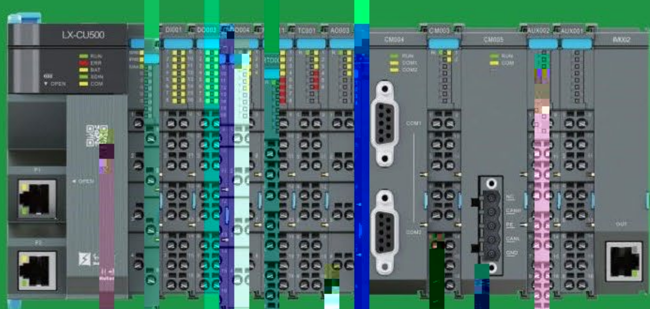


LX

PLCopen

2024.09

HollySys [redacted]

2

100176

010-5898 1588

400-811-1999

010-58981514

010-5898 1558

http://www.hollysys.com/

Email PLC@hollysys.com

http://weibo.com/hollysysplc

1

1.1

LX PLCopen

1.2

1.3

LX PLC

1.4

1.5

1.5.1

“ ” “ ” “ ” “ ”

1.5.2

2

2.1

LX	LX	
LX	LX	LX
LX	FA-AutoThink	FA-AutoThink
LX	LX	
LX	LX	

2.2

•

3

V1.0	2024 9	

4

4.1

	SMC_AxisInit		
	MC_Power		
	MC_Home		
	MC_Stop		
	MC_MoveJog	Jog	/
	MC_MoveAbsolute		
	MC_MoveRelative		
	MC_MoveVelocity		
	SMC_SyncMoveVelocity		
	MC_TorqueControl		
	SMC_SetTorqueLimit		/
	MC_SetPosition		
	MC_SetOverride		
	SMC_SetDynamicLimits		
	MC_ReadDigitalInput		
	MC_ReadActualPosition		
	MC_ReadActualVelocity		
	MC_ReadActualTorque		
	MC_ReadStatus		PLCopen

	MC_ReadAxisInfo		
	MC_ReadAxisError		
	SMC_ReadAxisLimitStatus		
	MC_Reset		
	MC_TouchProbe		
	MC_AbortTrigger		MC_TouchProbe
	SMC_CamIn		
	MC_GearIn		
	SMC_MoveLink	/	
	MC_GroupEnable		
	MC_GroupDisable		
	MC_GroupStop		
	SMC_MoveLinearAbsolute2D	2	2
	SMC_MoveLinearRelative2D	2	2
	SMC_MoveCircularRelative2D	2	2
	MC_ GroupSetOverride		
	SMC_GroupSetDynamicLimits		
	SMC_GroupCornerSpeedLimits		
	SMC_GroupCircularSpeedLimits		

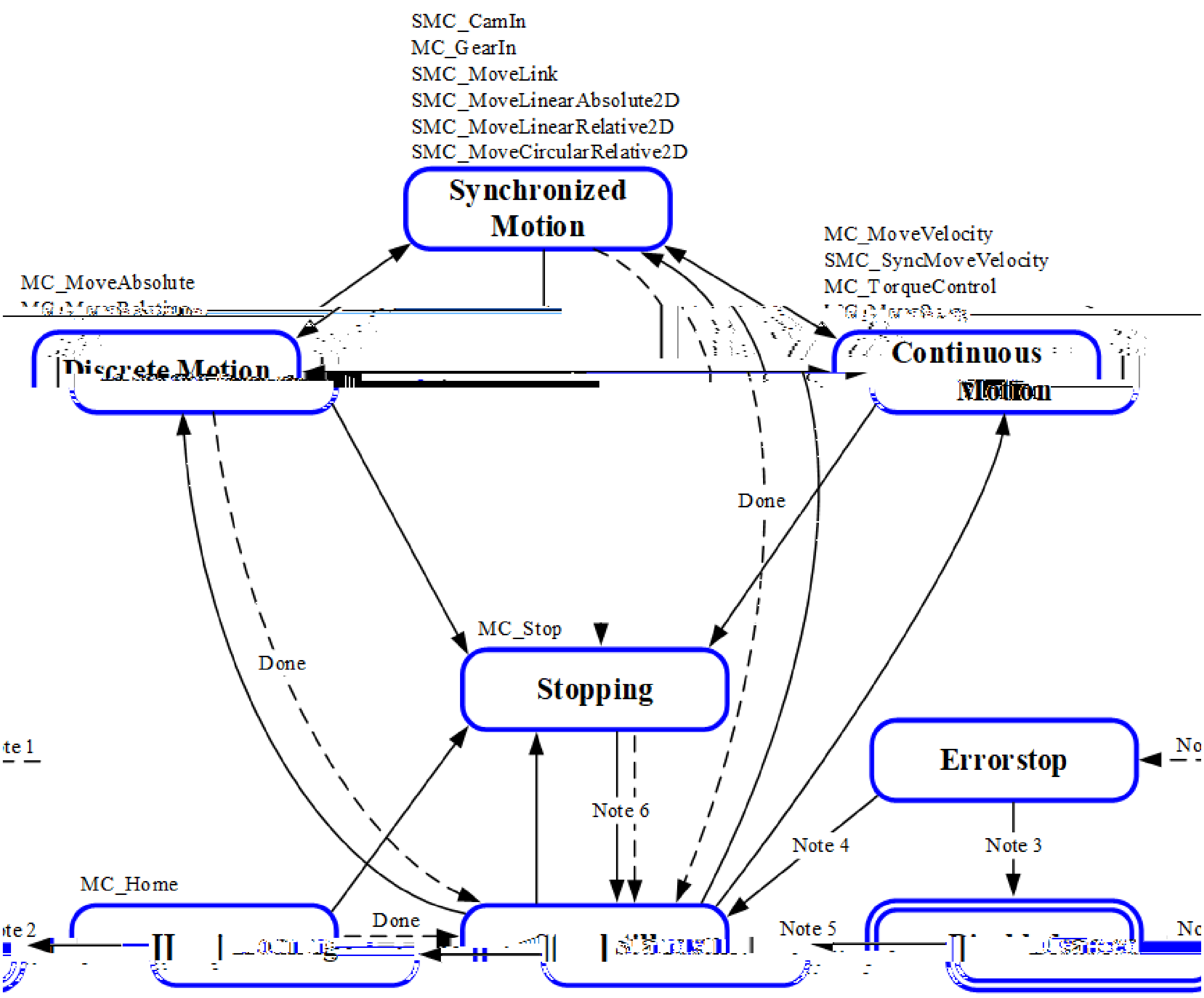
4.2

FB/FUN	FB FUN FB FB FUN
	LD
ST	ST
	LD ST

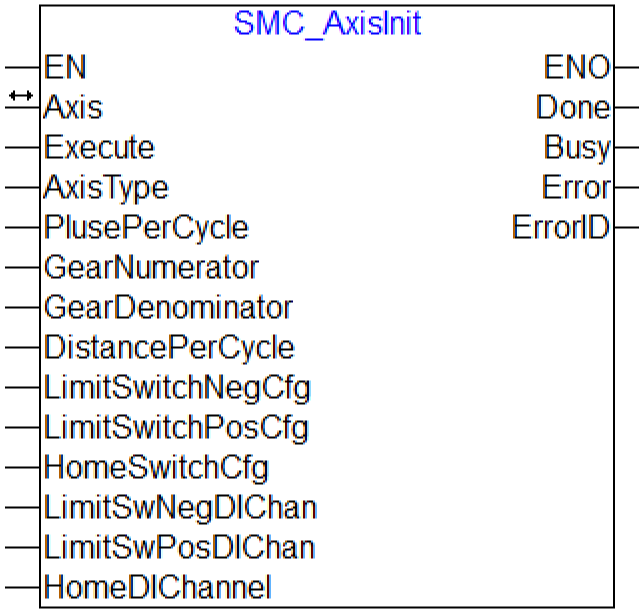
5

5.1 PLcopen

PLCopen



Disabled	



5.2.1

IN_OUT	Axis			-	-	AXIS_REF
INPUT	Execute			-	TRUE/FALSE	BOOL
	AxisType			-	-1 Unused 0 Virtual 50 EtherCAT_Position 51 EtherCAT_Speed 52 EtherCAT_Torque 53 EtherCAT_Encoder	EMAXISTYPE
	PlusePerCycle	/		-		LREAL
	GearNumerator			1		UDINT
	GearDenominator			1		UDINT

	DistancePerCycle	1:1 / 1:1		-		LREAL
	LimitSwitchNegCfg			-1	-1 0 DI 1 DI 10 DI 20 DI	INT
	LimitSwitchPosCfg			-1	-1 0 DI 1 DI 10 DI 20 DI	INT
	HomeSwitchCfg			0	-1 0 DI 10 DI	INT
	LimitSwNegDIChan	DI PIN Digital inputs 0x60FD bit PDO 0x60FD DI DI I %IX100.1 I bit 100×8 + 1		0		UDINT
	LimitSwPosDIChan			1		UDINT
	HomeDIChannel			2		UDINT

OUTPUT	Done			FALSE	TRUE/FALSE	BOOL
	Busy			FALSE	TRUE/FALSE	BOOL
	Error			FALSE	TRUE/FALSE	BOOL
	ErrorID			0	-	SMC_ERROR

5.2.2

Execute

Done TRUE Error

TRUE ErrorID

AxisType LX

Unused -1

Virtual 0

EtherCAT_Position 50 EtherCAT

EtherCAT_Speed 51 EtherCAT

EtherCAT_Torque 52 EtherCAT

EtherCAT_Encoder 53 EtherCAT

EtherCAT_Encoder

RepMode

RepMode=2

RepMode=0

PlusePerCycle / (1) pulse/rev

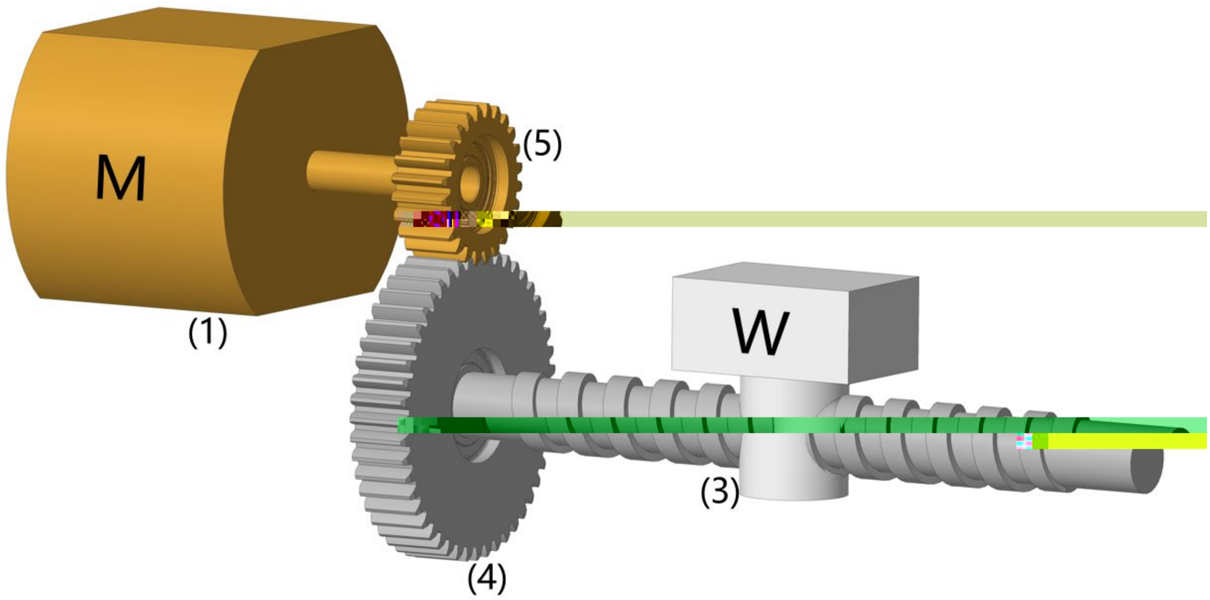
GearNumerator (4)

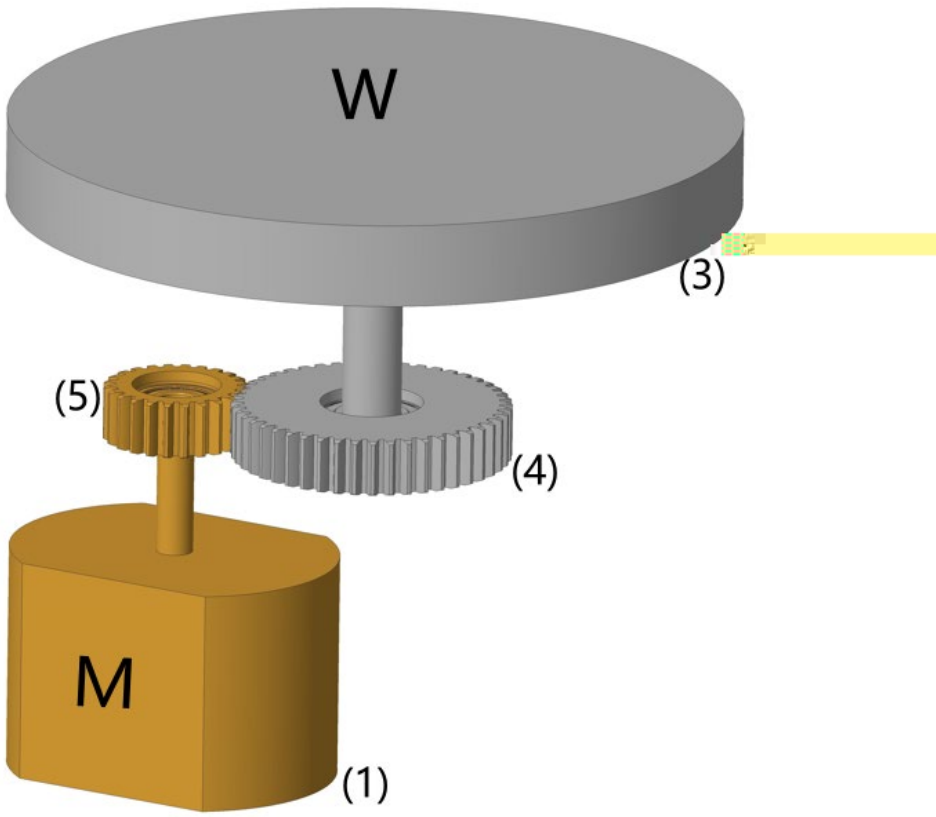
GearDenominator (5)

DistancePerCycle (2)

/rev DistancePerCycle

(3) /rev




W
(3)
(5)
(4)
M
(1)

$$脉冲数[pulse] = \frac{(1)PlusePerCycle[LREAL] * (4)GearNumerator[UDINT]}{(3)DistancePerCycle[LREAL] * (5)GearDenominator[UDINT]} * 工作台移动量[用户单位]$$

LimitSwitchPosCfg/LimitSwitchNegCfg

LimitSwNegDIChan/LimitSwPosDIChan

“ DI” DI

Digital inputs 0x60FD bit PDO 0x60FD

DI 1 0x60FD bit 1

“ DI” DI DI I

DI DI %IX4.1 I

4*8+1=33 33

1

0

DI DI

DI DI

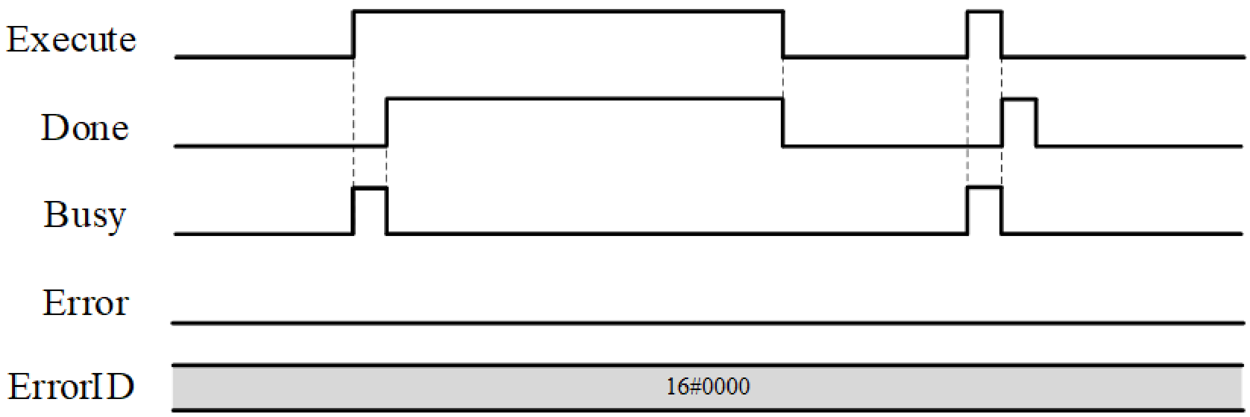
DI Digtial Inputs I DI

Digtial Inputs bit

HomeSwitchCfg HomeDIChannel

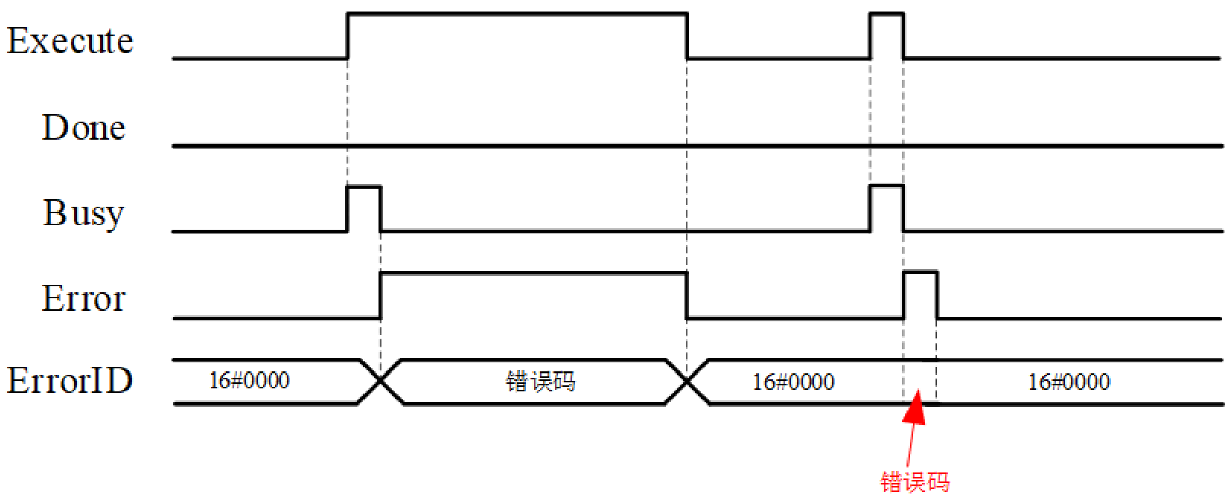
1 Execute Execute

Done



2 Execute Execute

Error ErrorID



5.2.3

SMC_AxisInit

1 EtherCAT_Position

2 8388608 10000

3 10- DI

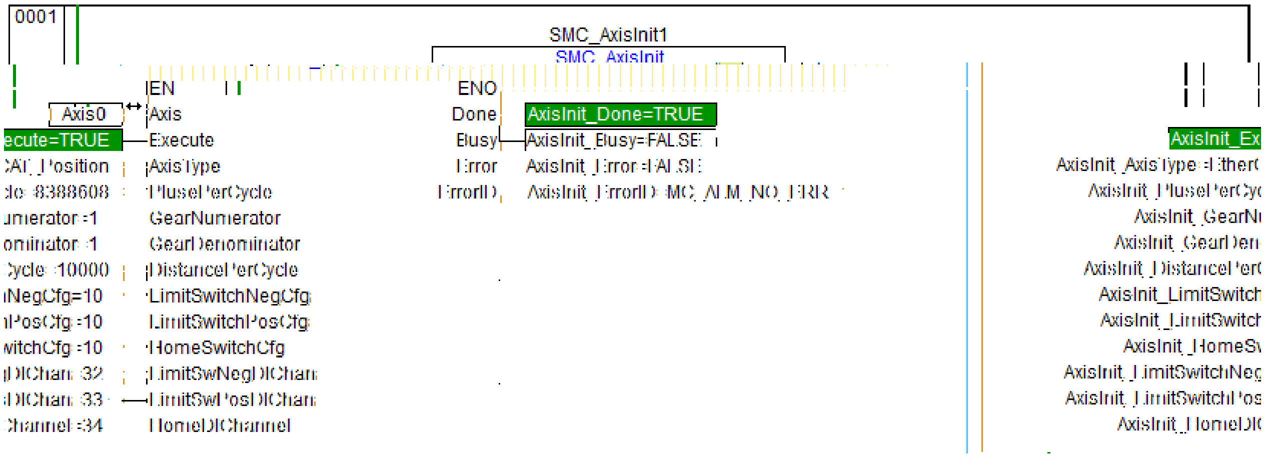
4 32 I %IX4.0(4*8=32)

33(4) I %IX4.1 34 I %IX4.2

SMC_AxisInit

1	SMC_AxisInit1	SMC_AxisInit	-	
2	AxisInit_Execute	BOOL	FALSE	
3	AxisInit_AxisType	EMAXISTYPE	EtherCAT_Position	
4	AxisInit_PlusePerCycle	LREAL	8388608	
5	AxisInit_GearNumerator	UDINT	1	
6	AxisInit_GearDenominator	UDINT	1	
7	AxisInit_DistancePerCycle	LREAL	10000	
8	AxisInit_LimitSwitchNegCfg	INT	10	
9	AxisInit_LimitSwitchPosCfg	INT	10	
10	AxisInit_HomeSwitchCfg	INT	10	
11	AxisInit_LimitSwitchNegDIChan	UDINT	32	DI
12	AxisInit_LimitSwitchPosDIChan	UDINT	33	DI
13	AxisInit_HomeDIChannel	UDINT	34	DI
14	AxisInit_Done	BOOL	FALSE	
15	AxisInit_Busy	BOOL	FALSE	
16	AxisInit_Error	BOOL	FALSE	
17	AxisInit_ErrorID	SMC_ERROR	MC_ALM_NO_ERR	ID

LD



• MC_Power

ST

```
SMC_AxisInit1(
Execute := AxisInit_Execute,
AxisType := AxisInit_AxisType,
PlusePerCycle := AxisInit_PlusePerCycle,
GearNumerator := AxisInit_GearNumerator,
GearDenominator := AxisInit_GearDenominator,
DistancePerCycle := AxisInit_DistancePerCycle,
LimitSwitchNegCfg := AxisInit_LimitSwitchNegCfg,
LimitSwitchPosCfg := AxisInit_LimitSwitchPosCfg,
HomeSwitchCfg := AxisInit_HomeSwitchCfg,
LimitSwNegDIChan := AxisInit_LimitSwitchNegDIChan,
LimitSwPosDIChan := AxisInit_LimitSwitchPosDIChan,
HomeDIChannel := AxisInit_HomeDIChannel,
Axis := Axis0,
Done=>AxisInit_Done,
Busy=>AxisInit_Busy,
Error=>AxisInit_Error,
```

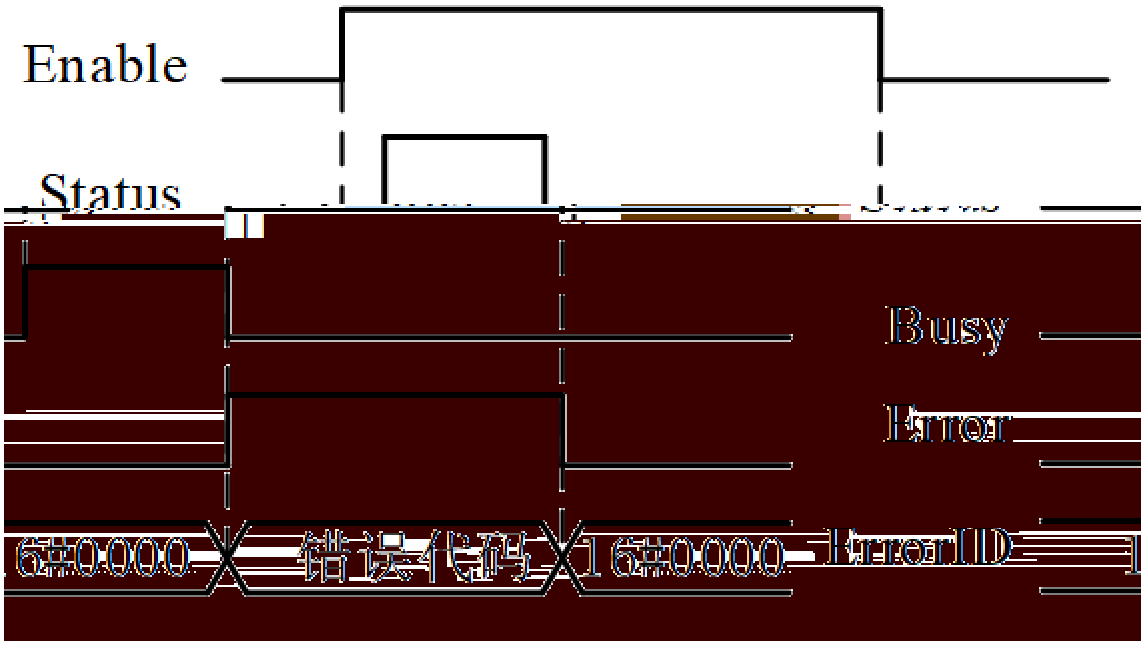
```
ErrorID=>AxisInit_ErrorID);
```

5.2.4

PlusePerCycle GearNumerator

GearDenominator DistancePerCycle LimitSwitchNegCfg LimitSwNegDIChan

LimitSwitchPosCfg LimitSwPosDIChan (i)3.1 (2 >>BDC 5)-241002 T22 >1>B2s>>BDC 5



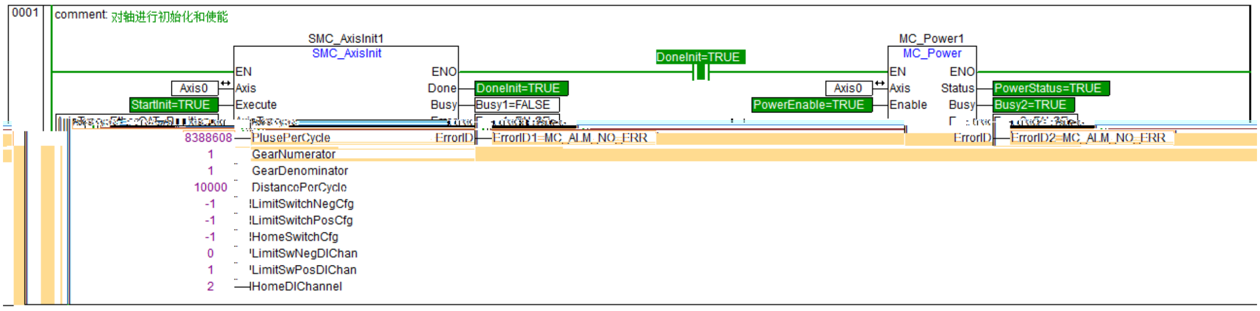
5.3.3

MC_Power

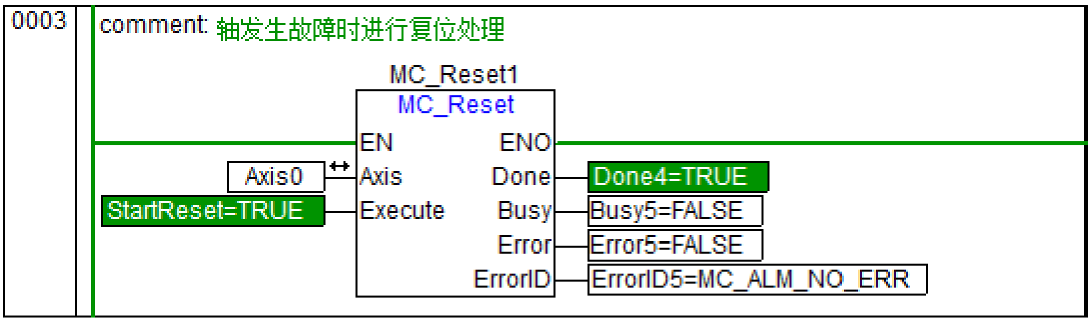
Axis0	AXIS_REF		0
StartInit	BOOL	FALSE	TRUE
DoneInit	BOOL	FALSE	TRUE
PowerEnable	BOOL	FALSE	TRUE
PowerStatus	BOOL	FALSE	TRUE
StartReset	BOOL	FALSE	TRUE

1

ON



2 MC_Reset

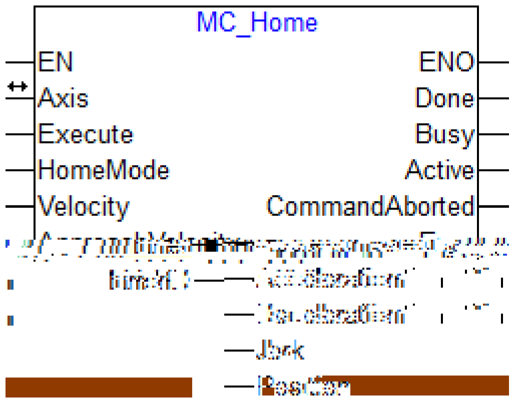


ST

1

```
(*                *)
SMC_AxisInit1(
Execute := StartInit,
AxisType := 50,
PlusePerCycle := 10000,
GearNumerator := 1,
GearDenominator := 1,
DistancePerCycle := 10000,
LimitSwitchNegCfg := -1,
LimitSwitchPosCfg := -1,
HomeSwitchCfg := -1,
LimitSwNegDIChan := 0,
LimitSwPosDIChan := 1,
HomeDIChannel := 2,
Axis := Axis0,
Done=> DoneInit,
Busy=>Busy1,
Error=>Error1,
```

5.4 MC_Home



5.4.1

IN_OUT	Axis			-	-	AXIS_REF
INPUT	Execute			-	TRUE/FALSE	BOOL
	HomeMode			-	CIA402 1/2 3/5 7/11 17/18 19/21 23/27 33/34 35	SMC_HOME_MODE
	Velocity			-		LREAL
	ApproachVelocity			-		LREAL
	Acceleration			-		LREAL
	Deceleration			-		LREAL
	Jerk	0 S		0	0/	LREAL
	Position	Position		0	/ /0	LREAL
OUTPUT	Done			FALSE	TRUE/FALSE	BOOL
	Busy			FALSE	TRUE/FALSE	BOOL

	Active	TRUE		FALSE	TRUE/FALSE	BOOL
	CommandAborted			FALSE	TRUE/FALSE	BOOL
	Error			FALSE	TRUE/FALSE	BOOL
	ErrorID			0	-	SMC_ERROR

5.4.2

Z

HomeMode CIA402 1/2 3/5 7/11 17/18

19/21 23/27 33/34 35 15

SMC_AxisInit

DI DI SMC_AxisInit

MC_Power

Z

Execute

Execute=TRUE Execute=FALSE MC_Home

Homing HomeMode Velocity

ApproachVelocity ApproachVelocity Velocity

Acceleration Deceleration Jerk S

Position Busy=TRUE Home

Active=TRUE Done=TRUE

Homing Stopping MC_Stop MC_Home

CommandAboarted=TRUE MC_ReadAxisInfo IsHomed

MC_Home Home

1		N-OT Z	Z	Z Z
2		P-OT Z	Z	Z Z
3		HMSW Z	Z	Z Z
5		HMSW Z	Z	Z Z
7		HMSW P-OT Z	Z	Z Z
11		HMSW N-OT Z	Z	Z Z
17		N-OT	N-OT↓	
18		P-OT	P-OT↓	
19		HMSW	HMSW↓	
21		HMSW	HMSW↓	
23		HMSW P-OT	HMSW↓	
27		HMSW N-OT	HMSW↓	
33		Z	Z	Z
34		Z	Z	Z

35

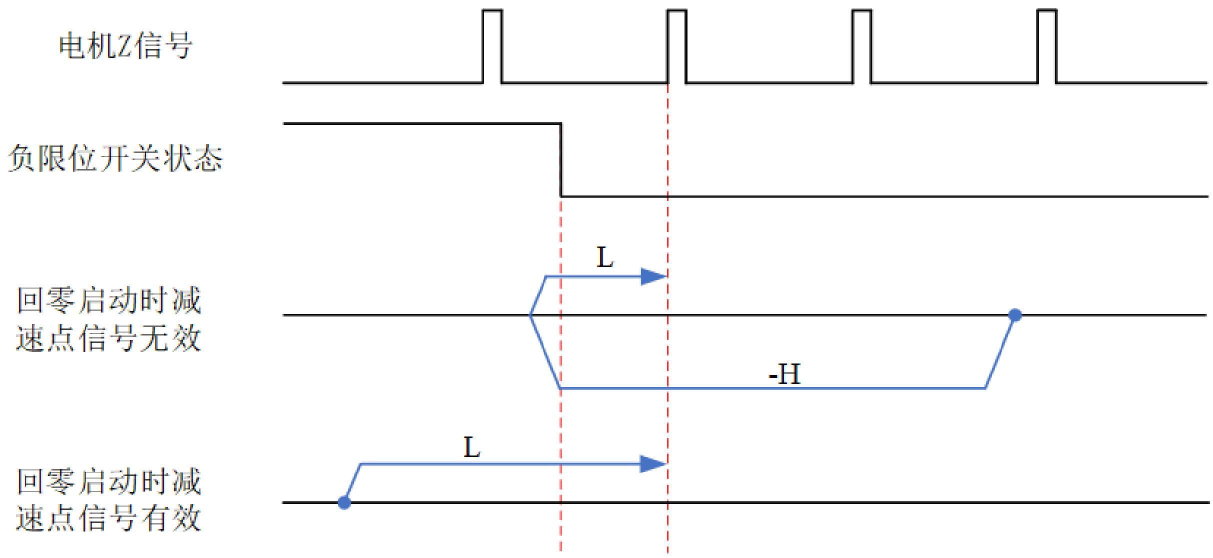
1

Z

2

Z

1



• “H” Velocity “L” ApproachVelocity “-”

2 2 Z

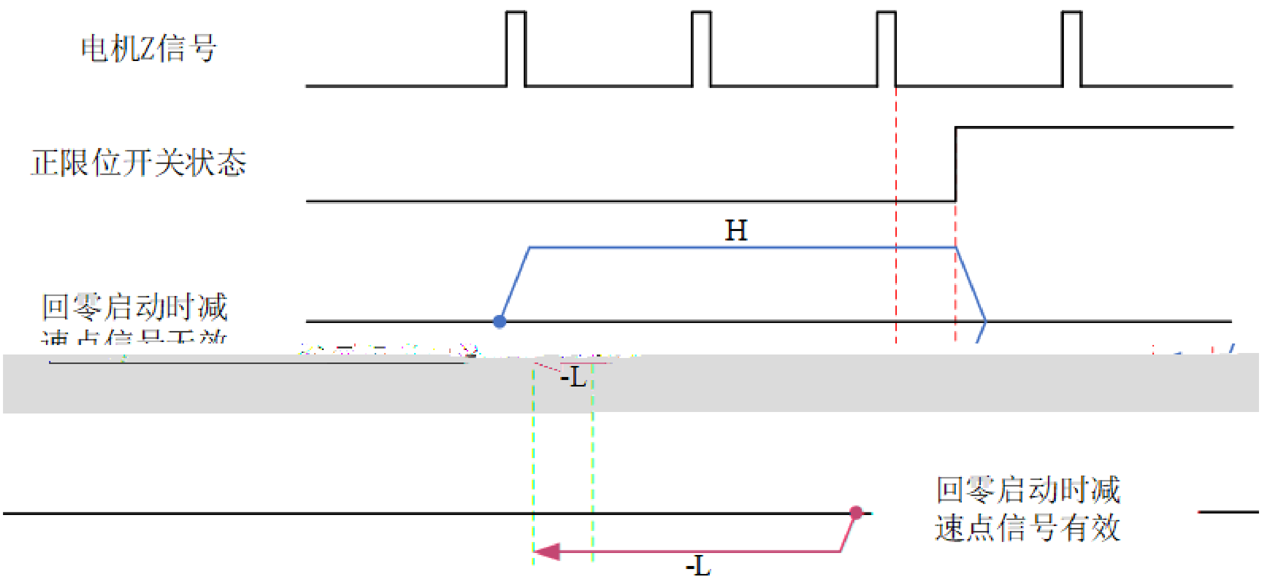
1:

Z

2:

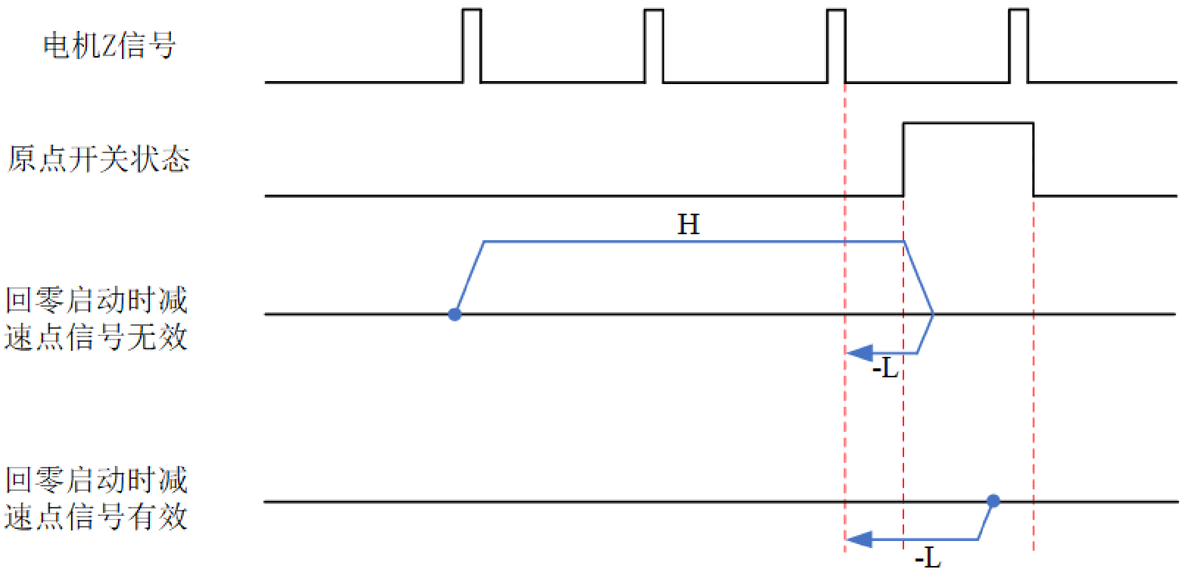
Z

2



3 3 Z

1



4 5 Z

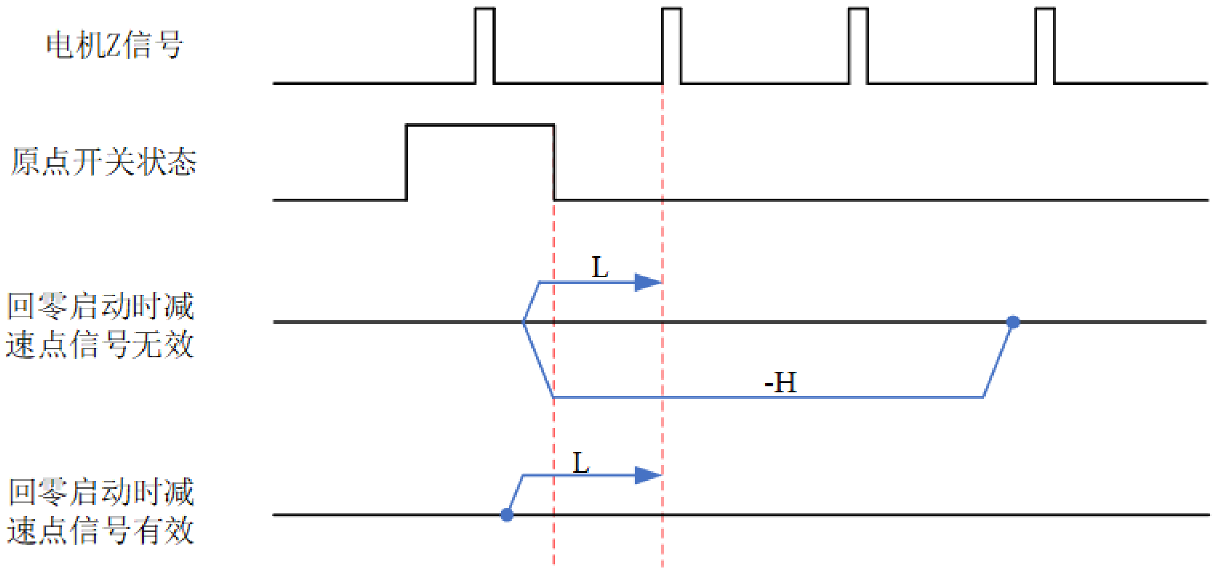
1

Z

2

Z

5



5 7 Z

1

Z

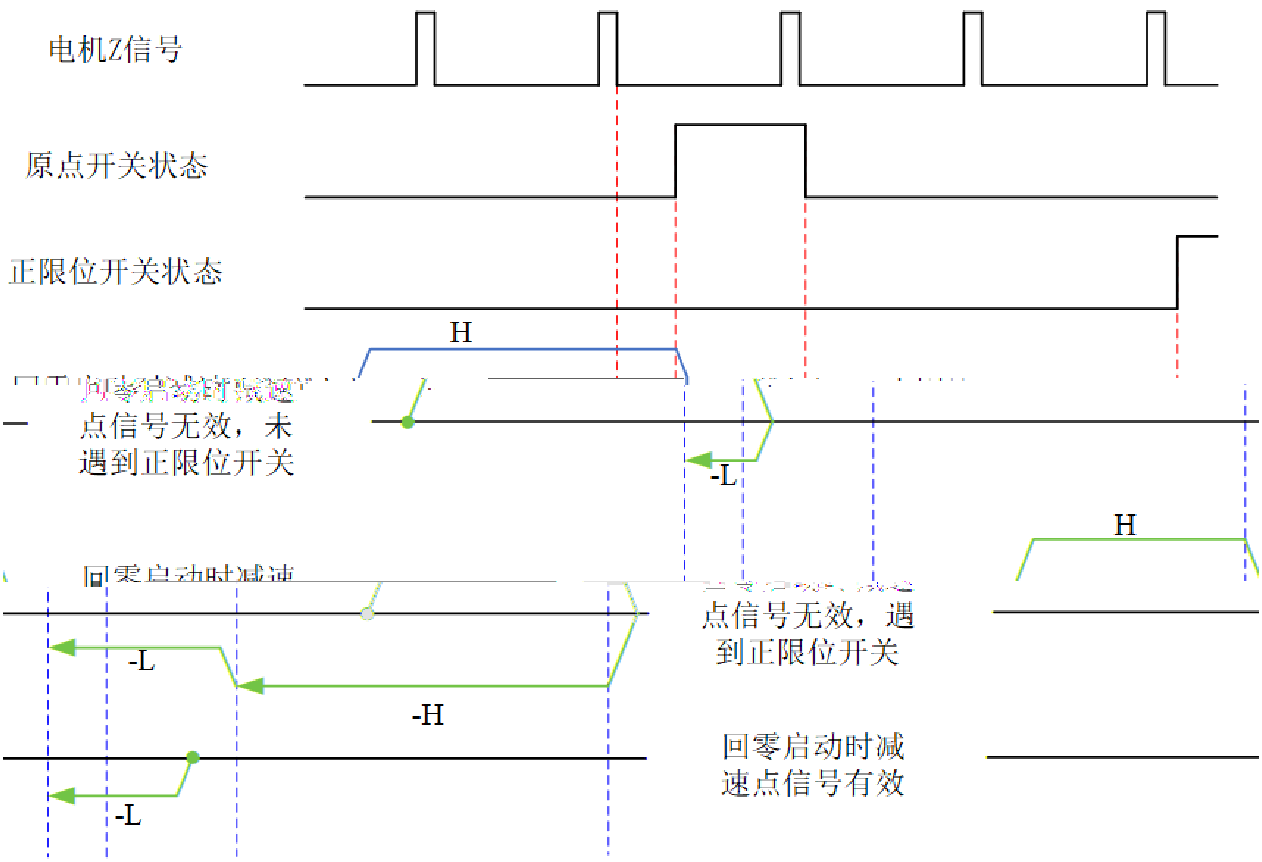
2

Z X.”

Z

3

Z ƀ


电机Z信号
原点开关状态
正限位开关状态
H
点信号无效，未
遇到正限位开关
-L
H
回零启动时减速
点信号无效，遇
到正限位开关
-L
-H
回零启动时减
速点信号有效
-L

6 11 Z

1

Z

2

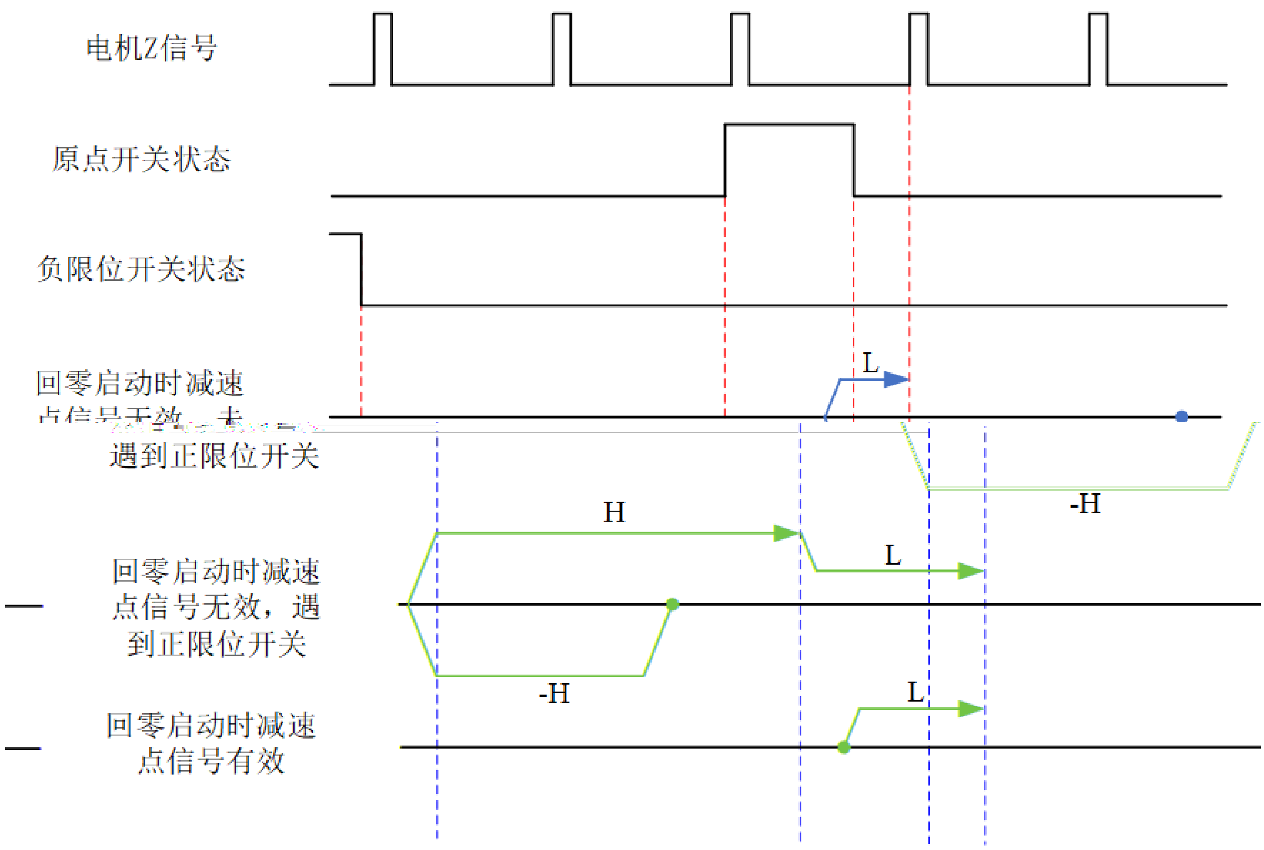
Z

3

Z

;

11

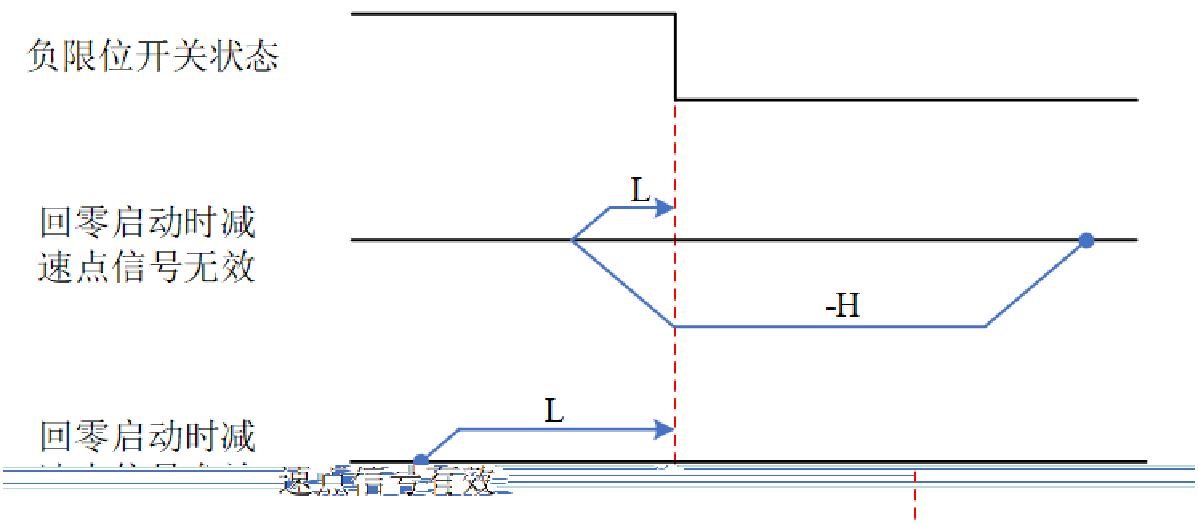

电机Z信号
原点开关状态
负限位开关状态
回零启动时减速
遇到正限位开关
L
H
L
-H
回零启动时减速点信号无效，遇到正限位开关
-H
L
回零启动时减速点信号有效

7 17

1

2

17

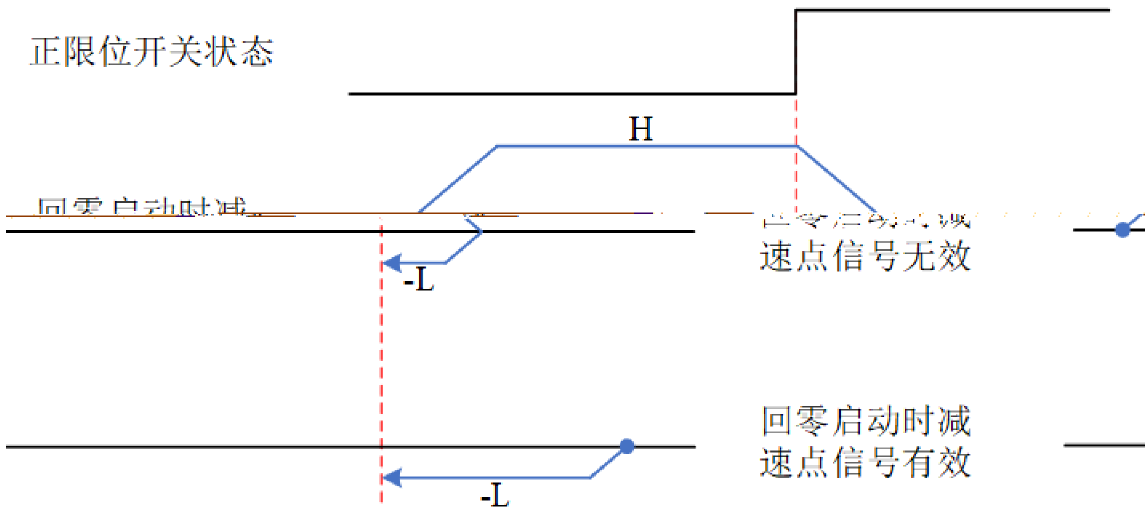


8 18

1

2

18

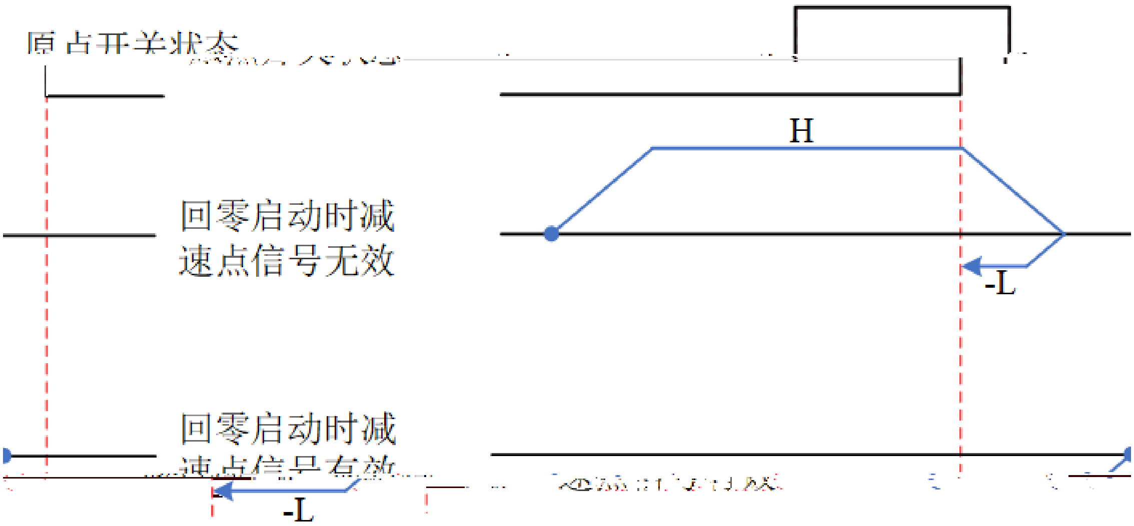


9 19

1

2

19

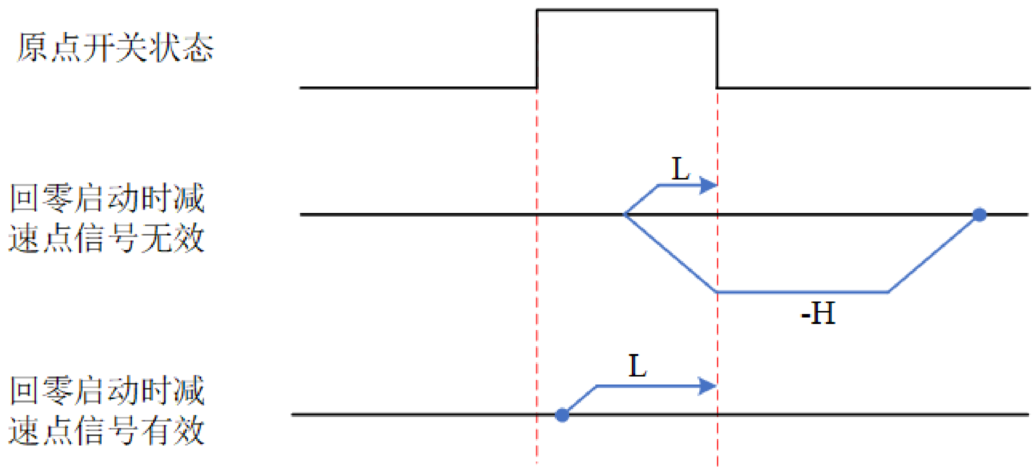


10 21

1

2

21



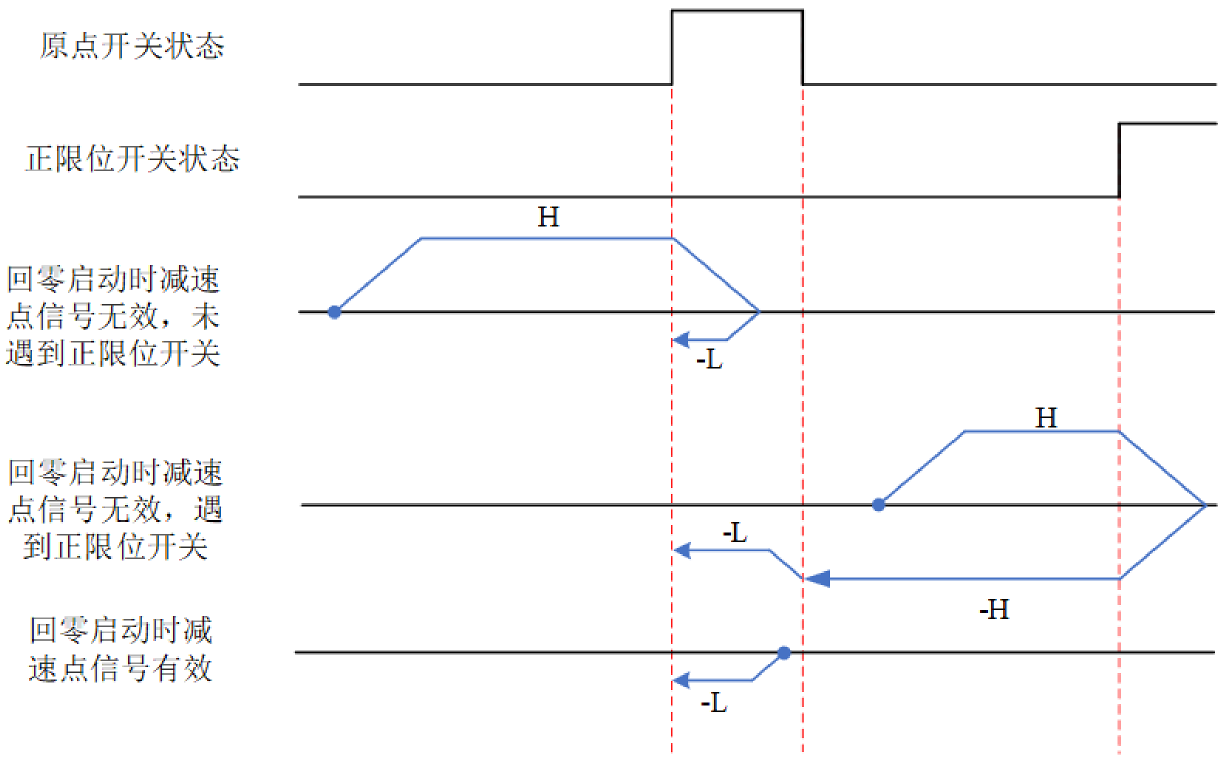
11 23

1

2

3

23


原点开关状态
正限位开关状态
H
回零启动时减速点信号无效，未遇到正限位开关
-L
H
回零启动时减速点信号无效，遇到正限位开关
-L
-H
回零启动时减速点信号有效
-L

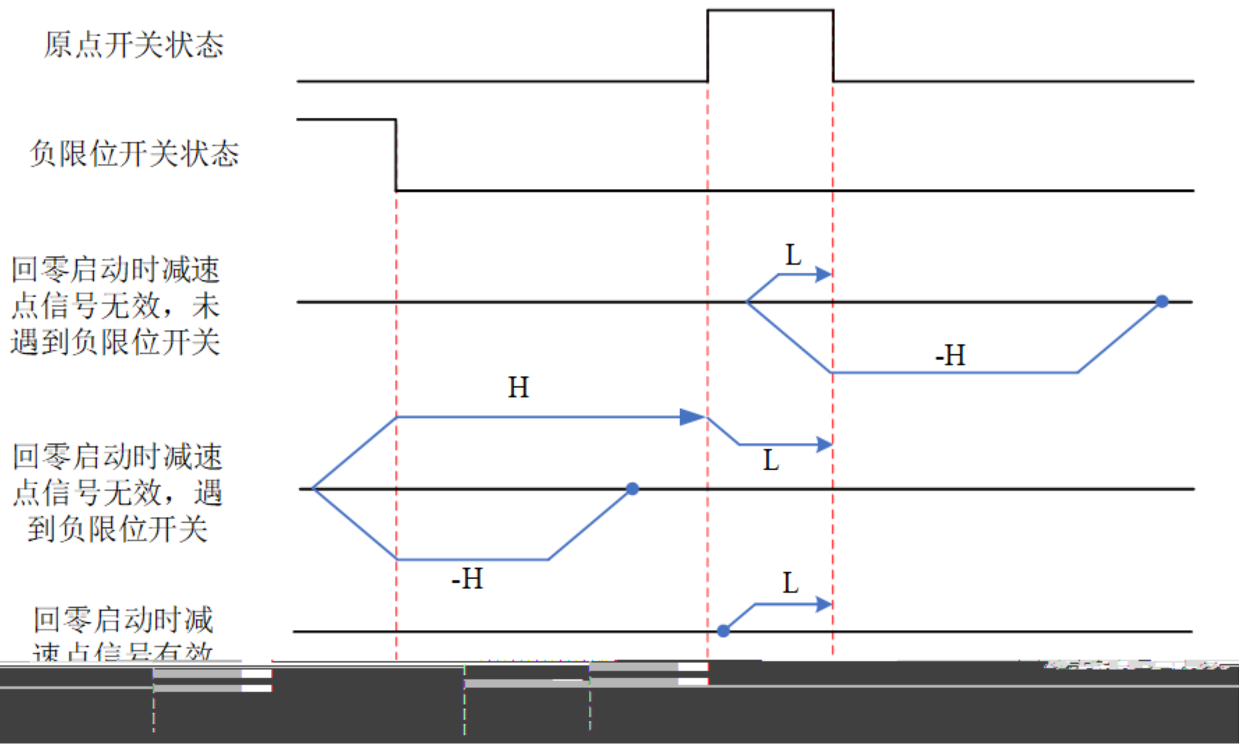
12 27

1

2

3

27

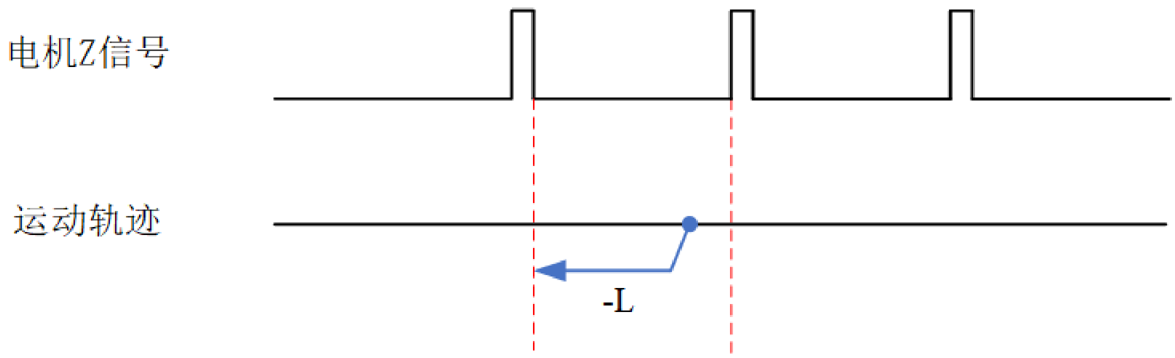


原点开关状态
负限位开关状态
回零启动时减速
点信号无效，未
遇到负限位开关
L
-H
H
回零启动时减速
点信号无效，遇
到负限位开关
L
-H
L
回零启动时减

13 33 Z

Z

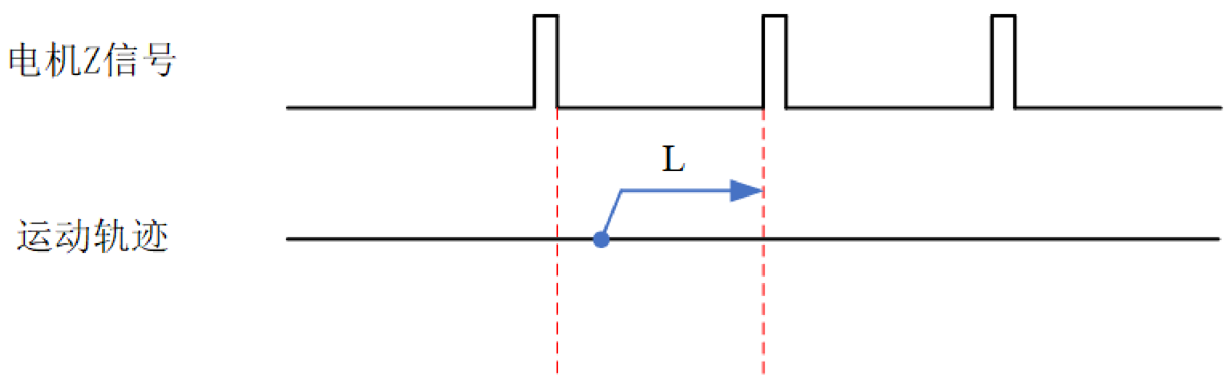
33



14 34 Z

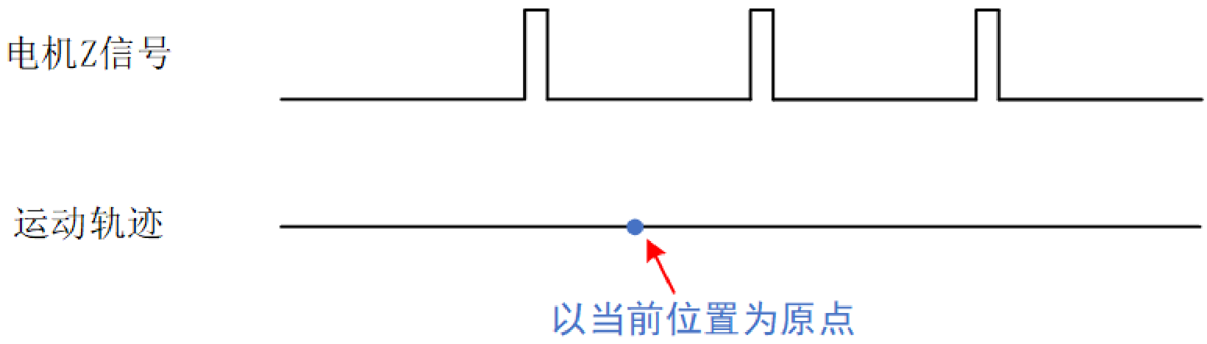
Z

34



15 35

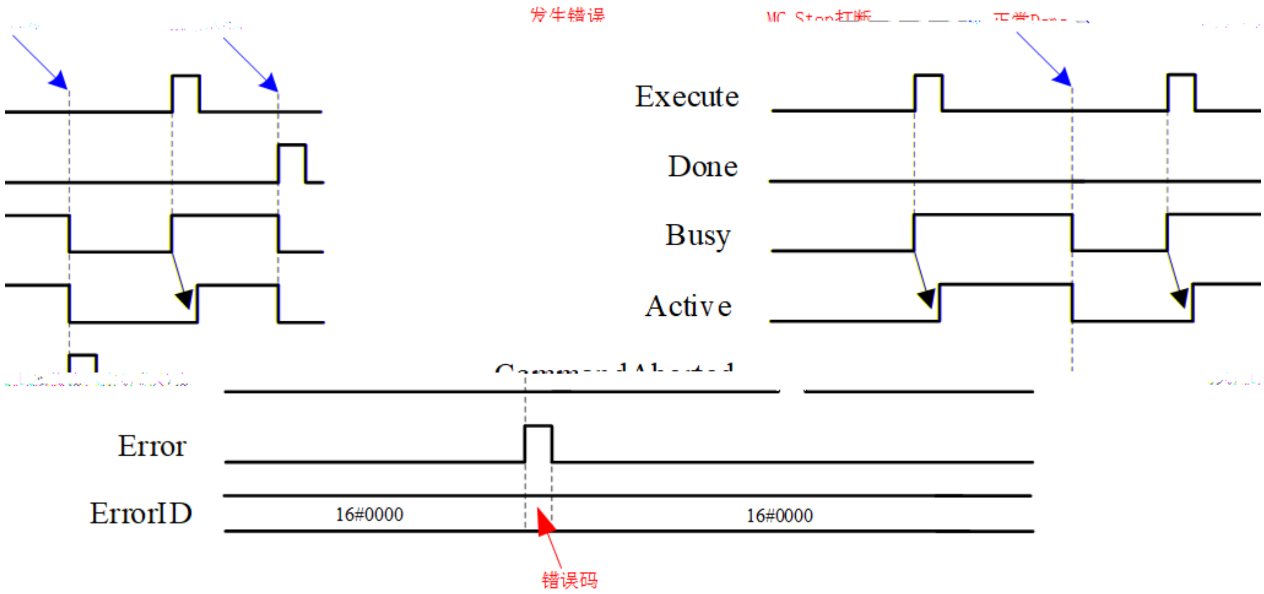
35



1

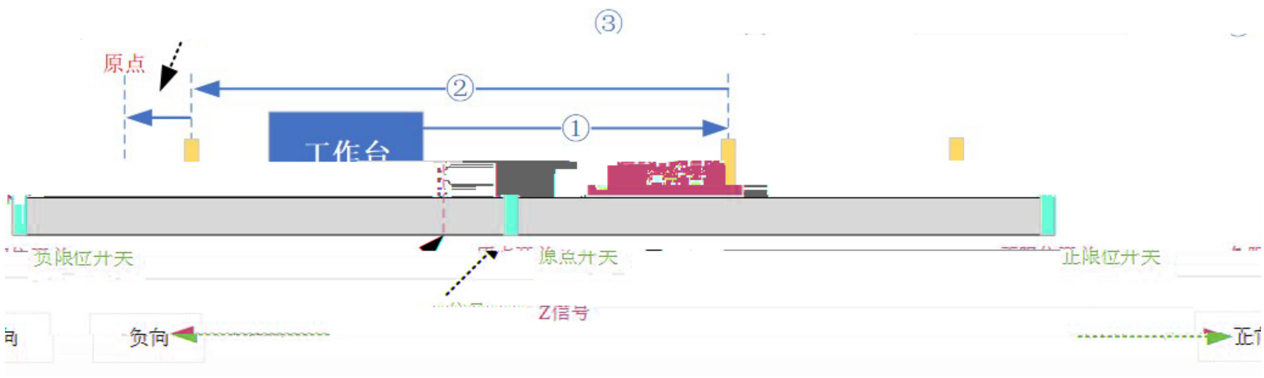
Homing

19



5.4.3

Z



MC_Home 7

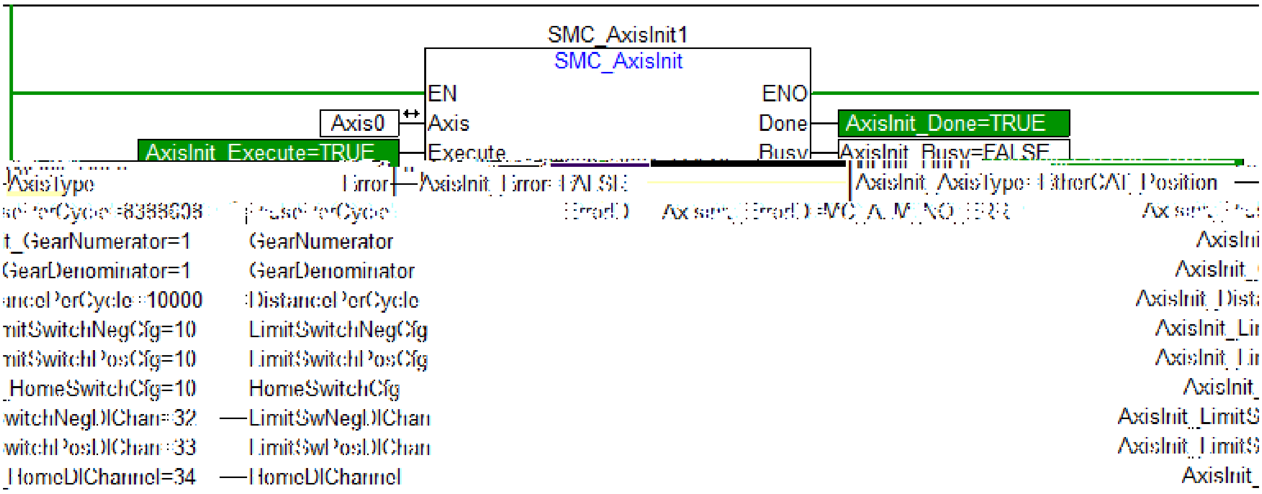
7 Velocity

Velocity ApproachVelocity Z

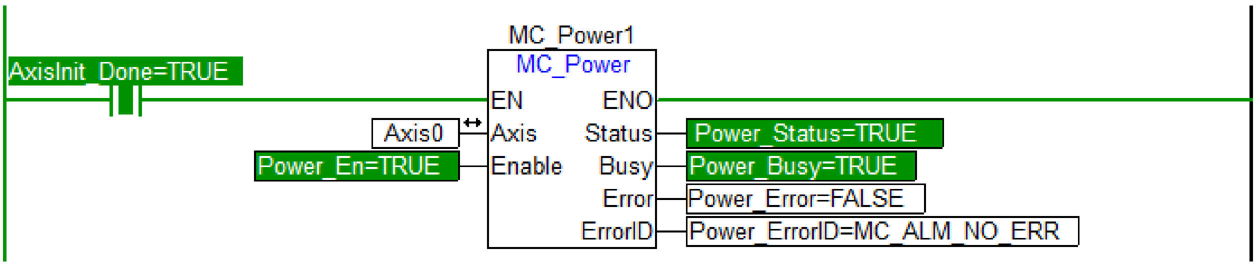
1	MC_Home	MC_HOME	-	
2	Home_En	BOOL	FALSE	
3	Home_HomeMode	SMC_HOME_MODE	MC_HM1_NEG_LIMIT_Z	
4	Home_Vel	LREAL	10000	
5	Home_Vel2	LREAL	5000	
6	Home_Acc	LREAL	10000	
7	Home_Dec	LREAL	10000	
8	Home_Jerk	LREAL	0	
9	Home_Pos	LREAL	0	
10	Home_Done	BOOL	FALSE	
11	Home_Busy	BOOL	FALSE	
12	Home_Active	BOOL	FALSE	
13	Home_Abort	BOOL	FALSE	
14	Home_Error	BOOL	FALSE	
15	Home_ErrorID	SMC_ERROR	MC_ALM_NO_ERR	
16	MC_MoveAbsolute	MC_MOVEABSOLUTE	-	
17	MoveAbs_En	BOOL	FALSE	
18	MoveAbs_Pos	LREAL	20000	
19	MoveAbs_Vel	LREAL	10000	
20	MoveAbs_Acc	LREAL	100000000	
21	MoveAbs_Dec	LREAL	100000000	
22	MoveAbs_Dir	MC_DIRECTION	mcPositiveDirection	
23	MoveAbs_Jerk	LREAL	0	
24	MoveAbs_Buff	MC_BUFFER_MODE	Aborting	
25	MoveAbs_Done	BOOL	FALSE	
26	MoveAbs_Busy	BOOL	FALSE	
27	MoveAbs_Active	BOOL	FALSE	
28	MoveAbs_Abort	BOOL	FALSE	
29	MoveAbs_Error	BOOL	FALSE	
30	MoveAbs_ErrorID	SMC_ERROR	MC_ALM_NO_ERR	

LD

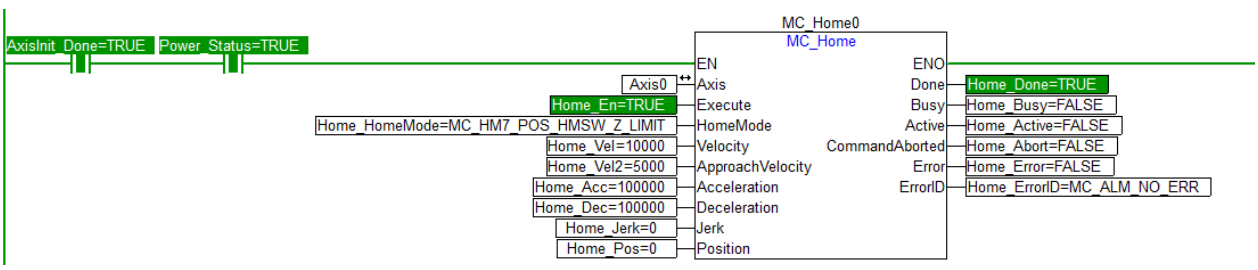
1 SMC_AxisInit EtherCAT_Poaition



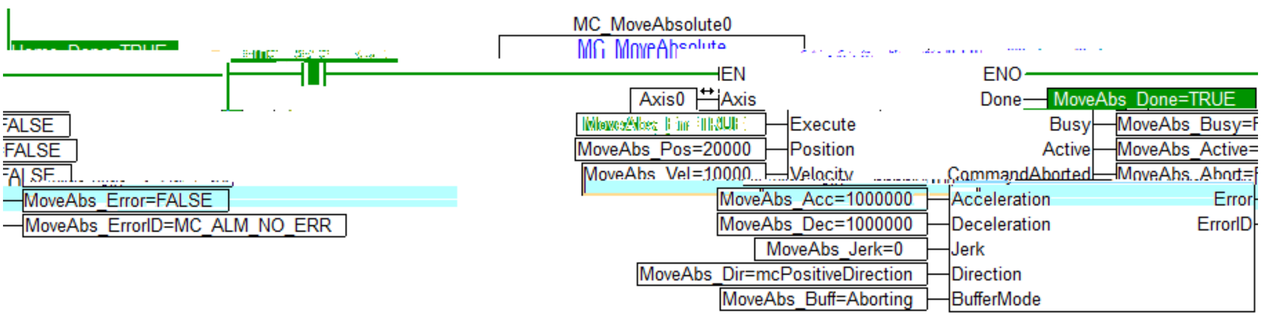
2 MC_Home MC_Power



3 MC_Home 7



4



ST

```
    Execute := Home_En,
    HomeMode := Home_HomeMode,
    Velocity := Home_Vel,
    ApproachVelocity := Home_Vel2,
    Acceleration := Home_Acc,
    Deceleration := Home_Dec,
    Jerk := Home_Jerk,
    Position := Home_Pos,
    Axis := Axis0,
    Done=>Home_Done,
    Busy=>Home_Busy,
    Active=>Home_Active,
    CommandAborted=>Home_Abort,
    Error=>Home_Error,
    ErrorID=>Home_ErrorID);
END_IF
```

4

```
IF TRUE=Home_Done THEN
    MC_MoveAbsolute0(
    Execute := MoveAbs_En,
    Position := MoveAbs_Pos,
    Velocity := MoveAbs_Vel,
    Acceleration := MoveAbs_Acc,
    Deceleration := MoveAbs_Dec,
    Jerk := MoveAbs_Jerk,
    Direction := MoveAbs_Dir,
    BufferMode := MoveAbs_Buff,
    Axis := Axis0,
    Done=>MoveAbs_Done,
    Busy=>MoveAbs_Busy,
    Active=>MoveAbs_Active,
    CommandAborted=>MoveAbs_Abort,
    Error=>MoveAbs_Error,
    ErrorID=>MoveAbs_ErrorID);
END_IF
```

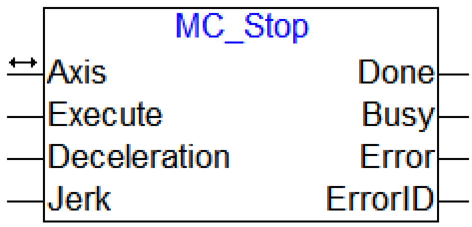
5.4.4

Z 1

MC_TouchProbe HMC_GantryInit 1

Z

5.5 MC_Stop



5.5.1

IN_OUT	Axis			-	-	AXIS_REF
INPUT	Execute			-	TRUE/FALSE	BOOL
	Deceleration			-		LREAL
	Jerk	0 S		0	0/	LREAL
OUTPUT	Done			FALSE	TRUE/FALSE	BOOL
	Busy			FALSE	TRUE/FALSE	BOOL
	Error			FALSE	TRUE/FALSE	BOOL
	ErrorID			0	-	SMC_ERROR

5.5.2

1

Deceleration( ) Jerk( )

2

Stopping

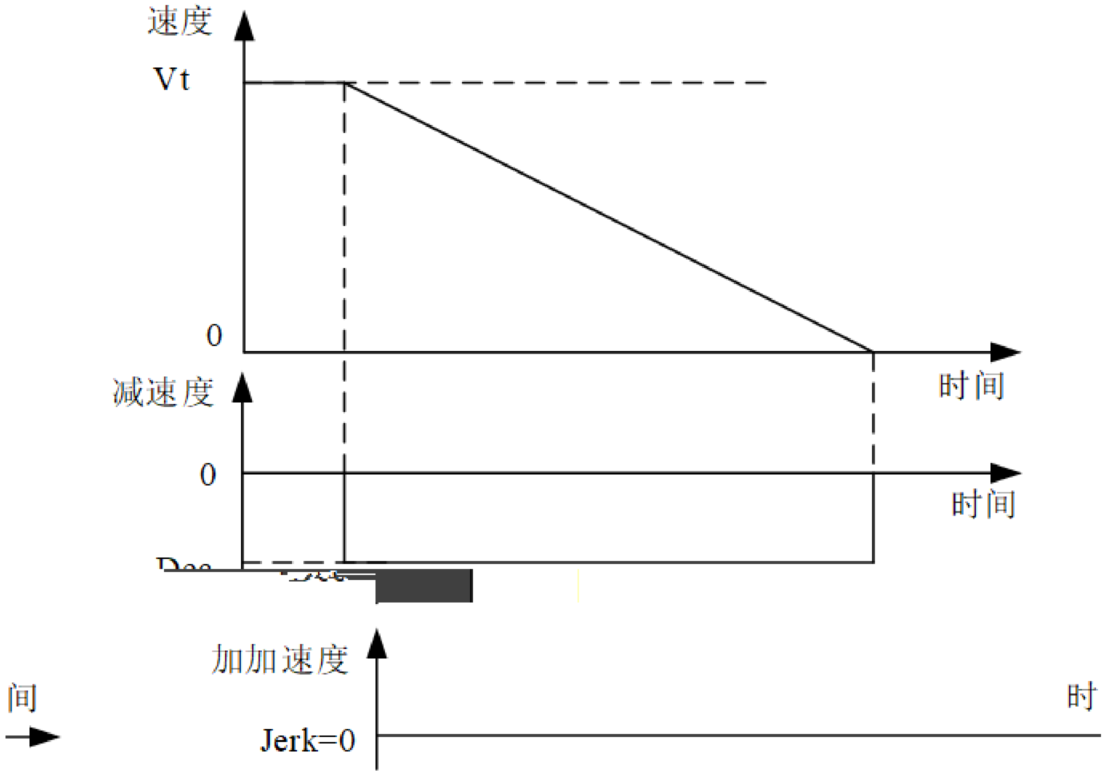
3

Axis ID Deceleration Jerk

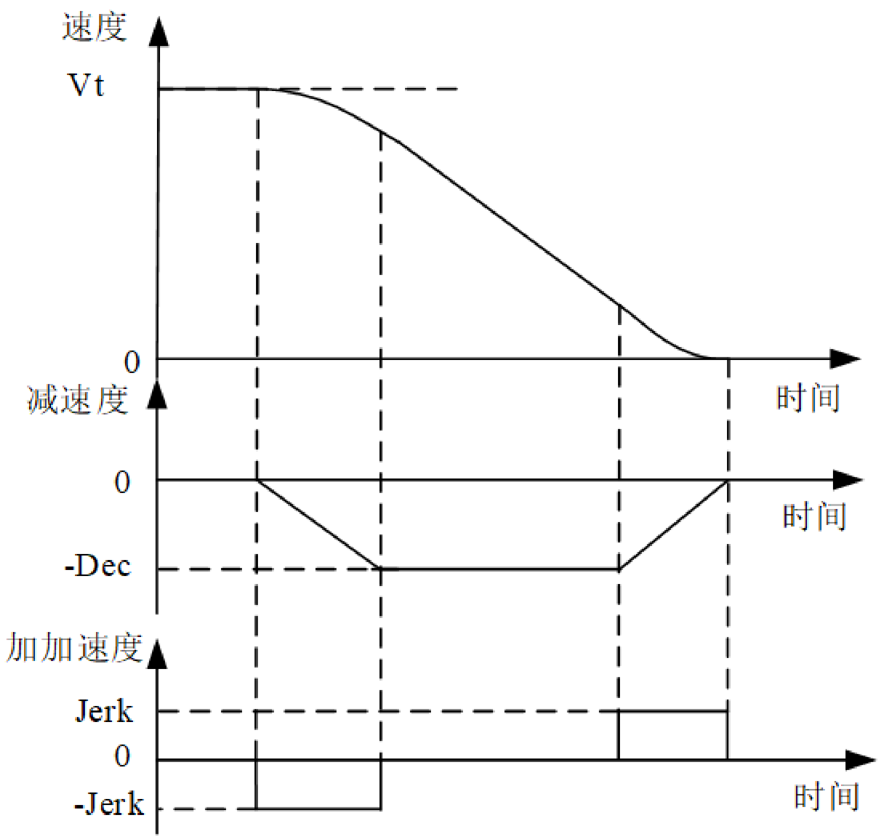
4

Jerk=0 Jerk>0 S

Jerk=0



Jerk>0 S



Virtual 0

EtherCAT_Position 50 EtherCAT

EtherCAT_Speed 51 EtherCAT

EtherCAT_Torque 52 EtherCAT

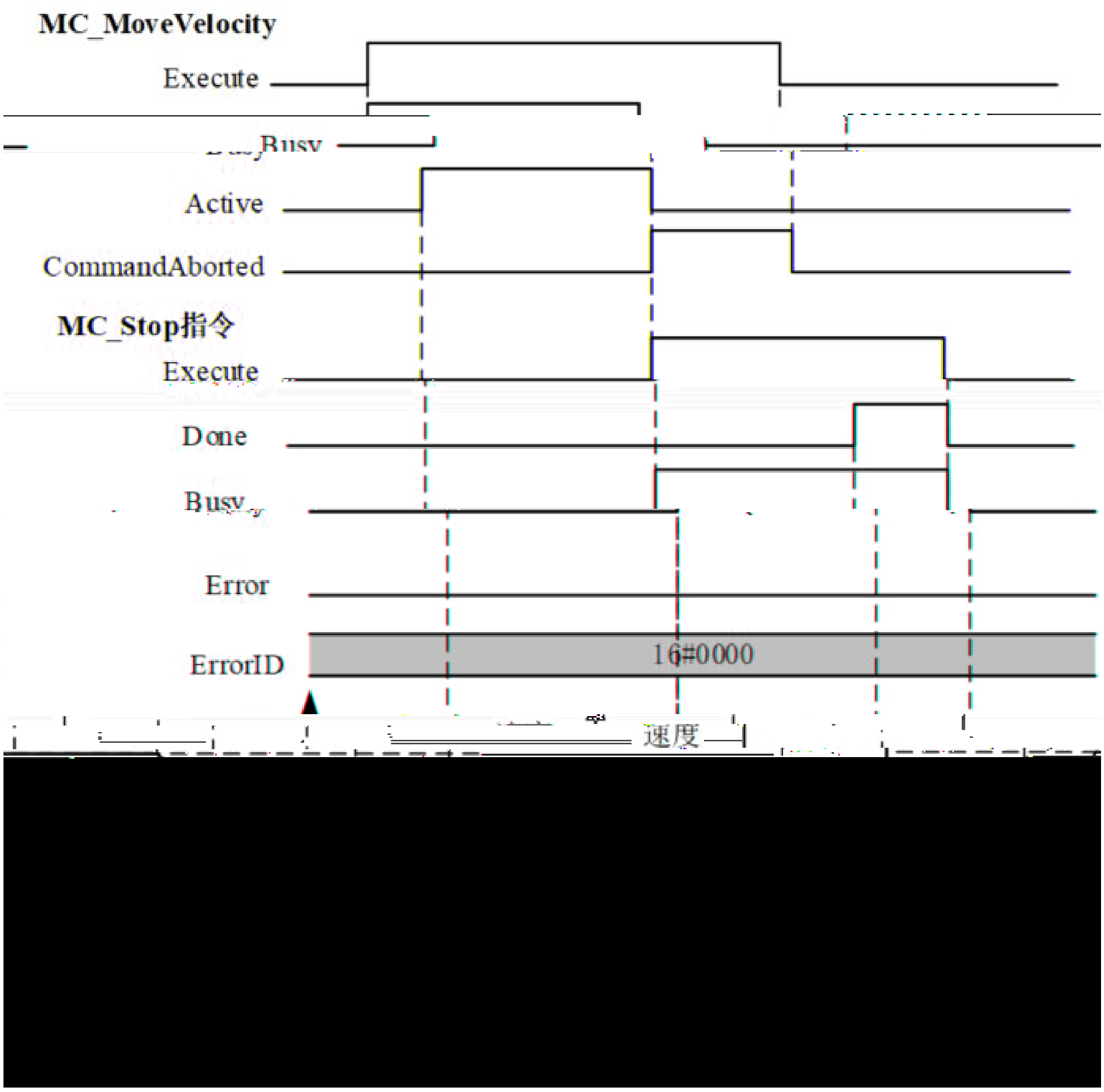
1

Execute Busy TRUE Stopping

Done TRUE Execute Busy Done

TRUE Stopping Execute Busy Done FALSE

StandStill

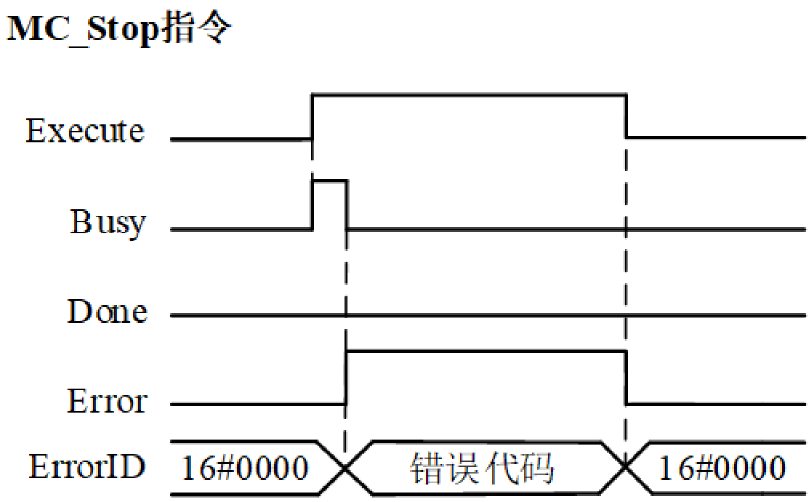


2

Disabled ErrorStop

Error=True ErrorID

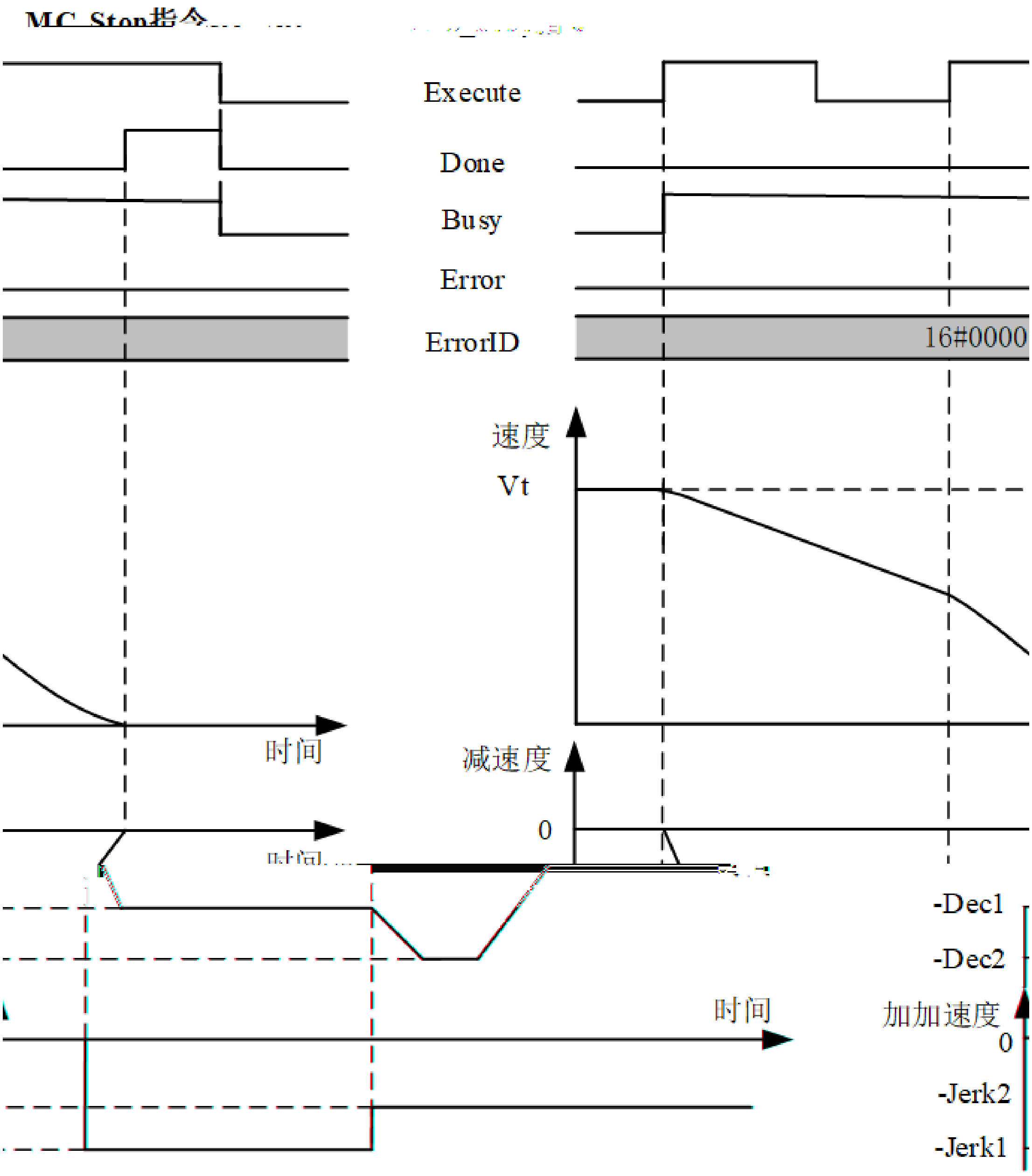
Execute



Execute MC_Stop

3

MC_Stop Deceleration Jerk



5.5.3

Axis0 AXIS_REF MC_Stop MC_MoveVelocity

MC_ReadStatus

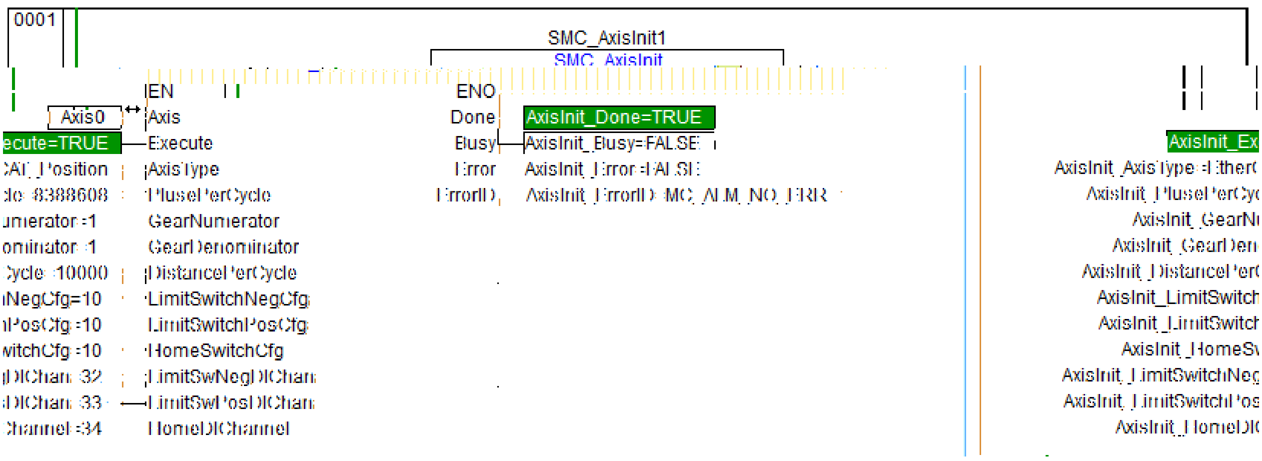
Deceleration 10000 Jerk 0

Axis0	AXIS_REF	-	0
Power1_Status	BOOL	FALSE	TRUE
MV1_Active	BOOL	FALSE	TRUE
MV1_Comd	BOOL	FALSE	TRUE
Stop1_Exe	BOOL	FALSE	
Stop1_Done	BOOL	FALSE	Done=TRUE
Switch1	BOOL	FALSE	Switch TRUE

