

免责声明



技术规格

PLCopen - 技术委员会 2 - 工作组

运动控制功能块

(前称第 1 部分和第 2 部分)

版本 2.0, 已发布

该文档以“现状”提供，并可能在未来进行补充、修改或修正。PLCOPEN 特此声明不提供任何形式的明示或暗示的担保或条件，包括但不限于任何标题、非侵权、特定用途的适销性担保或条件。在任何情况下，PLCOPEN 对因本文件中的任何缺陷、错误或遗漏，或因任何人对本文件的使用或依赖而产生的任何损失或损害概不负责。版权所有 (C) 2009-2011 由 PLCopen 所有。保留所有权利。

2011 年 3 月 17 日。

TC2 任务组运动控制 2011 年 3 月 17 日 (C) 1999 - 2011 版权由 PLCopen 拥有，运动控制功能块版本 2.0，已出版页面 1/141

Function blocks for motion control

以下规范是在 PLCopen 运动控制工作组内制定的。该规范由以下运动控制工作组成员撰写：

Table of Contents

- 1. 一般. ...8
- 1.1. 目标. ...9

1.1.1. 语言上下文目标. ...9	
1.1.2. 一组功能块的定义. ...9	
1.1.3. 定义的功能块概述. .10	
1.1.4. 合规性和可移植性... ..10	
1.1.5. 名称的长度和缩短方式. .11	
1.1.6. 历史. ..11	
2. 模型..13	
2.1. 状态图. ..13	
2.2. 错误处理. ..16	
2.2.1. 集中式与分散式. .16	
2.2.2. 缓冲命令.. ..17	
2.2.3. '使能' 输入的定时示例. .17	
2.3. 定义. .19	
2.4. FB 接口. ..19	
2.4.1. 一般规则. ..19	
2.4.2. 中止与缓冲模式. .22	
2.4.3. AXIS_REF 数据类型. ..24	
2.4.4. 技术单位. ..24	
2.4.5. 为什么命令输入是边缘敏感的. ..25	
2.4.6. 输入'连续更新' ..25	
2.5. 示例 1: 相同功能块实例控制轴的不同运动..26	
2.6. 示例 2: 不同功能块实例控制轴的运动. ..27	
3. 单轴功能块. ...29	
3.1. MC_ 电源..29	
3.2. MC_ 家. ..30	
3.3. MC_ 停止.. ..31	
3.4. MC_ 停顿. ..33	
3.5. MC_ 绝对移动. ..35	
3.6. MC_ 相对移动. ..37	
3.7. MC_ 加法移动. ..39	
3.8. MC_ 重叠移动. ..41	
3.9. MC_ 重叠停顿. ..44	
3.10. MC_ 移动速度. ..45	
3.11. MC_MOVECONTINUOUSABSOLUTE . ..47	
3.12. MC_MoveContinuousRelative. .50	
3.13. MC_TORQUECONTROL ..53	
3.14. MC_PositionProfile. .56	
3.15. MC_VELOCITYPROFILE. .58	
3.16. MC_ACCELERATIONPROFILE . ..60	
3.17. MC_SETPOSITION63	
3.18. MC_SETOVERRIDE64	
3.19. MC_READPARAMETER & MC_READBOOLPARAMETER. 6.66	
3.20. MC_WriteParameter & MC_WriteBoolParameter ...68	
3.21. MC_READDIGITALINPUT . ..70	
3.22. MC_READDIGITALOUTPUT.. ..71	
3.23. MC_WriteDigitalOutput . ..72	

3.24. MC_READACTUALPOSITION. ..	73
3.25. MC_READACTUALVELOCITY ...	74
3.26. MC_READACTUALTORQUE . ..	75
3.27. MC_READSTATUS.. ...	76
3.28. ...	77

MC_READMOTIONSTATE

TC2 任务组运动控制 2011 年 3 月 17 日 (C) 1999 - 2011 版权所有, PLCopen
运动控制功能块第 2.0 版, 出版

Table of Figures

图 1: 带用户选项的三角形...	8
图 2: FB 状态图...	15
图 3: 具有集中错误处理的功能块...	16
图 4: 具有分散错误处理的功能块...	16
图 5: 带有'ENABLE' 输入的错误处理示例...	17
图 6: 带有'ENABLE' 输入的错误行为第二示例...	18
图 7: 相关 FBS 中'EXECUTE' / 'DONE' 的行为...	21
图 8: 相关 FBS 中'EXECUTE' / 'INXXX' 的行为...	22
图 9: 执行复杂运动的功能块...	25
图 10: 与 SFC 一起使用单个 FB ...	26
图 11: 单个 FB 使用的时序图...	27
图 12: 级联功能块的示例...	27
图 13: 级联功能块示例的时序图...	28
图 14: 具有 LD 的级联功能块示例...	28
图 15: MC_STOP 时序图...	31
图 16: MC_STOP 与 MC_MOVEVELOCITY 结合的行为...	32
图 17: MC_HALT 的示例...	34
图 18: MC_MOVEABSOLUTE 的时序图...	36
图 19: MC_MOVERELATIVE 的时序图...	38
图 20: MC_MOVEADDITIVE 的时序图...	40
图 21: MC_MOVESUPERIMPOSED 的时序图...	42
图 22: MC_MOVESUPERIMPOSED 对从属轴的影响示例...	43
图 23: MC_MOVESUPERIMPOSED 对 MC_MOVEABSOLUTE 的影响示例...	43
图 24: MC_MOVEVELOCITY 时序图...	46
图 25: MC_MOVECONTINUOUSABSOLUTE 示例...	49
图 26: 上述例子的 MC_MOVECONTINUOUSABSOLUTE 时间图...	49
图 27: MC_MOVECONTINUOUSRELATIVE 具有正方向的时间图...	51
2011 年 3 月 17 日	
图 28: MC_MOVECONTINUOUSRELATIVE 具有负方向的时间图	51
图 29: MC_MOVECONTINUOUSRELATIVE 的示例. ...	52
图 30: MC_TORQUECONTROL 的第一个示例...	54
图 31: MC_TORQUECONTROL 示例的程序...	55
图 32: MC_TORQUECONTROL 的第二个示例...	55

图 33: 时间 / MC_POSITIONPROFILE 的示例. ...57	
图 34: MC_ACCELERATIONPROFILE, 仅 10 段...61	
图 35: 结果 MC_POSITIONPROFILE.. ...62	
图 36: MC_SETOVERRIDE 的图形解释. ...65	
图 37: MC_DIGITALCAMSWITCH 的示例. ...83	
图 38: SWITCH01 的详细描述. ...83	
图 39: 负方向的示例. ...84	
图 40: MC_TOUCHPROBE 的时间示例. ...86	
图 41: 触发事件被接受的窗口示例 (针对模数轴)86	
图 42: 凸轮轮廓示例. ...88	
图 43: 齿轮时间图. ...95	
图 44: MC_GEARINPOS 的时间图. ...99	
图 45: 'SYNCMODES' 'SLOWDOWN' (绿色) 与'CATCHUP' (红色) 在不同的从动体初速度下的 差异示例. ...99	
图 46: 从动体初速度与主驱动体相同方向的 MC_GEARINPOS 示例. ...100	
图 47: 从动体初速度与主驱动体相反方向的 MC_GEARINPOS 示例. 反向主方向...100	
图 48: MC_PHASING 的时序示例 - 绝对和相对...102	
图 49: MC_PHASING 的示例 - 绝对和相对...103	
图 50: MC_COMBINEAXES 的应用示例...107	
图 51: MC_COMBINEAXES 示例的相应时序图...108	
图 52: 一个简单钻井单元的示例...109	
图 53: 药物左侧无混合、右侧带混合的时序图...109	
图 54: 带有功能块图的解决方案...110	
TC2 工作组运动控制 2011 年 3 月 17 日 (C) 1999 - 2011 版权由 PLCopen 的运动控制功能块版本 2.0, 发布页面 6/141	
图 55: 钻井示例中的直线步骤过渡链在 SFC 中...110	
图 56: 同轴上两个 MC_MOVEABSOLUTE 的基本示例...111	
图 57: 上述示例的时序图, 无 FBI 和 FB2 之间的干扰 ('中止' 模式) ...111	
图 58: 上述示例的时序图, FB2 中断 FB1 ('中止' 模式) ...112	
图 59: 上述示例的时序图, 在 '缓冲' 模式下...113	
图 60: 上述示例的时序图, 模式为 'BLENDINGLOW' ...114	
图 61: 上述示例的时序图, 模式为 'MERGING1' ...115	
图 62: 上述示例的时序图, 模式为 'BLENDINGNEXT' 运动...116	
图 63: 上述示例的时序图, 模式为 'BLENDINGHIGH' 运动...117	
图 64: PLCOPEN 运动控制标志...141	
表格概览	
表 1: 定义的功能块概述...10	
表 2: 通用规则...21 表 3: 枚举类型 MC_BUFFER_MODE. ...23	
表 4: 相关 FBS 上缓冲命令的概述. ...23	
表 5: MC_READPARAMETER 和 MC_WRITEPARAMETER 的参数...67	
表 6: 支持的数据类型..120	
表 7: 支持的派生数据类型. ...120	
表 8: 功能块的简要概述. ...121	
表 9: MC_READPARAMETERS 的参数, 包括 MC_READPARAMETER 和 MC_WRITE(BOOL)PARAMETER 和 MC_WRITE(BOOL)PARAMETER 180	

1. General

运动控制市场展示了各种不兼容的系统和解决方案。在使用不同系统的商业环境中，这种不兼容性给最终用户带来了可观的成本，学习变得混乱，工程变得困难，市场增长的过程也减缓了。

标准化无疑会减少这些负面因素。标准化不仅意味着编程语言本身(因为标准化是通过全球 IEC 61131-3 标准实现的)，还包括将接口标准化为不同的运动控制解决方案。通过这种方式，这些运动控制解决方案的编程对硬件的依赖性降低了。应用软件的可重用性增加了，培训和支持所涉及的成本也减少了。

用户已请求 PLCopen 帮助解决这个问题，这促成了运动控制工作组的成立。该工作组通过标准化运动控制的功能块定义了程序员的接口。

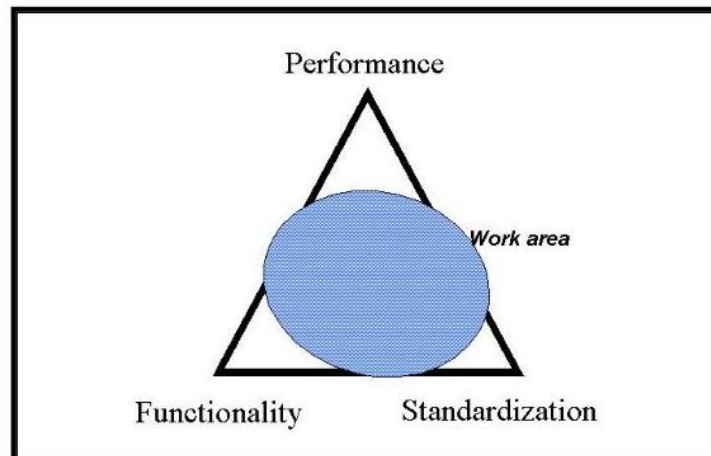


图 1: 带用户选项的三角形

对于该活动的定位，请查看图 1。这个三角形的角落有以下用户选项：

- 性能
- 功能
- 标准化。

在实践中，用户编写的程序与硬件紧密耦合，使用专用功能，以尽可能获得其环境所要求的最高性能。这限制了用户在目标硬件和控制软件可重用性方面的选择，并增加了培训投资。第二个用户选项允许提供非常广泛的软件功能。这对用户非常有帮助，但很少能带来高性能。此外，培训成本也增加了。

第三个角落，即标准化，主要集中于来自不同供应商的不同系统之间的重用性，包括集成、分布式和网络化系统，以及降低培训投资。由于该定义的一般特征，在不同架构上的性能可能不如硬编码那样最优。因此，不应期望标准化提供最大性能，但可以接近最大功能，这意味着三角形的底部非常短。

第一个规范作为运动控制功能块的独立库发布。它包括单轴和多轴的运动功能、多个多轴、若干管理任务以及状态图。该规范为用户提供了一个标准的命令集和与底层架构无关的结构。

这个结构可以用于许多平台和架构。通过这种方式，可以决定在开发周期的后期阶段将使用哪个架构。对于机器制造商来说，优势包括降低支持不同平台的成本，以及更独立地开发应用软件的自由，而不会限制机器的生产力。除了这些好处之外，系统维护更容易，教育周期更短。这是一个重大进步，并且越来越得到用户和供应商的认可。

随着第 1 部分的发布，大家认识到需要额外的功能。第 1 部分为一套与 TC2 任务组运动控制相关的规范提供了基础：

第 1 部分 - PLCopen 运动控制功能块

第 2 部分 - PLCopen 运动控制 - 扩展，在新发布的 2.0 中与第 1 部分合并

第 3 部分 - PLCopen 运动控制 - 用户指南

第 4 部分 - PLCopen 运动控制 - 协调运动

第 5 部分 - PLCopen 运动控制 - 回原点扩展

第 6 部分 - PLCopen 运动控制 - 流体动力扩展

随着底层文档的发布，第 1 部分 - PLCopen 运动控制功能块版本 2.0，第 2 部分 - PLCopen 运动控制扩展已被整合到基本文档中。

PLCopen 运动控制用户指南，第 3 部分，是对 PLCopen 运动控制功能块的补充，不应被视为独立文档。

1.1. Objectives

运动控制功能块适用于 IEC 61131-3 语言，考虑以下因素：

- 1 简单性 - 易于使用，面向应用程序构建者以及安装和维护
- 2 效率 - 功能块的数量，旨在提高设计（和理解）的效率
- 3 一致性 - 符合 IEC 61131-3 标准
- 4 通用性 - 硬件独立
- 5 灵活性 - 未来扩展/应用范围
- 6 完整性 - 非强制性但足够

1.1. Language context goals

- 专注于根据 IEC 61131-3 规范定义功能块接口和行为以及数据类型。
- 这些功能块和数据类型可以在所有 IEC 61131-3 语言中使用。
- 本文档中的示例以文本和图形 IEC 61131-3 语言进行了信息性呈现。
- 功能块的内容可以用任何编程语言实现（例如 IEC 61131-3 ST、C），甚至可以在固件或硬件中实现。因此，不应期望其内容在不同平台间可移植。
- 利用 PLCopen 交换标准，简化了由这些功能块和数据类型组成的可重用应用程序。
- 本规范应视为一个没有硬件依赖的开放框架。它在不同平台上的实现提供了开放性，例如完全集成、集中式或分布式系统。功能块的实际实现超出了本标准的范围。

1.1.2. Definition of a set of Function Blocks

一个基本问题涉及标准化功能块的粒度或模块化。极端情况是每个轴一个功能块与命令级功能。上述目标可以通过功能块的模块化设计更容易地实现。模块化创造了更高水平的可扩展性、灵活性和可重配置性。然后可以从这些大型块（派生功能块）创建，例如整个轴，以便于应用程序的构建和浏览。

如果可行，在此指定的功能块可以实现为一个函数（例如 MC_ReadParameter）。

1.1.3. Overview of the defined Function Blocks

以下表格概述了定义的功能块，分为管理（不驱动运动）和与运动相关的两个集合。

Administrative		Motion	
Single Axis	Multiple Axis	Single Axis	Multiple Axis
MC_Power	MC_CamTableSelect	MC_Home	MC_CamIn
MC_ReadStatus		MC_Stop	MC_CamOut
MC_ReadAxisError		MC_Halt	MC_GearIn
MC_ReadParameter		MC_MoveAbsolute	MC_GearOut
MC_ReadBoolParameter		MC_MoveRelative	MC_GearInPos
MC_WriteParameter		MC_MoveAdditive	MC_PhasingAbsolute
MC_WriteBoolParameter		MC_MoveSuperimposed	MC_PhasingRelative
MC_ReadDigitalInput		MC_MoveVelocity	MC_CombineAxis
MC_ReadDigitalOutput		MC_MoveContinuousAbsolute	
MC_WriteDigitalOutput		MC_MoveContinuousRelative	
MC_ReadActualPosition		MC_TorqueControl	
MC_ReadActualVelocity		MC_PositionProfile	
MC_ReadActualTorque		MC_VelocityProfile	
MC_ReadAxisInfo		MC_AccelerationProfile	
MC_ReadMotionState			
MC_SetPosition			
MC_SetOverride			
MC_TouchProbe			
MC_DigitalCamSwitch			
MC_Reset			
MC_AbortTrigger			
MC_HaltSuperimposed			

表 1：定义的功能模块概述

1.1.4. Compliance and Portability

本工作的目标是实现对作用于一个轴的运动控制功能块的可移植性，并在用户界面、输入/输出变量、参数和所用单位方面向用户提供与本文件中描述的相同功能。

允许在一个应用程序中从不同供应商结合多个 MC 库，由系统集成商或最终用户自行解决。

声称遵守此 PLCopen 规范的实现应提供一组（意即一个或多个）运动控制功能块，至少包括基本的输入和输出变量，标记为此文档中功能块定义表中的“B”。

对于更高级别的系统和将来的扩展，任何扩展的输入和输出变量的子集，标记为表中的“E”，均可实现。

特定于供应商的附加项标记为“V”。

有关合规性和 PLCopen 运动控制标志使用的更具体信息，请参阅附录 B。

- 基本输入/输出变量是强制性的，在表中用字母“B”标记 - 扩展输入/输出变量是可选的，在表中用字母“E”标记 - 特定于供应商的附加项在供应商的合规文档中用“V”标记

任何供应商都可以向本文件中规定的任何功能块添加特定于供应商的参数。

注意：

根据 IEC 61131-3 规范，输入变量可以未连接或未由用户参数化。在这种情况下，功能块将使用前一次调用功能块实例的值，或在第一次调用的情况下使用初始值。每个功能块输入都有一个定义的初始值，通常为 0。

在功能块和参数中列出的数据类型 REAL（例如速度、加速度、距离等）可以

TC2 任务组运动控制 2011 年 3 月 17 日 (C) 1999 - 2011 版权由 PLCopen 运动控制功能块版本 2.0, 已发布页 10/141 转换为 SINT、INT、DINT 或 LREAL, 而不被视为对该标准的不合规, 只要它在整个功能块和参数集合中是一致的。实现允许扩展数据类型, 只要基本数据类型保持不变。例如: WORD 可以更改为 DWORD, 但不能更改为 REAL。

在这个新版本的规范中, 任何不再指定的 FB 和输入可以保留在供应商的系统中, 以保持兼容性, 从而避免对现有 FB 的不兼容更改。

1.1.5. Length of names and ways to shorten them

有些系统只支持名称中有限数量的重要字符。针对这些情况, 提供了更短名称的规则。这些名称仍被视为合规, 尽管必须在认证文件中提及。

缩短名称的规则列表:

Command	Cmd
Position	Pos
Velocity	Vel
Acceleration	Acc
Deceleration	Decel
Absolute	Abs
Relative	Rel
Actual	Act
Superimposed	SupImp
Additive	Add
Parameter	Par
Continuous	Cont
GearRatioDenominatorM1	RatioDenM1
GearRatioNumeratorM1	RatioNumM1

结果符合的名称示例:

CommandAborted	CmdAborted
MC_MoveContinuousRelative	MC_MoveContRel
MC_ReadParameter	MC_ReadPar

1.1.6. History

Part 1 的第一次正式发布是在 2001 年 11 月。自那时以来, 收到了来自用户和实施者的反馈。2004 年, 决定发布 Part 1 的新版本 1.1, 包含将反馈纳入规范后的更改。该更新于 2005 年 4 月发布。2005 年 9 月, Part 2 - Extensions 的第一次正式发布也公布了。

在那之后, 两个部分都维护了一份更正和补充。在 2008 年, 接受了将 Part 1 和 Part 2 合并为一个新的 Part 1 的提案, 该新版本将以版本 2.0 的形式发布, 即您现在查看的文档。基本上, 这两个功能块集合已被合并。此外, 还进行了几个总体更改。这些更改包括 (但不限于):

- 对状态图表示的简化, 包括删除过渡命令
- 新输入 'ContinuousUpdate' 扩展了相关运动相关功能块的行为

- 采用描述导致输出'Active'行为的更改
- 在某些功能块中删除了中止模式
- 在 mcAborting 枚举中进行了更改
- MC_Phasing 和 MC_MoveContinuous 功能块分为相对和绝对版本
- 新功能块 MC_ReadMotionState、MC_ReadAxisInfo、MC_CombineAxes 和 MC_HaltSuperimposed
- 关于 Camming 的描述
- MC_CamTableSelect 的功能通过输入'ExecutionMode'得到扩展。'Periodic'的描述更为精准。
- MC_CamIn 的功能通过输入'MasterStartDistance'和'MasterSyncPosition'得到扩展。
- 新输入'MasterValueSource'和 MC_CamIn, MC_GearIn, MC_GearInPos, MC_ReadMotionState, MC_CombineAxes 中对应的数据类型

TC2 任务组运动控制 2011 年 3 月 17 日 (C) 1999-2011 版权归 PLCopen 运动控制功能块所有, 版本 2.0, 发布页 11/141

- MC_SetPosition 的输入'Mode'现在被称为'Relative' (与第 4 部分保持一致)
- 对于所有 PLCopen 运动控制规范, 统一了功能块、枚举元素、数据类型、结构、输入和输出的命名规范。
- 'InVelocity'、'InGear'、'InTorque'和'InSync'输出的行为在达到相应的 SET 值后发生变化
- 功能块 FBs MC_ReadAxisInfo、MC_PhasingRelative 和 MC_PhasingAbsolute 被添加到状态图未列出的功能块中
- 输入'Axis'、'Master'和'Slave'的描述发生了变化
- 输出'Busy'、'Error'和'ErrorID'的描述发生了变化

2. Model

以下功能块 (FB) 库旨在通过与 IEC 61131-3 标准中定义的语言元素一致的方式控制轴。工作组决定将一个轴的所有方面封装到单个功能块中并不实用。保留的解决方案是提供一组以命令为导向的功能块, 这些功能块参考了轴, 例如抽象数据类型 "Axis", 提供了灵活性、易用性和可重用性。

基于 IEC 61131-3 的实现 (例如通过功能块和 SFC) 将专注于功能块的接口 (外观和感觉/'代理')。本规范不定义功能块的内部操作。

这带来了在本章中描述的一些后果。

2.1. The State Diagram

以下图表规范性地定义了多个运动控制功能块“同时”激活时轴的高层行为。这种运动轮廓的组合在构建更复杂的轮廓或处理程序中的异常情况时非常有用。(在实际实现中, 可能会在较低级别定义其他状态)。

基本规则是运动命令总是按顺序执行, 即使 PLC 具有真正的并行处理能力。这些命令作用于轴的状态图。

轴始终处于定义的状态之一（见下图）。任何导致状态转换的运动命令都会改变轴的状态，因此，修改当前运动的计算方式。

状态图是轴实际状态的抽象层，可以与循环（PLC）程序中的 I/O 点的图像相比较。

当发出相应的运动命令时，状态的变化会立即反映出来。（注意：“立即”的响应时间依赖于系统，与轴的状态或软件中的抽象层相关）

该图集中于单个轴。多个轴功能块，如 MC_CamIn、MC_GearIn 和 MC_Phasing，从状态图的角度来看，可以看作是都处于特定状态的多个单轴。例如，CAM 主轴可以处于“连续运动”状态，而相应的从轴则处于“同步运动”状态。将从轴连接到主轴并不影响主轴的状态。

状态图中的箭头显示了状态之间可能的状态转换。由于发出命令而导致的状态转换用实线箭头表示。虚线箭头用于表示当轴的命令终止或发生与系统相关的转换（如与错误相关的）时的状态转换。使轴转移到相应运动状态的运动命令列在状态之上。当轴已经处于相应的运动状态时，这些运动命令也可以被发出。

关于状态的备注：

--	--

TC2 任务组运动控制 2011 年 3 月 17 日 (C) 1999 - 2011 版权所有 PLCopen 页 13/141

静止功率已开启，轴上没有错误，并且没有运动指令在轴上激活。

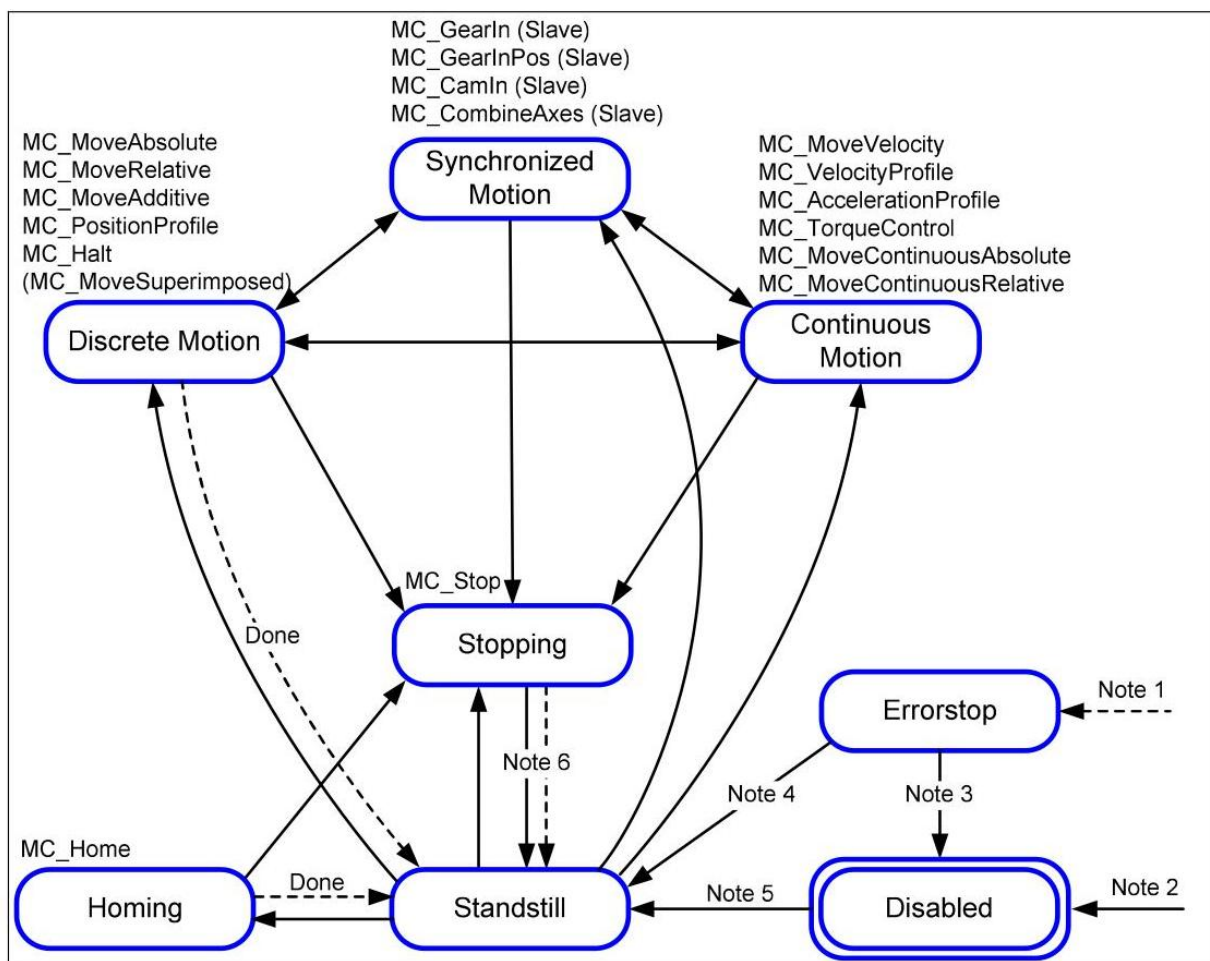
关于指令的备注：

MC_Stop	Calling the FB MC_Stop in state 'Standstill' changes the state to 'Stopping' and back to 'Standstill' when 'Execute' = FALSE. The state 'Stopping' is kept as long as the input 'Execute' is true. The 'Done' output is set when the stop ramp is finished.
MC_MoveSuperimposed	MC_MoveSuperimposed issued in state 'Standstill' brings the axis to state 'Diser Motion'. Issued in any other state the state of the axis is not influence
MC_GearOut, MC_CamOut	Change the state of the slave axis from 'SynchronizedMotion' to 'ContinuousMotion'; issuing one of these FBs in any other state generates an error.

在状态图中未列出的功能块不影响状态图，意味着每当它们被调用时，状态不会改变。它们是：

- MC_ReadStatus
- MC_ReadAxisError
- MC_ReadParameter
- MC_ReadBoolParameter
- MC_WriteParameter
- MC_WriteBoolParameter
- MC_ReadDigitalInput
- MC_ReadDigitalOutput
- MC_WriteDigitalOutput
- MC_ReadActualPosition
- MC_ReadActualVelocity
- MC_ReadActualTorque
- MC_ReadMotionState
- MC_SetPosition
- MC_SetOverride

- MC_AbortTrigger
- MC_TouchProbe
- MC_DigitalCamSwitch
- MC_CamTableSelect
- MC_ReadAxisInfo
- MC_PhasingRelative
- MC_PhasingAbsolute
- MC_HaltSuperimposed



- Note 1: From any state. An error in the axis occurred.
- Note 2: From any state. MC_Power.Enable = FALSE and there is no error in the axis.
- Note 3: MC_Reset AND MC_Power.Status = FALSE
- Note 4: MC_Reset AND MC_Power.Status = TRUE AND MC_Power.Enable = TRUE
- Note 5: MC_Power.Enable = TRUE AND MC_Power.Status = TRUE
- Note 6: MC_Stop.Done = TRUE AND MC_Stop.Execute = FALSE

图 2: FB 状态图

2.2. Error handling

所有对驱动/运动控制的访问均通过功能块进行。这些功能块在内部提供对输入数据的基本错误检查。具体是如何实现的取决于具体的实现。例如，如果 MaxVelocity 设置为 6000，并且 FB 的速度输入设置为 10,000，则系统要么减慢速度，要么生成错误。在智能驱动通过网络与系统连接的情况下，MaxVelocity 参数可能存储在驱动中。FB 必须处理驱动内部生成的错误。在另一种实现中，MaxVelocity 值可以存储在本地。在这种情况下，FB 将在本地生成错误。

2.2.1. Centralized versus Decentralized

在使用运动控制功能块时，可以选择集中式和分散式错误处理方法。

集中式错误处理用于简化功能块的编程。错误反应与发生错误的实例无关，始终相同。

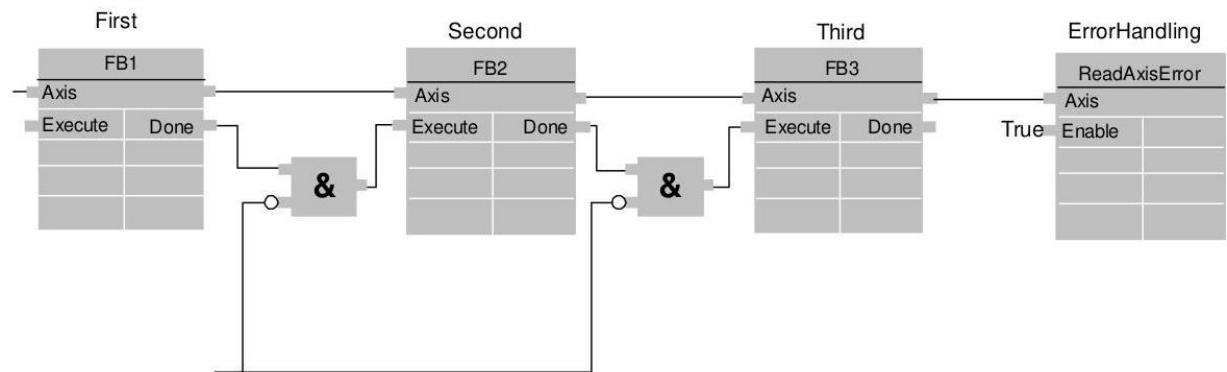


图 3：带有集中式错误处理的功能块

分散式错误处理则允许根据发生错误的功能块采取不同的反应。

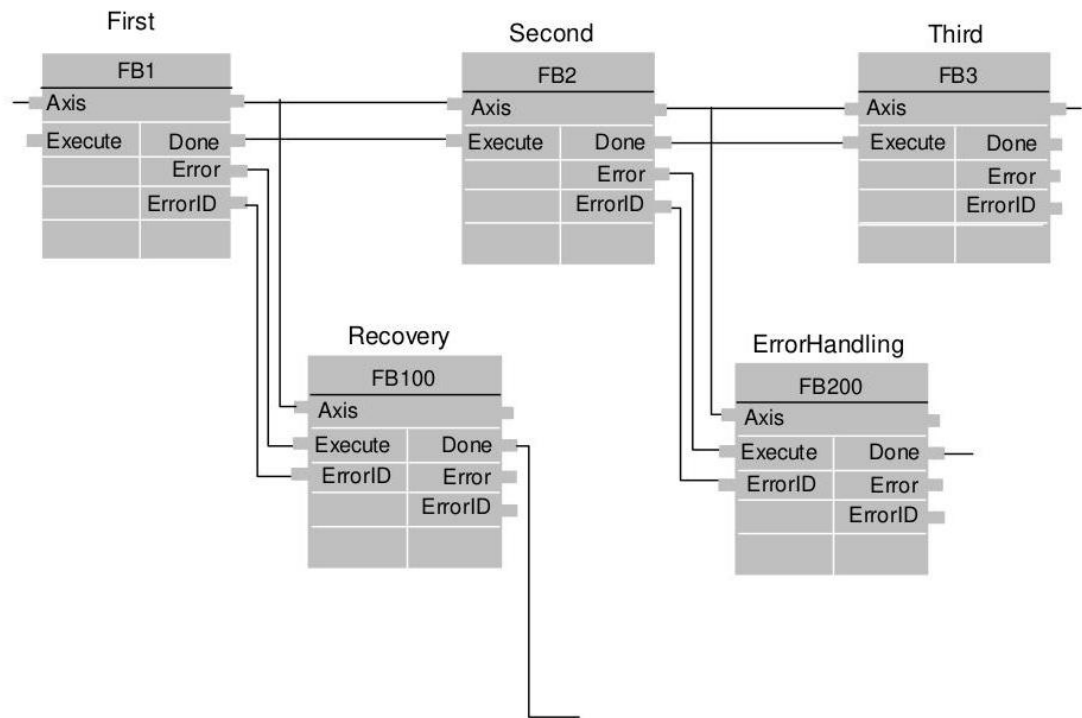


图 4：带有分散式错误处理的功能块

2.2.2. Buffered Commands

所有缓冲的命令将在适用的轴移动到状态“ErrorStop”时被中止。适用的中止功能块 (FBs) 的“Error”输出被设置。任何后续命令将被拒绝，错误输出被设置（不允许的操作 - 请参见状态图）。

如果一个 FB 出现错误（例如由于参数设置错误），则错误输出被设置，行为取决于应用程序。例如，在两个 FB 的情况下，第一个 FB 实例 FB1 在一个轴上执行任何运动命令。在同一轴上，在缓冲模式下启动第二个 FB 实例 FB2 的新命令。该命令被缓冲并等待 FB1 完成。在第一个实例 FB1 完成其命令之前，让以下情况之一发生：

1. 该轴进入状态“ErrorStop”（例如由于后续错误或过热）。FB1 设置输出“Error”。FB2（以及任何其他等待在该轴上执行缓冲命令的 FB 实例）设置其“Error”输出，并通过输出“ErrorID”显示它无法执行其任务，因为该轴处于不允许其操作的状态。所有缓冲的命令被清除。在 MC_Reset 重置轴错误后，它可以再次被控制。

2. FB1 设置其“Error”输出（例如由于无效的参数设置）。FB2 变为活跃状态，并在此之后立即执行给定命令，应用程序应该处理错误情况。

2.2.3. Timing example for the 'Enable' input

示例 1：图左侧显示了正常操作。图右侧在操作过程中发生了错误。这个错误迫使'Valid' 输出被重置。输出'Busy' 保持高电平。在错误被重置后，正常操作程序将被恢复，这可能需要一些时间。

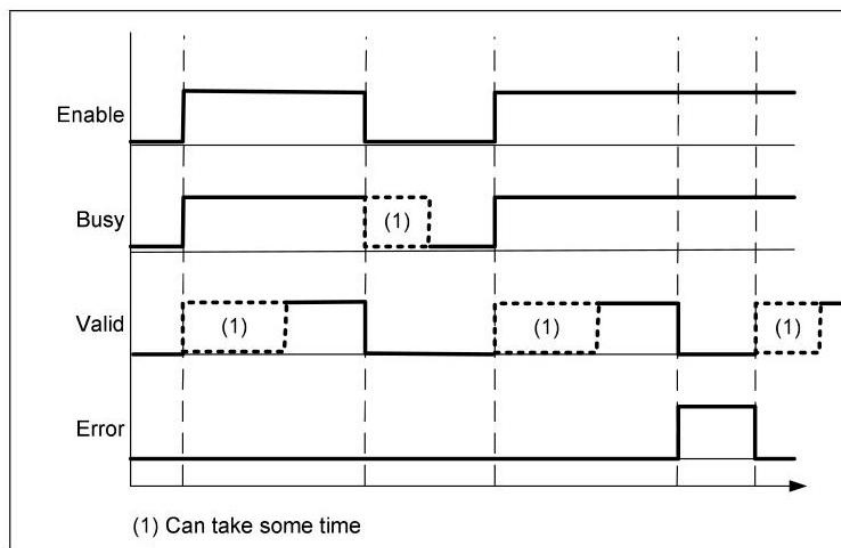


图 5：带有'Enable'输入的错误处理示例

第二个示例在右侧显示了一个无法自动清除的错误。在错误被设置后，输出'Busy' 和'Valid' 被重置。FB 需要在'Enable' 输入上有一个上升沿才能继续。

