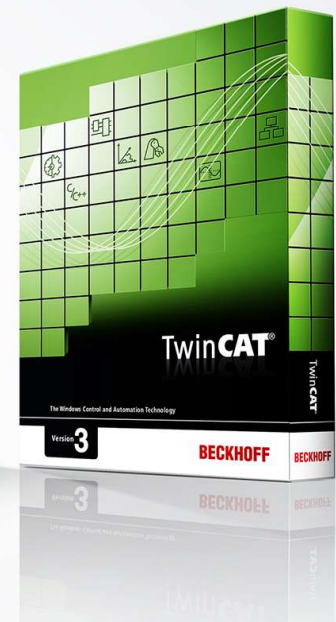
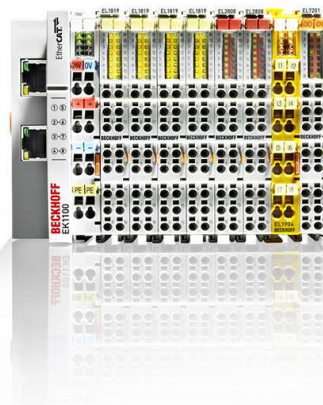


Motion Control Library Version 2

BECKHOFF

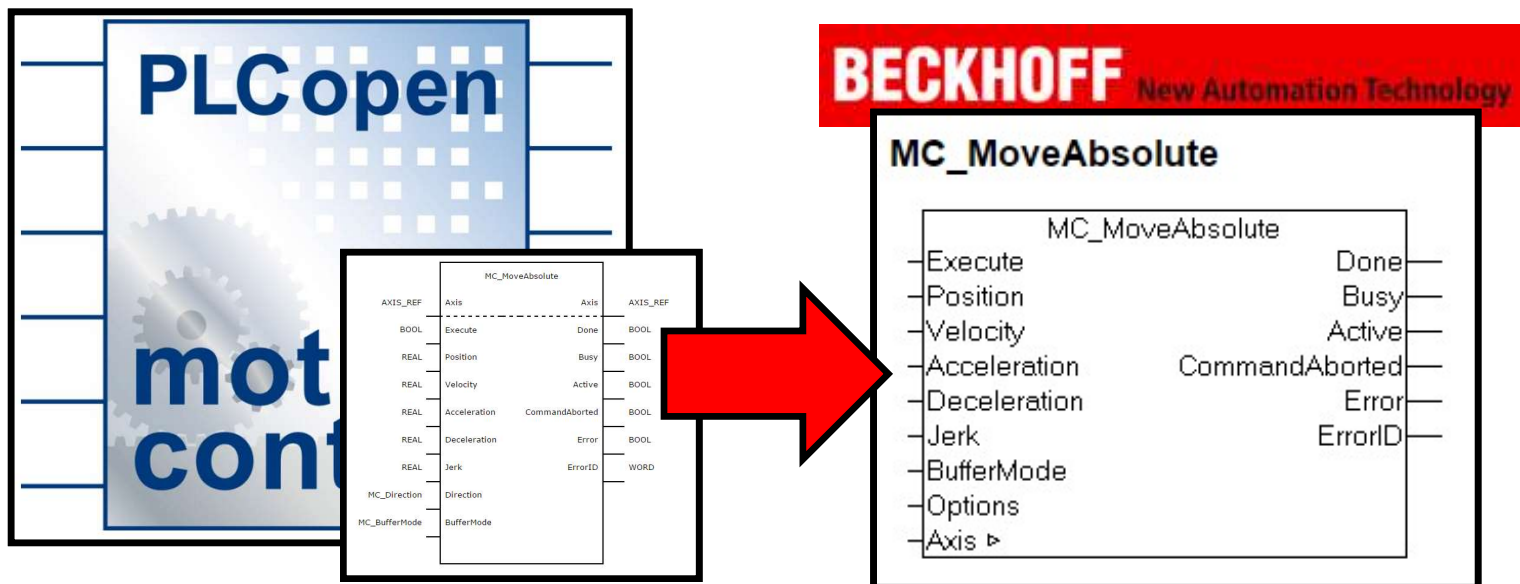


- **Introduction**
- Function Block Rules
- Type Axis_Ref
- Common Blocks
- Buffered Moves

- International Electrotechnical Commission(IEC) is the leading producer of global 'electrotechnology' standards
- IEC 61131-3:
 - standardizes PLC programming languages
 - Beckhoff conforms to this standard
 - provides a basis for PLCopen
- PLCopen is an organization working to extend and implement the 61131-3 for industrial automation

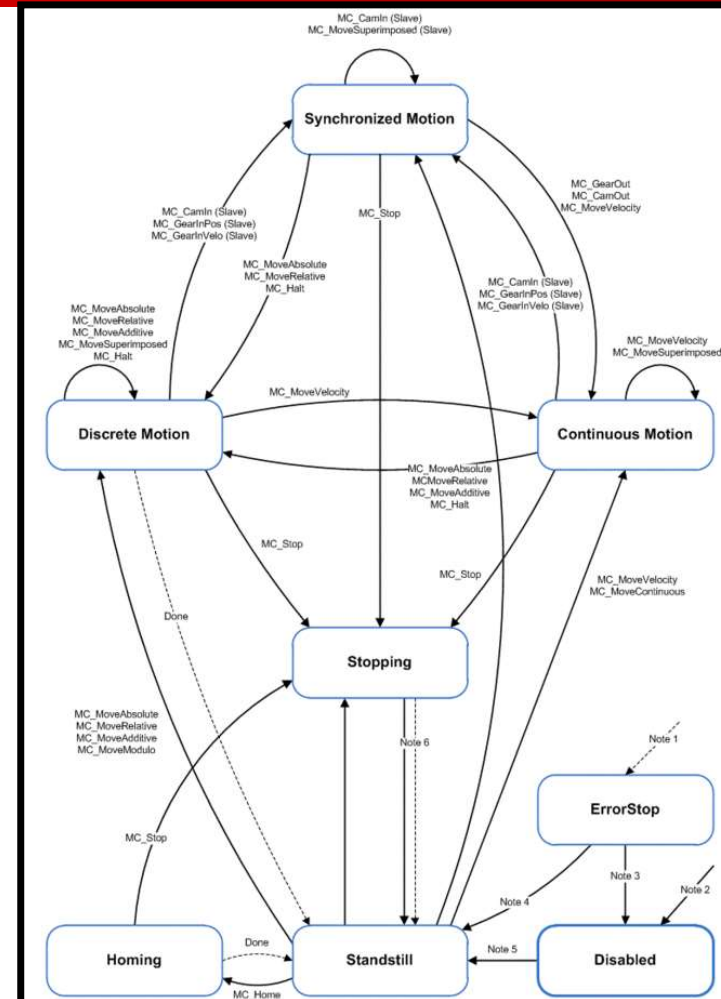


- PLCopen has created specifications for motion function blocks
 - Version 1.1 in April 2005
 - Version 2.0 in March 2011
- Beckhoff implements many of these blocks in its Motion Control Version 2 Library



Block Selection Aided with State Diagram

- Many Blocks available for:
 - Discrete movements
 - Continuous movements
 - Homing
 - Stopping
 - Synchronization
- [State diagram](#) available to aid in block selection



- Introduction
- **Function Block Rules**
- Type Axis_Ref
- Common Blocks
- Buffered Moves

- **Function Blocks May Have Components:**

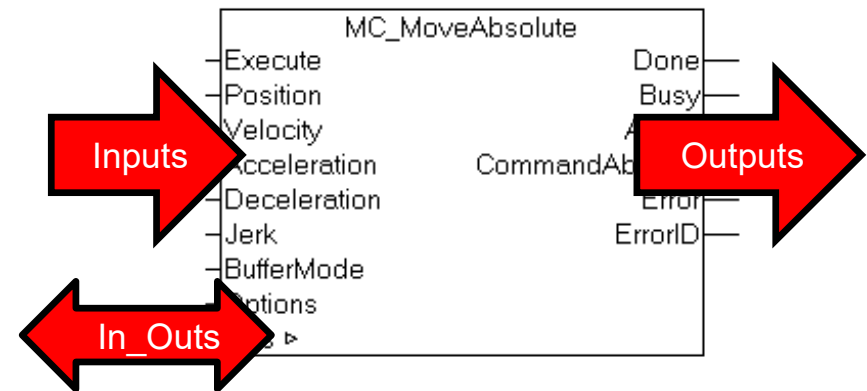
- Inputs
 - Passed by value
- Outputs
 - Passed by value
- In_Outs
 - Passed by reference