1

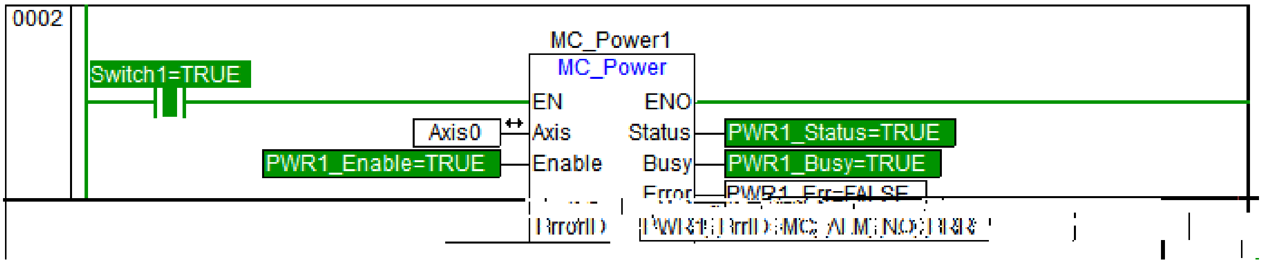
SMC_AxisInit Axis0 50 EtherCAT_Position

SMC_AxisInit SMC_AxisInit



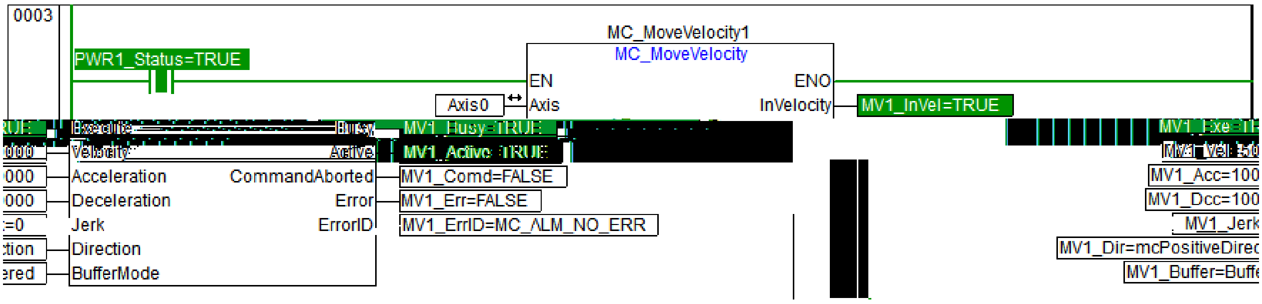
2

0002 Switch1 TRUE MC_Power



3

0003 MC_MoveVelocity

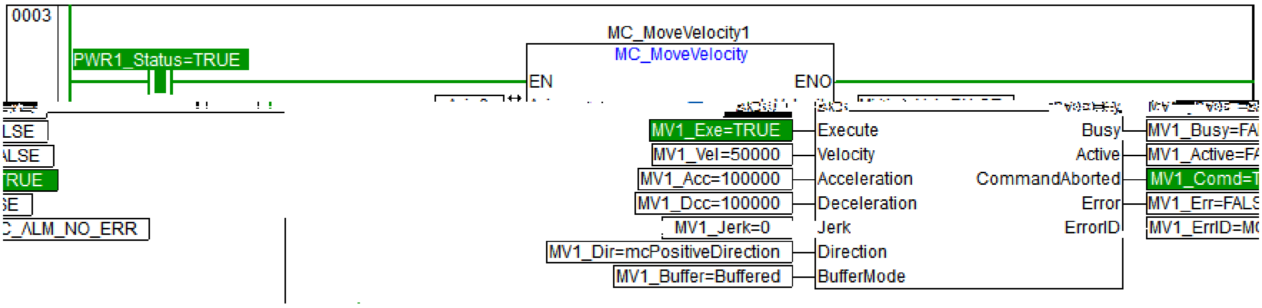


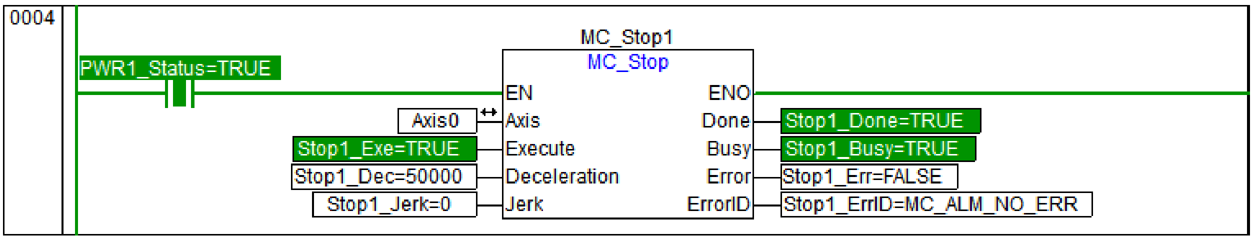
4

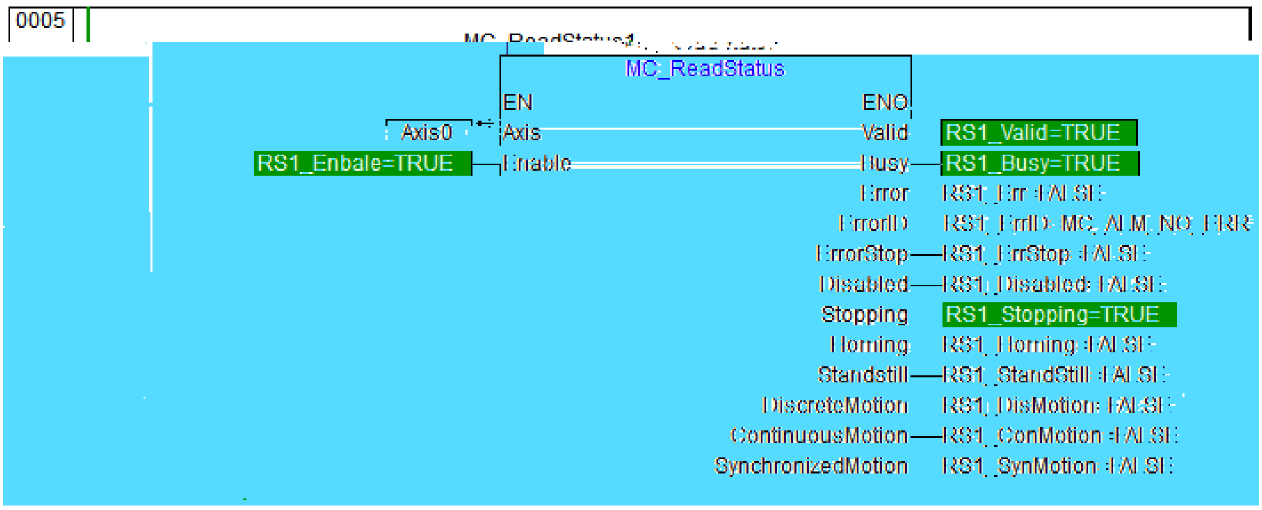
0004 MC_Stop MC_MoveLocity MC_MoveLocity

CommandAborted TRUE

MC_Stop Execute 0005 Stopping







ST

ST Deceleration 10000 Jerk 0

Axis0	AXIS_REF	-	0
Power1_Status	BOOL	FALSE	TRUE
MV1_Active	BOOL	FALSE	TRUE
Stop1_Exe	BOOL	FALSE	
Stop1_Done	BOOL	FALSE	Done=TRUE
Switch1	BOOL	FALSE	Switch TRUE
InitFlag	BOOL	FALSE	TRUE

1

Axis0 50 EtherCAT_Position SMC_AxisInit

```
SMC_AxisInit1(
Execute := AxisInit_Execute,
AxisType := AxisInit_AxisType,
PlusePerCycle := AxisInit_PlusePerCycle,
GearNumerator := AxisInit_GearNumerator,
GearDenominator := AxisInit_GearDenominator,
DistancePerCycle := AxisInit_DistancePerCycle,
LimitSwitchNegCfg := AxisInit_LimitSwitchNegCfg,
LimitSwitchPosCfg := AxisInit_LimitSwitchPosCfg,
HomeSwitchCfg := AxisInit_HomeSwitchCfg,
LimitSwNegDIChan := AxisInit_LimitSwitchNegDIChan,
LimitSwPosDIChan := AxisInit_LimitSwitchPosDIChan,
HomeDIChannel := AxisInit_HomeDIChannel,
Axis := Axis0,
Done=>AxisInit_Done,
Busy=>AxisInit_Busy,
Error=>AxisInit_Error,
ErrorID=>AxisInit_ErrorID);
```

2

```
IF FALSE=InitFlag THEN
Stop1_Dec:=10000;(*      MC_Stop                      *)
Stop1_Jerk:= 0; (*      MC_Stop         Jerk      *)
InitFlag:=TRUE;
END_IF
```

3

Switch1 TRUE MC_Power

```
IF Switch1 THEN
 MC_Power1(
 Enable := PWR1_Enable,
 Axis := Axis0,
 Status=> PWR1_Status,
```

```
 Busy=> PWR1_Busy,
 Error=> PWR1_Err,
 ErrorID=> PWR1_ErrID);
END_IF
```

4

MC_MoveVelocity Axis0

```
IF  PWR1_Status THEN
   MC_MoveVelocity1(
   Execute := MV1_ Exe,
   Velocity := MV1_ Vel,
   Acceleration := MV1_ Acc,
   Deceleration := MV1_ Dcc,
   Jerk := MV1_ Jerk,
   Direction := MV1_ Dir,
   BufferMode := MV1_ Buffer,
   Axis := Axis0,
   InVelocity=> MV1_ InVel,
   Busy=> MV1_ Busy,
   Active=> MV1_ Active,
   CommandAborted=> MV1_ Comd,
   Error=> MV1_ Err,
   ErrorID=> MV1_ Error);
END_IF
```

5 MC_Stop

MC_Stop MC_MoveVelocity

```
IF  PWR1_Status THEN
   MC_Stop1(
   Execute := Stop1_Exe,
   Deceleration := Stop1_Dec,
   Jerk := Stop1_Jerk,
   Axis := Axis0,
   Done=> Stop1_Done,
   Busy=> Stop1_Busy,
   Error=> Stop1_Err,
   ErrorID=> Stop1_ErrID);
```

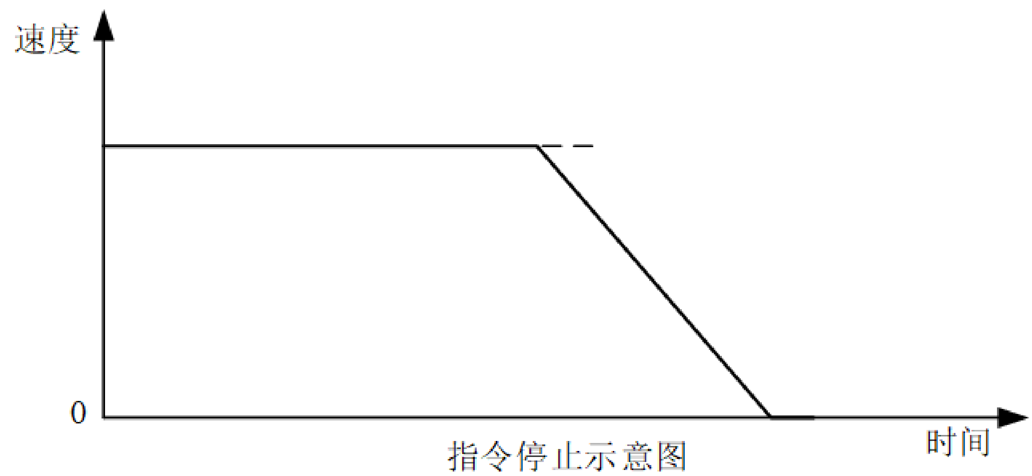
```
END_IF
```

6 MC_ReadStatus

MC_ReadStatus

```
MC_ReadStatus1(
Enable := RS1_Enbale,
Axis := Axis0,
Valid=> RS1_Valid,
Busy=> RS1_Busy,
Error=> RS1_Err,
ErrorID=> RS1_ErrID,
ErrorStop=> RS1_ErrStop,
Disabled=> RS1_Disabled,
Stopping=> RS1_Stopping,
Homing=> RS1_Homing,
Standstill=> RS1_StandStill,
DiscreteMotion=> RS1_DisMotion,
ContinuousMotion=> RS1_ConMotion,
SynchronizedMotion=> RS1_SynMotion);
```

MC_MoveVelocity



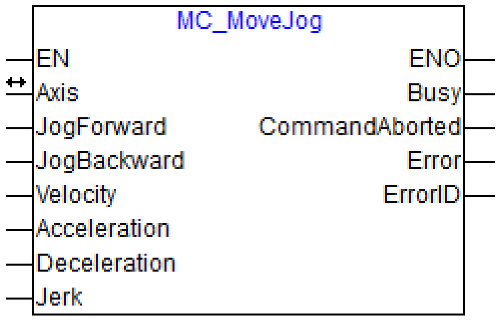
5.5.4

MC_Stop Execute Stopping

Execute StandStill

5.6 MC_MoveJog JOG

/



5.6.1

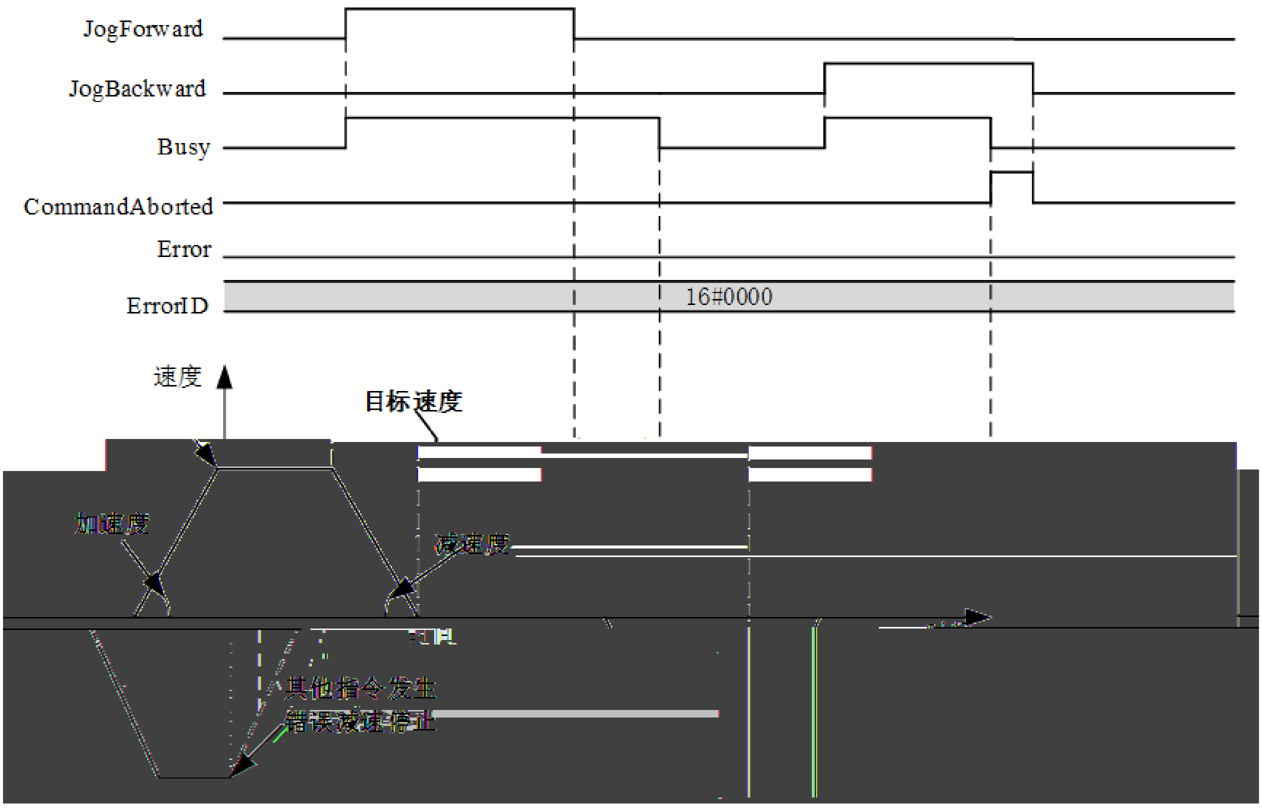
IN_OUT	Axis					

1

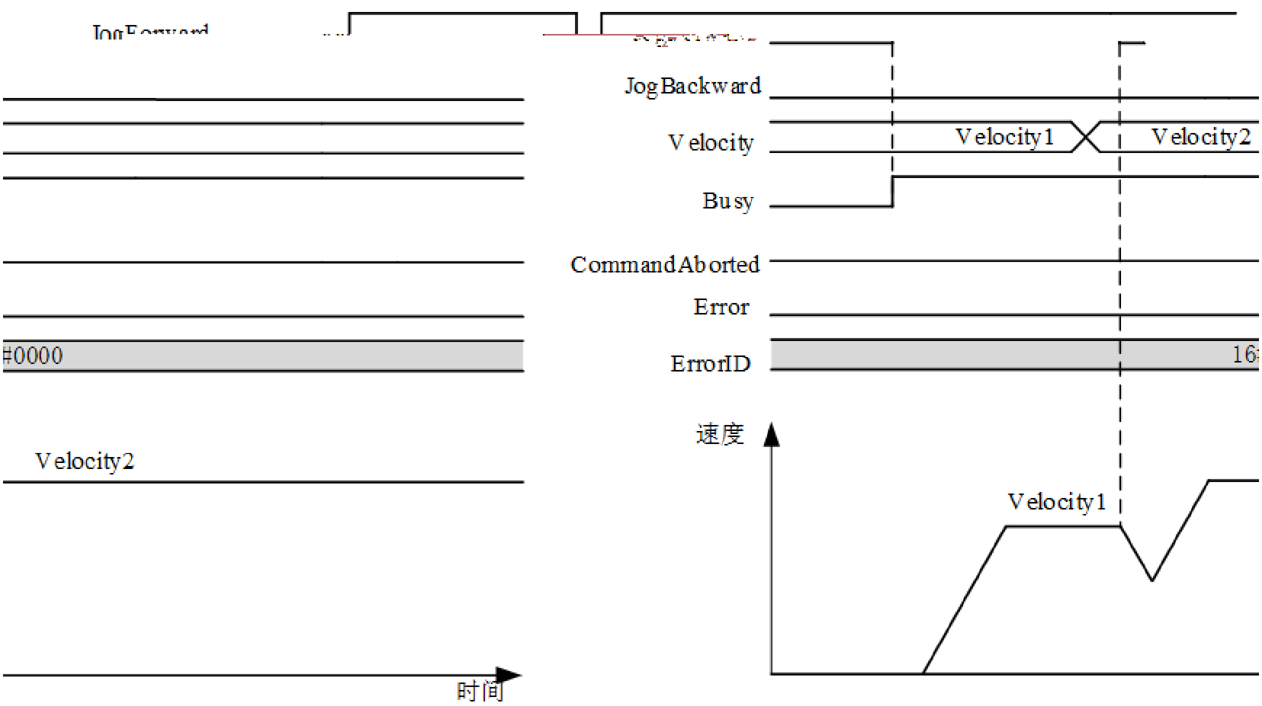
JogForward( ) TRUE

JogBackward( ) TRUE

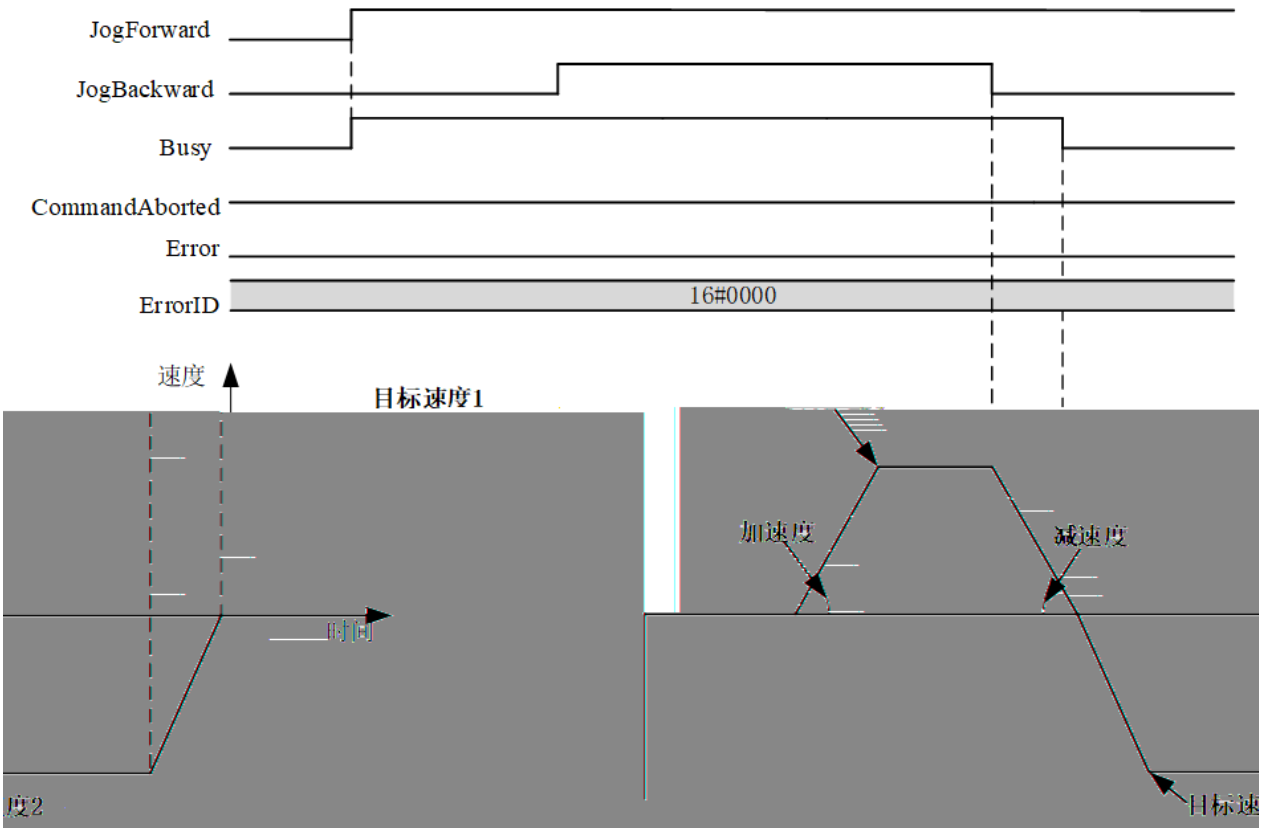
2


JogForward
JogBackward
Busy
CommandAborted
Error
ErrorID
16#0000
速度
目标速度
加速度
减速度
其他指令发生
错误减速停止

2


JogBackward
Velocity
Velocity1
Velocity2
Busy
CommandAborted
Error
ErrorID
#0000
速度
时间

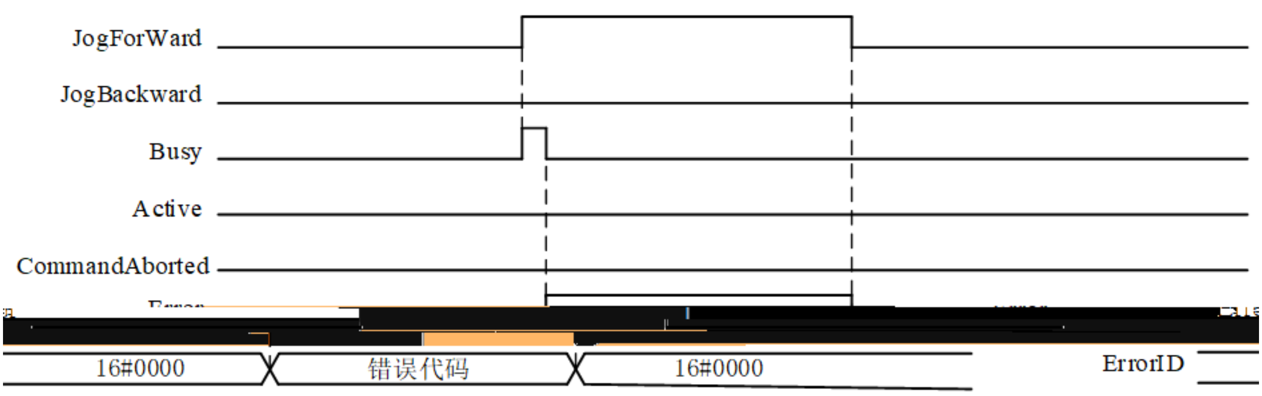
3



4

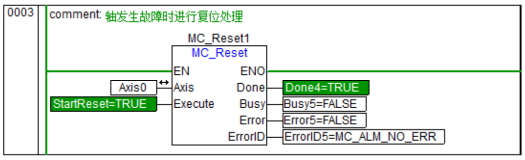
Error( ) TRUE

ErrorID( )



5.6.3

MC_MoveJog



ST

1

```
(*                    *)
SMC_AxisInit1(
Execute := StartInit,
AxisType := 50,
PlusePerCycle := 10000,
GearNumerator := 1,
GearDenominator := 1,
DistancePerCycle := 10000,
LimitSwitchNegCfg := -1,
LimitSwitchPosCfg := -1,
HomeSwitchCfg := -1,
LimitSwNegDIChan := 0,
LimitSwPosDIChan := 1,
HomeDIChannel := 2,
Axis := Axis0,
Done=> DoneInit,
Busy=>Busy1,
Error=>Error1,
ErrorID=>ErrorID1);
(*                    *)
IF DoneInit THEN
   MC_Power1(
   Enable := PowerEnable,
   Axis := Axis0,
   Status=> PowerStatus,
   Busy=>Busy2,
   Error=>Error2,
   ErrorID=>ErrorID2);
END_IF
```

2

```
(*                    *)

MC_MoveJog1(
JogForward := FwMoveJog,
JogBackward := BwMoveJog,
Velocity := 10000,
Acceleration := 20000,
Deceleration := 20000,
Jerk := 0,
Axis := Axis0,
Busy=>BusyMoveJog,
CommandAborted=>ComMoveJog,
Error=>ErrorMoveJog,
ErrorID=>ErrorID_MoveJog1);
```

3

```
(*                         *)

MC_MoveJog2(
JogForward := FwMoveJog1,
JogBackward := BwMoveJog1,
Velocity := 20000,
Acceleration := 40000,
Deceleration := 40000,
Jerk := 0,
Axis := Axis0,
Busy=>BusyMoveJog1,
CommandAborted=>ComMoveJog1,
Error=>ErrorMoveJog1,
ErrorID=>ErrorID_MoveJog2);
```

4

```
MC_Reset1(
Execute := Exe5,
Axis := Axis0,
Done=>Done4,
Busy=>Busy5,
Error=>Error5,
```

ErrorID=>ErrorID5);

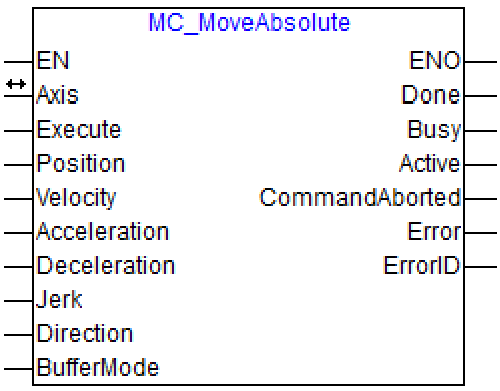
5.6.4

0

TRUE

5.6.5

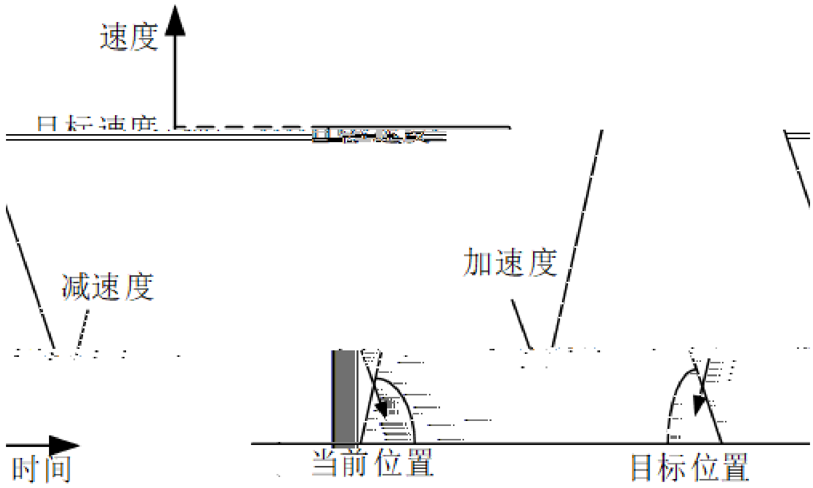
5.7 MC_MoveAbsolute



5.7.1

IN_OUT	Axis			-	-	AXIS_REF

	Deceleration			-		LREAL
	Jerk	0 S		0	0/	LREAL
	Direction	0: 1: 2: 3: 4:		4	0: mcPositiveDirection 1: mcShortestWay 2: mcNegativeêBÀBBfayaE(0E4:Di	



Direction

Direction mcPositiveDirection

Direction mcShortestWay

Direction mcNegativeDirection

Direction mcCurrentDirection

Direction mcNoDirection

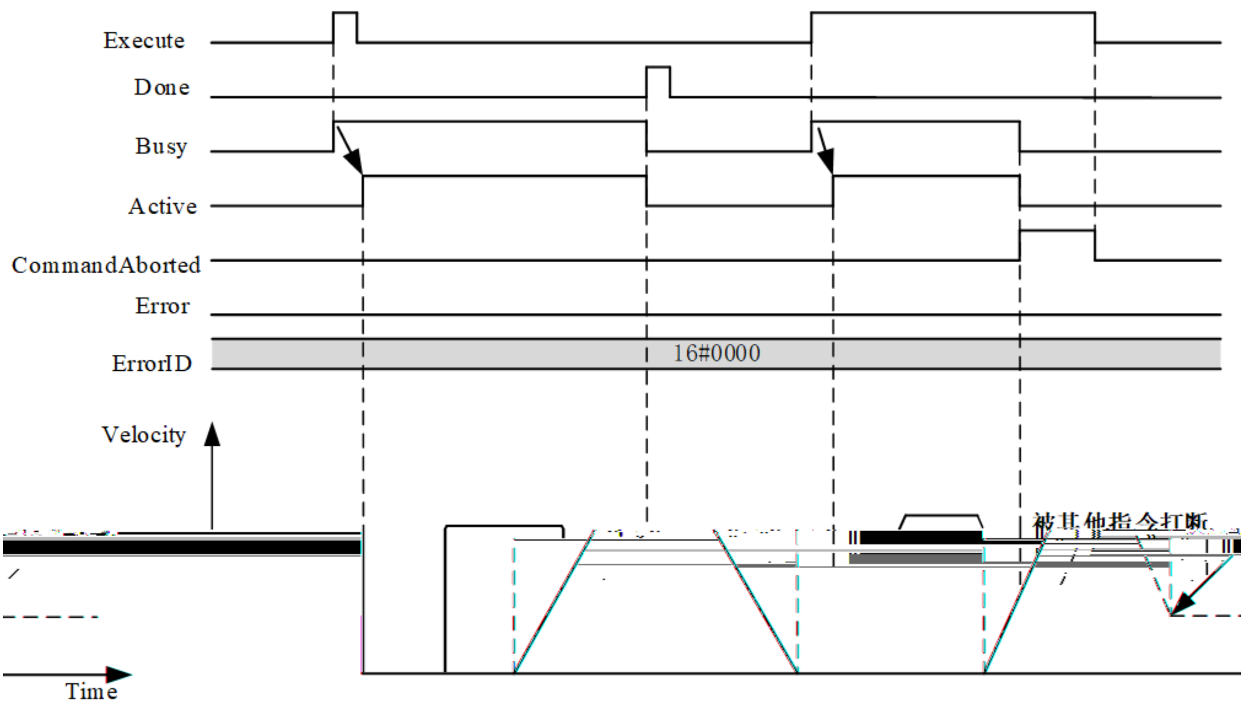
BufferMode

BufferMode

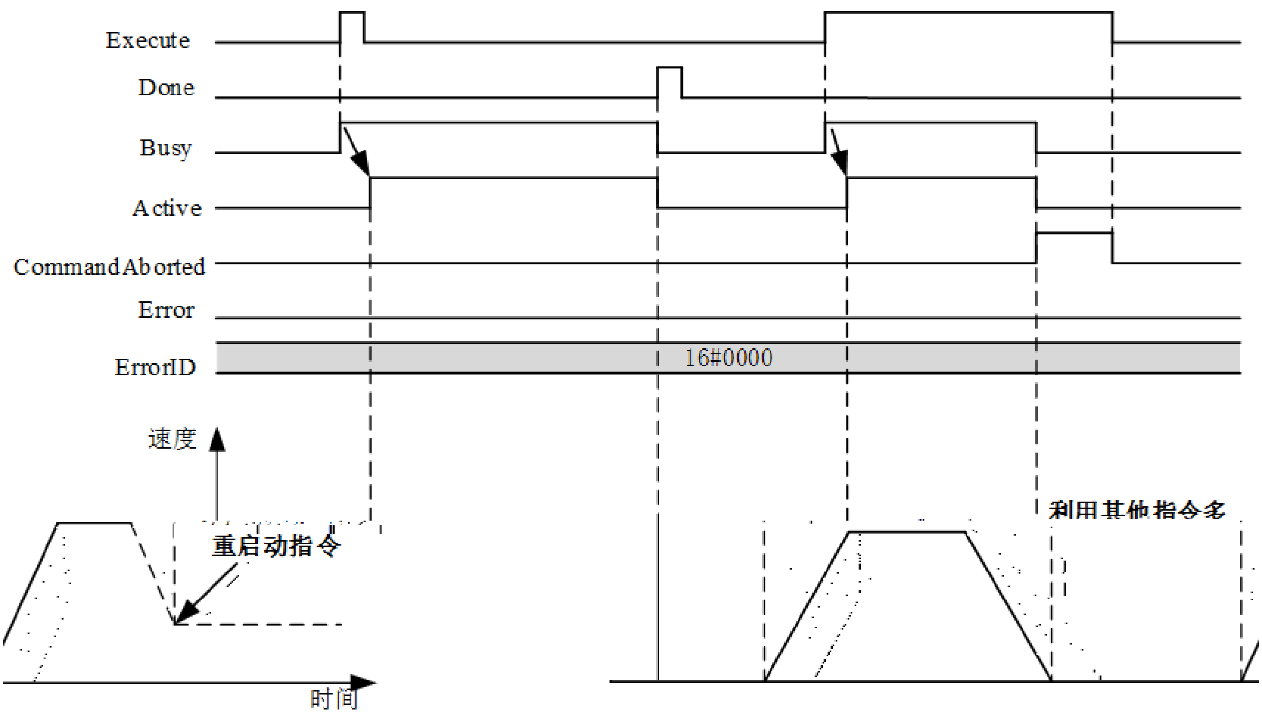
BufferMode Aborting( ) Execute TRUE

Position Velocity

Acceleration Deceleration Jerk



1



2

Error FALSE

ErrorID

5.7.3

MC_MoveAbsolute

Axis0	AXIS_REF		0
StartInit	BOOL	FALSE	TRUE
DoneInit	BOOL	FALSE	TRUE
PowerEnable	BOOL	FALSE	TRUE
PowerStatus	BOOL	FALSE	TRUE
StartMoveAbs	BOOL	FALSE	TRUE
BusyMoveAbs	BOOL	FALSE	TRUE
ActiveMoveAbs	BOOL	FALSE	TRUE
DoneMoveAbs	BOOL	FALSE	TRUE

1

```
(*                    *)
SMC_AxisInit1(
Execute := StartInit,
AxisType := 50,
PlusePerCycle := 10000,
GearNumerator := 1,
GearDenominator := 1,
DistancePerCycle := 10000,
LimitSwitchNegCfg := -1,
LimitSwitchPosCfg := -1,
HomeSwitchCfg := -1,
LimitSwNegDIChan := 0,
LimitSwPosDIChan := 1,
HomeDIChannel := 2,
Axis := Axis0,
Done=>Done1,
Busy=>Busy1,
Error=>Error1,
ErrorID=>ErrorID1);
(*                    *)
IF DoneInit THEN
  MC_Power1(
  Enable := PowerEnable,
  Axis := Axis0,
  Status=> PowerStatus,
  Busy=>Busy2,
  Error=>Error2,
  ErrorID=>ErrorID2);
END_IF
```

2 MC_MoveAbsolute

```
IF PowerStatus THEN
(*                          *)
MC_MoveAbsolute1(
Execute := StartMoveAbs,
```

```
Position := 10000,
Velocity := 1000,
Acceleration := 2000,
Deceleration := 2000,
Jerk := 0,
Direction := 4,
BufferMode := 0,
Axis := Axis0,
Done=> DoneMoveAbs,
Busy=> BusyMoveAbs,
Active=> ActiveMoveAbs,
CommandAborted=> ComMoveAbs,
Error=> ErrorMoveAbs,
ErrorID=>ErrorID3);
(*                              *)
MC_MoveAbsolute2(
Execute := StartMoveAbs1,
Position := 50000,
Velocity := 2000,
Acceleration := 5000,
Deceleration := 5000,
Jerk := 0,
Direction := 4,
BufferMode :=0,
Axis := Axis0,
Done=> DoneMoveAbs1,
Busy=> BusyMoveAbs1,
Active=> ActiveMoveAbs1,
CommandAborted=> ComMoveAbs1,
Error=> ErrorMoveAbs1,
ErrorID=>ErrorID4);
END_IF
```

3 MC_Reset

```
IF ErrorMoveAbs THEN
 (*                              *)
 MC_Reset1(
```

```
 Execute := StartReset,
 Axis := Axis0,
 Done=>Done4,
 Busy=>Busy5,
 Error=>Error5,
 ErrorID=>ErrorID5);
END_IF
```

5.7.4

Direction mcCurrentDirection

5.7.5

BufferMode ( )

5.8 MC_MoveRelative

5.8.1

IN_OUT	Axis			-	-	AXIS_REF

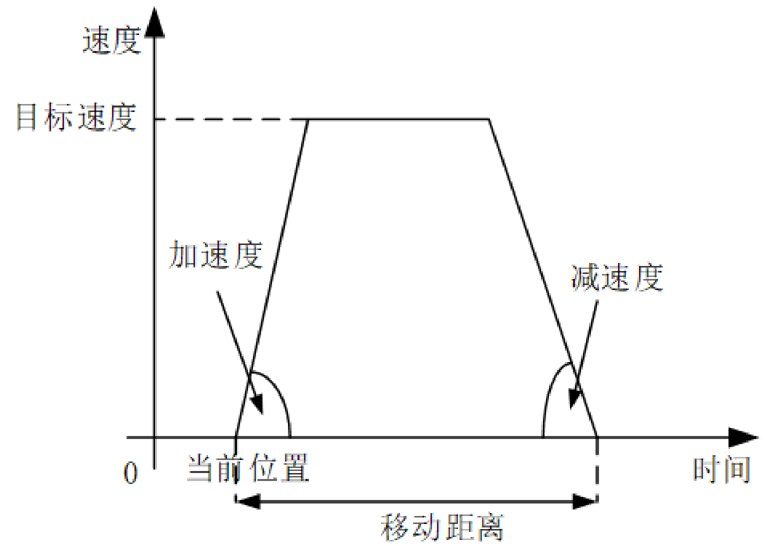
Execute

INPUT

5.8.2

Execute Distance

Velocity Acceleration Deceleration Jerk



Distance

BufferMode

BufferMode ( ) BufferMode 0 Aborting 1 Buffered 2 BlendingLow 3: BlendingPrevious 4: BlendingNext 5: BlendingHigh

BufferMode 0 Aborting BufferMode 1 Buffered BufferMode 2 BlendingLow BufferMode 3: BlendingPrevious BufferMode 4: BlendingNext BufferMode 5: BlendingHigh

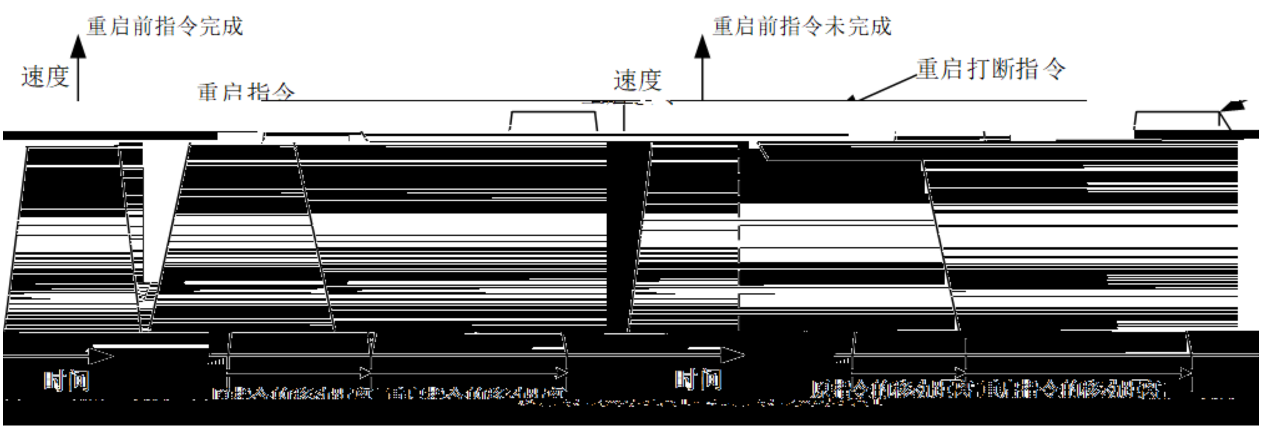
BufferMode

BufferMode

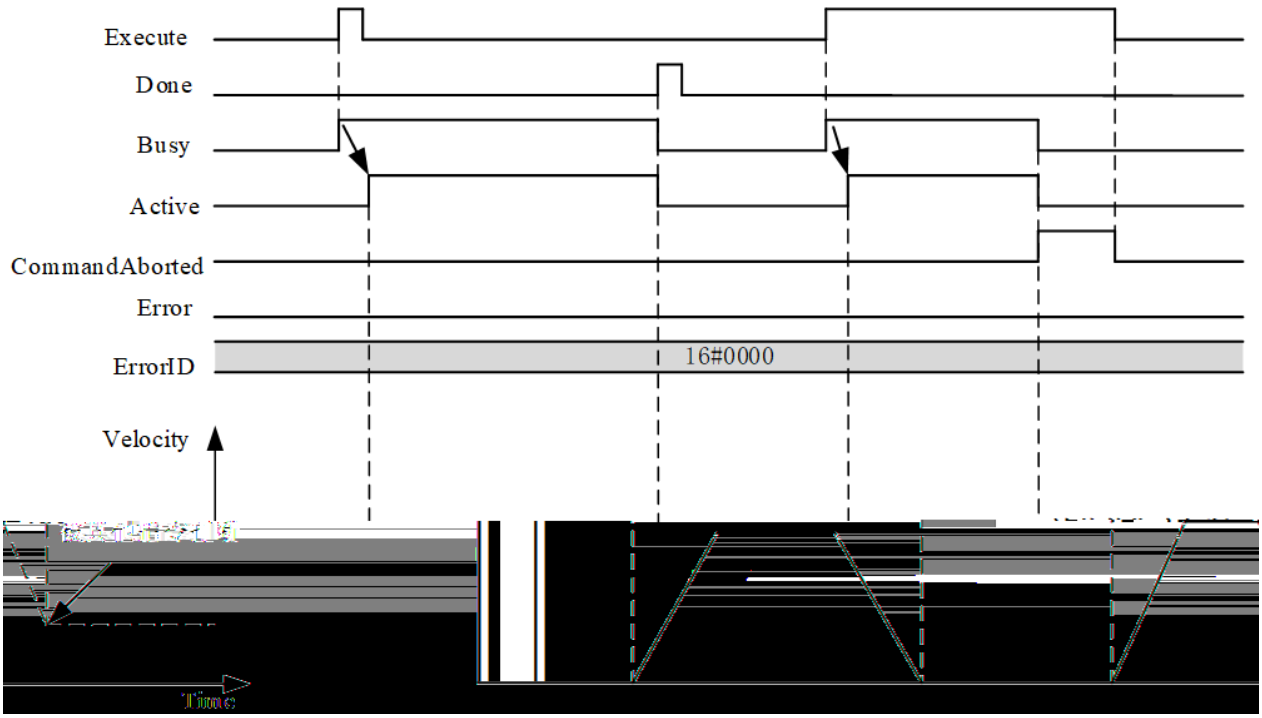
BufferMode Aborting( ) Execute TRUE

Distance Velocity

Acceleration Deceleration Jerk



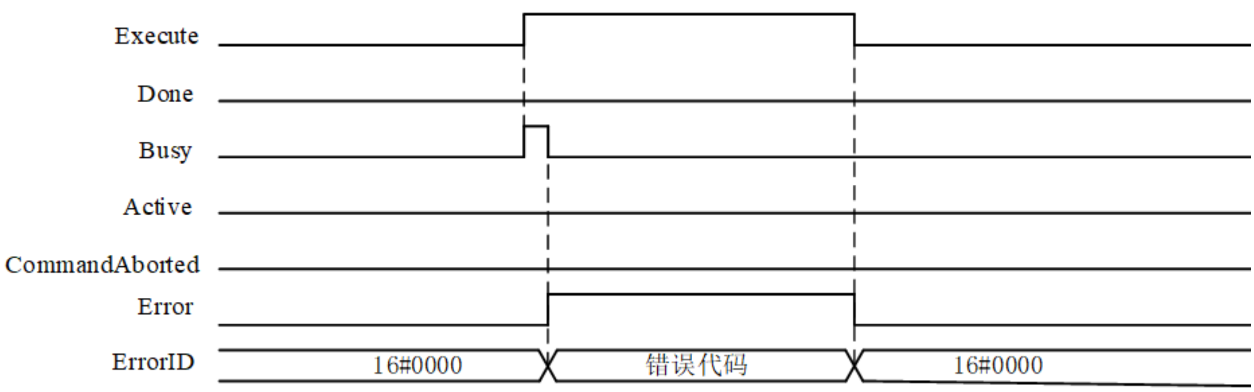
1



2

Error FALSE

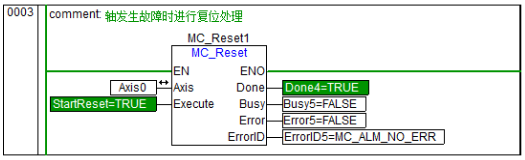
ErrorID



5.8.3

MC_MoveRelative

Axis0	AXIS_REF		0
StartInit	BOOL	FALSE	TRUE
DoneInit	BOOL	FALSE	TRUE
PowerEnable	BOOL	FALSE	TRUE
PowerStatus	BOOL	FALSE	TRUE
StartMoveRel	BOOL	FALSE	TRUE
BusyMoveRel	BOOL	FALSE	TRUE
ActiveMoveRel	BOOL	FALSE	TRUE
DoneMoveRel	BOOL	FALSE	TRUE
ComMoveRel	BOOL	FALSE	TRUE
ErrorMoveRel	BOOL	FALSE	TRUE
StartReset	BOOL	FALSE	TRUE



ST

1

```
(*                    *)
SMC_AxisInit1(
Execute := StartInit,
AxisType := 50,
PlusePerCycle := 10000,
GearNumerator := 1,
GearDenominator := 1,
DistancePerCycle := 10000,
LimitSwitchNegCfg := -1,
LimitSwitchPosCfg := -1,
HomeSwitchCfg := -1,
LimitSwNegDIChan := 0,
LimitSwPosDIChan := 1,
HomeDIChannel := 2,
Axis := Axis0,
Done=> DoneInit,
Busy=>Busy1,
Error=>Error1,
ErrorID=>ErrorID1);
(*                    *)
IF DoneInit THEN
   MC_Power1(
   Enable := PowerEnable,
   Axis := Axis0,
   Status=> PowerStatus,
   Busy=>Busy2,
   Error=>Error2,
   ErrorID=>ErrorID2);
END_IF
```

2

```
(*                         *)
IF PowerStatus THEN
 MC_MoveRelative1(
 Execute := StartMoveRel,
 Distance := 10000,
 Velocity := 1000,
 Acceleration := 2000,
 Deceleration := 2000,
 Jerk := 0,
 BufferMode := 0,
 Axis := Axis0,
 Done=> DoneMoveRel,
 Busy=> BusyMoveRel,
 Active=> ActiveMoveRel,
 CommandAborted=> ComMoveRel,
 Error=> ErrorMoveRel,
 ErrorID=>ErrorID3);
```

3

```
(*                         *)
 MC_MoveRelative2(
 Execute := StartMoveRel1,
 Distance := 20000,
 Velocity := 2000,
 Acceleration := 5000,
 Deceleration := 5000,
 Jerk := 0,
 BufferMode := 0,
 Axis := Axis0,
 Done=> DoneMoveRel1,
 Busy=> BusyMoveRel1,
 Active=> ActiveMoveRel1,
 CommandAborted=> ComMoveRel1,
 Error=> ErrorMoveRel1,
 ErrorID=>ErrorID4);
END_IF
```

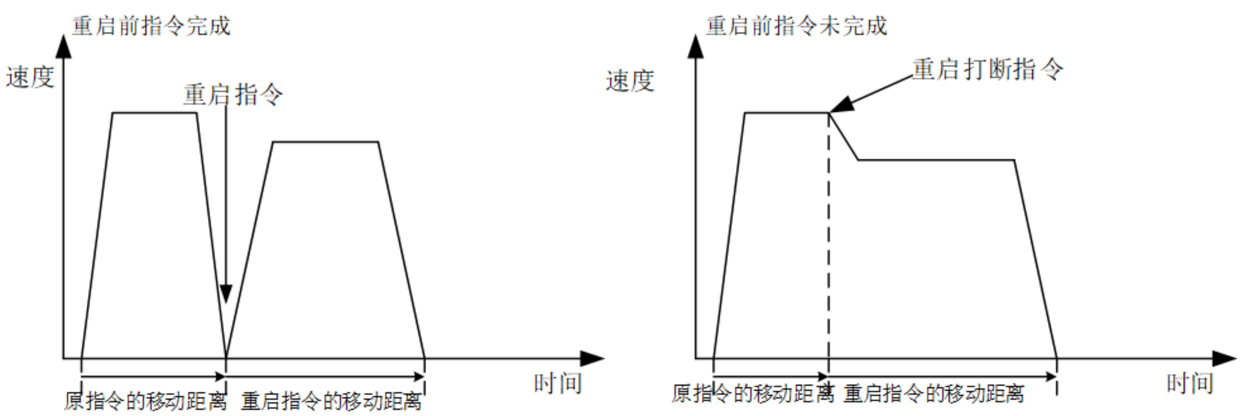
4

```
(*                                   *)
MC_Reset1(
Execute := StartReset,
Axis := Axis0,
```

```
Done=>Done4,
Busy=>Busy5,
Error=>Error5,
ErrorID=>ErrorID5);
```

5.8.4

Distance 0



5.8.5

MC_MoveAbsolute

BufferMode ( )

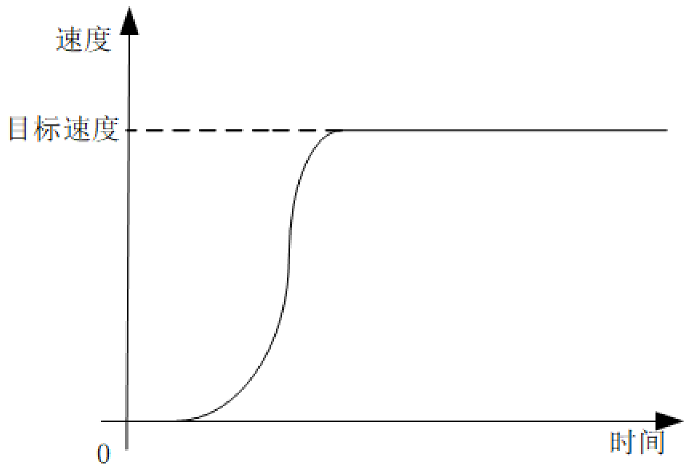
5.9 MC_MoveVelocity

	ErrorID			0	-	SMC_ERROR

5.9.2

Execute Velocity

Acceleration Deceleration Jerk



Direction

Direction mcPositiveDirection

Direction mcNegativeDirection

Direction mcCurrentDirection

BufferMode Aborting( )

Execute TRUE

BufferMode Aborting( ) Velocity

Acceleration Deceleration

BufferMode

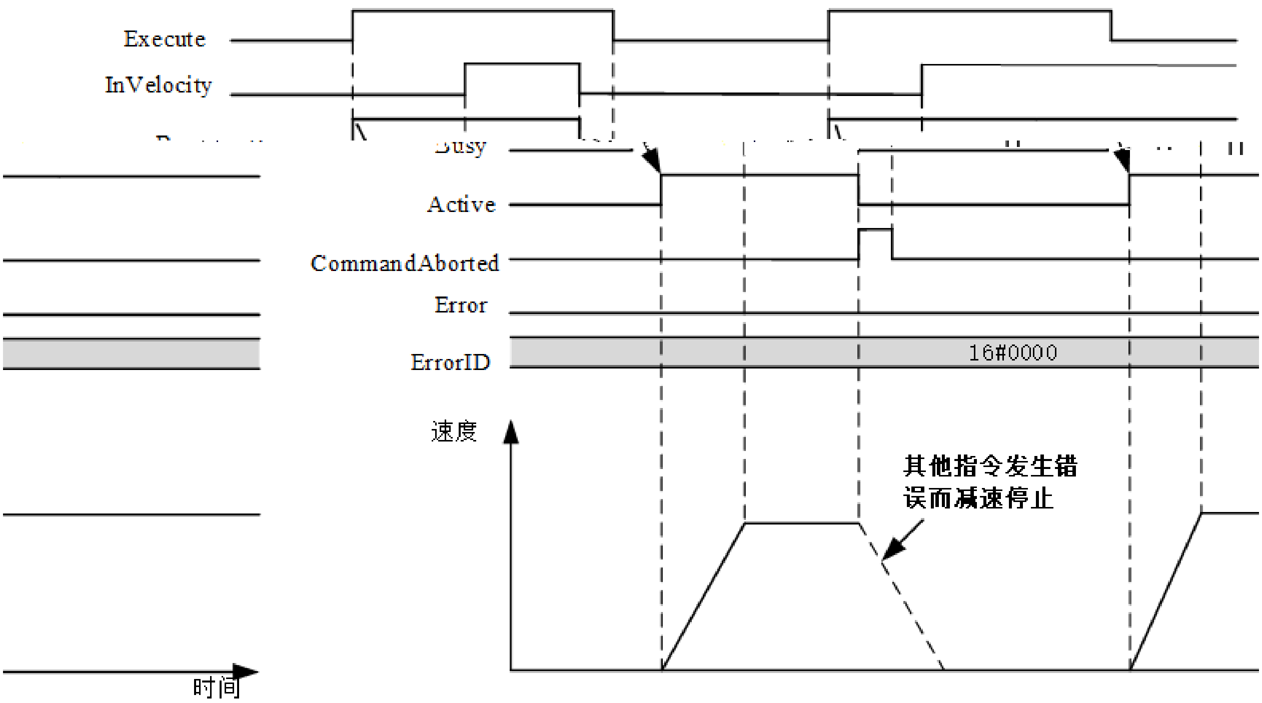
InVelocity( ) InVelocity(

) TRUE MC_SetOverride InVelocity( )

FALSE InVelocity( ) TRUE MC_SetOverride

InVelocity( ) TRUE

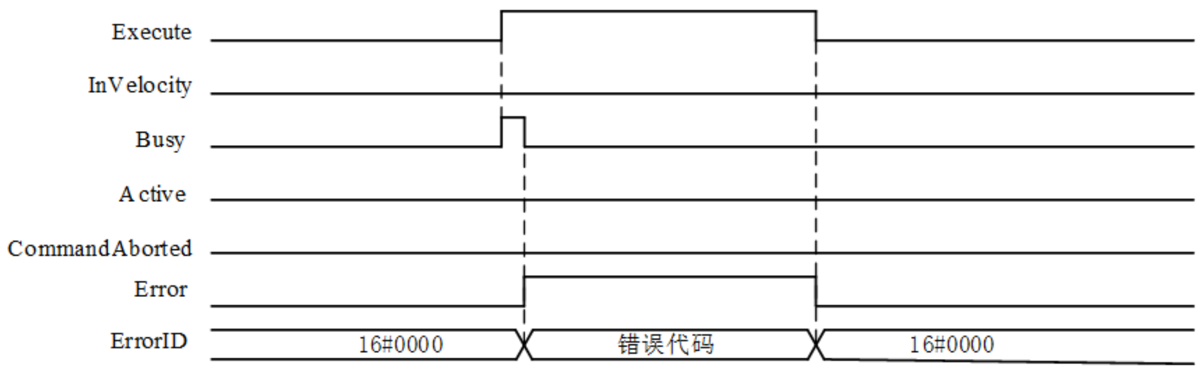
1



2

Error TRUE

ErrorID( )



5.9.3

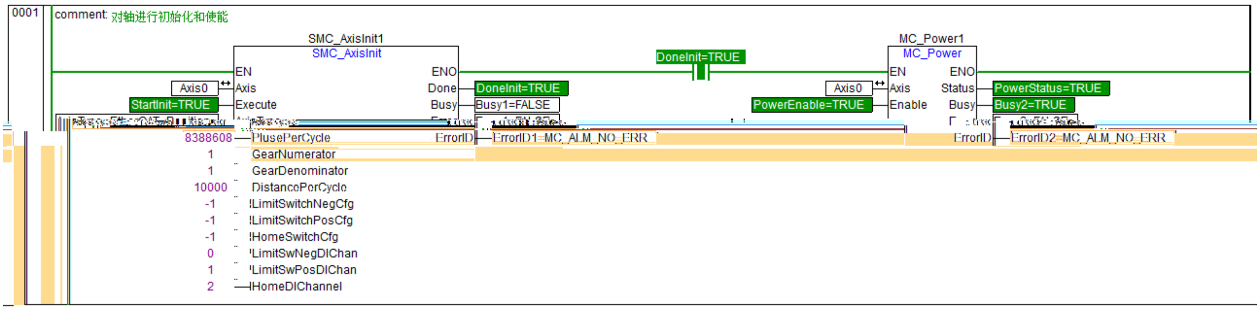
MC_MoveVelocity

Axis0	AXIS_REF		0
StartInit	BOOL	FALSE	TRUE
DoneInit	BOOL	FALSE	TRUE
PowerEnable	BOOL	FALSE	TRUE
PowerStatus	BOOL	FALSE	TRUE
StartMoveVel	BOOL	FALSE	TRUE
BusyMoveVel	BOOL	FALSE	TRUE
ActiveMoveVel	BOOL	FALSE	TRUE
InVelMoveVel	BOOL	FALSE	TRUE
ComMoveVel	BOOL	FALSE	TRUE
ErrorMoveVel	BOOL	FALSE	TRUE
StartReset	BOOL	FALSE	TRUE

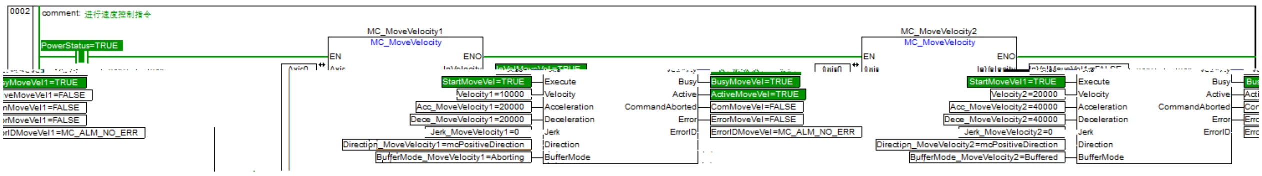
Velocity 10000 Acceleration Deceleration 20000 Jerk 0 Direction BufferMode Aborting( )

1

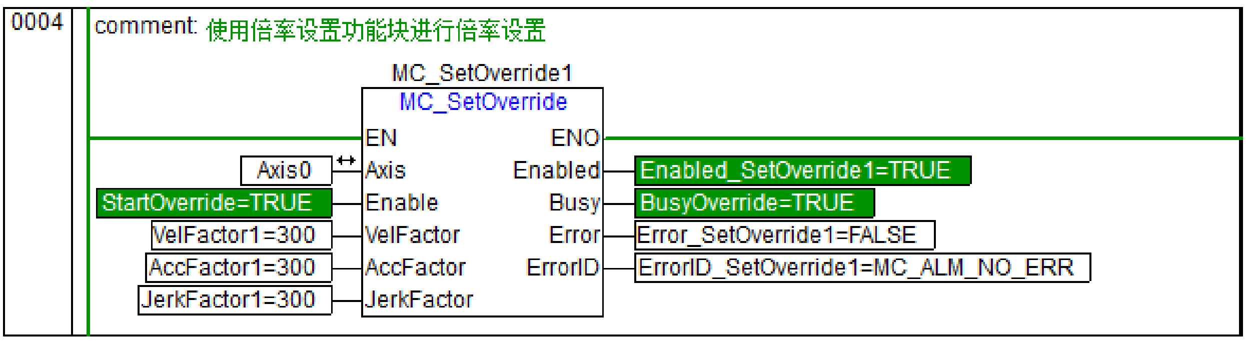
ON



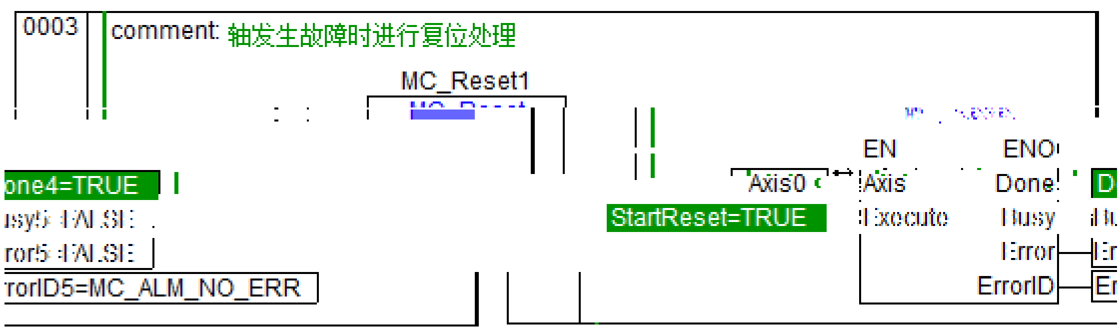
2



3 MC_SetOverride InVelocity



4 MC_Reset



ST

1

(* *)

```
SMC_AxisInit1(
Execute := StartInit,
AxisType := 50,
PlusePerCycle := 10000,
GearNumerator := 1,
GearDenominator := 1,
DistancePerCycle := 10000,
LimitSwitchNegCfg := -1,
LimitSwitchPosCfg := -1,
HomeSwitchCfg := -1,
LimitSwNegDIChan := 0,
LimitSwPosDIChan := 1,
HomeDIChannel := 2,
Axis := Axis0,
Done=> DoneInit,
Busy=>Busy1,
Error=>Error1,
ErrorID=>ErrorID1);
(*                    *)
IF DoneInit THEN
    MC_Power1(
    Enable := PowerEnable,
    Axis := Axis0,
    Status=> PowerStatus,
    Busy=>Busy2,
    Error=>Error2,
    ErrorID=>ErrorID2);
END_IF
```

2 MC_MoveAbsolute

```
IF PowerStatus THEN
(*                          *)
MC_MoveAbsolute1(
Execute := StartMoveAbs,
Position := 10000,
Velocity := 1000,
Acceleration := 2000,
```

```
  Busy=>Busy5,
  Error=>Error5,
  ErrorID=>ErrorID5);
END_IF
```

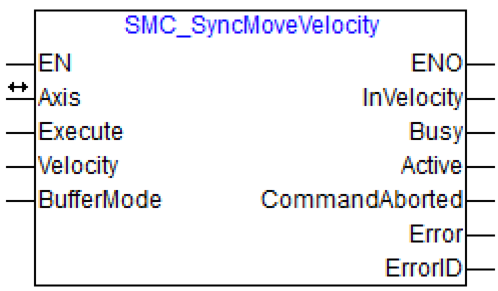
5.9.4

Direction mcCurrentDirection

5.9.5

BufferMode ( )

5.10 SMC_SyncMoveVelocity



5.10.1

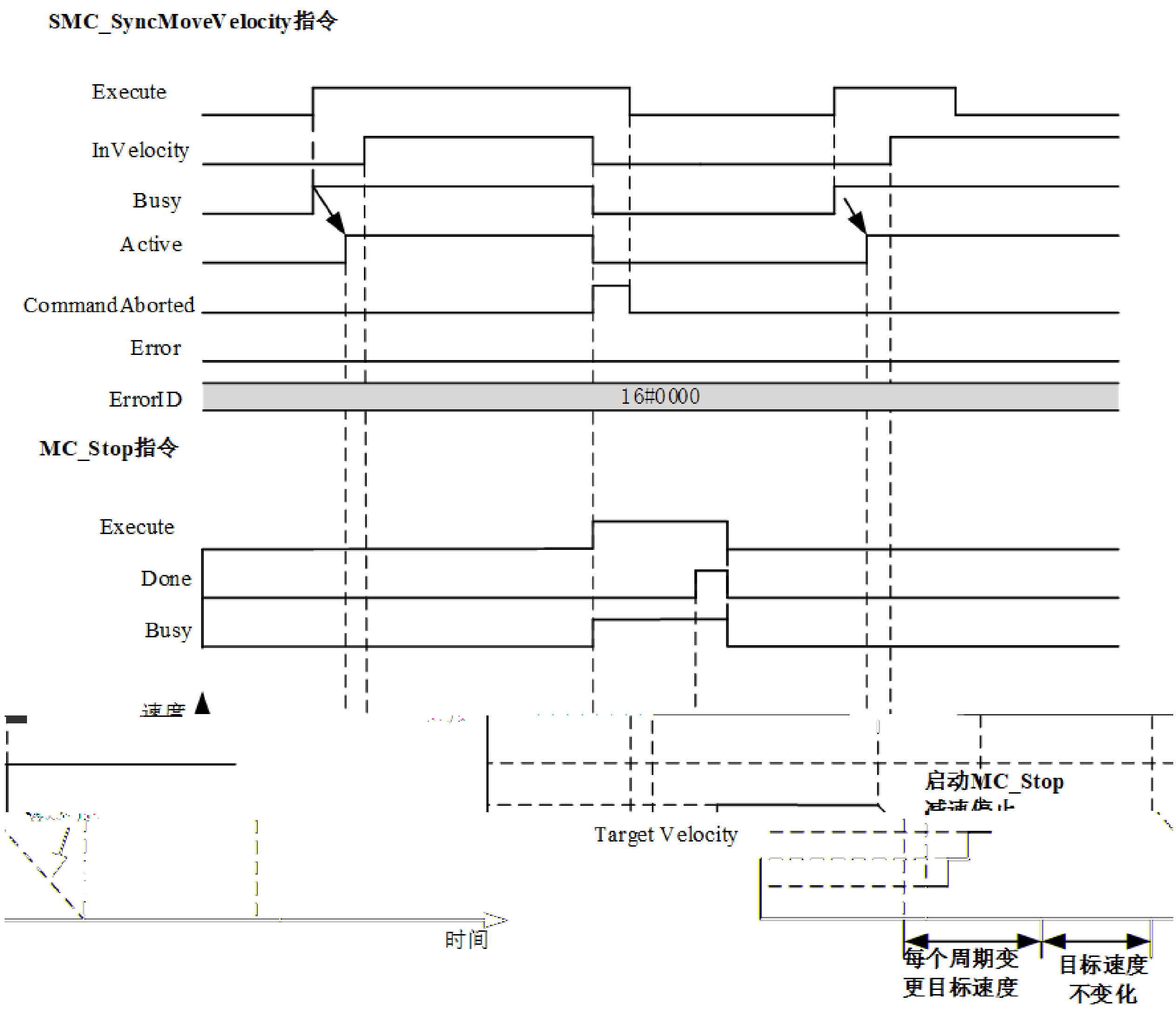
IN_OUT	Axis			-	-	AXIS_REF
INPUT	Execute			-	TRUE/FALSE	BOOL
	Velocity			-	/ /0	LREAL
	BufferMode	0 1		0	0: Aborting 1: Buffered	M248.28 n24 87.84 16

0

BufferMode( )

MC_Stop

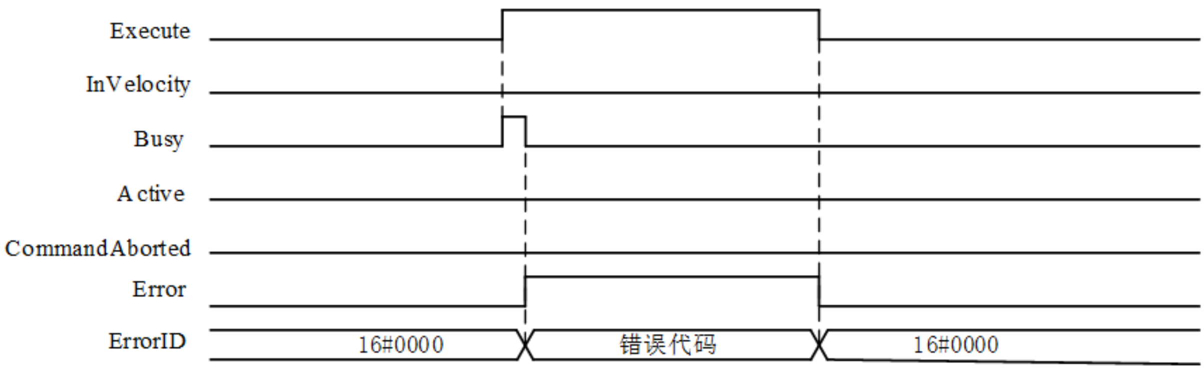
1



2

Error( ) TRUE FALSE

ErrorID( )



5.10.3

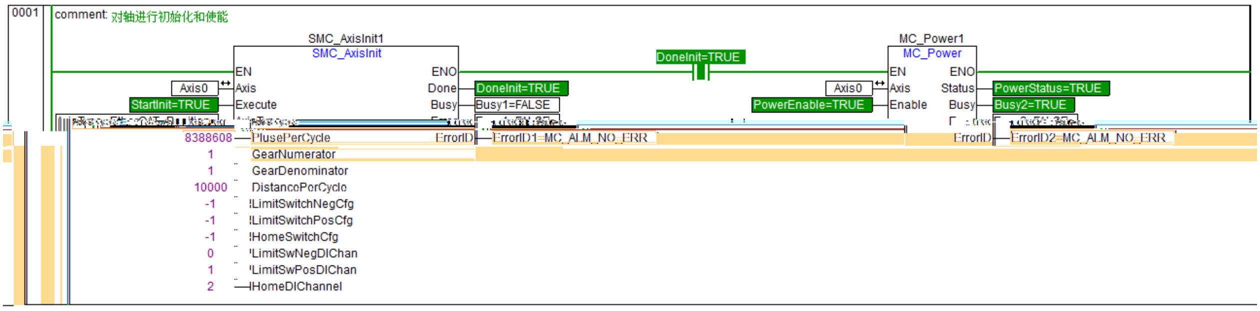
SMC_SyncMoveVelocity

Axis0	AXIS_REF		0
StartInit	BOOL	FALSE	TRUE
DoneInit	BOOL	FALSE	TRUE
PowerEnable	BOOL	FALSE	TRUE
PowerStatus	BOOL	FALSE	TRUE
StartSyncMove	BOOL	FALSE	TRUE
BusySyncMove	BOOL	FALSE	TRUE
ActiveSyncMove	BOOL	FALSE	TRUE
InVelocitySyncMove	BOOL	FALSE	TRUE
ComSyncMove	BOOL	FALSE	TRUE
ErrorSyncMove	BOOL	FALSE	TRUE
StartReset	BOOL	FALSE	TRUE

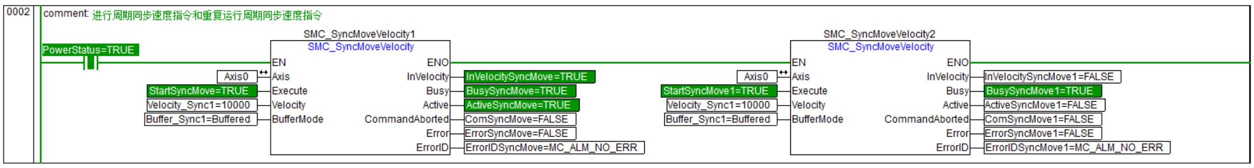
Velocity 10000 BufferMode Aborting( )

1

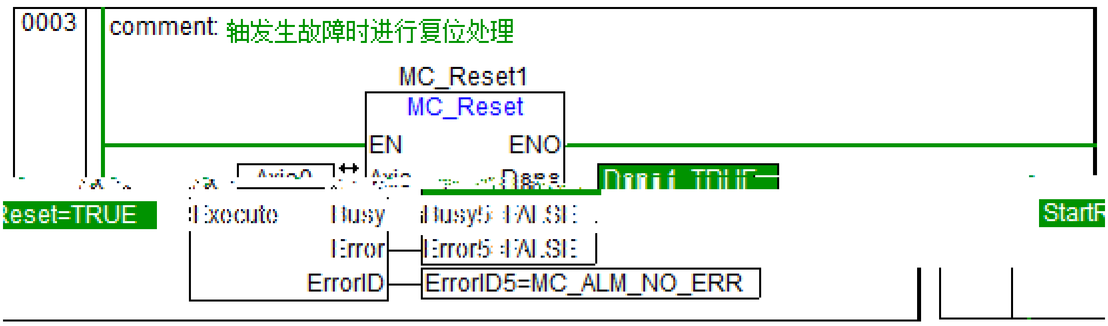
ON



2



3 MC_Reset



ST

1

```
SMC_AxisInit1(
Execute := StartInit,
AxisType := 50,
PlusePerCycle := 10000,
GearNumerator := 1,
GearDenominator := 1,
DistancePerCycle := 10000,
LimitSwitchNegCfg := -1,
```

```
HomeDIChannel := 2,
Axis := Axis0,
Done=> DoneInit,
Busy=>Busy1,
Error=>Error1,
ErrorID=>ErrorID1);
```

2

```
IF DoneInit THEN
   MC_Power1(
   Enable := PowerEnable,
   Axis := Axis0,
   Status=> PowerStatus,
   Busy=>Busy2,
   Error=>Error2,
   ErrorID=>ErrorID2);
END_IF
```

3

```
IF PowerStatus THEN
SMC_SyncMoveVelocity1(
Execute := StartSyncMove,
Velocity := 10000,
BufferMode := Buffer_Sync1,
Axis := Axis0,
InVelocity=>InVelocitySyncMove,
Busy=>BusySyncMove,
Active=>ActiveSyncMove,
CommandAborted=>ComSyncMove,
Error=>ErrorSyncMove,
ErrorID=>ErrorIDSyncMove);
```

4

```
   SMC_SyncMoveVelocity2(
   Execute := StartSyncMove1,
   Velocity := 10000,
   BufferMode := Buffer_Sync1,
   Axis := Axis0,
   InVelocity=>InVelocitySyncMove1,
```

```
   Busy=>BusySyncMove1,
   Active=>ActiveSyncMove1,
   CommandAborted=>ComSyncMove1,
   Error=>ErrorSyncMove1,
   ErrorID=>ErrorIDSyncMove1);
END_IF
```

5

```
MC_Reset1(
Execute := StartReset,
Axis := Axis0,
Done=>Done4,
Busy=>Busy5,
Error=>Error5,
ErrorID=>ErrorID5);
```

5.10.4

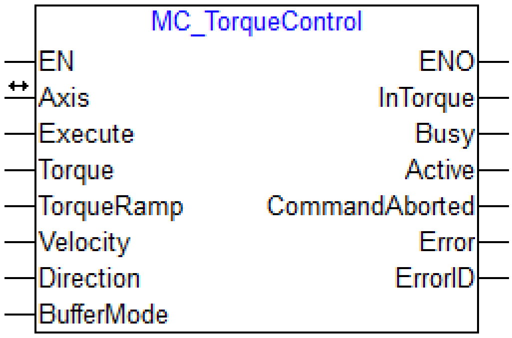
EtherCAT

MC_SetOverride

5.10.5

BufferMode ( )

5.11 MC_TorqueControl



5.11.1

IN_OUT	Axis			-	-	AXIS_REF
INPUT	Execute			-	TRUE/FALSE	BOOL
	Torque	1%		300	0 ~ 1000	LREAL
	TorqueRamp	1% /		-		LREAL
	Velocity			-		LREAL
	Direction	0: 2:		0	0: mcPositiveDirection 2: mcNegativeDirection	MC_DIRECTION
	BufferMode	0 1		0	0: Aborting 1: Buffered	MC_BUFFER_MODE
OUTPUT	InTorque	5%		OFF	TRUE/FALSE	BOOL
	Busy			FALSE	TRUE/FALSE	BOOL
	Active	TRUE		FALSE	TRUE/FALSE	BOOL
	CommandAborted			FALSE	TRUE/FALSE	BOOL
	Error			FALSE	TRUE/FALSE	BOOL
	ErrorID			0	-	SMC_ERROR

5.11.2

EtherCAT

CST CST

Execute Torque TorqueRamp Velocity Direction

Execute TRUE Continuous Motion

0x607F 0x800

Velocity

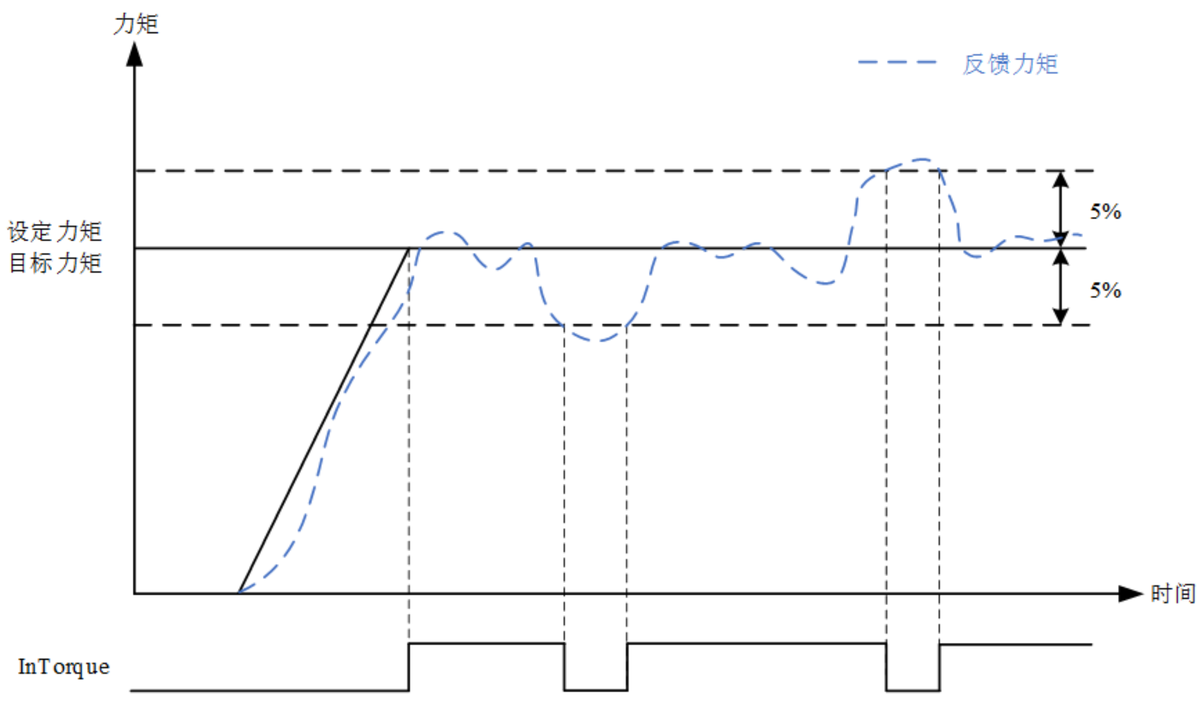
0x607F

0x607F

4

5

5%



0x800

Controlword (6040 hex)

Statusword (6041 hex)

Mode of Operation (6060 hex)

Mode of Operation Display (6061 hex)

Target Torque(6071 hex)

Torque Actual Value (6077 hex)

Velocity Actual Value (606C hex)

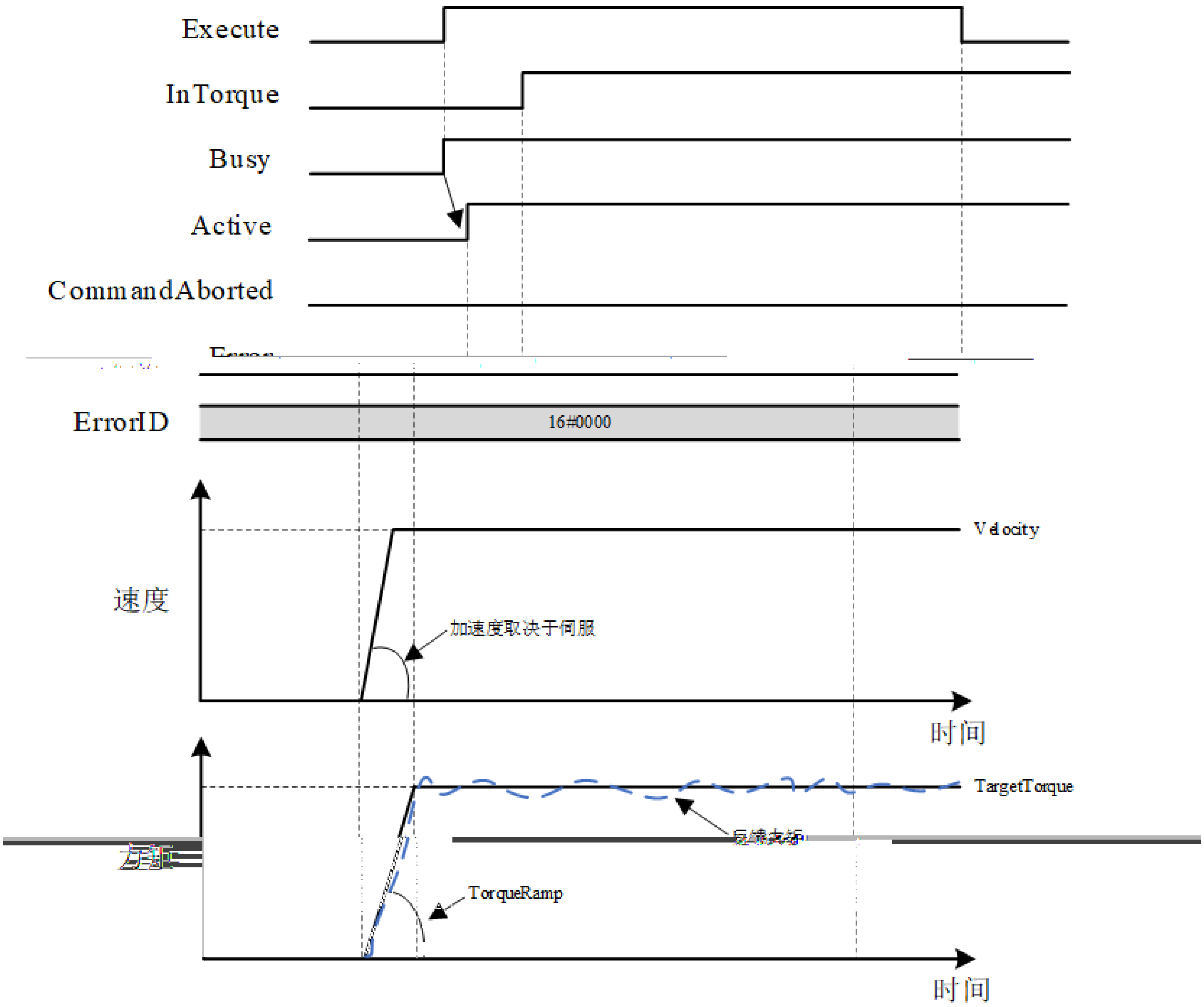
MC_Stop

TorqueRamp 0

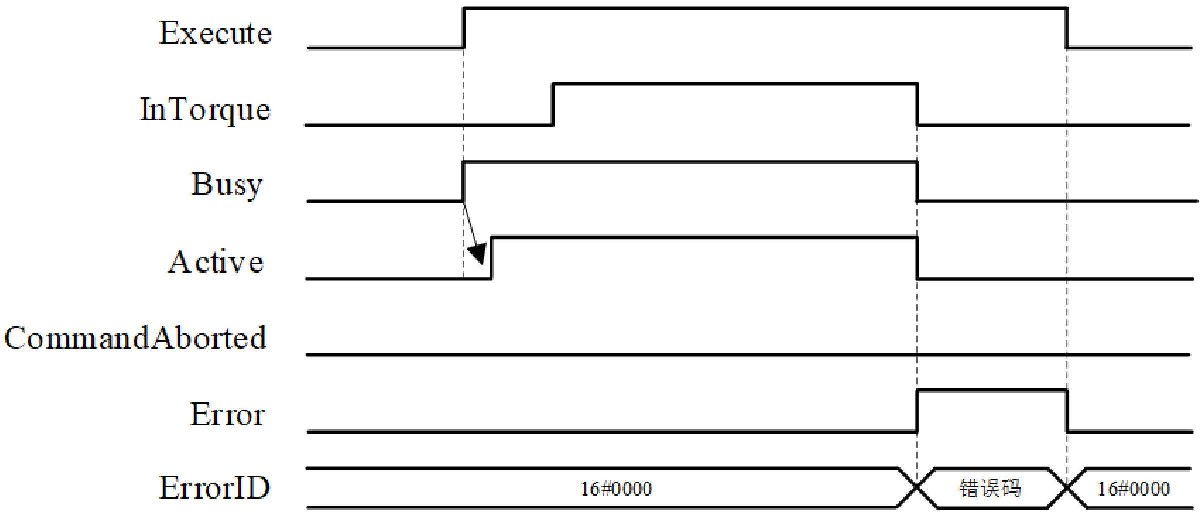
MC_Stop MC_TorqueControl ContinuousMotion Stopping