2.3. Definitions

在本文中使用以下值级别：指令值/设定值/实际值：

- 指令值 - 是基于功能块输入的值，可以作为配置生成器的（一个）输入。
- 设定值 - 位于“较低”的水平，更接近于执行器。它是即将发送到伺服环路（例如，执行器）的最新值（由配置生成器生成），例如执行器将使用的下一个值。
- 实际值 - 是反馈系统中可用的最新值。

2.4. FB interface

[illegible]

Table 2: General Rules

‘执行’/‘完成’风格的功能块 (FBs) 的行为如下所示:

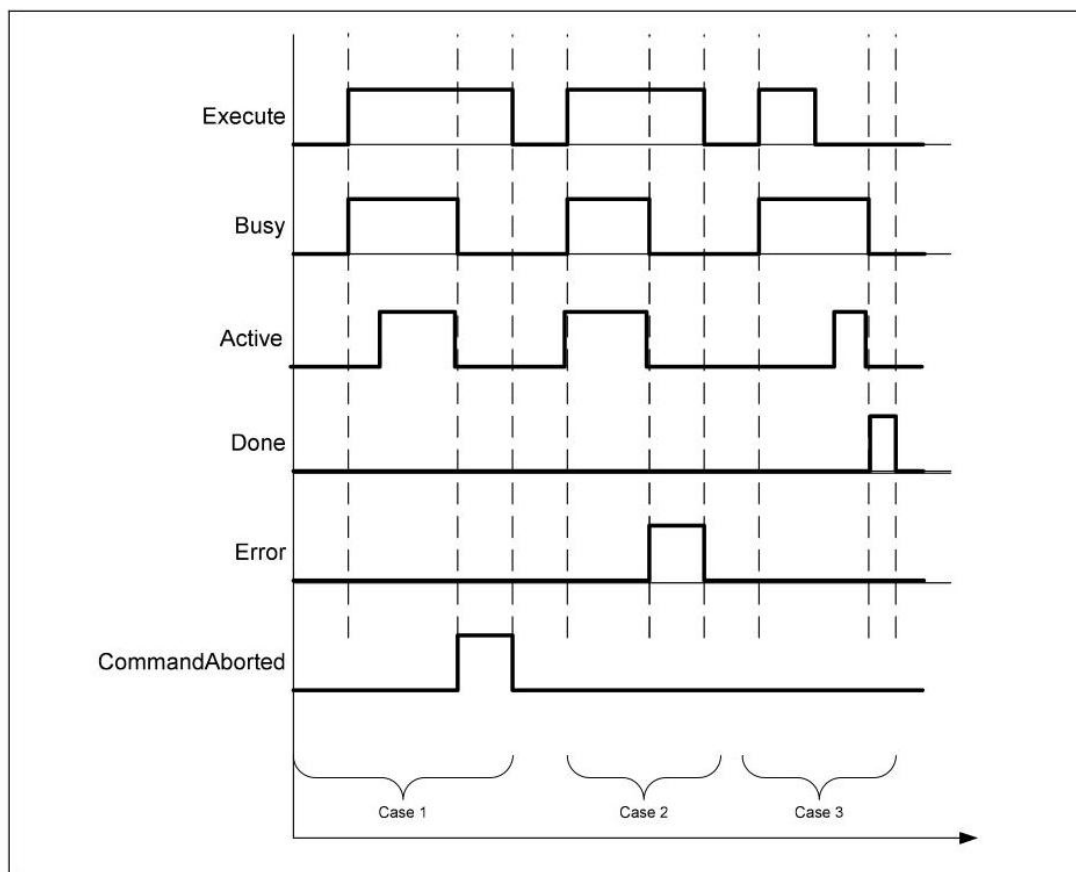


图 7: '执行' / '完成' 在相关功能块中的行为

‘执行’ / ‘Inxxx’ 风格的功能块 (FBs) 的行为如下所示:

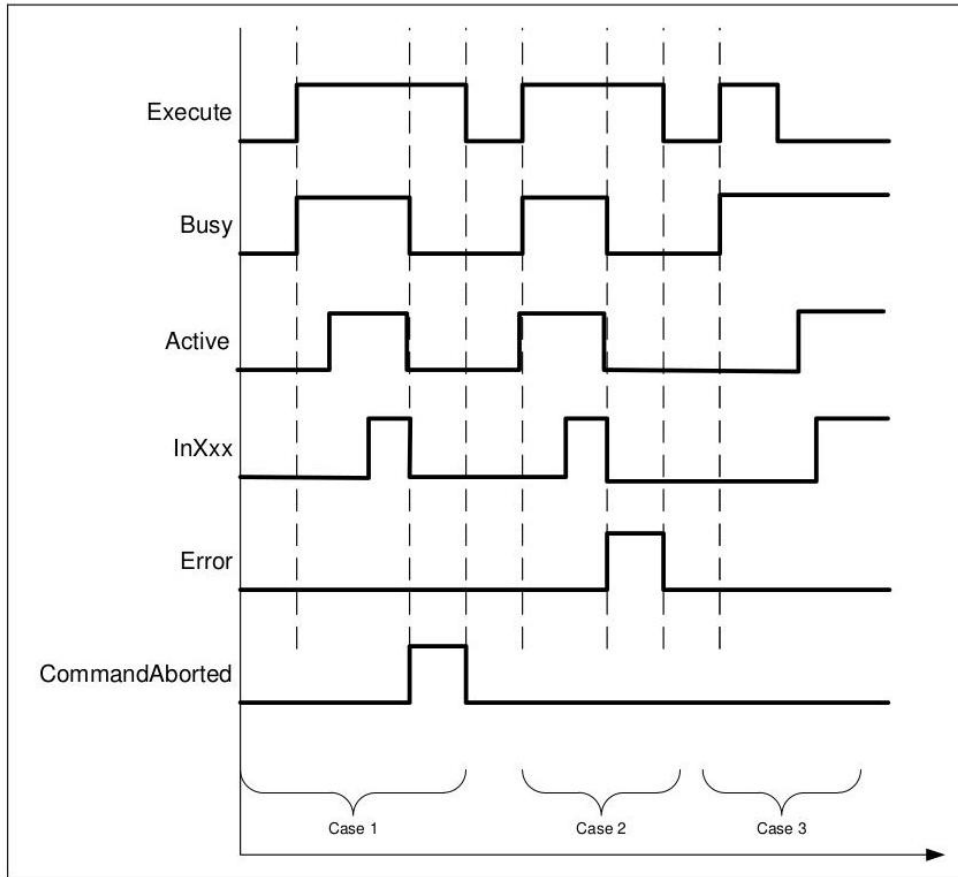


图 8: '执行' / 'Inxxx' 在相关功能块中的行为

2.4.2. Aborting versus Buffered modes

一些功能块 (FBs) 有一个名为“BufferMode”的输入。通过这个输入，功能块可以在“非缓冲模式”（默认行为）或“缓冲模式”下工作。这两种模式之间的区别在于它们何时开始行动：- 非缓冲模式下的命令会立即执行，即使这会中断另一项运动。缓冲模式下的命令则会等待，直到当前功能块设置其“完成”输出（或“就位”或“速度合适”等）。

缓冲模式还有多种选项。因此，该输入是类型为 MC_BUFFER_MODE 的枚举 (ENUM)。已确定以下模式：

- Aborting	Default mode without buffering. The next FB aborts an ongoing motion and the command affects the axis immediately. The buffer is cleared
- Buffered	The next FB affects the axis as soon as the previous movement is 'Done'. There is no blending.
- BlendingLow	The next FB controls the axis after the previous FB has finished (equivalent to 'Buffered'), but the axis will not stop between the movements. The velocity is blended with the lowest velocity of both commands (1 and 2) at the first end-position (1)
- BlendingPrevious	blending with the velocity of FB 1 at end-position of FB 1
- BlendingNext	blending with velocity of FB 2 at end-position of FB 1
- BlendingHigh	blending with highest velocity of FB 1 and FB 2 at end-position of FB 1

ENUM 被定义如下：

No.	MC_BUFFER_MODE	Description
0	mcAborting	Start FB immediately (default mode)
1	mcBuffered	Start FB after current motion has finished
2	mcBlendingLow	The velocity is blended with the lowest velocity of both FBs
3	mcBlendingPrevious	The velocity is blended with the velocity of the first FB
4	mcBlendingNext	The velocity is blended with velocity of the second FB
5	mcBlendingHigh	The velocity is blended with highest velocity of both FBs

Table 3: The ENUM type MC_BUFFER_MODE

允许在这些定义的枚举后添加供应商特定的扩展。

附录 A 中列出的示例描述了这些模式的不同行为。

下表概述了对定义的功能块的影响：

Function block	Can be specified as a buffered command	Can be followed by a buffered command	Relevant signal to activate the next buffered FB
MC_Power	No	Yes	Status
MC_Home	Yes	Yes	Done
MC_Stop	No	Yes	Done AND NOT Execute
MC_Halt	Yes	Yes	Done
MC_MoveAbsolute	Yes	Yes	Done
MC_MoveRelative	Yes	Yes	Done
MC_MoveAdditive	Yes	Yes	Done
MC_MoveSuperimposed	No	No	—
MC_HaltSuperimposed	No	No	—
MC_MoveVelocity	Yes	Yes	InVelocity
MC_MoveContinuousAbsolute & MC_MoveContinuousRelative	Yes	Yes	InEndVelocity
MC_TorqueControl	Yes	Yes	InTorque
MC_PositionProfile	Yes	Yes	Done
MC_VelocityProfile	Yes	Yes	Done
MC_AccelerationProfile	Yes	Yes	Done
MC_CanIn	Yes	Yes - (in single mode)	EndOfProfile
MC_CanOut	No	Yes	Done
MC_GearIn	Yes	Yes	InGear
MC_GearOut	No	Yes	Done
MC_GearInPos	Yes	Yes	InSync
MC_PhasingRelative & MC_PhasingAbsolute	Yes	No	—
MC_CombineAxes	Yes	Yes	InSync

Table 4: Overview of the buffered commands on the relevant FBs

注意：此处未列出的（管理）FB 基本上不进行缓冲，也不能跟随缓冲 FB。然而，供应商可以选择支持各种缓冲/混合模式。

如果一个正在进行的运动被另一个运动中止，则可能由于减速限制导致制动距离不足。

在旋转轴中，可以添加模数。在轴不应改变方向并反转以达到命令位置的情况下，模数轴可以回到绝对位置所指定的最早重复。

在线性系统中，导致的超调可以通过反转来解决，因为每个位置都是唯一的，因此不需要添加模数以达到正确位置。

2.4.3. AXIS_REF Data type

AXIS_REF 是一个包含对应轴信息的结构体。它在本文档中定义的所有运动控制功能块中用作 VAR_IN_OUT。此结构体的内容依赖于实现，最终可能为空。如果该结构体中有元素，供应商应支持对它们的访问，但这超出了本文档的范围。此结构体的刷新率也依赖于实现。根据 IEC 61131-3，允许为活动的功能块切换 AXIS_REF，例如从 Axis1 切换到 Axis2。然而，这种行为在不同平台上可能有所不同，不鼓励这样做。

AXIS_REF 数据类型声明：

```

TYPE
  AXIS_REF : STRUCT
    (内容依赖于实现)
  END_STRUCT
END_TYPE

```

示例：

```

TYPE
  AXIS_REF : STRUCT
    AxisNo: UINT;

```

```

AxisName: STRING (255);
...
END_STRUCT
END_TYPE

```

2.4.4. Technical Units

对物理量的唯一规格是在长度单位（记作 [u]）上，该单位与其导数一致，即（速度 [u/s]；加速度 [u/s²]；震动 [u/s³]）。然而，单位 [u] 并没有具体规定（取决于制造商）。仅仅规定了它与其他单位的关系。

2.4.5. Why the command input is edge sensitive

不同功能块的“Execute”输入在本文中描述的情况下总是通过其上升沿触发该功能。原因在于，边缘触发的“Execute”可能在执行先前命令期间命令新的输入值。这种方法的优点是能够精确管理运动命令执行的时刻。在集中式和分散式轴管理模型中，组合不同的功能块变得更加容易。“Done”输出可以用于触发运动的下一部分。

下面给出的示例旨在解释功能块执行的行为。该图说明了控制同一轴的三个功能块“First”、“Second”和“Third”的顺序。这三个功能块可以是各种绝对或相对的移动命令。当“First”完成时，其上升输出‘First.Done’触发‘Second.Execute’。输出‘Second.Done’和‘In13’触发‘Third.Execute’。

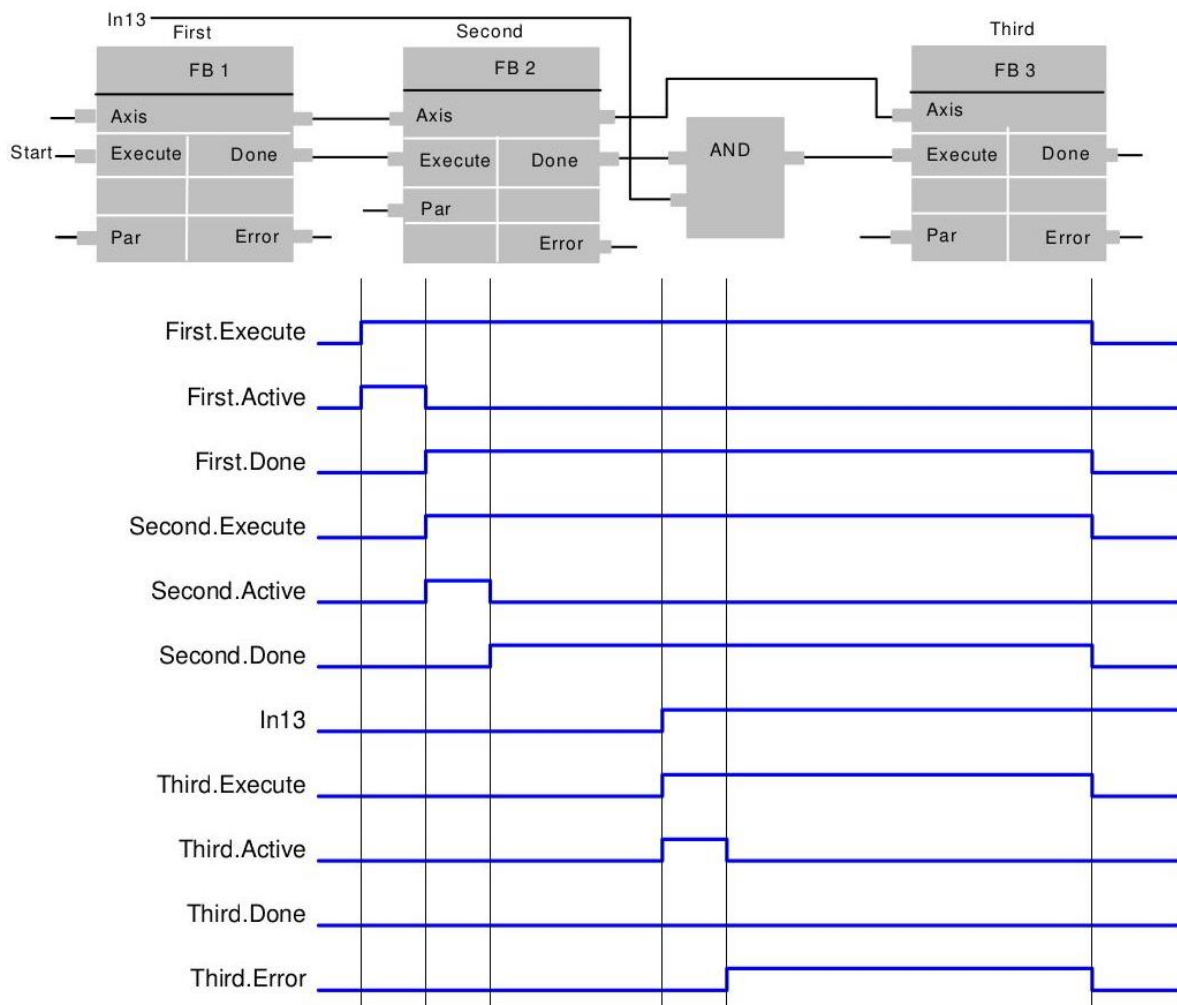


图 9: 执行复杂运动的功能块

2.4.6. The input 'ContinuousUpdate'

如前一章所述，输入'Execute'会触发一次新的运动。在此输入的上升沿，其他功能块输入的值会定义该运动。在版本 1.1 之前，有一个通用规则，即对这些输入参数的后续更改不会影响正在进行的运动。

然而，有许多应用示例需要连续更改参数。用户可以重新触发功能块的'Execute'输入，但这使应用变得复杂。

因此，引入了输入'ContinuousUpdate'。它是所有适用功能块的扩展输入。如果它为 TRUE，当功能块被触发（上升'Execute'）时，只要保持为 TRUE，功能块就会使用输入变量的当前值并将其应用于正在进行的运动。这不会影响功能块的总体行为，也不会影响状态图。换句话说，它只会影响正在进行的运动，且其影响在功能块不再'Busy'或输入'ContinuousUpdate'被设置为 FALSE 时结束。（备注：某些输入如'BufferMode'可能并不打算在每个周期更改。然而，这必须在应用中处理，并且在规范中并不被禁止。）

如果在'Execute'输入的上升沿，'ContinuousUpdate'为 FALSE，则在整个运动过程中输入参数的变化将被忽略，适用以前版本的原始行为。

'ContinuousUpdate'并不是功能块'Execute'输入的重新触发。对先前中止、停止或完成的功能块的重新触发，将重新获得对轴的控制并修改其状态图。与此相反，'ContinuousUpdate'仅影响正在进行的运动。

此外，相对输入的'ContinuousUpdate'（例如，MC_MoveRelative 中的'Distance'）始终指的是初始条件（在'Execute'的上升沿）。示例：

- MC_MoveRelative 在'Position' 0 处启动，'Distance' 为 100，'Velocity' 为 10，且'ContinuousUpdate' 设置为 TRUE。'Execute' 被设置，因此运动开始移动到位置 100。
- 在执行运动的同时（让驱动器处于位置 50），输入的“距离”被更改为 130，“速度”更改为 20。
- 轴会加速（至新的“速度”20），并在“位置”130 停止，并设置输出“完成”，不接受任何新值。

2.5. Example 1: the same Function Block instance controls different motions of an axis

图 10: 与 SFC 一起使用单个 FB 的示例，其中功能块 FB1 用于控制“AxisX”，并设置三个不同的'速度'值。在序列功能图（SFC）中，'速度' 10、20 和 0 被分配给 V。为了通过上升沿触发'执行'输入，变量 E 被逐步设置和重置。

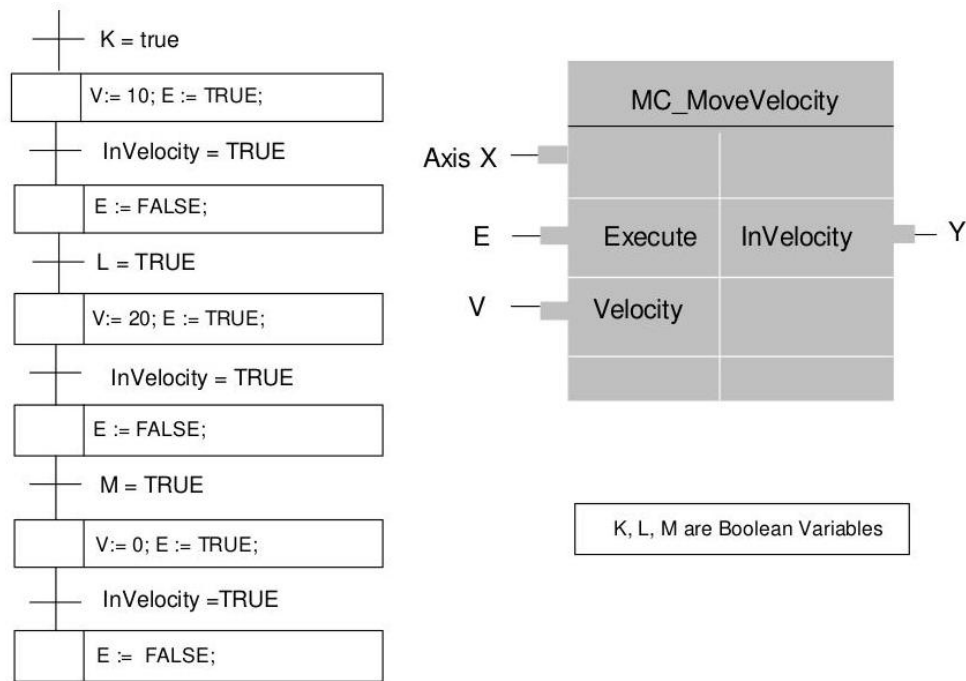


图 10: 与 SFC 一起使用单个 FB
以下时间图解释了它是如何工作的。

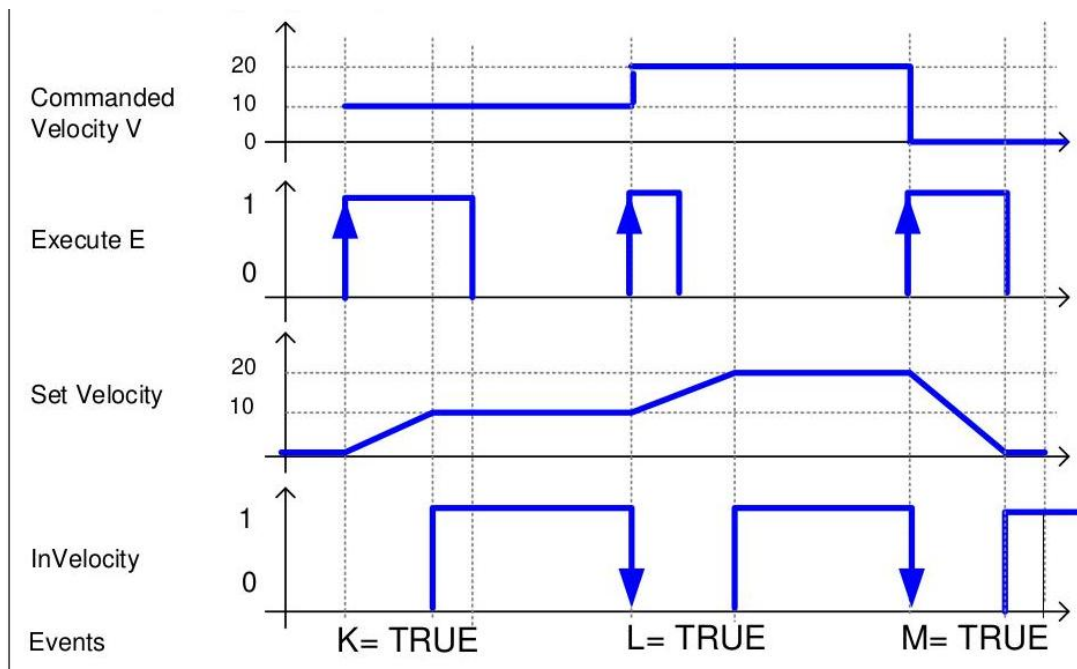


图 11: 单个 FB 使用的时间图

注意: 如果在 'InVelocity' 被设置的情况下, 使用相同的命令速度重新触发执行输入, 则输出 'InVelocity' 的行为取决于实现 (例如: 重置一个周期或完全不重置)

2.6. Example 2: different Function Block instances control the motions of an axis

与同一轴相关的不同实例可以控制轴上的运动。每个实例将“负责”全局轮廓的一个部分。

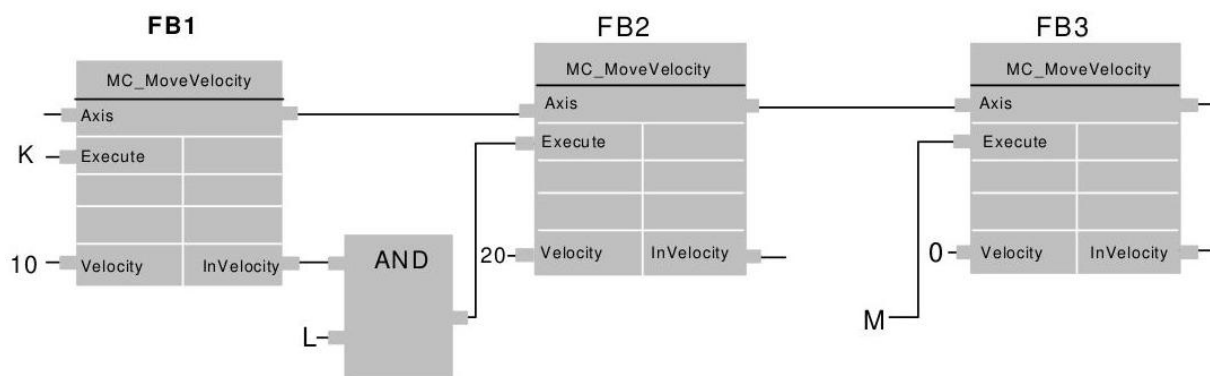


图 12: 级联功能块的示例
相应的时序图:

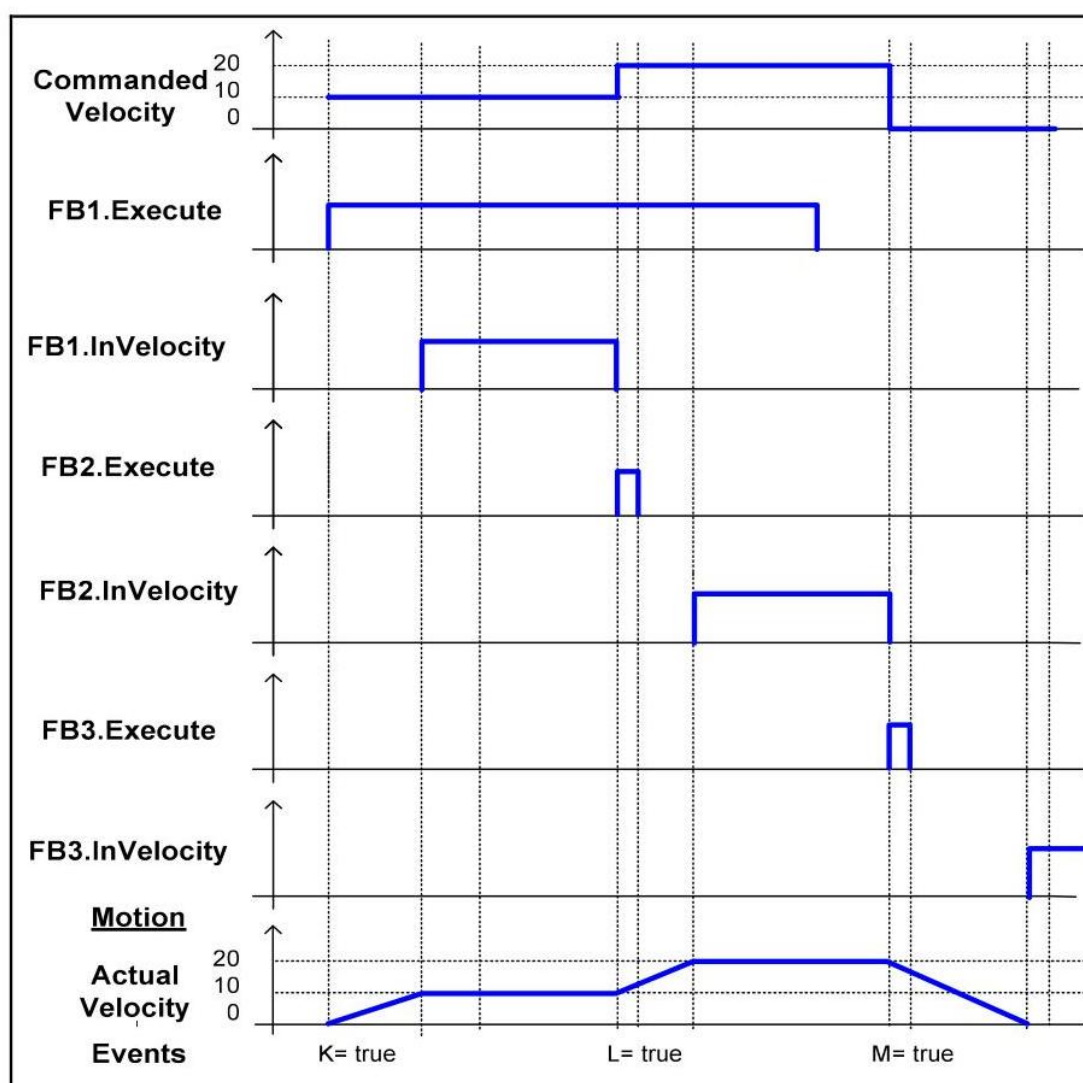


图 13: 示例级联功能块的时序图
用 LD 编写的相应解决方案可以如下所示:

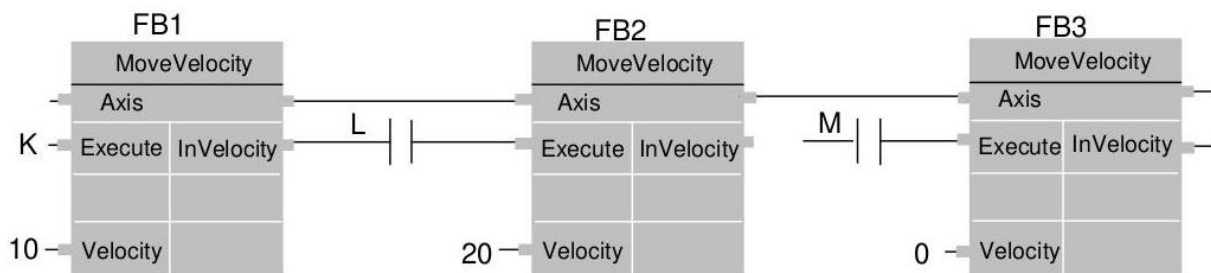


图 14: 带有 LD 的级联功能块示例

3. Single-Axis Function Blocks

3.1. MC_Power

FB-Name		MC_Power		
This Function Block controls the power stage (On or Off).				
VAR_IN_OUT				
	B	Axis	AXIS REF	Reference to the axis
VAR INPUT				
	B	Enable	BOOL	As long as 'Enable' is true, power is being enabled
	E	EnablePositive	BOOL	As long as 'Enable' is true, this permits motion in positive direction
	E	EnableNegative	BOOL	As long as 'Enable' is true, this permits motion in negative direction
VAR OUTPUT				
	B	Status	BOOL	Effective state of the power stage
	E	Valid	BOOL	If true, a valid set of outputs is available at the FB
	B	Error	BOOL	Signals that an error has occurred within the Function Block
	E	ErrorID	WORD	Error identification

注意事项:

- 'Enable' 输入使驱动器中的功率阶段启用，而不是 FB 本身
- 如果在'Disabled' 状态下调用 MC_Power FB，且'Enable' = TRUE，则轴状态将变为'Standstill'。
- 当 Command 为 TRUE 且 Status 仍然为 false 时，可以用定时器 FB 和与功能（带反转状态输入）设置一个错误变量。这表明功率阶段存在硬件问题。
- 如果电源故障（即使在运行期间），将产生转移到'ErrorStop' 状态。

'EnablePositive' 和 'EnableNegative' 都对电平敏感。

'EnablePositive' & 'EnableNegative' 都可以为真。

每个轴只应发出 1 个 FB MC_Power。

3.3. MC_Stop

运动控制功能模块版本 2.0，发布于第 31 页/141 页

FB-Name		MC_Stop	
This Function Block controls the power stage (On or Off). It allows stop signaling Function Block outputs. While the axis is in state "Stopping", all other FBs can position any motion in its motor area. After the axis has reached "Statically stop", the "Stop" output is set to "FALSE" automatically. The axis can move in the state "Stopping" as long as "Release" is set to "FALSE" or "Statically stop" as not yet reached. As soon as "Stop" and "Release" is FALSE the axis goes to state "Statically stop".			
B	Axis	AXIS REF	Reference to the axis
VAR IN_OUT			
B	Enable	BOOL	Start the action of setting stop
E	EnablePositive	BOOL	Value of the "EnablePositive" input
E	EnableNegative	BOOL	Value of the "EnableNegative" input
VAR OUTPUT			
B	Status	BOOL	Axis currently stopped
B	Stop	BOOL	The FB is finished and new output values are to be assigned
E	Valid	BOOL	Movement authorized by switching of power stage (positioning is started)
B	Error	BOOL	Signals that an error has occurred within the Function Block
E	ErrorID	WORD	Error identification

- 注意：
- 1. 此功能块主要用于紧急停止功能或异常情况
 - 2. 只要“执行”保持高电平，轴将保持在“停止”状态，并且可能不会执行任何其他运动命令。
 - 3. 如果“减速” = 0，功能块的行为为具体实现相关

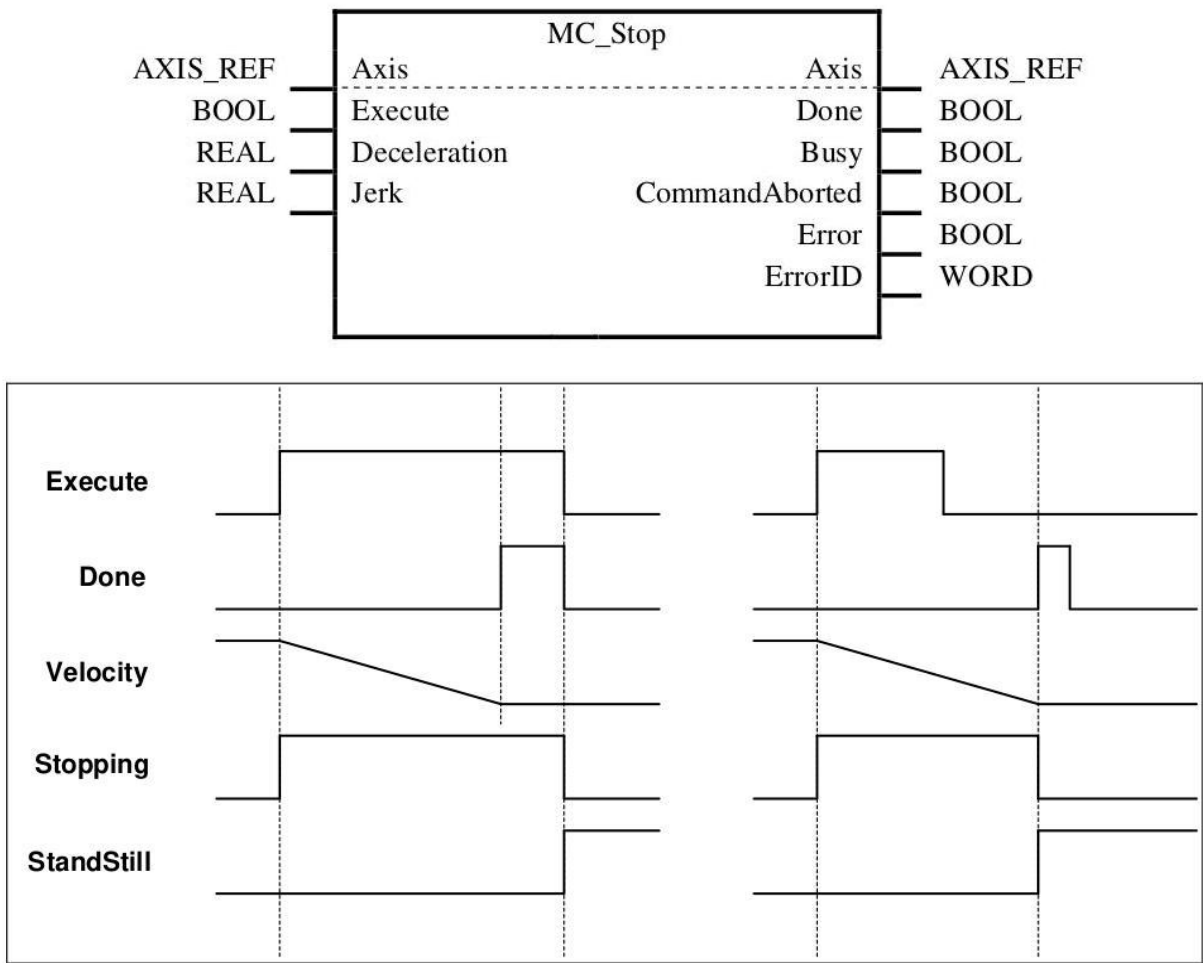


图 15: MC_Stop 时序图

TC2 任务组运动控制 2011 年 3 月 17 日 (C) 1999 - 2011 版权归 PLCopen 所有

下面的例子展示了与 MC_MoveVelocity 结合时的行为。

- a) 一个旋转轴通过 FB MC_Stop 进行缓慢下降。
- b) 只要 MC_Stop 参数“执行” = TRUE，轴将拒绝运动命令。FB MC_MoveVelocity 报告一个错误，指示 MC_Stop 命令忙碌。

