MITSUBISHI ELECTRIC

Programmable Controller

MELSEC iQ-R series

MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

-RD78G4
-RD78G8
-RD78G16
-RD78G32
-RD78G64
-RD78GHV
-RD78GHW

SAFETY PRECAUTIONS

(Read these precautions before using this product.)

Before using MELSEC iQ-R series programmable controllers, please read the manuals for the product and the relevant manuals introduced in those manuals carefully, and pay full attention to safety to handle the product correctly.

Make sure that the end users read this manual and then keep the manual in a safe place for future reference.

INTRODUCTION

Thank you for purchasing the Mitsubishi Electric MELSEC iQ-R series programmable controllers.

This manual describes the Motion control function blocks, variables, and creation methods required for programming with the Motion module.

Before using this product, please read this manual and the relevant manuals carefully and develop familiarity with the functions and performance of the MELSEC iQ-R series programmable controller to handle the product correctly.

When applying the program examples provided in this manual to an actual system, ensure the applicability and confirm that it will not cause system control problems.

Please make sure that the end users read this manual.

Relevant products

RD78G4, RD78G8, RD78G16, RD78G32, RD78G64, RD78GHV, RD78GHW

Point

Symbols used in this manual are shown below.

- Underlined variables (AxisName, etc.): Variables defined by users

CONTENTS

SAFETY PRECAUTIONS 1
INTRODUCTION 1
RELEVANT MANUALS 5
TERMS 6
GENERIC TERMS AND ABBREVIATIONS 8

CHAPTER 1 MOTION CONTROL FB OVERVIEW 9

1.1 Motion Control FBs 9
How to use Motion control FBs 9
Operation of this function for each system status 9
Motion control FB types 10
Motion control types 14
Error processing 14
Units used in control 15
I/O variables used in Motion control FBs 16
Precautions 18
1.2 Motion Control FB Configuration 19
Ladder 19
FBD/LD (Function block diagram/ladder) 20
ST (Structured text) 20

CHAPTER 2 VARIABLES AND MOTION CONTROL FB 21

2.1 List of Variables 21
Axis variables 23
Axes group variables 46
System variables 53
Other structure variables 63
2.2 List of Enumerators 75
ENUM enumerators 75
2.3 List of Motion Control FBs 82
Management FBs 82
Operation FBs 83
Standard FBs 84

CHAPTER 3 MOTION CONTROL FB 85

3.1 Management FBs 85
MC_GroupEnable 85
MC_GroupDisable 88
MC_Power 91
MC_SetPosition 95
MCv_SetTorqueLimit 102
MC_SetOverride 110
MC_ReadParameter 114
MC_WriteParameter 120
MC_Reset 126
MC_GroupReset 129
MC_TouchProbe 133
MC_AbortTrigger 147

MC_CamTableSelect . . . 153
MCv_ChangeCycle . . . 160
MCv_AllPower . . . 170
MC_GroupSetOverride . . . 174
MCv_MotionErrorReset . . . 178
3.2 Operation FBs . . . 182
MC_Home . . . 182
MC_Stop . . . 190
MC_GroupStop . . . 195
MC_MoveAbsolute . . . 200
MC_MoveRelative . . . 212
MCv_Jog . . . 220
MC_MoveVelocity . . . 229
MC_TorqueControl . . . 236
MCv_SpeedControl . . . 246
MCv_MoveLinearInterpolateAbsolute . . . 254
MCv_MoveLinearInterpolateRelative . . . 270
MCv_MoveCircularInterpolateAbsolute . . . 285
MCv_MoveCircularInterpolateRelative . . . 302
MC_CamIn . . . 321
MC_GearIn . . . 347
MC_CombineAxes . . . 357
MCv_BacklashCompensationFilter . . . 369
MCv_SmoothingFilter . . . 382
MCv_DirectionFilter . . . 391
MCv_SpeedLimitFilter . . . 399
3.3 Standard FBs . . . 407
MCv_ReadProfileData . . . 407
MCv_WriteProfileData . . . 419

CHAPTER 4 MOTION MODULE PROGRAMS 430

4.1 Motion Module Programs . . . 430
Execution of programs . . . 431
Label initialization function . . . 437
Label memory . . . 437
4.2 Motion Module Program Creation . . . 438
Program creation procedure for ST programs . . . 438
Creation procedure for programs using the public label function . . . 462
Registering the Motion module FB library . . . 472
Data transfers using the Motion module buffer memory . . . 477

APPENDICES 478

Appendix 1 FBD/LD Sample Program . . . 478
Settings in GX Works3 . . . 478
Settings in Motion Control Setting Function . . . 493
Appendix 2 ST Sample Program . . . 496
Settings in GX Works3 . . . 496
Settings in Motion Control Setting Function . . . 500
REVISIONS . . . 514
WARRANTY . . . 515

TRADEMARKS 516

RELEVANT MANUALS

Manual name [manual number]	Description	Available form
MELSEC iQ-R Programming Manual (Motion Control Function Blocks) [IB-0300533ENG] (This manual)	Motion control function blocks, variables, and programming	Print book e-Manual PDF
MELSEC iQ-R Motion Module User's Manual (Startup) [IB-0300406ENG]	Specifications, procedures before operation, system configuration, and wiring of the Motion module	Print book e-Manual PDF
MELSEC iQ-R Motion Module User's Manual (Application) [IB-0300411ENG]	Functions, I/O signals, variables, labels, programming, and troubleshooting of the Motion module	Print book e-Manual PDF
MELSEC iQ-R Motion Module User's Manual (Network) [IB-0300426ENG]	Functions, parameter settings, troubleshooting, and buffer memory of CC-Link IE TSN	Print book e-Manual PDF
MELSEC iQ-R Programming Manual (Motion Module Instructions, Standard Functions/Function Blocks) [IB-0300431ENG]	Instructions for the Motion module and standard functions/ function blocks	Print book e-Manual PDF
Motion Module Quick Start Guide [L03191ENG]	Describes system startup, parameter settings, and programming methods for first-time users of the Motion module	e-Manual PDF
Motion Module Quick Start Guide (PLC CPU Ladder Program) [L03194ENG]	Describes system startup, parameter settings, and programming methods for first-time users of the Motion module	e-Manual PDF
MELSEC iQ-R Programming Manual (Program Design) [SH-081265ENG]	Program specifications (ladder, ST, FBD/LD, and SFC programs)	e-Manual PDF
GX Works3 Operating Manual [SH-081215ENG]	System configuration, parameter settings, and online operations of GX Works3	e-Manual PDF

For programs, refer to the following.

MELSEC iQ-R Programming Manual (Program Design)

Point

e-Manual refers to the Mitsubishi Electric FA electronic book manuals that can be browsed using a dedicated tool.

e-Manual has the following features:

- Required information can be cross-searched in multiple manuals.
- Other manuals can be accessed from the links in the manual.
- The hardware specifications of each part can be found from the product figures.
- Pages that users often browse can be bookmarked.
- Sample programs can be copied to an engineering tool.

TERMS

Unless otherwise specified, this manual uses the following terms.

Term	Description
Aborting, Buffered, Blending	Various modes of the buffer mode
Actual position	An address that is rounded a position address returned by a slave device. It is affected by a current position change.
Actual velocity	A value obtained by converting the actual value from the slave device on the real axis into the technical units of the axis
Administrative FB	A motion control FB that takes an axis or an axes group for the argument and does not change the axis status or the axes group status by execution
Axes group variable	An AXES_GROUP type variable instance including parameters and data related to the axes group
Axis variable	An AXIS_*type variable instance including parameters and data related to the axis
Axis	A target to carry out the motion control
Buffering	That a motion control FB enters the standby status in multiple start
Buffering FB	A motion control FB that is carried out multiple start and waits for execution (Busy is TRUE)
Buffer memory	A memory in an intelligent function module, where data (such as setting values and monitoring values) are stored.
Buffer mode	A generic term for multiple start
Commanded position	A command position address that is specified in the motion FB
Commanded velocity	Set velocity that is specified in the motion FB
Continuous update	Continuously reflecting input values to the control while the ContinuousUpdate input of the motion control FB is TRUE
Cumulative current position	A command position address that is not rounded by a ring counter. It is affected by a current position change.
Cyclic transmission	A function by which data are periodically exchanged among stations on the same network
Dedicated instruction	An instruction for using functions of the module
Device	Various memory data in a module. There are devices handled in each bit and in each word.
Disconnection	A process of stopping data link if a data link error occurs
Execute	Setting TRUE to the Execute/Enable input for the motion control FB.
External signal high-accuracy input	An input signal linked with signal detection time of the slave station
General FB	A motion control FB that does not take an axis or an axes group for the argument
Global label	A label that is enabled for all program data when creating multiple program data in the project. The global label in the motion system can be made public as the module label (global label) in the control CPU module.
GX Works3	The product name of the software package for the MELSEC programmable controllers
Hardware reboot	Turning ON the power of the system again or resetting
Input variable	An input argument of FB
Intelligent function module	A module that has functions other than input and output, such as an A/D converter module and D/A converter module
Jerk	Chronological change ratio of the acceleration or the deceleration
Label	A variable used for a program
Link device	A device in a module on CC-Link IE
Link refresh	Automatic data transfer between a link device of the Motion module and a device in a CPU module
Machine feed value	A commanded position address that is not rounded by the ring counter. It is based on the home position, and it is not affected by a current position change.
Module label	A label that represents one of memory areas (I/O signals and buffer memory areas) specific to each module in a given character string. GX Works3 automatically generates this label, which can be used as a global label in the CPU module.
Motion area	Software that performs the motion control in the Motion module
Motion control station	A slave station to exchange cyclic data by a slave label and motion control
Motion control FB	An FB that is related to the motion control. Its name starts with MC_ or MCv_.
Motion FB	A motion control FB that takes an axis or an axes group for the argument and changes the axis status or the axes status by execution
Motion module	Another term for the RD78G(H)
Motion synchronization station	Another term for motion control station
MR-J5-G	MR-J5-_G_(-RJ) servo amplifier
MR-J5D-G	MR-J5D_-_G_ servo amplifier
MR-J5W-G	MR-J5W_-_G servo amplifier
MR-JET-G	MR-JET-_G servo amplifier
MR Configurator2	A product name of servo setup software
Multiple axes interpolation control	Control in which multiple axes work such as linear interpolation and circular interpolation. Axes to be cooperated by an axes group are specified.

Term	Description
Multiple start	Executing another motion control FB to the same axis while a motion control FB is executed
Network area	Software that performs the network control in the Motion module
Next FB	Another motion control FB that starts while an axis and an axes group are operated
Object	Various data of a slave device compatible with CANopen
On-going FB	Motion control FB that is in execution on the axis and the axes group (FB that Busy is TRUE)
Output variable	An output argument of FB
Previous FB	The previous motion control FB of a buffering FB (If only one FB is performed buffering, it indicates on-going FB)
RD78G	Another term for the MELSEC iQ-R series Motion module (compatible with CC-Link IE TSN)
RD78GH	
Real axis	An axis that is linked with a slave device on network
Real drive axis	An axis that is linked with a slave device which supports CC-Link IE TSN compatible with csp/csv/cst mode (sequential command) of the CiA402 drive profile
Real encoder axis	An axis that generates a commanded position from the current position of the encoder connected with a slave device. It is used for the master axis of the single axis synchronous control.
Reconnection	A process of restarting data link when a station recovers from an error
Retrigger	Starting up the Execute input again while executing an FB to a motion control FB that has the Execute input
RWr	Remote register of a link device. Word data input from a slave station to the master station (For some areas in a local station, data are input in the opposite direction.)
RWw	A remote register of the link device. Word data output from the master station to a slave station (For some areas in a local station, data are output in the opposite direction.)
RX	Remote input of the link device. Bit data input from a slave station to the master station (For some areas in a local station, data are input in the opposite direction.)
RY	Remote output of the link device. Bit data output from the master station to a slave station (For some areas in a local station, data are output in the opposite direction.)
Safety communications	A function to exchange safety data between safety stations on the same network
Set position	A commanded position address rounded by the ring counter. It is affected by a current position change.
Set velocity	A current control value that is generated by motion operation
Single axis synchronization	A control to output the position information (command) of Slave (slave axis) 1-axis that is synchronized with Master (master axis)
Software reboot	Resetting only motion area and network area without a hardware reboot
Standard station	A slave station other than motion synchronization (control) stations
Start	Executing a motion FB for the first time to an axis or an axes group
Switching speed	• The set velocity of the on-going motion control FB when specifying Aborting • The set velocity when the on-going motion control FB reached the target position when specifying Blending
System basic cycle	A basic cycle of fixed cycle processing such as motion operation processing
System start	Carrying out an initialization processing of the Motion module by power ON or a reboot of the system
Target position	A commanded position address that is specified in the motion FB
Transient transmission	A function of data communication unperiodically among nodes (station) on network. A function used to send message to the target station when requested by a link dedicated instruction or the engineering tool Communication is available with station on another network via relay station, or gateway.
Under-control FB	Motion control FB that is in execution on the axis and the axes group (FB that Active is TRUE)
Virtual axis	An axis that is not linked with a slave device on network
Virtual encoder axis	An axis that generates commanded position from a variable. It is used for the master axis of the single axis synchronous control.
Virtual drive axis	A virtual axis that can generate a command virtually
Virtual linked axis	An axis that connects FBs and transmits a command in the single axis synchronous control

GENERIC TERMS AND ABBREVIATIONS

Unless otherwise specified, this manual uses the following generic terms and abbreviations.

Generic term/abbreviation	Description
CC-Link IE	A generic term for the following items: • CC-Link IE TSN • CC-Link IE Controller Network (MELSEC iQ-R CC-Link IE Controller Network User's Manual (Application)) • CC-Link IE Field Network (MELSEC iQ-R CC-Link IE Field Network User's Manual (Application))
CPU module	An abbreviation for the MELSEC iQ-R series CPU module
csp	An abbreviation for cyclic synchronous position mode (One of the control modes for the driver side)
cst	An abbreviation for cyclic synchronous torque mode (One of the control modes for the driver side)
csv	An abbreviation for cyclic synchronous velocity mode (One of the control modes for the driver side)
ct	The abbreviation for continuous operation to torque control mode (One of the control modes for the driver side)
GOT	A generic term for Mitsubishi Electric Graphic Operation Terminal GOT1000 and GOT2000 series
hm	An abbreviation for homing mode (One of the control modes for the driver side)
MCFB	An abbreviation for Motion Control FB
MR-J5(W)-G	A generic term for MR-J5-_G_(-RJ)/MR-J5W_-_G/MR-J5D_-_G_ servo amplifier
PDO	An abbreviation for Process Data Object. Aggregation of application objects transferred periodically between multiple CANopen nodes.
RAS	An abbreviation for Reliability, Availability, and Serviceability. This term refers to the overall usability of automated equipment.
RD78G(H)	A generic term for RD78G_, RD78GH_ (high performance version)
SDO	An abbreviation for Service Data Object. A message to access an object entry in the object dictionary of a CANopen node. Data is sent/received between the stations aperiodically.
SLMP	A generic term for SeamLess Message Protocol. This protocol is used to access an SLMP-compatible device from an external device (such as a personal computer or HMI (Human Machine Interface)) or an SLMP-compatible module (such as the Ethernet-equipped module or modules on CC-Link IE TSN).
Operation profile data	A generic term for waveform data used for each control
Engineering tool	A generic term for GX Works3 and MR Configurator2
Safety station	A generic term for a station that performs safety communications and standard communications
Slave station	A generic term for a local station and remote station on CC-Link IE TSN
Software	A generic term for a firmware which configures the motion system. It consists of add-ons, baseSystem software and boot software.
Data link	A generic term for a cyclic transmission and a transient transmission
Drive unit	A generic term for motor drive devices such as a servo amplifier
Motion system	A generic term for software that performs the motion control and the network control
Reboot	A generic term for hardware reboot and software reboot

1 MOTION CONTROL FB OVERVIEW

This section describes how to use Motion control FBs and the common specifications.
The Motion control FBs that can be used in the Motion system include FBs established by PLCopen®. The basic specifications of I/O signals follow the PLCopen® Motion control FBs.

1.1 Motion Control FBs

This section describes the Motion control FBs.

How to use Motion control FBs

The procedure for using Motion control FBs is described below.

1. Create the Motion control FB instance (global label or local label) using the engineering tool.
2. Create a program which sets the I/O argument for the Motion control FB instance.
3. The control will be executed when Execute command (Execute) or Enable (Enable) of the Motion control FB is set to TRUE.

Point

When one Motion control FB instance is being executed (including during buffering), the instance cannot be executed on multiple axes by changing the axis input (AXIS_REF input or AXES_GROUP_REF input). The axis input of the Motion control FB can only be changed when an instance is not being executed. When the axis input is changed while an instance is being executed, a warning will occur and the change will be ignored. Therefore, ensure that the number of Motion control FB instances created is equal to the number of axes being controlled at the same time.

Operation of this function for each system status

Motion control FBs can only be executed during RUN.
○: Possible, ×: Not possible

System status	Operation availability
STOP	×
RUN	○
Moderate error	×
Major error	×

Motion control FB types

Motion control FBs are classified according the operation details and the execution method.

Management FBs/Operation FBs/Standard FBs

Motion control FBs are classified into the following types according to the operation details.

Type	Operation details
Management FB	• A Motion control FB that takes an axis or an axes group for the argument and does not change the axis status or the axes group status by execution. (There are some exceptions.) • In most cases, a management FB can execute multiple instances to an axis or an axes group at the same time.
Operation FB	• A Motion control FB that takes an axis or an axes group for the argument and changes the axis status or the axes group status by execution. • In most cases, an operation FB can be executed to only one axis or axes group. However, some FBs can be executed at the same time. • In most cases, the axis status or the axes group status will not be changed even if a management FB is executed while an operation FB is being executed. However, some FBs can cause a specific state transition.
Standard FB	• A Motion control FB that does not take an axis or axes group for the argument. • A standard FB can execute multiple instances at the same time. Since it is not related to the axis, it does not affect either operation FBs or management FBs.

Execute command (Execute) type/Enable (Enable) type

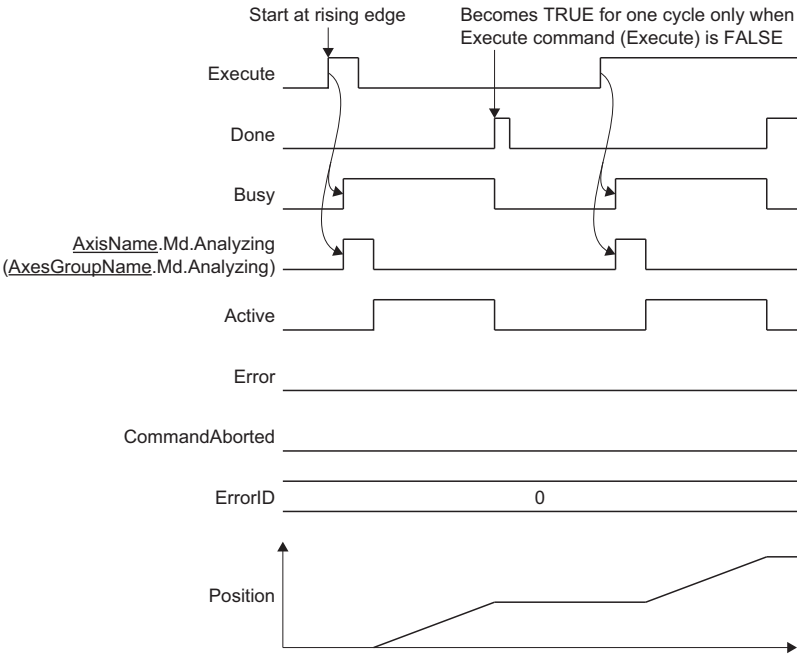
Some Motion control FBs are executed with Execute command (Execute), while others are executed with Enable (Enable).

Type	Execute command (Execute) type	Enable (Enable) type	Other types
Management FB	• MC_GroupEnable (Axes Group Enabled) • MC_GroupDisable (Axes Group Disabled) • MC_SetPosition (Current Position Change) • MCv_SetTorqueLimit (Torque Limit Value) • MC_WriteParameter (Parameter Write) • MC_Reset (Axis Error Reset) • MC_GroupReset (Axes Group Error Reset) • MC_TouchProbe (Touch Probe Enabled) • MC_AbortTrigger (Touch Probe Disabled) • MC_CamTableSelect (Cam Table Selection) • MCv_ChangeCycle (Current Value Change per Cycle) • MCv_MotionErrorReset (Motion Error Reset)	• MC_Power (Operation Available) • MC_SetOverride (Override Value Setting) • MC_ReadParameter (Parameter Read) • MCv_AllPower (All Axes Operation Available) • MC_GroupSetOverride (Axes Group Override Value Setting)	—
Operation FB	• MC_Home (OPR) • MC_Stop (Forced Stop) • MC_GroupStop (Group Forced Stop) • MC_MoveAbsolute (Absolute Value Positioning) • MC_MoveRelative (Relative Value Positioning) • MC_MoveVelocity (Speed Control) • MC_TorqueControl (Torque Control) • MCv_SpeedControl (Speed Control (Including Position Loop)) • MCv_MoveLinearInterpolateAbsolute (Absolute Value Linear Interpolation Control) • MCv_MoveLinearInterpolateRelative (Relative Value Linear Interpolation Control) • MCv_MoveCircularInterpolateAbsolute (Absolute Value Circular Interpolation Control) • MCv_MoveCircularInterpolateRelative (Relative Value Circular Interpolation Control) • MC_CamIn (Cam Operation Start) • MC_GearIn (Gear Operation Start) • MC_CombineAxes (Addition/Subtraction Positioning)	• MCv_BacklashCompensationFilter (Backlash Compensation Filter) • MCv_SmoothingFilter (Smoothing Filter) • MCv_DirectionFilter (Moving Direction Restriction Filter) • MCv_SpeedLimitFilter (Speed Limit Filter)	• MCv_Jog (JOG)
Standard FB	• MCv_ReadProfileData (Profile Read) • MCv_WriteProfileData (Profile Write)	—	—

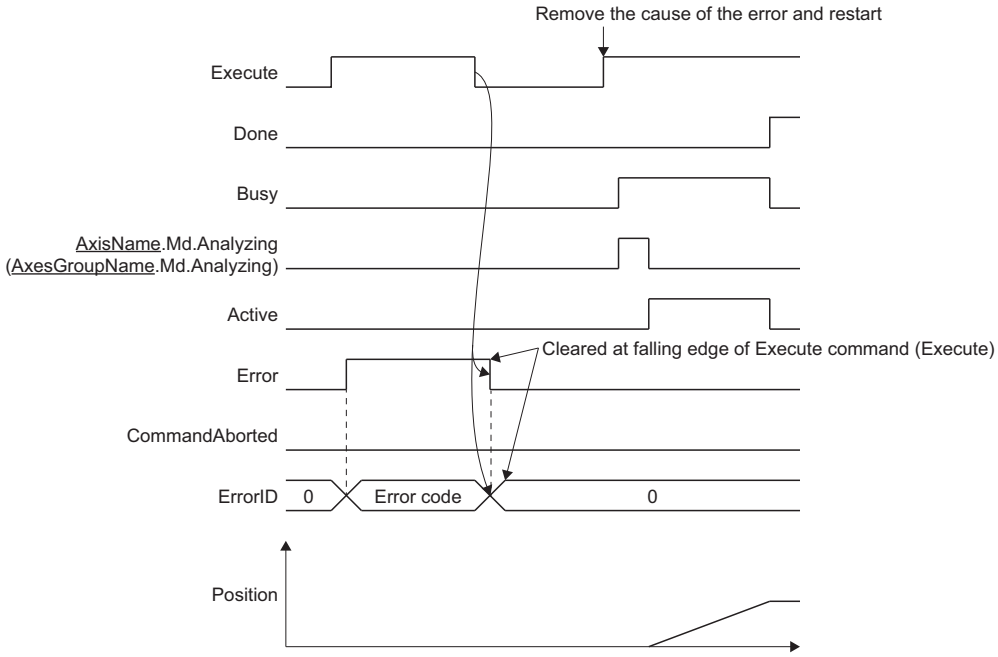
The basic operation of each Motion control FB by Execute command (Execute command) and Enable (Enable) is shown below. Note that some Motion control FBs have different specifications.

■Basic operation of Execute command (Execute) type Motion control FBs

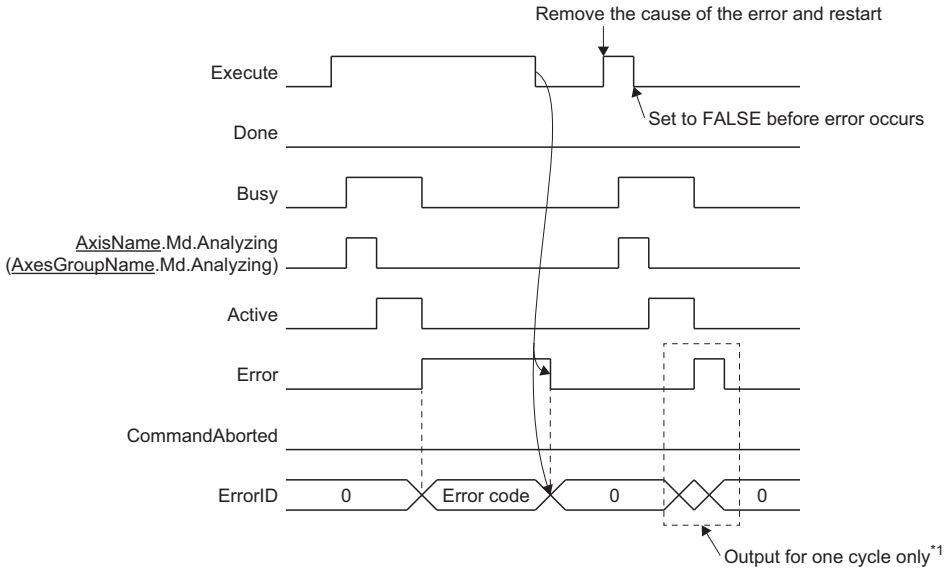
- Execute command (Execute) type FBs read the input parameters at the rising edge of Execute command (Execute) and then start operation. Once operation has started, the operation will be continued until it is completed even if Execute command (Execute) is set to FALSE.
- When operation has started, only one output variable among Executing (Busy), Execution completion (Done), Error (Error), and Abortion of execution (CommandAborted) becomes TRUE.
- Execution completion (Done), Error (Error), Error code (ErrorID), and Abortion of execution (CommandAborted) are reset at the falling edge of Execute command (Execute). Executing (Busy) and Controlling (Active) are not affected.
- When changing the input parameter during operation, the change is reflected at restart (retrigger) of Execute command (Execute) or by continuous update using Continuous update (ContinuousUpdate).
- Analyzing (AxisName.Md.Analyzing/AxesGroupName.Md.Analyzing) becomes TRUE from the rising edge of Executing (Busy), and Analyzing (AxisName.Md.Analyzing/AxesGroupName.Md.Analyzing) becomes FALSE after starting the operation.
- When Execute command (Execute) is used in pulse, Execution completion (Done) becomes TRUE for one cycle only.
- The timing chart for Execute command (Execute) type Motion control FBs is shown below.
  - When the FB is normal



• When the I/O variables are abnormal



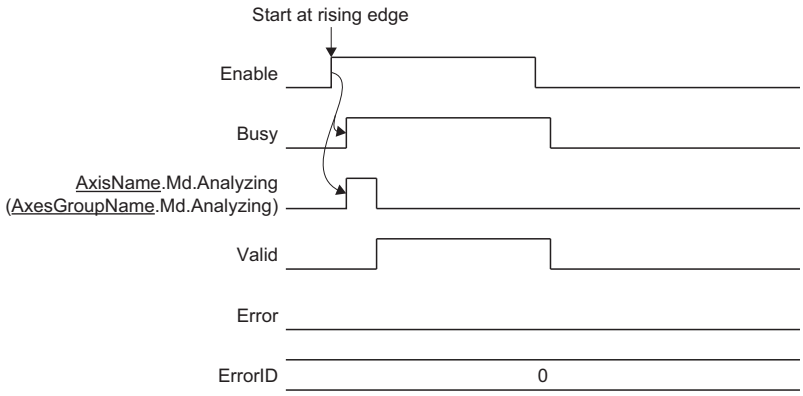
• When the input variables are abnormal



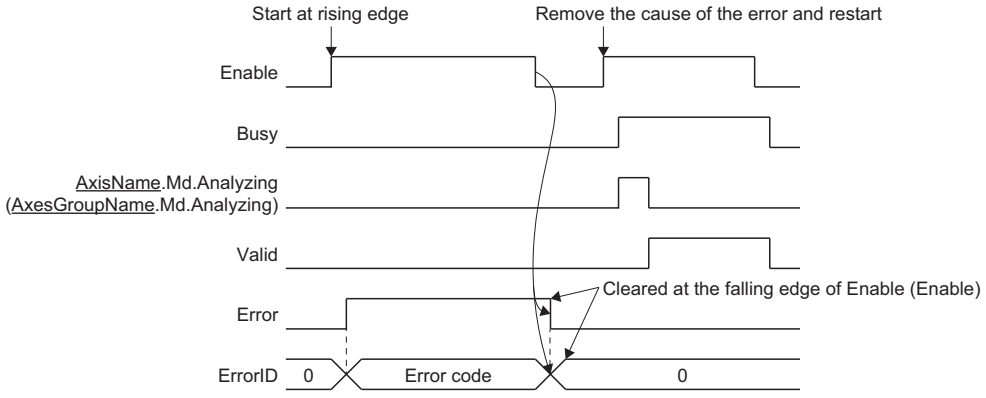
*1 The following operation is carried out when the termination condition of the FB (Execute command (Execute) is FALSE) is met.
For FBs not related to the axis or FBs which do not stop with decelerating, Error (Error) becomes TRUE for one cycle only and Error code (ErrorID) is output.
For FBs that need to perform a deceleration stop, Error (Error) becomes TRUE until the stop with decelerating and Error code (ErrorID) is maintained. At stop completion of the axis, Error (Error) becomes FALSE and Error code (Error ID) is cleared.

■Basic operation of Enable (Enable) type Motion control FBs

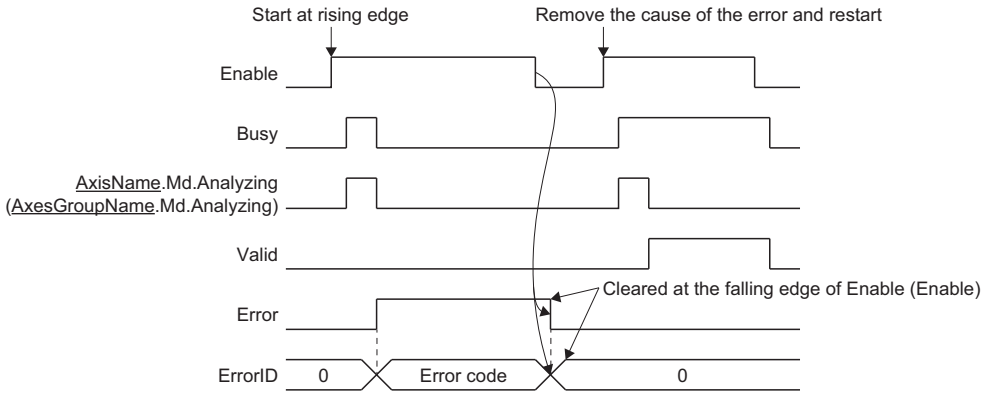
- Enable (Enable) type Motion control FBs are continuously executed while Enable (Enable) is TRUE.
- Output value valid (Valid) indicates that the output is a valid value. After Output value valid (Valid) becomes FALSE, each output does not change.
- Only one output variable among Output value valid (Valid)/Enabled (Enabled)/Executing (Busy), Error (Error), and Abortion of execution (CommandAborted) becomes TRUE.
- Analyzing (AxisName.Md.Analyzing/AxesGroupName.Md.Analyzing) becomes TRUE from the rising edge of Executing (Busy), and Analyzing (AxisName.Md.Analyzing/AxesGroupName.Md.Analyzing) becomes FALSE after starting the operation.
- The timing chart for Enable (Enable) type Motion control FBs is shown below.
  - When the FB is normal



  - When the I/O variables are abnormal



  - When the input variables are abnormal



Motion control types

The following types of axis and axes group control can be executed by operation type Motion control FBs.

Large classification	Middle classification	Small classification	Description
Axis control	Single axis control	Positioning control	Control in which Axis status (AxisName.Md.AxisStatus) is set to "5: During positioning operation (DiscreteMotion)" and the axis is moved to the target position.
		Continuous control	Control in which Axis status (AxisName.Md.AxisStatus) is set to "6: During continuous operation (ContinuousMotion)" and continuous control is performed for the axis.
		Synchronous control	Control which has Master axis (Master) and Slave axis (Slave) as I/O variables and performs synchronous control for the axis with the Axis status (AxisName.Md.AxisStatus) of Slave axis (Slave) set to "7: During synchronous operation (SynchronizedMotion)".
		Homing control	Control in which Axis status (AxisName.Md.AxisStatus) is set to "3: During home position return (Homing)" and continuous control is performed for the axis.
Axes group control	Multiple axes control	Positioning control	Control in which Axes group status (AxesGroupName.Md.GroupStatus) is set to "5: Operating (GroupMoving)" and the axis is moved to the target position.

Error processing

If an error occurs while the Motion control FB is executed, Error (Error) becomes TRUE and the error code is output to Error code (ErrorID). If using an axis, Axis status (AxisName.Md.AxisStatus) transits to "1: Stopping on error (ErrorStop)" at this time. If using an axes group, Axes group status (AxesGroupName.Md.GroupStatus) transits to "1: Stopping on error (GroupErrorStop)" at this time.

When the axis status of an available axis transits to the "1: Stopping on error (ErrorStop)" status, all buffering FBs are aborted. Error (Error) of the aborted FBs becomes TRUE.

After that, an error reset needs to be executed to start the axis or axes group.

For how to check errors of the axis or axes group and how to reset the error, refer to the following.

MELSEC iQ-R Motion Module User's Manual (Application)

Point

One of the following values is output to Error code (ErrorID). (The value output depends on the control.) Note that warning codes are not output.

- Axis error code (AxisName.Md.ErrorID)
- Axes group error code (AxesGroupName.Md.ErrorID)
- Latest Motion system error code (System.Md.ErrorID)

Errors (including warnings) when executing a Motion control FB on the CPU module side will be output as a Motion module error.

Error codes that are output to Motion control FB Error code (ErrorID) are as follows.

Error code	Description
0400H	No response from the Motion module within the regulated time period. Execute the FB again.
1C00H	These error codes correspond to the dedicated instruction error codes (1800H to 180FH) used in the FB. For details on these dedicated instruction error codes, refer to "User Function Execution Instruction" in the following manual. MELSEC iQ-R Programming Manual (Motion Module Instructions, Standard Functions/Function Blocks)
1C01H	
1C02H	
1C03H	
1C04H	
1C05H	
1C06H	
1C07H	
1C0FH	

Precautions

When the input number of the Motion module that is specified in the Motion control FB argument is incorrect, or when the Motion module for executing FBs cannot be identified, there will be no operation, or an error code will be output on the CPU module side.

Units used in control

Units for the position, velocity, acceleration/deceleration, and jerk that are used in the Motion system follow the technical units of the axis to be used.

The following types exist for the control values, such as the position and the velocity, that are used in the Motion system.

For details of the axis technical units and control values related to the position and the velocity, refer to the following.

MELSEC iQ-R Motion Module User's Manual (Application)

Type	Description
Commanded value	A value (target value) based on an input to the Motion control FB. (Commanded position and commanded velocity, etc.)
Set value	The current control value that is generated by Motion operation. (Set position and set velocity, etc.)
Actual value	A value obtained by converting the actual value received from the slave device by the real axis into the technical unit of the axis. (Actual position and actual velocity, etc.)

I/O variables used in Motion control FBs

This section describes the I/O variables used in Motion control FBs.
For Motion control FBs, the I/O variables, input variables, and output variables must be defined.

I/O variables

Variables such as Axis information (Axis) and Axes group info (AxesGroup) that set the axis and axes group of the driver to be controlled.
Produce and initialize the axis and axes groups using the Axis setting (☞ Page 448 Axis) and Axes group setting (☞ Page 458 Axes group) in the engineering tool.
The set axis and set axes group are assigned as an axis variable or an axes group variable in the global label data.
For details of the axis and axes groups, refer to the following.
MELSEC iQ-R Motion Module User's Manual (Application)

Input variables

These variables set operation conditions such as the target position and the commanded velocity.

Output variables

These variables output the FB status, driver status, whether there is an error, etc.

Classes

The following shows the classes of the I/O variables, input variables, and output variables.

Variable	Class
I/O variable	VAR_IN_OUT
Input variable	VAR_INPUT
Output variable	VAR_OUTPUT

Data types

Variables are classified into different types based on their bit length, processing method, value range, etc.
For details of data types, refer to the following.
☞ Page 22 Data types

Point

[About variable arrays]
Arrays must be set for certain variables.
Arrays are expressed after the data type with an element enclosed in "[ ]" marks.
When setting a label, set the range of the element shown within the "[ ]" marks. An error will occur during conversion of the program if the range set does not match the range of the element.
Arrays are set in "Data Type Selection" inside the global label or local label area of the engineering tool. For details, refer to the following.
☞ Page 461 Registering labels
<Example>
• For data type "LREAL[0..15]"
Set [16] as the element.

Omission of input arguments

When omitting FB inputs, the default value that is defined for each FB is applied. For details of the default values, refer to the section for the Motion control FB to be used.
When omitting the input for the velocity, etc. in the multiple start FB, the input value of the previous FB is carried over.

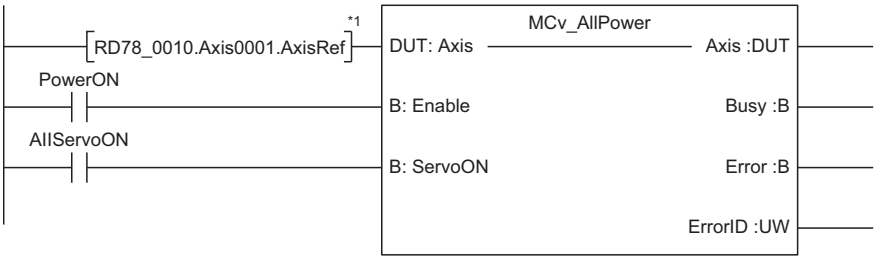
Refresh timing of inputs/outputs

Each argument of the FB is refreshed at the call timing of the FB. To control the input/output of the FB synchronizing with the operation cycle, call the FB from a fixed cycle program whose cycle is the same as the operation cycle.

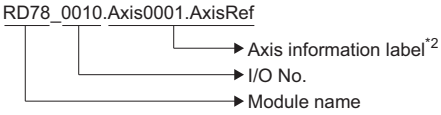
Specification of I/O No.

When using a Motion control FB on the CPU module side, the I/O No. of the target Motion module must be set in the I/O variables that are used in the Motion control FB (example: I/O No. (AxisName.AxisRef.StartIO)).
When setting the I/O No. on the CPU module side, AXIS_REF of the axis variables or axes group variables (AxisName.AxisRef for axis variables and AxesGroupName.AxesGroupRef for axes group variables) that were made public are used as Motion module labels.
The members (Axis No. (AxisNo), Axes group No. (GroupNo), I/O No. (StartIO)) in AXIS_REF of the axis variables and axes group variables already have the I/O No. specified, so the I/O No. can be set without designation by just setting the axis label.
When setting the I/O No., using the Axis information (AxisName.AxisRef) label to perform the setting is recommended. In addition, any Axis information (AxisName.AxisRef) label can be set if the label is made public as a Motion module label.

Ex.
When the label for Axis 1 (Axis0001) of the Motion module (I/O No: 0010) is used and MCv_AllPower (All Axes Operation Available) is executed from the CPU module side



*1 When using a label that was made public on the CPU module side, set the label with "Module name + Start I/O No." input at the start of the label.



*2 Information for the following members is included in the Axis information (AxisName.AxisRef) label.

Name	Setting value	Label name
Axis No.	1	Axis0001.AxisRef.AxisNo
Start I/O No.	H001*3	Axis0001.AxisRef.StartIO

*3 The start I/O number is the first 3 digits when expressed in 4-digit hexadecimal.

When using Motion control FBs on the Motion module side, the I/O No. does not need to be set in the I/O variable (example: I/O No. (AxisName.AxisRef.StartIO)). (It is ignored if set.)

Precautions

- I/O arguments of the Motion control FB are refreshed at the call timing of the FB, but the cycle that the control is actually executed in differs depending on the FB. For example, operation FBs perform the control at the operation cycle regardless of the FB call task (normal/fixed cycle). For details, refer to the section for the FB to be used.
- For Motion control FBs in operation (Executing (Busy) is TRUE), call the FB. If the FB is written in a control syntax such as an IF statement and the Motion control FB is not called, changes to the input variable values will not be reflected in control, and it will no longer be possible to update output variables such as errors and Execution Completion.
- The FB imports input values when Execution command (Execute) or Enable (Enable) is set to TRUE. Therefore, set input values before setting Execution command (Execute) and Enable (Enable) to TRUE. In addition, when changing multiple input arguments by a retrigger/continuous update of the FB, change them by the same task as the FB call task to match the import timing of each argument.
- When changing Execution command (Execute) to FALSE after it has been set to TRUE, wait until after Executing (Busy) has changed to the TRUE status before setting Execution command (Execute) to FALSE.
- The Motion control function uses real number data (floating-point data) for the axis current position and the command target position, etc. Therefore, it may include a truncation error. For example, when repeatedly executing positioning in relative position specification, such as with MC_MoveRelative (Relative Value Positioning), the truncation error may cumulate between the set position and the specified movement value. If the truncation error is a problem, execute positioning in absolute position specification, such as with MC_MoveAbsolute (Absolute Value Positioning). In addition, the truncation error that occurs in the real number data processing within the function can be suppressed by setting real number type parameters related to the Motion control function so that there is no number after the decimal point.
  For details of the real number data (floating-point data) used in Motion control FBs, refer to the following.
  MELSEC iQ-R Programming Manual (Motion Module Instructions, Standard Functions/Function Blocks)

Ex.

When using Setting A

In the following example, if the truncation error is a problem when using Setting A, Setting B becomes a change candidate.

Setting item	Setting detail	
	Setting A	Setting B
Position command unit	[mm]	[μm]
Ring counter upper limit value	100.3[mm]	100300[μm]
Ring counter lower limit value	-100.7[mm]	-100700[μm]
The minimum scale of the target position/movement amount specified by FB input	0.05[mm]	50[μm]

1.2 Motion Control FB Configuration

This section describes how to create a program using Motion control FBs.

The programming languages that can be used to create a program using Motion control FBs on the CPU module side or Motion module side are shown below.

○: Can be created, ×: Cannot be created

Programming language	CPU module side	Motion module side
Ladder	○	×
Function block diagram/ladder (FBD/LD)	○	×
Structured text (ST)	○	○

Point

To use labels registered to the Motion module when creating a program on the CPU module side, said labels must be made public to the CPU module by using the public label function. For details, refer to the following.

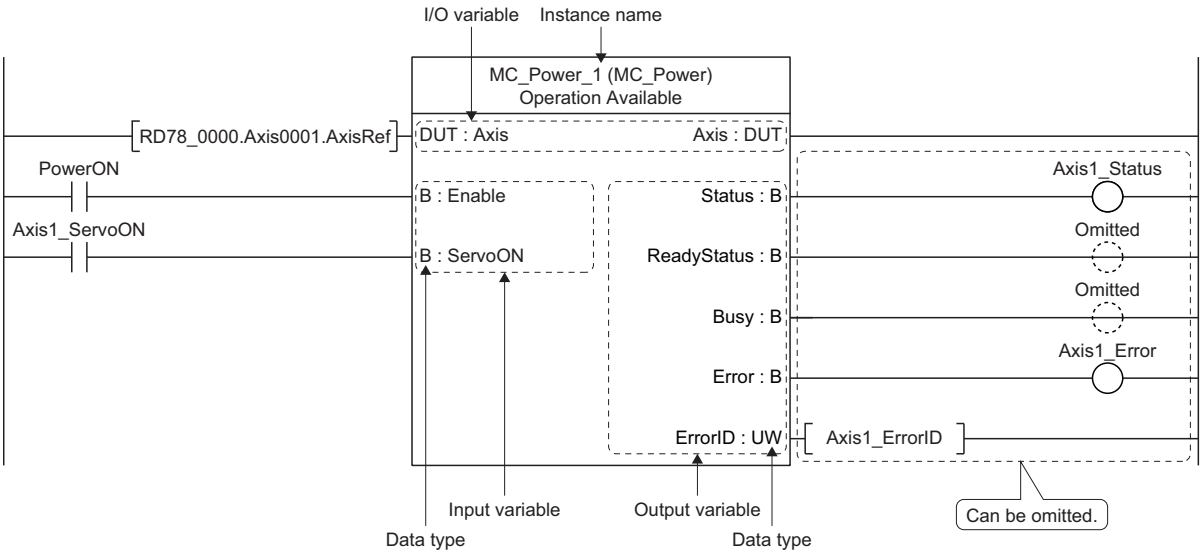
☞ Page 462 Creation procedure for programs using the public label function

Ladder

In the ladder language, the Motion control FB is arranged as follows.

Ex.

If MC_Power (Operation Available)



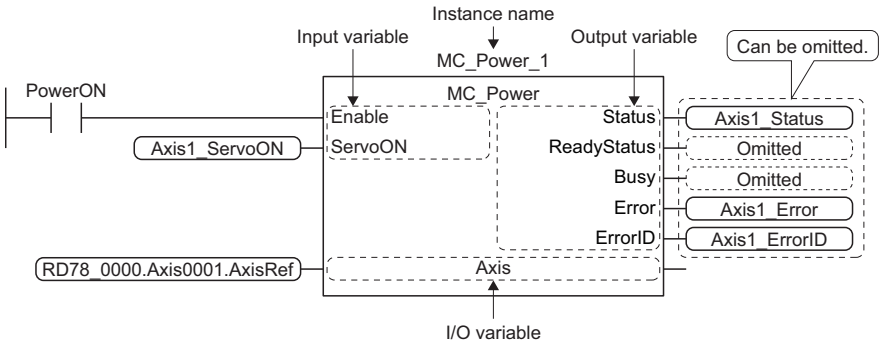
Name	Description
Instance name	The instance name assigned to each FB. The instance name can be changed.
I/O variable	These variables set axis variable names such as Axis information (Axis) and Axes group information (AxesGroup), or those for the driver to be controlled.
Input variable	These variables set operation conditions such as the target position and the commanded velocity. The setting of these variables can be omitted. When omitted, the default value is used.
Output variable	These variables output the FB status and driver status.
Data type	Displays the data type of I/O variables, input variables, and output variables. For details of the data type of I/O variables, input variables, and output variables, refer to the following. ☞ Page 22 Data types

FBD/LD (Function block diagram/ladder)

In the FBD/LD language, the Motion control FB is arranged as follows.

Ex.

If MC_Power (Operation Available)



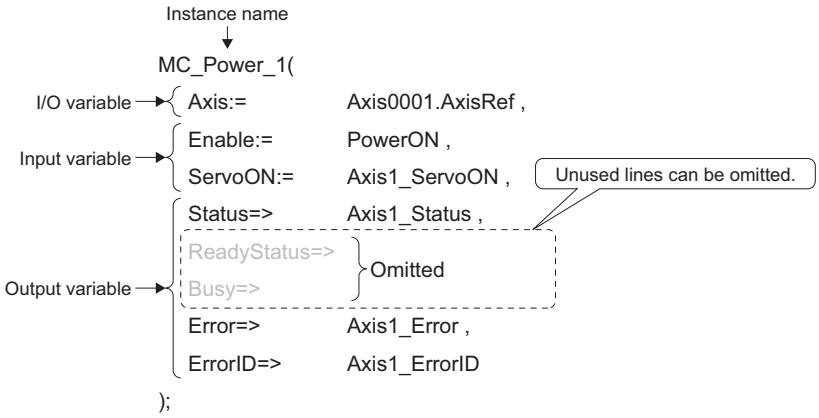
Name	Description
Instance name	The instance name assigned to each FB. The instance name can be changed.
I/O variable	These variables set axis variable names such as Axis information (Axis) and Axes group information (AxesGroup), or those for the driver to be controlled.
Input variable	These variables set operation conditions such as the target position and the commanded velocity. The setting of these variables can be omitted. When omitted, the default value is used.
Output variable	These variables output the FB status and driver status.

ST (Structured text)

In the ST language, the Motion control FB is arranged as follows.

Ex.

If MC_Power (Operation Available)



Name	Description
Instance name	The instance name assigned to each FB. The instance name can be changed.
I/O variable	These variables set axis variable names such as Axis information (Axis) and Axes group information (AxesGroup), or those for the driver to be controlled.
Input variable	These variables set operation conditions such as the target position and the commanded velocity. The setting of these variables can be omitted. When omitted, the default value is used.
Output variable	These variables output the FB status and driver status.

2 VARIABLES AND MOTION CONTROL FB

This section describes the lists of variables and Motion control FBs that are used in programs.

2.1 List of Variables

The following types of variables are handled in Motion control.

Variable	Description	Reference
Axis variable	Variables for axes configured with parameter data such as the axis type or monitor data such as the axis current position.	☞ Page 23 Axis variables
Axes group variable	Variables for axes groups configured with parameter data such as structuring axes or monitor data such as the axis group status.	☞ Page 46 Axes group variables
System variable	Variables for systems configured with parameter data relating to addons or monitor data such as the operation cycle.	☞ Page 53 System variables
Other structure variable	Variables for structures in a classification not listed above.	☞ Page 63 Other structure variables

Point

In this manual, variables listed with a portion of the variable underlined indicates that the variable name is defined by the user.

AxisName.AxisRef.AxisNo

↓

Variable defined by the user

How to read the list of variables

Each item in the list of variables has the following meaning.

■Import

Item	Description
Operation cycle	The timing for executing each operation cycle.
Normal	The timing for executing normal tasks of the Motion system.
Logging setting cycle	The timing for correcting the logging target data.
Service	The timing for executing the Motion service processing control.
Immediately	When an event occurs.
At start	When the FB starts.
At stop	When the FB stops.
At system start	At Motion system start.
When connecting device	At a slave device connection.
Ready ON	When "PLC READY (Y0)" turns ON.
System basic cycle	The timing for executing the system basic cycle processing.
Operation cycle (GroupEnable)	The timing for executing each operation cycle (GroupEnable).
When axes group is valid	The timing when the axes group becomes valid.

■Data types

Data type	Description
BOOL	Bit
WORD(HEX)	Word [unsigned]/bit string [16-bit] (hexadecimal)
WORD(UINT)	Word [unsigned]/bit string [16-bit]
DWORD(HEX)	Double word [unsigned]/bit string [32-bit] (hexadecimal)
DWORD(UDINT)	Double word [unsigned]/bit string [32-bit]
INT	Word [signed]
DINT	Double word [signed]
REAL	Single-precision real number
LREAL	Double-precision real number
TIME	Time
STRING(□)	Character string *: □ is a settable character string excluding Null. *: For STRING types, enclose the set character string in single quotation marks (').
WSTRING(□)	Character string [Unicode] *: □ is a settable character string excluding Null. *: For WSTRING types, enclose the set character string in double quotation marks (").

Point

For details of the data type, refer to the following.

MELSEC iQ-R Programming Manual (Motion Module Instructions, Standard Functions/Function Blocks)

■Attributes

Display	Description
LIST_WRITE_BACK	Label list (Write back to initial value possible)
LIST_READ_ONLY	Label list (Write back to initial value not possible)

Axis variables

Axis variables are assigned the following data types based on the axis type.

Axis type	Data type
Real drive axis	AXIS_REAL
Real encoder axis	AXIS_ENCODER
Virtual drive axis	AXIS_VIRTUAL
Virtual encoder axis	AXIS_VIRTUAL_ENCODER
Virtual link axis	AXIS_VIRTUAL_LINK

Point

The axis variables that can be set differ depending on the axis type used. (☞ Page 41 Variables settable as the axis variable for each axis type)

Each axis data type has the following members.

Member name	Data type*1*2	Description	Reference
AxisRef	AXIS_REF	The data structure for input/output of the motion control FBs. The type is fixed regardless of the axis type.	☞ Page 23 AxisName.AxisRef. (Axis information)
PrConst	AXIS_☐_PRM_CONST	Stores the axis parameter data (constant). Opens the setting value at the axis variable initialization. Reload to the control is not executed after the axis variable initialization.	☞ Page 24 AxisName.PrConst. (Axis parameter constant)
Pr	AXIS_☐_PRM	Stores the axis parameter data. Opens the default value at the axis variable initialization. Reload to the control is executed after the axis variable initialization. The fetch timing to the control changes depending on the parameter.	☞ Page 27 AxisName.Pr. (Axis parameter)
Md	AXIS_☐_MONI	Stores the axis monitor data. Executes the refresh in the fixed cycle for each monitor data.	☞ Page 31 AxisName.Md. (Axis monitor data)
Cd	AXIS_☐_CMD	Stores the axis control command data. Acquires the latest value every control operation cycle and uses it for the control.	☞ Page 40 AxisName.Cd. (Axis control data)

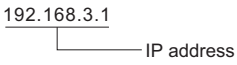
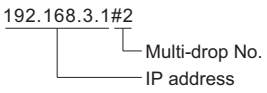
*1 ☐: The data type of each axis type

*2 The members of the data type differ depending on the axis type.

AxisName.AxisRef. (Axis information)

Variable name	Name	Import	Data type	Attribute	Description
AxisNo	Axis No.	—	WORD(UINT)	LIST_WRITE_BACK	This variable sets the axis No. • 0: Not set • 1 to 10000: Setting axis No.
StartIO	I/O No.	—	WORD(HEX)	LIST_WRITE_BACK	This variable sets the I/O No. • 000H to 0FFH: Start I/O number (the first 3 digits when expressed in 4-digit hexadecimal.) *: This variable is set when it is used on the CPU module side. The setting is ignored when used from the Motion module side.

AxisName.PrConst. (Axis parameter constant)

Variable name	Name	Import	Data type	Attribute	Description
AddressOfStation	Station address setting	At system start	WSTRING(63)	LIST_WRITE_BACK	Sets network address (IP address) of the slave station with a character string. Specify the multi-drop No. in order to distinguish logic axes when one station includes multiple logic axes, like the multi-axis drive unit. ■For a single axis device  ■For a multi-axis device <Example> C-axis of MR-J5W-G  *: The multi-drop No. is set as "# + No. (decimal)". · #0: A-axis · #1: B-axis · #2: C-axis *: The multi-drop No. can be omitted. When it is omitted, it is regarded as "#0".
AxisType	Axis type setting	At system start	INT (MC_AXIS_TYPE) ☞ Page 78 MC_AXIS_TYPE	LIST_WRITE_BACK	This variable sets the axis type. • 0: Real drive axis (DriveAxis) • 2: Real encoder axis (EncoderAxis) • 3: Virtual drive axis (VirtualDriveAxis) • 4: Virtual encoder axis (VirtualEncoderAxis) • 5: Virtual link axis (VirtualLinkAxis)
Encoder_AxisType	Real encoder axis type setting	At system start	INT (MC_ENCODER_AXIS_TYPE) ☞ Page 78 MC_ENCODER_AXIS_TYPE	LIST_WRITE_BACK	This variable sets the type of the real encoder axis. • 1: Via drive unit (Drive)
Encoder_CounterDisableSignal	Counter disabling signal	At system start	SIGNAL_SELECT ☞ Page 66 SIGNAL_SELECT (Signal select)	LIST_WRITE_BACK	This variable sets the signal to switch to the counter disable. Settings and operations specific to Counter disabling signal (Encoder_CounterDisableSignal) are shown below. ■Target (Target) Data type can only be set to [VAR], [AXIS], [DEV], and [CONST]. ■Signal detection method (Detection) Allows detection at the following setting level only. • 0: Detection at TRUE (HighLevel) • 1: Detection at FALSE (LowLevel) ■Compensation time (CompensationTime) Can only be set to 0.0[s]. ■Filter time (FilterTime) The setting range is "0.0 to 5.0[s]". *: If a value outside the range is set, "Out of Parameter Range (Axis) (error code: 1D80H)" occurs.
Encoder_RingCout_LowerValue	Encoder ring counter lower limit value	At system start	DINT	LIST_WRITE_BACK	This variable sets the encoder ring counter lower limit value. ■One-word PosActualValue • -32768 to 32767 *: If a value outside the 1-word range for is set, "Out of Parameter Range (Axis) (error code: 1D80H)" occurs. ■Two-word PosActualValue, or if omitted • -2147483648 to 2147483647

Variable name	Name	Import	Data type	Attribute	Description
Encoder_RingCout_UpperValue	Encoder ring counter upper limit value	At system start	DINT	LIST_WRITE_BACK	This variable sets the encoder ring upper limit value. ■One-word PosActualValue • -32768 to 32767 *: If a value outside the 1-word range for is set, "Out of Parameter Range (Axis) (error code: 1D80H)" occurs. ■Two-word PosActualValue, or if omitted • -2147483648 to 2147483647
HwStrokeLimit_FlsSignal	Upper limit signal	At system start	SIGNAL_SELECT ☞ Page 66 SIGNAL_SELECT (Signal select)	LIST_WRITE_BACK	This variable sets the signal to use the upper limit signal (FLS). Settings and operations specific to Upper limit signal (HwStrokeLimit_FlsSignal) are shown below. ■IO No. (StartIO) Ignores the input value. ■Target (Target) If Target is not set, signal disabled is determined and the signal undetection status is always set. Data type can only be set to [VAR], [DEV], and [CONST]. *: If an unusable data type is set, "Out of Parameter Range (Axis) (error code: 1D80H)" occurs. ■Signal detection method (Detection) Allows detection at the following setting level only. • 0: Detection at TRUE (HighLevel) • 1: Detection at FALSE (LowLevel) *: If a value outside the range is set, "Out of Parameter Range (Axis) (error code: 1D80H)" occurs. ■Compensation time (CompensationTime) Ignores the input value. ■Filter time (FilterTime) The setting range is "0.0 to 5.0[s]". *: If a value outside the range is set, "Out of Filter Time Setting Range of Each Axis Signal Warning (warning code: 0D24H)" occurs and Filter time is operated at "0.0".

Variable name	Name	Import	Data type	Attribute	Description
HwStrokeLimit_RlsSignal	Lower limit signal	At system start	SIGNAL_SELECT ☞ Page 66 SIGNAL_SELECT (Signal select)	LIST_WRITE_BACK	This variable set the signal to use the lower limit signal (RLS). Settings and operations specific to Lower limit signal (HwStrokeLimit_RlsSignal) are shown below. ■IO No. (StartIO) Ignores the input value. ■Target (Target) If Target is not set, signal disabled is determined and the signal undetection status is always set. Data type can only be set to [VAR], [DEV], and [CONST]. *: If an unusable data type is set, "Out of Parameter Range (Axis) (error code: 1D80H)" occurs. ■Signal detection method (Detection) Allows detection at the following setting level only. • 0: Detection at TRUE (HighLevel) • 1: Detection at FALSE (LowLevel) *: If a value outside the range is set, "Out of Parameter Range (Axis) (error code: 1D80H)" occurs. ■Compensation time (CompensationTime) Ignores the input value. ■Filter time (FilterTime) The setting range is "0.0 to 5.0[s]". *: If a value outside the range is set, "Out of Filter Time Setting Range of Each Axis Signal Warning (warning code: 0D24H)" occurs and Filter time is operated at "0.0".
OperationCycle	Control cycle setting	At system start	INT	LIST_WRITE_BACK	This variable sets the control operation cycle. • 0: Operates at the first operation cycle.
PositioningRange	Positioning range	At system start	LREAL	LIST_WRITE_BACK	This variable sets the positioning range. • 0.0, 200000000.0 to 1000000000000.0
PosRestoration_AbsPosBase	Absolute position reference setting	At system start	INT (MC_POS_SOURCE) ☞ Page 79 MC_POS_SOURCE	LIST_WRITE_BACK	When using the absolute position system, this variable sets the current position to be the basis of the current position restoration. • 3: Feed machine position (FeedMachinePosition)
PosRestoration_AbsPosEnable	Absolute position control setting	At system start	INT (MC_ABS_SYSTEM) ☞ Page 78 MC_ABS_SYSTEM	LIST_WRITE_BACK	This variable sets the absolute position control. • 0: Disable absolute position system (ABSDisabled) • 1: Enable absolute position system (Enabled) • -1: Automatic setting (Acquire from connected device) (Auto)
RingCount_Enable	Ring counter enabled selection	At system start	BOOL	LIST_WRITE_BACK	This variable sets whether to enable/disable the ring counter. • FALSE: Disabled • TRUE: Enabled
RingCount_LowerValue	Ring counter lower limit value	At system start	LREAL	LIST_WRITE_BACK	This variable sets the ring counter lower limit value. • -10000000000.0 to 10000000000.0 *: Not imported when the ring counter is disabled
RingCount_UpperValue	Ring counter upper limit value	At system start	LREAL	LIST_WRITE_BACK	This variable sets the ring counter upper limit value. • -10000000000.0 to 10000000000.0 *: Not imported when the ring counter is disabled
SlaveEmulate_Enable	Slave emulation enabled	At system start	BOOL	LIST_WRITE_BACK	This variable sets whether the axis is used as an emulate axis. • FALSE: Invalid • TRUE: Valid

Variable name	Name	Import	Data type	Attribute	Description
SlaveObject	Slave object data	At system start	SLAVE_OBJECT_REAL ☞ Page 72 SLAVE_OBJECT_REAL	LIST_WRITE_BACK	This variable sets the slave object data to the real drive axis.
SlaveObject	Slave object data	At system start	SLAVE_OBJECT_ENCODER ☞ Page 74 SLAVE_OBJECT_ENCODER	LIST_WRITE_BACK	This variable sets the slave object data to the real encoder axis.
SlaveObject	Slave object data	At system start	SLAVE_OBJECT_VIRTUAL_ENCODER ☞ Page 74 SLAVE_OBJECT_VIRTUAL_ENCODER	LIST_WRITE_BACK	This variable sets the slave object data to the virtual encoder axis.
TorqueLimit_Max	Torque limit maximum value	At system start	LREAL	LIST_WRITE_BACK	This variable sets the maximum value that can be specified as the torque limit value. (Common in the positive direction and the negative direction) • 0.0 to 1000.0[%]
TorqueLimit_NegativeInitial	Negative direction torque limit initial value	At system start	LREAL	LIST_WRITE_BACK	This variable sets the initial value of Negative direction torque limit value (AxisName.Cd.TorqueLimit_Negative). • 0.0 to 1000.0[%]
TorqueLimit_PositiveInitial	Positive direction torque limit initial value	At system start	LREAL	LIST_WRITE_BACK	This variable sets the initial value of Positive direction torque limit value (AxisName.Cd.TorqueLimit_Positive). • 0.0 to 1000.0[%]
FastOperationMode	High-speed mode setting	At system start	WORD(HEX)	LIST_WRITE_BACK	This variable switches the operation process of the set axis to high-speed mode. • 5FE2H: High-speed mode • Others: Normal mode *: Only the real drive axis is supported.

AxisName.Pr. (Axis parameter)

Variable name	Name	Import	Data type	Attribute	Description
AccelerationLimit	Acceleration limit value	Ready ON	LREAL	LIST_WRITE_BACK	This variable sets the acceleration limit value. • 0.0000, 0.0001 to 2147483647.0[U/s^2] *: When the acceleration/deceleration method is set to "1: Acceleration/deceleration time-fixed method (mcFixedTime)", limiting of the acceleration is not performed. *: When set to "0.0000", limiting of the acceleration is not performed. *: When a real number less than 0.0001 is set, the value is imported as "0.0000".
AccelerationZeroBehavior	Operation selection at start acceleration/ deceleration 0	Ready ON	INT (MC_ACC_ZERO_MODE) ☞ Page 79 MC_ACC_ZERO_MODE	LIST_WRITE_BACK	This variable sets the behavior when "0.0" is set as the acceleration, the deceleration, or the acceleration/deceleration time at start. • -1: Error (Not started) (ACCError) • 1: Maximum acceleration/deceleration (MaximumAcceleration)
CmdInPos_Width	Command in-position width	Ready ON	LREAL	LIST_WRITE_BACK	This variable sets the command in-position width. • 0.0: Function disabled • 0.000000001 to 10000000000.0: Function enabled *: When a real number less than 0.000000001 is set, the value is imported as "0.0".

Variable name	Name	Import	Data type	Attribute	Description
DecelerationLimit	Deceleration limit value	Ready ON	LREAL	LIST_WRITE_BACK	This variable sets the deceleration limit value. • 0.0000, 0.0001 to 2147483647.0[U/s^2] *: When the acceleration/deceleration method is set to "1: Acceleration/deceleration time-fixed method (mcFixedTime)", limiting of the acceleration is not performed. *: When set to 0.0000, limiting of the deceleration is not performed. *: When a real number less than 0.0001 is set, the value is imported as "0.0000".
Drive_UnitConvRatioNum	Driver unit conversion numerator	Ready ON	DWORD(UDINT)	LIST_WRITE_BACK	This variable sets the numerator used to convert the command unit of the Motion system to the command unit of the driver. • 1 to 2147483647
Drive_UnitConvRatioDen	Driver unit conversion denominator	Ready ON	DWORD(UDINT)	LIST_WRITE_BACK	This variable sets the denominator used to convert the command unit of the Motion system to the command unit of the driver. • 1 to 2147483647
ForcedStop_Signal	Forced stop signal	Ready ON	SIGNAL_SELECT ☞ Page 66 SIGNAL_SELECT (Signal select)	LIST_WRITE_BACK	This variable sets the signal to use the forced stop. Settings and operations specific to Forced stop signal (ForcedStop_Signal) are shown below. ■IO No. (StartIO) Ignores the input value. ■Target (Target) If Target is not set, signal disabled is determined and the signal undetection status is always set. Data type can only be set to [VAR], [DEV], and [CONST]. *: If an unusable data type is set, "Out of Parameter Range (Axis) (error code: 1D80H)" occurs. ■Signal detection method (Detection) Allows detection at the following setting level only. • 0: Detection at TRUE (HighLevel) • 1: Detection at FALSE (LowLevel) *: If a value outside the range is set, "Out of Parameter Range (Axis) (error code: 1D80H)" occurs. ■Compensation time (CompensationTime) Ignores the input value. ■Filter time (FilterTime) The setting range is "0.0 to 5.0[s]". *: If a value outside the range is set, "Out of Filter Time Setting Range of Each Axis Signal Warning (warning code: 0D24H)" occurs and Filter time is operated at "0.0".
Homing_Required	Home position return required or not	Ready ON	BOOL	LIST_WRITE_BACK	This variable sets whether homing is required or not. When homing is not required, set this variable to FALSE. When set to FALSE, the homing request will not become TRUE at the current position restoration. • FALSE: Homing is unnecessary • TRUE: Homing is necessary
JerkLimit	Jerk limit value	Ready ON	LREAL	LIST_WRITE_BACK	This variable sets the jerk limit value. • 0.0000, 0.0001 to 2147483647.0[U/s^3] *: When set to "0.0000", limiting of the jerk is not performed. *: When a real number less than 0.0001 is set, the value is imported as "0.0000".
OverrunOperation	Operation setting at overrun	Ready ON	INT (MC_DECEL_STOP_MODE) ☞ Page 79 MC_DECEL_STOP_MODE	LIST_WRITE_BACK	This variable sets the operation performed when the target position is exceeded during positioning operation. • 1: Immediately stop (ImmediateStop) • 2: Maximum acceleration/deceleration (KeepCurrentAcc)

Variable name	Name	Import	Data type	Attribute	Description
StartableAtUnhomed	Start permission at home position return uncompleted	Ready ON	BOOL	LIST_WRITE_BACK	This variable sets whether axis start is allowed or not when homing is incomplete. • FALSE: Not allowed • TRUE: Allowed
StopMode_Deceleration	Deceleration at stop	Ready ON	LREAL	LIST_WRITE_BACK	This variable sets the deceleration/deceleration time at deceleration stop by stop cause occurrence. ■When using acceleration/deceleration method specifying acceleration/deceleration (acceleration/deceleration specification method) • 0.0000, 0.0001 to 2147483647.0[U/s^2] ■ When using acceleration/deceleration method specifying acceleration/deceleration time (acceleration/deceleration time-fixed method) • 0.000000 to 0.000001 to 8400.0[s] *: If this variable is set to "0", immediately stops regardless of the setting of Operation selection at start acceleration/deceleration 0 (AccelerationZeroBehavior) *: Clamps by the upper limit value when the upper limit value is the range at stop by factors, and is treated as "0 (Immediate stop)" when the lower limit value is the range.
StopMode_DecelerationCurve	Stop selection at deceleration to stop	Ready ON	INT (MC_STOP_CURVE_MODE) ☞ Page 79 MC_STOP_CURVE_MODE	LIST_WRITE_BACK	This variable sets the operation performed when a stop cause occurs during deceleration (including a stop cause and automatic deceleration). • 1: Recreate deceleration curve (OverrideCurve)
StopMode_General	Stop selection at stop by factors	Ready ON	INT (MC_DECEL_STOP_MODE) ☞ Page 79 MC_DECEL_STOP_MODE	LIST_WRITE_BACK	This variable sets the operation performed when a stop cause occurs. • 1: Immediate stop (ImmediateStop) • 2: Keep current acceleration/deceleration (KeepCurrentAcc) • 3: Alternative acceleration/deceleration (AlternativeAcc) *: Immediate stop will be executed when executing an FB without deceleration specification.
StopMode_HwStrokeLimit	Stop selection at hardware stroke limit error occurrence	Ready ON	INT (MC_DECEL_STOP_MODE) ☞ Page 79 MC_DECEL_STOP_MODE	LIST_WRITE_BACK	This variable sets the operation performed when a hardware stroke limit error occurs. • 1: Immediate stop (ImmediateStop) • 2: Keep current acceleration/deceleration (KeepCurrentAcc) • 3: Alternative acceleration/deceleration (AlternativeAcc) *: Immediate stop will be executed when executing an FB without deceleration specification.
StopMode_ServoOff	Process selection at servo OFF command during operation	Ready ON	INT (MC_DECEL_STOP_MODE) ☞ Page 79 MC_DECEL_STOP_MODE	LIST_WRITE_BACK	This variable sets the operation performed when ServoON becomes FALSE during operation. • 0: Ignore (Ignore) • 4: Servo OFF after immediate stop (ServoOffAfterImmediateStop) • 5: Servo OFF after deceleration to stop (ServoOffAfterDecelStop)
StopMode_SwStrokeLimit	Stop selection at software stroke limit error occurrence	Ready ON	INT (MC_DECEL_STOP_MODE) ☞ Page 79 MC_DECEL_STOP_MODE	LIST_WRITE_BACK	This variable sets the operation performed when a software stroke limit error occurs. • 1: Immediate stop (ImmediateStop) • 2: Keep current acceleration/deceleration (KeepCurrentAcc) • 3: Alternative acceleration/deceleration (AlternativeAcc) *: Immediate stop will be executed when executing an FB without deceleration specification.

Variable name	Name	Import	Data type	Attribute	Description
StopOption_DriverTargetIgnored	Driver Command Discard Detection Setting	Ready ON	BOOL	LIST_WRITE_BACK	This variable sets whether to detect the command discard status of the driver module and stop at an error during operation of the axis or not. • FALSE: Detection Disabled • TRUE: Detection Enabled
StopSignal	Stop signal	Ready ON	SIGNAL_SELECT ☞ Page 66 SIGNAL_SELECT (Signal select)	LIST_WRITE_BACK	This variable sets the signal to use the stop signal (STOP) of the external input signal. Settings and operations specific to Stop signal (StopSignal) are shown below. ■IO No. (StartIO) Ignores the input value. ■Target (Target) If Target is not set, signal disabled is determined and the signal undetection status is always set. Data type can only be set to [VAR], [DEV], and [CONST]. *: If an unusable data type is set, "Out of Parameter Range (Axis) (error code: 1D80H)" occurs. ■Signal detection method (Detection) Allows detection at the following setting level only. • 0: Detection at TRUE (HighLevel) • 1: Detection at FALSE (LowLevel) *: If a value outside the range is set, "Out of Parameter Range (Axis) (error code: 1D80H)" occurs. ■Compensation time (CompensationTime) Ignores the input value. ■Filter time (FilterTime) The setting range is "0.0 to 5.0[s]". *: If a value outside the range is set, "Out of Filter Time Setting Range of Each Axis Signal Warning (warning code: 0D24H)" occurs and Filter time is operated at "0.0".
SwStrokeLimit_Lower	Software stroke limit lower value	Ready ON	LREAL	LIST_WRITE_BACK	This variable sets the software stroke limit lower value. • -10000000000.0 to 10000000000.0
SwStrokeLimit_Target	Software stroke limit target	Ready ON	INT (MC_POS_SOURCE) ☞ Page 79 MC_POS_SOURCE	LIST_WRITE_BACK	This variable sets the target of the software stroke limit. • -1: Disabled (Invalid) • 1: Set position (SetPosition) • 3: Feed machine position (FeedMachinePosition)
SwStrokeLimit_Upper	Software stroke limit upper value	Ready ON	LREAL	LIST_WRITE_BACK	This variable sets the software stroke limit upper value. • -10000000000.0 to 10000000000.0
Unit_Position	Position command unit	Ready ON	DWORD(HEX)	LIST_WRITE_BACK	This variable sets the position command unit to be used in the Motion control. • 00000000H: pulse • 00010000H: m • FD010000H: mm • FA010000H: μm • F7010000H: nm • 00410000H: degree • FA410000H: $\times 10^{-6}$degree • 00B40000H: Revolution • 00C00000H: inch • 00FF0000H: arbitrary unit
Unit_PositionString	Position command unit string	Ready ON	WSTRING(31)	LIST_WRITE_BACK	This variable sets the command unit to be used in the Motion control as a character string. This variable is set when Position command unit (Unit_Position) is set to "00FF0000H: arbitrary unit".

Variable name	Name	Import	Data type	Attribute	Description
Unit_Velocity	Velocity command unit	Ready ON	DWORD(HEX)	LIST_WRITE_BACK	This variable sets the velocity command unit to be used in the Motion control. • 00000300H: U/s • 00004700H: U/min • FD000300H: U/ms • FA000300H: U/μs • F7000300H: U/ns
VelocityLimit_Negative	Negative direction speed limit value	Ready ON	LREAL	LIST_WRITE_BACK	This variable sets the negative direction speed limit value. • 0.0001 to 2500000000.0
VelocityLimit_OverOperation	Operation setting at speed limit value exceeded	Ready ON	INT (MC_VELOCITY_LIMIT_MODE) ☞ Page 77 MC_VELOCITY_LIMIT_MODE	LIST_WRITE_BACK	This variable sets the operation performed at speed limit value exceeded. • 0: Ignore (Ignore) • 3: Immediate stop (ImmediateStop) *: When set to "0: Ignore (Ignore)", "Speed Limit Value Over Warning (warning code: 0D03H)" occurs at the time of detecting the set velocity exceeding the speed limit value, and the command exceeding the speed limit value is output as it is. *: When set to "3: Immediate stop (ImmediateStop)", "Speed Limit Value Over during Controlling (error code: 1A36H)" occurs at the time of detecting the set velocity exceeding the speed limit value, and the positioning control in execution stops.
VelocityLimit_Positive	Positive direction speed limit value	Ready ON	LREAL	LIST_WRITE_BACK	This variable sets the positive direction speed limit value. • 0.0001 to 2500000000.0

AxisName.Md. (Axis monitor data)

Variable name	Name	Import	Data type	Attribute	Description
AccelerationLimit	Acceleration limit value	Ready ON	LREAL	LIST_READ_ONLY	This variable stores the acceleration limit value.
AccelerationOverride	Acceleration override coefficient	Operation cycle	LREAL	LIST_READ_ONLY	This variable stores the acceleration override factor.
AccelerationZeroBehavior	Operation selection at start acceleration/ deceleration 0	Ready ON	INT (MC_ACC_ZERO_MODE) ☞ Page 79 MC_ACC_ZERO_MODE	LIST_READ_ONLY	This variable sets the behavior when "0.0" is set as the acceleration, the deceleration, or the acceleration/deceleration time at start. • -1: Error (Not started) (ACCError) • 1: Maximum acceleration/deceleration (MaximumAcceleration)
ActualPosition	Feedback position	Operation cycle	LREAL	LIST_READ_ONLY	This variable stores the actual current position. The value of "The actual position = the current position × driver unit conversion denominator/ numerator" is set. The actual position is a value which is rounded with the ring counter range.
ActualVelocity	Feedback speed	Operation cycle	LREAL	LIST_READ_ONLY	This variable stores the actual velocity. This velocity is calculated from the difference of the actual position. The value is positive when moving to the positive direction (Address increase direction), and the value is negative when moving to the negative direction (Address decrease direction).
Analyzing	Analyzing	Operation cycle	BOOL	LIST_READ_ONLY	This variable displays the status of positioning analyzing. • FALSE: Not analyzing • TRUE: Analyzing

Variable name	Name	Import	Data type	Attribute	Description
AutoDeceleration	Automatically decelerating	Operation cycle	BOOL	LIST_READ_ONLY	This variable displays the status of auto deceleration processing. TRUE is stored while the auto deceleration processing is performed. When the multiple start is performed, TRUE is set while the auto deceleration processing is executed during the last positioning point execution. FALSE is set when the control change is performed. During axes group operation, stores FALSE in (AxisName.Md.AutoDeceleration) of the structuring axis. • FALSE: Not during auto deceleration • TRUE: During auto deceleration
AxisName	Axis name	At system start	WSTRING(127)	LIST_READ_ONLY	This variable displays the axis name.
AxisStatus	Axis status	Operation cycle	INT (MC_AXIS_STATUS) ☞ Page 76 MC_AXIS_STATUS	LIST_READ_ONLY	This variable displays the axis status. • -1: Axis variable not initialized/axis parameter error (Invalid) • 0: Axis disabled (Disabled) • 1: Stopping on error (ErrorStop) • 2: Decelerating to stop (Stopping) • 3: During home position return (Homing) • 4: Standby (Standstill) • 5: During positioning operation (DiscreteMotion) • 6: During continuous operation (ContinuousMotion) • 7: During synchronous operation (SynchronizedMotion)
BufferingFBs	Number of buffering FBs	At start/at stop	INT	LIST_READ_ONLY	This variable displays the number of buffering FBs (0 to 2).
CmdInPos	Command in-position	Operation cycle	BOOL	LIST_READ_ONLY	This variable indicates whether the remaining distance to the target position is equal to or less than Command in-position width (AxisName.Md.CmdInPos_Width). • FALSE: More than Command in-position width • TRUE: Less than Command in-position width
CmdInPos_Width	Command in-position width	Ready ON	LREAL	LIST_READ_ONLY	This variable displays the command in-position width. Imports the setting value of Command in-position width (AxisName.Pr.CmdInPos_Width). When Command in-position width (AxisName.Md.CmdInPos_Width) is set to "0.0", the command in-position function of the axes group is disabled.
CommandedAcceleration	Specified acceleration	Operation cycle	LREAL	LIST_READ_ONLY	This variable stores the acceleration specified by the user. (Unit: U/s^2) When the acceleration/deceleration method is set to "1: Acceleration/deceleration time-fixed method (mcFixedTime)", this variable stores the specified acceleration/deceleration time. (Unit: s) During axes group operation, stores "0.0" in Specified acceleration (AxisName.Md.CommandedAcceleration) of the structuring axis.
CommandedDeceleration	Specified deceleration	Operation cycle	LREAL	LIST_READ_ONLY	This variable stores the deceleration specified by the user. (Unit: U/s^2) When the acceleration/deceleration method is set to "1: Acceleration/deceleration time-fixed method (mcFixedTime)", it is an illegal value. During axes group operation, stores "0.0" in Specified deceleration (AxisName.Md.CommandedDeceleration) of the structuring axis.

Variable name	Name	Import	Data type	Attribute	Description
CommandedJerk	Specified jerk	Operation cycle	LREAL	LIST_READ_ONLY	This variable stores the jerk specified by the user. (Unit: U/s^3) During axes group operation, stores "0.0" in Specified jerk (AxisName.Md.CommandedJerk) of the structuring axis.
CommandedPosition	Specified position	Operation cycle	LREAL	LIST_READ_ONLY	This variable stores the Target position (Position) and Movement amount (Distance) set in the positioning Motion control FB being controlled. Stores "0.0" after the positioning is completed.
CommandedVelocity	Specified speed	Operation cycle	LREAL	LIST_READ_ONLY	This variable stores the Specified speed imported to the operation Motion control FB being controlled. Stores "0.0" after the control is completed. During axes group operation, stores "0.0" in Specified speed (AxisName.Md.CommandedVelocity) of the structuring axis.
Cst_SetTorque	Command current torque at torque control	Operation cycle	LREAL	LIST_READ_ONLY	This variable stores the set torque at the cyclic synchronous torque mode.
Cst_TargetTorque	Target torque at torque control	Operation cycle	LREAL	LIST_READ_ONLY	This variable stores the target torque at the cyclic synchronous torque mode.
CumulativePosition	Cumulative current position	Operation cycle	LREAL	LIST_READ_ONLY	This variable stores the current position address. When the current position is changed, the address is changed to the current position change value. The ring address is "-10000000000.0 ≤ cumulative current position < 10000000000.0".
DecelerationLimit	Deceleration limit value	Ready ON	LREAL	LIST_READ_ONLY	This variable stores the deceleration limit value.
Drive_RPDO[1..64]	RPDO (Slave device→Motion system) Mapping	When connecting device	DWORD(HEX)	LIST_READ_ONLY	This variable displays the RPDO mapping information.
Drive_TPDO[1..64]	TPDO (Motion system→Slave device) Mapping	When connecting device	DWORD(HEX)	LIST_READ_ONLY	This variable displays the TPDO mapping information.
Drive_UnitConvRatioNum	Driver unit conversion numerator	Ready ON	DWORD(UDINT)	LIST_READ_ONLY	This variable stores the numerator used to convert to the command unit of the driver.
Drive_UnitConvRatioDen	Driver unit conversion denominator	Ready ON	DWORD(UDINT)	LIST_READ_ONLY	This variable stores the denominator used convert to the command unit of the driver.
Driver_Mode	Driver control mode	Operation cycle	INT (MC_DRIVE_MODE) ☞ Page 77 MC_DRIVE_MODE	LIST_READ_ONLY	This variable displays the driver control mode status. • 6: Homing (hm) • 8: Position control (csp) • 9: Velocity control (csv) • 10: Torque control (cst) • -104: Continuous operation to torque control (ct)
Driver_ReadyOn	Driver ready ON status	Operation cycle	BOOL	LIST_READ_ONLY	This variable displays the connected driver status. This variable becomes TRUE when Driver status (AxisName.Md.Driver_State) is set to one of the following. • 2: Fault Reaction Active • 5: Switched On • 6: Operation Enable • 7: Quick Stop Active
Driver_ServoOn	Driver servo ON status	Operation cycle	BOOL	LIST_READ_ONLY	This variable displays the connected driver status. This variable becomes TRUE when Driver status (AxisName.Md.Driver_State) is set to one of the following. • 6: Operation Enable • 7: Quick Stop Active

Variable name	Name	Import	Data type	Attribute	Description
Driver_State	Driver status	Immediately	INT (MC_DRIVE_STATE) ☞ Page 79 MC_DRIVE_STATE	LIST_READ_ONLY	This variable displays the driver status. • 0: Not Ready To Switch On • 1: Fault • 2: Fault Reaction Active • 3: Switch On Disabled • 4: Ready To Switch On • 5: Switched On • 6: Operation Enable • 7: Quick Stop Active • -1: Invalid
DriverError	Driver module error detection	Immediately	BOOL	LIST_READ_ONLY	This variable displays the drive unit error status. • FALSE: No drive unit error • TRUE: Drive unit error present
DriverErrorID	Driver module error detection	Immediately	WORD(HEX)	LIST_READ_ONLY	This variable displays the upper 16 bits of "Current alarm (2A41H)" of the slave object at a drive unit error occurrence.
DriverErrorDetailID	Driver module error detail code	Immediately	WORD(HEX)	LIST_READ_ONLY	This variable displays the lower 16 bits of "Current alarm (2A41H)" of the slave object at a drive unit error occurrence.
Encoder_Connected	Connection status	Operation cycle	BOOL	LIST_READ_ONLY	This variable displays the virtual encoder axis connection status. • FALSE: Not connected • TRUE: Connected
Encoder_CounterDisable	Counter disabled	Operation cycle	BOOL	LIST_READ_ONLY	This variable displays the status of the input from the encoder. • FALSE: Counter enabled • TRUE: Counter disabled
Error	Axis error detection	Immediately	BOOL	LIST_READ_ONLY	This variable displays the axis error status. • FALSE: No axis error • TRUE: Axis error present
ErrorID	Axis error code	Immediately	WORD(HEX)	LIST_READ_ONLY	This variable displays the error code of the generated axis error.
FeedMachinePosition	Feed function position	Operation cycle	LREAL	LIST_READ_ONLY	This variable stores the current position address by the machine coordinate. The cumulative value from the homing complete position is set. The address is not changed even if the current position is changed. The ring address is "-10000000000.0 ≤ feed function position < 10000000000.0".
FollowupDisable	Follow-up disabled	Operation cycle	BOOL	LIST_READ_ONLY	This variable displays the follow-up disable status. When it is disabled, the follow-up is not performed. • FALSE: Follow-up enabled • TRUE: Follow-up disabled
ForcedStop_Released	Forced stop cancelling	Operation cycle	BOOL	LIST_READ_ONLY	This variable displays the forced stop cancel status. • FALSE: Forced stop • TRUE: Forced stop release
ForcedStop_Signal	Forced stop signal	Ready ON	SIGNAL_SELECT ☞ Page 66 SIGNAL_SELECT (Signal select)	LIST_READ_ONLY	This variable displays the input status of the Forced stop signal. Outputs specific to Forced stop signal (ForcedStop_Signal) are shown below. ■IO No. (StartIO) "0" is always displayed. ■Target (Target) The fetch result of the parameter is displayed. ■Signal detection method (Detection) The fetch result of the parameter is displayed. ■Compensation time (CompensationTime) "0.0" is always displayed. ■Filter time (FilterTime) The fetch result of the parameter is displayed.

Variable name	Name	Import	Data type	Attribute	Description
Homing_Complete	Home position return completed	Immediately	BOOL	LIST_READ_ONLY	This variable displays whether homing completed normally or not. This variable becomes FALSE when operation starts or when homing is required. • FALSE: Homing uncompleted • TRUE: Homing completed
Homing_Request	Home position return request	Immediately	BOOL	LIST_READ_ONLY	This variable displays whether homing is required or not. This variable becomes FALSE when homing is completed. • FALSE: No homing request • TRUE: Homing request present
Homing_Required	Home position return required or not	Ready ON	BOOL	LIST_READ_ONLY	This variable displays whether homing is required for the axis or not. If set to FALSE, the homing request will not become TRUE at the current position restoration. • FALSE: Homing is unnecessary • TRUE: Homing is necessary
Homing_Status	Home position return operation status	Operation cycle	WORD(HEX)	LIST_READ_ONLY	This variable displays the homing status of the driver. • 0000H: Homing procedure in progress • 0001H: Homing procedure interrupted or not started • 0002H: Homing attained (target not reached) • 0003H: Homing procedure completed successfully • 0004H: Homing error occurred (velocity other than 0) • 0005H: Homing error occurred (velocity is 0) • FFFFH: Homing procedure is not in progress
HwStrokeLimit_FlsStatus	Upper limit signal status	Operation cycle	BOOL	LIST_READ_ONLY	This variable displays the detection status of the upper limit signal (FLS). • FALSE: Detecting the upper limit signal (FLS) • TRUE: Not detecting the upper limit signal (FLS) *: Displays the detection status which reflects the signal detection method or the filter time but not the signal input state.
HwStrokeLimit_Override	Hardware stroke limit override	At start	STRING(15)	LIST_READ_ONLY	Displays the acceptance status of the enabled/disabled switching command of the hardware stroke limit. • DISABLE: Check disabled • ONLY_INSIDE: Disables the check only for movement toward the setting range. • Other than the above: No disabling request
HwStrokeLimit_RlsStatus	Lower limit signal status	Operation cycle	BOOL	LIST_READ_ONLY	This variable displays the detection status of the lower limit signal (RLS). • FALSE: Detecting the lower limit signal (RLS) • TRUE: Not detecting the lower limit signal (RLS) *: Displays the detection state which reflects the signal detection method or the filter time but not the signal input state.
InVelocity	Target speed reached	Operation cycle	BOOL	LIST_READ_ONLY	This variable displays whether the target velocity has been reached or not. During axes group operation, stores FALSE in Target speed reached (AxisName.Md.InVelocity) of the structuring axis. • FALSE: Not reached • TRUE: Reached
Io_PosActualValue	Object data_PosActualValue	Operation cycle	DINT	LIST_READ_ONLY	This variable displays the value of object data PosActualValue (the position feedback received from the slave device). For a real encoder axis or virtual encoder axis, displays the input pulse value [encoder pulse unit] acquired from the encoder.

Variable name	Name	Import	Data type	Attribute	Description
Io_PosEncoderResolution	Object data_PosEncoderResolution	Operation cycle	DWORD(UDINT)	LIST_READ_ONLY	This variable displays the value of object data PosEncoderResolution. For a real encoder axis or virtual encoder axis, displays the synchronous encoder resolution.
Io_Statusword	Object data_Statusword	Operation cycle	WORD(HEX)	LIST_READ_ONLY	This variable displays the value of object data Statusword. For a real encoder axis, the encoder status is emulated and displayed on the Motion module side.
Io_TargetPos	Object data_TargetPos	Operation cycle	DINT	LIST_READ_ONLY	This variable displays the value of object data TargetPos (the position command to be sent to the slave device). For a virtual drive axis, displays a value which is converted into the command value in a driver unit from the cumulative current position.
Io_TargetVelocity	Object data_TargetVelocity	Operation cycle	DINT	LIST_READ_ONLY	This variable displays the value of object data TargetVelocity (the velocity command to be sent to the slave device).
Io_TorqueActualValue	Object data_TorqueActualValue	Operation cycle	INT	LIST_READ_ONLY	This variable displays the value of object data TorqueActualValue (the torque feedback received from the slave device).
Io_VelActualValue	Object data_VelActualValue	Operation cycle	DINT	LIST_READ_ONLY	This variable displays the value of object data VelActualValue (the velocity feedback received from the slave device).
JerkLimit	Jerk limit value	Ready ON	LREAL	LIST_READ_ONLY	This variable stores the jerk limit value.
OperationCycle	Control cycle	At system start	INT	LIST_READ_ONLY	This variable stores the value of the control operation cycle.
OverrunOperation	Operation setting at overrun	Ready ON	INT (MC_DECEL_STOP_MODE) ☞ Page 79 MC_DECEL_STOP_MODE	LIST_READ_ONLY	This variable displays the input status of the overrun operation setting. • 1: Immediately stop (ImmediateStop) • 2: Keep current acceleration/deceleration (KeepCurrentAcc)
PosRestoration_Status	Current position restoration status	Service	INT (MC_POS_RESTORATION_STATUS) ☞ Page 81 MC_POS_RESTORATION_STATUS	LIST_READ_ONLY	This variable displays the current position restoration status. • 0: Not executed (NotExecute) • 1: Waiting restoration request (WaitingRequest) • 2: Restoration completed by incremental system (RestoredInIncSystem) • 3: Restored in absolute position system (Home position return uncompleted) (RestoredInAbsSystemUnHomed) • 4: Restored in absolute position system (RestoredInAbsSystem) *: 0: Not executed (NotExecute) is displayed when the slave device is disconnected.
ProfileID	Execution profile ID No.	Operation cycle	WORD(UINT)	LIST_READ_ONLY	This variable stores the profile ID currently being executed.
SetAcceleration	Set current acceleration	Operation cycle	LREAL	LIST_READ_ONLY	This variable stores the set current acceleration (the acceleration calculated from the difference of the set velocity).*1 (Unit: U/s^2) The sign indicates whether acceleration or deceleration is in progress. "0.0" is stored when the target velocity is reached. During axes group operation, Set current acceleration (SetAcceleration) of the structuring axis is stored. • 0.0: During stop or fixed-speed • Plus sign: During acceleration • Minus sign: During deceleration

Variable name	Name	Import	Data type	Attribute	Description
SetPosition	Set position	Operation cycle	LREAL	LIST_READ_ONLY	This variable stores the current position address.[*1] The set position is a value equal to the cumulative current value rounded with the ring counter range.
SetVelocity	Set velocity	Operation cycle	LREAL	LIST_READ_ONLY	This variable stores the set output velocity.[*1] This velocity is calculated from the difference of the set position. The value is positive when moving to the positive direction (Address increase direction), and the value is negative when moving to the negative direction (Address decrease direction). During axes group operation, the set velocity of the structuring axis is stored.
SlaveEmulate_Enable	Slave emulating	Service	BOOL	LIST_READ_ONLY	Displays whether driver emulate operation is valid or invalid. • FALSE: Invalid • TRUE: Valid
StartableAtUnhomed	Start permission at home position return uncompleted	Ready ON	BOOL	LIST_READ_ONLY	This variable displays the start status when homing is incomplete. • FALSE: Not allowed • TRUE: Allowed
StopMode_Deceleration	Deceleration at stop	Ready ON	LREAL	LIST_READ_ONLY	This variable displays the input status of deceleration at stop.
StopMode_DecelerationCurve	Stop selection at deceleration to stop	Ready ON	INT (MC_STOP_CURVE_MODE) ☞ Page 79 MC_STOP_CURVE_MODE	LIST_READ_ONLY	This variable displays the input status of stop selection at deceleration to stop. • 1: Recreate deceleration curve (OverrideCurve)
StopMode_General	Stop selection at stop by factors	Ready ON	INT (MC_DECEL_STOP_MODE) ☞ Page 79 MC_DECEL_STOP_MODE	LIST_READ_ONLY	This variable displays the input status of stop selection at stop by factors. • 1: Immediate stop (ImmediateStop) • 2: Keep current acceleration/deceleration (KeepCurrentAcc) • 3: Alternative acceleration/deceleration (AlternativeAcc)
StopMode_HwStrokeLimit	Stop selection at hardware stroke limit error occurrence	Ready ON	INT (MC_DECEL_STOP_MODE) ☞ Page 79 MC_DECEL_STOP_MODE	LIST_READ_ONLY	This variable displays the input status of stop selection at the hardware stroke limit error occurrence. • 1: Immediate stop (ImmediateStop) • 2: Keep current acceleration/deceleration (KeepCurrentAcc) • 3: Alternative acceleration/deceleration (AlternativeAcc)
StopMode_ServoOff	Process selection at servo OFF command during operation	Ready ON	INT (MC_DECEL_STOP_MODE) ☞ Page 79 MC_DECEL_STOP_MODE	LIST_READ_ONLY	This variable displays the input status of process selection at servo OFF command during operation. • 0: Ignore (Ignore) • 4: Servo OFF after immediate stop (ServoOffAfterImmediateStop) • 5: Servo OFF after deceleration to stop (ServoOffAfterDecelStop)
StopMode_SwStrokeLimit	Stop selection at software stroke limit error occurrence	Ready ON	INT (MC_DECEL_STOP_MODE) ☞ Page 79 MC_DECEL_STOP_MODE	LIST_READ_ONLY	This variable displays the input status of stop selection at software stroke limit error occurrence • 1: Immediate stop (ImmediateStop) • 2: Keep current acceleration/deceleration (KeepCurrentAcc) • 3: Alternative acceleration/deceleration (AlternativeAcc)
StopOption_DriverTargetIgnored	Driver command discard detection setting	Ready ON	BOOL	LIST_READ_ONLY	This variable displays the status of driver command discard detection setting. • FALSE: Detection disabled • TRUE: Detection enabled

Variable name	Name	Import	Data type	Attribute	Description
StopSignal	Stop signal	Ready ON	SIGNAL_SELECT ☞ Page 66 SIGNAL_SELECT (Signal select)	LIST_READ_ONLY	This variable sets the signal to use the stop signal (STOP) of the external input signal. Settings and operations specific to Stop signal (StopSignal) are shown below. ■IO No. (StartIO) "0" is always displayed. ■Target (Target) The fetch result of the parameter is displayed. ■Signal detection method (Detection) The fetch result of the parameter is displayed. ■Compensation time (CompensationTime) "0.0" is always displayed. ■Filter time (FilterTime) The fetch result of the parameter is displayed.
StopStatus	Stop status	Operation cycle	BOOL	LIST_READ_ONLY	This variable displays the input status of the stop signal (STOP) of the external input signal. • FALSE: Stop processing released • TRUE: During stop processing
SwStrokeLimit_Lower	Software stroke limit lower value	Ready ON	LREAL	LIST_READ_ONLY	This variable displays the software stroke limit lower value.
SwStrokeLimit_Override	Software stroke limit override	At start	STRING(15)	LIST_READ_ONLY	This variable displays the software stroke limit valid/invalid status. • DISABLE: Check disabled • ONLY_INSIDE: Disables the check only for movement toward the setting range. • Other than the above: Check enabled (no disable request)
SwStrokeLimit_Target	Software stroke limit target	Ready ON	INT (MC_POS_SOURCE) ☞ Page 79 MC_POS_SOURCE	LIST_READ_ONLY	This variable displays the software stroke limit target. • -1: Disabled (Invalid) • 1: Set position (SetPosition) • 3: Feed machine position (FeedMachinePosition)
SwStrokeLimit_Upper	Software stroke limit upper value	Ready ON	LREAL	LIST_READ_ONLY	This variable displays the software stroke limit upper value.
TargetAcceleration	Target acceleration	Operation cycle	LREAL	LIST_READ_ONLY	This variable stores the maximum acceleration used during acceleration. (Unit: U/s^2) During axes group operation, stores "0.0" in Target acceleration (AxisName.Md.TargetAcceleration) of the structuring axis.
TargetDeceleration	Target deceleration	Operation cycle	LREAL	LIST_READ_ONLY	This variable stores the maximum deceleration used during acceleration. (Unit: U/s^2) During axes group operation, stores "0.0" in Target deceleration (AxisName.Md.TargetDeceleration) of the structuring axis.
TargetVelocity	Target velocity	Operation cycle	LREAL	LIST_READ_ONLY	This variable stores the actual target velocity whose override and velocity limit value are considered. ■For positioning control The value becomes more than "0.0" regardless of the movement direction. ■For other than positioning control The value is positive when moving to the positive direction (Address increase direction), and the value is negative when moving to the negative direction (Address decrease direction). Stores "0.0" after the control is completed. During axes group operation, stores "0.0" in Target velocity (AxisName.Md.TargetVelocity) of the structuring axis.
TorqueLimit_Negative	Negative direction torque limit value	Operation cycle	LREAL	LIST_READ_ONLY	This variable displays the negative direction torque limit value that is currently valid.
TorqueLimit_Positive	Positive direction torque limit value	Operation cycle	LREAL	LIST_READ_ONLY	This variable displays the positive direction torque limit value that is currently valid.

Variable name	Name	Import	Data type	Attribute	Description
Unit_Position	Position command unit	Ready ON	DWORD(HEX)	LIST_READ_ONLY	This variable displays the position command unit to be used in the Motion control. • 00000000H: pulse • 00010000H: m • FD010000H: mm • FA010000H: μm • F7010000H: nm • 00410000H: degree • FA410000H: $\times 10^{-6}$degree • 00B40000H: Revolution • 00C00000H: inch • 00FF0000H: arbitrary unit
Unit_PositionDisplay	Position command unit display	Ready ON	WSTRING(31)	LIST_READ_ONLY	This variable displays the currently controlled position command unit as a character string.
Unit_PositionString	Position command unit string	Ready ON	WSTRING(31)	LIST_READ_ONLY	When Position command unit (AxisName.Pr.Unit_Position) is set to "00FF0000H: arbitrary unit", this variable displays the currently controlled position command unit set in Position command unit string (AxisName.Pr.Unit_PositionString) as a character string.
Unit_Velocity	Velocity command unit	Ready ON	DWORD(HEX)	LIST_READ_ONLY	This variable displays the velocity command unit to be used in the Motion control. • 00000300H: U/s • 00004700H: U/min • FD000300H: U/ms • FA000300H: U/μs • F7000300H: U/ns
Unit_VelocityDisplay	Velocity command unit display	Ready ON	WSTRING(31)	LIST_READ_ONLY	This variable outputs the currently controlled velocity command unit as a character string.
UseInGroup	Using axes group	Operation cycle	BOOL	LIST_READ_ONLY	This variable displays whether the specified axis is being used in the axes group or not. • FALSE: Not used • TRUE: Used
VelocityLimit_Negative	Negative direction speed limit value	Ready ON	LREAL	LIST_READ_ONLY	This variable stores the address decreasing direction speed limit value.
VelocityLimit_OverOperation	Operation setting at speed limit value exceeded	Ready ON	INT (MC_VELOCITY_LIMIT_MODE) ☞ Page 77 MC_VELOCITY_LIMIT_MODE	LIST_READ_ONLY	This variable displays the operation when the speed limit value is exceeded during control. • 0: Ignore (Ignore) • 3: Immediate stop (ImmediateStop)
VelocityLimit_Positive	Positive direction speed limit value	Ready ON	LREAL	LIST_READ_ONLY	This variable stores the address increasing direction speed limit value.
VelocityOverride	Speed override coefficient	Operation cycle	LREAL	LIST_READ_ONLY	This variable displays the velocity override factor.
Warning	Axis warning detection	Immediately	BOOL	LIST_READ_ONLY	This variable displays the axis warning status. • FALSE: No axis warning • TRUE: Axis warning present
WarningID	Axis warning code	Immediately	WORD(HEX)	LIST_READ_ONLY	This variable stores the warning code when an axis warning occurs.

*1 The stored value includes an error because a floating-point error occurs.

AxisName.Cd. (Axis control data)

Variable name	Name	Import	Type	Attribute	Range
AccelerationOverride	Acceleration override coefficient	Operation cycle	LREAL	LIST_READ_ONLY	This variable sets the acceleration override factor. • 0.01 to 10.00
Encoder_Connect	Connection command	Operation cycle	BOOL	LIST_READ_ONLY	This variable switches the connection status of the virtual encoder axis input between valid/invalid. • FALSE: Invalid • TRUE: Valid
Encoder_CounterDisable	Counter disabled	Operation cycle	BOOL	LIST_READ_ONLY	This variable switches to the counter disable. • FALSE: Not executed • TRUE: Executes the counter disable request
Encoder_InputValue	Encoder input value	Operation cycle	DINT	LIST_READ_ONLY	This variable sequentially sets the values used as the virtual encoder axis input values.
ErrorReset	Axis error reset	Service	BOOL	LIST_READ_ONLY	This variable resets axis errors/warnings and drive unit errors. This variable becomes FALSE when the error reset is completed. • FALSE: Not executed • TRUE: Executes the error reset
FollowupDisable	Follow-up disabled	Operation cycle	BOOL	LIST_READ_ONLY	This variable disables the follow-up. • FALSE: Not executed • TRUE: Executes the counter disable request
Homing_ClearRequest	Home position return request clear	Operation cycle	BOOL	LIST_READ_ONLY	This variable clears the homing request. Home position return request clear (Homing_ClearRequest) becomes FALSE automatically after Home position return request (Homing_Request) is set to FALSE. • FALSE: Not executed • TRUE: Executes the homing request clear
HwStrokeLimit_Override	Hardware stroke limit override	At start	STRING(15)	LIST_READ_ONLY	This variable temporarily switches the hardware stroke limit check between valid/invalid. • DISABLE: Check disabled • ONLY_INSIDE: Disables the check only for movement toward the setting range. • Other than the above: No disabling request
SwStrokeLimit_Override	Software stroke limit override	At start	STRING(15)	LIST_READ_ONLY	This variable temporarily switches the software stroke limit check between valid/invalid. • DISABLE: Check disabled • ONLY_INSIDE: Disables the check only for movement toward the setting range. • Other than the above: Check enabled (no disabling request)
TorqueLimit_Negative	Negative direction torque limit value	Operation cycle	LREAL	LIST_READ_ONLY	This variable sets the negative direction torque limit value. • 0.0 to 1000.0[%]
TorqueLimit_Positive	Positive direction torque limit value	Operation cycle	LREAL	LIST_READ_ONLY	This variable sets the positive direction torque limit value. • 0.0 to 1000.0[%]
VelocityOverride	Speed override coefficient	Operation cycle	LREAL	LIST_READ_ONLY	This variable sets the velocity override factor. • 0.00 to 10.00

Variables settable as the axis variable for each axis type

The variables that can be set and used as the axis variable differ for each axis type.

The variables that can be set for each axis type are shown below.

■AxisName.AxisRef. (Axis information)

○: Settable, ×: Not settable

Variable name	Name	Axis type				
		Real drive axis	Real encoder axis	Virtual drive axis	Virtual encoder axis	Virtual link axis
AxisNo	Axis No.	○	○	○	○	○
StartIO	IO No.	○	○	○	○	○

■AxisName.PrConst. (Axis parameter constant)

○: Settable, ×: Not settable

Variable name	Name	Axis type				
		Real drive axis	Real encoder axis	Virtual drive axis	Virtual encoder axis	Virtual link axis
AddressOfStation	Station address setting	○	○	×	×	×
AxisType	Axis type setting	○	○	○	○	○
Encoder_AxisType	Real encoder axis type setting	×	○	×	×	×
Encoder_CounterDisableSignal	Counter disabling signal	×	○	×	○	×
Encoder_RingCout_LowerValue	Encoder ring counter lower limit value	×	×	×	○	×
Encoder_RingCout_UpperValue	Encoder ring counter upper limit value	×	×	×	○	×
HwStrokeLimit_FlsSignal	Upper limit signal	○	×	○	×	×
HwStrokeLimit_RlsSignal	Lower limit signal	○	×	○	×	×
OperationCycle	Control cycle setting	○	○	○	○	○
PositioningRange	Positioning range	○	○	○	○	○
PosRestoration_AbsPosBase	Absolute position reference setting	○	×	×	×	×
PosRestoration_AbsPosEnable	Absolute position management setting	○	○	○	○	○
RingCount_Enable	Ring counter enabled selection	○	○	○	○	○
RingCount_LowerValue	Ring counter lower limit value	○	○	○	○	○
RingCount_UpperValue	Ring counter upper limit value	○	○	○	○	○
SlaveEmulate_Enable	Slave emulation enabled	○	×	×	×	×
SlaveObject	Slave object data	○	○	×	○	×
TorqueLimit_Max	Torque limit maximum value	○	×	×	×	×
TorqueLimit_NegativeInitial	Negative direction torque limit initial value	○	×	×	×	×
TorqueLimit_PositiveInitial	Positive direction torque limit initial value	○	×	×	×	×
FastOperationMode	High-speed mode setting	○	○	○	○	○

■AxisName.Pr. (Axis parameter)

○: Settable, ×: Not settable

Variable name	Name	Axis type				
		Real drive axis	Real encoder axis	Virtual drive axis	Virtual encoder axis	Virtual link axis
AccelerationLimit	Acceleration limit value	○	×	○	×	×
AccelerationZeroBehavior	Operation selection at start acceleration/ deceleration 0	○	×	○	×	×
CmdInPos_Width	Command in-position width	○	×	○	×	×
DecelerationLimit	Deceleration limit value	○	×	○	×	×
Drive_UnitConvRatioNum	Driver unit conversion numerator	○	○	×	○	×
Drive_UnitConvRatioDen	Driver unit conversion denominator	○	○	×	○	×
ForcedStop_Signal	Forced stop signal	○	×	○	×	×
Homing_Required	Home position return required or not	○	○	○	○	○
JerkLimit	Jerk limit value	○	×	○	×	×
OverrunOperation	Operation setting at overrun	○	×	○	×	×
StartableAtUnhomed	Start permission at home position return uncompleted	○	○	○	○	○
StopMode_Deceleration	Deceleration at stop	○	×	○	×	×
StopMode_DecelerationCurve	Stop selection at deceleration to stop	○	×	○	×	×
StopMode_General	Stop selection at stop by factors	○	×	○	×	×
StopMode_HwStrokeLimit	Stop selection at hardware stroke limit error occurrence	○	×	○	×	×
StopMode_ServoOff	Process selection at servo OFF command during operation	○	×	○	×	×
StopMode_SwStrokeLimit	Stop selection at software stroke limit error occurrence	○	×	○	×	×
StopOption_DriverTargetIgnored	Driver command discard detection setting	○	×	×	×	×
StopSignal	Stop signal	○	×	○	×	×
SwStrokeLimit_Lower	Software stroke limit lower value	○	×	○	×	×
SwStrokeLimit_Target	Software stroke limit target	○	×	○	×	×
SwStrokeLimit_Upper	Software stroke limit upper value	○	×	○	×	×
Unit_Position	Position command unit	○	○	○	○	○
Unit_PositionString	Position command unit string	○	○	○	○	○
Unit_Velocity	Speed command unit	○	○	○	○	○
VelocityLimit_Negative	Negative direction speed limit value	○	×	○	×	×
VelocityLimit_OverOperation	Operation setting at speed limit value exceeded	○	×	○	×	×
VelocityLimit_Positive	Position direction speed limit value	○	×	○	×	×

■AxisName.Md. (Axis monitor data)

○: Settable, ×: Not settable

Variable name	Name	Axis type				
		Real drive axis	Real encoder axis	Virtual drive axis	Virtual encoder axis	Virtual link axis
AccelerationLimit	Acceleration limit value	○	×	○	×	×
AccelerationOverride	Acceleration override coefficient	○	×	○	×	×
AccelerationZeroBehavior	Operation selection at start acceleration/ deceleration 0	○	×	○	×	×
ActualPosition	Feedback position	○	○	×	×	×
ActualVelocity	Feedback speed	○	○	×	×	×
Analyzing	Analyzing	○	○	○	○	○
AutoDeceleration	Automatically decelerating	○	×	○	×	×
AxisName	Axis name	○	○	○	○	○
AxisStatus	Axis status	○	○	○	○	○
BufferingFBs	Number of buffering FBs	○	×	○	×	×
CmdInPos	Command in-position	○	×	○	×	×
CmdInPos_Width	Command in-position width	○	×	○	×	×
CommandedAcceleration	Specified acceleration	○	×	○	×	×
CommandedDeceleration	Specifed deceleration	○	×	○	×	×
CommandedJerk	Specified jerk	○	×	○	×	×
CommandedPosition	Specified position	○	×	○	×	×
CommandedVelocity	Specified speed	○	×	○	×	×
Cst_SetTorque	Command current torque at torque command	○	×	×	×	×
Cst_TargetTorque	Target torque at torque control	○	×	×	×	×
CumulativePosition	Cumulative current position	○	○	○	○	○
DecelerationLimit	Deceleration limit value	○	×	○	×	×
Drive_RPDO[1..64]	RPDO (Slave device → Motion system) mapping	○	×	×	×	×
Drive_TPDO[1..64]	TPDO (Motion system → Slave device) mapping	○	×	×	×	×
Drive_UnitConvRatioNum	Driver unit conversion numerator	○	○	×	×	×
Drive_UnitConvRatioDen	Driver unit conversion denominator	○	○	×	×	×
Driver_Mode	Driver control mode	○	×	×	×	×
Driver_ReadyOn	Driver ready ON status	○	×	×	×	×
Driver_ServoOn	Driver servo ON status	○	×	×	×	×
Driver_State	Driver status	○	×	×	×	×
DriverError	Drive module error detection	○	×	×	×	×
DriverErrorID	Drive module error code	○	×	×	×	×
DriverErrorDetailID	Drive module error detail code	○	×	×	×	×
Encoder_Connected	Connection status	×	×	×	○	×
Encoder_CounterDisable	Counter disabled	×	○	×	○	×
Error	Axis error detection	○	○	○	○	○
ErrorID	Axis error code	○	○	○	○	○
FeedMachinePosition	Feed machine position	○	×	×	×	×
FollowupDisable	Follow-up disabled	○	×	×	×	×
ForcedStop_Released	Forced stop cancelling	○	×	○	×	×
ForcedStop_Signal	Forced stop signal	○	×	○	×	×
Homing_Complete	Home position return completed	○	○	○	○	○
Homing_Request	Home position return request	○	○	○	○	○
Homing_Required	Home position return required or not	○	○	○	○	○
Homing_Status	Home position return operation status	○	×	×	×	×
HwStrokeLimit_FlsStatus	Upper limit signal status	○	×	○	×	×

Variable name	Name	Axis type				
		Real drive axis	Real encoder axis	Virtual drive axis	Virtual encoder axis	Virtual link axis
HwStrokeLimit_Override	Hardware stroke limit override	○	×	○	×	×
HwStrokeLimit_RlsStatus	Lower limit signal status	○	×	○	×	×
InVelocity	Target speed reached	○	×	○	×	×
Io_PosActualValue	Object data_PosActualValue	○	○	×	○	×
Io_PosEncoderResolution	Object data_PosEncoderResolution	×	○	×	○	×
Io_Statusword	Object data_Statusword	○	○	×	×	×
Io_TargetPos	Object data_TargetPos	○	×	○	×	×
Io_TargetVelocity	Object data_TargetVelocity	○	×	×	×	×
Io_TorqueActualValue	Object data_TorqueActualValue	○	×	×	×	×
Io_VelActualValue	Object data_VelActualValue	○	×	×	×	×
JerkLimit	Jerk limit value	○	×	○	×	×
OperationCycle	Control cycle	○	○	○	○	○
OverrunOperation	Operation setting at overrun	○	×	○	×	×
PosRestoration_Status	Current position restoration status	○	○	○	○	○
ProfileID	Execution profile ID No.	○	×	○	×	○
SetAcceleration	Command current acceleration	○	×	○	×	×
SetPosition	Command current position	○	○	○	○	○
SetVelocity	Command current speed	○	×	○	○	×
SlaveEmulate_Enable	Slave emulating	○	×	×	×	×
StartableAtUnhomed	Start permission at home position return uncompleted	○	○	○	○	○
StopMode_Deceleration	Deceleration at stop	○	×	○	×	×
StopMode_DecelerationCurve	Stop selection at deceleration to stop	○	×	○	×	×
StopMode_General	Stop selection at stop by factors	○	×	○	×	×
StopMode_HwStrokeLimit	Stop selection at hardware stroke limit error occurrence	○	×	○	×	×
StopMode_ServoOff	Process selection at servo OFF command during operation	○	×	○	×	×
StopMode_SwStrokeLimit	Stop selection at software stroke limit error occurrence	○	×	○	×	×
StopOption_DriverTargetIgnored	Driver command discard detection setting	○	×	×	×	×
StopSignal	Stop signal	○	×	○	×	×
StopStatus	Stop status	○	×	○	×	×
SwStrokeLimit_Lower	Software stroke limit lower value	○	×	○	×	×
SwStrokeLimit_Override	Software stroke limit override	○	×	○	×	×
SwStrokeLimit_Target	Software stroke limit target	○	×	○	×	×
SwStrokeLimit_Upper	Software stroke limit upper value	○	×	○	×	×
Sync_PhaseShift	Phase compensation amount	○	×	○	×	○
TargetAcceleration	Target acceleration	○	×	○	×	×
TargetDeceleration	Target deceleration	○	×	○	×	×
TargetVelocity	Target speed	○	×	○	×	×
TorqueLimit_Negative	Negative direction torque limit value	○	×	×	×	×
TorqueLimit_Positive	Positive direction torque limit value	○	×	×	×	×
Unit_Position	Position command unit	○	○	○	○	○
Unit_PositionDisplay	Position command unit display	○	○	○	○	○
Unit_PositionString	Position command unit string	○	○	○	○	○
Unit_Velocity	Speed command unit	○	○	○	○	○
Unit_VelocityDisplay	Speed command unit display	○	○	○	○	○
UseInGroup	Using axes group	○	○	○	○	○
VelocityLimit_Negative	Negative direction speed limit value	○	×	○	×	×

Variable name	Name	Axis type				
		Real drive axis	Real encoder axis	Virtual drive axis	Virtual encoder axis	Virtual link axis
VelocityLimit_OverOperation	Operation setting at speed limit value exceeded	○	×	○	×	×
VelocityLimit_Positive	Positive direction speed limit value	○	×	○	×	×
VelocityOverride	Speed override coefficient	○	×	○	×	×
Warning	Axis warning detection	○	○	○	○	○
WarningID	Axis warning code	○	○	○	○	○

■AxisName.Cd. (Axis control data)

○: Settable, ×: Not settable

Variable name	Name	Axis type				
		Real drive axis	Real encoder axis	Virtual drive axis	Virtual encoder axis	Virtual link axis
AccelerationOverride	Acceleration override coefficient	○	×	○	×	×
Encoder_Connect	Connection command	×	×	×	○	×
Encoder_CounterDisable	Counter disabled	×	○	×	○	×
Encoder_InputValue	Encoder input value	×	×	×	○	×
ErrorReset	Axis error reset	○	○	○	○	○
FollowupDisable	Follow-up disabled	○	×	×	×	×
Homing_ClearRequest	Home position return request clear	○	○	○	○	○
HwStrokeLimit_Override	Hardware stroke limit override	○	×	○	×	×
SwStrokeLimit_Override	Software stroke limit override	○	×	○	×	×
TorqueLimit_Negative	Negative direction torque limit value	○	×	×	×	×
TorqueLimit_Positive	Positive direction torque limit value	○	×	×	×	×
VelocityOverride	Speed override coefficient	○	×	○	×	×

Axes group variables

The data type of axes group variables is expressed as AXES_GROUP.

Each axes group data type has the following members.

Member name	Data type	Description	Reference
AxesGroupRef	AXES_GROUP_REF	The data structure for input/output of the Motion control FB.	☞ Page 46 AxesGroupName.AxesGroupRef. (Axes group information)
Pr	AXES_GROUP_PRM	Stores the parameter data for the axes group. Opens the default value at axes group variable generation. Executes reimporting to the control even after axes group initialization. The timing to import to the control changes depending on the parameter.	☞ Page 46 AxesGroupName.Pr. (Axes group parameter)
Md	AXES_GROUP_MONI	Stores the monitor data for the axes group. Executes refresh in the fixed cycle for each monitor data.	☞ Page 48 AxesGroupName.Md. (Axes group monitor data)
Cd	AXES_GROUP_CMD	Stores the command data for axes group control. Gets the newest value every control operation cycle and uses it for control.	☞ Page 52 AxesGroupName.Cd. (Axes group control data)

AxesGroupName.AxesGroupRef. (Axes group information)

Variable name	Name	Import	Data type	Attribute	Description
GroupNo	Axes group No.	—	WORD(UINT)	LIST_WRITE_BACK	This variable sets the axes group No. • 0: Not set • 1 to 10000: Setting axes group No.
StartIO	IO No.	—	WORD(HEX)	LIST_WRITE_BACK	This variable sets the I/O No.

AxesGroupName.Pr. (Axes group parameter)

Variable name	Name	Import	Data type	Attribute	Description
AccelerationLimit	Acceleration limit value	Ready ON	LREAL	LIST_WRITE_BACK	This variable sets the acceleration limit value. • 0.0000, 0.0001 to 2147483647.0 *: When the acceleration/deceleration method is set to "1: Acceleration/deceleration time-fixed method (mcFixedTime)", limiting of the acceleration calculated from the specified acceleration/deceleration time is performed. *: When set to "0.0000", limiting of the acceleration is not performed. *: When a real number less than 0.0001 is set, the value is imported as "0.0000".
AccelerationZeroBehavior	Operation selection at start acceleration/deceleration 0	Ready ON	INT (MC_ACC_ZERO_MODE) ☞ Page 79 MC_ACC_ZERO_MODE	LIST_WRITE_BACK	This variable sets the behavior when "0.0" is set as the acceleration, the deceleration, or the acceleration/deceleration time at start. • -1: Error (Not started) (ACCError) • 1: Maximum acceleration/deceleration (MaximumAcceleration)
Axis[1..16]	Structuring axis	Ready ON	AXIS_REF ☞ Page 23 AxisName.AxisRef. (Axis information)	LIST_WRITE_BACK	This variable sets the Axis No. (AxisNo) of Axis information (AxisName.AxisRef) that configures the axes group.
CmdInPos_Width	Command in-position width	Ready ON	LREAL	LIST_WRITE_BACK	This variable sets the command in-position width on the composite axis of the axes group. • 0.0: Function disabled • 0.000000001 to 10000000000.0: Function enabled *: When a real number less than 0.000000001 is set, the value is imported as "0.0".

Variable name	Name	Import	Data type	Attribute	Description
DecelerationLimit	Deceleration limit value	Ready ON	LREAL	LIST_WRITE_BACK	This variable sets the deceleration limit value. • 0.0000, 0.0001 to 2147483647.0 *: When the acceleration/deceleration method is set to "1: Acceleration/deceleration time-fixed method (mcFixedTime)", limiting of the deceleration calculated from the specified acceleration/deceleration time is performed. *: When set to "0.0000", limiting of the deceleration is not performed. *: When a real number less than 0.0001 is set, the value is imported as "0.0000".
JerkLimit	Jerk limit value	Ready ON	LREAL	LIST_WRITE_BACK	This variable sets the jerk limit value. • 0.0000, 0.0001 to 2147483647.0 *: When set to "0.0000", limiting of the jerk is not performed. *: When a real number less than 0.0001 is set, the value is imported as "0.0000".
OverrunOperation	Operation setting at overrun	Ready ON	INT (MC_DECEL_STOP_MODE) ☞ Page 79 MC_DECEL_STOP_MODE	LIST_WRITE_BACK	This variable sets the operation performed when the target position is exceeded during operation. • 1: Immediately stop (ImmediateStop)
StopMode_Deceleration	Deceleration at stop	Ready ON	LREAL	LIST_WRITE_BACK	This variable sets the deceleration/deceleration time at deceleration stop by stop cause occurrence. ■When using acceleration/deceleration method specifying acceleration/deceleration (acceleration/deceleration specification method) • 0.0000, 0.0001 to 2147483647.0 ■When using acceleration/deceleration method specifying acceleration/deceleration time (acceleration/deceleration time-fixed method) • 0.000000, 0.000001 to 8400.0 *: If this variable is set to "0", immediately stops regardless of the setting of Operation selection at start acceleration/deceleration 0 (AccelerationZeroBehavior) *: Clamps by the upper limit value when the upper limit value is the range at stop by factors, and is treated as "0 (Immediate stop)" when the lower limit value is the range.
StopMode_DecelerationCurve	Stop selection at deceleration to stop	Ready ON	INT (MC_STOP_CURVE_MODE) ☞ Page 79 MC_STOP_CURVE_MODE	LIST_WRITE_BACK	This variable sets the operation performed when a stop cause occurs during deceleration (including a stop cause and automatic deceleration). • 1: Recreate deceleration curve (OverrideCurve)
StopMode_ErrorInGroup	Structuring axis operation selection at axis stop by factors	Ready ON	INT (MC_DECEL_STOP_MODE) ☞ Page 79 MC_DECEL_STOP_MODE	LIST_WRITE_BACK	During axes group operation, when the driver servo is turned OFF on the structuring axis and an immediate stop axis error occurs, this variable sets the operation of the axis on which an error has not occurred. • 1: Immediate stop (ImmediateStop)
StopMode_General	Stop selection at stop by factors	Ready ON	INT (MC_DECEL_STOP_MODE) ☞ Page 79 MC_DECEL_STOP_MODE	LIST_WRITE_BACK	This variable sets the operation performed when a stop cause occurs. • 1: Immediate stop (ImmediateStop) • 2: Keep current acceleration/deceleration (KeepCurrentAcc) • 3: Alternative acceleration/deceleration (AlternativeAcc) *: Immediate stop will be executed when executing an FB without deceleration specification.

Variable name	Name	Import	Data type	Attribute	Description
Unit_Position	Position command unit	When axes group is valid	DWORD(HEX)	LIST_WRITE_BACK	This variable sets the position command unit to be used in the Motion control. • 00000000H: pulse • 00010000H: m • FD010000H: mm • FA010000H: μm • F7010000H: nm • 00410000H: degree • FA410000H: $\times 10^{-6}$degree • 00B40000H: Revolution • 00C00000H: inch • 00FF0000H: arbitrary unit
Unit_PositionString	Position command unit string	When axes group is valid	WSTRING(31)	LIST_WRITE_BACK	This variable sets the command unit to be used in the Motion control as a character string. This variable is set when Position command unit (Unit_Position) is set to "00FF0000H: arbitrary unit".
Unit_Velocity	Velocity command unit	When axes group is valid	DWORD(HEX)	LIST_WRITE_BACK	This variable sets the velocity command unit to be used in the Motion control. • 00000300H: U/s • 00004700H: U/min • FD000300H: U/ms • FA000300H: U/μs • F7000300H: U/ns
VelocityLimit	Speed limit value	Ready ON	LREAL	LIST_WRITE_BACK	This variable sets the speed limit value of the axes group. • 0.0001 to 2500000000.0

AxesGroupName.Md. (Axes group monitor data)

Variable name	Name	Import	Data type	Attribute	Description
AccelerationLimit	Acceleration limit value	Ready ON	LREAL	LIST_READ_ONLY	This variable stores the acceleration limit value.
AccelerationOverride	Acceleration override coefficient	Operation cycle	LREAL	LIST_READ_ONLY	This variable stores the acceleration override factor.
AccelerationZeroBehavior	Operation selection at start acceleration/ deceleration 0	Ready ON	INT (MC_ACC_ZERO_MODE) ☞ Page 79 MC_ACC_ZERO_MODE	LIST_READ_ONLY	This variable sets the behavior when "0.0" is set as the acceleration, the deceleration, or the acceleration/deceleration time at start. • -1: Error (Not started) (ACCError) • 1: Maximum acceleration/deceleration (MaximumAcceleration)
ActualVelocity	Feedback speed	Operation cycle (GroupEnable)	LREAL	LIST_READ_ONLY	This variable stores the actual velocity. This velocity is the vector speed of the actual velocity of the structuring axes. The value becomes more than "0.0" regardless of the movement direction of the structuring axes. Stores "0.0" when the axes group is invalid.
Analyzing	Analyzing	Operation cycle	BOOL	LIST_READ_ONLY	This variable displays the status of positioning processing of the axes group. • FALSE: Not analyzing • TRUE: Analyzing
AutoDeceleration	Automatically decelerating	Operation cycle	BOOL	LIST_READ_ONLY	This variable displays the status of automatic decelerating. TRUE is stored while the auto deceleration processing is being performed. When the multiple start is executed, this variable becomes TRUE while the auto deceleration processing is being performed during the last positioning point execution. This variable becomes FALSE when the control change is performed. • FALSE: Not during auto deceleration • TRUE: During auto deceleration

Variable name	Name	Import	Data type	Attribute	Description
Axis[1..16]	Structuring axis	Immediately	AXIS_REF ☞ Page 23 AxisName.AxisRef. (Axis information)	LIST_READ_ONLY	This variable sets the Axis No. (AxisNo) of Axis information (AxisName.AxisRef) that configures the axes group.
BufferingFBs	Number of buffering FBs	At start/at stop	INT	LIST_READ_ONLY	This variable displays the number of buffering FBs (0 to 2).
CmdInPos	Command in-position	Operation cycle	BOOL	LIST_READ_ONLY	This variable indicates whether the remaining distance to the target position is equal to or less than Command in-position width (AxesGroupName.Md.CmdInPos_Width). • FALSE: More than Command in-position width • TRUE: Less than Command in-position width
CmdInPos_Width	Command in-position width	Ready ON	LREAL	LIST_READ_ONLY	This variable displays the command in-position width on the composite axis of the axes group. The value displayed is the imported setting value of Command in-position width (AxesGroupName.Pr.CmdInPos_Width). When the value of Command in-position width (AxesGroupName.Md.CmdInPos_Width) is "0.0", the command in-position function of the axes group is disabled.
CommandedAcceleration	Specified acceleration	Operation cycle	LREAL	LIST_READ_ONLY	This variable stores the acceleration specified by the user. (Unit: U/s^2) When the acceleration/deceleration method is set to "1: Acceleration/deceleration time-fixed method (mcFixedTime)", this variable stores the specified acceleration/deceleration time. (Unit: s)
CommandedDeceleration	Specified deceleration	Operation cycle	LREAL	LIST_READ_ONLY	This variable stores the acceleration specified by the user. (Unit: U/s^2) When the acceleration/deceleration method is set to "1: Acceleration/deceleration time-fixed method (mcFixedTime)", it is an illegal value.
CommandedJerk	Specified jerk	Operation cycle	LREAL	LIST_READ_ONLY	This variable stores the jerk specified by the user. (Unit: U/s^3)
CommandedVelocity	Specified speed	Operation cycle (GroupEnable)	LREAL	LIST_READ_ONLY	This variable stores the specified speed imported to the operation Motion control FB being controlled. Stores "0.0" after the control is completed. *: When Speed mode (VelocityMode) is set to Reference axis speed (ReferenceAxisSpeed) or Long axis speed (LongAxisSpeed) for linear interpolation control, stores Specified speed of the applicable axis.
DecelerationLimit	Deceleration limit value	Ready ON	LREAL	LIST_READ_ONLY	This variable stores the deceleration limit value.
Error	Axes group error detection	Immediately	BOOL	LIST_READ_ONLY	This variable displays the axes group error status. • FALSE: No axes group error • TRUE: Axes group error present
ErrorID	Axes group error code	Immediately	WORD(HEX)	LIST_READ_ONLY	This variable displays the error code of the generated axes group error.
GroupName	Axes group name	Immediately	WSTRING(127)	LIST_READ_ONLY	This variable stores the axes group name.
GroupStatus	Axes group status	Operation cycle	INT (MC_AXES_GROUP_STATUS) ☞ Page 76 MC_AXES_GROUP_STATUS	LIST_READ_ONLY	This variable displays the axes group status. • -1: Axes group variable not initialized/Axes group parameter error (Invalid) • 0: Axes group disabled (GroupDisabled) • 1: Stopping on error (GroupErrorStop) • 2: Decelerating to stop (GroupStopping) • 4: Standby (GroupStandby) • 5: Operating (GroupMoving)

Variable name	Name	Import	Data type	Attribute	Description
InterpolationAxes	Interpolation axes	Operation cycle	DWORD(HEX)	LIST_READ_ONLY	This variable displays the structuring axis that is executing the interpolation control in the axes group as a bit.
InVelocity	Target speed reached	Operation cycle	BOOL	LIST_READ_ONLY	This variable displays whether the target velocity has been reached or not. • FALSE: Not reached • TRUE: Reached
JerkLimit	Jerk limit value	Ready ON	LREAL	LIST_READ_ONLY	This variable stores the jerk limit value.
NumberOfAxes	Number of structuring axes	Immediately	INT	LIST_READ_ONLY	This variable displays the number of structuring axes in the axes group.
OverrunOperation	Operation setting at overrun	Ready ON	INT (MC_DECEL_STOP_MODE) ☞ Page 79 MC_DECEL_STOP_MODE	LIST_READ_ONLY	This variable displays the input status of the overrun operation setting. • 1: Immediately stop (ImmediateStop)
SetAcceleration	Set current acceleration	Operation cycle	LREAL	LIST_READ_ONLY	This variable stores the set current acceleration (the acceleration calculated from the difference of the set velocity). (Unit: U/s^2) The sign indicates whether acceleration or deceleration is in progress. "0.0" is stored when the target velocity is reached. • 0.0: During stop or fixed-speed • Plus sign: During acceleration • Minus sign: During deceleration *: The stored value includes an error because a floating-point error occurs.
SetVelocity	Set velocity	Operation cycle (GroupEnable)	LREAL	LIST_READ_ONLY	During axes group operation, this variable stores the set output velocity. The value becomes more than "0.0" regardless of the movement direction of the structuring axis. Stores "0.0" after the control is completed. *: When Speed mode (VelocityMode) is set to Reference axis speed (ReferenceAxisSpeed) or Long axis speed (LongAxisSpeed) for linear interpolation control, stores the absolute value of the set velocity of the applicable axis.
StopMode_Deceleration	Deceleration at stop	Ready ON	LREAL	LIST_READ_ONLY	This variable displays the input status of deceleration at stop.
StopMode_DecelerationCurve	Stop selection at deceleration to stop	Ready ON	INT (MC_STOP_CURVE_MODE) ☞ Page 79 MC_STOP_CURVE_MODE	LIST_READ_ONLY	This variable displays the input status of stop selection at deceleration to stop. • 1: Recreate deceleration curve (OverrideCurve)
StopMode_ErrorInGroup	Structuring axis operation selection at axis stop by factors	Ready ON	INT (MC_DECEL_STOP_MODE) ☞ Page 79 MC_DECEL_STOP_MODE	LIST_READ_ONLY	This variable displays the input status of structuring axis operation selection at axis stop by factors. • 1: Immediate stop (ImmediateStop)
StopMode_General	Stop selection at stop by factors	Ready ON	INT (MC_DECEL_STOP_MODE) ☞ Page 79 MC_DECEL_STOP_MODE	LIST_READ_ONLY	This variable displays the input status of stop selection at stop by factors. • 1: Immediate stop (ImmediateStop) • 2: Keep current acceleration/deceleration (KeepCurrentAcc) • 3: Alternative acceleration/deceleration (AlternativeAcc)
TargetAcceleration	Target acceleration	Operation cycle	LREAL	LIST_READ_ONLY	This variable stores the maximum acceleration used during acceleration. (Unit: U/s^2)
TargetDeceleration	Target deceleration	Operation cycle	LREAL	LIST_READ_ONLY	This variable stores the maximum deceleration used during acceleration. (Unit: U/s^2)

Variable name	Name	Import	Data type	Attribute	Description
TargetVelocity	Target velocity	Operation cycle (GroupEnable)	LREAL	LIST_READ_ONLY	This variable stores the actual target velocity whose override and velocity limit value are considered. The value becomes more than "0.0" regardless of the movement direction of the structuring axis. Stores "0.0" after the control is completed. *: When Speed mode (VelocityMode) is set to Reference axis speed (ReferenceAxisSpeed) or Long axis speed (LongAxisSpeed) for linear interpolation control, stores the target velocity of the applicable axis.
Unit_Position	Position command unit	When axes group is valid	DWORD(HEX)	LIST_READ_ONLY	This variable displays the position command unit to be used in the Motion control. • 00000000H: pulse • 00010000H: m • FD010000H: mm • FA010000H: μm • F7010000H: nm • 00410000H: degree • FA410000H: $\times 10^{-6}$degree • 00B40000H: Revolution • 00C00000H: inch • 00FF0000H: arbitrary unit
Unit_PositionDisplay	Position command unit display	When axes group is valid	WSTRING(31)	LIST_READ_ONLY	This variable displays the currently controlled position command unit as a character string.
Unit_PositionString	Position command unit string	When axes group is valid	WSTRING(31)	LIST_READ_ONLY	When Position command unit (AxesGroupName.Pr.Unit_Position) is set to "00FF0000H: arbitrary unit", this variable displays the currently controlled position command unit set in Position command unit string (AxesGroupName.Pr.Unit_PositionString) as a character string.
Unit_Velocity	Velocity command unit	When axes group is valid	DWORD(HEX)	LIST_READ_ONLY	This variable displays the velocity command unit to be used in the Motion control. • 00000300H: U/s • 00004700H: U/min • FD000300H: U/ms • FA000300H: U/μs • F7000300H: U/ns
Unit_VelocityDisplay	Velocity command unit display	When axes group is valid	WSTRING(31)	LIST_READ_ONLY	This variable outputs the currently controlled velocity command unit as a character string.
VelocityLimit	Speed limit value	Ready ON	LREAL	LIST_READ_ONLY	This variable stores the speed limit value of the axes group.
VelocityOverride	Speed override coefficient	Operation cycle	LREAL	LIST_READ_ONLY	This variable displays the velocity override factor.
Warning	Axes group warning detection	Immediately	BOOL	LIST_READ_ONLY	This variable displays the axis warning status. • FALSE: No axes group warning • TRUE: Axes group warning present
WarningID	Axes group warning code	Immediately	WORD(HEX)	LIST_READ_ONLY	This variable stores the warning code when an axes group warning occurs.

AxesGroupName.Cd. (Axes group control data)

Variable name	Name	Import	Data type	Attribute	Description
AccelerationOverride	Acceleration override coefficient	Operation cycle	LREAL	LIST_READ_ONLY	This variable sets the acceleration override factor. • 0.01 to 10.00
ErrorReset	Axes group error reset	Service	BOOL	LIST_READ_ONLY	This variable resets axes group errors and warnings. • FALSE: Not executed • TRUE: Executes the error reset
VelocityOverride	Speed override coefficient	Operation cycle	LREAL	LIST_READ_ONLY	This variable sets the velocity override factor. • 0.00 to 10.00

System variables

The data type of system variables is expressed as SYSTEM type.

The system data type has the following members.

Member name	Data type	Description	Reference
PrConst	SYS_CONST	Stores the parameter data (constant) for the system. Opens the setting value at system variable initialization. Does not execute reimporting to the control after system variable initialization.	☞ Page 53 System.PrConst. (System parameter constant)
Pr	SYS_PRM	Stores the parameter data for the system. Opens the setting value at system variable initialization. The timing to import to the control changes depending on the parameter.	☞ Page 56 System.Pr. (System parameter)
Md	SYS_MONI	Stores the monitor data for the system. Executes refresh in the fixed cycle for each monitor data.	☞ Page 57 System.Md. (System monitor data)
Cd	SYS_CMD	Stores the command data for the system. Gets the newest value from each Motion service processing control and uses it for control.	☞ Page 61 System.Cd. (System control data)
LoggingRef	LOGGING_REF[1..10]	The data structure for input/output of the logging control function block.	☞ Page 62 System.LoggingRef.

System.PrConst. (System parameter constant)

Variable name	Name	Import	Data type	Attribute	Description
Addon_AbsSystem	Add-on AbsSystem parameter	At system start	ADDON_PARAM ☞ Page 67 ADDON_PARAM	LIST_WRITE_BACK	This variable sets the maximum capacity for the memory used in Add-on AbsSystem parameter (Addon_AbsSystem).
Addon_Axis	Add-on Axis parameter	At system start	ADDON_PARAM ☞ Page 67 ADDON_PARAM	LIST_WRITE_BACK	This variable sets the maximum capacity for the memory used in Add-on Axis parameter (Addon_Axis).
Addon_ExternalSignal	Add-on ExternalSignal parameter	At system start	ADDON_PARAM ☞ Page 67 ADDON_PARAM	LIST_WRITE_BACK	This variable sets the maximum capacity for the memory used in Add-on ExternalSignal parameter (Addon_ExternalSignal).
Addon_FileTransfer	Add-on FileTransfer parameter	At system start	ADDON_PARAM ☞ Page 67 ADDON_PARAM	LIST_WRITE_BACK	This variable sets the maximum capacity for the memory used in Add-on FileTransfer parameter (Addon_FileTransfer).
Addon_Logging	Add-on Logging parameter	At system start	ADDON_PARAM ☞ Page 67 ADDON_PARAM	LIST_WRITE_BACK	This variable sets the maximum capacity for the memory used in Add-on Logging parameter (Addon_Logging).
Addon_MotionControl_AxisFilter	Add-on MotionControl_AxisFilter parameter	At system start	ADDON_PARAM ☞ Page 67 ADDON_PARAM	LIST_WRITE_BACK	This variable sets the maximum capacity for the memory used in Add-on MotionControl_AxisFilter parameter (Addon_MotionControl_AxisFilter).
Addon_MotionControl_General	Add-on MotionControl_General parameter	At system start	ADDON_PARAM ☞ Page 67 ADDON_PARAM	LIST_WRITE_BACK	This variable sets the maximum capacity for the memory used in Add-on MotionControl_General parameter (Addon_MotionControl_General).
Addon_MotionControl_Sync	Add-on MotionControl_Sync parameter	At system start	ADDON_PARAM ☞ Page 67 ADDON_PARAM	LIST_WRITE_BACK	This variable sets the maximum capacity for the memory used in Add-on MotionControl_Sync parameter (Addon_MotionControl_Sync).
Addon_MotionEngine	Add-on MotionEngine parameter	At system start	ADDON_PARAM ☞ Page 67 ADDON_PARAM	LIST_WRITE_BACK	This variable sets the maximum capacity for the memory used in Add-on MotionEngine parameter (Addon_MotionEngine).
Addon_MotionEventHist	Add-on MotionEventHist parameter	At system start	ADDON_PARAM ☞ Page 67 ADDON_PARAM	LIST_WRITE_BACK	This variable sets the maximum capacity for the memory used in Add-on MotionEventHist parameter (Addon_MotionEventHist).
Addon_NetworkDriver_CCIETSN	Add-on NetworkDriver_CCIETSN parameter	At system start	ADDON_PARAM ☞ Page 67 ADDON_PARAM	LIST_WRITE_BACK	This variable sets the maximum capacity for the memory used in Add-on NetworkDriver_CCIETSN parameter (Addon_NetworkDriver_CCIETSN).
Addon_PackagingApp	Add-on PackagingApp parameter	At system start	ADDON_PARAM ☞ Page 67 ADDON_PARAM	LIST_WRITE_BACK	This variable sets the maximum capacity for the memory used in Add-on PackagingApp parameter (Addon_PackagingApp).

