

使用 Unity Pro 的 Premium 和 Atrium

SERCOS® 运动的运动控制
用户手册

05/2010

本档中提供的信息包含有关此处所涉及产品的性能的一般说明和 / 或技术特性。本档并非用于（也不代替）确定这些产品针对特定用户应用的适用性或可靠性。任何此类用户或集成者都有责任就相关特定应用或使用方面对产品执行适当且完整的风险分析、评估和测试。Schneider Electric 或是其任何附属机构或子公司都不对误用此处包含的信息而承担责任。如果您有关于改进或更正此出版物的任何建议，或者从中发现错误，请通知我们。

未经 Schneider Electric 明确书面许可，不得以任何形式、通过任何电子或机械手段（包括影印）复制本档的任何部分。

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

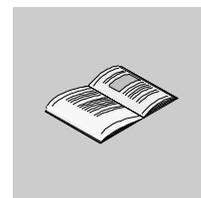
当设备用于具有技术安全要求的应用时，必须遵守有关的使用说明。

如果在我们的硬件产品上不正确地使用 Schneider Electric 软件或认可的软件，则可能导致人身伤害、损害或不正确的操作结果。

不遵守此信息可能导致人身伤害或设备损坏。

© 2010 Schneider Electric。保留所有权利。

目录



安全信息	11
关于本书	13
部分 I Premium PLC 中的 SERCOS(r) 软件包	15
章 1 SERCOS(r) 架构	17
简介	18
SERCOS(r) 架构简介	20
TSX CSY •4 模块的开发	22
章 2 实现方法	25
简介	26
实施独立轴	28
协同轴组或从轴组的实现	30
安装阶段概述	31
章 3 快速入门	33
示例描述	34
轨道描述	36
手动模式中的命令描述	37
要求和方法	38
变速控制器配置	39
TSX CSY 84 模块配置	40
创建 IODDT 和项目变量	41
离散量输入 / 输出模块的符号表示	42
模块 TSX CSY 84 的 IODDT 描述	43
初步处理的编程	47
SFC 编程	48
转换编程	49
动作编程	52
后续处理的编程	56
程序传输	59
调整参数	60
调试模式简介	61

部分 II	模块 TSX CSY 84/164 的硬件安装	63
章 4	TSX CSY 84 / 164 模块介绍	65
	模块的环境简介	66
	模块的物理演示	68
章 5	安装模块	71
	在 PLC 工作站机架中安装模块	72
	在 PLC 工作站中安装模块	73
	由 TSX CSY 84 或 CSY 164 工作站管理的应用专用通道数	74
	安装注意事项	75
	模块显示	76
	发生内部错误后的模块初始化	80
	TSX CSY 84 和 TSX CSY 164 模块操作模式	81
章 6	多轴控制系统的描述	83
	SERCOS(r) Premium PLC 的多轴控制系统	84
	SERCOS(r) 环路网络	86
章 7	光缆	89
	预装配光缆	90
	用于制作自定义光缆的工具箱	91
章 8	特性、标准和工作条件	93
	模块的特性	94
	SERCOS(r) 网络的特性	95
	标准和工作条件	96
章 9	兼容伺服驱动器	97
	变速控制器列表	97
部分 III	TSX CSY 84 和 TSX CSY 164 模块通用的运动控制	99
章 10	功能描述	101
	模块功能	102
	"SERCOS(r)" 功能	104
	"实轴" 功能	105
	"虚轴" 功能	106
	"远程轴" 功能	107
	组的概念	108
	"协同轴组" 功能	109
	"从轴组" 功能	111
	"凸轮配置文件" 功能	113
章 11	配置	115
	关联 IODDT	116
	模块配置	117
	通道配置	120
	SERCOS(r) 功能的配置 (通道 0)	121

配置独立轴（通道 1 到 12）	122
外部测量轴的配置（通道 13 到 16）	126
协同轴组的配置（通道 17 到 20）	128
从轴组的配置（通道 21 到 24）	129
凸轮配置文件的配置（通道 25 到 31）	132
章 12 TSX CSY 84 和 TSX CSY 164 模块编程	135
12.1 运动状态位	136
RAMPING 位 (%Ir.m.c.0)	138
STEADY 位 (%Ir.m.c.1)	139
STOPPING 位 (%Ir.m.c.2)	140
PROFILE_END 位 (%Ir.m.c.3)	141
IN_POSITION 位 (%Ir.m.c.4)	142
AXIS_HOMING 位 (%Ir.m.c.5)	143
AXIS_HOMED 位 (%Ir.m.c.6)	144
AXIS_NOT_FOLLOWING 位 (%Ir.m.c.7)	145
HOLDING 位 (%Ir.m.c.8)	146
RESUMING 位 (%Ir.m.c.9)	147
DRIVE_ENABLED 位 (%Ir.m.c.10)	148
DRIVE_DIAG 位 (%Ir.m.c.11)	149
DRIVE_WARNING 位 (%Ir.m.c.12)	150
DRIVE_FAULT 位 (%Ir.m.c.13)	151
DRIVE_DISABLED 位 (%Ir.m.c.14)	152
AXIS_SUMMARY_FLT 位 (%Ir.m.c.15)	153
AXIS_COMM_OK 位 (%Ir.m.c.16)	154
AXIS_IS_LINKED 位 (%Ir.m.c.17)	155
AXIS_IN_CMD 位 (%Ir.m.c.18)	156
AXIS_AT_TARGET 位 (%Ir.m.c.20)	157
AXIS_POS_LIMIT 位 (%Ir.m.c.21)	158
AXIS_NEG_LIMIT 位 (%Ir.m.c.22)	159
AXIS_WARNING 位 (%Ir.m.c.23)	160
BIAS_REMAIN 位 (%Ir.m.c.24)	161
AXIS_MANUAL_MODE BIT (%Ir.m.c.25)	162
DRIVE_REALTIME_BIT1 位 (%Ir.m.c.26)	163
DRIVE_REALTIME_BIT2 位 (%Ir.m.c.27)	164
AXIS_HOLD 位 (%Ir.m.c.28)	165
AXIS_HALT 位 (%Ir.m.c.29)	166
AXIS_FASTSTOP 位 (%Ir.m.c.30)	167
AXIS_READY 位 (%Ir.m.c.31)	168
CONF_OK 位 (%Ir.m.c.32)	169

12.2	运动控制功能	170
	运动控制功能: 一般信息	171
	ALLOW 位和 CONTROL 位的管理	172
	可在跳变沿上修改的运动控制位	173
	状态更改时可修改的控制位	174
	状态位和字	175
	手动模式	176
	Set_Functional_Mode 功能	178
	回归功能	179
	参考点与回归功能	181
	ForcedHomed 功能	182
	Unhome 功能	183
	SetPosition 功能	184
12.3	运动功能	185
	关于运动功能	186
	MoveType 数据类型	188
	MoveImmed 功能	189
	MoveQueue 功能	191
	GetMoveQueueLength 功能	193
	GetSpeedOverride 和 SetSpeedOverride 功能	194
	EnableRealTimeCtrlBit 功能	195
12.4	序列停止后的运动功能	196
	功能概述	197
	"MoveImmed" 功能	198
	带有 ABS_MOVE_NO_ROLLOVER 的 "MoveImmed" 功能	200
	"MoveQueue" 功能	203
	带有 ABS_MOVE_NO_ROLLOVER 的 "MoveQueue" 功能	205
12.5	实际速度 / 位置功能	207
	GetActualSpeed 功能	208
	GetCommandSpeed 功能	209
	模数计数器服务	210
	GetUnrolledPosition 功能	211
	GetUnrolledCommandedPosition 功能	212
	GetCommandedPosition 功能	213
12.6	跟随功能	214
	关于跟随功能	215
	"目标序列激活" 功能	217
	"序列激活前的剩余时间估计" 功能	219
	触发器引起的从轴组序列停止 (Unlink)	221
	协同轴组的运动功能与 MMS Quantum 功能兼容	223
	FollowOn 415 功能描述	225

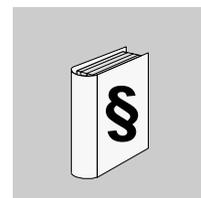
章 13	调整 TSX CSY 84 和 TSX CSY 164 模块	227
13.1	SERCOS(r) 模块调整：一般信息	228
	SERCOS(r) 模块调整屏幕描述：一般信息	229
	调整原理	231
13.2	实轴、虚轴或远程轴的参数调整	232
	SERCOS(r) 模块的调整屏幕描述：实轴或虚轴	233
	SERCOS(r) 模块的调整屏幕描述：远程轴	235
	实轴的 TRF_RECIPE 屏幕描述	236
	TRF_RECIPE	238
	TRF_RECIPE 示例	240
	速度参数：DefaultSpeed	241
	速度参数：SpeedLimit	242
	关于加速度 / 减速度参数	243
	加速度 / 减速度参数：Accel	244
	加速度 / 减速度参数：Decel	245
	加速度 / 减速度参数：AccelMax	246
	加速度 / 减速度参数：DecelMax	247
	加速度 / 减速度参数：AccelType	248
	数据类型 AccelerationType	250
	精度参数：GearRatio	251
	位置限制和模数参数：PositionLimit	253
	关于模数	255
	位置限制和模数参数：RolloverLimit	256
	位置限制和模数参数：EnableRollover	257
	关于正确位置带	258
	位置限制参数：InPositionBand	259
	关于启用模式	260
	位置限制参数：EnableMode	261
	从轴禁用行为	262
13.3	从轴组的参数	263
	SERCOS(r) 模块的调整屏幕描述：从轴	264
	关于从轴组	265
	SlaveMode 数据	268
	从轴组参数：FollowerMode	269
	从轴组参数：FollowerConfig	270
	从轴组参数：FollowerRatio	271
	关于主轴偏移	273
	从轴组参数：MasterOffset	274
	从轴组参数：MasterTriggerPosition	276

	从轴组参数: AbsFollowerBias	278
	从轴组参数: FollowerBias	280
	从轴组中的停止的传播	281
	紧急停止倾斜度的校正功能	282
13.4	凸轮配置文件调整	284
	关于凸轮配置文件	285
	凸轮配置文件参数: 长度	286
	凸轮配置文件参数: LookUpFollowerPosition 功能	287
	凸轮配置文件参数: Coord	288
	凸轮配置文件参数: InterpType	290
	凸轮配置文件参数: ProfileId	291
	凸轮配置文件的 TRF_RECIPe 屏幕描述	292
	TRF_RECIPe 指令	294
13.5	SERCOS(r) 环路参数	296
	SERCOS(r) 模块的调整屏幕描述: SERCOS(r) 环路	297
	光功率调整图表	299
	SERCOS(r) 环路参数: OpticalPower	300
13.6	SERCOS(r) IDN 读 / 写功能	301
	伺服驱动器参数: IDN_S 和 IDN_P	302
	伺服驱动器参数: IDN_US 和 IDN_UP	304
	标准 SERCOS 标识号 (IDN)	306
	自定义电报的 IDN	309
章 14	调试 TSX CSY 84 / 164 模块	311
	调试屏幕描述: 一般信息	312
	调试屏幕用户界面	315
	调试: 启用	316
	调试: 诊断	317
	调试: 运动	319
	调试: 发送命令 (自动模式)	321
	调试: 手动命令 (手动模式)	322
	调试: 参考	323
	调试: 跟随	324
	调试: 驱动器	325
	调试: 位置	326
	模块诊断	327
	通道诊断	328
章 15	诊断和维护	329
15.1	SERCOS(r) 阶段诊断简介	330
	关于 SERCOS(r) 阶段标准	331
	GetActualPhase 功能	332
	GetCommandedPhase 功能	333

15.2	读取环路的实际数据简介	334
	GetSercosAddress 功能	335
	GetNumberOfDrivesInRing 功能	336
	IsLoopUp 功能	337
15.3	轴标识	338
	GetNumberInSet 功能	338
15.4	状态和故障信息	339
	隐式交换对象	340
	可通过 GetMotionFault 命令访问的故障	341
	GetMotionFault 功能	343
	可通过 GetMotionWarning 命令访问的故障	344
	GetMotionWarning 功能	346
15.5	故障和警告	347
	可通过显式读取访问的模块故障	348
	可通过显式读取访问的通道故障	349
	配置和调整错误列表	350
	故障寄存器	352
	WRITE_CMD 命令的错误代码列表	356
	WRITE_CMD 命令：编程错误	357
	WRITE_CMD 命令：通讯错误	359
	WRITE_CMD 命令：系统错误	360
	WRITE_CMD 命令：系统警告	362
	TRF_RECIPE 命令显式写入故障	363
	操作模式	364
	故障的逻辑图	365
	伺服驱动器故障	367
15.6	TSX CSY 84 的性能	369
	TSX CSY 84 的性能	369
章 16	SERCOS 语言对象	373
16.1	SERCOS 模块的语言对象和 IODDT	374
	与 TSX CSY 84 模块关联的 IODDT 简介	375
	与应用专用功能关联的隐式交换语言对象	376
	与应用专用功能关联的显式交换语言对象	377
	使用显式对象管理交换和报告	379
	接口语言	383
	参数管理	385
	WRITE_PARAM 和 READ_PARAM	386
	SAVE_PARAM 和 RESTORE_PARAM	387
	WRITE_CMD	388
	WRITE_CMD 示例	390
	READ_STS	392
	TRF_RECIPE 和 MOD_PARAM	393

16.2	专门与 SERCOS 模块关联的语言对象和 IODDT	394
	有关 T_CSY_CMD 类型的 IODDT 隐式交换对象的详细信息	395
	有关 T_CSY_CMD 类型的 IODDT 显式交换对象的详细信息	397
	有关 T_CSY_RING 类型的 IODDT 隐式交换对象的详细信息	399
	有关 T_CSY_RING 类型的 IODDT 显式交换对象的详细信息	401
	有关 T_CSY_TRF 类型的 IODDT 显式交换对象的详细信息	404
	有关 T_CSY_IND 类型的 IODDT 隐式交换对象的详细信息	406
	有关 T_CSY_IND 类型的 IODDT 显式交换对象的详细信息	410
	有关 T_CSY_FOLLOW 类型的 IODDT 隐式交换对象的详细信息	414
	有关 T_CSY_FOLLOW 类型的 IODDT 显式交换对象的详细信息	417
	有关 T_CSY_COORD 类型的 IODDT 隐式交换对象的详细信息	424
	有关 T_CSY_COORD 类型的 IODDT 显式交换对象的详细信息	426
	有关 T_CSY_CAM 类型的 IODDT 隐式交换对象的详细信息	429
	有关 T_CSY_CAM 类型的 IODDT 显式交换对象的详细信息	430
16.3	适用于所有模块的 IODDT 类型的 T_GEN_MOD	433
	类型为 T_GEN_MOD 的 IODDT 的语言对象的详细信息	433
章 17	附录	435
17.1	可访问功能列表	435
	SERCOS(r) 通道 0 的可访问功能	436
	实轴的可访问功能	438
	虚轴的可访问功能	442
	远程轴的可访问功能	445
	协同轴组的可访问功能	447
	从轴组的可访问功能	449
	凸轮配置文件的可访问功能	451
部分 IV	特定于 TSX CSY 164 的功能	453
章 18	兼容性	455
	兼容性	455
章 19	TSX CSY 164 模块的配置	457
	TSX CSY 164 模块的配置	458
	配置通道	459
章 20	特定于 TSX CSY 164 的功能	461
	组的动态重新配置	462
	轴间位置偏差的监控功能	464
章 21	TSX CSY 164 的性能	469
	TSX CSY 164 的性能	469
索引	473

安全信息



重要信息

声明

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



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



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

危險

“危險”表示极可能存在危險，如果不遵守说明，可导致严重的人身伤害甚至死亡。

警告

“警告”表示可能存在危險，如果不遵守说明，可导致严重的人身伤害甚至死亡，或设备损坏。

▲ 注意

“注意”表示可能存在危险，如果不遵守说明，可导致严重的人身伤害或设备损坏。

注意

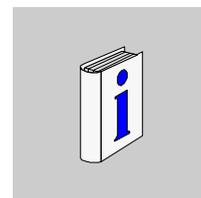
注意(无安全警告符号)，表示存在潜在的危险，如果忽视，可能导致设备损坏。

请注意

电气设备的安装、操作、维修和维护工作仅限于合格人员执行。对于使用本资料所引发的任何后果，Schneider Electric 概不负责。

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

关于本书



概览

文档范围

本手册介绍了如何使用 SERCOS® TSX CSY 84/164 模块在 Premium 和 Atrium PLC 上安装运动控制。

有效性说明

本文档适用于 Unity Pro v5.0。

关于产品的资讯

警告

意外的设备操作

应用此产品要求在控制系统的设计和编程方面有经验。只允许具有此类经验的人士编程、安装、改动和应用此产品。

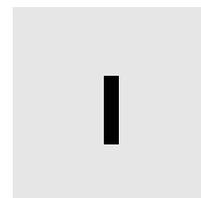
请遵守所有当地和国家 / 地区的安全法规和标准。

如果不遵守这些说明，将会导致死亡、严重伤害或设备损坏。

用户意见

欢迎对本书提出意见。您可以给我们发邮件，我们的邮件地址是 techcomm@schneider-electric.com。

Premium PLC 中的 SERCOS(r) 软件包



本部分主题

本部分介绍 SERCOS(r) 架构，并描述 Premium PLC 中用于实现 SERCOS(r) 运动控制的方法。

注意：本节中，所有与 TSX CSY 84 安装相关的数据也适用于 TSX CSY 164。并且将 TSX CSY 84 作为示例。

本部分包含了哪些内容？

本部分包括以下各章：

章	章节标题	页
1	SERCOS(r) 架构	17
2	实现方法	25
3	快速入门	33

SERCOS(r) 架构



本章主题

本章提供主轴与从轴之间的数字链路的简要介绍，这些链路在标准 EN 61491 "用于命令单元和训练设备之间的实时通讯的串行数据链路" 中定义。

本章包含了哪些内容？

本章包含了以下主题：

主题	页
简介	18
SERCOS(r) 架构简介	20
TSX CSY •4 模块的开发	22

简介

数字链路

轴控模块（主站）与智能伺服驱动器（从站）之间的数字链路是由与工业电器的设备相关的欧洲标准 EN 61491 定义的。

使用分布式架构能够直接将项目 I/O（位置编码器、紧急停止等）连接到伺服驱动器，从而降低接线成本。

使用光纤链路能够进行高速交换（2 或 4 MHz）并确保抗电磁干扰性。

交换的数据

通过数字链路交换的数据有两种类型：

- 由主设备交换到从设备（位置控制等）或由从设备交换到主设备的循环数据（位置测量等）。主站和各个从站之间的循环数据交换限于每个 SERCOS(r) 循环 8 个读对象和 8 个写对象。
- 非循环数据：复杂命令、参数写入或参数读取等。
对于每个循环，这些交换通过 2 个读取模式的保留字节和 2 个写入模式的保留字节进行。因此，交换一个对象（例如为了完成一次参数读取）需要几个循环。

标识交换的对象

所有对象都可使用标识号进行访问：IDN

该标准可以标识 31,768 个对象并指定大约 300 个 IDN（例如，IDN 40 = 速度值）。

所有对象都包含下列字段：名称（最多 64 个字符）、属性、单位、最大值、最小值、值。

操作模式

总线操作模式包含以下 5 个阶段：

在启动时

阶段	操作模式
阶段 0	总线环路测试。伺服驱动器处于中继器模式。
阶段 1	确定总线上的从站。
阶段 2	伺服驱动器系统配置。
阶段 3	循环交换编程。 伺服驱动器参数设置。

在正常操作中

阶段	操作模式
阶段 4	活动循环交换。

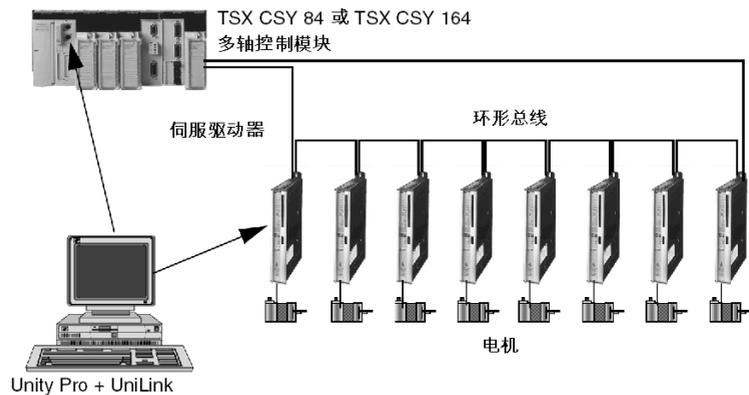
由于每个伺服驱动器作为总线上的中继器，因此，系统在电源断电、通讯错误、伺服驱动器之一发生故障或总线的电源连接断开时，将切换到阶段 0。

注意：某些参数 (IDN) 只能在阶段 3 中访问（请参见标准 EN 61491）。以下功能 GetActualPhase、GetCommandedPhase 和 SetCommandedPhase 指示活动阶段并在阶段 3 中设置活动阶段。

SERCOS(r) 架构简介

概述

SERCOS(r) 架构的概述如下：



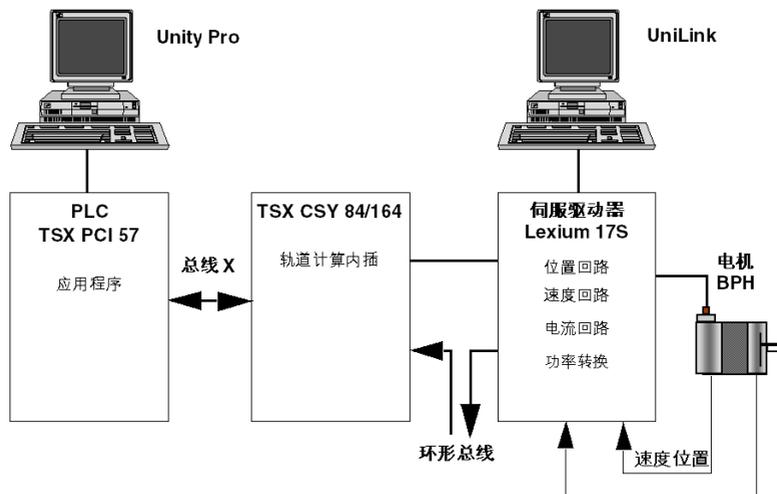
Premium 包简介

Premium 包由以下部分组成：

- Premium/Atrium PLC 系列
- 轴控模块 TSX CSY 84，可通过总线环路控制多达 8 个伺服驱动器
- 轴控模块 TSX CSY 164，可通过总线环路控制多达 16 个伺服驱动器
- Lexium 17S 伺服驱动器系列（4 A 至 56 A）：LXM 17S
- 电机系列：BPH...
- 用于配置 TSX CSY 84/164 模块和对运动项目编程的 Unity Proware
- 可用于输入参数和调整伺服驱动器的 UniLink 软件

功能图

下图显示由运动控制器和伺服驱动器执行的不同功能：



注意：可以使用非 Premium/Atrium 系列的伺服驱动器。在此情况下，它将由自己的配置软件配置，而不由 UniLink 配置。

各个模块执行的功能

轴控模块 TSX CSY 84/164 执行轨道计算并内插若干个轴。

伺服驱动器控制位置、速度和扭矩回路。它还提供电源转换以控制电机。编码器信息（当前位置、当前速度）发送到伺服驱动器。

PLC 与运动控制器之间的交换通过位于机架底部的总线 X 进行。

运动控制器与伺服驱动器之间的交换通过 SERCOS(r) 环路总线进行。

TSX CSY •4 模块的开发

概览

模块的版本显示在:

- 位于模块侧面的标签上
- 在线模式下调试屏幕的模块区域中

	CSY 84 V1.1	CSY 84 V1.2	CSY 84 V1.3	CSY 84 V1.2	CSY 164 V1.0	CSY 164 V1.3	CSY 164 V1.6
通过递增编码器或 Lexium 17S 和 17HP 伺服驱动器上的 SSI 关闭位置回路的选项 (1) (无文档记录)	X	X	X	X	X	X	X
将命令位置切换为扭矩的选项 (1) 和 (2) (无文档记录)	X	X	X	X	X	X	X
将命令位置切换为速度命令的选项 (1) 和 (2) (无文档记录)	X	X	X	X	X	X	X
读取或写入伺服驱动器参数	X	X	X	X	X	X	X
带外部编码器的位置回路 (1)	X	X	X	X	X	X	X
扭矩模式 (1)	X	X	X	X	X	X	X
速度模式 (1)	X	X	X	X	X	X	X
手动模式 (4)		X	X	X	X	X	X
发生故障或独立轴被禁用 (FREEWHEEL_STOP) 时的行为 (3)			X	X	X	X	X
从轴组中的停止的传播			X	X	X	X	X
紧急停止倾斜度的校正功能			X	X	X	X	X
锁存命令后的停止行为选择			X	X	X	X	X
从轴组的成员轴在 Unlink 后在绝对位置的停止			X	X		X	X
从轴组的成员轴在绝对位置的停止			X	X		X	X
“目标序列激活”功能			X	X		X	X
模数计数器服务			X	X		X	X
切换 MOVE 中的 ACC 和 DEC 参数			X	X		X	X
触发器引起的从轴组序列停止 (Unlink)			X	X		X	X
单独通道的动态配置				X	X	X	X
组的动态重新配置				X	X	X	X
轴之间的偏差的监控功能				X	X	X	X

	CSY 84 V1.1	CSY 84 V1.2	CSY 84 V1.3	CSY 84 V1.2	CSY 164 V1.0	CSY 164 V1.3	CSY 164 V1.6
独立轴的回转限制以上的绝对运动				X			X
与 MMS Quantum 兼容的协同轴的运动功能				X			X
功能 GetPositionCounter 和 SetPositionCounter				X			X
实轴的新调整功能: EnableDriveWarning 和 DisableDriveWarning				X			X

- (1) 要求修改 TSX CSY 84/164 (Sercos.cfg) 模块的文件。
- (2) 要求最低版本的 Lexium 驱动器 (MHDS xxxxN 00 ≥ SV 1.3, MHDA xxxx ≥ SV 1.4), 对 Lexium HP 和 Lexium 选件 AS 伺服驱动器无限制。
- (3) 要求最低版本的 Lexium MHDx 5.51 伺服驱动器。
要求 Unity V2.1 或更高版本才能访问调试屏幕。

实现方法

2

本章主题

本章描述用于实现独立轴或内插轴的运动的全局方法。

本章包含了哪些内容？

本章包含了以下主题：

主题	页
简介	26
实施独立轴	28
协同轴组或从轴组的实现	30
安装阶段概述	31

简介

TSX CSY 84 模块功能

TSX CSY 84 可用于执行以下功能：

- 连接到伺服驱动器的 8 个实轴（通道 1 至 8），
- 4 个虚轴（通道 9 至 12），
- 4 个带远程输入的轴（通道 13 至 16），
- 4 组协同轴（通道 17 至 20）。每个组可用于 2 至 8 个轴之间的线性内插，
- 4 组从轴（通道 21 至 24）。每个组最多可以包含 7 个轴：1 个主轴和 6 个从轴。

注意：TSX CSY 84 模块还提供 7 个凸轮配置文件（通道 25 至 31）。

TSX CSY 164 模块功能

TSX CSY 164 可用于执行以下功能：

- 16 个独立轴（通道 1 至 16），
通道 1 至 16 可分别支持一个实轴功能、一个虚轴功能或一个外部设定点；
- 4 组协同轴（通道 17 至 20）。每个组可用于 2 至 8 个轴之间的线性内插，
- 4 组从轴（通道 21 至 24）。每个组最多可以包含 7 个轴：1 个主轴和 6 个从轴。
- 7 个凸轮配置文件（通道 25 至 31）。

注意：TSX CSY 84/164 模块还提供 7 个凸轮配置文件（通道 25 至 31）。

实轴

实轴是物理轴，它通过 SERCOS(r) 环路总线控制伺服驱动器。

虚轴

虚轴不是物理轴。但是可将它用于协调多个物理轴的运动。例如，虚轴可以是从轴组中的主轴。

虚轴还可以在调整或调试阶段用于模拟实主轴（与过程无关）。

远程轴

远程轴可用于将远程位置数据项上载到模块。通常，TSX CSY 84/164 模块必须在由外部机制控制并连接到伺服驱动器的辅助位置输出的编码器中占据一个序列位置。

协同轴组

协同轴组包含相互配合运动的轴。该集合中的某个轴（定义为协同主轴）作为该集合的运动的的速度参考。

将计算其他协同轴的加速度和速度，以便所有轴同时运动。

从轴组

从轴组包含一个主轴和多个从轴，从轴跟随主轴的运动。存在两种跟随主轴的方式：

- 比率模式：每个从轴根据已在配置中定义的关系（称为随动关系）跟随主轴。例如，从轴的位置等于主轴的位置乘以比率 x ，
- 凸轮模式：从轴按照凸轮配置文件跟随主轴。

凸轮配置文件可用于实现电子凸轮，以便简化对复杂运动进行编程的任务。点表用于根据主轴位置定义从轴位置。

实施独立轴

简介

独立轴可以是连接到伺服驱动器的实轴、虚轴或远程轴。
从轴组或协同轴组包含独立轴（实轴、虚轴或远程轴）的集合。

实轴的实现方法

能够实现实轴之前，必须启用通道 0（所有通道 0 的 ALLOW 位：`%Qr.m.0.18`、`%Qr.m.0.26` 和 `%Qr.m.0.31` 都处于状态 1）。

实轴的实现方法以 3 个阶段进行：

- 阶段 1：使用 UniLink 软件配置伺服驱动器，
- 阶段 2：使用 Unity Pro 配置编辑器（使用的所有轴的模块声明和参数配置）配置 TSX CSY 84 模块，
- 阶段 3：项目程序编写，将此程序传输至 PLC 以及项目调试。

注意：通过 Unilink 软件启用伺服驱动器将禁止向伺服驱动器发送模块命令。因此，必须在退出 Unilink 软件前禁用伺服驱动器。

虚轴的实现方法

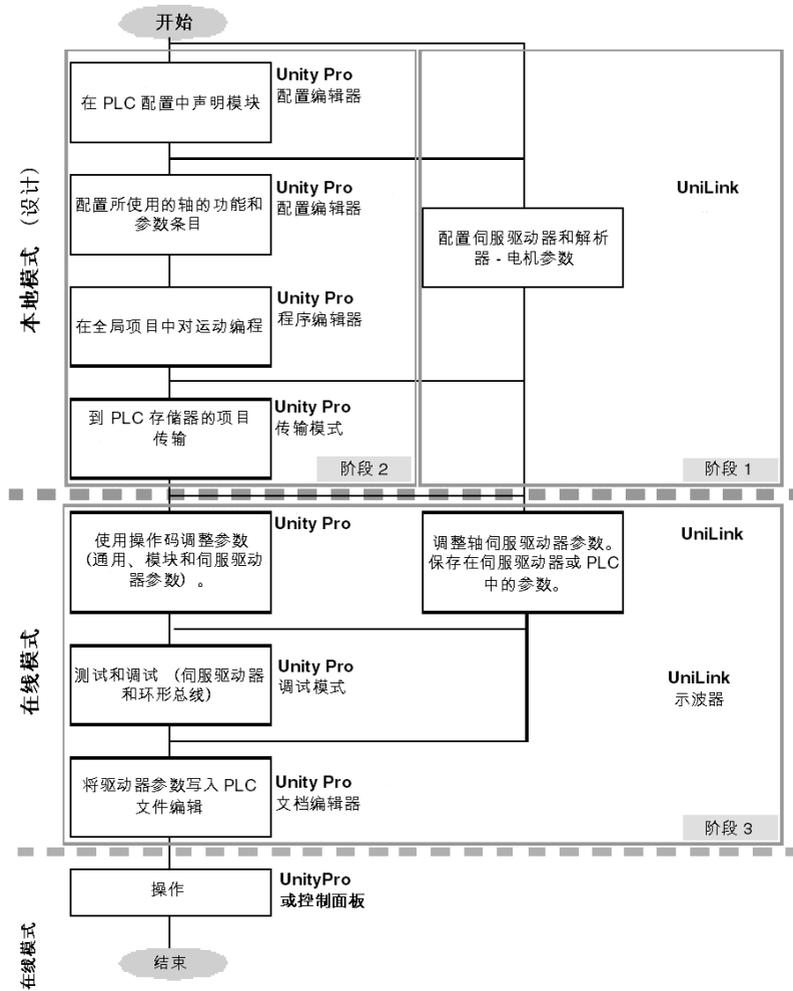
虚轴未连接到伺服驱动器（它不是物理轴）。除了链接到不存在的伺服驱动器的操作外，虚轴的实现方式与实轴相同。

远程输入的实现方法

远程输入所具有的功能远少于其他种类的单独轴。其实现方法与实轴或虚轴的实现方法完全相同，其中只需配置位置数据。在此情况下，链接到伺服驱动器或运动编程的所有操作都不存在。

实轴的实现方法

概述



协同轴组或从轴组的实现

协同轴组

协同轴是实轴或虚轴。可通过首先实现实轴（前几页描述的方法），然后实现该组，从而实现协同轴组。

从轴组

在从轴组中，主轴可以是实轴、虚轴或外部轴，从轴可以是实轴或虚轴。实现从轴组首先必须按照前几页描述的方法实现构成此组的各个轴，然后实现组。

安装阶段概述

简介

应用专用模块的软件安装是在以下模式下通过不同的 Unity Pro 编辑器完成的：

- 离线模式
- 在线模式

如果没有可以连接的处理器，Unity Pro 允许您使用仿真器执行初始测试。在这种情况下，安装 (参见第 32 页) 有所不同。

建议采用下面的安装阶段顺序，但可以更改某些阶段的顺序（例如，从配置阶段开始）。

存在处理器情况下的安装阶段

下表说明存在处理器情况下的各个安装阶段：

阶段	描述	模式
变量声明	应用专用模块的 IODDT 类型变量和项目变量的声明。	离线 (1)
编程	项目编程。	离线 (1)
配置	模块声明。	离线
	模块通道配置。	
	输入配置参数。	
关联	IODDT 与已配置通道的关联（变量编辑器）。	离线 (1)
生成	项目生成（链路的分析和编辑）。	离线
传输	将项目传输到 PLC。	在线
调整 / 调试	从调试屏幕、动态数据表进行项目调试。	在线
	修改程序和调整参数。	
文档	生成文档文件以及打印与项目相关的其他信息。	在线 (1)
操作 / 诊断	显示项目的监督控制所必需的其他信息。	在线
	项目和模块的诊断。	
说明：		
(1)	还可以在其他模式中执行这些阶段。	

针对仿真器的实施阶段

下表显示了针对仿真器的各安装阶段。

阶段	描述	模式
变量声明	应用专用模块的 IODDT 类型变量和项目变量的声明。	离线 (1)
编程	项目编程。	离线 (1)
配置	模块声明。	离线
	模块通道配置。	
	输入配置参数。	
关联	IODDT 与已配置模块的关联（变量编辑器）。	离线 (1)
生成	项目生成（链路的分析和编辑）。	离线
传输	将项目传输到仿真器。	在线
仿真	不带输入 / 输出的程序仿真。	在线
调整 / 调试	从调试屏幕、动态数据表进行项目调试。	在线
	修改程序和调整参数。	
说明：		
(1)	还可以在其他模式中执行这些阶段。	

注意：仿真器仅用于离散量或模拟量模块。

本章的目标

本章提供使用 Premium PLC 的 SERCOS(r) 包实现运动控制应用的示例。

注意：以 TSX CSY 84 模块为例。用于 TSX CSY 84 配置的所有步骤都与用于 TSX CSY 164 模块的步骤相同。

本章包含了哪些内容？

本章包含了以下主题：

主题	页
示例描述	34
轨道描述	36
手动模式中的命令描述	37
要求和方法	38
变速控制器配置	39
TSX CSY 84 模块配置	40
创建 IODDT 和项目变量	41
离散量输入 / 输出模块的符号表示	42
模块 TSX CSY 84 的 IODDT 描述	43
初步处理的编程	47
SFC 编程	48
转换编程	49
动作编程	52
后续处理的编程	56
程序传输	59
调整参数	60
调试模式简介	61

示例描述

简介

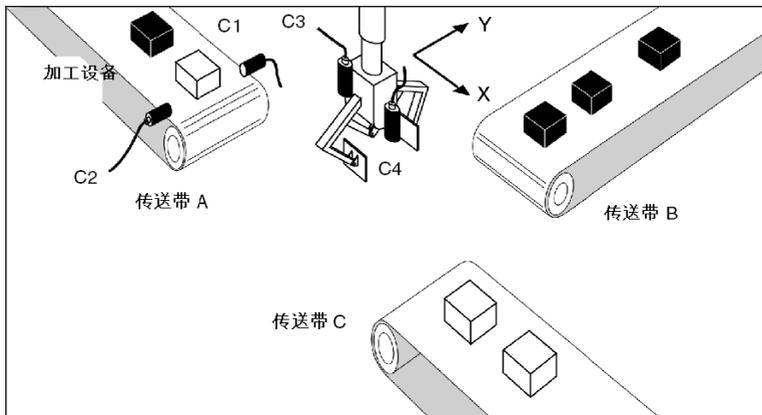
本示例仅用于教学目的。它使您能够理解 TSX CSY 84 轴控模块的每个实现阶段，而不必阅读整个手册。

传输设备

传输设备确保移走加工产品。此设备由一个机械爪构成，它可以在与地面平行的某个轴（X 和 Y 轴）上进行空间移动。

当某个工件出现在移除传送带 A 上时，该机械爪自动定位它，以便根据工件类型将其放在传送带 B 或 C 上。然后，该机械爪返回原位，随时准备在检测到新的加工工件时立即抓取它。

示意图



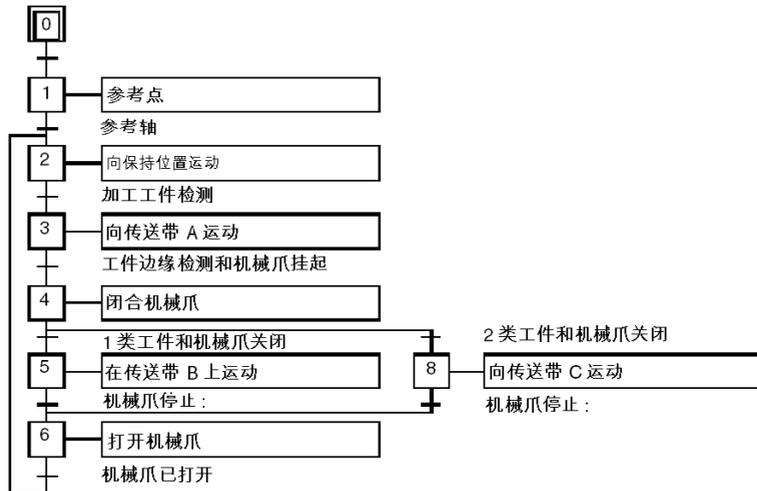
输入 / 输出

输入 / 输出如下：

C1	加工工件检测单元
C2	工件类型识别传感器
C3	机械爪打开 / 关闭检测传感器
C4	工件边沿检测单元（在机械爪中），它连接到连接器事件输入
O/F	机械爪：机械爪打开 / 关闭命令

项目 Grafcet

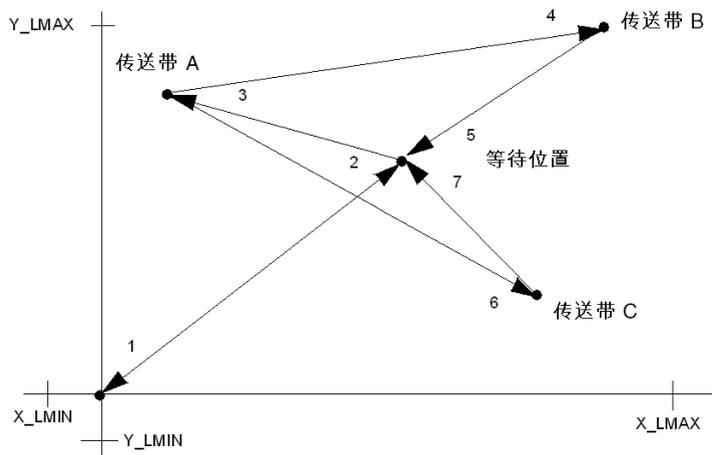
项目 Grafcet 如下:



轨道描述

轨道示意图

下图描述轨道：



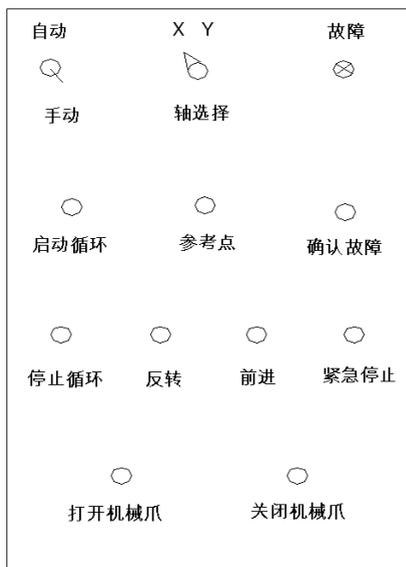
描述

1	以速度 V_{p0} 回归
2	以速度 V_{ret} 运动到等待位置 (X_{att} , Y_{att}), 停止
3	以速度 V_A 运动到 A 传送带 (X_A , Y_A), 直到检测到加工工件
4	以速度 V_B 在传送带 B (X_B , Y_B) 上运动, 停止
6	以速度 V_C 在传送带 C (X_C , Y_C) 上运动, 停止
5, 7	以速度 V_{ret} 运动到等待位置 (X_{att} , Y_{att}), 停止

手动模式中的命令描述

人机界面的前面板

以下命令重新组合在前面板上，以便在发生安装错误时手动控制运动部件。命令和 LED 由离散量输入和输出模块运行



命令描述

- **自动 / 手动**: 操作模式选择器开关
- **启动循环**: 自动执行循环
- **停止循环**: 自动停止循环
- **X / Y 轴选择**: 手动模式中用于控制的轴选择
- **参考点**: 所选轴上的手动参考点
- **正向 / 反转**: 所选轴的正向或负向手动运动控制
- **故障**: 集中指示所有硬件和项目故障的 LED
- **故障确认**: 确认故障存在的命令
- **紧急停止**: 立即停止运动部件（无论选择哪种模式）
- **打开机械爪**: 机械爪打开命令
- **关闭机械爪**: 关闭机械爪的命令

要求和方法

要求

为了将本描述限制到用于轴命令的特定功能，应假设已经执行了以下操作：

- Unity Proware 已安装
- UniLink 软件已安装
- 硬件已安装

方法

TSX CSY 84 模块的实施需要：

- 使用 UniLink 软件对 Lexium 17S 变速控制器进行配置
- 使用 Unity Proware 对 TSX CSY 84 模块进行配置

变速控制器配置

变速控制器连接

访问 UniLink 软件将显示下面的屏幕：



单击 COM1 或 COM2 可连接到变速控制器（这取决于编程终端上用于连接到变速控制器的串行口）。

配置电机和 SERCOS(r) 地址

有关如何配置电机和 SERCOS(r) 地址的信息，请参考 UniLink 软件用户手册。

TSX CSY 84 模块配置

PLC 配置的软件声明

启动 Unity Proware，选择命令 "文件"/"新建"，然后选择 Premium 处理器。

从项目浏览器访问配置编辑器。操作如下：

- 双击选项卡 **X 总线**。

然后选择 PLC 配置中的每个组件。本项目已选择了以下组件：

- 机架 0 和机架 1：TSX RKY 8E
- 处理器：TSX P57 304
- 电源模块：TSX PSY 2600 用于机架 0，TSX PSY 5500 用于机架 1
- 32 路输入模块：位于机架 0 的位置 2 中的 TSX DEY 32D2K
- 32 路输出模块：位于机架 0 的位置 3 中的 TSX DSY 32T2K
- 轴控模块：位于机架 1 的位置 2 中的 TSX CSY 84

处理器循环时间应该长于 8 毫秒（模块循环周期的 2 倍）。

轴配置

双击 TSX CSY 84 模块（机架 1 的位置 2），这将显示配置屏幕。

- 在通道级别区域中选择："实轴 1"，然后选择 "实轴" 功能
- 输入参数值

根据下面用于实轴类型的轴 1 屏幕输入：

限制		单位		运动	
<input checked="" type="checkbox"/> 位置检查		类型	线性	<input type="checkbox"/> 模数	
最大位置	5.000000e+005	位置	毫米	最大模数	0.000000e+000
最小位置	-5.000000e+005	速度	毫米/分	最小模数	0.000000e+000
最高速度	1.200000e+004	加速度	微米/平方秒	正确位置带	1.000000e+005
最大加速度	1.200000e+004	比例系数		加速度	1.000000e+004
最大减速度	1.200000e+004	分子	2.000000e+000	减速度	1.000000e+004
启用位置带		分母	1.000000e+000	加速度类型	矩形 100%
<input type="checkbox"/> 启用					
容差	0.000000e+000				

- 为第二个轴（实轴 2）输入相同参数
- 使用 "编辑"/"确认" 命令或图标确认输入
- 在配置编辑器的主屏幕上，使用 "编辑"/"确认" 命令或图标确认配置

注意：最高速度 (VMax) 针对 6000 转 / 分钟的电机、每转运动 2 毫米而设置。

创建 IODDT 和项目变量

创建和关联 IODDT

我们将创建两个类型为 `T_CSY_IND`（独立轴）的 IODDT，然后将它们与 TSX CSY 84 模块的通道 1 和 2 关联。如果您不知道如何执行此操作，请参考相应的段落。

这两个 IODDT 按如下进行声明和关联：

IODDT	关联的通道	注释
Axis_1	%CH1.2.1	
Axis_2	%CH1.2.2	

内部变量

以下内部变量是在变量编辑器中创建的：

符号	地址	注释
Speed_reached	%MF102	
Man_speed	%MF104	

离散量输入 / 输出模块的符号表示

离散量输入模块符号

离散量输入模块位于机架 0 的插槽 2 中。其符号如下

符号	地址	注释
Sensor_2	%I0.2.1	工件识别传感器（0= 类型 2， 1 = 类型 1）
Sensor_3	%I0.2.2	机械爪打开 / 机械爪关闭传感器
Auto_man	%I0.2.3	模式选择开关（0 = 自动； 1 = 手动）
Start_cycle	%I0.2.4	
Stop_cycle	%I0.2.5	
x_y_selection	%I0.2.6	手动选择要运动或回归的轴（1 = x； 0 = y）
Home_man	%I0.2.7	
前进	%I0.2.8	
反转	%I0.2.9	
Ack_fault	%I0.2.10	
Fault_type	%I0.2.11	
Enable_axes	%I0.2.12	确认轴
Open_grab	%I0.2.13	
Close_grab	%I0.2.14	

输出模块符号

离散量输入模块位于机架 0 的插槽 3 中。其符号如下

符号	地址	注释
Grab	%Q0.3.0	打开 / 关闭机械爪的执行器命令
Fault	%Q0.3.1	故障指示

模块 TSX CSY 84 的 IODDT 描述

概览

我们已声明了两个 IODDT：

- Axis_1：与 TSX CSY 84 的通道 1 关联
- Axis_2：与 TSX CSY 84 的通道 2 关联

关联了这两个 IODDT 后，我们就获得了以下段落中描述的对对应关系。

IODDT Axis_1

此示例中使用的 IODDT 的元素如下：

符号	地址	注释
Axis_1.Ramping	%I1.2.1	
Axis_1.Steady	%I1.2.1.1	
Axis_1.Stopping	%I1.2.1.2	
Axis_1.Profile_end	%I1.2.1.3	
Axis_1.In_position	%I1.2.1.4	
Axis_1.Axis_homing	%I1.2.1.5	
Axis_1.Axis_homed	%I1.2.1.6	
Axis_1.Axis_not_following	%I1.2.1.7	
Axis_1.Holding	%I1.2.1.8	
Axis_1.Resuming	%I1.2.1.9	
Axis_1.Drive_enable	%I1.2.1.10	
Axis_1.Drive_diag	%I1.2.1.11	
Axis_1.Drive_warning	%I1.2.1.12	
Axis_1.Drive_flt	%I1.2.1.13	
Axis_1.Drive_disable	%I1.2.1.14	
Axis_1.Axis_summary_flt	%I1.2.1.15	
Axis_1.Axis_comm_ok	%I1.2.1.16	
Axis_1.Axis_is_linked	%I1.2.1.17	
Axis_1.Axis_in_command	%I1.2.1.18	
Axis_1.Axis_capture	%I1.2.1.19	
Axis_1.Axis_at_target	%I1.2.1.20	
Axis_1.Axis_pos_limit	%I1.2.1.21	
Axis_1.Axis_neg_limit	%I1.2.1.22	
Axis_1.Drive_realtime_bit1	%I1.2.1.26	

符号	地址	注释
Axis_1.Drive_realtime_bit2	%I1.2.1.27	
Axis_1.Axis_hold	%I1.2.1.28	
Axis_1.Axis_halt	%I1.2.1.29	
Axis_1.Axis_faststop	%I1.2.1.30	
Axis_1.Axis_ready	%I1.2.1.31	
Axis_1.Position	%IF1.2.1	
Axis_1.Control_acquire	%Q1.2.1.2	
Axis_1.Control_enable	%Q1.2.1.10	
Axis_1.Control_resume	%Q1.2.1.12	
Axis_1.Control_clear_ftt	%Q1.2.1.15	
Axis_1.Allow_capture	%Q1.2.1.16	
Axis_1.Allow_acquire	%Q1.2.1.18	
Axis_1.Allow_enable	%Q1.2.1.26	
Axis_1.Allow_follow	%Q1.2.1.27	
Axis_1.Allow_resume	%Q1.2.1.28	
Axis_1.Error_cmd	%MW1.2.1.19	
Axis_1.Return_1	%MD1.2.1.20	
Axis_1.Param_cmd_1	%MD1.2.1.27	
Axis_1.Param_cmd_2	%MD1.2.1.29	
Axis_1.Return_2	%MF1.2.1.22	
Axis_1.Return_3	%MF1.2.1.24	
Axis_1.Param_cmd_3	%MF1.2.1.31	
Axis_1.Param_cmd_4	%MF1.2.1.33	

IODDT Axis_2

此示例中使用的 IODDT 的元素如下：

符号	地址	注释
Axis_2.Ramping	%I1.2.2	
Axis_2.Steady	%I1.2.2.1	
Axis_2.Stopping	%I1.2.2.2	
Axis_2.Profile_end	%I1.2.2.3	
Axis_2.In_position	%I1.2.2.4	
Axis_2.Axis_homing	%I1.2.2.5	
Axis_2.Axis_homed	%I1.2.2.6	
Axis_2.Axis_not_following	%I1.2.2.1	
Axis_2.Holding	%I1.2.2.8	
Axis_2.Resuming	%I1.2.2.9	
Axis_2.Drive_enable	%I1.2.2.10	
Axis_2.Drive_diag	%I1.2.2.11	
Axis_2.Drive_warning	%I1.2.2.12	
Axis_2.Driveflt	%I1.2.2.13	
Axis_2.Drive_disable	%I1.2.2.14	
Axis_2.Axis_summaryflt	%I1.2.2.15	
Axis_2.Axis_comm_ok	%I1.2.2.16	
Axis_2.Axis_is_linked	%I1.2.2.17	
Axis_2.Axis_in_command	%I1.2.2.18	
Axis_2.Axis_capture	%I1.2.2.19	
Axis_2.Axis_at_target	%I1.2.2.20	
Axis_2.Axis_pos_limit	%I1.2.2.21	
Axis_2.Axis_neg_limit	%I1.2.2.22	
Axis_2.Drive_realtime_bit1	%I1.2.2.26	
Axis_2.Drive_realtime_bit2	%I1.2.2.27	
Axis_2.Axis_hold	%I1.2.2.28	
Axis_2.Axis_halt	%I1.2.2.29	
Axis_2.Axis_faststop	%I1.2.2.30	
Axis_2.Axis_ready	%I1.2.2.31	
Axis_2.Position	%IF1.2.2	
Axis_2.Return_1	%MD1.2.2.20	
Axis_2.Param_cmd_1	%MD1.2.2.27	

符号	地址	注释
Axis_2.Param_cmd_2	%MD1.2.2.29	
Axis_2.Return_2	%MF1.2.2.22	
Axis_2.Return_3	%MF1.2.2.24	
Axis_2.Param_cmd_3	%MF1.2.2.31	
Axis_2.Param_cmd_4	%MF1.2.2.47	

初步处理的编程

预备段

本段（放在任务的开头）用于初始化图：

```
! (* 图初始化 *)
```

```
IF NOT Start_cycle OR %S1 THEN SET %S21; END_IF;
```

```
! (* 运动值初始化 *)
```

```
%MF50 := %MF52 := 1000.0; (* 就绪位置 *)
```

```
%MF54 := 2000.0; (* 运动到传送带 B, 轴 X*)
```

```
%MF56 := 1500.0; (* 运动到传送带 B, 轴 Y*)
```

```
%MF58 := 2000.0; (* 运动到传送带 C, 轴 X*)
```

```
%MF60 := 500.0; (* 运动到传送带 C, 轴 Y*)
```

```
%MF102 := 10000.0; (* 自动速度 *)
```

```
%MF104 := 2000.0; (* 手动速度 *)
```

```
%MF106:= 1000.0; (* 参考点速度 *)
```

```
(* 通道 0 初始化 *)
```

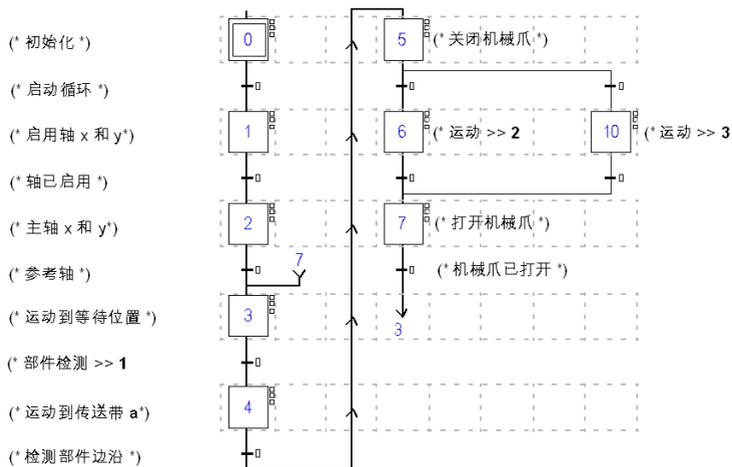
```
%Q1.2.26 := NOT %S0;
```

```
%L10;
```

SFC 编程

SFC 段

示例中的图的描述



连接器列表:

1	(* 加工工件检测 *)
2	(* 传送带 B 运动 *)
3	(* 传送带 C 运动 *)

转换编程

步骤 0 -> 1

- 转换的内容
!(*dcy 和自动模式并且无错误 *)
%I0.2.4 AND NOT %I0.2.3 AND NOT Axis_1.Axis_summaryflt AND
NOT Axis_2.Axis_summaryflt

步骤 1 -> 2

- 转换的内容
!(*%m1 手动控制 // %mw2.x.3 = 1 -> ERR WRITE_CMD*)
Axis_1.Drive_enable AND Axis_2.Drive_enable

步骤 2 -> 3

- 转换的内容
!(* 验证是否引用了驱动器并发出了命令 *)
Axis_1.Axis_homed AND Axis_2.Axis_homed
AND (Axis_1.Err_write_cmd = 0) AND (Axis_2.Error_cmd = 0)
AND NOT Axis_1.Cmd_err
AND NOT Axis_2.Cmd_err (* 检查交换 *)

步骤 3 -> 4

- 转换的内容
!(* 验证 => 指令结束并在正确的位置 *)
Axis_1.Profile_end AND Axis_2.Profile_end AND
Axis_1.In_position AND Axis_2.In_position
(* 检查轴 *)
AND %I3.0 (* 部件存在 *)
AND (Axis_1.Error_cmd = 0) AND (Axis_2.Error_cmd = 0) AND
NOT Axis_1.Cmd_err
AND NOT Axis_2.Cmd_err (* 检查交换 *)

步骤 4 -> 5

- 转换的内容
!(* 验证 => 指令结束并在正确的位置 *)
Axis_1.Profile_end AND Axis_2.Profile_end AND
Axis_1.In_position AND Axis_2.In_position
(* 检查轴 *)
AND %I3.0 (* 部件存在 *)
AND (Axis_1.Error_cmd = 0) AND (Axis_2.Error_cmd = 0) AND
NOT Axis_1.Cmd_err
AND NOT Axis_2.Cmd_err (* 检查交换 *)

步骤 5 -> 6

- 转换的内容
!(* 加工工件识别 *)
%I0.2.1 AND %I0.2.2

步骤 6 -> 7

- %X(6) -> %X(7)
!(* 验证 => 指令结束并在正确的位置 *)
Axis_1.Profile_end AND Axis_2.Profile_end AND
Axis_1.In_position AND Axis_2.In_position
(* 检查轴 *)
AND %I3.0 (* 部件存在 *)
AND (Axis_1.Error_cmd = 0) AND (Axis_2.Error_cmd = 0) AND
NOT Axis_1.Cmd_err
AND NOT Axis_2.Cmd_err (* 检查交换 *)

步骤 7 -> 3

- %X(7) -> %X(3)
!(* 打开机械爪?*)
NOT %I0.2.2

步骤 5 -> 10

- %X(5) -> %X(10)
!(* 加工工件识别 *)
NOT %I0.2.1 AND %I0.2.2

步骤 10 -> 7

- %X(10) -> %X(7)
!(* 验证 => 指令结束并在正确的位置 *)
Axis_1.Profile_end AND Axis_2.Profile_end AND
Axis_1.In_position AND Axis_2.In_position
(* 检查轴 *)
AND %I3.0 (* 部件存在 *)
AND (Axis_1.Error_cmd = 0) AND (Axis_2.Error_cmd = 0) AND
NOT Axis_1.Cmd_err
AND NOT Axis_2.Cmd_err (* 检查交换 *)

动作编程

步 1

- 步停用时的动作内容

```
!(* 轴通道 1 和 2 的确认 *)
```

```
SET Axis_1.Control_enable; SET Axis_1.Control_acquire; SET  
Axis_1.Control_enable; SET Axis_1.Control_acquire;
```

- 步激活时的动作内容

```
!(* 值初始化 *)
```

```
RESET Axis_1.Control_enable; RESET Axis_1.Control_acquire;  
RESET Axis_1.Control_enable; RESET Axis_1.Control_acquire;
```

步 2

- 步激活时的动作内容

```
!(* 通道 2 参考点 *)
```

```
Axis_2.Action_cmd := 6034; (** 参考点 **)  
Axis_2.Param_cmd_1 := 0; (** 参数 1**)  
Axis_2.Param_cmd_2 := 0; (** 参数 2**)  
Axis_2.Param_cmd_3 := 1.0; (**1 => 正向, -1 为负向 **)  
Axis_2.Param_cmd_4 := %MF106; (** 速度 **)  
WRITE_CMD(Axis_2);
```

```
!(* 通道 1 参考点 *)
```

```
Axis_1.Action_cmd := 6034; (** 参考点 **)  
Axis_1.Param_cmd_1 := 0; (** 参数 1**)  
Axis_1.Param_cmd_2 := 0; (** 参数 2**)  
Axis_1.Param_cmd_3 := 1.0; (**1 => 正向, -1 => 负向 **)  
Axis_1.Param_cmd_4 := %MF106; (** 速度 **)  
WRITE_CMD(Axis_1);
```

步 3

- 步激活时的动作内容


```
!(* 通道 2 绝对运动 *)
Axis_2.Action_cmd := 513; (** 立即运动 **)
Axis_2.Param_cmd_1 := 0; (** 运动类型: -1 错误、0 ABS_MOVE、1
INCR_MOVE、2 CONT_MOVE、3 ABS_MOVE_POS、4
ABS_MOVE_NEG**)
Axis_2.Param_cmd_2 := 0; (** 参数_2**)
Axis_2.Param_cmd_3 := %MF50; (方向/连续 1.0, 位置/绝对 **)
Axis_2.Param_cmd_4 := %MF102; (** 速度 **)
WRITE_CMD(Axis_2);
!(* 通道 1 绝对运动 *)
Axis_1.Action_cmd := 513; (** 立即运动 **)
Axis_1.Param_cmd_1 := 0; (** 运动类型: -1 错误、0 ABS_MOVE、1
INCR_MOVE、2 CONT_MOVE、3 ABS_MOVE_POS、4
ABS_MOVE_NEG**)
Axis_1.Param_cmd_2 := 0; (** 参数_2**)
Axis_1.Param_cmd_3 := %MF52; (方向/连续 1.0, 位置/绝对 **)
Axis_1.Param_cmd_4 := %MF102; (** 速度 **)
WRITE_CMD(Axis_1);
```

步 4

- 步激活时的动作内容


```
!(* 通道 1 绝对运动 *)
Axis_1.Action_cmd := 513; (** 立即运动 **)
Axis_1.Param_cmd_1 := 0; (** 运动类型: -1 错误、0 ABS_MOVE、1
INCR_MOVE、2 CONT_MOVE、3 ABS_MOVE_POS、4
ABS_MOVE_NEG**)
Axis_1.Param_cmd_2 := 0; (** 参数_2**)
Axis_1.Param_cmd_3 := 0.0; (方向/连续 1.0, 位置/绝对 **)
Axis_1.Param_cmd_4 := %MF102; (** 速度 **)
WRITE_CMD(Axis_1);
!(* 通道 2 绝对运动 *)
Axis_2.Action_cmd := 513; (** 立即运动 **)
Axis_2.Param_cmd_1 := 0; (** 运动类型: -1 错误、0 ABS_MOVE、1
INCR_MOVE、2 CONT_MOVE、3 ABS_MOVE_POS、4
ABS_MOVE_NEG**)
Axis_2.Param_cmd_2 := 0; (** 参数_2**)
Axis_2.Param_cmd_3 := 1000.0; (方向/连续 1.0, 位置/绝对 **)
Axis_2.Param_cmd_4 := %MF102; (** 速度 **)
WRITE_CMD(Axis_2);
```

步 5

- 步激活时的动作内容
!(* 关闭机械爪 *)
SET %Q0.3.0;

步 6

- 步激活时的动作内容
!(* 通道 1 绝对运动 *)
Axis_1.Action_cmd := 513; (** 立即运动 **)
Axis_1.Param_cmd_1 := 0; (** 运动类型: -1 错误、0 ABS_MOVE、1 INCR_MOVE、2 CONT_MOVE、3 ABS_MOVE_POS、4 ABS_MOVE_NEG**)
Axis_1.Param_cmd_2 := 0; (** 参数_2**)
Axis_1.Param_cmd_3 := %MF54; (方向/连续 1.0, 位置/绝对**)
Axis_1.Param_cmd_4 := %MF102; (** 速度 **)
WRITE_CMD(Axis_1);
!(* 通道 2 绝对运动 *)
Axis_2.Action_cmd := 513; (** 立即运动 **)
Axis_2.Param_cmd_1 := 0; (** 运动类型: -1 错误、0 ABS_MOVE、1 INCR_MOVE、2 CONT_MOVE、3 ABS_MOVE_POS、4 ABS_MOVE_NEG**)
Axis_2.Param_cmd_2 := 0; (** 参数_2**)
Axis_2.Param_cmd_3 := %MF56; (方向/连续 1.0, 位置/绝对*****)
Axis_2.Param_cmd_4 := %MF102; (** 速度 **)
WRITE_CMD(Axis_2);

步 7

- 步激活时的动作内容
(* 打开机械爪 *)
RESET %Q0.3.0;

步 10

● 步激活时的动作内容

```
!(* 通道 1 绝对运动 *)
```

```
Axis_1.Action_cmd := 513; (** 立即运动 **)
```

```
Axis_1.Param_cmd_1 := 0; (** 运动类型: -1 错误、0 ABS_MOVE、1  
INCR_MOVE、2 CONT_MOVE、3 ABS_MOVE_POS、4  
ABS_MOVE_NEG**)
```

```
Axis_1.Param_cmd_2 := 0; (** 参数_2**)
```

```
Axis_1.Param_cmd_3 := %MF58; (方向/连续 1.0, 位置/绝对 **)
```

```
Axis_1.Param_cmd_4 := %MF102; (** 速度 **)
```

```
WRITE_CMD(Axis_1);
```

```
!(* 通道 2 绝对运动 *)
```

```
Axis_2.Action_cmd := 513; (** 立即运动 **)
```

```
Axis_2.Param_cmd_1 := 0; (** 运动类型: -1 错误、0 ABS_MOVE、1  
INCR_MOVE、2 CONT_MOVE、3 ABS_MOVE_POS、4  
ABS_MOVE_NEG**)
```

```
Axis_2.Param_cmd_2 := 0; (** 参数_2**)
```

```
Axis_2.Param_cmd_3 := %MF60; (方向/连续 1.0, 位置/绝对 **)
```

```
Axis_2.Param_cmd_4 := %MF102; (** 速度 **)
```

```
WRITE_CMD(Axis_2);
```

后续处理的编程

后续段

后续段是完成任务并处理要执行的动作的段。

下面对该段进行描述

!(*----- 复位错误管理 -----*)

```
IF %IO.2.10 THEN SET Axis_1.Control_clear_flg;
    SET Axis_2.Control_clear_flg;
    SET Axis_1.Control_acquire;
    SET Axis_2.Control_acquire;
ELSE RESET Axis_1.Control_clear_flg;
    RESET Axis_2.Control_clear_flg;
    RESET Axis_1.Control_acquire;
    RESET Axis_2.Control_acquire;

END_IF;
```

!(*----- 手动模式管理 -----*)

```
(* 初始化 G7*)
IF %IO.2.3 THEN SET %S21;
END_IF;
(* 启用轴的手动模式 *)
IF %I2.10 THEN SET Axis_1.Control_enable;
    SET Axis_2.Control_enable;
    SET Axis_1.Control_acquire;
    SET Axis_2.Control_acquire;
ELSE RESET Axis_1.Control_enable;
    RESET Axis_2.Control_enable;
    RESET Axis_1.Control_acquire;
    RESET Axis_2.Control_acquire;

END_IF;
```

(***** 正向 x 通道手动模式 *****)

```
IF RE%IO.2.8 AND %IO.2.6 THEN
    Axis_1.Action_cmd := 513; (** 立即运动 **)
    Axis_1.Param_cmd_1 := 2; (** 运动类型 2 = CONT_MOVE **)
    Axis_1.Param_cmd_2 := 0; (** 参数 2 **)
    Axis_1.Param_cmd_3 := 0.0; (** 方向 **)
    Axis_1.Param_cmd_4 := %MF104; (** 速度 **)
    WRITE_CMD (Axis_1);
```

(*** 负向 x 通道手动模式 *****)**

```
IF RE%I0.2.9 AND %I0.2.6 THEN
    Axis_1.Action_cmd := 513; (** 立即运动 **)
    Axis_1.Param_cmd_1 := 2; (** 运动类型 2 = CONT_MOVE**)
    Axis_1.Param_cmd_2 := 0; (** 参数 2**)
    Axis_1.Param_cmd_3 := -1.0; (** 方向 **)
    Axis_1.Param_cmd_4 := %MF104; (** 速度 **)
    WRITE_CMD (Axis_1);
END_IF;
IF FE %I0.2.9 AND %I0.2.6 THEN Axis_1.Action_cmd := 407;
WRITE_CMD(Axis_1);
END_IF;
```

(*** 正向 y 通道手动模式 *****)**

```
IF RE%I0.2.8 AND NOT %I0.2.6 THEN
    Axis_2.Action_cmd := 513; (** 立即运动 **)
    Axis_2.Param_cmd_1 := 2; (** 运动类型 2 = CONT_MOVE**)
    Axis_2.Param_cmd_2 := 0; (** 参数 2**)
    Axis_2.Param_cmd_3 := 0.0; (** 方向 **)
    Axis_2.Param_cmd_4 := %MF104; (** 速度 **)
    WRITE_CMD (Axis_2);
END_IF;
IF FE %I0.2.8 AND NOT %I0.2.6 THEN Axis_2.Action_cmd := 407;
WRITE_CMD(Axis_2);
END_IF;
```

(*** 负向 y 通道手动模式 *****)**

```
IF RE%I0.2.9 AND NOT %I0.2.6 THEN
    Axis_2.Action_cmd := 513; (** 立即运动 **)
    Axis_2.Param_cmd_1 := 2; (** 运动类型 2 = CONT_MOVE**)
    Axis_2.Param_cmd_2 := 0; (** 参数 2**)
    Axis_2.Param_cmd_3 := -1.0; (** 方向 **)
    Axis_2.Param_cmd_4 := %MF104; (** 速度 **)
    WRITE_CMD (Axis_2);
END_IF;
IF FE %I0.2.9 AND NOT %I0.2.6 THEN Axis_2.Action_cmd := 407;
WRITE_CMD(Axis_2);
END_IF;
```

(***** 通道 x 的手动回归 *****)

```
IF RE %IO.2.7 AND %IO.2.6 THEN
    Axis_1.Action_cmd := 6034; (** 参考点 **)
    Axis_1.Param_cmd_1 := 0; (** 参数 1 **)
    Axis_1.Param_cmd_2 := 0; (** 参数 2 **)
    Axis_1.Param_cmd_3 := 1.0; (** 方向 **)
    Axis_1.Param_cmd_4 := %MF106; (** 速度 **)
    WRITE_CMD (Axis_1);
END_IF;
```

