Motion Axis Time Cam (MATC)

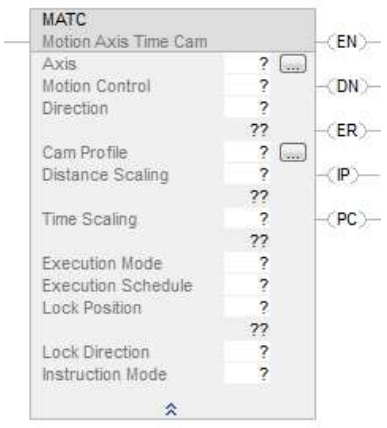
This information applies to the CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, and GuardLogix 5580 controllers. Controller differences are noted where applicable.

The Motion Axis Time Cam (MATC) instruction provides electronic camming of an axis as a function of time, according to the specified Cam Profile. Time cams allow the execution of complex motion profiles other than the built-in trapezoidal, or S-curve move profiles. When executed, the specified Axis is synchronized in time using a time Cam Profile established by the Logix Designer Cam Profile Editor, or by a previously executed Motion Calculated Cam Profile (MCCP) instruction. The direction of axis motion relative to the cam profile is defined by a very flexible Direction input parameter.

The camming Direction may be explicitly set as the Same or Opposite or set relative to the current camming direction as Reverse or Unchanged. The cam profile can be configured via the Execution Schedule parameter to execute Immediately or Pending completion of a currently executing time cam profile. The cam profile can also be executed Once or Continuously by specifying the desired Execution Mode. Distance and Time Scaling functionality can be used to scale axis motion based on a standard cam profile without having to create a new cam table and calculate a new cam profile.

Available Languages

Ladder Diagram



Function Block

This instruction is not available in function block.

Structured Text

MATC(Axis, MotionControl, Direction, CamProfile, DistanceScaling, TimeScaling, ExecutionMode, ExecutionSchedule, LockPosition, LockDirection, InstructionMode);

Operands

Ladder Diagram

Operand	Type **CompactLogix 5370, Compact GuardLogix 5370, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480**	Type **ControlLogix 5570, GuardLogix 5570, ControlLogix 5580, and GuardLogix 5580 controllers**	Format	Description
Axis	AXIS_CIP_DRIVE AXIS_GENERIC_DRIVE	AXIS_CIP_DRIVE AXIS_GENERIC_DRIVE AXIS_VIRTUAL AXIS_SERVO AXIS_SERVO_DRIVE	Tag	The name of the axis to which the cam profile is applied. Ellipsis launches Axis Properties dialog.
Motion Control	MOTION_INSTRUCTION	MOTION_INSTRUCTION	Tag	Structure used to access block status parameters.

Direction	UINT32	UINT32	Immediate or Tag	Relative direction of the slave axis to the master axis: 0 = Same – the axis position values in the cam profile are added to the command position of the axis. 1 = Opposite – the axis position values in the cam profile are subtracted from the command position of the axis creating axis motion in the other direction from that implied in the original cam table. Or relative to the current or previous camming direction: 2 = Reverse – the current or previous direction of the position cam is changed either from Same to Opposite or vice versa. When executed for the first time with Reverse selected, the control defaults the direction to Opposite. 3 = Unchanged – this allows other cam parameters to be changed without altering the current or previous camming direction. When executed for the first time with Unchanged selected, the control defaults the direction to Same.
Cam Profile	CAM_PROFILE	CAM_PROFILE	Array	Tag name of the calculated cam profile array. Only the zero array element ([0]) is allowed for the Cam Profile array. Ellipsis launches Cam Profile Editor.
Distance Scaling	REAL	REAL	Immediate or Tag	Scales the total distance covered by the axis through the cam profile.
Time Scaling	REAL	REAL	Immediate or Tag	Scales the time interval covered by the cam profile.
Execution Mode	UINT32	UINT32	Immediate	Determines how the cam motion behaves when the time moves beyond the end point of the cam profile. The options are: 0 = Once – When the time cam execution time exceeds the time range in the cam profile, the MATC instruction completes, the axis motion stops, and the Time Cam Status bit is cleared. 1 = Continuous – The cam profile motion is executed indefinitely.
Execution Schedule	UINT32	UINT32	Immediate	Selects the method used to execute the cam profile. Options are: Immediate – The slave axis is immediately locked to the master axis and the position camming process begins. Pending – Lets you blend a new position cam execution after an in process position cam is finished.