Variable name	Name	Import	Data type	Attribute	Description
Addon_PlcInstruction	Add-on PlcInstruction parameter	At system start	ADDON_PARAM ☞ Page 67 ADDON_PARAM	LIST_WRITE_BACK	This variable sets the maximum capacity for the memory used in Add-on PlcInstruction parameter (Addon_PlcInstruction).
Addon_ProfileControl	Add-on ProfileControl parameter	At system start	ADDON_PARAM ☞ Page 67 ADDON_PARAM	LIST_WRITE_BACK	This variable sets the maximum capacity for the memory used in Add-on ProfileControl parameter (Addon_ProfileControl).
Addon_Program_ST	Add-on Program_ST parameter	At system start	ADDON_PARAM ☞ Page 67 ADDON_PARAM	LIST_WRITE_BACK	This variable sets the maximum capacity for the memory used in Add-on Program_ST parameter (Addon_Program_ST).
Addon_ServoDriver_CANopen	Add-on ServoDriver_CANopen parameter	At system start	ADDON_PARAM ☞ Page 67 ADDON_PARAM	LIST_WRITE_BACK	This variable sets the maximum capacity for the memory used in Add-on ServoDriver_CANopen parameter (Addon_ServoDriver_CANopen).
Addon_ServoSystemRecorder	Add-on ServoSystemRecorder parameter	At system start	ADDON_PARAM ☞ Page 67 ADDON_PARAM	LIST_WRITE_BACK	This variable sets the maximum capacity for the memory used in Add-on ServoSystemRecorder parameter (Addon_ServoSystemRecorder).
Addon_SignalIO	Add-on SignalIO parameter	At system start	ADDON_PARAM ☞ Page 67 ADDON_PARAM	LIST_WRITE_BACK	This variable sets the maximum capacity for the memory used in Add-on SignalIO parameter (Addon_SignalIO).
BuffermemoryRefreshCycle	Buffer memory refresh cycle setting	At system start	CYCLE_PARAM ☞ Page 67 CYCLE_PARAM	LIST_WRITE_BACK	This variable specifies each setting value for the buffer memory refresh cycle. Setting and operations specific to Buffer memory refresh cycle setting (BuffermemoryRefreshCycle) are shown below. ■Cycle setting (Cycle) • 0: Synchronizes with the basic cycle of network • -11: Synchronizes with the basic cycle of network • -12: Synchronizes with the middle speed cycle of network • -13: Synchronizes with the low speed cycle of network • -110001 to -118192: Multiplication by n of the first operation cycle (1 to 8192)
EventHistoryCpu_Exclude	Event excluding CPU event history	At system start	STRING(255)	LIST_WRITE_BACK	This variable sets an event code to have not register an event to the CPU module.
EventHistoryMotion_Capacity	Motion event history capacity	At system start	INT	LIST_WRITE_BACK	This variable sets the size of the event history file. • 1 to 2048[kbyte]
EventHistoryMotion_Exclude	Motion event history excluded event	At system start	STRING(255)	LIST_WRITE_BACK	This variable sets an event code to have not register an event to the motion system.
EventHistoryMotion_Path	Motion event history path	At system start	STRING(127)	LIST_WRITE_BACK	This variable sets the store destination of the event history file.
ExcludeWarning	Excluded warning	At system start	STRING(255)	LIST_WRITE_BACK	This variable sets a warning to not be detected. <Example> • "": Not set • "0x1000, 0x1001": Will not detect warnings 0x1000 and 0x1001. • "0x1000-0x1010": Will not detect warnings 0x1000 to 0x1010.
FastOperationMode	High-speed mode setting	At system start	WORD(HEX)	LIST_WRITE_BACK	This variable switches the operation process of the entire system to high-speed mode. • 50EFH: High-speed mode • Others: Normal mode
FileTransfer_AcFile_lch	File transfer access control (File in latch drive)	At system start	WORD(HEX)	LIST_READ_ONLY	This variable sets the access control for files in the latch drive. • 0: Read protected/write protected • 1: Read protected/write permitted • 2: Read permitted/write protected • 3: Read permitted/write permitted
FileTransfer_AcFile_ram	File transfer access control (File in RAM drive)	At system start	WORD(HEX)	LIST_READ_ONLY	This variable sets the access control for files in the RAM drive. • 0: Read protected/write protected • 1: Read protected/write permitted • 2: Read permitted/write protected • 3: Read permitted/write permitted

Variable name	Name	Import	Data type	Attribute	Description
FileTransfer_AcFile_rom	File transfer access control (File in user drive)	At system start	WORD(HEX)	LIST_READ_ONLY	This variable sets the access control for files in the user drive. • 0: Read protected/write protected • 1: Read protected/write permitted • 2: Read permitted/write protected • 3: Read permitted/write permitted
FileTransfer_AcFile_sdc	File transfer access control (File in SD memory card)	At system start	WORD(HEX)	LIST_READ_ONLY	This variable sets the access control for files in the SD memory card. • 0: Read protected/write protected • 1: Read protected/write permitted • 2: Read permitted/write protected • 3: Read permitted/write permitted
FileTransfer_AcFile_sys	File transfer access control (File in system drive)	At system start	WORD(HEX)	LIST_READ_ONLY	This variable sets the access control for files in the system drive. • 0: Read protected/write protected • 1: Read protected/write permitted • 2: Read permitted/write protected • 3: Read permitted/write permitted
FileTransfer_AcLabel	File transfer access control (Label)	At system start	WORD(HEX)	LIST_READ_ONLY	This variable sets the access control for labels. • 0: Read protected/write protected • 1: Read protected/write permitted • 2: Read permitted/write protected • 3: Read permitted/write permitted
FileTransfer_LogCapacity	File transfer log capacity	At system start	INT	LIST_WRITE_BACK	This variable sets the log file capacity. • -1 to 2048[kbyte]
LabelMemorySize	Label memory size	At system start	INT	LIST_WRITE_BACK	This variable sets the label memory size. • 128 to 8192[kbyte]
Link_MotionStationRefreshType	Refresh method of send/receive data at Motion control station	At system start	INT (MC_STATION_REFRESH_MODE) ☞ Page 80 MC_STATION_REFRESH_MODE	LIST_WRITE_BACK	This variable sets the send/receive data refresh method for the Motion control station. • 0: Response preferred method (EmphasisResponse) • 1: Operation cycle preferred method (EmphasisOperationCycle)
Logging_AutoLoggingEnable	Auto logging enabled	At system start	BOOL	LIST_READ_ONLY	This variable sets whether the auto logging function is prohibited or allowed when logging setting files are stored in the SD memory card. • FALSE: Prohibited • TRUE: Allowed
MeasuresPrograms	Measurement flag for program information	At system start	BOOL	LIST_WRITE_BACK	This variable sets whether to measure the program execution time, program execution number, and program execution status. The measurement data is stored in the PROGRAM_INFO structure of each program. • FALSE: Do not measure • TRUE: Measure
OperationCycle[1]	Operation cycle setting	At system start	CYCLE_PARAM ☞ Page 67 CYCLE_PARAM	LIST_WRITE_BACK	This variable sets the first operation cycle. Setting and operations specific to Operation cycle setting (OperationCycle[1]) are shown below. ■Cycle setting (Cycle) • 0: Synchronizes with the basic cycle of network
SoftReboot_Enable	Soft reboot enabled	At system start	BOOL	LIST_WRITE_BACK	This variable sets whether software reboot execution is allowed or prohibited. When prohibited, the software reboot is not executed. • FALSE: Prohibited • TRUE: Allowed
MultiThreadMode	For manufacturer setting	At system start	WORD(HEX)	LIST_WRITE_BACK	—

System.Pr. (System parameter)

Variable name	Name	Import	Data type	Attribute	Description
ForcedStop_Signal	All axes forced stop signal	Ready ON	SIGNAL_SELECT ☞ Page 66 SIGNAL_SELECT (Signal select)	LIST_WRITE_BACK	This variable sets the signal to use the all axes forced stop. Settings and operations specific to All axes forced stop signal (ForcedStop_Signal) are shown below. ■ IO No. (StartIO) Ignores the input value. ■Target (Target) If Target is not set, signal disabled is determined and the signal undetection status is always set. Data type can only be set to [VAR], [DEV], and [CONST]. *: If an unusable data type is set, "Out of Parameter Range (System) (error code: 1D82H)" occurs. ■Signal detection method (Detection) Allows detection at the following setting level only. • 0: Detection at TRUE (HighLevel) • 1: Detection at FALSE (LowLevel) *: If edge detection specification is set, "Out of Parameter Range (System) (error code: 1D82H)" occurs. ■Compensation time (CompensationTime) Ignores the input value. ■Filter time (FilterTime) The setting range is "0.0 to 5.0[s]". *: If a value outside the range is set, "Out of Filter Time Setting Range of System Signal Warning (warning code: 0F0FH)" occurs and Filter time is operated at "0.0".
StopMode_All	Stop selection at all axes stop by factors	Ready ON	INT (MC_DECEL_STOP_MODE) ☞ Page 79 MC_DECEL_STOP_MODE	LIST_WRITE_BACK	This variable sets whether to stop immediately or decelerate to stop when an all axes stop cause occurs. • 1: Immediate stop (ImmediateStop) • 2: Keep current acceleration/deceleration (KeepCurrentAcc) • 3: Alternative acceleration/deceleration (AlternativeAcc) *: Immediate stop will be executed when executing an instruction without deceleration specification.
StopMode_AllDeceleration	Deceleration at all axes stop	Ready ON	LREAL	LIST_WRITE_BACK	This variable sets the deceleration/deceleration time at deceleration stop when an all axes stop cause occurred. ■When using acceleration/deceleration method specifying acceleration/deceleration (acceleration/deceleration specification method) • 0.0000, 0.0001 to 2147483647.0 ■When using acceleration/deceleration method specifying acceleration/deceleration time (acceleration/deceleration time-fixed method) • 0.000000, 0.000001 to 8400.0 *: If this variable is set to "0", immediately stops regardless of the setting of Operation selection at start acceleration/deceleration 0 (AccelerationZeroBehavior). *: Clamps by the upper limit value when the upper limit value is the range at stop by factors, and is treated as "0 (Immediate stop)" when the lower limit value is the range.

System.Md. (System monitor data)

Variable name	Name	Import	Data type	Attribute	Description
Addon_AbsSystem	Add-on AbsSystem monitor	Service	ADDON_MONI ☞ Page 67 ADDON_MONI	LIST_READ_ONLY	This variable displays the maximum capacity for the memory used in Add-on AbsSystem parameter (Addon_AbsSystem).
Addon_Axis	Add-on Axis monitor	Service	ADDON_MONI ☞ Page 67 ADDON_MONI	LIST_READ_ONLY	This variable displays the maximum capacity for the memory used in Add-on Axis parameter (Addon_Axis).
Addon_ExternalSignal	Add-on ExternalSignal monitor	Service	ADDON_MONI ☞ Page 67 ADDON_MONI	LIST_READ_ONLY	This variable displays the maximum capacity for the memory used in Add-on ExternalSignal parameter (Addon_ExternalSignal).
Addon_FileTransfer	Add-on FileTransfer monitor	Service	ADDON_MONI ☞ Page 67 ADDON_MONI	LIST_READ_ONLY	This variable displays the maximum capacity for the memory used in Add-on FileTransfer parameter (Addon_FileTransfer).
Addon_Logging	Add-on Logging monitor	Service	ADDON_MONI ☞ Page 67 ADDON_MONI	LIST_READ_ONLY	This variable displays the maximum capacity for the memory used in Add-on Logging parameter (Addon_Logging).
Addon_MotionControl_AxisFilter	Add-on MotionControl_AxisFilter monitor	Service	ADDON_MONI ☞ Page 67 ADDON_MONI	LIST_READ_ONLY	This variable displays the maximum capacity for the memory used in Add-on MotionControl_AxisFilter parameter (Addon_MotionControl_AxisFilter).
Addon_MotionControl_General	Add-on MotionControl_General monitor	Service	ADDON_MONI ☞ Page 67 ADDON_MONI	LIST_READ_ONLY	This variable displays the maximum capacity for the memory used in Add-on MotionControl_General parameter (Addon_MotionControl_General).
Addon_MotionControl_Sync	Add-on MotionControl_Sync monitor	Service	ADDON_MONI ☞ Page 67 ADDON_MONI	LIST_READ_ONLY	This variable displays the maximum capacity for the memory used in Add-on MotionControl_Sync parameter (Addon_MotionControl_Sync).
Addon_MotionEngine	Add-on MotionEngine monitor	Service	ADDON_MONI ☞ Page 67 ADDON_MONI	LIST_READ_ONLY	This variable displays the maximum capacity for the memory used in Add-on MotionEngine parameter (Addon_MotionEngine).
Addon_MotionEventHist	Add-on MotionEventHist monitor	Service	ADDON_MONI ☞ Page 67 ADDON_MONI	LIST_READ_ONLY	This variable displays the maximum capacity for the memory used in Add-on MotionEventHist parameter (Addon_MotionEventHist).
Addon_NetworkDriver_CCIETSN	Add-on NetworkDriver_CCIETSN monitor	Service	ADDON_MONI ☞ Page 67 ADDON_MONI	LIST_READ_ONLY	This variable displays the maximum capacity for the memory used in Add-on NetworkDriver_CCIETSN parameter (Addon_NetworkDriver_CCIETSN).
Addon_PackagingApp	Add-on PackagingApp monitor	Service	ADDON_MONI ☞ Page 67 ADDON_MONI	LIST_READ_ONLY	This variable displays the maximum capacity for the memory used in Add-on PackagingApp parameter (Addon_PackagingApp).
Addon_PlcInstruction	Add-on PlcInstruction monitor	Service	ADDON_MONI ☞ Page 67 ADDON_MONI	LIST_READ_ONLY	This variable displays the maximum capacity for the memory used in Add-on PlcInstruction parameter (Addon_PlcInstruction).
Addon_ProfileControl	Add-on ProfileControl monitor	Service	ADDON_MONI ☞ Page 67 ADDON_MONI	LIST_READ_ONLY	This variable displays the maximum capacity for the memory used in Add-on ProfileControl parameter (Addon_ProfileControl).
Addon_Program_ST	Add-on Program_ST monitor	Service	ADDON_MONI ☞ Page 67 ADDON_MONI	LIST_READ_ONLY	This variable displays the maximum capacity for the memory used in Add-on Program_ST parameter (Addon_Program_ST).
Addon_ServoDriver_CANopen	Add-on ServoDriver_CANopen monitor	Service	ADDON_MONI ☞ Page 67 ADDON_MONI	LIST_READ_ONLY	This variable displays the maximum capacity for the memory used in Add-on ServoDriver_CANopen parameter (Addon_ServoDriver_CANopen).
Addon_ServoSystemRecorder	Add-on ServoSystemRecorder monitor	Service	ADDON_MONI ☞ Page 67 ADDON_MONI	LIST_READ_ONLY	This variable displays the maximum capacity for the memory used in Add-on ServoSystemRecorder parameter (Addon_ServoSystemRecorder).
Addon_SignalIO	Add-on SignalIO monitor	Service	ADDON_MONI ☞ Page 67 ADDON_MONI	LIST_READ_ONLY	This variable displays the maximum capacity for the memory used in Add-on SignalIO parameter (Addon_SignalIO).

Variable name	Name	Import	Data type	Attribute	Description
Addon_SimpleMotion	Add-on SimpleMotion monitor	Service	ADDON_MONI ☞ Page 67 ADDON_MONI	LIST_READ_ONLY	This variable displays the maximum capacity for the memory used in Add-on SimpleMotion parameter (Addon_SimpleMotion).
BackupRestoreStatus	Backup restore information	Service	INT	LIST_READ_ONLY	This variable displays the status of backup restore. • 0: Backup restore not executed • 1: Backup stop required • 2: Backup stop • 3: Backup stop cancelled • 4: Restored data saving • 5: Restored data saving completed
BuffermemoryFreeMcfbArea	Buffer memory MCFB communication area free size	Immediately	DWORD(UDINT)	LIST_READ_ONLY	This variable displays the number of unused points of the Motion control FB area.
BuffermemoryRefreshCycle	Buffer memory refresh cycle monitor	Immediately	CYCLE_MONI ☞ Page 67 CYCLE_MONI	LIST_READ_ONLY	This variable stores each monitor value for the buffer memory refresh cycle.
Error	Motion system error detection	Immediately	BOOL	LIST_READ_ONLY	This variable displays the error status. • FALSE: No error • TRUE: Error present
ErrorID	Latest Motion system error code	Immediately	WORD(HEX)	LIST_READ_ONLY	This variable displays the latest error code.
EventHistoryMotion_ClearStatus	Motion event history clear executing state	Service	INT (MC_EXECUTION_STATE) ☞ Page 80 MC_EXECUTION_STATE	LIST_READ_ONLY	This variable displays the execution status of the event history clear. • 0: Waiting execution request (Ready) • 1: Executing (Executing) • 2: Execution completed (Done) • -1: Error occurrence (Error_)
EventHistoryMotion_Path	Motion event history path	At system start	STRING(127)	LIST_READ_ONLY	This variable stores the store destination (absolute path) of the event history file.
FileTransfer_State	File transfer executing state	Service	INT (MC_EXECUTION_STATE) ☞ Page 80 MC_EXECUTION_STATE	LIST_READ_ONLY	This variable displays the execution state of the file transfer. • 0: Waiting execution request (Ready) • 1: Executing (Executing) • -1: Error occurrence (Error_)
ForcedStop_Released	Forced stop cancelling	Operation cycle	BOOL	LIST_READ_ONLY	This variable displays the forced stop cancel status. • FALSE: All axes forced stop • TRUE: All axes forced stop release
ForcedStop_Signal	All axes forced stop signal	Ready ON	SIGNAL_SELECT ☞ Page 66 SIGNAL_SELECT (Signal select)	LIST_READ_ONLY	This variable displays the input status of the All axes forced stop signal. Outputs specific to All axes forced stop signal (ForcedStop_Signal) are shown below. ■IO No. (StartIO) "0" is always displayed. ■Target (Target) The fetch result of the parameter is displayed. ■Signal detection method (Detection) The fetch result of the parameter is displayed. ■Compensation time (CompensationTime) "0.0" is always displayed. ■Filter time (FilterTime) The fetch result of the parameter is displayed.
HwStatus_OperationgTime	Cumulative power-on time	Service	DWORD(UDINT)	LIST_READ_ONLY	This variable stores the cumulative power-on time (minute) of the Motion module.
HwStatus_RomWriteCycle	Flash ROM write count index value	Service	DWORD(UDINT)	LIST_READ_ONLY	This variable stores the count index value of the flash ROM.
LabelMemoryFreeSize	Label memory free size	Immediately	INT	LIST_READ_ONLY	This variable stores the free space of the label memory.
MemorySize	System memory size	Service	ADDON_PARAM ☞ Page 67 ADDON_PARAM	LIST_READ_ONLY	This variable stores the maximum usage amount for the system memory (RAM) and system memory (backup RAM).

Variable name	Name	Import	Data type	Attribute	Description
MemoryUsage	System memory usage	Service	ADDON_MONI ☞ Page 67 ADDON_MONI	LIST_READ_ONLY	This variable stores the current usage amount and the maximum usage amount for the system memory (RAM) and system memory (backup RAM).
NetworkError	Network error detection	Immediately	BOOL	LIST_READ_ONLY	This variable displays the status of network error occurrences. • FALSE: No network error • TRUE: Network error present
NetworkErrorID	Network error code	Immediately	WORD(HEX)	LIST_READ_ONLY	This variable stores the error code when a network error occurs.
OperationCycle[1]	Operation cycle monitor	Immediately	CYCLE_MONI ☞ Page 67 CYCLE_MONI	LIST_READ_ONLY	This variable stores the monitor value of the first operation cycle.
Program_EI	EI flag	Immediately	BOOL	LIST_READ_ONLY	This variable displays the command execution status of operation cycle execution type programs. • FALSE: During DI (Disabled) • TRUE: During EI (Enabled)
Program_NormalTaskCycle	Normal task processing time monitor	Normal	CYCLE_MONI ☞ Page 67 CYCLE_MONI	LIST_READ_ONLY	This variable stores the monitor value of normal tasks. Outputs specific to Normal task processing time monitor (Program_NormalTaskCycle) are shown below. ■Setting cycle (Cycle) Always stores "0.0" because no cycle setting exists. ■Cycle over (CycleOver) Always becomes FALSE because the cycle over detection is not performed.
Ready	Ready	Service	BOOL	LIST_READ_ONLY	This variable displays the ON/OFF status of "READY (X0)". • FALSE: OFF (Not READY) • TRUE: ON (READY)
ServoSystemRecorder_Enable	Servo system recorder enabled	Service	BOOL	LIST_READ_ONLY	This variable displays whether logging collection by the servo system recorder is enabled or disabled. • FALSE: Disabled • TRUE: Enabled
StopMode_All	Stop selection at all axes stop by factors	Ready ON	INT (MC_DECEL_STOP_MODE) ☞ Page 79 MC_DECEL_STOP_MODE	LIST_READ_ONLY	This variable displays the input status of stop selection at all axes stop by factors. • 1: Immediate stop (ImmediateStop) • 2: Keep current acceleration/deceleration (KeepCurrentAcc) • 3: Alternative acceleration/deceleration (AlternativeAcc)
StopMode_AllDeceleration	Deceleration at all axes stop	Ready ON	LREAL	LIST_READ_ONLY	This variable displays the input status of deceleration at all axes stop.
Storage_Ich	Latch drive information	Service	STORAGE_INFORMATION ☞ Page 68 STORAGE_INFORMATION	LIST_READ_ONLY	This variable displays the latch drive information.
Storage_ram	RAM drive information	Service	STORAGE_INFORMATION ☞ Page 68 STORAGE_INFORMATION	LIST_READ_ONLY	This variable displays the RAM drive information.
Storage_rom	User drive information	Service	STORAGE_INFORMATION ☞ Page 68 STORAGE_INFORMATION	LIST_READ_ONLY	This variable displays the user drive information.
Storage_sdc	SD memory card information	Service	STORAGE_INFORMATION ☞ Page 68 STORAGE_INFORMATION	LIST_READ_ONLY	This variable displays the SD memory card information.

Variable name	Name	Import	Data type	Attribute	Description
Storage_sdcForcedDisabled	SD memory card force disabled	Service	BOOL	LIST_READ_ONLY	This variable displays whether the usage of SD memory card is being forced stopped. • FALSE: Not forced stopped • TRUE: Force stopped
Storage_sdcInserted	SD memory card inserted	Service	BOOL	LIST_READ_ONLY	This variable displays whether an SD memory card is inserted. • FALSE: Not inserted • TRUE: Inserted
Storage_sdcProtected	SD memory card protected	Service	BOOL	LIST_READ_ONLY	This variable displays whether the write protect switch of the SD memory card is ON/OFF. • FALSE: Write protect switch OFF • TRUE: Write protect switch ON
Storage_sys	System drive information	Service	STORAGE_INFORMATION ☞ Page 68 STORAGE_INFORMATION	LIST_READ_ONLY	This variable displays the system drive information.
Sync	Synchronization flag	Service	BOOL	LIST_READ_ONLY	This variable displays the ON/OFF status of "Synchronization flag (X1)". • FALSE: OFF (Module access disabled) • TRUE: ON (Module access enabled)
SystemBaseCycle	System basic cycle monitor	Immediately	CYCLE_MONI ☞ Page 67 CYCLE_MONI	LIST_READ_ONLY	This variable stores the monitor value for the system basic cycle.
SystemBaseCycle_Counter	System basic cycle counter	At system start	WORD(UINT)	LIST_READ_ONLY	This variable stores the system basic cycle counter.
Version_BaseSystemSw	Basic system software version	At system start	WORD(UINT)	LIST_READ_ONLY	This variable displays the version of the basic system software.
Version_BootSw	Boot software version	At system start	WORD(UINT)	LIST_READ_ONLY	This variable displays the version of the boot software.
Version_NetworkBootSw	Network boot software version	At system start	WORD(UINT)	LIST_READ_ONLY	This variable displays the version of the network boot software.
Warning	Motion system warning detection	Immediately	BOOL	LIST_READ_ONLY	This variable displays the warning status. • FALSE: No warning • TRUE: Warning present
WarningID	Latest Motion system warning code	Immediately	WORD(HEX)	LIST_READ_ONLY	This variable stores the latest warning code.
WDTerror	WDT error	Immediately	BOOL	LIST_READ_ONLY	This variable displays the WDT error status. • FALSE: No WDT error • TRUE: WDT error present

System.Cd. (System control data)

Variable name	Name	Import	Data type	Attribute	Description
BackupRestore	Backup restore request	Service	WORD(HEX)	LIST_READ_ONLY	This variable executes backup stop and restored data saving. • 0000H: No request • ABCDH: Backup stop request • FFFEH: Restored data saving request
ErrorReset	System error reset	Service	BOOL	LIST_READ_ONLY	This variable resets all errors and warnings. • FALSE: Not executed • TRUE: Executes the system error reset
EventHistoryMotion_Clear	Motion event history clear execution request	Service	BOOL	LIST_READ_ONLY	This variable performs the event history clear execution requests. • FALSE: Not executed • TRUE: Executes the clear
FileTransfer_Command	File transfer command	Service	STRING(255)	LIST_READ_ONLY	This variable sets the command for the file transfer. When the file transfer is completed, the command is cleared automatically.
FileTransfer_Execute	File transfer execute request	Service	BOOL	LIST_READ_ONLY	This variable performs the file transfer execution requests. When the file transfer is completed, this variable becomes FALSE automatically. • FALSE: Not executed • TRUE: Executes the file transfer
SequenceReady*1*2	PLC ready	Service	BOOL	LIST_READ_ONLY	This variable sets "PLC READY (Y0)" to RUN/STOP. • FALSE: STOP • TRUE: RUN *: Becomes TRUE/FALSE automatically by rising/falling of the Motion module "Y0".
SoftRebootRequest	Soft reboot request	Service	STRING(15)	LIST_READ_ONLY	This variable executes the software reboot. • MOTION_RESET: Executes a reset of the Motion area only. • MOTION_CLEAR: Executes a clear of the memory after rebooting the Motion area. • MOTION_QCLEAR: Executes a quick clear of the memory after rebooting the Motion area. • Other than above: Not executed *: The memory is deleted after the software reboot is completed (after the system restart).
Storage_sdcForcedDisable	SD memory card force disabled command	Service	BOOL	LIST_READ_ONLY	This variable sets whether to restrict usage of the forced stop for the SD memory card. • FALSE: Forced stop release • TRUE: Forced stop
Storage_sdcRemovalProhibit	SD memory card removal prohibition	Service	BOOL	LIST_READ_ONLY	This variable sets whether to prohibit removal of the SD memory card. • FALSE: Removal allowed • TRUE: Removal prohibited

*1 When SequenceReady (PLC Ready) is set to the public label, do not operate "Y0" of the Motion module. Doing so may cause STOP/RUN to not switch correctly.

*2 The STOP/RUN status of the CPU module and the Motion module do not interlock. When in the axis controlled state by Motion control FB on the CPU module side, the axis control does not stop even if the CPU module is turned to STOP. Monitor the CPU operating status with a Motion module side program as necessary and stop the axis control. The CPU operating status can be confirmed with "CPU operating status of own station (SB004C)" or "CPU status of own station (SW004B)". In addition, when a stop error occurs in the CPU module, the Motion module also turns to STOP automatically.

System.LoggingRef.

The data type of logging control variables is expressed as LOGGING_REF type.

The logging control data type has the following members.

Member name	Data type	Description	Reference
Cd	LOGGING_CMD	Stores the command data for the logging control. Gets the newest value from each logging setting cycle and uses it for control.	☞ Page 62 System.LoggingRef[1..10].Cd. (Logging control data)
Md	LOGGING_MONI	Stores the monitor data for the logging control. Executes refresh in the fixed cycle for each monitor data.	☞ Page 62 System.LoggingRef[1..10].Md. (Logging monitor data)

■System.LoggingRef[1..10].Cd. (Logging control data)

Variable name	Name	Import	Data type	Attribute	Description
SettingPath	Logging setting path	Immediately	STRING(127)	LIST_READ_ONLY	This variable sets the storage location path (folder name, file name) of the logging setting.
StartLog	Logging request	Service	BOOL	LIST_READ_ONLY	This variable requests the execution of logging. • FALSE: STOP • TRUE: RUN
Trigger	Trigger request	Logging setting cycle	BOOL	LIST_READ_ONLY	When "Trigger command" is set for a trigger condition, this variable requests the trigger. • FALSE: No request • TRUE: Request

■System.LoggingRef[1..10].Md. (Logging monitor data)

Variable name	Name	Import	Data type	Attribute	Description
ErrStatus	Error status	Service	WORD(UINT)	LIST_READ_ONLY	This variable displays error codes related to logging.
LogDataPath	Logging data file path	Service	STRING(127)	LIST_READ_ONLY	This variable displays the path (file name) of the logging data file.
LogDataSavedCount	Number of saved files	Service	WORD(UINT)	LIST_READ_ONLY	This variable displays the number of saved data logging files.
LoggingStatus	Logging status	Service	BOOL	LIST_READ_ONLY	This variable displays the logging execution status. • FALSE: Stop • TRUE: Executing
LogType	Logging type	Logging setting cycle	INT (MC_LOGGING_TYPE) ☞ Page 80 MC_LOGGING_TYPE	LIST_READ_ONLY	This variable displays the logging type being executed. • 0: None (_None) • 1: Continuous logging (ContinuousLogging) • 2: Trigger logging (TriggerLogging)
SamplingTime[0..1]	Data collection interval	Service	DWORD(UDINT)	LIST_READ_ONLY	This variable displays the data collection interval in units of ns. • [0] Sampling interval lower 32 bits • [1] Sampling interval upper 32 bits
SaveStatus	Save status	Service	INT (MC_LOGGING_SAVE_STATUS) ☞ Page 80 MC_LOGGING_SAVE_STATUS	LIST_READ_ONLY	This variable displays the status of collected data being saved to files. • 0: Unsaved (UnSave) • 1: Saving (Saving) • 2: Saved (Saved) • 3: All files saved (FullySaved)
SettingPath	Logging setting file	Immediately	STRING(127)	LIST_READ_ONLY	This variable displays the path (file name) of the logging setting file during the logging.
TriggerCount	Trigger detection count	Logging setting cycle	INT	LIST_READ_ONLY	This variable displays the trigger detection count.
TrigerCountIgnored	Trigger ignored count	Logging setting cycle	INT	LIST_READ_ONLY	This variable displays the ignored trigger count for triggers that occurred again while collecting data.
Triggered	Triggered	Logging setting cycle	BOOL	LIST_READ_ONLY	This variable displays the triggered status. • FALSE: Not triggered • TRUE: Triggered

Variable name	Name	Import	Data type	Attribute	Description
TriggerStatus[1..32]	Trigger status	Logging setting cycle	BOOL	LIST_READ_ONLY	This variable displays the status of each trigger condition. • FALSE: Trigger condition not established • TRUE: Trigger condition established
TriggerStatusWord	Trigger status(word)	Logging setting cycle	DWORD(HEX)	LIST_READ_ONLY	This variable displays the status of each trigger condition as a bit (b0 to b31). This will reflect the same status as Trigger status (TriggerStatus[1..32]) variable.

Other structure variables

MC_CAM_REF

Variable name	Name	Data type	Description
ProfileData	Profile	PROFILE_DATA ☞ Page 66 PROFILE_DATA	This variable sets the cam data profile.

MC_CAM_ID

Variable name	Name	Data type	Description
ProfileID	Profile ID	PROFILE_ID ☞ Page 66 PROFILE_ID	This variable sets the cam data profile ID.

MC_TRIGGER_REF

Variable name	Name	Data type	Description
Signal	Trigger signal	SIGNAL_SELECT ☞ Page 66 SIGNAL_SELECT (Signal select)	This variable sets the trigger signal.

MC_INPUT_REF

Variable name	Name	Data type	Description
Signal	Input signal	SIGNAL_SELECT ☞ Page 66 SIGNAL_SELECT (Signal select)	This variable sets the input signal.

MC_OUTPUT_REF

Variable name	Name	Data type	Description
Signal	Output signal	SIGNAL_SELECT ☞ Page 66 SIGNAL_SELECT (Signal select)	This variable sets the output signal.

TARGET_REF (Input signal)

Variable name	Name	Data type	Description
StartIO	IO No.	WORD(HEX)	This variable sets the I/O No.
Target	Target	WSTRING(63)	This variable sets the signal used in control with the following character string formats. *: The setting is not case-sensitive. In addition, spaces are ignored. ■[Classification](Type)Data name.Bit position@Target modification • [Classification]*1 • (Type)*2 • Data name*1 • .Bit position*3 • @Target modification*4 ■[Classification](Data type)WSTRING type label^ • [Classification]*1 • (Data type)*2 • WSTRING type label^*8

*1 The data classification is set in "[Classification]", and the target data is set in "Data name". The method for setting the data name differs depending on the set classification.

Classification	Target	Data name		Setting
		Format	Description	
[OBJ]	CANopen object of the slave device (which has been mapped to cyclic data)	[OBJ]0x□□□□**##	Sets the index, subindex, and size of the object. • □□□□: Index • **: Subindex • ##: Size (in bits) *: Match the (type) setting with the object size. *: Some of the objects can be used as an external signal high-accuracy input. *: Objects with a size of "1 byte" cannot be used.	[OBJ]0x607A0020 *: For an MR-J5(W)-G Target position (Obj.607Ah: 00h) 4 byte object(20H)
[VAR]	Label of the Motion system	[VAR]Label name	Sets a label in the Motion system. *: When setting a local label, set the POU name in "@Target modification".	[VAR]ADunit10.OutputEnable
[AXIS]	Axis data	[AXIS]MC_SOURCE	Sets an MC_SOURCE enumeration type enumerator. (Type) for the data can only be set to "LREAL". • mcSetValue (Set value) • mcActualValue (Actual value) • mcLatestSetValue (Latest set value) • mcLatestActualValue (Latest actual value) *: This data cannot be written.	[AXIS]mcSetValue
[DEV]	Device in the Motion system (Including buffer memory and link device of the module itself)	[DEV]Data name	Sets an device in the module itself. • Buffer memory (G) • Remote input (RX) • Remote output (RY) • Remote register (RWw, RWr) • Link special relay (SB) • Link special register (SW) *: For information about restrictions when using devices, refer to "Direct access from motion built-in program to link devices" in the following manual. MELSEC iQ-R Motion Module User's Manual (Network)	[DEV]RWr10.5 [DEV]G11500000.1
[CONST]	Constant	[CONST]Constant [CONST]0xConstant	Sets a floating decimal (E format is also available), decimal integer, or hexadecimal integer. *: This data cannot be written.	[CONST]1000 [CONST]0x100

*2 This is written when explicitly setting the data type. The following data types can be set.
When "(BOOL)" is specified for data set in Classification having a data type other than BOOL type, it will be handled as though "Bit position.0" was omitted.

Type	Setting example
(BOOL)	■If [DEV](BOOL)G11500000 The setting will be handled as [DEV](BOOL)G11500000.0.
(INT)	
(DINT)	
(WORD)	
(DWORD)	
(REAL)	
(LREAL)	

*3 When the bit position is added to data of the [Classification] shown below, the data is handled as though it is BOOL type.
When (Type) and the bit position are both set, the setting for (Type) is ignored and the data becomes BOOL type data.

Classification	Bit position setting range
[OBJ]	0 to F
[VAR] (only WORD type, DWORD type, INT type, DINT type)	
[DEV]	

*4 Sets the auxiliary information used for specifying the data. The details differ depending on the set [Classification].
Certain functions that use signals require the target modification to be set. When specifying the target modification for data types and functions that do not require it, the specification will be ignored.

Classification	Target modification		Setting example
	Format	Description	
[OBJ]	@Station address	Sets the station address (IP address to identify the slave device). When multiple logic axes are included in 1 station such as with a multi-axis drive unit, set multi-drop numbers to identify certain logic axes. @192.168.3.10#2 Multi-drop No. IP address *: The multi-drop No. is set as "# + No. (decimal)". · #0: A-axis · #1: B-axis · #2: C-axis *: The multi-drop No. can be omitted. When it is omitted, it is regarded as "#0".	@192.168.3.10 @192.168.3.10#2
[VAR]	@POU name	The POU name is set when setting local labels. When @POU is not written, the data is handled as a global label regardless of the setting which uses the same label name for a global label and a local label. *: POU: Program Organization Unit	
[AXIS]	@Position @CumulativePos	Sets the target data (position). @Position and @CumulativePos are set with the MC_SOURCE type enumerator set in "Data name".*5	@Position @CumulativePos
[DEV]	—	The target modification cannot be set. It is ignored if set.	
[CONST]	@+adding value @-subtracting value	Used to change the read value each time the value is referred. Setting the target modification allows the value to be changed each cycle. When the data type is REAL or LREAL, this setting is invalid.	[CONST](INT)0@-10*6 [CONST](BOOL)0x00.3@+1*7

*5 The following enumerators can be set.
○: Settable, ×: Not settable

MC_SOURCE type enumerator	Description	Modification	
		@Position	@CumulativePos
1: Set value (mcSetValue)	Set value of previous operation cycle	○	○
2: Actual value (mcActualValue)	Actual value of previous operation cycle	○	×
101: Latest set value (mcLatestSetValue)	Set value of latest operation cycle	○	○
102: Latest actual value (mcLatestActualValue)	Actual value of latest operation cycle	○	×

*6 The referred value becomes "0, -10, -20, -30, …".
*7 The referred value switches between TRUE/FALSE every eight (2^3) cycles. (Adds 1 byte value "+1" every cycle, then retrieves the status of bit 3.)
For [CONST] bit specification, "0 to 7" can be used.

*8 When setting to a format with more than 63 characters, use WSTRING type labels and use the following method to perform the setting.
(1) Declare the WSTRING type global label in the Motion system, and store the following portion: Data name.Bit position@Target modification.
(2) Set the label name from (1) as the WSTRING type label of TARGET_REF. (Add "^" to the end of the name to indicate reference.)

SIGNAL_SELECT (Signal select)

Variable name	Name	Data type	Description
Source	Signal	TARGET_REF ☞ Page 64 TARGET_REF (Input signal)	This variable sets BOOL type signals used in control. For Source (Signal), only the following data classifications can be set. • [VAR] • [DEV] • [CONST] • [OBJ] (which has been mapped to cyclic data)
Detection	Signal detection method	INT (MC_SIGNAL_LOGIC) ☞ Page 79 MC_SIGNAL_LOGIC	This variable sets the signal logic. • 0: Detection at TRUE (HighLevel) • 1: Detection at FALSE (LowLevel) • 2: Detection at FALSE→TRUE(rising edge) (RisingEdge) • 3: Detection at TRUE→FALSE (falling edge) (FallingEdge) • 4: Detection at rising edge/falling edge (BothEdges)
CompensationTime	Compensation time	LREAL	This variable sets the compensation time of the I/O signal in units of seconds. *: The setting range differs for each function.
FilterTime	Filter time	LREAL	This variable sets the filter time for removing chattering of the input signal in units of seconds. *: The setting range differs for each function.

PROFILE_DATA

Variable name	Name	Data type	Description
Location	Calculation profile data storage location	FILE_LOCATION ☞ Page 66 FILE_LOCATION	This variable sets the file name and storage location of the calculation profile.
ID	Profile ID	PROFILE_ID ☞ Page 66 PROFILE_ID	This variable sets the number for the profile ID of the calculation profile.

FILE_LOCATION

Variable name	Name	Data type	Description
FileName	File name	STRING(63)	This variable sets the file name. (Up to 63 characters including the extension can be set.)*1
Path	Folder specification	STRING(63)	This variable sets the folder path in which the file is stored. (Up to 63 characters can be set.)*1

*1 Set the total character amount for File name (FileName) and Folder specification (Path) to within 127 characters (including null).

PROFILE_ID

Variable name	Name	Data type	Description
Number	Profile ID No.	WORD(UINT)	This variable sets the Profile ID No. • 0 to 60000

CYCLE_PARAM

Variable name	Name	Data type	Description
Cycle	Cycle setting	DINT	This variable sets the cycle. • 0: Synchronizes with the basic cycle of network • -11: Synchronizes with the basic cycle of network • -12: Synchronizes with the middle speed cycle of network • -13: Synchronizes with the low speed cycle of network • -100001 to -100256: Multiplication by n of the system basic processing (1 to 256) • -118192 to -110001: Multiplication by n of the first operation cycle (1 to 8192)
NumOfCycleOverWngDetectTimes	Number of cycle over warning detections	WORD(UINT)	This variable sets the number of cycles until the cycle over warning is detected for each cycle. When set to "0", warnings are not detected. • 0 to 65535
NumOfCycleOverErrDetectTimes	Number of cycle over error detections	WORD(UINT)	This variable sets the number of cycles until the cycle over error is detected for each cycle. • 0 to 65535
CycleOverErrorType	Cycle over error selection	INT (MC_ERROR_CLASS) ☞ Page 80 MC_ERROR_CLASS	This variable sets the assignment of the over error for each cycle. • 2: Minor error (MinorError) • 3: Moderate error (ModerateError)

CYCLE_MONI

Variable name	Name	Data type	Description
ProcessingTime	Processing time	DWORD(UDINT)	This variable stores the processing time of each cycle in units of [ns].
MaximumProcessingTime	Maximum processing time	DWORD(UDINT)	This variable stores the maximum processing time of each cycle in units of [ns].
Cycle	Setting cycle	DWORD(UDINT)	This variable stores the cycle setting of each cycle in units of [ns].
CycleOver	Cycle over	BOOL	This variable becomes TRUE when the cycle over is detected for each cycle.

ADDON_PARAM

Variable name	Name	Data type	Description
RamSizeMax	Maximum RAM size	DWORD(UDINT)	This variable sets the maximum amount [kbyte] of system memory (RAM) used in the add-ons.
BackupRamSizeMax	Maximum backup RAM size	DWORD(UDINT)	This variable sets the maximum amount [kbyte] of system memory (backup RAM) being used in the add-ons.

ADDON_MONI

Variable name	Name	Data type	Description
RamUsage	RAM usage	DWORD(UDINT)	This variable stores the current amount [kbyte] of system memory (RAM) being used in the add-ons.
RamMaxUsage	RAM usage maximum	DWORD(UDINT)	This variable stores the maximum amount [kbyte] of system memory (RAM) being used in the add-ons.
BackupRamUsage	Backup RAM usage	DWORD(UDINT)	This variable stores the current amount [kbyte] of system memory (backup RAM) being used in the add-ons.
BackupRamMaxUsage	Backup RAM usage maximum	DWORD(UDINT)	This variable stores the maximum amount [kbyte] of system memory (backup RAM) being used in the add-ons.
Version	Version	STRING(15)	This variable stores the version information of the add-ons.

STORAGE_INFORMATION

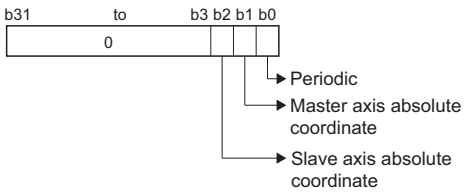
Variable name	Name	Data type	Description
Capacity	Capacity	DWORD(UDINT)	This variable stores the capacity [kbyte] of the drive.
FreeSpace	Free space	DWORD(UDINT)	This variable stores the free space [kbyte] of the drive.
Mount	Mount status	BOOL	This variable becomes TRUE when file access is enabled at drive mount status.

INSTANCE_ID

Variable name	Name	Data type	Description
StartIO	IO No.	WORD(HEX)	This variable sets the I/O No.
Number	Instance ID	WORD(UINT)	This variable sets the number for the instance ID.

PROFILE_CAM_DATA

Variable name	Name (device offset)	Data type	Description
Interpolate	Specifying interpolation method (+0)	INT	This variable sets the specifying interpolation method for the cam data. • 0: Linear interpolation • 1: Specifying for each section • 2: Spline interpolation
Resolution	Resolution (+2)	DWORD(UDINT)	This variable sets the cam resolution amount for one cycle. • 8 to 65535
InputUnitString	Input unit string (+4)	WSTRING(31)	This variable sets the unit for input data with a character string. When omitted, the setting is handled as "pulse".
OutputUnitString	Output unit string (+36)	WSTRING(31)	This variable sets the unit for output data with a character string. When omitted, the setting is handled as "pulse".
StartPoint	Start point*1 (+68)	LREAL	This variable sets the start point. Set this variable to "0.0". *: When set to a value other than "0.0", "Calculation Profile Incorrect (error code: 3410H)" occurs.
StartStroke	Initial stroke amount*1 (+72)	LREAL	This variable sets the stroke amount for the start point. (-Stroke amount (Stroke) ≤ Initial stroke amount (StartStroke) ≤ Stroke amount (Stroke)) • -100.0000000 to 100.0000000
StartVelocity	Initial speed*2 (+76)	LREAL	This variable sets the velocity for the start point. • -2500000000.0 < Initial speed (StartVelocity) ≤ 2500000000.0
StartAcceleration	Initial acceleration*2 (+80)	LREAL	This variable sets the acceleration for the start point. • -2147483647.0 < Initial acceleration (StartAcceleration) ≤ 2147483647.0
CycleLength	Length per cycle*1 (+84)	LREAL	This variable sets the input amount that is required for one cycle. (0.00000000000001 ≤ Length per cycle (CycleLength) ≤ 10000000000.0) • 0.00000000000001 to 10000000000.0
CycleMin	Minimum value per cycle (+88)	LREAL	This variable sets the minimum value per cycle. Set this variable to "0.0". *: When set to a value other than "0.0", "Calculation Profile Incorrect (error code: 3410H)" occurs.
CycleMax	Maximum value per cycle (+92)	LREAL	This variable sets the maximum value per cycle. (Minimum value per cycle (CycleMin) < Maximum value per cycle (CycleMax) ≤ 10000000000.0) • -10000000000.0 to 10000000000.0
CycleTime	Time per cycle*2 (+96)	LREAL	This variable sets the time per cycle. (0.001 < Time per cycle (CycleTime) ≤ 100000.0) • 0.001 to 100000.000
Stroke	Stroke amount (+100)	LREAL	This variable sets the stroke amount. (0.00000000000001 ≤ Stroke amount (Stroke) ≤ 10000000000.0) • 0.00000000000001 to 10000000000.0

Variable name	Name (device offset)	Data type	Description
NumberOfSections	Number of sections*1 (+104)	WORD(UINT)	This variable sets the number of sections for the cam data. ■When set to "1: Specifying for each section" • 1 to 360 ■When set to "2: Spline interpolation" • 3 to 360 *: When set to "0: Linear interpolation", this variable is not used. It is ignored if set.
Options	Options (+106)	DWORD(HEX)	Set Options by bit specification.  ■Periodic (Bit 0) • 0: Non periodic • 1: Periodic ■Master axis absolute coordinate (Bit 1) • 0: Relative coordinate • 1: Setting not possible*3 ■Slave axis absolute coordinate (Bit 2) • 0: Relative coordinate • 1: Absolute coordinate *: Set bits 3 to 31 to "0".

*1 Not used when Specifying interpolation method (Interpolate) is set to "0: Linear interpolation". It is ignored if set.

*2 Only used when the cam curve is "12: 5th curve (Adjustment) (FifthCurve_SpeedDesignation)". It is ignored for other curves.

*3 If "1" is set, "Out of Input Absolute Coordinate Range (error code: 341DH)" occurs.

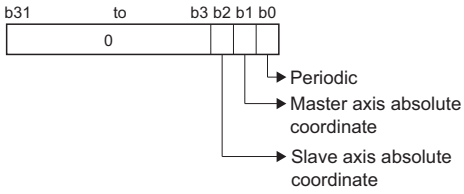
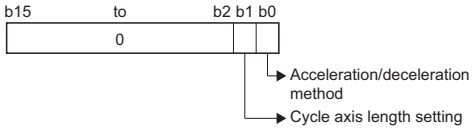
PROFILE_CAM_ELEMENT

Variable name	Name (device offset)	Data type	Description
CurveType	Cam curve type (+0)	INT (MC_CAM_CURVE_TYPE) ☞ Page 78 MC_CAM_CURVE_TYPE	This variable sets the cam curve. • 0: Constant speed (ConstantSpeed) • 1: Constant acceleration (ConstantAcceleration) • 2: Distorted trapezoid (DistortedTrapezoid) • 3: Distorted sine (DistortedSine) • 4: Distorted constant speed (DistortedConstantSpeed) • 5: Cycloid (Cycloid) • 6: 5th curve (FifthCurve) • 7: Trapecloid (Trapecloid) • 8: Reverse trapecloid (ReverseTrapecloid) • 9: Double hypotenuse (DoubleHypotenuse) • 10: Reverse double hypotenuse (ReverseDoubleHypotenuse) • 11: Single hypotenuse (SingleHypotenuse) • 12: 5th curve (Adjustment) (FifthCurve_SpeedDesignation)
EndPoint	End point*1 (+4)	LREAL	This variable sets the position (the current value per cycle) to the length per cycle. • 0.0 < End point (EndPoint) ≤ Length per cycle (CycleLength)
Stroke	Stroke (+8)	LREAL	This variable sets the stroke position. • The absolute value of Stroke (Stroke) ≤ Stroke amount (Stroke)
RangeP1	Curve application range (P1) (+12)	LREAL	This variable sets the curve application range (start point: P1, end point: P2) of the cam curve. Set this variable within the range of "P1 < P2". Note that when set to "P1 = P2 = 0", "P1 = 0" and "P2 = 1" are applied. • 0.0 to 1.0
RangeP2	Curve application range (P2) (+16)	LREAL	
RangeL1	Acceleration/deceleration range compensation (Range L1) (+20)	LREAL	This variable sets the acceleration/deceleration range (L1, L2) for the cam curve. The range that can be set differs depending on the cam curve. When set to "L1 = L2 = 0.0000", the default value for each cam curve is applied. The setting value is ignored for curves that do not use L1 or L2. • 0.0001 < Range (L1, L2) < 1.0000
RangeL2	Acceleration/deceleration range compensation (Range L2) (+24)	LREAL	
EndVelocity	End point speed*2 (+28)	LREAL	This variable sets the end point velocity of the cam curve. • -2500000000.0 < End point speed (EndVelocity) < 2500000000.0
EndAcceleration	End point acceleration*2 (+32)	LREAL	This variable sets the end point acceleration of the cam curve. • -2147483647.0 < End point acceleration (EndAcceleration) < 2147483647.0

*1 When the end point exceeds the length per cycle before reaching the final section, the section number at the time is determined as the final section number and the end point is overwritten with the length per cycle.

*2 The setting items used to read/write differ depending on Specifying interpolation method of the calculation profile.

PROFILE_ROTARY_CUTTER

Variable name	Name (device offset)	Data type	Description
Resolution	Resolution (+0)	DWORD(UDINT)	This variable sets the resolution of the generated cam. • 8 to 32768
InputUnitString	Input unit string (+2)	WSTRING(31)	This variable sets the input unit character string.
Options	Options (+34)	DWORD(HEX)	Set Options by bit specification.  ■Periodic (Bit 0) • 0: Non periodic • 1: Periodic ■Master axis absolute coordinate (Bit 1) • 0: Relative coordinate • 1: Setting not possible*1 ■Slave axis absolute coordinate (Bit 2) • 0: Relative coordinate • 1: Absolute coordinate *: Set bits 3 to 31 to "0".
AutoGenerationOptions	Auto-generation option (+36)	WORD(HEX)	This variable sets the auto-generation options with bit specification.  ■ Acceleration/deceleration method (Bit 0) • 0: Trapezoidal acceleration/deceleration • 1: S-curve acceleration/deceleration ■Synchronous axis length setting (Bit 1) • 0: Diameter • 1: Cycle length *: Set bits 2 to 15 to "0".
NumberOfCutter	Number of cutter (+37)	WORD(UINT)	This variable sets the number of cutters. • Setting range: 1 to 256
SyncSectionAccRatio	Synchronous section acceleration ratio (+40)	LREAL	Set this variable to precisely adjust the synchronous speed in the synchronous section. The value becomes "Synchronous section speed = Synchronous speed × (100% + Acceleration ratio)". • -50.0 to 50.0
SheetLength	Sheet length (+44)	LREAL	This variable sets the sheet length. (0.0 < Sheet length (SheetLength) ≤ 10000000000.0) • 0.00000000000001 to 10000000000.0
SheetSyncWidth	Sheet synchronization width (+48)	LREAL	This variable sets the sheet synchronization width (seal width). When the synchronous speed section for retracting operation is required in front of and behind the sheet synchronization width, add the retracting width. (0.0 < Sheet synchronization width (SheetSyncWidth) < Sheet length (SheetLength)) (0.0 < Sheet synchronization width (SheetSyncWidth) < Cycle length/Number of cutter (NumberOfCutter)) • 0.00000000000001 to 10000000000.0
SyncAxisLength	Synchronous axis length (+52)	LREAL	This variable sets the rotary cutter axis length. ■When Auto-generation options is set to "0: Diameter" Calculated as "Cycle length = setting value × π". ■When Auto-generation options is set to "1: Cycle length" Calculated as "Cycle length = setting value". (0.0 < Cycle length < 10000000000.0) • 0.00000000000001 to 10000000000.0

Variable name	Name (device offset)	Data type	Description
SyncPositionAdjustment	Synchronous position tuning (+56)	LREAL	This variable sets the position adjustment of the synchronous section. Negative value: The synchronous section is adjusted to the sheet start side. 0: The center of the sheet is in the synchronous section. Positive value: The synchronous section is adjusted to the sheet end side. (Absolute value of Synchronous position tuning (SyncPositionAdjustment) < Sheet length (SheetLength) / 2) • -10000000000.0 to 10000000000.0
AccDecWidth	Acceleration/deceleration width (+60)	LREAL	This variable sets the sheet width (one side) of the acceleration/deceleration area. When a negative value is set, the acceleration/deceleration width is calculated to be the maximum. (2 × Acceleration/deceleration width (AccDecWidth) ≤ Sheet length (SheetLength) - Sheet synchronization width (SheetSyncWidth)) • -10000000000.0 to 10000000000.0
AsyncSpd	Asynchronous speed result (+64)	LREAL	When the auto-generation is successfully completed, this variable stores the asynchronous speed as the ratio of the synchronous speed.

*1 If "1" is set, "Out of Input Absolute Coordinate Range (error code: 341DH)" occurs.

PROGRAM_INFO

Variable name	Name	Data type	Description
ExecutionTime	Execution time	DWORD(UDINT)	This variable stores the execution time of the program in units of [μs]. 0 clears at STOP→RUN.
Number	Execution count	WORD(UINT)	This variable counts the number of executions for the program. After exceeding 65535, the value returns to 0. 0 clears at STOP→RUN. • 0 to 65535
Status	Status	WORD(HEX)	This variable stores the execution status of the program. • 0: Standby (MC_PROGRAM_WAITING) • 1: Executing (MC_PROGRAM_RUNNING) • 3: Error stop (MC_PROGRAM_ERROR) • 4: Standby by DI (MC_PROGRAM_DI)

SLAVE_OBJECT_REAL

Variable name	Name	Data type	Description
ControlWord	ControlWord	WSTRING(63)	This variable requests switching of the status to the driver device.
EncoderIncrements	EncoderIncrements	WSTRING(63)	This variable gets the encoder resolution of the driver device.
FollowingErrActualValue	FollowingErrActualValue	WSTRING(63)	This variable is the droop pulse of the driver device.
GearRatioMotorRevolutions	GearRatioMotorRevolutions	WSTRING(63)	This variable sets the servo motor axis rotation number (numerator) to the driver device.
HomeOffset	HomeOffset	WSTRING(63)	This variable sets the difference between zero position and homing position of the machine coordinate system in the driver device.
MaxMotorSpeed	MaxMotorSpeed	WSTRING(63)	This variable gets the maximum speed of the servo motor from the driver device.
MaxTorque	MaxTorque	WSTRING(63)	This variable gets the maximum torque of the servo motor from the driver device.
ModesOfOp	ModesOfOp	WSTRING(63)	This variable requests the switching the control mode to the driver device.
ModesOfOpDisp	ModesOfOpDisp	WSTRING(63)	This variable gets the control mode of the driver device.
NegativeTorqueLimitValue	NegativeTorqueLimitValue	WSTRING(63)	This variable sets negative direction torque limit value in the driver device.
Polarity	Polarity	WSTRING(63)	This variable sets the rotation direction selection of driver device.
PosActualValue	PosActualValue	WSTRING(63)	This variable is the current value of the driver device.

Variable name	Name	Data type	Description
PosEncoderResolution	PosEncoderResolution	WSTRING(63)	This variable returns the total number of Sub Index(=2).
PosEncoderResolutionMotorRevolutions	PosEncoderResolutionMotorRevolutions	WSTRING(63)	This variable is the number of rotations of the driver device.
PositiveTorqueLimitValue	PositiveTorqueLimitValue	WSTRING(63)	This variable sets the positive direction torque limit value in the driver device.
ShaftRevolutions	ShaftRevolutions	WSTRING(63)	This variable sets the number of drive axis rotations (denominator) of the driver device.
SiUnitPos	SiUnitPos	WSTRING(63)	This variable sets gets the SI unit position of the driver device.
SiUnitVel	SiUnitVel	WSTRING(63)	This variable sets gets the SI unit velocity of the driver device.
StatusWord	StatusWord	WSTRING(63)	This variable gets the status of the driver device.
SupportedDriveModes	SupportedDriveModes	WSTRING(63)	This variable gets the control mode supported by the driver device.
TargetPos	TargetPos	WSTRING(63)	This variable is the command position to output to the driver device.
TargetTorque	TargetTorque	WSTRING(63)	This variable is the command torque to output to the driver device.
TargetVelocity	TargetVelocity	WSTRING(63)	This variable is the set velocity to output to the driver device.
TargetActualValue	TorqueActualValue	WSTRING(63)	This variable is the current torque of the driver device.
VelActualValue	VelActualValue	WSTRING(63)	This variable is the current speed of the driver device.
vControlDil1	ControlDi1	WSTRING(63)	This variable sets the input device in driver device.
vControlDil2	ControlDi2	WSTRING(63)	This variable sets the input device in driver device.
vControlDil3	ControlDi3	WSTRING(63)	This variable sets the input device in driver device.
vControlDil4	ControlDi4	WSTRING(63)	This variable sets the input device in driver device.
vControlDil5	ControlDi5	WSTRING(63)	This variable sets the input device in driver device.
vControlDil6	ControlDi6	WSTRING(63)	This variable sets the input device in driver device.
vControlDil7	ControlDi7	WSTRING(63)	This variable sets the input device in driver device.
vCurrentAlarm	CurrentAlarm	WSTRING(63)	This variable gets the occurring alarm from the driver device.
vEncoderStatus1	EncoderStatus1	WSTRING(63)	This variable gets the encoder status from the driver device.
vHomeAbsCounter	HomeAbsCounter	WSTRING(63)	This variable gets the encoder multiple revolution counter saved as the home position from the driver device.
vHomeCycleCounter	HomeCycleCounter	WSTRING(63)	This variable gets the encoder position within one revolution saved as the home position from the driver device.
vInitialAbsCounter	InitialAbsCounter	WSTRING(63)	This variable gets the encoder multiple revolution counter at power-on from the driver device.
vInitialCycleCounter	InitialCycleCounter	WSTRING(63)	This variable gets the encoder position within one revolution at power-on from the driver device.
vInitialPos	InitialPos	WSTRING(63)	This variable gets the current position at power-on from the driver device.
vMaxAbsCounter	MaxAbsCounter	WSTRING(63)	This variable gets the maximum value of the encoder multiple revolution counter from the driver device.
vStatusDo1	StatusDo1	WSTRING(63)	This variable gets the output device status of the driver device.
vStatusDo2	StatusDo2	WSTRING(63)	This variable gets the output device status of the driver device.
vStatusDo3	StatusDo3	WSTRING(63)	This variable gets the output device status of the driver device.
vStatusDo4	StatusDo4	WSTRING(63)	This variable gets the output device status of the driver device.
vStatusDo5	StatusDo5	WSTRING(63)	This variable gets the output device status of the driver device.
vStatusDo6	StatusDo6	WSTRING(63)	This variable gets the output device status of the driver device.
vStatusDo7	StatusDo7	WSTRING(63)	This variable gets the output device status of the driver device.
vSupportedControlDi1	SupportedControlDi1	WSTRING(63)	This variable gets the input device supported by the driver device.
vSupportedControlDi2	SupportedControlDi2	WSTRING(63)	This variable gets the input device supported by the driver device.
vSupportedControlDi3	SupportedControlDi3	WSTRING(63)	This variable gets the input device supported by the driver device.
vSupportedControlDi4	SupportedControlDi4	WSTRING(63)	This variable gets the input device supported by the driver device.
vSupportedControlDi5	SupportedControlDi5	WSTRING(63)	This variable gets the input device supported by the driver device.

Variable name	Name	Data type	Description
vSupportedControlDi6	SupportedControlDi6	WSTRING(63)	This variable gets the input device supported by the driver device.
vSupportedControlDi7	SupportedControlDi7	WSTRING(63)	This variable gets the input device supported by the driver device.
vSupportedStatusDo1	SupportedStatusDo1	WSTRING(63)	This variable gets the output device supported by the driver device.
vSupportedStatusDo2	SupportedStatusDo2	WSTRING(63)	This variable gets the output device supported by the driver device.
vSupportedStatusDo3	SupportedStatusDo3	WSTRING(63)	This variable gets the output device supported by the driver device.
vSupportedStatusDo4	SupportedStatusDo4	WSTRING(63)	This variable gets the output device supported by the driver device.
vSupportedStatusDo5	SupportedStatusDo5	WSTRING(63)	This variable gets the output device supported by the driver device.
vSupportedStatusDo6	SupportedStatusDo6	WSTRING(63)	This variable gets the output device supported by the driver device.
vSupportedStatusDo7	SupportedStatusDo7	WSTRING(63)	This variable gets the output device supported by the driver device.
vSyncAbsCounter	SyncAbsCounter	WSTRING(63)	This variable gets the encoder multiple revolution counter from the driver device.
vSyncCycleCounter	SyncCycleCounter	WSTRING(63)	This variable gets the encoder multiple revolution counter from the driver device.
vVelLimitValue	VelLimitValue	WSTRING(63)	This variable sets the speed limit value in the driver device.
vWatchDogCounterDI	WatchDogCounterDI	WSTRING(63)	This variable notifies the watch dog counter value to the driver device.
vWatchDogCounterUI	WatchDogCounterUI	WSTRING(63)	This variable gets the watch dog counter value from the driver device.
vEncoderStatus2	EncoderStatus2	WSTRING(63)	This variable gets the scale measurement encoder status from the driver device.
vScaleCycleCounter	ScaleCycleCounter	WSTRING(63)	This variable gets the position within one revolution of the scale measurement encoder from the driver device.
vScaleAbsCounter	ScaleAbsCounter	WSTRING(63)	This variable gets the multiple revolution counter of the scale measurement encoder from the driver device.
vScaleMeasurementEncoderResolution	ScaleMeasurementEncoderResolution	WSTRING(63)	This variable gets the scale measurement encoder resolution from the driver device.
vScaleMeasurementEncoderReceptionStatus	ScaleMeasurementEncoderReceptionStatus	WSTRING(63)	This variable gets the encoder position within one revolution from the driver device.

SLAVE_OBJECT_ENCODER

Variable name	Name	Data type	Description
PosActualValue	PosActualValue	WSTRING(63)	This variable sets the data storing the synchronous encoder input. When the I/O device is connected, imports the value from the set location.

SLAVE_OBJECT_VIRTUAL_ENCODER

Variable name	Name	Data type	Description
PosActualValue	PosActualValue	WSTRING(63)	Set the string format data used as the encoder input value. The settable types are only [VAR], [DEV], and [CONST]. ■When [VAR] is set • Sets data of the (INT), (DINT), (WORD), and (DWORD) data types. • When [VAR] is used, local labels cannot be set. ■When [DEV] or [CONST] is set • Sets data of the (INT), (DINT), (WORD), and (DWORD) data types. *: When this setting is omitted, uses Encoder input value (AxisName.Cd.Encoder_InputValue).

2.2 List of Enumerators

The list of enumerators used for various parameters, monitor data, and Motion control FBs is shown below.

ENUM enumerators

The enumeration type constant used for various parameters, monitor data, and Motion control FBs actually uses INT type values.

"Enumeration type name__enumerator" INT type global labels are available on the engineering tool.

Note that INT type global labels can only be used for programs created on the Motion module side.

Use a constant for programs created on the CPU module side.

○: Settable, ×: Not settable

Enumerator	CPU module side			Motion module side
	Ladder	FBD/LD	ST	ST
INT type global label	×	×	×	○*1
Constant	○	○	○	○

*1 Use Motion control setting function version "1.010L" or later.

Ex.

When using MC_BUFFER_MODE type enumerator "mcBuffered" on the engineering tool

· When using an INT type global label: Set "MC_BUFFER_MODE__mcBuffered".

· When using with a constant: Set "1".

MC_BUFFER_MODE

Enumerator	Setting value	Description
mcAborting	0	Aborting
mcBuffered	1	Buffered
mcBlendingLow	2	BlendingLow
mcBlendingPrevious	3	BlendingPrevious
mcBlendingNext	4	BlendingNext
mcBlendingHigh	5	BlendingHigh

MC_EXECUTION_MODE

Enumerator	Setting value	Description
mcImmediately	0	Execute immediately
mcQueued	1	Execute at completion
mcNextExecute	2	Execute at next start
mcSpeculatively	3	Execute speculatively

MC_COMBINE_MODE

Enumerator	Setting value	Description
mcAddAxes	0	Add positions of 2 input axes
mcSubAxes	1	Subtract positions of 2 input axes

MC_DIRECTION

Enumerator	Setting value	Description
mcPositiveDirection	1	Positive direction
mcNegativeDirection	2	Negative direction
mcShortestWay	3	Shortest path
mcCurrentDirection	4	Current direction

MC_SOURCE

Enumerator	Setting value	Description
mcSetValue	1	Set value
mcActualValue	2	Actual value
mcLatestSetValue	101	Latest set value
mcLatestActualValue	102	Latest actual value

MC_CIRC_MODE

Enumerator	Setting value	Description
mcBorder	0	Border point specification
mcCenter	1	Center point specification
mcRadius	2	Radius specification

MC_CIRC_PATHCHOICE

Enumerator	Setting value	Description
mcCW	0	CW
mcCCW	1	CCW
mcShortWay	2	Shortcut
mcLongWay	3	Detour
mcCWLongWay	4	CW detour
mcCCWLongWay	5	CCW detour

MC_START_MODE

Enumerator	Setting value	Description
mcImmediate	0	Immediate
mcAbsolute	1	Absolute
mcRelative	2	Relative

MC_AXIS_STATUS

Enumerator	Setting value	Description
Invalid	-1	Axis variable not initialized/axis parameter error
Disabled	0	Axis disabled
ErrorStop	1	Stopping on error
Stopping	2	Decelerating to stop
Homing	3	During home position return
Standstill	4	Standby
DiscreteMotion	5	During positioning operation
ContinuousMotion	6	During continuous operation
SynchronizedMotion	7	During synchronous operation

MC_AXES_GROUP_STATUS

Enumerator	Setting value	Description
Invalid	-1	Axes group variable not initialized/axes group parameter error
GroupDisabled	0	Axes group disabled
GroupErrorStop	1	Stopping on error
GroupStopping	2	Decelerating to stop
GroupHoming	3	Reserve
GroupStandby	4	Standby
GroupMoving	5	Operating

MC_DRIVE_MODE

Enumerator	Setting value	Description
NoModeChange	0	Do not change
pp	1	Reserve
vl	2	Reserve
pv	3	Reserve
tq	4	Reserve
Reserved	5	Reserve
hm	6	hm (Home position return)
ip	7	Reserve
csp	8	csp (Position control)
csv	9	csv (Speed control)
cst	10	cst (Torque control)
cstca	11	Reserve
ct	-104	ct (Continuous operation to torque control)

MC_VELOCITY_LIMIT_MODE

Enumerator	Setting value	Description
Ignore	0	Ignore
ClampWithRamp	1	Clamp
Truncate	2	Truncate
ImmediateStop	3	Immediate stop
ClampWithoutRamp	4	Clamp (Without ramp at deceleration)

MC_INTERPOLATE_SPEED_MODE

Enumerator	Setting value	Description
VectorSpeed	0	Vector speed
LongAxisSpeed	1	Long axis speed
ReferenceAxisSpeed	2	Reference axis speed

MC_SYNTHESIZING_MODE

Enumerator	Setting value	Description
Addition	0	Addition
Substitution	1	Substitution
Combine	2	Combination
Interpolation	3	Interpolation

MC_PROFILE_COMBINE_MODE

Enumerator	Setting value	Description
Relative	0	Relative
InputRelative	1	Input relative
OutputRelative	2	Output relative
Absolute	3	Absolute

MC_RECORD_MODE

Enumerator	Setting value	Description
OneShot	0	One shot mode
RecordCount	1	Frequency specified mode
RingBuffer	2	Ring buffer mode

MC_CLUTCH_MODE

Enumerator	Setting value	Description
Exponent	0	Exponent
Linear	1	Linear
LinearInputFollow	2	Linear (Input follow-up)

MC_CAM_CURVE_TYPE

Enumerator	Setting value	Description
ConstantSpeed	0	Constant speed
ConstantAcceleration	1	Constant acceleration
DistortedTrapezoid	2	Distorted trapezoid
DistortedSine	3	Distorted sine
DistortedConstantSpeed	4	Distorted constant speed
Cycloid	5	Cycloid
FifthCurve	6	5th curve
Trapecloid	7	Trapecloid
ReverseTrapecloid	8	Reverse trapecloid
DoubleHypotenuse	9	Double hypotenuse
ReverseDoubleHypotenuse	10	Reverse double hypotenuse
SingleHypotenuse	11	Single hypotenuse
FifthCurve_SpeedDesignation	12	5th curve (Adjustment)

MC_AXIS_TYPE

Enumerator	Setting value	Description
DriveAxis	0	Real drive axis
EncoderAxis	2	Real encoder axis
VirtualDriveAxis	3	Virtual drive axis
VirtualEncoderAxis	4	Virtual encoder axis
VirtualLinkAxis	5	Virtual link axis

MC_ENCODER_AXIS_TYPE

Enumerator	Setting value	Description
IoModule	0	Via I/O device
Drive	1	Via drive module

MC_ABS_SYSTEM

Enumerator	Setting value	Description
ABSDisabled	0	Disable absolute position system
Enabled	1	Enable absolute position system
Auto	-1	Automatic setting (Acquire from connected device)

MC_DRIVE_STATE

Enumerator	Setting value	Description
NotReadyToSwitchOn	0	Not ready to switch on
Fault	1	Fault
FaultReactionActive	2	Fault reaction active
SwitchOnDisabled	3	Switch on disabled
ReadyToSwitchOn	4	Ready to switch on
SwitchedOn	5	Switched on
OperationEnable	6	Operation enable
QuickStopActive	7	Quick stop active
Invalid	-1	Invalid

MC_DECEL_STOP_MODE

Enumerator	Setting value	Description
Ignore	0	Ignore
ImmediateStop	1	Immediate stop
KeepCurrentAcc	2	Keep current acceleration/deceleration
AlternativeAcc	3	Alternative acceleration/deceleration
ServoOffAfterImmediateStop	4	Servo OFF after immediate stop
ServoOffAfterDecelStop	5	Servo OFF after deceleration to stop

MC_STOP_CURVE_MODE

Enumerator	Setting value	Description
RapidCurve	0	Recreate deceleration curve at increased deceleration
OverrideCurve	1	Recreate deceleration curve
ContinueCurve	2	Maintain deceleration curve

MC_POS_SOURCE

Enumerator	Setting value	Description
Invalid	-1	Invalid
SetPosition	1	Set position
CumulativePosition	2	Cumulative current position
FeedMachinePosition	3	Feed machine position
CommandedPosition	4	Specified position
ActualPosition	5	Feedback position

MC_ACC_ZERO_MODE

Enumerator	Setting value	Description
ACCError	-1	Error (Not started)
MaximumAcceleration	1	Maximum acceleration/deceleration

MC_SIGNAL_LOGIC

Enumerator	Setting value	Description
HighLevel	0	Detection at TRUE
LowLevel	1	Detection at FALSE
RisingEdge	2	Detection at FALSE→TRUE (rising edge)
FallingEdge	3	Detection at TRUE→FALSE (falling edge)
BothEdges	4	Detection at rising edge/falling edge

MC_STATION_REFRESH_MODE

Enumerator	Setting value	Description
EmphasisResponse	0	Response preferred method
EmphasisOperationCycle	1	Operation cycle preferred method

MC_ERROR_CLASS

Enumerator	Setting value	Description
_None	0	None
Warning	1	Warning
MinorError	2	Minor error
ModerateError	3	Moderate error
MajorError	4	Major error

MC_LOGGING_TYPE

Enumerator	Setting value	Description
_None	0	None
ContinuousLogging	1	Continuous logging
TriggerLogging	2	Trigger logging

MC_LOGGING_SAVE_STATUS

Enumerator	Setting value	Description
UnSave	0	Unsaved
Saving	1	Saving
Saved	2	Saved
FullySaved	3	All files saved

MC_EXECUTION_STATE

Enumerator	Setting value	Description
Ready	0	Waiting execution request
Executing	1	Executing
Done	2	Execution completed
Error_	-1	Error occurrence

MC_ACTIVATION_CMD

Enumerator	Setting value	Description
_None	0	None
Authorize	1	Authorize activation key
SetSuperPassword	10	Register supervisor password
SetKey	12	Register activation key
VerifyKey	13	Verify activation key
Initialize	-1	Initialize

MC_ACTIVATION_STATE

Enumerator	Setting value	Description
NoRegisted	0	Not registered
Authorized	1	Authentication completed
SuperPasswordValid	10	Supervisor password registered
UnAuthorized	12	Activation key registered (Waiting authentication)
VerifyDone	13	Verification completed
Lockout	-1	Lockout

MC_ADDON_LIMIT_CMD

Enumerator	Setting value	Description
_None	0	None
Regist	1	Register
TemporaryNoLimitation	10	Temporarily release
Initialize	-1	Initialize

MC_ADDON_LIMIT_STATE

Enumerator	Setting value	Description
NoLimitation	0	Not restricted
Limitation	1	Restricted
TemporaryNoLimitation	10	Released
Lockout	-1	Lockout

MC_SWITCHING_REQUEST

Enumerator	Setting value	Description
NoRequest	0	No request
Enable_	1	Enable
Disable	2	Disable

MC_POS_RESTORATION_STATUS

Enumerator	Setting value	Description
NotExecute	0	Not executed
WaitingRequest	1	Waiting restoration request
RestoredInIncSystem	2	Restoration completed by incremental system
RestoredInAbsSystemUnHomed	3	Restored in absolute position system (Home position return uncompleted)
RestoredInAbsSystem	4	Restored in absolute position system

MC_TRANSITION_MODE

Enumerator	Setting value	Description
TMNone	0	Standard

2.3 List of Motion Control FBs

The list of Motion control FBs is shown below.

Management FBs

The list of management Motion control FBs is shown below.

Motion control FB	Name	Description	Reference
MC_GroupEnable	Axes Group Enabled	Transits the specified axes group status from "0: Axes group disabled (GroupDisabled)" to "4: Standby (GroupStandby)".	☞ Page 85 MC_GroupEnable
MC_GroupDisable	Axes Group Disabled	Transits the specified axes group status to "0: Axes group disabled (GroupDisabled)".	☞ Page 88 MC_GroupDisable
MC_Power	Operation Available	Switches a specified axis to the operation possible status.	☞ Page 91 MC_Power
MC_SetPosition	Current Position Change	Changes the current position (commanded position, actual position) of the specified axis.	☞ Page 95 MC_SetPosition
MCv_SetTorqueLimit	Torque Limit Value	Executes a torque limit value change.	☞ Page 102 MCv_SetTorqueLimit
MC_SetOverride	Override Value Setting	Changes the target velocity, the target acceleration, and the target deceleration of the specified axis.	☞ Page 110 MC_SetOverride
MC_ReadParameter	Parameter Read	Reads objects of the slave devices.	☞ Page 114 MC_ReadParameter
MC_WriteParameter	Parameter Write	Writes objects of the slave devices.	☞ Page 120 MC_WriteParameter
MC_Reset	Axis Error Reset	Resets errors and warnings of the axis.	☞ Page 126 MC_Reset
MC_GroupReset	Axes Group Error Reset	Resets errors and warnings of the axes group and each axis in the axes group.	☞ Page 129 MC_GroupReset
MC_TouchProbe	Touch Probe Enabled	Records optional data when the trigger event occurs.	☞ Page 133 MC_TouchProbe
MC_AbortTrigger	Touch Probe Disabled	Disables the latch that is being executed.	☞ Page 147 MC_AbortTrigger
MC_CamTableSelect	Cam Table Selection	Stores the specified calculation profile (cam data) in the open area.	☞ Page 153 MC_CamTableSelect
MCv_ChangeCycle	Current Value Change per Cycle	Changes the current value per cycle of the specified calculation profile control FB.	☞ Page 160 MCv_ChangeCycle
MCv_AllPower	All Axes Operation Available	Switches every axis to the operation possible status.	☞ Page 170 MCv_AllPower
MC_GroupSetOverride	Axes Group Override Value Setting	Changes the target velocity, the target acceleration, and the target deceleration of the specified axes group.	☞ Page 174 MC_GroupSetOverride
MCv_MotionErrorReset	Motion Error Reset	Resets all errors and warnings of the Motion system.	☞ Page 178 MCv_MotionErrorReset

Operation FBs

The list of operation Motion control FBs is shown below.

Motion control FB	Name	Description	Name
MC_Home	OPR	Executes homing for the specified axis.	☞ Page 182 MC_Home
MC_Stop	Forced Stop	Decelerates the specified axis to a stop.	☞ Page 190 MC_Stop
MC_GroupStop	Group Forced Stop	Decelerates the specified axes group to a stop.	☞ Page 195 MC_GroupStop
MC_MoveAbsolute	Absolute Value Positioning	Executes positioning after setting the target position of the absolute position.	☞ Page 200 MC_MoveAbsolute
MC_MoveRelative	Relative Value Positioning	Executes positioning after setting the movement amount of the relative position.	☞ Page 212 MC_MoveRelative
MCv_Jog	JOG	Executes JOG operation according to the target velocity.	☞ Page 220 MCv_Jog
MC_MoveVelocity	Speed Control	Switches the driver to csv, then performs velocity control according to the specified velocity.	☞ Page 229 MC_MoveVelocity
MC_TorqueControl	Torque Control	Switches the driver to cst, then performs torque control according to the specified target torque.	☞ Page 236 MC_TorqueControl
MCv_SpeedControl	Speed Control (Including Position Loop)	Executes speed control including the position loop.	☞ Page 246 MCv_SpeedControl
MCv_MoveLinearInterpolateAbsolute	Absolute Value Linear Interpolation Control	Specifies the target position of the absolute position of the set axes group, then executes positioning by linear interpolation control.	☞ Page 254 MCv_MoveLinearInterpolateAbsolute
MCv_MoveLinearInterpolateRelative	Relative Value Linear Interpolation Control	Specifies the movement amount of the relative position of the set axes group, then executes positioning by linear interpolation control.	☞ Page 270 MCv_MoveLinearInterpolateRelative
MCv_MoveCircularInterpolateAbsolute	Absolute Value Circular Interpolation Control	Executes positioning with 2-axis circular interpolation using the structuring axes of the set axes group by setting the end point and sub point of the absolute position.	☞ Page 285 MCv_MoveCircularInterpolateAbsolute
MCv_MoveCircularInterpolateRelative	Relative Value Circular Interpolation Control	Executes positioning with 2-axis circular interpolation using the structuring axes of the set axes group by setting the relative position from the current position at start to the end point and sub point.	☞ Page 302 MCv_MoveCircularInterpolateRelative
MC_CamIn	Cam Operation Start	Starts cam operation according to the specified cam data.	☞ Page 321 MC_CamIn
MC_GearIn	Gear Operation Start	Starts gear operation according to the specified gear ratio.	☞ Page 347 MC_GearIn
MC_CombineAxes	Addition/Subtraction Positioning	Adds or subtracts the movement amounts of the two specified master axes, then performs positioning using said value as the commanded position.	☞ Page 357 MC_CombineAxes
MCv_BacklashCompensationFilter	Backlash Compensation Filter	Performs filter processing according to the movement direction to compensate the backlash amount in the machine system.	☞ Page 369 MCv_BacklashCompensationFilter
MCv_SmoothingFilter	Smoothing Filter	Performs filter processing according to the specified frequency.	☞ Page 382 MCv_SmoothingFilter
MCv_DirectionFilter	Moving Direction Restriction Filter	Performs filter processing to restrict traveling for the set movement direction.	☞ Page 391 MCv_DirectionFilter
MCv_SpeedLimitFilter	Speed Limit Filter	Performs filter processing to restrict to the velocity of the set limit value.	☞ Page 399 MCv_SpeedLimitFilter

Standard FBs

The list of Motion control FBs used to perform axis control is shown below.

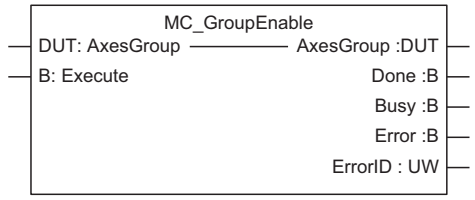
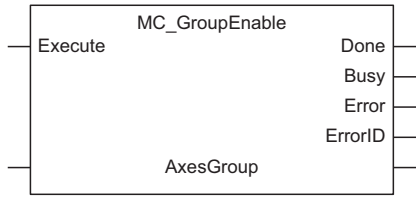
Motion control FB	Name	Description	Reference
MCv_ReadProfileData	Profile Read	Reads the specified calculation profile data from the open area or file.	☞ Page 407 MCv_ReadProfileData
MCv_WriteProfileData	Profile Write	Writes the specified calculation profile data to the open area or file.	☞ Page 419 MCv_WriteProfileData

3 MOTION CONTROL FB

3.1 Management FBs

MC_GroupEnable

Transits the specified Axes group status (AxesGroupName.Md.GroupStatus) from "0: Axes group disabled (GroupDisabled)" to "4: Standby (GroupStandby)".

Ladder	FBD/LD	ST
		MC_GroupEnable(AxesGroup:= ?AXES_GROUP_REF? , Execute:= ?BOOL? , Done=> ?BOOL? , Busy=> ?BOOL? , Error=> ?BOOL? , ErrorID=> ?WORD?);

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Axes Group Enabled	9	5	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
AxesGroup	Axes group information	AXES_GROUP_REF	At start	—	Omission not possible	This variable sets the axes group. For the variables used (AxesGroupName.AxesGroupRef.), refer to the following. ☞ Page 46 AxesGroupName.AxesGroupRef. (Axes group information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Execute command	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MC_GroupEnable (Axes Group Enabled).

■Output variables

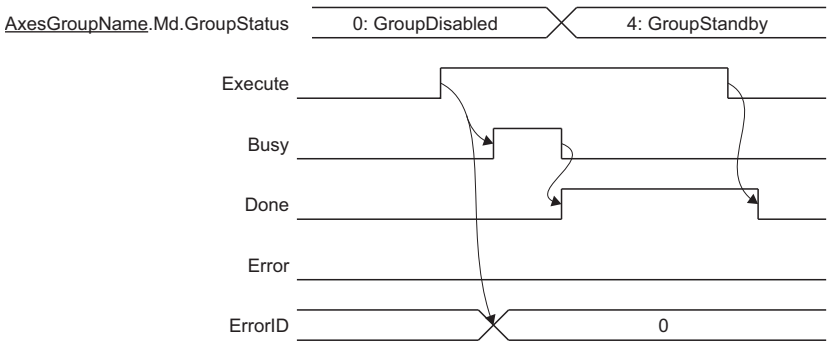
Output variable	Name	Data type	Default value	Description
Done	Execution completion	BOOL	FALSE	This variable becomes TRUE when the axes group status transits to "4: Standby (GroupStandby)".
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MC_GroupEnable (Axes Group Enabled) is executed.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. 📖MELSEC iQ-R Motion Module User's Manual (Application)

Processing details

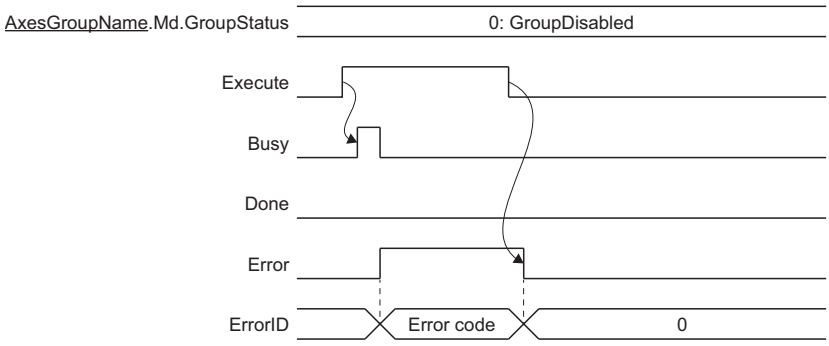
- Transits Axes group status (AxesGroupName.Md.GroupStatus) of the AXES_GROUP_REF structure from "0: Axes group disabled (GroupDisabled)" to "4: Standby (GroupStandby)".
- MC_GroupEnable (Axes Group Enabled) is executed when Execute command (Execute) becomes TRUE, and Executing (Busy) becomes TRUE when the processing is started normally.
- When the processing completes normally, Execution completion (Done) becomes TRUE and Executing (Busy) becomes FALSE.
- When an error occurs in MC_GroupEnable (Axes Group Enabled), Error (Error) becomes TRUE and the error code is stored in Error code (ErrorID). For details of error codes, refer to the following.
  MELSEC iQ-R Motion Module User's Manual (Application)
- For the axes group state transition, refer to the following.
  MELSEC iQ-R Motion Module User's Manual (Application)
- MC_GroupEnable (Axes Group Enabled) can only be executed when Axis status (AxisName.Md.AxisStatus) of all structuring axes is "4: Standby (Standstill)" or "0: Axis disabled (Disabled)".
- Using axes group (AxisName.Md.UseInGroup) becomes TRUE when Axes group status (AxesGroupName.Md.GroupStatus) changes to "4: Standby (GroupStandby)".
- When executing MC_GroupEnable (Axes Group Enabled) with another axes group set that includes a structuring axis for which Using axes group (AxisName.Md.UseInGroup) is TRUE, "Axes Group Configuration Axis Is in Use (error code: 3496H)" occurs.
- If the error occurs in any of the structuring axes, Axes group status (AxesGroupName.Md.GroupStatus) changes to "1: Stopping on error (GroupErrorStop)".
- When MC_GroupEnable (Axes Group Enabled) is executed for an axes group that has already been enabled, Execution completion (Done) becomes TRUE and the execution ends.

■Timing chart

- When the operation completes normally



- When an error occurs



Precautions

- Axes group No. (GroupNo) of the AXES_GROUP_REF structure must be set in advance before MC_GroupEnable (Axes Group Enabled) is executed.
- MC_GroupEnable (Axes Group Enabled) can only be executed when Axis status (AxisName.Md.AxisStatus) of all structuring axes set in the axes group is "4: Standby (Standstill)" or "0: Axis disabled (Disabled)".
- If a parameter error occurs at axes group variable initialization, "Out of Parameter Range (Axes Group) (error code: 1D81H)" occurs. In this case, Axes group status (AxesGroupName.Md.GroupStatus) changes to "-1: Axes group variable not initialized/Axes group parameter error (Invalid)". Refreshing of the monitor data, etc. is not executed for axes groups whose status is " -1: Axes group variable not initialized/Axes group parameter error (Invalid)". If the axes group is specified by the user program, "Out of Axes Group No. Range (error code: 3402H)" occurs.

Program example

In the following program example, the axes group enable command (bGroupEnable) is set to TRUE and Axes group 1 (AxesGroup001) is enabled.

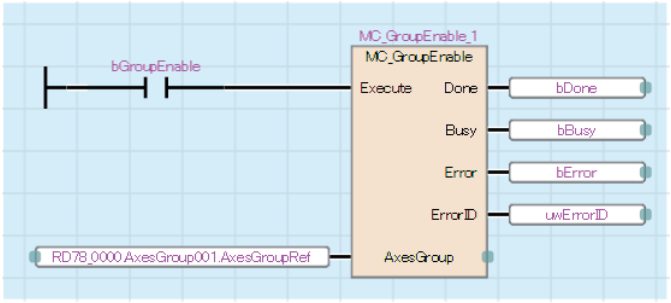
■Axes group

Axes group No.	Label name	Data type	Comment
1	AxesGroup001	AXES_GROUP_REF	Axes group 1

■Labels used

Label name	Data type	Comment
MC_GroupEnable_1	MC_GroupEnable	Axes Group Enabled FB
bGroupEnable	Bit	Axes group enable command
bDone	Bit	Execution completion
bBusy	Bit	Executing
bError	Bit	Error
uwErrorID	Word [unsigned]/bit string [16-bit]	Error code

■FBD/LD program (CPU module side)

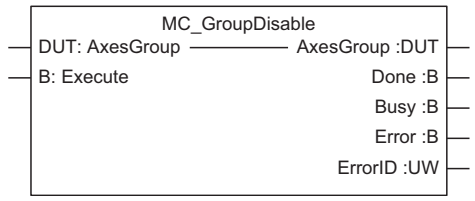
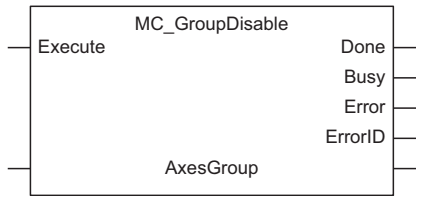


■ST program (Motion module side)

```
MC_GroupEnable_1(
  AxesGroup:= AxesGroup001.AxesGroupRef ,
  Execute:= bGroupEnable ,
  Done=> bDone ,
  Busy=> bBusy ,
  Error=> bError ,
  ErrorID=> uwErrorID
);
```

MC_GroupDisable

Transits the specified Axes group status (AxesGroupName.Md.GroupStatus) to "0: Axes group disabled (GroupDisabled)" and disables the axes group.

Ladder	FBD/LD	ST
		MC_GroupDisable(AxesGroup:= ?AXES_GROUP_REF? , Execute:= ?BOOL? , Done=> ?BOOL? , Busy=> ?BOOL? , Error=> ?BOOL? , ErrorID=> ?WORD?);

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Axes Group Disabled	9	5	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
AxesGroup	Axes group information	AXES_GROUP_REF	At start	—	Omission not possible	This variable sets the axes group. For the variables used (AxesGroupName.AxesGroupRef.), refer to the following. ☞ Page 46 AxesGroupName.AxesGroupRef. (Axes group information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Execute command	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MC_GroupDisable (Axes Group Disabled).

■Output variables

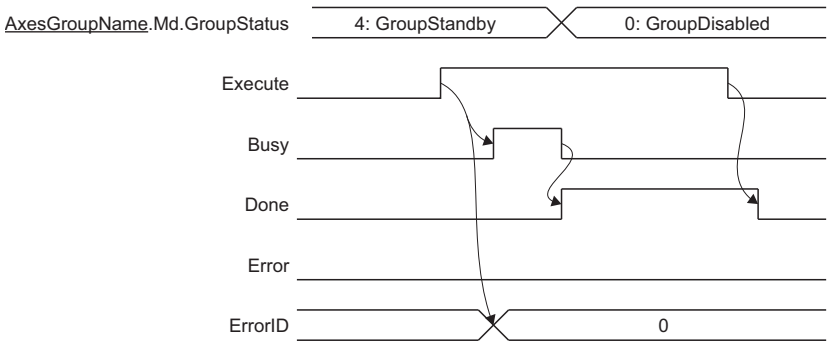
Output variable	Name	Data type	Default value	Description
Done	Execution completion	BOOL	FALSE	This variable becomes TRUE when the axes group status transits to "0: Axes group disabled (GroupDisabled)".
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MC_GroupDisable (Axes Group Disabled) is executed.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. 📖MELSEC iQ-R Motion Module User's Manual (Application)

Processing details

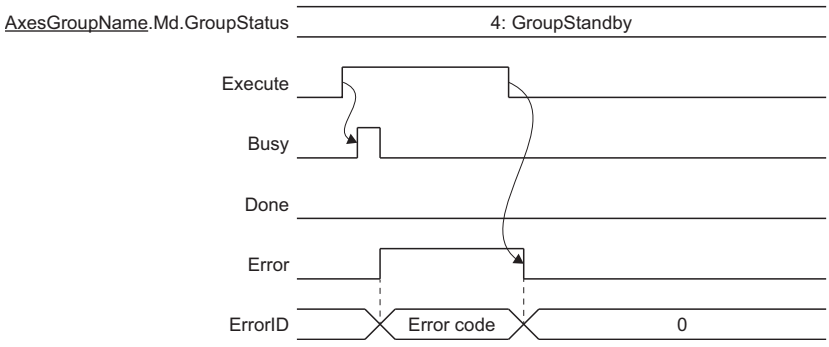
- Transits Axes group status (AxesGroupName.Md.GroupStatus) of the AXES_GROUP_REF structure to "0: Axes group disabled (GroupDisabled)".
- MC_GroupDisable (Axes Group Disabled) is executed when Execute command (Execute) becomes TRUE, and Executing (Busy) becomes TRUE when the processing is started normally.
- When the processing completes normally, Execution completion (Done) becomes TRUE and Executing (Busy) becomes FALSE.
- When an error occurs in MC_GroupDisable (Axes Group Disabled), Error (Error) becomes TRUE and the error code is stored in Error code (ErrorID). For details of error codes, refer to the following.
  MELSEC iQ-R Motion Module User's Manual (Application)
- For the axes group state transition, refer to the following.
  MELSEC iQ-R Motion Module User's Manual (Application)
- MC_GroupDisable (Axes Group Disabled) can only be executed when Axes group status (AxesGroupName.Md.AxisStatus) is "4: Standby (GroupStandby)" or "1: Stopping on error (GroupErrorStop)". When executing MC_GroupDisable (Axes Group Disabled) while the axes group status is "1: Stopping on error (GroupErrorStop)" and the axes group is in operation, the axes group status transits to "0: Axes group disabled (GroupDisabled)" after the stop completes. When MC_GroupDisable (Axes Group Disabled) is executed while Axes group status (AxesGroupName.Md.GroupStatus) is other than "4: Standby (GroupStandby)" or "1: Stopping on error (GroupErrorStop)", "Axes Group Status Incorrect (Axes Group Disable) (error code: 1A97H)" occurs and the FB being executed stops.
- When Axes group status (AxesGroupName.Md.GroupStatus) changes to "0: Axes group disabled (GroupDisabled)", Using axes group (AxisName.Md.UseInGroup) of the structuring axis becomes FALSE.
- When MC_GroupDisable (Axes Group Disabled) is executed for an axes group that has already been disabled, Execution completion (Done) becomes TRUE and the execution ends.

■Timing chart

- When the operation completes normally



- When an error occurs



Precautions

- Axes group No. (GroupNo) of the AXES_GROUP_REF structure must be set in advance before MC_GroupDisable (Axes Group Disabled) is executed.
- If a parameter error occurs at axes group variable initialization, "Out of Parameter Range (Axes Group) (error code: 1D81H)" occurs. In this case, Axes group status (AxesGroupName.Md.GroupStatus) changes to "-1: Axes group variable not initialized/Axes group parameter error (Invalid)". Refreshing of the monitor data, etc. is not executed for axes groups whose status is " -1: Axes group variable not initialized/Axes group parameter error (Invalid)". If the axes group is specified by the user program, "Out of Axes Group No. Range (error code: 3402H)" occurs.

Program example

In the following program example, the axes group disable command (bGroupDisable) is set to TRUE and Axes group 1 (AxesGroup001) is disabled.

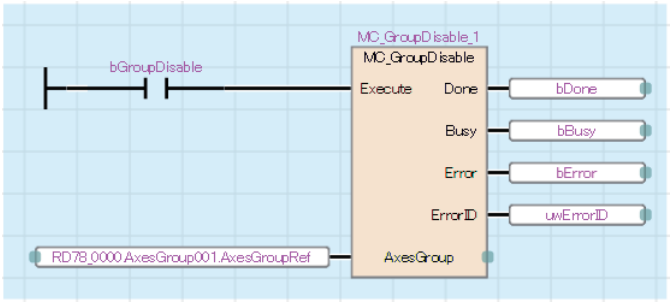
■Axes group

Axes group No.	Label name	Data type	Comment
1	AxesGroup001	AXES_GROUP_REF	Axes group 1

■Labels used

Label name	Data type	Comment
MC_GroupDisable_1	MC_GroupDisable	Axes Group Disabled FB
bGroupDisable	Bit	Axes group disable command
bDone	Bit	Execution completion
bBusy	Bit	Executing
bError	Bit	Error
uwErrorID	Word [unsigned]/bit string [16-bit]	Error code

■FBD/LD program (CPU module side)

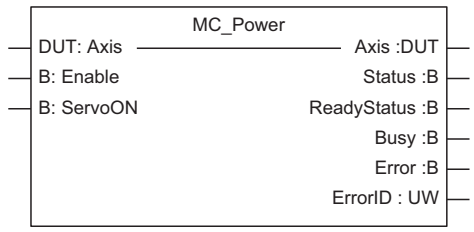
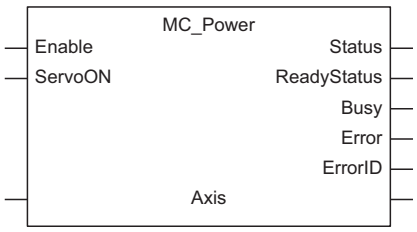


■ST program (Motion module side)

```
MC_GroupDisable_1(
  AxesGroup:= AxesGroup001.AxesGroupRef ,
  Execute:= bGroupDisable ,
  Done=> bDone ,
  Busy=> bBusy ,
  Error=> bError ,
  ErrorID=> uwErrorID
);
```

MC_Power

Switches a specified axis to the operation possible status.

Ladder	FBD/LD	ST
		MC_Power(Axis:= ?AXIS_REF? , Enable:= ?BOOL? , ServoON:= ?BOOL? , Status=> ?BOOL? , ReadyStatus=> ?BOOL? , Busy=> ?BOOL? , Error=> ?BOOL? , ErrorID=> ?WORD?);

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Operation Available	10	6	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
Axis	Axis information	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information) *: The axis information is established when the FB is called for the first time after switching “PLC READY (Y0)” to ON.

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Enable	Enable	BOOL	Always	TRUE, FALSE	FALSE	When this variable is set to TRUE, the axis control is valid and the axis becomes operation possible status. When this variable is set to FALSE, the axis control is invalid and the operation possible status is cancelled for the axis.
ServoON	Servo ON request	BOOL	Always	TRUE, FALSE	FALSE	When this variable is TRUE, specifies the servo ON request.

■Output variables

Output variable	Name	Data type	Default value	Description
Status	Operable	BOOL	FALSE	This variable becomes TRUE when the axis becomes operation possible status.
ReadyStatus	Ready ON status	BOOL	FALSE	This variable becomes TRUE when Driver READY turns ON.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MC_Power (Operation Available) is executed.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. MELSEC iQ-R Motion Module User’s Manual (Application)

Processing details

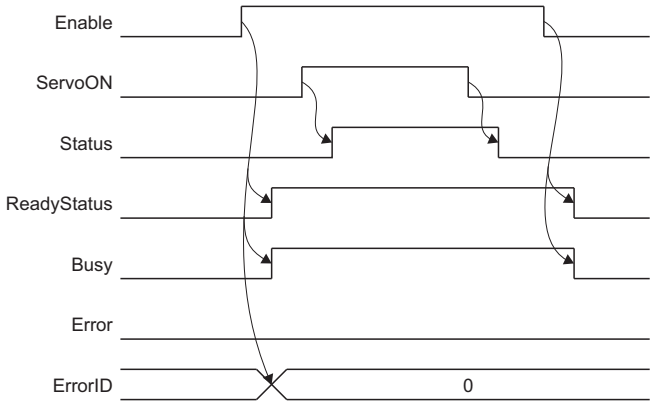
- Initializes the information of the specified axis and switches the axis status to the operation possible status.
- When the inputs for Enable (Enable) and Servo ON request (ServoON) are TRUE, the selected axis is switched to the operation possible status.
- Executing (Busy) becomes TRUE when the processing is started.
- Operable (Status) becomes TRUE when the drive unit has been switched to the operation possible status.
- When an error occurs in MC_Power (Operation Available), Error (Error) becomes TRUE and the error code is stored in Error code (ErrorID). For details of error codes, refer to the following.
  MELSEC iQ-R Motion Module User's Manual (Application)
- Axis status (AxisName.Md.AxisStatus) of the AXIS_MONI structure transits from "0: Axis disabled (Disabled)" to "4: Standby (Standstill)". For the axis state transition, refer to the following.
  MELSEC iQ-R Motion Module User's Manual (Application)
- Inputting Enable (Enable) and Servo ON request (ServoON) switches the servo ON/OFF status and the driver status of the axis as shown below.

Input variables		Output variables		Servo ON/OFF status	Driver status (AxisName.Md.Driver_State)
Enable (Enable)	Servo ON request (ServoON)	Ready ON status (ReadyStatus)	Operable (Status)		
TRUE	TRUE	TRUE	TRUE	Servo ON	6: Operation Enable
	FALSE	TRUE	FALSE	Servo OFF	5: Switched On
FALSE	TRUE	FALSE	FALSE	Servo OFF	3: Switch On Disabled
	FALSE	FALSE	FALSE	Servo OFF	3: Switch On Disabled

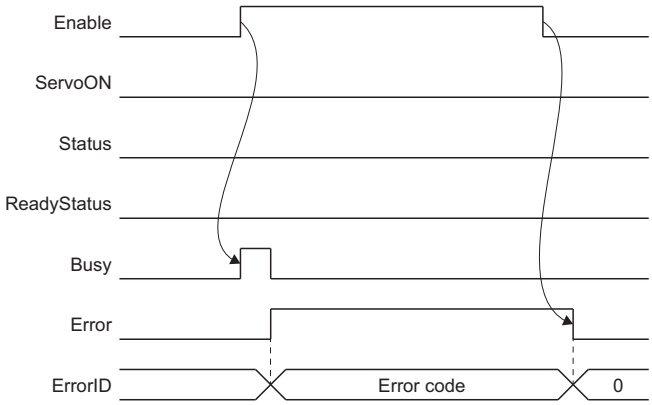
- If a real axis is rotated by external force during servo OFF, follow up processing is performed.
- The servo ON/OFF control can be operated regardless of the control mode. The control mode used during servo OFF depends on the specification of the driver.
- MC_Power (Operation Available) is sent to the driver while a drive unit error is occurring, so Enable (Enable) and Servo ON request (ServoON) do not need to be turned from FALSE to TRUE again.

■Timing chart

- When the operation completes normally



- When an error occurs



Precautions

- Axis No. (AxisNo) of the AXIS_REF structure must be set in advance before MC_Power (Operation Available) is executed.
- MC_Power (Operation Available) establishes the axis information the first time that the FB is called after "PLC READY (Y0)" is switched to ON. Changes made to the axis information while Enable (Enable) is FALSE are not reflected.
- Do not set two or more MC_Power (Operation Available) for one axis. The operation is not guaranteed if 2 or more of this FB are set.
- If MC_Power (Operation Available) is executed to the virtual axis, both Operable (Status) and Ready ON status (ReadyStatus) become TRUE regardless of the Servo ON request (ServoON) input.
- The servo ON/OFF status when stopping the program is maintained while "Ready (X0)" is OFF.

Program example

In the following program example, the servo ON request command (bServoON) is set to TRUE and Axis 1 (Axis0001) is changed to operation possible status.

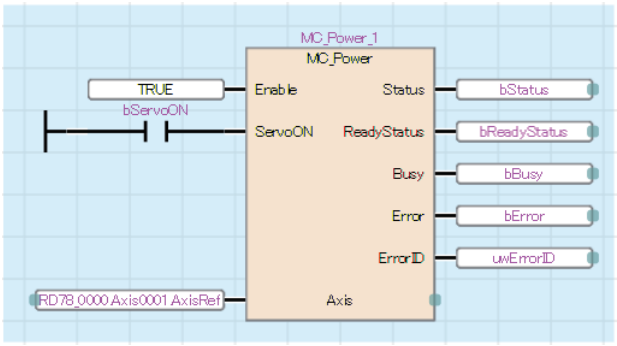
■Axis

Axis No.	Label name	Data type	Comment
1	Axis0001	AXIS_REF	Axis 1

■Labels used

Label name	Data type	Comment
MC_Power_1	MC_Power	Operation Available FB
bServoON	Bit	Servo ON request command
bStatus	Bit	Operable
bReadyStatus	Bit	Ready ON status
bBusy	Bit	Executing
bError	Bit	Error
uwErrorID	Word [unsigned]/bit string [16-bit]	Error code

■FBD/LD program (CPU module side)



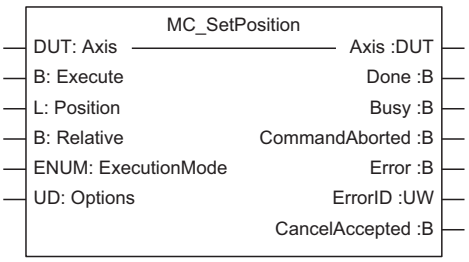
■ST program (Motion module side)

```
MC_Power_1(
  Axis:= Axis0001.AxisRef ,
  Enable:= TRUE ,
  ServoON:= bServoON ,
  Status=> bStatus ,
  ReadyStatus=> bReadyStatus ,
  Busy=> bBusy ,
  Error=> bError ,
  ErrorID=> uwErrorID
);
```

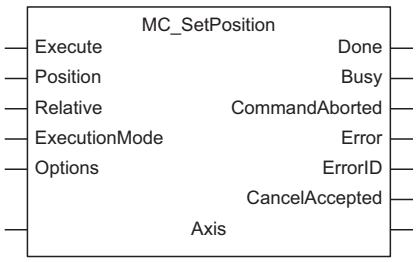
MC_SetPosition

Changes the current position (commanded position, actual position) of the specified axis.

Ladder



FBD/LD



ST

```
MC_SetPosition(
   Axis:= ?AXIS_REF? ,
   Execute:= ?BOOL? ,
   Position:= ?LREAL? ,
   Relative:= ?BOOL? ,
   ExecutionMode:= ?INT? ,
   Options:= ?DWORD? ,
   Done=> ?BOOL? ,
   Busy=> ?BOOL? ,
   CommandAborted=> ?BOOL? ,
   Error=> ?BOOL? ,
   ErrorID=> ?WORD? ,
   CancelAccepted=> ?BOOL?
);
```

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Current Position Change	32	7	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
Axis	Axis information	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Execute command	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MC_SetPosition (Current Position Change).
Position	Target position	LREAL	At start	-10000000000.0 to 10000000000.0	0.0	This variable sets the target position value to change. The setting value changes depending on the setting of Relative position selection (Relative). For details, refer to the following. ☞ Page 97 Target position (Position)
Relative	Relative position selection	BOOL	At start	TRUE, FALSE	FALSE	This variable sets whether to change the current position with the relative distance or the absolute position. • FALSE: Current position change with absolute position • TRUE: Current position change with relative distance
ExecutionMode	Execution mode	INT (MC_EXECUTION_MODE)	At start	1, 3	3	This variable sets the execution method used to execute MC_SetPosition (Current Position Change). • 1: Execute at completion (mcQueued)*1 • 3: Execute speculatively (mcSpeculatively)*2 For details, refer to the following. ☞ Page 97 Execution mode (ExecutionMode)
Options	Options	DWORD(HEX)	At start	00000000H to 00010000H	00000000H	This variable sets the function options for MC_SetPosition (Current Position Change) by specifying the bit. For details, refer to the following. ☞ Page 98 Options (Options)

*1 Executes after the previous FB is completed.

*2 Can only execute when there is no FB being executed on the specified axis.

■Output variables

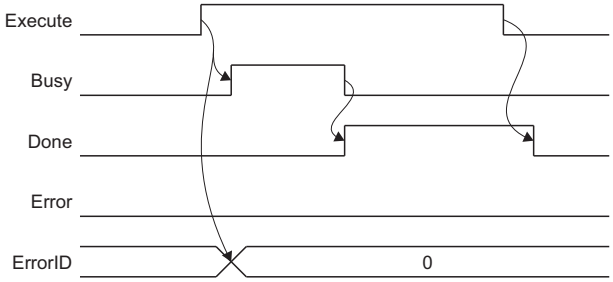
Output variable	Name	Data type	Default value	Description
Done	Execution completion	BOOL	FALSE	This variable becomes TRUE when the current position change is completed.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MC_SetPosition (Current Position Change) is executed.
CommandAborted	Abortion of execution	BOOL	FALSE	This variable becomes TRUE when execution of MC_SetPosition (Current Position Change) is aborted.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. MELSEC iQ-R Motion Module User's Manual (Application)
CancelAccepted	Cancel acceptance	BOOL	FALSE	This variable becomes TRUE when MC_SetPosition (Current Position Change) accepted the cancel.

Processing details

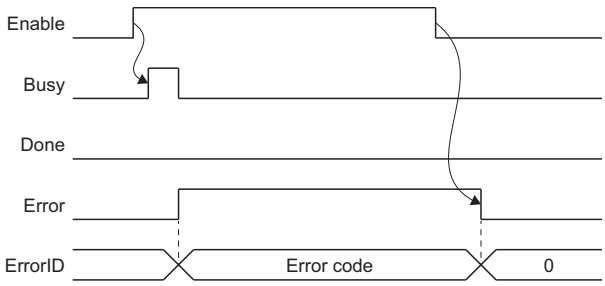
- Changes the current position of the specified axis.
- When Relative position selection (Relative) is TRUE, changes the current position to a position obtained by adding the target position (relative distance) to the current position.
- When Relative position selection (Relative) is FALSE, changes current position to the target position (absolute position).
- MC_SetPosition (Current Position Change) is executed when Execute command (Execute) becomes TRUE, and Executing (Busy) becomes TRUE when the processing is started normally.
- When the processing completes and the current position is changed, Execution completion (Done) becomes TRUE.
- When Axis status (AxisName.Md.AxisStatus) becomes "0: Axis disabled (Disabled)" or "1: Stopping on error (ErrorStop)" before processing completes after executing MC_SetPosition (Current Position Change) with Execution mode (ExecutionMode) set to "1: Execute at completion (mcQueued)", "MC_SetPosition Instruction Error (error code: 344EH)" occurs.
- When an error occurs in MC_SetPosition (Current Position Change), Error (Error) becomes TRUE and the error code is stored in Error code (ErrorID). For details of error codes, refer to the following.
  MELSEC iQ-R Motion Module User's Manual (Application)

■Timing chart

- When the operation completes normally



- When an error occurs



■Target position (Position)

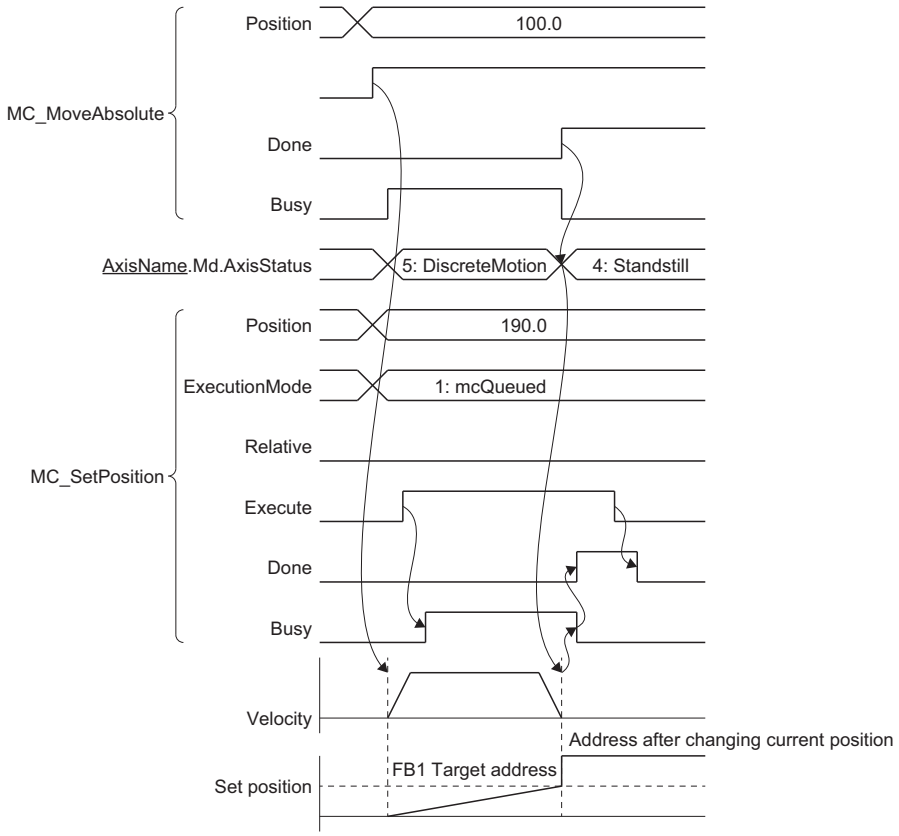
The values that can be set in Target position (Position) differ depending on whether Relative position selection (Relative) is TRUE or FALSE.

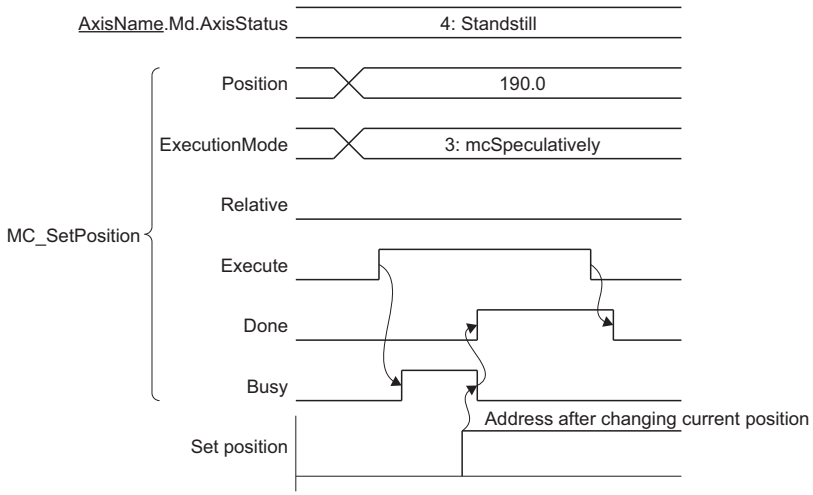
Set Target position (Position) within the following setting range.

Relative position selection (Relative)	Target position (Position) setting range
FALSE (Current position change with absolute position)	■When the ring counter is disabled -10000000000.0 ≤ Setting value < 10000000000.0 ■When the ring counter is enabled Ring counter lower limit value ≤ Setting value < Ring counter upper limit value
TRUE (Current position change with relative distance)	■When the ring counter is disabled -10000000000.0 ≤ Setting value ≤ 10000000000.0 ■When the ring counter is enabled -(Ring counter upper limit value - ring counter lower limit value) / 2 ≤ Setting value ≤ (Ring counter upper limit value - ring counter lower limit value) / 2

■Execution mode (ExecutionMode)

When MC_SetPosition (Current Position Change) is executed during axis operation, the timing for the change depends on the setting of Execution mode (ExecutionMode). The timing for Execution mode (ExecutionMode) is shown below.

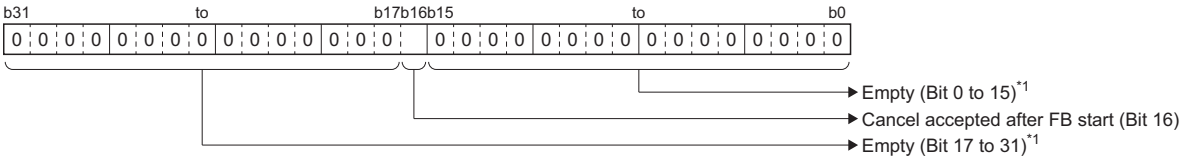
Setting value	Description
1: Execute at completion (mcQueued)	Executes after the FB being executed completes. At rising edge detection of Execute command (Execute), Executing (Busy) becomes TRUE and remains TRUE until Axis status (AxisName.Md.AxisStatus) changes to "4: Standby (Standstill)". Current position change operation is started when Axis status (AxisName.Md.AxisStatus) changes to "4: Standby (Standstill)". 

Setting value	Description
3: Execute speculatively (mcSpeculatively)	When Axis status (AxisName.Md.AxisStatus) of the relevant axis is "4: Standby (Standstill)" at rising edge detection of Execute command (Execute), the current position change can be executed. When Axis status (AxisName.Md.AxisStatus) of the relevant axis is not "4: Standby (Standstill)" or when an FB is being executed, "MC_SetPosition Instruction Error (error code: 344EH)" occurs. 

■Options (Options)

Sets the function options for MC_SetPosition (Current Position Change) by specifying the bit.

The contents set by specifying the bit are shown below.



*1 Set empty areas to "0". When a value other than "0" is set, "Out of Options Range (error code: 1A4EH)" occurs.

Bit	Name	Description
16	Cancel accepted after FB start	Sets whether or not to allow a cancel after MC_SetPosition (Current Position Change) is executed. • 0: Do not allow • 1: Allow

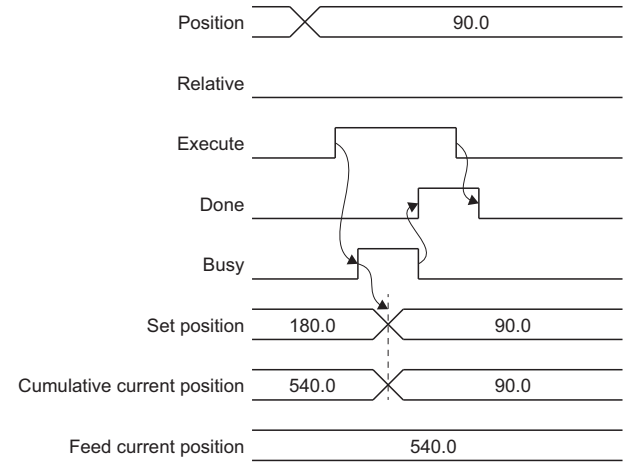
■Operation overview

- When MC_SetPosition (Current Position Change) is executed while the axis is stopped, the current position change operation starts at rising edge detection of Execute command (Execute.) In the current position change, the set position and the cumulative current position are changed to the position set in Target position (Position). Target position (Position) becomes "Current position change with relative distance" when Relative position selection (Relative) is TRUE and "Current position change with absolute position" when Relative position selection (Relative) is FALSE.

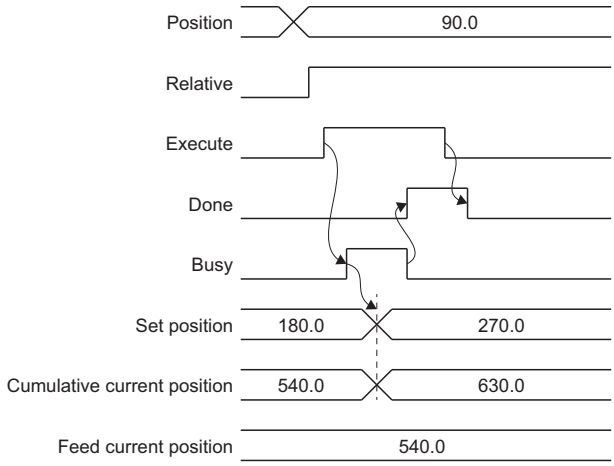
Ex.

When executing the current position change to "90.0" at a stop with a ring counter lower limit value of "0.0", a ring counter upper limit value of "360.0", and a set position of "180.0" (a cumulative current position of "540.0").

<When Relative position selection (Relative) is FALSE (Absolute position)>



<When Relative position selection (Relative) is TRUE (Relative distance)>



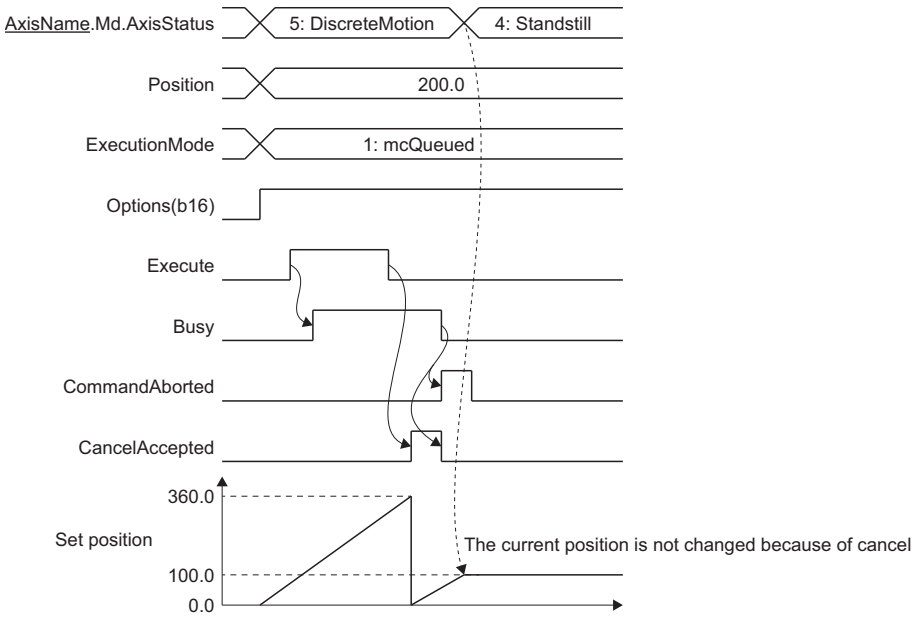
■Cancel

A current position change that is in the standby status can be canceled after MC_SetPosition (Current Position Change) execution.

- To enable cancel, execute MC_SetPosition (Current Position Change) with the status of "Cancel accepted after FB start (Options (Options): Bit 16)" set to "1: Allow".
- Cancel is started at falling edge detection of Execute command (Execute).
- Cancel is accepted only when Executing (Busy) of the output pin is TRUE.
- When MC_SetPosition (Current Position Change) accepts the cancel, Cancel acceptance (CancelAccepted) of the output pin becomes TRUE.
- When cancel is completed, Abortion of execution (CommandAborted) of the output pin becomes TRUE.
- When cancelled, the set position is not changed.

Ex.

After starting with Execution mode (ExecutionMode) set to "1: Execute at completion (mcQueued)", the change is cancelled while Axis Status (AxisName.Md.AxisStatus) is waiting to switch to "4: Standby (Standstill)".



Precautions

- A current position change can be executed during axis operation. Although the target position of the MC_Set Position (Current Position Change) being executed will not change if the current position is changed during axis operation, the operation to the target value will be changed because of the current position being changed.

Program example

In the following program example, the current position change command (bSetPosition) is set to TRUE and the current position of Axis 1 (Axis0001) is changed to "100.0".

■Axis

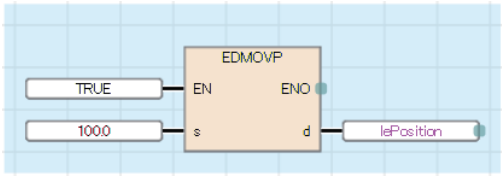
Axis No.	Label name	Data type	Comment
1	Axis0001	AXIS_REF	Axis 1

■Labels used

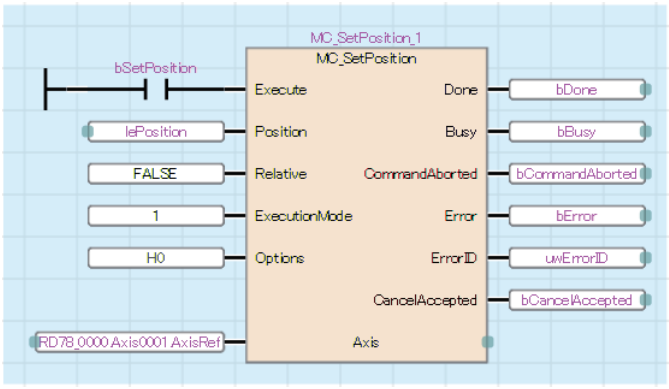
Label name	Data type	Comment
MC_SetPosition_1	MC_SetPosition	Current Position Change FB
bSetPosition	Bit	Current position change command
lePosition	Double-precision real number	Target position
bDone	Bit	Ready
bBusy	Bit	Executing
bCommandAborted	Bit	Abortion of execution
bError	Bit	Error
uwErrorID	Word [unsigned]/bit string [16-bit]	Error code
bCancelAccepted	Bit	Cancel acceptance

■FBD/LD program (CPU module side)

- Data setting for Current Position Change



- Current Position Change



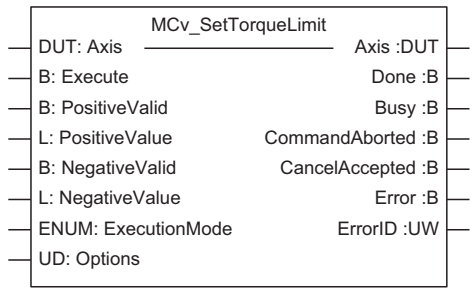
■ST program (Motion module side)

```
//-----Data setting for Current Position Change-----
lePosition:= 100.0;

//-----Current Position Change-----
MC_SetPosition_1(
   Axis:= Axis0001.AxisRef ,
   Execute:= bSetPosition ,
   Position:= lePosition ,
   Relative:= FALSE ,
   ExecutionMode:= MC_EXECUTION_MODE__mcQueued ,
   Options:= H00000000 ,
   Done=> bDone ,
   Busy=> bBusy ,
   CommandAborted=> bCommandAborted ,
   Error=> bError ,
   ErrorID=> uwErrorID ,
   CancelAccepted=> bCancelAccepted
);
```

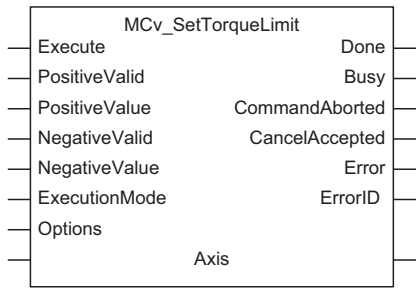
MCv_SetTorqueLimit

Executes a torque limit value change

Ladder



FBD/LD



ST

```
MCv_SetTorqueLimit(
  Axis:= ?AXIS_REF? ,
  Execute:= ?BOOL? ,
  PositiveValid:= ?BOOL? ,
  PositiveValue:= ?LREAL? ,
  NegativeValid:= ?BOOL? ,
  NegativeValue:= ?LREAL? ,
  ExecutionMode:= ?INT? ,
  Options:= ?DWORD? ,
  Done=> ?BOOL? ,
  Busy=> ?BOOL? ,
  CommandAborted=> ?BOOL? ,
  CancelAccepted=> ?BOOL? ,
  Error=> ?BOOL? ,
  ErrorID=> ?WORD?
);
```

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Torque Limit Value	48	8	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
Axis	Axis information	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Execute command	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MCv_SetTorqueLimit (Torque Limit Value).
PositiveValid	Positive direction torque limit valid	BOOL	At start/ Retrigger possible	TRUE, FALSE	FALSE	This variable sets whether to enable or disable the positive direction torque limit setting value change. • FALSE: Disables the positive direction torque limit. • TRUE: Enables the positive direction torque limit.
PositiveValue	Positive direction torque limit value	LREAL	At start/ Retrigger possible	0.0 to 1000.0[%]	0.0	This variable sets the positive direction torque limit value. Imports the value when Positive direction torque limit valid (PositiveValid) is TRUE.
NegativeValid	Negative direction torque limit valid	BOOL	At start/ Retrigger possible	TRUE, FALSE	FALSE	This variable sets whether to enable or disable the negative direction torque limit setting value change. • FALSE: Disables the negative direction torque limit. • TRUE: Enables the negative direction torque limit.
NegativeValue	Negative direction torque limit value	LREAL	At start/ Retrigger possible	0.0 to 1000.0[%]	0.0	This variable sets the negative direction torque limit value. Imports the value when Negative direction torque limit valid (NegativeValid) is TRUE.

Input variable	Name	Data type	Input	Setting range	Default value	Description
ExecutionMode	Execution mode	INT (MC_EXECUTION_MODE)	At start	0, 1, 3	0	This variable sets the timing for executing MCv_SetTorqueLimit (Torque Limit Value). • 0: Execute immediately (mcImmediately) • 1: Execute at completion (mcQueued) • 3: Execute speculatively (mcSpeculatively) For details, refer to the following. ☞ Page 105 Execution mode (ExecutionMode)
Options	Options	DWORD(HEX)	At start	00000000H to 00010000H	00000000H	This variable sets the function options for MCv_SetTorqueLimit (Torque Limit Value) by specifiying the bit. For details, refer to the following. ☞ Page 106 Options (Options)

■Output variables

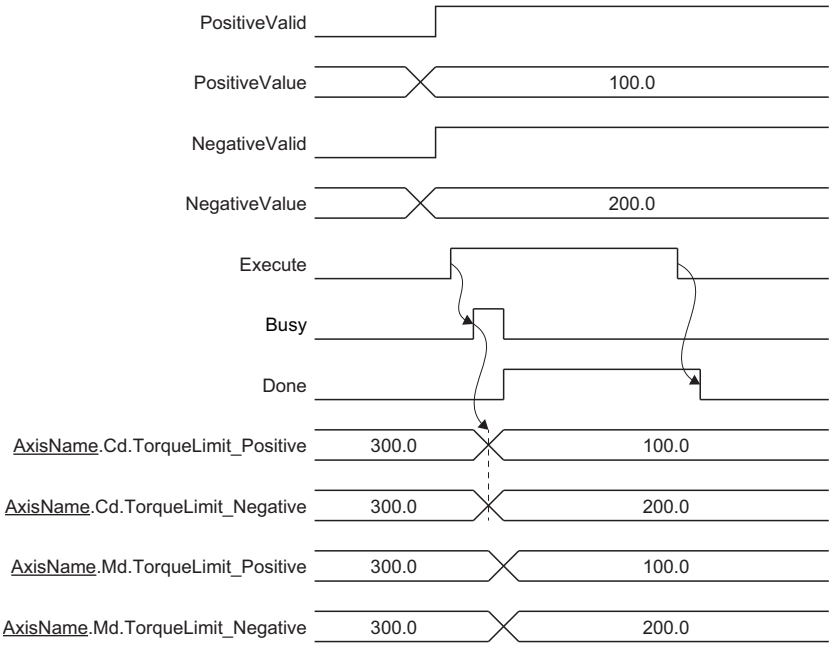
Output variable	Name	Data type	Default value	Description
Done	Execution completion	BOOL	FALSE	This variable becomes TRUE when the torque limit value change completes normally.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MCv_SetTorqueLimit (Torque Limit Value) is executed.
CommandAborted	Abortion of execution	BOOL	FALSE	This variable becomes TRUE when execution of MCv_SetTorqueLimit (Torque Limit Value) is aborted.
CancelAccepted	Cancel acceptance	BOOL	FALSE	This variable becomes TRUE when MCv_SetTorqueLimit (Torque Limit Value) accepts the cancel.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. 📖MELSEC iQ-R Motion Module User's Manual (Application)

Processing details

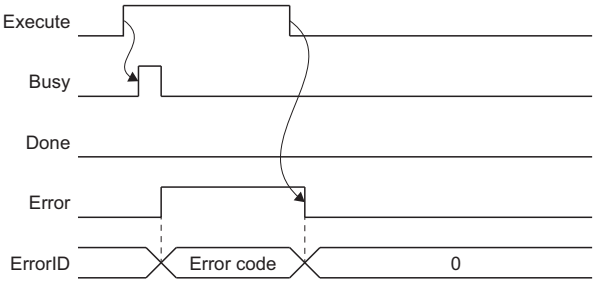
- Changes a positive direction torque limit value or negative direction torque limit value.
- When Positive direction torque limit valid (PositiveValid) is TRUE, a positive direction torque limit value is changed to the value of Positive direction torque limit value (PositiveValue).
- When Negative direction torque limit valid (NegativeValid) is TRUE, a negative direction torque limit value is changed to the value of Negative direction torque limit value (NegativeValue).
- MCv_SetTorqueLimit (Torque Limit Value) is executed when Execute command (Execute) is TRUE, and Executing (Busy) becomes TRUE when the processing starts normally.
- The timing for the torque limit change is set in Execution mode (ExecutionMode).
- When the processing is completed and the torque limit value is changed, Execution completion (Done) becomes TRUE.
- After starting MCv_SetTorqueLimit (Torque Limit Value) with Execution mode (ExecutionMode) set to "1: Execute at completion (mcQueued)", if Axis status (AxisName.Md.AxisStatus) becomes "0: Axis disabled (Disabled)" or "1: Stopping on error (ErrorStop)" before processing completes, processing is aborted and Abortion of execution (CommandAborted) becomes TRUE.
- When an error occurs in MCv_SetTorqueLimit (Torque Limit Value), Error (Error) becomes TRUE and the error code is stored in Error code (ErrorID). For details of error codes, refer to the following.
  📖MELSEC iQ-R Motion Module User's Manual (Application)
- When Positive direction torque limit valid (PositiveValid) and Negative direction torque limit valid (NegativeValid) are both FALSE, the torque limit change is not performed and Execution completion (Done) becomes TRUE.

■Timing chart

• When the operation completes normally

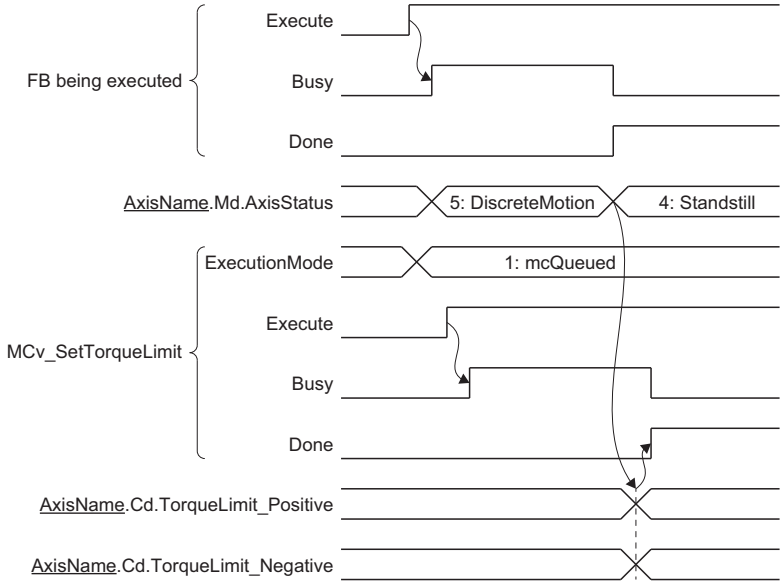
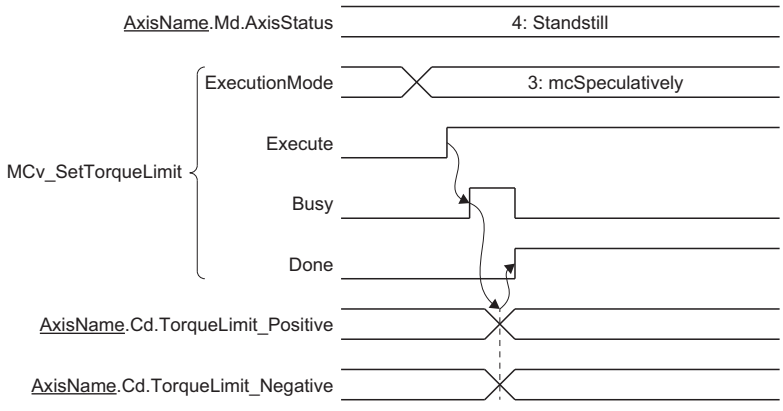


• When an error occurs



■Execution mode (ExecutionMode)

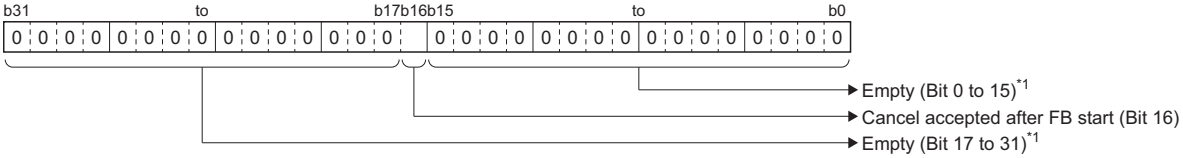
When executing MCv_SetTorqueLimit (Torque Limit Value) during axis operation, the timing of the change depends on the setting of Execution mode (ExecutionMode). The timing for Execution mode (ExecutionMode) is shown below.

Setting value	Description
0: Execute immediately (mcImmediately)	Executes immediately regardless of Axis status (AxisName.Md.AxisStatus). Positive direction torque limit value (AxisName.Cd.TorqueLimit_Positive) and Negative direction torque limit value (AxisName.Cd.TorqueLimit_Negative) are changed at rising edge detection of Execute command (Execute).
1: Execute at completion (mcQueued)	Executes after the FB being executed completes. At rising edge detection of Execute command (Execute), Executing (Busy) becomes TRUE and remains TRUE until Axis status (AxisName.Md.AxisStatus) changes to "4: Standby (Standstill)". As soon as Axis status (AxisName.Md.AxisStatus) changes to "4: Standby (Standstill)", changes Positive direction torque limit value (AxisName.Cd.TorqueLimit_Positive) and Negative direction torque limit value (AxisName.Cd.TorqueLimit_Negative). When Axis status (AxisName.Md.AxisStatus) is "4: Standby (Standstill)" at rising edge detection of Execute command (Execute), executes the FB immediately. When Axis status (AxisName.Md.AxisStatus) becomes "0: Axis disabled (Disabled)" or "1: Stopping on error (ErrorStop)" before processing completes, processing is aborted and Abortion of execution (CommandAborted) becomes TRUE. 
3: Execute speculatively (mcSpeculatively)	When Axis status (AxisName.Md.AxisStatus) of the relevant axis is "4: Standby (Standstill)" at rising edge detection of Execute command (Execute), the torque limit value change is executed. When Axis status (AxisName.Md.AxisStatus) of the relevant axis is not "4: Standby (Standstill)" or when an FB is being executed, "MCv_SetTorqueLimit Instruction Error (error code: 3450H)" occurs and the torque limit value change is not executed. 

■Options (Options)

Sets the function options for MCv_SetTorqueLimit (Torque Limit Value) by specifying the bit.

The contents set by specifying the bit are shown below.

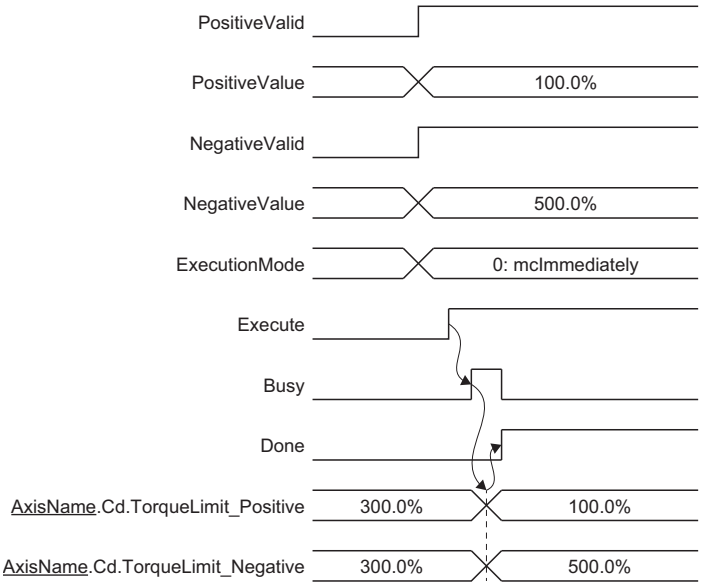


*1 Set empty areas to "0". When a value other than "0" is set, "Out of Options Range (error code: 1A4EH)" occurs.

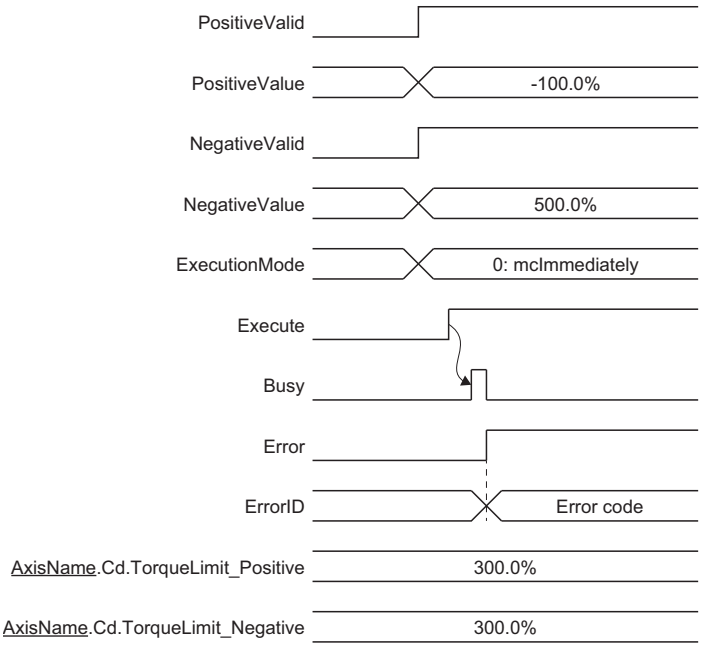
Bit	Name	Description
16	Cancel accepted after FB start	Sets whether or not to allow a cancel after MCv_SetTorqueLimit (Torque Limit Value) is executed. • 0: Do not allow • 1: Allow

■Operation overview

- In MCv_SetTorqueLimit (Torque Limit Value), Positive direction torque limit value (AxisName.Cd.TorqueLimit_Positive) and Negative direction torque limit value (AxisName.Cd.TorqueLimit_Negative) are changed.



- If Execute command (Execute) is set to TRUE while a value outside the range is set in Positive direction torque limit value (PositiveValue) or Negative direction torque limit value (NegativeValue), an error occurs in MCv_SetTorqueLimit (Torque Limit Value) and Positive direction torque limit value (AxisName.Cd.TorqueLimit_Positive) and Negative direction torque limit value (AxisName.Cd.TorqueLimit_Negative) will not be changed.



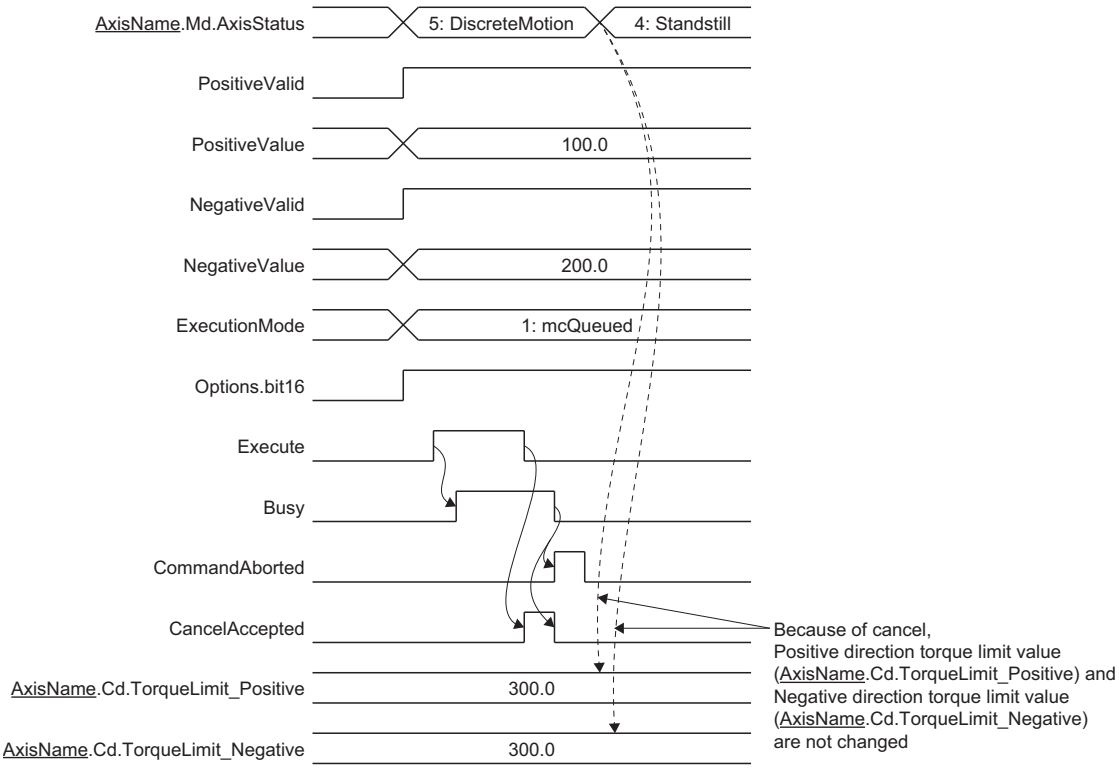
■Cancel

A torque limit value change that is in the standby status can be canceled after MCv_SetTorqueLimit (Torque Limit Value) execution.

- To enable cancel, execute MCv_SetTorqueLimit (Torque Limit Value) with the status of "Cancel accepted after FB start (Options (Options): Bit 16)" set to "1: Allow".
- Cancel is started at falling edge detection of Execute command (Execute).
- Cancel is accepted only when Executing (Busy) of the output pin is TRUE.
- When MCv_SetTorqueLimit (Torque Limit Value) accepts the cancel, Cancel acceptance (CancelAccepted) of the output pin becomes TRUE.
- When cancel is completed, Abortion of execution (CommandAborted) of the output pin becomes TRUE.
- When cancelled, Positive direction torque limit value (AxisName.Cd.TorqueLimit_Positive) and Negative direction torque limit value (AxisName.Cd.TorqueLimit_Negative) will not be changed.

Ex.

After starting with Execution mode (ExecutionMode) set to "1: Execute at completion (mcQueued)", the change is cancelled while Axis Status (AxisName.Md.AxisStatus) is waiting to switch to "4: Standby (Standstill)".



Precautions

- In MCv_SetTorqueLimit (Torque Limit Value), changing of Positive direction torque limit value (AxisName.Cd.TorqueLimit_Positive) and Negative direction torque limit value (AxisName.Cd.TorqueLimit_Negative) is performed.

Program example

In the following program example, the torque limit value change command (bSetTorqueLimit) is set to TRUE, then the positive direction torque limit value is changed to "100.0" and the negative direction torque limit value is changed to "200.0" for Axis 1 (Axis0001).