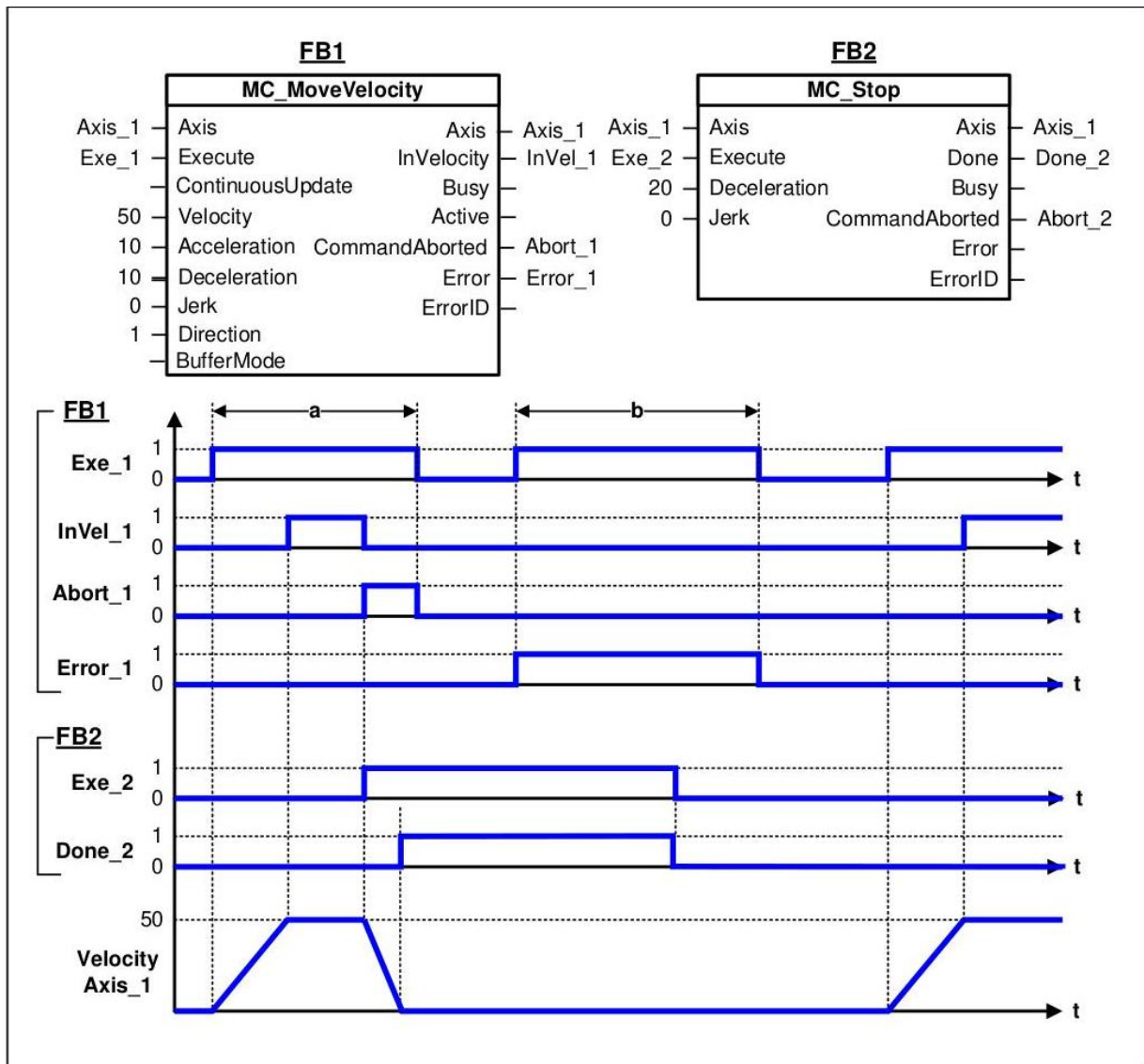


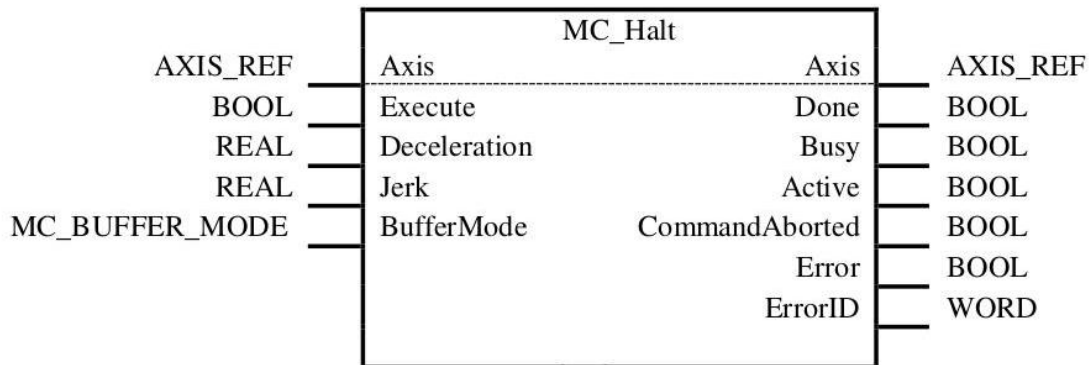
图 16: MC_Stop 与 MC_MoveVelocity 结合时的行为

3.4. MC_Halt

FB-Name		MC_Halt	
This Function Block commands a controlled motion stop. The axis is moved to the state 'DiscreteMotion', until the velocity is zero. With the 'Done' output set, the state is transferred to 'Standstill'			
VAR_IN_OUT			
B	Axis	AXIS_REF	Reference to the axis
VAR_INPUT			
B	Execute	BOOL	Start the action at rising edge
E	Deceleration	REAL	Value of the 'Deceleration' [$\frac{\text{u}}{\text{s}^2}$]
E	Jerk	REAL	Value of the 'Jerk' [$\frac{\text{u}}{\text{s}^3}$]
E	BufferMode	MC BUFFER MODE	Defines the chronological sequence of the FB. See 2.4.2 Aborting versus Buffered modes
VAR OUTPUT			
B	Done	BOOL	Zero velocity reached
E	Busy	BOOL	The FB is not finished and new output values are to be expected
E	Active	BOOL	Indicates that the FB has control on the axis
E	CommandAborted	BOOL	'Command' is aborted by another command
B	Error	BOOL	Signals that an error has occurred within the Function Block
E	ErrorID	WORD	Error identification

- MC_Halt 用于在正常操作条件下停止轴。在非缓冲模式下，可以在轴减速期间设置另一个运动命令，这将中止 MC_Halt 并立即执行。
- 如果此命令处于活动状态，则可以发出下一个命令。例如，一辆无驾驶员的车辆检测到障碍物并需要停车。发出 MC_Halt 命令。在达到“静止”之前，障碍物被移除，且可以通过设置另一个运动命令

继续运动，因此车辆不会停车。



下面的示例显示了与 MC_MoveVelocity 结合使用时的行为。

a) 一个旋转轴通过功能块 MC_Halt 减速。

b) 另一个运动命令覆盖了 MC_Halt 命令。MC_Halt 允许这样做，而与 MC_Stop 形成对比。该轴可以再次加速，而无需达到“静止”。

3.5. MC_MoveAbsolute

FB Name		MC_MoveAbsolute	
		This Function Block commands a controlled motion to a specified absolute position.	
		VAR_IN_OUT	
1	Axis	AXIS_REF	Reference to the axis
		VAR_INPUT	
1	Execute	BOOL	Start the motion at rising edge
2	ContinuousUpdate	BOOL	See 2.4.5 The input "ContinuousUpdate"
3	Position	REAL	Commanded Position for the motion (in technical unit [s]) (negative or positive)
4	Velocity	REAL	Value of the maximum "Velocity" (not necessarily constant) [mm/s]
5	Acceleration	REAL	Value of the "Acceleration" (always positive) (increasing energy of the motor) [m/s²]
6	Deceleration	REAL	Value of the "Deceleration" (always positive) (decreasing energy of the motor) [m/s²]
7	Jerk	REAL	Value of the "Jerk" [m/s³] (always positive)
8	Direction	MC_DIRECTION	Error type (1 of 4 values: noPositionDirection, noStart, noStop, noSignetDirection, noCommandDirection)
9	BufferMode	MC_BUFFER_MODE	Define the interrupting response of the FB. See 2.4.2 "Aborting versus Buffered modes"
		VAR_OUTPUT	
1	Done	BOOL	Commanded position finally reached
2	Busy	BOOL	The FB is no finished and new output values can't be requested
3	Active	BOOL	Indicates that the FB has control on the axis
4	CommandAborted	BOOL	"Command" is aborted by another command
5	Error	WORD	Signals that an error has occurred within the Function Block
6	ErrorID	WORD	Error identification

Note: The calculations with velocity and/or for higher output are results. If there is only one mathematical solution to reach the "CommandedPosition" (like in linear motion), the value the next "Direction" is chosen. For results with acceleration and/or deceleration, the value of the next "Direction" is chosen. For results with acceleration and/or deceleration, the value of the next "Direction" is chosen.

rate: The position before the start of "CommandPosition" of MC_MoveAbsolute to the command

AXIS_REF			
	MC_MoveAbsolute		
	AxisAxis		AXIS_REF
BOOL	Execute	Done	BOOL
BOOL	ContinuousUpdate	Busy	BOOL
REAL	Position	Active	BOOL
REAL	Velocity	CommandAborted	BOOL
REAL	Acceleration	Error	BOOL
REAL	Deceleration	ErrorID	WORD
REAL	Jerk		
MC_DIRECTION	Direction		
MC_BUFFER_MODE	BufferMode		

TC2 任务组运动控制 2011 年 3 月 17 日 (C) 1999 - 2011 版权归 PLCopen 所有第 35 页/共 141 页
下图展示了两个绝对移动功能块的组合示例：

1. 时序图的左侧部分说明了在第一个功能块之后调用第二个功能块的情况。如果第一个功能块达到命令位置 6000（并且速度为 0），则输出“完成”导致第二个功能块移动到位置 10000。

2. 时序图的右侧部分说明了在第一个功能块仍在执行期间启动第二个移动功能块的情况。在这种情况下，第一个运动被测试信号在第一个功能块的恒定速度下中断并中止。第二个功能块直接移动到位置 10000，尽管位置 6000 尚未达到。

MoveAbsolute - Example

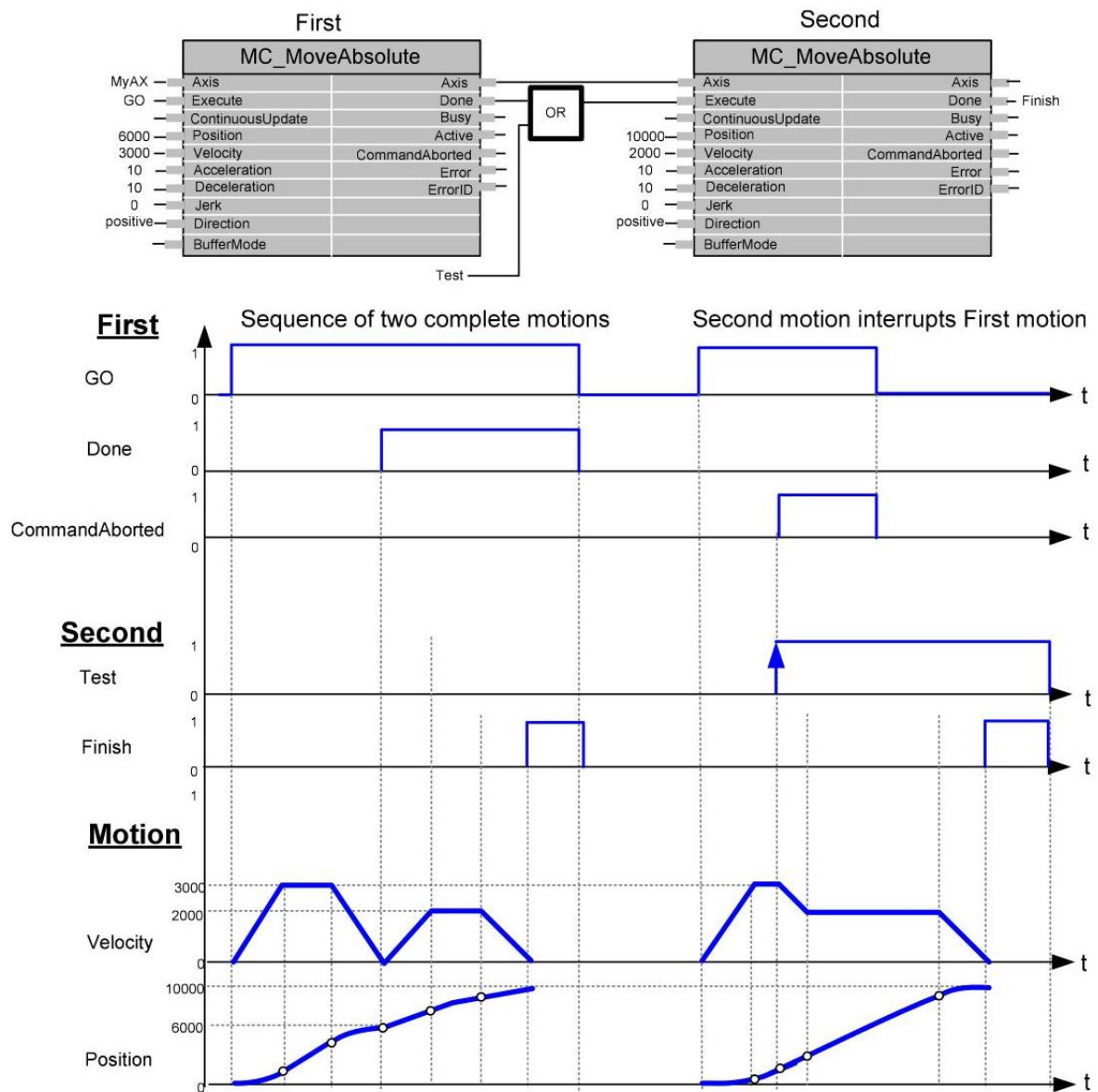
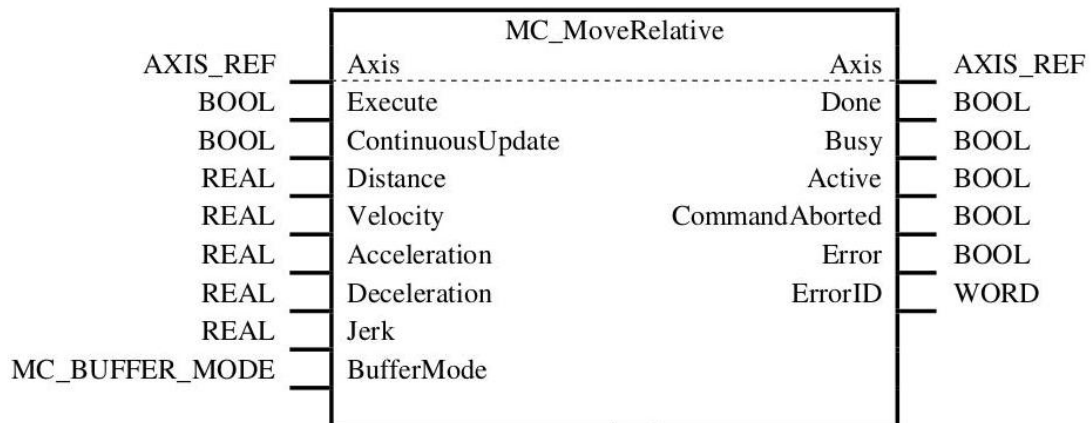


图 18: MC_MoveAbsolute 时序图

图例说明: 这些示例基于功能块的两个实例: 实例“第一个”和“第二个”。

3.6. MC_MoveRelative

FB-Name		MC_MoveRelative		
This Function Block commands a controlled motion of a specified distance relative to the set position at the time of the execution.				
VAR IN OUT				
	B	Axis	AXIS_REF	Reference to the axis
VAR_INPUT				
	B	Execute	BOOL	Start the motion at rising edge
	E	ContinuousUpdate	BOOL	See 2.4.6 The input 'ContinuousUpdate'
	B	Distance	REAL	Relative distance for the motion (in technical unit [u])
	E	Velocity	REAL	Value of the maximum velocity (not necessarily reached) [u/s]
	E	Acceleration	REAL	Value of the acceleration (increasing energy of the motor) [u/s ²]
	E	Deceleration	REAL	Value of the deceleration (decreasing energy of the motor) [u/s ²]
	E	Jerk	REAL	Value of the Jerk [u/s ³]
	E	BufferMode	MC BUFFER MODE	Defines the chronological sequence of the FB. See 2.4.2 Aborting versus Buffered modes
VAR OUTPUT				
	B	Done	BOOL	Commanded distance reached
	E	Busy	BOOL	The FB is not finished and new output values are to be expected
	E	Active	BOOL	Indicates that the FB has control on the axis
	E	CommandAborted	BOOL	'Command' is aborted by another command
	B	Error	BOOL	Signals that an error has occurred within the Function Block
	E	ErrorID	WORD	Error identification
Notes: This action completes with velocity zero if no further actions are pending				



下图展示了两个相对移动功能块组合的示例。

1. 时间图的左侧部分描述了在第一个功能块之后调用第二个功能块的情况。如果第一个功能块达到指令距离 6000（且速度为 0），那么输出“完成”会导致第二个功能块移动指令距离 4000，并将轴移动到最终位置 10000。

2. 时间图的右侧部分说明了如果第二个移动功能块在第一个功能块仍然在执行时开始执行的情况。在这种情况下，第一个运动被测试信号在第一个功能块的恒定速度下中断并终止。第二个功能块在实际位置 3250 的基础上增加 4000 的距离，并将轴移动到最终位置 7250。TC2 任务组运动控制 2011 年 3 月 17 日 (C) 1999 - 2011 版权所有，PLCopen 运动控制功能块，第 37/141 页

MoveRelative - Example

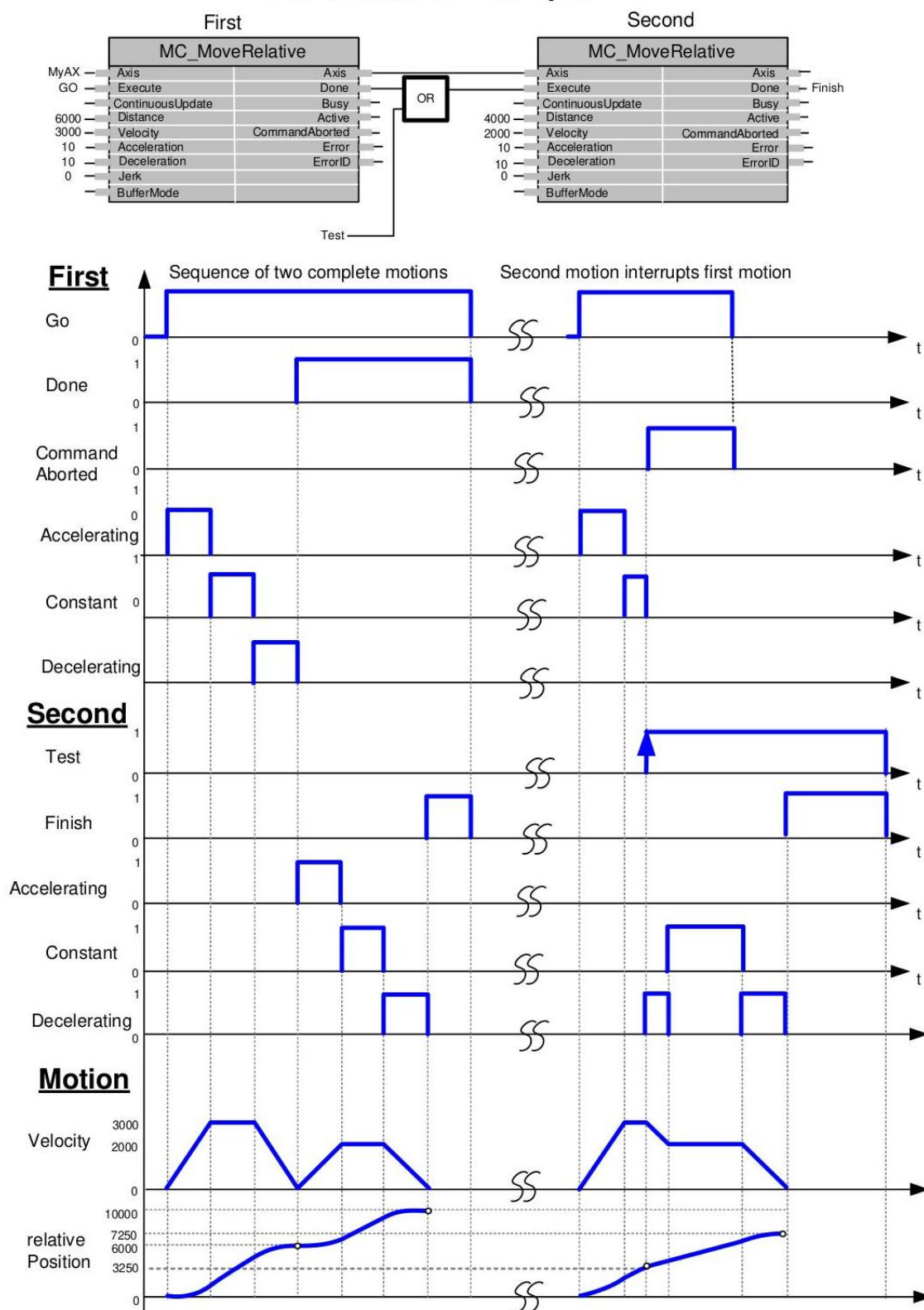
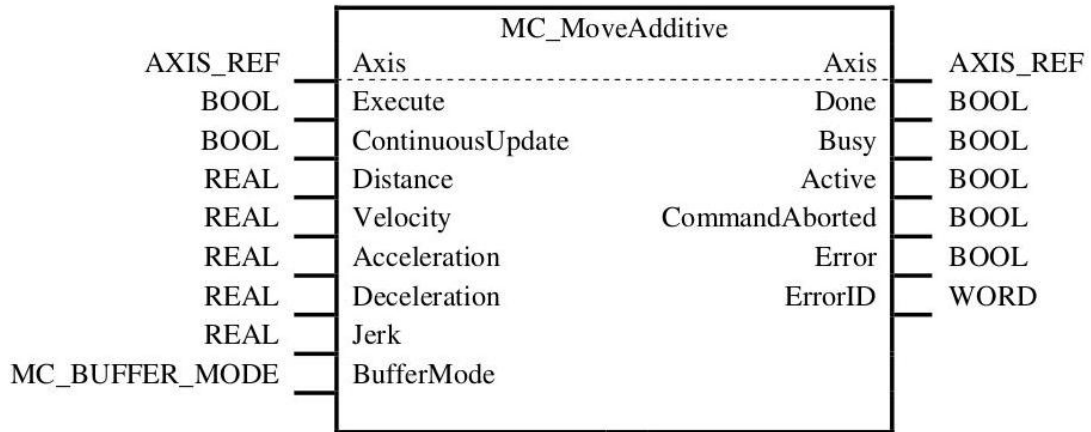


图 19: MC_MoveRelative 的时间图

3.7. MC_MoveAdditive

FBI Name		MC_MoveAdditive	
This Function Block commands a controlled motion of a specified relative distance additional to the most recent axis moved position in the axis state 'DiscreteMotion'. The most recent commanded position may be the result of a prior 'MC_MoveAdditive' motion which was aborted. If the FBI is activated in the axis state 'ContinuousMotion' the predefined relative distance is added to the set position at the time of the execution.			
VAR: IN, OUT			
Reference to the axis			
B	Axis	AXIS_REF	
VAR INPUT			
B	Execute	BOOL	Starts the motion at rising edge
E	ContinuousUpdate	BOOL	See 2.4.6 The input 'ContinuousUpdate'
B	Distance	REAL	Relative distance for the motion (in coordinate units [m])
B	Velocity	REAL	Value of the maximum velocity that is internally restricted [%v]
B	Acceleration	REAL	Value of the acceleration (increasing energy of the motor) [%v/s]
B	Deceleration	REAL	Value of the deceleration (decreasing energy of the motor) [%v/s]
B	Jerk	REAL	Value of the jerk [%v/s ²]
E	BufferMode	MC_BUFFER_MODE	Defines the decelerated sequence of the FBI. See 2.4.2 Aborting versus Bufferred motion
VAR OUTPUT			
B	Done	BOOL	Commanded distance reached
E	Busy	BOOL	The FBI is not finished and has output values set to be set-point
E	Active	BOOL	Indicates that the FBI has started due to the set-point
E	CommandAborted	BOOL	Command is aborted for another command
B	Error	BOOL	Signals that an error has occurred within the Function Block
E	ErrorID	WORD	Error identification
Notes: -			



下图展示了在“DiscreteMotion”状态下两个功能块组合的两个示例：

1. 时序图的左侧部分说明了当第二个功能块在第一个功能块之后被调用的情况。

如果第一个功能块达到指令距离 6000（且速度为 0），则输出“Done”会导致第二个功能块移动指令距离 4000，并将轴移动到结果位置 10000。

2. 时序图的右侧部分说明了当第二个移动功能块在第一个功能块仍在执行时开始执行的情况。在这种情况下，第一个运动在第一个功能块的恒定速度下被测试信号中断并终止。第二个功能块在之前命令的位置 6000 上加上距离 4000，并将轴移动到结果位置 10000。

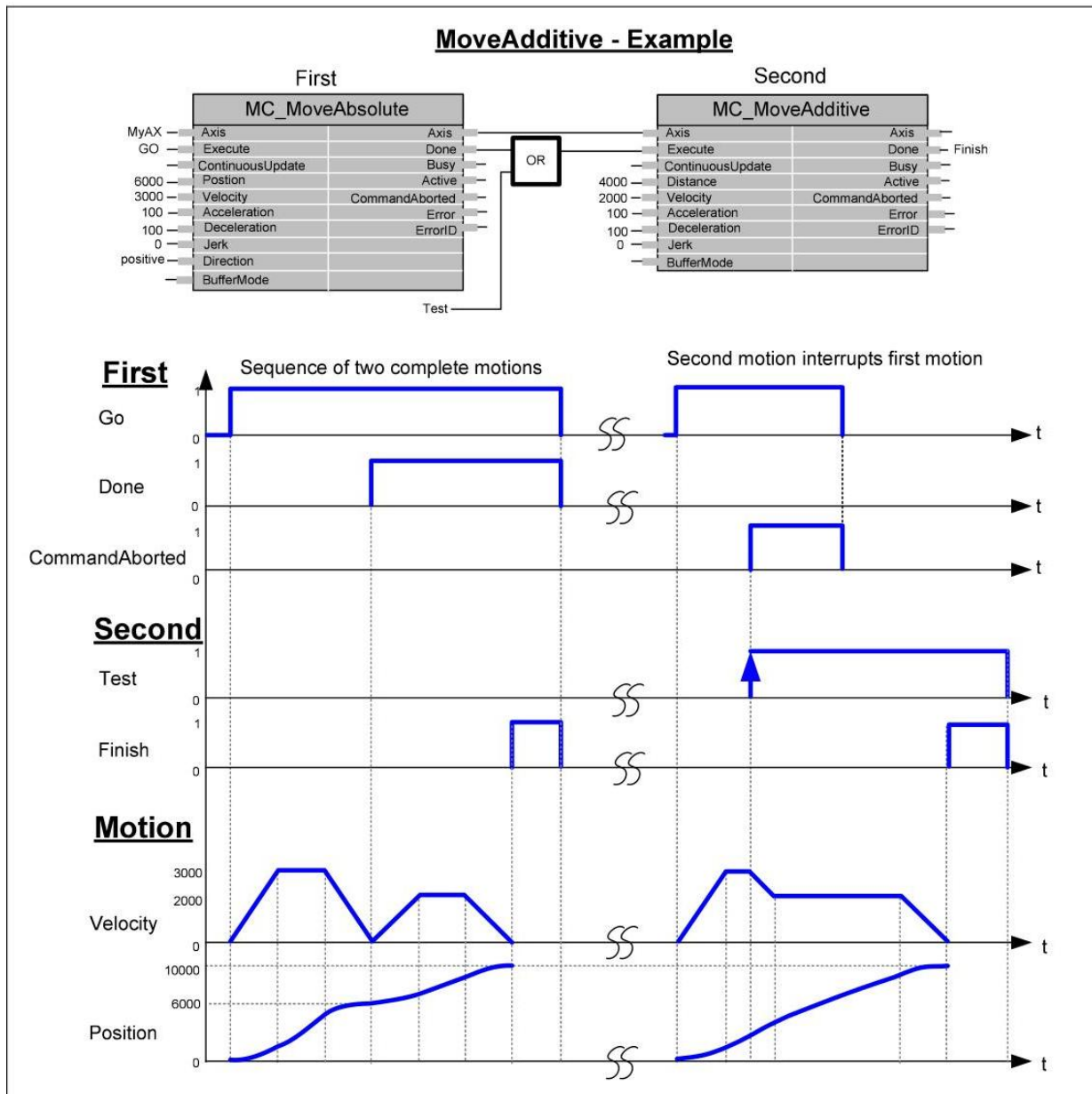


图 20: MC_MoveAdditive 的时序图

3.8. MC_MoveSuperimposed

FB-Name			MC_MoveSuperimposed	
This Function Block commands a controlled motion of a specified relative distance additional to an existing motion. The existing Motion is not interrupted, but is superimposed by the additional motion				
VAR_IN_OUT				
	B	Axis	AXIS_REF	Reference to the axis
VAR INPUT				
	B	Execute	BOOL	Start the motion at rising edge
	E	ContinuousUpdate	BOOL	See 2.4.6 The input 'ContinuousUpdate'
	B	Distance	REAL	Additional distance that is to be superimposed (in technical unit [u])
	E	VelocityDiff	REAL	Value of the velocity difference of the additional motion (not necessarily reached) [u/s]
	E	Acceleration	REAL	Value of the acceleration (increasing energy of the motor) [u/s ²]
	E	Deceleration	REAL	Value of the deceleration (decreasing energy of the motor) [u/s ²]
	E	Jerk	REAL	Value of the Jerk [u/s ³]
VAR OUTPUT				
	B	Done	BOOL	Additional distance superimposed to the ongoing motion
	E	Busy	BOOL	The FB is not finished and new output values are to be expected
	E	CommandAborted	BOOL	'Command' is aborted by another command
	B	Error	BOOL	Signals that an error has occurred within the Function Block
	E	ErrorID	WORD	Error identification
	E	CoveredDistance	REAL	Displays continuously the covered distance contributed by this FB since it was started

注意：

- 如果 MC_MoveSuperimposed 被激活，那么在主动模式下，除了 MC_MoveSuperimposed 之外的任何其他命令都会中止两个运动命令：即 MC_MoveSuperimposed 和基础运动命令。在任何其他模式下，基础运动命令不会被中止。
- 如果 MC_MoveSuperimposed 被激活并且另一个 MC_MoveSuperimposed 被命令，则仅中止正在进行的 MC_MoveSuperimposed 命令，并被新的 MC_MoveSuperimposed 替代，但基础运动命令不会被中止。
- FB MC_MoveSuperimposed 会改变正在进行的运动的速度，并在适用时改变指令位置，在所有相关状态下均有效。
- 在“静止”状态下，FB MC_MoveSuperimposed 的表现类似于 MC_MoveRelative。
- “加速度”、“减速度”和“突变”的值是对正在进行的运动的附加值，而不是绝对值。这样，无论是否同时发生 MC_MoveSuperimposed FB，基础 FB 始终会在相同的时间段内完成其工作。
- 输出“激活”的行为与缓冲 FB 的行为不同。

AXIS_REF	MC_MoveSuperimposed		
	Axis	Axis	AXIS_REF
BOOL	Execute	Done	BOOL
BOOL	ContinuousUpdate	Busy	BOOL
REAL	Distance	Active	BOOL
REAL	VelocityDiff	CommandAborted	BOOL
REAL	Acceleration	Error	BOOL
REAL	Deceleration	ErrorID	WORD
REAL	Jerk	CoveredDistance	REAL

在带有模态轴的卡口过程中 MC_MoveSuperimposed 的示例。绿色部分显示了从属位置同时显示了使用和不使用 MC_MoveSuperimposed 的情况：

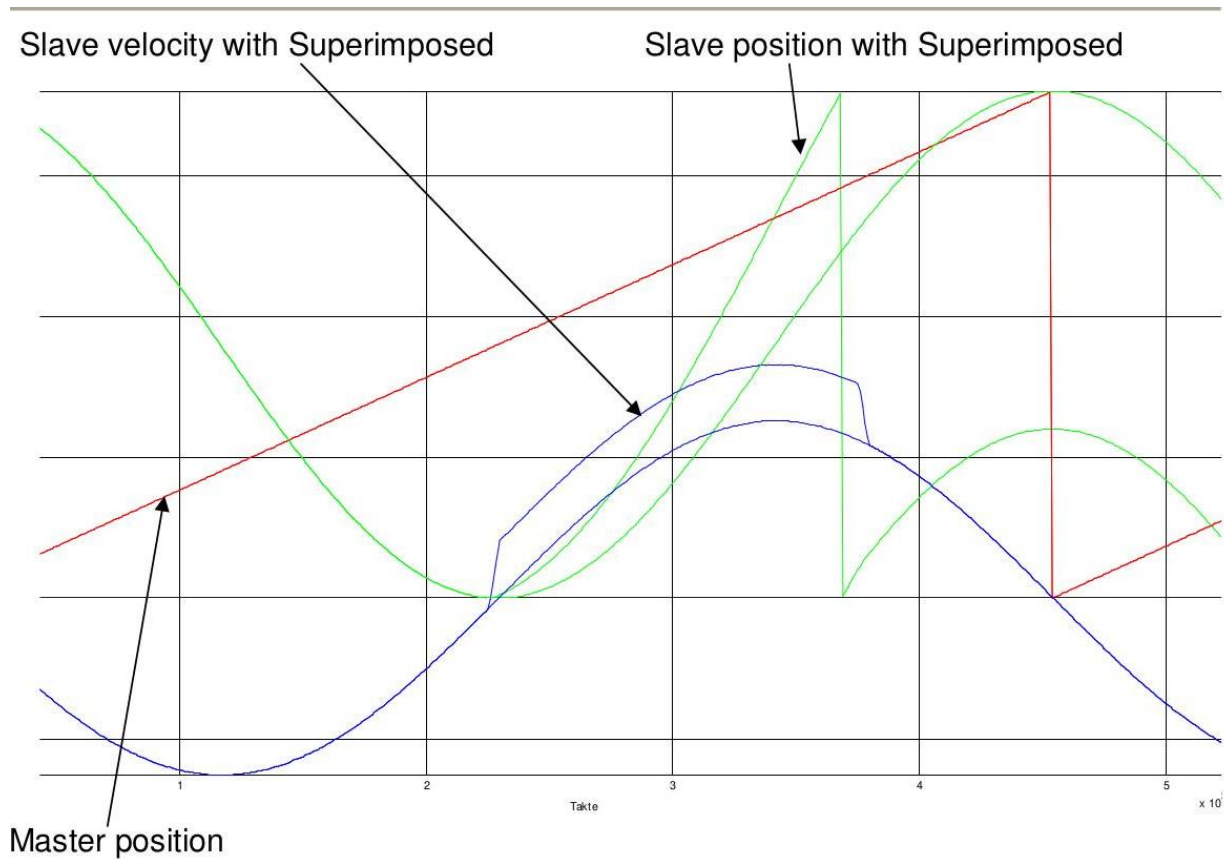


图 22: `MC_MoveSuperimposed` 对从属轴的影响示例

注意：在从属速度下，双线显示了在卡口过程中同步运动时 `MoveSuperimposed` 的效果。这适用于相关的从属位置。

下一个示例显示了 `MC_MoveSuperimposed` 对 `MC_MoveAbsolute` 的作用。即使基础的离散运动完成，`MC_MoveSuperimposed` 仍然会继续其运动。

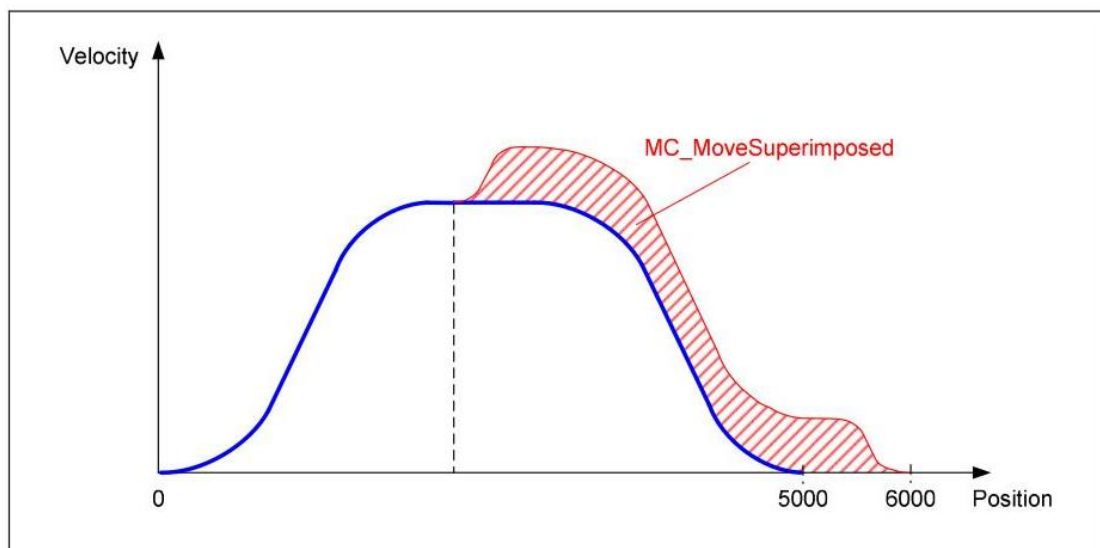
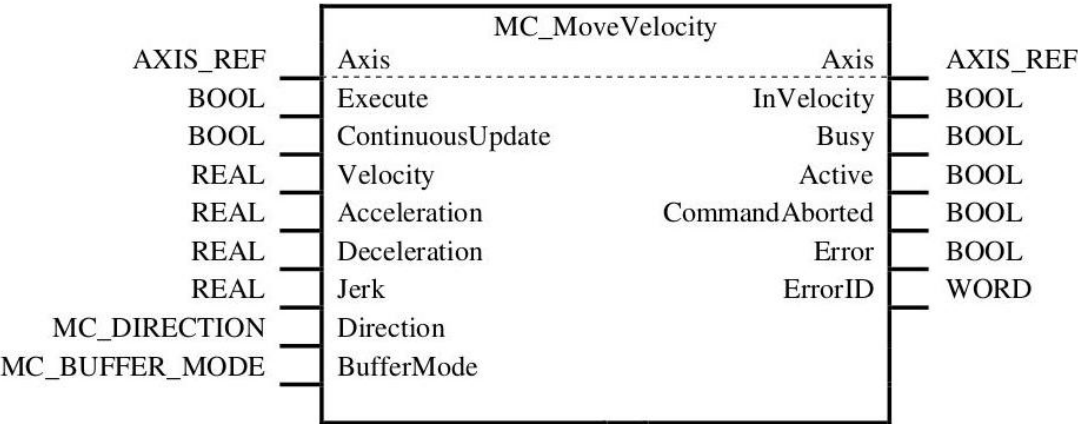


图 23: `MC_MoveSuperimposed` 对 `MC_MoveAbsolute` 的影响示例

3.10. MC_MoveVelocity

FB-Name		MC_MoveVelocity		
This Function Block commands a never ending controlled motion at a specified velocity				
VAR_IN_OUT				
	B	Axis	AXIS REF	Reference to the axis
VAR INPUT				
	B	Execute	BOOL	Start the motion at rising edge
	E	ContinuousUpdate	BOOL	See 2.4.6 The input 'ContinuousUpdate'
	B	Velocity	REAL	Value of the maximum velocity [u/s]. Can be a signed value
	E	Acceleration	REAL	Value of the acceleration (increasing energy of the motor) [u/s ²]
	E	Deceleration	REAL	Value of the deceleration (decreasing energy of the motor) [u/s ²]
	E	Jerk	REAL	Value of the Jerk [u/s ³]
	E	Direction	MC DIRECTION	Enum type (1-of-3 values: mcPositiveDirection, mcNegativeDirection, and mcCurrentDirection. Note: shortest way not applicable)
	E	BufferMode	MC BUFFER MODE	Defines the chronological sequence of the FB. See 2.4.2 Aborting versus Buffered modes
VAR OUTPUT				
	B	InVelocity	BOOL	Commanded velocity reached
	E	Busy	BOOL	The FB is not finished and new output values are to be expected
	E	Active	BOOL	Indicates that the FB has control on the axis
	E	CommandAborted	BOOL	'Command' is aborted by another command
	B	Error	BOOL	Signals that an error has occurred within the Function Block
	E	ErrorID	WORD	Error identification
Notes: - To stop the motion, the FB has to be interrupted by another FB issuing a new comm - The signal 'InVelocity' has to be reset when the block is aborted by another block.				

