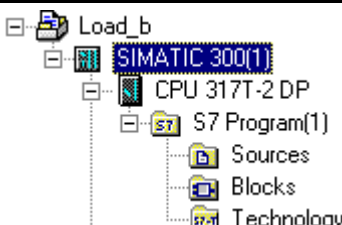
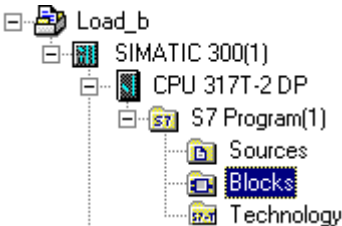
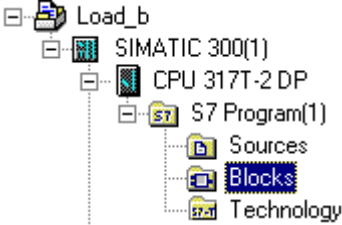
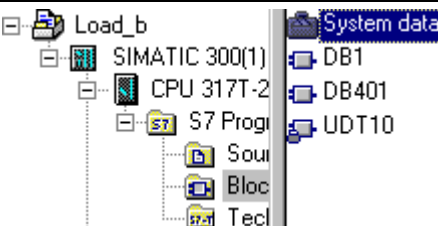
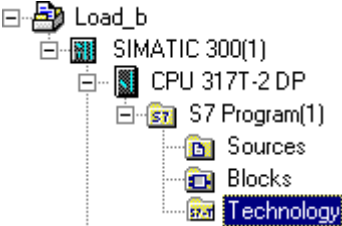
8.2.1 STEP 7 中的装载命令

当您使用 SIMATIC Manager、HW Config 或 S7T Config 时，始终可以将您的组态数据下载到目标系统，也就是下载到您的 T-CPU。您可以下载的对象：

用户程序（OB、FC、FB、DB）	项目的块文件夹中的程序段和 DB。
系统数据	系统数据包含 CPU 的永久的和编程的参数，即集成工艺的硬件组态和所有组态数据。
硬件组态	硬件组态数据是在 HW Config 中创建的，反映站的硬件组态。
工艺	这些数据包含集成工艺的组态数据和工艺对象的组态数据。
集成工艺的固件	集成工艺的当前固件打包到 SDB 中，并保存到 T-CPU 的装载存储器中。固件下载基于下面列出的条件： - 在执行“保存并编译”操作时选中了“生成工艺数据”(Generate technology data) 复选框。该复选框位于 HW Config 中属性对话框的“工艺”(Technology) >“工艺系统数据”(Technology system data) 选项卡中。 如果在执行“保存并编译”操作时未生成工艺系统数据（复选框未选中），则从装载存储器中删除所有固件数据。 - 集成工艺的当前固件尚未保存到装载存储器中。如果它已存在于装载存储器中，则不再重新下载固件。
驱动器组态	在 S7T Config 中，您可以使用集成的 STARTER 组态 MICROMASTER 和 SINAMICS 驱动器。

在下面，您会发现可以使用各种工具下载的对象列表：

在“SIMATIC Manager”中下载

标记	菜单命令	下载的内容
	目标系统 (Target system) > 下载 (Download)	• 用户程序 • 硬件组态 • 工艺 • 集成工艺的固件 ¹⁾
选择不起作用	PLC > 将站上传到 PG (Upload Station to PG)	• 用户程序 • 硬件组态 • 工艺（仅备份副本；无法在 S7T Config 中进行编辑） • 集成工艺的固件
	目标系统 (Target system) > 下载 (Download)	• 用户程序 • 硬件组态 • 工艺 • 集成工艺的固件 ¹⁾
	PLC > 将用户程序下载到存储卡 (Download user program to memory card)	• 用户程序 • 硬件组态 • 工艺 • 集成工艺的固件 ¹⁾ 开始下载之前将删除装载存储器的内容。
	目标系统 (Target system) > 下载 (Download)	• 硬件组态 • 工艺 • 集成工艺的固件 ¹⁾
	不下载	该块文件夹和“工艺对象”(technology objects) 图标仅代表用于组态工艺的对象。

8.2 下载命令

“HW Config”工具的装载命令

标记	菜单命令	下载的内容
不需要/不能进行选择	目标系统 (Target system) - > 下载到模块 (Download to Module)	• 硬件组态 • 工艺 • 集成工艺的固件 ¹⁾

“S7T Config”工具中的装载命令

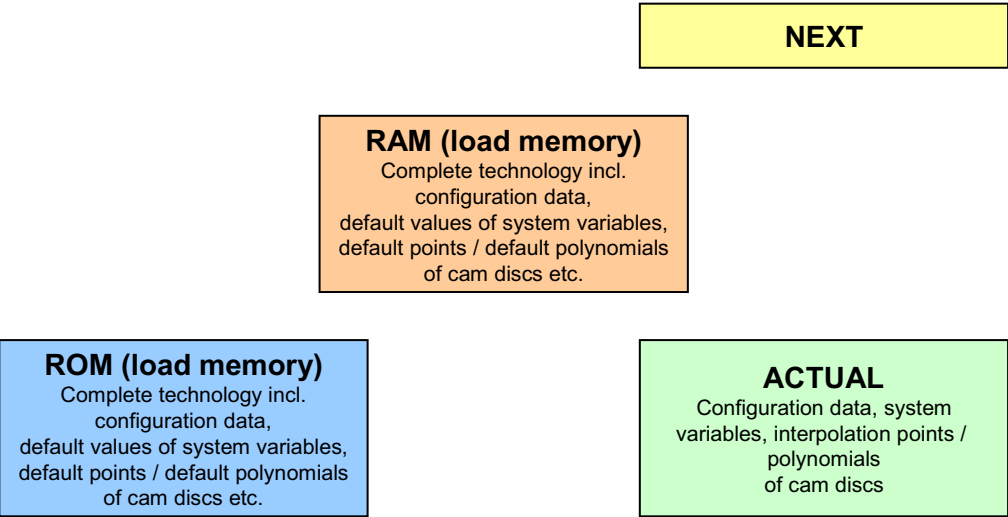
1) 如果集成工艺固件版本与 T-CPU 装载存储器中的版本相同，则 T-CPU 在下载后不自动执行存储器复位。

所描述的全部下载操作也可以定向到编程设备/PC 的 MMC 接口中的 MMC。使用该卡可以减少下载时间。同样也适用于 T-CPU MICROBOX T 中的 CF 卡。

8.2.2 S7T Config 中的装载命令 — 存储器组织

8.2.2.1 集成工艺的存储器组织

下图显示了与集成工艺关联的存储区：



ROM（装载存储器）

“ROM”存储器是 T-CPU 中装载存储器的一部分。对于 CPU 31xT，该存储器位于微型存储卡 (Micro Memory Card, MMC) 上；对于 Microbox T，则位于 CF 卡上。整个“工艺”都存储在该存储区中。这包括以下数据：

- 组态数据（在 S7T Config 中定义）
- 系统变量的默认值
- 凸轮盘的默认插补点/多项式（在 S7T Config 中组态）。

RAM（装载存储器）

“RAM”存储区也位于 T-CPU 的装载存储器中。整个“工艺”都装载到该存储区中。这包括以下数据：

- 组态数据（在 S7T Config 中定义）
- 系统变量的默认值
- 凸轮盘的默认插补点/多项式（在 S7T Config 中组态）。

ACTUAL

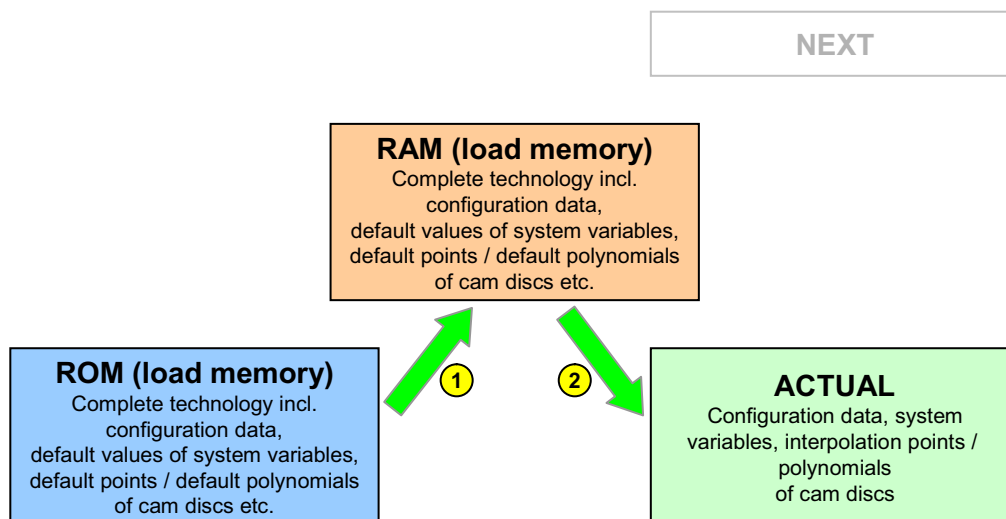
“ACTUAL”存储区是集成工艺的组件。“ACTUAL”存储区用于保存和处理实际运行系统的组态数据、系统变量以及凸轮盘的插补点和多项式。

NEXT

“NEXT”存储区也是集成工艺的组件。该存储区用于临时存储未立即激活的更改。

8.2.2.2 接通电源和 CPU 存储器复位

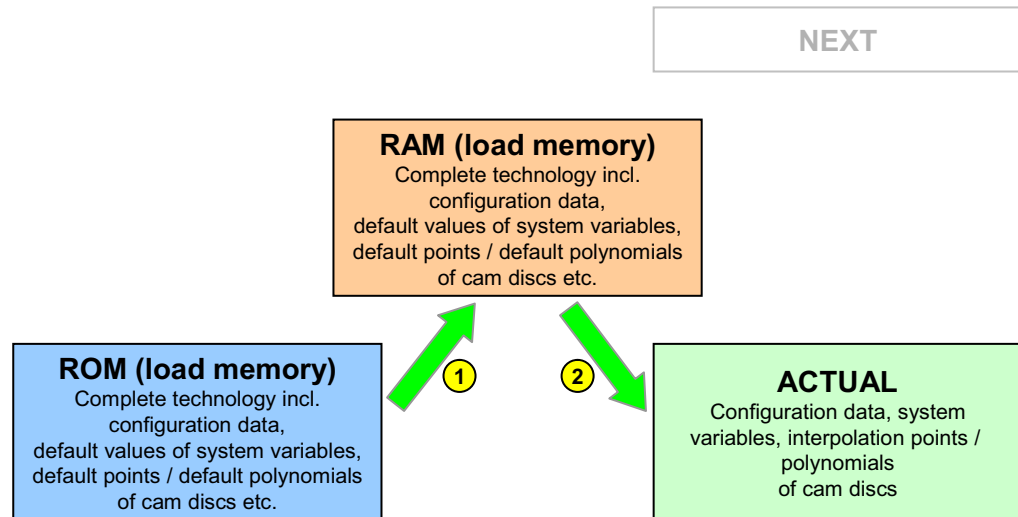
下图显示了接通电源以后以及 T-CPU 的存储器复位后将加载的存储区。



- ① 整个工艺将从 ROM 传送到 RAM。
- ② 组态数据、系统变量的缺省值（包括凸轮盘的缺省插补点/多项式）将从“RAM”传送到“ACTUAL”。

8.2.2.3 重启工艺对象

下图显示了重启（例如通过调用工艺功能“MC_Reset”、输入参数 *Restart = TRUE*）后哪些存储区将复制到工艺对象。



- ① 相关工艺对象的组态数据和系统变量的缺省值将从“ROM”传送到“RAM”。
如果工艺对象是凸轮盘，将包括缺省插补点/多项式。
- ② 组态数据、系统变量的缺省值（包括凸轮盘的缺省插补点/多项式）将从“RAM”传送到“ACTUAL”。

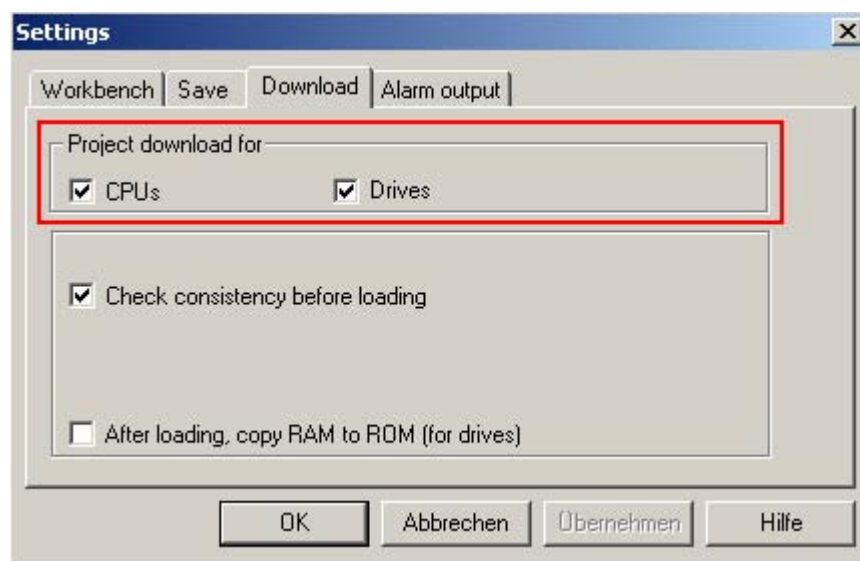
8.2.2.4 S7T Config 中的装载命令

目标系统/目标设备

S7T Config 在处理下载命令时区分“目标系统”和“目标设备”对象。

“目标设备”始终是在浏览器中选择的对象。如果目标设备是驱动器，则包括所有嵌套的驱动器对象。

通过选择**选项 (Options) > 自定义 (Customize)** 命令定义“目标系统”。



在“下载”(Download) 选项卡中定义“目标系统”的目标设备。

- 如果 T-CPU 应该属于“目标系统”，请设置“CPU”复选框。
- 如果浏览器中列出的所有驱动器都应该属于“目标系统”，请激活“驱动器”(Drives) 复选框。

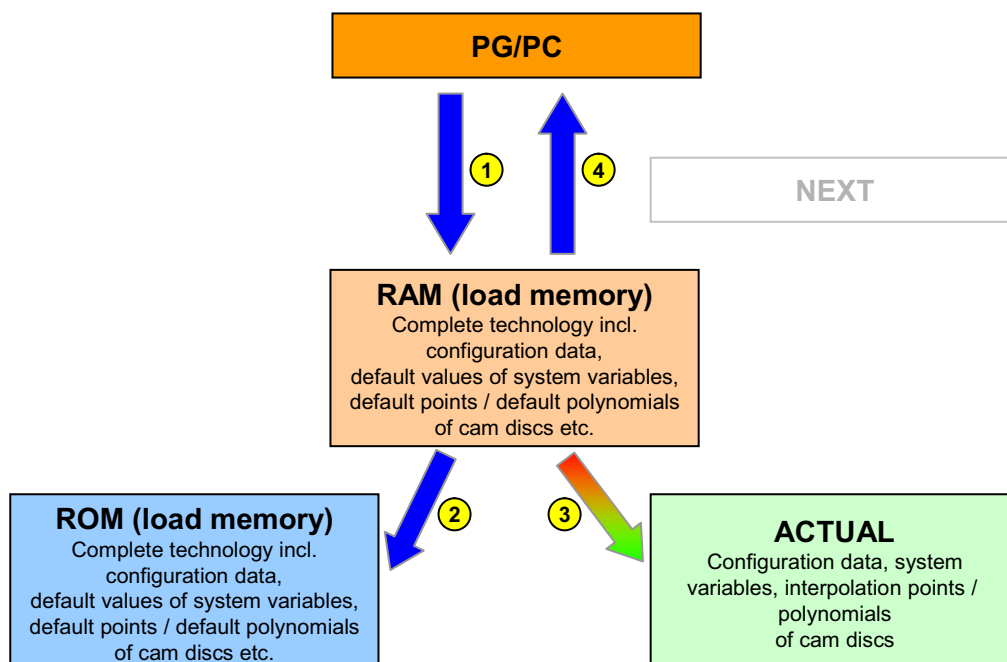
说明

“目标系统”的目标设备必须在线。

通过选择**目标系统 (Target system) > 选择目标设备 (Select target device)** 命令，定义在线连接的目标设备。

下载至 T-CPU

下图显示了存储器对将数据下载到 T-CPU 的 S7T Config 命令的响应。



菜单命令或快捷菜单命令：

- 项目 (Project) > 下载到目标系统 (Download to target system)
- 目标系统 (Target system) > 下载 (Download) > 项目到目标系统 (Project to target system)
- “目标系统”(Target system) > “下载”(Download) > “到目标设备”(to target device) (如果工艺是在浏览器中选择的)

菜单命令目标系统 (Target system) > 上传 (Upload) > 组态数据到 PG (Configuration data to PG)

- ① 项目的增量数据将从编程设备/PC 下载到“RAM”。
- ② 项目的增量数据将从“RAM”下载到“ROM”。
- ③ 当 T-CPU 从 STOP 转换到 RUN 时，组态数据、系统变量的默认值和默认插补点/多项式将从“RAM”传送到“ACTUAL”。

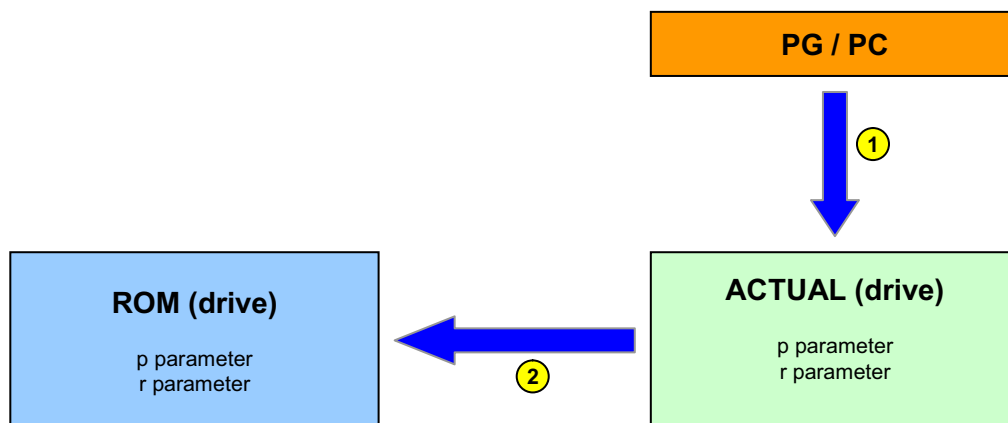
菜单命令目标系统 (Target system) > 上传 (Upload) > 组态数据到 PG (Configuration data to PG)

- ④ 该命令将组态数据从“RAM”上传到编程设备/PC 上的项目数据。

8.2 下载命令

下载到驱动器

下图显示了存储器对将数据下载到驱动器的 S7T Config 命令的响应。



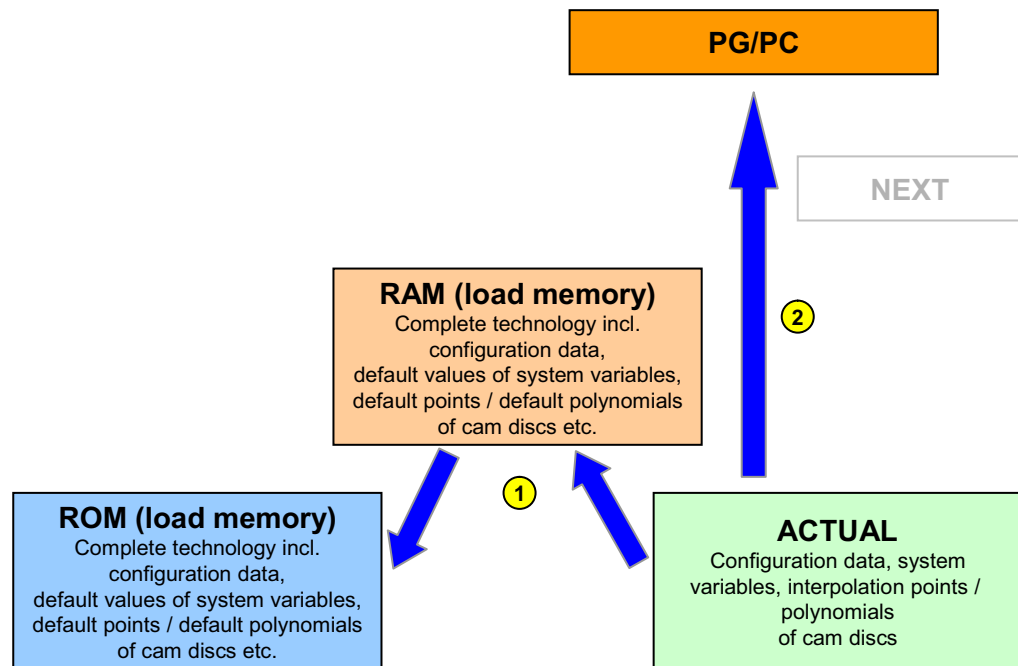
菜单命令或快捷菜单命令：

- 目标设备 (Target device) > 下载到目标设备 (Download to target device)
- 目标系统 (Target system) > 下载 (Download) > 到目标设备 (To target device)

- ① 将驱动器特定的项目数据从 PG/PC 下载到驱动器的“ACTUAL”。
- ② 将驱动器特定的项目数据从“ACTUAL”传送到驱动器的“ROM”。

8.2.2.5 从 Actual 复制到 ROM

下图显示了通过选择“目标系统”(Target system) > “从 Actual 复制到 ROM”(Copy Actual to ROM) 命令复制的存储区。



- ① 组态数据将从“Actual”被传送到“RAM”，然后从“RAM”再传送到“ROM”。
- ② 如果您在菜单的对话框中设置了“将组态数据上传到 PG”(Upload configuration data to PG)，项目 1 中定义的组态数据也会上传到编程设备/PC。

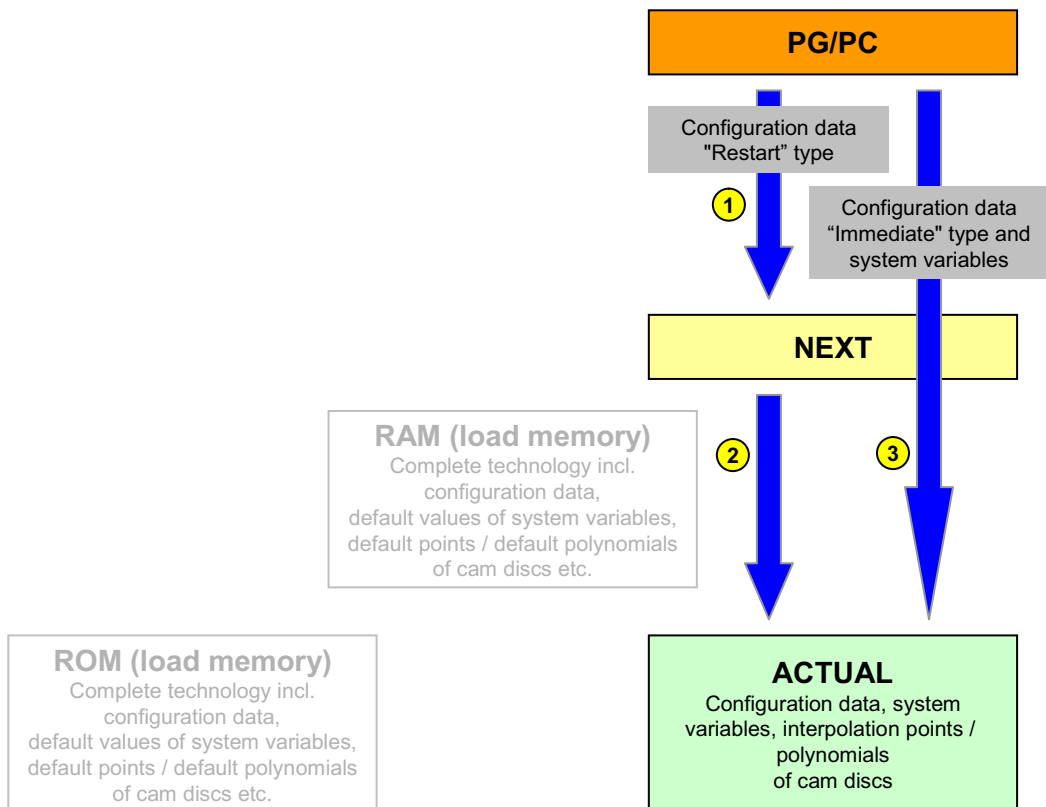
8.2.2.6 S7T Config 中的在线更改

下图显示了集成工艺的存储区上的在线更改的作用。

对话框中的在线更改、工艺对象的专家列表中的在线更改以及通过调用工艺功能“MC_WriteParameter”进行的在线更改

在 S7T Config 的在线模式下，关闭对话框或专家列表中的字段时，将接受已修改的值并将其传送到下面显示的存储区。这也适用于工艺功能“MC_WriteParameter”的写入操作。

您必须区分“重启”类型和“立即”类型的已更改的组态数据。在线模式下，您可以通过专家列表的“作用”(Effects) 列中的条目或参数列表中的条目来识别类型。

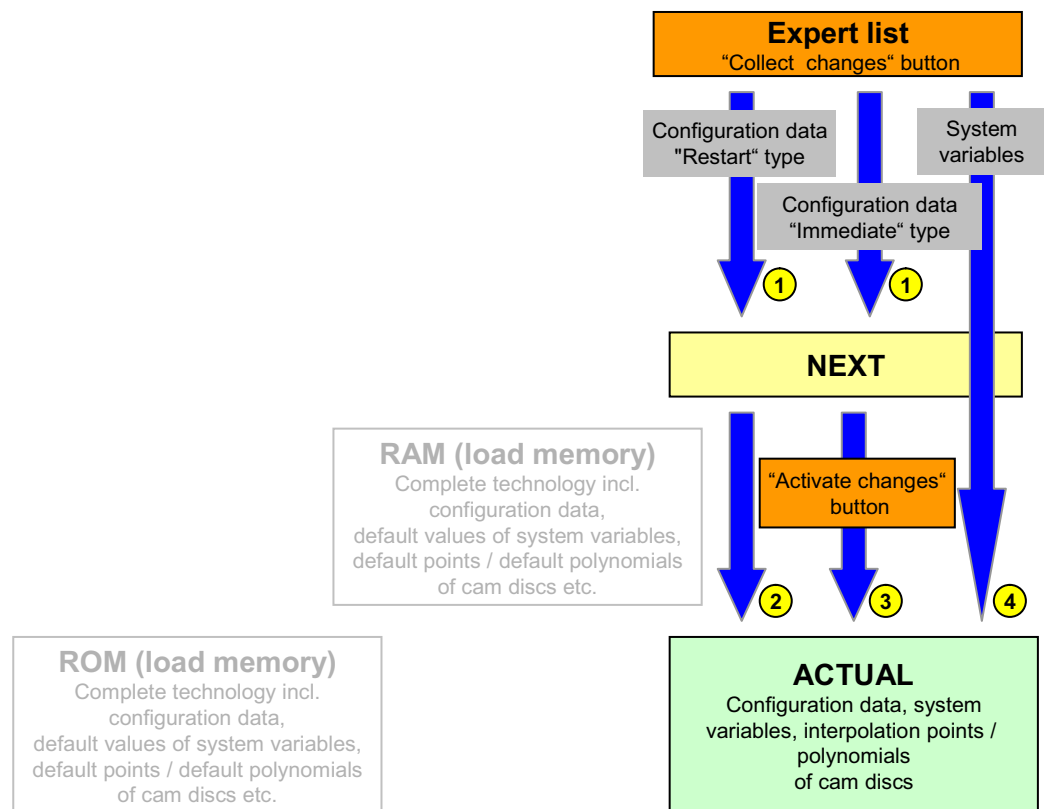


- ① 在在线模式下修改的“重启”类型的组态数据最初被下载到“NEXT”存储区。
- ② 当执行工艺对象“重启”（例如：通过使用工艺功能“MC_Reset”设置输入参数 *Restart = TRUE*）时，更改的值将从存储区“NEXT”传送到“ACTUAL”。
- ③ “立即”类型的组态数据的更改值和系统变量的更改值将直接下载到“ACTUAL”。将绕过存储区“NEXT”。

使用“集中更改”(Collect changes) 按钮功能的在线更改

在在线模式下，工艺对象的专家列表具有“集中更改”(Collect changes) 和“激活更改”(Activate changes) 按钮。编辑选定的组态数据时，使用“集中更改”(Collect changes) 按钮来保持一致性。

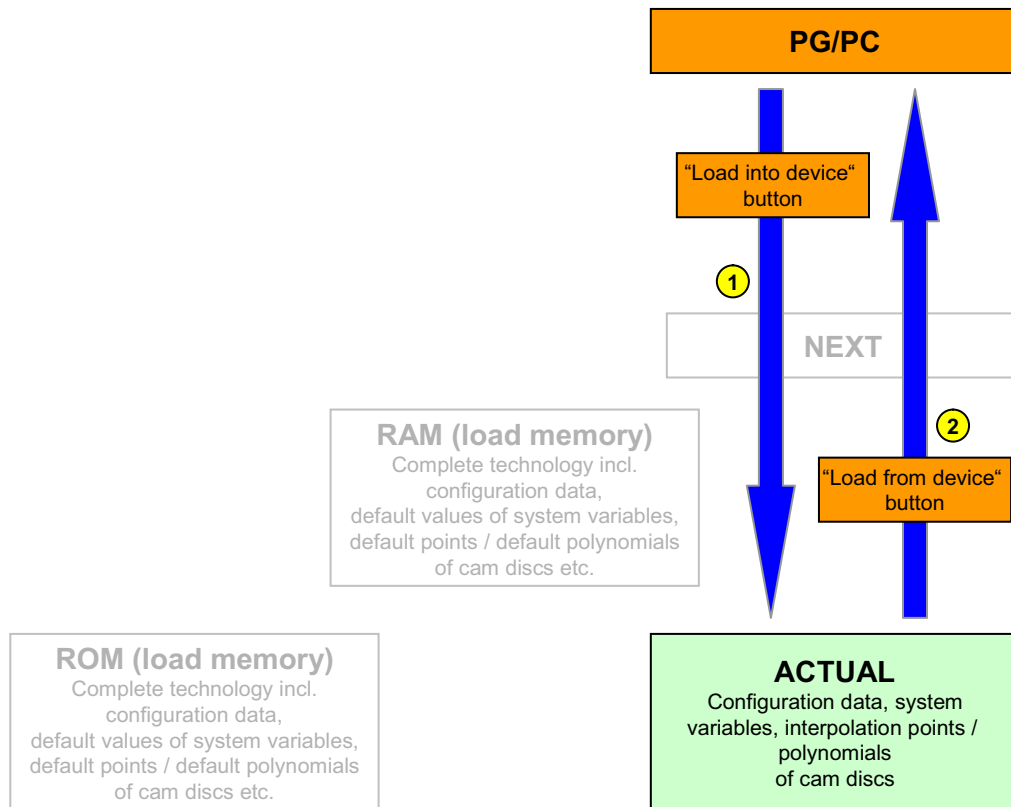
通过单击“集中更改”(Collect changes)，您可以打开组态数据的专家列表中的“下一个值”(Next value) 列，并灰显“实际值”(Actual value) 列。



- ① “下一个值”(Next value) 列中更改的任何值都将立即传送到存储区“NEXT”。这也适用于“重启”和“立即”类型的组态数据。
- ② 如果将“重启”应用到工艺对象（例如通过使用“MC_Reset”工艺功能设置输入参数 *Restart = TRUE*），则更改的值将从存储区“NEXT”传送到“ACTUAL”（仅适用于“重启”类型的组态数据）。
- ③ 如果“集中更改”(Collect changes) 处于活动状态，“立即”类型的所有组态数据也将传送到存储区“NEXT”。
单击“激活更改”(Activate changes) 将所有收集的更改应用于集成工艺。
- ④ 系统变量的更改值将直接下载到“ACTUAL”。将绕过存储区“NEXT”。

凸轮盘的在线更改

在 S7T Config 的在线模式下，您可以将凸轮盘从凸轮盘创建工具“CamEdit”传送到“ACTUAL”，反之亦然。您可以仅将 T-CPU 当前未使用的凸轮盘传送到“ACTUAL”。



- ① 单击“下载到设备”(Download to device) 以插补 CamEdit 中表示的凸轮盘，并将其下载到集成工艺的存储区“ACTUAL”。
- ② 单击“从设备上上传”(Upload from device) 将插补的凸轮盘从存储区“ACTUAL”上传到 CamEdit。

8.3 工艺系统时钟

8.3.1 设置工艺系统周期时钟

集成工艺在单个处理周期内处理您的任务。根据任务，您可以调整工艺系统时钟，使其符合您的要求，并提高运动控制应用程序的性能。

在 S7T Config 中选择工艺，然后选择目标系统 (Target system) > 设置系统时钟 (Set system clocks)。

System Cycle Clocks - Technology

Cycle clock ratios

Cycle clock name	FACTOR	Reference cycle clock	Cycle time
Bus cycle clock			5 ms
Servo	1	x Bus cycle clock =	5 ms
Ipo	3	x Servo =	15 ms
Ipo_2	2	x Ipo =	30 ms

Technology parameterization

Job monitoring time (1 - 5,000 ms): ms

Technology DB update time (n * IPO): ms

Maximum shutdown time (1000 - 60000 ms): ms

Number of tolerated IPO overflows:

Number of tolerated IPO2 overflows:

Network settings PROFIBUS(1)

☒ Equidistant bus cycle activated

Equidistant cycle: ms

OK Cancel Help

周期时钟比率

在“周期时钟比率”(Cycle clock ratio) 中设置集成工艺的时钟。从相应下拉列表选择一个合适的比率。绝对周期时间会显示在右侧的列中。在此对话框中定义的周期时间：

- **总线时钟**

不能在此对话框中设置总线时钟。“总线时钟”在 HW Config 中设置为 DP(DRIVE) 设置中的“等距 DP 周期”。DP(DRIVE) 接口的等距 DP 周期构成了工艺系统时钟的基础时钟。

DP 周期时钟必须为 0.250 ms 的整数倍。如果不是，则在 HW Config 中更改该值。

- **伺服 (Servo) (位置控制周期)**

系统还将在此周期内计算轴位置控制。位置控制周期可以设置为 DP 周期的整数倍。T-CPU 支持因子 1 和 2。从下拉列表框中选择一个倍增因子，以设置需要的位置控制周期。

通常，应输入因子 1。如果设置因子 2，则会降低控制系统的动态性能。但是，可减少计算时间以用于处理其它任务。最大和平均命令执行时间可以从“MCDevice”工艺数据块的变量 *MaxLoopDuration* 和 *CmdLoopDuration* 中读取。

必须在驱动器上将 DP 周期和位置控制周期之间的减速比设置为“主站应用周期”。要相互进行状态监视，该设置是必需的。有关详细信息，请参考驱动器文档。

- **lpo (插补器周期)**

默认情况下，在“插补器周期”内计算轴、外部编码器、输出凸轮和测量输入。从下拉列表框中选择一个倍增因子，以设置需要的插补器周期

- **lpo_2 (插补器周期 2)** “插补器周期 2”用于控制具有较低优先级的轴的运动。该值确定了用于计算较低优先级等级的轴、外部编码器、输出凸轮和测量输入的时序。从下拉列表框中选择一个因子，以设置需要的插补器周期 2。

工艺参数化

在对话框的左边单击下拉框的选择按钮以查看该区域中的内容。该区域中的输入框：

- **命令监视时间 (Command monitoring time) (Job_Processing_Task)**
设置该时间以确定所有并行的运动控制命令的最大执行时间。
如果超出了命令监视时间，则 T-CPU 会更改为安全操作模式 STOP。一条错误消息将输入到诊断缓冲区中。
- **工艺 DB 更新时间 (Technology DB update time)**
从下拉列表选择一个条目以确定集成工艺用于更新工艺 DB 的基础周期（另请参阅“更新工艺 DB”）。频繁更新操作将延长运动控制命令的执行时间（请参阅“命令监视时间”）。
如果在上一更新周期尚未完成时即启动一个新的工艺 DB 更新，我们将这种情况称为“溢出”。当使用固件版本为 V3.1.x 或更低版本的集成工艺运行时，T-CPU 将在检测到工艺 DB 更新溢出时跳转到 STOP。固件版本为 V3.2.x 或更高版本的集成工艺容许此类溢出，这意味着将放弃不可执行的更新，并且不会调用工艺同步中断 OB 65。可以从工艺 DB“MCDevice”上的 *TODBTaskOverflows* 变量中读取溢出的次数，从而了解工艺 DB 更新的失败率。
- **最大关闭时间 (Maximum shutdown time)**
输入集成工艺的关闭监视时间。如果在该时间内未完成集成工艺的关闭，T-CPU 将自动跳转到 STOP。
- **IPO 溢出/IPO2 溢出的容许个数 (Number of tolerated IPO overflows / IPO2 overflows)**
当插补器执行时间超出所组态的插补器周期时，将发生“IPO 溢出”。在这种情况下，以下插补器无法在选定周期内启动，将被忽略。如果插补器的处理在下一个周期内完成，则插补器将正常启动。
每个忽略的插补器将被添加到“IPO 溢出”数目中。从下拉列表框中选择 T-CPU 进入 STOP 操作模式之前容许的连续“IPO 溢出”数目。可以单独设置 IPO 和 IPO2 的溢出数目。

网络设置

在此区域中检查 HW Config 中的 DP(DRIVE) 设置。

8.3.2 检查集成工艺上的负载

工艺对象和凸轮盘 IPO 使用存储器的典型用法在 T-CPU 的“CPU 数据”文档中进行了介绍。根据该文档中介绍的值确定存储区分配的近似百分比值。该近似值通常不应超过 80 %。

您可以检查 S7T Config 中的当前存储器负载。

检查集成工艺上的负载

如何检查集成工艺上的负载。

1. 下载 T-CPU 所需的所有工艺对象。

验证凸轮盘是否包含所需的所有插补点。使用工艺功能

“MC_CamClear”、“MC_CamSectorAdd”和“MC_CamInterpolate”在运行时动态更改凸轮时，请使用这些功能创建插补点。

2. 在 S7T Config 中更改为“在线”模式。

3. 在 S7T Config 中选择“目标系统”(Target system) > “设备诊断”(Device diagnostics) 命令。

“设备诊断”(Device diagnostics) 对话框随即打开。

4. 选择“系统负载”(System load) 选项卡。

该选项卡显示当前的存储器负载和集成工艺的“由系统任务导致的 CPU 负载”(CPU load due to system tasks)。指示的存储区分配值通常不应超过 80 %。

“由系统任务导致的 CPU 负载”(CPU load due to system tasks) 指示了集成工艺上由位置控制和插补器周期 (ipo 和 ipo2) 导致的负载，用百分比表示（例如 40%）。剩余的时间段（例如 60%）可用于工艺 DB 更新和命令处理。

您可以通过调整系统时钟修改 CPU 负载。

- 较短的系统时钟可以提高控制性能，但是会减少用于命令处理的时间段。工艺 DB 更新可能需要一个较长的时钟。
- 但是较长的系统时钟会增加命令处理和工艺 DB 更新的计算时间。这会降低控制性能。

“由系统导致的 CPU 负载”(CPU load due to system tasks) 的值通常应该介于 40% 和 60% 之间。

8.4 优化位置控制器

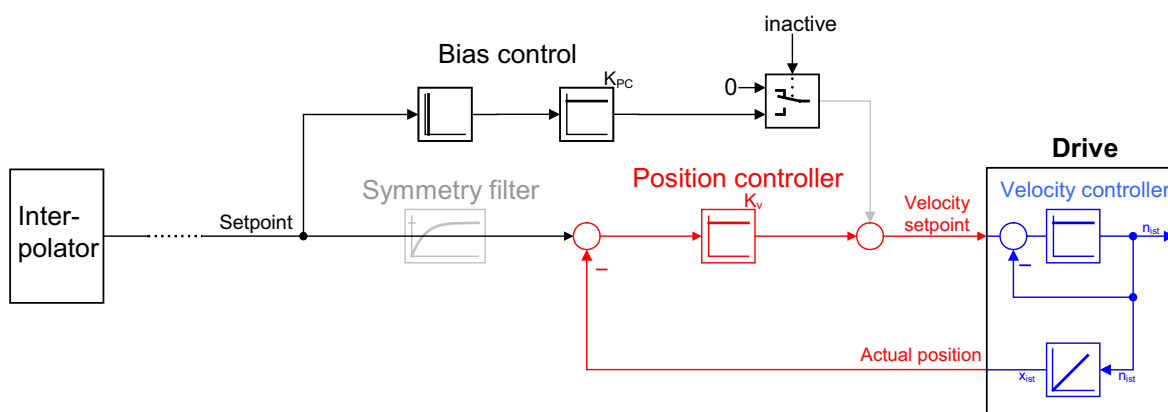
8.4.1 优化位置控制器 — 概述

如果您将焦点置于没有振荡的精确驱动器定位上，请调节轴的位置控制器。以下描述了一个实际示例，可以根据需要进行相应改变。

8.4.2 准备调谐

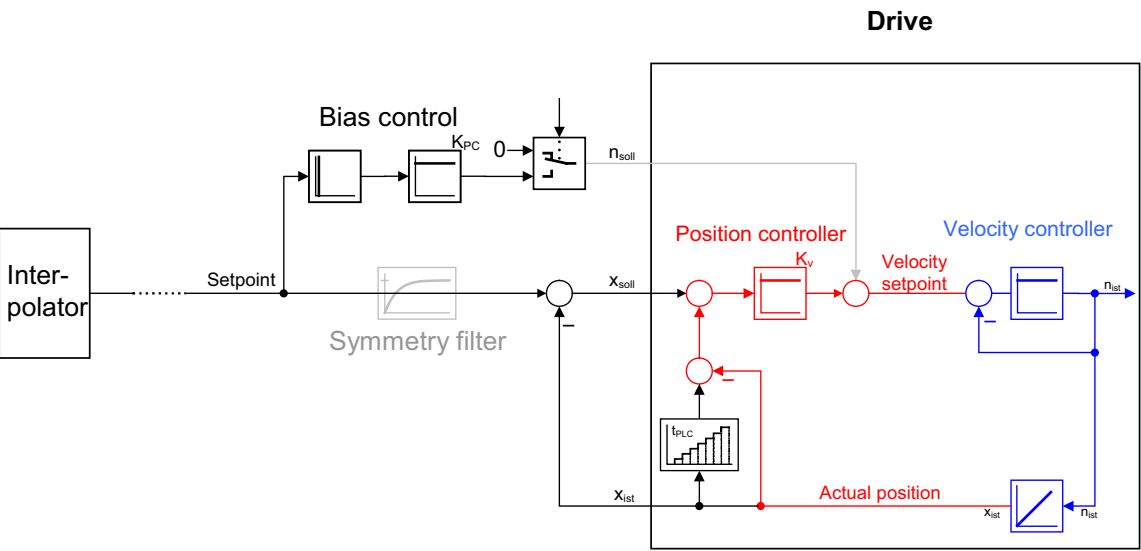
必须仅使用位置控制器的基本功能来进行调节，即必须禁用“预先控制”和“平衡滤波器”功能。

没有 DSC 的位置控制回路的方框图



在系统中操作没有 DSC（Dynamic Servo Control，动态伺服控制）的位置控制回路时，系统会计算 T-CPU 中的位置控制器和驱动器的速度控制器。在不同的周期内进行计算参数。

具有 DSC 的位置控制回路的方框图



对于具有 DSC（Dynamic Servo Control，动态伺服控制）的位置控制，系统将计算驱动器中的位置控制器和速度控制器。这使您可以为位置控制器设置高出很多的增益系数 (K_v)。位置和速度控制参数在同一周期中计算。因此，位置控制及其次级控制回路（即速度控制器）将以完全相同的速度计算。

如果您要在具有 DSC 的调节状态下运行驱动器，请勿在调节期间禁用动态伺服控制 (DSC, Dynamic Servo Control)。

先决条件

- 您已组态轴，不包括加速限制和控制器设置的输入。
- 您已调试较低级别的控制回路，并已设置速度控制回路，该速度控制回路在应用预先控制设置时不受超调限制。
- 驱动器、编码器和工艺对象比例匹配。

通过进行下一部分中概述的设置来调节位置控制器。在更改初始设置的所有值之前，请记住记下这些值，以便在调节后可以撤消这些设置。

- **禁用速度预先控制**
在轴 **(Axis) > 控制 (Control)** 对话框的“静态控制器数据”(Static controller data) 上，取消激活“预先控制”(Pre-control) 复选框。
- **精插补器“恒定的加速插补”**
从“精插补器”(Fine interpolator) 下拉列表框中选择“恒定的加速插补”(constant acceleration interpolation)。此下拉列表框位于“轴”(Axis) > “控制”(Control) 对话框的“静态控制器数据”(Static controller data) 选项卡 中。
- **禁用漂移补偿**
在轴 **(Axis) > 控制 (Control)** 对话框的“静态控制器数据”(Static controller data) 选项卡 中，取消激活“漂移补偿”(Drift compensation) 复选框。
- **禁用摩擦补偿**
在轴 **(Axis) > 控制 (Control)** 对话框的“摩擦补偿”(Friction compensation) 选项卡 中，取消激活“启用摩擦补偿”(Enable friction compensation) 复选框。此设置仅在专家模式下可用。
- **取消激活平衡滤波器**
将组态数据元素
TypeOfAxis.NumberOfDataSets.DataSet_x.ControllerStruct.PV_Controller.balanceFilterMode 的值改为 *Off*。
此外，也可以使用工艺功能“MC_WriteParameter”（通过值为 89 的参数号 2035 的）取消激活平衡滤波器。