Lock Position	REAL	REAL	Immediate or Tag	Position on the Master Axis where a Slave should start following the master after the move has been initiated on the Slave Axis. See the Structure section below for more information.
Lock Direction	UINT32	UINT32	Immediate Real or Tag	Determines the direction of the Master Axis that generates Slave motion. See the Structure section below for more information.
Instruction Mode	UINT32	UINT32	Immediate	Specifies if a MATC should be executed in Time Driven Mode or Master Driven Mode See the Structure section for more information.

Structured Text

The operands are the same as those for the relay ladder MATC instruction.

Operand	**Type** **ControlLogix 5580 and GuardLogix 5580 controllers**	**Type** **CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380, and GuardLogix 5580 controllers**	**Type** **CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, and GuardLogix 5570 controllers**	**Format**	**Description**
Axis	AXIS_SERVO AXIS_SERVO_DRIVE AXIS_GENERIC_DRIVE	AXIS_CIP_DRIVE AXIS_GENERIC_DRIVE	AXIS_CIP_DRIVE AXIS_GENERIC_DRIVE AXIS_VIRTUAL AXIS_SERVO AXIS_SERVO_DRIVE	Tag	The name of the axis to which the cam profile is applied. Ellipsis launches Axis Properties dialog.
Motion Control		MOTION_INSTRUCTION	MOTION_INSTRUCTION	Tag	Structure used to access block status parameters.
Direction		UINT32	UINT32	Immediate or Tag	Relative direction of the slave axis to the master axis: 0 = Same – the axis position values in the cam profile are added to the command position of the axis. 1 = Opposite – the axis position values in the cam profile are subtracted from the

					command position of the axis creating axis motion in the other direction from that implied in the original cam table. Or relative to the current or previous camming direction: 2 = Reverse – the current or previous direction of the position cam is changed either from Same to Opposite or vice versa. When executed for the first time with Reverse selected, the control defaults the direction to Opposite. 3 = Unchanged – this allows other cam parameters to be changed without altering the current or previous camming direction. When executed for the first time with Unchanged selected, the control defaults the direction to Same.

Cam Profile		CAM_PROFILE	CAM_PROFILE	Array	Tag name of the calculated cam profile array. Only the zero array element ([0]) is allowed for the Cam Profile array. Ellipsis launches Cam Profile Editor.
Distance Scaling		REAL	REAL	Immediate or Tag	Scales the total distance covered by the axis through the cam profile.
Time Scaling		REAL	REAL	Immediate or Tag	Scales the time interval covered by the cam profile.
Execution Mode		UINT32	UINT32	Immediate	Determines how the cam motion behaves when the time moves beyond the end point of the cam profile. The options are: 0 = Once – When the time cam execution time exceeds the time range in the cam profile, the MATC instruction completes, the axis motion stops, and the Time Cam Status bit is cleared. 1 = Continuous – The cam profile motion is executed indefinitely.

Execution Schedule		UINT32	UINT32	Immediate	Selects the method used to execute the cam profile. Options are: Immediate – The slave axis is immediately locked to the master axis and the position camming process begins. Pending – Lets you blend a new position cam execution after an in process position cam is finished.
Lock Position		REAL	REAL	Immediate or Tag	Position on the Master Axis where a Slave should start following the master after the move has been initiated on the Slave Axis. See the Structure section below for more information.
Lock Direction		UINT32	UINT32	Immediate, Real, or Tag	Determines the direction of the Master Axis that generates Slave motion. See the Structure section below for more information.
Instruction Mode		UINT32	UINT32	Immediate	Specifies if a MATC should be executed in Time Driven Mode or Master Driven Mode

For the array operands, you do not have to include the array index. If you do not include the index, the instruction starts with the first element in the array ([0]).
See Structured Text Syntax for more information on the syntax of expressions within structured text.