■Axis

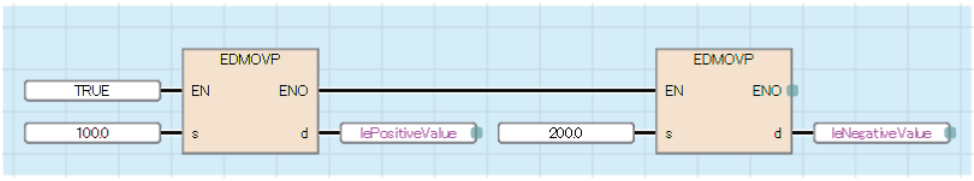
Axis No.	Label name	Data type	Comment
1	Axis0001	AXIS_REF	Axis 1

■Labels used

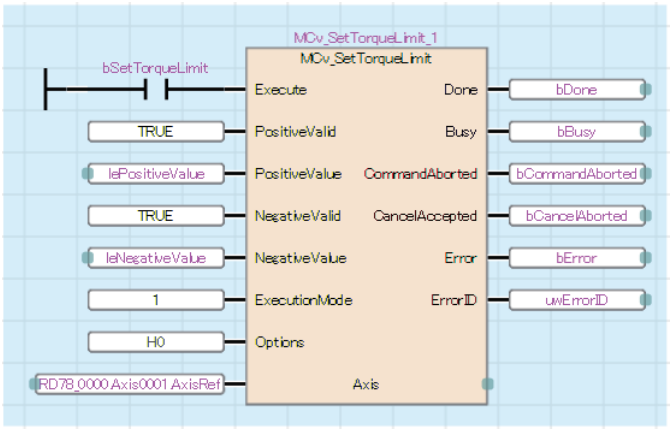
Label name	Data type	Comment
MCv_SetTorqueLimit_1	MCv_SetTorqueLimit	Torque Limit Value FB
bSetTorqueLimit	Bit	Torque limit value change command
lePositiveValue	Double-precision real number	Positive direction torque limit value
leNegativeValue	Double-precision real number	Negative direction torque limit value
bDone	Bit	Ready
bBusy	Bit	Executing
bCommandAborted	Bit	Abortion of execution
bCancelAccepted	Bit	Cancel acceptance
bError	Bit	Error
uwErrorID	Word [unsigned]/bit string [16-bit]	Error code

■FBD/LD program (CPU module side)

• Data setting for torque limit value change



• Torque limit value change

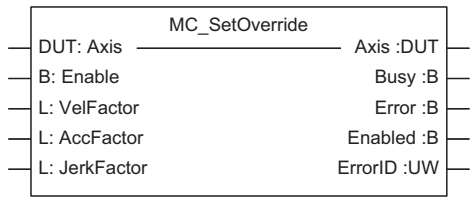
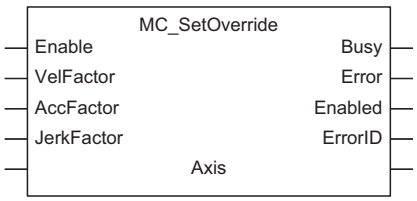


■ST program (Motion module side)

```
//-----Data setting for torque limit value change-----
lePositiveValue := 100.0;
leNegativeValue:= 200.0;

//-----Torque limit value change-----
MCv_SetTorqueLimit_1(
   Axis:= Axis0001.AxisRef ,
   Execute:= bSetTorqueLimit ,
   PositiveValid:= TRUE ,
   PositiveValue:= lePositiveValue ,
   NegativeValid:= TRUE ,
   NegativeValue:= leNegativeValue ,
   ExecutionMode:= MC_EXECUTION_MODE__mcQueued ,
   Options:= H00000000 ,
   Done=> bDone ,
   Busy=> bBusy ,
   CommandAborted=> bCommandAborted ,
   CancelAccepted=> bCancelAccepted ,
   Error=> bError ,
   ErrorID=> uwErrorID
);
```

MC_SetOverride

Changes the target velocity, the target acceleration, and the target deceleration of the specified axis.

Ladder	FBD/LD	ST
		MC_SetOverride(Axis:= ?AXIS_REF? , Enable:= ?BOOL? , VelFactor:= ?LREAL? , AccFactor:= ?LREAL? , JerkFactor:= ?LREAL? , Enabled=> ?BOOL? , Busy=> ?BOOL? , Error=> ?BOOL? , ErrorID=> ?WORD?);

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Override Value Setting	40	6	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
Axis	Axis information	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Enable	Enable	BOOL	Always	TRUE, FALSE	FALSE	When this variable is TRUE, executes MC_SetOverride (Override Value Setting).
VelFactor	Velocity override factor	LREAL	Always	0.00 to 10.00	0.00	This variable sets the velocity override factor. While Enable (Enable) is TRUE, values are always imported.
AccFactor	Acceleration override factor	LREAL	Always	0.00, 0.01 to 10.00	0.00	This variable sets the acceleration override factor. While Enable (Enable) is TRUE, values are always imported. When this variable is set to "0.0", the acceleration override factor is not changed and the control is performed with the previous value.
JerkFactor	Jerk override factor	LREAL	Always	0.0	0.0	Set this variable to "0.0". *: When set to a value other than "0.0", "Out of Jerk Override Coefficient (JerkFactor) Range (error code: 349EH)" occurs.

■Output variables

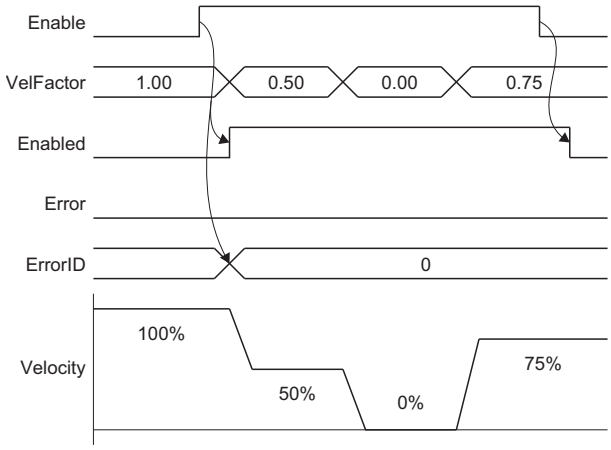
Output variable	Name	Data type	Default value	Description
Enabled	Enabled	BOOL	FALSE	This variable becomes TRUE when the set override value is correct.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MC_SetOverride (Override Value Setting) is executed.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. MELSEC iQ-R Motion Module User's Manual (Application)

Processing details

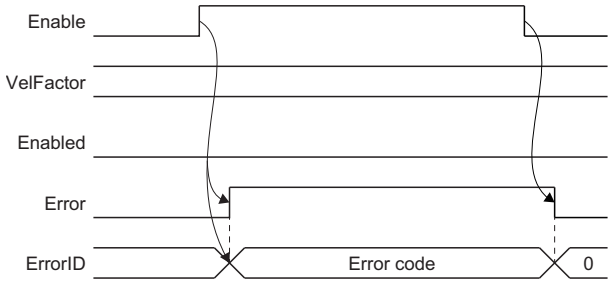
- Changes the target velocity, the target acceleration, and the target deceleration of the specified axis.
- The current operating target velocity, target acceleration, and target deceleration values are changed by multiplying each value by the override factor.
- MC_SetOverride (Override Value Setting) is executed when Enable (Enable) becomes TRUE. Enabled (Enabled) becomes TRUE while the override factor is valid.
- If the override factor value is changed while Enable (Enable) is TRUE, the new override factor is reflected.
- When an error occurs in MC_SetOverride (Override Value Setting), Error (Error) becomes TRUE and the error code is stored in Error code (ErrorID). For details of error codes, refer to the following.
  MELSEC iQ-R Motion Module User's Manual (Application)
- When the value for Velocity override factor (VelFactor) is set to "0.00", the axis stops without Axis status (AxisName.Md.AxisStatus) changing to "4: Standby (Standstill)".
- When the value for Acceleration override factor (AccFactor) is set to "0.00", the acceleration override factor is not changed and the previous acceleration override factor is maintained.

■Timing chart

- When the operation completes normally

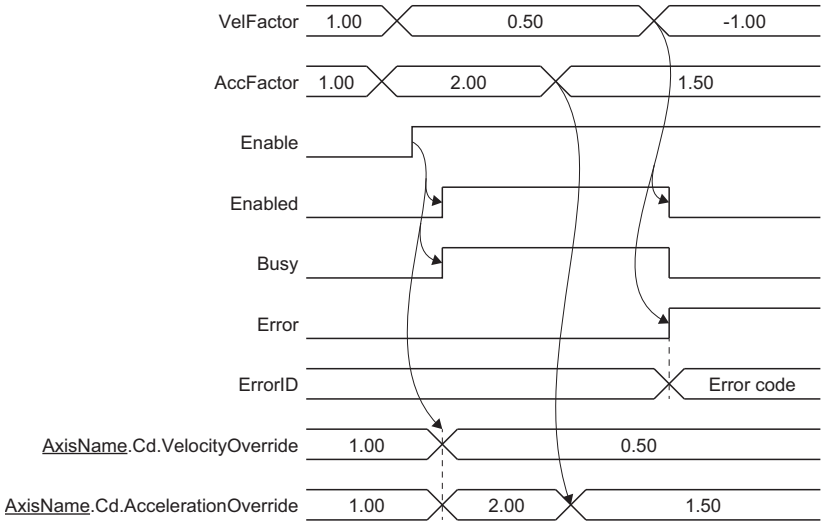


- When an error occurs



■Operation overview

- In MC_SetOverride (Override Value Setting), Speed override coefficient (AxisName.Cd.VelocityOverride) and Acceleration override coefficient (AxisName.Cd.AccelerationOverride) are changed.



- If a value outside the range is set as the override factor, an error occurs in MC_SetOverride (Override Value Setting) and subsequent values will not be imported. To import values again, start up Enable (Enable) again.

Precautions

- Do not arrange two or more MC_SetOverride (Override Value Setting) in the same axis. Operation is not guaranteed when two or more of this FB are arranged.
- In single axis control, only the override factor set with the axis control data has an effect.
- Do not operate Speed override coefficient (AxisName.Cd.VelocityOverride) or Acceleration override coefficient (AxisName.Cd.AccelerationOverride) directly while executing MC_SetOverride (Override Value Setting).
- When the speed after Speed override is out of range, refer to "Velocity Range" in the following manual.
  MELSEC iQ-R Motion Module User's Manual (Application)
- When the acceleration, deceleration, acceleration time, and deceleration time after acceleration override is out of range, refer to "Acceleration/deceleration processing function" in the following manual.
  MELSEC iQ-R Motion Module User's Manual (Application)

Program example

In the following program example, the override value change command (bSetOverride) is set to TRUE, then the target velocity, target acceleration, and target deceleration of Axis 1 (Axis0001) are changed to a value multiplied by a velocity override factor of "1.0" and an acceleration/deceleration override factor of "2.0".

■Axis

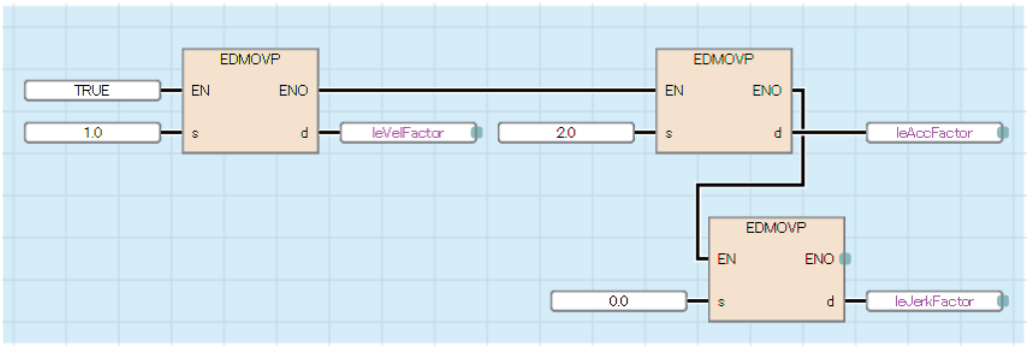
Axis No.	Label name	Data type	Comment
1	Axis0001	AXIS_REF	Axis 1

■Labels used

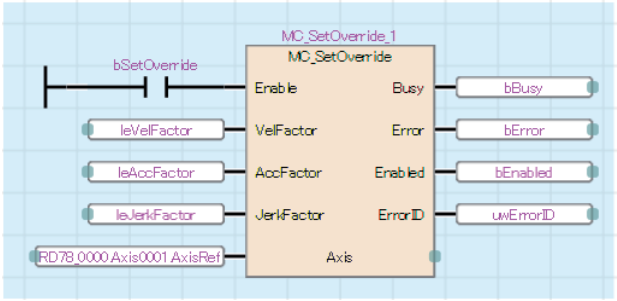
Label name	Data type	Comment
MC_SetOverride_1	MC_SetOverride	Override Value Setting FB
bSetOverride	Bit	Override value change command
leVelFactor	Double-precision real number	Velocity override factor
leAccFactor	Double-precision real number	Acceleration/deceleration override factor
leJerkFactor	Double-precision real number	Jerk override factor
bEnabled	Bit	Enabled
bBusy	Bit	Executing
bError	Bit	Error
uwErrorID	Word [unsigned]/bit string [16-bit]	Error code

■FBD/LD program (CPU module side)

• Data setting for override value change



• Override value change

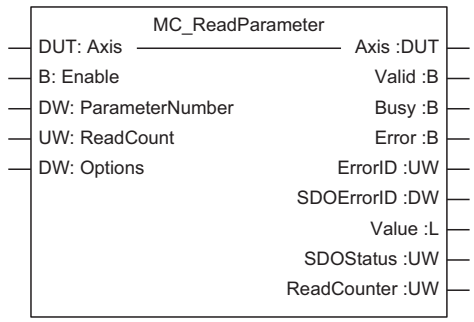
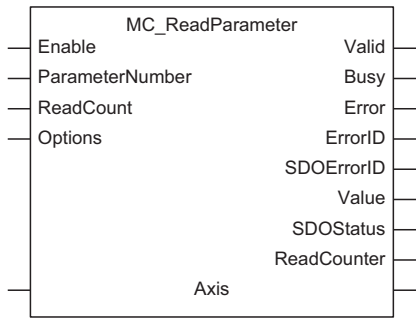


■ST program (Motion module side)

```
//-----Data setting for override value change-----
leVelFactor:= 1.00;
leAccFactor:= 2.00;
leJerkFactor:= 0.0;

//-----Override value change-----
MC_SetOverride_1(
   Axis:= Axis0001.AxisRef ,
   Enable:= bSetOverride ,
   VelFactor:= leVelFactor ,
   AccFactor:= leAccFactor ,
   JerkFactor:= leJerkFactor ,
   Enabled=> bEnabled ,
   Busy=> bBusy ,
   Error=> bError ,
   ErrorID=> uwErrorID
);
```

MC_ReadParameter

Reads objects of the slave devices.

Ladder	FBD/LD	ST
		MC_ReadParameter(Axis:= ?AXIS_REF? , Enable:= ?BOOL? , ParameterNumber:= ?DWORD? , ReadCount:= ?WORD? , Options:= ?DWORD? , Valid=> ?BOOL? , Busy=> ?BOOL? , Error=> ?BOOL? , ErrorID=> ?WORD? , SDOErrorID=> ?DWORD? , Value=> ?LREAL? , SDOStatus=> ?WORD? , ReadCounter=> ?WORD?);

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Parameter Read	24	28	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
Axis	Axis information	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Enable	Enable	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MC_ReadParameter (Parameter Read).
ParameterNumber	Parameter No.	DWORD(HEX)	Always	00010000H to FFFFFFFFH	00000000H	This variable sets the objects for the slave devices. For details, refer to the following. ☞ Page 116 Parameter No. (ParameterNumber)
ReadCount	Number of reading times	WORD(UINT)	Always	0 to 65535	0	This variable sets the number of number of reading times for parameters. When set to "0", this variable performs continuous read until Enable (Enable) becomes FALSE. Reimport is performed when the parameter No. is changed.
Options	Options	DWORD(HEX)	Always	00000000H to 00010000H	00000000H	This variable sets the function options for MC_ReadParameter (Parameter Read) by specifying the bit. Reimport is performed when the parameter No. is changed. For details, refer to the following. ☞ Page 117 Options (Options)

■Output variables

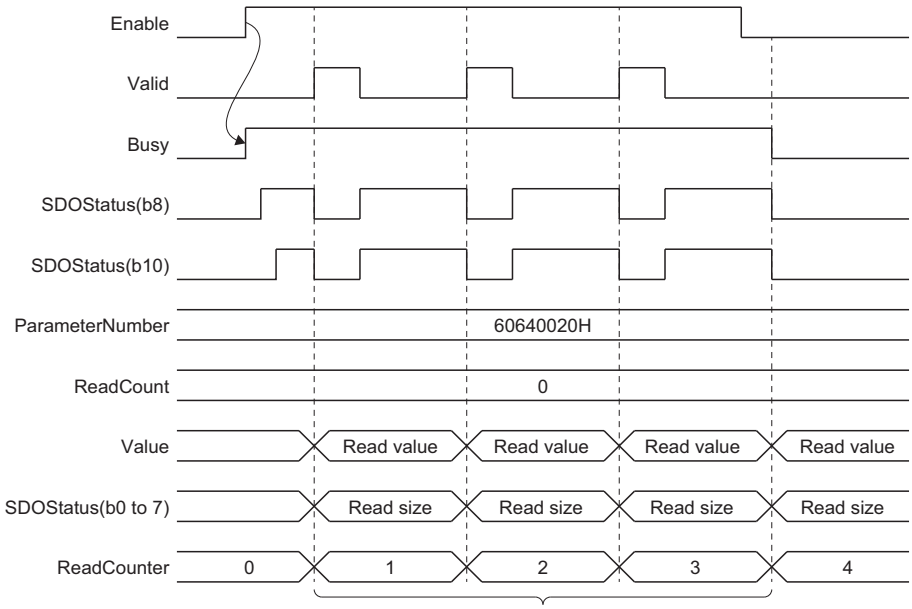
Output variable	Name	Data type	Default value	Description
Valid	Output value valid	BOOL	FALSE	This variable becomes TRUE when the output value is valid.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MC_ReadParameter (Parameter Read) is executed.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. 📖MELSEC iQ-R Motion Module User's Manual (Application)
SDOErrorID	Transient error code	DWORD(HEX)	00000000H	When an error occurs in SDO communication, this variable returns the response code (SDO Abort Code).
Value	Read value	LREAL	0.0	This variable outputs the read value from the specified parameters. This value is stored as LREAL type even when the target parameters are integer data.
SDOStatus	SDO transfer status	WORD(UINT)	0	This variable stores the processing status of the transient request. For details, refer to the following. ☞ Page 117 SDO transfer status (SDOStatus)
ReadCounter	Number of reading times	WORD(UINT)	0	This variable stores the number of reading times for parameters. When Number of reading times (ReadCount) is set to "0", the ring counter is "0 to 65535".

Processing details

- Reads the slave objects by setting a value later than 00010000H in Parameter No. (ParameterNumber). In this case, the object data is sent/received using the transient transmission function.
- When a transient request error (response code (SDO Abort Code)) is received from the slave device, Error (Error) becomes TRUE, "SDO Communication Error (error code: 1800H)" is stored in Error code (ErrorID), and the response code (SDO Abort Code) from the slave device is stored in Transient error code (SDOErrorID).

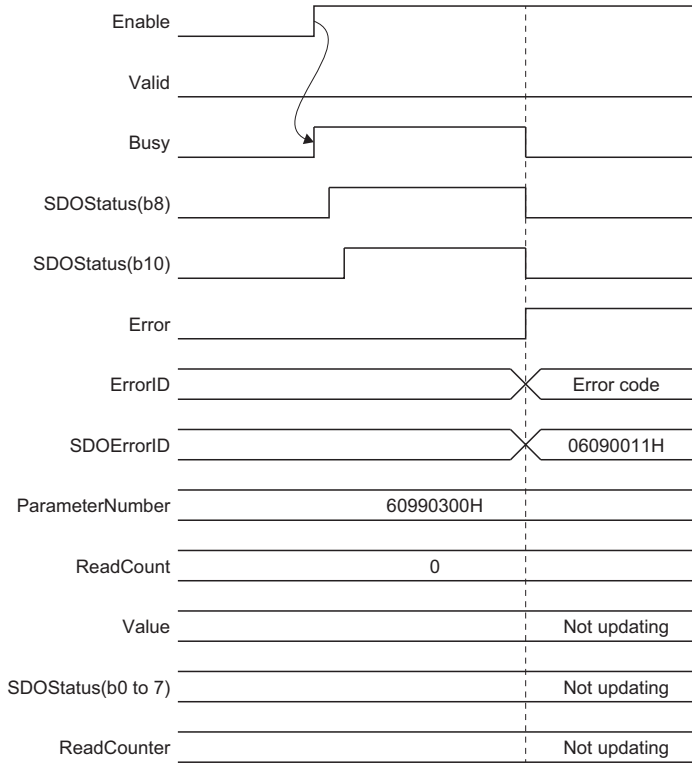
■Timing chart

- When the operation completes normally

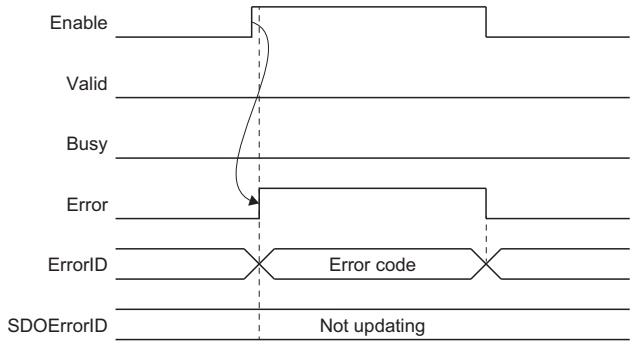


Executes read for the number of read times set in Number of reading times (ReadCount) until Enable (Enable) becomes FALSE.
(When Number of reading times (ReadCount) is set to "0", executes continuously until Enable (Enable) becomes FALSE .)

- When an error occurs
  - At SDO communication error occurrence



  - When I/O variables or input variables are abnormal

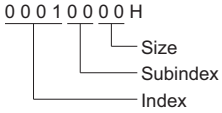


■Parameter No. (ParameterNumber)

- The classification of Parameter No. (ParameterNumber) set when reading parameters is shown below.

Parameter No.	Description
00010000H to FFFFFFFFH	Slave device object

- The contents of Parameter No. (ParameterNumber) when setting the object of the slave device are shown below.



- When setting the size of the objects of the slave device, specify 1 byte (08H)/ 2 bytes (10H)/ 4 bytes (20H)/ 8 bytes (40H) in bits. If "00H" is specified, the parameter read is performed with default size. If a value other than above setting values is specified, "Out of Parameter No. Range (error code: 34A2H)" occurs.
  When the size specified with Parameter No. (ParameterNumber) differs from default size of the slave objects, the operations depend on the specifications of the driver.

Ex.

When setting object index "6099H", UNSIGNED32 object of subindex "02H", and a size of "00H".
Set "60990200H".

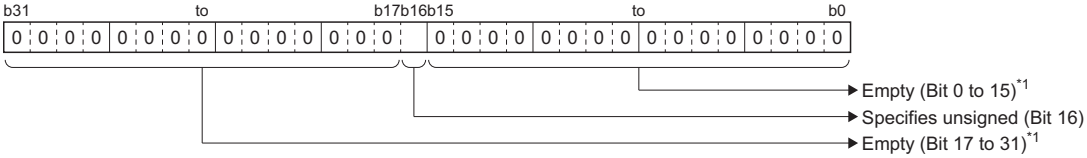
Point

For the objects of each slave device, refer to the slave device manual.

■Options (Options)

Sets the function options for MC_ReadParameter (Parameter Read) by specifying the bit.

The contents set by specifying the bit are shown below.

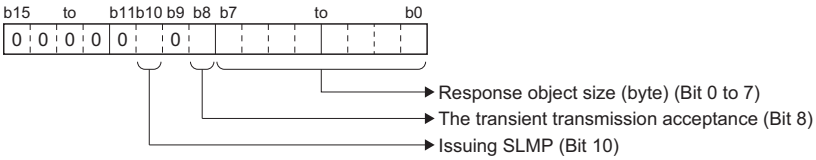


*1 Set empty areas to "0". When a value other than "0" is set, "Out of Options Range (error code: 1A4EH)" occurs.

Bit	Name	Description
16	Specifies unsigned	Sets whether to store the read value of the integer data with or without a sign. • 0: Signed • 1: Unsigned

■SDO transfer status (SDOStatus)

Stores the processing status of the transient request.



Bit	Name	Description
0 to 7	Response object size (byte)	Stores the object size sent by the slave devices when the process is completed.
8	The transient transmission acceptance	Stores the transient transmission status. • 0: Unexecuted transient transmission • 1: Accepting transient transmission
10	Issuing SLMP	Stores the SLMP status. • 0: Unexecuted SLMP • 1: Issuing SLMP

Precautions

- Up to four transient transmissions can be executed toward the same axis. If MC_ReadParameter (Parameter Read) is executed for a slave object towards an axis which is already executing four transient transmissions, "Parameter Reading And Writing FB Execution Disabled Error (error code: 34A1H)" occurs and the next FB is not executed. (The FB being executed continues.)
  For the driver homing method, obtains home position data of the driver by the transient transmission function. Therefore, if a parameter read for a slave object is executed when MC_Home (OPR) is being executed, "ABS Reference Point Read Error (error code: 1AAAH)" might occur in MC_Home (OPR).
- When reading the integer type/BOOL type parameters or slave objects, the value read with MC_ReadParameter (Parameter Read) is converted into a LREAL type or BOOL type and output.

Ex.

Reading parameters while specifying two bytes size in MC_ReadParameter (Parameter Read).

Parameter value	Read value (Value) (Data type: LREAL)	
	Specifies unsigned (Options (Options): Bit 16) "0: Signed"	Specifies unsigned (Options (Options): Bit 16) "1: Unsigned"
10000	10000.0	10000.0
-20000	-20000.0	45536.0
65535	-1.0	65535.0
63356	0.0	0.0
131701	-1.0	65535.0

- Values changed with MC_ReadParameter (Parameter Read) are not saved. To use the changed values at next power ON, save the parameters.
  For slave object devices, refer to the slave device manual for the parameter saving method.
- When a slave device in transient transmission is disconnected, all transient requests that target the disconnected slave device return an error, and "SDO Communication Error (error code: 1800H)" occurs.

Program example

In the following program example, the program read command (bReadParameter) is set to TRUE, and the object values for index "6099H", subindex "02H", and size "00H" of Axis1 (Axis0001) are read.

■Axis

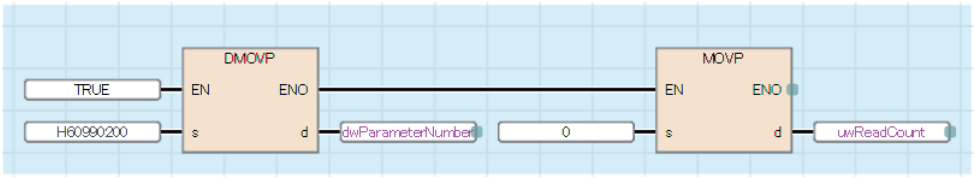
Axis No.	Label name	Data type	Comment
1	Axis0001	AXIS_REF	Axis 1

■Labels used

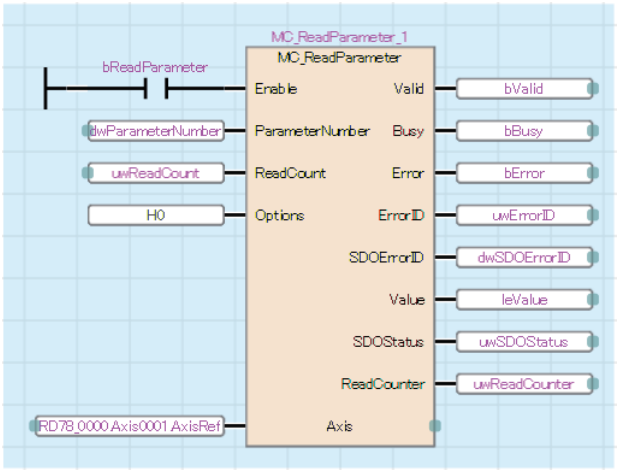
Label name	Data type	Comment
MC_ReadParameter_1	MC_ReadParameter	Parameter Read FB
bReadParameter	Bit	Program read command
dwParameterNumber	Double word [unsigned]/bit string [32-bit] (hexadecimal)	Program No.
uwReadCount	Word [unsigned]/bit string [16-bit]	Number of reading times
bValid	Bit	Output value valid
bBusy	Bit	Executing
bError	Bit	Error
uwErrorID	Word [unsigned]/bit string [16-bit]	Error code
dwSDOErrorID	Double word [unsigned]/bit string [32-bit] (hexadecimal)	Transient error code
leValue	Double-precision real number	Read value
uwSDOStatus	Word [unsigned]/bit string [16-bit]	SDO transfer status
uwReadCounter	Word [unsigned]/bit string [16-bit]	Number of reading times

■FBD/LD program (CPU module side)

- Data setting for parameter read



- Parameter read



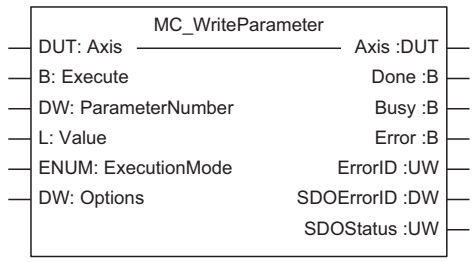
■ST program (Motion module side)

```
//-----Data setting for parameter read-----
dwParameterNumber:= H60990200;
uwReadCount:= 0;

//-----Parameter read-----
MC_ReadParameter_1(
   Axis:= Axis0001.AxisRef ,
   Enable:= bReadParameter ,
   ParameterNumber:= dwParameterNumber ,
   ReadCount:= uwReadCount ,
   Options:= H00000000 ,
   Valid=> bValid ,
   Busy=> bBusy ,
   Error=> bError ,
   ErrorID=> uwErrorID ,
   SDOErrorID=> dwSDOErrorID ,
   Value=> leValue ,
   SDOStatus=> uwSDOStatus ,
   ReadCounter=> uwReadCounter
);
```

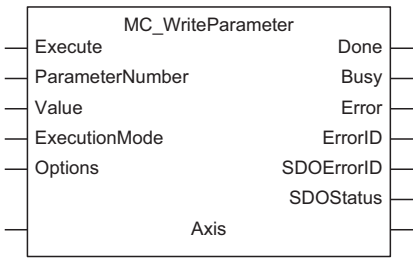
MC_WriteParameter

Writes objects of the slave devices.

Ladder



FBD/LD



ST

```
MC_WriteParameter(
   Axis:= ?AXIS_REF? ,
   Execute:= ?BOOL? ,
   ParameterNumber:= ?DWORD? ,
   Value:= ?LREAL? ,
   ExecutionMode:= ?INT? ,
   Options:= ?DWORD? ,
   Done=> ?BOOL? ,
   Busy=> ?BOOL? ,
   Error=> ?BOOL? ,
   ErrorID=> ?WORD? ,
   SDOErrorID=> ?DWORD? ,
   SDOStatus=> ?WORD?
);
```

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Parameter Write	32	14	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
Axis	Axis information	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Execute command	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MC_WriteParameter (Parameter Write).
ParameterNumber	Parameter No.	DWORD(HEX)	At start	00010000H to FFFFFFFFH	00000000H	This variable sets the objects for the slave devices. For details, refer to the following. ☞ Page 122 Parameter No. (ParameterNumber)
Value	Setting value	LREAL	At start	—	0.0	This variable specifies the setting value of the set parameter.
ExecutionMode	Execution mode	INT (MC_EXECUTION_MODE)	At start	0, 1	0	This variable sets the writing method for the set parameter. • 0: Execute immediately (mcImmediately) • 1: Execute at completion (mcQueued) For details, refer to the following. ☞ Page 123 Execution mode (ExecutionMode)
Options	Options	DWORD(HEX)	At start	00000000H to 00010000H	00000000H	This variable sets the function options for MC_WriteParameter (Parameter Write) by specifying the bit. For details, refer to the following. ☞ Page 123 Options (Options)

■Output variables

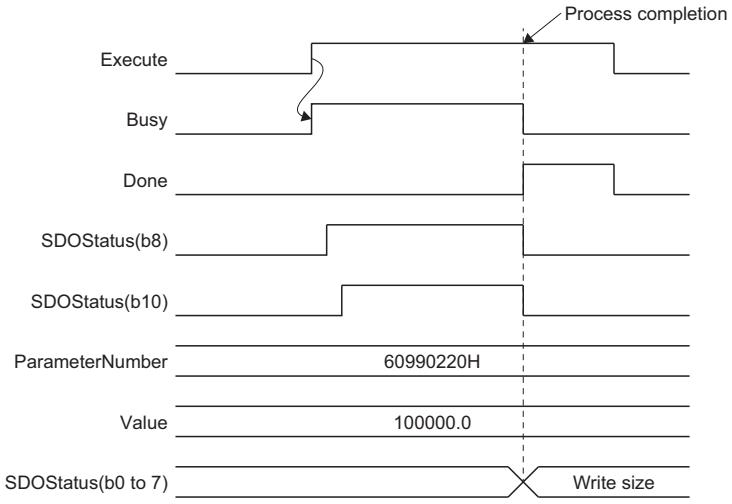
Output variable	Name	Data type	Default value	Description
Done	Execution completion	BOOL	FALSE	This variable becomes TRUE when the parameter writing is completed.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MC_WriteParameter (Parameter Write) is executed.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. MELSEC iQ-R Motion Module User's Manual (Application)
SDOErrorID	Transient error code	DWORD(HEX)	00000000H	When an error occurs in SDO communication, this variable returns the response code (SDO Abort Code).
SDOStatus	SDO transfer status	WORD(UINT)	0	This variable stores the processing status of the transient request. For details, refer to the following. Page 123 SDO transfer status (SDOStatus)

Processing details

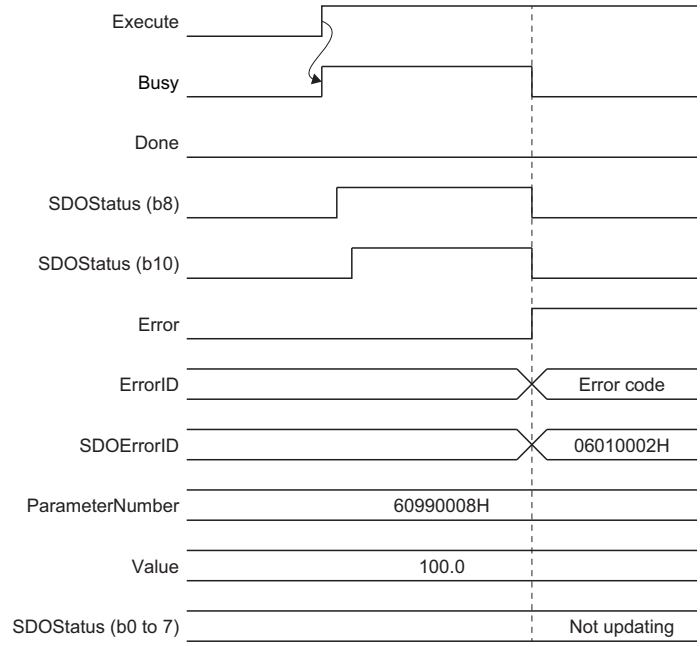
- Writes the slave objects by setting a value later than 00010000H in Parameter No. (ParameterNumber). In this case, the object data is sent/received using the transient transmission function.
- When a transient request error (response code (SDO Abort Code)) is received from the slave device, Error (Error) becomes TRUE, "SDO Communication Error (error code: 1800H)" is stored in Error code (ErrorID), and the response code (SDO Abort Code) from the slave device is stored in Transient error code (SDOErrorID).

■Timing chart

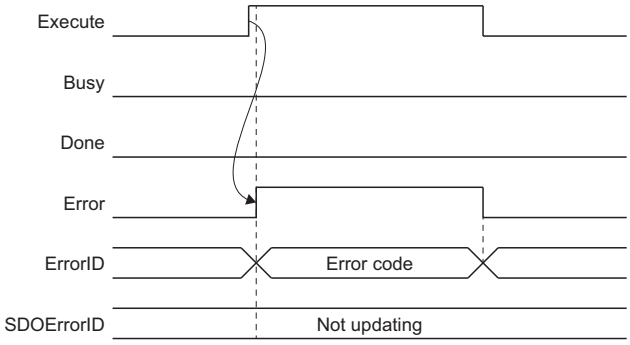
- When the operation completes normally



- When an error occurs
  - At SDO communication error occurrence



  - When I/O variables or input variables are abnormal

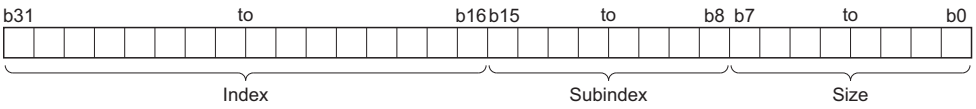


■Parameter No. (ParameterNumber)

- The classification of Parameter No. (ParameterNumber) set when writing parameters is shown below.

Parameter No.	Description
00010000H to FFFFFFFFH	Slave device object

- The contents of Parameter No. (ParameterNumber) when setting the object of the slave device are shown below.



- When setting the size of the objects of the slave device, specify 1 byte (08H)/ 2 bytes (10H)/ 4 bytes (20H)/ 8 bytes (40H) in bits. If "00H" is specified, the parameter write is performed at 4 bytes. If a value other than the above setting values is specified, "Out of Parameter No. Range (error code: 34A2H)" occurs.
  When the size specified with Parameter No. (ParameterNumber) differs from default size of the slave objects, the operations depend on the specifications of the driver.

Ex.

When setting object index "6099H", UNSIGNED32 object of subindex "02H", and a size of "00H".
Set "60990200H".

■Execution mode (ExecutionMode)

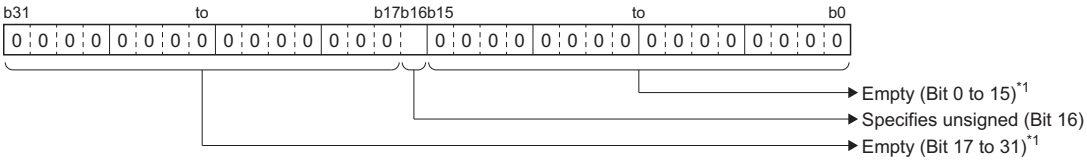
When executing a parameter write, set the timing of the write in Execution mode (ExecutionMode).

Setting value	Description
0: Execute immediately (mcImmediately)	The write is executed immediately. The axis operations might be affected because the reflection timing depends on each parameter.
1: Execute at completion (mcQueued)	The write is executed after the axis stops (Axis status (AxisName.Md.AxisStatus) is "0: Axis disabled (Disabled)" or "4: Standby (Standstill)").

■Options (Options)

Sets the function options for MC_WriteParameter (Parameter Write) by specifying the bit.

The contents set by specifying the bit are shown below.

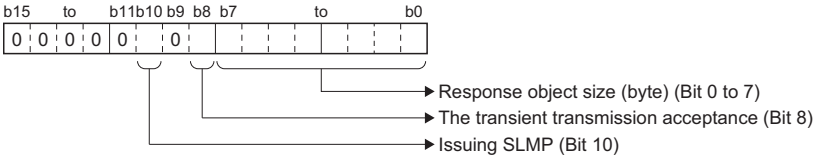


*1 Set empty areas to "0". When a value other than "0" is set, "Out of Options Range (error code: 1A4EH)" occurs.

Bit	Name	Description
16	Specifies unsigned	Sets whether to store the read value of the integer data with or without a sign. • 0: Signed • 1: Unsigned

■SDO transfer status (SDOStatus)

Stores the processing status of the transient request.



Bit	Name	Description
0 to 7	Response object size (byte)	Stores the object size sent by the slave devices when the process is completed.
8	The transient transmission acceptance	Stores the transient transmission status. • 0: Unexecuted transient transmission • 1: Accepting transient transmission
10	Issuing SLMP	Stores the SLMP status. • 0: Unexecuted SLMP • 1: Issuing SLMP

Precautions

- Up to four transient transmissions can be executed toward the same axis. If MC_WriteParameter (Parameter Write) is executed for a slave object towards an axis which is already executing four transient transmissions, "Parameter Reading And Writing FB Execution Disabled Error (error code: 34A1H)" occurs and the next FB is not executed. (The FB being executed continues.)
  For the driver homing method, obtains home position data of the driver by the transient transmission function. Therefore, if a parameter write for a slave object is executed when MC_Home (OPR) is being executed, "ABS Reference Point Read Error (error code: 1AAAH)" might occur in MC_Home (OPR).
- When writing the integer type/BOOL type parameters or slave objects, the setting value of MC_WriteParameter (Parameter Write) is converted into a LREAL type or BOOL type and imported.

Ex.
Writing parameters while specifying two bytes size in MC_WriteParameter (Parameter Write).

Setting value (Value) (Data type: LREAL)	Write value (2 bytes data)	
	Specifies unsigned (Options (Options): Bit 16) "0: Signed"	Specifies unsigned (Options (Options): Bit 16) "1: Unsigned"
10000.5	10000	10000
-10000.4	-10000	55536
65535.0	-1	65535
-65535.0	1	1
63356.0	0	0
131701.0	-1	65535

- Values changed with MC_WriteParameter (Parameter Write) are not saved. To use the changed values at next power ON, save the parameters.
  For slave object devices, refer to the slave device manual for the parameter saving method.
- When a slave device in transient transmission is disconnected, all transient requests that target the disconnected slave device return an error, and "SDO Communication Error (error code: 1800H)" occurs.

Program example

In the following program example, the parameter write command (bWriteParameter) is set to TRUE, and "100.0" is written to the object values for Axis1 (Axis0001) index "6099H", subindex "02H", and size "00H".

■Axis

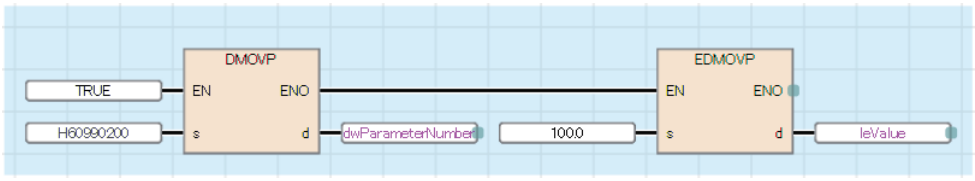
Axis No.	Label name	Data type	Comment
1	Axis0001	AXIS_REF	Axis 1

■Labels used

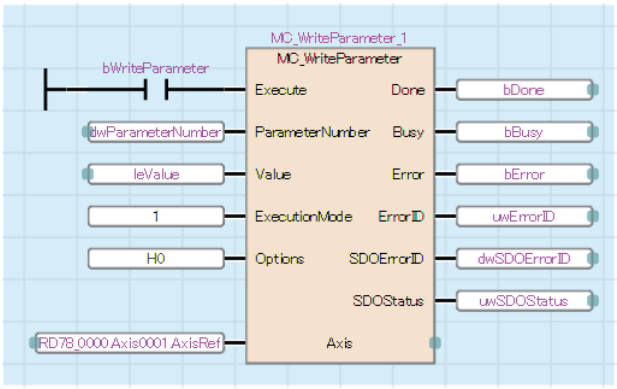
Label name	Data type	Comment
MC_WriteParameter_1	MC_WriteParameter	Parameter Write FB
bWriteParameter	Bit	Parameter write command
dwParameterNumber	Double word [unsigned]/bit string [32-bit] (hexadecimal)	Parameter No.
leValue	Double-precision real number	Setting value
bDone	Bit	Execution completion
bBusy	Bit	Executing
bError	Bit	Error
uwErrorID	Word [unsigned]/bit string [16-bit]	Error code
dwSDOErrorID	Double word [unsigned]/bit string [32-bit] (hexadecimal)	Transient error code
uwSDOStatus	Word [unsigned]/bit string [16-bit]	SDO transfer status

■FBD/LD program (CPU module side)

• Data setting for parameter write



• Parameter write

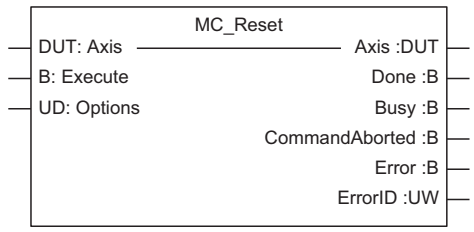
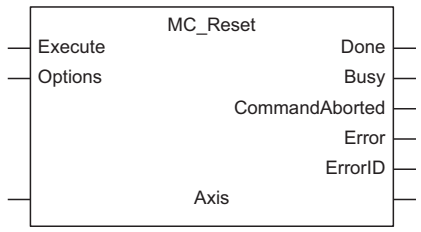


■ST program (Motion module side)

```
//-----Data setting for parameter write-----
dwParameterNumber:= H60990200;
leValid:= 100.0;

//-----Parameter write-----
MC_WriteParameter_1(
   Axis:= Axis0001.AxisRef ,
   Execute:= bWriteParameter ,
   ParameterNumber:= dwParameterNumber ,
   Value:= leValue ,
   ExecutionMode:= MC_EXECUTION_MODE__mcQueued ,
   Options:= H00000000 ,
   Done=> bDone ,
   Busy=> bBusy ,
   Error=> bError ,
   ErrorID=> uwErrorID ,
   SDOErrorID=> dwSDOErrorID ,
   SDOStatus=> uwSDOStatus
);
```

MC_Reset

Resets errors and warnings of the axis.

Ladder	FBD/LD	ST
		MC_Reset(Axis:= ?AXIS_REF? , Execute:= ?BOOL? , Options:= ?DWORD? , Done=> ?BOOL? , Busy=> ?BOOL? , CommandAborted=> ?BOOL? , Error=> ?BOOL? , ErrorID=> ?WORD?);

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Axis Error Reset	16	6	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
Axis	Axis information	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Execute command	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MC_Reset (Axis Error Reset).
Options	Options	DWORD(HEX)	At start	00000000H	00000000H	Set this variable to "00000000H". *: When a value other than "00000000H" is set, "Out of Options Range (error code: 1A4EH)" occurs.

■Output variables

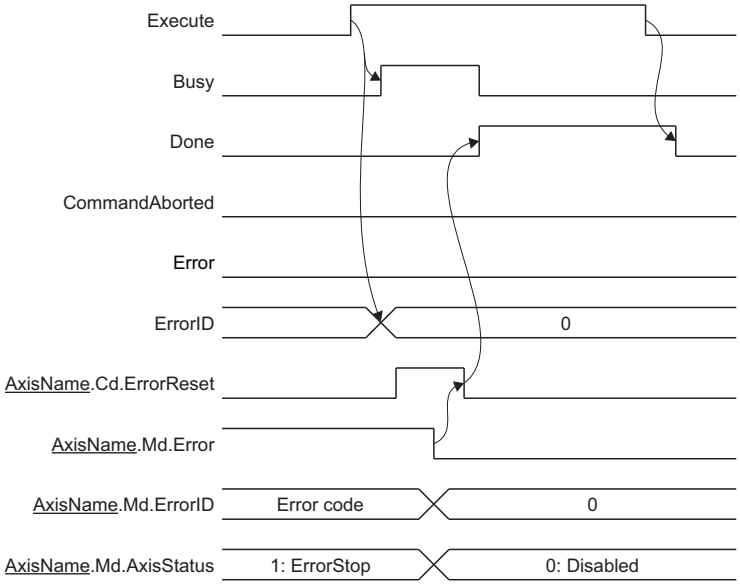
Output variable	Name	Data type	Default value	Description
Done	Execution completion	BOOL	FALSE	This variable becomes TRUE when the reset has been completed.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MC_Reset (Axis Error Reset) is executed.
CommandAborted	Abortion of execution	BOOL	FALSE	This variable becomes TRUE when MC_Reset (Axis Error Reset) has been aborted because of timeout. This variable becomes FALSE when Execute command (Execute) becomes FALSE.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. MELSEC iQ-R Motion Module User's Manual (Application)

Processing details

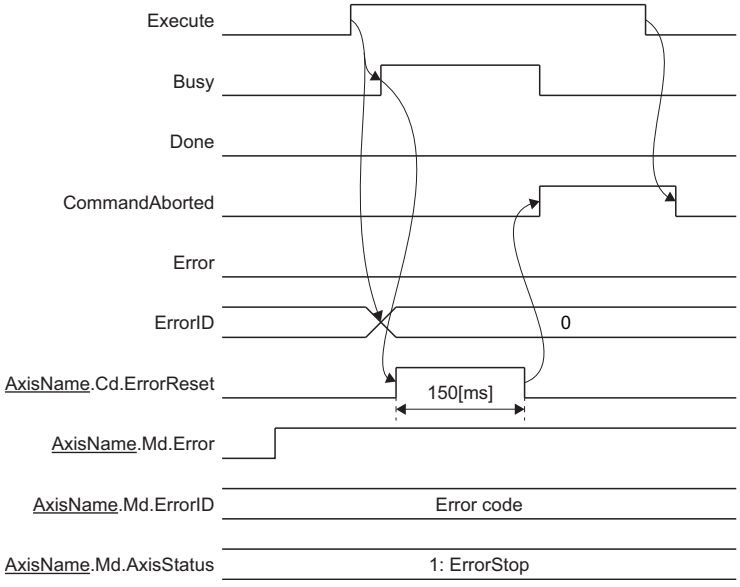
- Executes MC_Reset (Axis Error Reset) when Execute command (Execute) is TRUE, then Executing (Busy) becomes TRUE as the process starts and sets Axis error reset (AxisName.Cd.ErrorReset) of the target axis to TRUE.
- Executing (Busy) becomes FALSE and Execution completion (Done) becomes TRUE when clearing of the errors and warnings is completed.
- If executed when Axis status (AxisName.Md.AxisStatus) is other than "1: Stopping on error (ErrorStop)", Executing (Busy) becomes FALSE and Execution completion (Done) becomes TRUE after the warnings are cleared.
- Errors and warnings are not cleared if Execute command (Execute) is set to TRUE while the error and warning causes still remain. In this case, Abortion of execution (CommandAborted) becomes TRUE. Turn Execute command (Execute) to FALSE temporarily and clear the error and warning causes, then set Execute command (Execute) to TRUE again.
- When an error occurs in MC_Reset (Axis Error Reset), Error (Error) becomes TRUE and the error code is stored in Error code (ErrorID). For details of error codes, refer to the following.
  MELSEC iQ-R Motion Module User's Manual (Application)

■Timing chart

- When the operation completes normally



- At timeout



Precautions

- Do not operate Axis error reset (AxisName.Cd.ErrorReset) directly while executing MC_Reset (Axis Error Reset).
- Use MC_Reset (Axis error reset) without operating Axis error reset (AxisName.Cd.ErrorReset) to perform the error reset from the CPU module.

Program example

In the following program example, the error reset command (bErrorReset) is set to TRUE and errors and warnings are reset for Axis 1 (Axis0001).

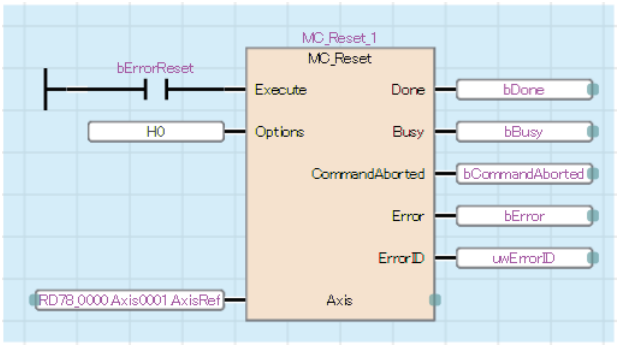
■Axis

Axis No.	Label name	Data type	Comment
1	Axis0001	AXIS_REF	Axis 1

■Labels used

Label name	Data type	Comment
MC_Reset_1	MC_Reset	Axis Error Reset FB
bErrorReset	Bit	Error reset command
bDone	Bit	Execution completion
bBusy	Bit	Executing
bCommandAborted	Bit	Abortion of execution
bError	Bit	Error
uwErrorID	Word [unsigned]/bit string [16-bit]	Error code

■FBD/LD program (CPU module side)

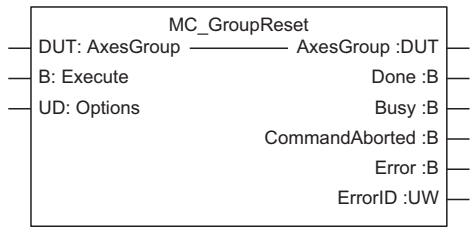
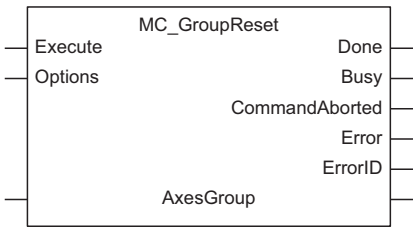


■ST program (Motion module side)

```
MC_Reset_1(
   Axis:= Axis0001.AxisRef ,
   Execute:= bErrorReset ,
   Options:= H00000000 ,
   Done=> bDone ,
   Busy=> bBusy ,
   CommandAborted=> bCommandAborted ,
   Error=> bError ,
   ErrorID=> uwErrorID
);
```

MC_GroupReset

Resets errors and warnings of the axes group and each axis in the axes group.

Ladder	FBD/LD	ST
		MC_GroupReset(AxesGroup:= ?AXES_GROUP_REF? , Execute:= ?BOOL? , Options:= ?DWORD? , Done=> ?BOOL? , Busy=> ?BOOL? , CommandAborted=> ?BOOL? , Error=> ?BOOL? , ErrorID=> ?WORD?);

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Axes Group Error Reset	16	6	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
AxesGroup	Axes group information	AXES_GROUP_REF	At start	—	Omission not possible	This variable sets the axes group. For the variables used (AxesGroupName.AxesGroupRef.), refer to the following. ☞ Page 46 AxesGroupName.AxesGroupRef. (Axes group information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Execute command	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MC_GroupReset (Axes Group Error Reset).
Options	Options	DWORD(HEX)	At start	00000000H	00000000H	Set this variable to "00000000H". *: When a value other than "00000000H" is set, "Out of Options Range (error code: 1A4EH)" occurs.

■Output variables

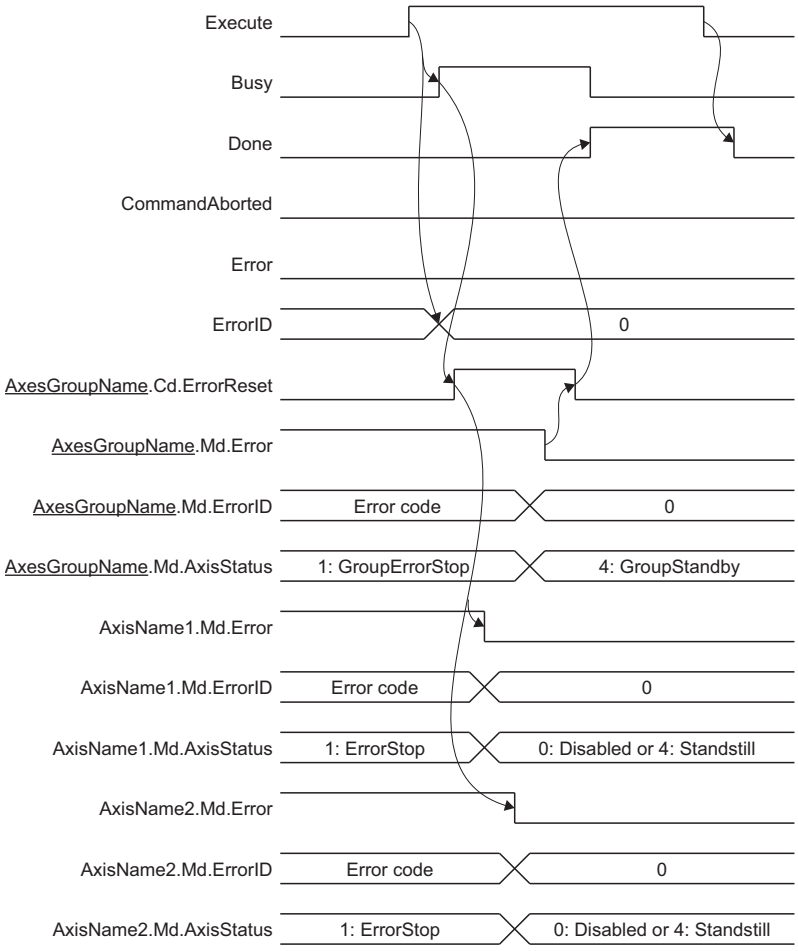
Output variable	Name	Data type	Default value	Description
Done	Execution completion	BOOL	FALSE	This variable becomes TRUE when the reset of all axes in the axes group has been completed.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MC_GroupReset (Axes Group Error Reset) is executed.
CommandAborted	Abortion of execution	BOOL	FALSE	This variable becomes TRUE when MC_GroupReset (Axes Group Error Reset) has been aborted because of timeout. This variable becomes FALSE when Execute command (Execute) becomes FALSE.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. 📖MELSEC iQ-R Motion Module User's Manual (Application)

Processing details

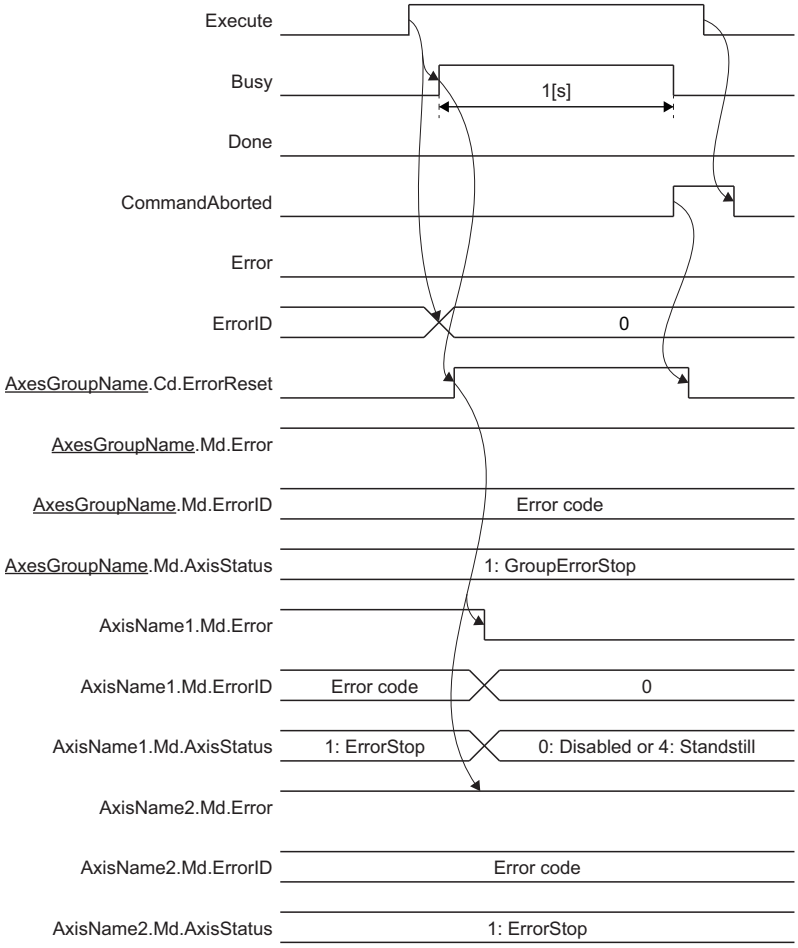
- Executes MC_GroupReset (Axes Group Error Reset) at the rising edge detection of Execute command (Execute), then Executing (Busy) becomes TRUE as the process starts and sets Axes group error reset (AxesGroupName.Cd.ErrorReset) of the target axes group to TRUE.
- Executing (Busy) becomes FALSE and Execution completion (Done) becomes TRUE when clearing of the errors and warnings is completed for the axes group or individual axis.
- If executed when Axes group status (AxesGroupName.Md.AxisStatus) is other than "1: Stopping on error (GroupErrorStop)", Executing (Busy) becomes FALSE and Execution completion (Done) becomes TRUE after the warnings are cleared.
- Axis errors and warnings are not cleared if Execute command (Execute) is set to TRUE while the error and warning causes still remain. In this case, if the errors are not cleared within one second after the FB is executed, Abortion of execution (CommandAborted) becomes TRUE and sets Axes group error reset (AxesGroupName.Cd.ErrorReset) to FALSE. Set Execute command (Execute) to FALSE temporarily and clear the error and warning causes, then set Execute command (Execute) to TRUE again.
- When an error occurs in MC_GroupReset (Axes Group Error Reset), Error (Error) becomes TRUE and the error code is stored in Error code (ErrorID). For details of error codes, refer to the following.
  MELSEC iQ-R Motion Module User's Manual (Application)

■Timing chart

- When the operation completes normally



- At timeout



Precautions

- Do not operate Axes group error reset (AxesGroupName.Cd.ErrorReset) directly while executing MC_GroupReset (Axes Group Error Reset).
- Use MC_GroupReset (Axes Group Error Reset) without operating Axes group error reset (AxesGroupName.Cd.ErrorReset) to perform the error reset from the CPU module.

Program example

In the following program example, the axes group error reset command (bGroupErrorReset) is set to TRUE and errors and warnings are reset for Axes group 1 (AxesGroup001).

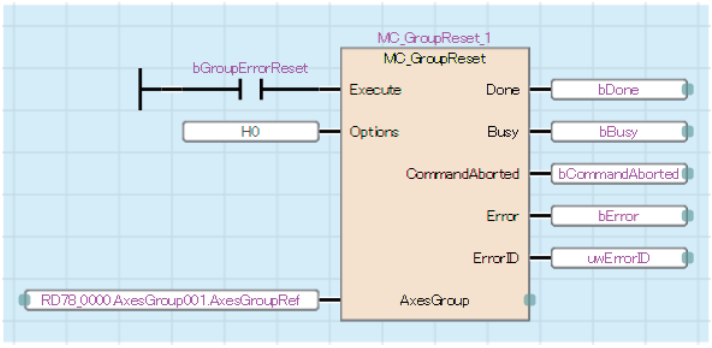
■Axes group

Axes group No.	Label name	Data type	Comment
1	AxesGroup001	AXES_GROUP_REF	Axes group 1

■Labels used

Label name	Data type	Comment
MC_GroupReset_1	MC_GroupReset	Axes Group Error Reset FB
bGroupErrorReset	Bit	Axes group error reset command
bDone	Bit	Execution completion
bBusy	Bit	Executing
bCommandAborted	Bit	Abortion of execution
bError	Bit	Error
uwErrorID	Word [unsigned]/bit string [16-bit]	Error code

■FBD/LD program (CPU module side)



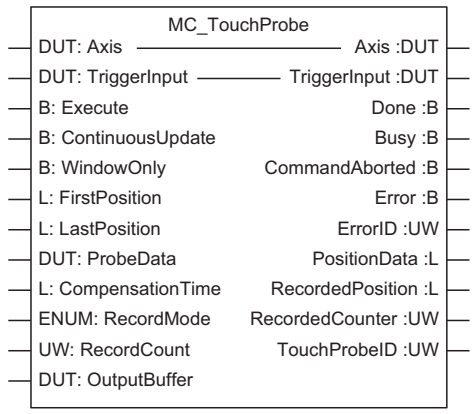
■ST program (Motion module side)

```
MC_GroupReset_1(
   AxesGroup:= AxesGroup001.AxesGroupRef ,
   Execute:= bGroupErrorReset,
   Options:= H00000000 ,
   Done=> bDone ,
   Busy=> bBusy ,
   CommandAborted=>  bCommandAborted,
   Error=> bError ,
   ErrorID=> uwErrorID
);
```

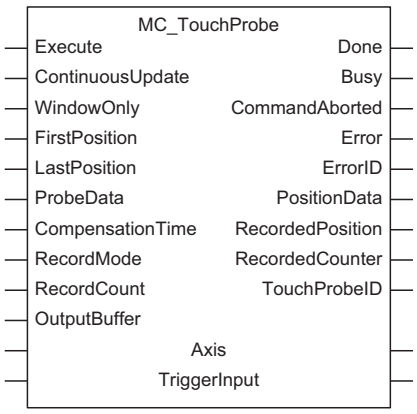
MC_TouchProbe

Records optional data when the trigger event occurs.

Ladder



FBD/LD



ST

```
MC_TouchProbe(
  Axis:= ?AXIS_REF? ,
  TriggerInput:= ?MC_TRIGGER_REF? ,
  Execute:= ?BOOL? ,
  ContinuousUpdate:= ?BOOL? ,
  WindowOnly:= ?BOOL? ,
  FirstPosition:= ?LREAL? ,
  LastPosition:= ?LREAL? ,
  ProbeData:= ?TARGET_REF? ,
  CompensationTime:= ?LREAL? ,
  RecordMode:= ?INT? ,
  RecordCount:= ?WORD? ,
  OutputBuffer:= ?TARGET_REF? ,
  Done=> ?BOOL? ,
  Busy=> ?BOOL? ,
  CommandAborted=> ?BOOL? ,
  Error=> ?BOOL? ,
  ErrorID=> ?WORD? ,
  PositionData=> ?LREAL? ,
  RecordedPosition=> ?LREAL? ,
  RecordedCounter=> ?WORD? ,
  TouchProbeID=> ?WORD?
);
```

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Touch Probe Enabled	480	28	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
Axis	Axis information	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)
TriggerInput	Trigger input signal	MC_TRIGGER_REF	At start	—	Omission not possible	This variable sets the trigger input signal. For details, refer to the following. ☞ Page 138 Trigger input signal (TriggerInput)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Execute command	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MC_TouchProbe (Touch Probe Enabled).
ContinuousUpdate	Continuous update	BOOL	At start	TRUE, FALSE	FALSE	This variable sets whether to enable or disable continuous change of Enabled window (WindowOnly), Lower limit position (FirstPosition), and Upper limit position (LastPosition). • FALSE: Disable • TRUE: Enable
WindowOnly	Enabled window	BOOL	At start/ Retrigger possible/ Continuous update possible	TRUE, FALSE	FALSE	This variable sets whether or not to only detect triggers that occur in the valid area defined by the lower limit position and the upper limit position. • FALSE: Detect in entire area • TRUE: Detect in set area only For details, refer to the following. ☞ Page 139 Enabled window(WindowOnly)/ Lower limit position (FirstPosition)/Upper limit position (LastPosition)

Input variable	Name	Data type	Input	Setting range	Default value	Description
FirstPosition	Lower limit position	LREAL	At start/ Retrigger possible/ Continuous update possible	—	0.0	This variable sets the lower limit position of the trigger event valid area. The valid area includes the lower limit position. For details, refer to the following. ☞ Page 139 Enabled window(WindowOnly)/ Lower limit position (FirstPosition)/Upper limit position (LastPosition)
LastPosition	Upper limit position	LREAL	At start/ Retrigger possible/ Continuous update possible	—	0.0	This variable sets the upper limit position of the trigger event valid area. The valid area includes the upper limit position. For details, refer to the following. ☞ Page 139 Enabled window(WindowOnly)/ Lower limit position (FirstPosition)/Upper limit position (LastPosition)
ProbeData	Probe data	TARGET_REF	At start	—	—	This variable sets the probe data. For details, refer to the following. ☞ Page 140 Probe data (ProbeData)
CompensationTime	Compensation time	LREAL	At start	-5.0 to 5.0 [s]	0.0	This variable sets the compensation value for the delay time of the touch probe processing. Set a positive value to compensate for a delay. For details, refer to the following. ☞ Page 141 Compensation time (CompensationTime)
RecordMode	Latch mode	INT (MC_RECORD_MODE)	At start	0 to 2	0	This variable sets the latch method for the data. • 0: One shot mode (OneShot) • 1: Frequency specified mode (RecordCount) • 2: Ring buffer mode (RingBuffer) For details, refer to the following. ☞ Page 142 Latch mode (RecordMode)
RecordCount	Number of latch times	WORD(UINT)	At start	1 to 65535	1	This variable sets the number of latch times. For details, refer to the following. ☞ Page 142 Number of latch times (RecordCount)
OutputBuffer	Storage location of latch data	TARGET_REF	At start	—	—	This variable sets the storage location of the latch data. When omitted, latch data is output only in Latch position (RecordedPosition). For details, refer to the following. ☞ Page 143 Storage location of latch data (OutputBuffer)

■Output variables

Output variable	Name	Data type	Default value	Description
Done	Completion	BOOL	FALSE	This variable becomes TRUE when the output value is valid.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MC_TouchProbe (Touch Probe Enabled) is executed.
CommandAborted	Abortion of execution	BOOL	FALSE	This variable becomes TRUE when execution of MC_TouchProbe (Touch Probe Enabled) is aborted by an error or multiple start.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. 📖MELSEC iQ-R Motion Module User's Manual (Application)
PositionData	Current value monitor data	LREAL	0.0	This variable stores the current value of the data to be latched.
RecordedPosition	Latch position	LREAL	0.0	This variable stores the current value when a trigger event is occurring. When latch is executed multiple times, the data is overwritten by the latest data.

Output variable	Name	Data type	Default value	Description
RecordedCounter	Number of latch times	WORD(UINT)	0	This variable stores the number of times that latch is executed.
TouchProbeID	Probe ID	WORD(UINT)	0	This variable stores an ID unique to MC_TouchProbe (Touch Probe Enabled).

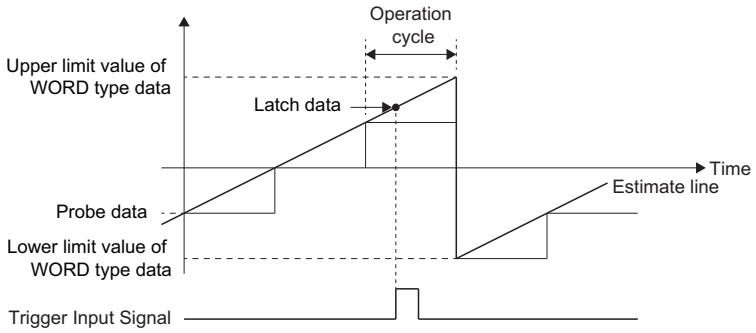
Processing details

- Calculations for the latched data are estimated at rising edge/falling edge/both directions of the signal specified in Trigger input signal (TriggerInput).
- When Enabled window (WindowOnly) is TRUE, confirms whether the latch data at trigger event occurrence is within the range of Lower limit position (FirstPosition) and Upper limit position (LastPosition) or not. Data outside the range are not latched.
- The latch data is stored to the storage location set in Latch position (RecordedPosition) or Storage location of latch data (OutputBuffer) according to Latch mode (RecordMode), and Number of latch times (RecordedCounter) is updated.
- The probe data during operation cycle interval is calculated by estimation. The value calculated by estimation when Trigger input signal (TriggerInput) is input is the latch data. The value is calculated as shown in the figure below.

■When the probe data is WORD type

• When using an external signal high-accuracy input

The trigger input timing is the time when the slave station detects the signal.

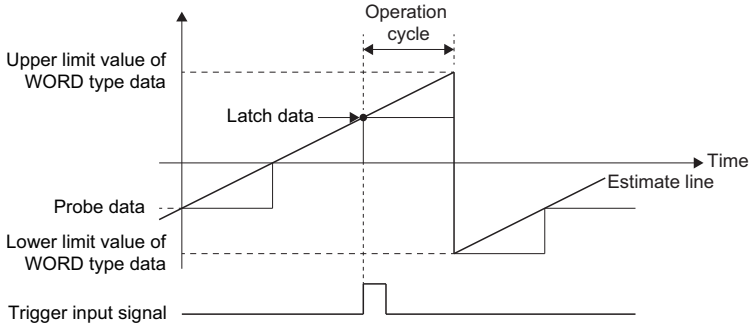


*: For how to set high-accuracy input for selected external signals, refer to the following.
MELSEC iQ-R Motion Module User's Manual (Application)

*: For the accuracy of signal detection time, refer to the manual of the slave device to be used.

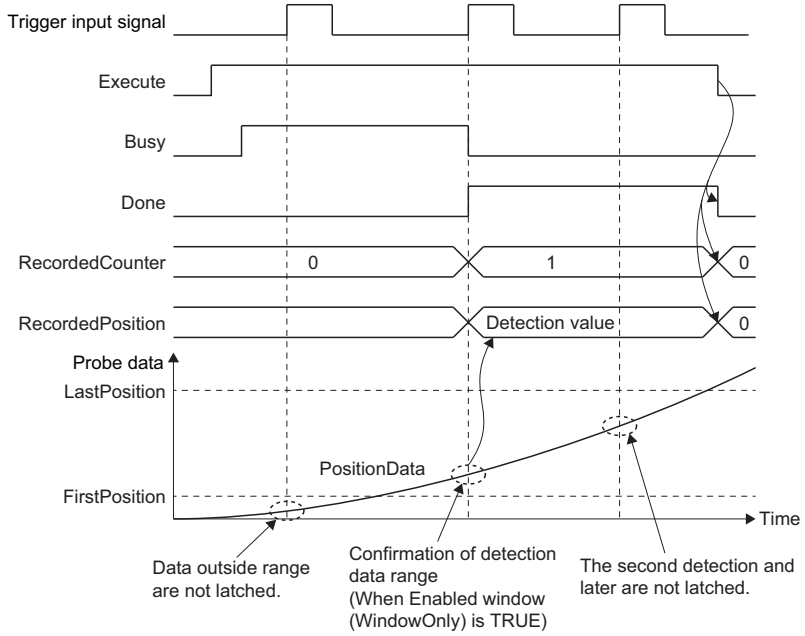
• When not using an external signal high-accuracy input

The trigger input timing is the operation cycle.



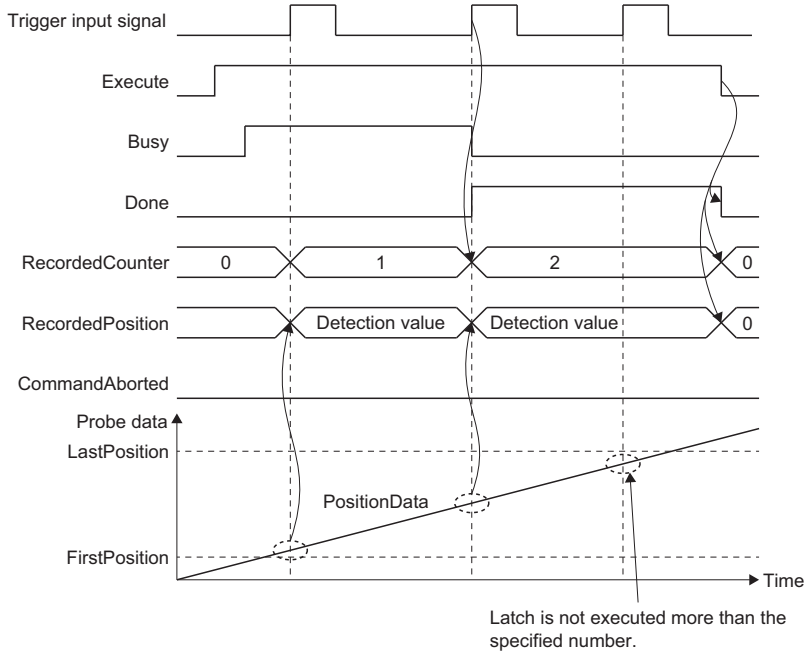
■Timing chart

- When the operation completes normally
  - When Latch mode (RecordMode) is "0: One shot mode (OneShot)"

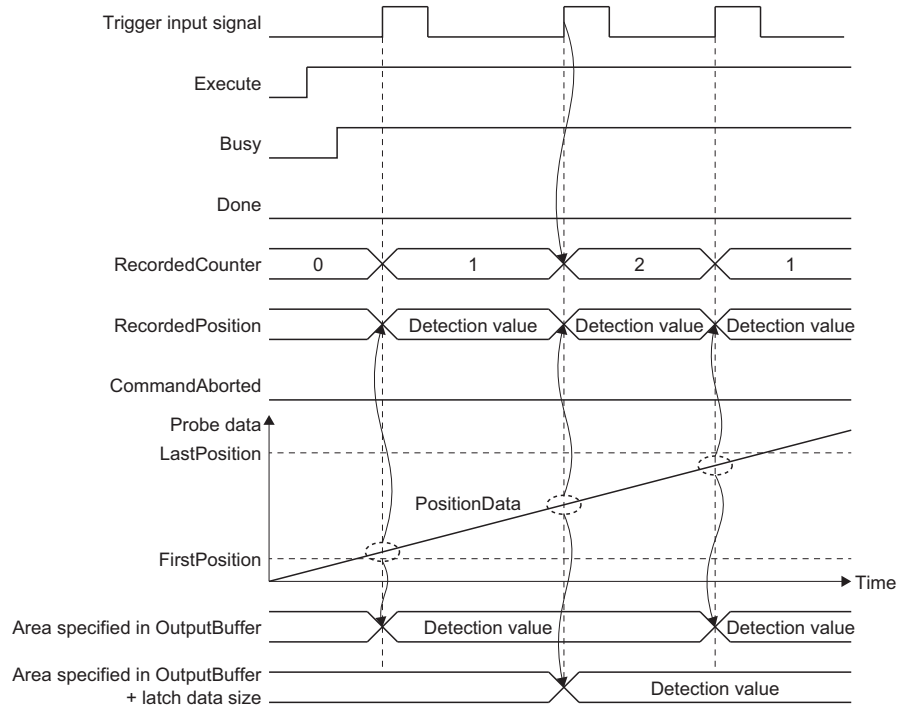


  - When Latch mode (RecordMode) is "1: Frequency specified mode (RecordCount)"

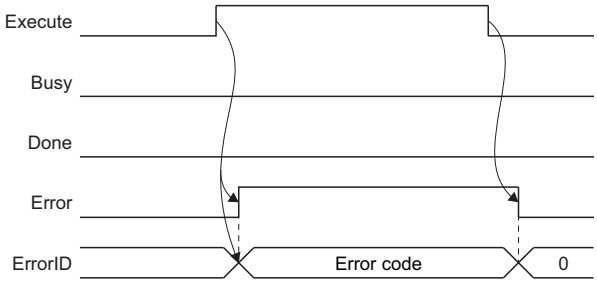
    *: When Number of latch times (RecordCount) is two times



• When Latch mode (RecordMode) is "2: Ring buffer mode (RingBuffer)"
*: When Number of latch times (RecordCount) is two times



• When an error occurs



■Trigger input signal (TriggerInput)

Trigger input signal is set in the MC_TRIGGER_REF structure. For the MC_TRIGGER_REF structure, refer to the following.

☞ Page 63 MC_TRIGGER_REF

The setting range for Trigger input signal (Signal) of the MC_TRIGGER_REF structure set in Trigger input signal (TriggerInput) is shown below.

The type for Trigger signal (Signal) is SIGNAL_SELECT structure. For the SIGNAL_SELECT structure, refer to the following.

☞ Page 66 SIGNAL_SELECT (Signal select)

Structure	Variable name	Type	Setting range
SIGNAL_SELECT (Signal select)	Source (Signal)	TARGET_REF ☞ Page 64 TARGET_REF (Input signal)	■Type • BOOL ■Data category • [VAR] • [DEV] • [CONST] • [OBJ]*1*2
	Detection (Signal detection method)	INT (MC_SIGNAL_LOGIC) ☞ Page 79 MC_SIGNAL_LOGIC	• 2:Detection at FALSE→TRUE (rising edge) (RisingEdge) • 3:Detection at TRUE→FALSE (falling edge) (FallingEdge) • 4:Detection at rising edge/falling edge (BothEdges)
	CompensationTime (Compensation time)	LREAL	-5.0 to 5.0[s]
	FilterTime (Filter time)	LREAL	0.0 to 5.0[s]

*1 When the target modification is omitted, the object of Axis Information (Axis) axis is referred. However, if the Axis Information (Axis) axis does not have a station address, or if a station address has not been set, an error will occur.
When the station address for the target modification has been specified, the Axis Information (Axis) is ignored.
When the referred station has the Slave Emulate function enabled and also has a set station address, the emulating object will be referred.

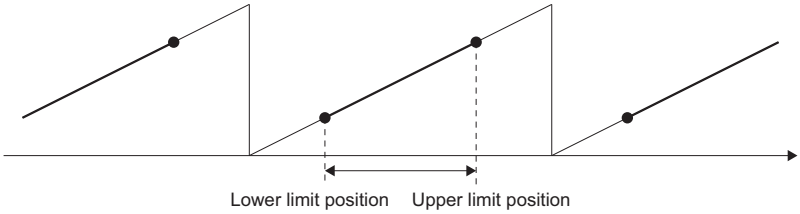
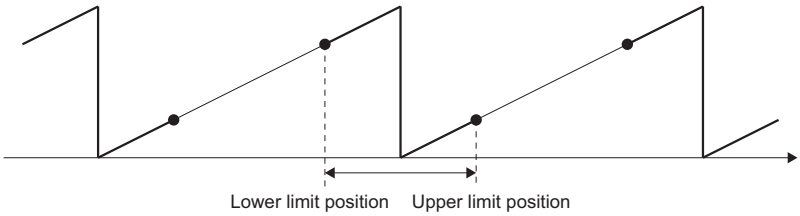
*2 An external signal high-accuracy input can be used. For details, refer to the following.
MELSEC iQ-R Motion Module User's Manual (Application)

Point

When using an external signal high-accuracy input, set the Filter time (TriggerInput.Signal.FilterTime) to "0". If set to a value other than "0", the signal detection time will not be correctly obtained.
If the slave device has an input filter, use this slave device filter.
If the slave device does not have an input filter, use the Window enabled (WindowOnly), Lower limit position (FirstPosition), and Upper limit position (LastPosition) of Touch Probe Enabled (MC_TouchProbe) together to prevent a false detection of the trigger signal due to noise.

■Enabled window(WindowOnly)/Lower limit position (FirstPosition)/Upper limit position (LastPosition)

When Enabled window(WindowOnly) is valid (TRUE), triggers are detected only within the range between Lower limit position (FirstPosition) and Upper limit position (LastPosition) of the probe data. When it is invalid (FALSE), the touch probe processing is executed for the entire range.

Lower limit position (FirstPosition)/Upper limit position (LastPosition) relation	Touch probe processing
Lower limit position (FirstPosition) ≤ Upper limit position (LastPosition)	The touch probe processing is executed when the probe data is more than Lower limit position (FirstPosition) and less than Upper limit position (LastPosition). 
Lower limit position (FirstPosition) > Upper limit position (LastPosition)	The touch probe processing is executed when the probe data is less than Upper limit position (LastPosition) or more than Lower limit position (FirstPosition). 

■Probe data (ProbeData)

Set the data to latch in the TARGET_REF structure. For the TARGET_REF structure, refer to the following.

☞ Page 64 TARGET_REF (Input signal)

The settable data categories are shown below.

Data classification	Description
[AXIS]	The axis data of Axis information (Axis) is referred. The target modification (@Position, etc.) needs to be set.
[VAR]	Axis information (Axis) is ignored.
[DEV]	
[OBJ] (which has been mapped to cyclic data)	• When the station address for the target modification has been specified, the Axis Information (Axis) is ignored. • When the target modification is omitted, the object of Axis Information (Axis) axis is referred. However, if the Axis Information (Axis) axis does not have a station address, or if a station address has not been set, an error will occur. • When the referred station has the Slave Emulate function enabled and also has a set station address, the emulating object will be referred.

- If the addon_base_MotionControlGeneral version is "earlier than 1.15", as latched data will be stored as LREAL type, when setting non-LREAL type data in Target (Target) of the TARGET_REF structure, set "(LREAL)" explicitly to the data type.
- In the following setting example, Probe data (ProbeData) is set in Target (Target) of the TARGET_REF structure.

Ex.

Target (Target) setting example

· [DEV](LREAL)G11500000

· [VAR]OutputBufferData1*1

· [VAR](LREAL)OutputBufferData2*2

*1 OutputBufferData1 is a LREAL type instance.

*2 OutputBufferData2 is a WORD type instance.

- The data is updated as the ring counter. The upper limit value and lower limit value of the ring counter differ depending on the data classification and data type of Probe data (ProbeData).

Data classification	Description
[AXIS]	■When the target modification is set to "@Position" • The ring counter upper limit value and ring counter lower limit value of the axis are used. ■When the target modification is set to "@CumulativePos" • The positioning range upper limit value and positioning range lower limit value of the axis are used.
[VAR]	The maximum value and minimum value of the data type are used. <Example> When [VAR] is set to a Word type label. • Ring counter upper limit value: 65535 • Ring counter lower limit value: 0
[DEV]	

Restriction

Ensure that the change amount per operation cycle satisfies the following formula. Failing to do so may cause the actual change amount of the probe data and the latched change amount to not match.

$$\text{Change amount per operation cycle} < \frac{|(\text{Ring counter maximum value}) - (\text{Ring counter minimum value}) + 1|}{2}$$

■Compensation time (CompensationTime)

Compensates for the delay time of the touch probe processing. Set this variable to compensate for delays specific to the touch probe processing separately from the compensation time of the trigger input signal. To compensate for delays, set a positive number. To compensate for the progression, set a negative number.
The total compensation time is shown below.

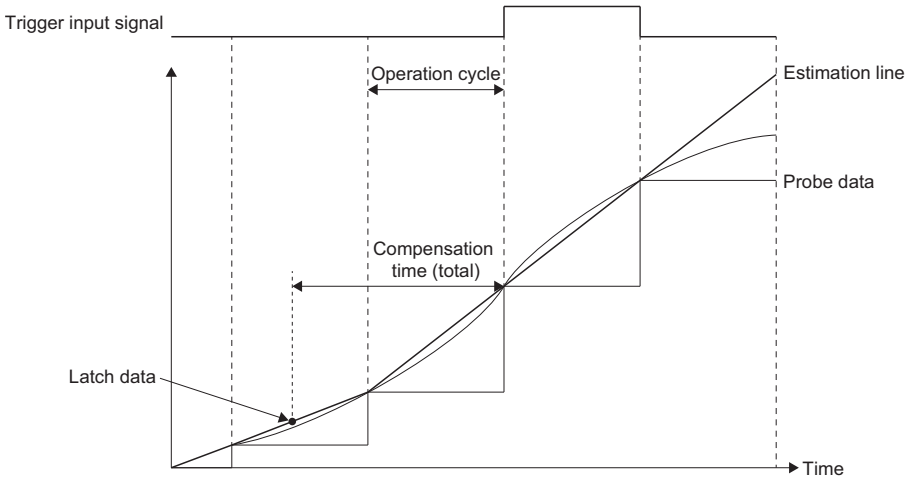
Compensation time (CompensationTime) - Compensation time (TriggerInput.Signal.CompensationTime)

Point

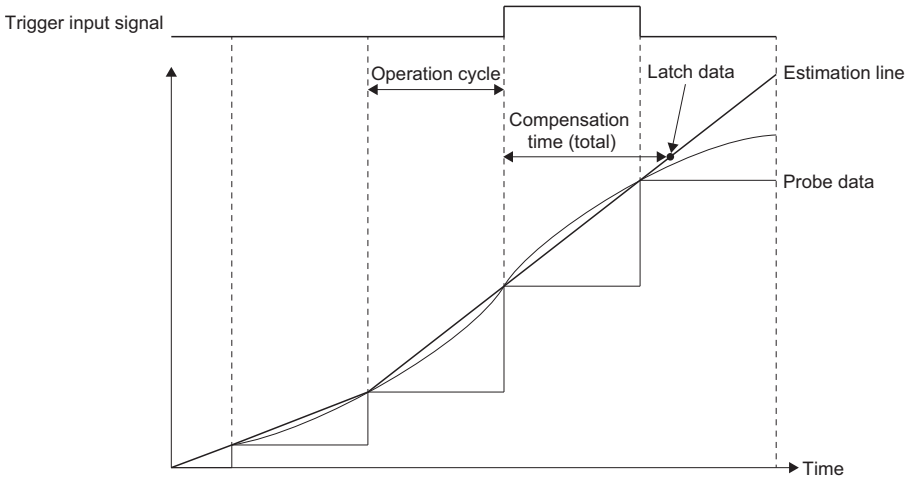
When compensating for the delay time, the past value is used for latch data estimate calculation. As such, the maximum accuracy is reached when the compensation time is passed after Executing (Busy) becomes TRUE. In addition, memory is consumed as shown below depending on the compensation time and the operation cycle.

- Memory consumption = 24 bytes × (total compensation time ÷ operation cycle)

- When compensating for delays (when the total compensation time is a positive number)
  The estimate is calculated based on the operation cycle data after latching and before the compensation time is passed.



- When compensating for the progression (when the total compensation time is a positive number)
  The estimate is calculated based on the operation cycle data at latching.



■Latch mode (RecordMode)

Sets the latch method for the data. The latch methods are shown below.

Setting value	Description
0: One shot mode (OneShot)	Stores latch data one time only.
1: Frequency specified mode (RecordCount)	• Stores latch data a number of times equal to the setting value of Number of latch times (RecordCount). • Starting from the storage location specified in Storage location of latch data (OutputBuffer), stores latch data in an area shifted for the latch data size each latch. • To stop the touch probe function before latching completes for the number of times set in Number of latch times (RecordCount), execute MC_AbortTrigger (Touch Probe Disabled).
2: Ring buffer mode (RingBuffer)	• Stores latch data to the ring buffer a number of times equal to the setting value of Number of latch times (RecordCount). The latch operation is always performed. • Starting from the storage location specified in Storage location of latch data (OutputBuffer), shifts for the latch data size each latch and stores latch data to the ring buffer. • To stop the touch probe function before latching completes for the number of times set in Number of latch times (RecordCount), execute MC_AbortTrigger (Touch Probe Disabled).

- Based on the setting of Latch mode (RecordMode), the following values are stored in Number of latch times (RecordedCounter) or Latch position (RecordedPosition).

Latch mode (RecordMode)	Number of latch times (RecordedCounter)	Latch position (RecordedPosition)
0: One shot mode (OneShot)	Becomes "1" at latch operation.	Stores latch data one time only.
1: Frequency specified mode (RecordCount)	Adds "+1" at each latch operation.	Stores the latest latch data at each latch operation.
2: Ring buffer mode (RingBuffer)	Adds "+1" at each latch operation. Returns to "0" when the value of Number of latch times (RecordCount) is exceeded.	Stores the latest latch data at each latch operation.

■Number of latch times (RecordCount)

Based on the setting of Latch mode (RecordMode), the following values are stored in Number of latch times (RecordedCount).

Latch mode (RecordMode)	Number of latch times (RecordCount)
0: One shot mode (OneShot)	Ignored.
1: Frequency specified mode (RecordCount)	Sets the number of latch times for the data. When omitted, "Latch Count Incorrect (error code: 3408H)" occurs.
2: Ring buffer mode (RingBuffer)	Sets the number of times for the ring buffer. When omitted, "Latch Count Incorrect (error code: 3408H)" occurs.

■Storage location of latch data (OutputBuffer)

Set the storage location of the latched data in the TARGET_REF structure. For the TARGET_REF structure, refer to the following.

☞ Page 64 TARGET_REF (Input signal)

- The usable data classifications are [VAR] and [DEV].
- As latched data will be stored as LREAL type, when setting non-LREAL type data in Target (Target) of the TARGET_REF structure, set "(LREAL)" explicitly to the data type.
- The storage location of the latched data is shown below.

Latch mode (RecordMode)	Storage location
0: One shot mode (OneShot)	The area specified in Storage location of latch data (OutputBuffer).
1: Frequency specified mode (RecordCount)	The following area: Area specified in Storage location of latch data (OutputBuffer) + Number of latch times (RecordedCounter) × latch data size
2: Ring buffer mode (RingBuffer)	

- Storage location of latch data (OutputBuffer) can be omitted. When omitted, the latch data is only stored in Latch position (RecordedPosition).
- A setting example in which Storage location of latch data (OutputBuffer) is set in Target (Target) of the TARGET_REF structure is shown below.

Ex.

When the following settings are used

Example	Description	Setting
1	• When Latch mode (RecordMode) is "0: One shot mode (OneShot)". • When Number of latch times (RecordCount) is "1" while Latch mode (RecordMode) is "1: Frequency specified mode (RecordCount)" or "2: Ring buffer mode (RingBuffer)".	• [DEV](LREAL)G11500000 • [VAR]OutputBufferData1*1
2	• When Number of latch times (RecordCount) is "100" while Latch mode (RecordMode) is "1: Frequency specified mode (RecordCount)" or "2: Ring buffer mode (RingBuffer)".	• [DEV](LREAL[0..99])G11500000 • [VAR]OutputBufferData2*2 • [VAR](LREAL)OutputBufferData3*3

*1 "OutputBufferData1" is a LREAL type instance
*2 "OutputBufferData2" is a LREAL[0..99] type instance
*3 "OutputBufferData3" is a WORD[0..399] type instance

- When the number of elements set in Storage location of latch data (OutputBuffer) is less than the value of Number of latch times (RecordCount), "Latch Data Storage Shortage (error code: 349FH)" occurs. "Latch Data Storage Shortage (error code: 349FH)" also occurs when access outside the device or label range is made during latch operation.
- The relation between the setting value and operation of each item is shown below.

Latch mode (RecordMode)	Number of latch times (RecordCount)	Storage location of latch data (OutputBuffer)	Detection operation	Data storage location
0: One shot mode (OneShot)	Omit/Set	Omit	One time	Latch position (RecordedPosition)
		Set		Latch position (RecordedPosition)/buffer
1: Frequency specified mode (RecordCount)	Omit	Omit/Set	Error	—
	Set	Omit	Setting number	Latch position (RecordedPosition)
		Set		Latch position (RecordedPosition)/buffer
2: Ring buffer mode (RingBuffer)	Omit	Omit/Set	Error	—
	Set	Omit	Always	Latch position (RecordedPosition)
		Set		Latch position (RecordedPosition)/buffer

Precautions

- When operation cycle over occurs after Executing (Busy) of MC_TouchProbe (Touch Probe Enabled) becomes TRUE, the accuracy of the estimate calculation may decrease.
- The I/O argument refresh for Motion control FBs is performed when the FB is called, but control is performed in the operation cycle regardless of the FB call task (Normal/Operation cycle).
- Executes MC_TouchProbe (Touch Probe Enabled) even if the target axis is unconnected, has an error occur on it, etc.

Program example

In the following program example, touch probe start (bTouchProbe) is set to TRUE and the actual position of Axis 1 (Axis0001) is calculated by estimation at each rise of the touch probe signal (G_bTouchProbeSignal) in the Motion system. After the actual position is stored in Latch position (leRecordedPosition), Touch Probe Disabled start (bAbortTrigger) is set to TRUE and the latch operation is stopped.

■Axis

Axis No.	Label name	Data type	Comment
1	Axis0001	AXIS_REF	Axis 1

■I/O data

Data name	Data category	Data type	Source category	Source	Signal detection method
SignalData0001	Trigger signal	MC_TRIGGER_REF	Global label	G_bTouchProbeSignal	Detection at FALSE→TRUE (rising edge) (RisingEdge)

■Global labels used

Label name	Data type	Comment	Public label
G_bTouchProbeSignal	Bit	Touch probe signal	Enabled

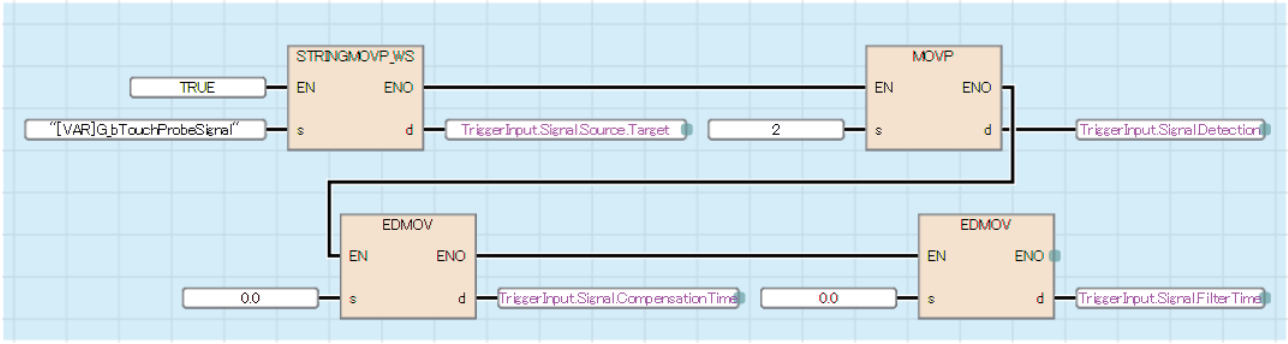
■Labels used

Label name	Data type	Comment
MC_TouchProbe_1	MC_TouchProbe	Touch Probe Enabled FB
bTouchProbe	Bit	Touch probe start
leFirstPosition	Double-precision real number	Lower limit position
leLastPosition	Double-precision real number	Upper limit position
leCompensationTime	Double-precision real number	Compensation time
ProbeData	TARGET_REF	Probe data
uwRecordCount	Word [unsigned]/bit string [16-bit]	Number of latch times
OutputBuffer	TARGET_REF	Storage location of latch data
bTPDone	Bit	Touch Probe Enabled Completion
bTPBusy	Bit	Touch Probe Enabled Executing
bTPCommandAborted	Bit	Touch Probe Enabled Abortion of execution
bTPError	Bit	Touch Probe Enabled Error
uwTPErrorID	Word [unsigned]/bit string [16-bit]	Touch Probe Enabled Error code
lePositionData	Double-precision real number	Current value monitor data
leRecordedPosition	Double-precision real number	Latch position
uwRecordedCounter	Word [unsigned]/bit string [16-bit]	Number of latch times
uwTouchProbeID	Word [unsigned]/bit string [16-bit]	Probe ID
leLatchPos_array	Double-precision real number(0..1)	Variable for storing latch position
MC_AbortTrigger_1	MC_AbortTrigger	Touch Probe Disabled FB
bAbortTrigger	Bit	Touch Probe Disabled start
bATDone	Bit	Touch Probe Disabled Completion
bATBusy	Bit	Touch Probe Disabled processing
bATCommandAborted	Bit	Touch Probe Disabled Abortion of execution
bATError	Bit	Touch Probe Disabled Error
uwATErrorID	Word [unsigned]/bit string [16-bit]	Touch Probe Disabled Error code
TriggerInput*1	MC_TRIGGER_REF	Trigger input signal

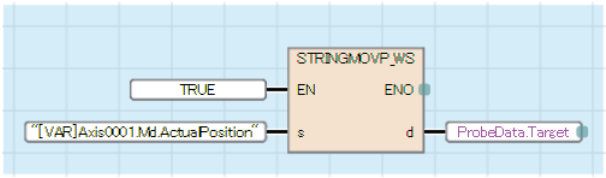
*1 Only used in FBD/LD programs.

■FBD/LD program (CPU module side)

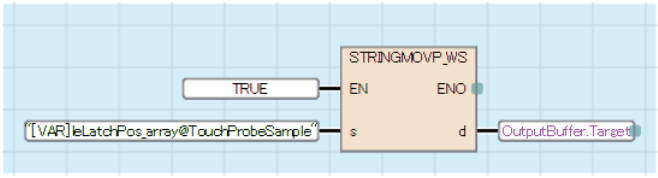
- Trigger input signal (I/O data)



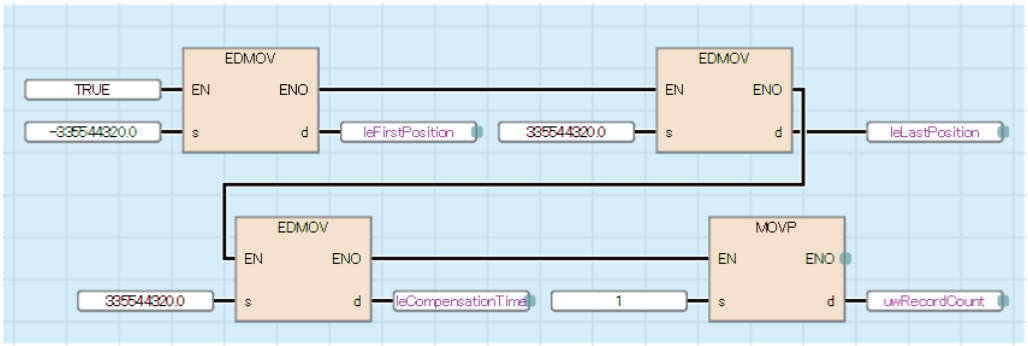
- Data for latch target



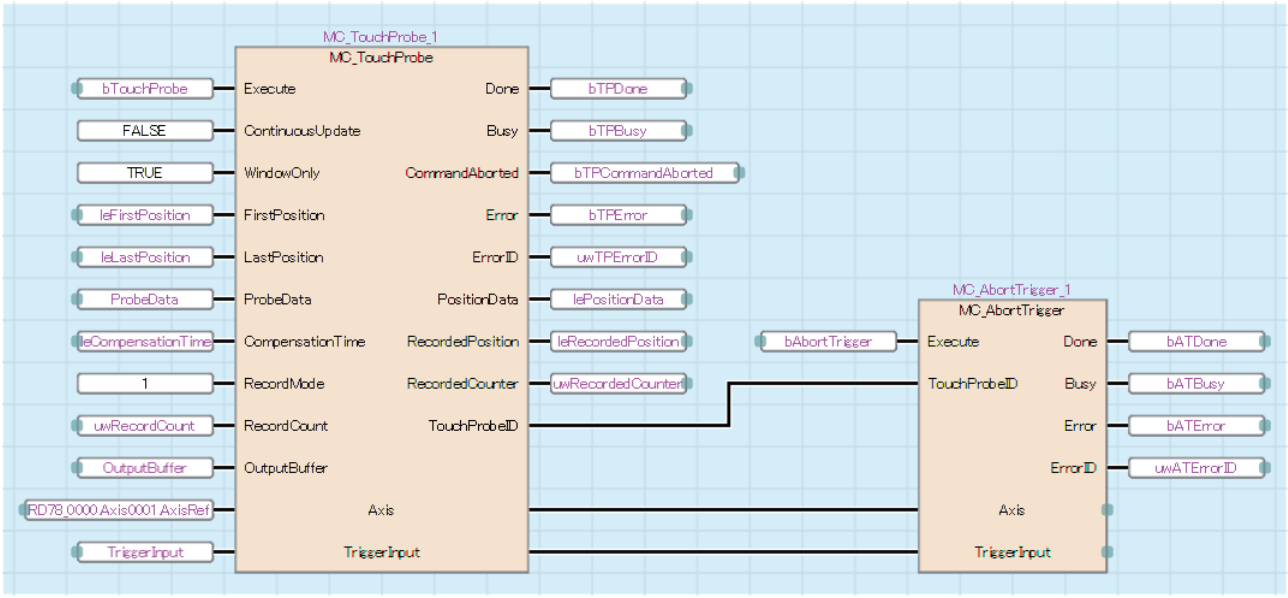
- Touch probe processing



- Data setting for touch probe



- Touch Probe Enabled/Disabled

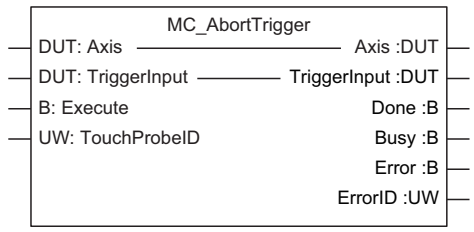
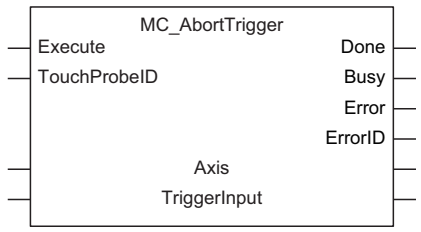


■ST program (Motion module side)

```
//-----Data for latch target-----
ProbeData.Target:= "[VAR]Axis0001.Md.ActualPosition";

//-----Touch probe processing-----
OutputBuffer.Target:= "[VAR]leLatchPos_array@TouchProbeSample";

//-----Data setting for touch probe-----
leFirstPosition:= -335544320.0;
leLastPosition:= 335544320.0;
leCompensationTime:= 0.0;
uwRecordCount:= 1;

//-----Touch Probe Enabled-----
MC_TouchProbe_1(
   Axis:= Axis0001.AxisRef ,
   TriggerInput:= SignalData0001 ,
   Execute:= bTouchProbe ,
   ContinuousUpdate:= FALSE ,
   WindowOnly:= TRUE ,
   FirstPosition:= leFirstPosition ,
   LastPosition:= leLastPosition ,
   ProbeData:= ProbeData ,
   CompensationTime:= leCompensationTime ,
   RecordMode:= MC_RECORD_MODE__RecordCount ,
   RecordCount:= uwRecordCount ,
   OutputBuffer:= OutputBuffer ,
   Done=> bTPDone ,
   Busy=> bTPBusy ,
   CommandAborted=> bTPCommandAborted ,
   Error=>  bTPError ,
   ErrorID=> uwTPErrorID ,
   PositionData=> lePositionData ,
   RecordedPosition=> leRecordedPosition ,
   RecordedCounter=> uwRecordedCounter ,
   TouchProbeID=> uwTouchProbeID
);

//-----Touch Probe Disabled-----
MC_AbortTrigger_1(
   Axis:= Axis0001.AxisRef ,
   TriggerInput:= SignalData0001 ,
   Execute:= bAbortTrigger ,
   TouchProbeID:= MC_TouchProbe_1.TouchProbeID ,
   Done=> bATDone ,
   Busy=>  bATBusy ,
   Error=>  bATError ,
   ErrorID=> uwATErrorID
);
```

MC_AbortTrigger

Disables the latch that is being executed.

Ladder	FBD/LD	ST
		MC_AbortTrigger(Axis:= ?AXIS_REF? , TriggerInput:= ?MC_TRIGGER_REF? , Execute:= ?BOOL? , TouchProbeID:= ?WORD? , Done=> ?BOOL? , Busy=> ?BOOL? , Error=> ?BOOL? , ErrorID=> ?WORD?);

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Touch Probe Disabled	172	6	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
Axis	Axis information	AXIS_REF	At start	—	Omission possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)
TriggerInput	Trigger input signal	MC_TRIGGER_REF	At start	—	Omission possible	This variable sets the trigger input signal. For details, refer to the following. ☞ Page 149 Trigger input signal (TriggerInput)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Execute command	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MC_AbortTrigger (Touch Probe Disabled).
TouchProbeID	Touch probe ID	WORD(UINT)	At start	1 to 65535	0	This variable sets the specific ID of the touch probe to disable. Sets the touch probe ID to store in Probe ID (TouchProbeID) that was started by Execute command (Execute) of MC_TouchProbe (Touch Probe Enabled).

■Output variables

Output variable	Name	Data type	Default value	Description
Done	Execution completion	BOOL	FALSE	This variable becomes TRUE when the output value is valid.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MC_AbortTrigger (Touch Probe Disabled) is executed.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. MELSEC iQ-R Motion Module User's Manual (Application)

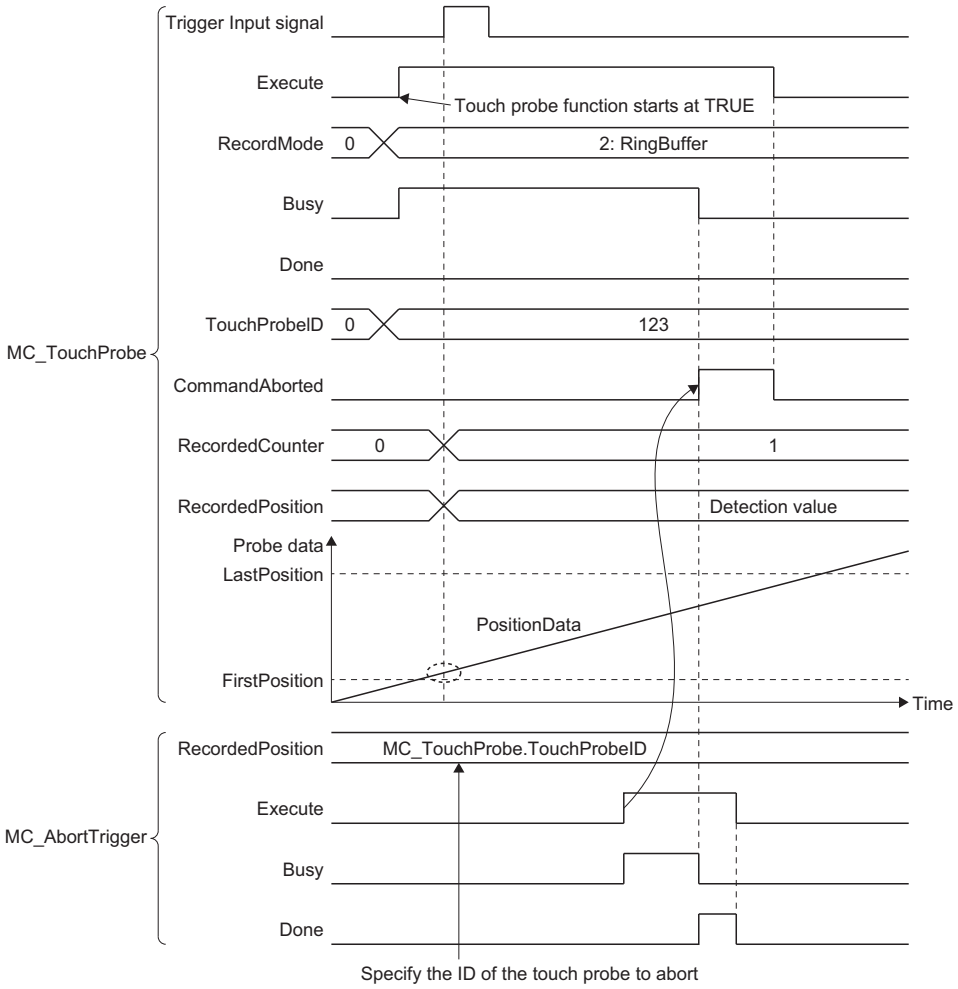
Processing details

- Disables the MC_TouchProbe (Touch Probe Enabled) set in Touch probe ID (TouchProbeID).
- When MC_AbortTrigger (Touch Probe Disabled) is executed, if the MC_TouchProbe (Touch Probe Enabled) set in Touch probe ID (TouchProbeID) is not in operation, Execution completion (Done) immediately becomes TRUE.
- Axis information (Axis) and Trigger input signal (TriggerInput) are ignored and can be omitted.

■Timing chart

- When the operation completes normally

  Latch mode (RecordMode) is "2: Ring buffer mode (RingBuffer)"



- When an error occurs

  For the timing chart when an error occurs, refer to the following.

  ☞ Page 11 Basic operation of Execute command (Execute) type Motion control FBs

■Trigger input signal (TriggerInput)

Trigger input signal is set in the MC_TRIGGER_REF structure. For the MC_TRIGGER_REF structure, refer to the following.

☞ Page 63 MC_TRIGGER_REF

The setting range for Trigger input signal (Signal) of the MC_TARGET_REF structure set in Trigger input signal (TriggerInput) is shown below.

The type for Trigger signal (Signal) is SIGNAL_SELECT structure. For the SIGNAL_SELECT structure, refer to the following.

☞ Page 66 SIGNAL_SELECT (Signal select)

Structure	Variable name	Type	Setting range
SIGNAL_SELECT (Signal select)	Source (Signal)	TARGET_REF ☞ Page 64 TARGET_REF (Input signal)	■Type • BOOL ■Data category • [VAR] • [DEV] • [CONST]
	Detection (Signal detection method)	INT (MC_SIGNAL_LOGIC) ☞ Page 79 MC_SIGNAL_LOGIC	• 2:Detection at FALSE→TRUE (rising edge) (RisingEdge) • 3: Detection at TRUE→FALSE (falling edge) (FallingEdge) • 4: Detection at rising edge/falling edge (BothEdges)
	CompensationTime (Compensation time)	LREAL	-5.0 to 5.0[s]
	FilterTime (Filter time)	LREAL	0.0 to 5.0[s]

Program example

In the following program example, touch probe start (bTouchProbe) is set to TRUE and the actual position of Axis 1 (Axis0001) is calculated by estimation at each rise of the touch probe signal (G_bTouchProbeSignal) in the Motion system. After the actual position is stored in Latch position (leRecordedPosition), Touch Probe Disabled start (bAbortTrigger) is set to TRUE and the latch operation is stopped.

■Axis

Axis No.	Label name	Data type	Comment
1	Axis0001	AXIS_REF	Axis 1

■I/O data

Data name	Data category	Data type	Source category	Source	Signal detection method
SignalData0001	Trigger signal	MC_TRIGGER_REF	Global label	G_bTouchProbeSignal	Detection at FALSE→TRUE (rising edge) (RisingEdge)

■Global labels used

Label name	Data type	Comment	Public label
G_bTouchProbeSignal	Bit	Touch probe signal	Enabled

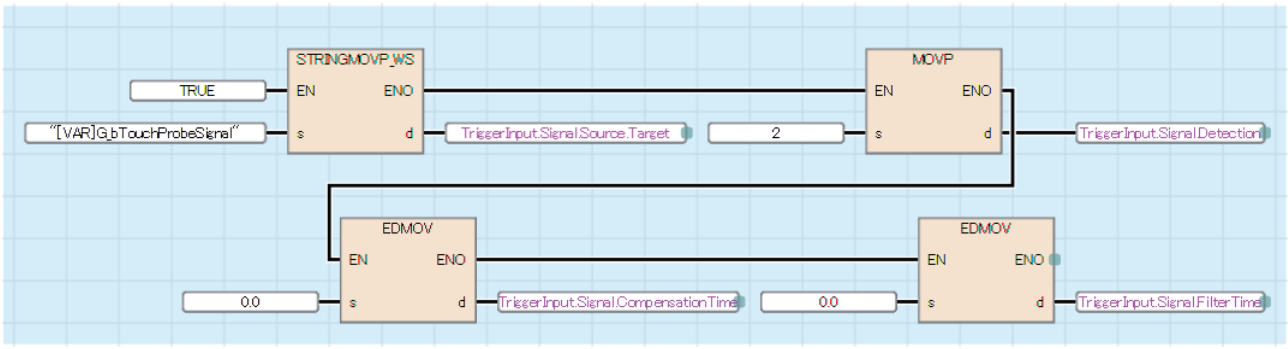
■Labels used

Label name	Data type	Comment
MC_TouchProbe_1	MC_TouchProbe	Touch Probe Enabled FB
bTouchProbe	Bit	Touch probe start
leFirstPosition	Double-precision real number	Lower limit position
leLastPosition	Double-precision real number	Upper limit position
leCompensationTime	Double-precision real number	Compensation time
ProbeData	TARGET_REF	Probe data
uwRecordCount	Word [unsigned]/bit string [16-bit]	Number of latch times
OutputBuffer	TARGET_REF	Storage location of latch data
bTPDone	Bit	Touch Probe Enabled Completion
bTPBusy	Bit	Touch Probe Enabled Executing
bTPCommandAborted	Bit	Touch Probe Enabled Abortion of execution
bTPError	Bit	Touch Probe Enabled Error
uwTPErrorID	Word [unsigned]/bit string [16-bit]	Touch Probe Enabled Error code
lePositionData	Double-precision real number	Current value monitor data
leRecordedPosition	Double-precision real number	Latch position
uwRecordedCounter	Word [unsigned]/bit string [16-bit]	Number of latch times
uwTouchProbeID	Word [unsigned]/bit string [16-bit]	Probe ID
leLatchPos_array	Double-precision real number(0..1)	Variable for storing latch position
MC_AbortTrigger_1	MC_AbortTrigger	Touch Probe Disabled FB
bAbortTrigger	Bit	Touch Probe Disabled start
bATDone	Bit	Touch Probe Disabled Completion
bATBusy	Bit	Touch Probe Disabled processing
bATCommandAborted	Bit	Touch Probe Disabled Abortion of execution
bATError	Bit	Touch Probe Disabled Error
uwATErrorID	Word [unsigned]/bit string [16-bit]	Touch Probe Disabled Error code
TriggerInput*1	MC_TRIGGER_REF	Trigger input signal

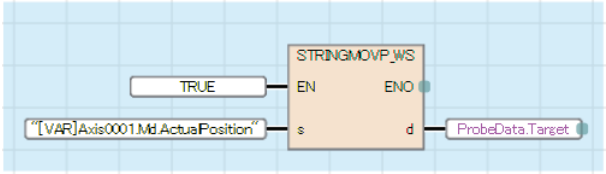
*1 Only used in FBD/LD programs.

■FBD/LD program (CPU module side)

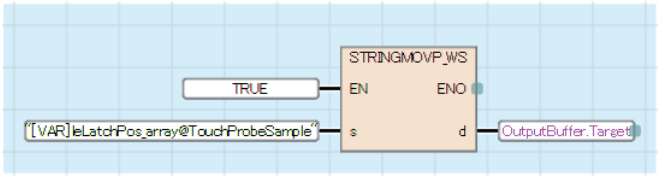
- Trigger input signal (I/O data)



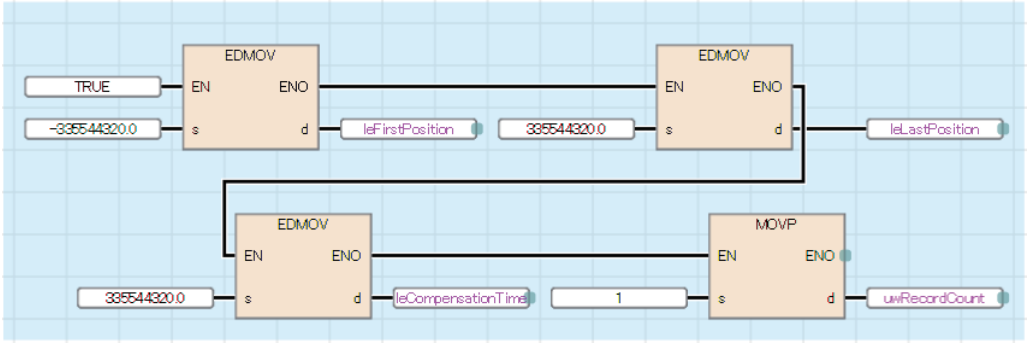
- Data for latch target



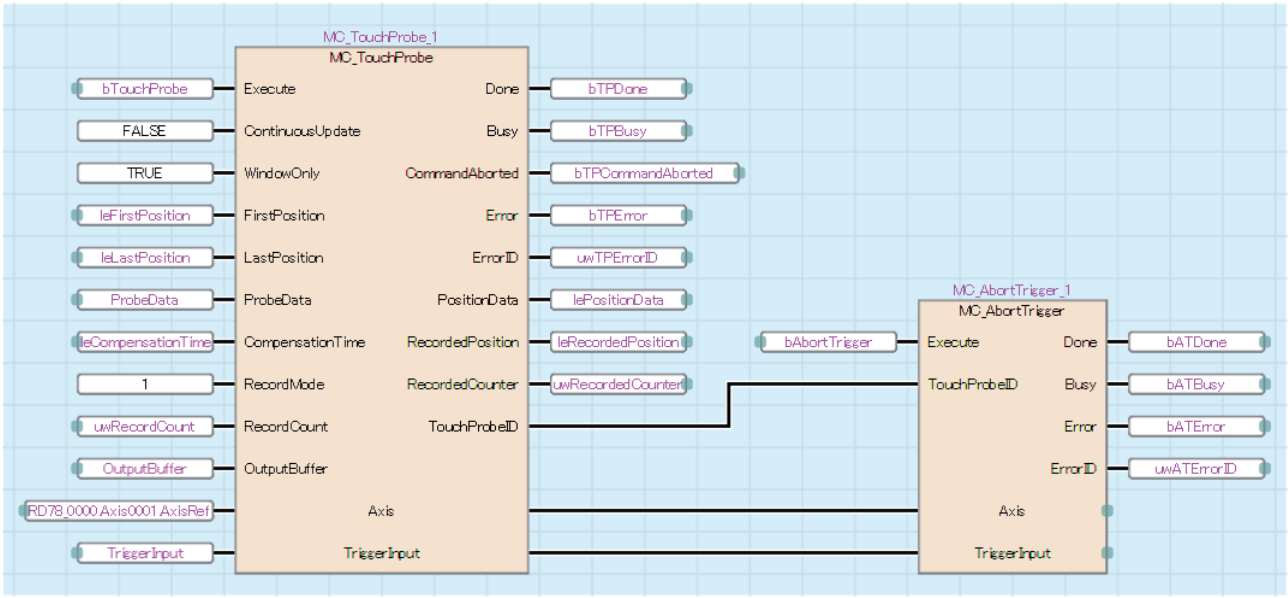
- Touch probe processing



- Data setting for touch probe



- Touch Probe Enabled/Disabled



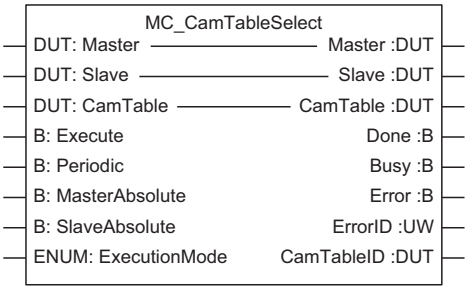
■ST program (Motion module side)

```
//-----Data for latch target-----
ProbeData.Target:= "[VAR]Axis0001.Md.ActualPosition";

//-----Touch probe processing-----
OutputBuffer.Target:= "[VAR]leLatchPos_array@TouchProbeSample";

//-----Data setting for touch probe-----
leFirstPosition:= -335544320.0;
leLastPosition:= 335544320.0;
leCompensationTime:= 0.0;
uwRecordCount:= 1;

//-----Touch Probe Enabled-----
MC_TouchProbe_1(
   Axis:= Axis0001.AxisRef ,
   TriggerInput:= SignalData0001 ,
   Execute:= bTouchProbe ,
   ContinuousUpdate:= FALSE ,
   WindowOnly:= TRUE ,
   FirstPosition:= leFirstPosition ,
   LastPosition:= leLastPosition ,
   ProbeData:= ProbeData ,
   CompensationTime:= leCompensationTime ,
   RecordMode:= MC_RECORD_MODE__RecordCount ,
   RecordCount:= uwRecordCount ,
   OutputBuffer:= OutputBuffer ,
   Done=> bTPDone ,
   Busy=> bTPBusy ,
   CommandAborted=> bTPCommandAborted ,
   Error=>  bTPError ,
   ErrorID=> uwTPErrorID ,
   PositionData=> lePositionData ,
   RecordedPosition=> leRecordedPosition ,
   RecordedCounter=> uwRecordedCounter ,
   TouchProbeID=> uwTouchProbeID
);

//-----Touch Probe Disabled-----
MC_AbortTrigger_1(
   Axis:= Axis0001.AxisRef ,
   TriggerInput:= SignalData0001 ,
   Execute:= bAbortTrigger ,
   TouchProbeID:= MC_TouchProbe_1.TouchProbeID ,
   Done=> bATDone ,
   Busy=>  bATBusy ,
   Error=>  bATError ,
   ErrorID=> uwATErrorID
);
```

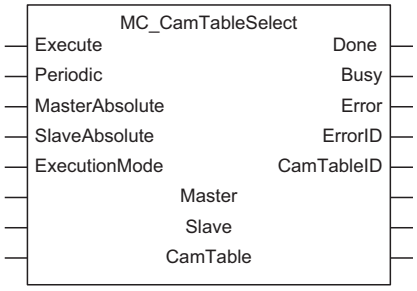
MC_CamTableSelect

Stores the specified calculation profile (cam data) in the open area.

Ladder



FBD/LD



ST

```
MC_CamTableSelect(
   Master:= ?AXIS_REF? ,
   Slave:= ?AXIS_REF? ,
   CamTable:= ?MC_CAM_REF? ,
   Execute:= ?BOOL? ,
   Periodic:= ?BOOL? ,
   MasterAbsolute:= ?BOOL? ,
   SlaveAbsolute:= ?BOOL? ,
   ExecutionMode:= ?INT? ,
   Done=> ?BOOL? ,
   Busy=> ?BOOL? ,
   Error=> ?BOOL? ,
   ErrorID=> ?WORD? ,
   CamTableID=> ?MC_CAM_ID?
);
```

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Cam Table Selection	158	18	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
Master	Master axis	AXIS_REF	At start	—	Omission possible	When using MC_CamTableSelect (Cam Table Selection) on the Motion module side, this variable can be omitted as the setting is ignored. When using MC_CamTableSelect (Cam Table Selection) on the CPU module side, this variable sets IO No. (Master.StartIO). When used on the CPU module side, Axis No. (Master.AxisNo) is ignored. ☞ Page 23 AxisName.AxisRef. (Axis information)
Slave	Slave axis	AXIS_REF	At start	—	Omission possible	When using MC_CamTableSelect (Cam Table Selection) on the Motion module side, this variable can be omitted as the setting is ignored. When using MC_CamTableSelect (Cam Table Selection) on the CPU module side, this variable sets IO No. (Slave.StartIO). When used on the CPU module side, Axis No. (Slave.AxisNo) is ignored. ☞ Page 23 AxisName.AxisRef. (Axis information)
CamTable	Cam table	MC_CAM_REF	At start	—	Omission not possible	This variable sets the calculation profile (cam data). For details, refer to the following. ☞ Page 155 Cam table (CamTable)

Point

When using MC_CamTableSelect (Cam Table Selection) on the CPU module side, IO No. (StartIO) of Master axis (Master) and Slave axis (Slave) is used, but Axis No. (AxisNo) of Master axis (Master) and Slave axis (Slave) is not.

When using MC_CamTableSelect (Cam Table Selection) on the Motion module side, Master axis (Master) and Slave axis (Slave) do not provide axis information in the cam data and are not used in control even if set.

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Start	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MC_CamTableSelect (Cam Table Selection).
Periodic	Periodic	BOOL	At start	TRUE, FALSE	FALSE	This variable sets the operation to execute the calculation profile (cam data). • FALSE: Periodic • TRUE: Non periodic For details, refer to the following. ☞ Page 156 Periodic (Periodic)
MasterAbsolute	Master axis absolute coordinate	BOOL	At start	FALSE	FALSE	This variable sets the coordinate of the master axis. • FALSE: Relative coordinate • TRUE: Setting not possible For details, refer to the following. ☞ Page 156 Master axis absolute coordinate (MasterAbsolute) *: When this variable is set to TRUE, "Out of Input Absolute Coordinate Range (error code: 341DH)" occurs.
SlaveAbsolute	Slave axis absolute coordinate	BOOL	At start	TRUE, FALSE	FALSE	This variable sets the coordinate of the slave axis. • FALSE: Relative coordinate • TRUE: Absolute coordinate For details, refer to the following. ☞ Page 157 Slave axis absolute coordinate (SlaveAbsolute)
ExecutionMode	Execution mode	INT (MC_EXECUTION_MODE)	At start	0, 1, 3	0	This variable sets the timing for executing MC_CamTableSelect (Cam Table Selection). • 0: Execute immediately (mcImmediately) • 1: Execute at completion (mcQueued) • 3: Execute speculatively (mcSpeculatively) For details, refer to the following. ☞ Page 157 Execution mode (ExecutionMode)

■Output variables

Output variable	Name	Data type	Default value	Description
Done	Completion	BOOL	FALSE	This variable becomes TRUE when control is completed. At completion of operation, this variable changes according to the status of Start (Execute). ■When Start (Execute) is TRUE This variable remains TRUE until Start (Execute) becomes FALSE. ■When Start (Execute) is FALSE This variable becomes TRUE for one cycle only.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MC_CamTableSelect (Cam Table Selection) is executed.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. 📖MELSEC iQ-R Motion Module User's Manual (Application)
CamTableID	Cam table ID	MC_CAM_ID	0	This variable outputs the profile ID.

Processing details

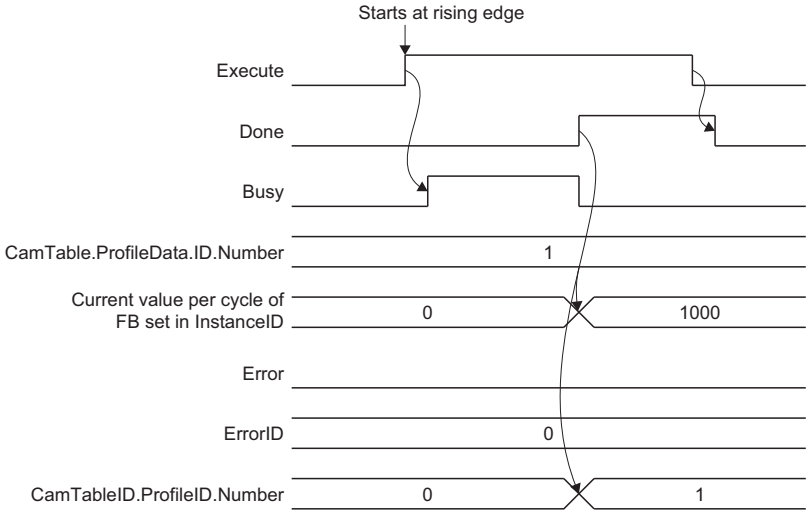
- Opens the calculation profile (cam data) file set in Calculation profile storage location (Location) of the PROFILE_DATA structure in the open area by using MC_CamTableSelect (Cam Table Selection).
- Data opened in the open area is assigned the Profile ID (Number) of the Profile_ID structure.

Point

- For details of creating calculation profiles (cam data), refer to the following.
  MELSEC iQ-R Motion Module User's Manual (Application)
- MC_CamTableSelect (Cam Table Selection) is performed with Motion service processing. It may take time to open depending on the processing details and the calculation profile.
- When Profile_ID (ID) of the PROFILE_DATA structure, Periodic (Periodic), Master axis absolute coordinate (MasterAbsolute), and Slave axis absolute coordinate (SlaveAbsolute) are present as input variables for MC_CamTableSelect (Cam Table Selection), the file setting values are ignored and the open is performed while referencing the input variable settings. (The setting values of the file will not be overwritten.)

■Timing chart

- When the operation completes normally



- When an error occurs
  For the timing chart when an error occurs, refer to the following.
  Page 11 Basic operation of Execute command (Execute) type Motion control FBs

■Cam table (CamTable)

Sets the calculation profile with the MC_CAM_REF type PROFILE_DATA structure. For the PROFILE_DATA structure, refer to the following.
Page 66 PROFILE_DATA
References Calculation profile storage location (Location) of the PROFILE_DATA structure and performs open for the calculation profile that File name (FileName) and Folder specification (Path) indicate.
Profile ID No. (Number) of the PROFILE_ID structure indicated by Profile ID (ID) of the PROFILE_DATA structure is assigned in the open area.
The operation changes depending on the value set in Profile ID No. (Number).

Profile ID No. (Number)	Operation
When set to "0"	An available ID is automatically assigned and stored in Profile ID No. (Number). If an MC_CamTableSelect (Cam Table Selection) with Profile ID No. (Number) set to "0" is executed multiple times to the same calculation profile, the data will be opened in multiple different open areas. When no ID is available, "Insufficient Calculation Profile ID (error code: 3452H)" occurs.
When set to "1 to 60000"	The open is performed without changing the set ID. If specifying an ID that has already been opened, the open data will be overwritten.

The relationship between the value of Profile ID No. (Number) at open and the open location area is shown below.

Profile ID No. (Number) value		Open location area
Before executing MC_CamTableSelect (Cam Table Selection)	After executing MC_CamTableSelect (Cam Table Selection)	
0	Stores 1 to 60000	New area
1 to 60000		Open data is overwritten if already existing. When open data does not exist, a new open area is assigned.

- If the calculation profile set in Calculation profile storage location (Location) does not exist, "No Calculation Profile (error code: 340FH)" occurs and the open processing is not performed.
- If the calculation profile set in Calculation profile storage location (Location) is broken or there is a problem in the format, "Calculation Profile Error (error code: 3412H)" occurs and the open processing is not performed.

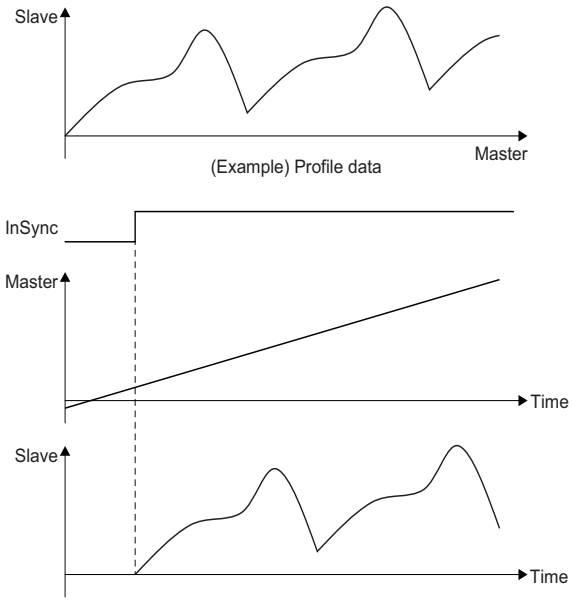
■Periodic (Periodic)

Sets the operation performed by Periodic (Periodic) at calculation profile control execution.

Setting value	Description
FALSE (Non periodic)	The control ends when it executes until the end point of the calculation profile.
TRUE (Periodic)	Continuously repeats calculation profile execution.

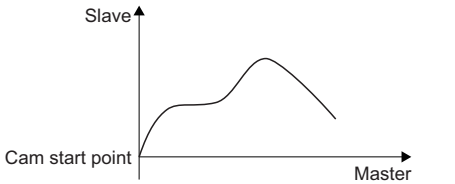
■Master axis absolute coordinate (MasterAbsolute)

Sets the operation performed by Master axis absolute coordinate (MasterAbsolute) at calculation profile control execution. (When the input unit is time specification, it is ignored.)

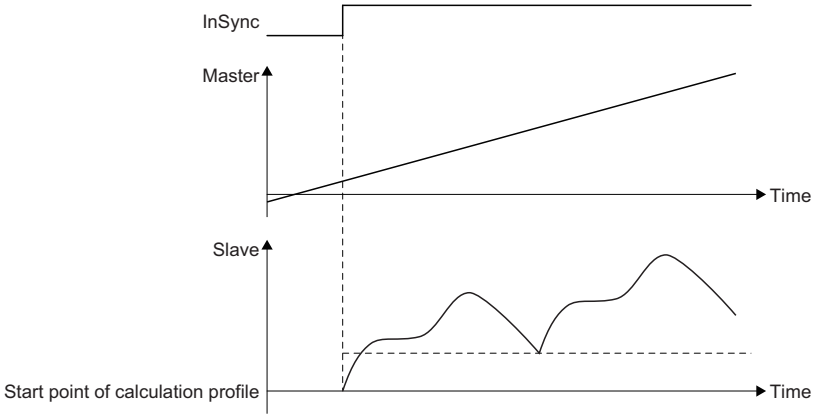
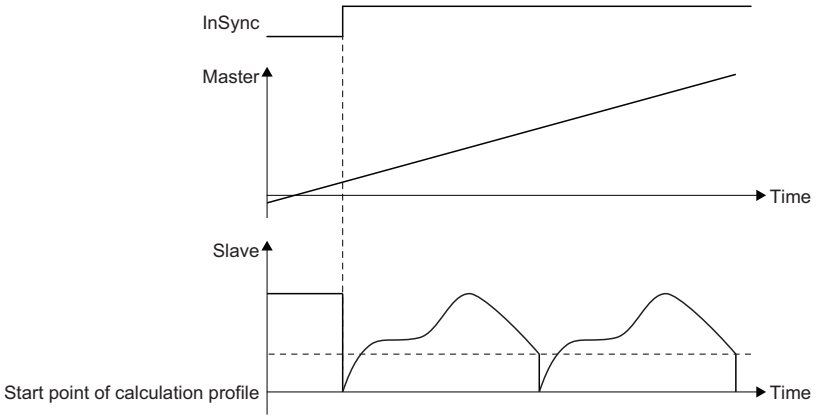
Setting value	Description
FALSE (Relative coordinate)	The current Master axis (Master) value becomes the start point of the calculation profile. When In synchronization (InSync) becomes TRUE, Slave axis (Slave) is operated according to the relative movement amount of Master axis (Master). 
TRUE	Cannot be set to TRUE. If set to TRUE, "Out of Input Absolute Coordinate Range (error code: 341DH)" occurs.

■Slave axis absolute coordinate (SlaveAbsolute)

Sets the operation performed by Slave axis absolute coordinate (SlaveAbsolute) at calculation profile control execution.



<Example> Start point of calculation profile

Setting value	Description
FALSE (Relative coordinate)	When In synchronization (InSync) becomes TRUE, Slave axis (Slave) is operated based on the current value. 
TRUE (Absolute coordinate)	Operates so that Slave axis (Slave) at the time that In synchronization (InSync) becomes TRUE is always the start point at the start of each cycle of the calculation profile. When there is a limit on the value of Master axis (Master), such as the cam data length per cycle, and the start point and end point of the calculation profile are different when Periodic (Periodic) is "TRUE: Periodic", the command is output in one operation cycle in order to return to the first Slave axis (Slave) at the start of the next cycle. 

■Execution mode (ExecutionMode)

Sets the timing for executing MC_CamTableSelect (Cam Table Selection).

Setting value	Description
0: Execute immediately (mcImmediately)	Immediately writes the contents of the open area. The writing may affect the control in execution. Note that if the calculation profile format and resolution do not match when performing open during FB execution, "Calculation Profile Operating (error code: 3411H)" occurs.
1: Execute at completion (mcQueued)	Opens after waiting for the FB being executed to complete. When multiple FBs are waiting, the FBs will be executed in order of task priority. FBs of the same priority are executed according to the start order.
3: Execute speculatively (mcSpeculatively)	"Calculation Profile Operating (error code: 3411H)" occurs and the open area is not changed.

Point

When no FB is in execution, the FB is executed immediately regardless of the setting of Execution mode (ExecutionMode). When FBs in execution are accessing the open area, Executing (Busy) of each FB accessing the open area becomes TRUE. Use Execution mode (ExecutionMode) for the interlock as needed.

Precautions

- When rewriting the open area during control, some waveform patterns may cause the stroke value may change rapidly and apply shock to the machine. Create calculation profiles so that the waveform does not become discontinuous before or after the stroke value changes.

Program example

In the following program example, the cam table select command (bCamTableSelect) is set to TRUE and the calculation profile (Cam data 1) is opened in the open area and assigned a profile ID.

■Axis

Axis No.	Label name	Data type	Comment
1	Axis0001	AXIS_REF	Axis 1

■Calculation profile

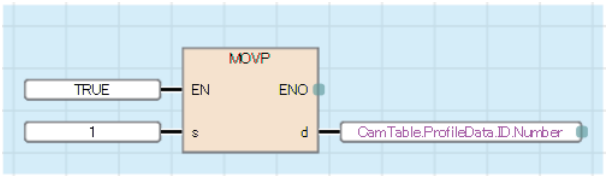
Profile ID	Label name	Data type	Comment
1	ProfileData0001	MC_CAM_REF	Cam data 1

■Labels used

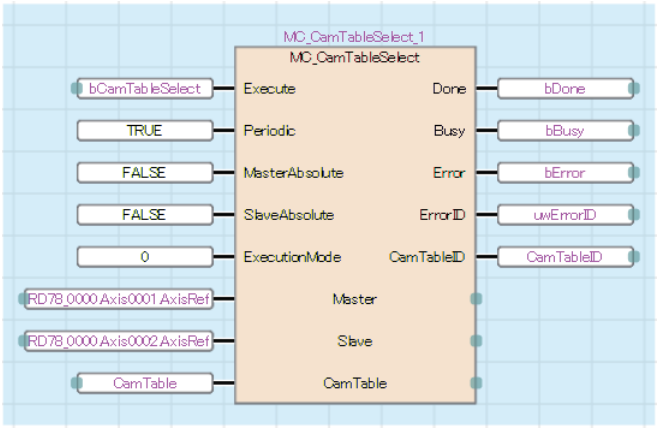
Label name	Data type	Comment
MC_CamTableSelect_1	MC_CamTableSelect	Cam Table Selection FB
CamTable	MC_CAM_REF	Cam table
bCamTableSelect	Bit	Cam table select command
bDone	Bit	Execution completion
bBusy	Bit	Executing
bError	Bit	Error
uwErrorID	Word [unsigned]/bit string [16-bit]	Error code
CamTableID	MC_CAM_ID	Cam table ID

■FBD/LD program (CPU module side)

- Calculation profile (cam data) setting



- Cam Table Selection



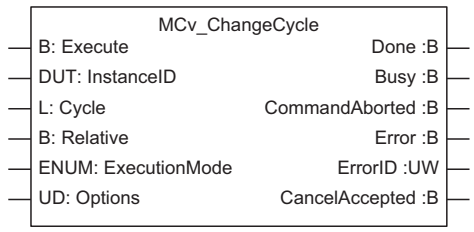
■ST program (Motion module side)

```
//-----Calculation profile (cam data) setting-----
CamTable.ProfileData:= ProfileData0001.ProfileData;

//-----Cam Table Selection-----
MC_CamTableSelect_1(
    CamTable:= CamTable ,
    Execute:= bCamTableSelect ,
    Periodic:= TRUE ,
    MasterAbsolute:= FALSE ,
    SlaveAbsolute:= FALSE ,
    ExecutionMode:= MC_EXECUTION_MODE__mcImmediately ,
    Done=> bDone ,
    Busy=> bBusy ,
    Error=> bError ,
    ErrorID=> uwErrorID ,
    CamTableID=> CamTableID
);
```

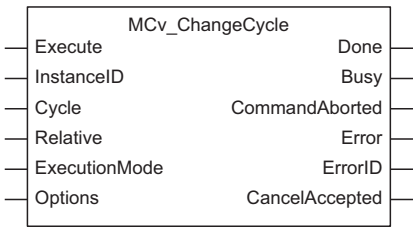
MCv_ChangeCycle

Changes the current value per cycle of the specified calculation profile control FB.

Ladder



FBD/LD



ST

```
MCv_ChangeCycle(
   Execute:= ?BOOL? ,
   InstanceID:= ?INSTANCE_ID? ,
   Cycle:= ?LREAL(0..2)? ,
   Relative:= ?BOOL? ,
   ExecutionMode:= ?INT? ,
   Options:= ?DWORD? ,
   Done=> ?BOOL? ,
   Busy=> ?BOOL? ,
   CommandAborted=> ?BOOL? ,
   Error=> ?BOOL? ,
   ErrorID=> ?WORD? ,
   CancelAccepted=> ?BOOL?
);
```

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Current Value Change per Cycle	48	8	Subroutine type	Real-time execution

Setting data

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Start	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MCv_ChangeCycle (Current Value Change per Cycle).
InstanceID	Instance ID	INSTANCE_ID	At start	1 to 65535	0	This variable sets the instance ID of the calculation profile control FB whose current value per cycle will be changed. The instance ID is valid until "PLC READY (Y0)" turns ON→OFF.
Cycle	Current value per cycle	LREAL[0..2]	At start	—	0.0	This variable sets the current value per cycle to change. • Sets the absolute position when Relative selection (Relative) is FALSE. • Sets the relative distance when Relative selection (Relative) is TRUE. For details, refer to the following. ☞ Page 162 Current value per cycle (Cycle)
Relative	Relative selection	BOOL	At start	TRUE, FALSE	FALSE	This variable sets the method used to change the current value per cycle • FALSE: Absolute position • TRUE: Relative distance For details, refer to the following. ☞ Page 163 Relative Selection (Relative)
ExecutionMode	Execution mode	INT (MC_EXECUTION_MODE)	At start	0, 1, 3	0	This variable sets the timing for executing MCv_ChangeCycle (Current Value Change per Cycle). • 0: Execute immediately (mcImmediately) • 1: Execute at completion (mcQueued) • 3: Execute speculatively (mcSpeculatively) For details, refer to the following. ☞ Page 164 Execution mode (ExecutionMode)
Options	Options	DWORD(HEX)	At start	00000000H to 00010000H	00000000H	This variable sets the function options for MCv_ChangeCycle (Current Value Change per Cycle) by specifying the bit. For details, refer to the following. ☞ Page 166 Options (Options)

■Output variables

Output variable	Name	Data type	Default value	Description
Done	Completion	BOOL	FALSE	This variable becomes TRUE when the control is completed. This variable changes as shown below based on the status of Start (Execute) at operation completion. ■When Start (Execute) is TRUE This variable remains TRUE until Start (Execute) becomes FALSE. ■When Start (Execute) is FALSE This variable becomes TRUE for one cycle only.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MCv_ChangeCycle (Current Value Change per Cycle) is executed.
CommandAborted	Abortion of execution	BOOL	FALSE	This variable becomes TRUE when execution of MCv_ChangeCycle (Current Value Change per Cycle) is aborted because of an error or multiple start.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. MELSEC iQ-R Motion Module User's Manual (Application)
CancelAccepted	Cancel acceptance	BOOL	FALSE	This variable becomes TRUE when MCv_ChangeCycle (Current Value Change per Cycle) accepts the cancel.