(***** 通道 y 的手动回归 *****)

```
IF RE %IO.2.7 AND NOT %IO.2.6 THEN
    Axis_2.Action_cmd := 6034; (** 参考点 **)
    Axis_2.Param_cmd_1 := 0; (** 参数 1 **)
    Axis_2.Param_cmd_2 := 0; (** 参数 2 **)
    Axis_2.Param_cmd_3 := 1.0; (** 方向 **)
    Axis_2.Param_cmd_4 := %MF106; (** 速度 **)
    WRITE_CMD (Axis_2);
END_IF;
```

! (*-----HALT 指令期间的停止循环 -----*)

```
IF RE%IO.2.5 THEN
    Axis_1.Action_cmd := 407; WRITE_CMD (Axis_1);
    Axis_2.Action_cmd := 407; WRITE_CMD (Axis_2);
END_IF;
```

! (*----- 错误类型读取 -----*)

```
IF RE%IO.2.11 THEN
    Axis_1.Action_cmd := 5510; WRITE_CMD (Axis_1);
    Axis_2.Action_cmd := 5510; WRITE_CMD (Axis_2);
END_IF;
```

! (*----- 通道禁用 -----*)

```
(* 禁用通道 2 (y 轴) 上的驱动器 *)
IF %M4 THEN Axis_2.Action_cmd := 402; WRITE_CMD (Axis_2); END_IF;
(* 禁用通道 1 (x 轴) 上的驱动器 *)
IF %M4 THEN Axis_1.Action_cmd := 402; WRITE_CMD (Axis_1); END_IF;
```

程序传输

传输程序的过程

在配置项目并输入程序之后，必须按如下过程将这些内容传输到 PLC 处理器的存储器中：

步骤	操作
1	从工具栏单击 标准模式 图标选择标准操作模式。
2	使用命令 PLC →连接 将终端连接到 PLC。
3	激活命令 PLC →将项目传输到 PLC 。
4	确认传输。

调整参数

简介

如果需要，能够调整配置参数可能很重要。

存在多种调整配置参数的可能方法：

- 可以返回离线模式并从配置屏幕直接修改参数
- Unity Pro 还允许您以在线模式或使用项目调整参数方面的功能修改参数

恢复 / 保存参数

此服务可用于网络通道、实轴、虚轴和远程轴以及从轴。

调试模式简介

简介

在执行和传输应用程序后，通常需要对其进行调试。

Unity Pro 将展示如何在类似如下的工具的帮助下进行调试：

- 调试屏幕：
 - 其中有许多应用程序状态指示器
 - 提供对某些变量的状态更改的访问

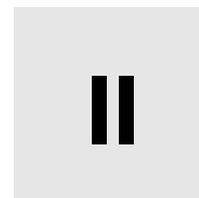
注意：调试功能必须在在线模式下使用。

调试过程

可以通过下面的方法执行程序调试：

步骤	操作
1	将 PLC 设置为运行模式。
2	显示 TSX CSY 84/164 模块的调试屏幕。 注： 可以同时显示语言编辑器以便理解应用程序的开发。
3	开始执行图。

模块 TSX CSY 84/164 的硬件安装



本部分主题

本部分介绍 TSX CSY 84、85 和 164 模块的硬件安装。

注意：本节中，与 TSX CSY 84 安装相关的所有数据也适用于 TSX CSY 164 和 TSX CSY 85。此外，以 TSX CSY 84 为例。

本部分包含了哪些内容？

本部分包括以下各章：

章	章节标题	页
4	TSX CSY 84 / 164 模块介绍	65
5	安装模块	71
6	多轴控制系统的描述	83
7	光缆	89
8	特性、标准和工作条件	93
9	兼容伺服驱动器	97

TSX CSY 84 / 164 模块介绍

4

本章主题

本章介绍 TSX CSY 84 和 TSX CSY 164 多轴控制模块。

本章包含了哪些内容？

本章包含了以下主题：

主题	页
模块的环境简介	66
模块的物理演示	68

模块的环境简介

简介

TSX CSY 84 和 TSX CSY 164 模块是 Premium 系列中的双格式应用专用模块，可安装在 TSX/PCI 57 PLC 工作站的 TSX RKY●● 机架中。它们是用于 Premium/Atrium PLC 的 SERCOS(r) 解决方案的元素之一，支持创建多轴控制系统。

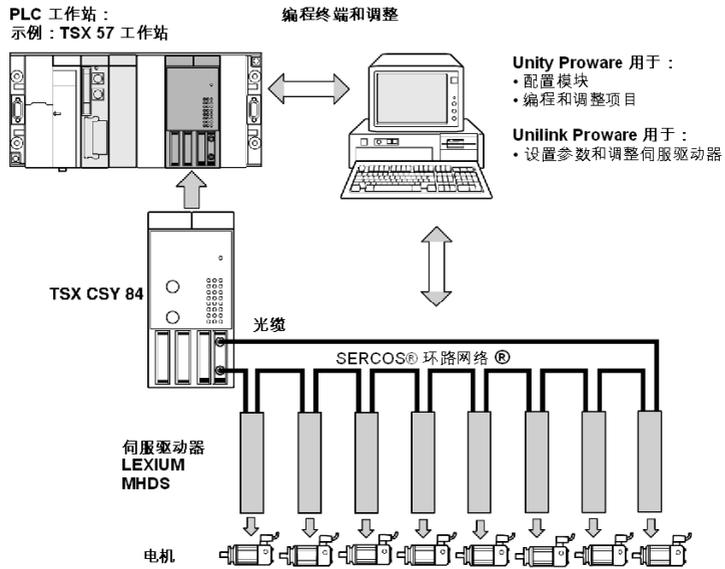
用于 Premium PLC 的 SERCOS(r) 解决方案的元素简介

用于 Premium PLC 的 SERCOS(r) 解决方案由以下元素组成：

- 一个由以下组件组成的 TSX/PCI 57 PLC 工作站：
 - 一个或多个机架，
 - 电源模块，
 - 处理器模块，
 - 项目所需的不同模块。
- 一个或多个 TSX CSY 84 或 TSX CSY 164 多轴控制模块，每个模块最多能够控制 SERCOS(r) 网络中的 8 个伺服驱动器。
- 一系列 5 LEXIUM MHDS 伺服驱动器，
- 一系列电机，
- 塑料光缆（0.3 至 16.5 米长）：
 - 确保环路网络结构中模块 / 伺服驱动器和伺服驱动器 / 伺服驱动器的物理连接，
 - 作为 TSX CSY 84 或 TSX CSY 164 模块（主站）与伺服驱动器（从站）之间的数字链路介质；此数字链路由欧洲标准 EN61491 定义。
- Unity Proware，用于配置 TSX CSY 84 和 TSX CSY 164 模块并对运动项目编程。
- UniLink LXM 17 软件，用于设置和调整伺服驱动器。

安装概述

下图显示了使用 TSX CSX 84 模块的 SERCOS(r) 多轴控制安装的示例



模块的物理演示

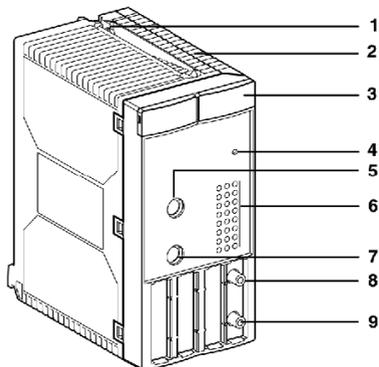
前面板描述

可在这些模块的前面板上找到以下组件：

- 一个包含 6 个 LED 的显示功能块，用于查看和诊断模块状态
- 一组（24 个）LED，用于查看和诊断各个模块通道
- 两个用于连接光缆的连接器，这些光缆构成模块与变速控制器之间的链路
- 两个 8 引脚的 Mini DIN 连接器

模块外观

下图显示了 TSX CSY 84 模块及其各个元素。



元素和功能

编号	元素	功能
1	螺钉	将模块安装到 TSX RKY ●● 机架
2	模块外壳	执行以下功能： <ul style="list-style-type: none"> • 支持和保护电路板 • 将模块锁定在插槽中
3	包含 6 个 LED 的显示功能块： <ul style="list-style-type: none"> • 绿色 RUN LED • 红色 ERR LED • 红色 I/O LED • 黄色 SER LED • 黄色 COM LED • 黄色 INI LED 	<p>这些 LED 使用户能够直观地了解模块的状态和错误。</p> <p>指示模块的操作模式。</p> <p>指示内部模块错误。</p> <p>指示外部模块错误或应用错误。</p> <p>指示 SERCOS(r) 网络上的通讯量在正常流动。</p> <p>无意义</p> <p>指示模块处于重新初始化阶段。</p>
4	凹陷按钮	用于初始化模块。
5	COM2 8 引脚 Mini DIN 连接器	保留。
6	24 个 LED 指示灯	用于直观观察和诊断模块通道。
7	COM1 8 引脚 Mini DIN 连接器	保留。
8	TX 发送 SMA 连接器	用于连接 SERCOS(r) 环路网络的发送光缆。
9	RX 接收 SMA 连接器。	用于连接 SERCOS(r) 环路网络的接收光缆。

安装模块

5

本章主题

本章描述用于 TSX CSY 84 / 164 多轴控制模块的安装操作。

本章包含了哪些内容？

本章包含了以下主题：

主题	页
在 PLC 工作站机架中安装模块	72
在 PLC 工作站中安装模块	73
由 TSX CSY 84 或 CSY 164 工作站管理的应用专用通道数	74
安装注意事项	75
模块显示	76
发生内部错误后的模块初始化	80
TSX CSY 84 和 TSX CSY 164 模块操作模式	81

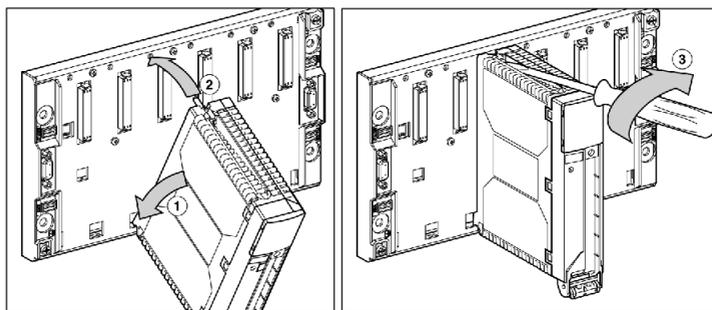
在 PLC 工作站机架中安装模块

简介

TSX CSY 84/164 模块可安装在用于 TSX 57/PCX 57/PCIX 57 PLC 的 TSX RKY ●● 机架上的所有可用插槽中（保留用于电源模块和处理器的插槽除外）。此双格式模块在 TSX RKY ●● 机架上占据 2 个插槽。

示意图

下图介绍标准格式 Premium 系列模块在 TSX RKY ●● 机架上的安装过程。该过程与用于双格式模块的过程相同。



过程

下表描述要执行的操作。

步骤	动作
1	将模块反面的引脚定位到位于机架底部的中心孔中。
2	旋转模块以使其与机架接触。
3	使用位于模块上半部的螺钉，将模块固定在机架上。最大拧紧扭矩：2.0 牛米。

在 PLC 工作站中安装模块

简介

TSX CSY 84 / 164 模块可安装在位于 PLC 工作站的主 X 总线段上的所有机架中。

安装模块

下图表示 TSX CSY 模块在属于主 X 总线段的机架中的安装。该模块可安装在支持处理器的机架中，以及安装在 X 总线上已有的所有其他机架中。支持 TSX CSY 模块的机架与支持处理器的机架之间的距离不得超过 100 米。



注意：TSX CSY 84 / 164 模块不能安装在属于远程 X 总线段（通过 TSX REY 200 模块扩展）的机架中。

由 TSX CSY 84 或 CSY 164 工作站管理的应用专用通道数

应用专用通道的定义

应用专用模块（TSX CTY• 计数模块、TSX CAY• 轴控模块、TSX CFY• 步进控制模块、TSX YSP Y• 称重模块、TSX CCY 1128 电子凸轮模块、TSX CSY 84 或 TSX CSY 164 运动控制模块等）具有的通道数可能为 1 到 n，具体取决于模块的类型（请参见*使用 Unity Pro 的 Premium 和 Atrium, 处理器、机架和电源模块, 实施手册*）；这些通道称为应用专用通道。

计算工作站上的应用专用通道的数量

目的在于定义：

- 要安装的处理器电源，
- 工作站上可安装的最大应用专用模块数量。

TSX CSY 84 模块的应用专用模块数量

一个 TSX CSY 84 模块最多可有 32 个应用专用通道。只应该计算已配置的应用专用通道。

TSX CSY 84 模块最多管理 8 个与伺服驱动器关联的实轴（通道 1 至 8）。除了这些实轴外，该模块还可管理：

- 4 个虚轴（通道 9 至 12），
- 4 个带远程输入的轴（通道 13 至 16），
- 4 组协同轴（通道 17 至 20）。
- 4 组从轴（通道 21 至 24）。
- 7 个凸轮配置文件（通道 25 至 31）。

通道 0 (SERCOS(r)) 是数字总线管理通道。

TSX CSY 164 模块的应用专用模块数量

- 通道 1 至 16 可分别支持一个实轴功能、一个虚轴功能或一个外部设定点：
- 4 组协同轴（通道 17 至 20）。
- 4 组从轴（通道 21 至 24）。
- 7 个凸轮配置文件（通道 25 至 31）。

通道 0 (SERCOS(r)) 是数字总线管理通道。

安装注意事项

简介

为了保证最佳操作条件，有必要在安装和移除模块、在模块前面板插入和拔出连接器以及调整其固定螺钉时采取某些预防措施。

安装和移除模块

安装和移除模块应该在断电情况下进行。但是，可以在不切断机架电源的情况下安装和移除模块，而不会有任何损坏模块的风险。

拧紧和拧松模块前面板上的光纤连接器

可以在模块已通电时拧紧和拧松模块前面板上的光纤连接器，而不会有任何损坏连接器的风险。

为了避免项目功能失常，最好在断电情况下执行这些操作。

拧紧模块的固定螺钉的扭矩

拧紧扭矩：2.0 牛米。

模块显示

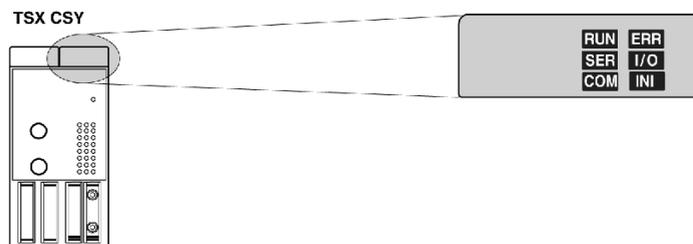
用途

TSX CSY 84 或 TSX CSY 164 模块配有两个显示单元：

- 标准 Premium 中的显示功能块由 6 个 LED 组成，它们的功能是通知用户：
 - 模块的操作模式：模块操作正常、出错或关闭
 - 内部或外部模块操作错误
- 一组（24 个）用于显示模块的应用专用通道（实轴、虚轴等）状态的 LED。

显示功能块的演示

下图显示模块的显示功能块的物理外观及其六个 LED 的位置。



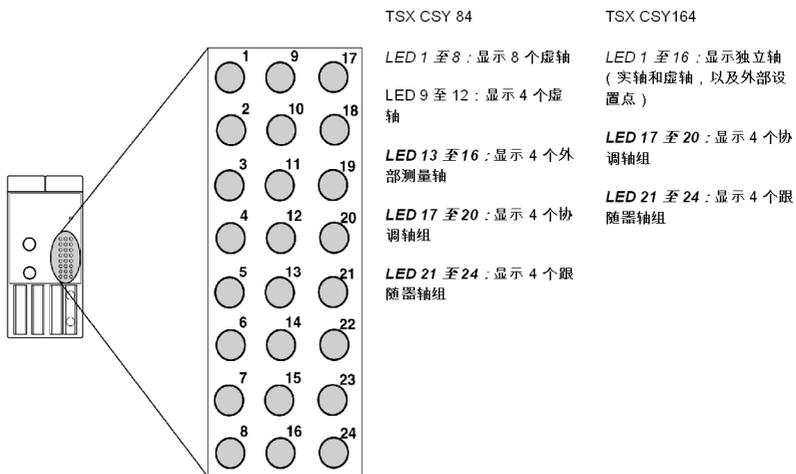
各个显示功能块 LED 的状态及其含义

下表显示了显示功能块上的每个 LED 的不同状态及其含义。

LED	颜色	状态	含义
RUN	绿色	亮	模块处于正常操作模式
		灭	模块故障、模块断电、模块处于初始化阶段或项目缺失
ERR	红色	亮	内部模块错误： • 模块出现故障
		闪烁	• 模块启动 • 通讯错误 • 项目缺失、无效或执行错误
		灭	操作正常，无故障
I/O	红色	亮	外部模块错误： • 接线错误
		闪烁	无意义
		灭	操作正常，无故障
INI	黄色	亮	无意义
		闪烁	模块处于复位阶段
		灭	工作正常
SER	黄色	亮	无意义
		闪烁	SERCOS(r) 网络上的通讯量处于正常工作模式
		灭	SERCOS 网络上没有通讯量
COM	黄色	-	未使用

应用专用通道的显示 LED 的演示

下图显示了某些应用专用通道的 24 个显示 LED 的物理特性。



应用专用通道显示 LED 的状态和含义

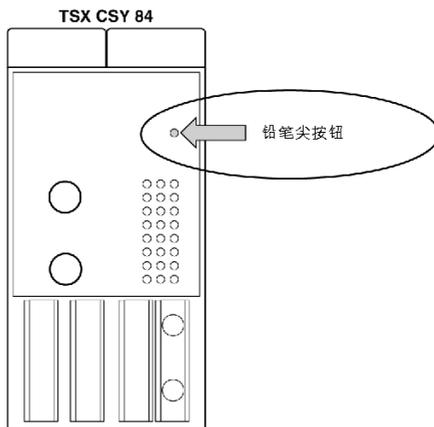
下表显示了应用专用通道显示 LED 的不同状态：

LED	CSY 84 分配	CSY 164 分配	状态	含义	
1 2 3 4 5 6 7 8	实轴 1 实轴 2 实轴 3 实轴 4 实轴 5 实轴 6 实轴 7 实轴 8	实轴、虚轴或外部设 定点	亮	轴处于正常操作模式	
9 10 11 12	虚轴 1 虚轴 2 虚轴 3 虚轴 4		闪烁		轴处于配置模式或出错
13 14 15 16	外部设定点 1 外部设定点 2 外部设定点 3 外部设定点 4		灭		轴未配置或配置错误
17 18 19 20	协同轴组 1 协同轴组 2 协同轴组 3 协同轴组 4				
21 22 23 24	跟随器轴组 1 跟随器轴组 2 跟随器轴组 3 跟随器轴组 4				

发生内部错误后的模块初始化

如何初始化模块

使用位于前面板上的笔尖按钮初始化模块，如下图所示。



注意： 有关笔尖按钮上的物理动作的建议

应用于此按钮的压力应当适度。

必须将该工具的控制点向与模块前面板垂直的方向按压，并置于访问窗口的中心部位。

不遵守此建议可能导致该按钮损坏。

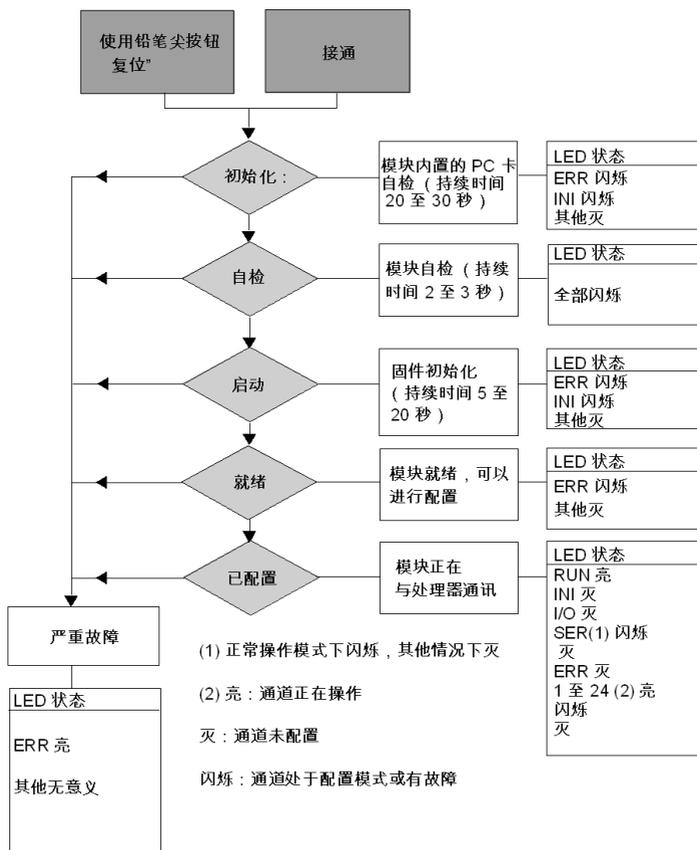
初始化的结果

模块在初始化阶段重新启动，其方式与在通电时相同。操作模式如下文所述。

TSX CSY 84 和 TSX CSY 164 模块操作模式

模块操作模式概述

下面的内容概述不同操作模式阶段，并针对每个步骤给出模块前面板上的指示灯状态。



多轴控制系统的描述

6

本章主题

本章描述 TSX CSY 84 和 TSX CSY 164 模块如何与 SERCOS(r) 网络配置中的变速控制器连接以形成多轴控制。

本章包含了哪些内容？

本章包含了以下主题：

主题	页
SERCOS(r) Premium PLC 的多轴控制系统	84
SERCOS(r) 环路网络	86

SERCOS(r) Premium PLC 的多轴控制系统

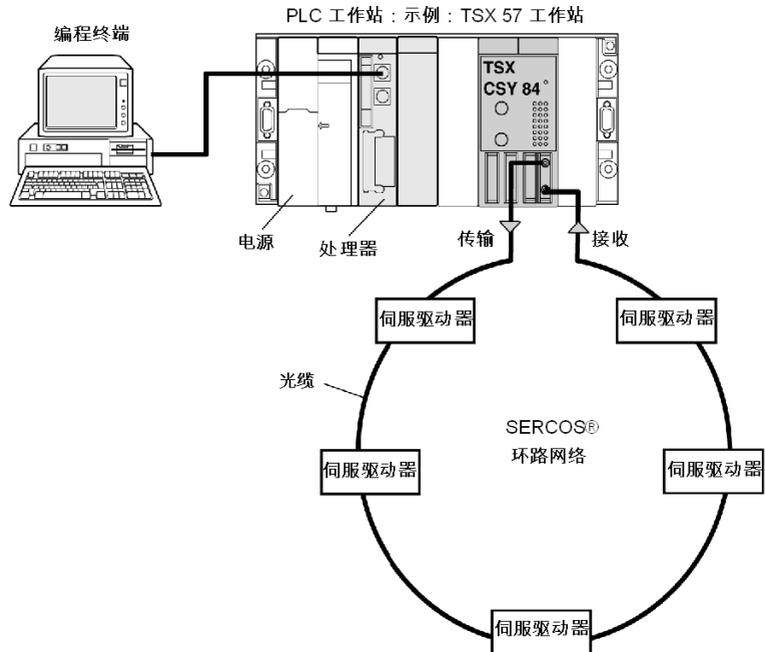
SERCOS(r) 多轴控制系统的架构

Premium PLC 上的 SERCOS(r) 多轴控制系统的架构包括:

- 一个 TSX/PCI 57 PLC 工作站,
- 一个 TSX CSY 84 或 TSX CSY 164 多轴控制模块,
- 控制与不同轴关联的电机的伺服驱动器,
- 一个光纤 SERCOS(r) 网络。

示意图

下图给出了 SERCOS(r) 多轴控制系统的架构的示例。



工作原理

该多轴控制系统由 TSX CSY 84 或 TSX CSY 164 多轴控制模块和伺服驱动器组成，这些组件在一个网络中通过光缆互连。

通过光纤网络互连的驱动器充当独立轴。

由 TSX CSY 84 或 TSX CSY 164 模块发送的运动命令将发送到网络中的每个驱动器，模块又反过来接收网络中每个轴的实际位置值。

由 TSX CSY 84 模块管理的最大实轴数量

一个 TSX CSY 84 模块最多管理 8 个与伺服驱动器关联的实轴。

除了这些实轴外，该模块还可管理：

- 4 个虚轴，
- 4 个远程轴，
- 4 个协同轴组，
- 4 个从轴组，
- 7 个凸轮配置文件。

TSX CSY 164 模块的应用专用模块数量

- 通道 1 至 16 可分别支持一个实轴功能、一个虚轴功能或一个外部设定点；
- 4 组协同轴（通道 17 至 20）。
- 4 组从轴（通道 21 至 24）。
- 7 个凸轮配置文件（通道 25 至 31）。

项目开发

项目开发可在终端 (PC) 上用 Unity Pro 软件实现，并允许：

- 轴配置，
- 试运行，
- 项目调整和诊断。

SERCOS(r) 环路网络

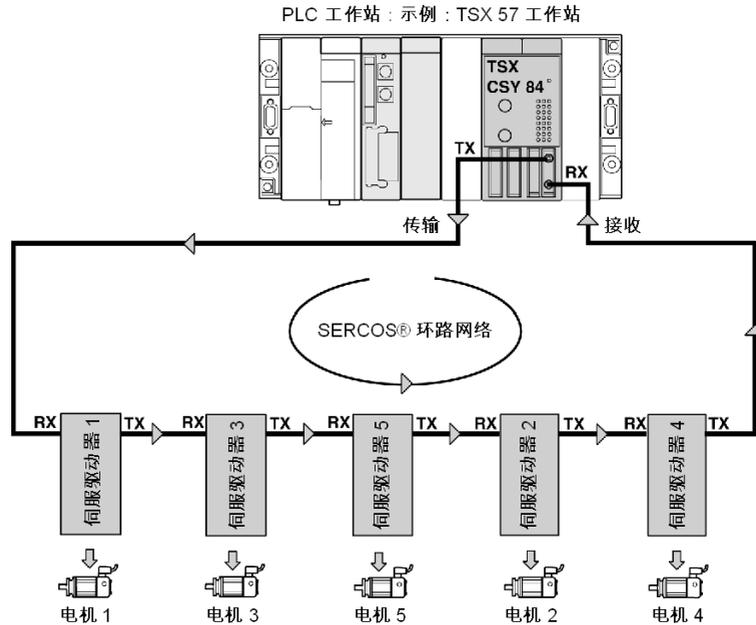
概览

通过 SERCOS(r) 光纤环路网络，TSX CSY 84/164 可以：

- 将由项目程序定义的运动控制指令传输到每个轴，
- 反过来接收伺服驱动器中的不同轴位置传感器发送的实际数据。

SERCOS(r) 环路网络

下图显示了 SERCOS(r) 网络配置的示例，其中包含五个通过光缆互连到 TSX CSY 84 模块的 LEXIUM 伺服驱动器。



发送命令和接收数据

该模块协调网络上安装的不同轴的运动活动：

- 将不同轴的运动控制指令发送到网络和伺服驱动器，
- 接收网络中每个轴的实际数据，并根据需要对该数据执行必要的处理。

通过 (TX) 连接器和光缆，模块向第一个驱动器发送运动指令，后者解释并执行这些指令；然后这些指令被发送到下一个驱动器。

环路中的最后一个驱动器通过光缆将所有轴的实际数据发送到连接器 (RX) 上。

注意：SERCOS(r) 网络中的一个伺服驱动器断电将导致环路断开，从而造成系统故障。

循环时间

数据在网络上单向传输，典型循环时间为 4 毫秒。如果交换的数据量允许，此时间可在配置模式中缩短至 2 毫秒。

传输速度

传输速度缺省定义为 4 兆波特。如果驱动器无法支持此速度，可在配置模式中将其降低到 2 兆波特。

不同网段的最大长度

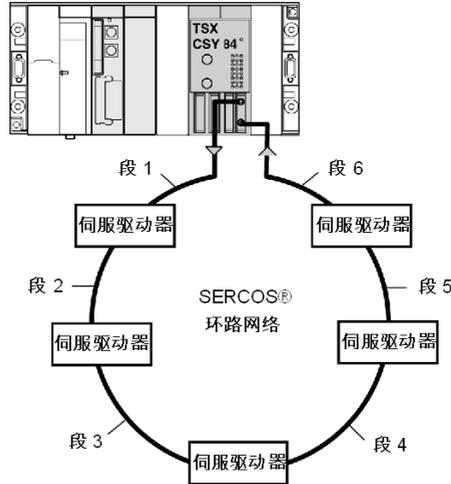
使用 Schneider Electric 建议的塑料光缆，每个 SERCOS(r) 网络段的最大长度限制为 40 米。

不同网段的示意图

不同的 SERCOS(r) 网段：

- TSX CSY 84 模块 / 驱动器：段 1
- 驱动器 / 驱动器：段 2 至 n
- 驱动器 / 模块：段 n+1

下图显示连接了 5 个伺服驱动器的 SERCOS(r) 网络的不同段。



发射器的光功率调整取决于段的长度

SERCOS(r) 网络（TSX CSY 84 模块和驱动器）的每个元素都有一个光发射器。

对于每个光发射器，操作员必须根据段长度调整光功率。

- 段 1（模块 / 第一个驱动器）的光功率：始终由 TSX CSY 84 模块光发射器提供。光功率调整是在配置模式中通过 Unity Pro 执行的，方法是根据段长度输入光功率的百分比。

段长度（以米为单位）	光功率（作为总功率的百分比）
$0 < L < 15$	66%
$15 < L < 40$	100%

- 其他段（驱动器 / 驱动器和最后一个驱动器 / 模块）的光功率：始终由驱动器的光发射器提供。光功率通过 UniLink 软件进行调整，方法是只设置段长度的参数。

光缆



本章主题

本章介绍用于连接 SERCOS 网络的各个组件（TSX CSY 84 / 164 模块和变速控制器）的光缆。

本章包含了哪些内容？

本章包含了以下主题：

主题	页
预装配光缆	90
用于制作自定义光缆的工具箱	91

预装配光缆

概览

Schneider Automation 提供一系列 1 毫米直径的塑料光缆，用于连接 SERCOS 网络的不同组件（TSX CSY 84/164 模块和变速控制器）。每条光缆的两端配有一个 SMA 类型的连接器。

光缆列表

下表提供每种光缆的参考号和长度。

参考号	长度
990 MCO 00001	0.3 米
990 MCO 00003	0.9 米
990 MCO 00005	1.5 米
990 MCO 00015	4.5 米
990 MCO 00055	16.5 米
990 MCO 00075	22.5 米
990 MCO 000125	37.5 米

建议

在安装光缆时，必须遵守以下建议：

注意： 有关光缆曲率半径的建议。

对于此类光缆，最小曲率半径必须大于 25 毫米。

不遵守这一点可能导致光缆损坏。

注意： 有关安装时应用于光缆的张力的建议。

安装时应用于光缆的最大张力一定不能超过 6 千克。

不遵守这一点可能导致光缆损坏。

最高允许温度：-40°C+80°C。

用于制作自定义光缆的工具箱

概览

Schneider Automation 提供两个用于制作自定义光缆的工具箱：

- 1 个工具箱
- 1 个包含光缆和连接器的设备工具箱

工具箱

下表提供工具箱的参考号和物品。

参考号	物品	
	数量	描述
990 MCO KIT 00	1	用于制作光缆的操作说明
	1	光缆剥开工具
	1	连接器夹钳
	1	25 W / 110 V 烙铁

设备工具包

下表提供设备工具包的参考号和物品。

参考号	物品	
	数量	描述
990 MCO KIT 01	12	SMA 类型的连接器
	12	绝缘套管
	1	30 米塑料光缆

特性、标准和工作条件

8

本章主题

本章介绍 TSX CSY 84 模块和 SERCOS 网络的不同特性。

本章包含了哪些内容？

本章包含了以下主题：

主题	页
模块的特性	94
SERCOS(r) 网络的特性	95
标准和工作条件	96

模块的特性

电气特性

下表给出模块的电气特性：

参数描述	值	
	典型值	最大值
模块从 5V 机架电源中消耗的电流	1.8 A	2 A
在模块中损耗的功率	9 W	10 W
光纤输出	符合标准 EN 61491	

工作温度和存储温度 / 湿度 / 海拔高度

下表给出模块的电气特性：

参数描述	值
工作温度	0 至 60° C
储存温度	-25° C 到 70° C
湿度（无冷凝）	5% 到 95%
工作海拔高度	0 至 2000 米

SERCOS(r) 网络的特性

特性表

下表给出 SERCOS(r) 网络的主要特性。

参数描述	值
地址	1...254
波特率	2 或 4 兆波特，可通过软件配置
循环时间	4 毫秒

标准和工作条件

标准

与适用于 Premium/Atrium PLC 的标准完全相同。

标准 EN 61491:

用于工业机器的电气设备。用于控制单元与驱动器之间的实时通讯的串行数据链路。

工作条件和环境要求

与适用于 Premium/Atrium PLC 的工作条件和环境要求完全相同。

兼容伺服驱动器

9

变速控制器列表

Schneider Electric 系列中的变速控制器列表

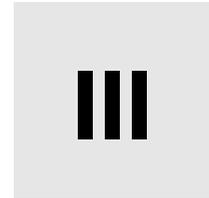
Schneider Electric 提供一系列与 Premium PLC 上的 SERCOS 系列兼容的变速控制器。下表提供每种变速控制器的参考号和描述。

参考号	描述
MHD• 1004 •00	LEXIUM 变速控制器, 连续电流 1.5 A 有效值
MHD• 1008 •00	LEXIUM 变速控制器, 连续电流 3 A 有效值
MHD• 1017 •00	LEXIUM 变速控制器, 连续电流 6 A 有效值
MHD• 1028 •00	LEXIUM 变速控制器, 连续电流 10 A 有效值
MHD• 1056 •00	LEXIUM 变速控制器, 连续电流 20 A 有效值
MHD• 1112 •00	LEXIUM 变速控制器, 连续电流 40 A 有效值
MHD• 1198 •00	LEXIUM 变速控制器, 连续电流 70 A 有效值

其他变速控制器

所有符合 EN 61491 的变速控制器都可以与 TSX CSY 84/164 模块一起使用。

TSX CSY 84 和 TSX CSY 164 模块 通用的运动控制



本节主题

本节介绍 TSX CSY 84 和 TSX CSY 164 模块，并描述了使用这些模块时如何实现运动控制。

注意：本节中，所有与 TSX CSY 84 安装相关的数据也适用于 TSX CSY 164。并且将 TSX CSY 84 作为示例。

本部分包含了哪些内容？

本部分包括以下各章：

章	章节标题	页
10	功能描述	101
11	配置	115
12	TSX CSY 84 和 TSX CSY 164 模块编程	135
13	调整 TSX CSY 84 和 TSX CSY 164 模块	227
14	调试 TSX CSY 84 / 164 模块	311
15	诊断和维护	329
16	SERCOS 语言对象	373
17	附录	435

本章主题

本章描述 TSX CSY 84 模块的功能特点以及它可执行的功能。

本章包含了哪些内容？

本章包含了以下主题：

主题	页
模块功能	102
"SERCOS(r)" 功能	104
"实轴" 功能	105
"虚轴" 功能	106
"远程轴" 功能	107
组的概念	108
"协同轴组" 功能	109
"从轴组" 功能	111
"凸轮配置文件" 功能	113

模块功能

概览

TSX CSY 84/164 模块有五种类型的应用专用功能：

- SERCOS® 功能，通过通道 0 安装（SERCOS® 环路）
- 独立轴功能（实轴、虚轴、远程轴），通过通道 1 - 16 安装
- 协同轴组功能，通过通道 17 - 20 安装
- 从轴组功能，通过通道 21 - 24 安装
- 凸轮配置文件功能，通过通道 25- 31 安装

注意：当使用 TSX CSY 84 模块时，Unity 接受 MOD_PARAM 函数，但其不可用。因此，MOD_PARAM 函数不可用于 TSX CSY 84 模块，仅可用于 TSX CSY 164 模块。

命令交换

对于每种应用专用功能（SERCOS® 环路、独立轴、协同轴组、从组和凸轮配置文件），执行交换的速率为：每扫描两次 MAST PLC 任务，就对每个通道执行 1 个命令。

下面的 32 个命令位使命令获得授权并得以确认：

- 16 位 CONTROL（控制），上升沿上有效（%Qr.m.C.0 到 %Qr.m.C.15）
- 16 位 ALLOW（允许），下降沿上有效（%Qr.m.C.16 到 %Qr.m.C.31）

性能优化

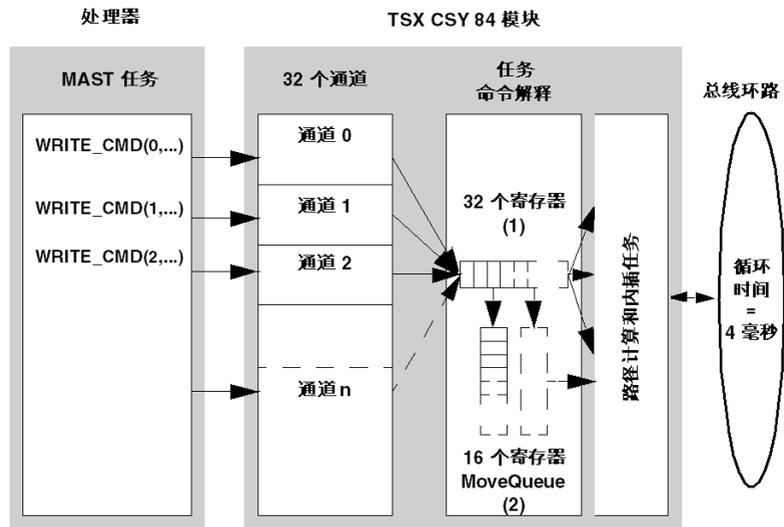
在 MAST 任务的每个循环中，将为所有通道安装隐式交换（%I，%Q）。为了优化性能，建议尽可能使隐式交换（%Q）优先于显式交换（WRITE _CMD）。

重要事项

TSX CSY 84 模块必须在 MAST 任务中配置。

交换的图形表示

WRITE_CMD 命令的交换根据下面的原理进行:



(1) 32 个寄存器 (每通道 1 个)

(2) 用于 32 个命令的 16 个寄存器 (8 个实轴、4 个虚轴、4 个协同轴组)

"SERCOS(r)" 功能

概览

SERCOS(r) 功能由 TSX CSY 84 和 TSX CSY 164 模块的通道 0 执行。它控制总线环路。在通电时或在总线关闭期间，轴控模块执行总线的自动学习。

通道 0

通道 0 用缺省参数进行配置。这些参数通过以下任一方式进行修改：

- 使用配置或调整屏幕，
- 或通过使用显示交换的项目。

可访问的服务

通道 0 支持使用以下服务（请参见 *与应用专用功能关联的显式交换语言对象*，第 377 页）：

- READ_PARAM
- WRITE_PARAM
- WRITE_CMD
- SAVE_PARAM
- RESTORE_PARAM
- READ_STS

" 实轴 " 功能

概览

" 实轴 " 功能由 TSX CSY 84 模块的通道 1 – 8 以及 TSX CSY 164 模块的通道 1 – 16 执行。它可通过总线环路控制伺服驱动器以创建物理轴（最多 8 个可能的物理轴）。

注意：轴控模块和伺服驱动器的参数分别通过 Unity Pro 和 UniLink 运行。因此，如果用户使用 UniLink 配置器更改模块和伺服驱动器所使用的参数的值，则此更改不会在轴控模块中更新。因此，在执行了此类操作之后，用户必须使用 Unity Pro 调整模块中的参数。

有关如何管理参数的更多信息，（请参见 *参数管理*，第 385 页）。

注意：对应于实轴 1 的伺服驱动器应该位于 SERCOS(r). 环路的地址 1 上。

可访问的服务

通道 1 到 8 使得下面的服务可供使用（请参见 *与应用专用功能关联的显式交换语言对象*，第 377 页）：

- READ_PARAM
- WRITE_PARAM
- WRITE_CMD
- SAVE_PARAM
- RESTORE_PARAM
- READ_STS
- TRF_RECIPE

"虚轴" 功能

概览

"虚轴" 功能由 TSX CSY 84 模块的通道 9 到 12 以及 TSX CSY 164 模块的通道 1 到 16 执行。

虚轴不是物理轴，但是可用于协调多个物理轴的运动，还可以是从轴组的主轴。还可以用它来测试系统而不移动轴。

可访问的服务

通过 TSX CSY 84 模块的通道 9 到 12 以及 TSX CSY 164 模块的通道 1 到 16 可使用下列服务：（请参见 *与应用专用功能关联的显式交换语言对象*，第 377 页）：

- READ_PARAM
- WRITE_PARAM
- WRITE_CMD
- SAVE_PARAM
- RESTORE_PARAM
- READ_STS

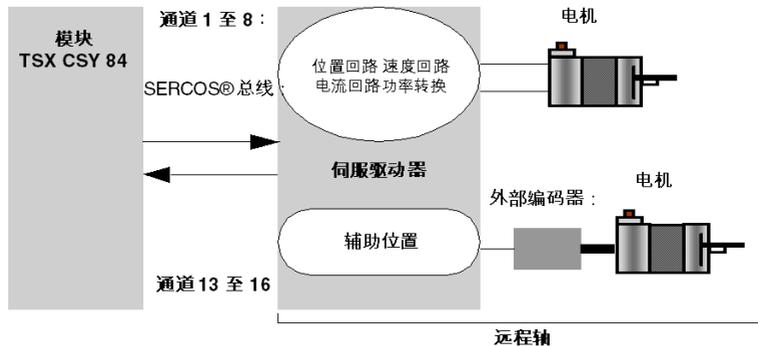
"远程轴" 功能

概览

"远程轴"功能由 TSX CSY 84 模块的通道 13 到 16 以及 TSX CSY 164 模块的通道 1 到 16 执行。

远程轴可用于集成由远程系统传递的位置数据。

下图从伺服驱动器结合 TSX CSY 84 模块的角度演示了实轴与远程轴之间的区别：



远程轴

根据远程轴功能的配置，可通过下面的对象访问远程轴功能：

- REMOTE_POSITION (%QDr.m.c.0) 语言对象 (隐式交换)
- 伺服驱动器的辅助输入

注意：在实轴所支持的**远程轴**通道上，故障标志是在对应的**实轴**通道上实现的。

可访问的服务

TSX CSY 84 模块的通道 13 至 16 和 TSX CSY 164 模块的通道 1 至 16 允许使用以下服务：(请参见与**应用专用功能**关联的**显式交换语言对象**，第 377 页)：

- READ_PARAM
- WRITE_PARAM
- WRITE_CMD
- SAVE_PARAM
- RESTORE_PARAM
- READ_STS

组的概念

概览

TSX CSY 84/164 模块使用组的概念来命令轴。某个轴组上的任何动作都会使该组中的成员轴承受完全相同的动作。

运动组

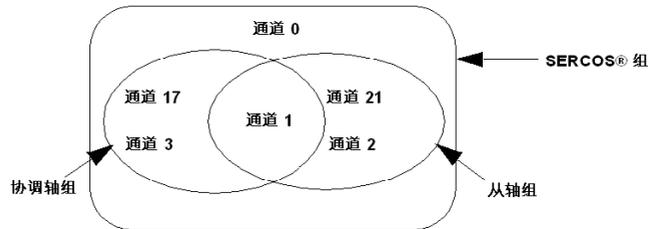
所有运动组都有它们自己的运动控制命令接口。

- 单独的轴的运动控制接口始终作用于该轴。
- SERCOS(r) 环路的运动控制接口作用于该串上的所有运动组轴。
- 协调轴组或从轴组的运动控制接口只有在运动组受控 (Get) 时才有效。

这些寄存器中的控制位的位置控制导致对一组轴而不是对单个轴实施对应的动作。运动组每隔一个 SERCOS(r) 时钟周期扫描一次运动控制接口。对控制位的后续更改必须至少相隔一个 SERCOS(r) 时钟周期。一般情况下，应该检查运动组的运动状态寄存器以确定某个控制位的动作是否成功。

有关 TSX CSY 84 模块的配置示例

假设下列组已在配置中定义



- 通道 0: SERCOS(r) 组
- 通道 21: 从轴组 21, 包含实轴 1 和 2
- 通道 17: 协调轴组 17, 包含实轴 1 和 3

网络通道（通道 0）上的任何动作都会影响组 17 和 21，以及实轴 1、2 和 3。

从轴组（通道 21）上的任何动作都会影响实轴 1 和 2。

协调轴组（通道 17）上的任何动作都会影响实轴 1 和 3。

注意：为了使用某个实轴，此规则要求将 SERCOS(r) 组（通道 0）的允许位设置为 1。

“协同轴组”功能

概览

“协同轴组”功能由 TSX CSY 84 和 TSX CSY 164 模块的通道 17 到 20 执行。

协同轴组是协同运动的物理轴（最多 8 个）的集合。组中有一个轴定义为协同主轴。

每个轴的位置由运动控制定义。协同组运动的参照速度是协同主轴的速度。计算其他轴的加速度和速度是为了使所有轴同时开始和结束运动。

注意：必须在协同组得到确认之前和在实现协同运动之前，设定由该协同组使用的所有轴的参考点。

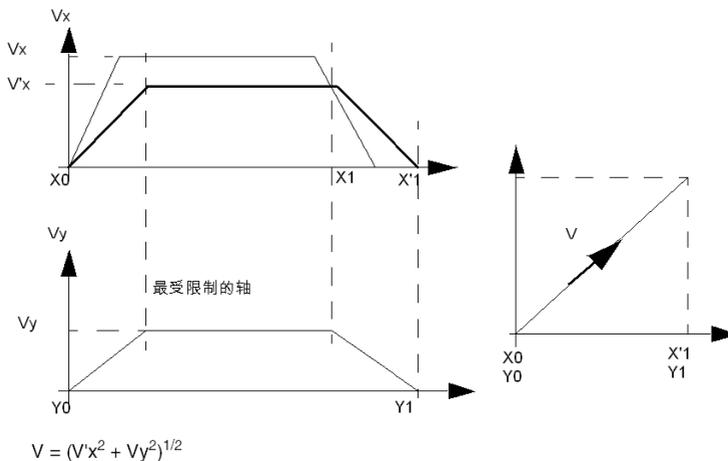
注意：当轴配置在从组中时，建议将一个组与某个给定轴关联。

注意：轴成员故障必须在组级别得到确认 (CLEAR_FLT)。

参照速度

协同轴组运动的参照速度必须相对于限制最严格的轴进行计算。该算法将降低大多数活动轴的速度。

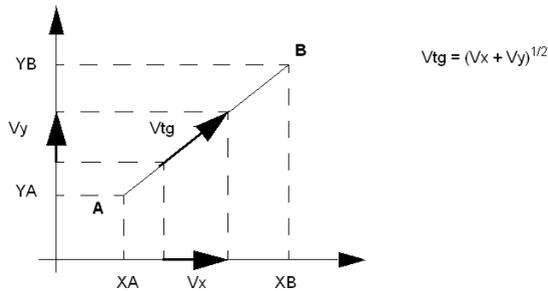
例如，在 2 轴系统的情况下



切向速度

切向速度是相对于由 WRITE_CMD (MOVE) 指令定义的初始速度在 X 轴上的投影计算的。如果没有机械限制，则为其他速度输入最大速度。

2 轴系统情况下的示例



运动部件以切向速度 Vtg 从点 A (XA, YA) 移动到点 B (XB, YB), Vtg 在 X 和 Y 上分别投影为 Vx 和 Vy。

内插器使用 WRITE_CMD (MOVE) 指令中的 Vx (限制最严格的轴) 值计算速度 Vy, 然后计算 Vtg。

可访问的服务

通道 17 到 20 使得下面的服务可供使用 (请参见 *与应用专用功能关联的显式交换语言对象*, 第 377 页):

- WRITE_CMD
- READ_STS
- MOD_PARAM (参见第 462 页), 仅用于 TSX CSY 164 模块。此功能不可用于 TSX CSY 84 模块。

“从轴组”功能

概览

“从轴组”功能由 TSX CSY 84/164 模块的通道 21 到 24 执行。

从轴组由一个主轴和多个从轴（最多 6 个）组成，从轴跟随主轴的运动。

有两种方式跟随主轴：

- 比率模式：从轴根据在配置中定义的称为“随动关系”的关系（从轴的位置与主轴相同：设定点或实际值，乘以比率）跟随主轴
- 凸轮模式：从轴根据凸轮配置文件（即一个点表，它将从轴位置作为主轴位置的函数：设定点或当前值）跟随主轴。

主轴可为实轴或虚轴，或者为外部设定点。从轴为实轴或虚轴。

注意：当轴配置在从组中时，建议将一个组与某个给定轴关联。

注意：轴成员故障必须在组级别得到确认 (CLEAR_FLT) 。



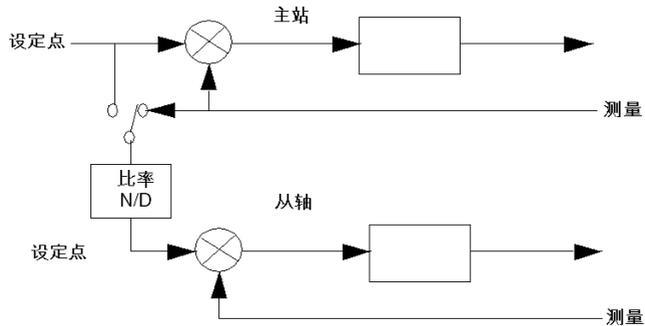
应用程序的意外行为

在使用主从配置时，确保从轴性能与主轴速度和加速法则特性相兼容。

如果不遵守这些说明，将会导致受伤或设备损坏。

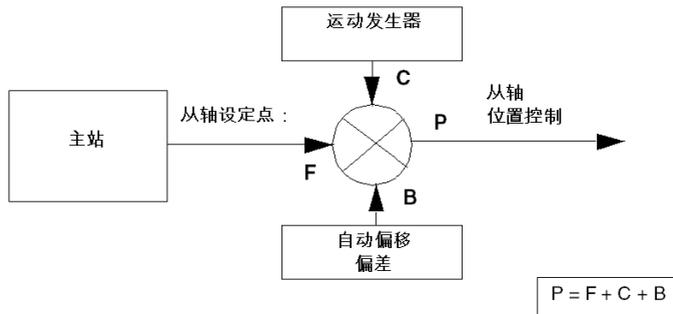
比率模式

比率是通过输入分子和分母定义的。比率模式（传动）能管理切换（当前值，设定点）、偏移（偏差）、启动条件和跟随者停止



跟随停止和偏差

这些功能能够向从轴设定点添加运动或自动偏移。这使得具有后续激活的运动可预测，或者在随后停用跟随之后继续运动。



可访问的服务

通道 21 到 24 使得下面的服务可供使用（请参见与应用专用功能关联的显式交换语言对象，第 377 页）：

- READ_PARAM
- WRITE_PARAM
- WRITE_CMD（Move 除外）
- SAVE_PARAM
- RESTORE_PARAM
- READ_STS
- MOD_PARAM（参见第 377 页），仅用于 TSX CSY 164 模块。此功能不可用于 TSX CSY 84 模块。

" 凸轮配置文件 " 功能

概览

" 凸轮配置文件 " 功能由 TSX CSY 84 模块的通道 25 到 31 执行。凸轮配置文件由从轴组使用，以根据称为凸轮配置文件的点表跟随主轴。

凸轮配置文件是一个 2 列的点表：

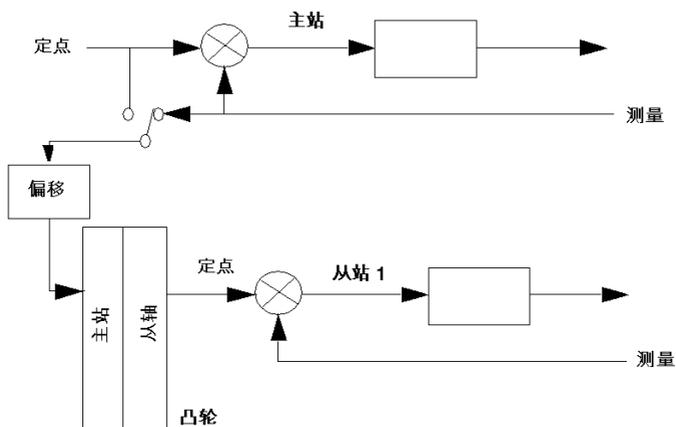
- 一列定义从轴组中主轴（通常为虚轴）的位置
- 一列定义与主轴位置关联的从轴位置

两个连续点之间的内插可推导出该表未提供的位置。内插可以是线性的或立方的。

主轴的位置必须是递增的。例如，它们不能从 360 度转到 0 度。这对应于递减位置。

凸轮配置文件图

凸轮配置文件控制的从轴的图示如下：

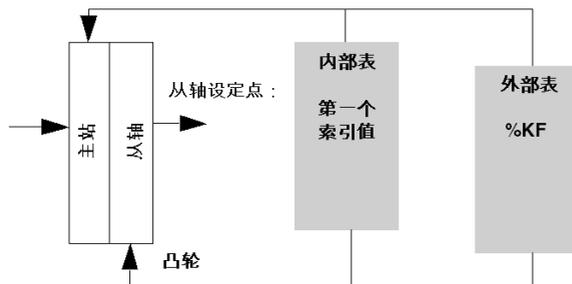


凸轮

凸轮的实现可通过：

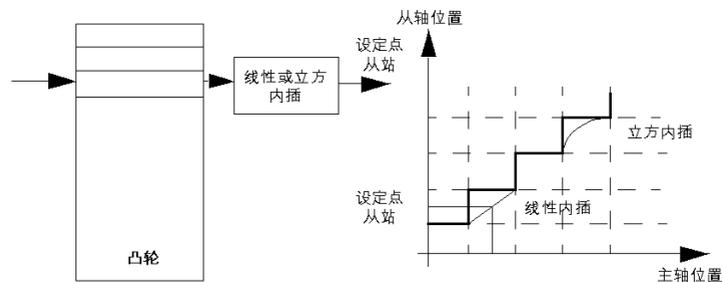
- 轴控模块中的内部表，由初始表值和固定增量定义
- 或者，轴控模块的外部表（包含在 PLC 内），定义为地址 %KF

示意图



内插

如果凸轮项对应于两个连续点之间的值，则可通过线性或立方内插（配置时定义）计算从轴设定点



可访问的服务

通道 25 到 31 使得下面的服务可供使用（请参见接口语言，第 383 页）：

- WRITE_CMD
- READ_STS
- TRF_RECIPE

本章主题

本章描述 TSX CSY 84 和 SX CSY 85 模块的配置屏幕和它们可执行的一些功能："SERCOS(r) 功能"、"实轴"、"虚轴"等。

本章包含了哪些内容？

本章包含了以下主题：

主题	页
关联 IODDT	116
模块配置	117
通道配置	120
SERCOS(r) 功能的配置（通道 0）	121
配置独立轴（通道 1 到 12）	122
外部测量轴的配置（通道 13 到 16）	126
协同轴组的配置（通道 17 到 20）	128
从轴组的配置（通道 21 到 24）	129
凸轮配置文件的配置（通道 25 到 31）	132

关联 IODDT

提示

变量编辑器用来为 TSX CSY 84 模块创建和关联 IODDT。请参考描述 IODDT 及其与应用专用模块中不同通道关联 (参见 *Unity Pro, 操作模式* ,) 的文档。

模块配置

简介

在 "配置" 模式下, 可以定义每个 TSX CSY 84 模块路径的操作特性。

此模式在离线模式下可访问。

TSX CSY 84 模块占据机架中的两个位置。为了使用此模块, 有必要选择为固定该模块而分配的两个插槽中最靠右的位置。

添加模块

下面的对话框用于将 TSX CSY 84 模块添加到配置中:

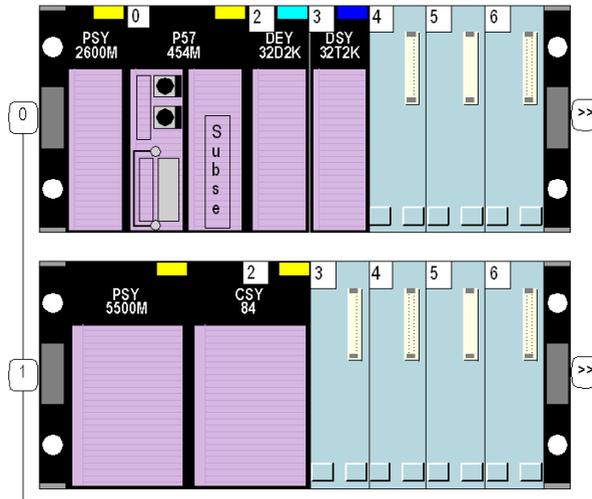


操作如下:

- 选择 **运动**,
- 选择模块的参考号 (TSX CSY 84),
- 然后拖放至所选插槽。

声明模块

为了表明模块已在机架的配置中声明，该模块与其参考号一起出现在机架中。



删除模块

若要将模块从其位置上删除：

- 单击模块将其选中，
- 按 键，这将打开一个对话框，
- 确认删除该模块。

应用专用通道

智能模块（计数模块、轴命令模块等）的所有通道都称为“应用专用通道”。
TSX CSY 84 模块的 32 个通道都是应用专用通道。

PLC 所拥有的应用专用通道的数量取决于其电源及其参考号。若要了解每个处理器的属性，请参考处理器文档。

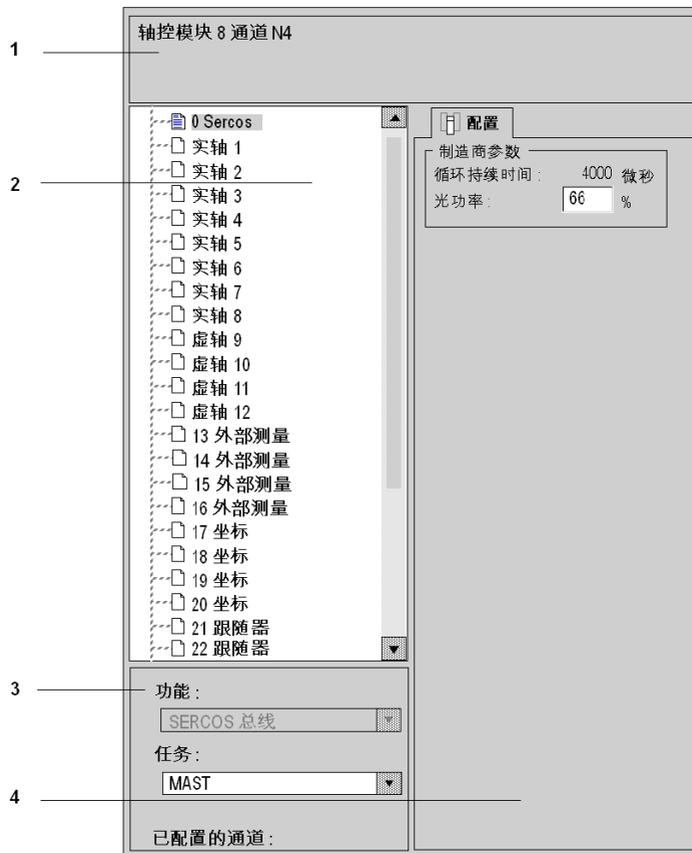
访问模块配置屏幕

要访问模块配置屏幕，请双击其图形表示 (CSY 84) 或：

- 选择该模块（单击它），
- 从**编辑**下拉菜单中激活**打开模块**命令。

配置屏幕

模块配置屏幕如下：



配置屏幕描述

字段	描述
1	此区给出了模块及其在 PLC 中的地理地址的目录参考（机架编号和在机架中的位置）。
2	此区域名为 通道区域 ，用于选择要配置的通道。
3	此区域名为 常规参数区域 ，用于配置与所选通道关联的常规参数。
4	此区域取决于所选择的选项卡：在此例中，它是在区域 2 中选择的通道的配置区域。在本地模式下只提供配置选项卡。

通道配置

简介

配置通道涉及到定义与该通道关联的应用专用功能的参数：

- 通道 0: SERCOS(r) 功能
- 通道 1 至 12: 独立轴（实轴或虚轴）
- 通道 13 至 16: 远程轴
- 通道 17 至 20: 协调轴组
- 通道 21 至 24: 从轴组
- 通道 25 至 31: 凸轮配置文件

配置检查

在验证参数条目的同时，会执行配置检查。如果存在不一致性（例如：最大加速度 < 最小加速度），则会有一条错误消息显示该类型的错误。此外，受影响的参数也会以红色显示在屏幕上。

SERCOS(r) 功能的配置（通道 0）

简介

缺省情况下，配置了管理 SERCOS(r) 总线的通道 0。该配置屏幕允许您输入光功率并指定总线参数值。

配置屏幕

TSX CSY 84 和 TSX CSY 164 模块的 SERCOS(r) 功能配置屏幕如下：



参数描述

SERCOS(r) 环路的参数在**制造商参数**字段中定义：

参数	描述
循环持续时间	SERCOS(r) 总线的循环时间：2 或 4 毫秒。缺省值为 4 毫秒。新值将在切换到 SERCOS(r) 阶段 0（SERCOS(r) 环路起始处或执行 SetCommandedPhase 功能），然后切换到阶段 4 后生效。
光功率	调整光功率，这对于第一段（TSX CSY 84 或 TSX CSY 164 模块与第一个伺服驱动器之间）是必要的。此功率（百分比形式）必须与段长度相关（请见调整图表）。缺省为 66%。

配置独立轴（通道 1 到 12）

简介

独立轴可以是实轴（通道 1 到 8）或虚轴（通道 9 到 12）。通过配置实轴可以对物理轴（它使用伺服驱动器）进行控制。在此情况下，必须确保在 TSX CSY 84 模块配置屏幕中输入的参数与在伺服驱动器配置期间定义的参数之间保持一定程度的一致性。

例如，虚轴可充当跟随器集中的主轴。这类轴的所有参数都是在轴命令模块的配置屏幕中定义的。

配置屏幕

独立轴的配置屏幕显示如下。有 5 个参数输入区域：**限制、启用位置检查、单位、比例系数和运动。**

The screenshot shows a configuration window for an independent axis. It is divided into several sections:

- 限制 (Limits):** Includes checkboxes for '位置检查' (checked) and '启用位置带' (unchecked). Below are input fields for '最大位置' (3.000000e+002), '最小位置' (-3.000000e+002), '最高速度' (1.000000e+005), '最大加速度' (1.000000e+005), and '最大减速度' (1.000000e+004). A '公差' (tolerance) field is set to 0.000000e+000.
- 单位 (Units):** A '类型' (type) dropdown is set to '线性' (linear). Below are dropdowns for '位置' (毫米), '速度' (毫米/秒), and '加速度' (毫米/平方秒).
- 比例系数 (Ratio Coefficient):** Fields for '分子' (1.000000e+000) and '分母' (1.000000e+000).
- 运动 (Motion):** Includes a '模数' (checked) checkbox. Below are input fields for '最大模数' (2.000000e+001), '最小模数' (-2.000000e+001), '正确位置带' (7.000000e+000), '加速度' (1.000000e+000), and '减速度' (1.000000e+000). The '加速度类型' (acceleration type) dropdown is set to '梯形 125%'.