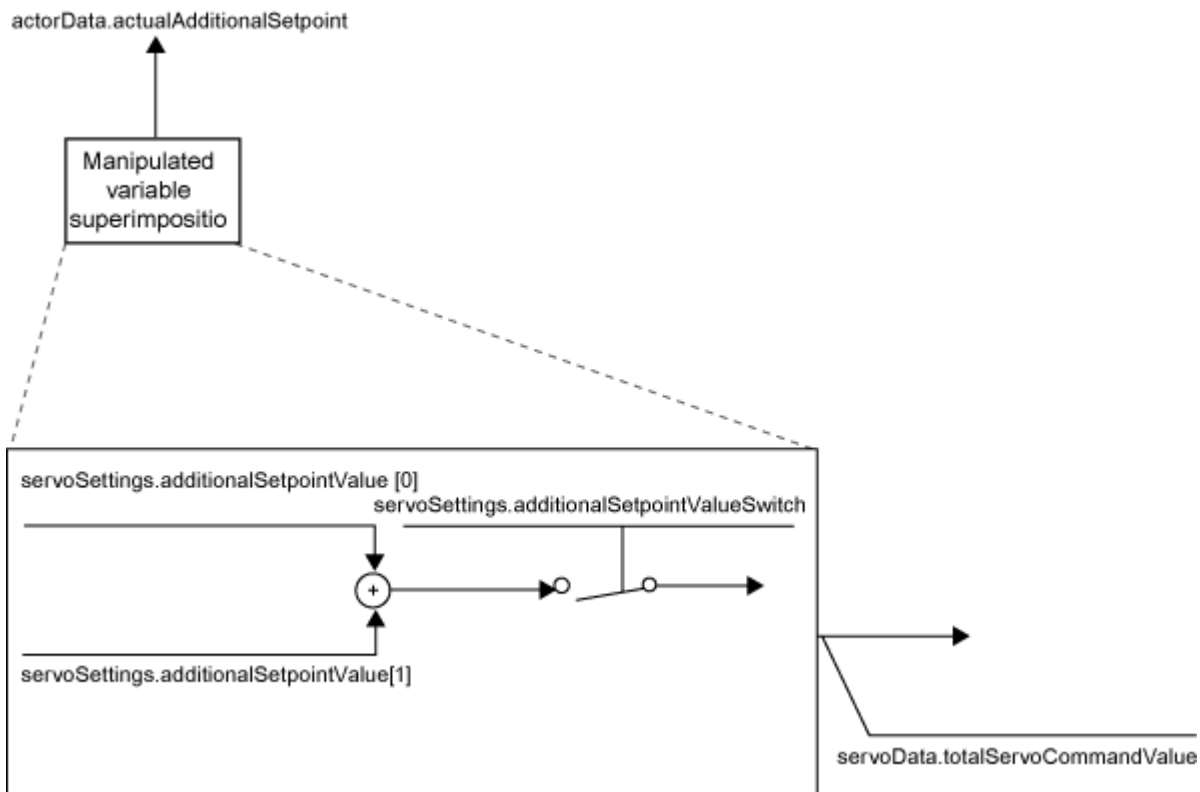
8.4 优化位置控制器

- 禁用跟随误差监视

在**轴 (Axis) > 监视 (Monitoring)** 对话框的“跟随误差监视”(Following error monitoring) 选项卡中，取消激活“动态跟随误差监视激活”(dynamic following error monitoring active) 复选框。

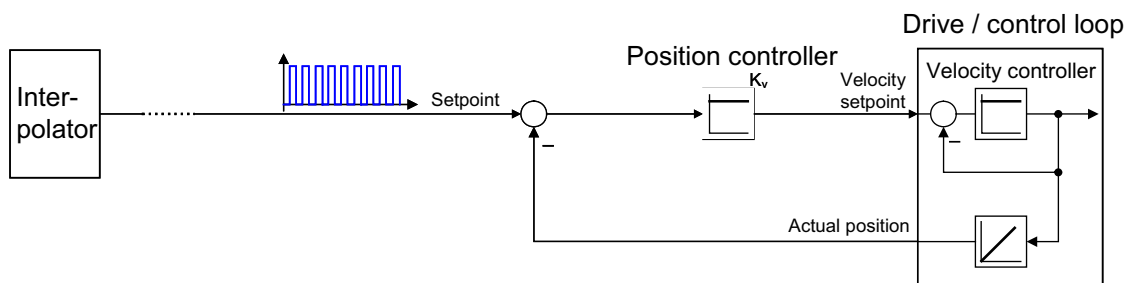
- 定位窗口/静止窗口

在**轴 (Axis) > 监视 (Monitoring)** 对话框的“定位窗口”(Positioning window) 和“静止窗口”(Standstill window) 输入框中，输入一个较高的值。较高的值可以防止这些监视功能在调节期间发出任何响应。



8.4.3 用于控制器调谐的用户程序

用户程序中实现了一个测试程序，它在位置控制器处设置了阶跃响应值。



以下示例显示了一个实际功能块。您可以将其内容从在线帮助中复制到 STEP 7，作为 STL 源文件。

该功能块生成一个阶跃函数。使用工艺功能“MC_MoveRelative”，轴位置会增加 100 mm，然后返回到其初始位置。

FB 130 PositionControl 的输入参数

参数	数据类型	描述
<i>Axis_No</i>	INT	输入轴工艺 DB 的编号
<i>Enable_Axis</i>	BOOL	通过设置该参数可以启用和禁用轴 <i>TRUE</i> = 启用轴 <i>FALSE</i> = 禁用轴
<i>Start_Function</i>	BOOL	<i>TRUE</i> = 启动功能 <i>FALSE</i> = 停止功能
<i>Reset_Axis</i>	BOOL	使用该参数“复位”轴
<i>Restart_Axis</i>	BOOL	使用该参数“重启”轴

先决条件:

您的项目必须包含以下工艺功能。将工艺功能从 S7-Technology 的块库复制到您的块文件夹:

- FB 401 MC_Power
- FB 402 MC_Reset
- FB 411 MC_MoveRelative

8.4 优化位置控制器

生成功能块的步骤:

将下面显示的块代码复制到 STL 源文件。如果您的项目中已存在 FB 130，请更改 FB 130 的块编号。从 STL 源中编译块。

STL 源代码

```
FUNCTION_BLOCK FB 130
TITLE = PositionControl
VERSION :1.0
VAR_INPUT
    Axis_No :INT;
    Enable_Axis :BOOL ;
    Start_Function :BOOL ;
    Reset_Axis :BOOL ;
    Restart_Axis :BOOL ;
END_VAR
VAR
    MC_Power_Axis :FB 401; // MC_Power
    MC_Reset_Axis :FB 402; // MC_Reset
    MC_Move_Axis :FB 411; // MC_MoveRelative
    Start_Optimation :BOOL ;
    Forward :BOOL ;
    Backward :BOOL ;
    Forward_Store :BOOL ;
END_VAR
BEGIN
NETWORK
TITLE =Enable Axis
    U      #Enable_Axis;
    =      #MC_Power_Axis.Enable;
    L      #Axis_No;
    T      #MC_Power_Axis.Axis;
NETWORK
TITLE =Reset/Restart Axis
    L      #Axis_No;
    T      #MC_Reset_Axis.Axis;
    U      #Reset_Axis;
    O      #Restart_Axis;
    =      #MC_Reset_Axis.Execute;
    U      #Restart_Axis;
    =      #MC_Reset_Axis.Restart;
```

FUNCTION_BLOCK FB 130

NETWORK

TITLE =Generation of Test function for Optimization of Axis

```

L      #Axis_No;
T      #MC_Move_Axis.Axis;
U      #Start_Function;
FP     #Start_Optimization;
O(     ;
U      #Start_Function;
U      #MC_Move_Axis.Done;
UN     #Forward_Store;
)      ;
=      #Forward;
U      #Forward;
S      #Forward_Store;
U      #Forward;
SPB    FWD:
U      #Start_Function;
U      #Forward_Store;
U      #MC_Move_Axis.Done;
=      #Backward;
U      #Backward;
R      #Forward_Store;
U      #Backward;
SPB    BWD;
SPA    GO:
FWD:L  1.000000e+002;
T      #MC_Move_Axis.Distance;
L      1.000000e+002;
T      #MC_Move_Axis.Velocity;
L      1.000000e+004;
T      #MC_Move_Axis.Acceleration;
T      #MC_Move_Axis.Deceleration;
SPA    GO:
BWD:L  -1.000000e+002;
T      #MC_Move_Axis.Distance;
L      1.000000e+002;
T      #MC_Move_Axis.Velocity;
L      1.000000e+004;
T      #MC_Move_Axis.Acceleration;
T      #MC_Move_Axis.Deceleration;
GO:U   #Forward;
O      #Backward;
=      #MC_Move_Axis.Execute;

```

8.4 优化位置控制器

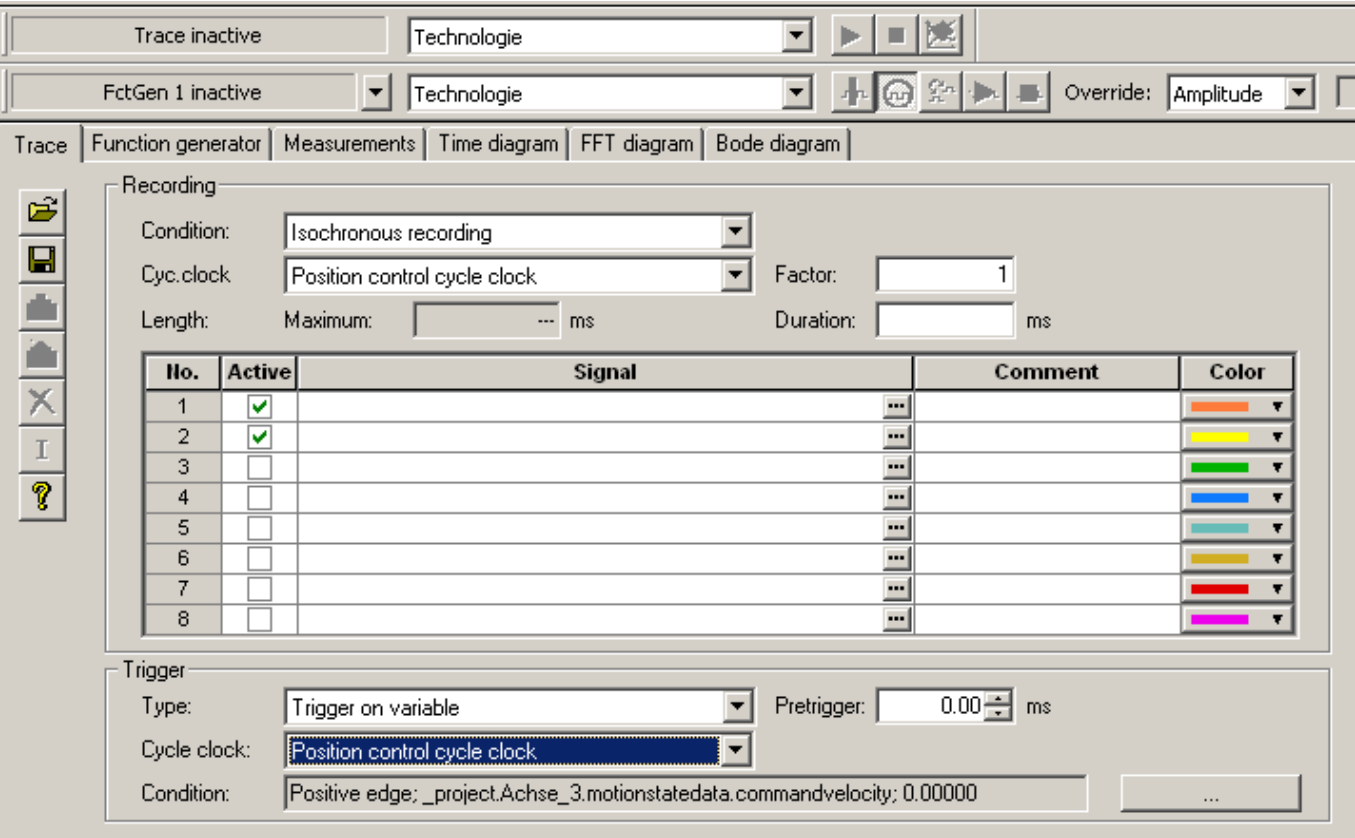
FUNCTION_BLOCK FB 130

```
NETWORK
TITLE =Call Motion Commands
    CALL #MC_Power_Axis ;
    CALL #MC_Reset_Axis ;
    CALL #MC_Move_Axis ;
END_FUNCTION_BLOCK
```

8.4.4 优化位置控制器

在该过程中，您可以调整位置控制器的增益系数 Kv。在定位期间使用 S7T Config 的 TraceTool 记录和分析轴的速度曲线。先前描述的用户程序可以控制轴在两个位置之间的运动。

TraceTool 中的设置



在 S7T Config 的 TraceTool 中，设置以下值：

“信号”(Signals) 下拉列表

信号	值
信号 1	<code>_to."AxisName".motionstatedata. commandvelocity</code>
信号 2	<code>_to."AxisName".motionstatedata. actualvelocity</code>

可在相应选择窗口的“项目名称”(Project name) > “工艺名称”(Technology name) > TO > “轴名称”(Axis name) 路径中获得记录信号。

“记录”(Recording) 下拉列表

记录	值
条件	同步记录
时钟	位置控制周期
系数	1
持续时间	大约 5000 ms 或更长

“触发”(Trigger) 下拉列表

触发	值
类型	触发变量 — 上升沿
变量	<code>_to."AxisName".motionstatedata. commandvelocity</code>
时钟	位置控制周期
预触发	大约 6 ms 或更长
阈值	大约 0.01000 (该值被用作触发阈值)

这些变量位于相应选择窗口的“项目名称”(Project name) > “工艺名称”(Technology name) > TO > “轴名称”(Axis name) 路径中。

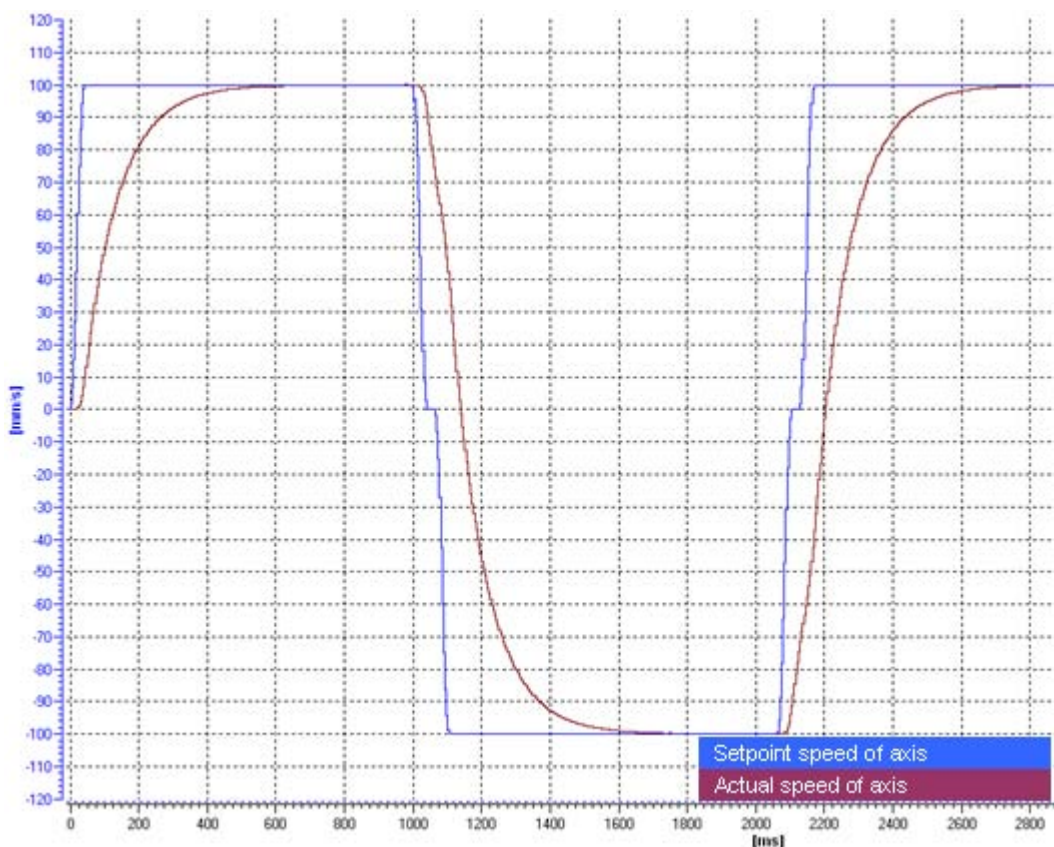
记录的跟踪趋势的分析以及 Kv 系数的调整

在下面显示的趋势中，我们采用一个较低的增益系数 Kv，通过该系数对位置控制器调节进行迭代调整。

在用户程序中通过工艺功能“MC_WriteParameter”的参数 2033 或在组态数据元素 *TypeOfAxis.NumberOfDataSets.DataSet_x.ControllerStruct.PV_Controller.kv* 中设置 Kv 系数。

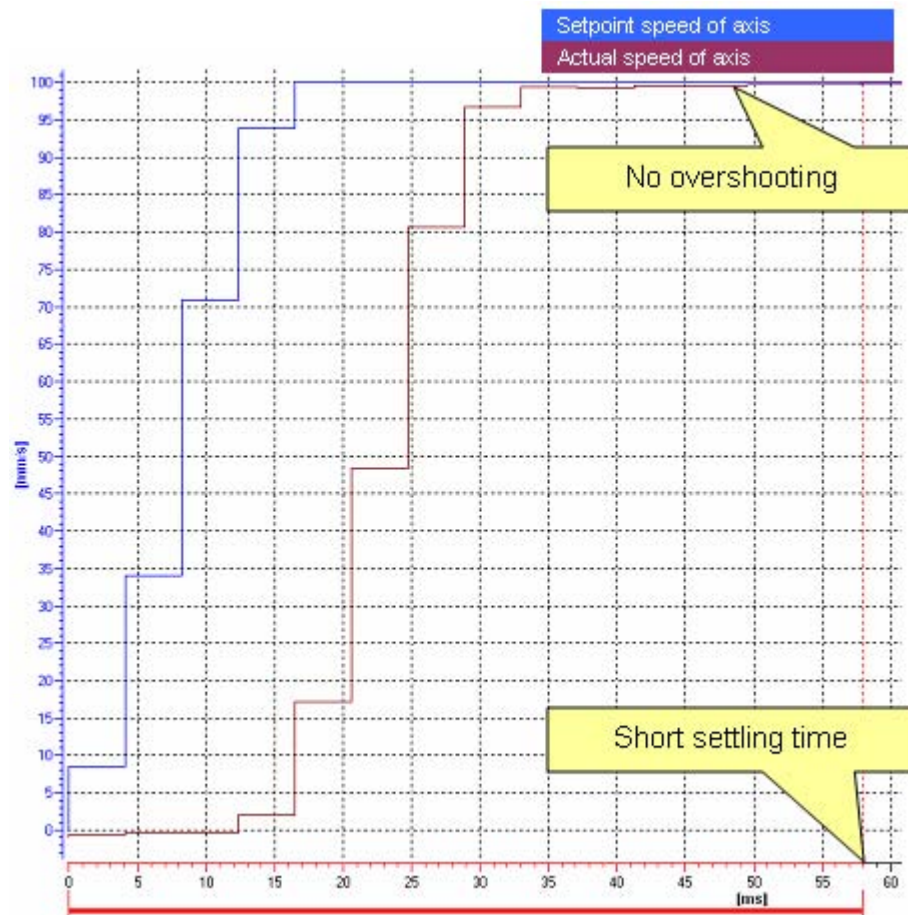
也可以在轴 (Axis) > 控制 (Control) 对话框的“静态控制器数据”(Static controller data) 选项卡中，设置 Kv 系数。

趋势曲线 — 增益系数 Kv 太低



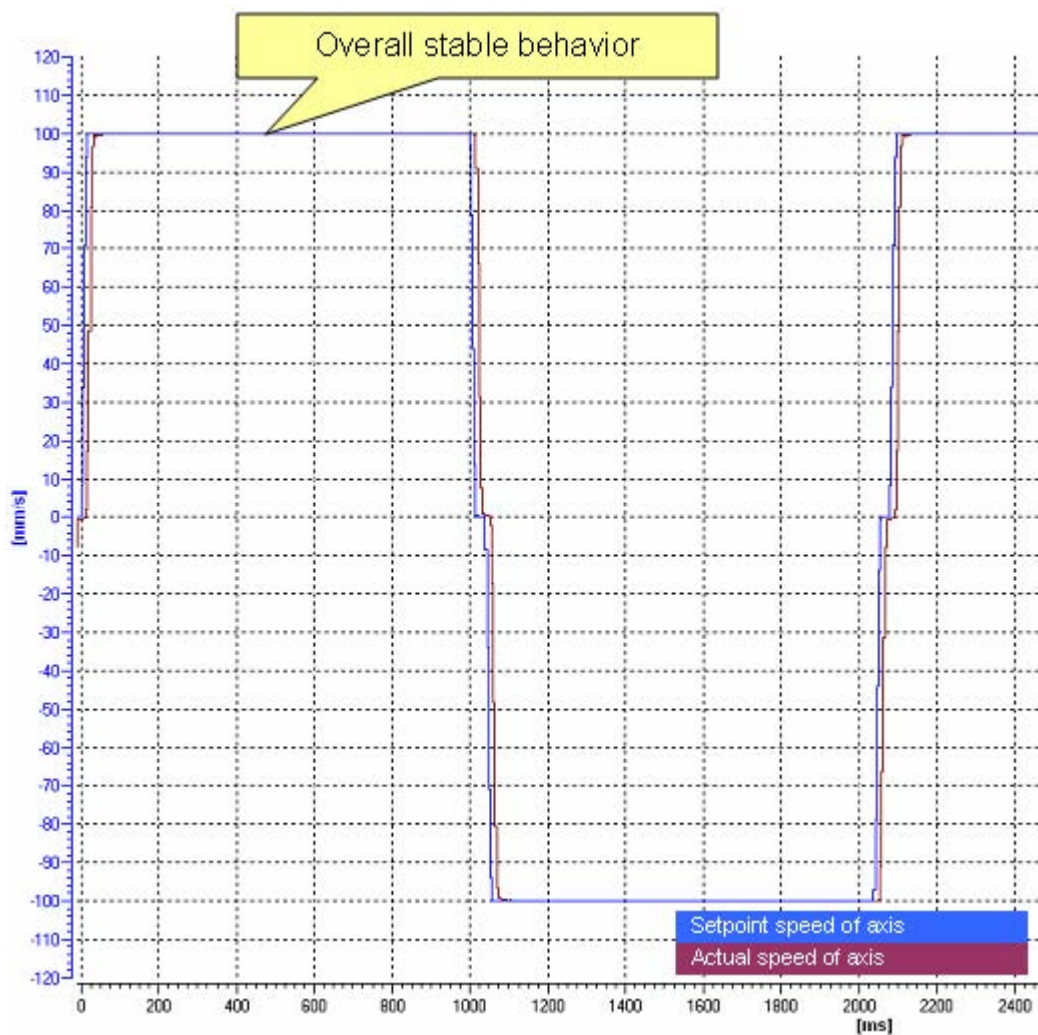
尽管该趋势曲线没有任何超调，但是存在很长的稳定时间。必须增大 Kv 系数以调节位置控制器。

趋势曲线 — 优化的增益系数 Kv (详细视图)



增大 K_v 系数时，请注意趋势曲线的以下属性：

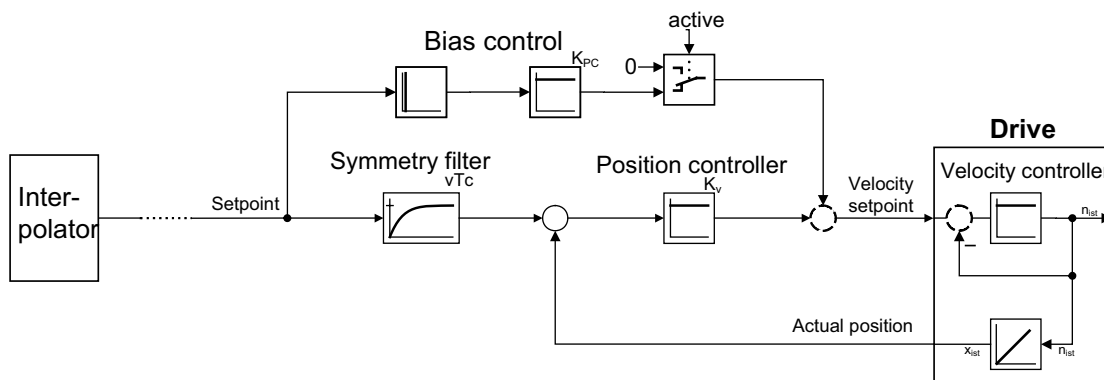
- 曲线显示出较短的稳定时间。
- 趋势没有显示任何反向的实际值。
- 激活设定值时不会出现超调。
- 趋势曲线显示整体上稳定的响应（趋势曲线中没有振荡）。



8.4.5 调节平衡滤波器

调节增益系数 K_v 之后，继续调节平衡滤波器的时间常数 vTc (velocity Time constant, 速度时间常数)。

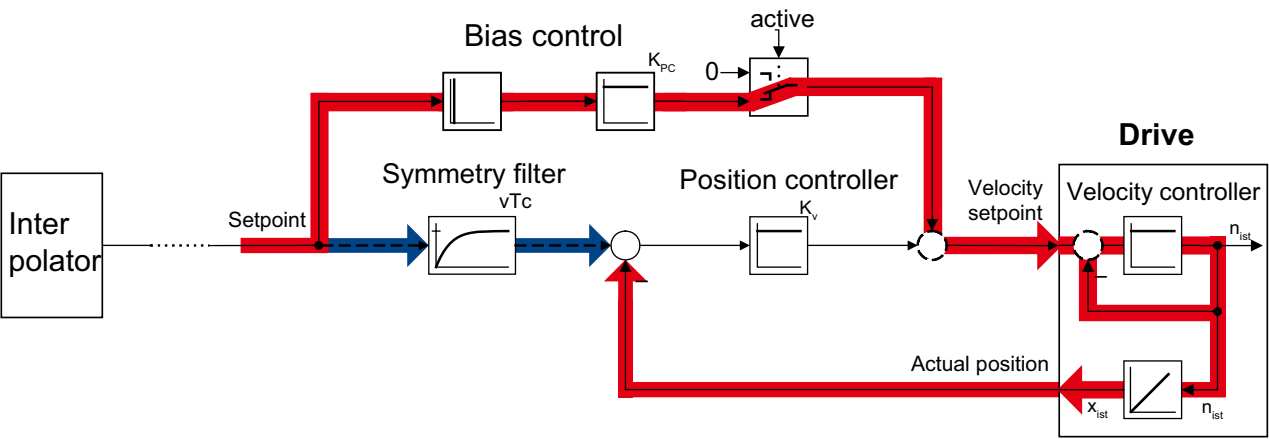
位置控制的方框图



设定值更改要求快速精确地稳定控制器。因此，T-CPU 使用预先控制和位置控制器来控制设定值偏差。平衡滤波器补偿预先控制功能的运行时间延迟。

- **预先控制**
用于对设定值更改进行高速响应的微分功能。预先控制功能可以立即稳定大部分的设定值更改。
- **位置控制器**
位置控制器因其比例作用而运行地较慢，但是明显具有更高的精度。其任务是精确调整预先控制功能尚未解决的剩余控制误差。在调节好的控制回路中的误差，跟设定值变化相比较而言是很小的。
- **平衡滤波器**
平衡滤波器补偿由预先控制回路引起的运行时间延迟。

平衡滤波器功能



预先控制功能对更改设定值有微分作用，这意味着生成的值将以相当大的速度传送到驱动器。驱动器中的速度控制器将通过预先控制功能校正部分设定值更改输出；驱动器将移动。实际值更改将反馈到位置控制器。

现在，位置控制器必须校正已定义的位置设定值和预先控制生成的实际值偏差之间的剩余误差。因此，位置控制器需要校正时刻的设定值。

平衡滤波器必须补偿由预先控制回路引起的运行时间延迟。这就需要将设定值延迟传送到位置控制器。

调节平衡滤波器

使用上述用户程序和 **TraceTool** 来调节平衡滤波器。与前面所做的更改不同，更改要记录的信号的设置：

记录	值
信号 1	<code>_project."axis name".servodata.controllerdifference</code>
信号 2	<code>_project."axis name"..servodata.followingerror</code>

必须激活平衡滤波器和预先控制才能调节平衡滤波器。

- **激活平衡滤波器**

将组态数据元素

`TypeOfAxis.NumberOfDataSets.DataSet_x.ControllerStruct.PV_Controller.balanceFilterMode` 的值更改为原始值 `Mode_1` 或 `Mode_2`。

此外，也可以通过工艺功能“MC_WriteParameter”的参数编号 `2035` 复位平衡滤波器。为 `Mode_1` 选择值 `90`，或者为 `Mode_2` 选择值 `91`。

- **启用速度预先控制**

在轴 **(Axis) > 控制 (Control)** 对话框的“静态控制器数据”(Static controller data) 上，激活“预先控制”(Pre-control) 复选框。

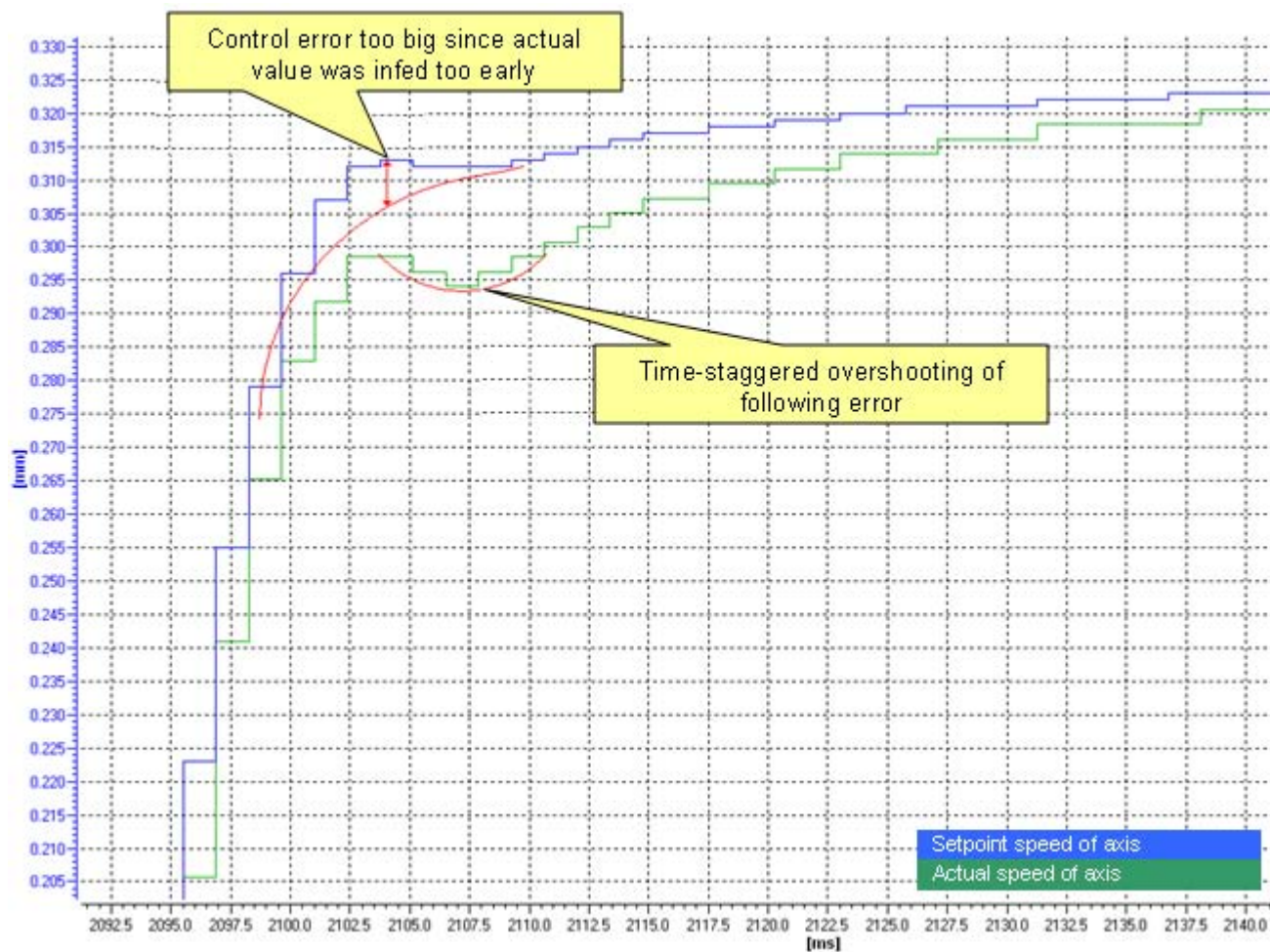
平衡滤波器调节的目标是校正预先控制需要的时间段。

- **设置时间常数 vTc**

在用户程序中通过工艺功能“MC_WriteParameter”的参数编号 `2063` 或在组态数据元素 `TypeOfAxis.NumberOfDataSets.DataSet_x.DynamicData.velocityTimeConstant` 中设置时间常数 vTc（velocity Time constant，速度时间常数）。

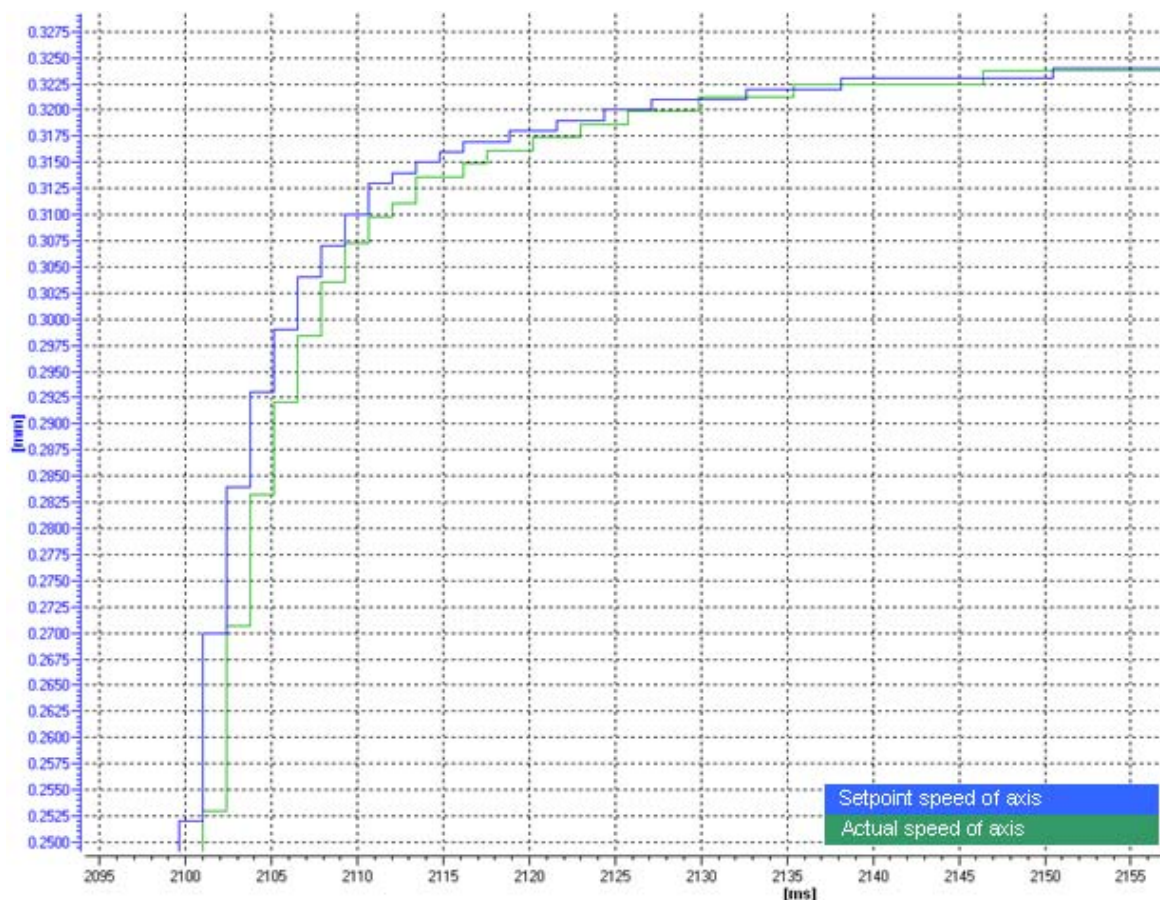
您也可以在轴 **(Axis) > 控制 (Control)** 对话框的“动态控制器数据”(Dynamic controller data) 选项卡中，设置时间常数 vTc（velocity Time constant，速度时间常数）。如果在该对话框中选中了“专家”(Expert) 复选框，该选项卡便可见。在该选项卡中，编辑参数“替换时间速度控制回路”(Substitution time speed control loop) 的值。

时间常数 vTc 设置不够长的示例



设定值输出地太早，因而不能进行位置控制。因为设定值导致了所返回的实际值，因此设定值与实际值的偏差太高。因此，控制变量中位置控制器输出的值太高；速度控制器将过度响应，并将增大以下误差。

时间常数 vTc 的理想设置的示例



调节平衡滤波器以优化性能。位置控制器将及时接收带有实际值的设定值。位置控制器只需校正预先控制尚未校正的微小偏差。

当您调节完平衡滤波器后，便成功完成了位置控制器的调节。

说明

注意事项：

- 如果在没有超调的情况下组态恒定速度阶段，则跟随误差不能采用零值。
- 因为位置控制周期包括在速度控制回路的等效时间常数 (vTc) 的计算中，该时间常数的最大值等于 **16** 个位置控制器时钟。死区时间元件的最大缓冲长度为 **16**。
- 完成『准备调节』一章描述的更改后，请勿忘记恢复原始值。
- 请确保还同时更改了离线项目中 **Kv** 系数的调节值和时间常数 vTc 的调节值

8.4.6 计算位置控制环的等效时间常量

位置控制回路的等效时间常数在以下情况下才有效：

- 通过在“MC_Power”工艺功能中设置 *StopMode = 0* 紧急断开
- 通过在“MC_MoveVelocity”工艺功能中设置 *PositionControl = FALSE*，从位置控制模式转换为速度控制模式。

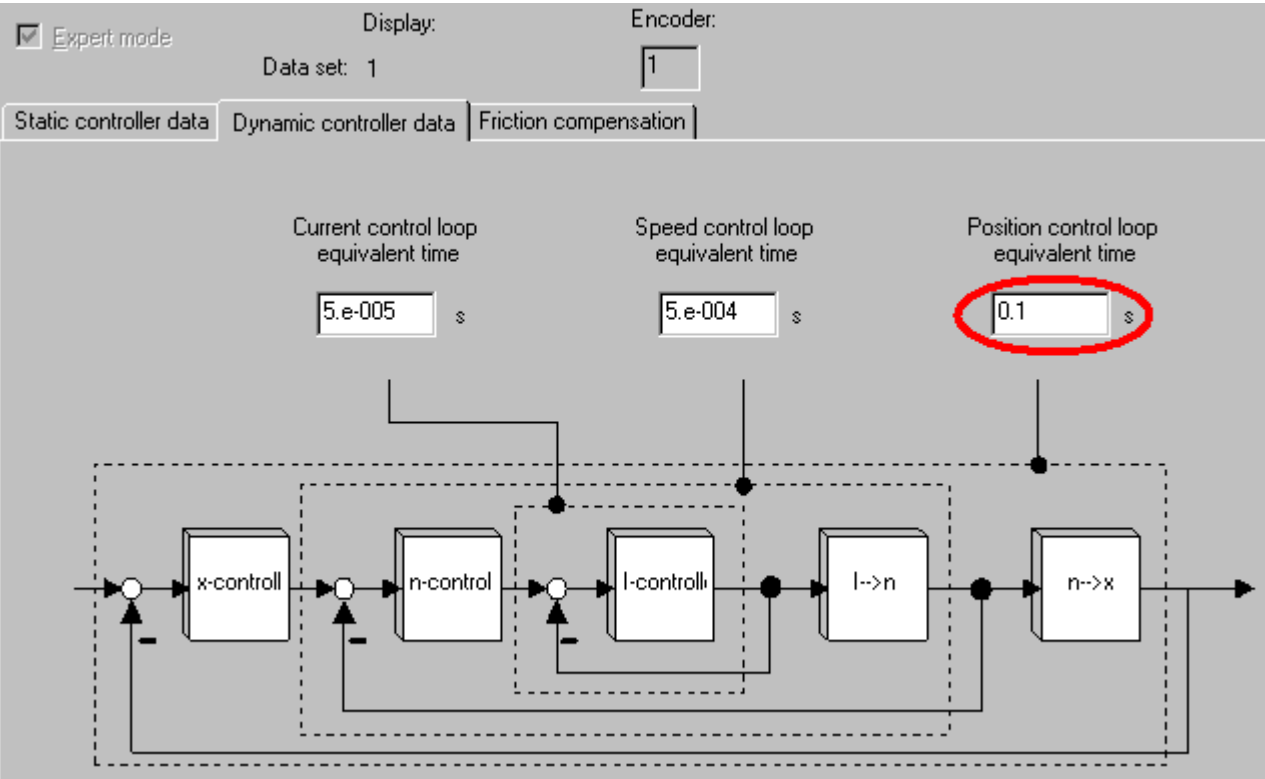
注意

如果位置控制回路的等效时间常数尚未调节充分，则轴在前面所述情况下可能会执行不必要的补偿运动。

要确定位置控制回路 pTc 的等效时间常数：

带预控制的比例作用控制器	vTc <> 0	pTc = vTc + Tservo/2
	vTc = 0	pTc = 0
不带预控制的比例作用控制器		pTc = 1/Kv

在这种情况下，速度控制回路 (vTc, speed control circuit) 所用的等效时间常数可以按照『调节平衡滤波器』一节中的描述确定。



在轴 (Axis) > 控制 (Control) 对话框的“动态控制器数据”(Dynamic controller data) 选项卡中，输入计算出的位置控制回路的等效时间常数。仅当设置了专家模式时，才可选择该选项卡。

也可以在 S7T Config 专家列表中的组态数据元素

TypeOfAxis.NumberOfDataSets.DataSet_x.DynamicData.velocityTimeConstant 中设置位置控制回路的等效时间。

说明

使用紧急断开命令验证计算出的位置控制回路的等效时间常数值。要执行紧急断开命令：

1. 通过设置输入参数 *StopMode* = 0 在工艺功能“MC_Power”上启用轴。
2. 通过设置输入参数 *PositionControl* = TRUE 移动具有工艺功能“MC_MoveVelocity”的轴。
3. 在“MC_Power”工艺功能的 *Enable* 输入参数中生成下降沿。

在 S7T Config 的 TraceTool 中，设置以下系统变量：

- *motionstatedata.commandvelocity* (轴速度设定值)
- *motionstatedata.actualvelocity* (实际轴速度)
- *positioningstate.commandposition* (轴位置设定值)
- *positioningstate.actualposition* (实际轴位置)
- *stopemergencycommand* (紧急关闭/停止命令的执行状态)

使用系统变量 *stopemergencycommand* 限制要分析的记录周期。

如果在已记录的系统变量中产生阶跃响应或超调，则更改位置控制回路的等效时间常数的值。

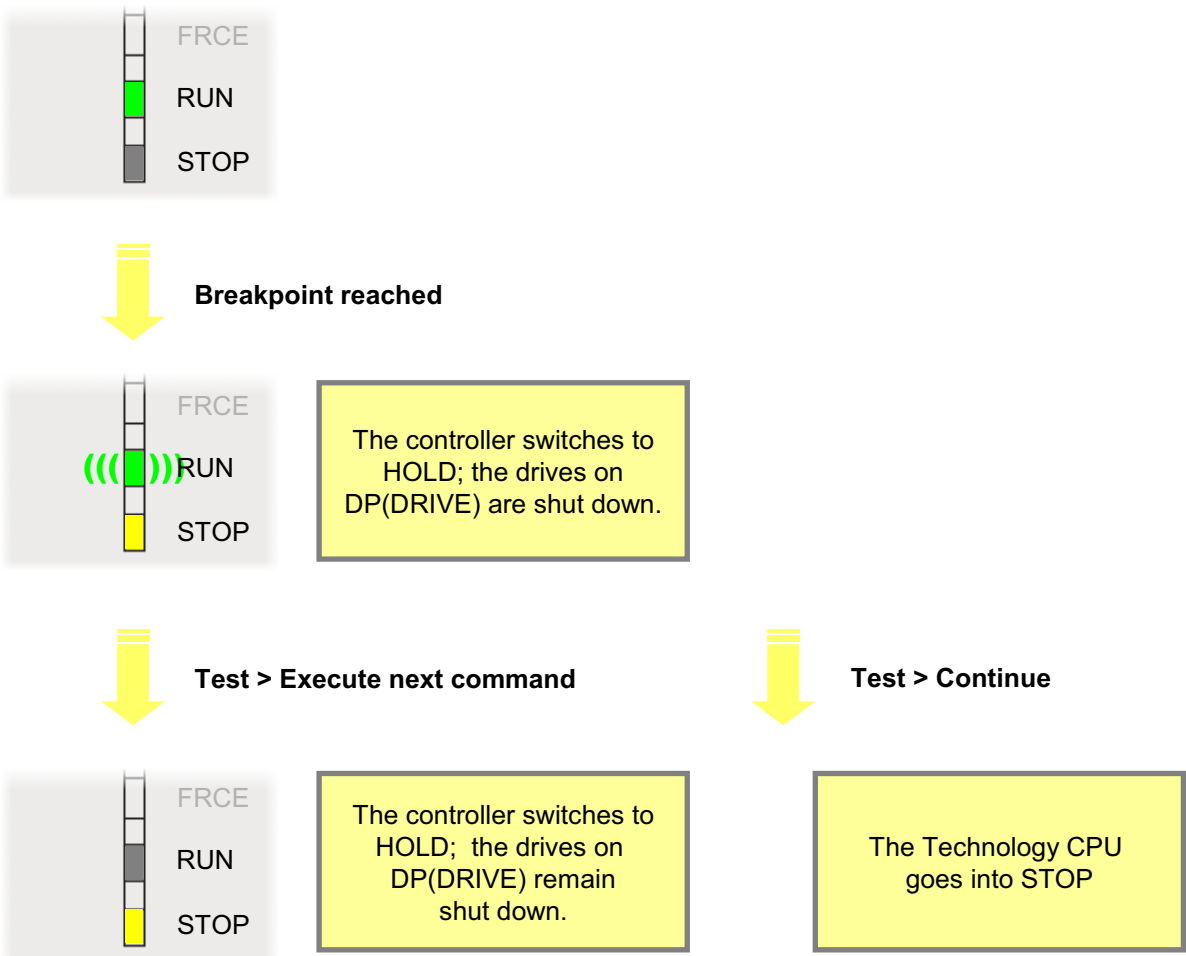
8.5 使用断点进行测试

8.5.1 通过断点测试

可在 STEP 7 的程序编辑器中通过断点测试用户程序。更多信息，请参阅 STEP 7 文档。

在下图所示的 T-CPU 中，由于安全考虑事项而有各种特性：

RUN state



已达到的断点

达到断点后，PLC 通常更改为“HOLD”；“RUN”LED 以 0.5 Hz 频率闪烁，“STOP”LED 则永久亮起。在“HOLD”模式下，PLC 及其集成工艺的输出将关闭。