- **Block Memory is Retained:**

- While TwinCAT in Run Mode, internal block memory maintained

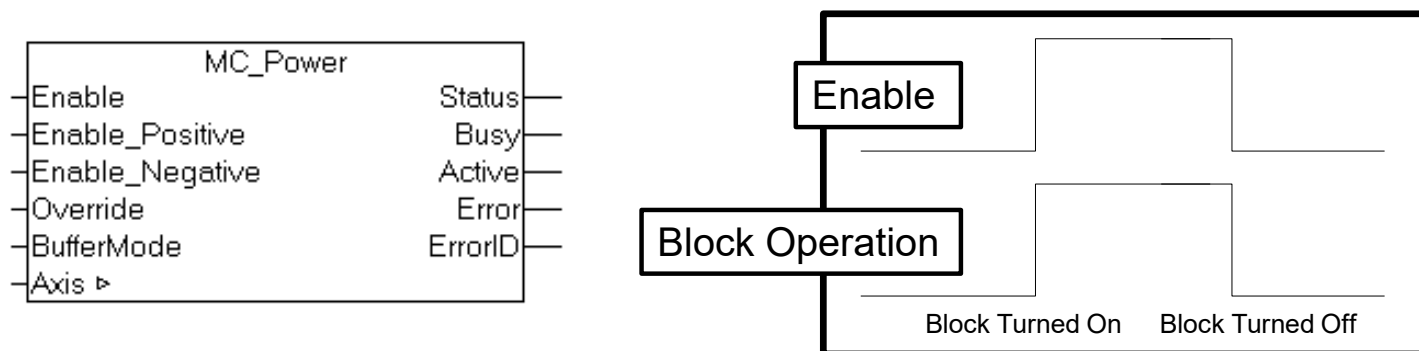
- **Blocks are Instantiated:**

- Many blocks of the same type can be used

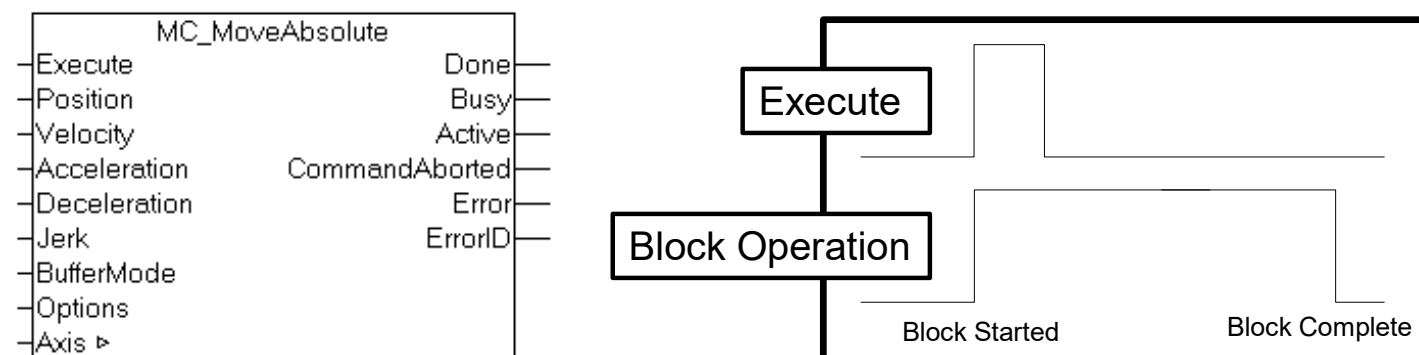


Most Blocks have an Enable or Execute

- Blocks with an Enable operate while the Enable is TRUE

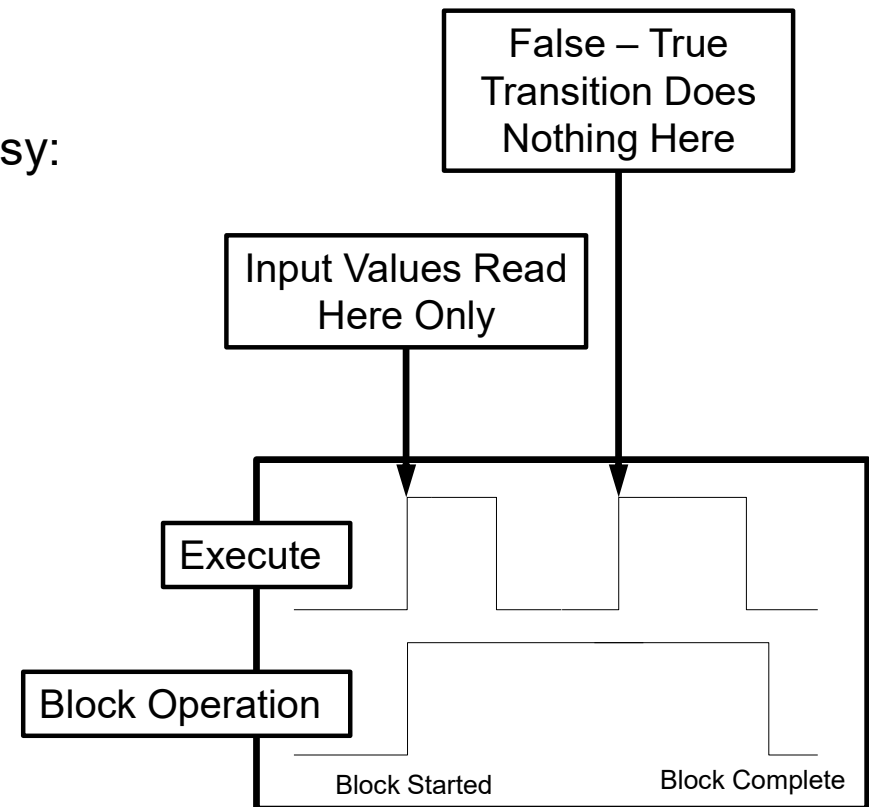


- Blocks with an Execute are triggered with a TRUE



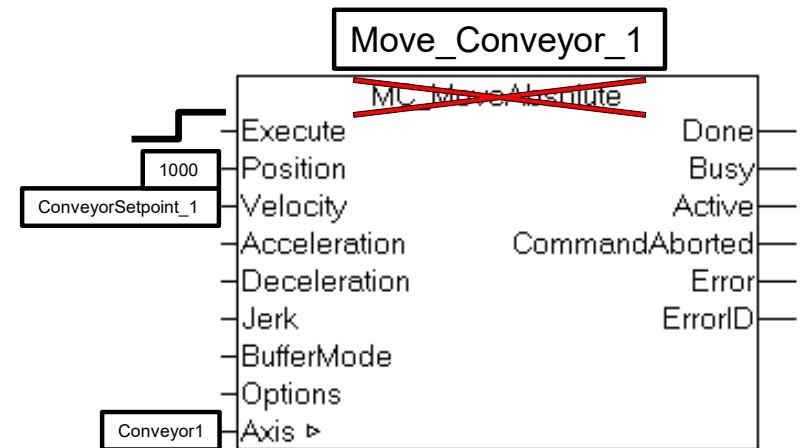
Repeated Block Execution has no Effect

- A rising edge will trigger a block:
 - Input values will be read in
- The block ignores any change in inputs while busy:
 - Retriggering the same block has no effect
 - Changing the inputs will be ignored
- After completion or error:
 - Retrigger block with rising edge to read new input values



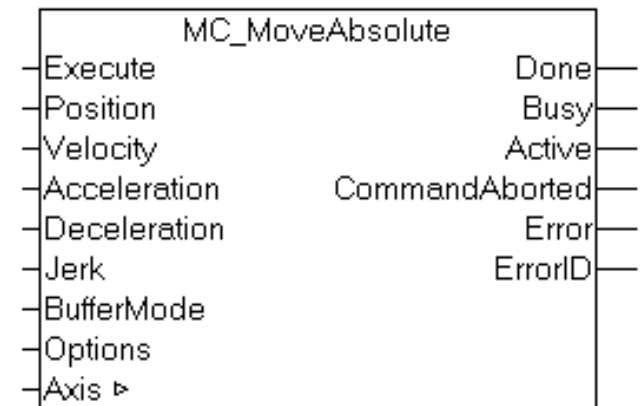
A Function Block Must be Called Correctly

- Only a function block **instance** may be called
- **In_Outs need** to be populated
- **Inputs** can be left blank, passed literal values, or passed variable values
 - Unspecified inputs will default to existing values
- **Outputs** can be variables or left blank
 - Unspecified outputs will not be passed at the time of call



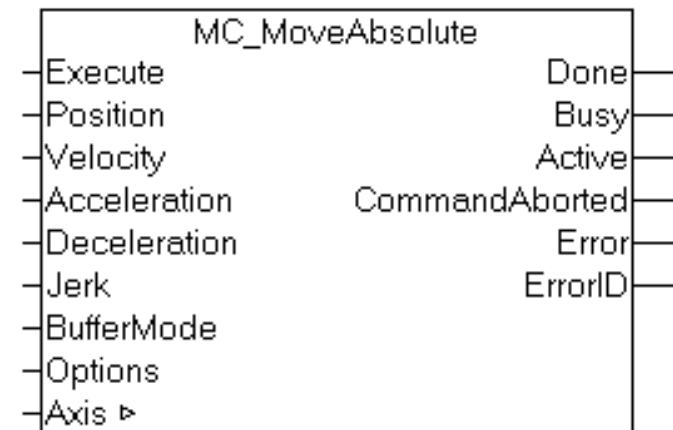
Common Block Outputs Have Specific Meanings

- **Done**
 - Block has successfully completed
- **Busy**
 - Block triggered, but not yet completed
- **Active**
 - Block triggered, axis currently responding
- **Command Aborted**
 - Block interrupted by another block
- **Error**
 - Something went wrong
- **ErrorID**
 - Code number of error



- The following outputs are exclusive:

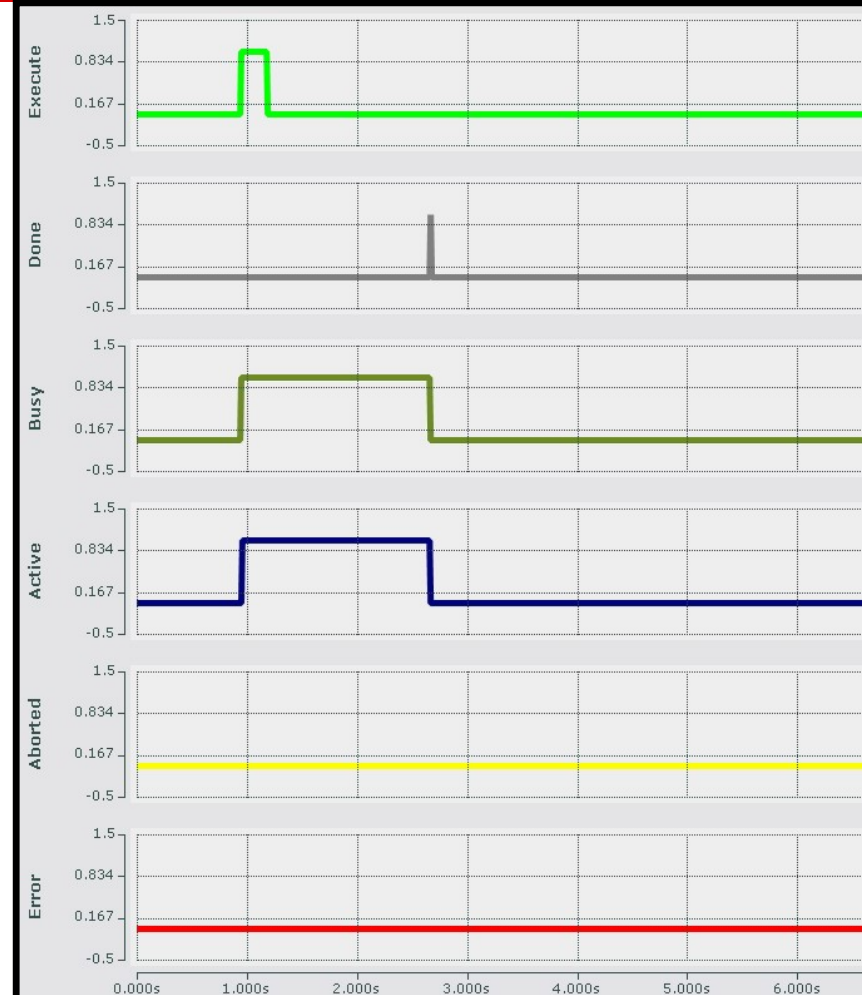
- **Done**
 - Successful completion
- **Busy/Active***
 - Block being processed
- **Error**
 - Something went wrong
- **Command Aborted**
 - Block interrupted



Single Scan Done Bit Results without Execute

BECKHOFF

1. Triggering the block causes it to be **Busy**
2. Each block remains **Busy** while the NC works on the request
3. When the NC completes the command, **Busy** turns FALSE
4. When **Busy** turns FALSE the **Done** bit turns TRUE for one scan if **Execute** FALSE



Function Block Rules

Done Bit Stays on While Execute On

- If completion successfully occurs, **Done** bit stays TRUE as long as **Execute** is TRUE

BECKHOFF



Single Scan Aborted Bit Results without Execute

BECKHOFF

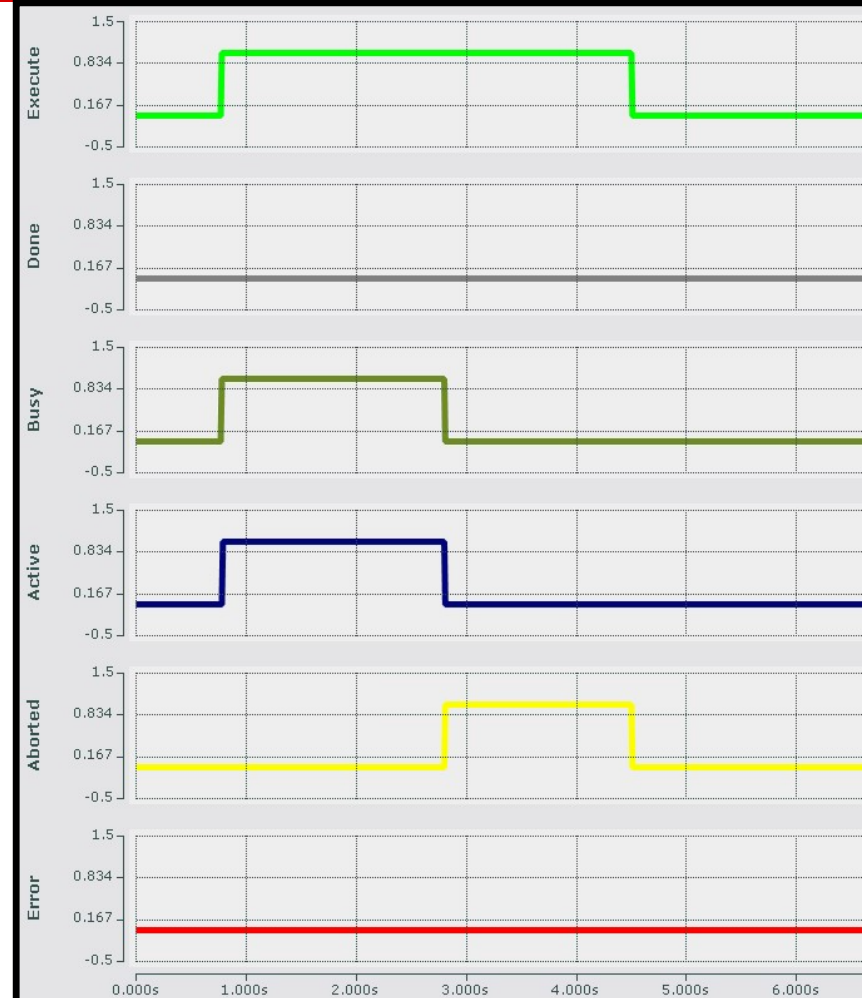
1. Triggering the block causes it to be **Busy**
2. Each block remains **Busy** while the NC works on the request
3. If a second block interrupts the active block, **Busy** will turn FALSE
4. **CommandAborted** will then turn TRUE to show interrupt occurred, but for one scan only if **Execute** FALSE



Single Scan Aborted Bit Results without Execute

BECKHOFF

- If block interrupted, **CommandAborted** stays TRUE as long as **Execute** is TRUE



Function Block Rules

Single Scan Error Bit Results without Execute

1. Triggering the block causes it to be **Busy**
2. Each block remains **Busy** while the NC works on the request
3. If a problem prevents block from completion, **Busy** turns FALSE
4. When **Busy** turns FALSE, **Error** turns TRUE for one scan only if **Execute is** FALSE