"限制" 区域参数

描述

参数	描述
位置检查	对于受限机器，此复选框可用于启用位置限制检查。将轴位置与在配置期间定义的位置限制进行比较。当轴达到其某个限制时，其运动停止并生成错误。 对于无限轴，不应选择此框。
最大位置	最大位置限制。此值作为浮点数输入。
最小位置	最小位置限制。此值作为浮点数输入。
最高速度	最大允许速度。此值与在伺服驱动器（实轴）中定义的值无关。此值作为浮点数输入。将最高速度设置为 0 将禁用速度监视。
最大加速度	最大允许加速度。此值与在伺服驱动器（实轴）中定义的值无关。此值作为浮点数输入。
最大减速度	最大允许减速度。此值与在伺服驱动器（实轴）中定义的值无关。此值作为浮点数输入。

" 启用位置检查 " 区域参数

描述

参数	描述
启用	此复选框用于启用位置检查。 当该轴禁用时： <ul style="list-style-type: none"> ● 如果其运动小于公差，则当轴重新激活时，将返回到它上一次的位置。 ● 如果运动大于公差，则当轴重新激活时，它将保留在其新位置上。
公差	监视窗口的值。此值作为浮点数输入。

" 单位 " 区域参数

描述

参数	描述
类型	用于表示位置、速度和加速度读数的物理单位的类型：角度、线性、线性英制或编码器点。
位置	位置单位 <ul style="list-style-type: none"> ● 角度：千分之一弧度、弧度、度、弧分、转 ● 线性：微米、毫米、厘米、米 ● 线性英制：英寸、英尺、码、英里 ● 编码器点：点
速度	速度单位。 <ul style="list-style-type: none"> ● 角度：千分之一弧度 / 秒、弧度 / 秒、弧度 / 分、度 / 秒、度 / 分、弧分 / 秒、转 / 秒、转 / 分 ● 线性：微米 / 秒、毫米 / 秒、毫米 / 分、厘米 / 秒、厘米 / 分、米 / 秒、米 / 分 ● 线性英制：英寸 / 秒、英寸 / 分、英尺 / 秒、英尺 / 分、码 / 分、英里 / 秒 ● 编码器点：点 / 毫秒、点 / 秒、点 / 分
加速度	加速度单位 <ul style="list-style-type: none"> ● 角度：千分之一弧度 / 秒²、弧度 / 秒²、度 / 秒²、弧分 / 秒²、转 / 秒²、转 / 分 / 秒 ● 线性：微米 / 秒²、毫米 / 秒²、厘米 / 秒²、米 / 秒²、米 / 分²、克 / 秒 ● 线性英制：英寸 / 秒²、英尺 / 秒²、码 / 分²、英里 / 秒² ● 编码器点：点 / 毫秒²、点 / 秒²

" 比例系数 " 区域参数

描述

参数	描述
分子	比例系数分子。此值作为浮点数输入。
分母	比例系数分母。此值作为浮点数输入。

"运动" 区域参数

描述

参数	描述
模数	对于无限轴，此复选框可用于激活模数功能。
最大模数	模数上限。此值作为浮点数输入。
最小模数	模数下限。此值作为浮点数输入。
正确位置带	正确位置带的值。此值作为浮点数输入。
加速度	为运动定义的加速度值。此值作为浮点数输入。
减速度	为运动定义的减速度值。此值作为浮点数输入。
加速度类型	加速度类型：矩形 100%、梯形 125%、梯形 150%、梯形 175% 或三角型 200%。

重调

对于以角度单位（度）定义位置的伺服驱动器，模块将根据其参考（以转数计）和其速度（以转 / 秒计）执行重调操作。

例如，如果轴配置为比例系数为 1/1 的角度轴：

- 位置单位为转数，速度单位为转 / 秒：位置为 1 和速度为 1 的增量运动将在 1 秒内执行 1 转。
- 位置单位为度，速度单位为转 / 分：位置为 360 和速度为 60 的增量运动将在 1 秒内执行 1 转。

更改单位类型

当单位类型相对于伺服驱动器（保持以转数为单位）发生变化时，模块的参照单位为毫米（线性类型）和英寸（线性英制类型）。模块进行的转换等于：

- 伺服驱动器的 1 转 = 模块的 1 毫米（线性类型）
- 伺服驱动器的 1 转 = 模块的 1 英寸（线性英制类型）

例如，如果轴配置为比例系数为 1/1 的线性：

- 位置单位为毫米，速度单位为毫米 / 秒：位置为 1 和速度为 1 的增量运动将在 1 秒内执行 1 转。
- 位置单位为毫米，速度单位为毫米 / 秒：位置为 1000 和速度为 1000 的增量运动将在 1 秒内执行 1 转（或 1 秒内 1000 毫米）。

使用比例系数

请考虑通过轴移动传送带的项目；例如，1 次轴转将传送带移动 100 毫米。如果希望以毫米表示位置：

- 位置单位将配置为毫米，速度为毫米 / 秒，比例系数分子将等于 100，而分母将保持为 1。
- 位置 1000 且速度 1000 的增量运动将在 1 秒内移动传送带 1 米（或 10 次轴转）（速度 1000 毫米 / 秒）。

外部测量轴的配置（通道 13 到 16）

简介

外部测量通道用于将外部位置信息返回到模块。配置外部测量涉及到配置实轴和虚轴，其中只有位置信息有效。

配置屏幕

外部测量的配置屏幕如下。有 3 个参数输入区域：**比例系数**、**模数**和**单位**，以及一个用来选择地址的字段。

地址

此字段允许您定义位置信息的来源：

- 虚拟：位置由项目程序写入输出寄存器 %QDr.m.c.0 ，
- 1 到 8：位置在 " 地址 " 字段选项（1 到 8）中定义的实轴的辅助位置输入上读取。

" 比例系数 " 区域参数

描述

参数	描述
分子	比例系数分子。此值作为浮点数输入。
分母	比例系数分母。此值作为浮点数输入。

" 模数 " 区域参数

描述

参数	描述
活动	对于无限轴，此复选框用于激活模数交叉授权。
最大模数	模数上限。此值作为浮点数输入。
最小模数	模数下限。此值作为浮点数输入。

" 单位 " 区域参数

描述

参数	描述
位置	位置单位：毫米、毫米、厘米、米、英寸、英尺、码、密耳、千分之一弧度、弧度、度、弧分、转、脉冲

内部单位

内部单位对于线性（公制）类型为毫米，英制线性类型为英寸，角度类型为转。选择单位度表示对于比例系数 1/1，编码器 1 个脉冲的增量将更改位置 360 度。对于单位厘米和比例系数 1/1，编码器 1 个脉冲的增量将更改位置 1/10 毫米。

协同轴组的配置（通道 17 到 20）

简介

协同轴组是相互协调运动的实轴或虚轴的集合。要配置协同轴组，有必要事先配置属于该组的独立轴。

配置屏幕

协同轴组的配置屏幕如下。此屏幕允许您定义将相互协同的轴的列表（最多 8 个轴）。

轴 X 到轴 E

这 8 个字段允许您选择将相互协同操作的轴（轴 1 到 12）。值 N 表示该轴不参与协同。协同的主轴是在轴 X 字段中定义的轴。

注意：这些轴必须按顺序配置。例如，不允许轴 X 的值为 N，而轴 Y 的值为 1。

协同轴组的轴的加速配置文件缺省为矩形。虽然分别采用的每个轴都可能有一个已定义的配置文件，但可以使用 SetAccelType 命令更改协同轴组的配置文件。

"特殊功能" 区域的参数

描述

参数	描述
确认	此复选框用于启用特殊功能的激活。
功能编号	此字段用于输入要激活的特殊功能的编号。
地址表 %K	此字段用于输入参数表的起始地址，该表包含与该特殊功能关联的参数。

从轴组的配置（通道 21 到 24）

简介

跟随器集是包含若干从轴（最多 6 个从轴）的一组轴，从轴跟踪主轴的运动。
 主轴可为实轴、虚轴，或者外部设定点。从轴为实轴或虚轴。

配置屏幕

从轴组的配置屏幕显示如下。它有 7 个区域，这些区域用来配置主轴和 6 个可能的从轴。



主轴

此字段用于定义主轴的编号（1 到 16）。值为 N 表示尚未选择主轴。

"从轴"区域

6 个从轴区域（1 到 6）完全相同。这些区域只有在定义了主轴编号的情况下才会显示为活动状态。

"从轴" 区域参数

描述

参数	描述
从轴 1 (到 6)	用来设置从轴的编号 (轴 1 到 12)。
测量	当此按钮选中时, 从轴跟随主轴的测量位置。此按钮与 "设定点" 按钮互斥。
设定点	当此按钮选中时, 从轴跟随主轴设定点的位置。此按钮与 "测量" 按钮互斥。
传动系统	当选中此按钮时, 从轴以比率模式跟随主轴; 也就是说, 按照 "比率" 字段确定的比率。此按钮与 "凸轮" 按钮互斥。
凸轮	当此按钮选中时, 从轴以凸轮模式跟随主轴; 也就是说, 遵循编号为 "编号" 字段内所选号码的凸轮配置文件。此按钮与 "传动系统" 按钮互斥。
比率	在 "比率" 模式下, 这两个字段用于输入分子和分母, 它们定义主轴和从轴之间的比率。这些值是作为浮点数输入的。
启动	用来选择启动条件: <ul style="list-style-type: none"> ● 立即 ● 当以偏移值递增的主轴位置达到在 "触发" 字段中所定义的阈值时 (CW 行程) ● 当主轴位置加上偏移值达到在 "触发" 字段中所定义的阈值时 (CCW 方向) ● 当主轴位置加上偏移值大于等于在 "触发" 字段中所定义的阈值时 ● 当主轴位置加上偏移值小于等于在 "触发" 字段中所定义的阈值时
编号	在凸轮模式下, 此字段用来选择凸轮配置文件编号 (25 和 31 之间)。
偏移	在凸轮模式下, 此字段用于输入偏移值, 该值将与主轴位置相加以定义从轴位置。 从轴的结果位置将按照凸轮配置文件主轴表中的索引给出。此索引等于主轴当前位置加上偏移 (例如, 对于主轴坐标, 凸轮配置文件定义为 0 到 1000)。要开始跟随等于 100000 的主轴位置, 偏移值应等于 -100000.0。 例如, 此偏移用于从同一凸轮配置文件定义正弦和余弦函数。此值作为浮点数输入。
偏差保持 (余差)	当此框选中时: <ul style="list-style-type: none"> ● 动态偏移自动与主轴位置相加, 以定义从轴的位置。 ● 当与主轴的链接被移除时, 附加从轴运动不会中止。

参数	描述
触发器	当启动条件取决于主轴位置相对于阈值的关系时，此字段用来输入阈值。此值作为浮点数输入。 当主轴的当前位置 + 偏移 > (或 <、>、<) 阈值 (触发值) 时，将发生激活。
跟随时停止	当此框选中时，如果确认主轴和从轴之间的链接，则将根据自动确定的减速配置文件停止从轴任何可能的附加运动。
停止主轴 / 故障	当此框选中时，当在主轴和从轴之间出现等高线差异 (故障) 时，主轴将停止。

从轴位置

当轴为从轴时，其位置只取决于它所跟随的主轴的位置。此时，将忽略该轴的配置参数 (位置限制、最高速度、最大加速度等)。若要保护项目，请在伺服驱动器中配置这些 (安全) 参数。

比例系数

在从轴组中，当用于轴的单位类型 (线性、角度等) 相同时，比例系数与单位无关。例如，如果主轴以米配置，而从轴以厘米配置 (不同单位，但为相同类型：线性)，并且如果使用的比例系数为 1/1，则这意味着主轴每行进 1 毫米，从轴也将移动 1 毫米。

如果主轴和从轴单位类型不同，则这些单位必须转换为单位类型的参照单位 (线性类型为毫米，线性英制类型为英寸，角度类型为转数)。例如，如果主轴的单位配置为米，而从轴的单位配置为转数 (不同类型的单位：线性和角度)，并且主轴移动 1 米时从轴完成 1 转，则比例系数必须如下定义：

- 1 米 = 1000 毫米 (线性类型的参照单位)
- 1 转 = 1 转 (角度类型的参照单位)

因此，比例系数 = 1000/1 (主轴行进 1000 毫米时，从轴完成 1 转)。

凸轮配置文件的配置（通道 25 到 31）

简介

凸轮配置文件用于通过脉冲表来定义从轴相对于主轴位置的位置。

模块 TSX CSY 84/164

在 TSX CSY 84 和 TSX CSY 164 模块上，凸轮配置文件配置是在通道 25 到 31 上实现的。

配置屏幕

凸轮配置文件的配置屏幕有 3 个允许您定义主轴和从轴点表的区域。

The screenshot shows a configuration window with the following fields and options:

- 线性** (Linear) and **立方** (Cubic) radio buttons.
- 脉冲数** (Pulse count) input field with value 0.
- 主轴增量** (Main axis increment) section:
 - 单位 (Unit): 微米 (micrometers)
 - 固定 (Fixed) radio button selected, 起始值 (Start value) input field: 0.0000000e+000
 - 变量 (Variable) radio button, 递增 (Increment) input field: 0.0000000e+000
- 从轴增量** (Sub-axis increment) section:
 - 单位 (Unit): 微米 (micrometers)
 - 固定 (Fixed) radio button selected, 地址表 %KF (Address table %KF) input field: 0
 - 变量 (Variable) radio button

表参数

描述

参数	描述
线性	当选中此按钮时，两个连续凸轮配置文件点之间的内插是线性的。此按钮与 "立方" 按钮互斥。
立方	当选中此按钮时，两个连续凸轮配置文件点之间的内插是立方的。此按钮与 "线性" 按钮互斥。
点数	此字段可用于输入用来定义凸轮配置文件的点数。

主轴增量区域的参数

描述

参数	描述
单位	用来定义主轴增量使用的单位。所选单位可以是轴定义的单位的子单位（例如，轴为厘米，增量为毫米）。
固定	当此按钮选中时，两个连续凸轮配置文件点之间的增量总是相同的。此按钮与“可变”按钮互斥。
起始值	对于固定增量，此字段可用于输入凸轮配置文件的起始值。此值作为浮点数输入。
增量	对于固定增量，此字段可用于定义增量的值。此值作为浮点数输入。
可变	当选中此按钮时，两个连续凸轮配置文件点之间的增量可变。这些点的值由常量字表 %KF 定义，该表长度等于点数。
地址表 %KF	对于可变增量，此字段可用于输入主轴的点表的起始地址。

注意：凸轮配置文件总是循环的。必须确保从轴表中的第一个值和最后一个值相等。

在循环运动的情况下，该表将完全描述模数，并且将添加一个附加脉冲（模数 + 1），该脉冲的从轴值将为表中给出的第一个值。

例如，对于模数 360°，值为 0 到 359，该表如下：表（主轴、从轴）：(0, **x0**); (1, **x1**); (2, **x2**); ...; (359, **x359**); (360, **x0**)

在线性运动的情况下，如果从轴表中的最后一个值不等于第一个值，则会添加附加点（例如，最后的点将重复多次，从轴值将逐渐逼近表中的第一个值）。

从轴增量区域的参数

描述

参数	描述
单位	用于定义从轴增量使用的单位。所选单位可以是轴定义的单位的子单位（例如，轴为厘米，增量为毫米）。
固定	当此按钮选中时，两个连续凸轮配置文件点之间的增量总是相同的。此按钮与“可变”按钮互斥。
起始值	对于固定增量，此字段可用于输入凸轮配置文件的起始值。此值作为浮点数输入。
增量	对于固定增量，此字段可用于定义增量的值。此值作为浮点数输入。
可变	当选中此按钮时，两个连续凸轮配置文件点之间的增量可变。这些点的值由常量字表 %KF 定义，该表长度等于点数。
地址表 %KF	对于可变增量，此字段可用于输入从轴的点表的起始地址。

TSX CSY 84 和 TSX CSY 164 模块 编程

12

本章的目标

本章描述与轴运动关联的不同功能。

本章包含了哪些内容？

本章包含了以下部分：

节	主题	页
12.1	运动状态位	136
12.2	运动控制功能	170
12.3	运动功能	185
12.4	序列停止后的运动功能	196
12.5	实际速度 / 位置功能	207
12.6	跟随功能	214

12.1 运动状态位

本节主题

本节描述可通过属于通道 IODDT (位 %Ir.m.c.0 到 %Ir.m.c.31) 的隐式交换或通过 GetStatus 功能访问的运动状态位。

本节包含了哪些内容?

本节包含了以下主题:

主题	页
RAMPING 位 (%Ir.m.c.0)	138
STEADY 位 (%Ir.m.c.1)	139
STOPPING 位 (%Ir.m.c.2)	140
PROFILE_END 位 (%Ir.m.c.3)	141
IN_POSITION 位 (%Ir.m.c.4)	142
AXIS_HOMING 位 (%Ir.m.c.5)	143
AXIS_HOMED 位 (%Ir.m.c.6)	144
AXIS_NOT_FOLLOWING 位 (%Ir.m.c.7)	145
HOLDING 位 (%Ir.m.c.8)	146
RESUMING 位 (%Ir.m.c.9)	147
DRIVE_ENABLED 位 (%Ir.m.c.10)	148
DRIVE_DIAG 位 (%Ir.m.c.11)	149
DRIVE_WARNING 位 (%Ir.m.c.12)	150
DRIVE_FAULT 位 (%Ir.m.c.13)	151
DRIVE_DISABLED 位 (%Ir.m.c.14)	152
AXIS_SUMMARY_FLT 位 (%Ir.m.c.15)	153
AXIS_COMM_OK 位 (%Ir.m.c.16)	154
AXIS_IS_LINKED 位 (%Ir.m.c.17)	155
AXIS_IN_CMD 位 (%Ir.m.c.18)	156
AXIS_AT_TARGET 位 (%Ir.m.c.20)	157
AXIS_POS_LIMIT 位 (%Ir.m.c.21)	158
AXIS_NEG_LIMIT 位 (%Ir.m.c.22)	159
AXIS_WARNING 位 (%Ir.m.c.23)	160

主题	页
BIAS_REMAIN 位 (%Ir.m.c.24)	161
AXIS_MANUAL_MODE BIT (%Ir.m.c.25)	162
DRIVE_REALTIME_BIT1 位 (%Ir.m.c.26)	163
DRIVE_REALTIME_BIT2 位 (%Ir.m.c.27)	164
AXIS_HOLD 位 (%Ir.m.c.28)	165
AXIS_HALT 位 (%Ir.m.c.29)	166
AXIS_FASTSTOP 位 (%Ir.m.c.30)	167
AXIS_READY 位 (%Ir.m.c.31)	168
CONF_OK 位 (%Ir.m.c.32)	169

RAMPING 位 (%Ir.m.c.0)

描述

此位（状态 1）指示有序运动配置文件为加速或减速。

应用于实轴或虚轴功能

加速或减速配置文件类型。

应用于协同轴组

加速或减速配置文件类型。

AXIS_IS_LINKED 成员轴也将其 RAMPING 位设置为 1。

应用于从轴组

如果成员轴中的一个或多个 RAMPING 位设置为 1，则此位设置为 1。

在比率模式下，当从轴的比率更改为新值时，成员轴 RAMPING 位设置为 1。

应用于 SERCOS(r) 功能

如果轴中的 RAMPING 位设置为 1，则此位设置为 1。

STEADY 位 (%Ir.m.c.1)

描述

此位（状态 1）指示运动配置文件要求匀速。

应用于实轴或虚轴功能

匀速配置文件类型。

应用于协同轴组

匀速配置文件类型。

AXIS_IS_LINKED 成员轴也将其 STEADY 位设置为 1。

应用于从轴组

如果成员轴中的所有 STEADY 位设置为 1，则此位设置为 1。

如果从轴的运动只是原样仿照主轴运动（无附加运动），则 STEADY 位保持清空。

应用于 SERCOS(r) 功能

如果成员轴中的所有 STEADY 位设置为 1，则此位设置为 1。

STOPPING 位 (%Ir.m.c.2)

描述

此位（状态 1）指示有序运动配置文件为带有运动停止的减速类型。

在 IN_POSITION (%Ir.m.c.4) 位设置为 1 之前，它保持设置为 1。

当 PROFILE_END (%Ir.m.c.3) 位转为状态 1 时，STOPPING 位设置为 0。

注意：在由于复位 ALLOW_MOVE 或 ALLOW_RESUME 位而减速的情况下，STOPPING 位不会设置为状态 1。

应用于实轴或虚轴功能

带有运动停止的减速类型运动配置文件。

应用于协同轴组

带有运动停止的减速类型运动配置文件。

AXIS_IS_LINKED 成员轴也将其 STOPPING 位设置为 1。

应用于从轴组

如果成员轴中的一个或多个 STOPPING 位设置为 1，则此位设置为 1。

STOPPING 位不会检查轴是否稳定，因为轴可能响应其主轴的运动而运动。

应用于 SERCOS(r) 功能

如果轴中的 STOPPING 位设置为 1，则此位也设置为 1。

PROFILE_END 位 (%Ir.m.c.3)

描述

此位（状态 1）指示所要求的运动已结束。

轴在 IN_POSITION (%Ir.m.c.4) 设置为 1 或 STOPPING (%Ir.m.c.2) 清空之前可能仍然静止。

当轴禁用或达到停止时，PROFILE_END 设置为 1。

应用于实轴或虚轴功能

运动配置文件已将其最后一条命令发送到驱动器。轴可能仍然静止。

应用于协同轴组

协同轴组的运动配置文件已将其最后一条命令发送到所有成员轴驱动器。

AXIS_IS_LINKED 成员也将其 PROFILE_END 位设置为 1。成员轴可能仍然静止。

应用于从轴组

如果所有轴成员 PROFILE_END 位都设置为 1，则此位也设置为 1。这不表示成员轴不在运动。它们可能在跟随其主轴的运动。

应用于 SERCOS(r) 功能

如果轴中的所有 PROFILE_END 位都设置为 1，则此位也设置为 1。

IN_POSITION 位 (%Ir.m.c.4)

描述

此位（状态 1）指示轴位置处于正确位置带中，并且 STOPPING (%lr.m.c.2) 位为 1。

当轴禁用或达到停止时，IN_POSITION 设置为 0。

适用于实轴功能

轴位于正确位置带中（STOPPING 位为 1）。

适用于虚轴功能

一旦 PROFILE_END (%lr.m.c.3) 设置为 1，也就设置了 IN_POSITION。

应用于协同轴组

正确位置带（STOPPING 位为 1）中的所有轴以及不同成员轴的 IN_POSITION 位 (AXIS_IS_LINKED) 都设置为 1。

应用于从轴组

此位在下列情况下设置为 1：

- 如果所有成员轴 IN_POSITION 位设置为 1
- 跟随被禁用

注意：当成员轴跟踪主轴时，无论其是否运动，其 IN_POSITION 位都设置为 0。

应用于 SERCOS(r) 功能

如果所有成员轴 IN_POSITION 位都设置为 1，则此位设置为 1。

AXIS_HOMING 位 (%Ir.m.c.5)

描述

此位（状态 1）指示轴正在执行参考点（参见第 179 页）类型功能。

适用于实轴功能

参考点类型功能。

适用于虚轴功能

无意义

适用于协同轴组、从轴组或 SERCOS(r) 环路功能

如果一个或多个成员轴 AXIS_HOMING 位 (AXIS_IS_LINKED) 设置为 1，则此位设置为 1。

AXIS_HOMED 位 (%Ir.m.c.6)

描述

此位（状态 1）指示已正确执行了参考点功能。

适用于实轴功能

轴位置读数以源位置作为参考。

此位在以下情况下不会设置为 1：

- 从未执行或未能完成参考点功能
- SERCOS(r) 网络中断或驱动器检测到位置反馈设备中存在故障
- 在 Unhome 功能之后

注意：此位在 ForcedHomed 功能之后设置为 1。

适用于虚轴功能

无意义。

适用于协同轴组、从轴组或 SERCOS(r) 环路功能

如果一个或多个成员轴 AXIS_HOMED 位 (AXIS_IS_LINKED) 设置为 1，则此位设置为 1。

AXIS_NOT_FOLLOWING 位 (%Ir.m.c.7)

描述

此位（状态 1）指示驱动器在执行特殊操作（例如，参考点、快速停止或停止）时忽略控制器的运动配置文件。

注意：并非所有 SERCOS(r) 驱动器上都有此位。

适用于实轴功能

在驱动器执行特殊操作时，控制器受抑止。

适用于虚轴功能

如果轴禁用，则该位设置为 1，如果轴启用，则设置为 0。

适用于协同轴组、从轴组或 SERCOS(r) 环路功能

如果一个或多个成员轴 AXIS_NOT_FOLLOWING 位设置为 1，则此位设置为 1。

HOLDING 位 (%Ir.m.c.8)

描述

此位（状态 1）指示该轴：

- 停止不动，
- 或者运动减速停止。

应用于实轴或虚轴功能

轴已停止，或者正在停止。

应用于协同轴组

协同轴组中的轴已经停止，或者正在停止过程中。

AXIS_IS_LINKED 成员轴也将其 HOLDING 位设置为 1。

应用于从轴组或 SERCOS(r) 环路功能

如果一个或多个成员轴 HOLDING 位设置为 1，则此位设置为 1。

RESUMING 位 (%Ir.m.c.9)

描述

此位（状态 1）指示轴在暂停后继续。轴再次加速以实现运动配置文件。

应用于实轴或虚轴功能

轴在暂停后继续。

应用于协同轴组

协同轴组在暂停之后轴继续。

AXIS_IS_LINKED 成员轴也将其 RESUMING 位设置为 1。

应用于从轴组或 SERCOS(r) 环路功能

如果一个或多个成员轴 RESUMING 位设置为 1，则此位设置为 1。

DRIVE_ENABLED 位 (%Ir.m.c.10)

描述

此位（状态 1）指示驱动器已激活并且关联的电机已加电。

注意：如果 AXIS_READY (%Ir.m.c.31) 位设置为 1，则可能正在轴上执行运动命令。

适用于实轴功能

驱动器已激活，电机已加电。

适用于虚轴功能

轴沿 CONTROL_ENABLE 位上升沿激活。

适用于协同轴组、从轴组或 SERCOS(r) 环路功能

如果所有成员轴 DRIVE_ENABLED 位 (AXIS_IS_LINKED) 设置为 1，则此位设置为 1。

DRIVE_DIAG 位 (%Ir.m.c.11)

描述

此位（状态 1）指示对 3 类 SERCOS(r) 诊断位的修改。

IDN 0013 变量包含当前信息。

读取 IDN 0013 变量时，此位设置为 0。

注意：3 类 SERCOS(r) 位不受 SERCOS(r) 标准 IDN 0013 的影响。

适用于实轴功能

3 类诊断位已更改（信息）。

适用于虚轴功能

无意义

适用于协同轴组、从轴组或 SERCOS(r) 环路功能

如果一个或多个成员轴 DRIVE_DIAG 位 (AXIS_IS_LINKED) 设置为 1，则此位设置为 1。

DRIVE_WARNING 位 (%Ir.m.c.12)

描述

此位（状态 1）指示对 2 类 SERCOS(r) 诊断位的修改。
IDN 0012 变量包含当前信息。
读取 IDN 0012 变量时，此位设置为 0。

适用于实轴功能

2 类诊断位已更改（警告）。

适用于虚轴功能

无意义

适用于协同轴组、从轴组或 SERCOS(r) 环路功能

如果一个或多个成员轴 DRIVE_WARNING 位 (AXIS_IS_LINKED) 设置为 1，则此位设置为 1。

DRIVE_FAULT 位 (%Ir.m.c.13)

描述

此位（状态 1）指示对 1 类 SERCOS(r) 诊断位的修改。

IDN 0011 变量包含当前信息。

此位在下列情况下设置为 0:

- 读取 IDN 0011 变量时
- 从 CONTROL_CLEAR_FLT 上升沿

适用于实轴功能

1 类诊断位已更改（缺省）。

适用于虚轴功能

无意义

适用于协同轴组、从轴组或 SERCOS(r) 环路功能

如果一个或多个成员轴 DRIVE_FLT 位 (AXIS_IS_LINKED) 设置为 1，则此位设置为 1。

DRIVE_DISABLED 位 (%Ir.m.c.14)

描述

此位（状态 1）指示驱动器已停用，并且关联的电机已断电。

适用于实轴功能

驱动器已停用，电机已断电。

适用于虚轴功能

轴已在 ALLOW_ENABLE 位复位后停止。

适用于协同轴组、从轴组或 SERCOS(r) 环路功能

如果所有成员轴 DRIVE_DISABLED 位 (AXIS_IS_LINKED) 都设置为 1，则此位设置为 1。

AXIS_SUMMARY_FLT 位 (%Ir.m.c.15)

描述

此位（状态 1）指示：

- 故障伺服驱动器 (DRIVE_FLT)
- SERCOS(r) 通讯故障 (AXIS_COM_OK 为 0)
- 故障运动配置文件控制器 (AXIS_POS_LIMIT 或 AXIS_NEG_LIMIT)

它也可能指示从轴组或协同组故障（所有 TSX CSY 84 和 TSX CSY 164 模块通用的新功能）：

- 具有一个故障轴的确认组将通过由 GET_MOTION_FAULT 功能提供的数据标记为故障 (%Ir.m.c.15 为 1)，即：组成员故障 (MOTION_FAULT 字的位 02 (MF_MEMBER_FAULT))。

以及故障和警告传播（所有 TSX CSY 84 和 TSX CSY 164 模块通用的新功能）：

- 实轴故障

在 GetMotionFault (参见第 343 页) 功能的帮助下可访问有关故障的信息。

注意： 状态为 0 的 ALLOW_NOT_FLT 位将造成用户故障。

应用于实轴或虚轴功能

运动故障。

应用于协同轴组、从轴组或 SERCOS(r) 环路功能

如果一个或多个成员轴 AXIS_SUMMARY_FLT 位 (AXIS_IS_LINKED) 设置为 1，则此位设置为 1。

AXIS_COMM_OK 位 (%Ir.m.c.16)

描述

此位（状态 1）指示驱动器已成功加电，并且运动控制器和驱动器之间的通讯处于活动状态。

适用于实轴功能

运动控制器和 SERCOS(r) 驱动器之间的循环通讯已经建立。

适用于虚轴功能

无意义

应用于协同轴组或从轴组功能

如果所有成员轴 AXIS_COM_OK 位都设置为 1，则此位设置为 1。

应用于 SERCOS(r) 功能

在不同的可能情况下位为 1 的含义。

未配置轴	至少配置了一个轴
SERCOS(r) 循环通讯已建立并且可能正在配置轴。	所有 AXIS_IS_LINKED 成员轴也将其 AXIS_COMM_OK 位设置为 1。

AXIS_IS_LINKED 位 (%Ir.m.c.17)

描述

此位（状态 1）指示轴为轴组的有效的一部分。该轴正响应来自 Co-ordinatedGroup（协同轴组）或 SlaveGroup（从轴组）的命令。

注意：在从轴集中，主轴不是成员轴。

应用于实轴和虚轴功能

协同轴组或从轴组的有效部分。

应用于协同轴组

非有效位

当协同轴组激活或正处于活动状态时，成员轴 AXIS_IS_LINKED 位设置为 1。

成员轴按照协同轴组的运动配置文件进行运动。

注意：不允许对协同轴组的成员轴直接进行操作。

应用于从轴组

如果所有成员从轴位 (AXIS_IS_LINKED) 设置为 1，则此位设置为 1。

成员轴根据跟随器比率或凸轮配置文件，按照从轴组主轴的运动配置文件进行运动。

注意：允许对成员轴发出运动命令，并且运动命令将添加到跟随主轴的运动而产生的运动中。

应用于 SERCOS(r) 功能

如果一个或多个 AXIS_IS_LINKED 位设置为 1，则此位设置为 1。

AXIS_IN_CMD 位 (%Ir.m.c.18)

描述

此位（状态 1）指示轴正处在活动状态，并且可接收命令。

应用于实轴和虚轴功能

活动轴。

应用于协同轴组

此位在下列情况下设置为 1：

- 协同轴组已激活
- 成员轴 (AXIS_IS_LINKED) 尚未脱离协同轴组的控制

成员轴不响应发送给它们的命令。

协同轴组 AXIS_IN_CMD 位在 ALLOW_AQUIRE 或 ALLOW_ENABLE 位复位后设置为 0。

注意：运动命令是发给协同轴组的，而不是发给成员轴的。

应用于从轴组

此位在下列情况下设置为 1：

- 协同轴组已激活
- 成员轴尚未脱离协同轴组的控制

成员轴也将其 AXIS_IN_CMD 位设置为 1，因为成员轴将响应发送给它们的运动命令，并且将跟随主轴（如果启用跟随）。两个运动配置文件将累加。

应用于 SERCOS(r) 功能

无意义。

AXIS_AT_TARGET 位 (%Ir.m.c.20)

描述

此位（状态 1）指示轴的位置：

- 处于正确位置带中
- 已经达到运动配置文件的末尾（PROFILE_END 为 1）

当发出新的运动命令时，AXIS_AT_TARGET 清空。

注意：与 IN_POSITION 不同的是，AXIS_AT_TARGET 不会在轴被禁用时更改，也不会暂停动作完成后设置，除非暂停位置达到原始运动配置文件的目标位置。

应用于实轴或虚轴功能

轴处于正确位置带中，同时 PROFILE_END 为 1。

应用于协同轴组或 SERCOS(r) 环路功能

当所有成员轴 AXIS_AT_TARGET 位 (AXIS_IS_LINKED) 设置为 1 时，此位设置为 1。

应用于从轴组

如果所有成员轴 AXIS_AT_TARGET 位都设置为 1，则此位设置为 1。

注意：如果跟随主轴，则成员轴 AXIS_AT_TARGET 位设置为 0。

AXIS_POS_LIMIT 位 (%Ir.m.c.21)

描述

此位（状态 1）指示轴已经达到或超过上限（最大位置）。

此位的有效性要求 DRIVE_ENABLED 和 AXIS_HOMED 位设置为 1。

如果故障消失，此位使用当前值中的 CONTROL_CLEAR_FLT 位上升沿进行复位。

应用于实轴和虚轴功能

接受命令的轴的位置达到或超过上限。

应用于协同轴组、从轴组和 SERCOS(r) 环路功能

如果有一个成员轴 AXIS_POS_LIMIT 位 (AXIS_IS_LINKED) 设置为 1，则此位设置为 1。

AXIS_NEG_LIMIT 位 (%Ir.m.c.22)

描述

此位（状态 1）指示轴已经达到或超过下限（最小位置）。

此位的有效性要求 DRIVE_ENABLED 和 AXIS_HOMED 位设置为 1。

如果故障消失，此位使用当前值中的 CONTROL_CLEAR_FLT 位上升沿进行复位。

应用于实轴和虚轴功能

接受命令的轴的位置达到或低于下限。

应用于协同轴组、从轴组或 SERCOS(r) 环路功能

如果有一个成员轴 AXIS_NEG_LIMIT 位 (AXIS_IS_LINKED) 设置为 1，则此位设置为 1。

AXIS_WARNING 位 (%Ir.m.c.23)

描述

此位（状态为 1）指示：

- 检测到轴运动警告 (参见第 341 页)。

或者从轴组或协同组故障（所有 TSX CSY 84 和 TSX CSY 164 模块通用的新功能）：

- 带有一个故障轴的未确认组将使用由 GET_MOTION_WARNING 功能提供的数据标记为警告 (%Ir.m.c.23 为 1)，即：**组成员故障** (MOTION_WARNING 字的位 05 (MW_MEMBER_FAULT))。

Unity Pro 屏幕上显示的文本如下："组成员故障"。

以及故障和警告传播（所有 TSX CSY 84 和 TSX CSY 164 模块通用的新功能）：

- 由于故障传播而停止的轴（由组或由警告功能停止）将使用由 GET_MOTION_WARNING 功能提供的数据标记为警告 (%Ir.m.c.23 为 1)，即：**链接轴故障** (MOTION_WARNING 字的位 02 (MW_STOP_BY_SET))。
- 带有一个警告轴的确认组或未确认组将使用由**组成员警告** (MOTION_WARNING 字的位 04 (MW_MEMBER_WARNING) 功能) 提供的数据标记为警告 (%Ir.m.c.23 为 1)。

Unity Pro 屏幕上显示的文本如下："组成员警告"。

GetMotionWarning 功能显示当前运动警告。

AXIS_WARNING 位使用 CONTROL_CLEAR_FLT 上升沿设置为 0。

应用于实轴和虚轴功能

检测到轴运动警告。

应用于协同轴组

如果有一个成员轴 AXIS_WARNING 位 (AXIS_IS_LINKED) 设置为 1，则此位设置为 1。

BIAS_REMAIN 位 (%Ir.m.c.24)

描述

此位（状态 1）指示已将一个偏移添加到命令位置。

注意：此位仅涉及实轴和虚轴功能。

AXIS_MANUAL_MODE BIT (%Ir.m.c.25)

描述

此位用于通过读取以下离散量输入的值来检查当前哪种模式处于活动状态:

- %Ir.m.c.25 = 0 : 自动模式 (缺省情况)
- %Ir.m.c.25 = 1 : 手动模式

DRIVE_REALTIME_BIT1 位 (%Ir.m.c.26)

适用于实轴功能

此位反映驱动器的 "SERCOS(r) 实时 " 状态位 1 的状态。

适用于虚轴功能

无意义。

应用于协同轴组和 SERCOS(r) 轴组功能

如果一个或多个成员轴 DRIVE_REALTIME_BIT1 位 (AXIS_IS_LINKED) 设置为 1，则此位设置为 1。

DRIVE_REALTIME_BIT2 位 (%Ir.m.c.27)

适用于实轴功能

此位反映驱动器的 "SERCOS(r) 实时 " 状态位 2 的状态。

适用于虚轴功能

无意义。

应用于协同轴组、从轴组和 SERCOS(r) 环路功能

如果一个或多个成员轴 DRIVE_REALTIME_BIT2 位 (AXIS_IS_LINKED) 设置为 1，则此位设置为 1。

AXIS_HOLD 位 (%Ir.m.c.28)

描述

此位（状态 1）指示轴已按照 Hold 功能转为停止。

此位在下列情况下设置为 0：

- 下面所提到的任何一位设置为 1：
 - AXIS_HALT
 - AXIS_FAST_STOP
 - AXIS_HOMING
 - DRIVE_DISABLED
 - AXIS_SUMMARY_FLT
- 轴处于非活动状态，并且 AXIS_IN_COMMAND 位设置为 0。

应用于实轴和虚轴功能

轴停止不动。

应用于协同轴组、从轴组和 SERCOS(r) 环路功能

如果所有成员轴 AXIS_HOLD 位 (AXIS_IS_LINKED) 设置为 1，则此位设置为 1。

AXIS_HALT 位 (%Ir.m.c.29)

描述

此位（状态 1）指示轴已停止，并且不能接收运动命令。

当 ALLOW_MOVE 位设置为 0 时，此位设置为 1。

此位在下列情况下设置为 0：

- 下面所提到的任何一位设置为 1：
 - AXIS_HALT
 - AXIS_FAST_STOP
 - AXIS_HOMING
 - DRIVE_DISABLED
 - AXIS_SUMMARY_FLT
- 轴处于非活动状态，并且 AXIS_IN_COMMAND 位设置为 0。

注意：ALLOW_MOVE 位在运动控制寄存器中设置为 0。

应用于实轴和虚轴功能

轴停止，并且不可能接受运动命令。

应用于协同轴组、从轴组和 SERCOS(r) 环路功能

如果所有成员轴 AXIS_HALT 位 (AXIS_IS_LINKED) 设置为 1，则此位设置为 1。

AXIS_FASTSTOP 位 (%Ir.m.c.30)

描述

此位（状态 1）指示轴具有 "快速停止" 状态。

该位在下列情况下设置为 0:

- 当轴禁用时 (ALLOW_ENABLE)
- 轴上出现故障之后
- 轴重新激活时 (CONTROL_ENABLE)
- 如果 DRIVE_DISABLED 或 AXIS_SUMMARY_FLT 位中任一位设置为 1
- 如果轴处于非活动状态，并且 AXIS_IN_COMMAND 位设置为 0

应用于实轴和虚轴功能

轴处于快速停止状态。

应用于协同轴组、从轴组和 SERCOS(r) 环路功能

如果所有成员轴 AXIS_FASTSTOP 位 (AXIS_IS_LINKED) 设置为 1，则此位设置为 1。

AXIS_READY 位 (%Ir.m.c.31)

描述

此位（状态 1）指示轴准备接收运动命令。

轴处于活动状态，并且下列位：

- AXIS_IN_COMMAND 设置为 1
- DRIVE_DISABLED 设置为 1
- AXIS_HOMING 设置为 0
- AXIS_HOLD 设置为 0
- AXIS_HALT 设置为 0
- AXIS_FAST_STOP 设置为 0
- AXIS_NOT_FOLLOWING 设置为 0
- AXIS_SUMMARY_FLT 设置为 0

应用于实轴、虚轴和协同轴组功能

轴准备接收运动命令。

应用于从轴组

从轴集处于活动状态，并且准备接收命令以跟随主轴。

应用于 SERCOS(r) 功能

如果所有独立轴（实轴、虚轴和外部设定点轴）AXIS_READY 位已设置，则此位设置为 1。

CONF_OK 位 (%Ir.m.c.32)

描述

此位（状态 1）指示轴已配置。

注意：此位仅涉及实轴和虚轴功能。

12.2 运动控制功能

本节主题

本节描述运动控制功能。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
运动控制功能：一般信息	171
ALLOW 位和 CONTROL 位的管理	172
可在跳变沿上修改的运动控制位	173
状态更改时可修改的控制位	174
状态位和字	175
手动模式	176
Set_Functional_Mode 功能	178
回归功能	179
参考点与回归功能	181
ForcedHomed 功能	182
Unhome 功能	183
SetPosition 功能	184

运动控制功能：一般信息

运动控制功能

运动控制功能用于对给定的 MotionSet（实轴、虚轴、远程轴、从轴组、网络或协同轴组）执行以下功能：

- 复位 MotionSet 的故障状态 (ClearFault)，
- 启用或禁用 MotionSet（EnableDrive 和 DisableDrive），
- 停止 MotionSet（FastStop 和 Halt）- 立即或非立即，
- 保持或恢复 MotionSet（Hold 和 Resume），
- 将运动轴回归至主位置 (Home)，
- 将运动轴移离回归点（请参见 Forcedhome）。Unhome
- 释放和获取属于从轴组或协同轴组一部分的隔离轴（Release 和 Acquire），
- 启用或禁用从轴组（FollowOn 和 FollowOff），
- 获取和启用或释放和禁用跟随组中的从轴（Link 和 UnLink）。

执行这些操作的功能发送运动控制，然后返回到应用服务器，有时是在命令完成之前返回。

应用服务器必须监控 MotionGroup 状态，以确定命令执行是否完成。

应用服务器监控轴的状态位 (%I) 以检查运动状态。

运动控制

许多命令还通过运动控制位发送。

有些运动命令要求在其配置中有 ALLOW 位才能正确执行。

缺省情况下，ALLOW 位在复位模块时设置为 1（网络通道的 ALLOW_ENABLE 位除外），以便能够授权所有对应的运动命令。

ALLOW 位和 CONTROL 位的管理

原理

运动控制位（请参见专门与 SERCOS 模块关联的语言对象和 IODDT, 第 394 页）以如下方式进行管理：

- 启动程序时，将通道 0 的 ALLOW_ENABLE 位 (%Qr.m.c.0.26) 设置为 1
- 使用 CONTROL / ALLOW 位激活 / 停用功能：
 - 当 CONTROL 位处于上升沿并且 ALLOW 位为 1 时，将激活该功能
 - ALLOW 位为零时，将停用该功能

ALLOW 位

ALLOW 位在该位的下降沿（从状态 1 变为状态 0）激活命令。在该位处于状态 0 时，它具有一个禁止操作。当 ALLOW 位从状态 0 变为状态 1 时，它移除该抑止操作，并且允许关联的命令。

当预先用符号表示 ALLOW 位时，符号以 ALLOW_ 开头（请参见专门与 SERCOS 模块关联的语言对象和 IODDT, 第 394 页）。

注：ALLOW 位初始化为状态 1。

CONTROL 位

CONTROL 位在该位的上升沿激活命令。发出该命令；如果运动集处于允许执行命令的状态，则该命令生效。因此，在激活命令之前，必须检查运动集的运动状态。CONTROL 位具有对应的 ALLOW 位。CONTROL 位无法激活命令的一个常见原因在于：与此命令关联的 ALLOW 位或者 SERCOS(r) 通道具有 0 值。

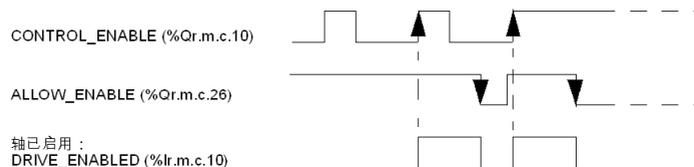
CONTROL 位的名称以 CONTROL_ 开头（请参见专门与 SERCOS 模块关联的语言对象和 IODDT, 第 394 页）。

特殊情况

脉冲周期必须大于模块循环时间（缺省 Cycle_time = 4 毫秒），以便模块“看见”该命令。

示例

ENABLE 功能



可在跳变沿上修改的运动控制位

ALLOW 位

下表列出了 ALLOW 位：

语言对象	名称	描述
%Qr.m.c.18	ALLOW_ACQUIRE	下降沿：释放受控轴。当释放轴时， AXIS_IN_COMMAND 运动状态位设置为 0。 抑止操作（状态 0）：阻止获得由此运动组控制的轴。 关联的 CONTROL 位：CONTROL_ACQUIRE
%Qr.m.c.26	ALLOW_ENABLE	下降沿：停用受控轴。当停用轴时， DRIVE_DISABLED（运动状态）位设置为 1。 抑止操作（状态 0）：阻止激活运动组。 关联的 CONTROL 位：CONTROL_ENABLE
%Qr.m.c.27	ALLOW_FOLLOW	下降沿：抑止从轴组或此类组的成员进行跟随。当跟随无效时， AXIS_IS_LINKED（运动状态）位设置为 0。 抑止操作（状态 0）：抑止激活跟随。 关联的 CONTROL 位：CONTROL_FOLLOW
%Qr.m.c.28	ALLOW_RESUME	下降沿：向受控轴发送暂停命令。当运动配置文件暂停（零速度）时， AXIS_HOLD 运动状态位设置为 1。 抑止操作（状态 0）：不允许恢复。无论轴何时被激活，都保持暂停状态。 关联的 CONTROL 位：CONTROL_RESUME
%Qr.m.c.29	ALLOW_MOVE	下降沿：发出暂停命令。在暂停过程开始时，设置 AXIS_HALT 运动状态位。 抑止操作 (0)：抑止运动命令。无论轴何时激活，都保持暂停状态。
%Qr.m.c.30	ALLOW_NOT_FASTSTOP	下降沿：向受控轴发出快速停止命令。 AXIS_FASTSTOP 运动状态位设置为 1。 抑止操作（状态 0）：抑止运动，并且在轴激活时，仍然保持快速停止状态。 上升沿：如果提供了轴，则取消快速停止。 AXIS_FASTSTOP 运动状态位设置为 0。
%Qr.m.c.31	ALLOW_NOT_FLT	下降沿：触发用户错误。AXIS_SUMMARY_FLT 运动状态位设置为 1。 抑止操作（状态 0）：强制发出用户错误。

状态更改时可修改的控制位

CONTROL 位

下表列出了 CONTROL 位：

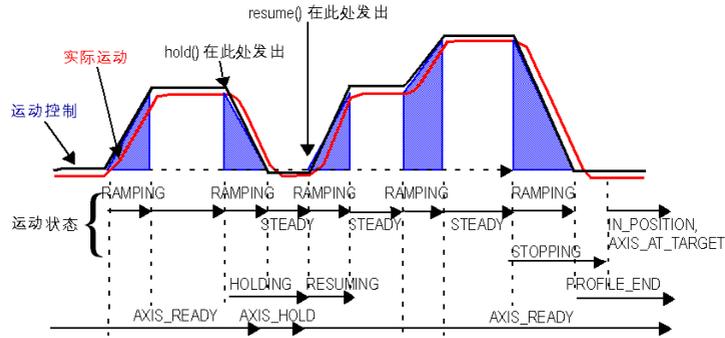
语言对象	名称	描述
%Qr.m.c.2	CONTROL_ACQUIRE	上升沿：获取受控轴的命令，并将其链接到 MovementGroup（运动组）。如果该操作正确执行，则设置 AXIS_IN_COMMAND 位。 关联的 ALLOW 位：ALLOW_ACQUIRE
%Qr.m.c.10	CONTROL_ENABLE	上升沿：激活受控轴。如果驱动器有效，则 DRIVE_ENABLED 位设置为 1。 关联的 ALLOW 位：ALLOW_ENABLE
%Qr.m.c.11	CONTROL_FOLLOW	上升沿：为 SlaveGroup（从轴组）或此类组的成员激活跟随。AXIS_IS_LINKED 位在跟随有效时设置为 1。 关联的 ALLOW 位：ALLOW_FOLLOW
%Qr.m.c.12	CONTROL_RESUME	上升沿：从暂停中恢复 (ALLOW_RESUME)。AXIS_HOLD（运动状态）位在恢复开始时设置为 0。 关联的 ALLOW 位：ALLOW_RESUME
%Qr.m.c.15	CONTROL_CLEAR_FLT	上升沿：清除运动故障 (MovementFault)。如果该操作正确执行，则 AXIS_SUMMARY_FLT 位设置为 0。
%Qr.m.c.4	CONTROL_JOG_POS	上升沿：正向执行连续运动。 下降沿：停止当前运动
%Qr.m.c.5	CONTROL_JOG_NEG	上升沿：反向执行连续运动。 下降沿：停止当前运动
%Qr.m.c.13	CONTROL_INC_POS	上升沿：正向执行增量运动。
%Qr.m.c.14	CONTROL_INC_NEG	上升沿：反向执行增量运动。

状态位和字

简介

状态位 (%I) 指示轴控模块（控制器）的状态以及变速控制器的状态。

下图显示了所示的各种运动条件下的运动概况。



手动模式

简介

此模式允许使用特定命令沿实轴或虚轴执行连续运动或增量运动。

注意：要使用此模式，可：

- 通过具有 TSX CSY 84 模块（SV 版本 ≥ 1.2 ）或 TSX CSY 164 模块（任何版本）的程序
- 通过 Unity Pro 调试屏幕（2.1 版或更高版本）。

手动运动使用缺省情况下配置的速度（VMax 的 1/2）。但是可以：

- 通过调试屏幕或通过程序修改缺省速度配置（请参见（参见第 241 页））
- 通过使用 Set_Speed_Override 功能修改运动速度（请参见（参见第 194 页））
- 通过调试屏幕修改所选位置单位中的增量值

也可以通过调试屏幕或通过程序让轴参考原点或某特定点（请参见（参见第 179 页）、（参见第 182 页）和（参见第 184 页））。

以下命令不可用于手动模式：

- MoveImmed（参见第 189 页）和 MoveQueue（参见第 191 页）
- Pause（%Qr.m.c.28）和 Resume（%Qr.m.c.12）
- Following（%Qr.m.c.11）和 Enable（%Qr.m.c.2）

访问手动模式

可以从调试屏幕或通过程序来访问手动模式（请参见（参见第 178 页））。

注意：缺省情况下，配置后轴处于自动模式。

其他命令

有 4 个命令可用于执行其他运动。

名称	说明
Jog+	用于上升沿, 执行正向连续运动 (1)。 用于下降沿, 停止正在进行的运动。
Jog-	用于上升沿, 执行负向连续运动 (1)。 用于下降沿, 停止正在进行的运动。
Inc+	用于上升沿, 执行参考轴上的正向增量运动 (2)。
Inc-	用于上升沿, 执行参考轴上的负向增量运动 (2)。
图例	
(1)	运动限制值为: <ul style="list-style-type: none"> ● 参考轴的软件限制 ● 非参考轴的轴理论位置的最大值
(2)	以所选单位的千分之几为单位的增量。 所选增量的值必须在轴的运动限制值减去目标位置容差所得值的范围内。

激活条件

仅在以下条件下才能激活各个命令:

- 必须启用伺服驱动器
- 为指定轴激活手动模式
- 轴已停止
- 无法同时发送两个命令
- 如果轴已被参考且不在软件限制范围内, 则必须采用能使轴返回到软件限制范围内的运动方向。

注意: 如果没有满足这些条件中的任何一个, 则系统会显示一则与缺少条件相关的消息。

例外: 如果在相同的循环期间, 对于同一个轴, 模块在执行 Jog 命令时收到下降沿, 而在手动执行另一个命令时收到上升沿, 则该模块仅下降沿考虑在内。

Set_Functional_Mode 功能

描述

Set_Functional_Mode 允许访问手动模式。

注意：此模式可用于实轴和虚轴。

如何使用此功能

此功能是在 WRITE_CMD 指令（请参见 *WRITE_CMD*, 第 388 页）的帮助下实现的。

语言对象

下表显示与 Set_Functional_Mode 功能关联的不同语言对象。

对象	名称	功能代码	描述
%MWr.m.c.26	ACTION_CMD	2572	命令
%MDr.m.c.20	PARAM_CMD_1	-	要写入的值 (1)
%MWr.m.c.19	ERROR_CMD	-	错误报告。请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页。
说明			
(1) 写入值： 写入 0 以更改到手动模式 写入 1 以中止手动模式			

关联的离散量输入

可以通过读取以下离散量输入的值来检查当前处于活动状态的模式：

对象	名称	描述
%Ixy.i.25	AXIS_MANUAL_MODE	0: 自动模式（缺省情况） 1: 手动模式

注意：如果在运动正在执行时更改了模式，则此输入的值在电机停止后更改。

回归功能

说明

此功能允许将参考点请求发送到速度驱动器。

注意：应事先在驱动器上定义参考点的类型。

注：可设定的位置限制在轴的参考点正确完成之后，或者调用了 ForcedHome（强制回归）功能之后才生效。

注：行程终端开关必须连接在参考点之前，因为驱动器可能使用这些输入来确定应何时将其运动反向。

在参考点操作期间，任何在结束之前中断此操作的尝试将导致轴停止，并发出故障信号。因此，在轴可再次运动之前，需要确认此故障。

该功能在实轴上的操作

轴必须启用，并且处于 AXIS_IN_COMMAND 状态。为了使命令可成功执行，轴决不能处于运动、暂停或故障状态，并且必须将 ALLOW_MOVE 运动控制位设置为 1。

如何使用此功能

此功能可：

- 通过位的位置控制功能直接实现（隐式交换）
- 使用 WRITE_CMD（请参见 *WRITE_CMD*, 第 388 页）实现

关联的语言对象

隐式交换：下表显示与此功能关联的不同语言对象。

对象	名称	有效性	说明
%Qr.m.c.29	ALLOW_MOVE	状态 1	允许位
%Qr.m.c.228	ALLOW_RESUME	状态 1	允许位
%Qr.m.c.30	ALLOW_NOT_FASTSTOP	状态 1	允许位
%Ir.m.c.5	AXIS_HOMED	-	状态位

通过 WRITE_CMD 交换：下表显示与此功能关联的不同语言对象。

对象	名称	功能代码	说明
%MWr.m.c.26	ACTION_CMD	6034	命令
%MFr.m.c.31	PARAM_CMD_3	-	参考点过程的方向：1.0（正向）或 -1.0（反向）
%MFr.m.c.33	PARAM_CMD_4	-	以速度单位表示的参考点过程的速度
%MWr.m.c.19	ERROR_CMD	-	错误报告。请参阅 <i>WRITE_CMD</i> 命令：编程错误，第 357 页。

参考点与回归功能

如何使 SERCOS(r) 驱动器回归

回归开关应连接到驱动器输入。当运动方向背离指定源时，源开关的理想位置是与解析器的零脉冲偏离 180 度，或者位于编码器的顶部。如果将此转变置于解析器 null 脉冲或编码器标记脉冲的 +/- 45 度的区域内，则将导致回归故障（除非驱动器的配置软件禁用回归故障）。如果驱动器支持，该开关的放置位置可通过读取回归开关放置错误参数（SERCOS(r) IDN S-0298 或 IDN US-0298）来确定。

回归序列分成两步：

- 查找源开关
- 移至源位置

在其中任一步骤中，向运动控制器发送 Halt、FastStop 或 DisableDrive 功能将导致回归中止并出现 AXIS_SUMMARY_FLT 故障，并且使轴处于未回归状态。如果在第一步中发出 Hold 功能，则回归将中止并出现 AXIS_SUMMARY_FLT 故障，并使轴处于未回归状态。第二步允许 Hold 和 Resume 功能。

注意：在此步骤中，驱动器在以指定的源方向运动时将搜索参考点开关。如果驱动器遇到 EOT 开关，或者遇到回归开关但未处于指定的回归方向上时，它可能反转方向（这取决于驱动器制造商）。（请参见 SERCOS(r) 驱动器手册中的回归描述）。

通常，此操作涉及下列事件：

- 驱动器确定向哪个方向开始运动轴：
 - 如果源开关指示 " 不存在 "(NOT PRESENT)，则轴朝指定的参考点方向移动。
 - 如果检测到源开关，则轴朝指定的参考点方向的反方向运动，以放弃源开关。

ForcedHomed 功能

描述

此功能用于强制确定轴的参考点。此功能用作手动归位过程的一部分。应用程序可将轴移到一个已知位置，发送 SetPosition 功能将某个已知值赋予位置读数，然后发送 ForcedHome 功能。然后，运动控制器使软件位置限制生效，并且将 AXIS_HOMED 位设置为 1。

如何使用此功能

此功能是在 WRITE_CMD 指令（请参见 *WRITE_CMD*, 第 388 页）的帮助下实现的。

关联的语言对象

通过 WRITE_CMD 进行交换：下表显示与此功能关联的不同语言对象。

对象	名称	功能代码	描述
%MWr.m.c.26	ACTION_CMD	6039	命令
%MWr.m.c.19	ERROR_CMD	-	错误报告。请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页。

Unhome 功能

说明

此功能取消轴的参考点。

如果轴未被引用，则运动控制器的软件位置限制无效。执行 Unhome 功能将导致软件位置限制被忽略。

小心

意外的轴运动

当轴未被引用时，请使用与驱动器直接相连的行程终端 (EOT) 开关。

如果不遵守这些说明，则会导致设备损坏。

如何使用此功能

此功能是在指令 WRITE_CMD (参见第 388 页) 的帮助下实现的。

关联的语言对象

隐式交换：下表显示与此功能关联的不同语言对象。

对象	名称	有效性	说明
%Ir.m.c.6	AXIS_HOMED	-	状态位

通过 WRITE_CMD 进行交换：下表显示与此功能关联的不同语言对象。

对象	名称	功能代码	说明
%MWr.m.c.26	ACTION_CMD	6038	命令
%MWr.m.c.19	ERROR_CMD	-	错误报告。请参阅 <i>WRITE_CMD</i> 命令：编程错误，第 357 页。

SetPosition 功能

说明

SetPosition 功能将轴的位置读数更改为给定值。仅当轴未运动时才能更改位置。位置读数在轴控模块中进行修改。它不更改驱动器本身的位置值。这意味着，如果轴试运行软件工具通过与驱动器直接连接来监控该驱动器，则位置显示不会显示更改后的位置值。但是，由轴控模块监控或控制的位置将反映新参数。

Home 功能清除由 SetPosition 功能设置的所有先前位置设置。



小心

应用程序的意外行为 - BIAS_REMAIN 模式

只要 %Ir.m.c.24 位的值为 1，就不要发送 SET_POSITION 命令。

如果不遵守这些说明，将会导致受伤或设备损坏。

注意：在使用 BIAS_REMAIN 序列的情况下，对于从轴上结果偏移的识别将通过 %Ir.m.c.24 位反映出来。

如何使用此功能

此功能是在 WRITE_CMD (参见第 388 页) 指令的帮助下实现的。

关联的语言对象

通过 WRITE_CMD 进行交换：下表显示与此功能关联的不同语言对象。

对象	名称	功能代码	说明
%MWr.m.c.26	ACTION_CMD	2053	命令
%MFr.m.c.31	PARAM_CMD_3	-	要写入的值。
%MWr.m.c.19	ERROR_CMD	-	错误报告。请参阅 <i>WRITE_CMD</i> 命令：编程错误，第 357 页。

12.3 运动功能

本节主题

本节描述运动功能。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
关于运动功能	186
MoveType 数据类型	188
MoveImmed 功能	189
MoveQueue 功能	191
GetMoveQueueLength 功能	193
GetSpeedOverride 和 SetSpeedOverride 功能	194
EnableRealTimeCtrlBit 功能	195

关于运动功能

MOVE 功能的结构

运动功能的基本结构如下：

< 运动功能 >(< 运动类型 >, < 目标位置 >, < 到达目标的速度 >)

其中：

- 运动功能为下列功能之一：
 - MoveImmed**：中止任何正在进行的运动配置文件，并移到新的目标位置。
 - MoveQueue**：将某个运动配置文件结合到正在进行的配置文件的结尾，以到达新的目标位置。
- 运动类型属于下列类型之一：
 - ABS_MOVE**：绝对运动（需要已对轴设定参考）。目标位置被视为以源位置为参考点的绝对位置值。运动将采用最短路线（在激活了模数功能时要小心）。
 - ABS_MOVE_POS**：激活模数功能的轴或协同轴的正向绝对运动。
 - ABS_MOVE_NEG**：激活模数功能的轴或协同轴的负向绝对运动。
 - ABS_MOVE_NO_ROLLOVER**：配置有模数的独立轴的绝对运动。可达到超过所配置模数的值。
 - INCR_MOVE**：增量运动。目标位置被视为以当前位置（对于 MoveImmed）或上次结合的目标位置 (MoveQueue) 为参考点的增量值。
 - CONT_MOVE**：连续运动。目标位置指示运动到软件限制的连续运动配置文件的
- 目标位置是以轴的缺省位置单位指定目标位置的实数。
- 到达目标的速度是以轴的缺省速度单位指定接近速度的实数。

功能

MOVE 功能开始运动。运动功能将在运动配置文件完成之前返回到应用程序。应用程序使用轴状态位 (%I) 以确定运动配置文件的状态。（请参见 *SERCOS 语言对象*，第 373 页。）

如果 ABS_MOVE 功能指定的目标位置超过了运动轴的最大或最小位置，将生成 MovementError，因为运动轴无法到达目标。如果 INCR_MOVE 运动功能生成的目标超出限制，则不会生成 MovementError，因为目标点的计算可能由于运动排队机制而延迟。此类 INCR_MOVE 运动被转换为到达限制的目标位置，并在到达限制时生成 AXIS_SUMMARY_FAULT。大于运动轴最大速度限制的接近速度被转换为速度限制处的接近速度。

运动状态

IN_POSITION 和 AXIS_AT_TARGET 位在运动开始时设置为 0。在指示运动轴加速或减速时设置 RAMPING 位。在指示运动轴以匀速运动时，将 STEADY 位设置为 1。在指示运动轴减速到达目标位置时，将 STOPPING 设置为 1。在 STOPPING 已设置为 1 后，当实际位置和目标位置在正确位置带内时，IN_POSITION 设置为 1。当控制器命令伺服元件保留在目标位置时，PROFILE_END 设置为 1。在 PROFILE_END 已设置为 1 后，当实际位置和目标位置在正确位置带内时，将设置 AXIS_AT_TARGET。当轴已到达 PROFILE_END 并且为 IN_POSITION 时，STOPPING 将设置为 0。

MoveType 数据类型

说明

轴向某个位置的运动（运动）可以是递增的、连续的或绝对的：

- 递增的：运动相对于当前位置进行
- 连续的：轴向用作运动目标位置的最大或最小位置运动
- 绝对的：轴向某个绝对位置运动

数据类型的值

下表列出了各种数据类型值：

运动类型	值
ABS_MOVE	0
INCR_MOVE	1
CONT_MOVE	2
ABS_MOVE_POS	3
ABS_MOVE_NEG	4
ABS_MOVE_UNLINK *	10
CONT_MOVE_UNLINK *	12
ABS_MOVE_POS_UNLINK *	13
ABS_MOVE_NEG_UNLINK *	14
ABS_MOVE_NO_ROLLOVER *	20
ABS_MOVE_NO_ACC_CORRECTION **	30
INCR_MOVE_NO_ACC_CORRECTION **	31

* 本章未讨论的功能；请参见下一章。

** 本章未讨论的功能；请参见下一章（参见第 223 页）。

MovImmmed 功能

说明

此功能导致指定轴以指定速度向给定位置运动。该运动立即应用，并忽略当前正在进行的任何其他运动。正在进行的运动和任何排队列入的运动都被取消。

立即运动可以是：

- 递增的：运动相对于当前位置进行。
- 连续的：轴向最大或最小位置运动。最小或最大位置用作运动功能的目标位置。
- 绝对的：绝对运动导致轴运动到某个绝对位置。

如何使用此功能

此功能是在 `WRITE_CMD` 指令（请参见 *WRITE_CMD*, 第 388 页）的帮助下实现的。

关联的语言对象

通过 `WRITE_CMD` 进行交换：下表显示与此功能关联的不同语言对象。

对象	名称	功能代码	说明
%MWr.m.c.26	ACTION_CMD	513	命令
%MDr.m.c.27	PARAM_CMD_1	-	运动类型 (1)
%MDr.m.c.29	PARAM_CMD_2	-	应用于已配置的加速度或减速度的系数，显示为千分数 ($1 \leq x \leq 10000$)。请参见操作码 2172、实轴 (参见第 438 页) 或虚轴 (参见第 442 页)。
%MFr.m.c.A (3)	PARAM_CMD_B (3)	-	轴 N (2) 的目标位置
%MFr.m.c.C (4)	PARAM_CMD_D (4)	-	到达 N 轴 (5) 的目标的接近速度
%MWr.m.c.19	ERROR_CMD	-	错误报告。请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页。
说明			
(1): INCR_MOVE、ABS_MOVE、ABS_MOVE_POS、ABS_MOVE_NEG、ABS_MOVE_NO_ROLLOVER 或 CONT_MOVE			
(2): N= 1 表示独立轴，N 包括 2 和 8 之间的值表示协同轴组。			
(3): A = [31+2(N-1)], B = [3+(N-1)]			
(4): C = [31+2(N+N'-1)], D = [3+(N+N'-1)]			
(5): N = 轴在组中的位置			

协同轴组的参数示例

对于具有 2 个成员轴的运动:

- PARAM_CMD_1 (%MDr.m.c.27) = 运动类型
- PARAM_CMD_3 (%MFr.m.c.31) = 轴 1 的位置
- PARAM_CMD_4 (%MFr.m.c.33) = 轴 2 的位置
- PARAM_CMD_5 (%MFr.m.c.35) = 轴 1 的接近速度
- PARAM_CMD_6 (%MFr.m.c.37) = 轴 2 的接近速度

MoveQueue 功能

说明

此功能导致指定轴以指定速度向给定位置运动。

此功能附加到任何当前正在执行或等待执行的运动。MoveQueue 混合到当前轨道而不在先前执行的运动控制结尾停止。最多可以对 32 个运动进行排队。

队列中的所有条目在执行 MoveImmed、Halt 或 FastStop 功能或者出现 AXIS_SUMMARY_FAULT 或 DRIVE_FAULT 时被删除。

MoveQueue 功能可以是：

- **递增**：运动相对于上次排入队列的目标位置进行
- **绝对**：绝对运动导致轴运动到某个绝对位置。
- **连续**：轴向最大或最小位置运动。最小或最大位置用作运动功能的目标位置。



应用程序的意外行为 - MOVEQUEUE 功能行为

从不会执行在连续运动后发出的 MoveQueue。连续运动将在达到位置限制时发生故障，并删除排入队列的运动。

如果不遵守这些说明，将会导致死亡、严重伤害或设备损坏。

关联的语言对象

通过 `WRITE_CMD` 进行交换：下表显示与此功能关联的不同语言对象。

对象	名称	功能代码	说明
%MWr.m.c.26	ACTION_CMD	520	命令
%MDr.m.c.27	PARAM_CMD_1	-	运动类型 (1)
%MDr.m.c.29	PARAM_CMD_2	-	应用于已配置的加速度或减速度的系数，显示为千分数 ($1 \leq x \leq 10000$)。请参见操作码 2172、实轴 (参见第 438 页) 或虚轴 (参见第 442 页)。
%MFr.m.c.A (3)	PARAM_CMD_B (3)	-	轴 N (2) 的目标位置
%MFr.m.c.C (4)	PARAM_CMD_D (4)	-	到达 N' 轴 (5) 的目标的接近速度
%MWr.m.c.19	ERROR_CMD	-	错误报告。请参阅 <code>WRITE_CMD</code> 命令：编程错误，第 357 页。
图例			
(1): INCR_MOVE、ABS_MOVE、ABS_MOVE_POS、ABS_MOVE_NEG、ABS_MOVE_NO_ROLLOVER 或 CONT_MOVE			
(2): N = 1 表示独立轴，N 包括 2 和 8 之间的值表示协同轴组。			
(3): $A = [31+2(N-1)]$ ， $B = [3+(N-1)]$			
(4): $C = [31+2(N+N'-1)]$ ， $D = [3+(N+N'-1)]$			
(5): N' = 轴在组中的位置			

协同轴组的参数示例

对于具有 2 个成员轴的运动：

- PARAM_CMD_1 (%MDr.m.c.27) = 运动类型
- PARAM_CMD_3 (%MFr.m.c.31) = 轴 1 的位置
- PARAM_CMD_4 (%MFr.m.c.33) = 轴 2 的位置
- PARAM_CMD_5 (%MFr.m.c.35) = 轴 1 的接近速度
- PARAM_CMD_6 (%MFr.m.c.37) = 轴 2 的接近速度

GetMoveQueueLength 功能

描述

GetMoveQueueLength 功能返回指定轴在队列中的条目数。

当队列中的条目数大于 1 时，轴将无波动地沿连续路径运动。条目是使用 MoveQueue 功能放在队列中的。条目在规划轴的运动（当前运动和下一个运动）时从队列中删除。

如果在 MoveQueue 功能将第一个条目放入队列后立即执行 GetMoveQueueLength，则返回值将为 0。这是因为该条目一放入队列中即对轴的运动进行规划。在轨道发生器删除该条目后，才能执行 GetMoveQueueLength 功能。最多可以管理 34 个运动：队列中的 32 个运动、轨道生成器中存在的当前和下一个运动。

注意：GetMoveQueue 仅返回正在等待的条目数量，而不考虑当前或下一个运动。

如何使用此功能

此功能是在 WRITE_CMD 指令（请参见 *WRITE_CMD*, 第 388 页）的帮助下实现的。

关联的语言对象

通过 WRITE_CMD 进行交换：下表显示与此功能关联的不同语言对象。

对象	名称	功能代码	描述
%MWr.m.c.26	ACTION_CMD	9510	命令
%MDr.m.c.20	RETURN_CMD_1	-	命令的结果
%MWr.m.c.19	ERROR_CMD	-	错误报告。请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页。

GetSpeedOverride 和 SetSpeedOverride 功能

描述

GetSpeedOverride 功能向轴发送相应的速度调制系数。

SetSpeedOverride 功能设置轴的相关速度调制系数。

该值被显式指定为百分比。

系数介于 0 和 100 之间。

调制系数为 0% 使轴不能移动。

调制系数为 100% 使轴以命令的速度运动。

注意：系数影响加速度和减速度值。

注意：无论速度调制系数值如何，轴都跟随同一轨道。

如何使用这些功能

这些功能是在 WRITE_CMD 指令（请参见 *WRITE_CMD*, 第 388 页）的帮助下实现的。

关联的语言对象

通过 WRITE_CMD 进行交换：下表显示与这些功能关联的不同语言对象。

对象	名称	功能代码	描述
%MWr.m.c.26	ACTION_CMD	1513	读取命令
		2513	写入命令
%MFr.m.c.22	RETURN_CMD_2	-	读取命令的结果
%MFr.m.c.31	PARAM_CMD_3	-	要写入的值
%MWr.m.c.19	ERROR_CMD	-	错误报告。请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页。

EnableRealTimeCtrlBit 功能

描述

此功能用于启用或禁用实时控制位 1 和 2 (分别为 %Qr.m.c.6 和 %Qr.m.c.7)。启用这些控制位会导致通过 SERCOS 网络将它们发送到驱动器。

注意：缺省情况下，这些位是停用的。

有关与这些控制位关联的命令的详细信息，请参考 IDN 文档。

如何使用此功能

此功能是在指令 WRITE_CMD (参见第 388 页) 的帮助下实现的。

关联的语言对象

通过 WRITE_CMD 进行交换：下表显示与此功能关联的不同语言对象。

对象	名称	功能代码	描述
%MWr.m.c.26	ACTION_CMD	2564	命令
%MDr.m.c.27	PARAM_CMD_1	-	1 = 启用 0 = 禁用
%MWr.m.c.19	ERROR_CMD	-	错误报告。请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页。

12.4 序列停止后的运动功能

本节的目标

本节描述使运动能够在序列禁用时排序的功能。这些命令基于 "MoveImmed" 和 "MoveQueue"。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
功能概述	197
"MoveImmed" 功能	198
带有 ABS_MOVE_NO_ROLLOVER 的 "MoveImmed" 功能	200
"MoveQueue" 功能	203
带有 ABS_MOVE_NO_ROLLOVER 的 "MoveQueue" 功能	205

功能概述

目的

目标是在序列禁用时对运动排序。

这些功能使用“MoveImmed (参见第 198 页)”和“MoveQueue (参见第 203 页)”命令对两种不同类型的运动执行。

运动类型

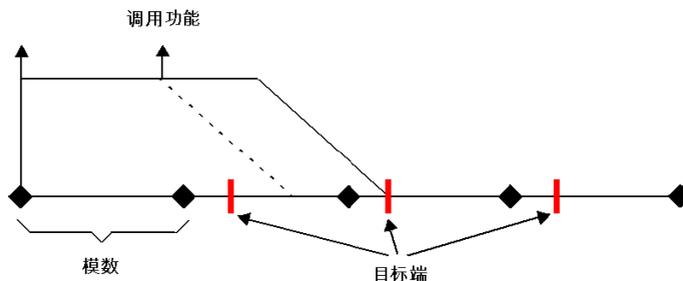
下表列出了各种数据类型值：

运动类型	值	说明
ABS_MOVE_UNLINK	10	绝对运动（如果轴配置了模数，则尽可能接近）
CONT_MOVE_UNLINK	12	连续运动
ABS_MOVE_POS_UNLINK	13	绝对运动，正向实施（带活动模数的轴）
ABS_MOVE_NEG_UNLINK	14	绝对运动，负向实施（带活动模数的轴）
ABS_MOVE_NO_ROLLOVER	20	配置有模数的独立轴的绝对运动。可达到超过所配置模数的值。

