Motion Coordinated Shutdown (MCSD)

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.

Use the Motion Coordinated Shutdown (MCSD) instruction to perform a controlled shutdown of all the axes in the named coordinate system.

Important: Tags used for the motion control attribute of instructions should only be used once. Re-use of the motion control tag in other instructions can cause unintended operation. This may result in damage to equipment or personal injury.

Important: Risk of Velocity and/or End Position Overshoot

If you change move parameters dynamically by any method, that is by changing move dynamics (MCD or MCCD) or by starting a new instruction before the last one has completed, be aware of the risk of velocity and/or end position overshoot.

A Trapezoidal velocity profile can overshoot if maximum deceleration is decreased while the move is decelerating or is close to the deceleration point.

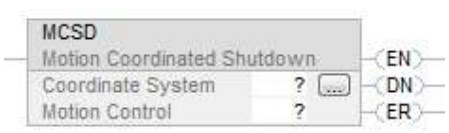
An S-curve velocity profile can overshoot if:

maximum deceleration is decreased while the move is decelerating or close to the deceleration point; or

maximum acceleration jerk is decreased and the axis is accelerating. Keep in mind, however, that jerk can be changed indirectly if it is specified in % of time.

Available Languages

Ladder Diagram



Function Block

This instruction is not available in function block.

Structured Text

MCSD(CoordinateSystem, MotionControl);

Operands

Ladder Diagram and Structured Text

Operand	Type	Format	Description
Coordinate System	COORDINATE_SYSTEM	Tag	Coordinated group of axes.
Motion Control	MOTION_INSTRUCTION	Tag	Structure used to access instruction status parameters.

Search

- Quick Start Steps
- Logix Designer
- Module Information
- Instruction Set
  - Logix 5000 Controllers Instruction and Application Considerations
  - Logix Designer Application Instruction Set
  - Interpret the Attribute Tables
  - Array Concepts
  - CIP Axis Attributes
  - Module Configuration Attributes
  - Bit Addressing
  - Common Attributes
  - Data Conversions
  - Elementary data types
  - LINT data types
  - Floating Point Values
  - Immediate values
  - Index Through Arrays
  - Math Status Flags
  - Motion Error Codes (.ERR)
  - Structures
  - Equipment Sequence instructions
  - Equipment Phase Instructions
  - Alarm Instructions
  - Advanced Math Instructions
  - Array (File)/Misc Instructions
  - Array (File)/Shift Instructions
  - ASCII Conversion Instructions
  - ASCII Serial Port Instructions
  - ASCII String Instructions
  - Bit Instructions
  - Compare Instructions
  - Debug Instructions
  - Drives Instructions
  - Drive Safety Instructions
  - For/Break Instructions
  - Filter Instructions
  - Function Block Attributes
  - Structured Text Attributes
  - Compute/Math Instructions
  - Move/Logical Instructions
  - Input/Output Instructions
  - License Instructions
  - Math Conversion Instructions
  - Metal Form Instructions
  - Motion Configuration

See Structured Text Syntax for more information on the syntax of expressions within structured text.

Coordinate System

The Coordinate System operand specifies the set of motion axes that define the dimensions of a Cartesian coordinate system. For this release the coordinate system supports up to three (3) primary axes. Only the axes configured as primary axes (up to 3) are included in the coordinate velocity calculations.

Motion Control

The following control bits are affected by the MCSD instruction.

Mnemonic	Description
.EN (Enable) Bit 31	The Enable bit sets when the rung transitions from false to true. It resets when the rung goes from true to false.
.DN (Done) Bit 29	The Done bit sets when the coordinated shutdown is successfully initiated. It resets when the rung transitions from false to true.
.ER (Error) Bit 28	The Error bit sets when the coordinated shutdown fails to initiate successfully. It resets when the rung transitions from false to true.

MCSD is a transitional instruction:

- In relay ladder, toggle the Rung-condition-in from cleared to set each time the instruction should execute.
- In structured text, condition the instruction so that it only executes on a transition. See Structured Text Syntax.