BECKHOFF



Sustained Execute Keeps Errors On

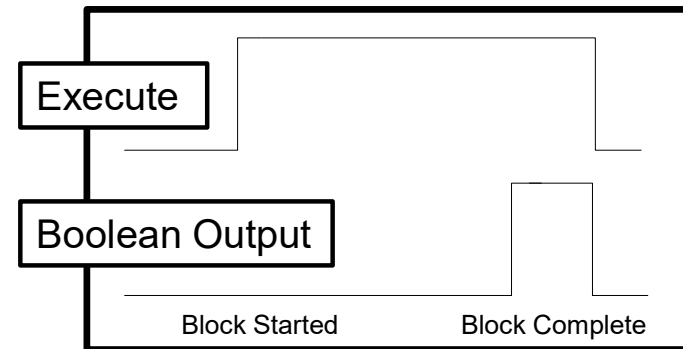
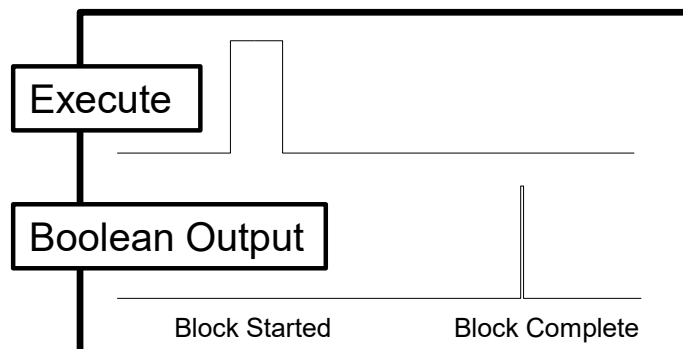
- If problem occurs, **Error** stays TRUE while **Execute** TRUE

BECKHOFF



Outputs, A General Summary

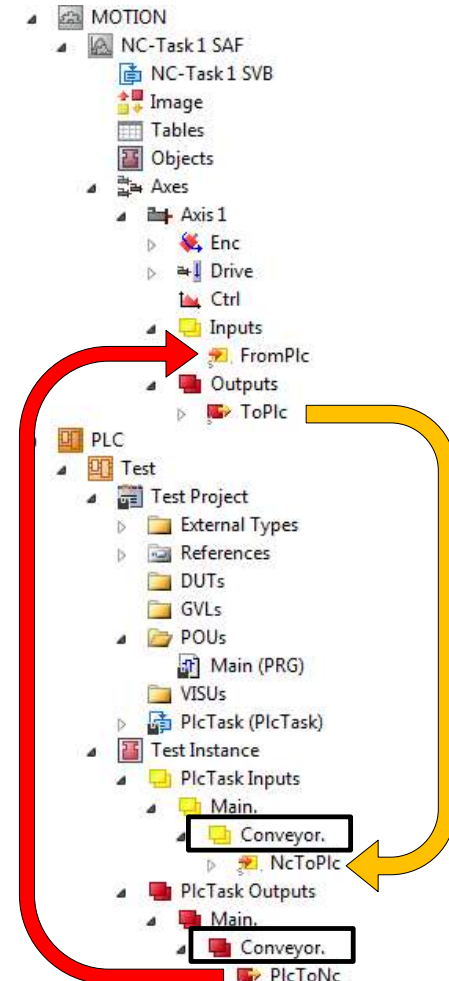
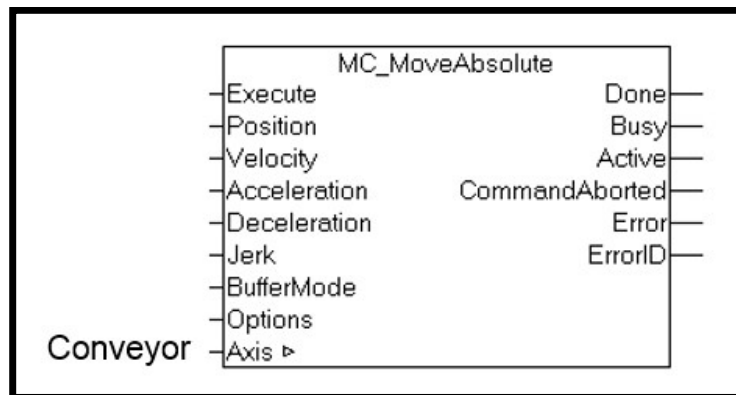
- Boolean Outputs are exclusive
 - **Busy** and **Active** can be on together
- A falling **Execute** resets these outputs:
 - The relevant outputs stays TRUE while Execute is TRUE
 - The output is TRUE for **one scan only** if Execute is FALSE
- A falling **Execute** has no effect on block execution



Axis_Ref Assigns an Axis to a Function Block

- Axis_Ref is a custom data type which refers a motion function block to a NC Axis
 - Block calls *require* Axis_Ref
- Axis_Ref contains process data to link the NC Axis to the PLC

```
PROGRAM MAIN  
VAR  
    Conveyor:AXIS_REF;  
END_VAR
```



- Introduction
- Function Block Rules
- **Type Axis_Ref**
- Common Blocks
- Buffered Moves

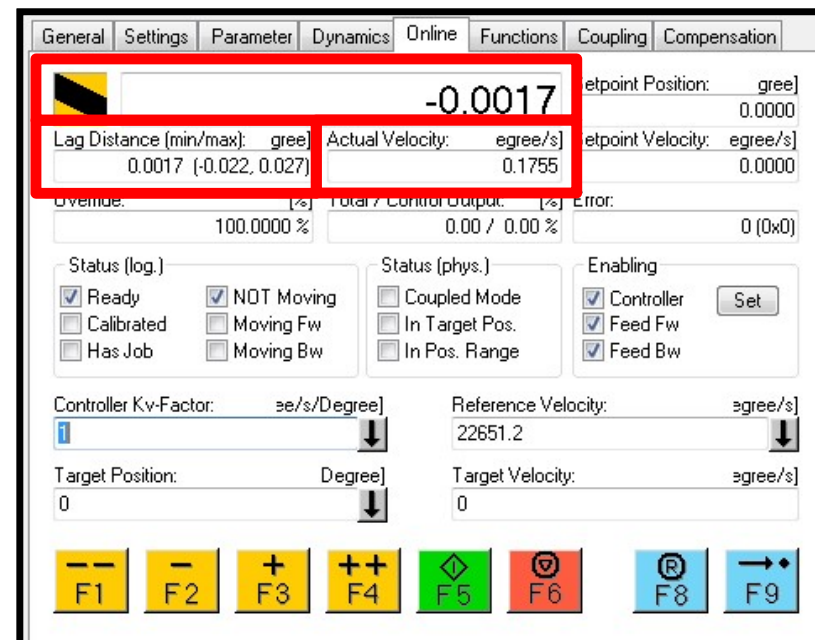
- Axis_Ref contains:
 - Basic structures for data transfer to and from the PLC
 - **PlcToNc** is an Output
 - **NcToPlc** is an Input
 - A **Status** structure with additional axis information

```
TYPE AXIS_REF :  
VAR_INPUT  
    PlcToNc AT %Q* : PLCTONC_AXIS_REF;  
END_VAR  
VAR_OUTPUT  
    NcToPlc AT %I* : NCTOPLC_AXIS_REF;  
    ADS      : ST_AdsAddress;  
    Status   : ST_AxisStatus;  
END_VAR  
END_TYPE
```

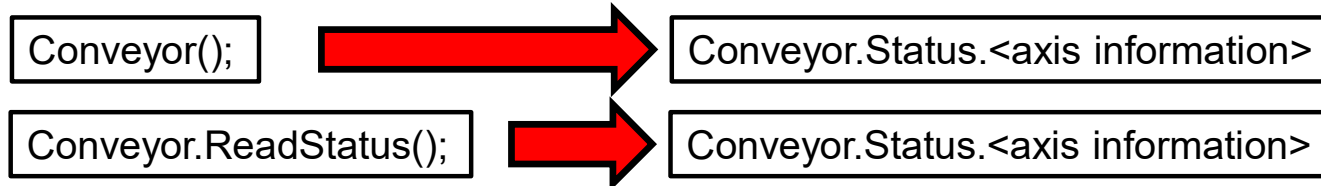
- Basic data read through main NcToPlc structure
 - No update request required

Conveyor.NcToPlc.<Axis Information>

- Information includes
 - Position
 - Velocity
 - Acceleration
 - Target Position
 - Lag Position
 - NC Error Code



- Call **Axis** instance OR **ReadStatus** to update **Status** structure
 - Call every PLC cycle to instantly update Status information



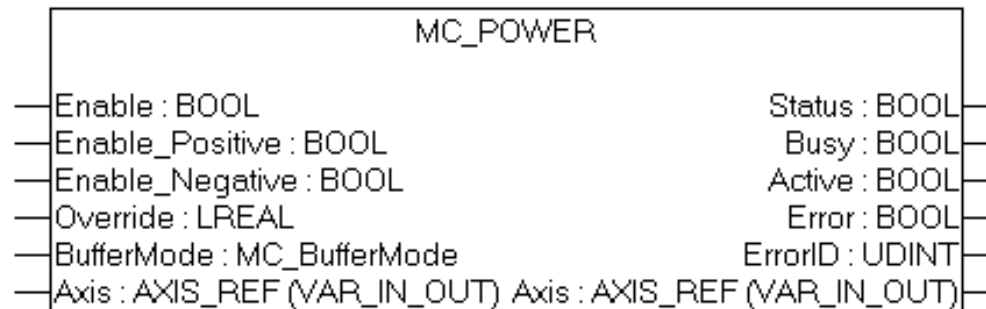
- Accelerating
- CamDataQueued
- CamScalingPending
- CamTableQueued
- CmdBuffered
- Compensating
- ConstantVelocity
- ContinuousMotion
- ControlLoopClosed
- Coupled
- CycleCounter
- Decelerating
- Disabled
- DiscreteMotion
- DriveDeviceError
- Error
- ErrorID
- ErrorPropagationDelayed
- ErrorStop
- ExternalLatchValid

- ExtSetPointGenEnabled
- HasBeenStopped
- HasJob
- Homed
- Homing
- HomingBusy
- InPositionArea
- InTargetPosition
- IoDataInvalid
- MotionCommandsLocked
- MotionState
- Moving
- NcApplicationRequest
- NcCycleCounter
- NegativeDirection
- NewTargetPosition
- NotMoving
- Operational
- OpMode
- PositiveDirection

- ProtectedMode
- PTPmode
- SoftLimitMaxExceeded
- SoftLimitMinExceeded
- StandStill
- Stopping
- SynchronizedMotion
- UpdateCycleTime
- UpdateTaskIndex

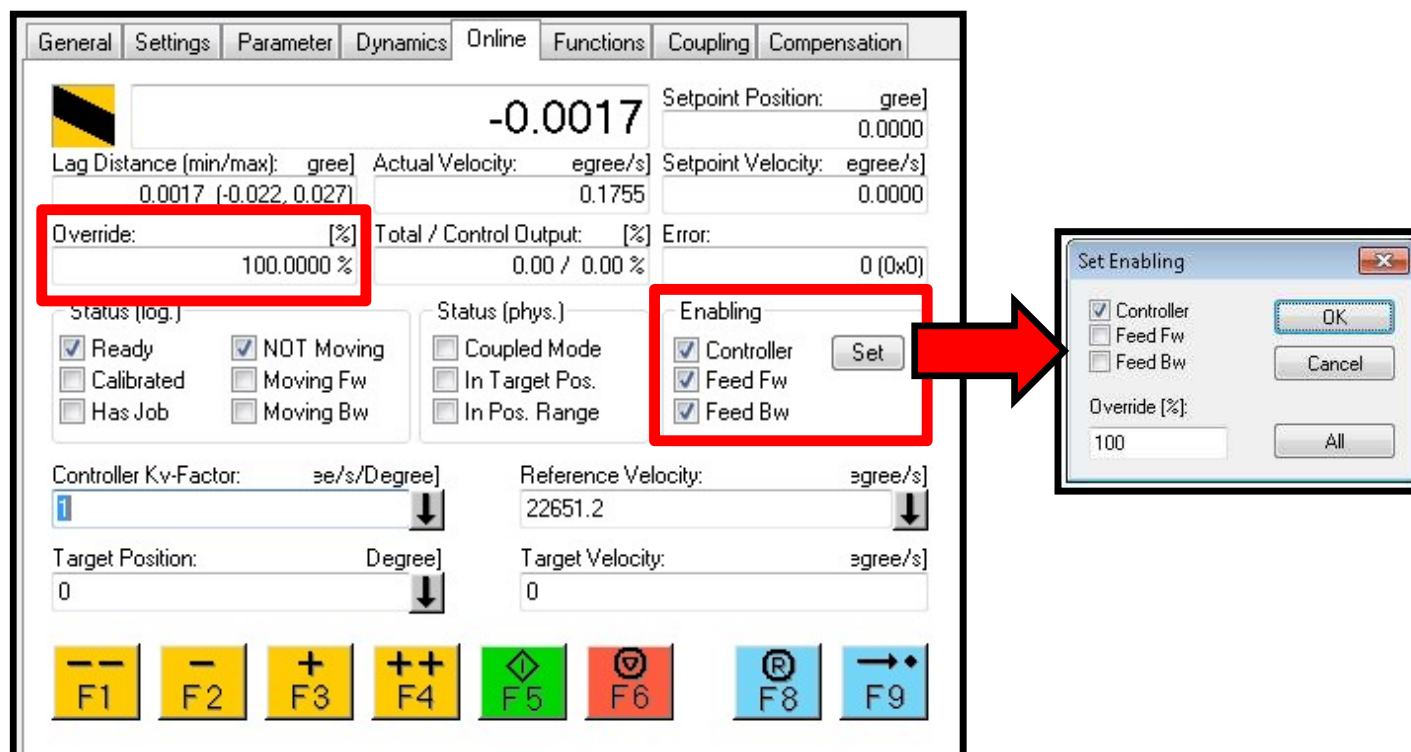
- Introduction
- Function Block Rules
- Type Axis_Ref
- **Common Blocks**
 - **Standard**
 - Informational
- Buffered Moves