"MoveImmed" 功能

目的

此命令用于对从轴组的成员轴执行序列停用，然后用于执行运动（无颠簸或暂停）。



特性

此命令基于当前 "MoveImmed" (参见第 189 页) 命令（操作码 513）。

此命令涉及**实轴**或**虚轴**，这些轴成为在**比率**、**设定点跟随**和 **BIAS_REMAIN** 中配置的**从轴组**的一部分。这些轴必须是**已链接的**。

初始条件:

- 主轴是实轴或虚轴，
- 主轴的速度在序列停用与 4 个 SERCOS 循环后的这段时间之间趋于常量，

当这一类型的运动正在进行时，"**RESUMING**" 状态位设置为 1。该位由系统在运动完成后设置为 0。

对象	类型	符号	说明	注释
%MWr.m.c.19	字	ERROR_CMD		错误代码 *
%MDr.m.c.20	双字	RETURN_CMD_1		未使用
%MFr.m.c.22	浮点数	RETURN_CMD_2		未使用
%MFr.m.c.24	浮点数	RETURN_CMD_3		未使用
%MWr.m.c.26	字	ACTION_CMD	513	MoveImmed
%MDr.m.c.27	双字	PARAM_CMD_1	10 或 12 或 13 或 14	类型

对象	类型	符号	说明	注释
%MDr.m.c.29	双字	PARAM_CMD_2	加速 / 减速系数	应用于已配置的加速度或减速度的系数，显示为千分数 ($1 \leq x \leq 10000$)。请参见操作码 2172、实轴 (参见第 439 页) 或虚轴 (参见第 442 页)。
%MFr.m.c.31	浮点数	PARAM_CMD_3	目标位置	
%MFr.m.c.33	浮点数	PARAM_CMD_4	目标速度	

*** 命令被拒绝:**

- "MoveImmed" 命令被拒绝的所有情况，
- 轴必须是已链接的 (CMD_NOT_ALLOWED: 命令被拒绝 = 50)，
- 当序列停用已经在进行时，不允许执行任何运动排序功能 (CMD_NOT_ALLOWED: 命令被拒绝 = 50)。

位 7 MW_MERGE_UNLINK (参见第 344 页) 的命令中止的情况:

某些条件至关重要，必须满足它们才能执行命令。

如果这些条件之一不满足，则会终止命令:

- 在比率、设定点跟随和 BIAS_REMAIN 模式中进行从轴设置，
- 轴的状态必须为: AXIS_READY = 1，
- 组的主轴不得为远程轴。
- 轴上不得存在任何附加运动，并且在运动进行中不能出现暂停 (AXIS_HOLD = 0 和 HOLDING = 0)。

注:

在从站带有活动模数并实施方向运动类型 (13 或 14) 的情况下:

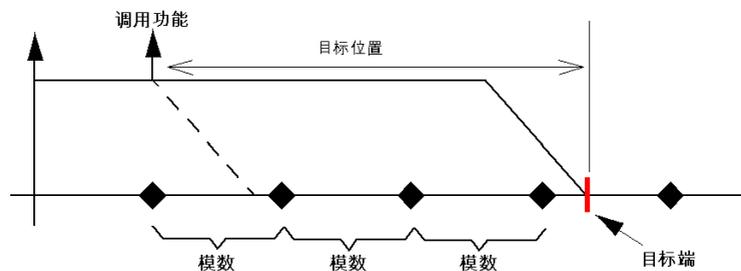
- 如果轴已停止，则运动将 "最多" 为一转，
- 如果轴先前已在运动中:
 - 允许行程距离和减速斜坡，轴将能在停止前切换到请求的目标速度，
 - 否则，模块将计算要执行的最少转数，以便在请求的绝对位置停止 (考虑减速斜坡)。

带有 ABS_MOVE_NO_ROLLOVER 的“MoveImmed”功能

目的

此命令用于对从轴组的成员轴执行序列停用，然后用于执行运动（无颠簸或暂停）。

它允许对独立轴配置模数以到达超越模数正向限制的绝对位置。



示例：配置有 360 度模数的轴的情况

- 目标位置：1200 度

结果：运动结束时的最终位置将是：

（目标位置 - 模数个数）

$$1200 - 3 \times 360 = 120 \text{ 度}$$

特性

此命令基于当前“MoveImmed” (参见第 189 页) 命令 (操作码 513)。

此命令涉及**实轴**或**虚轴**，这些轴成为在**比率、设定点跟随**和 **BIAS_REMAIN** 中配置的**从轴组**的一部分。这些轴必须是**已链接的**。

初始条件:

- 主轴是实轴或虚轴，
- 主轴的速度在序列停用与 4 个 SERCOS 循环后的这段时间之间趋于常量，

当这一类型的运动正在进行时，“**RESUMING**”状态位设置为 1。该位由系统在运动完成后设置为 0。

对象	类型	符号	值	注释
%MW.r.m.c.19	字	ERROR_CMD		错误代码 *
%MDr.m.c.20	双字	RETURN_CMD_1		未使用
%MFr.m.c.22	浮点数	RETURN_CMD_2		未使用
%MFr.m.c.24	浮点数	RETURN_CMD_3		未使用
%MW.r.m.c.26	字	ACTION_CMD	513	MoveImmed
%MDr.m.c.27	双字	PARAM_CMD_1	20	类型
%MDr.m.c.29	双字	PARAM_CMD_2	加速 / 减速系数	应用于已配置的加速度和减速度的系数，显示为千分数 ($1 \leq x \leq 10000$)。请参见操作码 2172、实轴 (参见第 439 页) 或虚轴 (参见第 442 页)。
%MFr.m.c.31	浮点数	PARAM_CMD_3	目标位置	
%MFr.m.c.33	浮点数	PARAM_CMD_4	目标速度	

* 命令被拒绝:

- “MoveImmed” 命令被拒绝的所有情况，
- 轴必须是已链接的 (CMD_NOT_ALLOWED: 命令被拒绝 = 50)，
- 当序列停用已经在进行时，不允许执行任何运动排序功能 (CMD_NOT_ALLOWED: 命令被拒绝 = 50)。

位 7 MW_MERGE_UNLINK (参见第 344 页) 的命令中止情况:

某些条件至关重要，必须满足它们才能执行命令。

如果这些条件之一不满足，则会终止命令：

- 在比率、设定点跟随和 BIAS_REMAIN 模式中进行从轴设置，
- 轴的状态必须为：AXIS_READY = 1，
- 组的主轴不得为远程轴。
- 轴上不得存在任何附加运动，并且在运动进行中不能出现暂停（AXIS_HOLD = 0 和 HOLDING = 0）。

注：

在从站带有活动模数并实施方向运动类型 (20) 的情况下：

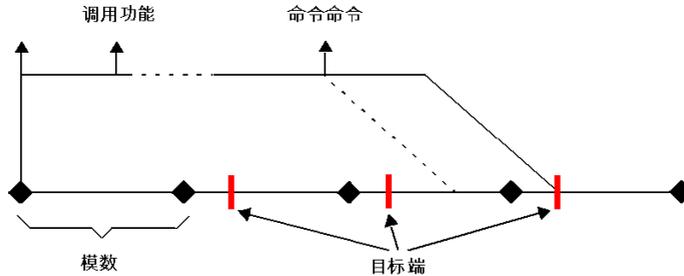
- 如果轴已停止，则运动将“最多”为一转，
- 如果轴先前已在运动中：
 - 允许行程距离和减速斜坡，轴将能在停止前切换到请求的目标速度，
 - 否则，模块将计算要执行的最少转数，以便在请求的绝对位置停止（考虑减速斜坡）。

"MoveQueue" 功能

目的

此命令用于准备运动，该运动将在从轴组的成员轴的序列被停用时执行（无颠簸或暂停）。

此命令不停用轴的序列。



特性

此命令基于当前 "MoveQueue" (参见第 191 页) 命令 (操作码 520)。

此命令涉及**实轴**或**虚轴**，这些轴成为在**比率**、**设定点跟随**和**BIAS_REMAIN**中配置的**从轴组**的一部分。这些轴必须是**已链接的**。

初始条件:

- 主轴是实轴或虚轴，
- 主轴的速度在序列停用与 4 个 SERCOS 循环后的这段时间之间趋于常量，

序列停用后 ("MoveQueue" 挂起) 在从轴上执行附加运动的事实由状态位 **RESUMING** 发送到该轴。此位在执行运动后由系统设置为 0。

对象	类型	符号	说明	注释
%MWr.m.c.19	字	ERROR_CMD		错误代码
%MDr.m.c.20	双字	RETURN_CMD_1		未使用
%MFr.m.c.22	浮点数	RETURN_CMD_2		未使用
%MFr.m.c.24	浮点数	RETURN_CMD_3		未使用
%MWr.m.c.26	字	ACTION_CMD	520	MoveQueue
%MDr.m.c.27	双字	PARAM_CMD_1	类型	10 或 12 或 13 或 14

对象	类型	符号	说明	注释
%MDr.m.c.29	双字	PARAM_CMD_2	加速 / 减速系数	应用于已配置的加速度或减速度的系数，显示为千分数 ($1 \leq x \leq 10000$)。请参见操作码 2172、实轴 (参见第 438 页) 或虚轴 (参见第 442 页)。
%MFr.m.c.31	浮点数	PARAM_CMD_3	目标位置	
%MFr.m.c.33	浮点数	PARAM_CMD_4	目标速度	

命令被拒绝:

- "MoveQueue" 命令被拒绝 (参见第 357 页) 的所有现有情况
- 轴必须是已链接的 (CMD_NOT_ALLOWED: 命令被拒绝 = 50),
- 当序列停用已经在进行时, 不允许执行任何运动排序功能 (CMD_NOT_ALLOWED: 命令被拒绝 = 50)。

位 7 MW_MERGE_UNLINK (参见第 344 页) 的命令中止的情况:

命令传输和 Unlink 激活之间的某些条件至关重要, 必须满足这些条件才能执行命令。

如果缺少其中一个条件, 则会终止要排序的运动:

- 在比率、设定点跟随和 BIAS_REMAIN 模式中进行从轴设置,
- 轴的状态必须为: AXIS_READY = 1 (参见第 168 页),
- 组的主轴不得为远程轴。
- 在序列禁用时, 轴上一定不能有任何附加运动, 并且进行中不能有暂停 (AXIS_HOLD = 0 和 HOLDING = 0)

注:

在从站带有活动模数并实施方向运动类型 (13 或 14) 的情况下:

- 如果轴已停止, 则运动将 "最多" 为一转,
- 如果轴先前已在运动中:
 - 允许行程距离和减速斜坡, 轴将能在停止前切换到请求的目标速度,
 - 否则, 模块将计算要执行的最少转数, 以便在请求的绝对位置停止 (考虑减速斜坡)。

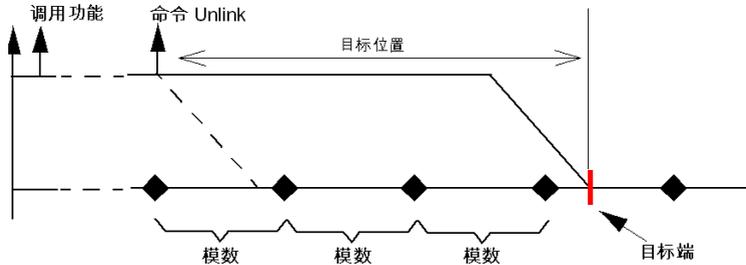
带有 ABS_MOVE_NO_ROLLOVER 的“MoveQueue”功能

目的

此命令用于准备运动，该运动将在从轴组的成员轴的序列被停用时执行（无颠簸或暂停）。

此命令不停用轴的序列。

它允许对独立轴配置模数以到达超越模数正向限制的绝对位置。



示例：配置有 360 度模数的轴的情况

- 目标位置：1200 度

结果：运动结束时的最终位置将是：

（目标位置 - 模数个数）

$$1200 - 3 \times 360 = 120 \text{ 度}$$

特性

此命令基于当前“MoveQueue”（参见第 191 页）命令（操作码 520）。

此命令涉及**实轴**或**虚轴**，这些轴成为在比率、设定点跟随和 BIAS_REMAIN 中配置的**从轴组**的一部分。这些轴必须是**已链接的**。

初始条件：

- 主轴是实轴或虚轴，
- 主轴的速度在序列停用与 4 个 SERCOS 循环后的这段时间之间趋于常量，

序列停用后（“MoveQueue”挂起）在从轴上执行附加运动的事实由状态位 **RESUMING** 发送到该轴。此位在执行运动后由系统设置为 0。

对象	类型	符号	值	注释
%MW.r.m.c.19	字	ERROR_CMD		错误代码
%MDr.m.c.20	双字	RETURN_CMD_1		未使用
%MFr.m.c.22	浮点数	RETURN_CMD_2		未使用

对象	类型	符号	值	注释
%MFr.m.c.24	浮点数	RETURN_CMD_3		未使用
%MW.r.m.c.26	字	ACTION_CMD	520	MoveQueue
%MDr.m.c.27	双字	PARAM_CMD_1	20	类型
%MDr.m.c.29	双字	PARAM_CMD_2	加速 / 减速系数	应用于已配置的加速度或减速度的系数，显示为千分数 ($1 \leq x \leq 0000$)。请参见操作码 2172、实轴 (参见第 438 页) 或虚轴 (参见第 442 页)。
%MFr.m.c.31	浮点数	PARAM_CMD_3	目标位置	
%MFr.m.c.33	浮点数	PARAM_CMD_4	目标速度	

命令被拒绝：

- “MoveQueue” 命令被拒绝 (参见第 357 页) 的所有现有情况
- 轴必须是已链接的 (CMD_NOT_ALLOWED: 命令被拒绝 = 50)，
- 当序列停用已经在进行时，不允许执行任何运动排序功能 (CMD_NOT_ALLOWED: 命令被拒绝 = 50)。

位 7 MW_MERGE_UNLINK (参见第 344 页) 的命令中止情况：

命令传输和 Unlink 激活之间的某些条件至关重要，必须满足这些条件才能执行命令。

如果缺少其中一个条件，则会终止要排序的运动：

- 在比率、设定点跟随和 BIAS_REMAIN 模式中进行从轴设置，
- 轴的状态必须为：AXIS_READY = 1 (参见第 168 页)，
- 组的主轴不得为远程轴。
- 在序列禁用时，轴上一定不能有任何附加运动，并且进行中不能有暂停 (AXIS_HOLD = 0 和 HOLDING = 0)

注：

在从站带有活动模数并实施方向运动类型 (20) 的情况下：

- 如果轴已停止，则运动将“最多”为一转，
- 如果轴先前已在运动中：
 - 允许行程距离和减速斜坡，轴将能在停止前切换到请求的目标速度，
 - 否则，模块将计算要执行的最少转数，以便在请求的绝对位置停止 (考虑减速斜坡)。

12.5 实际速度 / 位置功能

本节主题

本小节描述实际速度和位置的功能。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
GetActualSpeed 功能	208
GetCommandSpeed 功能	209
模数计数器服务	210
GetUnrolledPosition 功能	211
GetUnrolledCommandedPosition 功能	212
GetCommandedPosition 功能	213

GetActualSpeed 功能

描述

这是一个运动参数设置功能。GetActualSpeed 功能返回指定轴的当前运动速度值。

通过 READ_PARAM 的命令

无关联的内部字。

通过 WRITE_CMD 的命令

ACTION_CMD (%MWr.m.c.26) = 5065

RETURN_CMD_1 (%MDr.m.c.20) = 0

RETURN_CMD_2 (%MFr.m.c.22) = 速度

RETURN_CMD_3 (%MFr.m.c.24) = 0

ERROR_CMD (%MWr.m.c.19):

代码	名称	描述
2014	PROXY_NOT_CONNECTED	运动轴从未成功配置。

GetCommandSpeed 功能

描述

GetCommandSpeed 功能返回指定轴的当前速度值。

如何使用此功能

此功能是在 `WRITE_CMD` 指令（请参见 *WRITE_CMD*, 第 388 页）的帮助下实现的。

关联的语言对象

下表显示与此功能关联的不同语言对象。

对象	名称	功能代码	描述
%MWr.m.c.26	ACTION_CMD	5066	命令
%MFr.m.c.22	RETURN_CMD_2	-	命令的结果
%MWr.m.c.19	ERROR_CMD	-	错误报告。请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页。

模数计数器服务

目的

对模数数量计数。

概览

此命令涉及：

- 使用模数配置的**实轴**，
- 使用模数配置的**虚轴**。

对象	类型	符号	说明	注释
%MWr.m.c.19	字	ERROR_CMD		
%MDr.m.c.20	双字	RETURN_CMD_1	NB_MODULO	模数数量（正向或负向）
%MFr.m.c.22	浮点数	RETURN_CMD_2		
%MFr.m.c.24	浮点数	RETURN_CMD_3		
%MWr.m.c.26	字	ACTION_CMD	573	ReadNBModulo
%MDr.m.c.27	双字	PARAM_CMD_1	0	
%MDr.m.c.29	双字	PARAM_CMD_2	0	
%MFr.m.c.31	浮点数	PARAM_CMD_3	0	
%MFr.m.c.33	浮点数	PARAM_CMD_4	0	

在 TSX CSY 内部，当前位置（NB_MODULO 的值）在以下情况下复位为 0：

- 回归 6034 (参见第 179 页)
- 旋转中的位置强制 2053 (参见第 184 页)

GetUnrolledPosition 功能

描述

GetUnrolledPosition 功能返回无模数的轴位置。

注意：轴不需要轴模数即可支持使用此功能。

如何使用此功能

此功能是在 WRITE_CMD 指令（请参见 *WRITE_CMD*, 第 388 页）的帮助下实现的。

关联的语言对象

通过 WRITE_CMD 进行交换：下表显示与此功能关联的不同语言对象。

对象	名称	功能代码	描述
%MWr.m.c.26	ACTION_CMD	546	命令
%MFr.m.c.22	RETURN_CMD_2	-	命令的结果
%MWr.m.c.19	ERROR_CMD	-	错误报告。请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页。

GetUnrolledCommandedPosition 功能

描述

GetUnrolledCommandedPosition 功能返回当前正被发送到驱动器的即时揭示的位置设置点（无模数）。

当轴运动时，即时位置命令连续更改。仅当轴已停止运动并且 `AXIS_AT_TARGET` 位设置为 1 时，它才等于运动配置文件的目标位置。

注意： 无需激活轴模数即可使用此功能。

如何使用此功能

此功能是在 `WRITE_CMD` 指令（请参见 *WRITE_CMD*, 第 388 页）的帮助下实现的。

关联的语言对象

通过 `WRITE_CMD` 进行交换：下表显示与此功能关联的不同语言对象。

对象	名称	功能代码	描述
%MWr.m.c.26	ACTION_CMD	547	命令
%MFr.m.c.22	RETURN_CMD_2	-	命令的结果
%MWr.m.c.19	ERROR_CMD	-	错误报告。请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页。

GetCommandedPosition 功能

描述

GetCommandedPosition 功能返回当前正在被发送到驱动器的即时位置设置点。当轴运动时，即时位置命令连续更改。仅当轴已停止运动并且 AXIS_AT_TARGET 位设置为 1 时，它才等于运动配置文件的目标位置。

如何使用此功能

此功能是在 WRITE_CMD 指令（请参见 *WRITE_CMD*, 第 388 页）的帮助下实现的。

关联的语言对象

通过 WRITE_CMD 进行交换：下表显示与此功能关联的不同语言对象。

对象	名称	功能代码	描述
%MWr.m.c.26	ACTION_CMD	1053	命令
%MFr.m.c.22	RETURN_CMD_2	-	命令的结果
%MWr.m.c.19	ERROR_CMD	-	错误报告。请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页。

12.6 跟随功能

本节主题

本节描述以下功能。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
关于跟随功能	215
"目标序列激活"功能	217
"序列激活前的剩余时间估计"功能	219
触发器引起的从轴组序列停止 (Unlink)	221
协同轴组的运动功能与 MMS Quantum 功能兼容	223
FollowOn 415 功能描述	225

关于跟随功能

激活跟随功能

可以通过调用 Link (WRITE_CMD) 或 FollowOn (隐式交换) 功能激活跟随功能。可以通过调用 UnLink (WRITE_CMD) 或 FollowOff (隐式交换) 功能禁用跟随功能。

跟随不会一激活就启动：发出激活跟随的命令将启动一个事件序列，在所有事件完成前不会考虑跟随。对于从轴，`AXIS_IS_LINKED` 位指示跟随的实际启动时间。

一旦跟随实际开始，从轴将跟随主轴。如果主轴受多轴运动控制器的控制，应用程序将命令主轴运动，以实现所需的跟随运动。

活动跟随的运动

激活跟随时，主轴可能正在执行某个运动。

- 在比率模式中，从轴将加速到所需的跟随速度。
- 在凸轮模式中，从轴将快速加速以实现由凸轮配置文件指定的所需速度。

激活跟随时，从轴可能正在执行某个运动命令，在下列条件下：

- 如果 `FOLL_ON_HALT SlaveMode` 位已设置，则会在触发跟随时向正在进行的运动命令发出一条暂停命令。这将取消该运动，结果从轴运动将仅基于跟随主轴。
- 如果 `FOLL_ON_HALT SlaveMode` 位未设置为 1，则不会停止正在进行的运动命令。结果从轴运动将是跟随主轴的运动加上正在进行的运动命令的运动之和。

停止跟随的运动

停止跟随时，主轴可能正在执行某个运动。

- 在比率模式下，从轴使用所配置的减速度减速到 0。这将基于跟随主轴停止从轴的运动。
- 在凸轮模式下，从轴会立即停止跟随主轴，而没有任何减速度。

停止跟随时，从轴可能正在执行某个运动（除了跟随主轴以外），在下列条件下：

- 如果 FOLL_BIAS_REMAINS SlaveMode 位设置为 0，则向正在进行的运动配置文件发出一个停止命令。从轴在跟随主轴的运动的任何减速过程结尾和正在进行的运动命令停止时达到零速度。然后，跟随被禁用，并且 AXIS_IS_LINKED 运动状态位被设置为 0。
- 如果 FOLL_BIAS_REMAINS SlaveMode 位已设置，则不会停止正在进行的运动命令。因此，跟随主轴的运动为零，并且 AXIS_IS_LINKED 位设置为 0。**从轴继续根据正在进行的运动命令运动。** AbsoluteSlaveBias 保持有效，直到从轴接收到停止或禁用命令 (%lr.m.c.24 = 1)。这意味着任何 ABS_MOVE 命令都将偏移 AbsoluteSlaveBias 的值。

为了在 AbsoluteSlaveBias 保持有效时命令从轴进行绝对运动，您必须从命令的位置中减去 AbsoluteSlaveBias：

命令的位置 = 所需位置 - AbsoluteSlaveBias

从轴完成暂停或禁用命令后，AbsoluteSlaveBias 自动复位，此时可发送正常位置命令。

在比率和凸轮模式中激活跟随

在每种模式中，激活跟随都会导致运动控制器计算新的 SlaveBias 值。选择新值是为了阻止从轴在给定当前主轴位置和从轴比率（对于比率模式）或凸轮配置文件（对于凸轮模式）的情况下运动。可以在激活跟随后将 SlaveBias 调整到任何所需值，或者通过向从轴发出增量运动命令进行调整。

注

在比率模式下，SlaveBias 值随 SlaveRatio 而更改。请在从轴的 RAMPING 位复位后监控 SlaveBias。

注意：SlaveBias：主轴与从轴之间的位置增量。

"目标序列激活" 功能

目的

此功能用于将从轴和主轴同步到预设位置。

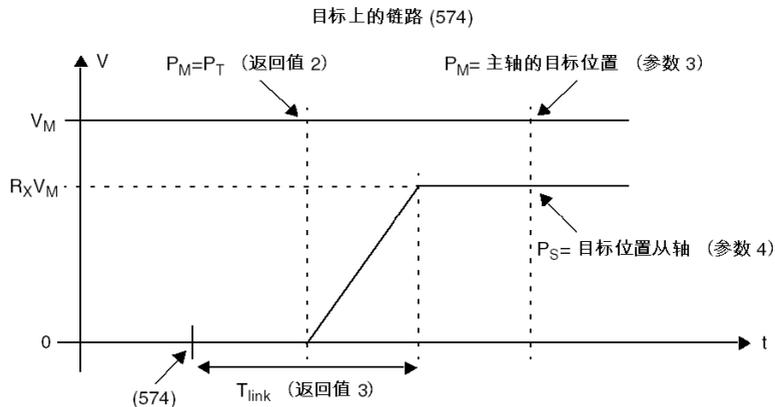
此同步使用 "SetMasterTriggerPosition" 和 "FollowOn" 功能执行。

操作

当主轴正在运动时，从轴停止。

用户使用操作码 OPCODE 574 "LinkOnTarget" 激活该功能。一旦接收到请求，该模块将计算主轴的阈值位置。

当主轴的位置超过由系统计算的阈值时，从轴将根据设置的加速斜坡开始其运动。



从轴将考虑:

- "从轴 / 主轴的目标位置" 扭矩,
- 当前主轴位置,
- 当前主轴速度,
- 当前从轴位置,
- 主轴 / 从轴比率,
- 为从轴配置的加速度。

此功能适用于主轴在该功能的启动时间和序列激活 "位 **AXIS_IS_LINKED = 1** (参见第 155 页)" 之间保持匀速的情况。而且，主轴不能是远程轴。

"LinkOnTarget" 功能

- 允许**两个旋转方向**（正向和负向），
- 主轴可以是**实轴或虚轴**，
- 主轴可以用模数配置，
- 从轴可以用模数配置。

命令由发送到从轴组通道的 WRITE_CMD 给出：

对象	类型	符号	说明	注释
%MWr.m.c.19	字	ERROR_CMD		命令被拒绝 *
%MDr.m.c.20	双字	RETURN_CMD_1		
%MFr.m.c.22	浮点数	RETURN_CMD_2		主轴的激活位置 (等同于 "GetMasterTrigger Position (参见 第 222 页)" 功能)
%MFr.m.c.24	浮点数	RETURN_CMD_3		从轴序列激活前的 剩余时间 (以秒为 单位)
%MWr.m.c.26	字	ACTION_CMD	574	LinkOnTarget
%MDr.m.c.27	双字	PARAM_CMD_1	SlaveNumber	从轴到链路的 SERCOS 地址
%MDr.m.c.29	双字	PARAM_CMD_2	0	未使用
%MFr.m.c.31	浮点数	PARAM_CMD_3	MasterPosition	主轴的目标位置
%MFr.m.c.33	浮点数	PARAM_CMD_4	SlavePosition	从轴的目标位置

*** 命令被拒绝：**

- 从轴已链接：69
- 从轴正在运动中：70
- 主轴已停止：72
- 主轴旋转方向错误：71
- 主轴未配置：37
- 主轴是远程轴：37
- 从轴未在此组中配置：37
- 从轴未配置：7003
- 组未配置：7002

注：

主轴的速度必须在序列激活请求与序列激活这两个时间之间保持匀速（"位
AXIS_IS_LINKED = 1（参见第 155 页）"）。

" 序列激活前的剩余时间估计 " 功能

目的

估计从轴组中的从轴序列激活前的剩余时间。

操作

此功能可用于从轴组。从轴将趋于停止。主轴正在运动。正在从轴上进行序列激活请求。此功能返回从轴排序激活生效 (" 位 **AXIS_IS_LINKED =1** (参见第 155 页)") 之前的估计剩余时间，即主轴到达从轴正在等待的位置阈值所需的时间，以及从轴达到其设定点速度 (主轴速度 * 序列比率) 所需的时间。此功能假设主轴速度将在启动此功能与从轴排序激活这两个时刻之间保持匀速。

"GetTimeToLink" 功能

- 允许两个旋转方向，
- 主轴可以是**实轴**或**虚轴**，
- 主轴可以用模数配置，
- 从轴可以用模数配置。

命令由发送到从轴组通道的 WRITE_CMD 给出：

对象	类型	符号	说明	注释
%MWr.m.c.19	字	ERROR_CMD		命令被拒绝 *
%MDr.m.c.20	双字	RETURN_CMD_1		
%MFr.m.c.22	浮点数	RETURN_CMD_2		从轴序列激活前的估计剩余时间
%MFr.m.c.24	浮点数	RETURN_CMD_3		
%MWr.m.c.26	字	ACTION_CMD	575	GetTimeToLink
%MDr.m.c.27	双字	PARAM_CMD_1	SlaveNumber	从轴的 SERCOS 地址
%MDr.m.c.29	双字	PARAM_CMD_2		
%MFr.m.c.31	浮点数	PARAM_CMD_3		
%MFr.m.c.33	浮点数	PARAM_CMD_4		

*** 命令被拒绝:**

- 主轴未配置: 37
- 主轴是远程轴: 37
- 从轴未配置: 37
- 从轴已链接: 69
- 从轴正在运动中: 70
- 主轴已停止: 72
- 从轴加速度为零: 1
- 主轴旋转方向错误: 71

注:

主轴的速度必须在使用此命令与序列激活这两个时间之间保持匀速 ("位 **AXIS_IS_LINKED = 1** (参见第 155 页)")。

触发器引起的从轴组序列停止 (Unlink)

目的

作为从轴组一部分的某个轴在位置激活时发生序列停止。

"SetTriggerUnlink" 功能

此命令涉及：

使用模数或不使用模数配置并且已链接的实轴或虚轴。

此命令由 "SetTriggerUnlink" 命令基于 "Write_cmd" 执行：

对象	类型	符号	说明	注释
%MWr.m.c.19	字	ERROR_CMD		错误代码 *
%MDr.m.c.20	双字	RETURN_CMD_1		
%MFr.m.c.22	浮点数	RETURN_CMD_2		
%MFr.m.c.24	浮点数	RETURN_CMD_3		
%MWr.m.c.26	字	ACTION_CMD	2180	SetTriggerUnlink
%MDr.m.c.27	双字	PARAM_CMD_1	从站号	
%MDr.m.c.29	双字	PARAM_CMD_2	触发器类型	零 (0), ** > (1), < (2), ≥ (3), ≤(4).
%MFr.m.c.31	浮点数	PARAM_CMD_3	触发器位置	
%MFr.m.c.33	浮点数	PARAM_CMD_4	0	

接收到命令时，轴（链接组的一部分）自动在触发器条件变为真时立即解除链接 (Unlink)。

** 触发器类型：

- 0 = 立即序列停止，
- 1 = 在主轴位置超过正向阈值时序列停止，
- 2 = 在主轴位置超过负向阈值时序列停止，
- 3 = 在主轴位置大于或等于阈值时序列停止，
- 4 = 在主轴位置小于或等于阈值时序列停止。

* 错误代码：该命令在以下条件下被拒绝：

- 轴未链接 (CMD_NOT_ALLOW (50))，
- 触发器位置在软件停止范围外 (RANGE_ERROR (1))，
- 触发器类型未知 (RANGE_ERROR (1))，
- 从轴编号无效。

读取由 SetTriggerUnlink 命令分配的上一个位置触发器

"GetTriggerUnlinkPosition" 命令用于读取由前一个 "SetTriggerUnlink" 分配的上一个位置触发器:

对象	类型	符号	说明	注释
%MWr.m.c.19	字	ERROR_CMD		错误代码 *
%MDr.m.c.20	双字	RETURN_CMD_1		
%MFr.m.c.22	浮点数	RETURN_CMD_2	触发器位置	
%MFr.m.c.24	浮点数	RETURN_CMD_3		
%MWr.m.c.26	字	ACTION_CMD	1180	GetTriggerUnlink Position
%MDr.m.c.27	双字	PARAM_CMD_1	从站号	
%MDr.m.c.29	双字	PARAM_CMD_2		
%MFr.m.c.31	浮点数	PARAM_CMD_3		
%MFr.m.c.33	浮点数	PARAM_CMD_4		

* 错误代码: 如果从轴编号无效, 则命令被拒绝。

读取由 SetTriggerUnlink 命令分配的上一个触发器类型

"GetTriggerUnlinkType" 命令用于读取由前一个 "SetTriggerUnlink" 分配的上一个触发器类型:

对象	类型	符号	说明	注释
%MWr.m.c.19	字	ERROR_CMD		错误代码 *
%MDr.m.c.20	双字	RETURN_CMD_1	触发器类型	零 (0) > (1), < (2), ≥ (3), ≤ (4).
%MFr.m.c.22	浮点数	RETURN_CMD_2		
%MFr.m.c.24	浮点数	RETURN_CMD_3		
%MWr.m.c.26	字	ACTION_CMD	1181	GetTriggerUnlink Type
%MDr.m.c.27	双字	PARAM_CMD_1	从站号	
%MDr.m.c.29	双字	PARAM_CMD_2		
%MFr.m.c.31	浮点数	PARAM_CMD_3		
%MFr.m.c.33	浮点数	PARAM_CMD_4		

* 错误代码: 如果从轴编号无效, 则命令被拒绝。

协同轴组的运动功能与 MMS Quantum 功能兼容

说明

此功能导致指定轴以指定速度向给定位置运动。该运动立即应用，并忽略当前正在进行的任何其他运动。正在进行的运动和任何排入队列的运动都被取消。这些运动与 MMS Quantum 模块生成的轨道相同。

立即运动可以是：

- 递增的：运动相对于当前位置 (31) 进行。
- 绝对的：绝对运动导致轴运动到某个绝对位置 (30)。

如何使用此功能

此功能是在 `WRITE_CMD` 指令（请参见 *WRITE_CMD*, 第 388 页）的帮助下实现的。

关联的语言对象

通过 `WRITE_CMD` 进行交换：下表显示与此功能关联的不同语言对象。

对象	名称	功能代码	说明
%MWr.m.c.26	ACTION_CMD	513 或 520	命令
%MDr.m.c.27	PARAM_CMD_1	-	运动类型 (1)
%MDr.m.c.29	PARAM_CMD_2	-	应用于已配置的加速度或减速度的系数，显示为千分数 ($1 \leq x \leq 10000$)。请参见操作码 2172、实轴 (参见第 438 页) 或虚轴 (参见第 442 页)。
%MFr.m.c.A (3)	PARAM_CMD_B (3)	-	轴 N (2) 的目标位置
%MFr.m.c.C (4)	PARAM_CMD_D (4)	-	到达 N 轴 (5) 的目标的接近速度
%MWr.m.c.19	ERROR_CMD	-	错误报告。请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页。
说明			
(1): ABS_MOVE_NO_ACC_CORRECTION, INCR_MOVE_NO_ACC_CORRECTION			
(2): N = 1 表示独立轴，N 包括 2 和 8 之间的值表示协同轴组。			
(3): A = [31+2(N-1)], B = [3+(N-1)]			
(4): C = [31+2(N+N'-1)], D = [3+(N+N'-1)]			
(5): N = 轴在组中的位置			

示例

对于具有 2 个成员轴的运动:

- PARAM_CMD_1 (%MDr.m.c.27) = 运动类型
- PARAM_CMD_3 (%MFr.m.c.31) = 轴 1 的位置
- PARAM_CMD_4 (%MFr.m.c.33) = 轴 2 的位置
- PARAM_CMD_5 (%MFr.m.c.35) = 轴 1 的接近速度
- PARAM_CMD_6 (%MFr.m.c.37) = 轴 2 的接近速度

FollowOn 415 功能描述

简介

FollowOn 415 功能指定轴（已是多个 FollowerSet 的成员）将链接到哪个 FollowerSet。

通过添加轴的标识符作为 FollowOn 415 功能的参数，FollowerSet 的每个轴成员都可以分别链接到其主轴。

操作模式

当一个轴属于多个 FollowerSet 时，FollowOn 415 功能允许选择将激活其监控功能的 FollowerSet。运动会链接到已激活其监控功能的 FollowerSet 的主轴。

说明

FollowOn 415 功能激活对以下对象的监控功能：

- 在不使用任何参数时的 FollowerSet 的每一个轴成员
- 将其标识符用作参数的轴

此功能仅适用于一个 FollowerSet。

注意：

该操作不适用于以下轴：

- 不属于 FollowerSet 的轴。
- 已激活监控功能的轴。

命令

指令的语法：

```
ACTION_CMD (%MWxy.i.26) = 415
```

命令参数

下表列出了适用于 FollowOn 415 功能的命令参数：

标准符号	值	地址
PARAM_CMD_1	要链接的组的从轴成员地址。	%MDxy.i.27
PARAM_CMD_2	0	%MDxy.i.29
PARAM_CMD_3	0.0	%MDxy.i.31
PARAM_CMD_4	0.0	%MDxy.i.33

返回命令值

下表列出了 FollowOn 415 功能的命令返回值：

标准符号	值	地址
RETURN_CMD_1	0	%MDxy.i.20
RETURN_CMD_2	0.0	%MDxy.i.22
RETURN_CMD_3	0.0	%MDxy.i.24

错误代码

下表列出了 FollowOn 415 功能发送的错误代码：

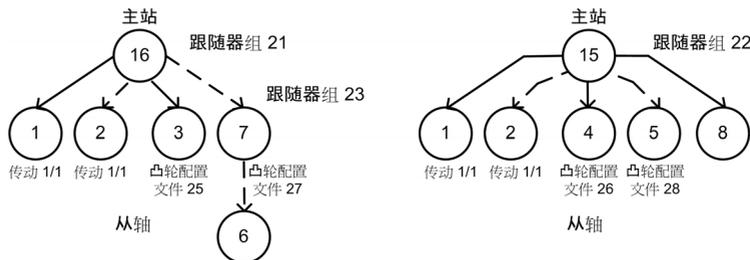
错误代码	标准符号	含义	地址
38	NO_SUCH_AXIS	该轴不属于此 FollowerSet。	-
42	AXIS_NOT_ALLOWED	该轴未经过验证。	-
50	CMD_NOT_ALLOWED	不允许监控该轴。	-
69	AXIS_LINKED	此轴已链接。	-
7002	BAD_ID	FollowerSet 未配置。	-
7003	WARN_BAD_DATA	从轴未配置。	-
-	ERROR_CMD	显式写入 WRITE_CMD 命令时可能出现的错误。	%MWxy.i.19

注意：在将轴链接到不属于另一个 FollowerSet 的主轴之前，必须停止具有以下输出的轴监控功能：

```
%Qxy.i.27 :ALLOW_FOLLOW = 0
```

示例

在下面的示例中，轴 1 和 2（传动 1/1）为跟随器组 21 和 22 共用。即使跟随器组 21 和 22 都处于活动状态，轴 1 和 2 也可以使用 FollowOn 415 功能随时在组 21 和 22 之间切换。



调整 TSX CSY 84 和 TSX CSY 164 模块

13

本章的目标

本章描述调整 SERCOS(r) 应用专用参数的原理。

用于 TSX CSY 84 的所有步骤与用于 TSX CSY 164 模块的步骤相同。

本章包含了哪些内容？

本章包含了以下部分：

节	主题	页
13.1	SERCOS(r) 模块调整：一般信息	228
13.2	实轴、虚轴或远程轴的参数调整	232
13.3	从轴组的参数	263
13.4	凸轮配置文件调整	284
13.5	SERCOS(r) 环路参数	296
13.6	SERCOS(r) IDN 读 / 写功能	301

13.1 SERCOS(r) 模块调整：一般信息

本节主题

本小节介绍 SERCOS(r) 模块的调整屏幕，并给出显式交换的一般原理的一些提示。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
SERCOS(r) 模块调整屏幕描述：一般信息	229
调整原理	231

SERCOS(r) 模块调整屏幕描述：一般信息

简介

调整屏幕显示 SERCOS(r) 通道的调整参数。

这些参数可在离线或在线模式中修改。

它还用于访问配置屏幕和调试屏幕。

调整屏幕与配置屏幕的结构非常相似。

注意：有关配置、调整和调试项目的过程简介，请参见关于常用应用专用功能的部分。

示意图

下图显示了 TSX CSY 84 模块的调整屏幕的示例：



描述

下表显示了调试屏幕的各个元素。

地址	元素	功能
1	选项卡	前景中的选项卡指示当前模式（此示例中为 调试 ）。使用各选项卡可选择相应的模式。可用的模式为： <ul style="list-style-type: none"> ● 调试，只能在在线模式下访问， ● 诊断（故障），只能在在线模式下访问， ● 调整， ● 配置。
2	模块区域	显示模块的简短标题。 同一区域中有 3 个 LED，用于指示在线模式中的模块状态： <ul style="list-style-type: none"> ● RUN，指示模块的操作模式。 ● ERR，指示内部模块错误。 ● I/O，指示模块外部故障或应用故障。
3	通道区域	允许您： <ul style="list-style-type: none"> ● 通过单击设备参考显示选项卡： <ul style="list-style-type: none"> ● 描述：提供设备特性。 ● I/O 对象（参见 <i>Unity Pro, 操作模式</i>，）提供输入 / 输出对象的预览符号。 ● 故障：提供对设备故障的访问（在线模式下）。 ● 选择通道， ● 显示符号，即用户（通过变量编辑器）定义的通道名。
4	常规参数区域	用于取消强制位和查看计数功能： <ul style="list-style-type: none"> ● 取消强制：用于取消强制使用强制位的按钮。 ● 功能：取消已配置的计数功能。此信息不能修改。 ● 任务：显示已配置的 MAST 或 FAST 任务。此信息不能修改。
5	当前参数区域	此区域显示 I/O 状态和正在进行的计数的不同参数。如果计数寄存器的内容在输入故障后不可用，则消息或 LED 无效测量 显示为红色。 注 ：仅对于 CTY 2A/4A 模块，可以在位于右侧参数区域的窗口中选择要在扩展屏幕中显示的功能。对于 CTY 2C 模块，所有功能都会自动显示。

调整原理

简介

调整功能用于查看和修改模块的操作参数。

TSX CSY 84/164 模块参数的调整是通过以下方式之一进行的：

- 直接使用 Unity Pro 软件 1.0 或更高版本的调整屏幕（离线或在线模式），
- 或在项目与轴控模块之间使用显式交换（参见第 377 页）（在线）。

注意：当使用 Set... 调整功能（WRITE_CMD 指令）更改参数时，不保证模块与内部字 %MW 中包含的值一致。

启用显式交换的通道

显式命令用于与以下通道交换参数：

- 通道 0（网络）
- 通道 1 至 16（独立轴）
- 通道 21 至 24（从轴组）

13.2 实轴、虚轴或远程轴的参数调整

本节主题

本小节介绍实轴、虚轴或远程轴的调整屏幕以及不同可访问参数。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
SERCOS(r) 模块的调整屏幕描述：实轴或虚轴	233
SERCOS(r) 模块的调整屏幕描述：远程轴	235
实轴的 TRF_RECIPE 屏幕描述	236
TRF_RECIPE	238
TRF_RECIPE 示例	240
速度参数：DefaultSpeed	241
速度参数：SpeedLimit	242
关于加速度 / 减速度参数	243
加速度 / 减速度参数：Accel	244
加速度 / 减速度参数：Decel	245
加速度 / 减速度参数：AccelMax	246
加速度 / 减速度参数：DecelMax	247
加速度 / 减速度参数：AccelType	248
数据类型 AccelerationType	250
精度参数：GearRatio	251
位置限制和模数参数：PositionLimit	253
关于模数	255
位置限制和模数参数：RolloverLimit	256
位置限制和模数参数：EnableRollover	257
关于正确位置带	258
位置限制参数：InPositionBand	259
关于启用模式	260
位置限制参数：EnableMode	261
从轴禁用行为	262

SERCOS(r) 模块的调整屏幕描述：实轴或虚轴

简介

此屏幕用于在在线模式下直接修改实轴或虚轴的配置参数。

注意： 有关不同调整屏幕的公共部分的更多信息，请参见 *SERCOS(r)* 模块调整屏幕描述：一般信息，第 229 页。

示意图

下图显示了在实轴或虚轴中配置的 SERCOS(r) 通道的调整区域。

The screenshot shows a configuration window for a SERCOS(r) module. It is divided into several sections:

- 限制 (Limits):** Includes a checked '位置检查' (Position Check) box. Below it are input fields for '最大位置' (5.000000e+006), '最小位置' (5.000000e+002), '最高速度' (5.000000e+005), '最大加速度' (5.000000e+004), and '最小减速度' (5.000000e+002). At the bottom of this section are '启用位置检查' (Enable Position Check) with an unchecked '活动' (Active) box and a '公差' (Tolerance) field set to 0.000000e+000.
- 单位 (Units):** A '类型' (Type) dropdown set to '线性' (Linear). Below are dropdowns for '位置' (微米), '速度' (微米/秒), and '加速度' (微米/平方秒). A '比例系数' (Scale Factor) section has '分子' (分子) and '分母' (分母) fields both set to 1.000000e+000.
- 运动 (Motion):** An unchecked '模数' (Modulus) box. Below are input fields for '最大模数' (0.000000e+000), '最小模数' (0.000000e+000), '正确位置带' (7.000000e+000), '加速度' (1.000000e+002), and '减速度' (1.000000e+001). At the bottom is an '加速度类型' (Acceleration Type) dropdown set to '矩形 100%'.

描述

下表显示了调整区域的限制。

元素	关联的参数	关联的位 / 字语言	符号 / 描述
位置检查	-	%MWr.m.c.35:X4	此复选框用于激活位置限制的控制。
最大位置	PositionLimits	%MWr.m.c.55	POSITION_MAX
最小位置		%MWr.m.c.57	POSITION_MIN
最高速度	SpeedLimit	%MWr.m.c.53	SPEED_MAX
最大加速度	AccelMax	%MWr.m.c.49	ACCEL_MAX
最小加速度	DecelMax	%MWr.m.c..	DECEL_MAX

下表显示了调整区域的验证位置控制。

元素	关联的参数	关联的位 / 字语言	描述
活动	-	%MWr.m.c.35:X1	此复选框用于启用位置检查。
公差	-	%MWr.m.c.43	监控窗口的值（不能修改）

下表显示了调整区域的**单位**。此信息不能修改。

元素	关联的参数	关联的语言字	符号
类型	-	%KW _r .m.c.34	-
位置	PositionUnits	%MW _r .m.c.65	-
速度	VelocityUnits	%MW _r .m.c.64	-
加速度	AccelUnits	%MW _r .m.c.63	-

下表显示了调整区域的**比例系数**。

元素	关联的参数	关联的语言字	符号
分子	GearRatio	%MW _r .m.c.59	SCALE_NUMERATOR
分母		%MW _r .m.c.61	SCALE_DENOMINATOR

下表显示了调整区域的**运动**。

元素	关联的参数	关联的位 / 字语言	符号
模数	EnableRollover	%MW _r .m.c.35:X0	ENABLE_ROLLOVER
最大模数	RolloverLimit	%MW _r .m.c.45	ROLLOVER_MAX
最小模数		%MW _r .m.c.47	ROLLOVER_MIN
正确位置带	InPositionBand	%MFr.m.c.41	IN_POSITION_BAND
加速度	Accel	%MW _r .m.c.36	ACCEL
减速度	Decel	%MW _r .m.c.38	DECEL
加速度类型	AccelType	%MW _r .m.c.40	ACCEL_TYPE

SERCOS(r) 模块的调整屏幕描述：远程轴

简介

此屏幕用于在在线模式下直接修改远程轴的配置参数。

注意： 有关不同调整屏幕的公共部分的更多信息，请参见 *SERCOS (r)* 模块调整屏幕描述：一般信息，第 229 页。

示意图

下图显示了在远程轴中配置的 SERCOS(r) 通道的调整区域。

描述

下表显示了调整区域的**比例系数**。

元素	关联的参数	关联的位 / 字语言	符号 / 描述
分子	FollowerRatio	%MFr.m.c.Y (2)	NUMERATOR_X (1)
分母			DENOMINATOR_X (1)
说明			
(1): X 包括在 1 和 8 之间			
(2): Y 的值，请参见 <i>从轴组参数: FollowerRatio</i> ，第 271 页。			

下表显示了调整区域的**模数**。

元素	关联的参数	关联的位 / 字语言	符号 / 描述
活动	EnableRollover	%MWr.m.c.35:X0	ENABLE_ROLLOVER
最大值	RolloverLimit	%MWr.m.c.45	ROLLOVER_MAX
最小值		%MWr.m.c.47	ROLLOVER_MIN

下表显示了调整区域的**单位**。此信息不能修改。

元素	关联的参数	关联的语言字	符号
位置	PositionUnits	%MWr.m.c.65	-

实轴的 TRF_RECIPE 屏幕描述

简介

此屏幕位于实轴调整屏幕的底部，用于在 PLC 与伺服驱动器之间传输调整参数。

此功能必须在在线模式下使用。

注意：有关不同调整屏幕的公共部分的更多信息，请参见 *SERCOS (r)* 模块调整屏幕描述：一般信息，第 229 页。

示意图

下图显示了与 TRF_RECIPE 功能关联的屏幕。

The screenshot shows a control interface for the TRF_RECIPE function. It is titled "传输伺服驱动器的参数" (Parameters for transmitting to the servo driver). The interface is divided into several sections:

- 传输方向 (Transmission Direction):** Contains radio buttons for "读取" (Read) and "写入" (Write). The "写入" option is selected. There are also checkboxes for "主轴" (Main axis) and "从轴" (Slave axis), both of which are checked. A "传输" (Transmit) button and a radio button for "传输正在进行" (Transmission in progress) are also present.
- 参数 (Parameters):** Labeled "传输伺服驱动器参数到 PLC 存储器" (Transmit servo driver parameters to PLC memory). It includes a dropdown menu for "地址表" (Address table) set to "%MW", a text input for "100", and a text input for "长度" (Length) set to "200".
- 结果 (Results):** Labeled "传输错误" (Transmission error) with a text input showing "22", and "读取的表长度" (Read table length) with a text input showing "0".

描述

下表显示了 TRF_RECIPE 区域的**传输方向**。

元素	关联的语言字	符号
PLC -> 伺服驱动器	%MWr.m.c.10	ACTION_TRF
伺服驱动器 -> PLC		

下表显示了 TRF_RECIPE 区域的**参数**。

元素	关联的语言字	描述
地址表	%MDr.m.c.11	可以修改两个参数： 用于存储数据的对象存储器的类型： <ul style="list-style-type: none"> ● 读写状态的 %MW ● 只读的 %KW
	立即值	表的开始地址
长度	立即值	表长度

下表显示了 TRF_RECIPE 区域的结果。

元素	关联的语言字	符号
传输错误	%MWr.m.c.3	此字段在传输无效的情况下提供错误代码 (参见第 363 页)。
读取的表长度	%MDr.m.c.4	表的实际长度

TRF_RECIPE

提示

此服务用于读取或写入与 " 实轴 " 功能相关的变速控制器参数。

注意：此服务同时用于读取和写入凸轮配置文件以及开始特殊功能的执行（请参见有关这些功能的信息）。

TRF_RECIPE 指令的语法

TRF_RECIPE (IODDT_Var, 长度, 地址 %MW): 从变速控制器参数、凸轮配置文件或特殊功能的参数传入 / 传出入以地址 %MW 开始的表。要传输的表长度由 length 参数定义。要执行的动作由字 %MWr.m.c.10 (ACTION_TRF) 定义。

示例：TRF_RECIPE (%CH1.4.3, 10, 100)（写入位于机架 1 位置 4 中的模块的实轴 3 的变速控制器参数，表长度为 10，表地址从 %MW100 开始）。

TRF_RECIPE 接口

要执行的命令在字 %MWr.m.c.10 中定义，命令的结果在字 %MWr.m.c.3 至 %MWr.m.c.8 中提供。

地址	类型	符号	含义
%MWr.m.c.3	字	ERROR_TRF	TRF_RECIPE 命令的写入错误
%MDr.m.c.4	双字	RETURN_TRF_1	功能的返回值 1
%MFr.m.c.6	浮点数	RETURN_TRF_2	功能的返回值 2
%MFr.m.c.8	浮点数	RETURN_TRF_3	功能的返回值 3
%MWr.m.c.10	字	ACTION_TRF	要执行的动作
%MDr.m.c.11	双字	PARAM_TRF_1	参数 1
%MDr.m.c.13	双字	PARAM_TRF_2	参数 2
%MFr.m.c.15	浮点数	PARAM_TRF_3	参数 3
%MFr.m.c.17	浮点数	PARAM_TRF_4	参数 4

TRF_RECIPE 执行的动作

可使用 TRF_RECIPE 服务执行的动作为：

功能	ACTION_TRF (%MWr.m.c.10)	含义
实轴 (1)	16001	将变速控制器参数写入到 PLC 存储器。
实轴 (1)	26001	从 PLC 存储器中清除变速控制器参数。

说明

(1)	PARAM_TRF_1 至 PARAM_TRF_4 = 0
-----	-------------------------------

交换

描述

1	将参数写入保存的存储器（实轴，WRITE_CMD）
2	从保存的存储器中清除参数（实轴，WRITE_CMD）
3	将参数写入 PLC 存储器（实轴，ACTION_TRF = 16001）
4	从 PLC 存储器中清除参数（实轴，ACTION_TRF = 26001）

TRF_RECIPE 示例

示例 1: 保存变速控制器参数

将变速控制器参数从位于机架 1 位置 4 中的模块的实轴 3 传输到内部字 %MW100 至 %MW1100，以便保存它们。表的长度为 1000 个字。所使用的 IODDT 类型的变量 T_CSY_TRF 名为 VAR_TRF_1，并与模块的通道 3 关联。

```
! (* 如果没有正在进行的参数交换，则将参数表传输到 PLC 中 *)  
VAR_TRF_1.ACTION_TRF := 16001;  
TRF_RECIPE (VAR_TRF_1, 1000, 100);  
END_IF;
```

速度参数: DefaultSpeed

描述

此参数用于定义轴以 SpeedUnits（速度单位）给出的缺省运动速度值。以等于 0 的速度运动。

注意：缺省情况下，系统应用一个等于 SpeedLimit 值的十分之一的缺省值。

关联的功能

此参数的关联功能包括：

- GetDefaultSpeed，它发送针对实际缺省值的读取请求，
- SetDefaultSpeed，它发送针对新的缺省值的写入请求。

这些功能使用 WRITE_CMD 指令。

关联的语言对象

下表显示与 DefaultSpeed 参数关联的不同语言对象。

对象	名称	功能代码	描述
%MWr.m.c.26	ACTION_CMD	1065	读取命令。
		2065	写入命令。
%MFr.m.c.22	RETURN_CMD_2	-	读取的结果。
%MFr.m.c.31	PARAM_CMD_3	-	要写入的值。
%MWr.m.c.19	ERROR_CMD	-	错误报告。（请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页）

速度参数: SpeedLimit

描述

此参数用于定义以 SpeedUnits（速度单位）给出的轴的最大速度。

此限制适用于由轴控模块发送到变速控制器的速度。

最大速度必须大于或等于实际速度。

注意：变速控制器具有自己的速度限制。

关联的功能

此参数的关联功能包括：

- GetSpeedLimit，它发送针对实际最大速度的读取请求，
- SetSpeedLimit，它发送针对新的最大速度的写入请求。

这些功能使用（根据语言对象）WRITE_CMD、READ_PARAM 或 WRITE_PARAM 指令。

语言对象

下表显示与 SpeedLimit 参数关联的不同语言对象。

对象	名称	功能代码	指令类型	描述
%MFr.m.c.53	SPEED_MAX	-	READ_PARAM	读取命令。
			WRITE_PARAM	写入命令。
%MW.r.m.c.26	ACTION_CMD	1066	WRITE_CMD	读取命令。
		2066		写入命令。
%MFr.m.c.22	RETURN_CMD_2	-	-	读取的结果。
%MFr.m.c.31	PARAM_CMD_3	-	-	要写入的值。
%MW.r.m.c.19	ERROR_CMD	-	-	错误报告。（请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页）。

关于加速度 / 减速度参数

定义

加速度用于提高轴沿某个运动配置文件运动的速度。加速度值被限制为最大加速度的值。

减速度用于降低轴沿某个运动配置文件的速度。减速度值被限制为最大减速度值。

加速度 / 减速度斜坡应用于该轴的运动命令。

运动计划

轨道发生器使用此加速度 / 减速度计算发送到受控驱动器的命令。受控驱动器具有独立的加速度 / 减速度参数。对于回归，驱动器使用 SERCOS(r) 标准回归加速度 / 减速度参数（标识号 - IDN S-0-0042）。对于紧急停止，驱动器可以使用特殊的减速度值。（请参见驱动器的专用 (P) IDN 列表。）

最大加速度

伺服驱动器的最大加速度 / 减速度限制（SERCOS(r) 标准 IDN S-0-0138 双向加速度限制或 IDN S-0-0136 正向加速度限制）必须大于运动控制器的 AccelMax 和 DecelMax 参数。

加速度 / 减速度参数: Accel

描述

此参数用于设置要应用于某个轴上的运动命令的加速斜坡。
此加速度的值使用 AccelerationUnit 指定。

关联的功能

此参数的关联功能包括:

- GetAccel, 它发送针对指定轴的实际加速斜坡值的读取请求
- SetAccel, 它发送针对指定轴的新加速斜坡值的写入请求

这些功能使用 (根据语言对象) WRITE_CMD、READ_PARAM 或 WRITE_PARAM 指令。

语言对象

下表显示与 Accel 参数关联的不同语言对象。

对象	名称	功能代码	指令类型	描述
%MFr.m.c.36	ACCEL	-	READ_PARAM	读取命令
			WRITE_PARAM	写入命令
%MW.r.m.c.26	ACTION_CMD	1041	WRITE_CMD	读取命令
		2041		写入命令
%MFr.m.c.22	RETURN_CMD_2	-	-	读取的结果。
%MFr.m.c.31	PARAM_CMD_3	-	-	要写入的值。
%MW.r.m.c.19	ERROR_CMD	-	-	错误报告。(请参见 <i>WRITE_CMD</i> 命令: 编程错误, 第 357 页)

加速度 / 减速度参数: Decel

描述

此参数用于设置要应用于某个轴上的运动命令的减速斜坡。
此减速度的值使用 AccelerationUnit 指定。

关联的功能

此参数的关联功能包括:

- GetDecel, 它发送针对指定轴的实际减速斜坡值的读取请求
- SetDecel, 它发送针对指定轴的新减速斜坡值的写入请求

这些功能使用 (根据语言对象) WRITE_CMD、READ_PARAM 或 WRITE_PARAM 指令。

语言对象

下表显示与 AccMax 参数关联的不同语言对象。

对象	名称	功能代码	指令类型	描述
%MFr.m.c.38	DECEL	-	READ_PARAM	读取命令
			WRITE_PARAM	写入命令
%MW.r.m.c.26	ACTION_CMD	1042	WRITE_CMD	读取命令
		2042		写入命令
%MFr.m.c.22	RETURN_CMD_2	-	-	读取的结果。
%MFr.m.c.31	PARAM_CMD_3	-	-	要写入的值。
%MW.r.m.c.19	ERROR_CMD	-	-	错误报告。(请参见 <i>WRITE_CMD</i> 命令: 编程错误, 第 357 页)

加速度 / 减速度参数: AccelMax

描述

此参数用于定义应用于某个轴的最大加速度。

此加速度的值使用单位 UnitsAcceleration 来指定。

注意: 最大加速度必须大于或等于当前加速度参数。

注意: 控速驱动器 (IDN 标准标识号 SERCOSS-0-0138 - 双向加速度限制, 或 IDN S-0-0136 - 正向加速度限制) 的最大加速度必须大于运动控制器的 AccelerationMax 参数。

相关功能

与此参数相关的功能为:

- GetAccelMax, 它发送读取当前最大加速度值的请求
- SetAccelMax, 它发送写入新的最大加速度值的请求

这些功能使用 (根据语言对象) WRITE_CMD、READ_PARAM 或 WRITE_PARAM 指令。

语言对象

下表显示与 AccelMax 参数关联的不同语言对象。

对象	名称	功能代码	指令类型	描述
%MFr.m.c.49	ACCEL_MAX	-	READ_PARAM	读取命令
			WRITE_PARAM	写入命令
%MWr.m.c.26	ACTION_CMD	1116	WRITE_CMD	读取命令
		2116		写入命令
%MFr.m.c.22	RETURN_CMD_2	-	-	读取的结果。
%MFr.m.c.31	PARAM_CMD_3	-	-	要写入的值。
%MWr.m.c.19	ERROR_CMD	-	-	错误报告。(请参见 <i>WRITE_CMD</i> 命令: 编程错误, 第 357 页)

加速度 / 减速度参数: DecelMax

描述

此参数用于定义应用于某个轴的最大减速度。

此减速度的值使用 AccelerationUnit 指定。

注意：最大减速度必须大于或等于实际减速度参数。

关联的功能

此参数的关联功能包括：

- GetDecelMax，它发送针对实际最大加速度值的读取请求
- SetDecelMax，它发送针对新的最大加速度值的写入请求

这些功能使用（根据语言对象）WRITE_CMD、READ_PARAM 或 WRITE_PARAM 指令。

语言对象

下表显示与 DecelMax 参数关联的不同语言对象。

对象	名称	功能代码	指令类型	描述
%MFr.m.c.51	DECEL_MAX	-	READ_PARAM	读取命令
			WRITE_PARAM	写入命令
%MWr.m.c.26	ACTION_CMD	1117	WRITE_CMD	读取命令
		2117		写入命令
%MFr.m.c.22	RETURN_CMD_2	-	-	读取的结果。
%MFr.m.c.31	PARAM_CMD_3	-	-	要写入的值。
%MWr.m.c.19	ERROR_CMD	-	-	错误报告。（请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页）

加速度 / 减速度参数: AccelType

描述

此功能用于定义轴的加速类型。

加速类型指定在更改轴的速度时使用的加速配置文件的形状。运动控制器提供三种主要的加速配置文件类型：

- **矩形**：此加速配置文件在最短的时间内更改速度。但是，它还对轴产生最大的撞击量
- **三角形**：此加速配置文件花最长的时间更改速度（比矩形加速配置文件慢两倍）。但是，它对轴产生最小的撞击量
- **梯形**：梯形配置文件比三角形加速配置文件更快但是比矩形加速配置文件更慢地更改速度。对轴的冲击量比三角形大，但是比矩形小。存在三种梯形加速配置文件选择，分别由每种选择完成速度更改所花的时间比矩形配置文件长多少来指定：
 - 比矩形配置文件长 1.25 倍
 - 长 1.5 倍
 - 长 1.75 倍

可在任何时刻修改加速类型。它将应用于队列中的运动。

一般情况下，运动控制器比轴当前所运动到的位置提前（两个点数）处理队列中命令。

注意：这是更改协同轴组的加速类型的唯一方法。加速类型缺省为矩形。

加速代码

可能的加速类型如下：

代码	类型	描述
0	RECTANGULAR_ACCEL	$t1 = 0, T_{acc} = T_{accrec}$
1	TRIANGULAR_ACCEL	$T_{acc} = 2T_{accrec}$
2	ACCEL_TRAP_125	$t2 = 3t1, T_{acc} = 1.25T_{accrec}$
3	ACCEL_TRAP_150	$t1 = t2, T_{acc} = 1.5T_{accrec}$
4	ACCEL_TRAP_175	$t1 = 3t2, T_{acc} = 1.75T_{accrec}$

关联的功能

此参数的关联功能包括：

- GetAccelType，它发送针对实际加速类型的读取请求
- SetAccelType，它发送针对新加速类型的写入请求

这些功能使用（根据语言对象）WRITE_CMD、READ_PARAM 或 WRITE_PARAM 指令。

关联的语言对象

下表显示与 `AccelType` 参数关联的不同语言对象。

对象	名称	功能代码	指令类型	描述
%MFr.m.c.40	ACCEL_TYPE	-	READ_PARAM	读取命令
			WRITE_PARAM	写入命令
%MWr.m.c.26	ACTION_CMD	1540	WRITE_CMD	读取命令
		2540		写入命令
%MFr.m.c.20	RETURN_CMD_1	-	-	读取的结果。
%MFr.m.c.27	PARAM_CMD_1	-	-	要写入的值。
%MWr.m.c.19	ERROR_CMD	-	-	错误报告。(请参见 <code>WRITE_CMD</code> 命令： 编程错误，第 357 页)

数据类型 AccelerationType

描述

加速类型指定在更改轴的速度时使用的加速配置文件的形状。运动控制器提供三种主要的加速配置文件类型：

- 矩形：此加速配置文件在最短的时间内更改速度。但是，它还对轴产生最大的撞击量
- 三角形：此加速配置文件花最长的时间更改速度（比矩形加速配置文件长两倍）。但是，它对轴产生最小的撞击量
- 梯形：梯形配置文件比三角形加速配置文件更快但是比矩形加速配置文件更慢地更改速度。对轴的冲击量比三角形大，但是比矩形小。存在三种梯形加速配置文件选择，分别由每种选择完成速度更改所花的时间比矩形配置文件长多少来指定：
 - 比矩形配置文件长 1.25 倍
 - 长 1.5 倍
 - 长 1.75 倍

数据类型的值

下表列出了各种数据值：

加速度类型	值
RECTANGULAR_ACCEL	0
TRIANGULAR_ACCEL	1
ACCEL_TRAP_125	2
ACCEL_TRAP_150	3
ACCEL_TRAP_175	4

精度参数: GearRatio

描述

此参数用于设置轴位置相对于轴的驱动器位置的比率。

比率

该比率表示电机与轴运动点（滑块、轮盘、心轴等）之间的物理传送。仅当传送发生变化时才应该更改此比率。必须将它设置为某个有效值才能进行正确的轴定位。比率功能为 GetGearRatio 和 SetGearRatio。

比率的定义为：

轴位置单位数 / 驱动器位置单位数

此比率与附加到电机的反馈设备的精度以及轴的缺省单位系统的精度无关。它取决于轴单位数：是线性的还是角度的。

下面是一个示例，其中 SERCOS(r) 驱动器上的驱动器位置单位数设置为转数：

- 一个导螺杆附加到电机并将轴移动 2 英寸 / 电机转，因此该比率设置为英寸 (2) / 转 (1)。
- 传送每 10 电机转将轮盘移动一转，因此比率设置为转 (1) / 转 (10)。

虚轴和比率

伺服驱动器不控制虚轴。驱动器位置单位数必须是编码器反馈计数。轴位置单位数可以是任何角度或线性单位数。该比率应该表示位置反馈设备的典型精度。例如：

比率：转 (1) / 反馈计数 (65536)

比率：毫米 (1) / 反馈计数 (1000)

远程轴和 GearRatio

RemoteSetpoint 的位置反馈是该轴的反馈寄存器。放在该寄存器中的 32 位整数被假设具有反馈计数单位数。选择该比率是为了转换为所需的轴单位数。例如：

比率：转 (1) / 反馈计数 (10000)

比率：毫米 (1) / 反馈计数 (1000)

关联的功能

此参数的关联功能包括：

- GetGearRatio，它发送针对实际比率的读取请求
- SetGearRatio，它发送针对新比率的写入请求

这些功能使用（根据语言对象）WRITE_CMD、READ_PARAM 或 WRITE_PARAM 指令。

关联的语言对象

下表显示与 Accel 参数关联的不同语言对象。

对象	名称	功能代码	指令类型	描述
%MFr.m.c.59	SCALE_NUMERATOR	-	READ_PARAM	读取命令
			WRITE_PARAM	写入命令
%MFr.m.c.61	SCALE_DENOMINATOR	-	READ_PARAM	读取命令
			WRITE_PARAM	写入命令
%MWr.m.c.26	ACTION_CMD	1500	WRITE_CMD	读取命令
		2500		写入命令
%MFr.m.c.22	RETURN_CMD_2	-	-	读取的结果（分子）。
%MFr.m.c.24	RETURN_CMD_3	-	-	读取的结果（分母）。
%MFr.m.c.31	PARAM_CMD_3	-	-	要写入的值（分子）。
%MFr.m.c.33	PARAM_CMD_4	-	-	要写入的值（分母）。
%MWr.m.c.19	ERROR_CMD	-	-	错误报告。（请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页）

位置限制和模数参数: PositionLimit

说明

此功能用于定义轴的正限位和负限位。

这些限位的值指定为 PositionUnit。

不能将正限位和负限位设置为导致当前位置超出新限位之外的值。在更改限位前，请使用 SetPosition 功能或某个运动功能将当前位置更改为在新限位之内的值。

注意：位置限制在执行 Home 参考点功能之前不是活动的。

关联的功能

此参数的关联功能包括：

- GetPositionLimit，它发送针对指定轴的实际正限位和负限位值（功能参数 PARAM_CMD_1）的读取请求
- SetPositionLimit，它发送针对指定轴的新正限位和负限位值（功能参数 PARAM_CMD_1）的写入请求

这些功能使用（根据语言对象）WRITE_CMD、READ_PARAM 或 WRITE_PARAM 指令。

注意：当需要增量运动并且目标位置超出**位置限制**（实轴或虚轴的 %MFr.m.c.55 和 %MFr.m.c.57）时，目标位置会将**位置限制**作为其目标，并且轴将在运动结束时切换到**限制故障**。协同类型的运动也是如此。

语言对象

下表显示与 PositionLimit 参数关联的不同语言对象。

对象	名称	功能代码	指令类型	说明
%MFr.m.c.55	POSITION_MAX	-	READ_PARAM	读取命令
			WRITE_PARAM	写入命令
%MFr.m.c.57	POSITION_MIN	-	READ_PARAM	读取命令
			WRITE_PARAM	写入命令
%MWFr.m.c.26	ACTION_CMD	1505	WRITE_CMD	读取命令
		2505		写入命令
%MFr.m.c.22	RETURN_CMD_2	-	-	读取的结果。
%MDr.m.c.27	PARAM_CMD_1	-	-	要写入的值： 1：正限位 -1：负限位
%MFr.m.c.31	PARAM_CMD_3	-	-	要写入的值：位置。
%MWFr.m.c.19	ERROR_CMD	-	-	错误报告。请参阅（ <i>WRITE_CMD</i> 命令： 编程错误，第 357 页）