MC_TorqueControl CommandAbort

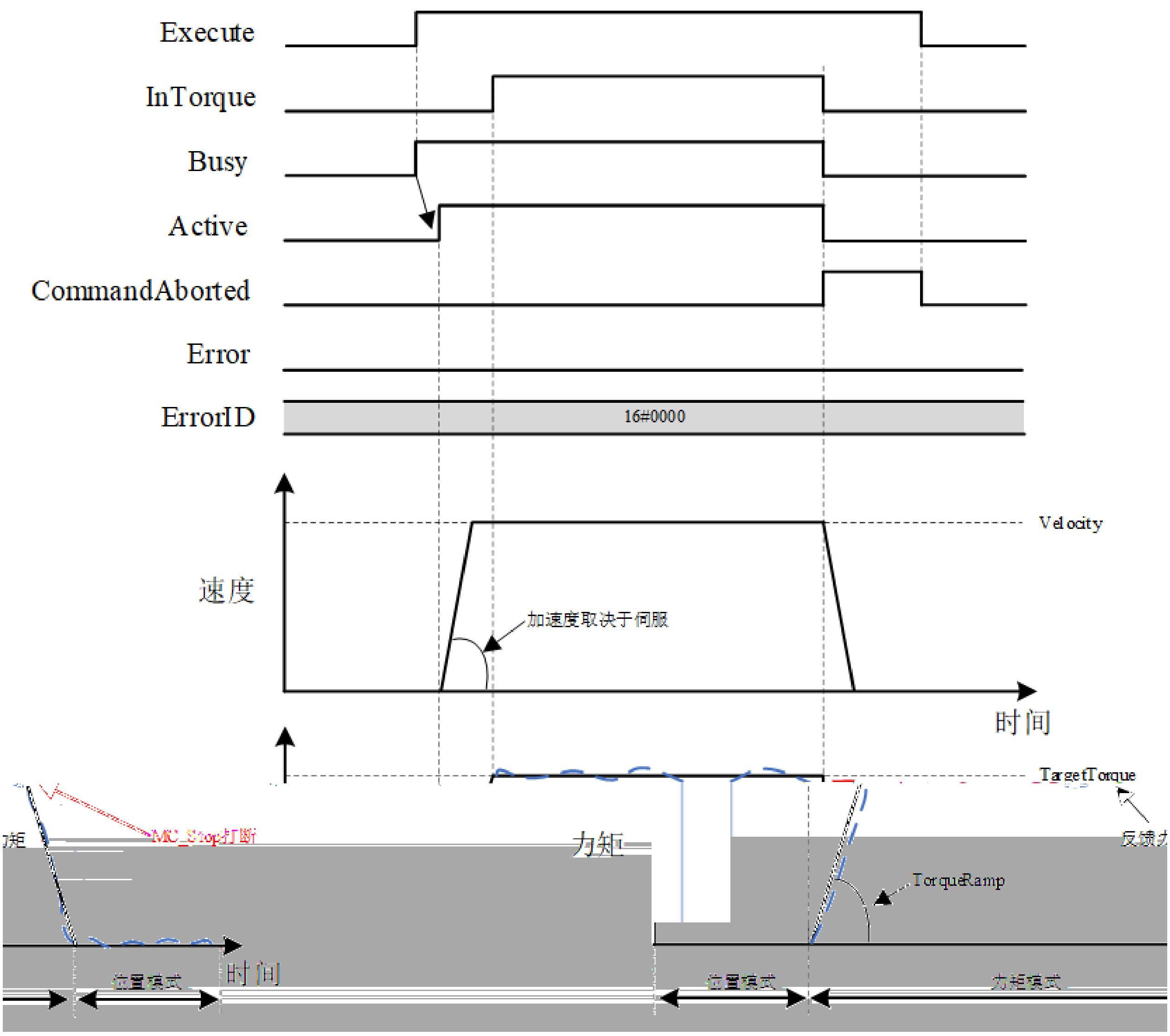
1


Execute
InTorque
Busy
Active
CommandAborted
ErrorID
16#0000
速度
Velocity
加速度取决于伺服
时间
TargetTorque
TorqueRamp
时间

2


Execute
InTorque
Busy
Active
CommandAborted
Error
ErrorID
16#0000
错误码
16#0000

3 MC_Stop



4 Execute MC_Stop

5.11.3

MC_TorqueControl MC_Stop

PDO

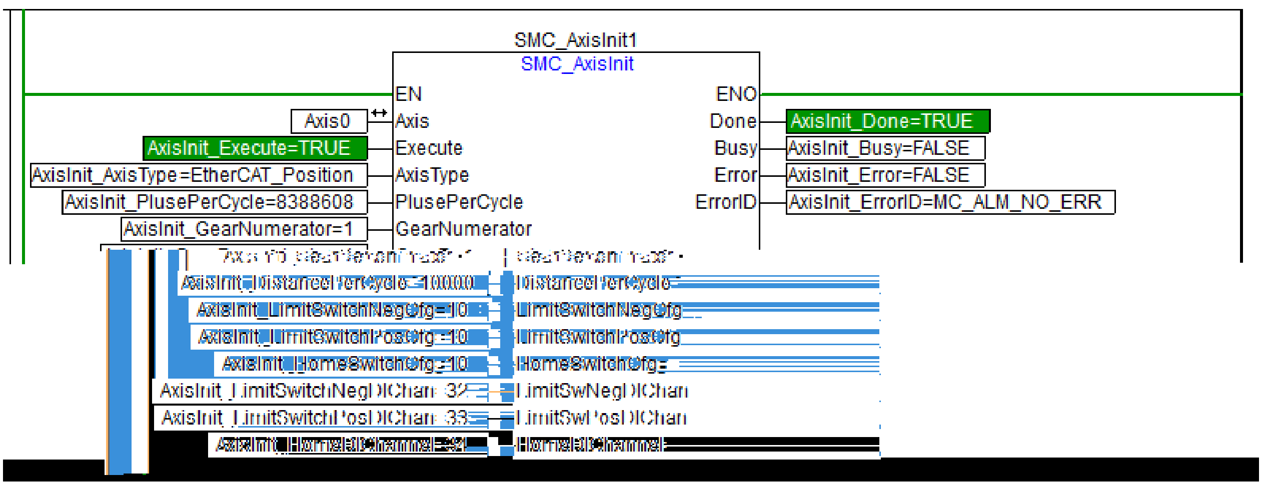
SMC_AxisInit MC_Power

1	MC_TorqueControl0	MC_TORQUECONTROL	-	
2	TorqueControl_En	BOOL	FALSE	
3	TorqueControl_Torque	LREAL	0	
4	TorqueControl_TorqueRamp	LREAL	0	
5	TorqueControl_Vel	LRE AL	0	
6	TorqueControl_Dir	MC_DIRECTION	moPositiveDirection	
7	TorqueControl_Buffer	MC_BUFFER_MODE	Aborting	
8	TorqueControl_InTorque	BOOL	FALSE	
9	TorqueControl_Busy	BOOL	FALSE	

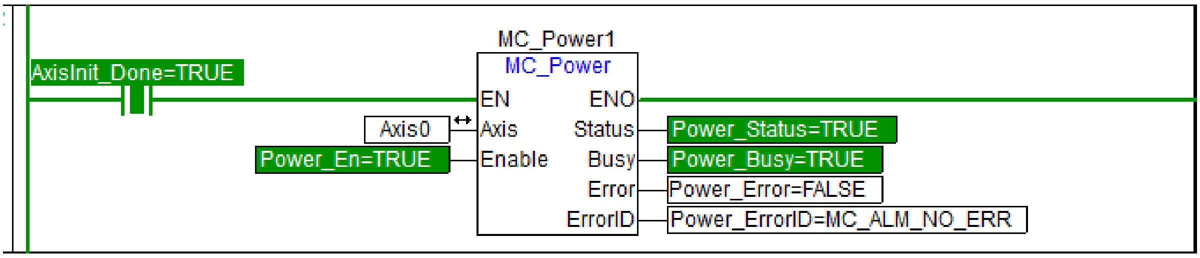
10	TorqueControl_Active	BOOL	FALSE	
11	TorqueControl_Abort	BOOL	FALSE	
12	TorqueControl_Error	BOOL	FALSE	
13	TorqueControl_ErrorID	SMC_ERROR	MC_ALM_NO_ERR	
14	MC_ReadActualTorque1	MC_READACTUALTORQUE	-	
15	ReadTorque_En	BOOL	FALSE	
16	ReadTorque_Valid	BOOL	FALSE	
17	ReadTorque_Busy	BOOL	FALSE	
18	ReadTorque_Error	BOOL	FALSE	
19	ReadTorque_ErrorID	SMC_ERROR	MC_ALM_NO_ERR	
20	ReadTorque_Torque	LREAL	0	
21	MC_Stop1	MC_STOP	-	
22	Stop_En	BOOL	FALSE	
23	Stop_Dec	LREAL	0	
24	Stop_Jerk	LREAL	0	
25	Stop_Done	BOOL	FALSE	
26	Stop_Busy	BOOL	FALSE	
27	Stop_Error	BOOL	FALSE	
28	Stop_ErrorID	SMC_ERROR	MC_ALM_NO_ERR	

LD

1 SMC_AxisInit

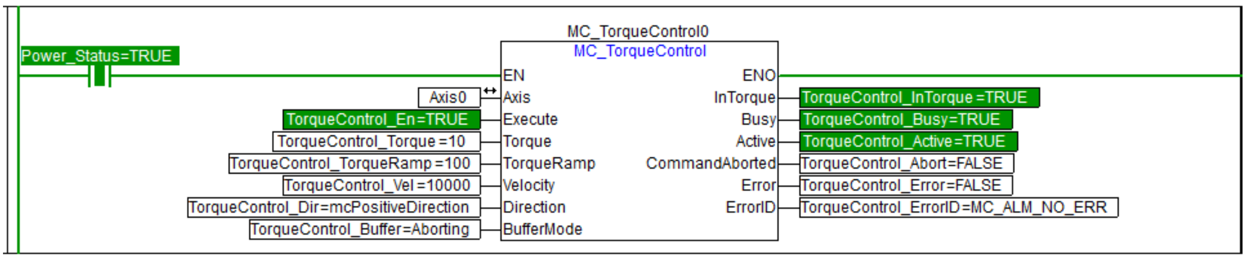


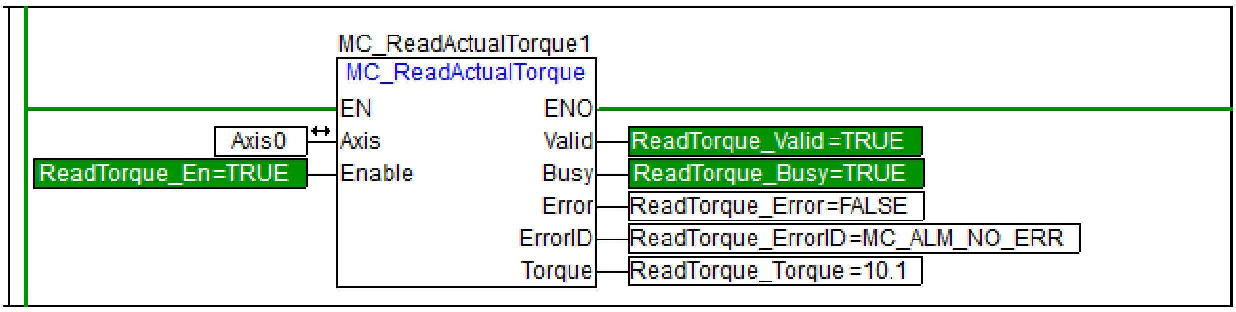
2 MC_Power



3 MC_TorqueControl

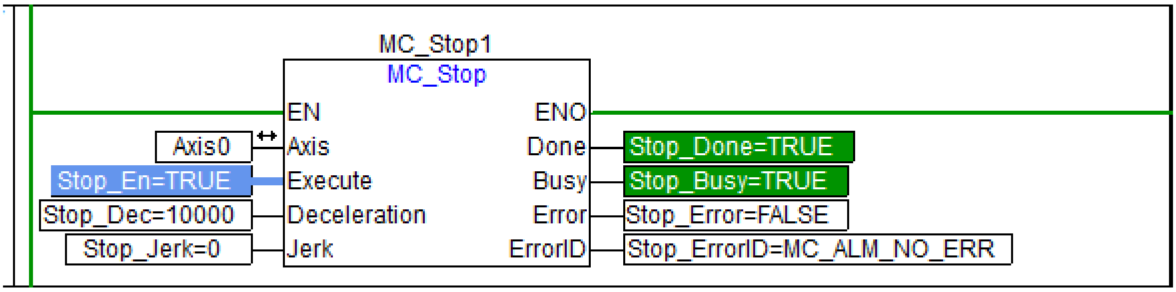
MC_ReadActualTorque

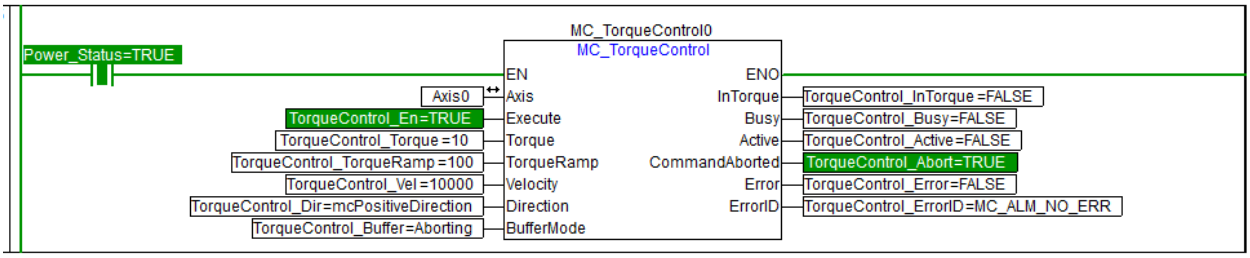




4 MC_Stop

MC_TorqueControl CommandAborted





ST

1 SMC_AxisInit

```
SMC_AxisInit1(
Execute := AxisInit_Execute,
AxisType := AxisInit_AxisType,
PlusePerCycle := AxisInit_PlusePerCycle,
GearNumerator := AxisInit_GearNumerator,
GearDenominator := AxisInit_GearDenominator,
DistancePerCycle := AxisInit_DistancePerCycle,
LimitSwitchNegCfg := AxisInit_LimitSwitchNegCfg,
LimitSwitchPosCfg := AxisInit_LimitSwitchPosCfg,
HomeSwitchCfg := AxisInit_HomeSwitchCfg,
LimitSwNegDIChan := AxisInit_LimitSwitchNegDIChan,
LimitSwPosDIChan := AxisInit_LimitSwitchPosDIChan,
HomeDIChannel := AxisInit_HomeDIChannel,
Axis := Axis0,
Done=>AxisInit_Done,
Busy=>AxisInit_Busy,
Error=>AxisInit_Error,
ErrorID=>AxisInit_ErrorID);
```

2 MC_Power

```
IF TRUE = AxisInit_Done THEN
   Power_En := TRUE;
```

```
ELSE
   Power_En := FALSE;
END_IF;
MC_Power1(
Enable := Power_En,
Axis := Axis0,
Status=>Power_Status,
Busy=>Power_Busy,
Error=>Power_Error,
ErrorID=>Power_ErrorID);
```

3 MC_TorqueControl

MC_ReadActualTorque

```
IF TRUE = Power_Status THEN
   MC_TorqueControl0(
   Execute := TorqueControl_En,
   Torque := TorqueControl_Torque,
   TorqueRamp := TorqueControl_TorqueRamp,
   Velocity := TorqueControl_Vel,
   Direction := TorqueControl_Dir,
   BufferMode := TorqueControl_Buffer,
   Axis := Axis0,
   InTorque=>TorqueControl_InTorque,
   Busy=>TorqueControl_Busy,
   Active=>TorqueControl_Active,
   CommandAborted=>TorqueControl_Abort,
   Error=>TorqueControl_Error,
   ErrorID=>TorqueControl_ErrorID);
END_IF
MC_ReadActualTorque1(
Enable := ReadTorque_En,
Axis := Axis0,
Valid=>ReadTorque_Valid,
Busy=>ReadTorque_Busy,
Error=>ReadTorque_Error,
ErrorID=>ReadTorque_ErrorID,
Torque=>ReadTorque_Torque);
```

4 MC_Stop

MC_TorqueControl CommandAborted

```
MC_Stop1(
```

	Busy			FALSE	TRUE/FALSE	BOOL
	Error			FALSE	TRUE/FALSE	BOOL
	ErrorID			0	-	SMC_ERROR

5.12.2

CIA402

Enable( ) TRUE PositiveEnable( ) TRUE PositiveValue( ) NegativeEnable( ) TRUE NegativeValue( )

Enable( ) FALSE

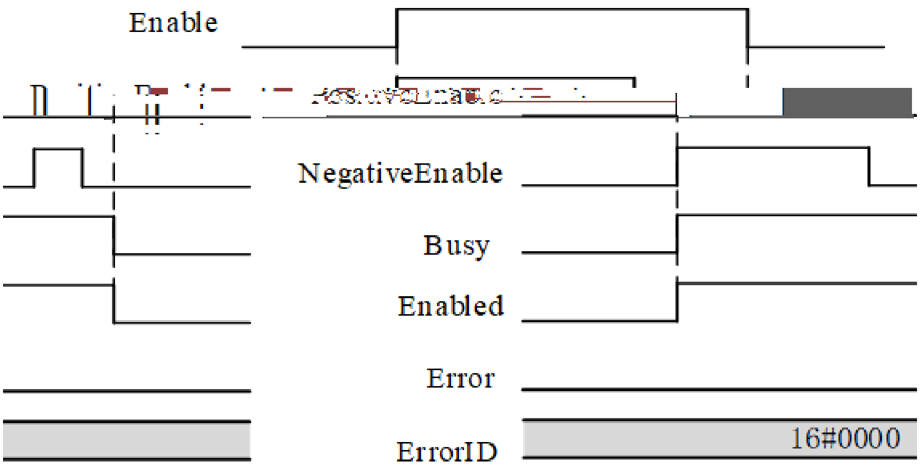
PositiveEnable( ) NegativeEnable( ) FALSE

1%

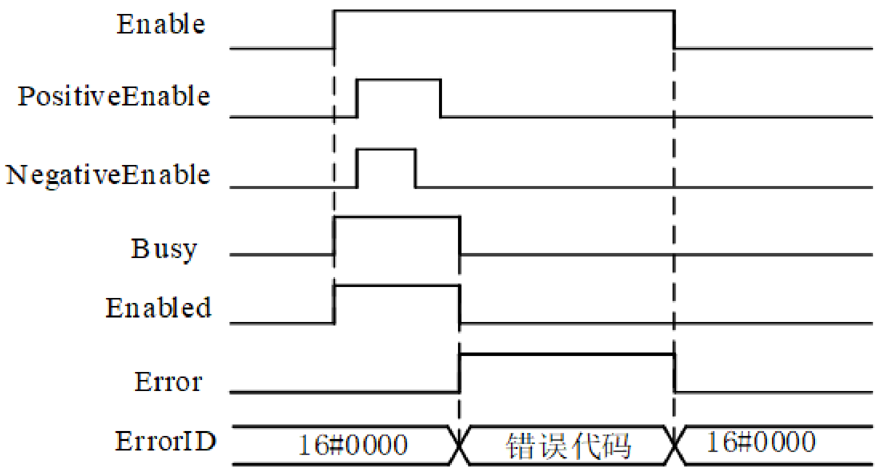
Enable Enable TRUE

PositiveEnable( ) NegativeEnable( ) TRUE

1



2



5.12.3

SMC_SetTorqueLimit /

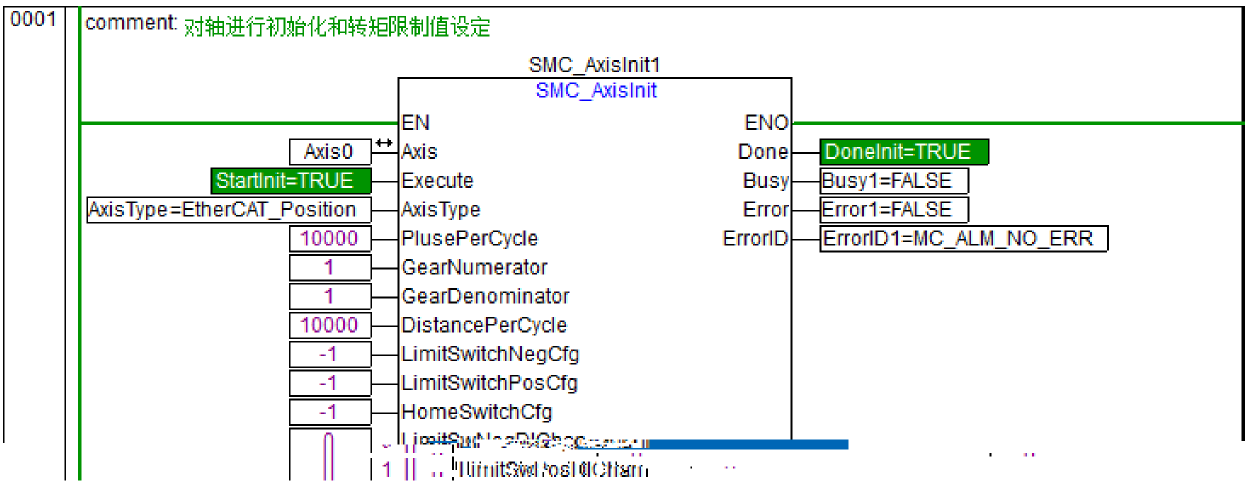
Axis0	AXIS_REF		0
StartInit	BOOL	FALSE	TRUE
DoneInit	BOOL	FALSE	TRUE
SetTorqEnable	BOOL	FALSE	TRUE
PosSetTorqEnable	BOOL	FALSE	TRUE
NegSetTorqEnable	BOOL	FALSE	TRUE

EnabledSetTor	BOOL	FALSE	TRUE
BusySetTor	BOOL	FALSE	TRUE
ErrorSetTor	BOOL	FALSE	TRUE

PositiveValue NegativeValue 300 SMC_SetTorqueLimit

Enable( ) TRUE

1



2 /

ST

1

```
SMC_AxisInit1(
Execute := StartInit,
AxisType := 50,
```

```
PlusePerCycle := 10000,
GearNumerator := 1,
GearDenominator := 1,
DistancePerCycle := 10000,
LimitSwitchNegCfg := -1,
LimitSwitchPosCfg := -1,
HomeSwitchCfg := -1,
LimitSwNegDIChan := 0,
LimitSwPosDIChan := 1,
HomeDIChannel := 2,
Axis := Axis0,
Done=> DoneInit,
Busy=>Busy1,
Error=>Error1,
ErrorID=>ErrorID1);
```

2

```
SMC_SetTorqueLimit1(
Enable := SetTorqEnable,
PositiveEnable := PosSetTorqEnable,
PositiveValue := 300,
NegativeEnable := NegSetTorqEnable,
NegativeValue := 300,
Axis := Axis0,
Enabled=>EnabledSetTor,
Busy=>BusySetTor,
Error=>ErrorSetTor,
ErrorID=>ErrorID_SetTorq1);
```

5.12.4

350 300

5.12.5

MC_TorqueControl

5.13 MC_SetPosition

5.13.1

IN_OUT	Axis			-	-	AXIS_REF
INPUT	Execute			-	TRUE/FALSE	BOOL
	Position			-	/ /0	LREAL
	Relative	FALSE Position TRUE Position		FALSE	TRUE/FALSE	BOOL
OUTPUT	Done			FALSE	TRUE/FALSE	BOOL
	Busy			FALSE	TRUE/FALSE	BOOL
	Error			FALSE	TRUE/FALSE	BOOL
	ErrorID			0	-	SMC_ERROR

5.13.2

1

Execute Axis ID

Position Relative

2

Relative=FALSE

Position

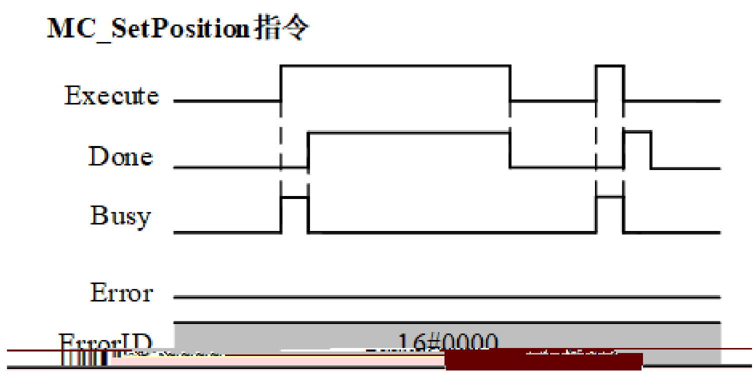
Relative=TRUE

Position

1

Execute Busy TRUE Done

TRUE Busy FALSE Done

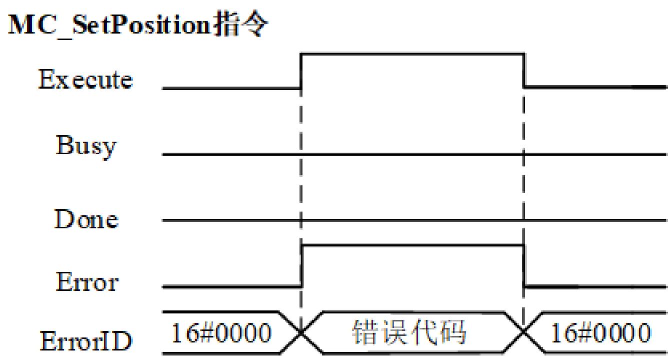


2

Unused

Error=True ErrorID

Execute



- Execute MC_SetPosition

5.13.3

MC_SetPosition

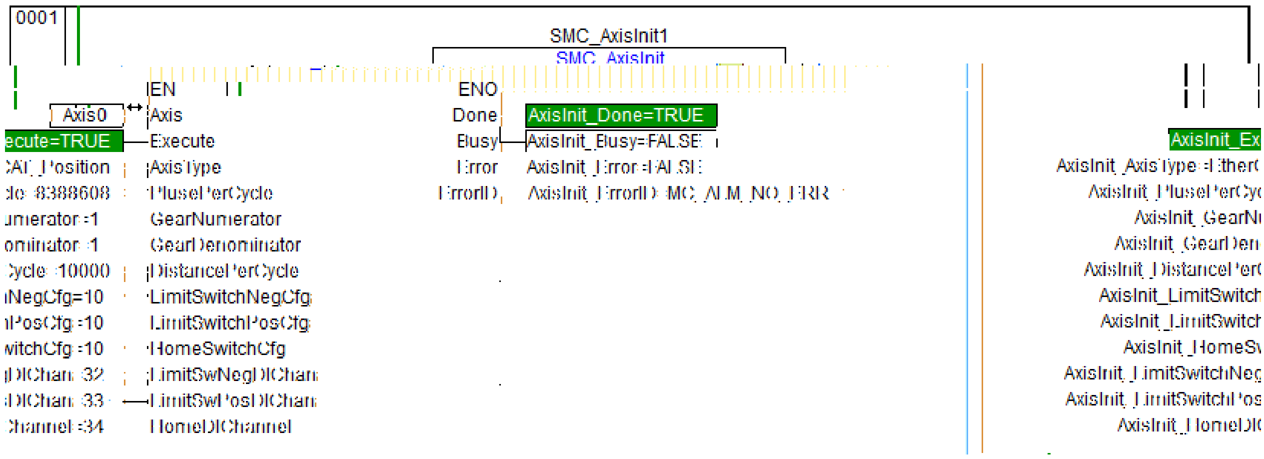
Position 0 Relative FALSE 0

Axis0	AXIS_REF	-	0
SetPos_Done	BOOL	FALSE	TRUE
Switch1	BOOL	FALSE	Switch TRUE

1

Axis0.AxisID = 0 50 EtherCAT_Position SMC_AxisInit

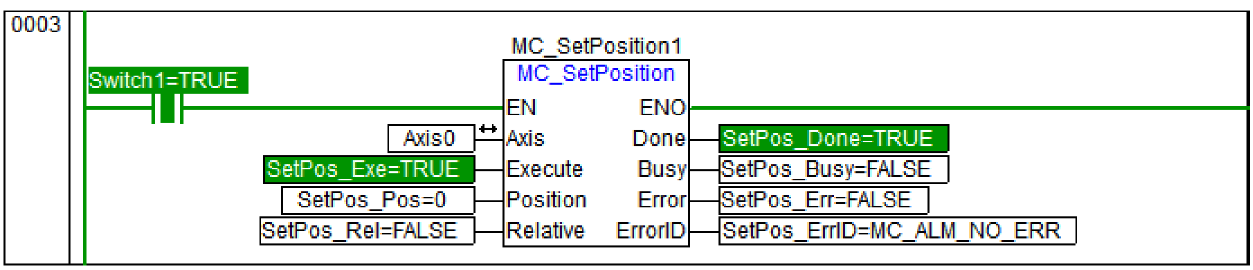
SMC_AxisInit



2 MC_SetPosition

MC_SetPosition Axis0

Switch1=TRUE



ST

ST Position 10000 Relative TRUE

10000

1

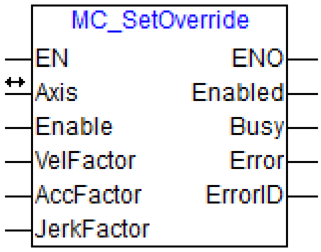
ST

```
GearNumerator := AxisInit_GearNumerator,
GearDenominator := AxisInit_GearDenominator,
DistancePerCycle := AxisInit_DistancePerCycle,
LimitSwitchNegCfg := AxisInit_LimitSwitchNegCfg,
LimitSwitchPosCfg := AxisInit_LimitSwitchPosCfg,
HomeSwitchCfg := AxisInit_HomeSwitchCfg,
LimitSwNegDIChan := AxisInit_LimitSwitchNegDIChan,
LimitSwPosDIChan := AxisInit_LimitSwitchPosDIChan,
HomeDIChannel := AxisInit_HomeDIChannel,
Axis := Axis0,
Done=>AxisInit_Done,
Busy=>AxisInit_Busy,
Error=>AxisInit_Error,
ErrorID=>AxisInit_ErrorID);
```

3 MC_SetPosition

```
  Relative:=TRUE;(*                    *)

  Position:=10000; (*                  *)
IF  Switch1 THEN
  MC_SetPosition1(
  Execute := Exe_SetPos,
  Position := Position,
  Relative := Relative,
  Axis := Axis0,
  Done=>Done_Pos,
  Busy=>Busy_Pos,
```



5.14.1

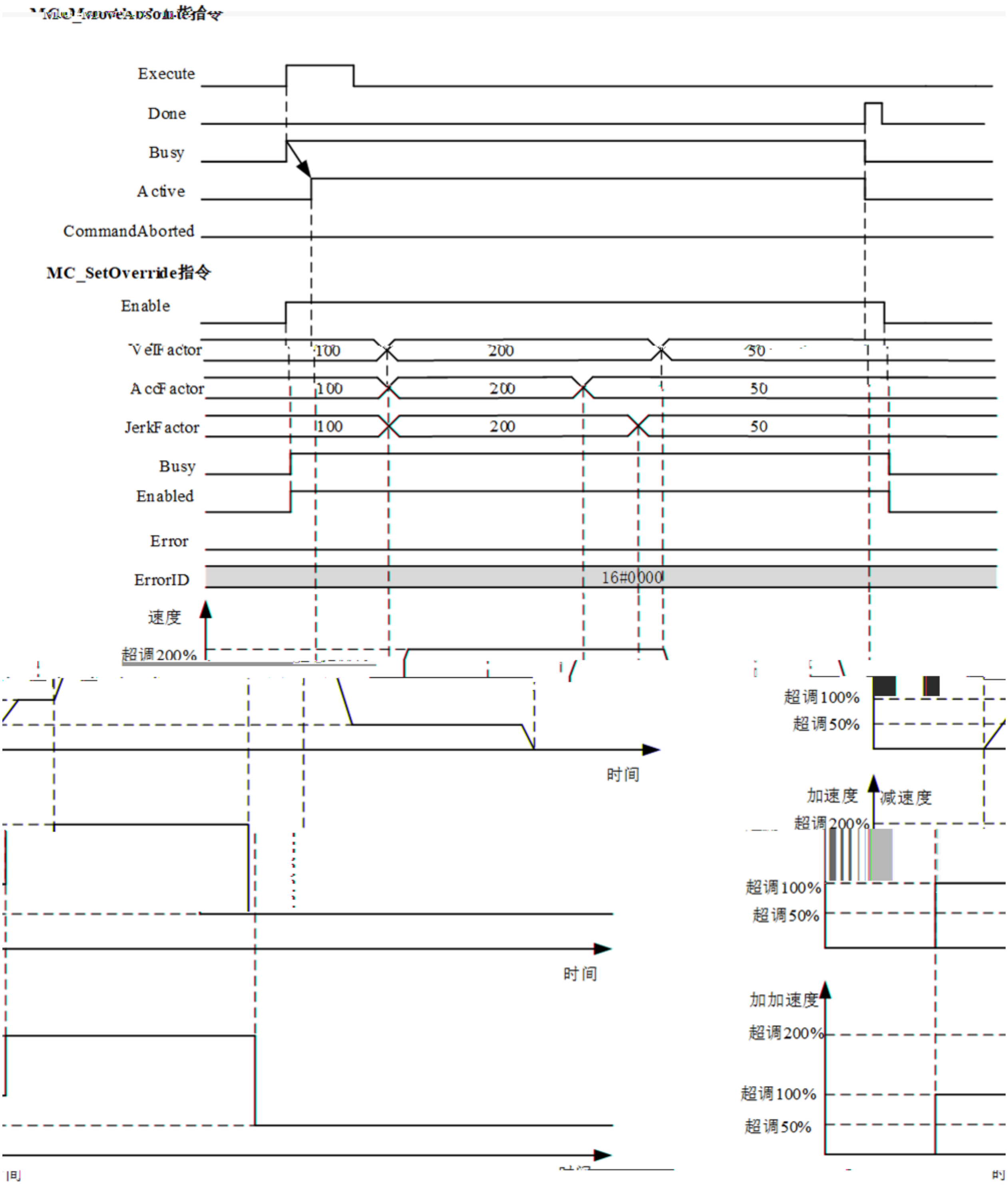
IN_OUT	Axis			-	-	AXIS_REF
	Enable			-	TRUE/FALSE	BOOL

INPUT

Enable( ) TRUE Enable( ) FALSE

1 MC_MoveAbsolute( )

Enable( ) TRUE Busy( ) Enabled( ) TRUE Enable( ) FALSE Busy( ) Enabled( ) FALSE MC_MoveAbsolute( )

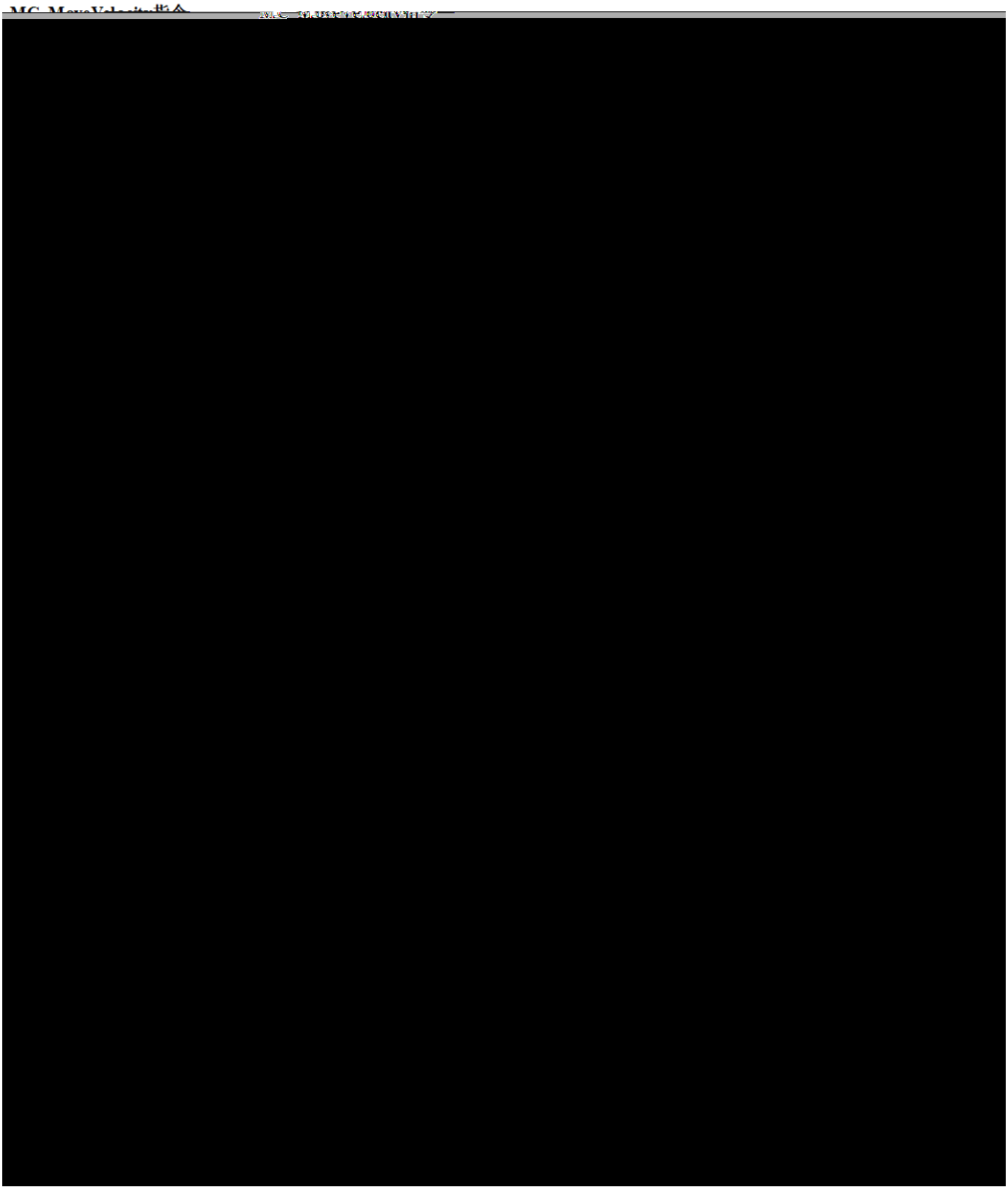


2 MC_MoveVelocity( ) InVelocity( ) TRUE

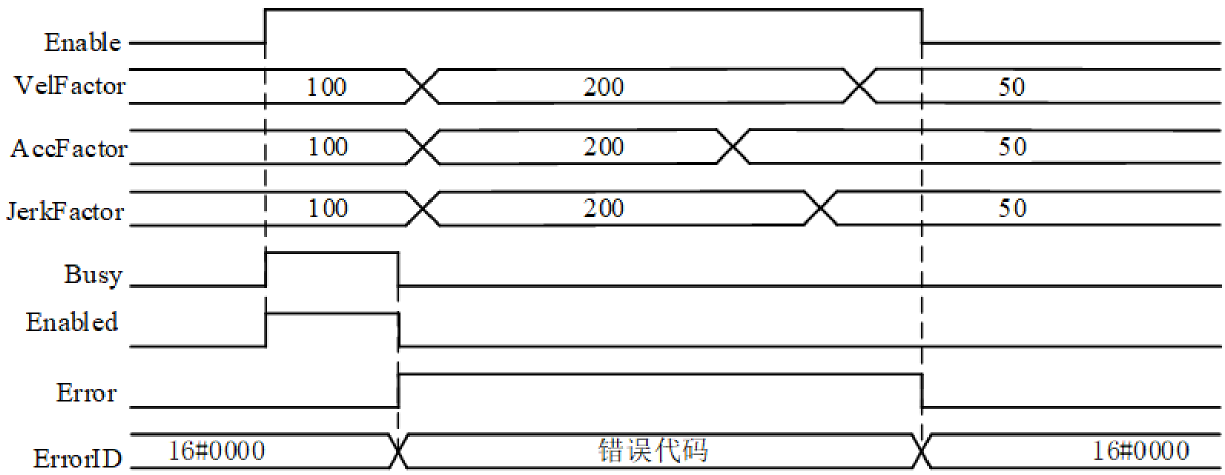
InVelocity( ) FALSE

InVelocity( ) TRUE MC_SetOverride Enable( ) FALSE

MC_MoveVelocity指令



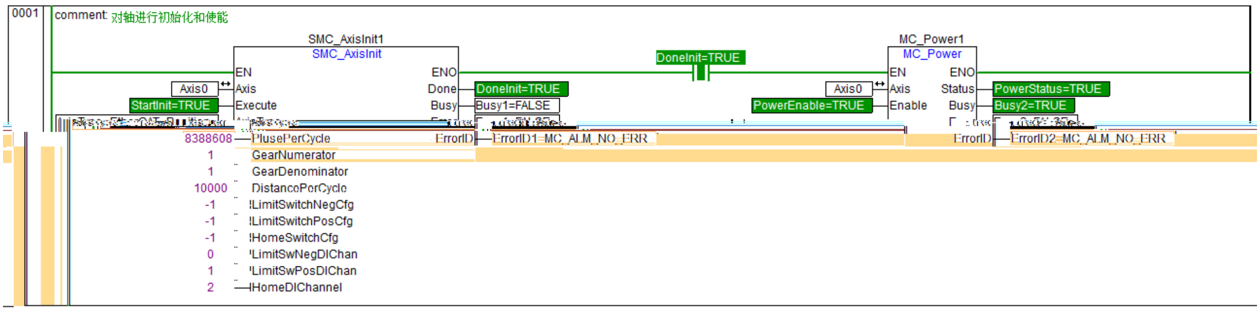
3


Enable
VelFactor
100
200
50
AccFactor
100
200
50
JerkFactor
100
200
50
Busy
Enabled
Error
ErrorID
16#0000
错误代码
16#0000

MC_MoveVelocity MC_MoveVelocity 20000 MC_SetOverride

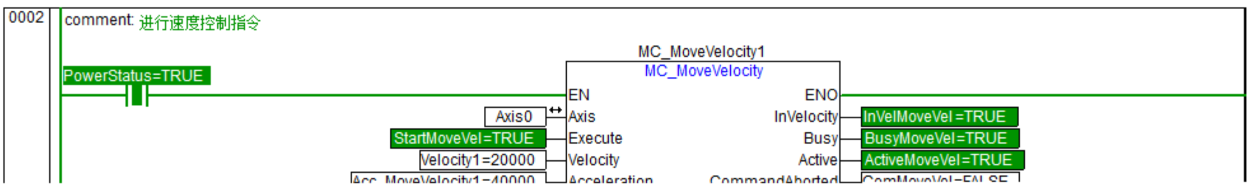
Enable( ) TRUE

1 ID 0 EtherCAT_Position



2 MC_MoveVelocity InVelocity TRUE MC_MoveVelocity

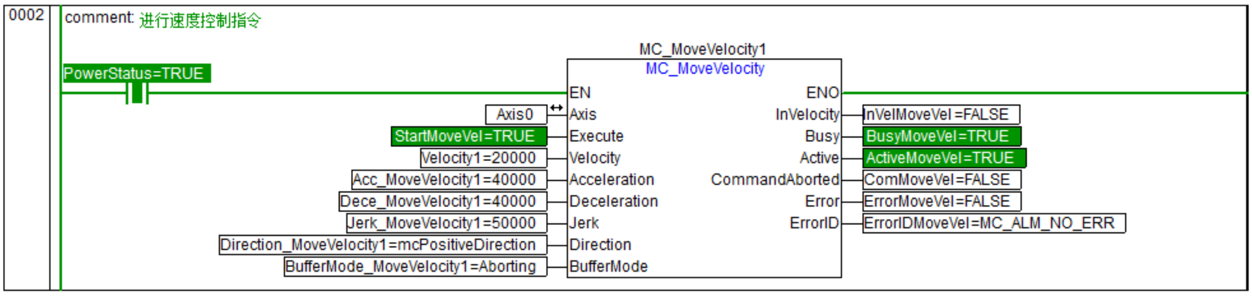
20000

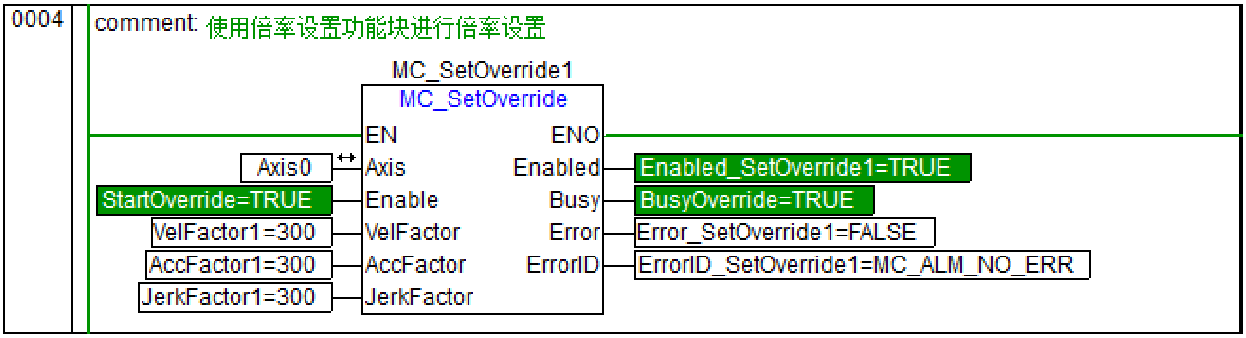


3 MC_SetOverride Enable

MC_MoveVelocity InVelocity FALSE

InVelocity TRUE





ST

1

```
SMC_AxisInit1(
Execute := StartInit,
AxisType := 50,
PlusePerCycle := 10000,
GearNumerator := 1,
GearDenominator := 1,
DistancePerCycle := 10000,
LimitSwitchNegCfg := -1,
LimitSwitchPosCfg := -1,
HomeSwitchCfg := -1,
LimitSwNegDIChan := 0,
LimitSwPosDIChan := 1,
HomeDIChannel := 2,
Axis := Axis0,
Done=> DoneInit,
Busy=>Busy1,
Error=>Error1,
ErrorID=>ErrorID1);
(*                    *)
IF DoneInit THEN
```

```
   MC_Power1(
   Enable := PowerEnable,
   Axis := Axis0,
   Status=> PowerStatus,
   Busy=>Busy2,
   Error=>Error2,
   ErrorID=>ErrorID2);
END_IF
```

2 MC_MoveVelocity

```
MC_MoveVelocity1(
Execute := StartMoveVel,
Velocity := 20000,
Acceleration := 40000,
Deceleration := 40000,
Jerk := 50000,
Direction := 0,
BufferMode := 0,
Axis := Axis0,
InVelocity=> InVelMoveVel,
Busy=> BusyMoveVel,
Active=> ActiveMoveVel,
CommandAborted=> ComMoveVel,
Error=> ErrorMoveVel,
ErrorID=> ErrorIDMoveVel);
```

3 MC_SetOverride

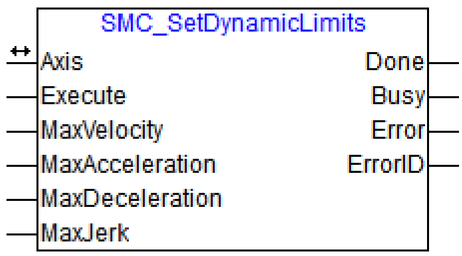
```
MC_SetOverride1(
Enable := StartOverride,
VelFactor := 300,
AccFactor := 300,
JerkFactor := 300,
Axis := Axis0,
Enabled=>EnabledOverride,
Busy=>BusyOverride,
Error=>ErrorOverride,
ErrorID=>ErrorID);
```

5.14.4

5.14.5

SMC_AxisInit MC_Power

5.15 SMC_SetDynamicLimits



5.15.1

IN_OUT	Axis			-	-	AXIS_REF
INPUT	Execute			-	TRUE/FALSE	BOOL
	MaxVelocity			1E100	/0	LREAL
	MaxAcceleration			1E100		LREAL
	MaxDeceleration			1E100		LREAL
	MaxJerk			1E100		LREAL
OUTPUT	Done			FALSE	TRUE/FALSE	BOOL
	Busy			FALSE	TRUE/FALSE	BOOL
	Error			FALSE	TRUE/FALSE	BOOL

	ErrorID			0	-	SMC_ERROR

5.15.2

1

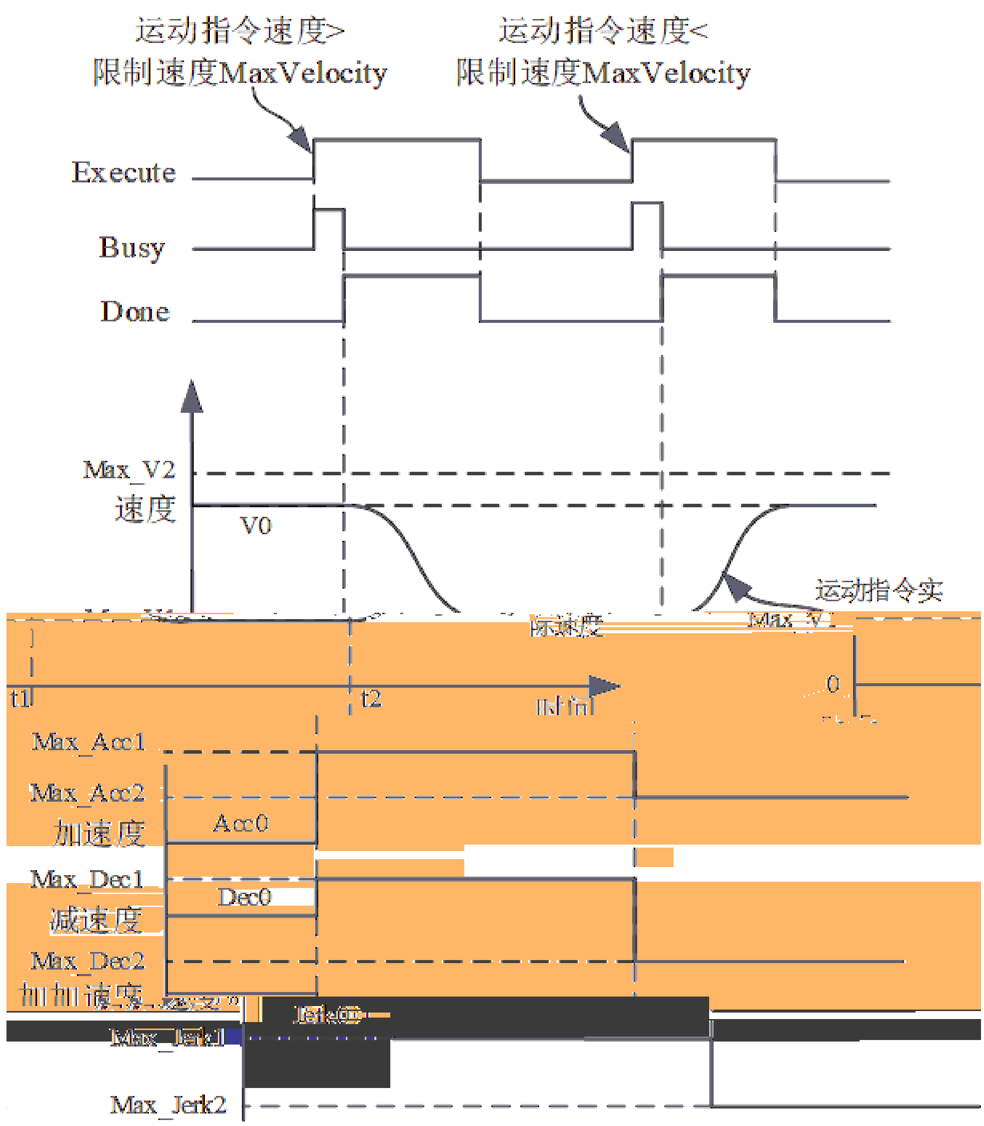
Execute Axis ID

MaxVelocity MaxAcceleration MaxDeceleration MaxJerk

2

MaxVelocity

MaxAcceleration MaxDeceleration MaxJerk



V0 Acc0 Dec0

Jerk0

t1 MaxVelocity Max_V1

MaxVelocity V0 MaxVelocity

Max_Acc1 Max_Dec1 Max_Jerk1

t2 MaxVelocity MaxVelocity= Max_V2 MaxVelocity

V0

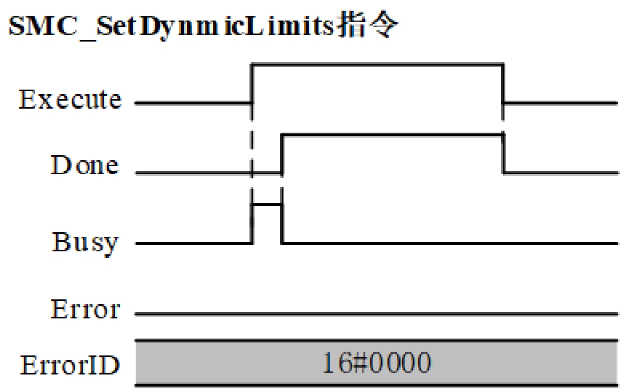
Max_Acc2 Max_Dec2 Max_Jerk2

1

Execute Busy TRUE Done

TRUE Busy FALSE Execute Done TRUE Execute

Done FALSE



2

Unused

Error=True ErrorID

Execute

- Execute SMC_SetDynamicLimits

5.15.3

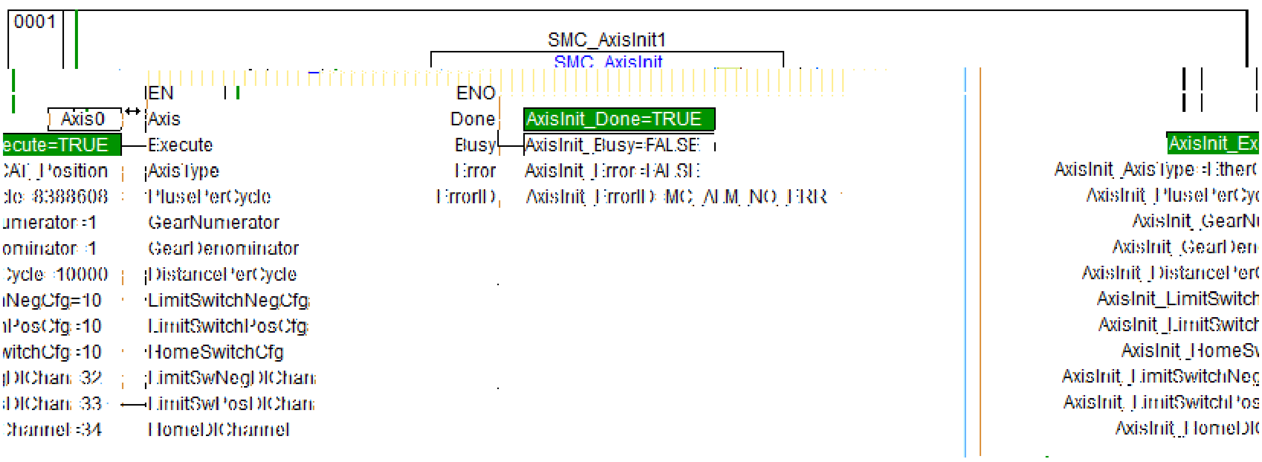
SMC_SetDynamicLimits

SMC_SetDynamicLimits MaxVelocity 20000 MaxAcceleration MaxDeceleration MaxJerk 50000 MC_Movelocity MC_Movelocity 50000 100000 SMC_SetDynamicLimits 20000 50000

Axis0	AXIS_REF	-	0
Power1_Status	BOOL	FALSE	TRUE
MV1_Vel	LREAL	-	Velocity
MV1_Active	BOOL	FALSE	TRUE
MV1_InVel	BOOL	FALSE	TRUE
Switch1	BOOL	FALSE	Switch TRUE

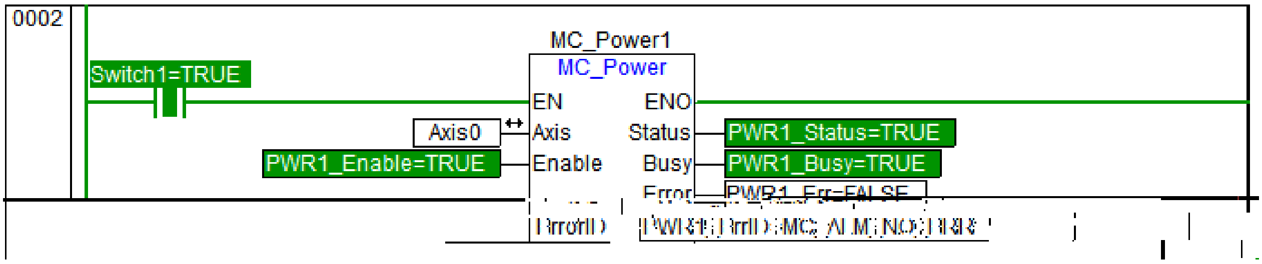
1

Axis0 AxisID 0 50 EtherCAT_Position SMC_AxisInit Axis0 SMC_AxisInit



2

Switch1 TRUE MC_Power

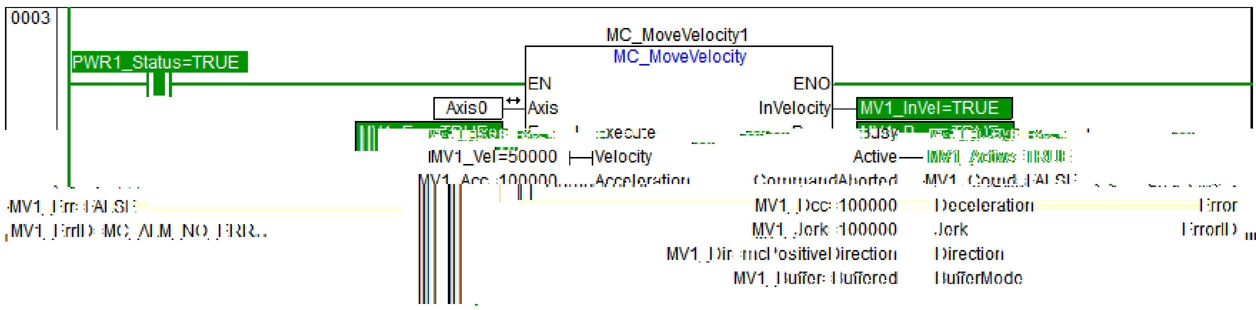


3

0003 MC_MoveVelocity

MC_MoveVelocity InVelocity TRUE MC_MoveVelocity

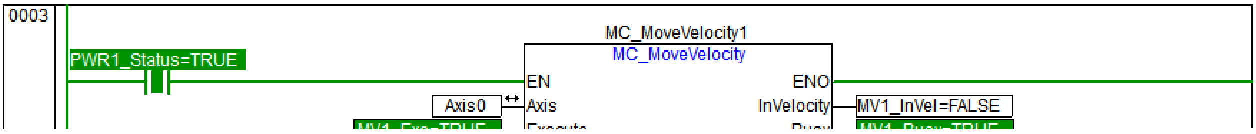
MV1_Vel= 50000



4

0004 SMC_SetDynamicLimits1 Execute 0003

MC_MoveVelocity InVelocity FALSE MC_MoveVelocity



ST

Axis0	AXIS_REF	-	0
Power1_Status	BOOL	FALSE	TRUE
MV1_Vel	LREAL	-	Velocity
MV1_Active	BOOL	FALSE	TRUE
MV1_InVel	BOOL	FALSE	TRUE
Switch1	BOOL	FALSE	Switch TRUE
InitFlag	BOOL	FALSE	TRUE

1

SMC_AxisInit Axis0 50 EtherCAT_Position

SMC_AxisInit

```
SMC_AxisInit1(
Execute := AxisInit_Execute,
AxisType := AxisInit_AxisType,
PlusePerCycle := AxisInit_PlusePerCycle,
GearNumerator := AxisInit_GearNumerator,
GearDenominator := AxisInit_GearDenominator,
DistancePerCycle := AxisInit_DistancePerCycle,
LimitSwitchNegCfg := AxisInit_LimitSwitchNegCfg,
```

```
LimitSwitchPosCfg := AxisInit_LimitSwitchPosCfg,
HomeSwitchCfg := AxisInit_HomeSwitchCfg,
LimitSwNegDIChan := AxisInit_LimitSwitchNegDIChan,
LimitSwPosDIChan := AxisInit_LimitSwitchPosDIChan,
HomeDIChannel := AxisInit_HomeDIChannel,
Axis := Axis0,
Done=>AxisInit_Done,
Busy=>AxisInit_Busy,
Error=>AxisInit_Error,
ErrorID=>AxisInit_ErrorID);
```

2

```
IF  FALSE = InitFlag THEN
( )]TJ-0.002 Tc 0.0 [illegible]
```

```
   ErrorID=> PWR1_ErrID);
END_IF
```

4

MC_MoveVelocity

```
IF  PWR1_Status THEN
   MC_MoveVelocity1(
   Execute := MV1_ Exe,
   Velocity := MV1_ Vel,
   Acceleration := MV1_ Acc,
   Deceleration := MV1_ Dcc,
   Jerk := MV1_ Jerk,
   Direction := MV1_ Dir,
   BufferMode := MV1_ Buffer,
   Axis := Axis0,
   InVelocity=> MV1_ InVel,
   Busy=> MV1_ Busy,
   Active=> MV1_ Active,
   CommandAborted=> MV1_ Comd,
   Error=> MV1_ Err,
   ErrorID=> MV1_ Error);
END_IF
```

5

SMC_SetDynamicLimits

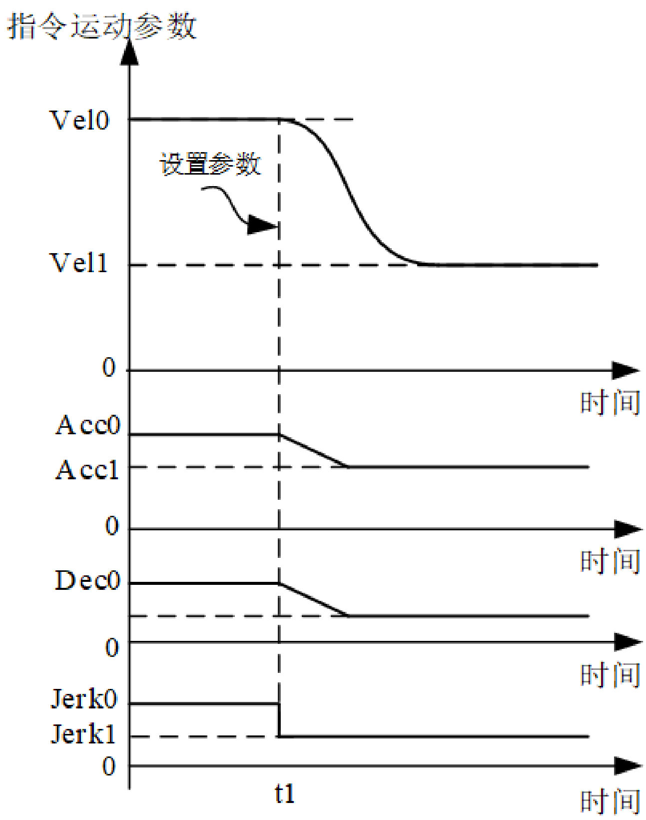
```
IF  Switch1 THEN
   SMC_SetDynamicLimits1(
   Execute := Dyn_Exe,
   MaxVelocity := Dyn_MaxVel,
   MaxAcceleration := Dyn_MaxAcc,
   MaxDeceleration := Dyn_ MaxDec,
   MaxJerk := Dyn_MaxJerk,
   Axis := Axis0,
   Done=> Dyn_Done,
   Busy=> Dyn_Busy,
   Error=> Dyn_Err,
   ErrorID=> Dyn_ErrID);
```

```
END_IF
```

MC_MoveVelocity 0~t1

MC_MoveVelocity t1

SMC_SetDynamicLimits MC_MoveVelocity

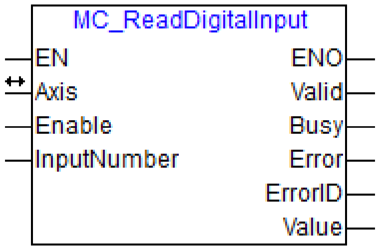


- Vel0 Acc0 Dec0 Jerk0
- Vel1 Acc1 Dec1 Jerk1

5.15.4

MaxVelocity 0

5.16 MC_ReadDigitalInput



5.16.1

IN_OUT	Axis			-	-	AXIS_REF
INPUT	Enable			-	TRUE/FALSE	BOOL
	InputNumber	Input Digtal inputs bit		-	0 ~ 31	INT
OUTPUT	Valid			-	TRUE/FALSE	BOOL
	Busy			FALSE	TRUE/FALSE	BOOL
	Error			FALSE	TRUE/FALSE	BOOL
	ErrorID			0	-	SMC_ERROR
	Value			-	TRUE/FALSE	BOOL

5.16.2

CIA402

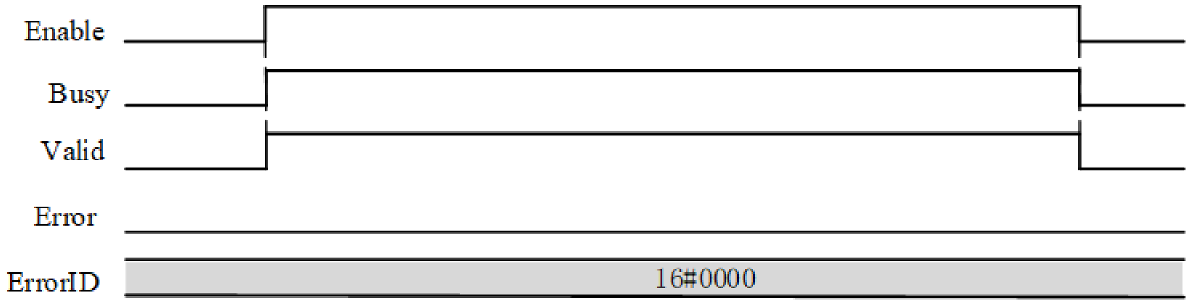
1 Input

2 Enable( ) TRUE Enable( ) FALSE

Enable( ) TRUE Busy( ) Valid( ) TRUE

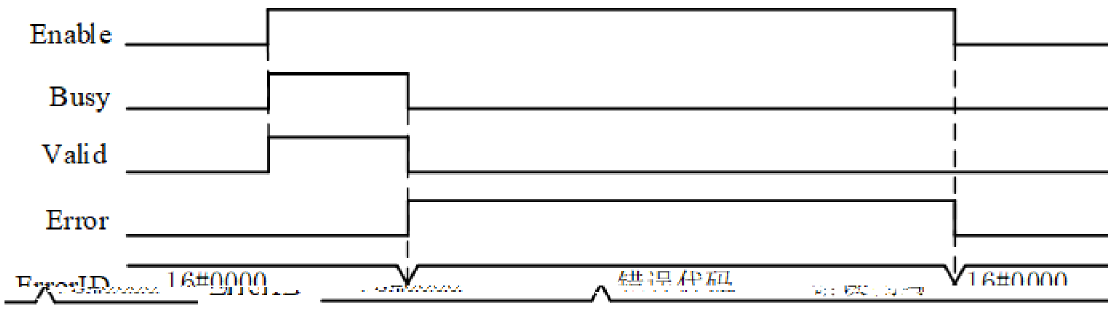
Value( ) TRUE Enable( ) FALSE

1



2

Error( ) TRUE FALSE



5.16.3

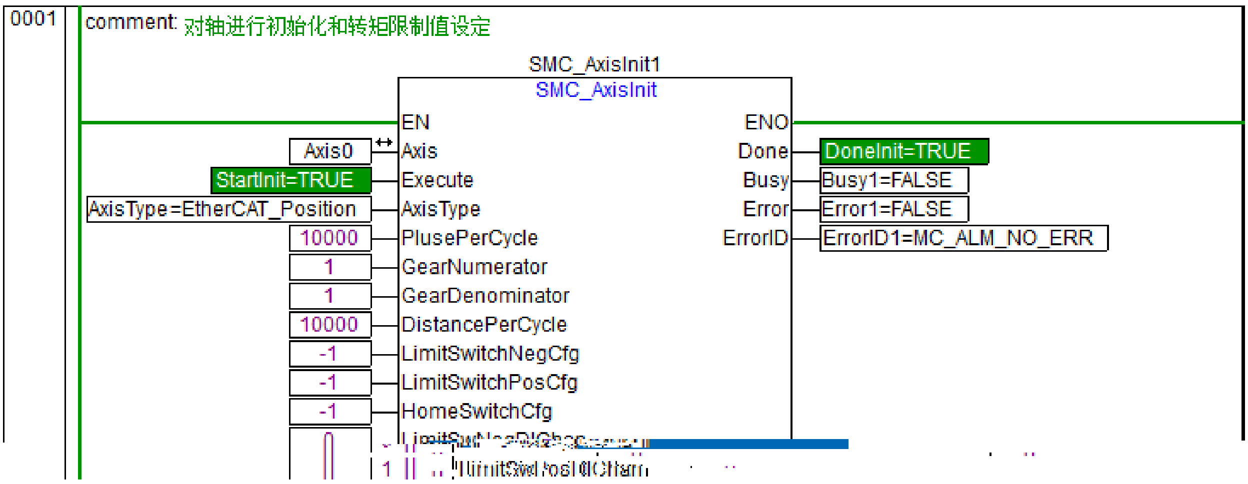
MC_ReadDigitalInput

Axis0	AXIS_REF		0
StartInit	BOOL	FALSE	TRUE
DoneInit	BOOL	FALSE	TRUE
ReadInputEnable	BOOL	FALSE	TRUE
ValidReadInput	BOOL	FALSE	TRUE
BusyReadInput	BOOL	FALSE	TRUE
ValueReadInput	BOOL	FALSE	TRUE
ErrorReadInput	BOOL	FALSE	TRUE

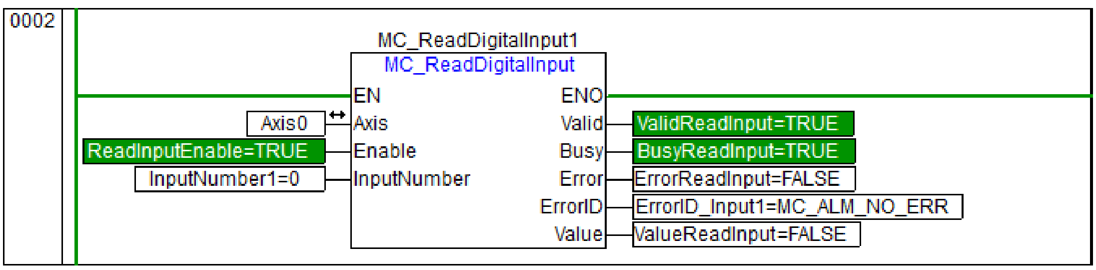
MC_ReadDigitalInput Enable( )

TRUE

1



2



ST

1

```
SMC_AxisInit1(
Execute := StartInit,
AxisType := 50,
PlusePerCycle := 10000,
GearNumerator := 1,
GearDenominator := 1,
DistancePerCycle := 10000,
LimitSwitchNegCfg := -1,
LimitSwitchPosCfg := -1,
```

```
HomeSwitchCfg := -1,
LimitSwNegDIChan := 0,
LimitSwPosDIChan := 1,
HomeDIChannel := 2,
Axis := Axis0,
Done=> DoneInit,
Busy=>Busy1,
Error=>Error1,
ErrorID=>ErrorID1);
```

2

```
MC_ReadDigitalInput1(
Enable := ReadInputEnable,
InputNumber := 0,
Axis := Axis0,
Valid=>ValidReadInput,
Busy=>BusyReadInput,
Error=>ErrorReadInput,
ErrorID=>ErrorID_Input1,
Value=>ValueReadInput);
```

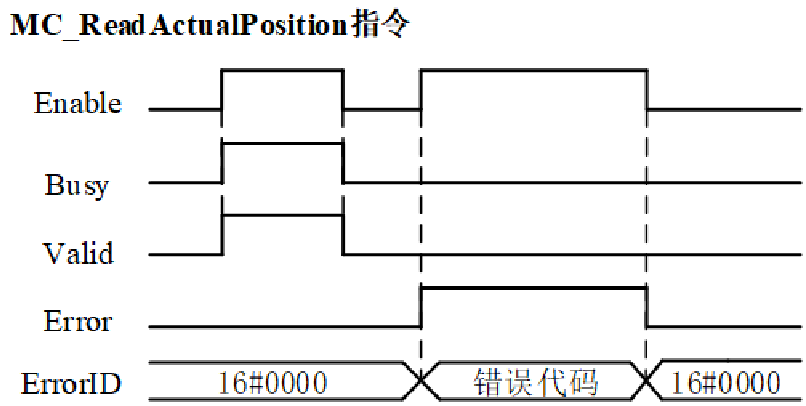
5.16.4

EtherCAT

Digtial Inputs PDO 0x60FD

5.16.5

5.17 MC_ReadActualPosition



- Busy Valid TRUE Position
- Busy Valid FLASE Position

Error True ErrorID

Enable

5.17.3

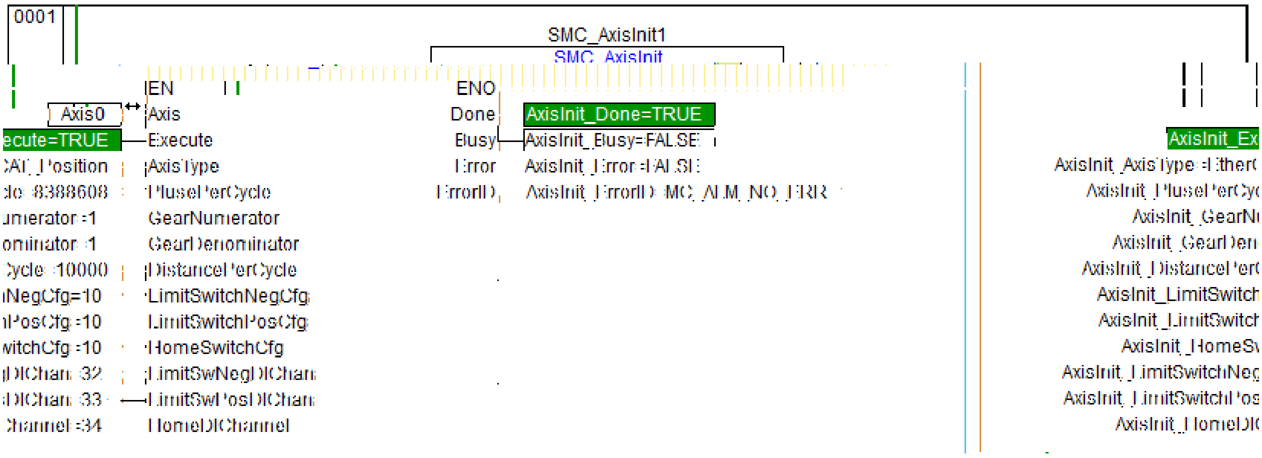
MC_Movelocity MC_ReadActualPosition

Axis0	AXIS_REF	-	0
Power1_Status	BOOL	FALSE	TRUE
ReadActPos_Valid	BOOL	FALSE	TRUE
ReadActPos_Pos	LREAL	-	
Switch1	BOOL	FALSE	Switch TRUE

1

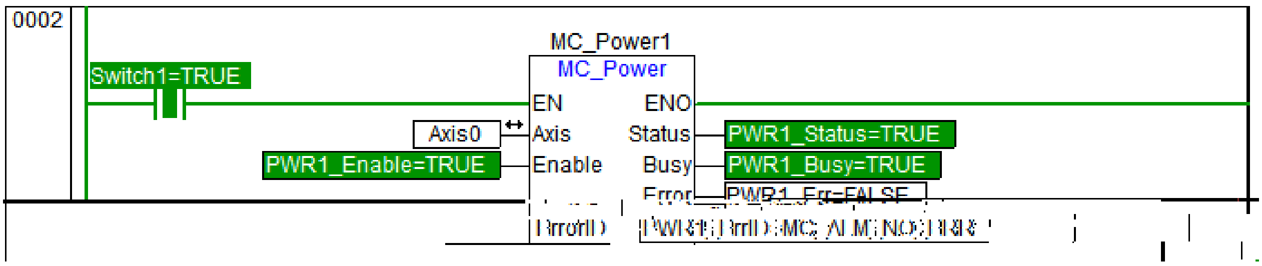
Axis0 AxisID 0 50 EtherCAT_Position SMC_AxisInit

Axis0 SMC_AxisInit

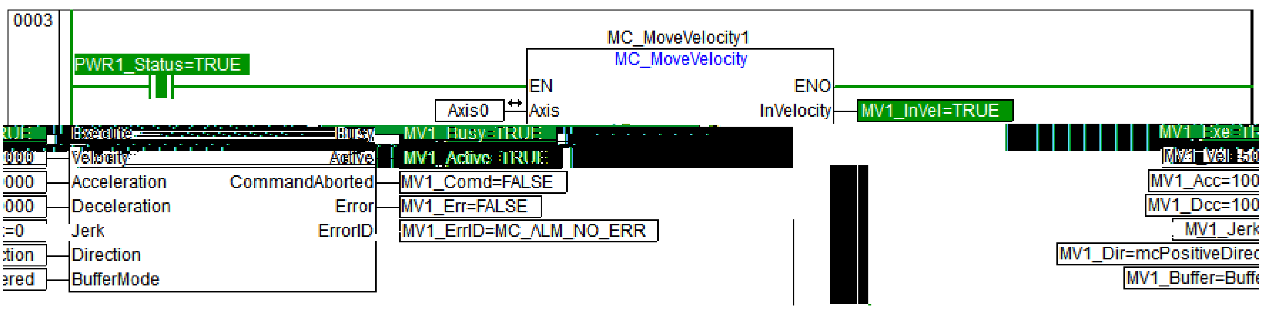


2

Switch1 TRUE MC_Power



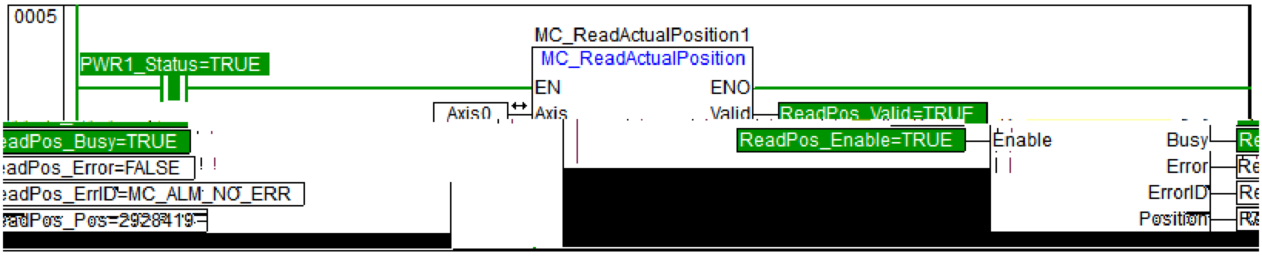
3 0003 MC_MoveVelocity



4

0005 MC_ReadActualPosition ReadPos_Valid

TRUE ReadPos_Pos



ST

1 ST LD

2 Axis0

```
Axis0.AxisID:=0;
```

3

SMC_AxisInit Axis0 50 EtherCAT_Position

SMC_AxisInit

```
SMC_AxisInit1(
Execute := AxisInit_Execute,
AxisType := AxisInit_AxisType,
PlusePerCycle := AxisInit_PlusePerCycle,
GearNumerator := AxisInit_GearNumerator,
GearDenominator := AxisInit_GearDenominator,
DistancePerCycle := AxisInit_DistancePerCycle,
LimitSwitchNegCfg := AxisInit_LimitSwitchNegCfg,
LimitSwitchPosCfg := AxisInit_LimitSwitchPosCfg,
HomeSwitchCfg := AxisInit_HomeSwitchCfg,
LimitSwNegDIChan := AxisInit_LimitSwitchNegDIChan,
LimitSwPosDIChan := AxisInit_LimitSwitchPosDIChan,
HomeDIChannel := AxisInit_HomeDIChannel,
Axis := Axis0,
Done=>AxisInit_Done,
Busy=>AxisInit_Busy,
Error=>AxisInit_Error,
ErrorID=>AxisInit_ErrorID);
```

4

Switch1 TRUE MC_Power

```
IF Switch1 THEN
```

```
    MC_Power1(
    Enable := PWR1_Enable,
    Axis := Axis0,
    Status=> PWR1_Status,
    Busy=> PWR1_Busy,
    Error=> PWR1_Err
    ErrorID=> PWR1_ErrID);
END_IF
```

5 MC_MoveVelocity

```
IF  PWR1_Status THEN
    MC_MoveVelocity1(
    Execute := MV1_ Exe,
    Velocity := MV1_ Vel,
    Acceleration := MV1_ Acc,
    Deceleration := MV1_ Dcc,
    Jerk := MV1_ Jerk,
    Direction := MV1_ Dir,
    BufferMode := MV1_ Buffer,
    Axis := Axis0,
    InVelocity=> MV1_ InVel,
    Busy=> MV1_ Busy,
    Active=> MV1_ Active,
    CommandAborted=> MV1_ Comd,
    Error=> MV1_ Err,
    ErrorID=> MV1_ Error);
END_IF
```

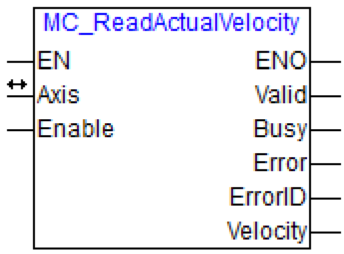
6 MC_ReadActualPosition

```
IF  PWR1_Status THEN
    MC_ReadActualPosition1(
    Enable := ReadPos_Enable,
    Axis := Axis0,
    Valid=> ReadPos_Valid,
    Busy=> ReadPos_Busy,
    Error=> ReadPos_Err,
    ErrorID=> ReadPos_ErrID,
    Position=> ReadPos_Pos);
END_IF
```

5.17.4

Position

5.18 MC_ReadActualVelocity



5.18.1

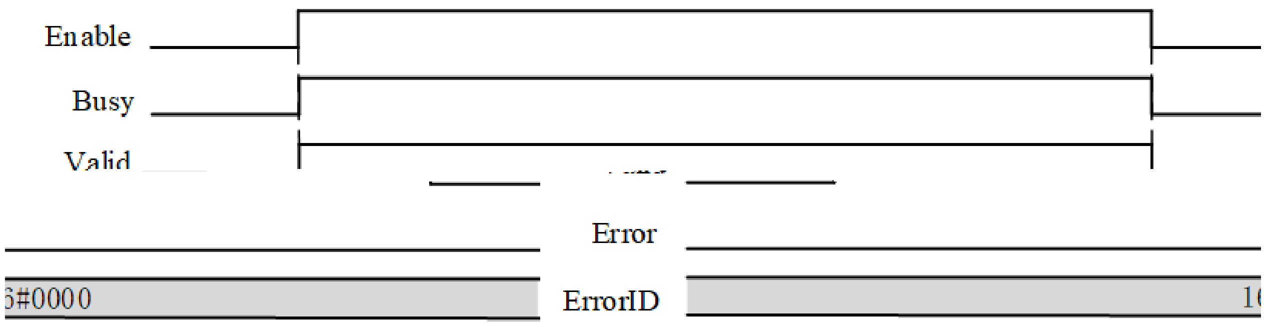
IN_OUT	Axis			-	-	AXIS_REF
INPUT	Enable			-	TRUE/FALSE	BOOL
OUTPUT	Valid			-	TRUE/FALSE	BOOL
	Busy			FALSE	TRUE/FALSE	BOOL
	Error			FALSE	TRUE/FALSE	BOOL
	ErrorID			0	-	SMC_ERROR
	Velocity			-	/ /0	LREAL

5.18.2

1

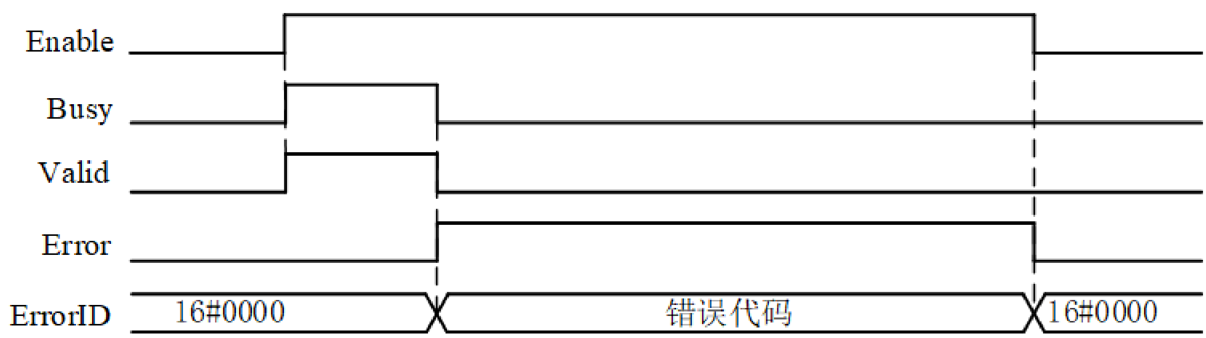
2 Enable( ) TRUE Enable( ) FALSE

1



2

Error( ) TRUE FALSE



5.18.3

MC_ReadActualVelocity

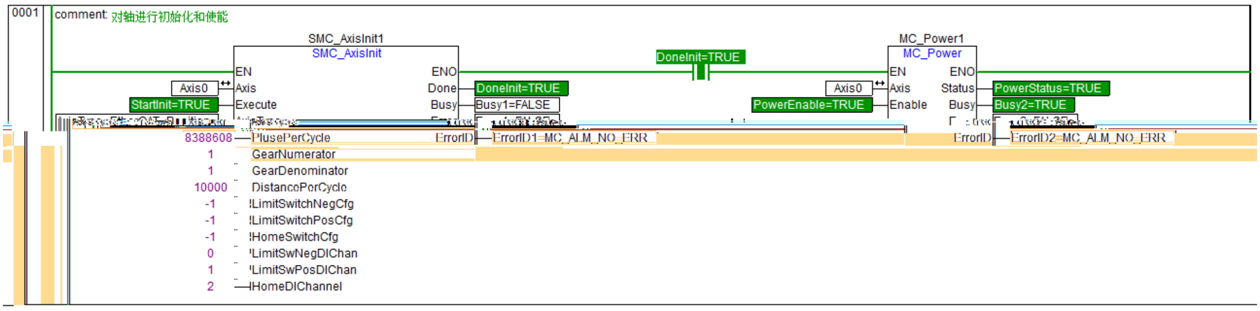
Axis0	AXIS_REF		0
StartInit	BOOL	FALSE	TRUE
DoneInit	BOOL	FALSE	TRUE
ReadVelEnable	BOOL	FALSE	TRUE
ValidReadVel	BOOL	FALSE	TRUE
BusyReadVel	BOOL	FALSE	TRUE
ValueReadVel	LREAL		
ErrorReadVel	BOOL	FALSE	TRUE

MC_ReadActualVelocity Enable( )

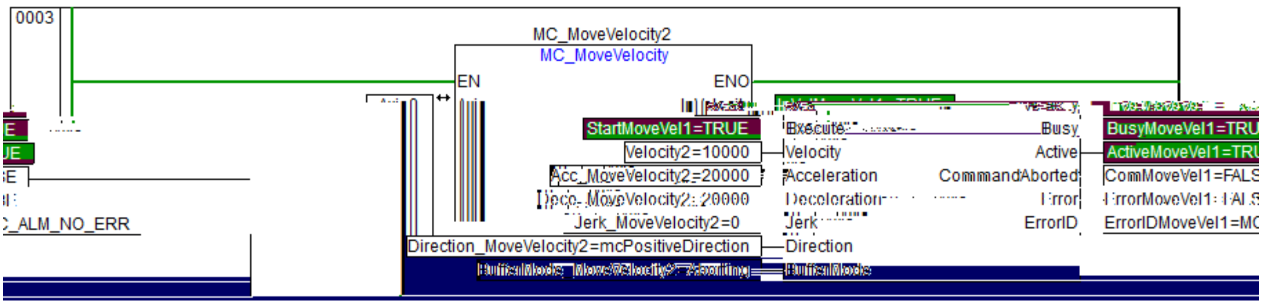
TRUE

1

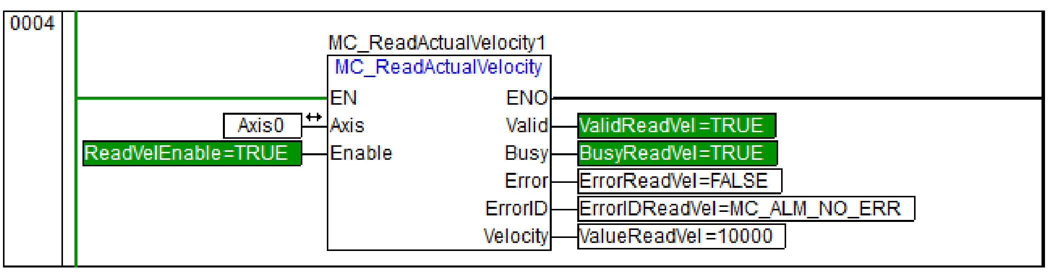
ON



2 MC_MoveVelocity Velocity 10000



3 MC_ReadActualVelocity



ST

1

```
SMC_AxisInit1(
Execute := StartInit,
AxisType := 50,
```

```
PlusePerCycle := 10000,
GearNumerator := 1,
GearDenominator := 1,
DistancePerCycle := 10000,
LimitSwitchNegCfg := -1,
LimitSwitchPosCfg := -1,
HomeSwitchCfg := -1,
LimitSwNegDIChan := 0,
LimitSwPosDIChan := 1,
HomeDIChannel := 2,
Axis := Axis0,
Done=> DoneInit,
Busy=>Busy1,
Error=>Error1,
ErrorID=>ErrorID1);
(*                    *)
IF DoneInit THEN
   MC_Power1(
   Enable := PowerEnable,
   Axis := Axis0,
   Status=> PowerStatus,
   Busy=>Busy2,
   Error=>Error2,
   ErrorID=>ErrorID2);
END_IF
```

2

```
(*                          *)
MC_MoveVelocity1(
Execute := StartMoveVel,
Velocity := 10000,
Acceleration := 20000,
Deceleration := 20000,
Jerk := 0,
Direction := 0,
BufferMode := 0,
Axis := Axis0,
```

```
InVelocity=>InVelMoveVel,
Busy=>BusyMoveVel,
Active=>ActiveMoveVel,
CommandAborted=>ComMoveVel,
Error=>ErrorMoveVel,
ErrorID=>ErrorIDMoveVel);
```

3

- MC_ReadActualVelocity1(
- Enable := ReadVelEnable,
- Axis := Axis0,
- Valid=>ValidReadVel,
- Busy=>BusyReadVel,
- Error=>ErrorReadVel,
- ErrorID=>ErrorIDReadVel,
- Velocity=>ValueReadVel);

5.18.4

Ethercat

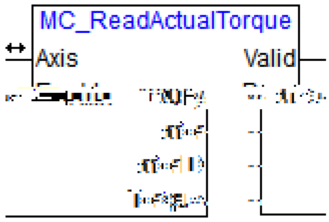
5.18.5

MC_MoveVelocity MC_MoveVelocity

ErrorID

5.19 MC_ReadActualTorque

CIA402



5.19.1

IN_OUT	Axis			-	-	AXIS_REF
INPUT	Enable			-	TRUE/FALSE	BOOL
OUTPUT	Valid			-	TRUE/FALSE	BOOL
	Busy			FALSE	TRUE/FALSE	BOOL
	Error			FALSE	TRUE/FALSE	BOOL
	ErrorID			0	-	SMC_ERROR
	Torque	1%		-	/ /0	LREAL