- Inputs:
 - **Enable** is a general enable of the Axis
 - **Enable_Positive** allows forward motion
 - **Enable_Negative** allows backward motion
 - **Override** will scale the commanded velocity
 - 0 to 100 in Percent
- Outputs:
 - **Status** shows when the axis is enabled

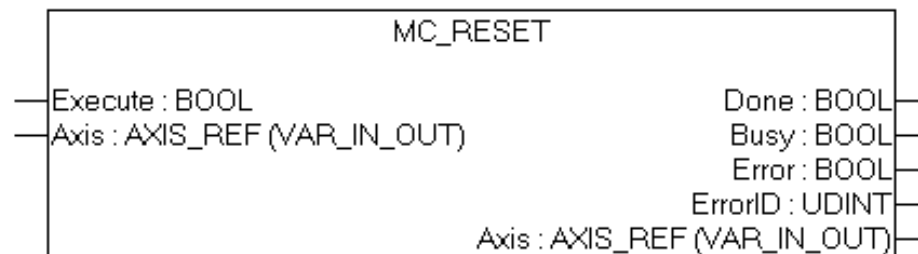


Click Block for More Information

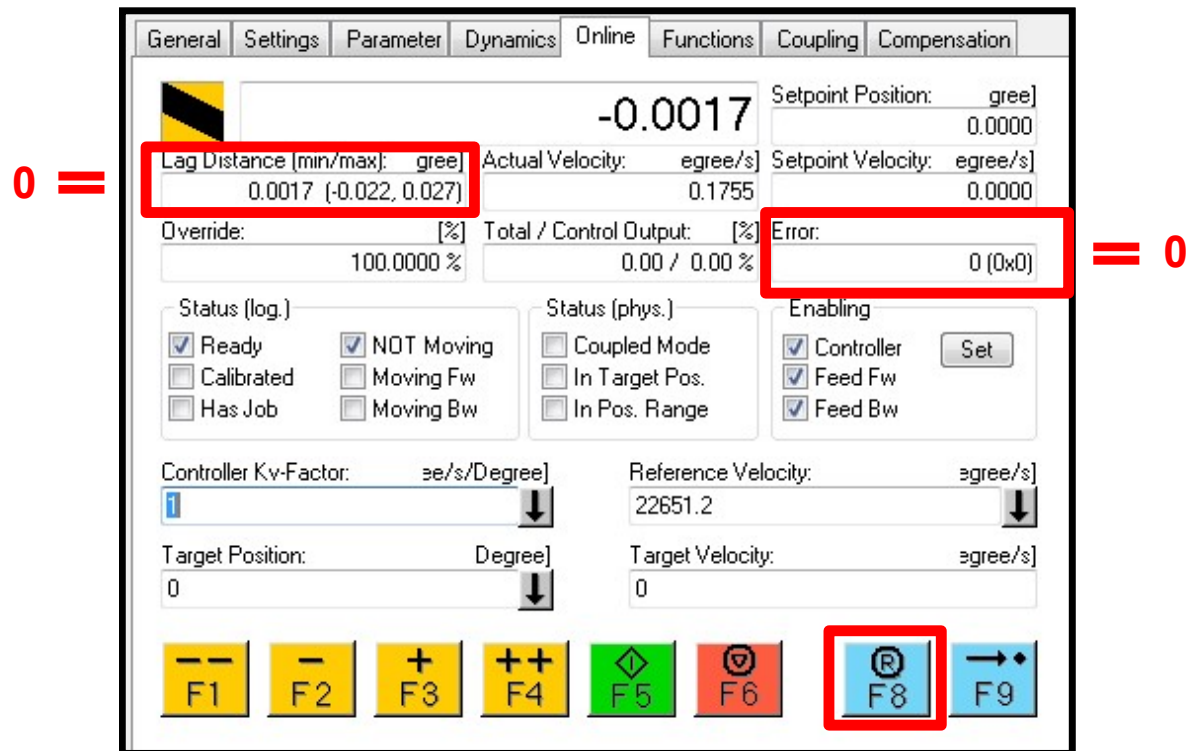
- MC_Power offers the same functionality as the **Enabling** interface of the **Online** tab



- Inputs:
 - **Execute** will:
 - Clear the current NC error for the specified axis
 - Clear any residual position error
 - Sets the current position as the target(final setpoint) position for the current move

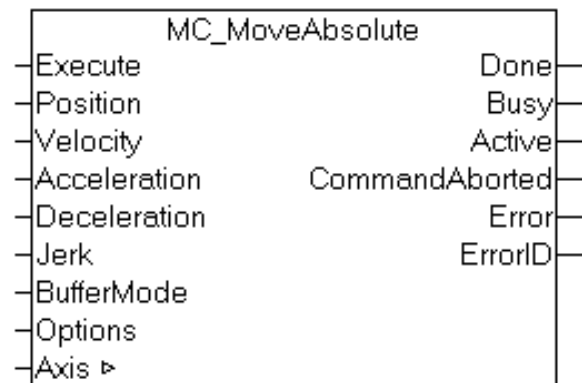


- MC_Reset offers the same functionality as the **Reset button(F8)** of the **online tab**



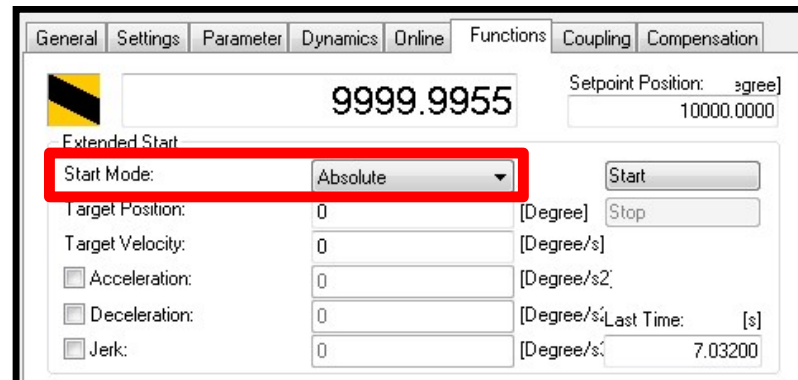
MC_MoveAbsolute moves Axis to New Position

- **Position:** Location relative to zero which axis moves to on **Execute**
- **Velocity:** Speed of travel to new position(must be > 0)
- Not specifying **Acceleration**, **Deceleration**, and **Jerk** will result in default parameters from the axis being used



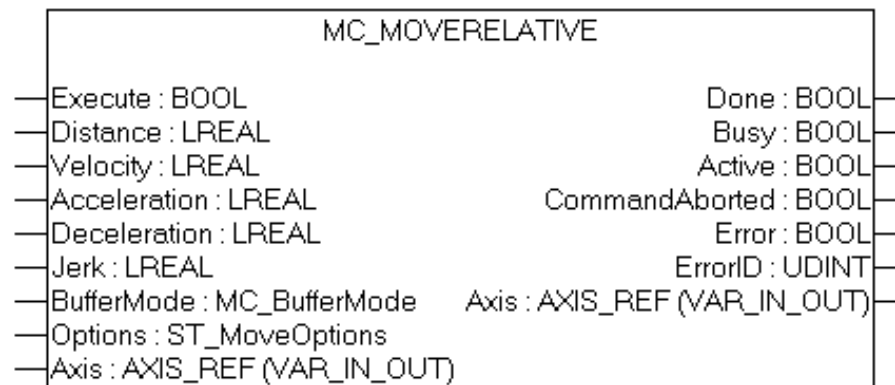
Click Block for More Information

- Functionality can also be found in the Functions tab under the Absolute Start Mode



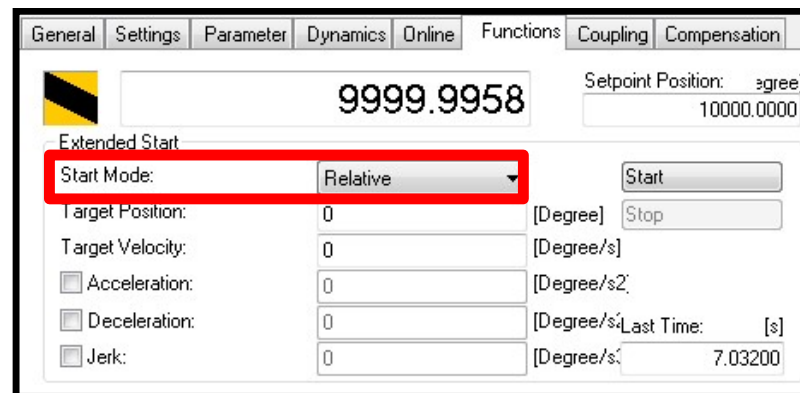
MC_MoveRelative moves Axis to New Position

- **Distance**: Location relative to current axis position which axis moves to on **Execute**
- **Velocity**: Speed of travel to new position(>0)
- Not specifying **Acceleration**, **Deceleration**, and **Jerk** will result in default parameters from the axis being used



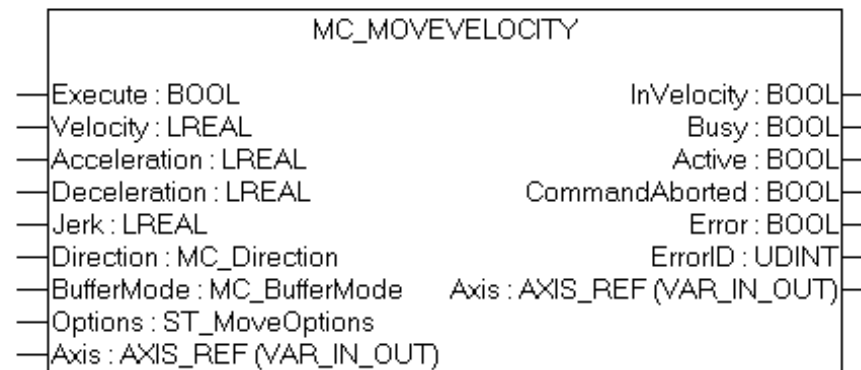
Click Block for More Information

- Operation can also be found in the Functions tab under the Relative Start Mode



MC_MoveVelocity moves Axis Continuously

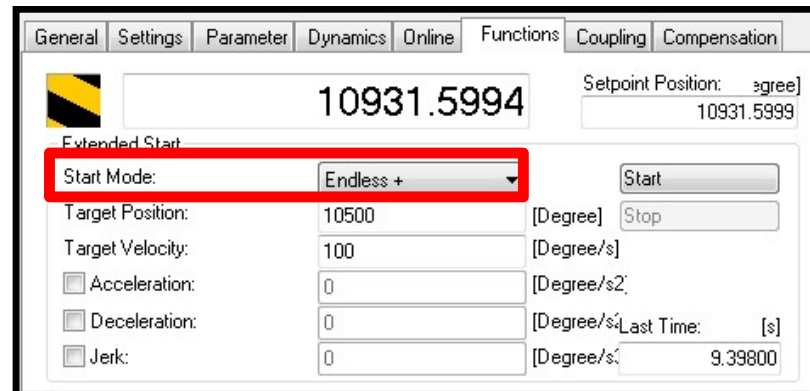
- **Velocity**: Speed of continuous travel
- **Direction** is an enumerated value used to choose forward or backward motion