菜单命令**测试 (Test) > 执行下一条语句 (Execute next statement)**

达到断点后，您可以通过选择**测试 (Test) > 执行下一条语句 (Execute next statement)** 命令继续测试您的用户程序。

测试 (Test) > 恢复运行 (Resume) 命令

当您执行**测试 (Test) > 恢复运行 (Resume)** 命令时，T-CPU 将更改为“STOP”。“重启”T-CPU 以恢复“RUN”模式。

8.6 使用监视表进行测试

8.6.1 使用监视表监视系统变量

在 S7T Config 中使用监视表监视或控制工艺对象的系统变量。S7T Config 中的监视表的功能堪比 STEP 7 的变量表。

您可以在监视表中对不同变量（多个工艺对象的系统变量、多个程序的用户变量）进行编组，并监视该组。

说明

在监视表中监视变量需要有在线连接。

如何创建监视表：

1. 打开浏览器，然后选择“MONITOR”元素。
2. 选择**插入 (Insert) > 监视表 (Watch table)** 命令。
3. 输入监视表的名称。
4. 单击“确定”(OK) 进行确认。

8.6 使用监视表进行测试

要向监视表添加变量：

- 1. 在浏览器中，选择包含要添加到监视表（变量表显示在符号浏览器中）的变量的元素。
- 2. 在符号浏览器中，选择包含要添加到监视表的变量的行。
- 3. 从右键快捷菜单中选择“移动到监视表”(Move to watch table)。
- 4. 根据需要重复步骤 1 到 3。

要在监视表中监视变量：

- 1. 建立到 T-CPU 的在线连接。
- 2. 打开监视表。
- 3. 监视“状态值”(Status value) 列中变量的当前值。

要在监视表中控制变量：

Watch_table_1:					
	Name	Plain text	IO address	Data type	Unit
1					

- 1. 建立到 T-CPU 的在线连接。
- 2. 在“控制值”(Control value) 列中输入要控制的值。 将选中单元格中的复选框。
- 3. 单击“立即控制”(Control immediately) ， 将要控制的值传送到 T-CPU。 如果您不想将该值传送到 T-CPU， 请取消选中该复选框。

8.7 将 TraceTool 用于测试

8.7.1 总览 — TraceTool

您可以使用 S7T Config 的 TraceTool 跟踪和保存信号图以及变量的值。例如，记录的数据可以用于分析机床的移动顺序以及在用户程序中进行故障诊断。为了获得较好的诊断结果，您可以使用功能发生器将定义的设定值分配给变量，同时记录其它变量的值。可以在图中显示和编辑记录的测量结果。

有关 TraceTool 用户界面和操作的信息，请参考 S7T Config 在线帮助。

8.7.2 跟踪 CPU 用户程序的值

工艺数据组件“Trace”是跟踪工具和应用程序之间的接口。可将 2 个数据类型为 DINT 的值、2 个数据类型为 DWORD 的值以及 4 个数据类型为 REAL 的值从应用程序中写入跟踪 DB 的各个变量中，以便进行跟踪。用户程序不能覆盖 *ErrorID* 变量。

集成工艺在工艺 DB 更新周期内读取 DB 数据。可以使用目标系统 (Target system) > 设置系统时钟 (Set system clocks) 命令在 S7T Config 中设置工艺 DB 更新周期。

不管 CPU 中的保持性设置如何，数据时钟都是非保持性的。

***S7_TraceDINT[0..1]* 变量**

S7_TraceDINT 变量包含具有两个数据类型为 DINT 的元素的 ARRAY。

当在 S7T Config 的 TraceTool 中的工艺 (*Technology*) > *userdata1* > *user1* 和 *user2* 中选择信号时，可以选择这些值来进行跟踪。

***S7_TraceDWORD[0..1]* 变量**

S7_TraceDWORD 变量包含具有两个数据类型为 DWORD 的元素的 ARRAY。如果您要使用 S7T 跟踪工具跟踪位信息（例如，状态字或错误字），则可以使用这些变量。您可以使用跟踪工具选择以常规 SIMATIC 编号表示的位。

当在 S7T Config 的 TraceTool 中的工艺 (*Technology*) > *userdata1* > *user3* 和 *user4* 中选择信号时，可以选择这些值来进行跟踪。

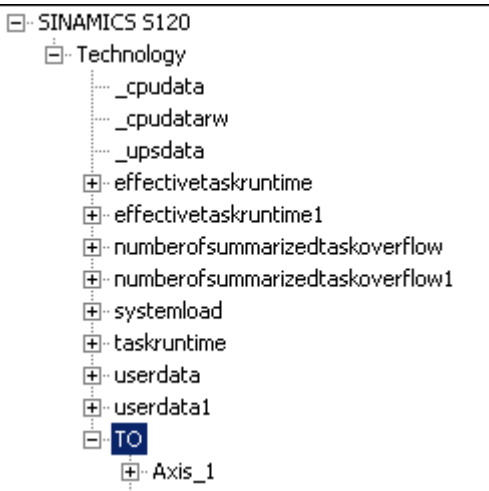
***S7_TraceREAL[0..3]* 变量**

S7_TraceREAL 变量包含具有四个数据类型为 REAL 的元素的 ARRAY。

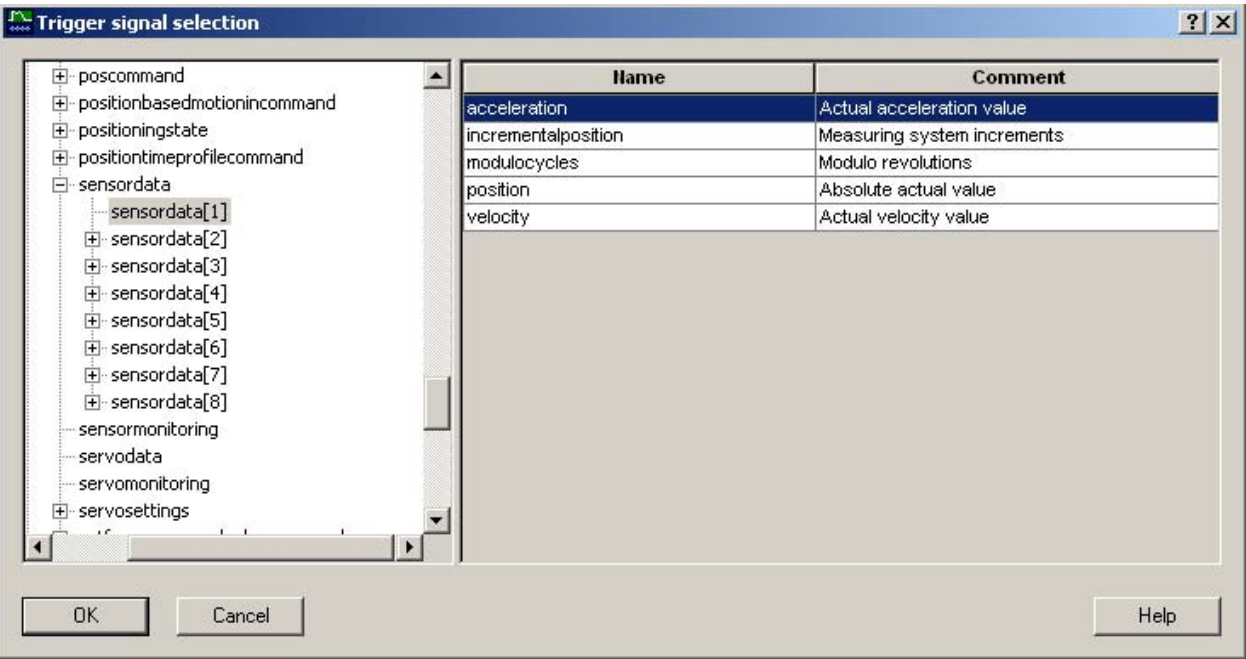
当在 S7T Config 的 TraceTool 中的工艺 (*Technology*) > *userdata1* > *user5*, *user6*, *user7* 和 *user8* 中选择信号时，可以选择这些值来进行跟踪。

8.7.3 跟踪控制器数据

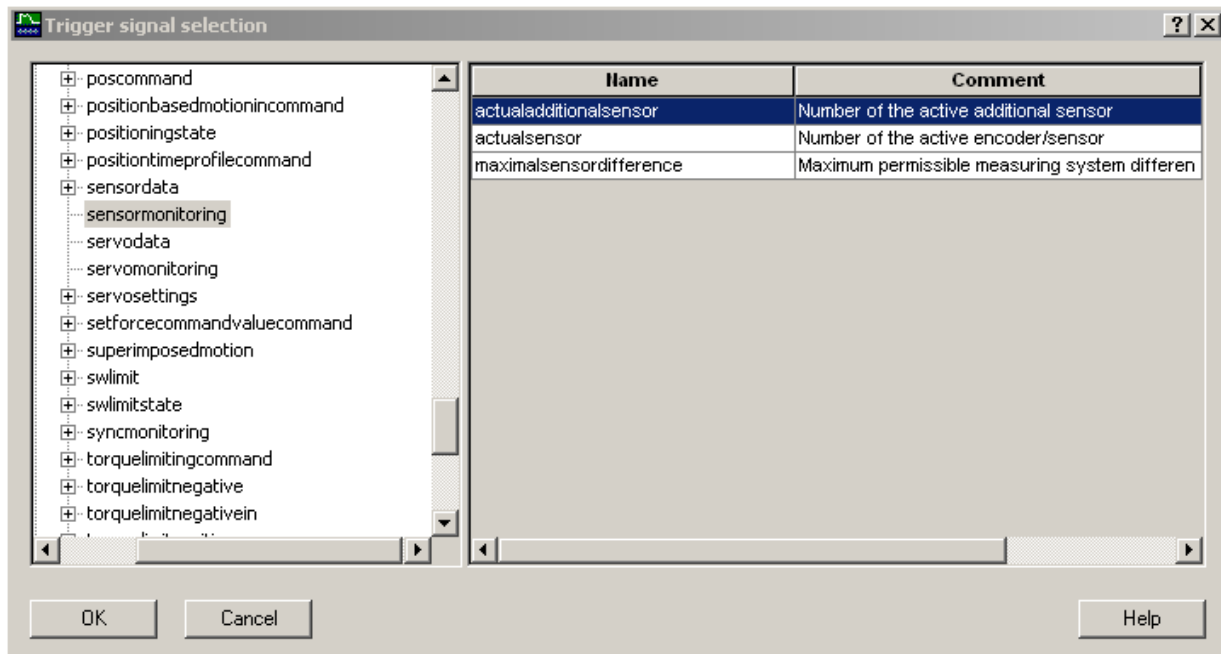
跟踪工具对于监视动态控制器数据非常有用。下面列出了一些有用的跟踪值，有助于您优化控制器。通过选择项目名称 (Project name) > 工艺 (Technology) > TO > 轴名称 (Axis name)，可以在“跟踪”的信号选择中查看跟踪值：



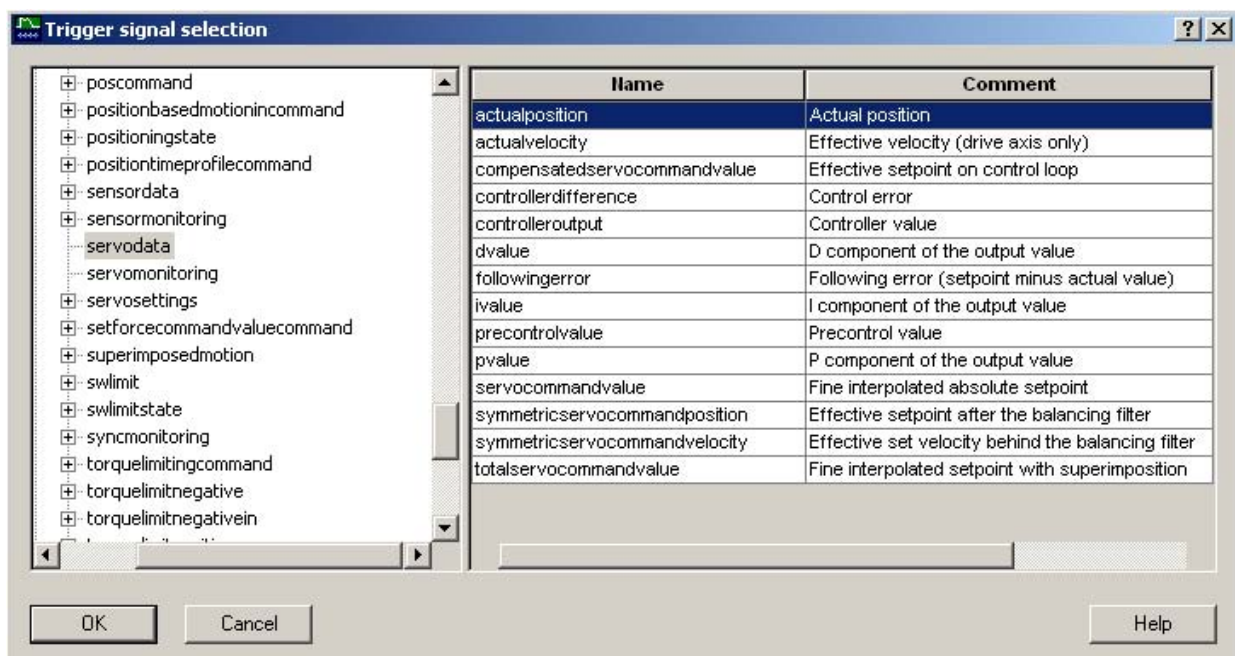
sensordata



sensormonitoring



servodata



8.7 将 TraceTool 用于测试

servomonitoring

+ sensordata + sensormonitoring + servodata + servomonitoring + servosettings + setforcecommandvaluecommand + superimposedmotion + swlimit	<table><tr><th>name</th><th>Comment</th></tr><tr><td>controllerdifferenceerror</td><td>Control error monitoring (drive axis only)</td></tr><tr><td>controlstate</td><td>Position control status</td></tr><tr><td>dynamicfollowingerror</td><td>Dynamic following error (error limit reached)</td></tr><tr><td>dynamicfollowingwarning</td><td>Dynamic following error (warning limit reached)</td></tr><tr><td>positioning</td><td>Positioning monitoring</td></tr><tr><td>stillstand</td><td>Standstill monitoring</td></tr></table>	name	Comment	controllerdifferenceerror	Control error monitoring (drive axis only)	controlstate	Position control status	dynamicfollowingerror	Dynamic following error (error limit reached)	dynamicfollowingwarning	Dynamic following error (warning limit reached)	positioning	Positioning monitoring	stillstand	Standstill monitoring
name	Comment														
controllerdifferenceerror	Control error monitoring (drive axis only)														
controlstate	Position control status														
dynamicfollowingerror	Dynamic following error (error limit reached)														
dynamicfollowingwarning	Dynamic following error (warning limit reached)														
positioning	Positioning monitoring														
stillstand	Standstill monitoring														

servosettings

Trigger signal selection

+

movingtoendstopcommand

+

override

+

pluslimitsofdynamics

+

poscommand

+

positionbasedmotionincommand

+

positioningstate

+

positiontimeprofilecommand

+

sensordata

+

sensormonitoring

+

servodata

+

servomonitoring

-

servosettings

additionalcommandvalue

additionalsetpointvalue

+

setforcecommandvaluecommand

+

superimposedmotion

+

swlimit

Name	Comment
additionalcommandvalue	
additionalfoutputvalue	Interface-specific superimposition of the limiting output value
additionalqoutputvalue	Interface-specific superimposition of the direct output value
additionalsetpointvalue	
setpointoffsetcompensation	Offset compensation

OK

Cancel

Help

8.8 使用控制面板进行测试

8.8.1 引言 — 控制面板

控制面板用于控制和监视各个轴和驱动器。 您可以使用它移动驱动器或轴。 这使调试工程师可以：

- 在启动程序驱动的协调轴运动之前，单独测试每个设备组件
- 仿真错误条件，以测试是否可以从控制面板完全操作各个轴，或者测试这部分是否已引发问题
- 将轴快速移动到另一个位置，而与程序“退回轴”无关
- 来回移动轴以实现优化目的（控制器优化）
- 执行主动回原点
- 在相对/绝对模式下定位
- 设置和复位轴使能信号
- 触发轴复位
- 通过添加绝对值编码器偏移量（“回原点”(Homing) 对话框）调整绝对值编码器

说明

当控制面板处于活动状态且 S7T Config 具有控制优先级时，您只能编辑立即生效的组态数据。

必须先返回控制优先级，您才能编辑任何要求重启的组态数据。重新启动后必须重新请求控制优先级

 **警告**

只能在调试阶段和出于测试目的而使用轴控制面板。

8.8 使用控制面板进行测试

8.8.2 启动轴控制面板

验证是否满足以下条件：

- 轴已组态。
- S7T Config 处于在线模式。
- 工艺已下载到目标系统。

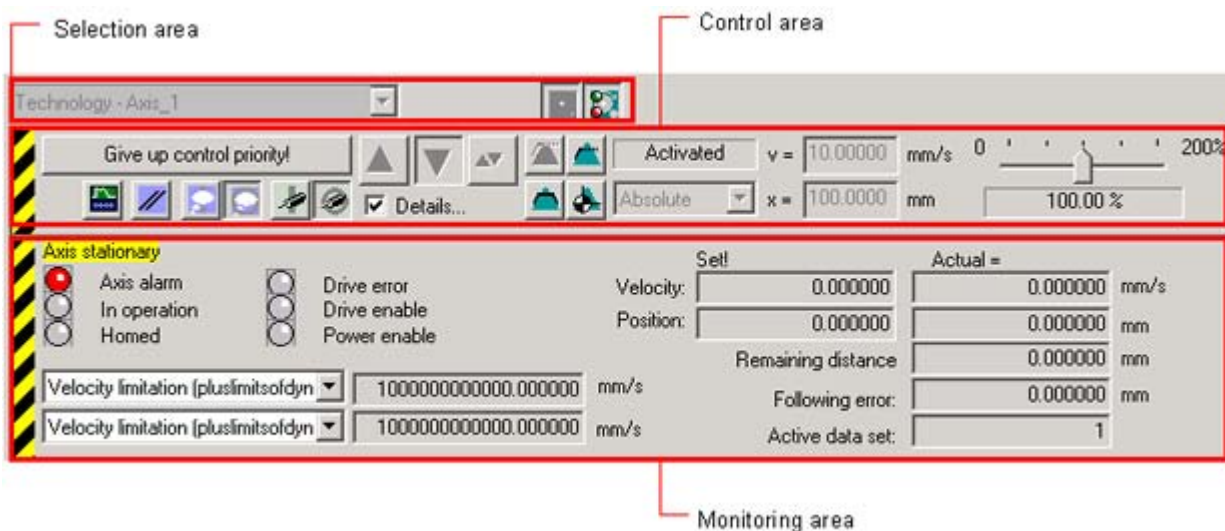
要启动轴控制面板：

1. 在浏览器中，打开“轴”(AXES) 文件夹。
2. 选择需要的轴。
3. 选择**编辑 (Edit) > 打开对象 (Open Object)**。
4. 选择**轴 (Axis) > 控制面板 (Control Panel)** 命令。

“控制面板”(Control panel) 选项卡打开，以详细视图显示。控制面板现在处于监视模式。

8.8.3 轴控制面板的布局

轴控制面板由若干区域组成：



轴控制面板（CP 处于控制模式）

- **选择区域**

您可以通过按下相应按钮来隐藏或显示其它区域。

可以在监视模式下选择轴。 可以选择该工艺的所有已组态的轴。

- **控制区域**（通过按 按钮显示和隐藏）。

在该区域中，您可以在选定轴上采用控制优先级，或将运动命令传送到轴。

在监视模式下，仅“跟踪编辑器”(Trace Editor) 按钮处于活动状态。

- **监视区域**（使用 按钮显示和隐藏）

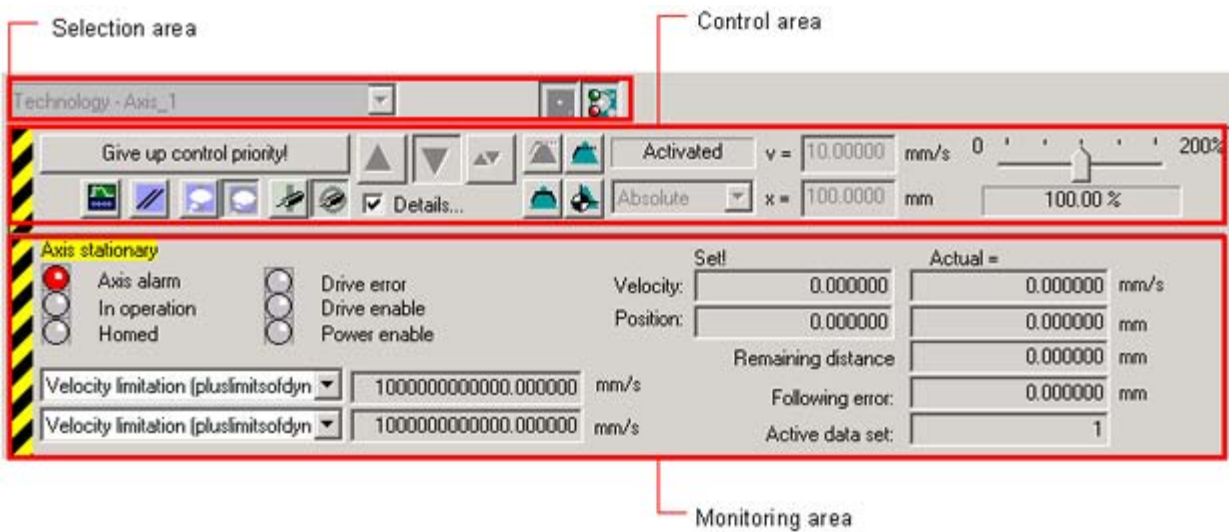
可以在该区域中监视选定轴的值：

- 在监视模式下：
例如，您可以测试在用户程序中启动的轴运动。
- 在控制模式下：
您可以测试由输出到轴的命令启动的运动。

8.8.4 使用控制面板监视轴值

在监视模式和控制模式下，您可以在该区域中监视选定轴的值：

- 在监视模式下（例如，在刚刚启动控制面板后）：
例如，您可以测试由程序启动的运动。
- 在控制模式下（采用控制优先级后）：
您可以测试由输出到轴的命令启动的运动。



控制面板的监视区域

该视图由选定轴（速度控制轴、定位轴或同步轴）的工艺确定：

- 速度设定值和实际值
- 位置设定值和实际值
- 剩余距离
- 跟随误差
- 可以从下拉列表框中选择的其它两个系统变量。

轴的运动状态（例如“轴处于停止状态”或“轴正在加速”）可以显示在带有彩色背景的文本框中。

有颜色的“LED”会通知您：

- 轴的报警和启用状态：
 - 活动的工艺报警（红色）
有关更多信息，请选择详细视图上的“报警”(Alarms) 选项卡。
 - 活动（绿色）
该“LED”指示跟随模式是否已取消。 仅能在此状态下执行轴命令。
 - 轴返回原点（绿色）。
- 驱动器的监视功能和使能信号：
 - 活动的驱动器错误（红色）
 - 驱动器启用（绿色）
 - 功率部件启用（脉冲使能 — 绿色）

8.8.5 采用控制优先级

检查是否满足下列条件：

- T-CPU 处于 STOP 模式。
- 选择 STEP 7 变量表，然后激活**变量 (Variable) > 启用外设输出 (Enable Peripheral Outputs)** 命令

 **警告**

必须严格遵守相关安全规则。
仅当启用了编程设备/CPU 状态监视并设置了恰当短的监视时间时，才能在控制模式下使用轴控制面板！
否则，当编程设备和 T-CPU 之间的通信出现问题时，轴可能执行不可控的运动。

按照下面概述的步骤进行操作：

1. 在控制区域中单击“采用控制优先级！”(Assume control priority !)。将输出一个警告消息框。
2. 在 S7T Config 中读取警告消息！
3. 请勿更改状态标记监视默认设置。
只能在例外情况下，且遵照所有危险警告进行更改。
4. 检查紧急停止设置。 根据需要进行更改。
 - 单击“扩展>>”(Extend>>)。
 - 出于安全考虑，应设置“当切换到其它 Windows 应用程序时紧急停止”(Emergency-Stop when changing to another Windows application) 复选框。 监视与该功能无关。
 - 设置可以触发紧急停止的键
 - <空格键>
 - <Esc>

CP 在采用控制优先级后将处于控制模式下。 控制区域中原来灰显的按钮和字段现在都处于活动状态。

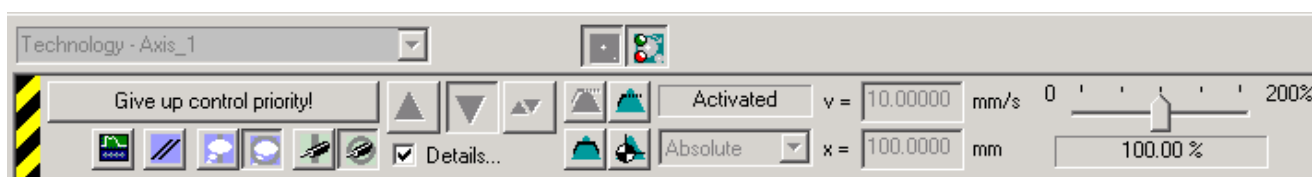
说明
当控制面板处于控制模式下时，空格信号将被解释为停止轴的控制信号。 这也会影响编程设备上运行的其它应用程序（例如文字处理器）。 如果为紧急停止功能分配空格键，该键在文字处理器中将不再具有输入空格字符的功能。

8.8.6 控制轴

检查是否满足下列条件：

- 轴 CP 处于控制模式下。
- CP 已采用控制优先级。

控制区域中的按钮处于活动状态。



轴 CP 的控制区域

轴 CP 的控制区域中的控制元素：

命令	控制元素	说明
设置/复位使能信号		该命令用于切换轴上的使能信号。 状态 • 驱动器已启用 • 脉冲使能（功率启用）已发出 在监视区域中的相应“LED”中指示。 为了可以在轴上处理运动命令，必须发出所有使能并取消后续操作。 对于速度控制轴，忽略位置控制启用。
详细信息	复选框	如果选中了该复选框，某些命令会打开一个斜坡独的参数输入框。 如果取消选中该复选框，您可以在其旁边的输入域中输入选定的参数。
速度控制的轴运动		该命令指定轴以速度控制形式启动。 预设速度设定值，轴将沿着组态的斜坡加速到该设定值。该命令可用于所有轴类型。定位轴和同步轴是速度控制的。 通过单击“启动”(Start) 或“点动”(Jog) 启动运动。

8.8 使用控制面板进行测试

命令	控制元素	说明
位置控制轴运动		该命令指定轴以位置控制形式启动。 预设速度设定值，轴将沿着组态的斜坡加速到该设定值。该命令可用于定位轴和跟随轴。 通过单击“启动”(Start) 或“点动”(Jog) 启动运动。
相对/绝对定位		该命令可以定位轴（定位轴或跟随轴）。位置指定可以是绝对的，也可以是相对的。也可以沿“最短路径”定位模数轴。 通过单击“启动”(Start) 启动运动。
轴回原点		使用绝对测量系统时，仅在调试期间需要执行一次回原点。完成此操作后，在打开机器时，位置值已知。 如果是增量测量系统，机器必须在每次打开时都回原点。 通过单击“启动”(Start) 启动运动。
-	输入框	输入框用于在取消选中“详细信息”(Details) 复选框时输入位置和速度设定值
速度倍率	滚动条控件	预设速度乘以设置的倍率系数（0% 到 200%）。 出于安全考虑，在启动控制面板时将该值设置为 100。 该值影响所有后续命令。
仿真打开		该命令用于设置轴仿真模式。设定值输出将受到抑制。
仿真关闭		该命令用于将轴从仿真模式返回正常操作状态。在仿真模式下，上一个有效的设定值将被激活以输出。 注意： 由于在仿真模式下禁用了驱动器，因此当您退出仿真模式时，驱动器可能执行调整运动。 在您仍处于仿真模式下时，可以通过将轴移动到你初始位置，来避免出现这些调整运动。
复位轴		轴复位到初始状态。 由于参数化不正确而导致轴处于错误定义的状态时，应使用该命令。
启动运动		将启动组态的上一个运动命令（速度预设 [speed preset]、使用位置控制启动轴 [start axis with position control]、定位轴 [position axis] 或将轴返回原点 [home axis]）。
停止运动		最后一次启动的运动将停止。

命令	控制元素	说明
点动		“速度预设”(Speed preset) 和“以位置控制形式启动轴运动”(position-controlled start of the axis motion) 命令也可以在点动模式下执行。只要您在该按钮上按住鼠标右键，运动便会继续。
执行跟踪记录		启动 TraceTool。

注意

采取所有安全措施后，再将轴投入运行。

要使用轴 CP 移动轴：

1. 单击“详细信息”(Details)。

2. 单击“启用轴”(Enable axis)。

在显示的窗口中，检查是否置位了所有使能信号，以及是否禁用了跟踪模式。单击“确定”(OK) 进行确认。

3. 单击运动命令按钮（“速度预设”(Speed preset)、**“以位置控制形式启动轴”(Start axis positioncontrolled)**、“定位轴”(Position axis) 或**“将轴返回原点”(Home axis)**）。

这将打开下一个对话框。在该对话框中，输入您的参数。单击“确定”(OK) 进行确认。

4. 在“速度倍率”(Speed Override) 移位寄存器中设置适当的倍率值。

5. 单击“开始”(Start)。

这将启动运动。

当运动处于活动状态时，您可以修改速度倍率系数。

6. 您可以通过单击“停止”(Stop) 或按空格键来停止运动。

可以通过单击“启动”(Start) 恢复运动。

7. 根据需要重复步骤 3 到 6。

说明

当使用“速度预设”(Speed preset) 和“以位置控制形式启动轴”(Start axis position controlled) 命令时，您还可以通过单击“点动”(Jog) 来启动运动。只要需要，按住鼠标左键便可运动，距离不限。当释放鼠标左键时，运动会停止。

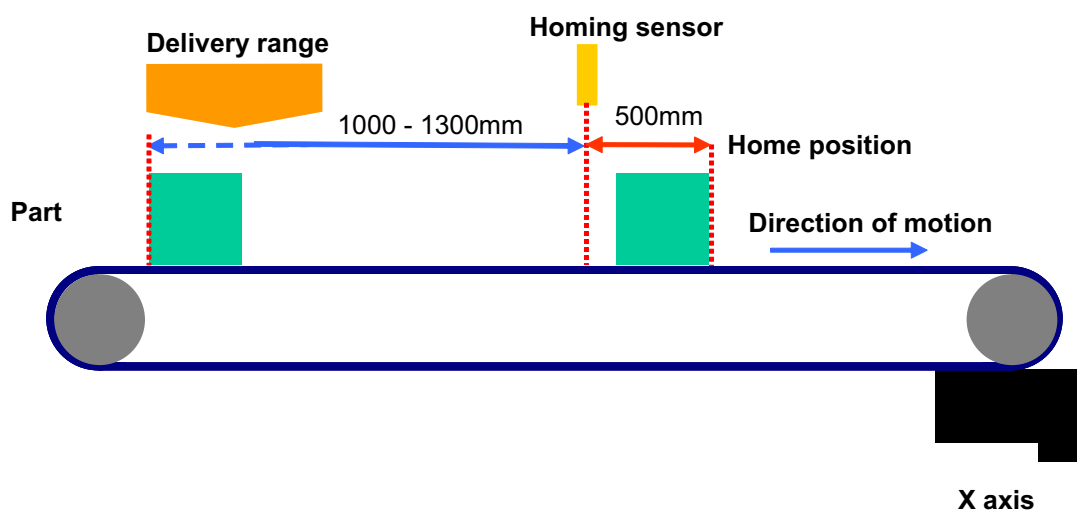
您还可以通过按空格键来停止轴。

8.8 使用控制面板进行测试

附录

9.1 应用实例

9.1.1 应用实例“使用目标传感器定位”



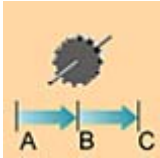
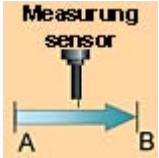
该示例显示了在传送带上执行定位任务的典型应用。

工件被移动到定义的结束位置。卸货区域是可变的，因此在开始运动时工件的当前位置未知。

在结束位置下游 500 mm 处连接一个目标传感器，使您可以精确定位该工件。目标传感器连接到驱动器的数字输入。触发传感器时，会在传送带上将实际位置值设置为 500 mm。将保存部件位置和测量输入数据，且由于 T-CPU 的集成工艺而没有出现可察觉的延迟。实际位置值已知，并且可以高度精确地完成定位。

工艺对象

在 S7T Config 中，组态可用于控制传送带驱动器和目标传感器的工艺对象。该示例中使用的工艺对象：

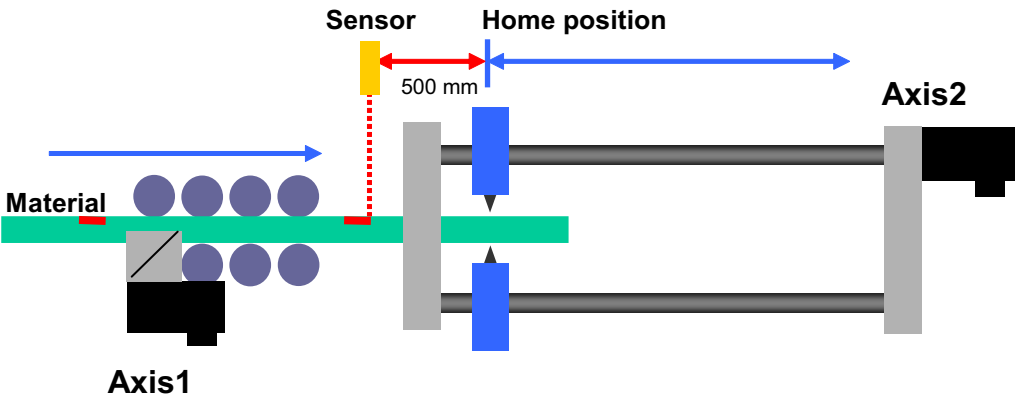
工艺对象： • 定位轴（传送带） • 测量输入（目标传感器）		

工艺功能

通过调用下面所列的用户程序中的功能块，来执行解决该任务所需的工艺功能：

	工艺功能	任务/说明
1.	MC_Power	启用轴
2.	MC_MeasuringInput	设置参数并使能测量输入（目标传感器）
3.	MC_MoveVelocity	使用速度预设移动轴
4.	MC_Home	检测到目标传感器后校正位置（实际位置值 — 目标传感器位置）。因此，检测到目标传感器时，将位置值复位为零。
5.	MC_MoveAbsolute	将轴绝对定位到 500 mm 处。

9.1.2 应用实例“飞剪”



该示例显示了将连续生产的材料剪切为定长的典型应用。

进入材料由进给轴（轴 1）连续进给，并由剪切轴（轴 2）“实时”剪为定长。

材料上的压痕指示了剪切位置。传感器记录压痕的位置，并将进给轴（轴 1）的实际值设置为 -500 mm。系统在检测到传感器时启动剪切轴的同步操作。系统在执行此同步同步操作时，剪切机刀片会接近压痕。剪切轴达到同步之后，立即剪切材料。剪切轴在完成剪切操作之后将返回其原点位置，并等待下一个压痕。

工艺对象

在 S7T Config 中组态工艺对象。本例中使用的工艺对象：

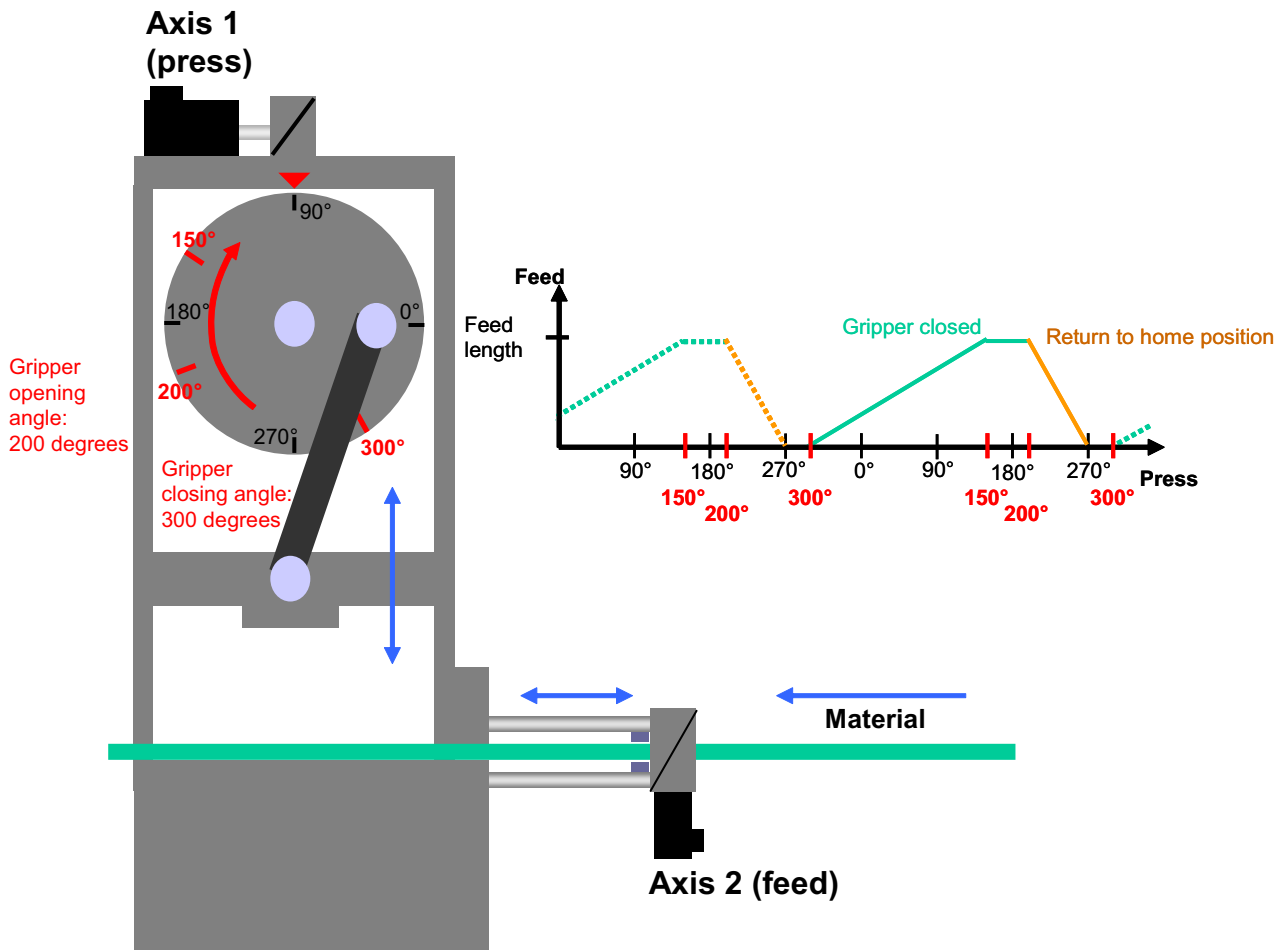
工艺对象： • 定位轴（进给轴） • 同步轴（剪切轴） • 测量输入（传感器）			

工艺功能

通过调用下面所列的用户程序中的功能块，来执行解决该任务所需的工艺功能：

	工艺功能	任务/说明
1.	MC_Power	启用进给轴
2.	MC_Power	启用剪切轴
3.	MC_Home	剪切轴在基本位置处返回原点，使其位置值为 0 mm
4.	MC_MeasuringInput	设置参数并启用传感器
5.	MC_MoveVelocity	以预设速度移动进给轴
6.	MC_Home	当传感器检测到压痕时，将在进给轴上将位置值设置为 -500 mm。"模式 4 中的"MC_Home"（位置倍率）： $X_{act} = X_{act} - \text{"传感器位置"} - 500 \text{ mm}$
7.	MC_GearIn	当传感器检测到压痕时，将启动剪切轴的同步操作
8.	MC_GearIn	返回"InGear"值时剪切材料（达到齿轮同步）
9.	MC_MoveAbsolute	剪切轴将返回其原点位置
		按步骤 5 到 10 作进一步循环处理

9.1.3 应用实例“夹具进给”



该示例显示了通过夹具进给机制进给材料所使用的压力。

夹具将在 300° 角位置处关闭，然后在角位置 150° 处将材料推到压力位置。

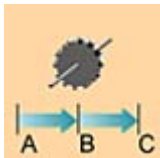
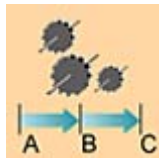
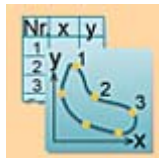
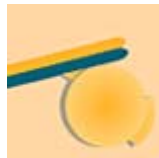
夹具在施压操作过程中保持关闭状态，到达 200° 处时再打开。

夹具进给从角位置 200° 返回其 270° 处的原点位置。

进给轴的运动与死轴的旋转移动直接成比例。因此，进给轴通过凸轮盘工艺连接到压力轴。夹具释放和关闭运动通过凸轮开关功能（在 300° 处关闭，在 200° 处释放）控制。

工艺对象

在 S7T Config 中组态工艺对象。本例中使用的工艺对象：

工艺对象 • 定位轴（压力轴） • 同步轴（进给轴） • 凸轮盘（连接进给轴和压力轴） • 输出凸轮（夹具控制）				

工艺功能

通过调用下面所列的用户程序中的功能块，来执行解决该任务所需的工艺功能：

	工艺功能	任务/说明
1.	MC_Power	启用进给轴
2.	MC_Power	启用压力轴
3.	MC_Home	压力轴在角位置 0 ° 处回原点
4.	MC_Home	进给轴在 原点位置返回原点
5.	MC_CamIn	进给轴和压力轴通过凸轮盘连接在一起
6.	MC_CamSwitch	用于控制夹具箝位的、基于位置的凸轮的设置
7.	MC_MoveVelocity	以预设速度移动压力轴
		活动的生产过程
		关闭压力：
a.	MC_MoveAbsolute	将压力轴移动到角位置 0 ° 处
b.	MC_Halt	“正常停止”进给轴
c.	MC_Halt	“正常停止”压力轴

9.2 问题、提示和技巧

9.2.1 列表 — 问题、提示和技巧

如何利用新性能特性 (页 944)

如何将 T-CPU 替换为其它类型 (页 946)

如何识别固件版本 (页 948)

如何升级 CPU 31xT-2 DP 的固件 (页 949)

如何升级集成工艺的固件 (页 950)

如何转换工艺 (页 952)

如何转换工艺以及如何升级工艺包 (页 953)

如何重新创建工艺 DB (页 957)

移植后要将哪些内容下载到 PLC (页 960)

如何使用 S7-Technology 分析早期版本中的项目? (页 961)

如何使用 S7-Technology 修改早期版本中的项目? (页 963)

使用“另存为”并进行重组时,要遵循哪些规定 (页 965)

为什么为特定 SDB 的在线和离线数据分配了不同的时间戳? (页 965)

STEP 7 报告“存储空间不足”(Insufficient memory space) (页 966)

超时后, CPU 偶尔会跳转到 STOP (页 966)

对工艺数据执行“保存并编译全部”(Save and compile all) 功能时出错 (页 967)

Internet 上的附加信息

9.2.2 如何利用新性能特性？

问题：

如何利用“S7-Technology 中有哪些新增内容”中介绍的 S7-Technology 的性能特性？

解答：

要利用 S7-Technology 的新性能特性，必须安装下图中所列的固件版本。通过检查集成工艺和 CPU 的固件版本来启动。

	S7-Technology						
	V1.0	V2.0	V2.0 + SP1	V3.0	V3.0 + SP1	V3.0 + SP2	V4.1
CPU 31xT-2 DP							
FW-CPU	V2.1	V2.3	V2.3	V2.4	V2.4	V2.4	V2.6
FW-Technology	V3.0	V3.1	V3.1	V3.2	V3.2	V3.2	V4.1.1
S7T Config	V3.0	V3.1	V3.2	V3.2	V3.2	V4.0	V4.1
Microbox T							
WinLC T					V4.2	V4.2	V4.2
FW-Technology					V3.2	V3.2	V3.2
S7T Config					V3.2	V4.0	V4.1

按如下所述执行升级步骤。

升级为新版本的 S7-Technology

下述升级适用于所有 S7-Technology V3.x、V4.x 等版本的升级。

1. 升级 CPU 固件 (页 949)
2. 升级工艺固件 (页 950)
3. 转换工艺并升级工艺包 (页 953)
4. 重新创建工艺 DB (页 957)
5. 更新工艺功能 (页 959)
6. 下载到 PLC (页 960)

说明

升级可能导致项目数据的存储器要求增加。如果在当前 MMC 卡上只保留了较少的存储器储备，则随时准备一个较大的 Micro Memory Card (MMC)。

升级为 S7-Technology 的服务包

下面描述了从 S7-Technology 的一个版本升级到其相应的服务包（例如从 V3.0 升级到 V3.0 + SP2）。

1. 转换工艺 (页 952)
2. 更新工艺功能 (页 959)
3. 下载到 PLC (页 960)

说明

运行升级之前创建项目数据的备份副本。

执行升级时，请勿打开任何其它对话框窗口（例如专家列表）。

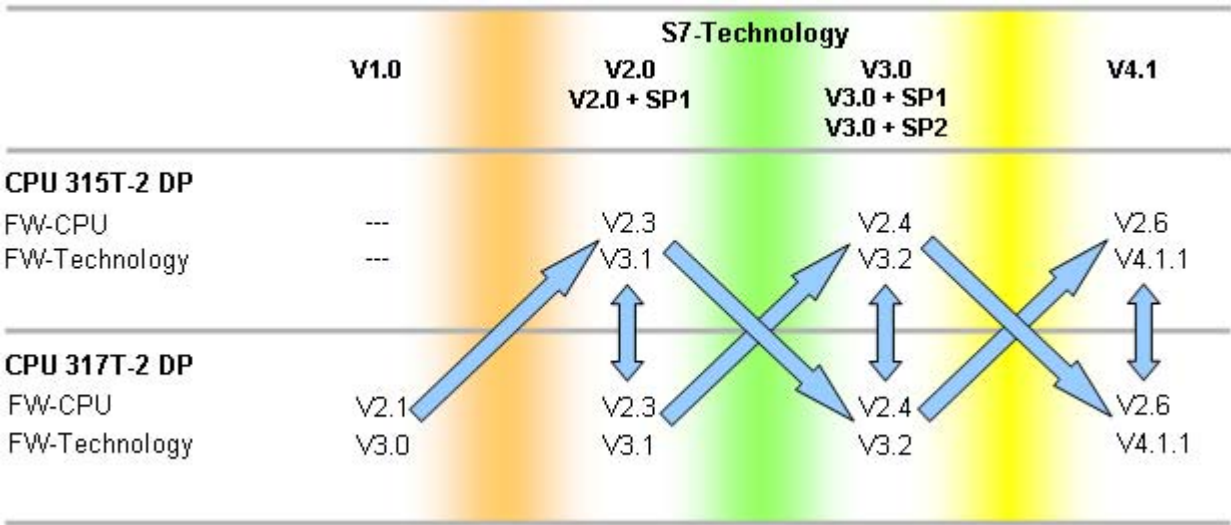
9.2.3 如何将 T-CPU 替换为其它类型？

问题：

可以将 T-CPU 替换为其它类型并重复使用其组态数据吗？

解答：

下面列出的 T-CPU 支持互换和重复使用它们的组态数据：



如果您替换 T-CPU，则可能会忽略某些版本。

说明

如果您使用文件 (File) > 另存为 (Save as) 命令将当前项目保存在 SIMATIC Manager 中，并选中“通过重新组织（慢速）”(With reorganization [slow]) 复选框，请在替换 T-CPU 之前执行以下步骤。