关于模数

模数限位

可以配置单个轴在该轴越出正或负模数限位时回转位置读数。回转位置读数将约束位置读数，使其保持在负模数限位与正模数限位之间的值范围内。

例如，在控制轮盘并希望轴报告约束在 0 和 360 度之间的位置读数的时候。如果在轴到达 360 度的正模数限位时激活了模数，则回转位置读数，以读取 0 度的负限位值。

位置读数被约束为大于或等于负模数限位，或小于正模数限位。位置读数决不等于正模数限位。在这个轮盘示例中，模数限位介于 0 和 360 度之间，轴从不会返回 360 度的位置读数。但它会返回 0 度。

正模数限位与负模数限位之间的差称为模数范围。这个轮盘示例具有 360 度（或 1 转）的模数范围。

正模数限位和负模数限位可以是任意值，只要负限位小于且不等于正限位。负限位不必是负数，只需小于正限位。对于这个轮盘示例，常见的负限位和正限位设置可以是 0 和 360 度，或者 -180 和 +180 度。不太常见的设置可以是 360 和 720 度。所有这些设置都具有 360 度的模数范围。

可以对某个轴激活或禁用模数。

在此情况下，不回转位置读数，轴已经过的行程距离基于 0 位置来报告。也就是说，距离被指定为回归命令和 SetPosition 功能的结果。此位置测量称为无模数的位置。

模数限位和位置限制

正负模数限位独立于正负位置限制。位置限制与轴的无模数位置进行比较，以确定轴是否已到达行程终端。这些限制应该被禁用

如果为位置限制设置的值位于已激活模数的轴的模数范围内，则多轴运动控制器将阻止轴运动越过位置限制。轴在此配置中没有无限制的运动。

在某些项目中，您可能希望将位置限制设置为大于模数限位的值。例如，假设您的项目控制一个可能有一些附加电缆的轮盘。该轮盘也许只能旋转几转，电缆就严重扭曲而损坏了。在这样的项目中，您肯定希望设置位置限制，以便限制运动，从而防止电缆损坏。但是，您可能还希望将模数限位设置为一个轮盘转。例如，如果电缆在五个轮盘转之后变得紊乱，则项目会将正位置限制设置为该值（5 转），把负位置限制设置为对应值（-5 转）。模数限位可设置在 0 度和 360 度。这将阻止轴从其源位置运动 5 转以上。

位置限制和模数参数：RolloverLimit

描述

此参数用于定义轴的正模数限位和负模数限位。

轴不需要轴模数即可支持使用此功能。此功能不激活回转。（使用 EnableRollover 功能）。回转限制可在启用轴时更改，但是不能在轴运动时更改。负模数限位必须始终小于正模数限位。

关联的功能

此参数的关联功能包括：

- GetRolloverLimit，它发送针对指定轴的实际正限位和负限位值（功能参数 PARAM_CMD_1）的读取请求
- SetRolloverLimit，它发送针对指定轴的新正限位和负限位值（功能参数 PARAM_CMD_1）的写入请求

这些功能使用（根据语言对象）WRITE_CMD、READ_PARAM 或 WRITE_PARAM 指令。

语言对象

下表显示与 RolloverLimit 参数关联的不同语言对象。

对象	名称	功能代码	指令类型	描述
%MFr.m.c.45	ROLLOVER_MAX	-	READ_PARAM	读取命令
			WRITE_PARAM	写入命令
%MFr.m.c.47	ROLLOVER_MIN	-	READ_PARAM	读取命令
			WRITE_PARAM	写入命令
%MWr.m.c.26	ACTION_CMD	1539	WRITE_CMD	读取命令
		2539		写入命令
%MFr.m.c.22	RETURN_CMD_2	-	-	读取的结果。
%MDr.m.c.27	PARAM_CMD_1	-	-	要写入的值： 1：正限位 -1：负限位
%MFr.m.c.31	PARAM_CMD_3	-	-	要写入的值：模数
%MWr.m.c.19	ERROR_CMD	-	-	错误报告。（请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页）

位置限制和模数参数: EnableRollover

描述

对于给定的轴，此参数用于确定是否考虑位置值。

运动命令的位置参数依据回转位置指定轴的所需目标位置。

可以在打开跟随时更改此参数。必须将正负回转限制配置为所需的值。

语言对象

此功能使用（根据语言对象）WRITE_CMD、READ_PARAM 或 WRITE_PARAM 指令。

下表显示与 Enable_Rollover 参数关联的不同语言对象。

对象	名称	功能代码	指令类型	描述
%MWr.m.c.35.0	ENABLE_ROLLOVER	1	WRITE_PARAM	写入命令: 激活
		0		写入命令: 禁用
%MWr.m.c.26	ACTION_CMD	411	WRITE_CMD	写入命令: 激活
		412		写入命令: 禁用
%MWr.m.c.19	ERROR_CMD	-	-	错误报告。(请参见 <i>WRITE_CMD</i> 命令: 编程错误, 第 357 页)

关于正确位置带

正确位置带功能

正确位置带功能为 GetInPositionBand 和 SetInPositionBand（请参见“轴位置调整”）。正确位置带是围绕用于确定 IN_POSITION 和 AXIS_AT_TARGET 状态的目标位置的公差。如果实际位置和目标位置之间的绝对差小于正确位置带，则模块字按如下报告 IN_POSITION 和 AXIS_AT_TARGET 值：

- IN_POSITION 位在轴已在目标位置停止（STOPPING 状态位）并且轴位置在目标位置的正确位置带内时被设置。如果位置超出正确位置带，或者在下一个运动的开头，则不会将该位设置为 1。它不会在保持或快速停止期间设置为 1，而是在暂停完成并中止运动配置文件时设置。IN_POSITION 在已停止时设置为 0，在重新激活时设置为 1。
- AXIS_AT_TARGET 在轴已到达目标位置的运动配置文件结尾（PROFILE_END 状态位）并且轴位置在目标位置的正确位置带内时被设置。如果位置超出正确位置带，或者在下一个运动的开头，则不会将该位设置为 1。如果轴在目标位置停止之前，暂停、快速停止、禁用驱动器或故障中断了运动配置文件，那么即使轴位置最终在目标的正确位置带内，也不会设置该位。禁用或重新激活驱动器不会修改 AXIS_AT_TARGET 位的状态。如果设置了 AXIS_POS_LIMIT 或 AXIS_NEG_LIMIT，则不会将此位设置为 1。AXIS_AT_TARGET 与 IN_POSITION 相同，只不过：
 - 它不会在禁用驱动器时被更改
 - 它在暂停结束时设置为 0，在重新激活驱动器后也不会更改
 - 如果轴在限位处，则将它设置为 0
 - 它在 PROFILE_END 被设置后接受检查，而 IN_POSITION 在 STOPPING 被设置为 1 后接受检查

在正常的运动功能（如 Halt）之后，AXIS_AT_TARGET 可能不会设置为 1。等待 PROFILE_END 和 IN_POSITION 被设置，然后作为单独指令检查 AXIS_AT_TARGET。

位置限制参数: InPositionBand

描述

此参数用于设置指定轴的正确位置带的值。
此值以 PositionUnit 给出。

关联的功能

此参数的关联功能包括:

- GetInPositionBand, 它发送针对实际位置值的读取请求
- SetInPositionBand, 它发送针对新位置值的写入请求

这些功能使用 (根据语言对象) WRITE_CMD、READ_PARAM 或 WRITE_PARAM 指令。

关联的语言对象

下表显示与 InPositionBand 参数关联的不同语言对象。

对象	名称	功能代码	指令类型	描述
%MFr.m.c.41	IN_POSITION_BAND	-	READ_PARAM	读取命令
			WRITE_PARAM	写入命令
%MWr.m.c.26	ACTION_CMD	1035	WRITE_CMD	读取命令
		2035		写入命令
%MFr.m.c.22	RETURN_CMD_2	-	-	读取的结果。
%MFr.m.c.31	PARAM_CMD_3	-	-	要写入的值
%MWr.m.c.19	ERROR_CMD	-	-	错误报告。(请参见 <i>WRITE_CMD</i> 命令: 编程错误, 第 357 页)

关于启用模式

轴启用模式

轴的启用模式通过 SetEnableMode 功能指定，它控制最初在启用驱动器时发送给它的位置命令的值。

两种启用模式（在 SETPOINT 和 CURRENT VALUE 之间选择）描述如下：

值	模式	描述
0	MESURE	从驱动器读取轴的反馈位置，并命令驱动器在启用时将伺服元件移动到该位置。这样防止轴在启用时运动。这是缺省启用模式，并且还是强烈建议使用的启用模式。
1	SETPOINT	禁用轴时的最后命令的位置用作启用驱动器时发送到驱动器的位置命令。此模式允许轴在连续启用和停用时继续伺服到相同位置。如果轴在被禁用时运动，则一经启用，它就立即锁定回先前的设置点。锁定运动的配置文件仅由驱动器中的伺服增益参数控制，该参数可能导致急剧运动。仅当能够确定禁用期间的任何运动非常小时，才应该使用此模式。EnablePositionBand（公差）用于控制被禁用的命令位置的最大允许偏差。如果轴在被禁用时的运动小于公差，则使用被禁用的命令位置并且轴将锁定回去。如果轴在被禁用时的运动大于公差，则使用轴的当前位置，并且轴将在被禁用时保持静止。一个示例情况是带有制动器的轴，该制动器只能在控制模块的电源下运动。

SETPOINT 模式中的公差

正确位置带是在定义的启用模式为 SETPOINT 启用模式时使用的公差带。如果定义的是 CURRENTVALUE 启用模式，则正确位置带没有效果。

如果将轴配置为 SETPOINT 模式，并且轴在停用后的运动距离小于正确位置带，则轴控模块将：

1. 检索在停用轴之前发出的最后一个命令位置
2. 在重新激活轴时命令轴运动到此位置

如果轴运动距离超过正确位置带，则模块将通过临时切换到 CURRENT VALUE 模式来重新激活该轴。模块将：

1. 读取轴的位置
2. 命令轴在被重新激活时保持到此位置

这样，可以防止轴在 SETPOINT 模式下重新激活时运动距离超过正确位置带。

位置限制参数: EnableMode

说明

此参数设置轴激活模式: CURRENT VALUE 或 SETPOINT。

缺省模式为 ACTUAL 模式, 并且是强烈建议的模式。



警告

应用程序的意外行为 - ENABLEMODE 行为

仅当读取启用模式描述并清楚了解轴在被禁用时的运动距离是否小于 EnablePositionBand 之后, 才应设置 COMMANDED 启用模式, 这样, 轴将在禁用前“锁定”回上一个设定点位置。应该使用足够的安全措施 (如没有手动制动释放的轴制动器以及较小的启用跟随的错误限制), 以防止在禁用时出现不希望的运动。

如果不遵守这些说明, 将会导致死亡、严重伤害或设备损坏。

关联的功能

此参数的关联功能包括:

- GetEnableMode, 它发送针对指定轴的实际激活模式的读取请求
- SetEnableMode, 它发送针对指定轴的新激活模式的写入请求

这些功能使用 WRITE_CMD 指令。

关联的语言对象

下表显示与 EnableMode 参数关联的不同语言对象。

对象	名称	功能代码	说明
%MWr.m.c.26	ACTION_CMD	1524	读取命令
		2524	写入命令
%MDr.m.c.20	RETURN_CMD_1	-	读取的结果。
%MDr.m.c.27	PARAM_CMD_1	-	要写入的值: 激活模式。 0: 测量 1: 设定点
%MWr.m.c.19	ERROR_CMD	-	错误报告。请参阅 (<i>WRITE_CMD</i> 命令: 编程错误, 第 357 页)

从轴禁用行为

概览

这一开发任务的目的是能够选择当发出实轴锁存命令时的驱动器停止行为：

- 通过立即释放 " 飞轮 " 中的扭矩（对于机械链接的轴的情况）
- 或按照紧急停止的倾斜度进行制动 (IDN3022)

因此有两种可能的解决方案：

- 配置驱动器
- 配置模块

配置驱动器的选择

此解决方案支持在系列中的所有驱动器上使用这种可能性。

配置模块的选择

根据实轴参数中的 %MWr.m.c.35 (FUNCTION_VALIDATION) 字的位 6 (_FREEWHEEL_STOP) 的值，结果为：

- 如果该位等于 0，则驱动器将在切换为飞轮之前，按照紧急停止倾斜度进行制动来停止（如果驱动器以前是用 Unilink 配置的，则 ACTFAULT=1）
- 如果该位等于 1，则驱动器将立即切换为飞轮。

注意：此选择不能通过配置或参数屏幕来访问。要激活此选择，必须使用 WRITE_PARAM，然后使用 SAVE_PARAM。

13.3 从轴组的参数

本节主题

本节描述从轴组的调整参数。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
SERCOS(r) 模块的调整屏幕描述：从轴	264
关于从轴组	265
SlaveMode 数据	268
从轴组参数：FollowerMode	269
从轴组参数：FollowerConfig	270
从轴组参数：FollowerRatio	271
关于主轴偏移	273
从轴组参数：MasterOffset	274
从轴组参数：MasterTriggerPosition	276
从轴组参数：AbsFollowerBias	278
从轴组参数：FollowerBias	280
从轴组中的停止的传播	281
紧急停止倾斜度的校正功能	282

SERCOS(r) 模块的调整屏幕描述：从轴

简介

此屏幕用于在本地或在线模式下直接修改从轴的配置参数。

注意：不能删除主轴或从轴。

注意：有关不同调整屏幕的公共部分的更多信息，请参见 *SERCOS(r)* 模块调整屏幕描述：一般信息，第 229 页。

示意图

下图显示了在实轴或虚轴中配置 SERCOS(r) 通道的调整区域。



描述

下表显示了在从轴中配置 SERCOS(r) 通道的调整区域。

元素	关联的参数	关联的位 / 字语言	描述
主轴	MASTER_CHANNEL	%MWr.m.c.35	主轴编号（1 至 16，N 为无法访问）
从轴	SLAVE_CHANNEL_Z	%MWr.m.c.36	从轴 Z 的编号
读数 / 设定点	FOLL_TYPE_Z	%MWr.m.c.37:X2	从轴 Z 的跟随类型
传动 / 凸轮	FOLL_POSITION_Z)	%MWr.m.c.37:X1	从轴 Z 比率模式或凸轮模式
编号	NUMERATOR_Z	%MFr.m.c.38	从轴 Z 分子
偏移	DENOMINATOR_Z	%MFr.m.c.40	从轴 Z 分母
启动	FOLL_START_Z	%MWr.m.c.37:X8 至 X10	启动条件
偏差	FOLL_BIAS_REMAINS_Z	%MWr.m.c.37:X7	偏差值
触发器	TRIGGER_POSITIO_Z	%MFr.m.c.42	从轴 Z 的触发器值
跟随时停止	-	%MWr.m.c.37:X3	如果链接已删除，则停止从轴 Z。
主轴 / 定义已停止	-	%MWr.m.c.37:X6	主轴由于跟随错误而停止

关于从轴组

从轴组

从轴组是允许一组轴跟随主轴运动的轴集合。存在两种跟随主轴的主要模式：

- **比率**：从轴按 " 比率 " 跟随主轴。
- **凸轮**：从轴相对于主轴的运动在凸轮配置文件中列出。

从轴组被告知哪个轴是主轴，但是主轴并未安装在从轴组中。只存在单个主轴，所有从轴都跟随它。主轴可以是任何类型的单个轴。可以更改您希望指定为主轴的轴。

希望跟随主轴的所有轴都必须安装在从轴组中。总共可以将六个轴安装到从轴组中。每个从轴可以按照它自己的比率或凸轮配置文件跟随主轴。多个从轴可共享一个凸轮配置文件。

使用从轴组

要使用从轴组，一般按如下步骤进行：

- 配置属于该从轴组的所有单个轴。
- 配置从轴组。
- 通过对从轴组发送 EnableDrive 功能启用从轴组中的所有驱动器。
- 使用 Acquire 和 FollowOn 位启用从轴跟随，或通过执行 FollowOff 和 Release 位停止从轴。

可以命令从轴在激活或停止跟随时执行运动。激活跟随后向从轴发出的任何运动命令都会导致附加运动以及从轴跟随主轴所导致的运动。

阈值位置

您也可以选择建立主轴的阈值位置。SetMasterTriggerPosition 功能设置主轴的一个特定位置，在此位置将针对指定轴触发跟随。SlaveMode 指示是否要使用主轴阈值位置。当使用此功能并且激活了跟随时，跟随实际上会在主轴到达主轴阈值位置时开始。

偏差值

当跟随发生时，偏差值指示从轴所在的位置与它应在的位置（根据凸轮或从轴比率）之间的距离。在调用 GetFollowerBias 功能之前，应确保跟随已实际发生。（不要依赖跟随已激活的事实。）对于某个从轴，其 AXIS_IS_LINKED 运动状态将在跟随发生时设置。如果在跟随未发生时调用 GetFollowerBias，则返回值将指示从轴的命令位置。

如果在主轴位于给定点时希望从轴位于某个已知点，请使用偏差。此功能对于控制从轴只是按某个比率跟随主轴速度的应用没有用处。当从轴需要运动某个已知的递增距离时（例如，“注册”运动），此功能也没有用处。

在比率模式中，在激活跟随或修改比率后，当从轴加速至跟随速度时偏差值将更改。

偏差值在跟随开始时锁存。在凸轮模式下，从轴应在的位置由当前主轴位置和凸轮配置文件中的对应条目确定。在比率模式下，使用下面的公式确定从轴应在的位置：

从轴应在位置 = 当前主轴位置 * 比率

当跟随开始时，运动控制器设置从轴当前位置与应在位置之间的距离偏差值。此偏差阻止从轴在跟随开始时跳到它应在的位置。因此，从轴的跟随运动将偏离它应在的位置，除非将偏差设置为零。要让偏差复位，必须向从轴发送一个增量运动，其运动值等于从 GetFollowerBias 功能返回的值的负值。

非零偏差

您可能希望使用非零偏差。在凸轮模式下，可以将运动偏移表中列出的值。在比率模式下，可以改变线性关系，以便主轴位置与从轴位置的线图不会通过此点：

当主轴位置 = 0 时从轴位置 = 0

也就是说，在比率模式下，对于任何特定的主轴位置，您可以控制从轴应在的位置。从轴的偏差通过向该从轴发送运动命令进行修正。

可在任何时间针对任何轴更改从轴模式。这允许您从凸轮模式更改到比率模式或反向更改，以及更改凸轮配置文件或从轴比率。

主轴偏移

有些应用程序通过 MasterOffset 功能得到了简化。请考虑这样一种情况，其中一个从轴应跟随正弦波配置文件，而另一个从轴应跟随余弦配置文件。对于此情况，包含正弦波数据的单个凸轮配置文件可由两个从轴共享，其中分别为正弦波和余弦跟随器设置 0 度和 90 度的 MasterOffset 值。

状态位

从轴的运动状态位与发送到该轴的运动命令的状态有关。例如，设置 `PROFILE_END` 位并不表示该从轴是静止的。如果主轴运动，则该从轴将运动。当向该从轴发送某个运动命令时，`PROFILE_END` 位被清除，直到该从轴本身完成此运动。其他状态位具有类似的行为，这些行为反映从轴运动命令的状态而不是主轴的状态。

SlaveMode 数据

描述

在从轴组中，存在两种跟随主轴的主要方法：

- 比率：从轴使用比率跟随主轴。
- 凸轮系统：从轴相对于主轴的运动由凸轮配置文件定义。

一旦所有轴都已安装到从轴组中，则通过指定要应用于每个轴的 SlaveMode 类型来为该组的轴配置跟随。下表列出了 SlaveMode 条件。位 0 是最低有效位 (LSB)。

SlaveMode 数据

下表列出了各种数据值：

名称	位	描述
FOLL_WHERE	0	值 = 0：必需
FOLL_TYPE	1	确定跟随类型： 0 = 比率 1 = 凸轮
FOLL_POSITION	2	确定要跟随的位置： 0 = 当前值 1 = 设置点
FOLL_FOL_ON_HALT	3	确定要在打开跟随时执行的动作： 0 = 不暂停从轴上任何正在进行的运动配置文件 1 = 暂停从轴上任何正在进行的运动配置文件
保留	4, 5	
FOLL_HALT_MASTER	6	确定打开跟随后，主轴是否应该在从轴发生故障时暂停： 0 = 主轴保持运动 1 = 暂停主轴
FOLL_BIAS_REMAINS	7	确定要在关闭跟随时执行的动作： 0 = 暂停从轴上的任何附加运动配置文件 1 = 不暂停从轴上的任何附加运动配置文件
FOLL_START	8... 10	确定启动条件： 0 = 立即 1 = 当主轴位置超过正向的主轴触发位置时启动 2 = 当主轴位置超过负向的主轴触发位置时启动 3 = 当主轴位置大于或等于主轴触发位置时启动 4 = 当主轴位置小于或等于主轴触发位置时启动

从轴组参数: FollowerMode

描述

此参数包含指定的从轴正在进行的跟随模式。

关联的功能

与此参数关联的功能为：

- GetFollowerMode，它发送针对实际跟随模式的读取请求

此功能使用 WRITE_CMD 指令。

语言对象

下表显示与 FollowerMode 参数关联的不同语言对象。

对象	名称	功能代码	指令类型	描述
%MWr.m.c.26	ACTION_CMD	1529	WRITE_CMD	读取命令
%MDr.m.c.20	RETURN_CMD_1	-	-	读取的结果。
%Mr.m.c.i.27	PARAM_CMD_1	-	-	要写入的值：从轴组中从轴的标识符 (1)。
%MWr.m.c.19	ERROR_CMD	-	-	错误报告。（请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页）
说明				
(1)	此标识符包含在常量字 %KWr.m.c.0 中			

从轴组参数: FollowerConfig

描述

此功能用于设置指定轴的跟随器模式和对应的从轴配置文件（跟随器比率或凸轮配置文件）。

注意：凸轮配置文件必须在被引用之前配置。

注意：可以在打开跟随时更改此参数。但是，如果在凸轮模式下或在切换到凸轮模式时激活跟随，则当新的运动配置文件与旧的配置文件显著不同时，从轴的速度可能会剧烈更改。

关联的功能

与此参数关联的功能为：

- SetFollowerConfig，它发送针对新的跟随模式和运动配置文件的写入请求
此功能使用 WRITE_CMD 指令。

关联的语言对象

下表显示与 FollowerConfig 参数关联的不同语言对象。

对象	名称	功能代码	指令类型	描述
%MWr.m.c.26	ACTION_CMD	420	WRITE_CMD	写入命令
%MDr.m.c.27	PARAM_CMD_1	-	-	要写入的值：从轴组中从轴的标识符 (1)。
%MDr.m.c.29	PARAM_CMD_2	-	-	要写入的值：跟随器模式。请参见 <i>SlaveMode</i> 数据，第 268 页
%MFr.m.c.31	PARAM_CMD_3	-	-	要写入的值： 用于比率模式的分子 用于凸轮模式的运动配置文件
%MFr.m.c.33	PARAM_CMD_4	-	-	要写入的值：用于比率模式的分母
%MWr.m.c.19	ERROR_CMD	-	-	错误报告。（请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页）
说明				
(1)	此标识符包含在常量字 %KWr.m.c.0 中			

从轴组参数: FollowerRatio

描述

此参数用于更改指定从轴的比率。

比率分子指示在主轴运动由分母所示的距离时，从轴将运动多长的距离。比率值可在任何时候修改。

该比率不会即刻更改，以便在激活从轴和发生跟随运动时实现平滑的运动。

关联的功能

此参数的关联功能包括：

- GetFollowerRatio，它发送针对指定轴的实际位置公差的读取请求
 - SetFollowerRatio，它发送针对指定轴的实际位置公差的写入请求
- 这些功能使用（根据语言对象）WRITE_CMD、READ_PARAM 或 WRITE_PARAM 指令。

语言对象

下表显示与 FollowerRatio 参数关联的不同语言对象。

对象	名称	功能代码	指令类型	描述
%MFr.m.c.38	NUMERATOR_1	-	READ_PARAM	从轴 1 读取命令
			WRITE_PARAM	从轴 1 写入命令
%MFr.m.c.40	DENOMINATOR_1	-	READ_PARAM	从轴 1 读取命令
			WRITE_PARAM	从轴 1 写入命令
%MFr.m.c.46	NUMERATOR_2	-	READ_PARAM	从轴 2 读取命令
			WRITE_PARAM	从轴 2 写入命令
%MFr.m.c.48	DENOMINATOR_2	-	READ_PARAM	从轴 2 读取命令
			WRITE_PARAM	从轴 2 写入命令
%MFr.m.c.54	NUMERATOR_3	-	READ_PARAM	从轴 3 读取命令
			WRITE_PARAM	从轴 3 写入命令
%MFr.m.c.56	DENOMINATOR_3	-	READ_PARAM	从轴 3 读取命令
			WRITE_PARAM	从轴 3 写入命令
%MFr.m.c.62	NUMERATOR_4	-	READ_PARAM	从轴 4 读取命令
			WRITE_PARAM	从轴 4 写入命令
%MFr.m.c.64	DENOMINATOR_4	-	READ_PARAM	从轴 4 读取命令
			WRITE_PARAM	从轴 4 写入命令

对象	名称	功能代码	指令类型	描述
%MFr.m.c.70	NUMERATOR_5	-	READ_PARAM	从轴 5 读取命令
			WRITE_PARAM	从轴 5 写入命令
%MFr.m.c.72	DENOMINATOR_5	-	READ_PARAM	从轴 5 读取命令
			WRITE_PARAM	从轴 5 写入命令
%MFr.m.c.78	NUMERATOR_6	-	READ_PARAM	从轴 6 读取命令
			WRITE_PARAM	从轴 6 写入命令
%MFr.m.c.80	DENOMINATOR_6	-	READ_PARAM	从轴 6 读取命令
			WRITE_PARAM	从轴 6 写入命令
%MWr.m.c.26	ACTION_CMD	1114	WRITE_CMD	读取命令
		2114		写入命令
%MFr.m.c.22	RETURN_CMD_2	-	-	读取分子变量的结果。
%MFr.m.c.24	RETURN_CMD_3	-	-	读取分母变量的结果。
%MDr.m.c.27	PARAM_CMD_1	-	-	要写入的值：要发回其比率的轴的标识符。
%MFr.m.c.31	PARAM_CMD_3	-	-	要写入的值：比率分子。
%MFr.m.c.33	PARAM_CMD_4	-	-	要写入的值：比率分母。
%MWr.m.c.19	ERROR_CMD	-	-	错误报告。（请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页）

关于主轴偏移

主轴偏移的值

FollowerAxis 跟随主轴位置与 MasterOffset 值之和。

通常，MasterOffset 值在关闭跟随时更改。MasterOffset 可在打开跟随时更改，但是要小心避免不希望的运动。MasterOffset 立即应用，无需任何混合。打开跟随时，MasterOffset 中的较大更改会导致跟随器快速运动到新的跟随位置。

主轴偏移功能

MasterOffset 可选功能：GetMasterOffset 和 SetMasterOffset 可简化某些应用。请考虑这样一种情况，其中一个跟随器轴应跟随正弦波配置文件，而另一个跟随器轴应跟随余弦配置文件。对于此情况，包含正弦波数据的单个凸轮配置文件可由两个跟随器轴共享，其中分别为正弦波和余弦跟随器设置 0 度和 90 度的 MasterOffset 值。

从轴组参数: MasterOffset

描述

此参数用于定义指定从轴的主轴偏移值。

每个从轴可以有一个应用于主轴位置的唯一偏移。当您希望将单个凸轮配置文件用于多个跟随器，但每个跟随器的凸轮形状要相对于另一个跟随器的凸轮旋转一定角度时，这一点很有帮助。

注意：MasterOffset 可在打开跟随时更改，但是要小心避免不希望的运动。应该将较大的更改作为一系列较小的增量混合进来，以避免跟随器位置的立即更改。

关联的功能

此参数的关联功能包括：

- GetMasterOffset，它发送针对指定轴的实际主轴偏移值的读取请求
- SetMasterOffset，它发送针对指定轴的新主轴偏移值的写入请求

这些功能使用（根据语言对象）WRITE_CMD、READ_PARAM 或 WRITE_PARAM 指令。

关联的语言对象

下表显示与 MasterOffset 参数关联的不同语言对象。

对象	名称	功能代码	指令类型	描述
%MFr.m.c.40	DENOMINATOR_1	-	READ_PARAM	从轴 1 读取命令
			WRITE_PARAM	从轴 1 写入命令
%MFr.m.c.48	DENOMINATOR_2	-	READ_PARAM	从轴 2 读取命令
			WRITE_PARAM	从轴 2 写入命令
%MFr.m.c.56	DENOMINATOR_3	-	READ_PARAM	从轴 3 读取命令
			WRITE_PARAM	从轴 3 写入命令
%MFr.m.c.64	DENOMINATOR_4	-	READ_PARAM	从轴 4 读取命令
			WRITE_PARAM	从轴 4 写入命令
%MFr.m.c.72	DENOMINATOR_5	-	READ_PARAM	从轴 5 读取命令
			WRITE_PARAM	从轴 5 写入命令
%MFr.m.c.80	DENOMINATOR_6	-	READ_PARAM	从轴 6 读取命令
			WRITE_PARAM	从轴 6 写入命令
%MWr.m.c.26	ACTION_CMD	1532	WRITE_CMD	读取命令
		2532		写入命令
%MFr.m.c.22	RETURN_CMD_2	-	-	读取的结果：位置变量。
%MDr.m.c.27	PARAM_CMD_1	-	-	要写入的值：从轴组中从轴的标识符 (1)。
%MFr.m.c.31	PARAM_CMD_3	-	-	要写入的值：位置。
%MWr.m.c.19	ERROR_CMD	-	-	错误报告。(请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页)
说明				
(1)	此标识符包含在常量字 %KWr.m.c.0 中			

从轴组参数: MasterTriggerPosition

描述

此参数用于设置主轴的一个特定位置，在此位置将对指定轴触发跟随。
仅当跟随器模式指示跟随启动条件应该使用主轴触发位置时，才会使用此值。

关联的功能

此参数的关联功能包括：

- GetMasterTriggerPosition，它发送针对实际主轴位置值的读取请求
- SetTriggerPosition，它发送针对新主轴位置值的写入请求

这些功能使用（根据语言对象）WRITE_CMD、READ_PARAM 或 WRITE_PARAM 指令。

关联的语言对象

下表显示与 MasterTriggerPosition 参数关联的不同语言对象。

对象	名称	功能代码	指令类型	描述
%MFr.m.c.42	TRIGGER_POSITION_1	-	READ_PARAM	从轴 1 读取命令
			WRITE_PARAM	从轴 1 写入命令
%MFr.m.c.50	TRIGGER_POSITION_2	-	READ_PARAM	从轴 2 读取命令
			WRITE_PARAM	从轴 2 写入命令
%MFr.m.c.58	TRIGGER_POSITION_3	-	READ_PARAM	从轴 3 读取命令
			WRITE_PARAM	从轴 3 写入命令
%MFr.m.c.66	TRIGGER_POSITION_4	-	READ_PARAM	从轴 4 读取命令
			WRITE_PARAM	从轴 4 写入命令
%MFr.m.c.74	TRIGGER_POSITION_5	-	READ_PARAM	从轴 5 读取命令
			WRITE_PARAM	从轴 5 写入命令
%MFr.m.c.82	TRIGGER_POSITION_6	-	READ_PARAM	从轴 6 读取命令
			WRITE_PARAM	从轴 6 写入命令
%MWr.m.c.26	ACTION_CMD	1531	WRITE_CMD	读取命令
		2531		写入命令

对象	名称	功能代码	指令类型	描述
%MFr.m.c.22	RETURN_CMD_2	-	-	读取的结果：位置变量。
%MDr.m.c.27	PARAM_CMD_1	-	-	要写入的值：从轴组中从轴的标识符 (1)。
%MFr.m.c.31	PARAM_CMD_3	-	-	要写入的值：位置。
%MWr.m.c.19	ERROR_CMD	-	-	错误报告。(请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页)
说明				
(1)	此标识符包含在常量字 %KWr.m.c.0 中			

从轴组参数: AbsFollowerBias

描述

此参数用于停止跟随，并在不停止从轴的情况下将从轴移动到某个绝对位置。

绝对偏差

绝对偏差是由 GetFollowerBias 功能返回的偏差与从轴的附加命令位置之间的差。绝对偏差可以与绝对运动命令 (ABS_MOVE) 结合使用，用于将跟随器轴定位到某个可计算的位置。

从轴的命令位置

从轴的命令位置是跟随主轴的命令位置和偏差之和：

$Command_pos = SlaveMethod (master_pos + master_offset) + 偏差;$

其中 FollowerMethod 表示：

- **比率**：将主轴参数乘以 FollowerRatio
- **凸轮**：按给定的主轴参数查找凸轮配置文件

偏差

偏差也包含两部分：

$偏差 = additive_pos_command + abs_bias;$

其中：

- **additive_pos_command** 是由发送到跟随器轴本身的任何运动命令生成的命令位置。
- **abs_bias** 是每当跟随器模式被更改（如激活从轴，或更改凸轮配置文件或跟随器比率）时锁存的值。计算锁存值是为了避免在使用新模式时以任何形式直接跳到命令的位置。此锁存值称为 "绝对偏差"。

使用绝对偏差

绝对偏差可用于将跟随器轴定位到某个已知的绝对位置（如果来自 FollowerMethod 功能的值已知并且是常量）。可以使用绝对偏差的一种情况是在传动期间用于停止跟随主轴，但是要保持跟随器轴以连续速度运动。将 FOLL_BIAS_REMAINS 跟随器模式位设置为 1，并使用 SetFollowerConfig 功能配置该模式。停止跟随，并立即发出以所需速度沿所需方向连续立即运动的命令。从轴比率以减速率减速至 0，同时立即运动以加速率加速至所需速度。这两个斜坡将彼此同时进行，在两个斜坡完成时轴将以所需速度运动（跟随器轴的 RAMPING 位被清除）。在停用 FOLL_BIAS_REMAINS 和跟随的比率模式中，位置命令计算的 SlaveMethod 功能部分为 0，从而得出：

$$\text{pos_command} = \text{additive_pos_command} + \text{abs_bias};$$

现在，可以通过发出绝对运动命令并在绝对偏差的补偿下将跟随器轴运动至某个绝对目标，或者：

```
foll_axis.moveImmed (ABS_MOVE, target - abs_bias, speed);
```

一旦此运动的配置文件完成，则向跟随器发出暂停或禁用驱动器命令，然后可以发出正常运动命令，而不受偏差值的影响。

返回的值为绝对偏差。如果激活了模数功能，则返回带模数的绝对偏差。

关联的功能

与此参数关联的功能为：

- GetFollowerMode，它发送针对实际跟随模式的读取请求

此功能使用 WRITE_CMD 指令。

关联的语言对象

下表显示与 AbsFollowerBias 参数关联的不同语言对象。

对象	名称	功能代码	指令类型	描述
%MWr.m.c.26	ACTION_CMD	1526	WRITE_CMD	读取命令
%MFr.m.c.22	RETURN_CMD_2	-	-	读取的结果。
%MDr.m.c.27	PARAM_CMD_1	-	-	要写入的值：从轴组中从轴的标识符 (1)。
%MWr.m.c.19	ERROR_CMD	-	-	错误报告。（请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页）
说明				
(1)	此标识符包含在常量字 %KWr.m.c.0 中			

从轴组参数: FollowerBias

描述

此参数包含偏差值。
如果启用了回转，则会将偏差作为回转值返回。

关联的功能

与此参数关联的功能为：

- GetFollowerMode，它发送针对实际跟随模式的读取请求
此功能使用 WRITE_CMD 指令。

关联的语言对象

下表显示与 FollowerBias 参数关联的不同语言对象。

对象	名称	功能代码	指令类型	描述
%MWr.m.c.26	ACTION_CMD	1527	WRITE_CMD	读取命令
%MFr.m.c.22	RETURN_CMD_2	-	-	读取的结果。
%MDr.m.c.27	PARAM_CMD_1	-	-	要写入的值：从轴组中从轴的标识符 (1)。
%MWr.m.c.19	ERROR_CMD	-	-	错误报告。（请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页）
说明				
(1)	此标识符包含在常量字 %KWr.m.c.0 中			

从轴组中的停止的传播

主轴停止

选择存在从轴故障时的主轴停止类型：

选择列表：

- 无主轴停止（缺省值）
- 根据已配置的停止倾斜度正常停止主轴
- 通过锁存命令停止主轴

注意：此选择不能通过配置或参数屏幕来访问。要激活此选择，必须使用 WRITE_PARAM 或 SAVE_PARAM。

描述：

管理位：%MWr.m.c.37:X11 (FOLL_FAULT_MASTER_x)。

- 位 11 的值为 1 且位 6 设置为 0 将触发无主轴停止
- 从轴组参数中的 FOLLOWER_MODE 字的位 11 (FOLL_FAULT_MASTER) 和位 6 (FOLL_HALT_MASTER) 的值为 1 意味着主轴将在发生从轴故障时按照紧急停止倾斜度停止。
- 位 11 的值为 0 且位 6 设置为 1 将触发主轴正常停止（等同于 STOP）

从轴停止

选择存在主轴故障时的从轴停止：

- 在主轴序列上停止（缺省值）
- 通过锁存命令停止从轴

描述：

管理位：%MWr.m.c.37:X12 (FOLL_FAULT_SLAVE_x)。

从轴组参数中的 FOLLOWER_MODE 字的位 12 (FOLL_FAULT_SLAVE) 的值为 1 意味着从轴将在发生主轴故障时按照紧急停止倾斜度停止。

（Lexium 上的 DECSTOP 减速斜坡可使用 Unilink、IDN P 3022 或 ASCII 码进行配置）。

注意：这些位的动作仅在写入组配置 %Kx 常量字后才进行设置。为此，应对组执行 MOD-PARAM。

紧急停止倾斜度的校正功能

概览

此功能管理 Lexium 驱动器中的 IDN3022（或在运行 Unilink 时为 DECSTOP 参数）的计算和修改。

它允许配置从轴组中的从轴驱动器的紧急停止倾斜度，以便在发生故障时所有轴都同时停止。

为此，需要创建一个可在从轴组上使用的新操作码 (570)。此命令将导致按照已配置的倾斜度**计算主轴和从轴减速斜坡**，然后将数据发送到组中的所有实轴驱动器（包括主轴）。

此功能的参数是以在组主轴配置期间选择的单位表示的**减速倾斜度**。如果主轴是外部设定点，我们会将单位选择为位置单位 / 平方秒。 s^2

对象	类型	符号	说明	注释
%MW.r.m.c.19	字	ERROR_CMD		错误代码
%MDr.m.c.20	双字	RETURN_CMD_1		未使用
%MFr.m.c.22	浮点数	RETURN_CMD_2		未使用
%MFr.m.c.24	浮点数	RETURN_CMD_3		未使用
%MW.r.m.c.26	字	ACTION_CMD	570	ADJUST_3022
%MDr.m.c.27	双字	PARAM_CMD_1	0	未使用
%MDr.m.c.29	双字	PARAM_CMD_2	0	未使用
%MFr.m.c.31	浮点数	PARAM_CMD_3	减速度	
%MFr.m.c.33	浮点数	PARAM_CMD_4	0	未使用

示例

```
IF NOT %MW104.21.0:X1 THEN (* WRITE_CMD 未正在进行 *)
%MW104.21.26:=570;          (* 动作 570 = ADJUST_3022 *)
%MF104.21.31:=1000.0;      (* 主轴减速度 = 1000 转 /
分钟 / 秒 *)
WRITE_CMD %CH 104.21;
END_IF;
```

命令被拒绝:

- 从轴组是包含凸轮配置文件型从轴的从轴组 => **错误代码 62**
- 从轴或主轴的减速度超过在配置期间输入的最大减速度 => **错误代码 63**。注: 在请求零倾斜度时也会出现此错误代码。
- 为轴计算的 IDN"DECSTOP" 被驱动器拒绝 (参数超出范围) => **错误代码 1002**

注意: 仅当驱动器为 Lexium 驱动器 ≥ 5.50 时, 此功能才有效。它只能由其中的所有从轴都按比率配置的组访问。倾斜度的调整不保存, 可能会在停电或驱动器复位时丢失。

13.4 凸轮配置文件调整

本节主题

本节描述用于凸轮配置文件的调整功能。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
关于凸轮配置文件	285
凸轮配置文件参数：长度	286
凸轮配置文件参数：LookUpFollowerPosition 功能	287
凸轮配置文件参数：Coord	288
凸轮配置文件参数：InterpType	290
凸轮配置文件参数：ProfileId	291
凸轮配置文件的 TRF_RECIPE 屏幕描述	292
TRF_RECIPE 指令	294

关于凸轮配置文件

简介

凸轮配置文件与给定主轴位置时的从轴位置相关。凸轮配置文件是主轴和从轴位置坐标表。该表中未显式列出的位置点通过在给定点之间内插来导出。内插可以是线性的或立方的。

主轴位置和增量

主轴位置必须在该表中以递增顺序（从配置文件中的第一个元素到最后一个元素）排列。例如，主轴位置不能越过 360 度到 0 度，因为这些主轴位置不是递增的。

主轴位置之间的增量可以是可变的。增量不必是固定的。例如，以下是可接受的主轴位置值：0、1、2、100、300、360。可以使用任何有效的位置单位。

典型应用

需要平滑运动的应用，其中运动从一个迭代通过配置文件进行到下一个迭代。为了使这种转换尽可能平滑，第一个数据点的从轴位置值应该与最后一个数据点的从轴位置值相同。例如，如果主轴位置以度为单位，从轴位置以英寸为单位，并且主轴的第一个位置在 0 度，从轴在 0 英寸，则对应的最后一个位置将是主轴在 360 度，从轴在 0 英寸。

建立凸轮配置文件

凸轮配置文件表中的每个点都有一个表示凸轮配置文件的两个坐标轴的位置值。这意味着凸轮配置文件的每个点有两个值：一个表示主轴位置，一个表示从轴位置。SetCoord 功能同时更改凸轮配置文件中现有点的这两个值。GetCoord 功能同时返回凸轮配置文件中指定点的这两个值。

修改内插类型

在凸轮配置文件点表中，不在这些表中的位置可通过在给定点之间内插来导出。支持线性或立方内插。通过 GetInterpType 和 SetInterpType 功能来支持对内插类型进行修改。

凸轮配置文件参数：长度

描述

此参数包含为配置文件指定的点数。

该点数根据配置文件配置而异。

下表介绍了不同的配置可能性。

配置 ...	则 ...
来自文件	自动确定点数
来自保持寄存器	这些寄存器之一标识长度
无数据并且应由 SetCoord 功能定义	此功能调用包含点数

关联的功能

与此参数关联的功能为：

- GetLength，它发送针对配置文件中的点数的读取请求

此功能使用 WRITE_CMD 指令。

关联的语言对象

下表显示与 Length 参数关联的不同语言对象。

对象	名称	功能代码	指令类型	描述
%MWr.m.c.26	ACTION_CMD	534	WRITE_CMD	读取命令
%MDr.m.c.20	RETURN_CMD_2	-	-	读取的结果。
%MWr.m.c.19	ERROR_CMD	-	-	错误报告。（请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页）

凸轮配置文件参数: LookUpFollowerPosition 功能

描述

此功能包含配置文件从轴的位置，用以提供指定的主轴位置。
从轴位置通过使用为该配置文件配置的内插类型来导出。

关联的功能

关联的功能为：

- GetLookUpFollowerPosition，它发送针对配置文件从轴位置的读取请求
此功能使用 WRITE_CMD 指令。

语言对象

下表显示与 Length 参数关联的不同语言对象。

对象	名称	功能代码	指令类型	描述
%MWr.m.c.26	ACTION_CMD	537	WRITE_CMD	读取命令
%MFr.m.c.22	RETURN_CMD_2	-	-	读取的结果：从轴位置。
%MFr.m.c.31	PARAM_CMD_3	-	-	要写入的值：主轴位置
%MWr.m.c.19	ERROR_CMD	-	-	错误报告。（请参见 <i>WRITE_CMD</i> 命令： 编程错误，第 357 页）

凸轮配置文件参数: Coord

说明

此功能用于修改现有凸轮配置文件点的坐标。

点索引从零开始。因此，要修改凸轮配置文件中的第一个条目，应使用索引值 0。

凸轮配置文件被指定为坐标位置点的表。这些坐标使用 PosCoord 数据结构进行访问。每个点都有一个用于配置文件的两个坐标轴的位置值。这意味着凸轮配置文件的每个点有两个值：一个表示主轴位置，一个表示从轴位置。

主轴位置必须从凸轮配置文件中的第一个元素到最后一个元素，处于递增顺序。例如，主轴位置不能越过 360 度到 0 度，因为这些主轴位置不是递增的。

主轴位置值之间的增量可以是可变的。增量不必是固定的。例如，以下是可接受的主轴位置值：0、1、2、100、300、360。可以使用任何有效的位置单位。



小心

意外的设备行为

在使用凸轮配置文件的跟随器组中，当打开了跟随时，不要修改配置文件，不要调用功能 Configure、SetCoord 或 SetInterpType。

如果不遵守这些说明，将会导致受伤或设备损坏。

注意：在某些特殊情况下，可以在打开跟随时调用 SetCoord 功能更改配置文件。但是，您只能更改与从轴正在使用的最近配置文件点相隔两个配置文件点条目的配置文件点。未能确保此限制将导致不希望的从轴运动。

关联的功能

此参数的关联功能包括：

- GetCoord，它发送针对现有凸轮配置文件点的实际坐标的读取请求
- SetCoord，它发送针对现有凸轮配置文件点的新坐标的写入请求

这些功能使用 WRITE_CMD 指令。

语言对象

下表介绍了不同的功能。

对象	名称	功能代码	指令类型	说明
%MWr.m.c.26	ACTION_CMD	532	WRITE_CMD	读取命令
		533		写入命令
%MFr.m.c.22	RETURN_CMD_2	-	-	读取的结果：凸轮配置文件表中的主轴位置点。
%MFr.m.c.24	RETURN_CMD_3	-	-	读取的结果：凸轮配置文件表中的从轴位置点。
%MDr.m.c.27	PARAM_CMD_1	-	-	要写入的值：必需的点索引。
%MFr.m.c.31	PARAM_CMD_3			要写入的值：主轴位置
%MFr.m.c.33	PARAM_CMD_4			要写入的值：从轴位置
%MWr.m.c.19	ERROR_CMD	-	-	错误报告。请参阅 (<i>WRITE_CMD</i> 命令：编程错误，第 357 页)

凸轮配置文件参数: InterpType

说明

此参数用于定义内插样式，该内插样式用于生成不必在配置文件表中起作用的位置。

立方内插算法迫使该表中的点的一阶导数成为连续的，使得表中任何点的一阶导数与上一点和下一点之间的线段的斜率匹配。



意外的设备行为

在使用凸轮配置文件的 FollowerSet 中，当打开了跟随时，不要修改配置文件，也不要调用功能 Configure、SetCoord 或 SetInterpType。

如果不遵守这些说明，将会导致受伤或设备损坏。

关联的功能

此参数的关联功能包括：

- GetInterpType，它发送针对实际内插样式的读取请求
- SetInterpType，它发送针对新内插样式的写入请求

这些功能使用 WRITE_CMD 指令。

关联的语言对象

下表显示与 InterpType 参数关联的不同语言对象。

对象	名称	功能代码	指令类型	说明
%MWr.m.c.26	ACTION_CMD	530	WRITE_CMD	读取命令
		531		写入命令
%MDr.m.c.20	RETURN_CMD_1	-	-	读取的结果。
%MDr.m.c.27	PARAM_CMD_1	-	-	要写入的值：内插类型 0: 线性 1: 立方
%MWr.m.c.19	ERROR_CMD	-	-	错误报告。请参阅 (<i>WRITE_CMD</i> 命令： 编程错误，第 357 页)

凸轮配置文件参数: ProfileId

描述

此参数包含配置文件标识。

关联的功能

与此参数关联的功能为:

- GetProfileId, 它发送针对配置文件的从轴跟随器位置的读取请求
此功能使用 WRITE_CMD 指令。

语言对象

下表显示与 ProfileId 参数关联的不同语言对象。

对象	名称	功能代码	指令类型	描述
%MWr.m.c.26	ACTION_CMD	528	WRITE_CMD	读取命令
%MDr.m.c.27	RETURN_CMD_1	-	-	读取的结果
%MWr.m.c.19	ERROR_CMD	-	-	错误报告。(请参见 <i>WRITE_CMD</i> 命令: 编程错误, 第 357 页)

凸轮配置文件的 TRF_RECIPE 屏幕描述

简介

此屏幕位于凸轮配置文件调整屏幕的底部，用于在 PLC 与伺服驱动器之间传输调整参数。

此功能必须在在线模式下使用。

注意：有关不同调整屏幕的公共部分的更多信息，请参见 *SERCOS(r)* 模块调整屏幕描述：一般信息，第 229 页。

示意图

下图显示了与 TRF_RECIPE 功能关联的屏幕。

描述

下表显示了 TRF_RECIPE 区域的**传输方向**。

元素	关联的语言字	符号 / 描述
读 / 写	%MWr.m.c.10	请参见 TRF_RECIPE 选择要传输的表
" 主轴 " 表		
" 从轴 " 表		

下表显示了 TRF_RECIPe 区域的**参数**。

元素	关联的语言字	描述
地址表	%MDr.m.c.11	可以修改两个参数： 用于存储数据的对象存储器的类型： <ul style="list-style-type: none"> ● 读写状态的 %MW ● 只读的 %KW
	立即值	表的开始地址
长度	立即值	表长度

下表显示了 TRF_RECIPe 区域的**结果**。

元素	关联的语言字	描述
传输错误	%MWr.m.c.3	此字段在传输无效的情况下提供错误代码 (参见第 363 页)。
读取的表长度	%MDr.m.c.4	表的实际长度

TRF_RECIPE 指令

提示

此指令使您能够读取或写入凸轮配置文件以及开始特殊功能的执行。

注：给定的 TRF_RECIPE 参数长度定义了所交换的任何类型的信息的字数（双精度或浮点）。

注意：此服务还用于读取或写入 " 实轴 " 功能的变速控制器参数（请参见 *TRF_RECIPE*, 第 238 页）。

注意：参数交换（READ_PARAM、WRITE_PARAM、SAVE_PARAM、RESTORE_PARAM）不能与 TRF_RECIPE 指令同时执行。

TRF_RECIPE 指令的语法

TRF_RECIPE (IODDT_VAR, 长度, 地址 %MW): 从 %MW 字的表中传输凸轮配置文件参数。要传输的表长度由 length 参数定义。要执行的动作由字 %MW_r.m.c.10 (ACTION_TRF) 定义。

示例：TRF_RECIPE (IODDT_VAR, 100,200) 读取位于机架 1 的位置 4 中模块的凸轮配置文件 25 的 " 从轴 " 表，然后将该表传输到 PLC 中从地址 200 开始的位置。对应于声明为 T_CSY_CAM 类型并与机架 1 中模块 4 的通道 25 关联的 IODDT_VAR 变量。

TRF_RECIPE 接口

要执行的命令在字 %MW_r.m.c.10 中定义，命令的结果在字 %MW_r.m.c.3 至 %MW_r.m.c.8 中提供。

地址	类型	符号	含义
%MW _r .m.c.10	字	ACTION_TRF	要执行的动作
%MW _r .m.c.3	字	ERROR_TRF	TRF_RECIPE 命令的写入错误
%MD _r .m.c.4	双字	RETURN_TRF_1	功能的返回值 1
%MFr.m.c.6	浮点数	RETURN_TRF_2	功能的返回值 2
%MFr.m.c.8	浮点数	RETURN_TRF_3	功能的返回值 3
%MD _r .m.c.11	双字	PARAM_TRF_1	参数 1
%MD _r .m.c.13	双字	PARAM_TRF_2	参数 2
%MFr.m.c.15	浮点数	PARAM_TRF_3	参数 3
%MFr.m.c.17	浮点数	PARAM_TRF_4	参数 4

TRF_RECIPE 执行的动作

可使用 TRF_RECIPE 服务执行的动作为：

功能	ACTION_TRF (%MWr.m.c.10)	含义
凸轮配置文件	1	读取 " 主轴 " 表。
凸轮配置文件	2	读取 " 从轴 " 表。
凸轮配置文件	3	读取 " 主轴 " 和 " 从轴 " 表。
凸轮配置文件	101	写入 " 主轴 " 表。
凸轮配置文件	102	写入 " 从轴 " 表。
凸轮配置文件	103	写入 " 主轴 " 和 " 从轴 " 表。

说明

将驱动器写入模块	PARAM_TRF_1 = 0. On MW
将模块写入驱动器	PARAM_TRF_1 = 1. On KW
读取	PARAM_TRF_1 = 0.
	PARAM_TRF_2 = OFFSET
	PARAM_TRF_3 和 PARAM_TRF_4 = 0
	RETURN_TRF_1(%MDr.m.c.4) = 读取或写入的表长度
	RETURN_TRF_2 和 RETURN_TRF_3 = 0

监控交换

以下两个位可用于监控在 TRF_RECIPE 指令的帮助下执行的交换：

位	含义
%MWr.m.c.0.3	交换正在进行。
%MWr.m.c.1.3	交换不正确。

13.5 SERCOS(r) 环路参数

本节主题

本节介绍 SERCOS(r) 环路的调整屏幕并描述关联的调整参数。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
SERCOS(r) 模块的调整屏幕描述：SERCOS(r) 环路	297
光功率调整图表	299
SERCOS(r) 环路参数：OpticalPower	300

SERCOS(r) 模块的调整屏幕描述：SERCOS(r) 环路

简介

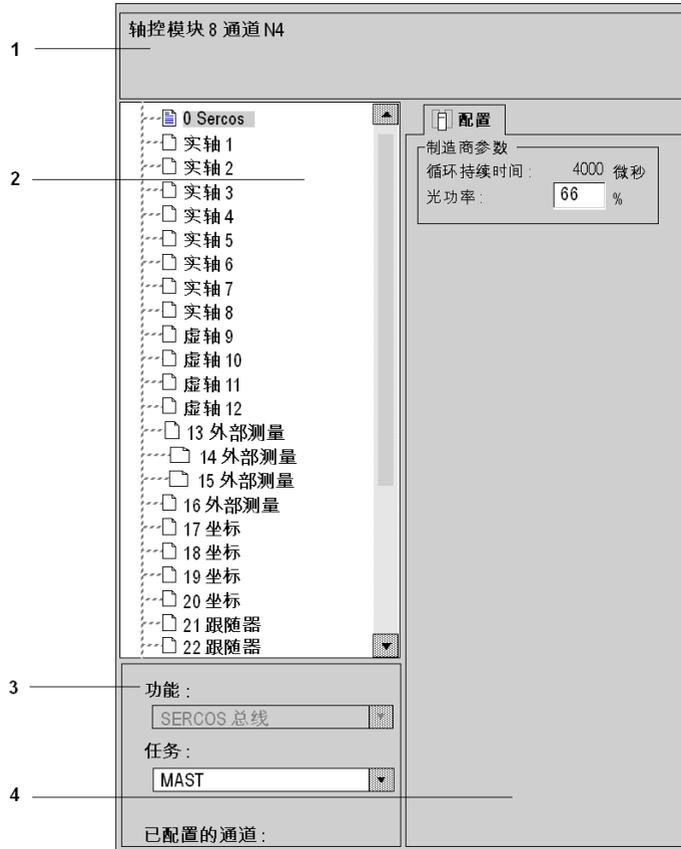
此屏幕用于在离线或在线模式下直接修改光功率的值。

它取消当前环路循环时间的值。

注意：有关不同调整屏幕的公共部分的更多信息，请参见 *SERCOS(r)* 模块调整屏幕描述：一般信息，第 229 页。

示意图

下图显示了 TSX CSY 84 模块上的 SERCOS(r) 环路的调整区域（该配置与 TSX CSY 164 模块上的配置相同）：



描述**屏幕描述**

字段	描述
1	此区给出了模块及其在 PLC 中的地理地址的目录参考（机架编号和在机架中的位置）。
2	此区域名为 通道区域 ，用于选择要配置的通道。
3	此区域名为 常规参数区域 ，用于配置与所选通道关联的常规参数。
4	此区域取决于所选择的选项卡：在此例中，它是在区域 2 中选择的通道的配置区域。在本地模式下只提供配置选项卡。

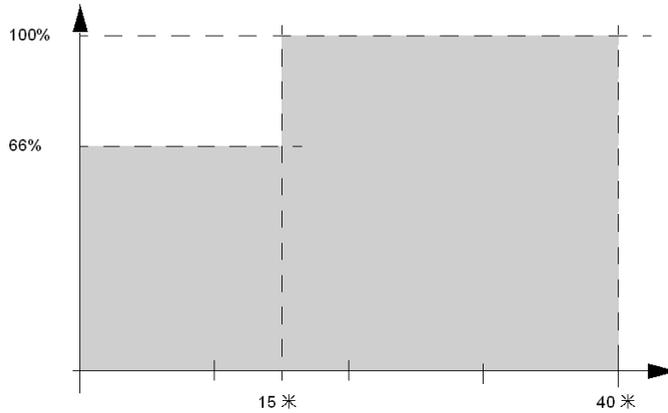
配置选项卡描述

元素	关联的参数	关联的位 / 字语言	符号 / 描述
循环时间	-	-	当前环路循环时间的值。
光功率	-	MWr.m.c.37	光功率调整

光功率调整图表

光功率

TSX CSY 84 / 164 模块的光功率以百分比给出，它是第一段（在 TSX CSY 84 模块或 TSX CSY 164 模块和第一个驱动器之间）的长度的函数。光功率的配置或调整必须符合以下图表：



SERCOS(r) 环路参数: OpticalPower

描述

此参数使您能够设置 SERCOS(r) 回路发射器 (LED) 的光功率（光输出）。

该值作为百分比给出，实际更改精度为 20% 的幂。

提供的最小非 0 光功率为万分之 66。提供的全功率为 100%。如果该值为 0%，则会切断 LED 输出。

注意：光功率的缺省值为万分之 66。此值通常适用于所有 SERCOS(r) 驱动器和所有光缆长度。如果运动控制器与第一个驱动器之间的光纤长度极短，可能需要降低光功率设置。在此情况下，降低光功率可以防止驱动器的接收器 LED 电路超速运行。

关联的功能

此参数的关联功能包括：

- GetOpticalPower，它发送针对实际发射器光功率的读取请求
- SetOpticalPower，它发送针对新的发射器光功率的写入请求

这些功能使用（根据语言对象）WRITE_CMD、READ_PARAM 或 WRITE_PARAM 指令。

关联的语言对象

下表显示与 OpticalPower 参数关联的不同语言对象。

对象	名称	功能代码	指令类型	描述
%MWr.m.c.26	ACTION_CMD	1547	WRITE_CMD	读取命令
		2547		写入命令
%MFr.m.c.22	RETURN_CMD_2	-	-	读取的结果。
%MFr.m.c.31	PARAM_CMD_3	-	-	要写入的值
%MWr.m.c.19	ERROR_CMD	-	-	错误报告。（请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页）

13.6 SERCOS(r) IDN 读 / 写功能

本节标题

本节描述用于 SERCOS(r) ID 的读 / 写功能并列出了标准 ID。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
伺服驱动器参数：IDN_S 和 IDN_P	302
伺服驱动器参数：IDN_US 和 IDN_UP	304
标准 SERCOS 标识号 (IDN)	306
自定义电报的 IDN	309

伺服驱动器参数: IDN_S 和 IDN_P

描述

这些参数使您能够按项目直接控制伺服驱动器（独立于 TSX CSY 84 模块）。

它们可划分为两个类别：

- IDN_S 认可的标准（SERCOS(r) 标准），请参见 *标准 SERCOS 标识号 (IDN)*，第 306 页
- IDN_P 认可的所有者（特定于驱动器制造商，请参见有关伺服驱动器的信息）

注意：此制造商信息应该指定：

- 参数的大小（16 或 32 位）
- 参数是有符号的还是无符号的
- 授权该参数的 SERCOS(r) 回路阶段
- 参数的合法值
- 必须使用什么类型的比例调整（如果有）来解释数据

关联的功能

此参数的关联功能包括：

- GetIDN_S 和 GetIDN_P，它们发送针对 SERCOS(r) 标识号 (IDN) 的当前参数的读取请求。这些参数通过轴控模块来读取。
- SetIDN_S 和 SetIDN_P，它们发送针对 SERCOS(r) 标识号 (IDN) 的新参数的写入请求。

注意：(IDN) SERCOS(r)：分别为 S-0-xxxx（标准参数）和 P-0-xxxx（产品专用参数）

这些功能只能用于从 SERCOS(r) 伺服驱动器读取或向其中写入 16 位或 32 位参数。不能使用这些功能访问 SERCOS(r) 可变长度的文本参数。

在读取时，16 和 32 位参数以 32 位值的形式返回。16 位 SERCOS(r) IDN 值基于 16 位 SERCOS(r) IDN 的类型扩展为 32 位返回值。

例如，如果 16 位 SERCOS(r) 参数是有符号参数，则将 16 位值带符号扩展为 32 位有符号值。同样地，如果 16 位 SERCOS(r) 参数是无符号参数，则通过在 32 位返回值的最高 16 位放置零来扩展该 16 位。

注意：有些 SERCOS(r) IDN 参数具有与它们关联的单位和比例系数。此外，这些功能不允许转换参数（从使用轴缺省单位的值转换为使用驱动器单位的值，或反向转换）。（因此应使用 IDN_US 和 IDN_UP 功能）

这些功能使用 WRITE_CMD 指令。

语言对象

下表显示与 OpticalPower 参数关联的不同语言对象。

对象	名称	功能代码	指令类型	描述
%MWr.m.c.26	ACTION_CMD	1556	WRITE_CMD	读取命令 IDN_S
		2556		写入命令 IDN_S
%MWr.m.c.26		1557		读取命令 IDN_P
		2557		写入命令 IDN_P
%MDr.m.c.20	RETURN_CMD_1	-	-	读取的结果。
%MDr.m.c.27	PARAM_CMD_1	-	-	要写入的值：IDNSercos。1 至 4095 的格式类型。但是，允许使用位 12 至 14。这些位指示正在选择哪些参数集。（请参见您的伺服驱动器手册以了解支持哪些参数集）
%MDr.m.c.29	PARAM_CMD_2	-	-	要写到伺服驱动器的值
%MFr.m.c.31	PARAM_CMD_3	-	-	要写入的值：AddressSercos。如果正在 BusSERCOSaxisID 999 上使用 GetIDN 功能，则此参数是必需的。
%MWr.m.c.19	ERROR_CMD	-	-	错误报告。（请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页）

读取命令 IDN_S

具有 SERCOS 标识号 (IDN) 的当前参数由读命令通过轴控制模块读取。下面的示例说明如何使用此命令和所有 IDN（无符号字节长度为 2）读取 IDN100 值。

示例：

IDN100 为 2 无符号字节长度。PLC 寄存器（读命令）将无符号值置于有符号值中。

如果该 IDN100 值超过 32767，则 PLC 寄存器将返回如下结果：

- 32768 -> -32768
- 32769 -> -32767
- 35000 -> -30536

当读命令返回负值时，可以通过加上 65536 将该值转换为其正的等价值。

伺服驱动器参数: IDN_US 和 IDN_UP

描述

这些参数与单位和比率系数相关联。

它们使您能够按项目直接控制伺服驱动器（独立于 TSX CSY 84 模块）。

它们可划分为两个类别：

- IDN_US 认可的用户标准（标准 SERCOS^(r)），（请参见 *标准 SERCOS 标识号 (IDN)*, 第 306 页）
- IDN_UP 认可的用户所有者（特定于驱动器制造商，请参见有关驱动器的信息）。

注意：此制造商信息应该指定：

- 参数是否有单位
- 授权该参数的 SERCOS(r) 回路阶段
- 参数的合法值

伺服驱动器手册中有关如何对参数进行比率调整以及如何确定单位的信息在此情况下不适用。运动控制器将使用伺服驱动器提供的信息来转换参数。

关联的功能

此参数的关联功能包括：

- GetIDN_US 和 GetIDN_UP, 它们发送针对 SERCOS(r) 标识号 (IDN) 的当前参数的读取请求（以已从驱动器单位转换为缺省轴单位的浮点数形式）
- SetIDN_US 和 SetIDN_UP, 它们发送针对 SERCOS(r) 标识号 (IDN) 的新参数的写入请求（以用此参数的缺省轴单位表示的浮点数形式）。运动控制器将此数据值从缺省轴单位转换为驱动器单位，并将转换后的值写入伺服驱动器。

注意：(IDN) SERCOS(r): 分别为 S-0-xxxx（标准参数）和 P-0-xxxx（产品专用参数）

注意：这些功能不允许您访问没有单位的 SERCOS^(r) 参数或可变长度的文本参数。

注意：这些参数还可以由 Get/SetIDN_S 或 Get/SetIDN_P 功能读 / 写，因此观察所使用的单位是必要的。

这些功能使用 WRITE_CMD 指令。

语言对象

下表显示与 OpticalPower 参数关联的不同语言对象。

对象	名称	功能代码	指令类型	描述
%MWr.m.c.26	ACTION_CMD	1558	WRITE_CMD	读取请求 IDN_US
		2558	WRITE_CMD	写入请求 IDN_US
%MWr.m.c.26	ACTION_CMD	1559	WRITE_CMD	读取请求 IDN_UP
		2559	WRITE_CMD	写入请求 IDN_UP
%MFr.m.c.22	RETURN_CMD_2	-	-	读取报告：浮点参数的值 (1)。
%MDr.m.c.27	PARAM_CMD_1	-	-	IDNSercos。1 至 4095 的格式类型。但是，允许使用位 12 至 14。这些位指示正在选择哪些参数集。（请参见您的伺服驱动器手册以了解支持哪些参数集）
%MFr.m.c.31	PARAM_CMD_3	-	-	AddressSercos。如果正在 BusSERCOSaxisID 999 上使用 GetIDN 功能，则此参数是必需的。
%MFr.m.c.33	PARAM_CMD_4	-	-	变量的内容：浮点参数的值 (1)。
%MWr.m.c.19	ERROR_CMD	-	-	错误报告。（请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页）
说明				
(1)	此值以该类型的参数的缺省轴单位表示（例如，位置、速度、加速度、扭矩和时间的单位）。			

标准 SERCOS 标识号 (IDN)

位置参数

下表列出位置参数的 IDN:

IDN	标准 IDN 名称
S-0-0032	主操作模式
S-0-0053	位置反馈值 2 (外部反馈)
S-0-0055	位置极性参数
S-0-0116	循环反馈 1 的精度
S-0-0117	循环反馈 2 的精度
S-0-0121	负载传动系统的输入转数
S-0-0122	负载传动系统的输出转数
S-0-0123	馈给常量
S-0-0206	驱动器打开延迟时间
S-0-0207	驱动器关闭延迟时间
S-0-0256	倍增系数 1
S-0-0257	倍增系数 2

速度参数

下表列出速度参数的 IDN:

IDN	标准 IDN 名称
S-0-0037	增加的速度命令值
S-0-0040	速度反馈值

扭矩 / 电机参数

下表列出扭矩 / 电机参数的 IDN:

IDN	标准 IDN 名称
S-0-0084	扭矩反馈值
S-0-0092	双向扭矩限制值
S-0-0111	电机连续堵转电流

限制参数

下表列出限制参数的 IDN:

IDN	标准 IDN 名称
S-0-0038	正向速度限制值
S-0-0039	负向速度限制值
S-0-0049	正向位置限制值
S-0-0050	负向位置限制值
S-0-0091	双向速度限制值
S-0-0114	电机的负载限制
S-0-0136	正向加速度限制
S-0-0138	双向加速度限制值

诊断

下表列出诊断的 IDN:

IDN	标准 IDN 名称
S-0-0011	1 类诊断 (C1D)
S-0-0012	2 类诊断 (C2D)
S-0-0013	3 类诊断 (C3D)
S-0-0041	回归速度
S-0-0042	回归加速度
S-0-0147	回归参数
S-0-0148	驱动器发出的回归命令
S-0-0298	归零转换距离
S-0-0400	归零转换
S-0-0403	位置反馈值状态

比率调整参数

下表列出比率调整参数的 IDN:

IDN	标准 IDN 名称
S-0-0044	速度数据比率调整类型
S-0-0086	扭矩数据比率调整类型
S-0-0093	扭矩 / 力数据比率调整因子
S-0-0094	扭矩 / 力数据比率调整指数
S-0-0160	加速度数据比率调整类型

探测器

下表列出探测器的 IDN:

IDN	标准 IDN 名称
S-0-0130	探测器值上升沿 1
S-0-0131	探测器值下降沿 1
S-0-0169	探测器控制参数
S-0-0170	探测器循环过程命令
S-0-0179	探测器状态

增益参数

下表列出增益参数的 IDN:

IDN	标准 IDN 名称
S-0-0296	速度前馈增益

驱动器、驱动器备份以及校验和

下表列出驱动器、驱动器备份以及校验和的 IDN:

IDN	标准 IDN 名称
S-0-0192	备份操作数据的 IDN 列表
S-0-0271	驱动器 ID
S-0-0263	负载工作存储器过程命令
S-0-0264	备份工作存储器过程命令

自定义电报的 IDN

自定义电报

下表列出自定义电报的 IDN:

IDN	标准 IDN 名称
S-0-0016	AT 循环数据的配置列表
S-0-0024	MDT 的配置列表
S-0-0130	探测器 1 上升沿
S-0-0131	探测器 1 下降沿
S-0-0037	附加的速度命令
S-0-0053	位置反馈值 2 (外部编码器)
S-0-0084	扭矩反馈值
S-0-0185	AT 中的可配置数据记录的长度
S-0-0187	AT 中的可配置数据的 IDN 列表
S-0-0188	MDT 中的可配置数据的 IDN 列表
S-0-0186	MDT 中的可配置数据记录的长度

本章的目标

本章描述 SERCOS(r) 通道的调试功能。

用于 TSX CSY 84 的所有步骤都与用于 TSX CSY 164 模块的步骤相同。

本章包含了哪些内容？

本章包含了以下主题：

主题	页
调试屏幕描述：一般信息	312
调试屏幕用户界面	315
调试：启用	316
调试：诊断	317
调试：运动	319
调试：发送命令（自动模式）	321
调试：手动命令（手动模式）	322
调试：参考	323
调试：跟随	324
调试：驱动器	325
调试：位置	326
模块诊断	327
通道诊断	328

调试屏幕描述：一般信息

简介

可以对所有类型的通道使用不同的调试屏幕，以显示模块通道的输入和输出状态以及任何故障。

它还使您控制语言对象（设置为 1 或 0，强制 / 取消强制位，等）。

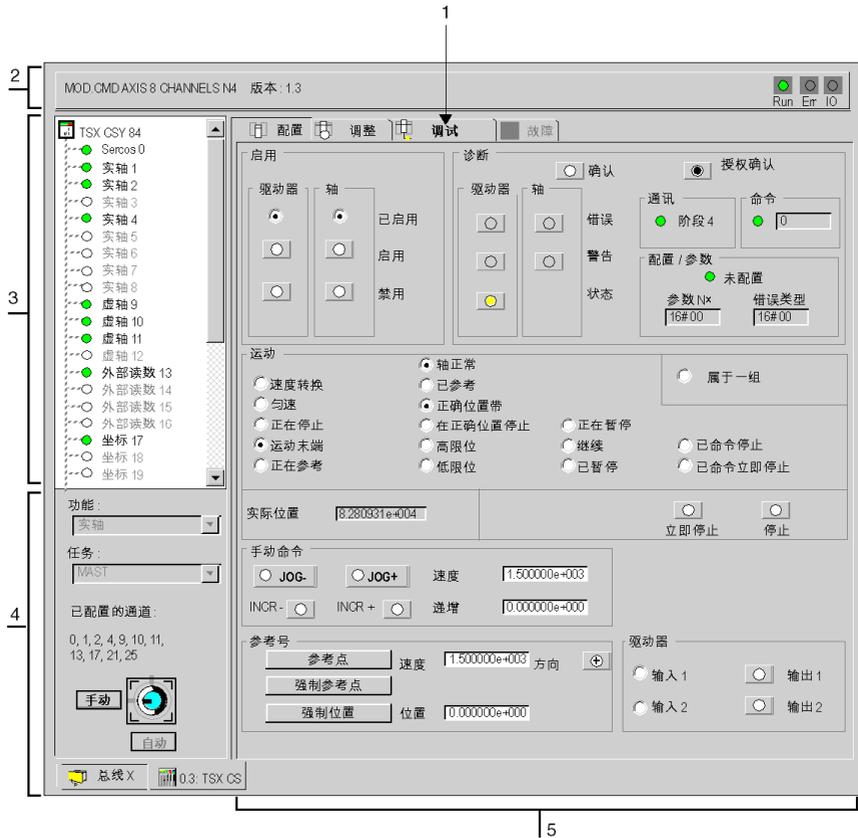
这些参数可在在线模式下访问。

它还用于访问配置屏幕和调整屏幕。

注意：有关配置、调整和调试项目的过程简介，请参见关于常用应用专用功能的部分。

示意图

下图显示了 TSX CSY 84 模块的调试屏幕示例 (TSX CSY 164 模块的调试屏幕与此完全相同)：



描述

下表显示了调试屏幕的各个元素。

地址	元素	功能
1	选项卡	前景中的选项卡指示当前模式（此示例中为 调试 ）。使用各选项卡可选择相应的模式。可用的模式为： <ul style="list-style-type: none"> ● 调试，只能在在线模式下访问， ● 诊断（故障），只能在在线模式下访问， ● 调整， ● 配置。
2	模块区域	显示模块的简短标题。 同一区域中有 3 个指示灯，用于指示在线模式中的模块状态： <ul style="list-style-type: none"> ● RUN，指示模块的操作模式。 ● ERR，指示内部模块错误。 ● I/O，指示模块外部故障或应用故障。
3	通道区域	允许您： <ul style="list-style-type: none"> ● 通过单击设备参考，显示选项卡： <ul style="list-style-type: none"> ● 描述：提供设备特性。 ● I/O 对象（参见 <i>Unity Pro, 操作模式</i>，）提供输入/输出对象的预览符号。 ● 故障：提供对设备故障的访问（在线模式下）。 ● 选择通道， ● 显示符号，即用户（通过变量编辑器）定义的通道名。
4	常规参数区域	用于强制位和查看计数功能： <ul style="list-style-type: none"> ● 取消强制：用于取消强制使用强制位的按钮。 ● 功能：取消已配置的计数功能。此信息不能修改。 ● 任务：显示已配置的 MAST 或 FAST 任务。此信息不能修改。
5	当前参数区域	此区域显示输入、输出的状态和正在进行的计数的不同参数。如果计数寄存器的内容在输入故障后不可用，则消息或 LED 无效测量 显示为红色。

调试屏幕用户界面

简介

调试屏幕通过 LED 提供对输入和输出位状态的访问。

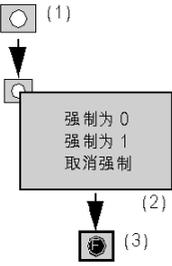
它还使您能够使用命令按钮控制语言对象 (%Q)

注意：将鼠标光标定位到所需 LED/ 按钮上可查看对应的语言对象。

调试屏幕还使您能够修改和发送运动控制。

命令按钮

下表显示了命令按钮的不同可能方面。

命令	功能
	与此位关联的位处于位置 0。单击可将该位移至位置 1。
	与此位关联的位处于位置 1。单击可将该位移至位置 0。
	<p>当某个对象可强制时，右击对应的按钮 (1) 将显示一个菜单 (2)，其中提供对强制命令的访问：强制为 0、强制为 1 或取消强制。通过单击选择了某个命令后，将应用强制，并且强制状态会在按钮级别 (3) 指示：</p> <ul style="list-style-type: none"> ● F 表示强制为 0 ● 反转视频（在示例中）中的 F 表示强制为 1。 <p>注：当某个语言对象被强制时，鼠标左键无法操作</p>
	此按钮与当前在位置 0 中的错误位关联。
	此按钮与当前在位置 1 中的错误位关联。单击此按钮可查看有关故障的信息。

调试：启用

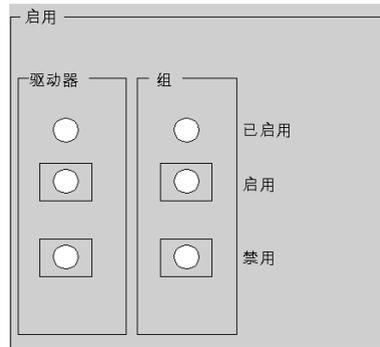
简介

此区域用于启用或禁用轴。为使之有效，必须在模块级别和在驱动器级别启用轴。

注意：此区域涉及实轴、虚轴和远程轴以及协同轴组和从轴组。

示意图

下图显示了调试屏幕的 " 启用 " 区域。



描述

下表描述了 " 启用 " 区域的不同驱动器区域元素。

元素	关联的语言对象	描述
LED 已启用		请参见 <i>DRIVE_ENABLED</i> 位 (%I.r.m.c.10), 第 148 页。
按钮启用	%Qr.m.c.10	在驱动器级别启用轴。
按钮禁用	%Qr.m.c.26	在驱动器级别禁用轴。

下表描述了 " 启用 " 区域的不同轴区域元素。

元素	关联的语言对象	描述
LED 已启用		请参见 <i>AXIS_IN_CMD</i> 位 (%I.r.m.c.18), 第 156 页。
按钮启用	%Qr.m.c.2	在轴级别启用轴。
按钮禁用	%Qr.m.c.18	在轴级别禁用轴。

调试：诊断

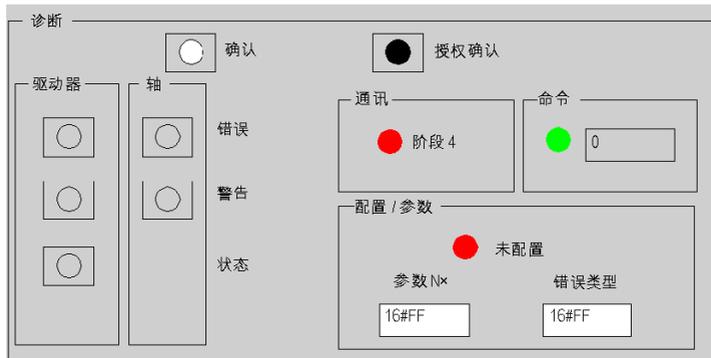
简介

此区域用于查看和确认轴故障。

注意：此区域涉及不同类型的轴、轴组和凸轮配置文件。

示意图

下图显示了调试屏幕的 " 诊断 " 区域。



描述

下表描述了 " 诊断 " 区域外的不同元素。这些元素在凸轮配置文件屏幕中不可用。

元素	关联的语言对象	描述
确认按钮	%Qr.m.c.15	故障确认
授权确认按钮	%Qr.m.c.31	确认故障

下表描述了 " 诊断 " 区域的不同驱动器区域元素。这些元素在凸轮配置文件屏幕中不可用。

元素	描述
错误 LED 错误按钮	请参见 <i>DRIVE_FAULT</i> 位 (%I.r.m.c.13)，第 151 页。 (1)
警告 LED 警告按钮	请参见 <i>DRIVE_WARNING</i> 位 (%I.r.m.c.12)，第 150 页。 (1)
状态 LED 状态按钮	请参见 <i>DRIVE_DIAG</i> 位 (%I.r.m.c.11)，第 149 页。 (1)
(1): 当 LED 亮时，单击该按钮将打开一个显示故障类型的窗口。	

下表描述了 " 诊断 " 区域的不同**轴**区域元素。此区域在凸轮配置文件屏幕中不存在。

元素	描述
错误 LED 错误按钮	请参见 <i>AXIS_SUMMARY_FLT</i> 位 (%lr.m.c.15), 第 153 页。 (1), 注: 如果存在 SERCOS(r) 通讯错误, 则不考虑驱动器故障数据。
警告 LED 警告按钮	请参见 <i>AXIS_WARNING</i> 位 (%lr.m.c.23), 第 160 页。 (1)
(1): 当 LED 亮时, 单击该按钮将打开一个显示故障类型的窗口。	

下表描述了 " 诊断 " 区域的不同**通讯**区域元素。此区域在凸轮配置文件屏幕和虚轴中不存在或不可用。

元素	关联的语言对象	描述
阶段 4 LED	%lr.m.c.16	请参见 <i>AXIS_COMM_OK</i> 位 (%lr.m.c.16), 第 154 页。

下表描述 " 诊断 " 区域的不同**命令**区域元素。

元素	关联的语言对象	描述
LED	%MWr.m.c.1:X2	显式命令故障 (Action_CMD)。
字段	%MWr.m.c.19	可能的值: 0: 参数正常 否则为错误代码。请参见 write_cmd: 编程错误命令章节 (参见第 357 页)

下表描述了 " 诊断 " 区域的不同**配置 / 参数设置**区域元素。

元素	关联的语言对象	描述
LED 未配置	%lr.m.c.32	通道已配置。
N° 参数字段	%IW2r.m.c.:X0 至 X7	指示参数错误。请参见故障寄存器章节 (参见第 352 页)
错误类型字段	%IW2r.m.c.:X8 至 X15	指示错误类型。请参见 (参见第 350 页)

调试：运动

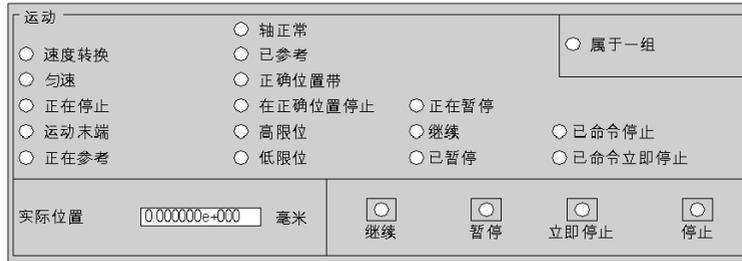
简介

此区域用于查看模块状态位和控制所选轴的运动。

注意： 它涉及实轴或虚轴以及协同轴组。

示意图

下图显示了调试屏幕的 "运动" 区域。



描述

下表描述了 "运动" 区域的各个元素。

元素	关联的语言对象	描述
速度转换 LED		请参见 <i>RAMPING</i> 位 (%I.r.m.c.0), 第 138 页。
匀速 LED		请参见 <i>STEADY</i> 位 (%I.r.m.c.1), 第 139 页。
正在停止 LED		请参见 <i>STOPPING</i> 位 (%I.r.m.c.2), 第 140 页。
运动结束 LED		请参见 <i>PROFILE_END</i> 位 (%I.r.m.c.3), 第 141 页。
正在参考 LED		请参见 <i>AXIS_HOMING</i> 位 (%I.r.m.c.5), 第 143 页。
轴正常 LED (仅单轴)		请参见 <i>AXIS_READY</i> 位 (%I.r.m.c.31), 第 168 页。
已参考 (仅单轴) LED		请参见 <i>AXIS_HOMED</i> 位 (%I.r.m.c.6), 第 144 页。
正确位置带 LED		请参见 <i>IN_POSITION</i> 位 (%I.r.m.c.4), 第 142 页。
在正确位置停止 LED		请参见 <i>AXIS_AT_TARGET</i> 位 (%I.r.m.c.20), 第 157 页。
上限位置 LED		请参见 <i>AXIS_POS_LIMIT</i> 位 (%I.r.m.c.21), 第 158 页。
下限位置 LED		请参见 <i>AXIS_NEG_LIMIT</i> 位 (%I.r.m.c.22), 第 159 页。
正在暂停 LED		请参见 <i>HOLDING</i> 位 (%I.r.m.c.8), 第 146 页。
正在恢复 LED		请参见 <i>RESUMING</i> 位 (%I.r.m.c.9), 第 147 页。
完成暂停 LED		请参见 <i>AXIS_HOLD</i> 位 (%I.r.m.c.28), 第 165 页。

元素	关联的语言对象	描述
已命令停止 LED	请参见 <i>AXIS_HALT</i> 位 (%I.r.m.c.29), 第 166 页。	
已命令立即停止 LED	请参见 <i>AXIS_FASTSTOP</i> 位 (%I.r.m.c.30), 第 167 页。	
属于一组 (仅单轴) LED	请参见 <i>AXIS_IS_LINKED</i> 位 (%I.r.m.c.17), 第 155 页。	
恢复按钮	%Qr.m.c.12	停止后恢复控制。
暂停按钮	%Qr.m.c.28	临时停止轴。
立即停止 LED	%Qr.m.c.30	紧急停止命令。
停止 LED	%Qr.m.c.29	停止轴。
当前位置 (仅单轴) 字段	%IFr.m.c.0	当前位置的值。

调试：发送命令（自动模式）

简介

此区域特定于自动模式，用于通过显式交换启动运动控制。

示意图

下图显示了调试屏幕的 "发送命令" 区域。

The screenshot shows a control panel titled "发送命令". At the top left, there is a "代码" (Code) input field containing the value "0". To its right is a circular LED indicator labeled "正在执行命令" (Command in Progress), which is currently unlit. Below these are four rows of input fields for parameters and feedback values:

- 参数 1 (Parameter 1): 0
- 参数 2 (Parameter 2): 0
- 参数 3 (Parameter 3): 0.000000e+000
- 参数 4 (Parameter 4): 0.000000e+000
- 反馈 1 (Feedback 1): 0
- 反馈 2 (Feedback 2): 0.000000e+000
- 反馈 3 (Feedback 3): 0.000000e+000

At the bottom right of the panel is a button labeled "发送命令" (Send Command).

描述

下表描述了 "发送命令" 区域的各个元素。

元素	关联的语言对象	描述
代码字段	%MWr.m.c.26	功能代码
正在执行命令 LED	%MWr.m.c.0:X1	(参见第 389 页)
参数字段	%MWr.m.c.27 至 33 (1) %MWr.m.c.27 至 33 (2)	功能参数
返回值 N 字段	%MDr.m.c.20 %MFr.m.c.22 %MFr.m.c.24	命令的结果
命令按钮	-	通过 WRITE_CMD 的显式交换。
注： 有关更多信息，请参考对应于所需功能的文档。		
(1): 实轴、虚轴和从轴组。		
(2): 协同轴组		

调试：手动命令（手动模式）

简介

此区域特定于手动模式，用于启动连续 (JOG) 或增量 (INC) 运动控制。

注意： 它涉及实轴和虚轴。

示意图

下图显示了调试屏幕的 "手动命令" 区域。



描述

下表描述了 "手动命令" 区域的各个元素。

元素	关联的语言对象	描述
JOG+	%Qr.m.c.4	请参见 <i>手动模式</i> ，第 176 页。 与此命令关联的控制钮为按钮。
JOG-	%Qr.m.c.5	请参见 <i>手动模式</i> ，第 176 页。 与此命令关联的控制钮为按钮。
INC+	%Qr.m.c.13	请参见 <i>手动模式</i> ，第 176 页。
INC-	%Qr.m.c.14	请参见 <i>手动模式</i> ，第 176 页。
速度	-	缺省配置的值（Smax 的 1/2）。此值可修改。此值可由该屏幕或通过 PARAM_CMD_3 由命令 SetDefaultSpeed (2065) 修改。
增量	%QDr.m.c	所需的增量值。

调试：参考

简介

此区域特定于手动模式，可用于将某个轴的参考点设为某个给定位置（即“主位置”）。

示意图

下图显示了调试屏幕的“参考”区域。



描述

下表描述了“参考”区域的各个元素。

元素	描述
参考点	设置回归参考点（请参见 <i>回归功能</i> ，第 179 页）。
速度	用于到达主位置的运动速度。
方向	用于到达主位置的旋转方向（正向或负向）
强制参考点	轴的位置成为回归点（请参见 <i>ForcedHomed</i> 功能，第 182 页）。
强制位置	修改回归点而不将轴的参考点设置为该点（请参见 <i>SetPosition</i> 功能，第 184 页）。
位置	主位置强制值。

调试：跟随

简介

此区域用于激活或禁止从轴的跟随。

注意：此区域涉及实轴和虚轴。

示意图

下图显示了调试屏幕的 "跟随" 区域。



描述

下表描述了 "跟随" 区域的各个元素。

元素	关联的语言对象	描述
LED 偏差已激活	%lr.m.c.24	偏移已添加到命令位置
按钮跟随	%Qr.m.c.11	状态更改时可修改的控制位, 第 174 页
按钮停止跟随	%Qr.m.c.27	可在跳变沿上修改的运动控制位, 第 173 页

调试：驱动器

简介

此区域用于控制驱动器输入和输出。这些输入 / 输出由用户定义。

注意：此区域仅在实轴屏幕中存在。

示意图

下图显示了调试屏幕的 " 驱动器 " 区域。



描述

下表描述了 " 驱动器 " 区域的各个元素。

元素	关联的语言对象	描述
LED 输入 1	%I26	请参见 <i>DRIVE_REALTIME_BIT1</i> 位 (%l.r.m.c.26), 第 163 页
LED 输入 2	%I27	请参见 <i>DRIVE_REALTIME_BIT2</i> 位 (%l.r.m.c.27), 第 164 页
按钮输出 1	%Q6	驱动器命令
按钮输出 2	%Q7	

调试：位置

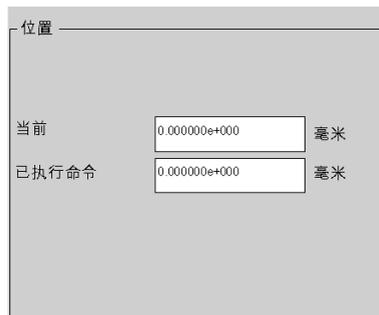
简介

此区域用于查看项目中的外部轴位置。

注意：此区域仅在远程轴屏幕中存在。

示意图

下图显示了调试屏幕的 "位置" 区域。



描述

下表描述了 "位置" 区域的各个元素。

元素	关联的语言对象	描述
字段 当前位置	%IF0	当前位置的值。
字段 命令的位置	%QD0	由项目定义的值。

模块诊断

概览

模块诊断功能可以显示当前故障（如果有），并根据类别对故障分类：

- **内部错误：**
 - 模块功能失常。
 - 正在进行自动测试
- **外部错误：**
 - 故障端子块，
- **其他错误：**
 - 配置故障：
 - 模块缺失或关闭，
 - 故障通道 (参见 *使用 Unity Pro 的 Premium 和 Atrium, 模拟量输入 / 输出模块, 用户手册*)。

故障模块由某些 LED 变红来指示，例如：

- 在机架级的配置编辑器中：
 - 机架号 LED，
 - 模块机架位置编号 LED。
- 在模块级的配置编辑器中：
 - 按照故障类型的 **Err** 和 **I/O** LED，
 - **通道**区域中的**通道** LED，
 - **故障**选项卡。

过程

下表介绍了访问模块故障屏幕的过程。

步骤	操作
1	访问 " 调试模块 " 屏幕。
2	<p>单击通道区域中的模块参考号，然后选择故障选项卡。</p> <p>结果：将出现模块故障列表。</p> <div data-bbox="445 1198 871 1404" data-label="Image"> <p>The screenshot shows a window titled '故障' (Fault) with a red indicator light. Below the title bar are three main sections: '内部错误' (Internal Error), '外部错误' (External Error), and '其他错误 故障通道' (Other Error Fault Channel). Each section contains a list of fault details, though the text is too small to read clearly.</p> </div> <p>请注意：模块诊断屏幕在配置故障、严重损坏或缺失模块时不可用。此时将显示下面的消息：模块缺失或与此位置配置的模块不同。</p>

通道诊断

概览

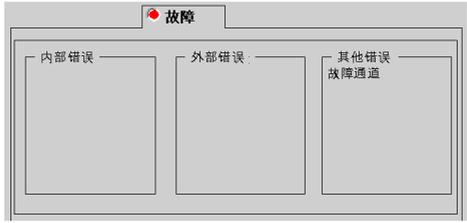
诊断功能通道可以显示当前故障（如果有），并根据类别对故障分类：

- **内部错误：**
 - 通道出现故障。
- **外部错误：**
 - 链路或传感器电源故障，
- **其他错误：**
 - 故障端子块
 - 配置故障
 - 通讯故障

故障通道由**诊断** LED 变红来指示，该 LED 位于配置编辑器的**错误**列中。

过程

下表介绍了用于访问诊断通道屏幕的过程。

步骤	操作
1	访问调试模块屏幕。
2	<p>单击诊断按钮（位于通道故障的错误列中）。 结果：出现通道故障列表。</p>  <p>请注意：有关通道的诊断信息也可以通过程序（指令 READ_STS）来访问。</p>

本章主题

本章描述各种诊断功能，给出由轴控模块生成的错误列表，并描述状态位。

本章包含了哪些内容？

本章包含了以下部分：

节	主题	页
15.1	SERCOS(r) 阶段诊断简介	330
15.2	读取环路的实际数据简介	334
15.3	轴标识	338
15.4	状态和故障信息	339
15.5	故障和警告	347
15.6	TSX CSY 84 的性能	369

15.1 SERCOS(r) 阶段诊断简介

本节主题

本节描述 SERCOS(r) 阶段的诊断参数功能。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
关于 SERCOS(r) 阶段标准	331
GetActualPhase 功能	332
GetCommandedPhase 功能	333

关于 SERCOS(r) 阶段标准

标准 IEC 1491

IEC 1491 标准（基于 SERCOS(r) 的国际伺服驱动器通讯协议标准）定义了与 SERCOS(r) 驱动器通讯的以下五个阶段。

SERCOS(r) 阶段

下表定义各个阶段：

阶段	含义
0	回路未通讯。
1	控制器正在查找回路上的驱动器。
2	正在与驱动器交换 SERCOS(r) 回路定时参数。
3	到驱动器的服务通道正在正常工作，所有定时参数都在使用中。GetIDN 和 SetIDN 功能已就绪，可以操作。
4	运动控制器正在向驱动器发送循环通讯信息。回路已启动，允许使用运动命令。

GetActualPhase 功能

描述

此功能返回实际的 SERCOS(r) 回路阶段。

如何使用此功能

此功能是在 WRITE_CMD 指令（请参见 *WRITE_CMD*, 第 388 页）的帮助下实现的。

关联的语言对象

通过 **WRITE_CMD** 进行交换：下表显示与此功能关联的不同语言对象。

对象	名称	功能代码	描述
%MWr.m.c.26	ACTION_CMD	550	命令
%MDr.m.c.20	RETURN_CMD_1	-	命令的结果
%MWr.m.c.19	ERROR_CMD	-	错误报告。请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页。

GetCommandedPhase 功能

描述

GetCommandedPhase 功能返回运动控制器正在尝试建立的 SERCOS(r) 回路阶段。

注意：使用 GetActualPhase 功能确定当前的回路阶段。

SetCommandedPhase 功能请求运动控制器将 SERCOS(r) 回路更改到指定的阶段。

SERCOS(r) 阶段必须始终以增量方式切换阶段。如果 SetCommandedPhase 值低于当前回路阶段，则运动控制器将阶段切换到阶段 0，并将回路顺序往上调整到指定的阶段。如果回路在阶段 4，并且命令了更低的阶段，则运动轴将发生故障。

注意：如果回路上的所有驱动器都允许切换到下一个阶段，则运动控制器只能前进到下一个 SERCOS(r) 阶段。因此，所命令的阶段可能永远不会实现。使用 GetCommandedPhase 功能确定所命令的阶段，并使用 GetActualPhase 功能发现 SERCOS(r) 回路的实际阶段。

如何使用这些功能

这些功能是在 WRITE_CMD 指令（请参见 *WRITE_CMD*, 第 388 页）的帮助下实现的。

关联的语言对象

通过 WRITE_CMD 进行交换：下表显示与这些功能关联的不同语言对象。

对象	名称	功能代码	描述
%MWr.m.c.26	ACTION_CMD	1545	读取命令
		2545	写入命令
%MDr.m.c.20	RETURN_CMD_1	-	读取命令的结果
%MDr.m.c.27	PARAM_CMD_1	-	要写入的值
%MWr.m.c.19	ERROR_CMD	-	错误报告。请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页。

15.2 读取环路的实际数据简介

本节主题

本节描述用于读取环路的实际数据（SERCOS(r) 地址、驱动器数量等）的功能。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
GetSercosAddress 功能	335
GetNumberOfDrivesInRing 功能	336
IsLoopUp 功能	337

GetSercosAddress 功能

描述

此功能返回该轴的 SERCOS(r) 地址。

注意：只有实轴和远程轴才存在此功能。

注意：轴的 SERCOS(r) 地址还在对应通道的常量字 %KWr.m.c.0 中提供。

如何使用此功能

此功能是在 WRITE_CMD 指令（请参见 *WRITE_CMD*, 第 388 页）的帮助下实现的。

关联的语言对象

通过 WRITE_CMD 进行交换：下表显示与此功能关联的不同语言对象。

对象	名称	功能代码	描述
%MWr.m.c.26	ACTION_CMD	549	命令
%MDr.m.c.20	RETURN_CMD_1	-	命令的结果 (1)
%MDr.m.c.27	PARAM_CMD_1	-	要写入的值 (2)
%MWr.m.c.19	ERROR_CMD	-	错误报告。请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页。
说明			
(1) 如果地址 = 25350：未在该地址配置轴或该地址无效。			
(2)：此值对应于回路中存在的实轴的表索引（0 = 第一个轴，1 = 第二个轴，以此类推）。如果此功能是在对应于要请求其地址的轴的通道上发出的，则不需要此参数。			

GetNumberOfDrivesInRing 功能

描述

此功能返回光纤 SERCOS(r) 网络环路中的轴数量。

如何使用此功能

此功能是在 WRITE_CMD 指令（请参见 *WRITE_CMD*, 第 388 页）的帮助下实现的。

关联的语言对象

通过 WRITE_CMD 进行交换：下表显示与此功能关联的不同语言对象。

对象	名称	功能代码	描述
%MWr.m.c.26	ACTION_CMD	548	命令
%MDr.m.c.20	RETURN_CMD_1	-	命令的结果
%MWr.m.c.19	ERROR_CMD	-	错误报告。请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页。

IsLoopUp 功能

描述

此功能指示 SERCOS(r) 网络环路（SERCOS(r) 环路）是否已准备好与运动控制器通讯以及是否处于阶段 4。

可能的值为：

- 如果回路处于 "最佳" 状态（积极通讯），则为 1 = (TRUE)
- 如果回路处于 "低迷" 状态（阶段 4 通讯未建立），则为 0 = (FALSE)

使用 GetActualPhase 功能确定当前的回路阶段。

如何使用此功能

此功能是在 WRITE_CMD 指令（请参见 *WRITE_CMD*, 第 388 页）的帮助下实现的。

关联的语言对象

下表显示与此功能关联的不同语言对象。

对象	名称	功能代码	描述
%MWr.m.c.26	ACTION_CMD	543	命令
%MDr.m.c.20	RETURN_CMD_1	-	命令的结果
%MWr.m.c.19	ERROR_CMD	-	错误报告。请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页。

15.3 轴标识

GetNumberInSet 功能

描述

此功能通过以下方式传输组中包含的轴数量：

- SERCOS(r) 功能：该功能传输在项目中配置的实轴、虚轴、远程轴、协同轴组和从轴组的总数。
- 协同轴和从轴组：该功能传输组中包含的独立轴数量
- 实轴、虚轴和远程轴：此功能传输值 1，表示该轴是独立轴

如何使用此功能

此功能是在 WRITE_CMD 指令（请参见 *WRITE_CMD*, 第 388 页）的帮助下实现的。

关联的语言对象

下表显示与此功能关联的不同语言对象。

对象	名称	功能代码	描述
%MWr.m.c.26	ACTION_CMD	541	命令
%MDr.m.c.20	PARAM_CMD_1	-	命令的结果
%MWr.m.c.19	ERROR_CMD	-	错误报告。请参见 <i>WRITE_CMD</i> 命令：编程错误，第 357 页。

15.4 状态和故障信息

本节主题

本节描述读取故障和状态信息的功能。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
隐式交换对象	340
可通过 GetMotionFault 命令访问的故障	341
GetMotionFault 功能	343
可通过 GetMotionWarning 命令访问的故障	344
GetMotionWarning 功能	346

隐式交换对象

故障模块

%I.r.m.MOD.ERR 位处于状态 1 指示位于位置 r.m 中的模块出现故障。故障原因在 %MWr.m.MOD.2 内部字中给出。

故障通道

%I.r.m.c.ERR 位处于状态 1 指示位于位置 r.m 中的模块的通道 i 出现故障。故障原因在 %MWr.m.c.2 内部字中给出。

可通过 GetMotionFault 命令访问的故障

描述

MotionFault 数据指示导致在轴的 MotionStatus 字中出现 AXIS_SUMMARY_FLT 状态的错误类型。此类错误通过设置 MotionFault 字中的一个位进行记录，每种错误类型对应于一个唯一的位。多个不同错误设置它们在 MotionFault 字中的对应位。相同类型的多个错误设置同一个位，因此 MotionFault 字不指示发生了多少个相同种类的错误。向运动轴发出 ClearFault 功能可清除 MotionFault 字。

运动故障

下表列出 MotionFault 数据类型中包含的运动故障：

名称	位	描述
MF_MOVE_BUT_NOT_ENABLED	0	运动命令已到达运动内插器，但是驱动器已禁用。仅当驱动器在运动命令正在启动时被禁用，才会发生此类错误。
MF_SERCOS_RATE_TOO_FAST	3	相对于所选的循环时间，配置的通道过多。
MF_CONTROL_CONFLICT	10	与驱动器配置工具发生控制冲突。
MF_DRIVE_FAULT	13	驱动器发生故障。带 SERCOS(r) 标准 IDN S-0-0011 参数使用 GetIDN_S 和 GetIDN_P 功能，以确定原因。
MF_REQUESTED_FAULT	15	MotionControl ALLOW_NOT_FAULT 位未设置。在设置 ALLOW_NOT_FAULT 位并发出 ClearFault 功能之前，轴保持故障状态。
MF_COMM_FAULT	16	SERCOS(r) 通讯光纤回路已失去与驱动器的通讯。
MF_AXIS_LIMIT_FAULT	21	轴已经到达正或负软件位置限制。
MF_PHASE3_CONFIG_PROBLEM	23	阶段 3 配置问题
MF_PHASE0_SERCOS_ERROR	24	阶段 0 出错。
MF_PHASE0_SERCOS_ERROR	25	阶段 1 出错。
MF_PHASE0_SERCOS_ERROR	26	阶段 2 出错。
MF_PHASE0_SERCOS_ERROR	27	阶段 3 出错。
MF_PHASE0_SERCOS_ERROR	28	阶段 4 出错。
MF_MOVE_WHILE_HALT	29	运动命令已到达内插器，但是 MotionControl ALLOW_MOVE 位未设置为 1。仅当驱动器恰好在运动命令启动时停止，才会发生此类错误。

驱动器故障

如果还存在 SERCOS(r) 通讯错误 (MF_COMM_FAULT)，则不会考虑驱动器故障数据 (MF_DRIVE_FAULT)。事实上，即使通讯仍然能够进行，更改到阶段 3 也会触发通讯错误。

GetMotionFault 功能

描述

此功能传输有关某个轴的运动故障 (MovementFault) 信息。MovementFault 字 (参见第 341 页) 通过发送到轴的 ClearFault 功能复位为零。

如何使用此功能

此功能是在 WRITE_CMD 指令 (请参见 WRITE_CMD, 第 388 页) 的帮助下实现的。

关联的语言对象

下表显示与此功能关联的不同语言对象。

对象	名称	功能代码	描述
%MWr.m.c.26	ACTION_CMD	5510	命令
%MDr.m.c.20	RETURN_CMD_1	-	命令的结果
%MWr.m.c.19	ERROR_CMD	-	错误报告。请参见 WRITE_CMD 命令：编程错误，第 357 页。

可通过 GetMotionWarning 命令访问的故障

说明

运动警告 (MotionWarning) 是由运动控制器生成的，当通过运动控制 (MotionControl) 寄存器向运动集 (MotionSet) 发出运动命令，而当前时间不允许该命令时，则会生成运动警告。运动警告数据类型信息记录这些警告。

记录警告的方式是设置运动警告寄存器中的某个位，每种类型的警告对应一个唯一的位。不同类型的多个警告设置运动警告字中其对应的位。相同类型的多个警告设置同一个位，因此运动警告字不会指出发生了多少相同类型的警告。向轴发出 ClearFault（清除故障）功能可将运动警告字复位为零。使用 GetMotionWarning（获得运动警告）功能可读取运动警告。

运动警告

下表列出了运动警告数据类型中包含的运动警告：

名称	位	说明
MW_SURV_WARNING	0	如果两个轴有偏出警报阈值的运动，则模块将在这两个故障轴上触发警告。
MW_SURV_WARNING_PROP	1	如果两个轴有偏出警报阈值的运动，则模块在这两个故障轴上触发故障，停止所有列出的轴，并在非故障轴上触发警告 (MW_SURV_WARNING_PROP)。
MW_STOP_BY_SET	2	链接的轴上有故障。（在传播了故障之后停止的轴将报告为警告）。
MW_MERGE_UNLINK	7	运动在序列停用之后中止。
MW_AXIS_NOT_HOMED	6	未使用
MW_AXIS_TRAVEL_LIMIT	13	轴已到达限位开关或目标位置已超出行程范围。来自 Lexium 驱动器 15 的 IDN 12 的副本。要重新验证运动，首先必须确认它。
MW_AXIS_IS_LINKED	17	由于运动轴是 CoordinatedGroup（协同轴组）或 SlaveGroup（从轴组）的成员，因此运动控制命令未执行。
MW_AXIS_POS_LIMIT	21	轴已超过其正位限制。
MW_AXIS_NEG_LIMIT	22	轴已超过其负位限制。
MW_SIMULTANEOUS_MANUAL_CMDS	24	同时手动命令
MW_AXIS_MANUAL_MODE	25	未使用
MW_STOP_TO_MANUAL_MODE	26	在轴运动期间已激活手动模式。
MW_NOTALLOWED_AT_THIS_TIME	31	命令 doManualCmd 不可用：轴尚未准备就绪，不可用于手动命令。

对于协同轴组

下表列出了 CoordinatedGroup（协同轴组）的 MotionWarning（运动警告）数据类型中包含的运动警告：

名称	位	说明
MW_AXIS_IS_MOVING	3	由于运动轴正在运动，运动控制命令未执行。
MW_MEMBER_WARNING	4	成员上的警告。（在所所有的警告轴将作为警告报告的非启用组上启用）。
MW_MEMBER_FAULT	5	组成员上的故障。（在所的故障轴将作为警告报告的非启用组上启用）。
MW_CANNOT_ENABLE	10	驱动器将拒绝确认。
MW_LINK_TARGET_MISSED	11	仅限于组：功能 DynamicLink (576)。设定点与主站和从站轴实际上已到达的位置之间的偏移位置错误。
MW_ACQUIRE_DISALLOWED	18	CoordinatedGroup（协同轴组）或 SlaveGroup（从轴组）无法获得其成员的控制，因为一个或多个运动轴成员已经是某个协同轴组或从轴组的成员。

GetMotionWarning 功能

描述

此功能传输有关运动轴的 MovementWarning 位 (参见第 344 页)。这些位给出警告。它们带有 MW_ 前缀。

如何使用此功能

此功能是在 WRITE_CMD 指令 (请参见 WRITE_CMD, 第 388 页) 的帮助下实现的。

关联的语言对象

下表显示与此功能关联的不同语言对象。

对象	名称	功能代码	描述
%MWr.m.c.26	ACTION_CMD	5511	命令
%MDr.m.c.20	RETURN_CMD_1	-	命令的结果
%MWr.m.c.19	ERROR_CMD	-	错误报告。请参见 WRITE_CMD 命令：编程错误，第 357 页。

15.5 故障和警告

本节标题

本节说明可通过显式读取访问的故障和警告列表。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
可通过显式读取访问的模块故障	348
可通过显式读取访问的通道故障	349
配置和调整错误列表	350
故障寄存器	352
WRITE_CMD 命令的错误代码列表	356
WRITE_CMD 命令：编程错误	357
WRITE_CMD 命令：通讯错误	359
WRITE_CMD 命令：系统错误	360
WRITE_CMD 命令：系统警告	362
TRF_RECIPE 命令显式写入故障	363
操作模式	364
故障的逻辑图	365
伺服驱动器故障	367

可通过显式读取访问的模块故障

简介

字 %MWr.m.MOD.2 指示在位置 r.m 中的模块上出现的任何故障。

故障列表

位 %MWr.m.MOD.2.0 至 %MWr.m.MOD.2.15 用于诊断模块故障：

位	含义
0	内部故障：模块无法操作
1	功能故障：外部故障、通讯故障或应用故障（请参见通道状态字 %MWr.m.c.2）
2	端子块故障
3	模块正在执行自检
4	保留
5	配置故障：硬件和软件配置不同
6	模块缺失或关闭
7 到 15	保留

可通过显式读取访问的通道故障

简介

字 %MWr.m.c.2 指示在位置 r.m 中的模块的通道 i 上出现的任何故障。

故障列表

位 %MWr.m.c.2.0 至 %MWr.m.c.2.15 用于诊断通道故障：

位	含义
0	外部故障 0: 伺服驱动器故障
1	外部故障 1: 轴的通讯故障
2	保留
3	外部故障 2
4	内部故障
5	配置故障: 硬件和软件配置不同
6	通讯故障
7	项目配置故障、调整或命令故障
8	风扇故障 (仅通道 0)
9	过热 (仅通道 0)
10	温度传感器故障 (仅通道 0)
11	正在创建运动对象
12	配置故障 (通道 0 除外)
13	保留
14	通道 LED 状态: 常亮
15	通道 LED 状态: 闪烁

配置和调整错误列表

简介

字 %lWr.m.c.2 指示可能出现的编程故障。

- 最低有效字节 (LSB) 包含错误类型的返回代码
- 最高有效字节 (MSB) 包含导致错误的字段的各个寄存器中的地址。

故障列表

下表给出 LSB 的内容（错误代码，以十六进制表示）、故障的名称和描述：

错误代码	名称	描述
1	ERREUR_PLAGE	尝试分配超出范围的值
2	CORRESPONDANCE_UNITES	尝试分配不兼容的单位
3	UNITE_NON_SUPPORTEE	单位不支持或未知
4	ERREUR_DE_CHARGEMENT	伺服驱动器在上载时发生故障
5	ERREUR_DE_DECHARGEMENT	伺服驱动器在下载时发生故障
6	OBJET_VIDE	指向对象的意外空指针
7	ERREUR_DEFINITION_UNITES	未能设置伺服驱动器中的单位
8	UNITES_NON_DEFINIES	单位未设置
9	CHAINE_TROP_GRANDE	字符串对 MovementString 而言过大
A	INDEX_INCORRECT	组中的索引无效
B	VALEUR_INCORRECTE	命令中的值不正确
C	VALEUR_ENUM_INCORRECTE	序列中的值不正确
D	JETON_INCORRECT	输入中的令牌无效
E	VOIE_RETOUR_INCORRECTE	命令的反馈通道无效
F	PERIPHERIQUE_RETOUR_INCORRECT	命令的反馈外设无效
10	FREQ_HORLOGE_RETOUR_INCORRECTE	反馈时钟频率无效
11	ALIMENTATION_RETOUR_INCORRECTE	反馈电源无效
12	RESOLUTION_RETOUR_INCORRECTE	反馈精度无效
13	ADR_REG_MAINTIEN_INCORRECTE	保持寄存器地址无效
14	REGISTRES_MAINTIEN_NON_CONFIGURES	未配置保持寄存器数据库
15	REGISTRES_MAINTIEN_VIDE	保持寄存器数据库为空
16	BLOC_REG_TROP_GRAND	保持寄存器块过大

错误代码	名称	描述
17	BLOC_REG_NE_CORRESPOND_PAS	保持寄存器块与数据库不符
18	AUTORISATION_ACCES_IMPOSSIBLE	无法授予对保持寄存器块的访问权限
19	LIBERATION_ACCES_IMPOSSIBLE	无法释放对保持寄存器块的访问权限
1A	ERREUR_OUVERTURE_FICHER	文件打开失败
1B	ERREUR_ECRITURE_FICHER	文件写入失败
1C	ERREUR_LECTURE_FICHER	文件读取失败
1D	ERREUR_FERMETURE_FICHER	文件关闭失败
1E	ERREUR_RECHERCHE_FICHER	文件搜索失败
1F	ERREUR_SYNTAXE	输入格式错误
20	ERREUR_EFFACEMENT_DEFAULT	清除故障功能无法清除故障
21	TAG_MANQUANT	tags.cfg 中缺少标签
22	AUCUN_AXE_DISPONIBLE	没有可用的轴对象
23	AXES_TROP_NOMBREUX	配置中的轴过多
24	AXES_EN_DOUBLE	配置中的轴重复
25	AXE_INCORRECT	轴缺失或无效
26	AXE_INTROUVABLE	未找到轴对象或配置文件
27	NOMBRE_COORD_ERRONNE	值在轴上以不同的坐标数字表示
28	PERIPHERIQUE_HORS_COMMANDE	运动轴未激活
29	ERREUR_DEFAULT_MOUVEMENT	运动控制器上发生运动故障
2A	AXE_NON_ACTIF	未启用伺服驱动器
2B	DEPASSEMENT_DELAI_COMMANDE	命令超时
2C	BUS_SERCOS_INCORRECT	只能配置一个 SERCOS(r) 总线
2D	ERREUR_CHANGEMENT_NOM	轴重命名失败
2E	CMD_IMPOSSIBLE_AVEC_CETTE_CONFIG	无法在当前配置中执行命令
2F	CORRESPONDANCE_TYPE	对象类型错误
30	VARIATEUR_DOIT_ETRE_DESACTIVE	必须禁用伺服驱动器才能执行命令
31	VARIATEUR_DOIT_ETRE_ACTIVE	必须激活伺服驱动器才能执行命令
32	CMD_NON_AUTORISEE	此时不允许使用此命令
33	ERREUR_DEFAULT_VARIATEUR	由于伺服驱动器故障，无法完成命令

故障寄存器

SERCOS(r) 通道情况下的故障寄存器

下表提供了通道 0 情况下 MSB 的内容（十六进制表示的错误代码）以及故障寄存器：

错误代码	寄存器
0	REG_VERSION_AXE
1	REG_ID_AXE
2	REG_BITS_CONFIGURATION
3	REG_NOMBRE_AXES
4	REG_TEMPS_CYCLE
5	REG_DEBIT
6	REG_PUISSANCE_OPTIQUE
7	REG_MEMBRE_S1

独立轴情况下的故障寄存器

下表提供了独立轴情况下 MSB 的内容（十六进制表示的错误代码）以及故障寄存器：

错误代码	寄存器
0	REG_VERSION_AXE
1	REG_ID_AXE
2	REG_BITS_CONFIGURATION
3	REG_ADRESSE_SERCOS
4	REG_ACCELERATION
6	REG_DECELERATION
8	REG_TYPE_ACCELERATION
9	REG_FENETRE_AU_POINT
B	REG_FENETRE_DE_CONTROLE
D	REG_MODULO_MAX
F	REG_MODULO_MIN
11	REG_ACCELERATION_MAX
13	REG_DECELERATION_MAX
15	REG_VITESSE_MAX
17	REG_POSITION_MAX
19	REG_POSITION_MIN

错误代码	寄存器
1B	REG_NUMERATEUR_FACTEUR_ECHELLE
1D	REG_DENOMINATEUR_FACTEUR_ECHELLE
1F	REG_UNITES_ACCELERATION
20	REG_UNITES_VITESSE
21	REG_UNITES_POSITION
22	REG_RETOUR_RA

协同轴组情况下的故障寄存器

下表提供了协同轴组情况下 MSB 的内容（十六进制表示的错误代码）以及故障寄存器：

错误代码	寄存器
0	REG_VERSION_AXE
1	REG_ID_AXE
2	REG_BITS_CONFIGURATION
3	REG_NOMBRE_MEMBRES
4	REG_MEMBRE_C1
5	REG_MEMBRE_C2
6	REG_MEMBRE_C3
7	REG_MEMBRE_C4
8	REG_MEMBRE_C5
9	REG_MEMBRE_C6
A	REG_MEMBRE_C7
B	REG_MEMBRE_C8
C	REG_ACCELERATION
E	REG_DECELERATION
10	REG_TYPE_ACCELERATION
11	REG_ACCELERATION_MAX
13	REG_DECELERATION_MAX
15	REG_VITESSE_MAX
17	REG_UNITES_ACCELERATION
18	REG_UNITES_VITESSE

从轴组情况下的故障寄存器

下表提供了从轴组情况下 MSB 的内容（十六进制表示的错误代码）以及故障寄存器：

错误代码	寄存器
0	REG_VERSION_AXE
1	REG_ID_AXE
2	REG_BITS_CONFIGURATION
3	REG_NOMBRE_MEMBRES
4	REG_ID_MAITRE
5	REG_MEMBRE_S1
6	REG_MODE_SUIVEUR_S1
7	REG_CAME_OU_NUMERATEUR_S1
9	REG_OFFSET_OU_DENOMINATEUR_S1
B	REG_TRIGGER_S1
D	REG_MEMBRE_S2
E	REG_MODE_SUIVEUR_S2
F	REG_CAME_OU_NUMERATEUR_S2
11	REG_OFFSET_OU_DENOMINATEUR_S2
13	REG_TRIGGER_S2
15	REG_MEMBRE_S3
16	REG_MODE_SUIVEUR_S3
17	REG_CAME_OU_NUMERATEUR_S3
19	REG_OFFSET_OU_DENOMINATEUR_S3
1B	REG_TRIGGER_S3
1D	REG_MEMBRE_S4
1E	REG_MODE_SUIVEUR_S4
1F	REG_CAME_OU_NUMERATEUR_S4
21	REG_OFFSET_OU_DENOMINATEUR_S4
23	REG_TRIGGER_S4
25	REG_MEMBRE_S5
26	REG_MODE_SUIVEUR_S5
28	REG_CAME_OU_NUMERATEUR_S5
2A	REG_OFFSET_OU_DENOMINATEUR_S5
2C	REG_TRIGGER_S5
2E	REG_MEMBRE_S6

错误代码	寄存器
2F	REG_MODE_SUIVEUR_S6
30	REG_CAME_OU_NUMERATEUR_S6
32	REG_OFFSET_OU_DENOMINATEUR_S6
34	REG_TRIGGER_S6
36	REG_MEMBRE_S7
37	REG_MODE_SUIVEUR_S7
38	REG_CAME_OU_NUMERATEUR_S7
3A	REG_OFFSET_OU_DENOMINATEUR_S7
3C	REG_TRIGGER_S7
3E	REG_MEMBRE_S8
3F	REG_MODE_SUIVEUR_S8
40	REG_CAME_OU_NUMERATEUR_S8
42	REG_OFFSET_OU_DENOMINATEUR_S8
44	REG_TRIGGER_S8

WRITE_CMD 命令的错误代码列表

简介

%MWr.m.c.19 字指示显式写入 WRITE_CMD 命令时可能出现的错误。

轴控模块生成三种类型的错误:

- 编程错误 (参见第 357 页)
- 通讯错误 (参见第 359 页)
- 系统错误 (参见第 360 页)

WRITE_CMD 命令：编程错误

编程错误列表

下表列出编程错误的错误代码、名称和描述。错误代码 1 至 999 保留用于此类型的错误。

错误代码	描述
1	尝试分配超出范围的值
2	尝试分配不兼容的单位
3	单位不支持或未知
4	下载期间发生伺服驱动器故障
5	上载期间发生伺服驱动器故障
6	指向对象的意外空指针
7	未能设置伺服驱动器中的单位
8	单位未设置
9	字符串对 MotionString 来说太大
10	集合的索引无效
11	命令中的值不正确
12	序列中的值不正确
13	输入中的令牌无效
14	命令的反馈通道无效
15	命令的反馈外设无效
16	反馈时钟频率无效
17	反馈电源无效
18	反馈精度无效
19	保持寄存器地址无效
20	未配置保持寄存器数据库
24	无法授予对保持寄存器块的访问权限
25	无法释放对保持寄存器块的访问权限
26	文件打开失败
27	文件写入失败
28	文件读取失败
29	文件关闭失败
30	文件搜索失败
31	输入格式错误
32	清除故障功能无法清除故障

错误代码	描述
33	tags.cfg 中缺少标签
34	没有可用的轴对象
35	配置中的轴过多
36	配置中的轴重复
37	轴缺失或无效
38	未找到轴对象或配置文件
39	值在轴上以不同的坐标数字表示
40	运动轴未激活
41	运动控制器上发生运动故障
42	伺服驱动器未启用
43	命令超时
44	只能配置一个 SERCOS(r) 总线
45	轴重命名失败
46	无法在当前配置中执行命令
47	对象类型错误
48	必须禁用伺服驱动器才能执行命令
49	必须启用伺服驱动器才能执行命令
50	此时不允许对命令授权
51	由于伺服驱动器故障，无法完成命令
60	手动模式在与协调组或从轴组关联的轴上被拒绝
61	自动控制在手动模式下在轴上被拒绝
62	从轴组有凸轮配置文件从轴
63	从轴或主轴的减速度超过配置的最大减速度值
64	TRF_RECIPE 26200 被拒绝：实例已经启用
65	TRF_RECIPE 26200 被拒绝：报警阈值超过故障阈值
66	TRF_RECIPE 26200 被拒绝：阈值为负
67	TRF_RECIPE 26200 被拒绝：分母为零
68	TRF_RECIPE 16200 被拒绝：实例已禁用

WRITE_CMD 命令：通讯错误

通讯错误列表

下表列出了通讯错误的错误代码、名称和描述。错误代码 1000 至 1999 保留用于此类型的错误。

错误代码	描述
1000	目标未响应
1001	通讯错乱
1002	SERCOS(r) 错误
1003	没有来自驱动器的操作码回显
1004	SERCOS(r) 回路未就绪
1005	SERCOS(r) 错误
1006	SERCOS(r) 读取失败（循环通道）

WRITE_CMD 命令：系统错误

系统错误列表

下表列出了系统错误的错误代码、名称和描述。错误代码 2000 至 2999 保留用于此类型的错误。

错误代码	描述
2000	无法分类的错误
2001	组件中有错误
2002	组件中有错误
2003	调试中使用的系统故障
2004	驱动器得到不允许使用的命令故障
2005	驱动器得到非法命令故障
2006	驱动器出现编程错误故障
2007	命令 ID 无效
2008	无法将对象链接到驱动器接口
2009	信号量创建失败
2010	信号量删除失败
2011	信号量锁定失败
2012	信号量解锁失败
2013	信号量查询失败
2014	运动轴从未成功配置。
2015	命令尚未实施
2016	scx_unique 失败
2017	队列创建失败
2018	队列已满
2019	队列 ID 无效
2020	未知队列状态
2021	队列查询失败
2022	事件组创建失败
2023	事件组挂起错误
2024	事件组清除错误
2025	事件标志创建失败
2026	对象查找失败
2027	未找到对象管理器
2028	运动计划者状态无效

错误代码	描述
2029	存储器分配失败
2030	尝试从操作系统获取任务 ID 时出错
2031	试图安装过多的错误处理程序
2032	线程创建失败
2033	线程破坏器失败
2034	配置线程时出问题
2035	尝试挂起线程时出错
2036	尝试继续执行线程时出错
2037	创建对象时出错
4000	有效值电流故障
4001	驱动器过热情况
4002	驱动器过热情况
4005	解析器或编码器反馈故障
4006	一般驱动器故障（相位错误）
4007	驱动器短路故障
4009	驱动器电压故障
4011	跟随错误故障
4012	驱动器检测到通讯故障
4013	硬件行程终端故障
4015	回归、数字量输出或控制冲突故障（来自两个源）
4016	SERCOS(r) 主设备检测到通讯故障
5001	警戒时钟已失效，禁用所有轴

WRITE_CMD 命令：系统警告

系统警告列表

下表列出了系统警告的错误代码、名称和说明。

错误代码	说明
7001	程序无法识别此子程序编号
7002	轴 ID 对于此子程序无效
7003	参数数据超出范围
7004	SubNum/SubNumEcho 协议错误
7005	没有及时完成子程序
7010	程序无法识别此子程序编号
7011	在命令中输入了错误值
7777	调用了 Sample_User 子程序

TRF_RECIPe 命令显式写入故障

简介

%MWr.m.c.3 字指示显式写入 TRF_RECIPe 命令时的可能错误。

产生的故障与 WRITE_CMD 命令显式写入故障相同（请参见前几页），另外添加了几个特定于 TRF_RECIPe 命令的故障。

特定于 TRF_RECIPe 命令的错误列表

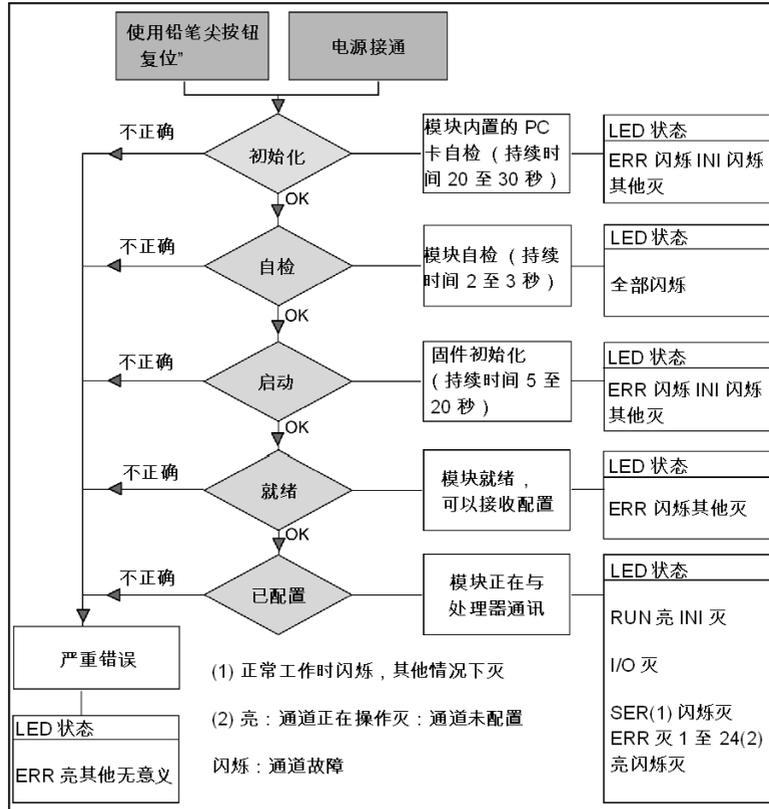
下表给出特定于 TRF_RECIPe 命令的错误的错误代码、名称和描述：

错误代码	名称	描述
6	TABLE_DOES_NOT_EXIST	凸轮配置文件被配置为固定的而不是可变的。无法读取或写入凸轮配置文件表。
19	PLC_CAPACITY_EXCEEDED	请求的传输超过 PLC 的 %MW 或 %KW 的容量。
22	LONGUEUR_INSUFFISANTE	与 TRF_RECIPe 关联的表对于要交换的数据来说不够大。

操作模式

操作模式概述

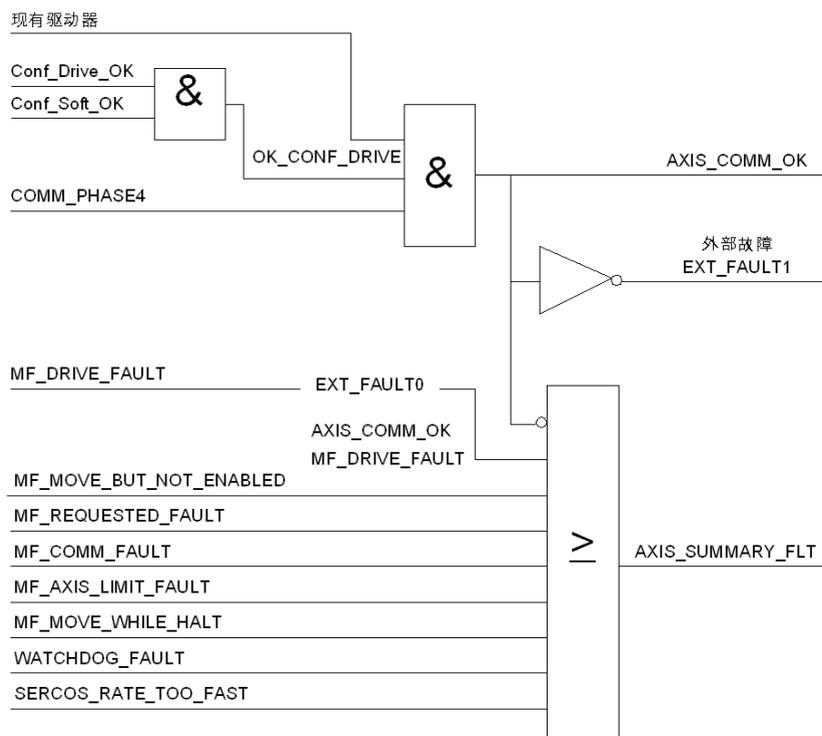
TSX CSY 84 / 164 模块的操作模式如下：



故障的逻辑图

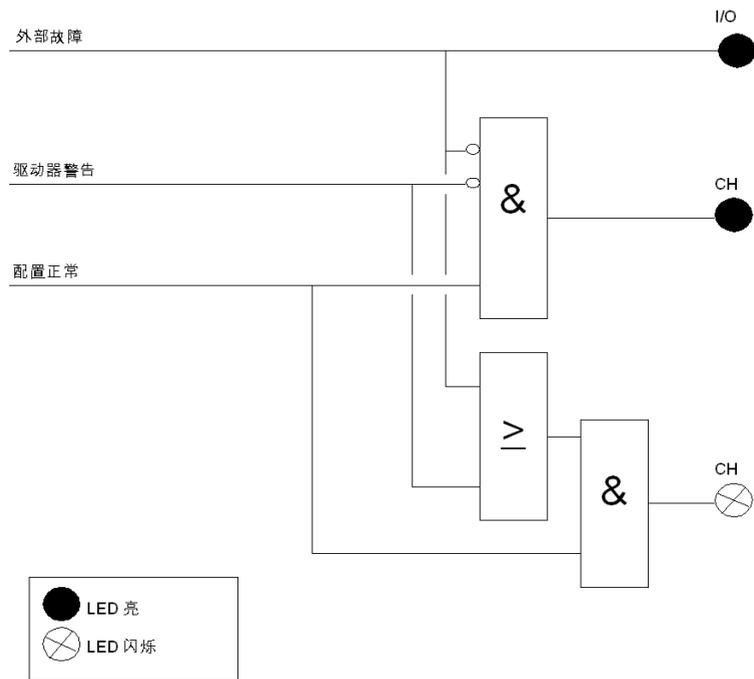
故障生成

下面的逻辑图说明了 AXIS_SUMMARY_FLT 和外部故障的生成：



LED 的激活

下面的逻辑图说明了 I/O 和 CH LED 的激活:



伺服驱动器故障

简介

可以使用某些功能将故障信息上载到项目。

这些功能是：

- 用于 IDN_11 的 GET_IDN
- 用于 IDN_12 的 GET_IDN
- 用于 IDN_13 的 GET_IDN

注意：有关 GET_IDN 功能的更多信息，请参见（*SERCOS (r)* IDN 读 / 写功能，第 301 页）。

"IDN11" 故障

下面的故障是对 IDN_11 执行 GET_IDN 功能的结果。

位	描述
0	由于过载而停止 (IDN 114)
1	放大器的停止温度 (IDN 203)
2	电机的停止温度 (IDN 204)
3	由于冷却故障而停止 (IDN 205)
4	命令电压错误
5	反馈错误
6	通讯系统错误
7	强度过大错误
8	过压错误
9	欠压错误
10	电源相错误
11	位置偏差过大 (IDN 159)
12	传送错误 (IDN 14)
13	超过限制 (IDN 49 和 50)
14	保留
15	特定制造商错误 (IDN 129)

"IDN11" 故障

下面的故障是对 IDN_12 执行 GET_IDN 功能的结果。

位	描述
0	过载报警 (IDN 314)
1	放大器过热报警 (IDN 311)
2	电机过热报警 (IDN 312)
3	由于冷却故障而报警 (IDN 313)
4 到 14	保留
15	特定制造商错误 (IDN 129)

"IDN13" 故障

下面的故障是对 IDN_13 执行 GET_IDN 功能的结果。

位	描述
0	N 返回状态 = N 命令 (IDN 330)
1	[返回值 N 的绝对值] 的状态 = 0 (IDN 331)
2	N 返回状态 < [Nx 的绝对值] (IDN 332)
3	[T 的绝对值] 的状态 >= [Tx 的绝对值] (IDN 333)
4	[T 的绝对值] 的状态 >= [T 的绝对限制值] (IDN 334)
5	[N 命令的绝对值] 的状态 >= [N 的绝对限制值] (IDN 335)
6	位置中的状态 (IDN 336)
7	[P 的绝对值] 的状态 <= [Px 的绝对值] (IDN 337)
8	保留
9	[实轴 N 的绝对值] 的状态 >= 最大心轴速度 (IDN 339)
10	[实轴 N 的绝对值] 的状态 >= 最大心轴速度 (IDN 340)
11 到 14	保留
15	特定制造商错误 (IDN 129)

15.6 TSX CSY 84 的性能

TSX CSY 84 的性能

概览

该模块的性能为：

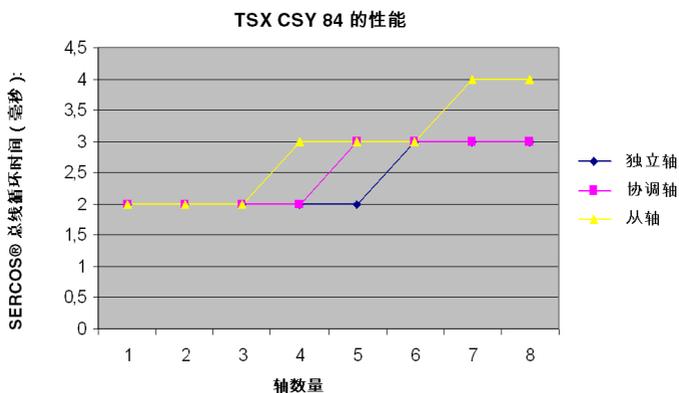
- SERCOS(r) 总线循环时间：
 - SERCOS(r) 单位循环时间 (tick) 缺省等于 4 毫秒
- 立即命令的处理 = 2 个 SERCOS(r) 单位循环时间 (tick)
- 排队命令的处理 = 2 个 SERCOS(r) 单位循环时间 (tick) * 排队命令的数量
- 远程测量的处理 = 2 个 SERCOS 单位循环时间 (tick) + 1 个用于确认从轴上参考命令的 SERCOS(r) 单位循环时间 (tick)
- 轴之间的偏差的监控周期：20 毫秒（与单位循环时间 (tick) 无关）
- 循环时间的动态修改（通过 Write-cycle 命令）将从以下值中选择：
 - SERCOS(r) 单位循环时间 (tick) 为 2 毫秒
 - SERCOS(r) 单位循环时间 (tick) 为 3 毫秒
 - SERCOS(r) 单位循环时间 (tick) 为 4 毫秒

在强制进入 SERCOS(r) 环路的阶段 0 的过程中需确认。

SERCOS(r) 单位循环时间 (tick) 是根据所配置的轴数量和应用类型来定义的（请参见 "SERCOS(r) 循环时间选择指南"）。

注意：Lexium 每隔 250 微秒执行一次内插。

SERCOS(r) 循环时间选择指南：



测试条件:

警告: 图中所示的性能是在一定条件下实现的, 这些条件保证大约 25% 的 SERCOS(r) CPU 时间缓冲。MAST 任务的周期配置为 20 毫秒。每隔 10 个 PLC 循环执行一次通过 MOVE_QUEUE 进行的运动命令传输。

结果,

- 对于 "独立轴" 配置: MOVE_QUEUE 每隔 200 毫秒发送到所有轴
- 对于 "从轴" 配置, MOVE_QUEUE 命令每隔 200 毫秒发送到主轴
- 对于 "协调轴" 配置, MOVE_QUEUE 命令每隔 200 毫秒发送到所有协调轴组

WRITE_PARAM 命令在此时间周期中发送, 以便执行参数调整。

使用程序修改循环时间

```
(* SERCOS (r) 循环时间更改 => %MW30 = 2000 或 3000 或 4000 *)
IF %M30 THEN      %MW101.0.26:=2565;
    %MD101.0.27:=%MW30;
    IF NOT %MW101.0:X1 THEN
        WRITE_CMD %CH101.0;
        RESET %M30;
        SET %M31;
    END_IF;
END_IF;

(* 转换到 SERCOS (r) 阶段 0 *)
IF %M31 THEN      %MW101.0.26:=2545;
    %MD101.0.27:=0;
    IF NOT %MW101.0:X1 THEN
        WRITE_CMD %CH101.0;
        RESET %M31;
        SET %M32;
    END_IF;
END_IF;

(* 到 SERCOS (r) 阶段 4 的新转换 *)
IF %M32 THEN      %MW101.0.26:=2545;
    %MD101.0.27:=4;
    IF NOT %MW101.0:X1 THEN
        WRITE_CMD %CH101.0;
        RESET %M32;
    END_IF;
END_IF;

(* 这里, 我们可以预期转换到阶段 4 (操作码 550 的结果为 %MD101.0.20),
然后确认转换到阶段 0 所导致的故障 *)
```

本章主题

本章描述与 SERCOS 模块关联的语言对象。

本章包含了哪些内容？

本章包含了以下部分：

节	主题	页
16.1	SERCOS 模块的语言对象和 IODDT	374
16.2	专门与 SERCOS 模块关联的语言对象和 IODDT	394
16.3	适用于所有模块的 IODDT 类型的 T_GEN_MOD	433

16.1 SERCOS 模块的语言对象和 IODDT

本节主题

本节介绍与 SERCOS 模块的语言对象和 IODDT 相关的一般信息。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
与 TSX CSY 84 模块关联的 IODDT 简介	375
与应用专用功能关联的隐式交换语言对象	376
与应用专用功能关联的显式交换语言对象	377
使用显式对象管理交换和报告	379
接口语言	383
参数管理	385
WRITE_PARAM 和 READ_PARAM	386
SAVE_PARAM 和 RESTORE_PARAM	387
WRITE_CMD	388
WRITE_CMD 示例	390
READ_STS	392
TRF_RECIPE 和 MOD_PARAM	393

