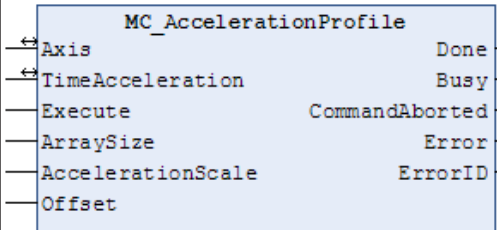


常用 MC 指令详解

1 单轴指令

MC\_AccelerationProfile

1) 指令格式

指令	名称	图形表现	ST 表现
MC_AccelerationProfile	加速度轮廓指令		<pre>MC_AccelerationProfile(   Axis:= ,   TimeAcceleration:= ,   Execute:= ,   ArraySize:= ,   AccelerationScale:= ,   Offset:= ,   Done=&gt; ,   Busy=&gt; ,   CommandAborted=&gt; ,   Error=&gt; ,   ErrorID=&gt; );</pre>

2) 相关变量

◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴，即 AXIS_REF_SM3 的一个实例
TimeAcceleration	轴加速时间和加速度描述	MC_TA_REF			轴加速时间和加速度数据描述，加速数据由多组数据组成

◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
Execute	执行条件	BOOL	TRUE,FALSE	FALSE	输入的一个上升沿将启动功能块的处理
ArraySize	动态数组	INT	数据范围	0	运行轮廓中使用的数组个数
AccelerationScale	综合因子	LREAL	“正数”+”0”	1	MC_TA_REF 中加速度或减速度的比例因子
Offset	偏移	LREAL		0	加减速度的总体偏移值

◆ 输出变量

输出变量	名称	数据类型	有效范围	初始值	描述
Done	指令执行完成	BOOL	TRUE,FALSE	FALSE	轴指令执行完成，置为 TRUE
Busy	指令正在执行	BOOL	TRUE,FALSE	FALSE	当前指令正在执行中，置为 TRUE
CommandAbort	指令被中断	BOOL	TRUE,FALSE	FALSE	当前指令被中断，置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时，置为 TRUE
ErrorID	错误代码	SMC_ERROR	参阅 SMC_ERROR	0	异常发生时，输出错误代码

3) 功能说明

本功能块为时间段和加减速度的轮廓运动模型，运行模式为 Discrete Motion, 按用户在 TimeAcceleration 变量中设定的数据运行。

本功能块运行状态为 Standstill 中，指令运行时的状态为 Discrete Motion, 其他状态无法运行。

启动指令为 Execute 的上升沿启动，本指令在 Discrete Motion 重复运行速度都是在上一轮的叠加，容易引起系统故障。

TimeAcceleration 为 MC\_TA\_REF 数据类型；

MC\_TA\_REF 具体描述如下：

成员	类型	初始值	描述
Number_of_pairs	INT	0	轮廓路径的段数
IsAbsolute	BOOL	TRUE	绝对运动 (TRUE) 和相对运动选择
MC_TA_Array	ARRAY[1..N] OF SMC_TA		时间和加速值的数组

SMC\_TA 具体描述如下：

成员	类型	初始值	描述
delta_time	TIME	TIME#0ms	加速度段的时间
acceleration	LREAL	0	当前的加速度值

注：设置的加速度体现在速度的变化上，所有的加速度变化按 S 曲线的方式变化，从最终的结果变化为 [ 起始加速度为 A。

终止加速度为 B] (A+B)/2 的加速度数据体现在最终的速度上；

4) 时序图

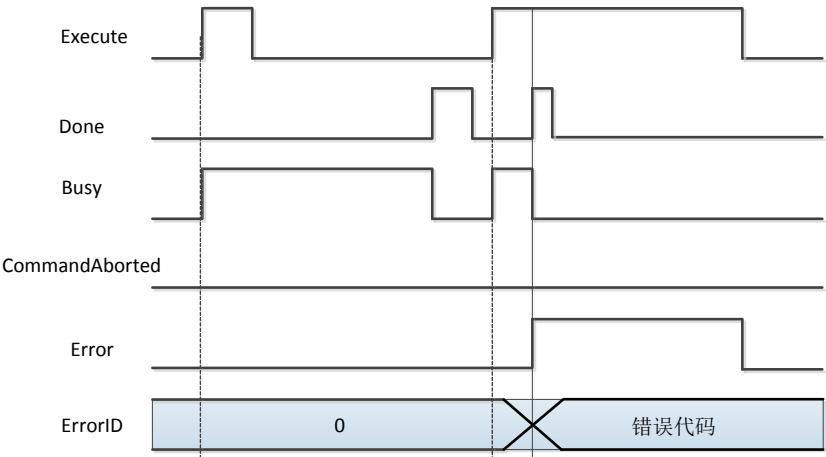
条件 MC\_TA\_Array 通过其他方式已经设置；

轴必须处于 Standstill 状态指令才能运行；

功能块的 Execute 必须有上升沿的条件；

功能块的 Done 表示指令正常执行完成；

功能块的 Busy 表示当前功能块正在执行中；



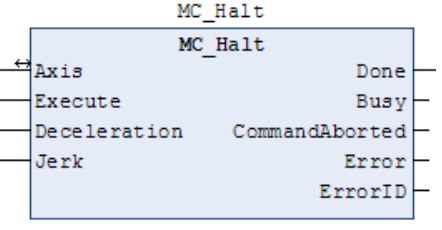
5) 错误说明

错误的出现为轴状态不是在 Standstill 中启动指令或指令系统中的参数错误，出现轴错误只能清除错误后才开始运行。

【注意】：请阅读“附录 C 错误代码说明”以了解相关错误代码说明。

## MC\_Halt

### 1) 指令格式

指令	名称	图形表现	ST 表现
MC_Halt	轴正常停止命令		<pre>MC_Halt(     Axis:= ,     Execute:= ,     Deceleration:= ,     Jerk:= ,     Done=&gt; ,     Busy=&gt; ,     CommandAborted=&gt; ,     Error=&gt; ,     ErrorID=&gt; );</pre>

### 2) 相关变量

#### ◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴，即 AXIS_REF_SM3 的一个实例

#### ◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
Execute	执行条件	BOOL	TRUE,FALSE	FALSE	输入的一个上升沿将启动功能块的处理
Deceleration	减速度	LREAL	“正数”+“0”	0	功能块的减速度 (u/S^2)
Jerk	跃度	LREAL	“正数”+“0”	0	指定跃度 [指令单位 /S^3]

#### ◆ 输出变量

输出变量	名称	数据类型	有效范围	初始值	描述
Done	指令执行完成	BOOL	TRUE,FALSE	FALSE	轴指令执行完成，置为 TRUE
Busy	指令正在执行	BOOL	TRUE,FALSE	FALSE	当前指令正在执行中，置为 TRUE
CommandAborted	指令被中断	BOOL	TRUE,FALSE	FALSE	当前指令被中断，置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时，置为 TRUE
ErrorID	错误代码	SMC_ERROR	参阅 SMC_ERROR	0	异常发生时，输出错误代码

### 3) 功能说明

本功能块为在正常运行的情况下停止一个轴的运动，而其它一个运动轴指令再运行时可以终止本指令的运行。

本功能块运行状态为运行状态 (Motion) 才能运行，其他状态无法运行。

启动指令为 Execute 的上升沿启动；

指令运行中的状态为 Discrete Motion，运行完成后状态为 Standstill。

### 4) 时序图

轴必须处于运行状态 (Motion) 指令才能运行；

功能块的 Execute 必须有上升沿的条件；

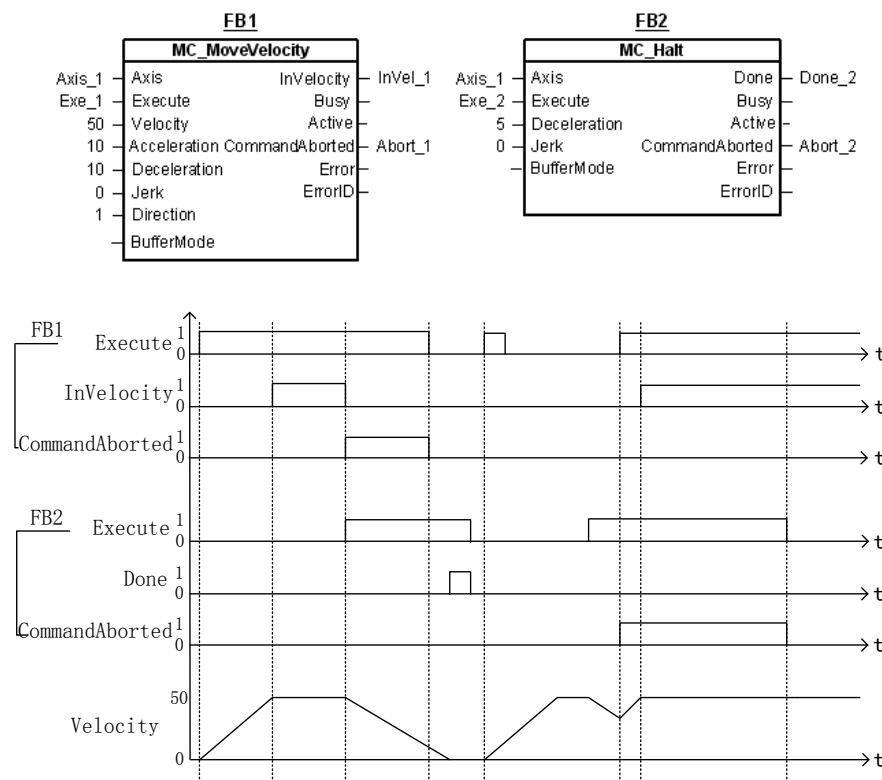
功能块的 Done 表示指令正常执行完成；

功能块的 Busy 表示当前功能块正在执行中；

功能块的 **CommandAborted** 表示指令被其他运动控制指令中断，此时标志位为 **TRUE**；

例程：在执行 **MC\_MoveVelocity** 指令和 **MC\_Halt** 指令在不同的时序操作中对应的标志位的变化；

对 **CommandAborted** 的处理描述如下图的时序描述。



5) 错误说明

错误的出现为轴状态不是在 **Standstill** 中启动指令或指令系统中的参数错误，出现轴错误只能清除错误后才开始运行。

【注意】：请阅读 [“附录 C 错误代码说明”](#) 以了解相关错误代码说明。

## MC\_Home

### 1) 指令格式

指令	名称	图形表现	ST 表现
MC_Home	轴回零指令		<pre> MC_Home (     Axis:= Axis,     Execute:= ,     Position:= ,     Done=&gt; ,     Busy=&gt; ,     CommandAborted=&gt; ,     Error=&gt; ,     ErrorID=&gt; ); </pre>

### 2) 相关变量

#### ◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴，即 AXIS_REF_SM3 的一个实例

#### ◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
Execute	执行条件	BOOL	TRUE,FALSE	FALSE	输入的一个上升沿将启动功能块的处理
Position	轴到达位置	LREAL	数据范围	0	代表轴位置的回零位置

#### ◆ 输出变量

输出变量	名称	数据类型	有效范围	初始值	描述
Done	指令执行完成	BOOL	TRUE,FALSE	FALSE	轴指令执行完成，置为 TRUE
Busy	指令正在执行	BOOL	TRUE,FALSE	FALSE	当前指令正在执行中，置为 TRUE
CommandAbort	指令被中断	BOOL	TRUE,FALSE	FALSE	当前指令被中断，置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时，置为 TRUE
ErrorID	错误代码	SMC_ERROR	参阅 SMC_ERROR	0	异常发生时，输出错误代码

### 3) 功能说明

本功能块为回零操作，Position 数据位轴的零点位置。

本功能块运行状态为 Standstill 中，指令运行时的状态为 homing, 其他状态无法运行。

启动指令为 Execute 的上升沿启动指令。

汇川伺服设置说明：

- ◆ 在使用每根伺服轴在零点回归时必须设定伺服参数的回归模式；设置模式可以手工设置伺服的功能码；
- ◆ 通过 AM600 从站的启动参数也可以配置相应的功能码；采用通讯的方式必须设置如下索引和子索引的数据；

项目	索引	子索引	描述
回零方式	0x6098		根据伺服手册可以选择设定的具体参数
找原点速度	0x6099	0x01	一般定义的速度相对比较高，减少归零时间
找零点速度	0x6099	0x02	一般定义的速度相对较低
原点回归加减速	0x609A		在原点回归时的加减速变化
原点回归超时时间	0x2005	0x24	回归时间超过设定时间，系统报 Err.601 错误

AM600 的画面设置参考：

从站

专家过程数据

过程数据(PDO)

启动参数

在线

在线CoE

功能码列表

EtherCAT I/O映射

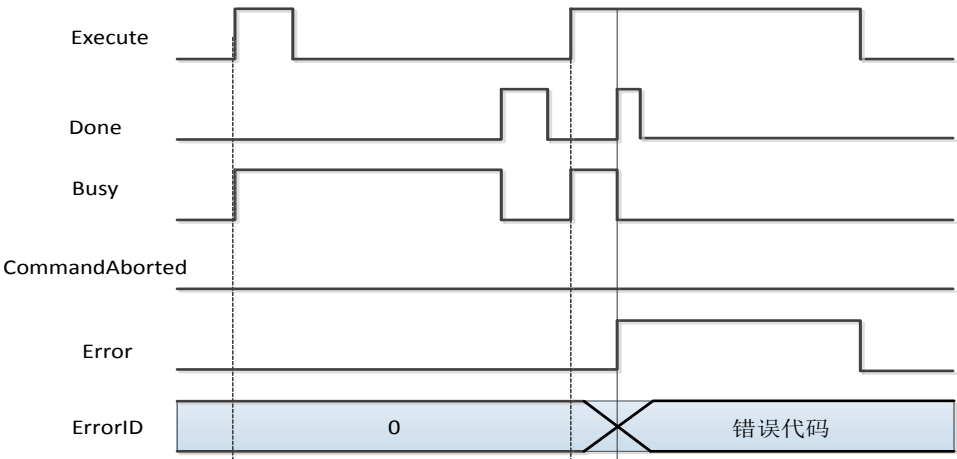
状态

信息

Add Edit Delete Move Up Move Down

行	索引：子索引	名称	值	位长度	如果有错，则退出	如果有错，则跳行	下一行	注释
1	16#6098:16#00	Homing method	1	8	<input type="checkbox"/>	<input type="checkbox"/>	0	

4) 时序图



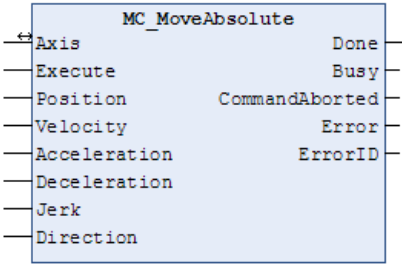
## MC\_MoveAbsolute

轴按绝对位置运行 ( 单位按轴设置 ), 绝对位置由 **Position** 指定; 本指令运行前设置好相关的参数, 加速度 (Acceleration)、减

速度 (Deceleration)、运行速度 (Velocity) 和加减速模式的跃度 (Jerk); 对加速度 (Acceleration) 或减速度 (Deceleration) 的赋值为 0

指令运行错误; 在运行过程中, 一定要关注本指令运行的完整过程, 从用户程序的设计角度避免其他指令中断此指令。

### 1) 指令格式

指令	名称	图形表现	ST 表现
MC_MoveAbsolute	轴绝对位置控制指令		<pre>MC_MoveAbsolute(     Axis:= ,     Execute:= ,     Position:= ,     Velocity:= ,     Acceleration:= ,     Deceleration:= ,     Jerk:= ,     Direction:= ,     Done=&gt; ,     Busy=&gt; ,     CommandAborted=&gt; ,     Error=&gt; ,     ErrorID=&gt; );</pre>

### 2) 相关变量

#### ◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴, 即 AXIS_REF_SM3 的一个实例

#### ◆ 输入变量

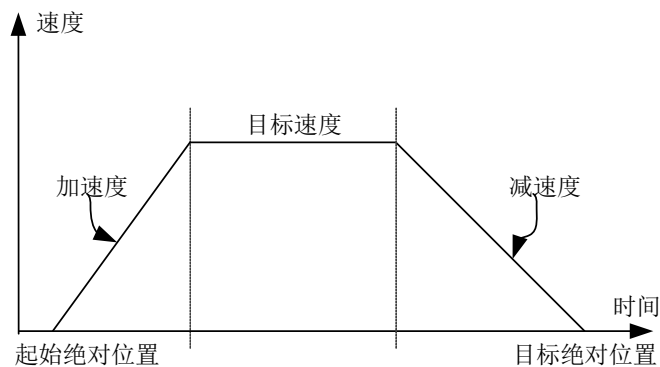
输入变量	名称	数据类型	有效范围	初始值	描述
Execute	执行条件	BOOL	TRUE,FALSE	FALSE	输入的一个上升沿将启动功能块的处理
Position	轴到达位置	LREAL	数据范围	0	此位置为轴的绝对位置数据
Velocity	运行速度	LREAL	数据范围	0	轴运行到目标位置的最大速度
Acceleration	加速度	LREAL	数据范围	0	速度变大时加速度值
Deceleration	减速度	LREAL	数据范围	0	速度变小时减速度值
Jerk	跃度	LREAL	数据范围	0	曲线加减速的斜率变化值
Direction	指令极性	MC_DIRECTION	Negative,shortest Positive, current,fastest	shortest	Negative: 反向移动; Shortest: 根据最短路径选择方向; Positive: 正向移动; Current: 按当前方向移动; Fastest: 自动选择最快的方向移动; ( 此功能轴在旋转模式下有效 )

#### ◆ 输出变量

输出变量	名称	数据类型	有效范围	初始值	描述
Done	指令执行完成	BOOL	TRUE,FALSE	FALSE	轴指令执行完成, 置为 TRUE
Busy	指令正在执行	BOOL	TRUE,FALSE	FALSE	当前指令正在执行中, 置为 TRUE
CommandAbort	指令被中断	BOOL	TRUE,FALSE	FALSE	当前指令被中断, 置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时, 置为 TRUE
ErrorID	错误代码	SMC_ERROR	参阅 SMC_ERROR	0	异常发生时, 输出错误代码

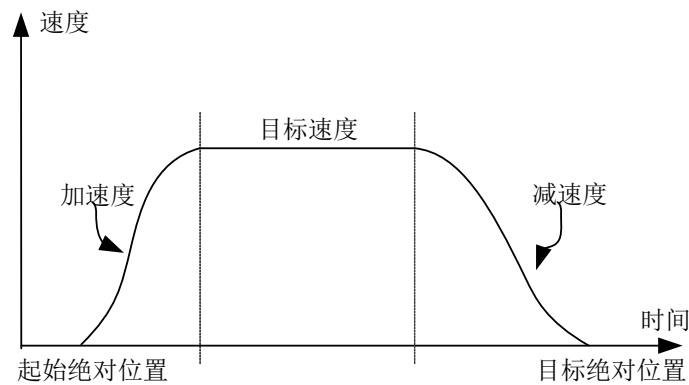
3) 功能说明

- ◆ 本功能块为轴绝对定位指令，Position 数据为轴的绝对位置。
- ◆ 本功能块运行状态为 Standstill 中，指令运行时的状态为 Discrete Motion, 一个完整的运行过程一定要控制轴的不同运动状态。
- ◆ 启动指令为 Execute 的上升沿启动，本指令在 Discrete Motion 可以重复上升沿有效，每次都可以刷新最新的 Position 位置。
- ◆ Acceleration 或 Deceleration 为零，指令运行都为异常状态，但轴的状态为 Discrete Motion；
- ◆ 梯形加减速动作  
Velocity、Acceleration 和 Deceleration 有数据；而 Jerk 为 0；



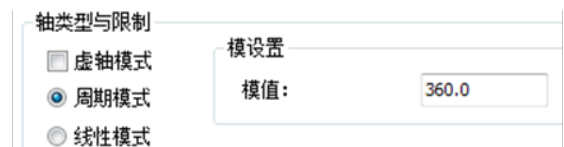
- ◆ S 曲线加减速动作

Velocity、Acceleration、Deceleration 和 Jerk 有数据；



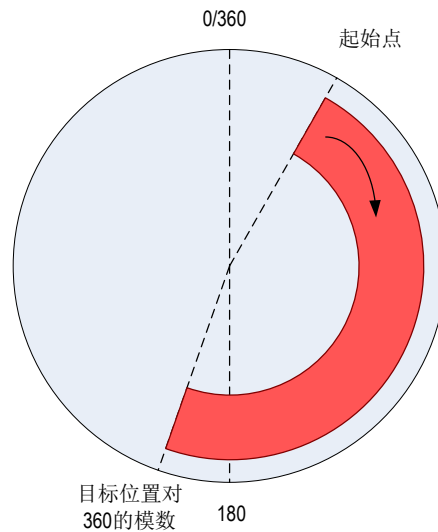
- ◆ 轴在周期模式下绝对定位

- ① 轴旋转周期设置为 360、Direction 设置为 Positive。



当 Position 对 360 的模值 (Position/360 取余，比如 Position 为 380 则对 360 的模值为 20，Position 为 350 则对 360 模值为 350) > 起始绝对位置时，此时轴朝正向运行 Position 对 360 的模值 - 起始绝对位置的距离。