1. 启动 S7T Config。
2. 按照所显示的信息和指令进行操作。然后关闭 S7T Config。

使用 CPU 和相同版本的工艺固件替换 T-CPU

1. 在 HW Config 中替换 T-CPU（按照『如何升级集成工艺的固件』一章的描述）
2. 选择站 (Station) > 保存 (Save)
3. 关闭 HW Config。
4. 在 SIMATIC Manager 中，双击“工艺对象”(Technological objects)。通过选择“选项”(Options) > “组态工艺”(Configure technology) 命令打开 S7T Config。
5. 选择“项目”(Project) > “保存并编译全部”(Save and compile all) 菜单命令。
6. 关闭 S7T Config。
7. 将项目数据下载到 PLC（按照『升级后要将哪些内容下载到 PLC』一章的描述）。

使用 CPU 和不同固件版本的工艺替换 T-CPU

要替换 T-CPU，请像升级一样进行操作。有关详细信息，请参考『如何利用新功能特性』一章。

说明

当您替换 T-CPU 时，请考虑任何不同的数据量。

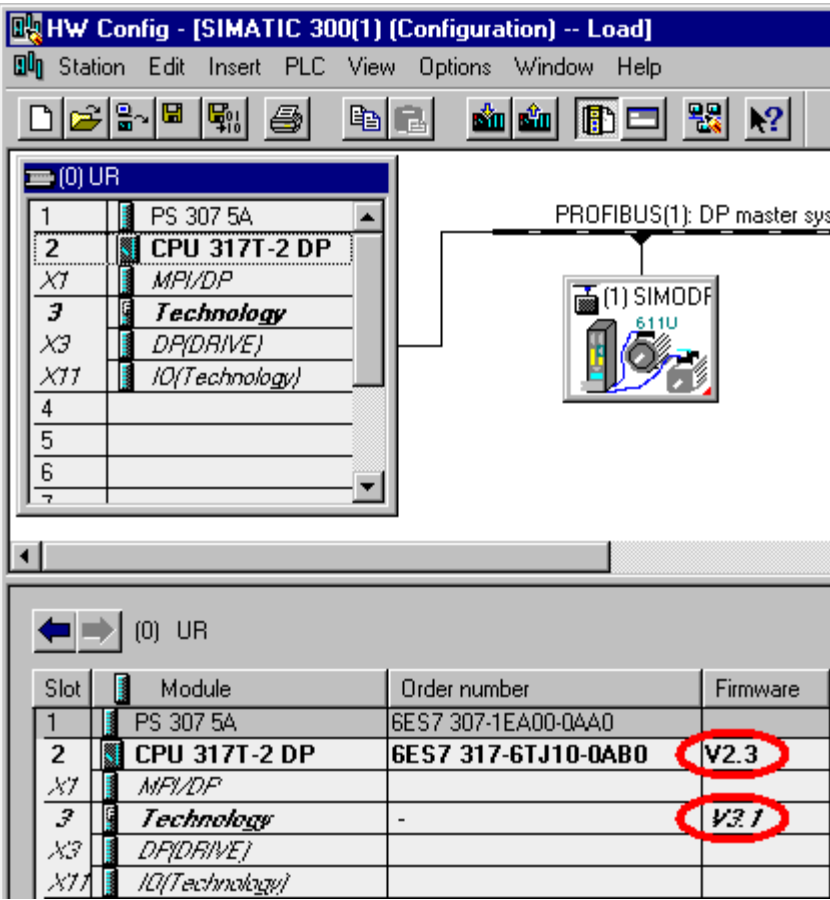
9.2.4 如何识别固件版本？

问题：

该文档多次涉及不同的固件版本。 如何识别系统中使用的 T-CPU 或工艺的固件版本？

解答：

打开 HW Config，然后标记顶部的站窗口（在 317T-2 DP CPU 的示例中，表示如下）。
底部的站窗口会显示 CPU 和工艺的固件版本。



说明

如果您对离线和在线数据的一致性不确定，请在在线模式下打开 HW Config。

9.2.5 如何升级 CPU 31xT-2 DP 的固件？

问题：

S7-Technology 中提供的新性能特性可能需要升级 CPU 31xT-2 DP 的固件。如何升级 CPU 31xT-2 DP 的固件？

解答：

升级 CPU 固件之前请检查相关的固件版本：

- 按照『识别固件版本』一章的描述识别 CPU 31xT-2 DP 的固件版本。
- 升级所需的固件版本已在『如何利用新性能特性？』一章中指定。

从 MMC 卡中执行固件升级。所需组件：

- 带有 STEP 7 Basis 和 MMC 卡适配器的编程设备/PC。
- 容量至少为 8 MB 的 MMC 卡。

可以从我们的 Internet“服务和支持”页面上下载当前的固件更新：

<http://www.siemens.com/automation/service&support>

在 SIMATIC Manager 中，选择 **PLC > 更新操作系统 (Update Operating System)** 命令。
有关操作步骤的详细信息，请参考 STEP 7 在线帮助。

从 CPU 31xT 的 Bootblock 版本 A030705 起，MMC 卡不在线也可以进行固件更新（通过网络）。可在 CPU 31xT 的诊断缓冲区中识别 Bootblock 版本。

说明

请注意 CPU 31xT-2 DP 的硬件版本中的差异。

例如，硬件版本在集成工艺的存储器负载方面有所不同。有关更多信息，请参考相关的《CPU 数据》手册，第 6.6 章『CPU 集成工艺中的存储器』。

9.2.6 如何升级集成工艺的固件？

问题：

您可能需要升级集成工艺的固件，以便可以使用 **S7-Technology** 的新性能特性。如何升级集成工艺的固件？

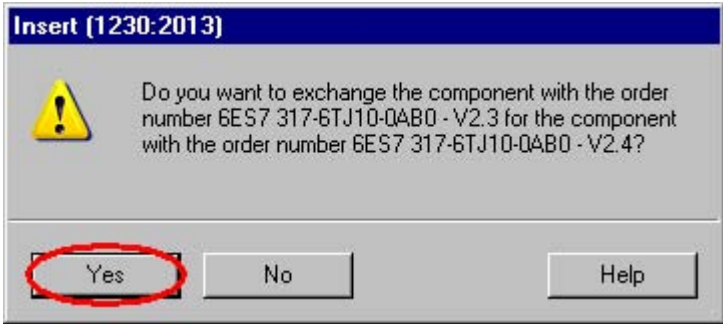
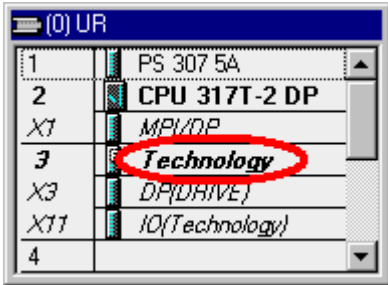
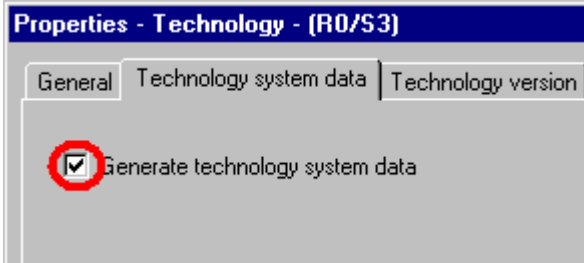
解答：

升级集成工艺的固件之前，请检查相关的固件版本：

- 可以按照『如何识别固件版本？』一章的描述识别集成工艺的固件版本。
- 升级所需的固件版本已在『如何利用新性能特性？』一章中指定。

按照下列步骤升级固件： 下图显示了 **CPU 317T-2 DP** 的升级过程。根据所用的 T-CPU，HW Config 中的表示可能会略有不同：

1.	打开 HW Config。
2.	从硬件目录中选择“SIMATIC T-CPU”配置文件。
3.	打开包含 T-CPU 或固件版本的文件夹，如下图所示。将所需的组件拖放到 CPU 上的插槽中。

4.	将出现一个对话框。单击“是”(Yes) 确认组件替换。  A dialog box titled "Insert (1230:2013)" with a yellow warning icon. The text asks: "Do you want to exchange the component with the order number 6ES7 317-6TJ10-QAB0 - V2.3 for the component with the order number 6ES7 317-6TJ10-QAB0 - V2.4?". At the bottom, there are three buttons: "Yes" (circled in red), "No", and "Help".
5.	双击 T-CPU 的“工艺”(Technology) 条目。  A screenshot of a configuration table for a T-CPU. The table has two columns: an index column and a component name column. The rows are: 1 PS 307 5A, 2 CPU 317T-2 DP, 3 Technology (circled in red), 4 DP(DRIVE), 5 IO(Technology). The table is titled "[0] UR".
6.	将打开“属性 — 工艺”(Properties - Technology) 对话框。 在“工艺系统数据”(Technology system data) 选项卡上，选中“生成工艺系统数据”(Generate technology system data) 复选框。  A screenshot of the "Properties - Technology - (R0/S3)" dialog box. It has three tabs: "General", "Technology system data", and "Technology version". The "Technology system data" tab is selected. Under this tab, there is a checkbox labeled "Generate technology system data" which is checked and circled in red.
7.	单击“确定”(OK) 关闭“属性 — 工艺”(Properties - Technology) 对话框。
8.	选择站 (Station) > 保存 (Save)
9.	关闭 HW Config。

9.2.7 如何转换工艺？

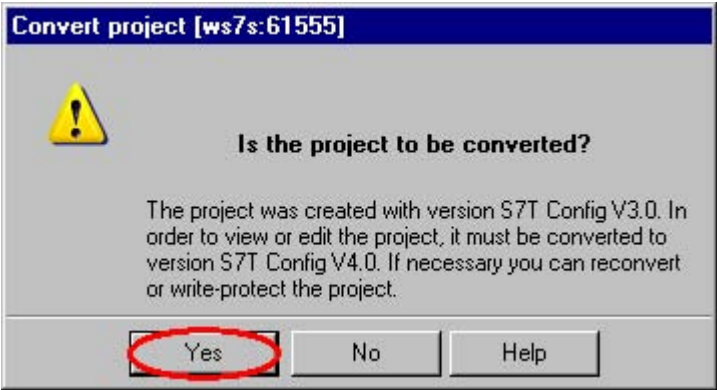
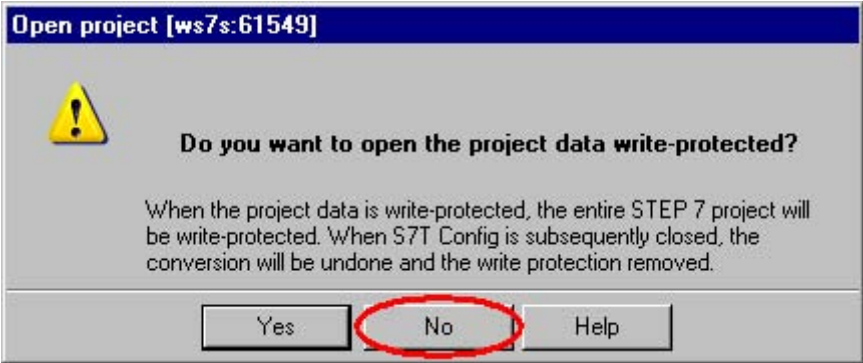
问题：

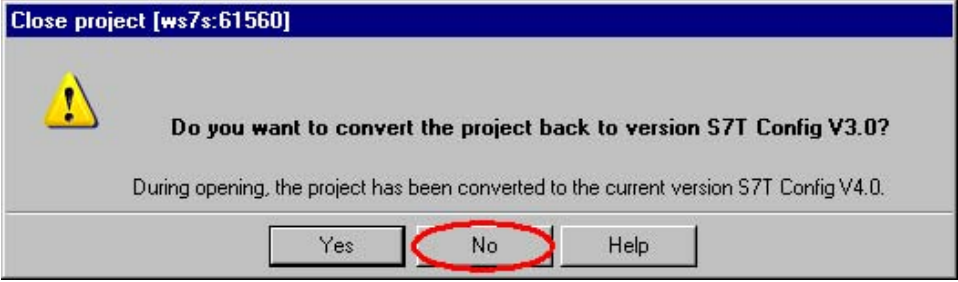
新版本的 S7-Technology 可能需要转换工艺。 如何转换工艺？

解答：

可能需要转换工艺的组态已在『如何利用新性能特性』一章中指定。

如下所述，进行工艺转换：
(根据转换，图释中的文本内容和版本名称可能发生变化)

1.	在 SIMATIC Manager 中，选择“工艺对象”(Technological objects)，然后选择“选项” (Options) > “组态工艺” (Configure technology)
2.	S7T Config 将打开，并出现一个对话框。  单击“是”(Yes) 确认该对话框。 现在项目已转换。
3.	下一个对话框将打开：  单击“否”(No) 确认此问题。

4.	选择项目 (Project) > 保存并编译全部 (Save and compile all)
5.	关闭 S7T Config。
6.	关闭 S7T Config 时会出现一个对话框。  单击“否”(No) 进行确认。

9.2.8 如何转换工艺以及如何升级工艺包？

问题：

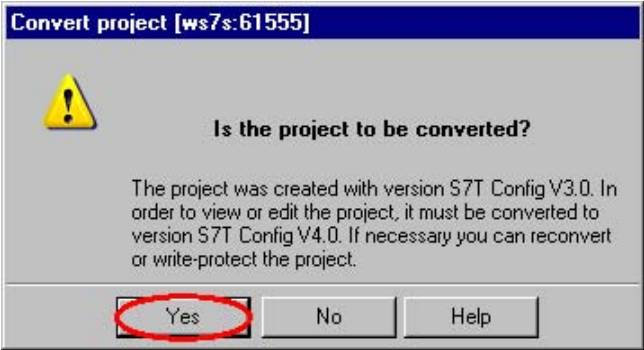
当您应用新版本的 S7-Technology 时，可能需要转换工艺并升级工艺包。如何执行？

解答：

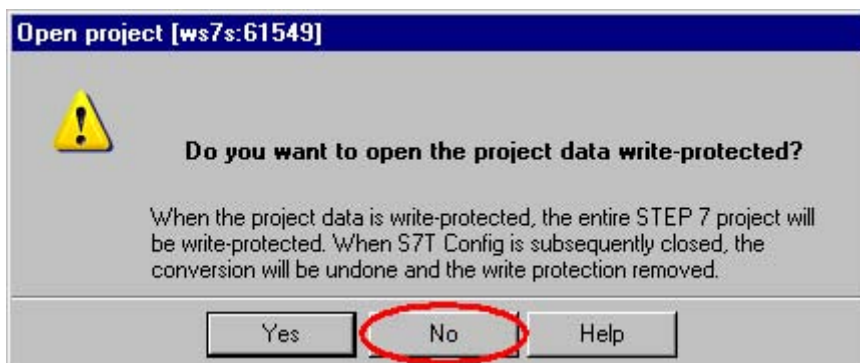
『如何利用新性能特性』一章中指定了可能需要这些任务的组态。

步骤：

（根据转换，图解中的文本内容和版本名称可能发生变化）

1.	在 SIMATIC Manager 中选择“工艺对象”(Technological objects)，然后选择“选项”(Options) > “组态工艺”(Configure technology)。
2.	S7T Config 将打开，并出现一个对话框。  单击“是”(Yes) 确认该对话框。现在项目已转换。

3. 下一个对话框将打开：

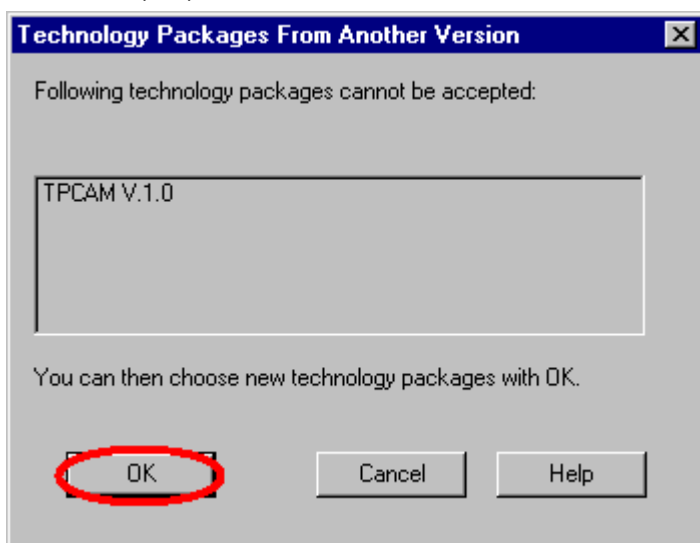


单击“否”(No) 确认该问题。

将在下一步中自动保存并编译该项目。由于升级操作未完成，可能会生成某些警告和错误消息。请忽略这些警告和错误，并单击“确定”(OK) 确认消息框。

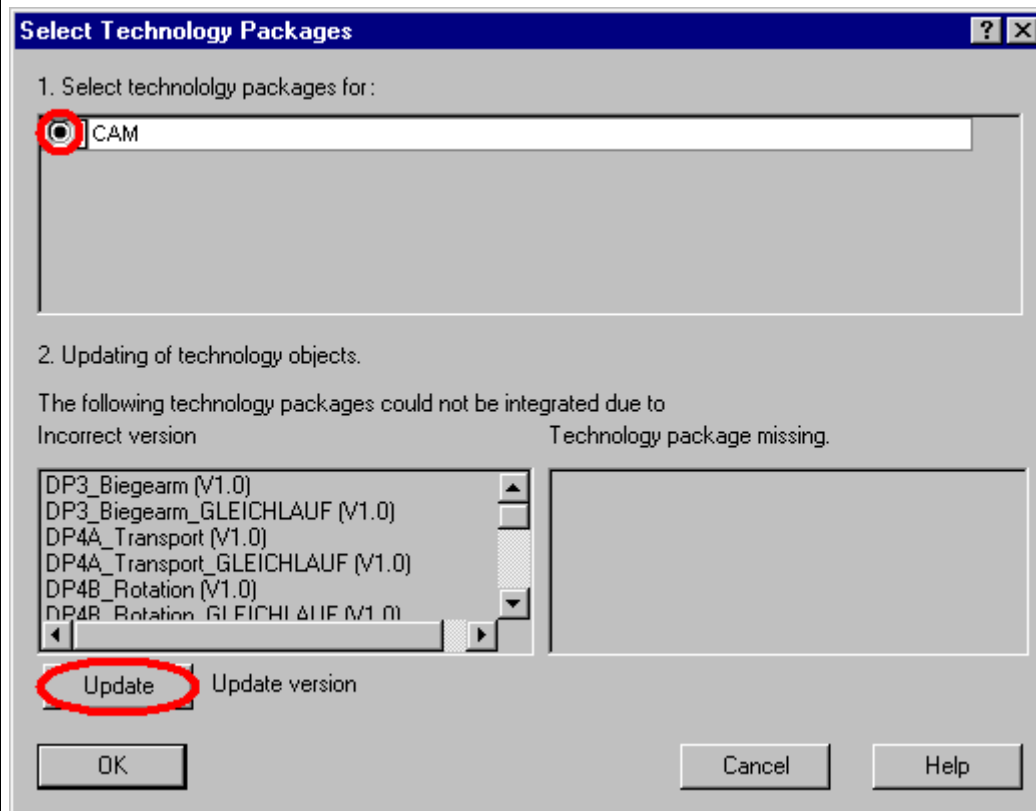
4. 从浏览器中选择工艺，然后选择**编辑 (Edit) > 选择工艺包 (Select technology packages)** 命令。

5. 单击“确定”(OK) 确认此消息框。

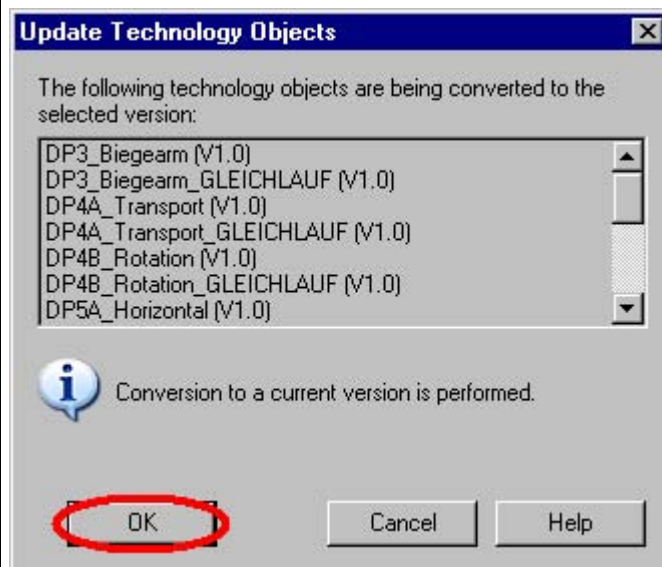


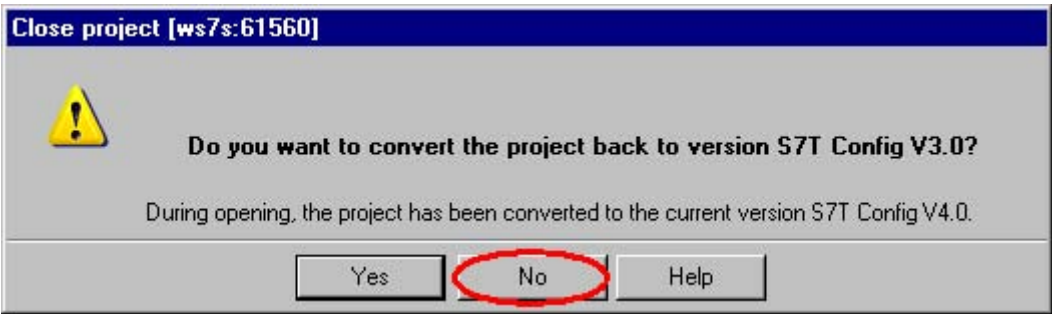
“选择工艺包”(Select technology packages) 对话框随即打开。

6. 选择“CAM”工艺包，并通过单击“更新”(Update) 进行确认。



7. 单击“确定”(OK) 确认此消息框。



8.	单击“是”(Yes) 确认下一条消息。系统将更新工艺包。
	转换较旧项目的说明 此转换将更改某些系统变量的单位。因此，S7T Config 的详细视图在转换期间输出警告，指示无法设置某些系统变量（例如，<i>userdefaultqfaxis.maxderivative.foutput</i>）。 由于这些系统变量不在 T-CPU 中使用，因此可以忽略警告。
9.	单击“确定”(OK) 确认“更新工艺对象完成”(Update of technology objects completed) 消息。
10.	单击“确定”(OK) 关闭“选择工艺包”(Select technology packages) 对话框。
11.	选择项目 (Project) > 保存并编译全部 (Save and compile all)
11a.	如果未出现任何错误消息，请继续步骤 12。 如果出现任何错误消息，请继续执行步骤 11b。
11b.	选择项目 (Project) > 检查一致性 (Check consistency) 命令找到错误。 转换较旧项目的说明 出现错误消息的可能原因： - 在轴和外部编码器上，组态数据元素 <i>TypeOfAxis.NumberOfEncoders.Encoder1.DriverInfo.EncoderNumberOnDevice</i> 的值不为 0。 例外情况：外部编码器返回轴的其它实际值。先决条件是使用标准消息帧 4 或 6，或使用消息帧 103、104 或 106。 补救措施：将组态数据元素设置为 0。 - 在没有编码器的位置控制轴上，组态数据元素 <i>TypeOfAxis.NumberOfDataSets.DataSet_1.EncoderNumber.EncoderNumber</i> 的值不为 0。 补救措施：将组态数据元素设置为 0。
11c.	消除错误原因并继续执行步骤 11。 如果想在以后消除错误原因，也可以通过选择项目 (Project) > 保存 (Save) 命令保存项目，并继续执行步骤 12。
12.	关闭 S7T Config。
13.	关闭 S7T Config 时会出现一个对话框。  单击“否”(No) 进行确认。

9.2.9 如何重新创建工艺 DB?

问题:

应用新版本的 S7-Technology 之后，可能需要重新创建工艺 DB。如何重新创建工艺 DB?

解答:

可能需要重新创建工艺 DB 的组态已在『如何利用新性能特性』一章中定义。

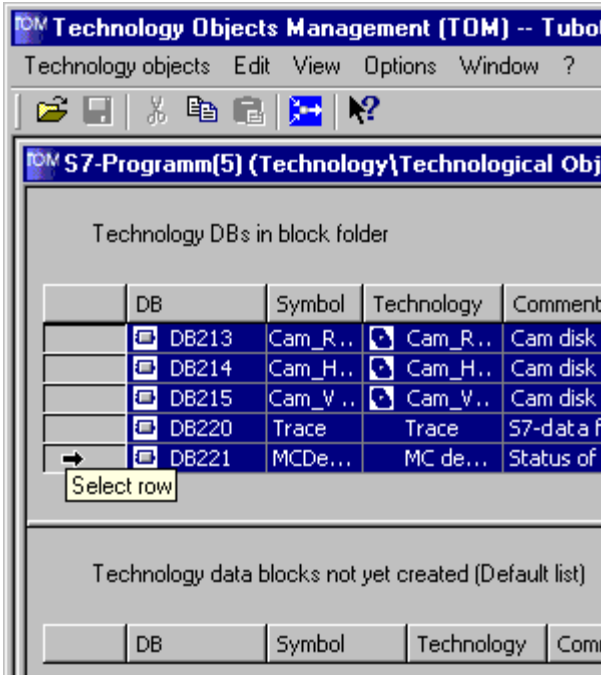
按照下列步骤操作，重新创建工艺 DB:

1.

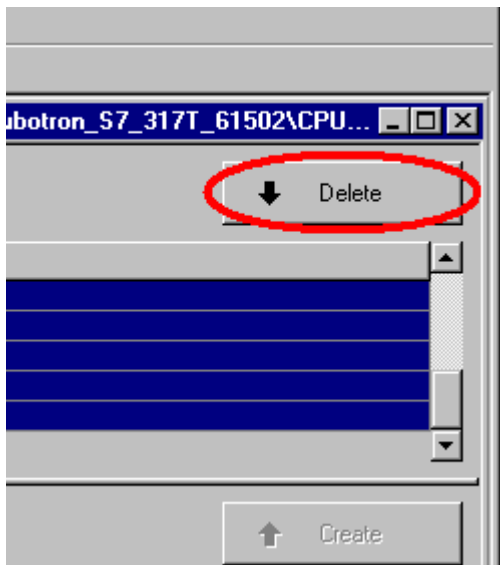
运行“工艺对象管理”(Technology Objects Management)。

2.

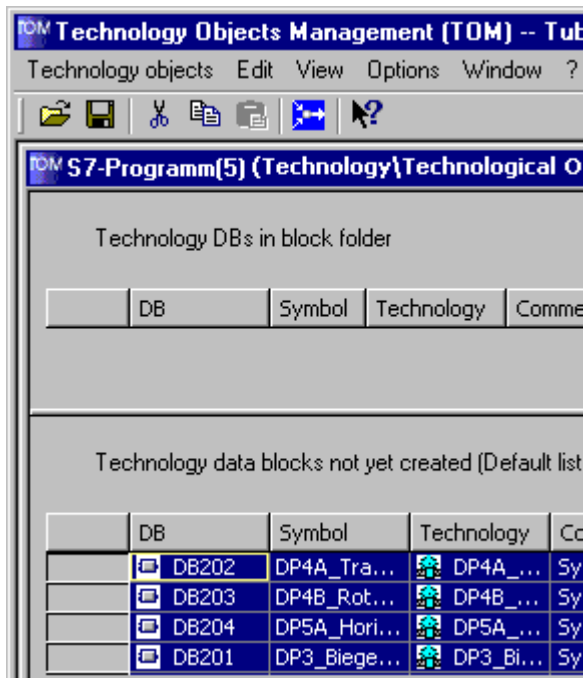
选择所有现有的工艺 DB

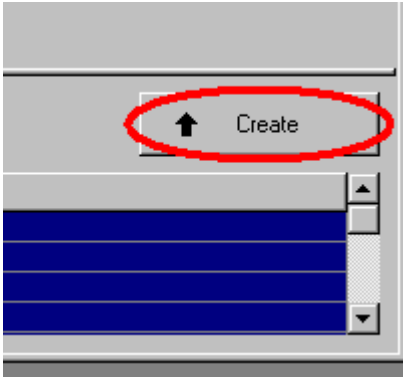


3. 接着，单击“删除”(Delete)。



删除之后，在工艺 DB 区域中标记所有尚未生成的工艺 DB。



4.	单击“生成”(Generate) 以生成具有新结构的工艺 DB。 
5.	关闭“工艺对象管理”(Technology Objects Management)

9.2.10 升级后如何更新工艺功能？

问题：

升级后如何更新工艺功能？

解答：

要使用已升级版本的功能，需要在项目的块文件夹中更新工艺功能。 步骤：

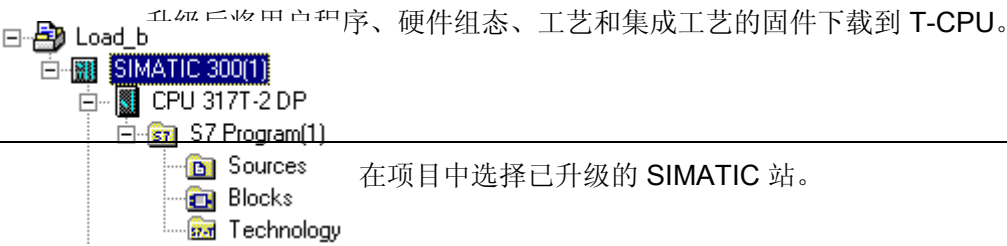
1.	在 SIMATIC Manager 的块文件夹中选择“块”(Blocks) 对象，然后选择快捷菜单命令“ 检查块一致性 ”(Check block consistency)。
2.	清除所有显示的错误。
3.	在 SIMATIC Manager 中，选择菜单命令“ 文件 ”(File) > “ 打开 ”(Open)，然后选择“ 库 ”(Libraries) 选项卡。 打开“S7-Tech”库和相应固件版本的目录。
4.	使用库文件夹的相应工艺功能更新块文件夹中的所有工艺功能。
5.	在 SIMATIC Manager 的块文件夹中选择“块”(Blocks) 对象，然后再次选择快捷菜单命令“ 检查块一致性 ”(Check block consistency)。
6.	清除所有显示的错误。

9.2.11 升级后需要将哪些内容下载到 PLC?

问题:

升级后需要将哪些数据下载到 PLC?

解答:



1.	Sources Blocks Technology 在项目中选择已升级的 SIMATIC 站。
2.	选择 PLC > 下载 (Download) T-CPU 将在下载后自动执行存储器复位 ¹⁾。 下载完成后，才能使用新性能特性。

¹⁾ 如果集成工艺固件版本与 T-CPU 装载存储器中的版本相同，则 T-CPU 在下载后不自动执行存储器复位。

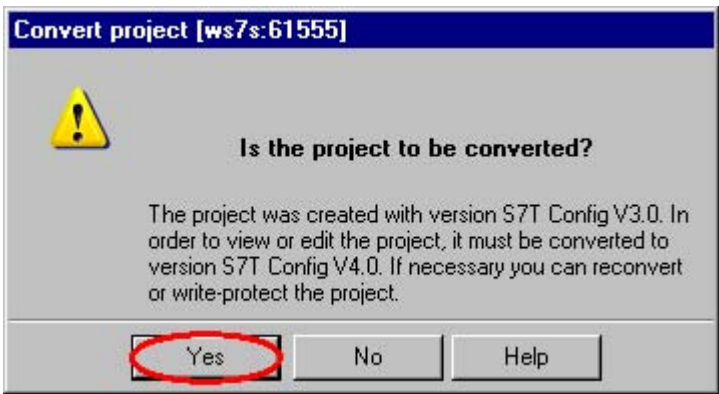
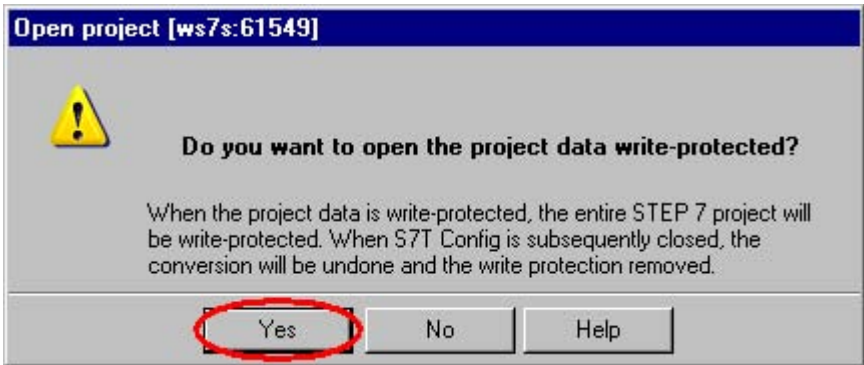
9.2.12 如何使用 S7-Technology 分析早期版本中的项目？

问题：

当前项目是在早期版本的 S7-Technology 中创建的。可以使用当前 S7-Technology 分析早期版本中的项目吗？

解答：

可以。可以使用当前 S7-Technology 分析早期版本中的项目。写保护您的项目数据。步骤：

1.	在 SIMATIC Manager 中，选择“工艺对象”(Technological objects)，然后选择“选项” (Options) > “组态工艺” (Configure technology)。
2.	打开 S7T Config 时会出现一个对话框（如下所示）： 
3.	转换项目以便可以在 S7T Config 中查看它。单击“是”(Yes) 确认此问题。
4.	下一个对话框将打开：  单击“是”(Yes) 确认此问题。 整个 STEP 7 项目将继承写保护设置。如果您稍后关闭了 S7T Config，则转换会被撤消，写保护会被清除。

5.	执行必要的分析。
6.	关闭 S7T Config。
7.	关闭该工具时会出现下面所示的对话框。 Close project [ws7s:61560]  Do you want to convert the project back to version S7T Config V3.0? During opening, the project has been converted to the current version S7T Config V4.0. Yes No Help 单击“是”(Yes) 确认此问题，以便将项目重新转换为其原始版本（该转换被拒绝）。

说明

浏览器图标在在线分析期间的含义：

- 具有绿色图标的对象是一致的
- 具有红色/绿色图标的对象的状态不确定，即，不能确定它们状态为“一致”还是“不一致”。

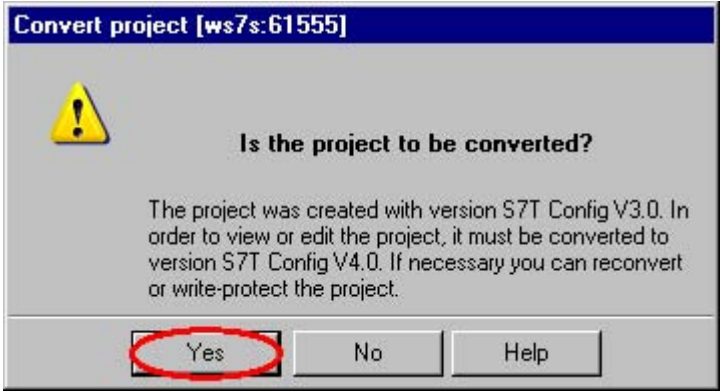
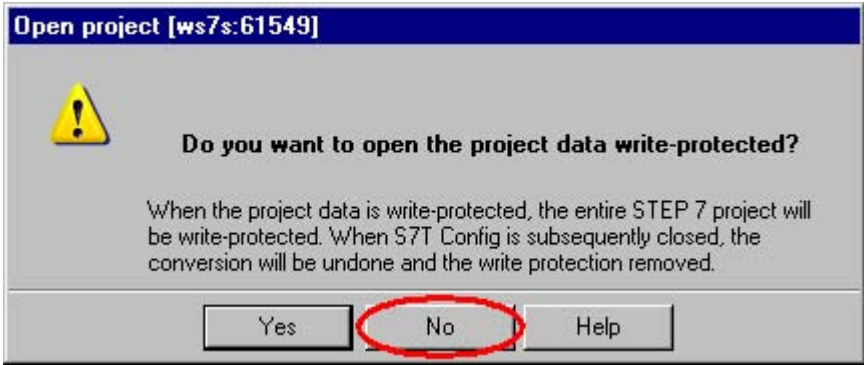
9.2.13 如何使用 S7-Technology 更改早期版本中的项目？

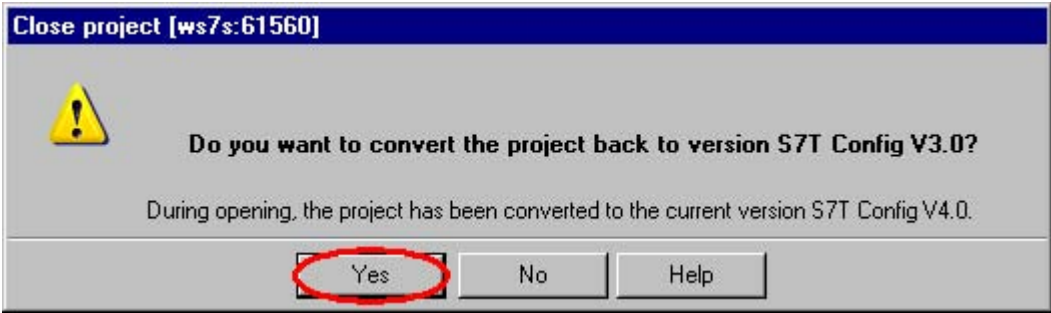
问题：

当前项目是在早期版本的 S7-Technology 中创建的。我可以使用当前版本的 S7-Technology 更改项目内容并保持对原始版本的向下兼容性吗？

解答：

可以，您可以更改项目以使其保持对原始版本的兼容性。步骤：

1.	在 SIMATIC Manager 中，选择“工艺对象”(Technological objects)，然后选择“选项” (Options) > “组态工艺” (Configure technology)。
2.	打开 S7T Config 时会出现一个对话框（如下所示）： 
3.	您需要在 S7T Config 中转换项目以对其进行编辑。单击“是”(Yes) 确认此问题。
4.	下一个对话框将打开：  单击“否”(No) 确认此问题。
5.	进行相关更改。
6.	选择项目 (Project) > 保存并编译全部 (Save and compile all)

6a.	如果您尚未重新创建或删除任何工艺对象，请继续步骤 7。 如果您重新插入或删除了工艺对象，请打开“工艺对象管理”(Technology Objects Management)（通过在 SIMATIC Manager 中双击“工艺对象”[Technological objects]）。
6b.	按照『创建和管理工艺 DB』中的描述创建或删除相应的工艺 DB。
6c.	关闭“工艺对象管理”(Technology Objects Management)。
7.	关闭 S7T Config。
8.	关闭该工具时会出现下面所示的对话框。 
9.	单击“是”(Yes) 确认此问题，以便将项目重新转换为其原始版本。

说明

转换的项目或重新转换的项目的离线/在线比较，对于完全相同的项目数据，可能会返回不一致。

原因： 后来的版本中进行了内部数据结构调整。

补救措施： 执行以下步骤以更正显示的不一致：

- 1. 在 S7T Config 中“保存并编译全部”
- 2. 将项目数据下载到 T-CPU

在相关版本的 S7-Technology 中执行这些步骤。

如果您在关闭 S7T Config 之前忘记重新转换项目，则可以通过在 S7T Config 中调用项目 (Project) > 以旧项目格式保存 (Save in old project format) 命令来满足要求。

9.2.14 使用“另存为”并进行重组时，要遵循什么规定？

如果出现以下情况，请打开 S7T Config：

- 替换模块后
- 复制某个站后
- 执行“文件”(File) > “另存为”(Save as) 命令并选中“通过重新组织（慢速）”(With reorganization [slow]) 复选框后

请按照屏幕上的指示操作，然后关闭 S7T Config。

如果您在早期版本的 S7-Technology 中创建、保存和重新组织了某个项目，系统可能会输出一条消息，报告某些工艺对象无法复制。

您可以忽略此消息，因为实际上已复制了这些工艺对象。

9.2.15 为什么某些系统 DB 具有不同的在线/离线时间戳？

问题：

项目中的某些 SDB 与 T-CPU 中的 SDB 相比，具有不同的时间戳。

此时间戳差异会导致出现问题吗？

解答：

如果在 S7T Config 中使用“目标系统”(Target system) > “下载”(Download) > “工艺到目标系统”(Technology to target system) 命令将工艺下载到 T-CPU，则给系统数据块类型 3311 分配 CPU 的集成实时时钟的时间戳。此时，此类不同的时间戳不会导致出现问题。

在 T-CPU 的系统数据块类型 3311 下载过程中，给它分配 CPU 的集成实时时钟的时间戳。可能会给具有相同内容的在线/离线 SDB 分配不同的时间戳。如果 S7T Config 的在线视图中的所有驱动器都标记为绿色，您可以认为您的工艺组态是一致的。

说明

如果在 Windows 中设置了“自动夏令时/冬令时”(Automatic summertime/wintertime)，系统数据的时间戳在夏令时期间会多一个小时。

9.2.16 STEP 7 报告“存储空间不足”(Insufficient memory space)

问题:

将块下载到 T-CPU 期间，STEP 7 会返回消息“模块上的存储空间不足，是否要压缩存储空间？”(Insufficient memory space on the module. Do you want to compress load memory?) 压缩数据后该消息仍然出现，但是有足够的工作存储器可用。

解答:

该消息的原因可能是：

- T-CPU 包含项目中未列出的“孤立”工艺 DB。可以根据创建语言“TechObj”识别工艺 DB。

通过在 SIMATIC Manager 中选择视图 (View) > 详细信息 (Details) 检查组态。
从 T-CPU 中删除所有未使用的工艺 DB。

9.2.17 超时后，CPU 偶尔会跳转到 STOP

问题:

尽管已调节系统时钟，但是当在连续模式下操作时，由于集成工艺超时，CPU 偶尔会跳转到 STOP。

解答:

基于命令速率和可能的出错原因，集成工艺上的系统负载会产生波动。如果选定的系统时钟太短，则可能会在集成工艺上触发超时，从而导致 CPU 跳转至 STOP。

如果发生此情况，请延长系统时钟，以便系统在较长的时间内不再响应超时错误。

9.2.18 对工艺数据执行“保存并编译全部”(Save and compile all) 功能时出错

问题:

我仅激活了默认参数值，但是，当在 S7T Config 中对工艺数据执行“保存并编译全部”(Save and compile all) 功能时却出现错误，为什么？

解答:

出于安全考虑，S7T Config 中所有安全相关和功能相关的参数或地址都分配了默认值，以防止执行“保存并编译全部”(Save and compile all) 功能。

这就是为什么在设置安全相关和功能相关的参数时始终应该履行某些注意事项，并应该验证它们的值和逻辑环境。

9.3 S7T Config 中的专家列表

9.3.1 使用专家列表

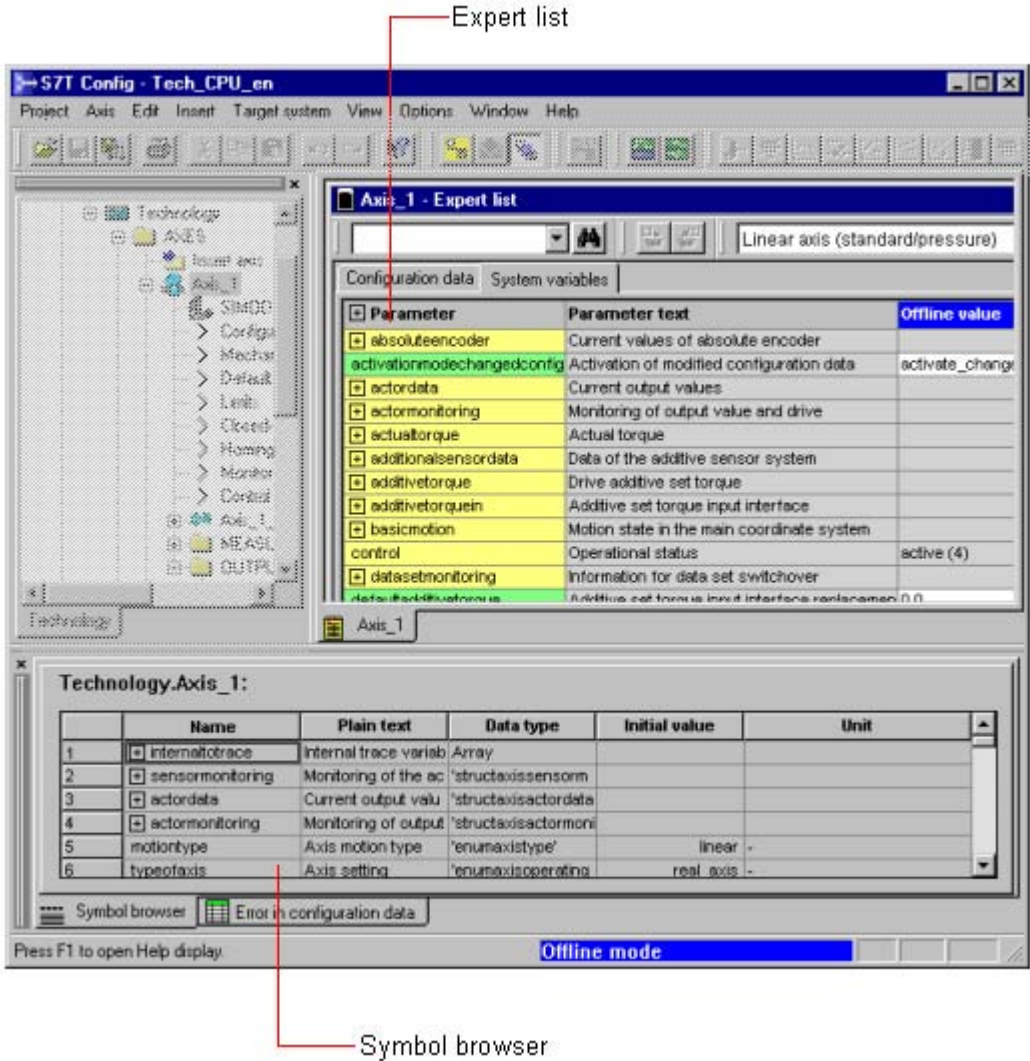
处理以下章节所述设置必须具备使用 S7T Config 中专家列表的经验。S7T Config 中的专家列表提供对工艺对象所有组态数据和系统变量的读/写访问权。该列表包含在向导或在编程窗口中无法设置的数据。