For the operands that require you to select from available options, enter your selection as:

This Operand	Has These Options Which You	
	Enter as Text	Or Enter as a Number
ExecutionMode	once continuous	0 1
ExecutionSchedule	immediate pending	0 1
Lock Position	No enumeration	Immediate, Real, or Tag
Lock Direction	none immediateforwardnly immediatereverseonly positionforward positionreverse	0 1 2 3 4
Instruction Mode	timedrivenmode masterdrivenmode	0 1

MOTION_INSTRUCTION Structure

Mnemonic	Description
.EN (Enable) Bit 31	The enable bit is set when the rung transitions from false-to-true and stays set until the rung goes false.
.DN (Done) Bit 29	The done bit is set when the axis time cam instruction is successfully initiated.
.ER (Error) Bit 28	The error bit indicates when the instruction detects an error, such as if the axis is not configured.
.IP (In Process) Bit 26	The in process bit is set on positive rung transition and cleared when terminated by a stop command, merge, shutdown, or servo fault.
.PC (Process Complete) Bit 27	The Process Complete bit is cleared on positive rung transition and set in Once Execution Mode, when the time leaves the time range defined by the currently active cam profile.

Description

The MATC instruction executes a time cam profile set up by a previous Motion Calculated Cam Profile (MCCP) instruction or, alternatively, by the Logix Designer Cam Profile Editor. Time cams provide the capability of implementing complex motion profiles other than the built-in trapezoidal and S-curve motion profiles provided. No maximum velocity, acceleration, or deceleration limits are used in this instruction. The speed, acceleration, and deceleration of the slave axis are completely determined by the designated cam profile derived from the associated cam table.

Important: The maximum velocity, acceleration, or deceleration limits established during axis configuration do not apply to electronic camming.

Camming Direction

Cams can be configured to add or subtract their incremental contribution to the axis command position. Control over this behavior is via the Direction parameter.

Camming in the Same Direction

When Same is selected or entered as the Direction for the MATC instruction, the axis position values calculated from the cam profile are added to the command position of the axis. This is the most common operation, as the profile position values are used just as entered in the original cam table. That is, consecutive increasing profile values result in axis motion in the positive direction and vice-versa.

Camming in the Opposite Direction

When Opposite is selected or entered as the Direction, the axis position values calculated from the cam profile are subtracted from the command position of the axis. Thus, axis motion is in the opposite direction from that implied by the original cam table. That is, consecutive increasing profile values result in axis motion in the negative direction and vice-versa.

Changing the Cam Profile

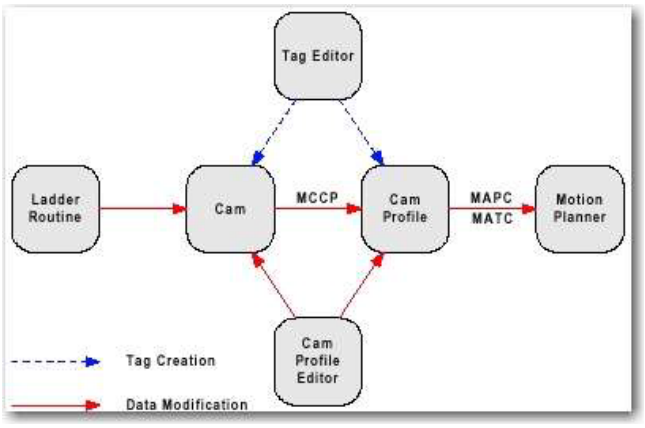
When Unchanged is selected or entered as the Direction, other time cam parameters may be changed while preserving the current or previous camming direction (same or opposite). This is useful when the current direction is not known or not important. For first time execution of a cam with Unchanged selected, the control defaults the direction to Same.

Changing the Camming Direction

When Reverse is selected the current or previous direction of the time cam is changed from Same to Opposite or from Opposite to Same. For first time execution of a cam with Reverse selected, the control defaults the direction to Opposite.

Specifying the Cam Profile

To execute a MATC instruction, a calculated Cam Profile data array tag must be specified. Cam Profile array tags may be created by the Logix Designer tag editor or the MATC instruction using the built-in Cam Profile Editor, or by executing an Motion Calculate Cam Profile (MCCP) instruction on an existing Cam array.