Master Driven Speed Control (MDSC) and the MCSD Instruction

When the coordinate system is shut down:

- The IP bit of the Master Driven Coordinate Control (MDCC) instruction is reset on an axis that is shutdown.
- The AC bit of the MDCC instruction resets when the axis is stopped as it is shutdown.
- The MCSD instruction clears the pending Master Axis for all future coordinate system motion instructions.

Affects Math Status Flags

No

Major/Minor Faults

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

Execution

Ladder Diagram

Condition/State	Action Taken
Prescan	The .EN, .DN, .ER, and .IP bits are cleared to false.

- Motion Configuration Instructions
- Motion Event Instructions
- Motion Group Instructions
- Motion Move Instructions
- Motion State Instructions
- Multi-Axis Coordinated Motion Instructions
  - Master Driven Coordinated Control (MDCC)
  - Motion Calculate Transform Position (MCTP)
  - Motion Coordinated Change Dynamics (MCCD)
  - Motion Coordinated Circular Move (MCCM)
  - Motion Coordinated Transform with Orientation (MCTO)
  - Motion Coordinated Path Move (MCPM)
  - Motion Calculate Transform Position with Orientation (MCTPO)
  - Motion Coordinated Linear Move (MCLM)
  - Motion Coordinated Shutdown (MCSD)
  - Motion Coordinated Shutdown Reset (MCSR)
  - Motion Coordinated Stop (MCS)
  - Motion Coordinated Transform (MCT)
  - Speed, acceleration, deceleration, and jerk enumerations for coordinated motion
  - Status Bits for Motion Instructions (MCLM, MCCM) when MDCC Is Active
  - Change between master driven and time driven modes for Coordinated Motion instructions
  - Choose a Termination Type
  - Common Action Table for Slave Coordinate System and Master Axis
  - Input and Output Parameters Structure for Coordinate System Motion Instructions
  - Returned Calculated Data Parameter for Coordinated System Motion Instruction

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.

MCSD Changes to Status Bits

Status bits provide a means for monitoring the progress of the motion instruction. There are three types of Status bits that provide pertinent information. They are: Axis Status bits, Coordinate System Status bits, and Coordinate Motion Status bits. When the MCS instruction initiates, the status bits undergo the following changes.

Axis Status Bits

Bit Name	**Meaning**
CoordinatedMoveStatus	Cleared

Coordinate System Status Bits

Bit Name	**Meaning**
ShutdownStatus	Sets when MCSD is executed and all associated axes are shutdown.
ReadyStatus	Cleared after MCSD executes.

Coordinated Motion Status Bits

Bit Name	**Meaning**
AccelStatus	Cleared after MCSD executes.
DecelStatus	Cleared after MCSD executes.
ActualPosToleranceStatus	Cleared after MCSD executes.
CommandPosToleranceStatus	Cleared after MCSD executes.
StoppingStatus	Cleared after MCSD executes.

- Logical and Move Instructions
- Program Control Instructions
- Sequencer Instructions
- Special Instructions
- Timer and Counter Instructions
- Trigonometric Instructions
- Process Control Instructions
- Select/Limit Instructions
- Sequential Function Chart (SFC) Instructions
- Statistical Instructions
- Safety Instructions

Studio 5000 Logix Designer Glossary

MoveStatus	Cleared after MCSD executes.
MoveTransitionStatus	Cleared after MCSD executes.
MovePendingStatus	Cleared after MCSD executes.
MovePendingQueueFullStatus	Cleared after MCSD executes.

Examples

Ladder Diagram

Structured Text

```
MCSD(myMcsdCoordinateSystem,myMcsdMotionControl);
```

See also

Common Action Table for Slave and Master Axis

Structured Text Syntax

Motion Error Codes (.ERR)

Multi-Axis Coordinated Motion Instructions

Common Attributes

How are we doing?