每个组态的工艺对象都有单独的专家列表可供使用。

 小心
专家模式适用于有经验的用户。不正确的输入可能会导致驱动器的不可控响应！ 系统不对您的输入进行似然性检查！ 不正确的更改可能会导致工艺对象不一致。

调用专家列表：

- 1. 从浏览器中选择工艺对象（例如轴），以查看或编辑其组态数据或系统变量。
 - 2. 从快捷菜单中选择**专家 (Expert) > 专家列表 (Expert list)** 命令。
- 将在工作区中打开专家列表。



专家列表的左窗格中显示系统变量和组态数据的树结构。右窗格输出系统变量和组态数据的表格视图。

从树结构中选择一个元素，以查看该表中的相应参数值。

需要重启的值和所有后来更改的组态数据在执行重启之前均以棕色表示。

单击您要编辑的值。键入一个新值或选择一个符号名称（灰显的值不能编辑）。使用 **ENTER** 保存您的更改。设置“集中更改”(Collect changes) 复选框来更改一组值。进行更改，然后通过单击“激活”(Activate) 确认您的输入。将保存并接受您的所有更改。

在线模式下更改的有效性

- 将立即应用修改的系统变量。
- 根据“有效性”(Effectiveness) 列的设置应用组态数据中的更改。
 - “立即”

当您按 ENTER 键或单击“激活”(Activate) 时激活值。
 - “重启”

仅当您通过调用“MC_Reset”工艺功能重启工艺对象之后，才激活值。
 - “下载”

无法在在线模式下编辑值。更改为离线模式，编辑工艺，然后将您的更改下载到目标系统。
- 当 T-CPU 从 STOP 更改为 RUN 时，会保留对系统变量和组态数据所作的更改。
- 对系统变量和组态数据所作的更改不具有保持性，在断开、接通电源时会丢失。

从浏览器中选择工艺，然后选择**目标设备 (Target device) > 将当前数据复制到 ROM (Copy current data to ROM)**，以便将组态数据的更改保存到非易失存储器。

为了将值永久地保存到项目中，请在离线模式下进行更改，保存并编译工艺数据，然后将这些数据下载到目标设备。您还可以通过执行**目标系统 (Target system) > 上传 (Upload) > 组态数据到 PG (Configuration data to PG)** 命令，将对组态数据所作的更改上传到工艺的项目数据中。然后，通过调用菜单命令**项目 (Project) > 保存并编译全部 (Save and Compile All)** 保存并编译工艺数据

离线模式下更改的有效性

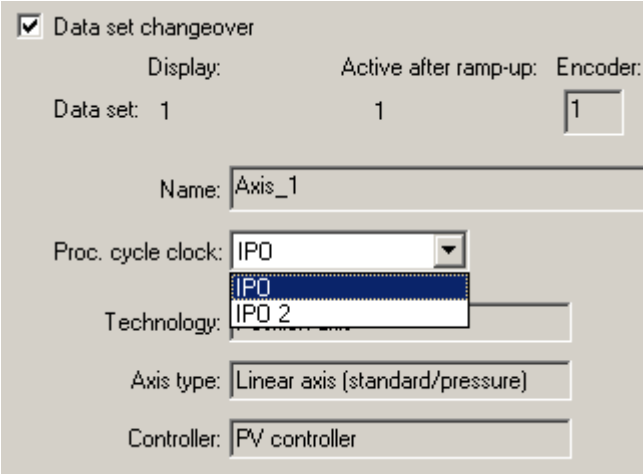
离线模式更改中的更改仅影响工艺项目数据。为了激活 T-CPU 中的更改，请调用**项目 (Project) > 保存并编译全部 (Save and Compile All)** 命令，然后将这些数据下载到目标系统（菜单命令**目标系统 [Target system] > 下载 [Download] > 工艺数据到目标系统 [Technology Data to Target System]**），保存和编译工艺数据。

9.3.2 轴设置

9.3.2.1 分配插补器周期 2

缺省情况下，在插补器周期内处理轴、外部编码器、输出凸轮和测量输入。也可以为较低优先级等级的工艺对象分配插补器周期 2。

在**轴 (Axis) > 组态 (Configuration)** 对话框中，从“执行周期”(Execution cycle) 下拉列表框中选择所需的插补器周期。



必须在相同的处理周期内操作同步轴和相应的同步对象。对同步轴的处理周期所做的任何更改也必须应用于同步对象：

1. 在 S7T Config 的浏览器中，选择相应同步轴的同步对象。
2. 从快捷菜单中选择**专家 (Expert) > 专家列表 (Expert list)** 命令。
3. 可以在 *Execution.executionlevel* 组态数据元素中更改处理周期时钟。

说明

不能在线分配插补器周期！

在 SIMATIC 管理器中，“保存并编译全部”工艺数据，然后将系统数据下载到 T-CPU。

9.3.2.2 扭矩设置

您可以通过设置“MC_SetTorqueLimit”和“MC_MoveToEndPos”工艺功能（不适用于跟随误差检测），来减小驱动电机的扭矩。扭矩以 N/m 为单位进行定义。为此，您必须在组态数据元素“*TypeOfAxis.SetPointDriverInfo.DriveData.maxTorque*”中设置电机的参考扭矩。

百分比减少

为了使用减少百分比来重写绝对值，请将参考扭矩设置为 100.0 N/m:

组态数据元素: *TypeOfAxis.SetPointDriverInfo.DriveData.maxTorque = 100.0 Nm*

在调用“MC_SetTorqueLimit”时，通过设置参数 *MaxTorque = 25.0* 将电机的最大扭矩减小到 25%。

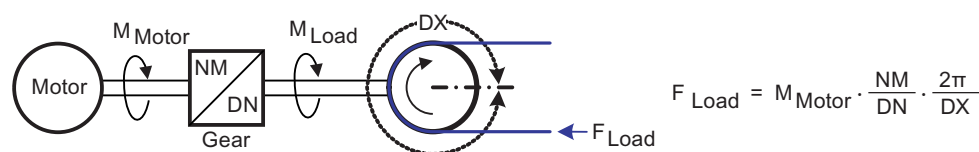
计算负载端的扭矩

您可以计算负载端的电机扭矩，同时考虑齿轮负载和丝杠螺距。以下部分中将忽略摩擦损耗。

回转轴:



线性轴（此处将计算力）:



$M_{电机}$	驱动电机扭矩（以 [N/m] 为单位）
$M_{负载}$	齿轮负载端的扭矩（以 [N/m] 为单位）
$F_{负载}$	负载力（以 [N] 为单位）
NM	电机旋转次数
DN	负载旋转次数
DX	每转的负载进给（例如丝杠螺距，以 [m] 为单位）

9.3.3 监视功能

9.3.3.1 编码器监视功能

- 监视增量编码器的零标记

您可以激活监视两个编码器零标记之间增量数的功能。如果在定义的距离内未达到编码器零标记，则触发报警。

在使用编码器零标记的回原点模式下，当轴通过参考凸轮后，将监视距离。

- 对绝对编码器实际值所作的有效更改

用户可以激活对绝对编码器实际值所作的有效更改进行监视的功能。

- 实际速度

可以监视允许的实际速度最大值。如果超过最大值，会输出系统变量 *sensordata.sensormonitoring.velocity* 为 *limitexceeded*。速度不只限于该值。

9.3.3.2 编码器限制频率监视

系统监视编码器限制频率。它还监视实际速度/实际加速度，以检测驱动器控制回路中的错误。如果实际值超过了编码器限制频率，则工艺 DB 会报告错误。在工艺 DB 的 *ErrorStatus*“SensorFreqViolation”中指示该错误。

9.4 工艺参数

9.4.1 工艺参数 0001 至 0999 — 轴

利用本章提供的信息来读取或更改用户程序中工艺对象的参数。工艺对象的参数可以使用“MC_ReadSysParameter”工艺功能来读取，并使用“MC_WriteParameter”工艺功能来写入。

编辑工艺参数的辅助条件：

- 如果在参数中输入的新值等于旧值，则不需要在列表中指定的 **Restart**。此外，在更改虚拟轴的参数时，也不必执行 **Restart**。是否需要 **Restart** 在相应工艺 DB 的 *Statusword.RequestRestart* 变量中指明。
- 由参数编号和索引定义的参数必须在系统中可用并且允许写访问。必须使用工艺功能“MC_ChangeDataset”在 T-CPU 中启用由索引定义的数据记录。

关键字

编号	“MC_ReadSysParameter”和“MC_WriteParameter”工艺功能的输入参数 <i>ParameterNumber</i> 的参数编号
索引	索引可用于多次访问现有参数。以下区域中存在多个参数： • 数据集（Dataset_1 到 Dataset_16） • 编码器（Encoder_1 到 Encoder_8） • 同步对象（1 代表同步对象，2 代表叠加的同步对象） 可以从固件版本为 V3.1.x 和更高版本的集成工艺使用索引。
DB 参数中的名称 （“正常”显示）	如果在 STEP 7 中打开“MC_ReadSysParameter”或“MC_WriteParameter”工艺功能的实例数据块，则在“DB 参数”中会显示一个组态对话框。 在“正常”显示模式下显示参数名称
DB 参数中的名称 （“专家”显示）	如果在 STEP 7 中打开“MC_ReadSysParameter”或“MC_WriteParameter”工艺功能的实例数据块，则在“DB 参数”中会显示一个组态对话框。 在“专家”显示模式下显示参数名称 此名称对应于系统变量的名称或 S7T Config 的“专家”列表中组态数据元素的名称。
说明	参数说明

参数类型	指的是 S7T Config 的“专家”列表中的参数类型“系统变量”或“组态数据”。
数据类型	指定参数的数据类型。 可以在“DINT 值的列表”中找到“DINT Enum”数据类型的可用值。
最小值/最大值	指定参数的上限值和下限值。
访问	Read: 可以读取参数。 Write: 可以更改参数。
活动	立即: 在参数中所做的更改立即生效。 Restart: 参数中的更改需要重启才能生效, 意味着重新初始化工艺对象时生效。这种情况仅在工艺对象 TO 被禁用的情况下才可能发生。
适用对象	对于轴参数, 指定该参数适用的轴类型。
固件版本	可从中使用参数的集成工艺的固件版本。

编号	索引	参数说明	
1	-	DB 参数中的名称 (“正常”显示模式)	Setpoints.Position.Target position
		DB 参数中的名称 (“专家”显示模式)	<i>positioningstate.commandposition</i>
		说明	轴位置设定值
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read
		活动	-
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
2	-	DB 参数中的名称 (“正常”显示模式)	Limits.Position and velocity.Software limit switch.Negative
		DB 参数中的名称 (“专家”显示模式)	<i>swlimit.minusposition</i>
		说明	负软件限位开关
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	
3	-	DB 参数中的名称 (“正常”显示模式)	Limits.Position and velocity.Software limit switch.Positive
		DB 参数中的名称 (“专家”显示模式)	<i>swlimit.plusposition</i>
		说明	正软件限位开关
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
4	-	DB 参数中的名称 (“正常”显示模式)	Limits.Position and velocity.Software limit switch.Activation
		DB 参数中的名称 (“专家”显示模式)	<i>swlimit.state</i>
		说明	激活软件限位开关
		参数类型	系统变量
		数据类型	DINT EnumActiveInactive (页 1148)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	
6	1-16	DB 参数中的名称 (“正常”显示模式)	Monitoring.Following error monitoring.Activation
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfDataSets.DataSet_x.DynamicFollowing.enable</i>
		说明	激活动态跟随误差监视
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
7	1-16	DB 参数中的名称 (“正常”显示模式)	Monitoring.Following error monitoring.Maximum
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfDataSets.DataSet_x.DynamicFollowing.maxPositionTolerance</i>
		说明	跟随误差特性的上阈值
		参数类型	组态数据元素
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	
8	-	DB 参数中的名称 (“正常”显示模式)	Limits.Position and velocity.Velocity.Maximum
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.MaxVelocity.maximum</i>
		说明	允许的最大速度
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	Restart
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
10	-	DB 参数中的名称 (“正常”显示模式)	Actual values.Motion.Velocity
		DB 参数中的名称 (“专家”显示模式)	<i>motionstatedata.actualvelocity</i>
		说明	实际轴速度
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E12
		最大值	+1E12
		访问	Read
		活动	-
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
11	-	DB 参数中的名称 (“正常”显示模式)	Setpoints.Motion.Velocity
		DB 参数中的名称 (“专家”显示模式)	<i>motionstatedata.commandvelocity</i>
		说明	轴速度设定值
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E12
		最大值	+1E12
		访问	Read
		活动	-
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
12	-	DB 参数中的名称 (“正常”显示模式)	Limits.Dynamic response.Acceleration.Maximum
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.MaxAcceleration.maximum
		说明	允许的最大加速度值
		参数类型	组态数据元素
		数据类型	REAL
		最小值	-1E12
		最大值	+1E12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
16	-	DB 参数中的名称 (“正常”显示模式)	Limits.Dynamic response.Jerk.Maximum
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.MaxJerk.maximum
		说明	允许的最大冲击值
		参数类型	组态数据元素
		数据类型	REAL
		最小值	-1E12
		最大值	+1E12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

9.4.2 工艺参数 1000 到 1999 — 轴

利用本章提供的信息来读取或更改用户程序中工艺对象的参数。工艺对象的参数可以使用“MC_ReadSysParameter”工艺功能来读取，并使用“MC_WriteParameter”工艺功能来写入。

编辑工艺参数的辅助条件：

- 如果在参数中输入的新值等于旧值，则不需要在列表中指定的 **Restart**。此外，在更改虚拟轴的参数时，也不必执行 **Restart**。是否需要 **Restart** 在相应工艺 DB 的 *Statusword.RequestRestart* 变量中指明。
- 由参数编号和索引定义的参数必须在系统中可用并且允许写访问。必须使用工艺功能“MC_ChangeDataset”在 T-CPU 中启用由索引定义的数据记录。

关键字

编号	“MC_ReadSysParameter”和“MC_WriteParameter”工艺功能的输入参数 <i>ParameterNumber</i> 的参数编号
索引	索引可用于多次访问现有参数。以下区域中存在多个参数： • 数据集（Dataset_1 到 Dataset_16） • 编码器（Encoder_1 到 Encoder_8） • 同步对象（1 代表同步对象，2 代表叠加的同步对象） 可以从固件版本为 V3.1.x 和更高版本的集成工艺使用索引。
DB 参数中的名称 （“正常”显示）	如果在 STEP 7 中打开“MC_ReadSysParameter”或“MC_WriteParameter”工艺功能的实例数据块，则在“DB 参数”中会显示一个组态对话框。 在“正常”显示模式下显示参数名称
DB 参数中的名称 （“专家”显示）	如果在 STEP 7 中打开“MC_ReadSysParameter”或“MC_WriteParameter”工艺功能的实例数据块，则在“DB 参数”中会显示一个组态对话框。 在“专家”显示模式下显示参数名称 此名称对应于系统变量的名称或 S7T Config 的“专家”列表中组态数据元素的名称。
说明	参数说明
参数类型	指的是 S7T Config 的“专家”列表中的参数类型“系统变量”或“组态数据”。
数据类型	指定参数的数据类型。 可以在“DINT 值的列表”中找到“DINT Enum”数据类型的可用值。
最小值/最大值	指定参数的上限值和下限值。

访问	读取： 可以读取参数。 写入： 可以更改参数。
活动	立即： 在参数中所做的更改立即生效。 重启： 参数中的更改需要重启才能生效，意味着重新初始化工艺对象时生效。 这种情况仅在工艺对象 TO 被禁用的情况下才可能发生。
适用对象	对于轴参数，指定该参数适用的轴类型。
固件版本	可从中使用参数的集成工艺的固件版本。

编号	索引	参数说明	
1100	-	DB 参数中的名称 (“正常”显示模式)	Mechanics.Spindle settings.Spindle pitch for each revolution of the axis
		DB 参数中的名称 (“专家”显示模式)	<i>LeadScrew.pitchVal</i>
		说明	轴每转的丝杠螺距
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	+1E12
		访问	Read, Write
		活动	Restart
		适用对象	定位轴、同步轴
		固件版本	
1110	-	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Extrapolator.Extrapolation time
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.Extrapolation.ExtrapolationTime</i>
		说明	主轴的推断时间
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	+1E12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
1111	-	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Extrapolator.Filter.Mode
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.Extrapolation.Filter.Mode
		说明	实际值平滑的选择标准
		参数类型	组态数据元素
		数据类型	DINT EnumAxisFilterMode (页 1149)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		适用对象	定位轴、同步轴
		固件版本	
1112	-	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Extrapolator.Filter.Activation
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.Extrapolation.Filter.Enable
		说明	激活状态
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
1113	-	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Extrapolator.Filter.Time constant
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.Extrapolation.Filter.TimeConstant</i>
		说明	用于 PT1 平滑和求平均值的时间常量
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	
1114	-	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Extrapolator.Tolerance range.Activation
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.Extrapolation.ToleranceRange.Enable</i>
		说明	激活公差范围
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
1115	-	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Extrapolator.Tolerance range.Tolerance window
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.Extrapolation.ToleranceRange.Value
		说明	公差窗口
		参数类型	组态数据元素
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	
1116	-	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Extrapolator.Velocity signal
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis Extrapolation.extrapolatedVelocitySwitch
		说明	选择速度主值
		参数类型	组态数据元素
		数据类型	DINT EnumAxisExtrapolatedVelocitySwitch (页 1149)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
1120	-	DB 参数中的名称 (“正常”显示模式)	Homing.Distance to the bero
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.Homing.beroDistance</i>
		说明	与回原点输出凸轮/外部零标记的最大距离
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	
1121	-	DB 参数中的名称 (“正常”显示模式)	Homing.Approach direction
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.Homing.direction</i>
		说明	回原点方向
		参数类型	组态数据元素
		数据类型	DINT EnumDirectionType (页 1157)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
1122	-	DB 参数中的名称 (“正常”显示模式)	Homing.Bero monitoring activation
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.Homing.enableBeroDistance</i>
		说明	监视模式激活状态
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	
1123	-	DB 参数中的名称 (“正常”显示模式)	Homing.Homing required
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.Homing.referencingNecessary</i>
		说明	需要回原点
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
1130	-	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Extrapolator.Position filter.Activation
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.Extrapolation.ExtrapolationPositionFilter.enable
		说明	激活过滤器
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		适用对象	同步轴
		固件版本	
1131	-	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Extrapolator.Position filter.T1
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.Extrapolation.ExtrapolationPositionFilter.T1
		说明	第一个时间常量
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
1132	-	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Extrapolator.Position filter.T2
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.Extrapolation.ExtrapolationPositionFilter.T2
		说明	第二个时间常量
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	
1201	-	DB 参数中的名称 (“正常”显示模式)	Mechanics.Modulo.Activation
		DB 参数中的名称 (“专家”显示模式)	Modulo.state
		说明	激活模数设置
		参数类型	组态数据元素
		数据类型	DINT EnumActiveInactive (页 1148)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
1202	-	DB 参数中的名称 (“正常”显示模式)	Mechanics.Modulo.Length
		DB 参数中的名称 (“专家”显示模式)	<i>Modulo.length</i>
		说明	模数长度
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	Restart
		适用对象	定位轴、同步轴
		固件版本	
1203	-	DB 参数中的名称 (“正常”显示模式)	Mechanics.Modulo.Starting value
		DB 参数中的名称 (“专家”显示模式)	<i>Modulo.startValue</i>
		说明	模数起始值
		参数类型	组态数据元素
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	Restart
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
1211	-	DB 参数中的名称 (“正常”显示模式)	Monitoring.Acceleration.Activation
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.ActualAccelerationMonitoring.enable
		说明	激活实际加速度监视
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	
1212	-	DB 参数中的名称 (“正常”显示模式)	Monitoring.Acceleration.Maximum value
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.ActualAccelerationMonitoring.maximum
		说明	允许的最大加速度实际值
		参数类型	组态数据元素
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
1221	-	DB 参数中的名称 (“正常”显示模式)	Monitoring.Velocity.Activation
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.ActualVelocityMonitoring.enable
		说明	激活实际速度监视
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
1222	-	DB 参数中的名称 (“正常”显示模式)	Monitoring.Velocity.Maximum value
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.ActualVelocityMonitoring.maximum
		说明	允许的最大实际速度
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
1231	-	DB 参数中的名称 (“正常”显示模式)	Position control.Static controller data.Drift compensation
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.Drift.enable</i>
		说明	激活漂移补偿
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
1241	-	DB 参数中的名称 (“正常”显示模式)	Limits.Position and velocity.Emergency stop delay
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.EmergencyRampGenerator.maxDeceleration</i>
		说明	减速斜坡的延迟
		参数类型	组态数据元素
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
1251	-	DB 参数中的名称 (“正常”显示模式)	Position control.Static controller data.Fine interpolator type
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.FineInterpolator._type</i>
		说明	精插补器类型
		参数类型	组态数据元素
		数据类型	DINT EnumAxisFineInterpolatorMode (页 1150)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
1261	-	DB 参数中的名称 (“正常”显示模式)	Position control.Friction compensation.Amplitude
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.Friction.amplitude</i>
		说明	摩擦补偿特性的振幅
		参数类型	组态数据元素
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
1262	-	DB 参数中的名称 (“正常”显示模式)	Position control.Friction compensation.Decay time constant
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.Friction.decayTime</i>
		说明	衰减时间常量
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
1263	-	DB 参数中的名称 (“正常”显示模式)	Position control.Friction compensation.Standstill decay time constant
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.Friction.delayTimeStandStill</i>
		说明	停止信号的延迟时间
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	$2^{31} \times \text{伺服系统时间} \times 9$
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
1264	-	DB 参数中的名称 (“正常”显示模式)	Position control.Friction compensation.Activation
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.Friction.enable</i>
		说明	激活摩擦补偿
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
1265	-	DB 参数中的名称 (“正常”显示模式)	Position control.Friction compensation.Standstill maximum velocity
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.Friction.maxVeloStandStill</i>
		说明	停止信号的最大值
		参数类型	组态数据元素
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
1271	-	DB 参数中的名称 (“正常”显示模式)	Monitoring.Synchronous operation monitoring.Actual value tolerance
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.GearingPosTolerance.actualValueTolerance
		说明	允许的实际值公差
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	
1272	-	DB 参数中的名称 (“正常”显示模式)	Monitoring.Synchronous operation monitoring.Setpoint tolerance
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.GearingPosTolerance.commandValueTolerance
		说明	允许的设定值公差
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	

编号	索引	参数说明	
1273	-	DB 参数中的名称 (“正常”显示模式)	Monitoring.Synchronous operation monitoring.Activation of actual value monitoring
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.GearingPosTolerance.enableActualValue</i>
		说明	监视实际值公差
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	
1274	-	DB 参数中的名称 (“正常”显示模式)	Monitoring.Synchronous operation monitoring.Activation of setpoint monitoring
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.GearingPosTolerance.enableCommandValue</i>
		说明	监视设定值公差
		参数类型	组态数据元素
		数据类型	DINT EnumGearingPosToleranceCommandValue (页 1161)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	

编号	索引	参数说明	
1275	-	DB 参数中的名称 (“正常”显示模式)	Monitoring.Synchronous operation monitoring.Message to master
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.GearingPosTolerance.enableErrorReporting</i>
		说明	公差监视的错误报告
		参数类型	组态数据元素
		数据类型	DINT EnumErrorReporting (页 1158)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	
1281	-	DB 参数中的名称 (“正常”显示模式)	Limits.Position and velocity. Hardware limit switch.Activation
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.HWEndPos.enable</i>
		说明	激活硬件限位开关
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
1284	-	DB 参数中的名称 (“正常”显示模式)	Limits.Position and velocity. Hardware limit switch.Mode
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.HWEndPos.mode
		说明	硬件限位开关相对于轴行程范围的位置。
		参数类型	组态数据元素
		数据类型	DINT EnumMountSwitch (页 1164)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
1291	-	DB 参数中的名称 (“正常”显示模式)	Limits.Dynamic response.Acceleration.Activation of the monitoring
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.MaxAcceleration.enableSetPointMonitoring
		说明	激活受控变量监视
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
1301	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Dead zone compensation.Center point
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NeutralBand.centreValue
		说明	死区范围的中心点
		参数类型	组态数据元素
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
1302	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Dead zone compensation.Activation
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NeutralBand.enable
		说明	激活死区补偿
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
1303	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Dead zone compensation.Expansion
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NeutralBand.range
		说明	扩大死区范围
		参数类型	组态数据元素
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
1311	-	DB 参数中的名称 (“正常”显示模式)	Monitoring.Positioning and standstill monitoring.Message delay time
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.PositionMonitoring.posWinTolDelayTime
		说明	消息激活延迟
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
1312	-	DB 参数中的名称 (“正常”显示模式)	Monitoring.Positioning and standstill monitoring.Tolerance window delay time
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.PositionMonitoring.posWinTolTime
		说明	到达公差窗口之前持续的延迟时间
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	
1313	-	DB 参数中的名称 (“正常”显示模式)	Monitoring.Positioning and standstill monitoring.Tolerance
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.PositionMonitoring.tolerance
		说明	定位窗口的宽度
		参数类型	组态数据元素
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
1321	-	DB 参数中的名称 (“正常”显示模式)	Mechanics.Measuring system.Reversal of direction of rotation
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.SetPointDriverInfo.InvertSetPoint.invSetPoint
		说明	激活旋转方向反转
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
1331	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Actual value smoothing.Activation
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.SmoothingFilter.enable
		说明	激活过滤器功能
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
1332	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Actual value smoothing.Mode
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.SmoothingFilter.mode
		说明	用于实际值平滑的计算方法
		参数类型	组态数据元素
		数据类型	DINT EnumAxisFilterMode (页 1149)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
1333	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Actual value smoothing.Time constant
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.SmoothingFilter.timeConstant
		说明	用于 PT1 平滑的时间常量
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
1341	-	DB 参数中的名称 (“正常”显示模式)	Monitoring.Positioning and standstill monitoring.Standstill message delay time
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.StandStillMonitoring.delayTimeToActivate</i>
		说明	消息激活延迟
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	
1342	-	DB 参数中的名称 (“正常”显示模式)	Monitoring.Positioning and standstill monitoring.Position tolerance in standstill
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.StandStillMonitoring.stillStandTolerance</i>
		说明	允许的停止位置公差
		参数类型	组态数据元素
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
1351	-	DB 参数中的名称 (“正常”显示模式)	Monitoring.Standstill signal.Message delay time
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.StandStillSignal.delayTimeToActivate</i>
		说明	触发停止信号的延迟时间
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	214748
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
1352	-	DB 参数中的名称 (“正常”显示模式)	Monitoring.Standstill signal.Degree of filtering
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.StandStillSignal.filterDegree</i>
		说明	过滤等级
		参数类型	组态数据元素
		数据类型	DINT
		最小值	0
		最大值	15
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
1353	-	DB 参数中的名称 (“正常”显示模式)	Monitoring.Standstill signal.Filter frequency
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.StandStillSignal.filterFrequency</i>
		说明	过滤器频率
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
1354	-	DB 参数中的名称 (“正常”显示模式)	Monitoring.Standstill signal.Velocity limit
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.StandStillSignal.maxVeloStandStill</i>
		说明	停止信号的速度限值
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
1400	-	DB 参数中的名称 (“正常”显示模式)	Position control.Static controller data.Manipulated variable limit.Activation
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.SpeedLimitation.Enable</i>
		说明	速度范围限制的激活状态
		参数类型	组态数据元素
		数据类型	DINT EnumAxisFilterMode (页 1149)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
1401	-	DB 参数中的名称 (“正常”显示模式)	Position control.Static controller data.Manipulated variable limit.Lower limit
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.SpeedLimitation.MinSpeed</i>
		说明	速度范围下限
		参数类型	组态数据元素
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
1402	-	DB 参数中的名称 (“正常”显示模式)	Position control.Static controller data.Manipulated variable limit.Upper limit
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.SpeedLimitation.MaxSpeed
		说明	速度范围上限
		参数类型	组态数据元素
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

9.4.3 工艺参数 2000 至 2999 — 轴

利用本章提供的信息来读取或更改用户程序中工艺对象的参数。工艺对象的参数可以使用“MC_ReadSysParameter”工艺功能来读取，并使用“MC_WriteParameter”工艺功能来写入。

编辑工艺参数的辅助条件：

- 如果在参数中输入的新值等于旧值，则不需要在列表中指定的 **Restart**。此外，在更改虚拟轴的参数时，也不必执行 **Restart**。是否需要 **Restart** 在相应工艺 DB 的 *Statusword.RequestRestart* 变量中指明。
- 由参数编号和索引定义的参数必须在系统中可用并且允许写访问。必须使用工艺功能“MC_ChangeDataset”在 T-CPU 中启用由索引定义的数据记录。

关键字

编号	“MC_ReadSysParameter”和“MC_WriteParameter”工艺功能的输入参数 <i>ParameterNumbe</i> 的参数编号
索引	索引可用于多次访问现有参数。以下区域中存在多个参数： - 数据集（Dataset_1 到 Dataset_16） - 编码器（Encoder_1 到 Encoder_8） - 同步对象（1 代表同步对象，2 代表叠加的同步对象） 可以从固件版本为 V3.1.x 和更高版本的集成工艺使用索引。
DB 参数中的名称 （“正常”显示）	如果在 STEP 7 中打开“MC_ReadSysParameter”或“MC_WriteParameter”工艺功能的实例数据块，则在“DB 参数”中会显示一个组态对话框。 在“正常”显示模式下显示参数名称
DB 参数中的名称 （“专家”显示）	如果在 STEP 7 中打开“MC_ReadSysParameter”或“MC_WriteParameter”工艺功能的实例数据块，则在“DB 参数”中会显示一个组态对话框。 在“专家”显示模式下显示参数名称 此名称对应于系统变量的名称或 S7T Config 的“专家”列表中组态数据元素的名称。
含义	参数说明
参数类型	指的是 S7T Config 的“专家”列表中的参数类型“系统变量”或“组态数据”。
数据类型	指定参数的数据类型。 可以在“DINT 值的列表”中找到“DINT Enum”数据类型的可用值。
最小值/最大值	指定参数的上限值和下限值。
访问	Read: 可以读取参数。 Write: 可以更改参数。
活动	立即: 在参数中所做的更改立即生效。 Restart: 参数中的更改需要重启才能生效，意味着重新初始化工艺对象时生效。这种情况仅在工艺对象 TO 被禁用的情况下才可能发生。
适用对象	对于轴参数，指定该参数适用的轴类型。
固件版本	可从中使用参数的集成工艺的固件版本。

编号	索引	参数说明	
2000	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Data block switching.Data blocks count
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfData Sets.numberOfDataSets
		含义	数据记录的数目
		参数类型	组态数据元素
		数据类型	DINT
		最小值	0
		最大值	16
		访问	Read
		活动	-
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
2001	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Data block switching.Data block initialization
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfDataSets.initDataSet
		含义	数据记录初始化
		参数类型	组态数据元素
		数据类型	DINT
		最小值	0
		最大值	16
		访问	Read, Write
		活动	Restart
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
2002	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Data block switching.Data block switching mode
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfData Sets.changeMode
		含义	数据记录更改的模式
		参数类型	组态数据元素
		数据类型	DINT EnumChangeMode (页 1156)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
2003	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Data block switching.Rise limitation
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfDataSets.smoothingTimeByChangeDifference
		含义	用于平滑受控变量跳跃的时间常量
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
2010	1-16	DB 参数中的名称 (“正常”显示模式)	Limits.Fixed stop.Following error
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfDataSets.DataSet_x.ClampingMonitoring.followingErrorDeviation</i>
		含义	检测固定挡块所需的跟随误差的声明
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	
2011	1-16	DB 参数中的名称 (“正常”显示模式)	Limits.Fixed stop.Position tolerance
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfDataSets.DataSet_x.ClampingMonitoring.positionTolerance</i>
		含义	紧固状态下实际值与设定值之间允许的偏移量
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
2012	1-16	DB 参数中的名称 (“正常”显示模式)	Limits.Fixed stop.Fixed stop detection mode
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfDataSets.DataSet_x.ClampingMonitoring.recognitionMode
		含义	扭矩监视激活模式
		参数类型	组态数据元素
		数据类型	DINT EnumRecognitionMode (页 1165)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		适用对象	定位轴、同步轴
		固件版本	
2020	1-16	DB 参数中的名称 (“正常”显示模式)	Mechanics.Load gearbox.Load revolutions count
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfData Sets.DataSet_x.Gear denFactor
		含义	变速箱系数的分母
		参数类型	组态数据元素
		数据类型	DINT
		最小值	0
		最大值	2147483647
		访问	Read, Write
		活动	Restart
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
2021	1-16	DB 参数中的名称 (“正常”显示模式)	Mechanics.Load gearbox.Motor revolutions count
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfData Sets.DataSet_x.Gear.numFactor</i>
		含义	变速箱系数的分子
		参数类型	组态数据元素
		数据类型	DINT
		最小值	0
		最大值	2147483647
		访问	Read, Write
		活动	Restart
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
2031	1-16	DB 参数中的名称 (“正常”显示模式)	Position control.Static controller data.PV controller.DSC activation
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfDataSets.DataSet_x.ControllerStruct.PV_Controller.enableDSC</i>
		含义	激活 DSC
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
2032	1-16	DB 参数中的名称 (“正常”显示模式)	Position control.Static controller data.PV controller.Weighting factor
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfDataSets.DataSet_x.ControllerStruct.PV_Controller.kpc
		含义	预控制加权
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	150.0
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	
2033	1-16	DB 参数中的名称 (“正常”显示模式)	Position control.Static controller data.PV controller.kp
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfDataSets.DataSet_x.ControllerStruct.PV_Controller.kv
		含义	比例控制器增益
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	< < 1E+7
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
2034	1-16	DB 参数中的名称 (“正常”显示模式)	Position control.Static controller data.PV controller.Activation of the precontrol
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfDataSets.DataSet_x.ControllerStruct.PV_Controller.preCon
		含义	激活预控制功能
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	重启
		适用对象	定位轴、同步轴
		固件版本	
2035	1-16	DB 参数中的名称 (“正常”显示模式)	Position control.Static controller data.PV controller.Activation/mode balancing filter
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfDataSets.DataSet_x.ControllerStruct.PV_Controller.balanceFilterMode
		含义	平衡滤波器模式
		参数类型	组态数据元素
		数据类型	DINT EnumBalanceFilterMode (页 1153)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
2041	1-16	DB 参数中的名称 (“正常”显示模式)	Position control.Static controller data.PD controller.kp
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfDataSets.DataSet_x.ControllerStruct.PD_Controller.kp
		含义	比例控制器增益
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	100
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	
2042	1-16	DB 参数中的名称 (“正常”显示模式)	Position control.Static controller data.PD controller.Delay time
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfDataSets.DataSet_x.ControllerStruct.PD_Controller.decayTime
		含义	DT1 元素的时间常量
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
2043	1-16	DB 参数中的名称 (“正常”显示模式)	Position control.Static controller data.PD controller.kd
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfDataSets.DataSet_x.ControllerStruct.PD_Controller.kd
		含义	DT1 元素的增益
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	
2051	1-16	DB 参数中的名称 (“正常”显示模式)	Position control.Dynamic compensation.Activation
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfDataSets.DataSet_x.DynamicComp.enable
		含义	激活动态补偿
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
2052	1-16	DB 参数中的名称 (“正常”显示模式)	Position control.Dynamic compensation.First time constant
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfDataSets.DataSet_x.DynamicComp.T1</i>
		含义	第一个时间常量
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	
2053	1-16	DB 参数中的名称 (“正常”显示模式)	Position control.Dynamic compensation.Second time constant
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfDataSets.DataSet_x.DynamicComp.T2</i>
		含义	第二个时间常量
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
2061	1-16	DB 参数中的名称 (“正常”显示模式)	Position control.Dynamic controller data.Position control loop time constant
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfDataSets.DataSet_x.DynamicData.positionTimeConstant</i>
		含义	位置控制回路的等效时间常数
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	
2062	1-16	DB 参数中的名称 (“正常”显示模式)	Position control.Dynamic controller data. 扭矩控制回路时间常量
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfDataSets.DataSet_x.DynamicData.torqueTimeConstant</i>
		含义	扭矩控制回路的等效时间常量
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
2063	1-16	DB 参数中的名称 (“正常”显示模式)	Position control.Dynamic controller data. 速率控制回路时间常量
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfDataSets.DataSet_x.DynamicData.velocityTimeConstant</i>
		含义	速度控制回路的等效时间常量
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
2071	1-16	DB 参数中的名称 (“正常”显示模式)	Monitoring.Following error monitoring.Minimum
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfDataSets.DataSet_x.DynamicFollowing.minPositionTolerance</i>
		含义	跟随误差特性的下阈限
		参数类型	组态数据元素
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
2072	1-16	DB 参数中的名称 (“正常”显示模式)	Monitoring.Following error monitoring.Minimum velocity
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfDataSets.DataSet_x.DynamicFollowing. minVelocity</i>
		含义	跟随误差特性的速度阈值下限
		参数类型	组态数据元素
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	
2073	1-16	DB 参数中的名称 (“正常”显示模式)	Monitoring.Following error monitoring.Warning limit
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfDataSets.DataSet_x.DynamicFollowing. warningLimit</i>
		含义	跟随误差监视的警告限值
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	100.0
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
2075	1-16	DB 参数中的名称 (“正常”显示模式)	Data record.Encoder number.Encoder number
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfDataSets.DataSet_x.EncoderNumber.EncoderNumber
		含义	传感器编号
		参数类型	组态数据元素
		数据类型	DINT
		最小值	0
		最大值	8
		访问	Read
		活动	-
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
2081	1-16	DB 参数中的名称 (“正常”显示模式)	Position control.Static controller data.PID controller.kp
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfDataSets.DataSet_x.ControllerStruct.PID_Controller.kp
		含义	比例增益
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
2082	1-16	DB 参数中的名称 (“正常”显示模式)	Position control.Static controller data.PID controller.Weighting factor
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfDataSets.DataSet_x.ControllerStruct.PID_Controller.kpc
		含义	预控制加权
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	150
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	
2083	1-16	DB 参数中的名称 (“正常”显示模式)	Position control.Static controller data.PID controller.ki
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfDataSets.DataSet_x.ControllerStruct.PID_Controller.ki
		含义	积分器的增益
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
2084	1-16	DB 参数中的名称 (“正常”显示模式)	Position control.Static controller data.PID controller.Activation of the precontrol
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfDataSets.DataSet_x.ControllerStruct.PID_Controller.preCon</i>
		含义	激活预控制功能
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		适用对象	定位轴、同步轴
		固件版本	
2085	1-16	DB 参数中的名称 (“正常”显示模式)	Position control.Static controller data.PID controller.kd
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfDataSets.DataSet_x.ControllerStruct.PID_Controller.kd</i>
		含义	微分增益
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
2086	1-16	DB 参数中的名称 (“正常”显示模式)	Position control.Static controller data.PID controller.Delay time
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfDataSets.DataSet_x.ControllerStruct.PID_Controller.DecayTime
		含义	DT1 元素的延迟时间
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	
2087	1-16	DB 参数中的名称 (“正常”显示模式)	Position control.Static controller data.PID controller.Integrator limitation activation
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfDataSets.DataSet_x.ControllerStruct.PID_Controller.EnableAntiWindUp
		含义	激活积分器限限制
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	重启
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
2088	1-16	DB 参数中的名称 (“正常”显示模式)	Position control.Static controller data.PID controller.Activation/mode balancing filter
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfDataSets.DataSet_x.ControllerStruct.PID_Controller.BalanceFilterMode
		含义	平衡滤波器设置
		参数类型	组态数据元素
		数据类型	DINT EnumBalanceFilterMode (页 1153)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

9.4.4 工艺参数 3000 至 3999 — 轴

利用本章提供的信息来读取或更改用户程序中工艺对象的参数。工艺对象的参数可以使用“MC_ReadSysParameter”工艺功能来读取，并使用“MC_WriteParameter”工艺功能来写入。

编辑工艺参数的辅助条件：

- 如果在参数中输入的新值等于旧值，则不需要在列表中指定的 **Restart**。此外，在更改虚拟轴的参数时，也不必执行 **Restart**。是否需要 **Restart** 在相应工艺 DB 的 *Statusword.RequestRestart* 变量中指明。
- 由参数编号和索引定义的参数必须在系统中可用并且允许写访问。必须使用工艺功能“MC_ChangeDataset”在 T-CPU 中启用由索引定义的数据记录。

关键字

编号	“MC_ReadSysParameter”和“MC_WriteParameter”工艺功能的输入参数 <i>ParameterNumber</i> 的参数编号
索引	索引可用于多次访问现有参数。以下区域中存在多个参数： • 数据集（Dataset_1 到 Dataset_16） • 编码器（Encoder_1 到 Encoder_8） • 同步对象（1 代表同步对象，2 代表叠加的同步对象） 可以从固件版本为 V3.1.x 和更高版本的集成工艺使用索引。
DB 参数中的名称 （“正常”显示）	如果在 STEP 7 中打开“MC_ReadSysParameter”或“MC_WriteParameter”工艺功能的实例数据块，则在“DB 参数”中会显示一个组态对话框。 在“正常”显示模式下显示参数名称
DB 参数中的名称 （“专家”显示）	如果在 STEP 7 中打开“MC_ReadSysParameter”或“MC_WriteParameter”工艺功能的实例数据块，则在“DB 参数”中会显示一个组态对话框。 在“专家”显示模式下显示参数名称 此名称对应于系统变量的名称或 S7T Config 的“专家”列表中组态数据元素的名称。
说明	参数说明
参数类型	指的是 S7T Config 的“专家”列表中的参数类型“系统变量”或“组态数据”。
数据类型	指定参数的数据类型。 可以在“DINT 值的列表”中找到“DINT Enum”数据类型的可用值。

9.4 工艺参数

最小值/最大值	指定参数的上限值和下限值。
访问	Read: 可以读取参数。 Write: 可以更改参数。
活动	立即: 在参数中所做的更改立即生效。 Restart: 参数中的更改需要重启才能生效, 意味着重新初始化工艺对象时生效。这种情况仅在工艺对象 TO 被禁用的情况下才可能发生。
适用对象	对于轴参数, 指定该参数适用的轴类型。
固件版本	可从中使用参数的集成工艺的固件版本。