The data within the Cam Profile array can be modified at compile time using the Cam Profile Editor, or at run-time with the Motion Calculate Cam Profile (MCCP) instruction. In the case of run-time changes, a Cam array must be created in order to use the MCCP instruction.

All but the status and type elements of the Cam Profile array element structure are hidden from the Logix Designer tag editor. The status parameter is used to indicate that the Cam Profile array element has been calculated. If execution of a camming instruction is attempted with any uncalculated elements in a cam profile, the instruction errors. The type parameter determines the type of interpolation applied between this cam array element and the next cam element.

Cam Profile Array Checks

The Status member of the first element in the cam profile array is special and used for data integrity checks. For this reason, the MATC must always specify the cam profile with the starting index set to 0. This first cam profile element Status member can have the following values

Status Variables	**Description**
0	Cam profile element has not been calculated.
1	Cam profile element is being calculated.
2	Cam profile element has been calculated
n	Cam profile element has been calculated and is currently being used by (n-2) MAPC or MATC instructions.

Before starting a cam on a specified axis, the MATC instructions checks if the cam profile array has been calculated by checking the value of the first cam profile element's Status member. If Status is 0 or 1 then the cam profile has not been calculated yet and the MATC instruction errors. If the cam profile array has been completely calculated (Status > 1), the instruction then increments the Status member indicating that it is in use by this axis.

When the cam completes, or terminates, the Status member of the first cam profile array element is decremented to maintain track of the number of cams actively using the associated cam profile.

Linear and Cubic Interpolation

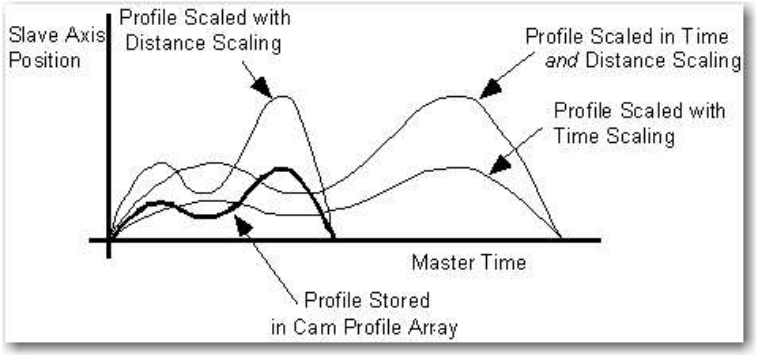
Time cams are fully interpolated. This means that if the current master time value does not correspond exactly with a point in the cam table associated with the cam profile, the slave axis position is determined by linear or cubic interpolation between the adjacent points. In this way, the smoothest possible slave motion is provided.

Each point in the Cam array that was used to generate the Cam Profile can be configured for linear or cubic interpolation.

Electronic camming remains active through any subsequent execution of jog, or move processes for the slave axis. This allows electronic camming motions to be superimposed with jog, or move profiles to create complex motion and synchronization.

Scaling Time Cams

A time cam profile can be scaled in both time and distance when it is executed. This scaling is useful to allow the stored profile to be used only for the form of the motion with the scaling used to define the time or distance over which the profile is executed, as shown below.



When a cam profile array is specified by an MATC instruction, the master coordinate values defined by the cam profile array take on the time units (seconds) and the slave values take on the units of the slave axis. By contrast, the Time and Distance Scaling parameters are unitless values that are simply used as multipliers to the cam profile.

By default, both the Time and Distance Scaling parameters are set to 1. To scale a time cam profile, enter a Time Scaling or Distance Scaling value other than 1.

Increasing the Time Scaling value of a cam profile decreases the velocities and accelerations of the profile, while increasing the Distance Scaling value increases the velocities and accelerations of the profile. To maintain the velocities and accelerations of the scaled profile approximately equal to those of the unscaled profile, the Time Scaling and Distance Scaling values should be equal. For example, if the Distance Scaling value of a profile is 2, the Time Scaling value should also be 2 to maintain approximately equal velocities and accelerations during execution of the scaled time cam.

Important: Decreasing the Time Scaling value or increasing the Distance Scaling of a time cam increases the required velocities and accelerations of the profile. This can cause a motion fault if the capabilities of the drive system are exceeded.