Click Block for More Information

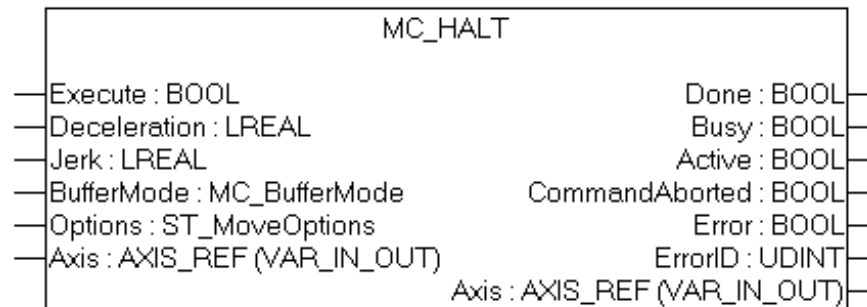
MC_MoveVelocity moves Axis Continuously

- Operation can also be found in the Functions tab under the Endless Start Mode

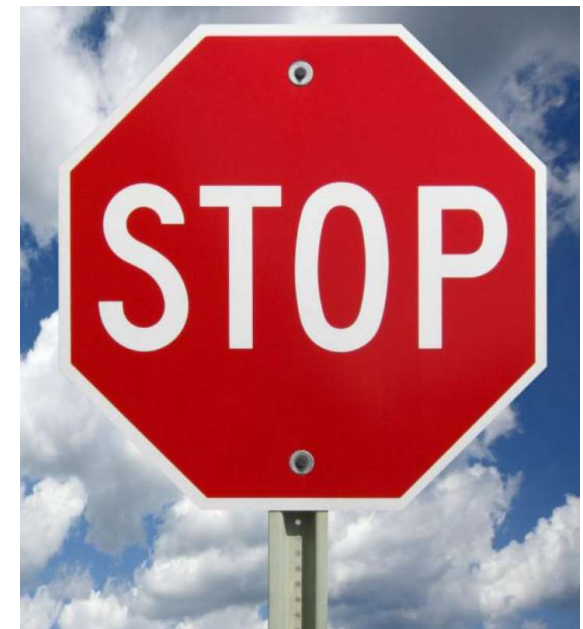


MC_Halt is an Interruptible Stop Command

- **Deceleration** allows the rate of velocity reduction to be set
 - Lower deceleration rates than the current move are ignored
 - MC_Halt will not cause a move to occur which passes the original setpoint
- **Options** are not available

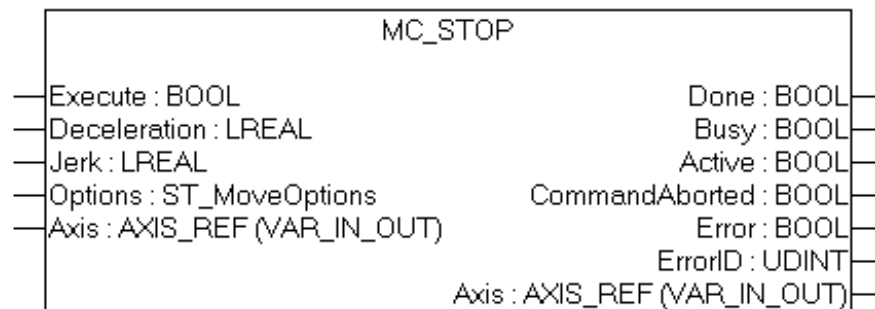


Click Block for More Information



MC_Stop is a Non-interruptible Stop Command

- **Execute** TRUE will prevent further motion commands until Execute is FALSE AND velocity is zero.
- **Deceleration** allows the rate of velocity reduction to be set
 - Lower deceleration rates than current move are ignored
- **Options** are not available

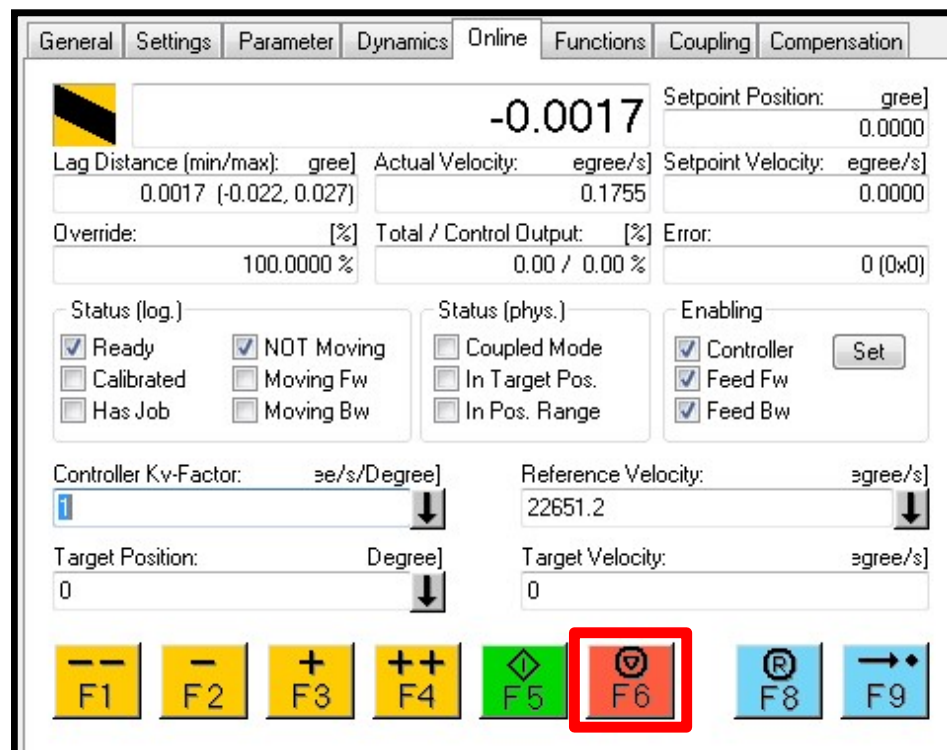


Click Block for More Information



MC_Stop is a Non-interruptible Stop Command

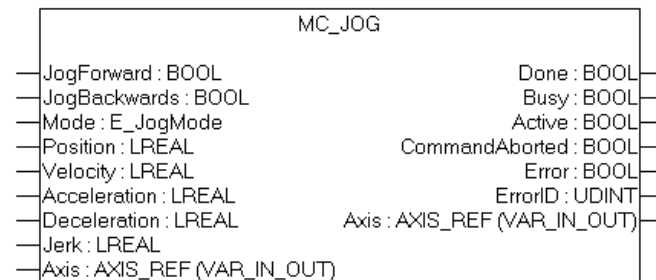
- Operation can also be found in the Online tab by pressing the Red Button(F6)



MC_Jog Allows for Manual Positioning Moves

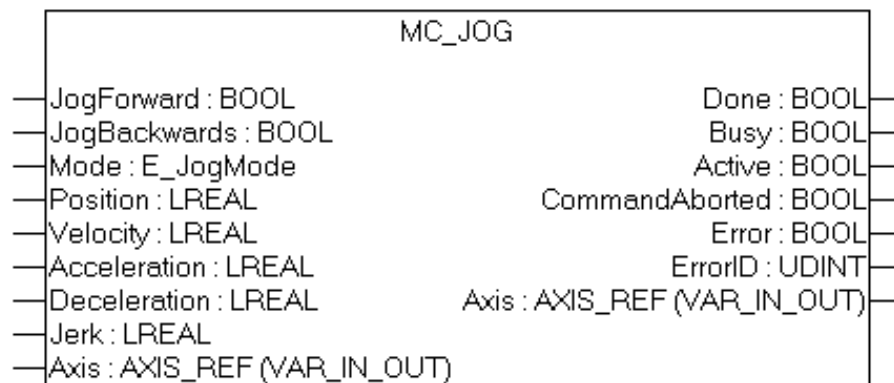
- **JogForward** or **JogBackward** TRUE will move the axis forward or backward until FALSE using the velocity as specified in the axis parameters

General Settings Parameter Dynamics Online Functions Coupling Compensation					
Parameter	Offline Value	Online Value	Type	Unit	
+ Maximum Dynamics:					
+ Default Dynamics:					
- Manual Motion and Homing:					
Homing Velocity (towards plc cam)	4.852933105...	4.852933105468...	F	mm/s	
Homing Velocity (off-plate cam)	4.852933105...	4.852933105468...	F	mm/s	
Manual Velocity (Fast)	145.5879931...	145.5879931640...		mm/s	
Manual Velocity (Slow)	24.26466552...	24.26466552734...		mm/s	
Jog Increment (Forward)	5.0	5.0		mm	
Jog Increment (Backward)	5.0	5.0		mm	
+ Fast Axis Stop:					



MC_Jog Allows for Manual Positioning Moves

- Using **Mode** will change block functionality.
- The following modes are available:
 - MC_JogMode_Standard_Slow(use slow parameter)
 - MC_JogMode_Standard_Fast(use fast parameter)
 - MC_JogMode_Continuous(use **Velocity** input)
 - MC_JogMode_Inching(execute a relative move)
 - MC_JogMode_InchingModulo(execute a relative move)



Click Block for More Information

MC_Jog Allows for Manual Positioning Moves

- Default block functionality can be observed using the fast and slow jog buttons from the online tab of the axis

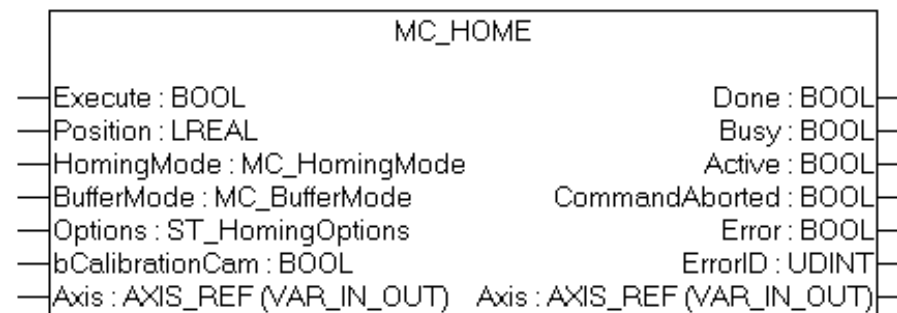
The screenshot displays the Beckhoff MC_Jog control interface, specifically the 'Online' tab. The interface is organized into several sections:

- General Information:** Shows a yellow and black diagonal flag icon, a position value of -0.0017, and a Lag Distance (min/max) of 0.0017 (-0.022, 0.027).
- Velocity and Error:** Displays Actual Velocity at 0.1755 egrade/s, Setpoint Velocity at 0.0000 egrade/s, and an Error of 0 (0x0).
- Override:** Shows an Override of 100.0000 %.
- Status (log.):** Includes checkboxes for Ready (checked), Calibrated, Has Job, NOT Moving (checked), Moving Fw, and Moving Bw.
- Status (phys.):** Includes checkboxes for Coupled Mode, In Target Pos., and In Pos. Range.
- Enabling:** Includes checkboxes for Controller (checked), Feed Fw, and Feed Bw, along with a 'Set' button.
- Controller Kv-Factor:** Set to 22651.2 ae/s/Degree.
- Reference Velocity:** Set to 22651.2 egrade/s.
- Target Position:** Set to 0 Degree.
- Target Velocity:** Set to 0 egrade/s.

At the bottom, there is a row of function buttons: F1 (yellow, minus), F2 (yellow, minus), F3 (yellow, plus), F4 (yellow, plus), F5 (green, diamond), F6 (red, circle with X), F8 (blue, circle with R), and F9 (blue, arrow). The F1, F2, F3, and F4 buttons are highlighted with a red rectangle.