Processing details

- Changes the current value per cycle of the calculation profile control FB to the set value at execution of control.
- Sets Instance ID (InstanceID) of the public variable of a calculation profile control FB (such as MC_CamIn) in InstanceID (InstanceID).
- Upon performing the current value per cycle change, the public variable of the calculation profile control FB (such as MC_CamIn) changes as shown below.

Public variable	Updated value	Remark
Current value per cycle (InputPerCycle)	Value set in Current value per cycle (Cycle)	—
Reference value (Reference)	Stroke value equivalent to Current value per cycle (Cycle)	Updates Reference value (Reference) in order to fix Output value (OutputData).
Output value (OutputData)	No value updated	—

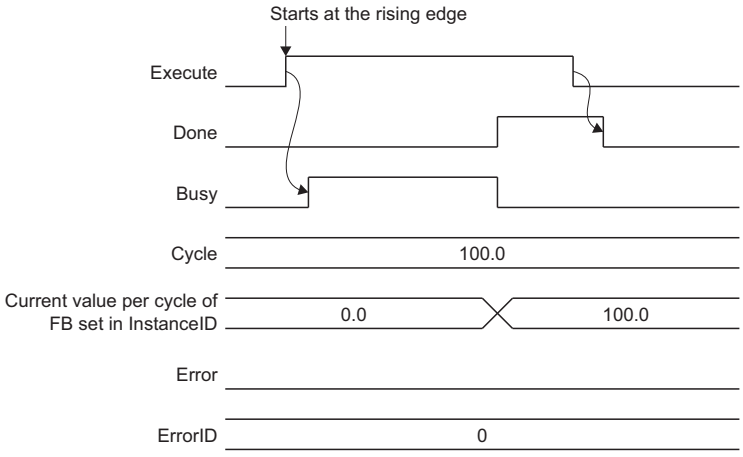
- To set the current value per cycle to be changed, set "Absolute position (FALSE)" or "Relative distance (TRUE)" in Relative selection (Relative).

Point

Reference value (Reference) is updated in order to fix Output value (OutputData) when performing current value per cycle change. This occurs even when Slave axis absolute coordinate (SlaveAbsolute) of MC_CamTableSelect (Cam Table Selection) is set to TRUE (Absolute coordinate).

■Timing chart

• When the operation completes normally



• When an error occurs

For the timing chart when an error occurs, refer to the following.

☞ Page 11 Basic operation of Execute command (Execute) type Motion control FBs

■Current value per cycle (Cycle)

Sets the current value per cycle to be changed.

Values settable for Current value per cycle (Cycle) differ depending on the setting of Relative selection (Relative). Set the variable within the setting range shown below. If the variable is set outside the range, "MCv_ChangeCycle Instruction Error (error code: 3497H (Cycle is out of range (detail code: 1)))" occurs.

Current value per cycle (Cycle)	Setting value of Relative selection (Relative)	Setting range of Current value per cycle (Cycle)
Cycle[0]	FALSE (Absolute position)	0.0 ≤ Setting value < Current value per cycle
	TRUE (Relative distance)	-(Cycle length) / 2 ≤ Setting value ≤ (Cycle length) / 2
Cycle[1], Cycle[2]	FALSE (Absolute position)	0.0
	TRUE (Relative distance)	

■Relative Selection (Relative)

The value of Current value per cycle (Cycle) is changed as shown below based on the setting of Relative selection (Relative).

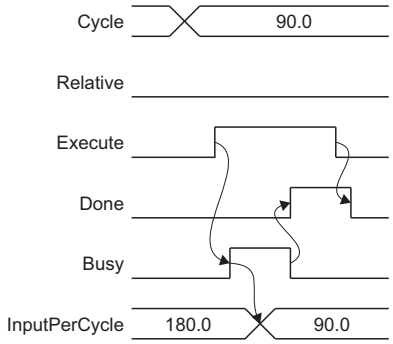
Setting value of Relative selection (Relative)	Description
FALSE (Absolute position)	Changes Current value per cycle (InputPerCycle)*1 to a value equal to Current value per cycle (Cycle) added to the presently set Current value per cycle (InputPerCycle)*1.
TRUE (Relative distance)	Current value per cycle (InputPerCycle)*1 is changed to the value of Current value per cycle (Cycle).

*1 Public variable of calculation profile control FBs (MC_CamIn, etc.)

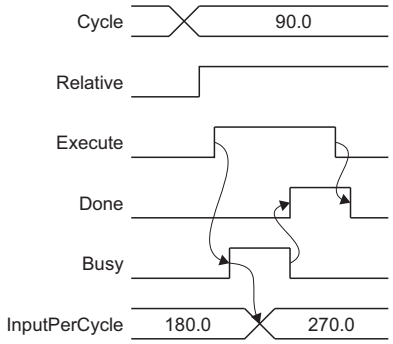
Ex.

When public variable Current value per cycle (InputPerCycle) is "180.0" at stop, and a current value per cycle change of "90.0" is set in Current value per cycle (Cycle)

• When Relative selection (Relative) is FALSE (Absolute position)



• When Relative selection (Relative) is TRUE (Relative distance)



The following table shows whether or not exceeding the current value per cycle is possible when the current value per cycle is changed.

○: Exceeding possible, ×: Exceeding not possible

Setting value of Relative selection (Relative)	Relationship between 1 cycle and amount moved by the current value per cycle	Exceeding Possible
FALSE (Absolute position)	1 cycle > movement amount of the current value per cycle	×
TRUE (Relative distance)	1 cycle > movement amount of the current value per cycle	○
	1 cycle ≤ movement amount of the current value per cycle	○*1

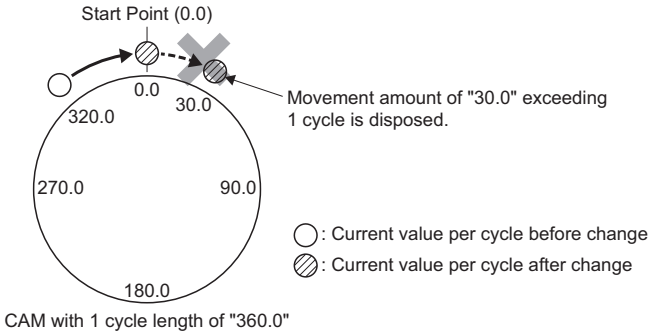
*1 The operation when exceeding occurs will differ based on the set values for Periodic (Periodic) set under MC_CamTableSelect (Cam Table Selection)

Setting value of Periodic (Periodic)	Details
FALSE (Non periodic)	Movement amount exceeding 1 cycle is disposed.
TRUE (Periodic)	Changes the current value per cycle when exceeding 1 cycle.

Ex.

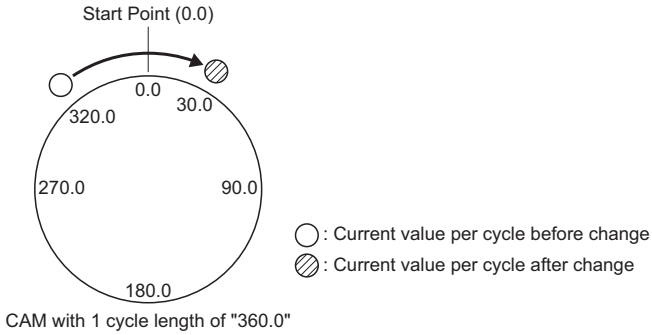
When the Current Value per Cycle (Cycle) changes from "320.0" to "30.0"

• When Periodic (Periodic) is FALSE (Non periodic).



CAM with 1 cycle length of "360.0"

• When Periodic (Periodic) is TRUE (Periodic).

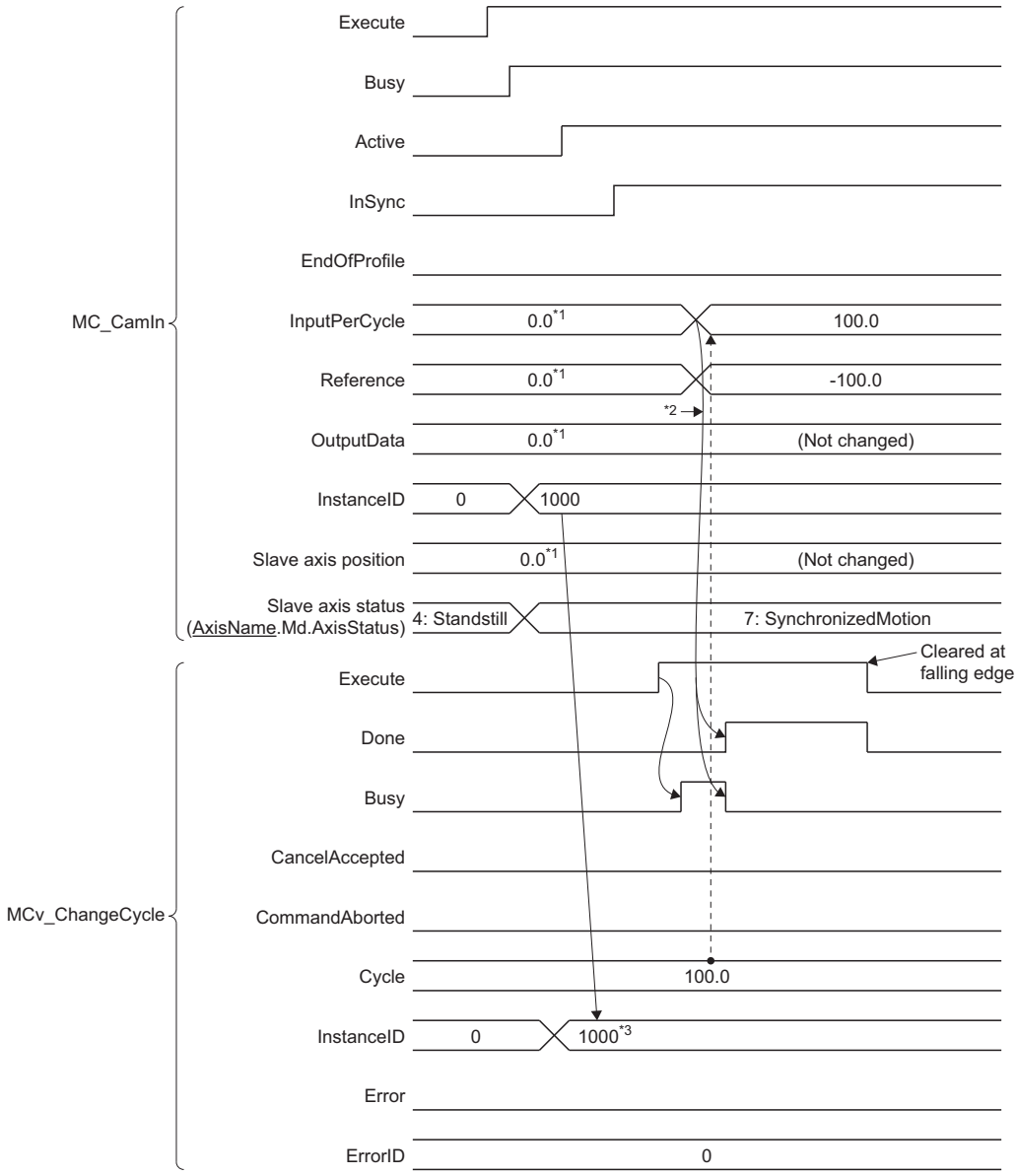


CAM with 1 cycle length of "360.0"

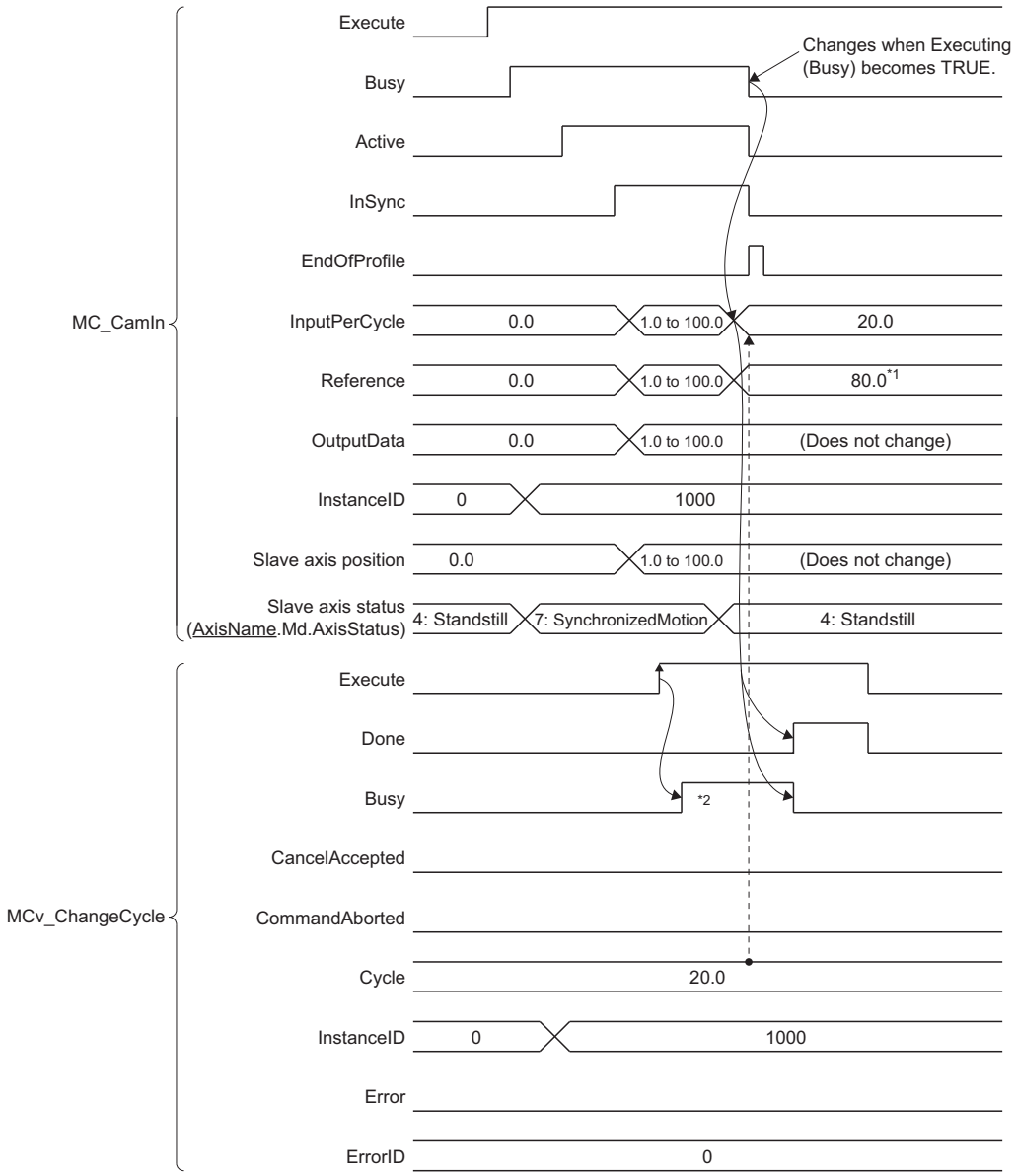
■Execution mode (ExecutionMode)

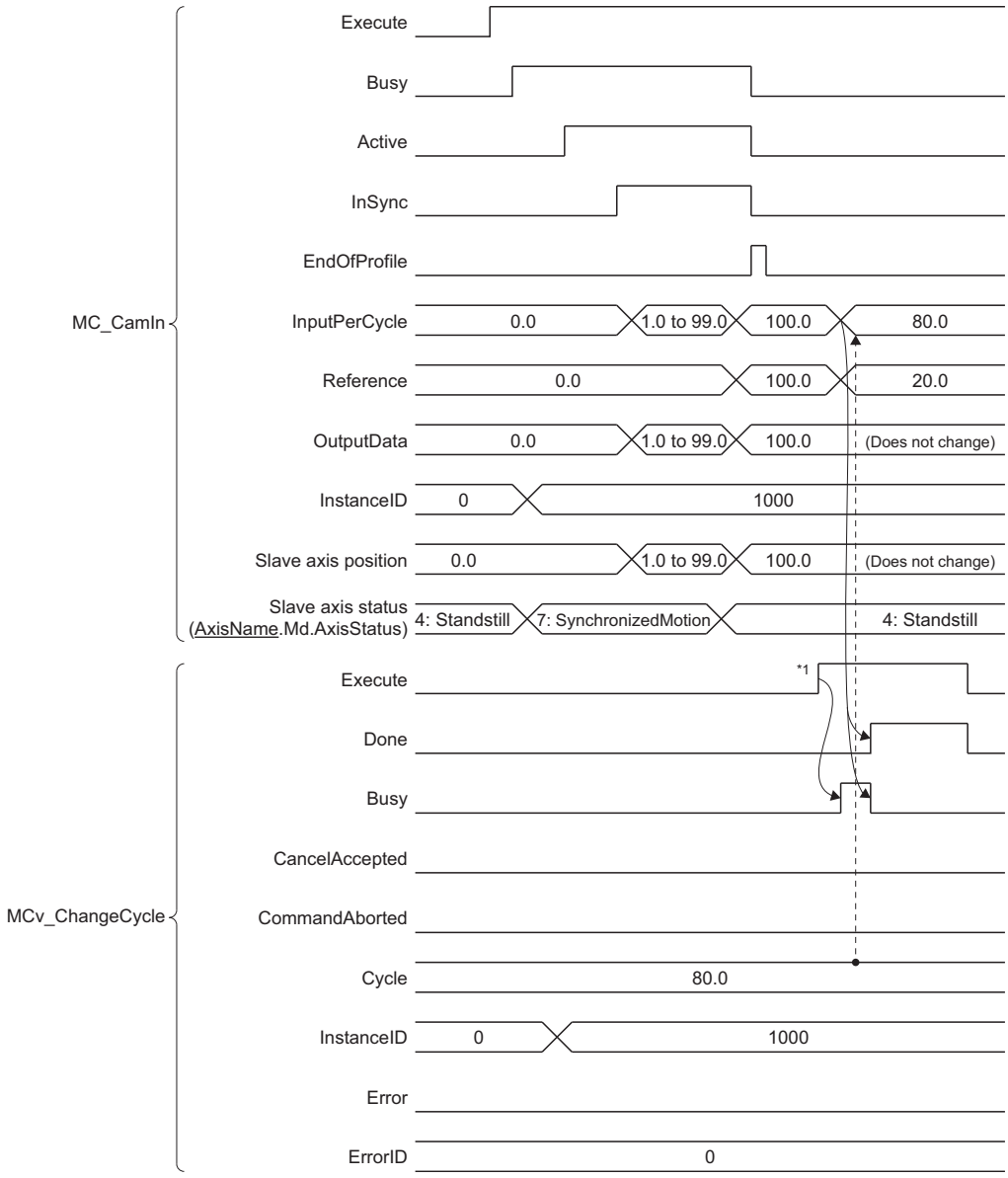
When MCv_ChangeCycle (Current Value Change per Cycle) is executed, the timing for the current value per cycle change differs depending on the setting of Execution mode (ExecutionMode). The timing for Execution mode (ExecutionMode) is shown below.

Setting value	Description
0: Execute immediately (mcImmediately)	Immediately executes using Start (Execute) rising edge detection regardless of whether Executing (Busy) of the calculation profile control FB (MC_CamIn, etc.) is TRUE. <Example> When current value per cycle change is executed while Executing (Busy) of MC_CamIn (Cam Operation Start) is TRUE and "100.0" is set in Current value per cycle (Cycle) (The current value for Master axis (Master) of MC_CamIn (Cam Operation Start) is considered as not being changed.)



*1: Does not change because the main axis does not move.
*2: At current value change per cycle completion, Completion (Done) becomes TRUE and Executing (Busy) becomes FALSE.
*3: The instance ID of the FB executing the current value change per cycle is set manually.

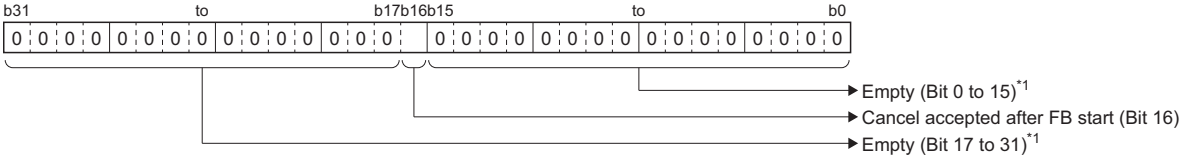
Setting value	Description
1: Execute at completion (mcQueued)	Executing (Busy) becomes TRUE at Start (Execute) rising edge detection. Executes after waiting for Executing (Busy) of the calculation profile control FB (MC_CamIn, etc.) to change from TRUE to FALSE. <Example> When current value per cycle change is executed with Current value per cycle (Cycle) set to "20.0"  *1: A refresh is generated at cycle completion (100.0) and by current value per cycle change (-20.0). *2: Executing (Busy) of MCv_ChangeCycle (Current Value Change per Cycle) waits at TRUE until Executing (Busy) of MC_CamIn (Cam Operation Start) becomes FALSE.

Setting value	Description
3: Execute speculatively (mcSpeculatively)	Executes when Executing (Busy) of the calculation profile control FB (MC_CamIn, etc.) is FALSE. <Example> When current value per cycle change is executed with Current value per cycle (Cycle) set to "80.0".  *1: Sets Start (Execute) of MCv_ChangeCycle (Current Value Change per Cycle) to TRUE while Executing (Busy) of MC_CamIn (Cam Operation Start) is FALSE.

■Options (Options)

Sets the function options used for MCv_ChangeCycle (Current Value Change per Cycle) by specifying the bit.

Contents that are set by specifying the bit are shown below.



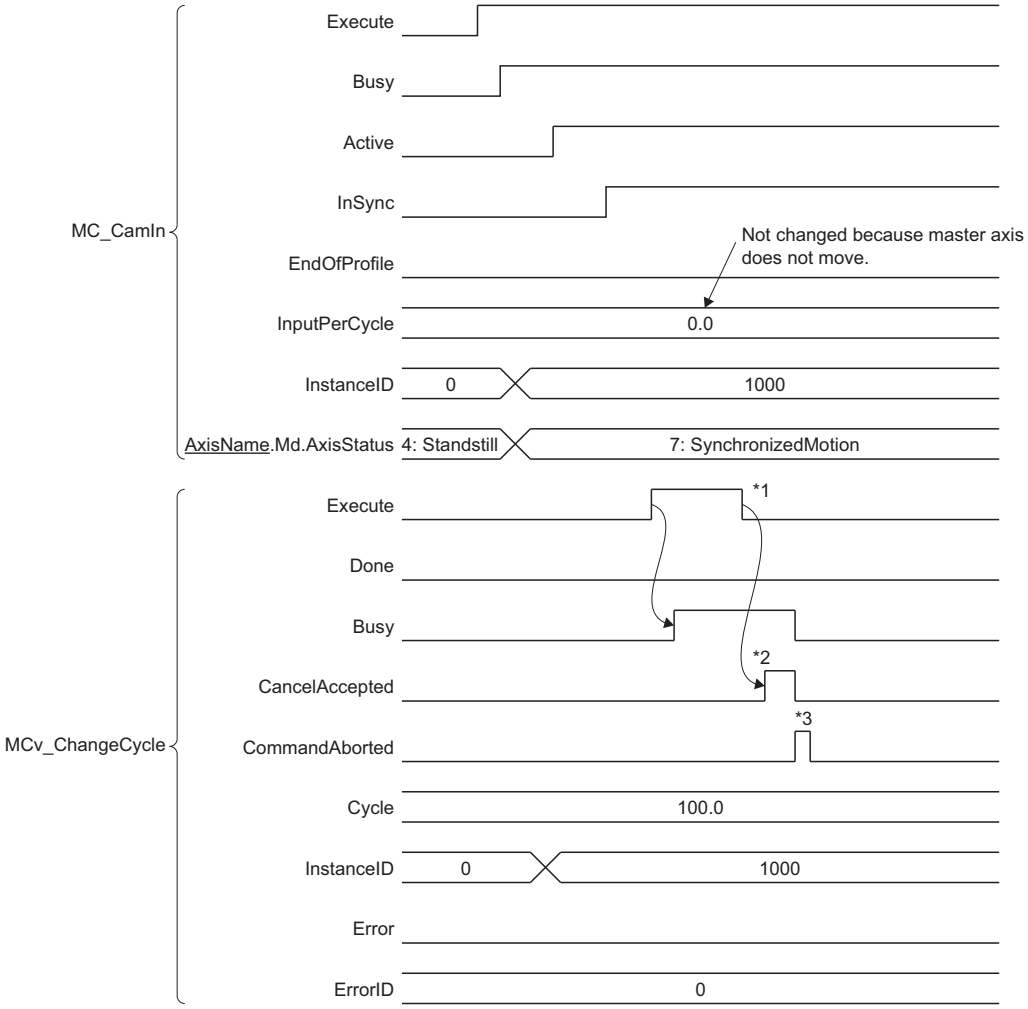
*1 Set empty areas to "0". When a value other than "0" is set, "Out of Options Range (error code: 1A4EH)" occurs.

Bit	Name	Description
16	Cancel accepted after FB start	Sets whether or not to allow a cancel after MCv_ChangeCycle (Current Value Change per Cycle) is executed. • 0: Do not allow • 1: Allow

- A current value per cycle change that is in standby status can be cancelled after FB execution by setting Cancel accepted after FB start.
  - The cancel is started at falling edge detection of Start (Execute).
  - The cancel is accepted only when Executing (Busy) is TRUE.
  - When the FB accepts the cancel, Cancel acceptance (CancelAccepted) becomes TRUE.
  - When the cancel is completed, Abortion of execution (CommandAborted) becomes TRUE.
  - When the cancel was performed, Current value per cycle (InputPerCycle) is not changed.

Ex.

After starting with Execution mode (ExecutionMode) set to "1: Execute at completion (mcQueued)", the cancel is performed while Executing (Busy) is waiting for the switch from TRUE to FALSE.



*1 When Cancel accepted after FB start (Options (Options): Bit 16) is set to "1: Allow". If "0: Do not allow" is set, cancelling is disabled.

*2 CancelAccepted becomes TRUE when the cancel is accepted, but becomes FALSE when the cancel completes or the current value per cycle change completes.

*3 CommandAborted becomes TRUE when the cancel is completed, but becomes FALSE one cycle later because Start (Execute) is FALSE.

Program example

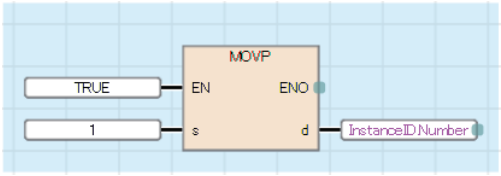
In the following program example, the current value change per cycle command (bChangeCycle) is set to TRUE and Instance ID (InstanceID) changes the current value per cycle of "1" to "90.0".

■Labels used

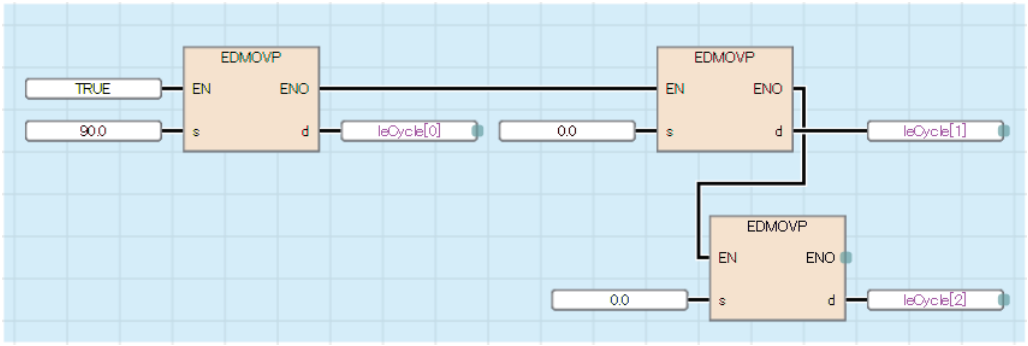
Label name	Data type	Comment
MCv_ChangeCycle_1	MCv_ChangeCycle	Current Value Change per Cycle FB
bChangeCycle	Bit	Current value change per cycle command
InstanceID	INSTANCE_ID	Instance ID
leCycle	Double-precision real number(0..2)	Current value per cycle
bDone	Bit	Execution completion
bBusy	Bit	Executing
bCommandAborted	Bit	Abortion of execution
bError	Bit	Error
uwErrorID	Word [unsigned]/bit string [16-bit]	Error code
bCancelAccepted	Bit	Cancel acceptance

■FBD/LD program (CPU module side)

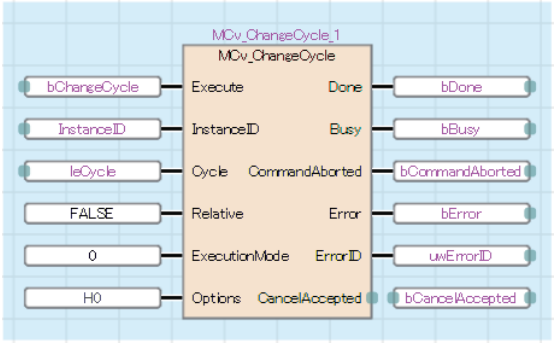
- Calculation profile setting



- Data setting for Current Value Change per Cycle



- Current Value Change per Cycle

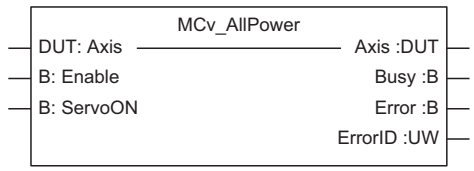
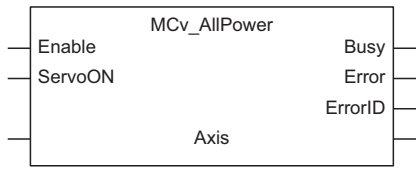


■ST program (Motion module side)

```
//-----Calculation profile setting-----
InstanceID.Number:= 1;

//-----Data setting for Current Value Change per Cycle-----
leCycle[0]:= 90.0;
leCycle[1]:= 0.0;
leCycle[2]:= 0.0;

//-----Current Value Change per Cycle-----
MCv_ChangeCycle_1(
   Execute:= bChangeCycle ,
   InstanceID:= InstanceID ,
   Cycle:= leCycle ,
   Relative:= FALSE ,
   ExecutionMode:= MC_EXECUTION_MODE__mcImmediately ,
   Options:= H00000000 ,
   Done=> bDone ,
   Busy=> bBusy ,
   CommandAborted=> bCommandAborted ,
   Error=> bError ,
   ErrorID=> uwErrorID ,
   CancelAccepted=> bCancelAccepted
);
```

MCv_AllPower

Switches every axis to the operation possible status.

Ladder	FBD/LD	ST
		MCv_AllPower(Axis:= ?AXIS_REF? , Enable:= ?BOOL? , ServoON:= ?BOOL? , Busy=> ?BOOL? , Error=> ?BOOL? , ErrorID=> ?WORD?);

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
All Axes Operation Possible	10	4	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
Axis	Axis information	AXIS_REF	At start	—	Omission possible	When using MCv_Allpower (All Axes Operation Available) on the Motion module side, this variable can be omitted as this setting is ignored. When using MCv_Allpower (All Axes Operation Available) on the CPU module side, this variable sets IO No. (Master.StartIO). When used on the CPU module side, Axis No. (AxisNo) is ignored. ☞ Page 23 AxisName.AxisRef. (Axis information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Enable	Enable	BOOL	Always	TRUE, FALSE	FALSE	When this variable is set to TRUE, axis control is enabled and the axis status switches to operation available status. When this variable is set to FALSE, axis control is disabled and operation available status is cancelled for the axis.
ServoON	Servo ON request	BOOL	Always	TRUE, FALSE	FALSE	When this variable is set to TRUE, requests the servo ON.

■Output variables

Output variable	Name	Data type	Default value	Description
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MCv_AllPower (All Axes Operation Available) is executed.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. MELSEC iQ-R Motion Module User's Manual (Application)

Processing details

- Initializes the information of every axis and switches the axis status to the operation available status.
- When the Enable (Enable) and Servo ON request (ServoON) inputs are set to TRUE, every axis is switched to the operation available status.
- Starting the processing causes Executing (Busy) to become TRUE.
- When used on the Motion module side, ignores the setting of Axis information (Axis). When used on the CPU module side, sets the I/O No. in I/O No. (StartIO) of Axis information (Axis). For specifying the I/O No., refer to the following.
  ☞ Page 17 Specification of I/O No.
- When an error occurs in MCv_AllPower (All Axes Operation Available), Error (Error) becomes TRUE and the error code is stored in Error code (ErrorID). For details of error codes, refer to the following.
  MELSEC iQ-R Motion Module User's Manual (Application)
- The servo ON/OFF status and the driver status of all real axes can be switched as follows by inputting Enable (Enable) and Servo ON request (ServoON).

Input variables		Servo ON/OFF status	Driver status (AxisName.Md.Driver_State)
Enable (Enable)	Servo ON request (ServoON)		
TRUE	TRUE	Servo ON	6: Operation Enable
	FALSE	Servo OFF	5: Switched On
FALSE	TRUE	Servo OFF	3: Switch On Disabled
	FALSE	Servo OFF	3: Switch On Disabled

- If the real axis is rotated by external force while in servo OFF status, follow up processing is performed.
- The servo ON/OFF control can be operated regardless of the control mode. The control mode used while in servo OFF status depends on the specification of the driver.
- Since MCv_AllPower (All Axes Operation Available) is sent to the driver while a drive unit error is occurring, there is no need to turn Enable (Enable) and Servo ON request (ServoON) from FALSE to TRUE again.

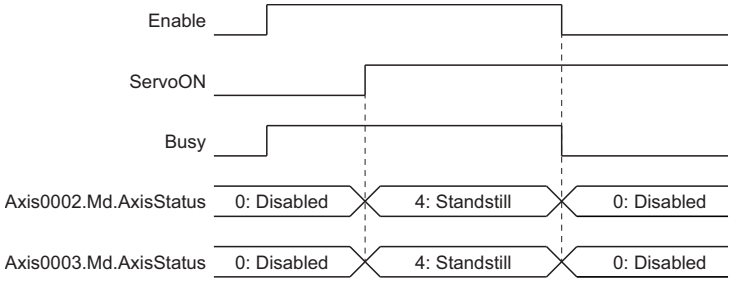
Point

Use MCv_AllPower (All Axes Operation Available) and MC_Power (Operation Available) together if executing servo OFF individually.
When MCv_AllPower (All Axes Operation Available) and MC_Power (Operation Available) are used together, the MC_Power (Operation Available) command is given priority.

■Timing chart

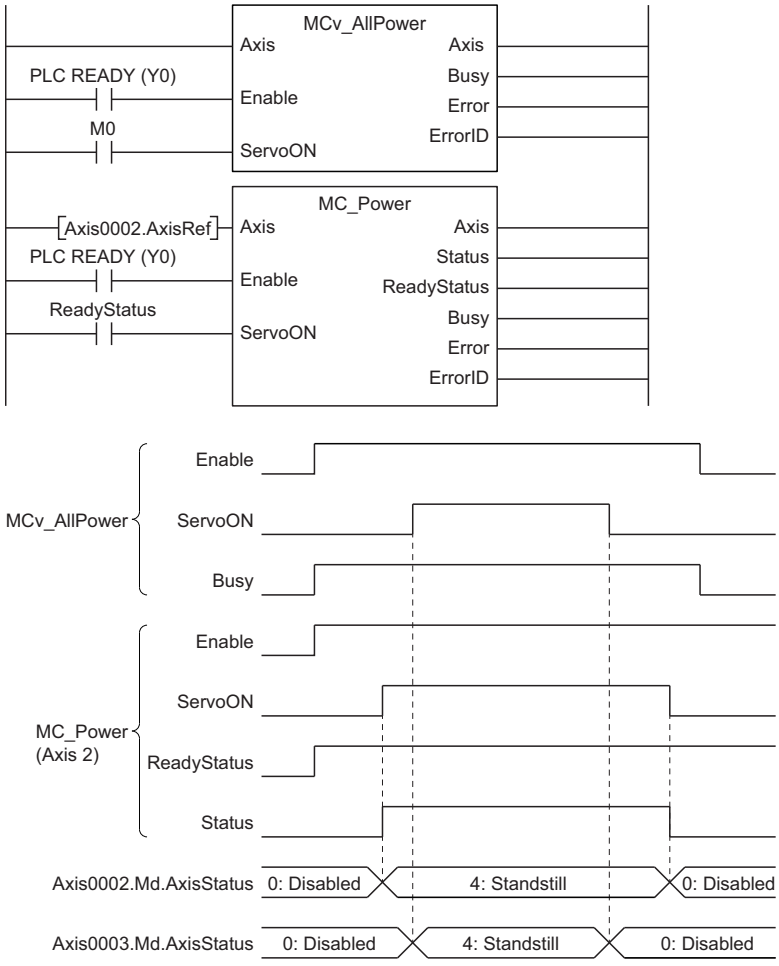
Ex.

When Axis 2 (Axis0002) and Axis 3 (Axis0003) are set as real drive axes



Ex.

When used together with MC_Power for Axis 2 (Axis0002)



Precautions

- For axes setting MC_Power (Operation Available), the MC_Power (Operation Available) command is given priority for switching to operable status.

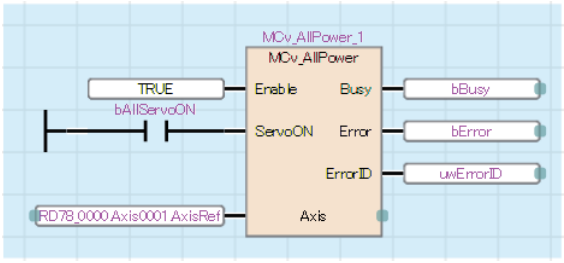
Program example

In the following program example, all axes servo ON/OFF (bAllServoON) is set to TRUE and all axes are set to operation available status.

■Labels used

Label name	Data type	Comment
MCv_AllPower_1	MCv_AllPower	All axes servo ON FB
bAllServoON	Bit	All axes servo ON/OFF
bBusy	Bit	Executing
bError	Bit	Error
uwErrorID	Word [unsigned]/bit string [16-bit]	Error code

■FBD/LD program

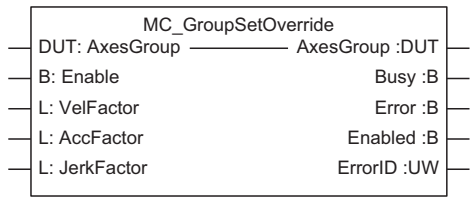
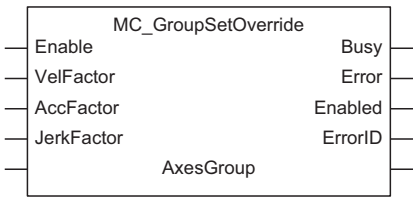


■ST program (Motion module side)

```
MCv_AllPower_1(
   Enable:= TRUE ,
   ServoON:= bAllServoON ,
   Busy=> bBusy ,
   Error=> bError ,
   ErrorID=> uwErrorID
);
```

MC_GroupSetOverride

Changes the target velocity, the target acceleration, and the target deceleration of the specified axes group.

Ladder	FBD/LD	ST
		MC_GroupSetOverride(AxesGroup:= ?AXES_GROUP_REF? , Enable:= ?BOOL? , VelFactor:= ?LREAL? , AccFactor:= ?LREAL? , JerkFactor:= ?LREAL? , Enabled=> ?BOOL? , Busy=> ?BOOL? , Error=> ?BOOL? , ErrorID=> ?WORD?);

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Axes Group Override Value Setting	40	6	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
AxesGroup	Axes group information	AXES_GROUP_REF	At start	—	Omission not possible	This variable sets the axes group. For the variables used (AxesGroupName.AxesGroupRef.), refer to the following. ☞ Page 46 AxesGroupName.AxesGroupRef. (Axes group information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Enable	Enable	BOOL	Always	TRUE, FALSE	FALSE	When this variable is TRUE, executes MC_GroupSetOverride (Axes Group Override Value Setting).
VelFactor	Velocity override factor	LREAL	Always	0.00 to 10.00	0.00	This variable sets the velocity override factor. When Enable (Enable) is TRUE, values are always imported.
AccFactor	Acceleration override factor	LREAL	Always	0.00, 0.01 to 10.00	0.00	This variable sets the acceleration override factor. When Enable (Enable) is TRUE, values are always imported. When "0.00" is set, the acceleration override factor is not changed and control is performed using the previous value.
JerkFactor	Jerk override factor	LREAL	Always	0.0	0.0	Set this variable to "0.0". *: If a value other than "0.0" is set, "Out of Jerk Override Coefficient (JerkFactor) Range (error code: 349EH)" occurs.

■Output variables

Output variable	Name	Data type	Default value	Description
Enabled	Enabled	BOOL	FALSE	This variable becomes TRUE when the set override value is correct.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MC_GroupSetOverride (Axes Group Override Value Setting) is executed.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.

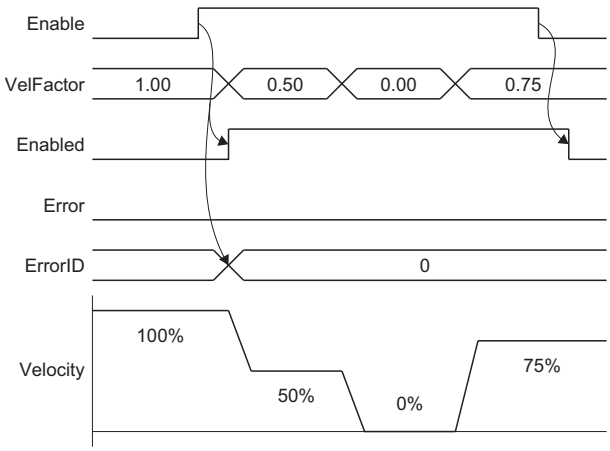
Output variable	Name	Data type	Default value	Description
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. 📖MELSEC iQ-R Motion Module User's Manual (Application)

Processing details

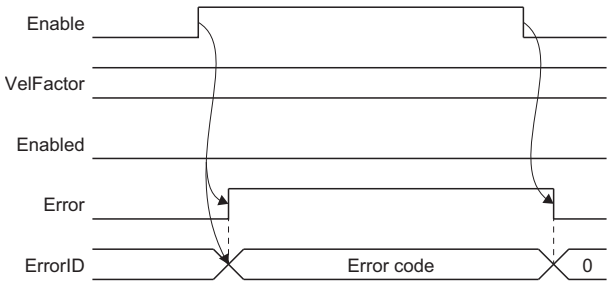
- Changes the target velocity, the target acceleration, and the target deceleration of the specified axes group.
- The value after the change is equal to the current value during operation for the target velocity, target acceleration, and target deceleration multiplied by the override factor.
- MC_GroupSetOverride (Axes Group Override Value Setting) is executed when Enable (Enable) becomes TRUE. Enabled (Enabled) becomes TRUE while the override factor is valid.
- If the override factor value is changed while Enable (Enable) is TRUE, the new override factor is reflected.
- When an error occurs in MC_GroupSetOverride (Axes Group Override Value Setting), Error (Error) becomes TRUE and the error code is stored in Error code (ErrorID). For details of error codes, refer to the following.
  📖MELSEC iQ-R Motion Module User's Manual (Application)
- When Velocity override factor (VelFactor) is set to "0.00", the axis is stopped without Axis status (AxisName.Md.AxisStatus) changing to "4: Standby (Standstill)".
- When Acceleration override factor (AccFactor) is set to "0.00", the acceleration override factor is not changed, and the previous acceleration override factor is maintained.

■Timing chart

- When the operation completes normally

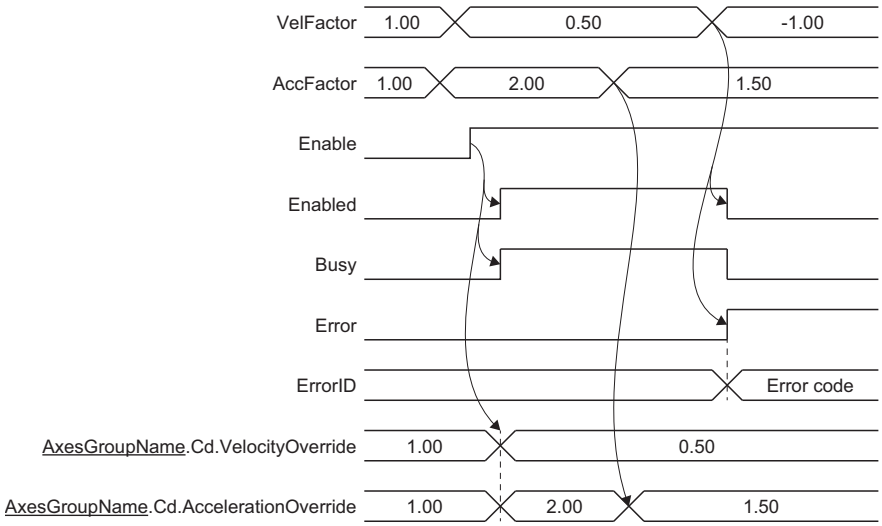


- When an error occurs



■Operation overview

- In MC_GroupSetOverride (Axes Group Override Value Setting), Velocity override factor (AxesGroupName.Cd.VelocityOverride) and Acceleration override factor (AxesGroupName.Cd.AccelerationOverride) are changed.



- If the override factor is set to a value outside the range, an error will occur in MC_GroupSetOverride (Axes Group Override Value Setting) and subsequent values will not be imported. To import values again, restart Enable (Enable).

Precautions

- Do not assign two or more MC_GroupSetOverride (Axes Group Override Value Setting) FBs in the same axes group. Operation is not guaranteed when two or more of this FB are assigned in the same axes group.
- For axes group control, only the override factor set with the axes group control data has an effect.
- While executing MC_GroupSetOverride (Axes Group Override Value Setting), do not perform direct operation of Velocity override factor (AxesGroupName.Cd.VelocityOverride) and Acceleration override factor (AxesGroupName.Cd.AccelerationOverride).
- When the speed after Speed override is out of range, refer to "Velocity Range" in the following manual.
  MELSEC iQ-R Motion Module User's Manual (Application)
- When the acceleration, deceleration, acceleration time, and deceleration time after acceleration override is out of range, refer to "Acceleration/deceleration processing function" in the following manual.
  MELSEC iQ-R Motion Module User's Manual (Application)

Program example

In the following program example, axes group override value change command (bGroupSetOverride) is set to TRUE and the target velocity, target acceleration, and target deceleration of Axes group 1 (AxesGroup001) are changed to values multiplied by a velocity override factor of "1.0" and an acceleration override factor of "2.0".

■Axes group

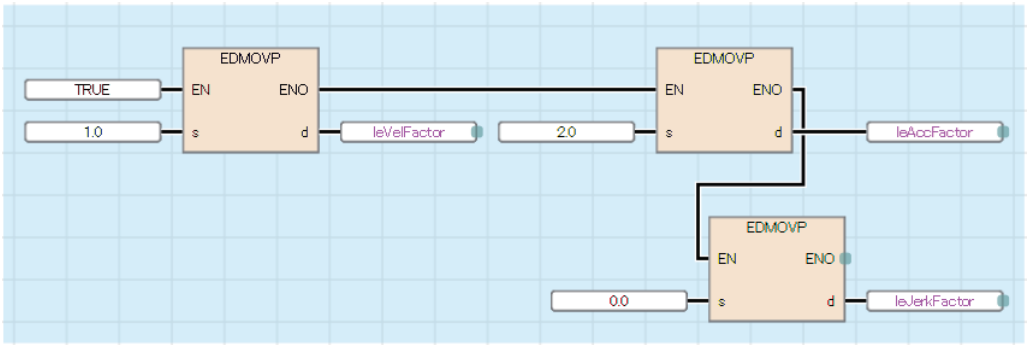
Axes group No.	Label name	Data type	Comment
1	AxesGroup001	AXES_GROUP_REF	Axes group 1

■Labels used

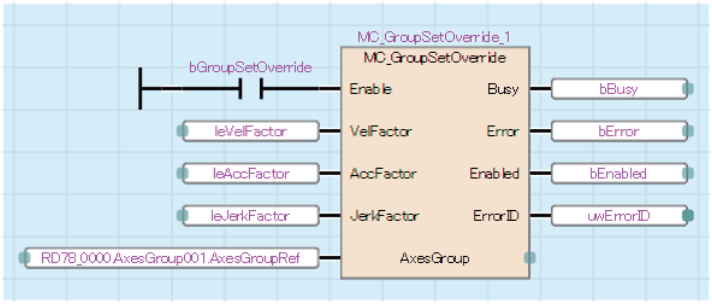
Label name	Data type	Comment
MC_GroupSetOverride_1	MC_GroupSetOverride	Axes Group Override Value Setting FB
bGroupSetOverride	Bit	Axes group override value change command
leVelFactor	Double-precision real number	Velocity override factor
leAccFactor	Double-precision real number	Acceleration override factor
leJerkFactor	Double-precision real number	Jerk override factor
bEnabled	Bit	Enabled
bBusy	Bit	Executing
bError	Bit	Error
uwErrorID	Word [unsigned]/bit string [16-bit]	Error code

■FBD/LD program (CPU module side)

• Data setting for Axes Group Override Value Setting



• Axes group override value change



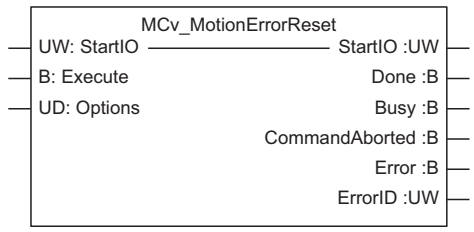
■ST program (Motion module side)

```
//-----Data setting for Axes Group Override Value Setting-----
leVelFactor:= 1.00;
leAccFactor:= 2.00;
leJerkFactor:= 0.0;

//-----Axes group override value change-----
MC_GroupSetOverride_1(
   AxesGroup:= AxesGroup001.AxesGroupRef ,
   Enable:= bGroupSetOverride ,
   VelFactor:= leVelFactor ,
   AccFactor:= leAccFactor ,
   JerkFactor:= leJerkFactor ,
   Enabled=> bEnabled ,
   Busy=> bBusy ,
   Error=> bError ,
   ErrorID=> uwErrorID
);
```

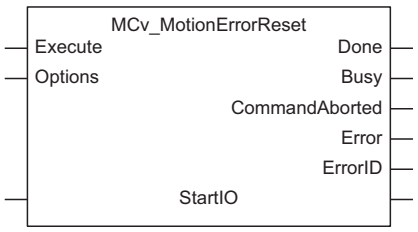
MCv_MotionErrorReset

Resets all errors and warnings of the Motion system.

Ladder



FBD/LD



ST

```
MCv_MotionErrorReset(
  StartIO:= ?WORD? ,
  Execute:= ?BOOL? ,
  Options:= ?DWORD? ,
  Done=> ?BOOL? ,
  Busy=> ?BOOL? ,
  CommandAborted=> ?BOOL? ,
  Error=> ?BOOL? ,
  ErrorID=> ?WORD?
);
```

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Motion Error Reset	8	6	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
StartIO	I/O No.	WORD(HEX)	At start	000H to 0FEH	Omission possible	This variable sets the start I/O No. (First 3 digits when displayed as 4 hexadecimal digits). When using MCv_MotionErrorReset (Motion Error Reset) on the Motion module side, this variable can be omitted as the setting is ignored. When using MCv_MotionErrorReset (Motion Error Reset) on the CPU module side, this variable cannot be omitted.

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Execute command	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MCv_MotionErrorReset (Motion Error Reset).
Options	Options	DWORD(HEX)	At start	00000000H	00000000H	Set this variable to "00000000H". *: When a value other than "00000000H" is set, "Out of Options Range (error code: 1A4EH)" occurs.

■Output variables

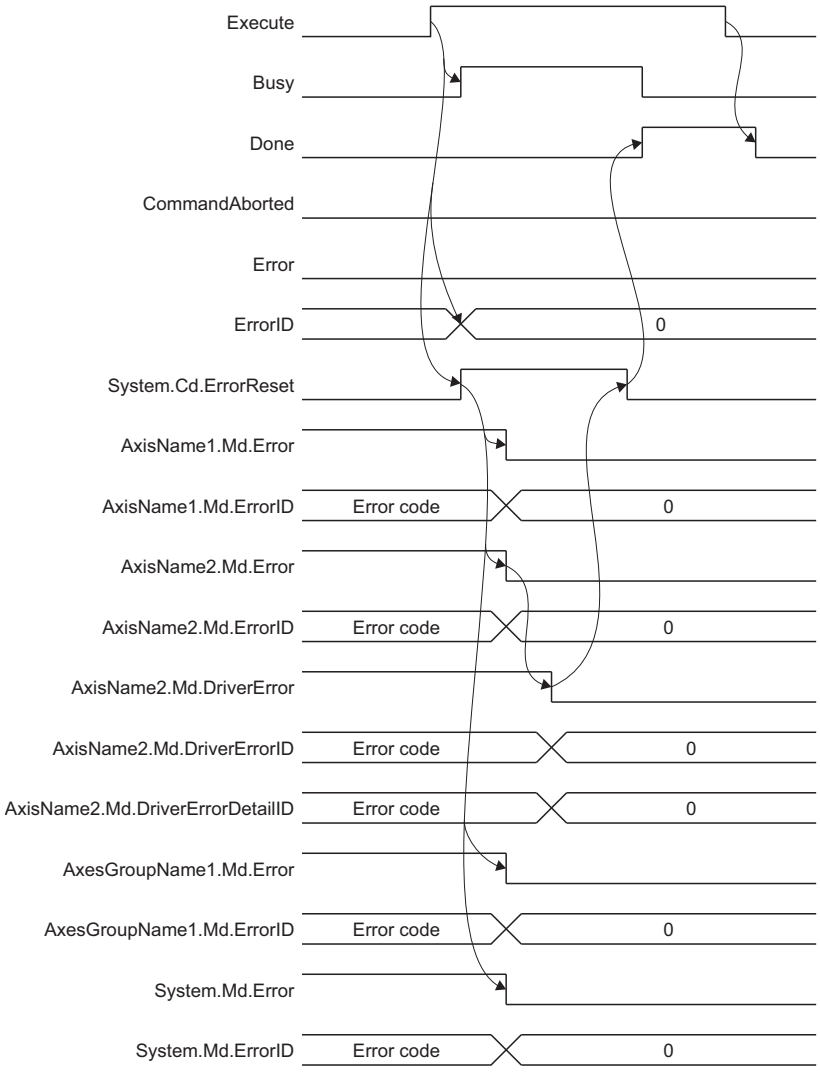
Output variable	Name	Data type	Default value	Description
Done	Execution completion	BOOL	FALSE	This variable becomes TRUE when the reset is completed.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MCv_MotionErrorReset (Motion Error Reset) is executed.
CommandAborted	Abortion of execution	BOOL	FALSE	This variable becomes TRUE when MCv_MotionErrorReset (Motion Error Reset) has been aborted because of timeout. This variable becomes FALSE when Execute command (Execute) becomes FALSE.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. MELSEC iQ-R Motion Module User's Manual (Application)

Processing details

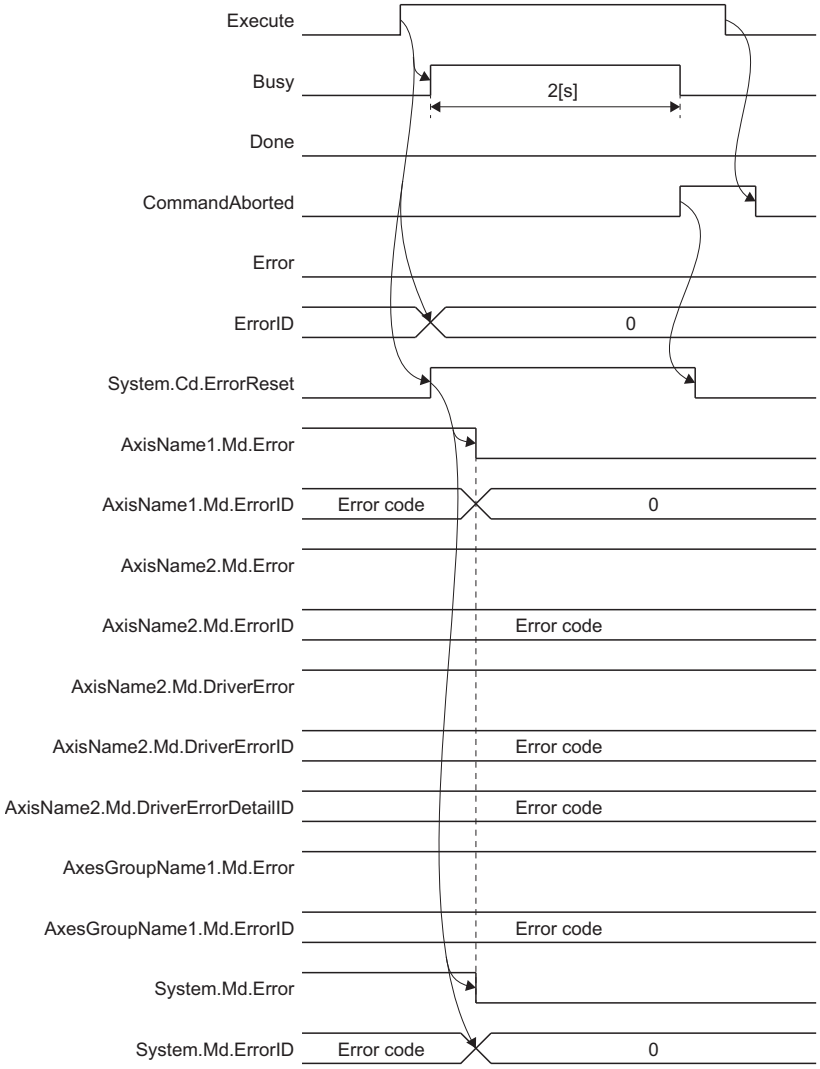
- Executes MCv_MotionErrorReset (Motion Error Reset) when Execution command (Execute) is set to TRUE, following which Executing (Busy) becomes TRUE when the processing starts and changes System error reset (System.Cd.Error.Reset) to TRUE.
- Once errors and warnings have been cleared for the Motion system, Executing (Busy) becomes FALSE and Execution completion (Done) becomes TRUE.
- Setting Execution command (Execute) to TRUE while the causes of the errors and warnings remain does not clear the errors and warnings. In addition, if the errors are not cleared within two seconds after this FB is executed, Abortion of execution (CommandAborted) becomes TRUE and changes System error reset (System.Cd.ErrorReset) to FALSE. Temporarily set Execution command (Execute) to FALSE, remove the causes of the errors and warnings, then set Execution command (Execute) to TRUE again.

■Timing chart

- When the operation completes normally



- At timeout



Precautions

- Do not directly operate System error reset (System.Cd.ErrorReset) while MCv_MotionErrorReset (Motion Error Reset) is being executed.
- To perform the error reset from the CPU module, use MCv_MotionErrorReset (Motion Error Reset) without operating System error reset (System.Cd.ErrorReset).

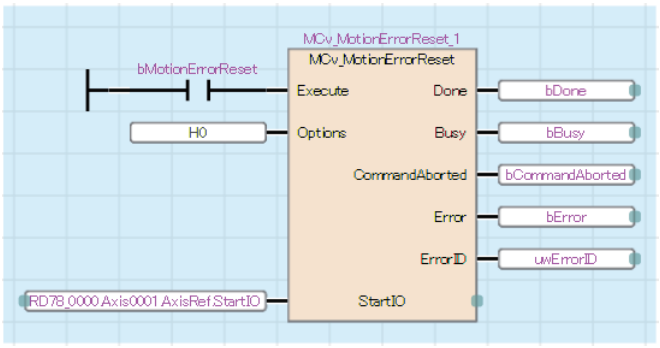
Program example

In the following program example, error reset (bMotionErrorReset) is set to TRUE and all errors and warnings of the Motion system are reset.

■Labels used

Label name	Data type	Comment
MCv_MotionErrorReset_1	MCv_MotionErrorReset	Motion Error Reset FB
bMotionErrorReset	Bit	Error reset
bDone	Bit	Execution completion
bBusy	Bit	Executing
bCommandAborted	Bit	Abortion of execution
bError	Bit	Error
uwErrorID	Word [unsigned]/bit string [16-bit]	Error code

■FBD/LD program (CPU module side)



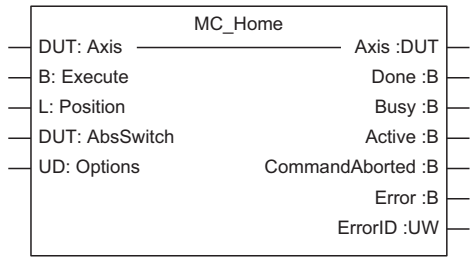
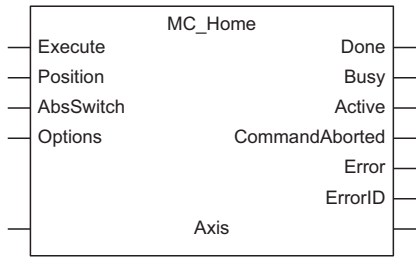
■ST program (Motion module side)

```
MCv_MotionErrorReset_1(
   Execute:= bMotionErrorReset ,
   Options:= H00000000 ,
   Done=> bDone ,
   Busy=> bBusy ,
   CommandAborted=> bCommandAborted ,
   Error=> bError ,
   ErrorID=> uwErrorID
);
```

3.2 Operation FBs

MC_Home

Executes homing for the specified axis.

Ladder	FBD/LD	ST
		MC_Home(Axis:= ?AXIS_REF? , Execute:= ?BOOL? , Position:= ?LREAL? , AbsSwitch:= ?MC_INPUT_REF? , Options:= ?DWORD? , Done=> ?BOOL? , Busy=> ?BOOL? , Active=> ?BOOL? , CommandAborted=> ?BOOL? , Error=> ?BOOL? , ErrorID=> ?WORD?);

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
OPR	188	8	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
Axis	Axis information	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)

■Input variables

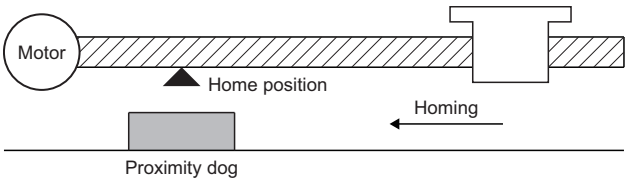
Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Execute command	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MC_Home (OPR).
Position	Target position	LREAL	At start	-10000000000.0 to 10000000000.0	0.0	This variable sets the home position address. Set the address within the following range. • -10000000000.0 ≤ Setting value <10000000000.0 *: When the ring counter is enabled, the ring counter range is used.
AbsSwitch	Home position switch	MC_INPUT_REF	At start	—	—	This variable sets the proximity dog signal that is transmitted to the slave device in the driver homing method. For details, refer to the following. ☞ Page 186 Home position switch (AbsSwitch)
Options	Options	DWORD(HEX)	At start	00000000H to 00010000H	00000000H	This variable sets the function options for MC_Home (OPR) by specifying the bit. For details, refer to the following. ☞ Page 187 Options (Options)

■Output variables

Output variable	Name	Data type	Default value	Description
Done	Execution completion	BOOL	FALSE	This variable becomes TRUE when the homing is completed.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MC_Home (OPR) is executed.
Active	Controlling	BOOL	FALSE	This variable becomes TRUE while MC_Home (OPR) is controlling the axis.
CommandAborted	Abortion of execution	BOOL	FALSE	This variable becomes TRUE when execution of MC_Home (OPR) is aborted.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. MELSEC iQ-R Motion Module User's Manual (Application)

Processing details

- In homing, a machine home position is established. The address information stored in the Motion system and driver is not used during this process. The position mechanically established after the homing becomes the start point, or "home position", for positioning control.



- For the homing method, "Driver homing method" and "Data set homing method" are available.
  The homing method used at homing start is determined based on the following conditions.

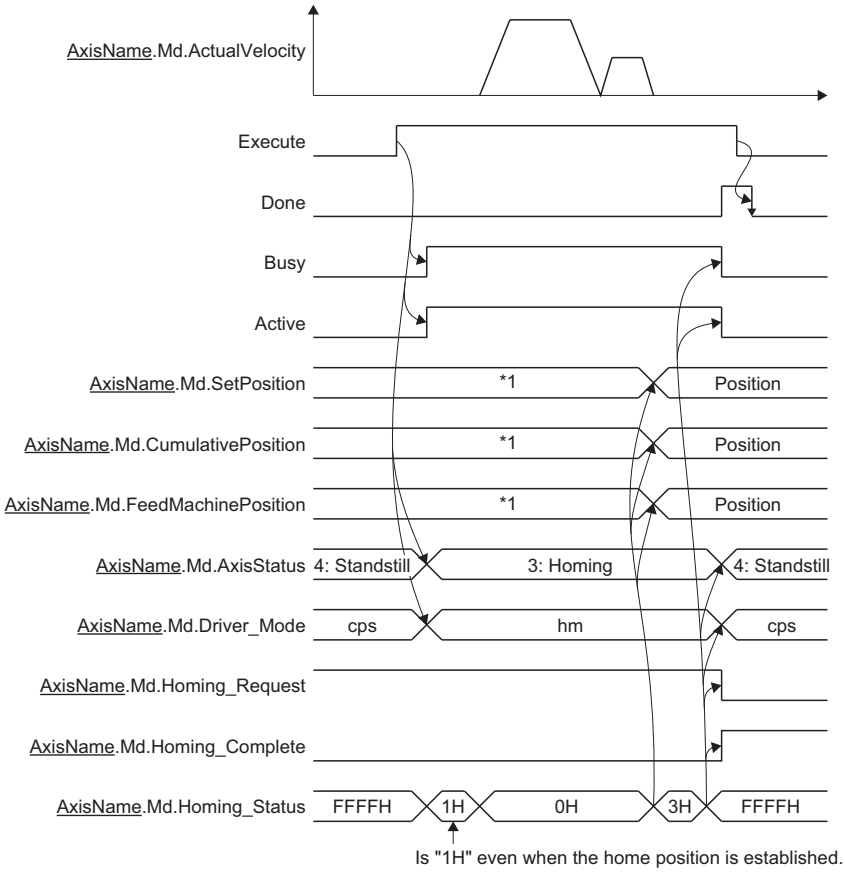
Homing method	Condition for homing method
Driver homing method	Driver homing method is used when all of the following conditions are satisfied. • The axis type is real drive axis. • The driver supports Homing mode. • "Home offset (607CH)" is set to a slave object.
Data set homing method	Data set homing method is used when the conditions listed above are not satisfied.

■Driver homing method

The driver is switched to the Homing mode, and the homing is executed based on the positioning pattern set on the driver side.

Change the homing data of the driver with MC_WriteParameter (Parameter Write) to change the homing method or each type of parameter. Homing operation and settable parameters vary depending on the specification of the driver. For details of the driver, refer to the driver manual.

- Timing chart
  - Normal operation



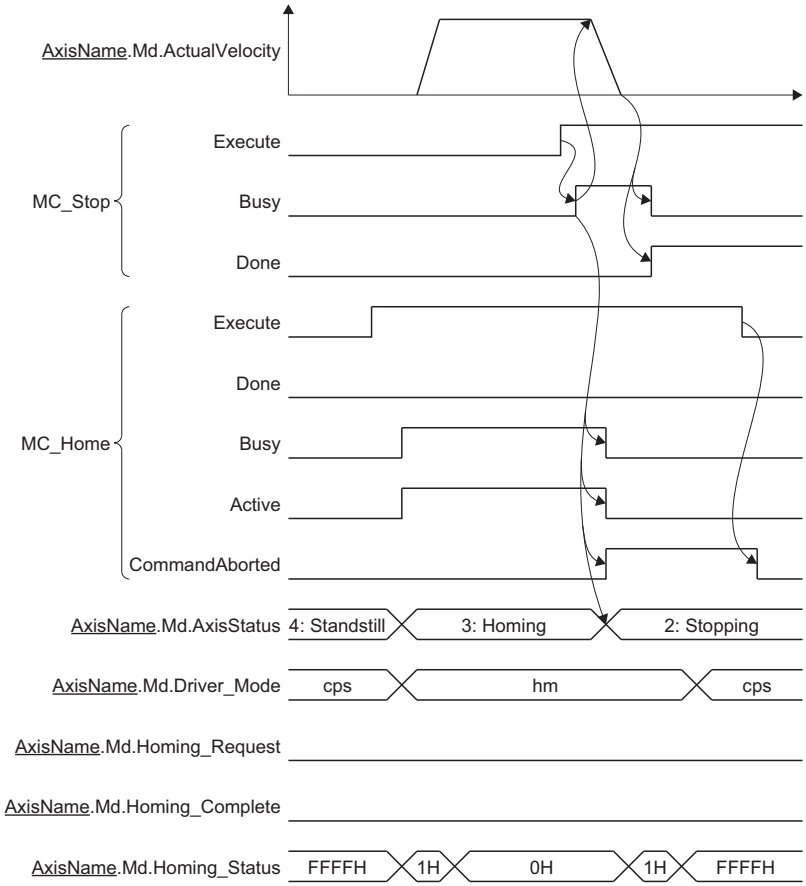
*1: When Follow-up enabled/disabled selection (Options (Options): Bit 16) is set to "0: Follow-up disabled", it will not update during homing. When set to "1: Follow-up enabled", it will update with the current position changes during homing.

  - When an error occurs

    For the timing chart when an error occurs, refer to the following.

    ☞ Page 11 Basic operation of Execute command (Execute) type Motion control FBs

• When MC_Stop (Forced Stop) is executed

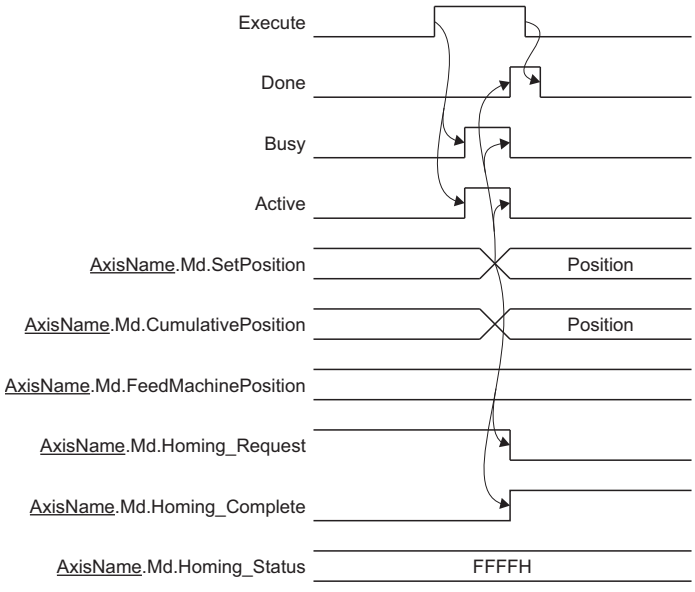


- When MC_Stop (Forced Stop) is set to TRUE when homing, the "HALT" signal is sent to the driver. If using a driver that does not support HALT, MC_Stop (Forced Stop) will not stop the driver, so use the forced stop instead.
- Stop processing at homing depends on the driver specifications. Therefore, Deceleration (Deceleration) and Jerk (Jerk) of MC_Stop (Forced Stop) are ignored.

■Data set homing method

Homing is executed for a virtual axis or a real axis that does not have the home position information in the slave device side. The homing is completed in the Motion system, and external signals, etc. are not used. The Target position (Position) at homing is registered in the Motion system as the home position, and Command Current Position (AxisName.Md.SetPosition) and Cumulative Current Position (AxisName.Md.CumulativePosition) are rewritten to Target position (Position).

• Timing chart



■Homing request

- When it is necessary to execute homing, the Motion system changes Home position return request (AxisName.Md.Homing_Request) to TRUE. Home position return request (AxisName.Md.Homing_Request) becomes TRUE in the following cases.
  - When the system's power supply was turned ON, or when a reset is performed.*1
  - When the axis type is changed.
  - When homing is started. (The homing request flag will not become FALSE unless homing is completed normally.)
  - When axis driver unit conversion (numerator/denominator) is changed.
  - When current position restoration is executed.
  - When the absolute position data in the Motion system is erased because of a memory error or other cause.
  - When the driver's power supply is turned ON.*1*2
  - When "Absolute position erased" is detected on the driver side.*2
  - When "Polarity (Obj. 607EH) (b7: position polarity)" of a driver object is changed.*2
  - When the electronic gear is changed on the driver side.*2
  - When a driver or motor encoder change is detected.
  - When using the emulate function, if the driver was changed to a driver different from the one that was connected at the previous establishment of the home position.
  - When it was detected that the absolute position in the scale measurement encoder is erased.*3
  - When the encoder resolution of the scale measurement encoder is changed.*3
  - When the upper limit value/lower limit value of the encoder ring counter is changed.*4

*1 For an absolute position system, does not become TRUE if homing is complete at reset or system/drive power OFF.
*2 Real drive axis only.
*3 Real encoder axis (via drive module) only.
*4 Virtual encoder axis only.

Point

When Home Position Return Request (AxisName.Md.Homing_Request) becomes TRUE, it is recorded in the event history.
The details can be confirmed in the event history.

- Execute a homing request clear in systems that do not require homing. Changing Home position return request clear (AxisName.Cd.Homing_ClearRequest) to TRUE causes Home position return request (AxisName.Md.Homing_Request) to become FALSE.

Point

Directly rewriting Home position return request (AxisName.Md.Homing_Request) to FALSE with a program, etc. will not clear the homing request in the Motion system. Be sure to use Home position return request clear (AxisName.Cd.Homing_ClearRequest) to set the homing request to FALSE.

■Home position switch (AbsSwitch)

Sets the proximity dog signal that is transmitted to the slave device in the driver homing method.
Home position switch (AbsSwitch) is set with an MC_INPUT_REF structure. For details of MC_INPUT_REF structures, refer to the following.
☞ Page 63 MC_INPUT_REF
The setting range for Input signal (AbsSwitch.Signal) set in Home position switch (AbsSwitch) is shown below.

Trigger signal (TriggerInput.Signal) is SIGNAL_SELECT structure type.

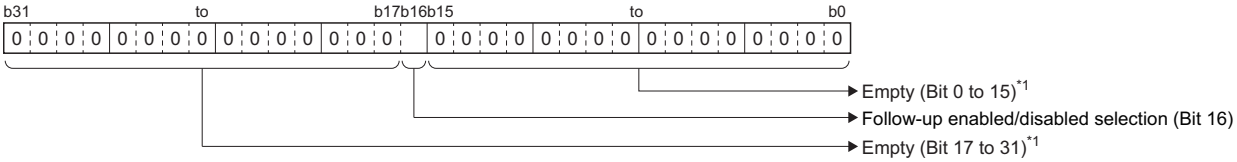
Structure	Variable name	Type	Setting range
SIGNAL_SELECT (Signal select)	Source (Signal)	TARGET_REF ☞ Page 64 TARGET_REF (Input signal)	■Type • BOOL ■Data category • [OBJ]*1 • [VAR] • [DEV] • [CONST]
	Detection (Signal detection method)	INT (MC_SIGNAL_LOGIC) ☞ Page 79 MC_SIGNAL_LOGIC	• 0: Detection at TRUE (HighLevel) • 1: Detection at FALSE (LowLevel)
	CompensationTime (Compensation time)	LREAL	0.0[s]
	FilterTime (Filter time)	LREAL	0.0 to 5.0[s]

*1 An external signal high-accuracy input cannot be used. If a signal for an external signal high-accuracy input is set, it will operate with the same accuracy as a normal signal.

■Options (Options)

Sets the function options used for MC_Home (OPR) by specifying the bit.

Contents that are set by specifying the bit are shown below.



*1 Set empty areas to "0". When a value other than "0" is set, "Out of Options Range (error code: 1A4EH)" occurs.

Bit	Name	Description
16	Follow-up enabled/disabled selection*2	Enables or disables the set position follow-up during driver homing. • 0: Follow-up disabled • 1: Follow-up enabled

*2 When a value other than "0" is set when using Add-on MotionEngine version of "1.18 or earlier", "Out of Options Range (error code: 1A4EH)" occurs.

Precautions

- FBs other than MC_Stop (Forced Stop) cannot start during homing.
- The software stroke limit check for the Target position (Position) at homing start is shown below.

Homing method	Description
Driver homing method	The software stroke limit check is not performed.
Data set homing method	The software stroke limit check is performed.

- The software stroke limit check of the start position is not performed at homing start.
- The software stroke limit is not checked during homing.
- The movement direction at homing completion is in the positive direction.

■Precautions for the driver homing method

- The homing cannot be started during servo-off. Thus, the driver homing methods (Method 35 and 37 (Data set method)), cannot be executed during servo OFF.
- When the external signals (the hardware stroke limit specified with axis variables and the proximity dog specified with Home position switch (AbsSwitch)) are assigned, the external signals are transmitted to the driver. However, when the position accuracy is required in the proximity dog based (not Z-phase based) method, using the driver built-in DI is recommended.
- When Follow-up enabled/disabled selection (Options (Options): Bit 16) is set to "0: Follow-up disabled", the set position is not followed up during homing with the driver homing method. In this case, do not execute synchronous control by setting the axis that is in homing as the master axis. Doing this may cause a "Speed Range Over during Controlling (error code: 1AE8H)" to occur in the slave axis.

- When Follow-up enabled/disabled selection (Options (Options): Bit 16) is set to "1: Follow-up enabled", and synchronous control is executed by setting the axis that is in homing as the master axis, the slave axis operations depending on the master axis data source selection are shown in the following table.

Master axis data source selection (MasterValueSource)	Slave axis operation during synchronous control
1: Set value (mcSetValue) 101: Latest set value (mcLatestSetValue)	Synchronous control continues even after homing completes.
2: Actual value (mcActualValue) 102: Latest actual value (mcLatestActualValue)	Synchronizes with the master axis until homing completes, but a servo alarm [AL. 031.1 (Servo motor speed error)], or [AL. 035.1 (Command frequency error)] may occur without synchronous control being executed at the end.

- When Home position return operation status (AxisName.Md.Homing_Status) is "0002H: Homing attained (target not reached)", and a stop cause occurs, the set position that has been followed up changes significantly. Due to this, even when "1: Set value (mcSetValue)" or "101: Latest set value (mcLatestSetValue)" are set for the Master axis data source selection (MasterValueSource), a servo alarm [AL. 031.1 (Servo motor speed error)] or [AL. 035.1 (Command frequency error) may occur in the slave axis.
- If the following slave object items are not mapped, backup is not guaranteed.
  Home cycle counter (2D3DH)
  Home ABS counter (2D3EH)
- The DOG signal that has been input in the servo amplifier at MR-J5(W)-G connection can be specified as the home position switch signal. For details of the setting method, refer to the following.
  MELSEC iQ-R Motion Module User's Manual (Application)
- If the hardware stroke limit signal is detected by the Motion module during homing, the "HALT" signal is sent to the driver. To perform a stop from the driver side with the limit switch signal, set Hardware stroke limit override (AxisName.Cd.HwStrokeLimit_Override) to "DISABLE (Check disabled)", then temporarily disable the hardware stroke limit check for the Motion module.
- If Hardware stroke limit override (AxisName.Cd.HwStrokeLimit_Override) is set to "ONLY_INSIDE (Check disabled only to inside the range direction)", "Start Not Possible (error code: 1AADH)" occurs and homing is not performed.

Program example

In the following program example, the homing command (bHomingCMD) is set to TRUE and homing is performed for Axis 1 (Axis0001) with Home position switch "[DEV](BOOL)G11478000.1".

■Axis

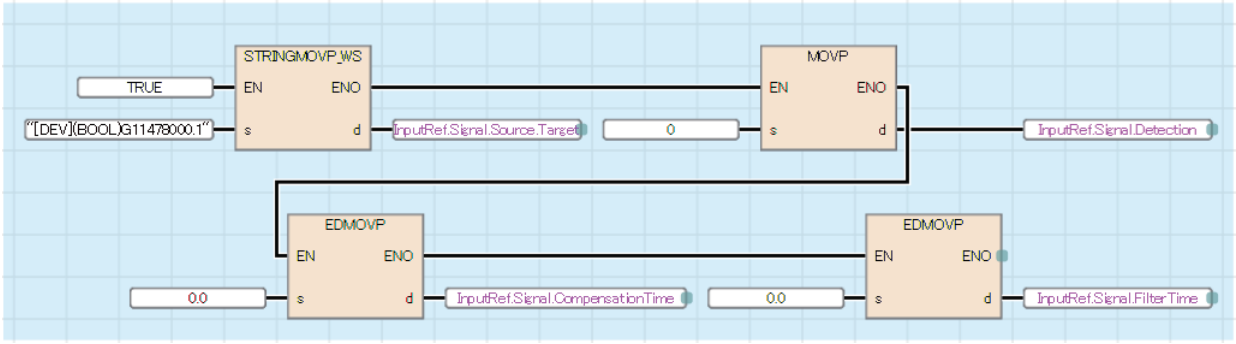
Axis No.	Label name	Data type	Comment
1	Axis0001	AXIS_REF	Axis 1

■Labels used

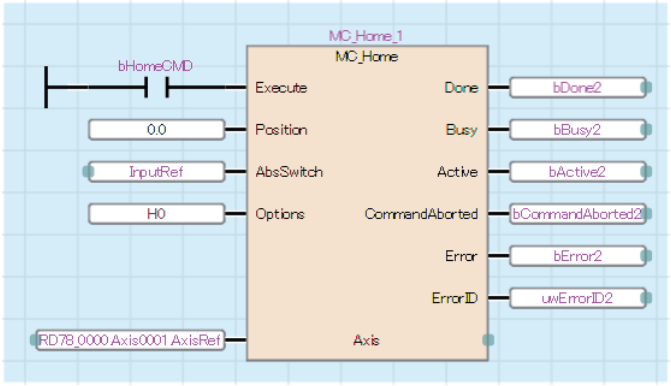
Label name	Data type	Comment
MC_Home_1	MC_Home	OPR FB
bHomingCMD	Bit	Homing command
InputRef	MC_INPUT_REF	Home position switch
bDone	Bit	Execution completion
bBusy	Bit	Executing
bActive	Bit	Controlling
bCommandAborted	Bit	Abortion of execution
bError	Bit	Error
uwErrorID	Word [unsigned]/bit string [16-bit]	Error code

■FBD/LD program (CPU module side)

• Data setting for Home position switch



• Homing



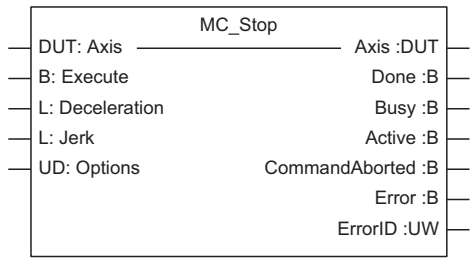
■ST program (Motion module side)

```
//-----Data setting for Home position switch-----
InputRef.Signal.Source.Target:= "[DEV](BOOL)G11478000.1";
InputRef.Signal.Detection:= MC_SIGNAL_LOGIC__HighLevel;
InputRef.Signal.CompensationTime:= 0.0;
InputRef.Signal.FilterTime:= 0.0;

//-----Homing-----
MC_Home_1(
   Axis:= Axis0001.AxisRef ,
   Execute:= bHomingCMD ,
   Position:= 0.0 ,
   AbsSwitch:= InputRef ,
   Options:= H00000000 ,
   Done=> bDone ,
   Busy=> bBusy ,
   Active=> bActive ,
   CommandAborted=> bCommandAborted ,
   Error=> bError ,
   ErrorID=> uwErrorID
);
```

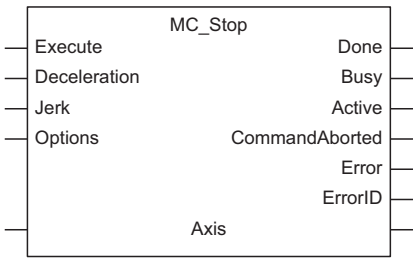
MC_Stop

Decelerates the specified axis to a stop.

Ladder



FBD/LD



ST

```
MC_Stop(
   Axis:= ?AXIS_REF? ,
   Execute:= ?BOOL? ,
   Deceleration:= ?LREAL? ,
   Jerk:= ?LREAL? ,
   Options:= ?DWORD? ,
   Done=> ?BOOL? ,
   Busy=> ?BOOL? ,
   Active=> ?BOOL? ,
   CommandAborted=> ?BOOL? ,
   Error=> ?BOOL? ,
   ErrorID=> ?WORD?
);
```

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Forced Stop	36	8	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
Axis	Axis information	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Execute command	BOOL	At start	TRUE, FALSE	FALSE	When this variable becomes TRUE, executes MC_Stop (Forced Stop).
Deceleration	Deceleration	LREAL	At start/ Retrigger possible	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the deceleration. Contents will differ depending on the acceleration/ deceleration method. For details, refer to "Acceleration/deceleration processing function" in the following manual. 📖MELSEC iQ-R Motion Module User's Manual (Application) *: The acceleration/deceleration method takes over the method specified in the on-going control.
Jerk	Jerk	LREAL	At start/ Retrigger possible	0.0	0.0	Set this variable to "0.0". *: When a value other than "0.0" is set, "Out of Jerk Range (error code: 1A13H)" occurs.
Options	Options	DWORD(HEX)	At start	00000000H	00000000H	Set this variable to "00000000H". *: When a value other than "00000000H" is set, "Out of Options Range (error code: 1A4EH)" occurs.

■Output variables

Output variable	Name	Data type	Default value	Description
Done	Execution completion	BOOL	FALSE	This variable becomes TRUE when the velocity reaches 0.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MC_Stop (Forced Stop) is executed.
Active	Controlling	BOOL	FALSE	This variable becomes TRUE while MC_Stop (Forced Stop) is controlling the axis.

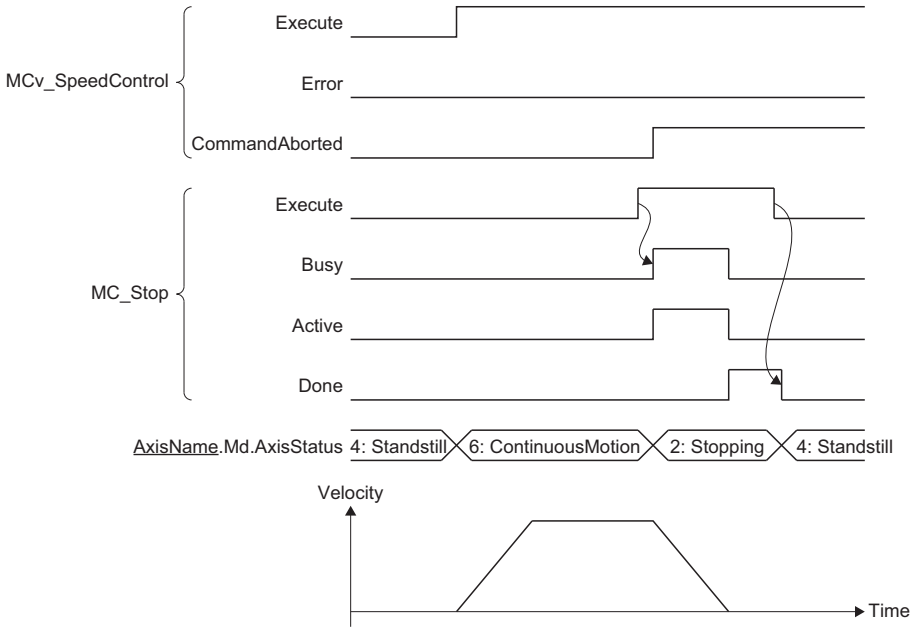
Output variable	Name	Data type	Default value	Description
CommandAborted	Abortion of execution	BOOL	FALSE	This variable becomes TRUE when execution of MC_Stop (Forced Stop) is aborted.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. MELSEC iQ-R Motion Module User's Manual (Application)

Processing details

- MC_Stop (Forced Stop) sets Deceleration (Deceleration) and decelerates the under-control FB to a stop.
- When MC_Stop (Forced Stop) is executed, Abortion of execution (CommandAborted) becomes TRUE for the FB being executed and Axis status (AxisName.Md.AxisStatus) transits to "2: Decelerating to Stop (Stopping)". While Execute command (Execute) is TRUE or if the velocity has not reached "0.0", "2: Decelerating to Stop (Stopping)" is maintained. When Execution completion (Done) becomes TRUE and Execute command (Execute) becomes FALSE at stop completion, the axis status changes to "4: Standby (Standstill)".

■Timing chart

- When the operation completes normally
  When MC_Stop (Forced Stop) is executed during MC_SpeedControl (Speed Control (Including Position Loop))

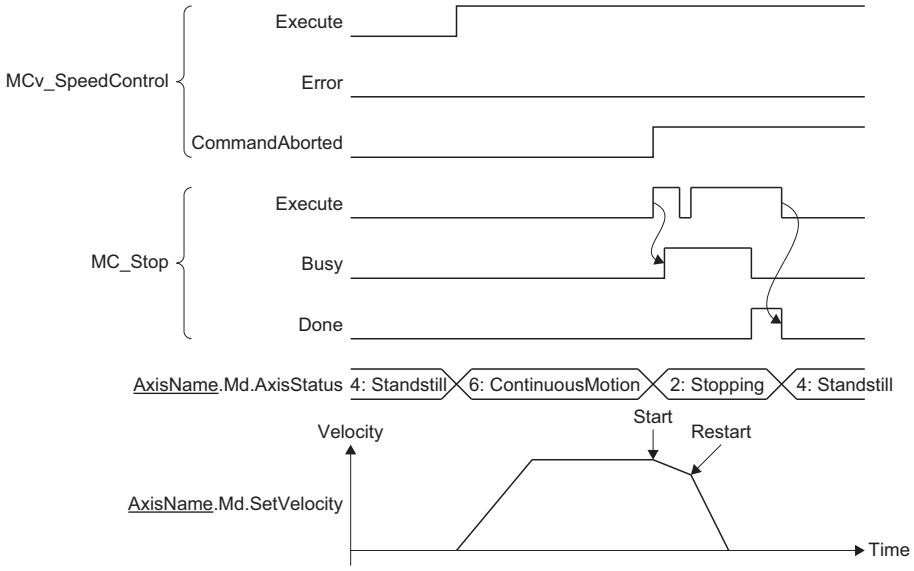


- When an error occurs
  For the timing chart when an error occurs, refer to the following.
  ☞ Page 11 Basic operation of Execute command (Execute) type Motion control FBs

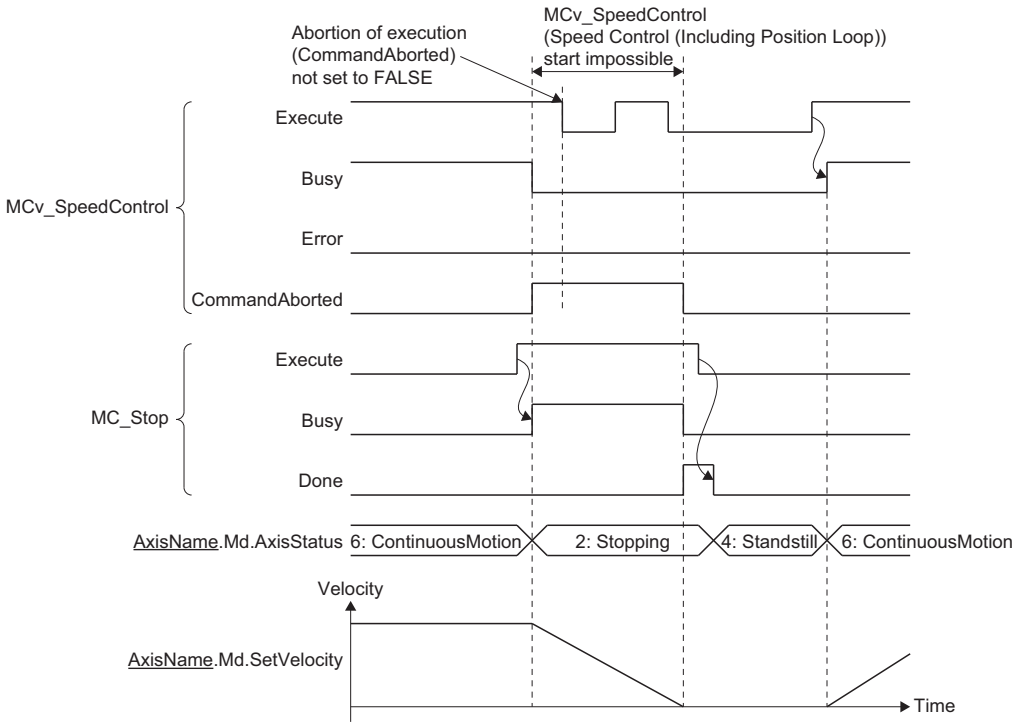
■Operation overview

- When input variable Execute command (Execute) is TRUE during deceleration stop with MC_Stop (Forced Stop), operation commands other than MC_Stop (Forced Stop) are not accepted.
- If Deceleration (Deceleration) is set to "0.0" or omitted, immediate stop will be executed.
- The acceleration/deceleration method and the jerk take over the method specified in the on-going control and decelerate.
- If MC_GroupStop (Group Forced Stop) is executed during single axis synchronous control, synchronization to the master axis will be canceled.

- When the setting value of Deceleration (Deceleration) is changed and MC_Stop (Forced Stop) is restarted, the deceleration stop is executed from the restart point based on the value of Deceleration (Deceleration) in MC_Stop (Forced Stop).



- When an FB that operates axes other than MC_Stop (Forced Stop) is executed while Axis status (AxisName.Md.AxisStatus) is set to "2: Decelerating to Stop (Stopping)", "Start Not Possible (error code: 1AADH)" occurs and Error (Error) becomes TRUE for MC_Stop (Forced Stop) and Axis status (AxisName.Md.AxisStatus) changes to "1: Stopping on Error (ErrorStop)".
- After the axis stop, the axis status is transited to "4: Standby (Standstill)" by executing Axis error reset (AxisName.Cd.ErrorReset).
- Even if MC_Stop (Forced Stop) is used to set Execute command (Execute) to FALSE for the FB whose execution was aborted, once the stop operation starts, Abortion of execution (CommandAborted) remains TRUE until the stop is completed. After the stop completes, Abortion of execution (CommandAborted) becomes FALSE if Execute command (Execute) is FALSE for the FB whose execution was aborted.



Program example

In the following program example, the axis stop request (bStop) is set to TRUE, and Axis 1 (Axis0001) is decelerated to a stop at a deceleration of "100000.0".

■Axis

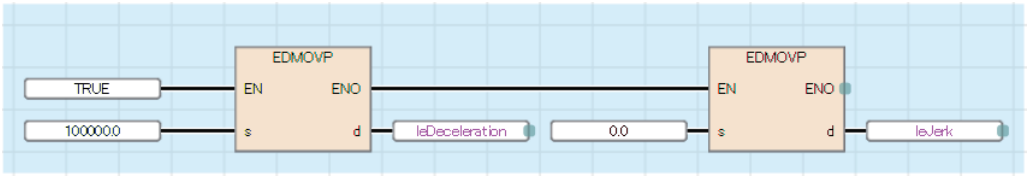
Axis No.	Label name	Data type	Comment
1	Axis0001	AXIS_REF	Axis 1

■Labels used

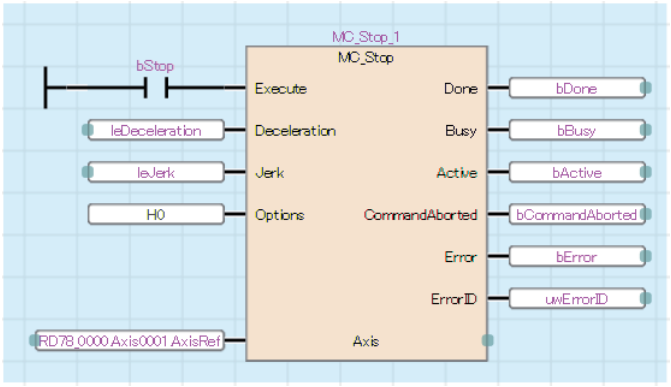
Label name	Data type	Comment
MC_Stop_1	MC_Stop	Forced Stop FB
bStop	Bit	Axis stop request
leDeceleration	Double-precision real number	Deceleration
leJerk	Double-precision real number	Jerk
bDone	Bit	Completion
bBusy	Bit	Executing
bActive	Bit	Controlling
bCommandAborted	Bit	Abortion of execution
bError	Bit	Error
uwErrorID	Word [unsigned]/bit string [16-bit]	Error code

■FBD/LD program (CPU module side)

• Data setting for axis stop



• Axis stop



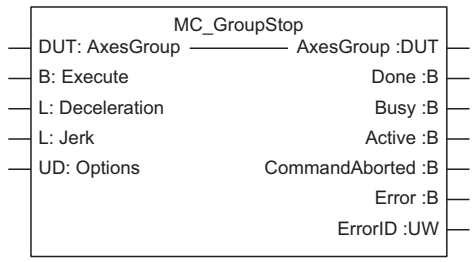
■ST program (Motion module side)

```
//-----Data setting for axis stop-----
leDeceleration:= 100000.0;
leJerk:= 0.0;

//-----Axis stop-----
MC_Stop_1(
   Axis:= Axis0001.AxisRef ,
   Execute:= bStop ,
   Deceleration:= leDeceleration ,
   Jerk:= leJerk ,
   Options:= H00000000 ,
   Done=> bDone ,
   Busy=> bBusy ,
   Active=> bActive ,
   CommandAborted=> bCommandAborted ,
   Error=> bError ,
   ErrorID=> uwErrorID
);
```

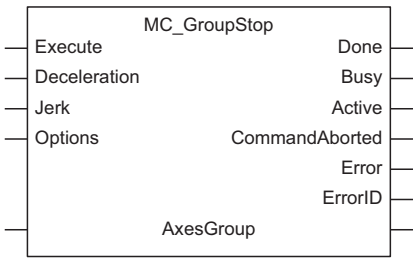
MC_GroupStop

Decelerates the specified axes group to a stop.

Ladder



FBD/LD



ST

```
MC_GroupStop(
  AxesGroup:= ?AXES_GROUP_REF? ,
  Execute:= ?BOOL? ,
  Deceleration:= ?LREAL? ,
  Jerk:= ?LREAL? ,
  Options:= ?DWORD? ,
  Done=> ?BOOL? ,
  Busy=> ?BOOL? ,
  Active=> ?BOOL? ,
  CommandAborted=> ?BOOL? ,
  Error=> ?BOOL? ,
  ErrorID=> ?WORD?
);
```

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Group Forced Stop	36	8	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
AxesGroup	Axes group information	AXES_GROUP_REF	At start	—	Omission not possible	This variable sets the axes group. For the variables used (AxesGroupName.AxesGroupRef.), refer to the following. ☞ Page 46 AxesGroupName.AxesGroupRef. (Axes group information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Execute command	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MC_GroupStop (Group Forced Stop).
Deceleration	Deceleration	LREAL	At start/ Retrigger possible	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the deceleration. Contents will differ depending on the acceleration/ deceleration method. For details, refer to "Acceleration/deceleration processing function" in the following manual. 📖MELSEC iQ-R Motion Module User's Manual (Application) *: The acceleration/deceleration method takes over the method specified in the on-going control.
Jerk	Jerk	LREAL	At start	0.0	0.0	Set this variable to "0.0". *: If a value other than "0.0" is set, "Out of Jerk Range (error code: 1A13H)" occurs.
Options	Options	DWORD(HEX)	At start	00000000H	00000000H	Set this variable to "00000000H". *: When a value other than "00000000H" is set, "Out of Options Range (error code: 1A4EH)" occurs.

■Output variables

Output variable	Name	Data type	Default value	Description
Done	Execution completion	BOOL	FALSE	This variable becomes TRUE when the velocity reaches 0.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when operating the velocity toward 0.

Output variable	Name	Data type	Default value	Description
Active	Controlling	BOOL	FALSE	This variable becomes TRUE while MC_GroupStop (Group Forced Stop) is controlling the axes.
CommandAborted	Abortion of execution	BOOL	FALSE	This variable becomes TRUE when execution of MC_GroupStop (Group Forced Stop) is aborted.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. MELSEC iQ-R Motion Module User's Manual (Application)

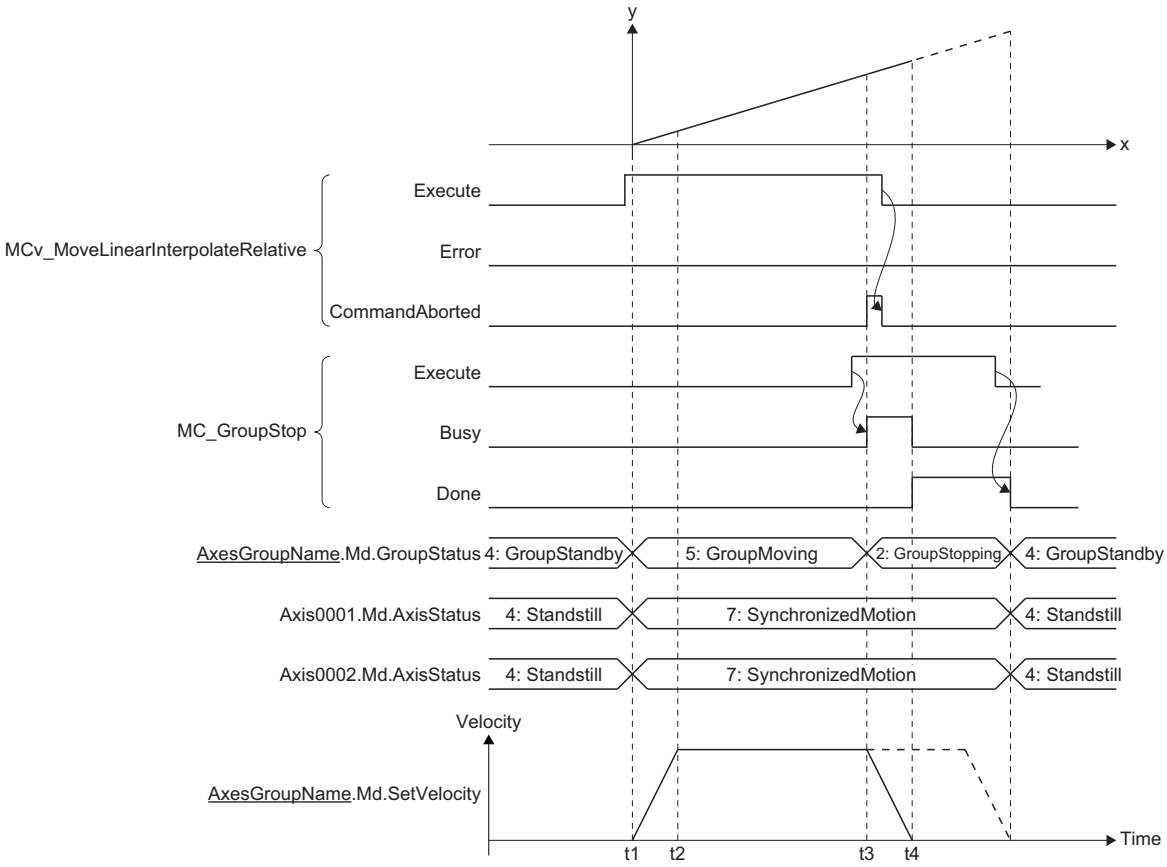
Processing details

- MC_GroupStop (Group Forced Stop) sets Deceleration (Deceleration) and decelerates the under-control FB to a stop. The axes group passes through the previous operation path and decelerates to a stop.
- When MC_GroupStop (Group Forced Stop) is executed, Abortion of execution (CommandAborted) of the FB being executed becomes TRUE and Axes group status (AxesGroupName.Md.AxisStatus) transits to "2: Decelerating to Stop (GroupStopping)". While Execute command (Execute) is TRUE or if the velocity has not reached 0, "2: Decelerating to Stop (GroupStopping)" is maintained. When Execution completion (Done) becomes TRUE and Execute command (Execute) becomes FALSE at stop completion, the axis group status changes to "4: Standby (GroupStandby)".

■Timing chart

- When the operation completes normally

  When MC_GroupStop (Group Forced Stop) is executed during MCv_MoveLinearInterpolateRelative (Relative Value Linear Interpolation Control)



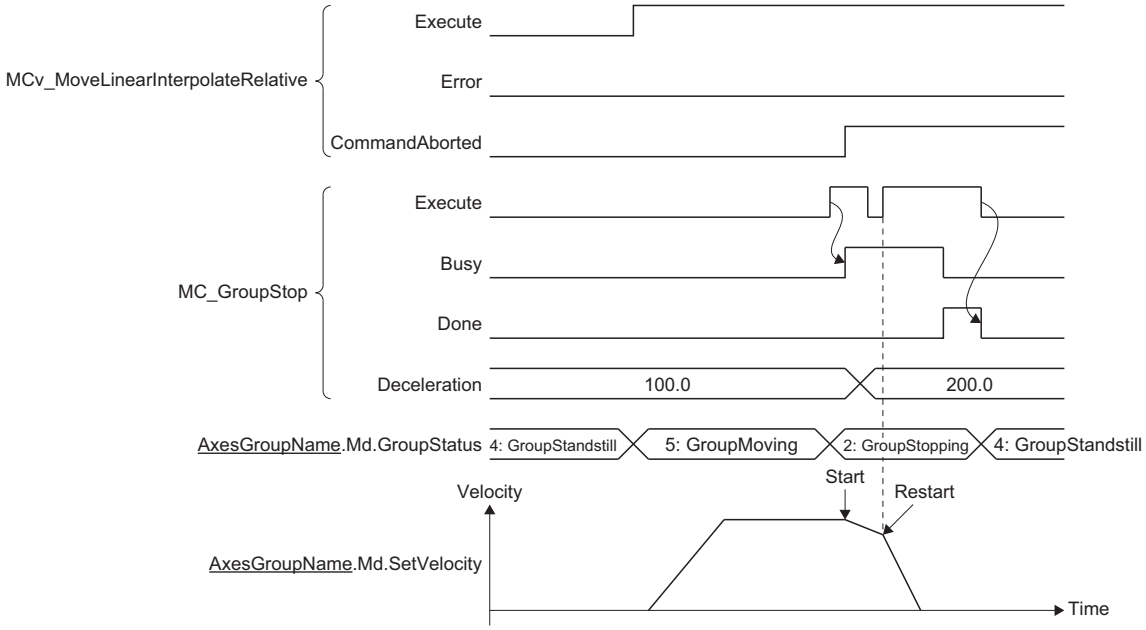
- When an error occurs

  For the timing chart when an error occurs, refer to the following.

  ☞ Page 11 Basic operation of Execute command (Execute) type Motion control FBs

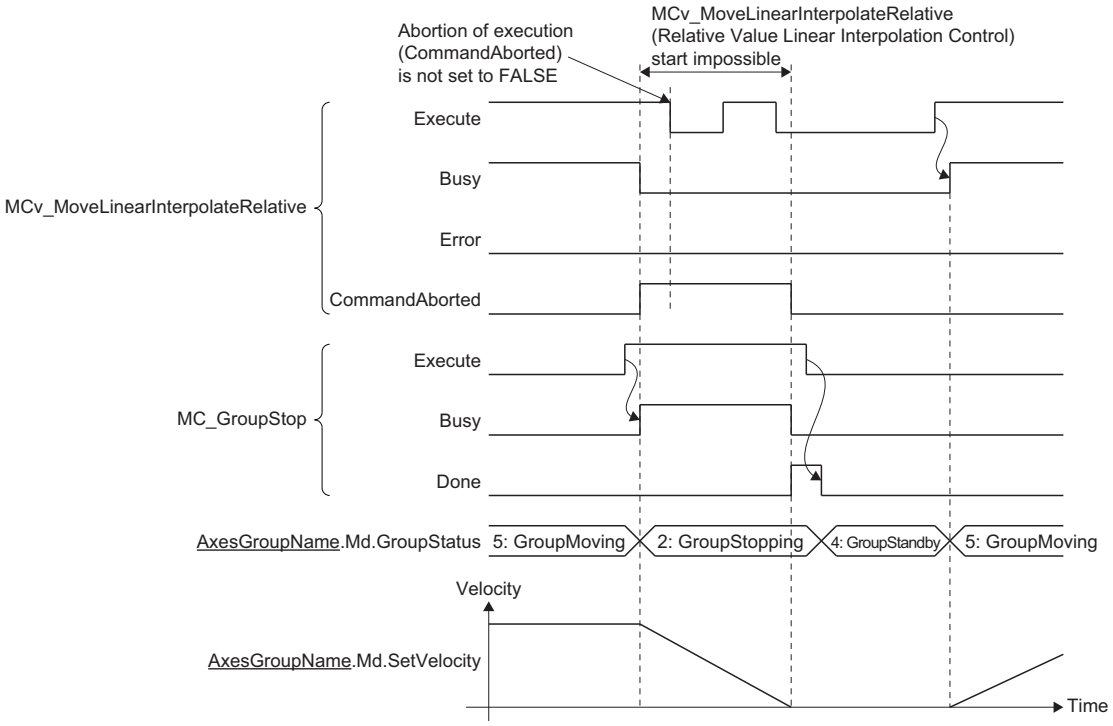
■Operation overview

- When input variable Execute command (Execute) is TRUE during deceleration stop with MC_GroupStop (Group Forced Stop), operation commands other than MC_GroupStop (Group Forced Stop) are not accepted.
- If Deceleration (Deceleration) is set to "0.0" or omitted, immediate stop will be executed.
- The acceleration/deceleration method and the jerk take over the method specified in the on-going control and decelerate.
- When the setting value of Deceleration (Deceleration) is changed and MC_GroupStop (Group Forced Stop) is restarted, the deceleration stop is executed from the restart point based on the value of Deceleration (Deceleration) in MC_GroupStop (Group Forced Stop) and Jerk (Jerk) of the under-control FB.



- When an FB that operates axes other than MC_GroupStop (Forced Stop) is executed while Axes group status (AxesGroupName.Md.AxisStatus) is set to "2: Decelerating to Stop (GroupStopping)", "Start Not Possible (error code: 1AADH)" occurs and Error (Error) becomes TRUE for MC_GroupStop (Group Forced Stop) and Axes group status (AxesGroupName.Md.AxisStatus) changes to "1: Stopping on Error (GroupErrorStop)".

- Even if MC_GroupStop (Group Forced Stop) is used to set Execute command (Execute) to FALSE for the FB whose execution was aborted, once the stop operation starts, Abortion of execution (CommandAborted) remains TRUE until the stop is completed. After the stop completes, Abortion of execution (CommandAborted) becomes FALSE if Execute command (Execute) is FALSE for the FB whose execution was aborted.



Program example

In the following program example, the axes group stop request (bGroupStop) is set to TRUE, and Axes group 1 (AxesGroup001) is decelerated to a stop at a deceleration of "100000.0".

■Axes group

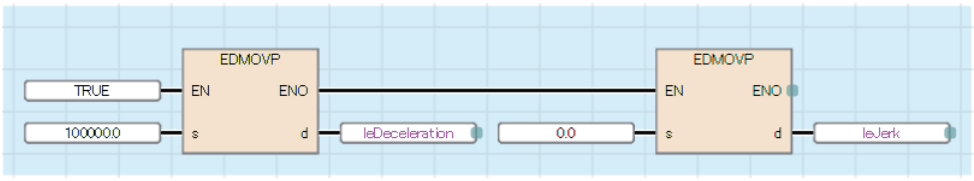
Axes group No.	Label name	Data type	Comment
1	AxesGroup001	AXES_GROUP_REF	Axes group 1

■Labels used

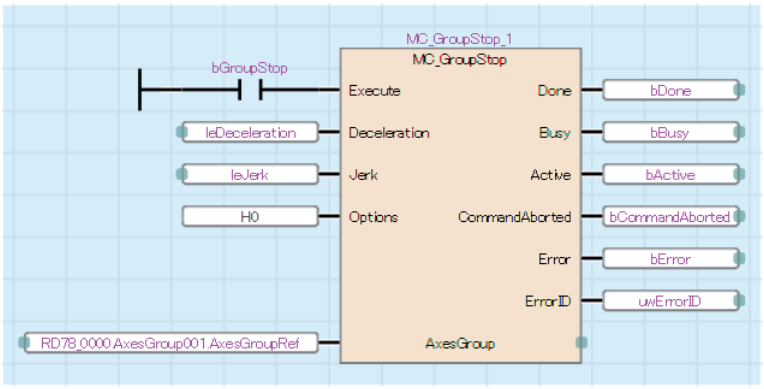
Label name	Data type	Comment
MC_GroupStop_1	MC_GroupStop	Axes group stop FB
bGroupStop	Bit	Axes group stop request
leDeceleration	Double-precision real number	Deceleration
leJerk	Double-precision real number	Jerk
bDone	Bit	Completion
bBusy	Bit	Executing
bActive	Bit	Controlling
bCommandAborted	Bit	Abortion of execution
bError	Bit	Error
uwErrorID	Word [unsigned]/bit string [16-bit]	Error code

■FBD/LD program (CPU module side)

• Data setting for axes group stop



• Axes group stop



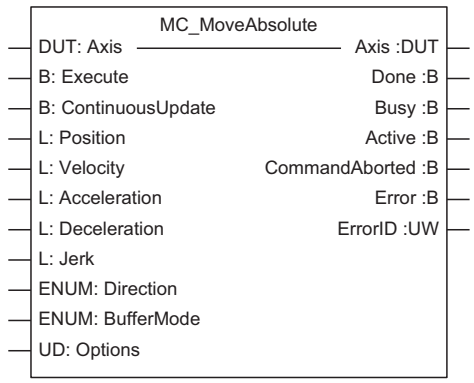
■ST program (Motion module side)

```
//-----Data setting for axes group stop-----
leDeceleration:= 100000.0;
leJerk:= 0.0;

//-----Axes group stop-----
MC_GroupStop_1(
   AxesGroup:= AxesGroup001.AxesGroupRef ,
   Execute:= bGroupStop ,
   Deceleration:= leDeceleration ,
   Jerk:= leJerk ,
   Options:= H00000000 ,
   Done=> bDone ,
   Busy=> bBusy ,
   Active=> bActive ,
   CommandAborted=> bCommandAborted ,
   Error=> bError ,
   ErrorID=> uwErrorID
);
```

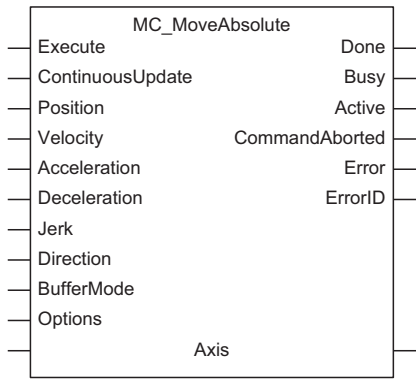
MC_MoveAbsolute

Executes positioning after setting the target position of the absolute position.

Ladder



FBD/LD



ST

```
MC_MoveAbsolute(
   Axis:= ?AXIS_REF? ,
   Execute:= ?BOOL? ,
   ContinuousUpdate:= ?BOOL? ,
   Position:= ?LREAL? ,
   Velocity:= ?LREAL? ,
   Acceleration:= ?LREAL? ,
   Deceleration:= ?LREAL? ,
   Jerk:= ?LREAL? ,
   Direction:= ?INT? ,
   BufferMode:= ?INT? ,
   Options:= ?DWORD? ,
   Done=> ?BOOL? ,
   Busy=> ?BOOL? ,
   Active=> ?BOOL? ,
   CommandAborted=> ?BOOL? ,
   Error=> ?BOOL? ,
   ErrorID=> ?WORD?
);
```

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Absolute Value Positioning	64	8	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
Axis	Axis information	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Execute command	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MC_MoveAbsolute (Absolute Value Positioning).
ContinuousUpdate	Continuous update	BOOL	At start	TRUE, FALSE	FALSE	This variable sets whether to enable or disable continuous change of Target position (Position), Velocity (Velocity), Acceleration (Acceleration), and Deceleration (Deceleration). • FALSE: Disable • TRUE: Enable
Position	Target position	LREAL	At start/ Retrigger possible/ Continuous update possible	-10000000000.0 to 10000000000.0	0.0	This variable sets the target position of the absolute position. The settable range varies for each setting. For details, refer to the following. ☞ Page 202 Target position (Position)
Velocity	Velocity	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0, 0.0001 to 2500000000.0	0.0	This variable sets the velocity. For details, refer to the following. ☞ Page 203 Velocity (Velocity)

Input variable	Name	Data type	Input	Setting range	Default value	Description
Acceleration	Acceleration	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the acceleration. For details, refer to the following. ☞ Page 203 Acceleration (Acceleration)
Deceleration	Deceleration	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the deceleration. For details, refer to the following. ☞ Page 203 Deceleration (Deceleration)
Jerk	Jerk	LREAL	At start	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the jerk. For details, refer to the following. ☞ Page 204 Jerk (Jerk)
Direction	Direction selection	INT (MC_DIRECTION)	At start	1 to 3	0	When software stroke limit is disabled, this variable sets the direction used to move from the current position to the target position. • 1: Positive direction (mcPositiveDirection) • 2: Negative direction (mcNegativeDirection) • 3: Shortest path (mcShortestWay) *: When this setting is omitted, "Out of Direction Selection Range (error code: 1A37H)" occurs. For details, refer to the following. ☞ Page 204 Direction selection (Direction)
BufferMode	Buffer mode	INT (MC_BUFFER_MODE)	At start	0 to 5	0	This variable sets the buffer mode. • 0: Aborting (mcAborting) • 1: Buffered (mcBuffered) • 2: BlendingLow (mcBlendingLow) • 3: BlendingPrevious (mcBlendingPrevious) • 4: BlendingNext (mcBlendingNext) • 5: BlendingHigh (mcBlendingHigh) For details, refer to the following. ☞ Page 208 Buffer mode (BufferMode)
Options	Options	DWORD(HEX)	At start	00000000H to 00010021H	00000000H	This variable sets the function options for MC_MoveAbsolute (Absolute Value Positioning). For details, refer to the following. ☞ Page 208 Options (Options)

■Output variables

Output variable	Name	Data type	Default value	Description
Done	Execution completion	BOOL	FALSE	This variable becomes TRUE when the target position is reached.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MC_MoveAbsolute (Absolute Value Positioning) is being executed. This variable becomes FALSE after reaching the target position.
Active	Controlling	BOOL	FALSE	This variable becomes TRUE while MC_MoveAbsolute (Absolute Value Positioning) is controlling the axis. This variable becomes FALSE after reaching the target position.
CommandAborted	Abortion of execution	BOOL	FALSE	This variable becomes TRUE when execution of MC_MoveAbsolute (Absolute Value Positioning) is aborted. This variable becomes FALSE when Execute command (Execute) becomes FALSE.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. 📖MELSEC iQ-R Motion Module User's Manual (Application)

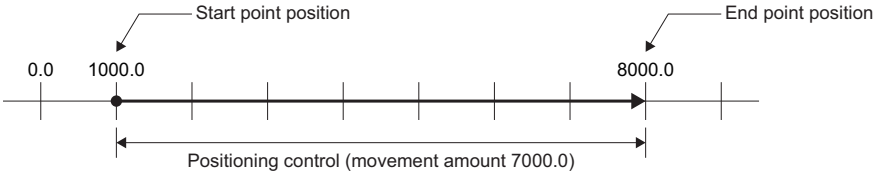
Processing details

- Sets Target position (Position), Velocity (Velocity), Acceleration (Acceleration), Deceleration (Deceleration), Jerk (Jerk), Direction selection (Direction), Buffer mode (BufferMode), and Options (Options), then executes positioning from the current position at start (start point position) to the specified position (end point position) set in Target position (Position).

Ex.

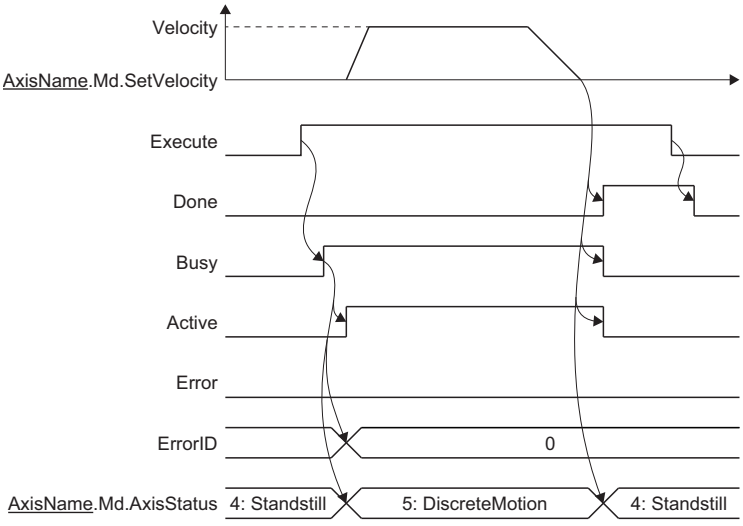
When the start point position (current stop position) is "1000.0" and Target position (Position) is set to "8000.0"

· Positioning is performed in the positive direction for a movement amount of "7000.0 (8000.0 - 1000.0)"



■Timing chart

- When the operation completes normally



- When an error occurs

  For the timing chart when an error occurs, refer to the following.

  ☞ Page 11 Basic operation of Execute command (Execute) type Motion control FBs

■Target position (Position)

Sets the target position of the absolute position. The valid range differs depending on whether the software stroke limit is valid/invalid, Direction selection (Direction), and the setting for Target position specification exceeding the ring counter (Options (Options): Bit 16).

- When the software stroke limit is valid (when Software stroke limit target (AxisName.Md.SwStrokeLimit_Target) is set to a value other than "-1: Disabled (Invalid)")

  The valid range is as follows regardless of Direction selection (Direction) and the setting for Target position specification exceeding the ring counter (Options (Options): Bit 16).

  If a value outside the range is set, "Out of Target Position Range (error code: 1A05H)" occurs and the axis will not start.

  In addition, even when the value set is within the valid range, "Software Stroke Limit Over (Target Position) (error code: 1A00H)" occurs and the axis will not start if a position which exceeds the software stroke limit is specified as the target position.

Valid range
Ring counter lower limit value ≤ Target position < Ring counter upper limit value

- When the software stroke limit is invalid (when Software stroke limit target (AxisName.Md.SwStrokeLimit_Target) is set to "-1: Disabled (Invalid)")

  Setting "Target position specification exceeding the ring counter (Options (Options): Bit 16)" to "1: Allow" allows execution of positioning that exceeds the ring counter range by one revolution or more.

  If a value outside the range is set, "Out of Target Position Range (error code: 1A05H)" occurs and the axis will not start.

Direction selection (Direction)	Target position specification exceeding the ring counter (Options (Options): Bit 16)	Valid range
1: Positive direction (mcPositiveDirection)	0: Do not allow	Ring counter lower limit value ≤ Target position < Ring counter upper limit value
	1: Allow	Ring counter lower limit value ≤ Target position < Positioning range upper limit value
2: Negative direction (mcNegativeDirection)	0: Do not allow	Ring counter lower limit value ≤ Target position < Ring counter upper limit value
	1: Allow	Positioning range upper limit value ≤ Target position < Ring counter upper limit value
3: Shortest path (mcShortestWay)	0: Do not allow	Ring counter lower limit value ≤ Target position < Ring counter upper limit value
	1: Allow	Ring counter lower limit value ≤ Target position < Ring counter upper limit value

- For the direction used to move to the target position, refer to the following.

  ☞ Page 204 Direction selection (Direction)

■Velocity (Velocity)

Sets the commanded velocity used in MC_MoveAbsolute (Absolute Value Positioning).

Setting range
0.0, 0.0001 to 2500000000.0[*1][*2]

*1 To perform floating-point operation, the lower limit value of the commanded velocity is limited by the following restrictions. If the velocity that is converted from the target velocity into the operation cycle is less than "0.00001", "Out of Calculation Cycle Converted Speed Range (error code: 1AE4H)" occurs. (If the velocity is changed, "Out of Calculation Cycle Converted Speed Range Warning (warning code: 0D2FH)" occurs instead). To improve the precision of floating-point operation, change Position command unit (AxisName.Pr.Unit_Position) and Speed command unit (AxisName.Pr.Unit_Velocity) and perform settings so that the velocity converted into the operation cycle is not less than "0.00001".

*2 When a Motion control FB with a set velocity of "0.0" performs a multiple start, the set velocity of the previous Motion control FB is used.

■Acceleration (Acceleration)

Sets the acceleration used in MC_MoveAbsolute (Absolute Value Positioning).

The setting range differs depending on the setting for Acceleration/deceleration method setting (Options (Options): Bit 0 to 2).

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000[*1], positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	0.000000[*1], positive numbers from 0.000001 to 8400.0[s]

*1 The operation performed at start varies depending on the setting of Operation selection at start acceleration/deceleration 0 (AxisName.Pr.AccelerationZeroBehavior). When changing the acceleration or deceleration, the changes are not accepted.

■Deceleration (Deceleration)

Sets the deceleration used in MC_MoveAbsolute (Absolute Value Positioning).

Specifies the setting range if Acceleration/deceleration method setting (Options (Options): Bit 0 to 2) is set to "0: Acceleration/deceleration specification method (mcAccDec)". Deceleration (Deceleration) is not used if the acceleration/deceleration method is set to "1: Acceleration/deceleration time-fixed method (mcFixedTime)".

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000[*1], positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	Not used

*1 The operation performed at start varies depending on the setting of Operation selection at start acceleration/deceleration 0 (AxisName.Pr.AccelerationZeroBehavior). When changing the acceleration or deceleration, the changes are not accepted.

■Jerk (Jerk)

Sets the jerk used in MC_MoveAbsolute (Absolute Value Positioning).

Specifies the setting range if Acceleration/deceleration method setting (Options (Options): Bit 0 to 2) is set to "0: Acceleration/deceleration specification method (mcAccDec)". Jerk (Jerk) is not used if the acceleration/deceleration method is set to "1: Acceleration/deceleration time-fixed method (mcFixedTime)".

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000, positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	Not used

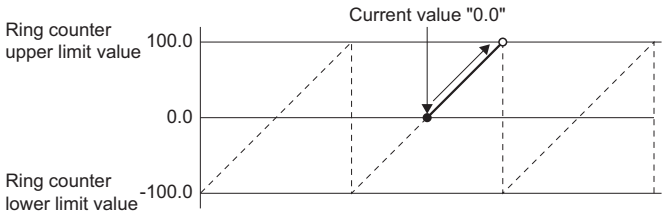
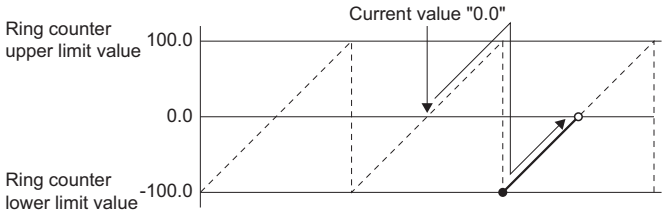
■Direction selection (Direction)

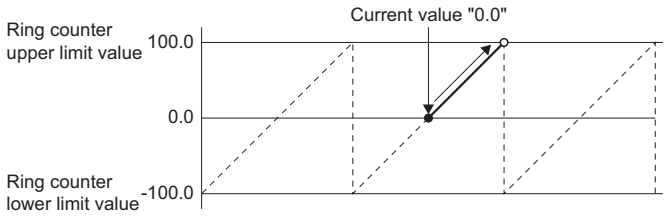
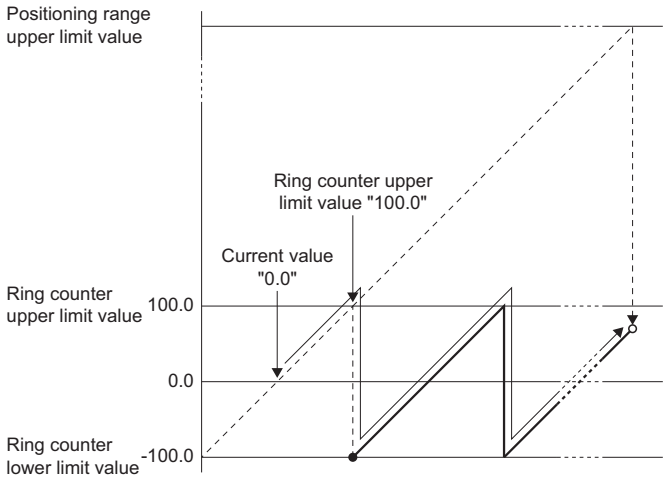
The setting for Direction selection (Direction) is ignored when the software stroke limit is valid. Performs positioning control in a direction which does not cross over into the area outside the software stroke limit range. However, when the value does not cross over into the area outside the software stroke limit range both in the positive direction and the negative direction, positioning control is performed in the direction closest to the target position (the direction in which the absolute value of the movement amount is shortest) based on the current position. If the distance is the same in both the positive direction and the negative direction, control is performed using the current direction.

When the software stroke limit is invalid, the direction used to move from the current position to the target position can be set by selecting from "1: Positive direction (mcPositiveDirection)", "2: Negative direction (mcNegativeDirection)", or "3: Shortest path (mcShortestWay)" in Direction selection (Direction).

- 1: Positive direction (mcPositiveDirection)

  Performs positioning from the current position to the target position in the positive direction (address increasing).

Target position specification exceeding the ring counter (Options (Options): Bit 16)	Setting range	Operation
■0: Do not allow The setting range of the target position is "Ring counter lower limit value ≤ Target position < Ring counter upper limit value".	Current value ≤ Target position < Ring counter upper limit value	
	Ring counter lower limit value ≤ Target position < Current value	

Target position specification exceeding the ring counter (Options (Options): Bit 16)	Setting range	Operation
■1: Allow Allows positioning operation that overflows the ring counter upper limit value to be performed. For this setting, the ring counter upper limit value is used as a reference to perform positioning using the amount of movement exceeded as the relative movement amount. If the target position is set to a value smaller than the ring counter lower limit value, positioning does not start and "Out of Target Position Range (error code: 1A05H)" occurs.	• Current value ≤ Target position < Ring counter upper limit value • Ring counter lower limit value ≤ Target position < Current value	
	Ring counter upper limit value ≤ Target position < Positioning range upper limit value	

Ex.

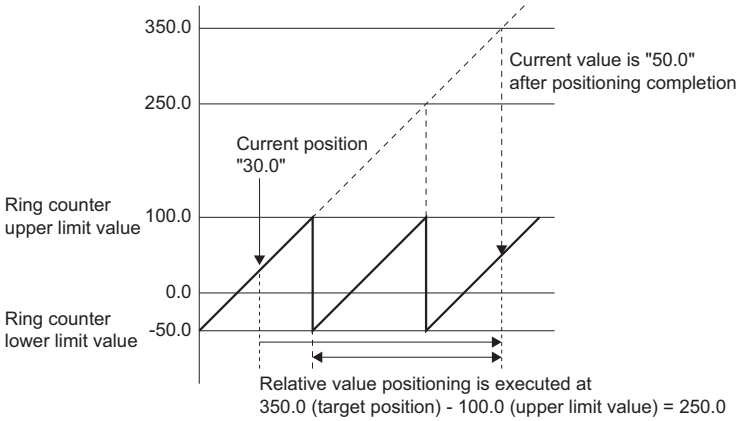
Operation when set to the values shown below (performs relative value positioning after exceeding the ring counter upper limit value).

· Ring counter upper limit value: 100.0

· Ring counter lower limit value: -50.0

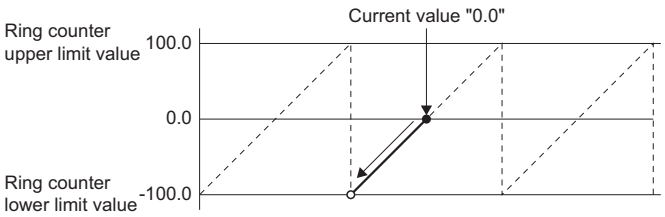
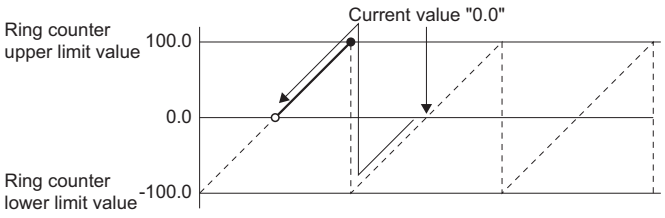
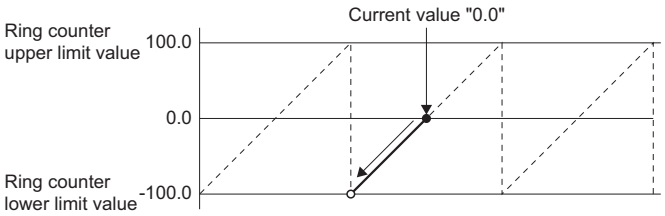
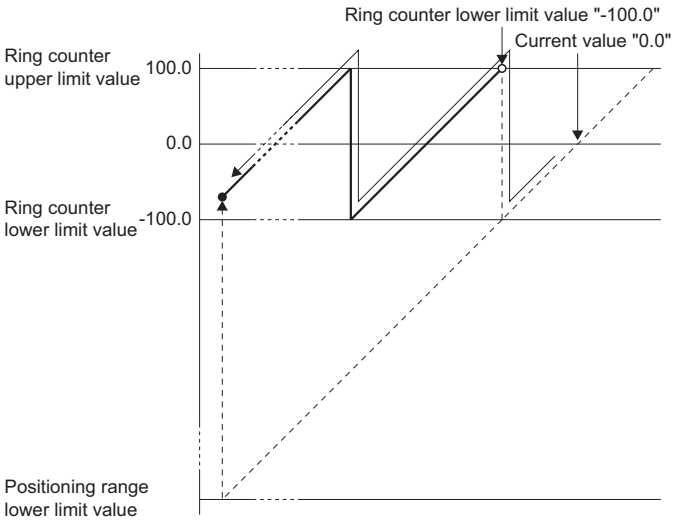
· Current position: 30.0

· Target position: 350.0



• 2: Negative direction (mcNegativeDirection)

Performs positioning in the negative direction (address decreasing) from the current position to the target position.

Target position specification exceeding the ring counter (Options (Options): Bit 16)	Setting range	Operation
■0: Do not allow The setting range of the target position is "Ring counter lower limit value ≤ Target position < Ring counter upper limit value".	Ring counter lower limit value ≤ Target position ≤ Current value	
	Current value < Target position < Ring counter upper limit value	
■1: Allow Allows positioning operation that underflows the ring counter lower limit value to be performed. For this setting, the ring counter lower limit value is used as a reference to perform positioning using the amount of movement exceeded as the relative movement amount. If the target position is set to a value larger than the ring counter upper limit value, positioning does not start and "Out of Target Position Range (error code: 1A05H)" occurs.	• Ring counter lower limit value ≤ Target position ≤ Current value • Positioning range lower limit value < Target position < Ring counter upper limit value	
	Positioning range lower limit value ≤ Target position < Ring counter lower limit value	

Ex.

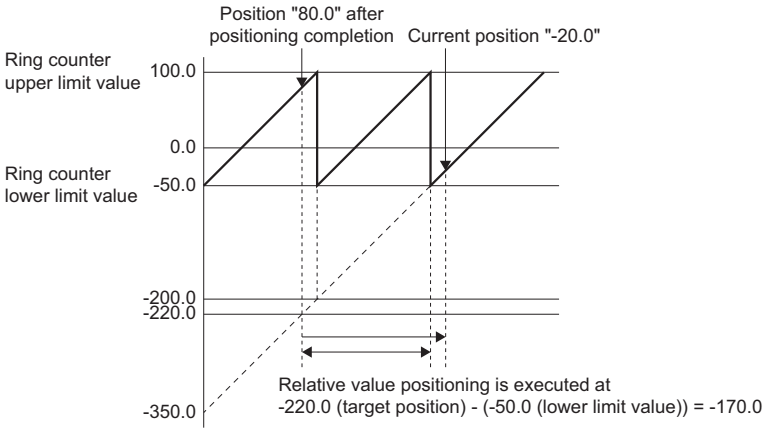
Operation when set to the values shown below

· Ring counter upper limit value: 100.0

· Ring counter lower limit value: -50.0

· Current position: -20.0

· Target position: -220.0



- 3: Shortest path (mcShortestWay)

  Based on the current position, positioning control is performed in the direction closest to the target position (the direction in which the absolute value of the movement amount is shortest). If the distance is the same in both the positive direction and the negative direction, control is performed using the current direction. Regardless of the setting for Target position specification exceeding the ring counter (Options (Options): Bit 16), the setting range for the target position is "Ring counter lower limit value ≤ Target position < Ring counter upper limit value". If a value outside the range is set, "Out of Target Position Range (error code: 1A05H)" occurs and the axis will not start.

Ex.

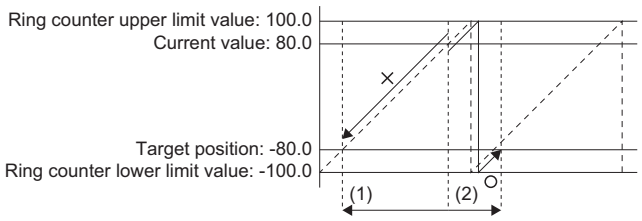
Operation when set to the values shown below

· Ring counter upper limit value: 100.0

· Ring counter lower limit value: -100.0

· Current position: 80.0

· Target position: -80.0



*1 Since the movement amount of (1) is "160.0" and the movement amount of (2) is "40.0", positioning moves in the positive direction from (2).

■Buffer mode (BufferMode)

Sets the operation used to perform multiple start (buffer mode).

The following buffer modes are settable in MC_MoveAbsolute (Absolute Value Positioning).

Setting value	Description
0: Aborting (mcAborting)	Aborts (cancels) the under-control FB and executes the next FB immediately.
1: Buffered (mcBuffered)	Buffers the next FB on the under-control FB. If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after completion of the under-control FB.
2: BlendingLow (mcBlendingLow)	Buffers the next FB on the under-control FB.*1 If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. When the under-control FB reaches the target position, the lower target velocity between the under-control FB and the buffering FB is used as the switching speed.
3: BlendingPrevious (mcBlendingPrevious)	Buffers the next FB on the under-control FB.*1 If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. When the under-control FB reaches the target position, the target velocity of the under-control FB is used as the switching speed.
4: BlendingNext (mcBlendingNext)	Buffers the next FB on the under-control FB.*1 If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. When the under-control FB reaches the target position, the target velocity of the buffering FB is used as the switching speed.
5: BlendingHigh (mcBlendingHigh)	Buffers the next FB on the under-control FB.*1 If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. When the under-control FB reaches the target position, the higher target velocity between the under-control FB and the buffering FB is used as the switching speed.

*1 Stops are not performed between the under-control FB and the buffering FB.

Point

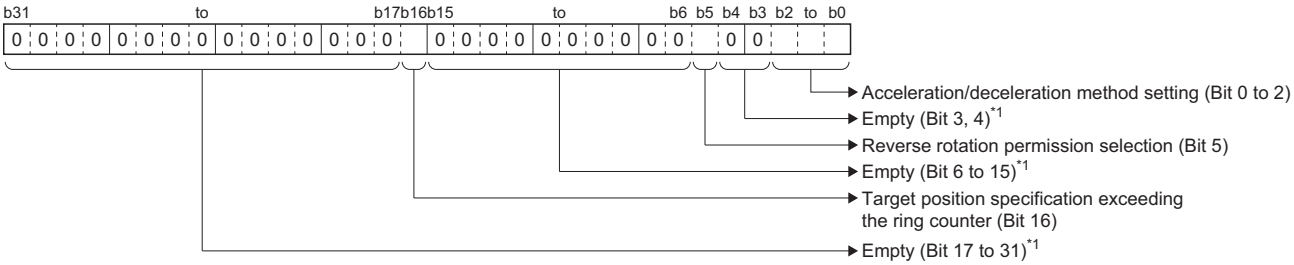
For details of multiple start (buffer mode), refer to the following.

MELSEC iQ-R Motion Module User's Manual (Application)

■Options (Options)

Sets the function options used for MC_MoveAbsolute (Absolute Value Positioning) by specifying the bit.

Contents that are set by specifying the bit are shown below.



*1 Set empty areas to "0". When a value other than "0" is set, "Out of Options Range (error code: 1A4EH)" occurs.

Bit	Name	Description
0 to 2	Acceleration/deceleration method setting	Sets the acceleration/deceleration method used to perform control. • 0: Acceleration/deceleration specification method (mcAccDec) • 1: Acceleration/deceleration time-fixed method (mcFixedTime)
5	Reverse rotation permission selection	Sets whether to allow reverse rotation when the operation direction of the FB being executed and the buffering FB are different. • 0: Allow • 1: Do not allow *: This bit is valid when setting "0: Aborting (mcAborting)" in Buffer mode (BufferMode) and when changing the target position.
16	Target position specification exceeding the ring counter	Sets whether to allow a target position that exceeds the ring counter upper limit value or lower limit value when the software stroke limit is invalid. • 0: Do not allow • 1: Allow

• Acceleration/deceleration method setting (Bit 0 to 2)

Setting value	Description
0: Acceleration/deceleration specification method (mcAccDec)	An acceleration/deceleration method using Acceleration (Acceleration), Deceleration (Deceleration), and Jerk (Jerk) set in MC_MoveAbsolute (Absolute Value Positioning).
1: Acceleration/deceleration time-fixed method (mcFixedTime)	An acceleration/deceleration method using the acceleration/deceleration time specified in MC_MoveAbsolute (Absolute Value Positioning) regardless of the velocity. The acceleration/deceleration time does not use Acceleration (Acceleration), Deceleration (Deceleration), and Jerk (Jerk).

• Reverse rotation permission selection (Bit 5)

Setting value	Description
0: Allow	Allows reverse rotation. Performs a deceleration stop once, then starts operation to the changed direction once the deceleration stop is completed.
1: Do not allow	Does not allow reverse rotation. When switched, "Overrun Error (error code: 1A7EH)" occurs and a deceleration stop is performed.

■Necessary objects

When using MC_MoveAbsolute (Absolute Value Positioning), set the following slave objects for the axis.

• Target position (607AH)

If the slave object is not set, "Necessary Slave Object Unset (error code: 1AA8H)" occurs and the axis will not start.

For details of slave object settings, refer to the following.

MELSEC iQ-R Motion Module User's Manual (Application)

Program example

In the following program example, the single axis absolute value positioning start (bMoveAbsolute) is set to TRUE and absolute positioning control is performed for Axis 1 (Axis0001) using the following settings.