5.19.2

CIA402

1 Enable TRUE

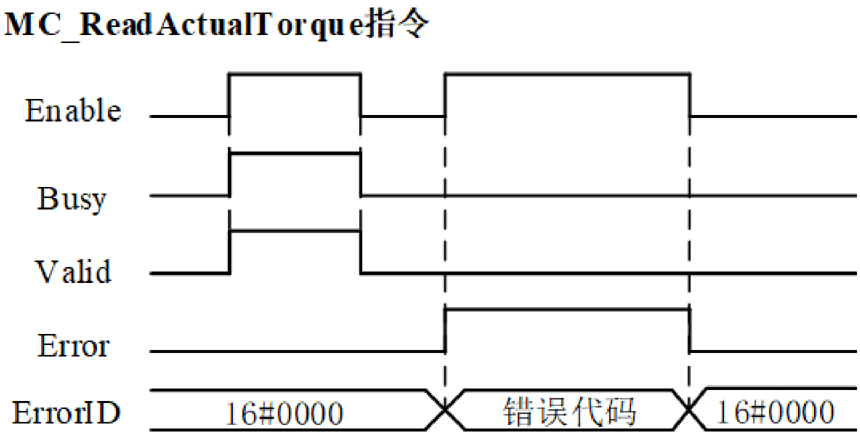
Valid 1%

2

EtherCAT_Position 50 EtherCAT

EtherCAT_Speed 51 EtherCAT

EtherCAT_Torque 52 EtherCAT



- Busy Valid TRUE Torque
- Busy Valid FLASE Torque 0

Error True ErrorID

Enable

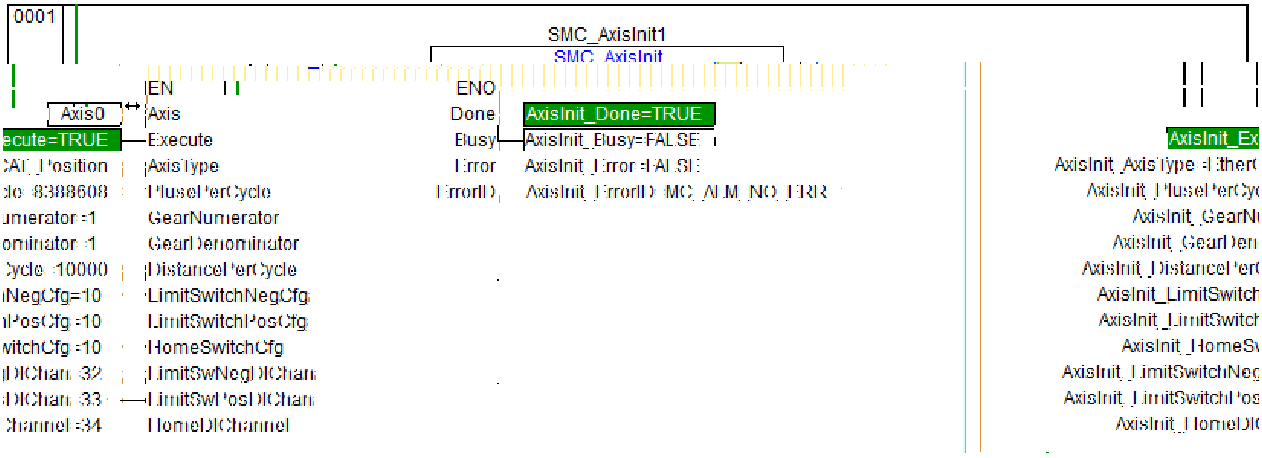
5.19.3

Axis0	AXIS_REF	-	0
Power1_Status	BOOL	FALSE	TRUE
ReadTor_Valid	BOOL	FALSE	TRUE
ReadTor_Tor	LREAL	-	1%
Switch1	BOOL	FALSE	Switch TRUE

1

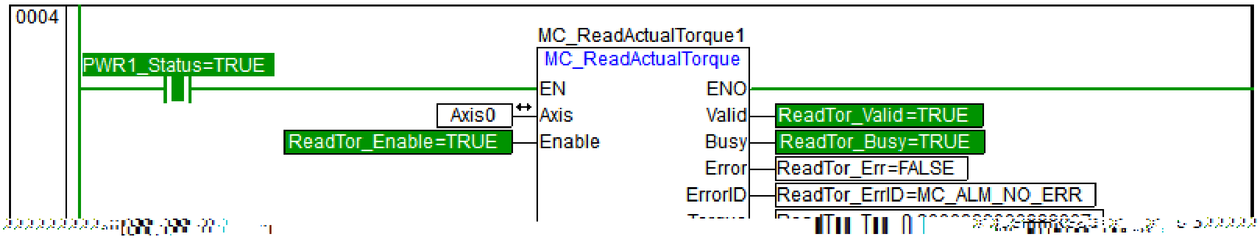
Axis0 AxisID 0 50 EtherCAT_Position SMC_AxisInit

Axis0 SMC_AxisInit


0001
SMC_AxisInit1
SMC_AxisInit
EN
ENO
Axis0
Axis
Done
AxisInit_Done=TRUE
Execute
Busy
AxisInit_Busy=FALSE
AxisType
Error
PluselPerCycle
ErrorID
GearNumerator
GearDenominator
DistancePerCycle
LimitSwitchNegCfg
LimitSwitchPosCfg
HomeSwitchCfg
LimitSwNegDIChan
LimitSwPosDIChan
HomeDIChannel

2

Switch1



ST

ST LD

1 Axis0

```
Axis0.AxisID:=0;
```

2

SMC_AxisInit Axis0 50 EtherCAT_Position

SMC_AxisInit

```
SMC_AxisInit1(
Execute := AxisInit_Execute,
AxisType := AxisInit_AxisType,
PlusePerCycle := AxisInit_PlusePerCycle,
GearNumerator := AxisInit_GearNumerator,
GearDenominator := AxisInit_GearDenominator,
DistancePerCycle := AxisInit_DistancePerCycle,
LimitSwitchNegCfg := AxisInit_LimitSwitchNegCfg,
LimitSwitchPosCfg := AxisInit_LimitSwitchPosCfg,
HomeSwitchCfg := AxisInit_HomeSwitchCfg,
LimitSwNegDIChan := AxisInit_LimitSwitchNegDIChan,
LimitSwPosDIChan := AxisInit_LimitSwitchPosDIChan,
HomeDIChannel := AxisInit_HomeDIChannel,
Axis := Axis0,
Done=>AxisInit_Done,
```

```
IF  Switch1 THEN
   MC_Power1(
   Enable := PWR1_Enable,
   Axis := Axis0,
   Status=> PWR1_Status,
   Busy=> PWR1_Busy,
   Error=> PWR1_Err
   ErrorID=> PWR1_ErrID);
END_IF
```

4 MC_MoveVelocity

```
IF  PWR1_Status THEN
   MC_MoveVelocity1(
   Execute := MV1_ Exe,
   Velocity := MV1_ Vel,
   Acceleration := MV1_ Acc,
   Deceleration := MV1_ Dcc,
   Jerk := MV1_ Jerk,
   Direction := MV1_ Dir,
   BufferMode := MV1_ Buffer,
   Axis := Axis0,
   InVelocity=> MV1_ InVel,
   Busy=> MV1_ Busy,
   Active=> MV1_ Active,
   CommandAborted=> MV1_ Comd,
   Error=> MV1_ Err,
   ErrorID=> MV1_ Error);
END_IF
```

5 MC_ReadActualTorque

```
IF  MC_Power1. Status THEN
   MC_ReadActualTorque1(
   Enable := ReadTor_Enable,
   Axis := Axis0,
   Valid=> ReadTor_Valid,
   Busy=> ReadTor_Busy,
   Error=> ReadTor_Err,
   ErrorID=> ReadTor_ErrID,
   Torque=> ReadTor);
```

```
END_IF
```

5.19.4

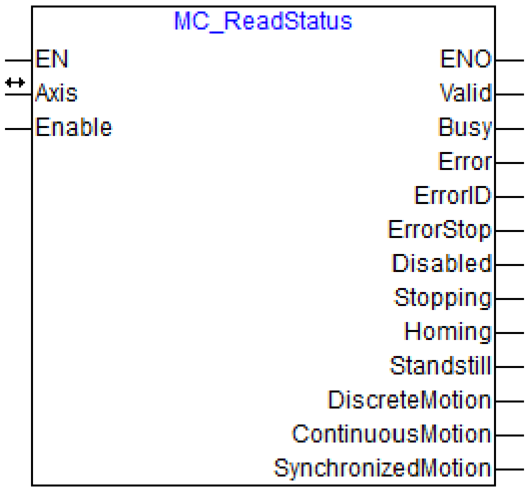
CIA402

16#6077 PDO 0x60FD

PDO

5.20 MC_ReadStatus

PLCopen



5.20.1

IN_OUT	Axis			-	-	AXIS_REF
INPUT	Enable			-	TRUE/FALSE	BOOL
	Valid			-	TRUE/FALSE	BOOL
	Busy			FALSE	TRUE/FALSE	BOOL
OUTPUT	Error			FALSE	TRUE/FALSE	BOOL
	ErrorID			0	-	SMC_ERROR
	ErrorStop	TRUE ErrorStop		FALSE	TRUE/FALSE	BOOL

	Disabled	TRUE	Disabled		FALSE	TRUE/FALSE	BOOL
	Stopping	TRUE	Stopping		FALSE	TRUE/FALSE	BOOL
	Homing	TRUE	Homing		FALSE	TRUE/FALSE	BOOL
	Standstill	TRUE	Standstill		FALSE	TRUE/FALSE	BOOL
	DiscreteMotion	TRUE	DiscreteMotion		FALSE	TRUE/FALSE	BOOL
	ContinuousMotion	TRUE	ContinuousMotion		FALSE	TRUE/FALSE	BOOL
	SynchronizedMotion	TRUE	SynchronizedMotion		FALSE	TRUE/FALSE	BOOL

5.20.2

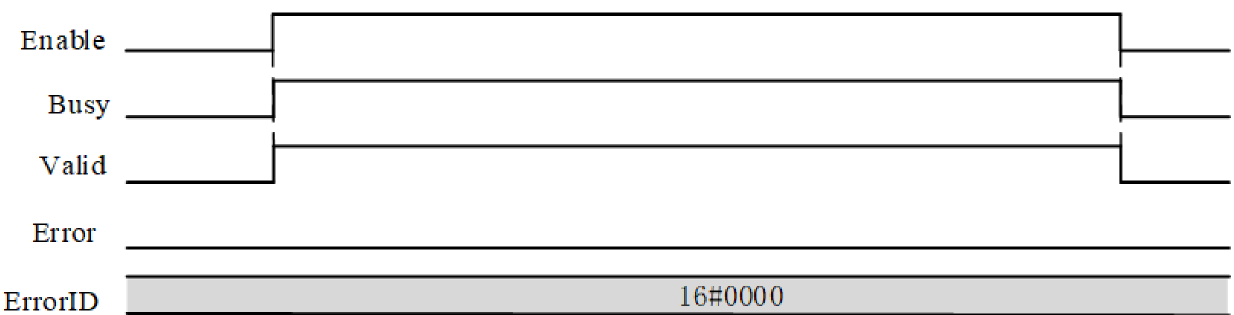
PLCopen

1 PLCopen

2 Enable( ) TRUE PLCopen Enable( )

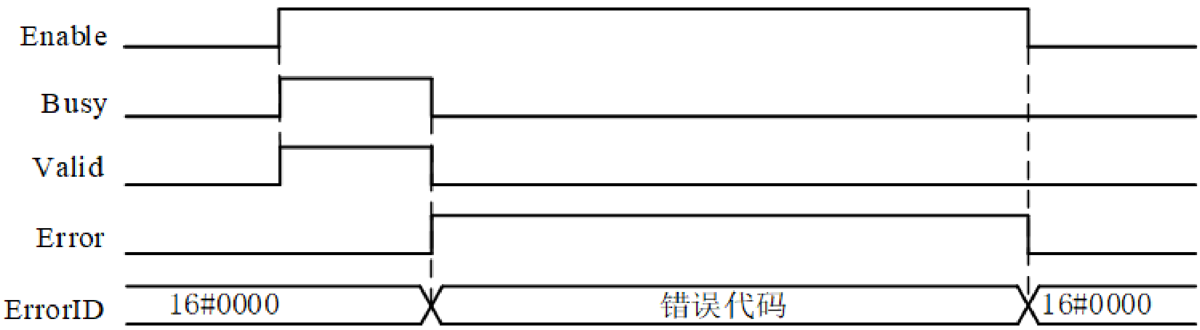
FALSE

1



2

Error( ) TRUE FALSE



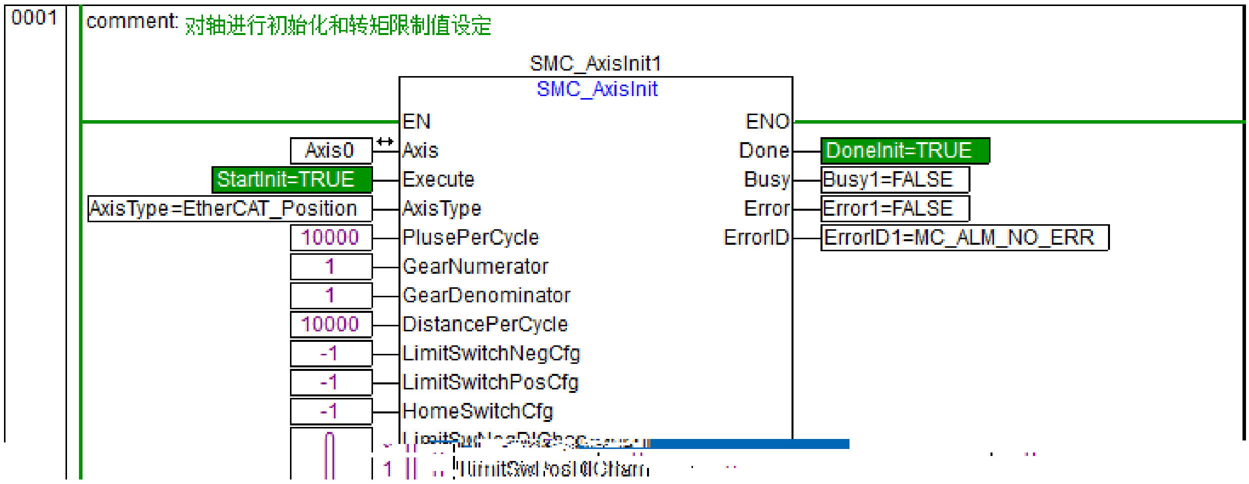
5.20.3

MC_ReadStatus PLCopen

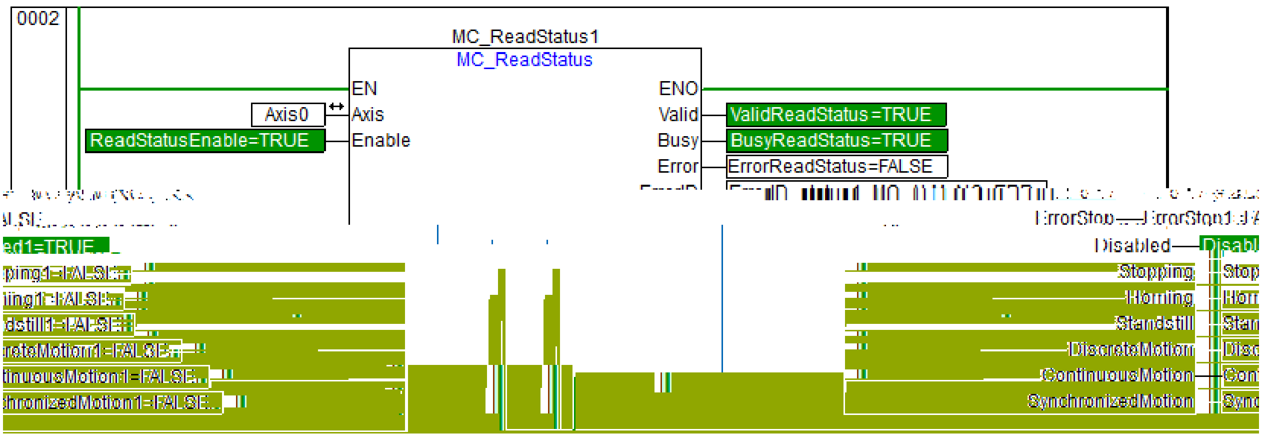
Axis0	AXIS_REF		0
StartInit	BOOL	FALSE	TRUE
DoneInit	BOOL	FALSE	TRUE
ReadStatusEnable	BOOL	FALSE	TRUE
ValidReadStatus	BOOL	FALSE	TRUE
BusyReadStatus	BOOL	FALSE	TRUE
ErrorReadStatus	BOOL	FALSE	TRUE

MC_ ReadStatus Enable( ) TRUE

1 PLCopen



2 PLCopen



ST

1

```
SMC_AxisInit1(
Execute := StartInit,
AxisType := 50,
PlusePerCycle := 10000,
GearNumerator := 1,
GearDenominator := 1,
DistancePerCycle := 10000,
LimitSwitchNegCfg := -1,
LimitSwitchPosCfg := -1,
HomeSwitchCfg := -1,
LimitSwNegDIChan := 0,
LimitSwPosDIChan := 1,
HomeDIChannel := 2,
Axis := Axis0,
```

```
Done=> DoneInit,
Busy=>Busy1,
Error=>Error1,
ErrorID=>ErrorID1);
```

2 PLCopen

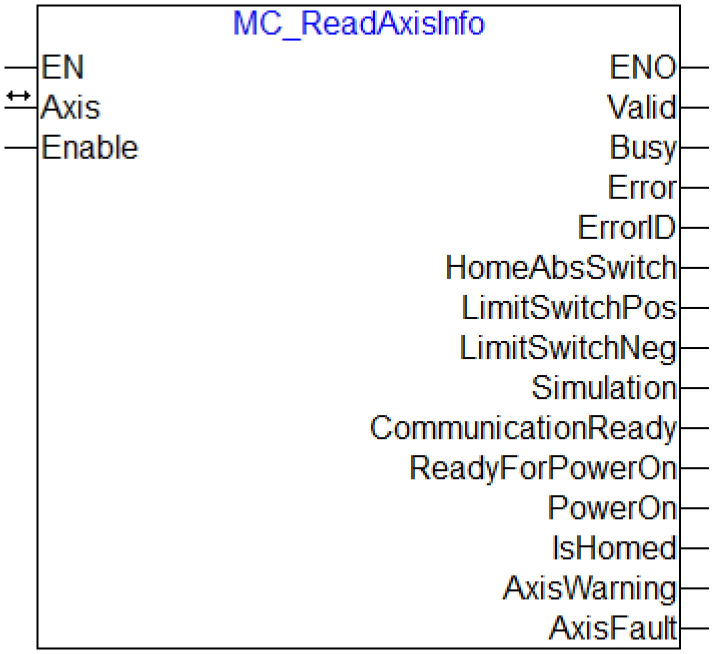
```
MC_ReadStatus1(
Enable := ReadStatusEnable,
Axis := Axis0,
Valid=>ValidReadStatus,
Busy=>BusyReadStatus,
Error=>ErrorReadStatus,
ErrorID=>ErrorID_status1,
ErrorStop=>ErrorStop1,
Disabled=>Disabled1,
Stopping=>Stopping1,
Homing=>Homing1,
Standstill=>Standstill1,
DiscreteMotion=>DiscreteMotion1,
ContinuousMotion=>ContinuousMotion1,
SynchronizedMotion=>SynchronizedMotion1);
```

5.20.4

Enable Enable

5.20.5

5.21 MC_ReadAxisInfo



5.21.1

IN_OUT	Axis			-	-	AXIS_REF
INPUT	Enable			-	TRUE/FALSE	BOOL
OUTPUT	Valid			-	TRUE/FALSE	BOOL
	Busy			FALSE	TRUE/FALSE	BOOL
	Error			FALSE	TRUE/FALSE	BOOL
	ErrorID			0	-	SMC_ERROR
	HomeAbsSwitch			FALSE	TRUE/FALSE	BOOL
	LimitSwitchPos			FALSE	TRUE/FALSE	BOOL
	LimitSwitchNeg			FALSE	TRUE/FALSE	BOOL
		FALSE		FALSE	TRUE/FALSE	BOOL
	CommunicationReady			FALSE	TRUE/FALSE	BOOL
	ReadyForPowerOn			FALSE	TRUE/FALSE	BOOL
	PowerOn			FALSE	TRUE/FALSE	BOOL
	IsHomed			FALSE	TRUE/FALSE	BOOL
	AxisWarning			FALSE	TRUE/FALSE	BOOL
	AxisFault			FALSE	TRUE/FALSE	BOOL

5.21.2

Enable TRUE Valid TRUE

Error TRUE ErrorID Enable

FALSE

HomeAbsSwitch TRUE FALSE

LimitSwitchPos LimitSwitchNeg TRUE

FALSE

Simulation TRUE FALSE

Simulation FALSE

CommunicationReady EtherCAT

CommunicationReady TRUE FALSE

ReadyForPowerOn

ReadyForPowerOn TRUE FALSE

PowerOn PowerOn

TRUE FALSE

IsHomed TRUE

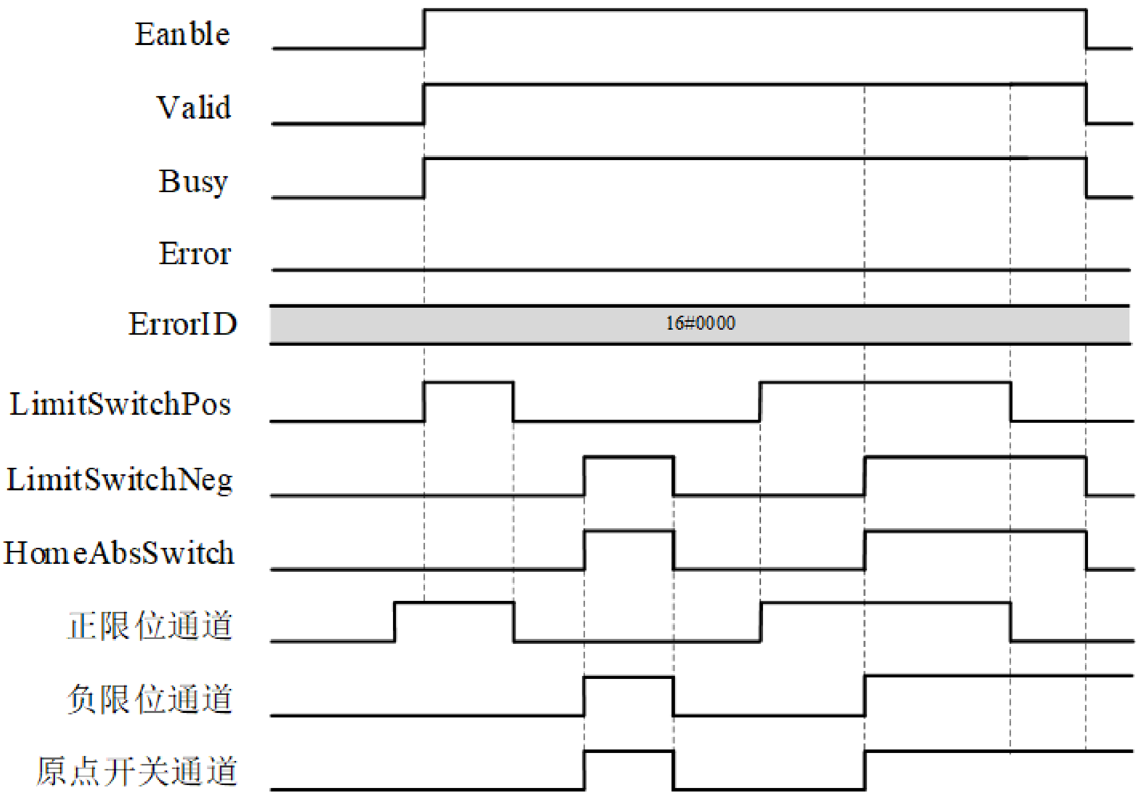
FALSE

AxisWarning EtherCAT

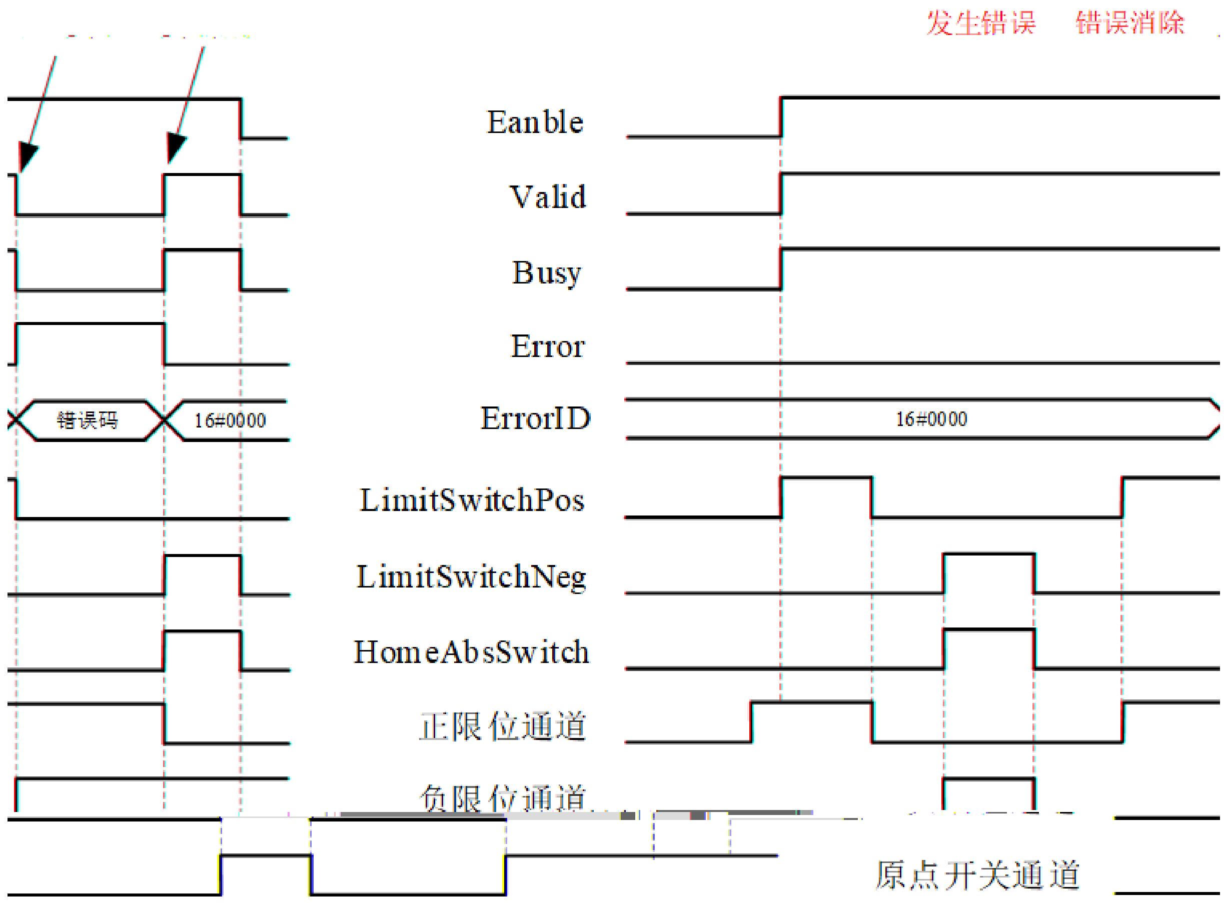
AxisWarning TRUE FALSE

AxisFault TRUE FALSE

1


Eanble
Valid
Busy
Error
ErrorID
16#0000
LimitSwitchPos
LimitSwitchNeg
HomeAbsSwitch
正限位通道
负限位通道
原点开关通道

2



5.21.3

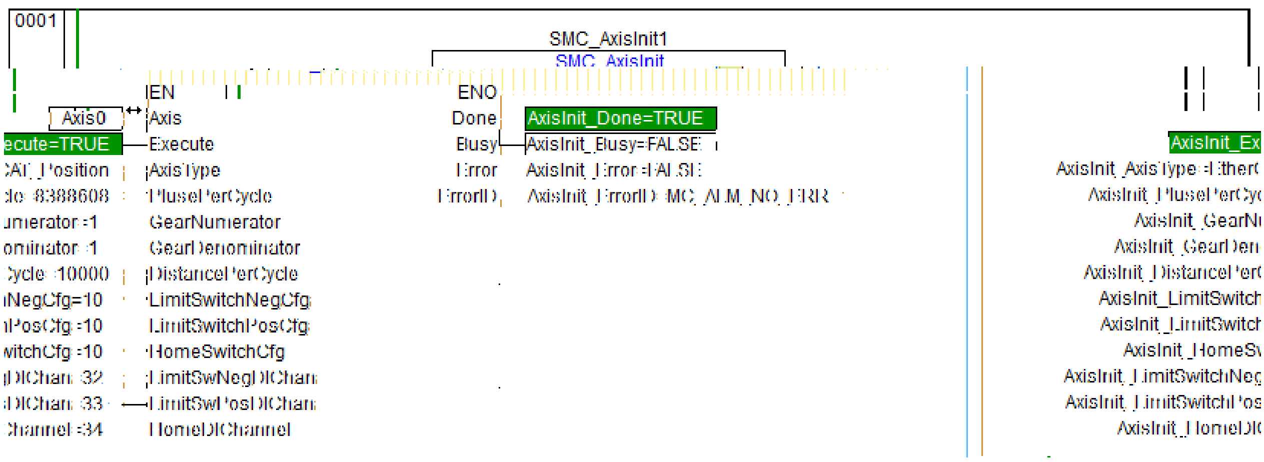
MC_ReadAxisInfo Axis0 Axis0

Axis0 PowerOn MC_Power Axis0

8	ReadAxisInfo_LimitSwitchPos	BOOL	FALSE	
9	ReadAxisInfo_LimitSwitchNeg	BOOL	FALSE	
10	ReadAxisInfo_Simulation	BOOL	FALSE	
11	ReadAxisInfo_ComRdy	BOOL	FALSE	
12	ReadAxisInfo_RdyPowerON	BOOL	FALSE	
13	ReadAxisInfo_PowerON	BOOL	FALSE	
14	ReadAxisInfo_Homed	BOOL	FALSE	
15	ReadAxisInfo_Warning	BOOL	FALSE	
16	ReadAxisInfo_Fault	BOOL	FALSE	
17	MC_Power1	MC_POWER		
18	Power_En	BOOL	FALSE	
19	Power_Status	BOOL	FALSE	
20	Power_Busy	BOOL	FALSE	
21	Power_Error	BOOL	FALSE	
22	Power_ErrorID	SMC_ERROR	MC_ALM_NO_ERR	ID

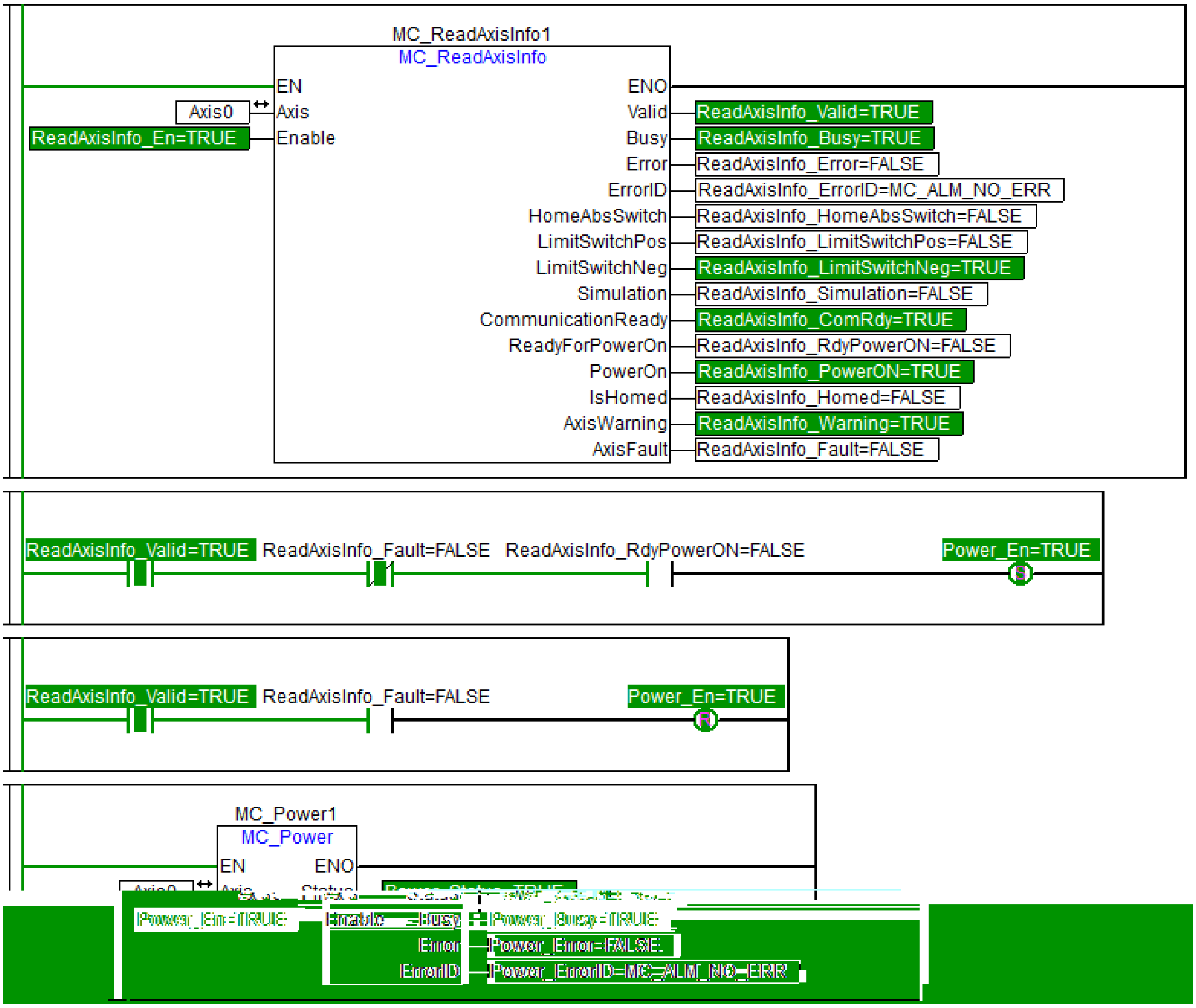
LD

1 SMC_AxisInit



2 MC_ReadAxisInfo

MC_Power



ST

1 MC_AxisInit

```
MC_ReadAxisInfo1(
Enable := ReadAxisInfo_En,
Axis := Axis0,
Valid=>ReadAxisInfo_Valid,
Busy=>ReadAxisInfo_Busy,
Error=>ReadAxisInfo_Error,
ErrorID=>ReadAxisInfo_ErrorID,
HomeAbsSwitch=>ReadAxisInfo_HomeAbsSwitch,
LimitSwitchPos=>ReadAxisInfo_LimitSwitchPos,
LimitSwitchNeg=>ReadAxisInfo_LimitSwitchNeg,
Simulation=>ReadAxisInfo_Simulation,
CommunicationReady=>ReadAxisInfo_ComRdy,
ReadyForPowerOn=>ReadAxisInfo_RdyPowerON,
PowerOn=>ReadAxisInfo_PowerON,
```

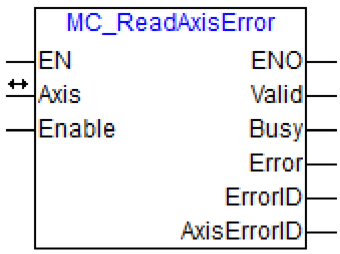
```
IsHomed=>ReadAxisInfo_Homed,
AxisWarning=>ReadAxisInfo_Warning,
AxisFault=>ReadAxisInfo_Fault);
```

2 MC_ReadAxisInfo

MC_Power

```
IF TRUE = ReadAxisInfo_Valid AND FALSE = ReadAxisInfo_Fault AND
TRUE = ReadAxisInfo_RdyPowerON THEN
   Power_En := TRUE;
ELSIF TRUE = ReadAxisInfo_Valid AND TRUE =  ReadAxisInfo_Fault THEN
   Power_En := FALSE;
END_IF;
MC_Power1(
Enable := Power_En,
Axis := Axis0,
Status=>Power_Status,
Busy=>Power_Busy,
Error=>Power_Error,
ErrorID=>Power_ErrorID);
```

5.22 MC_ReadAxisError



5.22.1

IN_OUT	Axis			-	-	AXIS_REF
INPUT	Enable			-	TRUE/FALSE	BOOL
OUTPUT	Valid			-	TRUE/FALSE	BOOL
	Busy			FALSE	TRUE/FALSE	BOOL
	Error			FALSE	TRUE/FALSE	BOOL

	ErrorID			0	-	SMC_ERROR
	AxisErrorID			0	-	DWORD

5.22.2

1

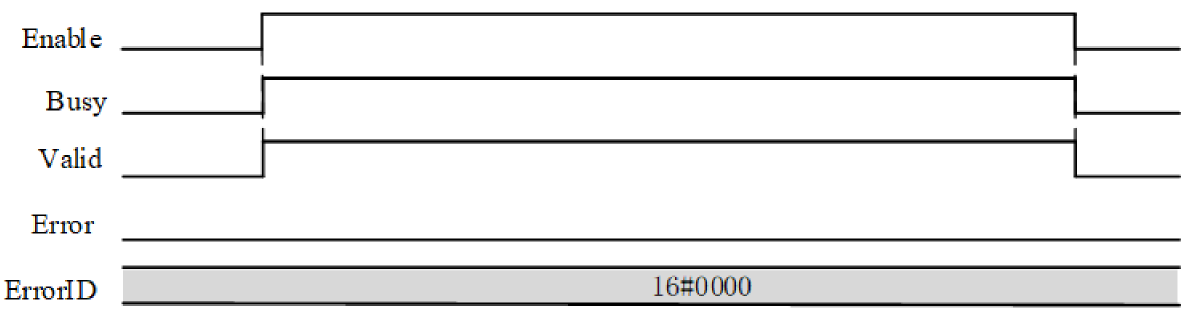
Valid TRUE AxisErrorID 0 AxisErrorID

2 Enable( ) TRUE Enable( ) FALSE

3

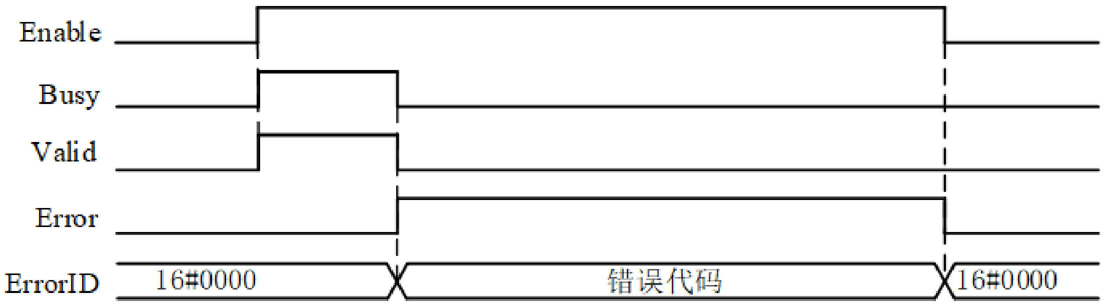
100001		
100002		
100003		
200000+		
0-65535		603F

1



2

Error( ))



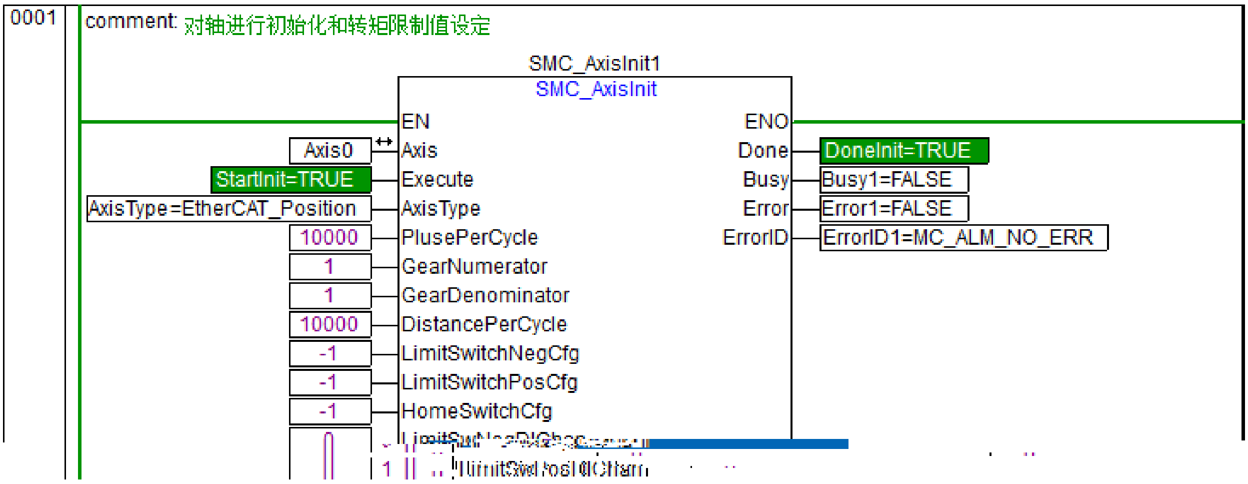
5.22.3

MC_ ReadAxisError

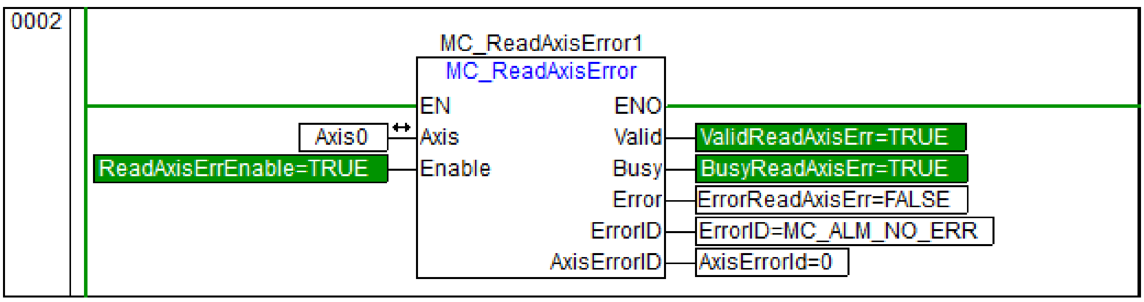
Axis0	AXIS_REF		0
StartInit	BOOL	FALSE	TRUE
DoneInit	BOOL	FALSE	TRUE
ReadAxisErrEnable	BOOL	FALSE	TRUE
ValidReadAxisErr	BOOL	FALSE	TRUE
BusyReadAxisErr	BOOL	FALSE	TRUE
ErrorReadAxisErr	BOOL	FALSE	TRUE
AxisErrorId	DWORD	0	

MC_ ReadAxisError Enable( ) TRUE

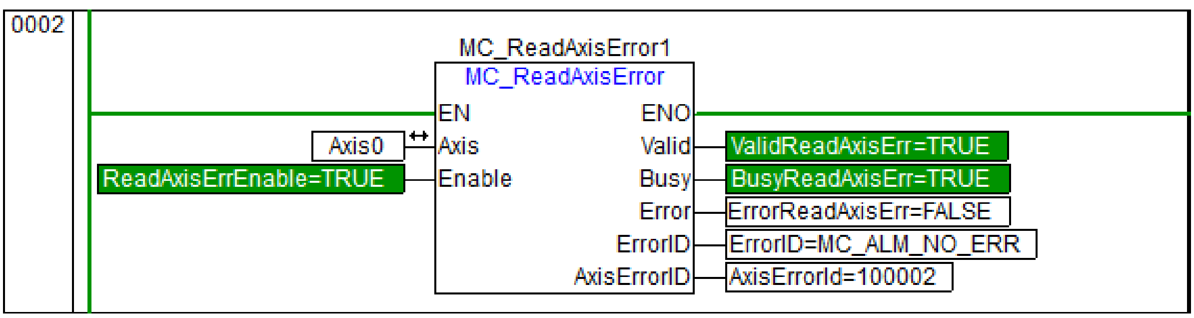
1



2



3



ST

1

```
SMC_AxisInit1(
Execute := StartInit,
AxisType := 50,
PlusePerCycle := 10000,
GearNumerator := 1,
GearDenominator := 1,
```

```
DistancePerCycle := 10000,
LimitSwitchNegCfg := -1,
LimitSwitchPosCfg := -1,
HomeSwitchCfg := -1,
LimitSwNegDIChan := 0,
LimitSwPosDIChan := 1,
HomeDIChannel := 2,
Axis := Axis0,
Done=> DoneInit,
Busy=>Busy1,
Error=>Error1,
ErrorID=>ErrorID1);
```

2

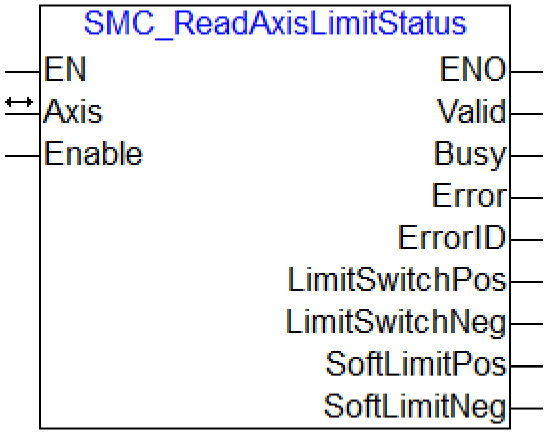
```
MC_ReadAxisError1(
Enable := ReadAxisErrEnable,
Axis := Axis0,
Valid=>ValidReadAxisErr,
Busy=>BusyReadAxisErr,
Error=>ErrorReadAxisErr ,
ErrorID=>ErrorID,
AxisErrorID=>AxisErrorId);
```

5.22.4

Enable

5.22.5

5.23 SMC_ReadAxisLimitStatus



5.23.1

IN_OUT	Axis			-	-	AXIS_REF
INPUT	Enable			-	TRUE/FALSE	BOOL
OUTPUT	Valid			-	TRUE/FALSE	BOOL
	Busy			FALSE	TRUE/FALSE	BOOL
	Error			FALSE	TRUE/FALSE	BOOL
	ErrorID			0	-	SMC_ERROR
	LimitSwitchPos			FALSE	TRUE/FALSE	BOOL
	LimitSwitchNeg			FALSE	TRUE/FALSE	BOOL
	SoftLimitPos			FALSE	TRUE/FALSE	BOOL
	SoftLimitNeg			FALSE	TRUE/FALSE	BOOL

5.23.2

Enable=TRUE

Valid Enable FALSE

TRUE

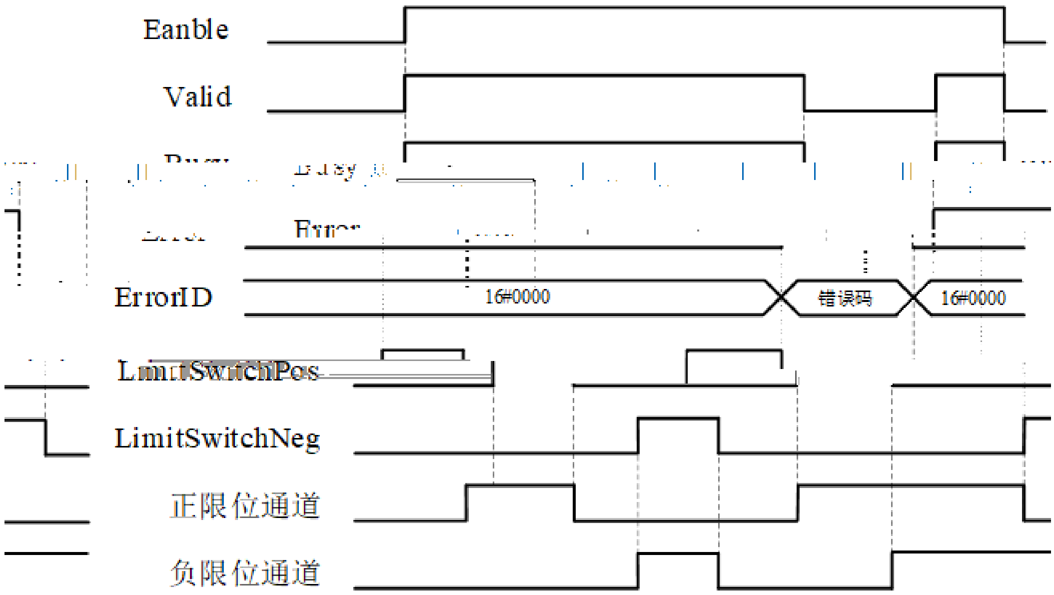
FALSE

FSLimit RSLimit MPos FSLimit

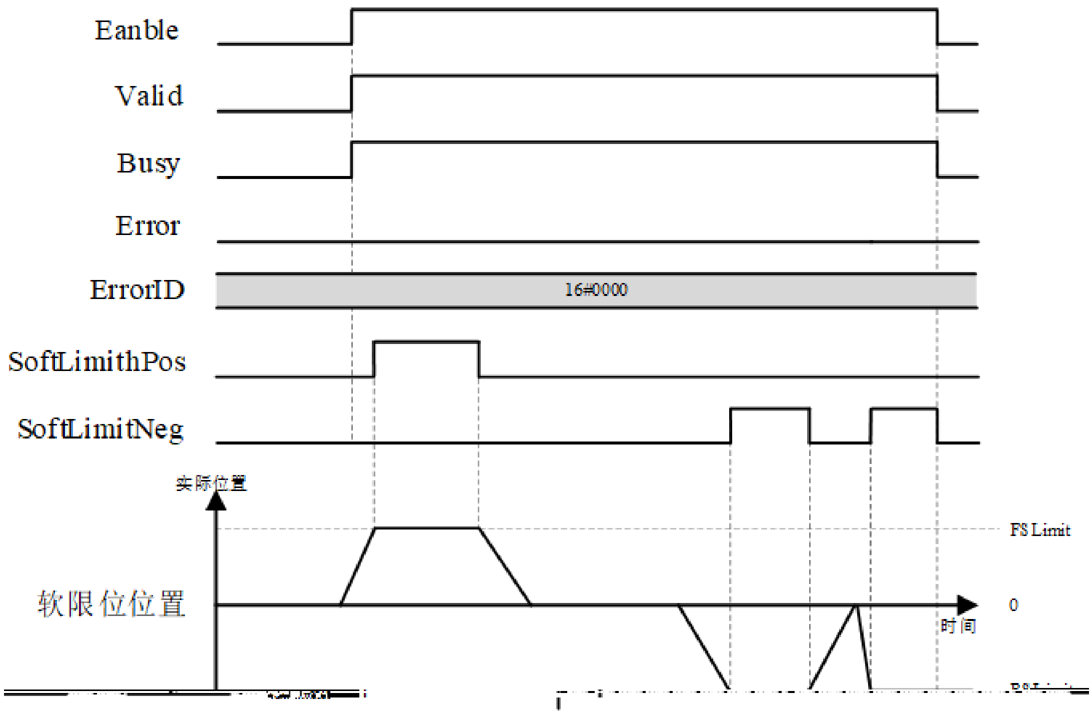
TRUE FALSE MPos RSLimit

TRUE FALSE

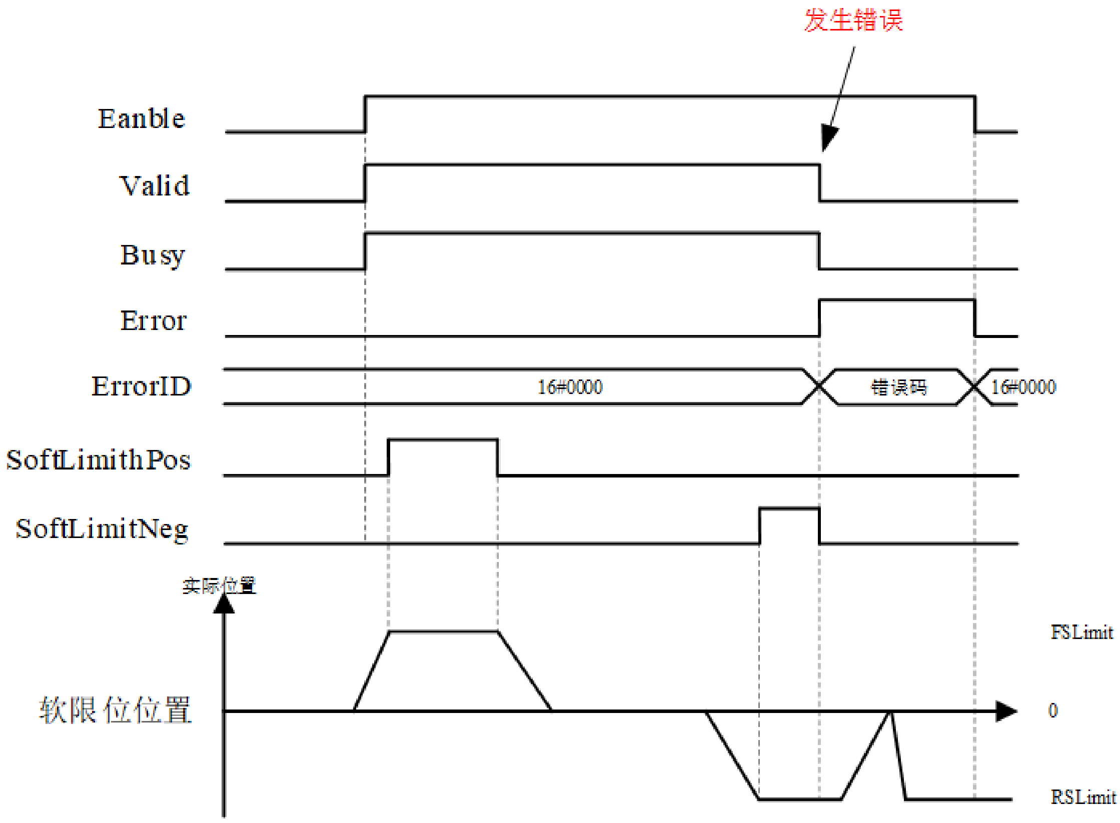
1



2


Eanble
Valid
Busy
Error
ErrorID
16#0000
SoftLimithPos
SoftLimitNeg
实际位置
软限位位置
FS Limit
0
时间

3



5.23.3

SMC_AxisInit

SMC_ReadAxisLimitStatus Axis0

%IX4.0 SMC_ReadAxisLimitStatus

LimitSwitchNeg TRUE

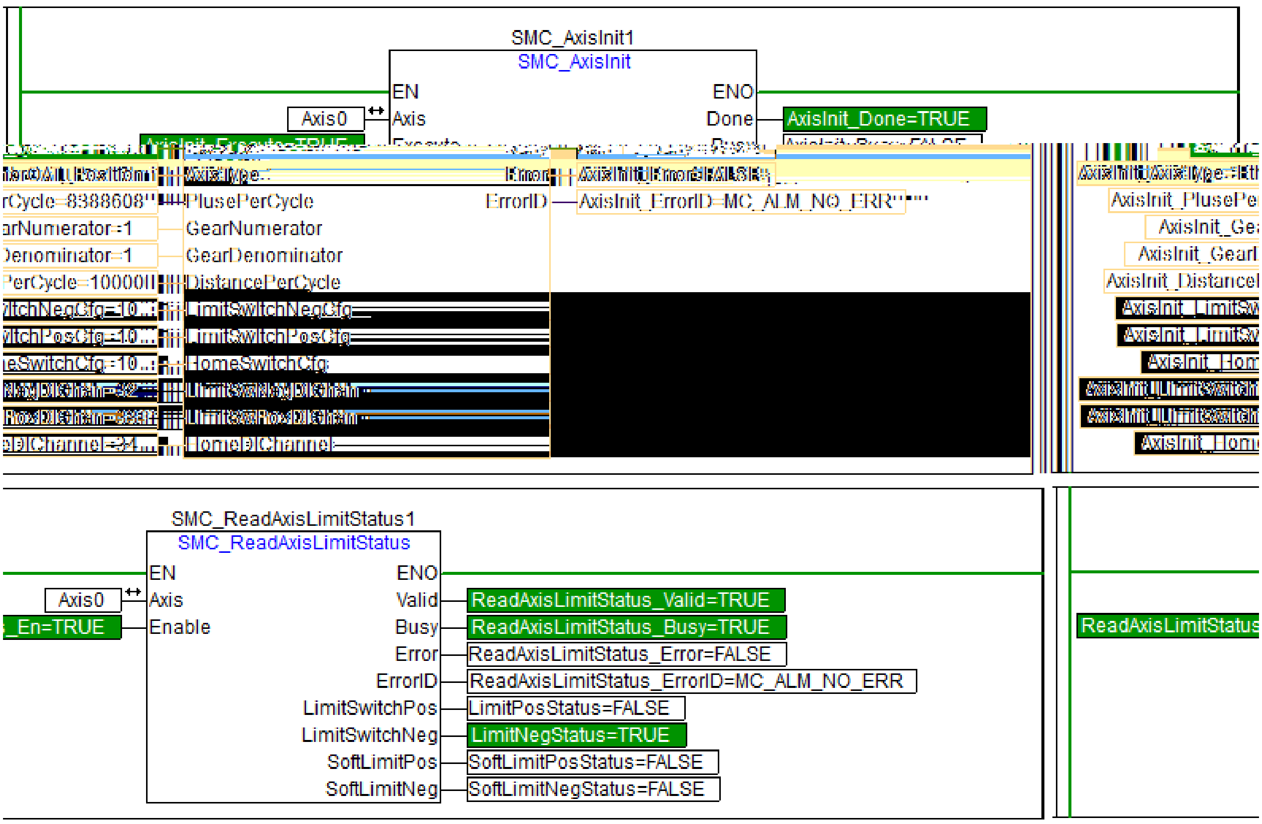
1	SMC_ReadAxisLimitStatus1	SMC_ReadAxisLimitStatus		
2	ReadAxisLimitStatus_En	BOOL	FALSE	
3	ReadAxisLimitStatus_Valid	BOOL	FALSE	
4	ReadAxisLimitStatus_Busy	BOOL	FALSE	
5	ReadAxisLimitStatus_Error	BOOL	FALSE	
6	ReadAxisLimitStatus_ErrorID	SMC_ERROR	MC_ALM_NO_ERROR	ID

7	LimitPosStatus	BOOL	FALSE	
8	LimitNegStatus	BOOL	FALSE	
9	SoftLimitPosStatus	BOOL	FALSE	
10	SoftLimitNegStatus	BOOL	FALSE	

LD

SMC_AxisInit SMC_ReadAxisLimitStatus

TRUE



ST

1 SMC_AxisInit

```
SMC_AxisInit1(
Execute := AxisInit_Execute,
AxisType := AxisInit_AxisType,
PlusePerCycle := AxisInit_PlusePerCycle,
GearNumerator := AxisInit_GearNumerator,
GearDenominator := AxisInit_GearDenominator,
DistancePerCycle := AxisInit_DistancePerCycle,
```

```
LimitSwitchNegCfg := AxisInit_LimitSwitchNegCfg,
LimitSwitchPosCfg := AxisInit_LimitSwitchPosCfg,
HomeSwitchCfg := AxisInit_HomeSwitchCfg,
LimitSwNegDIChan := AxisInit_LimitSwitchNegDIChan,
LimitSwPosDIChan := AxisInit_LimitSwitchPosDIChan,
HomeDIChannel := AxisInit_HomeDIChannel,
Axis := Axis0,
Done=>AxisInit_Done,
Busy=>AxisInit_Busy,
Error=>AxisInit_Error,
ErrorID=>AxisInit_ErrorID);
```

2 SMC_ReadAxisLimitStatus

```
SMC_ReadAxisLimitStatus1(
Enable := ReadAxisLimitStatus_En,
Axis := Axis0,
Valid=>ReadAxisLimitStatus_Valid,
Busy=>ReadAxisLimitStatus_Busy,
Error=>ReadAxisLimitStatus_Error,
ErrorID=>ReadAxisLimitStatus_ErrorID,
LimitSwitchPos=>LimitPosStatus,
LimitSwitchNeg=>LimitNegStatus,
SoftLimitPos=>SoftLimitPosStatus,
SoftLimitNeg=>SoftLimitNegStatus);
```

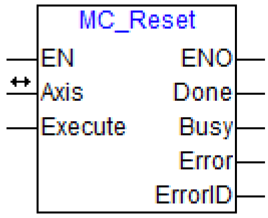
5.23.4

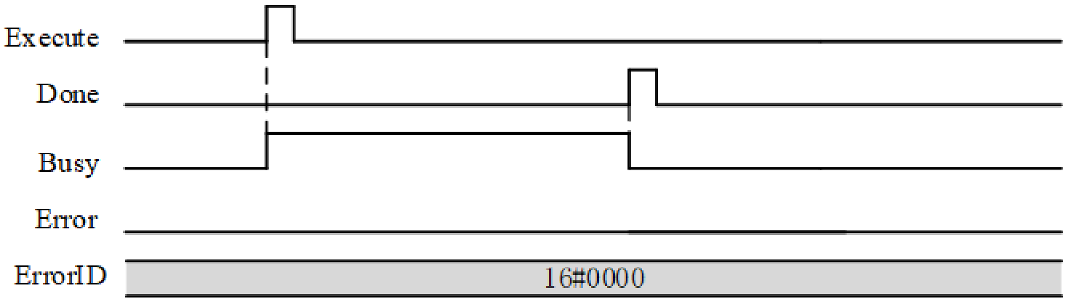
RepMode =1 2

5.23.5

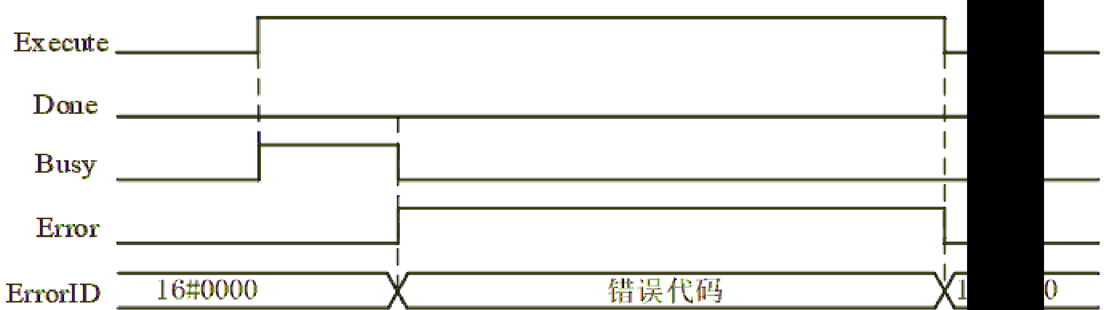
SMC_AxisInit

5.24 MC_Reset


MC_Reset
EN
ENO
Axis
Done
Execute
Busy
Error
ErrorID



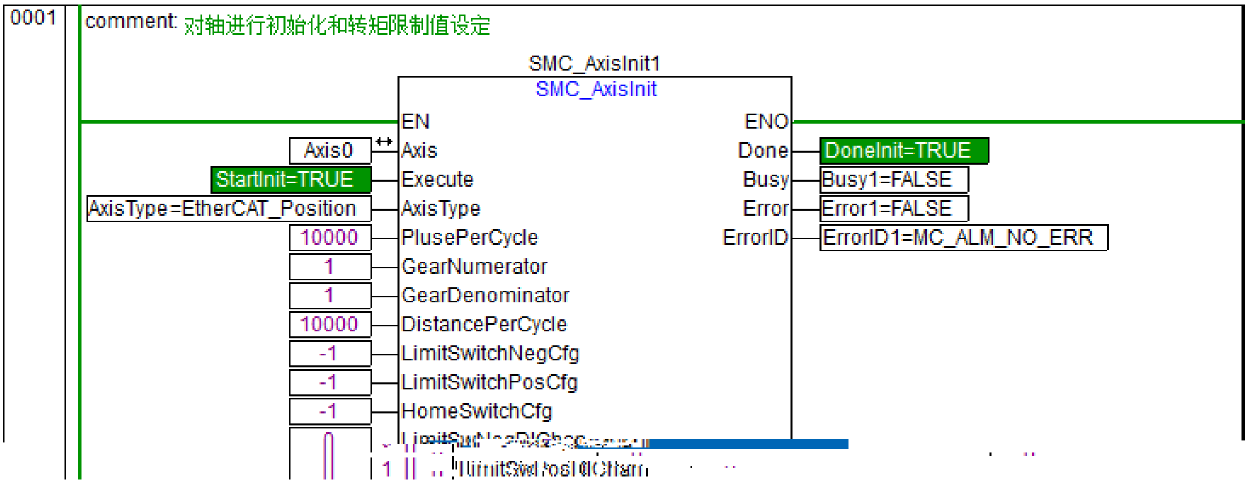
2



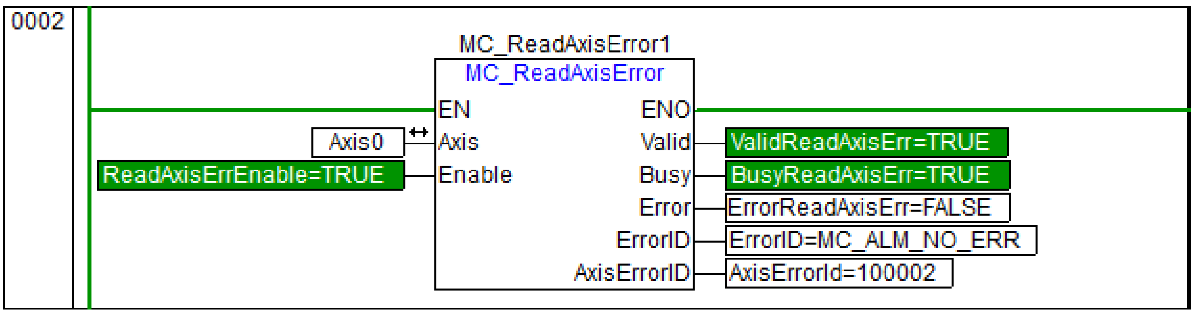
5.24.3

MC_ Reset

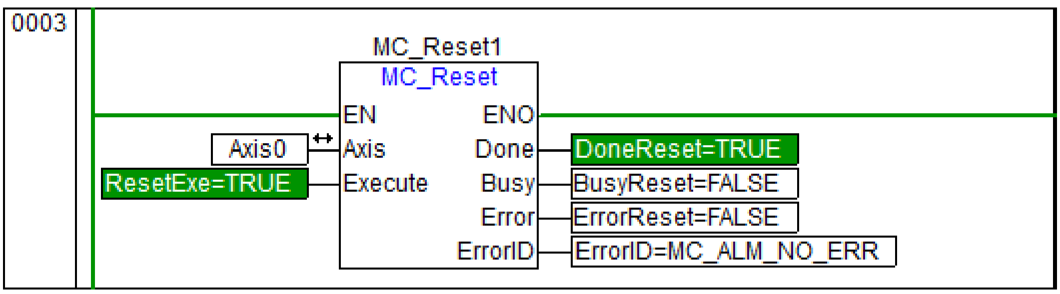
Axis0	AXIS_REF		0


0001
comment: 对轴进行初始化和转矩限制值设定
SMC_AxisInit1
SMC_AxisInit
EN
ENO
Axis0
Axis
Done
DoneInit=TRUE
StartInit=TRUE
Execute
Busy
Busy1=FALSE
AxisType=EtherCAT_Position
AxisType
Error
Error1=FALSE
10000
PlusePerCycle
ErrorID
ErrorID1=MC_ALM_NO_ERR
1
GearNumerator
1
GearDenominator
10000
DistancePerCycle
-1
LimitSwitchNegCfg
-1
LimitSwitchPosCfg
-1
HomeSwitchCfg

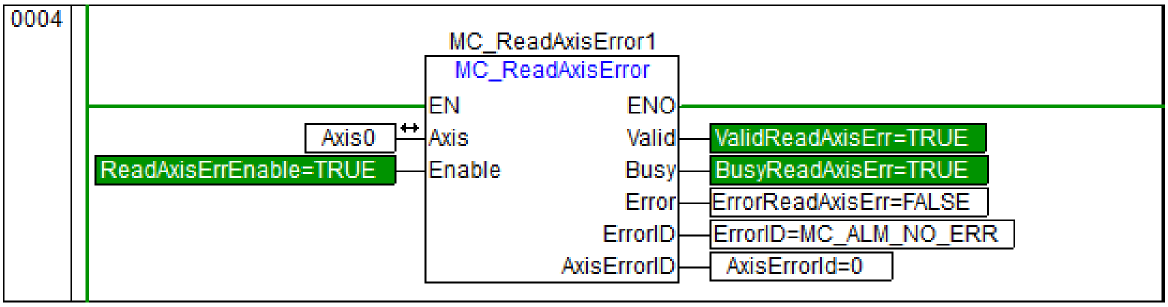
2


0002
MC_ReadAxisError1
MC_ReadAxisError
EN
ENO
Axis0
Axis
Valid
ValidReadAxisErr=TRUE
ReadAxisErrEnable=TRUE
Enable
Busy
BusyReadAxisErr=TRUE
Error
ErrorReadAxisErr=FALSE
ErrorID
ErrorID=MC_ALM_NO_ERR
AxisErrorID
AxisErrorId=100002

3 MC_Reset


0003
MC_Reset1
MC_Reset
EN
ENO
Axis0
Axis
Done
DoneReset=TRUE
ResetExe=TRUE
Execute
Busy
BusyReset=FALSE
Error
ErrorReset=FALSE
ErrorID
ErrorID=MC_ALM_NO_ERR

4



ST

1

```
SMC_AxisInit1(
Execute := StartInit,
AxisType := 50,
PlusePerCycle := 10000,
GearNumerator := 1,
GearDenominator := 1,
DistancePerCycle := 10000,
LimitSwitchNegCfg := -1,
LimitSwitchPosCfg := -1,
HomeSwitchCfg := -1,
LimitSwNegDIChan := 0,
LimitSwPosDIChan := 1,
HomeDIChannel := 2,
Axis := Axis0,
Done=> DoneInit,
Busy=>Busy1,
Error=>Error1,
ErrorID=>ErrorID1);
```

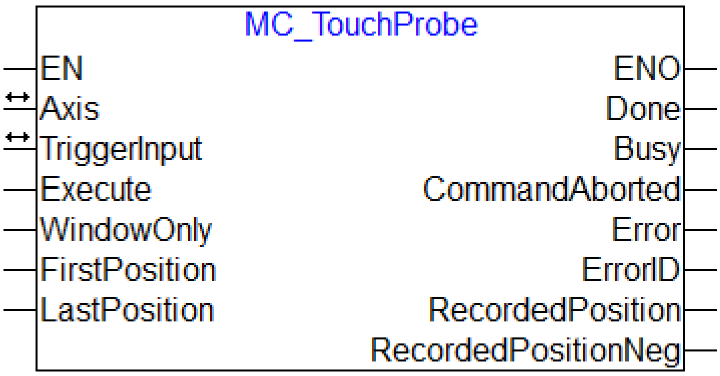
2 MC_Reset

```
MC_Reset1(
Execute := ResetExe,
Axis := Axis0,
Done=>DoneReset,
Busy=>BusyReset,
Error=>ErrorReset,
ErrorID=>ErrorID);
```

5.24.4

5.24.5

5.25 MC_TouchProbe



5.25.1

IN_OUT	Axis			-	-	AXIS_REF
	TriggerInput			-	MC_TRIGGER_REF	MC_TRIGGER_REF
INPUT	Execute			-	TRUE/FALSE	BOOL
		FALSE TRUE [FirstPosition, LastPosition]		FALSE	TRUE/FALSE	BOOL
	FirstPosition			0	/ /0	LREAL
	LastPosition			0	/ /0	LREAL
OUTPUT	Done			FALSE	TRUE/FALSE	BOOL

	Busy			FALSE	TRUE/FALSE	BOOL
	CommandAborted			FALSE	TRUE/FALSE	BOOL
	Error			FALSE	TRUE/FALSE	BOOL
	ErrorID			0	-	SMC_ERROR
	RecordedPosition			-	/ /0	LREAL
	RecordedPositionNeg			-	/ /0	LREAL

LatchMode	0	0	0	USINT
LatchID	0 Latch 1 1 Latch 2	0	0/1	USINT
DriveInput	0 IO 1 Z	0	0/1	UDINT
TriggerEdge	0 1 2	0	0/1/2	USINT
ContinuousTrigger		FALSE	TRUE/FALSE	BOOL

5.25.2

CIA402 EtherCAT ECI002

ErrorStop

DI Z

PDO

Touch probe function (60B8 hex)

Touch probe status (60B9 hex)

Touch probe pos1 pos value (60BA hex 1)

Touch probe pos1 neg value (60BB hex 1)

Touch probe pos2 pos value (60BC hex 2)

Touch probe pos2 neg value (60BD hex 2)

Excute TriggerInput TriggerInput

Excute

Error TRUE Busy TRUE LatchID

RecordedPosition RecordedPositionNeg Done TRUE

Busy Done Error

RecordedPosition RecordedPositionNeg

Done Excute

RecordedPosition

RecordedPositionNeg Done

Done

Done IEC

IEC

WindowOnly FALSE

WindowOnly TRUE

1

FirstPosition ≤ ≤ LastPosition FirstPosition LastPosition

2

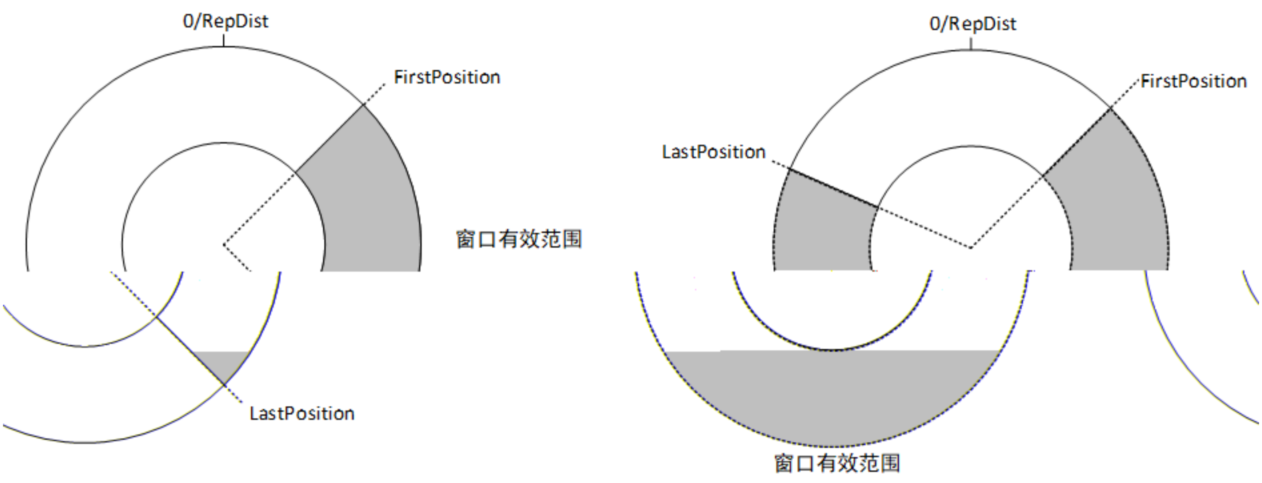
FirstPosition ≤ LastPosition) FirstPosition LastPosition

FirstPosition LastPosition

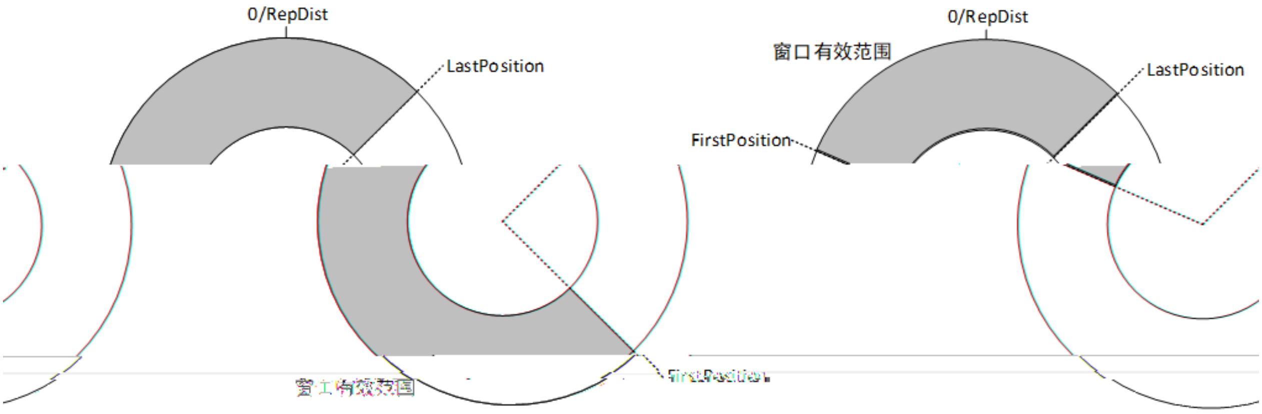
FirstPosition LastPosition

1 [0,RepDist]

FirstPosition ≤ LastPosition

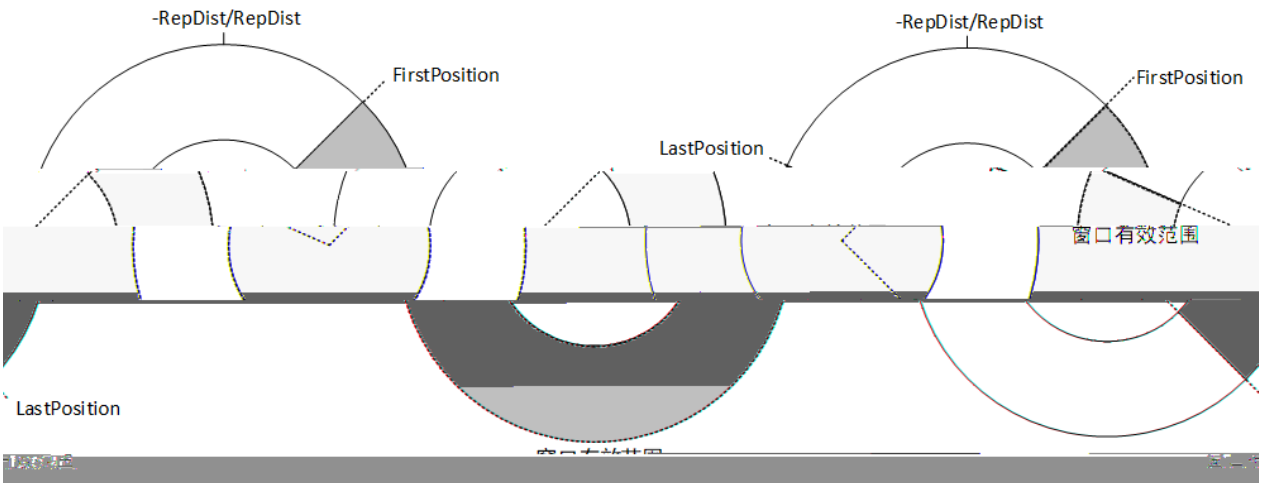


FirstPosition > LastPosition



2 [-RepDist,RepDist]

FirstPosition ≤ LastPosition



FirstPosition > LastPosition

1 2 ID

ID

ID

MC_AbortTrigger MC_TouchProbe

MC_AbortTrigge Done TRUE MC_TouchProb

CommandAborted

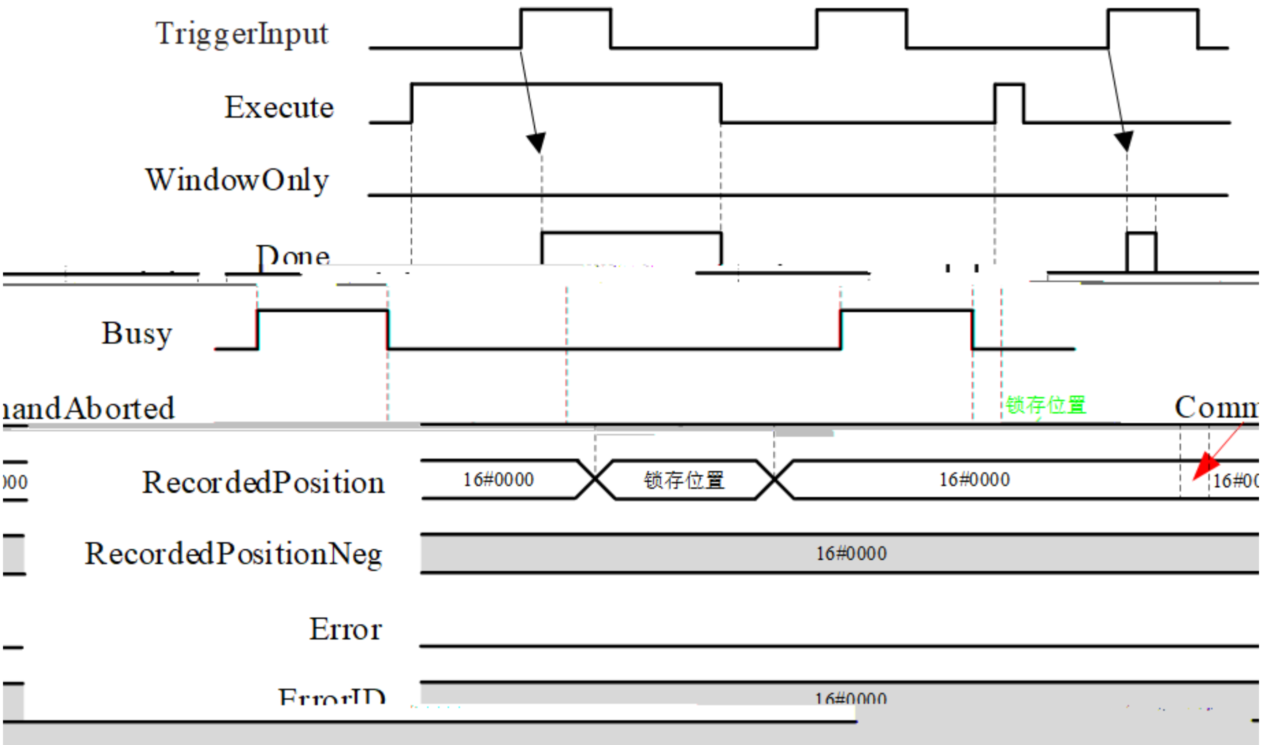
MC_Home HMC_GantryInit Z

1 Z 1

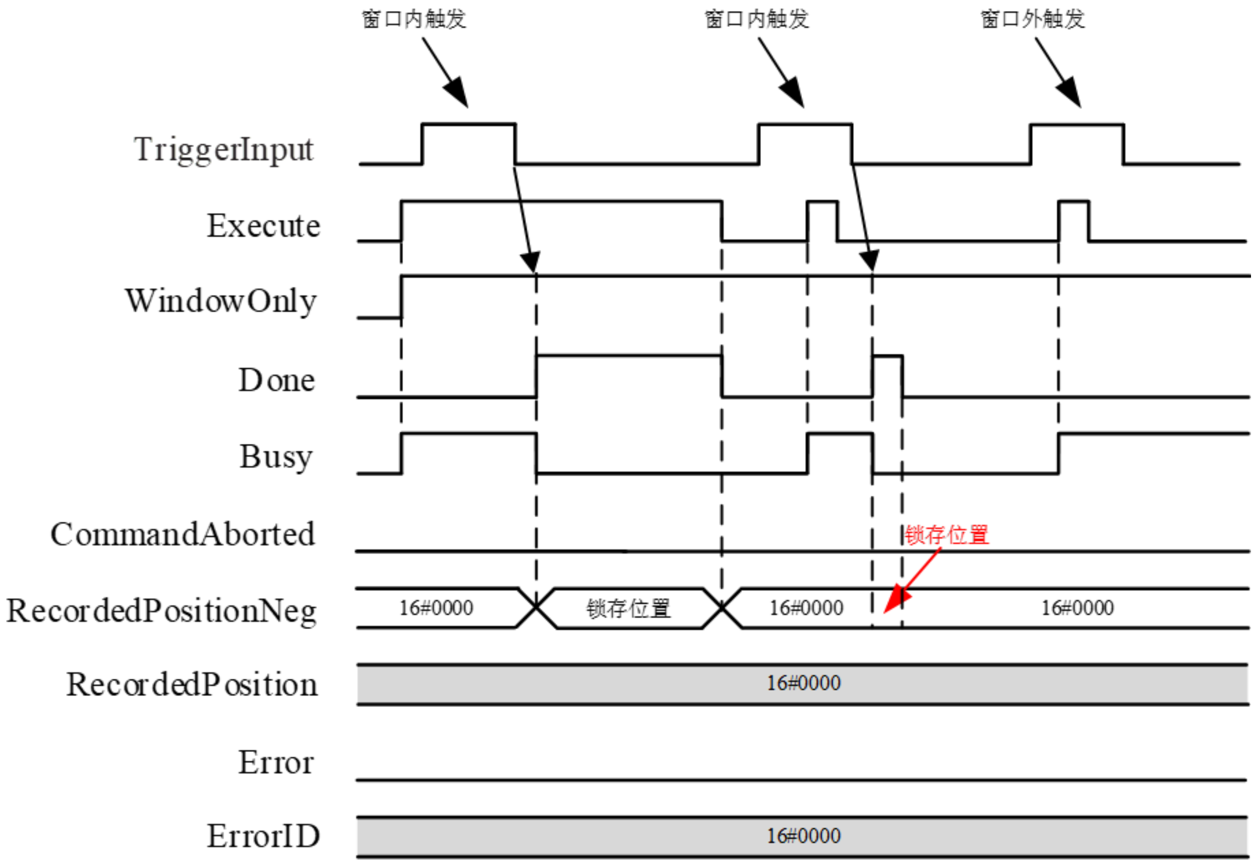
MC_Home HMC_GantryInit Z 1

MC_Home HMC_GantryInit

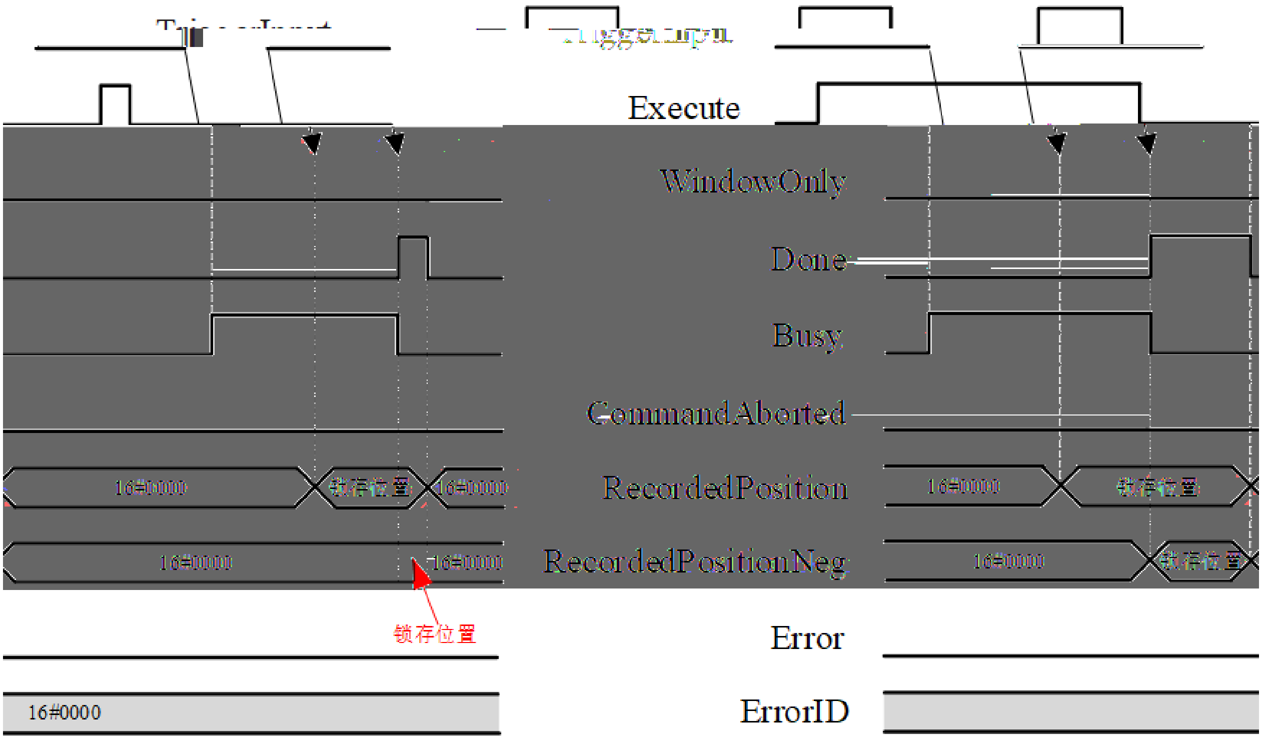
1 1 DI


TriggerInput
Execute
WindowOnly
Busy
锁存位置
RecordedPosition
16#0000
锁存位置
16#0000
RecordedPositionNeg
16#0000
Error

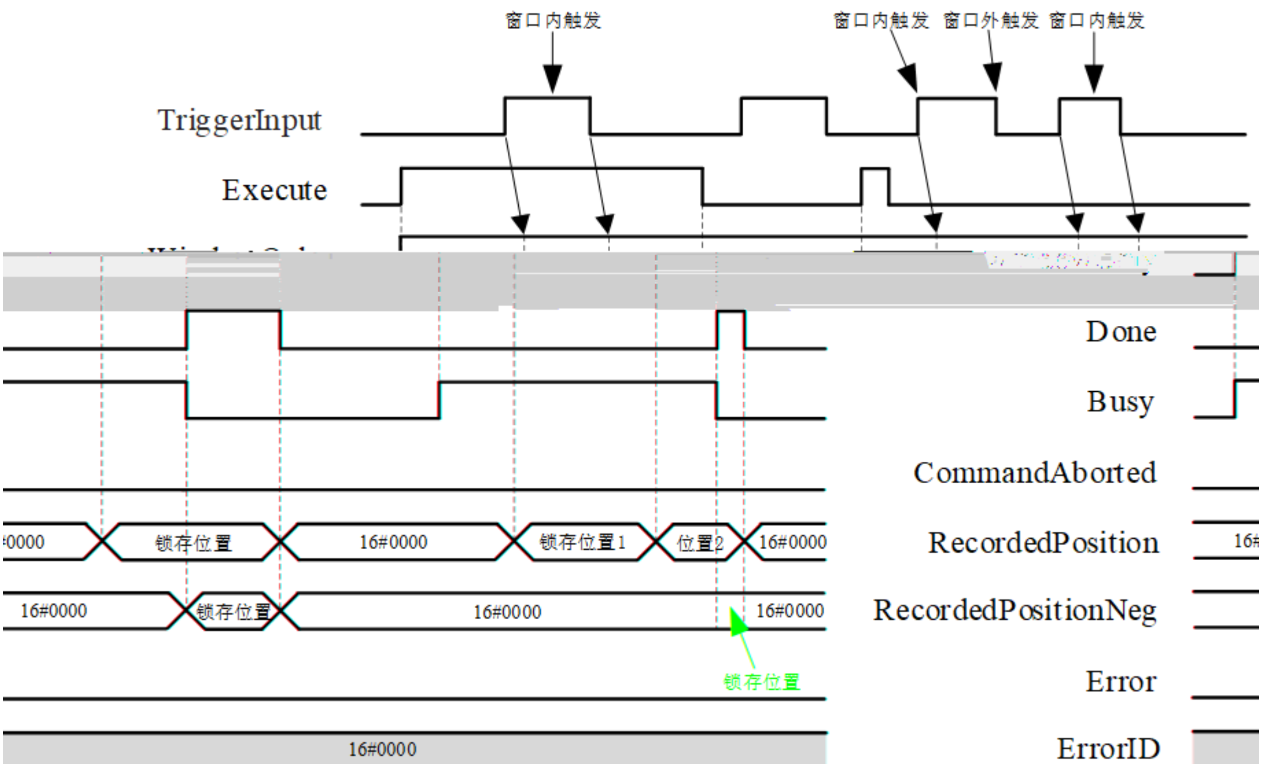
2 1 DI


窗口内触发
窗口内触发
窗口外触发
TriggerInput
Execute
WindowOnly
Done
Busy
CommandAborted
锁存位置
RecordedPositionNeg
16#0000
锁存位置
16#0000
16#0000
RecordedPosition
16#0000
Error
ErrorID
16#0000