与 TSX CSY 84 模块关联的 IODDT 简介

一般信息

IODDT 是由制造商预定义的。它们包括属于应用专用模块通道的输入 / 输出语言对象。

TSX CSY 84 模块有 7 种类型的 IODDT:

- T_CSY_CMD
- T_CSY_RING
- T_CSY_TRF
- T_CSY_IND
- T_CSY_FOLLOW
- T_CSY_COORD
- T_CSY_CAM

注意: 存在两种创建 IODDT 变量的方法:

- "I/O 对象" 选项卡 (参见 *Unity Pro, 操作模式* ,)
- 数据编辑器 (参见 *Unity Pro, 操作模式* ,)

语言对象类型

每种 IODDT 都包含一组可用于控制和检查它们如何操作的语言对象。

语言对象有两种类型:

- **隐式交换对象**, 在每个与模块关联的任务循环中自动交换,
- **显式交换对象**, 在项目请求时使用显式交换指令交换。

隐式交换涉及模块的状态、通讯信号、从轴等。

显式交换用于设置模块的参数和执行模块诊断。

与应用专用功能关联的隐式交换语言对象

概览

集成的应用专用接口或额外的模块可以自动增强用于对此接口或模块进行编程的语言对象应用。

这些对象对应于输入 / 输出图像和模块或集成应用专用接口的软件数据。

提示

当 PLC 处于运行或停止模式时，将在任务开始时，在 PLC 存储器中更新模块输入（%I 和 %IW）。

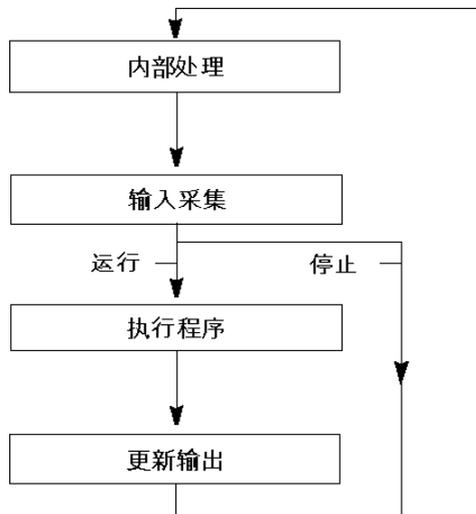
仅当 PLC 处于运行模式时，才会在任务结束时更新模块输出（%Q 和 %QW）。

注意：如果任务运行于停止模式，则根据所选配置的不同，可能出现以下两种情况之一：

- 输出设置为故障预置位置（故障预置模式），
- 输出保持其最后的值（维护模式）。

图

下图显示了 PLC 任务的操作循环（循环执行）。



与应用专用功能关联的显式交换语言对象

简介

显式交换是应用户程序的请求，使用以下指令执行的交换：

- READ_STS (参见 *Unity Pro, I/O 管理, 功能块库*) (读取状态字)
- WRITE_CMD (参见 *Unity Pro, I/O 管理, 功能块库*) (写入命令字)
- WRITE_PARAM (参见 *Unity Pro, I/O 管理, 功能块库*) (写入调整参数)
- READ_PARAM (参见 *Unity Pro, I/O 管理, 功能块库*) (读取调整参数)
- SAVE_PARAM (参见 *Unity Pro, I/O 管理, 功能块库*) (保存调整参数)
- RESTORE_PARAM (参见 *Unity Pro, I/O 管理, 功能块库*) (恢复调整参数)

这些交换适用于属于一个通道的一组相同类型的 %MW 对象 (状态、命令或参数)。

注意：

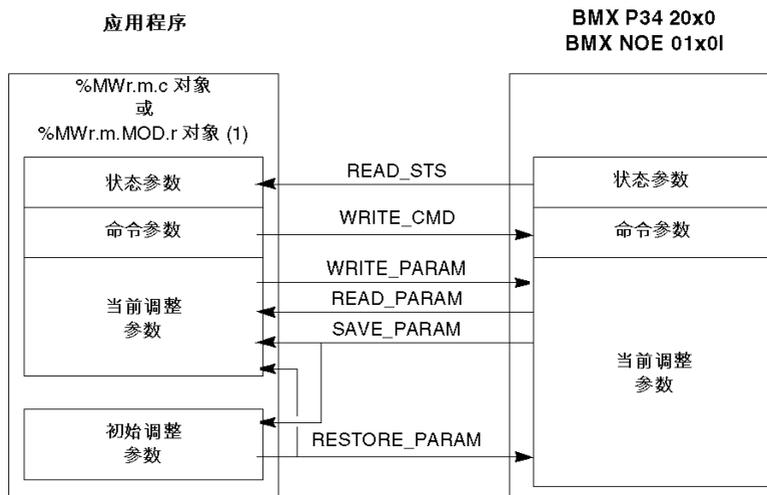
这些对象可以：

- 提供有关模块的信息 (如通道故障类型)
- 可以使用命令 (如切换命令) 控制模块
- 定义模块的操作模式 (在应用程序执行过程中保存和恢复调整参数)

注意： 为了避免同一通道同时发生多个显式交换，在调用对此通道寻址的任何 EF 之前，需要测试与该通道关联的 IODDT 的 EXCH_STS (%MW_{r.m.c.0}) 字的值。

使用显式指令的一般原则

下图显示了可以在应用程序和模块之间执行的各种类型的显式交换。



(1) 仅适用于 READ_STS 和 WRITE_CMD 指令。

管理交换

在显式交换期间，需要检查性能，以确保只在正确执行交换后才考虑数据。

为此提供了以下两种类型的信息：

- 与正在进行的交换有关的信息 (参见第 381 页)
- 交换报告 (参见第 382 页)

下图说明了管理交换的原理。



注意： 为了避免同一通道同时发生多个显式交换，在调用对此通道寻址的任何 EF 之前，需要测试与该通道关联的 IODDT 的 EXCH_STS (%MW_r.m.c.0) 字的值。

使用显式对象管理交换和报告

概览

当在 PLC 存储器与模块之间交换数据时，模块可能需要多个任务循环以确认此信息。所有 IO DDT 均使用以下两个字来管理交换：

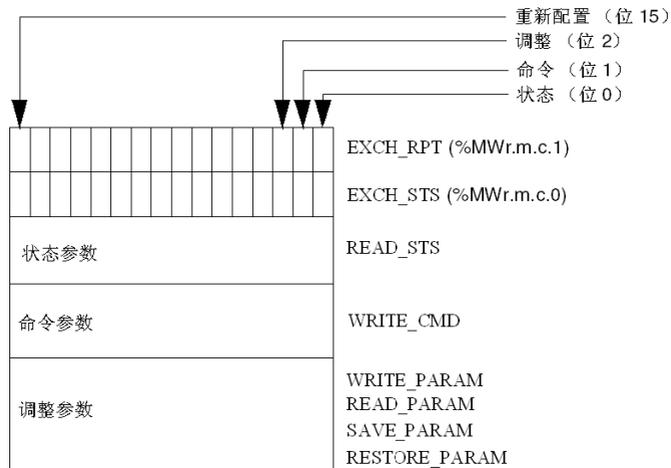
- EXCH_STS (%MWr.m.c.0)：正在交换
- EXCH_RPT (%MWr.m.c.1)：报告

注意：根据模块的位置，应用程序将检测不到显式交换的管理（例如，%MW0.0.MOD.0.0）：

- 对于机架内的模块，显式交换将立即在本地 PLC 总线上执行并在任务执行结束之前完成，因此，举例来说，当应用程序检查 %MW0.0.mod.0.0 位时，READ_STS 始终为已完成。
- 对于远程总线（例如，Fipio），显式交换与执行任务并不同步，因此应用程序可以进行检测。

示意图

下图显示了用于管理交换的各个有效位：



有效位的描述

字 EXCH_STS (%MW_r.m.c.0) 和 EXCH_RPT (%MW_r.m.c.1) 的每一位分别与一类参数关联:

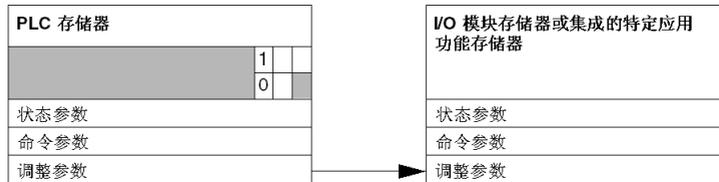
- 序号为 0 的位与状态参数关联:
 - STS_IN_PROGR 位 (%MW_r.m.c.0.0) 指示状态字的读请求是否正在进行,
 - STS_ERR 位 (%MW_r.m.c.1.0) 指定状态字的读请求是否被模块通道接受。
- 序号为 1 的位与命令参数关联:
 - CMD_IN_PROGR 位 (%MW_r.m.c.0.1) 指示命令参数是否正发送到模块通道,
 - CMD_ERR 位 (%MW_r.m.c.1.1) 指定命令参数是否被模块通道接受。
- 序号为 2 的位与调整参数关联:
 - ADJ_IN_PROGR 位 (%MW_r.m.c.0.2) 指示是否正在与模块通道交换调整参数 (通过 WRITE_PARAM、READ_PARAM、SAVE_PARAM、RESTORE_PARAM),
 - ADJ_ERR 位 (%MW_r.m.c.1.2) 指定调整参数是否被模块接受。如果交换正确执行, 则该位设置为 0。
- 序号为 15 的位指示从控制台对模块的通道 **c** 进行重新配置 (修改配置参数并对通道进行冷启动)。
- *r*、*m* 和 *c* 位表示以下元素:
 - **r** 位表示机架编号
 - **m** 位表示模块在机架中的位置
 - **c** 位表示模块中的通道编号

注意: **r** 表示机架编号, **m** 表示模块在机架中的位置, 而 **c** 表示模块中的通道编号。

注意: 根据 IODDT 类型 T_GEN_MOD, 模块级也存在交换字和报告字 EXCH_STS (%MW_r.m.MOD) 和 EXCH_RPT (%MW_r.m.MOD.1)。

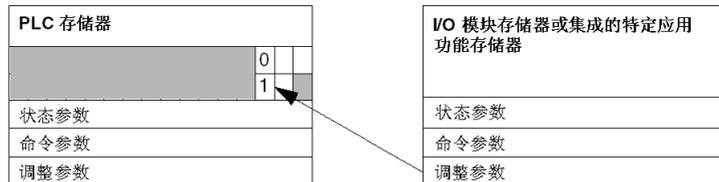
示例

阶段 1: 使用 `WRITE_PARAM` 指令发送数据。



当 PLC 处理器扫描到指令时，`%MWr.m.c` 中的正在交换位设置为 1。

阶段 2: 通过 I/O 模块和报告分析数据。



当在 PLC 存储器与模块之间交换数据时，模块的确认由 `ADJ_ERR` 位 (`%MWr.m.c.1.2`) 管理。

该位报告以下情况：

- 0: 交换正确
- 1: 交换错误

注意： 模块级没有调整参数。

显式交换的执行指示器：EXCH_STS

下表显示了显式交换的控制位：EXCH_STS (`%MWr.m.c.0`)。

标准符号	类型	访问	含义	地址
STS_IN_PROGR	BOOL	R	正在读取通道状态字	<code>%MWr.m.c.0.0</code>
CMD_IN_PROGR	BOOL	R	正在进行命令参数交换	<code>%MWr.m.c.0.1</code>
ADJ_IN_PROGR	BOOL	R	正在进行调整参数交换	<code>%MWr.m.c.0.2</code>
RECONF_IN_PROGR	BOOL	R	正在重新配置模块	<code>%MWr.m.c.0.15</code>

注意： 如果模块不存在或已断开连接，则不会将显式交换对象（如 `READ_STS`）发送到模块 (`STS_IN_PROG` (`%MWr.m.c.0.0`) = 0)，但会刷新字。

显式交换报告：EXCH_RPT

下表显示了报告位：EXCH_RPT (%MWr.m.c.1)。

标准符号	类型	访问	含义	地址
STS_ERR	BOOL	R	读取通道状态字时出错 (1 = 故障)	%MWr.m.c.1.0
CMD_ERR	BOOL	R	交换命令参数时出错 (1 = 故障)	%MWr.m.c.1.1
ADJ_ERR	BOOL	R	交换调整参数时出错 (1 = 故障)	%MWr.m.c.1.2
RECONF_ERR	BOOL	R	重新配置通道时出错 (1 = 故障)	%MWr.m.c.1.15

Modicon M340 计数模块用途

下表描述电源接通后在 Modicon M340 计数模块与系统之间实现的步骤。

步骤	操作
1	电源接通
2	系统发送配置参数。
3	系统通过 WRITE_PARAM 方法发送调整参数。 注： 操作完成时，位 %MWr.m.c.0.2 将切换为 0。

如果在应用程序开始时使用 WRITE_PARAM 命令，则必须等待位 %MWr.m.c.0.2 切换为 0。

接口语言

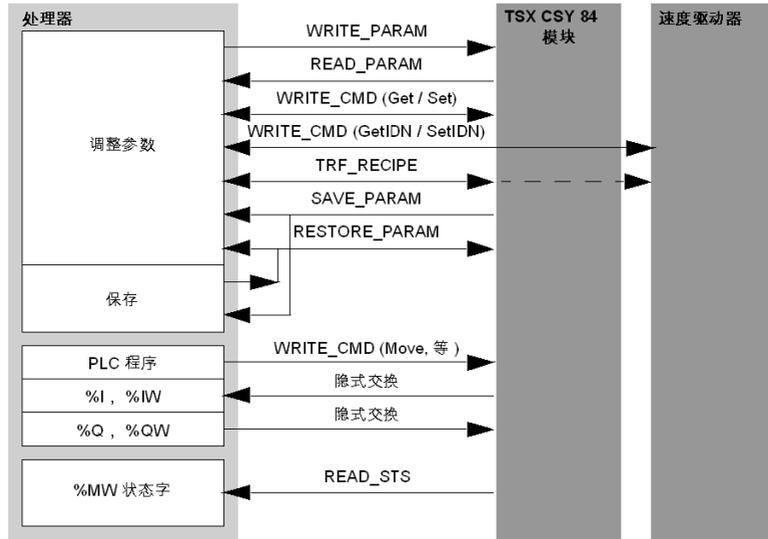
模块通道

TSX CSY 84 模块最多由 32 个通道构成，这些通道支持以下功能：

通道	支持的功能
%CHR.m.0	SERCOS(r) 功能
%CHR.m.1 至 %CHR.m.8	实轴
%CHR.m.9 至 %CHR.m.12	虚轴
%CHR.m.13 至 %CHR.m.16	远程轴
%CHR.m.17 至 %CHR.m.20	协同轴组
%CHR.m.21 至 %CHR.m.24	从轴组
%CHR.m.25 至 %CHR.m.31	凸轮配置文件

交换概述

处理器、轴控模块和变数控制器之间的交换按如下方式执行：



参数管理

一般信息

轴控模块的参数和伺服驱动器的参数通过 Unity Pro 和 UniLink 单独进行管理。例如，可以在轴控模块和伺服驱动器中定义不同的限制值。

轴控模块的参数可划分为两类：

- 仅由轴控模块使用的参数
- 同时由轴控模块和伺服驱动器使用的参数

注意：模块 TSX CSY 84/164 不支持用线性类型的单位操作的伺服驱动器。

轴控模块参数

"C"（控制器）类型的参数作为浮点数以轴控模块单位进行交换。这些参数与路径计算或凸轮配置文件相关。它们在 Unity Pro 配置屏幕中配置，然后由应用程序根据需要修改。这些参数的单位是在配置屏幕中定义的命令轴模块的单位。

伺服驱动器参数

当应用程序修改轴控模块中的参数时，伺服驱动器中的对应参数不会更新（例如，最小和最大位置限制、最小和最大加速度、最高速度、调试窗口）。

这些参数被认为是伺服驱动器系统参数。用户通过 UniLink 配置它们以便保护操作部件，并且没有必要在操作时更改它们（应用程序修改命令轴模块中的等同 "C" 类型参数）。

同时由轴控模块和伺服驱动器使用的参数

轴控模块不会根据这些等同参数值更新伺服驱动器参数。

"S" 类型的参数（SERCOS(r) 标准标识符 (IDN)）和 "P" 类型的参数（SERCOS(r) 所有者标识符 (IDN)）作为浮点数与轴控模块进行交换，而与伺服驱动器进行交换时可以使用任何数据类型（例如，加速度、速度和位置值等参数）。

如果应用程序要求将轴控模块参数的值与伺服驱动器参数同步，它可以读取 "S" 或 "P" 类型的参数 (TRF_RECIPE)，然后将它们写入 "C" 类型的参数 (WRITE_PARAM)。

"单位" 参数

单位是同时由轴控模块和伺服驱动器使用的参数。

单位通过 Unity Pro 进行配置。轴控模块单位到伺服驱动器单位的转换由模块 TSX XSY 84/164 使用用户配置的比率系数自动执行。

WRITE_PARAM 和 READ_PARAM

提示

这些服务允许在处理器（项目）与轴控模块之间交换调整参数。

READ_PARAM：通过 IODDT 显式读取轴控模块中的参数和更新调整字 %MW/D/Fr.m.c.r.。

WRITE_PARAM：显式写入轴控模块中的参数。此指令允许以编程方式修改配置中定义的调整值。

这两个指令适用于与 TSX CSY 84 模块关联的 IODDT 变量。下面几个段落将以一个名为 `Sercos_Channel`、类型为 **T_CSY_IND** 的变量为例。

指令 READ_PARAM 的语法

READ_PARAM (`Sercos_Channel`)：读取与 IODDT `Sercos_Channel` 关联的通道的调整参数。

指令 WRITE_PARAM 的语法

WRITE_PARAM (`Sercos_Channel`)：写入与 IODDT `Sercos_Channel` 关联的通道的调整参数。

监控交换

IODDT 的以下两个位可用于监控处理器与模块之间的调整参数交换：

标准符号	含义	位
ADJ_IN_PROGR	当交换正在进行时，此位设置为 1。当交换已完成时，此位复位为 0。	%MWr.m.c.0.2
ADJ_ERR	如果传送的参数超出范围或有错误，则此位设置为 1。	%MWr.m.c.1.2

SAVE_PARAM 和 RESTORE_PARAM

提示

这些服务可用于保存或恢复调整参数。

SAVE_PARAM : 显式备份轴控模块参数。这些参数取代配置期间定义的初始值。

RESTORE_PARAM : 显式恢复初始调整参数 (在配置时或上次备份时写入)。

这两个指令适用于与 TSX CSY 84 模块关联的 IODDT 变量。下面几个段落将以一个名为 Sercos_Channel 、类型为 **T_CSY_IND** 的变量为例。

SAVE_PARAM 指令的语法

SAVE_PARAM (Sercos_Channel) : 保存与 IODDT Sercos_Channel 关联的通道的调整参数。

RESTORE_PARAM 指令的语法

RESTORE_PARAM (Sercos_Channel) : 恢复与 IODDT Sercos_Channel 关联的通道的调整参数。

监控交换

IODDT 的以下两个位可用于监控处理器与模块之间的调整参数交换:

标准符号	含义	位
ADJ_IN_PROGR	当交换正在进行时, 此位设置为 1。当交换已完成时, 此位复位为 0。	%MWr.m.c.0.2
ADJ_ERR	如果传递的参数超出范围或有错误, 则此位设置为 1。	%MWr.m.c.1.2

WRITE_CMD

提示

通过此服务可以将命令发送到运动控制器。

WRITE_CMD：显式写入模块中的命令字。此操作通过内部字 %MW 执行，该内部字包含要执行的命令及其参数（例如某个运动控制）。

此指令适用于与 TSX CSY 84 和 TSX CSY 164 模块关联的 IODDT 类型的变量。对于独立轴，我们将以一个名为 `Sercos_Channel`、类型为 **T_CSY_IND** 的变量为例。对于协同轴，我们将以一个名为 `Sercos_Channel_coord`、类型为 **T_CSY_COORD** 的变量为例。

WRITE_CMD 指令的语法

WRITE_CMD (`Sercos_Channel`)：写入与 IODDT `Sercos_Channel`（协同轴）关联的通道的命令信息。

WRITE_CMD (`Sercos_Channel_coord`)（对于协同轴）。

WRITE_CMD 接口

要执行的命令在 `Sercos_Channel` IODDT 的 `ACTION_CMD` 字 (%MW.r.m.c.26) 中定义，命令的结果在下表描述的字中可用：

标准符号	类型	访问	含义	地址
ERROR_CMD	INT	读写	WRITE_CMD 命令写入错误	%MW.r.m.c.19
RETURN_CMD_1	DINT	读写	功能的返回值 1	%MDr.m.c.20
RETURN_CMD_2	REAL	读写	功能的返回值 2	%MFr.m.c.22
RETURN_CMD_3	REAL	读写	功能的返回值 3	%MFr.m.c.24
ACTION_CMD	INT	读写	要执行的动作	%MW.r.m.c.26
PARAM_CMD_1	DINT	读写	参数 1	%MDr.m.c.27
PARAM_CMD_2	DINT	读写	参数 2	%MDr.m.c.29
PARAM_CMD_3	REAL	读写	参数 3	%MFr.m.c.31
PARAM_CMD_4	REAL	读写	参数 4	%MFr.m.c.33

协同轴组情况下的 WRITE_CMD 接口

在协同轴组的情况下，运动功能使得向每个协同轴发送 2 个参数（位置和速度）变得必要。因此与此轴关联的 IODDT 更复杂一些。命令的结果和参数位于 Sercos_Channel_Coord IODDT 的以下字中：

标准符号	类型	访问	含义	地址
...
PARAM_CMD_1	DINT	读写	参数 1	%MDr.m.c.27
PARAM_CMD_2	DINT	读写	参数 2	%MDr.m.c.29
PARAM_CMD_3	REAL	读写	参数 3	%MFr.m.c.31
PARAM_CMD_4	REAL	读写	参数 4	%MFr.m.c.33
PARAM_CMD_5	REAL	读写	参数 5	%MFr.m.c.35
...
PARAM_CMD_18	REAL	读写	参数 18	%MFr.m.c.61

监控交换

IODDT 的以下两个位可用于监控模块中的命令信息的写入：

标准符号	含义	位
CMD_IN_PROGR	当交换正在进行时，此位设置为 1。当交换已完成时，此位复位为 0。	%MWr.m.c.0.1
CMD_ERR	如果传送的参数超出范围或有错误，则此位设置为 1。	%MWr.m.c.1.1

WRITE_CMD 示例

示例 1: 初始化故障

为了初始化机架 1 中的模块 4 的总线环路上的故障：将 Ring IODDT 声明为 T_CSY_RING 类型，并将其关联到机架 1 中插槽 4 上的 TSX CSY 84 或 TSX CSY 164 模块的通道 0:

```
! (* 如果 WRITE_CMD 不在进行中，则初始化故障 *)
  IF NOT Ring.CMD_IN_PROGR THEN Ring.ACTION_CMD:= 409;
  WRITE_CMD(Ring);
  END_IF;
```

示例 2: 写入光功率

定义（写入）光功率。我们使用同一 IODDT:

```
! (* 如果 WRITE_CMD 不在进行中，则写入（设置）光功率 *)
  IF NOT Ring.CMD_IN_PROGR THEN Ring.ACTION_CMD:= 2545;
  Ring.PARAM_CMD_3:= 51.5;      // 参数 3: 光功率 = 51.5
  WRITE_CMD(Ring);
  END_IF;
```

示例 3: 读取发射比率

读取发射比率 (BAUD_RATE):

```
! (* 如果 WRITE_CMD 不在进行中，则读取（获取）发射比率 *)
  IF NOT Ring.CMD_IN_PROGR THEN Ring.ACTION_CMD:= 1551;
  WRITE_CMD(Ring);
  END_IF;
! (* 如果 WRITE_CMD 已完成并且没有故障，则发射比率的值可在返回值 1 中访问 *)
  IF NOT Ring.CMD_IN_PROGR AND NOT Ring.CMD_ERR
  THEN BAUD_RATE:=                // 返回值 1 中的发射比率
  Ring.RETURN_CMD_1;
  END_IF;
```

示例 4: 运动实轴

用命令 "MOVE absolute immediate" 以速度 5 运动位于机架 1 中位置 105.2 的模块 4 中的实轴 3。我们声明 T_CSY_IND 类型的 Axis_3 IODDT，并将其与机架 1 的模块 4 的通道 3 关联：

```
! (* 如果 WRITE_CMD 不在进行中，则运动实轴 3*)
  IF NOT Axis_3.CMD_IN_PROGR THEN Axis_3.ACTION_CMD:= 513;
  Axis_3.PARAM_CMD_1:= 0;           // 参数 1: 运动的类型 = 绝对
  Axis_3.PARAM_CMD_3:= 105.2;      // 参数 3: 位置 = 105.2
  Axis_3.PARAM_CMD_4:= 5.0;       // 参数 4: 速度 = 5
  WRITE_CMD(Axis_3);
  END_IF;
```

示例 5: 读取从轴的位置

读取主轴在 102.5 时的跟随器从轴的相对位置。我们声明 T_CSY_FOLLOW 类型的 IODDT Follow，并将其与对应于跟随器从轴的通道关联：

```
! (* 如果 WRITE_CMD 不在进行中，则读取跟随器从轴的位置 *)
  IF NOT Follow.CMD_IN_PROGR THEN Follow.ACTION_CMD:= 537;
  Follow.PARAM_CMD_3:= 102.5;      // 参数 3: 主轴的位置 = 102.5
  WRITE_CMD(Follow);
  END_IF;
! (* 如果 WRITE_CMD 已完成并且没有故障，则从轴的位置可在返回值 2 中访问 *)
  IF NOT Follow.CMD_IN_PROGR AND NOT Follow.CMD_ERR
  THEN
    SLAVE_POSITION:=Follow.RETURN_CM // 返回值 2 中的位置
    D_2;
  END_IF;
```

READ_STS

提示

通过此服务可以显式读取与轴控模块（或不同通道）关联的状态字。

此指令可应用于能够与 TSX CSY 84 或 TSX CSY 164 模块关联的所有类型的 IODDT。对于独立轴，我们将以一个名为 `Serco_Channel`、类型为 `T_CSY_IND` 的变量为例。

还可将此指令应用于 TSX CSY 84 或 TSX CSY 164 模块。在此情况下，必须将其应用于 `T_CSY_CMD` 类型的 IODDT。让我们使用名为 `Sercos_module` 的变量示例。

READ_STS（模块）指令的语法

`READ_STS (Sercos_Module)`：从与 `Serco_Module` IODDT 关联的模块读取常规诊断信息。

READ_STS（通道）指令的语法

`READ_STS (Sercos_Channel)`：从与 `Sercos_Channel` IODDT 关联的通道读取常规诊断信息。

监控交换

IODDT 的以下两个位可用于监控处理器和模块之间的状态字交换：

标准符号	含义	位
STS_IN_PROGR	当读取正在进行时，此位设置为 1。当交换已完成时，此位复位为 0。	%MWr.m.c.0.0
STS_ERR	此位提供交换报告。它在发生故障时设置为 1。	%MWr.m.c.1.0

TRF_RECIPE 和 MOD_PARAM

提示

- **TRF_RECIPE:**

此指令在本手册中说明 (参见第 292 页)。

- **MOD_PARAM:**

此指令在本手册中说明 (参见第 462 页)。

16.2 专门与 SERCOS 模块关联的语言对象和 IODDT

本节主题

本节介绍与 SERCOS 模块关联的语言对象和 IODDT。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
有关 T_CSY_CMD 类型的 IODDT 隐式交换对象的详细信息	395
有关 T_CSY_CMD 类型的 IODDT 显式交换对象的详细信息	397
有关 T_CSY_RING 类型的 IODDT 隐式交换对象的详细信息	399
有关 T_CSY_RING 类型的 IODDT 显式交换对象的详细信息	401
有关 T_CSY_TRF 类型的 IODDT 显式交换对象的详细信息	404
有关 T_CSY_IND 类型的 IODDT 隐式交换对象的详细信息	406
有关 T_CSY_IND 类型的 IODDT 显式交换对象的详细信息	410
有关 T_CSY_FOLLOW 类型的 IODDT 隐式交换对象的详细信息	414
有关 T_CSY_FOLLOW 类型的 IODDT 显式交换对象的详细信息	417
有关 T_CSY_COORD 类型的 IODDT 隐式交换对象的详细信息	424
有关 T_CSY_COORD 类型的 IODDT 显式交换对象的详细信息	426
有关 T_CSY_CAM 类型的 IODDT 隐式交换对象的详细信息	429
有关 T_CSY_CAM 类型的 IODDT 显式交换对象的详细信息	430

有关 T_CS_Y_CMD 类型的 IO DDT 隐式交换对象的详细信息

概览

T_CS_Y_CMD IO DDT 拥有隐式交换对象，如下所述。此类 IO DDT 适用于 TSX_CS_Y_84 模块。

变量声明示例：IO DDT_VAR1 ，类型为 T_CS_Y_CMD

各个位的含义一般是针对该位的状态 1 给出的。在特定情况下，将解释两个位状态。

隐式交换输入对象列表

下表介绍适用于 TSX_CS_Y_84 模块的 T_CS_Y_CMD IO DDT 隐式交换输入对象。

标准符号	类型	访问	含义	地址
CH_ERROR	EBOOL	读	通道错误位	%lr.m.c.ERR
PROFILE_END	EBOOL	读	最后的配置文件命令已发送到模块	%lr.m.c.3
IN_POSITION	EBOOL	读	轴位于正确位置带中	%lr.m.c.4
AXIS_HOMED	EBOOL	读	在主位置引用了轴位置读数	%lr.m.c.6
HOLDING	EBOOL	读	轴保持在等待位置	%lr.m.c.8
RESUMING	EBOOL	读	轴在等候状态后一直在运动	%lr.m.c.9
DRIVE_ENABLED	EBOOL	读	驱动器已启用	%lr.m.c.10
DRIVE_FLT	EBOOL	读	驱动器正在执行 1 类诊断	%lr.m.c.13
AXIS_SUMMARY_FLT	EBOOL	读	驱动器故障	%lr.m.c.15
AXIS_IN_CMD	EBOOL	读	轴是活动的且可控制	%lr.m.c.18
AXIS_HOLD	EBOOL	读	轴已停止，正在等待命令	%lr.m.c.28
AXIS_HALT	EBOOL	读	轴已停止	%lr.m.c.29
AXIS_FASTSTOP	EBOOL	读	轴已快速停止	%lr.m.c.30
AXIS_READY	EBOOL	读	轴已准备响应命令	%lr.m.c.31

隐式交换输出对象列表

下表介绍适用于 TSX_CSX_84 模块的 T_CSX_CMD IODDT 隐式交换输出对象。

标准符号	类型	访问	含义	地址
CONTROL_ACQUIRE	EBOOL	读写	控制采集	%Qr.m.c.2
CONTROL_ENABLE	EBOOL	读写	控制启用	%Qr.m.c.10
CONTROL_RESUME	EBOOL	读写	停止后恢复控制	%Qr.m.c.12
CONTROL_CLEAR_FLT	EBOOL	读写	故障清除控制	%Qr.m.c.15
ALLOW_ACQUIRE	EBOOL	读写	采集启用控制	%Qr.m.c.18
ALLOW_ENABLE	EBOOL	读写	禁用轴控	%Qr.m.c.26
ALLOW_RESUME	EBOOL	读写	授权运动在使用 HOLD 命令停止后继续	%Qr.m.c.28
ALLOW_MOVE	EBOOL	读写	授权运动在使用 HALT 命令停止后继续	%Qr.m.c.29

有关 T_CSY_CMD 类型的 IODDT 显式交换对象的详细信息

概览

本部分介绍适用于 TSX CSY 84 模块的 T_CSY_CMD IODDT 的显式交换对象。它对字类型的对象分组，这些对象的位具有特殊含义。下面将详细介绍这些对象。

变量声明示例：IODDT_VAR1，类型为 T_CSY_CMD

注：

- 各个位的含义一般是针对该位的状态 1 给出的。在特定情况下，将解释每个位状态。
- 并未使用所有位。

显式交换执行的指示器：EXCH_STS

下表解释了 EXCH_STS 通道 (%MWr.m.c.0) 的各个交换控制位的含义。

标准符号	类型	访问	含义	地址
CMD_IN_PROGR	BOOL	读	正在进行命令参数交换	%MWr.m.c.0.1

显式交换报告：EXCH_RPT

下表解释了 EXCH_RPT 报告位 (%MWr.m.c.1) 的含义。

标准符号	类型	访问	含义	地址
CMD_ERR	BOOL	读	在交换命令参数期间发生故障	%MWr.m.c.1.1

WRITE_CMD 接口字

下表解释与 WRITE_CMD （其动作在字 ACTION_CMD 中指定）关联的变量的含义。这些变量由 WRITE_CMD (IODDT_VAR1) 更新。

标准符号	类型	访问	含义	地址
ERROR_CMD	INT	读写	WRITE_CMD 出错	%MWr.m.c.19
RETURN_CMD_1	DINT	读写	功能的返回值 1	%MDr.m.c.20
RETURN_CMD_2	REAL	读写	功能的返回值 2	%MFr.m.c.22
RETURN_CMD_3	REAL	读写	功能的返回值 3	%MFr.m.c.24
ACTION_CMD	INT	读写	要执行的动作	%MWr.m.c.26
PARAM_CMD_1	DINT	读写	参数 1	%MDr.m.c.27
PARAM_CMD_2	DINT	读写	参数 2	%MDr.m.c.29
PARAM_CMD_3	REAL	读写	参数 3	%MFr.m.c.31
PARAM_CMD_4	REAL	读写	参数 4	%MFr.m.c.33

有关 T_CS_Y_RING 类型的 IO DDT 隐式交换对象的详细信息

概览

T_CS_Y_RING IO DDT 拥有隐式交换对象，如下所述。此类 IO DDT 适用于 TSX_CS_Y_84 模块的通道 0。

变量声明示例：IO DDT_VAR1，类型为 T_CS_Y_RING

各个位的含义一般是针对该位的状态 1 给出的。在特定情况下，将解释两个位状态。

隐式交换输入对象列表

下表介绍适用于 TSX_CS_Y_84 模块的 T_CS_Y_RING IO DDT 隐式交换输入对象。

标准符号	类型	访问	含义	地址
CH_ERROR	EBOOL	读	通道错误位	%r.m.c.ERR
RAMPING	EBOOL	读	指示轴是在加速还是在减速	%r.m.c.0
STEADY	EBOOL	读	速度稳定	%r.m.c.1
STOPPING	EBOOL	读	运动正在减速至停止	%r.m.c.2
PROFILE_END	EBOOL	读	最后的配置文件命令已发送到模块	%r.m.c.3
IN_POSITION	EBOOL	读	轴位于正确位置带中	%r.m.c.4
AXIS_HOMING	EBOOL	读	轴正在回归。对于虚轴，此位停用	%r.m.c.5
AXIS_HOMED	EBOOL	读	在主位置以外引用了轴位置读数	%r.m.c.6
AXIS_NOT_FOLLOWING	EBOOL	读	驱动器不识别模块命令	%r.m.c.7
HOLDING	EBOOL	读	轴保持在等待位置	%r.m.c.8
RESUMING	EBOOL	读	轴在等候状态后一直在运动	%r.m.c.9
DRIVE_ENABLED	EBOOL	读	驱动器已启用	%r.m.c.10
DRIVE_DIAG	EBOOL	读	驱动器正在执行 3 类诊断	%r.m.c.11
DRIVE_WARNING	EBOOL	读	驱动器正在执行 2 类诊断	%r.m.c.12
DRIVE_FLT	EBOOL	读	驱动器正在执行 1 类诊断	%r.m.c.13
DRIVE_DISABLED	EBOOL	读	驱动器已禁用	%r.m.c.14
AXIS_SUMMARY_FLT	EBOOL	读	驱动器故障	%r.m.c.15
AXIS_COM_OK	EBOOL	读	驱动器与模块之间的通讯正常	%r.m.c.16
AXIS_IS_LINKED	EBOOL	读	此轴属于某个轴组	%r.m.c.17
AXIS_IN_CMD	EBOOL	读	轴是活动的且可控制	%r.m.c.18
AXIS_AT_TARGET	EBOOL	读	轴位于目标位置的正确位置带中	%r.m.c.20
AXIS_POS_LIMIT	EBOOL	读	轴已到达正限位	%r.m.c.21

标准符号	类型	访问	含义	地址
AXIS_NEG_LIMIT	EBOOL	读	轴已到达负限位	%lr.m.c.22
AXIS_WARNING	EBOOL	读	驱动器返回 MotionWarning 状态	%lr.m.c.23
AXIS_HOLD	EBOOL	读	轴已停止，正在等待命令	%lr.m.c.28
AXIS_HALT	EBOOL	读	轴已停止	%lr.m.c.29
AXIS_FASTSTOP	EBOOL	读	轴已快速停止	%lr.m.c.30
AXIS_READY	EBOOL	读	轴已准备响应命令	%lr.m.c.31
CONF_OK	EBOOL	读	通道已配置	%lr.m.c.32

隐式交换输出对象列表

下表介绍适用于 TSX_CSX_84 模块的 T_CSX_RING IODDT 隐式交换输出对象。

标准符号	类型	访问	含义	地址
CONTROL_ACQUIRE	EBOOL	读写	控制采集	%Qr.m.c.2
CONTROL_ENABLE	EBOOL	读写	控制启用	%Qr.m.c.10
CONTROL_FOLLOW	EBOOL	读写	一根轴或一组跟随器轴的跟随控制	%Qr.m.c.11
CONTROL_RESUME	EBOOL	读写	停止后恢复控制	%Qr.m.c.12
CONTROL_CLEAR_FLT	EBOOL	读写	故障清除控制	%Qr.m.c.15
ALLOW_ACQUIRE	EBOOL	读写	采集启用控制	%Qr.m.c.18
ALLOW_ENABLE	EBOOL	读写	禁用轴控	%Qr.m.c.26
ALLOW_FOLLOW	EBOOL	读写	取消跟随一根轴或一组跟随器轴的控制	%Qr.m.c.27
ALLOW_RESUME	EBOOL	读写	授权运动在使用 HOLD 命令停止后继续	%Qr.m.c.28
ALLOW_MOVE	EBOOL	读写	授权运动在使用 HALT 命令停止后继续	%Qr.m.c.29
ALLOW_NOT_FASTSTOP	EBOOL	读写	快速停止后控制	%Qr.m.c.30
ALLOW_NOT_FLT	EBOOL	读写	启用故障控制	%Qr.m.c.31

参数报告字

下表介绍适用于 TSX_CSX_84 模块的 T_CSX_RING IODDT 隐式交换输入字。

标准符号	类型	访问	含义	地址
PARAM_RPT	INT	读	指示编程故障的参数报告。最低有效字节包含错误代码，最高有效字节包含触发错误的字段的寄存器中的地址。	%lWr.m.c.2

有关 T_CSYS_RING 类型的 IODDT 显式交换对象的详细信息

概览

本部分介绍适用于 TSX CSY 84 模块的通道 0 的 T_CSYS_RING IODDT 显式交换对象。它对字类型的对象分组，这些对象的位具有特殊含义。下面将详细介绍这些对象。

变量声明示例：IODDT_VAR1，类型为 T_CSYS_RING

注：

- 各个位的含义一般是针对该位的状态 1 给出的。在特定情况下，将解释每个位状态。
- 并未使用所有位。

显式交换执行的指示器：EXCH_STS

下表介绍显式交换的控制位：EXCH_STS (%MWr.m.c.0)。这些变量由系统在每次显式交换时自动更新。

标准符号	类型	访问	含义	地址
STS_IN_PROGR	BOOL	读	正在读取通道状态字	%MWr.m.c.0.0
CMD_IN_PROGR	BOOL	读	正在进行命令参数交换	%MWr.m.c.0.1
ADJ_IN_PROGR	BOOL	读	正在进行调整参数交换	%MWr.m.c.0.2
TRF_IN_PROGR	BOOL	读	TRF_RECIPES 功能正在进行中	%MWr.m.c.0.3
RECONF_IN_PROGR	BOOL	读	模块重新配置	%MWr.m.c.0.15

显式交换报告: EXCH_RPT

下表介绍报告位: EXCH_RPT (%MWr.m.c.1)。这些变量由系统在每次显式交换时自动更新。

标准符号	类型	访问	含义	地址
STS_ERR	BOOL	读	通道状态字读取故障 (1 = 故障)	%MWr.m.c.1.0
CMD_ERR	BOOL	读	在交换命令参数期间发生故障 (1 = 故障)	%MWr.m.c.1.1
ADJ_ERR	BOOL	读	在交换调整参数期间发生故障 (1 = 故障)	%MWr.m.c.1.2
TRF_ERR	BOOL	读	TRF_RECIPÉ 功能正在进行时 发生故障	%MWr.m.c.1.3
RECONF_ERR	BOOL	读	在重新配置通道期间发生故障 (1 = 故障)	%MWr.m.c.1.15

通道故障字

下表介绍通道故障位: CH_FLT。这些变量由 READ_STS (IODDT_VAR1) 更新。

标准符号	类型	访问	含义	地址
EXT_FLT0	BOOL	读	外部故障 0: 伺服驱动器故障	%MWr.m.c.2.0
EXT_FLT1	BOOL	读	外部故障 1: 轴的通讯故障	%MWr.m.c.2.1
EXT_FLT2	BOOL	读	外部故障 2	%MWr.m.c.2.3
INT_FLT	BOOL	读	内部故障	%MWr.m.c.2.4
CONF_FLT	BOOL	读	配置故障: 硬件配置和软件配置不同	%MWr.m.c.2.5
COM_FLT	BOOL	读	通讯故障	%MWr.m.c.2.6
APPLI_FLT	BOOL	读	项目配置故障、调整或命令故障	%MWr.m.c.2.7
FAN_STOPPED	BOOL	读	风扇故障 (仅通道 0)	%MWr.m.c.2.8
OVER_TEMP	BOOL	读	过热 (仅通道 0)	%MWr.m.c.2.9
SENSOR_FLT	BOOL	读	温度传感器故障 (仅通道 0)	%MWr.m.c.2.10
PROCESS_CONF	BOOL	读	正在创建运动对象	%MWr.m.c.2.11
PROCESS_CONF_FAILED	BOOL	读	配置故障 (通道 0 除外)	%MWr.m.c.2.12

WRITE_CMD 接口字

下表解释与 WRITE_CMD（其动作在字 ACTION_CMD 中指定）关联的变量的含义。这些变量由 WRITE_CMD (IODDT_VAR1) 更新。

标准符号	类型	访问	含义	地址
ERROR_CMD	INT	读写	WRITE_CMD 出错	%MWr.m.c.19
RETURN_CMD_1	DINT	读写	功能的返回值 1	%MDr.m.c.20
RETURN_CMD_2	REAL	读写	功能的返回值 2	%MFr.m.c.22
RETURN_CMD_3	REAL	读写	功能的返回值 3	%MFr.m.c.24
ACTION_CMD	INT	读写	要执行的动作	%MWr.m.c.26
PARAM_CMD_1	DINT	读写	参数 1	%MDr.m.c.27
PARAM_CMD_2	DINT	读写	参数 2	%MDr.m.c.29
PARAM_CMD_3	REAL	读写	参数 3	%MFr.m.c.31
PARAM_CMD_4	REAL	读写	参数 4	%MFr.m.c.33

READ_PARAM、WRITE_PARAM 接口字

下表解释对于通道 1 至 16 可以使用 READ_PARAM 和 WRITE_PARAM 功能访问的参数含义。这些变量使用 READ_PARAM (IODDT_VAR1) 或 WRITE_PARAM (IODDT_VAR1) 来更新。还可以使用 SAVE_PARAM 和 RESTORE_PARAM 功能。

标准符号	类型	访问	含义	地址
CYCLE_TIME	INT	读写	SERCOS 环路循环时间	%MWr.m.c.35
BAUD_RATE	INT	读写	SERCOS 总线上的波特率（以波特为单位）	%MWr.m.c.36
OPTICAL_POWER	INT	读写	光纤中的光功率	%MWr.m.c.37

有关 T_CSY_TRF 类型的 IODDT 显式交换对象的详细信息

概览

本部分介绍适用于 TSX CSY 84 模块的通道 1 至 32 的 T_CSY_TRF IODDT 显式交换对象。它对字类型的对象分组，这些字的位具有特定的含义。下面将详细介绍这些对象。

变量声明示例：IODDT_VAR1 ，类型为 T_CSY_TRF

注：

- 各个位的含义一般是针对该位的状态 1 给出的。在特定情况下，将解释每个位状态。
- 并未使用所有位。

TRF_RECIPE 功能的执行标志

下表介绍指示 TRF_RECIPE 是否正在执行的控制位。

标准符号	类型	访问	含义	地址
TRF_IN_PROGR	BOOL	读	正在执行 TRF_RECIPE 功能	%MW.r.m.c.0.3

显式交换报告：EXCH_RPT

下表介绍特定于 TRF_RECIPE 功能的报告位。

标准符号	类型	访问	含义	地址
TRF_ERR	BOOL	读	正在执行 TRF_RECIPE 功能时发生故障	%MW.r.m.c.1.3

TRF_RECIPE 功能的对象

下表介绍与 TRF_RECIPE 功能关联的对象。这些对象由系统在每次使用 TRF_RECIPE 功能时自动更新。

标准符号	类型	访问	含义	地址
ERROR_TRF	INT	读	TRF_RECIPE 功能读取错误	%MWr.m.c.3
RETURN_TRF_1	DINT	读	TRF_RECIPE 功能的返回值 1	%MDr.m.c.4
RETURN_TRF_2	REAL	读	TRF_RECIPE 功能的返回值 2	%MFr.m.c.6
RETURN_TRF_3	REAL	读	TRF_RECIPE 功能的返回值 3	%MFr.m.c.8
ACTION_TRF	INT	读	要由 TRF_RECIPE 功能执行的动作	%MWr.m.c.10
PARAM_TRF_1	DINT	读	TRF_RECIPE 功能的参数 1	%MDr.m.c.11
PARAM_TRF_2	DINT	读	TRF_RECIPE 功能的参数 2	%MDr.m.c.13
PARAM_TRF_3	REAL	读	TRF_RECIPE 功能的参数 3	%MFr.m.c.15
PARAM_TRF_4	REAL	读	TRF_RECIPE 功能的参数 4	%MFr.m.c.17

有关 T_CSY_IND 类型的 IODDT 隐式交换对象的详细信息

概览

T_CSY_IND IODDT 拥有隐式交换对象，如下所述。此类 IODDT 适用于 TSX_CSY_84 模块的通道 21 至 16。

变量声明示例：IODDT_VAR1 ， 类型为 T_CSY_IND

各个位的含义一般是针对该位的状态 1 给出的。在特定情况下，将解释两个位状态。

隐式交换输入对象列表

下表介绍适用于 TSX_CSY_84 模块的 T_CSY_IND IODDT 隐式交换输入对象。

标准符号	类型	访问	含义	地址
CH_ERROR	EBOOL	读	通道错误位	%lr.m.c.ERR
RAMPING	EBOOL	读	指示轴是在加速还是在减速	%lr.m.c.0
STEADY	EBOOL	读	速度稳定	%lr.m.c.1
STOPPING	EBOOL	读	运动正在减速至停止	%lr.m.c.2
PROFILE_END	EBOOL	读	最后的配置文件命令已发送到模块	%lr.m.c.3
IN_POSITION	EBOOL	读	轴位于正确位置带中	%lr.m.c.4
AXIS_HOMING	EBOOL	读	轴正在回归。对于虚轴，此位停用	%lr.m.c.5
AXIS_HOMED	EBOOL	读	在主位置以外引用了轴位置读数	%lr.m.c.6
AXIS_NOT_FOLLOWING	EBOOL	读	驱动器不识别模块命令	%lr.m.c.7
HOLDING	EBOOL	读	轴保持在等待位置	%lr.m.c.8
RESUMING	EBOOL	读	轴在等候状态后一直在运动	%lr.m.c.9
DRIVE_ENABLED	EBOOL	读	驱动器已启用	%lr.m.c.10
DRIVE_DIAG	EBOOL	读	驱动器正在执行 3 类诊断	%lr.m.c.11
DRIVE_WARNING	EBOOL	读	驱动器正在执行 2 类诊断	%lr.m.c.12
DRIVE_FLT	EBOOL	读	驱动器正在执行 1 类诊断	%lr.m.c.13
DRIVE_DISABLED	EBOOL	读	驱动器已禁用	%lr.m.c.14
AXIS_SUMMARY_FLT	EBOOL	读	驱动器故障	%lr.m.c.15
AXIS_COM_OK	EBOOL	读	驱动器与模块之间的通讯正常	%lr.m.c.16
AXIS_IS_LINKED	EBOOL	读	此轴属于某个轴组	%lr.m.c.17
AXIS_IN_CMD	EBOOL	读	轴是活动的且可控制	%lr.m.c.18
AXIS_AT_TARGET	EBOOL	读	轴位于目标位置的正确位置带中	%lr.m.c.20
AXIS_POS_LIMIT	EBOOL	读	轴已到达正限位	%lr.m.c.21

标准符号	类型	访问	含义	地址
AXIS_NEG_LIMIT	EBOOL	读	轴已到达负限位	%lr.m.c.22
AXIS_WARNING	EBOOL	读	驱动器返回 MotionWarning 状态	%lr.m.c.23
BIAS_REMAIN	EBOOL	读	偏移已添加到命令位置	%lr.m.c.24
AXIS_MANUAL_MODE	EBOOL	读	轴在手动模式下工作	%lr.m.c.25
DRIVE_REALTIME_BIT1	EBOOL	读	驱动器位	%lr.m.c.26
DRIVE_REALTIME_BIT2	EBOOL	读	驱动器位	%lr.m.c.27
AXIS_HOLD	EBOOL	读	轴已停止，正在等待命令	%lr.m.c.28
AXIS_HALT	EBOOL	读	轴已停止	%lr.m.c.29
AXIS_FASTSTOP	EBOOL	读	轴已快速停止	%lr.m.c.30
AXIS_READY	EBOOL	读	轴已准备响应命令	%lr.m.c.31
CONF_OK	EBOOL	读	通道已配置	%lr.m.c.32

隐式交换输出对象列表

下表介绍适用于 TSX_CSY_84 模块的 T_CSY_IND IODDT 隐式交换输出对象。

标准符号	类型	访问	含义	地址
CONTROL_ACQUIRE	EBOOL	读写	控制采集	%Qr.m.c.2
CONTROL_JOG_POS	EBOOL	读写	手动模式：沿轴正向的可见运动命令	%Qr.m.c.4
CONTROL_JOG_NEG	EBOOL	读写	手动模式：沿轴负向的可见运动命令	%Qr.m.c.5
REAL_TIME_CTRL_BIT1	EBOOL	读写	驱动器位	%Qr.m.c.6
REAL_TIME_CTRL_BIT2	EBOOL	读写	驱动器位	%Qr.m.c.7
OPERATION_MODE_1	EBOOL	读写	操作模式选择	%Qr.m.c.8
OPERATION_MODE_2	EBOOL	读写	操作模式选择	%Qr.m.c.9
CONTROL_ENABLE	EBOOL	读写	控制启用	%Qr.m.c.10
CONTROL_FOLLOW	EBOOL	读写	一根轴或一组跟随器轴的跟随控制	%Qr.m.c.11
CONTROL_RESUME	EBOOL	读写	停止后恢复控制	%Qr.m.c.12
CONTROL_INC_POS	EBOOL	读写	手动模式：沿轴正向的递增运动命令	%Qr.m.c.13
CONTROL_INC_NEG	EBOOL	读写	手动模式：沿轴负向的递增运动命令	%Qr.m.c.14
CONTROL_CLEAR_FLT	EBOOL	读写	故障清除控制	%Qr.m.c.15
ALLOW_ACQUIRE	EBOOL	读写	采集启用控制	%Qr.m.c.18
ALLOW_ENABLE	EBOOL	读写	禁用轴控	%Qr.m.c.26
ALLOW_FOLLOW	EBOOL	读写	取消跟随一根轴或一组跟随器轴的控制	%Qr.m.c.27
ALLOW_RESUME	EBOOL	读写	授权运动在使用 HOLD 命令停止后继续	%Qr.m.c.28
ALLOW_MOVE	EBOOL	读写	授权运动在使用 HALT 命令停止后继续	%Qr.m.c.29
ALLOW_NOT_FASTSTOP	EBOOL	读写	快速停止后控制	%Qr.m.c.30
ALLOW_NOT_FLT	EBOOL	读写	启用故障控制	%Qr.m.c.31

实际位置

下表介绍适用于 TSX_CSY_84 模块的 T_CSY_IND IODDT 的隐式交换输入实数。

标准符号	类型	访问	含义	地址
POSITION	REAL	读	实际位置	%IFr.m.c.0

参数报告字

下表介绍适用于 TSX_CSY_84 模块的 T_CSY_IND IODDT 的隐式交换输入字。

标准符号	类型	访问	含义	地址
PARAM_RPT	INT	读	指示编程故障的参数报告。最低有效字节包含错误代码，最高有效字节包含触发错误的字段的寄存器中的地址。	%IW.r.m.c.2

仿真位置

下表介绍适用于 TSX_CSY_84 模块的 T_CSY_IND IODDT 隐式交换双精度输出字。

标准符号	类型	访问	含义	地址
REMOTE_POSITION	DINT	读写	实轴和虚轴：手动模式中的位置递增。 远程轴：仿真位置	%QDr.m.c.0

有关 T_CS_Y_IND 类型的 IO DDT 显式交换对象的详细信息

概览

本部分介绍适用于 TSX CSY 84 模块的通道 1 至 16 的 T_CS_Y_IND 类型的 IO DDT 显式交换对象。它对字类型的对象分组，这些对象的位具有特殊含义。下面将详细介绍这些对象。

变量声明示例：IO DDT_VAR1 ，类型为 T_CS_Y_IND

注：

- 各个位的含义一般是针对该位的状态 1 给出的。在特定情况下，将解释每个位状态。
- 并未使用所有位。

显式交换执行的指示器：EXCH_STS

下表介绍显式交换的控制位：EXCH_STS (%MWr.m.c.0)。这些变量由系统在每次显式交换时自动更新。

标准符号	类型	访问	含义	地址
STS_IN_PROGR	BOOL	读	正在读取通道状态字	%MWr.m.c.0.0
CMD_IN_PROGR	BOOL	读	正在进行命令参数交换	%MWr.m.c.0.1
ADJ_IN_PROGR	BOOL	读	正在进行调整参数交换	%MWr.m.c.0.2
RECONF_IN_PROGR	BOOL	读	模块重新配置	%MWr.m.c.0.15

显式交换报告：EXCH_RPT

下表介绍报告位：EXCH_RPT (%MWr.m.c.1)。这些变量由系统在每次显式交换时自动更新。

标准符号	类型	访问	含义	地址
STS_ERR	BOOL	读	通道状态字读取故障 (1 = 故障)	%MWr.m.c.1.0
CMD_ERR	BOOL	读	在交换命令参数期间发生故障 (1 = 故障)	%MWr.m.c.1.1
ADJ_ERR	BOOL	读	在交换调整参数期间发生故障 (1 = 故障)	%MWr.m.c.1.2
RECONF_ERR	BOOL	读	在重新配置通道期间发生故障 (1 = 故障)	%MWr.m.c.1.15

通道故障字

下表介绍通道故障位：CH_FLT。这些变量由 READ_STS (IODDT_VAR1) 更新。

标准符号	类型	访问	含义	地址
EXT_FLT0	BOOL	读	外部故障 0: 伺服驱动器故障	%MWr.m.c.2.0
EXT_FLT1	BOOL	读	外部故障 1: 轴的通讯故障	%MWr.m.c.2.1
EXT_FLT2	BOOL	读	外部故障 2	%MWr.m.c.2.3
INT_FLT	BOOL	读	内部故障	%MWr.m.c.2.4
CONF_FLT	BOOL	读	配置故障: 硬件配置和软件配置不同	%MWr.m.c.2.5
COM_FLT	BOOL	读	通讯故障	%MWr.m.c.2.6
APPLI_FLT	BOOL	读	项目配置故障、调整或命令故障	%MWr.m.c.2.7
PROCESS_CONF	BOOL	读	正在创建运动对象	%MWr.m.c.2.11
PROCESS_CONF_FAILED	BOOL	读	配置故障 (通道 0 除外)	%MWr.m.c.2.12

TRF_RECIPE 功能的对象

下表介绍与 TRF_RECIPE 功能关联的对象。这些对象由系统在每次使用 TRF_RECIPE 功能时自动更新。

标准符号	类型	访问	含义	地址
ERROR_TRF	INT	读	TRF_RECIPE 功能读取错误	%MWr.m.c.3
RETURN_TRF_1	DINT	读	TRF_RECIPE 功能的返回值 1	%MDr.m.c.4
RETURN_TRF_2	REAL	读	TRF_RECIPE 功能的返回值 2	%MFr.m.c.6
RETURN_TRF_3	REAL	读	TRF_RECIPE 功能的返回值 3	%MFr.m.c.8
ACTION_TRF	INT	读	要由 TRF_RECIPE 功能执行的动作	%MWr.m.c.10
PARAM_TRF_1	DINT	读	TRF_RECIPE 功能的参数 1	%MDr.m.c.11
PARAM_TRF_2	DINT	读	TRF_RECIPE 功能的参数 2	%MDr.m.c.13
PARAM_TRF_3	REAL	读	TRF_RECIPE 功能的参数 3	%MFr.m.c.15
PARAM_TRF_4	REAL	读	TRF_RECIPE 功能的参数 4	%MFr.m.c.17

WRITE_CMD 接口字

下表解释与 WRITE_CMD（其动作在字 ACTION_CMD 中指定）关联的变量的含义。这些变量由 WRITE_CMD（IODDT_VAR1）更新。

标准符号	类型	访问	含义	地址
ERROR_CMD	INT	读写	WRITE_CMD 出错	%MWr.m.c.19
RETURN_CMD_1	DINT	读写	功能的返回值 1	%MDr.m.c.20
RETURN_CMD_2	REAL	读写	功能的返回值 2	%MFr.m.c.22
RETURN_CMD_3	REAL	读写	功能的返回值 3	%MFr.m.c.24
ACTION_CMD	INT	读写	要执行的动作	%MWr.m.c.26
PARAM_CMD_1	DINT	读写	参数 1	%MDr.m.c.27
PARAM_CMD_2	DINT	读写	参数 2	%MDr.m.c.29
PARAM_CMD_3	REAL	读写	参数 3	%MFr.m.c.31
PARAM_CMD_4	REAL	读写	参数 4	%MFr.m.c.33

READ_PARAM、WRITE_PARAM 接口字

下表解释对于通道 1 至 16 可以使用 READ_PARAM 和 WRITE_PARAM 功能访问的参数的含义。这些变量使用 READ_PARAM（IODDT_VAR1）或 WRITE_PARAM（IODDT_VAR1）来更新。还可以使用 SAVE_PARAM 和 RESTORE_PARAM 功能。

标准符号	类型	访问	含义	地址
FUNCTION_VALIDATION	INT	读写	包含选择性验证位的字	%MWr.m.c.35
ACCEL	REAL	读写	加速度值	%MFr.m.c.36
DECEL	REAL	读写	减速度值	%MFr.m.c.38
ACCEL_TYPE	INT	读写	加速度类型	%MWr.m.c.40
IN_POSITION_BAND	REAL	读写	正确位置带的值	%MFr.m.c.41
ENABLE_POSITION_BAND	REAL	读写	监控窗口的值	%MFr.m.c.43
ROLLOVER_MAX	REAL	读写	最大回转	%MFr.m.c.45
ROLLOVER_MIN	REAL	读写	最小回转	%MFr.m.c.47
ACCEL_MAX	REAL	读写	最大加速度	%MFr.m.c.49
DECEL_MAX	REAL	读写	最小减速度	%MFr.m.c.51
SPEED_MAX	REAL	读写	最高速度	%MFr.m.c.53
POSITION_MAX	REAL	读写	最大位置	%MFr.m.c.55
POSITION_MIN	REAL	读写	最小位置	%MFr.m.c.57
SCALE_NUMERATOR	REAL	读写	比例系数分子（独立轴和 GetGearRatio 功能的配置）	%MFr.m.c.59

标准符号	类型	访问	含义	地址
SCALE_DENOMINATOR	REAL	读写	比例系数分母（独立轴和 GetGearRatio 功能的配置）	%MFr.m.c.61
ACCEL_UNIT	INT	读写	加速度单位	%MWr.m.c.63
SPEED_UNIT	INT	读写	速度单位	%MWr.m.c.64
POSITION_UNIT	INT	读写	位置单位	%MWr.m.c.65

有关 T_CSY_FOLLOW 类型的 IODDT 隐式交换对象的详细信息

概览

T_CSY_FOLLOW IODDT 拥有隐式交换对象，如下所述。此类 IODDT 适用于 TSX_CSY_84 模块的通道 21 至 24。

变量声明示例：IODDT_VAR1 ， 类型为 T_CSY_FOLLOW

各个位的含义一般是针对该位的状态 1 给出的。在特定情况下，将解释两个位状态。

隐式交换输入对象列表

下表介绍适用于 TSX_CSY_84 模块的 T_CSY_FOLLOW IODDT 隐式交换输入对象。

标准符号	类型	访问	含义	地址
CH_ERROR	EBOOL	读	通道错误位	%lr.m.c.ERR
RAMPING	EBOOL	读	指示轴是在加速还是在减速	%lr.m.c.0
STEADY	EBOOL	读	速度稳定	%lr.m.c.1
STOPPING	EBOOL	读	运动正在减速至停止	%lr.m.c.2
PROFILE_END	EBOOL	读	最后的配置文件命令已发送到模块	%lr.m.c.3
IN_POSITION	EBOOL	读	轴位于正确位置带中	%lr.m.c.4
AXIS_HOMING	EBOOL	读	轴正在回归。对于虚轴，此位停用	%lr.m.c.5
AXIS_HOMED	EBOOL	读	在主位置以外引用了轴位置读数	%lr.m.c.6
AXIS_NOT_FOLLOWING	EBOOL	读	驱动器不识别模块命令	%lr.m.c.7
HOLDING	EBOOL	读	轴保持在等待位置	%lr.m.c.8
RESUMING	EBOOL	读	轴在等候状态后一直在运动	%lr.m.c.9
DRIVE_ENABLED	EBOOL	读	驱动器已启用	%lr.m.c.10
DRIVE_DIAG	EBOOL	读	驱动器正在执行 3 类诊断	%lr.m.c.11
DRIVE_WARNING	EBOOL	读	驱动器正在执行 2 类诊断	%lr.m.c.12
DRIVE_FLT	EBOOL	读	驱动器正在执行 1 类诊断	%lr.m.c.13
DRIVE_DISABLED	EBOOL	读	驱动器已禁用	%lr.m.c.14
AXIS_SUMMARY_FLT	EBOOL	读	驱动器故障	%lr.m.c.15
AXIS_COM_OK	EBOOL	读	驱动器与模块之间的通讯正常	%lr.m.c.16
AXIS_IS_LINKED	EBOOL	读	此轴属于某个轴组	%lr.m.c.17
AXIS_IN_CMD	EBOOL	读	轴是活动的且可控制	%lr.m.c.18
AXIS_AT_TARGET	EBOOL	读	轴位于目标位置的正确位置带中	%lr.m.c.20

标准符号	类型	访问	含义	地址
AXIS_POS_LIMIT	EBOOL	读	轴已到达正限位	%lr.m.c.21
AXIS_NEG_LIMIT	EBOOL	读	轴已到达负限位	%lr.m.c.22
AXIS_WARNING	EBOOL	读	驱动器返回 MotionWarning 状态	%lr.m.c.23
AXIS_HOLD	EBOOL	读	轴已停止，正在等待命令	%lr.m.c.28
AXIS_HALT	EBOOL	读	轴已停止	%lr.m.c.29
AXIS_FASTSTOP	EBOOL	读	轴已快速停止	%lr.m.c.30
AXIS_READY	EBOOL	读	轴已准备响应命令	%lr.m.c.31
CONF_OK	EBOOL	读	通道已配置	%lr.m.c.32

隐式交换输出对象列表

下表介绍适用于 TSX_CSY_84 模块的 T_CSY_FOLLOW IODDT 隐式交换输出对象。

标准符号	类型	访问	含义	地址
CONTROL_ACQUIRE	EBOOL	读写	控制采集	%Qr.m.c.2
CONTROL_ENABLE	EBOOL	读写	控制启用	%Qr.m.c.10
CONTROL_FOLLOW	EBOOL	读写	一根轴或一组跟随器轴的跟随控制	%Qr.m.c.11
CONTROL_RESUME	EBOOL	读写	停止后恢复控制	%Qr.m.c.12
CONTROL_CLEAR_FLT	EBOOL	读写	故障清除控制	%Qr.m.c.15
ALLOW_ACQUIRE	EBOOL	读写	采集启用控制	%Qr.m.c.18
ALLOW_ENABLE	EBOOL	读写	禁用轴控	%Qr.m.c.26
ALLOW_FOLLOW	EBOOL	读写	取消一根轴或一组跟随器轴的跟随控制	%Qr.m.c.27
ALLOW_RESUME	EBOOL	读写	授权运动在使用 HOLD 命令停止后继续	%Qr.m.c.28
ALLOW_MOVE	EBOOL	读写	授权运动在使用 HALT 命令停止后继续	%Qr.m.c.29
ALLOW_NOT_FASTSTOP	EBOOL	读写	快速停止后控制	%Qr.m.c.30
ALLOW_NOT_FLT	EBOOL	读写	启用故障控制	%Qr.m.c.31

实际位置

下表介绍适用于 TSX_CSY_84 模块的 T_CSY_FOLLOW IODDT 隐式交换输入对象。

标准符号	类型	访问	含义	地址
POSITION	REAL	读	实际位置	%lFr.m.c.0

参数报告字

下表介绍适用于 TSX_CSY_84 模块的 T_CSY_FOLLOW IODDT 隐式交换条目字。

标准符号	类型	访问	含义	地址
PARAM_RPT	INT	读	指示编程故障的参数报告。最低有效字节包含错误代码，最高有效字节包含触发错误的字段的寄存器中的地址。	%IW.r.m.c.2

有关 T_CSY_FOLLOW 类型的 IODDT 显式交换对象的详细信息

概览

本部分介绍适用于 TSX CSY 84 模块的通道 21 至 24 的 T_CSY_FOLLOW IODDT 显式交换对象。它对字类型的对象分组，这些字的位具有特定的含义。下面将详细介绍这些对象。

变量声明示例：IODDT_VAR1，类型为 T_CSY_FOLLOW

注：

- 各个位的含义一般是针对该位的状态 1 给出的。在特定情况下，将解释每个位状态。
- 并未使用所有位。

显式交换执行的指示器：EXCH_STS

下表介绍显式交换的控制位：EXCH_STS (%MWr.m.c.0)。这些变量由系统在每次显式交换时自动更新。

标准符号	类型	访问	含义	地址
STS_IN_PROGR	BOOL	读	正在读取通道状态字	%MWr.m.c.0.0
CMD_IN_PROGR	BOOL	读	正在进行命令参数交换	%MWr.m.c.0.1
ADJ_IN_PROGR	BOOL	读	正在进行调整参数交换	%MWr.m.c.0.2
RECONF_IN_PROGR	BOOL	读	模块重新配置	%MWr.m.c.0.15

显式交换报告：EXCH_RPT

下表介绍报告位：EXCH_RPT (%MWr.m.c.1)。这些变量由系统在每次显式交换时自动更新。

标准符号	类型	访问	含义	地址
STS_ERR	BOOL	读	通道状态字读取故障 (1 = 故障)	%MWr.m.c.1.0
CMD_ERR	BOOL	读	在交换命令参数期间发生故障 (1 = 故障)	%MWr.m.c.1.1
ADJ_ERR	BOOL	读	在交换调整参数期间发生故障 (1 = 故障)	%MWr.m.c.1.2
RECONF_ERR	BOOL	读	在重新配置通道期间发生故障 (1 = 故障)	%MWr.m.c.1.15

通道故障字

下表介绍通道故障位：CH_FLT。这些变量由 READ_STS (IODDT_VAR1) 更新。

标准符号	类型	访问	含义	地址
EXT_FLT0	BOOL	读	外部故障 0: 伺服驱动器故障	%MWr.m.c.2.0
EXT_FLT1	BOOL	读	外部故障 1: 轴的通讯故障	%MWr.m.c.2.1
EXT_FLT2	BOOL	读	外部故障 2	%MWr.m.c.2.3
INT_FLT	BOOL	读	内部故障	%MWr.m.c.2.4
CONF_FLT	BOOL	读	配置故障: 硬件配置和软件配置不同	%MWr.m.c.2.5
COM_FLT	BOOL	读	通讯故障	%MWr.m.c.2.6
APPLI_FLT	BOOL	读	项目配置故障、调整或命令故障	%MWr.m.c.2.7
PROCESS_CONF	BOOL	读	正在创建运动对象	%MWr.m.c.2.11
PROCESS_CONF_FAILED	BOOL	读	配置故障 (通道 0 除外)	%MWr.m.c.2.12