• Operation

• Settings

Item	Setting value
Target position	80000.0
Velocity	50000.0
Acceleration	100000.0
Deceleration	100000.0
Jerk	0.0

■Axis

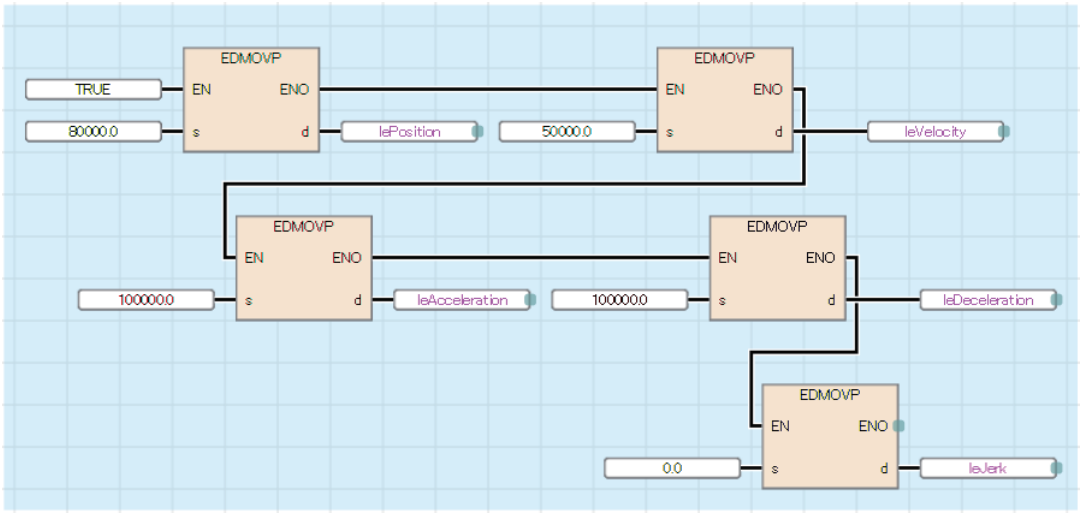
Axis No.	Label name	Data type	Comment
1	Axis0001	AXIS_REF	Axis 1

■Labels used

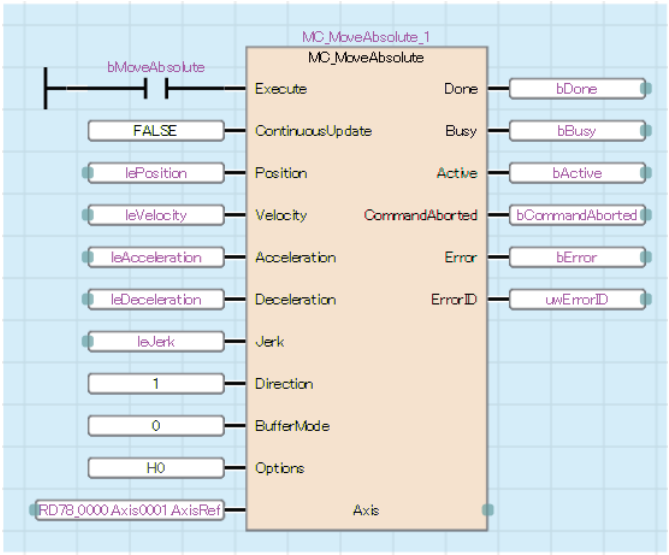
Label name	Data type	Comment
MC_MoveAbsolute_1	MC_MoveAbsolute	Absolute Value Positioning FB
bMoveAbsolute	Bit	Single axis absolute value positioning start
lePosition	Double-precision real number	Target position
leVelocity	Double-precision real number	Velocity
leAcceleration	Double-precision real number	Acceleration
leDeceleration	Double-precision real number	Deceleration
leJerk	Double-precision real number	Jerk
bDone	Bit	Execution completion
bBusy	Bit	Executing
bActive	Bit	Controlling
bCommandAborted	Bit	Abortion of execution
bError	Bit	Error
uwErrorID	Word [unsigned]/bit string [16-bit]	Error code

■FBD/LD program (CPU module side)

• Data setting for single axis positioning control



• Absolute Value Positioning



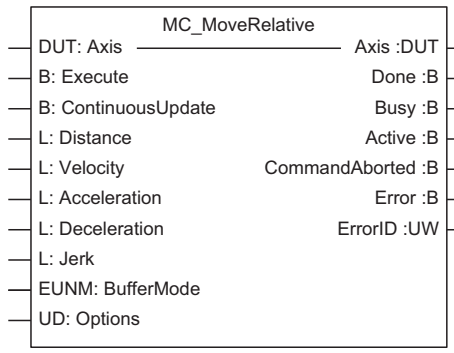
■ST program (Motion module side)

```
//-----Data setting for single axis positioning control-----
lePosition:= 80000.0;
leVelocity:= 50000.0;
leAcceleration:= 100000.0;
leDeceleration:= 100000.0;
leJerk:= 0.0;

//-----Absolute Value Positioning-----
MC_MoveAbsolute_1(
   Axis:= Axis0001.AxisRef ,
   Execute:= bMoveAbsolute ,
   ContinuousUpdate:= FALSE ,
   Position:= lePosition ,
   Velocity:= leVelocity ,
   Acceleration:= leAcceleration ,
   Deceleration:= leDeceleration ,
   Jerk:= leJerk ,
   Direction:= MC_DIRECTION__mcPositiveDirection ,
   BufferMode:= MC_BUFFER_MODE__mcAborting ,
   Options:= H00000000  ,
   Done=> bDone ,
   Busy=> bBusy ,
   Active=> bActive ,
   CommandAborted=> bCommandAborted ,
   Error=> bError ,
   ErrorID=> uwErrorID
);
```

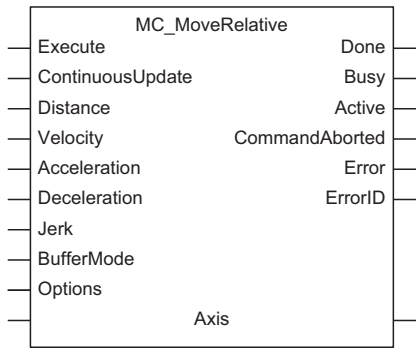
MC_MoveRelative

Executes positioning after setting the movement amount of the relative position.

Ladder



FBD/LD



ST

```
MC_MoveRelative(
   Axis:= ?AXIS_REF? ,
   Execute:= ?BOOL? ,
   ContinuousUpdate:= ?BOOL? ,
   Distance:= ?LREAL? ,
   Velocity:= ?LREAL? ,
   Acceleration:= ?LREAL? ,
   Deceleration:= ?LREAL? ,
   Jerk:= ?LREAL? ,
   BufferMode:= ?INT? ,
   Options:= ?DWORD? ,
   Done=> ?BOOL? ,
   Busy=> ?BOOL? ,
   Active=> ?BOOL? ,
   CommandAborted=> ?BOOL? ,
   Error=> ?BOOL? ,
   ErrorID=> ?WORD?
);
```

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Relative Value Positioning	64	8	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
Axis	Axis information	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Execute command	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MC_MoveRelative (Relative Value Positioning).
ContinuousUpdate	Continuous update	BOOL	At start	TRUE, FALSE	FALSE	This variable sets whether to enable or disable continuous change of Movement amount (Distance), Velocity (Velocity), Acceleration (Acceleration), and Deceleration (Deceleration). • FALSE: Disable • TRUE: Enable
Distance	Movement amount	LREAL	At start/ Retrigger possible/ Continuous update possible	-10000000000.0 to 10000000000.0	0.0	This variable sets the relative position from the current position at start to the end point. For details, refer to the following. ☞ Page 214 Movement amount (Distance)
Velocity	Velocity	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0, 0.0001 to 2500000000.0	0.0	This variable sets the velocity. For details, refer to the following. ☞ Page 215 Velocity (Velocity)
Acceleration	Acceleration	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the acceleration. For details, refer to the following. ☞ Page 215 Acceleration (Acceleration)

Input variable	Name	Data type	Input	Setting range	Default value	Description
Deceleration	Deceleration	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the deceleration. For details, refer to the following. ☞ Page 215 Deceleration (Deceleration)
Jerk	Jerk	LREAL	At start	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the jerk. For details, refer to the following. ☞ Page 215 Jerk (Jerk)
BufferMode	Buffer mode	INT (MC_BUFFER_MODE)	At start	0 to 5	0	This variable sets the buffer mode. • 0: Aborting (mcAborting) • 1: Buffered (mcBuffered) • 2: BlendingLow (mcBlendingLow) • 3: BlendingPrevious (mcBlendingPrevious) • 4: BlendingNext (mcBlendingNext) • 5: BlendingHigh (mcBlendingHigh) For details, refer to the following. ☞ Page 215 Buffer mode (BufferMode)
Options	Options	DWORD(HEX)	At start	00000000H to 00000021H	00000000H	This variable sets the function options for MC_MoveRelative (Relative Value Positioning). For details, refer to the following. ☞ Page 216 Options (Options)

■Output variables

Output variable	Name	Data type	Default value	Description
Done	Execution completion	BOOL	FALSE	This variable becomes TRUE when the relative position is reached.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MC_MoveRelative (Relative Value Positioning) is executed. This variable becomes FALSE after reaching the relative position.
Active	Controlling	BOOL	FALSE	This variable becomes TRUE while MC_MoveRelative (Relative Value Positioning) is controlling the axis. This variable becomes FALSE after reaching the relative position.
CommandAborted	Abortion of execution	BOOL	FALSE	This variable becomes TRUE when execution of MC_MoveRelative (Relative Value Positioning) is aborted. This variable becomes FALSE when Execute command (Execute) becomes FALSE.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. 📖MELSEC iQ-R Motion Module User's Manual (Application)

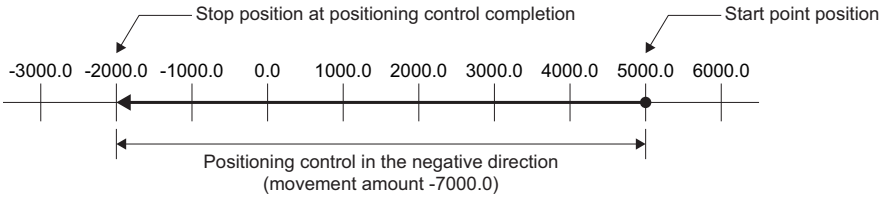
Processing details

- Sets Movement amount (Distance), Velocity (Velocity), Acceleration (Acceleration), Deceleration (Deceleration), Jerk (Jerk), and Buffer mode (BufferMode), then executes positioning from the current position at start (start point position) using the movement amount set in Movement amount (Distance). The movement direction is determined by the sign of the movement amount. Axis status (AxisName.Md.AxisStatus) becomes "5: During positioning operation (DiscreteMotion)".

Ex.

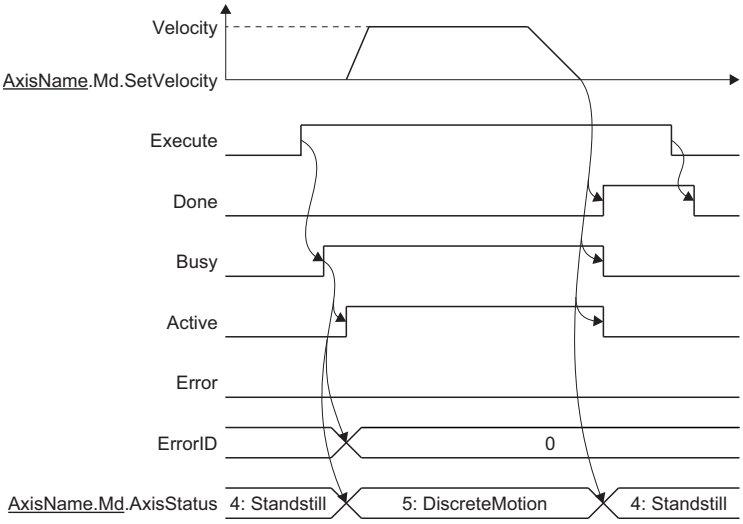
When the start point position (current stop position) is "5000.0" and the movement amount is set to "-7000.0"

· Positioning is performed to "-2000.0".



■Timing chart

- When the operation completes normally



- When an error occurs

  For the timing chart when an error occurs, refer to the following.

  ☞ Page 11 Basic operation of Execute command (Execute) type Motion control FBs

■Movement amount (Distance)

This variable sets the movement amount from the current position at start to the end point.

The movement direction is determined by the sign (+/-) of the movement amount.

Movement amount	Description
If positive (+)	Moves in the positive direction (address increasing direction).
If negative (-)	Moves in the negative direction (address decreasing direction).

When the movement amount is "0", the axis is not operated but Execution completion (Done) becomes TRUE.

If "current value + movement amount" overflows or underflows the software stroke limit upper limit value or software stroke limit lower limit value when the software stroke limit is valid, "Out of Target Position Range (error code: 1A05H)" occurs and the FB will not start.

■Velocity (Velocity)

Sets the commanded velocity used in MC_MoveRelative (Relative Value Positioning).

Setting value
0.0, 0.0001 to 2500000000.0[*1][*2]

*1 To perform floating-point operation, the lower limit value of the commanded velocity is limited by the following restrictions.
If the velocity that is converted from the target velocity into the operation cycle is less than "0.00001", "Out of Calculation Cycle Converted Speed Range (error code: 1AE4H)" occurs. (If the velocity is changed, "Out of Calculation Cycle Converted Speed Range Warning (warning code: 0D2FH)" occurs instead). To improve the precision of floating-point operation, change Position command unit (AxisName.Pr.Unit_Position) and Speed command unit (AxisName.Pr.Unit_Velocity) and perform settings so that the velocity converted into the operation cycle is not less than "0.00001".

*2 When a Motion control FB with a set velocity of "0.0" performs a multiple start, the set velocity of the previous Motion control FB is used.

■Acceleration (Acceleration)

Sets the acceleration used in MC_MoveRelative (Relative Value Positioning).

The setting range differs depending on the setting for Acceleration/deceleration method setting (Options (Options): Bit 0 to 2).

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000[*1], positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	0.000000[*1], positive numbers from 0.000001 to 8400.0[s]

*1 The operation performed at start varies depending on the setting of Operation selection at start acceleration/deceleration 0 (AxisName.Pr.AccelerationZeroBehavior). When changing the acceleration or deceleration, the changes are not accepted.

■Deceleration (Deceleration)

Sets the deceleration used in MC_MoveRelative (Relative Value Positioning).

Specifies the setting range if Acceleration/deceleration method setting (Options (Options): Bit 0 to 2) is set to "0: Acceleration/deceleration specification method (mcAccDec)". Deceleration (Deceleration) is not used if the acceleration/deceleration method is set to "1: Acceleration/deceleration time-fixed method (mcFixedTime)".

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000[*1], positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	Not used

*1 The operation performed at start varies depending on the setting of Operation selection at start acceleration/deceleration 0 (AxisName.Pr.AccelerationZeroBehavior). When changing the acceleration or deceleration, the changes are not accepted.

■Jerk (Jerk)

Sets the jerk used in MC_MoveRelative (Relative Value Positioning).

Specifies the setting range if Acceleration/deceleration method setting (Options (Options): Bit 0 to 2) is set to "0: Acceleration/deceleration specification method (mcAccDec)". Jerk (Jerk) is not used if the acceleration/deceleration method is set to "1: Acceleration/deceleration time-fixed method (mcFixedTime)".

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000, positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	Not used

■Buffer mode (BufferMode)

Sets the operation used to perform multiple start (buffer mode).

The following buffer modes are settable in MC_MoveRelative (Relative Value Positioning).

Setting value	Description
0: Aborting (mcAborting)	Aborts (cancels) the under-control FB and executes the next FB immediately.
1: Buffered (mcBuffered)	Buffers the next FB on the under-control FB. If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after completion of the under-control FB.
2: BlendingLow (mcBlendingLow)	Buffers the next FB on the under-control FB.[*1] If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. When the under-control FB reaches the target position, the lower target velocity between the under-control FB and the buffering FB is used as the switching speed.

Setting value	Description
3: BlendingPrevious (mcBlendingPrevious)	Buffers the next FB on the under-control FB.*1 If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. When the under-control FB reaches the target position, the target velocity of the under-control FB is used as the switching speed.
4: BlendingNext (mcBlendingNext)	Buffers the next FB on the under-control FB.*1 If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. When the under-control FB reaches the target position, the target velocity of the buffering FB is used as the switching speed.
5: BlendingHigh (mcBlendingHigh)	Buffers the next FB on the under-control FB.*1 If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. When the under-control FB reaches the target position, the higher target velocity between the under-control FB and the buffering FB is used as the switching speed.

*1 Stops are not performed between the under-control FB and the buffering FB.

Point

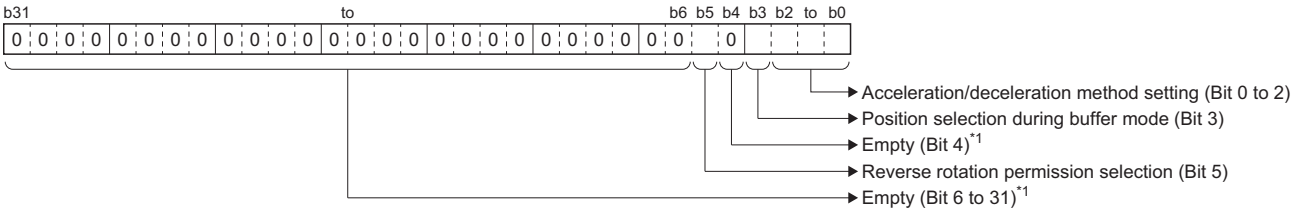
For details of multiple start (buffer mode), refer to the following.

MELSEC iQ-R Motion Module User's Manual (Application)

■Options (Options)

Sets the function options used for MC_MoveRelative (Relative Value Positioning) by specifying the bit.

Contents that are set by specifying the bit are shown below.



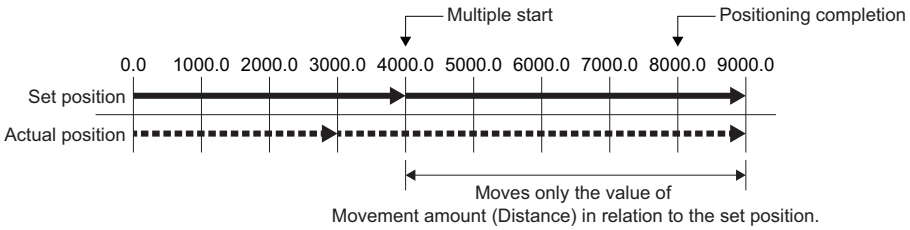
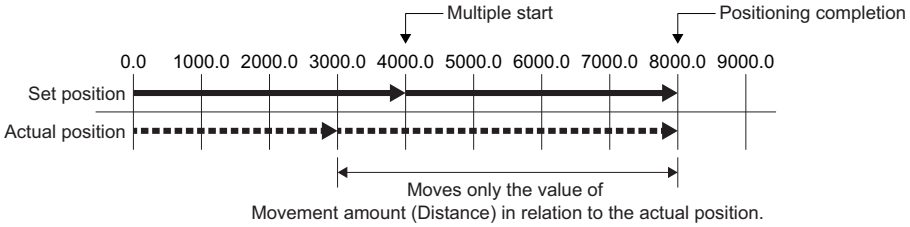
*1 Set empty areas to "0". When a value other than "0" is set, "Out of Options Range (error code: 1A4EH)" occurs.

Bit	Name	Description
0 to 2	Acceleration/deceleration method setting	Sets the acceleration/deceleration method used to perform control. • 0: Acceleration/deceleration specification method (mcAccDec) • 1: Acceleration/deceleration time-fixed method (mcFixedTime)
3	Position selection during buffer mode	Sets the position when executing relative positioning control by multiple start. • 0: Set position • 1: Actual position *: This bit is valid when setting "0: Aborting (mcAborting)" in Buffer mode (BufferMode).
5	Reverse rotation permission selection	Sets whether to allow reverse rotation when the operation direction of the FB being executed and the buffering FB are different. • 0: Allow • 1: Do not allow *: This bit is valid when setting "0: Aborting (mcAborting)" in Buffer mode (BufferMode) and when changing the target position.

• Acceleration/deceleration method setting (Bit 0 to 2)

Setting value	Description
0: Acceleration/deceleration specification method (mcAccDec)	An acceleration/deceleration method using Acceleration (Acceleration), Deceleration (Deceleration), and Jerk (Jerk) set in MC_MoveRelative (Relative Value Positioning).
1: Acceleration/deceleration time-fixed method (mcFixedTime)	An acceleration/deceleration method using the acceleration/deceleration time specified in MC_MoveRelative (Relative Value Positioning) regardless of the velocity. The acceleration/deceleration time does not use Acceleration (Acceleration), Deceleration (Deceleration), and Jerk (Jerk).

• Position selection during buffer mode (Bit 3)

Setting value	Description
0: Set position	Performs relative positioning control from the set position. <Example> When multiple start is performed with Movement amount (Distance) set to "5000.0" and Options (Options) set to "00000000H (Bit 3 is 0: Set position)" 
1: Actual position	Performs relative positioning control from the actual position. <Example> When multiple start is performed with Movement amount (Distance) set to "5000.0" and Options (Options) set to "00000008H (Bit 3 is 1: Actual position)". 

• Reverse rotation permission selection (Bit 5)

Setting value	Description
0: Allow	Allows reverse rotation. Performs a deceleration stop once, then starts operation to the changed direction once the deceleration stop is completed.
1: Do not allow	Does not allow reverse rotation. When switched, "Overrun Error (error code: 1A7EH)" occurs and a deceleration stop is performed.

■Necessary objects

When using MC_MoveRelative (Relative Value Positioning), set the following slave objects for the axis.

• Target position (607AH)

If the slave object is not set, "Necessary Slave Object Unset (error code: 1AA8H)" occurs and the axis will not start.

For details of slave object settings, refer to the following.

MELSEC iQ-R Motion Module User's Manual (Application)

Precautions

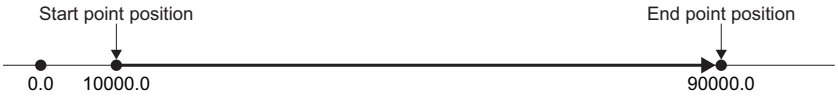
• The current position where MC_MoveRelative (Relative Value Positioning) was started for the first time is used as the start address even if a restart/continuous update is performed.

• Since floating point type processing is used, if relative positioning control is repeatedly executed, the set movement amount may not be reached due to a calculation error.

Program example

In the following program example, the single axis relative value positioning start (bMoveRelative) is set to TRUE and relative positioning control is performed for Axis 1 (Axis0001) using the following settings.

- Operation



- Settings

Item	Setting value
Target position	80000.0
Velocity	50000.0
Acceleration	100000.0
Deceleration	100000.0
Jerk	0.0

■Axis

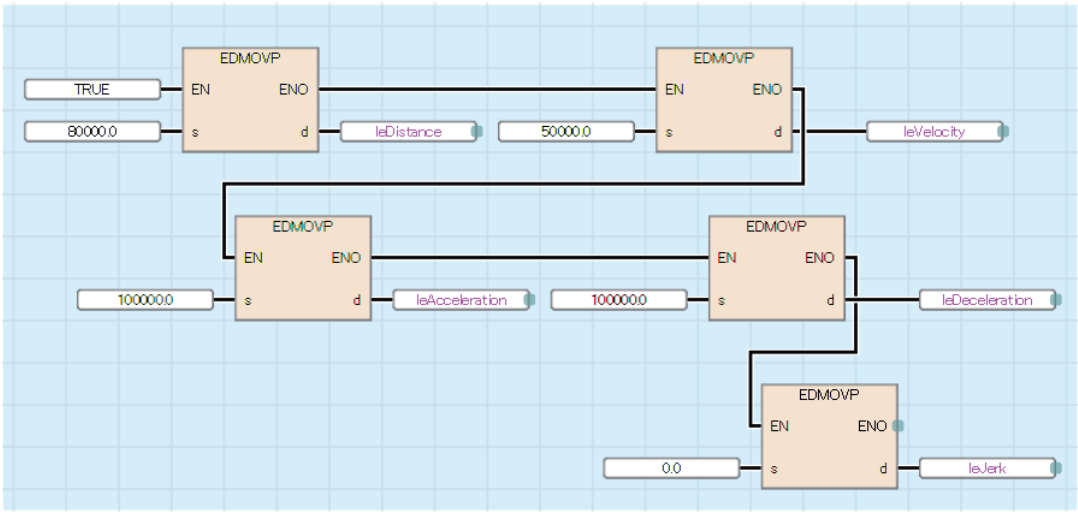
Axis No.	Label name	Data type	Comment
1	Axis0001	AXIS_REF	Axis 1

■Labels used

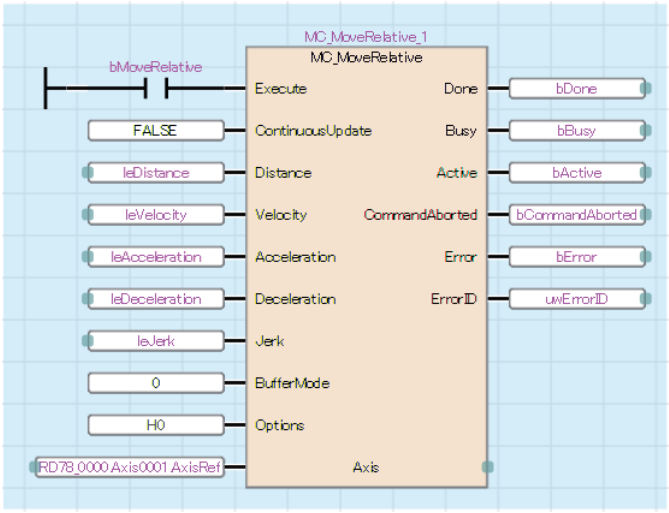
Label name	Data type	Comment
MC_MoveRelative_1	MC_MoveRelative	Relative Value Positioning FB
bMoveRelative	Bit	Single axis relative value positioning start
leDistance	Double-precision real number	Movement amount
leVelocity	Double-precision real number	Velocity
leAcceleration	Double-precision real number	Acceleration
leDeceleration	Double-precision real number	Deceleration
leJerk	Double-precision real number	Jerk
bDone	Bit	Execution completion
bBusy	Bit	Executing
bActive	Bit	Controlling
bCommandAborted	Bit	Abortion of execution
bError	Bit	Error
uwErrorID	Word [unsigned]/bit string [16-bit]	Error code

■FBD/LD program (CPU module side)

- Data setting for single axis positioning control



• Relative Value Positioning



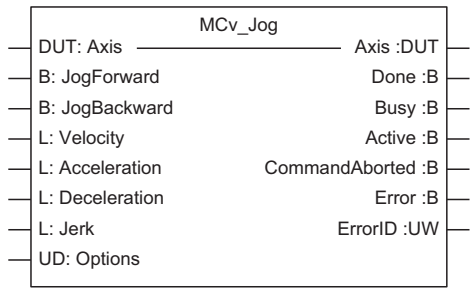
■ST program (Motion module side)

```
//-----Data setting for single axis positioning control-----
leDistance:= 80000.0;
leVelocity:= 50000.0;
leAcceleration:= 100000.0;
leDeceleration:= 100000.0;
leJerk:= 0.0;

//-----Relative Value Positioning-----
MC_MoveRelative_1(
   Axis:= Axis0001.AxisRef ,
   Execute:= bMoveRelative ,
   ContinuousUpdate:= FALSE ,
   Distance:= leDistance ,
   Velocity:= leVelocity ,
   Acceleration:= leAcceleration ,
   Deceleration:= leDeceleration ,
   Jerk:= leJerk ,
   BufferMode:= MC_BUFFER_MODE__mcAborting ,
   Options:= H00000000,
   Done=> bDone ,
   Busy=> bBusy ,
   Active=> bActive ,
   CommandAborted=> bCommandAborted ,
   Error=> bError ,
   ErrorID=> uwErrorID
);
```

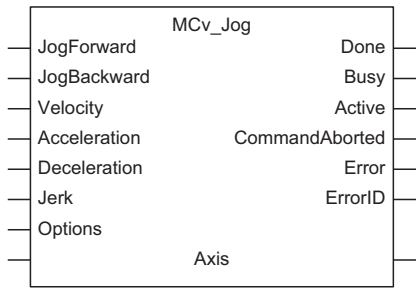
MCv_Jog

Executes JOG operation according to the target velocity.

Ladder



FBD/LD



ST

```
MCv_Jog(
   Axis:= ?AXIS_REF? ,
   JogForward:= ?BOOL? ,
   JogBackward:= ?BOOL? ,
   Velocity:= ?LREAL? ,
   Acceleration:= ?LREAL? ,
   Deceleration:= ?LREAL? ,
   Jerk:= ?LREAL? ,
   Options:= ?DWORD? ,
   Done=> ?BOOL? ,
   Busy=> ?BOOL? ,
   Active=> ?BOOL? ,
   CommandAborted=> ?BOOL? ,
   Error=> ?BOOL? ,
   ErrorID=> ?WORD?
);
```

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
JOG	52	8	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
Axis	Axis information	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
JogForward	Positive rotation JOG command	BOOL	Always	TRUE, FALSE	FALSE	When this variable is TRUE, executes MCv_Jog (JOG) in the positive direction. For details, refer to the following. ☞ Page 222 Positive rotation JOG command (JogForward)/Reverse rotation JOG command (JogBackward)
JogBackward	Reverse rotation JOG command	BOOL	Always	TRUE, FALSE	FALSE	When this variable is TRUE, executes MCv_Jog (JOG) in the reverse direction. For details, refer to the following. ☞ Page 222 Positive rotation JOG command (JogForward)/Reverse rotation JOG command (JogBackward)
Velocity	Velocity	LREAL	At start	0.0, 0.0001 to 2500000000.0	0.0	This variable sets the commanded velocity. For details, refer to the following. ☞ Page 224 Velocity (Velocity)
Acceleration	Acceleration	LREAL	At start	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the acceleration. For details, refer to the following. ☞ Page 224 Acceleration (Acceleration)
Deceleration	Deceleration	LREAL	At start	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the deceleration. For details, refer to the following. ☞ Page 224 Deceleration (Deceleration)
Jerk	Jerk	LREAL	At start	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the jerk. For details, refer to the following. ☞ Page 224 Jerk (Jerk)

Input variable	Name	Data type	Input	Setting range	Default value	Description
Options	Options	DWORD(HEX)	At start	00000000H to 00000001H	00000000H	This variable sets the function options for MCv_Jog (JOG) by specifying the bit. For details, refer to the following. ☞ Page 225 Options (Options)

■Output variables

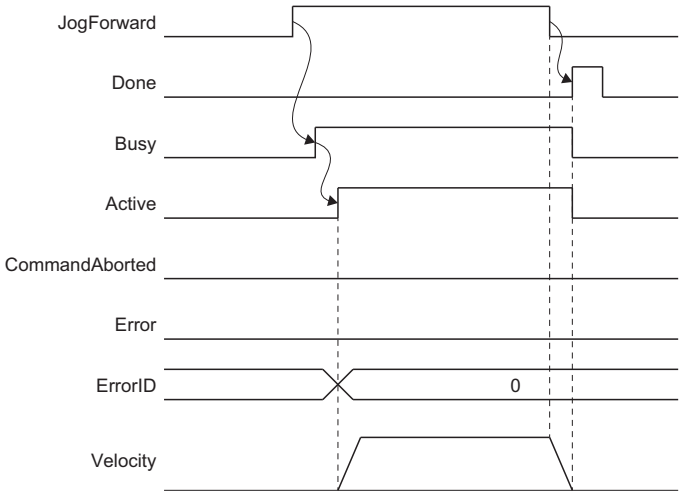
Output variable	Name	Data type	Default value	Description
Done	Execution completion	BOOL	FALSE	This variable becomes TRUE for one scan only when a deceleration stop completes by the JOG command turning OFF.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MCv_Jog (JOG) is executed.
Active	Controlling	BOOL	FALSE	This variable becomes TRUE while MCv_Jog (JOG) is controlling the axis.
CommandAborted	Abortion of execution	BOOL	FALSE	This variable becomes TRUE when execution of MCv_Jog (JOG) is aborted.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. 📖MELSEC iQ-R Motion Module User's Manual (Application)

Processing details

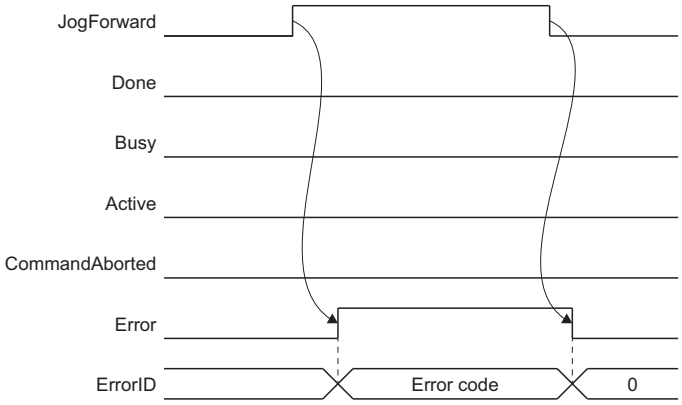
- The target axis moves in the specified direction when Positive rotation JOG command (JogForward) or Reverse rotation JOG command (JogBackward) is set to TRUE.
- Axis status (AxisName.Md.AxisStatus) is "6: During continuous operation (ContinuousMotion)" during JOG operation.
- A deceleration stop is performed when Positive rotation JOG command (JogForward) or Reverse rotation JOG command (JogBackward) is set to FALSE.
- Axis status (AxisName.Md.AxisStatus) changes to "4: Standby (Standstill)" at deceleration stop completion.
- If Error (Error) becomes TRUE during deceleration by Positive rotation JOG command (JogForward) or Reverse rotation JOG command (JogBackward) becoming FALSE, Error (Error) remains TRUE until Positive rotation JOG command (JogForward) or Reverse rotation JOG command (JogBackward) is set to TRUE.
- When another operation FB is started during JOG operation, the operation is performed based on the setting of Buffer mode (BufferMode) of the started operation FB.
- When JOG operation is started during another operation FB, the start request is ignored and "Start during Operation Warning (warning code: 0D01H)" occurs. Start JOG operation when Axis status (AxisName.Md.AxisStatus) is "4: Standby (Standstill)".
- To change the velocity during JOG operation, use the override function to perform the velocity change.
  For details of override function, refer to the following.
  📖MELSEC iQ-R Motion Module User's Manual (Application)

■Timing chart

• When the operation completes normally



• When an error occurs

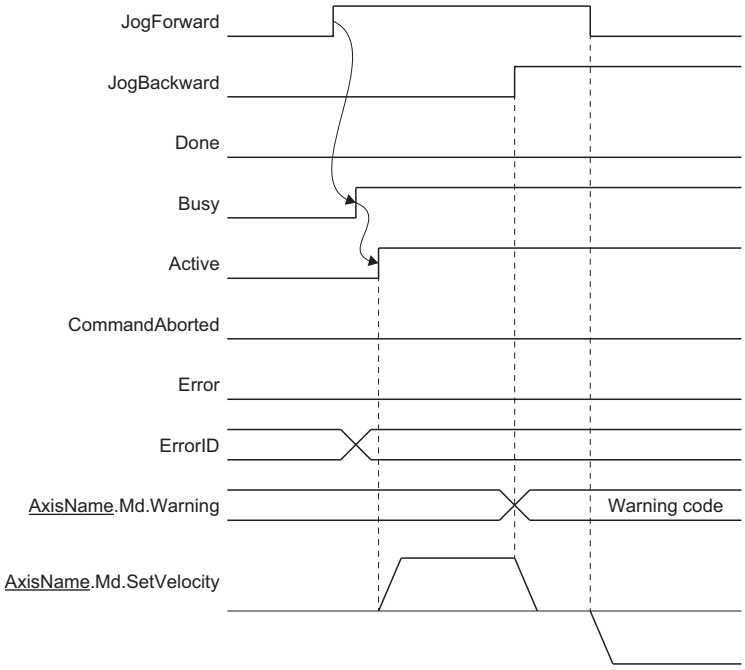


■Positive rotation JOG command (JogForward)/Reverse rotation JOG command (JogBackward)

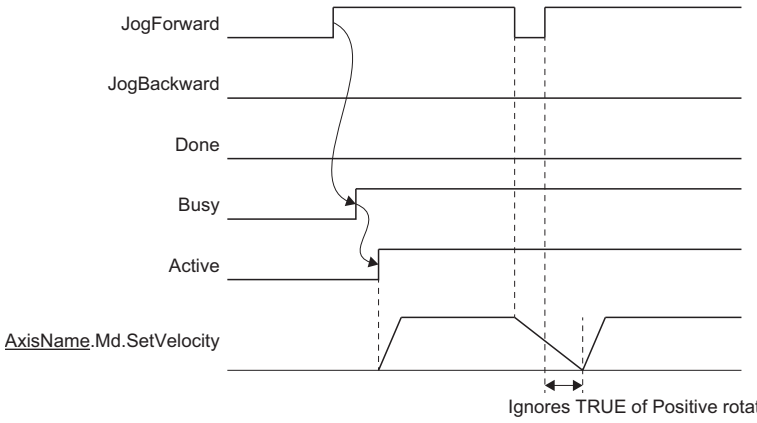
• Setting Positive rotation JOG command (JogForward) to TRUE starts positive rotation JOG operation, and setting Reverse rotation JOG command (JogBackward) to TRUE starts reverse rotation JOG operation, following which the target axis moves in the specified direction.

• If Positive rotation JOG command (JogForward) and Reverse rotation JOG command (JogBackward) are set to TRUE simultaneously, "JOG Command Input at Both Directions Warning (warning code: 0D02H)" occurs and JOG operation will not start. To perform JOG operation, first set the JOG commands for both directions to FALSE and wait for Abortion of execution (CommandAborted) to become FALSE, then set either Positive rotation JOG command (JogForward) or Reverse rotation JOG command (JogBackward) to TRUE.

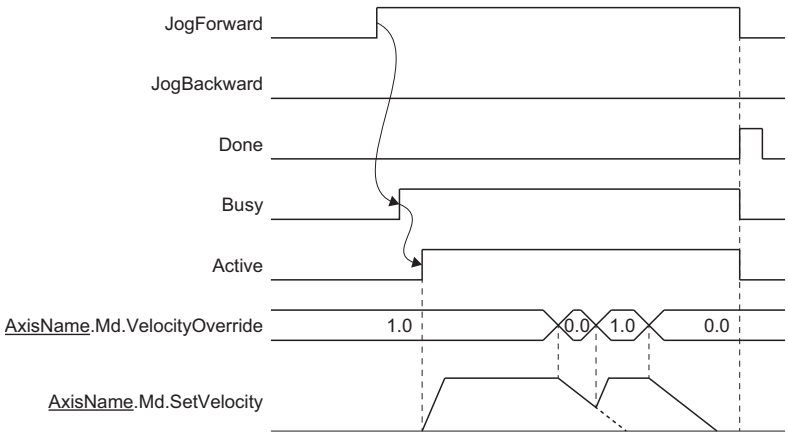
- During JOG operation, if the other JOG command (Positive rotation JOG command (JogForward) or Reverse rotation JOG command (JogBackward)) of MCv_JOG (JOG) becomes TRUE, "JOG Command Input at Both Directions Warning (warning code: 0D02H)" occurs and a deceleration stop is performed from the point where said TRUE was detected. When performing JOG operation, only set either Positive rotation JOG command (JogForward) or Reverse rotation JOG command (JogBackward) to TRUE.



- During the deceleration stop by Positive rotation JOG command (JogForward) or Reverse rotation JOG command (JogBackward) becoming FALSE, re-acceleration will not be performed even if the JOG command is set to TRUE. The acceleration will be performed after the deceleration stop is completed.



- To repeatedly perform deceleration stop and acceleration while moving by JOG operation, use the override function to perform the velocity change.



■Velocity (Velocity)

Sets the commanded velocity used in MCv_Jog (JOG).

The settings are imported when JOG operation is started by Positive rotation JOG command (JogForward) or Reverse rotation JOG command (JogBackward) becoming TRUE. Changes made during JOG operation are not accepted.

Setting range
0.0, 0.0001 to 2500000000.0[*1][*2]

*1 To perform floating-point operation, the lower limit value of the commanded velocity is limited by the following restrictions.
If the velocity that is converted from the target velocity into the operation cycle is less than "0.00001", "Out of Calculation Cycle Converted Speed Range (error code: 1AE4H)" occurs. (If the velocity is changed, "Out of Calculation Cycle Converted Speed Range Warning (warning code: 0D2FH)" occurs instead). To improve the precision of floating-point operation, change Position command unit (AxisName.Pr.Unit_Position) and Speed command unit (AxisName.Pr.Unit_Velocity) and perform settings so that the velocity converted into the operation cycle is not less than "0.00001".

*2 When a Motion control FB with a set velocity of "0.0" performs a multiple start, the set velocity of the previous Motion control FB is used.

■Acceleration (Acceleration)

Sets the acceleration used in MCv_Jog (JOG).

The settings are imported when JOG operation is started by the positive rotation JOG command (JogForward) or the reverse rotation JOG command (JogBackward) becoming TRUE. Changes made during JOG operation are not accepted.

The setting range differs depending on the setting for Acceleration/deceleration method setting (Options (Options): Bit 0 to 2).

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000[*1], positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	0.000000[*1], positive numbers from 0.000001 to 8400.0[s]

*1 The operation performed at start varies depending on the setting of Operation selection at start acceleration/deceleration 0 (AxisName.Pr.AccelerationZeroBehavior). When changing the acceleration or deceleration, the changes are not accepted.

■Deceleration (Deceleration)

Sets the deceleration used in MCv_Jog (JOG).

The settings are imported when JOG operation is started by Positive rotation JOG command (JogForward) or Reverse rotation JOG command (JogBackward) becoming TRUE. Changes made during JOG operation are not accepted.

Specifies the setting range if Acceleration/deceleration method setting (Options (Options): Bit 0 to 2) is set to "0: Acceleration/deceleration specification method (mcAccDec)". Deceleration (Deceleration) is not used if the acceleration/deceleration method is set to "1: Acceleration/deceleration time-fixed method (mcFixedTime)".

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000[*1], positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	Not used

*1 The operation performed at start varies depending on the setting of Operation selection at start acceleration/deceleration 0 (AxisName.Pr.AccelerationZeroBehavior). When changing the acceleration or deceleration, the changes are not accepted.

■Jerk (Jerk)

Sets the jerk used in MCv_Jog (JOG).

The settings are imported when JOG operation is started by Positive rotation JOG command (JogForward) or Reverse rotation JOG command (JogBackward) becoming TRUE. Changes made during JOG operation are not accepted.

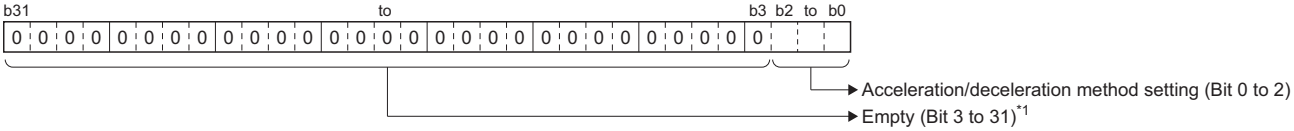
Specifies the setting range if Acceleration/deceleration method setting (Options (Options): Bit 0 to 2) is set to "0: Acceleration/deceleration specification method (mcAccDec)". Jerk (Jerk) is not used if the acceleration/deceleration method is set to "1: Acceleration/deceleration time-fixed method (mcFixedTime)".

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000, positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	Not used

■Options (Options)

Sets the function options used for MCv_Jog (JOG) by specifying the bit.

Contents that are set by specifying the bit are shown below.



*1 Set empty areas to "0". When a value other than "0" is set, "Out of Options Range (error code: 1A4EH)" occurs.

Bit	Name	Description
0 to 2	Acceleration/deceleration method setting	Sets the acceleration/deceleration method used to perform control. • 0: Acceleration/deceleration specification method (mcAccDec) • 1: Acceleration/deceleration time-fixed method (mcFixedTime)

• Acceleration/deceleration method setting (Bit 0 to 2)

Setting value	Description
0: Acceleration/deceleration specification method (mcAccDec)	An acceleration/deceleration method using Acceleration (Acceleration), Deceleration (Deceleration), and Jerk (Jerk) set in MCv_Jog (JOG).
1: Acceleration/deceleration time-fixed method (mcFixedTime)	An acceleration/deceleration method using the acceleration/deceleration time specified in MCv_Jog (JOG) regardless of the velocity. The acceleration/deceleration time does not use Acceleration (Acceleration), Deceleration (Deceleration), and Jerk (Jerk).

■Necessary objects

When using MCv_Jog (JOG), set the following slave objects for the axis.

• Target position (607AH)

If the slave object is not set, "Necessary Slave Object Unset (error code: 1AA8H)" occurs and the axis will not start.

For details of slave object settings, refer to the following.

📖MELSEC iQ-R Motion Module User's Manual (Application)

Precautions

- For safety, first set a small value in Velocity (Velocity) and check the movement, then gradually increase the value.
- Use the hardware stroke limit function when performing JOG operation near the upper limit or lower limit. If the hardware stroke limit function is not used, the workpiece may exceed the moving range and cause an accident.
- When the software stroke limit function is valid and Software Stroke Limit Override (AxisName.Cd.SwStrokeLimit_Override) is other than DISABLE (Check disabled) or ONLY_INSIDE (Check disabled only to inside the range direction), rewrites Software Stroke Limit Override (AxisName.Cd.SwStrokeLimit_Override) into ONLY_INSIDE (Check disabled only to inside the range direction) during MCv_Jog (JOG) execution, and rewrites it into a blank area at JOG operation completion.
- When hardware stroke limit function is valid and Hardware Stroke Limit Override (AxisName.Cd.HwStrokeLimit_Override) is other than DISABLE (Check disabled) or ONLY_INSIDE (Check disabled only to inside the range direction), rewrites Hardware Stroke Limit Override (AxisName.Cd.HwStrokeLimit_Override) into ONLY_INSIDE (Check disabled only to inside the range direction) during MCv_Jog (JOG) execution, and rewrites it into a blank area during JOG operation completion.
- Do not change Software Stroke Limit Override (AxisName.Cd.SwStrokeLimit_Override) and Hardware Stroke Limit Override (AxisName.Cd. HwStrokeLimit_Override) during MCv_Jog (JOG) execution.
- When multiple start is executed during JOG operation, the values of Software Stroke Limit Override (AxisName.Cd.SwStrokeLimit_Override) and Hardware Stroke Limit Override (AxisName.Cd. HwStrokeLimit_Override) at the multiple start are retained to the next FB in order to analyze the next FB.

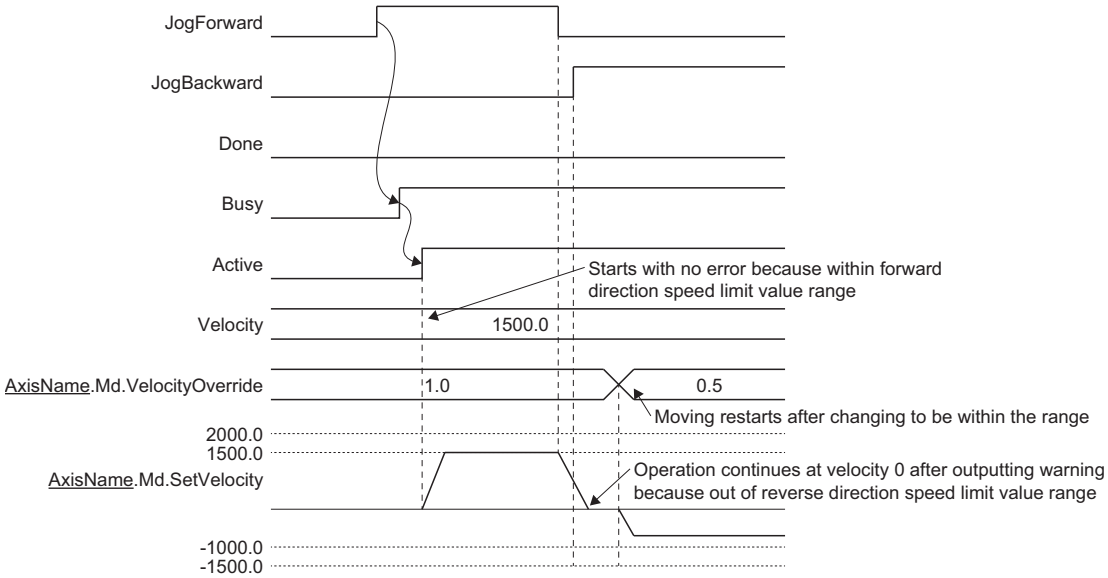
- When moving in the direction opposite that of the direction at start while in deceleration stop during JOG operation, "Speed Limit Value Over Warning on Direction Change (warning code: 0D20H)" occurs when the commanded velocity exceeds the velocity limit value. In addition, when the acceleration time exceeds 8400 seconds, "Acceleration Time Over Warning on Direction Change (warning code: 0D32H)" occurs. (The operation continues at velocity 0.) Movement starts after changing control to remove the cause.

Ex.

When the following values are set

· Positive Direction Speed Limit Value (AxisName.Pr.VelocityLimit_Positive): 2000.0

· Negative Direction Speed Limit Value (AxisName.Pr.VelocityLimit_Negative): 1000.0



Program example

In the following program example, the settings below are used to perform JOG operation for Axis 1 (Axis0001) by setting the JOG positive rotation command (bJogForward) to TRUE to perform positive JOG operation or by setting the JOG reverse rotation command (bJogBackward) to TRUE to perform reverse rotation JOG operation.

- Settings

Item	Setting value
Velocity	1500.0
Acceleration	2000.0
Deceleration	2000.0
Jerk	0.0

■Axis

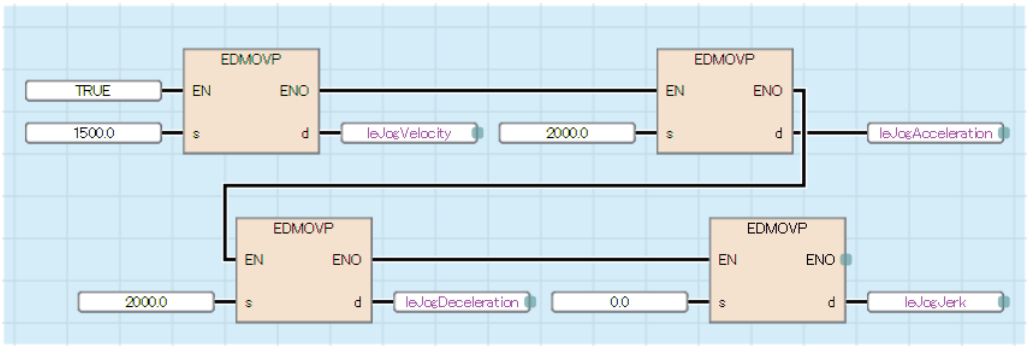
Axis No.	Label name	Data type	Comment
1	Axis0001	AXIS_REF	Axis 1

■Labels used

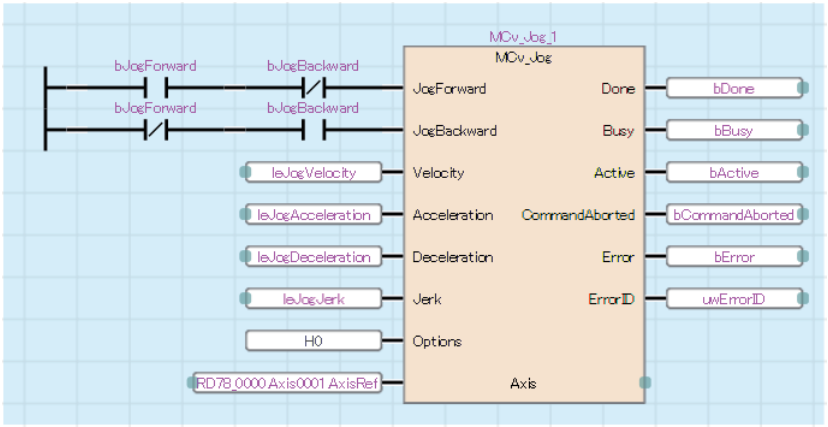
Label name	Data type	Comment
MCv_Jog_1	MCv_Jog	JOG FB
bJogForward	Bit	JOG positive rotation command
bJogBackward	Bit	JOG reverse rotation command
leJogVelocity	Double-precision real number	JOG velocity
leJogAcceleration	Double-precision real number	JOG acceleration
leJogDeceleration	Double-precision real number	JOG deceleration
leJogJerk	Double-precision real number	JOG jerk
bJogF	Bit	JOG positive rotation
bJogB	Bit	JOG reverse rotation
bDone	Bit	Execution completion
bBusy	Bit	Executing
bActive	Bit	Controlling
bCommandAborted	Bit	Abortion of execution
bError	Bit	Error
uwErrorID	Word [unsigned]/bit string [16-bit]	Error code

■FBD/LD program (CPU module side)

• Data setting for JOG operation



• Axis0001 JOG operation

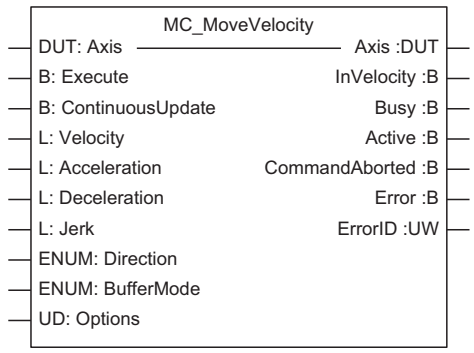
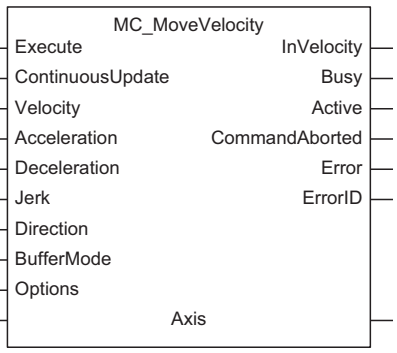


■ST program (Motion module side)

```
//-----Data setting for JOG operation-----
leJogVelocity:= 1500.0;
leJogAcceleration:= 2000.0;
leJogDeceleration:= 2000.0;
leJogJerk:= 0.0;

//-----Axis0001 JOG operation-----
bJogForward:= (bJogForward=TRUE) & (bJogBackward=FALSE);
bJogBackward:= (bJogForward=FALSE) & (bJogBackward=TRUE);

MCv_Jog_1(
   Axis:= Axis0001.AxisRef ,
   JogForward:= bJogForward ,
   JogBackward:= bJogBackward ,
   Velocity:= leJogVelocity ,
   Acceleration:= leJogAcceleration ,
   Deceleration:= leJogDeceleration ,
   Jerk:= leJogJerk ,
   Options:= H00000000 ,
   Done=> bDone ,
   Busy=> bBusy ,
   Active=> bActive ,
   CommandAborted=> bCommandAborted ,
   Error=> bError ,
   ErrorID=> uwErrorID
);
```

MC_MoveVelocity

Switches the driver to csv, then performs velocity control according to the specified velocity.

Ladder	FBD/LD	ST
		MC_MoveVelocity(Axis:= ?AXIS_REF? , Execute:= ?BOOL? , ContinuousUpdate:= ?BOOL? , Velocity:= ?LREAL? , Acceleration:= ?LREAL? , Deceleration:= ?LREAL? , Jerk:= ?LREAL? , Direction:= ?INT? , BufferMode:= ?INT? , Options:= ?DWORD? , InVelocity=> ?BOOL? , Busy=> ?BOOL? , Active=> ?BOOL? , CommandAborted=> ?BOOL? , Error=> ?BOOL? , ErrorID=> ?WORD?);

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Speed Control	56	8	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
Axis	Axis information	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Execute command	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MC_MoveVelocity (Speed Control).
ContinuousUpdate	Continuous update	BOOL	At start	TRUE, FALSE	FALSE	This variable sets whether to enable or disable continuous change of Velocity (Velocity), Acceleration (Acceleration), and Deceleration (Deceleration). • FALSE: Disable • TRUE: Enable
Velocity	Velocity	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0, ±0.0001 to ±2500000000.0	0.0	This variable sets the commanded velocity. When the velocity is negative, the axis moves in the negative direction. When "0.0" is set, the axis does not operate, but Axis status (AxisName.Md.AxisStatus) changes to "6: During continuous operation (ContinuousMotion)". For details, refer to the following. ☞ Page 231 Velocity (Velocity)
Acceleration	Acceleration	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the acceleration. For details, refer to the following. ☞ Page 232 Acceleration (Acceleration)

Input variable	Name	Data type	Input	Setting range	Default value	Description
Deceleration	Deceleration	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the deceleration. For details, refer to the following. ☞ Page 232 Deceleration (Deceleration)
Jerk	Jerk	LREAL	At start	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the jerk. For details, refer to the following. ☞ Page 232 Jerk (Jerk)
Direction	Direction selection	INT (MC_DIRECTION)	At start	1, 2	0	This variable sets the selected direction. • 1: Positive direction (mcPositiveDirection) • 2: Negative direction (mcNegativeDirection) *: If setting "2: Negative direction (mcNegativeDirection)" when Velocity (Velocity) is negative, the movement direction of the motor will be positive. *: When this setting is omitted, "Out of Direction Selection Range (error code: 1A37H)" occurs.
BufferMode	Buffer mode	INT (MC_BUFFER_MODE)	At start	0, 1	0	This variable sets the buffer mode. • 0: Aborting (mcAborting) • 1: Buffered (mcBuffered) For details, refer to the following. ☞ Page 232 Buffer mode (BufferMode)
Options	Options	DWORD(HEX)	At start	00000000H to 0002001H	00000000H	This variable sets the function options for MC_MoveVelocity (Speed Control) by specifying the bit. For details, refer to the following. ☞ Page 233 Options (Options)

■Output variables

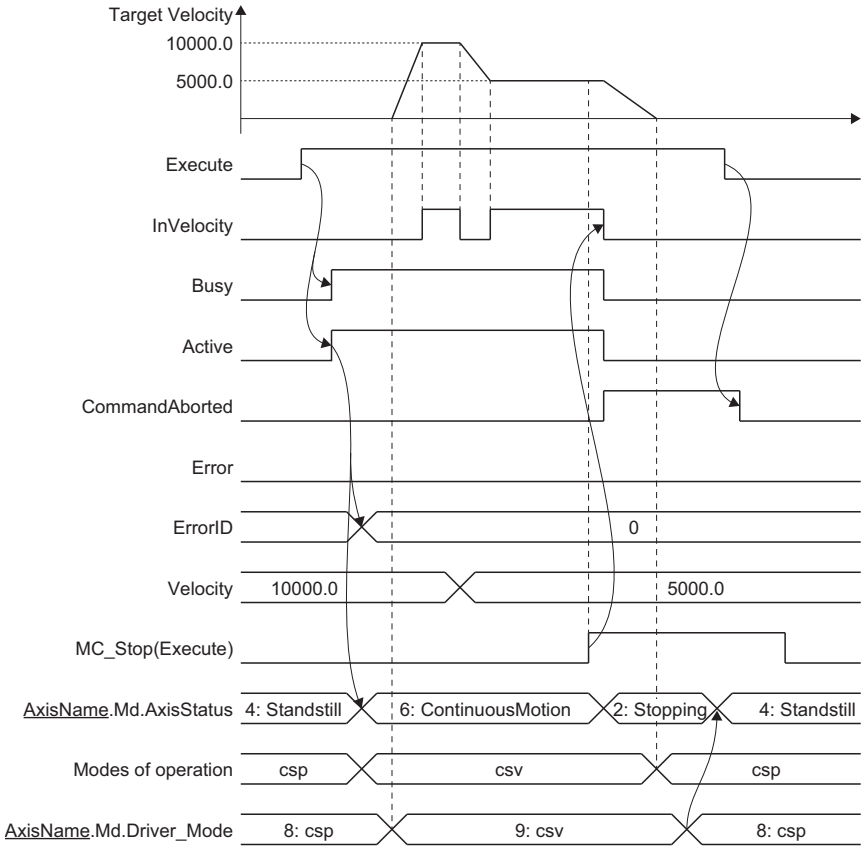
Output variable	Name	Data type	Default value	Description
InVelocity	Target speed reached	BOOL	FALSE	This variable becomes TRUE when the commanded velocity calculated by the Motion system reaches the target velocity. When the target velocity has been changed due to change at Continuous update (ContinuousUpdate) TRUE, this variable remains FALSE until reaching the post-change target velocity.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MC_MoveVelocity (Speed Control) is executed.
Active	Controlling	BOOL	FALSE	This variable becomes TRUE while MC_MoveVelocity (Speed Control) is controlling the axis.
CommandAborted	Abortion of execution	BOOL	FALSE	This variable becomes TRUE when execution of MC_MoveVelocity (Speed Control) is aborted. This variable becomes FALSE when Execute command (Execute) becomes FALSE.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. MELSEC iQ-R Motion Module User's Manual (Application)

Processing details

- MC_MoveVelocity (Speed Control) switches the driver control mode to csv and performs control. This function controls the commanded velocity based on the set Acceleration (Acceleration), Deceleration (Deceleration), and Jerk (Jerk). To finish MC_MoveVelocity (Speed Control), start MC_Stop (Forced Stop).

■Timing chart

- When the operation completes normally



- When an error occurs

  For the timing chart when an error occurs, refer to the following.

  ☞ Page 11 Basic operation of Execute command (Execute) type Motion control FBs

■Velocity (Velocity)

Sets the commanded velocity used in MC_MoveVelocity (Speed Control).

Setting range
0.0, ±0.0001 to ±2500000000.0[*1*2]

*1 To perform floating-point operation, the lower limit value of the commanded velocity is limited by the following restrictions. If the velocity that is converted from the target velocity into the operation cycle is less than "0.00001", "Out of Calculation Cycle Converted Speed Range (error code: 1AE4H)" occurs. (If the velocity is changed, "Out of Calculation Cycle Converted Speed Range Warning (warning code: 0D2FH)" occurs instead). To improve the precision of floating-point operation, change Position command unit (AxisName.Pr.Unit_Position) and Speed command unit (AxisName.Pr.Unit_Velocity) and perform settings so that the velocity converted into the operation cycle is not less than "0.00001".

*2 When a Motion control FB with a set velocity of "0.0" performs a multiple start, the set velocity of the previous Motion control FB is used.

■Acceleration (Acceleration)

Sets the acceleration used in MC_MoveVelocity (Speed Control).

The setting range differs depending on the setting for Acceleration/deceleration method setting (Options (Options): Bit 0 to 2).

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000[*1], positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	0.000000[*1], positive numbers from 0.000001 to 8400.0[s]

*1 The operation performed at start varies depending on the setting of Operation selection at start acceleration/deceleration 0 (AxisName.Pr.AccelerationZeroBehavior). When changing the acceleration or deceleration, the changes are not accepted.

■Deceleration (Deceleration)

Sets the deceleration used in MC_MoveVelocity (Speed Control).

Specifies the setting range if Acceleration/deceleration method setting (Options (Options): Bit 0 to 2) is set to "0: Acceleration/deceleration specification method (mcAccDec)". Deceleration (Deceleration) is not used if the acceleration/deceleration method is set to "1: Acceleration/deceleration time-fixed method (mcFixedTime)".

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000[*1], positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	Not used

*1 The operation performed at start varies depending on the setting of Operation selection at start acceleration/deceleration 0 (AxisName.Pr.AccelerationZeroBehavior). When changing the acceleration or deceleration, the changes are not accepted.

■Jerk (Jerk)

Sets the jerk used in MC_MoveVelocity (Speed Control).

Specifies the setting range if Acceleration/deceleration method setting (Options (Options): Bit 0 to 2) is set to "0: Acceleration/deceleration specification method (mcAccDec)". Jerk (Jerk) is not used if the acceleration/deceleration method is set to "1: Acceleration/deceleration time-fixed method (mcFixedTime)".

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000, positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	Not used

■Buffer mode (BufferMode)

Sets the operation used to perform multiple start (buffer mode).

The following buffer modes are settable in MC_MoveVelocity (Speed Control).

Setting value	Description
0: Aborting (mcAborting)	Aborts (cancels) the under-control FB and executes the next FB immediately.
1: Buffered (mcBuffered)	Buffers the next FB on the under-control FB. If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after completion of the under-control FB.

Point

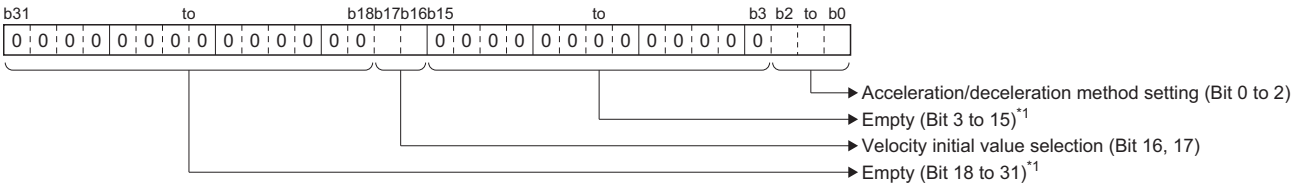
For details of multiple start (buffer mode), refer to the following.

MELSEC iQ-R Motion Module User's Manual (Application)

■Options (Options)

Sets the function options used for MC_MoveVelocity (Speed Control) by specifying the bit.

Contents that are set by specifying the bit are shown below.



*1 Set empty areas to "0". When a value other than "0" is set, "Out of Options Range (error code: 1A4EH)" occurs.

Bit	Name	Description
0 to 2	Acceleration/deceleration method setting	Sets the acceleration/deceleration method used to perform control. • 0: Acceleration/deceleration specification method (mcAccDec) • 1: Acceleration/deceleration time-fixed method (mcFixedTime)
16, 17	Velocity initial value selection	Sets the velocity initial value used when changing the control mode from csp (cyclic synchronous position mode) to csv (cyclic synchronous velocity mode). • 0: Set velocity • 1: Actual velocity • 2: Automatic selection

• Acceleration/deceleration method setting (Bit 0 to 2)

Setting value	Description
0: Acceleration/deceleration specification method (mcAccDec)	An acceleration/deceleration method using Acceleration (Acceleration), Deceleration (Deceleration), and Jerk (Jerk) set in MC_MoveAbsolute (Absolute Value Positioning).
1: Acceleration/deceleration time-fixed method (mcFixedTime)	An acceleration/deceleration method using the acceleration/deceleration time specified in MC_MoveAbsolute (Absolute Value Positioning) regardless of the velocity. The acceleration/deceleration time does not use Acceleration (Acceleration), Deceleration (Deceleration), and Jerk (Jerk).

• Velocity initial value selection (Bit 16, 17)

Setting value	Description
0: Set velocity	The velocity during command is used as the velocity commanded to the drive unit immediately after switching.
1: Actual velocity	The motor speed received from the drive unit at switching is used.*1
2: Automatic selection	The lower velocity between "0: Set velocity" and "1: Actual velocity" is used as the velocity commanded to the drive unit immediately after switching.*2

*1 When slave object "Velocity actual value (606CH)" is not mapped, the motor speed received from the drive unit at switching is not used. (Speed initial value selection becomes "0".)

*2 When slave object "Velocity actual value (606CH)" is not mapped, the velocity commanded to the drive unit immediately after switching becomes "0: Set velocity".

Precautions

- Speed override coefficient (AxisName.Cd.VelocityOverride) and Acceleration override coefficient (AxisName.Cd.AccelerationOverride) are valid.
- The set position and the feed machine position are updated by follow up.
- The switching time of the control mode depends on the specifications of the driver.
- When the stop cause occurs during the control mode switching, the axis will stop immediately.
- Do not start a positioning control FB while switching the control mode. Confirm that Driver control mode (AxisName.Md.Driver_Mode) has switched to "9: Velocity control (csv)" before starting a positioning control FB.
- Note the following precautions when using MR-J5(W)-G and switching from csp to csv without waiting for the motor to stop, or when switching from csv to csp.
  - Set servo parameter (extension setting) "ZSP disabled selection at control switching (PC76.1)" to "1: Disabled (control switching is performed regardless of the range of ZSP)" and disable monitoring of zero speed status. Note that doing so may cause vibration or impact to occur at control mode switching.
  - For the setting values of servo parameters (basic setting) "Electronic gear numerator (PA06)" and "Electronic gear denominator (PA07)", refer to the following.
    MELSEC iQ-R Motion Module User's Manual (Application)

Program example

In the following program example, the speed control command (bMoveVelocity) is set to TRUE, the mode is switched to velocity mode, and speed control is performed for Axis 1 (Axis0001) using the settings shown below.

• Setting

Item	Setting value
Velocity	100000.0
Acceleration	50000.0
Deceleration	50000.0
Jerk	0.0

■Axis

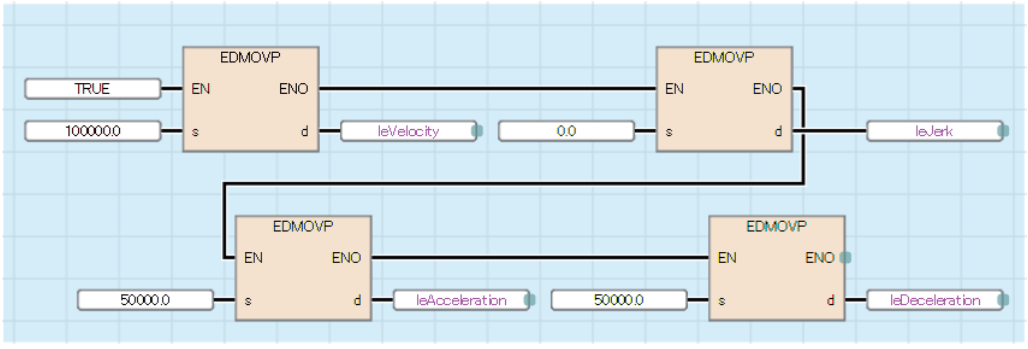
Axis No.	Label name	Data type	Comment
1	Axis0001	AXIS_REF	Axis 1

■Labels used

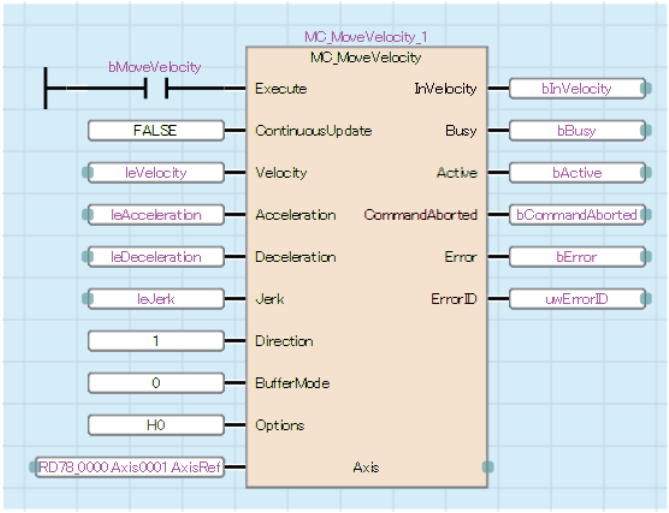
Label name	Data type	Comment
MC_MoveVelocity_1	MC_MoveVelocity	Speed Control FB
bMoveVelocity	Bit	Speed control command
leVelocity	Double-precision real number	Commanded velocity
leAcceleration	Double-precision real number	Acceleration
leDeceleration	Double-precision real number	Deceleration
leJerk	Double-precision real number	Jerk
bInVelocity	Bit	Target velocity reached
bBusy	Bit	Executing
bActive	Bit	Controlling
bCommandAborted	Bit	Abortion of execution
bError	Bit	Error
uwErrorID	Word [unsigned]/bit string [16-bit]	Error code

■FBD/LD program (CPU module side)

• Data setting for Speed Control



• Speed Control



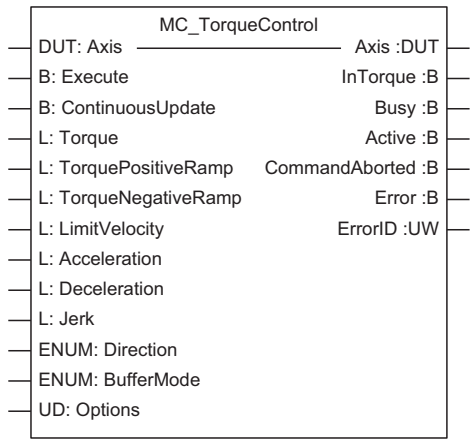
■ST program (Motion module side)

```
//-----Data setting for Speed Control-----
leVelocity:= 100000.0;
leAcceleration:= 50000.0;
leDeceleration:= 50000.0;
leJerk:= 0.0;

//-----Speed Control-----
MC_MoveVelocity_1(
   Axis:= Axis0001.AxisRef ,
   Execute:= bMoveVelocity ,
   ContinuousUpdate:= FALSE ,
   Velocity:= leVelocity ,
   Acceleration:= leAcceleration ,
   Deceleration:= leDeceleration ,
   Jerk:= leJerk ,
   Direction:= MC_DIRECTION__mcPositiveDirection ,
   BufferMode:= MC_BUFFER_MODE__mcAborting ,
   Options:= H00000000 ,
   InVelocity=> bInVelocity ,
   Busy=> bBusy ,
   Active=> bActive ,
   CommandAborted=> bCommandAborted ,
   Error=> bError ,
   ErrorID=> uwErrorID
);
```

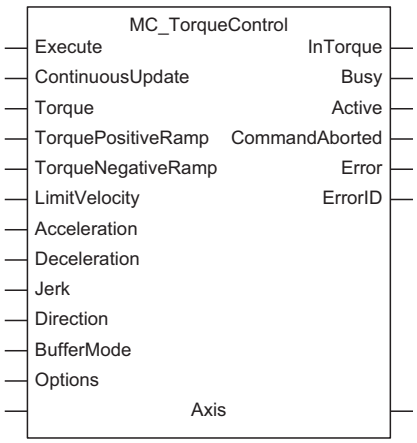
MC_TorqueControl

Switches the driver to cst, then performs torque control according to the specified target torque.

Ladder



FBD/LD



ST

```
MC_TorqueControl(
  Axis:= ?AXIS_REF? ,
  Execute:= ?BOOL? ,
  ContinuousUpdate:= ?BOOL? ,
  Torque:= ?LREAL? ,
  TorquePositiveRamp:= ?LREAL? ,
  TorqueNegativeRamp:= ?LREAL? ,
  LimitVelocity:= ?LREAL? ,
  Acceleration:= ?LREAL? ,
  Deceleration:= ?LREAL? ,
  Jerk:= ?LREAL? ,
  Direction:= ?INT? ,
  BufferMode:= ?INT? ,
  Options:= ?DWORD? ,
  InTorque=> ?BOOL? ,
  Busy=> ?BOOL? ,
  Active=> ?BOOL? ,
  CommandAborted=> ?BOOL? ,
  Error=> ?BOOL? ,
  ErrorID=> ?WORD?
);
```

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Torque Control	80	8	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
Axis	Axis information	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Execute command	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MC_TorqueControl (Torque Control).
ContinuousUpdate	Continuous update	BOOL	At start	TRUE, FALSE	FALSE	This variable sets whether to enable or disable continuous change of Target torque (Torque), Torque positive direction ramp (TorquePositiveRamp), Torque negative ramp (TorqueNegativeRamp), Limit velocity (LimitVelocity), Acceleration (Acceleration), and Deceleration (Deceleration). • FALSE: Disable • TRUE: Enable

Input variable	Name	Data type	Input	Setting range	Default value	Description
Torque	Target torque	LREAL	At start/ Retrigger possible/ Continuous update possible	-1000.0 to 1000.0[%]	0.0	This variable sets the set torque. Sets the ratio as a percentage to the rated torque of the servo motor used. The number of significant digits after the decimal point differs depending on the ID assigned to the slave object map. Values less than the number of significant digits are truncated. To output the set torque in the positive direction (address increasing direction), set a positive value. To output the set torque in the negative direction (address decreasing direction), set a negative value.
TorquePositiveRamp	Torque positive direction ramp	LREAL	At start/ Retrigger possible/ Continuous update possible	■When "0: Ramp method" 0.0 to 1000.0[%/s] ■When "1: Time constant method" or "2: Fixed time method" 0.0 to 8400.0[s]	0.0	This variable sets the positive direction target torque. For details, refer to the following. ☞ Page 240 Torque positive direction ramp (TorquePositiveRamp)/Torque negative direction ramp(TorqueNegativeRamp)
TorqueNegativeRamp	Torque negative direction ramp	LREAL	At start/ Retrigger possible/ Continuous update possible		0.0	This variable sets the negative direction target torque. ☞ Page 240 Torque positive direction ramp (TorquePositiveRamp)/Torque negative direction ramp(TorqueNegativeRamp)
LimitVelocity	Limit velocity	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0001 to 2500000000.0	0.0	This variable sets the velocity limit value for the cyclic torque mode. *: For the velocity limit value, refer to the driver manual. ■When using MR-J5(W)-G • Uses slave object "Velocity limit value (2D20H)" as the velocity limit value. • When slave object "Velocity limit value (2D20H)" is not mapped, velocity commands from the controller will not be enabled. (The setting value of servo parameter "Speed limit (PT67)" is enabled.) *: For details of slave objects, refer to the following. 📖MELSEC iQ-R Motion Module User's Manual (Application)
Acceleration	Acceleration	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the acceleration and acceleration/deceleration time used until the limit velocity is reached For details, refer to the following. ☞ Page 241 Acceleration (Acceleration)
Deceleration	Deceleration	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the deceleration used until the limit velocity is reached. For details, refer to the following. ☞ Page 241 Deceleration (Deceleration)
Jerk	Jerk	LREAL	At start	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the jerk used until the limit velocity is reached. For details, refer to the following. ☞ Page 241 Jerk (Jerk) *: When "0.0" is set or this setting is omitted, the jerk is not applied.
Direction	Direction selection	INT (MC_DIRECTION)	At start	0	0	Set this variable to "0". *: If a value other than "0" is set, "Out of Direction Selection Range (error code: 1A37H)" occurs.
BufferMode	Buffer mode	INT (MC_BUFFER_MODE)	At start	0, 1	0	This variable sets the buffer mode. • 0: Aborting (mcAborting) • 1: Buffered (mcBuffered) For details, refer to the following. ☞ Page 241 Buffer mode (BufferMode)

Input variable	Name	Data type	Input	Setting range	Default value	Description
Options	Options	DWORD(HEX)	At start	00000000H to 00060001H	00000000H	This variable sets the function options for MC_TorqueControl (Torque Control) by specifying the bit. For details, refer to the following. ☞ Page 242 Options (Options)

■Output variables

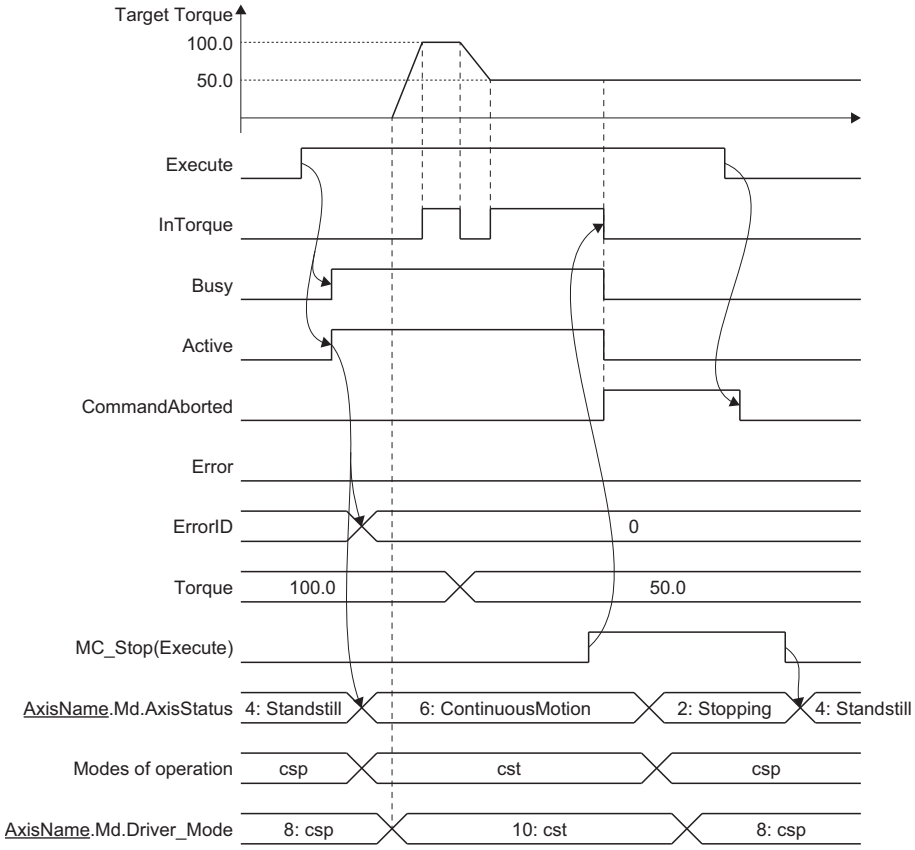
Output variable	Name	Data type	Default value	Description
InTorque	Target torque reached	BOOL	FALSE	This variable becomes TRUE when the target torque is reached. When the target torque has been changed due to change at Continuous update (ContinuousUpdate) TRUE, this variable remains FALSE until reaching the post-change target torque.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MC_TorqueControl (Torque Control) is executed.
Active	Controlling	BOOL	FALSE	This variable becomes TRUE while MC_TorqueControl (Torque Control) is controlling the axis.
CommandAborted	Abortion of execution	BOOL	FALSE	This variable becomes TRUE when execution of MC_TorqueControl (Torque Control) is aborted. This variable becomes FALSE when Execute command (Execute) becomes FALSE.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. 📖MELSEC iQ-R Motion Module User's Manual (Application)

Processing details

- MC_TorqueControl (Torque Control) switches the driver control mode to cst (cyclic torque mode) and executes control. This function controls the set torque based on the specified Torque positive direction ramp (TorquePositiveRamp) and Torque negative direction ramp (TorqueNegativeRamp). To finish MC_TorqueControl (Torque Control), start MC_Stop (Forced Stop).

■Timing chart

- When the operation completes normally



- When an error occurs

  For the timing chart when an error occurs, refer to the following.

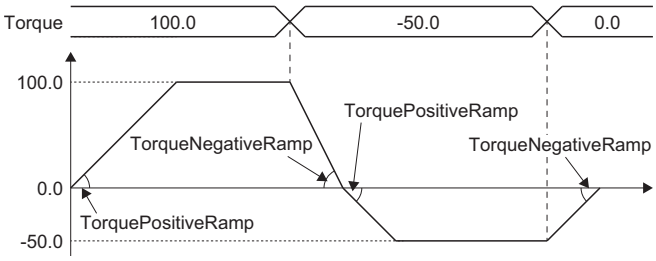
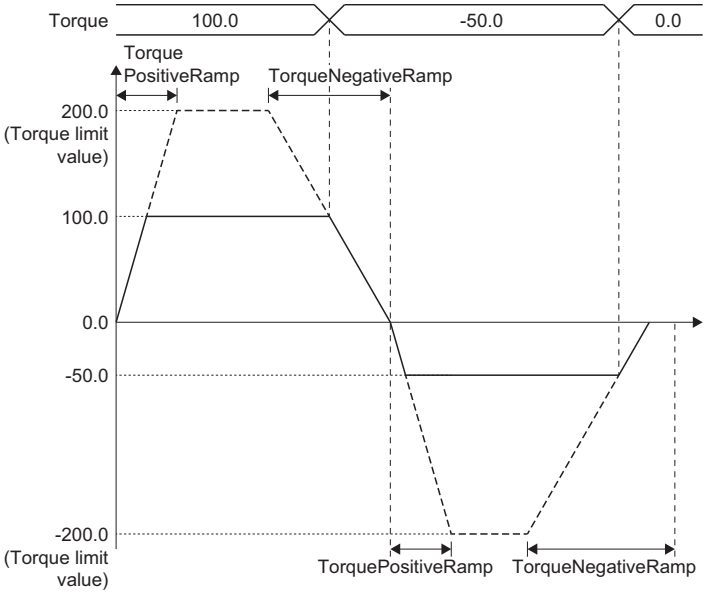
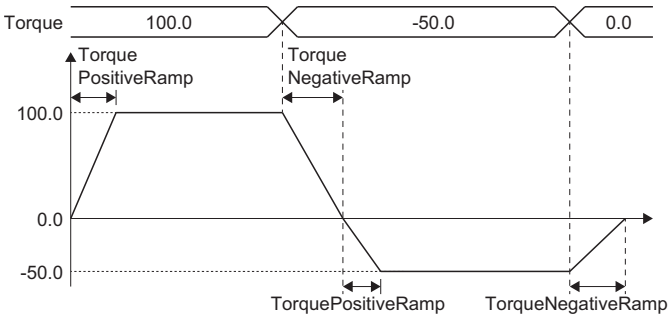
  ☞ Page 11 Basic operation of Execute command (Execute) type Motion control FBs

■Torque positive direction ramp (TorquePositiveRamp)/Torque negative direction ramp(TorqueNegativeRamp)

Sets the specified target torque.

Sets the setting value used until the target torque is reached from the current set torque.

The setting value used until the target torque is reached differs depending on the setting of Torque ramp function selection (Options (Options): Bit 16, 17).

Torque ramp function selection (Options (Options): Bit 16, 17)	Description	Setting range
0: Ramp method	Specify the ramp from the current set torque until the target torque is reached. 	0.0 to 1000.0[%/s]
1: Time constant method	Set the time until the set torque reaches the positive/negative torque limit value from "0". 	0.0 to 8400.0[s]
2: Fixed time method	Set the time from the current set torque until the target torque is reached. 	0.0 to 8400.0[s]

When the output torque direction is changed due to the target torque change, the set torque becomes "0.0" based on the setting value of Torque negative direction ramp (TorqueNegativeRamp). After that, the set torque becomes the target torque based on the setting value of Torque positive direction ramp (TorquePositiveRamp).

When Torque positive direction ramp (TorquePositiveRamp) and Torque negative direction ramp (TorqueNegativeRamp) are set to "0.0", the value reaches the target torque in one operation cycle.

■Acceleration (Acceleration)

Sets the acceleration used in MC_TorqueControl (Torque Control).

The setting range differs depending on the setting for Acceleration/deceleration method setting (Options (Options): Bit 0 to 2).

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000*1, positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	0.000000*1, positive numbers from 0.000001 to 8400.0[s]

*1 The operation performed at start varies depending on the setting of Operation selection at start acceleration/deceleration 0 (AxisName.Pr.AccelerationZeroBehavior). When changing the acceleration or deceleration, the changes are not accepted.

■Deceleration (Deceleration)

Sets the deceleration used in MC_TorqueControl (Torque Control).

Specifies the setting range if Acceleration/deceleration method setting (Options (Options): Bit 0 to 2) is set to "0: Acceleration/deceleration specification method (mcAccDec)". Deceleration (Deceleration) is not used if the acceleration/deceleration method is set to "1: Acceleration/deceleration time-fixed method (mcFixedTime)".

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000*1, positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	Not used

*1 The operation performed at start varies depending on the setting of Operation selection at start acceleration/deceleration 0 (AxisName.Pr.AccelerationZeroBehavior). When changing the acceleration or deceleration, the changes are not accepted.

■Jerk (Jerk)

Sets the jerk used in MC_TorqueControl (Torque Control).

Specifies the setting range if Acceleration/deceleration method setting (Options (Options): Bit 0 to 2) is set to "0: Acceleration/deceleration specification method (mcAccDec)". Jerk (Jerk) is not used if the acceleration/deceleration method is set to "1: Acceleration/deceleration time-fixed method (mcFixedTime)".

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000, positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	Not used

■Buffer mode (BufferMode)

Sets the operation used to perform multiple start (buffer mode).

The following buffer modes are settable in MC_TorqueControl (TorqueControl).

Setting value	Description
0: Aborting (mcAborting)	Aborts (cancels) the under-control FB and executes the next FB immediately.
1: Buffered (mcBuffered)	Buffers the next FB on the under-control FB. If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after completion of the under-control FB. Switches to torque control after the previous FB completes. When the previous FB is MC_TorqueControl (Torque Control), switches to torque control when Target torque reached (InTorque) is TRUE.

Point

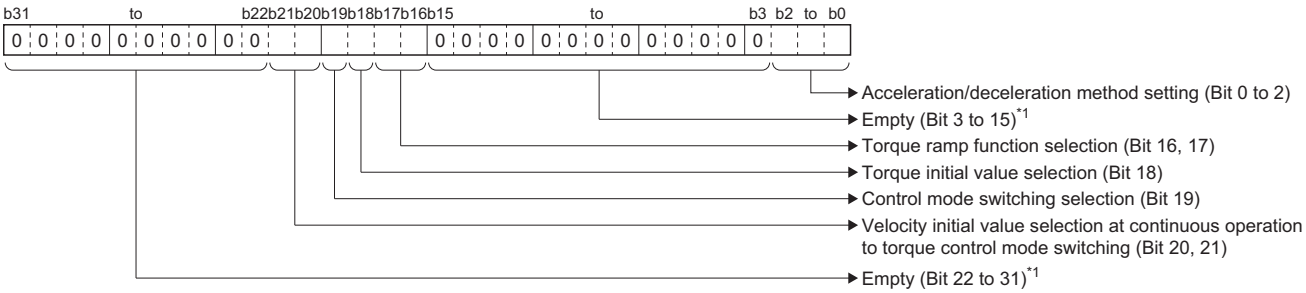
For details of multiple start (buffer mode), refer to the following.

MELSEC iQ-R Motion Module User's Manual (Application)

■Options (Options)

Sets the function options used for MC_TorqueControl (Torque Control) by specifying the bit.

Contents that are set by specifying the bit are shown below.



*1 Set empty areas to "0". When a value other than "0" is set, "Out of Options Range (error code: 1A4EH)" occurs.

Bit	Name	Description
0 to 2	Acceleration/deceleration method setting	Sets the acceleration/deceleration method used to perform control. • 0: Acceleration/deceleration specification method (mcAccDec) • 1: Acceleration/deceleration time-fixed method (mcFixedTime)
16, 17	Torque ramp function selection	Sets the method used until reaching the target torque from the set torque. • 0: Ramp method • 1: Time constant method • 2: Fixed time method *: For details, refer to the following. ☞ Page 240 Torque positive direction ramp (TorquePositiveRamp)/Torque negative direction ramp(TorqueNegativeRamp)
18	Torque initial value selection	Sets the torque initial value used when switching to cst (cyclic torque mode). • 0: Target torque • 1: Actual torque
19	Control mode switching selection*2	Sets the switching control mode. • 0: cst (cyclic synchronous torque mode) • 1: ct (continuous operation to torque control mode)
20, 21	Velocity initial value selection at continuous operation to torque control mode switching*3	Sets the velocity initial value used when changing the control mode from csp (cyclic synchronous position mode) to csv (cyclic velocity mode). • 0: Set velocity • 1: Actual velocity • 2: Automatic selection

*2 Switching to continuous operation to torque mode is possible when "Supported drive modes (Obj. 6502h) (Bit 20)" is TRUE. When this bit to set to "1: ct (Continuous operation to torque control mode)" while "Supported drive modes (Obj. 6502h) (Bit 20)" is FALSE, "Driver Control Mode Unsupported (error code: 1AE9H)" occurs.
For details of continuous operation to torque control mode, refer to the following.
MELSEC iQ-R Motion Module User's Manual (Application)

*3 Enabled only when Control mode switching selection (Options (Options): Bit 19) is set to "1: ct (Continuous operation to torque control mode)".

• Acceleration/deceleration method setting (Bit 0 to 2)

Setting value	Description
0: Acceleration/deceleration specification method (mcAccDec)	An acceleration/deceleration method using Acceleration (Acceleration), Deceleration (Deceleration), and Jerk (Jerk) set in MC_MoveAbsolute (Absolute Value Positioning).
1: Acceleration/deceleration time-fixed method (mcFixedTime)	An acceleration/deceleration method using the acceleration/deceleration time specified in MC_MoveAbsolute (Absolute Value Positioning) regardless of the velocity. The acceleration/deceleration time does not use Acceleration (Acceleration), Deceleration (Deceleration), and Jerk (Jerk).

• Torque initial value selection (Bit 18)

Setting value	Description
0: Target torque	Target torque (Torque) value at start is set as the set torque regardless of the value of Torque positive direction ramp (TorquePositiveRamp) and Torque negative direction ramp (TorqueNegativeRamp) immediately after the control mode switching.
1: Actual torque	The Torque actual value at switching is set as the set torque.

Point

Normally, set this bit to "0: Target torque". Set "1: Actual torque" only when switching the control mode immediately after the command to the motor is completed without waiting for the servo motor to stop.

- Velocity initial value selection at continuous operation to torque control mode switching (Bit 20, 21)

Setting value	Description
0: Set velocity	The velocity during command is used as the velocity commanded to the drive unit immediately after switching.
1: Actual velocity	The motor speed received from the drive unit at switching is used.*1
2: Automatic selection	The lower velocity between "0: Set velocity" and "1: Actual velocity" is used as the velocity commanded to the drive unit immediately after switching.*2

*1 When slave object "Velocity actual value (606CH)" is not mapped, the motor speed received from the drive unit at switching is not used. (Speed initial value selection becomes "0".)

*2 When slave object "Velocity actual value (606CH)" is not mapped, the velocity commanded to the drive unit immediately after switching becomes "0: Set velocity".

Precautions

- Speed override coefficient (AxisName.Cd.VelocityOverride) and Acceleration override coefficient (AxisName.Cd.AccelerationOverride) are valid.
- The set position and the feed machine position are updated by follow up.
- The switching time of the control mode depends on the specifications of the driver.
- When the stop cause occurs during the control mode switching, the axis will stop immediately.
- Do not start a positioning control FB while switching the control mode. Confirm that Driver control mode (AxisName.Md.Driver_Mode) has switched to "10: Torque control (cst)" before starting a positioning control FB.
- Note the following precautions when using MR-J5(W)-G and switching from csp to csv without waiting for the motor to stop, or when switching from csv to csp.
  - Set servo parameter (extension setting) "ZSP disabled selection at control switching (PC76.1)" to "1: Disabled (control switching is performed regardless of the range of ZSP)" and disable monitoring of zero speed status. Note that doing so may cause vibration or impact to occur at control mode switching.
  - For the setting values of servo parameters (basic setting) "Electronic gear numerator (PA06)" and "Electronic gear denominator (PA07)", refer to the following.
    MELSEC iQ-R Motion Module User's Manual (Application)

■Torque limit change during cst (cyclic torque mode)

- At restart or continuous update, if the target torque is changed to a value bigger than the torque limit value (Position direction torque limit value (AxisName.Md.TorqueLimit_Positive), Negative direction torque limit value (AxisName.Md.TorqueLimit_Negative)), "Torque Limit Value Over Warning (warning code: 0D12H)" occurs and the axis operates at the value before change.
- During torque control, if Position direction torque limit value (AxisName.Md.TorqueLimit_Positive), Negative direction torque limit value (AxisName.Md.TorqueLimit_Negative) are changed to a value smaller than the target torque, the target torque will be changed to the torque limit value. In this case, the set torque is changed in one cycle.

Program example

In the following program example, the torque control command (bTorqueControl) is set to TRUE, the mode is switched to torque mode, and torque control is performed for Axis 1 (Axis0001) using the settings shown below.

• Settings

Item	Setting value
Velocity	100000.0
Acceleration	50000.0
Deceleration	50000.0
Jerk	0.0

■Axis

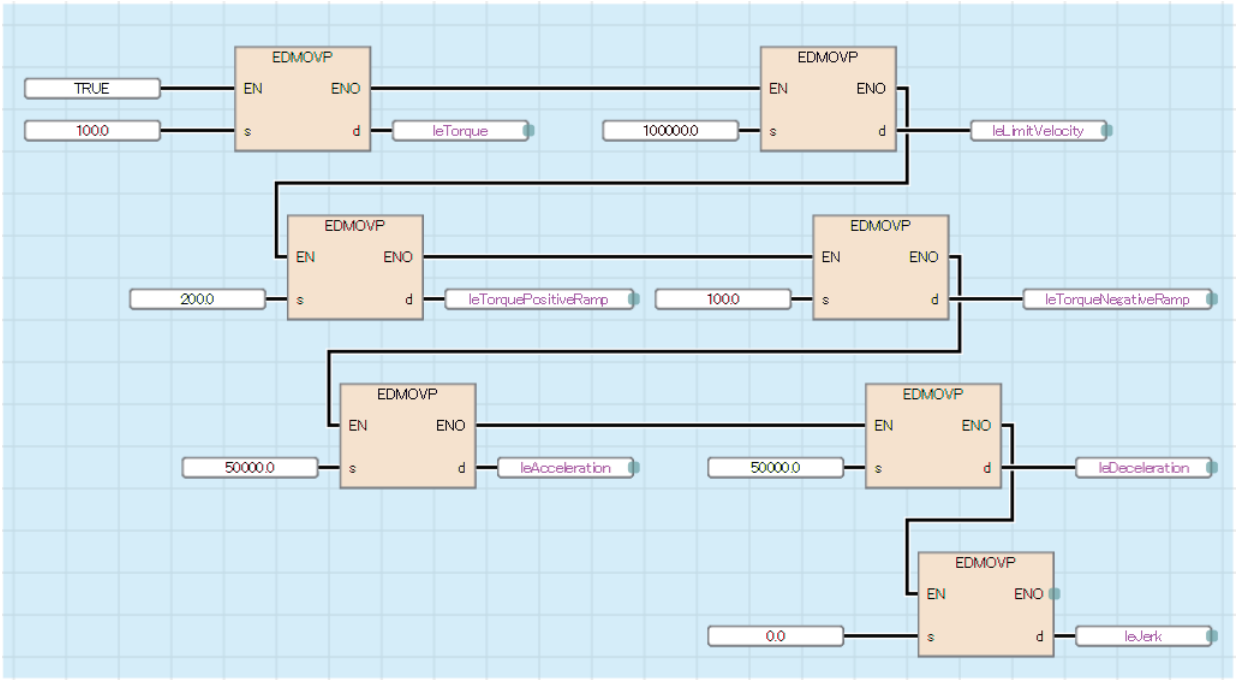
Axis No.	Label name	Data type	Comment
1	Axis0001	AXIS_REF	Axis 1

■Labels used

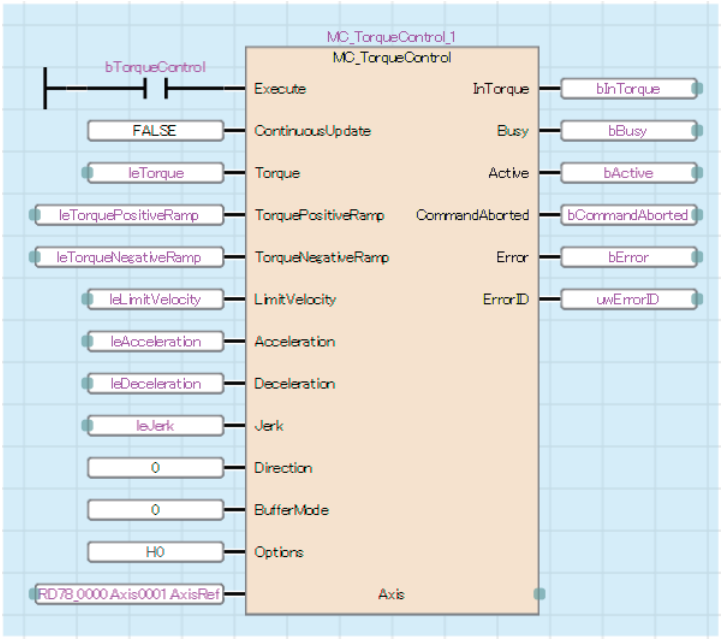
Label name	Data type	Comment
MC_TorqueControl_1	MC_TorqueControl	Torque Control FB
bTorqueControl	Bit	Torque control command
leTorque	Double-precision real number	Target torque
leTorquePositiveRamp	Double-precision real number	Torque positive direction ramp
leTorqueNegativeRamp	Double-precision real number	Torque negative direction ramp
leLimitVelocity	Double-precision real number	Limit velocity
leAcceleration	Double-precision real number	Acceleration
leDeceleration	Double-precision real number	Deceleration
leJerk	Double-precision real number	Jerk
bInTorque	Bit	Target torque reached
bBusy	Bit	Executing
bActive	Bit	Controlling
bCommandAborted	Bit	Abortion of execution
bError	Bit	Error
uwErrorID	Word [unsigned]/bit string [16-bit]	Error code

■FBD/LD program (CPU module side)

• Data setting for Torque Control



• Torque Control



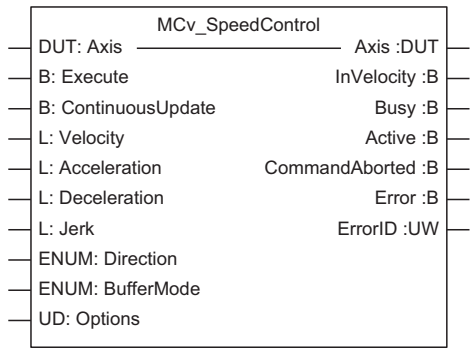
■ST program (Motion module side)

```
//-----Data setting for Torque Control-----
leTorque:= 100.0;
leTorquePositiveRamp:= 200.0;
leTorqueNegativeRamp:= 100.0;
leLimitVelocity:= 100000.0;
leAcceleration:= 50000.0;
leDeceleration:= 50000.0;
leJerk:= 0.0;

//-----Torque Control-----
MC_TorqueControl_1(
   Axis:= Axis0001.AxisRef ,
   Execute:= bTorqueControl ,
   ContinuousUpdate:= FALSE ,
   Torque:= leTorque ,
   TorquePositiveRamp:= leTorquePositiveRamp ,
   TorqueNegativeRamp:= leTorqueNegativeRamp ,
   LimitVelocity:= leLimitVelocity ,
   Acceleration:= leAcceleration ,
   Deceleration:= leDeceleration ,
   Jerk:= leJerk ,
   Direction:= 0 ,
   BufferMode:= MC_BUFFER_MODE__mcAborting ,
   Options:= H00000000 ,
   InTorque=> bInTorque ,
   Busy=> bBusy ,
   Active=> bActive ,
   CommandAborted=> bCommandAborted ,
   Error=> bError ,
   ErrorID=> uwErrorID
);
```

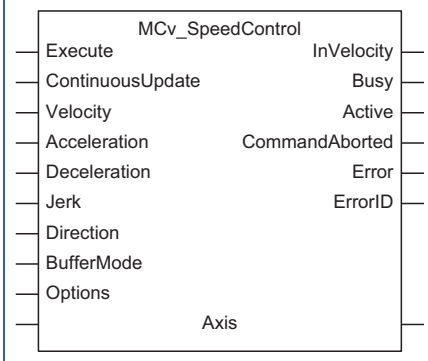
MCv_SpeedControl

Executes speed control including the position loop.

Ladder



FBD/LD



ST

```
MCv_SpeedControl(
  Axis:= ?AXIS_REF? ,
  Execute:= ?BOOL? ,
  ContinuousUpdate:= ?BOOL? ,
  Velocity:= ?LREAL? ,
  Acceleration:= ?LREAL? ,
  Deceleration:= ?LREAL? ,
  Jerk:= ?LREAL? ,
  Direction:= ?INT? ,
  BufferMode:= ?INT? ,
  Options:= ?DWORD? ,
  InVelocity=> ?BOOL? ,
  Busy=> ?BOOL? ,
  Active=> ?BOOL? ,
  CommandAborted=> ?BOOL? ,
  Error=> ?BOOL? ,
  ErrorID=> ?WORD?
);
```

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Speed Control (Including Position Loop)	56	8	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
Axis	Axis information	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Execute command	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MCv_SpeedControl (Speed Control (Including Position Loop)).
ContinuousUpdate	Continuous update	BOOL	At start	TRUE, FALSE	FALSE	This variable sets whether to enable or disable continuous change of Velocity (Velocity), Acceleration (Acceleration), and Deceleration (Deceleration). • FALSE: Disable • TRUE: Enable
Velocity	Velocity	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0, ±0.0001 to ±2500000000.0	0.0	This variable sets the commanded velocity. When the velocity is negative, the axis moves in the reverse direction. When this setting is omitted, the axis does not operate, but Axis status (AxisName.Md.AxisStatus) changes to "6: During continuous operation (ContinuousMotion)". For details, refer to the following. ☞ Page 248 Velocity (Velocity)
Acceleration	Acceleration	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the acceleration. For details, refer to the following. ☞ Page 248 Acceleration (Acceleration)

Input variable	Name	Data type	Input	Setting range	Default value	Description
Deceleration	Deceleration	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the deceleration. For details, refer to the following. ☞ Page 249 Deceleration (Deceleration)
Jerk	Jerk	LREAL	At start	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the jerk. For details, refer to the following. ☞ Page 249 Jerk (Jerk)
Direction	Direction selection	INT (MC_DIRECTION)	At start	1, 2	0	This variable sets the selected direction. • 1: Positive direction (mcPositiveDirection) • 2: Negative direction (mcNegativeDirection) *: If setting "2: Negative direction (mcNegativeDirection)" when Velocity (Velocity) is negative, the movement direction of the motor will be positive. *: When this setting is omitted, "Out of Direction Selection Range (error code: 1A37H)" occurs.
BufferMode	Buffer mode	INT (MC_BUFFER_MODE)	At start	0 to 5	0	This variable sets the buffer mode. • 0: Aborting (mcAborting) • 1: Buffered (mcBuffered) • 2: BlendingLow (mcBlendingLow) • 3: BlendingPrevious (mcBlendingPrevious) • 4: BlendingNext (mcBlendingNext) • 5: BlendingHigh (mcBlendingHigh) For details, refer to the following. ☞ Page 249 Buffer mode (BufferMode)
Options	Options	DWORD(HEX)	At start	00000000H to 00000001H	00000000H	This variable sets the function options for MCv_SpeedControl (Speed Control (Including Position Loop)) by specifying the bit. For details, refer to the following. ☞ Page 250 Options (Options)

■Output variables

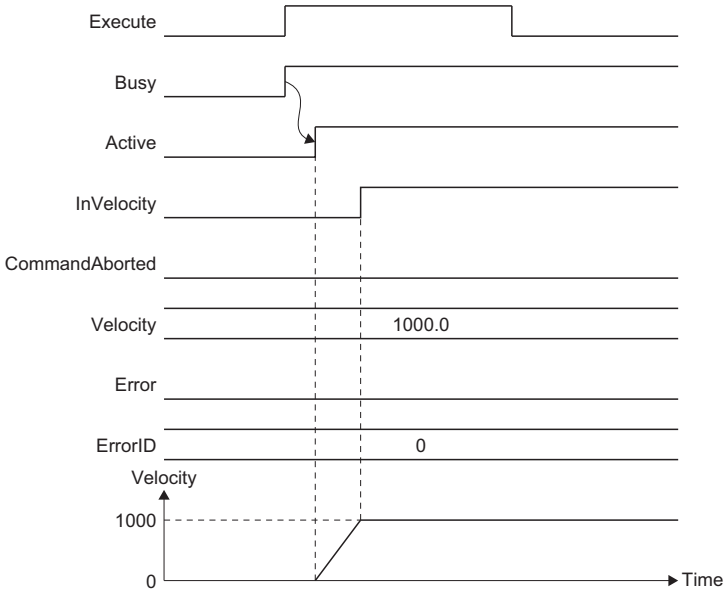
Output variable	Name	Data type	Default value	Description
InVelocity	Target speed reached	BOOL	FALSE	This variable becomes TRUE when the set velocity calculated by the Motion system reaches the target velocity. When the target velocity has been changed due to change at Continuous update (ContinuousUpdate) TRUE, this variable remains FALSE until the set velocity reaches the post-change target velocity.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MCv_SpeedControl (Speed Control (Including Position Loop)) is executed.
Active	Controlling	BOOL	FALSE	This variable becomes TRUE while MCv_SpeedControl (Speed Control (Including Position Loop)) is controlling the axis.
CommandAborted	Abortion of execution	BOOL	FALSE	This variable becomes TRUE when execution of MCv_SpeedControl (Speed Control (Including Position Loop)) is aborted. This variable becomes FALSE when Execute command (Execute) becomes FALSE.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. 📖MELSEC iQ-R Motion Module User's Manual (Application)

Processing details

- Sets the control mode of the driver to csp and executes the speed control for the specified axis using the specified velocity. To stop the axis, use MC_Stop (Forced Stop) or start another operation FB. To perform the speed control with the control mode of the driver set to csv, use MC_MoveVelocity (Speed Control).

■Timing chart

- When the operation completes normally



- When an error occurs

  For the timing chart when an error occurs, refer to the following.

  ☞ Page 11 Basic operation of Execute command (Execute) type Motion control FBs

■Velocity (Velocity)

Sets the commanded velocity used in MCv_SpeedControl (Speed Control (Including Position Loop)).

Setting range
0.0, ±0.0001 to ±2500000000.0[*1*2]

*1 To perform floating-point operation, the lower limit value of the commanded velocity is limited by the following restrictions. If the velocity that is converted from the target velocity into the operation cycle is less than "0.00001", "Out of Calculation Cycle Converted Speed Range (error code: 1AE4H)" occurs. (If the velocity is changed, "Out of Calculation Cycle Converted Speed Range Warning (warning code: 0D2FH)" occurs instead). To improve the precision of floating-point operation, change Position command unit (AxisName.Pr.Unit_Position) and Speed command unit (AxisName.Pr.Unit_Velocity) and perform settings so that the velocity converted into the operation cycle is not less than "0.00001".

*2 When a Motion control FB with a set velocity of "0.0" performs a multiple start, the set velocity of the previous Motion control FB is used.

■Acceleration (Acceleration)

Sets the acceleration used in MCv_SpeedControl (Speed Control (Including Position Loop)).

The setting range differs depending on the setting for Acceleration/deceleration method setting (Options (Options): Bit 0 to 2).

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000[*1], positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	0.000000[*1], positive numbers from 0.000001 to 8400.0[s]

*1 The operation performed at start varies depending on the setting of Operation selection at start acceleration/deceleration 0 (AxisName.Pr.AccelerationZeroBehavior). When changing the acceleration or deceleration, the changes are not accepted.

■Deceleration (Deceleration)

Sets the deceleration used in MCv_SpeedControl (Speed Control (Including Position Loop)).

Specifies the setting range if Acceleration/deceleration method setting (Options (Options): Bit 0 to 2) is set to "0: Acceleration/deceleration specification method (mcAccDec)". Deceleration (Deceleration) is not used if the acceleration/deceleration method is set to "1: Acceleration/deceleration time-fixed method (mcFixedTime)".

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000[*1], positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	Not used

*1 The operation performed at start varies depending on the setting of Operation selection at start acceleration/deceleration 0 (AxisName.Pr.AccelerationZeroBehavior). When changing the acceleration or deceleration, the changes are not accepted.

■Jerk (Jerk)

Sets the jerk used in MCv_SpeedControl (Speed Control (Including Position Loop)).

Specifies the setting range if Acceleration/deceleration method setting (Options (Options): Bit 0 to 2) is set to "0: Acceleration/deceleration specification method (mcAccDec)". Jerk (Jerk) is not used if the acceleration/deceleration method is set to "1: Acceleration/deceleration time-fixed method (mcFixedTime)".

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000, positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	Not used

■Buffer mode (BufferMode)

Sets the operation used to perform multiple start (buffer mode).

The following buffer modes are settable in MCv_SpeedControl (Speed Control (Including Position Loop)).

Setting value	Description
0: Aborting (mcAborting)	Aborts (cancels) the under-control FB and executes the next FB immediately.
1: Buffered (mcBuffered)	Buffers the next FB on the under-control FB. If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after completion of the under-control FB.
2: BlendingLow (mcBlendingLow)	Buffers the next FB on the under-control FB.[*1] If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. When the under-control FB reaches the target position, the lower target velocity between the under-control FB and the buffering FB is used as the switching speed.
3: BlendingPrevious (mcBlendingPrevious)	Buffers the next FB on the under-control FB.[*1] If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. When the under-control FB reaches the target position, the switching speed changes to the target velocity of the under-control FB.
4: BlendingNext (mcBlendingNext)	Buffers the next FB on the under-control FB.[*1] If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. When the under-control FB reaches the target position, the switching speed changes to the target velocity of the buffering FB.
5: BlendingHigh (mcBlendingHigh)	Buffers the next FB on the under-control FB.[*1] If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. When the under-control FB reaches the target position, the higher target velocity between the under-control FB and the buffering FB is used as the switching speed.

*1 Stops are not performed between the under-control FB and the buffering FB.

Point

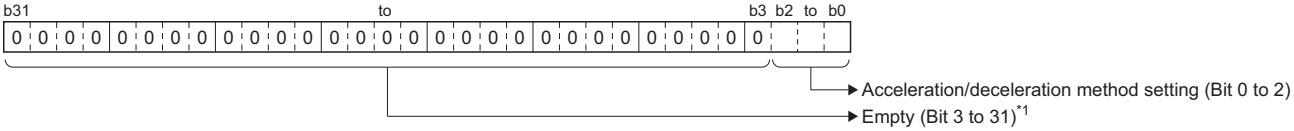
For details of multiple start (buffer mode), refer to the following.

MELSEC iQ-R Motion Module User's Manual (Application)

■Options (Options)

Sets the function options used for MCv_SpeedControl (Speed Control (Including Position Loop)) by specifying the bit.

Contents that are set by specifying the bit are shown below.



*1 Set empty areas to "0". When a value other than "0" is set, "Out of Options Range (error code: 1A4EH)" occurs.

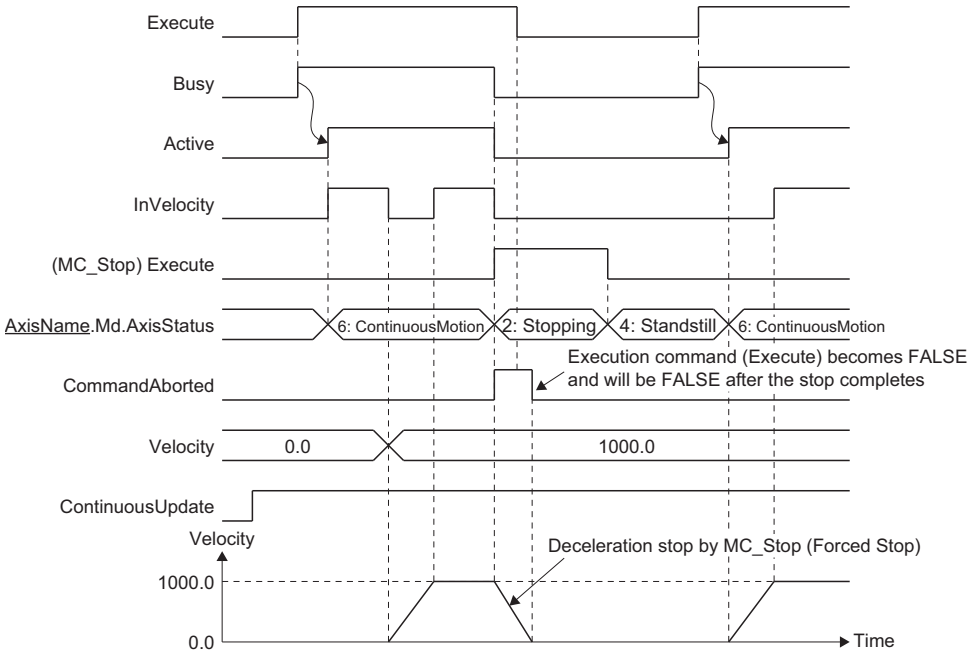
Bit	Name	Description
0 to 2	Acceleration/deceleration method setting	Sets the acceleration/deceleration method used to perform control. • 0: Acceleration/deceleration specification method (mcAccDec) • 1: Acceleration/deceleration time-fixed method (mcFixedTime)

• Acceleration/deceleration method setting (Bit 0 to 2)

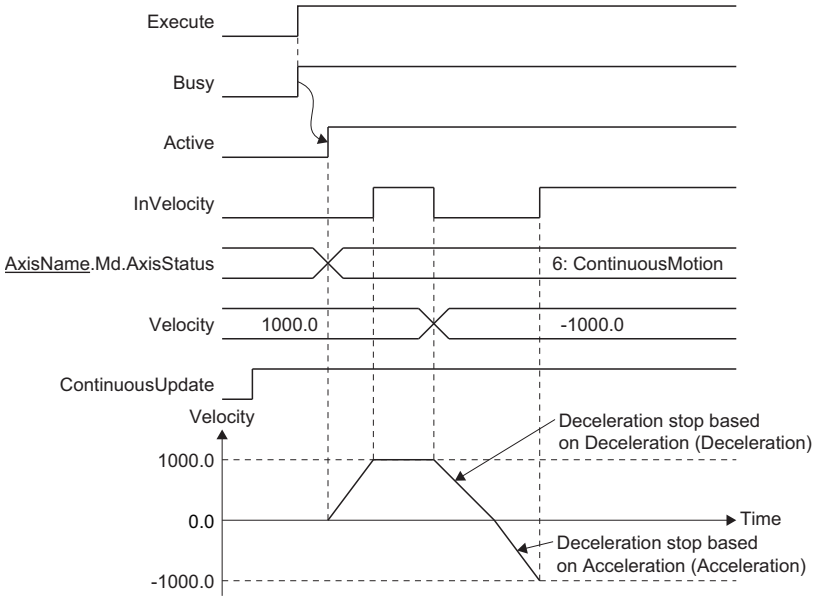
Setting value	Description
0: Acceleration/deceleration specification method (mcAccDec)	An acceleration/deceleration method using Acceleration (Acceleration), Deceleration (Deceleration), and Jerk (Jerk) set in MCv_SpeedControl (Speed Control (Including Position Loop)).
1: Acceleration/deceleration time-fixed method (mcFixedTime)	An acceleration/deceleration method using the acceleration/deceleration time specified in MCv_SpeedControl (Speed Control (Including Position Loop)) regardless of the velocity. The acceleration/deceleration time does not use Acceleration (Acceleration), Deceleration (Deceleration), and Jerk (Jerk).

■Operation overview

• The operation at start and when a stop cause occurs is shown below.



• If the sign of Velocity (Velocity) is reversed and the operation direction is changed when Continuous update (ContinuousUpdate) is TRUE, the axis accelerates toward the target velocity once after the deceleration stop.



■Necessary objects

When using MCv_SpeedControl (Speed Control (Including Position Loop)) for a single axis, set the following slave objects for the axis.

• Target position (607AH)

If the slave object is not set, "Necessary Slave Object Unset (error code: 1AA8H)" occurs and the axis will not start.

For details of slave object settings, refer to the following.

📖MELSEC iQ-R Motion Module User's Manual (Application)

Program example

In the following program example, the speed control command (bSpeedControl) is set to TRUE, the mode is switched to speed control including position loop, and speed control including position loop is performed for Axis 1 (Axis0001) using the settings shown below.

• Settings

Item	Setting value
Velocity	100000.0
Acceleration	50000.0
Deceleration	50000.0
Jerk	0.0

■Axis

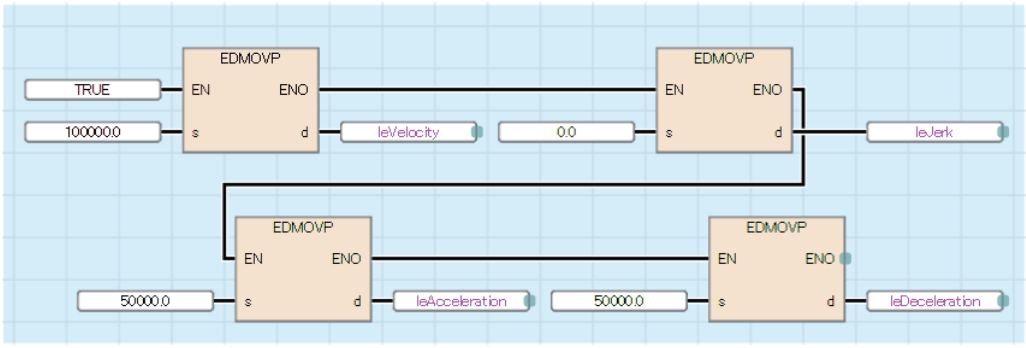
Axis No.	Label name	Data type	Comment
1	Axis0001	AXIS_REF	Axis 1

■Labels used

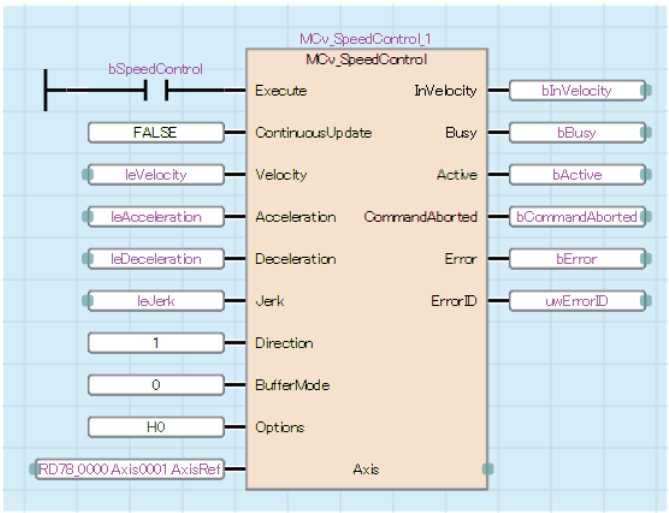
Label name	Data type	Comment
MCv_SpeedControl_1	MCv_SpeedControl	Speed Control FB
bSpeedControl	Bit	Speed control command
leVelocity	Double-precision real number	Commanded velocity
leAcceleration	Double-precision real number	Acceleration
leDeceleration	Double-precision real number	Deceleration
leJerk	Double-precision real number	Jerk
bInVelocity	Bit	Target velocity reached
bBusy	Bit	Executing
bActive	Bit	Controlling
bCommandAborted	Bit	Abortion of execution
bError	Bit	Error
uwErrorID	Word [unsigned]/bit string [16-bit]	Error code

■FBD/LD program (CPU module side)

• Data setting for Speed Control



• Speed Control (Including Position Loop)

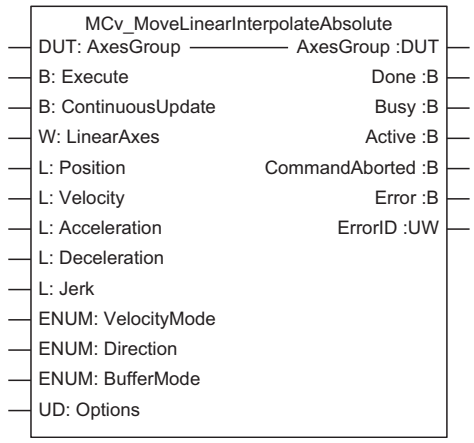
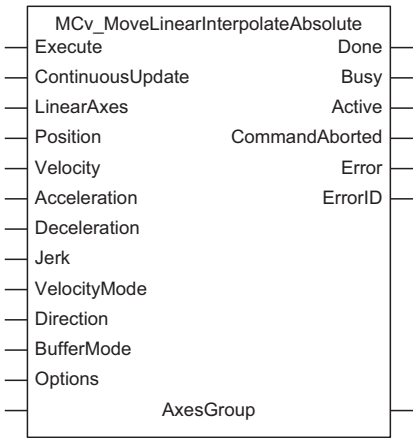


■ST program (Motion module side)

```
//-----Data setting for Speed Control-----
leVelocity:= 100000.0;
leAcceleration:= 50000.0;
leDeceleration:= 50000.0;
leJerk:= 0.0;

//-----Speed Control (Including Position Loop)-----
MCv_SpeedControl_1(
   Axis:= Axis0001.AxisRef ,
   Execute:= bSpeedControl ,
   ContinuousUpdate:= FALSE ,
   Velocity:= leVelocity ,
   Acceleration:= leAcceleration ,
   Deceleration:=leDeceleration ,
   Jerk:= leJerk ,
   Direction:= MC_DIRECTION__mcPositiveDirection ,
   BufferMode:= MC_BUFFER_MODE__mcAborting ,
   Options:= H00000000 ,
   InVelocity=> bInVelocity ,
   Busy=> bBusy ,
   Active=> bActive ,
   CommandAborted=> bCommandAborted ,
   Error=> bError ,
   ErrorID=> uwErrorID
);
```

MCv_MoveLinearInterpolateAbsolute

Specifies the target position of the absolute position of the set axes group, then executes positioning by linear interpolation control.

Ladder	FBD/LD	ST
		MCv_MoveLinearInterpolateAbsolute(AxesGroup:= ?AXES_GROUP_REF? , Execute:= ?BOOL? , ContinuousUpdate:= ?BOOL? , LinearAxes:= ?INT(0..15)? , Position:= ?LREAL(0..15)? , Velocity:= ?LREAL? , Acceleration:= ?LREAL? , Deceleration:= ?LREAL? , Jerk:= ?LREAL? , VelocityMode:= ?INT? , Direction:= ?INT(0..15)? , BufferMode:= ?INT? , Options:= ?DWORD? , Done=> ?BOOL? , Busy=> ?BOOL? , Active=> ?BOOL? , CommandAborted=> ?BOOL? , Error=> ?BOOL? , ErrorID=> ?WORD?);

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Absolute Value Linear Interpolation Control	248	8	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
AxesGroup	Axes group information	AXES_GROUP_REF	At start	—	Omission not possible	This variable sets the axes group. For the variables used (AxesGroupName.AxesGroupRef.), refer to the following. ☞ Page 46 AxesGroupName.AxesGroupRef. (Axes group information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Start	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MCv_MoveLinearInterpolateAbsolute (Absolute Value Linear Interpolation Control).
ContinuousUpdate	Continuous update	BOOL	At start	TRUE, FALSE	FALSE	This variable sets whether to enable or disable continuous change of Velocity (Velocity), Acceleration (Acceleration), and Deceleration (Deceleration). Continuous update re-imports input variables to the instance in execution without aborting the operation. • FALSE: Disable • TRUE: Enable

Input variable	Name	Data type	Input	Setting range	Default value	Description
LinearAxes	Linear interpolation axis	INT[0..15]	At start	1 to 16	0	This variable specifies the axes to be used in linear interpolation control from the structuring axes. Specifies the index No. (1 to 16) of the structuring axes in the array. *: When Velocity mode (VelocityMode) is set to "2: Reference axis speed (ReferenceAxisSpeed)", the first element of the array is regarded as the reference axis. For details, refer to the following. ☞ Page 260 Linear interpolation axis (LinearAxes)
Position	Target position	LREAL[0..15]	At start	-10000000000.0 to 10000000000.0	0.0	This variable sets the target position for linear interpolation. The target position is set as 1-dimensional array data. This data is treated as the absolute position for structuring axes 1 to 16. For details, refer to the following. ☞ Page 261 Target position (Position)
Velocity	Velocity	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0, 0.0001 to 2500000000.0	0.0	This variable sets the speed command value. For details, refer to the following. ☞ Page 261 Velocity (Velocity)
Acceleration	Acceleration	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the acceleration. For details, refer to the following. ☞ Page 261 Acceleration (Acceleration)
Deceleration	Deceleration	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the deceleration. For details, refer to the following. ☞ Page 262 Deceleration (Deceleration)
Jerk	Jerk	LREAL	At start	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the jerk. For details, refer to the following. ☞ Page 262 Jerk (Jerk)
VelocityMode	Speed mode	INT (MC_INTERPOLATE_SPEED_MODE)	At start	0 to 2	0	This variable sets the speed mode for interpolation control. • 0: Vector speed (VectorSpeed) • 1: Long axis speed (LongAxisSpeed) • 2: Reference axis speed (ReferenceAxisSpeed) For details, refer to the following. ☞ Page 262 Velocity mode (VelocityMode)
Direction	Direction selection	INT (MC_DIRECTION[0..15])	At start	1 to 3	0	This variable sets the selected direction. The direction selection is set as 1-dimensional array data. This data is treated as the direction selection for structuring axes 1 to 16. • 1: Positive direction (mcPositiveDirection) • 2: Negative direction (mcNegativeDirection) • 3: Shortest path (mcShortestWay) *: When this setting is omitted, "Out of Direction Selection Range (error code: 1A37H)" occurs. For details, refer to the following. ☞ Page 264 Direction selection (Direction)
BufferMode	Buffer mode	INT (MC_BUFFER_MODE)	At start	0 to 5	0	This variable sets the buffer mode. • 0: Aborting (mcAborting) • 1: Buffered (mcBuffered) • 2: BlendingLow (mcBlendingLow) • 3: BlendingPrevious (mcBlendingPrevious) • 4: BlendingNext (mcBlendingNext) • 5: BlendingHigh (mcBlendingHigh) For details, refer to the following. ☞ Page 264 Buffer mode (BufferMode)

Input variable	Name	Data type	Input	Setting range	Default value	Description
Options	Options	DWORD(HEX)	At start	00000000H to 00010001H	00000000H	This variable sets the function options for MCv_MoveLinearInterpolateAbsolute (Absolute Value Linear Interpolation Control) by specifying the bit. For details, refer to the following. ☞ Page 265 Options (Options)

■Output variables

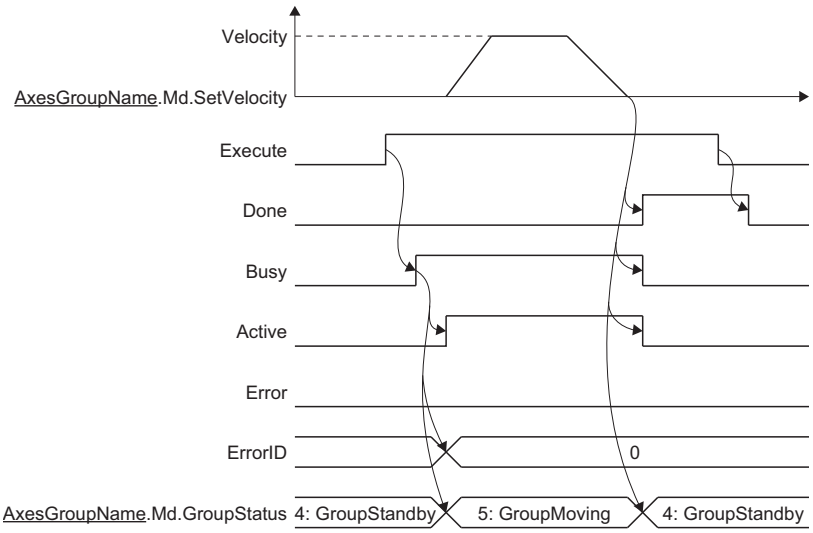
Output variable	Name	Data type	Default value	Description
Done	Completion	BOOL	FALSE	This variable becomes TRUE when the control is completed. This variable changes as shown below based on the status of Start (Execute) at operation completion. ■When Start (Execute) is TRUE This variable remains TRUE until Start (Execute) becomes FALSE. ■When Start (Execute) is FALSE This variable becomes TRUE for one cycle only.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MCv_MoveLinearInterpolateAbsolute (Absolute Value Linear Interpolation Control) is executed.
Active	Controlling	BOOL	FALSE	This variable becomes TRUE while MCv_MoveLinearInterpolateAbsolute (Absolute Value Linear Interpolation Control) is controlling the axis. When multiple MCv_MoveLinearInterpolateAbsolute (Absolute Value Linear Interpolation Control) FBs are executed for the same axes group, only the Controlling (Active) of one MCv_MoveLinearInterpolateAbsolute (Absolute Value Linear Interpolation Control) FB becomes TRUE.
CommandAborted	Abortion of execution	BOOL	FALSE	This variable becomes TRUE when execution of MCv_MoveLinearInterpolateAbsolute (Absolute Value Linear Interpolation Control) is aborted because of an error or multiple start.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. MELSEC iQ-R Motion Module User's Manual (Application)

Processing details

- Linear interpolation control specifies an axes group and performs interpolation control so that the path from the start point (move start point) to the end point becomes a straight line. In linear interpolation control, interpolation control using a maximum of 4 axes is performed.
- In MCv_MoveLinearInterpolateAbsolute (Absolute Value Linear Interpolation Control), specify the target position used for the absolute position and perform absolute value linear interpolation control.

■Timing chart

- When the operation completes normally



- When an error occurs

  For the timing chart when an error occurs, refer to the following.

  ☞ Page 11 Basic operation of Execute command (Execute) type Motion control FBs

■Operation overview of 1-axis linear interpolation control (absolute value)

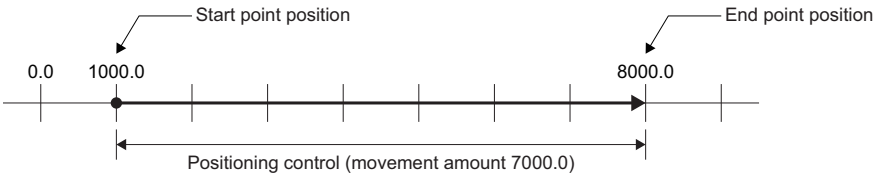
1-axis linear interpolation is performed from the current position at start (start point position) to the specified position (end point position) set in Target position (Position).

Ex.

When the start point position for structuring axis 1 is "1000.0" and Target position (Position) is set to "8000.0"

LinearAxes[0]:= 1

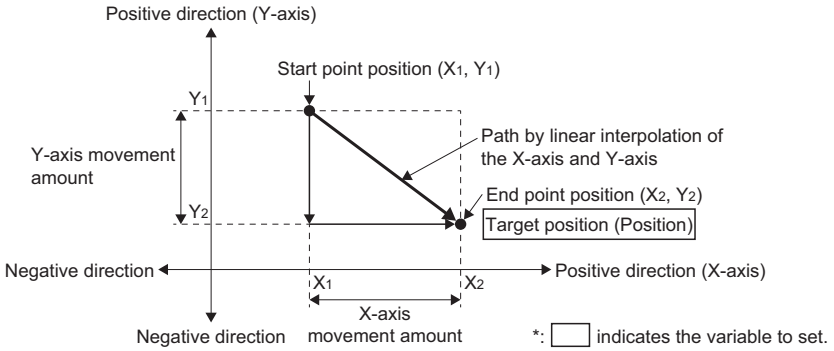
Position[0]:= 8000.0



■Operation overview of 2-axis linear interpolation control (absolute value)

2-axis linear interpolation is performed from the current position at start (start point position) to the specified position (end point position) set in Target position (Position).

The movement direction is determined by the start point position and the end point position of each axis.



Ex.

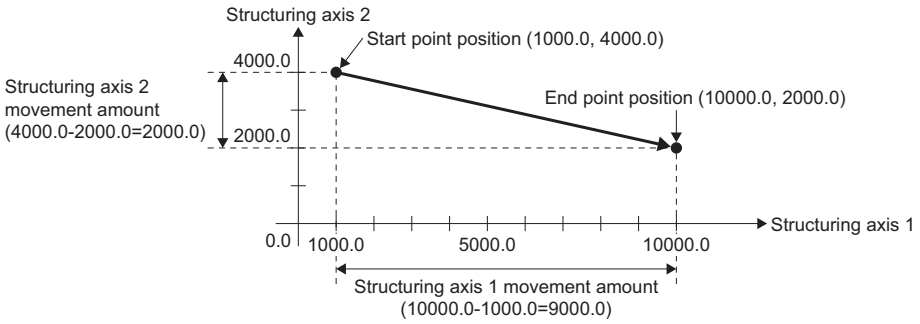
When the start point position is "Structuring axis 1: 1000.0, Structuring axis 2: 4000.0" and Target position (Position) is set to "Structuring axis 1: 10000.0, Structuring axis 2: 2000.0"

LinearAxes[0]:= 1

LinearAxes[1]:= 2

Position[0]:= 10000.0

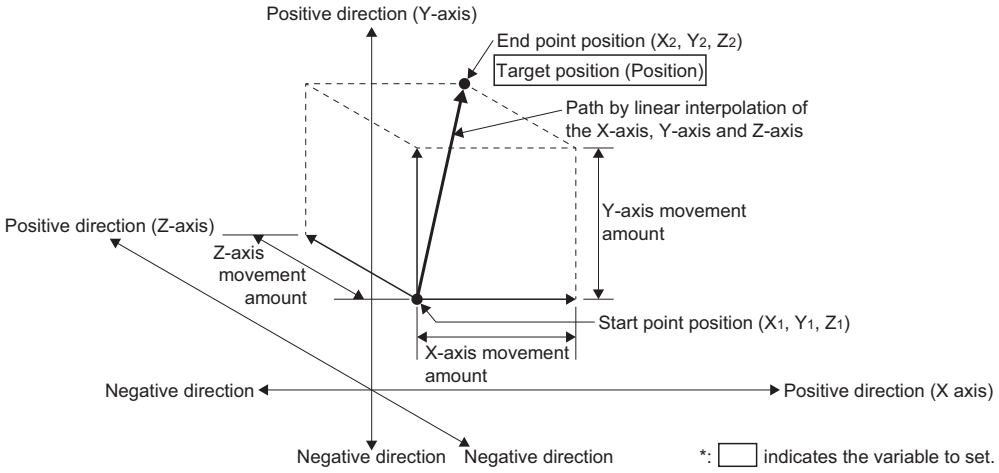
Position[1]:= 2000.0



■Operation overview of 3-axis linear interpolation control (absolute value)

3-axis linear interpolation is performed from the current position at start (start point position) to the specified position (end point position) set in Target position (Position).

The movement direction is determined by the start point position and the end point position of each axis.



Ex.

When the start position is "Structuring axis 1: 1000.0, Structuring axis 2: 2000.0, Structuring axis 3: 1000.0" and Target position (Position) is set to "Structuring axis 1: 7000.0, Structuring axis 2: 8000.0, Structuring axis 3: 4000.0"

LinearAxes[0]:= 1

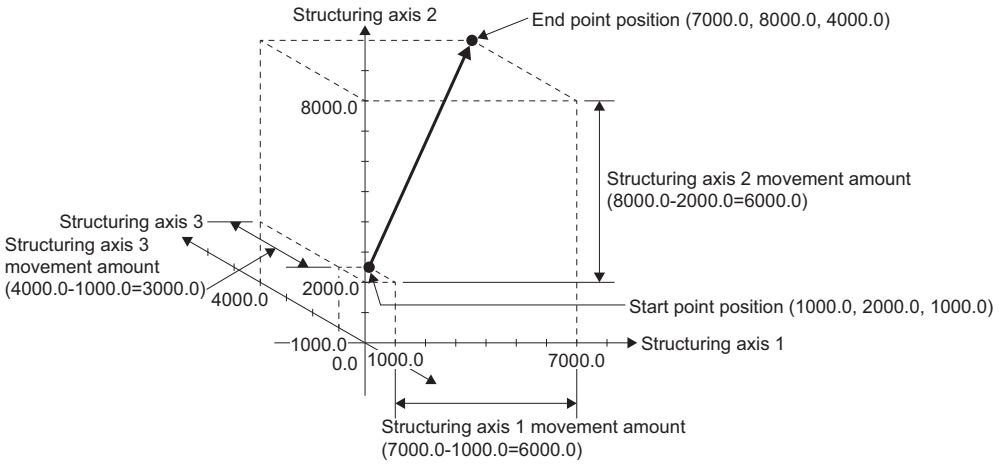
LinearAxes[1]:= 2

LinearAxes[2]:= 3

Position[0]:= 7000.0

Position[1]:= 8000.0

Position[2]:= 4000.0



■Linear interpolation axis (LinearAxes)

In linear interpolation control, linear interpolation is performed using arbitrary axes from the structuring axes set to the axes group.

Set structuring axes to perform linear interpolation by Linear interpolation axis (LinearAxes) from the structuring axes of the axes group. Linear interpolation axis (LinearAxes) has 16 array elements. For the arrays, first set only the index Nos. (1 to 16: structuring axis 1 to 16) of structuring axes to be used for interpolation control to the front, then set the rest of the arrays to "0". Arrays set to "0" can be omitted.

- The setting number of the linear interpolation axes must be set to the maximum number of the interpolation axes for interpolation control (4 axes) or less.
- The number of structuring axes to be used for interpolation control must be the number of axes registered to the structuring axes or less.
- When Velocity mode (VelocityMode) is "2: Reference axis speed (ReferenceAxisSpeed)", the structuring axis of the first element in Linear interpolation axis (LinearAxes) becomes the reference axis.
- When specifying the following, "Linear Interpolation Axis Setting Incorrect (error code: 1AB9H)" occurs and the axis will not start.
  - When specifying an structuring axis where an axis is not set
  - When the first element of Linear interpolation axis (LinearAxes) is "0"
  - When specifying the same index No. to Linear interpolation axis (LinearAxes) in duplicate
  - When the number of axes set in Linear interpolation axis (LinearAxes) exceeds the maximum number of interpolation axes

Ex.

When structuring axis 2, 3, and 4 are set to the linear interpolation axis

LinearAxes[0]:= 2;
LinearAxes[1]:= 3;
LinearAxes[2]:= 4;
LinearAxes[3]:= 0; *1
⋮
LinearAxes[15]:= 0; *1

*1 "LinearAxes[3]:=0; to LinearAxes[15]:=;0" can be omitted.

■Target position (Position)

Sets the target position for linear interpolation.

Target position (Position) has 16 array elements.

- The valid range differs depending on whether the software stroke limit is valid/invalid, Direction selection (Direction), and the setting for Target position specification exceeding the ring counter (Options (Options): Bit 16). For details, refer to the following.

  ☞ Page 202 Target position (Position)
- Position[0..15] indicates the target position of structuring axis 1 to 16. Set the position in relation to the structuring axes set in Linear interpolation axis (LinearAxes).
- Target position (Position) of structuring axes not set to the linear interpolation axes is ignored.
- Target position (Position) of axes after all structuring axes set to the linear interpolation axes can be omitted.

Ex.

When setting structuring axis 2, 3, 4 to the linear interpolation axes, and setting the target positions of structuring axis 2, 3, and 4 to 2000.0, 3000.0, and 4000.0 respectively

```
LinearAxes[0]:= 2;
LinearAxes[1]:= 3;
LinearAxes[2]:= 4;
LinearAxes[3]:= 0; *1
⋮
LinearAxes[15]:= 0; *1
Position[0]:= 2000.0;
Position[1]:= 3000.0;
Position[2]:= 4000.0;
Position[3]:= 0.0; *1
⋮
Position[15]:= 0.0; *1
```

*1 "LinearAxes[3]:=0; to LinearAxes[15]:=0;" and "Position[3]:=0.0; to Position[15]:=0.0;" can be omitted.

■Velocity (Velocity)

Sets the maximum speed of the path used in MCv_MoveLinearInterpolateAbsolute (Absolute Value Linear Interpolation Control).

Setting range
0.0, 0.0001 to 2500000000.0*1*2

*1 To perform floating-point operation, the lower limit value of the commanded velocity is limited by the following restrictions.
If the velocity that is converted from the target velocity into the operation cycle is less than "0.00001", "Out of Calculation Cycle Converted Speed Range (error code: 1AE4H)" occurs. (If the velocity is changed, "Out of Calculation Cycle Converted Speed Range Warning (warning code: 0D2FH)" occurs instead). To improve the precision of floating-point operation, change Position command unit (AxisName.Pr.Unit_Position) and Speed command unit (AxisName.Pr.Unit_Velocity) and perform settings so that the velocity converted into the operation cycle is not less than "0.00001".

*2 When a Motion control FB with a set velocity of "0.0" performs a multiple start, the set velocity of the previous Motion control FB is used.

■Acceleration (Acceleration)

Sets the acceleration used in MCv_MoveLinearInterpolateAbsolute (Absolute Value Linear Interpolation Control).

The setting range differs depending on the setting for Acceleration/deceleration method setting (Options (Options): Bit 0 to 2).

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000*1, positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	0.000000*1, positive numbers from 0.000001 to 8400.0[s]

*1 The operation performed at start varies depending on the setting of Operation selection at start acceleration/deceleration 0 (AxisName.Pr.AccelerationZeroBehavior). When changing the acceleration or deceleration, the changes are not accepted.

■Deceleration (Deceleration)

Sets the deceleration used in MCv_MoveLinearInterpolateAbsolute (Absolute Value Linear Interpolation Control).
Specifies the setting range if Acceleration/deceleration method setting (Options (Options): Bit 0 to 2) is set to "0: Acceleration/deceleration specification method (mcAccDec)". Deceleration (Deceleration) is not used if the acceleration/deceleration method is set to "1: Acceleration/deceleration time-fixed method (mcFixedTime)".

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000[*1], positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	Not used

*1 The operation performed at start varies depending on the setting of Operation selection at start acceleration/deceleration 0 (AxisName.Pr.AccelerationZeroBehavior). When changing the acceleration or deceleration, the changes are not accepted.