Cam Profile Execution Modes

Execution Modes of Once or Continuous can be selected to determine how the cam motion behaves when the time moves beyond the end point of the profile defined by the original cam table.

If Once is selected (default), the cam profile motion of the axis starts immediately. When the time cam execution time exceeds the time range defined by the cam profile, the MATC instruction completes, axis motion stops, and the Time Cam Status bit in the slave axis' Motion Status word is cleared.

When Continuous mode is selected, the specified cam profile, starts immediately and is executed indefinitely. With continuous operation, time is "unwound" to the beginning of the cam profile when it moves beyond the end of the cam profile, causing the cam profile to repeat indefinitely. This feature is particularly useful in rotary applications where it is necessary that the time cam run continuously in a rotary or reciprocating fashion. To generate smooth continuous motion using this technique, however, care must be taken in designing the cam points of the cam table to ensure that there are no position, velocity, or acceleration discontinuities between the start and end points of the calculated cam profile.

Execution Schedule

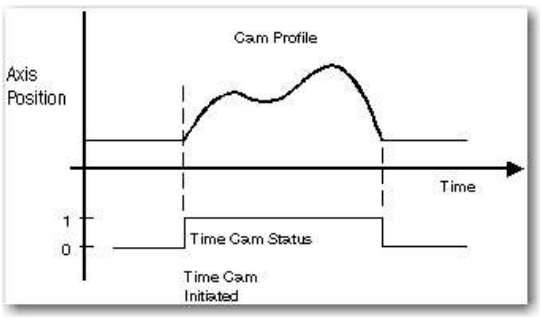
Control over the MATC instruction's execution schedule is via the Execution Schedule parameter.

Note that for the Master Driven Speed Control (MDSC) function, the Execution Schedule equal pending is used in both Master Driven Mode and Time Driven Mode.

Immediate Execution

By default, the MATC instruction is scheduled to execute immediately by virtue of the fact that the default setting of the Execution Schedule parameter is Immediate. In this case, there is no delay to the enabling of the time camming process.

As illustrated in the diagram below, when the MATC instruction is executed, the camming process is initiated on the specified axis and the Time Cam Status bit in the axis' Motion Status word is set. If the Execution Schedule parameter is set to Immediate, the axis is immediately locked to the time master coordinate according to the specified Cam Profile.

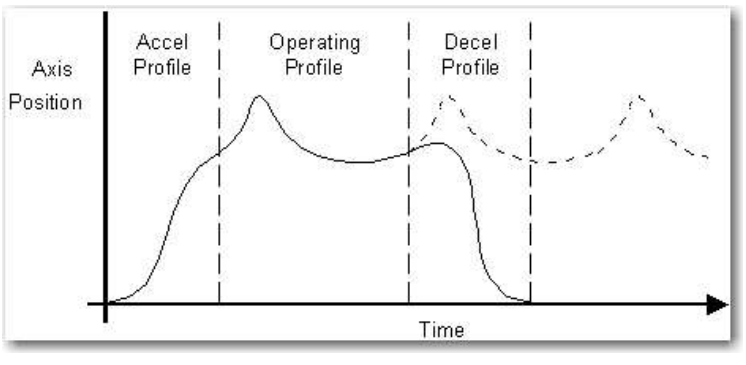


If an MATC instruction is executed on an axis that is already actively time camming, an Illegal Dynamic Change error is generated (error code 23). The only exception for this is if the Execution Schedule is specified as pending.

Pending Cam Execution

Alternatively, the MATC instruction's execution can, in effect, be deferred pending completion of a currently executing time cam. An Execution Schedule selection of Pending can thus be used to seamlessly blend two time cam profiles together without stopping motion.

The Pending execution feature is particularly useful in applications when the axis must be accelerated up to speed using a specific velocity profile. When this acceleration profile is done, it must be smoothly blended into a cam profile which is typically executed continuously. To stop the axis, the operating profile can be smoothly blended into a deceleration profile such that the axis stops at a known location as shown below.