编号	索引	参数说明	
3000	-	DB 参数中的名称 (“正常”显示模式)	Mechanics.Measuring system.Encoder count
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.numberOfEncoders
		说明	测量系统的数量
		参数类型	组态数据元素
		数据类型	DINT
		最小值	0
		最大值	8
		访问	Read
		活动	-
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
3010	1-8	DB 参数中的名称 (“正常”显示模式)	Mechanics.Measuring system.Encoder attachment type
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfEncoders.Encoder_x.AssemblyBase.assemblyBase</i>
		说明	编码器的连接类型
		参数类型	组态数据元素
		数据类型	DINT EnumAxisEncoderAssemblyType (页 1149)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
3011	1-8	DB 参数中的名称 (“正常”显示模式)	Mechanics.Measuring system.Motor side measuring gearbox.Encoder revolutions count
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfEncoders.Encoder_x.AdaptDrive.denFactor</i>
		说明	分母
		参数类型	组态数据元素
		数据类型	DINT
		最小值	1
		最大值	2147483647
		访问	Read, Write
		活动	Restart
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
3012	1-8	DB 参数中的名称 (“正常”显示模式)	Mechanics.Measuring system.Motor side measuring gearbox.Motor revolutions count
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_x.AdaptDrive.num Factor
		说明	计数器
		参数类型	组态数据元素
		数据类型	DINT
		最小值	1
		最大值	2147483647
		访问	Read, Write
		活动	Restart
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
3013	1-8	DB 参数中的名称 (“正常”显示模式)	Mechanics.Measuring system.Load side measuring gearbox.Encoder revolutions count
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_x.AdaptExtern.den Factor
		说明	分母
		参数类型	组态数据元素
		数据类型	DINT
		最小值	1
		最大值	2147483647
		访问	Read, Write
		活动	Restart
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
3014	1-8	DB 参数中的名称 (“正常”显示模式)	Mechanics.Measuring system.Load side measuring gearbox.Load revolutions count
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_x.AdaptExtern.numFactor
		说明	计数器
		参数类型	组态数据元素
		数据类型	DINT
		最小值	1
		最大值	2147483647
		访问	Read, Write
		活动	Restart
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
3015	1-8	DB 参数中的名称 (“正常”显示模式)	Mechanics.Measuring system.External measuring gearbox.Path per measuring wheel rotation
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_x.PathPerRevolution.length
		说明	每个测量轮旋转的路径
		参数类型	组态数据元素
		数据类型	REAL
		最小值	> 0.0
		最大值	1E+12
		访问	Read, Write
		活动	Restart
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
3016	1-8	DB 参数中的名称 (“正常”显示模式)	Mechanics.Measuring system.External measuring gearbox.Encoder revolutions count
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_x.AdaptLoad.denFactor
		说明	分母
		参数类型	组态数据元素
		数据类型	DINT
		最小值	1
		最大值	2147483647
		访问	Read, Write
		活动	Restart
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
3017	1-8	DB 参数中的名称 (“正常”显示模式)	Mechanics.Measuring system.External measuring gearbox.Measuring wheel revolutions count
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_x.AdaptLoad.numFactor
		说明	计数器
		参数类型	组态数据元素
		数据类型	DINT
		最小值	1
		最大值	2147483647
		访问	Read, Write
		活动	Restart
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
3021	1-8	DB 参数中的名称 (“正常”显示模式)	Mechanics.Backlash on reversal compensation.Absolute encoder.Activation
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_x.AbsBackLash.enable
		说明	间隙补偿的激活状态
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	
3022	1-8	DB 参数中的名称 (“正常”显示模式)	Mechanics.Backlash on reversal compensation.Absolute encoder.Direction
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_x.AbsBackLash.type
		说明	间隙补偿的方向
		参数类型	组态数据元素
		数据类型	DINT EnumBackLashType (页 1152)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
3023	1-8	DB 参数中的名称 (“正常”显示模式)	Mechanics.Backlash on reversal compensation.Absolute encoder.Preferred position
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_x.AbsBackLash.startUpDifference
		说明	首选位置
		参数类型	组态数据元素
		数据类型	DINT EnumBackLashDiff (页 1152)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	
3024	1-8	DB 参数中的名称 (“正常”显示模式)	Mechanics.Backlash on reversal compensation.Absolute encoder.Value
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_x.AbsBackLash.length
		说明	值
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
3025	1-8	DB 参数中的名称 (“正常”显示模式)	Mechanics.Backlash on reversal compensation.Absolute encoder.Velocity
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_x.AbsBackLash.velocity
		说明	间隙补偿速度
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	
3031	1-8	DB 参数中的名称 (“正常”显示模式)	Homing.Offset
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_x.AbsHomingEncoder.absShift
		说明	绝对编码器的偏移量
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
3041	1-8	DB 参数中的名称 (“正常”显示模式)	Configuration.Measuring system.Filter.Activation
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_x.Filter.enable
		说明	过滤器功能的激活状态
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
3042	1-8	DB 参数中的名称 (“正常”显示模式)	Configuration.Measuring system.Filter.Time constant
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_x.Filter.timeConstant
		说明	用于 PT1 平滑 (实际值平滑) 的时间常量
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
3043	1-8	DB 参数中的名称 (“正常”显示模式)	Configuration.Measuring system.Encoder count direction
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_X.InversCountDirection. encoderFeedbackPolarity
		说明	激活计数方向反转
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		适用对象	定位轴、同步轴
		固件版本	
3051	1-8	DB 参数中的名称 (“正常”显示模式)	Mechanics.Backlash on reversal compensation.Incremental encoder.Direction
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_x.IncBackLash_type
		说明	间隙补偿的有效方向
		参数类型	组态数据元素
		数据类型	DINT EnumBackLashType (页 1152)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
3052	1-8	DB 参数中的名称 (“正常”显示模式)	Mechanics.Backlash on reversal compensation.Incremental encoder.Activation
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_x.IncBackLash.enable
		说明	间隙补偿的激活状态
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	
3053	1-8	DB 参数中的名称 (“正常”显示模式)	Mechanics.Backlash on reversal compensation.Incremental encoder.Value
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_x.IncBackLash.length
		说明	间隙值
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
3054	1-8	DB 参数中的名称 (“正常”显示模式)	Mechanics.Backlash on reversal compensation.Incremental encoder.Velocity
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_x.IncBackLash.velocity
		说明	间隙补偿对应的运行速度
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	
3061	1-8	DB 参数中的名称 (“正常”显示模式)	Homing.Active homing.Approach direction
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_x.IncHomingEncoder.approachDirection
		说明	参考逼近方向
		参数类型	组态数据元素
		数据类型	DINT EnumAxisApproachDirection (页 1148)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
3063	1-8	DB 参数中的名称 (“正常”显示模式)	Homing.Zero mark activation monitoring
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfEncoders.Encoder_x.IncHomingEncoder.enableZeroMarkDistance</i>
		说明	激活编码器零标记监视
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
3064	1-8	DB 参数中的名称 (“正常”显示模式)	Homing.Active homing.Homing mode
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfEncoders.Encoder_x.IncHomingEncoder.homingMode</i>
		说明	回原点模式 — 主动回原点
		参数类型	组态数据元素
		数据类型	DINT EnumAxisHomingMode (页 1150)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
3066	1-8	DB 参数中的名称 (“正常”显示模式)	Homing.Passive homing.Approach direction
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfEncoders.Encoder_x.IncHomingEncoder.passiveApproachDirection</i>
		说明	期望逼近方向
		参数类型	组态数据元素
		数据类型	DINT EnumAxisPassiveApproachDirection (页 1151)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
3068	1-8	DB 参数中的名称 (“正常”显示模式)	Homing.Passive homing.Homing mode
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfEncoders.Encoder_x.IncHomingEncoder.passiveHomingMode</i>
		说明	回原点模式 — 被动回原点
		参数类型	组态数据元素
		数据类型	DINT EnumAxisPassiveHomingMode (页 1151)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
3070	1-8	DB 参数中的名称 (“正常”显示模式)	Homing.Active homing.Reference point offset
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfEncoders.Encoder_x.IncHomingEncoder.proceedShiftPos</i>
		说明	增量编码器中的参考点偏移
		参数类型	组态数据元素
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
3071	1-8	DB 参数中的名称 (“正常”显示模式)	Homing.Distance to zero mark
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfEncoders.Encoder_x.IncHomingEncoder.zeroMarkDistance</i>
		说明	回原点输出凸轮与编码器零标记之间的最大距离
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
3072	1-8	DB 参数中的名称 (“正常”显示模式)	Actual values.Homing.Actual value change
		DB 参数中的名称 (“专家”显示模式)	<i>homingcommand.positiondifference</i>
		说明	上次回原点期间的实际值更改
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read
		活动	-
		适用对象	定位轴、同步轴
		固件版本	
3073	1-8	DB 参数中的名称 (“正常”显示模式)	Configuration.Measuring system.Analog encoder.Activate position filter
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfEncoders.Encoder_X.AnalogSensor.PositionFilter.enable</i>
		说明	激活过滤器
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
3074	1-8	DB 参数中的名称 (“正常”显示模式)	Configuration.Measuring system.Analog encoder.Time constant position filter
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_X.AnalogSensor.PositionFilter.timeConstant
		说明	用于 PT1 平滑的时间常量
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
3075	1-8	DB 参数中的名称 (“正常”显示模式)	Configuration.Measuring system.Analog encoder.Weighting factor
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_X.AnalogSensor.ConversionData.factor
		说明	模拟测量值的加权系数
		参数类型	组态数据元素
		数据类型	REAL
		最小值	> 0.0
		最大值	1E+12
		访问	Read, Write
		活动	Restart
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
3076	1-8	DB 参数中的名称 (“正常”显示模式)	Configuration.Measuring system.Analog encoder.Offset
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_X.AnalogSensor.ConversionData.offset
		说明	偏移量
		参数类型	组态数据元素
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

9.4.5 工艺参数 4000 至 4999 — 轴

利用本章提供的信息来读取或更改用户程序中工艺对象的参数。工艺对象的参数可以使用“MC_ReadSysParameter”工艺功能来读取，并使用“MC_WriteParameter”工艺功能来写入。

编辑工艺参数的辅助条件：

- 如果在参数中输入的新值等于旧值，则不需要在列表中指定的 **Restart**。此外，在更改虚拟轴的参数时，也不必执行 **Restart**。是否需要 **Restart** 在相应工艺 DB 的 *Statusword.RequestRestart* 变量中指明。
- 由参数编号和索引定义的参数必须在系统中可用并且允许写访问。必须使用工艺功能“MC_ChangeDataset”在 T-CPU 中启用由索引定义的数据记录。

关键字

编号	“MC_ReadSysParameter”和“MC_WriteParameter”工艺功能的输入参数 “ParameterNumber”的参数编号
索引	索引可用于多次访问现有参数。以下区域中存在多个参数： - 数据集（Dataset_1 到 Dataset_16） - 编码器（Encoder_1 到 Encoder_8） - 同步对象（1 代表同步对象，2 代表叠加的同步对象） 可以从固件版本为 V3.1.x 和更高版本的集成工艺使用索引。
DB 参数中的名称 （“正常”显示）	如果在 STEP 7 中打开“MC_ReadSysParameter”或“MC_WriteParameter”工艺功能的实例数据块，则在“DB 参数”中会显示一个组态对话框。 在“正常”显示模式下显示参数名称
DB 参数中的名称 （“专家”显示）	如果在 STEP 7 中打开“MC_ReadSysParameter”或“MC_WriteParameter”工艺功能的实例数据块，则在“DB 参数”中会显示一个组态对话框。 在“专家”显示模式下显示参数名称 此名称对应于系统变量的名称或 S7T Config 的“专家”列表中组态数据元素的名称。
含义	参数说明
参数类型	指的是 S7T Config 的“专家”列表中的参数类型“系统变量”或“组态数据”。
数据类型	指定参数的数据类型。 可以在“DINT 值的列表”中找到“DINT Enum”数据类型的可用值。
最小值/最大值	指定参数的上限值和下限值。
访问	Read: 可以读取参数。 Write: 可以更改参数。
活动	立即: 在参数中所做的更改立即生效。 Restart: 参数中的更改需要重启才能生效，意味着重新初始化工艺对象时生效。这种情况仅在工艺对象 TO 被禁用的情况下才可能发生。
适用对象	对于轴参数，指定该参数适用的轴类型。
固件版本	可从中使用参数的集成工艺的固件版本。

编号	索引	参数说明	
4001	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Accept changes
		DB 参数中的名称 (“专家”显示模式)	<i>activationmodechanged configdata</i>
		含义	激活已修改的组态数据
		参数类型	系统变量
		数据类型	DINT EnumToActivationModeSetConfigData (页 1171)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
4010	1-8	DB 参数中的名称 (“正常”显示模式)	Homing.Absolute value encoder adjustment.Total offset
		DB 参数中的名称 (“专家”显示模式)	<i>absoluteencoder_x.totaloffsetvalue</i>
		含义	绝对值编码器的测量系统状态，计算的总偏移
		参数类型	系统变量
		数据类型	2x DWORD
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
4011	1-8	DB 参数中的名称 (“正常”显示模式)	Homing.Absolute value encoder adjustment.Measuring system status
		DB 参数中的名称 (“专家”显示模式)	<i>absoluteencoder_x.activationstate</i>
		含义	绝对值编码器的测量系统状态，包括已计算的总偏移
		参数类型	系统变量
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read
		活动	-
		适用对象	定位轴、同步轴
		固件版本	
4020	1-8	DB 参数中的名称 (“正常”显示模式)	Actual values.Data block switching.Active data block
		DB 参数中的名称 (“专家”显示模式)	<i>datasetmonitoring.actualdataset</i>
		含义	当前活动的数据记录号
		参数类型	系统变量
		数据类型	DINT
		最小值	0
		最大值	16
		访问	Read
		活动	-
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
4030	1-8	DB 参数中的名称 (“正常”显示模式)	Actual values.Motion.Overlaid movement.Position
		DB 参数中的名称 (“专家”显示模式)	<i>superimposedmotion.position</i>
		含义	叠加坐标系中的位置
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read
		活动	-
		适用对象	定位轴、同步轴
		固件版本	
4031	1-8	DB 参数中的名称 (“正常”显示模式)	Actual values.Motion.Overlaid movement.Velocity
		DB 参数中的名称 (“专家”显示模式)	<i>superimposedmotion.velocity</i>
		含义	叠加坐标系中的速度
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read
		活动	-
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
4040	1-8	DB 参数中的名称 (“正常”显示模式)	Actual values.Motion.Base motion.Position
		DB 参数中的名称 (“专家”显示模式)	<i>basicmotion.position</i>
		含义	基坐标系中的位置
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read
		活动	-
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
4041	1-8	DB 参数中的名称 (“正常”显示模式)	Actual values.Motion.Base motion.Velocity
		DB 参数中的名称 (“专家”显示模式)	<i>basicmotion.velocity</i>
		含义	基坐标系中的速度
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read
		活动	-
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
4050	1-8	DB 参数中的名称 (“正常”显示模式)	Actual values.Sensor.Status
		DB 参数中的名称 (“专家”显示模式)	<i>sensordata_x.state</i>
		含义	测量系统状态
		参数类型	系统变量
		数据类型	DINT EnumSensorState (页 1166)
		最小值	-
		最大值	-
		访问	Read
		活动	-
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
4121	-	DB 参数中的名称 (“正常”显示模式)	Limits.Dynamic response.Negative direction of travel.Negative acceleration
		DB 参数中的名称 (“专家”显示模式)	<i>minuslimitsofdynamics.negativeaccel</i>
		含义	负加速度/减速度的限值
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
4122	-	DB 参数中的名称 (“正常”显示模式)	Limits.Dynamic response.Negative direction of travel.Negative jerk
		DB 参数中的名称 (“专家”显示模式)	<i>minuslimitsofdynamics.negativeacceljerk</i>
		含义	加速结束和减速开始时的冲击限值
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
4123	-	DB 参数中的名称 (“正常”显示模式)	Limits.Dynamic response.Negative direction of travel.Positive acceleration
		DB 参数中的名称 (“专家”显示模式)	<i>minuslimitsofdynamics.positiveaccel</i>
		含义	正加速度/减速度的限值
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
4124	-	DB 参数中的名称 (“正常”显示模式)	Limits.Dynamic response.Negative direction of travel.Positive jerk
		DB 参数中的名称 (“专家”显示模式)	minuslimitsofdynamics.positiveacceljerk
		含义	加速开始和减速结束时的冲击限值
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
4125	-	DB 参数中的名称 (“正常”显示模式)	Limits.Position and velocity.Negative direction of travel.Velocity
		DB 参数中的名称 (“专家”显示模式)	minuslimitsofdynamics.velocity
		含义	速度限制
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
4141	-	DB 参数中的名称 (“正常”显示模式)	Setpoints.Override.Acceleration
		DB 参数中的名称 (“专家”显示模式)	<i>override.acceleration</i>
		含义	百分比形式的加速倍率
		参数类型	系统变量
		数据类型	REAL
		最小值	1.0
		最大值	1000.0
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
4142	-	DB 参数中的名称 (“正常”显示模式)	Setpoints.Override.Velocity
		DB 参数中的名称 (“专家”显示模式)	<i>override.velocity</i>
		含义	百分比形式的速度倍率
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	200.0
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
4151	-	DB 参数中的名称 (“正常”显示模式)	Limits.Dynamic response.Positive direction of travel.Negative acceleration
		DB 参数中的名称 (“专家”显示模式)	<i>pluslimitsofdynamics.negativeaccel</i>
		含义	负加速度/减速度的限值
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
4152	-	DB 参数中的名称 (“正常”显示模式)	Limits.Dynamic response.Negative direction of travel.Negative jerk
		DB 参数中的名称 (“专家”显示模式)	<i>pluslimitsofdynamics.negativeacceljerk</i>
		含义	加速结束和减速开始时的冲击限值
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
4153	-	DB 参数中的名称 (“正常”显示模式)	Limits.Dynamic response.Negative direction of travel.Positive acceleration
		DB 参数中的名称 (“专家”显示模式)	<i>pluslimitsofdynamics.positiveaccel</i>
		含义	正加速度/减速度的限值
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
4154	-	DB 参数中的名称 (“正常”显示模式)	Limits.Dynamic response.Negative direction of travel.Positive jerk
		DB 参数中的名称 (“专家”显示模式)	<i>pluslimitsofdynamics.positiveacceljerk</i>
		含义	加速开始和减速结束时的冲击限值
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
4155	-	DB 参数中的名称 (“正常”显示模式)	Limits.Position and velocity.Positive direction of travel.Velocity
		DB 参数中的名称 (“专家”显示模式)	<i>pluslimitsofdynamics.velocity</i>
		含义	速度限制
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
4162	-	DB 参数中的名称 (“正常”显示模式)	Actual values.Commands.Positioning.Distance to go
		DB 参数中的名称 (“专家”显示模式)	<i>poscommand.distancetogo</i>
		含义	当前距目标的剩余距离
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
4173	-	DB 参数中的名称 (“正常”显示模式)	Actual values.Homing.Reference position coordinates
		DB 参数中的名称 (“专家”显示模式)	<i>positioningstate.homeposition</i>
		含义	参考点坐标
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read
		活动	-
		适用对象	定位轴、同步轴
		固件版本	
4174	-	DB 参数中的名称 (“正常”显示模式)	Actual values.Homing.Bero status
		DB 参数中的名称 (“专家”显示模式)	<i>homingcommand.berostate</i>
		含义	参考输出凸轮处于活动状态 (使用编码器零标记和参考输出凸轮的回原点)
		参数类型	系统变量
		数据类型	DINT EnumActiveInactive (页 1148)
		最小值	-
		最大值	-
		访问	Read
		活动	-
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
4180	-	DB 参数中的名称 (“正常”显示模式)	Homing.Reference position coordinates
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault homing.homeposition</i>
		含义	参考点坐标
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	
4181	-	DB 参数中的名称 (“正常”显示模式)	Homing.Approach speed
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault homing.homingapproachvelocity</i>
		含义	参考点逼近速度
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	10000.0
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
4182	-	DB 参数中的名称 (“正常”显示模式)	Homing.Creep speed
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault homing.homingentryvelocity</i>
		含义	参考点进入速度
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	2000.0
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	
4183	-	DB 参数中的名称 (“正常”显示模式)	Homing.Shutdown speed
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault homing.homingreducedvelocity</i>
		含义	参考点关闭速度
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1000.0
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
4190	-	DB 参数中的名称 (“正常”显示模式)	Position control.Static controller data.Offset compensation
		DB 参数中的名称 (“专家”显示模式)	<i>servosettings.setpointoffsetcompensation</i>
		含义	模拟接口的偏移量补偿
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
4231	-	DB 参数中的名称 (“正常”显示模式)	Limits.Fixed stop.Clamping value
		DB 参数中的名称 (“专家”显示模式)	<i>userdefaultclamping.clampingvalue</i>
		含义	钳位值 (例如到固定挡块的行程)
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
4241	-	DB 参数中的名称 (“正常”显示模式)	Default.Default value.Dynamics.Direction
		DB 参数中的名称 (“专家”显示模式)	<i>userdefaultdynamics.direction</i>
		含义	方向选择
		参数类型	系统变量
		数据类型	DINT EnumDirection (页 1157)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
4242	-	DB 参数中的名称 (“正常”显示模式)	Default.Default value.Dynamics.Negative acceleration
		DB 参数中的名称 (“专家”显示模式)	<i>userdefaultdynamics.negativeaccel</i>
		含义	负加速度/减速度
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
4243	-	DB 参数中的名称 (“正常”显示模式)	Default.Default value.Dynamics.Jerk at end of deceleration
		DB 参数中的名称 (“专家”显示模式)	<i>userdefaultdynamics.negativeaccelendjerk</i>
		含义	减速结束时的冲击限值
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
4244	-	DB 参数中的名称 (“正常”显示模式)	Default.Default value.Dynamics.Jerk at start of deceleration
		DB 参数中的名称 (“专家”显示模式)	<i>userdefaultdynamics.negativeaccelstartjerk</i>
		含义	减速开始时的冲击限值
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
4245	-	DB 参数中的名称 (“正常”显示模式)	Default.Default value.Dynamics.Positive acceleration
		DB 参数中的名称 (“专家”显示模式)	<i>userdefaultdynamics.positiveaccel</i>
		含义	正加速度
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
4246	-	DB 参数中的名称 (“正常”显示模式)	Default.Default value.Dynamics.Jerk at end of acceleration
		DB 参数中的名称 (“专家”显示模式)	<i>userdefaultdynamics.positiveaccelendjerk</i>
		含义	加速结束时的冲击限值
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
4247	-	DB 参数中的名称 (“正常”显示模式)	Default.Default value.Dynamics.Jerk at start of acceleration
		DB 参数中的名称 (“专家”显示模式)	<i>userdefaultdynamics.positiveaccelstartjerk</i>
		含义	加速开始时的冲击限值
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
4248	-	DB 参数中的名称 (“正常”显示模式)	Default.Default value.Dynamics.Velocity profile
		DB 参数中的名称 (“专家”显示模式)	<i>userdefaultdynamics.profile</i>
		含义	速度曲线类型
		参数类型	系统变量
		数据类型	DINT EnumProfile (页 1165)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
4249	-	DB 参数中的名称 (“正常”显示模式)	Default.Default value.Dynamics.Stopping time
		DB 参数中的名称 (“专家”显示模式)	<i>userdefaultdynamics.stoptime</i>
		含义	紧急停止的指定时间
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
4250	-	DB 参数中的名称 (“正常”显示模式)	Default.Default value.Dynamics.Velocity
		DB 参数中的名称 (“专家”显示模式)	<i>userdefaultdynamics.velocity</i>
		含义	速度规范
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
4262	1-2	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Cam synchronization.Master offset
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault.cammingsettings.camstartpositionmaster</i>
		含义	具有“相对”主轴的凸轮起始位置的偏移
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	
4266	1-2	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Cam synchronization.Synchronization mode
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault.cammingsettings.synchronizingmode</i>
		含义	凸轮盘的同步条件
		参数类型	系统变量
		数据类型	DINT EnumSyncModeCamming (页 1167)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	

编号	索引	参数说明	
4267	1-2	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Cam synchronization.Desynchronization mode
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault.cammingsettings.syncoffmode</i>
		含义	凸轮盘去同步条件
		参数类型	系统变量
		数据类型	DINT EnumSyncOffModeCamming (页 1169)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	
4268	1-2	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Cam synchronization.Synchronization master position
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault.cammingsettings.syncpositionmaster</i>
		含义	同步的主位置
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	

编号	索引	参数说明	
4269	1-2	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Cam synchronization.Synchronization slave position
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault.cammingsettings.syncpositionsslave</i>
		含义	同步的从位置
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	
4270	1-2	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Cam synchronization.Slave synchronization direction
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault.cammingsettings.synchronizingDirection</i>
		含义	从同步的方向
		参数类型	系统变量
		数据类型	DINT EnumFollowingObjectSynchronizingDirection (页 1159)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	

编号	索引	参数说明	
4286	1-2	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Gear synchronization.Synchronization criteria
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault.gearingsettings.synchronizingmode</i>
		含义	齿轮传动同步条件
		参数类型	系统变量
		数据类型	DINT EnumSyncModeGearing (页 1168)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	
4287	1-2	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Gear synchronization.Desynchronization criteria
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault.gearingsettings.syncoffmode</i>
		含义	齿轮去同步条件
		参数类型	系统变量
		数据类型	DINT EnumSyncOffModeGearing (页 1169)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	

编号	索引	参数说明	
4288	1-2	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Gear synchronization.Master position
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault.gearingsettings.syncpositionmaster</i>
		含义	主轴的同步位置
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	
4289	1-2	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Gear synchronization.Slave position
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault.gearingsettings.syncpositionslave</i>
		含义	从轴的同步位置
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	

编号	索引	参数说明	
4291	1-2	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Gear synchronization.Slave synchronization direction
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault.gearingsettings.synchronizingDirection</i>
		含义	从同步的方向
		参数类型	系统变量
		数据类型	DINT EnumFollowingObjectSynchronizingDirection (页 1159)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	
4292	1-2	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Gear synchronization.Look ahead
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault.gearingsettings.synchronizewithlookahead</i>
		说明	相关主轴先行
		参数类型	系统变量
		数据类型	DINT EnumFollowingObjectSynchronizeWithLookAhead (页 1158)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	

编号	索引	参数说明	
4301	1-2	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Dynamics.Negative acceleration
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault.syncdynamics.negativeaccel</i>
		含义	负加速度
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	
4302	1-2	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Dynamics.Jerk at end of deceleration
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault.syncdynamics.negativeaccelendjerk</i>
		含义	减速结束时的冲击限值
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	

编号	索引	参数说明	
4303	1-2	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Dynamics.Jerk at start of deceleration
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault.syncdynamics.negativeaccelstartjerk</i>
		含义	减速开始时的冲击限值
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	
4304	1-2	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Dynamics.Positive acceleration
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault.syncdynamics.positiveaccel</i>
		含义	正加速度
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	

编号	索引	参数说明	
4305	1-2	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Dynamics.Jerk at end of acceleration
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault.syncdynamics.positiveaccelendjerk</i>
		含义	加速结束时的冲击限值
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	
4306	1-2	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Dynamics.Jerk at start of acceleration
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault.syncdynamics.positiveaccelstartjerk</i>
		含义	加速开始时的冲击限值
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	

编号	索引	参数说明	
4307	1-2	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Dynamics.Velocity
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault.syncdynamics.velocity</i>
		含义	速度
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	
4308	1-2	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Dynamics.Velocity profile
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault.syncdynamics.velocityprofile</i>
		含义	速度曲线的类型
		参数类型	系统变量
		数据类型	DINT EnumProfile (页 1165)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	

编号	索引	参数说明	
4321	1-2	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Synchronization.Desynchronization master position
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault.syncoffpositions.master</i>
		含义	主轴的去同步位置
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	
4322	1-2	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Synchronization.Desynchronization slave position
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault.syncoffpositions.slave</i>
		含义	从轴的去同步位置
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	

编号	索引	参数说明	
4331	1-2	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Synchronization.Synchronization path
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault.syncprofile.synclength</i>
		含义	引导轴特定同步曲线的同步长度
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	
4332	1-2	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Synchronization.Desynchronization path
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault.syncprofile.synccofflength</i>
		含义	引导轴特定同步曲线的去同步长度
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	

编号	索引	参数说明	
4333	1-2	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Synchronization.Desynchronization reference
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault.syncprofile.syncoffpositionreference</i>
		含义	去同步曲线的位置参考
		参数类型	系统变量
		数据类型	DINT EnumSyncOffPositionReference (页 1170)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	
4334	1-2	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Synchronization.Synchronization reference
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault.syncprofile.syncpositionreference</i>
		含义	同步曲线的位置参考
		参数类型	系统变量
		数据类型	DINT EnumSyncPositionReference (页 1170)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	

编号	索引	参数说明	
4335	1-2	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Synchronization.Profile reference
		DB 参数中的名称 (“专家”显示模式)	userdefault.syncprofile.syncprofilereference
		含义	同步曲线的参考变量
		参数类型	系统变量
		数据类型	DINT EnumSyncProfileReference (页 1171)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	
4336	-	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Absolut.Jerklimiting
		DB 参数中的名称 (“专家”显示模式)	<i>SyncinMotion.smoothAbsolute Synchronization</i>
		含义	绝对同步模式下同步期间的冲击限值
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	

编号	索引	参数说明	
4337	-	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Synchronization.Limiting
		DB 参数中的名称 (“专家”显示模式)	<i>SyncingMotion.synchronizingAdaption</i>
		含义	默认动态值的增加/限制
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	同步轴
		固件版本	
4338	-	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Actual values.Master position
		DB 参数中的名称 (“专家”显示模式)	<i>currentsyncposition.master</i>
		含义	主值
		参数类型	系统变量
		数据类型	REAL
		最小值	-3.402823E+38
		最大值	3.402823E+38
		访问	Read
		活动	-
		适用对象	同步轴
		固件版本	

编号	索引	参数说明	
4339	-	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Actual values.Slave position
		DB 参数中的名称 (“专家”显示模式)	<i>currentsyncposition.slave</i>
		含义	从值
		参数类型	系统变量
		数据类型	REAL
		最小值	-3.402823E+38
		最大值	3.402823E+38
		访问	Read
		活动	-
		适用对象	同步轴
		固件版本	
4350	1-2	DB 参数中的名称 (“正常”显示模式)	Actual values.Synchronous operation.Cam synchronization.Master offset
		DB 参数中的名称 (“专家”显示模式)	<i>cammingadjustments.master.offset</i>
		含义	主轴偏移
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read
		活动	-
		适用对象	同步轴
		固件版本	

编号	索引	参数说明	
4351	1-2	DB 参数中的名称 (“正常”显示模式)	Actual values.Synchronous operation.Cam synchronization.Master scale
		DB 参数中的名称 (“专家”显示模式)	<i>cammingadjustments.master.scale</i>
		含义	主轴缩放
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read
		活动	-
		适用对象	同步轴
		固件版本	
4360	1-2	DB 参数中的名称 (“正常”显示模式)	Actual values.Synchronous operation.Cam synchronization.Slave offset
		DB 参数中的名称 (“专家”显示模式)	<i>cammingadjustments.slave.offset</i>
		含义	从轴偏移
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read
		活动	-
		适用对象	同步轴
		固件版本	

编号	索引	参数说明	
4361	1-2	DB 参数中的名称 (“正常”显示模式)	Actual values.Synchronous operation.Cam synchronization.Slave scale
		DB 参数中的名称 (“专家”显示模式)	<i>cammingadjustments.slave.scale</i>
		含义	从轴缩放
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read
		活动	-
		适用对象	同步轴
		固件版本	
4370	1-2	DB 参数中的名称 (“正常”显示模式)	Actual values.Synchronous operation.Gear synchronization.Master offset
		DB 参数中的名称 (“专家”显示模式)	<i>gearingadjustments.master.offset</i>
		含义	主轴偏移
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read
		活动	-
		适用对象	同步轴
		固件版本	

编号	索引	参数说明	
4400	-	DB 参数中的名称 (“正常”显示模式)	Actual values.Limits.Software limit switch.SW minus limit switch status
		DB 参数中的名称 (“专家”显示模式)	<i>swlimitstate.swlimitswitchminus</i>
		含义	软件限位开关反向
		参数类型	系统变量
		数据类型	DINT EnumLimitExceededOk (页 1162)
		最小值	-
		最大值	-
		访问	Read
		活动	-
		适用对象	定位轴、同步轴
		固件版本	
4401	-	DB 参数中的名称 (“正常”显示模式)	Actual values.Limits.Software limit switch.SW plus limit switch status
		DB 参数中的名称 (“专家”显示模式)	<i>swlimitstate.swlimitswitchplus</i>
		含义	软件限位开关正向
		参数类型	系统变量
		数据类型	DINT EnumLimitExceededOk (页 1162)
		最小值	-
		最大值	-
		访问	Read
		活动	-
		适用对象	定位轴、同步轴
		固件版本	

编号	索引	参数说明	
4410	-	DB 参数中的名称 (“正常”显示模式)	Actual values.Limits.Hardware limit switch.HW minus limit switch status
		DB 参数中的名称 (“专家”显示模式)	<i>sensormonitoring.hwlimitswitchminus</i>
		含义	硬件反向限位开关
		参数类型	系统变量
		数据类型	DINT EnumLimitExceededOk (页 1162)
		最小值	-
		最大值	-
		访问	Read
		活动	-
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
4411	-	DB 参数中的名称 (“正常”显示模式)	Actual values.Limits.Hardware limit switch.HW plus limit switch status
		DB 参数中的名称 (“专家”显示模式)	<i>sensormonitoring.hwlimitswitchplus</i>
		含义	硬件正向限位开关
		参数类型	系统变量
		数据类型	DINT EnumLimitExceededOk (页 1162)
		最小值	-
		最大值	-
		访问	读取
		活动	-
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
4412	-	DB 参数中的名称 (“正常”显示模式)	Actual values.Measuring system.Active encoder
		DB 参数中的名称 (“专家”显示模式)	<i>sensormonitoring.actualsensors</i>
		含义	活动编码器/传感器的编号
		参数类型	系统变量
		数据类型	DINT
		最小值	1
		最大值	8
		访问	Read
		活动	-
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
4500	-	DB 参数中的名称 (“正常”显示模式)	Setpoints.Drive.Control words
		DB 参数中的名称 (“专家”显示模式)	<i>drivedata.stw</i>
		含义	控制字 1 和 2
		参数类型	系统变量
		数据类型	DWORD
		最小值	-
		最大值	-
		访问	Read, Write
		活动	-
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
4501	-	DB 参数中的名称 (“正常”显示模式)	Actual values.Drive.Status words
		DB 参数中的名称 (“专家”显示模式)	<i>drivedata.zsw</i>
		含义	状态字 1 和 2
		参数类型	系统变量
		数据类型	DWORD
		最小值	-
		最大值	-
		访问	Read
		活动	-
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
4502	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Drive.Torque.Resolution
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.SetPointDriverInfo.DriveData.torqueReductionGranularity</i>
		含义	扭矩减小的高精度
		参数类型	组态数据元素
		数据类型	DINT EnumAxisTorqueForceReductionGranularity (页 1152)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

9.4.6 工艺参数 5000 至 9999 — 轴

利用本章提供的信息来读取或更改用户程序中工艺对象的参数。工艺对象的参数可以使用“MC_ReadSysParameter”工艺功能来读取，并使用“MC_WriteParameter”工艺功能来写入。

编辑工艺参数的辅助条件：

- 如果在参数中输入的新值等于旧值，则不需要在列表中指定的 **Restart**。此外，在更改虚拟轴的参数时，也不必执行 **Restart**。是否需要 **Restart** 在相应工艺 DB 的 *Statusword.RequestRestart* 变量中指明。
- 由参数编号和索引定义的参数必须在系统中可用并且允许写访问。必须使用工艺功能“MC_ChangeDataset”在 T-CPU 中启用由索引定义的数据记录。

关键字

编号	“MC_ReadSysParameter”和“MC_WriteParameter”工艺功能的输入参数 “ParameterNumber”的参数编号
索引	索引可用于多次访问现有参数。以下区域中存在多个参数： • 数据集（Dataset_1 到 Dataset_16） • 编码器（Encoder_1 到 Encoder_8） • 同步对象（1 代表同步对象，2 代表叠加的同步对象） 可以从固件版本为 V3.1.x 和更高版本的集成工艺使用索引。
DB 参数中的名称 （“正常”显示）	如果在 STEP 7 中打开“MC_ReadSysParameter”或“MC_WriteParameter”工艺功能的实例数据块，则在“DB 参数”中会显示一个组态对话框。 在“正常”显示模式下显示参数名称
DB 参数中的名称 （“专家”显示）	如果在 STEP 7 中打开“MC_ReadSysParameter”或“MC_WriteParameter”工艺功能的实例数据块，则在“DB 参数”中会显示一个组态对话框。 在“专家”显示模式下显示参数名称 此名称对应于系统变量的名称或 S7T Config 的“专家”列表中组态数据元素的名称。
含义	参数说明
参数类型	指的是 S7T Config 的“专家”列表中的参数类型“系统变量”或“组态数据”。
数据类型	指定参数的数据类型。 可以在“DINT 值的列表”中找到“DINT Enum”数据类型的可用值。

最小值/最大值	指定参数的上限值和下限值。
访问	Read: 可以读取参数。 Write: 可以更改参数。
活动	立即: 在参数中所做的更改立即生效。 Restart: 参数中的更改需要重启才能生效, 意味着重新初始化工艺对象时生效。这种情况仅在工艺对象 TO 被禁用的情况下才可能发生。
适用对象	指定应用该参数的轴的类型。
固件版本	可从中使用参数的集成工艺的固件版本。

编号	索引	参数说明	
5000	-	DB 参数中的名称 (“正常”显示模式)	Hydraulic.Superposition.Setpoint
		DB 参数中的名称 (“专家”显示模式)	<i>servosettings.additionalqoutputvalue</i>
		含义	液压轴: 受控变量叠加
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
5001	-	DB 参数中的名称 (“正常”显示模式)	Hydraulic.Superposition.Activation
		DB 参数中的名称 (“专家”显示模式)	<i>servosettings.additionalqoutputvalueswitch</i>
		含义	液压轴：激活受控变量叠加
		参数类型	系统变量
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
5002	-	DB 参数中的名称 (“正常”显示模式)	Hydraulic.Setpoints.Qoutputvalue
		DB 参数中的名称 (“专家”显示模式)	<i>actordata.qoutputvalue</i>
		含义	液压轴：相对于 Q 输出的百分比形式的受控变量
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read
		活动	-
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
5003	-	DB 参数中的名称 (“正常”显示模式)	Hydraulic.Sliding friction compensation.Dynamic.Setpoint
		DB 参数中的名称 (“专家”显示模式)	<i>actordata.slidingfriction compensationvalue</i>
		含义	液压轴：与速度有关的滑动摩擦补偿值
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read
		活动	-
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
5004	-	DB 参数中的名称 (“正常”显示模式)	Hydraulic.Sliding friction compensation.Dynamic.Activation
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.SlidingFriction.enable</i>
		含义	液压轴：滑动摩擦补偿的激活
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
5005	-	DB 参数中的名称 (“正常”显示模式)	Hydraulic.Sliding friction compensation.Dynamic.Factor
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.SlidingFriction.FactorMotionControl
		含义	液压轴：按速度设定值的函数计算的滑动摩擦补偿的因子
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
5006	-	DB 参数中的名称 (“正常”显示模式)	Hydraulic.Sliding friction compensation.Static.Activation
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.AdditionalOffset.enable
		含义	液压轴：激活偏移注入
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
5007	-	DB 参数中的名称 (“正常”显示模式)	Hydraulic.Sliding friction compensation.Static.Negative direction
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.AdditionalOffset.offsetMotionControlNegative
		含义	液压轴：负运动方向的偏移
		参数类型	组态数据元素
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
5008	-	DB 参数中的名称 (“正常”显示模式)	Hydraulic.Sliding friction compensation.Static.Positive direction
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.AdditionalOffset.offsetMotionControlPositive
		含义	液压轴：正运动方向的偏移
		参数类型	组态数据元素
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
5009	-	DB 参数中的名称 (“正常”显示模式)	Hydraulic.Limits.Set value increasing
		DB 参数中的名称 (“专家”显示模式)	<i>userdefaultqfaxis.maxderivative.qoutput</i>
		含义	液压轴：受控变量的更改限值
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
5010	-	DB 参数中的名称 (“正常”显示模式)	Hydraulic.Sliding friction compensation.Static.Setpoint
		DB 参数中的名称 (“专家”显示模式)	<i>actordata.frictionadditionaloffsetvalue</i>
		含义	液压轴：与方向有关的滑动摩擦补偿值
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read
		活动	-
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
5011	-	DB 参数中的名称 (“正常”显示模式)	Position control.Static friction compensation.Activation
		DB 参数中的名称 (“专家”显示模式)	<i>actordata.frictioncompensation</i>
		含义	摩擦补偿激活的状态
		参数类型	系统变量
		数据类型	DINT EnumActiveInactive (页 1148)
		最小值	-
		最大值	-
		访问	Read
		活动	-
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
5012	-	DB 参数中的名称 (“正常”显示模式)	Position control.Static friction compensation.Setpoint
		DB 参数中的名称 (“专家”显示模式)	<i>actordata.frictioncompensationvalue</i>
		含义	摩擦补偿值
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read
		活动	-
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
5013	-	DB 参数中的名称 (“正常”显示模式)	Hydraulic.Limits.Set value limit exceeded
		DB 参数中的名称 (“专家”显示模式)	<i>actormonitoring.qoutputderivativelimitingstate</i>
		含义	液压轴：在 Q 输出处上升的限值
		参数类型	系统变量
		数据类型	DINT EnumLimitExceededOk (页 1162)
		最小值	-
		最大值	-
		访问	Read
		活动	-
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
5014	-	DB 参数中的名称 (“正常”显示模式)	Hydraulic.Valve active
		DB 参数中的名称 (“专家”显示模式)	<i>actormonitoring.qoutputstate</i>
		含义	液压轴：Q 输出激活
		参数类型	系统变量
		数据类型	DINT EnumActiveInactive (页 1148)
		最小值	-
		最大值	-
		访问	Read
		活动	-
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
5015	-	DB 参数中的名称 (“正常”显示模式)	Hydraulic.Dynamic controller data.Hydraulic time constant
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfDataSets.DynamicQFData.qOutputTimeConstant</i>
		含义	液压轴：液压控制回路的等效时间常量
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
5016	-	DB 参数中的名称 (“正常”显示模式)	Hydraulic.Dynamic controller data.Velocity control loop time constant
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfDataSets.DynamicQFData.velocityTimeConstant</i>
		含义	液压轴：速度控制回路的等效时间常量
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴
		固件版本	

编号	索引	参数说明	
5017	-	DB 参数中的名称 (“正常”显示模式)	Hydraulic.Dynamic controller data.Position control loop time constant
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfDataSets.DynamicQFData.positionTimeConstant</i>
		含义	液压轴：位置控制回路的等效时间常量
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
5018	-	DB 参数中的名称 (“正常”显示模式)	Hydraulic.Invert setpoint.In front of characteristic.Activation
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfDataSets.InvertQOutput.invSetPoint</i>
		含义	液压轴：受控变量叠加的激活
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
5019	-	DB 参数中的名称 (“正常”显示模式)	Hydraulic.Invert setpoint.After characteristic.Activation
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfDataSets.InvertSetPoint.Invert
		含义	液压轴: Q 输出反向的激活
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	
5020	1-8	DB 参数中的名称 (“正常”显示模式)	Actual values.Sensor.Velocity
		DB 参数中的名称 (“专家”显示模式)	sensordata.sensordata_x.velocity
		含义	实际速度值, 编码器值
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read
		活动	-
		适用对象	速度控制轴、定位轴、同步轴
		固件版本	

编号	索引	参数说明	
5065	1-8	DB 参数中的名称 (“正常”显示模式)	Configuration.Releases
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.DriveControllConfig.ReleaseDisableMode
		含义	要复位的位, ProfiDrive 协议
		参数类型	组态数据元素
		数据类型	DINT
		最小值	1
		最大值	127
		访问	Read, Write
		活动	立即
		适用对象	定位轴、同步轴
		固件版本	

9.4.7 工艺参数 — 外部编码器

利用本章提供的信息来读取或更改用户程序中工艺对象的参数。工艺对象的参数可以使用“MC_ReadSysParameter”工艺功能来读取，并使用“MC_WriteParameter”工艺功能来写入。

编辑工艺参数的辅助条件：

- 如果在参数中输入的新值等于旧值，则不需要在列表中指定的 **Restart**。此外，在更改虚拟轴的参数时，也不必执行 **Restart**。是否需要 **Restart** 在相应工艺 DB 的 *Statusword.RequestRestart* 变量中指明。
- 由参数编号和索引定义的参数必须在系统中可用并且允许写访问。必须使用工艺功能“MC_ChangeDataset”在 T-CPU 中启用由索引定义的数据记录。

关键字

编号	“MC_ReadSysParameter”和“MC_WriteParameter”工艺功能的输入参数 “ParameterNumber”的参数编号
索引	索引可用于多次访问现有参数。以下区域中存在多个参数： - 数据集（Dataset_1 到 Dataset_16） - 编码器（Encoder_1 到 Encoder_8） - 同步对象（1 代表同步对象，2 代表叠加的同步对象） 可以从固件版本为 V3.1.x 和更高版本的集成工艺使用索引。
DB 参数中的名称 （“正常”显示）	如果在 STEP 7 中打开“MC_ReadSysParameter”或“MC_WriteParameter”工艺功能的实例数据块，则在“DB 参数”中会显示一个组态对话框。 在“正常”显示模式下显示参数名称
DB 参数中的名称 （“专家”显示）	如果在 STEP 7 中打开“MC_ReadSysParameter”或“MC_WriteParameter”工艺功能的实例数据块，则在“DB 参数”中会显示一个组态对话框。 在“专家”显示模式下显示参数名称 此名称对应于系统变量的名称或 S7T Config 的“专家”列表中组态数据元素的名称。
说明	参数说明
参数类型	指的是 S7T Config 的“专家”列表中的参数类型“系统变量”或“组态数据”。
数据类型	指定参数的数据类型。 可以在“DINT 值的列表”中找到“DINT Enum”数据类型的可用值。
最小值/最大值	指定参数的上限值和下限值。
访问	Read: 可以读取参数。 Write: 可以更改参数。
活动	立即: 在参数中所做的更改立即生效。 Restart: 参数中的更改需要重启才能生效，意味着重新初始化工艺对象时生效。这种情况仅在工艺对象 TO 被禁用的情况下才可能发生。
固件版本	可从中使用参数的集成工艺的固件版本。

编号	索引	参数说明	
1100	-	DB 参数中的名称 (“正常”显示模式)	Mechanics.Spindle settings.Spindle pitch for each revolution of the axis
		DB 参数中的名称 (“专家”显示模式)	<i>LeadScrew.pitchVal</i>
		说明	轴每转的丝杠螺距
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	+1E12
		访问	Read, Write
		活动	Restart
		固件版本	
1110	-	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Extrapolator.Extrapolation time
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.Extrapolation.ExtrapolationTime</i>
		说明	主轴的推断时间
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	+1E12
		访问	Read, Write
		活动	立即
		固件版本	

编号	索引	参数说明	
1111	-	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Extrapolator.Filter.Mode
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.Extrapolation.Filter.Mode
		说明	实际值平滑的选择标准
		参数类型	组态数据元素
		数据类型	DINT EnumAxisFilterMode (页 1149)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		固件版本	
1112	-	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Extrapolator.Filter.Activation
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.Extrapolation.Filter.Enable
		说明	激活状态
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		固件版本	

编号	索引	参数说明	
1113	-	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Extrapolator.Filter.Time constant
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.Extrapolation.Filter.TimeConstant</i>
		说明	用于 PT1 平滑的时间常量
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		固件版本	
1114	-	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Extrapolator.Tolerance range.Activation
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.Extrapolation.ToleranceRange.Enable</i>
		说明	激活公差范围
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		固件版本	

编号	索引	参数说明	
1115	-	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Extrapolator.Tolerance range.Tolerance window
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.Extrapolation.ToleranceRange.Value
		说明	公差窗口
		参数类型	组态数据元素
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		固件版本	
1116	-	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Extrapolator.Velocity signal
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis Extrapolation.extrapolatedVelocitySwitch
		说明	选择速度主值
		参数类型	组态数据元素
		数据类型	DINT EnumAxisExtrapolationVelocitySwitch (页 1149)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		固件版本	

编号	索引	参数说明	
1130	-	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Extrapolator.Position filter.Activation
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.Extrapolation.ExtrapolationPositionFilter.enable
		说明	激活过滤器
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		固件版本	
1131	-	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Extrapolator.Position filter.T1
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.Extrapolation.ExtrapolationPositionFilter.T1
		说明	第一个时间常量
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		固件版本	

编号	索引	参数说明	
1132	-	DB 参数中的名称 (“正常”显示模式)	Synchronous operation.Extrapolator.Position filter.T2
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.Extrapolation.ExtrapolationPositionFilter.T2
		说明	第二个时间常量
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		固件版本	
1201	-	DB 参数中的名称 (“正常”显示模式)	Mechanics.Modulo.Activation
		DB 参数中的名称 (“专家”显示模式)	Modulo.state
		说明	激活模数设置
		参数类型	组态数据元素
		数据类型	DINT EnumActiveInactive (页 1148)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		固件版本	

编号	索引	参数说明	
1202	-	DB 参数中的名称 (“正常”显示模式)	Mechanics.Modulo.Length
		DB 参数中的名称 (“专家”显示模式)	<i>Modulo.length</i>
		说明	模数长度
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	Restart
		固件版本	
1203	-	DB 参数中的名称 (“正常”显示模式)	Mechanics.Modulo.Starting value
		DB 参数中的名称 (“专家”显示模式)	<i>Modulo.startValue</i>
		说明	模数起始值
		参数类型	组态数据元素
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	Restart
		固件版本	

编号	索引	参数说明	
1221	-	DB 参数中的名称 (“正常”显示模式)	Monitoring.Velocity.Activation
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.ActualVelocityMonitoring.enable
		说明	激活实际速度监视
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		固件版本	
1222	-	DB 参数中的名称 (“正常”显示模式)	Monitoring.Velocity.Maximum value
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.ActualVelocityMonitoring.maximum
		说明	允许的最大实际速度
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		固件版本	