- 负速度 * 负方向 = 正速度
- 结合 MC_MoveSuperimposed, 输出'InVelocity' 被设置, 只要该功能块 (MC_MoveVelocity) 对设置速度的贡献等于该功能块的命令速度。



下图显示了两个 MC_MoveVelocity 功能块组合的例子：

1. 时序图的左侧部分说明了当第二个功能块在第一个功能块完成后被调用的情况。如果第一个功能块达到命令速度 3000，则输出'First.InVelocity' 和信号 Next 使第二个功能块移动到速度 2000。在下一个周期中，'First.InVelocity' 被重置，而'First.CommandAborted' 被设置。因此，第二个功能块的'Execute' 被重置。一旦轴达到'速度' 2000，'Second.InVelocity' 被设置。
2. 时序图的右侧部分说明了当第二个移动功能块在第一个功能块尚未'InVelocity' 时开始执行的情况。以下序列如下：通过输入'First.Execute' 的 GO 重新启动第一个运动。当第一个功能块仍在加速以达到速度 3000 时，第一个功能块会被打断并中止，因为测试信号启动了第二个功能块的运行。现在第二个功能块正在运行并将速度减速到 2000。

MoveVelocity - Example

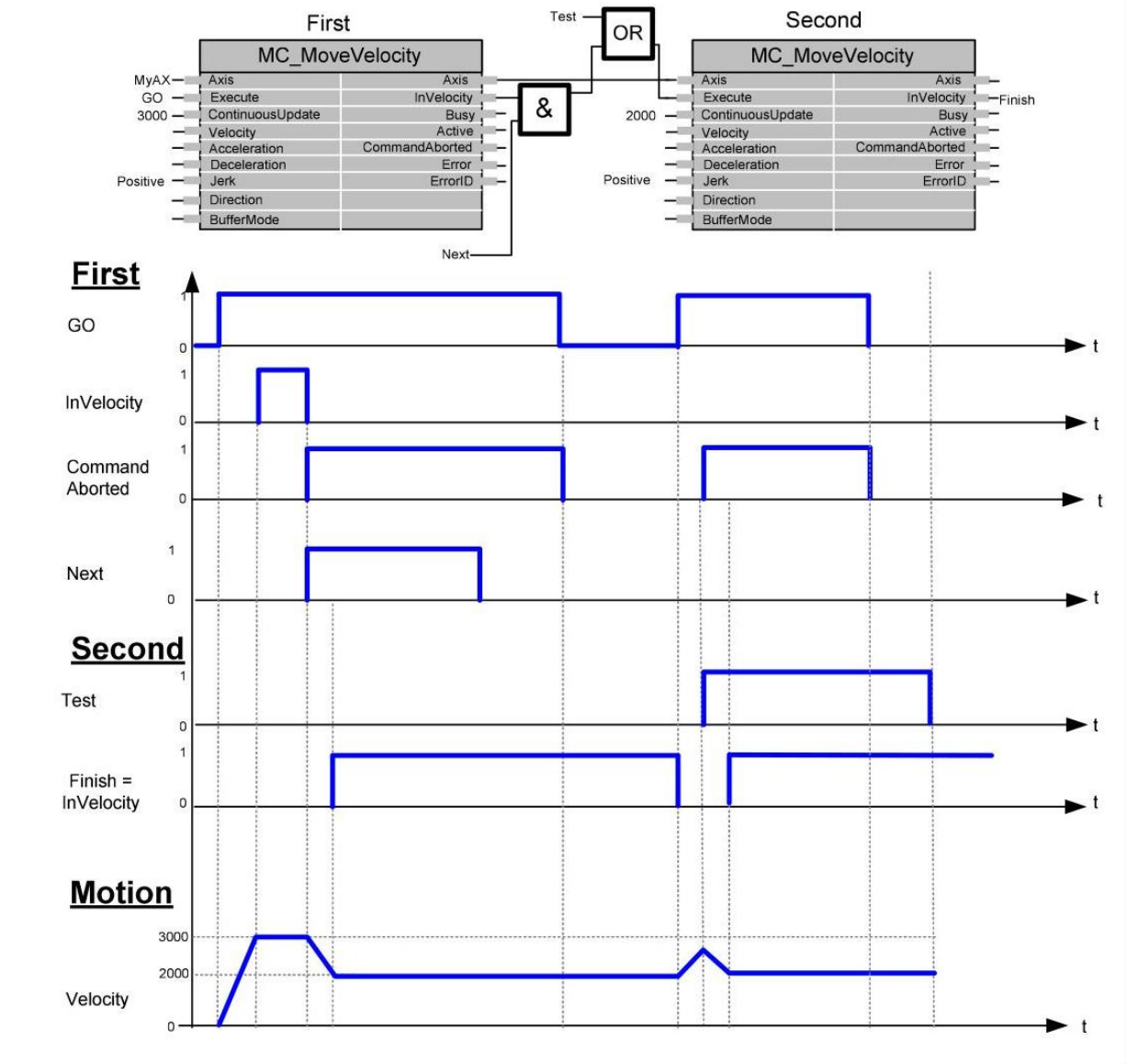


图 24: MC_MoveVelocity 时序图

注意: 2nd FB 处于'Aborting' 模式 (如果在缓冲模式下, 速度将达到 3000 才能作用于下一个功能块)。

3.11. MC_MoveContinuousAbsolute

FB-Name		MC_MoveContinuousAbsolute	
This Function Block commands a controlled motion to a specified absolute position ending with the specified velocity			
VAR_IN_OUT			
B	Axis	AXIS_REF	Reference to the axis
VAR INPUT			
B	Execute	BOOL	Start the motion at rising edge
E	ContinuousUpdate	BOOL	See 2.4.6 The input 'ContinuousUpdate'
B	Position	REAL	Commanded position for the motion (in technical unit [u]) (negative or positive)
E	EndVelocity	REAL	Value of the end velocity [u/s], Signed value
B	Velocity	REAL	Value of the maximum velocity [u/s]
E	Acceleration	REAL	Value of the acceleration [u/s ²]
E	Deceleration	REAL	Value of the deceleration [u/s ²]
E	Jerk	REAL	Value of the Jerk [u/s ³]
E	Direction	MC_DIRECTION	Enum type (1-of-4 values: noPositionDirection, noNegativeDirection, noCurrentDirection and noShort: noWay)
E	BufferMode	MC_BUFFERMODE	Define the chronological sequence of the FB. See 2.4.2 Aborting versus Buffered mode
VAR OUTPUT			
B	InEndVelocity	BOOL	Commanded distance reached and running at requested end velocity
E	Busy	BOOL	The FB is not finished and new output values are to be expected
E	Active	BOOL	Indicates that the FB has control on the axis
E	CommandAborted	BOOL	'Command' is aborted by another command
B	Error	BOOL	Signals that an error has occurred within the Function Block
E	ErrorID	WORD	Error identification
Notes: 1. If the commanded position is 0 and if no new motion command is put into the buffer, the axis continues to run with the specified 'EndVelocity'. 2. State 'ContinuousMotion' (member a of the enumeration 'MC_MoveAbsolute') is only set if 'ContinuousUpdate' is set to TRUE. This FB can be reduced by the combination of MC_MoveAbsolute and MC_MoveVelocity if BufferMode is less, abbreviated on this block.			

Notes: - If the commanded position is reached and no new motion command is put into the buffer, the axis continues to run with the specified 'EndVelocity' - State 'ContinuousMotion' (meaning: it will not stop by itself) - This FB can be replaced by the combination of MC_MoveAbsolute and MC_MoveVelocity if BufferMode is implemented on those FBs.

AXIS_REF		MC_MoveContinuousAbsolute		AXIS_REF
		AxisAxis		
BOOL		Execute	InEndVelocity	BOOL
BOOL		ContinuousUpdate	Busy	BOOL
REAL		Position	Active	BOOL
REAL		EndVelocity	CommandAborted	BOOL
REAL		Velocity	Error	BOOL
REAL		Acceleration	ErrorID	WORD
REAL		Deceleration		
REAL		Jerk		
MC_BUFFER_MODE		BufferMode		

MC_MoveContinuousAbsolute 的一个用例是线性切割机：

一个线性轴承载载着用于切割工件的激光装置。

从 lIdlePos 开始，工作链如下：



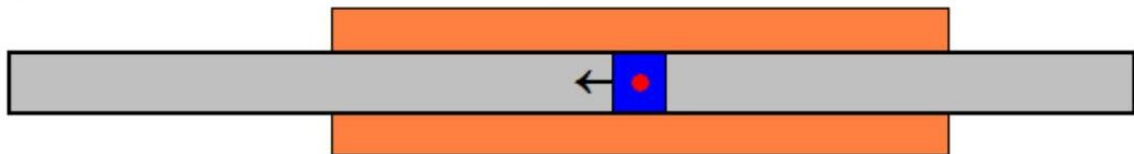
1. Move the laser with fast velocity over the position lStartCutPos. The laser is off during this movement:



2. Turn back and make sure to have the speed lCutVelocity when at lStartCutPos. At this position, switch the laser on:



3. Travel over the work piece with this constant speed while the laser is on:



4. When reaching lEndCutPos switch off the laser and move back to idle position with fast velocity:



在切割过程中，激光必须以固定速度移动，不能容忍加速或减速阶段。切割完成后，激光必须移动回到其等待位置。

这可以通过 FB MC_MoveContinuousAbsolute 以以下方式实现：

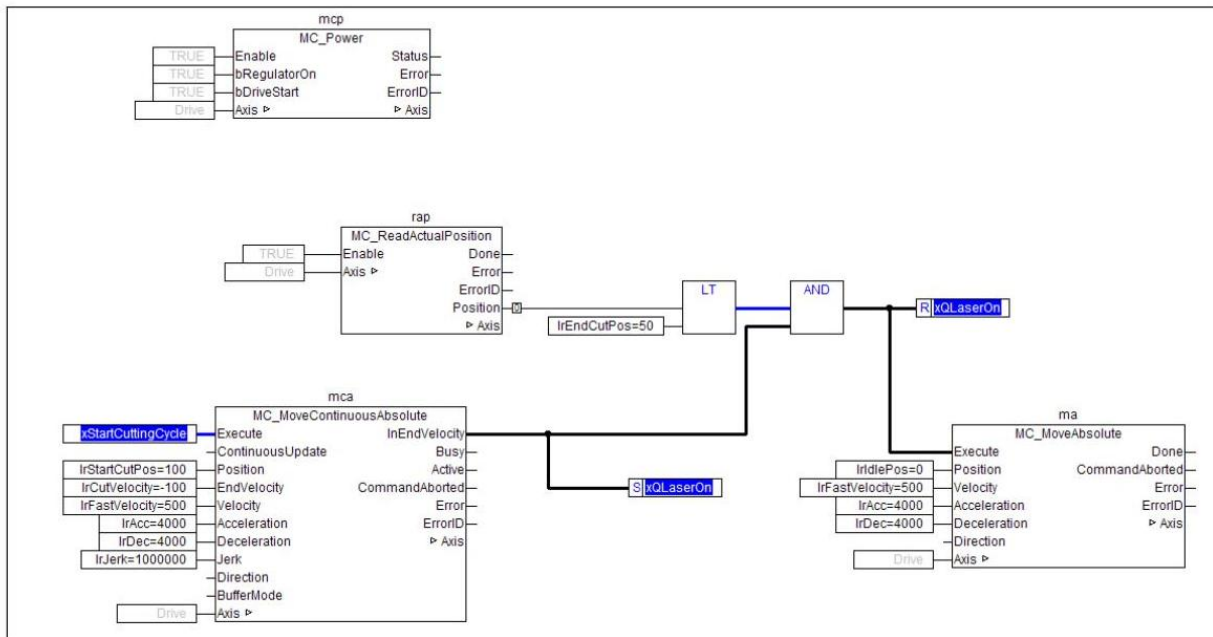


图 25: MC_MoveContinuousAbsolute 示例

在 xStartCuttingCycle 的上升沿开始，MC_MoveContinuousAbsolute 的实例'mca' 将以 IrFastVelocity 使轴移动到 IrStartCutPos，随后反向移动，并在再次到达 IrStartCutPos 时以 IrCutVelocity 的速度向负方向运动。在这一时刻，设置'InEndVelocity'，并且激光开启。由于没有其他运动 FB 中断此运动，MC_MoveContinuousAbsolute 将以当前速度继续向负方向移动。当轴超越位置 IrEndPos 时，激光关闭，MC_MoveAbsolute 的实例'ma' 将轴快速移动到其待机位置：

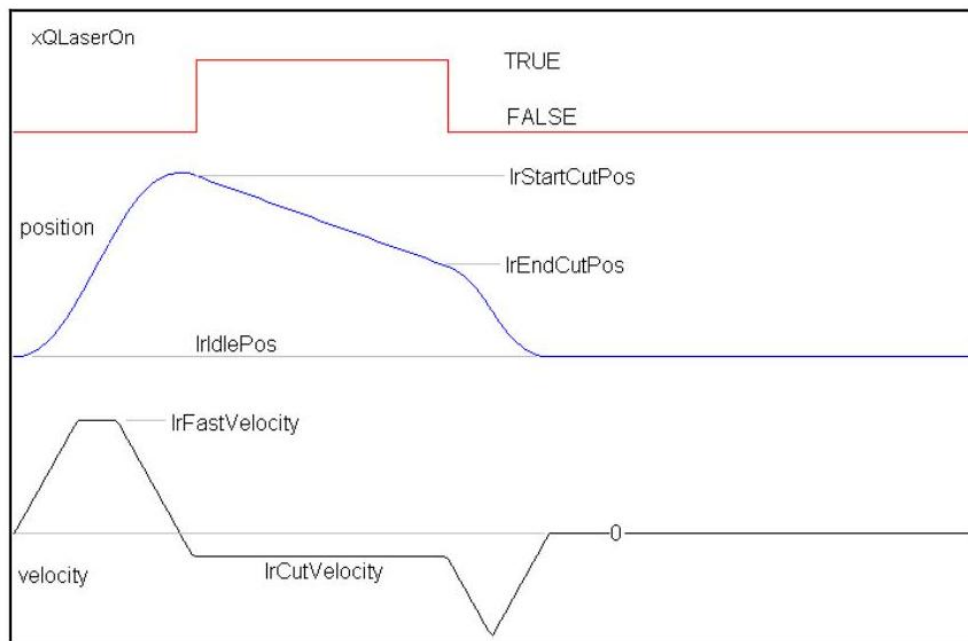


图 26: 上述示例的 MC_MoveContinuousAbsolute 时序图

3.12. MC_MoveContinuousRelative

FB-Name		MC_MoveContinuousRelative	
This Function Block commands a controlled motion of a specified relative distance ending with the specified velocity			
VAR_IN_OUT			
B	Axis	AXIS REF	Reference to the axis
VAR_INPUT			
B	Execute	BOOL	Start the motion at rising edge
E	ContinuousUpdate	BOOL	See 2.4.6 The input 'ContinuousUpdate'
B	Distance	REAL	Relative distance for the motion [u]
B	EndVelocity	REAL	Value of the end velocity [u/s]. Signed value
B	Velocity	REAL	Value of the maximum velocity [u/s]
E	Acceleration	REAL	Value of the acceleration [u/s ²]
E	Deceleration	REAL	Value of the deceleration [u/s ²]
E	Jerk	REAL	Value of the Jerk [u/s ³]
E	BufferMode	MC BUFFER MODE	Defines the chronological sequence of the FB. See 2.4. Aborting versus Buffered modes
VAR_OUTPUT			
B	InEndVelocity	BOOL	Commanded distance reached and running at requested end velocity
E	Busy	BOOL	The FB is not finished and new output values are to be expected
E	Active	BOOL	Indicates that the FB has control on the axis
E	CommandAborted	BOOL	'Command' is aborted by another command
B	Error	BOOL	Signals that an error has occurred within the Function Block
B	ErrorID	WORD	Error identification

备注：

- 如果到达了指令的位置且缓冲区中没有新的运动指令，轴将继续以指定的'EndVelocity' 运行。
- 状态'ContinuousMotion'（意思是：它不会自行停止）。
- 这个功能块在这里是针对不支持'BufferMode' 的系统的。
- 如果这些功能块实现了缓冲模式，则此功能块可以用 MC_MoveAbsolute 和 MC_MoveVelocity 的组合来替代。

MC_MoveContinuousRelative			
AXIS_REF	AxisAxis		AXIS_REF
BOOL	Execute	InEndVelocity	BOOL
BOOL	ContinuousUpdate	Busy	BOOL
REAL	Distance	Active	BOOL
REAL	EndVelocity	CommandAborted	BOOL
REAL	Velocity	Error	BOOL
REAL	Acceleration	ErrorID	WORD
REAL	Deceleration		
REAL	Jerk		
MC_BUFFER_MODE	BufferMode		

这两条采样轨迹显示了输入'EndVelocity' 值的正负符号的影响：1. 方向为正的'EndVelocity'：

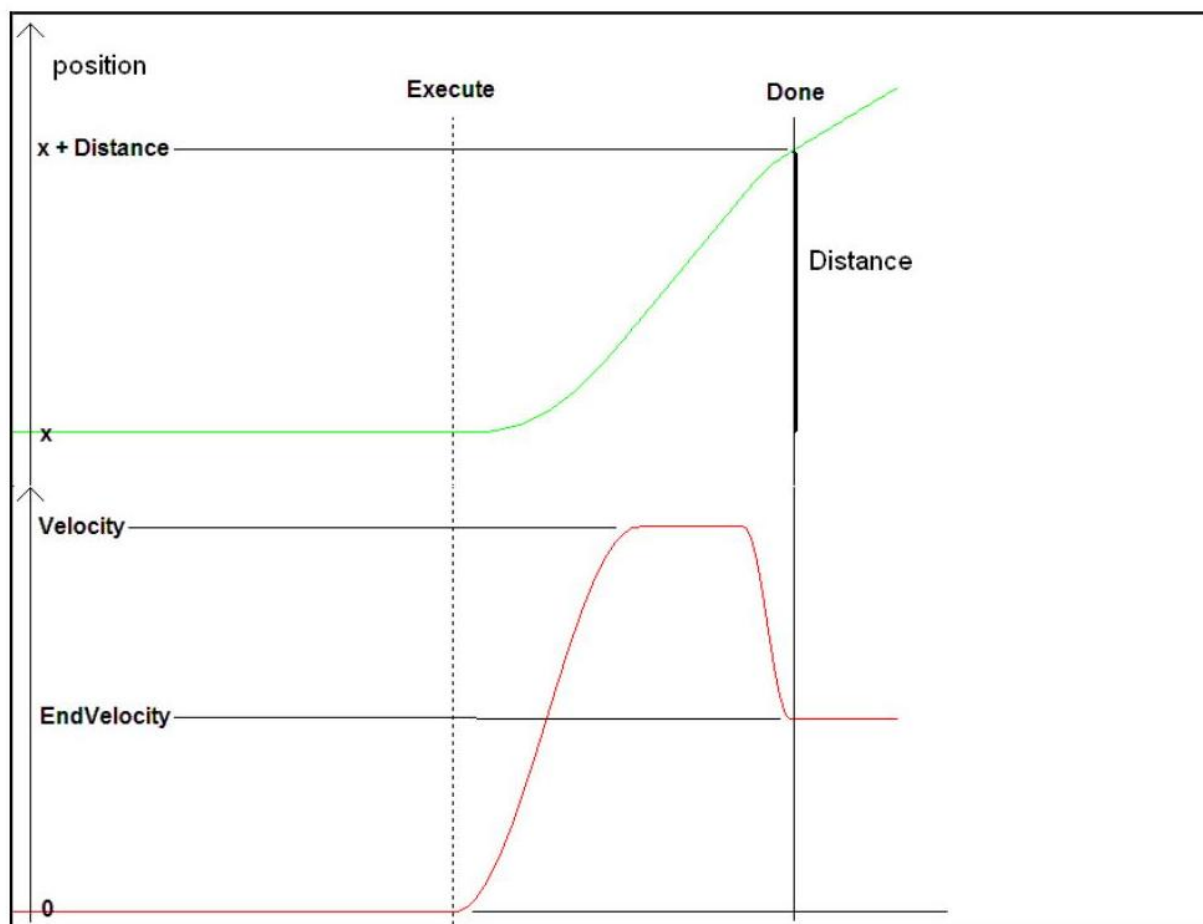


图 27: 方向为正的 MC_MoveContinuousRelative 定时图

'velocity' with negative direction:

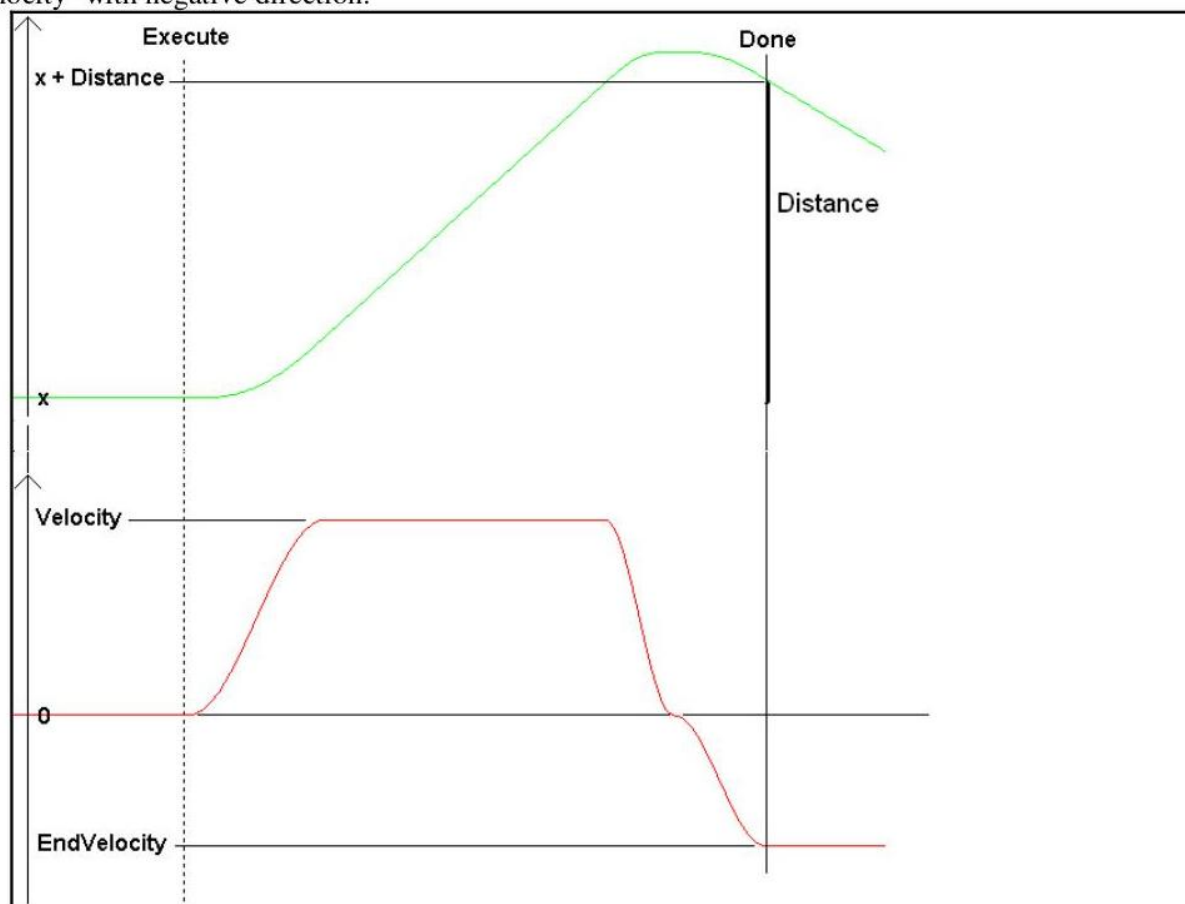


图 28: 方向为负的 `MC_MoveContinuousRelative` 定时图

`MC_MoveContinuousRelative` 的示例:

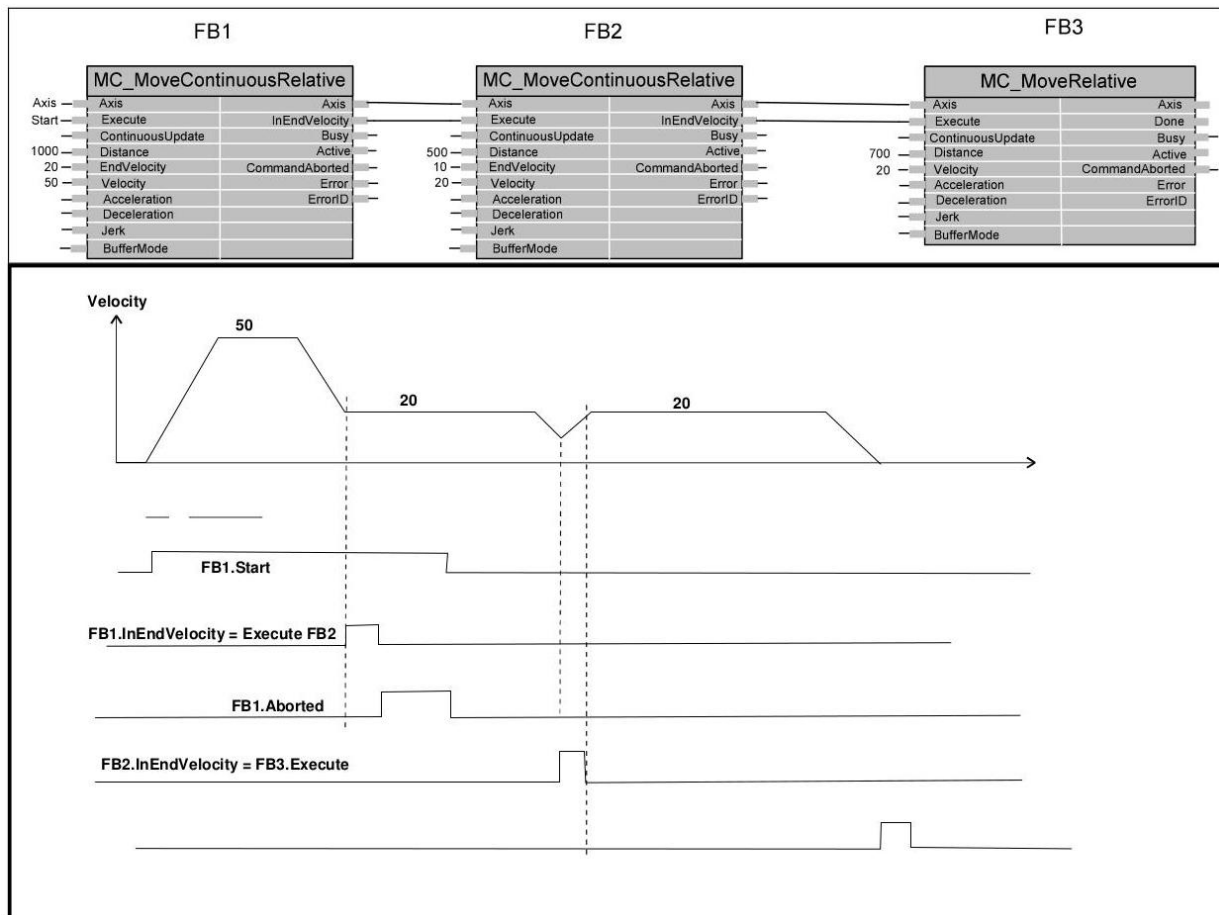
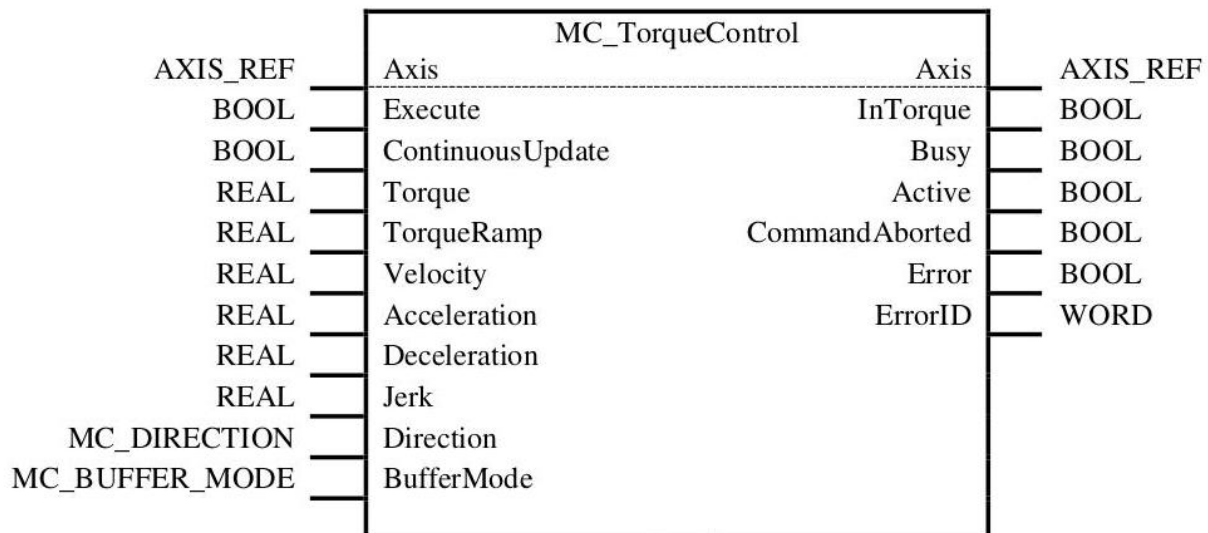


图 29: MC_MoveContinuousRelative 的示例

3.13. MC_TorqueControl

FB Name		MC_TorqueControl
This Function Block controls torque in a loop or in a batch mode. The output is a torque value (Torque) and the Function Block sets the "Torque" output of the user specified target to "Torque". This Function Block is applicable for both real and word. When there is no torque set, there is no applicable "Torque" output to be the position direction of velocity.		
VAR: IN: OUT		
1	Axis	AXIS_REF
2	Execute	BOOL
3	ContinuousUpdate	BOOL
4	Torque	REAL
5	TorqueRamp	REAL
6	Velocity	REAL
7	Acceleration	REAL
8	Deceleration	REAL
9	Jerk	REAL
10	Direction	MC_DIRECTION
11	BufferMode	MC_BUFFER_MODE
12	InTorque	BOOL
13	Busy	BOOL
14	Active	BOOL
15	CommandAborted	BOOL
16	Error	BOOL
17	ErrorID	WORD
18	BufferMode	MC_BUFFER_MODE



下面的例子展示了一个中间“电阻”负载的典型行为（见“减速”极限），以及一些“惯性”（见“扭矩斜坡”极限）。

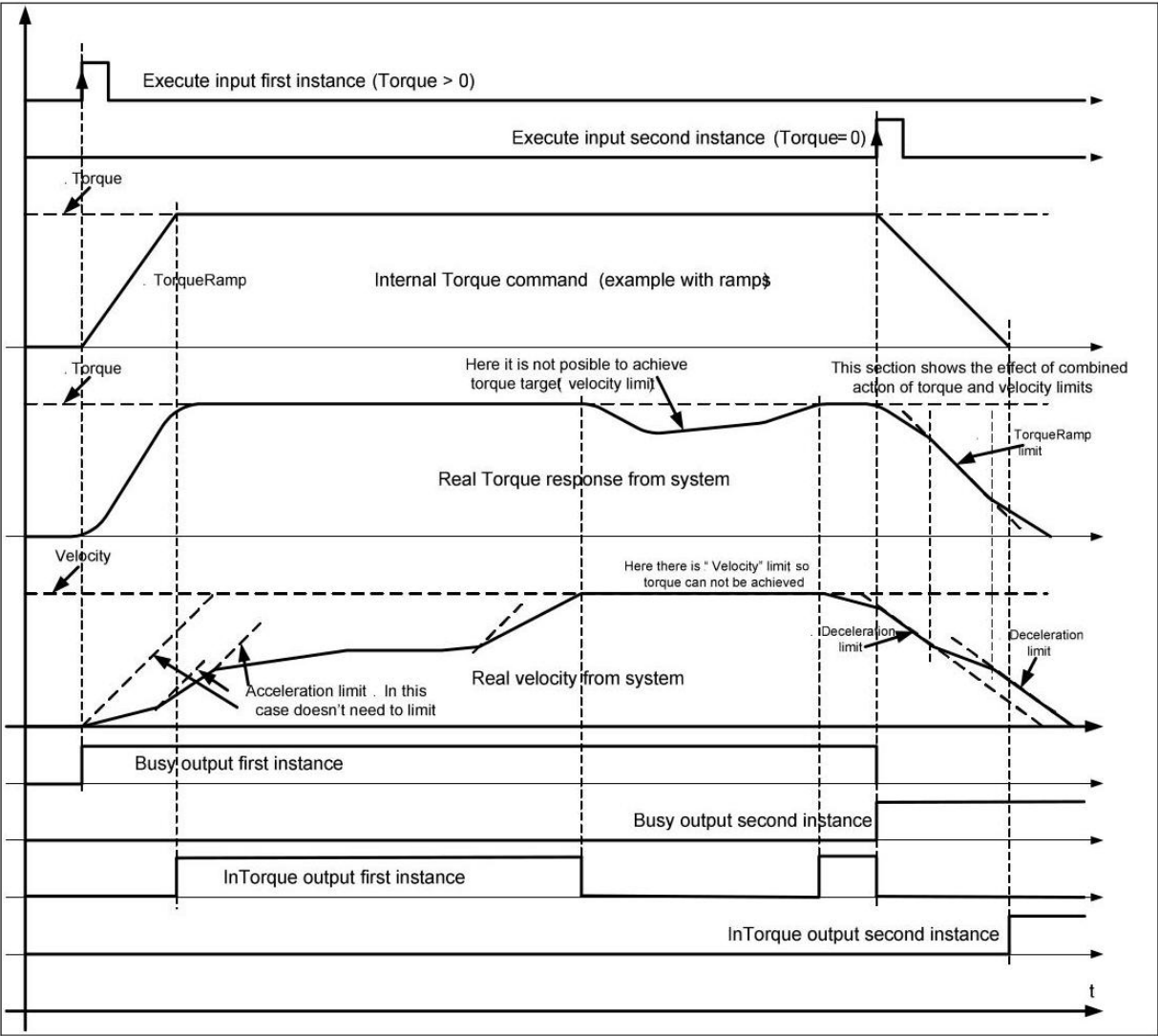


图 30: MC_TorqueControl 的第一个例子
这个例子可以在函数块图中实现，如下所示：

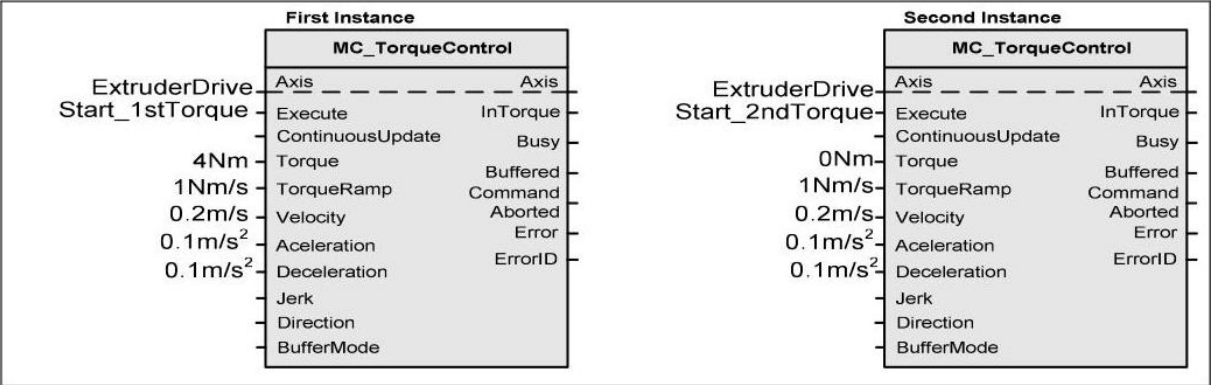


图 31: MC_TorqueControl 示例的程序

第二个例子（如下所示）使用了相反符号的“方向”和“扭矩”（例如保持或刹车控制）。在 FB 中：+ 方向 - 扭矩。这就像一个在材料上施加扭矩的展开应用，以及材料中的制动。当材料断裂时，如图中央所示，会导致实际扭矩值（绝对值）的下降：速度将下降，受限与“减速”VAR_INPUT 指定的最快“减速”极限，直到零速度（在没有张力的情况下有发生冲击断裂的风险，因此我们必须限制到最快）。在这种情况下，扭矩设定值可能无法实现。