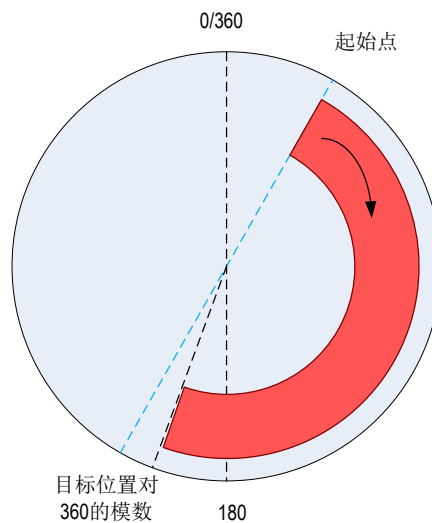




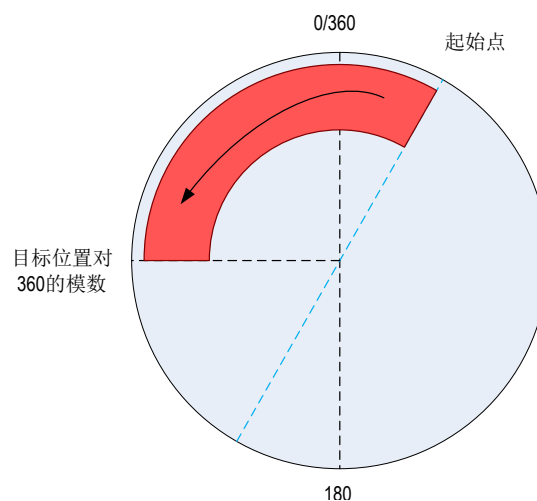
当  $\text{Position}$  对 360 的模值 ( $\text{Position}/360$  取余, 比如  $\text{Position}$  为 380 则对 360 的模值为 20)  $<$  起始绝对位置时, 此时轴朝正向运行  $360 - \text{起始绝对位置} + \text{Position}$  对 360 的模值的距离。

- ② 轴旋转周期设置为 360、Direction 设置为 shortest 或者 fastest。Position 对 360 的模值为 XPosition

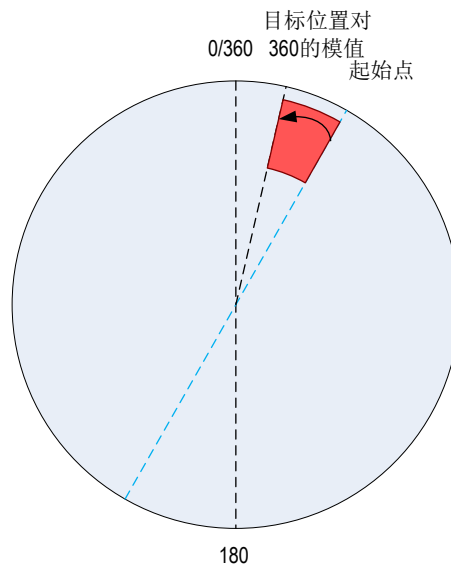
当  $0 < \text{XPosition} - \text{起始绝对位置} < 180$  时, 轴朝正向运行  $\text{XPosition} - \text{起始绝对位置}$  的距离。



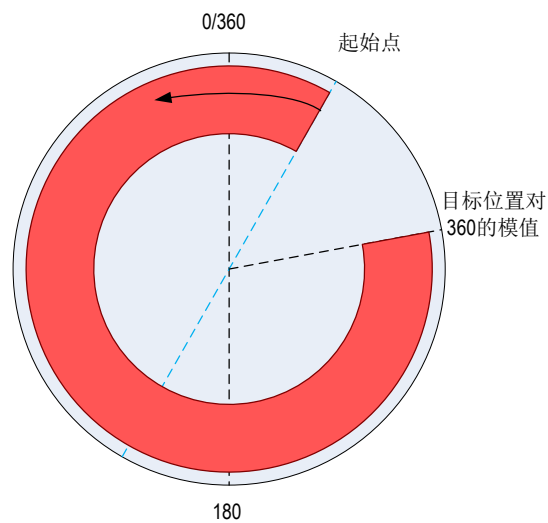
当  $180 < \text{XPosition} - \text{起始绝对位置}$  时, 轴朝反向运行  $360 - \text{XPosition} + \text{起始绝对位置}$  的距离。



当  $\text{XPosition} < \text{起始绝对位置}$  时, 轴朝反向运行  $\text{起始绝对位置} - \text{XPosition}$  的距离。

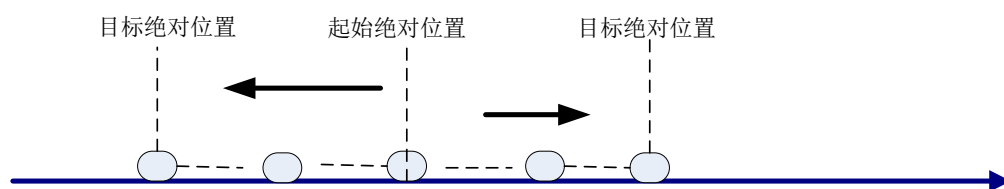


- ③ 轴旋转周期设置为 360、Direction 设置为 **shortest** 或者 **Negative**。Position 对 360 的模值为 XPosition 轴朝反向运行起始绝对位置 +360-XPosition 的距离。



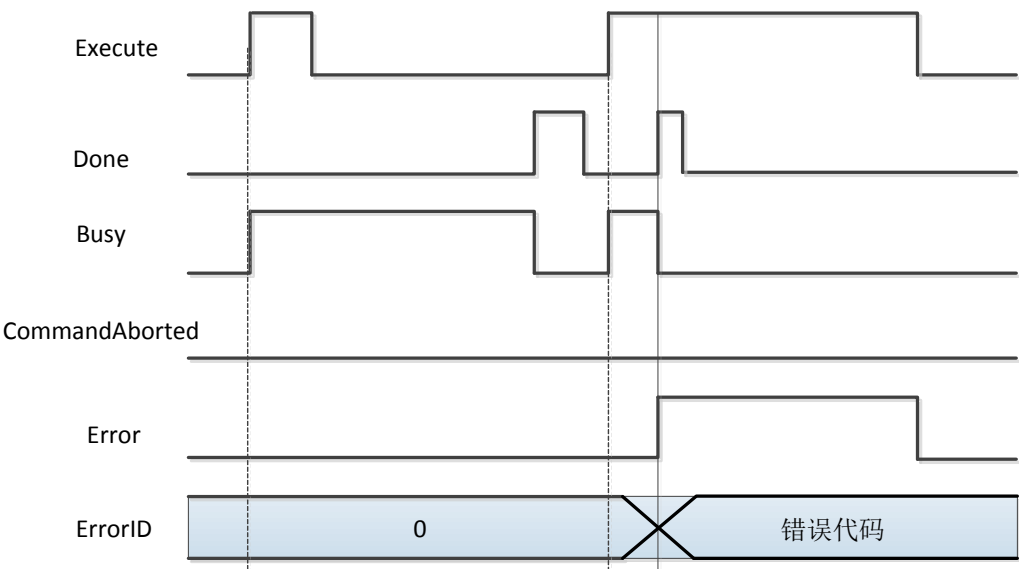
◆ 轴在线性模式下绝对定位

当目标绝对位置 > 起始位置，则正向移动目标绝对位置 - 起始位置距离，当目标位置 < 起始位置时则反向移动起始位置 - 目标位置的距离。线性模式下设定的运行方向不决定轴运行方向。



4) 时序图

- 轴必须处于 Standstill 状态指令才能运行；
- 功能块的 **Execute** 必须有上升沿的条件；
- 功能块的 **Done** 表示指令正常执行完成；
- 功能块的 **Busy** 表示当前功能块正在执行中；



MC\_MoveAdditive

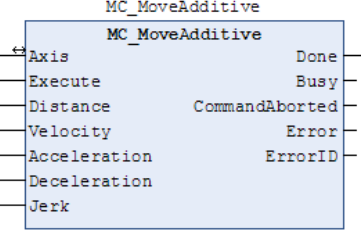
轴在原来指令位置上再叠加 Distance 指定的数据，用于运动轴控制过程的在线叠加位置；本指令运行前设置好相关的

参数，加速度 (Acceleration)、减速度 (Deceleration)、运行速度 (Velocity)；对加速度 (Acceleration) 或减速度 (Deceleration) 的赋值为 0

指令运行错误：本指令在 Discrete Motion 状态下可以在任何时刻添加 MC\_MoveAdditive 的相关执行过程；在 Continuous Motion

中只能在指令执行的某段中；MC\_MoveAdditive 在 standstill 状态下相当于 MC\_MoveRelative 指令。

1) 指令格式

指令	名称	图形表现	ST 表现
MC_MoveAdditive	叠加绝对运动指令		<pre>MC_MoveAdditive(   Axis:= ,   Execute:= ,   Distance:= ,   Velocity:= ,   Acceleration:= ,   Deceleration:= ,   Jerk:= ,   Done=&gt; ,   Busy=&gt; ,   CommandAborted=&gt; ,   Error=&gt; ,   ErrorID=&gt; );</pre>

2) 相关变量

◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴，即 AXIS_REF_SM3 的一个实例

◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
Execute	执行条件	BOOL	TRUE,FALSE	FALSE	输入的一个上升沿将启动功能块的处理
Distance	轴到达位置	LREAL	数据范围	0	此数据为叠加位置数据
Velocity	运行速度	LREAL	数据范围	0	轴运行到目标位置的最大速度
Acceleration	加速度	LREAL	数据范围	0	速度变大时加速度值
Deceleration	减速度	LREAL	数据范围	0	速度变小时减速度值
Jerk	跃度	LREAL	数据范围	0	曲线加减速的斜率变化值

◆ 输出变量

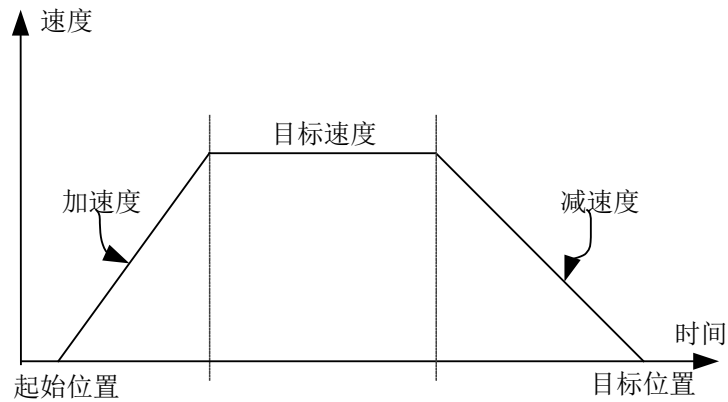
输出变量	名称	数据类型	有效范围	初始值	描述
Done	指令执行完成	BOOL	TRUE,FALSE	FALSE	轴指令执行完成，置为 TRUE
Busy	指令正在执行	BOOL	TRUE,FALSE	FALSE	当前指令正在执行中，置为 TRUE
CommandAbort	指令被中断	BOOL	TRUE,FALSE	FALSE	当前指令被中断，置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时，置为 TRUE
ErrorID	错误代码	SMC_ERROR	参阅 SMC_ERROR	0	异常发生时，输出错误代码

### 3) 功能说明

- ◆ 本功能块为叠加位置指令，Distance 数据为轴的叠加数据；
- ◆ 本功能块运行状态如果为 Discrete Motion, 使用情况会把其他指令的 CommandAbort 置位；
- ◆ 在 standstill 状态，本指令能独立运行，实现相对定位需求；
- ◆ Acceleration 或 Deceleration 为零，指令运行都为异常状态，但轴的状态为 Discrete Motion；
- ◆ 启动指令为 Execute 的上升沿启动。

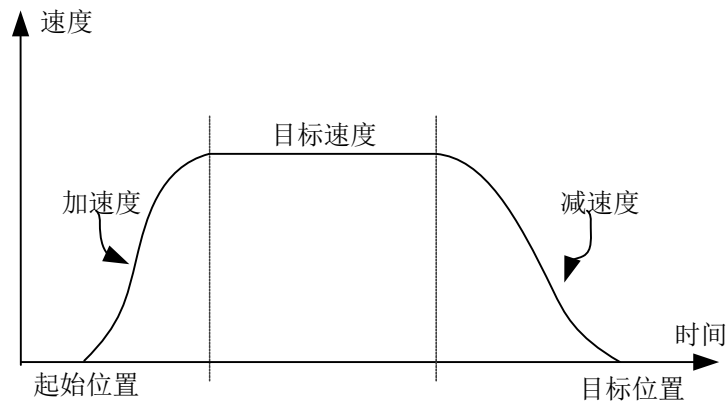
#### 梯形加减速动作

Velocity、Acceleration 和 Deceleration 有数据；而 Jerk 为 0；



#### S 曲线加减速动作

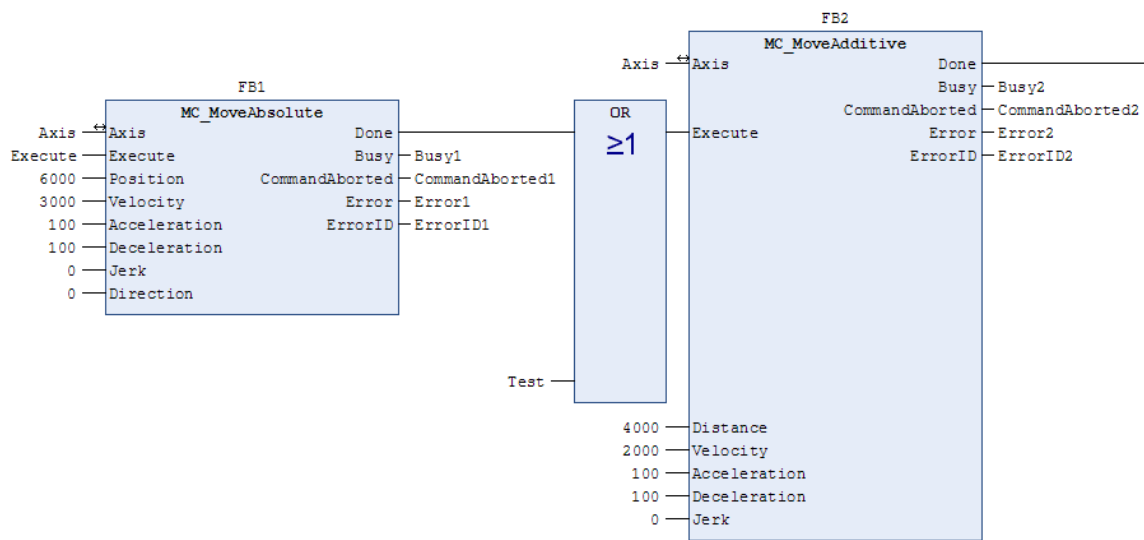
Velocity、Acceleration、Deceleration 和 Jerk 有数据；



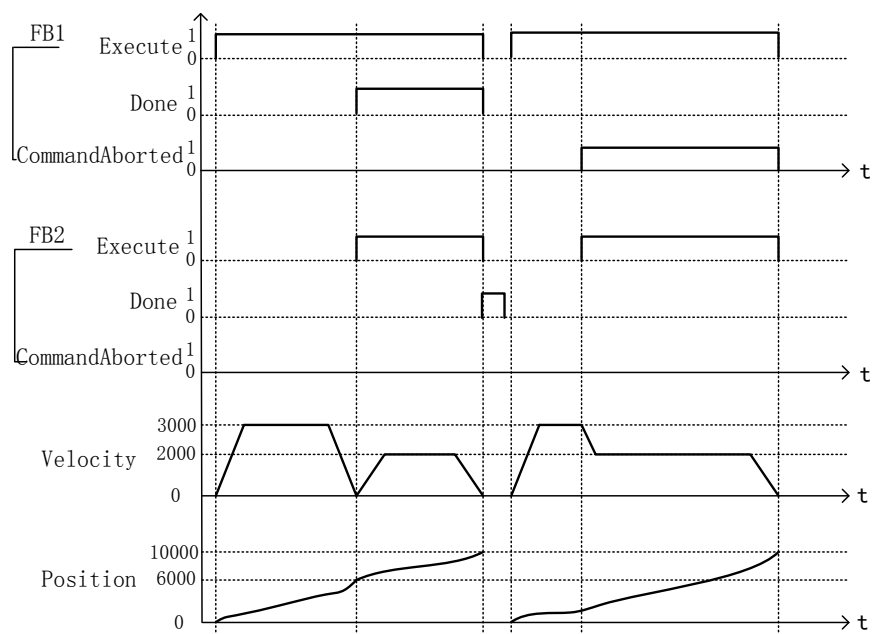
4) 时序图

- 轴必须处于 Standstill 状态指令才能运行；
- 功能块的 **Execute** 必须有上升沿的条件；
- 功能块的 **Done** 表示指令正常执行完成；
- 功能块的 **Busy** 表示当前功能块正在执行中；

◆ 举例说明



◆ 时序操作说明：



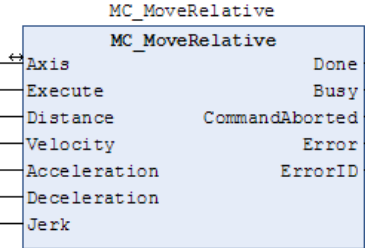
## MC\_MoveRelative

轴按相对位置运行 ( 单位按轴设置 ), 相对位置由 **Distance** 指定; 本指令运行前设置好相关的参数, 加速度 (Acceleration)、减

速度 (Deceleration)、运行速度 (Velocity) 和加减速模式的跃度 (Jerk); 对加速度 (Acceleration) 或减速度 (Deceleration) 的赋值为 0

指令运行错误; 在运行过程中, 一定要关注本指令运行的完整过程, 从用户程序的设计角度避免其他指令中断此指令。

### 1) 指令格式

指令	名称	图形表现	ST 表现
MC_MoveRelative	轴相对定位指令		<pre> MC_MoveRelative(   Axis:= ,   Execute:= ,   Distance:= ,   Velocity:= 10,   Acceleration:= ,   Deceleration:= ,   Jerk:= ,   Done=&gt; ,   Busy=&gt; ,   CommandAborted=&gt; ,   Error=&gt; ,   ErrorID=&gt; );           </pre>

### 2) 相关变量

#### ◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴, 即 AXIS_REF_SM3 的一个实例

#### ◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
Execute	执行条件	BOOL	TRUE,FALSE	FALSE	输入的一个上升沿将启动功能块的处理
Distance	运动相对位置	LREAL	数据范围	0	此数据为运动的相对位置
Velocity	运行速度	LREAL	数据范围	0	轴运行到目标位置的最大速度
Acceleration	加速度	LREAL	数据范围	0	速度变大时加速度值
Deceleration	减速度	LREAL	数据范围	0	速度变小时减速度值
Jerk	跃度	LREAL	数据范围	0	曲线加减速的斜率变化值

#### ◆ 输出变量

输出变量	名称	数据类型	有效范围	初始值	描述
Done	指令执行完成	BOOL	TRUE,FALSE	FALSE	轴指令执行完成, 置为 TRUE
Busy	指令正在执行	BOOL	TRUE,FALSE	FALSE	当前指令正在执行中, 置为 TRUE
CommandAbort	指令被中断	BOOL	TRUE,FALSE	FALSE	当前指令被中断, 置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时, 置为 TRUE
ErrorID	错误代码	SMC_ERROR	参阅 SMC_ERROR	0	异常发生时, 输出错误代码

3) 功能说明

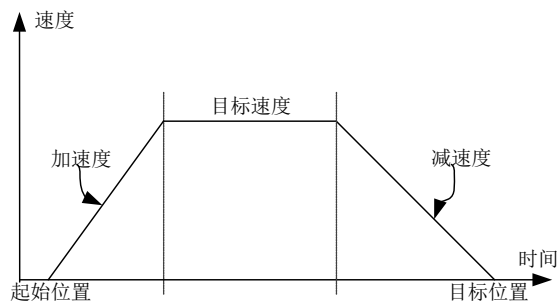
本功能块运行状态为 **Standstill** 中，指令运行时的状态为 **Discrete Motion**，在指令执行中关注本轴的运行状态，避免打断本轴的其他指令或被其他指令打断本轴的执行。

启动指令为 **Execute** 的上升沿启动，本指令在 **Discrete Motion** 可以重复上升沿有效，每次都可以刷新最新的 **Position** 位置。