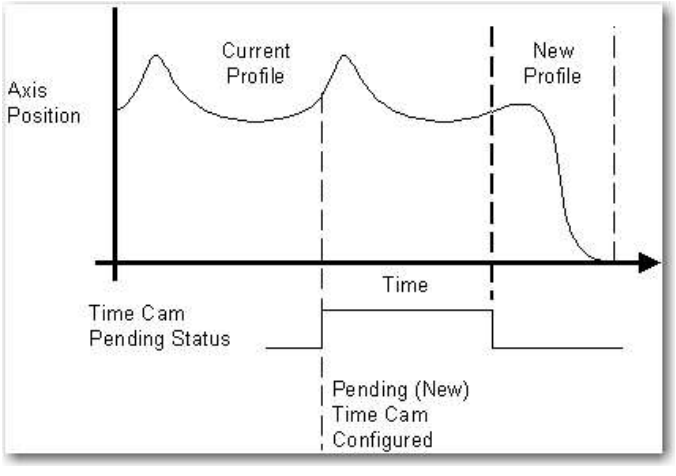
By executing the time cam profile as a Pending cam profile while the current profile is still executing, the appropriate cam profile parameters are set up ahead of time. This makes the transition from the current profile to the pending profile seamless – synchronization between the master time and slave axes position is maintained. To ensure smooth motion across the transition, however, the profiles must be designed such that no position, velocity, or acceleration discontinuities exist between the end of the current profile and the start of the new one. This is done using the Cam Profile Editor.

Once a pending time cam instruction has been executed, the new cam profile takes effect automatically (and becomes the current profile) when cam time passes through the end of the current profile. If the current cam is configured to execute once, the new profile is initiated at the completion of the pass through the current cam profile and the PC bit of the currently active MATC instruction is set. If the current cam is configured to execute continuously, the new profile is initiated at the completion of the current pass through the current cam profile and the IP bit of the currently active MATC instruction is cleared. The motion controller keeps track of time and the axis positions relative to the first profile at the time of the change and uses this information to maintain synchronization between the profiles.

If the Execution Schedule of an MATC instruction is set to Immediate and a time cam profile is currently in process, the MATC instruction generates an Illegal Dynamic Change error.

If an Execution Schedule of Pending is selected without a corresponding time cam profile in progress, the MATC instruction executes but no camming motion occurs until another MATC instruction with a non-pending Execution Schedule is initiated. This allows pending cam profiles to be preloaded prior to executing the initial cam. This method addresses cases where immediate cams would finish before the pending cam could be reliably loaded.

After a Pending time cam has been configured, the Time Cam Pending Status bit of the Motion Status word for the specified axis is set to 1 (true). When the pending (new) profile is initiated and becomes the current profile, Time Cam Pending Status bit is immediately cleared as shown below.



Master Driven Speed Control (MDSC) and Pending CAMS

For the Master Driven Speed Control (MDSC) function, a pending CAM is specified with an Execution Schedule of Pending.

Both a pending CAM and the active CAM must be in the same mode - either both in time driven mode or both in MDSC driven mode. An instruction that pends a CAM to an active CAM in a different mode (than the active CAM) will cause an error.

All pending CAMs must use the Immediate Forward or the Immediate Reverse Lock Direction. A pending CAM with a specified Lock Position with a Position Forward Only or Position Reverse Only Lock Direction will cause a LOCK_DIRECTION_CONFLICT (95) error. See MDSC Error Codes for a list of runtime error codes. A second pending CAM overlays the first pending CAM in memory.

Stopping a CAM

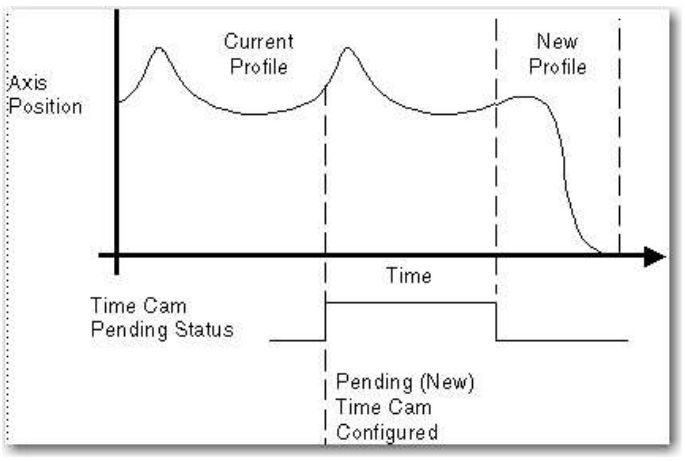
Like other motion generators (for example, jog, move, gear) active cams must be stopped by the various stop instructions, MAS, or MGS. CAM motion must also stop when the ControlLogix processor changes OS modes. The MAS instruction, in particular, must be able to specifically stop the camming process. This behavior should be identical to the MAS functionality that specifically stops a gearing process.