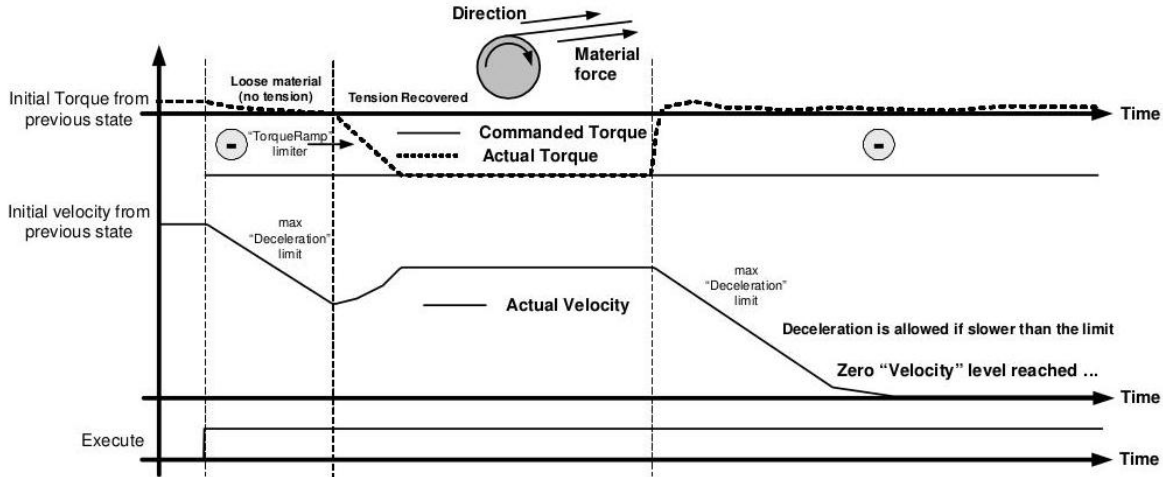


图 32: MC_TorqueControl 的第二个示例

注意：在一个放卷应用中（源自此刹车控制），目标是材料张力，而不是电机转矩。应考虑卷轴的瞬时直径，以转换“用户张力设定值”。此外，通过修改加速/减速的转矩设定值来进行额外的惯性补偿是常见的，这通常是根据瞬时重量数据（重量通常是根据直径估算的）进行的。此外，在放卷应用中，在松散材料的情况下（与材料断裂的情况相同），通常会施加一个负的慢速度参考，以“重绕”松散材料。在这种情况下，这必须由外部编程提供。

3.14. MC_PositionProfile

FB-Name		MC_PositionProfile		
This Function Block commands a time-position locked motion profile				
VAR_IN_OUT				
	B	Axis	AXIS_REF	Reference to the axis
	B	TimePosition	MC_TP_REF	Reference to Time / Position. Description - see note below
VAR_INPUT				
	B	Execute	BOOL	Start the motion at rising edge
	E	ContinuousUpdate	BOOL	See 2.4.6 The input 'ContinuousUpdate'
	E	TimeScale	REAL	Overall time scaling factor of the profile
	E	PositionScale	REAL	Overall Position scaling factor
	E	Offset	REAL	Overall offset for profile [u]
	E	BufferMode	MC BUFFER MODE	Defines the chronological sequence of the FB. See 2.4.2 Aborting versus Buffered modes
VAR OUTPUT				
	B	Done	BOOL	Profile completed
	E	Busy	BOOL	The FB is not finished and new output values are to be expected
	E	Active	BOOL	Indicates that the FB has control on the axis
	E	CommandAborted	BOOL	'Command' is aborted by another command
	B	Error	BOOL	Signals that an error has occurred within the Function block
	E	ErrorID	WORD	Error identification

注释：

- MC_TP_REF 是一个特定于供应商的数据类型。下面提供了该数据类型的一个示例：

- 时间/位置对的内容可以用 DeltaTime/Pos 表示，其中 Delta 可以是两个连续点之间的时间差。

- 类型

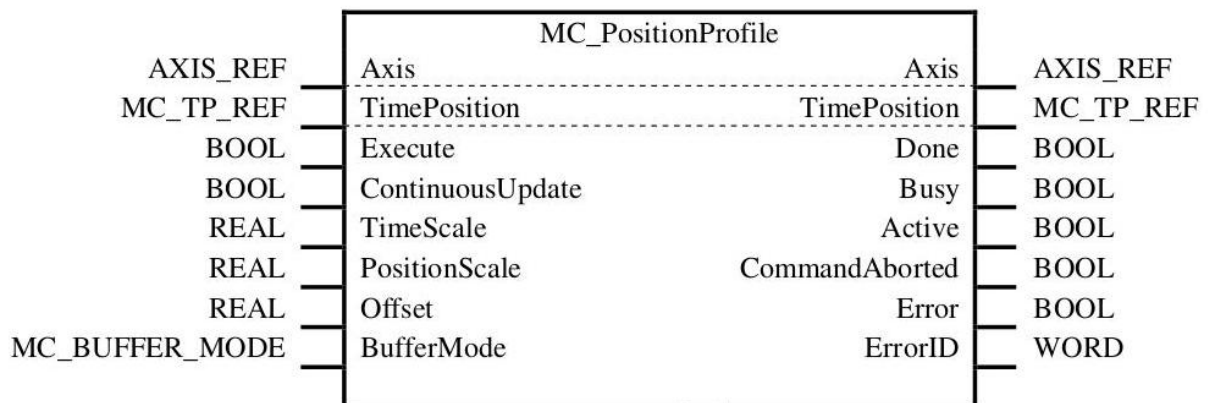
```
MC_TP : STRUCT
DeltaTime : TIME;
Position : REAL;
END_STRUCT;
END_TYPE
```

- 类型

```
MC_TP_REF : STRUCT
NumberOfPairs : WORD;
IsAbsolute : BOOL;
MC_TP_Array : ARRAY [1..N] OF MC_TP;
END_STRUCT;
END_TYPE
```

- 此功能并不意味着它一次又一次地运行一个配置文件：它可以在不同的配置文件之间切换。

- 或者，FB MC_CamIn 可以与虚拟主控结合使用。



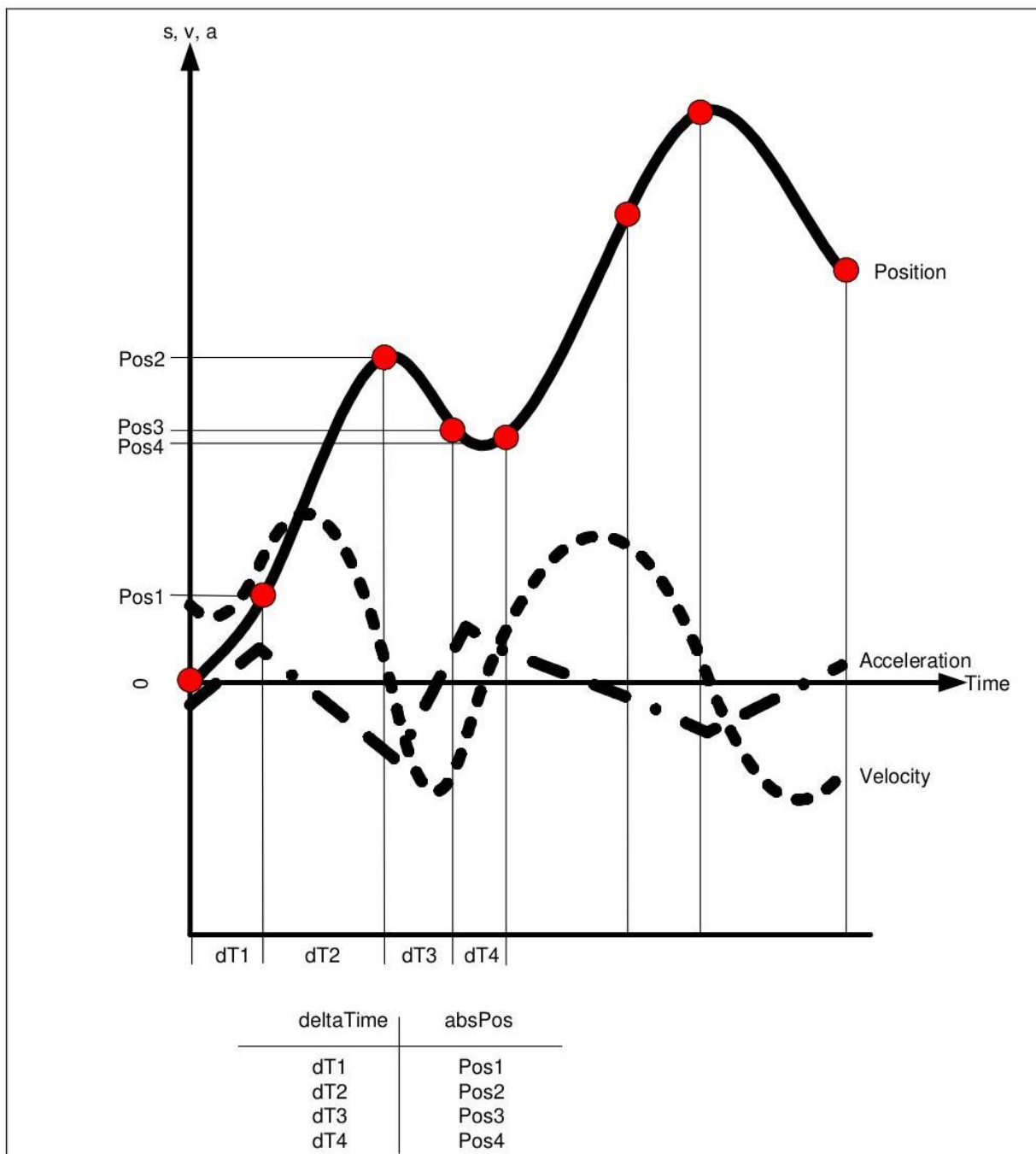


图 33: 时间/MC_PositionProfile 的示例

注：时间/速度和时间/加速度配置文件类似于“位置”配置文件，具有在“速度”或“加速度”线上采样点。

3.15. MC_VelocityProfile

FB-Name		MC_VelocityProfile		
his Function Block commands a time-velocity locked motion profile. The velocity in the final element in the profile should be maintained. The state remains 'ContinuousMotion'				
VAR_IN_OUT				
B	Axis	AXIS_REF	Reference to the axis	
B	TimeVelocity	MC_TV_REF	Reference to Time / Velocity. Description - see note below	
VAR_INPUT				
B	Execute	BOOL	Start the motion at rising edge	
E	ContinuousUpdate	BOOL	See 2.4.6 The input 'ContinuousUpdate'	
E	TimeScale	REAL	Overall time scaling factor of the profile	
E	VelocityScale	REAL	Overall velocity scaling factor of the profile	
E	Offset	REAL	Overall offset for profile [u/s]	
E	BufferMode	MC_BUFFER_MODE	Defines the chronological sequence of the FB. See 2.4.2 Aborting versus Buffered modes	
VAR_OUTPUT				
B	ProfileCompleted	BOOL	End of profile reached	
E	Busy	BOOL	The FB is not finished and new output values are to be expected	
E	Active	BOOL	Indicates that the FB has control on the axis	
E	CommandAborted	BOOL	'Command' is aborted by another command	
B	Error	BOOL	Signals that an error has occurred within the Function Block	
E	ErrorID	WORD	Error identification	

备注：

- MC_TV_REF 是一个特定于供应商的数据类型。下面给出了这个数据类型的一个示例：
- 时间/速度对的内容可以用 DeltaTime/Velocity 表示，其中 Delta 可以是两个连续点之间的时间差。速度可以是有符号值。

- 类型

```
MC_TV : STRUCT
DeltaTime : TIME;
Velocity : REAL;
END_STRUCT;
END_TYPE
```

- 类型

```
MC_TV_REF : STRUCT
NumberOfPairs : WORD;
MC_TV_Array : ARRAY [1..N] of MC_TV;
END_STRUCT;
END_TYPE
```

- 该功能并不意味着它会反复运行一个配置文件：它可以在不同的配置文件之间切换 - 作为此 FB 的替代，可以使用连接到虚拟主控的 CAM FB

3.16. MC_AccelerationProfile

运动控制的功能块

FB-Name		MC_AccelerationProfile	
This Function Block commands a time-acceleration locked motion profile. After finalizing the acceleration profile, the acceleration goes to 0 (and typically the final velocity is maintained). It stays in the state 'ContinuousMotion'.			
VAR_IN_OUT			
B	Axis	AXIS_REF	Reference to the axis
B	TimeAcceleration	MC_TA_REF	Reference to Time / Acceleration. Description - see note below
VAR INPUT			
B	Execute	BOOL	Start the motion at rising edge
E	ContinuousUpdate	BOOL	See 2.4.6 The input 'ContinuousUpdate'
E	TimeScale	REAL	Overall time scaling factor of the profile
E	AccelerationScale	REAL	Scale factor for acceleration amplitude
E	Offset	REAL	Overall offset for profile [u/s ²]
E	BufferMode	MC_BUFFER_MODE	Defines the chronological sequence of the FB. See 2.4.2 Aborting versus Buffered modes
VAR OUTPUT			
B	ProfileCompleted	BOOL	End of profile reached
E	Busy	BOOL	The FB is not finished and new output values are to be expected
E	Active	BOOL	Indicates that the FB has control on the axis
E	CommandAborted	BOOL	'Command' is aborted by another command
B	Error	BOOL	Signals that an error has occurred within the Function Block
E	ErrorID	WORD	Error identification

备注：

- MC_TA_REF 是一种特定于供应商的数据类型。以下是此数据类型的示例：
- 时间/加速度对的内容可以用 DeltaTime/加速度来表示，其中 Delta 可能是两个连续点之间的时间差。

- 类型

```
MC_TA : STRUCT
DeltaTime : TIME;
Acceleration : REAL;
END_STRUCT;
END_TYPE
```

- 类型

```
MC_TA_REF : STRUCT
NumberOfPairs : WORD;
MC_TA_Array : ARRAY [1..N] of MC_TA;
END_STRUCT;
END_TYPE
```

除此 FB 外，还可以使用与虚拟主控结合的 CAM FB。

	MC_AccelerationProfile		
AXIS_REF	AxisAxis		AXIS_REF
MC_TA_REF	TimeAcceleration	TimeAcceleration	MC_TA_REF
BOOL	Execute	ProfileCompleted	BOOL
BOOL	ContinuousUpdate	Busy	BOOL
REAL	TimeScale	Active	BOOL
REAL	AccelerationScale	CommandAborted	BOOL
REAL	Offset	Error	BOOL
MC_BUFFER_MODE	BufferMode	ErrorID	WORD

Example of an acceleration profile:

一个配置是由多个连续的“A 到 B”定位点构成的。虽然可视化简单，但为了获得平滑的轮廓却需要很多序列。这些需求往往超出了低端伺服电机的能力。

另一方面，通过使用适量的恒定加速段，可以定义出一个良好匹配的运动轮廓。采用这种方法，可以扩展低端伺服电机的能力范围。

可以进行匹配的对象包括：

1. 位置与时间轮廓
2. 主轴与从轴

优点：

- 紧凑的轮廓描述
- 自然的平滑轮廓特性

- 使用现有工具时编程抽象级别较高

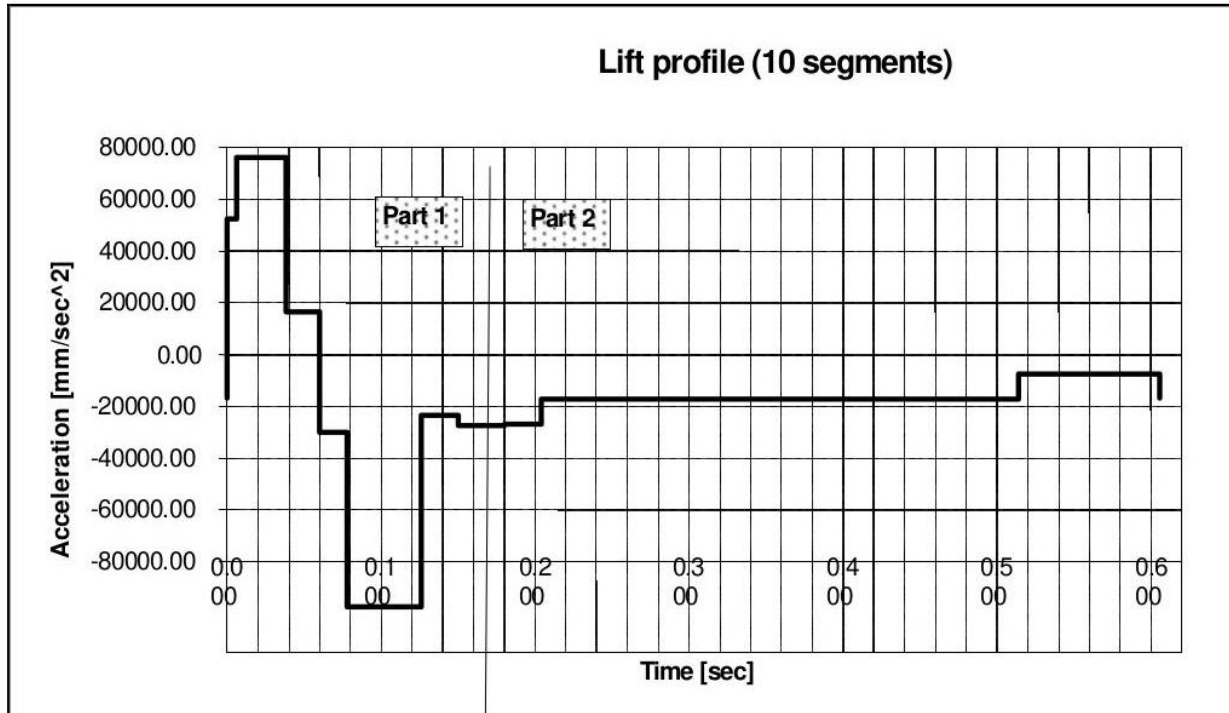


图 34: MC_AccelerationProfile, 仅 10 个段

3.17. MC_SetPosition

3.18. MC_SetOverride

[illegible]

	MC_SetOverride		
AXIS_REF	Axis	Axis	AXIS_REF
BOOL	Enable	Enabled	BOOL
REAL	VelFactor	Busy	BOOL
REAL	AccFactor	Error	BOOL
REAL	JerkFactor	ErrorID	WORD

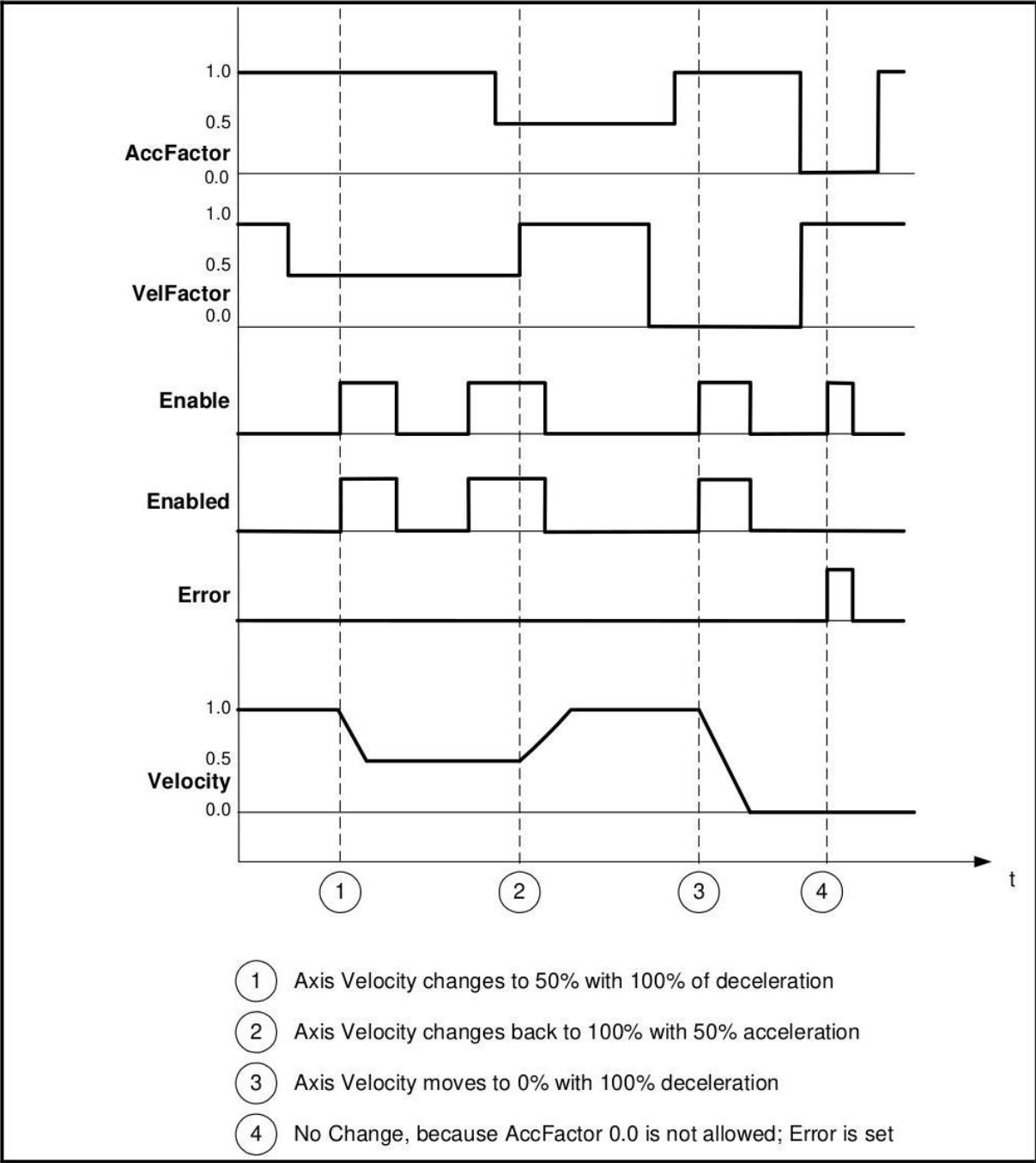
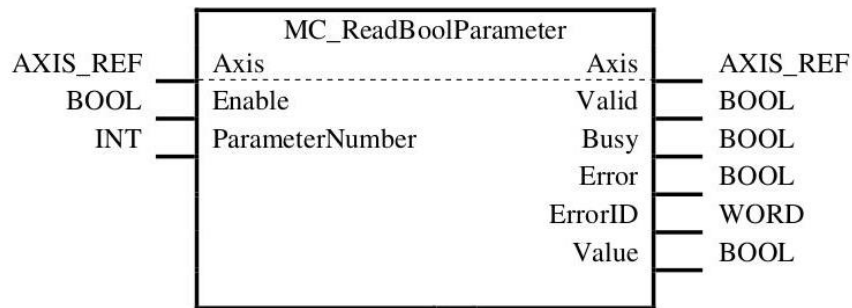


图 36: MC_SetOverride 的图形解释

3.19. MC_ReadParameter & MC_ReadBoolParameter

FB-Name			MC_ReadParameter		
his Function Block returns the value of a vendor specific parameter. The returned Value has to be converted to Rea if necessary. If not possible, the vendor has to supply a vendor specific FB to read the parameter.					
VAR_IN_OUT					
B	Axis	AXIS_REF	Reference to the axis		
VAR_INPUT					
B	Enable	BOOL	Get the value of the parameter continuously while enabled		
B	ParameterNumber	INT	Number of the parameter. One can also use symbolic paramete names which are declared as VAR CONST		
VAR_OUTPUT					
B	Valid	BOOL	A valid output is available at the FB		
E	Busy	BOOL	The FB is not finished and new output values are to be expected		
B	Error	BOOL	Signals that an error has occurred within the Function Block		
E	ErrorID	WORD	Error identification		
B	Value	REAL	Value of the specified parameter in the datatype, as specified by the vendor		
Note: The parameters are defined in the table below					

FB-Name		MC_ReadBoolParameter		
This Function Block returns the value of a vendor specific parameter with datatype BOOL				
VAR_IN_OUT				
B	Axis	AXIS_REF	Reference to the axis	
VAR_INPUT				
B	Enable	BOOL	Get the value of the parameter continuously while enabled	
B	ParameterNumber	INT	Number of the parameter. One can also use symbolic parameter names which are declared as VAR CONST	
VAR OUTPUT				
B	Valid	BOOL	A valid output is available at the FB	
E	Busy	BOOL	The FB is not finished and new output values are to be expected	
B	Error	BOOL	Signals that an error has occurred within the Function bloc	
E	ErrorID	WORD	Error identification	
B	Value	BOOL	Value of the specified parameter in the datatype, as specified by the vendor	
Note: The parameters are defined in the table below				



以下定义参数已经由工作组进行了标准化。如果供应商提供此功能，应使用这些参数。

所有定义的只读参数在初始化阶段可能是可写的（依赖于供应商）。

这些参数可用于应用程序，通常不用于诸如操作面板等调试工具（驱动器不可见 - 只有轴位置可见）。

注意：最常用的参数可以通过功能块访问，且未在此列出。

（注意：PN 是参数编号，请参见 FB MC_ReadParameter / MC_WriteParameter 及布尔版本）

PN	Name	Datatype	B/E	R/W	Comments
1	CommandedPosition	REAL	B	R	Commanded position
2	SWLimitPos	REAL	E	R/W	Positive Software limit switch position
3	SWLimitNeg	REAL	E	R/W	Negative Software limit switch position
4	EnableLimitPos	BOOL	E	R/W	Enable positive software limit switch
5	EnableLimitNeg	BOOL	E	R/W	Enable negative software limit switch
6	EnablePosLagMonitoring	BOOL	E	R/W	Enable monitoring of position lag
7	MaxPositionLag	REAL	E	R/W	Maximal position lag
8	MaxVelocitySystem	REAL	E	R	Maximal allowed velocity of the axis in the motion system
9	MaxVelocityAppl	REAL	B	R/W	Maximal allowed velocity of the axis in the application
10	ActualVelocity	REAL	B	R	Actual velocity
11	CommandedVelocity	REAL	B	R	Commanded velocity
12	MaxAccelerationSystem	REAL	E	R	Maximal allowed acceleration of the axis in the motion system
13	MaxAccelerationAppl	REAL	E	R/W	Maximal allowed acceleration of the axis in the application
14	MaxDecelerationSystem	REAL	E	R	Maximal allowed deceleration of the axis in the motion system
15	MaxDecelerationAppl	REAL	E	R/W	Maximal allowed deceleration of the axis in the application
16	MaxJerkSystem	REAL	E	R	Maximum allowed jerk of the axis in the motion system
17	MaxJerkAppl	REAL	E	R/W	Maximum allowed jerk of the axis in the application

表 5: MC_ReadParameter 和 MC_WriteParameter 的参数

任何供应商或用户在列表末尾添加扩展也被允许，尽管这可能会影响不同平台之间的可移植性。参数编号从 0 到 999 被保留用于标准。大于 999 的数字表示供应商特定参数。

3.20. MC_WriteParameter & MC_WriteBoolParameter

FB-Name		MC_WriteParameter	
This Function Block modifies the value of a vendor specific parameter			
VAR_IN_OUT			
B	Axis	AXIS_REF	Reference to the axis
VAR_INPUT			
B	Execute	BOOL	Write the value of the parameter at rising edge
B	ParameterNumber	INT	Number of the parameter (correspondence between number and parameter is specified in the table above)
B	Value	REAL	New value of the specified parameter
E	ExecutionMode	MC_EXECUTION_MODE	Defines the chronological sequence of the FB. mcImmediately - the functionality is immediately valid and may influence the on-going motion but not the state (note: is the de- fault behaviour) mcQueued - Same functionality as buffer mode 'Buffered'
VAR_OUTPUT			
B	Done	BOOL	Parameter successfully written
E	Busy	BOOL	The FB is not finished and new output values are to be expected.
B	Error	BOOL	Signals that an error has occurred within the Function Block
E	ErrorID	WORD	Error identification
Notes: The parameters are defined in the table above (under MC_ReadParameter, writing allowed)			

3.21. MC_ReadDigitalInput

3.22. MC_ReadDigitalOutput

3.25. MC_ReadActualVelocity

3.26. MC_ReadActualTorque

3.31. MC_Reset

3.32. MC_DigitalCamSwitch

FB-Name: MC_DigitalSwitch			
This Function Block is the analogy to switches on a motor shaft. It commands a group of discrete output bits to switch in analogy to a set of mechanical cam-controlled switches connected to an axis. Readout and hardware movement are allowed			
VAR_IN_OUT			
B	Axis	AXIS_REF	Reference to the axis
B	Switches	MC_CAMSWITCH_REF	Reference to the switching tables
E	OnCompensation	MC_OFFCOMP_REF	Reference to the signal outputs, directly related to the motor speed tracks, since 32 per function block (First output = first TrackNumber)
E	OffCompensation	MC_TRACK_REF	Reference to structures containing track related properties, e.g. the ON and OFF compensation per output/track
VAR_OUTPUT			
B	Ready	BOOL	Enables for 'Switches' outputs
B	OnPosition	REAL	At rise of READY, enables the different tracks. Based on address data is related to the lower TrackNumber. With data REF (in 1st comp. TRACK) the related TrackNumber is enabled
B	OffPosition	REAL	Defines the centre for each value (e.g. position, velocity, TrackNumber) on one value mcOnVelocity - TrackNumber on actual value
VAR_OUTPUT			
B	OnCompensation	REAL	The compensated tracks are enabled
B	OffCompensation	REAL	The FB is not finished and new output values are to be expected
B	Busy	BOOL	Signals that an error has occurred within the Function Block
B	Error	BOOL	Error identification
E	ErrorID	WORD	Error identification

MC_CAMSWITCH_REF is a pointer to a table of 32 pointers to MC_CAMSWITCH_REF, which contains pointers to the pointers data. MC_OFFCOMP_REF is a pointer to a single structure linked to the (physical) output. MC_TRACK_REF is a pointer to a single structure containing the tracking properties for each output. The output is a track. A track is a set of switches related to one output. It can contain the reference to the output also. The functionality is contained inside PDS - Phase of Position of Programme Data Table

Note: - MC_CAMSWITCH_REF is a vendor specific reference to the pattern data. - MC_OFFCOMP_REF is a vendor specific structure related to the (global) output. - MC_TRACK_REF is vendor specific structure containing the track properties, e.g. the compensation per 1K track is a set of entries related to one output). It can contain the reference to the output also. This functionality is sometimes called PES - Phase or Position or Programmable Limit Switch.

MC_CAMSWITCH_REF 结构中的基本元素

B/E	Parameter	Type	Description
B	TrackNumber	INT	TrackNumber is the reference to the track
B	FirstOnPosition [u]	REAL	Lower boundary where the switch is ON
B	LastOnPosition [u]	REAL	Upper boundary where the switch is ON
E	AxisDirection	INT	Both (=0; Default); Positive (1); Negative (2)
E	CamSwitchMode	INT	Position based (= 0; Default) ; Time based (= 1)
E	Duration	TIME	Coupled to time based CamSwitchMode

MC_TRACK_REF 数组结构中的基本元素

B/E	Parameter	Type	Description
E	OnCompensation	TIME	Compensation time with which the switching on is advanced or delayed in time per track.
E	OffCompensation	TIME	Time compensation the switching off is delayed per track
E	Hysteresis [u]	REAL	Distance from the switching point (in positive and negative direction) in which the switch is not executed until the axis has left this area, in order to avoid multiple switching around the switching point.