编号	索引	参数说明	
1331	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Actual value smoothing.Activation
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.SmoothingFilter.enable
		说明	激活过滤器功能
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		固件版本	
1332	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Actual value smoothing.Mode
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.SmoothingFilter.mode
		说明	用于实际值平滑的计算方法
		参数类型	组态数据元素
		数据类型	DINT EnumAxisFilterMode (页 1149)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		固件版本	

编号	索引	参数说明	
1333	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Actual value smoothing.Time constant
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.SmoothingFilter.timeConstant
		说明	用于 PT1 平滑的时间常量
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		固件版本	
1351	-	DB 参数中的名称 (“正常”显示模式)	Monitoring.Standstill signal.Message delay time
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.StandStillSignal.delayTimeToActivate
		说明	触发停止信号的延迟时间
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		固件版本	

编号	索引	参数说明	
1352	-	DB 参数中的名称 (“正常”显示模式)	Monitoring.Standstill signal.Degree of filtering
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.StandStillSignal.filterDegree
		说明	过滤等级
		参数类型	组态数据元素
		数据类型	DINT
		最小值	0
		最大值	15
		访问	Read, Write
		活动	立即
		固件版本	
1353	-	DB 参数中的名称 (“正常”显示模式)	Monitoring.Standstill signal.Filter frequency
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.StandStillSignal.filterFrequency
		说明	过滤器频率
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		固件版本	

编号	索引	参数说明	
1354	-	DB 参数中的名称 (“正常”显示模式)	Monitoring.Standstill signal.Velocity limit
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.StandStillSignal.maxVeloStandStill
		说明	停止信号的速度限值
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		固件版本	
1400	-	DB 参数中的名称 (“正常”显示模式)	Position control.Static controller data. Manipulated variable limit.Activation
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.SpeedLimitation.Enable
		说明	速度范围限制的激活状态
		参数类型	组态数据元素
		数据类型	DINT EnumAxisFilterMode (页 1149)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		固件版本	

编号	索引	参数说明	
1401	-	DB 参数中的名称 (“正常”显示模式)	Position control.Static controller data.Manipulated variable limit.Lower limit
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.SpeedLimitation.MinSpeed</i>
		说明	速度范围下限
		参数类型	组态数据元素
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		固件版本	
1402	-	DB 参数中的名称 (“正常”显示模式)	Position control.Static controller data.Manipulated variable limit.Upper limit
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.SpeedLimitation.MaxSpeed</i>
		说明	速度范围上限
		参数类型	组态数据元素
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		固件版本	

编号	索引	参数说明	
2020	-	DB 参数中的名称 (“正常”显示模式)	Mechanics.Load gearbox.Load revolutions count
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.Gear.denFactor
		说明	变速箱系数的分母
		参数类型	组态数据元素
		数据类型	DINT
		最小值	0
		最大值	2147483647
		访问	Read, Write
		活动	Restart
		固件版本	
2021	-	DB 参数中的名称 (“正常”显示模式)	Mechanics.Load gearbox.Motor revolutions count
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.Gear.numFactor
		说明	变速箱系数的分子
		参数类型	组态数据元素
		数据类型	DINT
		最小值	0
		最大值	2147483647
		访问	Read, Write
		活动	Restart
		固件版本	

编号	索引	参数说明	
3010	-	DB 参数中的名称 (“正常”显示模式)	Mechanics.Measuring system.Encoder attachment type
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_1.AssemblyBase. assemblyBase
		说明	编码器的连接类型
		参数类型	组态数据元素
		数据类型	DINT EnumAxisEncoderAssemblyType (页 1149)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		固件版本	
3011	-	DB 参数中的名称 (“正常”显示模式)	Mechanics.Measuring system.Motor side measuring gearbox.Encoder revolutions count
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_1.AdaptDrive.denF actor
		说明	分母
		参数类型	组态数据元素
		数据类型	DINT
		最小值	1
		最大值	2147483647
		访问	Read, Write
		活动	Restart
		固件版本	

编号	索引	参数说明	
3012	-	DB 参数中的名称 (“正常”显示模式)	Mechanics.Measuring system.Motor side measuring gearbox.Motor revolutions count
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_1.AdaptDrive.num Factor
		说明	计数器
		参数类型	组态数据元素
		数据类型	DINT
		最小值	1
		最大值	2147483647
		访问	Read, Write
		活动	Restart
		固件版本	
3013	-	DB 参数中的名称 (“正常”显示模式)	Mechanics.Measuring system.Load side measuring gearbox.Encoder revolutions count
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_1.AdaptExtern.den Factor
		说明	分母
		参数类型	组态数据元素
		数据类型	DINT
		最小值	1
		最大值	2147483647
		访问	Read, Write
		活动	Restart
		固件版本	

编号	索引	参数说明	
3014	-	DB 参数中的名称 (“正常”显示模式)	Mechanics.Measuring system.Load side measuring gearbox.Load revolutions count
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_1.AdaptExtern.numFactor
		说明	计数器
		参数类型	组态数据元素
		数据类型	DINT
		最小值	1
		最大值	2147483647
		访问	Read, Write
		活动	Restart
		固件版本	
3015	-	DB 参数中的名称 (“正常”显示模式)	Mechanics.Measuring system.External measuring gearbox.Path per measuring wheel rotation
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_1.PathPerRevolution. ength
		说明	每个测量轮旋转的路径
		参数类型	组态数据元素
		数据类型	REAL
		最小值	> 0.0
		最大值	1E+12
		访问	Read, Write
		活动	Restart
		固件版本	

编号	索引	参数说明	
3016	-	DB 参数中的名称 (“正常”显示模式)	Mechanics.Measuring system.External measuring gearbox.Encoder revolutions count
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_1.AdaptLoad.denFactor
		说明	分母
		参数类型	组态数据元素
		数据类型	DINT
		最小值	1
		最大值	2147483647
		访问	Read, Write
		活动	Restart
		固件版本	
3017	-	DB 参数中的名称 (“正常”显示模式)	Mechanics.Measuring system.External measuring gearbox.Measuring wheel revolutions count
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_1.AdaptLoad.numFactor
		说明	计数器
		参数类型	组态数据元素
		数据类型	DINT
		最小值	1
		最大值	2147483647
		访问	Read, Write
		活动	Restart
		固件版本	

编号	索引	参数说明	
3031	-	DB 参数中的名称 (“正常”显示模式)	Homing.Offset
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfEncoders.Encoder_1.AbsHomingEncoder.absShift</i>
		说明	绝对编码器的偏移量
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		固件版本	
3041	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Measuring system.Filter.Activation
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfEncoders.Encoder_1.Filter.enable</i>
		说明	过滤器功能的激活状态
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		固件版本	

编号	索引	参数说明	
3042	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Measuring system.Filter.Time constant
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_1.Filter.timeConstant
		说明	用于 PT1 平滑 (实际值平滑) 的时间常量
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		固件版本	
3043	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Measuring system.Encoder count direction
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_X.InversCountDirection. encoderFeedbackPolarity
		说明	激活计数方向反转
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		固件版本	

编号	索引	参数说明	
3063	-	DB 参数中的名称 (“正常”显示模式)	Homing.Zero mark activation monitoring
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfEncoders.Encoder_1.IncHomingEncoder.enableZeroMarkDistance</i>
		说明	激活编码器零标记监视
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		固件版本	
3066	-	DB 参数中的名称 (“正常”显示模式)	Homing.Passive homing.Approach direction
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfEncoders.Encoder_1.IncHomingEncoder.passiveApproachDirection</i>
		说明	期望逼近方向
		参数类型	组态数据元素
		数据类型	DINT EnumAxisPassiveApproachDirection (页 1151)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		固件版本	

编号	索引	参数说明	
3068	-	DB 参数中的名称 (“正常”显示模式)	Homing.Passive homing.Homing mode
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfEncoders.Encoder_1.IncHomingEncoder.passiveHomingMode</i>
		说明	回原点模式，被动回原点
		参数类型	组态数据元素
		数据类型	DINT EnumAxisPassiveHomingMode (页 1151)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		固件版本	
3071	-	DB 参数中的名称 (“正常”显示模式)	Homing.Distance to zero mark
		DB 参数中的名称 (“专家”显示模式)	<i>TypeOfAxis.NumberOfEncoders.Encoder_1.IncHomingEncoder.zeroMarkDistance</i>
		说明	BERO 和零标记之间的最大距离
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		固件版本	

编号	索引	参数说明	
3073	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Measuring system.Analog encoder.Activate position filter
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_1.AnalogSensor.PositionFilter.enable
		说明	激活过滤器
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		固件版本	
3074	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Measuring system.Analog encoder.Time constant position filter
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_1.AnalogSensor.PositionFilter.timeConstant
		说明	用于 PT1 平滑的时间常量
		参数类型	组态数据元素
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		固件版本	

编号	索引	参数说明	
3075	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Measuring system.Analog encoder.Weighting factor
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_1.AnalogSensor.ConversionData.factor
		说明	模拟测量值的加权系数
		参数类型	组态数据元素
		数据类型	REAL
		最小值	> 0.0
		最大值	1E+12
		访问	Read, Write
		活动	Restart
		固件版本	
3076	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Measuring system.Analog encoder.Offset
		DB 参数中的名称 (“专家”显示模式)	TypeOfAxis.NumberOfEncoders.Encoder_1.AnalogSensor.ConversionData.offset
		说明	偏移量
		参数类型	组态数据元素
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		固件版本	

编号	索引	参数说明	
4001	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Accept changes
		DB 参数中的名称 (“专家”显示模式)	<i>activationmodechanged.configdata</i>
		说明	激活已修改的组态数据
		参数类型	系统变量
		数据类型	DINT EnumToActivationModeSetConfigData (页 1171)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		固件版本	
4010	-	DB 参数中的名称 (“正常”显示模式)	Homing.Absolute value encoder adjustment.Total offset
		DB 参数中的名称 (“专家”显示模式)	<i>absoluteencoder.totaloffset value</i>
		说明	绝对值编码器的测量系统状态，计算的总偏移
		参数类型	系统变量
		数据类型	2x DWORD
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		固件版本	

编号	索引	参数说明	
4011	-	DB 参数中的名称 (“正常”显示模式)	Homing.Absolute value encoder adjustment.Measuring system status
		DB 参数中的名称 (“专家”显示模式)	<i>absoluteencoder.activationstate</i>
		说明	绝对值编码器的测量系统状态，包括已计算的总偏移
		参数类型	系统变量
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read
		活动	-
		固件版本	
4341	-	DB 参数中的名称 (“正常”显示模式)	Default.Default values.Synchronization position
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault.syncposition</i>
		说明	同步位置
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		固件版本	

编号	索引	参数说明	
4410	-	DB 参数中的名称 (“正常”显示模式)	Actual values.Limits.Hardware limit switch.HW minus limit switch status
		DB 参数中的名称 (“专家”显示模式)	<i>sensormonitoring.hwlimitswitchminus</i>
		说明	硬件反向限位开关
		参数类型	系统变量
		数据类型	DINT EnumLimitExceededOk (页 1162)
		最小值	-
		最大值	-
		访问	Read
		活动	-
		固件版本	
4411	-	DB 参数中的名称 (“正常”显示模式)	Actual values.Limits.Hardware limit switch.HW plus limit switch status
		DB 参数中的名称 (“专家”显示模式)	<i>sensormonitoring.hwlimitswitchplus</i>
		说明	硬件正向限位开关
		参数类型	系统变量
		数据类型	DINT EnumLimitExceededOk (页 1162)
		最小值	-
		最大值	-
		访问	Read
		活动	-
		固件版本	

9.4.8 工艺参数 — 输出凸轮

利用本章提供的信息来读取或更改用户程序中工艺对象的参数。工艺对象的参数可以使用“MC_ReadSysParameter”工艺功能来读取，并使用“MC_WriteParameter”工艺功能来写入。

编辑工艺参数的辅助条件：

- 如果在参数中输入的新值等于旧值，则不需要在列表中指定的 **Restart**。此外，在更改虚拟轴的参数时，也不必执行 **Restart**。是否需要 **Restart** 在相应工艺 DB 的 *Statusword.RequestRestart* 变量中指明。
- 由参数编号和索引定义的参数必须在系统中可用并且允许写访问。必须使用工艺功能“MC_ChangeDataset”在 T-CPU 中启用由索引定义的数据记录。

关键字

编号	“MC_ReadSysParameter”和“MC_WriteParameter”工艺功能的输入参数 “ParameterNumber”的参数编号
索引	索引可用于多次访问现有参数。以下区域中存在多个参数： • 数据集（Dataset_1 到 Dataset_16） • 编码器（Encoder_1 到 Encoder_8） • 同步对象（1 代表同步对象，2 代表叠加的同步对象） 可以从固件版本为 V3.1.x 和更高版本的集成工艺使用索引。
DB 参数中的名称 （“正常”显示）	如果在 STEP 7 中打开“MC_ReadSysParameter”或“MC_WriteParameter”工艺功能的实例数据块，则在“DB 参数”中会显示一个组态对话框。 在“正常”显示模式下显示参数名称
DB 参数中的名称 （“专家”显示）	如果在 STEP 7 中打开“MC_ReadSysParameter”或“MC_WriteParameter”工艺功能的实例数据块，则在“DB 参数”中会显示一个组态对话框。 在“专家”显示模式下显示参数名称 此名称对应于系统变量的名称或 S7T Config 的“专家”列表中组态数据元素的名称。
含义	参数说明
参数类型	指的是 S7T Config 的“专家”列表中的参数类型“系统变量”或“组态数据”。
数据类型	指定参数的数据类型。 可以在“DINT 值的列表”中找到“DINT Enum”数据类型的可用值。
最小值/最大值	指定参数的上限值和下限值。

访问	Read: 可以读取参数。 Write: 可以更改参数。
活动	立即: 在参数中所做的更改立即生效。 重启: 参数中的更改需要重启才能生效, 意味着重新初始化工艺对象时生效。 这种情况仅在工艺对象 TO 被禁用的情况下才可能发生。
固件版本	可从中使用参数的集成工艺的固件版本。

编号	索引	参数说明	
1361	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Activate cam output
		DB 参数中的名称 (“专家”显示模式)	LogAddress.enableOutput
		含义	输出凸轮输出的激活状态。无法更改高速输出凸轮硬件输出的状态。
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		固件版本	
1362	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Logic operation
		DB 参数中的名称 (“专家”显示模式)	LogAddress.logicOperation
		含义	输出凸轮信号的逻辑运算
		参数类型	组态数据元素
		数据类型	DINT EnumLogicOperation (页 1162)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		固件版本	

编号	索引	参数说明	
1371	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Cam type
		DB 参数中的名称 (“专家”显示模式)	<i>OcaType._type</i>
		说明	输出凸轮类型
		参数类型	组态数据元素
		数据类型	DINT EnumOutputCamType (页 1164)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		固件版本	
4001	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Accept changes
		DB 参数中的名称 (“专家”显示模式)	<i>activationmodechanged.configdata</i>
		说明	激活已修改的组态数据
		参数类型	系统变量
		数据类型	DINT EnumToActivationModeSetConfigData (页 1171)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		固件版本	

9.4.9 工艺参数 — 凸轮轨迹

利用本章提供的信息来读取或更改用户程序中工艺对象的参数。工艺对象的参数可以使用“MC_ReadSysParameter”工艺功能来读取，并使用“MC_WriteParameter”工艺功能来写入。

编辑工艺参数的辅助条件：

- 如果在参数中输入的新值等于旧值，则不需要在列表中指定的 **Restart**。此外，在更改虚拟轴的参数时，也不必执行 **Restart**。是否需要 **Restart** 在相应工艺 DB 的 *Statusword.RequestRestart* 变量中指明。
- 由参数编号和索引定义的参数必须在系统中可用并且允许写访问。必须使用工艺功能“MC_ChangeDataset”在 T-CPU 中启用由索引定义的数据记录。

关键字

编号	“MC_ReadSysParameter”和“MC_WriteParameter”工艺功能的输入参数 “ParameterNumber”的参数编号
索引	索引可用于多次访问现有参数。以下区域中存在多个参数： • 数据集（Dataset_1 到 Dataset_16） • 编码器（Encoder_1 到 Encoder_8） • 同步对象（1 代表同步对象，2 代表叠加的同步对象） 可以从固件版本为 V3.1.x 和更高版本的集成工艺使用索引。
DB 参数中的名称 （“正常”显示）	如果在 STEP 7 中打开“MC_ReadSysParameter”或“MC_WriteParameter”工艺功能的实例数据块，则在“DB 参数”中会显示一个组态对话框。 在“正常”显示模式下显示参数名称
DB 参数中的名称 （“专家”显示）	如果在 STEP 7 中打开“MC_ReadSysParameter”或“MC_WriteParameter”工艺功能的实例数据块，则在“DB 参数”中会显示一个组态对话框。 在“专家”显示模式下显示参数名称 此名称对应于系统变量的名称或 S7T Config 的“专家”列表中组态数据元素的名称。
说明	参数说明
参数类型	指的是 S7T Config 的“专家”列表中的参数类型“系统变量”或“组态数据”。
数据类型	指定参数的数据类型。 可以在“DINT 值的列表”中找到“DINT Enum”数据类型的可用值。

9.4 工艺参数

最小值/最大值	指定参数的上限值和下限值。
访问	Read: 可以读取参数。 Write: 可以更改参数。
活动	立即: 在参数中所做的更改立即生效。 Restart: 参数中的更改需要重启才能生效, 意味着重新初始化工艺对象时生效。这种情况仅在工艺对象 TO 被禁用的情况下才可能发生。
固件版本	可从中使用参数的集成工艺的固件版本。

编号	索引	参数说明	
1361	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Activate cam output
		DB 参数中的名称 (“专家”显示模式)	LogAddress.enableOutput
		含义	输出凸轮输出的激活状态。无法更改高速输出凸轮硬件输出的状态。
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		固件版本	

编号	索引	参数说明	
1371	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Cam type
		DB 参数中的名称 (“专家”显示模式)	<i>OcaType._type</i>
		说明	输出凸轮类型
		参数类型	组态数据元素
		数据类型	DINT EnumOutputCamType (页 1164)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		固件版本	
1372	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Cam type
		DB 参数中的名称 (“专家”显示模式)	OctType.camTrackType
		说明	输出凸轮类型
		参数类型	组态数据元素
		数据类型	DINT EnumCamTrackType (页 1156)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		固件版本	

编号	索引	参数说明	
4001	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Accept changes
		DB 参数中的名称 (“专家”显示模式)	<i>activationmodechanged.configdata</i>
		说明	激活已修改的组态数据
		参数类型	系统变量
		数据类型	DINT EnumToActivationModeSetConfigData (页 1171)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		固件版本	
5050	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Inversion
		DB 参数中的名称 (“专家”显示模式)	<i>OctTechnologicalCfg.InvertOutput</i>
		含义	凸轮轨迹输出的反向
		参数类型	组态数据元素
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	Restart
		固件版本	

编号	索引	参数说明	
5052	-	DB 参数中的名称 (“正常”显示模式)	Cam track.Output cam activation.Bit pattern
		DB 参数中的名称 (“专家”显示模式)	enablevalidcam
		含义	输出凸轮轨迹上各个输出凸轮的快速抑制/启用
		参数类型	系统变量
		数据类型	DWORD
		最小值	0
		最大值	FFFF FFFF
		访问	Read, Write
		活动	立即
		固件版本	
5060	-	DB 参数中的名称 (“正常”显示模式)	Cam track.Individual output cam.Starting position
		DB 参数中的名称 (“专家”显示模式)	Userdefault.singleCamsettings.cam.cam[1..32].startposition
		含义	输出凸轮起始位置
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		固件版本	

编号	索引	参数说明	
5061	-	DB 参数中的名称 (“正常”显示模式)	Cam track.Individual output cam.End position
		DB 参数中的名称 (“专家”显示模式)	<i>Userdefault.singleCamsettings.cam.cam[1..32].Endposition</i>
		含义	输出凸轮结束位置
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read, Write
		活动	立即
		固件版本	
5062	-	DB 参数中的名称 (“正常”显示模式)	Cam track.Individual output cam.ON length
		DB 参数中的名称 (“专家”显示模式)	<i>Userdefault.singleCamsettings.cam.cam[1..32].maxLength</i>
		含义	基于时间的凸轮最大激活长度
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		固件版本	

编号	索引	参数说明	
5063	-	DB 参数中的名称 (“正常”显示模式)	Cam track.Individual output cam.ON duration
		DB 参数中的名称 (“专家”显示模式)	<i>Userdefault.singleCamsettings.cam.cam[1..32].Onduration</i>
		含义	基于时间的输出凸轮的 ON 持续时间
		参数类型	系统变量
		数据类型	REAL
		最小值	0.0
		最大值	1E+12
		访问	Read, Write
		活动	立即
		固件版本	
5064	-	DB 参数中的名称 (“正常”显示模式)	Cam track.Individual output cam.Validity
		DB 参数中的名称 (“专家”显示模式)	<i>Userdefault.singleCamsettings.cam.cam[1..32].Validity</i>
		含义	有效性
		参数类型	系统变量
		数据类型	DINT EnumYesNo (页 1172)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		固件版本	

9.4.10 工艺参数 — 凸轮盘

利用本章提供的信息来读取或更改用户程序中工艺对象的参数。工艺对象的参数可以使用“MC_ReadSysParameter”工艺功能来读取，并使用“MC_WriteParameter”工艺功能来写入。

编辑工艺参数的辅助条件：

- 如果在参数中输入的新值等于旧值，则不需要在列表中指定的 **Restart**。此外，在更改虚拟轴的参数时，也不必执行 **Restart**。是否需要 **Restart** 在相应工艺 DB 的 *Statusword.RequestRestart* 变量中指明。
- 由参数编号和索引定义的参数必须在系统中可用并且允许写访问。必须使用工艺功能“MC_ChangeDataset”在 T-CPU 中启用由索引定义的数据记录。

关键字

编号	“MC_ReadSysParameter”和“MC_WriteParameter”工艺功能的输入参数 <i>ParameterNumber</i> 的参数编号
索引	索引可用于多次访问现有参数。以下区域中存在多个参数： • 数据集（Dataset_1 到 Dataset_16） • 编码器（Encoder_1 到 Encoder_8） • 同步对象（1 代表同步对象，2 代表叠加的同步对象） 可以从固件版本为 V3.1.x 和更高版本的集成工艺使用索引。
DB 参数中的名称 （“正常”显示）	如果在 STEP 7 中打开“MC_ReadSysParameter”或“MC_WriteParameter”工艺功能的实例数据块，则在“DB 参数”中会显示一个组态对话框。 在“正常”显示模式下显示参数名称
DB 参数中的名称 （“专家”显示）	如果在 STEP 7 中打开“MC_ReadSysParameter”或“MC_WriteParameter”工艺功能的实例数据块，则在“DB 参数”中会显示一个组态对话框。 在“专家”显示模式下显示参数名称 此名称对应于系统变量的名称或 S7T Config 的“专家”列表中组态数据元素的名称。
说明	参数说明
参数类型	指的是 S7T Config 的“专家”列表中的参数类型“系统变量”或“组态数据”。
数据类型	指定参数的数据类型。 可以在“DINT 值的列表”中找到“DINT Enum”数据类型的可用值。

最小值/最大值	指定参数的上限值和下限值。
访问	Read: 可以读取参数。 Write: 可以更改参数。
活动	立即: 在参数中所做的更改立即生效。 Restart: 参数中的更改需要重启才能生效, 意味着重新初始化工艺对象时生效。这种情况仅在工艺对象 TO 被禁用的情况下才可能发生。
固件版本	可从中使用参数的集成工艺的固件版本。

编号	索引	参数说明	
4001	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Accept changes
		DB 参数中的名称 (“专家”显示模式)	<i>activationmodechanged.configdata</i>
		说明	激活已修改的组态数据
		参数类型	系统变量
		数据类型	DINT EnumToActivation ModeSet ConfigData (页 1171)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		固件版本	
4201	-	DB 参数中的名称 (“正常”显示模式)	Actual values.Cam type
		DB 参数中的名称 (“专家”显示模式)	<i>interpolation.camMode</i>
		含义	插补条件
		参数类型	系统变量
		数据类型	DINT EnumCamMode (页 1155)
		最小值	-
		最大值	-
		访问	Read
		活动	-
		固件版本	

编号	索引	参数说明	
4202	-	DB 参数中的名称 (“正常”显示模式)	Actual values.Interpolation type
		DB 参数中的名称 (“专家”显示模式)	<i>interpolation.interpolationmode</i>
		含义	插补类型
		参数类型	系统变量
		数据类型	DINT EnumCamInterpolationMode (页 1153)
		最小值	-
		最大值	-
		访问	Read
		活动	-
		固件版本	
4203	-	DB 参数中的名称 (“正常”显示模式)	Actual values.Definition range.Starting point
		DB 参数中的名称 (“专家”显示模式)	<i>leadingrange.start</i>
		含义	范围的起点
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read
		活动	立即
		固件版本	

编号	索引	参数说明	
4204	-	DB 参数中的名称 (“正常”显示模式)	Actual values.Definition range.End point
		DB 参数中的名称 (“专家”显示模式)	leadingrange.end
		含义	范围的终点
		参数类型	系统变量
		数据类型	REAL
		最小值	-1E+12
		最大值	1E+12
		访问	Read
		活动	立即
		固件版本	
4211	-	DB 参数中的名称 (“正常”显示模式)	Default.Default values.Coordinate reference
		DB 参数中的名称 (“专家”显示模式)	<i>userdefault.campositionmode</i>
		含义	坐标设置的参考
		参数类型	系统变量
		数据类型	DINT EnumCamPositionMode (页 1155)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		固件版本	

9.4.11 工艺参数 — 测量输入

利用本章提供的信息来读取或更改用户程序中工艺对象的参数。工艺对象的参数可以使用“MC_ReadSysParameter”工艺功能来读取，并使用“MC_WriteParameter”工艺功能来写入。

编辑工艺参数的辅助条件：

- 如果在参数中输入的新值等于旧值，则不需要在列表中指定的 **Restart**。此外，在更改虚拟轴的参数时，也不必执行 **Restart**。是否需要 **Restart** 在相应工艺 DB 的 *Statusword.RequestRestart* 变量中指明。
- 由参数编号和索引定义的参数必须在系统中可用并且允许写访问。必须使用工艺功能“MC_ChangeDataset”在 T-CPU 中启用由索引定义的数据记录。

关键字

编号	“MC_ReadSysParameter”和“MC_WriteParameter”工艺功能的输入参数 <i>ParameterNumber</i> 的参数编号
索引	索引可用于多次访问现有参数。以下区域中存在多个参数： • 数据集（Dataset_1 到 Dataset_16） • 编码器（Encoder_1 到 Encoder_8） • 同步对象（1 代表同步对象，2 代表叠加的同步对象） 可以从固件版本为 V3.1.x 和更高版本的集成工艺使用索引。
DB 参数中的名称 （“正常”显示）	如果在 STEP 7 中打开“MC_ReadSysParameter”或“MC_WriteParameter”工艺功能的实例数据块，则在“DB 参数”中会显示一个组态对话框。 在“正常”显示模式下显示参数名称
DB 参数中的名称 （“专家”显示）	如果在 STEP 7 中打开“MC_ReadSysParameter”或“MC_WriteParameter”工艺功能的实例数据块，则在“DB 参数”中会显示一个组态对话框。 在“专家”显示模式下显示参数名称 此名称对应于系统变量的名称或 S7T Config 的“专家”列表中组态数据元素的名称。
说明	参数说明
参数类型	指的是 S7T Config 的“专家”列表中的参数类型“系统变量”或“组态数据”。
数据类型	指定参数的数据类型。 可以在“DINT 值的列表”中找到“DINT Enum”数据类型的可用值。

最小值/最大值	指定参数的上限值和下限值。
访问	Read: 可以读取参数。 Write: 可以更改参数。
活动	立即: 在参数中所做的更改立即生效。 Restart: 参数中的更改需要重启才能生效, 意味着重新初始化工艺对象时生效。这种情况仅在工艺对象 TO 被禁用的情况下才可能发生。
固件版本	可从中使用参数的集成工艺的固件版本。

编号	索引	参数说明	
4001	-	DB 参数中的名称 (“正常”显示模式)	Configuration.Accept changes
		DB 参数中的名称 (“专家”显示模式)	<i>activationmodechanged.configdata</i>
		说明	激活已修改的组态数据
		参数类型	系统变量
		数据类型	DINT EnumToActivationModeSetConfigData (页 1171)
		最小值	-
		最大值	-
		访问	Read, Write
		活动	立即
		固件版本	

9.4.12 DINT 值

9.4.12.1 EnumActiveInactive

EnumActiveInactive

值	含义	ValueDint	S7T Config
INACTIVE	非活动	0	61
ACTIVE	活动	1	4

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.2 EnumAxisApproachDirection

EnumAxisApproachDirection

值	含义	ValueDint	S7T Config
APPROACH_NEGATIVE	沿负方向逼近编码器零标记	15	0
APPROACH_POSITIVE	沿正方向逼近编码器零标记	16	1
EDGE_POS_SIDE_NEG	外部零标记负方向一侧的上升沿	17	2
EDGE_POS_SIDE_POS	外部零标记正方向一侧的上升沿	18	3
EDGE_NEG_SIDE_POS	外部零标记正方向一侧的下降沿	19	4
EDGE_NEG_SIDE_NEG	外部零标记负方向一侧的下降沿	20	5

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.3 EnumAxisEncoderAssemblyType

EnumAxisEncoderAssemblyType

值	含义	ValueDint	S7T Config
ASSEMBLY_BASE_DRIVE	驱动器侧	83	0
ASSEMBLY_BASE_LOAD	负载侧	84	1
ASSEMBLY_BASE_EXTERN	外部	85	2
ASSEMBLY_BASE_LINEAR	线性	86	3

"ValueDint" 列显示了"MC_WriteParameter"和"MC_ReadSysParameter"工艺功能的输入和输出参数的数值。"S7T Config"列包含 S7T Config 的专家列表中相应的数值。

9.4.12.4 EnumAxisExtrapolatedVelocitySwitch

ExtrapolatedVelocitySwitch

值	说明	ValueDint	S7T Config
DIFFERENTIATION	外推主值的微分	0	408
TRANSFER	外推速度的传输	1	409

"ValueDint" 列显示了"MC_WriteParameter"和"MC_ReadSysParameter"工艺功能的输入和输出参数的数值。"S7T Config"列包含 S7T Config 的专家列表中相应的数值。

9.4.12.5 EnumAxisFilterMode

EnumAxisFilterMode

值	含义	ValueDint	S7T Config
DEFAULT_MODE	按照 IPO 时钟与伺服时钟之间的比率函数计算的平均值	9	0
AVERAGING	平均值	10	1
PT1	通过滤波器平滑；使用时间常量调整	11	2

"ValueDint" 列显示了"MC_WriteParameter"和"MC_ReadSysParameter"工艺功能的输入和输出参数的数值。"S7T Config"列包含 S7T Config 的专家列表中相应的数值。

9.4.12.6 EnumAxisFineInterpolatorMode

EnumAxisFineInterpolatorMode

值	含义	ValueDint	S7T Config
DIRECT_MODE	无插补	12	0
LINEAR_MODE	线性插补	13	1
CUBIC_MODE	以恒定速度插补	14	2

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.7 EnumAxisHomingMode

EnumAxisHomingMode

值	含义	ValueDint	S7T Config
MODE_CAM_AND_ZM	参考凸轮和编码器零标记	22	1
MODE_ZM	仅编码器零标记	23	2
MODE_CAM	仅外部零标记	24	3
MODE_NO_REFERENCE	无模式	88	0

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.8 EnumAxisPassiveApproachDirection

EnumAxisPassiveApproachDirection

值	含义	ValueDint	S7T Config
APPROACH_NEGATIVE_PASSIVE	沿负方向逼近编码器零标记	15	0
APPROACH_POSITIVE_PASSIVE	沿正方向逼近编码器零标记	16	1
EDGE_POS_SIDE_NEG_PASSIVE	外部零标记负方向一侧的上升沿	17	2
EDGE_POS_SIDE_POS_PASSIVE	外部零标记正方向一侧的上升沿	18	3
EDGE_NEG_SIDE_POS_PASSIVE	外部零标记正方向一侧的下降沿	19	4
EDGE_NEG_SIDE_NEG_PASSIVE	外部零标记负方向一侧的下降沿	20	5
ACTUAL_DIRECTION_PASSIVE	下一个信号沿	21	6

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.9 EnumAxisPassiveHomingMode

EnumAxisPassiveHomingMode

值	含义	ValueDint	S7T Config
CAM_AND_ZM_PASSIVE	参考凸轮和编码器零标记	22	1
ZM_PASSIVE	仅编码器零标记	23	2
CAM_PASSIVE	仅外部零标记	24	3
DEFAULT_PASSIVE	默认值取决于编码器类型。具有编码器零标记： ZM_PASSIVE 不具有编码器零标记： CAM_PASSIVE	25	4

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.10 EnumAxisTorqueForceReductionGranularity

EnumAxisTorqueForceReductionGranularity

值	说明	ValueDint	S7T Config
BASIC	精度 1/100	1	16
STANDARD	精度 1/16384	0	0

"ValueDint" 列显示了"MC_WriteParameter"和"MC_ReadSysParameter"工艺功能的输入和输出参数的数值。"S7T Config"列包含 S7T Config 的专家列表中相应的数值。

9.4.12.11 EnumBackLashDiff

EnumBackLashDiff

值	含义	ValueDint	S7T Config
DIFF_POSITIVE	正方向	26	0
DIFF_NEGATIVE	负方向	27	1

"ValueDint" 列显示了"MC_WriteParameter"和"MC_ReadSysParameter"工艺功能的输入和输出参数的数值。"S7T Config"列包含 S7T Config 的专家列表中相应的数值。

9.4.12.12 EnumBackLashType

EnumBackLashType

值	含义	ValueDint	S7T Config
NEGATIVE	负方向	4	1
POSITIVE	正方向	5	0

"ValueDint" 列显示了"MC_WriteParameter"和"MC_ReadSysParameter"工艺功能的输入和输出参数的数值。"S7T Config"列包含 S7T Config 的专家列表中相应的数值。

9.4.12.13 EnumBalanceFilterMode

EnumBalanceFilterMode

值	含义	ValueDint	S7T Config
OFF	平衡滤波器不在活动状态	89	0
MODE_1	平衡滤波器处于活动状态	90	1
MODE_2	扩展的平衡滤波器处于活动状态	91	2

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.14 EnumCamInterpolationMode

EnumCamInterpolationMode

值	含义	ValueDint	S7T Config
B_SPLINE	使用贝塞尔样条函数的近似	28	25
C_SPLINE	使用三次样条函数的插补	29	38
LINEAR	线性插补	30	72

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.15 EnumCammingDirection

EnumCammingDirection

值	含义	ValueDint	S7T Config
USER_DEFAULT	用户默认设置/标准	2	149
EFFECTIVE	在程序中设置的最后方向	3	45
NEGATIVE	反向	4	85
POSITIVE	同向	5	107

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.16 EnumCammingMode

EnumCammingMode

值	含义	ValueDint	S7T Config
USER_DEFAULT	用户默认设置/标准	2	149
EFFECTIVE	上一个已编程的设置	3	45
CYCLIC	循环凸轮盘	31	35
NOCYCLIC	非循环凸轮盘	32	92

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.17 EnumCamMode

EnumCamMode

值	含义	ValueDint	S7T Config
CYCLIC_ABSOLUTE	绝对循环凸轮盘	35	36
CYCLIC_RELATIVE	相对循环凸轮盘	36	37
NO_CONSTRAINTS	非循环凸轮盘	37	96

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.18 EnumCamPositionMode

EnumCamPositionMode

值	含义	ValueDint	S7T Config
USER_DEFAULT	用户默认设置/标准	2	149
ACTUAL	具有缩放和偏移的表示法	33	7
BASIC	不具有缩放和偏移的表示法	34	16

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.19 EnumCamTrackType

EnumCamTrackType

值	说明	ValueDint	S7T Config
TYPE_WAY	基于位置的凸轮	60	0
TYPE_TIME	基于时间的凸轮	61	1
TYPE_TIME_WITH_MAX_LEN GTH	具有最大 ON 长度的基于时间的输出凸轮	64	4

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.20 EnumChangeMode

EnumChangeMode

值	含义	ValueDint	S7T Config
NEVER	不转换	92	0
IN_POSITION	轴位于定位窗口时进行转换	93	1
IN_STANDSTILL	轴速低于停止速度时进行转换	94	3
IMMEDIATELY	立即转换	95	7

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.21 EnumDirection

EnumDirection

值	含义	ValueDint	S7T Config
USER_DEFAULT	用户默认设置/标准	2	149
EFFECTIVE	在程序中设置的最后方向	3	45
NEGATIVE	负方向	4	85
POSITIVE	正方向	5	107
BY_VALUE	根据速度设定值的符号得出	38	24
SHORTEST_WAY	最短路径	39	121

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.22 EnumDirectionType

EnumDirectionType

值	含义	ValueDint	S7T Config
NEGATIVE	负方向	4	1
POSITIVE	正方向	5	0

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.23 EnumErrorReporting

EnumErrorReporting

值	含义	ValueDint	S7T Config
NO_REPORTING	无	40	0
COMMAND_VALUE_TOLERANCE	设定值	41	1
ACTUAL_VALUE_TOLERANCE	实际值	42	2

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.24 EnumFollowingObjectSynchronizeWithLookAhead

EnumFollowingObjectSynchronizeWithLookAhead

值	说明	ValueDint	S7T Config
STANDARD	具有 s 和 v 的预测值	112	358
EXTENDED_LOOK_AHEAD	具有 s、v 和 a 的预测值	113	359

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.25 EnumFollowingObjectSynchronizingDirection

EnumFollowingObjectSynchronizingDirection

值	含义	ValueDint	S7T Config
USER_DEFAULT	用户默认设置/标准	2	149
SHORTEST_WAY	没有预置方向的最短路径	96	121
SYSTEM_DEFINED	兼容性模式	97	316
SAME_DIRECTION	与主轴方向相同	98	317
POSITIVE_DIRECTION	正同步方向	99	319
NEGATIVE_DIRECTION	负同步方向	100	320

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.26 EnumForceDirection

EnumForceDirection

值	含义	ValueDint	S7T Config
USER_DEFAULT	用户默认设置/标准	2	149
EFFECTIVE	在程序中设置的最后有效方向	3	45
NEGATIVE	负有效方向	4	85
POSITIVE	正有效方向	5	107
BOTH	正有效方向和负有效方向	43	20

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.27 EnumGearingDirection

EnumGearingDirection

值	含义	ValueDint	S7T Config
USER_DEFAULT	用户默认设置/标准	2	149
EFFECTIVE	在程序中设置的最后方向	3	45
NEGATIVE	反向	4	85
POSITIVE	同向	5	107
BY_VALUE	符号	38	24
CURRENT	当前从方向	44	33
REVERSE	当前从方向的反向	45	116

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.28 EnumGearingMode

EnumGearingMode

值	含义	ValueDint	S7T Config
USER_DEFAULT	用户默认设置/标准	2	149
EFFECTIVE	上一个已编程的类型	3	45
GEARING_WITH_FRACTION	传动比为分子与分母比率的函数	46	55
GEARING_WITH_RATIO	浮点数形式的传动比	47	56

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.29 EnumGearingPosToleranceCommandValue

EnumGearingPosToleranceCommandValue

值	含义	ValueDint	S7T Config
NO_ACTIVATE	无	48	0
WITHOUT_JERK	无冲击	49	1
WITH_JERK	有冲击	50	2

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.30 EnumGearingType

EnumGearingType

值	含义	ValueDint	S7T Config
USER_DEFAULT	用户默认设置/标准	2	149
EFFECTIVE	上一个已编程的类型	3	45
ABSOLUTE	绝对齿轮传动	6	1
RELATIVE	相对变速箱	7	115

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.31 EnumLogicOperation

EnumLogicOperation

值	含义	ValueDint	S7T Config
OPERATION_OR	逻辑 OR	51	0
OPERATION_AND	逻辑 AND	52	1

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.32 EnumLimitExceededOk

EnumLimitExceededOk

值	含义	ValueDint	S7T Config
LIMIT_EXCEEDED		101	71
O_K_	在范围内	102	102

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.33 EnumMasterMode

EnumMasterMode

值	含义	ValueDint	S7T Config
USER_DEFAULT	用户默认设置/标准	2	149
EFFECTIVE	上一个已编程的主模式	3	45
ABSOLUTE	主轴的绝对参考	6	1
RELATIVE	主轴的相对参考	7	115

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.34 EnumMeasuredEdge

EnumMeasuredEdge

值	含义	ValueDint	S7T Config
USER_DEFAULT	用户默认设置/标准	2	149
FALLING_EDGE	下降沿（高 > 低）	53	86
RISING_EDGE	上升沿（低 > 高）	54	117
BOTH_EDGES	两种沿信号处的测量	55	224
BOTH_EDGES_FIRST_RISING	两种沿处的测量，从上升沿开始（从低到高）	56	225
BOTH_EDGES_FIRST_FALLING	两种沿处的测量，从下降沿开始（从高到低）	57	226

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.35 EnumMeasuringRangeMode

EnumMeasuringRangeMode

值	含义	ValueDint	S7T Config
USER_DEFAULT	用户默认设置/标准	2	149
WITHOUT_SPECIFIC_AREA	无指定范围的测量	58	168
WITH_SPECIFIC_AREA	在指定区域中的测量	59	170

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.36 EnumMountSwitch

EnumMountSwitch

值	含义	ValueDint	S7T Config
END_MOUNTED_SWITCH	允许行程范围之外的硬件限位开关始终处于活动状态	103	0
FLEXIBLE_MOUNTED_SWITCH	轴可经过限位开关	104	1

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.37 EnumOutputCamType

EnumOutputCamType

值	含义	ValueDint	S7T Config
TYPE_REVERSE	反向输出凸轮	63	3
TYPE_SWITCH	单方向的输出凸轮	62	2
TYPE_TIME	基于时间的凸轮	61	1
TYPE_WAY	基于位置的凸轮	60	0

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.38 EnumProfile

EnumProfile

值	含义	ValueDint	S7T Config
USER_DEFAULT	用户默认设置/标准	2	149
EFFECTIVE	上一个已编程的速度曲线	3	45
PARABOLIC	抛物线速度曲线/不可用	64	103
SINUSOIDAL	正弦速度曲线/不可用	65	122
SMOOTH	平稳加速曲线	66	124
TRAPEZOIDAL	梯形速度曲线	67	146

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.39 EnumRecognitionMode

EnumRecognitionMode

值	含义	ValueDint	S7T Config
DO_NOT_CLAMP	未检测到	105	0
CLAMP_BY_FOLLOWING_ERROR_DEVIATION	当超出跟随误差时	106	1
CLAMP_WHEN_TORQUE_LIMIT_REACHED	超出扭矩时/不可用（或待验证）	107	2

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.40 EnumSensorState

EnumSensorState

值	含义	ValueDint	S7T Config
NOT_VALID	无效	204	109
WAIT_FOR_VALID	等待验证	205	110
VALID	值有效	206	111

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.41 EnumSlaveMode

EnumSlaveMode

值	含义	ValueDint	S7T Config
USER_DEFAULT	用户默认设置/标准	2	149
EFFECTIVE	上一个已编程的从模式	3	45
ABSOLUTE	从轴的绝对参考	6	1
RELATIVE	从轴的相对参考	7	115

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.42 EnumSyncModeCamming

EnumSyncModeCamming

值	含义	ValueDint	S7T Config
USER_DEFAULT	用户默认设置/标准	2	149
EFFECTIVE	上一个已编程的设置	3	45
IMMEDIATELY	立即生效	8	60
AT_THE_END_OF_CAM_CYCLE	在活动凸轮结束时转换	68	14
NEXT_WITH_REFERENCE	在下一个引导轴位置/不可用	69	90
ON_MASTER_AND_SLAVE_POSITION	引导轴和跟随轴的默认同步位置	70	99
ON_MASTER_POSITION	引导轴的默认同步位置	71	100
IMMEDIATELY_AND_SLAVE_POSITION	立即生效和从轴的同步位置	108	315

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.43 EnumSyncModeGearing

EnumSyncModeGearing

值	含义	ValueDint	S7T Config
USER_DEFAULT	用户默认设置/标准	2	149
EFFECTIVE	上一个已编程的设置	3	45
IMMEDIATELY	立即生效	8	60
NEXT_WITH_REFERENCE	在下一个引导轴位置/不可用	69	90
ON_MASTER_AND_SLAVE_POSITION	引导轴和跟随轴的默认同步位置	70	99
ON_MASTER_POSITION	引导轴的默认同步位置	71	100
ON_SLAVE_POSITION	由从轴的同步位置指定	72	101
IMMEDIATELY_AND_SLAVE_POSITION	立即生效和从轴的同步位置	108	315
IMMEDIATELY_AND_BE_SYNCHRONOUS_AT_MASTER_POSITION	带有时基的立即绝对同步/不可用	74	370

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.44 EnumSyncOffModeCamming

EnumSyncOffModeCamming

值	含义	ValueDint	S7T Config
USER_DEFAULT	用户默认设置/标准	2	149
EFFECTIVE	上一个已编程的设置	3	45
IMMEDIATELY	立即生效	8	60
AT_THE_END_OF_CAM_CYCLE	凸轮循环结束	68	14
ON_MASTER_POSITION	在引导轴的位置	71	100
ON_SLAVE_POSITION	在跟随轴的位置	72	101

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.45 EnumSyncOffModeGearing

EnumSyncOffModeGearing

值	含义	ValueDint	S7T Config
USER_DEFAULT	用户默认设置/标准	2	149
EFFECTIVE	上一个已编程的设置	3	45
IMMEDIATELY	立即生效	8	60
ON_MASTER_POSITION	由引导轴的去同步位置指定	71	100
ON_SLAVE_POSITION	跟随轴的默认去同步位置	72	101

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.46 EnumSyncOffPositionReference

EnumSyncOffPositionReference

值	含义	ValueDint	S7T Config
USER_DEFAULT	用户默认设置/标准	2	149
EFFECTIVE	上一个已编程的设置	3	45
AXIS_STOPPED_AT_POSITION	在去同步位置之前停止	73	15
BEGIN_TO_STOP_WHEN_POSITION_REACHED	在去同步位置之后停止	74	18
STOP_SYMMETRIC_WITH_POSITION	在与去同步位置对称时停止	75	137

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.47 EnumSyncPositionReference

EnumSyncPositionReference

值	含义	ValueDint	S7T Config
USER_DEFAULT	用户默认设置/标准	2	149
EFFECTIVE	上一个已编程的设置	3	45
BE_SYNCHRONOUS_AT_POSITION	在同步位置之前同步	76	19
SYNCHRONIZE_SYMMETRIC	在与同步位置对称时同步	77	143
SYNCHRONIZE_WHEN_POSITION_REACHED	从同步位置同步	78	144