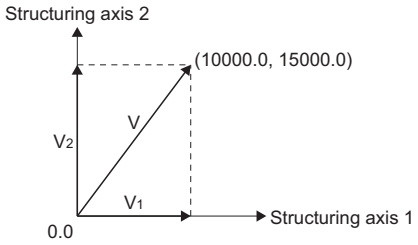
■Jerk (Jerk)

Sets the jerk used in MCv_MoveLinearInterpolateAbsolute (Absolute Value Linear Interpolation Control).
Specifies the setting range if Acceleration/deceleration method setting (Options (Options): Bit 0 to 2) is set to "0: Acceleration/deceleration specification method (mcAccDec)". Jerk (Jerk) is not used if the acceleration/deceleration method is set to "1: Acceleration/deceleration time-fixed method (mcFixedTime)".

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000, positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	Not used

■Velocity mode (VelocityMode)

Specifies the velocity mode used in linear interpolation control. When specifying "2: Reference axis speed (ReferenceAxisSpeed)" in Velocity mode (VelocityMode), the reference axis becomes the structuring axis specified to the first element of Linear interpolation axis (LinearAxes).

Setting value	Description
0: Vector speed (VectorSpeed)	Sets Velocity (Velocity) as the vector speed. The positioning speed (V_n) of each axis is calculated from the movement amount (D_n) of each axis by the Motion system based on the positioning speed (V) of the set control target. <Example> When using 2-axis linear interpolation control  • Structuring axis 1 movement amount (D_1): 10000.0[pulse] • Structuring axis 2 movement amount (D_2): 15000.0[pulse] • Axes group speed unit: [s] • Vector speed (V): 7000.0[pulse/s] For the above conditions, the positioning speed of each axis is calculated by the Motion system with the following calculation formulas. • Structuring axis 1 positioning speed: $V_1 = V \times D_1 / \sqrt{D_1^2 + D_2^2}$ • Structuring axis 2 positioning speed: $V_2 = V \times D_2 / \sqrt{D_1^2 + D_2^2}$ ■Point • For the vector speed specification, the set speed limit value is valid as the vector speed for Velocity (Velocity).

Setting value	Description
1: Long axis speed (LongAxisSpeed)	Sets Velocity (Velocity) as the velocity of the long axis. The control is carried out based on the positioning speed (the long axis velocity: V) of the interpolation axis whose movement amount is the largest in the specified positions set in each interpolation axis. The positioning speed (V_n) of the other interpolation axes is calculated by the Motion system from the movement amount (D_n) of each interpolation axis. <Example> When using 4-axis linear interpolation control • Structuring axis 1 movement amount (D_1): 10000.0[pulse] • Structuring axis 2 movement amount (D_2): 15000.0[pulse] • Structuring axis 3 movement amount (D_3): 5000.0[pulse] • Structuring axis 4 movement amount (D_4): 20000.0[pulse] • Structuring axis 4 speed unit: [s] • Long axis velocity (V): 7000.0[pulse/s] In the above case, the long axis is structuring axis 4 whose movement amount is the largest, and the control is carried out based on the long axis velocity of structuring axis 4. The positioning speed of the other structuring axes is calculated by the Motion system with the following calculation formulas. • Structuring axis 1 positioning speed: $V_1 = D_1 / D_4 \times V$ • Structuring axis 2 positioning speed: $V_2 = D_2 / D_4 \times V$ • Structuring axis 3 positioning speed: $V_3 = D_3 / D_4 \times V$ ■Point • For the long axis speed specification, the set speed limit value is valid as the long axis velocity for Velocity (Velocity). • Note that the vector speed may be larger than the speed limit value when specifying Long axis speed.
2: Reference axis speed (ReferenceAxisSpeed)	Sets Velocity (Velocity) as the velocity of the reference axis. The Motion system calculates the positioning speed (V_n) of the other interpolation axes from the movement amount (D_n) of each interpolation axis and performs control based on the positioning speed of the set reference axis (the reference axis velocity: V). The axis set in the first element of Linear interpolation axis (LinearAxes) becomes the reference axis used as the interpolation axis. If "0.0" is set for the movement amount of the interpolation axis specified as the reference axis, "Reference Axis Movement Amount 0 (error code: 1AABH)" occurs. <Example> When using 4-axis linear interpolation control • Structuring axis 1 movement amount (D_1): 10000.0[pulse] • Structuring axis 2 movement amount (D_2): 15000.0[pulse] • Structuring axis 3 movement amount (D_3): 5000.0[pulse] • Structuring axis 4 movement amount (D_4): 20000.0[pulse] • Reference axis velocity (V): 7000.0[pulse/s] In the above case, the long axis is structuring axis 4 whose movement amount is the largest, and the control is carried out based on the positioning speed set for structuring axis 4. The positioning speed of the other structuring axes is calculated by the Motion system with the following calculation formulas. • Structuring axis 1 positioning speed: $V_1 = D_1 / D_4 \times V$ • Structuring axis 2 positioning speed: $V_2 = D_2 / D_4 \times V$ • Structuring axis 3 positioning speed: $V_3 = D_3 / D_4 \times V$ ■Point • For the reference axis speed specification, the set speed limit value is valid as the reference axis velocity for Velocity (Velocity). • Note that the positioning speed may be larger than the set reference axis velocity for axes with a movement amount larger than the reference axis.

Restriction

- When "0: Vector speed (VectorSpeed)" is set in Velocity mode (VelocityMode) and the movement value of each axis exceeds "4,294,967,296.0(=2^{32})", "Out of Linear Movement Amount Range (error code: 1ADEH)" will occur and the axis will not start.
- When "0: Aborting (mcAborting)" is set in Buffer mode (BufferMode) and MCv_MoveLinearInterpolateAbsolute (Absolute Value Linear Interpolation Control) is executed by multiple start, specify "0: Vector speed (VectorSpeed)" in Velocity mode (VelocityMode). If a value other than "0: Vector speed (VectorSpeed)" is set, "Out of Speed Mode Range (error code: 1A61H)" occurs.

■Direction selection (Direction)

Specifies the direction used to move from the current position to the target position by selecting positive direction, negative direction, or shortest way. Direction selection (Direction) has 16 array elements.

- Direction [0..15] indicates the direction selection of structuring axis 1 to 16. Set the value for the structuring axis set in Linear interpolation axis (LinearAxes).
- The direction input of the structuring axis which is not set as the linear interpolation axis is ignored.
- Direction selection (Direction) of axes after all structuring axes set to the linear interpolation axes can be omitted.
- When a setting value is not set, "Out of Direction Selection Range (error code: 1A37H)" occurs and the axis will not start.
- A linear axis where the software stroke limit is valid ignores the setting of Direction selection (Direction). For details of operation of each direction selection, refer to the following.

☞ Page 204 Direction selection (Direction)

Setting value	Description
1: Positive direction (mcPositiveDirection)	Performs positioning in the positive direction (address increasing) from the current position to the target position.
2: Negative direction (mcNegativeDirection)	Performs positioning in the negative direction (address decreasing) from the current position to the target position.
3: Shortest path (mcShortestWay)	Based on the current position, positioning control is performed in the direction closest to the target position.

■Buffer mode (BufferMode)

Sets the operation used to perform multiple start (buffer mode).

The following buffer modes are settable in MCv_MoveLinearInterpolateAbsolute (Absolute Value Linear Interpolation Control).

Setting value	Description
0: Aborting (mcAborting)	Aborts (cancels) the under-control FB and executes the next FB immediately.
1: Buffered (mcBuffered)	Buffers the next FB on the under-control FB. If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after completion of the under-control FB.
2: BlendingLow (mcBlendingLow)	Buffers the next FB on the under-control FB.*1 If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. When the under-control FB reaches the target position, the lower target velocity between the under-control FB and the buffering FB is used as the switching speed.
3: BlendingPrevious (mcBlendingPrevious)	Buffers the next FB on the under-control FB.*1 If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. When the under-control FB reaches the target position, the switching speed changes to the target velocity of the under-control FB.
4: BlendingNext (mcBlendingNext)	Buffers the next FB on the under-control FB.*1 If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. When the under-control FB reaches the target position, the switching speed changes to the target velocity of the buffering FB.
5: BlendingHigh (mcBlendingHigh)	Buffers the next FB on the under-control FB.*1 If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. When the under-control FB reaches the target position, the switching speed changes to the higher target velocity value between the under-control FB and the buffering FB.

*1 Stops are not performed between the under-control FB and the buffering FB.

Point

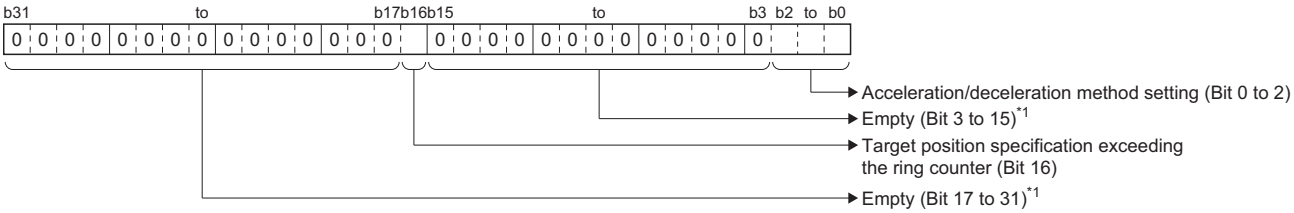
For details of multiple start (buffer mode), refer to the following.

MELSEC iQ-R Motion Module User's Manual (Application)

■Options (Options)

Sets the function options used for MCv_MoveLinearInterpolateAbsolute (Absolute Value Linear Interpolation Control) by specifying the bit.

Contents that are set by specifying the bit are shown below.



*1 Set empty areas to "0". When a value other than "0" is set, "Out of Options Range (error code: 1A4EH)" occurs.

Bit	Name	Description
0 to 2	Acceleration/deceleration method setting	Sets the acceleration/deceleration method used to perform control. • 0: Acceleration/deceleration specification method (mcAccDec) • 1: Acceleration/deceleration time-fixed method (mcFixedTime)
16	Target position specification exceeding the ring counter	Sets whether to allow a target position that exceeds the ring counter upper limit value or lower limit value when the software stroke limit is invalid. • 0: Do not allow • 1: Allow

• Acceleration/deceleration method setting (Bit 0 to 2)

Setting value	Description
0: Acceleration/deceleration specification method (mcAccDec)	An acceleration/deceleration method using Acceleration (Acceleration), Deceleration (Deceleration), and Jerk (Jerk) set in MCv_MoveLinearInterpolateAbsolute (Absolute Value Linear Interpolation Control).
1: Acceleration/deceleration time-fixed method (mcFixedTime)	An acceleration/deceleration method using the acceleration/deceleration time specified in MCv_MoveLinearInterpolateAbsolute (Absolute Value Linear Interpolation Control) regardless of the velocity. The acceleration/deceleration time does not use Acceleration (Acceleration), Deceleration (Deceleration), and Jerk (Jerk).

■Necessary objects

When using MCv_MoveLinearInterpolateAbsolute (Absolute Value Linear Interpolation Control), set the following slave objects for all structuring axes in the specified axes group.

• Target position (607AH)

If there is a structuring axis present that does not have the slave object set, "Necessary Slave Object Unset (error code: 1AA8H)" occurs and the axis will not start.

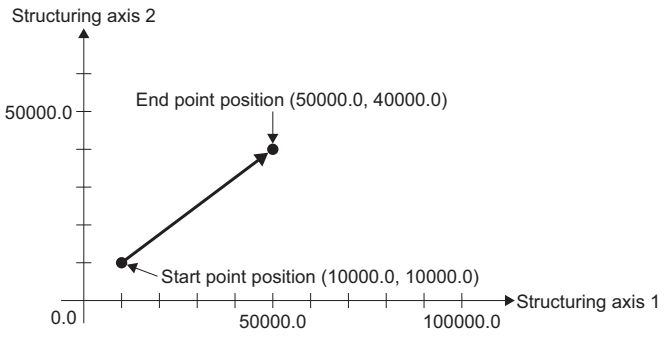
For details of slave object settings, refer to the following.

MELSEC iQ-R Motion Module User's Manual (Application)

Program example

In the following program example, the linear interpolation control start (bLinearInterpolateCMD) is set to TRUE and Axes group 1 (AxesGroup001) is enabled, following which 2-axis linear interpolation using absolute positioning control is performed for Axis group 1 (AxesGroup001) using the settings shown below.

• Operation



• Settings

Item	Setting value	
	Structuring axis 1	Structuring axis 2
Target position	50000.0	40000.0
Velocity	50000.0	
Acceleration	50000.0	
Deceleration	50000.0	
Jerk	0.0	
Direction selection	Positive direction	Positive direction

■Axes

Axis No.	Label name	Data type	Comment
1	Axis0001	AXIS_REF	Axis 1
2	Axis0002	AXIS_REF	Axis 2

■Axes group

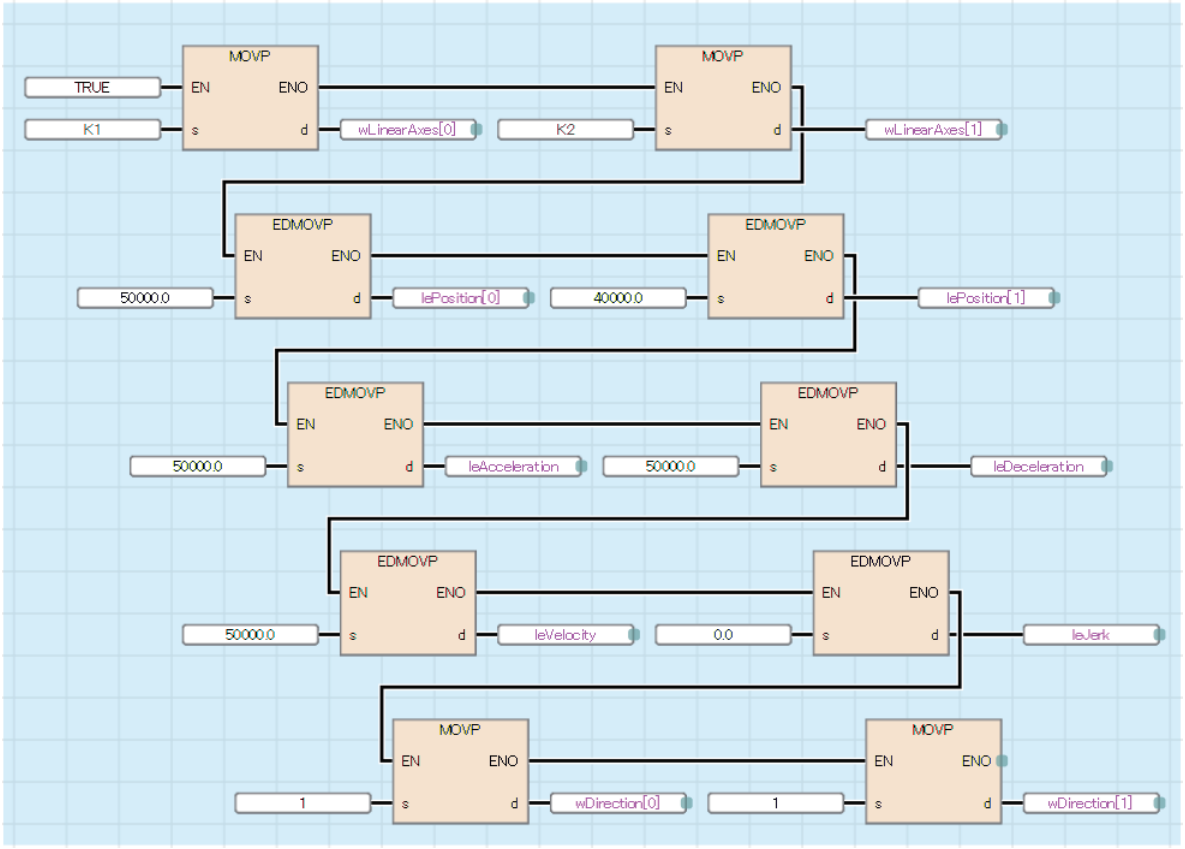
Axes group No.	Label name	Data type	Comment
1	AxesGroup001	AXES_GROUP_REF	Axes group 1

■Labels used

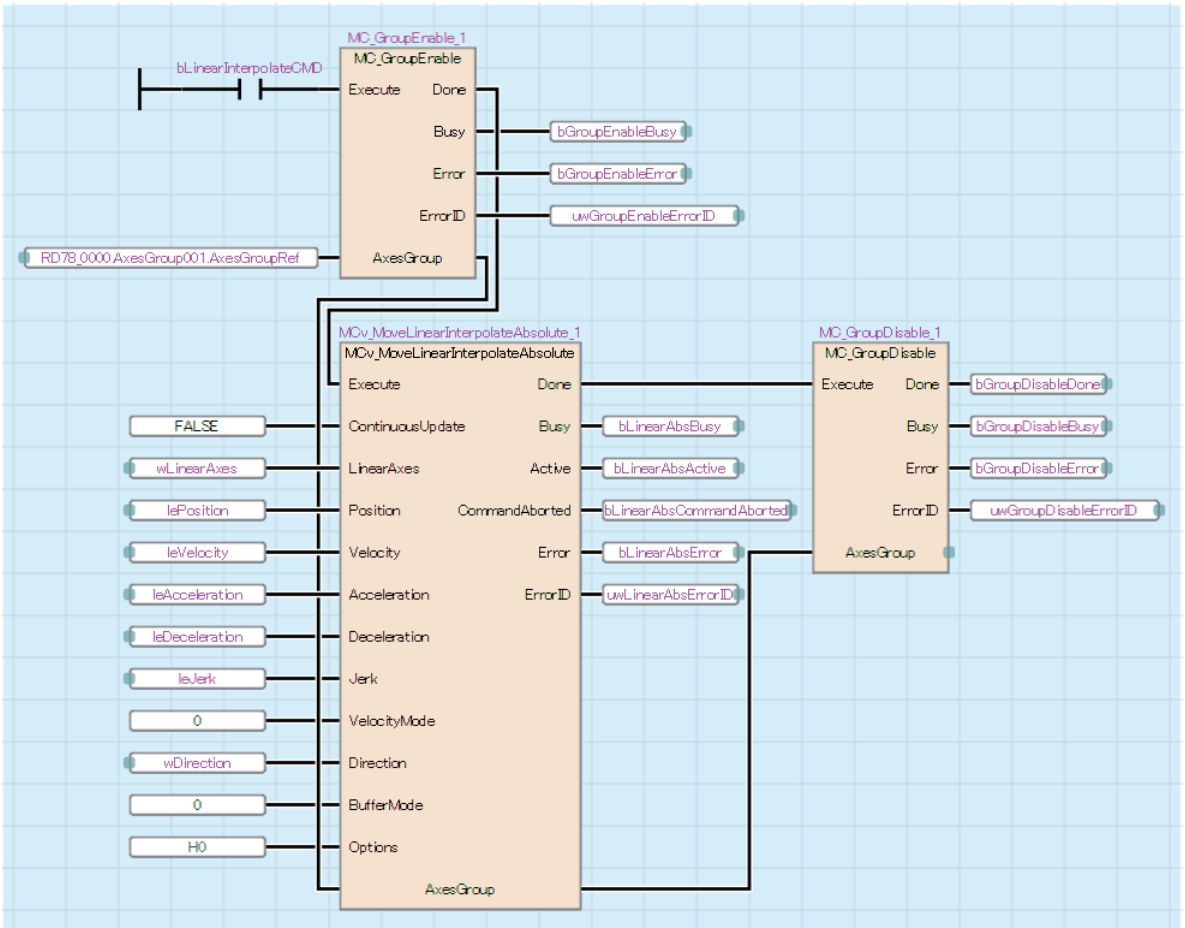
Label name	Data type	Comment
MCv_MoveLinearInterpolateAbsolute_1	MCv_MoveLinearInterpolateAbsolute	Absolute Value Linear Interpolation Control FB
wLinearAxes	Word [signed](0..15)	Linear interpolation axis
lePosition	Double-precision real number(0..15)	Target position
leVelocity	Double-precision real number	Velocity
leAcceleration	Double-precision real number	Acceleration
leDeceleration	Double-precision real number	Deceleration
leJerk	Double-precision real number	Jerk
wDirection	Word [signed](0..15)	Direction selection
bLinearAbsDone	Bit	Execution completion
bLinearAbsBusy	Bit	Executing
bLinearAbsActive	Bit	Controlling
bLinearAbsCommandAborted	Bit	Abortion of execution
bLinearAbsError	Bit	Error
uwLinearAbsErrorID	Word [unsigned]/bit string [16-bit]	Error code
MC_GroupEnable_1	MC_GroupEnable	Axes Group Enabled FB
bLinearInterpolateCMD	Bit	Linear interpolation control start
bGroupEnableDone	Bit	Axes Group Enabled Completion
bGroupEnableBusy	Bit	Executing
bGroupEnableError	Bit	Error
uwGroupEnableErrorID	Word [unsigned]/bit string [16-bit]	Error code
MC_GroupDisable_1	MC_GroupDisable	Axes Group Disabled FB
bGroupDisableDone	Bit	Axes Group Disabled Completion
bGroupDisableBusy	Bit	Executing
bGroupDisableError	Bit	Error
uwGroupDisableErrorID	Word [unsigned]/bit string [16-bit]	Error code

■FBD/LD program (CPU module side)

• Data setting for 2-axis linear interpolation control



• Axes Group Enabled/2-axis linear interpolation control/Axis Group Disabled



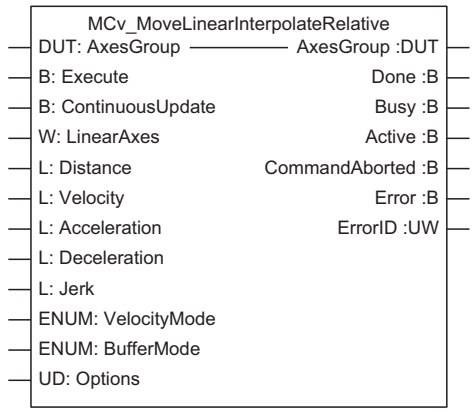
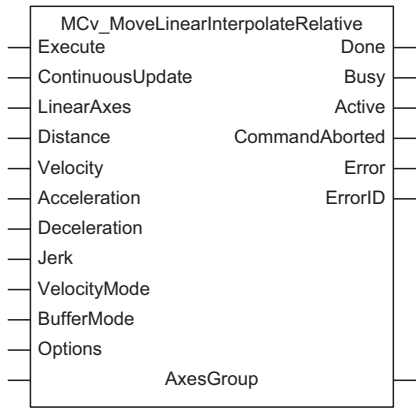
■ST program (Motion module side)

```
//-----Data setting for 2-axis linear interpolation control-----
wLinearAxes[0]:= 1;
wLinearAxes[1]:= 2;
lePosition[0]:= 50000.0;
lePosition[1]:= 40000.0;
leVelocity:= 50000.0;
leAcceleration:= 50000.0;
leDeceleration:= 50000.0;
leJerk:= 0.0;
wDirection[0]:= MC_DIRECTION__mcPositiveDirection;
wDirection[1]:= MC_DIRECTION__mcPositiveDirection;

//-----Axes Group Enabled-----
MC_GroupEnable_1(
   AxesGroup:= AxesGroup001.AxesGroupRef ,
   Execute:= bLinearInterpolateCMD ,
   Done=> bGroupEnableDone,
   Busy=> bGroupEnableBusy ,
   Error=> bGroupEnableError ,
   ErrorID=> uwGroupEnableErrorID
);

//-----2-axis linear interpolation control-----
MCv_MoveLinearInterpolateAbsolute_1(
   AxesGroup:= AxesGroup001.AxesGroupRef ,
   Execute:= bGroupEnableDone ,
   ContinuousUpdate:= FALSE ,
   LinearAxes:= wLinearAxes ,
   Position:= lePosition ,
   Velocity:= leVelocity ,
   Acceleration:= leAcceleration ,
   Deceleration:= leDeceleration ,
   Jerk:= leJerk ,
   VelocityMode:=  MC_INTERPOLATE_SPEED_MODE__VectorSpeed,
   Direction:= wDirection ,
   BufferMode:= MC_BUFFER_MODE__mcAborting ,
   Options:= H00000000 ,
   Done=> bLinearAbsDone ,
   Busy=> bLinearAbsBusy ,
   Active=> bLinearAbsActive ,
   CommandAborted=> bLinearAbsCommandAborted ,
   Error=> bLinearAbsError ,
   ErrorID=> uwLinearAbsErrorID
);

//-----Axes Group Disabled-----
MC_GroupDisable_1(
   AxesGroup:= AxesGroup001.AxesGroupRef ,
   Execute:= bLinearAbsDone ,
   Done=> bGroupDisableDone ,
   Busy=> bGroupDisableBusy ,
```

```
    Error=> bGroupDisableError ,
    ErrorID=> uwGroupDisableErrorID
);
```

MCv_MoveLinearInterpolateRelative

Specifies the movement amount of the relative position of the set axes group, then executes positioning by linear interpolation control.

Ladder	FBD/LD	ST
		MCv_MoveLinearInterpolateRelative(AxesGroup:= ?AXES_GROUP_REF? , Execute:= ?BOOL? , ContinuousUpdate:= ?BOOL? , LinearAxes:= ?INT(0..15)? , Distance:= ?LREAL(0..15)? , Velocity:= ?LREAL? , Acceleration:= ?LREAL? , Deceleration:= ?LREAL? , Jerk:= ?LREAL? , VelocityMode:= ?INT? , BufferMode:= ?INT? , Options:= ?DWORD? , Done=> ?BOOL? , Busy=> ?BOOL? , Active=> ?BOOL? , CommandAborted=> ?BOOL? , Error=> ?BOOL? , ErrorID=> ?WORD?);

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Relative Value Linear Interpolation Control	216	8	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
AxesGroup	Axes group information	AXES_GROUP_REF	At start	—	Omission not possible	This variable sets the axes group. For the variables used (AxesGroupName.AxesGroupRef.), refer to the following. ☞ Page 46 AxesGroupName.AxesGroupRef. (Axes group information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Start	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MCv_MoveLinearInterpolateRelative (Relative Value Linear Interpolation Control).
ContinuousUpdate	Continuous update	BOOL	At start	TRUE, FALSE	FALSE	This variable sets whether to enable or disable continuous change of Velocity (Velocity), Acceleration (Acceleration), and Deceleration (Deceleration). Continuous update re-imports input variables to the instance in execution without aborting the operation. • FALSE: Disable • TRUE: Enable

Input variable	Name	Data type	Input	Setting range	Default value	Description
LinearAxes	Linear interpolation axis	INT[0..15]	At start	1 to 16	0	This variable specifies the axes to be used in linear interpolation control from the structuring axes. Specifies the index No. (1 to 16) of the structuring axes in the array. *: When Velocity mode (VelocityMode) is set to "2: Reference axis speed (ReferenceAxisSpeed)", the first element of the array is regarded as the reference axis. For details, refer to the following. ☞ Page 276 Linear interpolation axis (LinearAxes)
Distance	Movement amount	LREAL[0..15]	At start	-10000000000.0 to 10000000000.0	0.0	This variable sets the movement amount from the current position at start to the end point. The target position is set as 1-dimensional array data. This data is treated as the relative position for structuring axes 1 to 16. For details, refer to the following. ☞ Page 276 Movement amount (Distance)
Velocity	Velocity	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0, 0.0001 to 2500000000.0	0.0	This variable sets the speed command value. For details, refer to the following. ☞ Page 277 Velocity (Velocity)
Acceleration	Acceleration	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the acceleration. For details, refer to the following. ☞ Page 277 Acceleration (Acceleration)
Deceleration	Deceleration	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the deceleration. For details, refer to the following. ☞ Page 277 Deceleration (Deceleration)
Jerk	Jerk	LREAL	At start	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the jerk. For details, refer to the following. ☞ Page 277 Jerk (Jerk)
VelocityMode	Speed mode	INT (MC_INTERPOLATE_SPEED_MODE)	At start	0 to 2	0	This variable sets the speed mode for interpolation control. • 0: Vector speed (VectorSpeed) • 1: Long axis speed (LongAxisSpeed) • 2: Reference axis speed (ReferenceAxisSpeed) For details, refer to the following. ☞ Page 278 Velocity mode (VelocityMode)
BufferMode	Buffer mode	INT (MC_BUFFER_MODE)	At start	0 to 5	0	This variable sets the buffer mode. • 0: Aborting (mcAborting) • 1: Buffered (mcBuffered) • 2: BlendingLow (mcBlendingLow) • 3: BlendingPrevious (mcBlendingPrevious) • 4: BlendingNext (mcBlendingNext) • 5: BlendingHigh (mcBlendingHigh) For details, refer to the following. ☞ Page 279 Buffer mode (BufferMode)
Options	Options	DWORD(HEX)	At start	00000000H to 00000005H	00000000H	This variable sets the function options for MCv_MoveLinearInterpolateRelative (Relative Value Linear Interpolation Control) by specifying the bit. For details, refer to the following. ☞ Page 280 Options (Options)

■Output variables

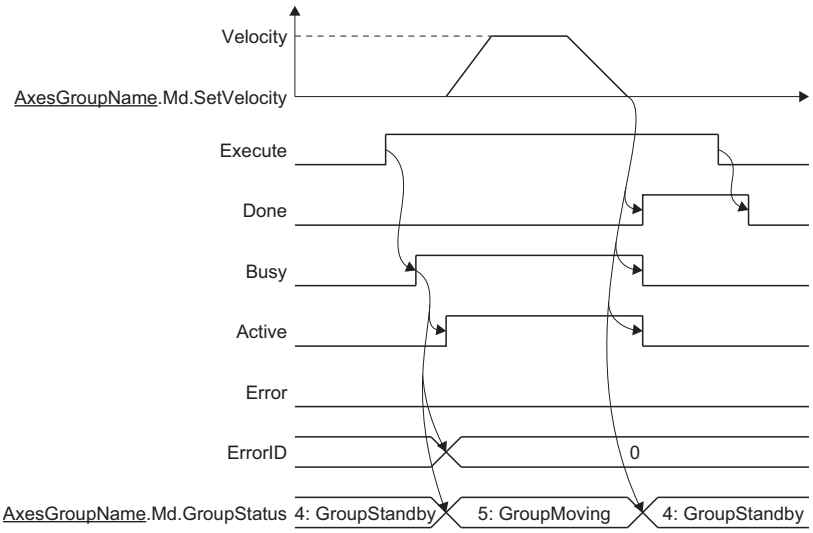
Output variable	Name	Data type	Default value	Description
Done	Completion	BOOL	FALSE	This variable becomes TRUE when the control is completed. This variable changes as shown below based on the status of Start (Execute) at operation completion. ■When Start (Execute) is TRUE This variable remains TRUE until Start (Execute) becomes FALSE. ■When Start (Execute) is FALSE This variable becomes TRUE for one cycle only.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MCv_MoveLinearInterpolateRelative (Relative Value Linear Interpolation Control) is executed.
Active	Controlling	BOOL	FALSE	This variable becomes TRUE while MCv_MoveLinearInterpolateRelative (Relative Value Linear Interpolation Control) is controlling the axis. When multiple MCv_MoveLinearInterpolateRelative (Relative Value Linear Interpolation Control) FBs are executed for the same axes group, only the Controlling (Active) of one MCv_MoveLinearInterpolateRelative (Relative Value Linear Interpolation Control) FB becomes TRUE.
CommandAborted	Abortion of execution	BOOL	FALSE	This variable becomes TRUE when execution of MCv_MoveLinearInterpolateRelative (Relative Value Linear Interpolation Control) is aborted because of an error or multiple start.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. MELSEC iQ-R Motion Module User's Manual (Application)

Processing details

- Linear interpolation control specifies an axes group and performs interpolation control so that the path from the start point (move start point) to the end point becomes a straight line. In linear interpolation control, interpolation control using a maximum of 4 axes is performed.
- In MCv_MoveLinearInterpolateRelative (Relative Value Linear Interpolation Control), specify the relative movement amount from the current position and perform relative value linear interpolation control.

■Timing chart

- When the operation completes normally



- When an error occurs

  For the timing chart when an error occurs, refer to the following.

  ☞ Page 11 Basic operation of Execute command (Execute) type Motion control FBs

■Operation overview of 1-axis linear interpolation control (relative value)

1-axis linear interpolation is performed from the current position at start (start point position) by the movement amount set in Movement amount (Distance).

The movement direction is determined by the sign (+/-) of the movement amount.

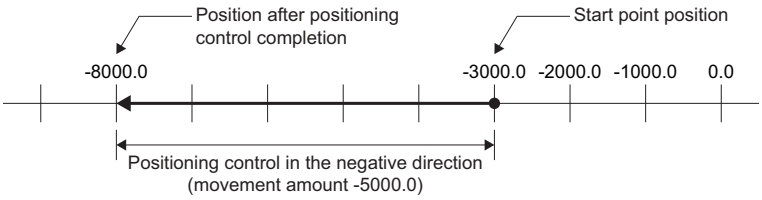
Movement amount	Description
If positive (+)	Positioning in the positive direction (address increasing direction)
If negative (-)	Positioning in the negative direction (address decreasing direction)

Ex.

When the start point position for structuring axis 1 is "-3000.0" and Movement amount (Distance) is set to "-5000.0"

LinearAxes[0]:= 1

Distance[0]:= -5000.0

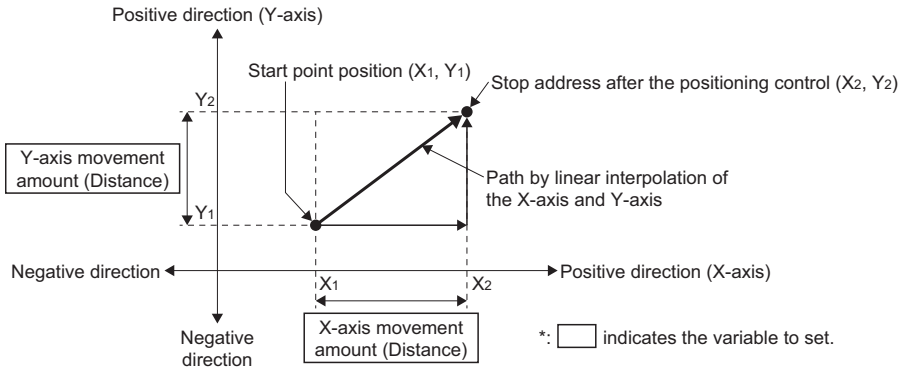


■Operation overview of 2-axis linear interpolation control (relative value)

2-axis linear interpolation is performed from the current position at start (start point position) by the movement amount set in Movement amount (Distance).

The movement direction is determined by the sign (+/-) of the movement amount.

Movement amount	Description
If positive (+)	Positioning in the positive direction (address increasing direction)
If negative (-)	Positioning in the negative direction (address decreasing direction)



Ex.

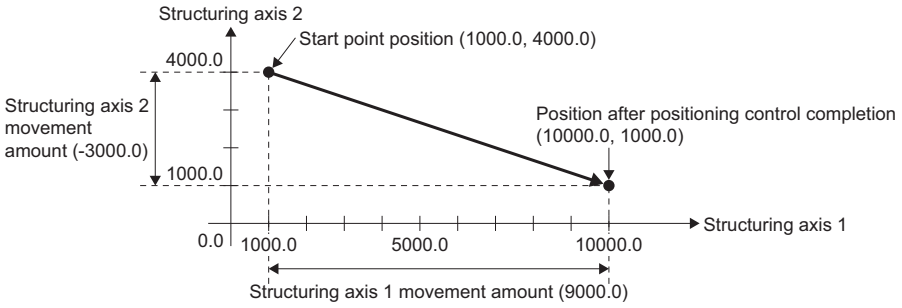
When the start point position is "Structuring axis 1: 1000.0, Structuring axis 2: 4000.0" and Movement amount (Distance) is set to "Structuring axis 1: 9000.0, Structuring axis 2: -3000.0"

LinearAxes[0]:= 1

LinearAxes[1]:= 2

Distance[0]:= 9000.0

Distance[1]:= -3000.0

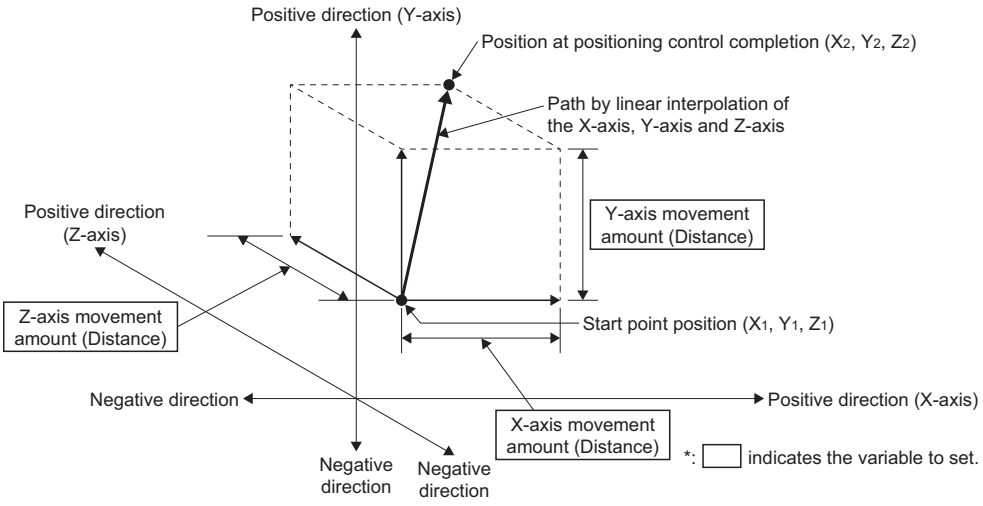


■Operation overview of 3-axis linear interpolation control (relative value)

3-axis linear interpolation is performed from the current position at start (start point position) by the movement amount set in Movement amount (Distance).

The movement direction is determined by the sign (+/-) of the movement amount.

Movement amount	Description
If positive (+)	Positioning in the positive direction (address increasing direction)
If negative (-)	Positioning in the negative direction (address decreasing direction)



Ex.

When the start position is "Structuring axis 1: 2000.0, Structuring axis 2: 1000.0, Structuring axis 3: 1000.0" and Movement amount (Distance) is set to "Structuring axis 1: 10000.0, Structuring axis 2: 5000.0, Structuring axis 3: 6000.0"

LinearAxes[0]:= 1

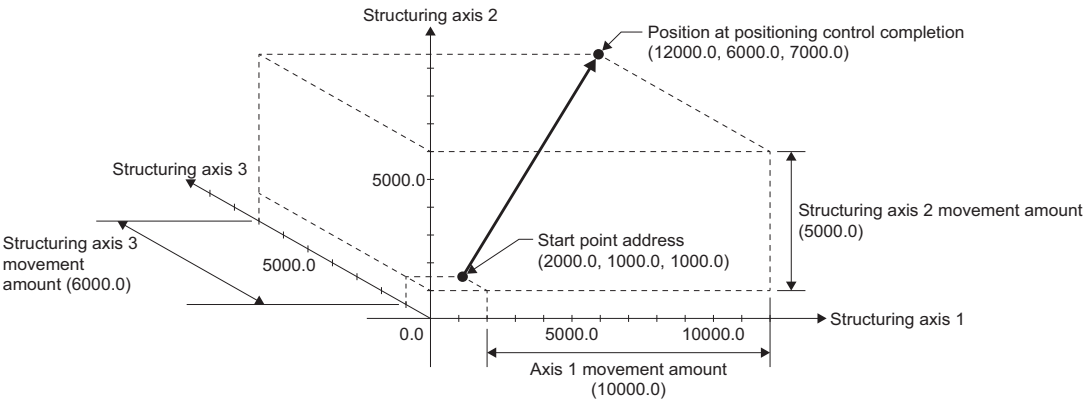
LinearAxes[1]:= 2

LinearAxes[2]:= 3

Distance[0]:= 10000.0

Distance[1]:= 5000.0

Distance[2]:= 6000.0



■Linear interpolation axis (LinearAxes)

In linear interpolation control, linear interpolation is performed using arbitrary axes from the structuring axes set to the axes group.

Set structuring axes to perform linear interpolation by Linear interpolation axis (LinearAxes) from the structuring axes of the axes group. Linear interpolation axis (LinearAxes) has 16 array elements. For the arrays, first set only the index Nos. (1 to 16: structuring axis 1 to 16) of structuring axes to be used for interpolation control to the front, then set the rest of the arrays to "0". Arrays set to "0" can be omitted.

- The setting number of the linear interpolation axes must be set to the maximum number of the interpolation axes for interpolation control (4 axes) or less.
- The number of structuring axes to be used for interpolation control must be the number of axes registered to the structuring axes or less.
- When Velocity mode (VelocityMode) is "2: Reference axis speed (ReferenceAxisSpeed)", the structuring axis of the first element in Linear interpolation axis (LinearAxes) becomes the reference axis.
- When specifying the following, "Linear Interpolation Axis Setting Incorrect (error code: 1AB9H)" occurs and the axis will not start.
  - When specifying an structuring axis where an axis is not set
  - When the first element of Linear interpolation axis (LinearAxes) is "0"
  - When specifying the same index No. to Linear interpolation axis (LinearAxes) in duplicate
  - When the number of axes set in Linear interpolation axis (LinearAxes) exceeds the maximum number of interpolation axes

Ex.

When structuring axis 2, 3, and 4 are set to the linear interpolation axis

LinearAxes[0]:= 2;
LinearAxes[1]:= 3;
LinearAxes[2]:= 4;
LinearAxes[3]:= 0 ;*1
⋮
LinearAxes[15]:= 0; *1

*1 "LinearAxes[3]:= 0; to LinearAxes[15]:= 0;" can be omitted.

■Movement amount (Distance)

Sets the movement amount from the current position at start to the end point. Movement amount (Distance) has 16 array elements.

- Distance[0..15] indicates the movement amount of structuring axis 1 to 16. Set the position in relation to the structuring axes set in Linear interpolation axis (LinearAxes).
- Movement amount (Distance) of structuring axes not set to the linear interpolation axes is ignored.
- Movement amount (Distance) of axes after all structuring axes set to the linear interpolation axes can be omitted.

Ex.

When setting structuring axis 2, 3, 4 to the linear interpolation axes, and setting the target positions of structuring axis 2, 3, and 4 to 2000.0, 3000.0, and 4000.0 respectively

LinearAxes[0]:= 2;
LinearAxes[1]:= 3;
LinearAxes[2]:= 4;
LinearAxes[3]:= 0; *1
⋮
LinearAxes[15]:= 0; *1
Position[0]:= 2000.0;
Position[1]:= 3000.0;
Position[2]:= 4000.0;
Position[3]:= 0.0; *1
⋮
Position[15]:= 0.0; *1

*1 "LinearAxes[3]:= 0; to LinearAxes[15]:= 0;" and "Position[3]:= 0.0; to Position[15]:= 0.0;" can be omitted.

■Velocity (Velocity)

Sets the maximum speed of the path used in MCv_MoveLinearInterpolateRelative (Relative Value Linear Interpolation Control).

Setting range
0.0, 0.0001 to 2500000000.0[*1][*2]

*1 To perform floating-point operation, the lower limit value of the commanded velocity is limited by the following restrictions. If the velocity that is converted from the target velocity into the operation cycle is less than "0.00001", "Out of Calculation Cycle Converted Speed Range (error code: 1AE4H)" occurs. (If the velocity is changed, "Out of Calculation Cycle Converted Speed Range Warning (warning code: 0D2FH)" occurs instead). To improve the precision of floating-point operation, change Position command unit (AxisName.Pr.Unit_Position) and Speed command unit (AxisName.Pr.Unit_Velocity) and perform settings so that the velocity converted into the operation cycle is not less than "0.00001".

*2 When a Motion control FB with a set velocity of "0.0" performs a multiple start, the set velocity of the previous Motion control FB is used.

■Acceleration (Acceleration)

Sets the acceleration used in MCv_MoveLinearInterpolateRelative (Relative Value Linear Interpolation Control).

The setting range differs depending on the setting for Acceleration/deceleration method setting (Options (Options): Bit 0 to 2).

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000[*1], positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	0.000000[*1], positive numbers from 0.000001 to 8400.0[s]

*1 The operation performed at start varies depending on the setting of Operation selection at start acceleration/deceleration 0 (AxisName.Pr.AccelerationZeroBehavior). When changing the acceleration or deceleration, the changes are not accepted.

■Deceleration (Deceleration)

Sets the deceleration used in MCv_MoveLinearInterpolateRelative (Relative Value Linear Interpolation Control).

Specifies the setting range if Acceleration/deceleration method setting (Options (Options): Bit 0 to 2) is set to "0: Acceleration/deceleration specification method (mcAccDec)". Deceleration (Deceleration) is not used if the acceleration/deceleration method is set to "1: Acceleration/deceleration time-fixed method (mcFixedTime)".

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000[*1], positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	Not used

*1 The operation performed at start varies depending on the setting of Operation selection at start acceleration/deceleration 0 (AxisName.Pr.AccelerationZeroBehavior). When changing the acceleration or deceleration, the changes are not accepted.

■Jerk (Jerk)

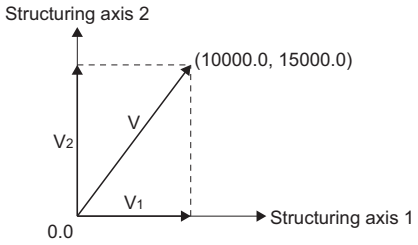
Sets the jerk used in MCv_MoveLinearInterpolateRelative (Relative Value Linear Interpolation Control).

Specifies the setting range if Acceleration/deceleration method setting (Options (Options): Bit 0 to 2) is set to "0: Acceleration/deceleration specification method (mcAccDec)". Jerk (Jerk) is not used if the acceleration/deceleration method is set to "1: Acceleration/deceleration time-fixed method (mcFixedTime)".

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000, positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	Not used

■Velocity mode (VelocityMode)

Specifies the velocity mode used in linear interpolation control. When specifying "2: Reference axis speed (ReferenceAxisSpeed)" in Velocity mode (VelocityMode), the reference axis becomes the structuring axis specified to the first element of Linear interpolation axis (LinearAxes).

Setting value	Description
0: Vector speed (VectorSpeed)	Sets Velocity (Velocity) as the vector speed. The positioning speed (V_n) of each axis is calculated from the movement amount (D_n) of each axis by the Motion system based on the positioning speed (V) of the set control target. <Example> When using 2-axis linear interpolation control  • Structuring axis 1 movement amount (D_1): 10000.0[pulse] • Structuring axis 2 movement amount (D_2): 15000.0[pulse] • Axes group speed unit: [s] • Vector speed (V): 7000.0[pulse/s] For the above conditions, the positioning speed of each axis is calculated by the Motion system with the following calculation formulas. • Structuring axis 1 positioning speed: $V_1 = V \times D_1 / \sqrt{D_1^2 + D_2^2}$ • Structuring axis 2 positioning speed: $V_2 = V \times D_2 / \sqrt{D_1^2 + D_2^2}$ ■Point • For the vector speed specification, the set speed limit value is valid as the vector speed for Velocity (Velocity).
1: Long axis speed (LongAxisSpeed)	Sets Velocity (Velocity) as the velocity of the long axis. The control is carried out based on the positioning speed (the long axis velocity: V) of the interpolation axis whose movement amount is the largest in the specified positions set in each interpolation axis. The positioning speed (V_n) of the other interpolation axes is calculated by the Motion system from the movement amount (D_n) of each interpolation axis. <Example> When using 4-axis linear interpolation control • Structuring axis 1 movement amount (D_1): 10000.0[pulse] • Structuring axis 2 movement amount (D_2): 15000.0[pulse] • Structuring axis 3 movement amount (D_3): 5000.0[pulse] • Structuring axis 4 movement amount (D_4): 20000.0[pulse] • Structuring axis 4 speed unit: [s] • Long axis velocity (V): 7000.0[pulse/s] In the above case, the long axis is structuring axis 4 whose movement amount is the largest, and the control is carried out based on the long axis velocity of structuring axis 4. The positioning speed of the other structuring axes is calculated by the Motion system with the following calculation formulas. • Structuring axis 1 positioning speed: $V_1 = D_1 / D_4 \times V$ • Structuring axis 2 positioning speed: $V_2 = D_2 / D_4 \times V$ • Structuring axis 3 positioning speed: $V_3 = D_3 / D_4 \times V$ ■Point • For the long axis speed specification, the set speed limit value is valid as the long axis velocity for Velocity (Velocity). • Note that the vector speed may be larger than the speed limit value when specifying Long axis speed.

Setting value	Description
2: Reference axis speed (ReferenceAxisSpeed)	Sets Velocity (Velocity) as the velocity of the reference axis. The Motion system calculates the positioning speed (V_n) of the other interpolation axes from the movement amount (D_n) of each interpolation axis and performs control based on the positioning speed of the set reference axis (the reference axis velocity: V). The axis set in the first element of Linear interpolation axis (LinearAxes) becomes the reference axis used as the interpolation axis. If "0" is set for the movement amount of the interpolation axis specified as the reference axis, "Reference Axis Movement Amount 0 (error code: 1AABH)" occurs. <Example> When using 4-axis linear interpolation control • Structuring axis 1 movement amount(D_1): 10000.0[pulse] • Structuring axis 2 movement amount (D_2): 15000.0[pulse] • Structuring axis 3 movement amount (D_3): 5000.0[pulse] • Structuring axis 4 movement amount (D_4): 20000.0[pulse] • Reference axis velocity (V): 7000.0[pulse/s] In the above case, the long axis is structuring axis 4 whose movement amount is the largest, and the control is carried out based on the positioning speed set for structuring axis 4. The positioning speed of the other structuring axes is calculated by the Motion system with the following calculation formulas. • Structuring axis 1 positioning speed: $V_1 = D_1 / D_4 \times V$ • Structuring axis 2 positioning speed: $V_2 = D_2 / D_4 \times V$ • Structuring axis 3 positioning speed: $V_3 = D_3 / D_4 \times V$ ■Point • For the reference axis speed specification, the set speed limit value is valid as the reference axis velocity for Velocity (Velocity). • Note that the positioning speed may be larger than the set reference axis velocity for axes with a movement amount larger than the reference axis.

Restriction

• When "0: Vector speed (VectorSpeed)" is set in Velocity mode (VelocityMode) and the movement value of each axis exceeds "4,294,967,296.0(=2^{32})", "Out of Linear Movement Amount Range (error code: 1ADEH)" will occur and the axis will not start.

■Buffer mode (BufferMode)

Sets the operation used to perform multiple start (buffer mode).

The following buffer modes are settable in MCv_MoveLinearInterpolateRelative (Relative Value Linear Interpolation Control).

Setting value	Description
0: Aborting (mcAborting)	Aborts (cancels) the under-control FB and executes the next FB immediately.
1: Buffered (mcBuffered)	Buffers the next FB on the under-control FB. If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after completion of the under-control FB.
2: BlendingLow (mcBlendingLow)	Buffers the next FB on the under-control FB.*1 If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. When the under-control FB reaches the target position, the lower target velocity between the under-control FB and the buffering FB is used as the switching speed.
3: BlendingPrevious (mcBlendingPrevious)	Buffers the next FB on the under-control FB.*1 If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. When the under-control FB reaches the target position, the switching speed changes to the target velocity of the under-control FB.
4: BlendingNext (mcBlendingNext)	Buffers the next FB on the under-control FB.*1 If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. When the under-control FB reaches the target position, the switching speed changes to the target velocity of the buffering FB.
5: BlendingHigh (mcBlendingHigh)	Buffers the next FB on the under-control FB.*1 If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. When the under-control FB reaches the target position, the switching speed changes to the higher target velocity value between the under-control FB and the buffering FB.

*1 Stops are not performed between the under-control FB and the buffering FB.

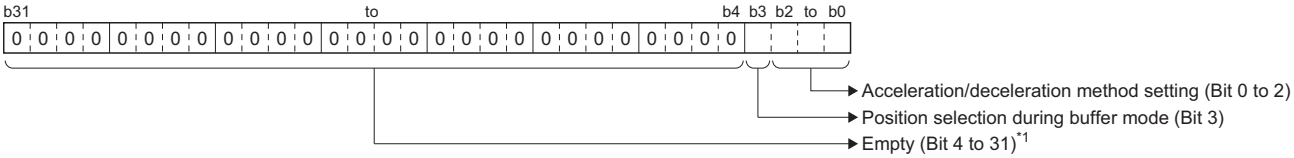
Point

For details of multiple start (buffer mode), refer to the following.
MELSEC iQ-R Motion Module User's Manual (Application)

■Options (Options)

Sets the function options used for MCv_MoveLinearInterpolateRelative (Relative Value Linear Interpolation Control) by specifying the bit.

Contents that are set by specifying the bit are shown below.



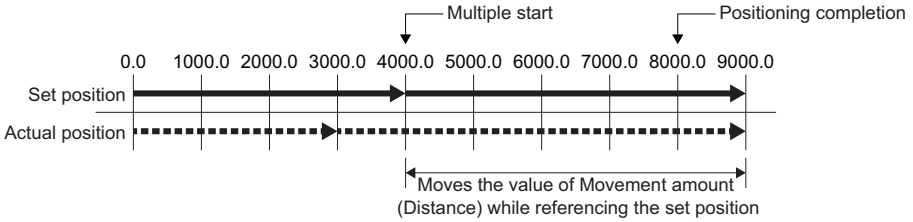
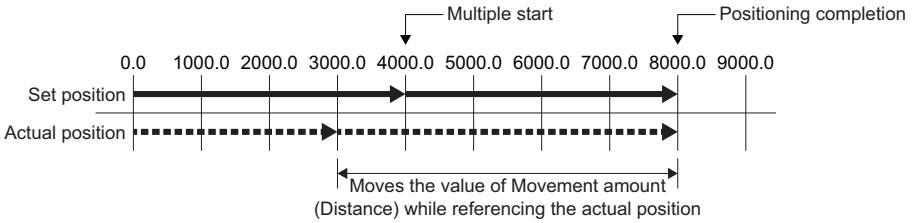
*1 Set empty areas to "0". When a value other than "0" is set, "Out of Options Range (error code: 1A4EH)" occurs.

Bit	Name	Description
0 to 2	Acceleration/deceleration method setting	Sets the acceleration/deceleration method used to perform control. • 0: Acceleration/deceleration specification method (mcAccDec) • 1: Acceleration/deceleration time-fixed method (mcFixedTime)
3	Position selection during buffer mode	Sets the position when executing relative positioning control by multiple start. • 0: Set position • 1: Actual position *: This bit is valid when setting "0: Aborting (mcAborting)" in Buffer mode (BufferMode).

• Acceleration/deceleration method setting (Bit 0 to 2)

Setting value	Description
0: Acceleration/deceleration specification method (mcAccDec)	An acceleration/deceleration method using Acceleration (Acceleration), Deceleration (Deceleration), and Jerk (Jerk) set in MCv_MoveLinearInterpolateRelative (Relative Value Linear Interpolation Control).
1: Acceleration/deceleration time-fixed method (mcFixedTime)	An acceleration/deceleration method using the acceleration/deceleration time specified in MCv_MoveLinearInterpolateRelative (Relative Value Linear Interpolation Control) regardless of the velocity. The acceleration/deceleration time does not use Acceleration (Acceleration), Deceleration (Deceleration), and Jerk (Jerk).

• Position selection during buffer mode (Bit 3)

Setting value	Description
0: Set position	Performs relative positioning control from the set position. <Example> When multiple start is performed with Movement amount (Distance) set to "5000.0" and Options (Options) set to "00000000H (Bit 3 is 0: Set position)" 
1: Actual position	Performs relative positioning control from the actual position. <Example> When multiple start is performed with the movement amount (Distance) set to "5000.0" and Options (Options) set to "00000008H (Bit 3 is 1: Actual position)". 

■Necessary objects

When using MCv_MoveLinearInterpolateRelative (Relative Value Linear Interpolation Control), set the following slave objects for all structuring axes in the specified axes group.

• Target position (607AH)

If there is a structuring axis present that does not have the slave object set, "Necessary Slave Object Unset (error code: 1AA8H)" occurs and the axis will not start.

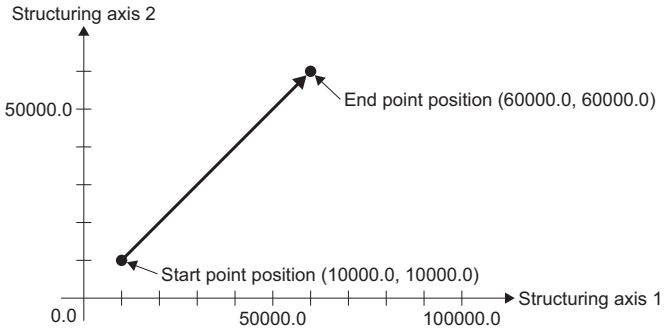
For details of slave object settings, refer to the following.

MELSEC iQ-R Motion Module User's Manual (Application)

Program example

In the following program example, the linear interpolation control start (bLinearInterpolateCMD) is set to TRUE and Axes group 1 (AxesGroup001) is enabled, following which 2-axis linear interpolation using relative positioning control is performed for Axis group 1 (AxesGroup001) using the settings shown below.

• Operation



• Setting

Item	Setting value	
	Structuring axis 1	Structuring axis 2
Target position	50000.0	50000.0
Velocity	50000.0	
Acceleration	50000.0	
Deceleration	50000.0	
Jerk	0.0	

■Axes

Axis No.	Label name	Data type	Comment
1	Axis0001	AXIS_REF	Axis 1
2	Axis0002	AXIS_REF	Axis 2

■Axes group

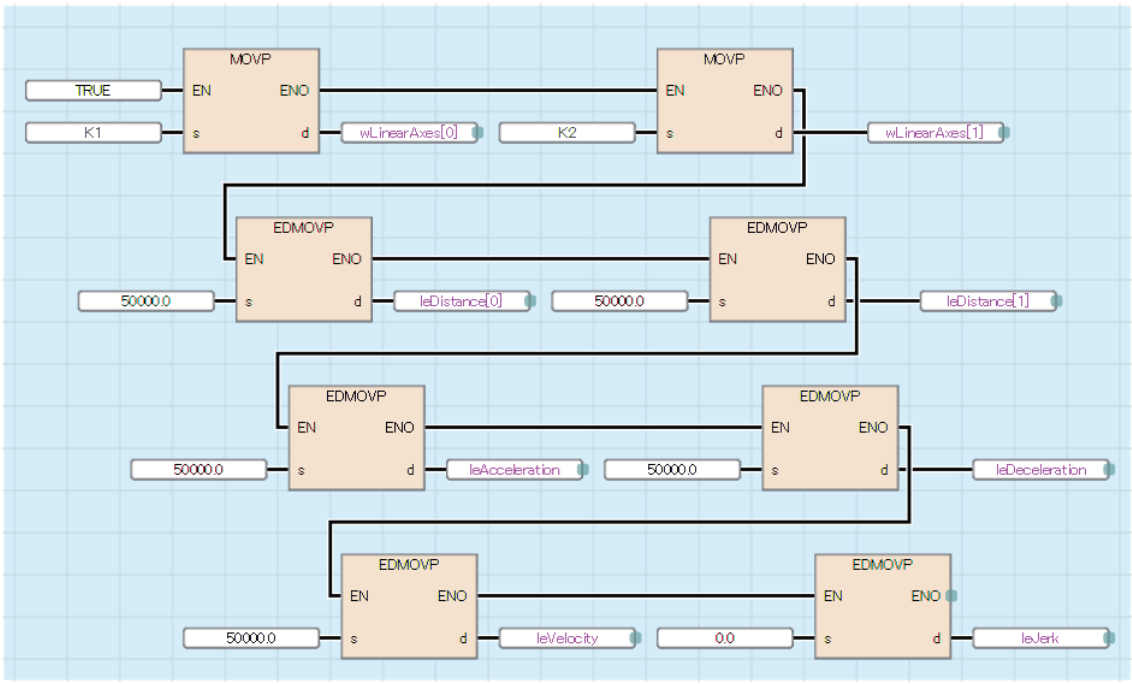
Axes group No.	Label name	Data type	Comment
1	AxesGroup001	AXES_GROUP_REF	Axes group 1

■Labels used

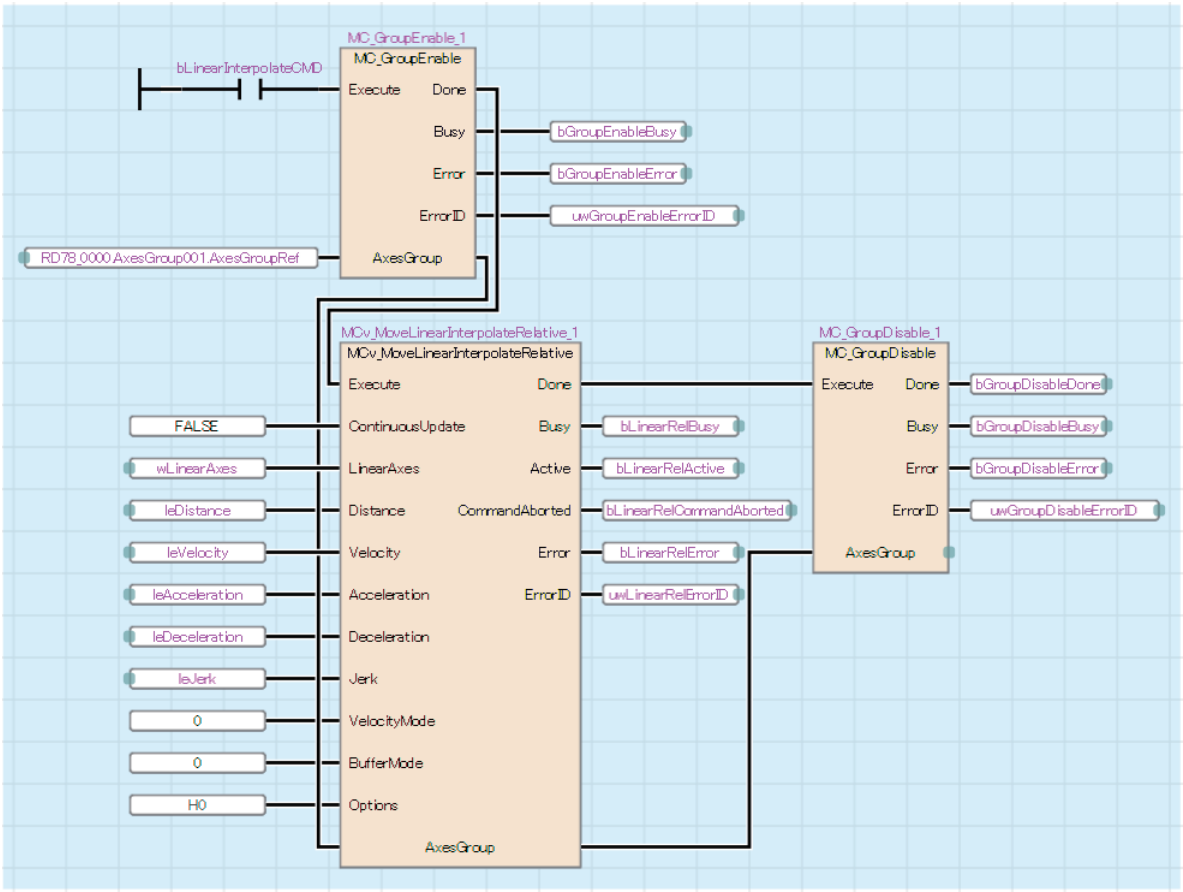
Label name	Data type	Comment
MCv_MoveLinearInterpolateRelative_1	MCv_MoveLinearInterpolateRelative	Relative Value Linear Interpolation Control FB
wLinearAxes	Word [signed](0..15)	Linear interpolation axis
leDistance	Double-precision real number	Movement amount
leVelocity	Double-precision real number	Velocity
leAcceleration	Double-precision real number	Acceleration
leDeceleration	Double-precision real number	Deceleration
leJerk	Double-precision real number	Jerk
bLinearRelDone	Bit	Execution completion
bLinearRelBusy	Bit	Executing
bLinearRelActive	Bit	Controlling
bLinearCommandAborted	Bit	Abortion of execution
bLinearRelError	Bit	Error
uwLinearRelErrorID	Word [unsigned]/bit string [16-bit]	Error code
MC_GroupEnable_1	MC_GroupEnable	Axes Group Enabled FB
bLinearInterpolateCMD	Bit	Linear interpolation control start
bGroupEnableDone	Bit	Axes Group Enabled Completion
bGroupEnableBusy	Bit	Executing
bGroupEnableError	Bit	Error
uwGroupEnableErrorID	Word [unsigned]/bit string [16-bit]	Error code
MC_GroupDisable_1	MC_GroupDisable	Axes Group Disabled FB
bGroupDisableDone	Bit	Axes Group Disabled Completion
bGroupDisableBusy	Bit	Executing
bGroupDisableError	Bit	Error
uwGroupDisableErrorID	Word [unsigned]/bit string [16-bit]	Error code

■FBD/LD program (CPU module side)

• Data setting for 2-axis linear interpolation control



• Axes Group Enabled/2-axis linear interpolation control/Axis Group Disabled



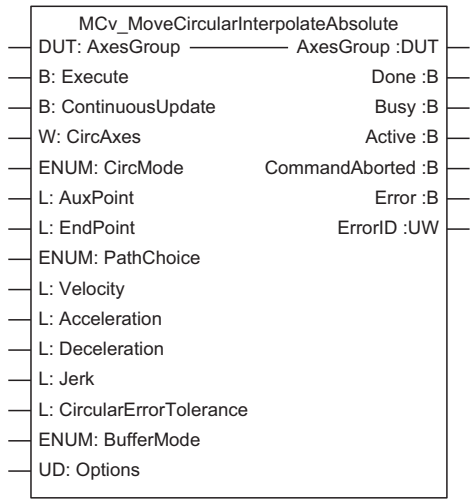
■ST program (Motion module side)

```
//-----Data setting for 2-axis linear interpolation control-----
wLinearAxes[0]:= 1;
wLinearAxes[1]:= 2;
leDistance[0]:= 50000.0;
leDistance[1]:= 50000.0;
leVelocity:= 50000.0;
leAcceleration:= 50000.0;
leDeceleration:= 50000.0;
leJerk:= 0.0;

//-----Axes Group Enabled-----
MC_GroupEnable_1(
   AxesGroup:= AxesGroup001.AxesGroupRef ,
   Execute:= bLinearInterpolateCMD,
   Done=> bGroupEnableDone,
   Busy=> bGroupEnableBusy ,
   Error=> bGroupEnableError ,
   ErrorID=> uwGroupEnableErrorID
);

//-----2-axis linear interpolation control-----
MCv_MoveLinearInterpolateRelative_1(
   AxesGroup:= AxesGroup001.AxesGroupRef ,
   Execute:= bGroupEnableDone ,
   ContinuousUpdate:= FALSE ,
   LinearAxes:= wLinearAxes ,
   Distance:= leDistance ,
   Velocity:= leVelocity ,
   Acceleration:= leAcceleration ,
   Deceleration:= leDeceleration ,
   Jerk:= leJerk ,
   VelocityMode:= MC_INTERPOLATE_SPEED_MODE__VectorSpeed ,
   BufferMode:= MC_BUFFER_MODE__mcAborting ,
   Options:= H00000000 ,
   Done=> bLinearRelDone ,
   Busy=> bLinearRelBusy ,
   Active=> bLinearRelActive ,
   CommandAborted=> bLinearRelCommandAborted ,
   Error=> bLinearRelError ,
   ErrorID=> uwLinearRelErrorID
);

//-----Axes Group Disabled-----
MC_GroupDisable_1(
   AxesGroup:= AxesGroup001.AxesGroupRef ,
   Execute:= bLinearRelDone ,
   Done=> bGroupDisableDone ,
   Busy=> bGroupDisableBusy ,
   Error=> bGroupDisableError ,
   ErrorID=> uwGroupDisableErrorID
);
```

MCv_MoveCircularInterpolateAbsolute

Executes positioning with 2-axis circular interpolation using the structuring axes of the set axes group by setting the end point and sub point of the absolute position.

Ladder



FBD/LD

ST

```
MCv_MoveCircularInterpolateAbsolute(
  AxesGroup:= ?AXES_GROUP_REF? ,
  Execute:= ?BOOL? ,
  ContinuousUpdate:= ?BOOL? ,
  CircAxes:= ?INT(0..1)? ,
  CircMode:= ?INT? ,
  AuxPoint:= ?LREAL(0..15)? ,
  EndPoint:= ?LREAL(0..15)? ,
  PathChoice:= ?INT? ,
  Velocity:= ?LREAL? ,
  Acceleration:= ?LREAL? ,
  Deceleration:= ?LREAL? ,
  Jerk:= ?LREAL? ,
  CircularErrorTolerance:= ?LREAL? ,
  BufferMode:= ?INT? ,
  Options:= ?DWORD? ,
  Done=> ?BOOL? ,
  Busy=> ?BOOL? ,
  Active=> ?BOOL? ,
  CommandAborted=> ?BOOL? ,
  Error=> ?BOOL? ,
  ErrorID=> ?WORD?
);
```

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Absolute Value Circular Interpolation Control	328	8	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
AxesGroup	Axes group information	AXES_GROUP_REF	At start	—	Omission not possible	This variable sets the axes group. For the variables used (AxesGroupName.AxesGroupRef.), refer to the following. ☞ Page 46 AxesGroupName.AxesGroupRef. (Axes group information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Start	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MCv_MoveCircularInterpolateAbsolute (Absolute Value Circular Interpolation Control).
ContinuousUpdate	Continuous update	BOOL	At start	TRUE, FALSE	FALSE	This variable sets whether to enable or disable continuous change of Velocity (Velocity), Acceleration (Acceleration), and Deceleration (Deceleration). Continuous update re-imports input variables to the instance in execution without aborting the operation. • FALSE: Disable • TRUE: Enable

Input variable	Name	Data type	Input	Setting range	Default value	Description
CircAxes	Circular interpolation axis	INT[0..1]	At start	1 to 16	0	This variable specifies the axes to be used in circular interpolation control from the structuring axes. Specifies the index No. (1 to 16) of the structuring axes in the array. The first element of the array is regarded as the reference axis. For details, refer to the following. ☞ Page 288 Circular interpolation axis (CircAxes)
CircMode	Circular interpolation mode	INT (MC_CIRC_MODE)	At start	0 to 2	0	This variable sets the circular interpolation designation method. • 0: Border point specification (mcBorder) • 1: Center point specification (mcCenter) • 2: Radius specification (mcRadius) For details, refer to the following. ☞ Page 289 Circular interpolation mode (CircMode)
AuxPoint	Sub point	LREAL[0..15]	At start	■When "0: Border point specification (mcBorder)" or "1: Center point specification (mcCenter)": -10000000000.0 to 10000000000.0 ■When "2: Radius specification (mcRadius)": 0.000001 to 2147483647.0	0.0	Sets the absolute position of the sub point (border point, center point, and radius) according to unit of the axis. The absolute position is set as 1-dimensional array data. This data changes as shown below based on the setting of Circular interpolation mode (CircMode). ■When Circular interpolation mode (CircMode) is set to "0: Border point specification (mcBorder)" or "1: Center point specification (mcCenter)": This data is treated as the absolute position for structuring axes 1 to 16. ■When Circular interpolation mode (CircMode) is set to "2: Radius specification (mcRadius)": Treats the first element as the radius and ignores the second and subsequent elements. For details, refer to the following. ☞ Page 292 Sub point (AuxPoint)
EndPoint	End point	LREAL[0..15]	At start	-10000000000.0 to 10000000000.0	0.0	This variable sets the position of the end point. The position is set as 1-dimensional array data. This data is treated as the absolute position for structuring axes 1 to 16. For details, refer to the following. ☞ Page 292 End point (EndPoint)
PathChoice	Path selection	INT (MC_CIRC_PATHCHOICE)	At start	0 to 5	0	This variable sets the rotation direction of the circular interpolation. • 0: CW (mcCW) • 1: CCW (mcCCW) • 2: Shortcut (mcShortWay) • 3: Detour (mcLongWay) • 4: CW Detour (mcCWLongWay) • 5: CCW Detour (mcCCWLongWay) *: When specifying "0: Border point specification (mcBorder)" in Circular interpolation mode (CircMode), the input is ignored. *: When specifying "2: Radius specification (mcRadius)" in Circular interpolation mode (CircMode), "0: mcCW" means the CW shortcut and "1: mcCCW" means the CCW shortcut. For details, refer to the following. ☞ Page 293 Path selection (PathChoice)
Velocity	Velocity	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0, 0.0001 to 2500000000.0	0.0	This variable sets the speed command value. For details, refer to the following. ☞ Page 294 Velocity (Velocity)

Input variable	Name	Data type	Input	Setting range	Default value	Description
Acceleration	Acceleration	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the acceleration. For details, refer to the following. ☞ Page 294 Acceleration (Acceleration)
Deceleration	Deceleration	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the deceleration. For details, refer to the following. ☞ Page 295 Deceleration (Deceleration)
Jerk	Jerk	LREAL	At start	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the jerk. For details, refer to the following. ☞ Page 295 Jerk (Jerk)
CircularErrorTolerance	Circular interpolation error tolerance	LREAL	At start	0.000001 to 100000.0	100.0	This variable sets the allowable range for circular interpolation errors. For details, refer to the following. ☞ Page 295 Circular interpolation error tolerance (CircularErrorTolerance)
BufferMode	Buffer mode	INT (MC_BUFFER_MODE)	At start	0 to 5	0	This variable sets the buffer mode. • 0: Aborting (mcAborting) • 1: Buffered (mcBuffered) • 2: BlendingLow (mcBlendingLow) • 3: BlendingPrevious (mcBlendingPrevious) • 4: BlendingNext (mcBlendingNext) • 5: BlendingHigh (mcBlendingHigh) For details, refer to the following. ☞ Page 296 Buffer mode (BufferMode)
Options	Options	DWORD(HEX)	At start	00000000H to 00000001H	00000000H	This variable sets the function options for MCv_MoveCircularInterpolateAbsolute (Absolute Value Circular Interpolation Control) by specifying the bit. For details, refer to the following. ☞ Page 296 Options (Options)

■Output variables

Output variable	Name	Data type	Default value	Description
Done	Completion	BOOL	FALSE	This variable becomes TRUE when the control is completed. This variable changes as shown below based on the status of Start (Execute) at operation completion. ■When Start (Execute) is TRUE This variable remains TRUE until Start (Execute) becomes FALSE. ■When Start (Execute) is FALSE This variable becomes TRUE for one cycle only.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MCv_MoveCircularInterpolateAbsolute (Absolute Value Circular Interpolation Control) is executed.
Active	Controlling	BOOL	FALSE	This variable becomes TRUE while MCv_MoveCircularInterpolateAbsolute (Absolute Value Circular Interpolation Control) is controlling the axis. When multiple MCv_MoveCircularInterpolateAbsolute (Absolute Value Circular Interpolation Control) FBs are executed for the same axes group, only the Controlling (Active) of one MCv_MoveCircularInterpolateAbsolute (Absolute Value Circular Interpolation Control) FB becomes TRUE.

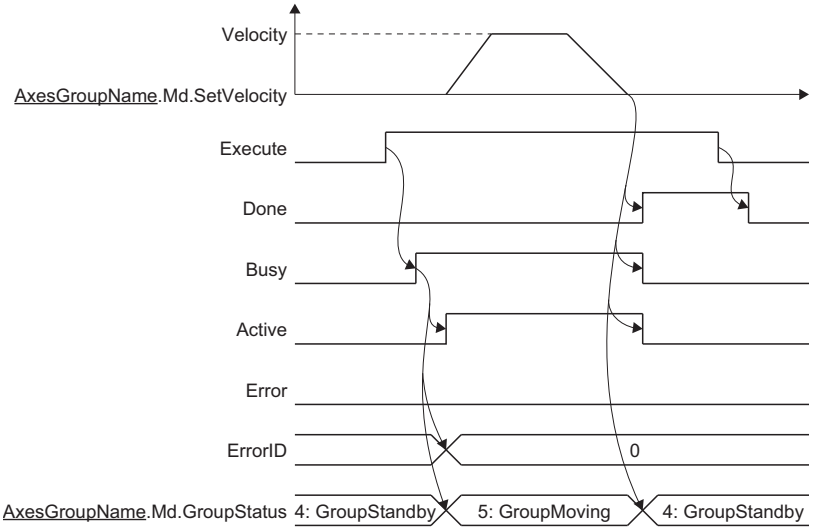
Output variable	Name	Data type	Default value	Description
CommandAborted	Abortion of execution	BOOL	FALSE	This variable becomes TRUE when execution of MCv_MoveCircularInterpolateAbsolute (Absolute Value Circular Interpolation Control) is aborted because of an error or multiple start.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. 📖MELSEC iQ-R Motion Module User's Manual (Application)

Processing details

- Circular interpolation control specifies an axes group and performs interpolation control mechanically so that the path from the start point (move start point) to the end point forms an arc for a linear axis which is orthogonally arranged.
- In circular interpolation control, interpolation control is performed using two arbitrary axes from the structuring axes set to the axes group.
- For circular interpolation control, the border point specification, center point specification, and radius specification circular interpolation methods are settable.
- In MCv_MoveCircularInterpolateAbsolute (Absolute Value Circular Interpolation Control), 2-axis circular interpolation control is performed by specifying the end point or sub point of the absolute position.

■Timing chart

- When the operation completes normally



- When an error occurs

  For the timing chart when an error occurs, refer to the following.

  ☞ Page 11 Basic operation of Execute command (Execute) type Motion control FBs

■Circular interpolation axis (CircAxes)

In circular interpolation control, circular interpolation is performed by using two arbitrary axes from the structuring axes set to the axes group.

The axes used to perform circular interpolation are specified from the structuring axes in the axes group by using Circular interpolation axis (CircAxes). The Circular interpolation axis (CircAxes) has two array elements. For the arrays, specify the index Nos. (1 to 16: structuring axis 1 to 16) of structuring axes to be used for interpolation control.

The first element of the array is regarded as the reference axis.

If the structuring axis specified does not have an axis set, "Circular Interpolation Axis Unset (error code: 1A62H)" occurs.

Ex.

When setting structuring axis 2 and 3 as the circular interpolation axes

CircAxes[0]:= 2;

CircAxes[1]:= 3;

■Circular interpolation mode (CircMode)

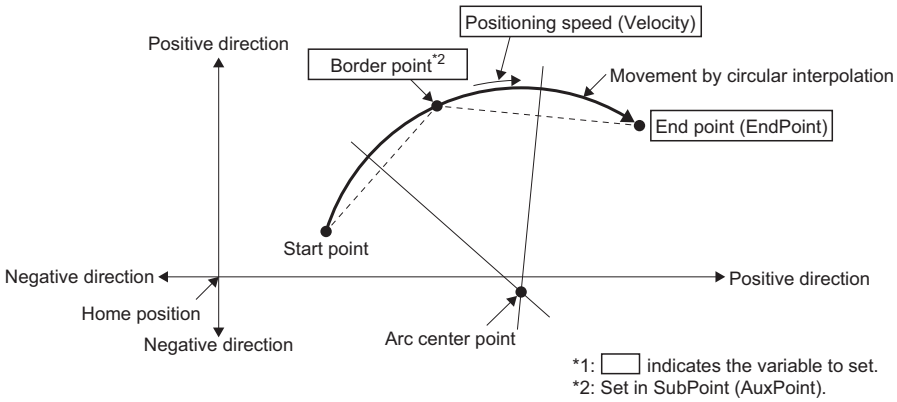
Sets the circular interpolation method (border point specification, center point specification, or radius specification) used to perform circular interpolation control.

The contents set in Sub point (AuxPoint) differ depending on the setting of Circular interpolation mode (CircMode).

Setting value	Sub point (AuxPoint)
0: Border point specification (mcBorder)	Sub point (AuxPoint) sets the border point on the arc connecting the start point and the end point. Positioning control with arc path that passes through the specified border point is performed using the two structuring axes set in Circular interpolation axis (CircAxes).
1: Center point specification (mcCenter)	Sub point (AuxPoint) sets the center point of the arc. Positioning control with an arc path having the center point at its center is performed using the two structuring axes set in Circular interpolation axis (CircAxes).
2: Radius specification (mcRadius)	Sub point (AuxPoint) sets the radius of the arc. Positioning control with an arc path having the specified radius is performed using the two structuring axes set in Circular interpolation axis (CircAxes).

The operation of each circular interpolation control set in Circular interpolation mode (CircMode) is shown below.

- Operation of circular interpolation control with "0: Border point specification (mcBorder)"
  In absolute value circular interpolation control with border point specification, positioning is carried out from the current position at start (start point address) to the address (end point address) set in End point (EndPoint) in an arc path that passes through the address (border point address) set in Sub point (AuxPoint). The resulting control path is an arc having as its center the intersection point between the perpendicular bisectors of the straight line between the current position at start and the border point address and the straight line between the border point address and the end point address. Note that a perfect circle cannot be drawn with border point specification.

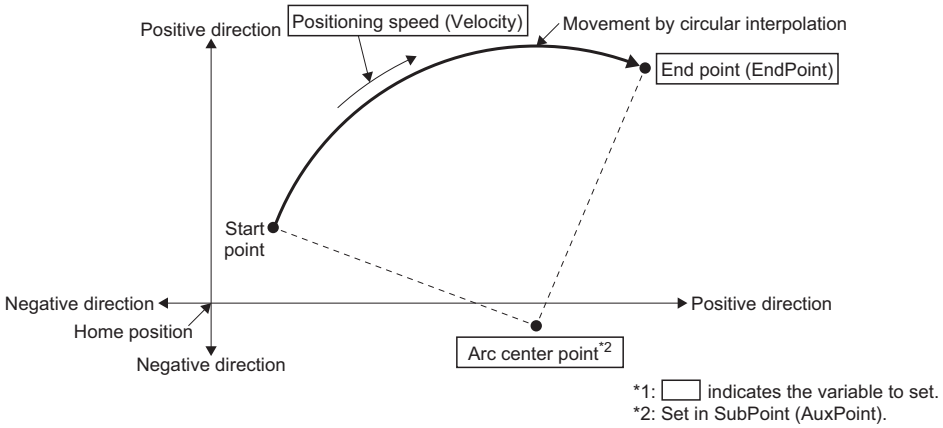


Restriction

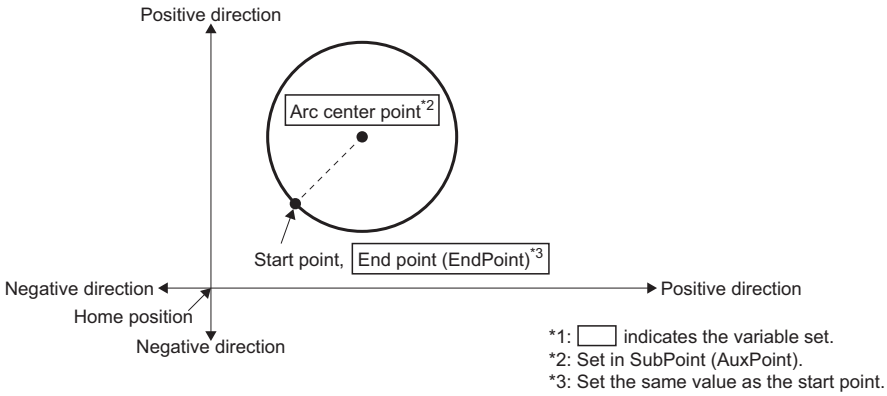
2-axis circular interpolation control with border point specification cannot be set in the following cases.

- When the radius exceeds "2147483647.0", the maximum radius for which 2-axis circular interpolation control is possible. ("Out of Radius Range (error code: 1A6CH)")
- When the border point address exceeds positioning range. ("Out of Boundary Point Address Range (error code: 1A64H)")
- When the end point address exceeds positioning range. ("Out of End Point Address Range (error code: 1A6DH)")
- When the start point address, border point address, and end point address are in the same straight line. ("Start Point - Boundary Point - End Point Address Same Straight Line (error code: 1A6AH)")
- When the start point address is the same as the end point address. ("Start Point - End Point Address Same Value (error code: 1A66H)")
- When the start point address is the same as the border point address. ("Start Point - Boundary Point Address Same Value (error code: 1A68H)")
- When the end point address is the same as the border point address. ("End Point - Boundary Point Address Same Value (error code: 1A69H)")

- Operation of circular interpolation control with "1: Center point specification (mcCenter)"
  In absolute value circular interpolation control operation with center point specification, positioning is carried out from the current position at start (start point address) to the end position set in End point (EndPoint) in an arc path having the center point set in Sub point (AuxPoint) as its center. For absolute value circular interpolation control with center point specification, the arc path is set using Path selection (PathChoice). For details, refer to the following.
  ☞ Page 293 Path selection (PathChoice)



  - To perform positioning for a perfect circle with a radius from the start point address to the arc center point, set the End point (EndPoint) to the same address as the start point. Setting the end point in End point (EndPoint) to the same value as the start point causes the end point and the start point to be identical.



<Example>
When End point (EndPoint) is set to the same value as the start point
AxesGroup001.Pr.Axis[1]:= Axis0001.AxisRef;
AxesGroup001.Pr.Axis[2]:= Axis0002.AxisRef;
EndPoint[0]:= Axis0001.Md.SetPosition;
EndPoint[1]:= Axis0002.Md.SetPosition;

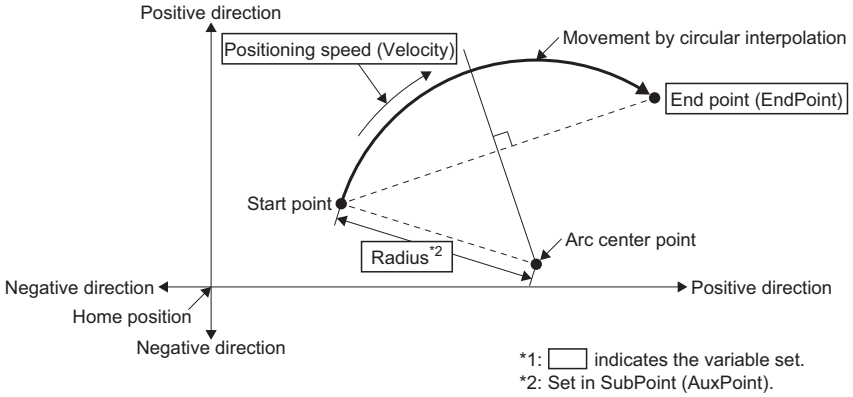
Set the set position of the structuring axis to be used in circular interpolation in End point (EndPoint) as shown above.

Restriction

2-axis circular interpolation control with center point specification cannot be set in the following cases.

- When the radius exceeds "2147483647.0", the maximum radius for which 2-axis circular interpolation control is possible. ("Out of Radius Range (error code: 1A6CH)")
- When the center point address exceeds positioning range. ("Out of Center Point Address Range (error code: 1A6BH)")
- When the start point address is the same as the center point address. ("Start Point - Center Point Address Same Value (error code: 1A65H)")
- When the end point address is the same as the center point address. ("End Point - Center Point Address Same Value (error code: 1A67H)")

- Operation of circular interpolation control with "2: Radius specification (mcRadius)"
  In absolute value circular interpolation control operation with radius specification, positioning is carried out from the current position at start (start point address) to the address (end point address) set in End point (EndPoint) in an arc path having the radius set in Sub point (AuxPoint). The resulting control path is an arc having as its center the intersection point between the perpendicular bisectors of the straight line between the current position at start and the end point address and the set radius. For absolute value circular interpolation control with radius specification, the arc path is set using Path selection (PathChoice). For details, refer to the following.
  ☞ Page 293 Path selection (PathChoice)



Restriction

2-axis circular interpolation control with radius specification cannot be set in the following cases.

- When the radius exceeds "2147483647.0", the maximum radius for which 2-axis circular interpolation control is possible. ("Out of Radius Range (error code: 1A6CH)")
- When the end point address exceeds positioning range. ("Out of End Point Address Range (error code: 1A6DH)")
- When the start point address is the same as the end point address. ("Start Point - End Point Address Same Value (error code: 1A66H)")
- When the distance between the start point address and the end point address is larger than the radius. ("Radius Setting Error (error code: 1A6EH)")

■Sub point (AuxPoint)

Sets the position of the sub point (border point, center point, or radius) used to perform circular interpolation.

Sub point (AuxPoint) has 16 array elements. The contents set in Sub point (AuxPoint) differ depending on the setting of Circular interpolation mode (CircMode).

Circular interpolation mode (CircMode)	Setting range	Sub point (AuxPoint) setting
0: Border point specification (mcBorder)	-10000000000.0 ≤ Setting value < 10000000000.0*1	Sub point (AuxPoint) sets the border point on the arc connecting the start point and the end point. AuxPoint[0..15] indicates the sub point position of structuring axis 1 to 16. • Sub point (AuxPoint) of structuring axes not set to the circular interpolation axes is ignored. • Sub point (AuxPoint) of axes after all structuring axes set to the circular interpolation axes can be omitted.
1: Center point specification (mcCenter)		Sub point (AuxPoint) sets the center point of the arc. AuxPoint[0..15] indicates the sub point position of structuring axis 1 to 16. • Sub point (AuxPoint) of structuring axes not set to the circular interpolation axes is ignored. • Sub point (AuxPoint) of axes after all structuring axes set to the circular interpolation axes can be omitted.
2: Radius specification (mcRadius)	0.000001 to 2147483647.0	Sub point (AuxPoint) sets the radius of the arc. AuxPoint[0] indicates the radius of the circle. • Inputs after AuxPoint[1] are ignored and can be omitted.

*1 When the ring counter is enabled, the ring counter range is used.

Ex.

When setting structuring axis 2 and 3 to the circular interpolation axes and setting the absolute position of the sub point for structuring axis 2 and 3 as 2000.0 and 3000.0 respectively.

CircAxes[0]:= 2;

CircAxes[1]:= 3;

AuxPoint[0]:= 2000.0;

AuxPoint[1]:= 3000.0;

AuxPoint[2]:= 0.0; *2

⋮

AuxPoint[15]:= 0.0; *2

*2 "AuxPoint[2]:= 0.0; to AuxPoint[15]:= 0.0;" can be omitted.

■End point (EndPoint)

Sets the end point of the absolute position.

End point (EndPoint) has 16 array elements.

EndPoint[0..15] indicates the end point position of structuring axis 1 to 16.

- End point (EndPoint) for structuring axes which are not set to the circular interpolation axis are ignored.
- End point (EndPoint) after all of the structuring axes which are set to the circular interpolation axis can be omitted.

Set End point (EndPoint) within the following range.

Setting range
-10000000000.0 ≤ Setting value < 10000000000.0*1

*1 When the ring counter is enabled, the ring counter range is used.

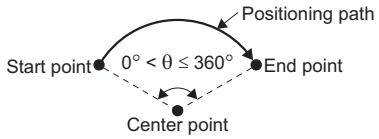
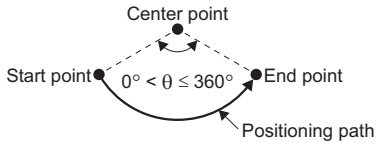
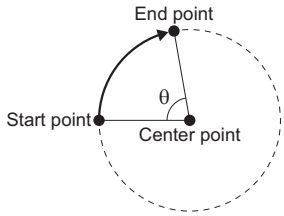
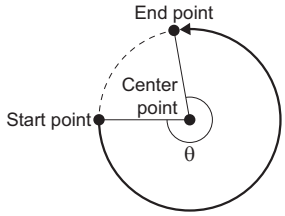
■Path selection (PathChoice)

Sets the rotation direction for circular interpolation. For the rotation direction, the first element of the array set in Circular interpolation axis (CircAxes) is regarded as the reference axis.

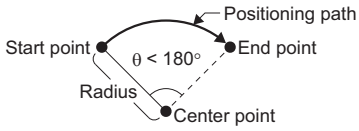
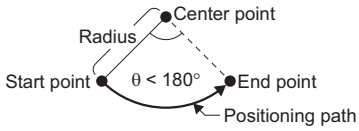
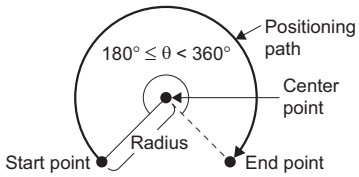
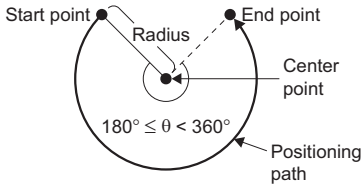
Path selection (PathChoice) is set when Circular interpolation mode (CircMode) is set to "1: Center point specification (mcCenter)" or "2: Radius specification (mcRadius)". When Circular interpolation mode (CircMode) is set to "0: Border point specification (mcBorder)", this input is omitted.

The following shows the setting value, setting description, arc center angle that can be controlled, and path of Path selection (PathChoice) according to the setting of Circular interpolation mode (CircMode).

• Path for "1: Center point specification (mcCenter)"

Setting value	Setting description	Arc center angle that can be controlled	Path
0: CW (mcCW)	CW	$0° < \theta \leq 360°$	
1: CCW (mcCCW)	CCW	$0° < \theta \leq 360°$	
2: Shortcut (mcShortWay)	Shortcut • When the start point, center point, and end point are on one straight line, becomes CW. (A semicircle is drawn in the CW direction.) • When the start point is same as the end point, "Path Choice Setting Incorrect (error code: 1AB7H)" occurs.	$0° < \theta \leq 180°$	
3: Detour (mcLongWay)	Detour • When the start point, center point, and end point are on one straight line, becomes CW. (A semicircle is drawn in the CW direction.) • When the start point is same as the end point, becomes CW. (A perfect circle is drawn in CW direction.)	$180° \leq \theta \leq 360°$	

• Path for "2: Radius specification (mcRadius)"

Setting value	Setting description	Arc center angle that can be controlled	Path
0: CW (mcCW)	CW (CW shortcut) • When the start point is same as the end point, "Path Choice Setting Incorrect (error code: 1AB7H)" occurs.	0° < θ < 180°	
1: CCW (mcCCW)	CCW (CCW shortcut) • When the start point is same as the end point, "Path Choice Setting Incorrect (error code: 1AB7H)" occurs.	0° < θ < 180°	
4: CW Detour (mcCWLongWay)	CW detour • When the start point, end point, and center point are on one straight line, a semicircle is drawn.	180° ≤ θ < 360°	
5: CCW Detour (mcCCWLongWay)	CCW detour • When the start point, end point, and center point are on one straight line, a semicircle is drawn.	180° ≤ θ < 360°	

■Velocity (Velocity)

Sets the maximum speed of the path used in MCv_MoveCircularInterpolateAbsolute (Absolute Value Circular Interpolation Control).

Setting range
0.0, 0.0001 to 2500000000.0[*1*2]

*1 To perform floating-point operation, the lower limit value of the commanded velocity is limited by the following restrictions.
If the velocity that is converted from the target velocity into the operation cycle is less than "0.00001", "Out of Calculation Cycle Converted Speed Range (error code: 1AE4H)" occurs. (If the velocity is changed, "Out of Calculation Cycle Converted Speed Range Warning (warning code: 0D2FH)" occurs instead). To improve the precision of floating-point operation, change Position command unit (AxisName.Pr.Unit_Position) and Speed command unit (AxisName.Pr.Unit_Velocity) and perform settings so that the velocity converted into the operation cycle is not less than "0.00001".

*2 When a Motion control FB with a set velocity of "0.0" performs a multiple start, the set velocity of the previous Motion control FB is used.

■Acceleration (Acceleration)

Sets the acceleration used in MCv_MoveCircularInterpolateAbsolute (Absolute Value Circular Interpolation Control).

The setting range differs depending on the setting for Acceleration/deceleration method setting (Options (Options): Bit 0 to 2).

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000[*1], positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	0.000000[*1], positive numbers from 0.000001 to 8400.0[s]

*1 The operation performed at start varies depending on the setting of Operation selection at start acceleration/deceleration 0 (AxisName.Pr.AccelerationZeroBehavior). When changing the acceleration or deceleration, the changes are not accepted.

■Deceleration (Deceleration)

Sets the deceleration used in MCv_MoveCircularInterpolateAbsolute (Absolute Value Circular Interpolation Control).
Specifies the setting range if Acceleration/deceleration method setting (Options (Options): Bit 0 to 2) is set to "0: Acceleration/deceleration specification method (mcAccDec)". Deceleration (Deceleration) is not used if the acceleration/deceleration method is set to "1: Acceleration/deceleration time-fixed method (mcFixedTime)".

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000*1, positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	Not used

*1 The operation performed at start varies depending on the setting of Operation selection at start acceleration/deceleration 0 (AxisName.Pr.AccelerationZeroBehavior). When changing the acceleration or deceleration, the changes are not accepted.

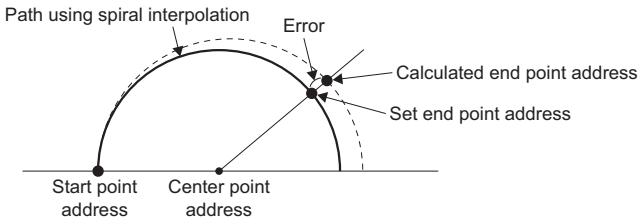
■Jerk (Jerk)

Sets the jerk used in MCv_MoveCircularInterpolateAbsolute (Absolute Value Circular Interpolation Control).
Specifies the setting range if Acceleration/deceleration method setting (Options (Options): Bit 0 to 2) is set to "0: Acceleration/deceleration specification method (mcAccDec)". Jerk (Jerk) is not used if the acceleration/deceleration method is set to "1: Acceleration/deceleration time-fixed method (mcFixedTime)".

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000, positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	Not used

■Circular interpolation error tolerance (CircularErrorTolerance)

In circular interpolation control with center point specification, the arc path calculated from the start position and center position may deviate from the end position set in End point (EndPoint).
Circular interpolation error tolerance (CircularErrorTolerance) sets the allowable width of errors between the calculated arc path and the end position.

Setting range	Description
Calculated error ≤ Circular interpolation error tolerance (CircularErrorTolerance)	Circular interpolation control to the set end point address is performed while the error compensation is carried out by spiral interpolation. 
Calculated error > Circular interpolation error tolerance (CircularErrorTolerance)	At the positioning start, "Circular Interpolation Error Allowable Value Over (error code: 1A71H)" occurs and the control will not start. The machine will immediately stop if the error is detected during positioning control.

■Buffer mode (BufferMode)

Sets the operation used to perform multiple start (buffer mode).

The following buffer modes are settable in MCv_MoveCircularInterpolateAbsolute (Absolute Value Circular Interpolation Control).

Setting value	Description
0: Aborting (mcAborting)	Aborts (cancels) the under-control FB and executes the next FB immediately.
1: Buffered (mcBuffered)	Buffers the next FB on the under-control FB. If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after completion of the under-control FB.
2: BlendingLow (mcBlendingLow)	Buffers the next FB on the under-control FB.*1 If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. When the under-control FB reaches the target position, the lower target velocity between the under-control FB and the buffering FB is used as the switching speed.
3: BlendingPrevious (mcBlendingPrevious)	Buffers the next FB on the under-control FB.*1 If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. When the under-control FB reaches the target position, the switching speed changes to the target velocity of the under-control FB.
4: BlendingNext (mcBlendingNext)	Buffers the next FB on the under-control FB.*1 If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. When the under-control FB reaches the target position, the switching speed changes to the target velocity of the buffering FB.
5: BlendingHigh (mcBlendingHigh)	Buffers the next FB on the under-control FB.*1 If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. When the under-control FB reaches the target position, the switching speed changes to the higher target velocity value between the under-control FB and the buffering FB.

*1 Stops are not performed between the under-control FB and the buffering FB.

Point

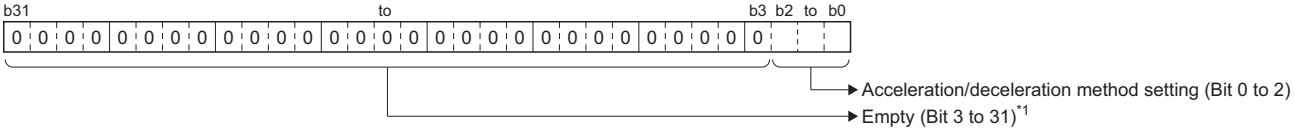
For details of multiple start (buffer mode), refer to the following.

MELSEC iQ-R Motion Module User's Manual (Application)

■Options (Options)

Sets the function options used for MCv_MoveCircularInterpolateAbsolute (Absolute Value Circular Interpolation Control) by specifying the bit.

Contents that are set by specifying the bit are shown below.



*1 Set empty areas to "0". When a value other than "0" is set, "Out of Options Range (error code: 1A4EH)" occurs.

Bit	Name	Description
0 to 2	Acceleration/deceleration method setting	Sets the acceleration/deceleration method used to perform control. • 0: Acceleration/deceleration specification method (mcAccDec) • 1: Acceleration/deceleration time-fixed method (mcFixedTime)

• Acceleration/deceleration method setting (Bit 0 to 2)

Setting value	Description
0: Acceleration/deceleration specification method (mcAccDec)	An acceleration/deceleration method using Acceleration (Acceleration), Deceleration (Deceleration), and Jerk (Jerk) set in MCv_MoveCircularInterpolateAbsolute (Absolute Value Circular Interpolation Control).
1: Acceleration/deceleration time-fixed method (mcFixedTime)	An acceleration/deceleration method using the acceleration/deceleration time specified in MCv_MoveCircularInterpolateAbsolute (Absolute Value Circular Interpolation Control) regardless of the velocity. The acceleration/deceleration time does not use Acceleration (Acceleration), Deceleration (Deceleration), and Jerk (Jerk).

■Necessary objects

When using MCv_MoveCircularInterpolateAbsolute (Absolute Value Circular Interpolation Control), set the following slave objects for all structuring axes in the specified axes group.

- Target position (607AH)

If there is a structuring axis present that does not have the slave object set, "Necessary Slave Object Unset (error code: 1AA8H)" occurs and the axis will not start.

For details of slave object settings, refer to the following.

MELSEC iQ-R Motion Module User's Manual (Application)

Precautions

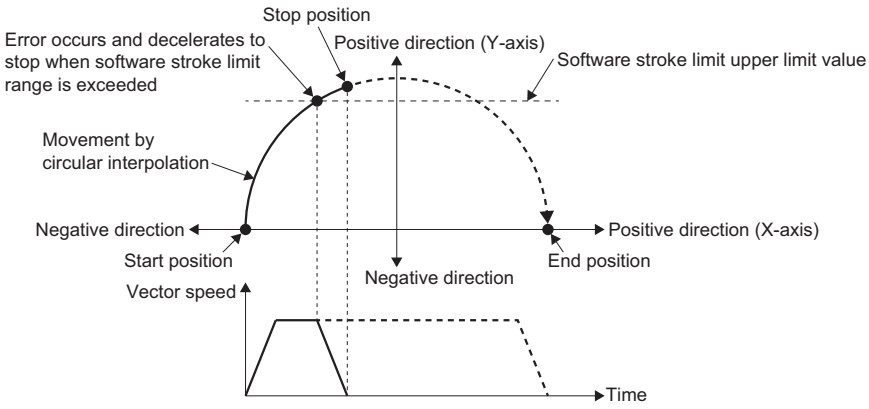
- Circular interpolation including an axis whose stroke limit is set to invalid cannot be performed. "Software Stroke Limit Invalid at Circular Interpolation (error code: 1A72H)" occurs and the operation does not start.
- When the interpolation path exceeds the stroke limit range during interpolation operation, "Software Stroke Limit Over (Forward Direction) (error code: 1A03H)" or "Software Stroke Limit Over (Reverse Direction) (error code: 1A04H)" occurs and the operation will stop.

Ex.

When the upper limit of the software stroke limit is exceeded in the positive direction of the Y-axis

For an immediate stop, stops at the same time that the error occurs.

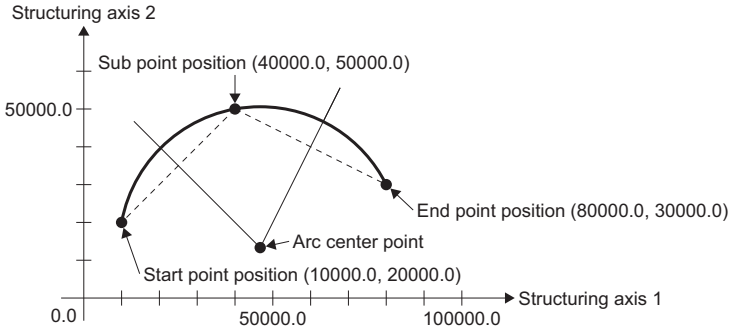
For a deceleration stop, decelerates to a stop along the arc of the circle as shown below.



Program example

In the following program example, the circular interpolation control start (bCircularInterpolateCMD) is set to TRUE and Axes group 1 (AxesGroup001) is enabled, following which 2-axis circular interpolation (sub point designation) using absolute positioning control is performed for Axis group 1 (AxesGroup001) using the settings shown below.

• Operation



• Settings

Item	Setting value	
	Structuring axis 1	Structuring axis 2
Sub point	40000.0	50000.0
End point	80000.0	30000.0
Velocity	50000.0	
Acceleration	50000.0	
Deceleration	50000.0	
Jerk	0.0	
Circular interpolation error tolerance	100.0	

■Axes

Axis No.	Label name	Data type	Comment
1	Axis0001	AXIS_REF	Axis 1
2	Axis0002	AXIS_REF	Axis 2

■Axes group

Axes group No.	Label name	Data type	Comment
1	AxesGroup001	AXES_GROUP_REF	Axes group 1

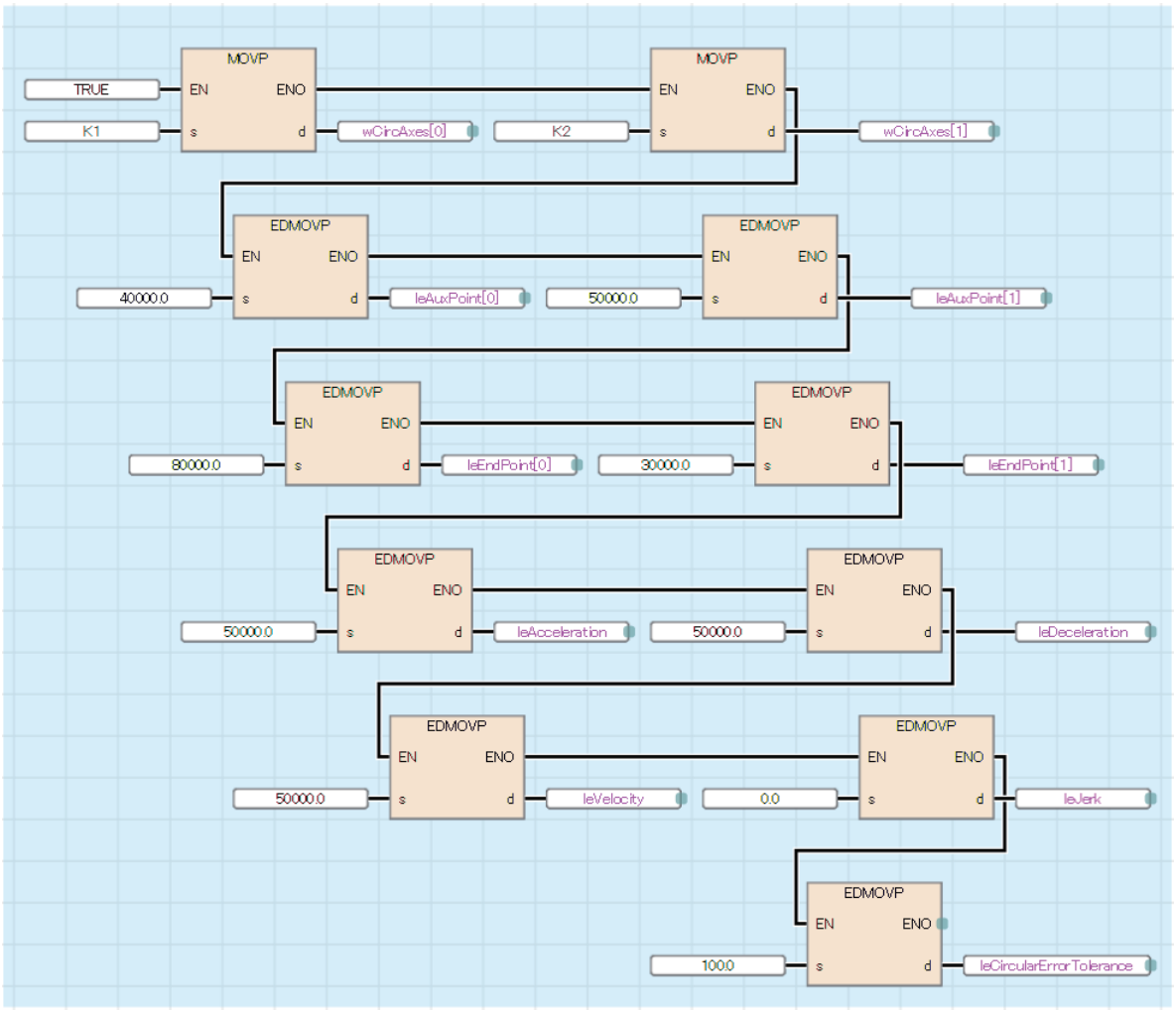
■Labels used

Label name	Data type	Comment
MCv_MoveCircularInterpolateAbsolute_1	MCv_MoveCircularInterpolateAbsolute	Absolute Value Circular Interpolation Control FB
wCircAxes	Word[signed](0..1)	Circular interpolation axis
leAuxPoint	Double-precision real number(0..15)	Sub point
leEndPoint	Double-precision real number(0..15)	End point
leVelocity	Double-precision real number	Velocity
leAcceleration	Double-precision real number	Acceleration
leDeceleration	Double-precision real number	Deceleration
leJerk	Double-precision real number	Jerk
leCircErrorTolerance	Double-precision real number	Circular interpolation error tolerance
bCircularAbsDone	Bit	Execution completion
bCircularAbsBusy	Bit	Executing
bCircularAbsActive	Bit	Controlling
bCircularCommandAborted	Bit	Abortion of execution
bCircularAbsError	Bit	Error
uwCircularAbsErrorID	Word [unsigned]/bit string [16-bit]	Error code
MC_GroupEnable_1	MC_GroupEnable	Axes Group Enabled FB

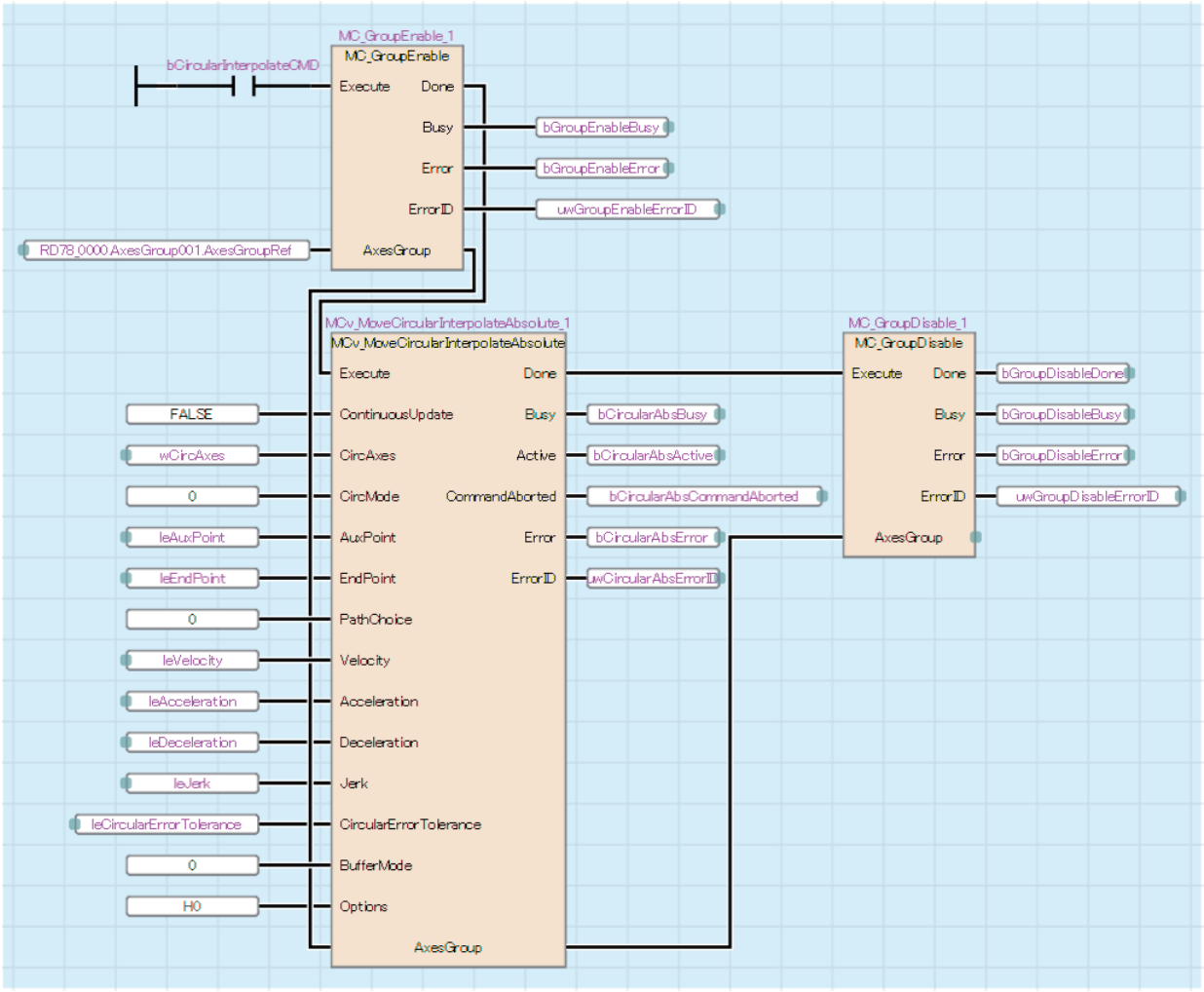
Label name	Data type	Comment
bCircularInterpolateCMD	Bit	Circular interpolation control start
bGroupEnableDone	Bit	Axes Group Enabled Completion
bGroupEnableBusy	Bit	Executing
bGroupEnableError	Bit	Error
uwGroupEnableErrorID	Word [unsigned]/bit string [16-bit]	Error code
MC_GroupDisable_1	MC_GroupDisable	Axes Group Disabled FB
bGroupDisableDone	Bit	Axes Group Disabled Completion
bGroupDisableBusy	Bit	Executing
bGroupDisableError	Bit	Error
uwGroupDisableErrorID	Word [unsigned]/bit string [16-bit]	Error code

■FBD/LD program (CPU module side)

- Data setting for 2-axis circular interpolation control



• Axes Group Enabled/Circular interpolation control (Border point specification)/Axis Group Disabled



■ST program (Motion module side)

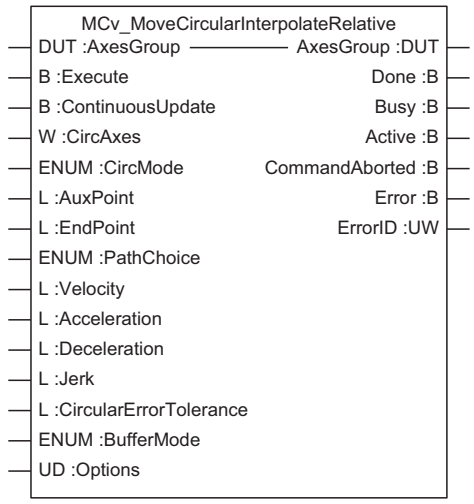
```
//-----Data setting for 2-axis circular interpolation control-----
wCircAxes[0]:= 1;
wCircAxes[1]:= 2;
leAuxPoint[0]:= 40000.0;
leAuxPoint[1]:= 50000.0;
leEndPoint[0]:= 80000.0;
leEndPoint[1]:= 30000.0;
leVelocity:= 50000.0;
leAcceleration:= 50000.0;
leDeceleration:= 50000.0;
leJerk:= 0.0;
leCircularErrorTolerance:= 100.0;

//-----Axes Group Enabled-----
MC_GroupEnable_1(
   AxesGroup:= AxesGroup001.AxesGroupRef ,
   Execute:= bCircularInterpolateCMD ,
   Done=> bGroupEnableDone,
   Busy=> bGroupEnableBusy ,
   Error=> bGroupEnableError ,
   ErrorID=> uwGroupEnableErrorID
);
```

```
//-----Circular interpolation control (Border point specification)-----
MCv_MoveCircularInterpolateAbsolute_1(
    AxesGroup:= AxesGroup001.AxesGroupRef ,
    Execute:= bGroupEnableDone ,
    ContinuousUpdate:= FALSE ,
    CircAxes:= wCircAxes ,
    CircMode:= MC_CIRC_MODE__mcBorder ,
    AuxPoint:= leAuxPoint ,
    EndPoint:= leEndPoint ,
    PathChoice:= MC_CIRC_PATHCHOICE__mcCW ,
    Velocity:= leVelocity ,
    Acceleration:= leAcceleration ,
    Deceleration:= leDeceleration ,
    Jerk:= leJerk ,
    CircularErrorTolerance:= leCircularErrorTolerance ,
    BufferMode:= MC_BUFFER_MODE__mcAborting ,
    Options:= H00000000 ,
    Done=> bCircularAbsDone ,
    Busy=> bCircularAbsBusy ,
    Active=> bCircularAbsActive ,
    CommandAborted=> bCircularAbsCommandAborted ,
    Error=> bCircuAbsError ,
    ErrorID=> uwCircularAbsErrorID
);

//-----Axes Group Disabled-----
MC_GroupDisable_1(
    AxesGroup:= AxesGroup001.AxesGroupRef ,
    Execute:= bCircularAbsDone ,
    Done=> bGroupDisableDone ,
    Busy=> bGroupDisableBusy ,
    Error=> bGroupDisableError ,
    ErrorID=> uwGroupDisableErrorID
);
```

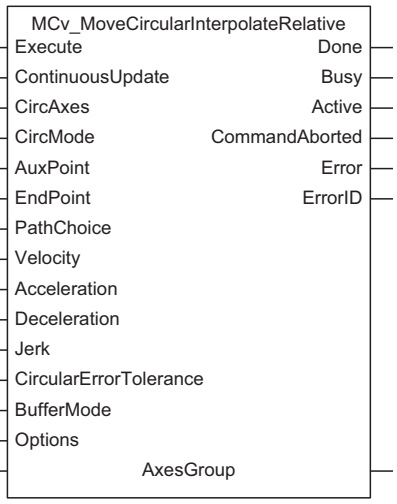
MCv_MoveCircularInterpolateRelative

Executes positioning with 2-axis circular interpolation using the structuring axes of the set axes group by setting the relative position from the current position at start to the end point and sub point.

Ladder



FBD/LD



ST

```
MCv_MoveCircularInterpolateRelative(
  AxesGroup:= ?AXES_GROUP_REF? ,
  Execute:= ?BOOL? ,
  ContinuousUpdate:= ?BOOL? ,
  CircAxes:= ?INT(0..1)? ,
  CircMode:= ?INT? ,
  AuxPoint:= ?LREAL(0..15)? ,
  EndPoint:= ?LREAL(0..15)? ,
  PathChoice:= ?INT? ,
  Velocity:= ?LREAL? ,
  Acceleration:= ?LREAL? ,
  Deceleration:= ?LREAL? ,
  Jerk:= ?LREAL? ,
  CircularErrorTolerance:= ?LREAL? ,
  BufferMode:= ?INT? ,
  Options:= ?DWORD? ,
  Done=> ?BOOL? ,
  Busy=> ?BOOL? ,
  Active=> ?BOOL? ,
  CommandAborted=> ?BOOL? ,
  Error=> ?BOOL? ,
  ErrorID=> ?WORD?
);
```

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Relative Value Circular Interpolation Control	328	8	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
AxesGroup	Axes group information	AXES_GROUP_REF	At start	—	Omission not possible	This variable sets the axes group. For the variables used (AxesGroupName.AxesGroupRef.), refer to the following. ☞ Page 46 AxesGroupName.AxesGroupRef. (Axes group information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Execute command	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MCv_MoveCircularInterpolateRelative (Relative Value Circular Interpolation Control).
ContinuousUpdate	Continuous update	BOOL	At start	TRUE, FALSE	FALSE	This variable sets whether to enable or disable continuous change of Velocity (Velocity), Acceleration (Acceleration), and Deceleration (Deceleration). Continuous update re-imports input variables to the instance in execution without aborting the operation. • FALSE: Disable • TRUE: Enable

Input variable	Name	Data type	Input	Setting range	Default value	Description
CircAxes	Circular interpolation axis	INT[0..1]	At start	1 to 16	0	This variable specifies the axes to be used in circular interpolation control from the structuring axes. Specifies the index No. (1 to 16) of the structuring axes in the array. The first element of the array is regarded as the reference axis. For details, refer to the following. ☞ Page 305 Circular interpolation axis (CircAxes)
CircMode	Circular interpolation mode	INT (MC_CIRC_MODE)	At start	0 to 2	0	This variable sets the circular interpolation designation method. • 0: Border point specification (mcBorder) • 1: Center point specification (mcCenter) • 2: Radius specification (mcRadius) For details, refer to the following. ☞ Page 306 Circular interpolation mode (CircMode)
AuxPoint	Sub point	LREAL[0..15]	At start	■When "0: Border point specification (mcBorder)" or "1: Center point specification (mcCenter)": -10000000000.0 to 10000000000.0 ■When "2: Radius specification (mcRadius)": 0.000001 to 2147483647.0	0.0	Sets the relative position of the sub point (border point, center point, and radius) according to unit of the axis. The relative position is set as 1-dimensional array data. This data changes as shown below based on the setting of Circular interpolation mode (CircMode). ■When Circular interpolation mode (CircMode) is set to "0: Border point specification (mcBorder)" or "1: Center point specification (mcCenter)": This data is treated as the relative position for structuring axes 1 to 16. ■When Circular interpolation mode (CircMode) is set to "2: Radius specification (mcRadius)": Treats the first element as the radius and ignores the second and subsequent elements. For details, refer to the following. ☞ Page 309 Sub point (AuxPoint)
EndPoint	End point	LREAL[0..15]	At start	-10000000000.0 to 10000000000.0	0.0	This variable sets the position of the end point. The position is set as 1-dimensional array data. This data is treated as the relative position for structuring axes 1 to 16. For details, refer to the following. ☞ Page 309 End point (EndPoint)
PathChoice	Path selection	INT (MC_CIRC_PATHCHOICE)	At start	0 to 5	0	This variable sets the rotation direction of the circular interpolation. • 0: CW (mcCW) • 1: CCW (mcCCW) • 2: Shortcut (mcShortWay) • 3: Detour (mcLongWay) • 4: CW Detour (mcCWLongWay) • 5: CCW Detour (mcCCWLongWay) *: When specifying "0: Border point specification (mcBorder)" in Circular interpolation mode (CircMode), the input is ignored. *: When specifying "2: Radius specification (mcRadius)" in Circular interpolation mode (CircMode), "0: mcCW" means the CW shortcut and "1: mcCCW" means the CCW shortcut. For details, refer to the following. ☞ Page 310 Path selection (PathChoice)
Velocity	Velocity	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0, 0.0001 to 2500000000.0	0.0	This variable sets the speed command value. For details, refer to the following. ☞ Page 311 Velocity (Velocity)

Input variable	Name	Data type	Input	Setting range	Default value	Description
Acceleration	Acceleration	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the acceleration. For details, refer to the following. ☞ Page 311 Acceleration (Acceleration)
Deceleration	Deceleration	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the deceleration. For details, refer to the following. ☞ Page 312 Deceleration (Deceleration)
Jerk	Jerk	LREAL	At start	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the jerk. For details, refer to the following. ☞ Page 312 Jerk (Jerk)
CircularErrorTolerance	Circular interpolation error tolerance	LREAL	At start	0.000001 to 100000.0	100.0	This variable sets the allowable range for circular interpolation errors. For details, refer to the following. ☞ Page 312 Circular interpolation error tolerance (CircularErrorTolerance)
BufferMode	Buffer mode	INT (MC_BUFFER_MODE)	At start	0 to 5	0	This variable sets the buffer mode. • 0: Aborting (mcAborting) • 1: Buffered (mcBuffered) • 2: BlendingLow (mcBlendingLow) • 3: BlendingPrevious (mcBlendingPrevious) • 4: BlendingNext (mcBlendingNext) • 5: BlendingHigh (mcBlendingHigh) For details, refer to the following. ☞ Page 313 Buffer mode (BufferMode)
Options	Options	DWORD(HEX)	At start	00000000H to 00000005H	00000000H	This variable sets the function options for MCv_MoveCircularInterpolateRelative (Relative Value Circular Interpolation Control) by specifying the bit. For details, refer to the following. ☞ Page 314 Options (Options)

■Output variables

Output variable	Name	Data type	Default value	Description
Done	Completion	BOOL	FALSE	This variable becomes TRUE when the control is completed. This variable changes as shown below based on the status of Start (Execute) at operation completion. ■When Start (Execute) is TRUE This variable remains TRUE until Start (Execute) becomes FALSE. ■When Start (Execute) is FALSE This variable becomes TRUE for one cycle only.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MCv_MoveCircularInterpolateRelative (Relative Value Circular Interpolation Control) is executed.
Active	Controlling	BOOL	FALSE	This variable becomes TRUE while MCv_MoveCircularInterpolateRelative (Relative Value Circular Interpolation Control) is controlling the axis. When multiple MCv_MoveCircularInterpolateRelative (Relative Value Circular Interpolation Control) FBs are executed for the same axes group, only the Controlling (Active) of one MCv_MoveCircularInterpolateRelative (Relative Value Circular Interpolation Control) FB becomes TRUE.

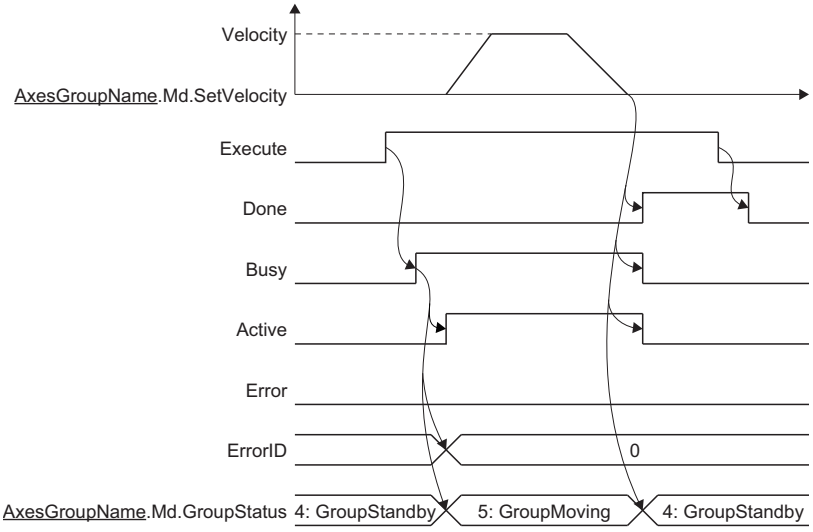
Output variable	Name	Data type	Default value	Description
CommandAborted	Abortion of execution	BOOL	FALSE	This variable becomes TRUE when execution of MCv_MoveCircularInterpolateRelative (Relative Value Circular Interpolation Control) is aborted because of an error or multiple start.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. MELSEC iQ-R Motion Module User's Manual (Application)

Processing details

- Circular interpolation control specifies an axes group and performs interpolation control mechanically so that the path from the start point (move start point) to the end point forms an arc for a linear axis which is orthogonally arranged.
- In circular interpolation control, interpolation control is performed using two arbitrary axes from the structuring axes set to the axes group.
- For circular interpolation control, the border point specification, center point specification, and radius specification circular interpolation methods are settable.
- In MCv_MoveCircularInterpolateRelative (Relative Value Circular Interpolation Control), 2-axis circular interpolation control is performed by specifying the movement amount from the current position to the end point or sub point.

■Timing chart

- When the operation completes normally



- When an error occurs

  For the timing chart when an error occurs, refer to the following.

  ☞ Page 11 Basic operation of Execute command (Execute) type Motion control FBs

■Circular interpolation axis (CircAxes)

In circular interpolation control, circular interpolation is performed by using two arbitrary axes from the structuring axes set to the axes group.

The axes used to perform circular interpolation are specified from the structuring axes in the axes group by using Circular interpolation axis (CircAxes). The Circular interpolation axis (CircAxes) has two array elements. For the arrays, specify the index Nos. (1 to 16: structuring axis 1 to 16) of structuring axes to be used for interpolation control.

The first element of the array is regarded as the reference axis.

If the structuring axis specified does not have an axis set, "Circular Interpolation Axis Unset (error code: 1A62H)" occurs.

Ex.

When setting structuring axis 2 and 3 as the circular interpolation axes

CircAxes[0]:= 2;

CircAxes[1]:= 3;

■Circular interpolation mode (CircMode)

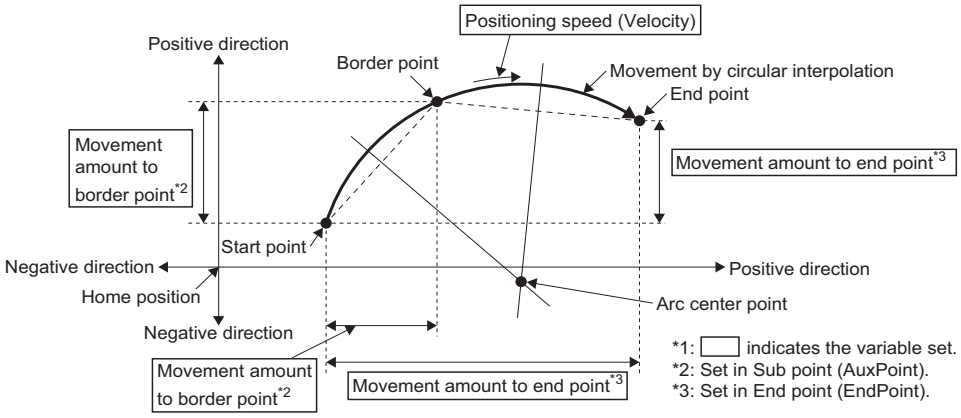
Sets the circular interpolation method (border point specification, center point specification, or radius specification) used to perform circular interpolation control.

The contents set in Sub point (AuxPoint) differ depending on the setting of Circular interpolation mode (CircMode).

Setting value	Sub point (AuxPoint)
0: Border point specification (mcBorder)	Sub point (AuxPoint) sets the border point on the arc connecting the start point and the end point. Positioning control with arc path that passes through the specified border point is performed using the two structuring axes set in Circular interpolation axis (CircAxes).
1: Center point specification (mcCenter)	Sub point (AuxPoint) sets the center point of the arc. Positioning control with an arc path having the center point at its center is performed using the two structuring axes set in Circular interpolation axis (CircAxes).
2: Radius specification (mcRadius)	Sub point (AuxPoint) sets the radius of the arc. Positioning control with an arc path having the specified radius is performed using the two structuring axes set in Circular interpolation axis (CircAxes).

The operation of each circular interpolation control based on the set circular interpolation mode is shown below.

- Operation of circular interpolation control with "0: Border point specification (mcBorder)"
  In relative value circular interpolation control with border point specification, positioning is carried out from the current position at start (start point address) to the movement amount address (end point address) set in End point (EndPoint) in an arc path that passes through the movement amount address (border point address) set in Sub point (AuxPoint). The resulting control path is an arc having as its center the intersection point between the perpendicular bisectors of the straight line between the current position at start and the border point address and the straight line between the border point address and the end point address.



Restriction

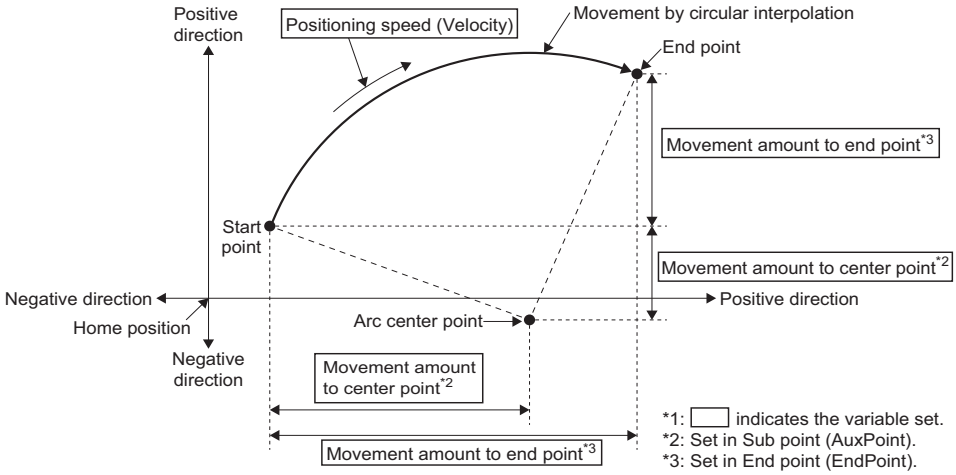
2-axis circular interpolation control with border point specification cannot be set in the following cases.

- When the radius exceeds "2147483647.0", the maximum radius for which 2-axis circular interpolation control is possible. ("Out of Radius Range (error code: 1A6CH)")
- When the border point address exceeds positioning range. ("Out of Boundary Point Address Range (error code: 1A64H)")
- When the end point address exceeds positioning range. ("Out of End Point Address Range (error code: 1A6DH)")
- When the start point address, border point address, and end point address are in the same straight line. ("Start Point - Boundary Point - End Point Address Same Straight Line (error code: 1A6AH)")
- When the start point address is the same as the end point address. ("Start Point - End Point Address Same Value (error code: 1A66H)")
- When the start point address is the same as the border point address. ("Start Point - Boundary Point Address Same Value (error code: 1A68H)")
- When the end point address is the same as the border point address. ("End Point - Boundary Point Address Same Value (error code: 1A69H)")

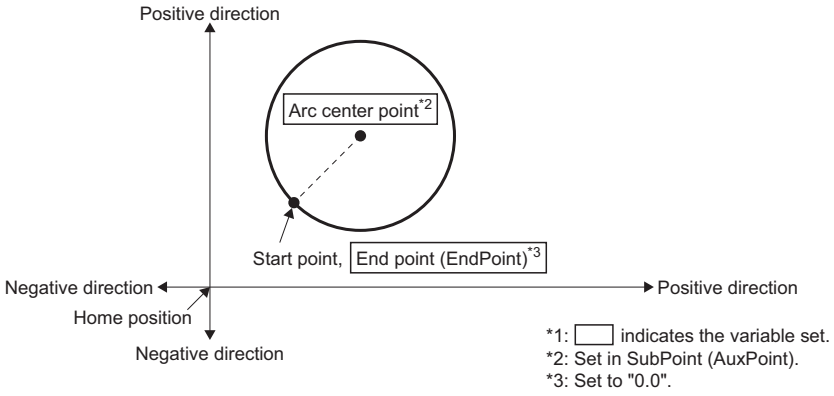
- Operation of circular interpolation control with "1: Center point specification (mcCenter)"

  In relative value circular interpolation control operation with center point specification, positioning is carried out from the current position at start (start point address) to the movement amount address set in End point (EndPoint) in an arc path having the movement amount address (center point address) set in Sub point (AuxPoint) as its center. For relative value circular interpolation control with center point specification, the arc path is set using Path selection (PathChoice). For details, refer to the following.

  ☞ Page 310 Path selection (PathChoice)



- Setting the movement amount to the end point as "0.0" in End point (EndPoint) causes the end point and the start point to be identical and performs positioning for a perfect circle with a radius from the start point address to the arc center point.

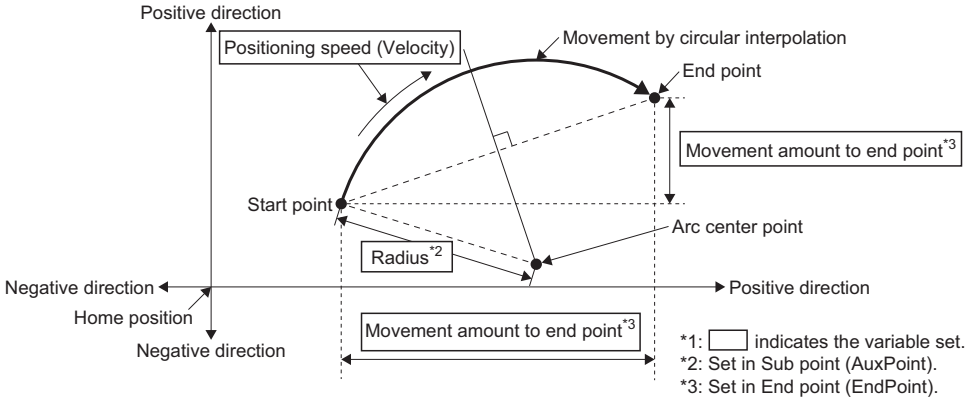


Restriction

2-axis circular interpolation control with center point specification cannot be set in the following cases.

- When the radius exceeds "2147483647.0", the maximum radius for which 2-axis circular interpolation control is possible. ("Out of Radius Range (error code: 1A6CH)")
- When the center point address exceeds positioning range. ("Out of Center Point Address Range (error code: 1A6BH)")
- When the start point address is the same as the center point address. ("Start Point - Center Point Address Same Value (error code: 1A65H)")
- When the end point address is the same as the center point address. ("End Point - Center Point Address Same Value (error code: 1A67H)")

- Operation of circular interpolation control with "2: Radius specification (mcRadius)"
  In relative value circular interpolation control operation with radius specification, positioning is carried out from the current position at start (start point address) to the movement amount address (end point address) set in End point (EndPoint) in an arc path having the radius set in Sub point (AuxPoint). The resulting control path is an arc having as its center the intersection point between the perpendicular bisectors of the straight line between the current position at start and the end point address and the set radius. For relative value circular interpolation control with radius specification, the arc path is set using Path selection (PathChoice). For details, refer to the following.
  ☞ Page 310 Path selection (PathChoice)



Restriction

2-axis circular interpolation control with center point specification cannot be set in the following cases.

- When the radius exceeds "2147483647.0", the maximum radius for which 2-axis circular interpolation control is possible. ("Out of Radius Range (error code: 1A6CH)")
- When the end point address exceeds positioning range. ("Out of End Point Address Range (error code: 1A6DH)")
- When the start point address is the same as the end point address. ("Start Point - End Point Address Same Value (error code: 1A66H)")
- When the distance between the start point address and the end point address is larger than the radius. ("Radius Setting Error (error code: 1A6EH)")

■Sub point (AuxPoint)

Sets the position of the sub point (border point, center point, or radius) used to perform circular interpolation.

Sub point (AuxPoint) has 16 array elements. The contents set in Sub point (AuxPoint) differ depending on the setting of Circular interpolation mode (CircMode).

Circular interpolation mode (CircMode)	Setting range	Sub point (AuxPoint) setting
0: Border point specification (mcBorder)	-10000000000.0 to 10000000000.0	Sub point (AuxPoint) sets the border point on the arc connecting the start point and the end point. AuxPoint[0..15] indicates the sub point position of structuring axis 1 to 16. • Sub point (AuxPoint) of structuring axes not set to the circular interpolation axes is ignored. • Sub point (AuxPoint) of axes after all structuring axes set to the circular interpolation axes can be omitted.
1: Center point specification (mcCenter)		Sub point (AuxPoint) sets the center point of the arc. AuxPoint[0..15] indicates the sub point position of structuring axis 1 to 16. • Sub point (AuxPoint) of structuring axes not set to the circular interpolation axes is ignored. • Sub point (AuxPoint) of axes after all structuring axes set to the circular interpolation axes can be omitted.
2: Radius specification (mcRadius)	0.000001 to 2147483647.0	Sub point (AuxPoint) sets the radius of the arc. AuxPoint[0] indicates the radius of the circle. • Inputs after AuxPoint[1] are ignored and can be omitted.

Ex.

When setting structuring axis 2 and 3 to the circular interpolation axes and setting the relative position of the sub point for structuring axis 2 and 3 as 2000.0 and 3000.0 respectively.

CircAxes[0]:= 2;

CircAxes[1]:= 3;

AuxPoint[0]:= 2000.0;

AuxPoint[1]:= 3000.0;

AuxPoint[2]:= 0.0; *1

⋮

AuxPoint[15]:= 0.0; *1

*1 "AuxPoint[2]:= 0.0; to AuxPoint[15]:= 0.0;" can be omitted.

■End point (EndPoint)

Sets the movement amount from the current position at start to the end point.

End point (EndPoint) has 16 array elements.

EndPoint[0..15] indicates the end point position of structuring axis 1 to 16.

- End point (EndPoint) for structuring axes which are not set to the circular interpolation axis are ignored.
- End point (EndPoint) after all of the structuring axes which are set to the circular interpolation axis can be omitted.

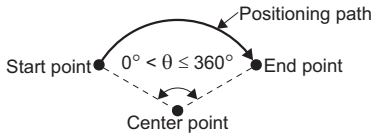
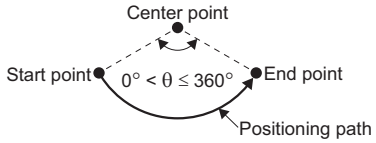
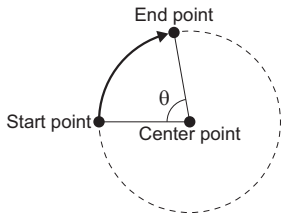
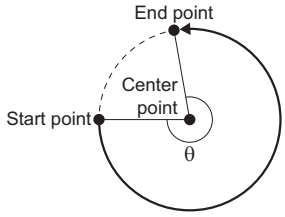
■Path selection (PathChoice)

Sets the rotation direction for circular interpolation. For the rotation direction, the first element of the array set in Circular interpolation axis (CircAxes) is regarded as the reference axis.

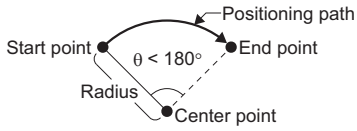
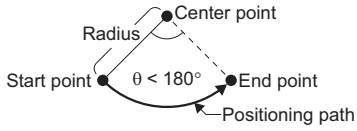
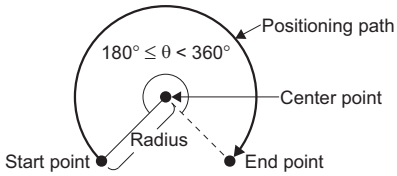
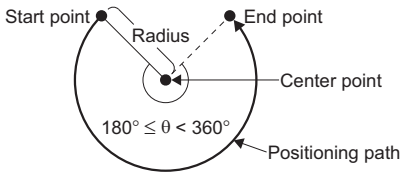
Path selection (PathChoice) is set when Circular interpolation mode (CircMode) is set to "1: Center point specification (mcCenter)" or "2: Radius specification (mcRadius)". When Circular interpolation mode (CircMode) is set to "0: Border point specification (mcBorder)", this input is omitted.

The following shows the setting value, setting description, arc center angle that can be controlled, and path of Path selection (PathChoice) according to the setting of Circular interpolation mode (CircMode).

• Path for "1: Center point specification (mcCenter)"

Setting value	Setting description	Arc center angle that can be controlled	Path
0: CW (mcCW)	CW	$0° < \theta \leq 360°$	
1: CCW (mcCCW)	CCW	$0° < \theta \leq 360°$	
2: Shortcut (mcShortWay)	Shortcut • When the start point, center point, and end point are on one straight line, becomes CW. (A semicircle is drawn in the CW direction.) • When the start point is same as the end point, "Path Choice Setting Incorrect (error code: 1AB7H)" occurs.	$0° < \theta \leq 180°$	
3: Detour (mcLongWay)	Detour • When the start point, center point, and end point are on one straight line, becomes CW. (A semicircle is drawn in the CW direction.) • When the start point is same as the end point, becomes CW. (A perfect circle is drawn in CW direction.)	$180° \leq \theta \leq 360°$	

• Path for "2: Radius specification (mcRadius)"

Setting value	Setting description	Arc center angle that can be controlled	Path
0: CW (mcCW)	CW • When the start point is same as the end point, "Path Choice Setting Incorrect (error code: 1AB7H)" occurs.	0° < θ < 180°	
1: CCW (mcCCW)	CCW • When the start point is same as the end point, "Path Choice Setting Incorrect (error code: 1AB7H)" occurs.	0° < θ < 180°	
4: CW Detour (mcCWLongWay)	CW detour • When the start point, end point, and center point are on one straight line, a semicircle is drawn.	180° ≤ θ < 360°	
5: CCW Detour (mcCCWLongWay)	CCW detour • When the start point, end point, and center point are on one straight line, a semicircle is drawn.	180° ≤ θ < 360°	

■Velocity (Velocity)

Sets the maximum speed of the path used in MCv_MoveCircularInterpolateRelative (Relative Value Circular Interpolation Control).