这个凸轮的定义有一个起始位置和一个结束位置，因此用户可以定义每一个单独的凸轮，具有第一个开启位置和最后一个开启位置（或时间）。这个功能块类似于机械凸轮，但具有额外的优点，即输出可以设置为一定的时间，并且可以提供时间补偿和滞后。

CamSwitchMode 可以是位置、时间或其他额外的特定于供应商的类型。

持续时间：时间，时间凸轮的输出为开启状态

时间补偿（OnCompensation 和 OffCompensation）可以是正值或负值。负值意味着输出在达到开关位置之前就会发生变化。

滞后：这个参数避免了在轴接近开关点并且实际位置在开关位置附近抖动时输出不断切换的现象。滞后是 MC_TRACK_REF 的一部分，这意味着每条轨道可以有不同的滞后。

MC_CAMSWITCH_REF 的示例

Parameter	Type	Switch01	Switch02	Switch03	Switch04	...	SwitchN
TrackNumber	INTEGER	1	1	1	2		
FirstOnPosition [u]	REAL	2000	2500	4000	3000		
LastOnPosition [u]	REAL	3000	3000	1000	—		
AxisDirection	INTEGER	1 = Pos	2 = Neg	0 = Both	0 = Both		
CamSwitchMode	INTEGER	0 = Position	0 = Position	0 = Position	1 = Time		
Duration	TIME	—	—	—	1350		

注意：值为示例

下面的示例使用了上面 MC_CAMSWITCH_REF 示例中的值。它既不使用 On/OffCompensation，也不使用滞后。

这是在轴连续向正方向移动时输出的行为。该轴是一个模态轴，模态长度为 5000u。

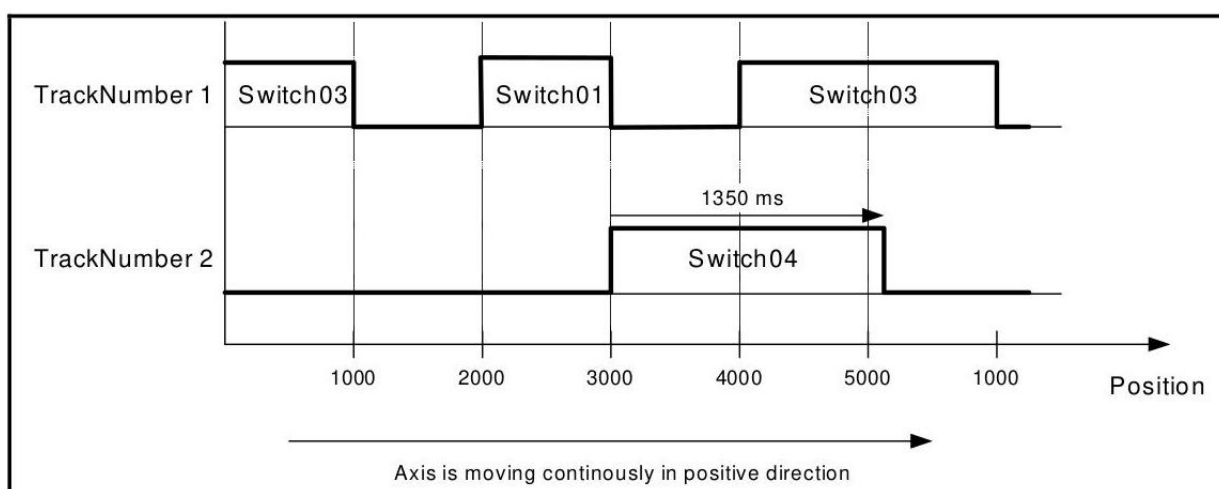


图 37: MC_DigitalCamSwitch 示例

Switch01 的详细描述。

该示例还额外使用了 OnCompensation -125ms 和 OffCompensation +250ms。

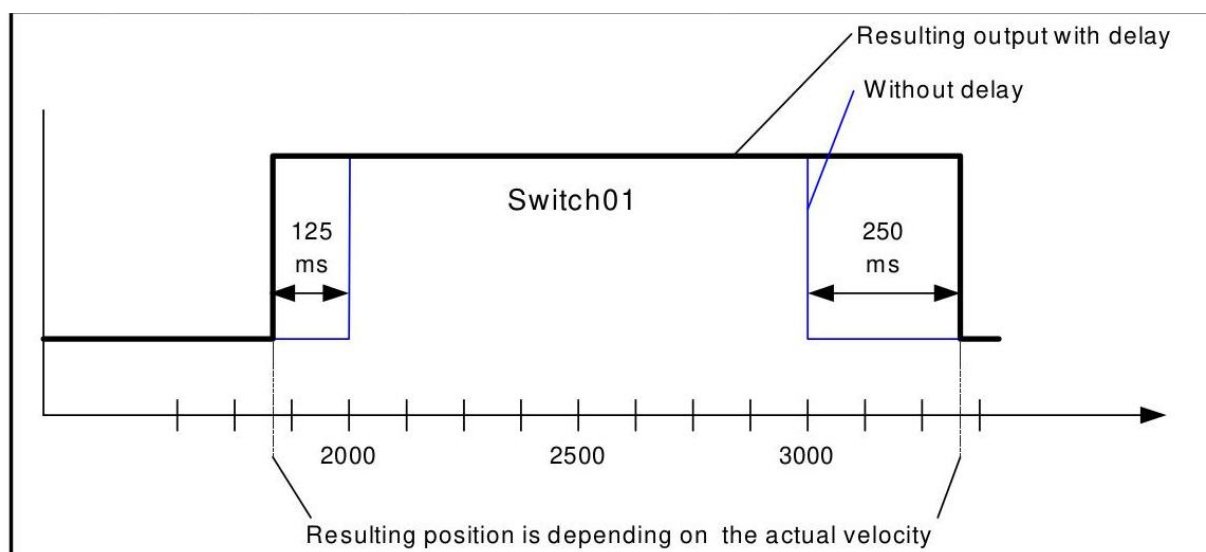


图 38: Switch01 的详细描述。

以下是在轴连续向负方向移动时输出的行为，没有使用 On/OffCompensation 和滞后。

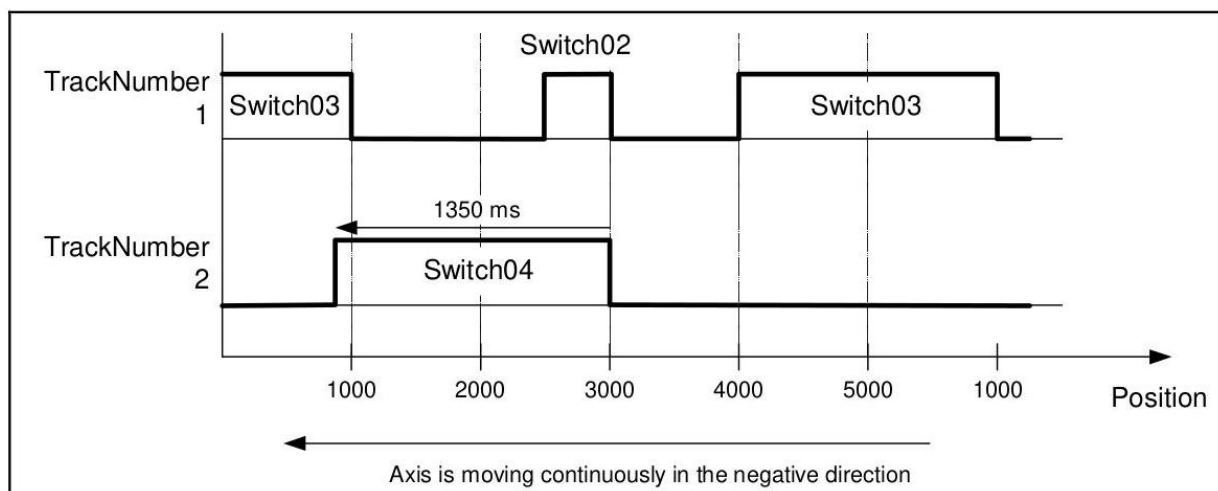


图 39: 负方向的示例

3.33. MC_TouchProbe

	FB-Name		MC_TouchProbe	
This Function Block is used to record an axis position at a trigger event				
VAR_IN_OUT				
	B	Axis	AXIS_REF	Reference to the axis
	E	TriggerInput	MC_TRIGGER_REF	Reference to the trigger signal source. Trigger input may be specified by the AXIS_REF.
VAR INPUT				
	B	Execute	BOOL	Starts touch probe recording at rising edge
	E	WindowOnly	BOOL	If SET, only use the window (defined hereunder) to accept trigger events
	E	FirstPosition	REAL	Start position from where (positive direction) trigger events are accepted (in technical units [u]). Value included in win- dow
	E	LastPosition	REAL	Stop position of the window (in technical units [u]). Value included in window.
VAR OUTPUT				
	B	Done	BOOL	Trigger event recorded
	E	Busy	BOOL	The FB is not finished and new output values are to be ex- pected
	E	CommandAborted	BOOL	'Command' is aborted by another command (MC_AbortTrigger)
	B	Error	BOOL	Signals that an error has occurred within the Function Block
	E	ErrorID	WORD	Error identification
	B	RecordedPosition	REAL	Position where trigger event occurred (in technical units [u])

备注:

1. 该设置旨在单次操作，也就是说，在“执行”处上升沿后的第一个事件有效于记录，仅此而已。可能的后续事件将被忽略。
2. 一个功能块实例应精确地表示一个探测命令。
3. 如果在同一探头和轴上有多个实例，则 MC_TRIGGER_REF 的元素应扩展以包含 TouchProbeID - 唯一探测命令的标识 - 这可以链接到 MC_AbortTrigger。

AXIS_REF	MC_TouchProbe		AXIS_REF
	Axis	Axis	
MC_TRIGGER_REF	TriggerInput	TriggerInput	MC_TRIGGER_REF
BOOL	Execute	Done	BOOL
BOOL	WindowOnly	Busy	BOOL
REAL	FirstPosition	CommandAborted	BOOL
REAL	LastPosition	Error	BOOL
		ErrorID	WORD
		RecordedPosition	REAL

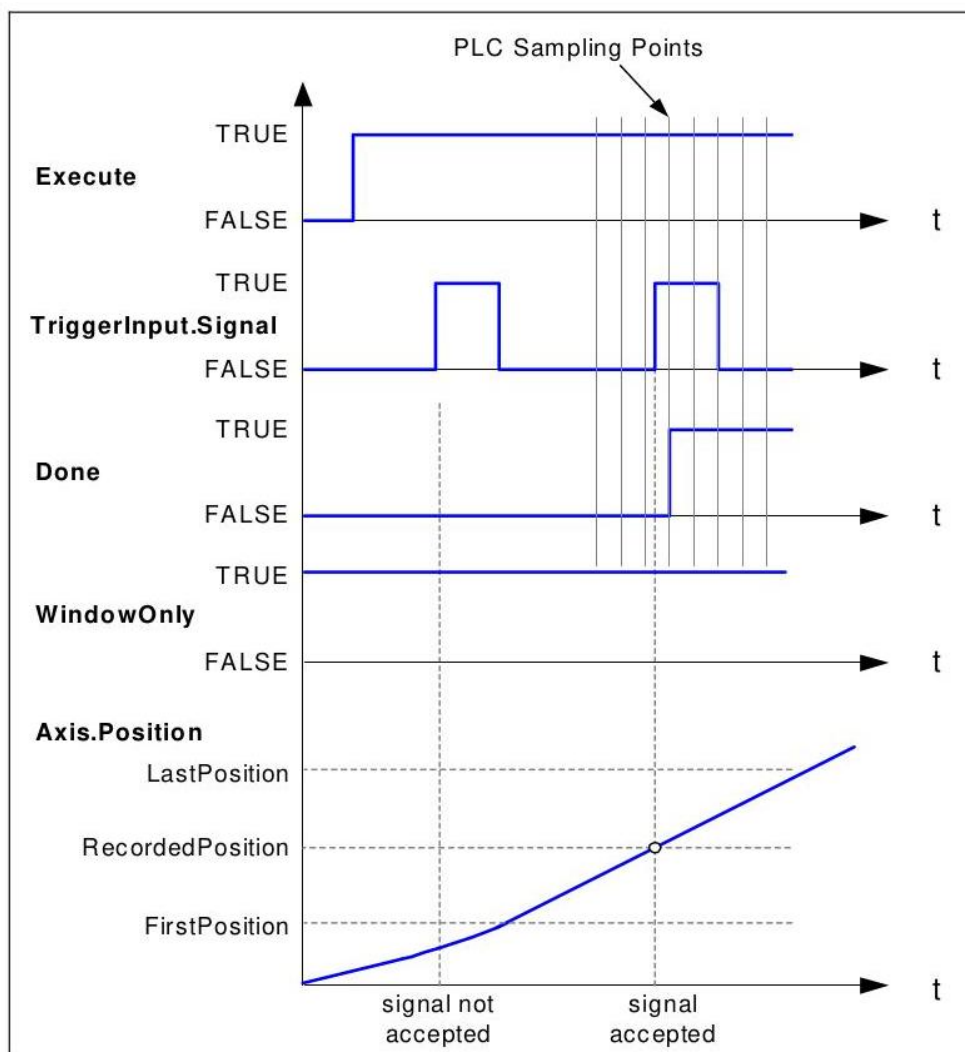


图 40: 定时示例 MC_TouchProbe

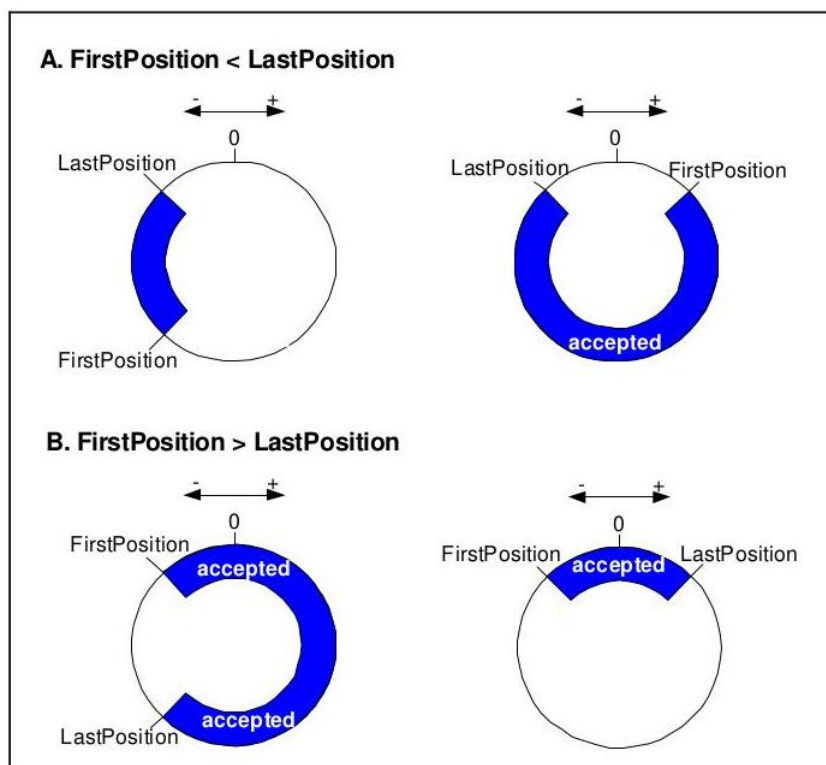


图 41: 触发事件被接受的窗口示例（适用于模轴）

4. Multi-Axis Function Blocks

通过多轴功能块，两个或多个轴之间存在同步关系。该同步可以与时间或位置相关。通常，这种关系是在一个主轴和一个或多个从轴之间。主轴可以是一个虚拟轴。

从状态图的角度来看，与凸轮和齿轮相关的多轴功能块可以被视为在一个状态下的主轴（例如：MC_MoveContinuous），而从轴在一个特定的同步状态下，称为“SynchronizedMotion”（见状态图，第 2.1 章）。

4.1. Remarks to Camming

机械凸轮是机械联动中一个旋转或滑动的零件，特别用于将旋转运动转换为线性运动或反之。它通常是旋转轮（例如，偏心轮）或轴（例如，不规则形状的圆柱体）的一部分，这些零件在其圆形路径上一个或多个点上撞击杠杆。凸轮可以是一个简单的齿，如用于向蒸汽锤传递动力脉冲的那种，或者是一个偏心盘或其他形状，产生的在跟随者中的平滑往复（来回）运动，跟随者是与凸轮接触的杠杆。

因此，凸轮在主轴和一个或多个从轴之间创建了一个连接，在位置/位置模式下（见下图）。

通过电动机和驱动器，可以创建相同的位置/位置关系，但在这种情况下通过所谓的凸轮表列出位置。因此，关系被转换为软件和控制。

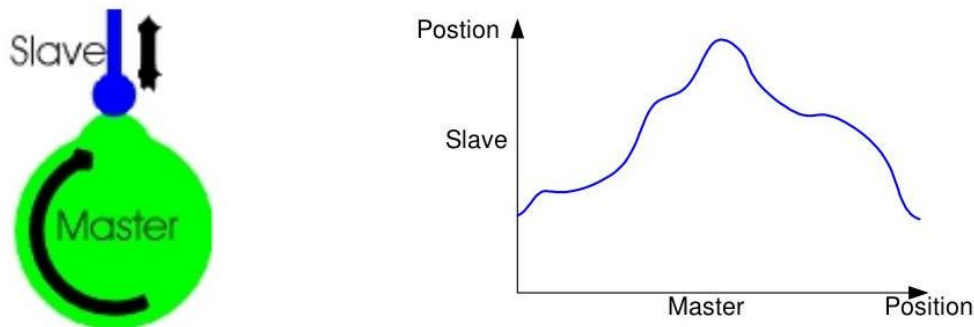


图 42：凸轮轮廓示意图

基本上，可以区分两种类型的凸轮用于模态和线性（或有限）主轴：

- 周期模式 - 周期模式 - 持续重复执行凸轮轮廓，即使凸轮轮廓与模态不匹配。这意味着对于一个模态为 360 度的模态轴，而凸轮轮廓指定为 90 度，它将在模态中执行 4 次。在反向模式下，轮廓是以相反的方式执行的。
- 非周期模式 - 凸轮轮廓仅运行一次。如果主位置在凸轮轮廓之外，从轴保持同步运动并保持最后位置。在反向模式下，达到“结束轮廓”位置后不执行凸轮轮廓。上述的 90 度示例只会运行一次。

拉升可以通过几个组合的凸轮表进行顺序执行，例如升坡、生产周期和降坡。在不同的凸轮曲线之间可能会有一个间隔（等待触发）。然而，可以使用缓冲模式或使用输出 'EndOfProfile' 来开始下一个配置。

CAM table

Camming 是通过一个表（二维 - 一起描述主轴和从轴位置）或两个表 - 分别针对主轴和从轴位置进行的。该表应该严格单调上升或下降，主轴可以前后反向。

在 CAM 运行期间，允许并且可以更改表格，并在 CAM 运行期间更改表中的元素。

CAM 表（主轴、从轴）的生成和填充由一个特定于供应商的外部工具执行。FB MC_CamIn 与表的耦合也是特定于供应商的。

主轴和从轴可能有不同的表示形式：

- 绝对值
- 相对于起始位置
- 相对步长（与先前位置的差异）
- 等距或不等距值。
- 多项式格式。在这种情况下，凸轮完全在从表中描述。主表为零。

拥有不同功能块以实现凸轮功能的优点是程序执行流程更加透明，以及执行性能更好。

注意：

不要求主轴保持静止

- 如果实际的主轴和从轴位置在执行 MC_CamIn 时与偏移值不对应，

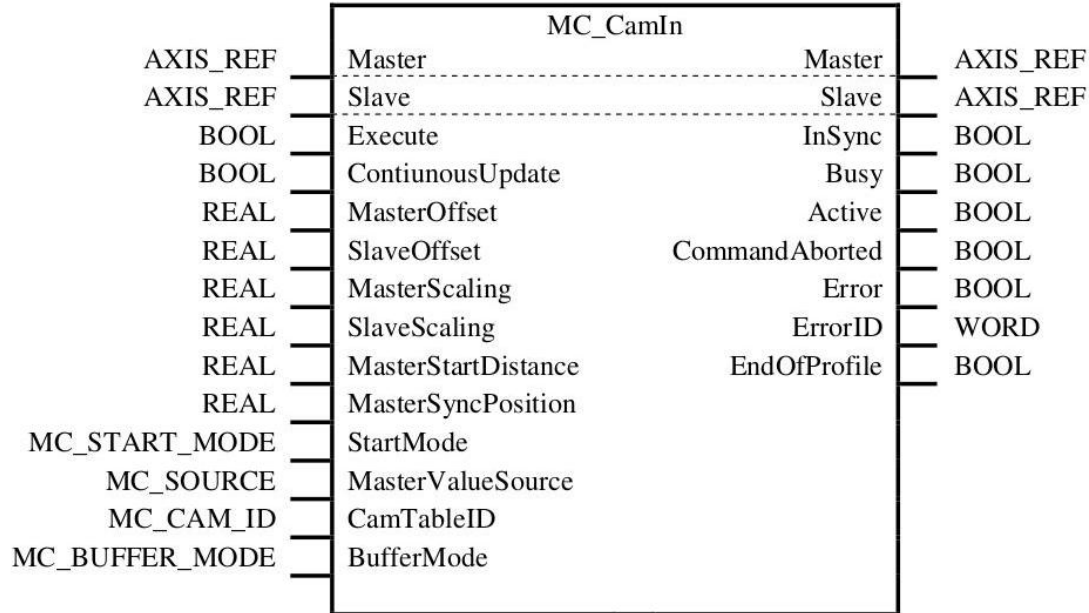
则会发生错误，或者系统会自动处理差异

凸轮被放置为绝对位置或相对于当前主轴和从轴位置。

绝对：主轴和从轴之间的轮廓被视为绝对关系。

相对：主轴和从轴之间的关系处于相对模式。

- Ramp-in 是特定于供应商的模式。它可以耦合到额外参数，例如主距离参数、加速度参数或其他供应商特定参数，使得从轴可以融入到凸轮轮廓中（“飞行耦合”）
- 这个 FB 并没有与 MC_CamTableSelect FB 合并，因为这种分离允许动态更改 - 一个机械类比是一个焊接了额外恒定层厚度的凸轮。由于这个原因，从属位置具有恒定的偏移，如果假设从属滑块是线性导向的，那么这个偏移可以被解释为主轴的轴偏移。



4.4. MC_CamOut

FB-Name		MC_CamOut	
This Function Block disengages the Slave axis from the Master axis immediate			
VAR_IN_OUT			
B	Slave	AXIS REF	Reference to the slave axis
VAR INPUT			
B	Execute	BOOL	Start to disengage the slave from the master
VAR OUTPUT			
B	Done	BOOL	Disengaging completed
E	Busy	BOOL	The FB is not finished and new output values are to be expected
B	Error	BOOL	Signals that an error has occurred within the Function Block
E	ErrorID	WORD	Error identification

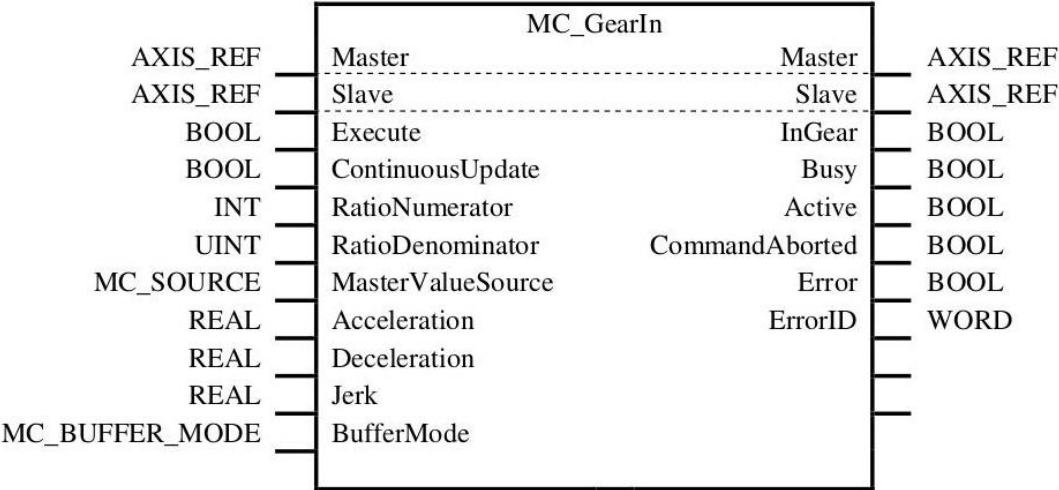
备注：

- 假设该命令后面跟着另一个命令，例如 MC_Stop、MC_GearIn 或其他任何命令。如果没有新的命令，默认条件应为：保持最后的速度。
- 在发出 FB 之后，直到发出下一个 FB，从动轴上没有活跃的 FB（这可能会导致问题，因为没有运动命令控制该轴）。另外，可以通过 MC_ReadActualVelocity 读取实际速度，并使用实际速度作为输入在从动轴上发出 MC_MoveVelocity。FB 出现在这里是由于兼容性原因。

4.5. MC_GearIn

FB-Name		MC_GearIn	
This Function Block commands a ratio between the VELOCITY of the slave and master axis.			
VAR: R, OUT			
R	Master	AXIS_REF	Reference to the master axis
R	Slave	AXIS_REF	Reference to the slave axis
VAR INPUT			
R	Execute	BOOL	Start the gearing process at the rising edge
E	ContinuousUpdate	BOOL	See 2.4.8 This input 'ContinuousUpdate'
R	RatioNumerator	INT	Gear ratio Numerator
R	RatioDenominator	UINT	Gear ratio Denominator
E	MasterValueSource	MC_SOURCE	Defines the source for synchronization: actSetValue - Synchronization on master set value actActualValue - Synchronization on master actual value
E	Acceleration	REAL	Acceleration for gearing in
E	Deceleration	REAL	Deceleration for gearing in
E	Jerk	REAL	Jerk of Gearing
E	BufferMode	MC_BUFFER_MODE	Defines the chronological sequence of the FB: See 2.4.2 Aborting versus Buffered modes
VAR OUTPUT			
R	InGear	BOOL	Is TRUE if the set value = the commanded value
E	Busy	BOOL	The FB is not finished and new output values are to be expected
E	Active	BOOL	Indicates that the FB has control on the axis
E	CommandAborted	BOOL	'Command' is aborted by another command
R	Error	BOOL	Signals that an error has occurred within the Function Block
E	ErrorID	WORD	Error identification

Notes: 1. The slave ramps up to the ratio of the master velocity and locks in when this is reached. Any lost distance during synchronization is not caught up. 2. The gearing ratio can be changed while MC_GearIn is running, using a consecutive MC_GearIn command without the necessity to MC_GearOut first. 3. After being 'InGear', a position locking or just a velocity locking is system specific.



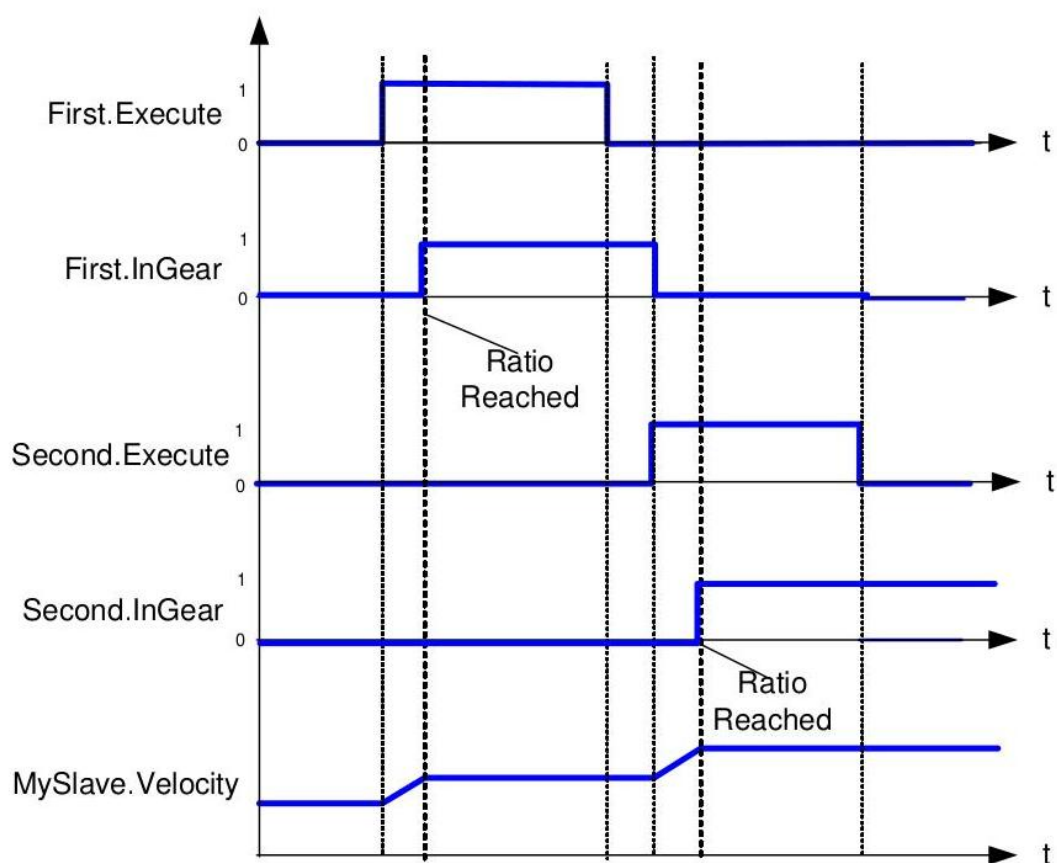
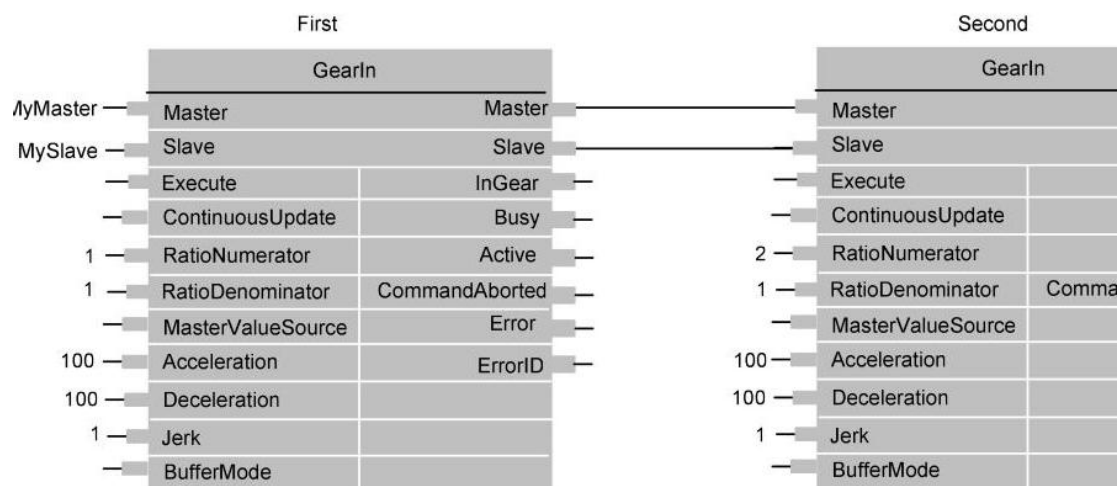


图 43: 齿轮时序图

4.6. MC_GearOut

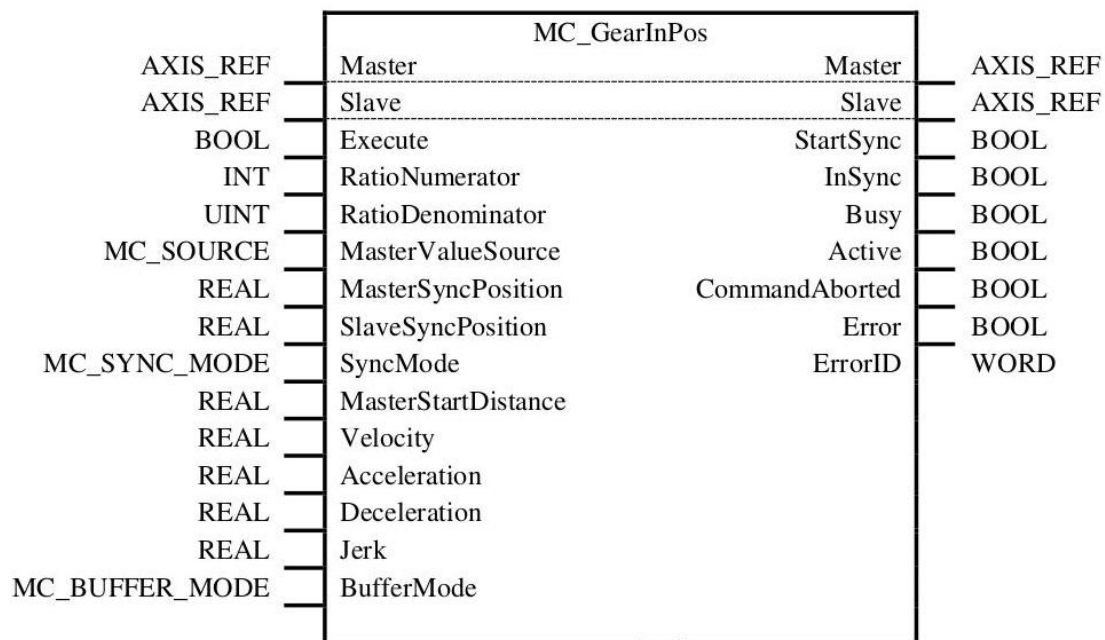
Notes:

FB-Name		MC_GearOut	
This Function Block disengages the Slave axis from the Master axis			
VAR_IN_OUT			
B	Slave	AXIS_REF	Reference to the slave axis
VAR INPUT			
B	Execute	BOOL	Start disengaging process at the rising edge
VAR OUTPUT			
B	Done	BOOL	Disengaging completed
E	Busy	BOOL	The FB is not finished and new output values are to be expected
B	Error	BOOL	Signals that an error has occurred within the Function Block
E	ErrorID	WORD	Error identification

备注：

- 假设此命令后面跟着另一个命令，例如 MC_Stop、MC_GearIn 或任何其他命令。如果没有新的命令，则默认条件应为：保持最后速度。
- 在发出 FB 之后，直到发出下一个 FB，奴从轴上没有活跃的 FB（这可能导致问题，因为没有运动命令控制该轴）。或者，可以通过 MC_ReadActualVelocity 读取实际速度，并对奴从轴发出带有实际速度作为输入的 MC_MoveVelocity。此处使用 FB 是出于兼容性原因。

1. 如果实现了'MasterStartDistance'，则任何先前的运动将持续，直到主设备以正确的方向（根据'MasterStartDistance'的符号）穿过'Master-SyncPosition' - 'MasterStartDistance'。此时，输出'StartSync' 被设置。当在同步发生之前对'Slave' 轴执行'Stop' 命令时，会抑制同步，并且功能块发出'CommandAborted'。
2. 如果未指定'MasterStartDistance'，系统本身可以根据其他相关输入计算'StartSync' 的设定点。
3. 'SyncModes' 中的'CatchUp' 和'SlowDown' 之间的区别在于同步所需的能量。'SlowDown' 相对于'CatchUp' 消耗最低的能量。