Merging from a CAM

Like other motion generators (for example, jog, move, gear) active CAMs must also be compliant with motion merge functionality. Moves and Jogs, in particular, must be able to merge from active camming. This behavior should be identical to the merge functionality applied to a gearing process.

Important: The MATC instruction execution completes in a single scan, thus the Done (.DN) bit and the In Process (.IP) bit are set immediately. The In Process (.IP) bit remains set until the initiated Time Camming process is superseded by another MATC instruction, or terminated by a Motion Axis Stop command, Merge operation, or Servo Fault Action.

In this transitional instruction, the relay ladder, toggle the Rung-condition-in from cleared to set each time the instruction should execute.



Structure

See Input and Output Parameters Structure for Single Axis Motion Instructions for the input and output parameters that are available for the MATC instruction via the Master Driven Speed Control (MDSC) function. Before any of these parameters is active, you must execute an MDAC instruction and it must be active (IP bit is set).

Affects Math Status Flags

No

Major/Minor Faults

None specific to this instruction. See Common Attributes for operand-related faults.

Execution

Ladder Diagram

Condition/State	Action Taken
Prescan	The .EN, .DN, .ER, and .IP bits are cleared to false.
Rung-condition-in is false	The .EN bit is cleared to false if either the .DN or .ER bit is true.
Rung-condition-in is true	The .EN bit is set to true and the instruction executes.
Postscan	N/A

Structured Text

Condition/State	Action Taken
Prescan	See Prescan in the Ladder Diagram table.
Normal execution	See Rung-condition-in is false, followed by rung is true in the Ladder Diagram table.
Postscan	See Postscan in the Ladder Diagram table.

Error Codes

See Motion Error Codes (.ERR) for Motion Instructions.

Extended Error Codes

Extended Error Codes provide additional instruction-specific information for the Error Codes that are generic to many instructions. Extended error codes for the PARAMETER_OUT_OF_RANGE(13) error code lists a number that refers to the number of the operands, as they are listed in the faceplate, from top to bottom, with the first operand being counted as zero. Therefore, for the MATC instruction, an extended error code of 5 would refer to the Time Scaling operand's value. You would then have to check your value against the accepted range of values for the instruction. See Motion Error Codes (.ERR) for Motion Instructions.

Changes to Status Bits

Motion Instruction Predefined Data Type Status Bits

See Status Bits for Motion Instructions (MAM, MATC, MAJ) When MDAC Is Active.

MATC Changes to Single Axis Status Bits

Bit Name	Meaning	
MotionStatus	**The motion status bit for your axis.**	
	Bit Number	**Meaning**
AccelStatus	0	The axis is not accelerating (FALSE state).
DecelStatus	1	The axis is not decelerating (FALSE state).
MoveStatus	2	The axis is not moving (FALSE state).
JogStatus	3	The axis is not jogging (FALSE state).
GearingStatus	4	The axis is not gearing (FALSE state).
HomingStatus	5	The axis is not homing (FALSE state).
StoppingStatus	6	The axis is stopping (TRUE state).
AxisHomedStatus	7	The axis is not homed (FALSE state).
PositionCamStatus	8	The axis is not position camming (FALSE state).
TimeCamStatus	9	The axis is not time camming (FALSE state).
PositionCamPendingStatus	10	The axis does not have a Position Cam Pending (FALSE state).
TimeCamPendingStatus	11	The axis does not have a Time Cam Pending (FALSE state).
GearingLockStatus	12	The axis is not in a Gear Locked condition (FALSE state).
PositionCamLockStatus	13	The axis is not in a Position Cam Locked condition (FALSE state).