Setting range
0.0, 0.0001 to 2500000000.0[*1*2]

*1 To perform floating-point operation, the lower limit value of the commanded velocity is limited by the following restrictions. If the velocity that is converted from the target velocity into the operation cycle is less than "0.00001", "Out of Calculation Cycle Converted Speed Range (error code: 1AE4H)" occurs. (If the velocity is changed, "Out of Calculation Cycle Converted Speed Range Warning (warning code: 0D2FH)" occurs instead). To improve the precision of floating-point operation, change Position command unit (AxisName.Pr.Unit_Position) and Speed command unit (AxisName.Pr.Unit_Velocity) and perform settings so that the velocity converted into the operation cycle is not less than "0.00001".

*2 When a Motion control FB with a set velocity of "0.0" performs a multiple start, the set velocity of the previous Motion control FB is used.

■Acceleration (Acceleration)

Sets the acceleration used in MCv_MoveCircularInterpolateRelative (Relative Value Circular Interpolation Control).

The setting range differs depending on the setting for Acceleration/deceleration method setting (Options (Options): Bit 0 to 2).

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000[*1], positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	0.000000[*1], positive numbers from 0.000001 to 8400.0[s]

*1 The operation performed at start varies depending on the setting of Operation selection at start acceleration/deceleration 0 (AxisName.Pr.AccelerationZeroBehavior). When changing the acceleration or deceleration, the changes are not accepted.

■Deceleration (Deceleration)

Sets the deceleration used in MCv_MoveCircularInterpolateRelative (Relative Value Circular Interpolation Control).
Specifies the setting range if Acceleration/deceleration method setting (Options (Options): Bit 0 to 2) is set to "0: Acceleration/deceleration specification method (mcAccDec)". Deceleration (Deceleration) is not used if the acceleration/deceleration method is set to "1: Acceleration/deceleration time-fixed method (mcFixedTime)".

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000*1, positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	Not used

*1 The operation performed at start varies depending on the setting of Operation selection at start acceleration/deceleration 0 (AxisName.Pr.AccelerationZeroBehavior). When changing the acceleration or deceleration, the changes are not accepted.

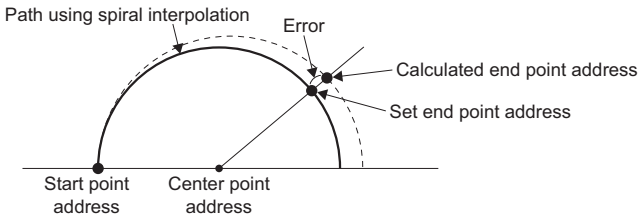
■Jerk (Jerk)

Sets the jerk used in MCv_MoveCircularInterpolateRelative (Relative Value Circular Interpolation Control).
Specifies the setting range if Acceleration/deceleration method setting (Options (Options): Bit 0 to 2) is set to "0: Acceleration/deceleration specification method (mcAccDec)". Jerk (Jerk) is not used if the acceleration/deceleration method is set to "1: Acceleration/deceleration time-fixed method (mcFixedTime)".

Acceleration/deceleration method setting (Options (Options): Bit 0 to 2)	Setting range
0: Acceleration/deceleration specification method (mcAccDec)	0.0000, positive numbers from 0.0001 to 2147483647.0[U/s^2]
1: Acceleration/deceleration time-fixed method (mcFixedTime)	Not used

■Circular interpolation error tolerance (CircularErrorTolerance)

In circular interpolation control with center point specification, the arc path calculated from the start position and center position may deviate from the end position set in End point (EndPoint).
Circular interpolation error tolerance (CircularErrorTolerance) sets the allowable width of errors between the calculated arc path and the end position.

Setting range	Description
Calculated error ≤ Circular interpolation error tolerance (CircularErrorTolerance)	Circular interpolation control to the set end point address is performed while the error compensation is carried out by spiral interpolation. 
Calculated error > Circular interpolation error tolerance (CircularErrorTolerance)	At the positioning start, "Circular Interpolation Error Allowable Value Over (error code: 1A71H)" occurs and the control will not start. The machine will immediately stop if the error is detected during positioning control.

■Buffer mode (BufferMode)

Sets the operation used to perform multiple start (buffer mode).

The following buffer modes are settable in MCv_MoveCircularInterpolateRelative (Relative Value Circular Interpolation Control).

Setting value	Description
0: Aborting (mcAborting)	Aborts (cancels) the under-control FB and executes the next FB immediately.
1: Buffered (mcBuffered)	Buffers the next FB on the under-control FB. If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after completion of the under-control FB.
2: BlendingLow (mcBlendingLow)	Buffers the next FB on the under-control FB.*1 If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. When the under-control FB reaches the target position, the lower target velocity between the under-control FB and the buffering FB is used as the switching speed.
3: BlendingPrevious (mcBlendingPrevious)	Buffers the next FB on the under-control FB.*1 If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. When the under-control FB reaches the target position, the switching speed changes to the target velocity of the under-control FB.
4: BlendingNext (mcBlendingNext)	Buffers the next FB on the under-control FB.*1 If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. When the under-control FB reaches the target position, the switching speed changes to the target velocity of the buffering FB.
5: BlendingHigh (mcBlendingHigh)	Buffers the next FB on the under-control FB.*1 If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. When the under-control FB reaches the target position, the switching speed changes to the higher target velocity value between the under-control FB and the buffering FB.

*1 Stops are not performed between the under-control FB and the buffering FB.

Point

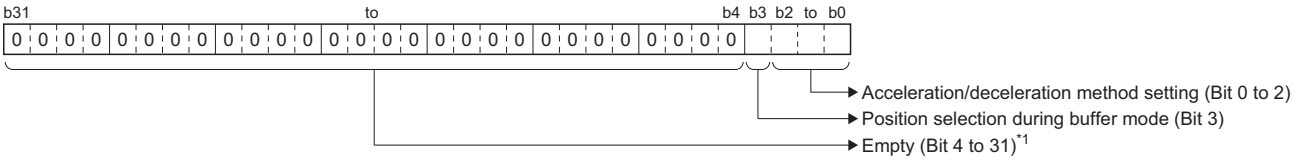
For details of multiple start (buffer mode), refer to the following.

MELSEC iQ-R Motion Module User's Manual (Application)

■Options (Options)

Sets the function options used for MCv_MoveCircularInterpolateRelative (Relative Value Circular Interpolation Control) by specifying the bit.

Contents that are set by specifying the bit are shown below.



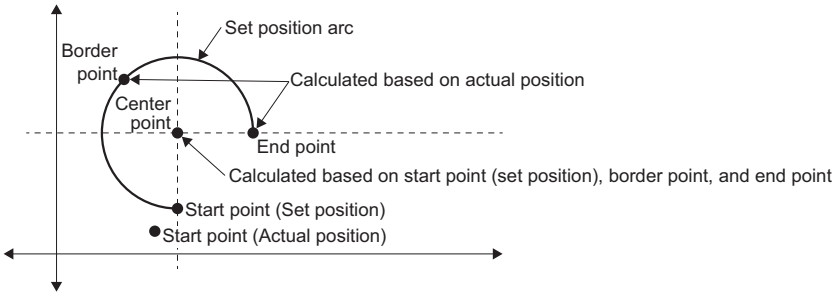
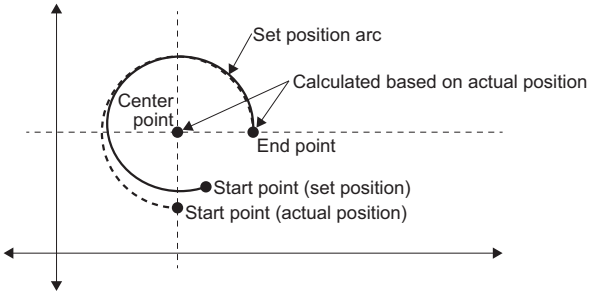
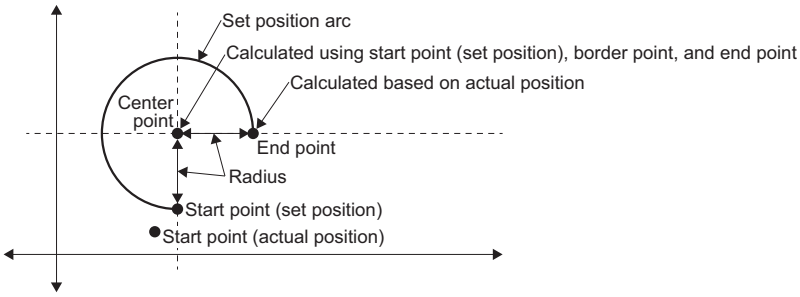
*1 Set empty areas to "0". When a value other than "0" is set, "Out of Options Range (error code: 1A4EH)" occurs.

Bit	Name	Description
0 to 2	Acceleration/deceleration method setting	Sets the acceleration/deceleration method used to perform control. • 0: Acceleration/deceleration specification method (mcAccDec) • 1: Acceleration/deceleration time-fixed method (mcFixedTime)
3	Position selection during buffer mode	Sets the position when executing relative positioning control by multiple start. • 0: Set position • 1: Actual position *: This bit is valid when setting "0: Aborting (mcAborting)" in Buffer mode (BufferMode).

• Acceleration/deceleration method setting (Bit 0 to 2)

Setting value	Description
0: Acceleration/deceleration specification method (mcAccDec)	An acceleration/deceleration method using Acceleration (Acceleration), Deceleration (Deceleration), and Jerk (Jerk) set in MCv_MoveCircularInterpolateRelative (Relative Value Circular Interpolation Control).
1: Acceleration/deceleration time-fixed method (mcFixedTime)	An acceleration/deceleration method using the acceleration/deceleration time specified in MCv_MoveCircularInterpolateRelative (Relative Value Circular Interpolation Control) regardless of the velocity. The acceleration/deceleration time does not use Acceleration (Acceleration), Deceleration (Deceleration), and Jerk (Jerk).

• Position selection during buffer mode (Bit 3)

Setting value	Description
0: Set position	Performs relative positioning control from the set position.
1: Actual position	Performs relative positioning control from the actual position. The sub point (border point, center point, and radius) and end point are calculated based on the actual position, and the set position is used as the start point. The operation for each type of Circular interpolation mode (CircMode) is shown below. ■When "0: Border point specification (mcBorder)" The border point and end point is calculated based on the actual position. The center point is determined by the start point (set position), border point, and end point.  ■When "1: Center point specification (mcCenter)" The center point and end point are calculated based on the actual position. The arc of the circle may be an elliptic circle in some cases.  ■When "2: Radius specification (mcRadius)" The end point is calculated based on the actual position. The center point is determined by the start point (set position), radius, and end point. 

■Necessary objects

When using MCv_MoveCircularInterpolateRelative (Relative Value Circular Interpolation Control), set the following slave objects for all structuring axes in the specified axes group.

• Target position (607AH)

If there is a structuring axis present that does not have the slave object set, "Necessary Slave Object Unset (error code: 1AA8H)" occurs and the axis will not start.

For details of slave object settings, refer to the following.

📖MELSEC iQ-R Motion Module User's Manual (Application)

Precautions

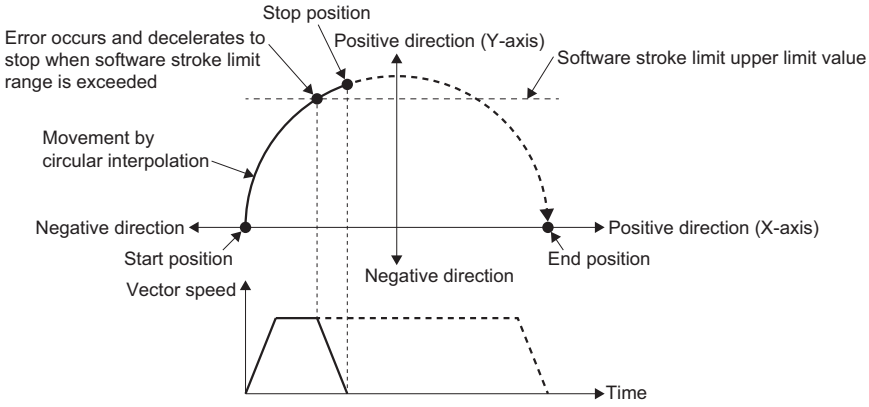
- Circular interpolation including an axis whose stroke limit is set to invalid cannot be performed. "Software Stroke Limit Invalid at Circular Interpolation (error code: 1A72H)" occurs and the operation does not start.
- When the interpolation path exceeds the stroke limit range during interpolation operation, "Software Stroke Limit Over (Forward Direction) (error code: 1A03H)" or "Software Stroke Limit Over (Reverse Direction) (error code: 1A04H)" occurs and the operation will stop.

Ex.

When the upper limit of the software stroke limit is exceeded in the positive direction of the Y-axis

For an immediate stop, stops at the same time that the error occurs.

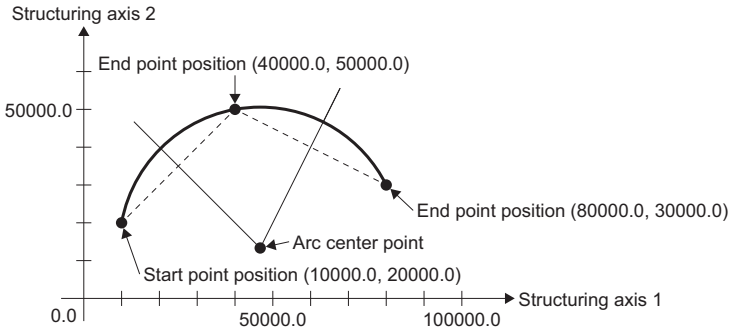
For a deleleration stop, decelerates to a stop along the arc of the circle as shown below.



Program example

In the following program example, the circular interpolation control start (bCircularInterpolateCMD) is set to TRUE and Axes group 1 (AxesGroup001) is enabled, following which 2-axis circular interpolation (sub point designation) using relative positioning control is performed for Axis group 1 (AxesGroup001) using the settings shown below.

- Operation



- Settings

Item	Setting value	
	Structuring axis 1	Structuring axis 2
Sub point	30000.0	30000.0
End point	70000.0	10000.0
Velocity	50000.0	
Acceleration	50000.0	
Deceleration	50000.0	
Jerk	0.0	
Circular interpolation error tolerance	100.0	

■Axes

Axis No.	Label name	Data type	Comment
1	Axis0001	AXIS_REF	Axis 1
2	Axis0002	AXIS_REF	Axis 2

■Axes group

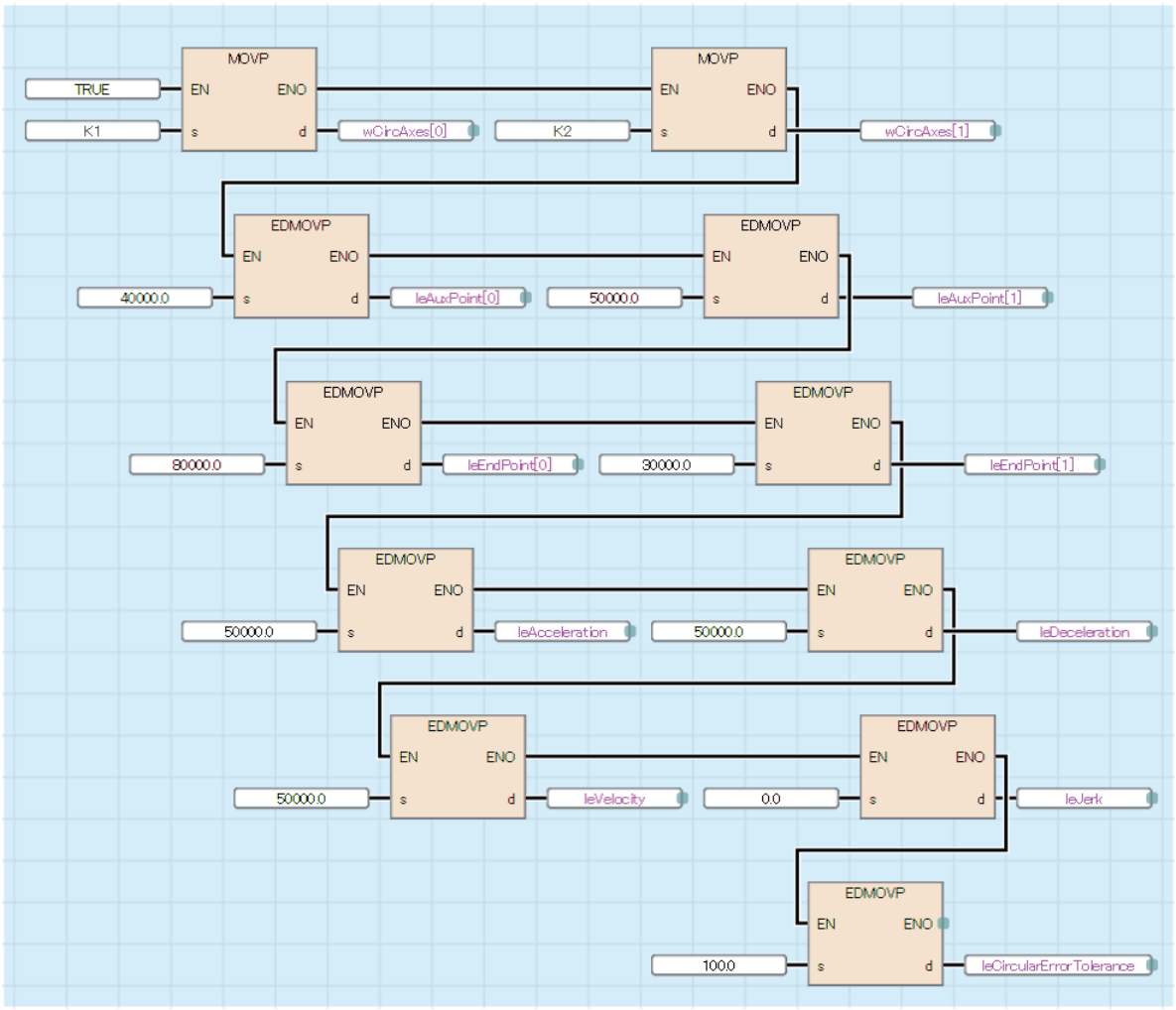
Axes group No.	Label name	Data type	Comment
1	AxesGroup001	AXES_GROUP_REF	Axes group 1

■Labels used

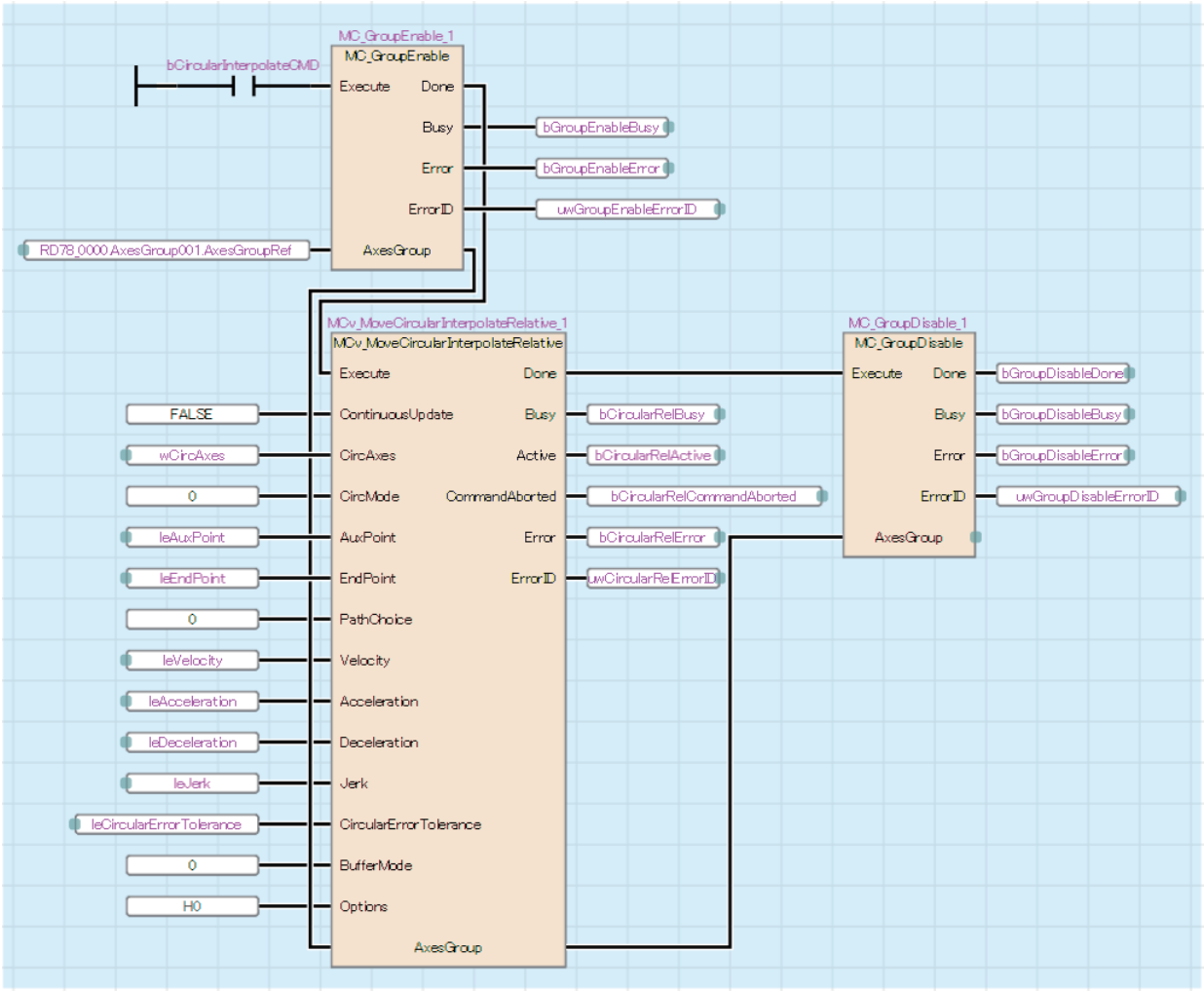
Label name	Data type	Comment
MCv_MoveCircularInterpolateRelative_1	MCv_MoveCircularInterpolateRelative	Relative Value Circular Interpolation Control FB
wCircAxes	Word[signed](0..1)	Circular interpolation axis
leAuxPoint	Double-precision real number(0..15)	Sub point
leEndPoint	Double-precision real number(0..15)	End point
leVelocity	Double-precision real number	Velocity
leAcceleration	Double-precision real number	Acceleration
leDeceleration	Double-precision real number	Deceleration
leJerk	Double-precision real number	Jerk
leCircularErrorTolerance	Double-precision real number	Circular interpolation error tolerance
bCircularRelDone	Bit	Execution completion
bCircularRelBusy	Bit	Executing
bCircularRelActive	Bit	Controlling
bCircularRelCommandAborted	Bit	Abortion of execution
bCircularRelError	Bit	Error
uwCircularRelErrorID	Word [unsigned]/bit string [16-bit]	Error code
MC_GroupEnable_1	MC_GroupEnable	Axes Group Enabled FB
bCircularInterpolateCMD	Bit	Circular interpolation control start
bGroupEnableDone	Bit	Axes Group Enabled Completion
bGroupEnableBusy	Bit	Executing
bGroupEnableError	Bit	Error
uwGroupEnableErrorID	Word [unsigned]/bit string [16-bit]	Error code
MC_GroupDisable_1	MC_GroupDisable	Axes Group Disabled FB
bGroupDisableDone	Bit	Axes Group Disabled Completion
bGroupDisableBusy	Bit	Executing
bGroupDisableError	Bit	Error
uwGroupDisableErrorID	Word [unsigned]/bit string [16-bit]	Error code

■FBD/LD program (CPU module side)

• Data setting for 2-axis circular interpolation control



• Axes Group Enabled/Circular interpolation control (Border point specification)/Axis Group Disabled



■ST program (Motion module side)

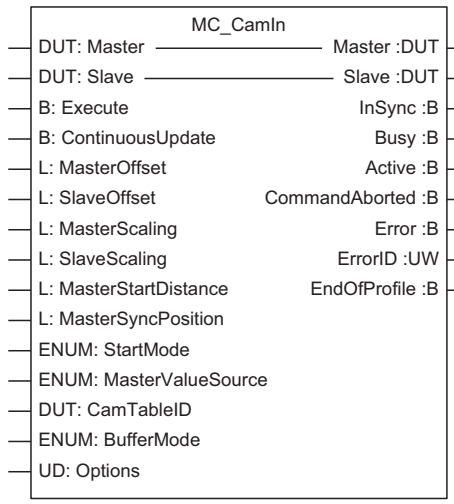
```
//-----Data setting for 2-axis circular interpolation control-----
wCircAxes[0]:= 1;
wCircAxes[1]:= 2;
leAuxPoint[0]:= 30000.0;
leAuxPoint[1]:= 30000.0;
leEndPoint[0]:= 70000.0;
leEndPoint[1]:= 10000.0;
leVelocity:= 50000.0;
leAcceleration:= 50000.0;
leDeceleration:= 50000.0;
leJerk:= 0.0;
leCircularErrorTolerance:= 100.0;

//-----Axes Group Enabled-----
MC_GroupEnable_1(
   AxesGroup:= AxesGroup001.AxesGroupRef ,
   Execute:= bCircularInterpolateCMD ,
   Done=> bGroupEnableDone ,
   Busy=> bGroupEnableBusy ,
   Error=> bGroupEnableError ,
   ErrorID=> uwGroupEnableErrorID
);
```

```
//-----Circular interpolation control (Border point specification)-----
MCv_MoveCircularInterpolateRelative_1(
    AxesGroup:= AxesGroup001.AxesGroupRef ,
    Execute:= bGroupEnableDone ,
    ContinuousUpdate:= FALSE ,
    CircAxes:= wCircAxes ,
    CircMode:= MC_CIRC_MODE__mcBorder ,
    AuxPoint:= leAuxPoint ,
    EndPoint:= leEndPoint ,
    PathChoice:= MC_CIRC_PATHCHOICE__mcCW ,
    Velocity:= leVelocity ,
    Acceleration:= leAcceleration ,
    Deceleration:= leDeceleration ,
    Jerk:= leJerk ,
    CircularErrorTolerance:= leCircularErrorTolerance ,
    BufferMode:= MC_BUFFER_MODE__mcAborting ,
    Options:= H00000000 ,
    Done=> bCircularRelDone ,
    Busy=> bCircularRelBusy ,
    Active=> bCircularRelActive ,
    CommandAborted=> bCircularRelCommandAborted ,
    Error=> bCircularRelError ,
    ErrorID=> uwCircularRelErrorID
);

//-----Axes Group Disabled-----
MC_GroupDisable_1(
    AxesGroup:= AxesGroup001.AxesGroupRef ,
    Execute:= bCircularRelDone ,
    Done=> bGroupDisableDone ,
    Busy=> bGroupDisableBusy ,
    Error=> bGroupDisableError ,
    ErrorID=> uwGroupDisableErrorID
);
```

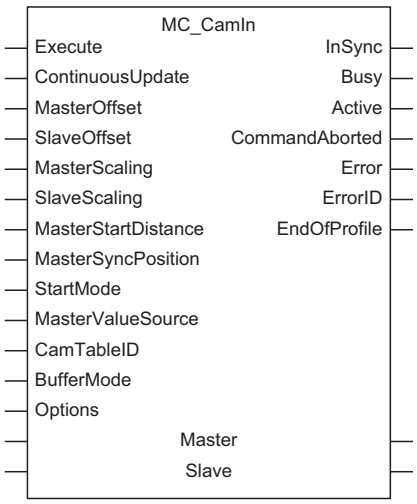
MC_CamIn

Starts cam operation according to the specified cam data.

Ladder



FBD/LD



ST

```
MC_CamIn(
  Master:= ?AXIS_REF? ,
  Slave:= ?AXIS_REF? ,
  Execute:= ?BOOL? ,
  ContinuousUpdate:= ?BOOL? ,
  MasterOffset:= ?LREAL? ,
  SlaveOffset:= ?LREAL? ,
  MasterScaling:= ?LREAL? ,
  SlaveScaling:= ?LREAL? ,
  MasterStartDistance:= ?LREAL? ,
  MasterSyncPosition:= ?LREAL? ,
  StartMode:= ?INT? ,
  MasterValueSource:= ?INT? ,
  CamTableID:= ?MC_CAM_ID? ,
  BufferMode:= ?INT? ,
  Options:= ?DWORD? ,
  InSync=> ?BOOL? ,
  Busy=> ?BOOL? ,
  Active=> ?BOOL? ,
  CommandAborted=> ?BOOL? ,
  Error=> ?BOOL? ,
  ErrorID=> ?WORD? ,
  EndOfProfile=> ?BOOL?
);
```

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Cam Operation Start	96	48	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
Master	Master axis	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)
Slave	Slave axis	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Execute command	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MC_ CamIn (Cam Operation Start).
ContinuousUpdate	Continuous update	BOOL	At start	TRUE, FALSE	FALSE	This variable sets whether to enable or disable continuous change of Master axis offset (MasterOffset), Slave axis offset (SlaveOffset), Master axis scaling (MasterScaling), Slave axis scaling (SlaveScaling), and Cam table ID (CamTableID). • FALSE: Disable • TRUE: Enable
MasterOffset	Master axis offset	LREAL	At start/ Retrigger possible/ Continuous update possible	-10000000000.0 to 10000000000.0	0.0	This variable shifts the phase of Master axis (Master) by the offset amount. For details, refer to the following. ☞ Page 333 Master axis offset (MasterOffset)

Input variable	Name	Data type	Input	Setting range	Default value	Description
SlaveOffset	Slave axis offset	LREAL	At start/ Retrigger possible/ Continuous update possible	-10000000000.0 to 10000000000.0	0.0	This variable shifts the displacement of the slave axis (Slave) by the offset amount. For details, refer to the following. ☞ Page 334 Slave axis offset (SlaveOffset)
MasterScaling	Master axis scaling	LREAL	At start/ Retrigger possible/ Continuous update possible	0.01 to 10.0	1.0	This variable expands/reduces the cam table length per cycle. For details, refer to the following. ☞ Page 335 Master axis scaling (MasterScaling)
SlaveScaling	Slave axis scaling	LREAL	At start/ Retrigger possible/ Continuous update possible	0.01 to 10.0	1.0	This variable expands/reduces the stroke amount of the cam table. For details, refer to the following. ☞ Page 335 Slave axis scaling (SlaveScaling)
MasterStartDistance	Master axis follow-up distance	LREAL	At start	-10000000000.0 to 10000000000.0	0.0	This variable sets the position (the relative position from Master axis synchronization start position (MasterSyncPosition)) of Master axis (Master) that will start synchronization by Output axis (OutputData).
MasterSyncPosition	Master axis synchronization start position	LREAL	At start	-10000000000.0 to 10000000000.0	0.0	This variable sets the position of Master axis (Master) that will start synchronization by Current value per cycle (InputPerCycle). *: If specifying target of Master axis synchronization start position pass check (Options (Options): Bit 21) is "1: Master axis command current position, or the feedback position", the valid range is within the ring counter range.
StartMode	Start mode	INT (MC_START_MODE)	At start	0, 1	0	This variable sets the timing for starting cam operation. • 0: Immediate (mcImmediate) • 1: Absolute (mcAbsolute) For details, refer to the following. ☞ Page 336 Start mode (StartMode)
MasterValueSource	Master axis data source selection	INT (MC_SOURCE)	At start	1, 2, 101, 102	1	This variable sets the data source of Master axis (Master). • 1: Set value (mcSetValue) • 2: Actual value (mcActualValue) • 101: Latest set value (mcLatestSetValue) • 102: Latest actual value (mcLatestActualValue) For details, refer to the following. ☞ Page 338 Master axis data source selection (MasterValueSource)
CamTableID	Cam table ID	MC_CAM_ID	At start/ Retrigger possible/ Continuous update possible	1 to 6000	0	This variable sets the ID for the cam. Open the cam ID to the open area with MC_CamTableSelect (Cam Table Selection) before use.
BufferMode	Buffer mode	INT (MC_BUFFER_MODE)	At start	0, 1	0	This variable sets the buffer mode. • 0: Aborting (mcAborting) • 1: Buffered (mcBuffered) For details, refer to the following. ☞ Page 339 Buffer mode (BufferMode)
Options	Options	DWORD(HEX)	At start	00000000H to 00210000H	00000000H	This variable sets the function options for MC_CamIn (Cam Operation Start) by specifying the bit. For details, refer to the following. ☞ Page 340 Options (Options)

■Output variables

Output variable	Name	Data type	Default value	Description
InSync	In synchronization	BOOL	FALSE	This variable becomes TRUE when Output value (OutputData) starts synchronization.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MC_ CamIn (Cam Operation Start) is executed.
Active	Controlling	BOOL	FALSE	This variable becomes TRUE when Current value per cycle (InputPerCycle) starts synchronization.
CommandAborted	Abortion of execution	BOOL	FALSE	This variable becomes TRUE when execution of MC_ CamIn (Cam Operation Start) is aborted. This variable becomes FALSE when Execute command (Execute) becomes FALSE.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. MELSEC iQ-R Motion Module User's Manual (Application)
EndOfProfile	Cam cycle completion	BOOL	FALSE	After Controlling (Active) becomes TRUE, this variable becomes TRUE only for 1 execution cycle of the POU (Program Component) that calls FB each time movement equal to the cycle length occurs. For details, refer to the following. Page 340 Cam cycle completion (EndOfProfile)

■Public variables

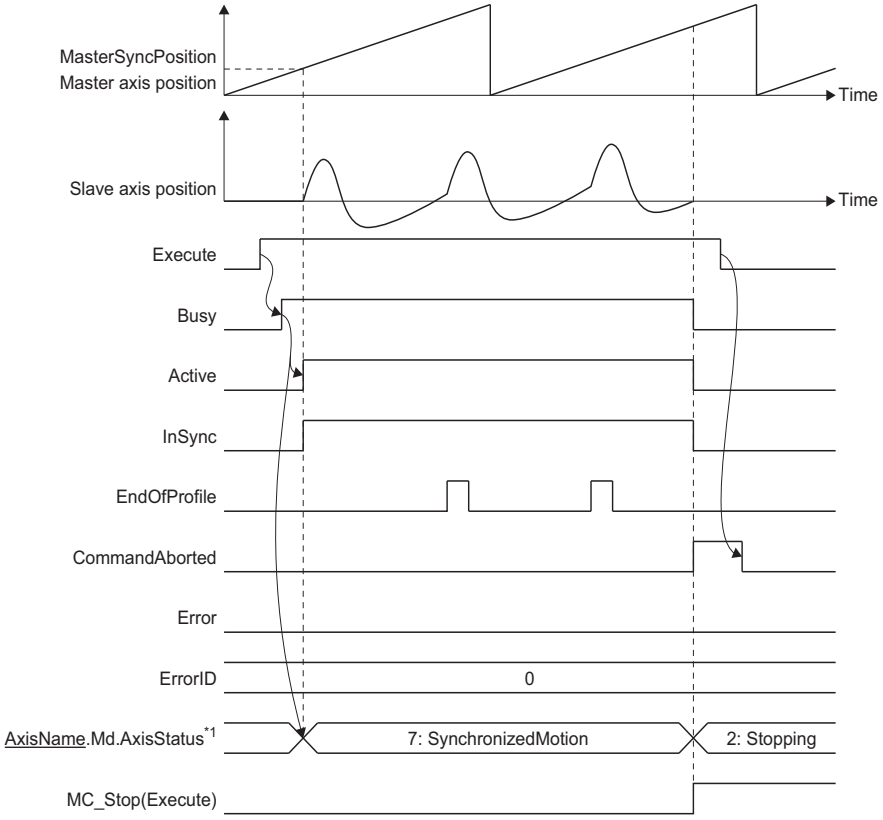
Public variable	Name	Data type	Default value	Description
InputPerCycle	Current value per cycle	LREAL	0.0	This variable stores the current value per cycle. For details, refer to the following. Page 340 Current value per cycle (InputPerCycle)
Reference	Reference value	LREAL	0.0	This variable stores the reference value. For details, refer to the following. Page 341 Reference value (Reference)
OutputData	Output value	LREAL	0.0	This variable stores the output value. For details, refer to the following. Page 343 Output value (OutputData)
InstanceID	Instance ID	WORD(UINT)	0	The instance ID. This variable is automatically set by the system when an instance is created. This instance ID is used in FB input, etc.

Processing details

- MC_CamIn (Cam Operation Start) sets Master axis offset (MasterOffset), Slave axis offset (SlaveOffset), Master axis scaling (MasterScaling), Slave axis scaling (SlaveScaling), Master axis follow-up distance (MasterStartDistance), Master axis synchronization start position (MasterSyncPosition), Start mode (StartMode), Master axis data source selection (MasterValueSource), Cam tableID (CamTableID), and Buffer mode (BufferMode), then executes the cam operation.
- To stop the operation, perform MC_Stop (Forced Stop).

■Timing chart

- When the operation completes normally
  - When Start Mode (StartMode) is set to "1: Absolute (mcAbsolute)" and Master axis absolute coordinate (MasterAbsolute) set in MC_CamTableSelect (Cam Table Selection) is "FALSE (Relative coordinate)"



*1 Axis status of Slave axis (Slave).

- When an error occurs

  For the timing chart when an error occurs, refer to the following.

  ☞ Page 11 Basic operation of Execute command (Execute) type Motion control FBs

■Starting cam operation

By setting Start Mode (StartMode), the synchronization timing for cam operation and reflection timing for the cam control data can be set.

- Synchronization timing for cam operation

  The table below shows the synchronization timing for Current value per cycle (InputPerCycle), Reference value (Reference), and Output value (OutputData) based on Start mode (StartMode) and related parameters (Master axis synchronization start position (MasterSyncPosition) and Master axis follow-up distance (MasterStartDistance)).

Start mode (StartMode)*1	Timing for starting synchronization		
	Current value per cycle (InputPerCycle)	Reference value (Reference)	Output value (OutputData)
0: Immediate (mcImmediate)	Synchronization starts when MC_CamIn (Cam Operation Start) is executed.		
1: Absolute (mcAbsolute)	Synchronization starts when the master axis position*2 passes*3 Master axis synchronization start position (MasterSyncPosition). Master axis follow-up distance (MasterStartDistance) is ignored.		

*1 For details of Start mode (StartMode), refer to the following.
☞ Page 336 Start mode (StartMode)

*2 Set the position data to be used based on Specifying target of Master axis synchronization start position pass check (Options (Options): Bit 21). For details, refer to the following.
☞ Page 340 Options (Options)

*3 The position of the Master axis has crossed over Master axis synchronization start position (MasterSyncPosition)(or, the Master axis synchronization start position (MasterSyncPosition) + Master axis follow-up distance (MasterStartDistance)). When these values are the same, the Master axis is considered to have not passed, and should the Master axis move from this value, it will be considered to have passed.

- Timing of control change by FB restart/continuous update

  The table below shows the timing for reflecting changed parameters in control by FB restart/continuous update based on Start Mode (StartMode).

Start Mode (StartMode)	Timing for reflecting parameters
0: Immediate (mcImmediate)	Immediate
1: Absolute (mcAbsolute)	When the current value per cycle (InputPerCycle) passes the 1st point on the CamTable

■Cam table information

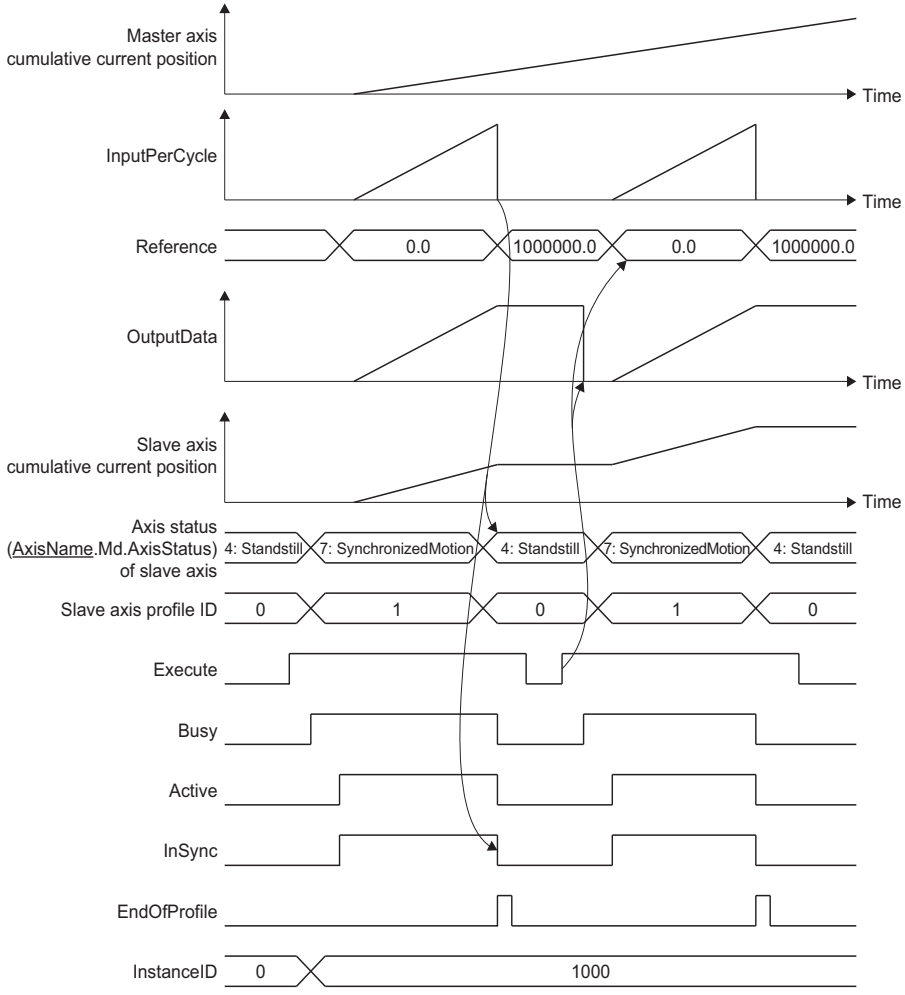
For cam table information, the following shows the cam operation at execution of MC_CamIn (Cam Operation Start) depending on the periodic mode, master axis absolute coordinate, and slave axis absolute coordinate set in MC_CamTableSelect (Cam Table Selection).

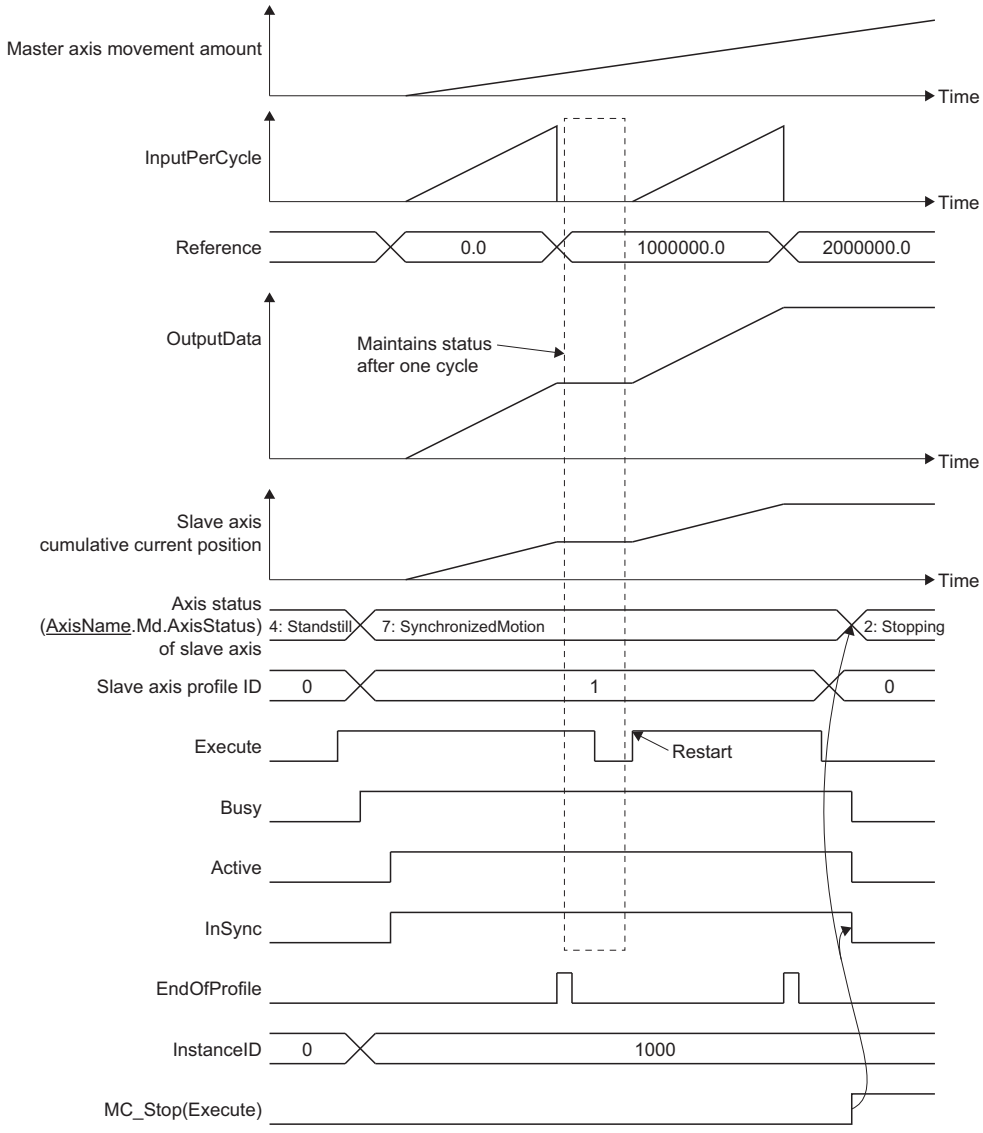
○: Used in control, ×: Not used in control

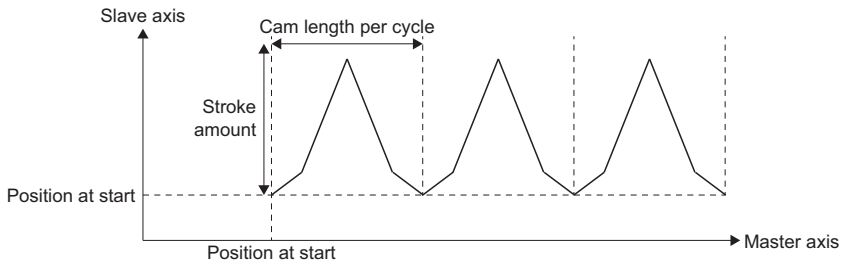
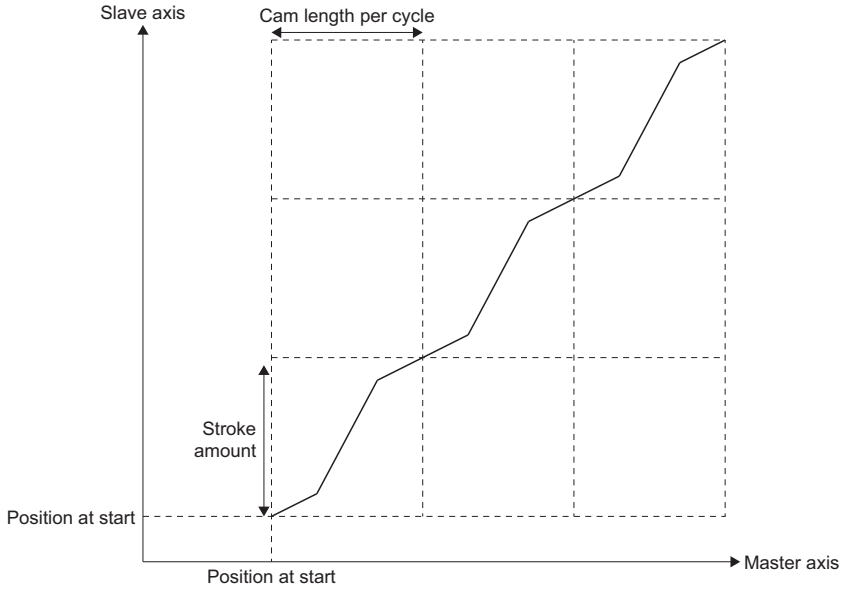
Name	Variable name	Specifying interpolation method (Interpolate)			Remark
		0: Linear interpolation	1: Specifying for each section	2: Spline interpolation	
Periodic	Periodic	○	○	○	These items can be set in the expand setting of the calculation profile using the engineering tool, or in the MC_CamTableSelect (Cam Table Selection) input variables.
Master axis absolute coordinate	MasterAbsolute	○	○	○	
Slave axis absolute coordinate	SlaveAbsolute	○	○	○	
Specifying interpolation method	Interpolate	○	○	○	These are items related to the control of data opened to the open area from a cam data format/rotary cutter format calculation profile.
Length per cycle	CycleLength	○	○	○	
Stroke amount	Stroke	○	○	○	
Start point	StartPoint	×	×	×	
Initial stroke amount	StartStroke	×	○	○	

• Periodic (Periodic)

The following shows the operation at execution of MC_CamIn (Cam Operation Start) by the setting of Periodic (Periodic).

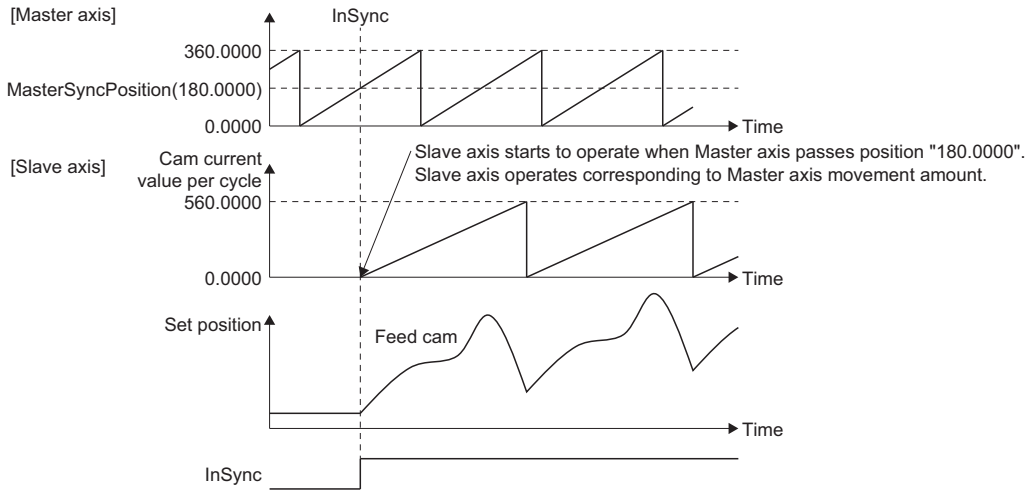
Periodic (Periodic)	Description
FALSE (Non periodic)	Operation is performed for one cycle only after Controlling (Active) becomes TRUE. Following this, the operation set in Specifying operation after one cycle operation (Options (Options): Bit 16) is performed. ■When Specifying operation after one cycle operation (Options (Options): Bit 16) is set to "0: Complete" Synchronization status is cancelled after one cycle, and Reference value (Reference) and Output value (OutputData) change to "0.0 (Default value)" when Execute command (Execute) is started again. 

Periodic (Periodic)	Description
FALSE (Non periodic)	■When Specifying operation after one cycle operation (Options (Options): Bit 16) is set to "1: Wait for restart" Synchronization status is maintained after one cycle, and Reference value (Reference) and Output value (OutputData) keep their value when Execute command (Execute) is started again. During a restart after one cycle has completed, even when values of input labels where continuous update is possible are changed, any input information will not be imported until Execute command (Execute) is started. If Execute command (Execute) is started, regardless of the Start mode (StartMode) settings, the slave axis will immediately start moving. While waiting for a restart after one cycle has completed, if a change to the current value per cycle is executed, this will be valid from the next cycle. 

Periodic (Periodic)	Description
TRUE (Periodic)	Continuously repeats cam table execution. The following shows the operation based on the cam operation (two-way cam or feed cam). ■Two-way cam When the position of Slave axis (Slave) used by Periodic for the cam table start point is the same each time, operates as a two-way cam.  ■Feed cam When the position of Slave axis (Slave) used by Periodic for the cam table start point has a different set position at the start point and the end point, operates as a feed cam. 

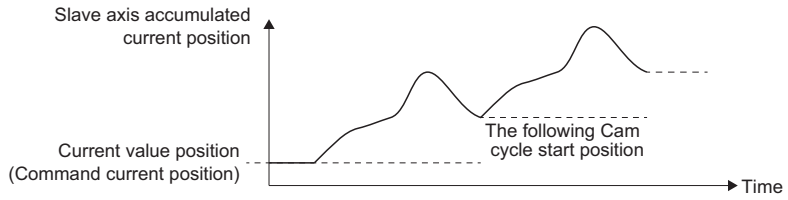
• Master axis absolute coordinate (MasterAbsolute)

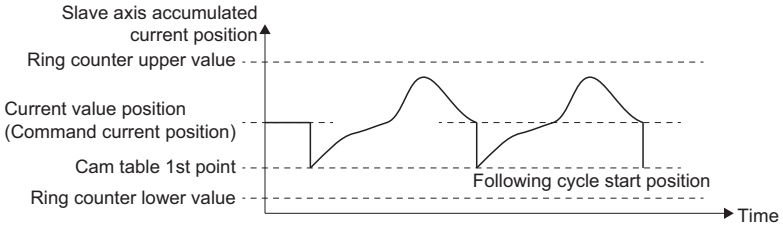
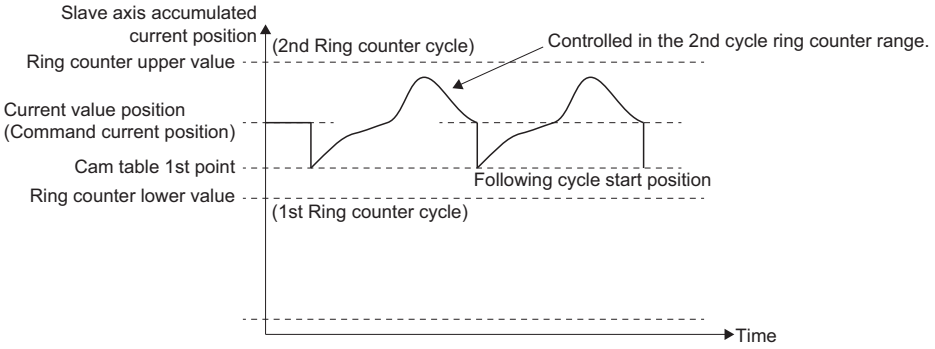
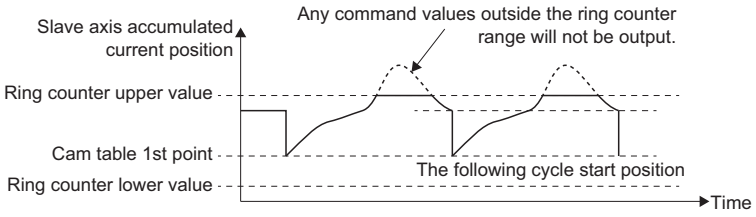
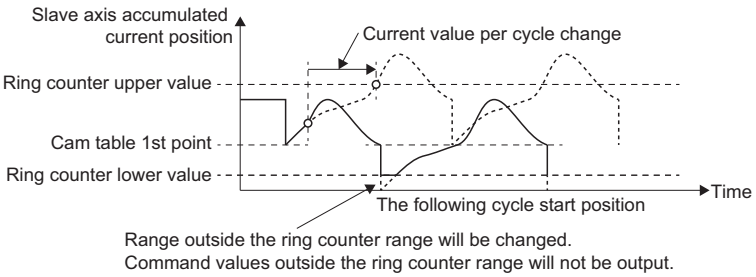
The following shows the cam operation depending on the setting of Master axis absolute coordinate (MasterAbsolute).

Master axis absolute coordinate (MasterAbsolute)	Description
FALSE (Relative coordinate)	The master axis synchronization start position (MasterSyncPosition) becomes the start point of the cam data. When In synchronization (InSync) becomes TRUE, cam operation is executed corresponding to the relative movement amount of Master axis (Master). Cam operation is executed continuously, even when the ring counters of Master axis (Master) and the cam table are not matched. <Example> When set to the following • Master axis: Ring counter of the current value: 0.0000 to 360.0000[degree] • Master axis synchronization start position (MasterSyncPosition): 180.0000[degree] • Cam: Cam length per cycle: 540.0000[degree] 

• Slave axis absolute coordinate (SlaveAbsolute)

The following shows the cam operation depending on the setting of Slave axis absolute coordinate (SlaveAbsolute).ve

Slave axis absolute coordinate (SlaveAbsolute)	Description
FALSE (Relative coordinate)	When In synchronization (InSync) becomes TRUE, the slave axis starts the operation of the cam table from the current position (set position). If Periodic (Periodic) of the cam table is set to "TRUE: Periodic", the next cycle starts from the stroke position (set position) where the last cycle completed. 

Slave axis absolute coordinate (SlaveAbsolute)	Description
TRUE (Absolute coordinate)	The cam is operated within the ring counter range including current values of the Slave axis (Slave) at the point when In synchronization (InSync) becomes TRUE. Command is output per operation cycle to return the command current position of the slave axis to the start point of the cam table when In synchronization (InSync) and when the cam cycle completion (EndOfProfile) become TRUE. If the command at this time is too large, the position command and speed command to the drive unit also become too large, causing a servo error ("AL. 035 (Command frequency error)") for the MR-J5(W)-G. The command current position of the Slave axis (Slave) will not output any values outside the ring counter range. **■When the Stroke Range is within the ring counter range of the Slave axis**  **■When the Stroke Range is within the ring counter range of the Slave axis (Cam started on the 2nd cycle)**  **■When the Stroke Range is outside the ring counter range of the Slave axis**  <Precautions> • If the ProfileControl (Calculation profile control) version is "earlier than 1.15", command values outside of the ring counter range will also be output. Values outside of the ring counter range will be rounded so that they are within the range. **■When current value per cycle is changed** • To compensate the Reference value (Reference) so that the Slave axis (Slave) does not operate, any range outside of the ring counter range will be changed. When the Slave axis (Slave) has moved and the current value per cycle has changed, change the Master axis offset (MasterOffset). 

- Length per cycle (CycleLength)

  For the length per cycle, the necessary input amount per cycle set. For details, refer to the following.

  ☞ Page 340 Current value per cycle (InputPerCycle)
- Stroke amount (Stroke)

  For the stroke amount, the stroke amount corresponding to the stroke ratio at 100% is set. For details, refer to the following.

  ☞ Page 343 Output value (OutputData)
- Start point (StartPoint) and Initial stroke amount (StartStroke)

  The Start Point (StartPoint) used for cam operation and the Initial stroke amount (StartStroke) will differ depending on the Specifying interpolation method (Interpolate).

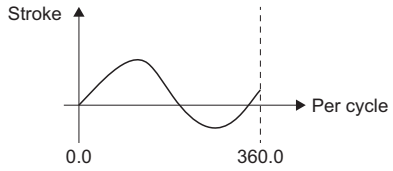
Specifying interpolation method (Interpolate)	Cam operation Start Point	Initial stroke amount for cam operation
0: Linear interpolation	Input value of the cam table 1st point	Output value of the cam table 1st point
1: Specifying for each section	Start Point (StartPoint) of the cam table 1st point	Initial stroke amount (StartStroke) of the cam table
2: Spline interpolation		

■Cam operation Start Point

Operating with a Current value per cycle (InputPerCycle) of "0.0".

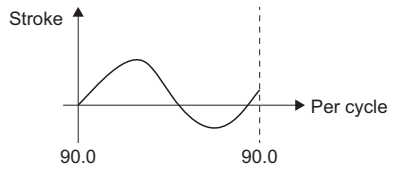
• Cam table

<Cam A (Cam operation Start Point "0.0")>



• Specifying interpolation method: 1: Specifying for each section
• Length per cycle: 360.0
• Start Point: 0.0
• Initial stroke amount: 0.0

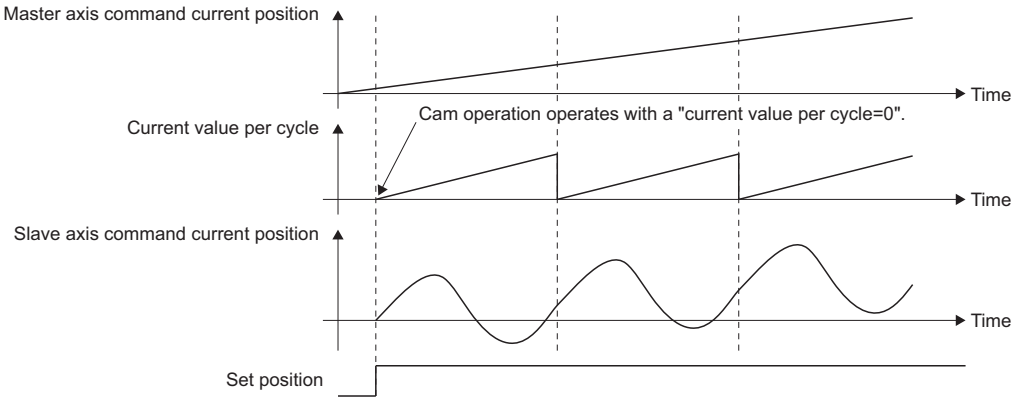
<Cam B (Cam operation Start Point "90.0")>



• Specifying interpolation method: 1: Specifying for each section
• Length per cycle: 360.0
• Start Point: 90.0
• Initial stroke amount: 0.0

• Cam operation

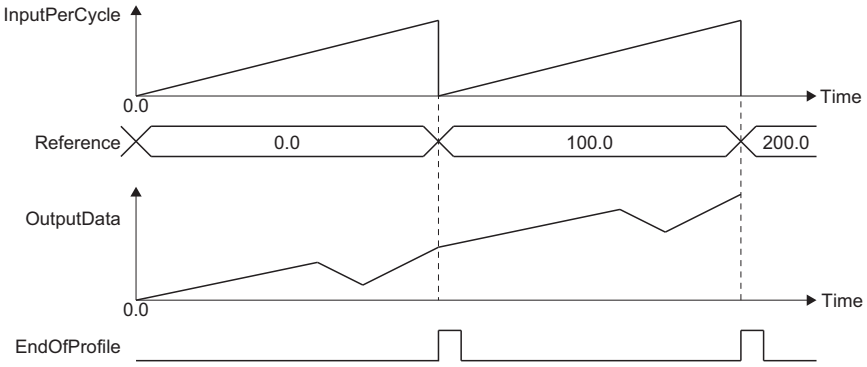
The operation in the following is performed regardless of whether cam A or cam B is used.



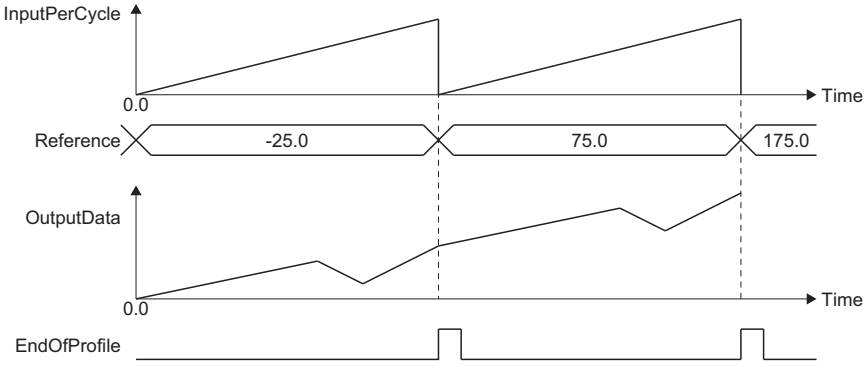
■Cam operation initial stroke amount

Operates so that the initial stroke amount equals the Output value (OutputData) at the start of synchronization. Because the initial stroke amount of the cam operation stops the Slave axis (Slave) from operating, the initial stroke amount from the reference axis (Reference) is decreased.

• When cam operation and Initial stroke amount (StartStroke) is "0.0"



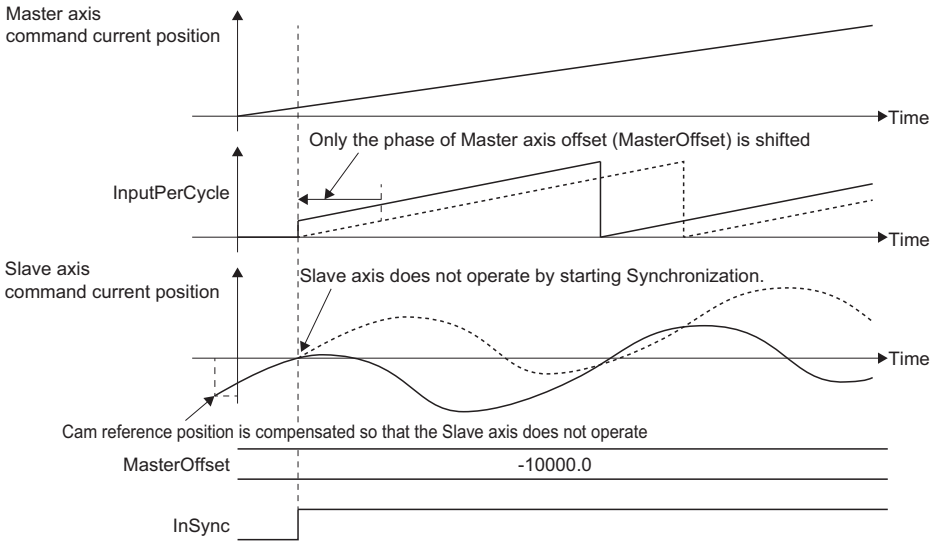
• When cam operation and Initial stroke amount (StartStroke) is "25.0"



■Master axis offset (MasterOffset)

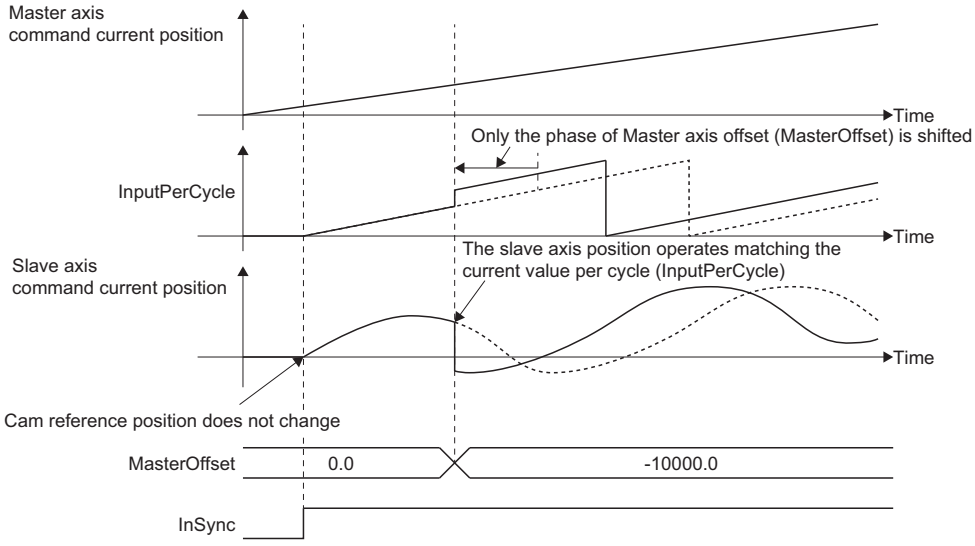
Setting Master axis offset (MasterOffset) shifts the phase of Master axis (Master) by the offset amount. (This does not affect Master axis follow-up distance (MasterStartDistance) or Master axis synchronization start position (MasterSyncPosition).) When a value other than "0.0" is set at operation start, the position of Master axis (Master) is changed at the rising edge of In synchronization (InSync) to the cam position per cycle position with the offset amount added. When this happens, the Reference value (Reference) is compensated so that the Slave axis (Slave) does not operate.

- When Start Mode (StartMode) is "0: Immediate (mcImmediate)"



When the master axis offset (MasterOffset) is changed while In Synchronization (InSync) is TRUE, the position of the Slave axis (Slave) operates matching the current value per cycle with Master axis Offset (MasterOffset) added.

- When Start Mode (StartMode) is "0: Immediate (mcImmediate)"

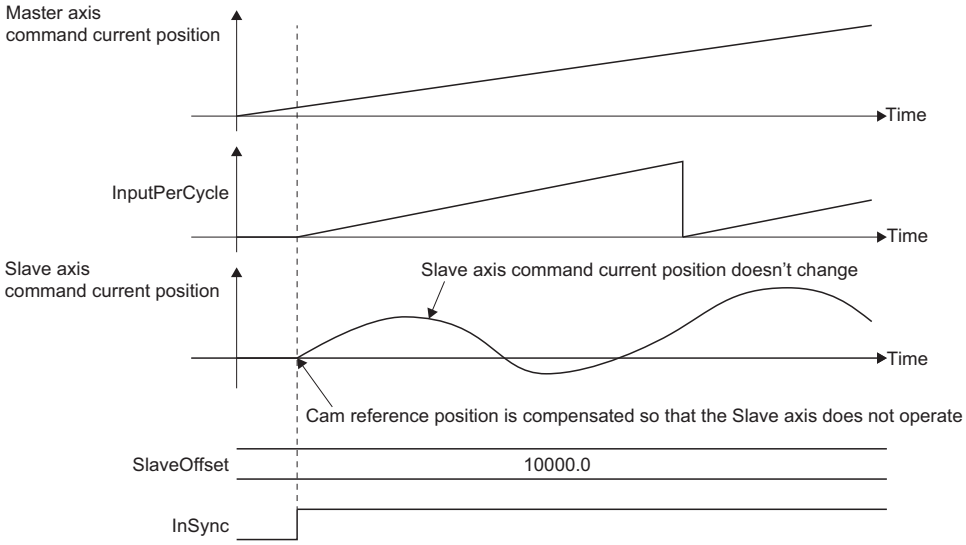


■Slave axis offset (SlaveOffset)

This variable shifts the displacement of the slave axis (Slave) by the offset amount.

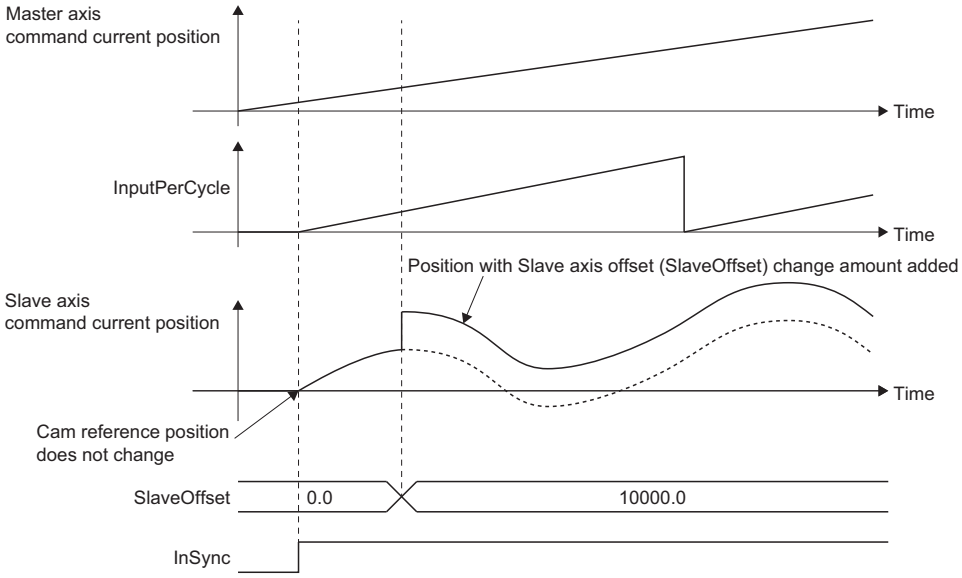
When a value other than "0.0" is set before the rising edge of In synchronization (InSync), the Reference value (Reference) is compensated so that the Slave axis (Slave) does not operate at the rising edge of In synchronization (InSync).

• When Start Mode (StartMode) is "0: Immediate (mcImmediate)"



When the slave axis offset (SlaveOffset) is changed while In Synchronization (InSync) is TRUE, the position of the Slave axis (Slave) operates matching the position with the offset amount added.

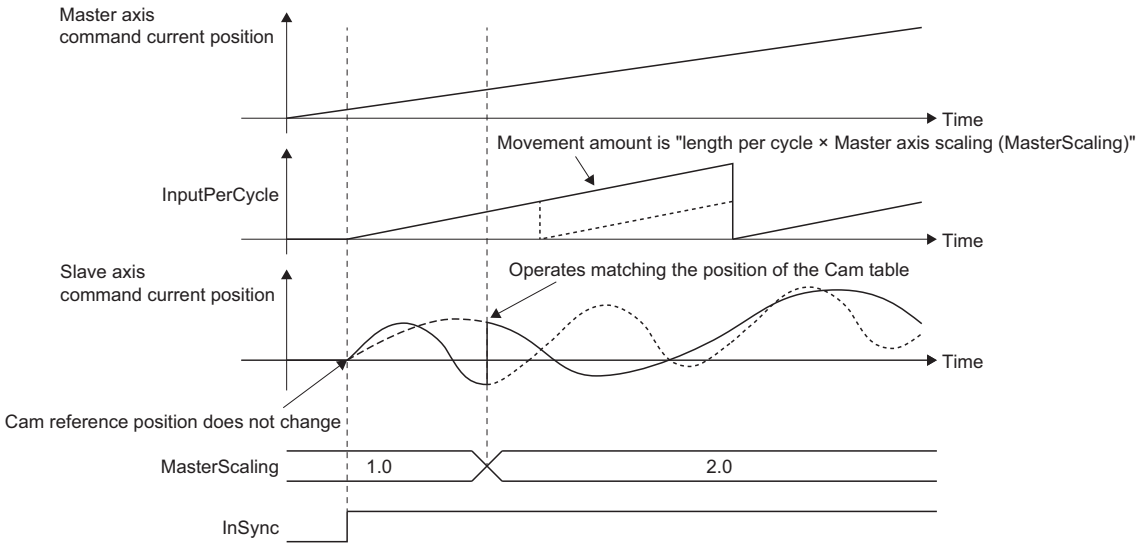
• When Start Mode (StartMode) is "0: Immediate (mcImmediate)"



■Master axis scaling (MasterScaling)

Master axis scaling (MasterScaling) expands/reduces the cam table length per cycle.

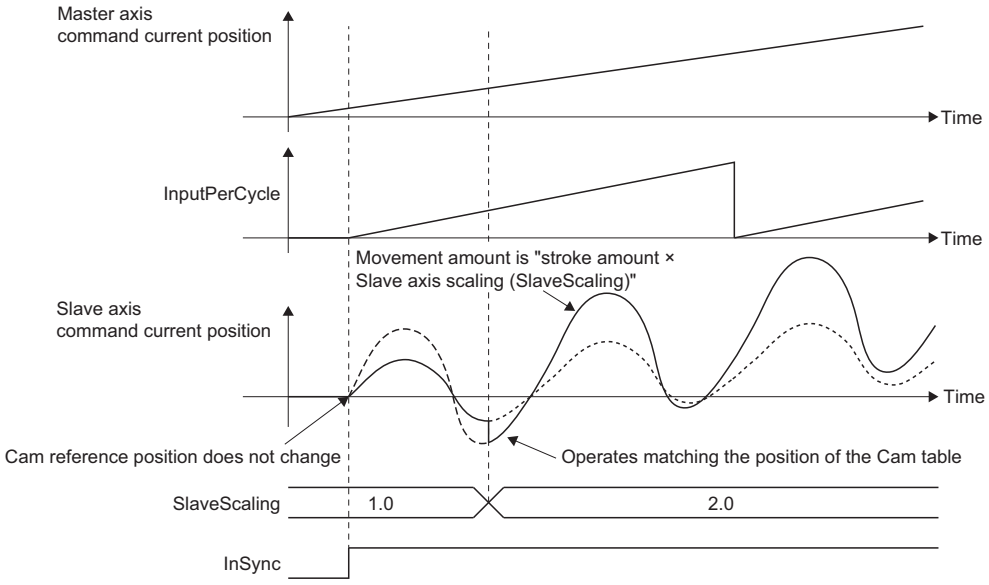
When changing during synchronization, a command equal to the movement amount is output to adjust the position of Slave axis (Slave) to the position of the cam table.



■Slave axis scaling (SlaveScaling)

Slave axis scaling (SlaveScaling) expands/reduces the stroke amount of the cam table.

When changing during synchronization, a command equal to the movement amount is output to adjust the position of Slave axis (Slave) to the position of the cam table.

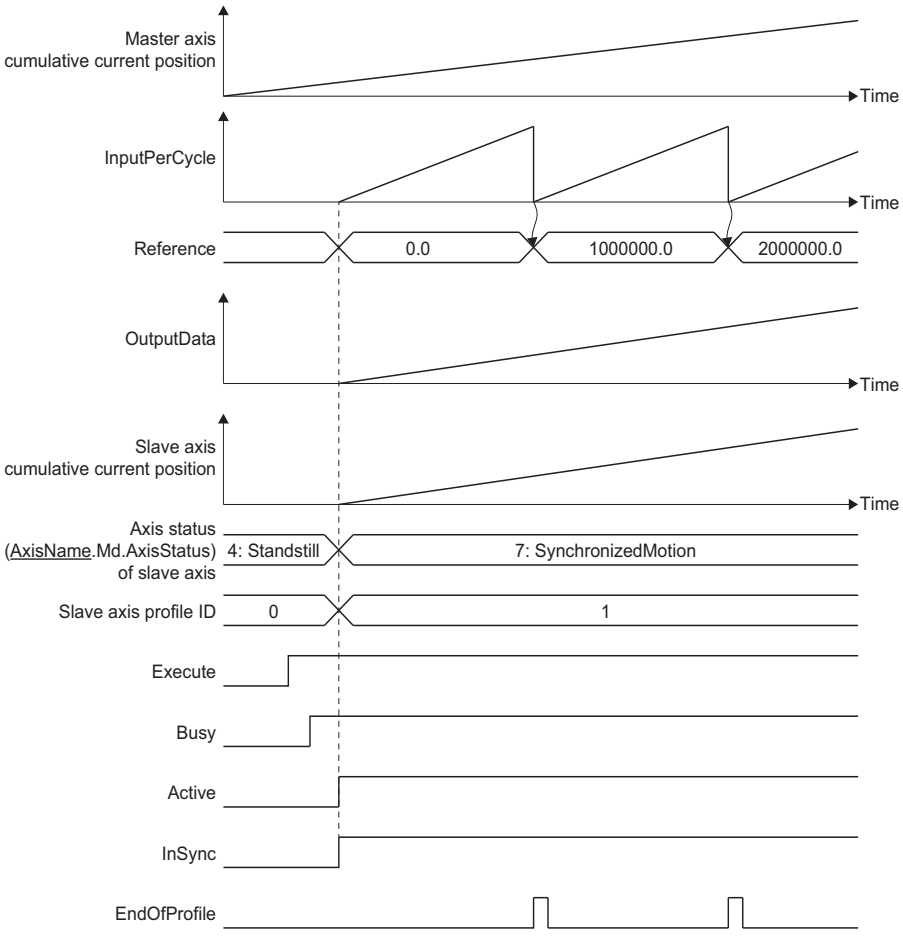


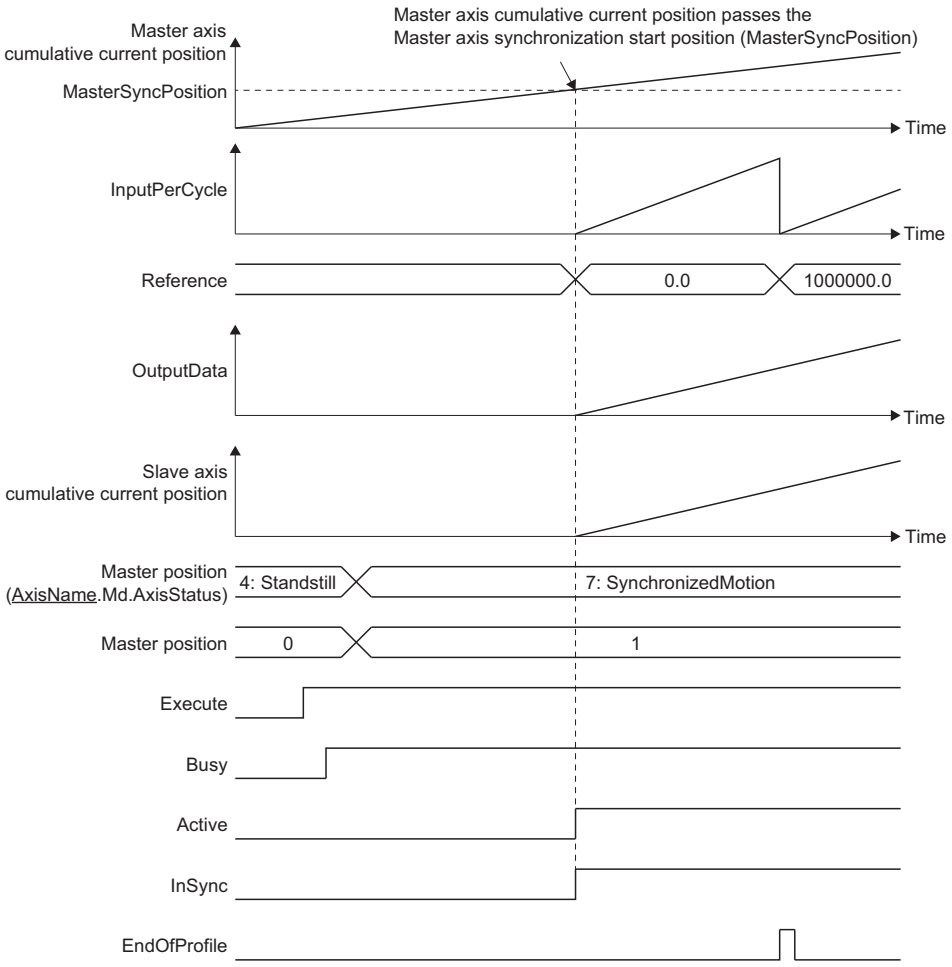
■Start mode (StartMode)

Sets the synchronization timing for cam operation and reflection timing for the cam control data.

When a value outside the range is set, "Out of Start Mode Range (error code: 1A47H)" occurs.

After Cam Operation Start (MC_CamIn) is executed, Controlling (Active) becomes TRUE if Current value per cycle (InputPerCycle) is synchronized, and In synchronization (InSync) becomes TRUE if Output value (OutputData) is synchronized.

Setting value	Description
0: Immediate (mcImmediate)	Synchronization is executed when MC_CamIn (Cam Operation Start) is started. Master axis synchronization start position (MasterSyncPosition) and Master axis follow-up distance (MasterStartDistance) are not used. The slave axis moves after Execute command (Execute) is started. 

Setting value	Description
1: Absolute (mcAbsolute)	Synchronization is executed when the position of the master axis passes Master axis synchronization start position (MasterSyncPosition). Master axis follow-up distance (MasterStartDistance) is not used. The slave axis moves after Execute command (Execute) is started and the master axis cumulative current position passes Master axis synchronization start position (MasterSyncPosition). Checking of whether the Master axis cumulative current position has passed the Master axis synchronization start position (MasterSyncPosition) begins at the point when the Slave axis status becomes "7: During synchronous operation (SynchronizedMotion)". 

■Master axis data source selection (MasterValueSource)

Slave axis (Slave) sets the position type of Master axis (Master) that will execute single axis synchronous control.
The operation may differ depending on the execution order of the FBs or if the operation cycles of Master axis (Master) and Slave axis (Slave) are different.
Operates with the same value as the set value when specifying Actual value in Master axis data source selection (MasterValueSource) while an axis type other than real axis is set in Master axis (Master).

Setting value	Description
1: Set value (mcSetValue)	Uses the commanded position of the master axis in the previous operation cycle.
2: Actual value (mcActualValue)	Uses the actual position of the master axis in the previous operation cycle.
101: Latest set value (mcLatestSetValue)	Uses the commanded position of the master axis in the current operation cycle.
102: Latest actual value (mcLatestActualValue)	Uses the actual position of the master axis in the current operation cycle.

Point

When Master axis data source selection (MasterValueSource) is set to "1: Set value (mcSetValue)" or "101: Latest set value (mcLatestSetValue)" and the master axis becomes servo OFF by a servo alarm or forced stop, the amount of the value change may be too large. This can be prevented by setting Master axis data source selection (MasterValueSource) to "2: Actual value (mcActualValue)" or "102: Latest actual value (mcLatestActualValue)".

■Buffer mode (BufferMode)

Sets the operation used to perform multiple start (buffer mode).

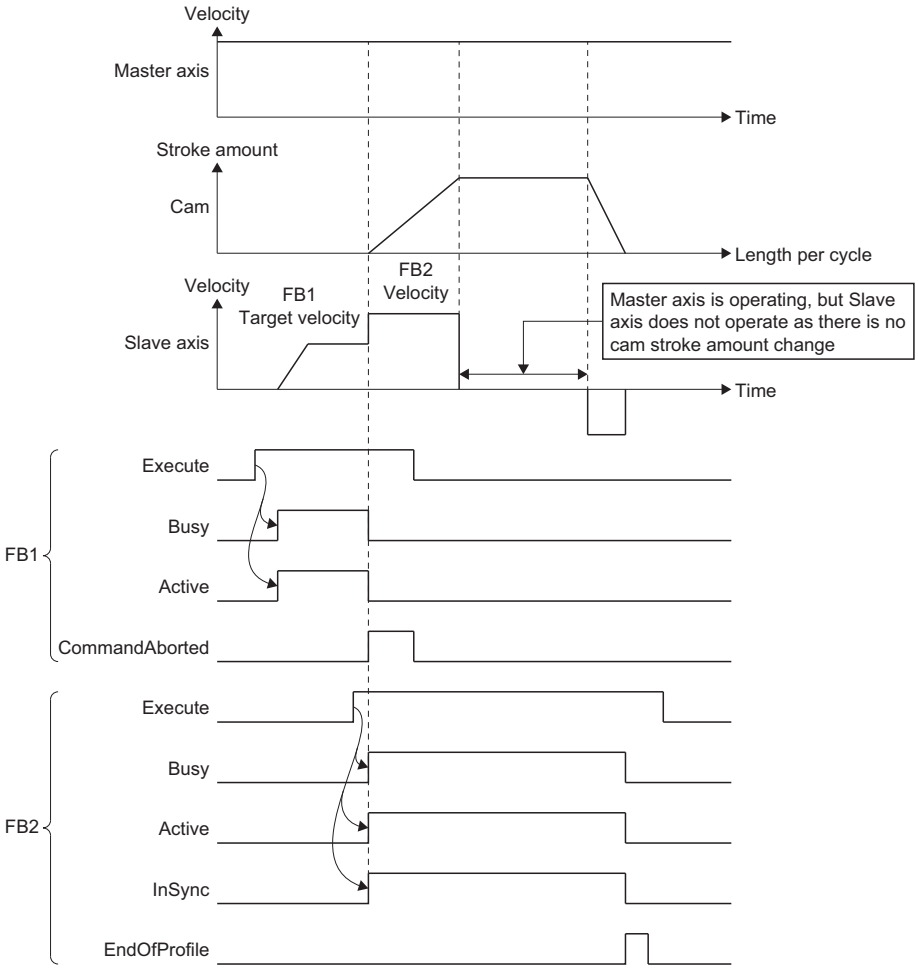
The following buffer modes are settable in MC_CamIn (Cam Operation Start).

The velocity for MC_CamIn (Cam Operation Start) follows the synchronizing FB (the master axis) and this velocity changes immediately based on the master axis velocity and the set cam data when the FB is switched. The switching condition for each mode is shown below.

Setting value	Description	Switching condition
0: Aborting (mcAborting)	Aborts (cancels) the under-control FB and executes the next FB immediately.	Always
1: Buffered (mcBuffered)	Buffers the next FB on the under-control FB. If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after completion of the under-control FB.	When Cam cycle completion (EndOfProfile) is TRUE

Ex.

When linked start FB2 (Slave axis (Slave): MC_CamIn (Cam Operation Start)) has Start mode (StartMode) set to "0: Immediately (mclmmediate)" and Buffer mode (BufferMode) is set to "0: Aborting (mcAborting)" while FB1 is being executed



Velocity of FB2 occurs after In synchronization (InSync) becomes TRUE.

When Start mode (StartMode) is set to a value other than "0: Immediate (mcImmediate)", In synchronization (InSync) does not become TRUE immediately after the FB starts so the velocity remains at 0 until Executing (Busy), Controlling (Active), or In synchronization (InSync) becomes TRUE.

Point

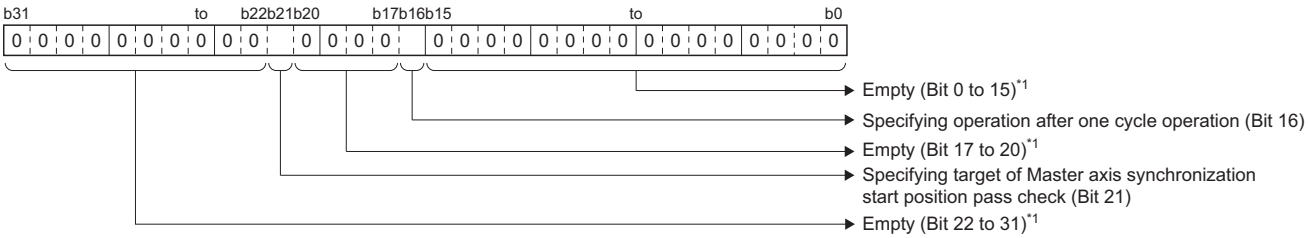
For details of multiple start (buffer mode), refer to the following.

MELSEC iQ-R Motion Module User's Manual (Application)

■Options (Options)

Function options for MC_CamIn (Cam Operation Start) are set by specifying the bit.

Contents that are set by specifying the bit are shown below.

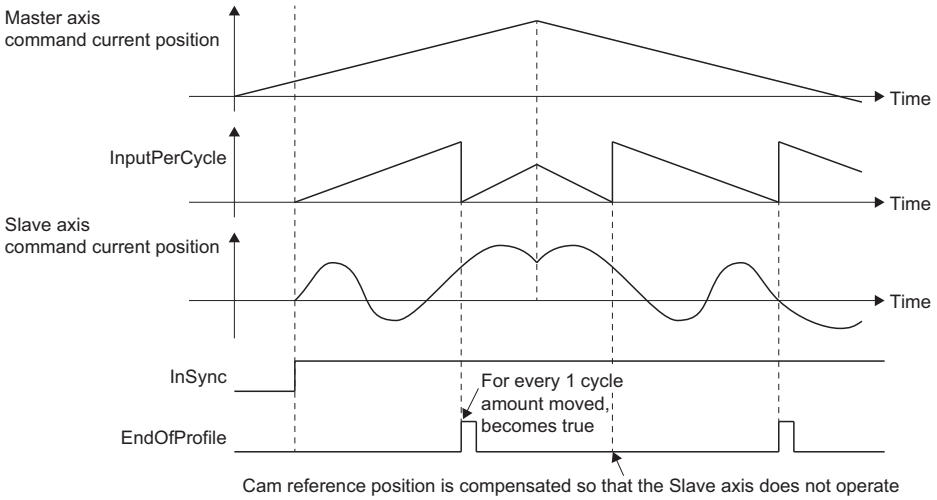


*1 Set empty areas to "0". When a value other than "0" is set, "Out of Options Range (error code: 1A4EH)" occurs.

Bit	Name	Description
16	Specifying operation after one cycle operation	Specifies the operation after one cycle operation when Periodic (Periodic) of MC_CamTableSelect (Cam Operation Start) is set to "FALSE: Periodic". • 0: Complete • 1: Wait for restart
21	Specifying target of Master axis synchronization start position pass check	Sets the target of the Master axis synchronization start position (MasterSyncPosition) pass check. • 0: Master axis cumulative current position, or where the feedback position has been converted to the cumulative position. • 1: Master axis command current position, or the feedback position.

■Cam cycle completion (EndOfProfile)

After controlling (Active) becomes TRUE, this variable becomes TRUE only for 1 execution cycle of the POU (Program Component) that calls FB each time movement equal to the cycle length occurs.



■Current value per cycle (InputPerCycle)

"0.0" is used as the initial value.

After Controlling (Active) becomes TRUE, the movement amount of Master axis (Master) is reflected as shown below.

Current value per cycle = (Master axis cumulative movement amount) + Master axis offset + Current value per cycle change value) MOD Current value per cycle

The monitor value range is the following cam length per cycle range set in the cam table.

0 ≤ Current value per cycle (InputPerCycle) of MC_CamIn (Cam Operation Start) < (Cam length per cycle × Master axis scaling)

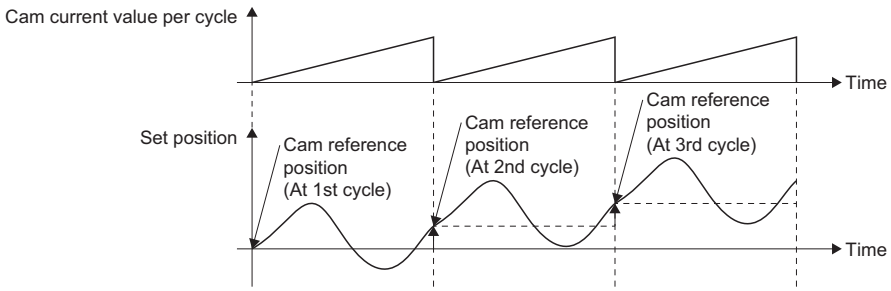
■Reference value (Reference)

When MC_CamIn (Cam Operation Start) is executed, the reference value is calculated from the cam position start (cam current value per cycle) based on the set position where In synchronization (InSync) becomes TRUE.

• For subtracting the initial stroke amount in the cam table, the formula for initial values are shown below.

Slave absolute coordinate (Slave Absolute)	ProfileControl (Calculation profile control)	Formula
FALSE (Relative coordinate)	Version "1.15" or later	Reference value = Slave axis position when synchronization starts - (Stroke value based on cam position (current value per cycle) when In synchronization (InSync) changes to TRUE + Initial stroke amount) × Slave axis scaling
	Version earlier than "1.15"	Reference value = 0 - (Stroke value based on cam position (current value per cycle) when In synchronization (InSync) changes to TRUE + Initial stroke amount) × Slave axis scaling
TRUE (Absolute coordinate)	—	Reference value = The cumulative current position corresponding to the command current position being 0 when In synchronization (In Sync) changes to TRUE.

• The reference values used in two-way cam operation and feed cam operation are shown below.

Operation	Description
When Two-way cam operation, or Slave absolute coordinate (SlaveAbsolute) is "TRUE (Absolute Coordinate)"	The reference value is not updated.
When Feed cam operation, as well as Slave absolute coordinate are "FALSE (relative coordinate)"	The reference value is calculated as shown below. • Reference value = (Original reference value + (Stroke value at the last point - Stroke value at the 1st point) × Slave axis scaling) 

• The reference values are updated with the following timing.

Update timing	Reference value calculation formula
When Current value per cycle (InputPerCycle) going in the address increasing direction passes the last point of the cam table	Original reference value + ((Stroke value at the last point - Stroke value at the 1st point) × Slave axis scaling)
When Current value per cycle (InputPerCycle) going in the address decreasing direction passes the 1st point of the cam table	Original reference value - ((Stroke value at the last point - Stroke value at the 1st point) × Slave axis scaling)
When the current value per cycle is changed (Completion (Done) is TRUE for MCv_ChangeCycle (Current Value Change per Cycle))	Original reference value - (Stroke value corresponding to current value per cycle after change × Slave axis scaling)

• The monitor value range is as follows depending on the ProfileControl (Calculation profile control).

ProfileControl (Calculation profile control)	Monitor value range
Version "1.15" or later	Lower limit value of the slave axis ring counter ≤ Reference value < Upper limit value of the slave axis ring counter
Version earlier than "1.15"	Lower limit value of the positioning range ≤ Reference value < Upper limit value of the positioning range

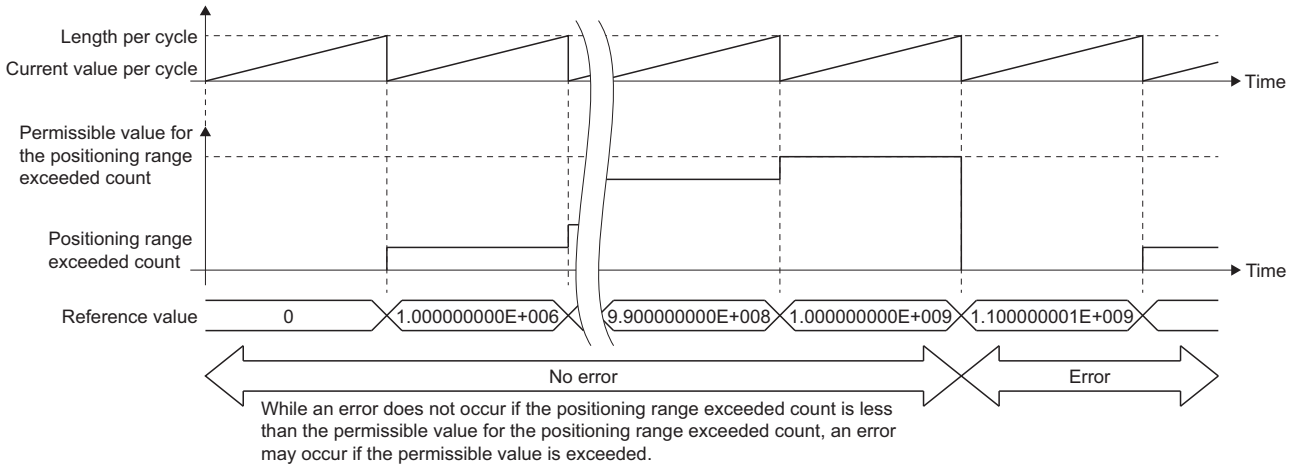
Restriction

<Reference value errors>

Performing any of the following operations may cause a reference value error to occur.

- If the number of TRUE outputs by Cam cycle completion (EndOfProfile) exceeds the permissible value after the FeedCam as well as In synchronization (InSync) become TRUE[*1].
- If the FeedCam as well as the setting for the following control parameters contains a decimal.
  - (Stroke value at the last point - Stroke value at the 1st point)
  - Slave axis scaling
- If current value per cycle change is repeated with relative selection.

*1: While an error does not occur if the positioning range exceeded count is less than the permissible value for the positioning range exceeded count, an error may occur if the permissible value is exceeded. (Operation is continued regardless of whether or not a truncation error has occurred.)



If the reference value error becomes too large, clear the synchronization status or reset it by executing a current value change with absolute position specification.

The permissible value for the positioning range exceeded count is a variable value calculated with the following formula.

Permissible value for the positioning range exceeded count = Limit value for the positioning range exceeded count / ((Output value at the last point - Output value at the 1st point) /slave scaling)

*: The limit value for the positioning range exceeded count is the upper limit value of the positioning range, "10000000000".

(Example)

When the output value at the 1st point is "0", the output value at the last point is "1000000", and the output coefficient is "10", the permissible value for the positioning range exceeded count becomes "100000".

■Output value (OutputData)

The default value is the initial stroke amount in the cam table.

When In synchronization (InSync) is TRUE, the output value is calculated as shown below.

- Output value = Cam reference position + ((Cam stroke amount × Stroke ratio corresponding to cam current value per cycle) x Slave axis scaling) + Slave axis offset

Alternatively,

- Output value = Cam reference position + ((Output value corresponding to cam current value per cycle) x Slave axis scaling) + Slave axis offset

The monitor value range is as follows depending on the ProfileControl (Calculation profile control).

ProfileControl (Calculation profile control)	Monitor value range
Version "1.15" or later	Lower limit value of the slave axis ring counter ≤ Output value < Upper limit value of the slave axis ring counter
Version earlier than "1.15"	Lower limit value of the positioning range ≤ Output value < Upper limit value of the positioning range

Precautions

- When Master axis offset (MasterOffset), Slave axis offset (SlaveOffset), Master axis scaling (MasterScaling), Slave axis scaling (SlaveScaling), or Cam table ID (CamTableID) are changed, the slave axis may move suddenly at the start of control or a control change. This sudden movement may cause a shock to the equipment, so it is important to carefully check the setting values and the change timing.
- When monitoring the axis being cam controlled with the engineering tool, the position, velocity, etc. is based on the unit setting of the axis. Units for the cycle length, stroke amount, etc. set in the calculation profile are not used.

Program example

In the following program example, the cam operation start command (bCamInStart) is set to TRUE, Axis2 (Axis0002) is set as the master axis to be started in JOG operation, and cam operation is performed for the slave axis (Axis0001) according to the calculation profile (cam data 1) and using the following settings.

- Settings

Item		Setting value
Data setting for JOG operation	Velocity	1000.0
	Acceleration	5000.0
	Deceleration	5000.0
	Jerk	0.0

■Axes

Axis No.	Label name	Data type	Comment
1	Axis0001	AXIS_REF	Axis 1
2	Axis0002	AXIS_REF	Axis 2

■Calculation profile

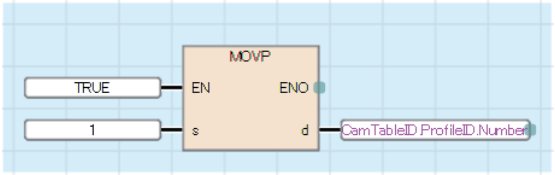
Profile ID	Label name	Data type	Comment
1	ProfileData0001	MC_CAM_REF	Cam data 1

■Labels used

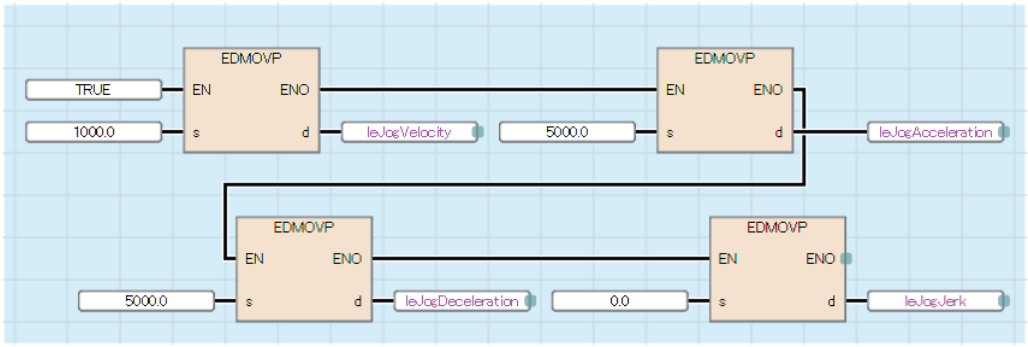
Label name	Data type	Comment
MC_CamIn_1	MC_CamIn	Cam Operation Start FB
bCamInStart	Bit	Cam operation start command
CamTableID	MC_CAM_ID	Cam table ID
bInSync	Bit	In synchronization
bCamBusy	Bit	Executing
bCamActive	Bit	Controlling
bCamCommandAborted	Bit	Abortion of execution
bCamError	Bit	Error
uwCamErrorID	Word [unsigned]/bit string [16-bit]	Error code
bEndOfProfile	Bit	Cam cycle completion
MCv_Jog_1	MCv_Jog	JOG FB
leJogVelocity	Double-precision real number	JOG velocity
leJogAcceleration	Double-precision real number	JOG acceleration
leJogDeceleration	Double-precision real number	JOG deceleration
leJogJerk	Double-precision real number	JOG jerk
bJogDone	Bit	Execution completion
bJogBusy	Bit	Executing
bJogActive	Bit	Controlling
bJogCommandAborted	Bit	Abortion of execution
bJogError	Bit	Error
uwJogErrorID	Word [unsigned]/bit string [16-bit]	Error code

■FBD/LD program (CPU module side)

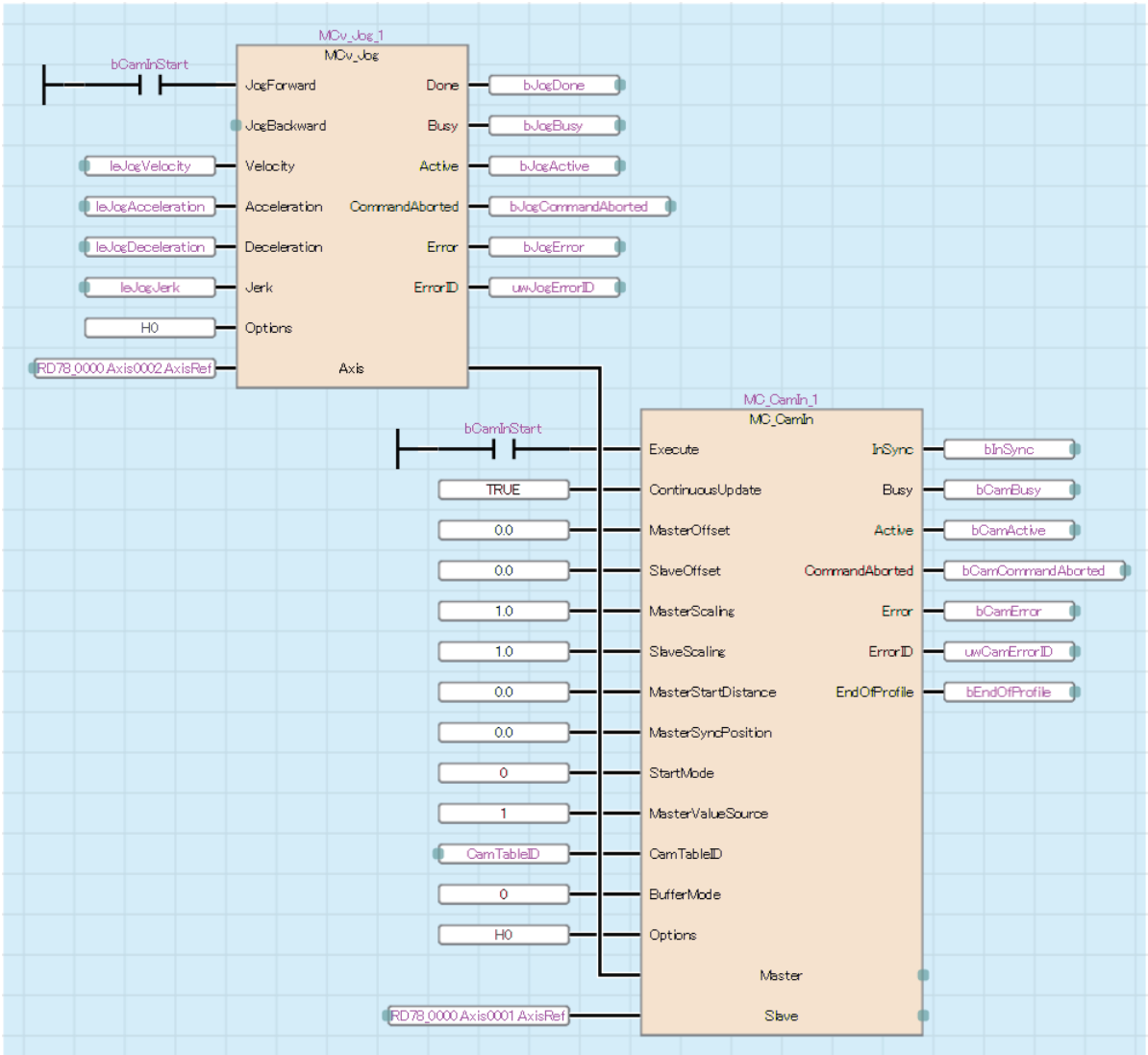
- Calculation profile (cam data) settings



- Data settings for JOG operation



- JOG operation/Cam operation start



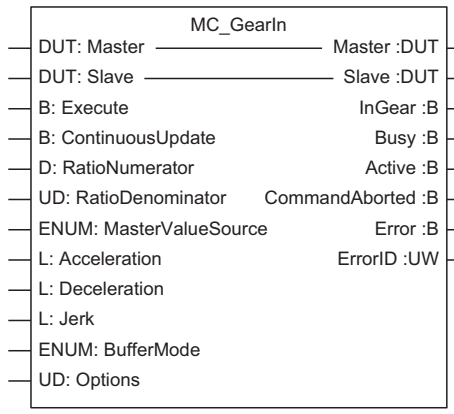
■ST program (Motion module side)

```
//-----Calculation profile (cam data) settings-----
CamTableID.ProfileID.Number:= 1;

//-----Data settings for JOG operation-----
leJogVelocity:= 1000.0;
leJogAcceleration:= 5000.0;
leJogDeceleration:= 5000.0;
leJogJerk:= 0.0;

//-----JOG-----
MCv_Jog_1(
   Axis:= Axis0002.AxisRef ,
   JogForward:= bCamInstart ,
   Velocity:= leJogVelocity ,
   Acceleration:= leJogAcceleration ,
   Deceleration:= leJogDeceleration ,
   Jerk:= leJogJerk ,
   Options:= H00000000 ,
   Done=> bJogDone ,
   Busy=> bJogBusy ,
   Active=> BJogActive ,
   CommandAborted=> bJogCommandAborted ,
   Error=> bJogError ,
   ErrorID=> uwJogErrorID
);

//-----Cam Operation Start-----
MC_CamIn_1(
   Master:= Axis0002.AxisRef ,
   Slave:= Axis0001.AxisRef ,
   Execute:= bCamInstart ,
   ContinuousUpdate:= TRUE ,
   MasterOffset:= 0.0 ,
   SlaveOffset:= 0.0 ,
   MasterScaling:= 1.0 ,
   SlaveScaling:= 1.0 ,
   MasterStartDistance:= 0.0 ,
   MasterSyncPosition:= 0.0 ,
   StartMode:= MC_START_MODE__mcImmediate ,
   MasterValueSource:= MC_SOURCE__mcSetValue ,
   CamTableID:= CamTableID ,
   BufferMode:= MC_BUFFER_MODE__mcAborting ,
   Options:= H00000000 ,
   InSync=> bInSync ,
   Busy=> bCamBusy ,
   Active=> bCamActive ,
   CommandAborted=> bCamCommandAborted ,
   Error=> bCamError ,
   ErrorID=> uwCamErrorID ,
   EndOfProfile=> bEndOfProfile
);
```

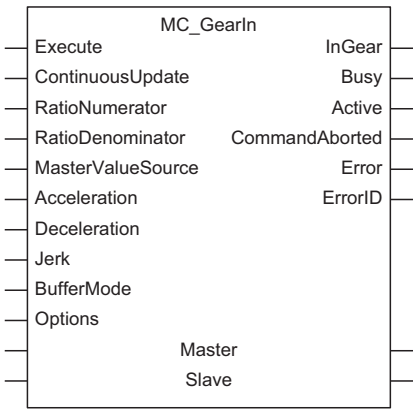
MC_GearIn

Starts gear operation according to the specified gear ratio.

Ladder



FBD/LD



ST

```
MC_GearIn(
  Master:= ?AXIS_REF? ,
  Slave:= ?AXIS_REF? ,
  Execute:= ?BOOL? ,
  ContinuousUpdate:= ?BOOL? ,
  RatioNumerator:= ?DINT? ,
  RatioDenominator:= ?DWORD? ,
  MasterValueSource:= ?INT? ,
  Acceleration:= ?LREAL? ,
  Deceleration:= ?LREAL? ,
  Jerk:= ?LREAL? ,
  BufferMode:= ?INT? ,
  Options:= ?DWORD? ,
  InGear=> ?BOOL? ,
  Busy=> ?BOOL? ,
  Active=> ?BOOL? ,
  CommandAborted=> ?BOOL? ,
  Error=> ?BOOL? ,
  ErrorID=> ?WORD?
);
```

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Gear Operation Start	64	8	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
Master	Master axis	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)
Slave	Slave axis	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Execute command	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MC_GearIn (Gear Operation Start).
ContinuousUpdate	Continuous update	BOOL	At start	TRUE, FALSE	FALSE	This variable sets whether to enable or disable continuous change of Gear ratio numerator (Ratio Numerator), Gear ratio denominator (RatioDenominator), Master axis data source selection (MasterValueSource), Acceleration (Acceleration), and Deceleration (Deceleration). • FALSE: Disable • TRUE: Enable
RatioNumerator	Gear ratio numerator	DINT	At start/ Retrigger possible/ Continuous update possible	-2147483648 to 2147483647	1	This variable sets the numerator value of Master axis (Master). For details, refer to the following. ☞ Page 352 Gear ratio numerator (RatioNumerator)/Gear ratio denominator (RatioDenominator)

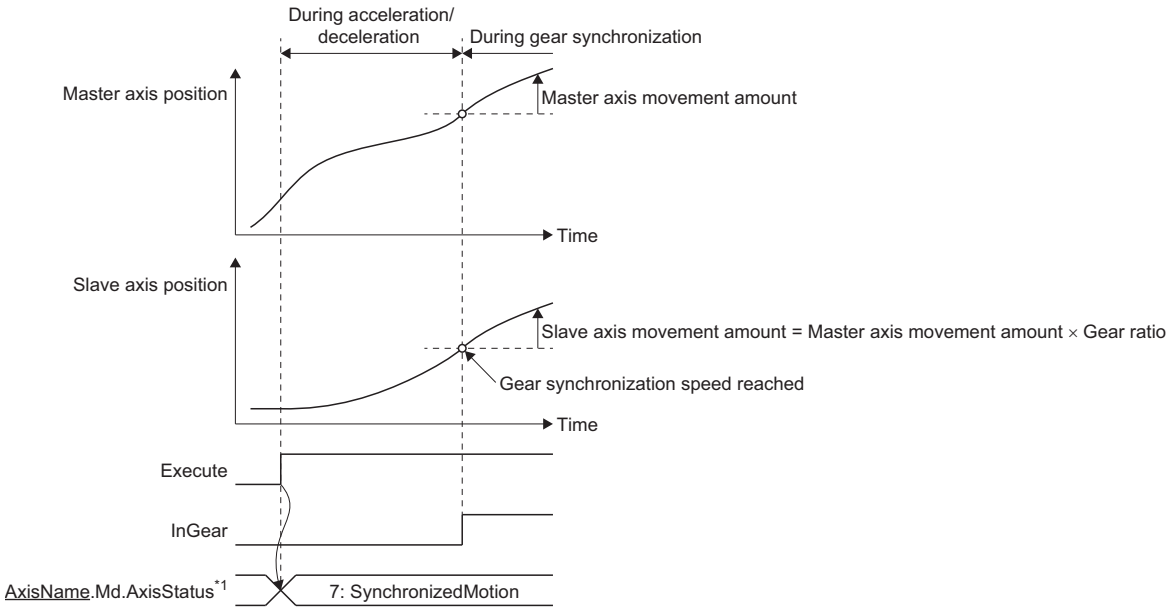
Input variable	Name	Data type	Input	Setting range	Default value	Description
RatioDenominator	Gear ratio denominator	DWORD(UDINT)	At start/ Retrigger possible/ Continuous update possible	1 to 2147483647	1	This variable sets the denominator value of Master axis (Master). For details, refer to the following. ☞ Page 352 Gear ratio numerator (RatioNumerator)/Gear ratio denominator (RatioDenominator)
MasterValueSource	Master axis data source selection	INT (MC_SOURCE)	At start	1, 2, 101, 102	1	This variable sets the data source of Master axis (Master). • 1: Set value (mcSetValue) • 2: Actual value (mcActualValue) • 101: Latest set value (mcLatestSetValue) • 102: Latest actual value (mcLatestActualValue) For details, refer to the following. ☞ Page 353 Master axis data source selection (MasterValueSource)
Acceleration	Acceleration	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the acceleration used for accelerating. When the target velocity is reached, Gear ratio reached (InGear) becomes TRUE and Slave axis (Slave) is controlled using a velocity equal to the velocity of Master axis (Master) converted by the gear ratio.
Deceleration	Deceleration	LREAL	At start/ Retrigger possible/ Continuous update possible	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the deceleration used for decelerating. When the target velocity is reached, Gear ratio reached (InGear) becomes TRUE and Slave axis (Slave) is controlled using a velocity equal to the velocity of Master axis (Master) converted by the gear ratio.
Jerk	Jerk	LREAL	At start	0.0000, 0.0001 to 2147483647.0	0.0	This variable sets the jerk at acceleration/ deceleration start.
BufferMode	Buffer mode	INT (MC_BUFFER_MODE)	At start	0, 1	0	This variable sets the buffer mode • 0: Aborting (mcAborting) • 1: Buffered (mcBuffered) For details, refer to the following. ☞ Page 353 Buffer mode (BufferMode)
Options	Options	DWORD(HEX)	At start	00000000H	00000000H	Set this variable to "00000000H". *: When a value other than "00000000H" is set, "Out of Options Range (error code: 1A4EH)" occurs.

■Output variables

Output variable	Name	Data type	Default value	Description
InGear	Gear ratio reached	BOOL	FALSE	This variable becomes TRUE when the gear synchronization speed is reached.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MC_GearIn (Gear Operation Start) is executed.
Active	Controlling	BOOL	FALSE	This variable becomes TRUE when Slave axis (Slave) is being controlled.
CommandAborted	Abortion of execution	BOOL	FALSE	This variable becomes TRUE when execution of MC_GearIn (Gear Operation Start) is aborted. This variable becomes FALSE when Execute command (Execute) becomes FALSE.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. 📖MELSEC iQ-R Motion Module User's Manual (Application)

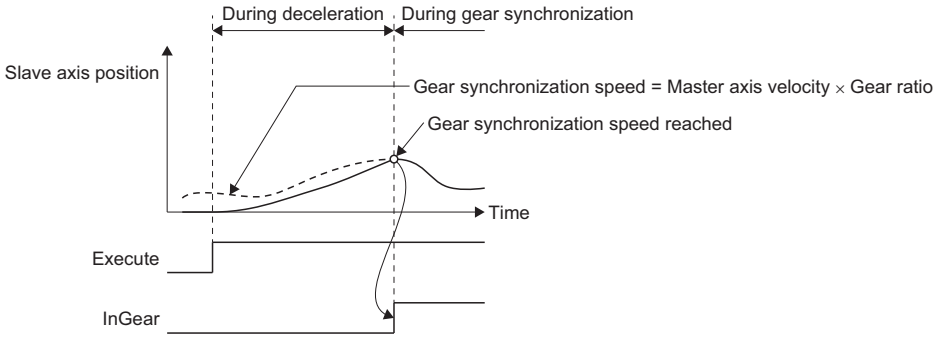
Processing details

- MC_GearIn (Gear Operation Start) sets Gear ratio numerator (RatioNumerator), Gear ratio denominator (RatioDenominator), Master axis data source selection (MasterValueSource), Acceleration (Acceleration), Deceleration (Deceleration), Jerk (Jerk), and Buffer mode (BufferMode), then starts the gear operation.
- To stop the operation, perform MC_Stop (Forced Stop).
- After executing this FB, Slave axis (Slave) performs acceleration/deceleration until reaching the target velocity, the value of which is the velocity of Master axis (Master) converted by the gear ratio. After reaching the target velocity, Gear ratio reached (InGear) becomes TRUE, and Slave axis (Slave) is controlled using a velocity equal to the velocity of Master axis (Master) converted by the gear ratio.



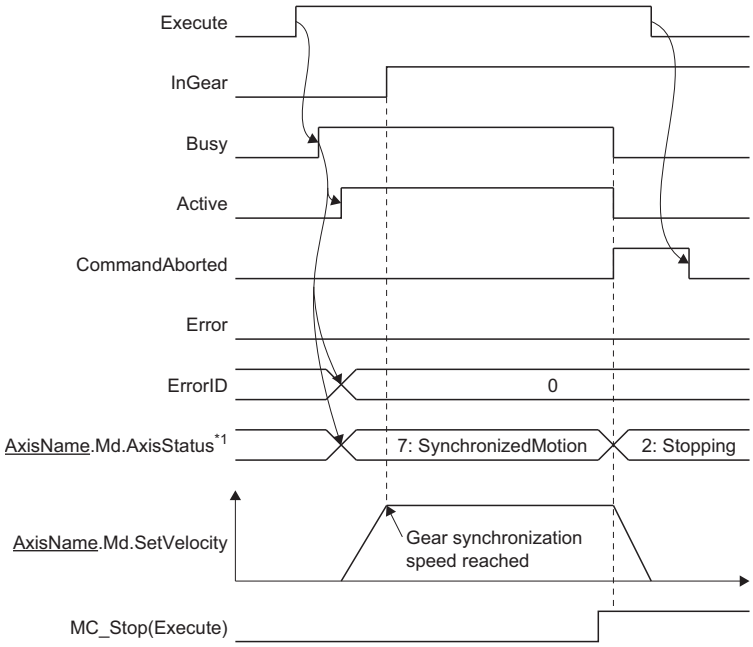
*1 The axis status of Slave axis (Slave).

- When the velocity of Master axis (Master) is changed during acceleration/deceleration, the gear synchronization speed is also updated.



■Timing chart

• When the operation completes normally



*1 The axis status of Slave axis (Slave).

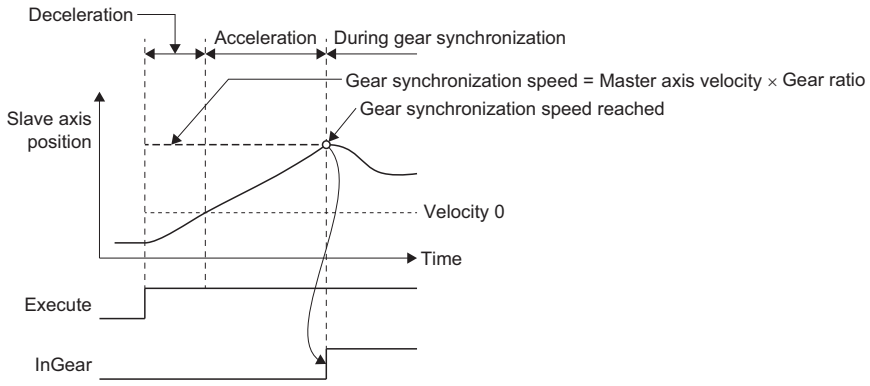
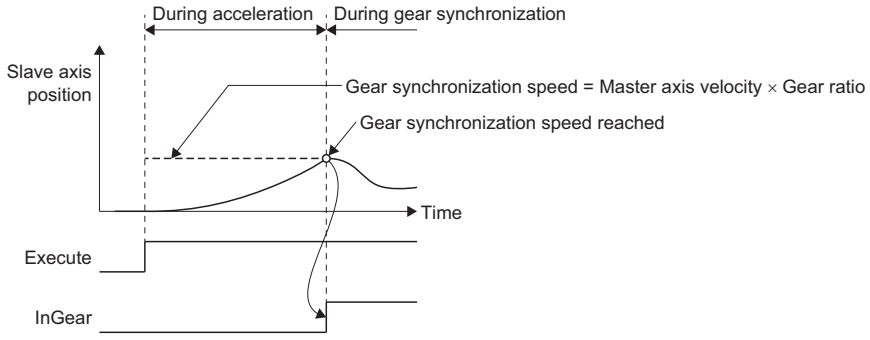
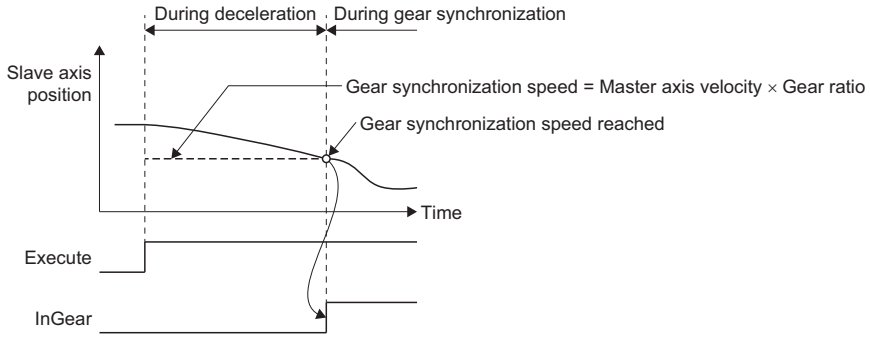
• When an error occurs

For the timing chart when an error occurs, refer to the following.

☞ Page 11 Basic operation of Execute command (Execute) type Motion control FBs

■Acceleration/deceleration until reaching the gear synchronization speed

- The acceleration/deceleration method for MC_GearIn (Gear Operation Start) is the acceleration/deceleration specification method.
- Depending on the velocity and target velocity of the slave axis (Slave) at operation start, one of the following 4 types of acceleration/deceleration is performed. The type of acceleration/deceleration will not change even if the target velocity is updated during the acceleration/deceleration.

Slave axis (Slave) velocity and target velocity	Acceleration/deceleration
Slave axis (Slave) velocity < 0 < Gear synchronization speed Alternatively, Slave axis (Slave) velocity > 0 > Gear synchronization speed	Deceleration is performed to velocity 0, then acceleration is performed from velocity 0 to the gear synchronization speed. 
Slave axis (Slave) velocity (Absolute value) < Gear synchronization speed (Absolute value)	Acceleration is performed until reaching the gear synchronization speed. 
Slave axis (Slave) velocity (Absolute value) > Gear synchronization speed (Absolute value)	Acceleration is performed until reaching the gear synchronization speed. 
Slave axis (Slave) velocity = Gear synchronization speed	Acceleration/deceleration is not performed. After operation is started, Gear ratio reached (InGear) becomes TRUE immediately and Slave axis (Slave) is controlled using a velocity equal to the velocity of Master axis (Master) converted by the gear ratio.

■Gear ratio numerator (RatioNumerator)/Gear ratio denominator (RatioDenominator)

Gear ratio numerator (RatioNumerator) and Gear ratio denominator (RatioDenominator) are set together.

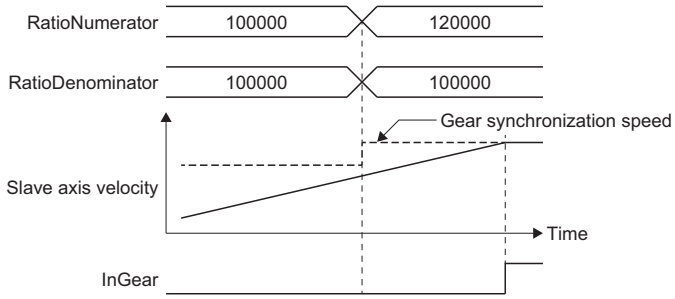
Gear ratio numerator (RatioNumerator) sets the numerator value for converting and transmitting the velocity of Master axis (Master).

Gear ratio denominator (RatioDenominator) sets the denominator value for converting and composing the movement amount of Master axis (Master).

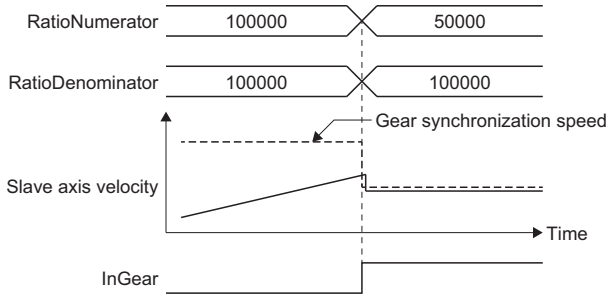
The velocity is converted as shown below.

$$\text{Master axis (Master) velocity after conversion} = \text{Master axis (Master) velocity before conversion} \times \frac{\text{Gear ratio numerator (RatioNumerator)}}{\text{Gear ratio denominator (RatioDenominator)}}$$

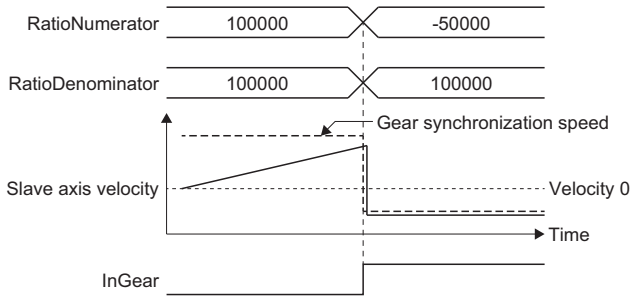
- Set a positive value in Gear ratio denominator (RatioDenominator).
- The velocity direction can be reversed and transmitted by setting a negative value in Ratio numerator (RatioNumerator).
  The gear synchronization speed is changed if the gear ratio is changed during acceleration/deceleration. Therefore, the velocity of Slave axis (Slave) is changed directly and the gear operation may start because of the change of the gear synchronization speed.
- The velocity of Master axis (Master) after conversion can be set to "0" by setting the value of Gear ratio numerator (RatioNumerator) to "0".
- The target velocity is changed if the gear ratio is changed during acceleration/deceleration. Therefore, changing the target velocity may directly change the velocity of Slave axis (Slave) and start the gear operation.
  - Changing the gear ratio during acceleration/deceleration
    - Gear ratio increase (set to a value higher than the velocity of Slave axis (Slave))



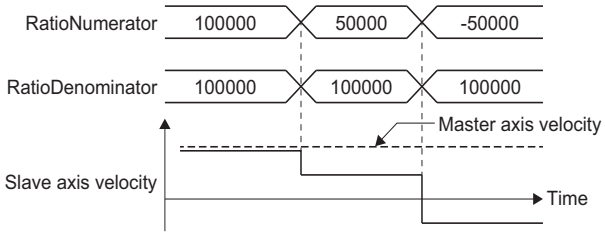
    - Gear ratio decrease (set to a value higher than the velocity of Slave axis (Slave))



    - Gear ratio direction reverse (set the gear synchronization speed to a value lower than "0")



- When the gear ratio is changed during gear synchronization, the velocity of Slave axis (Slave) is changed directly.
  - Gear ratio change during gear synchronization



■Master axis data source selection (MasterValueSource)

Slave axis (Slave) sets the position type of Master axis (Master) that will execute single axis synchronous control.
The operation may differ depending on the execution order of the FBs or if the operation cycles of Master axis (Master) and Slave axis (Slave) are different.
Operates with the same value as the set value when specifying Actual value in Master axis data source selection (MasterValueSource) while an axis type other than real axis is set in Master axis (Master).

Setting value	Description
1: Set value (mcSetValue)	Uses the commanded position of the master axis in the previous operation cycle.
2: Actual value (mcActualValue)	Uses the actual position of the master axis in the previous operation cycle.
101: Latest set value (mcLatestSetValue)	Uses the commanded position of the master axis in the current operation cycle.
102: Latest actual value (mcLatestActualValue)	Uses the actual position of the master axis in the current operation cycle.

Point

When Master axis data source selection (MasterValueSource) is set to "1: Set value (mcSetValue)" or "101: Latest set value (mcLatestSetValue)" and the master axis becomes servo OFF by a servo alarm or forced stop, the amount of the value change may be too large. This can be prevented by setting Master axis data source selection (MasterValueSource) to "2: Actual value (mcActualValue)" or "102: Latest actual value (mcLatestActualValue)".

■Buffer mode (BufferMode)

Sets the operation used to perform multiple start (buffer mode).
The following buffer modes are settable in MC_GearIn (Gear Operation Start).

Setting value	Description
0: Aborting (mcAborting)	Aborts (cancels) the under-control FB and executes the next FB immediately.
1: Buffered (mcBuffered)	Buffers the next FB on the under-control FB. If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after completion of the under-control FB.

Point

For details of multiple start (buffer mode), refer to the following.
MELSEC iQ-R Motion Module User's Manual (Application)

Precautions

- When the gear ratio is changed, the velocity of the slave axis is changed directly. To smooth the speed change, use the buffer mode together with MCv_SmoothingFilter (Smoothing Filter).
- When the velocity of Master axis (Master) after conversion exceeds the acceleration/deceleration upper limit value during acceleration/deceleration operation, "Acceleration Time Limit Over Warning (warning code: 0D04H)" or "Deceleration Time Limit Over Warning (warning code: 0D05H)" will occur and stop the acceleration/deceleration operation. Following this, operation is continued at the velocity from the point that the warning was detected. If the warnings shown above occur, adjust the velocity of Master axis (Master) after conversion, Acceleration (Acceleration), and Deceleration (Deceleration) so that the setting for each does not exceed the acceleration/deceleration upper limit value in order to restart the acceleration/deceleration operation.

Program example

In the following program example, the gear operation start command (bGearInStart) is set to TRUE, Axis 1 (Axis0001) is set as the master axis to be started in JOG operation, and gear operation is performed for the slave axis (Axis0002) according to the specified gear ratio and using the following settings.

• Settings

Item		Setting value
Data settings for gear operation	Acceleration	5000.0
	Deceleration	5000.0
	Jerk	0.0
Data settings for JOG operation	Velocity	1000.0
	Acceleration	5000.0
	Deceleration	5000.0
	Jerk	0.0

■Axes

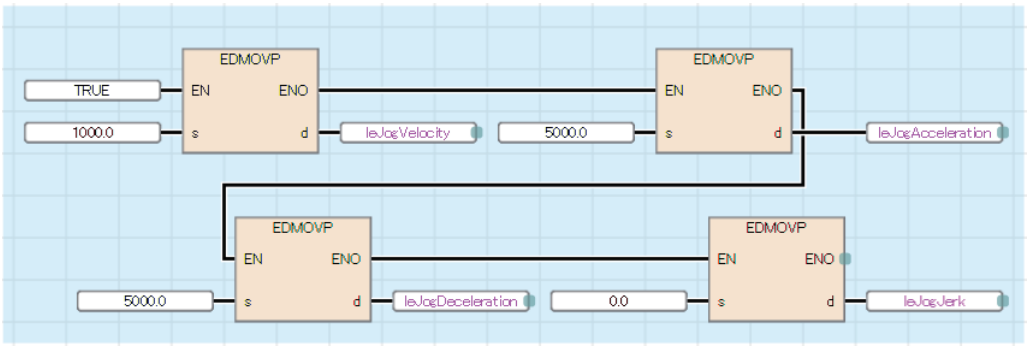
Axis No.	Label name	Data type	Comment
1	Axis0001	AXIS_REF	Axis 1
2	Axis0002	AXIS_REF	Axis 2

■Labels used

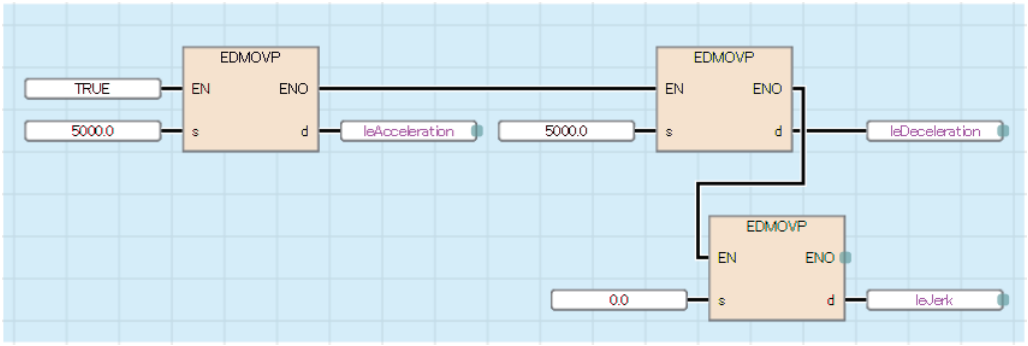
Label name	Data type	Comment
MC_GearIn_1	MC_GearIn	Gear Operation Start FB
bGearInStart	Bit	Gear operation start command
leAcceleration	Double-precision real number	Acceleration
leDeceleration	Double-precision real number	Deceleration
leJerk	Double-precision real number	Jerk
bInGear	Bit	In synchronization
bGearBusy	Bit	Executing
bGearActive	Bit	Controlling
bGearCommandAborted	Bit	Abortion of execution
bGearError	Bit	Error
uwGearErrorID	Word [unsigned]/bit string [16-bit]	Error code
MCv_Jog_1	MCv_Jog	JOG FB
leJogVelocity	Double-precision real number	JOG velocity
leJogAcceleration	Double-precision real number	JOG acceleration
leJogDeceleration	Double-precision real number	JOG deceleration
leJogJerk	Double-precision real number	JOG jerk
bJogDone	Bit	Execution completion
bJogActive	Bit	Controlling
bJogCommandAborted	Bit	Abortion of execution
bJogError	Bit	Error
uwJogErrorID	Word [unsigned]/bit string [16-bit]	Error code

■FBD/LD program (CPU module side)

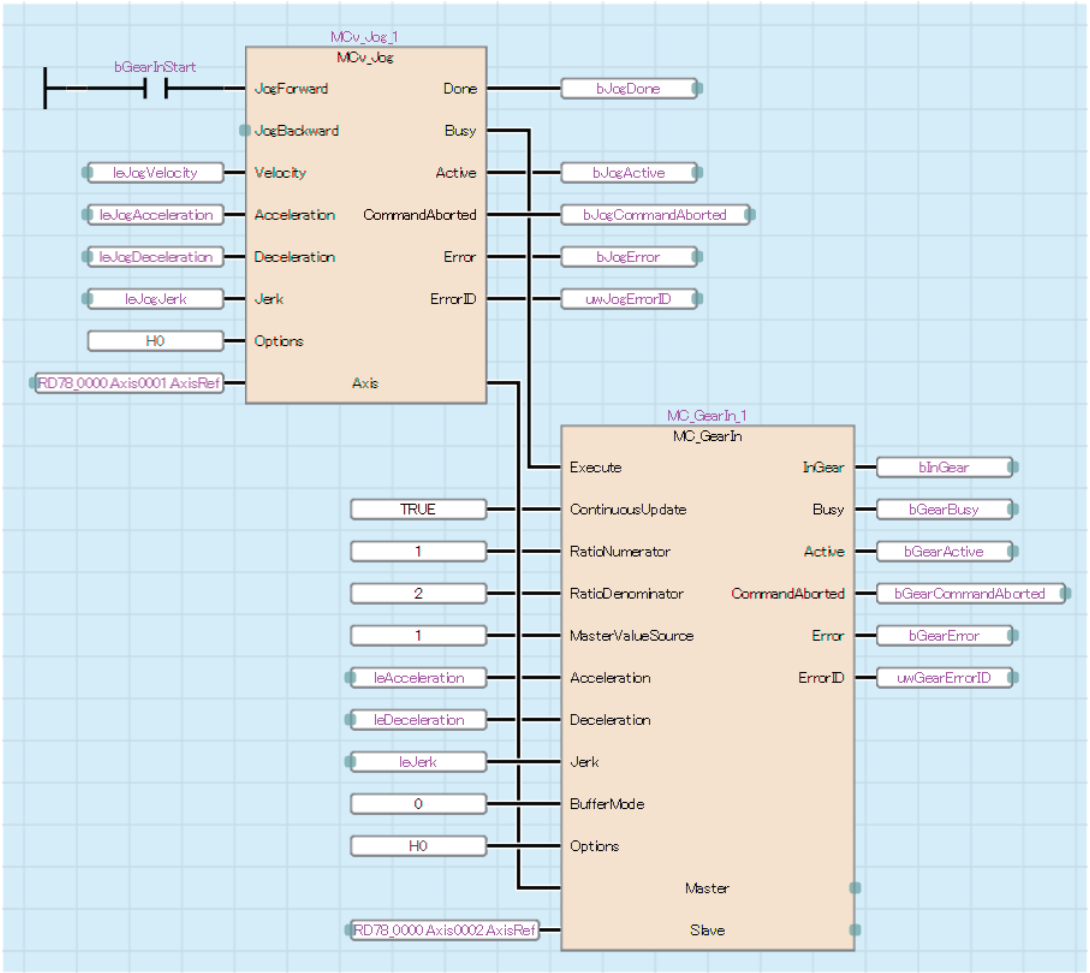
• Data settings for JOG operation



• Data settings for gear operation



• JOG/Gear Operation Start



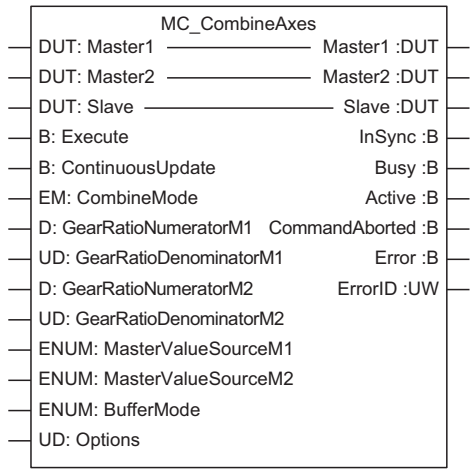
■ST program (Motion module side)

```
//-----Data settings for JOG operation-----
leJogVelocity:= 1000.0;
leJogAcceleration:= 5000.0;
leJogDeceleration:= 5000.0;
leJogJerk:= 0.0;

//-----Data settings for gear operation-----
leAcceleration:= 5000.0;
leDeceleration:= 5000.0;
leJerk:= 0.0;

//-----JOG-----
MCv_Jog_1(
   Axis:= Axis0001.AxisRef ,
   JogForward:= bGearInstart ,
   Velocity:= leJogVelocity ,
   Acceleration:= leJogAcceleration ,
   Deceleration:= leJogDeceleration ,
   Jerk:= leJogJerk ,
   Options:= H00000000 ,
   Done=> bJogDone ,
   Busy=> bJogBusy ,
   Active=> BJogActive ,
   CommandAborted=> bJogCommandAborted ,
   Error=> bJogError ,
   ErrorID=> uwJogErrorID
);

//-----Gear Operation Start-----
   MC_GearIn_1(
   Master:= Axis0001.AxisRef ,
   Slave:= Axis0002.AxisRef ,
   Execute:= bJogBusy ,
   ContinuousUpdate:= TRUE ,
   RatioNumerator:= 1 ,
   RatioDenominator:= 2 ,
   MasterValueSource:= MC_SOURCE__mcSetValue ,
   Acceleration:= leAcceleration ,
   Deceleration:= leDeceleration ,
   Jerk:= leJerk ,
   BufferMode:= MC_BUFFER_MODE__mcAborting ,
   Options:= H00000000 ,
   InGear=> bInGear ,
   Busy=> bGearBusy ,
   Active=> bGearActive ,
   CommandAborted=> bGearCommandAborted ,
   Error=> bGearError ,
   ErrorID=> uwGearErrorID
);
```

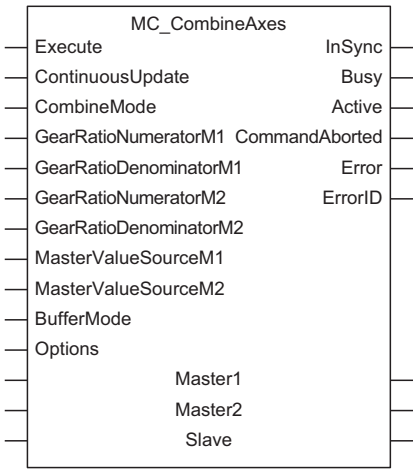
MC_CombineAxes

Adds or subtracts the movement amounts of the two specified master axes, then performs positioning using said value as the commanded position.