**Acceleration** 或 **Deceleration** 为零，指令运行都为异常状态，但轴的状态为 **Discrete Motion**；

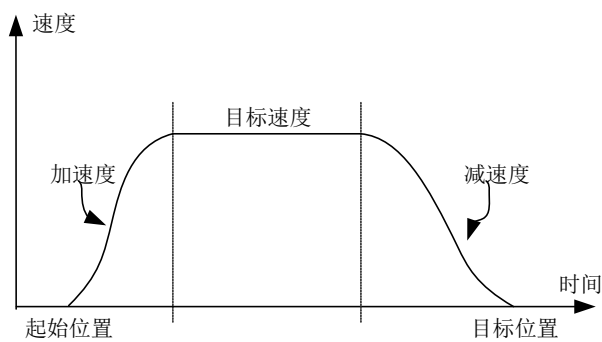
◆ 梯形加减速动作

**Velocity**、**Acceleration** 和 **Deceleration** 有数据；而 **Jerk** 为 0；



◆ S 曲线加减速动作

**Velocity**、**Acceleration**、**Deceleration** 和 **Jerk** 有数据；

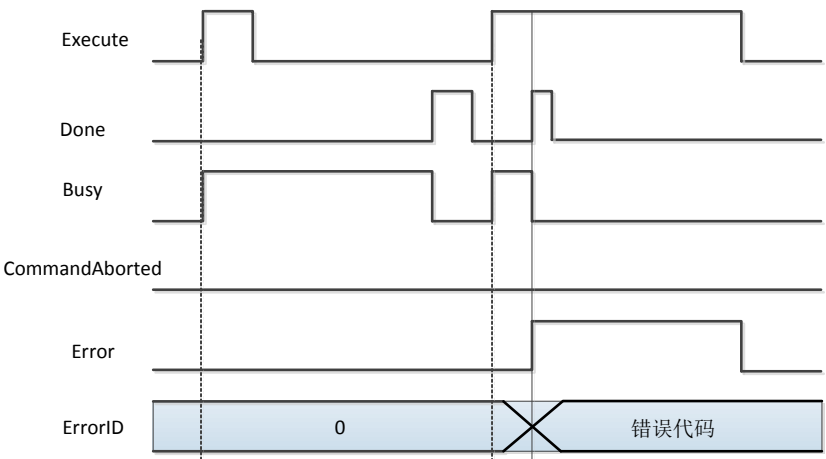


4) 时序图

功能块的 **Execute** 必须有上升沿的条件；

功能块的 **Done** 表示指令正常执行完成；

功能块的 **Busy** 表示当前功能块正在执行中；





MC\_MoveSuperImposed

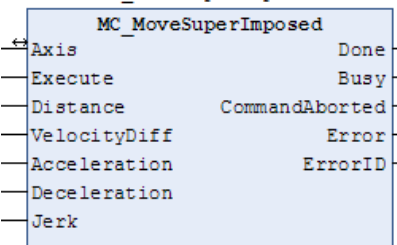
轴在原来指令速度和位置的基础上叠加速度和位置的数据在运行的指令上，对整个原来的指令执行时间模型上没有变化；

通过此指令能解决我们实际运行中由皮带和齿轮间隙误差补偿，能保证运动的一致性；指令运行时需要设置参数叠加位置

(Distance)、速度 (VelocityDiff)、加速度 (Acceleration)、减速度 (Deceleration)、运行速度 (Velocity)；对加速度 (Acceleration) 或减速

度 (Deceleration) 的赋值为 0 指令运行错误；MC\_MoveSuperImposed 在 standstill 状态下相当于 MC\_MoveRelative 指令。

1) 指令格式

指令	名称	图形表现	ST 表现
MC_MoveSuperImposed	叠加相对运动指令		<pre>MC_MoveSuperImposed(   Axis:= ,   Execute:= ,   Distance:= ,   VelocityDiff:= ,   Acceleration:= ,   Deceleration:= ,   Jerk:= ,   Done=&gt; ,   Busy=&gt; ,   CommandAborted=&gt; ,   Error=&gt; ,   ErrorID=&gt; );</pre>

2) 相关变量

◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴，即 AXIS_REF_SM3 的一个实例

◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
Execute	执行条件	BOOL	TRUE,FALSE	FALSE	输入的一个上升沿将启动功能块的处理
Distance	轴到达位置	LREAL	数据范围	0	此数据为叠加位置数据
VelocityDiff	叠加速度	LREAL	数据范围	0	轴运行叠加速度
Acceleration	加速度	LREAL	数据范围	0	速度变大时加速度值
Deceleration	减速度	LREAL	数据范围	0	速度变小时减速度值
Jerk	跃度	LREAL	数据范围	0	曲线加减速的斜率变化值

◆ 输出变量

输出变量	名称	数据类型	有效范围	初始值	描述
Done	指令执行完成	BOOL	TRUE,FALSE	FALSE	轴指令执行完成，置为 TRUE
Busy	指令正在执行	BOOL	TRUE,FALSE	FALSE	当前指令正在执行中，置为 TRUE
CommandAbort	指令被中断	BOOL	TRUE,FALSE	FALSE	当前指令被中断，置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时，置为 TRUE
ErrorID	错误代码	SMC_ERROR	参阅 SMC_ERROR	0	异常发生时，输出错误代码

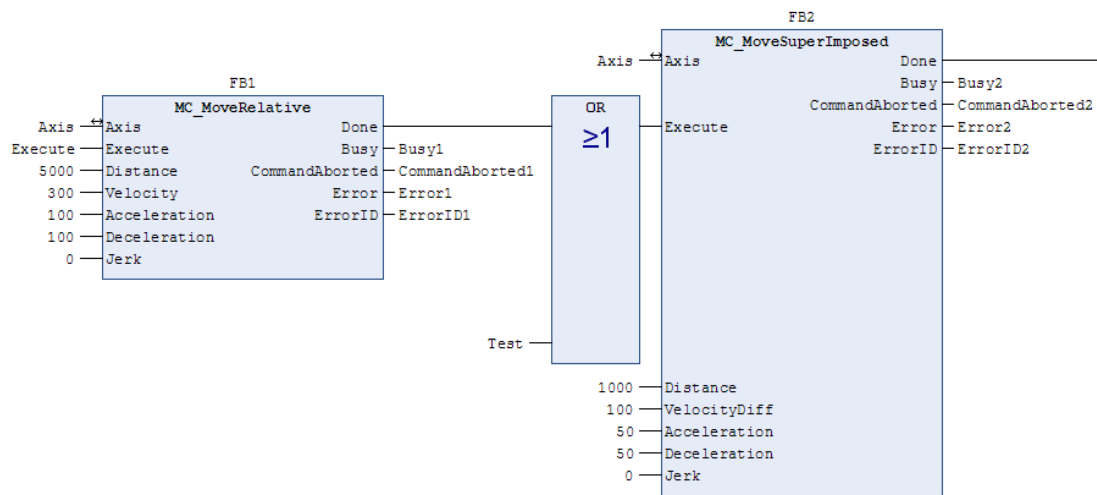
3) 功能说明

本功能块为叠加位置和速度指令，VelocityDiff 和 Distance 分别为叠加在其他指令上的速度和位置；  
在运动模式下 MC\_MoveSuperImposed 可以叠加在任何其他指令；  
MC\_MoveSuperImposed 也可以被 MC\_MoveSuperImposed 中止；  
在状态 StandStill 下，功能块 MC\_MoveSuperimposed 的动作类似于 MC\_MoveRelative；  
启动指令为 Execute 的上升沿启动。

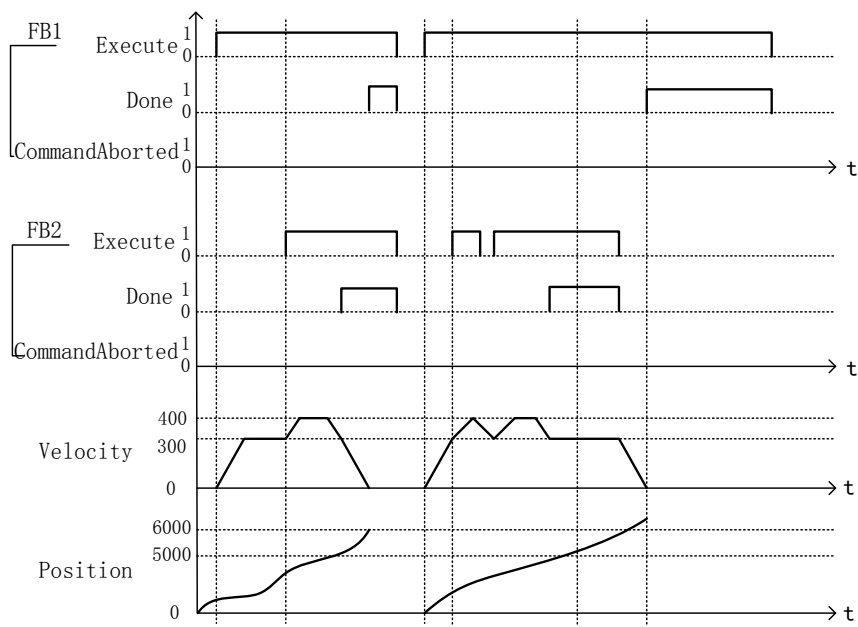
4) 时序图

功能块的 Execute 必须有上升沿的条件；  
功能块的 Done 表示指令正常执行完成；  
功能块的 Busy 表示当前功能块正在执行中；

◆ 举例说明



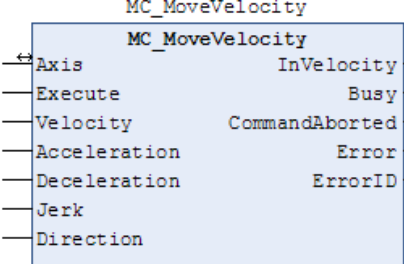
◆ 时序操作说明：



## MC\_MoveVelocity

使用驱动器位置控制模式下的进行模拟速度控制，在轴使能并指令有效的情况下，对 **Velocity** 的赋值能控制驱动器的速度。

### 1) 指令格式

指令	名称	图形表现	ST 表现
MC_MoveVelocity	速度控制指令		<pre>MC_MoveVelocity(     Axis:= ,     Execute:= ,     Velocity:= ,     Acceleration:= ,     Deceleration:= ,     Jerk:= ,     Direction:= ,     InVelocity=&gt; ,     Busy=&gt; ,     CommandAborted=&gt; ,     Error=&gt; ,     ErrorID=&gt; );</pre>

### 2) 相关变量

#### ◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴，即 AXIS_REF_SM3 的一个实例

#### ◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
Execute	执行条件	BOOL	TRUE,FALSE	FALSE	输入的一个上升沿将启动功能块的处理
Velocity	速度设定值	LREAL	数据范围	0	此数据为本指令的速度运行值
Acceleration	加速度	LREAL	数据范围	0	速度变大时加速度值
Deceleration	减速度	LREAL	数据范围	0	速度变小时减速度值
Jerk	跃度	LREAL	数据范围	0	曲线加减速的斜率变化值
Direction	运行方向	MC_Direction	positive, negative, current	current	为运行方向的指令操作

#### ◆ 输出变量

输出变量	名称	数据类型	有效范围	初始值	描述
InVelocity	达到设定速度的标志	BOOL	TRUE,FALSE	FALSE	设定的运行速度达到，置为 TRUE
Busy	指令正在执行	BOOL	TRUE,FALSE	FALSE	当前指令正在执行中，置为 TRUE
CommandAbort	指令被中断	BOOL	TRUE,FALSE	FALSE	当前指令被中断，置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时，置为 TRUE
ErrorID	错误代码	SMC_ERROR	参阅 SMC_ERROR	0	异常发生时，输出错误代码

3) 功能说明

更改 Velocity 参数，对驱动器的模拟速度控制。

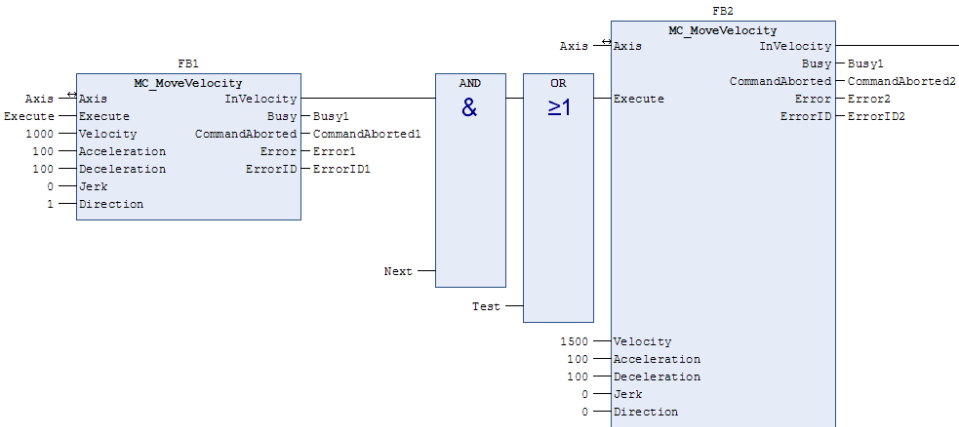
◆ 时序图

功能块的 **Execute** 必须有上升沿的条件；

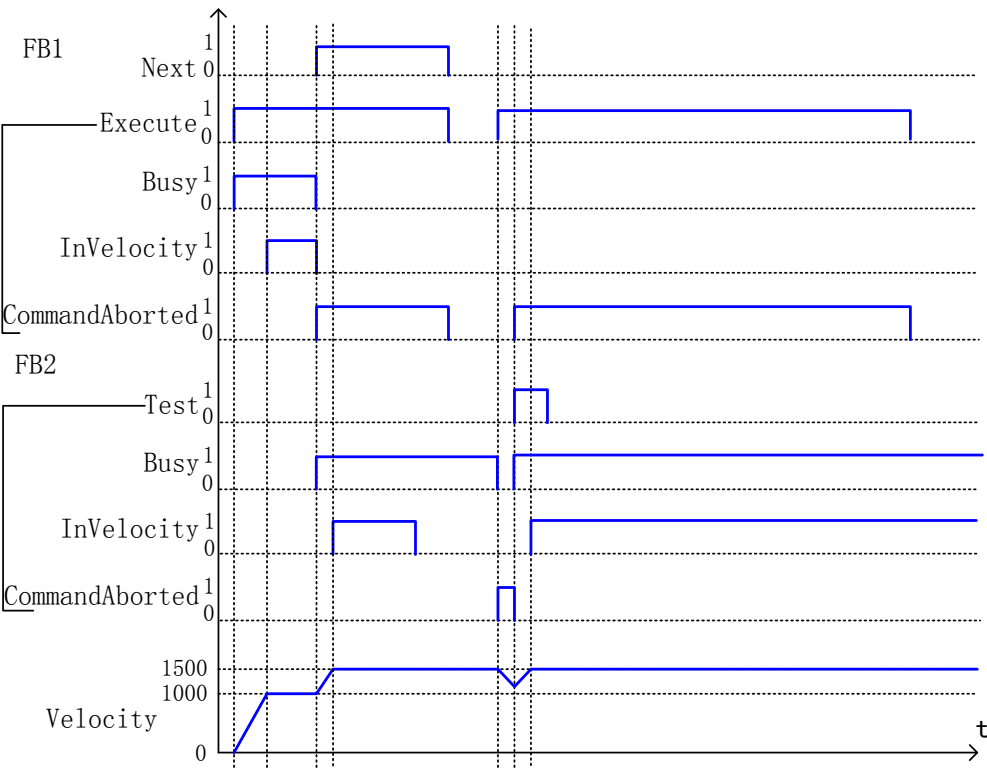
功能块的 **InVelocity** 表示指令的运行速度达到设定值；

功能块的 **Busy** 表示当前功能块正在执行中；

◆ 举例说明

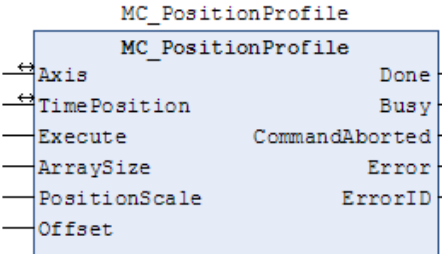


◆ 时序操作说明：



## MC\_PositionProfile

### 1) 指令格式

指令	名称	图形表现	ST 表现
MC_PositionProfile	位置轮廓指令		<pre> MC_PositionProfile(   Axis:= ,   TimePosition:= ,   Execute:= ,   ArraySize:= ,   PositionScale:= ,   Offset:= ,   Done=&gt; ,   Busy=&gt; ,   CommandAborted=&gt; ,   Error=&gt; ,   ErrorID=&gt; );           </pre>

### 2) 相关变量

#### ◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴，即 AXIS_REF_SM3 的一个实例
TimePosition	轴位置运行时间和位置描述	MC_TP_REF			轴位置运行时间和位置数据描述，数据由多组数据组成

#### ◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
Execute	执行条件	BOOL	TRUE,FALSE	FALSE	输入的一个上升沿将启动功能块的处理
ArraySize	动态数组	INT	数据范围	0	运行轮廓中使用的数组个数
PositionScale	综合因子	LREAL	“正数”+“0”	1	MC_TP_REF 中位置的比例因子
Offset	偏移	LREAL		0	位置的总体偏移值

#### ◆ 输出变量

输出变量	名称	数据类型	有效范围	初始值	描述
Done	指令执行完成	BOOL	TRUE,FALSE	FALSE	轴指令执行完成，置为 TRUE
Busy	指令正在执行	BOOL	TRUE,FALSE	FALSE	当前指令正在执行中，置为 TRUE
CommandAbort	指令被中断	BOOL	TRUE,FALSE	FALSE	当前指令被中断，置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时，置为 TRUE
ErrorID	错误代码	SMC_ERROR	参阅 SMC_ERROR	0	异常发生时，输出错误代码

3) 功能说明

本功能块为时间段和位置的轮廓运动模型，运行模式为 **Discrete Motion**，按用户在 **TimePosition** 变量中设定的数据运行。

本功能块运行状态为 **Standstill** 中，指令运行时的状态为 **Discrete Motion**，其他状态无法运行。

启动指令为 **Execute** 的上升沿启动，本指令在 **Discrete Motion** 重复运行。

**TimePosition** 为 **MC\_TP\_REF** 数据类型；

**MC\_TP\_REF** 具体描述如下：

成员	类型	初始值	描述
Number_of_pairs	INT	0	轮廓路径的段数
IsAbsolute	BOOL	TRUE	绝对运动 (TRUE) 和相对运动选择
MC_TP_Array	ARRAY[1..N] OF SMC_TP		时间和位置的数组

**SMC\_TP** 具体描述如下：

成员	类型	初始值	描述
delta_time	TIME	TIME#0ms	位置段的时间
position	LREAL	0	当前的位置值

注：按设置的位置数据对应的速度有变化时都按 **S** 曲线进行相关调整。

◆ 时序图

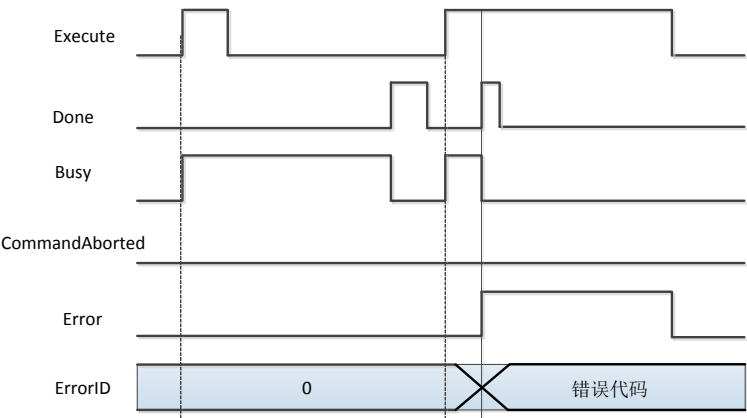
条件 **MC\_TP\_Array** 通过其他方式已经设置才能运行位置轮廓曲线指令；

轴必须处于 **Standstill** 状态指令才能运行；

功能块的 **Execute** 必须有上升沿的条件；

功能块的 **Done** 表示指令正常执行完成；

功能块的 **Busy** 表示当前功能块正在执行中；



4) 错误说明

错误的出现为轴状态不是在 **Standstill** 中启动指令或指令系统中的参数错误，出现轴错误只能清除错误后才开始运行。

【注意】：请阅读 [“附录 C 错误代码说明”](#) 以了解相关错误代码说明。

## MC\_Power

### 1) 指令格式

指令	名称	图形表现	ST 表现
MC_Power	轴使能指令		<pre> MC_Power(   Axis:= ,   Enable:= ,   bRegulatorOn:= ,   bDriveStart:= ,   Status=&gt; ,   bRegulatorRealState=&gt; ,   bDriveStartRealState=&gt; ,   Busy=&gt; ,   Error=&gt; ,   ErrorID=&gt; ); </pre>

### 2) 相关变量

#### ◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
Enable	有效	BOOL	TRUE,FALSE	FALSE	设置为 TRUE 则功能块开始处理
bRegulatorOn	使能状态	BOOL	TRUE,FALSE	FALSE	设置为 TRUE 则设置轴为使能状态
bDriveStart	允许驱动	BOOL	TRUE,FALSE	FALSE	设置为 TRUE 以关闭功能块的紧急停止处理