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.48 EnumSyncProfileReference

EnumSyncProfileReference

值	含义	ValueDint	S7T Config
USER_DEFAULT	用户默认设置/标准	2	149
EFFECTIVE	上一个已编程的设置	3	45
RELATE_SYNC_PROFILE_TO_LEADING_VALUE	特定于引导轴的同步曲线	79	113
RELATE_SYNC_PROFILE_TO_TIME	与时间相关的同步曲线	80	114

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.49 EnumToActivationModeSetConfigData

EnumToActivationModeSetConfigData

值	含义	ValueDint	S7T Config
ACTIVATE_CHANGED_CONFIG_DATA	立即激活组态数据	81	291
COLLECT_CHANGED_CONFIG_DATA	收集组态数据，不激活	82	292

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.4.12.50 EnumYesNo

EnumYesNo

值	含义	ValueDint	S7T Config
NO	激活：否	0	91
YES	激活：是	1	173

"ValueDint" 列显示了“MC_WriteParameter”和“MC_ReadSysParameter”工艺功能的输入和输出参数的数值。“S7T Config”列包含 S7T Config 的专家列表中相应的数值。

9.5 Internet 上的附加信息

9.5.1 Internet 上的附加信息

如果您的 PG/PC 提供 Internet 访问功能，则您可以从 Internet 上下载辅助信息。

如果您的 PG/PC 可以连接至 Internet，则还可以从下述 Internet URL 处获得有关 T-CPU 的附加信息：

您的 Siemens 联系伙伴

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

服务与支持

<http://www.siemens.com/automation/service&support>

所有 A&D 产品的工艺支持

<http://www.siemens.de/automation/support-request>

SIMATIC 文档

<http://www.ad.siemens.de/support>

培训中心

<http://www.sitrain.com>

常见问题

<http://support.automation.siemens.com>

索引

A

AccelerationOverride — 工艺 DB, 785, 791, 798
ACTUAL, 888, 889, 890, 893, 894
ACTUAL — 存储区, 886, 888, 889, 890
ActualAcceleration — 工艺 DB, 785, 791, 798
ActualPosition — 工艺 DB, 791, 798, 807
ActualVelocity — 工艺 DB, 785, 791, 798, 807
ADI4 上的增量编码器 — 编码器参数, 196
ADI4 上的绝对编码器 — 编码器参数, 196
AND 运算 — 输出凸轮, 361

B

BERO — 参考凸轮, 42
Bootblock 版本, 949
BRB 位, 425
Busy — 监视活动命令, 422

C

CamEdit, 344, 894
CamEdit — 插补 2, 346
CamTool, 28
CamType — 工艺 DB, 816, 820
CmdLoopDuration — 工艺 DB, 826
CmdLoopDuration — 时序, 401
CommandAborted — 监视活动命令, 422
CommandAcceleration — 工艺 DB, 785, 791, 798
CommandPosition — 工艺 DB, 791, 798
CommandVelocity — 工艺 DB, 785, 791, 798
CPU 31xT 的地址, 418

CPU 31xT 的地址空间, 418
CPU 31xT 的时序, 401
CPU 升级, 944
CPU 更新, 944
CPU 替换, 946
CPU 装载, 401
CyclicMode, 105

D

DB 参数 — MCDevice, 826
DB 参数 — 凸轮盘, 811
DB 参数 — 凸轮轨迹, 820
DB 参数 — 同步轴, 798
DB 参数 — 外部编码器, 807
DB 参数 — 定位轴, 791
DB 参数 — 测量输入, 813
DB 参数 — 跟踪, 825
DB 参数 — 输出凸轮, 816
DB 参数 — 速度控制轴, 785
DB2INT, 430
DBDevice, 829
DecelerationDistance — 工艺 DB, 791, 798
DenomGear — 工艺 DB, 798
Distance — 工艺 DB, 791, 798
DIStatus — 工艺 DB, 826
Done — 监视活动命令, 422
DoneFlag, 31, 782
DoneFlag — 工艺 DB, 826
DoneFlag — 生成, 782

DStatus — 工艺 DB, 826

DP 周期, 897

DP 周期 — 时序, 401

DP 驱动器优化 — 地址 0, 114

DP 驱动器优化 — 总线参数, 114

Drive ES Basic, 28

DriveMonitor, 28

DSC, 56

E

ErrorBuffer, 426, 427, 428

ErrorBuffer — 工艺 DB, 785, 791, 798, 807, 811, 813, 816, 820, 826

ErrorID — MC_ActivateTO, 712

ErrorID — MC_ChangeDataset, 479

ErrorID — MC_GearIn, 553

ErrorID - MC_MoveRelative, 501

ErrorID — MC_Stop, 467

ErrorID — MC_WriteRecord, 752

ErrorID — MCDevice/Trace 工艺 DB, 882

ErrorID — 凸轮工艺 DB, 864

ErrorID — 凸轮轨迹工艺 DB, 878

ErrorID — 外部编码器工艺 DB, 857

ErrorID — 工艺 DB, 785, 791, 798, 807, 811, 813, 816, 820, 825, 826

ErrorID— MC_SetCharacteristic, 543

ErrorStatus — 工艺 DB, 785, 791, 798, 807

ErrorStatus.xxx, 427

Execute — 监视活动命令, 422

Execution.ExecutionLevel, 412

F

FB 414 MC_MoveVelocity, 518

FB 415 MC_MoveToEndPos, 528

FB 430 MC_CamSwitch, 659

FB 431 MC_CamSwitchTime, 667

FB 433 MC_MeasuringInput, 691

FB 434 MC_CamClear, 631

FB 437 MC_SetTorqueLimit, 536

FB403 MC_Home, 452

FB404 MC_Stop, 464

FB405 MC_Halt, 470

FB410 MC_MoveAbsolute, 481

FB411 MC_MoveRelative, 493

FB412 MC_MoveAdditive, 504

FB413 MC_MoveSuperImposed, 510

FB420 MC_GearIn, 545

FB422 MC_GearOut, 556

FB423 MC_CamOut, 594

FB432 MC_ExternalEncoder, 697

FB439 MC_SetCharacteristic, 541

FB450 MC_ReadPeriphery, 733

FB451 MC_WritePeriphery, 739

FollowingError — 工艺 DB, 791, 798

H

HW Config, 114, 118

I

Internet 上的常见问题, 943

Internet 上的应用示例, 937

Internet 上的附加信息, 1173

IPO, 401

IPO 溢出, 401, 897

IPO 级, 58

L

LAD/FBD/STL, 28

M

MASTERDRIVES — 硬件组件, 27

MaxLoopDuration — 工艺 DB, 826

MaxLoopDuration — 时序, 401

MC_ActivateDPSlave, 19

MC_ActivateTO, 19

MC_ActivateTO — ErrorID, 712

MC_CamClear, 631

MC_CamIn — 偏移因子的影响, 582

MC_CamIn — 示例 — 同步条件

AT_MASTER_AND_SLAVE_POSITION, 590

MC_CamIn — 示例 — 周期结束时的凸轮转换, 589

MC_CamIn — 示例 — 带有引导轴位置的同步, 583, 585

MC_CamIn — 示例 — 循环相对同步, 583

MC_CamIn — 示例 — 循环绝对同步, 585

MC_CamIn — 缩放因子的影响, 582

MC_CamInSuperImposed — 叠加示例, 602

MC_CamInSuperImposed — 示例, 604

MC_CamOut, 594

MC_CamOut — 示例, 596

MC_CamSectorAdd — 两个凸轮区段之间的插补, 648

MC_CamSectorAdd — 凸轮区段系数的作用, 640

MC_CamSwitch, 659

MC_CamSwitch — 示例: 单方向的输出凸轮, 663

MC_CamSwitchTime, 667

MC_CamTrack, 19

MC_ChangeDataset — ErrorID, 479

MC_ChangeDataset — 应用示例, 478

MC_ChangeDataset — 编码器转换示例, 478

MC_ExternalEncoder, 697

MC_ExternalEncoder — 示例, 700

MC_GearIn, 545

MC_GearIn —, 549

MC_GearIn — ErrorID, 553

MC_GearIn — 不使用相移的绝对同步, 551

MC_GearIn — 使用相移的绝对同步, 551

MC_GearIn — 基于组态数据的同步, 549

MC_GearIn — 示例, 549

MC_GearIn — 示例 — 基于组态数据的同步, 550

MC_GearIn — 示例 — 相移, 551

MC_GearInSuperImposed — 示例 — 叠加, 565

MC_GearInSuperImposed — 示例 — 相对绝对, 567

MC_GearInSuperImposed — 示例 — 相移, 569

MC_GearOut, 556

MC_GearOut — 示例, 558

MC_GetCamPoint, 20

MC_Halt — 示例, 472

MC_Home, 452

MC_Home — 模式 6 示例, 459

MC_Home — 模式 7 示例, 460

MC_Home — 被动回原点示例, 458

MC_MeasuringInput, 691

MC_MeasuringInput — 示例, 693

MC_MoveAbsolute, 481

MC_MoveAbsolute — 模式 0 示例, 484

MC_MoveAbsolute — 模式 1 示例, 486

MC_MoveAbsolute — 模式 2 示例, 487

MC_MoveAbsolute — 添加运动示例, 486

MC_MoveAbsolute — 示例 — 覆盖运动 1, 487

MC_MoveAbsolute — 示例 — 覆盖运动 2, 488

MC_MoveAbsolute — 示例 — 覆盖运动 3, 489

MC_MoveAbsolute — 超驰运动示例, 484

MC_MoveAdditive, 504

MC_MoveAdditive — 示例, 507

MC_MoveRelative, 493

MC_MoveRelative — ErrorID, 501
MC_MoveRelative — 示例 — 覆盖运动 1, 498
MC_MoveRelative — 示例 — 覆盖运动 2, 499
MC_MoveRelative — 示例 — 覆盖运动 3, 500
MC_MoveRelative — 示例模式 0, 495
MC_MoveRelative — 示例模式 1, 497
MC_MoveRelative — 示例模式 2, 498
MC_MoveRelative — 示例添加运动, 497
MC_MoveRelative — 示例超驰运动, 495
MC_MoveSuperImposed, 510
MC_MoveSuperImposed — 示例 —, 514, 543
MC_MoveSuperImposed — 绝对定位的示例, 513
MC_MoveToEndPos, 528
MC_MoveToEndPos — 扭矩设置, 971
MC_MoveToEndPos — 示例, 532
MC_MoveVelocity, 518
MC_MoveVelocity — 示例 —, 524
MC_MoveVelocity — 示例 — 模式 0, 523
MC_MoveVelocity — 示例 — 模式 1, 524
MC_MoveVelocity — 示例 — 超驰运动, 523
MC_Phasing — 示例, 618
MC_Power — 液压轴示例, 448
MC_Power — 示例, 447
MC_ReadCamTrackData, 19
MC_ReadPeriphery, 733
MC_ReadPeriphery — 地址访问, 418, 420
MC_ReadPeriphery — 示例, 735
MC_Reset, 416, 426, 427, 428, 894
MC_Reset — 示例, 447
MC_SetCharacteristic, 541
MC_SetCharacteristic — ErrorID, 543
MC_SetCharacteristic — 示例, 543
MC_SetTorqueLimit, 536
MC_SetTorqueLimit — 扭矩设置, 971
MC_SetTorqueLimit — 示例, 538
MC_Stop, 464

MC_Stop — ErrorID, 467
MC_Stop — 示例, 466
MC_WriteCamTrackData, 19
MC_WriteParameter, 431, 894
MC_WriteParameter — 示例, 726
MC_WriteParameter — 示例 —, 728
MC_WriteParameter — 示例 — 驱动器控制字, 726
MC_WritePeriphery, 739
MC_WritePeriphery — 地址访问, 418, 420
MC_WritePeriphery — 示例, 741
MC_WriteRecord — ErrorID, 752
MCDevice, 31
MCDevice — 工艺 DB — 固件版本为 V3.1.x 的集成工艺, 826
MCDevice — 错误, 426
MCDevice/Trace 工艺 DB — ErrorID, 882
MeasureEnd — 工艺 DB, 813
MeasureStart — 工艺 DB, 813
MeasureValue — 工艺 DB, 813
MeasureValue1 — 工艺 DB, 813
MICROBOX T 的地址, 420
MICROBOX T 的地址空间, 420
MICROBOX T 的时序, 406
MICROMASTER — 硬件组件, 27
MMC — 存储区, 886
MMC 卡 — 存储区, 886
Mode — 工艺 DB, 816
motionstatedata.actualacceleration, 145, 215
motionstatedata.actualvelocity, 145, 215

N

NEXT, 894
NEXT — 存储区, 886
NumGear — 工艺 DB, 798

O

OB1 — 时序, 401
OB1 — 编程模型, 416
OB1 — 顺序模型和编程模型, 416
OB32 到 OB35 — 时序, 401
OB65, 829
OB65 — 时序, 401
OR 运算 — 输出凸轮, 361

P

PhaseShift — 工艺 DB, 798
PLCopen, 22
positioningstate.actualposition, 145, 215
POWER ON, 888
pTc, 918

Q

Q 阀, 35, 191, 541

R

RAM, 888, 889, 890, 893, 894
RAM — 存储区, 886
RAM 区域, 886, 888, 889, 890, 893, 894
ReadSysParameter, 31
REAL — 值范围, 783
REAL 值的范围 — REAL 值, 783
ROM, 888, 889, 890, 893, 894
ROM — 存储区, 886
ROM 区域, 886, 888, 889, 890, 893, 894

S

S7_TraceDINT — 工艺 DB, 825

S7_TraceDWORD — 工艺 DB, 825
S7_TraceREAL — 工艺 DB, 825
S7T Config, 28
S7T Config 中的专家列表, 967
S7T Config 中的在线更改, 894
S7T Config 中的装载命令, 890
S7T Config 的用户界面, 131
S7-Tech 库, 28
S7-Technology, 28
S7-Technology 中的新增内容, 19
sensordata, 924
sensordata.sensordata[n]. velocity, 145, 215
sensordata.sensordata[n].acceleration, 145, 215
sensordata.sensordata[n].incrementalposition, 145, 215
sensordata.sensordata[n].position, 145, 215
sensormonitoring, 924
servodata, 924
servomonitoring, 924
servosettings, 924
SIMATIC Manager, 28
SIMATIC S7-Technology, 28
SimoComU, 28
SIMODRIVE — 硬件组件, 27
SIMODRIVE 传感器 — 编码器参数, 375
SIMODRIVE 传感器等时 — 硬件组件, 27
SINAMICS — 硬件组件, 27
SpeedMode — 速度控制, 785, 791, 798
SpeedMode — 速度控制模式, 518
SpeedOverride — 工艺 DB, 785, 791, 798
STARTER V4.1, 19
STARTER 功能, 29
State — 工艺 DB, 816, 820
Status — 工艺 DB, 811, 813
Statusword — 工艺 DB, 785, 791, 798, 807

Statusword.Error, 427

Statusword.Errorstop, 427

STEP 7 — 组态工具, 28

STEP 7 中的装载命令, 884

STEP 7 中集成的符合 PLCopen 的运动控制功能, 22

STEP 7 报告存储空间不足, 966

STOP 到 RUN, 890

SyncStatus — 工艺 DB, 798

T

TargetPosition — 工艺 DB, 791, 798

T-CPU, 28, 114

T-CPU 的周期, 412

TODBTaskOverflows, 826

TOM, 28

TOM — 启动, 125

TOM — 用户界面, 126

TOM 用户界面, 126

Trace — 错误, 426

TraceTool, 31

TraceTool — 总览, 923

TypeOfAxis.NumberOfEncoders.Encoder_n. Filter, 145, 215

TypeOfAxis.SmoothingFilter, 145, 215

U

UDT 100, 682, 687

UpdateCounter — 工艺 DB, 785, 791, 798, 807, 811, 813, 816, 820, 826, 829

UpdateFlag — 工艺 DB, 785, 791, 798, 807, 811, 813, 816, 820, 826, 829

UserCount — 工艺 DB, 811

V

VDI — 定义运动控制任务的凸轮盘区段, 358

VDI — 运动法则, 356

VDI 向导, 356

vTc, 918

W

WinLC T MICROBOX T 的时序, 406

WriteParameter, 31

一

一致性, 961, 963

三

三次样条函数, 78

下

下载 — 用户程序组态数据固件, 884

不

不一致, 963

与

与增量编码器同步 — 外部编码器, 384

专

专家列表, 894

专用驱动器单元, 29

两

两个凸轮区段之间的插补, 648

为

为什么某些系统 DB 具有不同的在线/离线时间戳? , 965

主

主动回原点, 170, 239, 304

主动回原点 — 仅使用编码器零标记的回原点模式, 170, 239, 304

主动回原点 — 使用参考凸轮和编码器零标记的回原点模式, 170, 239, 304

主动回原点 — 参考点逼近的顺序, 170, 239, 304

主设定值 — 同步组, 58

主设定值耦合, 58

主轴 — 齿轮传动, 545

互

互连 — 外部编码器, 110

互连 — 测量输入, 108

仅

仅使用 BERO 的被动回原点模式, 178, 248, 313, 393

仅使用编码器零标记的主动回原点模式, 170, 239, 304

仅使用编码器零标记的被动回原点模式, 178, 248, 313, 393

从

从 Actual 复制到 ROM, 893

从另一个站中复制组态数据, 399

从轴 — 齿轮传动, 545

优

优先级 — 时序, 401

传

传动比, 62

传感器 — 驱动器 — 实际值记录, 145, 215

伺

伺服增益因子 Kv, 910

位

位置参考, 278

位置和速度 — 轴, 153, 222, 287

位置控制, 48

位置控制周期, 412, 897

位置控制周期时钟 — 时序, 401

位置控制器, 168, 302

位置控制器 — 调节, 901

位置控制器调节 — 总览, 901

位置控制回路的等效时间, 918

位置控制回路的等效时间常数, 918

位置控制模式, 518

位置控制轴的速度控制操作, 57, 518

位置校正, 185, 255, 320, 510

作

作业处理任务 — 作业监视时间, 897

作业监视时间, 897

使

使用, 965

使用 BERO 和编码器零标记的被动回原点模式, 178, 248, 313, 393

使用 CamEdit 创建凸轮, 343

使用 CamTool 创建凸轮盘, 355

使用 FB 411 MC_MoveRelative 进行相对定位, 493

使用 FB 421, 578

使用 FB 430 MC_CamSwitch 的基于位置的凸轮或单方向的输出凸轮, 659

使用 FB 431 MC_CamSwitchTime 的基于时间的凸轮, 667

使用 FB 433 C_MeasuringInput 的测量输入, 691

使用 FB 434 MC_CamClear 删除凸轮, 631

使用 FB 434 MC_CamClear 删除凸轮的内容, 631

使用 FB 437, 536

使用 FB 457 MC_ActivateDPSlave 取消激活/激活 DP 从站, 714

使用 FB403 MC_Home 来使轴回原点/设置轴, 452

使用 FB405 MC_Halt 进行正常停止, 470

使用 FB410 MC_MoveAbsolute 进行绝对定位, 481

使用 FB412 MC_MoveAdditive 相对定位到当前目标位置, 504

使用 FB413 MC_MoveSuperImposed 进行叠加定位, 510

使用 FB414, 518

使用 FB420, 545

使用 FB422 MC_GearOut 停止齿轮传动, 556

使用 FB423 MC_CamOut 停止凸轮传动, 594

使用 FB432 MC_ExternalEncoder 的外部编码器, 697

使用 FB450 MC_ReadPeriphery 读取工艺 I/O, 733

使用 FB451 MC_WritePeriphery 写入工艺 I/O, 739

使用 FC400 DB2INT 进行符号编程, 430

使用 TOM, 127

使用专家列表, 967

使用回原点模式的默认设置的被动回原点模式, 178, 248, 313, 393

使用工艺对象管理, 127

使用控制面板进行测试, 927

使用断点进行测试, 920

使用目标传感器定位的应用示例, 937

使用表格的示例, 780

使能外设输出, 932

倍

倍增因子 — 高精度, 144

值

值范围 — 工艺 DB, 783

偏

偏移, 829

偏移量补偿, 510

停

停止 — 基础, 189, 259, 325

停止 — 运行, 782

停止（零速度）监视, 187, 257, 323

停止信号 — 基础, 189, 259, 325

停止信号 — 外部编码器, 386

停止信号 — 设置, 386

停止和阻止新运动命令, 464

停止时间, 151, 220, 285

停止跟随轴（凸轮传动）, 594

停止跟随轴（同步操作）, 556

关

关闭时间, 897

具

具有 NIST 评估的编码器, 165, 234, 299, 392
具有最大 ON 长度的基于时间的输出凸轮, 105

写

写保护, 961

减

减速度, 157, 226, 291
减速比, 829

几

几何图形, 78, 339

凸

凸轮 — 插补类型, 81
凸轮 — 标准化输入, 342, 640
凸轮传动, 64
 响应特性, 64
 缩放和偏移, 64
凸轮传动 — 基础, 64
凸轮区段系数的作用, 640
凸轮工艺 DB — ErrorID, 864
凸轮插补, 78
凸轮激活时间, 93, 104, 105
凸轮点的导数, 20
凸轮盘, 64
 循环应用和非循环应用, 64
凸轮盘 — MC_CamClear, 631
凸轮盘 — 工艺 DB, 811
凸轮盘 — 工艺对象, 75
凸轮盘 — 应用, 77
凸轮盘 — 插补, 78

凸轮盘工艺对象, 75
凸轮盘应用, 77
凸轮盘的在线更改, 894
凸轮类型, 346
凸轮耦合, 58
凸轮轨迹, 19
凸轮轨迹 — 工艺 DB, 820
凸轮轨迹 — 工艺对象, 100
凸轮轨迹 — 组态, 366
凸轮轨迹 — 输出凸轮数据, 368
凸轮轨迹 — 默认值, 368
凸轮轨迹工艺 DB — ErrorID, 878
凸轮轨迹工艺对象, 100

分

分析项目, 961
分配凸轮, 263
分配引导轴和凸轮盘 — 实际值耦合, 263
分配引导轴和凸轮盘 — 设定值耦合, 263
分配执行周期, 412
分配插补器周期 2, 970
分配系统时钟, 412, 414
分配阀控曲线, 211

切

切换操作 — 切换输出凸轮, 92
切换方向阀, 35
切换精度, 363
切换输出凸轮, 92
 切换操作, 92
 反向切换操作, 92
切换输出凸轮 — 切换操作, 92

列

列表 — 问题、提示和技巧, 943

创

创建和管理工艺 DB, 128

创建电气轴, 134

删

删除 — 工艺数据块, 128

删除使能后扭矩限制保持不变, 20

删除凸轮, 631

删除某个轴的数据记录, 142, 212

到

到驱动器的路由, 883

制

制动控制, 20

功

功能块, 31

加

加速度, 157, 226, 291

加速度模型 — 取决于方向, 157, 226, 291

动

动态伺服控制 DSC, 56

动态参数 — 值范围, 783

动态响应, 151, 220, 285

动态响应 — 减速度, 157, 226, 291

动态响应 — 加速度, 157, 226, 291

动态响应 — 硬件限制, 157, 226, 291

动态响应 — 绝对值, 157, 226, 291

动态响应 — 软件限制, 157, 226, 291

动态响应调整, 53, 168, 302

动态控制器数据, 168, 302

动态控制器数据 — 液压轴, 236

包

包括参考点 — 实际值记录, 145, 215

升

升级 CPU 31xT-2 DP 的固件, 949

升级后下载, 960

升级后如何更新工艺功能? , 959

升级后需要将哪些内容下载到 PLC? , 960

升级工艺, 950

升级集成工艺的固件, 950

单

单方向的输出凸轮 — 示例, 663

压

压力标记校正, 510

去

去同步 — 位置参考, 278

去同步 — 同步操作, 278

参

参考凸轮 — 回原点, 42

反

反向输出, 94

取

取决于方向的加速度模型, 157, 226, 291

取决于状态 — 加速度模型, 157, 226, 291

取决于状态的加速度模型, 157, 226, 291

受

受控变量监视, 190

受控变量限制, 166, 300

变

变速泵, 35, 206

叠

叠加同步, 58, 69

叠加坐标系 — 基准坐标系, 40

叠加运动, 40, 69

同

同步 — MC_GearIn — 基于动态设置, 549

同步 — MC_GearIn — 基于组态数据, 549

同步 — 与引导轴相关的同步曲线, 267

同步 — 位置参考, 274

同步 — 同步操作, 267, 274

同步 — 外部编码器, 111

同步 — 延迟同步, 267

同步 — 曲线设置, 267

同步 — 组态, 267

同步 — 高级同步, 267

同步对象缺省值 — 主动态响应 (Master dynamic response) 选项卡, 338

同步对象缺省值 — 凸轮同步 (Cam synchronization) 选项卡, 334

同步对象默认值 — 凸轮同步 (Cam synchronization) 选项卡, 331

同步对象默认值 — 动态 (Dynamics) 选项卡, 336

同步对象默认值 — 齿轮同步 (Gear synchronization) 选项卡, 332

同步操作, 62

同步操作 — 位置参考, 274, 278

同步操作 — 分配凸轮, 263

同步操作 — 分配引导轴和凸轮盘, 263

同步操作 — 去同步, 278

同步操作 — 同步, 274

同步操作 — 实际值耦合, 263

同步操作 — 监视功能, 321

同步操作 — 设定值耦合, 263

同步操作中的响应特性, 62

同步条件, 274

同步状态, 280

同步监视, 69, 325

同步组, 58

同步轴 — 工艺 DB, 798

同步轴 — 工艺对象, 57

同步轴工艺对象的结构, 58

同步运动的校正, 64

启

启动 — 轴控制面板, 928

启动 OB, 416

启动 S7T Config, 130

启动工艺对象管理, 125

启用轴, 933

启用驱动器, 448

周

周期, 401
周期性中断, 416
周期性实际值的倍增因子, 196

命

命令执行 — 时序, 401
命令监视, 422

回

回原点, 42, 170, 239, 304
回原点 —, 170, 239, 304
回原点 — 增量编码器, 111
回原点 — 被动回原点 (Passive homing) 选项卡, 178, 248, 313, 393
回转轴, 34

固

固件升级, 944
固件更新, 944
固定挡块 — MC_MoveToEndPos, 528
固定终点挡板, 159, 228, 293
固定终点挡板 — 超出扭矩限制, 159, 228, 293
固定终点挡板 — 跟随误差监视, 159, 228, 293

在

在 HW Config 中组态 T-CPU, 114
在 HW Config 中组态驱动器, 118
在 S7T Config 中组态测量输入, 371
在同步操作期间复位轴位置, 62
在控制面板上监视轴值, 930
在线, 965
在线/离线比较, 963

块

块编辑器, 28

垂

垂直轴, 20

基

基于位置的凸轮, 91, 103
 凸轮激活范围, 91, 103
 切换特性, 91, 103
基于位置的凸轮
 凸轮激活范围, 91
 反向输出, 91
基于位置的凸轮
 凸轮激活范围, 103
基于位置的凸轮
 反向输出, 103
基于时间的凸轮, 93, 104
 切换特性, 93, 104, 105
 反向输出, 93, 104
 有效方向, 105
 有效方向, 93, 104
基于时间的凸轮 — 有效方向, 93, 104
基准坐标系 — 叠加坐标系, 40
基本同步对象, 69
 坐标, 69
基本知识
 需要的, 3
基本运动, 40

增

增益因子 Kv, 908
增量编码器 — 回原点, 111
增量编码器 — 外部编码器, 111

复

复制 — 工艺数据块, 128

外

外部编码器 — 与增量编码器同步, 384
外部编码器 — 互连, 110
外部编码器 — 动态伺服控制 — DSC, 110
外部编码器 — 同步, 111
外部编码器 — 工艺 DB, 807
外部编码器 — 接线, 110
外部编码器 — 机械, 387
外部编码器 — 监视功能, 972
外部编码器 — 直接回原点, 111
外部编码器 — 组态, 386
外部编码器 — 被动回原点, 111
外部编码器 — 高精度, 144
外部编码器 — 默认值, 383
外部编码器工艺 DB — ErrorID, 857

多

多项式, 342, 640

如

如何使用 S7-Technology 分析早期版本中的项目, 961
如何使用 S7-Technology 编辑较旧的项目, 963
如何利用新性能特性, 944
如何升级 CPU 31xT-2 DP 的固件, 949
如何升级集成工艺的固件, 950
如何将 T-CPU 替换为其它类型, 946
如何识别固件版本, 948
如何转换工艺, 952
如何转换工艺以及如何升级工艺包, 953
如何重新创建工艺 DB, 957

存

存储区 — NEXT, 886
存储区 NEXT, 894
存储器复位, 888

定

定义凸轮, 342, 640
定义凸轮盘, 342, 640
定义凸轮盘 — 缩放, 76
定义运动控制任务的凸轮盘区段 — VDI, 358
定位和零速度监视, 187, 257, 323
定位监视, 187, 257, 323
定位轴 — 工艺 DB — 固件版本为 V3.1.x 的集成工艺, 791
定位轴 — 工艺对象, 47

实

实时回原点 = 被动回原点, 178, 248, 313, 393
实际位置值, 163, 232, 297, 390
实际位置值倒置, 164, 233, 298, 391
实际值, 31
实际值 — 位置, 145, 215
实际值 — 加速度, 145, 215
实际值 — 速度, 145, 215
实际值耦合, 263
实际值记录, 145, 215, 972
实际存储区, 886, 893, 894
实际速度值, 164, 233, 298, 391

对

对工艺数据执行, 967
对模数轴的限制, 504
对轴特定的参数更改进行编程, 431
对项目进行写保护, 961

将

将凸轮盘用作阀控曲线, 206

工

工作范围和运动转换, 356

工艺 DB, 31, 128

工艺 DB — MCDevice, 826

工艺 DB — 值范围, 783

工艺 DB — 凸轮盘, 811

工艺 DB — 凸轮轨迹, 820

工艺 DB — 同步轴, 798

工艺 DB — 外部编码器, 807

工艺 DB — 定位轴, 791

工艺 DB — 测量传感器, 813

工艺 DB — 跟踪, 825

工艺 DB — 输出凸轮, 816

工艺 DB — 速度控制轴, 785

工艺 DB 更新, 897

工艺 DB (Axes 和 External Encoders) 中的错误和警告, 427

工艺 DB (Cam Disk 及 Measuring Input 和 Output Cam) 中的错误和警告, 428

工艺 DB (MCDevice 和 Trace) 中的错误, 426

工艺功能, 31

工艺功能和工艺 DB, 31

工艺功能的错误和警告, 425

工艺同步中断, 829

工艺同步中断 — 时序, 401

工艺对象

 用同一个时钟操作, 412

工艺对象 — 同步轴, 57

工艺对象 — 定位轴, 47

工艺对象 — 系统时钟, 412

工艺对象 — 速度控制轴, 46

工艺对象管理, 28

工艺对象管理的用户界面, 126

工艺数据块, 31

已

已完成的消息, 31

带

带有切换方向阀的变速泵的阀控曲线, 206

带编码器的液压轴的机械设置, 219

带编码器的电气轴的机械设置, 148, 282

常

常见问题, 943

平

平均 — 实际值记录, 145, 215

平衡滤波器 = 平衡滤波器, 913

应

应用示例 — MC_ChangeDataset, 478

应用示例夹具进给, 941

应用示例飞剪, 939

延

延迟, 98, 659, 667

引

引导轴 — 齿轮传动, 545

引言 — 控制面板, 927

循

循环创建凸轮轨迹, 106

微

微分 — 实际值记录, 145, 215

微分作用时间 — 输出凸轮, 98

微分作用时间输出凸轮, 98

总

总览 — TraceTool, 923

总览 — 监视功能, 186, 256, 321

总览 — 组态, 113

手

手册

用途, 3

适用范围, 3

打

打印标记, 691

执

执行, 416

执行周期, 897, 900

扭

扭矩减小的高精度, 20, 531, 538

扭矩控制, 168, 302

扭矩设置, 971

接

接口 TraceTool — 跟踪 DB, 923

接口 X1, 883

接口 X3:, 883

接线 — 外部编码器, 110

接线 — 测量输入, 108

接线 — 硬件限位开关, 154, 223, 288

接近开关 — 回原点, 42

接通电源, 782

接通电源和 CPU 存储器复位, 888

控

控制, 28

控制 — 动态控制器数据 (Dynamic controller data) 选项卡, 168, 236, 302

控制 — 摩擦补偿 (Friction compensation) 选项卡, 169, 237, 303

控制 — 附加补偿功能 (Additional compensation functions) 选项卡, 238

控制 — 静态控制器数据 (Static controller data) 选项卡, 166, 235, 300

控制信号补偿 — 限制, 541

控制器 — 调节, 901

控制轴, 933

控制面板 — 引言, 927

控制面板 — 控制轴, 933

控制面板 — 监视轴值, 930

控制面板 — 采用控制优先级, 932

推

推断, 20

推断时间, 163, 232, 297, 390

插

插入凸轮, 339, 342, 640

插入凸轮轨迹, 364
插入同步轴, 261
插入外部编码器, 375
插入测量输入, 371
插入电气轴, 134
插入输出凸轮, 359
插补 — CamEdit, 344
插补 — 校正选项, 344
插补 — 重叠区段, 344
插补 2 — CamEdit, 346
插补 2 — 主范围, 346
插补器周期, 412, 897
插补器周期 — 时序, 401
插补器周期 2, 897
插补器周期 2 — 时序, 401
插补条件, 78
插补点表, 342, 640
插补类型, 78, 81

摩

摩擦补偿, 169, 237, 303

收

收集更改, 894

数

数据记录转换, 45

断

断开电源, 782
断电和重启后工艺功能的响应, 782

新

新命令 — 活动的单个命令 (1), 771
新命令 — 活动的单个命令 (2), 773
新命令 — 活动的单个命令 (3), 775
新命令 — 活动的多个命令 (1), 776
新命令 — 活动的多个命令 (2), 778

方

方向阀 — 数字, 35
方向阀 — 比例, 35

无

无法复制工艺对象。 , 965
无限轴 — 模数轴, 34

时

时序, 401
时序 — CPU31xT, 401
时钟 — 时序, 401
时间常量, 168, 302
时间戳, 965
时间片, 401

更

更换 CPU — 绝对编码器调整数据的备份, 728
更改数据记录, 142, 212
更新工艺 DB, 401, 829

替

替换 T-CPU, 946

有

有效方向 — 输出凸轮, 94

未

未回原点的轴的运动, 185, 255, 320

机

机械 — 外部编码器, 387

机械 — 液压轴, 219

机械 — 电气轴, 148, 282

机械齿轮传动, 58

检

检查系统任务生成的 CPU 负载。 , 900

检查集成工艺上的负载, 900

模

模块上的存储空间不足, 966

模式 0 示例 — MC_MoveAbsolute, 484

模式 1 示例 — MC_MoveAbsolute, 486

模式 2 示例 — MC_MoveAbsolute, 487

模式 6 示例 — MC_Home, 459

模式 7 示例 — MC_Home, 460

模拟绝对编码器 — 编码器参数, 202

模数周期, 784

模数轴, 34, 481

模数轴 — 绝对位置, 784

模数轴的绝对位置, 784

模数长度与凸轮轨迹长度, 106

比

比例方向阀, 35

比例方向阀的阀控曲线, 206

油

油缸 — 液压, 35

沿

沿触发的驱动器启用功能, 448

活

活动命令, 422

活动的单个命令 1 — 新命令, 771

活动的单个命令 2 — 新命令, 773

活动的单个命令 3 — 新命令, 775

活动的多个命令 1 — 新命令, 776

活动的多个命令 2 — 新命令, 778

活动范围 — 测量输入, 109

流

流量 Q, 35

测

测量传感器 — 工艺 DB, 813

测量元素, 691

测量结果, 373

测量范围, 109, 691

测量范围 — 测量输入, 108, 109

测量输入 — 互连, 108

测量输入 — 功能, 108

测量输入 — 工艺对象, 107

测量输入 — 接线, 108

测量输入 — 活动范围, 109

测量输入 — 测量范围, 108, 109

测量输入 — 组态, 373

测量输入工艺对象, 107

消

消息帧类型 — 总览, 122

液

液压泵, 35

液压系统油箱, 35

液压轴, 35, 541

液压轴 — 确定并添加一个阀控曲线, 206

液压轴示例 — MC_Power, 448

添

添加数据记录, 142, 212

添加数据记录, 用于进行数据记录转换, 142, 212

添加阀控曲线, 206

清

清除轴启用, 933

溢

溢出 — 工艺 DB 更新, 897

滞

滞后, 96

滞后范围, 96

激

激活更改, 894

灵

灵活的机器概念, 19

熟

熟悉的 SIMATIC 领域, 21

特

特征, 944

特殊操作模式, 34

状

状态标记监视, 972

状态消息, 31

用

用 FB404 MC_Stop 来停止轴, 464

用户程序 — 跟踪值, 923

用途

手册, 3

监

监视 — 硬件限位开关, 154, 223, 288

监视 — 软件限位开关, 156, 225, 290

监视功能 — 停止信号 (Standstill signal) 选项卡, 189, 259, 325

监视功能 — 停止监视, 187, 257, 323

监视功能 — 同步操作, 321

监视功能 — 同步操作监视 (Synchronous operation monitoring) 选项卡, 325

监视功能 — 定位和停止监视 (Positioning and standstill monitoring) 选项卡, 187, 257, 323

监视功能 — 定位监视, 187, 257, 323

监视功能 — 总览, 186, 256, 321

监视功能 — 跟随误差监视 (Following error monitoring) 选项卡, 188, 258, 324

监视功能 — 速度监视 (Velocity monitoring) 选项卡, 190, 确认, 425, 426, 427, 428

260, 327

监视功能 (外部编码器), 972

监视活动命令, 422

监视系统变量, 921

监视表 — 创建, 921

监视表 — 监视系统变量, 921

直

直接回原点/设置回原点位置, 185, 255, 320

相

相位偏移 — 齿轮传动, 545

相对/绝对同步, 71

相对同步, 64, 69

与引导轴, 64

与跟随轴, 64

用于引导轴和跟随轴, 64

相对定位的示例 — MC_MoveSuperImposed, 514, 543

相对齿轮传动, 62

真

真实轴, 38

真实轴和虚拟轴, 38

硬

硬件组件, 27

硬件组件及其用途, 27

硬件限位开关 — 接线, 154, 223, 288

硬件限制, 157, 226, 291

确

确定并添加一个阀控曲线, 206

示

示例 — MC_CamInSuperImposed — 叠加, 602

示例 — MC_CamOut, 596

示例 — MC_ExternalEncoder, 700

示例 — MC_GearInSuperImposed — 相对绝对, 567

示例 — MC_GearOut, 558

示例 — MC_Halt, 472

示例 — MC_MeasuringInput, 693

示例 — MC_MoveAdditive, 507

示例 — MC_MoveToEndPos, 532

示例 — MC_Phasing, 618

示例 — MC_Power, 447

示例 — MC_ReadPeriphery, 735

示例 — MC_Reset, 447

示例 — MC_SetTorqueLimit, 538

示例 — MC_Stop, 466

示例 — MC_WriteParameter, 726

示例 — MC_WritePeriphery, 741

示例 — 使用目标传感器定位, 937

示例 — 夹具进给, 941

示例 — 电子凸轮控制, 99

示例 — 飞剪, 939

示例: 单方向的输出凸轮 — MC_CamSwitch, 663

示例模式 0 — MC_MoveRelative, 495

示例模式 0 — MC_MoveVelocity, 523

示例模式 1 — MC_MoveRelative, 497

示例模式 1 — MC_MoveVelocity, 524

示例模式 2 — MC_MoveRelative, 498

离

离线, 965

移

移至固定终点挡块, 159, 228, 293

立

立即, 894

立即类型, 894

符

符合 VDI 的工作范围, 356

符合 VDI 的运动法则, 356

符合 VDI 的运动转换, 356

等

等时, 27

等时模式, 19

类

类型 3311, 965

类型重启, 894

精

精细粒度的扭矩减小, 531, 538

系

系统变量, 31

系统号, 373

系统数据块, 965

系统时钟, 897, 900

紧

紧凑型和集成式, 21

紧固件 — MC_MoveToEndPos, 528

紧急停止, 918

线

线性轴, 34

组

组态 — 凸轮轨迹, 366

组态 — 外部编码器, 386

组态 — 总览, 113

组态 — 测量输入, 373

组态 — 轴, 147, 218, 281

组态 — 输出凸轮, 361

组态凸轮盘, 339

组态凸轮轨迹, 363

组态叠加同步, 264

组态同步轴, 134

组态外部编码器, 375

组态工具, 28

组态工艺对象, 130

组态数据, 31, 886, 888, 889, 890, 893, 894

组态液压轴 — 插入轴, 191

组态液压轴 — 通过 IM 174/ADI4 的 Q 输出, 196

组态液压轴 — 通过模拟输出模块的 Q 输出, 202

组态电气/同步轴, 134

组态电气轴, 134

组态输出凸轮, 359

终

终止同步操作, 556

终止消息, 31

绝

绝对值, 157, 226, 291
绝对凸轮传动和相对凸轮传动, 64
绝对同步, 69
绝对定位的示例 — MC_MoveSuperImposed, 513
绝对实际值的倍增因子, 196
绝对编码器 — 外部编码器, 111
绝对编码器调整, 44
绝对编码器调整数据的备份, 728
绝对齿轮传动, 62

编

编码器监视功能, 972
编码器系统, 373
编码器转换示例 — MC_ChangeDataset, 478
编码器限制频率监视, 972
编码器零标记 — 回原点, 42
编码器零标记监视, 972
编程 PC, 883
编程模型, 416
编程设备, 883
编辑较旧的项目, 963

缩

缩放, 347
缩放 — 定义凸轮盘, 76
缩放和移位, 347
缩放因子, 78

虚

虚拟轴, 38
虚拟轴 — 响应, 429
虚拟轴的响应, 429

行

行程范围, 154, 223, 288

被

被动回原点, 178, 248, 313, 393
被动回原点示例 — MC_Home, 458

装

装载 — 时序, 401

计

计算位置控制回路的等效时间常数, 918

设

设定值叠加, 52
设定值耦合, 263
设置倍率, 726
设置停止信号, 386
设置回原点位置, 185, 255, 320
设置实际值, 42
设置工艺系统时钟, 897

访

访问 CPU 31xT 的地址空间, 418
访问 MICROBOX T 的地址空间, 420

识

识别固件版本, 948

调

调节 — 位置控制器, 901

调节变量叠加, 55

调节平衡滤波器, 913

调试工具, 28

贝

贝塞尔样条函数, 78

起

起始位置, 93, 104

超

超时, 966

超时后, CPU 偶尔会跳转到 STOP, 966

超驰响应, 452

跟

跟踪, 31, 829, 923

跟踪 — 工艺 DB, 825

跟踪 CPU 用户程序的值, 923

跟踪控制器数据, 924

跟进模式, 464

跟随对象, 58

跟随误差, 528

跟随误差的动态监视, 188, 258, 324

跟随误差监视, 188, 258, 324

跟随轴 — 齿轮传动, 545

转

转换, 963

转换工艺, 952

软

软件组件, 31

软件限位开关, 156, 225, 290

软件限制, 157, 226, 291

轴

轴 — 位置和速度, 153, 222, 287

轴 — 组态, 147, 218, 281

轴参考位置, 105

轴向导, 134

轴和同步操作工艺对象的相互作用, 58

轴控制面板 — 启动, 928

轴控制面板 — 布局, 929

轴控制面板的布局, 929

轴类型, 34

输

输入 — 集成, 27

输出 — 集成, 27

输出凸轮 — MC_CamSwitchTime, 667

输出凸轮 — MC-CamSwitch, 659

输出凸轮 — 切换输出凸轮, 92

输出凸轮 — 反向输出, 91, 103

输出凸轮 — 响应, 94

输出凸轮 — 基于位置的凸轮, 91, 103

输出凸轮 — 基于时间的凸轮, 93, 104, 105

输出凸轮 — 工艺 DB, 816

输出凸轮 — 工艺对象, 89

输出凸轮 — 微分作用时间, 98

输出凸轮 — 有效方向, 94

输出凸轮 — 滞后, 96

输出凸轮 — 示例, 99

输出凸轮 — 组态, 361

输出凸轮切换点, 98

输出凸轮切换点的基于时间的偏移, 98

输出凸轮工艺对象, 89

输出凸轮数据 — 凸轮轨迹, 368

输出凸轮组态, 363

过

过滤器 — 实际值记录, 145, 215

运

运行 — 停止, 782

连

连接编程设备, 883

连续功能, 504

连续性检查, 78, 344

退

退回 — 手动, 154, 223, 288

退回 — 通过驱动器, 154, 223, 288

适

适用范围
手册, 3

选

选择消息帧类型, 122

通

通过同步操作创建轴, 261

通过调用 FB 415 MC_MoveToEndPos 移动到固定挡块/
紧固件, 528

通过调用 FB 439, 541

速

速度倍率, 726

速度控制器, 168, 302

速度控制模式 — SpeedMode, 518

速度控制轴 — 工艺 DB, 785

速度控制轴 — 工艺对象, 46

速度曲线, 151, 220, 285

速度误差监视, 190, 260, 327

逻

逻辑运算 — 输出凸轮, 361

采

采用控制优先级 — 控制面板, 932

重

重启, 782, 889

重启工艺对象, 889

重新创建工艺 DB, 957

重新转换, 963

重组织, 965

错

错误消息, 31

阀

阀受控变量, 206

阀控曲线, 35

阻

阻止该轴的启动, 470

附

附加补偿功能, 238

限

限制 —, 153, 222, 287

限制 — 动态响应 (Dynamic response) 选项卡, 157, 226, 预设的速度, 518

291

限制 — 固定终点挡板 (Fixed end stop) 选项卡, 159, 228, 293

限制值监视 — 实际值记录, 145, 215

限制频率, 972

集

集成 I/O, 27, 31, 418, 420

集成工艺, 28

集成工艺的 I/O 映像区, 418, 420

集成工艺的存储器组织, 886

集成工艺的存储概念, 886

零

零标记 — 编码器零标记, 42

需

需要的基本知识, 3

静

静态控制器数据, 235

静态控制器数据 — 基础, 166, 300

静态摩擦补偿, 169, 237, 303

非

非循环创建凸轮轨迹, 106

顺

顺序模型, 416

顺序模型和编程模型, 416

预

驱

驱动器, 118

驱动器 — 总览, 122

驱动器上的增量编码器 — 编码器参数, 134

高

高精度, 144

高精度插补器, 166, 300

高速输出凸轮, 363

默

默认 —, 151, 220, 285

默认值 — S7T Config, 967

默认值 — 凸轮轨迹, 368

默认值 — 外部编码器, 383

齿

齿轮传动, 62