MC_Home Changes Axis Position and Position Value

- **Execute** will cause axis to move until **bCalibrationCam** TRUE
- Axis will stop, and move in opposite direction until **bCalibrationCam** FALSE
- Axis Position value is then set equal to **Position**
- **HomingMode** allows for alternate homing sequences
- **Options** allow position lag to be cleared when homed

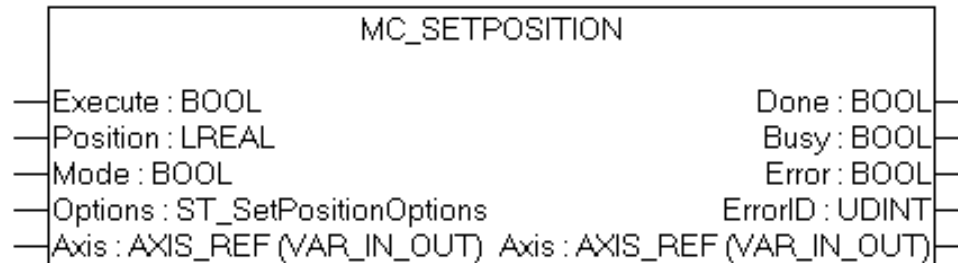


- Introduction
- Function Block Rules
- Type Axis_Ref
- **Common Blocks:**
 - Standard
 - **Informational**
- Buffered Moves

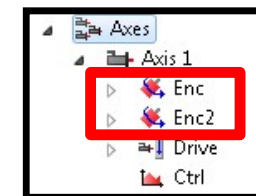
MC_Setposition changes the Axis Position Value

Inputs:

- **Execute** will change the position value of the axis
- **Mode** specifies if the new axis position is to be set equal to the **Position** input(absolute) or be adjusted by the **Position** input(relative)
- **Options** allow for clearing position lag OR for selecting which encoder is desired for position adjustment(if multiple encoders present)



Click Block for More Information

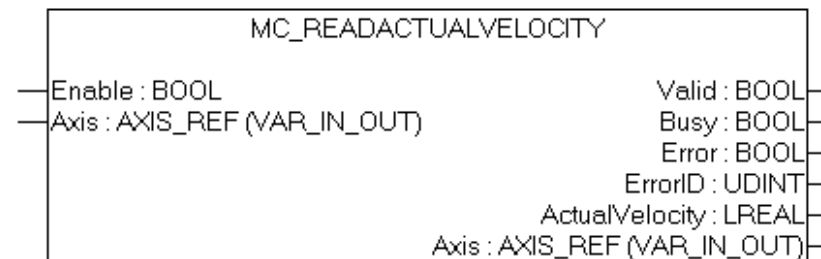
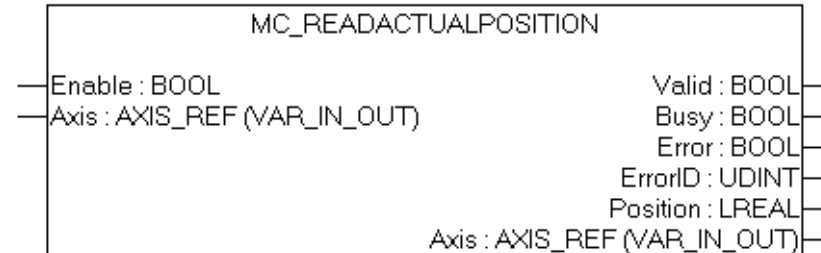


MC_Setposition changes the Axis Position Value

- MC_Setposition has the functionality of the **Set Actual Position** area of the **Functions Tab**

The screenshot displays the Beckhoff MC_Setposition control interface. At the top, there are tabs for General, Settings, Parameter, Dynamics, Online, Functions, Coupling, and Compensation. The 'Functions' tab is selected. Below the tabs, a large digital display shows the current position value '49.9717'. To the right of the display, the 'Setpoint Position' is set to '50.0000' in degrees. The interface is divided into several sections: 'Extended Start' with fields for Start Mode (Absolute), Target Position (0), Target Velocity (0), Acceleration (0), Deceleration (0), and Jerk (0); 'Raw Drive Output' with Output Mode (Percent) and Output Value (0); 'Set Actual Position' (highlighted with a red rectangle) with a dropdown set to 'Absolute' and a value of 0; and 'Set Target Position' with a dropdown set to 'Absolute' and a value of 0. Each section includes a 'Start' or 'Set' button. The 'Set Actual Position' section is the primary focus of the slide.

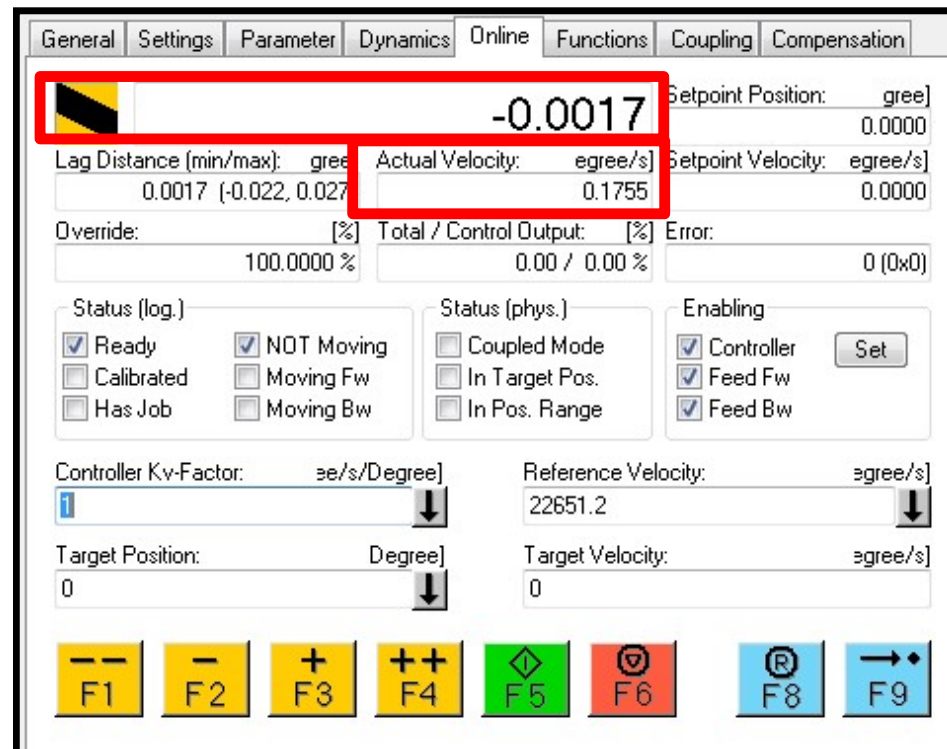
- Inputs:
 - Setting **Enable** TRUE will request the **Axis Position** or **Actual Velocity** to be returned from the NC
- Outputs:
 - Valid** TRUE if **Position** or **Actual Velocity** match current Axis value



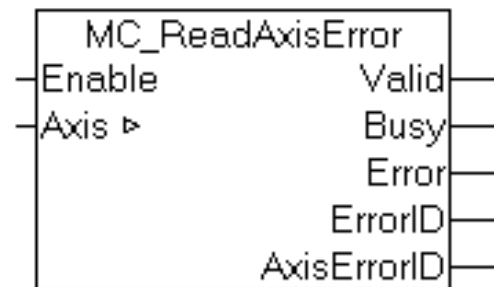
Click Block for More Information

MC_Read Blocks Return Axis Information

- MC_Read blocks will return the same information as Position and Actual Velocity in the online tab



- Inputs:
 - Enable** TRUE will cause current NC error to be acquired
 - Valid** will be TRUE if the value shown at **AxisErrorID** matches the current NC error of the axis



Click Block for More Information

- The block offers the same functionality as shown in the online tab of the axis

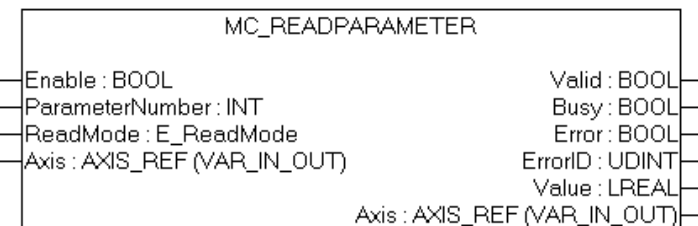
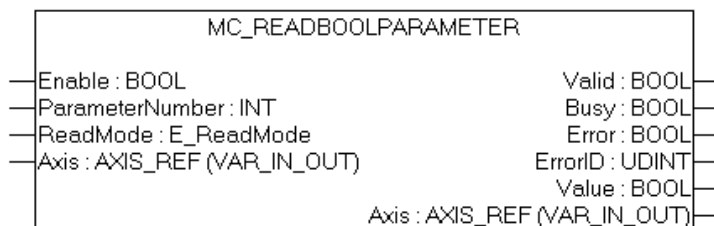
The screenshot displays the 'Online' tab of a Beckhoff axis control interface. The main display shows the current position as 3177.0305 degrees. Below this, the 'Error' field is highlighted with a red box, showing the value 16992 (0x4260). The interface includes several sections for status monitoring and control:

- Status (log.):** Includes checkboxes for Ready, Calibrated, Has Job, NOT Moving (checked), Moving Fw, and Moving Bw.
- Status (phys.):** Includes checkboxes for Coupled Mode, In Target Pos., and In Pos. Range.
- Enabling:** Includes checkboxes for Controller, Feed Fw, and Feed Bw, along with a 'Set' button.
- Control Parameters:** Includes fields for Controller Kv-Factor (set to 1), Reference Velocity (set to 22651.2), Target Position (set to 0), and Target Velocity (set to 0).
- Buttons:** A row of function buttons labeled F1 through F9, with F5 highlighted in green.

- Certain axis parameters are Floating(F) point numbers OR Booleans(B)
- This is shown in the Type column of the Parameter tab

General Settings Parameter Dynamics Online Functions Coupling Compensation					
	Parameter	Offline Value	Online Value	Type	Unit
-	Maximum Dynamics:				
	Maximum Velocity	20592.0	20592.0	F	Degree/s
	Maximum Acceleration	15000.0	15000.0	F	Degree/s2
	Maximum Deceleration	15000.0	15000.0	F	Degree/s2
-	Default Dynamics:				
	Default Acceleration	30888.0	30888.0	F	Degree/s2
	Default Deceleration	30888.0	30888.0	F	Degree/s2
	Default Jerk	92664.0	92664.0	F	Degree/s3
+	Manual Motion and Homing:				
+	Fast Axis Stop:				
-	Monitoring:				
	Position Lag Monitoring	TRUE		B	
	Maximum Position Lag Value	5.0		F	Degree
	Maximum Position Lag Filter Time	0.02		F	
	Position Range Monitoring	TRUE	TRUE	B	

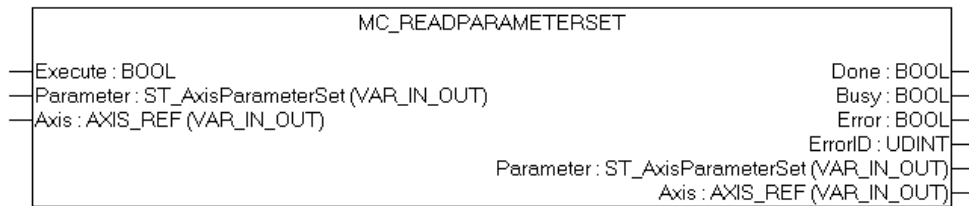
- Either of these parameters types can be read in with the MC_ReadBoolParameter OR MC_ReadParameter function blocks
- Inputs
 - **Enable** TRUE will cause a desired Axis Parameter to be read as chosen by the [ParameterNumber](#)
 - **ReadMode** specifies the frequency of the parameter update(each PLC cycle or once per rising edge of **Enable**)
- Output
 - **Valid** signifies the **Value** is current for the Axis parameter



Click Blocks for More Information

MC_ReadParameterSet Reads all Axis Parameters

- MC_ReadParameterSet reads a standard set of [parameters](#) (not identical to the parameter list of axis components) into one structure of data



Click Block for More Information



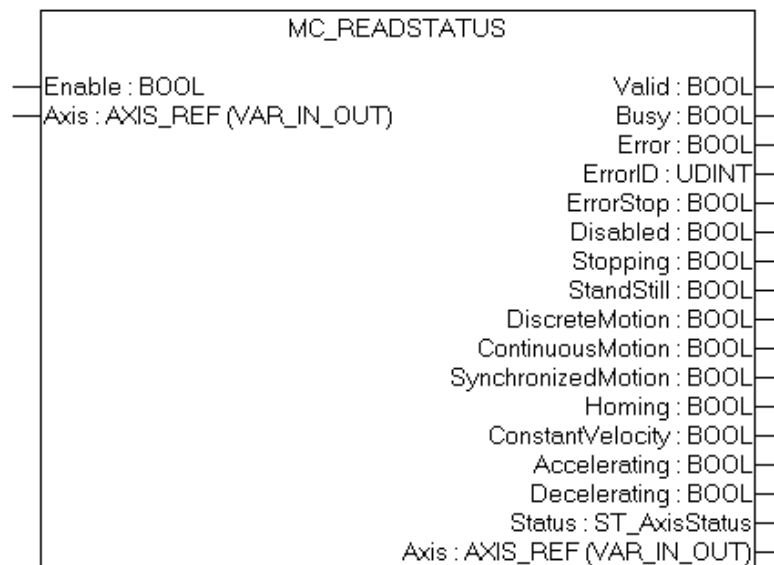
Parameter
+ Maximum Dynamics:
+ Default Dynamics:
+ Manual Motion and Homing:
+ Fast Axis Stop:
+ Monitoring:
+ Setpoint Generator:

Parameter
+ Output Settings:
+ Position and Velocity Scaling:
+ Torque and Acceleration Scaling:
+ Optional Position Command Output Smoothing Filter:
+ Other Settings:

Parameter
+ Monitoring:
+ Position Control Loop:
+ Other Settings:

Parameter
+ Encoder Evaluation:
+ Limit Switches:
+ Filter:
+ Homing:
+ Other Settings:

- When **Enable** TRUE, updates to axis information occur
- ST_AxisStatus holds expanded information regarding the axis



Click Block for More Information

MC_ReadStatus Looks at Status of Axis

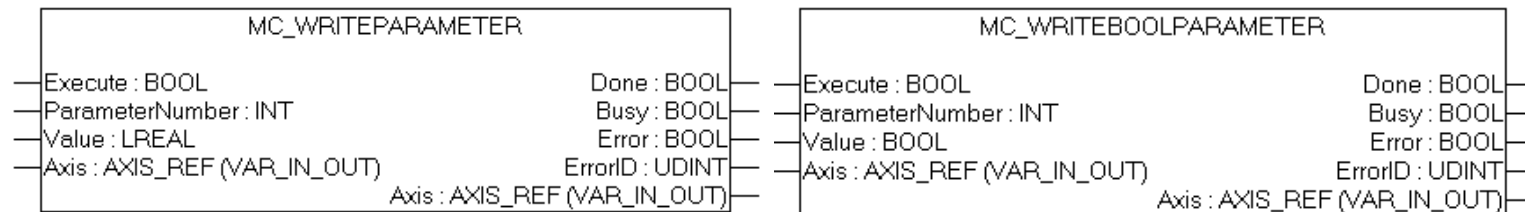
- Block makes information available as shown in the Online tab of each Axis (plus additional information)

The screenshot displays the 'Online' tab of the MC_ReadStatus block. The interface includes several sections:

- General:** Shows a yellow and black diagonal flag icon, a position value of -0.0017, and a Lag Distance (min/max) of 0.0017 (-0.022, 0.027).
- Actual Velocity:** 0.1755 egree/s.
- Setpoint Position:** 0.0000 gree.
- Setpoint Velocity:** 0.0000 egree/s.
- Override:** 100.0000 %.
- Total / Control Output:** 0.00 / 0.00 %.
- Error:** 0 (0x0).
- Status (log.):** A red box highlights this section, containing checkboxes for Ready (checked), Calibrated, Has Job, NOT Moving (checked), Moving Fw, and Moving Bw.
- Status (phys.):** Contains checkboxes for Coupled Mode, In Target Pos., and In Pos. Range.
- Enabling:** Contains checkboxes for Controller (checked), Feed Fw, and Feed Bw, along with a 'Set' button.
- Controller Kv-Factor:** 1 se/s/Degree.
- Reference Velocity:** 22651.2 egree/s.
- Target Position:** 0 Degree.
- Target Velocity:** 0 egree/s.
- Function Keys:** A row of buttons labeled F1 through F9, with F1-F4 in yellow, F5 in green, F6 in red, and F8-F9 in blue.

MC_Write Blocks Allow Parameter Changes

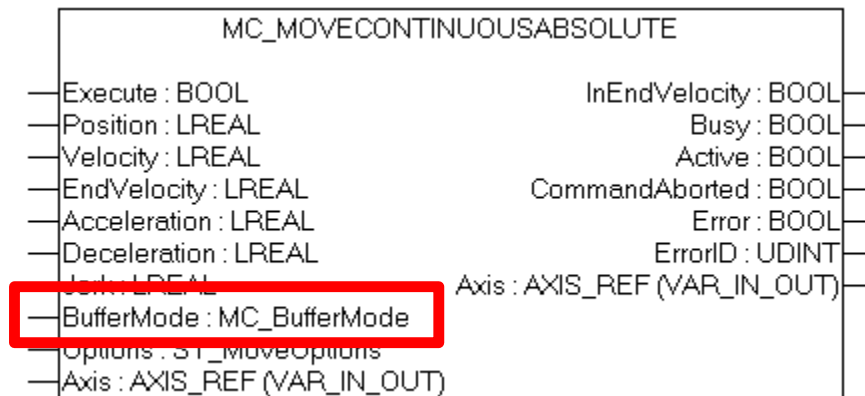
- Some parameters can be changed while the axis is operating
- Other parameters changes require that an axis be disabled
- MC_Write blocks allow a ParameterNumber to be selected and a **Value** specified to change Axis parameters
- Values written on **Execute**



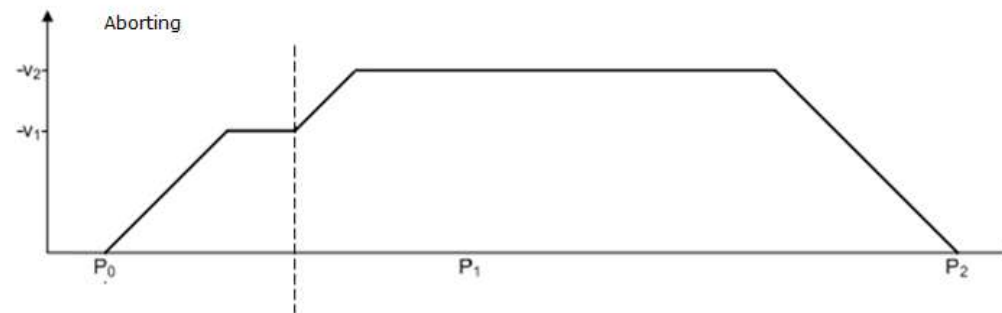
Click Block for More Information

- Introduction
- Function Block Rules
- Type Axis_Ref
- Common Blocks
- **Buffered Moves**

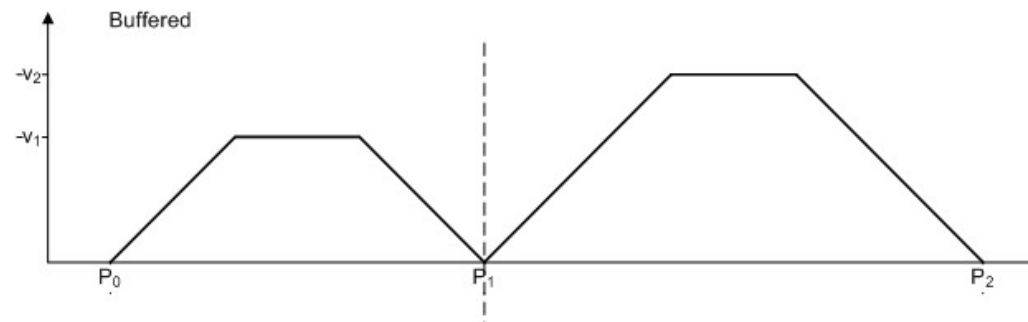
- **Buffering** calls a different block instance while one is currently active.
- Only one block can be buffered at a time.
- Many different buffering options are available through the [MC_BufferMode](#) enumeration
- Not all blocks can use BufferMode
 - Some blocks **do not have** it and others **have** it, but do not use it



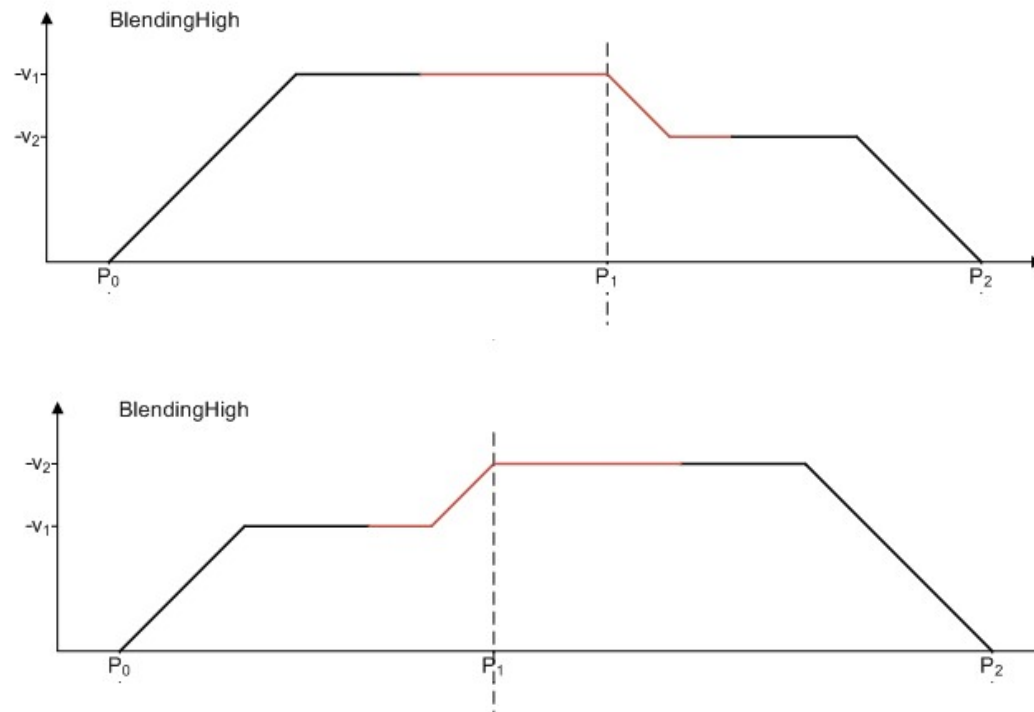
- ***Abort*** is the default BufferMode which immediately ends the first block and activates the second.



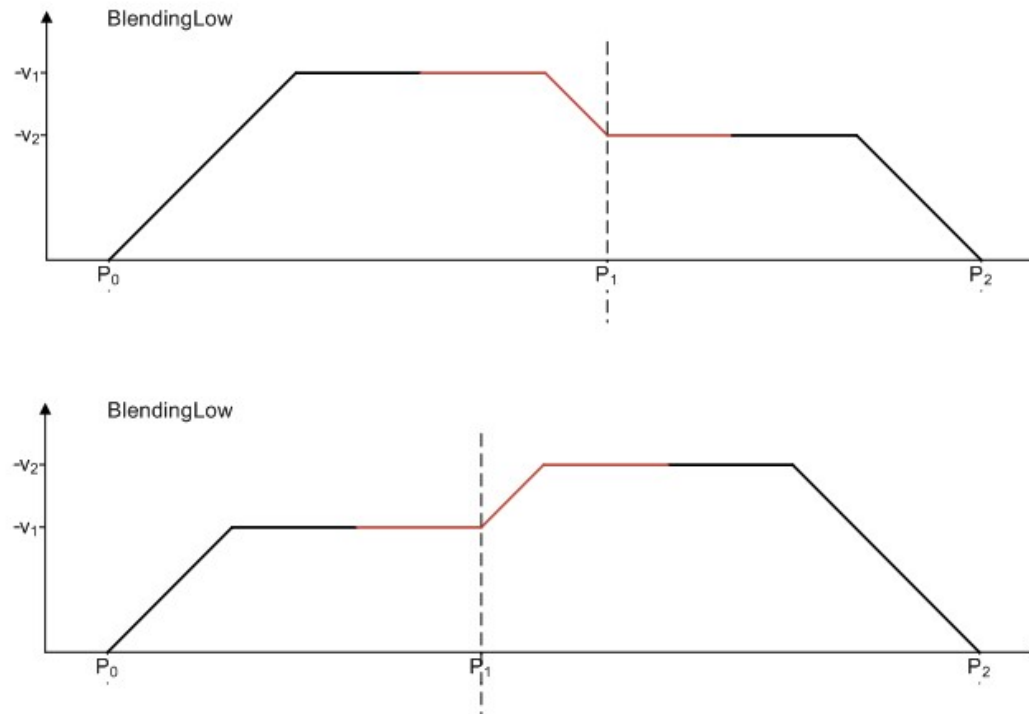
- ***Buffered*** is another BufferMode that brings axis to target before second block is activated