#### ◆ 输出变量

输出变量	名称	数据类型	有效范围	初始值	描述
Status	可运行状态	BOOL	TRUE,FALSE	FALSE	如果轴已经准备好运动，置为 TRUE
bRegulatorRealState	轴使能信号状态	BOOL	TRUE,FALSE	FALSE	当轴使能处于有效状态时，置为 TRUE
bDriveStartRealState	允许驱动状态	BOOL	TRUE,FALSE	FALSE	如果轴没有被快速停止机制中断，置为 TRUE
Busy	执行中	BOOL	TRUE,FALSE	FALSE	如果功能块的处理没有完成，置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时，置为 TRUE
ErrorID	错误代码	SMC_ERROR	参阅 SMC_ERROR	0	异常发生时，输出错误代码

#### ◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴，即 AXIS_REF_SM3 的一个实例

### 3) 功能说明

只有在输入 Enable 为 TRUE 的时候，其它的输入才会被功能块处理。

如果功能块 MC\_Power 已经被调用，并且 bRegulatorOn=FALSE，那么功能块将设置相关轴的轴状态（nAxisState）为 power\_off 状态，表明驱动器还没有做好运动准备。

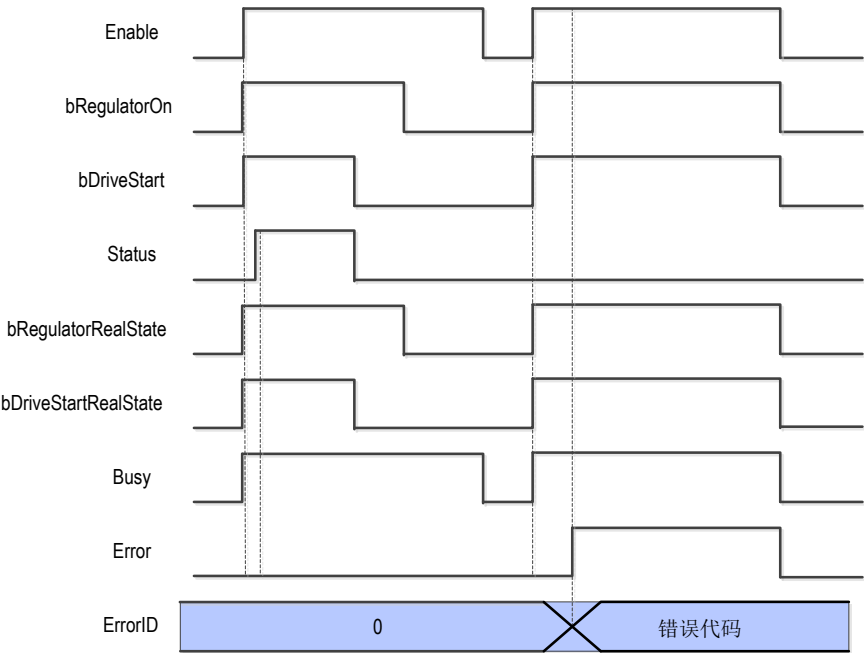
如果功能块 MC\_Power 已经被调用，并且 bRegulatorOn=TRUE，如果此时轴没有错误发生，那么功能块将设置相关轴的轴状态（nAxisState）为 standstill 状态；如果有错误发生，将输出相应的错误状态。

如果 Enable, bRegulatorOn 以及 bDriveStart 都为 TRUE，但是输出 Status 在一定时间后仍为 FALSE，那么输出 Error 将会被置位。当在使能状态情况下产生一个硬件问题，可能会发生这种情况。

如果使能信号丢失（通常在操作模式下），相关轴的 nAxisState 将会被置位 ErrorStop 状态。

◆ 时序图

将 Enable 设为 TRUE,bRegulatorOn 设为 TRUE,bDriveStart 设为 TRUE, 表示正在受理指令的 Busy 变为 TRUE, 然后轴进入使能 ON 状态, Status 状态变为 TRUE。



4) 错误说明

请勿在正在执行 MC\_Power 指令的轴中, 编写用于启动其它实例的 MC\_Power 指令的程序。原则上 1 根轴只能设 1 个 MC\_Power 指令。

如果在正在执行 MC\_Power 指令的轴中, 启动其它实例的 MC\_Power 指令, 则优先执行后执行的 MC\_Power 指令。

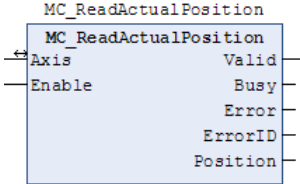
【注意】: 请阅读 “附录 C 错误代码说明” 以了解相关错误代码说明。



MC\_ReadActualPosition

指令读取驱动器运行的实际位置，保存在自己定义的变量单元中。

1) 指令格式

指令	名称	图形表现	ST 表现
MC_ReadActualPosition	实际位置读取指令		<pre>MC_ReadActualPosition(   Axis:= ,   Enable:= ,   Valid=&gt; ,   Busy=&gt; ,   Error=&gt; ,   ErrorID=&gt; ,   Position=&gt; );</pre>

2) 相关变量

◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴，即 AXIS_REF_SM3 的一个实例

◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
Enable	执行条件	BOOL	TRUE,FALSE	FALSE	为 TRUE 状态读取伺服的当前位置

◆ 输出变量

输出变量	名称	数据类型	有效范围	初始值	描述
Valid	位置数据可获取标志	BOOL	TRUE,FALSE	FALSE	能正确的获取驱动器的位置，置为 TRUE
Busy	指令正在执行	BOOL	TRUE,FALSE	FALSE	当前指令正在执行中，置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时，置为 TRUE
ErrorID	错误代码	SMC_ERROR	参阅 SMC_ERROR	0	异常发生时，输出错误代码
Position	获取到的轴位置	LREAL	轴位置	0	指令读出来的轴位置数据

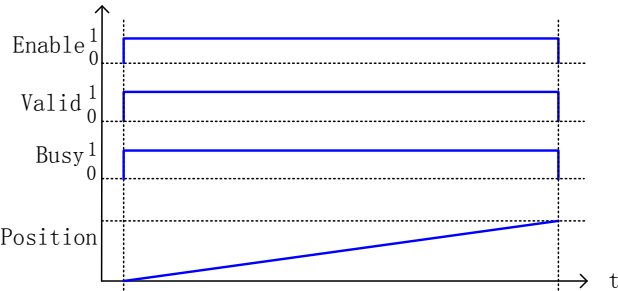
3) 功能说明

通过本指令读取驱动器中的实际位置指令，指令为 Enable 电平使能效应。指令可以重复多次使用，互不影响。

◆ 时序图

功能块的 Enable 必须为 TRUE 的条件；  
功能块的 Valid 表示读出的 Position 为有效的数据值；  
功能块的 Busy 表示当前功能块正在执行中；

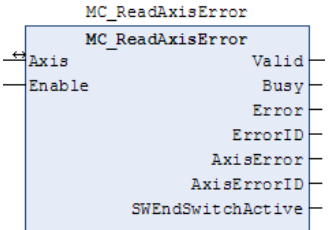
时序操作说明：



MC\_ReadAxisError

指令读取轴的错误情况，保存在自己定义的变量单元中。

1) 指令格式

指令	名称	图形表现	ST 表现
MC_ReadAxisError	读轴的错误状态		<pre>MC_ReadAxisError(   Axis:= ,   Enable:= ,   Valid=&gt; ,   Busy=&gt; ,   Error=&gt; ,   ErrorID=&gt; ,   AxisError=&gt; ,   AxisErrorID=&gt; ,   SWEndSwitchActive=&gt; );</pre>

2) 相关变量

◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴，即 AXIS_REF_SM3 的一个实例

◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
Enable	执行条件	BOOL	TRUE,FALSE	FALSE	为 TRUE 状态读取伺服的当前位置

◆ 输出变量

输出变量	名称	数据类型	有效范围	初始值	描述
Valid	错误数据可获取标志	BOOL	TRUE,FALSE	FALSE	能获取轴的错误数据，置为 TRUE
Busy	指令正在执行	BOOL	TRUE,FALSE	FALSE	当前指令正在执行中，置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时，置为 TRUE
ErrorID	错误代码	SMC_ERROR	参阅 SMC_ERROR	0	异常发生时，输出错误代码
AxisError	轴错误标示	BOOL	TRUE,FALSE	FALSE	读出轴为错误，对应的标示置位
AxisErrorID	轴错误码	DWORD		0	读出轴为错误码
SWEndSwitchActive	软限位开关有效	BOOL	TRUE,FALSE	FALSE	在指令读取中，检查软限位开关状态

3) 功能说明

通过 MC\_ReadAxisError 读取驱动器中的错误码，指令为 Enable 电平使能效应。指令可以重复多次使用，互不影响。

◆ 时序图

功能块的 Enable 必须为 TRUE 的条件；

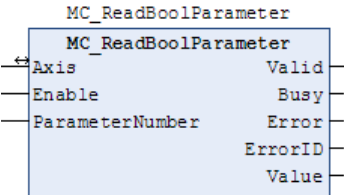
功能块的 Valid 表示读出的 AxisError 和 AxisErrorID 为有效的数据值；

功能块的 Busy 表示当前功能块正在执行中；

MC\_ReadBoolParameter

指令读取驱动轴的位参数，保存在自己定义的变量单元中。

1) 指令格式

指令	名称	图形表现	ST 表现
MC_ReadBoolParameter	读取轴的位参数		<pre>MC_ReadBoolParameter(   Axis:= ,   Enable:= ,   ParameterNumber:= ,   Valid=&gt; ,   Busy=&gt; ,   Error=&gt; ,   ErrorID=&gt; ,   Value=&gt; );</pre>

2) 相关变量

◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴，即 AXIS_REF_SM3 的一个实例

◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
Enable	执行条件	BOOL	TRUE,FALSE	FALSE	为 TRUE 状态读取伺服的当前位置
ParameterNumber	轴参数的序号	DINT		0	访问轴参数的索引和子索引和序号

注 :ParameterNumber (DINT) = -DWORD\_TO\_DINT(SHL(USINT\_TO\_DOWRD(usiDataLength), 24))( 对象字典中的数据长度 )

+ SHL(UINT\_TO\_DWORD(uiIndex), 8) ( 对象字典中的索引 -16BIT)

+ usisubIndex( 对象字典中的子索引 -8BIT))

usiDataLength: 按字节数填写；1 字节为 16#01；2 字节为 16#02；4 字节为 16#04 等。

◆ 输出变量

输出变量	名称	数据类型	有效范围	初始值	描述
Valid	位置数据可获取标志	BOOL	TRUE,FALSE	FALSE	能正确的获取驱动器的位置，置为 TRUE
Busy	指令正在执行	BOOL	TRUE,FALSE	FALSE	当前指令正在执行中，置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时，置为 TRUE
ErrorID	错误代码	SMC_ERROR	参阅 SMC_ERROR	0	异常发生时，输出错误代码
Value	获取到的轴位状态	BOOL	TRUE,FALSE	FALSE	指令读出来的轴位状态

### 3) 功能说明

通过 MC\_ReadBoolParam 读取驱动器中的位数据状态，指令为 Enable 电平使能效应。指令可以重复多次使用，互不影响。

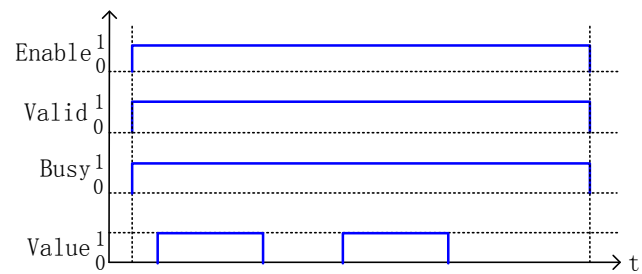
◆ 时序图

功能块的 Enable 必须为 TRUE 的条件；

功能块的 Valid 表示读出的 Valid 为有效的位状态数据；

功能块的 Busy 表示当前功能块正在执行中；

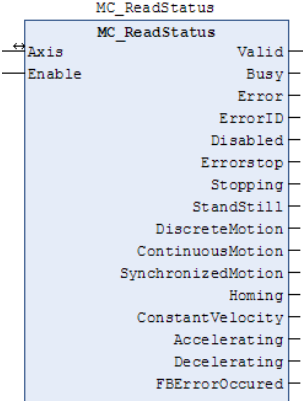
◆ 时序操作说明：



## MC\_ReadStatus

指令读取轴的状态数据，保存在自己定义的变量单元中。

### 1) 指令格式

指令	名称	图形表现	ST 表现
MC_ReadStatus	读轴的状态		<pre> MC_ReadStatus(   Axis:= ,   Enable:= ,   Valid=&gt; ,   Busy=&gt; ,   Error=&gt; ,   ErrorID=&gt; ,   Disabled=&gt; ,   Errorstop=&gt; ,   Stopping=&gt; ,   StandStill=&gt; ,   DiscreteMotion=&gt; ,   ContinuousMotion=&gt; ,   SynchronizedMotion=&gt; ,   Homing=&gt; ,   ConstantVelocity=&gt; ,   Accelerating=&gt; ,   Decelerating=&gt; ,   FBErrorOccured=&gt; ); </pre>

### 2) 相关变量

#### ◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴，即 AXIS_REF_SM3 的一个实例

#### ◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
Enable	执行条件	BOOL	TRUE,FALSE	FALSE	为 TRUE 状态读取伺服的当前位置

#### ◆ 输出变量

输出变量	名称	数据类型	有效范围	初始值	描述
Valid	错误数据可获取标志	BOOL	TRUE,FALSE	FALSE	能获得轴的错误数据，置为 TRUE
Busy	指令正在执行	BOOL	TRUE,FALSE	FALSE	当前指令正在执行中，置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时，置为 TRUE
ErrorID	错误代码	SMC_ERROR	参阅 SMC_ERROR	0	异常发生时，输出错误代码
Disabled	轴没使能状态	BOOL	TRUE,FALSE	FALSE	轴在没使能状态为 TRUE；
Errorstop	轴错误状态	BOOL	TRUE,FALSE	FALSE	轴在错误运行状态为 TRUE；
Stoping	轴停止过程状态	BOOL	TRUE,FALSE	FALSE	轴在停止过程中为 TRUE
StandStill	轴标准状态	BOOL	TRUE,FALSE	FALSE	轴在标准（能运行）状态中为 TRUE
DiscreteMotion	轴离散运动状态	BOOL	TRUE,FALSE	FALSE	轴在离散运动状态中为 TRUE
ContinuousMotion	轴连续运动状态	BOOL	TRUE,FALSE	FALSE	轴在连续运动状态中为 TRUE
SynchronizedMotion	轴同步运行状态	BOOL	TRUE,FALSE	FALSE	轴在同步运动状态中为 TRUE
Homing	轴回原点状态	BOOL	TRUE,FALSE	FALSE	轴在回原点状态中为 TRUE
ConstantVelocity	轴运行速度到达	BOOL	TRUE,FALSE	FALSE	轴达到运行速度中为 TRUE
Accelerating	轴加速过程状态	BOOL	TRUE,FALSE	FALSE	轴加速过程状态为 TRUE
Dccelerating	轴减速过程状态	BOOL	TRUE,FALSE	FALSE	轴减速过程状态为 TRUE
FBErrorOccured	轴功能块错误出现标志	BOOL	TRUE,FALSE	FALSE	轴功能块错误标志为 TRUE

### 3) 功能说明

通过 MC\_ReadStatus 对应轴的各种状态，指令为 **Enable** 电平使能效应。指令可以重复多次使用，互不影响。

功能块的 **Enable** 必须为 **TRUE** 的条件；

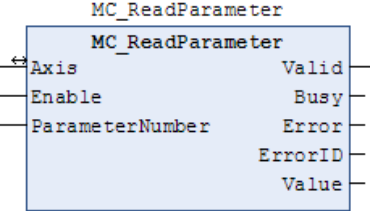
功能块的 **Valid** 表示读出的后面个状态标志的各种数据；

功能块的 **Busy** 表示当前功能块正在执行中。

## MC\_ReadParameter

指令读取驱动轴的参数，保存在自己定义的变量单元中。

### 1) 指令格式

指令	名称	图形表现	ST 表现
MC_ReadParameter	读取轴的参数		<pre>MC_ReadParameter(   Axis:= ,   Enable:= ,   ParameterNumber:= ,   Valid=&gt; ,   Busy=&gt; ,   Error=&gt; ,   ErrorID=&gt; ,   Value=&gt; );</pre>

### 2) 相关变量

#### ◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴，即 AXIS_REF_SM3 的一个实例

#### ◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
Enable	执行条件	BOOL	TRUE,FALSE	FALSE	为 TRUE 状态读取伺服的当前位置
ParameterNumber	轴参数的序号	DINT		0	访问轴参数的索引和子索引和序号

注:ParameterNumber (DINT) = -DWORD\_TO\_DINT(SHL(USINT\_TO\_DOWRD(usiDataLength), 24)( 对象字典中的数据长度 )

+ SHL(UINT\_TO\_DWORD(uiIndex), 8) ( 对象字典中的索引 -16BIT)

+ usisubIndex( 对象字典中的子索引 -8BIT)

usiDataLength: 按字节数填写；1 字节为 16#01；2 字节为 16#02；4 字节为 16#04 等。

#### ◆ 输出变量

输出变量	名称	数据类型	有效范围	初始值	描述
Valid	位置数据可获取标志	BOOL	TRUE,FALSE	FALSE	能正确的获取驱动器的位置，置为 TRUE
Busy	指令正在执行	BOOL	TRUE,FALSE	FALSE	当前指令正在执行中，置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时，置为 TRUE
ErrorID	错误代码	SMC_ERROR	参阅 SMC_ERROR	0	异常发生时，输出错误代码
Value	获取到的轴参数	LREAL		0	指令读出来的轴参数

### 3) 功能说明

通过 MC\_ReadParam 读取驱动器中的位数据状态，指令为 **Enable** 电平使能效应。指令可以重复多次使用，互不影响。

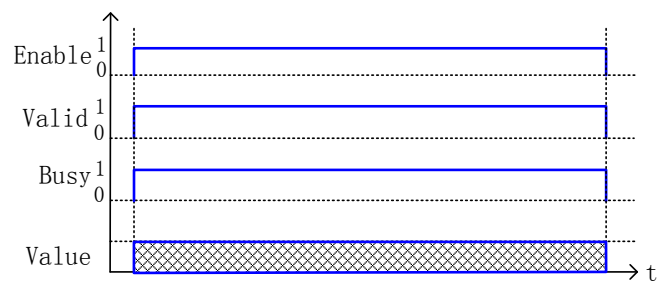
#### ◆ 时序图

功能块的 **Enable** 必须为 **TRUE** 的条件；

功能块的 **Valid** 表示读出的 **Valid** 为有效的位状态数据；

功能块的 **Busy** 表示当前功能块正在执行中；

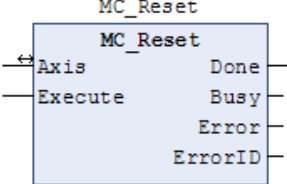
时序操作说明：





MC\_Reset

1) 指令格式

指令	名称	图形表现	ST 表现
MC_Reset	轴错误状态复位指令		<pre>MC_Reset(   Axis:= ,   Execute:= ,   Done=&gt; ,   Busy=&gt; ,   Error=&gt; ,   ErrorID=&gt; );</pre>

2) 相关变量

◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴，即 AXIS_REF_SM3 的一个实例

◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
Execute	执行条件	BOOL	TRUE,FALSE	FALSE	输入的一个上升沿将启动功能块的处理

◆ 输出变量

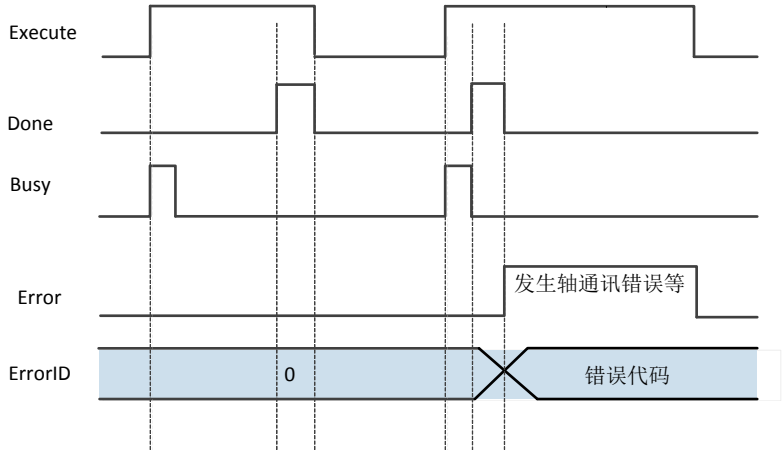
输出变量	名称	数据类型	有效范围	初始值	描述
Done	指令执行完成	BOOL	TRUE,FALSE	FALSE	轴指令执行完成，置为 TRUE
Busy	指令正在执行	BOOL	TRUE,FALSE	FALSE	当前指令正在执行中，置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时，置为 TRUE
ErrorID	错误代码	SMC_ERROR	参阅 SMC_ERROR	0	异常发生时，输出错误代码