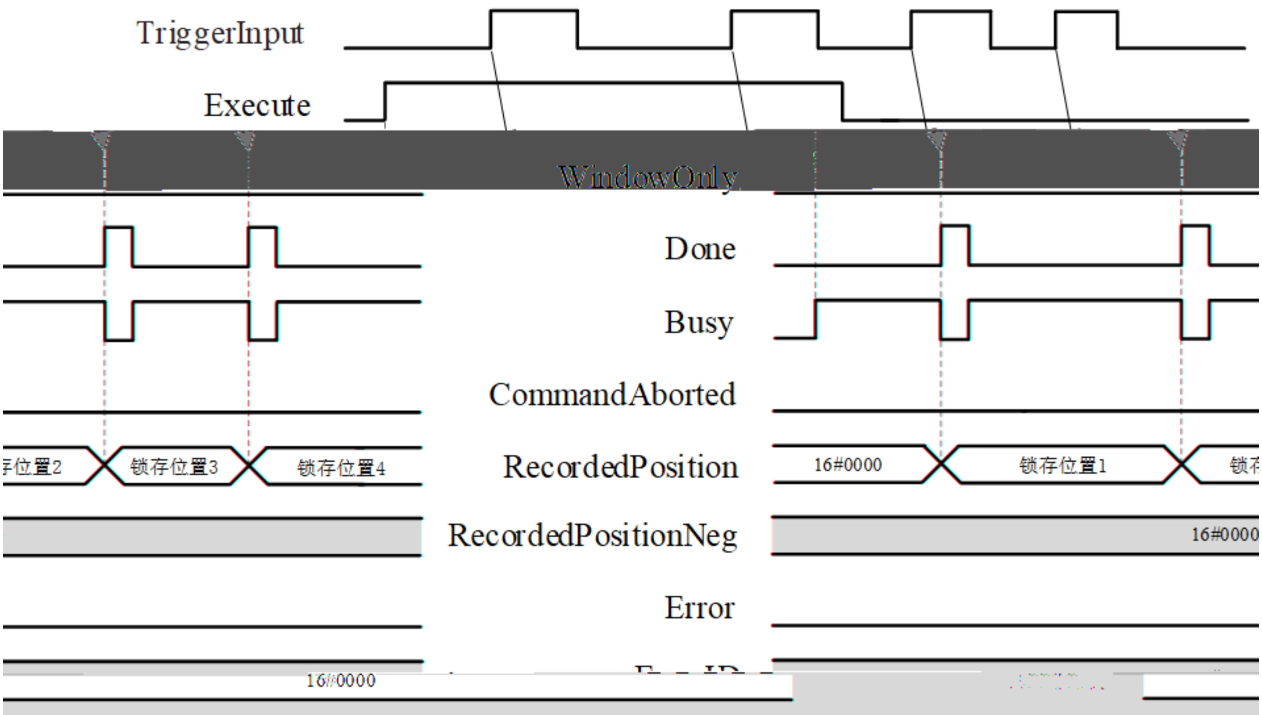
3 1 DI


TriggerInput
Execute
WindowOnly
Done
Busy
CommandAborted
RecordedPosition
RecordedPositionNeg
Error
ErrorID
16#0000
锁存位置

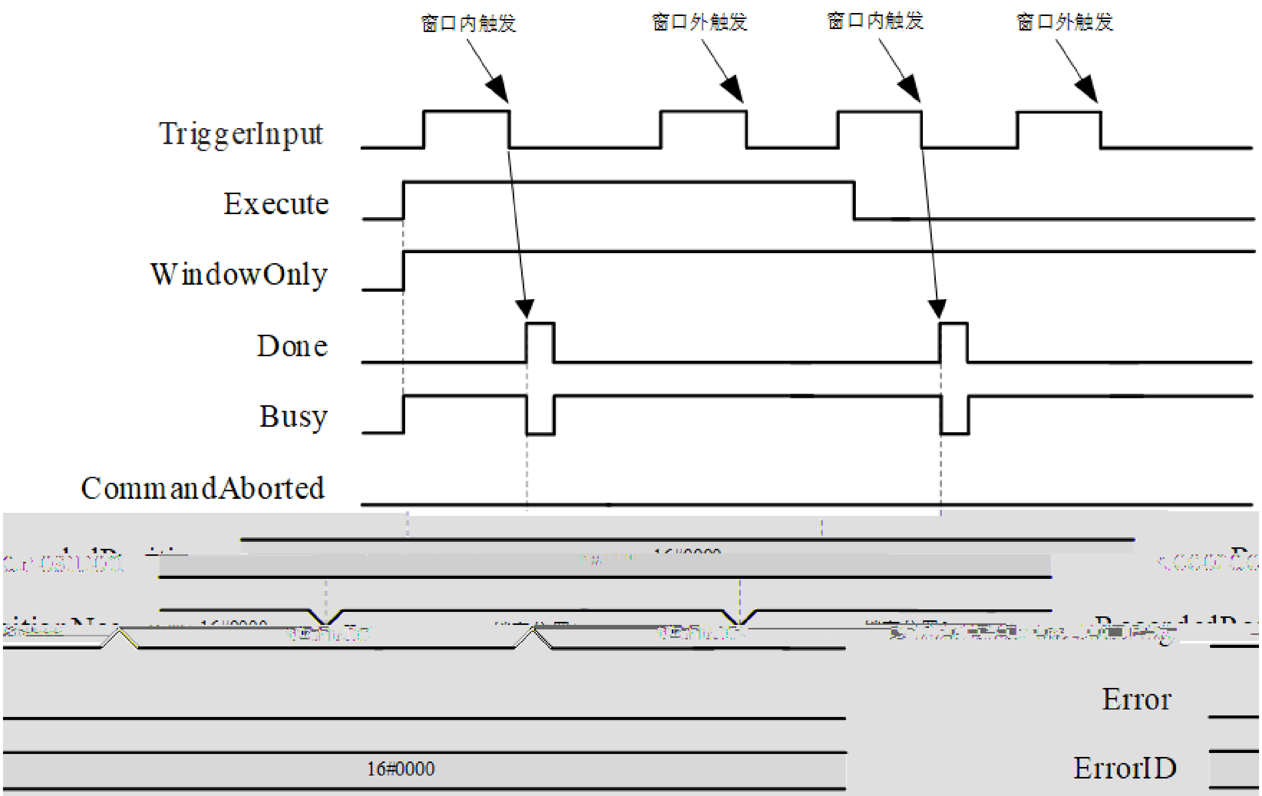
4 1 DI


窗口内触发
窗口内触发
窗口外触发
窗口内触发
TriggerInput
Execute
Done
Busy
CommandAborted
RecordedPosition
RecordedPositionNeg
Error
ErrorID
16#0000
锁存位置
锁存位置1
位置2

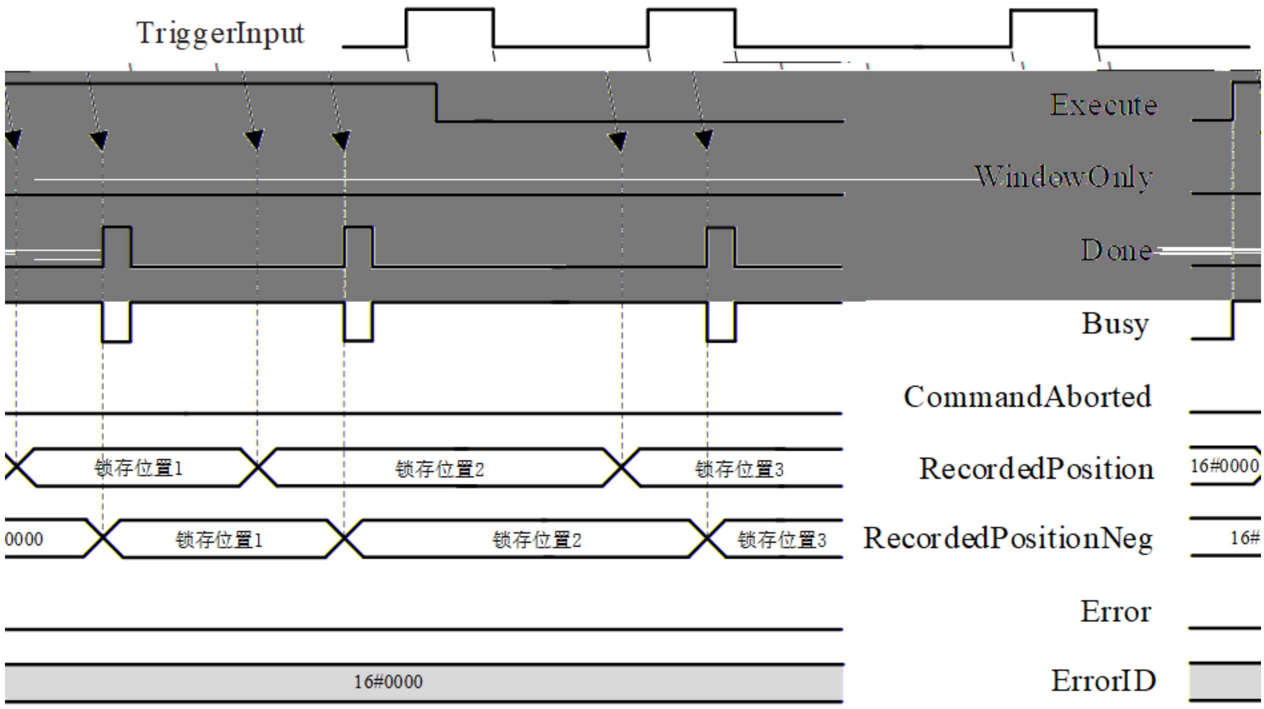
5 1 DI


TriggerInput
Execute
WindowOnly
Done
Busy
CommandAborted
RecordedPosition
锁存位置3
锁存位置4
16#0000
锁存位置1
RecordedPositionNeg
16#0000
Error
16#0000

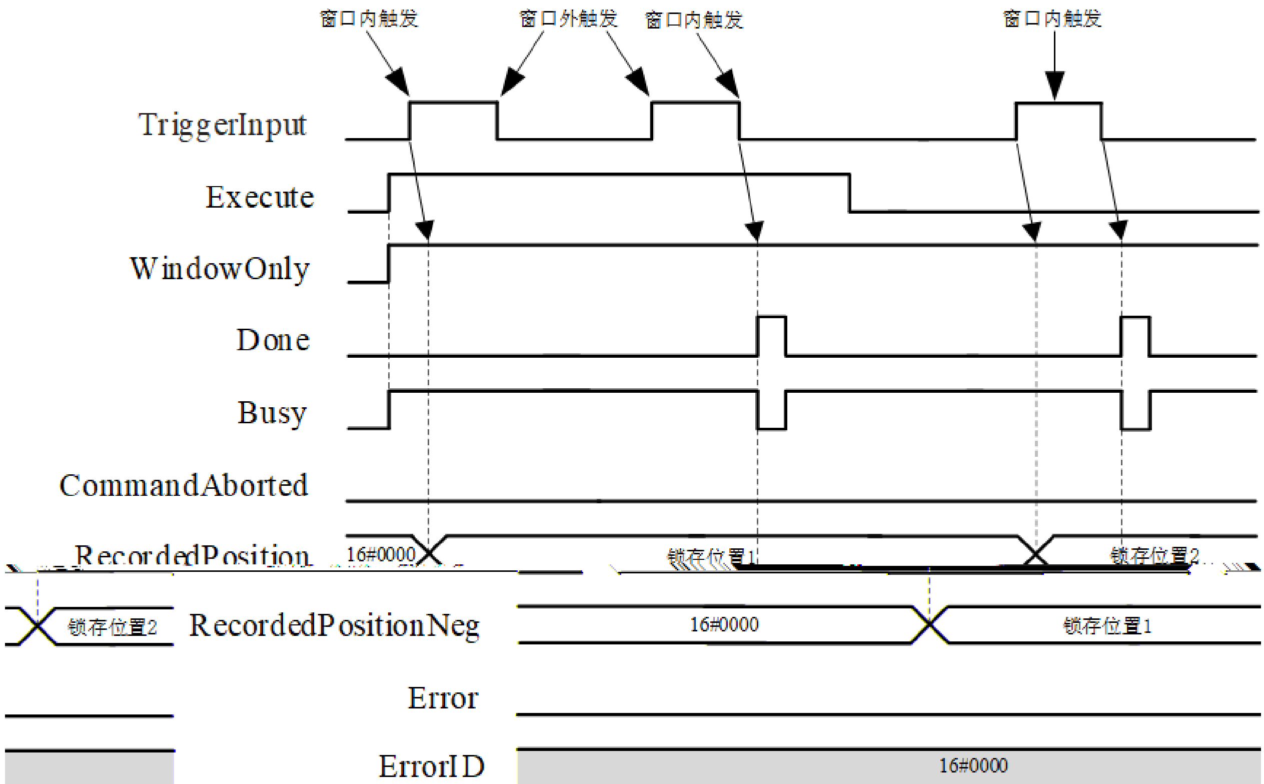
6 1 DI


窗口内触发
窗口外触发
窗口内触发
窗口外触发
TriggerInput
Execute
WindowOnly
Done
Busy
CommandAborted
Error
ErrorID
16#0000

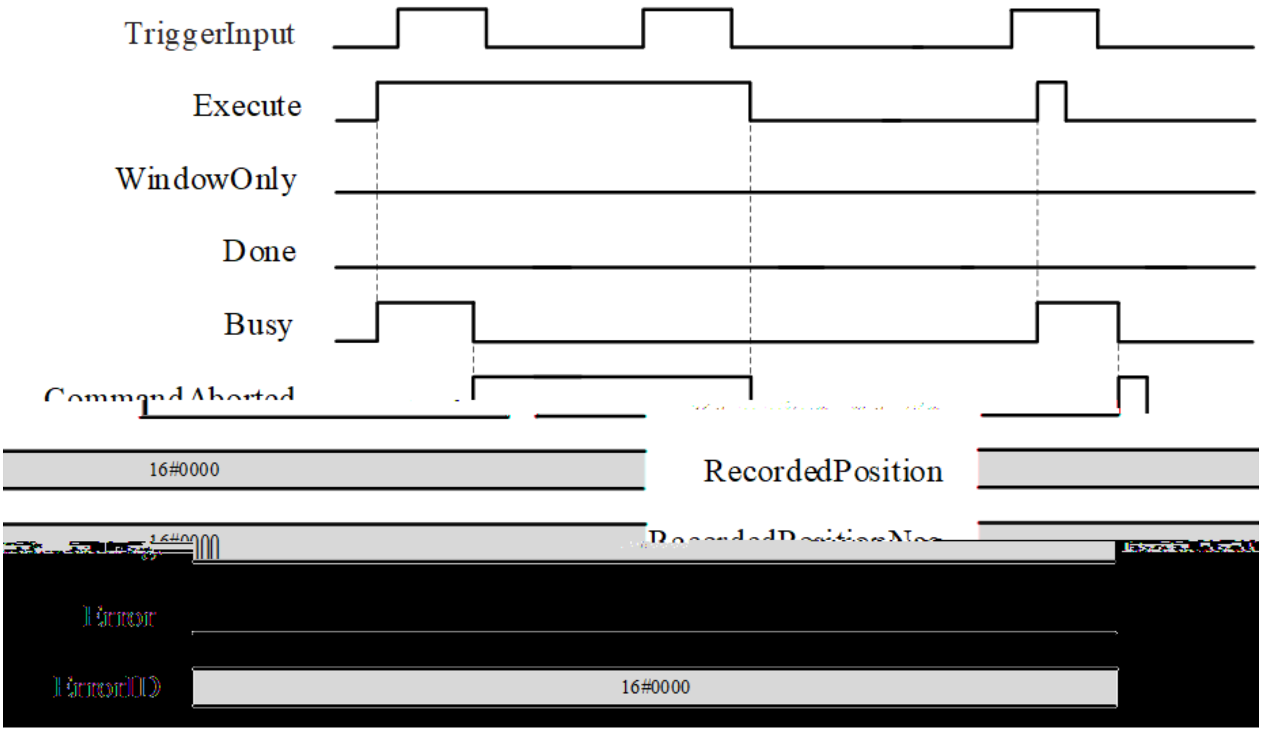
7 1 DI



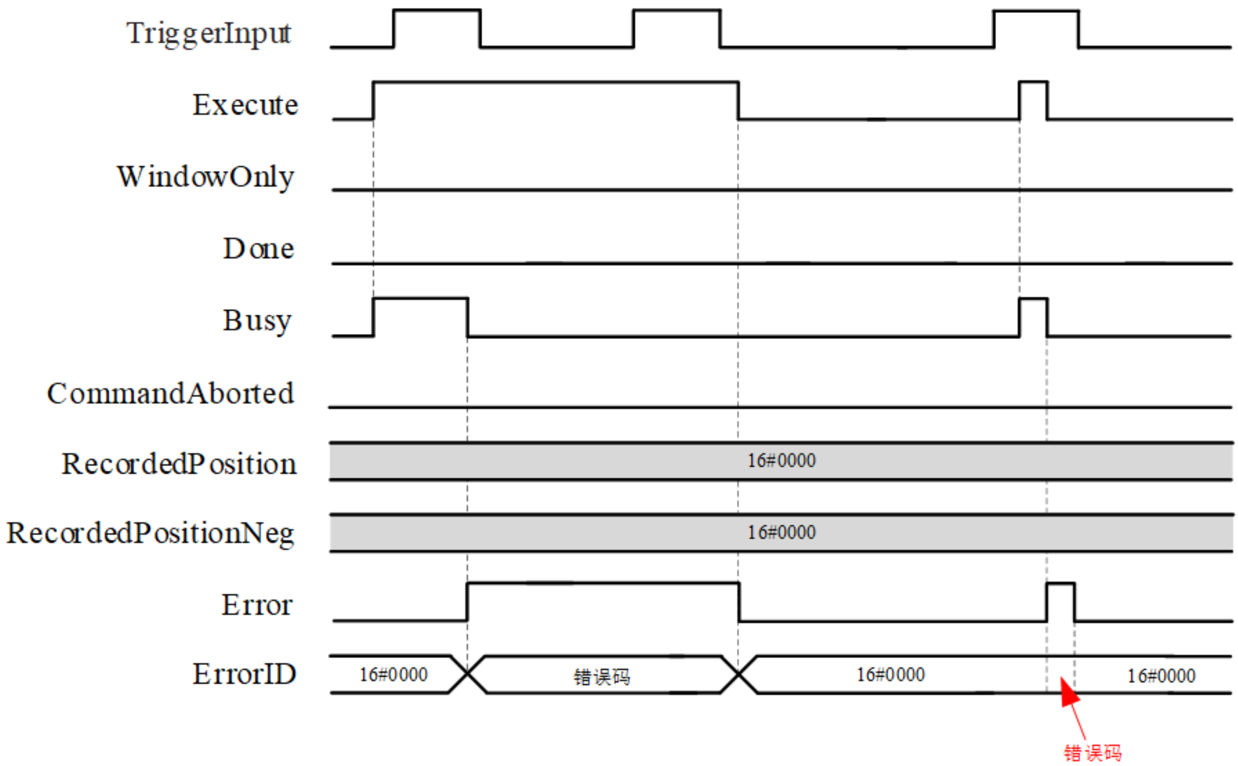
8 1 DI



9 1 DI MC_AbortTrigger


TriggerInput
Execute
WindowOnly
Done
Busy
16#0000
RecordedPosition
Error
ErrorID
16#0000

10 1 DI



5.25.3

MC_TouchProbe EtherCAT

1

IO

SMC_AxisInit

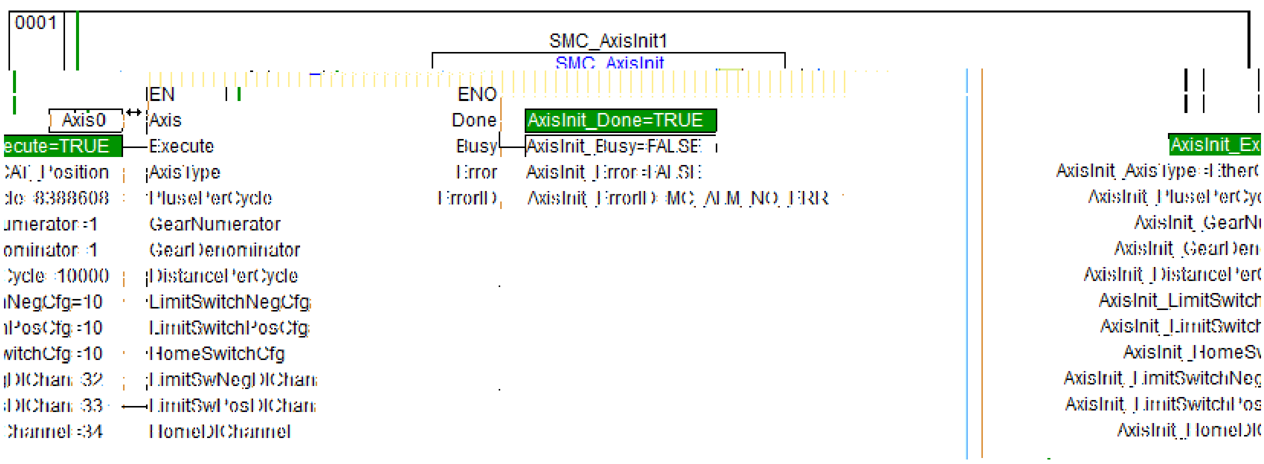
PDO

DI 1

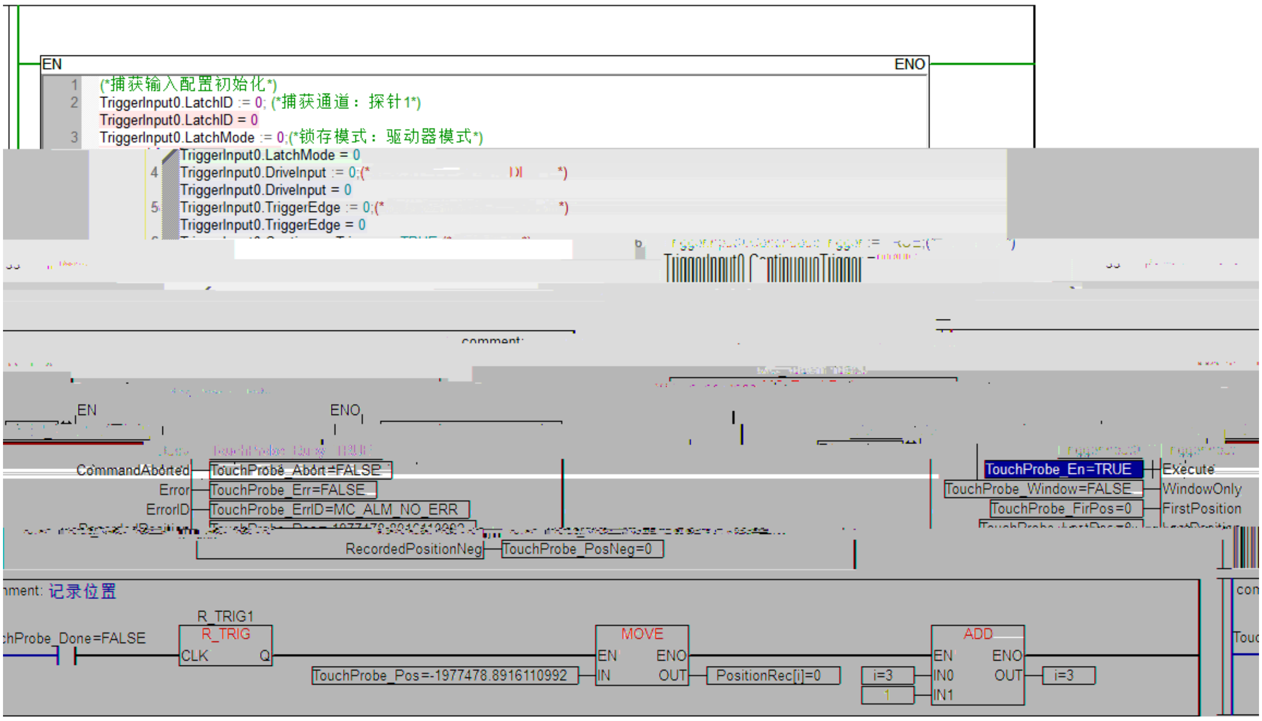
1	MC_TouchProbe0	MC_TOUCHPROBE	FALSE	
2	TriggerInput0	MC_TRIGGER_REF	FALSE	
3	TouchProbe_En	BOOL	FALSE	
4	TouchProbe_Window	BOOL	FALSE	
5	TouchProbe_FirPos	LREAL	0	
6	TouchProbe_LastPos	LREAL	0	
7	TouchProbe_Done	BOOL	FALSE	
8	TouchProbe_Busy	BOOL	FALSE	
9	TouchProbe_Abort	BOOL	FALSE	
10	TouchProbe_Err	BOOL	FALSE	
11	TouchProbe_ErrID	SMC_ERROR	MC_ALM.	ID
12	TouchProbe_Pos	LREAL	0	
13	TouchProbe_PosNeg	LREAL	0	
14	R_TRIG1	R_TRIG	FALSE	
15	PositionRec	ARRAY[0..10] OF LREAL		
16	i	INT	0	

LD

1 SMC_AxisInit



2 TriggerInput0 MC_TouchProbe



ST

1 MC_AxisInit

```
MC_ReadAxisInfo1(
Enable := ReadAxisInfo_En,
Axis := Axis0,
Valid=>ReadAxisInfo_Valid,
Busy=>ReadAxisInfo_Busy,
Error=>ReadAxisInfo_Error,
ErrorID=>ReadAxisInfo_ErrorID,
HomeAbsSwitch=>ReadAxisInfo_HomeAbsSwitch,
LimitSwitchPos=>ReadAxisInfo_LimitSwitchPos,
LimitSwitchNeg=>ReadAxisInfo_LimitSwitchNeg,
Simulation=>ReadAxisInfo_Simulation,
CommunicationReady=>ReadAxisInfo_ComRdy,
ReadyForPowerOn=>ReadAxisInfo_RdyPowerON,
PowerOn=>ReadAxisInfo_PowerON,
IsHomed=>ReadAxisInfo_Homed,
AxisWarning=>ReadAxisInfo_Warning,
AxisFault=>ReadAxisInfo_Fault);
```

2

```
TriggerInput0.LatchID := 0; (*                    1*)

TriggerInput0.LatchMode := 0;(*                    *)
```

```
TriggerInput0.DriveInput := 0;(*                        DI        *)

TriggerInput0.TriggerEdge := 0;(*                              *)

TriggerInput0.ContinuousTrigger := TRUE;(*              *)
```

3 MC_TouchProbe

```
(*MC_TouchProbe       *)

MC_TouchProbe0(
    Execute := TouchProbe_En,
    WindowOnly := TouchProbe_Window,
    FirstPosition := TouchProbe_FirPos,
    LastPosition := TouchProbe_LastPos,
    Axis := Axis0,
    TriggerInput := TriggerInput0,
    Done=>TouchProbe_Done,
    Busy=>TouchProbe_Busy,
    CommandAborted=>TouchProbe_Abort,
    Error=>TouchProbe_Err,
    ErrorID=>TouchProbe_ErrID,
    RecordedPosition=>TouchProbe_Pos,
    RecordedPositionNeg=>TouchProbe_PosNeg);
```

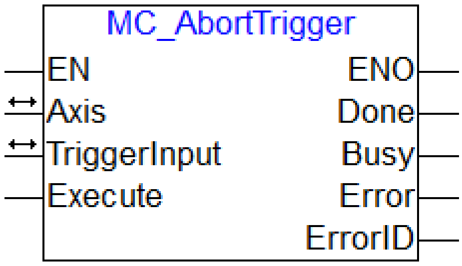
4

```
R_TRIG1(CLK:=TouchProbe_Done);
(*              *)

IF R_TRIG1.Q = TRUE THEN
    PositionRec[i] := TouchProbe_Pos;
    i:=i+1;
END_IF;
```

5.26 MC_AbortTrigger

MC_TouchProbe



5.26.1

IN_OUT	Axis			-	-	AXIS_REF
	TriggerInput			-	MC_TRIGGER_REF	MC_TRIGGER_REF
INPUT	Execute			-	TRUE/FALSE	BOOL
OUTPUT	Done			FALSE	TRUE/FALSE	BOOL
	Busy			FALSE	TRUE/FALSE	BOOL
	Error			FALSE	TRUE/FALSE	BOOL
	ErrorID			0	-	SMC_ERROR

5.26.2

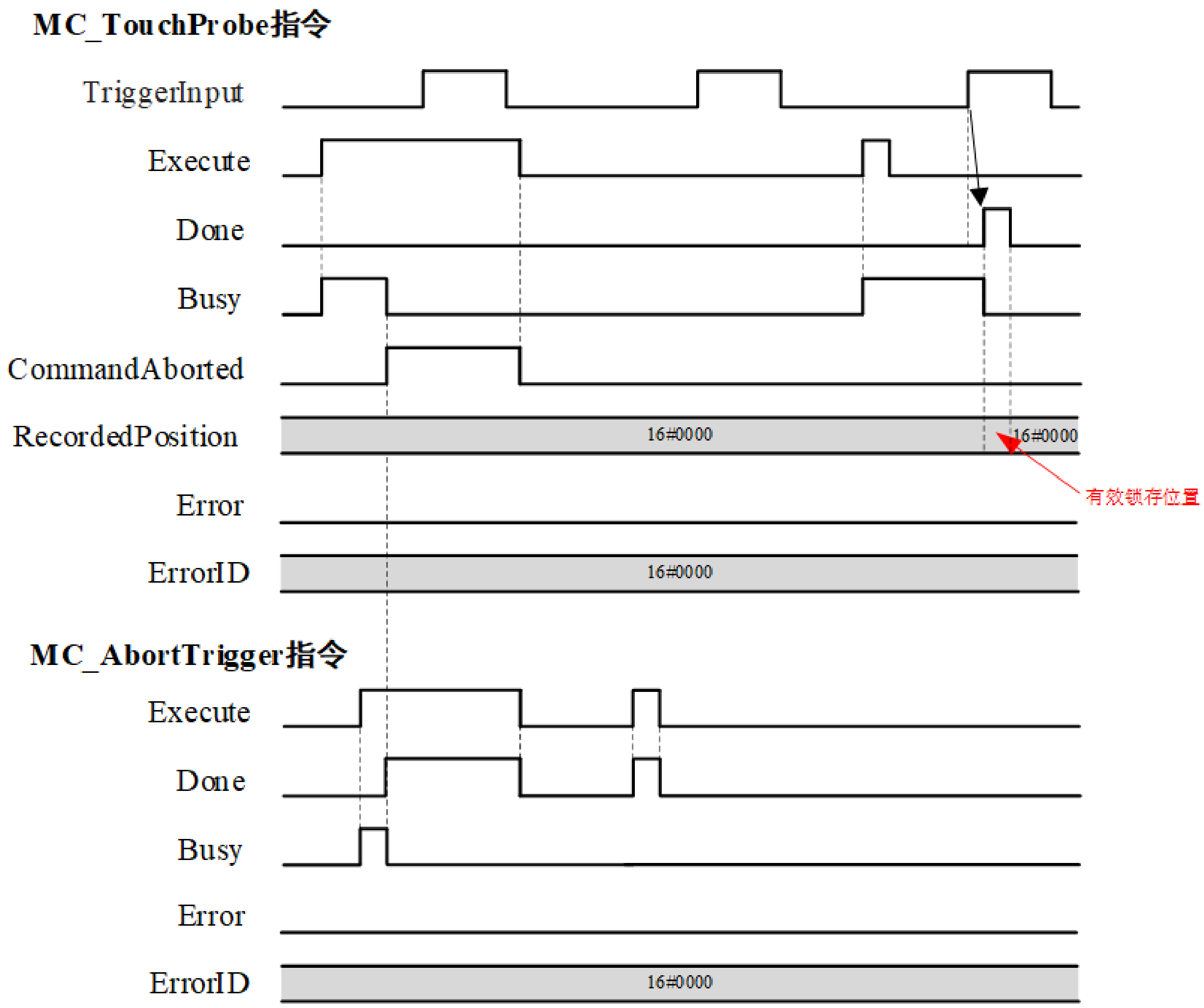
MC_TouchProbe EtherCAT

Axis( ) LatchID( )

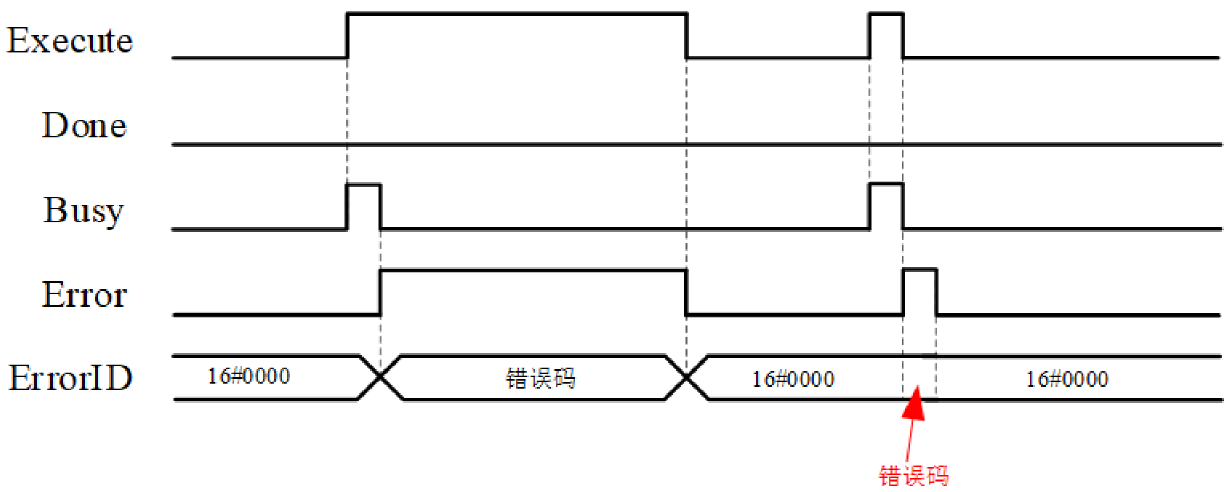
MC_TouchProbe MC_Home HMC_GantryInit

1 1 DI MC_AbortTrigger

1



2 MC_AbortTrigger



5.26.3

MC_AbortTrigger MC_TouchProbe

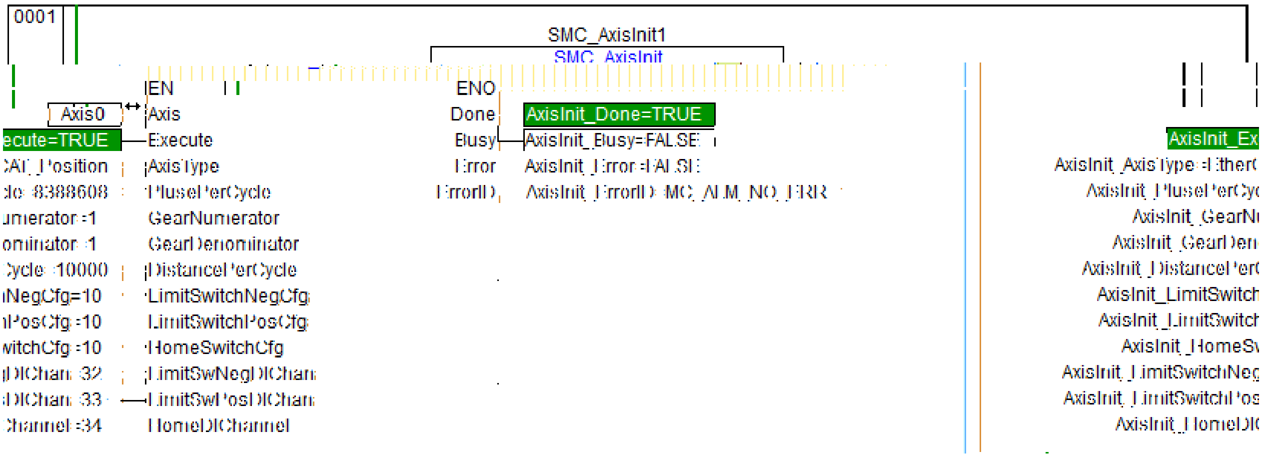
MC_TouchProbe CommandAborted TRUE MC_AbortTrigger Done TRUE

MC_TouchProbe MC_TouchProbe

1	MC_AbortTrigger0	MC_ABORTTRIGGER		
2	AbortTrigger_En	BOOL	FALSE	
3	AbortTrigger_Done	BOOL	FALSE	
4	AbortTrigger_Busy	BOOL	FALSE	
5	AbortTrigger_Error	BOOL	FALSE	
6	AbortTrigger_Error	MC_ERROR	MC_ALM_NO_ERROR	ID

LD

1 SMC_AxisInit

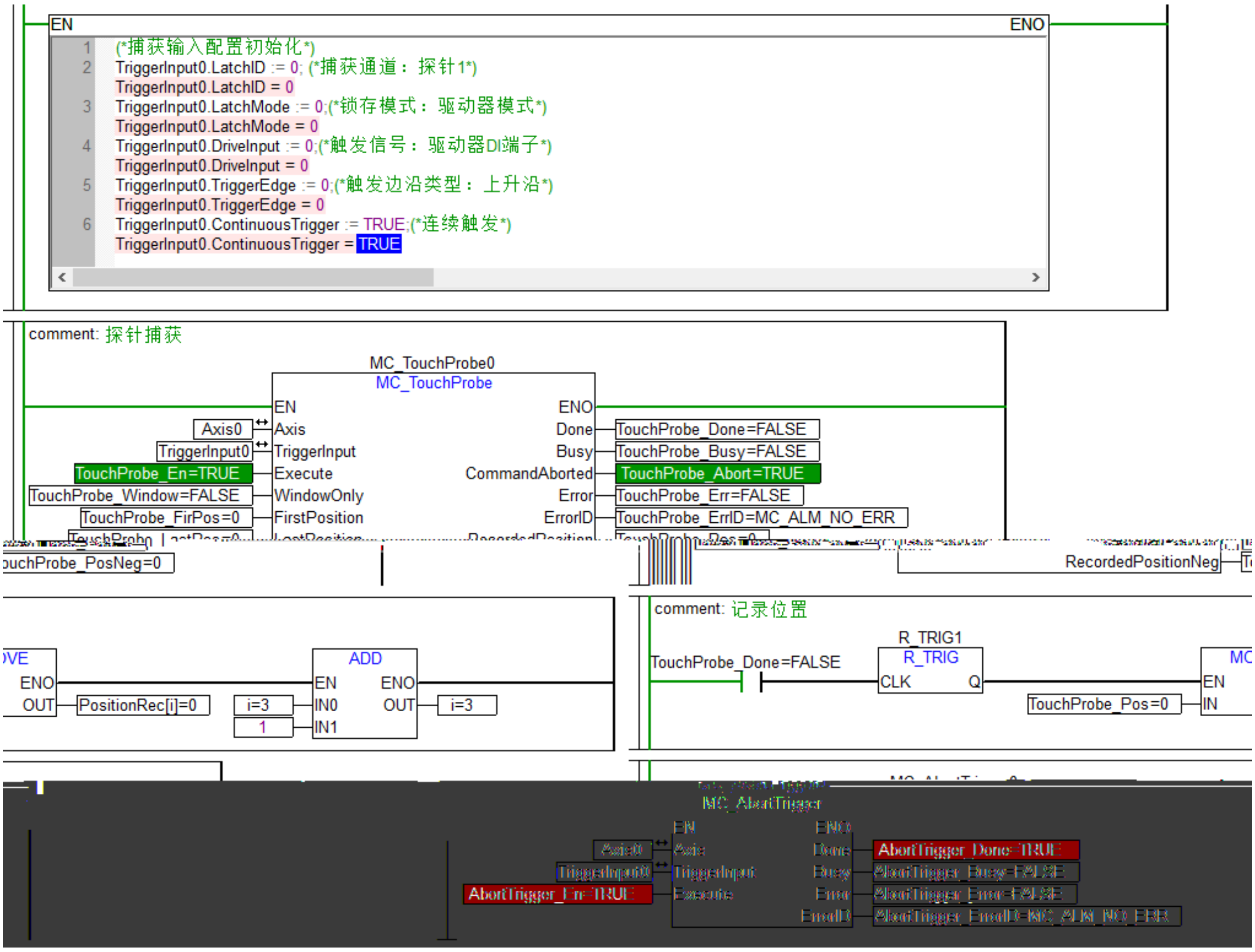


2 TriggerInput0 MC_TouchProbe

MC_TouchProbe Busy MC_AbortTrigger

MC_AbortTrigger Done TRUE MC_TouchProbe CommandAborted

TRUE



ST

1 MC_AxisInit

```
MC_ReadAxisInfo1(
Enable := ReadAxisInfo_En,
Axis := Axis0,
Valid=>ReadAxisInfo_Valid,
Busy=>ReadAxisInfo_Busy,
Error=>ReadAxisInfo_Error,
ErrorID=>ReadAxisInfo_ErrorID,
HomeAbsSwitch=>ReadAxisInfo_HomeAbsSwitch,
LimitSwitchPos=>ReadAxisInfo_LimitSwitchPos,
LimitSwitchNeg=>ReadAxisInfo_LimitSwitchNeg,
Simulation=>ReadAxisInfo_Simulation,
CommunicationReady=>ReadAxisInfo_ComRdy,
ReadyForPowerOn=>ReadAxisInfo_RdyPowerON,
PowerOn=>ReadAxisInfo_PowerON,
IsHomed=>ReadAxisInfo_Homed,
AxisWarning=>ReadAxisInfo_Warning,
AxisFault=>ReadAxisInfo_Fault);
```

2

```
TriggerInput0.LatchID := 0; (*                    1*)
TriggerInput0.LatchMode := 0;(*                          *)
TriggerInput0.DriveInput := 0;(*                       DI        *)
TriggerInput0.TriggerEdge := 0;(*                          *)
TriggerInput0.ContinuousTrigger := TRUE;(*            *)
```

3 MC_TouchProbe

```
(*MC_TouchProbe       *)
MC_TouchProbe0(
   Execute := TouchProbe_En,
   WindowOnly := TouchProbe_Window,
   FirstPosition := TouchProbe_FirPos,
   LastPosition := TouchProbe_LastPos,
   Axis := Axis0,
   TriggerInput := TriggerInput0,
   Done=>TouchProbe_Done,
   Busy=>TouchProbe_Busy,
   CommandAborted=>TouchProbe_Abort,
   Error=>TouchProbe_Err,
   ErrorID=>TouchProbe_ErrID,
   RecordedPosition=>TouchProbe_Pos,
   RecordedPositionNeg=>TouchProbe_PosNeg);
```

4

```
R_TRIG1(CLK:=TouchProbe_Done);
(*             *)
IF R_TRIG1.Q = TRUE THEN
   PositionRec[i] := TouchProbe_Pos;
   i:=i+1;
END_IF;
```

5 MC_TouchProbe Busy MC_AbortTrigger

MC_AbortTrigger Done TRUE MC_TouchProbe CommandAborted

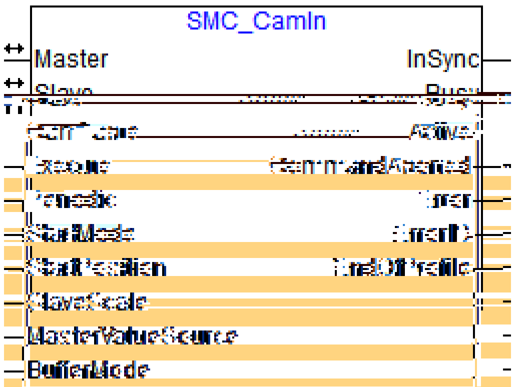
TRUE

```
(*MC_AbortTrigger       *)
MC_AbortTrigger0(
   Execute := AbortTrigger_En,
   Axis := Axis0,
```

```
TriggerInput := TriggerInput0,
Done=>AbortTrigger_Done,
Busy=>AbortTrigger_Busy,
Error=>AbortTrigger_Error,
ErrorID=>AbortTrigger_ErrorID);
```

6

6.1 SMC_CamIn



6.1.1

						• ;Ü

	StartPosition	DI DI I %IX100.1 I bit 100×8 + 1		0	/ /0; DI 0 ~ I byte * 8 - 1	LREAL
	SlaveScale			1		LREAL
	MasterValueSource	1:		1	1: mcActualValue	MC_SOURCE

3 3

- DI

= +

6.1.2

1

2 Execute

3 SMC_CAM_TABLE

4

5 MasterValueSource 1

6

EtherCAT

EtherCAT

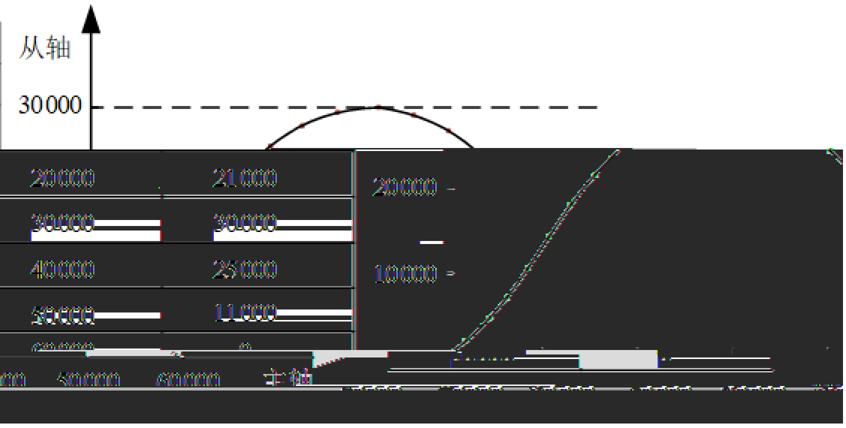
EtherCAT

1

2

凸轮表

主轴	从轴
0	0
10000	12000
20000	21000
30000	30000
40000	25000
50000	11000
60000	0



4

5

6

StandStill

0 10 11 DI

Mpos

StartPosition InSync( ) TRUE

[0 360°) StartPosition 100

StartPosition

凸轮位置数组

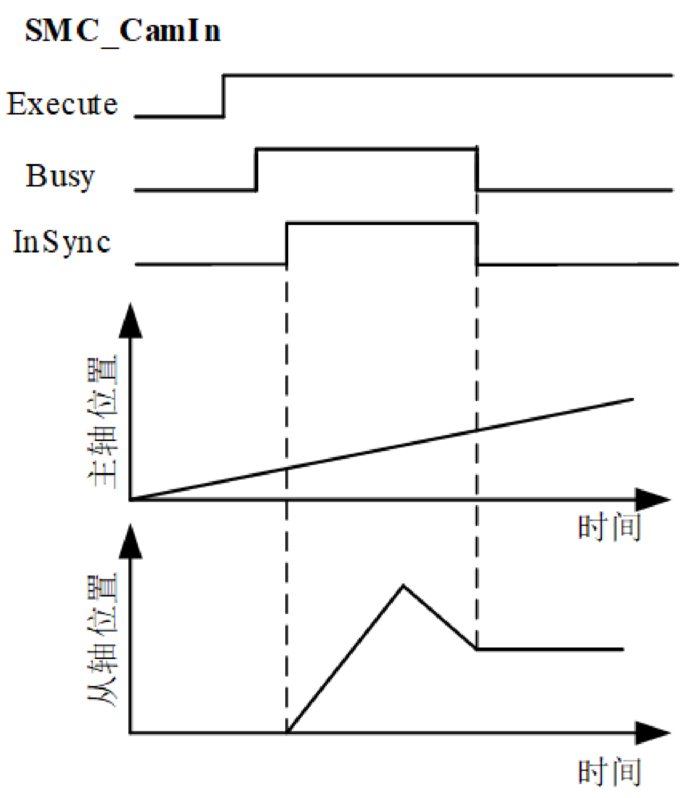
主轴	从轴
0	0
60	1000
120	1500
180	2000
240	1500
300	1000
360	0

StartPosition=100

主轴、从轴的绝对位置

主轴	从轴
100	0+从轴在凸轮位置数组起点处的绝对位置
160	1000+从轴在凸轮位置数组起点处的绝对位置
220	1500+从轴在凸轮位置数组起点处的绝对位置
280	2000+从轴在凸轮位置数组起点处的绝对位置
340	1500+从轴在凸轮位置数组起点处的绝对位置
40	1000+从轴在凸轮位置数组起点处的绝对位置
100	0+从轴在凸轮位置数组起点处的绝对位置

StartPosition



DI

DI 1

DI 1 Mpos StartPosition DI I

%IX100.1 I bit 100×8 + 1 StartPosition=801

DI StartPosition I

7

Periodic

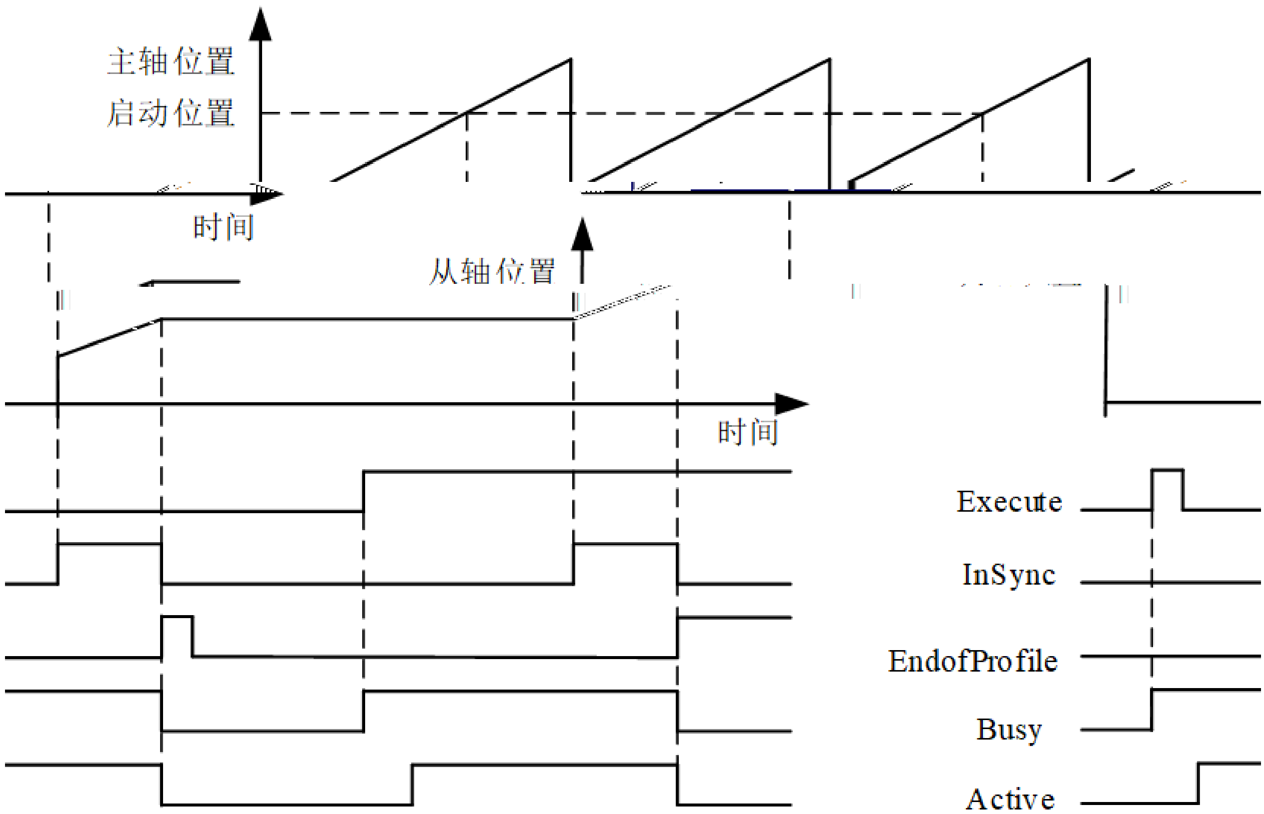
Periodic=FALSE

Periodic=TRUE

InSync

TRUE EndofProfile TRUE

Busy Active FALSE Execute

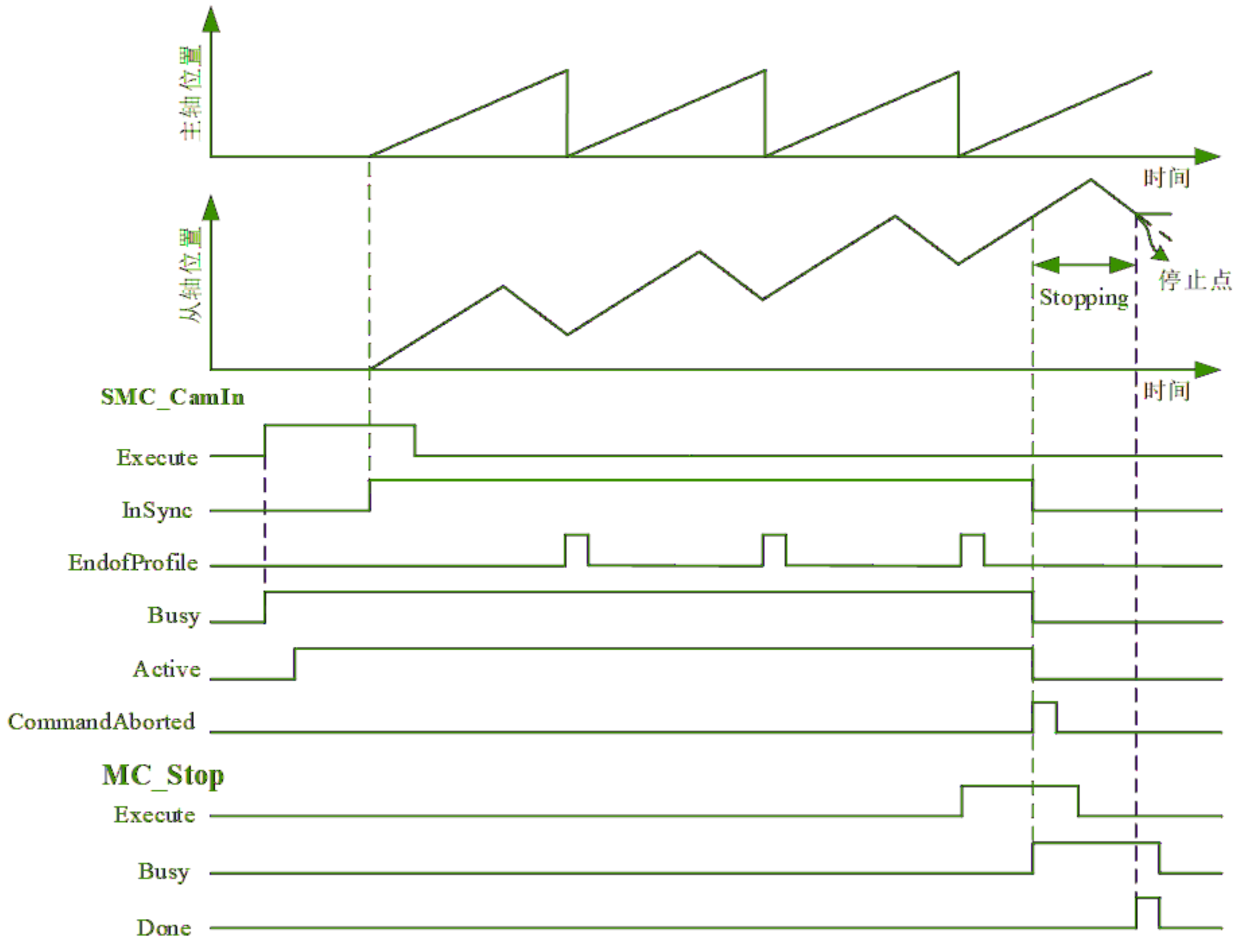


EndofProfile

MC_Stop CommandAborted

TRUE

MC_Stop



8

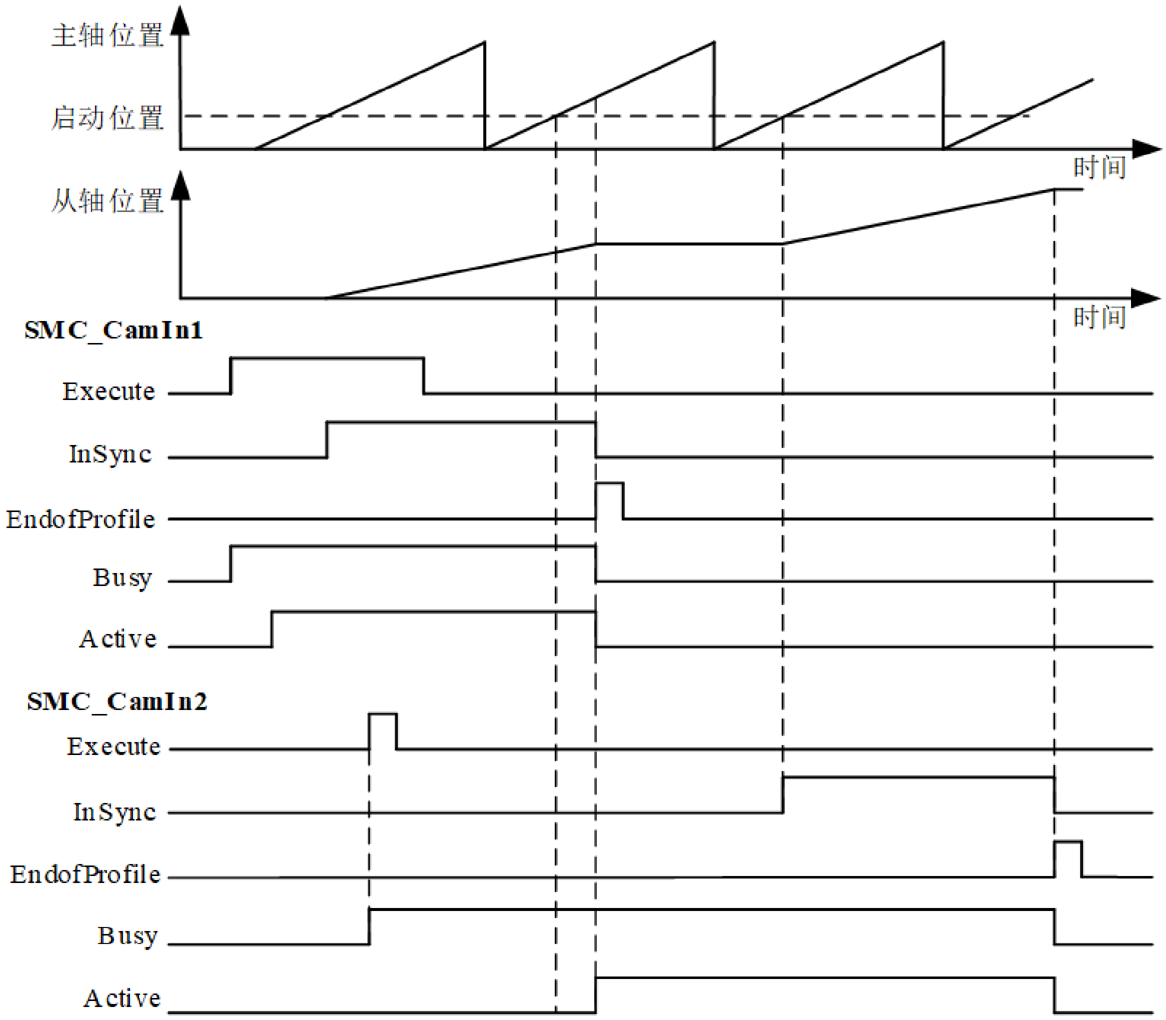
a BufferMode 0 Aborting

b BufferMode 1 Buffered

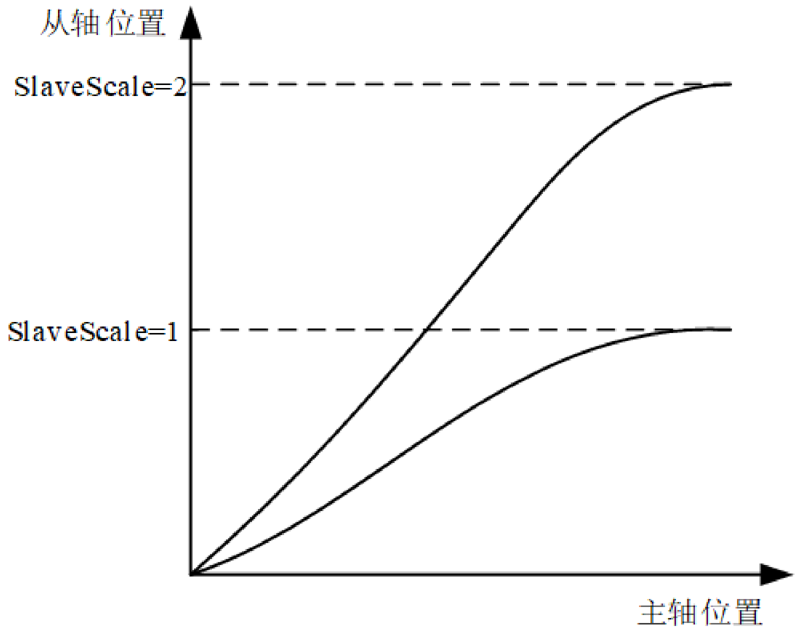
SMC_CamIn1 SMC_CamIn2

SMC_CamIn1 EndOfProfile TRUE SMC_CamIn2 Active

TRUE StartPosition( ) InSync( ) TRUE


主轴位置
启动位置
时间
从轴位置
时间
SMC_CamIn1
Execute
InSync
EndofProfile
Busy
Active
SMC_CamIn2
Execute
InSync
EndofProfile
Busy
Active

9



11 EndofProfile

EndofProfile TRUE

Execute FALSE EndofProfile FALSE

EndofProfile TRUE

EndofProfile FALSE EndofProfile

TRUE EndofProfile MC_Stop

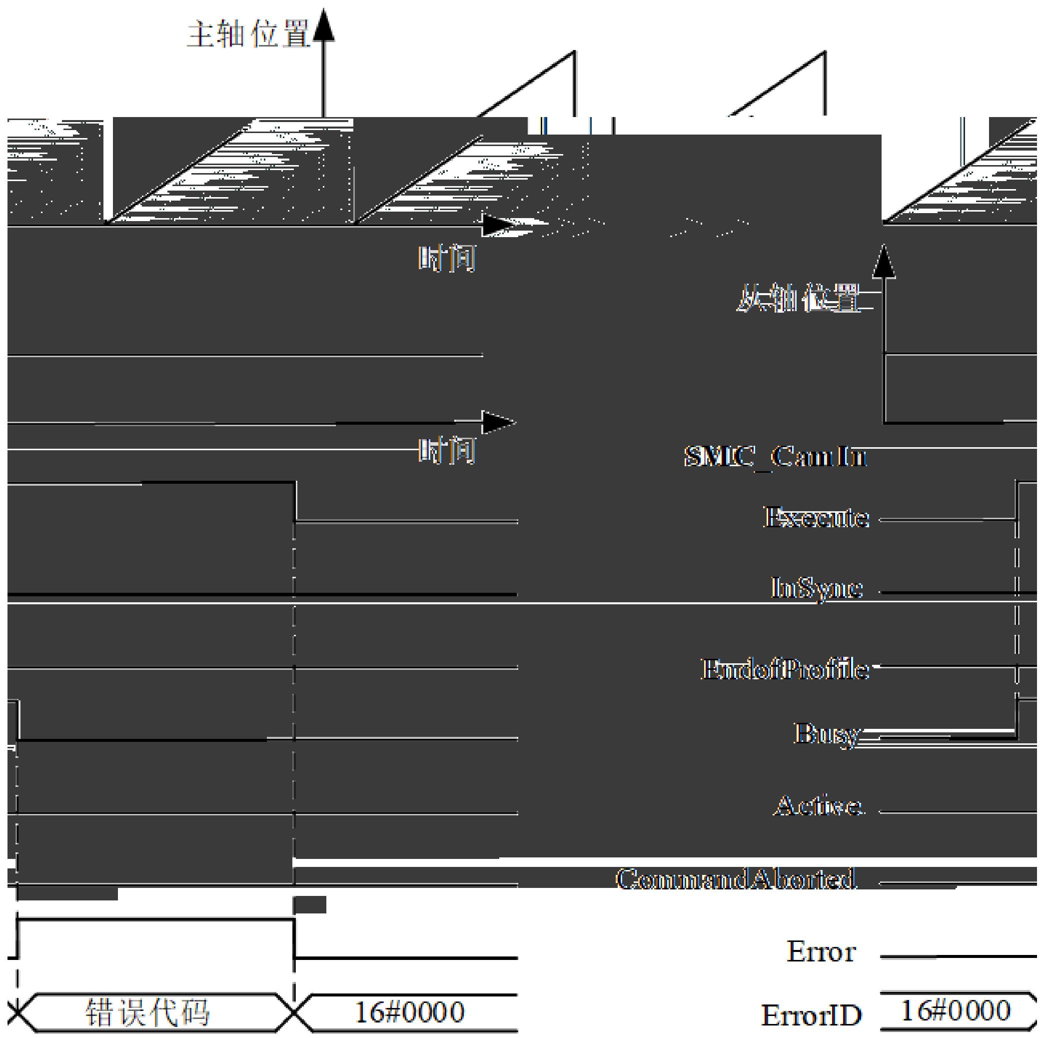
EndofProfile

12

Error True ErrorID

Execute

异常情况时序图



6.1.3

MasterAxis SlaveAxis AXIS_REF

MC_MoveVelocity MC_CamIn MC_Stop

Periodic( ) TRUE

1

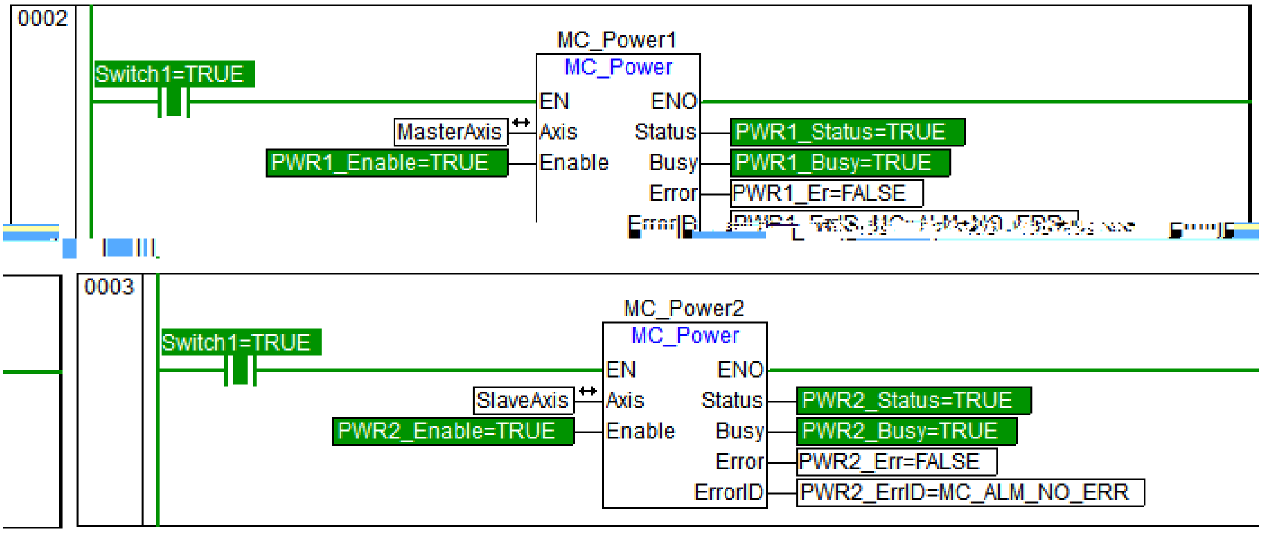
MasterAxis	AXIS_REF	-	
SlaveAxis	AXIS_REF	-	
Power1_Status	BOOL	FALSE	TRUE
Power2_Status	BOOL	FALSE	TRUE
MV_Exe	BOOL	FALSE	
MV_InVel	BOOL	FALSE	TRUE
CamTable	SMC_CAM_TABLE	-	
ArrTablePos	ARRAY OF LREAL	-	
PointerToPos	POINTER TO ARRAY OF LREAL	-	Pos[2][PosNum] PosNum
Cam_InSync	BOOL	FALSE	TRUE
Cam_EndProFile	BOOL	FALSE	TRUE
Switch1	BOOL	FALSE	Switch TRUE

2 MasterAxis AxisID 0 SlaveAxis AxisID 1

50 EtherCAT_Position SMC_AxisInit

SMC_AxisInit

3 0002 Switch1 TRUE



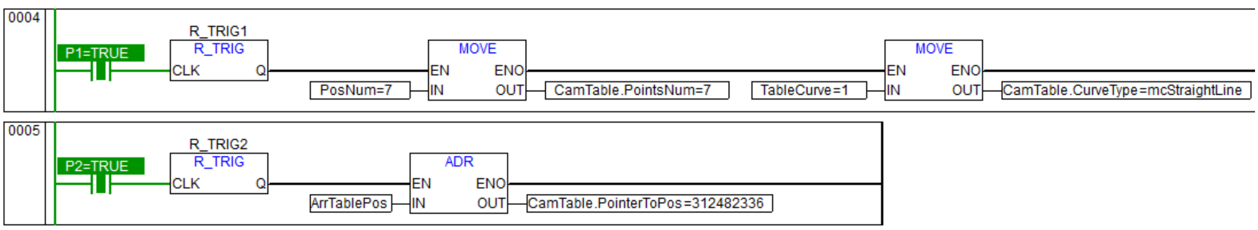
4 SMC_CAM_TABLE 0004

0005 CamTable

ArrTablePos [0][PosNum] ={0,60,120,180,240,300,360};

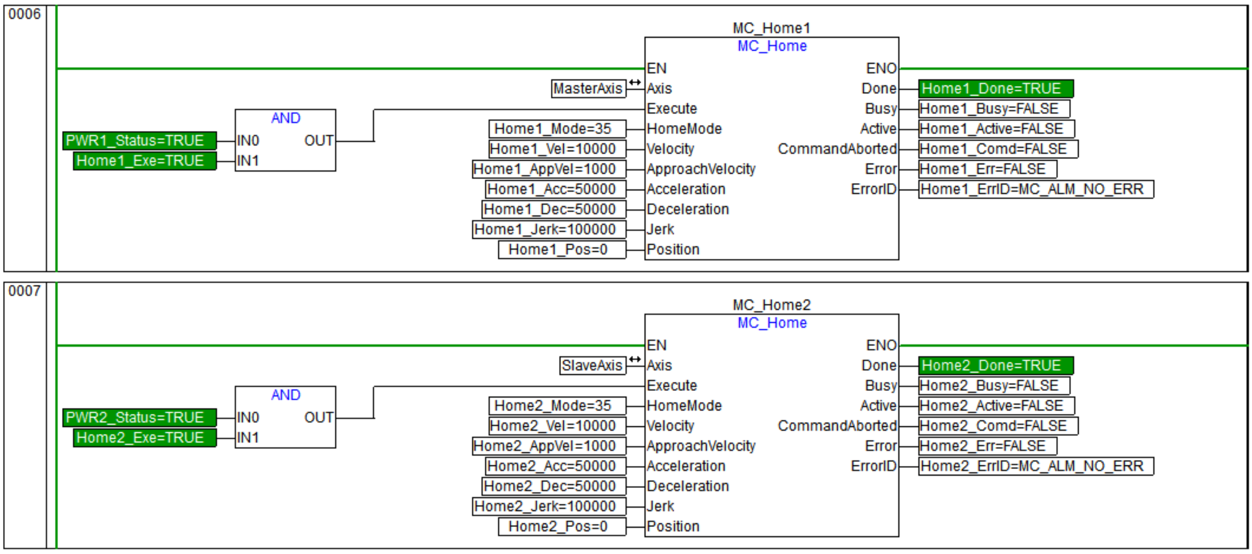
ArrTablePos [1][PosNum] ={0,1000, 2000,3000,2500,2000,1500};

PosNum=7;

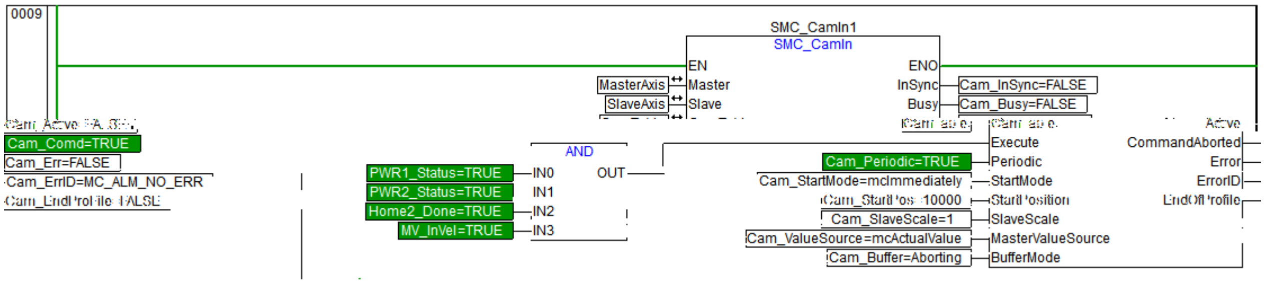


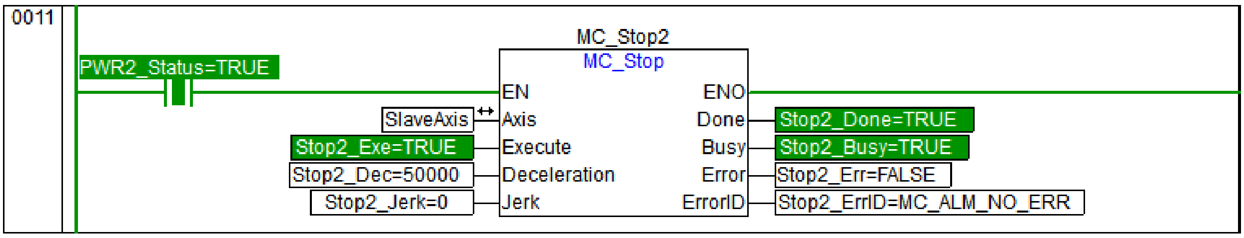
5 0006

0007 MC_Home



6 0008 MC_MoveVelocity( ) < פ « ⅞]


0009
SMC_CamIn1
SMC_CamIn
EN
ENO
MasterAxis
Master
SlaveAxis
Slave
InSync
Cam_InSync=FALSE
Busy
Cam_Busy=FALSE
Cam_Comd=TRUE
Cam_Err=FALSE
Cam_ErrID=MC_ALM_NO_ERR
AND
PWR1_Status=TRUE
PWR2_Status=TRUE
Home2_Done=TRUE
MV_InVel=TRUE
IN0
IN1
IN2
IN3
OUT
Cam_Periodic=TRUE
Cam_StartMode=mcImmediately
Cam_SlaveScale=1
Cam_ValueSource=mcActualValue
Cam_Buffer=Aborting
Execute
Periodic
StartMode
SlaveScale
MasterValueSource
BufferMode
CommandAborted
Error
ErrorID


0011
MC_Stop2
MC_Stop
PWR2_Status=TRUE
EN
ENO
SlaveAxis
Axis
Done
Stop2_Done=TRUE
Stop2_Exe=TRUE
Execute
Busy
Stop2_Busy=TRUE
Stop2_Dec=50000
Deceleration
Error
Stop2_Err=FALSE
Stop2_Jerk=0
Jerk
ErrorID
Stop2_ErrID=MC_ALM_NO_ERR

ST

10000

10000

1

MasterAxis	AXIS_REF	-	
SlaveAxis	AXIS_REF	-	
Power1_Status	BOOL	FALSE	TRUE
Power2_Status	BOOL	FALSE	TRUE
MV_Exe	BOOL	FALSE	
MV_InVel	BOOL	FALSE	TRUE
CamTable	SMC_CAM_TABLE	-	
ArrTablePos	ARRAY OF LREAL	-	
PointerToPos	POINTER TO ARRAY OF LREAL	-	Pos[2][PosNum] PosNum
Cam_InSync	BOOL	FALSE	TRUE
Cam_EndProFile	BOOL	FALSE	TRUE
Switch1	BOOL	FALSE	Switch TRUE
InitFlag	BOOL	FALSE	TRUE

2

MasterAxis AxisID 0 SlaveAxis AxisID 1 SMC_AxisInit

50 EtherCAT_Position SMC_AxisInit

3

```
   Status=> PWR1_Status,
   Busy=> PWR1_Busy,
Error=> PWR1_Err,
ErrorID=> PWR1_ErrID);
MC_Power2(
   Enable := PWR2_Enable,
   Axis := SlaveAxis,
   Status=> PWR2_Status,
   Busy=> PWR2_Busy,
   Error=> PWR2_Err,
   ErrorID=> PWR2_ErrID);
END_IF
```

4 SMC_CAM_TABLE

CamTable

```
IF FALSE=InitFlag THEN
CamTable.PointsNum := PointsNum ;          (*          *)
ArrTablePos[0,0]:=0; (*          *)
ArrTablePos[0,1]:=10000; (*          *)
…
…
ArrTablePos[0, PointsNum-1]:=60000; (*          *)
ArrTablePos[1,0]:=0; (*          *)
ArrTablePos[1,1]:=5000; (*          *)
…
…
ArrTablePos[1, PointsNum-1]:=10000; (*          *)
   CamTable.PointsNum := CurveType ;          (*          *)
CamTable.PointerToPos := ADR(ArrTablePos);   (*          *)
InitFlag:=TRUE;
END_IF
```

5 MC_Home

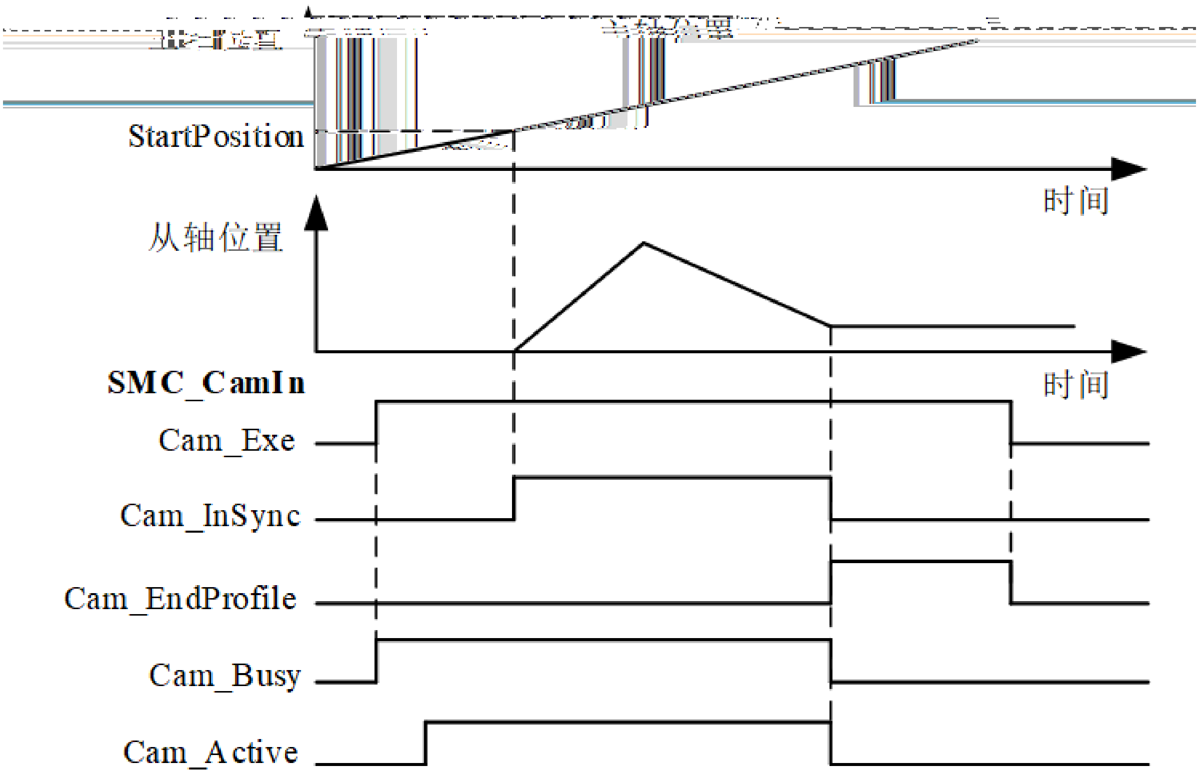
```
IF  PWR1_Status THEN
   MC_Home1(
   Execute := Home1_Exe,
   HomeMode := Home1_Mode,
   Velocity := Home1_Vel,
```

```
ApproachVelocity := Home1_AppVel,
Acceleration := Home1_Acc,
Deceleration := Home1_Dec,
Jerk := Home1_Jerk,
Position := Home1_Pos,
Axis := MasterAxis,
Done=> Home1_Done,
Busy=> Home1_Busy,
Active=> Home1_Active,
CommandAborted=> Home1_Comd,
Error=> Home1_Err,
ErrorID=> Home1_ErrID);
MC_Home2(
Execute := Home2_Exe,
HomeMode := Home2_Mode,
Velocity := Home2_Vel,
ApproachVelocity := Home2_AppVel,
Acceleration := Home2_Acc,
```

```
    Active=> Cam_Act,
    CommandAborted=> Cam_Comd,
    Error=> Cam_Err,
    ErrorID=> Cam_ErrID,
    EndOfProfile=> Cam_EndPro);
END_IF
```

7

```
IF  PWR1_Status AND Home1_Done THEN
    MC_MoveVelocity1(
    Execute := MV_Exe,
    Velocity := MV_Vel,
    Acceleration := MV_Acc,
    Deceleration := MV_Dcc,
    Jerk := MV_Jerk,
    Direction := MV_Dir,
    BufferMode := MV_Buffer,
    Axis := Axis0,
    InVelocity=> MV_InVel,
    Busy=> MV_Busy,
    Active=> MV_Active,
    CommandAborted=> MV_Comm,
    Error=> MV_Err,
    ErrorID=> MV_Error);
END_IF
```



6.1.4

3

DI StartPosition StartPosition I

Mpos

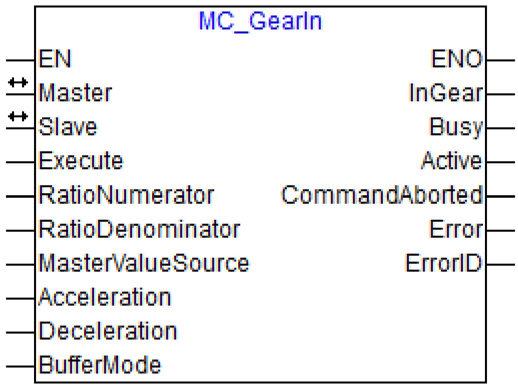
Mpos

Mpos Dpos

Mpos

Mpos

6.2 MC_GearIn



6.2.1

IN_OUT	Master			-	-	AXIS_REF
	Slave			-	-	AXIS_REF
	Execute					
INPUT						

		1				
OUTPUT	InGear			FALSE	TRUE/FALSE	BOOL
	Busy			FALSE	TRUE/FALSE	BOOL
	Active	TRUE		FALSE	TRUE/FALSE	BOOL
	CommandAborted			FALSE	TRUE/FALSE	BOOL
	Error			FALSE	TRUE/FALSE	BOOL
	ErrorID			0	-	SMC_ERROR

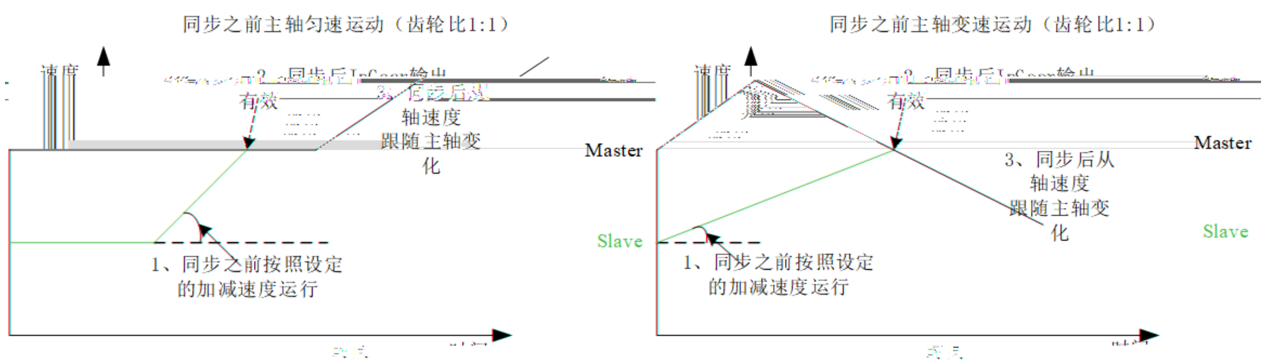
6.2.2

1

2

3

4

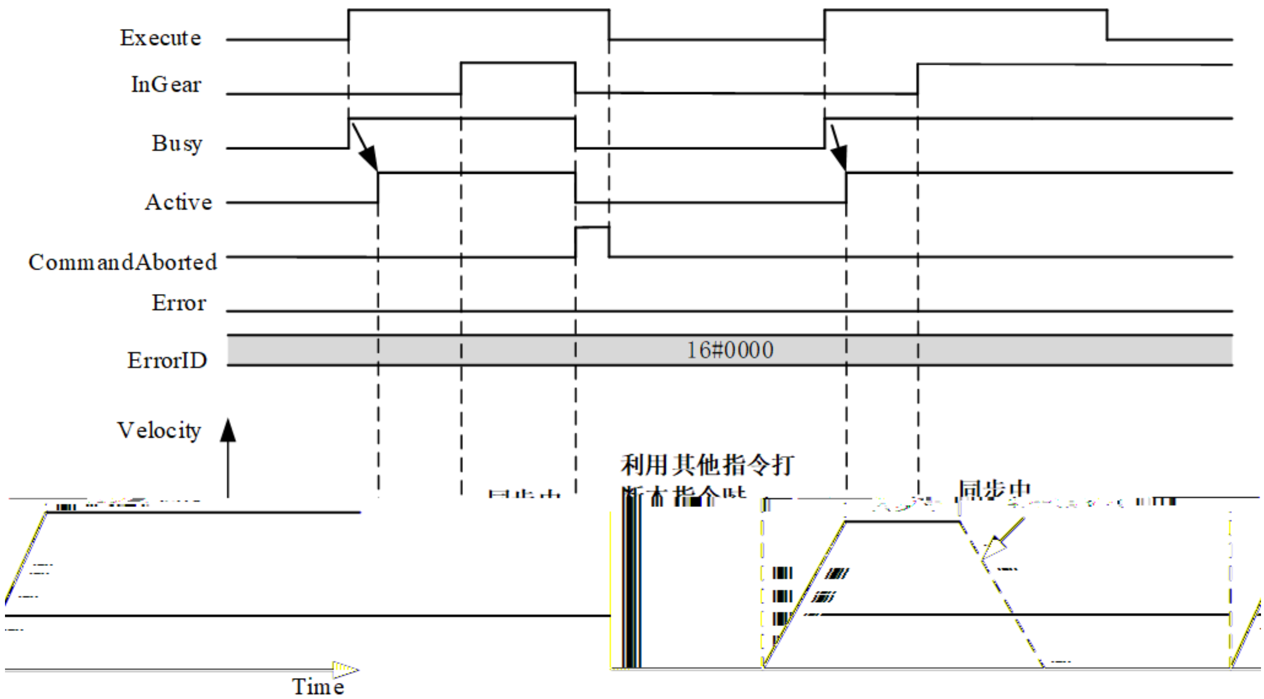


5 BufferMode

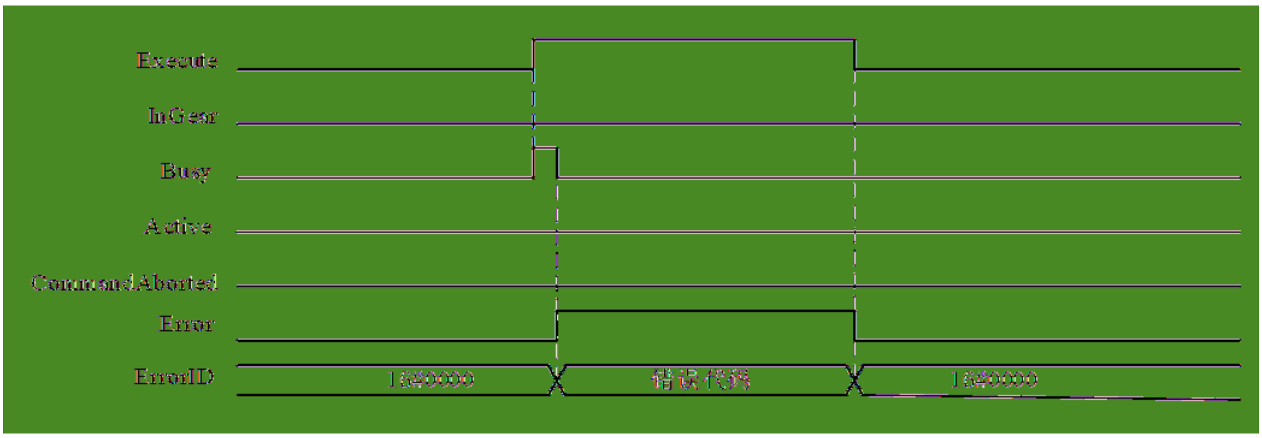
InGear

6 MC_Stop

1



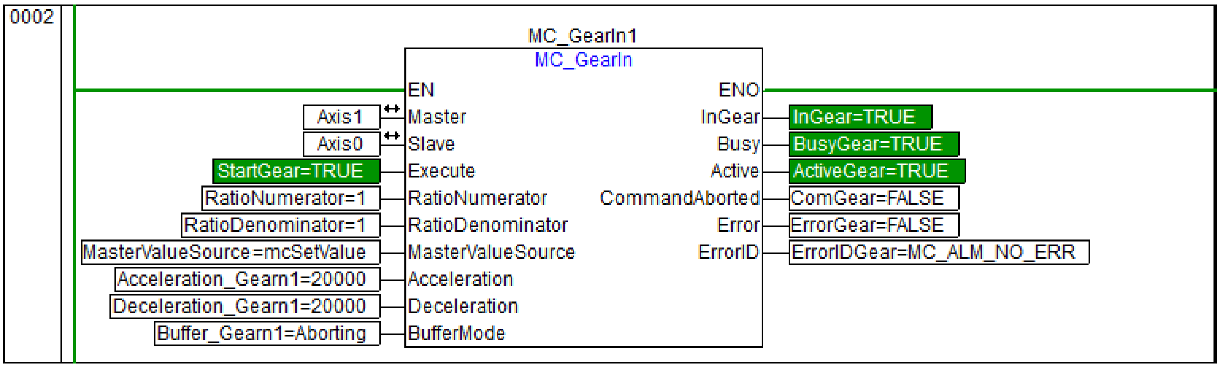
2



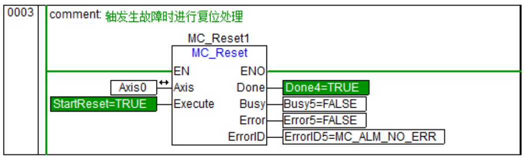
6.2.3

MC_GearIn

Axis0	AXIS_REF		0
StartInit	BOOL	FALSE	TRUE
DoneInit	BOOL	FALSE	TRUE
PowerEnable	BOOL	FALSE	[redacted]



3 MC_Reset



ST

1

```
SMC_AxisInit1(
Execute := StartInit,
AxisType := 50,
PlusePerCycle := 10000,
GearNumerator := 1,
GearDenominator := 1,
DistancePerCycle := 10000,
LimitSwitchNegCfg := -1,
LimitSwitchPosCfg := -1,
HomeSwitchCfg := -1,
LimitSwNegDIChan := 0,
LimitSwPosDIChan := 1,
HomeDIChannel := 2,
Axis := Axis0,
Done=> DoneInit,
Busy=>Busy1,
Error=>Error1,
ErrorID=>ErrorID1);
```