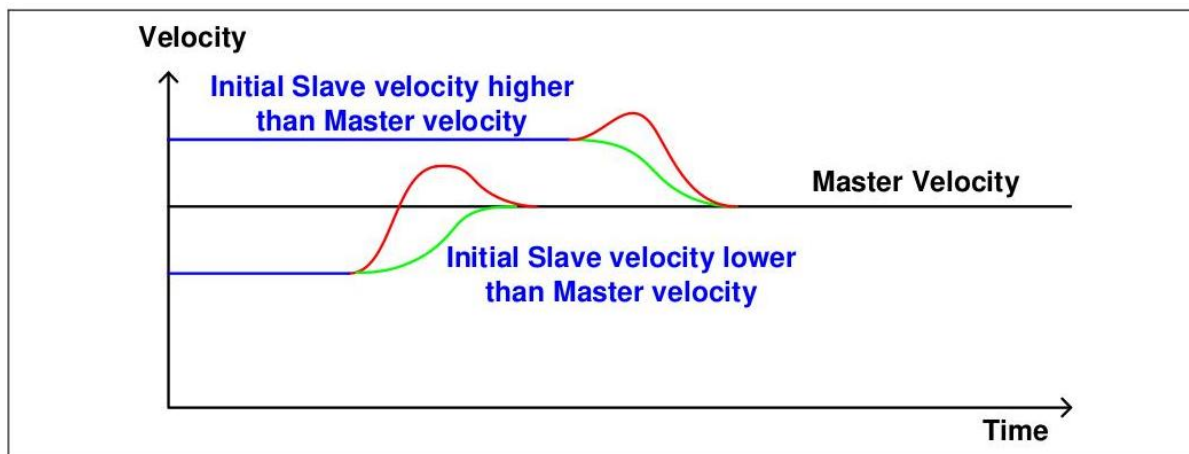


图 45: 'SyncModes' 'SlowDown' (绿色) 和 'CatchUp' (红色) 在不同奴从初始速度下的区别示例。

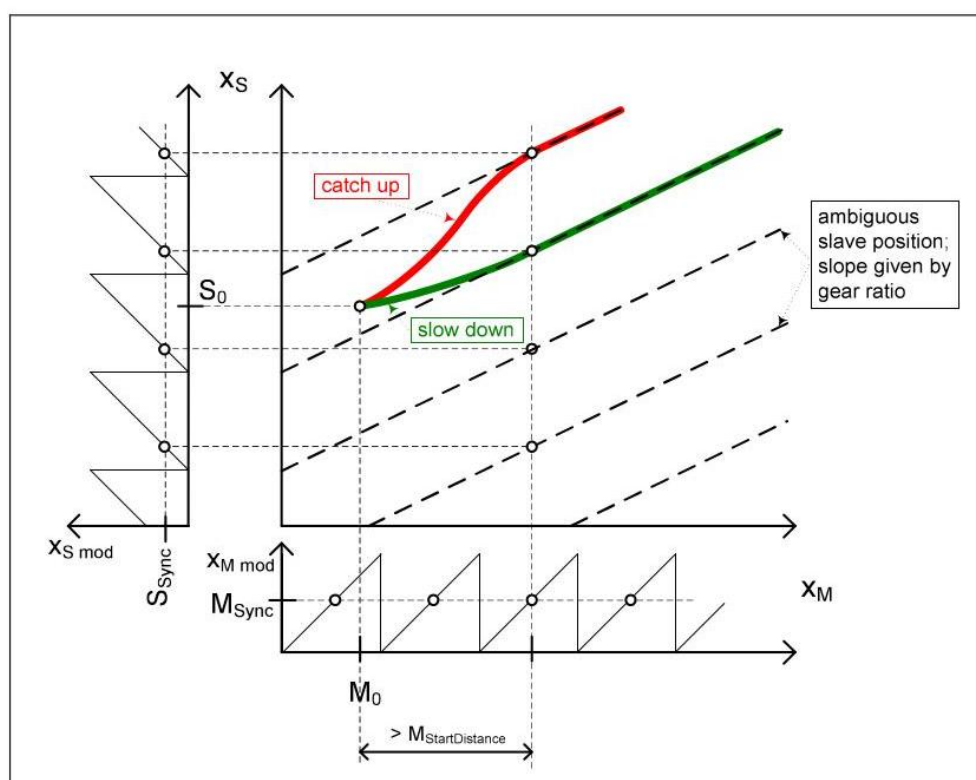


图 46: MC_GearInPos 的示例，其中从属体的初始速度与主质量方向相同。

4.8. MC_PhasingAbsolute

Parameter	Value	Description
MC_PhasingAbsolute	0	MC_PhasingAbsolute
MC_PhasingAbsolute	1	MC_PhasingAbsolute
MC_PhasingAbsolute	2	MC_PhasingAbsolute
MC_PhasingAbsolute	3	MC_PhasingAbsolute
MC_PhasingAbsolute	4	MC_PhasingAbsolute
MC_PhasingAbsolute	5	MC_PhasingAbsolute
MC_PhasingAbsolute	6	MC_PhasingAbsolute
MC_PhasingAbsolute	7	MC_PhasingAbsolute
MC_PhasingAbsolute	8	MC_PhasingAbsolute
MC_PhasingAbsolute	9	MC_PhasingAbsolute
MC_PhasingAbsolute	10	MC_PhasingAbsolute
MC_PhasingAbsolute	11	MC_PhasingAbsolute
MC_PhasingAbsolute	12	MC_PhasingAbsolute
MC_PhasingAbsolute	13	MC_PhasingAbsolute
MC_PhasingAbsolute	14	MC_PhasingAbsolute
MC_PhasingAbsolute	15	MC_PhasingAbsolute
MC_PhasingAbsolute	16	MC_PhasingAbsolute
MC_PhasingAbsolute	17	MC_PhasingAbsolute
MC_PhasingAbsolute	18	MC_PhasingAbsolute
MC_PhasingAbsolute	19	MC_PhasingAbsolute
MC_PhasingAbsolute	20	MC_PhasingAbsolute
MC_PhasingAbsolute	21	MC_PhasingAbsolute
MC_PhasingAbsolute	22	MC_PhasingAbsolute
MC_PhasingAbsolute	23	MC_PhasingAbsolute
MC_PhasingAbsolute	24	MC_PhasingAbsolute
MC_PhasingAbsolute	25	MC_PhasingAbsolute
MC_PhasingAbsolute	26	MC_PhasingAbsolute
MC_PhasingAbsolute	27	MC_PhasingAbsolute
MC_PhasingAbsolute	28	MC_PhasingAbsolute
MC_PhasingAbsolute	29	MC_PhasingAbsolute
MC_PhasingAbsolute	30	MC_PhasingAbsolute
MC_PhasingAbsolute	31	MC_PhasingAbsolute
MC_PhasingAbsolute	32	MC_PhasingAbsolute
MC_PhasingAbsolute	33	MC_PhasingAbsolute
MC_PhasingAbsolute	34	MC_PhasingAbsolute
MC_PhasingAbsolute	35	MC_PhasingAbsolute
MC_PhasingAbsolute	36	MC_PhasingAbsolute
MC_PhasingAbsolute	37	MC_PhasingAbsolute
MC_PhasingAbsolute	38	MC_PhasingAbsolute
MC_PhasingAbsolute	39	MC_PhasingAbsolute
MC_PhasingAbsolute	40	MC_PhasingAbsolute
MC_PhasingAbsolute	41	MC_PhasingAbsolute
MC_PhasingAbsolute	42	MC_PhasingAbsolute
MC_PhasingAbsolute	43	MC_PhasingAbsolute
MC_PhasingAbsolute	44	MC_PhasingAbsolute
MC_PhasingAbsolute	45	MC_PhasingAbsolute
MC_PhasingAbsolute	46	MC_PhasingAbsolute
MC_PhasingAbsolute	47	MC_PhasingAbsolute
MC_PhasingAbsolute	48	MC_PhasingAbsolute
MC_PhasingAbsolute	49	MC_PhasingAbsolute
MC_PhasingAbsolute	50	MC_PhasingAbsolute
MC_PhasingAbsolute	51	MC_PhasingAbsolute
MC_PhasingAbsolute	52	MC_PhasingAbsolute
MC_PhasingAbsolute	53	MC_PhasingAbsolute
MC_PhasingAbsolute	54	MC_PhasingAbsolute
MC_PhasingAbsolute	55	MC_PhasingAbsolute
MC_PhasingAbsolute	56	MC_PhasingAbsolute
MC_PhasingAbsolute	57	MC_PhasingAbsolute
MC_PhasingAbsolute	58	MC_PhasingAbsolute
MC_PhasingAbsolute	59	MC_PhasingAbsolute
MC_PhasingAbsolute	60	MC_PhasingAbsolute
MC_PhasingAbsolute	61	MC_PhasingAbsolute
MC_PhasingAbsolute	62	MC_PhasingAbsolute
MC_PhasingAbsolute	63	MC_PhasingAbsolute
MC_PhasingAbsolute	64	MC_PhasingAbsolute
MC_PhasingAbsolute	65	MC_PhasingAbsolute
MC_PhasingAbsolute	66	MC_PhasingAbsolute
MC_PhasingAbsolute	67	MC_PhasingAbsolute
MC_PhasingAbsolute	68	MC_PhasingAbsolute
MC_PhasingAbsolute	69	MC_PhasingAbsolute
MC_PhasingAbsolute	70	MC_PhasingAbsolute
MC_PhasingAbsolute	71	MC_PhasingAbsolute
MC_PhasingAbsolute	72	MC_PhasingAbsolute
MC_PhasingAbsolute	73	MC_PhasingAbsolute
MC_PhasingAbsolute	74	MC_PhasingAbsolute
MC_PhasingAbsolute	75	MC_PhasingAbsolute
MC_PhasingAbsolute	76	MC_PhasingAbsolute
MC_PhasingAbsolute	77	MC_PhasingAbsolute
MC_PhasingAbsolute	78	MC_PhasingAbsolute
MC_PhasingAbsolute	79	MC_PhasingAbsolute
MC_PhasingAbsolute	80	MC_PhasingAbsolute
MC_PhasingAbsolute	81	MC_PhasingAbsolute
MC_PhasingAbsolute	82	MC_PhasingAbsolute
MC_PhasingAbsolute	83	MC_PhasingAbsolute
MC_PhasingAbsolute	84	MC_PhasingAbsolute
MC_PhasingAbsolute	85	MC_PhasingAbsolute
MC_PhasingAbsolute	86	MC_PhasingAbsolute
MC_PhasingAbsolute	87	MC_PhasingAbsolute
MC_PhasingAbsolute	88	MC_PhasingAbsolute
MC_PhasingAbsolute	89	MC_PhasingAbsolute
MC_PhasingAbsolute	90	MC_PhasingAbsolute
MC_PhasingAbsolute	91	MC_PhasingAbsolute
MC_PhasingAbsolute	92	MC_PhasingAbsolute
MC_PhasingAbsolute	93	MC_PhasingAbsolute
MC_PhasingAbsolute	94	MC_PhasingAbsolute
MC_PhasingAbsolute	95	MC_PhasingAbsolute
MC_PhasingAbsolute	96	MC_PhasingAbsolute
MC_PhasingAbsolute	97	MC_PhasingAbsolute
MC_PhasingAbsolute	98	MC_PhasingAbsolute
MC_PhasingAbsolute	99	MC_PhasingAbsolute

- 相位移的“相位”、“速度”、“加速度”、“减速度”和“撞击”由 FB 控制。
- 供比较：MC_MoveSuperimposed 也可以用于作用于从动轴。MC_Phasing 作用于主侧，从从动轴的角度来看。

在这个例子中，从动轴跟随主轴（红色 - 周期性）具有正弦凸轮轮廓。两个从动位置（绿色）和从动速度（蓝色）被显示出来。相位作用在从动轴上的效果。

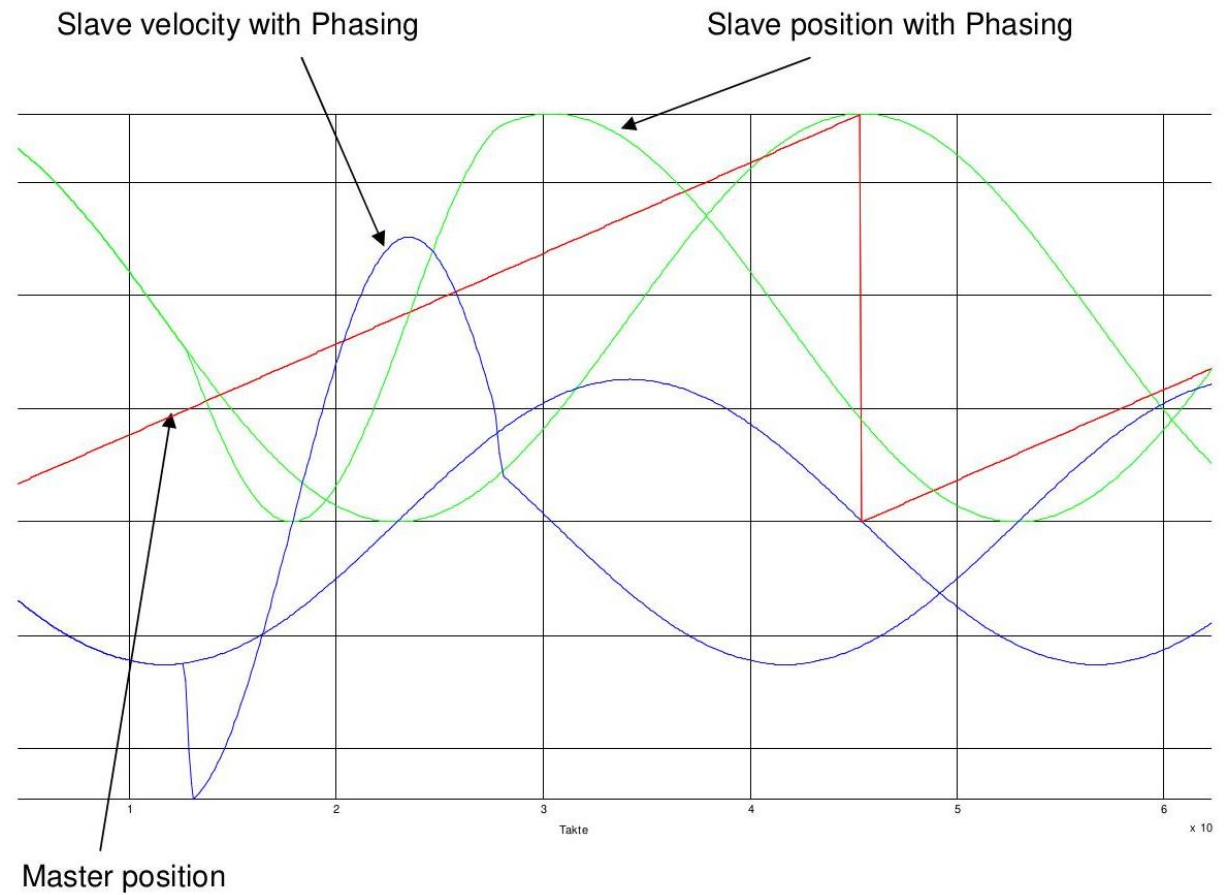


图 49: MC_Phasing 的示例 - 绝对和相对两者

4.9. MC_PhasingRelative

FB-Name: MC_PhasingRelative			
The Function Block combines the motion of 2 axes into a third axis with selectable combination method. Basically it is a calculation of a new position output based on the 2 position outputs of the input axes. The FB is referred to the state diagram like a synchronized motion type. An application example one can work with separate profiles synchronized to an object on a moving belt, or a rotating belt with flexible control distance to be cut.			
1	Start	Bool	Start the combination process at the rising edge
2	Stop	Bool	Stop the combination process at the rising edge
3	Reset	Bool	Reset the combination process at the rising edge
4	Axis1	Axis	Reference to the first master axis
5	Axis2	Axis	Reference to the second master axis
6	Axis3	Axis	Reference to the resulting combined axis. Can be a virtual axis or linked directly to a real axis
7	VarInput	Var	Start the combination process at the rising edge
8	VarOutput	Var	See 2.1.4 The input "Combined value"
9	CombinedValue	Var	Define the type of combination applied to Axis1 and Axis2: Addition of the 2 input axis positions, Subtraction of the 2 input axis positions
10	CombinedValue1	Var	Parameter for the gain factor for master axis 1 towards the slave
11	CombinedValue2	Var	Corresponding denominator for master axis 1
12	CombinedValue3	Var	Numerator for the gain factor for master axis 2 towards the slave
13	CombinedValue4	Var	Corresponding denominator for master axis 2
14	MasterVelocity1	Var	Define the master for synchronization for master axis 1: masterValue - Synchronization on master set value masterValue - Synchronization on master actual value
15	MasterVelocity2	Var	Define the master for synchronization for master axis 2: masterValue - Synchronization on master set value masterValue - Synchronization on master actual value
16	BufferMode	Var	Define the behavior of the axis modes are "Disabling", "Holdoff", "Braking"
17	VarOutput	Var	See 2.1.4 The input "Combined value"
18	IsTrue	Bool	Is TRUE if the set value is the commanded value
19	IsSet	Bool	The FB is not finished and new output values are to be expected
20	IsAct	Bool	Indicates that the FB has control on the combined axis
21	Commanded	Var	Commanded is shared by another command
22	Done	Bool	Signals that an error has occurred within the Function Block
23	ErrorID	Word	Error identification
c. To start the motion, the ED has to be interpreted by another ED inside a new segment			

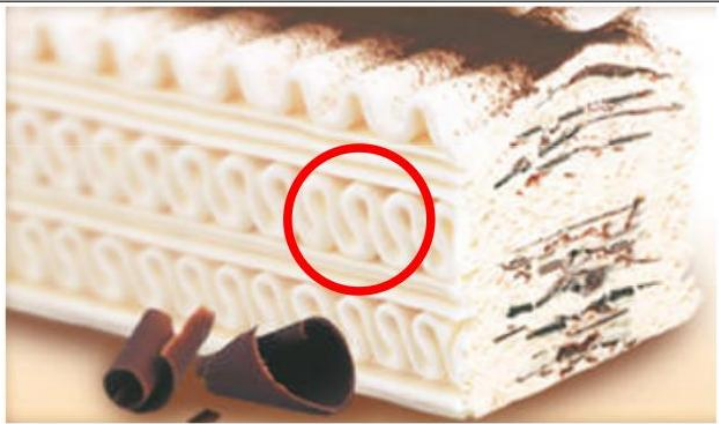
注意：

- 相位转换的“相位”、“速度”、“加速度”、“减速度”和“瞬变”由 FB 控制。
- 作为比较：MC_MoveSuperimposed 也可以用于作用于从动轴。MC_Phasing 作用于主侧，从从动侧来看。

4.10. MC_CombineAxes

FB-Name: MC_CombineAxes			
The Function Block combines the motion of 2 axes into a third axis with selectable combination method. Basically it is a calculation of a new position output based on the 2 position outputs of the input axes. The FB is referred to the state diagram like a synchronized motion type. An application example one can work with separate profiles synchronized to an object on a moving belt, or a rotating belt with flexible control distance to be cut.			
1	Start	Bool	Start the combination process at the rising edge
2	Stop	Bool	Stop the combination process at the rising edge
3	Reset	Bool	Reset the combination process at the rising edge
4	Axis1	Axis	Reference to the first master axis
5	Axis2	Axis	Reference to the second master axis
6	Axis3	Axis	Reference to the resulting combined axis. Can be a virtual axis or linked directly to a real axis
7	VarInput	Var	Start the combination process at the rising edge
8	VarOutput	Var	See 2.1.4 The input "Combined value"
9	CombinedValue	Var	Define the type of combination applied to Axis1 and Axis2: Addition of the 2 input axis positions, Subtraction of the 2 input axis positions
10	CombinedValue1	Var	Parameter for the gain factor for master axis 1 towards the slave
11	CombinedValue2	Var	Corresponding denominator for master axis 1
12	CombinedValue3	Var	Numerator for the gain factor for master axis 2 towards the slave
13	CombinedValue4	Var	Corresponding denominator for master axis 2
14	MasterVelocity1	Var	Define the master for synchronization for master axis 1: masterValue - Synchronization on master set value masterValue - Synchronization on master actual value
15	MasterVelocity2	Var	Define the master for synchronization for master axis 2: masterValue - Synchronization on master set value masterValue - Synchronization on master actual value
16	BufferMode	Var	Define the behavior of the axis modes are "Disabling", "Holdoff", "Braking"
17	VarOutput	Var	See 2.1.4 The input "Combined value"
18	IsTrue	Bool	Is TRUE if the set value is the commanded value
19	IsSet	Bool	The FB is not finished and new output values are to be expected
20	IsAct	Bool	Indicates that the FB has control on the combined axis
21	Commanded	Var	Commanded is shared by another command
22	Done	Bool	Signals that an error has occurred within the Function Block
23	ErrorID	Word	Error identification
c. To start the motion, the ED has to be interpreted by another ED inside a new segment			

MC_CombineAxes 可以生成特殊的同步运动，这在其他方法中是无法实现或复杂的。在以下示例中，一个 CAM FB 和一个齿轮 FB 的结果都与输送机主同步，以生成一个虚拟主用于最终轴的 MC_GearInPos 功能，以执行该运动。



另一个应用案例可以是带装饰的巧克力棒（模具中的单个棒）。添加剂使波浪与输送机同步，并为下一个波浪返回。

这个例子的具体应用可以是一个机器，用来在冰淇淋基础上方放置波浪形的冰淇淋层，这些基础在冰淇淋工厂的冷冻线中移动。加料轴必须以波浪形式与承载冰淇淋基础块的输送机同步。并且必须在特定的起始位置和波浪相位下完成，以实现预期的结果（因此是 GearInPos）。使用 CAM FB，可以轻松定义不同的波浪模式（例如，在冰淇淋顶部较长的波浪）。

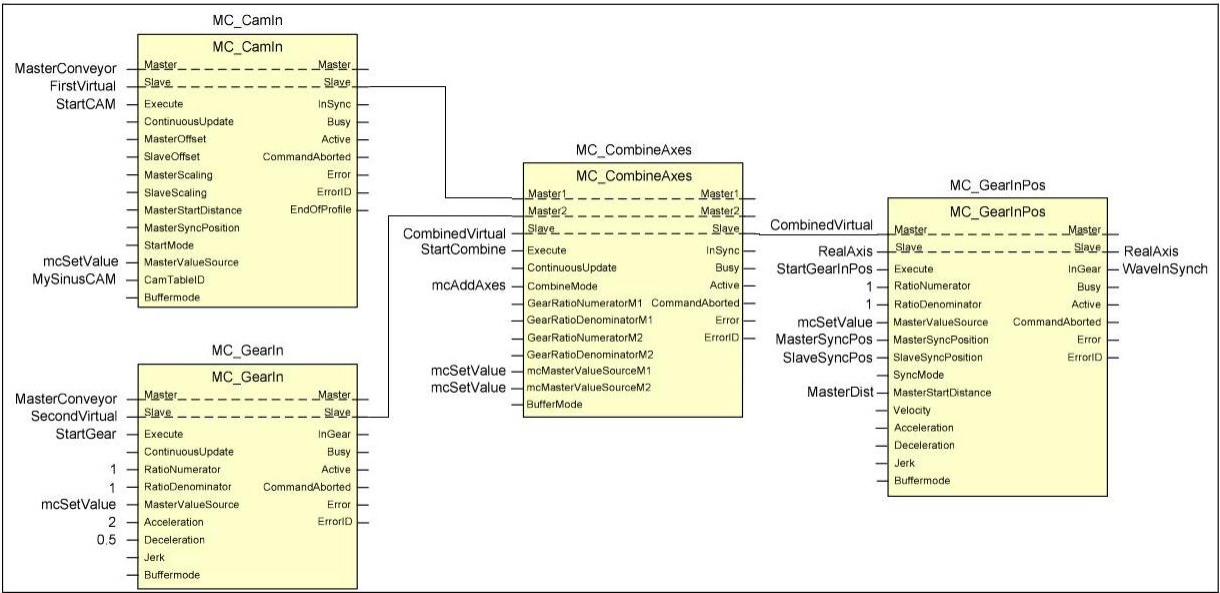


图 50: MC_CombineAxes 应用示例

5. MC FB 的应用 - 一个“中止”与“混合”的钻孔示例

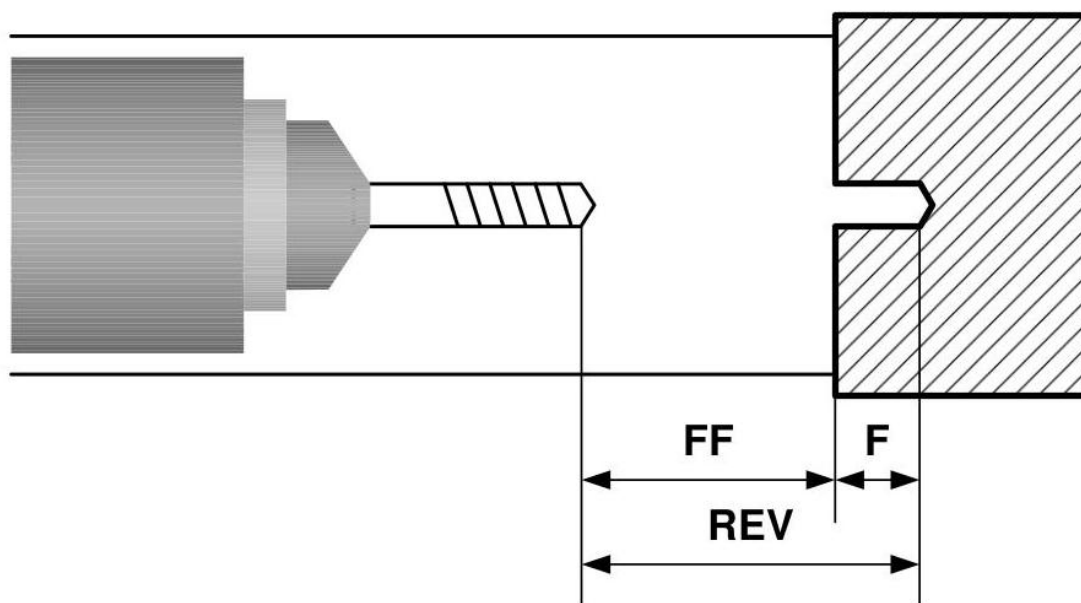


图 52：简单钻孔单元的示例

这个简单的钻孔单元的示例显示了两种模式之间的区别。

为了钻孔，必须完成以下步骤：

步骤 1：初始化，例如在上电时。步骤 2：向钻孔位置移动并开始旋转钻头。通过这种方式，在到达位置之前它将完全操作，然后检查这两个动作是否完成。

步骤 3：钻孔。

步骤 4：在钻完孔后，我们必须等待步进链序列完成停留，以清除孔中可能卡住的任何碎屑。

步骤 5：将钻头移回起始位置并关闭主轴。完成后退和停止主轴的操作，向步进链发出重新开始的信号。

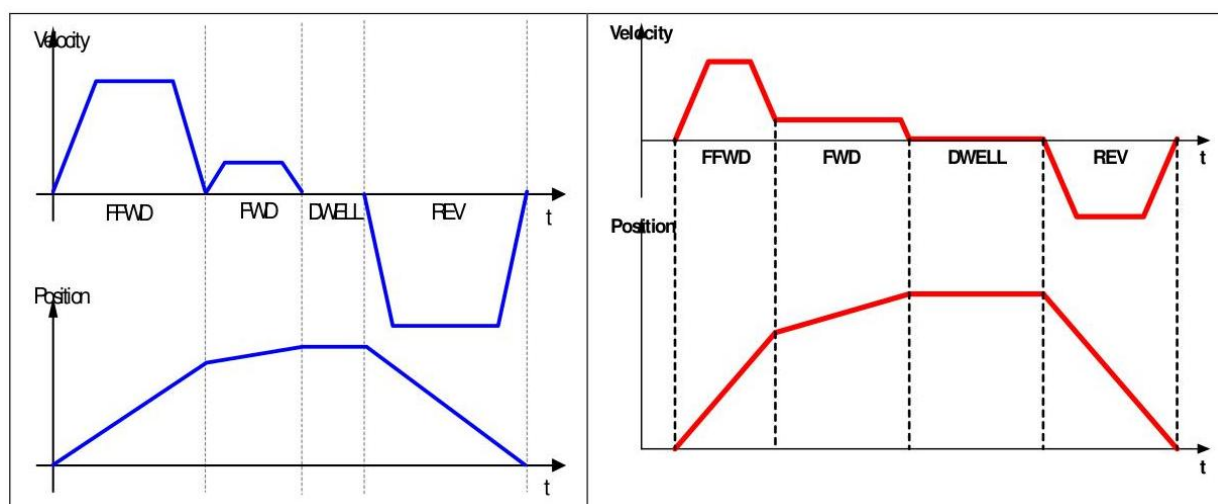


图 53：钻孔的时序图。左侧为不混合，右侧为混合

5.1. Solution with Function Block diagram

这两个例子可以用相同的 FBD 程序来描述。不同之处在于第二个功能块 MC_MoveRelative 中 'BufferMode' 的输入。本例中显示的模式是 'Aborting' 或 'BlendingLow'。

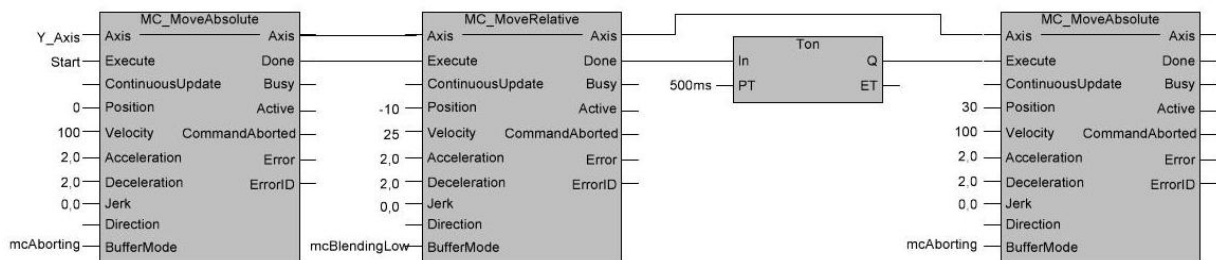


图 54：使用功能块图的解决方案

5.2. Solution with Sequential Function Chart

这是使用顺序功能图（SFC）指定顺序步骤的经典方法。SFC 实现了上述示例中给出的时序图。

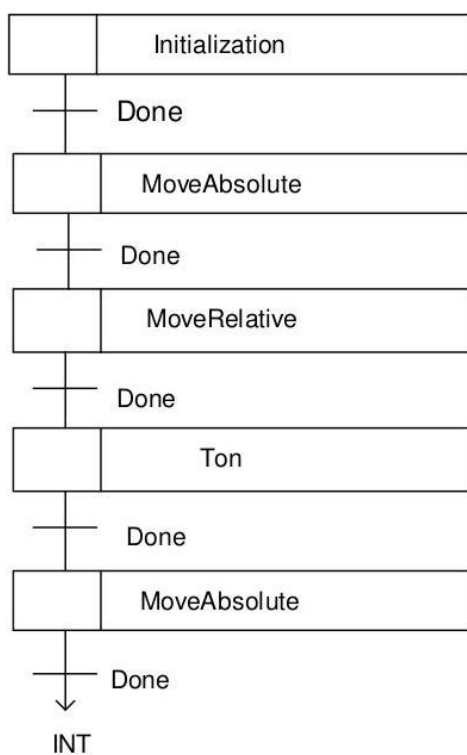


图 55：用于 SFC 中钻孔示例的直接步骤-转换链

Appendix A. Examples of the different buffer modes

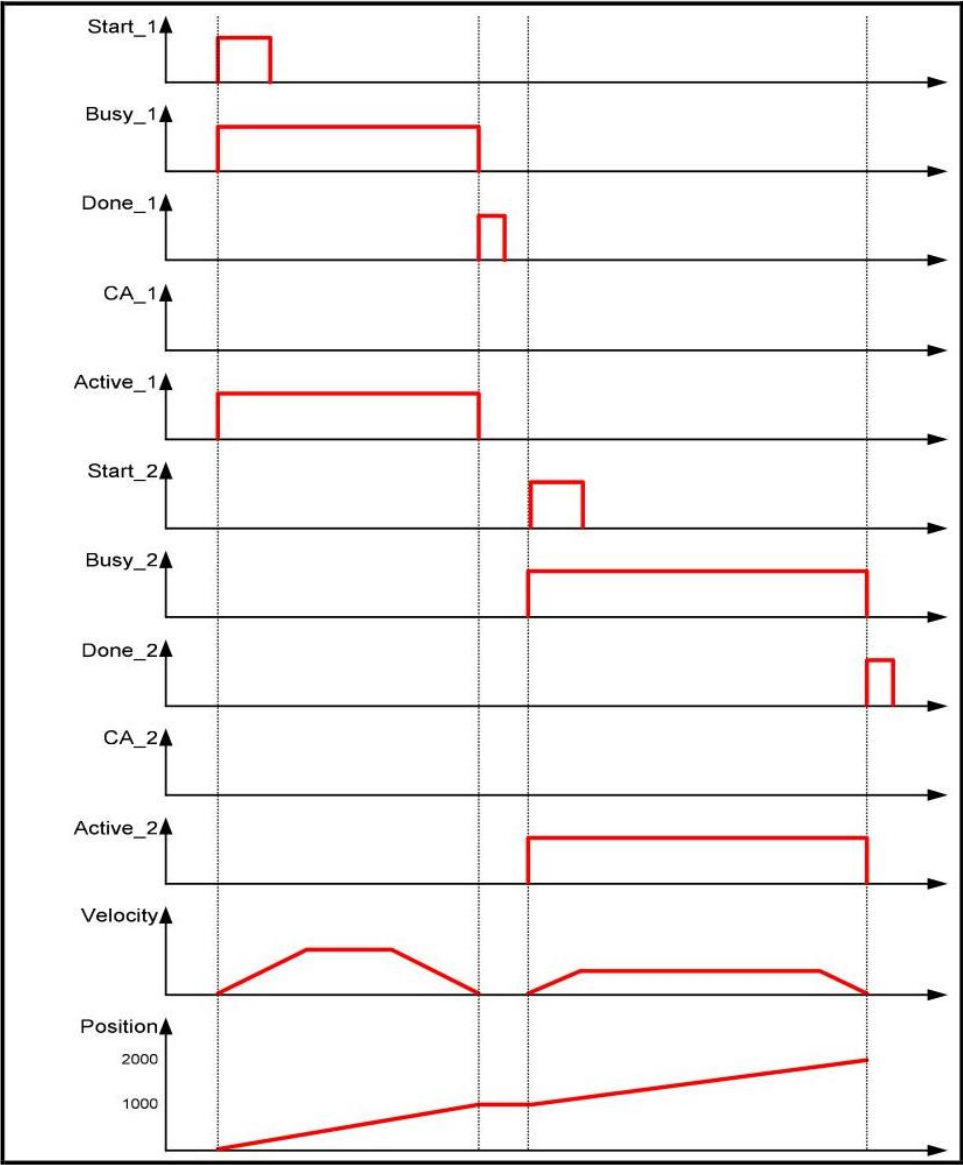


图 57: 上述示例的时序图，FB1 和 FB2 之间没有干扰（“中止” 模式）
示例 2: “中止” 动作

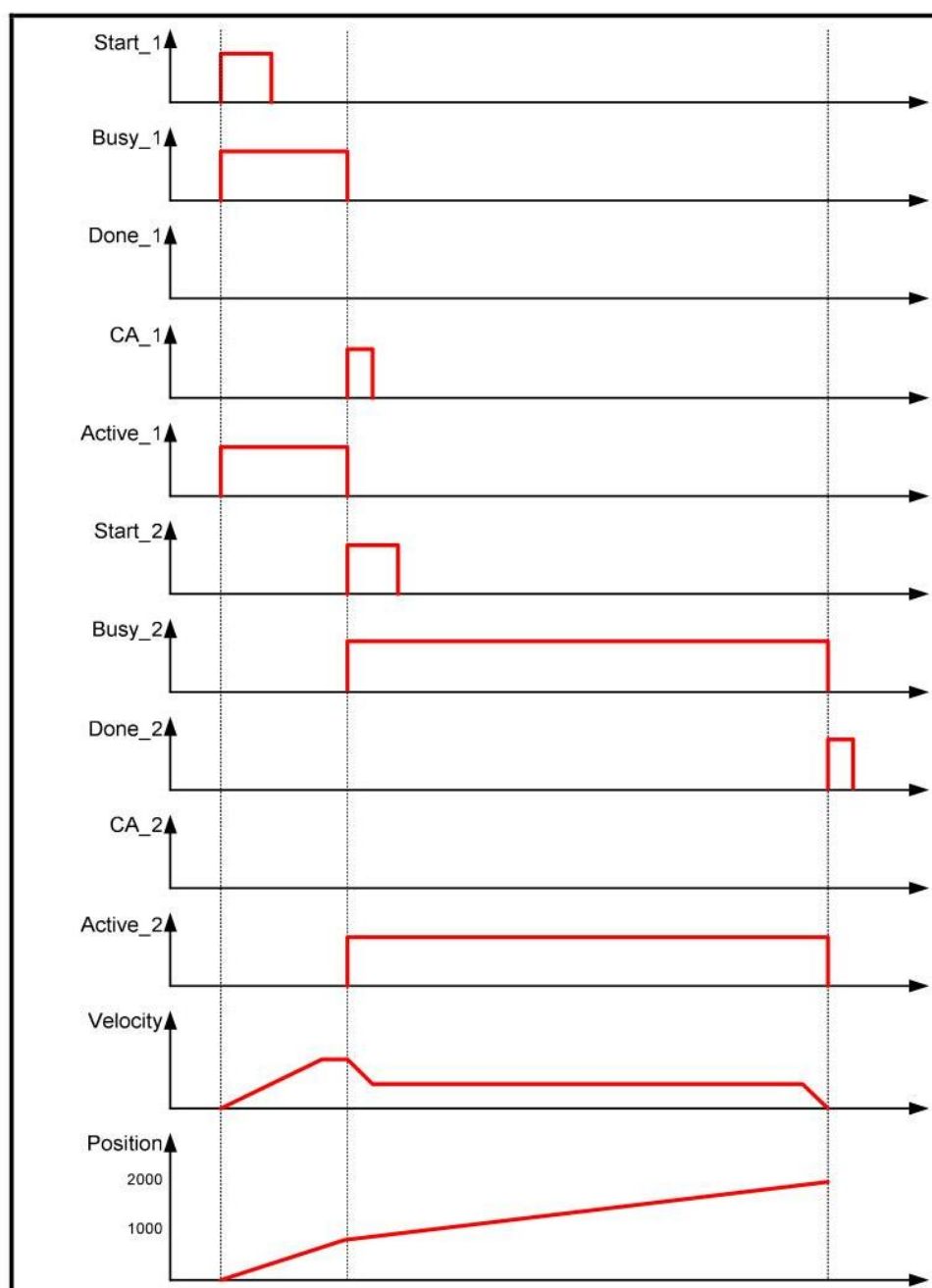
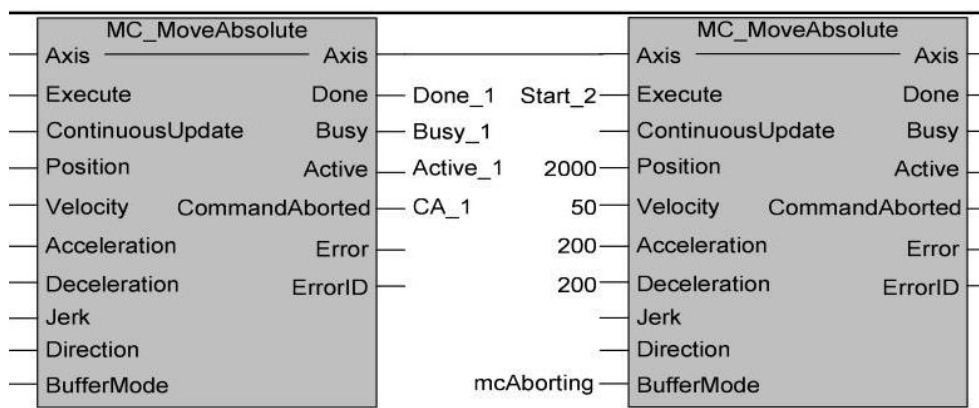


图 58: 上述示例的时序图, FB2 中断 FB1 (“中止” 模式)
 示例 3: “缓冲” 动作

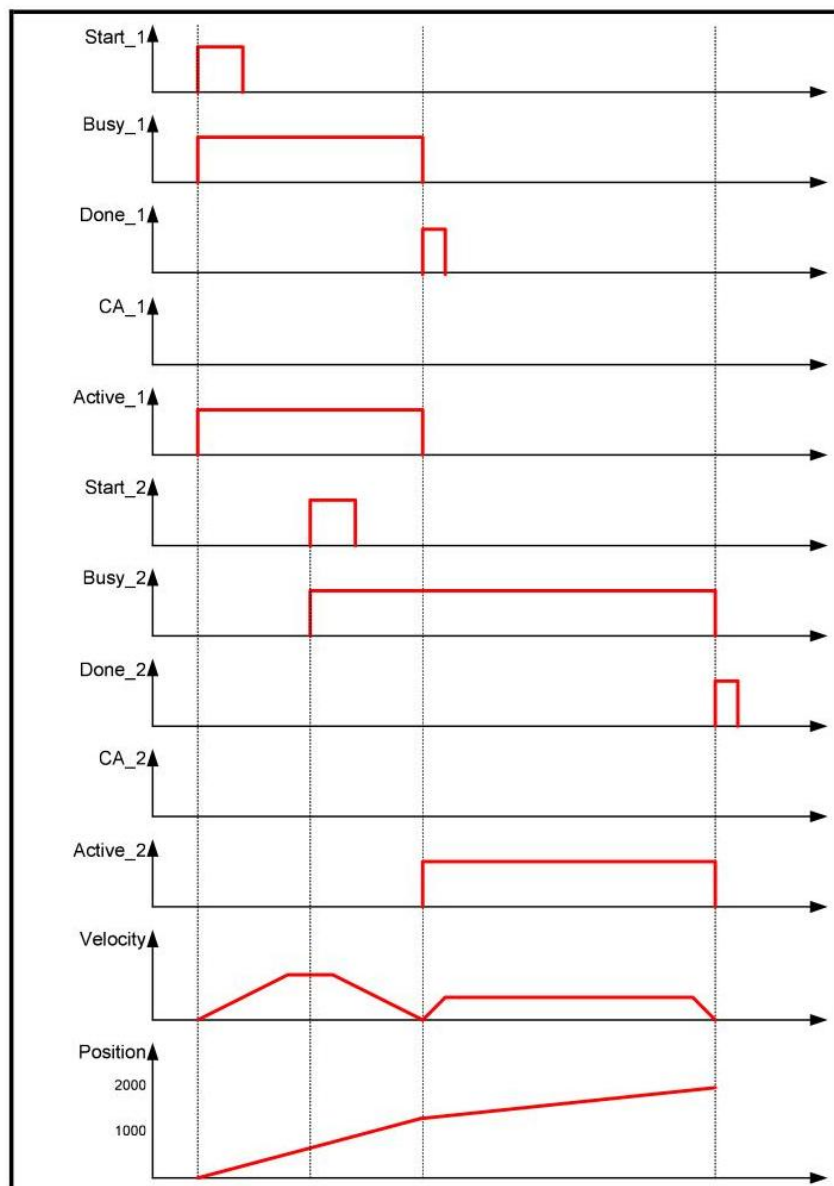
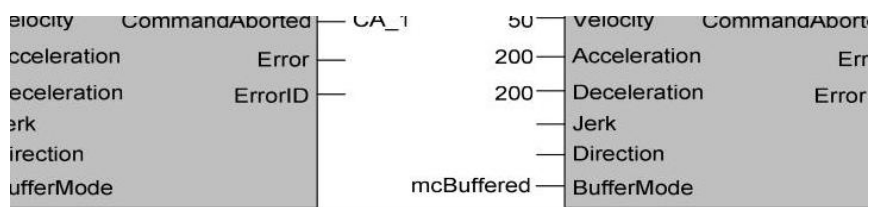