3) 功能说明

本功能块在轴通讯正常的情况下，把轴状态处于 errorstop 变为 Standstill，把轴的异常状态变为正常可运行的状态；

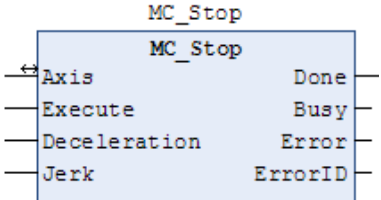
当出现轴 errorstop 无法复位，Axis.bCommunication 为 FLASE 状态，必须要重新建立主站和从站轴的通讯；  
在指令中的 Busy 标志位接通的时间非常短，使用时请注意；

◆ 时序图



MC\_Stop

1) 指令格式

指令	名称	图形表现	ST 表现
MC_Stop	轴停止命令		<pre>MC_Stop(   Axis:= ,   Execute:= ,   Deceleration:= ,   Jerk:= ,   Done=&gt; ,   Busy=&gt; ,   Error=&gt; ,   ErrorID=&gt; );</pre>

2) 相关变量

◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴，即 AXIS_REF_SM3 的一个实例

◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
Execute	执行条件	BOOL	TRUE,FALSE	FALSE	输入的一个上升沿将启动功能块的处理
Deceleration	减速度	LREAL	“正数”+”0”	0	功能块的减速度 (u/S^2)
Jerk	跃度	LREAL	“正数”+”0”	0	指定跃度 [指令单位 /S^3]

◆ 输出变量

输出变量	名称	数据类型	有效范围	初始值	描述
Done	指令执行完成	BOOL	TRUE,FALSE	FALSE	轴指令执行完成，置为 TRUE
Busy	指令正在执行	BOOL	TRUE,FALSE	FALSE	当前指令正在执行中，置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时，置为 TRUE
ErrorID	错误代码	SMC_ERROR	参阅 SMC_ERROR	0	异常发生时，输出错误代码

3) 功能说明

本功能块为在正常运行的情况下停止一个轴的运动，当轴处于 **stopping** 状态对本轴的任何指令都是无效的。

当轴状态处于 **stopping** 时 ,Execute 为 Flase 状态 ,Done 输出状态为 True, 轴状态变为 Standstill;

本功能块运行状态为运行状态 (Motion) 才能运行，其他状态无法运行。

启动指令为 Execute 的上升沿启动。

在 MC\_Stop 有效执行的过程 Busy 有效时，再一次启动 MC\_Stop 指令系统处于 Errorstop 状态

◆ 时序图

轴必须处于运行状态 (Motion) 指令才能运行；

功能块的 Execute 必须有上升沿的条件；

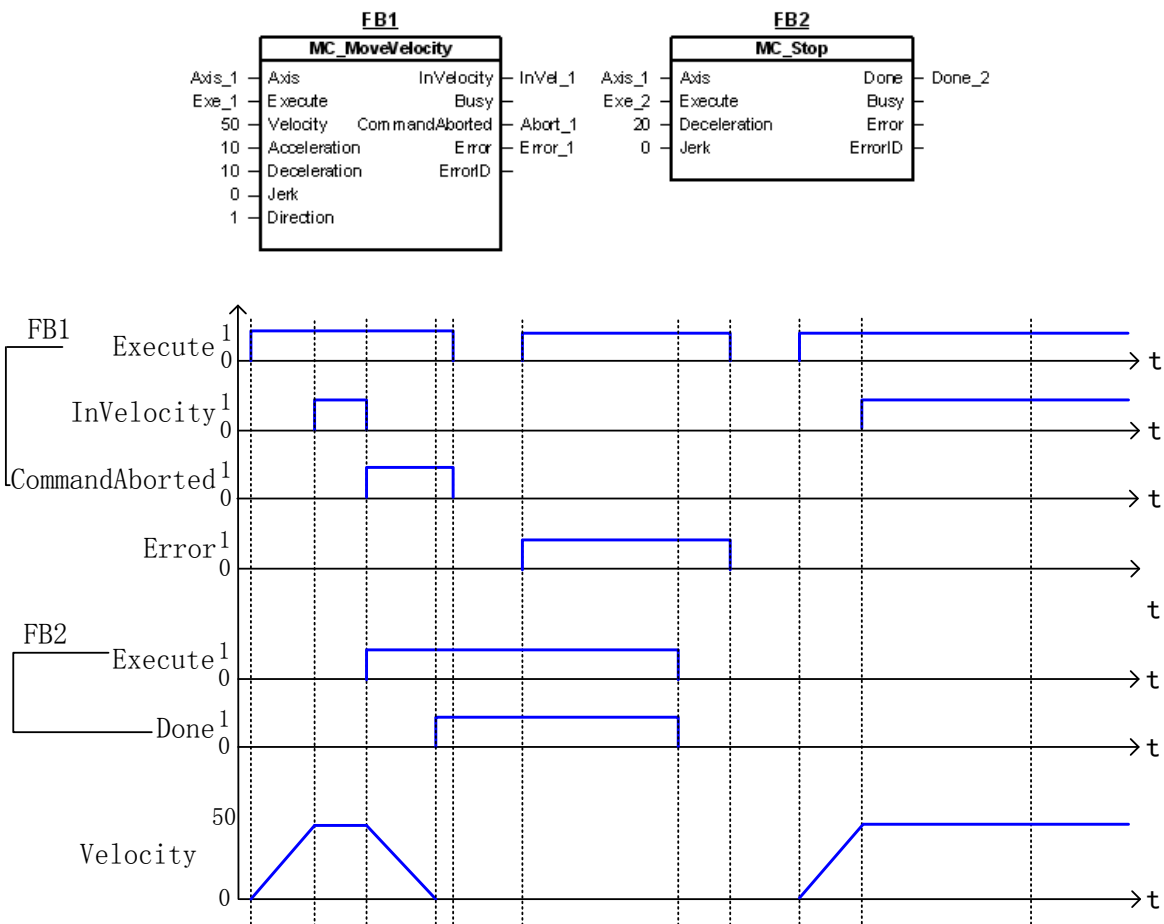
功能块的 Done 表示指令正常执行完成；

功能块的 Busy 表示当前功能块正在执行中；

功能块的 CommandAborted 表示指令被其他运动控制指令中断，此时标志位为 TRUE；

例程：在执行 MC\_MoveVelocity 指令和 MC\_Stop 指令在不同的时序操作中对应的标志位的变化；

对 CommandAborted 的处理描述如下图的时序描述。



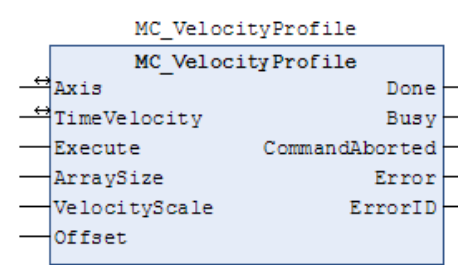
4) 错误说明

MC\_Stop 有重复性的指令运行时，错误标志 Error 为 True,ErrorID 为 SMC\_MS\_AXI 错误；

【注意】：请阅读“附录 C 错误代码说明”以了解相关错误代码说明。

MC\_VelocityProfile

1) 指令格式

指令	名称	图形表现	ST 表现
MC_VelocityProfile	速度轮廓指令		<pre>MC_VelocityProfile(   Axis:= ,   TimeVelocity:= ,   Execute:= ,   ArraySize:= ,   VelocityScale:= ,   Offset:= ,   Done=&gt; ,   Busy=&gt; ,   CommandAborted=&gt; ,   Error=&gt; ,   ErrorID=&gt; );</pre>

2) 相关变量

◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴，即 AXIS_REF_SM3 的一个实例
TimeVelocity	轴速度运行时间和速度描述	MC_TV_REF			轴速度运行时间和速度数据描述，由多组数据组成

◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
Execute	执行条件	BOOL	TRUE,FALSE	FALSE	输入的一个上升沿将启动功能块的处理
ArraySize	动态数组	INT	数据范围	0	运行轮廓中使用的数组个数
VelocityScale	速度因子	LREAL	“正数”，” 0”	1	速度的比例因子
Offset	偏移	LREAL		0	速度值的总体偏移值

◆ 输出变量

输出变量	名称	数据类型	有效范围	初始值	描述
Done	指令执行完成	BOOL	TRUE,FALSE	FALSE	指令执行完成，置为 TRUE
Busy	指令正在执行	BOOL	TRUE,FALSE	FALSE	当前指令正在执行中，置为 TRUE
CommandAbort	指令被中断	BOOL	TRUE,FALSE	FALSE	当前指令被中断，置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时，置为 TRUE
ErrorID	错误代码	SMC_ERROR	参阅 SMC_ERROR	0	异常发生时，输出错误代码

3) 功能说明

本功能块为时间段和速度的轮廓运动模型，运行模式为 Continuous Motion，按用户在 TimeVelocity 变量中设定的数据运行。

本功能块运行状态为 Standstill 中，指令运行时的状态为 Discrete Motion, 其他状态无法运行。

启动指令为 Execute 的上升沿启动，本指令在 Discrete Motion 重复运行。

TimeVelocity 为 MC\_TV\_REF 数据类型；

MC\_TV\_REF 具体描述如下：

成员	类型	初始值	描述
Number_of_pairs	INT	0	轮廓路径的段数
IsAbsolute	BOOL	TRUE	绝对运动 (TRUE) 和相对运动选择
MC_TV_Array	ARRAY[1..N] OF SMC_TV		时间和速度的数组

SMC\_TV 具体描述如下：

成员	类型	初始值	描述
delta_time	TIME	TIME#0ms	速度值段的时间
Velocity	LREAL	0	当前记录的速度值

注：整个速度的过程为 S 曲线加减速的方式，每段轮廓都速度都为叠加的计算方式；在指令重复运行时，速度也为叠加方式，在指令使用时避免速度超限的出现；重复运行一定把本轴的状态重回到 Standstill 状态。

◆ 时序图

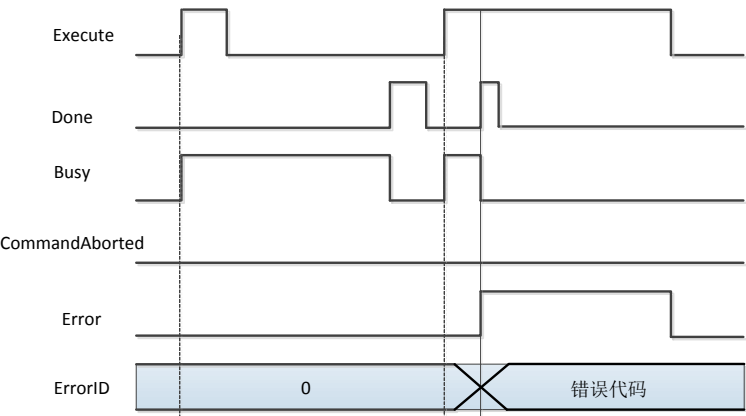
条件 MC\_TV\_Array 通过其他方式已经设置才能运行位置轮廓曲线指令；

轴必须处于 Standstill 状态指令才能运行；

功能块的 Execute 必须有上升沿的条件；

功能块的 Done 表示指令正常执行完成；

功能块的 Busy 表示当前功能块正在执行中；



4) 错误说明

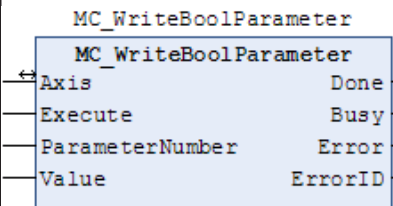
错误的出现为轴状态不是在 Standstill 中启动指令或指令系统中的参数错误，出现轴错误只能清除错误后才开始运行。

【注意】：请阅读“附录 C 错误代码说明”以了解相关错误代码说明。

MC\_WriteBoolParameter

指令设置驱动轴的位参数。

1) 指令格式

指令	名称	图形表现	ST 表现
MC_WriteBoolParameter	设置轴的位参数		<pre>MC_WriteBoolParameter(   Axis:= ,   Execute:= ,   ParameterNumber:= ,   Value:= ,   Done=&gt; ,   Busy=&gt; ,   Error=&gt; ,   ErrorID=&gt; );</pre>

2) 相关变量

◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴，即 AXIS_REF_SM3 的一个实例

◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
Execute	执行条件	BOOL	TRUE,FALSE	FALSE	为上升沿操作驱动一次设置操作
ParameterNumber	轴参数的序号	DINT		0	访问轴参数的索引和子索引和序号
Value	设定值	BOOL	TRUE,FALSE	FALSE	设置位参数值

注 :ParameterNumber (DINT) = -DWORD\_TO\_DINT(SHL(USINT\_TO\_DOWRD(usiDataLength), 24))( 对象字典中的数据长度 )

+ SHL(UINT\_TO\_DWORD(uiIndex), 8) ( 对象字典中的索引 -16BIT)

+ usisubIndex( 对象字典中的子索引 -8BIT)

usiDataLength: 按字节数填写；1 字节为 16#01；2 字节为 16#02；4 字节为 16#04 等。

◆ 输出变量

输出变量	名称	数据类型	有效范围	初始值	描述
Done	设置操作成功	BOOL	TRUE,FALSE	FALSE	设置操作成功置为 TRUE
Busy	指令正在执行	BOOL	TRUE,FALSE	FALSE	当前指令正在执行中，置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时，置为 TRUE
ErrorID	错误代码	SMC_ERROR	参阅 SMC_ERROR	0	异常发生时，输出错误代码

### 3) 功能说明

通过 `MC_WriteBoolParameter` 设置轴的位参数，指令为 `Execute` 上升沿触发。指令可以重复多次使用，互不影响。

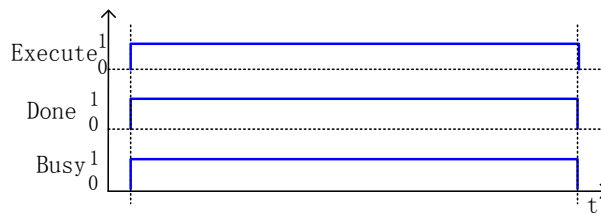
◆ 时序图

功能块的 `Execute` 必须为上升沿触发条件；

功能块的 `Done` 表示设置操作成功；

功能块的 `Busy` 表示当前功能块正在执行中；

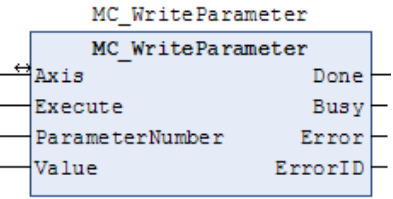
◆ 时序操作说明：



MC\_WriteParameter

指令读取驱动轴的参数，保存在自己定义的变量单元中。

1) 指令格式

指令	名称	图形表现	ST 表现
MC_WriteParameter	设置轴参数		<pre>MC_WriteParameter(   Axis:= ,   Execute:= ,   ParameterNumber:= ,   Value:= ,   Done=&gt; ,   Busy=&gt; ,   Error=&gt; ,   ErrorID=&gt; );</pre>

2) 相关变量

◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴，即 AXIS_REF_SM3 的一个实例

◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
Execute	执行条件	BOOL	TRUE,FALSE	FALSE	为上升沿操作驱动一次设置操作
ParameterNumber	轴参数的序号	DINT		0	访问轴参数的索引和子索引和序号
Value	设定值	LREAL			设置位参数值

注 :ParameterNumber (DINT) = -DWORD\_TO\_DINT(SHL(USINT\_TO\_DOWRD(usiDataLength), 24))( 对象字典中的数据长度 )

+ SHL(UINT\_TO\_DWORD(uiIndex), 8) ( 对象字典中的索引 -16BIT)

+ usisubIndex( 对象字典中的子索引 -8BIT)

usiDataLength: 按字节数填写；1 字节为 16#01；2 字节为 16#02；4 字节为 16#04 等。

◆ 输出变量

输出变量	名称	数据类型	有效范围	初始值	描述
Done	设置操作成功	BOOL	TRUE,FALSE	FALSE	设置操作成功置为 TRUE
Busy	指令正在执行	BOOL	TRUE,FALSE	FALSE	当前指令正在执行中，置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时，置为 TRUE
ErrorID	错误代码	SMC_ERROR	参阅 SMC_ERROR	0	异常发生时，输出错误代码

3) 功能说明

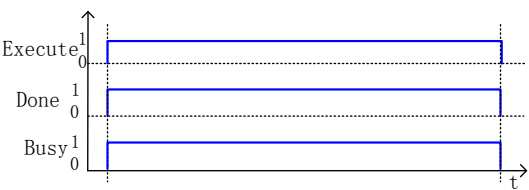
通过 MC\_WriteParameter 设置轴的位参数，指令为 Execute 上升沿触发。指令可以重复多次使用，互不影响。

◆ 时序图

功能块的 Execute 必须为上升沿触发条件；

功能块的 Done 表示设置操作成功；

功能块的 Busy 表示当前功能块正在执行中；

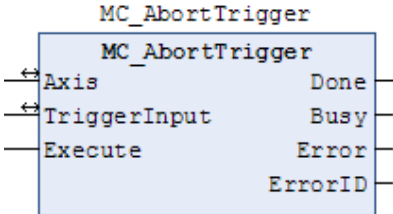




## MC\_AbortTrigger

功能块终止输入锁存相关事件的关联特性，和 MC\_Touchprobe 配套使用。

### 1) 指令格式

指令	名称	图形表现	ST 表现
MC_AbortTrigger	功能块终止事件关联		<pre>MC_AbortTrigger(   Axis:= ,   TriggerInput:= ,   Execute:= ,   Done=&gt; ,   Busy=&gt; ,   Error=&gt; ,   ErrorID=&gt; );</pre>

### 2) 相关变量

#### ◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴，即 AXIS_REF_SM3 的一个实例
TruggerInput	触发信号	TRIIGGER_REF	—	—	触发信号、触发属性等描述

#### ◆ TRIIGGER\_REF 说明：

结构体	元素	数据类型	初始值	描述
TRIIGGER_REF	iTriggerNumber	INT	-1	在驱动器模式下，锁定功能中的哪一个。 0: 探针 1 上升沿锁存 1: 探针 1 下降沿锁存 2: 探针 2 上升沿锁存 3: 探针 2 下降沿锁存 详细参考《IS620N 系列伺服设计维护使用手册》
	bFastLatching	BOOL	TRUE	指定锁存触发的模式： TRUE: 驱动器模式 FALSE: 控制器模式
	bInput	BOOL		当 bFastLatching=FLASE 时，由控制器 Input 信号触发
	bActive	BOOL		触发的有效信号

#### ◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
Execute	执行条件	BOOL	TRUE,FALSE	FALSE	为上升沿操作驱动一次设置操作

#### ◆ 输出变量

输出变量	名称	数据类型	有效范围	初始值	描述
Done	设置操作成功	BOOL	TRUE,FALSE	FALSE	设置操作成功置为 TRUE
Busy	指令正在执行	BOOL	TRUE,FALSE	FALSE	当前指令正在执行中，置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时，置为 TRUE
ErrorID	错误代码	SMC_ERROR	参阅 SMC_ERROR	0	异常发生时，输出错误代码

### 3) 功能说明

通过 MC\_AbortTrigger 功能块是把触发信号或属性和相关的触发指令终止其关联操作。

功能块的 Execute 必须为上升沿触发条件；

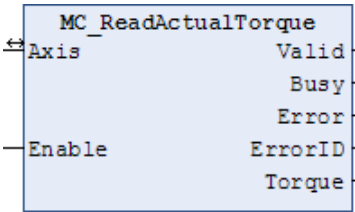
功能块的 Done 表示设置操作成功；

功能块的 Busy 表示当前功能块正在执行中；

# MC\_ReadActualTorque

指令读取驱动器运行的当前力矩值，读取的当前力矩值保存在自己定义的变量单元中。

## 1) 指令格式

指令	名称	图形表现	ST 表现
MC_ReadActualTorque	当前力矩值读取指令		<pre>MC_ReadActualTorque0(     Axis:= ,     Enable:= ,     Valid=&gt; ,     Busy=&gt; ,     Error=&gt; ,     ErrorID=&gt; ,     Torque=&gt; );</pre>

## 2) 相关变量

### ◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴，即 AXIS_REF_SM3 的一个实例

### ◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
Enable	执行条件	BOOL	TRUE,FALSE	FALSE	为 TRUE 状态读取伺服的当前位置

### ◆ 输出变量

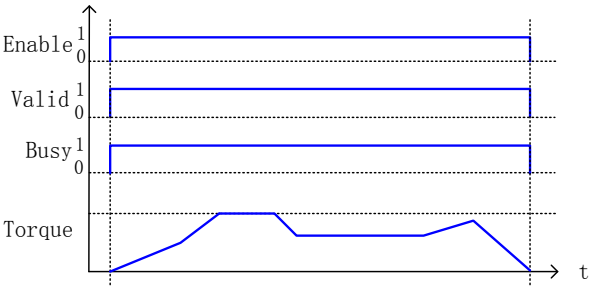
输出变量	名称	数据类型	有效范围	初始值	描述
Valid	当前力矩值可获取标志	BOOL	TRUE,FALSE	FALSE	能正确的获取驱动器的力矩值，置为 TRUE
Busy	指令正在执行	BOOL	TRUE,FALSE	FALSE	当前指令正在执行中，置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时，置为 TRUE
ErrorID	错误代码	SMC_ERROR	参阅 SMC_ERROR	0	异常发生时，输出错误代码
Torque	获取的当前力矩值	LREAL	力矩值	0	指令读出来的当前力矩数据