2

```
IF DoneInit THEN
   MC_Power1(
   Enable := PowerEnable,
   Axis := Axis0,
   Status=> PowerStatus,
```

```
   Busy=>Busy2,
   Error=>Error2,
   ErrorID=>ErrorID2);
END_IF
```

3

```
MC_GearIn1(
Execute := StartGear,
RatioNumerator := RatioNumerator,
RatioDenominator := RatioDenominator,
MasterValueSource := MasterValueSource,
Acceleration := Acceleration_Gearn1,
Deceleration := Deceleration_Gearn1,
BufferMode := Buffer_Gearn1,
Master := Axis1,
Slave := Axis0,
InGear=>InGear,
Busy=>BusyGear,
Active=>ActiveGear,
CommandAborted=>ComGear,
Error=>ErrorGear,
ErrorID=>ErrorIDGear);
```

4

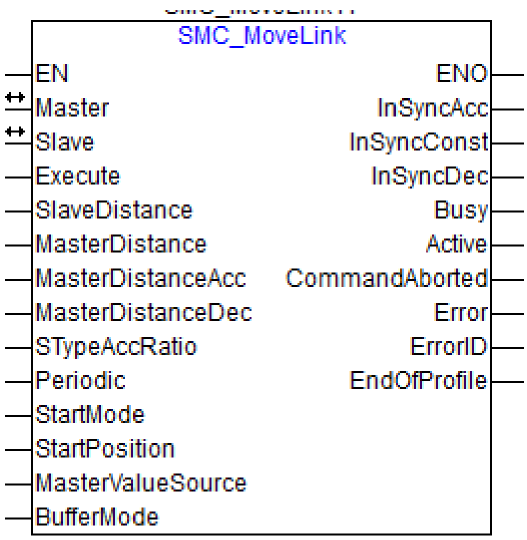
```
MC_Reset1(
Execute := StartReset,
Axis := Axis0,
Done=>Done4,
Busy=>Busy5,
Error=>Error5,
ErrorID=>ErrorID5);
```

6.2.4

0

6.2.5

6.3 SMC_MoveLink /



6.3.1

IN_OUT	Master			-	-	AXIS_REF
	Slave			-	-	AXIS_REF
INPUT	Execute			-	TRUE/FALSE	BOOL
	SlaveDistance			-	/ /0	LREAL
	MasterDistance			-		LREAL
	MasterDistanceAcc			-	/0	LREAL
	MasterDistanceDec			-	/0	LREAL
	STypeAccRatio	S 1%		0	0 ~ 100	LREAL
	Periodic	FALSE: TRUE:		FALSE	TRUE/FALSE	BOOL
	StartMode	0		0	0 mcAbsolute 10 mcImmediately	MC_CAMIN_STARTMODE

		10 11 DI			11 mcDITrigger	
	StartPosition	DI DI I %IX100.1 I bit 100×8 + 1		0	/ /0; DI 0 ~ I byte * 8 - 1	LREAL
	MasterValueSource	1:		1	1: mcActualValue	MC_SOURCE
	BufferMode	0 1		1	0: Aborting 1: Buffered	MC_BUFFER_MODE
OUTPUT	InSyncAcc			FALSE	TRUE/FALSE	BOOL
	InSyncConst			FALSE	TRUE/FALSE	BOOL
	InSyncDec			FALSE	TRUE/FALSE	BOOL
	Busy			FALSE	TRUE/FALSE	BOOL
	Active	TRUE		FALSE	TRUE/FALSE	BOOL
	CommandAborted			FALSE	TRUE/FALSE	BOOL
	Error			FALSE	TRUE/FALSE	BOOL
	ErrorID			0	-	SMC_ERROR
	EndOfProfile			FALSE	TRUE/FALSE	BOOL

6.3.2

1

2 Execute

3 MC_Stop

4 MasterDistance

Mpos0 [Mpos0 Mpos0+ MasterDistance)

MoveLink DI

= +

MC_Stop

SMC_CamIn

5 0

0 MoveLink InSyncAcc TRUE

InSyncDec TRUE InSyncConst TRUE

6

EtherCAT

EtherCAT

EtherCAT

StandStill

StartMode

0 10 11 DI

Mpos

StartPosition

StartPosition

DI

DI 1

DI 1 Mpos StartPosition DI I

%IX100.1 I bit 100×8 + 1 StartPosition=801 DI

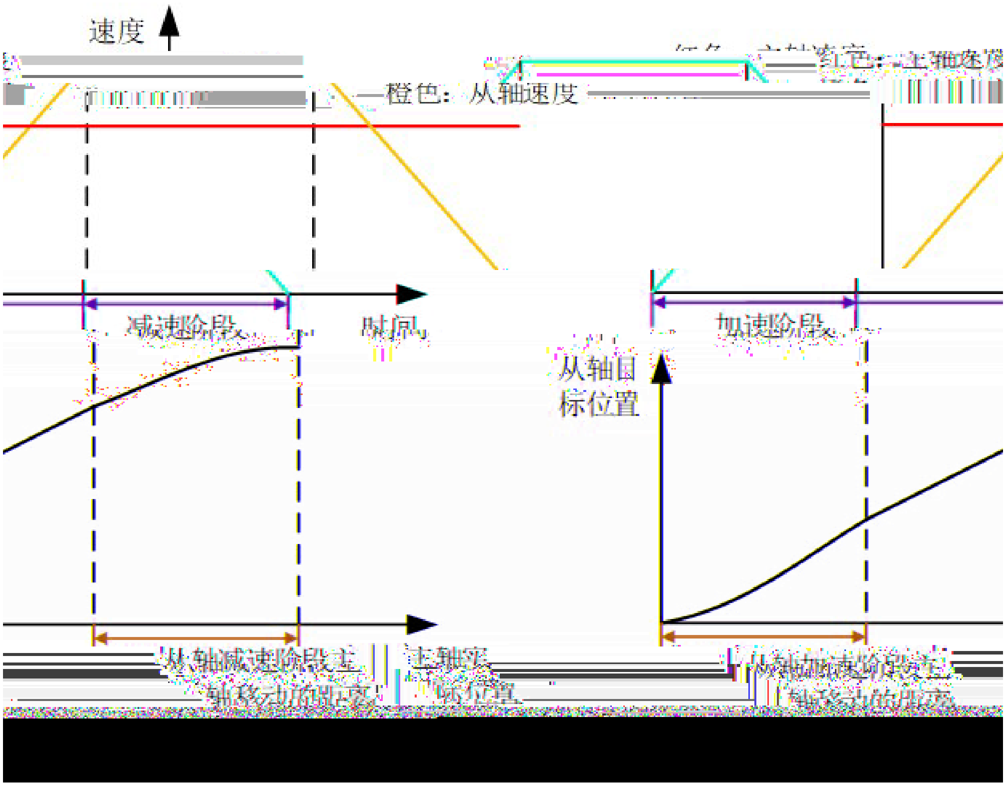
StartPosition I

S

S

(STypeAccRatio=0)

STypeAccRatio= 0



S (STypeAccRatio (0,100])

STypeAccRatio (0,100] S S

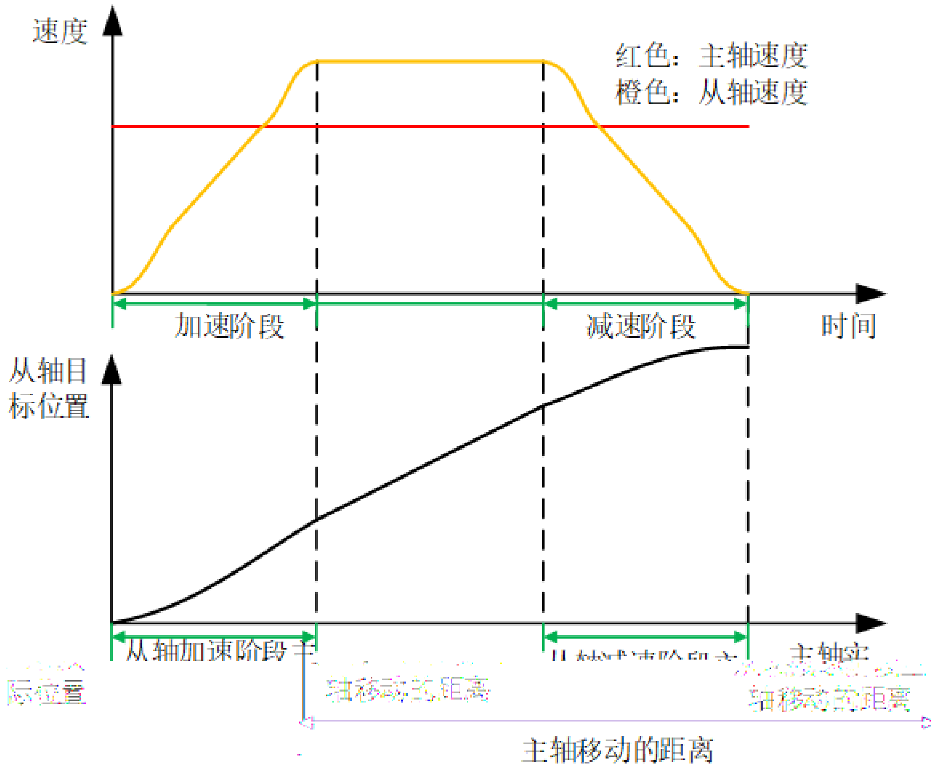
STypeAccRatio S /

STypeAccRatio/100/2

1- STypeAccRatio/100

STypeAccRatio/100/2

STypeAccRatio = 50/100



STypeAccRatio= 100 /

0.5

Periodic

Periodic

Periodic=False Periodic=TRUE

1

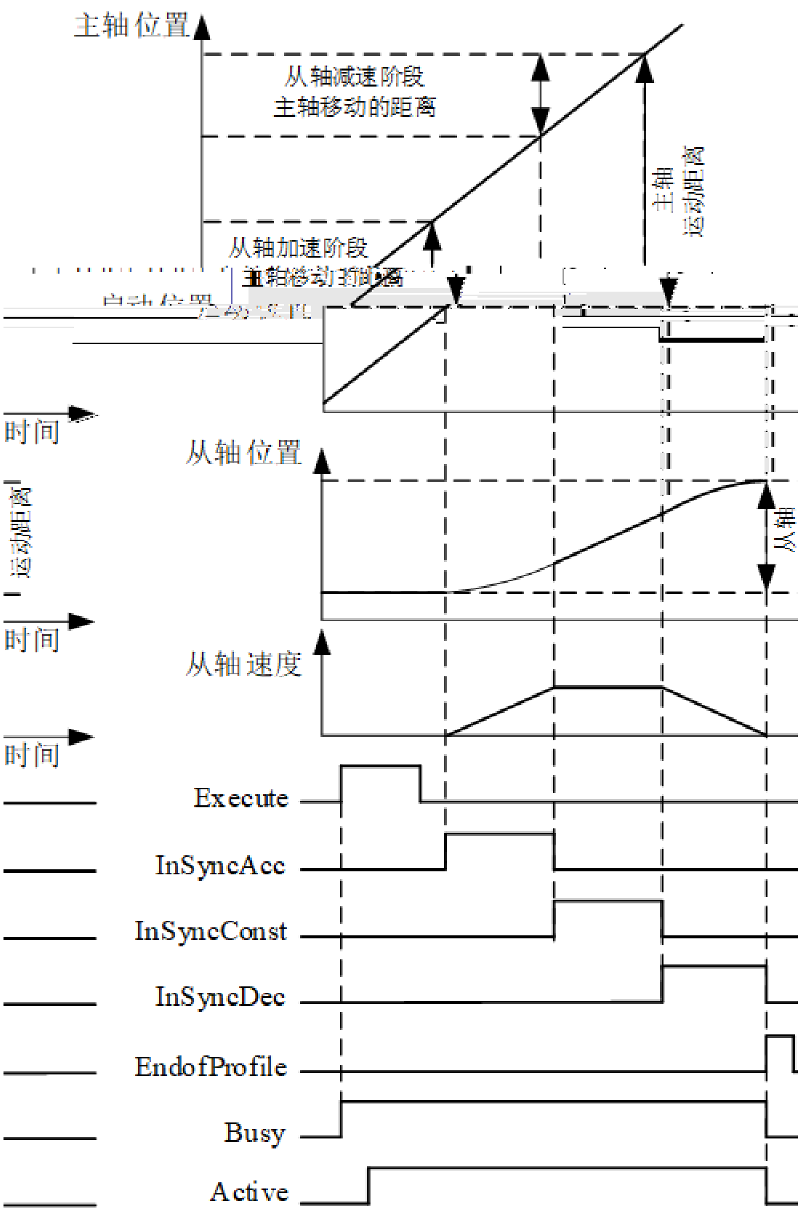
InSyncAcc TRUE

InSyncConst TRUE InSyncAcc FALSE

InSyncDec TRUE InSyncConst InSyncAcc FALSE

EndofProfile TRUE Busy Active

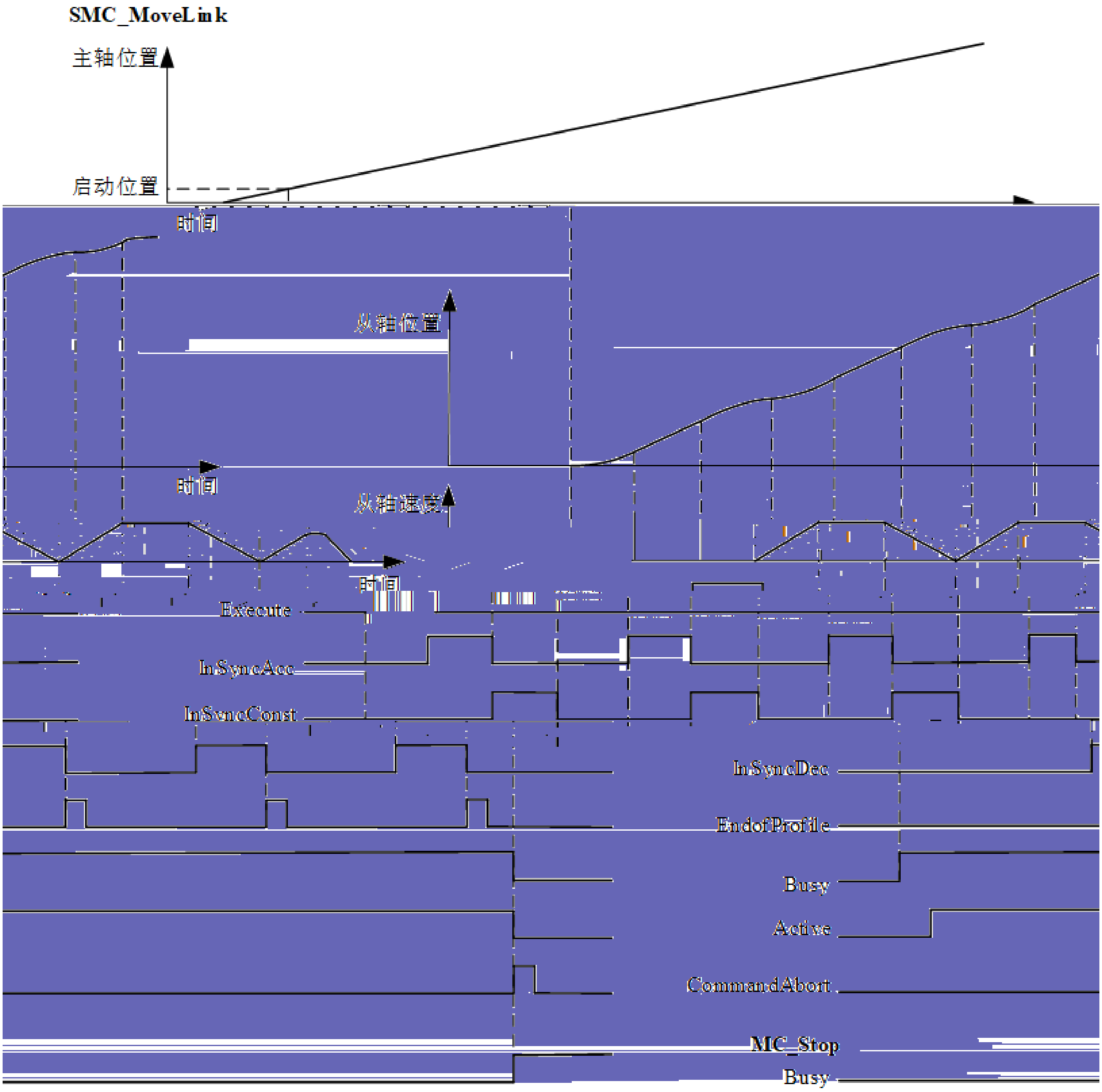
FALSE Execute



2

EndofProfile

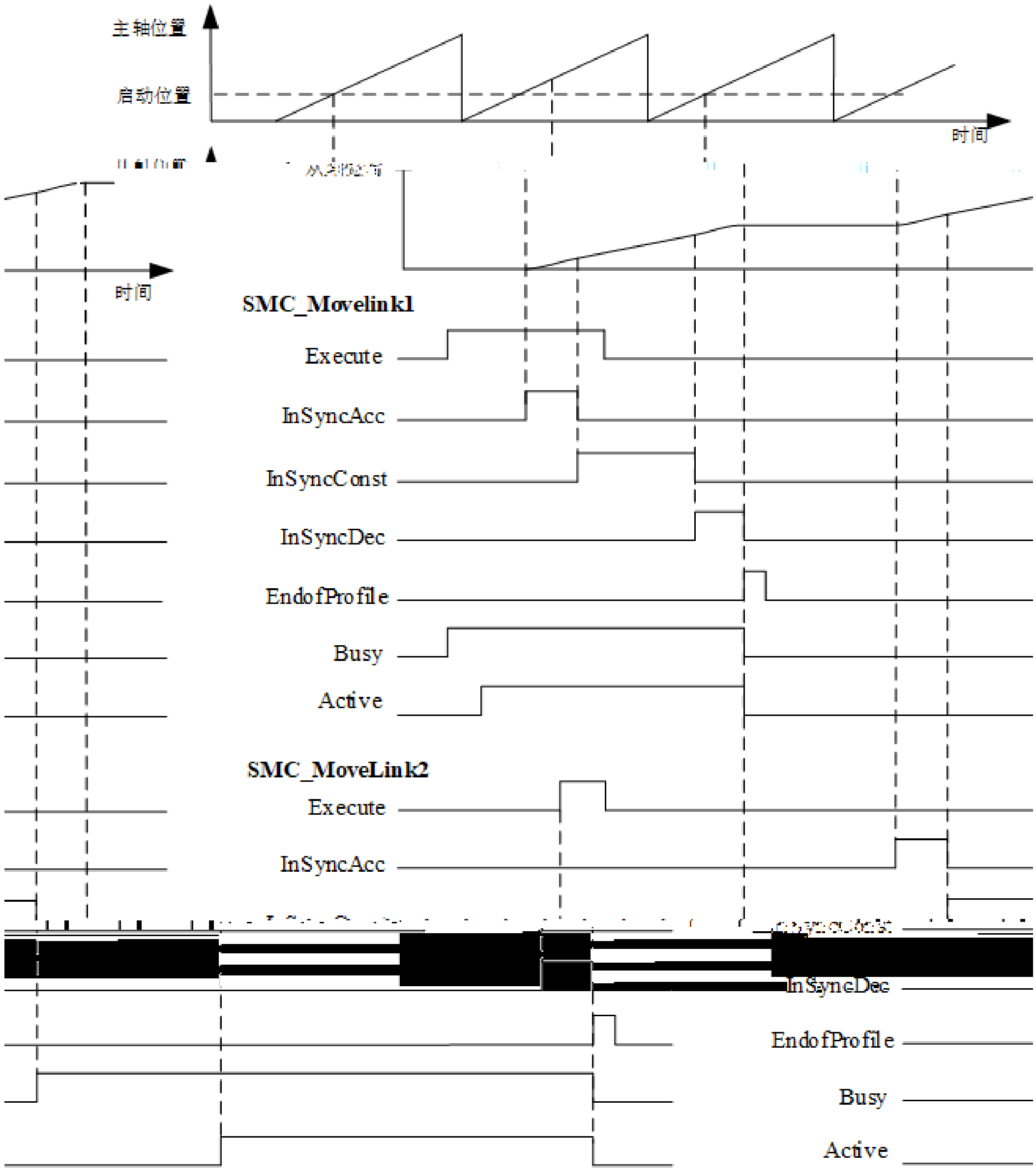
MC_Stop CommandAborted TRUE



BufferMode

a **BufferMode 0** Aborting

b **BufferMode 1** Buffered



EndofProfile

EndofProfile TRUE

Execute FALSE EndofProfile FALSE

EndofProfile TRUE

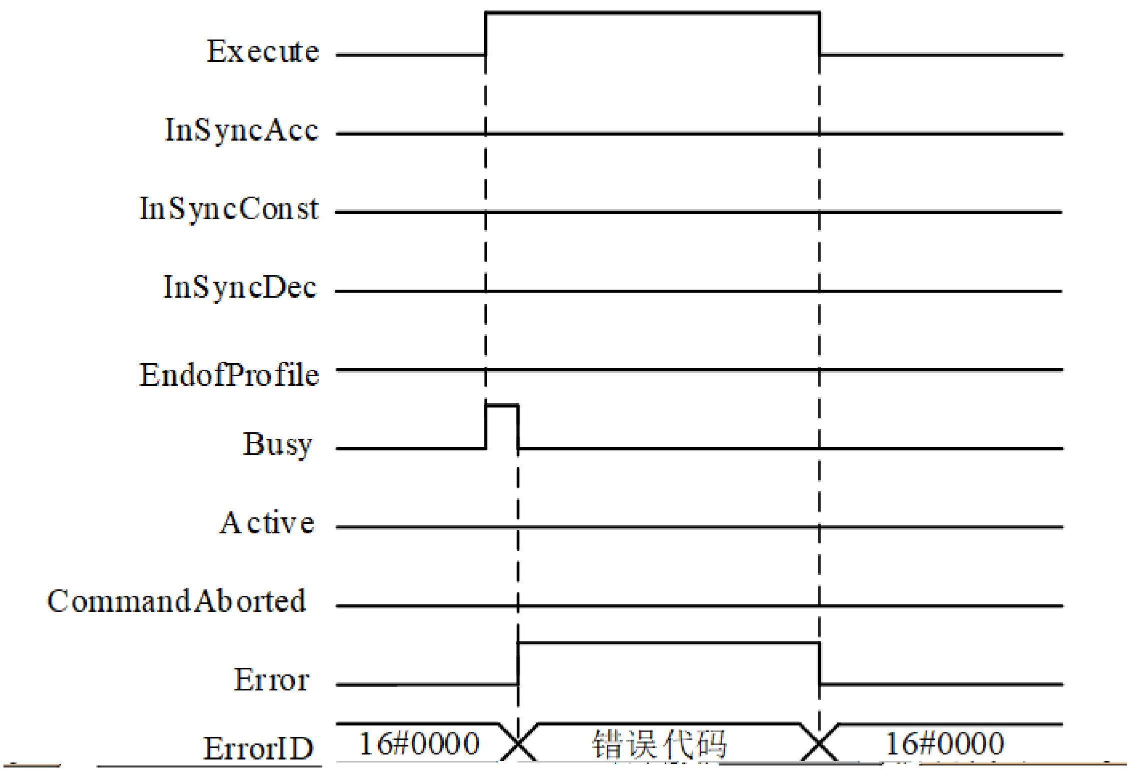
EndofProfile FALSE EndofProfile TRUE

EndofProfile

Error True ErrorID

Execute

SMC_MoveLink异常情况时序图

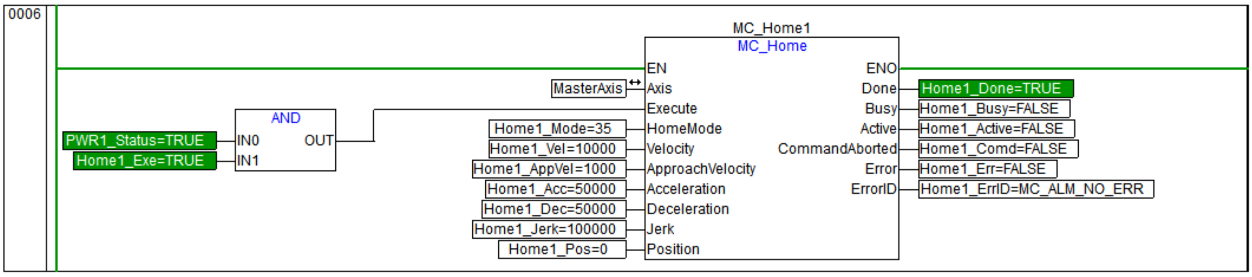


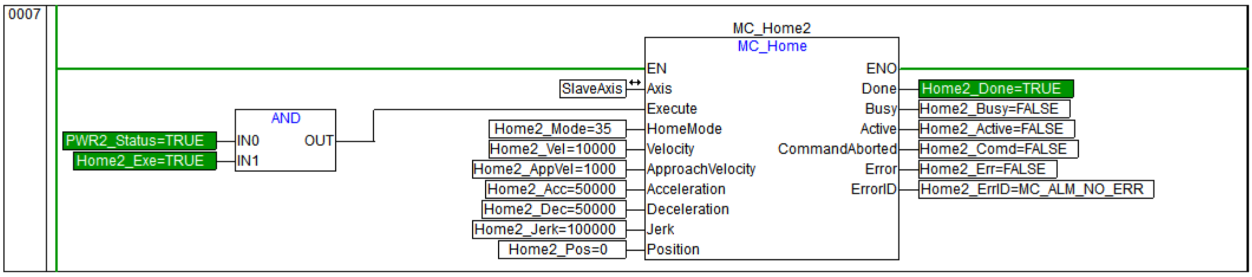
6.3.3

MasterAxis SlaveAxis AXIS_REF

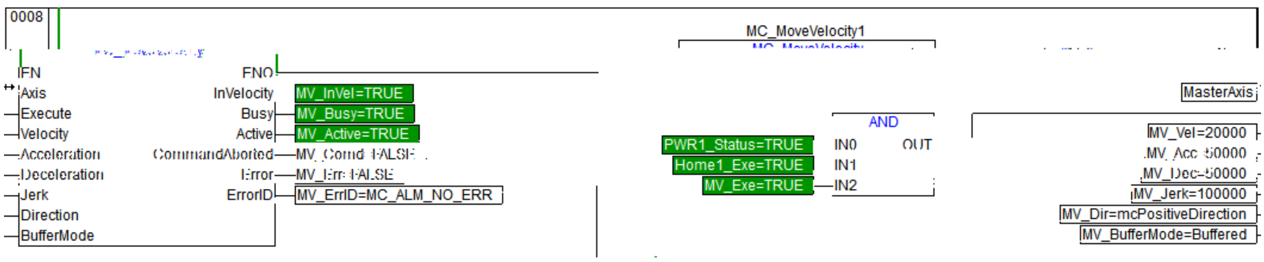
MC_MoveVelocity MC_MoveLink MC_Stop

MasterAxis	AXIS_REF	-	
SlaveAxis	AXIS_REF	-	
Power1_Status	BOOL	FALSE	TRUE
Power2_Status	BOOL	FALSE	TRUE
Switch1	BOOL	FALSE	Switch TRUE
MV_Exe	BOOL	FALSE	
MV_InVel	BOOL	FALSE	TRUE
ML_SlaveDis	LREAL	-	
ML_MasterDis	LREAL	-	
ML_DisAcc	LREAL	-	





4 0008 MC_MoveVelocity( )

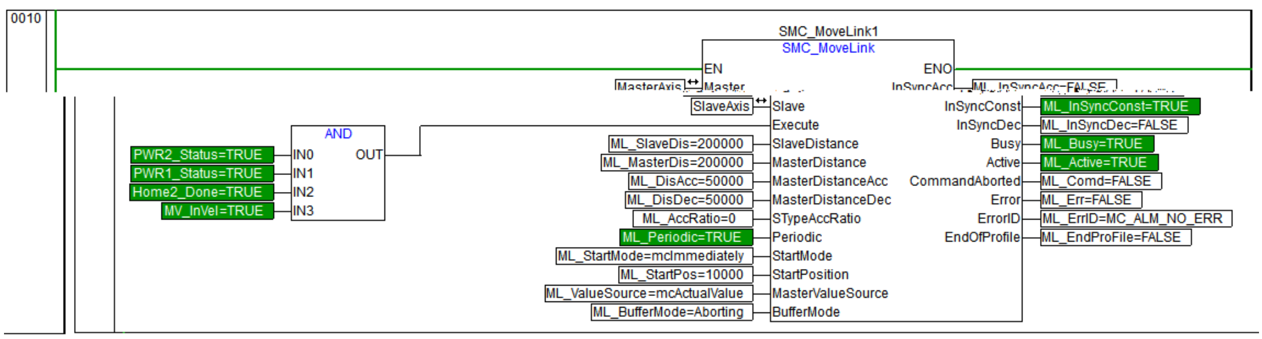


5 0009 MC_MoveVelocity MV_InVel TRUE

SMC_MoveLink1 ML_InSyncAcc TRUE

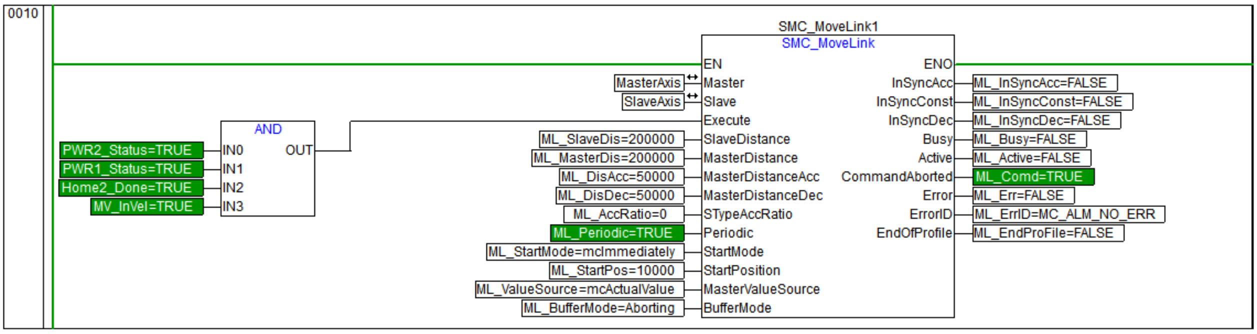
ML_InSyncAcc=TRUE

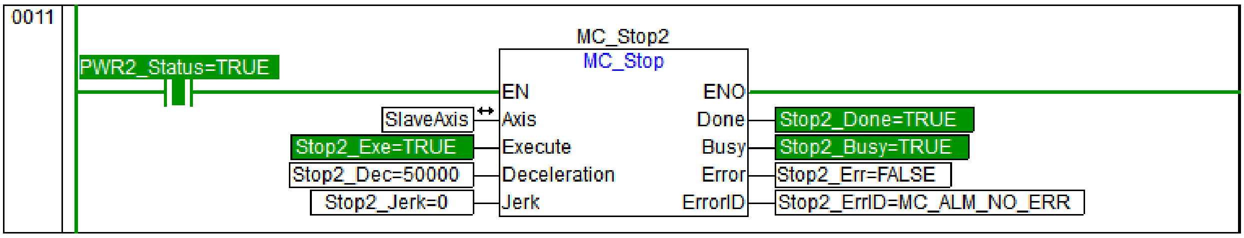
ML_InSyncConst =TRUE ML_InSyncDec =TRUE



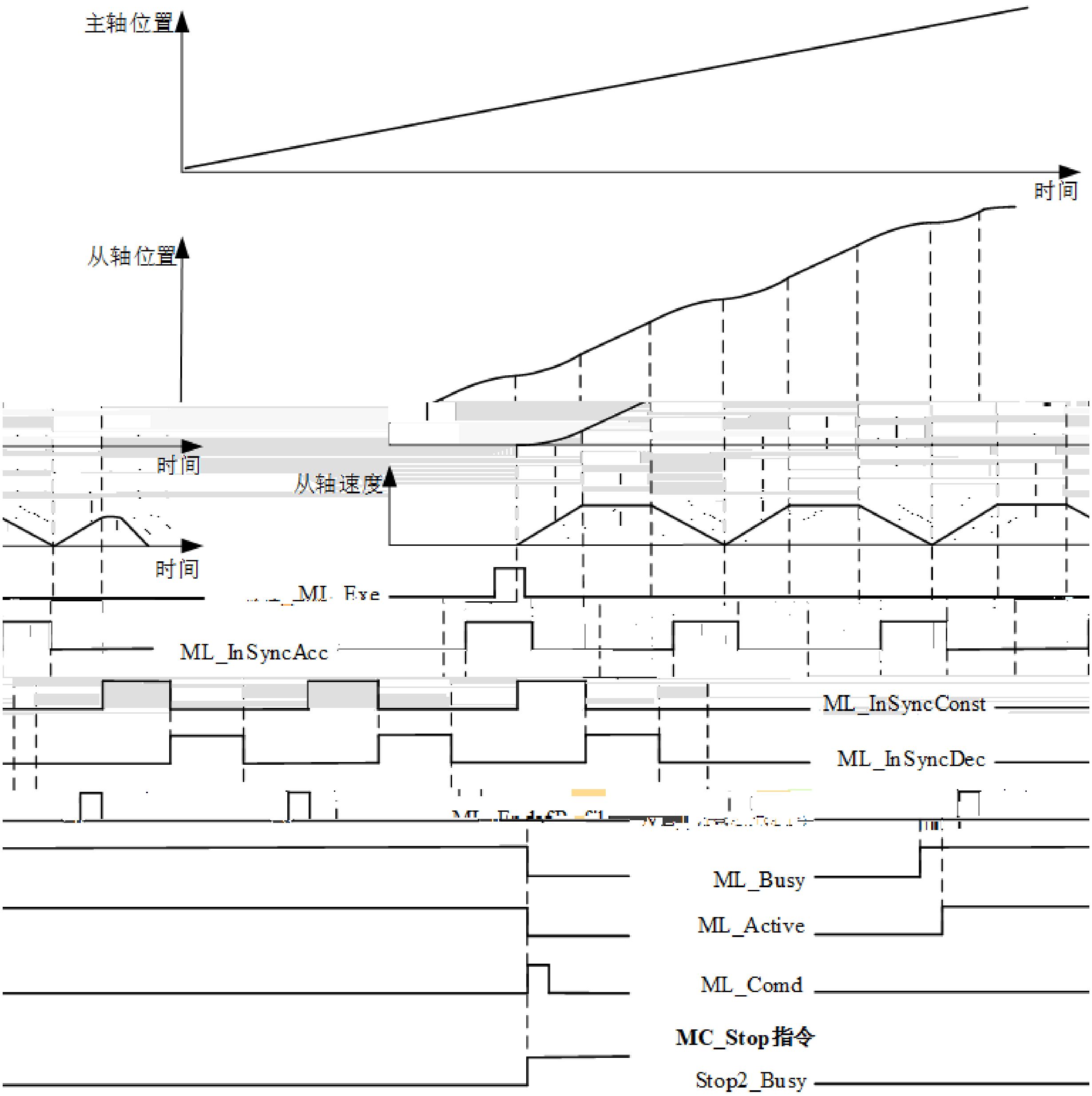
6 MC_Stop 0011

MC_Stop2 ML_Comd TRUE


0010
SMC_MoveLink1
SMC_MoveLink
AND
PWR2_Status=TRUE
PWR1_Status=TRUE
Home2_Done=TRUE
MV_InVel=TRUE
IN0
IN1
IN2
IN3
OUT
MasterAxis
SlaveAxis
ML_SlaveDis=200000
ML_MasterDis=200000
ML_DisAcc=50000
ML_DisDec=50000
ML_AccRatio=0
ML_Periodic=TRUE
ML_StartMode=mcImmediately
ML_StartPos=10000
ML_ValueSource=mcActualValue
ML_BufferMode=Aborting
EN
Master
Slave
Execute
SlaveDistance
MasterDistance
MasterDistanceAcc
MasterDistanceDec
STypeAccRatio
Periodic
StartMode
StartPosition
MasterValueSource
BufferMode
ENO
InSyncAcc
InSyncConst
InSyncDec
Busy
Active
CommandAborted
Error
ErrorID
EndOfProfile
ML_InSyncAcc=FALSE
ML_InSyncConst=FALSE
ML_InSyncDec=FALSE
ML_Busy=FALSE
ML_Active=FALSE
ML_Comd=TRUE
ML_Err=FALSE
ML_ErrID=MC_ALM_NO_ERR
ML_EndProFile=FALSE


0011
MC_Stop2
MC_Stop
PWR2_Status=TRUE
SlaveAxis
Stop2_Exe=TRUE
Stop2_Dec=50000
Stop2_Jerk=0
EN
Axis
Execute
Deceleration
Jerk
ENO
Done
Busy
Error
ErrorID
Stop2_Done=TRUE
Stop2_Busy=TRUE
Stop2_Err=FALSE
Stop2_ErrID=MC_ALM_NO_ERR

SMC_MoveLink指令



ST

100

100

MasterAxis	AXIS_REF	-	
SlaveAxis	AXIS_REF	-	

Power1_Status	BOOL	FALSE	TRUE
Power2_Status	BOOL	FALSE	TRUE
MV_Exe	BOOL	FALSE	
Switch1	BOOL	FALSE	Switch TRUE
InitFlag	BOOL	FALSE	TRUE
ML_SlaveDis	LREAL	-	
ML_MasterDis	LREAL	-	
ML_DisAcc	LREAL	-	
ML_DisDec	LREAL	-	

1

MasterAxis AxisID 0 SlaveAxis AxisID 1 SMC_AxisInit

50 SMC_AxisInit

SMC_AxisInit

2

```
IF FALSE = InitFlag THEN
 ML_SlaveDis:=200;
 ML_MasterDis:=200;
 ML_DisAcc:=10;
 ML_ DisDec:=10;
 ML_Periodic:0ID (=)-8.3 (200)-12.2 (;)]TJ0 Tc 0 Tw 9.301 0 Td( )TEMC ETBuf(k=P0²OfnÐ
```

```
   Axis := SlaveAxis,
   Status=> PWR2_Status,
   Busy=> PWR2_Busy,
   Error=> PWR2_Err
   ErrorID=> PWR2_ErrID);
END_IF
```

4 MC_Home

```
IF  PWR1_Status THEN
  MC_Home1(
  Execute := Home1_Exe,
  HomeMode := Home1_Mode,
  Velocity := Home1_Vel,
  ApproachVelocity := Home1_AppVel,
  Acceleration := Home1_Acc,
  Deceleration := Home1_Dec,
  Jerk := Home1_Jerk,
  Position := Home1_Pos,
  Axis := MasterAxis,
  Done=> Home1_Done,
  Busy=> Home1_Busy,
  Active=> Home1_Active,
  CommandAborted=> Home1_Comd,
  Error=> Home1_Err,
  ErrorID=> Home1_ErrID);
  MC_Home2(
  Execute := Home2_Exe,
  HomeMode := Home2_Mode,
  Velocity := Home2_Vel,
  ApproachVelocity := Home2_AppVel,
  Acceleration := Home2_Acc,
  Deceleration := Home2_Dec,
  Jerk := Home2_Jerk,
  Position := Home2_Pos,
  Axis := SlaveAxis,
  Done=> Home2_Done,
  Busy=> Home2_Busy,
  Active=> Home2_Active,
  CommandAborted=> Home2_Comd,
  Error=> Home2_Err,
  ErrorID=> Home2_ErrID);
END_IF
```

5

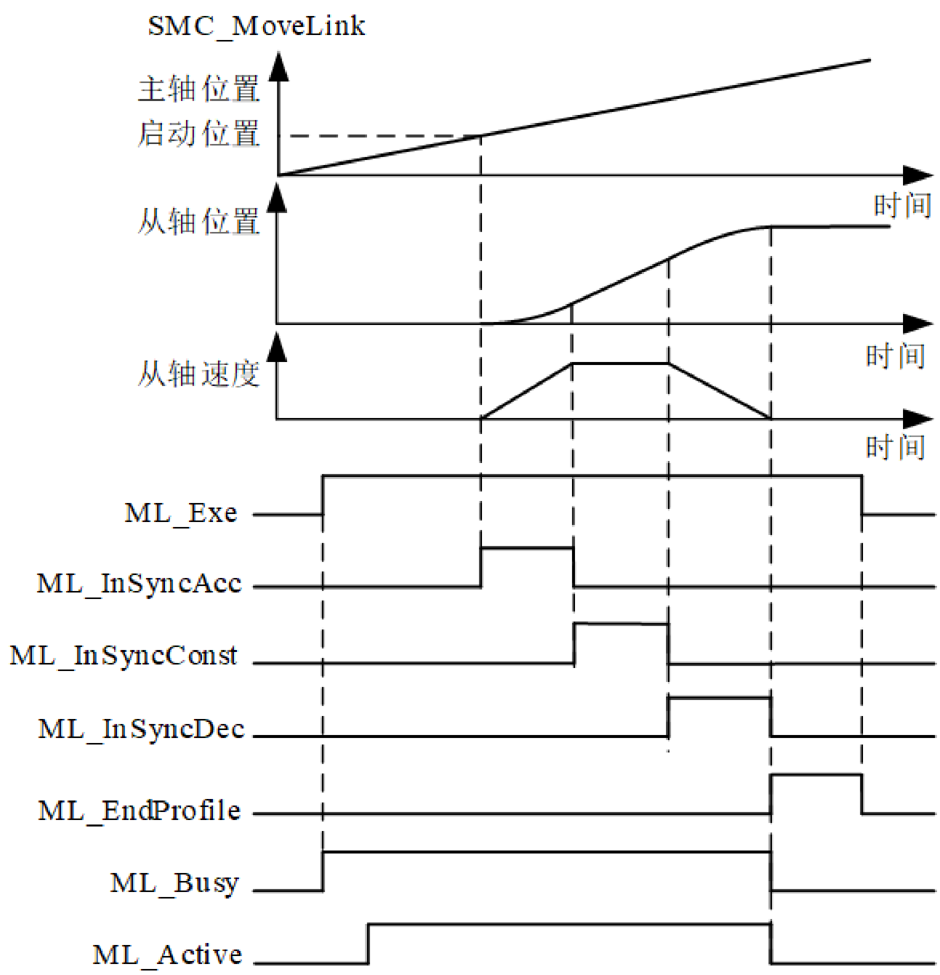
```
IF  PWR1_Status AND PWR2_Status AND Home1_Done THEN
```

```
  SMC_MoveLink1(
  Execute := ML_Exe,
  SlaveDistance := ML_SlaveDis,
  MasterDistance := ML_MasterDis,
  MasterDistanceAcc := ML_DisAcc,
  MasterDistanceDec := ML_ DisDec,
  STypeAccRatio := ML_AccRatio,
  Periodic := ML_Periodic,
  StartMode := ML_STartMode,
  StartPosition := ML_StartPosition,
  MasterValueSource := ML_ValueSource,
  BufferMode := ML_Buffer,
  Master := MasterAxis,
  Slave := SlaveAxis,
  InSyncAcc=> ML_InSyncAcc,
  InSyncConst=> ML_InSyncConst,
  InSyncDec=> ML_InSyncDec,
  Busy=> ML_Busy,
  Active=> ML_Act,
  CommandAborted=> ML_Comd,
  Error=> ML_Err,
  ErrorID=> ML_ErrID,
  EndOfProfile=> ML_EndPro);
END_IF
```

6

```
IF  PWR1_Status AND Home1_Done THEN
  MC_MoveVelocity1(
  Execute := MV_Exe,
  Velocity := MV_Vel,
  Acceleration := MV_Acc,
  Deceleration := MV_Dcc,
  Jerk := MV_Jerk,
  Direction := MV_Dir,
  BufferMode := MV_Buffer,
  Axis := Axis0,
  InVelocity=> MV_InVel,
  Busy=> MV_Busy,
  Active=> MV_Active,
  CommandAborted=> MV_Comm,
  Error=> MV_Err,
  ErrorID=> MV_Error);
END_IF
```

7



6.3.4

DI StartPosition StartPosition I

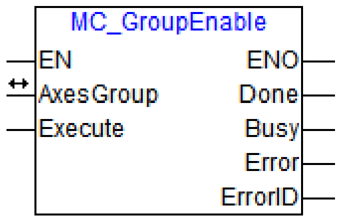
SlaveDistance 0 MoveLink MasterDistanceAcc InSyncAcc TRUE

MasterDistanceDec InSyncDec TRUE InSyncConst TRUE

MasterDistance( ) 1

7

7.1 MC_GroupEnable



7.1.1

IN_OUT	AxesGroup			-	-	AXES_GROUP_REF
INPUT	Execute			-	TRUE/FALSE	BOOL
OUTPUT	Done			FALSE	TRUE/FALSE	BOOL
	Busy			FALSE	TRUE/FALSE	BOOL
	Error			FALSE	TRUE/FALSE	BOOL
	ErrorID			0	-	SMC_ERROR

AxisInterp_ID	ID	255	0~63	USINT

AxisX_ID	X ID	255	0~63	USINT
AxisY_ID	Y ID	255	0~63	UDINT
ReservedPara1		255	-	USINT
ReservedPara2		255	-	USINT
ReservedPara3		255	-	USINT
ReservedPara4		255	-	USINT
ReservedPara5		255	-	USINT
AxesGroupDisabled	Disabled	TRUE	TRUE/FALSE	BOOL

- AT ID X ID Y ID

7.1.2

1 MC_GroupEnable AxesGroup Enable

2 ID

ID

3 AxesGroup AXES_GROUP_REF

AxesGroup.AxesGroupDisabled=FALSE

4 Execute MC_GroupEnable

AxesGroupDisabled FALSE AxesGroupDisabled

MC_GroupEnable MC_GroupDisable

5 MC_Power

6 MC_Power

7 MC_GroupDisable

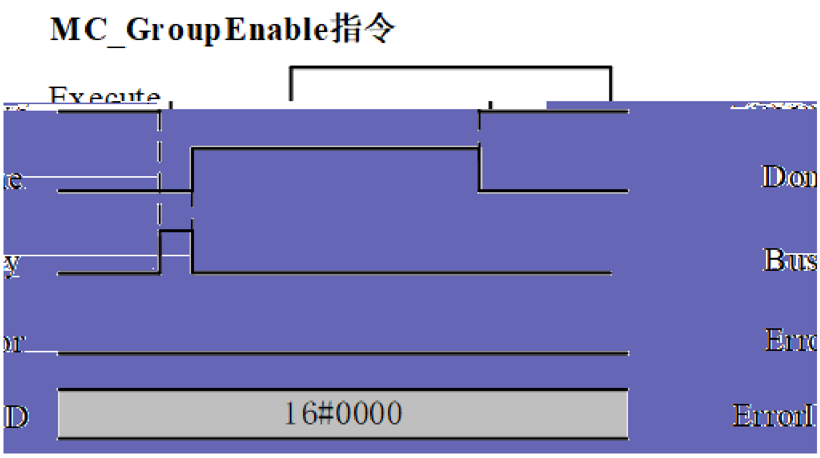
AxesGroupDisabled TRUE

8 EtherCat

Virtual

MC_GroupEnable

1

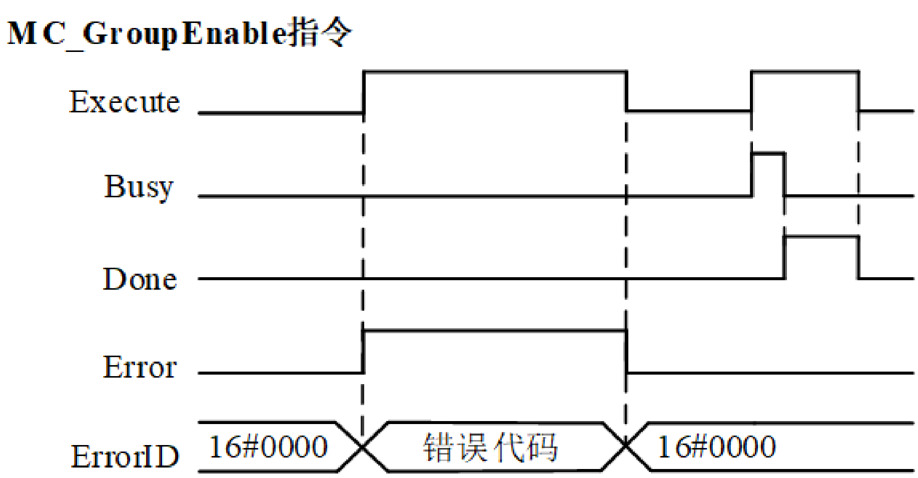


2

ID

Error=True ErrorID

Execute



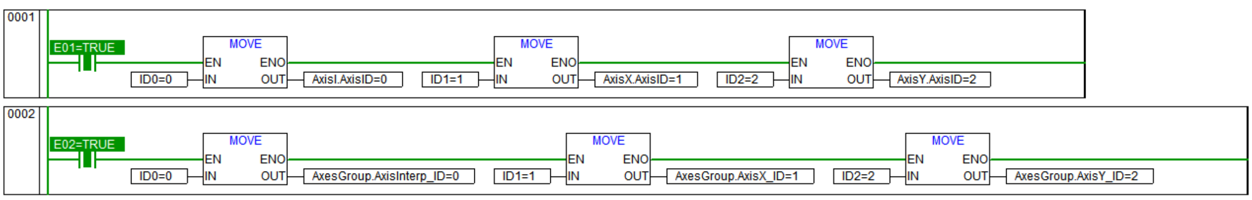
- MC_GroupEnable

7.1.3

AxesGroup	AXES_GROUP_REF	-	
AxisI	AXIS_REF	-	
AxisX	AXIS_REF	-	X
AxisY	AXIS_REF	-	Y
StartProg	BOOL	FALSE	TRUE
Power0_Status	BOOL	FALSE	TRUE
Power1_Status	BOOL	FALSE	X X TRUE
Power2_Status	BOOL	FALSE	Y Y TRUE
GrpEnable1_Exe	BOOL	FALSE	

1 ID

ID ID 0001 0002



2 SMC_AxisInit 0

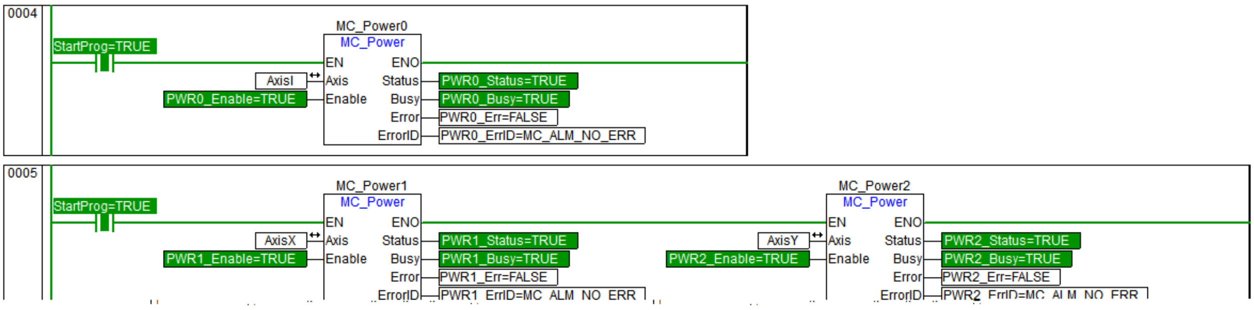
Virtual 50: EtherCAT_Position SMC_AxisInit

- MC_GroupEnable

  MC_Power

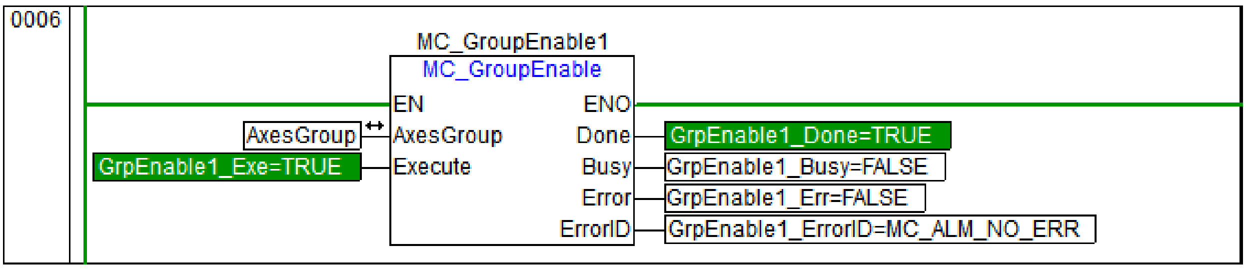
3 StartProg TRUE

0004 ~ 0005



4 MC_GroupEnable AxesGroupDisabled FALSE

0006



ST

AxesGroup	AXES_GROUP_REF	-	
AxisI	AXIS_REF	-	
AxisX	AXIS_REF	-	X
AxisY	AXIS_REF	-	Y
StartProg	BOOL	FALSE	TRUE
Power0_Status	BOOL	FALSE	

```
  AxesGroup.AxisY_ID:=AxisY.AxisID;
END_IF
```

2 SMC_AxisInit 0 Virtual

AxisX/AxisY 50 EtherCAT_Position SMC_AxisInit

3 StartProg TRUE

```
IF Start
```

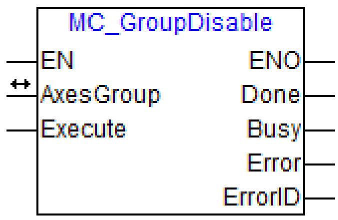

1Y

7.1.4

ID

AxesGroupDisabled MC_GroupEnable/MC_GroupDisable

7.2 MC_GroupDisable



7.2.1

IN_OUT	AxesGroup			-	-	AXES_GROUP_REF
INPUT	Execute			-	TRUE/FALSE	BOOL
OUTPUT	Done			FALSE	TRUE/FALSE	BOOL
	Busy			FALSE	TRUE/FALSE	BOOL
	Error			FALSE	TRUE/FALSE	BOOL
	ErrorID			0	-	SMC_ERROR

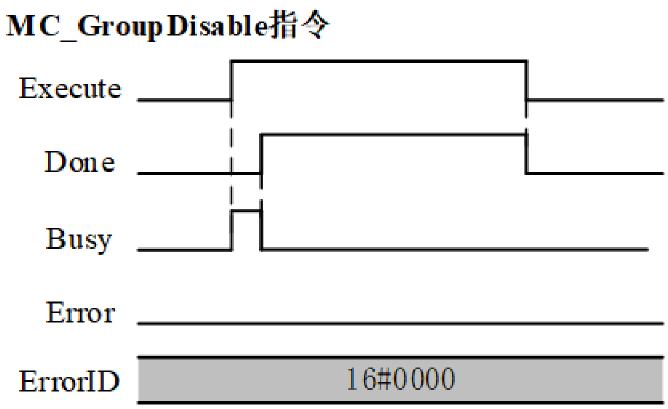
7.2.2

1 Execute MC_GroupDisable

AxesGroupDisabled TRUE Disable

2

1



2

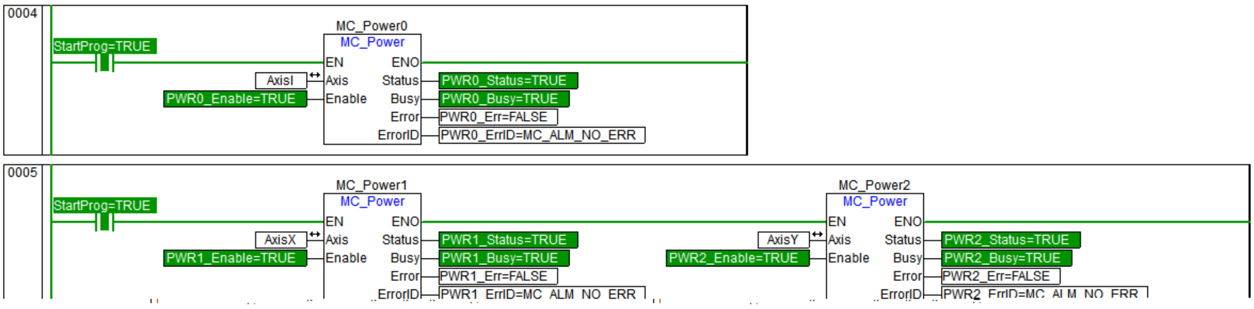
ID

Error=True ErrorID

7.2.3

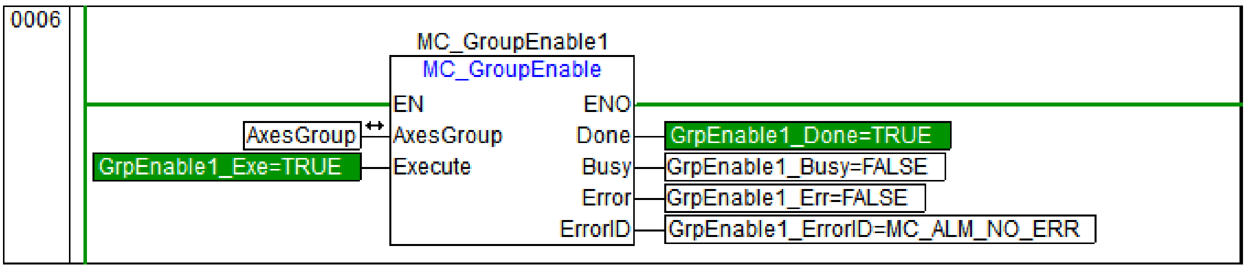
AxesGroup	AXES_GROUP_REF	-	
AxisI	AXIS_REF	-	
AxisX	AXIS_REF	-	X
AxisY	AXIS_REF	-	Y
StartProg	BOOL	FALSE	TRUE
Power0_Status	BOOL	FALSE	TRUE
Power1_Status	BOOL	FALSE	X X TRUE
Power2_Status	BOOL		

0004~0005



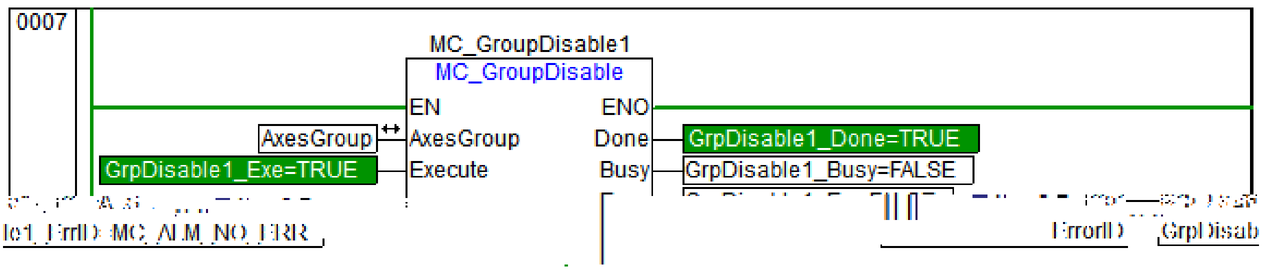
4 MC_GroupEnable AxesGroup. AxesGroupDisabled

FALSE 0006



5 MC_GroupDisable AxesGroupDisabled

TRUE 0007



ST

1 ID AxisI AxisX AxisY ID

```
IF Switch1 THEN
   AxisI.AxisID:=0;
   AxisX.AxisID:=1;
   AxisY.AxisID:=2;
   AxesGroup.AxisInterp_ID:=AxisI.AxisID;
   AxesGroup.AxisX_ID:=AxisX.AxisID;
   AxesGroup.AxisY_ID:=AxisY.AxisID;
END_IF
```

2 SMC_AxisInit 0 Virtual

AxisX/AxisY 50 EtherCAT_Position SMC_AxisInit

3 StartPro TRUE

```
IF StartPro THEN
   MC_Power0(                    (*          *)
   Enable := PWR0_Enable,
   Axis := AxisI,
   Status=> PWR0_Status,
   Busy=> PWR0_Busy,
   Error=> PWR0_Err,
   ErrorID=> PWR0_ErrID);
   MC_Power1(                    (*    AxisX     *)
   Enable := PWR1_Enable,
   Axis := AxisX,
   Status=> PWR1_Status,
   Busy=> PWR1_Busy,
   Error=> PWR1_Err,
   ErrorID=> PWR1_ErrID);
   MC_Power2(                    (*    AxisY     *)
   Enable := PWR2_Enable,
   Axis := AxisY,
   Status=> PWR2_Status,
   Busy=> PWR2_Busy,
   Error=> PWR2_Err,
   ErrorID=> PWR2_ErrID);
END_IF
```

4

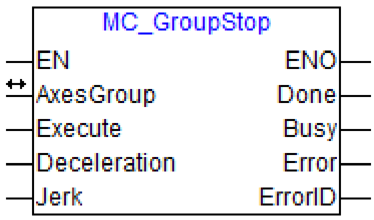
```
MC_GroupEnable1(
Execute := GrpEnable_Exe,
AxesGroup := AxesGroup,
Done=> GrpEnable_Done,
Busy=> GrpEnable_Busy,
Error=> GrpEnable_Err,
ErrorID=> GrpEnable_ErrorID);
```

5 MC_GroupDisable

```
MC_GroupDisable1(
Execute := GrpDisable_Exe,
AxesGroup := AxesGroup,
```

```
Done=> GrpDisable_Done,
Busy=> GrpDisable_Busy,
Error=> GrpDisable_Err,
ErrorID=> GrpDisable_ErrorID);
```

7.3 MC_GroupStop



7.3.1

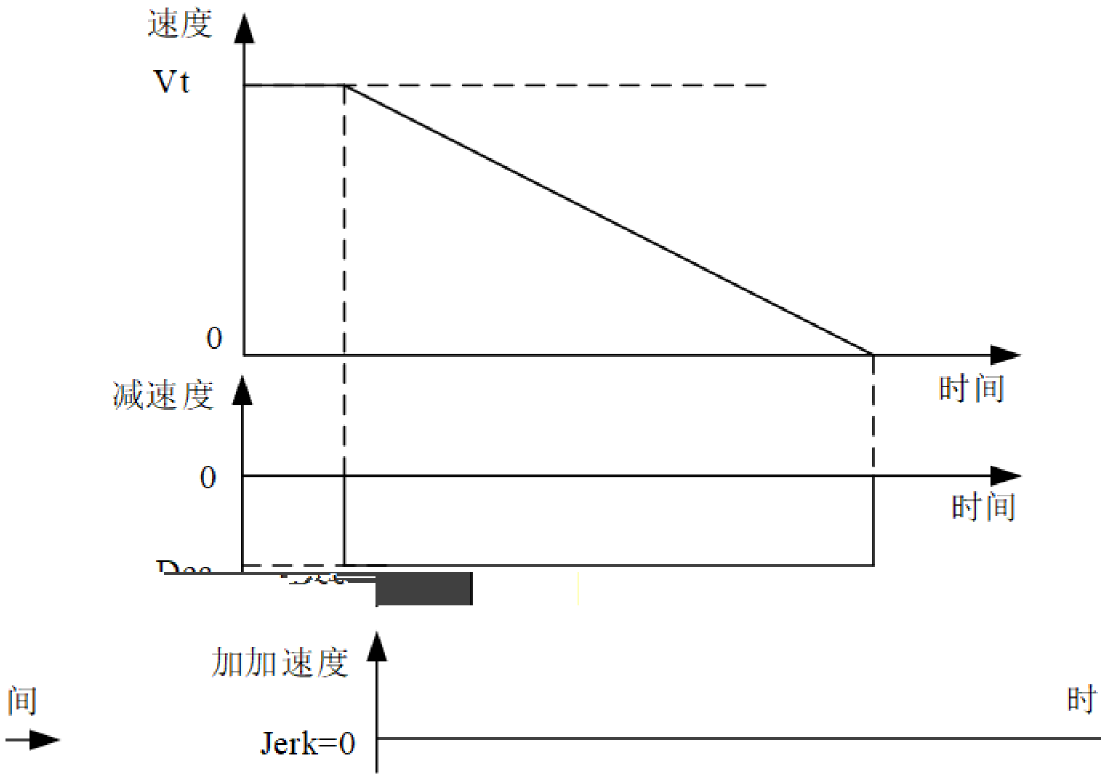
IN_OUT	AxesGroup			-	-	AXES_GROUP_REF
INPUT	Execute			-	TRUE/FALSE	BOOL
	Deceleration			-		LREAL
	Jerk	0 S		0	0/	LREAL
OUTPUT	Done			FALSE	TRUE/FALSE	BOOL
	Busy			FALSE	TRUE/FALSE	BOOL
	Error			FALSE	TRUE/FALSE	BOOL
	ErrorID			0	-	SMC_ERROR

7.3.2

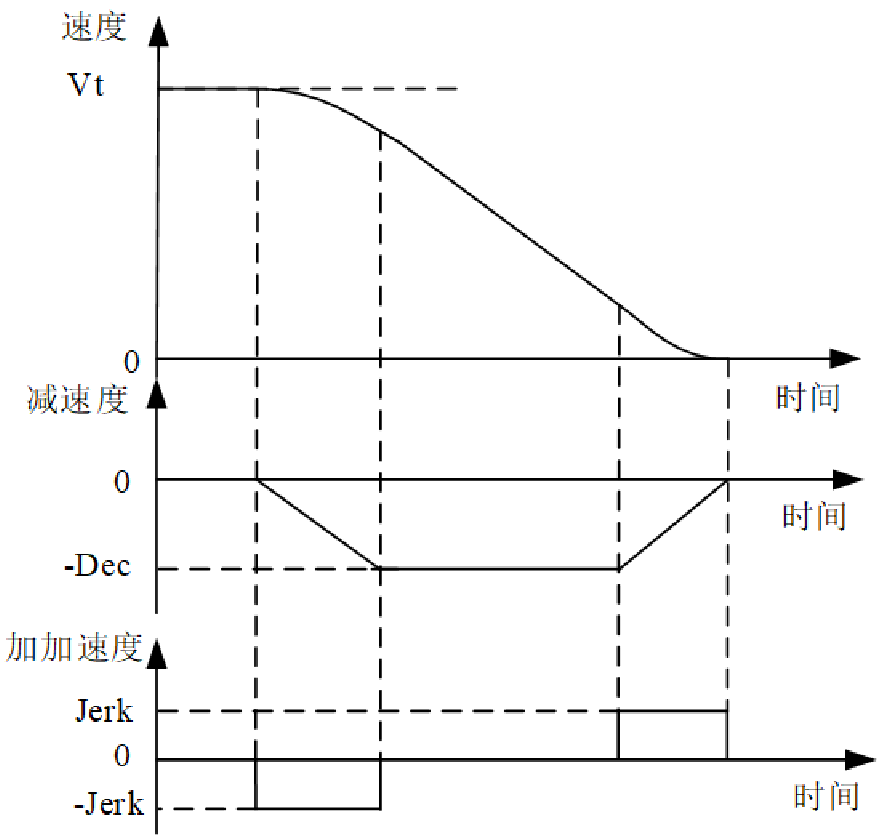
1

Deceleration( ) Jerk( )

2



Jerk>0 S



MC_GroupStop SMC_MoveLinearAbsolute(2

) SMC_MoveLinearRelative(2) SMC_MoveCircular2D(2

)

MC_GroupStop

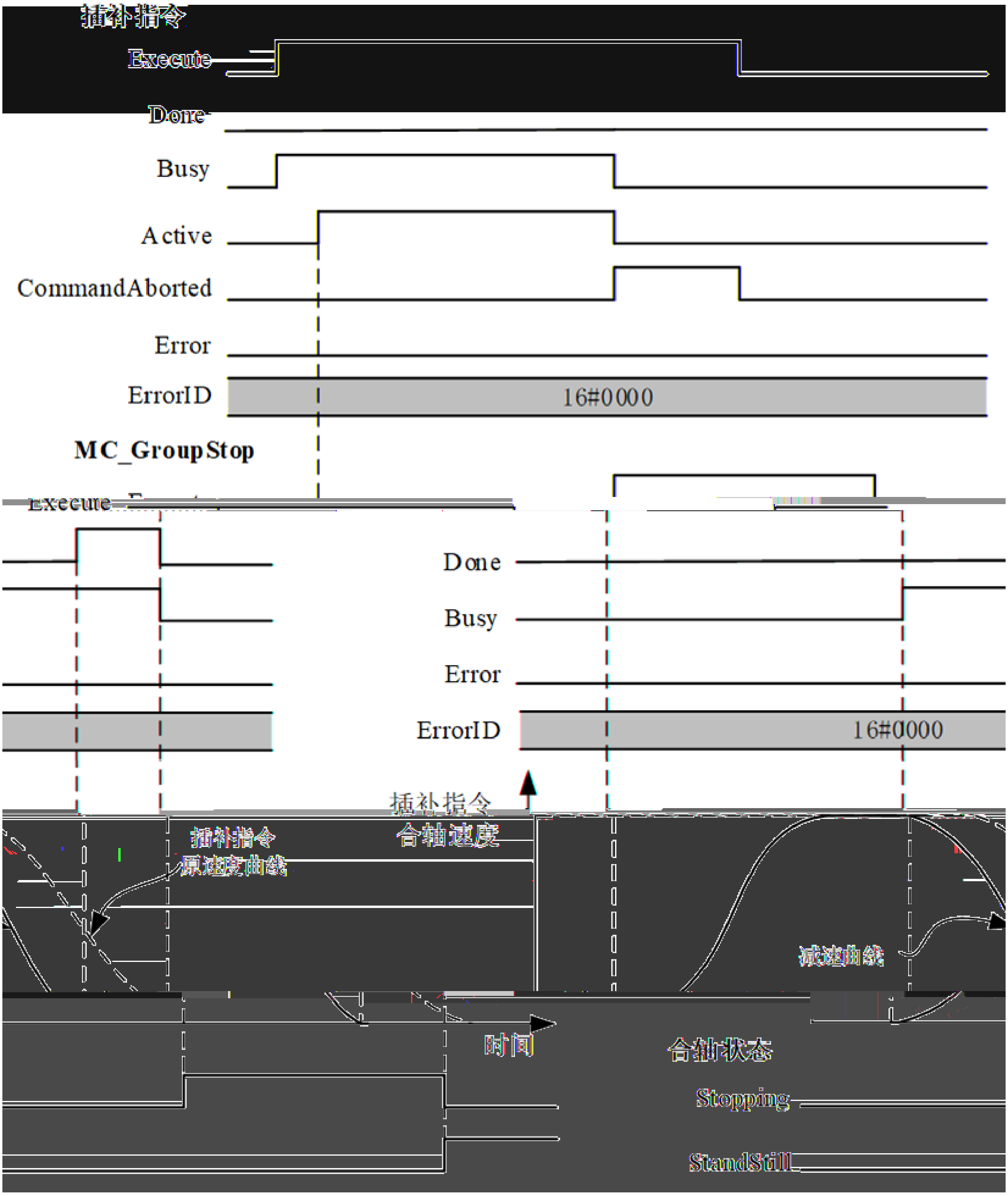
1

Execute Busy TRUE

Stopping Done

TRUE Execute Busy Done TRUE Stopping

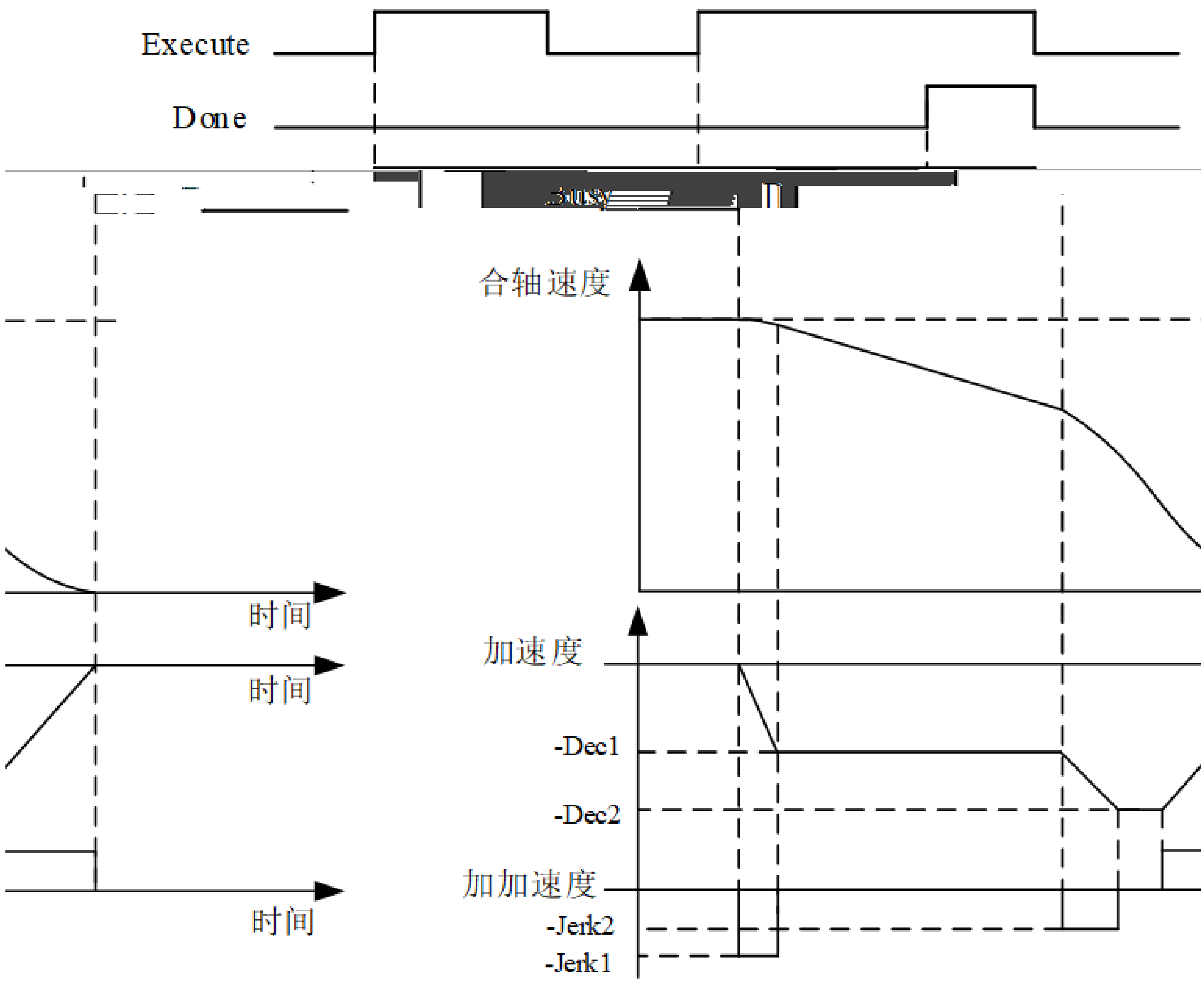
Execute Busy Done FALSE StandStill



2

MC_GroupStop Deceleration

MC_GroupStop指令

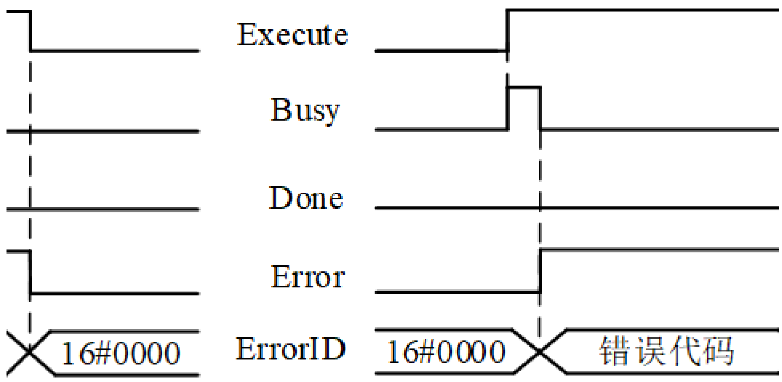


3

Disabled ErrorStop

Error=True ErrorID

Execute



- Execute MC_GroupStop

7.3.3

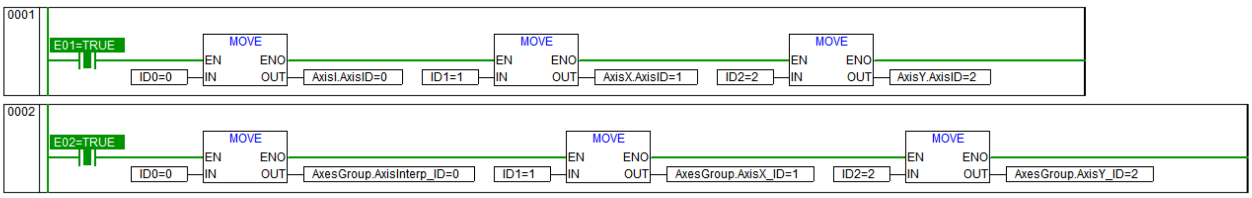
MC_GroupStop 2 MC_ReadStatus

AxesGroup	AXES_GROUP_REF	-	
AxisI	AXIS_REF	-	
AxisX	AXIS_REF	-	X
AxisY	AXIS_REF	-	Y
StartProg	BOOL	FALSE	TRUE
Power0_Status	BOOL	FALSE	TRUE
Power1_Status	BOOL	FALSE	X X TRUE
Power2_Status	BOOL	FALSE	Y Y TRUE
GrpEnableStatus	BOOL	TRUE	TRUE
GrpEnable1_Exe	BOOL	FALSE	
Rel2D1_Exe	BOOL	FALSE	
Rel2D1_	BOOL	FALSE	

Comdabort			
GrpStop1_Busy	BOOL	FALSE	Busy
GrpStop1_Don3	BOOL	FALSE	Done TRUE
Stopping	BOOL	FALSE	Stopping TRUE

1 ID

ID ID 0001 0002

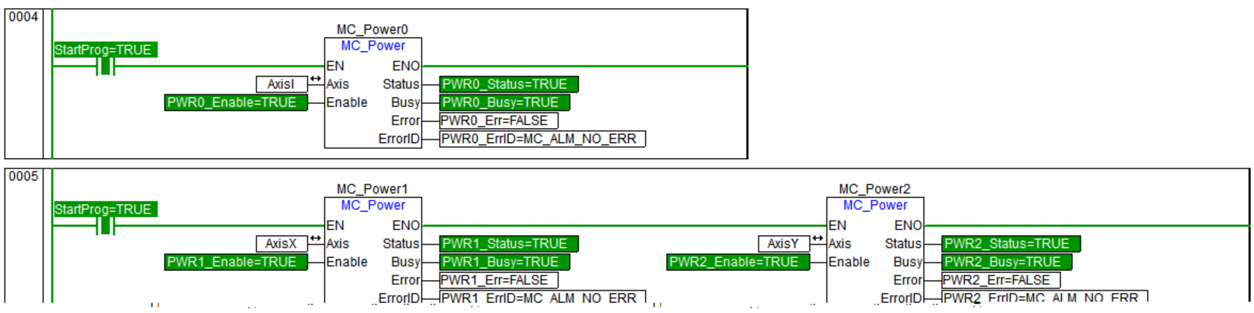


2 SMC_AxisInit 0

Virtual 50: EtherCAT_Position SMC_AxisInit

3 StartProg =TRUE MC_Power

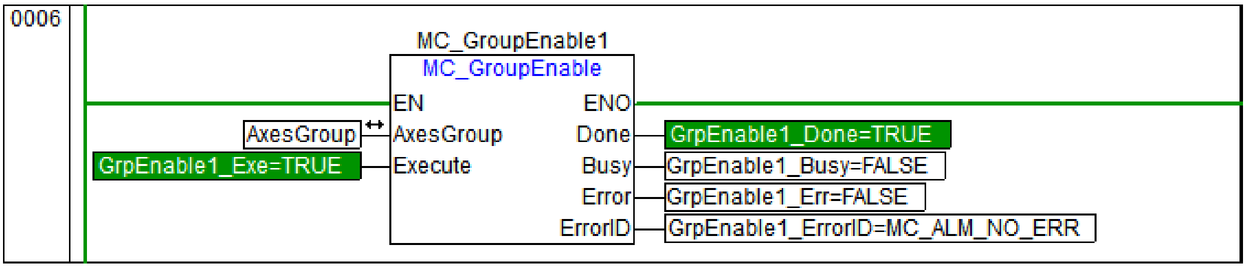
0004~0005

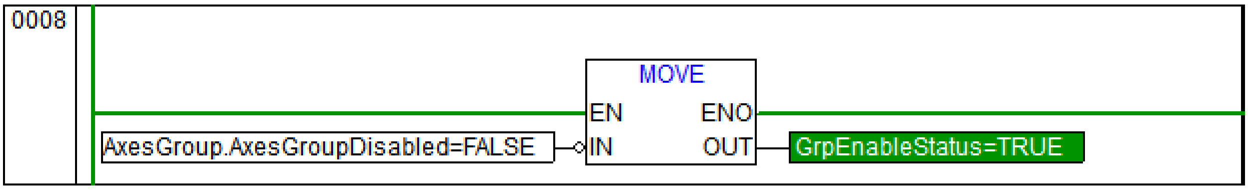


4 MC_GroupEnable AxesGroupDisabled FALSE

0006 AxesGroupDisabled

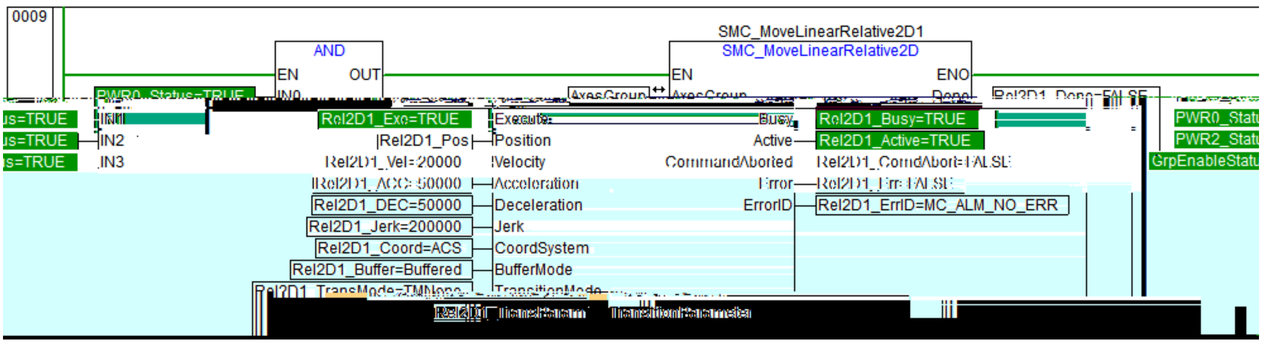
GrpEnableStatus





5 0009

SMC_MoveLinearRelative2D

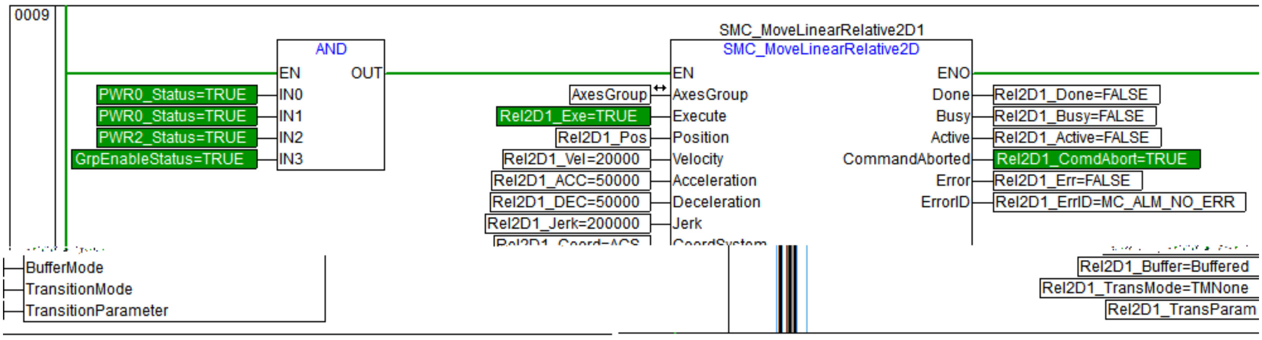


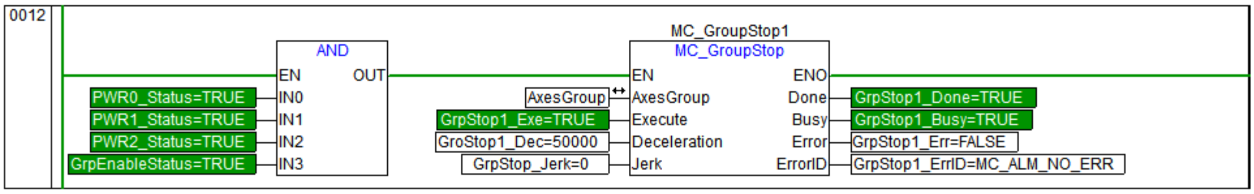
6 0012 MC_GroupStop 50000 Jerk 0

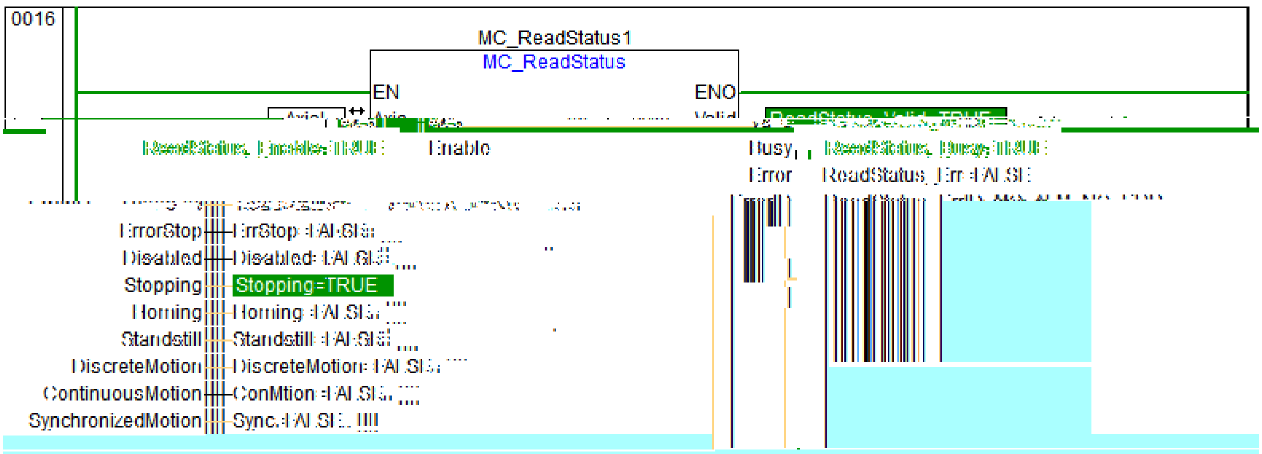
MC_GroupStop SMC_MoveLinearRelative2D1

CommandAborted TRUE 0009 0016

MC_ReadStatus1


0009
AND
EN
OUT
IN0
IN1
IN2
IN3
PWR0_Status=TRUE
PWR0_Status=TRUE
PWR2_Status=TRUE
GrpEnableStatus=TRUE
SMC_MoveLinearRelative2D1
SMC_MoveLinearRelative2D
AxesGroup
Rel2D1_Exe=TRUE
Rel2D1_Pos
Rel2D1_Vel=20000
Rel2D1_ACC=50000
Rel2D1_DEC=50000
Rel2D1_Jerk=200000
Execute
Position
Velocity
Acceleration
Deceleration
Jerk
BufferMode
TransitionMode
TransitionParameter
ENO
Done
Busy
Active
CommandAborted
Error
ErrorID
Rel2D1_Done=FALSE
Rel2D1_Busy=FALSE
Rel2D1_Active=FALSE
Rel2D1_ComdAbort=TRUE
Rel2D1_Err=FALSE
Rel2D1_ErrID=MC_ALM_NO_ERR
Rel2D1_Buffer=Buffered
Rel2D1_TransMode=TMNone
Rel2D1_TransParam


0012
AND
EN
OUT
IN0
IN1
IN2
IN3
PWR0_Status=TRUE
PWR1_Status=TRUE
PWR2_Status=TRUE
GrpEnableStatus=TRUE
MC_GroupStop1
MC_GroupStop
AxesGroup
GrpStop1_Exe=TRUE
GroStop1_Dec=50000
GrpStop_Jerk=0
Execute
Deceleration
Jerk
ENO
Done
Busy
Error
ErrorID
GrpStop1_Done=TRUE
GrpStop1_Busy=TRUE
GrpStop1_Err=FALSE
GrpStop1_ErrID=MC_ALM_NO_ERR


0016
MC_ReadStatus1
MC_ReadStatus
EN
ENO
Enable
Busy
Error
ErrorStop
Disabled
Stopping
Stopping=TRUE
Homing
Standstill
DiscreteMotion
ContinuousMotion
SynchronizedMotion

ST

```
   Error=> PWR2_Err,
   ErrorID=> PWR2_ErrID);
END_IF
```

4

```
MC_GroupEnable1(
Execute := GrpEnable_Exe,
AxesGroup := AxesGroup,
Done=> GrpEnable_Done,
Busy=> GrpEnable_Busy,
Error=> GrpEnable_Err,
ErrorID=> GrpEnable_ErrorID);
GrpEnableStatus:= AxesGroup. AxesGroupDisabled;
```

5 SMC_MoveLinearRelative2D_

```
IF PWR0_Status  AND PWR1_Status  AND PWR2_Status  AND GrpEnableStatus THEN
   SMC_MoveLinearRelative2D1(
   Execute := Rel2D1_Exe,
   Position := Rel2D1_Pos,
   Velocity := Rel2D1_Vel,
   Acceleration := Rel2D1_ACC,
   Deceleration := Rel2D1_DEC,
   Jerk := Rel2D1_Jerk,
   CoordSystem := Rel2D1_Coord,
   BufferMode := Rel2D1_Buffer,
   TransitionMode := Rel2D1_Trans,
   TransitionParameter := Rel2D1_Param,
   AxesGroup := AxesGroup,
   Done=> Rel2D1_Done,
   Busy=> Rel2D1_Busy,
   Active=> Rel2D1_Active,
   CommandAborted=> Rel2D1_ComdAbort,
   Error=> Rel2D1_Err,
   ErrorID=> Rel2D1_ErrID);
END_IF
```

6 MC_GroupStop 50000 Jerk 0

MC_GroupStop SMC_MoveLinearRelative2D1

CommandAborted TRUE MC_ReadStatus1

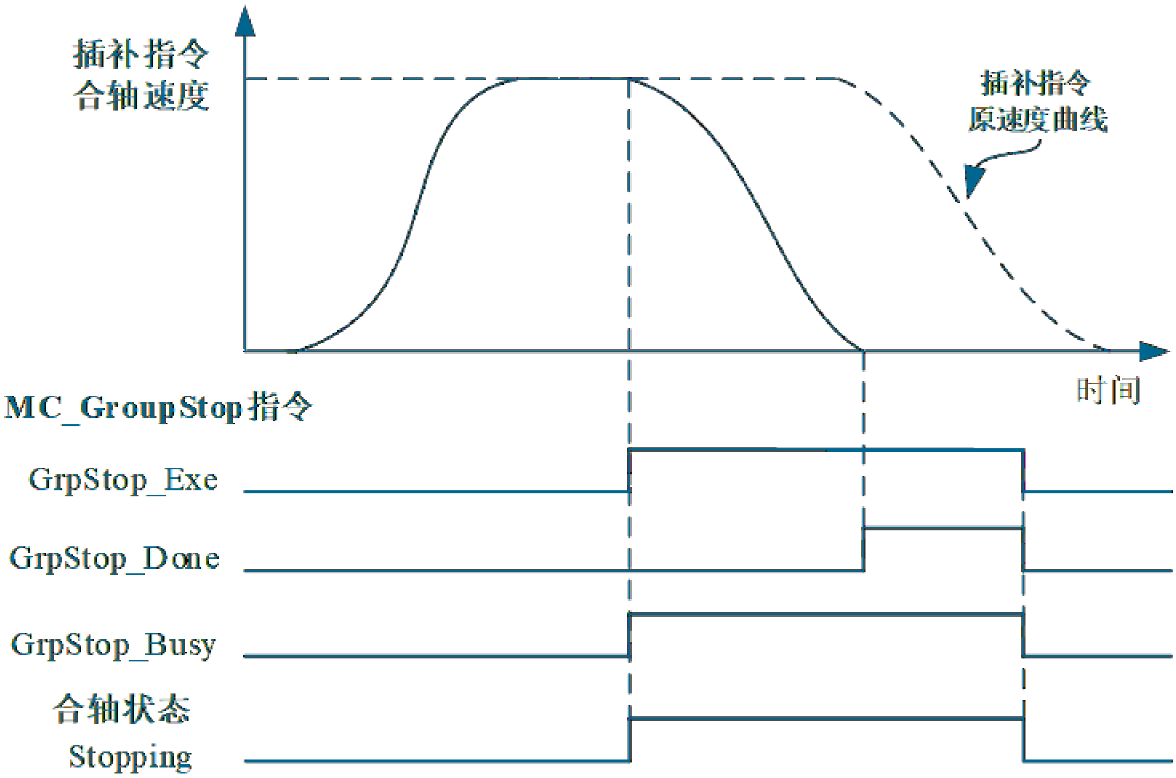
```
IF PWR0_Status  AND PWR1_Status  AND PWR2_Status  AND GrpEnableStatus THEN
   MC_GroupStop1(
```

```
   Execute := GrpStop_Exe,
   Deceleration := GrpStop_Dec,
   Jerk := GrpStop_Jerk,
   AxesGroup := AxesGroup,
   Done=> GrpStop_Done,
   Busy=> GrpStop_Busy,
   Error=> GrpStop_Err,
   ErrorID=> GrpStop_ErrID);
END_IF
```