TimeCamLockStatus	14	Time cam is Locked to Master in MDSC Mode (TRUE state). The bit is cleared when a MGS, MGSD, MAS, or MASD is executed (goes IP). If either the Slave or Master axis (or both) is paused by changing its speed to 0, then the TimeCamLockStatus bit stays set. Master Driven Mode The bit is set when the Lock Direction request is satisfied. The bit is not used when the enumeration is NONE. For the enumerations Immediate Forward Only and Immediate Reverse Only, the TimeCamLockStatus bit is set immediately when the MATC is initiated. For the enumeration Position Forward Only and Position Reverse Only, the bit is set when the Master Axis crosses the Master Lock Position in the specified direction. The TimeCamLockStatus bit is cleared when the Master Axis reverses direction and the Slave Axis stops following the Master Axis. The TimeCamLockStatus bit is set again when the Slave Axis resumes following the Master Axis. Time Driven Mode The bit is not used when the enumeration is NONE.
MasterOffsetMoveStatus	15	The axis is offset (TRUE state).
CoordinatedMotionStatus	16	Sets when the MDAC instruction executes (TRUE state). Clears when the instruction completes (FALSE state).
TransformStateStatus	17	The axis is part of an active transform (TRUE state).
ControlledByTransformStatus	18	The axis is moving because of a transform (TRUE state).
DirectVelocityControlStatus	19	The axis is not under Direct Velocity Control (FALSE state).
DirectTorqueControlStatus	20	The axis is not under Direct Torque Control (FALSE state).
JogLockStatus	24	The axis is not in a Jog Locked condition (FALSE state).
MasterOffsetMoveLockStatus	26	Master offset Move is Locked to master in MDSC Mode (TRUE state).
MaximumSpeedExceeded	27	Sets when the maximum axis speed that is specified in the axis configuration is exceeded during a move (TRUE state). Clears when the velocity is reduced below the limit (FALSE state).

Status Bits

If the Execution Schedule is set to Immediate, execution of the MATC instruction simply sets the Time Cam Status bit to True.

Bit Name	State	Meaning
TimeCamStatus	TRUE	Time Camming is Enabled.
TimeCamPendingStatus	FALSE	No pending Time Cam. Note that a pending Cam and the active Cam must be in the same mode - either both must be time driven or both must be MDSC driven. An instruction that pends a CAM to an active Cam in a different mode (than the active Cam) returns an error.

If the Execution Schedule is set to Pending, execution of the MATC instruction does not affect the current state of the Time Cam Status bits. Time Cam Pending Status bit is set to True immediately and transitions to False when the pending cam becomes the active cam.

Bit Name	State	Meaning
TimeCamStatus	N/A	Time Camming is Enabled.

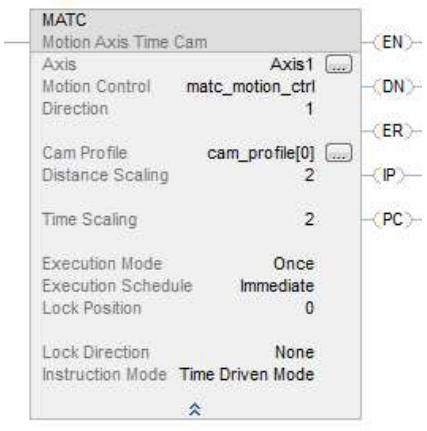
TimeCamPendingStatus	TRUE	Pending Time Cam. Note that a pending Cam and the active Cam must be in the same mode - either both must be time driven or both must be MDSC driven. An instruction that pends a CAM to an active Cam in a different mode (than the active Cam) returns an error.

Master Driven Speed Control (MDSC) and Motion Direct Command Support

The Motion Direct commands are not available in the instruction tree for the MATC instruction. You must program an MATC in one of the supported programming languages before you execute an MAM or MAJ in Time Driven Mode. A runtime error will occur if an MATC is not previously executed in an MAM and MAJ in Master Driven Mode.

Example

Ladder Diagram



Structured Text

```
MATC(Axis1,matc_motion_ctrl,1,cam_profile[0],2,2,Once,Immediate,0,None,TimeDrivenMode);
```

See also

Structured Text Syntax

MATC Flow Chart (True)

Motion Error Codes (.ERR)

Motion Move Instructions

Common Attributes

How are we doing?