## 3) 功能说明

通过 MC\_ReadActualTorque 读取驱动器中的当前力矩值指令，指令为 Enable 电平使能效应。指令可以重复多次使用，互不影响。

### ◆ 时序图

- 功能块的 Enable 必须为 TRUE 的条件；
- 功能块的 Valid 表示读出的 Torque 为有效的数据值；
- 功能块的 Busy 表示当前功能块正在执行中；



MC\_ReadActualVelocity

指令读取驱动器运行的当前速度值，读取的当前速度值保存在自己定义的变量单元中。

1) 指令格式

指令	名称	图形表现	ST 表现
MC_ReadActualVelocity	当前速度读取指令		<pre>MC_ReadActualVelocity0(   Axis:= ,   Enable:= ,   Valid=&gt; ,   Busy=&gt; ,   Error=&gt; ,   ErrorID=&gt; ,   Velocity=&gt; );</pre>

2) 相关变量

◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴，即 AXIS_REF_SM3 的一个实例

◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
Enable	执行条件	BOOL	TRUE,FALSE	FALSE	为 TRUE 状态读取当前轴速度

◆ 输出变量

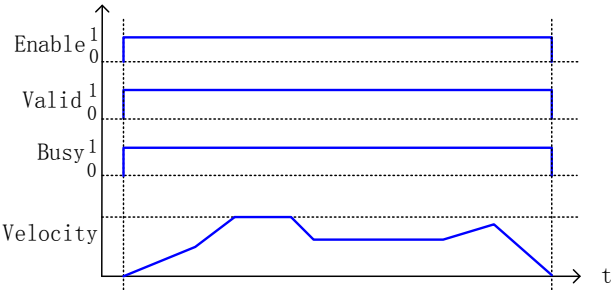
输出变量	名称	数据类型	有效范围	初始值	描述
Valid	当前速度值可获取标志	BOOL	TRUE,FALSE	FALSE	能正确的获取驱动器的速度值，置为 TRUE
Busy	指令正在执行	BOOL	TRUE,FALSE	FALSE	当前指令正在执行中，置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时，置为 TRUE
ErrorID	错误代码	SMC_ERROR	参阅 SMC_ERROR	0	异常发生时，输出错误代码
Velocity	获取的当前速度值	LREAL	速度值	0	指令读出来的当前速度数据

3) 功能说明

通过 MC\_ReadActualVelocity 读取驱动器中的当前速度值指令，指令为 Enable 电平使能效应。指令可以重复多次使用，互不影响。

◆ 时序图

功能块的 Enable 必须为 TRUE 的条件；  
功能块的 Valid 表示读出的 Velocity 为有效的数据值；  
功能块的 Busy 表示当前功能块正在执行中；



MC\_SetPosition

将指令中的位置数据设为当前轴的位置数据，对设置位置数据操作不会产生任何位移移动，用于产生坐标系的位移。

1) 指令格式

指令	名称	图形表现	ST 表现
MC_SetPosition	读取轴的参数		<pre>MC_SetPosition0(     Axis:= ,     Execute:= ,     Position:= ,     Mode:= ,     Done=&gt; ,     Busy=&gt; ,     Error=&gt; ,     ErrorID=&gt; );</pre>

2) 相关变量

◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴，即 AXIS_REF_SM3 的一个实例

◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
Execute	执行条件	BOOL	TRUE,FALSE	FALSE	为上升沿操作驱动一次设置操作
Position	轴位置数据	LREAL		0	位置数据
Mode	设定值	BOOL	TRUE,FALSE	FALSE	位置模式；TRUE：相对位置 (RELATIVE)；FALSE：绝对位置 (ABSOLUTE)；

◆ 输出变量

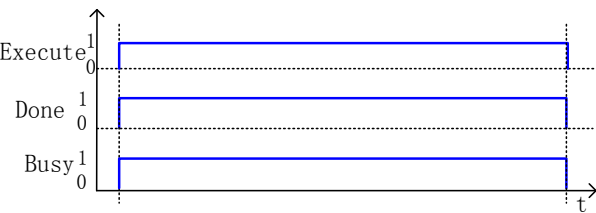
输出变量	名称	数据类型	有效范围	初始值	描述
Done	设置操作成功	BOOL	TRUE,FALSE	FALSE	设置操作成功置为 TRUE
Busy	指令正在执行	BOOL	TRUE,FALSE	FALSE	当前指令正在执行中，置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时，置为 TRUE
ErrorID	错误代码	SMC_ERROR	参阅 SMC_ERROR	0	异常发生时，输出错误代码

3) 功能说明

通过 MC\_SetPosition 设置轴的位置参数，不产生任何位移，但形成了坐标偏移；指令为 Execute 上升沿触发；指令可以重复多次使用，互不影响。

◆ 时序图

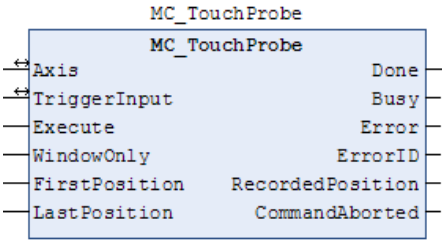
功能块的 Execute 必须为上升沿触发条件；  
功能块的 Done 表示设置操作成功；  
功能块的 Busy 表示当前功能块正在执行中；



## MC\_TouchProbe

指令通过外部信号触发，保存当前轴的位置数据。

### 1) 指令格式

指令	名称	图形表现	ST 表现
MC_TouchProbe	启用外部锁定		<pre>MC_TouchProbe(   Axis:= ,   TriggerInput:= ,   Execute:= ,   WindowOnly:= ,   FirstPosition:= ,   LastPosition:= ,   Done=&gt; ,   Busy=&gt; ,   Error=&gt; ,   ErrorID=&gt; ,   RecordedPosition=&gt; ,   CommandAborted=&gt; );</pre>

### 2) 相关变量

#### ◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴，即 AXIS_REF_SM3 的一个实例
TruggerInput	触发信号	TRIIGGER_REF	—	—	触发信号或触发属性等关联属性

#### ◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
Execute	执行条件	BOOL	TRUE,FALSE	FALSE	为上升沿操作驱动一次设置操作
WindowOnly	触发窗口	BOOL	TRUE,FALSE	FALSE	
FirstPosition	触发开始位置	LREAL	—	0	指定接收触发的开始位置
LastPosition	触发结束位置	LREAL	—	0	指定接收触发的结束位置

#### ◆ 输出变量

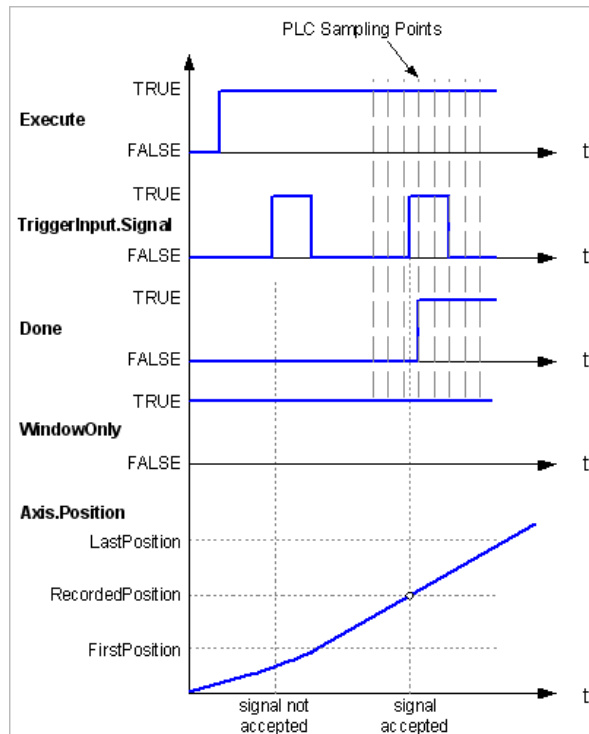
输出变量	名称	数据类型	有效范围	初始值	描述
Done	设置操作成功	BOOL	TRUE,FALSE	FALSE	设置操作成功置为 TRUE
Busy	指令正在执行	BOOL	TRUE,FALSE	FALSE	当前指令正在执行中，置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时，置为 TRUE
ErrorID	错误代码	SMC_ERROR	参阅 SMC_ERROR	0	异常发生时，输出错误代码
RecordedPosition	触发记录位置	LREAL	—	0	触发发生时当前的位置
CommandAbort	指令被中断	BOOL	TRUE,FALSE	FALSE	当前指令被中断，置为 TRUE

### 3) 功能说明

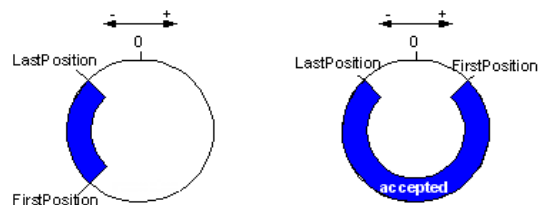
通过 MC\_TouchProbe 功能块 TruggerInput 的信号触发时记录运行轴的当前位置。

Execute 上升沿执行执行

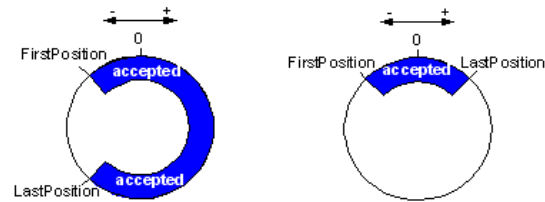
驱动器锁存时：驱动器采集到锁存信号会在记录位置，等到控制器



A. FirstPosition &lt; LastPosition



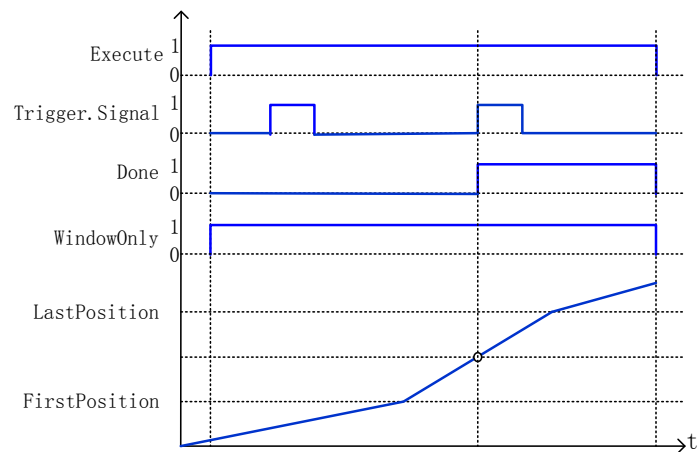
B. FirstPosition &gt; LastPosition



## ◆ 时序图

功能块的 **Execute** 必须为上升沿触发条件；

功能块的 **Done** 表示设置操作成功；



在 **Position** 为循环计数单元，触发信号的位置可以重复使用。

SMC\_MoveContinuousAbsolute

轴按绝对位置连续运行 ( 单位按轴设置 ), 绝对位置由 Distance 指定, 最后的运行速度 EndVelocity 运行;

本指令运行前设置好相关的参数, 加速度 (Acceleration)、减速度 (Deceleration) 和运行速度 (Velocity); 对加速度 (Acceleration) 或减速度 (Deceleration) 的赋值为 0 指令运行错误; 在运行过程中, 一定要关注本指令运行的完整过程, 从用户程序的设计角度避免其他指令中断此指令。

1) 指令格式

指令	名称	图形表现	ST 表现
MC_MoveContinuousAbsolute	轴绝对位置连续控制指令		

2) 相关变量

◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴, 即 AXIS_REF_SM3 的一个实例

◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
Execute	执行条件	BOOL	TRUE,FALSE	FALSE	输入的一个上升沿将启动功能块的处理
Distance	运动相对位置	LREAL	数据范围	0	此数据为运动的相对位置
Velocity	运行速度	LREAL	数据范围	0	轴运行到目标位置的最大速度
EndVelocity	运行结束速度	LREAL	数据范围	0	指令执行完成后的运行速度
EndVelocity-Direction	结束速度的方向	MC_Direction	positive, negative, current;	Current	可以使用: positive, negative, current; 不可使用: shortest, fastest
Acceleration	加速度	LREAL	数据范围	0	速度变大时加速度值
Deceleration	减速度	LREAL	数据范围	0	速度变小时减速度值
Direction	运行方向	shortest	数据范围	shortest	对于线性 / 直线轴: positive, negative; 对于旋转 / 圆周轴: positive, negative, current, shortest, fastest

◆ 输出变量

输出变量	名称	数据类型	有效范围	初始值	描述
InEndVelocity	指令位置到达	BOOL	TRUE,FALSE	FALSE	轴指令执行位置到达, 置为 TRUE
Busy	指令正在执行	BOOL	TRUE,FALSE	FALSE	当前指令正在执行中, 置为 TRUE
CommandAbort	指令被中断	BOOL	TRUE,FALSE	FALSE	当前指令被中断, 置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时, 置为 TRUE
ErrorID	错误代码	SMC_ERROR	参阅 SMC_ERROR	0	异常发生时, 输出错误代码

### 3) 功能说明

本功能块为轴绝对定位指令，**Distance** 数据为轴的绝对位置。

本功能块运行状态为 **Standstill** 中，指令运行时的状态为 **Discrete Motion**，一个完整的运行过程一定要控制轴的不同运动状态。

启动指令为 **Execute** 的上升沿启动，本指令在 **Discrete Motion** 可以重复上升沿有效，每次都可以刷新最新的 **Position** 位置。

**Acceleration** 或 **Deceleration** 为零，指令运行都为异常状态，但轴的状态为 **Discrete Motion**；

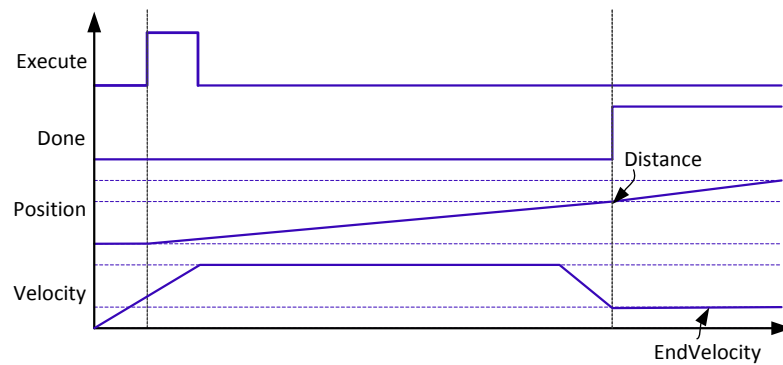
#### ◆ 时序图

轴必须处于 **Standstill** 状态指令才能运行；

功能块的 **Execute** 必须有上升沿的条件；

功能块的 **Done** 表示指令正常执行完成；

功能块的 **Busy** 表示当前功能块正在执行中；





SMC\_MoveContinuousRelative

轴按相对位置连续运行 ( 单位按轴设置 ), 绝对位置由 Distance 指定, 最后的运行速度 EndVelocity 运行;

本指令运行前设置好相关的参数, 加速度 (Acceleration)、减速度 (Deceleration) 和运行速度 (Velocity); 对加速度 (Acceleration) 或减速度 (Deceleration) 的赋值为 0 指令运行错误; 在运行过程中, 一定要关注本指令运行的完整过程, 从用户程序的设计 角度避免其他指令中断此指令。

1) 指令格式

指令	名称	图形表现	ST 表现
MC_MoveContinuousRelative	轴相对定位指令		

2) 相关变量

◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴, 即 AXIS_REF_SM3 的一个实例

◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
Execute	执行条件	BOOL	TRUE,FALSE	FALSE	输入的一个上升沿将启动功能块的处理
Distance	运动相对位置	LREAL	数据范围	0	此数据为运动的相对位置
Velocity	运行速度	LREAL	数据范围	0	轴运行到目标位置的最大速度
EndVelocity	运行结束速度	LREAL	数据范围	0	指令执行完成后的运行速度
EndVelocityDirection	结束速度的方向	MC_Direction	positive, negative, current;	Current	可以使用: positive, negative, current; 不可使用: shortest, fastest
Acceleration	加速度	LREAL	数据范围	0	速度变大时加速度值
Deceleration	减速度	LREAL	数据范围	0	速度变小时减速度值
Direction	运行方向	shortest	数据范围	shortest	对于线性 / 直线轴: positive, negative; 对于旋转 / 圆周轴: positive, negative, current, shortest, fastest

◆ 输出变量

输出变量	名称	数据类型	有效范围	初始值	描述
InEndVelocity	指令位置到达	BOOL	TRUE,FALSE	FALSE	轴指令执行位置到达, 置为 TRUE
Busy	指令正在执行	BOOL	TRUE,FALSE	FALSE	当前指令正在执行中, 置为 TRUE
CommandAbort	指令被中断	BOOL	TRUE,FALSE	FALSE	当前指令被中断, 置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时, 置为 TRUE
ErrorID	错误代码	SMC_ERROR	参阅 SMC_ERROR	0	异常发生时, 输出错误代码

### 3) 功能说明

本功能块运行状态为 **Standstill** 中，指令运行时的状态为 **Discrete Motion**，在指令执行中关注本轴的运行状态，避免打断本轴的其他指令或被其他指令打断本轴的执行。

启动指令为 **Execute** 的上升沿启动，本指令在 **Discrete Motion** 可以重复上升沿有效，每次都可以刷新最新的 **Position** 位置。

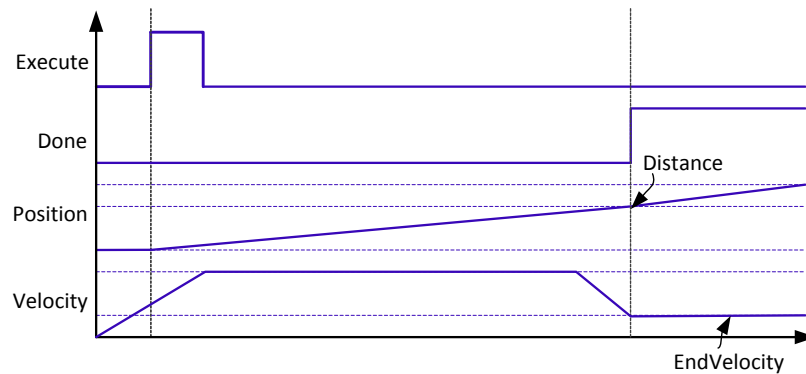
**Acceleration** 或 **Deceleration** 为零，指令运行都为异常状态，但轴的状态为 **Discrete Motion**；

#### ◆ 时序图

功能块的 **Execute** 必须有上升沿的条件；

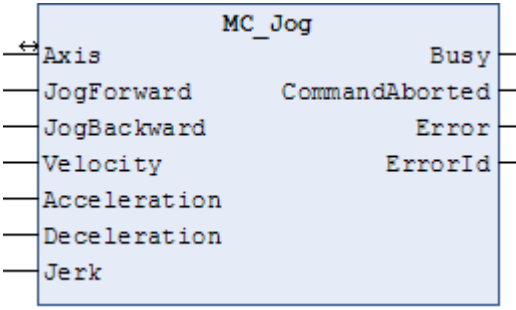
功能块的 **Done** 表示指令正常执行完成；

功能块的 **Busy** 表示当前功能块正在执行中；



MC\_Jog

1) 指令格式

指令	名称	图形表现	ST 表现
MC_Jog	轴点动指令		<pre>MC_Jog(     Axis:= ,     JogForward:= ,     JogBackward:= ,     Velocity:= ,     Acceleration:= ,     Deceleration:= ,     Jerk:= ,     Busy=&gt; ,     CommandAborted=&gt; ,     Error=&gt; ,     ErrorId=&gt; );</pre>

2) 相关变量

◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
JogForward	正向有效	BOOL	TRUE,FALSE	FALSE	设置为 TRUE 则开始正向移动；设置为 FALSE 则停止正向移动
JogBackward	负向有效	BOOL	TRUE,FALSE	FALSE	设置为 TRUE 则开始反向移动；设置为 FALSE 则停止反向移动
Velocity	目标速度	LREAL	正数或“0”	0	指定目标速度。单位：[ 指令单位 /s]
Acceleration	加速度	LREAL	正数或“0”	0	指定加速度。单位：[ 指令单位 /s]
Deceleration	减速度	LREAL	正数或“0”	0	指定减速度。单位：[ 指令单位 /s]

◆ 输出变量

输出变量	名称	数据类型	有效范围	初始值	描述
Busy	执行中	BOOL	TRUE,FALSE	FALSE	接收指令后，置为 TRUE
CommandAborted	执行中断	BOOL	TRUE,FALSE	FALSE	指令中止时，置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时，置为 TRUE
ErrorID	错误代码	SMC_ ERROR	参阅 SMC_ ERROR	0	异常发生时，输出错误代码

◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴，即 AXIS_REF_SM3 的一个实例

3) 功能说明

根据指定 Velocity（目标速度）执行点动运行。

需要正向运行时，将 JogForward（正向运行有效）置为 TRUE；需要反向运行时，将 JogBackward（负向运行有效）置为 TRUE。

同时将 JogForward（正向运行有效）和 JogBackward（负向运行有效）置为 TRUE，将不会有运动发生。

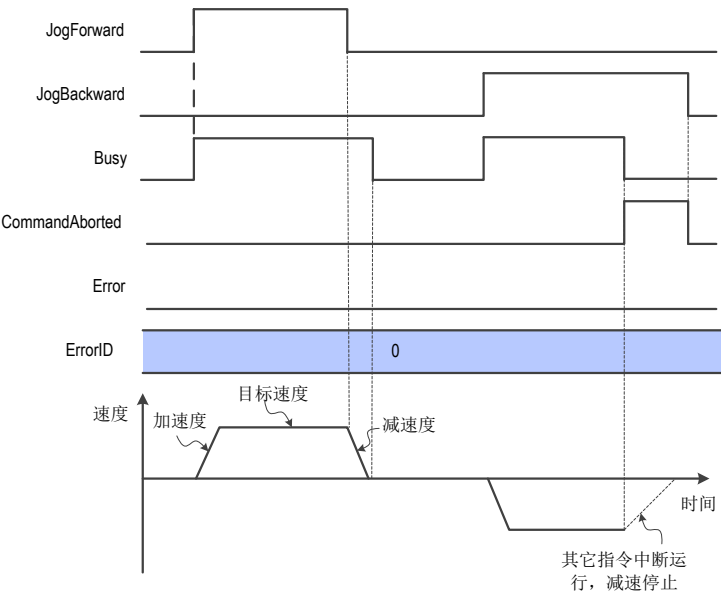
如果 MC\_Jog 指令的指令速度设置值超过轴参数中的点动最高速度，则以点动最高速度执行。

◆ 时序图

在启动 JogForward（正向运行有效）或 JogBackward（负向运行有效）的同时，Busy（执行中）变为 TRUE。

在 JogForward（正向运行有效）或 JogBackward（负向运行有效）的下降沿开始减速并停止轴的同时，Busy（执行中）变为 FALSE。

利用其它指令中止本指令时，CommandAborted（执行中断）变为 TRUE，Busy（执行中）变为 FALSE。

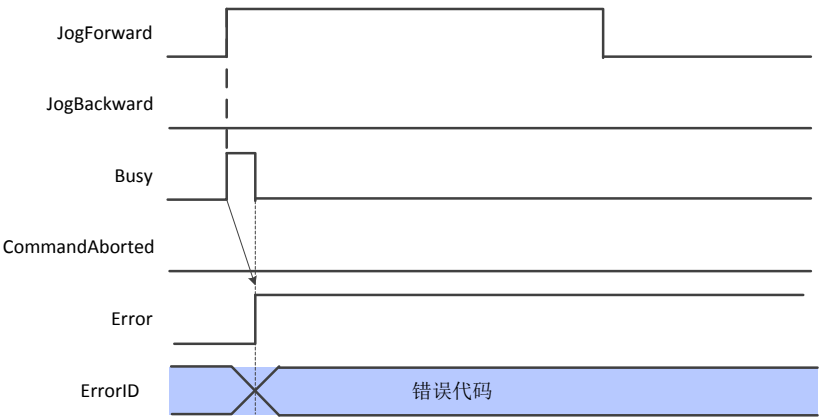


4) 错误说明

在执行本指令中发生异常时，Error（错误）变成 TRUE，轴停止动作。

可查看 ErrorID（错误代码）的输出值，了解发生异常的原因。

◆ 发生异常时的时序图

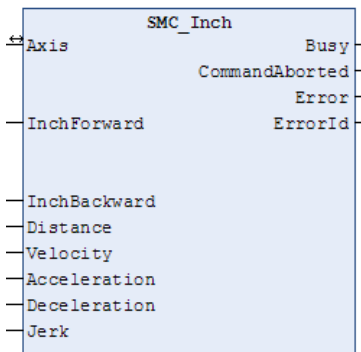


关于指令发生的异常，请阅读“附录 C 错误代码说明”以了解相关错误代码说明。。

SMC\_Inch

轴单步运动控制，通过程序能实现一步接一步的单步控制。

1) 指令格式

指令	名称	图形表现	ST 表现
SMC_Inch	轴相对定位指令		<pre>SMC_Inch0(   Axis:= ,   InchForward:= ,   InchBackward:= ,   Distance:= ,   Velocity:= ,   Acceleration:= ,   Deceleration:= ,   Jerk:= ,   Busy=&gt; ,   CommandAborted=&gt; ,   Error=&gt; ,   ErrorId=&gt; );</pre>

2) 相关变量

◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	—	—	映射到轴，即 AXIS_REF_SM3 的一个实例

◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
InchForward	正向执行	BOOL	TRUE,FALSE	FALSE	如果 InchForward 为 TRUE，轴将按照给定的速度运动 (Velocity, Acceleration, Deceleration) 以正向直到到达距离。输入必须被指定为 FALSE 然后再为 TRUE 再次启动运动。 如果 InchForward 在到达位置之前被设置为 FALSE，那么轴将立即减速到 0 并且 Busy 将会被设置为 FALSE。 如果输入 InchBackward 在仿真情况下被设置为 TRUE，那么将不会有运动产生。
InchBackward	反向执行	BOOL	TRUE,FALSE	FALSE	如果 InchBackward 为 TRUE，轴将会按照给定的速度值进行运动 (Velocity, Acceleration, Deceleration) 以正向运动到设定位置。然后输入必须被设置为 FALSE 然后再设置为 TRUE 启动另一个运动。 如果输入信号 InchForward 同时被设置为 TRUE，那么将不会有轴运动。
Distance	移动的距离	LREAL	数据范围	0	此数据为运动的距离
Velocity	运行速度	LREAL	数据范围	0	轴运行到目标位置的最大速度
Acceleration	加速度	LREAL	数据范围	0	速度变大时加速度值
Deceleration	减速度	LREAL	数据范围	0	速度变小时减速度值

◆ 输出变量

输出变量	名称	数据类型	有效范围	初始值	描述
Busy	指令正在执行	BOOL	TRUE,FALSE	FALSE	当前指令正在执行中，置为 TRUE
CommandAbort	指令被中断	BOOL	TRUE,FALSE	FALSE	当前指令被中断，置为 TRUE
Error	错误	BOOL	TRUE,FALSE	FALSE	异常发生时，置为 TRUE
ErrorID	错误代码	SMC_ ERROR	参阅 SMC_ ERROR	0	异常发生时，输出错误代码

3) 功能说明

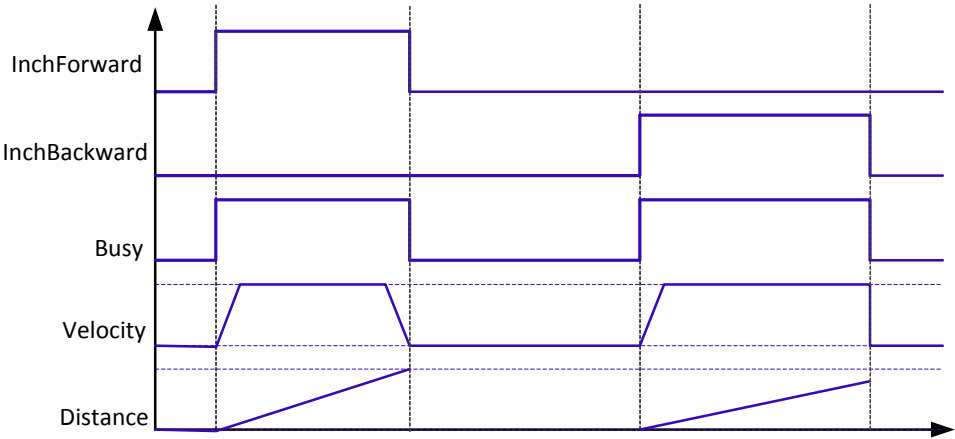
本功能块运行状态为 Standstill 中，指令运行时的状态为 Discrete Motion，在指令执行中关注本轴的运行状态，避免打断本轴的其他指令或被其他指令打断本轴的执行。

Acceleration 或 Deceleration 为零，指令运行都为异常状态，但轴的状态为 Discrete Motion；

◆ 时序图

功能块的 InchForward/ InchBackward 必须有 TRUE/FALSE 的条件；

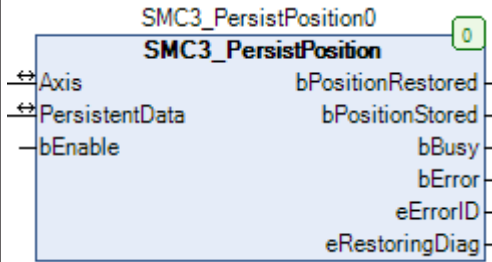
功能块的 Busy 表示当前功能块正在执行中；



## SMC3\_PersistPosition

该指令用来保持纪录实轴绝对值编码器的位置（断电重启控制器后，恢复断电前位置记录值）。如果伺服电机使用的是绝对值编码器，使用该功能块配合使用。

### 1) 指令格式

指令	名称	图形表现	ST 表现
SMC3_PersistPosition	轴位置保持		<pre>SMC3_PersistPosition0(   Axis:=   PersistentData:= ,   bEnable:= ,   bPositionRestored=&gt; ,   bPositionStored=&gt; ,   bBusy=&gt; ,   bError=&gt; ,   eErrorID=&gt; ,   eRestoringDiag=&gt; );</pre>

### 2) 相关变量

#### ◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	-	-	映射到轴，即 AXIS_REF_SM3 的一个实例
PersistentData	保持数据	SMC3_PersistPosition_Data			存储位置信息的断电保持型数据结构

#### ◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
bEnable	执行	BOOL	TRUE,FALSE	FALSE	True 功能块执行，false 不执行功能块 若要在初始化期间还原上次存储的位置， 则必须从应用程序启动时将该值置为 true

#### ◆ 输出变量

输出变量	名称	数据类型	有效范围	初始值	描述
bPositionRestored	位置恢复	BOOL	TRUE,FALSE	FALSE	TRUE, 轴重启后位置恢复完成
bPositionStored	位置保存	BOOL	TRUE,FALSE	FALSE	TRUE, 调用功能块后保存位置完成
bBusy	FB 执行中	BOOL	TRUE,FALSE	FALSE	TRUE, 功能块没有执行完成
bError	错误	BOOL	TRUE,FALSE	FALSE	TRUE, 异常发生
eErrorID	错误代码	SMC_ERROR		SMC_NO_ERROR	异常发生时，输出错误代码
eRestoringDiag	恢复诊断	SMC3_Persist-PositionDiag		SMC3_PersistPositionDiag. SMC3_PPD_RESTORING_OK	位置恢复中的诊断信息 SMC3_PPD_RESTORING_OK: 位置成功恢复 SMC3_PPD_AXIS_PROP_CHANGED: 轴参数有更改，无法恢复位置 SMC3_PPD_DATA_STORED_DURING_WRITING: 功能块从轴参数数据结构复制数据，而不是从 PersistentData 数据中复制。可能原因：非同步性持续变量、控制器崩溃死机

### 3) 功能说明

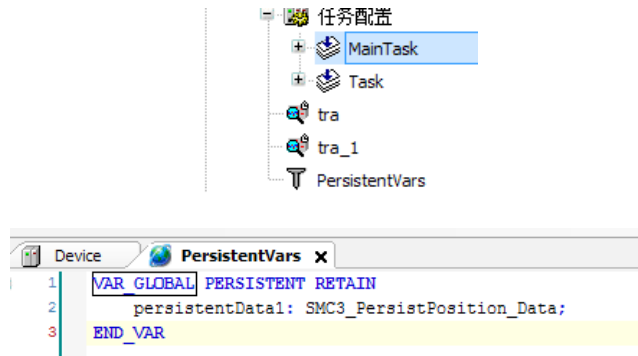
PLC 重启 bEnable 信号为 TRUE，则 bPositionRestroed 输出为 TRUE。

不支持虚轴跟逻辑轴。

在此声明：AM600 里面的轴实际位置为：偏移量 + 编码反馈位置（指令单位 Plus）\* 标尺，绝对值编码器断电所记录的位置为指令单位值。所以 PLC 重启后要恢复断电前“实际位置”需使用该功能块，并且为了记录断电前轴“实际位置”需将 SMC3\_PersistPosition\_Data 配成持续型变量”。

使用方法（实轴编码器为多圈绝对值时）：

#### ④ PersistentVars 中声明 SMC3\_PersistPosition\_Data 型数据



#### ⑤ PLC 主任务（EthCat 任务）中调用

##### ◆ 申明部分：

VAR

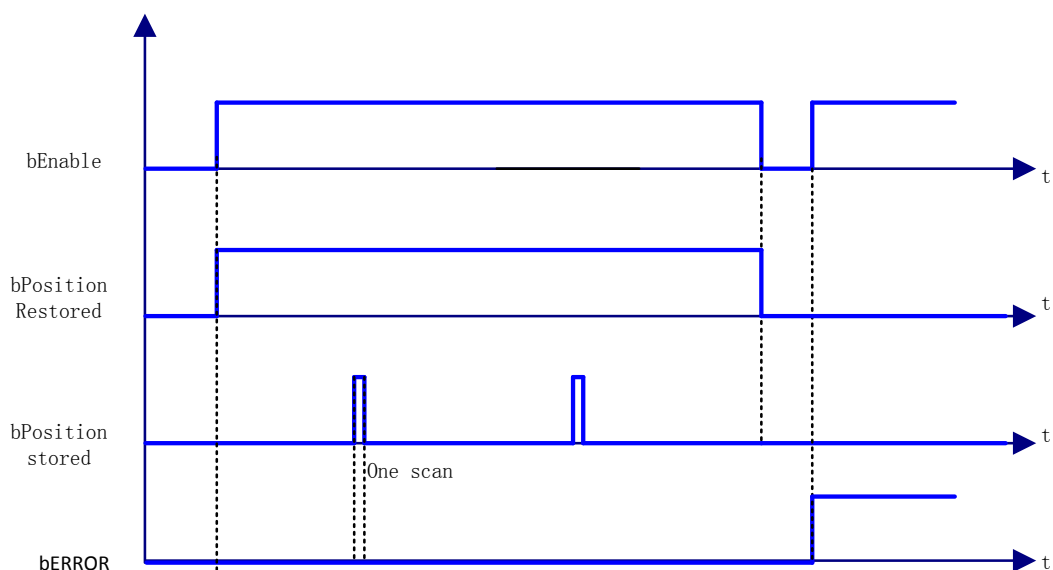
SMC3\_PersistPosition\_1:SMC3\_PersistPosition;

END\_VAR

程序部分：

```
1 //绝对值位置保存  
2 SMC3_PersistPosition_1(Axis:=X_Axis , PersistentData:=persistentData1 ,bEnable:=TRUE );
```

##### ◆ 时序图



### 4) 错误说明

输入轴为虚拟轴或者逻辑轴会导致错误输出；轴有错误时会导致错误输出。

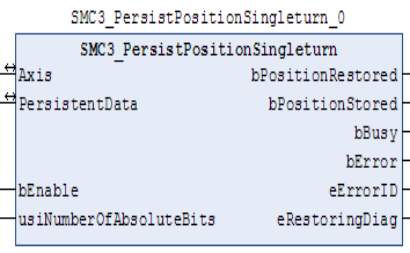
【注意】：请阅读“附录 C 错误代码说明”以了解相关错误代码说明。



## SMC3\_PersistPositionSingleturn

该指令用来保持纪录实轴绝对值编码器（单圈绝对值）的位置（断电重启控制器后，恢复断电前位置记录值）。如果伺服电机使用的是单圈绝对值编码器，使用该功能块配合使用。

### 1) 指令格式

指令	名称	图形表现	ST 表现
SMC3_PersistPositionSingleturn	轴位置保持		<pre>SMC3_PersistPositionSingleturn_0(   Axis:=   PersistentData:= ,   bEnable:= ,   usiNumberOfAbsoluteBits:= ,   bPositionRestored=&gt; ,   bPositionStored=&gt; ,   bBusy=&gt; ,   bError=&gt; ,   eErrorID=&gt; ,   eRestoringDiag=&gt; );</pre>

### 2) 相关变量

#### ◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	-	-	映射到轴，即 AXIS_REF_SM3 的一个实例
PersistentData	保持数据	SMC3_PersistPositionSingleturn_Data			存储位置信息的断电保持型数据结构

#### ◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
bEnable	执行	BOOL	TRUE,FALSE	FALSE	True 功能块执行，false 不执行功能块。若要在初始化期间还原上次存储的位置，则必须从应用程序启动时将该值置为 true
usiNumberOfAbsoluteBitesusiNumb erofAbsoluteBites	位数	UINT		16	多少位的绝对值编码器（如 20 位，24 位编码器等等）

#### ◆ 输出变量

输出变量	名称	数据类型	有效范围	初始值	描述
bPositionRestored	位置恢复	BOOL	TRUE,FALSE	FALSE	TRUE, 轴重启后位置恢复完成
bPositionStored	位置保存	BOOL	TRUE,FALSE	FALSE	TRUE, 调用功能块后保存位置完成
bBusy	FB 执行中	BOOL	TRUE,FALSE	FALSE	TRUE, 功能块没有执行完成
bError	错误	BOOL	TRUE,FALSE	FALSE	TRUE, 异常发生
eErrorID	错误代码	SMC_ERROR		SMC_NO_ERROR	异常发生时，输出错误代码
eRestoringDiag	恢复诊断	SMC3_PersistPositionDiag		SMC3_PersistPositionDiag. SMC3_PPD_RESTORING_OK	位置恢复中的诊断信息 SMC3_PPD_RESTORING_OK: 位置成功恢复 SMC3_PPD_AXIS_PROP_CHANGED: 轴参数有更改，无法恢复位置 SMC3_PPD_DATA_STORED_DURING_WRITING: 功能块从轴参数数据结构复制数据，而不是从 PersistentData 数据中复制。可能原因：非同步性持续变量、控制器崩溃死机

### 3) 功能说明

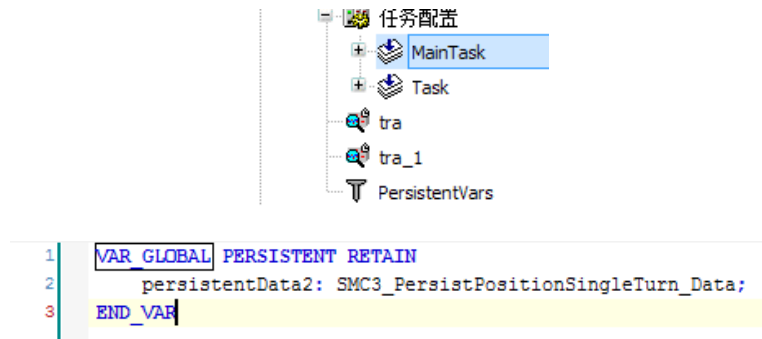
PLC 重启 bEnable 信号为 TRUE，则 bPositionRestored 输出为 TRUE。

不支持虚轴跟逻辑轴。

PLC 重启后要恢复断电前“实际位置”需使用该功能块，并且为了记录断电前轴“实际位置”需将 SMC3\_PersistPositionSingleTurn\_Data 配成持续型变量”。

使用方法（实轴编码器为多圈绝对值时）：

- ① PersistentVars 中声明 SMC3\_PersistPositionSingleTurn\_Data 型数据



- ② PLC 主任务（EthCat 任务）中调用

◆ 申明部分：

VAR

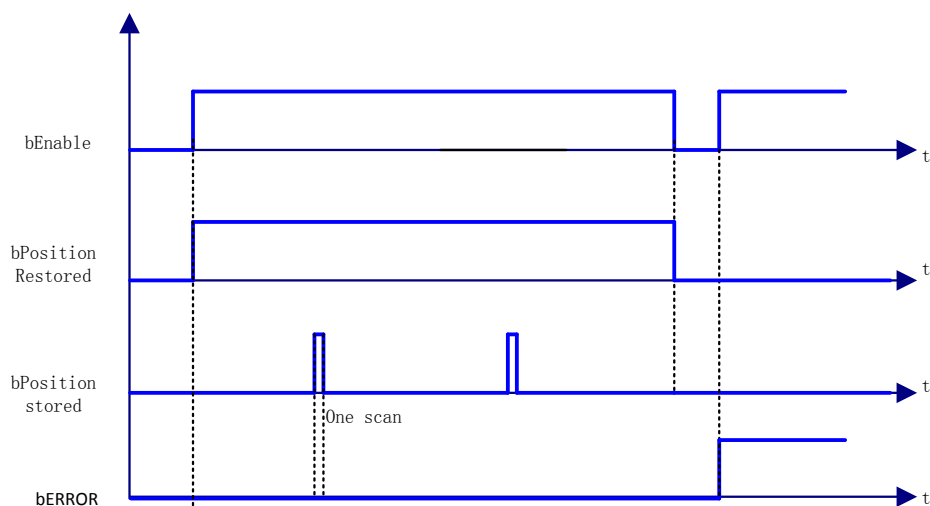
SMC3\_PersistPosition\_2: SMC3\_PersistPositionSingleTurn\_Data;