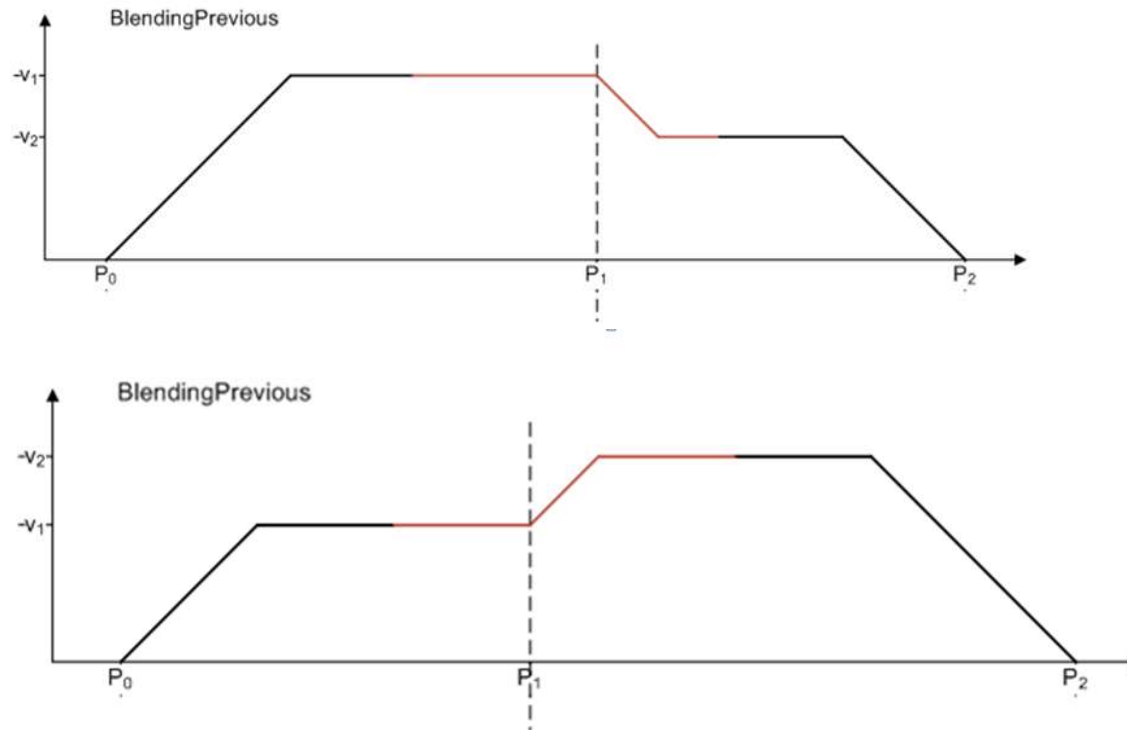
- **BlendingHigh** transitions moves so that the greater velocity of the two moves will be realized at the first target position



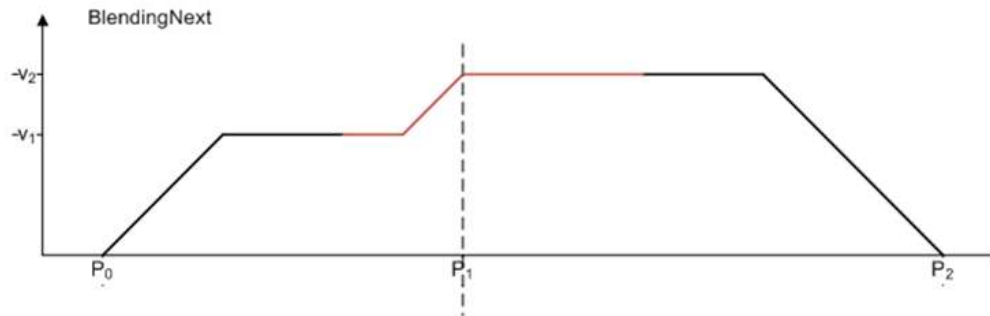
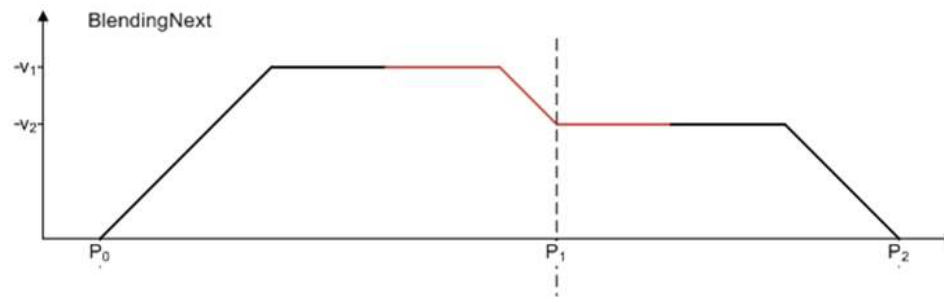
- **BlendingLow** transitions moves so that the lesser velocity of the two moves will be realized at the first target position



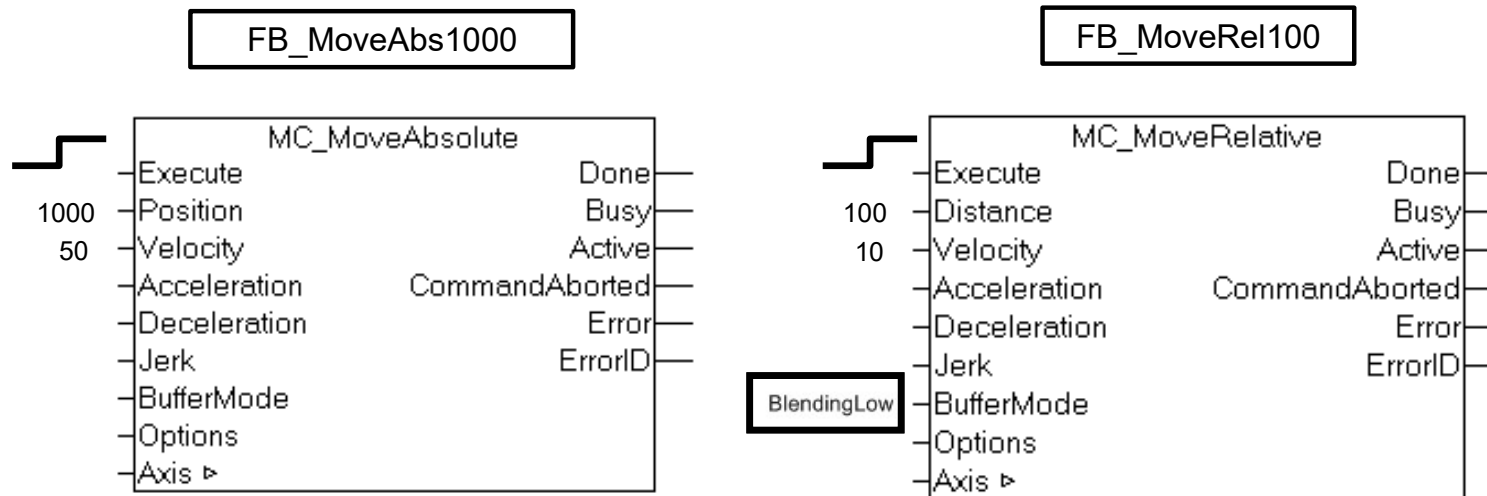
- **BlendingPrevious** transitions moves so that the velocity of the first move(previous move) will be realized at the first target position



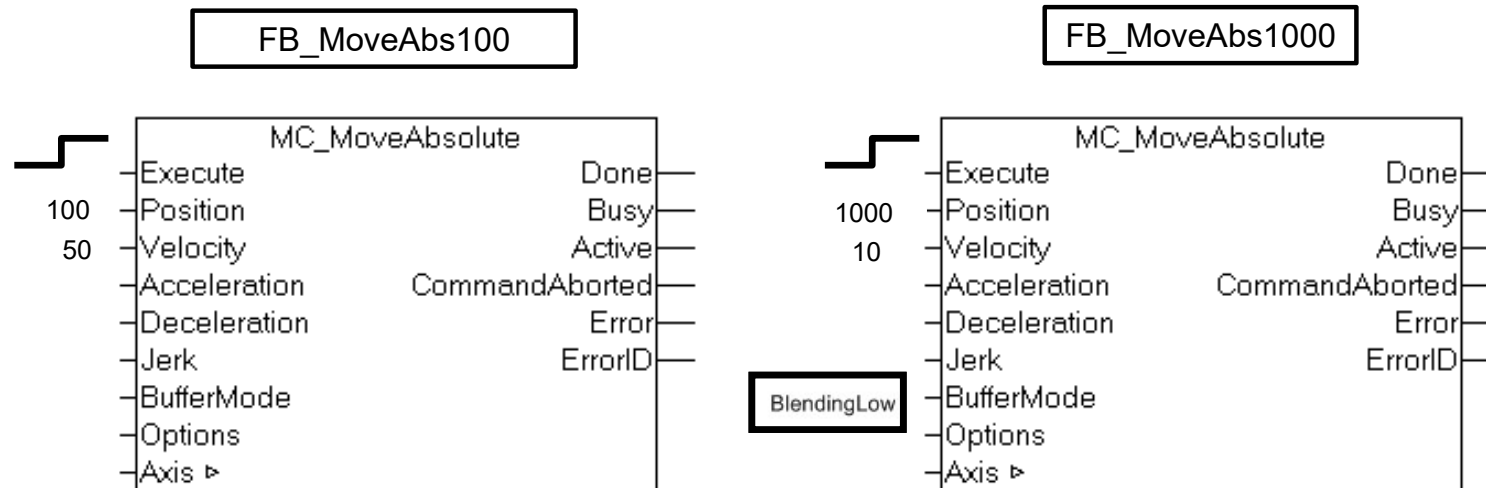
- **BlendingNext** transitions moves so that the velocity of the second move(next move) will be realized at the first target position



- **Buffering** is implemented on the second block instance.
 - The first block is the active block
 - The second block is the queued block



- If buffering the same block implementation, a ***new instance must be used***.
- Calling the same instance will have no effect until block completion, and no buffering is possible



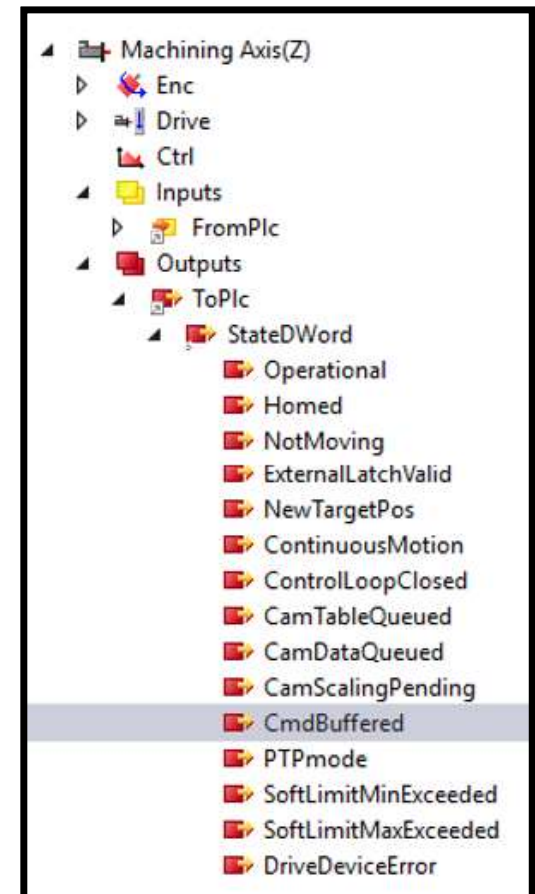
CmdBuffered Bit Shows Presence of Queued Move

BECKHOFF

- **CmdBuffered** bit:
 - TRUE when move waiting to execute
 - FALSE when no move pending
- If multiple buffered moves desired, wait until primary move complete, **CmdBuffered** false, to trigger new function block instance with BufferMode input
 - **CmdBuffered** will be true when new move loaded in que

```
GVL_IO.MachiningAxis.NcToPlc.StateDWord.24
```

```
GVL_IO.MachiningAxis();  
GVL_IO.MachiningAxis.Status.CmdBuffered
```



Beckhoff Automation LLC

USA Headquarters
13130 Dakota Avenue
Savage, MN 55378
USA

Phone: 1 952 890 0000
Fax: 1 952 890 2888
E-Mail: beckhoff.usa@beckhoff.com
Web: www.beckhoffautomation.com

Beckhoff Automation GmbH & Co. KG

Global Headquarters
Huelshorstweg 20
33415 Verl
Germany

Phone: +49 5246 963-0
Fax: +49 5246 963-198
E-Mail: info@beckhoff.com
Web: www.beckhoff.com

© Beckhoff Automation GmbH & Co. KG

All images are protected by copyright. The use and transfer to third parties is not permitted.

Beckhoff®, TwinCAT®, EtherCAT®, Safety over EtherCAT®, TwinSAFE®, XFC® and XTS® are registered trademarks of and licensed by Beckhoff Automation GmbH. Other designations used in this presentation may be trademarks whose use by third parties for their own purposes could violate the rights of the owners.

The information provided in this presentation contains merely general descriptions or characteristics of performance which in case of actual application do not always apply as described or which may change as a result of further development of the products. An obligation to provide the respective characteristics shall only exist if expressly agreed in the terms of contract.