7

```
MC_ReadStatus1(
Enable := ReadStatus Enable,
Axis := AxisI,
Valid=> ReadStatus_Valid,
Busy=> ReadStatus_Busy,
Error=> ReadStatus_Err,
ErrorID=> ReadStatus ErrID,
ErrorStop=>ErrStop,
Disabled=>Disabled,
Stopping=>Stopping,
Homing=>Homing,
Standstill=>Standstill,
DiscreteMotion=>DiscreteMotion,
ContinuousMotion=>ConMtion,
SynchronizedMotion=>Sync);
```

MC_GroupStop



7.3.4

MC_GroupStop

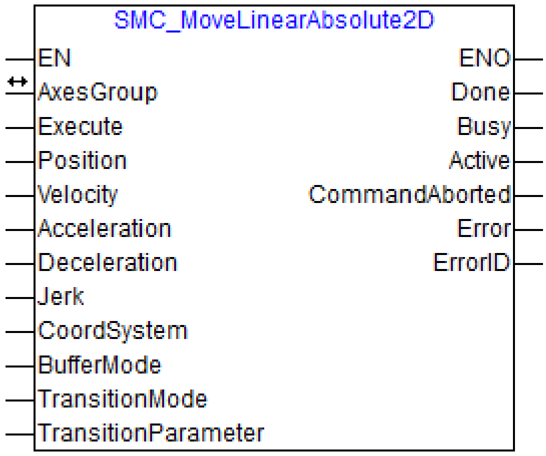
MC_GroupStop MC_GroupStop Execute

Stopping Execute

StandStill

7.4 SMC_MoveLinearAbsolute2D 2

2



7.4.1

IN_OUT	AxesGroup			-	-	AXES_GROUP_REF
INPUT	Execute			-	TRUE/FALSE	BOOL
	Position			-	/ /0	ARRAY[0..1] OF LREAL
	Velocity			-		LREAL
	Acceleration			-		LREAL
	Deceleration			-		LREAL
	Jerk	0 S		0	0/	LREAL
	CoordSystem	0		0	0: ACS	MC_COORD_SYSTEM
	BufferMode	1 2 3 4 5		1	1: Buffered 2: BlendingLow 3: BlendingPrevious 4: BlendingNext 5: BlendingHigh	MC_BUFFER_MODE
	TransitionMode			0	0 TMNone	MC_TRANSITION_MODE

		0				
	TransitionParameter			0	-	ARRAY[0..1] OF LREAL
OUTPUT	Done			FALSE	TRUE/FALSE	BOOL
	Busy			FALSE	TRUE/FALSE	BOOL
	Active	TRUE		FALSE	TRUE/FALSE	BOOL
	CommandAborted			FALSE	TRUE/FALSE	BOOL
	Error			FALSE	TRUE/FALSE	BOOL
	ErrorID			0	-	SMC_ERROR

7.4.2

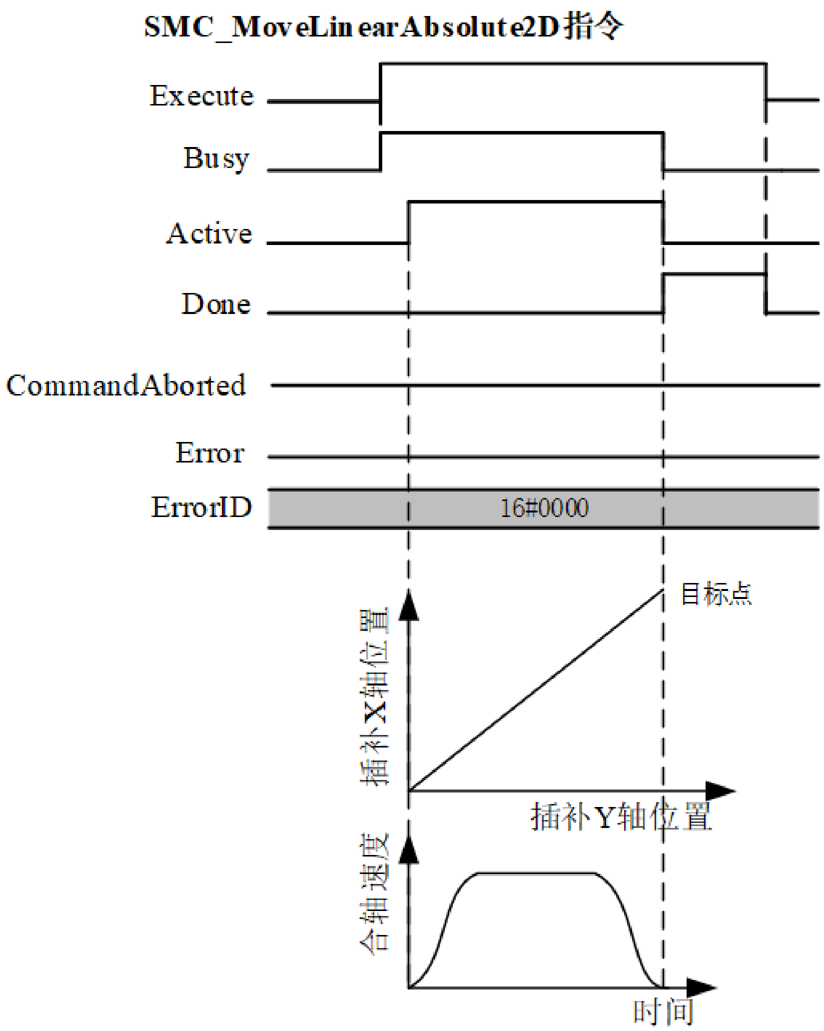
2

1 2

2

3 SMC_MoveLinearRelative2D( )

1

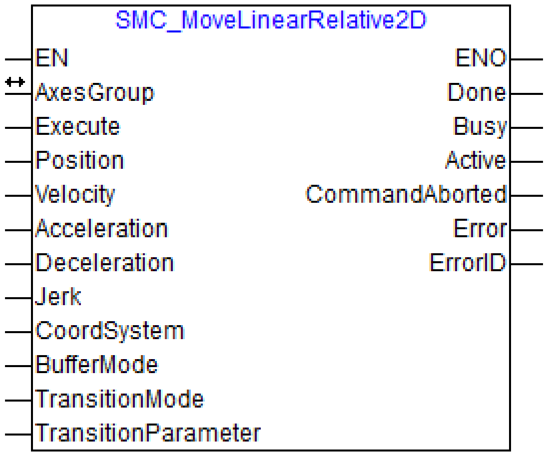


2

0 Unused

ID Error=True ErrorID

Execute



7.5.1

IN_OUT	AxesGroup			-	-	AXES_GROUP_REF
INPUT	Execute			-	TRUE/FALSE	BOOL
	Position			-	/ /0	ARRAY[0..1] OF LREAL
	Velocity			-		LREAL
	Acceleration			-		LREAL
	Deceleration			-		LREAL
	Jerk	0 S		0	0/	LREAL
	CoordSystem	0		0	0: ACS	MC_COORD_SYSTEM
	BufferMode	1 2 3 4 5		1	1: Buffered 2: BlendingLow 3: BlendingPrevious 4: BlendingNext 5: BlendingHigh	MC_BUFFER_MODE

	TransitionMode	0		0	0 TMNone	MC_TRANSITION_MODE
	TransitionParameter			0	-	ARRAY[0..1] OF LREAL
OUTPUT	Done			FALSE	TRUE/FALSE	BOOL
	Busy			FALSE	TRUE/FALSE	BOOL
	Active	TRUE		FALSE	TRUE/FALSE	BOOL
	CommandAborted			FALSE	TRUE/FALSE	BOOL
	Error			FALSE	TRUE/FALSE	BOOL
	ErrorID			0	-	SMC_ERROR

7.5.2

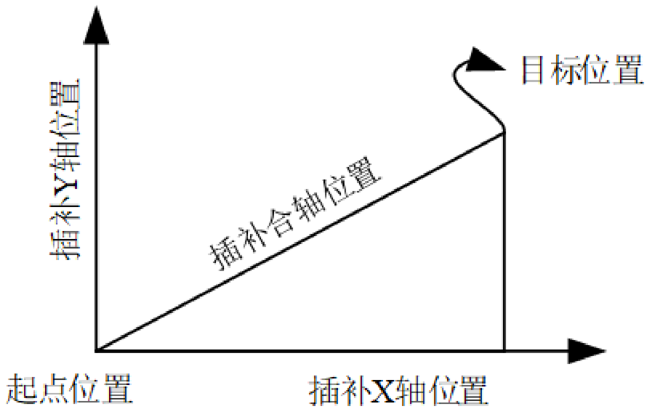
2

1 AxesGroup AXES_GROUP_REF

2

3

4



Execute AxesGroup

ID Position Velocity Acceleration Deceleration Jerk

Velocity Acceleration

Deceleration Jerk

BufferMode ( )

BufferMode ( ) BufferMode 1 Buffered 2

BlendingLow 3: BlendingPrevious 4: BlendingNext 5: BlendingHigh

BufferMode 1 Buffered

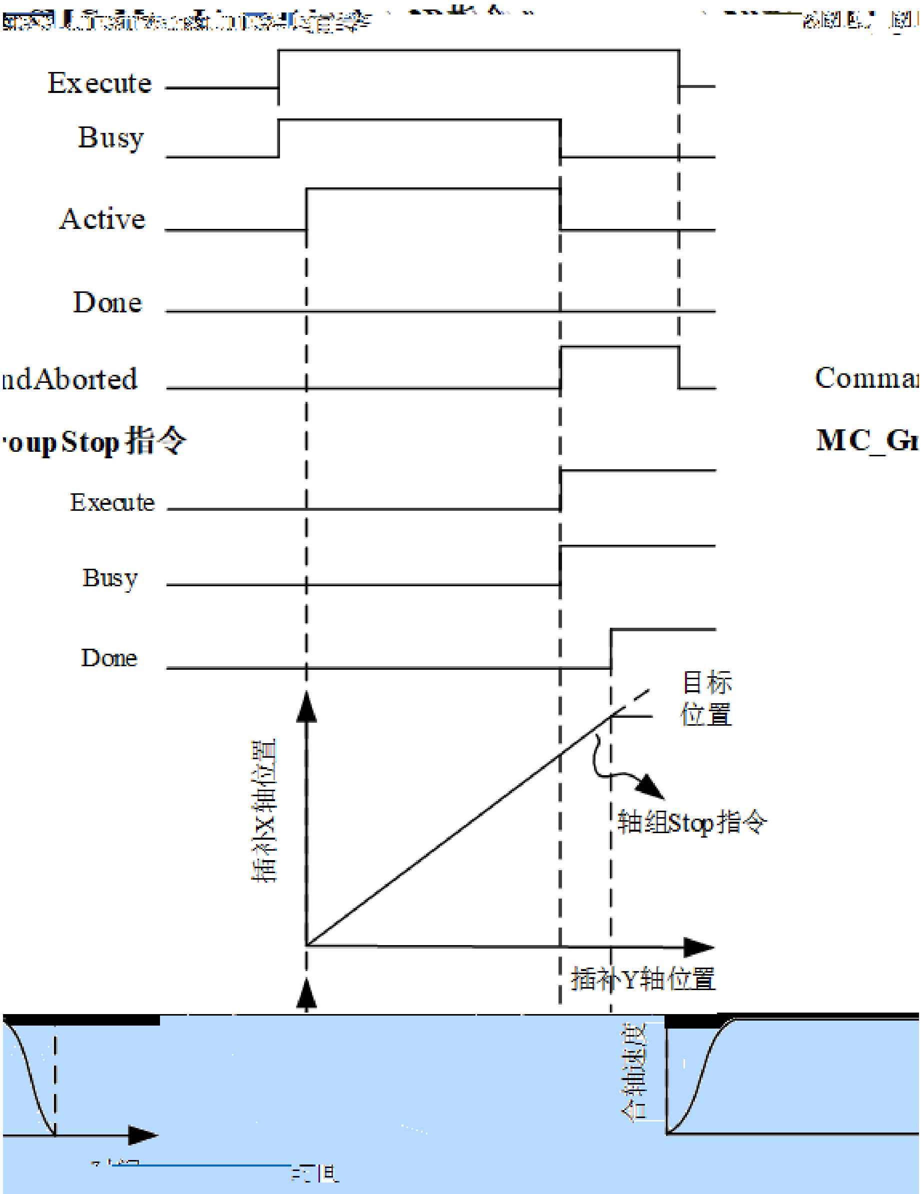
BufferMode 2 BlendingLow BufferMode 3:

BlendingPrevious BufferMode 4: BlendingNext

BufferMode 5: BlendingHigh

BufferMode

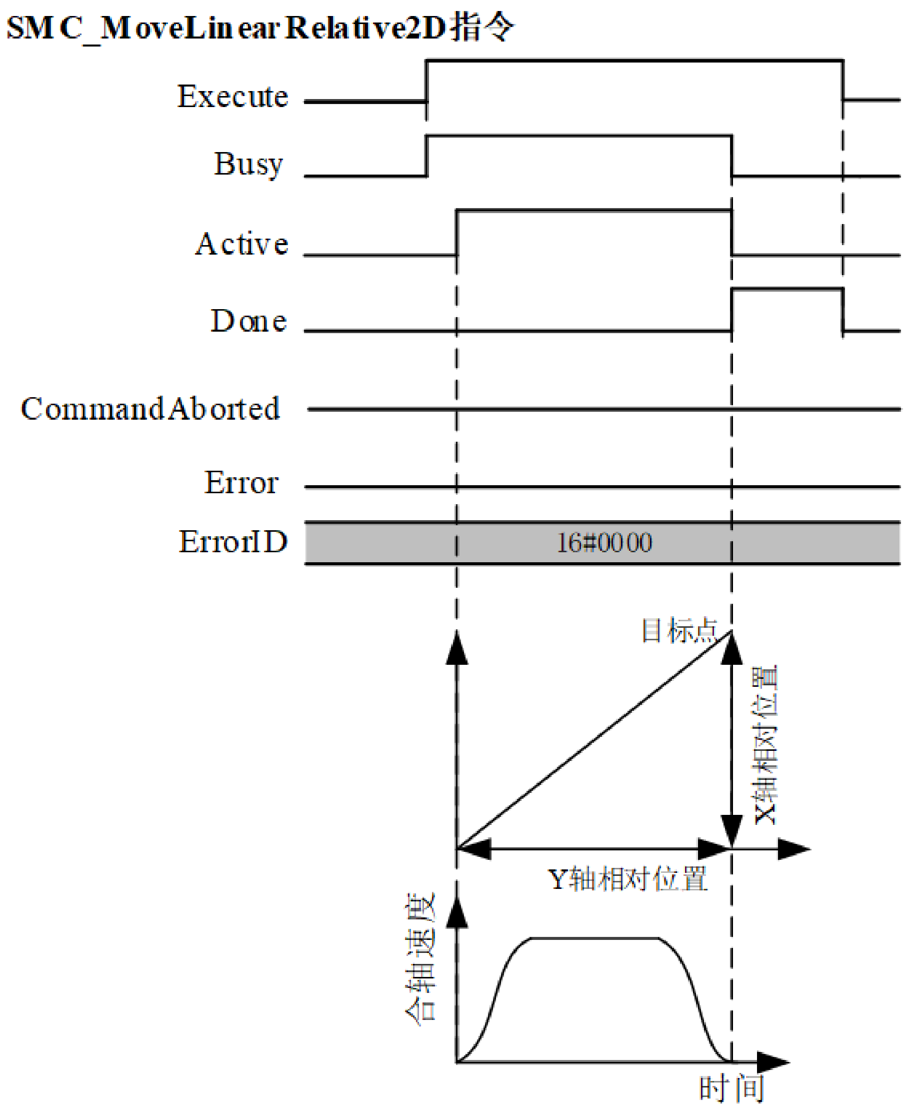
MC_GroupStop



CoordSystem( )

(ACS)

1



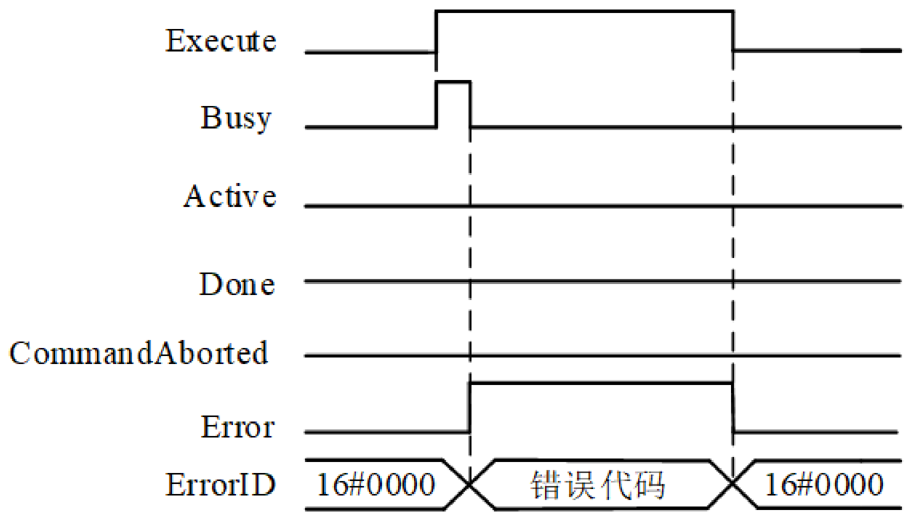
2

0 Unused

ID Error=True ErrorID

Execute

SMC_MoveLinearRelative2D指令



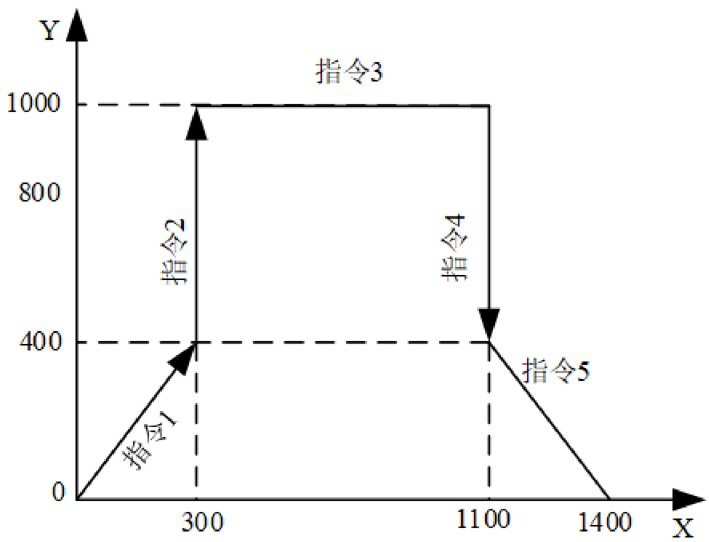
- Execute

7.5.3

5

BufferMode Buffered

1 Active TRUE 2-5



1 300,400 2 300,1000

3 1100,1000 4 1100,400

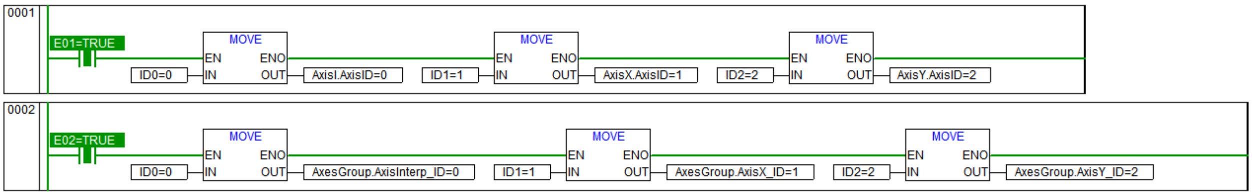
3 1400, 0

100 200 5000 Buffered

AxesGroup	AXES_GROUP_REF	-	
AxisI	AXIS_REF	-	
AxisX	AXIS_REF	-	X
AxisY	AXIS_REF	-	Y
StartProg	BOOL	FALSE	TRUE
Power0_Status	BOOL	FALSE	TRUE
Power1_Status	BOOL	FALSE	X X TRUE
Power2_Status	BOOL	FALSE	Y Y TRUE
GrpEnableStatus	BOOL	TRUE	TRUE
GrpEnable1_Exe	BOOL	FALSE	
AxisHomed	BOOL	FALSE	TRUE
Rel2D1_Exe	BOOL	FALSE	
Rel2D1_Active	BOOL	FALSE	Active TRUE

1 ID

ID ID 0001 0002

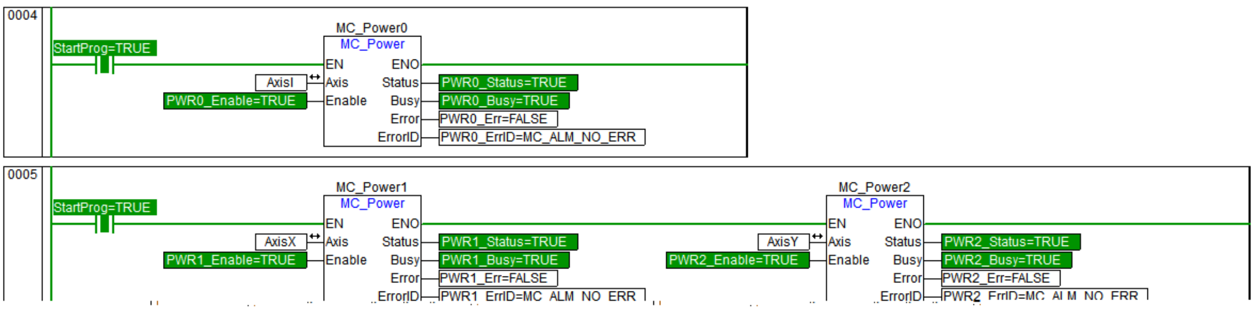


2 SMC_AxisInit 0

Virtual 50: EtherCAT_Position [SMC_AxisInit]

3 StartProg =TRUE MC_Power

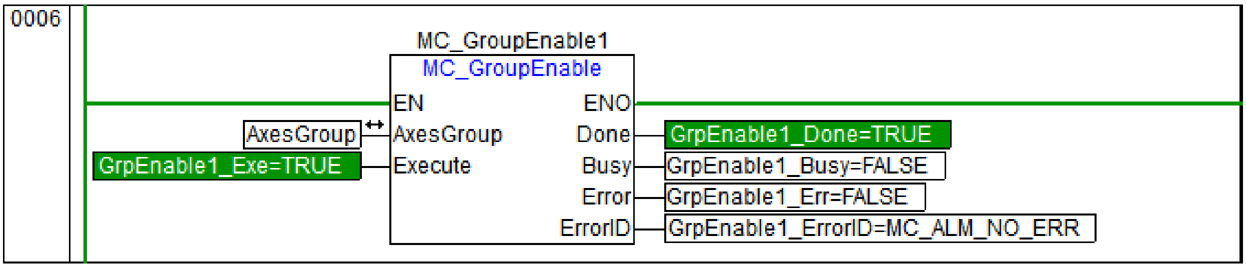
0004 ~ 0005

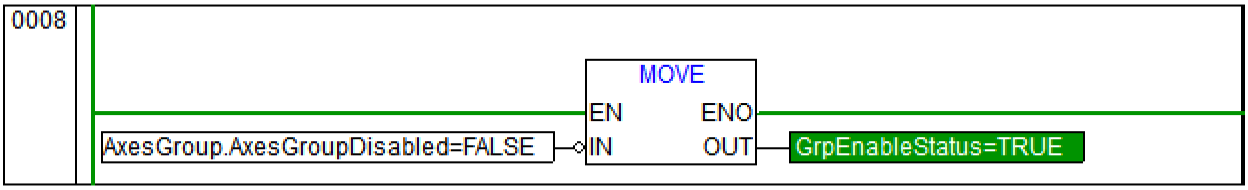


4 MC_GroupEnable AxesGroupDisabled FALSE

0006 AxesGroupDisabled

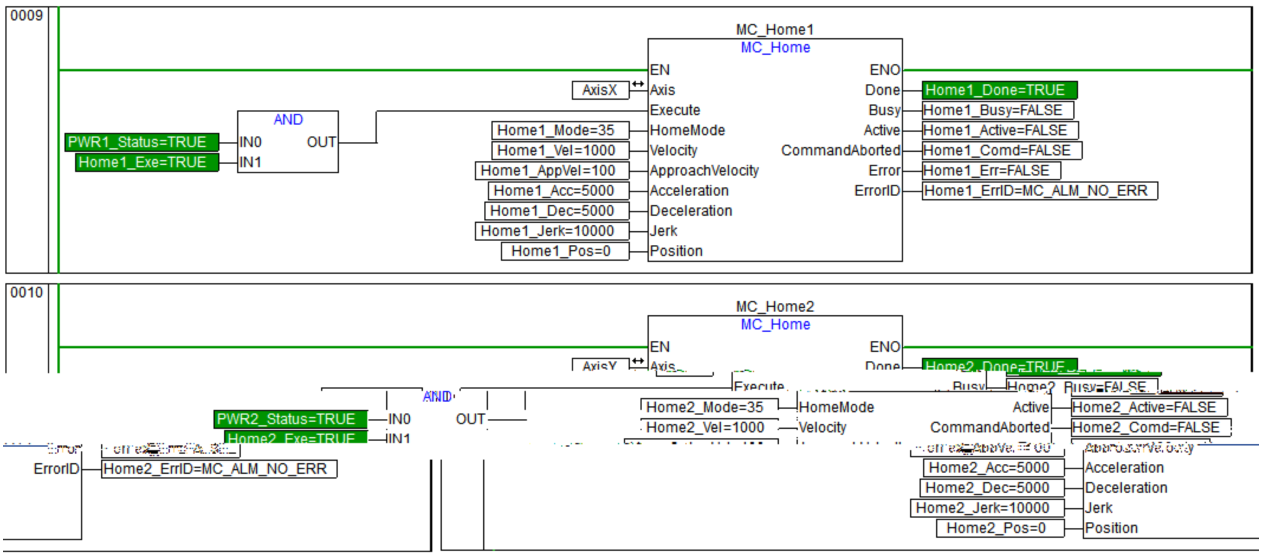
GrpEnableStatus

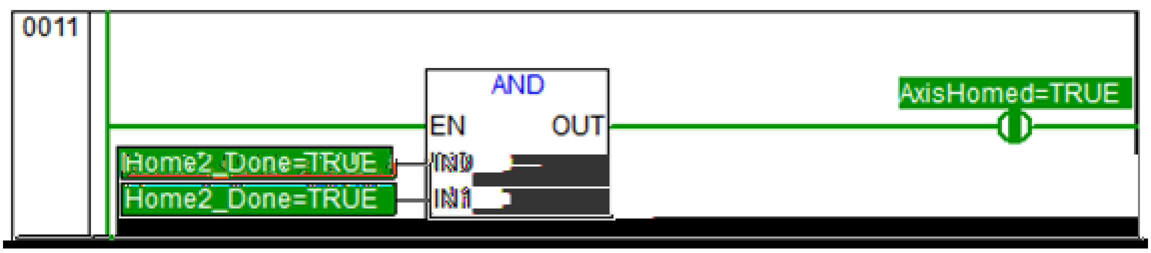




5

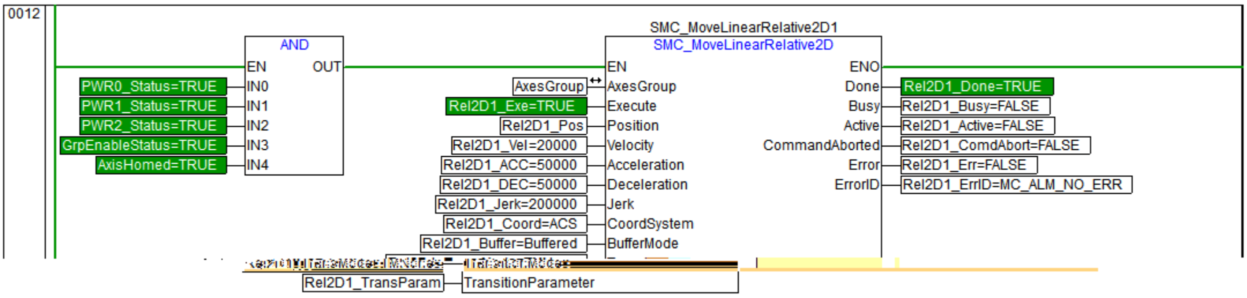
AxisHomed TRUE 0009-0011

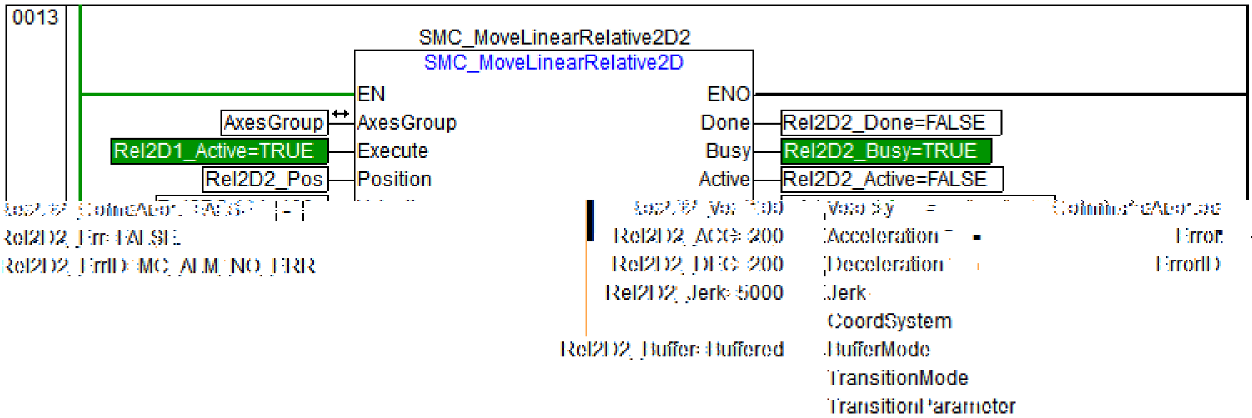


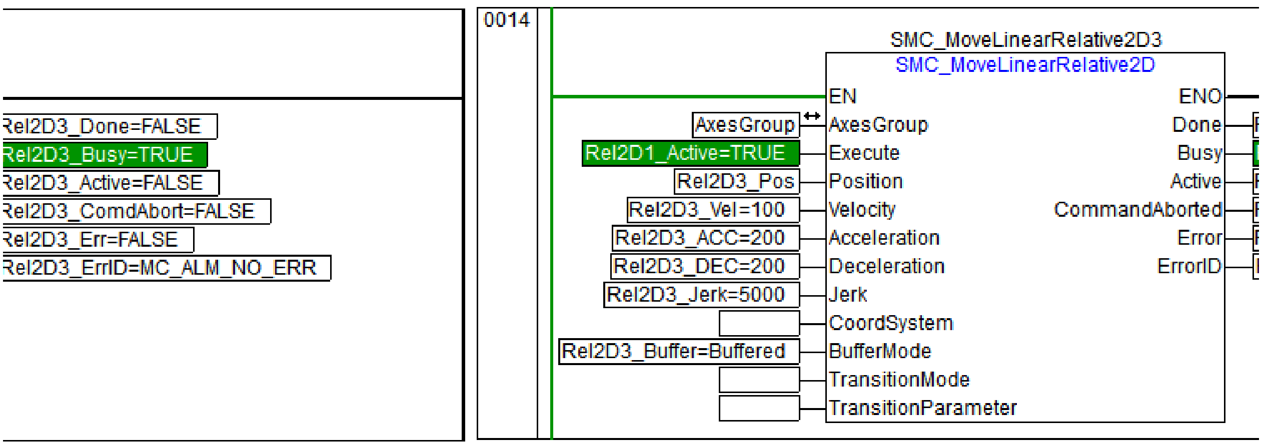


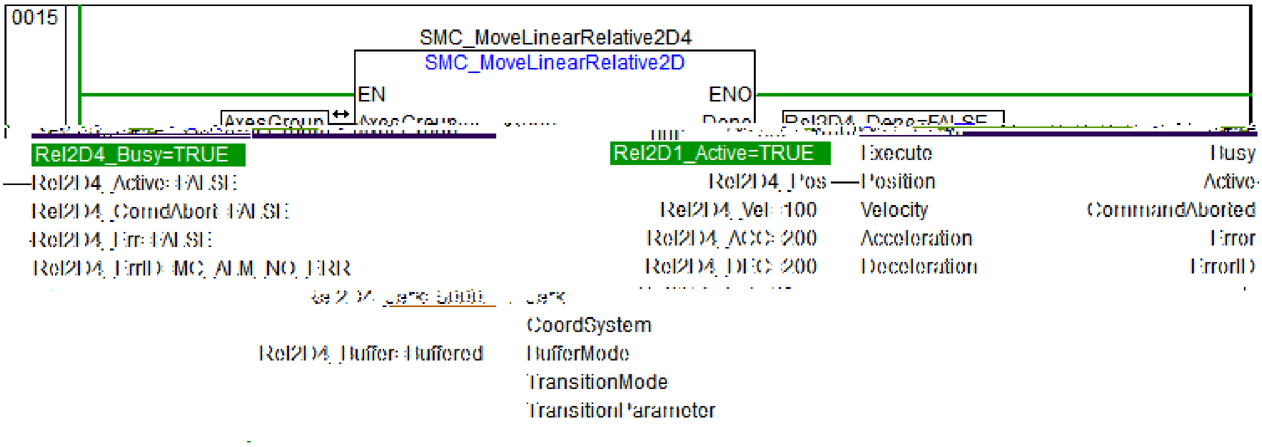
6 0011 1 1 Rel2D1_Active

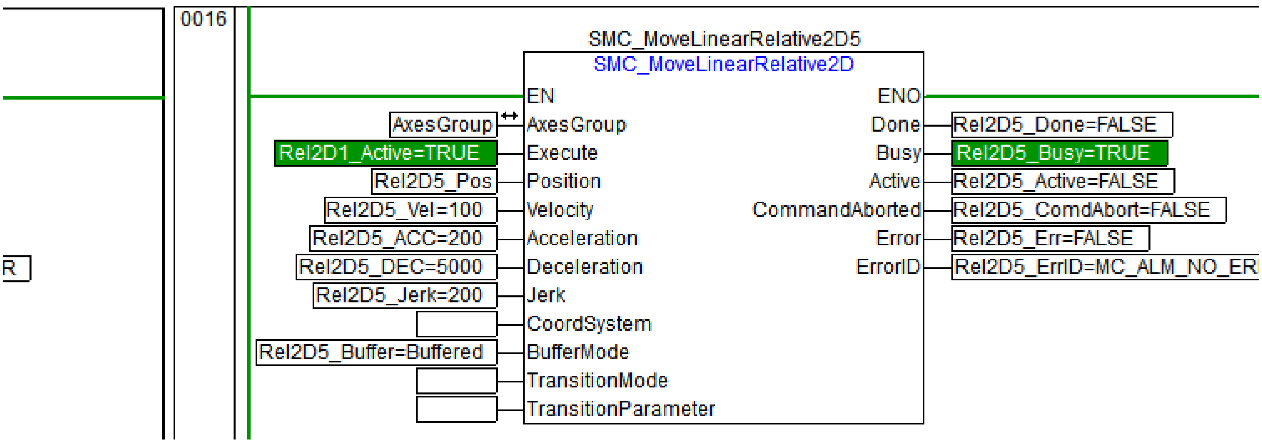
TRUE 2-5 0012-0016


0012
AND
EN
OUT
PWR0_Status=TRUE
PWR1_Status=TRUE
PWR2_Status=TRUE
GrpEnableStatus=TRUE
AxisHomed=TRUE
IN0
IN1
IN2
IN3
IN4
SMC_MoveLinearRelative2D1
SMC_MoveLinearRelative2D
AxesGroup
Rel2D1_Exe=TRUE
Rel2D1_Pos
Rel2D1_Vel=20000
Rel2D1_ACC=50000
Rel2D1_DEC=50000
Rel2D1_Jerk=200000
Rel2D1_Coord=ACS
Rel2D1_Buffer=Buffered
Rel2D1_TransParam
EN
AxesGroup
Execute
Position
Velocity
Acceleration
Deceleration
Jerk
CoordSystem
BufferMode
TransitionParameter
ENO
Done
Busy
Active
CommandAborted
Error
ErrorID
Rel2D1_Done=TRUE
Rel2D1_Busy=FALSE
Rel2D1_Active=FALSE
Rel2D1_ComdAbort=FALSE
Rel2D1_Err=FALSE
Rel2D1_ErrID=MC_ALM_NO_ERR


0013
SMC_MoveLinearRelative2D2
SMC_MoveLinearRelative2D
AxesGroup
Rel2D1_Active=TRUE
Rel2D2_Pos
EN
AxesGroup
Execute
Position
ENO
Done
Busy
Active
Rel2D2_Done=FALSE
Rel2D2_Busy=TRUE
Rel2D2_Active=FALSE
Rel2D2_Jerk=5000
Jerk
CoordSystem
TransitionMode


Rel2D3_Done=FALSE
Rel2D3_Busy=TRUE
Rel2D3_Active=FALSE
Rel2D3_ComdAbort=FALSE
Rel2D3_Err=FALSE
Rel2D3_ErrID=MC_ALM_NO_ERR
0014
SMC_MoveLinearRelative2D3
SMC_MoveLinearRelative2D
AxesGroup
Rel2D1_Active=TRUE
Rel2D3_Pos
Rel2D3_Vel=100
Rel2D3_ACC=200
Rel2D3_DEC=200
Rel2D3_Jerk=5000
Rel2D3_Buffer=Buffered
EN
AxesGroup
Execute
Position
Velocity
Acceleration
Deceleration
Jerk
CoordSystem
BufferMode
TransitionMode
TransitionParameter
ENO
Done
Busy
Active
CommandAborted
Error
ErrorID





ST

AxesGroup	AXES_GROUP_REF	-	
AxisI	AXIS_REF	-	
AxisX	AXIS_REF	-	X
AxisY	AXIS_REF	-	Y
StartProg	BOOL	FALSE	TRUE
Power0_Status	BOOL	FALSE	TRUE
Power1_Status	BOOL	FALSE	X X TRUE
Power2_Status	BOOL	FALSE	Y Y TRUE
GrpEnableStatus	BOOL	TRUE	TRUE
GrpEnable1_Exe	BOOL	FALSE	
AxisHomed	BOOL	FALSE	TRUE
InitFlag	BOOL	FALSE	TRUE
Rel2D1_Exe	BOOL	FALSE	

Rel2D1_Active	BOOL	FALSE	Active	TRUE

1 ID AxisI AxisX AxisY ID

```
IF Switch1 THEN
   AxisI.AxisID:=0;
   AxisX.AxisID:=1;
   AxisY.AxisID:=2;
   AxesGroup.AxisInterp_ID:=AxisI.AxisID;
   AxesGroup.AxisX_ID:=AxisX.AxisID;
   AxesGroup.AxisY_ID:=AxisY.AxisID;
END_IF
```

2

```
IF FALSE = InitFlag THEN
   (*                    1

   Rel2D1_Pos[0]:=300;
   Rel2D1_Pos[1]:=400;
   Rel2D1_Vel:=100;
   Rel2D1_ACC:=200;
   Rel2D1_DEC:=200;
   Rel2D1_Jerk:=5000;
   Rel2D1_Buffer:=Buffered;
   (*                    2

   Rel2D2_Pos[0]:= 0;
   Rel2D2_Pos[1]:=400;
   Rel2D2_Vel:=100;
   Rel2D2_ACC:=200;
   Rel2D2_DEC:=200;
   Rel2D2_Jerk:=5000;
   Rel2D2_Buffer:=Buffered;
   (*                    3

   Rel2D3_Pos[0]:=300;
   Rel2D3_Pos[1]:=400;
   Rel2D3_Vel:=100;
   Rel2D3_ACC:=200;
   Rel2D3_DEC:=200;
```

```
   Rel2D3_Jerk:=5000;
   Rel2D3_Buffer:=Buffered;
   (*                              4
   Rel2D4_Pos[0]:= 0;
   Rel2D4_Pos[1]:=400;
   Rel2D4_Vel:=100;
   Rel2D4_ACC:=200;
   Rel2D4_DEC:=200;
   Rel2D4_Jerk:=5000;
   Rel2D4_Buffer:=Buffered;
   (*                              5
   Rel2D5_Pos[0]:=300;
   Rel2D5_Pos[1]:=400;
   Rel2D5_Vel:=100;
   Rel2D5_ACC:=200;
   Rel2D5_DEC:=200;
   Rel2D5_Jerk:=5000;
   Rel2D5_Buffer:=Buffered;
   InitFlag:=TRUE;
END_IF
```

3 SMC_AxisInit 0 Virtual

AxisX/AxisY 50 EtherCAT_Position SMC_AxisInit

4 MC_Power

```
IF StartPro THEN
   MC_Power0(                          (*          *)
   Enable := PWR0_Enable,
   Axis := AxisI,
   Status=> PWR0_Status,
   Busy=> PWR0_Busy,
Error=> PWR0_Err,
ErrorID=> PWR0_ErrID);
MC_Power1(                          (*      AxisX      *)
   Enable := PWR1_Enable,
```

```
   Axis := AxisX,
   Status=> PWR1_Status,
   Busy=> PWR1_Busy,
Error=> PWR1_Err,
ErrorID=> PWR1_ErrID);
MC_Power2(                              (*       AxisY        *)
   Enable := PWR2_Enable,
   Axis := AxisY,
   Status=> PWR2_Status,
   Busy=> PWR2_Busy,
   Error=> PWR2_Err,
   ErrorID=> PWR2_ErrID);
END_IF
```

5

```
MC_GroupEnable1(
Execute := GrpEnable_Exe,
AxesGroup := AxesGroup,
Done=> GrpEnable_Done,
Busy=> GrpEnable_Busy,
Error=> GrpEnable_Err,
ErrorID=> GrpEnable_ErrorID);
GrpEnableStatus:= AxesGroup. AxesGroupDisabled;
```

6 MC_Home

```
IF  PWR1_Status THEN
   MC_Home1(
   Execute := Home1_Exe,
   HomeMode := Home1_Mode,
   Velocity := Home1_Vel,
   ApproachVelocity := Home1_AppVel,
   Acceleration := Home1_Acc,
   Deceleration := Home1_Dec,
   Jerk := Home1_Jerk,
   Position := Home1_Pos,
   Axis := AxisX,
   Done=> Home1_Done,
   Busy=> Home1_Busy,
```

```
   Active=> Home1_Active,
   CommandAborted=> Home1_Comd,
   Error=> Home1_Err,
   ErrorID=> Home1_ErrID);
   MC_Home2(
   Execute := Home2_Exe,
   HomeMode := Home2_Mode,
   Velocity := Home2_Vel,
   ApproachVelocity := Home2_AppVel,
   Acceleration := Home2_Acc,
   Deceleration := Home2_Dec,
   Jerk := Home2_Jerk,
   Position := Home2_Pos,
   Axis := AxisY,
   Done=> Home2_Done,
   Busy=> Home2_Busy,
   Active=> Home2_Active,
   CommandAborted=> Home2_Comd,
   Error=> Home2_Err,
   ErrorID=> Home2_ErrID);
END_IF
IF  Home1_Done  AND Home2_Done THEN
   AxisHomed:=TRUE;
END_IF
```

7 1 1 Rel2D1_Active TRUE

2-5

```
IF PWR0_Status  AND PWR1_Status  AND PWR2_Status  AND GrpEnableStatus AND AxisHomed THEN
   SMC_MoveLinearRelative2D1(            (*              1*)
   Execute := Rel2D1_Exe,
   Position := Rel2D1_Pos,
   Velocity := Rel2D1_Vel,
   Acceleration := Rel2D1_ACC,
   Deceleration := Rel2D1_DEC,
   Jerk := Rel2D1_Jerk,
   CoordSystem := Rel2D1_Coord,
   BufferMode := Rel2D1_Buffer,
```

```
TransitionMode := Rel2D1_Trans,
TransitionParameter := Rel2D1_Param,
AxesGroup := AxesGroup,
Done=> Rel2D1_Done,
Busy=> Rel2D1_Busy,
Active=> Rel2D1_Active,
CommandAborted=> Rel2D1_ComdAbort,
Error=> Rel2D1_Err,
ErrorID=> Rel2D1_ErrID);
SMC_MoveLinearRelative2D2(                (*            2*)
Execute := Rel2D1_Active,
Position := Rel2D2_Pos,
Velocity := Rel2D2_Vel,
Acceleration := Rel2D2_ACC,
Deceleration := Rel2D2_DEC,
Jerk := Rel2D2_Jerk,
CoordSystem := Rel2D2_Coord,
BufferMode := Rel2D2_Buffer,
TransitionMode := Rel2D2_Trans,
TransitionParameter := Rel2D2_Param,
AxesGroup := AxesGroup,
Done=> Rel2D2_Done,
Busy=> Rel2D2_Busy,
Active=> Rel2D2_Active,
CommandAborted=> Rel2D2_ComdAbort,
Error=> Rel2D2_Err,
ErrorID=> Rel2D2_ErrID);
SMC_MoveLinearRelative2D3(          (*            3*)
Execute := Rel2D1_Active,
Position := Rel2D3_Pos,
Velocity := Rel2D3_Vel,
Acceleration := Rel2D3_ACC,
Deceleration := Rel2D3_DEC,
Jerk := Rel2D3_Jerk,
CoordSystem := Rel2D3_Coord,
BufferMode := Rel2D3_Buffer,
TransitionMode := Rel2D3_Trans,
```

```
TransitionParameter := Rel2D3_Param,
AxesGroup := AxesGroup,
```

```
  AxesGroup := AxesGroup,
  Done=> Rel2D5_Done,
  Busy=> Rel2D5_Busy,
  Active=> Rel2D5_Active,
  CommandAborted=> Rel2D5_ComdAbort,
  Error=> Rel2D5_Err,
  ErrorID=> Rel2D5_ErrID);
END_IF
```

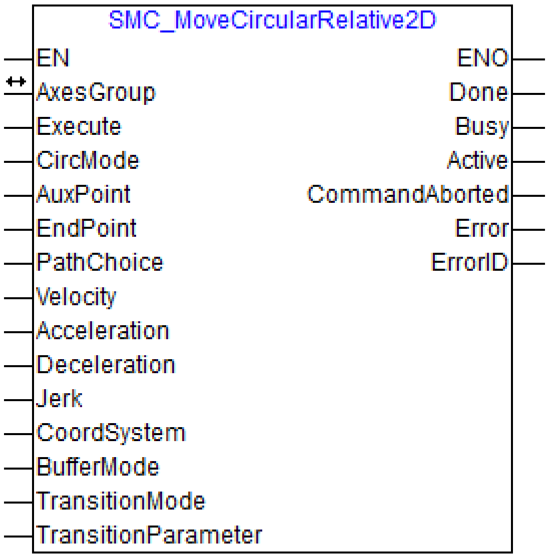
7.5.4

Positon 0

BufferMode ( ) 0 Aborting

7.6 SMC_MoveCircularRelative2D 2

2



7.6.1

IN_OUT	AxesGroup			-	-	AXES_GROUP_REF %2BBDC 9 02>-1B9

	Jerk	0 S		0	0/	LREAL
	CoordSystem	0		0	0: ACS	MC_COORD_SYSTEM
	BufferMode	1 2 3 4 5		1	1: Buffered 2: BlendingLow 3: BlendingPrevious 4: BlendingNext 5: BlendingHigh	MC_BUFFER_MODE
	TransitionMode	0		0	0 TMNone	MC_TRANSITION_MODE
	TransitionParameter			0	-	ARRAY[0..1] OF LREAL
OUTPUT	Done			FALSE	TRUE/FALSE	BOOL
	Busy			FALSE	TRUE/FALSE	BOOL
	Active	TRUE		FALSE	TRUE/FALSE	BOOL
	CommandAborted			FALSE	TRUE/FALSE	BOOL
	Error			FALSE	TRUE/FALSE	BOOL
	ErrorID			0	-	SMC_ERROR

7.6.2

2

1 AxesGroup AXES_GROUP_REF

2

3

2 2

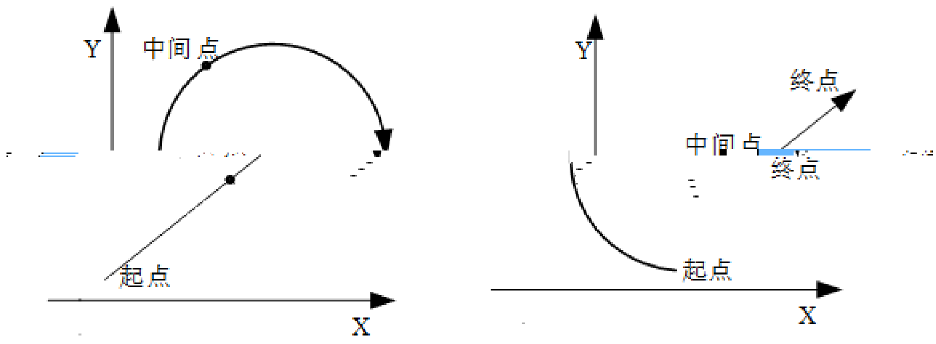
“ ” AuxPoint

EndPoint

Execute AxesGroup

ID Position Velocity Acceleration ضء ˜

ɣ



b / CircMode=CENTER

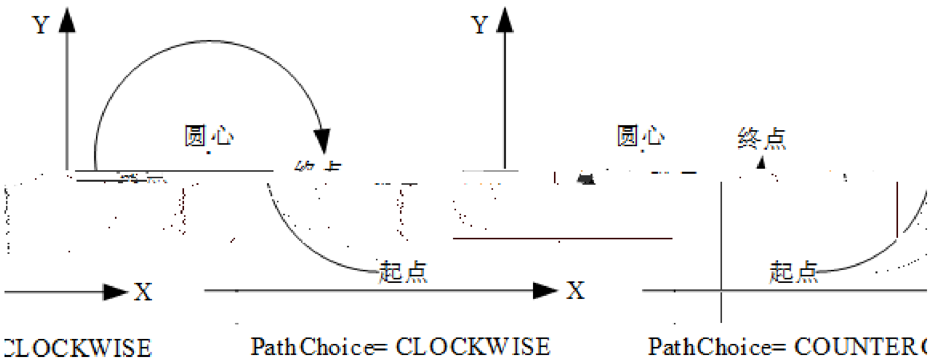
AuxPoint(X, Y) EndPoint(X, Y)

PathChoice( )

PathChoice

CircMode CENTER PathChoice 0 CLOCKWISE

1 COUNTERCLOCKWISE AuxPoint EndPoint



BufferMode ( )

2 BufferMode 1 Buffered 2 BlendingLow 3: BlendingPrevious 4: BlendingNext 5: BlendingHigh

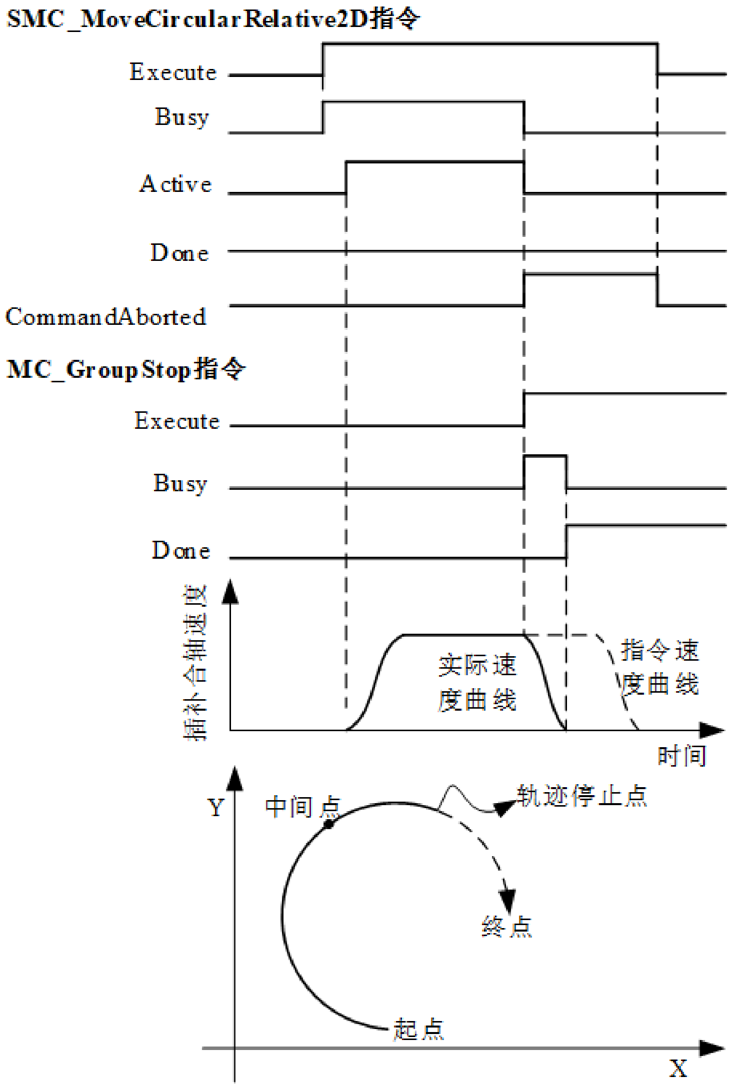
BufferMode 1 Buffered

BufferMode 2 BlendingLow BufferMode 3: BlendingPrevious BufferMode 4: BlendingNext

BufferMode 5: BlendingHigh

BufferMode

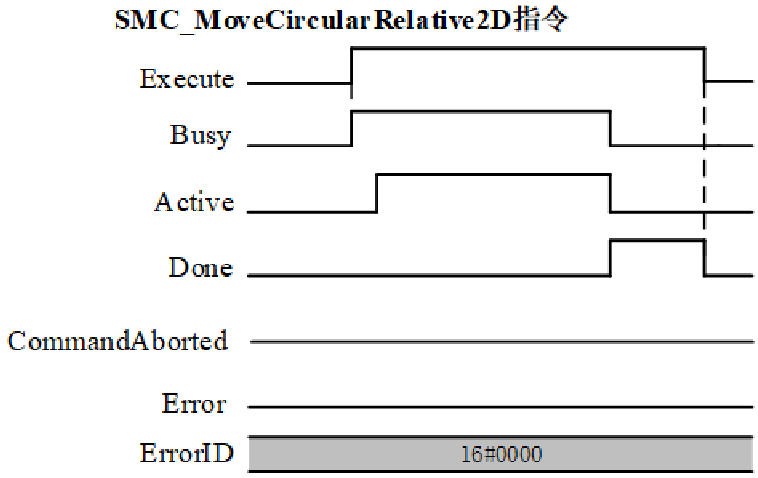
MC_GroupStop



CoordSystem( )

(ACS)

1

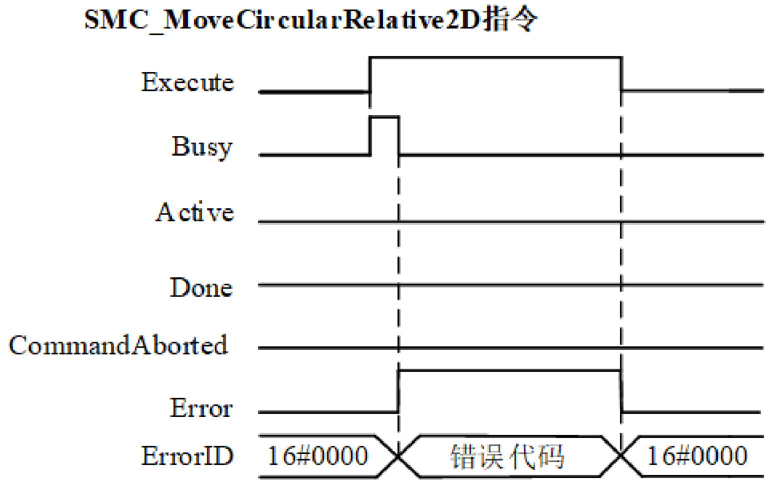


2

0 Unused

ID Error=True ErrorID

Execute

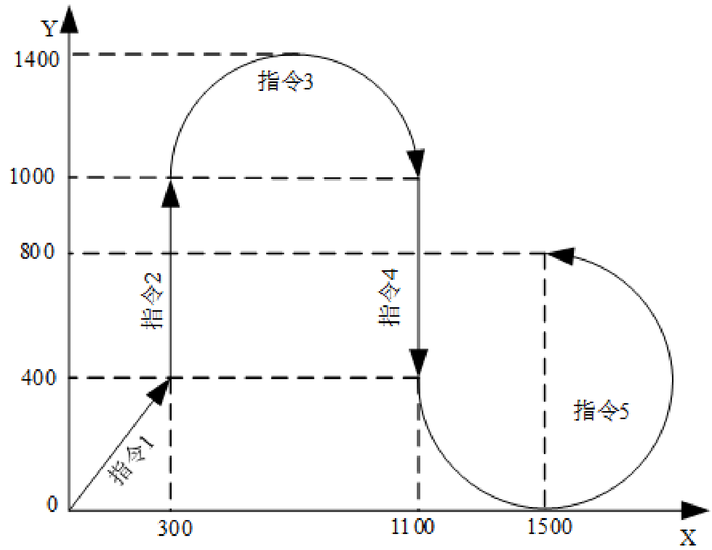


- Execute

7.6.3

BufferMode Buffered

1 Active TRUE 2-5



1 300,400 2 300,1000

3 1100,1000 4 1100,400

5 1500,800

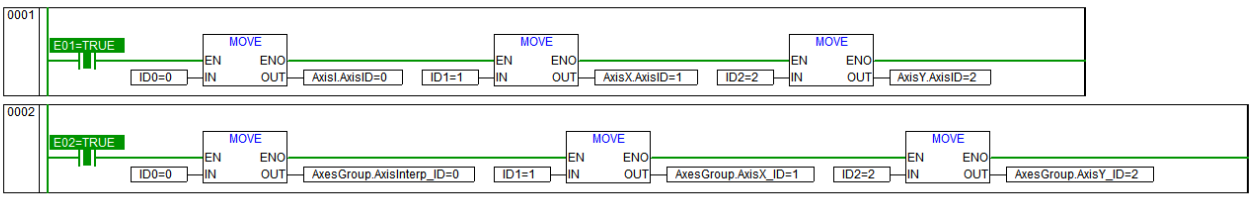
100 200 5000 Buffered

AxesGroup	AXES_GROUP_REF	-	
AxisI	AXIS_REF	-	
AxisX	AXIS_REF	-	X
AxisY	AXIS_REF	-	Y
StartProg	BOOL	FALSE	TRUE

Power1_Status	BOOL	FALSE	X X TRUE
Power2_Status	BOOL	FALSE	Y Y TRUE
Power0_Status	BOOL	FALSE	TRUE
GrpEnableStatus	BOOL	TRUE	TRUE
GrpEnable1_Exe	BOOL	FALSE	
AxisHomed	BOOL	FALSE	TRUE
Rel2D1_Exe	BOOL	FALSE	
Rel2D1_Active	BOOL	FALSE	Active TRUE

1 ID

ID ID 0001 0002

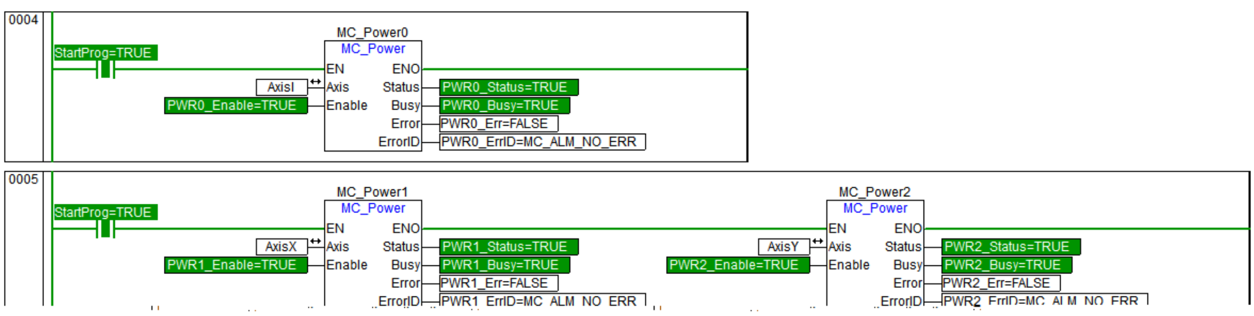


2 SMC_AxisInit 0

Virtual 50: EtherCAT_Position SMC_AxisInit

3 StartProg =TRUE MC_Power

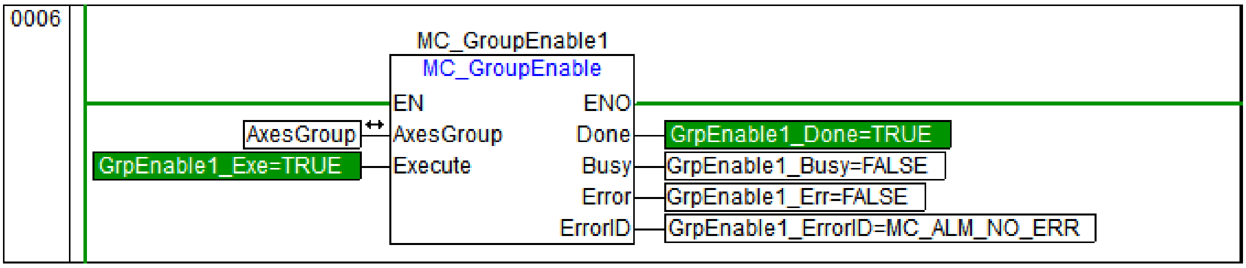
0004 ~ 0005

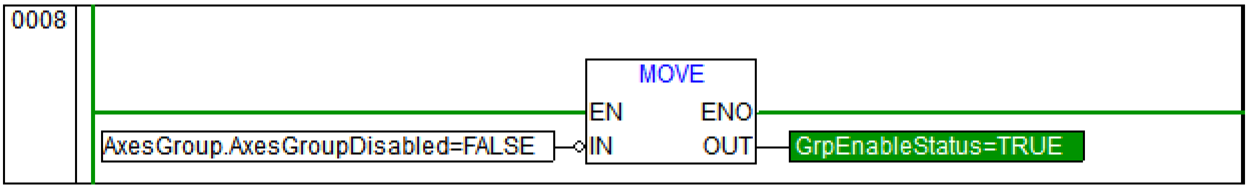


4 MC_GroupEnable AxesGroupDisabled FALSE

0006 AxesGroupDisabled

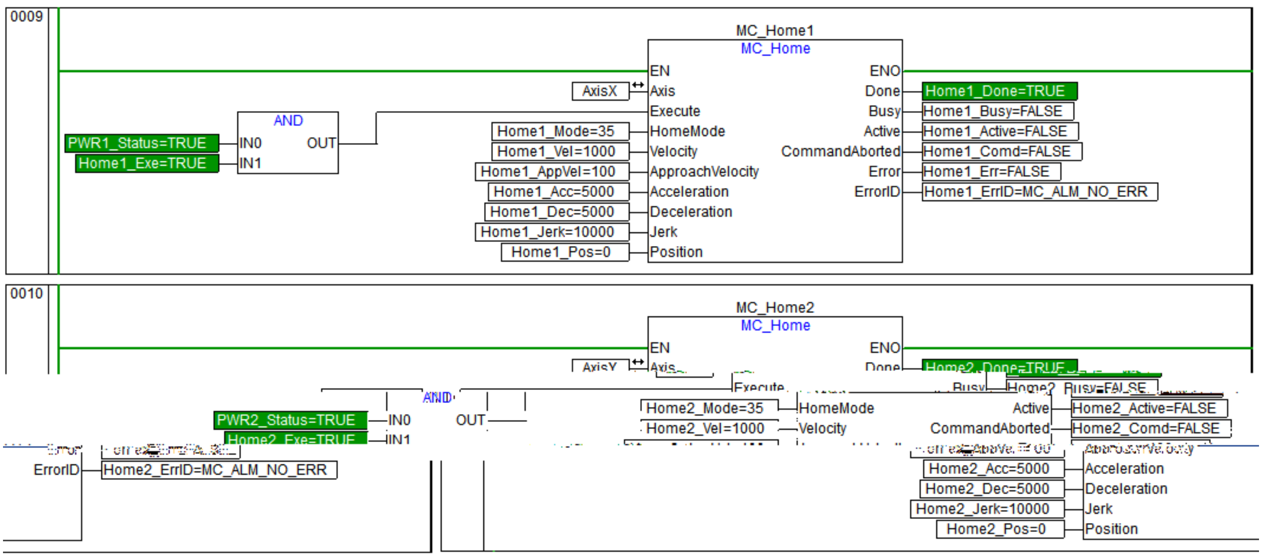
GrpEnableStatus 0008

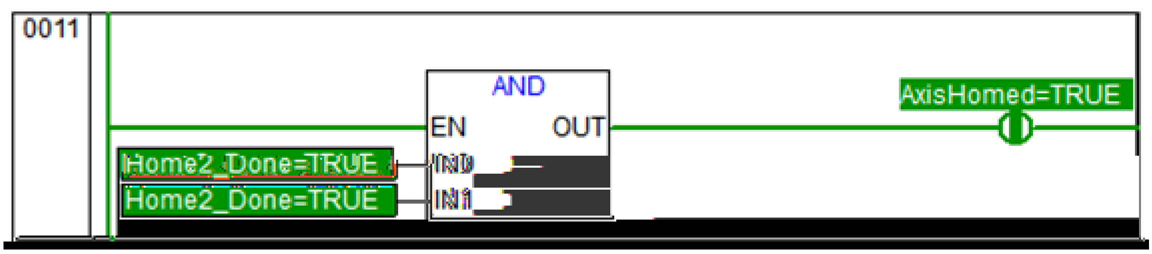




5

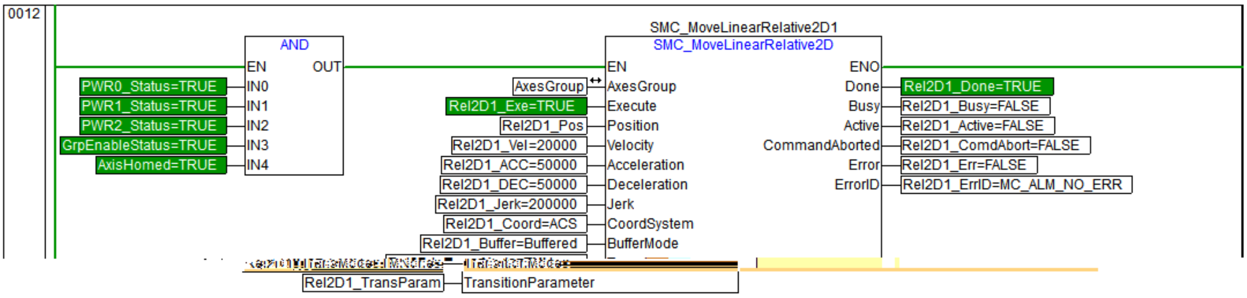
AxisHomed TRUE 0009-0011

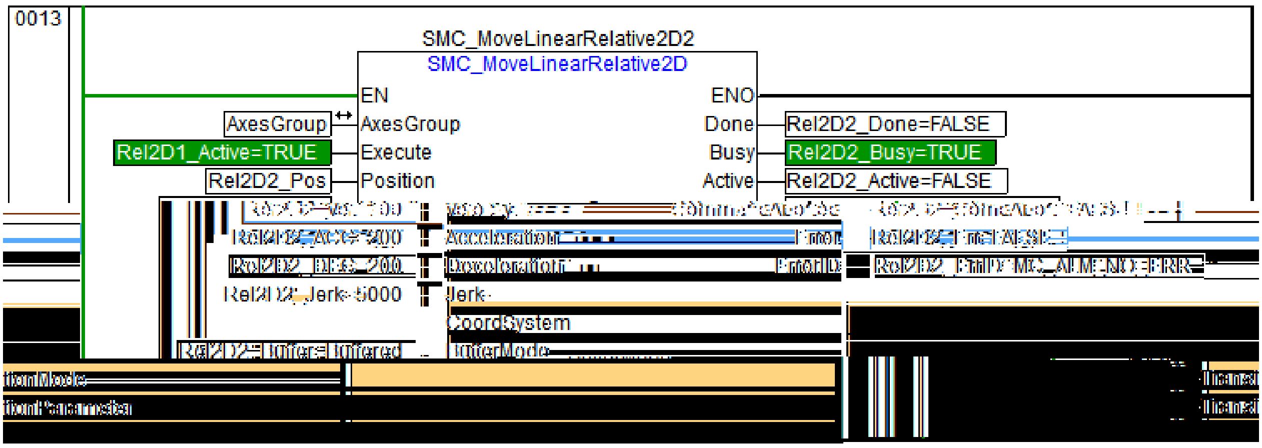


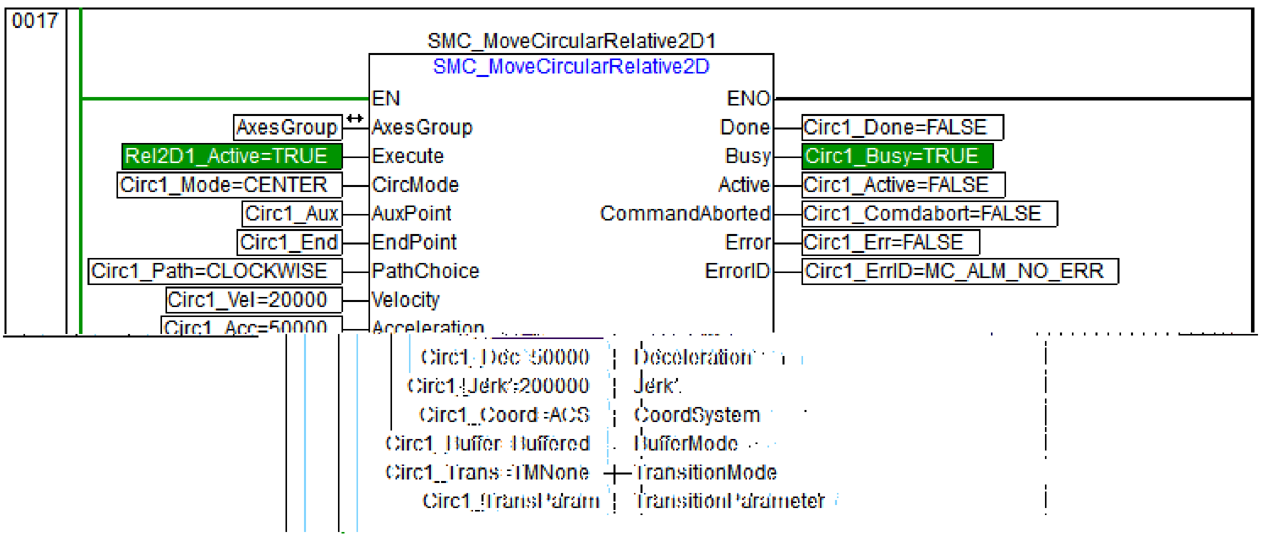


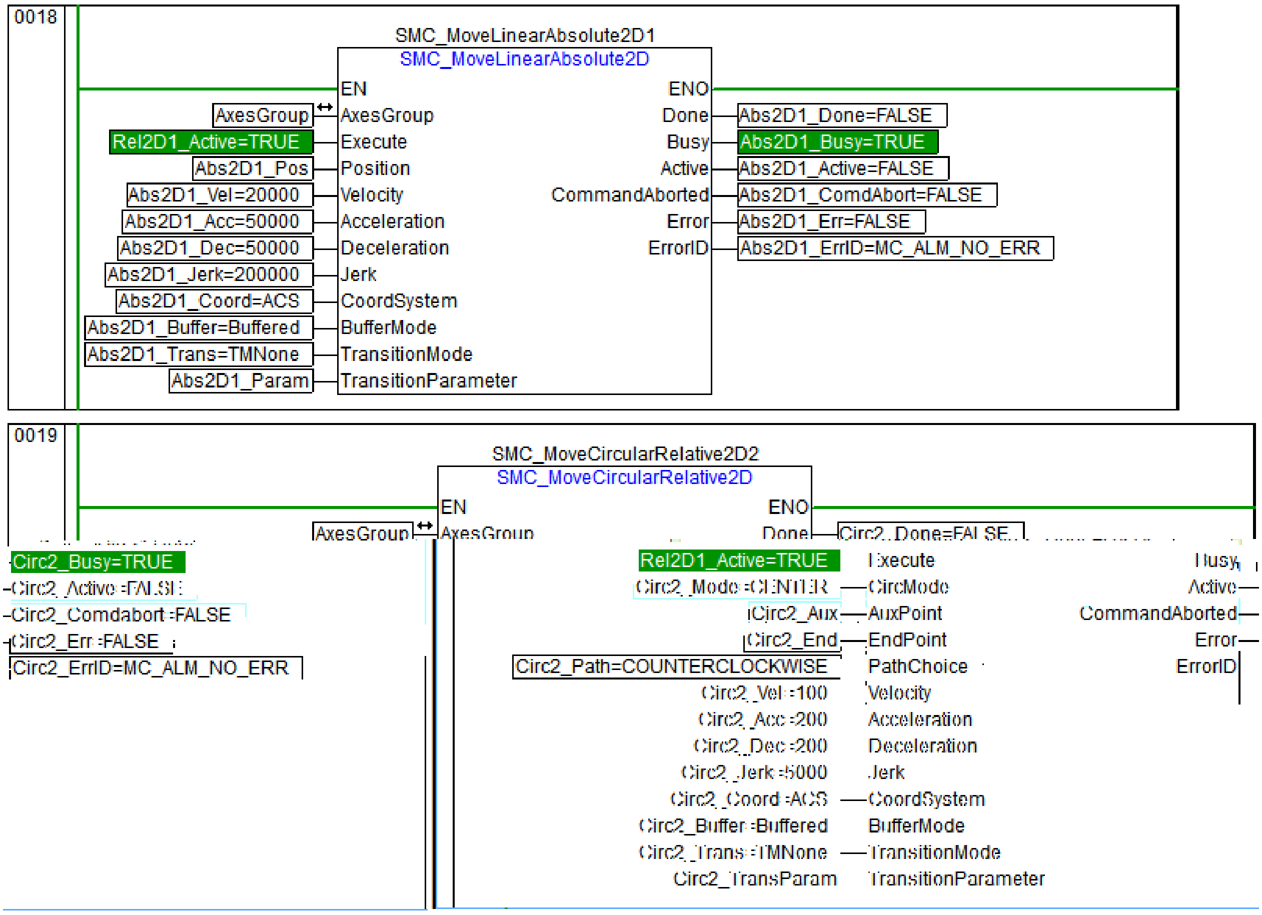
6 0011 1 1 Rel2D1_Active

TRUE 2-5 0012-0013/0017-0019


0012
AND
EN
OUT
PWR0_Status=TRUE
PWR1_Status=TRUE
PWR2_Status=TRUE
GrpEnableStatus=TRUE
AxisHomed=TRUE
IN0
IN1
IN2
IN3
IN4
SMC_MoveLinearRelative2D1
SMC_MoveLinearRelative2D
EN
ENO
AxesGroup
Rel2D1_Exe=TRUE
Rel2D1_Pos
Rel2D1_Vel=20000
Rel2D1_ACC=50000
Rel2D1_DEC=50000
Rel2D1_Jerk=200000
Rel2D1_Coord=ACS
Rel2D1_Buffer=Buffered
Rel2D1_TransParam
AxesGroup
Execute
Position
Velocity
Acceleration
Deceleration
Jerk
CoordSystem
BufferMode
TransitionParameter
Done
Busy
Active
CommandAborted
Error
ErrorID
Rel2D1_Done=TRUE
Rel2D1_Busy=FALSE
Rel2D1_Active=FALSE
Rel2D1_ComdAbort=FALSE
Rel2D1_Err=FALSE
Rel2D1_ErrID=MC_ALM_NO_ERR


0013
SMC_MoveLinearRelative2D2
SMC_MoveLinearRelative2D
EN
ENO
AxesGroup
Rel2D1_Active=TRUE
Rel2D2_Pos
AxesGroup
Execute
Position
Rel2D2_Jerk=5000
Jerk
CoordSystem
Done
Busy
Active
Rel2D2_Done=FALSE
Rel2D2_Busy=TRUE
Rel2D2_Active=FALSE


0017
SMC_MoveCircularRelative2D1
SMC_MoveCircularRelative2D
EN
ENO
AxesGroup
Rel2D1_Active=TRUE
Circ1_Mode=CENTER
Circ1_Aux
Circ1_End
Circ1_Path=CLOCKWISE
Circ1_Vel=20000
AxesGroup
Execute
CircMode
AuxPoint
EndPoint
PathChoice
Velocity
Acceleration
Deceleration
Jerk
CoordSystem
BufferMode
TransitionMode
Circ1_Trans=TMNone
Done
Busy
Active
CommandAborted
Error
ErrorID
Circ1_Done=FALSE
Circ1_Busy=TRUE
Circ1_Active=FALSE
Circ1_Comdabort=FALSE
Circ1_Err=FALSE
Circ1_ErrID=MC_ALM_NO_ERR



ST

AxesGroup	AXES_GROUP_REF	-	
AxisI	AXIS_REF	-	
AxisX	AXIS_REF	-	X
AxisY	AXIS_REF	-	Y
StartProg	BOOL	FALSE	TRUE
Power1_Status	BOOL	FALSE	X X TRUE
Power2_Status	BOOL	FALSE	Y Y TRUE
Power0_Status	BOOL	FALSE	TRUE

GrpEnableStatus	BOOL	TRUE	TRUE
GrpEnable1_Exe	BOOL	FALSE	
AxisHomed	BOOL	FALSE	TRUE
InitFlag	BOOL	FALSE	TRUE
Rel2D1_Exe	BOOL	FALSE	1
Rel2D1_Active	BOOL	FALSE	1 Active TRUE

1 ID AxisI AxisX AxisY ID

```
IF Switch1 THEN
   AxisI.AxisID:=0;
   AxisX.AxisID:=1;
   AxisY.AxisID:=2;
   AxesGroup.AxisInterp_ID:=AxisI.AxisID;
   AxesGroup.AxisX_ID:=AxisX.AxisID;
   AxesGroup.AxisY_ID:=AxisY.AxisID;
END_IF
```

2

```
IF FALSE = InitFlag THEN
   (*                    1                                    *)
   Rel2D1_Pos[0]:=300;
   Rel2D1_Pos[1]:=400;
   Rel2D1_Vel:=100;
   Rel2D1_ACC:=200;
   Rel2D1_DEC:=200;
   Rel2D1_Jerk:=5000;
   Rel2D1_Buffer:=Buffered;
   (*                    2                                    *)
   Rel2D2_Pos[0]:= 0;
   Rel2D2_Pos[1]:=400;
   Rel2D2_Vel:=100;
   Rel2D2_ACC:=200;
   Rel2D2_DEC:=200;
   Rel2D2_Jerk:=5000;
   Rel2D2_Buffer:=Buffered;
   (*             3                                    *)
```

```
  Circ1_Mode:= BORDER;
  Circ1_Aux [0] :=400;
  Circ1_Aux [1] :=400;
  Circ1_End [0] :=800;
  Circ1_End [1] :=0;
  Circ1_Path:= CLOCKWISE;
  Circ1_Vel:=100;
  Circ1_ACC:=200;
  Circ1_DEC:=200;
  Circ1_Jerk:=5000;
  Circ1_Buffer:=Buffered;
  (*                4                              *)
  Abs2D1_Pos[0]:= 1100;
  Abs2D1_Pos[1]:=400;
  Abs2D1_Vel:=100;
  Abs2D1_ACC:=200;
  Abs2D1_DEC:=200;
  Abs2D1_Jerk:=5000;
  AbsD1_Buffer:=Buffered;
  (*          5                              *)
  Circ1_Mode:= CENTER;
  Circ1_Aux [0] :=400;
  Circ1_Aux [1] := 0;
  Circ1_End [0] :=400;
  Circ1_End [1] :=400;
  Circ1_Path:= COUNTERCLOCKWISE;
  Circ1_Vel:=100;
  Circ1_ACC:=200;
  Circ1_DEC:=200;
  Circ1_Jerk:=5000;
  Circ1_Buffer:=Buffered;
  InitFlag:=TRUE;
END_IF
```

3 SMC_AxisInit 0 Virtual

AxisX/AxisY 50 EtherCAT_Position SMC_AxisInit

4 MC_Power

```
IF StartPro THEN
   MC_Power0(                                (*          *)
   Enable := PWR0_Enable,
   Axis := AxisI,
   Status=> PWR0_Status,
   Busy=> PWR0_Busy,
   Error=> PWR0_Err,
   ErrorID=> PWR0_ErrID);
   MC_Power1(                                (*      AxisX      *)
   Enable := PWR1_Enable,
   Axis := AxisX,
   Status=> PWR1_Status,
   Busy=> PWR1_Busy,
   Error=> PWR1_Err,
   ErrorID=> PWR1_ErrID);
   MC_Power2(                                (*      AxisY      *)
   Enable := PWR2_Enable,
   Axis := AxisY,
   Status=> PWR2_Status,
   Busy=> PWR2_Busy,
   Error=> PWR2_Err,
   ErrorID=> PWR2_ErrID);
END_IF
```

5

```
MC_GroupEnable1(
Execute := GrpEnable_Exe,
AxesGroup := AxesGroup,
Done=> GrpEnable_Done,
Busy=> GrpEnable_Busy,
Error=> GrpEnable_Err,
ErrorID=> GrpEnable_ErrorID);
GrpEnableStatus:= AxesGroup. AxesGroupDisabled;
```

6 MC_Home

```
IF  PWR1_Status THEN
   MC_Home1(
   Execute := Home1_Exe,
   HomeMode := Home1_Mode,
   Velocity := Home1_Vel,
   ApproachVelocity := Home1_AppVel,
   Acceleration := Home1_Acc,
   Deceleration := Home1_Dec,
   Jerk := Home1_Jerk,
   Position := Home1_Pos,
   Axis := AxisX,
   Done=> Home1_Done,
   Busy=> Home1_Busy,
   Active=> Home1_Active,
   CommandAborted=> Home1_Comd,
   Error=> Home1_Err,
   ErrorID=> Home1_ErrID);
   MC_Home2(
   Execute := Home2_Exe,
   HomeMode := Home2_Mode,
   Velocity := Home2_Vel,
   ApproachVelocity := Home2_AppVel,
   Acceleration := Home2_Acc,
   Deceleration := Home2_Dec,
   Jerk := Home2_Jerk,
   Position := Home2_Pos,
   Axis := AxisY,
   Done=> Home2_Done,
   Busy=> Home2_Busy,
   Active=> Home2_Active,
   CommandAborted=> Home2_Comd,
   Error=> Home2_Err,
   ErrorID=> Home2_ErrID);
END_IF
IF  Home1_Done  AND Home2_Done THEN
   AxisHomed:=TRUE;
END_IF
```

7		1	1	Rel2D1_Active	TRUE
	2-5				

IF PWR0_Status

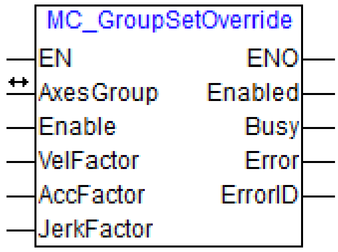
```
CommandAborted=> Rel2D2_ComdAbort,
Error=> Rel2D2_Err,
ErrorID=> Rel2D2_ErrID);
SMC_MoveCircularRelative2D1(                (*              3*)
Execute := Rel2D1_Active,
CircMode := Circ1_Mode ,
AuxPoint := Circ1_Aux,
EndPoint := Circ1_End,
PathChoice:= Circ1_Path,
Velocity := Circ1_Vel,
Acceleration := Circ1_ACC,
Deceleration := Circ1_DEC,
Jerk := Circ1_Jerk,
CoordSystem := Circ1_Coord,
BufferMode := Circ1_Buffer,
TransitionMode := Circ1_ Trans,
TransitionParameter := Circ1_Param,
AxesGroup := AxesGroup,
Done=> Circ1_Done,
Busy=> Circ1_Busy,
Active=> Circ1_Active,
CommandAborted=> Circ1_Abort,
Error=> Circ1_Err,
ErrorID=> Circ1_ErrID);
SMC_MoveLinearAbsolute2D1(           (*              4*)
Execute := Rel2D1_Active,
Position := Abs2D1_Pos,
Velocity := Abs2D1_Vel,
Acceleration := Abs2D1_ACC,
Deceleration := Abs2D1_DEC,
Jerk := Abs2D1_Jerk,
CoordSystem := Abs2D1_Coord,
BufferMode := Abs2D1_Buffer,
TransitionMode := Abs2D1_Trans,
TransitionParameter := Abs2D1_Param,
AxesGroup := AxesGroup,
Done=> Abs2D1_Done,
```

```
    Busy=> Abs2D1_Busy,
    Active=> Abs2D1_Active,
    CommandAborted=> Abs2D1_ComdAbort,
    Error=> Abs2D1_Err,
    ErrorID=> Abs2D1_ErrID);
    SMC_MoveCircularRelative2D2(                (*              5*)
    Execute := Rel2D1_Active,
    CircMode := Circ2_Mode ,
    AuxPoint := Circ2_Aux,
    EndPoint := Circ2_End,
    PathChoice:= Circ2_Path,
    Velocity := Circ2_Vel,
    Acceleration := Circ2_ACC,
    Deceleration := Circ2_DEC,
    Jerk := Circ2_Jerk,
    CoordSystem := Circ2_Coord,
    BufferMode := Circ2_Buffer,
    TransitionMode := Circ2_ Trans,
    TransitionParameter := Circ2_Param,
    AxesGroup := AxesGroup,
    Done=> Circ2_Done,
    Busy=> Circ2_Busy,
    Active=> Circ2_Active,
    CommandAborted=> Circ2_Abort,
    Error=> Circ2_Err,
    ErrorID=> Circ2_ErrID);
END_IF
```

7.6.4

BufferMode (   )   0   Aborting

7.7 MC_GroupSetOverride



7.7.1

IN_OUT	AxesGroup			-	-	AXES_GROUP_REF
INPUT	Enable			-	TRUE/FALSE	BOOL
	VelFactor	1%		100	0 ~ 500	LREAL
	AccFactor	1%		100	0 ~ 500	LREAL
	JerkFactor	1%		100	0 ~ 500	LREAL
OUTPUT	Enabled			-	TRUE/FALSE	BOOL
	Busy			FALSE	TRUE/FALSE	BOOL
	Error			FALSE	TRUE/FALSE	BOOL
	ErrorID			0	-	SMC_ERROR

7.7.2

1

SMC_MoveLinearAbsolute2D

SMC_MoveCircularRelative2D

SMC_MoveLinearRelative2D

2 [%]

×

= ×

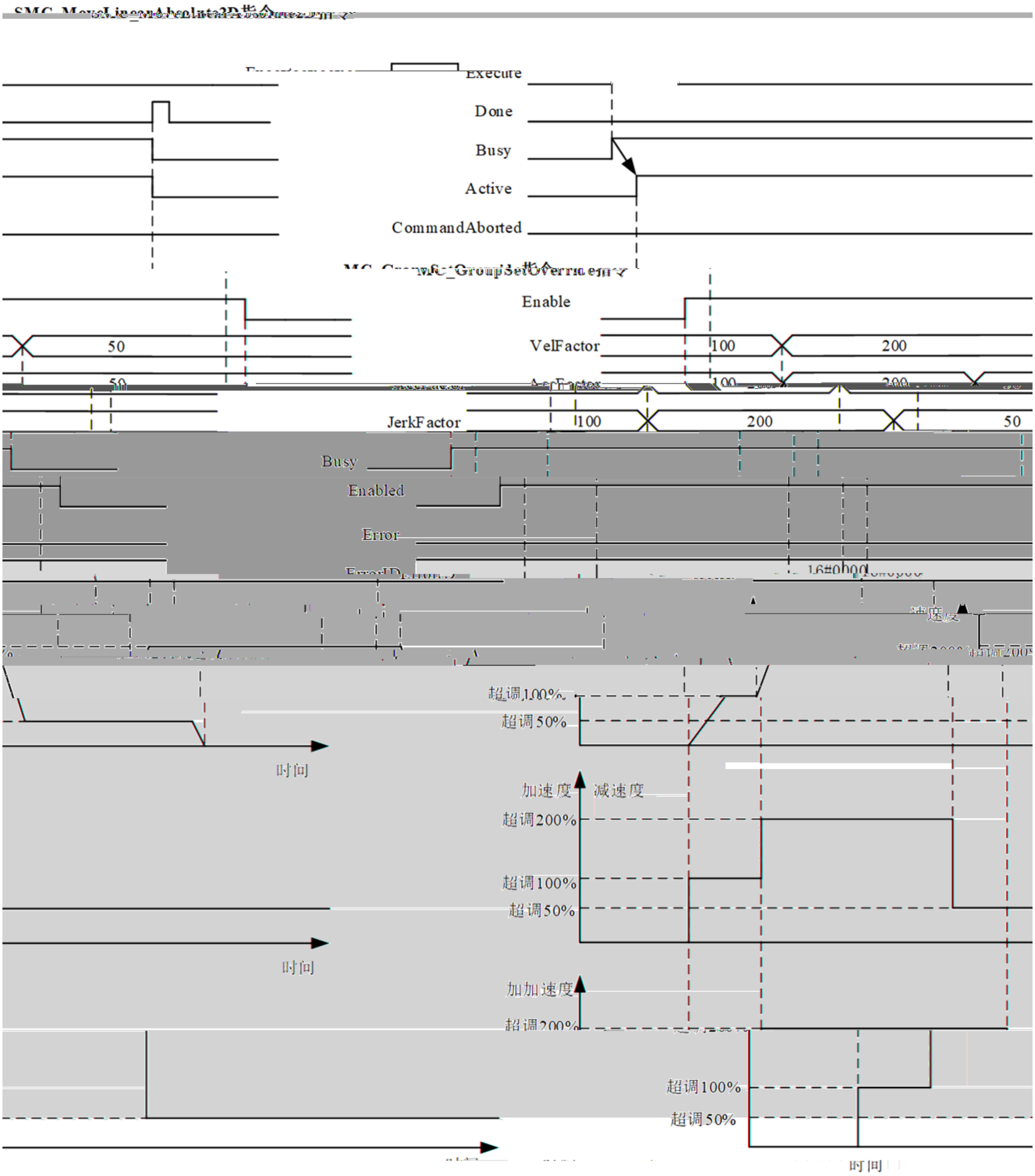
= ×

3

4 Enable( ) TRUE Enable( ) FALSE

1 SMC_ MoveLinearAbsolute2D (2)

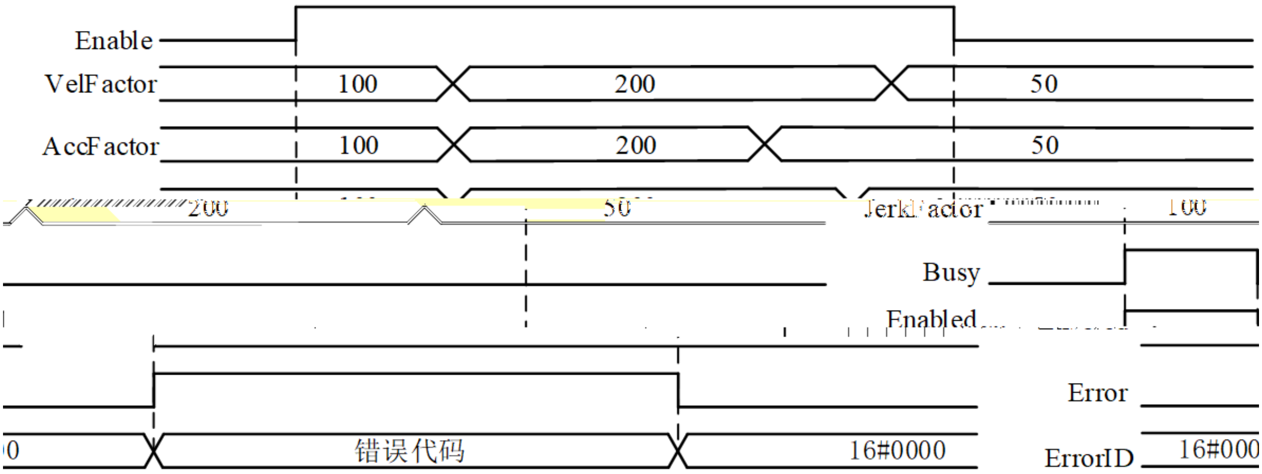
Enable( ) TRUE Busy( ) Enabled( ) TRUE Enable( ) FALSE Busy( ) Enabled( ) FALSE SMC_MoveLinearAbsolute2D (2)