END\_VAR

◆ 程序部分：

```
//绝对值位置保存
SMC3_PersistPosition_2(Axis:=Y_Axis , PersistentData:=persistentData2 , bEnable:=TRUE );
```

### 4) 时序图



### 5) 错误说明

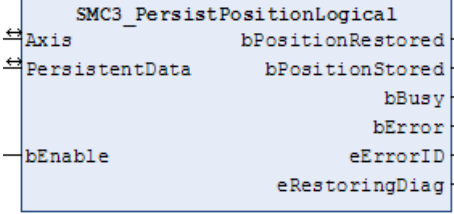
输入轴为虚拟轴或者逻辑轴会导致错误输出；轴有错误时会导致错误输出。

【注意】：请阅读“[附录 C 错误代码说明](#)”以了解相关错误代码说明。

## SMC3\_PersistPositionLogical

该指令用来保持记录逻辑轴（在实轴或者虚轴右键“添加设备”来选择添加逻辑轴）的位置（断电重启控制器后，恢复断电前位置记录值）。

### 1) 指令格式

指令	名称	图形表现	ST 表现
SMC3_PersistPositionLogical	轴位置保持		<pre>SMC3_PersistPositionLogical0(     Axis:= ,     PersistentData:= ,     bEnable:= ,     bPositionRestored=&gt; ,     bPositionStored=&gt; ,     bBusy=&gt; ,     bError=&gt; ,     eErrorID=&gt; ,     eRestoringDiag=&gt; );</pre>

### 2) 相关变量

#### ◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF_LOGICAL_SM3	-	-	映射到轴，即 AXIS_REF_SM3 的一个实例
PersistentData	保持数据	SMC3_PersistPositionLogical_Data			存储位置信息的断电保持型数据结构

#### ◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
bEnable	执行	BOOL	TRUE,FALSE	FALSE	True 功能块执行，false 不执行功能块 若要在初始化期间还原上次存储的位置，则必须从应用程序启动时将该值置为 true

#### ◆ 输出变量

输出变量	名称	数据类型	有效范围	初始值	描述
bPosition Restored	位置恢复	BOOL	TRUE,FALSE	FALSE	TRUE, 轴重启后位置恢复完成
bPosition Stored	位置保存	BOOL	TRUE,FALSE	FALSE	TRUE, 调用功能块后保存位置完成
bBusy	FB 执行中	BOOL	TRUE,FALSE	FALSE	TRUE, 功能块没有执行完成
bError	错误	BOOL	TRUE,FALSE	FALSE	TRUE, 异常发生
eErrorID	错误代码	SMC_ERROR		SMC_NO_ERROR	异常发生时，输出错误代码
eRestoring-Diag	恢复诊断	SMC3_PersistPositionDiag		SMC3_PersistPositionDiag. SMC3_PPD_RESTORING_OK	位置恢复中的诊断信息 SMC3_PPD_RESTORING_OK: 位置成功恢复 SMC3_PPD_AXIS_PROP_CHANGED: 轴参数有更改，无法恢复位置 SMC3_PPD_DATA_STORED_DURING_WRITING: 功能块从轴参数数据结构复制数据，而不是从 PersistentData 数据中复制。 可能原因：非同步性持续变量、控制器崩溃死机

3) 功能说明

PLC 重启 bEnable 信号为 TRUE，则 bPositionRestroed 输出为 TRUE。

不支持虚轴跟实轴。

PLC 重启后要恢复断电前“位置”需使用该功能块，并且为了记录断电前轴“实际位置”需将 SMC3\_PersistPositionLogical\_Data 配成持续型变量”。

◆ 使用方法（实轴编码器为多圈绝对值时）：

PersistentVars 中声明 SMC3\_PersistPositionLogical\_Data 型数据



```
1  VAR_GLOBAL PERSISTENT RETAIN
2  persistentData3: SMC3_PersistPositionLogical_Data;
3  END_VAR
```

PLC 主任务（EthCat 任务）中调用

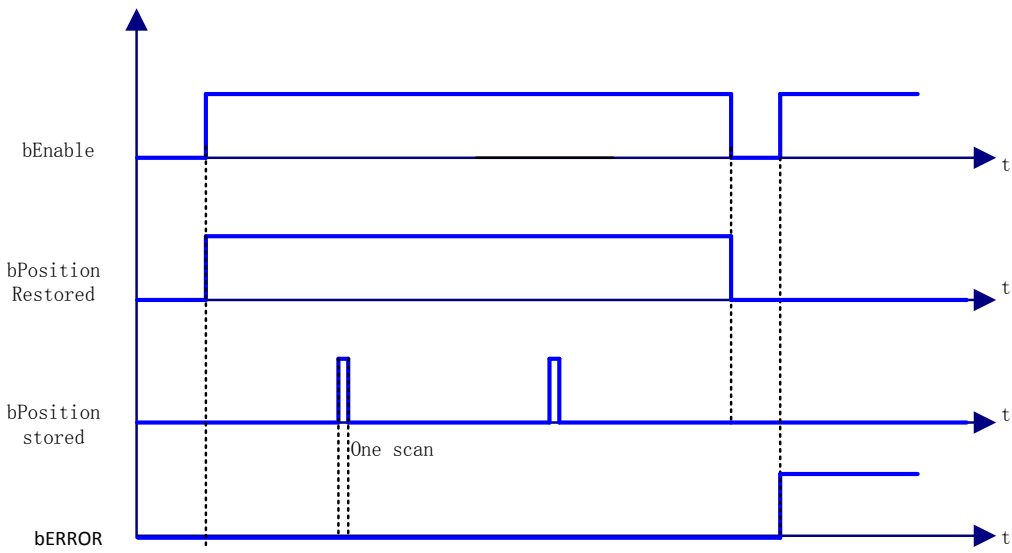
◆ 申明部分：

```
VAR
    SMC3_PersistPosition_3:SMC3_PersistPositionLogical;
END_VAR
```

◆ 程序部分：

```
1  //绝对值位置保存
2  SMC3_PersistPosition_1(Axis:=X_Axis , PersistentData:=persistentData1 ,bEnable:=TRUE );
```

◆ 时序图



4) 错误说明

输入轴为虚拟轴或者实轴会导致错误输出；轴有错误时会导致错误输出。

【注意】：请阅读“附录 C 错误代码说明”以了解相关错误代码说明。

## SMC\_Homing

轴回零指令，与 MC\_Homing 有区别，MC\_Homing 在轴配置处设置回零方式，该指令为控制器控制的回零方式。

### 1) 指令格式

指令	名称	图形表现	ST 表现
SMC_Homing	轴回零		<pre> SMC_Homing0(   Axis:=   ,   bExecute:=   ,   fHomePosition:=   ,   fVelocitySlow:=   ,   fVelocityFast:=   ,   fAcceleration:=   ,   fDeceleration:=   ,   fJerk:=   ,   nDirection:=   ,   bReferenceSwitch:=   ,   fSignalDelay:=   ,   nHomingMode:=   ,   bReturnToZero:=   ,   bIndexOccured:=   ,   fIndexPosition:=   ,   bIgnoreHWLimit:=   ,   bDone=&gt;   ,   bBusy=&gt;   ,   bCommandAborted=&gt;   ,   bError=&gt;   ,   nErrorID=&gt;   ,   bStartLatchingIndex=&gt; );           </pre>

### 2) 相关变量

#### ◆ 输入输出变量

输入输出变量	名称	数据类型	有效范围	初始值	描述
Axis	轴	AXIS_REF	-	-	映射到轴，即 AXIS_REF_SM3 的一个实例

#### ◆ 输入变量

输入变量	名称	数据类型	有效范围	初始值	描述
bExecute	执行	BOOL	TRUE,FALSE	FALSE	True 功能块执行，false 不执行功能块
fHomePosition	原点设定位置	LREAL		0	回零后原点设定位置，单位为用户标定后的单位
fVelocitySlow	慢速	LREAL		0	离开参考开关后慢速设定速度
fVelocityFast	快速	LREAL		0	离开参考开关置位时，快速设定速度
fAcceleration	加速度	LREAL		0	加速度设定值
fDeceleration	减速度	LREAL		0	减速度设定值
fJerk	加速度导数	LREAL		0	Jerk in [u/s³]
nDirection	回零方向	MC_DIRECTION		negative	回零开始方向，参考 MC_DIRECTION
bReferenceSwitch	参考开关	BOOL	TRUE,FALSE	FALSE	连接参考开关，TRUE: 参开开关触发，FALSE: 参考开关闭合

输入变量	名称	数据类型	有效范围	初始值	描述
fSignalDelay	延迟	LREAL		0	参考开关的传输时间，用来补偿死区时间。单位为秒。
nHomingMode	回零模式	SMC_HOMING_MODE			参考 SMC_HOMING_MODE
bReturnTozero	返回零位	BOOL	TRUE,FALSE	FALSE	TRUE: 回零完成后轴运行到位置零（注意：如果 fHomePosition=10，则回零完成后轴位置变为 10，bReturnTozero 为 ture 则回零完成后轴反向走 10 个单位到 0 位）
bIndexOccured		BOOL	TRUE,FALSE	FALSE	True, 标志脉冲记录，回零模式为 FAST_BSLOW_I_S_STOP，FAST_SLOW_I_S_STOP 时生效
fIndexPosition		LREAL		0	标志脉冲时记录的位置
bIgnoreHWLimit	忽略硬限位	BOOL	TRUE,FALSE	FALSE	TRUE, 设定硬件限位开关使能为 false，如果相同的物理开关用于硬件限位开关和参考开关，那么硬件控制将被设置为假

◆ 输出变量

输出变量	名称	数据类型	有效范围	初始值	描述
bDone		BOOL	TRUE,FALSE	FALSE	True, 回零完成
bBusy		BOOL	TRUE,FALSE	FALSE	True, 功能块正在生效中
bCommandAborted		BOOL	TRUE,FALSE	FALSE	True, 功能块被其他动作指令中断
Error		BOOL	TRUE,FALSE	FALSE	True, 错误发生
ErrorID		SMC_ERROR		0	错误代码，枚举型变量，查看帮助 smc_error 查看具体报警代码
bStartLatchingIndex		BOOL	TRUE,FALSE	FALSE	由“bIndexOccured”和“fIndexPosition”共同作用产生

◆ 回零模式（SMC\_HOMING\_MODE）

枚举名称	类型	初始值	描述
FAST_BSLOW_S_STOP	SMC_HOMING_MODE	0	按照设定方向以快速度走向原点开关，碰到原点开关后反向以慢速离开原点开关，离开后先执行 MC_setPosition 将当前位置设为 fHomePosition 设定值，然后执行 MC_stop
FAST_BSLOW_STOP_S	SMC_HOMING_MOD	1	按照设定方向以快速走向原点开关，碰到原点开关后反向以慢速离开原点开关，离开后先执行 MC_stop 将轴停止，然后执行 MC_setPosition，将当前位置设为 fHomePosition 设定值
FAST_BSLOW_I_S_STOP	SMC_HOMING_MOD	2	按照设定方向以快速走向原点开关，碰到原点开关后反向以慢速离开原点开关，bIndexOccured 信号到时先执行 MC_setPosition 再执行 MC_stop
FAST_SLOW_S_STOP	SMC_HOMING_MOD	4	按照设定方向以快速走向原点开关，碰到原点开关后以慢速离开原点开关，离开后先执行 MC_setPosition 将当前位置设为 fHomePosition 设定值，然后执行 MC_stop
FAST_SLOW_STOP_S	SMC_HOMING_MOD	5	按照设定方向以快速走向原点开关，碰到原点开关后以慢速离开原点开关，离开后先执行 MC_stop，然后执行 MC_setPosition 将当前位置设为 fHomePosition 设定值。
FAST_SLOW_I_S_STOP	SMC_HOMING_MOD	6	按照设定方向以快速走向原点开关，碰到原点开关后反向以慢速离开原点开关，bIndexOccured 信号到时先执行 MC_setPosition 再执行 MC_stop

3) 功能说明

SMC\_HOMING 通过 bExecute 的上升沿启动之后，轴将会按照速度 fVelocityFast 并以 nDirection 定义的方向开始运动，直到 bReferenceSwitch = FALSE。然后轴将会缓慢停止并按照相反的方向以速度

fVelocitySlow 离开参考开关。bReferenceSwitch = TRUE 后回零完成，

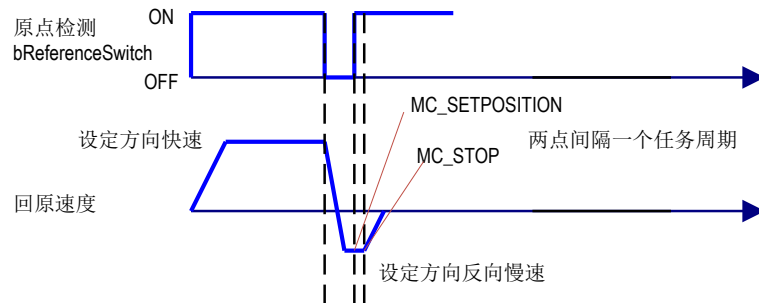
使能回零指令后 bReferenceSwitch 的状态为 ON->OFF->ON, 在 OFF->ON 的上升沿回零完成，设置参考位置。

参考位置 = fHomePosition + ((fSignalDelay \* 1000 + 1 个 DC 时钟周期) / 1000) \* fVelocitySlow 实际就是补偿了设置的 bReferenceSwitch 采样延迟和一个通讯周期位移延迟。

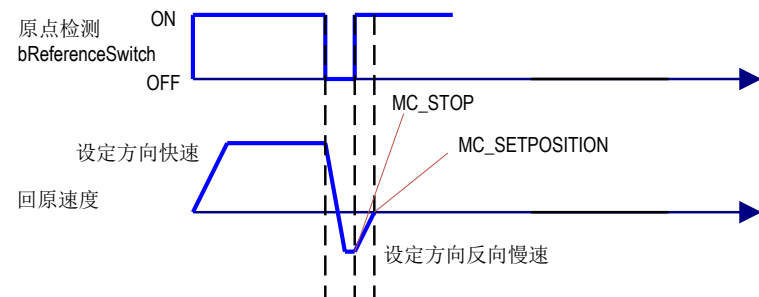
如果 bReturnToZero=TRUE, bReferenceSwitch 的状态在 OFF->ON 的上升沿将参考位置设置为 fHomePosition + ((fSignalDelay \* 1000 + 1 个 DC 时钟周期) / 1000) \* fVelocitySlow, 然后按照速度 fVelocityFast 运行到 0 位置。

注意：Done 完成信号后，轴位置设定为：fHomePosition。设定的时机跟 nHomingMode 有关（详情参考 SMC\_HOMING\_MODE）。下图为几种回零模式：

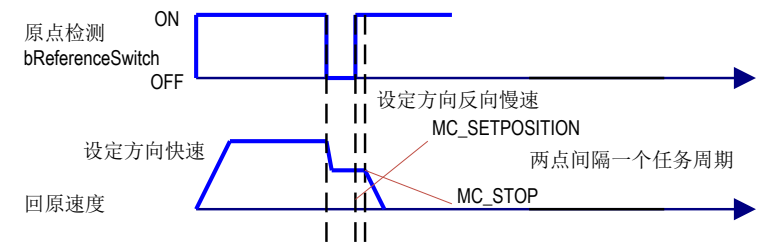
① 回零模式“0”时：



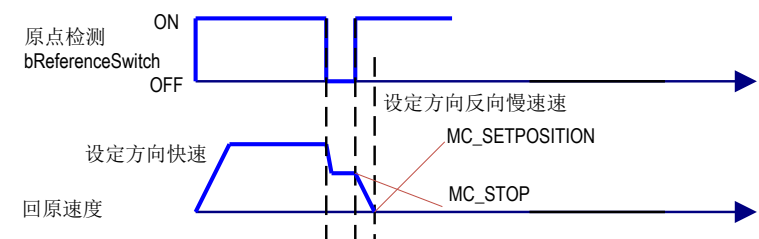
② 回零模式“1”时



③ 回零模式“4”时

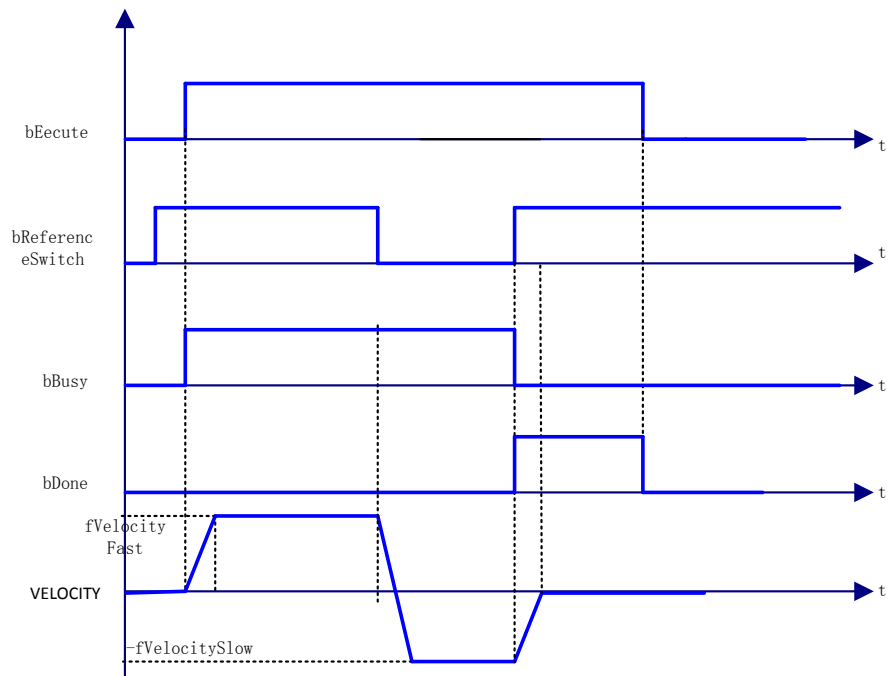


④ 回零模式“5”时

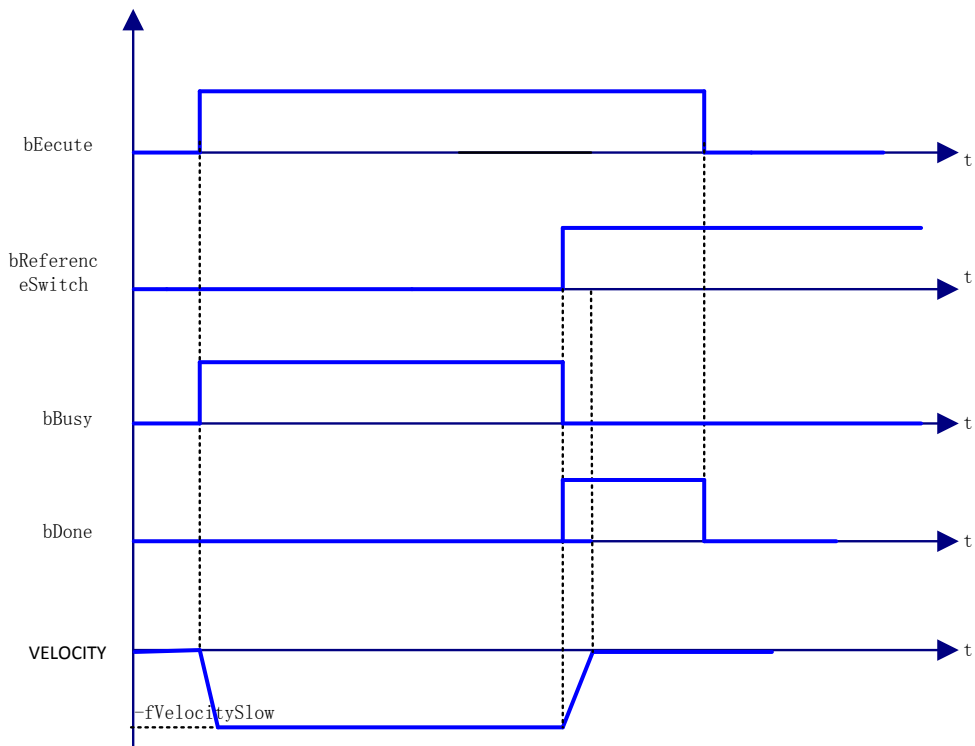


◆ 时序图

① 指令执行时 bReferenceSwitch TRUE 时



② 指令执行时 bReferenceSwitch FALSE 时



4) 错误说明

输入轴类型出错。

轴有错误。

轴没有使能

速度或加速度无效。

【注意】：请阅读“附录 C 错误代码说明”以了解相关错误代码说明。