图 59: 上述示例在“缓冲”模式下的时序图
(停止至速度 0, 并在该点开始 FB2, 无延迟)
示例 4: “混合低速”动作

Axis	Axis	Axis	Axis
Execute	Done_1	Start_2	Execute
ContinuousUpdate	Busy_1		ContinuousUpdate
Position	Active_1	2000	Position
Velocity	CommandAborted	CA_1	Velocity
Acceleration	Error	50	Acceleration
Deceleration	ErrorID	50	Deceleration
			Jerk
			Direction
BufferMode	mcBlendingLow	BufferMode	mcBlendingLow

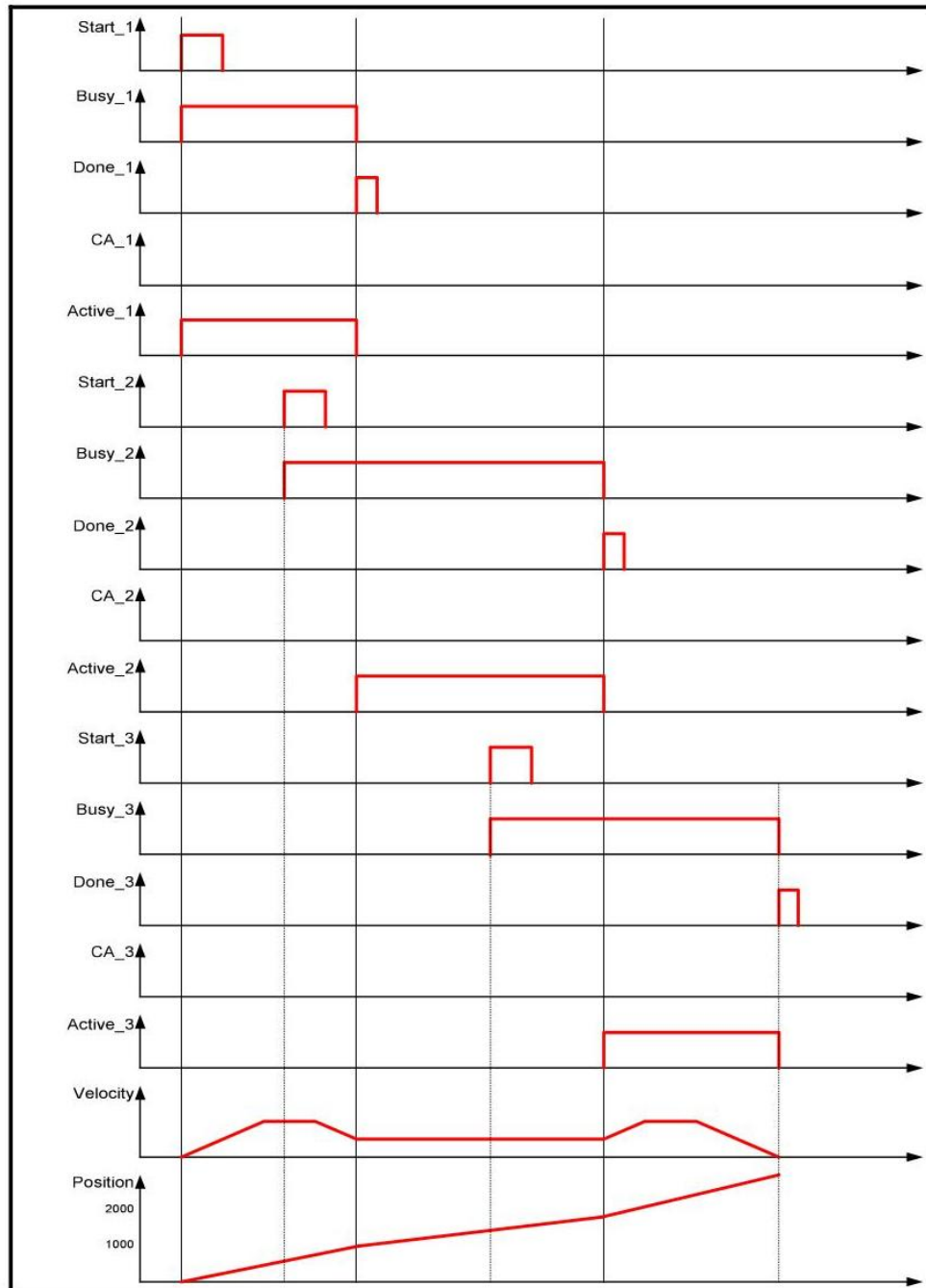


图 60: 上述示例在“混合低速”模式下的时序图

(从 FB1 的最终位置到 FB2 的最终位置使用最低速度 (= 速度 2)) 在混合 (和其他在同一轴上同时工作的功能块, 如 MC_MoveAdditive) 时, 系统必须在将位置传递给相关轴之前合并在该轴上工作的不同值。(C) 1999 - 2011 版权所有由 PLCopen 页 114/141

示例 5: “混合前一个” 动作

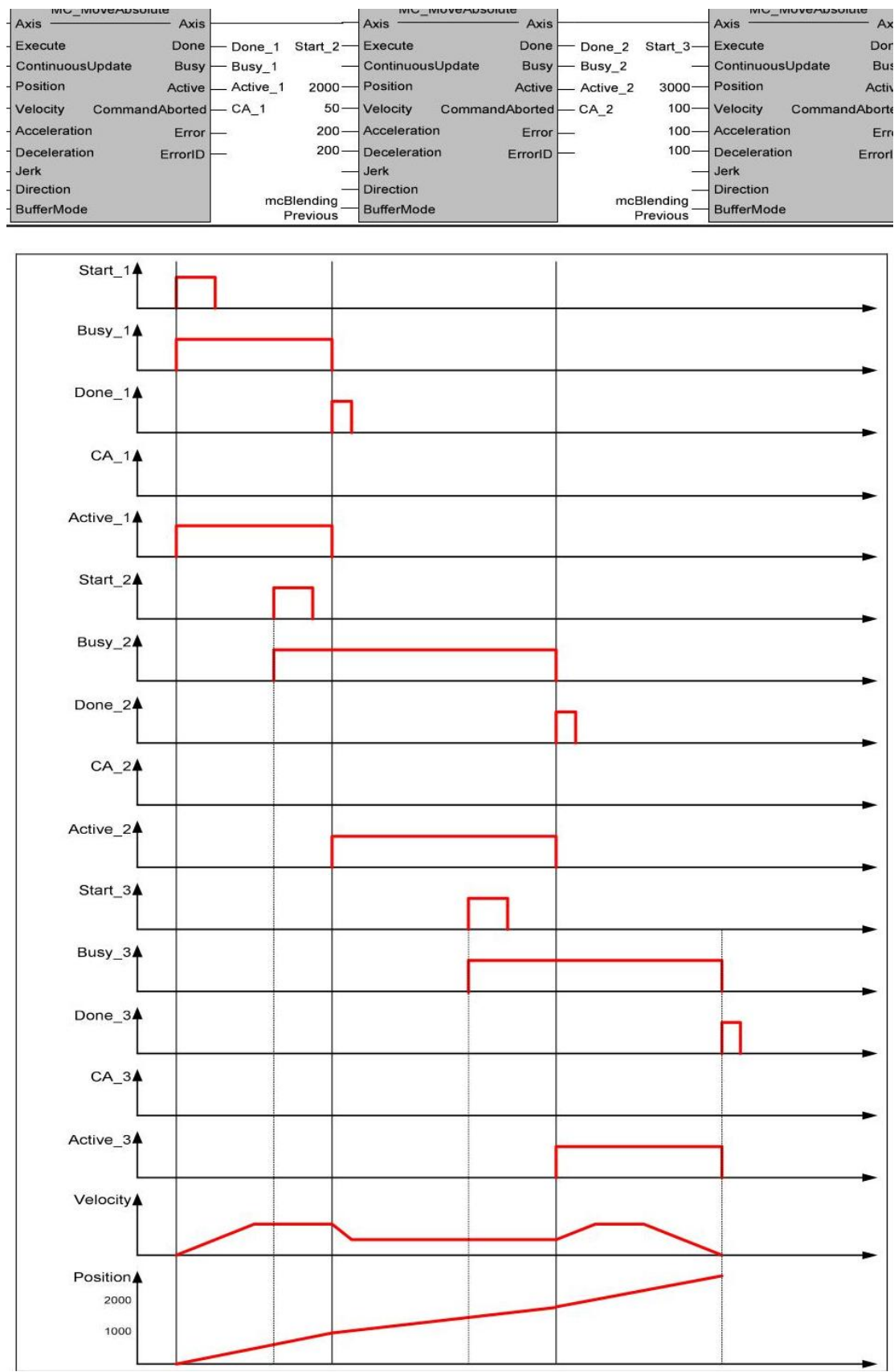


图 61: 带有模式'Merging1' 的上述示例的时序图

(在最终位置 FB1 处使用速度 FB1)
 示例 6: 'BlendingNext' 运动

Axis	Done	Start_2	Execute	Done	Start_3	Execute
ContinuousUpdate	Busy_1	ContinuousUpdate	Busy_2	ContinuousUpdate		
Active	Active_1	Position	Active	Active_2	Position	
CommandAborted	CA_1	Velocity	CommandAborted	CA_2	Velocity	Co
Error		Acceleration	Error		Acceleration	
ErrorID		Deceleration	ErrorID		Deceleration	
		Jerk			Jerk	
		Direction			Direction	
mcBlendingNext		BufferMode	mcBlendingNext		BufferMode	

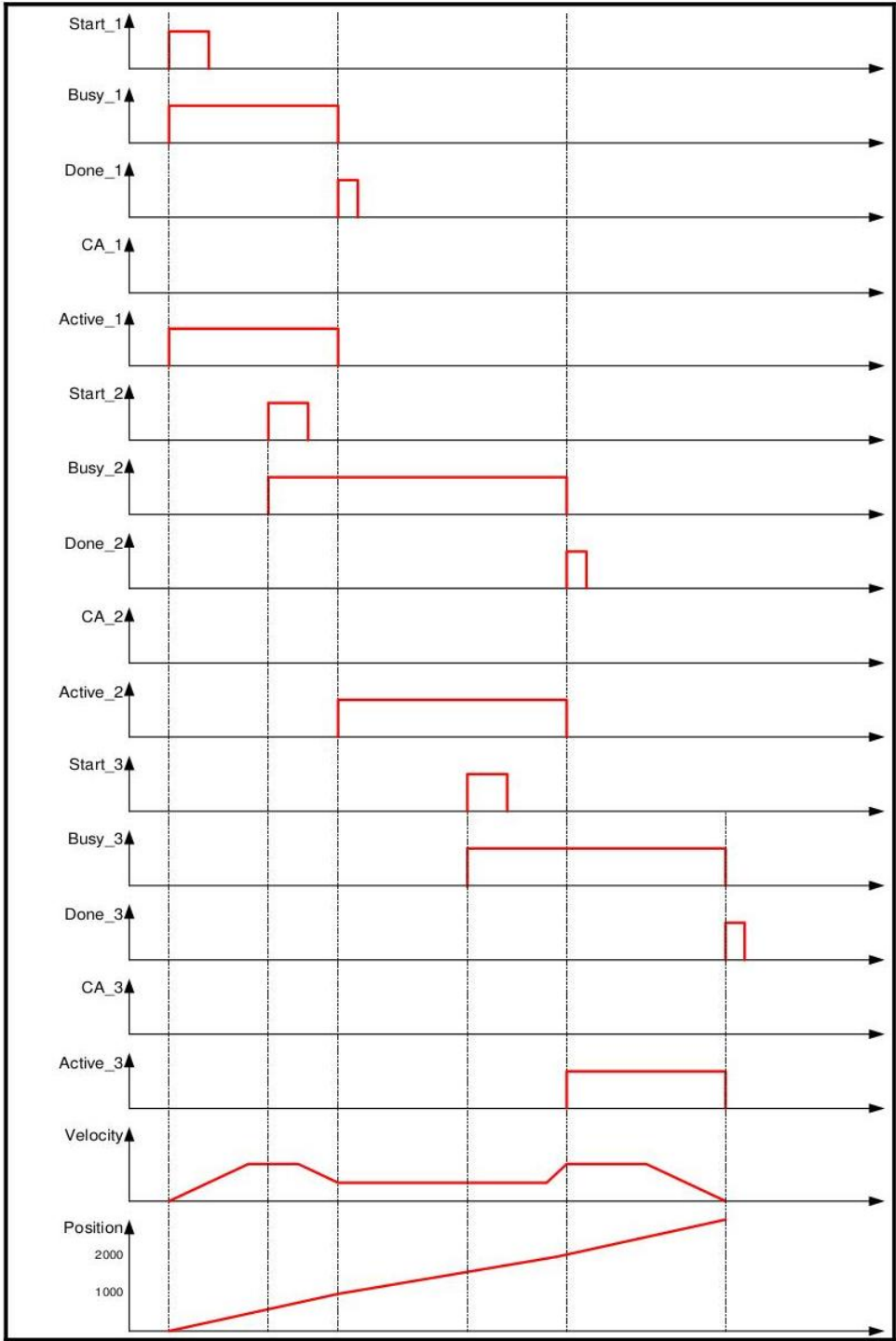


图 62: 带有模式'BlendingNext' 运动的上述示例的时序图

在跟随 MC_MoveVelocity 的 2nd FB 的情况下，所有混合模式都应该像混合之前的那样工作或产生错误。

示例 7: 'BlendingHigh' 运动

Done	Done_1	Start_2	Execute	Done	Done_2	Start_3	Execute
Busy	Busy_1		ContinuousUpdate	Busy	Busy_2		Continu
Active	Active_1	2000	Position	Active	Active_2	3000	Position
CommandAborted	CA_1	50	Velocity	CommandAborted	CA_2	100	Velocity
Error		200	Acceleration	Error		100	Acceler
ErrorID		200	Deceleration	ErrorID		100	Decele
			Jerk				Jerk
			Direction				Directic
	mcBlendingHigh		BufferMode		mcBlendingHigh		BufferM

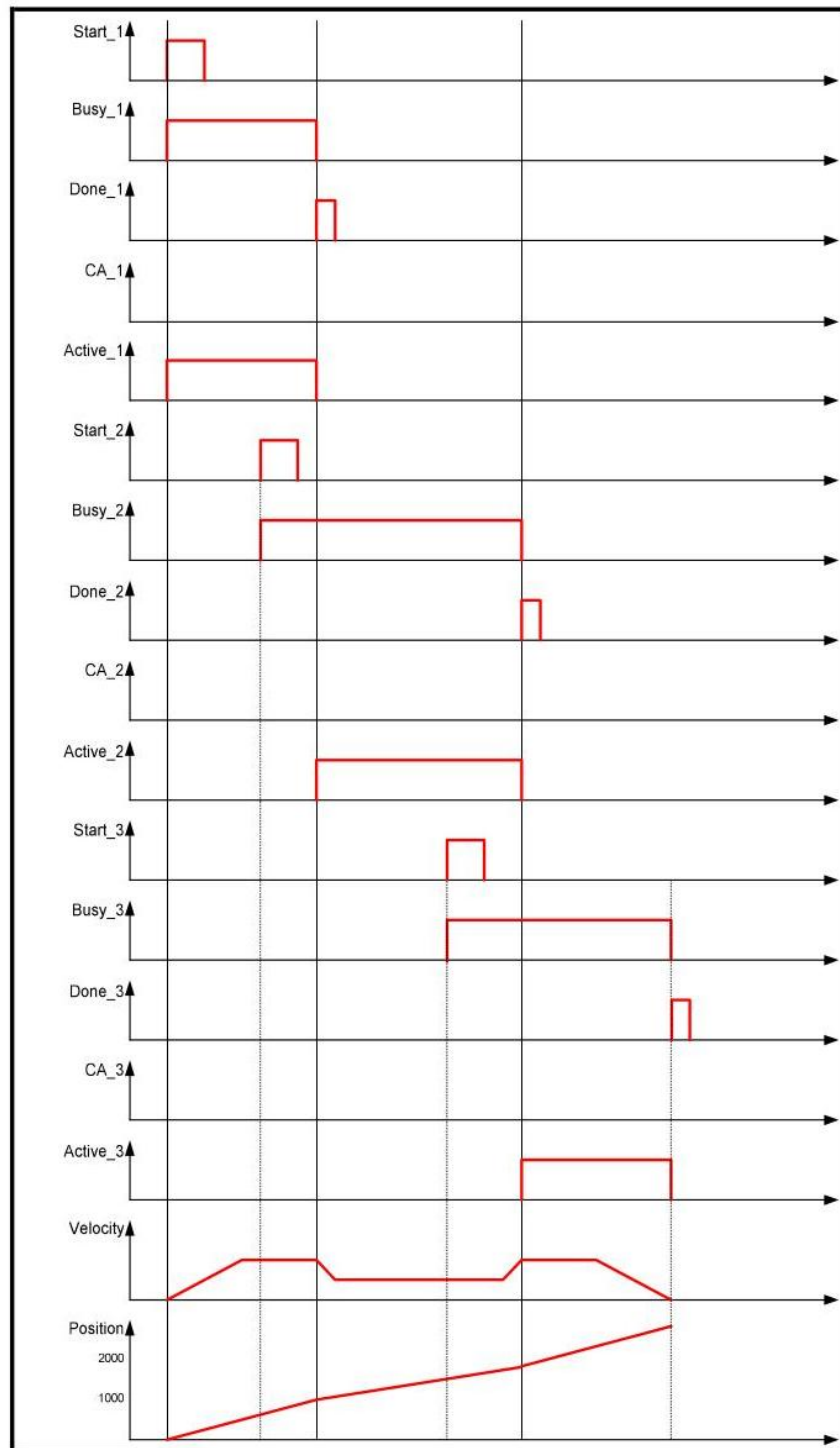


图 63: 带有模式'BlendingHigh' 运动的上述示例的时序图

Appendix B. Compliance Procedure and Compliance List

在本附录中列出了运动控制功能块供应商的合规声明要求。合规声明主要由两组组成：支持的数据类型和支持的功能块，以及适用的输入和输出。供应商需要根据其产品填写所使用的数据类型和功能块的表格，承诺支持该规范。

通过向 PLCopen 提交这些表格，并在 PLCopen 批准后，该列表将在 PLCopen 网站 (www.plcopen.org) 上发布，以及如附录 B 2 所述的支持的数据类型和附录 B 3 所述的功能块的简要概述。

除了此批准外，供应商还获得了 PLCopen 运动控制徽标的访问和使用权，如附录 B 4 所描述的：PLCopen 运动控制徽标及其使用。



Data types

在功能块和参数中列出的数据类型 REAL（例如速度、加速度、距离等）可以被更改为 SINT、INT、DINT 或 LREAL，而不被视为不合规，只要它们在整个功能块和参数集中的一致性得以保持。

实现允许扩展数据类型，只要保持基本数据类型。例如：WORD 可以更改为 DWORD，但不能更改为 REAL。

Function Blocks and Inputs and Outputs

一个声称符合此 PLCopen 规范的实现应提供一组用于运动控制的功能块，即一个或多个功能块，至少包含表中标记为“B”的基本输入和输出变量。这些输入和输出必须得到支持，以被视为合规。

对于更高层次的系统和未来的扩展，可以实现表中标记为“E”的扩展输入和输出变量的任意子集。

供应商特定的附加项标记为“V”，并可以在供应商文档中列出。

- 基本输入/输出变量是强制性的
- 扩展输入/输出变量是可选的
- 供应商特定的附加项

在表中标记为字母“B”

在表中标记为字母“E”

在供应商合规性文档中标记为“V”

所有供应商特定的项将不会在 PLCopen 网站上的比较表中列出，而是在详细的供应商特定列表中发布。

所有功能块的所有供应商特定输入和输出必须列在供应商的认证列表中。通过这种方式，供应商的认证列表描述了相关功能块的所有输入/输出，包括供应商特定的扩展，从而展示了供应商实际使用的完整功能块。

TC2 任务组运动控制 (C) 1999-2011 版权所有 PLCopen 第 118/141 页

附录 B 1. 供应商声明

Supplier name
Supplier address
City
Country
Telephone
Fax
Email address
Product Name
Product version
Release date

我在此声明，以下填写并提交的表格与我们的产品及随附的用户手册一致，如上所述。

代表姓名 (个人):

签名日期 (dd/mm/yyyy):

签名:

附录 B 2. 支持的数据类型

Defined datatypes with MC library:	Supported	If not supported, which datatype used
BOOL		
INT		
WORD		
REAL		
ENUM		
UINT		

表 6: 支持的数据类型

在规范中定义了以下派生数据类型。定义这些结构中哪些被使用在此

system.			
Derived datatypes:	Where used	Supported	Which structure
AXIS_REF	Nearly all FBs		
MC_DIRECTION (extended)	MC_MoveAbsolute MC_MoveVelocity MC_TorqueControl MC_MoveContinuousAbsolute		
MC_TP_REF	MC_PositionProfile		
MC_TV_REF	MC_VelocityProfile		
MC_TA_REF	MC_AccelerationProfile		
MC_CAM_REF	MC_CamTableSelect		
MC_CAM_ID (extended)	MC_CamTableSelect MC_CamIn		
MC_START_MODE (extended)	MC_CamIn MC_CamTableSelect		
MC_BUFFER_MODE	Buffered FBs		
MC_EXECUTION_MODE	MC_SetPosition MC_WriteParameter MC_WriteBoolParameter MC_WriteDigitalOutput MC_CamTableSelect		
MC_SOURCE	MC_ReadMotionState MC_CamIn MC_GearIn MC_GearInPos MC_CombineAxes MC_DigitalCamSwitch		
MC_SYNC_MODE	MC_GearInPos		
MC_COMBINE_MODE	MC_CombineAxes		
MC_TRIGGER_REF	MC_TouchProbe MC_AbortTrigger		
MC_INPUT_REF	MC_ReadDigitalInput		
MC_OUTPUT_REF	MC_DigitalCamSwitch MC_ReadDigitalOutput MC_WriteDigitalOutput		
MC_CAMSWITCH_REF	MC_DigitalCamSwitch		
MC_TRACK_REF	MC_DigitalCamSwitch		

表 7: 支持的派生数据类型
附录 B 3. 功能块概述

Single Axis Function Blocks	Supported as V1.0/ V1.1/ V2.0 or Not	Comments (<= 48 char.)
MC_Power		
MC_Home		
MC_Stop		
MC_Halt		
MC MoveAbsolute		
MC MoveRelative		
MC MoveAdditive		
MC_MoveSuperimposed		
MC_HaltSuperimposed		
MC_MoveVelocity		
MC_MoveContinuousAbsolute		
MC_MoveContinuousRelative		
MC_TorqueControl		
MC PositionProfile		
MC_VelocityProfile		
MC_AccelerationProfile		
MC SetPosition		
MC_SetOverride		
MC_ReadParameter & MC ReadBoolParameter		
MC_WriteParameter & MC WriteBoolParameter		
MC_ReadDigitalInput		
MC_ReadDigitalOutput		
MC_WriteDigitalOutput		
MC ReadActualPosition		
MC ReadActualVelocity		
MC_ReadActualTorque		
MC ReadStatus		
MC ReadMotionState		
MC ReadAxisInfo		
MC ReadAxisError		
MC_Reset		
MC_DigitalCamSwitch		
MC_TouchProbe		
MC_AbortTrigger		
Multi-Axis Function Blocks	Supported as V1.0/ V1.1/ V2.0 or Not	Comments (<= 48 char.)
MC CamTableSelect		
MC_CamIn		
MC_CamOut		
MC_GearIn		
MC_GearOut		
MC GearInPos		
MC_PhasingAbsolute		
MC_PhasingRelative		
MC CombineAxes		

表 8: 功能块的简要概述
附录 B 3.2 MC_Home

If Supported	MC_Home	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
VAR INPUT			
B	Execute		
B	Position		
E	BufferMode		
VAR OUTPUT			
B	Done		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

附录 B 3.3 MC_Stop

If Supported	MC_Stop	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
VAR_INPUT			
B	Execute		
E	Deceleration		
E	Jerk		
VAR OUTPUT			
B	Done		
E	Busy		
E	CommandAborted		
B	Error		
E	ErrorID		

附录 B 3.4 MC_Halt

If Supported	MC_Halt	Sup. Y/N	
VAR_IN_OUT			
B	Axis		
VAR_INPUT			
B	Execute		
E	Deceleration		
E	Jerk		
E	BufferMode		
VAR OUTPUT			
B	Done		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

附录 B 3.5 MC_MoveAbsolute

If Supported	MC_MoveAbsolute	Sup.Y/N	Comments
VAR_IN_OUT			
B	Axis		
VAR_INPUT			
B	Execute		
E	ContinuousUpdate		
B	Position		
B	Velocity		
E	Acceleration		
E	Deceleration		
E	Jerk		
B	Direction		
E	BufferMode		
VAR OUTPUT			
B	Done		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

附录 B 3.6 MC_MoveRelative

If Supported	MC_MoveRelative	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
VAR INPUT			
B	Execute		
E	ContinuousUpdate		
B	Distance		
E	Velocity		
E	Acceleration		
E	Deceleration		
E	Jerk		
E	BufferMode		
VAR OUTPUT			
B	Done		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

附录 B 3.7 MC_MoveAdditive

If Supported	MC_MoveAdditive	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
VAR INPUT			
B	Execute		
E	ContinuousUpdate		
B	Distance		
E	Velocity		
E	Acceleration		
E	Deceleration		
E	Jerk		
E	BufferMode		
VAR OUTPUT			
B	Done		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

附录 B 3.8 MC_MoveSuperimposed

If Supported	MC_MoveSuperimposed	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
VAR_INPUT			
B	Execute		
E	ContinuousUpdate		
B	Distance		
E	VelocityDiff		
E	Acceleration		
E	Deceleration		
E	Jerk		
VAR OUTPUT			
B	Done		
E	Busy		
E	CommandAborted		
B	Error		
E	ErrorID		
E	CoveredDistance		

附录 B 3.11 MC_MoveContinuousAbsolute

If Supported	MC_MoveContinuousAbsolute	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
VAR_INPUT			
B	Execute		
E	ContinuousUpdate		
B	Position		
B	EndVelocity		
B	Velocity		
E	Acceleration		
E	Deceleration		
E	Jerk		
E	Direction		
E	BufferMode		
VAR OUTPUT			
B	InEndVelocity		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

附录 B 3.12 MC_MoveContinuousRelative

If Supported	MC_MoveContinuousRelative	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
VAR_INPUT			
B	Execute		
E	ContinuousUpdate		
B	Distance		
B	EndVelocity		
B	Velocity		
E	Acceleration		
E	Deceleration		
E	Jerk		
E	BufferMode		
VAR OUTPUT			
B	InEndVelocity		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

附录 B 3.14 MC_PositionProfile

If Supported	MC_PositionProfile	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
B	TimePosition		
VAR_INPUT			
B	Execute		
E	ContinuousUpdate		
E	TimeScale		
E	PositionScale		
E	Offset		
E	BufferMode		
VAR OUTPUT			
B	Done		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

附录 B 3.15 MC_速度剖面

If Supported	MC_VelocityProfile	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
B	TimeVelocity		
VAR INPUT			
B	Execute		
E	ContinuousUpdate		
E	TimeScale		
E	VelocityScale		
E	Offset		
E	BufferMode		
VAR OUTPUT			
B	ProfileCompleted		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

附录 B 3.16 MC_ 加速度轮廓

If Supported	MC_AccelerationProfile	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
B	TimeAcceleration		
VAR_INPUT			
B	Execute		
E	ContinuousUpdate		
E	TimeScale		
E	AccelerationScale		
E	Offset		
E	BufferMode		
VAR OUTPUT			
B	ProfileCompleted		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

附录 B 3.17 MC_SetPosition

If Supported	MC_SetPosition	Sup.Y/N	Comments
VAR_IN_OUT			
B	Axis		
VAR_INPUT			
B	Execute		
B	Position		
E	Relative		
E	ExecutionMode		
VAR OUTPUT			
B	Done		
E	Busy		
B	Error		
E	ErrorID		

附录 B 3.19 MC_ReadParameter & MC_ReadBoolParameter

If Supported	MC_ReadParameter	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
VAR INPUT			
B	Enable		
B	ParameterNumber		
VAR OUTPUT			
B	Valid		
E	Busy		
B	Error		
E	ErrorID		
B	Value		

附录 B 3.20 MC_WriteParameter & MC_WriteBoolParameter

If Supported	MC_WriteParameter	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
VAR INPUT			
B	Execute		
B	ParameterNumber		
B	Value		
E	ExecutionMode		
VAR OUTPUT			
B	Done		
E	Busy		
B	Error		
E	ErrorID		

附录 B 3.22 MC_ReadDigitalOutput

If Supported	MC_ReadDigitalOutput	Sup.Y/N	Comments
VAR_IN_OUT			
B	Output		
VAR INPUT			
B	Enable		
E	OutputNumber		
VAR OUTPUT			
B	Valid		
E	Busy		
B	Error		
E	ErrorID		
B	Value		

附录 B 3.23 MC_WriteDigitalOutput

If Supported	MC_WriteDigitalOutput	Sup.Y/N	Comments
VAR_IN_OUT			
B	Output		
VAR_INPUT			
B	Execute		
E	OutputNumber		
B	Value		
E	ExecutionMode		
VAR OUTPUT			
B	Done		
E	Busy		
B	Error		
E	ErrorID		

附录 B 3.24 MC_ReadActualPosition

If Supported	MC_ReadActualPosition	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
VAR INPUT			
B	Enable		
VAR OUTPUT			
B	Valid		
E	Busy		
B	Error		
E	ErrorID		
B	Position		

附录 B 3.26 MC_ReadActualTorque

If Supported	MC_ReadActualTorque	Sup.Y/N	Comments
VAR_IN_OUT			
B	Axis		
VAR INPUT			
B	Enable		
VAR_OUTPUT			
B	Valid		
E	Busy		
B	Error		
E	ErrorID		
B	Torque		

附录 B 3.27 MC_ReadStatus

If Supported	MC_ReadStatus	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
VAR INPUT			
B	Enable		
VAR OUTPUT			
B	Valid		
E	Busy		
B	Error		
E	ErrorID		
B	ErrorStop		
B	Disabled		
B	Stopping		
E	Homing		
B	Standstill		
E	DiscreteMotion		
E	ContinuousMotion		
E	SynchronizedMotion		

附录 B 3.28 MC_ReadMotionState

If Supported	MC_ReadMotionState	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
VAR INPUT			
B	Enable		
E	Source		
VAR OUTPUT			
B	Valid		
E	Busy		
B	Error		
E	ErrorID		
E	ConstantVelocity		
E	Accelerating		
E	Decelerating		
E	DirectionPositive		
E	DirectionNegative		

附录 B 3.29 MC_ReadAxisInfo

If Supported	MC_ReadAxisInfo	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
VAR INPUT			
B	Enable		
VAR OUTPUT			
B	Valid		
E	Busy		
B	Error		
E	ErrorID		
E	HomeAbsSwitch		
E	LimitSwitchPos		
E	LimitSwitchNeg		
E	Simulation		
E	CommunicationReady		
E	ReadyForPowerOn		
E	PowerOn		
E	IsHomed		
E	AxisWarning		

附录 B 3.31 MC_Reset

If Supported	MC_Reset	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
VAR INPUT			
B	Execute		
VAR OUTPUT			
B	Done		
E	Busy		
B	Error		
E	ErrorID		

附录 B 3.32 MC_DigitalCamSwitch

If Supported	MC_DigitalCamSwitch	Sup.Y/N	Comments
VAR_IN_OUT			
B	Axis		
B	Switches		
E	Outputs		
E	TrackOptions		
VAR INPUT			
B	Enable		
E	EnableMask		
E	ValueSource		
VAR OUTPUT			
B	InOperation		
E	Busy		
B	Error		
E	ErrorID		

MC_CAMSWITCH_REF 数组结

构中的基本元素

B/E	Parameter	Sup.Y/N	Comments
B	TrackNumber		
B	FirstOnPosition [u]		
B	LastOnPosition [u]		
E	AxisDirection		
E	CamSwitchMode		
E	Duration		

附录 B 3.33 MC_TouchProbe

If Supported	MC_TouchProbe	Sup.Y/N	Comments
VAR_IN_OUT			
B	Axis		
E	TriggerInput		
VAR INPUT			
B	Execute		
E	WindowOnly		
E	FirstPosition		
E	LastPosition		
VAR OUTPUT			
B	Done		
E	Busy		
E	CommandAborted		
B	Error		
E	ErrorID		
B	RecordedPosition		

附录 B 3.34 MC_AbortTrigger

If Supported	MC_AbortTrigger	Sup.Y/N	Comments
VAR_IN_OUT			
B	Axis		
E	TriggerInput		
VAR INPUT			
B	Execute		
VAR_OUTPUT			
B	Done		
E	Busy		
B	Error		
E	ErrorID		

附录 B 3.36 MC_CamIn

If Supported	MC_CamIn	Sup. Y/N	Comments
VAR_IN_OUT			
B	Master		
B	Slave		
VAR_INPUT			
B	Execute		
E	ContinuousUpdate		
E	MasterOffset		
E	SlaveOffset		
E	MasterScaling		
E	SlaveScaling		
E	MasterStartDistance		
E	MasterSyncPosition		
E	StartMode		
E	MasterValueSource		
E	CamTableID		
E	BufferMode		
VAR OUTPUT			
B	InSync		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		
E	EndOfProfile		

附录 B 3.38 MC_GearIn

If Supported	MC_GearIn	Sup. Y/N	Comments
VAR_IN_OUT			
B	Master		
B	Slave		
VAR_INPUT			
B	Execute		
E	ContinuousUpdate		
B	RatioNumerator		
B	RatioDenominator		
E	MasterValueSource		
E	Acceleration		
E	Deceleration		
E	Jerk		
E	BufferMode		
VAR OUTPUT			
B	InGear		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

附录 B 3.40 MC_GearInPos

If Supported	MC_GearInPos	Sup.Y/N	Comments
VAR_IN_OUT			
B	Master		
B	Slave		
VAR INPUT			
B	Execute		
B	RatioNumerator		
B	RatioDenominator		
E	MasterValueSource		
B	MasterSyncPosition		
B	SlaveSyncPosition		
E	SyncMode		
E	MasterStartDistance		
E	Velocity		
E	Acceleration		
E	Deceleration		
E	Jerk		
E	BufferMode		
VAR OUTPUT			
E	StartSync		
B	InSync		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

附录 B 3.42 MC_PhasingRelative

If Supported	MC_PhasingRelative	Sup. Y/N	Comments
VAR_IN_OUT			
B	Master		
B	Slave		
VAR_INPUT			
B	Execute		
B	PhaseShift		
E	Velocity		
E	Acceleration		
E	Deceleration		
E	Jerk		
E	BufferMode		
VAR OUTPUT			
B	Done		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		
E	CoveredPhaseShift		

Appendix B 4. The PLCopen Motion Control Logo and Its Usage

为了快速识别符合要求的产品，PLCopen 开发了运动控制功能模块的标志：



图 64: PLCopen 运动控制标志

该运动控制标志归 PLCopen 所有并注册为商标。

为了免费使用此标志，相关公司必须满足以下所有要求：

1. 公司必须是 PLCopen 的投票成员；
2. 公司必须遵守现有规范，如 PLCopen 运动控制工作组所规定的规范，及由 PLCopen 发布的规范，其中本声明是其一部分；
3. 公司需以书面形式向 PLCopen 提交该符合性申请，明确说明适用的软件包及所有指定表格的支持元素，详见文件本身；
4. 如果不符合要求，由 PLCopen 决定，公司将收到 PLCopen 关于此事的书面声明。公司将有一个月的时间要么调整他们的软件包，使其符合要求，并提交新的符合声明，要么从所有技术或宣传材料中去除所有对该规范的引用，包括使用该标志；
5. 该标志必须按原样使用——即完整标志。可以调整大小，但必须保持原始比例和颜色设置。
6. 该标志必须在运动控制的背景下使用。

警告：该 PDF 由 302.AI 调用大语言模型一键生成，版权归原文作者所有。翻译内容可靠性无保障，请仔细鉴别并以原文为准。当前大语言模型: gpt-4o-mini-2024-07-18 为了防止大语言模型的意外谬误产生扩散影响，禁止移除或修改此警告。