WRITE_CMD 接口字

下表解释与 WRITE_CMD (其动作在字 ACTION_CMD 中指定) 关联的变量的含义。这些变量由 WRITE_CMD (IODDT_VAR1) 更新。

标准符号	类型	访问	含义	地址
ERROR_CMD	INT	读写	WRITE_CMD 出错	%MWr.m.c.19
RETURN_CMD_1	DINT	读写	功能的返回值 1	%MDr.m.c.20
RETURN_CMD_2	REAL	读写	功能的返回值 2	%MFr.m.c.22
RETURN_CMD_3	REAL	读写	功能的返回值 3	%MFr.m.c.24
ACTION_CMD	INT	读写	要执行的动作	%MWr.m.c.26
PARAM_CMD_1	DINT	读写	参数 1	%MDr.m.c.27
PARAM_CMD_2	DINT	读写	参数 2	%MDr.m.c.29
PARAM_CMD_3	REAL	读写	参数 3	%MFr.m.c.31
PARAM_CMD_4	REAL	读写	参数 4	%MFr.m.c.33

READ_PARAM、WRITE_PARAM 接口字

下表解释对于通道 21 至 24 可以使用 READ_PARAM 和 WRITE_PARAM 功能访问的参数含义。这些变量使用 READ_PARAM (IODDT_VAR1) 或 WRITE_PARAM (IODDT_VAR1) 来更新。还可以使用 SAVE_PARAM 和 RESTORE_PARAM 功能。

标准符号	类型	访问	含义	地址
MASTER_CHANNEL	INT	读写	主轴的编号 (1 至 16, N 无法访问)	%MWr.m.c.35
SLAVE_CHANNEL_1	INT	读写	从轴 1 的编号	%MWr.m.c.36
FOLL_DESCRIPTION_1	INT	读写	从轴 1 的描述。此字由下述有效位组成, 并具有变量名称以及三个未命名的影响启动条件的位: 位 8、9 和 10 设置为零 = 立即启动 位 8 设置为 1 并且位 9 和 10 设置为零 = 主轴位置到达负向触发器 位 9 设置为 1 并且位 8 和 10 设置为零 = 主轴位置到达正向触发器 位 8 和 9 设置为 1 并且位 10 设置为零 = 主轴阈值 > 位置 位 8 和 9 设置为零并且位 10 设置为 1 = 主轴阈值 < 位置	%MWr.m.c.37
FOLL_WHERE_1	BOOL	读	0 = 控制器	%MWr.m.c.37.0
FOLL_TYPE_1	BOOL	读	0 = 比率模式 1 = 凸轮模式	%MWr.m.c.37.1
FOLL_POSITION_1	BOOL	读	0 = 跟随实际位置 1 = 跟随命令的位置	%MWr.m.c.37.2
FOLL_FOL_ON_HALT_1	BOOL	读	1 = 如果主轴 / 从轴链路被移除, 则停止跟随器轴	%MWr.m.c.37.3
FOLL_HALT_MASTER_1	BOOL	读	1 = 在跟随出错时停止主轴	%MWr.m.c.37.6
FOLL_BIAS_REMAIN_1	BOOL	读	1 = 主轴位置上的动态偏移	%MWr.m.c.37.7
NUMERATOR_1	REAL	读写	从轴 1 的分子	%MFr.m.c.38
DENOMINATOR_1	REAL	读写	从轴 1 的分母	%MFr.m.c.40
TRIGGER_POSITION_1	REAL	读写	从轴 1 的触发器值	%MFr.m.c.42
SLAVE_CHANNEL_2	INT	读写	从轴 2 的编号	%MWr.m.c.44

标准符号	类型	访问	含义	地址
FOLL_DESCRIPTION_2	INT	读写	从轴 2 的描述。此字由下述有效位组成，并具有变量名称以及三个未命名的影响启动条件的位： 位 8、9 和 10 设置为零 = 立即启动 位 8 设置为 1 并且位 9 和 10 设置为零 = 主轴位置到达负向触发器 位 9 设置为 1 并且位 8 和 10 设置为零 = 主轴位置到达正向触发器 位 8 和 9 设置为 1 并且位 10 设置为零 = 主轴阈值 > 位置 位 8 和 9 设置为零并且位 10 设置为 1 = 主轴阈值 < 位置	%MWr.m.c.45
FOLL_WHERE_2	BOOL	读	0 = 控制器	%MWr.m.c.45.0
FOLL_TYPE_2	BOOL	读	0 = 比率模式 1 = 凸轮模式	%MWr.m.c.45.1
FOLL_POSITION_2	BOOL	读	0 = 跟随实际位置 1 = 跟随命令的位置	%MWr.m.c.45.2
FOLL_FOL_ON_HALT_2	BOOL	读	1 = 如果主轴 / 从轴链路被移除，则停止跟随器轴	%MWr.m.c.45.3
FOLL_HALT_MASTER_2	BOOL	读	1 = 在跟随出错时停止主轴	%MWr.m.c.45.6
FOLL_BIAS_REMAIN_2	BOOL	读	1 = 主轴位置上的动态偏移	%MWr.m.c.45.7
NUMERATOR_2	REAL	读写	从轴 2 的分子	%MFr.m.c.46
DENOMINATOR_2	REAL	读写	从轴 2 的分母	%MFr.m.c.48
TRIGGER_POSITION_2	REAL	读写	从轴 2 的触发器值	%MFr.m.c.50
SLAVE_CHANNEL_3	INT	读写	从轴 3 的编号	%MWr.m.c.52
FOLL_DESCRIPTION_3	INT	读写	从轴 3 的描述。此字由下述有效位组成，并具有变量名称以及三个未命名的影响启动条件的位： 位 8、9 和 10 设置为零 = 立即启动 位 8 设置为 1 并且位 9 和 10 设置为零 = 主轴位置到达负向触发器 位 9 设置为 1 并且位 8 和 10 设置为零 = 主轴位置到达正向触发器 位 8 和 9 设置为 1 并且位 10 设置为零 = 主轴阈值 > 位置 位 8 和 9 设置为零并且位 10 设置为 1 = 主轴阈值 < 位置	%MWr.m.c.53
FOLL_WHERE_3	BOOL	读	0 = 控制器	%MWr.m.c.53.0

标准符号	类型	访问	含义	地址
FOLL_TYPE_3	BOOL	读	0 = 比率模式 1 = 凸轮模式	%MWr.m.c.53.1
FOLL_POSITION_3	BOOL	读	0= 跟随实际位置 1 = 跟随命令的位置	%MWr.m.c.53.2
FOLL_FOL_ON_HALT_3	BOOL	读	1 = 如果主轴 / 从轴链路被移除, 则停止跟随器轴	%MWr.m.c.53.3
FOLL_HALT_MASTER_3	BOOL	读	1 = 在跟随出错时停止主轴	%MWr.m.c.53.6
FOLL_BIAS_REMAIN_3	BOOL	读	1 = 主轴位置上的动态偏移	%MWr.m.c.53.7
NUMERATOR_3	REAL	读写	从轴 3 的分子	%MFr.m.c.54
DENOMINATOR_3	REAL	读写	从轴 3 的分母	%MFr.m.c.56
TRIGGER_POSITION_3	REAL	读写	从轴 3 的触发器值	%MFr.m.c.58
SLAVE_CHANNEL_4	INT	读写	从轴 4 的编号	%MWr.m.c.60
FOLL_DESCRIPTION_4	INT	读写	从轴 4 的描述。此字由下述有效位组成, 并具有变量名称以及三个未命名的影响启动条件的位: 位 8、9 和 10 设置为零 = 立即启动 位 8 设置为 1 并且位 9 和 10 设置为零 = 主轴位置到达负向触发器 位 9 设置为 1 并且位 8 和 10 设置为零 = 主轴位置到达正向触发器 位 8 和 9 设置为 1 并且位 10 设置为零 = 主轴阈值 > 位置 位 8 和 9 设置为零并且位 10 设置为 1 = 主轴阈值 < 位置	%MWr.m.c.61
FOLL_WHERE_4	BOOL	读	0 = 控制器	%MWr.m.c.61.0
FOLL_TYPE_4	BOOL	读	0 = 比率模式 1 = 凸轮模式	%MWr.m.c.61.1
FOLL_POSITION_4	BOOL	读	0= 跟随实际位置 1 = 跟随命令的位置	%MWr.m.c.61.2
FOLL_FOL_ON_HALT_4	BOOL	读	1 = 如果主轴 / 从轴链路被移除, 则停止跟随器轴	%MWr.m.c.61.3
FOLL_HALT_MASTER_4	BOOL	读	1 = 在跟随出错时停止主轴	%MWr.m.c.61.6
FOLL_BIAS_REMAIN_4	BOOL	读	1 = 主轴位置上的动态偏移	%MWr.m.c.61.7
NUMERATOR_4	REAL	读写	从轴 4 的分子	%MFr.m.c.62
DENOMINATOR_4	REAL	读写	从轴 4 的分母	%MFr.m.c.64
TRIGGER_POSITION_4	REAL	读写	从轴 4 的触发器值	%MFr.m.c.66
SLAVE_CHANNEL_5	INT	读写	从轴 5 的编号	%MWr.m.c.68

标准符号	类型	访问	含义	地址
FOLL_DESCRIPTION_5	INT	读写	从轴 5 的描述。此字由下述有效位组成，并具有变量名称以及三个未命名的影响启动条件的位： 位 8、9 和 10 设置为零 = 立即启动 位 8 设置为 1 并且位 9 和 10 设置为零 = 主轴位置到达负向触发器 位 9 设置为 1 并且位 8 和 10 设置为零 = 主轴位置到达正向触发器 位 8 和 9 设置为 1 并且位 10 设置为零 = 主轴阈值 > 位置 位 8 和 9 设置为零并且位 10 设置为 1 = 主轴阈值 < 位置	%MWr.m.c.69
FOLL_WHERE_5	BOOL	读	0 = 控制器	%MWr.m.c.69.0
FOLL_TYPE_5	BOOL	读	0 = 比率模式 1 = 凸轮模式	%MWr.m.c.69.1
FOLL_POSITION_5	BOOL	读	0 = 跟随实际位置 1 = 跟随命令的位置	%MWr.m.c.69.2
FOLL_FOL_ON_HALT_5	BOOL	读	1 = 如果主轴 / 从轴链路被移除，则停止跟随器轴	%MWr.m.c.69.3
FOLL_HALT_MASTER_5	BOOL	读	1 = 在跟随出错时停止主轴	%MWr.m.c.69.6
FOLL_BIAS_REMAIN_5	BOOL	读	1 = 主轴位置上的动态偏移	%MWr.m.c.69.7
NUMERATOR_6	REAL	读写	从轴 6 的分子	%MFr.m.c.70
DENOMINATOR_6	REAL	读写	从轴 6 的分母	%MFr.m.c.72
TRIGGER_POSITION_6	REAL	读写	从轴 6 的触发器值	%MFr.m.c.74
SLAVE_CHANNEL_6	INT	读写	从轴 6 的编号	%MWr.m.c.76
FOLL_DESCRIPTION_6	INT	读写	从轴 6 的描述。此字由下述有效位组成，并具有变量名称以及三个未命名的影响启动条件的位： 位 8、9 和 10 设置为零 = 立即启动 位 8 设置为 1 并且位 9 和 10 设置为零 = 主轴位置到达负向触发器 位 9 设置为 1 并且位 8 和 10 设置为零 = 主轴位置到达正向触发器 位 8 和 9 设置为 1 并且位 10 设置为零 = 主轴阈值 > 位置 位 8 和 9 设置为零并且位 10 设置为 1 = 主轴阈值 < 位置	%MWr.m.c.77
FOLL_WHERE_6	BOOL	读	0 = 控制器	%MWr.m.c.77.0

标准符号	类型	访问	含义	地址
FOLL_TYPE_6	BOOL	读	0 = 比率模式 1 = 凸轮模式	%MWr.m.c.77.1
FOLL_POSITION_6	BOOL	读	0 = 跟随实际位置 1 = 跟随命令的位置	%MWr.m.c.77.2
FOLL_FOL_ON_HALT_6	BOOL	读	1 = 如果主轴 / 从轴链路被移除, 则停止跟随器轴	%MWr.m.c.77.3
FOLL_HALT_MASTER_6	BOOL	读	1 = 在跟随出错时停止主轴	%MWr.m.c.77.6
FOLL_BIAS_REMAIN_6	BOOL	读	1 = 主轴位置上的动态偏移	%MWr.m.c.77.7
NUMERATOR_6	REAL	读写	从轴 6 的分子	%MFr.m.c.78
DENOMINATOR_6	REAL	读写	从轴 6 的分母	%MFr.m.c.80
TRIGGER_POSITION_6	REAL	读写	从轴 6 的触发器值	%MFr.m.c.82

有关 T_CSY_COORD 类型的 IODDT 隐式交换对象的详细信息

概览

T_CSY_COORD IODDT 拥有隐式交换对象，如下所述。此类 IODDT 适用于 TSX_CSY_84 模块的通道 17 至 20。

变量声明示例：IODDT_VAR1，类型为 T_CSY_COORD

各个位的含义一般是针对该位的状态 1 给出的。在特定情况下，将解释两个位状态。

隐式交换输入对象列表

下表介绍适用于 TSX_CSY_84 模块的 T_CSY_COORD IODDT 的隐式交换输入对象。

标准符号	类型	访问	含义	地址
CH_ERROR	EBOOL	读	通道错误位	%lr.m.c.ERR
RAMPING	EBOOL	读	指示轴是在加速还是在减速	%lr.m.c.0
STEADY	EBOOL	读	速度稳定	%lr.m.c.1
STOPPING	EBOOL	读	运动正在减速至停止	%lr.m.c.2
PROFILE_END	EBOOL	读	最后的配置文件命令已发送到模块	%lr.m.c.3
IN_POSITION	EBOOL	读	轴位于正确位置带中	%lr.m.c.4
AXIS_HOMING	EBOOL	读	轴正在回归。对于虚轴，此位停用	%lr.m.c.5
AXIS_HOMED	EBOOL	读	在主位置以外引用了轴位置读数	%lr.m.c.6
AXIS_NOT_FOLLOWING	EBOOL	读	驱动器不识别模块命令	%lr.m.c.7
HOLDING	EBOOL	读	轴保持在等待位置	%lr.m.c.8
RESUMING	EBOOL	读	轴在等候状态后一直在运动	%lr.m.c.9
DRIVE_ENABLED	EBOOL	读	驱动器已启用	%lr.m.c.10
DRIVE_DIAG	EBOOL	读	驱动器正在执行 3 类诊断	%lr.m.c.11
DRIVE_WARNING	EBOOL	读	驱动器正在执行 2 类诊断	%lr.m.c.12
DRIVE_FLT	EBOOL	读	驱动器正在执行 1 类诊断	%lr.m.c.13
DRIVE_DISABLED	EBOOL	读	驱动器已禁用	%lr.m.c.14
AXIS_SUMMARY_FLT	EBOOL	读	驱动器故障	%lr.m.c.15
AXIS_COM_OK	EBOOL	读	驱动器与模块之间的通讯正常	%lr.m.c.16
AXIS_IS_LINKED	EBOOL	读	此轴属于某个轴组	%lr.m.c.17
AXIS_IN_CMD	EBOOL	读	轴是活动的且可控制	%lr.m.c.18
AXIS_AT_TARGET	EBOOL	读	轴位于目标位置的正确位置带中	%lr.m.c.20

标准符号	类型	访问	含义	地址
AXIS_POS_LIMIT	EBOOL	读	轴已到达正限位	%lr.m.c.21
AXIS_NEG_LIMIT	EBOOL	读	轴已到达负限位	%lr.m.c.22
AXIS_WARNING	EBOOL	读	驱动器返回 MotionWarning 状态	%lr.m.c.23
AXIS_HOLD	EBOOL	读	轴已停止，正在等待命令	%lr.m.c.28
AXIS_HALT	EBOOL	读	轴已停止	%lr.m.c.29
AXIS_FASTSTOP	EBOOL	读	轴已快速停止	%lr.m.c.30
AXIS_READY	EBOOL	读	轴已准备响应命令	%lr.m.c.31
CONF_OK	EBOOL	读	通道已配置	%lr.m.c.32

隐式交换输出对象列表

下表介绍适用于 TSX_CSY_84 模块的 T_CSY_COORD IODDT 隐式交换输出对象。

标准符号	类型	访问	含义	地址
CONTROL_ACQUIRE	EBOOL	读写	控制采集	%Qr.m.c.2
CONTROL_ENABLE	EBOOL	读写	控制启用	%Qr.m.c.10
CONTROL_FOLLOW	EBOOL	读写	一根轴或一组跟随器轴的跟随控制	%Qr.m.c.11
CONTROL_RESUME	EBOOL	读写	停止后恢复控制	%Qr.m.c.12
CONTROL_CLEAR_FLT	EBOOL	读写	故障清除控制	%Qr.m.c.15
ALLOW_ACQUIRE	EBOOL	读写	采集启用控制	%Qr.m.c.18
ALLOW_ENABLE	EBOOL	读写	禁用轴控	%Qr.m.c.26
ALLOW_FOLLOW	EBOOL	读写	取消跟随一根轴或一组跟随器轴的控制	%Qr.m.c.27
ALLOW_RESUME	EBOOL	读写	授权运动在使用 HOLD 命令停止后继续	%Qr.m.c.28
ALLOW_MOVE	EBOOL	读写	授权运动在使用 HALT 命令停止后继续	%Qr.m.c.29
ALLOW_NOT_FASTSTOP	EBOOL	读写	快速停止后控制	%Qr.m.c.30
ALLOW_NOT_FLT	EBOOL	读写	启用故障控制	%Qr.m.c.31

参数报告字

下表介绍适用于 TSX_CSY_84 模块的 T_CSY_COORD IODDT 隐式交换输入字。

标准符号	类型	访问	含义	地址
PARAM_RPT	INT	读	指示编程故障的参数报告。最低有效字节包含错误代码，最高有效字节包含触发错误的字段的寄存器中的地址。	%lWr.m.c.2

有关 T_CSY_COORD 类型的 IODDT 显式交换对象的详细信息

概览

本部分介绍适用于 TSX CSY 84 模块的通道 17 至 20 的 T_CSY_COORD IODDT 显式交换对象。它对字类型的对象分组，这些字的位具有特定的含义。下面将详细介绍这些对象。

变量声明示例：IODDT_VAR1，类型为 T_CSY_COORD

注：

- 各个位的含义一般是针对该位的状态 1 给出的。在特定情况下，将解释每个位状态。
- 并未使用所有位。

显式交换执行的指示器：EXCH_STS

下表介绍显式交换的控制位：EXCH_STS (%MWr.m.c.0)。这些变量由系统在每次显式交换时自动更新。

标准符号	类型	访问	含义	地址
STS_IN_PROGR	BOOL	读	正在读取通道状态字	%MWr.m.c.0.0
CMD_IN_PROGR	BOOL	读	正在进行命令参数交换	%MWr.m.c.0.1
ADJ_IN_PROGR	BOOL	读	正在进行调整参数交换	%MWr.m.c.0.2
RECONF_IN_PROGR	BOOL	读	模块重新配置	%MWr.m.c.0.15

显式交换报告：EXCH_RPT

下表介绍报告位：EXCH_RPT (%MWr.m.c.1)。这些变量由系统在每次显式交换时自动更新。

标准符号	类型	访问	含义	地址
STS_ERR	BOOL	读	通道状态字读取故障 (1 = 故障)	%MWr.m.c.1.0
CMD_ERR	BOOL	读	在交换命令参数期间发生故障 (1 = 故障)	%MWr.m.c.1.1
ADJ_ERR	BOOL	读	在交换调整参数期间发生故障 (1 = 故障)	%MWr.m.c.1.2
RECONF_ERR	BOOL	读	在重新配置通道期间发生故障 (1 = 故障)	%MWr.m.c.1.15

通道故障字

下表介绍通道故障位：CH_FLT。这些变量由 READ_STS (IODDT_VAR1) 更新。

标准符号	类型	访问	含义	地址
EXT_FLT0	BOOL	读	外部故障 0：伺服驱动器故障	%MWr.m.c.2.0
EXT_FLT1	BOOL	读	外部故障 1：轴的通讯故障	%MWr.m.c.2.1
EXT_FLT2	BOOL	读	外部故障 2	%MWr.m.c.2.3
INT_FLT	BOOL	读	内部故障	%MWr.m.c.2.4
CONF_FLT	BOOL	读	配置故障：硬件配置和软件配置不同	%MWr.m.c.2.5
COM_FLT	BOOL	读	通讯故障	%MWr.m.c.2.6
APPLI_FLT	BOOL	读	项目配置故障、调整或命令故障	%MWr.m.c.2.7
PROCESS_CONF	BOOL	读	正在创建运动对象	%MWr.m.c.2.11
PROCESS_CONF_FAILED	BOOL	读	配置故障（通道 0 除外）	%MWr.m.c.2.12

WRITE_CMD 接口字

下表解释与 WRITE_CMD（其动作在字 ACTION_CMD 中指定）关联的变量的含义。这些变量由 WRITE_CMD (IODDT_VAR1) 更新。

标准符号	类型	访问	含义	地址
ERROR_CMD	INT	读写	WRITE_CMD 出错	%MWr.m.c.19
RETURN_CMD_1	DINT	读写	功能的返回值 1	%MDr.m.c.20
RETURN_CMD_2	REAL	读写	功能的返回值 2	%MFr.m.c.22
RETURN_CMD_3	REAL	读写	功能的返回值 3	%MFr.m.c.24
ACTION_CMD	INT	读写	要执行的动作	%MWr.m.c.26
PARAM_CMD_1	DINT	读写	参数 1	%MDr.m.c.27
PARAM_CMD_2	DINT	读写	参数 2	%MDr.m.c.29
PARAM_CMD_3	REAL	读写	参数 3	%MFr.m.c.31
PARAM_CMD_4	REAL	读写	参数 4	%MFr.m.c.33
PARAM_CMD_5	REAL	读写	参数 5	%MFr.m.c.35
PARAM_CMD_6	REAL	读写	参数 6	%MFr.m.c.37
PARAM_CMD_7	REAL	读写	参数 7	%MFr.m.c.39
PARAM_CMD_8	REAL	读写	参数 8	%MFr.m.c.41
PARAM_CMD_9	REAL	读写	参数 9	%MFr.m.c.43
PARAM_CMD_10	REAL	读写	参数 10	%MFr.m.c.45
PARAM_CMD_11	REAL	读写	参数 11	%MFr.m.c.47

标准符号	类型	访问	含义	地址
PARAM_CMD_12	REAL	读写	参数 12	%MFr.m.c.49
PARAM_CMD_13	REAL	读写	参数 13	%MFr.m.c.51
PARAM_CMD_14	REAL	读写	参数 14	%MFr.m.c.53
PARAM_CMD_15	REAL	读写	参数 15	%MFr.m.c.55
PARAM_CMD_16	REAL	读写	参数 16	%MFr.m.c.57
PARAM_CMD_17	REAL	读写	参数 17	%MFr.m.c.59
PARAM_CMD_18	REAL	读写	参数 18	%MFr.m.c.61

有关 T_CSY_CAM 类型的 IODDT 隐式交换对象的详细信息

概览

T_CSY_CAM IODDT 拥有隐式交换对象，如下所述。此类 IODDT 适用于 TSX_CSY_84 模块的通道 25 至 31。

变量声明示例：IODDT_VAR1 ， 类型为 T_CSY_CAM

各个位的含义一般是针对该位的状态 1 给出的。在特定情况下，将解释两个位状态。

隐式交换输入对象列表

下表介绍适用于 TSX_CSY_84 模块的 T_CSY_CAM IODDT 隐式交换输入对象。

标准符号	类型	访问	含义	地址
CH_ERROR	EBOOL	读	通道错误位	%lr.m.c.ERR

有关 T_CSY_CAM 类型的 IODDT 显式交换对象的详细信息

概览

本部分介绍适用于 TSX CSY 84 模块的通道 25 至 31 的 T_CSY_CAM IODDT 显式交换对象。它对字类型的对象分组，这些字的位具有特定的含义。下面将详细介绍这些对象。

变量声明示例：IODDT_VAR1 ，类型为 T_CSY_CAM

注：

- 各个位的含义一般是针对该位的状态 1 给出的。在特定情况下，将解释每个位状态。
- 并未使用所有位。

显式交换执行的指示器：EXCH_STS

下表介绍显式交换的控制位：EXCH_STS (%MWr.m.c.0)。这些变量由系统在每次显式交换时自动更新。

标准符号	类型	访问	含义	地址
STS_IN_PROGR	BOOL	读	正在读取通道状态字	%MWr.m.c.0.0
CMD_IN_PROGR	BOOL	读	正在进行命令参数交换	%MWr.m.c.0.1
ADJ_IN_PROGR	BOOL	读	正在进行调整参数交换	%MWr.m.c.0.2
TRF_IN_PROGR	BOOL	读	正在执行 TRF_RECIPÉ 功能	%MWr.m.c.0.3
RECONF_IN_PROGR	BOOL	读	模块重新配置	%MWr.m.c.0.15

显式交换报告：EXCH_RPT

下表介绍报告位：EXCH_RPT (%MWr.m.c.1)。这些变量由系统在每次显式交换时自动更新。

标准符号	类型	访问	含义	地址
STS_ERR	BOOL	读	通道状态字读取故障 (1 = 故障)	%MWr.m.c.1.0
CMD_ERR	BOOL	读	在交换命令参数期间发生故障 (1 = 故障)	%MWr.m.c.1.1
ADJ_ERR	BOOL	读	在交换调整参数期间发生故障 (1 = 故障)	%MWr.m.c.1.2
TRF_ERR	BOOL	读	正在执行 TRF_RECIPe 功能时发生故障	%MWr.m.c.1.3
RECONF_ERR	BOOL	读	在重新配置通道期间发生故障 (1 = 故障)	%MWr.m.c.1.15

通道故障字

下表介绍通道故障位：CH_FLT。这些变量由 READ_STS (IODDT_VAR1) 更新。

标准符号	类型	访问	含义	地址
EXT_FLT0	BOOL	读	外部故障 0：伺服驱动器故障	%MWr.m.c.2.0
EXT_FLT1	BOOL	读	外部故障 1：轴的通讯故障	%MWr.m.c.2.1
EXT_FLT2	BOOL	读	外部故障 2	%MWr.m.c.2.3
INT_FLT	BOOL	读	内部故障	%MWr.m.c.2.4
CONF_FLT	BOOL	读	配置故障：硬件配置和软件配置不同	%MWr.m.c.2.5
COM_FLT	BOOL	读	通讯故障	%MWr.m.c.2.6
APPLI_FLT	BOOL	读	项目配置故障、调整或命令故障	%MWr.m.c.2.7
PROCESS_CONF	BOOL	读	正在创建运动对象	%MWr.m.c.2.11
PROCESS_CONF_FAILED	BOOL	读	配置故障（通道 0 除外）	%MWr.m.c.2.12

TRF_RECIPE 功能的对象

下表介绍与 TRF_RECIPE 功能关联的对象。这些对象由系统在每次使用 TRF_RECIPE 功能时自动更新。

标准符号	类型	访问	含义	地址
ERROR_TRF	INT	读	TRF_RECIPE 功能读取错误	%MWr.m.c.3
RETURN_TRF_1	DINT	读	TRF_RECIPE 功能的返回值 1	%MDr.m.c.4
RETURN_TRF_2	REAL	读	TRF_RECIPE 功能的返回值 2	%MFr.m.c.6
RETURN_TRF_3	REAL	读	TRF_RECIPE 功能的返回值 3	%MFr.m.c.8
ACTION_TRF	INT	读	要由 TRF_RECIPE 功能执行的动作	%MWr.m.c.10
PARAM_TRF_1	DINT	读	TRF_RECIPE 功能的参数 1	%MDr.m.c.11
PARAM_TRF_2	DINT	读	TRF_RECIPE 功能的参数 2	%MDr.m.c.13
PARAM_TRF_3	REAL	读	TRF_RECIPE 功能的参数 3	%MFr.m.c.15
PARAM_TRF_4	REAL	读	TRF_RECIPE 功能的参数 4	%MFr.m.c.17

WRITE_CMD 接口字

下表解释与 WRITE_CMD (其动作在字 ACTION_CMD 中指定) 关联的变量的含义。这些变量由 WRITE_CMD (IODDT_VAR1) 更新。

标准符号	类型	访问	含义	地址
ERROR_CMD	INT	读写	WRITE_CMD 出错	%MWr.m.c.19
RETURN_CMD_1	DINT	读写	功能的返回值 1	%MDr.m.c.20
RETURN_CMD_2	REAL	读写	功能的返回值 2	%MFr.m.c.22
RETURN_CMD_3	REAL	读写	功能的返回值 3	%MFr.m.c.24
ACTION_CMD	INT	读写	要执行的动作	%MWr.m.c.26
PARAM_CMD_1	DINT	读写	参数 1	%MDr.m.c.27
PARAM_CMD_2	DINT	读写	参数 2	%MDr.m.c.29
PARAM_CMD_3	REAL	读写	参数 3	%MFr.m.c.31
PARAM_CMD_4	REAL	读写	参数 4	%MFr.m.c.33

16.3 适用于所有模块的 IODDT 类型的 T_GEN_MOD

类型为 T_GEN_MOD 的 IODDT 的语言对象的详细信息

概览

Premium PLC 的所有模块均有关联的 T_GEN_MOD 类型的 IODDT。

注意

- 通常情况下，位含义是针对位状态为 1 给出的。特定情况下，会针对位的每个状态给出解释。
- 不是所有位都会用到。

对象列表

下表显示了 IODDT 的对象：

标准符号	类型	访问	含义	地址
MOD_ERROR	BOOL	读	模块错误位	%I.r.m.MOD.ERR
EXCH_STS	INT	读	模块交换控制字	%MWr.m.MOD.0
STS_IN_PROGR	BOOL	读	正在读取模块的状态字	%MWr.m.MOD.0.0
EXCH_RPT	INT	读	交换报告字	%MWr.m.MOD.1
STS_ERR	BOOL	读	读取模块状态字时出错	%MWr.m.MOD.1.0
MOD_FLT	INT	读	模块的内部错误字	%MWr.m.MOD.2
MOD_FAIL	BOOL	读	内部错误，模块故障	%MWr.m.MOD.2.0
CH_FLT	BOOL	读	故障通道	%MWr.m.MOD.2.1
BLK	BOOL	读	端子块故障	%MWr.m.MOD.2.2
CONF_FLT	BOOL	读	硬件或软件配置故障	%MWr.m.MOD.2.5
NO_MOD	BOOL	读	模块缺失或不工作	%MWr.m.MOD.2.6
EXT_MOD_FLT	BOOL	读	模块的内部错误字（仅限 Fipio 扩展）	%MWr.m.MOD.2.7
MOD_FAIL_EXT	BOOL	读	内部故障，模块无法使用（仅限 Fipio 扩展）	%MWr.m.MOD.2.8
CH_FLT_EXT	BOOL	读	故障通道（仅限 Fipio 扩展）	%MWr.m.MOD.2.9
BLK_EXT	BOOL	读	端子块故障（仅限 Fipio 扩展）	%MWr.m.MOD.2.10
CONF_FLT_EXT	BOOL	读	硬件或软件配置故障（仅限 Fipio 扩展）	%MWr.m.MOD.2.13
NO_MOD_EXT	BOOL	读	模块缺失或不工作（仅限 Fipio 扩展）	%MWr.m.MOD.2.14

17.1 可访问功能列表

本节的目标

本节回顾 TSX CSY 84/164 模块的不同可访问功能。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
SERCOS(r) 通道 0 的可访问功能	436
实轴的可访问功能	438
虚轴的可访问功能	442
远程轴的可访问功能	445
协同轴组的可访问功能	447
从轴组的可访问功能	449
凸轮配置文件的可访问功能	451

SERCOS(r) 通道 0 的可访问功能

调整功能

下表列出调整功能：

功能	代码	反馈	参数
GetOpticalPower	1547	反馈 2: 百分比	无
SetOpticalPower	2547	无	参数 3: 百分比

诊断功能

下表列出诊断功能：

功能	代码	反馈	参数
GetActualPhase	550	反馈 1: 阶段	无
GetAxisID	523	反馈 1: 标识符	无
GetCommandedPhase	1545	反馈 1: 阶段	无
GetCombinedControl	1534	反馈 1: 运动控制位	无
GetControl	1525	反馈 1: 运动控制位	无
GetLoopDiagnosticMode	1546	反馈 1: 诊断模式	无
GetMotionFault	5510	反馈 1: 故障列表	无
GetMotionWarning	5511	反馈 1: 警告列表	无
GetNumberOfDrivesInRing	548	返回值 1: 伺服驱动器数量	无
GetNumberInSet	541	反馈 1: 轴数	无
GetSercosAddress	549	反馈 1: 地址	参数 1: 轴
GetStatus	5509	反馈 1: 状态位	无
IsLoopUp	543	反馈 1: 0 / 1	无
SetCommandedPhase	2545	无	参数 1: 阶段
SetLoopDiagnosticMode	2546	无	参数 1: 诊断模式

用于 IDN 参数的读 / 写功能

下表列出用于 SERCOS(r) 伺服驱动器的 IDN 参数的读 / 写功能:

功能	代码	反馈	参数
GetIDN_S	1556	反馈 1: SERCOS(r) 参数	参数 1: 标识符
GetIDN_P	1557	反馈 1: SERCOS(r) 参数	参数 1: 标识符
GetIDN_US	1558	反馈 1: SERCOS(r) 参数	参数 1: 标识符
GetIDN_UP	1559	反馈 1: SERCOS(r) 参数	参数 1: 标识符
SetIDN_S	2556	无	参数 1: 标识符 参数 2: SERCOS(r) 参数
SetIDN_P	2557	无	参数 1: 标识符 参数 2: SERCOS(r) 参数
SetIDN_US	2558	无	参数 1: 标识符 参数 3: SERCOS(r) 参数
SetIDN_UP	2559	无	参数 1: 标识符 参数 3: SERCOS(r) 参数

实轴的可访问功能

运动控制功能

下表列出运动控制功能：

功能	代码	反馈	参数
ForcedHome	6039	无	无
Home	6034	无	参数 3: 方向 参数 4: 速度
Unhome	6038	无	无

运动功能

下表列出运动功能：

功能	代码	反馈	参数
MoveImmed	513	无	参数 1: 类型 参数 3: 位置 参数 4: 速度
MoveImmed_G30	554	无	参数 1: 类型 参数 2: 点数 参数 3: 路径点数 参数 4: 接近速度
MoveQueue	520	无	参数 1: 类型 参数 3: 位置 参数 4: 速度
EnableRealTimeCtrlBit	2564	无	参数 1: 启用 (1) 禁用 (0)

调整功能

下表列出调整功能：

功能	代码	反馈	参数
DisableRollover	412	无	无
EnableRollover	411	无	无
GetAccel	1041	反馈 2: 加速度	无
GetAccelMax	1116	反馈 2: 加速度	无
GetActualSpeed	5065	反馈 1: 速度	无
GetDecel	1042	反馈 2: 减速度	无
GetDecelMax	1117	反馈 2: 减速度	无
GetDefaultSpeed	1065	反馈 1: 速度	无
GetChangeACCDECMove	1172	反馈 1: 1: FCt 已启用; PARAM_CMD_2 已在操作码 513 和 520 中确认。 0: FCt 已禁用。	无
GetEnableMode	1524	反馈 1: 模式	无
GetEnablePositionBand	1538	反馈 2: 位置	无
GetInPositionBand	1035	反馈 2: 位置	无
GetPositionLimit	1505	反馈 2: 位置	参数 1: 方向
GetRolloverLimit	1539	反馈 2: 位置	参数 1: 方向
GetSpeedLimit	1066	反馈 2: 速度	无
GetSpeedOverride	1513	反馈 2: 百分比	无
GetUnrolled CommandedPosition	547	反馈 2: 位置	无
GetUnrolledPosition	546	反馈 2: 位置	无
SetAccel	2041	无	参数 3: 加速度
SetAccelMax	2116	无	参数 3: 加速度
SetDecel	2042	无	参数 3: 减速度
SetDecelMax	2117	无	参数 3: 减速度
SetChangeACCDECMove	2172	无	参数 1: 1: FCt 已启用; PARAM_CMD_2 已在操作码 513 和 520 中确认。 0: FCt 已禁用。

功能	代码	反馈	参数
SetDefaultSpeed	2065	无	参数 1: 速度
SetEnableMode	2524	无	参数 1: 模式
SetEnablePositionBand	2538	无	参数 3: 位置
SetInPositionBand	2035	无	参数 3: 位置
SetPosition	2053	无	参数 3: 位置
SetPositionLimit	2505	无	参数 1: 方向 参数 3: 位置
SetRolloverLimit	2539	无	参数 1: 方向 参数 3: 位置
SetSpeedLimit	2066	无	参数 3: 速度
SetSpeedOverride	2513	无	参数 3: 百分比
EnableDriveWarning	5512	无	参数 1 = 13
DisableDriveWarning	5513	无	参数 1 = 13

诊断功能

下表列出诊断功能:

功能	代码	反馈	参数
GetAxisId	523	反馈 1: 标识符	无
GetCommandedPosition	1053	反馈 2: 位置	无
GetCombinedControl	1534	反馈 1: 运动控制位	无
GetControl	1525	反馈 1: 运动控制位	无
GetMotionFault	5510	反馈 1: 故障列表	无
GetMotionWarning	5511	反馈 1: 警告列表	无
GetMoveQueueLength	9510	返回值 1: 长度	无
GetNumberInSet	541	反馈 1: 轴数	无
GetSercosAddress	549	反馈 1: 地址	无

配置功能

下表列出配置功能：

功能	代码	反馈	参数
GetAccelType	1540	反馈 1: 加速度类型	无
GetGearRatio	1500	反馈 2: 分子 反馈 3: 分母	无
SetAccelType	2540	无	参数 1: 加速度类型
SetGearRatio	2500	无	参数 3: 分子 参数 4: 分母

用于 IDN 参数的读 / 写功能

下表列出用于 SERCOS® 伺服驱动器的 IDN 参数的读 / 写功能：

功能	代码	反馈	参数
GetIDN_S	1556	反馈 1: SERCOS® 参数	参数 1: 标识符
GetIDN_P	1557	反馈 1: SERCOS® 参数	参数 1: 标识符
GetIDN_US	1558	反馈 1: SERCOS® 参数	参数 1: 标识符
GetIDN_UP	1559	反馈 1: SERCOS® 参数	参数 1: 标识符
SetIDN_S	2556	无	参数 1: 标识符 参数 2: SERCOS® 参数
SetIDN_P	2557	无	参数 1: 标识符 参数 2: SERCOS® 参数
SetIDN_US	2558	无	参数 1: 标识符 参数 3: SERCOS® 参数
SetIDN_UP	2559	无	参数 1: 标识符 参数 3: SERCOS® 参数

虚轴的可访问功能

运动控制功能

下表列出运动控制功能：

功能	代码	反馈	参数
ForcedHome	6039	无	无
Home	6034	无	参数 3: 方向 参数 4: 速度
Unhome	6038	无	无

运动功能

下表列出运动功能：

功能	代码	反馈	参数
MoveImmed	513	无	参数 1: 类型 参数 3: 位置 参数 4: 速度
MoveImmed_G30	554	无	参数 1: 类型 参数 2: 点数 参数 3: 路径点数 参数 4: 接近速度
MoveQueue	520	无	参数 1: 类型 参数 3: 位置 参数 4: 速度

调整功能

下表列出调整功能：

功能	代码	反馈	参数
DisableRollover	412	无	无
EnableRollover	411	无	无
GetAccel	1041	反馈 2: 加速度	无
GetAccelMax	1116	反馈 2: 加速度	无
GetActualSpeed	5065	反馈 1: 速度	无
GetDecel	1042	反馈 2: 减速度	无
GetDecelMax	1117	反馈 2: 减速度	无
GetDefaultSpeed	1065	反馈 1: 速度	无

功能	代码	反馈	参数
GetChangeACCDECMove	1172	反馈 1: 1: FCt 已启用: PARAM_CMD_2 已在操作码 513 和 520 中确认。 0: FCt 已禁用。	无
GetEnablePositionBand	1538	反馈 2: 位置	无
GetPositionLimit	1505	反馈 2: 位置	参数 1: 方向
GetRolloverLimit	1539	反馈 2: 位置	参数 1: 方向
GetSpeedLimit	1066	反馈 2: 速度	无
GetSpeedOverride	1513	反馈 2: 百分比	无
GetUnrolled CommandedPosition	547	反馈 2: 位置	无
GetUnrolledPosition	546	反馈 2: 位置	无
SetAccel	2041	无	参数 3: 加速度
SetAccelMax	2116	无	参数 3: 加速度
SetDecel	2042	无	参数 3: 减速度
SetDecelMax	2117	无	参数 3: 减速度
SetDefaultSpeed	2065	无	参数 1: 速度
SetChangeACCDECMove	2172	无	参数 1: 1: FCt 已启用: PARAM_CMD_2 已在操作码 513 和 520 中确认。 0: FCt 已禁用。
SetEnablePositionBand	2538	无	参数 3: 位置
SetPosition	2053	无	参数 3: 位置
SetPositionLimit	2505	无	参数 1: 方向 参数 3: 位置
SetRolloverLimit	2539	无	参数 1: 方向 参数 3: 位置
SetSpeedLimit	2066	无	参数 3: 速度
SetSpeedOverride	2513	无	参数 3: 百分比

诊断功能

下表列出诊断功能：

功能	代码	反馈	参数
GetAxisId	523	反馈 1: 轴标识符	无
GetCommandedPosition	1053	反馈 2: 位置	无
GetCombinedControl	1534	反馈 1: 运动控制位	无
GetControl	1525	返回值 1: 运动控制位	无
GetMotionFault	5510	反馈 1: 故障列表	无
GetMotionWarning	5511	返回值 1: 警告列表	无
GetMoveQueueLength	9510	反馈 1: 长度	无
GetNumberInSet	541	反馈 1: 轴数	无

配置功能

下表列出配置功能：

功能	代码	反馈	参数
GetAccelType	1540	反馈 1: 加速度类型	无
GetGearRatio	1500	反馈 2: 分子 反馈 3: 分母	无
SetAccelType	2540	无	参数 1: 加速度类型
SetGearRatio	2500	无	参数 3: 分子 参数 4: 分母

远程轴的可访问功能

运动控制功能

下表列出运动控制功能：

功能	代码	反馈	参数
ForcedHome	6039	无	无
Home	6034	无	参数 3：方向 参数 4：速度
Unhome	6038	无	无

调整功能

下表列出调整功能：

功能	代码	反馈	参数
DisableRollover	412	无	无
EnableRollover	411	无	无
GetRolloverLimit	1539	反馈 2：位置	参数 1：方向
GetUnrolledPosition	546	反馈 2：位置	无
SetPosition	2053	无	参数 3：位置
SetRolloverLimit	2539	无	参数 1：方向 参数 3：位置

诊断功能

下表列出诊断功能：

功能	代码	反馈	参数
GetAxisId	523	反馈 1：轴标识符	无
GetCombinedControl	1534	返回值 1：运动控制位	无
GetControl	1525	返回值 1：运动控制位	无
GetMotionFault	5510	反馈 1：故障列表	无
GetMotionWarning	5511	返回值 1：警告列表	无
GetNumberInSet	541	反馈 1：轴数	无
GetSercosAddress	549	反馈 1：地址	参数 1：轴

配置功能

下表列出配置功能：

功能	代码	反馈	参数
GetGearRatio	1500	反馈 2: 分子 反馈 3: 分母	无
SetGearRatio	2500	无	参数 3: 分子 参数 4: 分母

协同轴组的可访问功能

运动控制功能

下表列出运动控制功能：

功能	代码	反馈	参数
Home	6034	无	参数 3: 方向 参数 4: 速度

运动功能

下表列出运动功能。在 MoveImmed 和 MoveQueue 功能中，使用 n 个寄存器（从参数 3 开始，其中 n 是协调组中的轴数）来指定协调组中所有轴的位置。然后，使用 n 个附加寄存器（从参数 n+3 开始）来指定所有轴的速度。

功能	代码	反馈	参数
MoveImmed	513	无	参数 1: 类型 参数 (2 + n): 位置 参数 (n + 3) -> (2n + 3): 速度
MoveQueue	520	无	参数 1: 类型 参数 (2 + n): 位置 参数 (n + 3) -> (2n + 3): 速度

调整功能

下表列出调整功能：

功能	代码	反馈	参数
GetActualSpeed	5065	反馈 1: 速度	无
GetDefaultSpeed	1065	反馈 1: 速度	无
GetSpeedOverride	1513	反馈 2: 百分比	无
SetDefaultSpeed	2065	无	参数 1: 速度
SetSpeedOverride	2513	无	参数 3: 百分比

诊断功能

下表列出诊断功能：

功能	代码	反馈	参数
GetAxisId	523	反馈 1: 轴标识符	无
GetCombinedControl	1534	返回值 1: 运动控制位	无
GetControl	1525	返回值 1: 运动控制位	无
GetMotionFault	5510	反馈 1: 故障列表	无
GetMotionWarning	5511	返回值 1: 警告列表	无
GetMoveQueueLength	9510	反馈 1: 长度	无

从轴组的可访问功能

调整功能

下表列出调整功能：

功能	代码	反馈	参数
DisableRollover	412	无	无
EnableRollover	411	无	无
GetAbsFollowerBias	1526	反馈 1: 位置	参数 1: 轴标识符
GetCamProfile	1530	反馈 1: 配置文件标识符	参数 1: 轴标识符
GetFollowerBias	1527	反馈 2: 位置	参数 1: 轴标识符
GetFollowerMode	1529	反馈 1: 跟随器模式	参数 1: 轴标识符
GetFollowerRatio	1114	反馈 2: 分子 反馈 3: 分母	参数 1: 轴标识符
GetMasterOffset	1532	反馈 2: 位置	参数 1: 轴标识符
GetSpeedOverride	1513	反馈 2: 百分比	无
GetMasterTriggerPosition	1531	反馈 2: 位置	参数 1: 轴标识符
SetFollowerRatio	2114	无	参数 1: 轴标识符 参数 3: 分子 参数 4: 分母
SetMasterOffset	2532	无	参数 1: 轴标识符 参数 3: 位置
SetMasterTriggerPosition	2531	无	参数 1: 轴标识符 参数 3: 位置
SetSpeedOverride	2513	无	参数 3: 百分比

诊断功能

下表列出诊断功能：

功能	代码	反馈	参数
GetAxisId	523	反馈 1: 轴标识符	无
GetCombinedControl	1534	返回值 1: 运动控制位	无
GetControl	1525	返回值 1: 运动控制位	无
GetMotionFault	5510	反馈 1: 故障列表	无
GetMotionWarning	5511	返回值 1: 警告列表	无
GetNumberInSet	541	反馈 1: 轴数	无

配置功能

下表列出配置功能：

功能	代码	反馈	参数
GetMaster	1528	反馈 1: 轴标识符	无
SetFollowerConfig	420	无	参数 1: 轴标识符 参数 2: 跟随器模式 参数 3: 分子或配置文件 ID 参数 4: 分母
SetMaster	2528	无	参数 1: 轴标识符

凸轮配置文件的可访问功能

调整功能

下表列出调整功能：

功能	代码	反馈	参数
长度	534	反馈 1：表长度	无
LookUpFollowerPosition	537	反馈 2：从轴位置	参数 3：主轴位置

配置功能

下表列出配置功能：

功能	代码	反馈	参数
GetCoord	532	反馈 1：主轴位置 反馈 2：从轴位置	参数 1：表索引
GetInterpType	530	反馈 1：内插类型	无
SetCoord	533	无	参数 1：表索引 参数 3：主轴位置 参数 4：从轴位置
SetInterpType	531	无	参数 1：内插类型

特定于 TSX CSY 164 的功能

IV

本部分主题

本部分描述特定于 TSX CSY 164 模块的新功能。

本部分包含了哪些内容？

本部分包括以下各章：

章	章节标题	页
18	兼容性	455
19	TSX CSY 164 模块的配置	457
20	特定于 TSX CSY 164 的功能	461
21	TSX CSY 164 的性能	469

兼容性

软件兼容性

- 要使用 TSX CSY 164 模块开发项目，Unity Pro 软件的版本至少必须为 SV = 2.1，
- 对于 TSX CSY 84 模块，在重新配置 Unity Pro 应用程序后，程序将与 TSX CSY 164 模块兼容。

TSX CSY 164 模块的配置

19

本章的目标

本章描述模块 TSX CSY 164 的配置屏幕和它能执行的一些不同功能: "SERCOS(r) 功能"、"实轴"、"虚轴"等。

本章包含了哪些内容?

本章包含了以下主题:

主题	页
TSX CSY 164 模块的配置	458
配置通道	459

TSX CSY 164 模块的配置

简介

在 "配置" 模式中, 可以定义每个 TSX CSY 164 模块路径的操作特性。

模块的添加

在 "运动" 系列的 "添加模块" 对话框中, 选择 TSX CSY 164 模块:



请参见 "TSX CSY 84 模块的配置" (参见第 117 页) 一章。

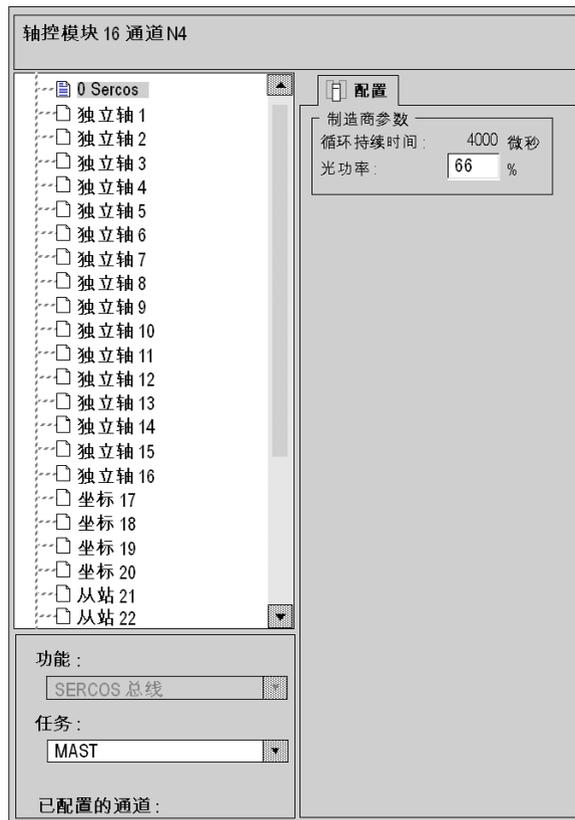
配置通道

简介

配置通道涉及到定义与该通道关联的应用专用功能的参数：

- 通道 0：SERCOS(r) 功能 (参见第 121 页)，
- 通道 1 至 16：独立轴（实轴、虚轴或远程轴），
- 通道 17 至 20：协调轴组 (参见第 128 页)，
- 通道 21 至 24：从轴组 (参见第 129 页)，
- 通道 25 至 31：凸轮配置文件 (参见第 132 页)。

示意图



独立轴的配置

独立轴（通道 1 至 16）能够支持实轴功能、虚轴功能和外部设定点：

- 通过配置实轴可以对物理轴（它使用伺服驱动器）进行控制。在此情况下，必须确保在 TSX CSY 164 模块配置屏幕中输入的参数与在伺服驱动器配置期间定义参数之间保持一致。
- 例如，虚轴可以作为从轴组中的主轴。此类轴的所有参数都在轴控模块的配置屏幕中定义。
- 远程轴可用于将远程位置数据项上载到模块。配置外部测量涉及到配置实轴和虚轴，其中只有位置信息有效。

协调轴的配置

TSX CSY 84 上的通道 17 至 20（协调轴组）允许使用以下服务：

- WRITE_CMD
- READ_STS

TSX CSY 164 上已添加了一个新命令：

- MOD_PARAM (参见第 462 页)

从轴的配置

TSX CSY 84 上的通道 21 至 24（从轴组）允许使用以下服务：

- READ_PARAM
- WRITE_PARAM
- WRITE_CMD (Move 除外)
- SAVE_PARAM
- RESTORE_PARAM
- READ_STS

TSX CSY 164 上已添加了一个新命令：

- MOD_PARAM (参见第 462 页)

特定于 TSX CSY 164 的功能

20

本章的目标

本章描述与 TSX CSY 164 模块关联的新功能。

本章包含了哪些内容？

本章包含了以下主题：

主题	页
组的动态重新配置	462
轴间位置偏差的监控功能	464

组的动态重新配置

描述

有一个新命令可用于从轴组和协调轴组，以便通过程序更改组中的成员轴。

此命令的作用与在线模式下的重新配置（通道紧急停止）相同。

它将允许：

- 动态更改从轴组的主轴
- 更改组中的成员轴

- **MOD_PARAM 指令的语法：**

MOD_PARAM %Chr.m.i: 写入 %Mx 个调整字至位于 r.m 地址（机架号，在机架中的位置）的模块组 i。

例如：写入从轴模式参数：写入位于插槽地址 4 的从轴组 24。

```
IF %M38 THEN
    MOD_PARAM (%CH4.24,0,0,0,0);
    RESET %M38;
END_IF;
```

- **检查交换：**

以下两个位可用于检查配置字中的写入：

%MWr.m.i.0.15: 当交换正在进行时，此位设置为 1。当交换已完成时，此位复位为 0。

%MWr.m.i.1.15: 当交换出错时（参数超出范围等），此位设置为 1

如果在运行 Unity Pro，则此功能编写如下：

- **Mod_Param (% Chr.m.c,0,0,0,0);**

括号中的参数始终等于 0。

此功能将使用参数字 (%MW) 作为接口。

关联的语言对象

此功能将使用参数 %MW 作为接口。调用此功能时，%MW 将被重新复制到常量 (%KW) 中，然后组将执行重新配置。

对于从轴组：

源存储器字	源存储器字助记符	目标常量字	注释
%MWx.y.36	SLAVE_CHANNEL_1	%KWx.y.3	从轴组的从轴 1 的编号
%MWx.y.37	FOLL_DESCRIPTION_1	%KWx.y.4	从轴组的从轴 1 的定义
%MWx.y.44	SLAVE_CHANNEL_2	%KWx.y.11	从轴组的从轴 2 的编号
%MWx.y.45	FOLL_DESCRIPTION_2	%KWx.y.12	从轴组的从轴 2 的定义
%MWx.y.52	SLAVE_CHANNEL_3	%KWx.y.19	从轴组的从轴 3 的编号
%MWx.y.53	FOLL_DESCRIPTION_3	%KWx.y.20	从轴组的从轴 3 的定义
%MWx.y.60	SLAVE_CHANNEL_4	%KWx.y.27	从轴组的从轴 4 的编号
%MWx.y.61	FOLL_DESCRIPTION_4	%KWx.y.28	从轴组的从轴 4 的定义
%MWx.y.68	SLAVE_CHANNEL_5	%KWx.y.35	从轴组的从轴 5 的编号
%MWx.y.69	FOLL_DESCRIPTION_5	%KWx.y.36	从轴组的从轴 5 的定义
%MWx.y.76	SLAVE_CHANNEL_6	%KWx.y.43	从轴组的从轴 6 的编号
%MWx.y.77	FOLL_DESCRIPTION_6	%KWx.y.44	从轴组的从轴 6 的定义

对于协调轴组：添加 Mod_Param 功能需要向接口语言添加下面的 %mW：

源存储器字	目标常量字	注释
%MWx.y.63	%KWx.y.4	协调轴组中的轴成员 0 的编号
%MWx.y.64	%KWx.y.5	协调轴组中的轴成员 1 的编号
%MWx.y.65	%KWx.y.6	协调轴组中的轴成员 2 的编号
%MWx.y.66	%KWx.y.7	协调轴组中的轴成员 3 的编号
%MWx.y.67	%KWx.y.8	协调轴组中的轴成员 4 的编号
%MWx.y.68	%KWx.y.9	协调轴组中的轴成员 5 的编号
%MWx.y.69	%KWx.y.10	协调轴组中的轴成员 6 的编号
%MWx.y.70	%KWx.y.11	协调轴组中的轴成员 7 的编号

调用此功能时，字 (%MW) 将被重新复制到常量 (%KW) 中，然后组将执行重新配置。

这些 %MW 的内容提供了协调轴组中所有成员轴的应用专用通道编号。

轴间位置偏差的监控功能

描述

通过轴间位置偏差的监控功能，可以监控模块的独立轴（实轴、虚轴或外部设定点）之间的任何偏差。

此功能可采用不同轴的列表进行多次实例化。

此功能将通过 3 个操作码确保：

- 启动监控
- 重新读取功能参数
- 停止监控

启动监控

通过在通道 0 上调用功能 **TRF_RECIPE**（操作码 26200）可启动监控。

调用该功能时，模块创建待监控的轴的不同位置的快照，然后：

- 如果两个轴的运动偏差超过**报警阈值**，模块将触发有关这两个故障轴的公告。
- 如果两个轴的运动偏差超过**故障阈值**，则模块将触发有关这两个故障轴的故障，停止所有成员轴，并触发有关非故障轴的公告。

对象	类型	说明	注释
%MF r.m.0.15	浮点数	实例编号	TRF_RECIPE 的参数，由模块转换为整数
%MF i	浮点数	报警阈值	以参考轴的位置单位表示
%MF i+2	浮点数	故障阈值	以参考轴的位置单位表示
%MW i+4	字	参考轴	
%MW i+5	字	轴 1	
%MF i+6	浮点数	轴 1 分子	
%MF i+8	浮点数	轴 1 分母	
..	
%MW i+5+5(N-1)	字	轴 N	
%MF i+5+5(N-1)+1	浮点数	轴 N 分子	
%MF i+5+5(N-1)+3	浮点数	轴 N 分母	

示例:

轴 8、12 和 7 的监控:

- 轴 8 以毫米为单位配置
- 轴 12 以 ° 为单位配置。它在轴 8 设置为 1 毫米时设置为 4°
- 轴 7 以转数为单位配置。它在轴 8 设置为 10 毫米时设置为 1 转

报警阈值为 4 毫米。

故障阈值为 10 毫米。

```
IF %MW104.0.0:X3 = 0 THEN (* 没有正在进行的 TRF_RECIPE *)
%MW104.0.10:= 26200;      (* 动作 26200 = GAP_CONTROL *)
%MF104.0.15:= 1.0;       (* 实例编号 *)
%MF150:=4.0;             (* 报警阈值 *)
%MF152:= 10.0;          (* 故障阈值 *)
%MW154:= 8;              (* 参考轴 *)
%MW155:= 12;            (* 要监控的第二个轴 *)
%MF156:= 4.0.0;         (* 第 2 个轴和参考轴之间的比率的分子 *)
%MF158:= 1.0;           (* 第 2 个轴和参考轴之间的比率的分子 *)
%MW160:=7.0;            (* 要监控的第三个轴 *)
%MF161:= 1.0;           (* 第 3 个轴和参考轴之间的比率的分子 *)
%MF163:= 10.0;          (* 第 3 个轴和参考轴之间的比率的分子 *)
TRF_RECIPE %CH 104.0 (15,150);
END_IF;
```

如果轴 8 在轴 12 仅运动 4° 时运动了 5 毫米，则会触发报警。

如果轴 8 在轴 12 仅运动 4° 时运动了 20 毫米，则会触发故障。

TRF_RECIPE 26200 被拒绝: 字 %MW.r.m.0.3 错误

- 数据的大小 (TRF_RECIPE 的第一个参数) 不正确 => 错误代码 23
- 实例编号无效 => 错误代码 10。实例编号必须介于 0 和 7 之间, 或最多只能同时有 8 个实例
- 实例已经是活动的 => 错误代码
- 报警阈值高于故障阈值 => 错误代码 65
- 阈值之一为负 => 错误代码 66
- 要监控的轴之一对应于尚未配置为实轴、虚轴或外部设定点轴的轴 => 错误代码 37
- 分母之一为零 => 错误代码 67

重新读取功能参数

通过在通道 0 上调用功能 **TRF_RECIPE**（操作码 16200）可为给定实例重新读取监控参数。

功能结果可在通用 %M 上读取。

对象	类型	说明	注释
%MF i	浮点数	报警阈值	以参考轴的位置单位表示
%MF i+2	浮点数	故障阈值	以参考轴的位置单位表示
%MW i+4	字	参考轴	
%MW i+5	字	轴 1	
%MF i+6	浮点数	轴 1 分子	
%MF i+8	浮点数	轴 1 分母	
..	
%MW i+5+5(N-1)	字	轴 N	
%MF i+5+5(N-1)+1	浮点数	轴 N 分子	
%MF i+5+5(N-1)+3	浮点数	轴 N 分母	

示例:

```
IF %MW104.0.0:X3 = 0 THEN (* 没有正在进行的 TRF_RECIPE *)
%MW104.0.10:= 16200;      (* 动作 26200 = GAP_CONTROL *)
%MF104.0.15:= 1.0        (* 要重新读取的实例 *)
TRF_RECIPE %CH 104.0 (15,150);
END_IF;
```

TRF_RECIPE 16200 被拒绝: 字 %MWr.m.0.3 错误

- 数据的大小（TRF_RECIPE 的第一个参数）不正确 => 错误代码 23
- 实例编号无效 => 错误代码 10。实例编号必须介于 0 和 7 之间，或最多只能同时有 8 个实例
- 实例不是活动的 => 错误代码 68

停止监控

通过在通道 0 上调用功能 **TRF_RECIPE**（操作码 14200）可删除给定实例的监控。

示例:

```
IF %MW104.0.0:X3 = 0 THEN (* 没有正在进行的 TRF_RECIPE *)
%MW104.0.10:= 14200;      (* 动作 14200 = GAP_CONTROL *)
%MF104.0.15:= 1.0        (* 要删除的实例 *)
TRF_RECIPE %CH 104.0 (.0);
END_IF;
```

TRF_RECIPE 14200 被拒绝: 字 %MWr.m.0.3 错误

- 实例编号无效 => 错误代码 10。实例编号必须介于 0 和 7 之间，或最多只能同时有 8 个实例
- 实例不是活动的 => 错误代码 68

检查监控活动

通过在通道 0 上调用功能 **TRF_RECIPE**（操作码 16201）可检查实例是否为活动的。

该功能结果可在通用 %MW 上读取。

```
TRF_RECIPE %CH4.0(1,150)
```

其中，1 为长度，150 为返回偏移值。

第一个字节中的每个位是对应实例的活动的映像（例如：位 0 => 实例 0，以此类推）

- 1=> 实例是活动的
- 0=> 实例已停用

示例：该实例是否为活动的？

```
IF %M9 AND NOT %MW4.0:X3 THEN
%MW4.0.10:=16201;
TRF_RECIPE %CH4.0(1,150);
RESET %M9;
END_IF;
```

结果可以在 %MW150 字上访问。

如果 %MW150 = 3，则实例 0 和 1 是活动的。

拒绝：字 %MWr.m,i,3 错误：

- 如果要读取的长度不是 1 ==> 错误代码 22
- 如果长度大于已配置的字 (%MW) 数 ==> 错误代码 19

TSX CSY 164 的性能

概览

该模块的性能为：

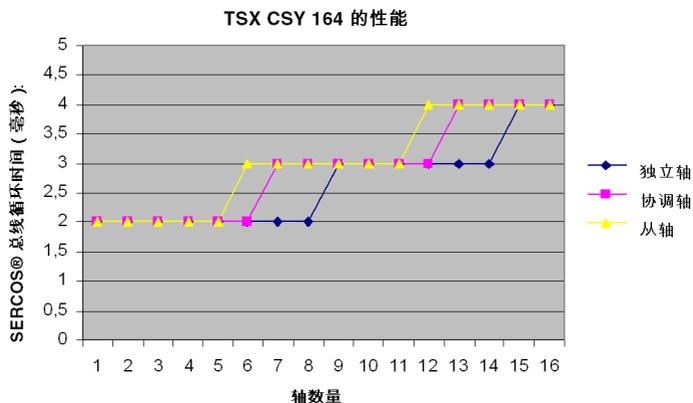
- SERCOS(r) 总线循环时间：
 - SERCOS(r) 单位循环时间 (tick) 缺省等于 4 毫秒
- 立即命令的处理 = 2 个 SERCOS(r) 单位循环时间 (tick)
- 排队命令的处理 = 2 个 SERCOS(r) 单位循环时间 (tick) * 排队命令的数量
- 远程测量的处理 = 2 个 SERCOS(r) 单位循环时间 (tick) + 1 个用于确认从轴上参考命令的 SERCOS(r) 单位循环时间 (tick)
- 轴之间的偏差的监控周期：20 毫秒（与单位循环时间 (tick) 无关）
- 循环时间的动态修改（通过 Write-cycle 命令）将从以下值中选择：
 - SERCOS(r) 单位循环时间 (tick) 为 2 毫秒
 - SERCOS(r) 单位循环时间 (tick) 为 3 毫秒
 - SERCOS(r) 单位循环时间 (tick) 为 4 毫秒

在强制进入 SERCOS(r) 环路的阶段 0 的过程中需确认。

SERCOS(r) 单位循环时间 (tick) 是根据所配置的轴数量和应用类型来定义的（请参见 "SERCOS(r) 循环时间选择指南"）。

注意：Lexium 每隔 250 微秒执行一次内插。

SERCOS(r)(r) 循环时间选择指南:

**测试条件:**

警告: 图中所示的性能是在一定条件下实现的, 这些条件保证大约 25% 的 SERCOS(r) CPU 时间缓冲。MAST 任务的周期配置为 20 毫秒。每隔 10 个 PLC 循环执行一次通过 MOVE_QUEUE 进行的运动命令传输。

结果,

- 对于 "独立轴" 配置: MOVE_QUEUE 每隔 200 毫秒发送到所有轴
- 对于 "从轴" 配置, MOVE_QUEUE 命令每隔 200 毫秒发送到主轴
- 对于 "协调轴" 配置, MOVE_QUEUE 命令每隔 200 毫秒发送到所有协调轴组

WRITE_PARAM 命令在此时间周期中发送, 以便执行参数调整。

使用程序修改循环时间

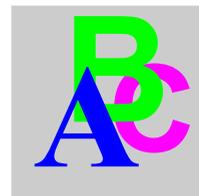
```
(* SERCOS (r) 循环时间更改 => %MW30 = 2000 或 3000 或 4000 *)
IF %M30 THEN      %MW101.0.26:=2565;
    %MD101.0.27:=%MW30;
    IF NOT %MW101.0:X1 THEN
        WRITE_CMD %CH101.0;
        RESET %M30;
        SET %M31;
    END_IF;
END_IF;

(* 转换到 SERCOS (r) 阶段 0 *)
IF %M31 THEN      %MW101.0.26:=2545;
    %MD101.0.27:=0;
    IF NOT %MW101.0:X1 THEN
        WRITE_CMD %CH101.0;
        RESET %M31;
        SET %M32;
    END_IF;
END_IF;

(* 到 SERCOS (r) 阶段 4 的新转换 *)
IF %M32 THEN      %MW101.0.26:=2545;
    %MD101.0.27:=4;
    IF NOT %MW101.0:X1 THEN
        WRITE_CMD %CH101.0;
        RESET %M32;
    END_IF;
END_IF;

(* 这里, 我们可以预期转换到阶段 4 (操作码 550 的结果为 %MD101.0.20),
然后确认转换到阶段 0 所导致的故障 *)
```

索引



- 990MCO000xxx, 90
- ABS_MOVE, 186
- ALLOW 位, 170
- CONT_MOVE, 186
- CONTROL 位, 170
- IDN, 301
- INCR_MOVE, 186
- MOD_PARAM, 462
- MoveImmed, 189, 223
- MoveQueue, 191
- READ_PARAM, 386
- READ_STS, 392
- RESTORE_PARAM, 387
- SAVE_PARAM, 387
- T_GEN_MOD, 433
- TRF_RECIPE
 - 一般信息, 238
 - 凸轮配置文件, 294
- TSXCSY164, 63
- TSXCSY84, 63
- TSXCSY85, 63
- WRITE_CMD, 388
- WRITE_PARAM, 386
- 功能, 101
- 功能代码, 436
- 参数设置, 374
- 参考点, 181
- 回归, 179
- 安装, 71
- 循环时间 (tick)
 - TSXCSY164, 469
 - TSXCSY84, 369
- 快速入门, 33
- 性能
 - TSXCSY164, 469
 - TSXCSY84, 369
- 所有模块的通道数据结构
 - T_GEN_MOD, 433
- 手动模式, 176
- 接线附件, 89
- 操作码, 436
- 故障管理, 347
- 用于 SERCOS 模块的通道数据结构
 - T_CSY_CAM, 394
 - T_CSY_CMD, 394
 - T_CSY_COORD, 394
 - T_CSY_FOLLOW, 394
 - T_CSY_IND, 394
 - T_CSY_RING, 394
 - T_CSY_TRF, 394
- 诊断, 76
 - IsLoopUp, 334
- 调整, 227
- 调试, 311
- 运动功能, 185
 - ABS_MOVE, 186
 - CONT_MOVE, 186
 - INCR_MOVE, 186
 - MoveImmed, 189, 223
 - MoveQueue, 191

运动控制功能
 CONTROL 位, *170*
 回归, *179*
运动状态位, *136*
配置, *115*
错误代码, *347*