2

Error( ) TRUE Enabled( ) Busy( )

FALSE Error( ) FALSE Enabled( ) Busy( ) TRUE



3

Enable Enable( ) TRUE

MC_GroupSetOverride MC_GroupSetOverride

7.7.3

MC_GroupSetOverride

AxisGrp	AXES_GROUP_REF		
StartOverride	BOOL	FALSE	TRUE
BusyOverride	BOOL	FALSE	TRUE
EnabledOverride	BOOL	FALSE	TRUE
ErrorOverride	BOOL	FALSE	TRUE

VelFactor AccFactor JerkFactor 300

SMC_MoveLinearAbsolute2D MC_GroupSetOverride Enable( )

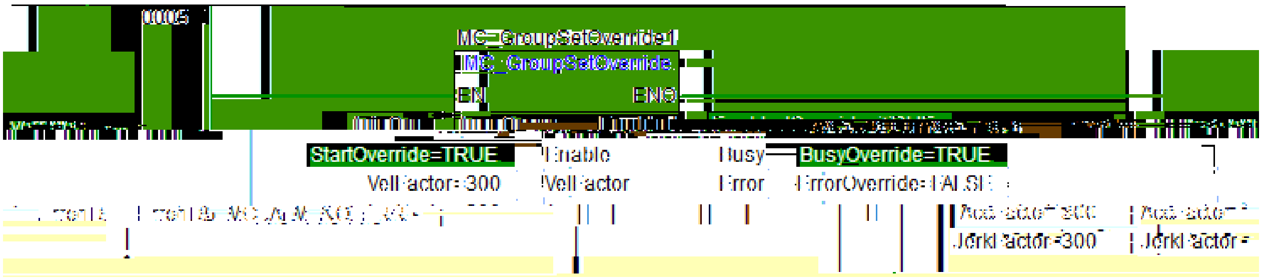
TRUE

1 X Y ID

SMC_MoveLinearAbsolute2D

2 SMC_MoveLinearAbsolute2D

2 MC_GroupSetOverride Enable



ST

1 X Y ID

SMC_MoveLinearAbsolute2D

2 SMC_MoveLinearAbsolute2D

2 MC_GroupSetOverride

```
MC_GroupSetOverride1(
Enable := StartOverride,
VelFactor := 300,
AccFactor := 300,
JerkFactor := 300,
Axis := Axis0,
Enabled=>EnabledOverride,
Busy=>BusyOverride,
Error=>ErrorOverride,
ErrorID=>ErrorID);
```

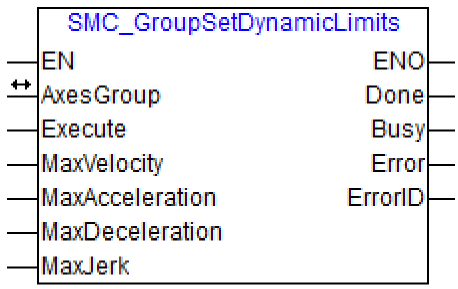
7.7.4

Enabled TRUE Enabled TRUE

7.7.5

SMC_MoveLinearAbsolute2D

7.8 SMC_GroupSetDynamicLimits



7.8.1

IN_OUT	AxesGroup			-	-	AXIS_REF
INPUT	Execute			-	TRUE/FALSE	BOOL
	MaxVelocity			1E100	/0	LREAL
	MaxAcceleration			1E100		LREAL
	MaxDeceleration			1E100		LREAL
	MaxJerk			1E100		LREAL
OUTPUT	Done			FALSE	TRUE/FALSE	BOOL
	Busy			FALSE	TRUE/FALSE	BOOL
	Error			FALSE	TRUE/FALSE	BOOL
	ErrorID			0	-	SMC_ERROR

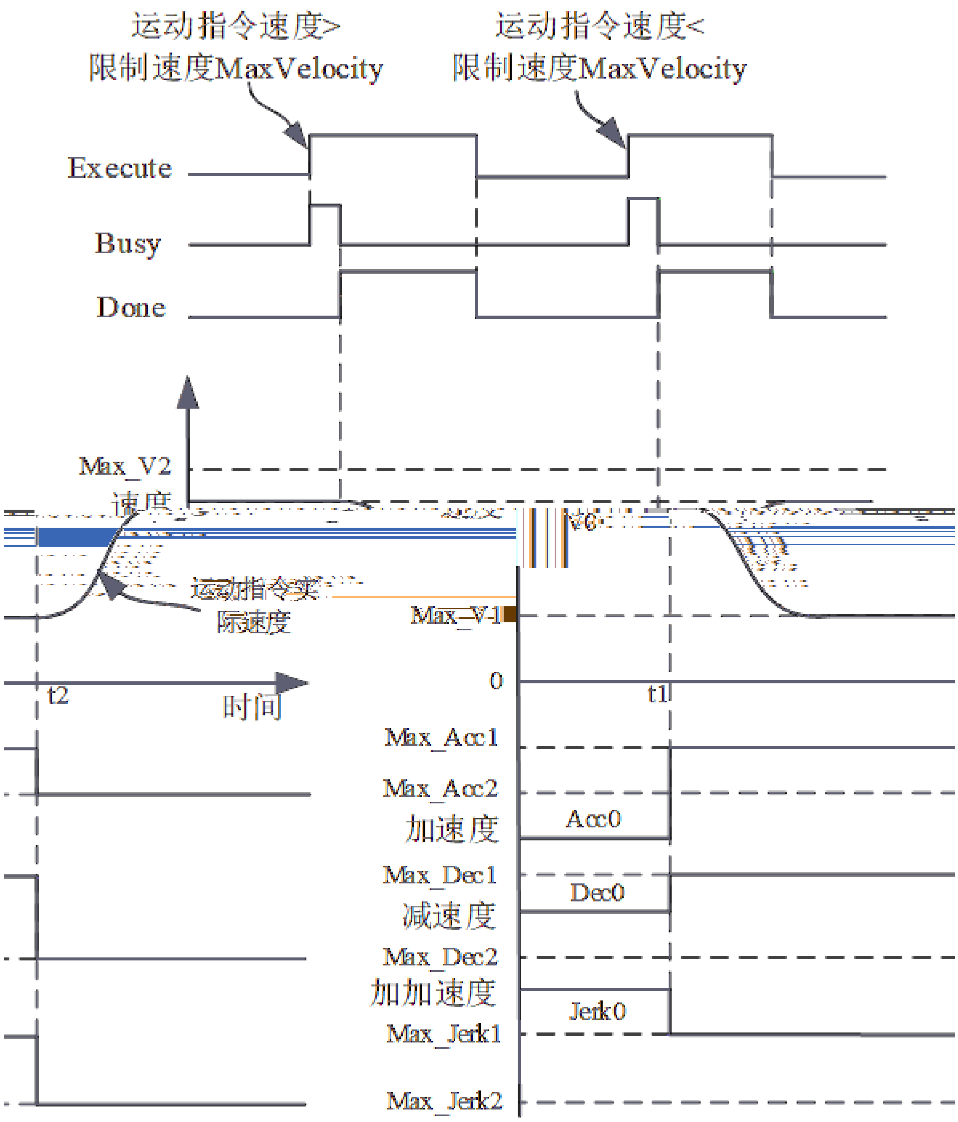
7.8.2

1

Execute AxesGroup

ID MaxVelocity MaxAcceleration MaxDeceleration MaxJerkC.9ﾞì Rmi

RRcOkC.9



V0 Acc0 Dec0

Jerk0

t1 MaxVelocity MaxVelocity= Max_V1 MaxVelocity

V0 MaxVelocity

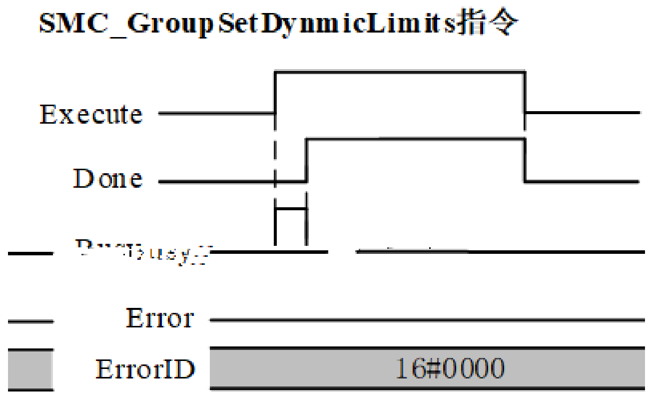
Max_Acc1 Max_Dec1 Max_Jerk1

t2 MaxVelocity MaxVelocity= Max_V2 MaxVelocity

V0

Max_Acc2 Max_Dec2 Max_Jerk2

1



2

Error=True ErrorID

Execute

• Execute SMC_SetGroupDynamicLimits

7.8.3

SMC_GroupSetDynamicLimits

1 SMC_MoveLinearRelative2D 20000

20000 500000

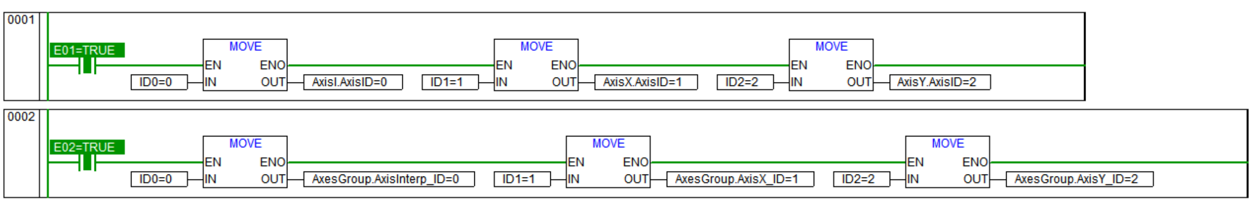
2 SMC_GroupSetDynamicLimits MaxVelocity 10000

MaxAcceleration MaxDeceleration 20000 MaxJerk 50000

AxesGroup	AXES_GROUP_REF	-	
AxisI	AXIS_REF	-	
AxisX	AXIS_REF	-	X
AxisY	AXIS_REF	-	Y
StartProg	BOOL	FALSE	TRUE
Power1_Status	BOOL	FALSE	X X TRUE
Power2_Status	BOOL	FALSE	Y Y TRUE
Power0_Status	BOOL	FALSE	TRUE
GrpEnableStatus	BOOL	TRUE	TRUE
GrpEnable1_Exe	BOOL	FALSE	
AxisHomed	BOOL	FALSE	TRUE
Rel2D1_Exe	BOOL	FALSE	
Rel2D1_Vel	BOOL	FALSE	
Dyn_Exe	BOOL	FALSE	

1 ID

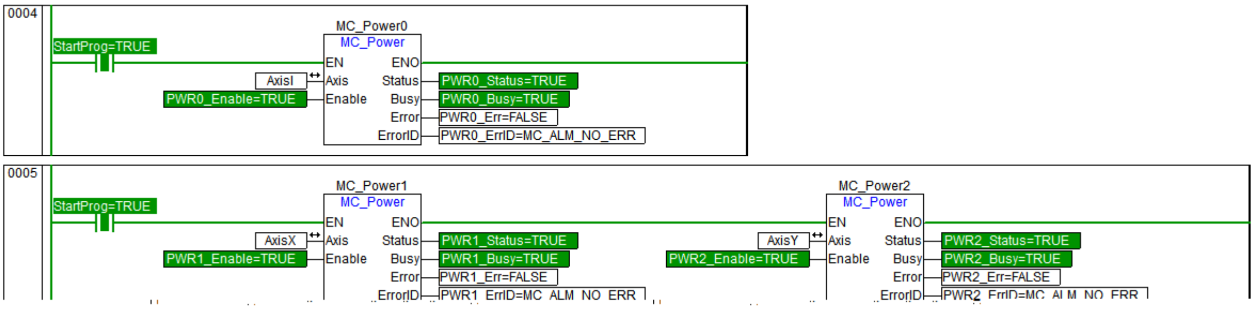
ID ID 0001 0002



2 SMC_AxisInit 0

3 StartProg =TRUE MC_Power

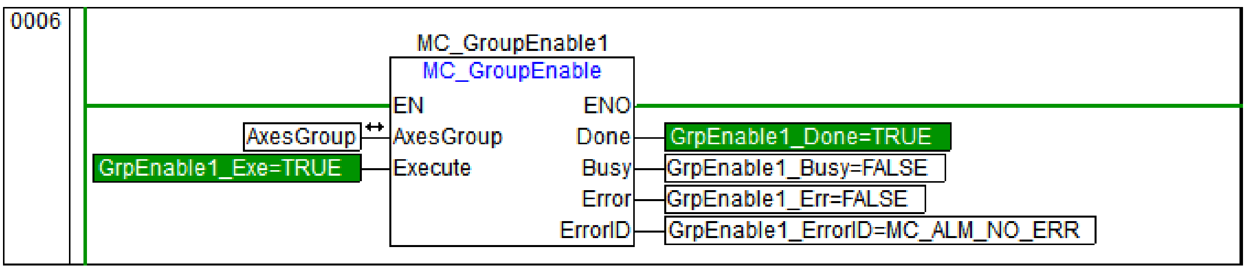
0004 ~ 0005

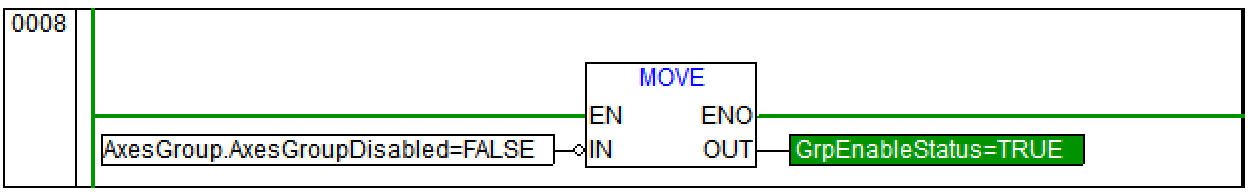


4 MC_GroupEnable AxesGroupDisabled FALSE

0006 AxesGroupDisabled

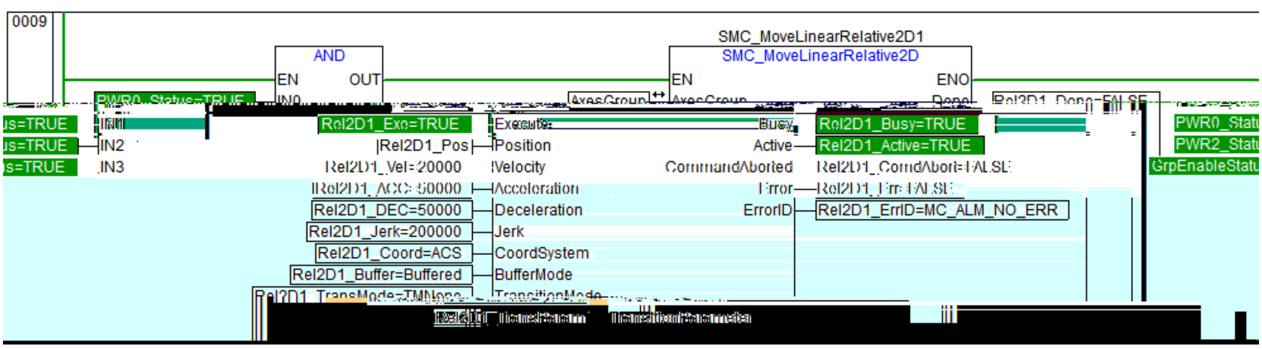
GrpEnableStatus



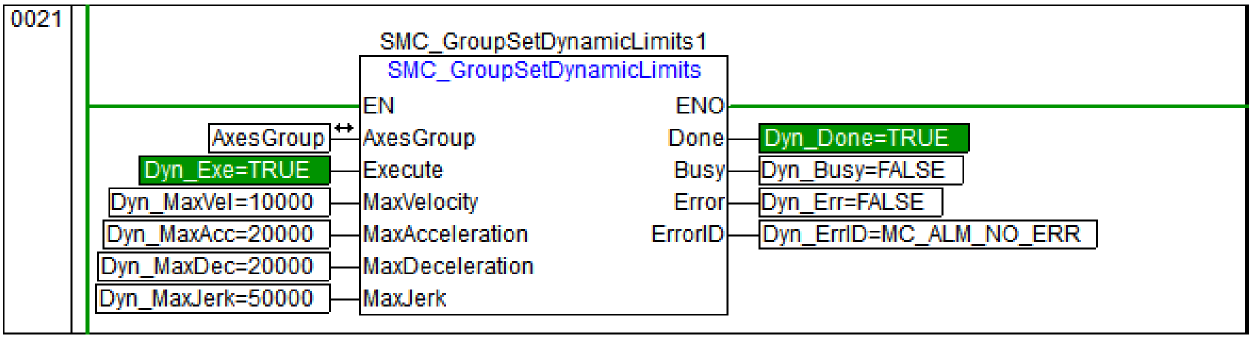


5 0009

SMC_MoveLinearRelative2D



6 0021 SMC_GroupSetDynamicLimits


0021
SMC_GroupSetDynamicLimits1
SMC_GroupSetDynamicLimits
EN
ENO
AxesGroup
AxesGroup
Done
Dyn_Done=TRUE
Dyn_Exe=TRUE
Execute
Busy
Dyn_Busy=FALSE
Dyn_MaxVel=10000
MaxVelocity
Error
Dyn_Err=FALSE
Dyn_MaxAcc=20000
MaxAcceleration
ErrorID
Dyn_ErrID=MC_ALM_NO_ERR
Dyn_MaxDec=20000
MaxDeceleration
Dyn_MaxJerk=50000
MaxJerk

ST

```
   Dyn_MaxVel:=10000;          (*            *)
   Dyn_MaxAcc:=20000;          (*              *)
   Dyn_MaxDec:=20000;          (*              *)
   Dyn_MaxJerk:=50000;         (*                *)
   InitFlag:=TRUE;
END_IF
```

2 SMC_AxisInit 0 Virtual

AxisX/AxisY 50 EtherCAT_Position SMC_AxisInit

3 MC_Power

```
IF StartPro THEN
   MC_Power0(                        (*            *)
   Enable := PWR0_Enable,
   Axis := AxisI,
   Status=> PWR0_Status,
   Busy=> PWR0_Busy,
   Error=> PWR0_Err,
   ErrorID=> PWR0_ErrID);
   MC_Power1(                        (*     AxisX      *)
   Enable := PWR1_Enable,
   Axis := AxisX,
   Status=> PWR1_Status,
   Busy=> PWR1_Busy,
   Error=> PWR1_Err,
   ErrorID=> PWR1_ErrID);
   MC_Power2(                        (*     AxisY      *)
   Enable := PWR2_Enable,
   Axis := AxisY,
   Status=> PWR2_Status,
   Busy=> PWR2_Busy,
   Error=> PWR2_Err,
   ErrorID=> PWR2_ErrID);
```

```
END_IF
```

4

```
MC_GroupEnable1(
Execute := GrpEnable_Exe,
AxesGroup := AxesGroup,
Done=> GrpEnable_Done,
Busy=> GrpEnable_Busy,
Error=> GrpEnable_Err,
ErrorID=> GrpEnable_ErrorID);
GrpEnableStatus:= AxesGroup. AxesGroupDisabled;
```

5

```
IF PWR0_Status  AND PWR1_Status  AND PWR2_Status  AND GrpEnableStatus THEN
   SMC_MoveLinearRelative2D1(
   Execute := Rel2D1_Exe,
   Position := Rel2D1_Pos,
   Velocity := Rel2D1_Vel,
   Acceleration := Rel2D1_ACC,
   Deceleration := Rel2D1_DEC,
   Jerk := Rel2D1_Jerk,
   CoordSystem := Rel2D1_Coord,
   BufferMode := Rel2D1_Buffer,
   TransitionMode := Rel2D1_Trans,
   TransitionParameter := Rel2D1_Param,
   AxesGroup := AxesGroup,
   Done=> Rel2D1_Done,
   Busy=> Rel2D1_Busy,
   Active=> Rel2D1_Active,
   CommandAborted=> Rel2D1_ComdAbort,
   Error=> Rel2D1_Err,
   ErrorID=> Rel2D1_ErrID);
END_IF
```

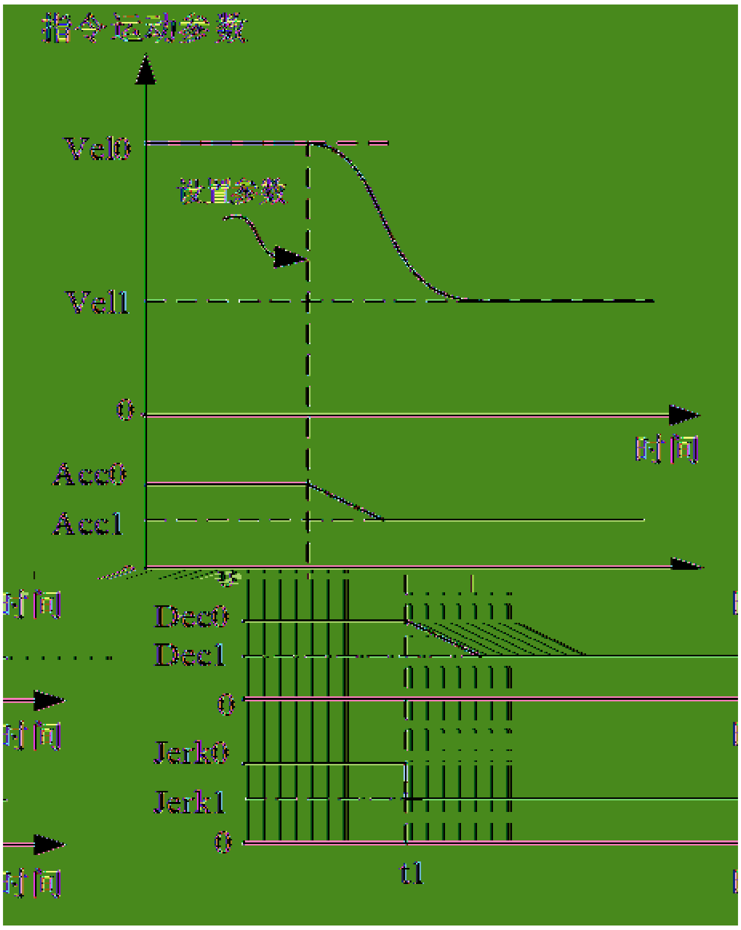
6 SMC_GroupSetDynamicLimits

```
SMC_SetGroupDynamicLimits1(
Execute := Dyn_Exe,
MaxVelocity := Dyn_MaxVel,
MaxAcceleration := Dyn_MaxAcc,
```

```
MaxDeceleration := Dyn_MaxDec,
MaxJerk := Dyn_MaxJerk,
AxesGroup := AxesGroup,
Done=> Dyn_Done,
Busy=> Dyn_Busy,
Error=> Dyn_Err,
ErrorID=> Dyn_ErrID);
```

0~t1

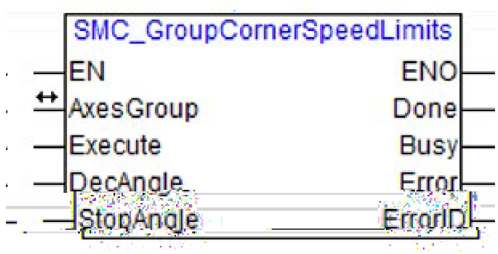
t1 SMC_GroupSetDynamicLimits



7.8.4

MaxVelocity 0

7.9 SMC_GroupCornerSpeedLimits



7.9.1

IN_OUT	AxesGroup			-	-	AXES_GROUP_REF
INPUT	Execute			-	TRUE/FALSE	BOOL
	DecAngle			30	0 ~ 30 ≤StopAngle	LREAL
	StopAngle			90	0 ~ 90 0	LREAL
OUTPUT	Done			FALSE	TRUE/FALSE	BOOL
	Busy			FALSE	TRUE/FALSE	BOOL
	Error			FALSE	TRUE/FALSE	BOOL
	ErrorID			0	-	SMC_ERROR

7.9.2

1

Execute AxesGroup

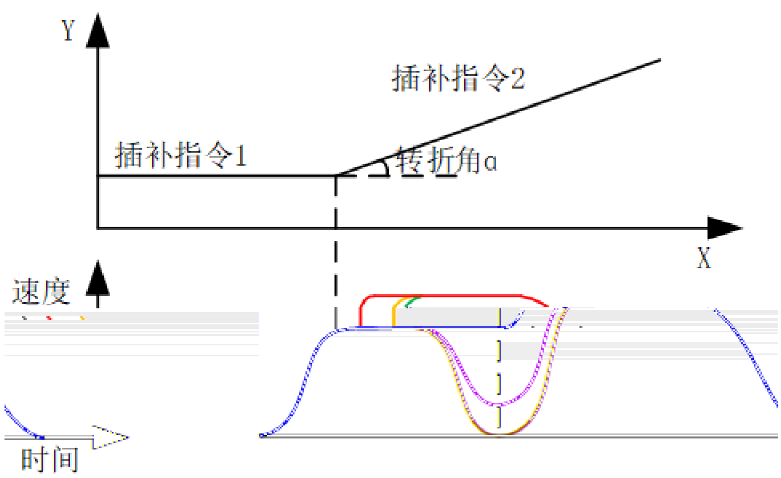
ID DecAngle StopAngle

2

DecAngle DecAngle

0 DecAngle DecAngle

3



2 1

1

α

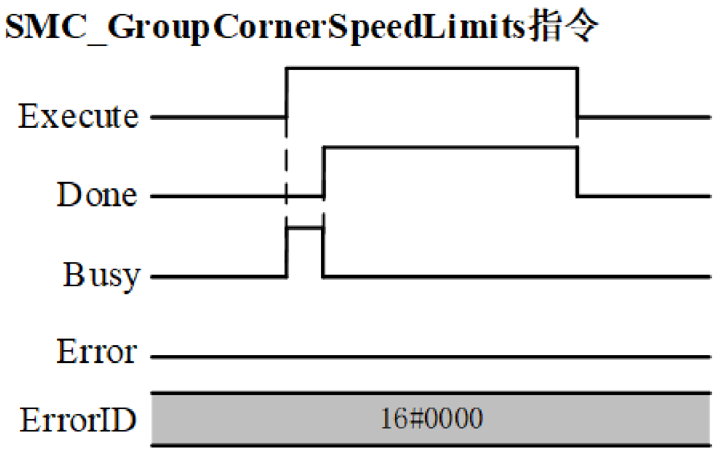
α

ɑ

ɑ

0

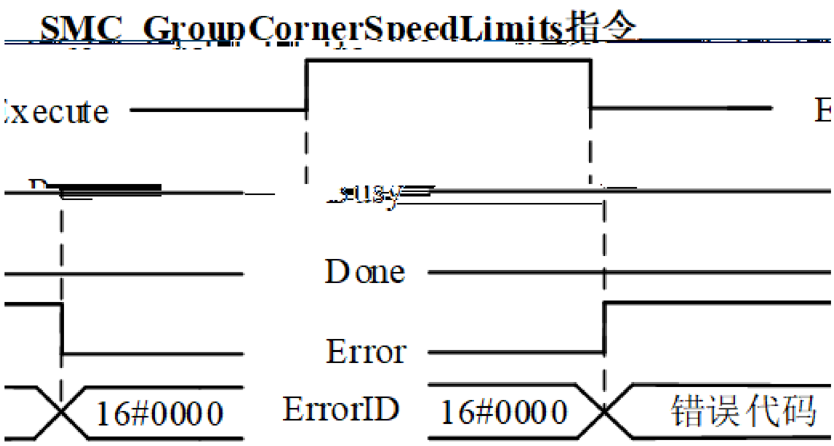
1



2

Error=True ErrorID

Execute



• Execute

SMC_SetGroupCornerSpeedLimits

7.9.3

1

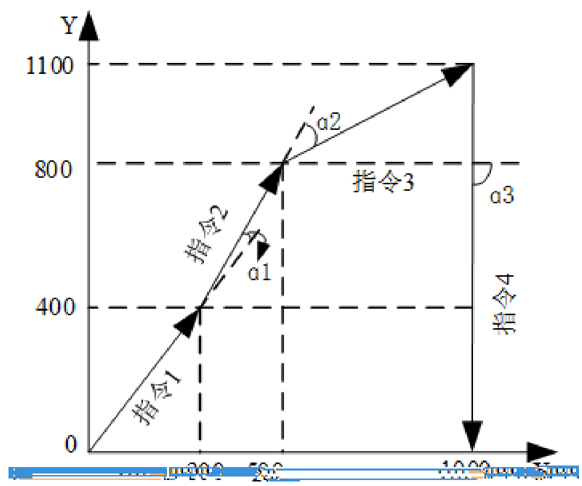
SMC_GroupCornerSpeedLimits

4

1 BufferMode Buffered 3

BufferMode 3 4 5

1 Active TRUE 2-4



1 300,400 2 500,800

3 1000,1100 4 1000, 0

1 2 α1 10° 2 3 32° 3 4

121°

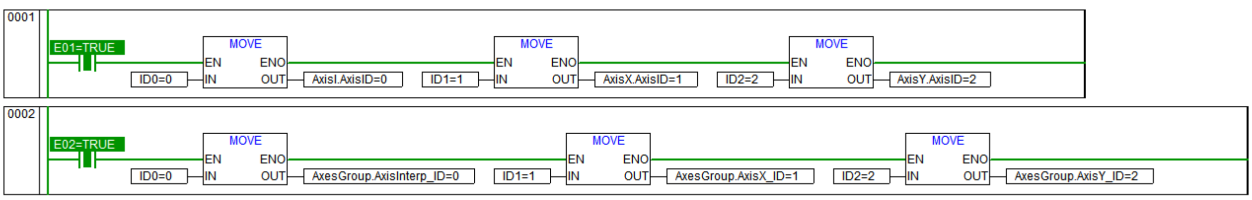
SMC_GroupCornerSpeedLimits DecAngle 20°

StopAngle 60°

AxesGroup	AXES_GROUP_REF	-	
AxisI	AXIS_REF	-	
AxisX	AXIS_REF	-	X
AxisY	AXIS_REF	-	Y
StartProg	BOOL	FALSE	TRUE
Power1_Status	BOOL	FALSE	X X TRUE
Power2_Status	BOOL	FALSE	Y Y TRUE
Power0_Status	BOOL	FALSE	TRUE
GrpEnableStatus	BOOL	TRUE	TRUE
AxisHomed	BOOL	FALSE	TRUE
Rel2D1_Active	BOOL	FALSE	1 Active TRUE

1 ID

ID ID 0001 0002

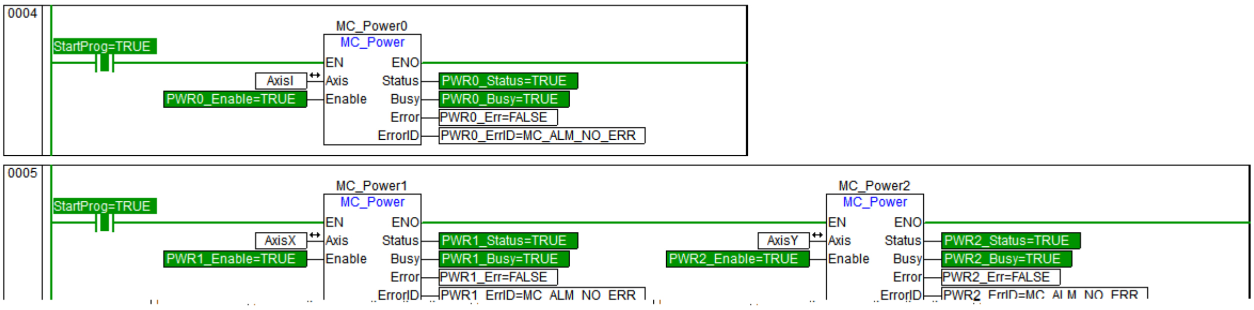


2 SMC_AxisInit 0

Virtual 50: EtherCAT_Position SMC_AxisInit

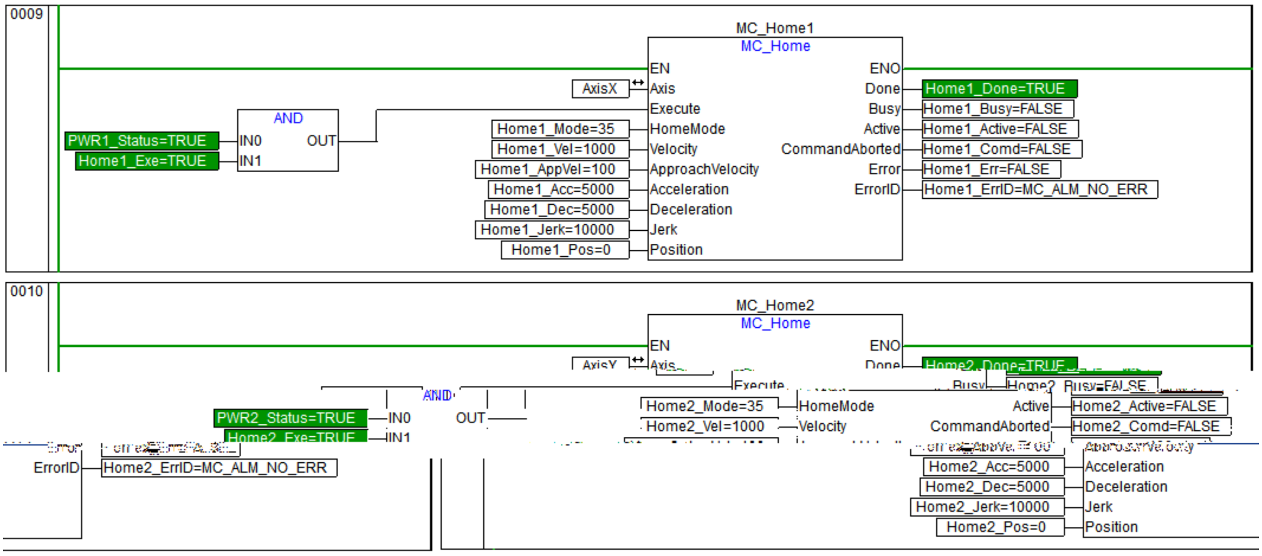
3 StartProg =TRUE MC_Power

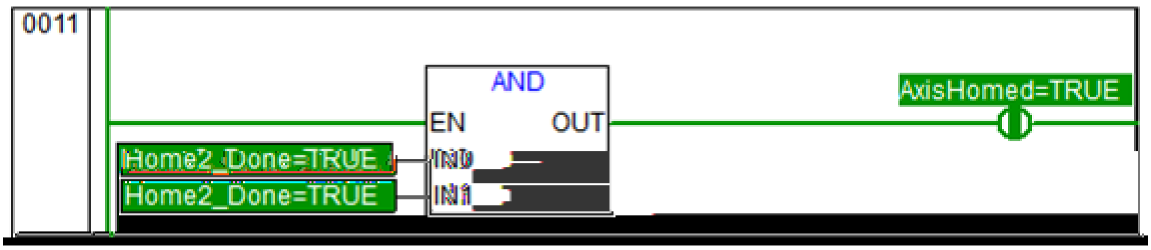
0004 ~ 0005



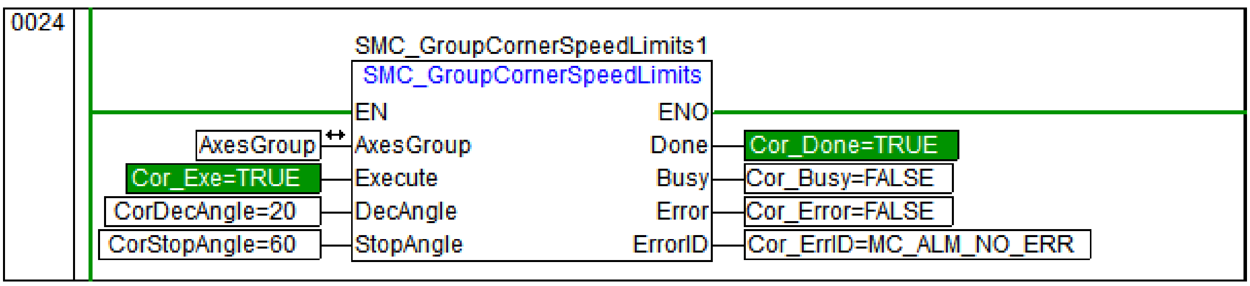
4 MC_GroupEnable AxesGroupDisabled FALSE

0006 &



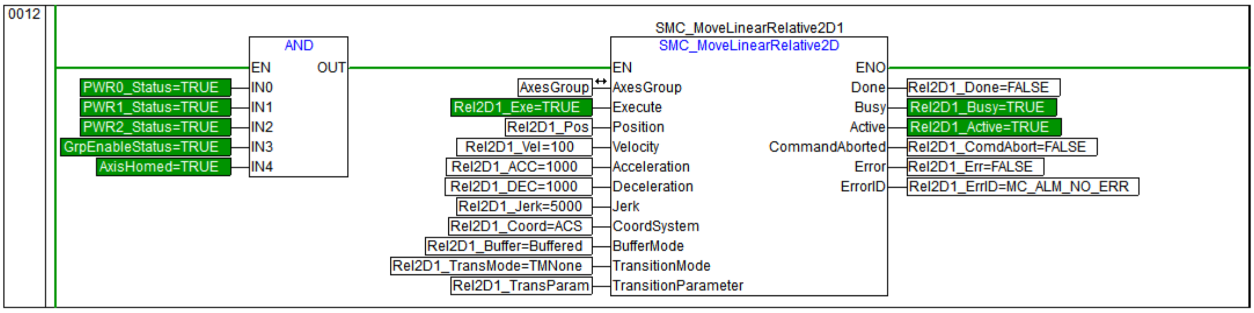


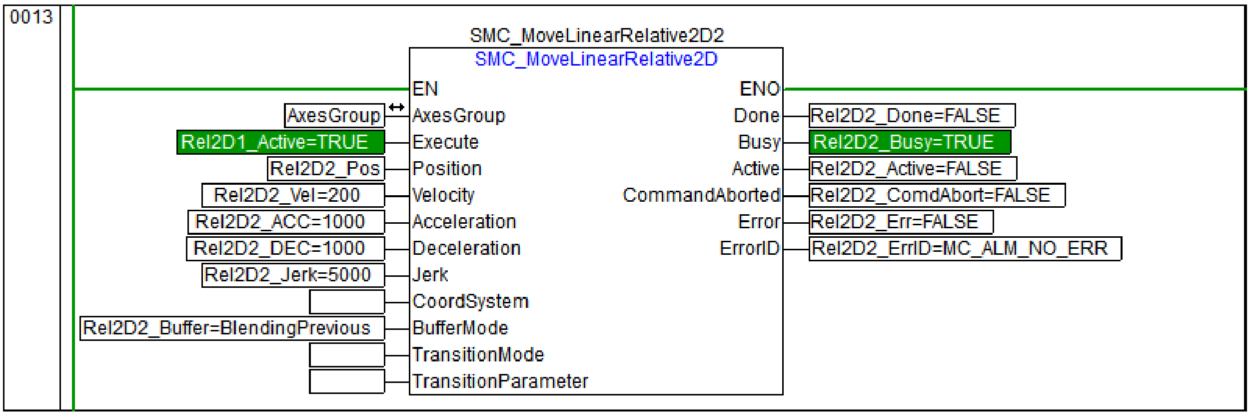
6 0024 SMC_GroupCornerSpeedLimits

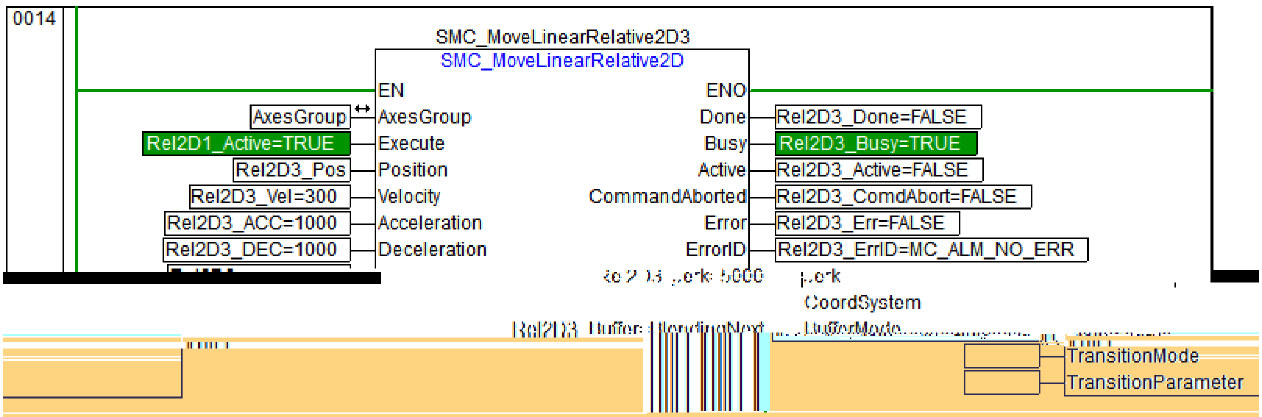


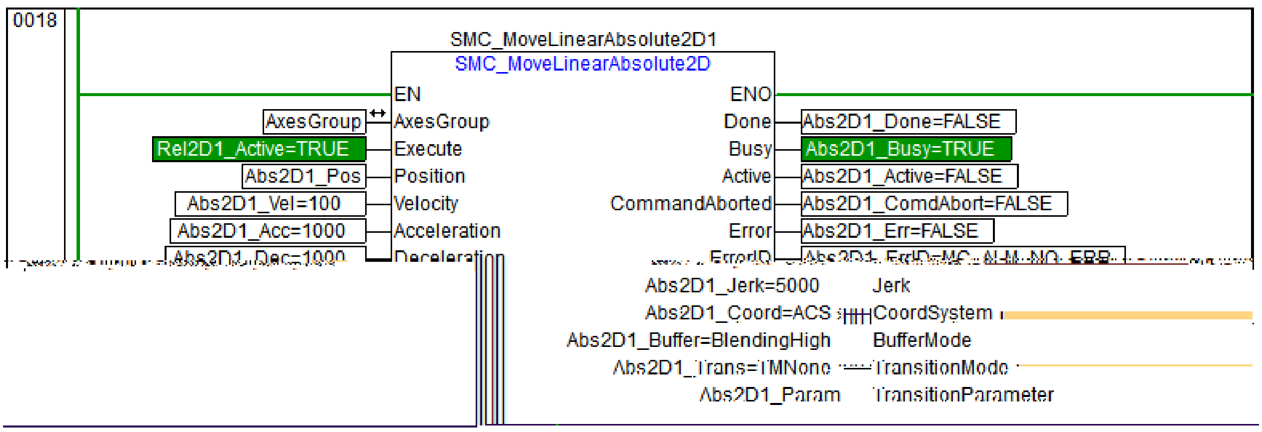
7 0011 1 1 Rel2D1_Active

TRUE 2-4 0012-0014 0018


0012
AND
EN
OUT
PWR0_Status=TRUE
PWR1_Status=TRUE
PWR2_Status=TRUE
GrpEnableStatus=TRUE
AxisHomed=TRUE
IN0
IN1
IN2
IN3
IN4
SMC_MoveLinearRelative2D1
SMC_MoveLinearRelative2D
EN
ENO
AxesGroup
Rel2D1_Exe=TRUE
Rel2D1_Pos
Rel2D1_Vel=100
Rel2D1_ACC=1000
Rel2D1_DEC=1000
Rel2D1_Jerk=5000
Rel2D1_Coord=ACS
Rel2D1_Buffer=Buffered
Rel2D1_TransMode=TMNone
Rel2D1_TransParam
AxesGroup
Execute
Position
Velocity
Acceleration
Deceleration
Jerk
CoordSystem
BufferMode
TransitionMode
TransitionParameter
Done
Busy
Active
CommandAborted
Error
ErrorID
Rel2D1_Done=FALSE
Rel2D1_Busy=TRUE
Rel2D1_Active=TRUE
Rel2D1_ComdAbort=FALSE
Rel2D1_Err=FALSE
Rel2D1_ErrID=MC_ALM_NO_ERR


0013
SMC_MoveLinearRelative2D2
SMC_MoveLinearRelative2D
EN
ENO
AxesGroup
Rel2D1_Active=TRUE
Rel2D2_Pos
Rel2D2_Vel=200
Rel2D2_ACC=1000
Rel2D2_DEC=1000
Rel2D2_Jerk=5000
Rel2D2_Buffer=BlendingPrevious
AxesGroup
Execute
Position
Velocity
Acceleration
Deceleration
Jerk
CoordSystem
BufferMode
TransitionMode
TransitionParameter
Done
Busy
Active
CommandAborted
Error
ErrorID
Rel2D2_Done=FALSE
Rel2D2_Busy=TRUE
Rel2D2_Active=FALSE
Rel2D2_ComdAbort=FALSE
Rel2D2_Err=FALSE
Rel2D2_ErrID=MC_ALM_NO_ERR


0014
SMC_MoveLinearRelative2D3
SMC_MoveLinearRelative2D
EN
ENO
AxesGroup
Rel2D1_Active=TRUE
Rel2D3_Pos
Rel2D3_Vel=300
Rel2D3_ACC=1000
Rel2D3_DEC=1000
AxesGroup
Execute
Position
Velocity
Acceleration
Deceleration
Done
Busy
Active
CommandAborted
Error
ErrorID
Rel2D3_Done=FALSE
Rel2D3_Busy=TRUE
Rel2D3_Active=FALSE
Rel2D3_ComdAbort=FALSE
Rel2D3_Err=FALSE
Rel2D3_ErrID=MC_ALM_NO_ERR
CoordSystem
TransitionMode
TransitionParameter


0018
SMC_MoveLinearAbsolute2D1
SMC_MoveLinearAbsolute2D
EN
ENO
AxesGroup
Rel2D1_Active=TRUE
Abs2D1_Pos
Abs2D1_Vel=100
Abs2D1_Acc=1000
AxesGroup
Execute
Position
Velocity
Acceleration
Done
Busy
Active
CommandAborted
Error
Abs2D1_Done=FALSE
Abs2D1_Busy=TRUE
Abs2D1_Active=FALSE
Abs2D1_ComdAbort=FALSE
Abs2D1_Err=FALSE
Abs2D1_Jerk=5000
Jerk
Abs2D1_Coord=ACS
CoordSystem
Abs2D1_Buffer=BlendingHigh
BufferMode
TransitionMode
Abs2D1_Param
TransitionParameter

ST

```
  Rel2D2_Pos[1]:=400;
  Rel2D2_Vel:=200;
  Rel2D2_ACC:=1000;,
  Rel2D2_DEC:=1000;,
  Rel2D2_Jerk:=5000;,
  Rel2D2_Buffer:= BlendingPrevious;
  (*                    3                                    *)
  Rel2D2_Pos[0]:= 500;
  Rel2D2_Pos[1]:=300;
  Rel2D2_Vel:=300;
  Rel2D2_ACC:=1000;,
  Rel2D2_DEC:=1000;,
  Rel2D2_Jerk:=5000;,
  Rel2D2_Buffer:= BlendingNext;
  (*                    4                                    *)
  Abs2D1_Pos[0]:= 1000;
  Abs2D1_Pos[1]:= 0;
  Abs2D1_Vel:=100;
  Abs2D1_ACC:=1000;,
  Abs2D1_DEC:=1000;,
  Abs2D1_Jerk:=5000;,
  AbsD1_Buffer:= BlendingHigh;
  (*                *
  CorDecAngle:=20;
  CorStopAngle:=60;
  InitFlag:=TRUE;      (*              *)
END_IF
```

3 SMC_AxisInit 0 Virtual

AxisX/AxisY 50 EtherCAT_Position [SMC_AxisInit]()

4 MC_Power

```
IF StartPro THEN
  MC_Power0(                    (*        *)
```

```
   Enable := PWR0_Enable,
   Axis := AxisI,
   Status=> PWR0_Status,
   Busy=> PWR0_Busy,
   Error=> PWR0_Err,
   ErrorID=> PWR0_ErrID);
   MC_Power1(                              (*      AxisX      *)
   Enable := PWR1_Enable,
   Axis := AxisX,
   Status=> PWR1_Status,
   Busy=> PWR1_Busy,
   Error=> PWR1_Err,
   ErrorID=> PWR1_ErrID);
   MC_Power2(                              (*      AxisY      *)
   Enable := PWR2_Enable,
   Axis := AxisY,
   Status=> PWR2_Status,
   Busy=> PWR2_Busy,
   Error=> PWR2_Err,
   ErrorID=> PWR2_ErrID);
END_IF
```

5

```
MC_GroupEnable1(
Execute := GrpEnable_Exe,
AxesGroup := AxesGroup,
Done=> GrpEnable_Done,
Busy=> GrpEnable_Busy,
Error=> GrpEnable_Err,
ErrorID=> GrpEnable_ErrorID);
GrpEnableStatus:= AxesGroup. AxesGroupDisabled;
```

6 MC_Home

```
IF  PWR1_Status THEN
   MC_Home1(
   Execute := Home1_Exe,
   HomeMode := Home1_Mode,
```

```
StopAngle := Cor_StopAngle,
AxesGroup := AxesGroup,
Done=> Cor_Done,
Busy=> Cor_Busy,
Error=> Cor_Err,
ErrorID=> Cor_ErrID);
```

8 1 1 Rel2D1_Active TRUE

2-4

```
IF PWR0_Status  AND PWR1_Status  AND PWR2_Status  AND GrpEnableStatus AND AxisHomed THEN
   SMC_MoveLinearRelative2D1(            (*              1*)
   Execute := Rel2D1_Exe,
   Position := Rel2D1_Pos,
   Velocity := Rel2D1_Vel,
   Acceleration := Rel2D1_ACC,
   Deceleration := Rel2D1_DEC,
   Jerk := Rel2D1_Jerk,
   CoordSystem := Rel2D1_Coord,
   BufferMode := Rel2D1_Buffer,
   TransitionMode := Rel2D1_Trans,
   TransitionParameter := Rel2D1_Param,
   AxesGroup := AxesGroup,
   Done=> Rel2D1_Done,
   Busy=> Rel2D1_Busy,
   Active=> Rel2D1_Active,
   CommandAborted=> Rel2D1_ComdAbort,
   Error=> Rel2D1_Err,
   ErrorID=> Rel2D1_ErrID);
   SMC_MoveLinearRelative2D2(                (*              2*)
   Execute := Rel2D1_Active,
   Position := Rel2D2_Pos,
   Velocity := Rel2D2_Vel,
   Acceleration := Rel2D2_ACC,
   Deceleration := Rel2D2_DEC,
   Jerk := Rel2D2_Jerk,
   CoordSystem := Rel2D2_Coord,
   BufferMode := Rel2D2_Buffer,
```

```
TransitionMode := Rel2D2_Trans,
TransitionParameter := Rel2D2_Param,
AxesGroup := AxesGroup,
Done=> Rel2D2_Done,
Busy=> Rel2D2_Busy,
Active=> Rel2D2_Active,
CommandAborted=> Rel2D2_ComdAbort,
Error=> Rel2D2_Err,
ErrorID=> Rel2D2_ErrID);
SMC_MoveLinearRelative2D3(                    (*                3*)
Execute := Rel2D1_Active,
Position := Rel2D3_Pos,
Velocity := Rel2D3w 1.108 0 Td[(E)8.02 e
```

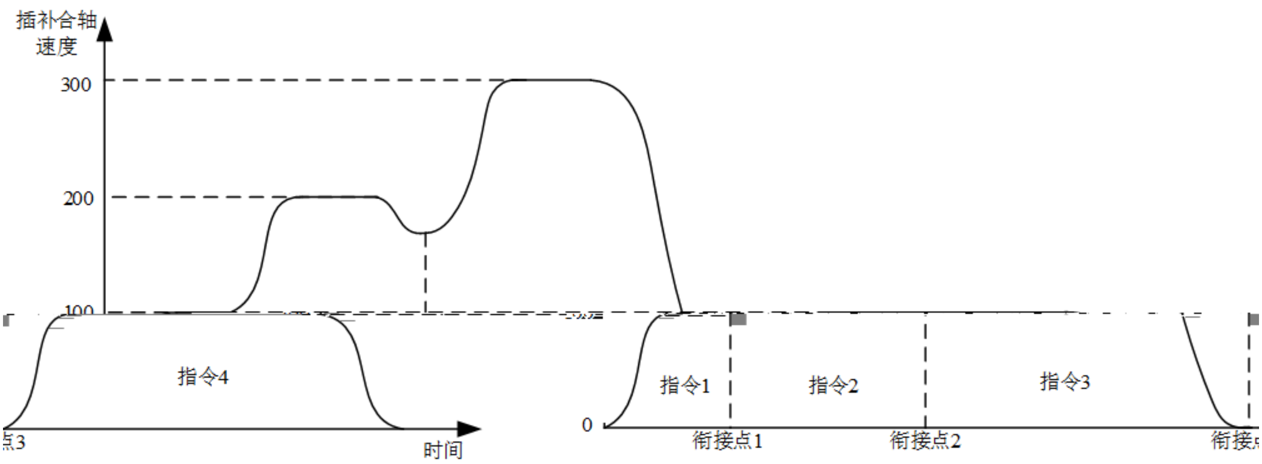
```
    TransitionParameter := Abs2D1_Param,
    AxesGroup := AxesGroup,
    Done=> Abs2D1_Done,
    Busy=> Abs2D1_Busy,
    Active=> Abs2D1_Active,
    CommandAborted=> Abs2D1_ComdAbort,
    Error=> Abs2D1_Err,
    ErrorID=> Abs2D1_ErrID);
END_IF
```

20° 60° 1 2

1 2 3

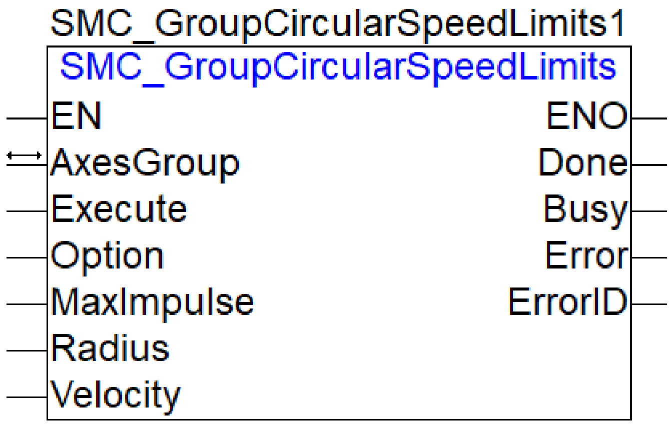
2 3 4

3 0



7.9.4

7.10 SMC_GroupCircularSpeedLimits



7.10.1

IN_OUT	AxesGroup			-	-	AXES_GROUP_REF
INPUT	Execute			-	TRUE/FALSE	BOOL
	Option	0 1		0	0/1	USINT
	MaxImpulse	Option 0		1E100		LREAL
	Radius	Option 1		1		LREAL
	Velocity	Option 1		1E30		LREAL
OUTPUT	Done			FALSE	TRUE/FALSE	BOOL
	Busy			FALSE	TRUE/FALSE	BOOL
	Error			FALSE	TRUE/FALSE	BOOL

	ErrorID			0	-	SMC_ERROR

7.10.2

1

Execute AxesGroup

ID Option MaxImpulse Radius Velocity

2

Option 2 Option 0

Option 1

Option 0 MaxImpulse

Radius Velocity

Option 1 Radius

Velocity MaxImpulse

MaxImpulse Option 0

Option 1 Velocity

Curvature MaxImpulse MaxImpulse $=Curvature^2 * dSpeed^3$

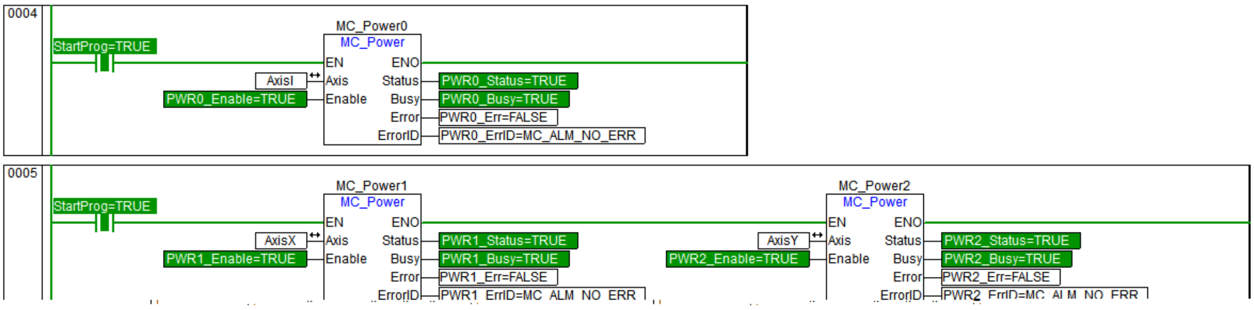
Curvature=1/Radius

1

Option 0

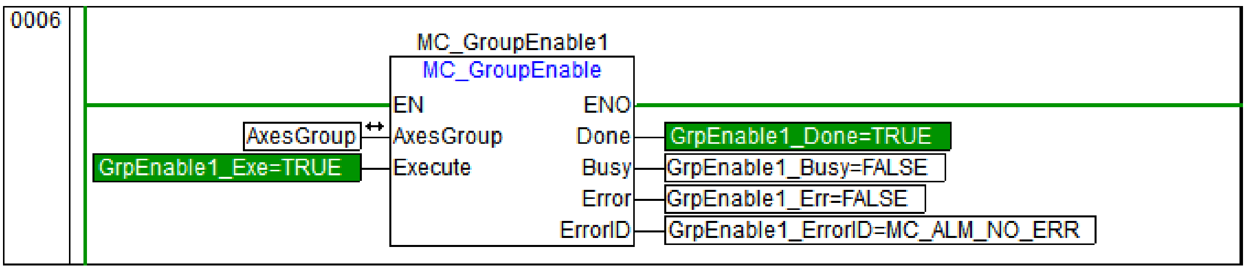
MaxImpulse 10000 10000

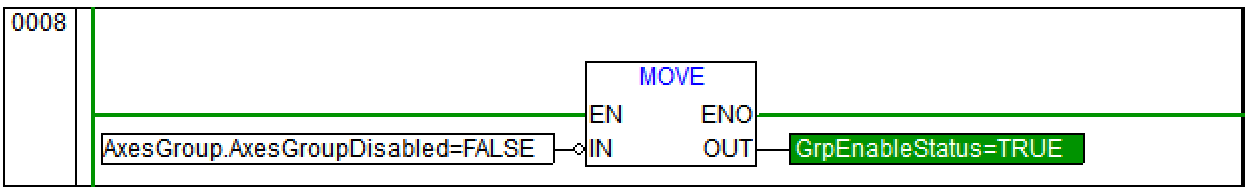
AxesGroup	AXES_GROUP_REF	-	
AxisI	AXIS_REF	-	
AxisX	AXIS_REF	-	X
AxisY	AXIS_REF	-	Y
StartProg	BOOL	FALSE	TRUE
Power1_Status	BOOL	FALSE	X X TRUE
Power2_Status	BOOL	FALSE	Y Y TRUE
Power0_Status	BOOL	FALSE	TRUE
GrpEnableStatus	BOOL	TRUE	TRUE
GrpEnable1_Exe	BOOL	FALSE	
Circ2D1_Exe	BOOL	FALSE	



4 MC_GroupEnable AxesGroupDisabled FALSE

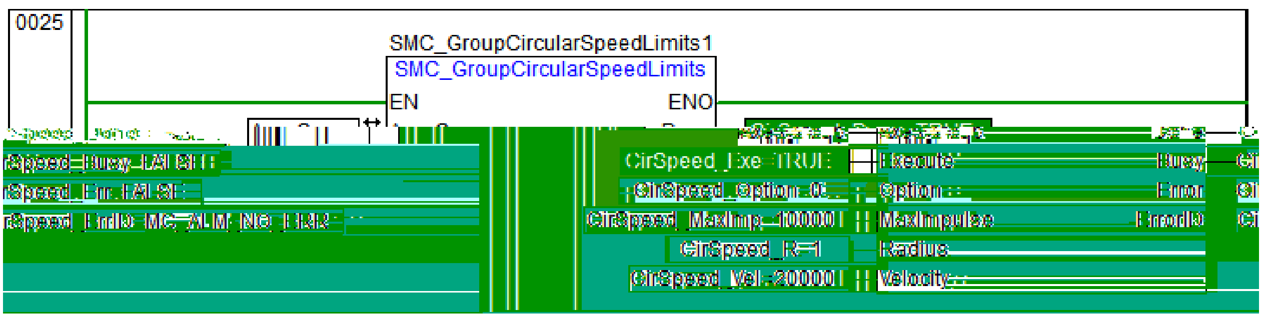
0006 0008 AxesGroupDisabled GrpEnableStatus



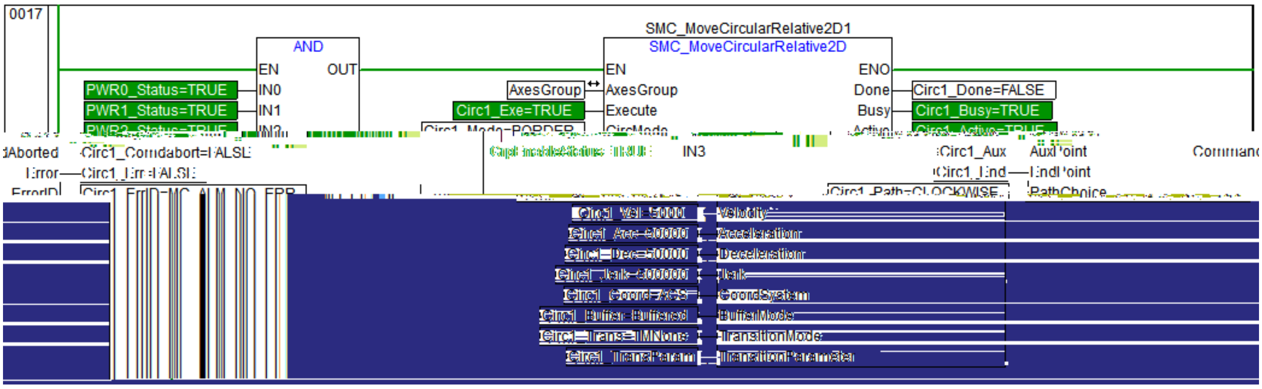


5 0025 SMC_GroupCircularSpeedLimits

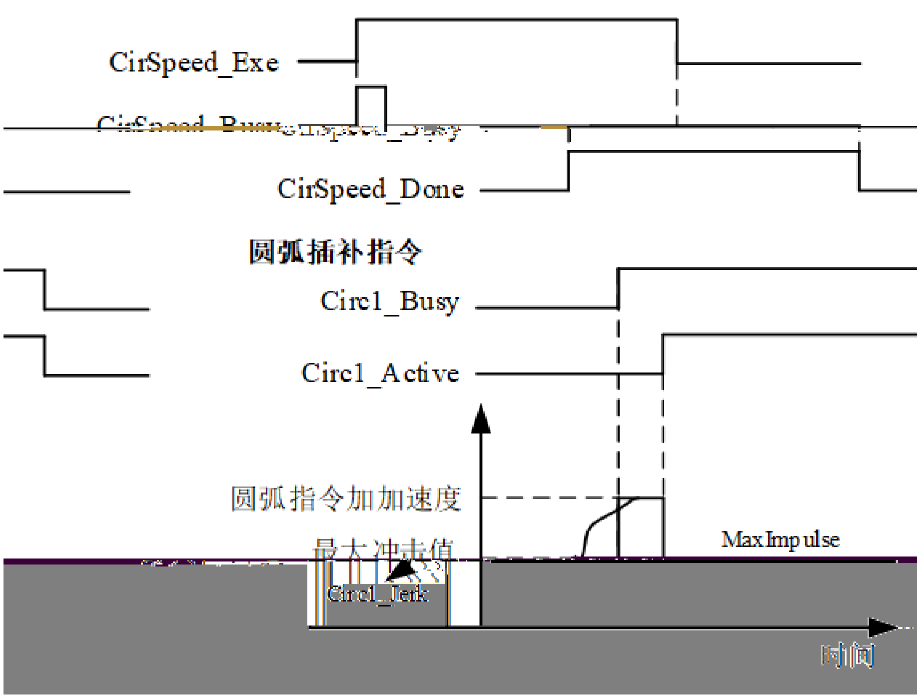
Jerk MaxImpulse 10000



6 0017



SMC_GroupCircularSpeedLimits指令



ST

SMC_GroupCircularSpeedLimits Option 1 1000

Velocity 1000 MaxImpulse = 1/1000 ² * 1000³ = 1000

SMC_MoveCircularRelative2D 1000

5000 50000 500000

1000 1000 MaxImpulse

AxesGroup	AXES_GROUP_REF	-	
AxisI	AXIS_REF	-	

```
    Circ1_ACC:=5000;
    Circ1_DEC:=5000;
    Circ1_Jerk:=50000;
    Circ1_Buffer:=Buffered;
    CirSpeed_Option:=1;
    CirSpeed_R:=1000;
    CirSpeed_Vel:=1000;
    InitFlag:=TRUE;
END_IF
```

3 SMC_AxisInit 0 Virtual

AxisX/AxisY 50 EtherCAT_Position SMC_AxisInit

4 MC_Power

```
IF StartPro THEN
    MC_Power0(                    (*        *)
    Enable := PWR0_Enable,
    Axis := AxisI,
    Status=> PWR0_Status,
    Busy=> PWR0_Busy,
    Error=> PWR0_Err,
    ErrorID=> PWR0_ErrID);
    MC_Power1(                    (*    AxisX    *)
    Enable := PWR1_Enable,
    Axis := AxisX,
    Status=> PWR1_Status,
    Busy=> PWR1_Busy,
    Error=> PWR1_Err,
    ErrorID=> PWR1_ErrID);
    MC_Power2(                    (*    AxisY    *)
    Enable := PWR2_Enable,
    Axis := AxisY,
    Status=> PWR2_Status,
    Busy=> PWR2_Busy,
    Error=> PWR2_Err,
```

```
    ErrorID=> PWR2_ErrID);
END_IF
```

5

```
MC_GroupEnable1(
Execute := GrpEnable_Exe,
AxesGroup := AxesGroup,
Done=> GrpEnable_Done,
Busy=> GrpEnable_Busy,
Error=> GrpEnable_Err,
ErrorID=> GrpEnable_ErrorID);
GrpEnableStatus:= AxesGroup. AxesGroupDisabled;
```

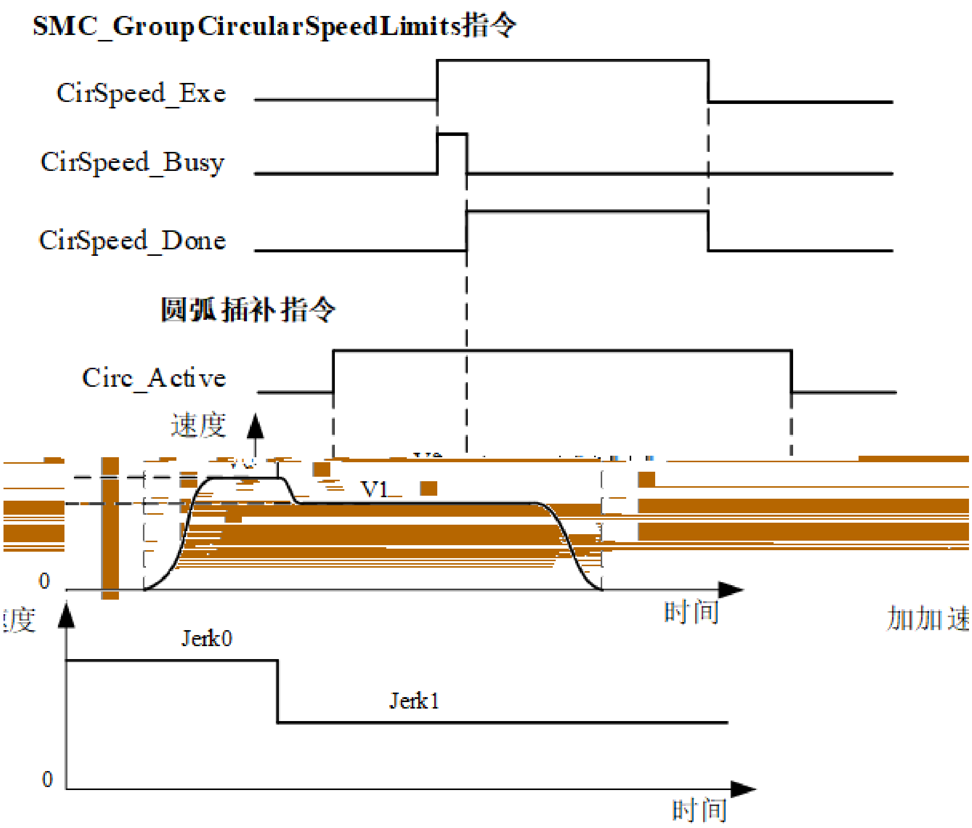
6

```
IF PWR0_Status  AND PWR1_Status  AND PWR2_Status  AND GrpEnableStatus THEN
```

3T/P AMCID3

```
SMC_GroupCircularSpeedLimits1(
Execute := CirSpeed_Exe,
Option := CirSpeed_Option,
MaxImpulse := CirSpeed_MaxImp,
Radius := CirSpeed_R,
Velocity := CirSpeed_Vel,
AxesGroup := AxesGroup,
Done=> CirSpeed_Done
Busy=> CirSpeed_Busy,
Error=> CirSpeed_Err,
ErrorID=> CirSpeed_ErrID);
```

8



7.10.4

MaxImpulse Radius Velocity

8

8.1

0	MC_ALM_NO_ERR	
2000	MC_ERR_AXIS_TYPE_UNUSED	/
2001	MC_ERR_AXISID_INVALID	ID / ID
2002	MC_ERR_AXISID_IS_REPEATED	/
2003	MC_ERR_AXIS_TYPE_NOT_SUPPORTED	/
2004	MC_ERR_SLAVEID_INVALID	ID / ID
2005	MC_ERR_SLAVE_OFFLINE	/ ID
2006	MC_ERR_SLAVE_FAULT	/ ID
2007	MC_ERR_SYSTEM_ERR	
2008	MC_ERR_AXIS_ERR	/
2009	MC_ERR_AXIS_MOTION_IS_REJECTED_IN_STOPPING	Stopping
2010	MC_ERR_AXIS_STATE_ERROR_STOP	/
2011	MC_ERR_AXIS_DISABLE	/
2012	MC_ERR_PDO_SETTING_MISSING	PDO

[redacted]

2204	MC_ERR_AXES_GROUP_ENABLED_FAILURE	
2950	MC_ERR_PDO_UNMAPPED_FOR_STATUS_WORD	PDO Statusword
2951	MC_ERR_PDO_UNMAPPED_FOR_ACTUAL_POS	PDO Position actual value
2952	MC_ERR_PDO_UNMAPPED_FOR_TOUCH_PROBE_STATUS	PDO Touch probe status
2953	MC_ERR_PDO_UNMAPPED_FOR_TOUCH_PROBE_POS_POS1	PDO Touch probe pos1 pos value
2954	MC_ERR_PDO_UNMAPPED_FOR_TOUCH_PROBE_NEG_POS1	PDO Touch probe pos1 neg value
2955	MC_ERR_PDO_UNMAPPED_FOR_TOUCH_PROBE_POS_POS2	PDO Touch probe pos2 pos value
2956	MC_ERR_PDO_UNMAPPED_FOR_TOUCH_PROBE_NEG_POS2	PDO Touch probe pos2 neg value
2957	MC_ERR_PDO_UNMAPPED_FOR_DIGITAL_INPUTS	PDO Digital inputs
2958	MC_ERR_PDO_UNMAPPED_FOR_OPERATE_MODE_DISPLAY	PDO Modes of operation display
2959	MC_ERR_PDO_UNMAPPED_FOR_ACTUAL_VELOCITY	PDO Velocity actual value
2960	MC_ERR_PDO_UNMAPPED_FOR_ACTUAL_TORQUE	PDO Torque actual value
2961	MC_ERR_PDO_UNMAPPED_FOR_ERROR_CODE	[redacted]

[redacted]

3004	MC_ERR_AXIS_INIT_HMSW_SWITCH_CHANNEL_NUM_ERR	
3005	MC_ERR_AXIS_INIT_HMSW_SWITCH_CHANNEL_TYPE_ERR	
3006	MC_ERR_AXIS_INIT_POS_LIMIT_SWITCH_NUM_ERR	
3007	MC_ERR_AXIS_INIT_NEG_LIMIT_SWITCH_NUM_ERR	
3008	MC_ERR_AXIS_INIT_HOME_IN_SWITCH_NUM_ERR	
3009	MC_ERR_AXIS_INIT_SET_USER_UNITS_ERR	
3010	MC_ERR_AXIS_INIT_AXIS_TYPE_CFG_TIMEOUT	
3011	MC_ERR_AXIS_INIT_SET_AXIS_TYPE_REPEAT	
3012	MC_ERR_AXIS_INIT_SET_AXIS_TYPE_TIMING_ERR	
3013	MC_ERR_AXIS_INIT_SET_OPERATION_MODE_ERR	
3014	MC_ERR_AXIS_INIT_NOT_ECAT_ENCODER_AXIS_TYPE	EtherCAT_Encoder
3015	MC_ERR_NEG_LIM_CHAN_AND_POS_LIM_CHAN_REUSED	
3016	MC_ERR_DRIVER_CHAN_CONFLICT_WITH_COMM_DI_CHAN	DI DI
3017	MC_ERR_NEG_LIM_CHAN_CONFLICT_WITH_POS_LIM	
3018	MC_ERR_POS_LIM_CHAN_CONFLICT_WITH_NEG_LIM	
3030	MC_ERR_AXIS_POWER_MOD_DISP_TIMEOUT	
3040	MC_ERR_HOME_APPROACH_VELOCITY_INVALID	APPROACH VELOCITY
3041	MC_ERR_HOME_CAPTURE_CHL_INVALID	
3042	MC_ERR_HOME_CAPTURE_CHL_BUSY	
3043	MC_ERR_HOME_MODE_Z_NOT_SUPPORTED	Z
3044	MC_ERR_HOME_HOME_MODE_INVALID	
3045	MC_ERR_HOME_AXIS_TYPE_FOR_Z_ERR	Z EtherCAT
3046	MC_ERR_HOME_HOME_CHL_CONFIG_ERR	
3047	MC_ERR_HOME_POT_CHL_CONFIG_ERR	
3048	MC_ERR_HOME_NOT_CHL_CONFIG_ERR	
3049	MC_ERR_HOME_ALL_DI_CONFIG_ERR	DI
3090	MC_ERR_MOVE_RELATIVE_DIST_INVALID	
3100	MC_ERR_MCMD_NOT_SYNC_MOVE_VELOCITY	SyncMoveVelocity
3101	MC_ERR_SYNC_MOVE_VELOCITY_CMD_DATA_NULL	SyncMoveVelocity
3120	MC_ERR_SET_JERKM_MODE_INVALID	
3130	MC_ERR_TORQUE_CONTROL_TORQUE_RAMP_INVALID	
3131	MC_ERR_TORQUE_CONTROL_TORQUE_INVALID	
3132	MC_ERR_TORQUE_CONTROL_VELOCITY_INVALID	

3133	MC_ERR_TORQUE_CONTROL_MAX_VELOCITY_CONFIG_ERR	
3160	MC_ERR_SET_OVERRIDE_VEL_FACTOR_INVALID	
3161	MC_ERR_SET_OVERRIDE_ACC_FACTOR_INVALID	
3162	MC_ERR_SET_OVERRIDE_JERK_FACTOR_INVALID	
3170	MC_ERR_MAX_VELOCITY_INVALID	
3171	MC_ERR_MAX_ACCELERATION_INVALID	
3172	MC_ERR_MAX_DECELERATION_INVALID	
3173	MC_ERR_MAX_JERK_INVALID	
3180	MC_ERR_READ_DIGITAL_INPUT_NUM_INVALID	
3190	MC_ERR_READ_LIMIT_STATUS_SOFT_LIMIT_PARA_ERR	
3191	MC_ERR_READ_LIMIT_STATUS_EACH_STATUS_CONFLICT	
3210	MC_ERR_RESET_CMD_BACK_INVALID	
3211	MC_ERR_RESET_AXIS_TIME_OUT_ERR	
3212	MC_ERR_RESET_AXIS_WAIT_TIME_OUT_ERR	
3220	MC_ERR_SET_TORQUE_LIMIT_POS_VALUE_INVALID	
3221	MC_ERR_SET_TORQUE_LIMIT_NEG_VALUE_INVALID	
3222	MC_ERR_SET_TORQUE_LIMIT_SLAVE_INVALID	
3223	MC_ERR_SDO_WRITE_POS_TORQUE_LIMITERR	SDO
3224	MC_ERR_SDO_WRITE_NEG_TORQUE_LIMITERR	SDO
3225	MC_ERR_SDO_WRITE_POS_TORQUE_LIMITOVERRUN	
3226	MC_ERR_SDO_WRITE_NEG_TORQUE_LIMITOVERRUN	
3230	MC_ERR_TOUCH_PROBE_LATCH_MODE_INVALID	
3231	MC_ERR_TOUCH_PROBE_LATCH_ID_INVALID	ID
3232	MC_ERR_TOUCH_PROBE_TRIGGER_SIGNAL_INVALID	
3233	MC_ERR_TOUCH_PROBE_TRIGGER_EDGE_INVALID	
3234	MC_ERR_TOUCH_PROBE_WINDOW_CONFIG_ERR_1	1 First Last
3235	MC_ERR_TOUCH_PROBE_WINDOW_CONFIG_ERR_2	2 1
3236	MC_ERR_TOUCH_PROBE_WINDOW_CONFIG_ERR_3	3 2
3237	MC_ERR_TOUCH_PROBE_IS_REUSED	
3238	MC_ERR_TOUCH_PROBE_SET_USED_ERR	
3239	MC_ERR_TOUCH_PROBE_SYSTEM_ERR	
3240	MC_ERR_TOUCH_PROBE_PARA_COMBINE_ERR	

3241	MC_ERR_TOUCH_PROBE_CHANNEL_INNER_USED	
3242	MC_ERR_TOUCH_PROBE_FB_IS_USING	
3243	MC_ERR_TOUCH_PROBE_PDO_CONFIG_CHECK_ERR	PDO
5000	MC_ERR_GEAR_RATIO_DENOMINATOR_INVALID	0
5001		

5151	MC_ERR_GANTRY_AXISARRY_OR_DICHLARRT_IS_NULL	DI
5152	MC_ERR_GANTRY_AXIS_TYPES_ARE_DIFFERENT	
5153	MC_ERR_GANTRY_AXIS_TYPE_SET_ERR	
5154	MC_ERR_GANTRY_INIT_INPUT_DICHL_REPEATED	
5155	MC_ERR_GANTRY_INIT_HOME_MODE_SET_ERR	33 34 35
5156	MC_ERR_GANTRY_INIT_AXIS_POS_FE_OVERRIDE	
5157	MC_ERR_GANTRY_INIT_INPUT_PARA_POSERR_SET_ERR	dPosErr 0
5158	MC_ERR_GANTRY_AXIS_NOT_REPMODE_0	
5159	MC_ERR_GANTRY_INIT_NOT_ECAT_AXIS_FOR_Z	Z EtherCAT
6010	MC_ERR_CIRC_INTERP_MODE_INVALID	
6011	MC_ERR_CIRC_MODE_INVALID	
6012	MC_ERR_CIRC_PATHCHOICE_INVALID	
6020	MC_ERR_CIRC_SPEEDLIMITS_OPTION_INVALID	
6021	MC_ERR_CIRC_SPEEDLIMITS_MAXIMPULSE_INVALID	
6022	MC_ERR_CIRC_SPEEDLIMITS_RADIUS_INVALID	
6023	MC_ERR_CIRC_SPEEDLIMITS_VELOCITY_INVALID	
6030	MC_ERR_CORNER_SPEEDLIMITS_DECANGLE_INVALID	
6031	MC_ERR_CORNER_SPEEDLIMITS_STOPANGLE_INVALID	

8.2

8.2.1

MC_BUFFER_MODE

0	Aborting	
1	Buffered	
2	BlendingLow	
3	BlendingPrevious	

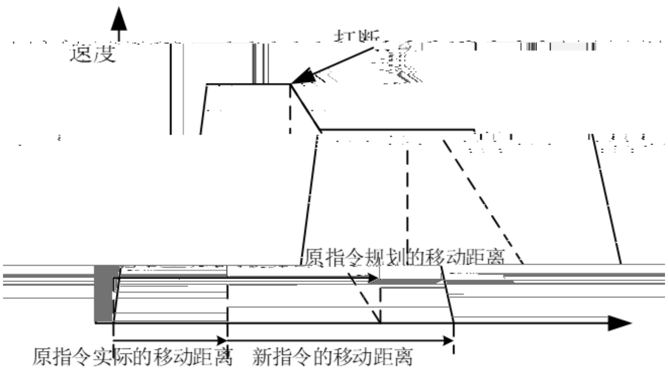
4	BlendingNext	
5	BlendingHigh	

8.2.1.1

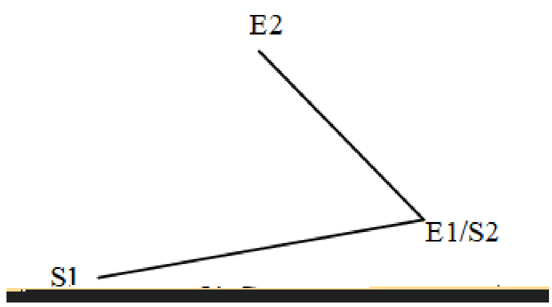
Aborting( )

Position Velocity

Acceleration Deceleration Jerk

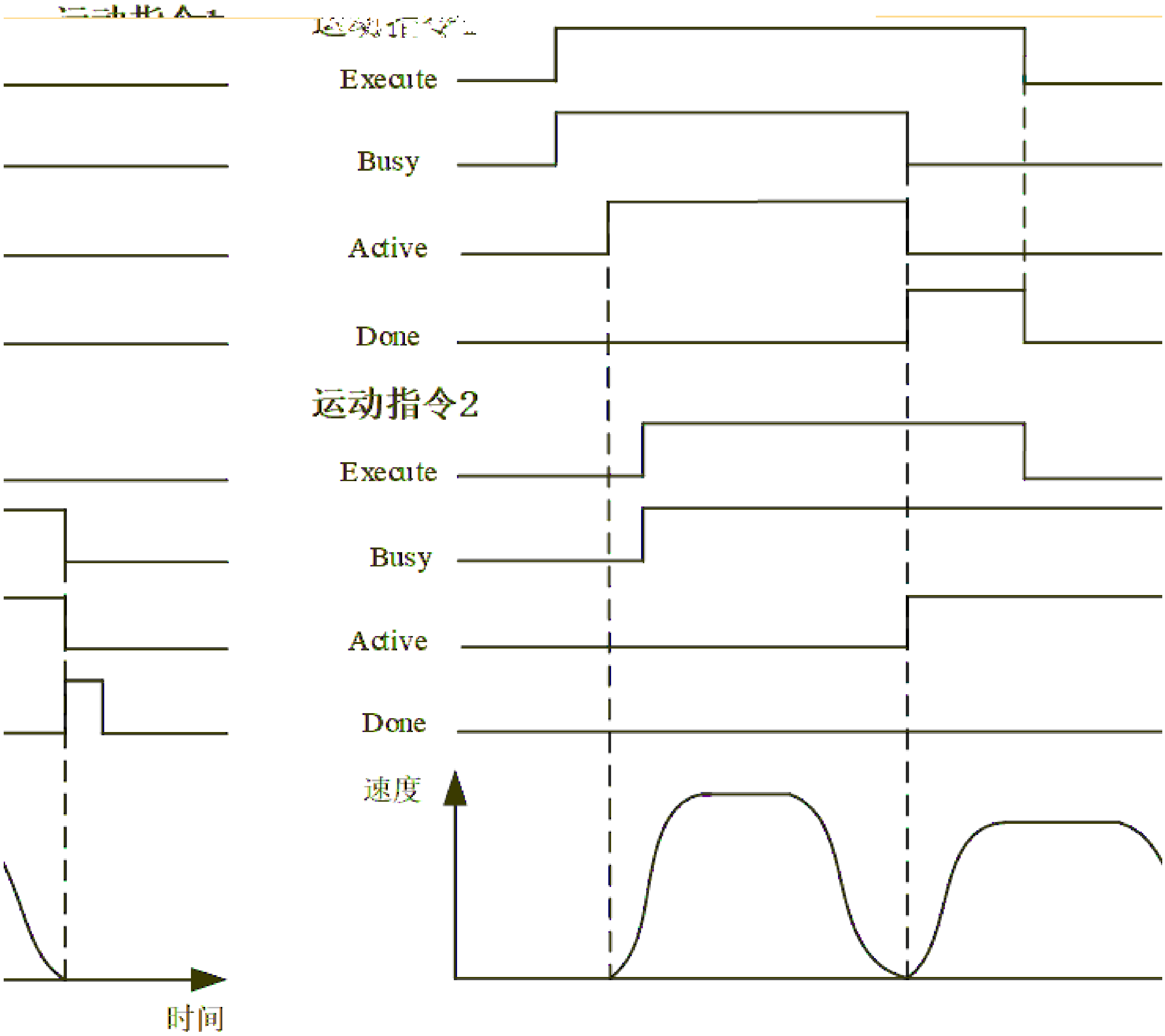


8.2.1.2



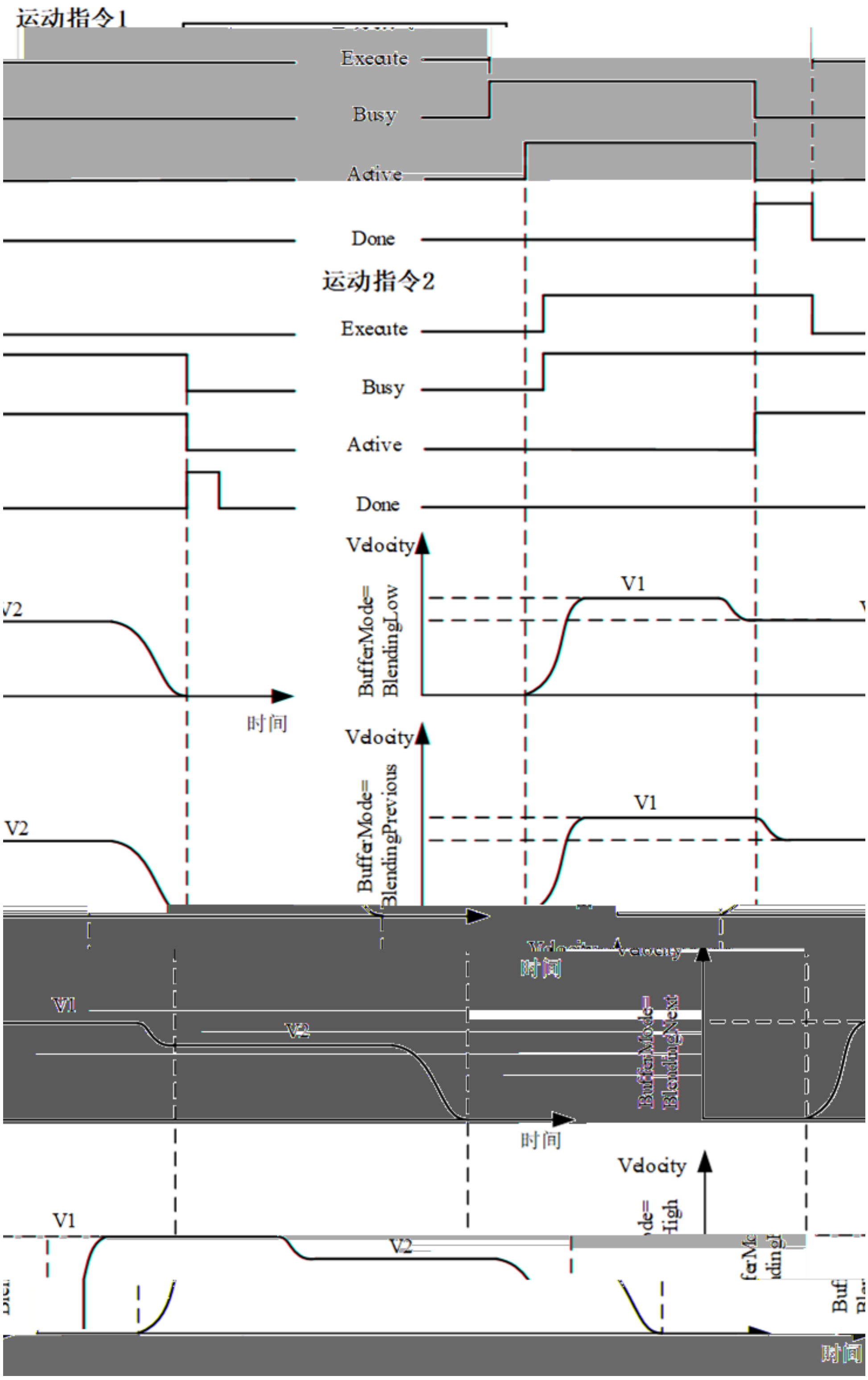
- S1:
- E1 :

- S2:
- E2 :



运动指令2
Execute
Busy
Active
Done
运动指令2
Execute
Busy
Active
Done
速度
时间

8.2.1.3


运动指令1
Execute
Busy
Active
Done
运动指令2
Execute
Busy
Active
Done
Velocity
BufferMode=
BlendingLow
V1
V2
时间
Velocity
BufferMode=
BlendingPrevious
V1
V2
时间
Velocity
BufferMode=
BlendingNext
V1
V2
时间
Velocity
V1
V2
时间

Beijing HollySys Intelligent Technologies Co., Ltd..

2

100176

010-5898 1588

400-811-1999

010-5898 1558

http://www.hollysys.com