Ladder



FBD/LD



ST

```
MC_CombineAxes(
  Master1:= ?AXIS_REF? ,
  Master2:= ?AXIS_REF? ,
  Slave:= ?AXIS_REF? ,
  Execute:= ?BOOL? ,
  ContinuousUpdate:= ?BOOL? ,
  CombineMode:= ?INT? ,
  GearRatioNumeratorM1:= ?DINT? ,
  GearRatioDenominatorM1:= ?DWORD? ,
  GearRatioNumeratorM2:= ?DINT? ,
  GearRatioDenominatorM2:= ?DWORD? ,
  MasterValueSourceM1:= ?INT? ,
  MasterValueSourceM2:= ?INT? ,
  BufferMode:= ?INT? ,
  Options:= ?DWORD? ,
  InSync=> ?BOOL? ,
  Busy=> ?BOOL? ,
  Active=> ?BOOL? ,
  CommandAborted=> ?BOOL? ,
  Error=> ?BOOL? ,
  ErrorID=> ?WORD?
);
```

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Addition/Subtraction Positioning	56	8	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
Master1	Master axis 1	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)
Master2	Master axis 2	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)
Slave	Slave axis	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Execute command	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MC_CombineAxes (Addition/Subtraction Positioning).
ContinuousUpdate	Continuous update	BOOL	At start	TRUE, FALSE	FALSE	This variable sets whether to enable or disable continuous change of Addition/subtraction method selection (CombineMode), Master axis 1 gear ratio numerator (GearRatioNumeratorM1), Master axis 1 gear ratio denominator (GearRatioDenominatorM1), Master axis 2 gear ratio numerator (GearRatioNumeratorM2), and Master axis 2 gear ratio denominator (GearRatioDenominatorM2). • FALSE: Disable • TRUE: Enable
CombineMode	Addition/subtraction method selection	INT (MC_COMBINE_MODE)	At start/ Retrigger possible/ Continuous update possible	0, 1	1	This variable sets the method used to combine the movement amounts of Master axis 1 (Master1) and Master axis 2 (Master2). • 0: Add positions of 2 input axes (mcAddAxes) • 1: Subtract positions of 2 input axes (mcSubAxes) For details, refer to the following. ☞ Page 361 Addition/subtraction method selection (CombineMode)
GearRatioNumeratorM1	Master axis 1 gear ratio numerator	DINT	At start/ Retrigger possible/ Continuous update possible	-2147483648 to 2147483647	1	This variable sets the numerator value of Master axis 1 (Master1). For details, refer to the following. ☞ Page 362 Master axis 1 gear ratio numerator (GearRatioNumeratorM1)/Master axis 1 gear ratio denominator (GearRatioDenominatorM1)
GearRatioDenominatorM1	Master axis 1 gear ratio denominator	DWORD(UDINT)	At start/ Retrigger possible/ Continuous update possible	1 to 2147483647	1	This variable sets the denominator value of Master axis 1 (Master1). For details, refer to the following. ☞ Page 362 Master axis 1 gear ratio numerator (GearRatioNumeratorM1)/Master axis 1 gear ratio denominator (GearRatioDenominatorM1)
GearRatioNumeratorM2	Master axis 2 gear ratio numerator	DINT	At start/ Retrigger possible/ Continuous update possible	-2147483648 to 2147483647	1	This variable sets the numerator value of Master axis 2 (Master2). For details, refer to the following. ☞ Page 362 Master axis 2 gear ratio numerator (GearRatioNumeratorM2)/Master axis 2 gear ratio denominator (GearRatioDenominatorM2)
GearRatioDenominatorM2	Master axis 2 gear ratio denominator	DWORD(UDINT)	At start/ Retrigger possible/ Continuous update possible	1 to 2147483647	1	This variable sets the denominator value of Master axis 2 (Master2). For details, refer to the following. ☞ Page 362 Master axis 2 gear ratio numerator (GearRatioNumeratorM2)/Master axis 2 gear ratio denominator (GearRatioDenominatorM2)
MasterValueSourceM1	Master axis 1 data source	INT (MC_SOURCE)	At start	1, 2, 101, 102	1	This variable sets the data source of Master axis 1 (Master1). • 1: Set value (mcSetValue) • 2: Actual value (mcActualValue) • 101: Latest set value (mcLatestSetValue) • 102: Latest actual value (mcLatestActualValue) For details, refer to the following. ☞ Page 362 Master axis 1 data source (MasterValueSourceM1)/Master axis 2 data source (MasterValueSourceM2)

Input variable	Name	Data type	Input	Setting range	Default value	Description
MasterValueSourceM2	Master axis 2 data source	INT (MC_SOURCE)	At start	1, 2, 101, 102	1	This variable sets the data source of Master axis 2 (Master2). • 1: Set value (mcSetValue) • 2: Actual value (mcActualValue) • 101: Latest set value (mcLatestSetValue) • 102: Latest actual value (mcLatestActualValue) For details, refer to the following. ☞ Page 362 Master axis 1 data source (MasterValueSourceM1)/Master axis 2 data source (MasterValueSourceM2)
BufferMode	Buffer mode	INT (MC_BUFFER_MODE)	At start	0, 1	0	This variable sets the buffer mode. • 0: Aborting (mcAborting) • 1: Buffered (mcBuffered) For details, refer to the following. ☞ Page 363 Buffer mode (BufferMode)
Options	Options	DWORD(HEX)	At start	00000000H	00000000H	Set this variable to "00000000H". *: When a value other than "00000000H" is set, "Out of Options Range (error code: 1A4EH)" occurs.

■Output variables

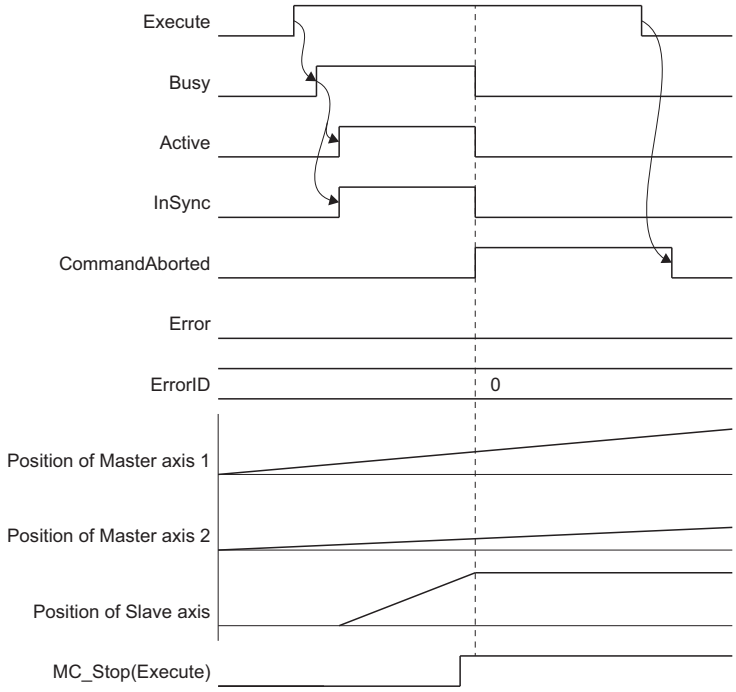
Output variable	Name	Data type	Default value	Description
InSync	In synchronization	BOOL	FALSE	This variable becomes TRUE when Slave axis (Slave) starts synchronization.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MC_CombineAxes (Addition/Subtraction Positioning) is executed.
Active	Controlling	BOOL	FALSE	This variable becomes TRUE when Slave axis (Slave) is being controlled.
CommandAborted	Abortion of execution	BOOL	FALSE	This variable becomes TRUE when execution of MC_GearIn (Gear Operation Start) is aborted. This variable becomes FALSE when Execute command (Execute) becomes FALSE.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. MELSEC iQ-R Motion Module User's Manual (Application)

Processing details

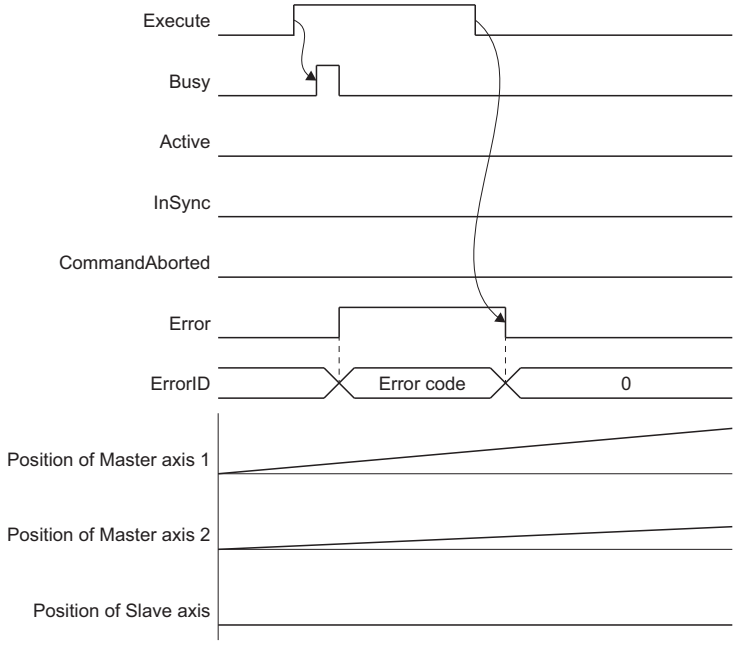
- In MC_CombineAxes (Addition/Subtraction Positioning), Addition/subtraction method selection (CombineMode), Master axis 1 gear ratio numerator (GearRatioNumeratorM1), Master axis 1 gear ratio denominator (GearRatioDenominatorM1), Master axis 2 gear ratio numerator (GearRatioNumeratorM2), Master axis 2 gear ratio denominator (GearRatioDenominatorM2), and Buffer mode (BufferMode) are set, then the addition/subtraction positioning is performed.
- To stop the operation, perform MC_Stop (Forced Stop).
- Combines the movement amounts of Master axis 1 (Master1) and Master axis 2 (Master2). In addition, a gear ratio can be set for each master axis. Combining the movement amounts adds or subtracts the numerical values regardless of the units of Master axis 1 (Master1) and Master axis 2 (Master2).

■Timing chart

- When the operation completes normally



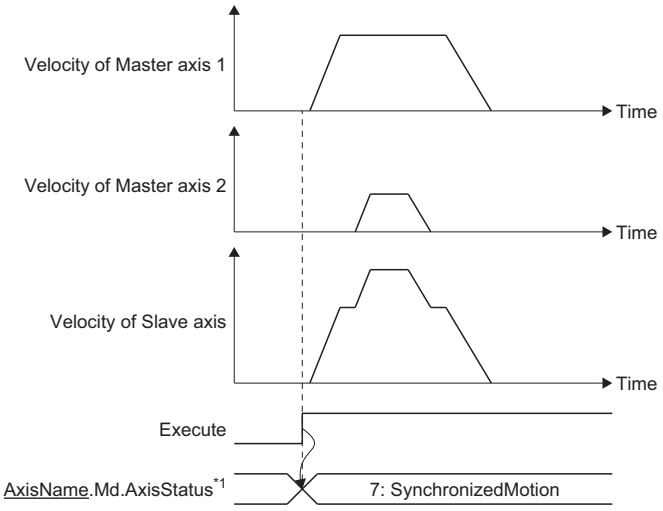
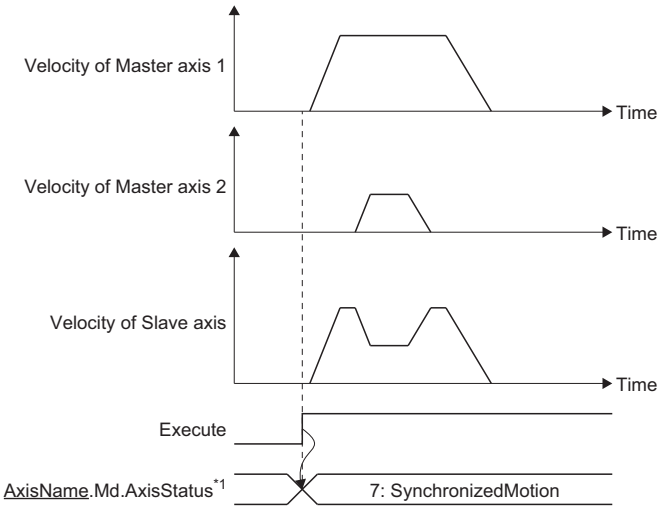
- When an error occurs



■Addition/subtraction method selection (CombineMode)

Sets the method used to combine the movement amounts of Master axis 1 (Master1) and Master axis 2 (Master2).

Combining the movement amounts adds or subtracts the numerical values regardless of the units of Master axis 1 (Master1) and Master axis 2 (Master2).

Setting value	Description
0: Add positions of 2 input axes (mcAddAxes)	Adds the movement amounts of Master axis 1 (Master1) and Master axis 2 (Master2) and outputs the value. • Movement amount of Slave axis (Slave) = Movement amount of Master axis 1 (Master1) + Master axis 2 (Master2) 
1: Subtract positions of 2 input axes (mcSubAxes)	Subtracts the movement amount of Master axis 2 (Master2) from Master axis 1 (Master1) and outputs the value. • Movement amount of Slave axis (Slave) = Movement amount of Master axis 1 (Master1) - Master axis 2 (Master2) 

*1 The axis status of Slave axis (Slave).

■Master axis 1 gear ratio numerator (GearRatioNumeratorM1)/Master axis 1 gear ratio denominator (GearRatioDenominatorM1)

Master axis 1 gear ratio numerator (GearRatioNumeratorM1) and Master axis 1 gear ratio denominator (GearRatioDenominatorM1) are set together.

Master axis 1 gear ratio numerator (GearRatioNumeratorM1) sets the numerator value used when converting and combining the movement amount of Master axis 1 (Master1).

Master axis 1 gear ratio denominator (GearRatioDenominatorM1) sets the denominator value used when converting and combining the movement amount of Master axis 1 (Master1).

The movement amount is converted as shown below.

$$\text{Master axis 1 (Master1) movement amount after conversion} = \text{Master axis 1 (Master1) movement amount before conversion} \times \frac{\text{Master axis 1 gear ratio numerator (GearRatioNumeratorM1)}}{\text{Master axis 1 gear ratio denominator (GearRatioDenominatorM1)}}$$

- Set a positive value in Master axis 1 gear ratio denominator (GearRatioDenominatorM1).
- The movement amount can be reversed and combined by setting a negative value in Master axis 1 gear ratio numerator (GearRatioNumeratorM1).
- To combine and have the movement amount of Master axis 1 (Master1) be "0" after conversion, set "0" as the setting value in Master axis 1 gear ratio numerator (GearRatioNumeratorM1).

■Master axis 2 gear ratio numerator (GearRatioNumeratorM2)/Master axis 2 gear ratio denominator (GearRatioDenominatorM2)

Master axis 2 gear ratio numerator (GearRatioNumeratorM2) and Master axis 2 gear ratio denominator (GearRatioDenominatorM2) are set together.

Master axis 2 gear ratio numerator (GearRatioNumeratorM2) sets the numerator value used when converting and combining the movement amount of Master axis 2 (Master2).

Master axis 2 gear ratio denominator (GearRatioDenominatorM2) sets the denominator value used when converting and combining the movement amount of Master axis 2 (Master2).

The movement amount is converted as shown below.

$$\text{Master axis 2 (Master2) movement amount after conversion} = \text{Master axis 2 (Master2) movement amount before conversion} \times \frac{\text{Master axis 2 gear ratio numerator (GearRatioNumeratorM2)}}{\text{Master axis 2 gear ratio denominator (GearRatioDenominatorM2)}}$$

- Set a positive value in Master axis 2 gear ratio denominator (GearRatioDenominatorM2).
- The movement amount can be reversed and combined by setting a negative value in Master axis 2 gear ratio numerator (GearRatioNumeratorM2).
- To combine and have the movement amount of Master axis 2 (Master2) be "0" after conversion, set "0" as the setting value in Master axis 2 gear ratio numerator (GearRatioNumeratorM2).

■Master axis 1 data source (MasterValueSourceM1)/Master axis 2 data source (MasterValueSourceM2)

Slave axis (Slave) sets the position type of Master axis 1 (Master1) or Master axis 2 (Master2) that will execute single axis synchronous control.

The operation may differ depending on the execution order of the FBs or if the operation cycles of Master axis 1 (Master1) or Master axis 2 (Master2) and Slave axis (Slave) are different.

Operates with the same value as the set value when specifying Actual value in Master axis 1 data source (MasterValueSourceM1) or Master axis 2 data source (MasterValueSourceM2) while an axis type other than real axis is set in Master axis 1 (Master1) or Master axis 2 (Master2).

Setting value	Description
1: Set value (mcSetValue)	Uses the commanded position of the master axis in the previous operation cycle.
2: Actual value (mcActualValue)	Uses the actual position of the master axis in the previous operation cycle.
101: Latest set value (mcLatestSetValue)	Uses the commanded position of the master axis in the current operation cycle.
102: Latest actual value (mcLatestActualValue)	Uses the actual position of the master axis in the current operation cycle.

Point

When Master axis 1 data source (MasterValueSourceM1) or Master axis 2 data source (MasterValueSourceM2) are set to "1: Set value (mcSetValue)" or "101: Latest set value (mcLatestSetValue)" and the master axis becomes servo OFF by a servo alarm or forced stop, the amount of the value change may be too large. This can be prevented by setting Master axis 1 data source (MasterValueSourceM1) or Master axis 2 data source (MasterValueSourceM2) to "2: Actual value (mcActualValue)" or "102: Latest actual value (mcLatestActualValue)".

■Buffer mode (BufferMode)

Sets the operation used to perform multiple start (buffer mode).

The following buffer modes are settable in MC_CombineAxes (Addition/Subtraction Positioning).

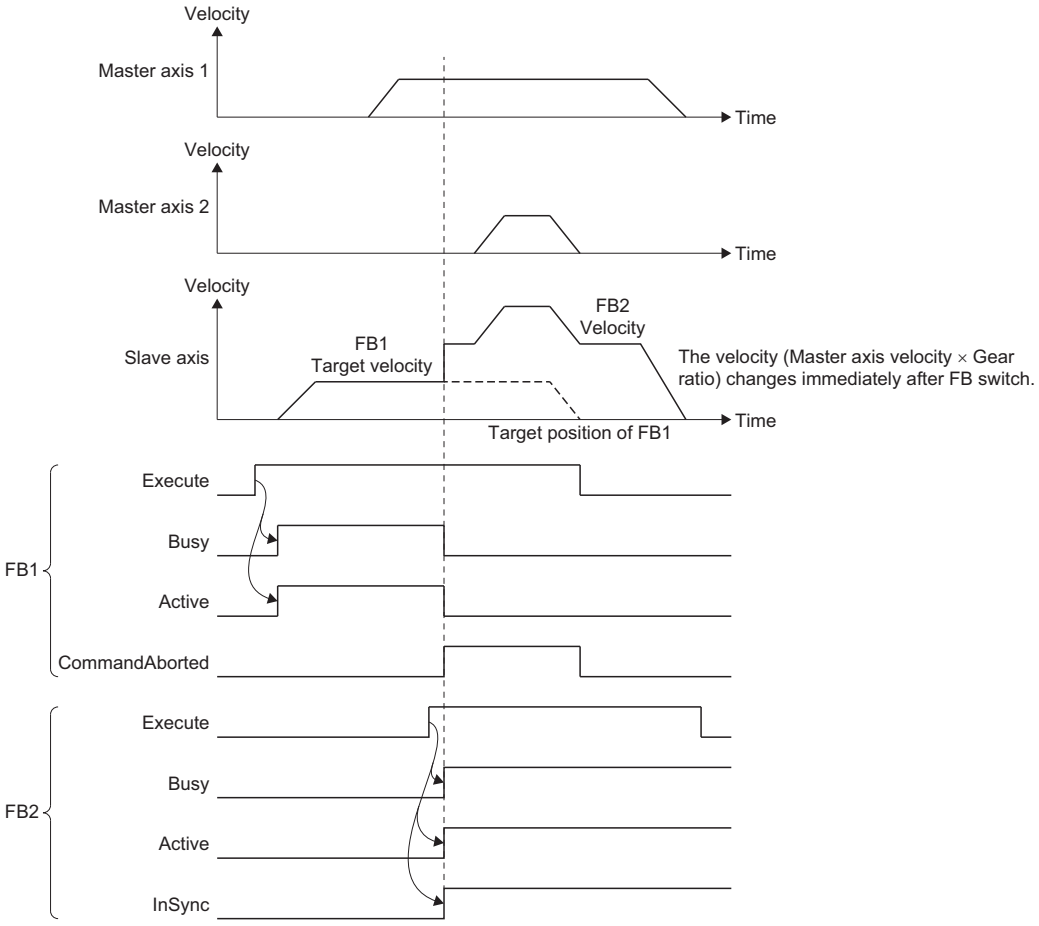
The velocity of MC_CombineAxes (Addition/Subtraction Positioning) follows the synchronizing FB (master axis) and changes immediately based on the master axis velocity and the set gear data when the FB is switched.

Setting value	Description
0: Aborting (mcAborting)	Aborts (cancels) the under-control FB and executes the next FB immediately.
1: Buffered (mcBuffered)	Buffers the next FB on the under-control FB. If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after completion of the under-control FB.

Ex.

When linked start FB2 (Slave axis (Slave): MC_CombineAxes (Addition/Subtraction Positioning)) has Buffer mode (BufferMode) set to "0: Aborting (mcAborting)" while FB1 is being executed

· When Addition/subtraction method selection (CombineMode) is set to "0: Add positions of 2 input axes (mcAddAxes)"



Velocity of FB2 occurs after In synchronization (InSync) becomes TRUE.

Point

For details of multiple start (buffer mode), refer to the following.

MELSEC iQ-R Motion Module User's Manual (Application)

Program example

In the following program example, the addition/subtraction positioning start command (bCombineAxesStart) is set to TRUE and operation of the slave axis (Axis0001) is performed using the added/subtracted movement amount value from the two master axes being Axis 2 (Axis0002) of the JOG 1 FB and Axis 3 (Axis0003) of the JOG 1 FB.

• Settings

Item		Setting value	
		Data for JOG 1	Data for JOG 2
Data settings for JOG operation	Velocity	1000.0	500.0
	Acceleration	5000.0	5000.0
	Deceleration	5000.0	5000.0
	Jerk	0.0	0.0

■Axes

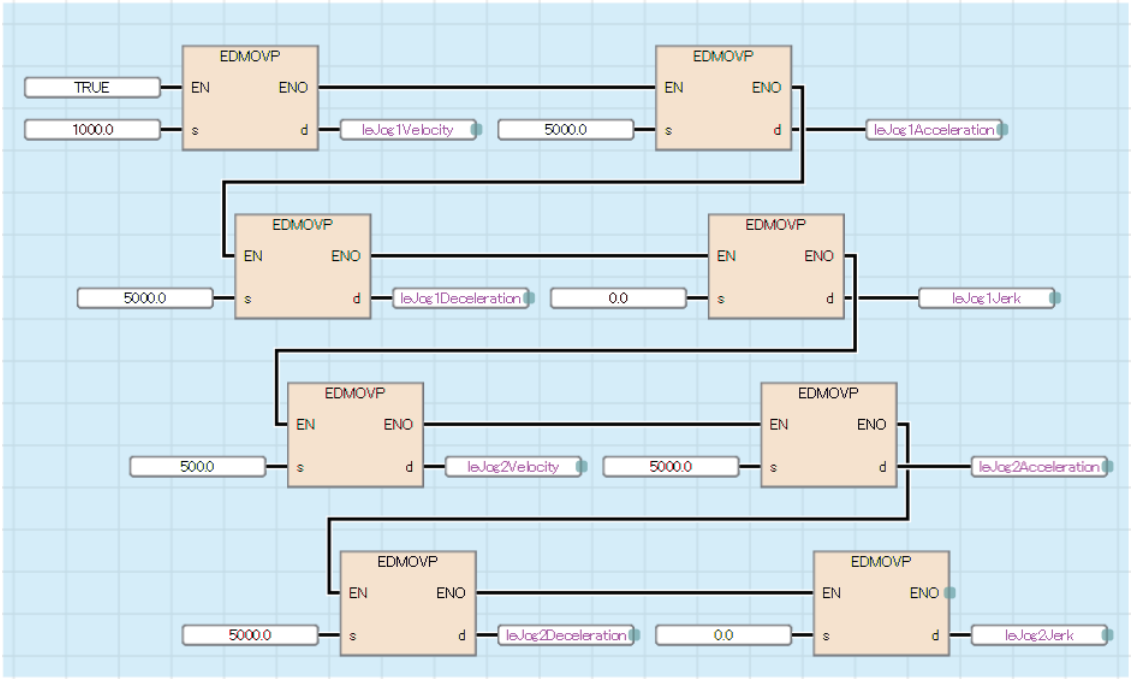
Axis No.	Label name	Data type	Comment
1	Axis0001	AXIS_REF	Axis 1
2	Axis0002	AXIS_REF	Axis 2
3	Axis0003	AXIS_REF	Axis 3

■Labels used

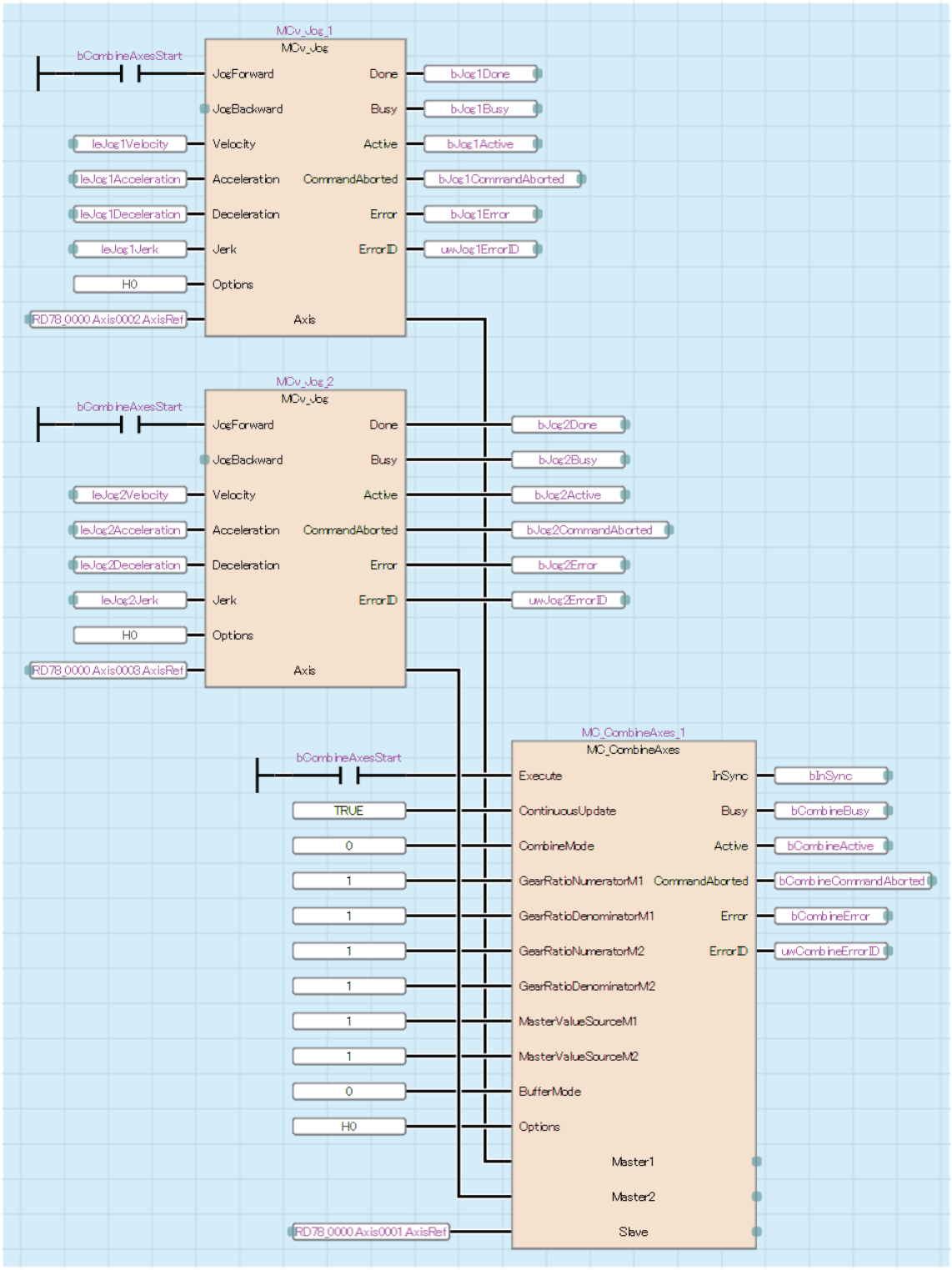
Label name	Data type	Comment
MC_CombineAxes_1	MC_CombineAxes	Addition/Subtraction Positioning FB
bCombineAxesStart	Bit	Addition/subtraction positioning start command
bInSync	Bit	In synchronization
bCombineBusy	Bit	Executing
bCombineActive	Bit	Controlling
bCombineCommandAborted	Bit	Abortion of execution
bCombineError	Bit	Error
uwCombineErrorID	Word [unsigned]/bit string [16-bit]	Error code
MCv_Jog_1	MCv_Jog	JOG 1 FB
leJog1Velocity	Double-precision real number	JOG velocity
leJog1Acceleration	Double-precision real number	JOG acceleration
leJog1Deceleration	Double-precision real number	JOG deceleration
leJog1Jerk	Double-precision real number	JOG jerk
bJog1Done	Bit	Execution completion
bJog1Busy	Bit	Executing
bJog1Active	Bit	Controlling
bJog1CommandAborted	Bit	Abortion of execution
bJog1Error	Bit	Error
uwJog1ErrorID	Word [unsigned]/bit string [16-bit]	Error code
MCv_Jog_2	MCv_Jog	JOG 2 FB
leJog2Velocity	Double-precision real number	JOG velocity
leJog2Acceleration	Double-precision real number	JOG acceleration
leJog2Deceleration	Double-precision real number	JOG deceleration
leJog2Jerk	Double-precision real number	JOG jerk
bJog2Done	Bit	Execution completion
bJog2Busy	Bit	Executing
bJog2Active	Bit	Controlling
bJog2CommandAborted	Bit	Abortion of execution
bJog2Error	Bit	Error
uwJog2ErrorID	Word [unsigned]/bit string [16-bit]	Error code

■FBD/LD program (CPU module side)

• Data settings for JOG operation



• JOG 1/JOG 2/Addition/Subtraction Positioning



■ST program (Motion module side)

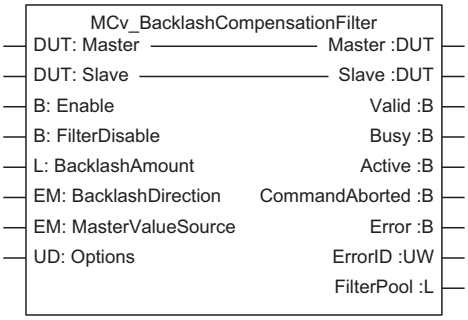
```
//-----Data settings for JOG operation-----
leJog1Velocity:= 1000.0;
leJog1Acceleration:= 5000.0;
leJog1Deceleration:= 5000.0;
leJog1Jerk:= 0.0;
leJog2Velocity:= 500.0;
leJog2Acceleration:= 5000.0;
leJog2Deceleration:= 5000.0;
leJog2Jerk:= 0.0;

//-----JOG 1-----
MCv_Jog_1(
   Axis:= Axis0002.AxisRef ,
   JogForward:= bCombineAxesStart ,
   Velocity:= leJog1Velocity ,
   Acceleration:= leJog1Acceleration ,
   Deceleration:= leJog1Deceleration ,
   Jerk:= leJog1Jerk ,
   Options:= H00000000 ,
   Done=> bJog1Done ,
   Busy=> bJog1Busy ,
   Active=> bJog1Active ,
   CommandAborted=> bJog1CommandAborted ,
   Error=> bJog1Error ,
   ErrorID=> uwJog1ErrorID
);

//-----JOG 2-----
MCv_Jog_2(
   Axis:= Axis0003.AxisRef ,
   JogForward:= bCombineAxesStart ,
   Velocity:= leJog2Velocity ,
   Acceleration:= leJog2Acceleration ,
   Deceleration:= leJog2Deceleration ,
   Jerk:= leJog2Jerk ,
   Options:= H00000000 ,
   Done=> bJog2Done ,
   Busy=> bJog2Busy ,
   Active=> bJog2Active ,
   CommandAborted=> bJog2CommandAborted ,
   Error=> bJog2Error ,
   ErrorID=> uwJog2ErrorID
);

//-----Addition/Subtraction Positioning-----
MC_CombineAxes_1(
   Master1:= Axis0002.AxisRef ,
   Master2:= Axis0003.AxisRef ,
   Slave:= Axis0001.AxisRef ,
   Execute:= bCombineAxesStart ,
   ContinuousUpdate:= TRUE ,
```

```
    CombineMode:= MC_COMBINE_MODE__mcAddAxes ,
    GearRatioNumeratorM1:= 1 ,
    GearRatioDenominatorM1:= 1 ,
    GearRatioNumeratorM2:= 1 ,
    GearRatioDenominatorM2:= 1 ,
    MasterValueSourceM1:= MC_SOURCE__mcSetValue ,
    MasterValueSourceM2:= MC_SOURCE__mcSetValue ,
    BufferMode:= MC_BUFFER_MODE__mcAborting ,
    Options:= H00000000 ,
    InSync=> bInSync ,
    Busy=> bCombineBusy ,
    Active=> bCombineActive ,
    CommandAborted=> bCombineCommandAborted ,
    Error=> bCombineError ,
    ErrorID=> uwCombineErrorID
);
```

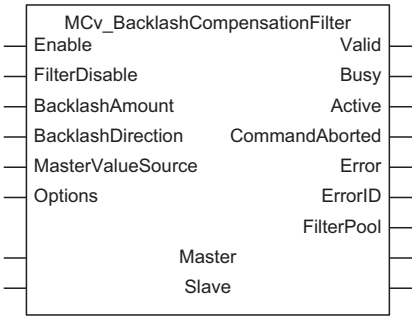
MCv_BacklashCompensationFilter

Performs filter processing according to the movement direction to compensate the backlash amount in the machine system.

Ladder



FBD/LD



ST

```
MCv_BacklashCompensationFilter(
  Master:= ?AXIS_REF? ,
  Slave:= ?AXIS_REF? ,
  Enable:= ?BOOL? ,
  FilterDisable:= ?BOOL? ,
  BacklashAmount:= ?LREAL? ,
  BacklashDirection:= ?INT? ,
  MasterValueSource:= ?INT? ,
  Options:= ?DWORD? ,
  Valid=> ?BOOL? ,
  Busy=> ?BOOL? ,
  Active=> ?BOOL? ,
  CommandAborted=> ?BOOL? ,
  Error=> ?BOOL? ,
  ErrorID=> ?WORD? ,
  FilterPool=> ?LREAL?
);
```

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Backlash Compensation Filter	40	24	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
Master	Master axis	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)
Slave	Slave axis	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Enable	Enable	BOOL	Always	TRUE, FALSE	FALSE	When this variable is TRUE, executes MCv_BacklashCompensationFilter (Backlash Compensation Filter).
FilterDisable	Filter disable	BOOL	Always	TRUE, FALSE	FALSE	This variable sets the method used for the filter processing. • FALSE: Operates according to the filter setting. • TRUE: Disables the filter and transmits the Master axis (Master) input to Slave axis (Slave) as it is. *: When this variable is set to TRUE, the value in Filter cumulated value (FilterPool) becomes "0" and the disposed value from Filter cumulated value (FilterPool) is stored in Disposed filter cumulated value (PurgedFilterPool).
BacklashAmount	Backlash amount	LREAL	At start	0.0 to 2500000000.0[U]	0.0	This variable sets the backlash amount used in the machine system.

Input variable	Name	Data type	Input	Setting range	Default value	Description
BacklashDirection	Backlash compensation direction	INT (MC_DIRECTION)	At start	1, 2	0	This variable sets the backlash compensation direction. • 1: Positive direction (mcPositiveDirection) • 2: Negative direction (mcNegativeDirection) *: When this setting is omitted, "Out of backlash compensation direction range (error code: 34A0H)" occurs. For details, refer to the following. ☞ Page 374 Backlash compensation direction (BacklashDirection)
MasterValueSource	Master axis data source selection	INT (MC_SOURCE)	At start	1, 101	1	This variable sets the data source of Master axis (Master). • 1: Set value (mcSetValue) • 101: Latest set value (mcLatestSetValue) For details, refer to the following. ☞ Page 374 Master axis data source selection (MasterValueSource)
Options	Options	DWORD(HEX)	At start	00000000H	00000000H	This variable sets the function options. For details, refer to the following. ☞ Page 375 Options (Options)

■Output variables

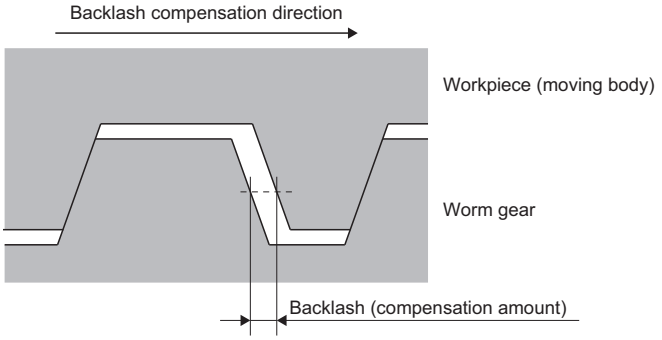
Output variable	Name	Data type	Default value	Description
Valid	Output value valid	BOOL	FALSE	This variable becomes TRUE when the output value is valid.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MCv_BacklashCompensationFilter (Backlash Compensation Filter) is executed.
Active	Controlling	BOOL	FALSE	This variable becomes TRUE when MCv_BacklashCompensationFilter (Backlash Compensation Filter) is controlling the axis.
CommandAborted	Abortion of execution	BOOL	FALSE	This variable becomes TRUE when execution of MCv_BacklashCompensationFilter (Backlash Compensation Filter) is aborted.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. 📖MELSEC iQ-R Motion Module User's Manual (Application)
FilterPool	Filter cumulated value	LREAL	0.0	This variable outputs the slave axis movement amount cumulated in filter processing. When the backlash filter is used, the compensation amount transmitted to the slave axis is the reversed positive/negative value in Filter cumulated value (FilterPool).

■Public variables

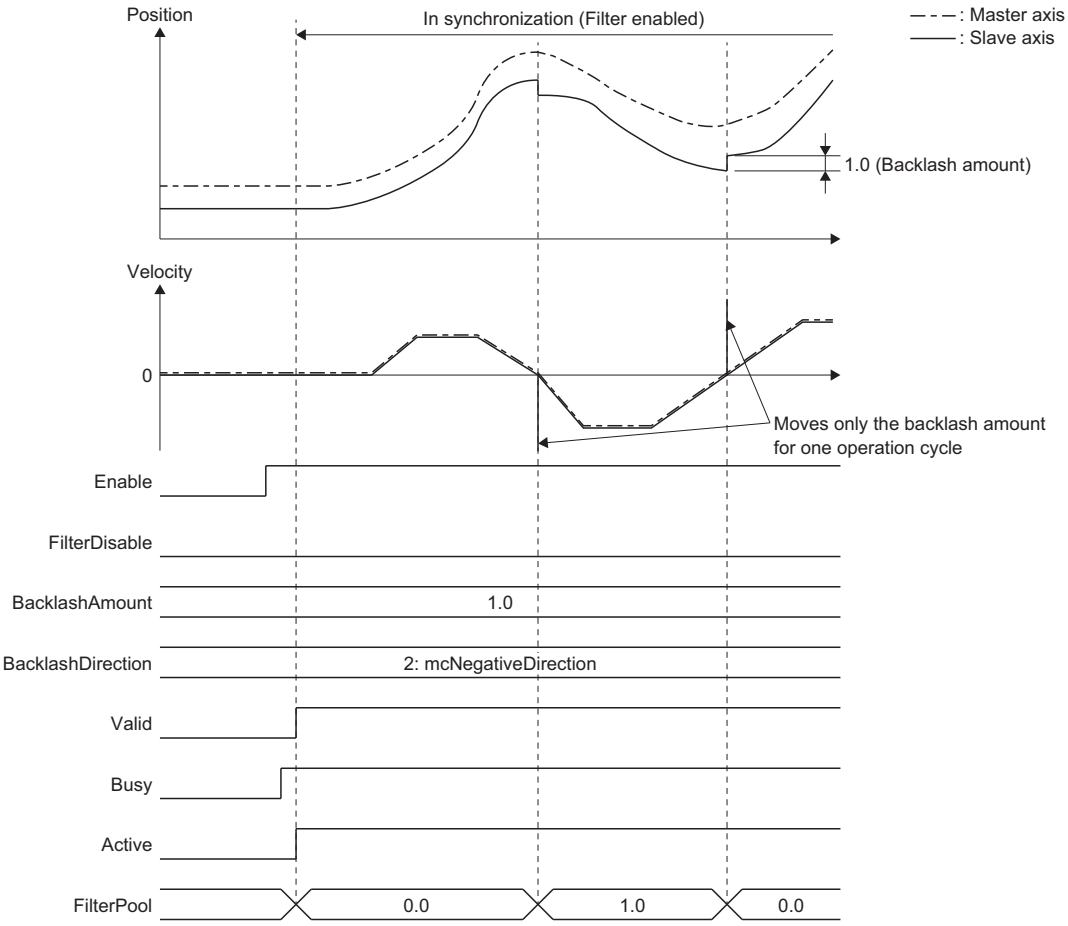
Public variable	Name	Data type	Default value	Description
PurgedFilterPool	Disposed filter cumulated value	LREAL	0.0	When Filter disable (FilterDisable) is set to TRUE, this variable stores the disposed value from Filter cumulated value (FilterPool). Disposed filter cumulated value (PurgedFilterPool) does not cumulate. This variable is 0 cleared at FB start.

Processing details

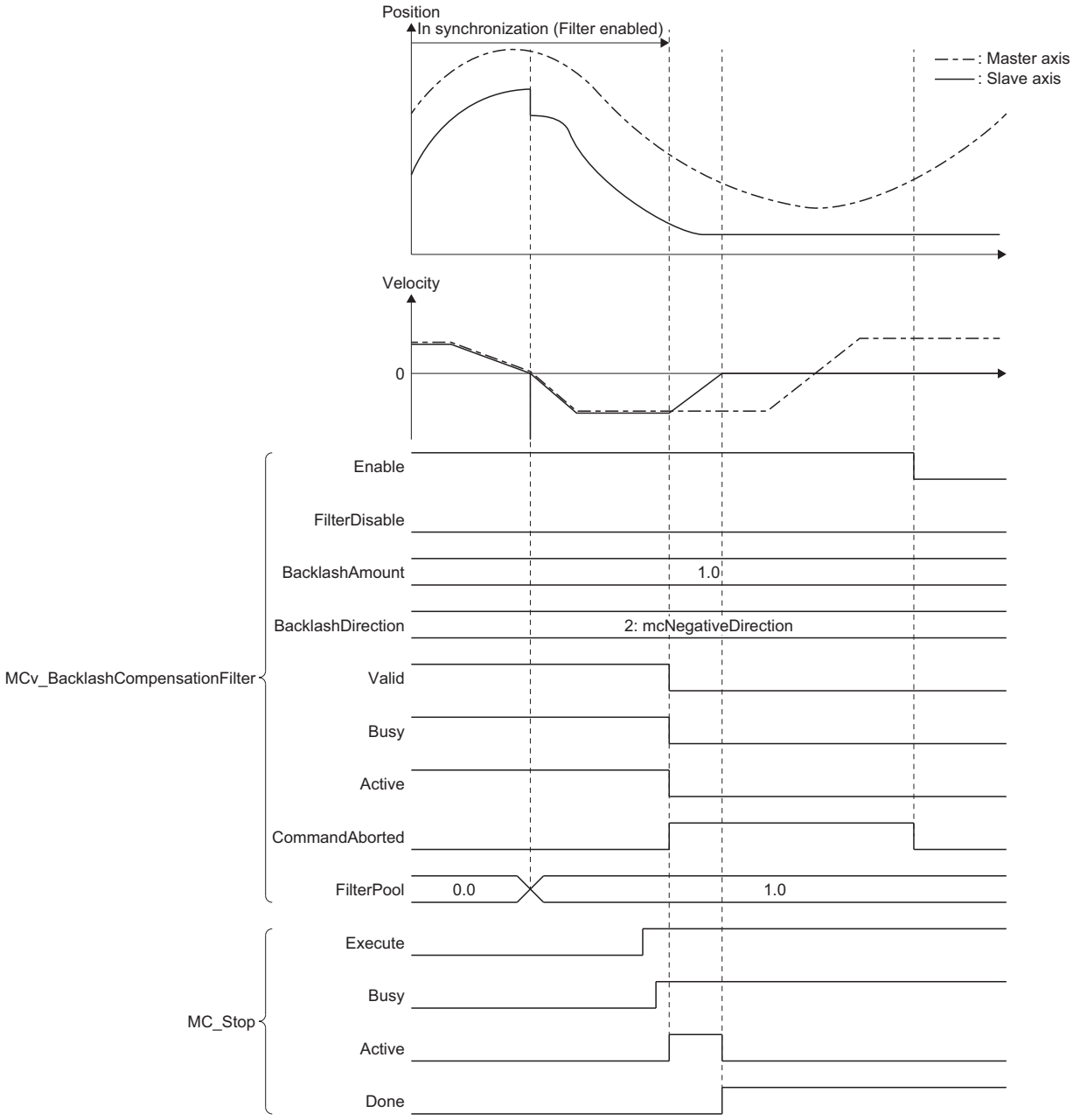
- The backlash compensation filter is used to compensate for the backlash amount in the machine system.



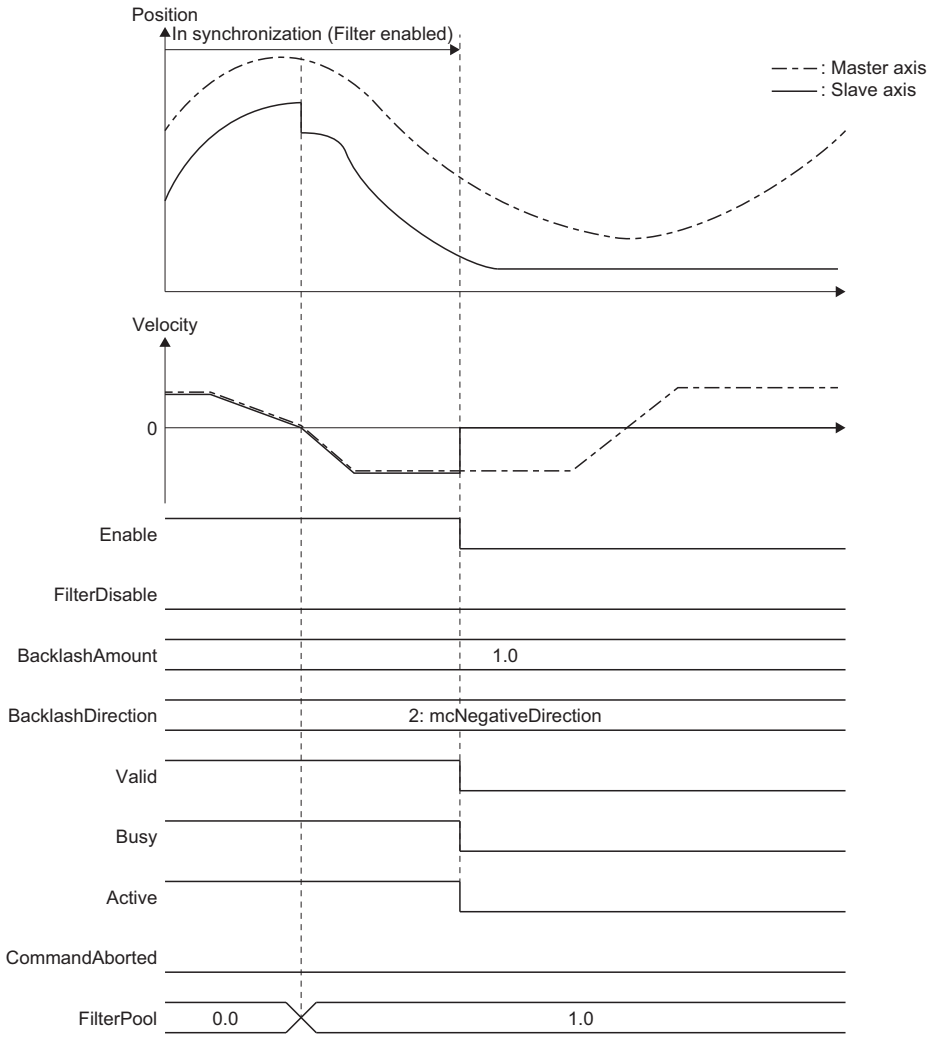
- When the movement direction of Master axis (Master) is the same as Backlash compensation direction (BacklashDirection), Slave axis (Slave) synchronizes to the position equivalent to the Master axis (Master) current position shifted in the Backlash direction (BacklashDirection) by the value of Backlash amount (BacklashAmount). When the movement direction of Master axis (Master) is the reverse of Backlash compensation direction (BacklashDirection), Slave axis (Slave) synchronizes to the Master axis (Master) current position.
  Whenever the Master axis (Master) movement direction changes, backlash compensation is performed with the Slave axis (Slave) extra movement amount limited to only the value of Backlash amount (BacklashAmount).



- This section describes the stop operation specific to the backlash compensation filter.
  When the stop factor occurs on Slave axis (Slave), Slave axis (Slave) starts the stop operation from the synchronous position. The compensated movement amount (the value from Filter cumulated value (FilterPool)) will not be restored to its original value.

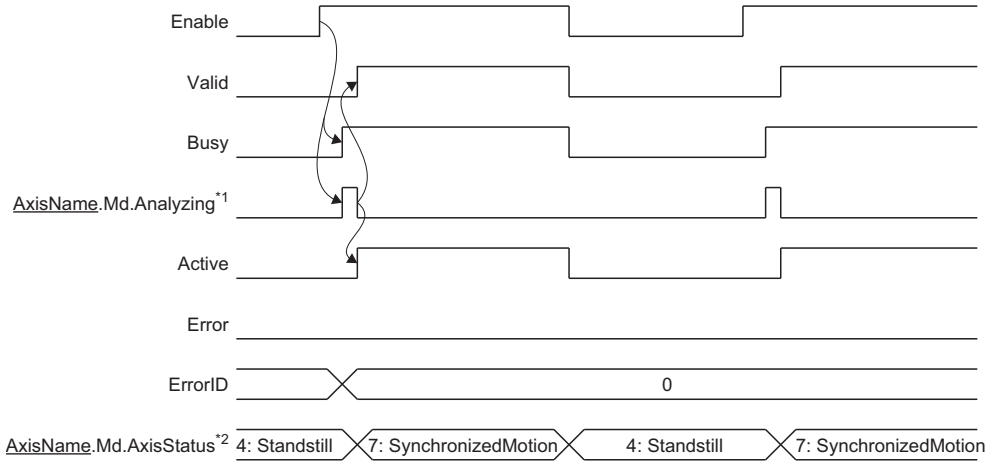


- Operation at filter disabled (at Enable (Enable) trailing edge)
  Slave axis (Slave) stops immediately at the synchronous position upon reaching the Enable (Enable) trailing edge. The compensated movement amount (the value from Filter cumulated value (FilterPool)) will not be restored to its original value.



■Timing chart

- When the operation completes normally



*1 "Analyzing" of Slave axis (Slave).
*2 The axis status of Slave axis (Slave).

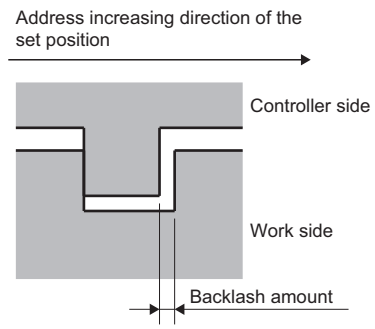
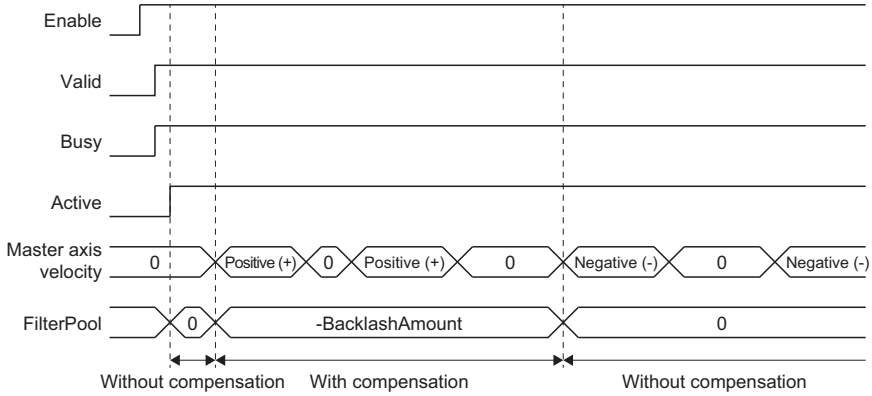
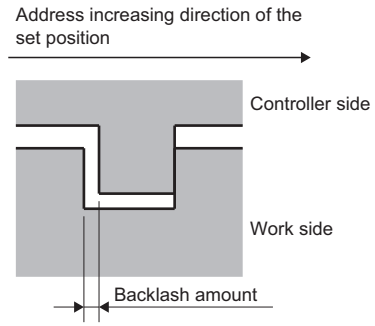
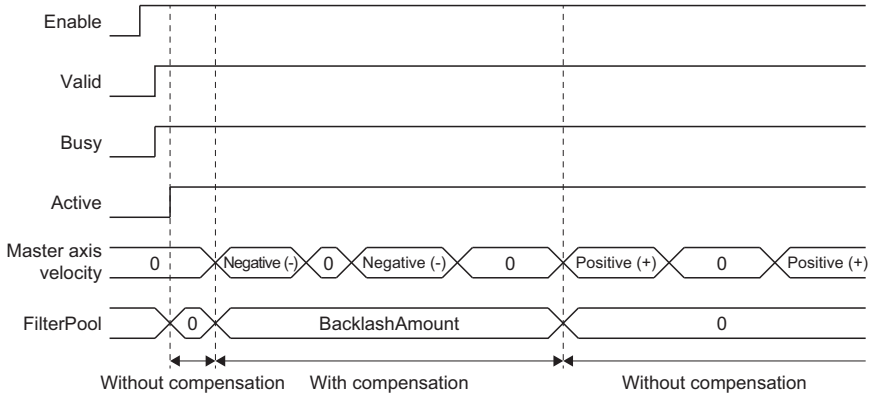
- When an error occurs
  For the timing chart when an error occurs, refer to the following.
  ☞ Page 13 Basic operation of Enable (Enable) type Motion control FBs

■Backlash compensation direction (BacklashDirection)

Sets the backlash compensation direction.

Outputs the compensation amount to Slave axis (Slave) when the Master axis (Master) movement direction is the value of Backlash compensation direction (BacklashDirection).

When the backlash compensation filter is enabled (when starting the FB with Filter disable (FilterDisable) set to FALSE, or when Filter disable (FilterDisable) reaches the trailing edge during execution), compensates to the address increasing direction of the Master axis (Master) set position so that the backlash status of the machine system is in the following relation.

Setting value	Description
1: Positive direction (mcPositiveDirection) 	When the movement direction of Master axis (Master) is the positive direction, Slave axis (Slave) synchronizes to the position equivalent to the Master axis (Master) current position shifted in the positive direction by the value of Backlash amount (BacklashAmount). 
2: Negative direction (mcNegativeDirection) 	When the movement direction of Master axis (Master) is the negative direction, Slave axis (Slave) synchronizes to the position equivalent to the Master axis (Master) current position shifted in the negative direction by the value of Backlash amount (BacklashAmount). 

■Master axis data source selection (MasterValueSource)

Slave axis (Slave) sets the position type of Master axis (Master) that will execute single axis synchronous control.

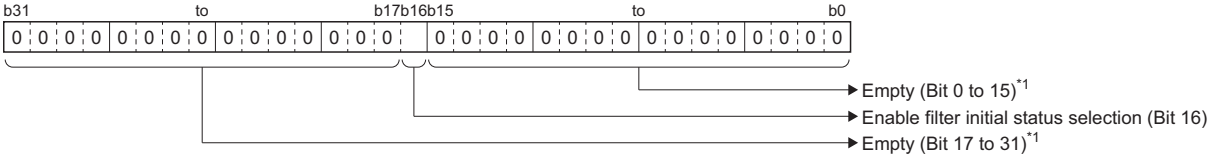
The operation may differ depending on the execution order of the FBs or if the operation cycles of Master axis (Master) and Slave axis (Slave) are different.

Setting value	Description
1: Set value (mcSetValue)	Uses the commanded position of the master axis in the previous operation cycle.
101: Latest set value (mcLatestSetValue)	Uses the commanded position of the master axis in the current operation cycle.

■Options (Options)

Sets the function options for MCv_BacklashCompensationFilter (Backlash Compensation Filter) by specifying the bit.

Contents that are set by specifying the bit are shown below.



*1 Set empty areas to "0". When a value other than "0" is set, "Out of Options Range (error code: 1A4EH)" occurs.

Bit	Name	Description
16	Enable filter initial status selection	Sets the initial status for when the backlash compensation filter is enabled (when starting the FB with Filter disable (FilterDisable) set to FALSE, or when Filter disable (FilterDisable) reaches the trailing edge during execution). • 0: First compensation *: If a value other than "0" is set, "Out of Options Range (error code: 1A4EH)" occurs and the function does not start.

• Enable filter initial status selection (Bit 16)

Setting value	Description
0: First compensation	Regardless of the Master axis (Master) current direction when the filter is enabled, starts filter operation assuming that the position relation of the Master axis (Master) current position and the Slave axis (Slave) current position do not have the backlash compensation added.

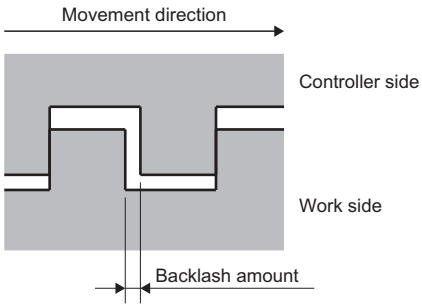
■Procedure for enabling the backlash compensation filter

This section describes the procedure for enabling the backlash compensation filter and a usage example.

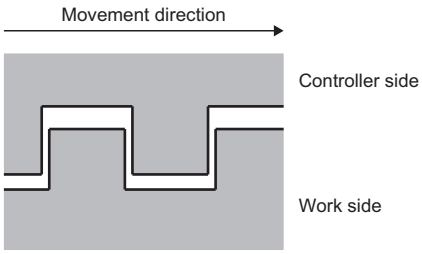
Perform this procedure after the machine position is established by homing, or when restarting the backlash compensation filter after cancelling it midway.

1. Move the real drive axis by homing or JOG operation etc., and maintain the status so that the backlash of the machine system is generated in only one direction as shown in the correct example below.

■Correct example



■Incorrect example



2. Match the Master axis (Master) set position with the real drive axis set position by using the current position change, etc.

3. Set Backlash compensation direction (BacklashDirection) so that it is the reverse direction of the direction that the real drive axis was moved to in Step 1., then start the backlash compensation filter. In addition, start any back part function blocks that are present.

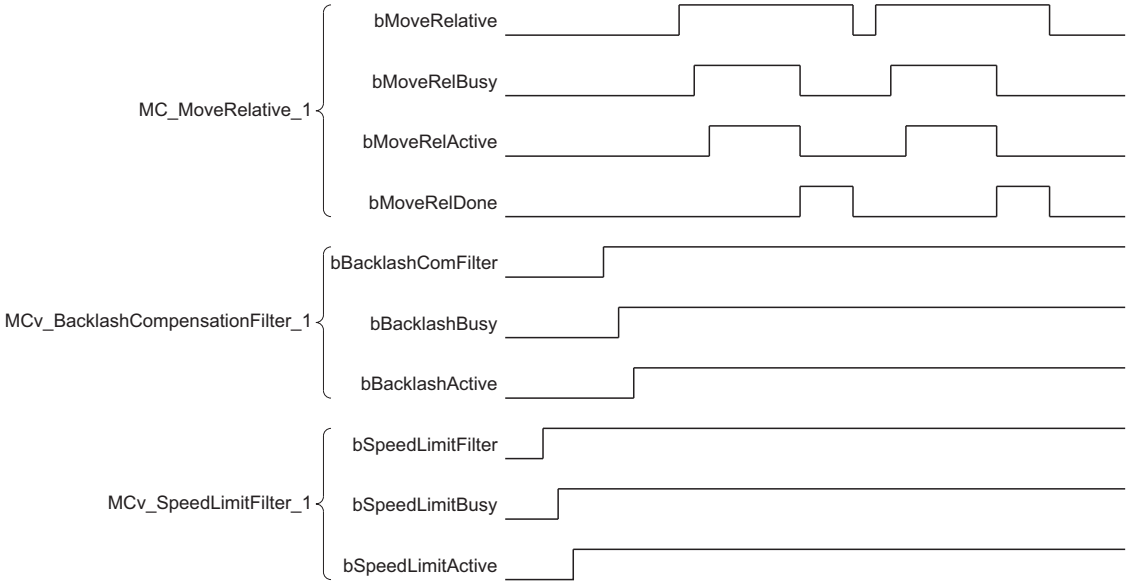
Precautions

- Backlash compensation is a function which the command direction and the backlash status should not change because of external force, etc. Correct compensation will not be performed when this function is used in the following cases.
  · A machine that has external force constantly applied in one direction, such as a vertical axis.
  · A mechanism (such as rack and pinion mechanism) in which the backlash amount changes depending on the machine position.
- When the Master axis (Master) movement direction changes, Slave axis (Slave) moves only the value of Backlash amount (BacklashAmount) for one operation cycle. Depending on the value, it may exceed the command frequency that the speed limit value and the driver unit can accept, so connect the speed limit filter or the smoothing filter to the backlash compensation filter back part.
- The Slave axis (Slave) current position (such as set position, machine feed position, etc.) will be the position to which the value of Backlash amount (BacklashAmount) compensation is added. Backlash amount (BacklashAmount) compensation will not be added to the Master axis (Master) current position.
- Note that if Filter disable (FilterDisable) is switched during operation, the Slave axis (Slave) velocity may change rapidly.
- After enabling the command filter, when the Master axis (Master) command is not transmitted to Slave axis (Slave) or the back part synchronous control FB because the stop factor occurred on Slave axis (Slave), the synchronous position relation deviates. Before enabling the command filter again, execute the synchronous positioning as needed.

Program example

In the following program example, the speed limit filter is connected to the back part of the backlash compensation filter and filter processing is performed.

- Timing chart



- Settings

Item		Setting value
Backlash amount		1.0
Data settings for single axis positioning	Movement amount	1000.0
	Velocity	20.0
	Acceleration	300.0
	Deceleration	300.0
	Jerk	0.0
Positive direction/negative direction limit value	Positive direction limit value	100.0
	Negative direction limit value	100.0

■Axes

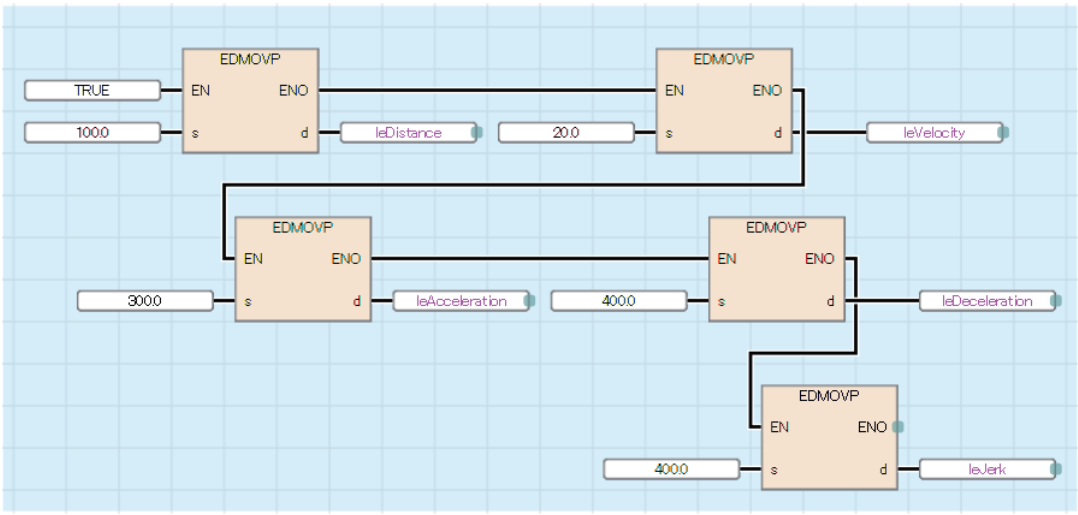
Axis No.	Label name	Data type	Comment
1	VirtualAxis0001	AXIS_REF	Axis 1 (Virtual drive axis)
2	VirtualAxis0002	AXIS_REF	Axis 2 (Virtual drive axis)
3	Axis0003	AXIS_REF	Axis 3 (Real drive axis)

■Labels used

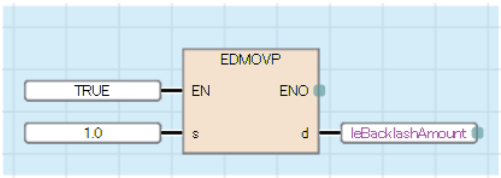
Label name	Data type	Comment
MC_MoveRelative_1	MC_MoveRelative	Relative Value Positioning FB
bMoveRelative	Bit	Single axis relative positioning start
leDistance	Double-precision real number	Movement amount
leVelocity	Double-precision real number	Velocity
leAcceleration	Double-precision real number	Acceleration
leDeceleration	Double-precision real number	Deceleration
leJerk	Double-precision real number	Jerk
bMoveRelDone	Bit	Execution completion
bMoveRelBusy	Bit	Executing
bMoveRelActive	Bit	Controlling
bMoveRelCommandAborted	Bit	Abortion of execution
bMoveRelError	Bit	Error
uwMoveRelErrorID	Word [unsigned]/bit string [16-bit]	Error code
MCv_BacklashCompensationFilter_1	MCv_BacklashCompensationFilter	Backlash Compensation Filter FB
bBacklashComFilter	Bit	Backlash compensation filter command
leBacklashAmount	Double-precision real number	Backlash amount
bBacklashValid	Bit	Output value valid
bBacklashBusy	Bit	Executing
bBacklashActive	Bit	Controlling
bBacklashCommandAborted	Bit	Abortion of execution
bBacklashError	Bit	Error
uwBacklashErrorID	Word [unsigned]/bit string [16-bit]	Error code
leBacklashFilterPool	Double-precision real number	Filter cumulated value
MCv_SpeedLimitFilter_1	MCv_SpeedLimitFilter	Speed Limit Filter
bSpeedLimitFilter	Bit	Speed limit filter command
lePositiveLimit	Double-precision real number	Positive direction limit value
leNegativeLimit	Double-precision real number	Negative direction limit value
bSpeedLimitValid	Bit	Output value valid
bSpeedLimitBusy	Bit	Executing
bSpeedLimitActive	Bit	Controlling
bSpeedLimitCommandAborted	Bit	Abortion of execution
bSpeedLimitError	Bit	Error
uwSpeedLimitErrorID	Word [unsigned]/bit string [16-bit]	Error code
leFilterPool	Double-precision real number	Filter cumulated value

■FBD/LD program (CPU module side)

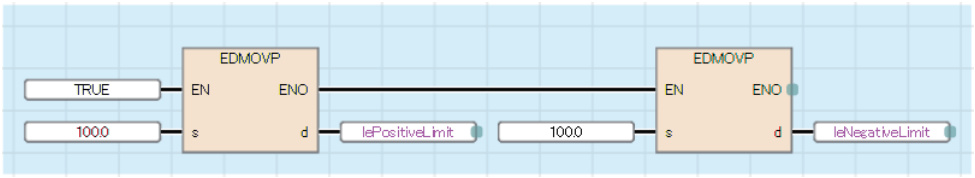
• Data settings for single axis positioning



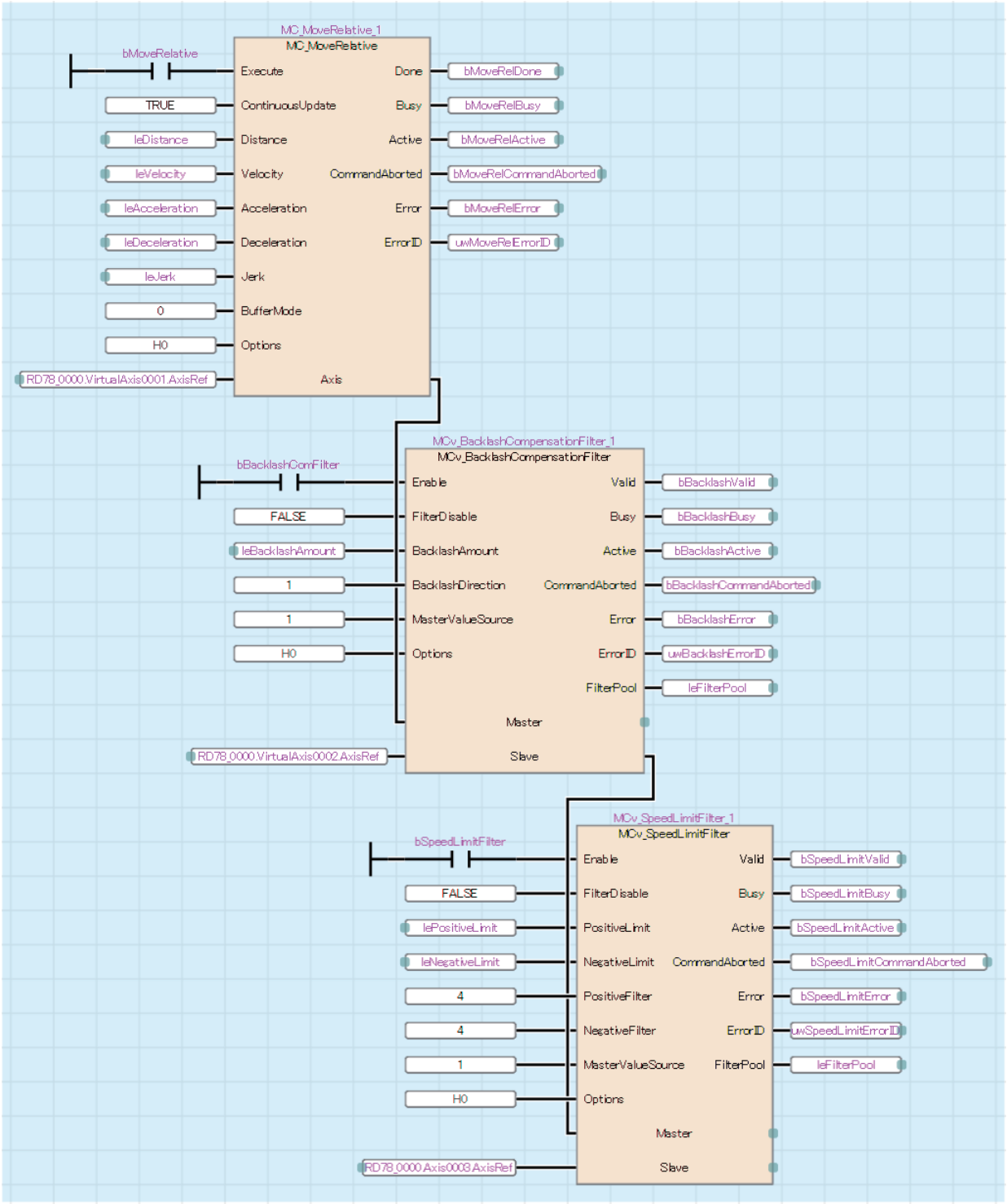
• Backlash amount



• Positive direction/negative direction limit value



• Relative Value Positioning/Backlash Compensation Filter/Speed Limit Filter



■ST program (Motion module side)

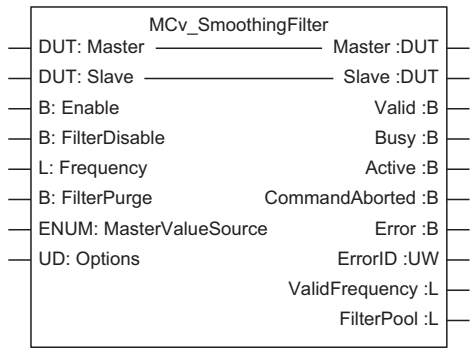
```
//-----Data settings for single axis positioning-----
leDistance:= 100.0;
leVelocity:= 20.0;
leAcceleration:= 300.0;
leDeceleration:= 400.0;
leJerk:= 0.0;

//-----Relative Value Positioning-----
MC_MoveRelative_1(
   Axis:= VirtualAxis0001.AxisRef ,
   Execute:= bMoveRelative ,
   ContinuousUpdate:= TRUE ,
   Distance:= leDistance ,
   Velocity:= leVelocity ,
   Acceleration:= leAcceleration ,
   Deceleration:= leDeceleration ,
   Jerk:= leJerk ,
   BufferMode:= MC_BUFFER_MODE__mcAborting ,
   Options:= H00000000 ,
   Done=> bMoveRelDone ,
   Busy=> bMoveRelBusy ,
   Active=> bMoveRelActive ,
   CommandAborted=> bMoveRelCommandAborted ,
   Error=> bMoveRelError ,
   ErrorID=> uwMoveRelErrorID
);

//-----Backlash amount-----
leBacklashAmount:= 1.0;

//-----Backlash Compensation Filter-----
MCv_BacklashCompensationFilter_1(
   Master:= MC_MoveRelative_1.Axis ,
   Slave:= VirtualAxis0002.AxisRef ,
   Enable:= bBacklashComFilter ,
   FilterDisable:= FALSE ,
   BacklashAmount:= leBacklashAmount ,
   BacklashDirection:= MC_DIRECTION__mcPositiveDirection ,
   MasterValueSource:= MC_SOURCE__mcSetValue ,
   Options:= H00000000 ,
   Valid=> bBacklashValid ,
   Busy=> bBacklashBusy ,
   Active=> bBacklashActive ,
   CommandAborted=> bBacklashCommandAborted ,
   Error=> bBacklashError ,
   ErrorID=> uwBacklashErrorID ,
   FilterPool=> leBacklashFilterPool
);

//-----Positive direction/negative direction limit value-----
lePositiveLimit:= 100.0;
```

```
leNegativeLimit:= 100.0;

//-----Speed Limit Filter-----
MCv_SpeedLimitFilter_1(
    Master:= MCv_BacklashCompensationFilter_1.Slave ,
    Slave:= Axis0003.AxisRef ,
    Enable:= bSpeedLimitFilter ,
    FilterDisable:= FALSE ,
    PositiveLimit:= lePositiveLimit ,
    NegativeLimit:= leNegativeLimit ,
    PositiveFilter:= MC_VELOCITY_LIMIT_MODE__ClampWithoutRamp ,
    NegativeFilter:= MC_VELOCITY_LIMIT_MODE__ClampWithoutRamp ,
    MasterValueSource:= MC_SOURCE__mcSetValue ,
    Options:= H00000000 ,
    Valid=> bSpeedLimitValid ,
    Busy=> bSpeedLimitBusy ,
    Active=> bSpeedLimitActive ,
    CommandAborted=> bSpeedLimitCommandAborted ,
    Error=> bSpeedLimitError ,
    ErrorID=> uwSpeedLimitErrorID ,
    FilterPool=> leFilterPool
);
```

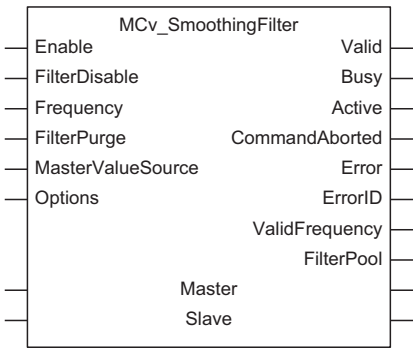
MCv_SmoothingFilter

Executes filter processing according to the specified frequency.

Ladder



FBD/LD



ST

```
MCv_SmoothingFilter(
  Master:= ?AXIS_REF? ,
  Slave:= ?AXIS_REF? ,
  Enable:= ?BOOL? ,
  FilterDisable:= ?BOOL? ,
  Frequency:= ?LREAL? ,
  FilterPurge:= ?BOOL? ,
  MasterValueSource:= ?INT? ,
  Options:= ?DWORD? ,
  Valid=> ?BOOL? ,
  Busy=> ?BOOL? ,
  Active=> ?BOOL? ,
  CommandAborted=> ?BOOL? ,
  Error=> ?BOOL? ,
  ErrorID=> ?WORD? ,
  ValidFrequency=> ?LREAL? ,
  FilterPool=> ?LREAL?
);
```

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Smoothing Filter	40	24	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
Master	Master axis	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)
Slave	Slave axis	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Enable	Enable	BOOL	Always	TRUE, FALSE	FALSE	When this variable is TRUE, executes MCv_SmoothingFilter (Smoothing Filter).
FilterDisable	Filter disable	BOOL	Always	TRUE, FALSE	FALSE	This variable sets the method used for the filter processing. • FALSE: Operates according to the filter setting. • TRUE: Disables the filter and transmits the Master axis (Master) input to Slave axis (Slave) as it is. For details, refer to the following. ☞ Page 385 Filter disable (FilterDisable)
Frequency	Frequency	LREAL	Always	0.20 to 250.00[Hz]	10.00	This variable sets the frequency to suppress vibration. For details, refer to the following. ☞ Page 385 Frequency (Frequency)
FilterPurge	Filter purge	BOOL	At start	TRUE, FALSE	FALSE	This variable sets the end operation with the filter cumulative value. • FALSE: Immediately stops the slave axis. • TRUE: The axis operates until reaching "0" and then stops. For details, refer to the following. ☞ Page 386 Filter purge (FilterPurge)

Input variable	Name	Data type	Input	Setting range	Default value	Description
MasterValueSource	Master axis data source selection	INT (MC_SOURCE)	At start	1, 2, 101, 102	1	This variable sets the data source of Master axis (Master). • 1: Set value (mcSetValue) • 2: Actual value (mcActualValue) • 101: Latest set value (mcLatestSetValue) • 102: Latest actual value (mcLatestActualValue) For details, refer to the following. ☞ Page 387 Master axis data source selection (MasterValueSource)
Options	Options	DWORD(HEX)	At start	00000000H	00000000H	Set this variable to "00000000H". *: When a value other than "00000000H" is set, "Out of Options Range (error code: 1A4EH)" occurs.

■Output variables

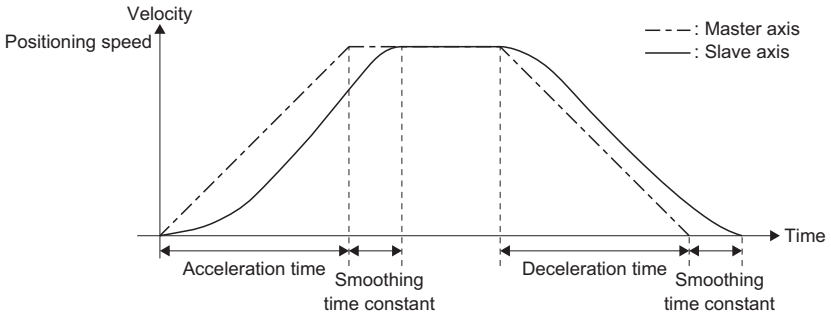
Output variable	Name	Data type	Default value	Description
Valid	Output value valid	BOOL	FALSE	This variable becomes TRUE when the output value is valid.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MCv_SmoothingFilter (Smoothing Filter) is executed.
Active	Controlling	BOOL	FALSE	This variable becomes TRUE when MCv_SmoothingFilter (Smoothing Filter) is controlling the axis.
CommandAborted	Abortion of execution	BOOL	FALSE	This variable becomes TRUE when execution of MCv_SmoothingFilter (Smoothing Filter) is aborted.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. 📖MELSEC iQ-R Motion Module User's Manual (Application)
ValidFrequency	Valid frequency	LREAL	0.00	This variable outputs the valid frequencies.
FilterPool	Filter cumulated value	LREAL	0.0	This variable outputs the slave axis movement amount cumulated in filter processing. Filter cumulated value (FilterPool) is cleared when Enable (Enable) is set to TRUE. If the value in Filter cumulated value (FilterPool) exceeds the positioning range, "Filter Cumulated Value Over Warning (warning code: 0D13H)" occurs. When this warning is detected, the error of Filter cumulated value (FilterPool) may increase.

■Public variables

Public variable	Name	Data type	Default value	Description
PurgedFilterPool	Disposed filter cumulated value	LREAL	0.0	When Filter disable (FilterDisable) is set to TRUE, this variable stores the disposed value from Filter cumulated value (FilterPool). Disposed filter cumulated value (PurgedFilterPool) does not cumulate. This variable is 0 cleared at FB start.

Processing details

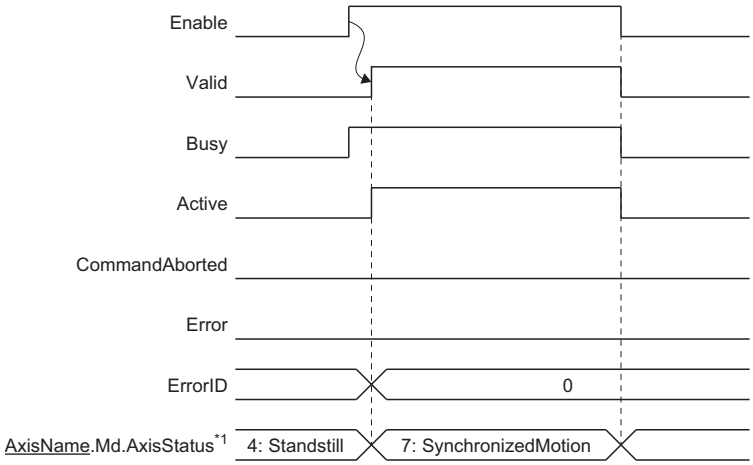
- Smoothing filter removes frequencies higher than the set Frequency (Frequency), and all waveforms higher than the setting value becomes smooth acceleration/deceleration waveforms.
- Smoothing filter outputs the result of filter processing executed for the Master axis (Master) value set in Master axis data source selection (MasterValueSource) to Slave axis (Slave).
- Axis status (AxisName.Md.AxisStatus) of Slave axis (Slave) is "7: During synchronous operation (SynchronizedMotion)" during smoothing filter execution.
- The time constant of smoothing is "1/ Frequency (Frequency)[s]", and the acceleration time and deceleration time become longer by the amount of the smoothing time constant. When Frequency (Frequency) is 10[Hz], the smoothing time constant is "0.1[s] = 100[ms]".



- If a smoothing filter is started while Slave axis (Slave) is in another operation FB, the start request will be ignored, and "Start Not Possible (error code: 1AADH)" occurs. Start the smoothing filter when Axis status (AxisName.Md.AxisStatus) of Slave axis (Slave) is "4: Standby (Standstill)".
- MC_Stop (Forced Stop) is the only function block that can be multiple started during smoothing filter execution.

■Timing chart

- When the operation completes normally



*1 The axis status of Slave axis (Slave).

- When an error occurs

  For the timing chart when an error occurs, refer to the following.

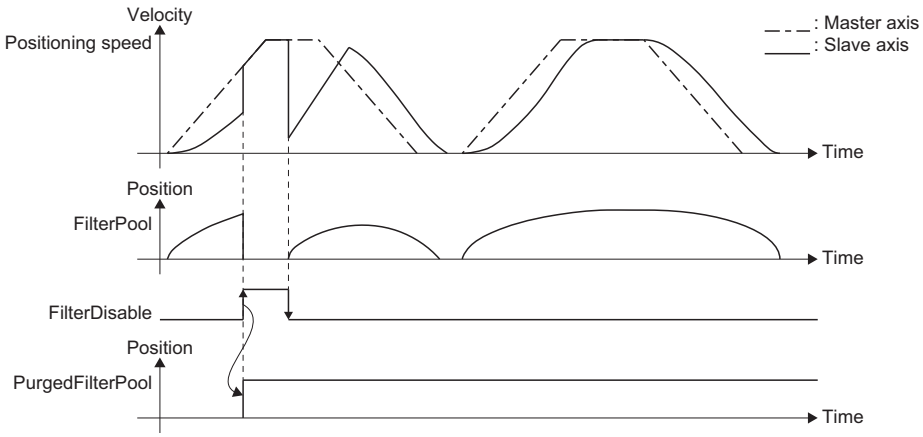
  ☞ Page 13 Basic operation of Enable (Enable) type Motion control FBs

■Filter disable (FilterDisable)

Sets the method used for the filter processing.

Setting value	Description
FALSE	Operates according to the filter setting.
TRUE	Disables the filter and transmits the Master axis (Master) input to Slave axis (Slave) as it is.

- When Filter disable (FilterDisable) is set to TRUE, the value in Filter cumulated value (FilterPool) becomes "0.0" and the disposed value from Filter cumulated value (FilterPool) is stored in Disposed filter cumulated value (PurgedFilterPool).



■Frequency (Frequency)

Sets the frequency to suppress vibration.

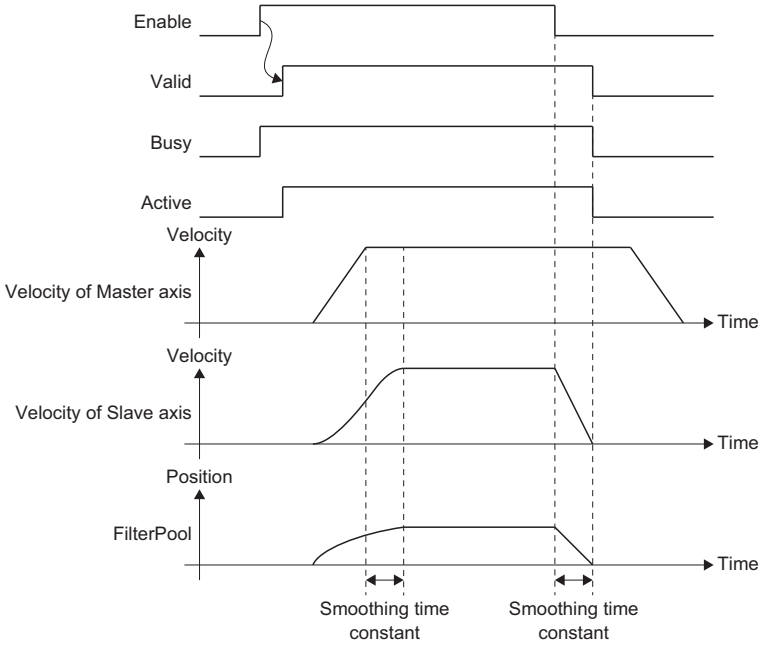
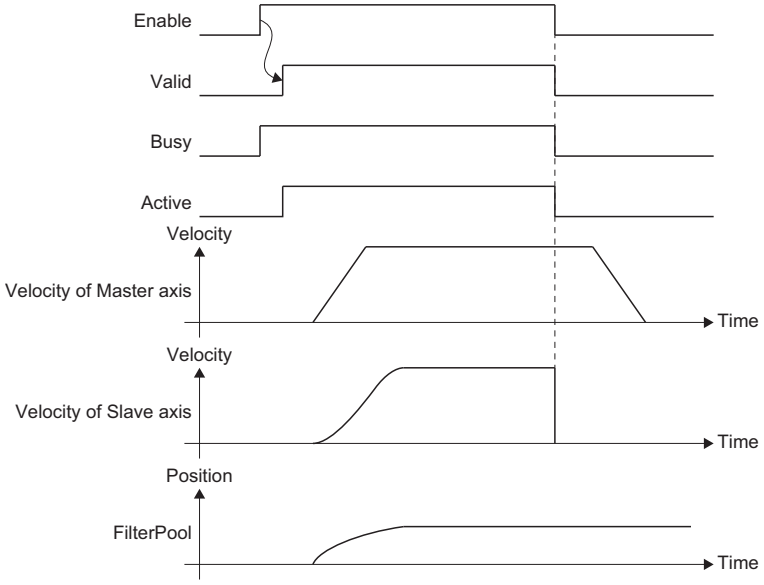
The following shows the settable range for Frequency (Frequency).

1/Slave axis operation cycle[s]	Setting range
250.00[Hz] or more	0.20 to 250.00[Hz]
Less than 250.00[Hz]	0.20 to (1/Slave axis operation cycle[s])[Hz]

- If a value outside the range is set in Frequency (Frequency), "Out of Frequency Specification Range (error code: 1A85H)" will occur.
- Valid frequencies can be checked in Valid frequency (ValidFrequency).
- To change the frequency, change Frequency (Frequency) when Filter cumulated value (FilterPool) is "0.0". Even if it is changed in a status other than "0.0", the change will not reflected until Filter cumulated value (FilterPool) becomes "0".
- Frequency (Frequency) cannot be changed to a frequency lower than Valid frequency (ValidFrequency). If a lower frequency is set, "Out of Frequency Specification Range (error code: 1A85H)" occurs.
- Filter cumulated value (FilterPool) may become "0.0" while the master axis operates with the operation pattern which repeats positive/reverse rotation. When the value of Frequency (Frequency) is changed while the filter operation is not settled, the filter operation will be stopped in the middle. Thus, it causes the deviation between the master axis and the slave axis. To change Frequency (Frequency), confirm the master axis stop and wait for the filter time constant before changing.

■Filter purge (FilterPurge)

Sets the operation for when Enable (Enable) is set to FALSE and Filter cumulated value (FilterPool) is other than "0.0".

Setting value	Description
TRUE	When Enable (Enable) becomes FALSE, the axis operates until Filter cumulated value (FilterPool) reaches "0.0" and then stops. 
FALSE	When Enable (Enable) becomes FALSE, immediately stops the slave axis regardless of the Filter cumulated value (FilterPool) status. 

■Master axis data source selection (MasterValueSource)

Slave axis (Slave) sets the position type of Master axis (Master) that will execute single axis synchronous control.
The operation may differ depending on the execution order of the FBs or if the operation cycles of Master axis (Master) and Slave axis (Slave) are different.
Operates with the same value as the set value when specifying Actual value in Master axis data source selection (MasterValueSource) while an axis type other than real axis is set in Master axis (Master).

Setting value	Description
1: Set value (mcSetValue)	Uses the commanded position of the master axis in the previous operation cycle.
2: Actual value (mcActualValue)	Uses the actual position of the master axis in the previous operation cycle.
101: Latest set value (mcLatestSetValue)	Uses the commanded position of the master axis in the current operation cycle.
102: Latest actual value (mcLatestActualValue)	Uses the actual position of the master axis in the current operation cycle.

Point

When Master axis data source selection (MasterValueSource) is set to "1: Set value (mcSetValue)" or "101: Latest set value (mcLatestSetValue)" and the master axis becomes servo OFF by a servo alarm or forced stop, the amount of the value change may be too large. This can be prevented by setting Master axis data source selection (MasterValueSource) to "2: Actual value (mcActualValue)" or "102: Latest actual value (mcLatestActualValue)".

Precautions

- For the smoothing filter, the operation load increases as the filter level increases. A rough standard for the operation load when the filter level is 5000 is as follows.

Motion module	Operation load
RD78G	Approximately 380μs
RD78GH	Approximately 240μs

- The filter level is calculated with the following formula. Digits after the decimal point are rounded/up down to the nearest integer. When the result of rounding is "0", the operation is carried out as 1 level.
  Filter level = (1 / frequency[Hz]) / operation cycle[s]
- The smoothing filter secures the memory to use for filter processing at start. The memory usage can be calculated with the following formula. If memory is insufficient, change Maximum RAM size (System.PrConst.Addon_MotionControl_AxisFilter.RamSizeMax) of Addon MotionControl_AxisFilter parameter (Addon_MotionControl_AxisFilter).
  Memory usage = Filter level × 8 [Bytes]
- The operation error may increase when the movement amount of the master axis or the slave axis is large and the following case applies. Confirm the unit setting of the master axis and the slave axis and decrease the movement range. If the error increases, clear the synchronization status or execute the current value change of absolute position specification.
  - When the total sum of the movement amount (the following formula) equivalent to filter level at control exceeds the number of significant digits which can be represented by double-precision floating-point (approximately 15 digits).

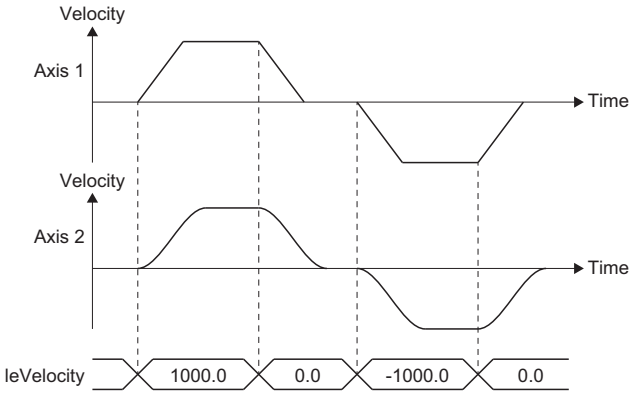
$$\sum_{i=0}^{\text{Filter level - 1}}$$ (i the position of Master axis before operation cycle - the position of Master axis at starting control)

- When the master axis operates in the positive and negative direction alternately every operation cycle, underflow occurs during internal operation and the position error of the master axis and the slave axis may occur.
- Note that if Filter disable (FilterDisable) is switched during operation, the Slave axis (Slave) velocity may change rapidly.
- After enabling the command filter, when the Master axis (Master) command is not transmitted to Slave axis (Slave) or the back part synchronous control FB because the stop factor occurred on Slave axis (Slave), the synchronous position relation deviates. Before enabling the command filter again, execute the synchronous positioning as needed.

Program example

In the following program example, filter processing is performed to change the trapezoidal acceleration/deceleration to a smooth acceleration/deceleration waveform.

- Operation



- Settings

Item		Setting value
Frequency		10.00
Data settings for speed control	Velocity	5000.0
	Acceleration	100.0
	Deceleration	100.0
	Jerk	0.0

■Axes

Axis No.	Label name	Data type	Comment
1	Axis0001	AXIS_REF	Axis 1
2	Axis0002	AXIS_REF	Axis 2

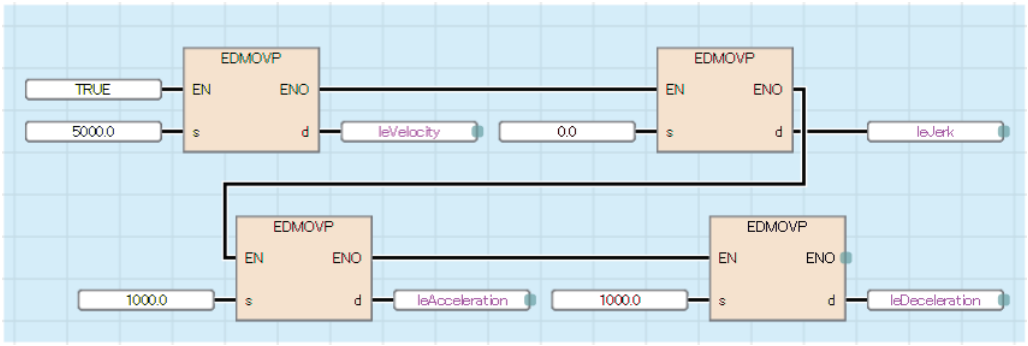
■Labels used

Label name	Data type	Comment
MCv_SpeedControl_1	MCv_SpeedControl	Speed Control (Including Position Loop) FB
bSpeedControl	Bit	Speed control command
leVelocity	Double-precision real number	Commanded velocity
leAcceleration	Double-precision real number	Acceleration
leDeceleration	Double-precision real number	Deceleration
leJerk	Double-precision real number	Jerk
bSpeedInVelocity	Bit	Target velocity reached
bSpeedBusy	Bit	Executing
bSpeedActive	Bit	Controlling
bSpeedCommandAborted	Bit	Abortion of execution
bSpeedError	Bit	Error
uwSpeedErrorID	Word [unsigned]/bit string [16-bit]	Error code
MCv_SmoothingFilter_1	MCv_SmoothingFilter	Smoothing Filter FB
bSmoothingFilter	Bit	Smoothing filter command
leFrequency	Double-precision real number	Frequency
bSmoothingValid	Bit	Output value valid
bSmoothingInVelocity	Bit	Target velocity reached
bSmoothingBusy	Bit	Executing
bSmoothingActive	Bit	Controlling
bSmoothingCommandAborted	Bit	Abortion of execution
bSmoothingError	Bit	Error
uwSmoothingErrorID	Word [unsigned]/bit string [16-bit]	Error code

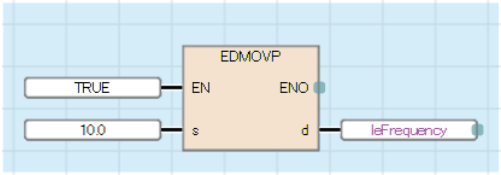
Label name	Data type	Comment
leSmoothingValidFrequency	Double-precision real number	Valid frequency
leSmoothingFilterPool	Double-precision real number	Filter cumulated value

■FBD/LD program (CPU module side)

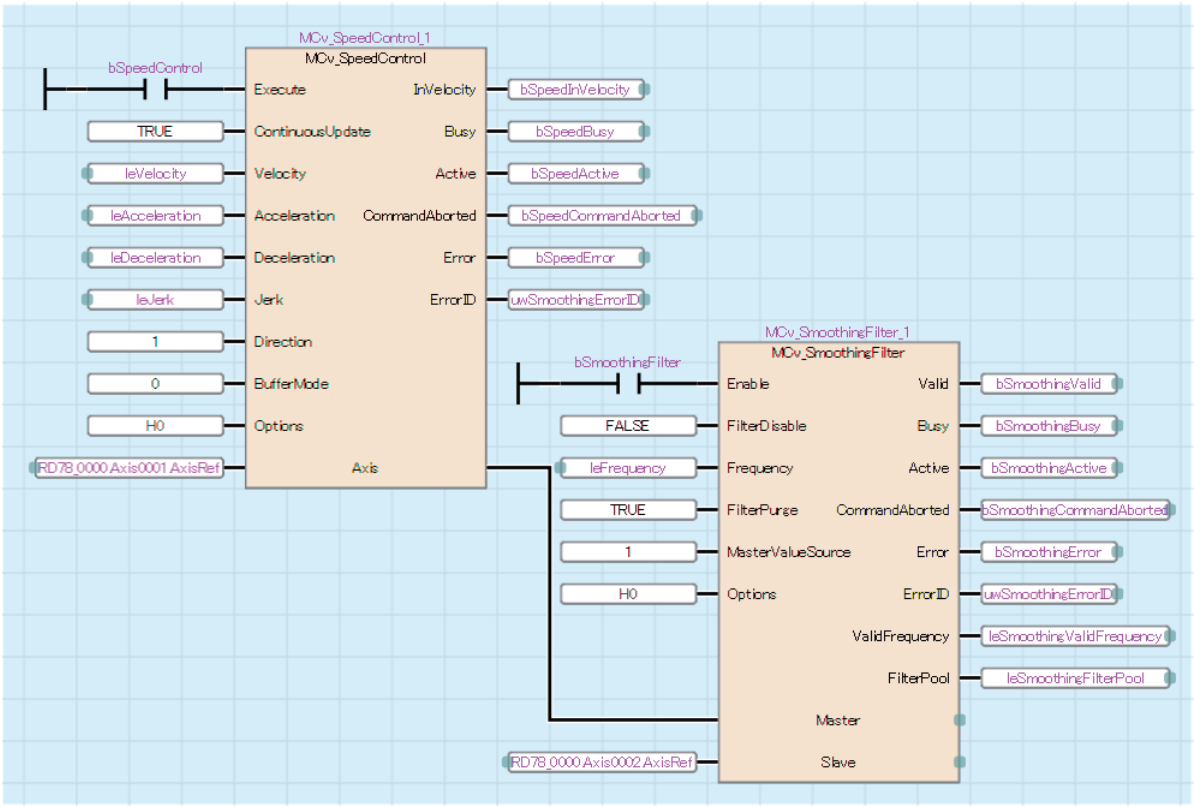
- Data settings for speed control



- Frequency



- Speed Control/Smoothing Filter

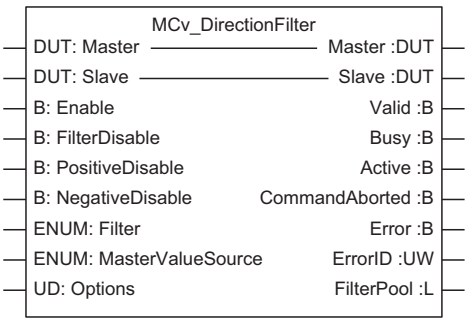
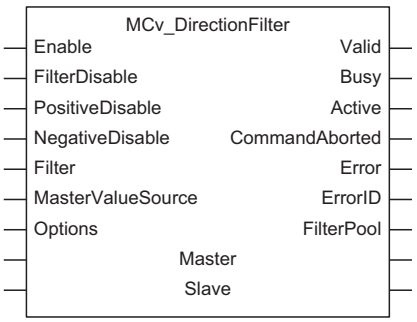


■ST program (Motion module side)

```
//-----Data settings for speed control-----
leVelocity:= 5000.0;
leAcceleration:= 1000.0;
leDeceleration:= 1000.0;
leJerk:= 0.0;

//-----Speed Control-----
MCv_SpeedControl_1(
   Axis:= Axis0001.AxisRef ,
   Execute:= bSpeedControl ,
   ContinuousUpdate:= TRUE ,
   Velocity:= leVelocity ,
   Acceleration:= leAcceleration ,
   Deceleration:=leDeceleration ,
   Jerk:= leJerk ,
   Direction:= MC_DIRECTION__mcPositiveDirection ,
   BufferMode:= MC_BUFFER_MODE__mcAborting ,
   Options:= H00000000 ,
   InVelocity=> bSpeedInVelocity ,
   Busy=> bSpeedBusy ,
   Active=> bSpeedActive ,
   CommandAborted=> bSpeedCommandAborted ,
   Error=> bSpeedError ,
   ErrorID=> uwSpeedErrorID
);

//-----Frequency-----
leFrequency:= 10.00;

//-----Smoothing Filter-----
MCv_SmoothingFilter_1(
   Master:= MCv_SpeedControl_1.Axis ,
   Slave:= Axis0002.AxisRef ,
   Enable:= bSmoothingFilter ,
   FilterDisable:= FALSE ,
   Frequency:= leFrequency ,
   FilterPurge:= TRUE ,
   MasterValueSource:= MC_SOURCE__mcSetValue ,
   Options:= H00000000 ,
   Valid=> bSmoothingValid ,
   Busy=> bSmoothingBusy ,
   Active=> bSmoothingActive ,
   CommandAborted=> bSmoothingCommandAborted ,
   Error=> bSmoothingError ,
   ErrorID=> uwSmoothingErrorID ,
   ValidFrequency=> leSmoothingValidFrequency ,
   FilterPool=> leSmoothingFilterPool
);
```

MCv_DirectionFilter

Performs filter processing to restrict traveling for the set movement direction.

Ladder	FBD/LD	ST
		MCv_DirectionFilter(Master:= ?AXIS_REF? , Slave:= ?AXIS_REF? , Enable:= ?BOOL? , FilterDisable:= ?BOOL? , PositiveDisable:= ?BOOL? , NegativeDisable:= ?BOOL? , Filter:= ?INT? , MasterValueSource:= ?INT? , Options:= ?DWORD? , Valid=> ?BOOL? , Busy=> ?BOOL? , Active=> ?BOOL? , CommandAborted=> ?BOOL? , Error=> ?BOOL? , ErrorID=> ?WORD? , FilterPool=> ?LREAL?);

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Moving Direction Restriction Filter	28	28	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
Master	Master axis	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)
Slave	Slave axis	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Enable	Enable	BOOL	Always	TRUE, FALSE	FALSE	When this variable is TRUE, executes MCv_DirectionFilter (Moving Direction Restriction Filter).
FilterDisable	Filter disable	BOOL	Always	TRUE, FALSE	FALSE	This variable sets the method used for the filter processing. • FALSE: Operates according to the filter setting. • TRUE: Disables the filter and transmits the Master axis (Master) input to Slave axis (Slave) as it is. For details, refer to the following. ☞ Page 394 Filter disable (FilterDisable)
PositiveDisable	Positive direction limit	BOOL	At start	TRUE, FALSE	FALSE	When this variable is set to TRUE, limits the axis operation to the positive direction. Slave axis (Slave) does not operate at Master axis (Master) forward rotation operation. When both Positive direction limit (PositiveDisable) and Negative direction limit (NegativeDisable) are set to TRUE, Slave axis (Slave) will not operate even if Master axis (Master) is operated. • FALSE: Not limited • TRUE: Limited

Input variable	Name	Data type	Input	Setting range	Default value	Description
NegativeDisable	Negative direction limit	BOOL	At start	TRUE, FALSE	FALSE	When this variable is set to TRUE, limits the axis operation to the negative direction. Slave axis (Slave) does not operate at Master axis (Master) reverse rotation operation. When both Positive direction limit (PositiveDisable) and Negative direction limit (NegativeDisable) are set to TRUE, Slave axis (Slave) will not operate even if Master axis (Master) is operated. • FALSE: Not limited • TRUE: Limited
Filter	Filter operation	INT (MC_VELOCITY_LIMIT_MODE)	At start	0, 2, 4	2	This variable sets the filter operation. • 0: Ignore (Ignore) • 2: Truncate (Truncate) • 4: Clamp (Without Ramp at Deceleration) (ClampWithoutRamp) For details, refer to the following. ☞ Page 394 Filter operation (Filter)
MasterValueSource	Master axis data source selection	INT (MC_SOURCE)	At start	1, 2, 101, 102	1	This variable sets the data source of the master axis (Master). • 1: Set value (mcSetValue) • 2: Actual value (mcActualValue) • 101: Latest set value (mcLatestSetValue) • 102: Latest actual value (mcLatestActualValue) For details, refer to the following. ☞ Page 395 Master axis data source selection (MasterValueSource)
Options	Options	DWORD(HEX)	At start	00000000H	00000000H	Set this variable to "00000000H". *: When a value other than "00000000H" is set, "Out of Options Range (error code: 1A4EH)" occurs.

■Output variables

Output variable	Name	Data type	Default value	Description
Valid	Output value valid	BOOL	FALSE	This variable becomes TRUE when the output value is valid.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MCv_DirectionFilter (Moving Direction Restriction Filter) is executed.
Active	Controlling	BOOL	FALSE	This variable becomes TRUE when MCv_DirectionFilter (Moving Direction Restriction Filter) is controlling the axis.
CommandAborted	Abortion of execution	BOOL	FALSE	This variable becomes TRUE when execution of MCv_DirectionFilter (Moving Direction Restriction Filter) is aborted.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. 📖MELSEC iQ-R Motion Module User's Manual (Application)
FilterPool	Filter cumulated value	LREAL	0.0	This variable outputs the slave axis movement amount cumulated in filter processing. When Enable (Enable) is set to FALSE while the status of Filter cumulated value (FilterPool) is other than "0.0", Filter cumulated value (FilterPool) retains the status from when it was set to FALSE. Filter cumulated value (FilterPool) is cleared the next time that Enable (Enable) is set to TRUE. If the value in Filter cumulated value (FilterPool) exceeds the positioning range, "Filter Cumulated Value Over Warning (warning code: 0D13H)" occurs. When this warning is detected, the error of Filter cumulated value (FilterPool) may increase.

■Public variables

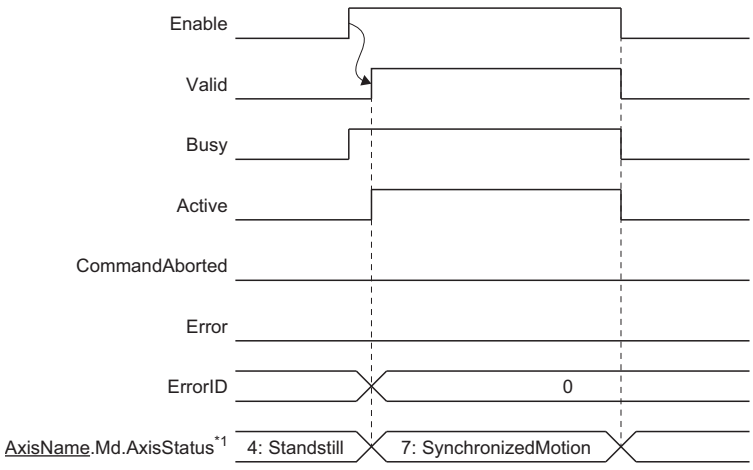
Public variable	Name	Data type	Default value	Description
PurgedFilterPool	Disposed filter cumulated value	LREAL	0.0	When Filter disable (FilterDisable) is set to TRUE, this variable stores the disposed value from Filter cumulated value (FilterPool). Disposed filter cumulated value (PurgedFilterPool) does not cumulate. This variable is 0 cleared at FB start.

Processing details

- Moving direction limit filter restricts the movement of the slave axis to one direction in relation to the master axis movement.
- Moving direction limit filter takes the result of filter processing executed for the Master axis (Master) value set in Master axis data source selection (MasterValueSource) and outputs it to Slave axis (Slave).
- Axis status (AxisName.Md.AxisStatus) of Slave axis (Slave) is "7: During synchronous operation (SynchronizedMotion)" during moving direction filter execution.
- If a moving direction limit filter is started while Slave axis (Slave) is in another operation FB, the start request will be ignored, and "Start Not Possible (error code: 1AADH)" occurs. Start the moving direction limit filter when Axis status (AxisName.Md.AxisStatus) of Slave axis (Slave) is "4: Standby (Standstill)".
- MC_Stop (Forced Stop) is the only function block that can be multiple started during smoothing filter execution.
- When setting Enable (Enable) to FALSE, perform this function block while the slave axis is in stop status. If Enable (Enable) is set to FALSE while the slave axis is in operation, the slave axis will stop immediately.

■Timing chart

- When the operation completes normally



*1 The axis status of Slave axis (Slave).

- When an error occurs

  For the timing chart when an error occurs, refer to the following.

  ☞ Page 13 Basic operation of Enable (Enable) type Motion control FBs

■Filter disable (FilterDisable)

Sets the method used for the filter processing.

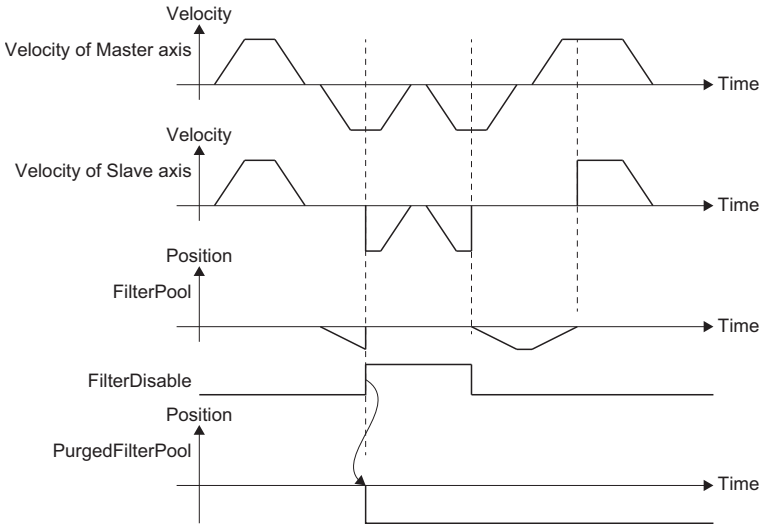
Setting value	Description
FALSE	Operates according to the filter setting.
TRUE	Disables the filter and transmits the Master axis (Master) input to Slave axis (Slave) as it is.

- When Filter disable (FilterDisable) is set to TRUE, the value in Filter cumulated value (FilterPool) becomes "0.0" and the disposed value from Filter cumulated value (FilterPool) is stored in Disposed filter cumulated value (PurgedFilterPool).

Ex.

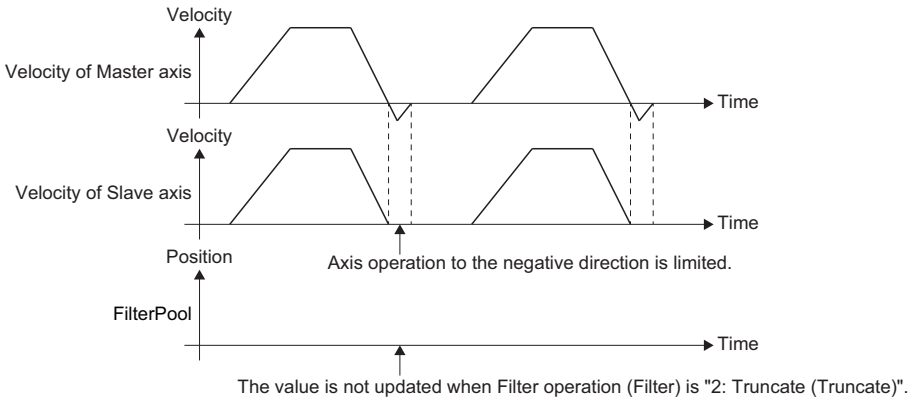
When set as shown below

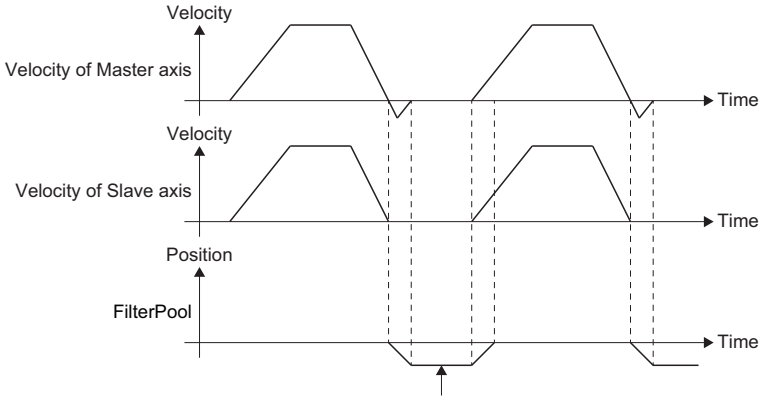
Input pin	Setting value
Filter operation (Filter)	4: Clamp (Without Ramp at Deceleration) (ClampWithoutRamp)
Positive direction limit (PositiveDisable)	FALSE (Not limited)
Negative direction limit (NegativeDisable)	TRUE (Limited)



■Filter operation (Filter)

- Sets the slave axis operation at control based on the setting of Filter operation (Filter).

Setting value	Description
0: Ignore (Ignore)	• The movement direction is not restricted. • Filter cumulated value (FilterPool) is not updated.
2: Truncate (Truncate)	• The movement amount to the restricted direction is ignored and is not added to Filter cumulated value (FilterPool). • The movement amount to the restricted direction will be truncated, and the position relation between the master axis and the slave axis will deviate by an amount equal to the movement amount to the restricted direction. <Example> When Positive direction limit (PositiveDisable) is FALSE, and Negative direction limit (NegativeDisable) is TRUE 

Setting value	Description
4: Clamp (Without Ramp at Deceleration) (ClampWithoutRamp)	• The movement amount to the restricted direction is added to Filter cumulated value (FilterPool). • Filter cumulated value (FilterPool) reflects the reverse direction movement. Thus, even if operation to the restricted direction is repeated, the position relation between the master axis and the slave axis will not deviate from that when the status of Filter cumulated value (FilterPool) is "0.0". <Example> When Positive direction limit (PositiveDisable) is FALSE (Not limited), and Negative direction limit (NegativeDisable) is TRUE (Limited)  The input travel value is accumulated as a filter cumulated value, and will be reflected when the input movement amount moves in the permitted direction.

- When setting a value outside the range to Filter operation (Filter), "Out of Filter Operation Specification Range (error code: 1A86H)" occurs.

■Master axis data source selection (MasterValueSource)

Slave axis (Slave) sets the position type of Master axis (Master) that will execute single axis synchronous control.

The operation may differ depending on the execution order of the FBs or if the operation cycles of Master axis (Master) and Slave axis (Slave) are different.

Operates with the same value as the set value when specifying Actual value in Master axis data source selection (MasterValueSource) while an axis type other than real axis is set in Master axis (Master).

Setting value	Description
1: Set value (mcSetValue)	Uses the commanded position of the master axis in the previous operation cycle.
2: Actual value (mcActualValue)	Uses the actual position of the master axis in the previous operation cycle.
101: Latest set value (mcLatestSetValue)	Uses the commanded position of the master axis in the current operation cycle.
102: Latest actual value (mcLatestActualValue)	Uses the actual position of the master axis in the current operation cycle.

Point

When Master axis data source selection (MasterValueSource) is set to "1: Set value (mcSetValue)" or "101: Latest set value (mcLatestSetValue)" and the master axis becomes servo OFF by a servo alarm or forced stop, the amount of the value change may be too large. This can be prevented by setting Master axis data source selection (MasterValueSource) to "2: Actual value (mcActualValue)" or "102: Latest actual value (mcLatestActualValue)".

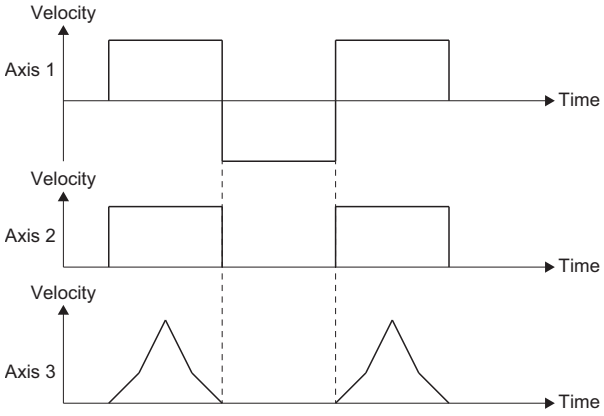
Precautions

- Note that if Filter disable (FilterDisable) is switched during operation, the Slave axis (Slave) velocity may change rapidly.
- After enabling the command filter, when the Master axis (Master) command is not transmitted to Slave axis (Slave) or the back part synchronous control FB because the stop factor occurred on Slave axis (Slave), the synchronous position relation deviates. Before enabling the command filter again, execute the synchronous positioning as needed.

Program example

In the following program example, filter processing is performed to restrict the direction of the cam operation.

- Operation



- Settings

Item	Setting value
Cam ID setting	1

■Axes

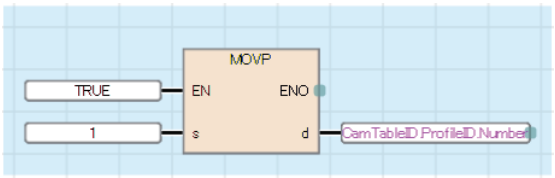
Axis No.	Label name	Data type	Comment
1	Axis0001	AXIS_REF	Axis 1
2	Axis0002	AXIS_REF	Axis 2
3	Axis0003	AXIS_REF	Axis 3

■Labels used

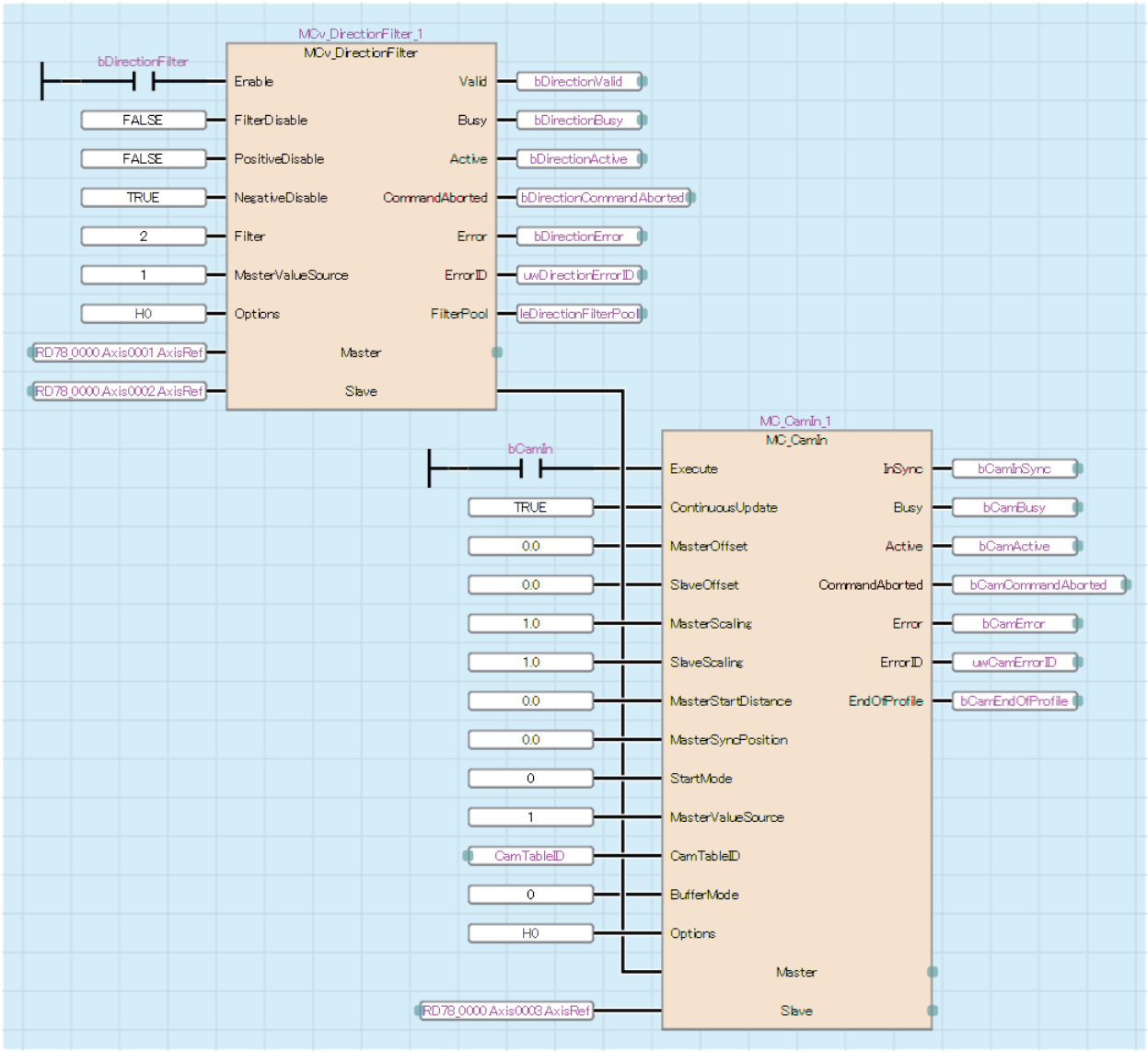
Label name	Data type	Comment
MCv_DirectionFilter_1	MCv_DirectionFilter	Moving Direction Restriction Filter FB
bDirectionFilter	Bit	Moving direction restriction filter command
bDirectionValid	Bit	Output value valid
bDirectionBusy	Bit	Executing
bDirectionActive	Bit	Controlling
bDirectionCommandAborted	Bit	Abortion of execution
bDirectionError	Bit	Error
uwDirectionErrorID	Word [unsigned]/bit string [16-bit]	Error code
leDirectionFilterPool	Double-precision real number	Disposed filter cumulated value
MC_CamIn_1	MC_CamIn	Cam Operation Start FB
bCamIn	Bit	Cam operation command
CamTableID	MC_CAM_ID	Cam ID
bCamInSync	Bit	In synchronization
bCamBusy	Bit	Executing
bCamActive	Bit	Controlling
bCamCommandAborted	Bit	Abortion of execution
bCamError	Bit	Error
uwCamErrorID	Word [unsigned]/bit string [16-bit]	Error code
bCamEndOfProfile	Bit	Cam cycle completion

■FBD/LD program (CPU module side)

• Cam ID settings



• Moving Direction Restriction Filter/cam operation



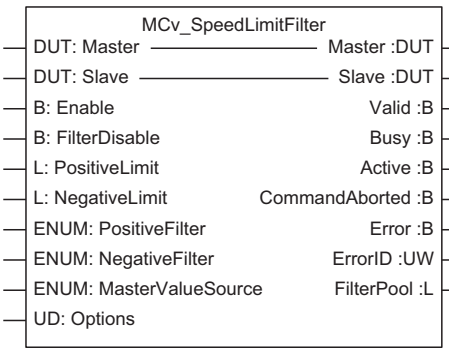
■ST program (Motion module side)

```
//-----Moving Direction Restriction Filter-----
MCv_DirectionFilter_1(
   Master:= Axis0001.AxisRef ,
   Slave:= Axis0002.AxisRef ,
   Enable:= bDirectionFilter ,
   FilterDisable:= FALSE ,
   PositiveDisable:= FALSE ,
   NegativeDisable:= TRUE ,
   Filter:= MC_VELOCITY_LIMIT_MODE__Truncate ,
   MasterValueSource:= MC_SOURCE__mcSetValue ,
   Options:= H00000000 ,
   Valid=> bDirectionValid ,
   Busy=> bDirectionBusy ,
   Active=> bDirectionActive ,
   CommandAborted=> bDirectionCommandAborted ,
   Error=> bDirectionError ,
   ErrorID=> uwDirectionErrorID ,
   FilterPool=> leDirectionFilterPool
);

//Cam ID settings
CamTableID.ProfileID.Number:= 1;

//Cam operation
MC_CamIn_1(
   Master:= MCv_DirectionFilter_1.Slave ,
   Slave:= Axis0003.AxisRef ,
   Execute:= bCamIn,
   ContinuousUpdate:= TRUE ,
   MasterOffset:= 0.0 ,
   SlaveOffset:= 0.0 ,
   MasterScaling:= 1.0 ,
   SlaveScaling:= 1.0 ,
   MasterStartDistance:= 0.0 ,
   MasterSyncPosition:= 0.0 ,
   StartMode:= MC_START_MODE__mcImmediate ,
   MasterValueSource:= MC_SOURCE__mcSetValue ,
   CamTableID:= CamTableID ,
   BufferMode:= MC_BUFFER_MODE__mcAborting ,
   Options:= H00000000 ,
   InSync=> bCamInSync ,
   Busy=> bCamBusy ,
   Active=> bCamActive ,
   CommandAborted=> bCamCommandAborted ,
   Error=> bCamError ,
   ErrorID=> uwCamErrorID ,
   EndOfProfile=> bCamEndOfProfile
);
```

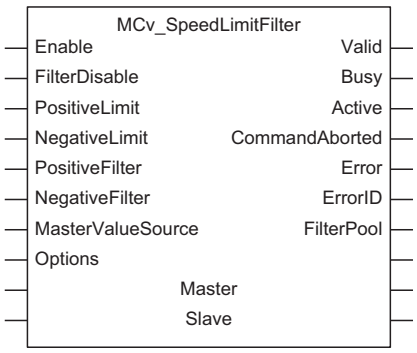
MCv_SpeedLimitFilter

Performs filter processing to restrict to the velocity of the set limit value.

Ladder



FBD/LD



ST

```
MCv_SpeedLimitFilter(
  Master:= ?AXIS_REF? ,
  Slave:= ?AXIS_REF? ,
  Enable:= ?BOOL? ,
  FilterDisable:= ?BOOL? ,
  PositiveLimit:= ?LREAL? ,
  NegativeLimit:= ?LREAL? ,
  PositiveFilter:= ?INT? ,
  NegativeFilter:= ?INT? ,
  MasterValueSource:= ?INT? ,
  Options:= ?DWORD? ,
  Valid=> ?BOOL? ,
  Busy=> ?BOOL? ,
  Active=> ?BOOL? ,
  CommandAborted=> ?BOOL? ,
  Error=> ?BOOL? ,
  ErrorID=> ?WORD? ,
  FilterPool=> ?LREAL?
);
```

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Speed Limit Filter	52	28	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
Master	Master axis	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)
Slave	Slave axis	AXIS_REF	At start	—	Omission not possible	This variable sets the axis. For the variables used (AxisName.AxisRef.), refer to the following. ☞ Page 23 AxisName.AxisRef. (Axis information)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Enable	Enable	BOOL	Always	TRUE, FALSE	FALSE	When this variable is TRUE, executes MCv_SpeedLimitFilter (Speed Limit Filter).
FilterDisable	Filter disable	BOOL	Always	TRUE, FALSE	FALSE	This variable sets the method used for the filter processing. • FALSE: Operates according to the filter setting. • TRUE: Disables the filter and transmits the Master axis (Master) input to Slave axis (Slave) as it is. For details, refer to the following. ☞ Page 402 Filter disable (FilterDisable)
PositiveLimit	Positive direction limit value	LREAL	At start	0.0, 0.0001 to 2500000000.0	0.0	This variable sets the speed limit value for the positive direction. Sets the upper limit value of the velocity output to Slave axis (Slave) in relation to the Master axis (Master) positive direction input velocity. When this variable is set to "0.0", Slave axis does not operate to the positive direction. When a value outside the range is set, "Out of Direction Limit Value Specification Range (error code: 1A87H)" occurs.

Input variable	Name	Data type	Input	Setting range	Default value	Description
NegativeLimit	Negative direction limit value	LREAL	At start	0.0, 0.0001 to 2500000000.0	0.0	This variable sets the speed limit value for the negative direction. Sets the upper limit value of the velocity output to Slave axis (Slave) in relation to the Master axis (Master) negative direction input velocity. When this variable is set to "0.0", Slave axis does not operate to the positive direction. When a value outside the range is set, "Out of Direction Limit Value Specification Range (error code: 1A87H)" occurs.
PositiveFilter	Positive direction filter operation	INT (MC_VELOCITY_LIMIT_MODE)	At start	0, 2, 4	2	This variable sets the filter operation for when the direction exceeds the positive direction limit. • 0: Ignore (Ignore) • 2: Truncate (Truncate) • 4: Clamp (Without Ramp at Deceleration) (ClampWithoutRamp) For details, refer to the following. ☞ Page 402 Positive direction filter operation (PositiveFilter)/Negative direction filter operation (NegativeFilter)
NegativeFilter	Negative direction filter operation	INT (MC_VELOCITY_LIMIT_MODE)	At start	0, 2, 4	2	This variable sets the filter operation for when the direction exceeds the negative direction limit. • 0: Ignore (Ignore) • 2: Truncate (Truncate) • 4: Clamp (Without Ramp at Deceleration) (ClampWithoutRamp) For details, refer to the following. ☞ Page 402 Positive direction filter operation (PositiveFilter)/Negative direction filter operation (NegativeFilter)
MasterValueSource	Master axis data source selection	INT (MC_SOURCE)	At start	1, 2, 101, 102	1	This variable sets the data source of Master axis (Master). • 1: Set value (mcSetValue) • 2: Actual value (mcActualValue) • 101: Latest set value (mcLatestSetValue) • 102: Latest actual value (mcLatestActualValue) For details, refer to the following. ☞ Page 403 Master axis data source selection (MasterValueSource)
Options	Options	DWORD(HEX)	At start	00000000H	00000000H	Set this variable to "00000000H". *: When a value other than "00000000H" is set, "Out of Options Range (error code: 1A4EH)" occurs.

■Output variables

Output variable	Name	Data type	Default value	Description
Valid	Output value valid	BOOL	FALSE	This variable becomes TRUE when the output value is valid.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MCv_SpeedLimitFilter (Speed Limit Filter) is executed.
Active	Controlling	BOOL	FALSE	Indicates that the FB is controlling the axis. This variable becomes TRUE when MCv_SpeedLimitFilter (Speed Limit Filter) is controlling the axis.
CommandAborted	Abortion of execution	BOOL	FALSE	This variable becomes TRUE when execution of MCv_SpeedLimitFilter (Speed Limit Filter) is aborted.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. MELSEC iQ-R Motion Module User's Manual (Application)

Output variable	Name	Data type	Default value	Description
FilterPool	Filter cumulated value	LREAL	0.0	This variable outputs the slave axis movement amount cumulated in filter processing. When Enable (Enable) is set to FALSE when the status of Filter cumulated value (FilterPool) is other than "0.0", Filter cumulated value (FilterPool) retains the status from when it was set to FALSE. Filter cumulated value (FilterPool) is cleared the next time that Enable (Enable) is set to TRUE. If the value in Filter cumulated value (FilterPool) exceeds the positioning range, "Filter Cumulated Value Over Warning (warning code: 0D13H)" occurs. When this warning is detected, the error of Filter cumulated value (FilterPool) may increase.

■Public variables

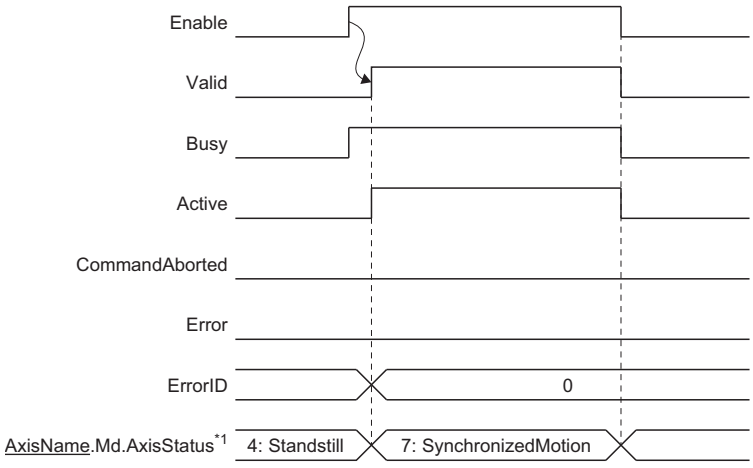
Public variable	Name	Data type	Default value	Description
PurgedFilterPool	Disposed filter cumulated value	LREAL	0.0	When Filter disable (FilterDisable) is set to TRUE, this variable stores the disposed value from Filter cumulated value (FilterPool). Disposed filter cumulated value (PurgedFilterPool) does not cumulate. This variable is 0 cleared at FB start.

Processing details

- Speed limit filter sets a specified control value to the input velocity of the master axis, and outputs the velocity of the set control value to the slave axis.
- Speed limit filter outputs the result of filter processing executed for the master axis (Master) value set in Master axis data source selection (MasterValueSource) to the slave axis (Slave).
- Axis status (AxisName.Md.AxisStatus) of Slave axis (Slave) is "7: During synchronous operation (SynchronizedMotion)" during speed limit filter execution.
- If a speed limit filter is started while Slave axis (Slave) is in another operation FB, the start request will be ignored, and "Start Not Possible (error code: 1AADH)" occurs. Start the speed limit filter when Axis status (AxisName.Md.AxisStatus) of Slave axis (Slave) is "4: Standby (Standstill)".
- MC_Stop (Forced Stop) is the only function block that can be multiple started during speed limit filter execution.
- When setting Enable (Enable) to FALSE, perform this function block while the slave axis is in stop status. If Enable (Enable) is set to FALSE while the slave axis is in operation, the slave axis will stop immediately.

■Timing chart

- When the operation completes normally



*1 The axis status of Slave axis (Slave).

• When an error occurs

For the timing chart when an error occurs, refer to the following.

☞ Page 13 Basic operation of Enable (Enable) type Motion control FBs

■Filter disable (FilterDisable)

Sets the method used for the filter processing.

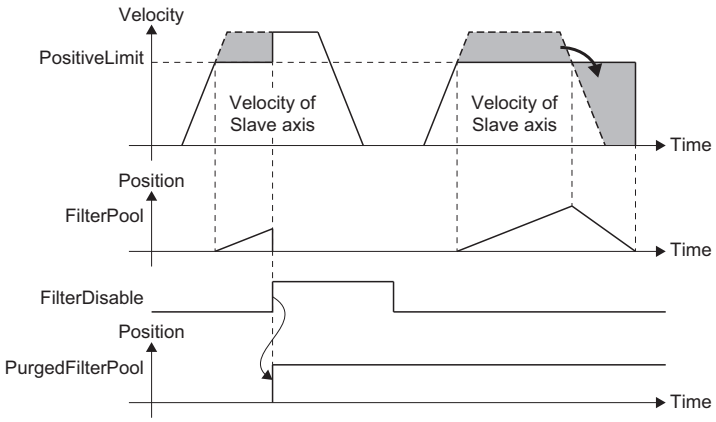
Setting value	Description
FALSE	Operates according to the filter setting.
TRUE	Disables the filter and transmits the Master axis (Master) input to Slave axis (Slave) as it is.

• When Filter disable (FilterDisable) is set to TRUE, the value in Filter cumulated value (FilterPool) becomes "0.0" and the disposed value from Filter cumulated value (FilterPool) is stored in Disposed filter cumulated value (PurgedFilterPool).

Ex.

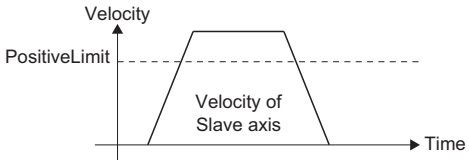
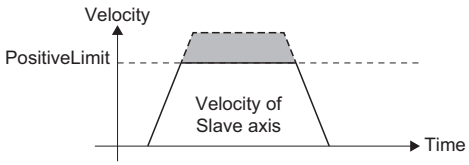
When set as shown below

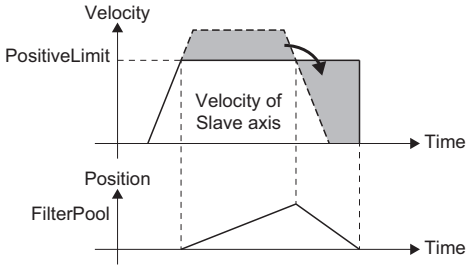
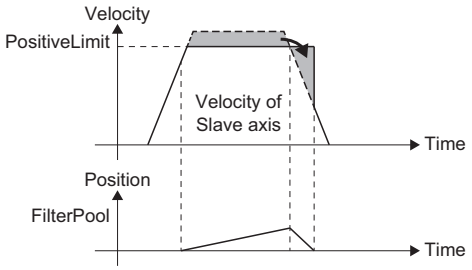
Input pin	Setting value
Positive direction filter operation (PositiveFilter)	4: Clamp (Without Ramp at Deceleration) (ClampWithoutRamp)



■Positive direction filter operation (PositiveFilter)/Negative direction filter operation (NegativeFilter)

• Sets the slave axis operation at control based on the setting of Positive direction filter operation (PositiveFilter) and Negative direction filter operation (NegativeFilter).

Setting value	Description
0: Ignore (Ignore)	• The velocity is not limited. • Filter cumulated value (FilterPool) is not updated. 
2: Truncate (Truncate)	• The amount exceeding the control value is not output to Slave axis (Slave). • The restricted movement amount will be truncated, and the position relation between the master axis and the slave axis will deviate by the truncated amount. 

Setting value	Description
4: Clamp (Without Ramp at Deceleration) (ClampWithoutRamp)	• The amount exceeding the control value is cumulated as Filter cumulated value (FilterPool) and output later. The Filter cumulated value (FilterPool) output does not operate deceleration. • When "0.0" is set as the control value, all the master axis input to the applicable direction will be cumulated as Filter cumulated value (FilterPool) and will not be output later to the slave axis. The cumulated amount will be subtracted by the reverse direction master axis input, and the slave axis will not operate until Filter cumulated value (FilterPool) is "0.0". ■When Filter cumulated value is greater than the decelerated distance  ■When Filter cumulated value is smaller than the decelerated distance 

- When setting a value outside the range to Filter operation (Filter), "Out of Filter Operation Specification Range (error code: 1A86H)" occurs.

■Master axis data source selection (MasterValueSource)

Slave axis (Slave) sets the position type of Master axis (Master) that will execute single axis synchronous control.
The operation may differ depending on the execution order of the FBs or if the operation cycles of Master axis (Master) and Slave axis (Slave) are different.
Operates with the same value as the set value when specifying Actual value in Master axis data source selection (MasterValueSource) while an axis type other than real axis is set in Master axis (Master).

Setting value	Description
1: Set value (mcSetValue)	Uses the commanded position of the master axis in the previous operation cycle.
2: Actual value (mcActualValue)	Uses the actual position of the master axis in the previous operation cycle.
101: Latest set value (mcLatestSetValue)	Uses the commanded position of the master axis in the current operation cycle.
102: Latest actual value (mcLatestActualValue)	Uses the actual position of the master axis in the current operation cycle.

Point

When Master axis data source selection (MasterValueSource) is set to "1: Set value (mcSetValue)" or "101: Latest set value (mcLatestSetValue)" and the master axis becomes servo OFF by a servo alarm or forced stop, the amount of the value change may be too large. This can be prevented by setting Master axis data source selection (MasterValueSource) to "2: Actual value (mcActualValue)" or "102: Latest actual value (mcLatestActualValue)".

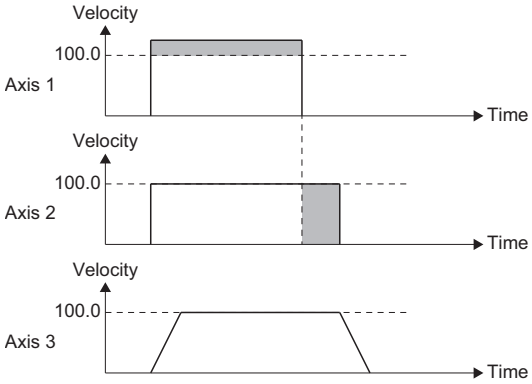
Precautions

- Note that if Filter disable (FilterDisable) is switched during operation, the Slave axis (Slave) velocity may change rapidly.
- After enabling the command filter, when the Master axis (Master) command is not transmitted to Slave axis (Slave) or the back part synchronous control FB because the stop factor occurred on Slave axis (Slave), the synchronous position relation deviates. Before enabling the command filter again, execute the synchronous positioning as needed.

Program example

In the following program example, filter processing is performed to limit the velocity of the synchronous encoder.

- Operation



- Settings

Item		Setting value
Positive direction/negative direction limit value	Positive direction limit value	100.0
	Negative direction limit value	0.0
Backlash amount		1.0

■Axes

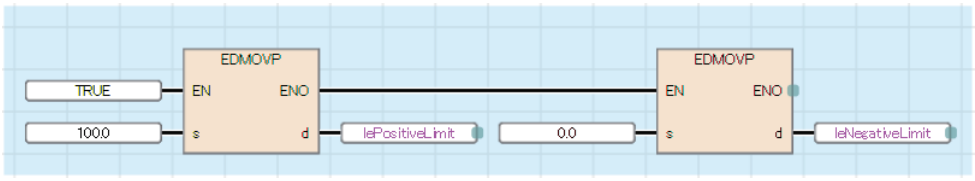
Axis No.	Label name	Data type	Comment
1	EncoderAxis0001	AXIS_REF	Axis 1
2	Axis0002	AXIS_REF	Axis 2
3	Axis0003	AXIS_REF	Axis 3

■Labels used

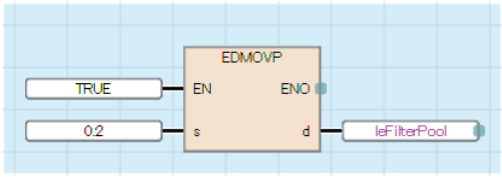
Label name	Data type	Comment
MCv_SpeedLimitFilter_1	MCv_SpeedLimitFilter	Speed Limit Filter FB
bSpeedLimitFilter	Bit	Speed limit filter command
lePositiveLimit	Double-precision real number	Positive direction limit value
leNegativeLimit	Double-precision real number	Negative direction limit value
bSpeedLimitValid	Bit	Output value valid
bSpeedLimitBusy	Bit	Executing
bSpeedLimitActive	Bit	Controlling
bSpeedLimitCommandAborted	Bit	Abortion of execution
bSpeedLimitError	Bit	Error
uwSpeedLimitErrorID	Word [unsigned]/bit string [16-bit]	Error code
leFilterPool	Bit	Filter cumulated value
MCv_SmoothingFilter_1	MCv_SmoothingFilter	Smoothing Filter FB
bSmoothingFilter	Bit	Smoothing filter command
leFrequency	Double-precision real number	Frequency
bSmoothingValid	Bit	Output value valid
bSmoothingInVelocity	Bit	Target velocity reached
bSmoothingBusy	Bit	Executing
bSmoothingActive	Bit	Controlling
bSmoothingCommandAborted	Bit	Abortion of execution
bSmoothingError	Bit	Error
uwSmoothingErrorID	Word [unsigned]/bit string [16-bit]	Error code
leSmoothingValidFrequency	Double-precision real number	Valid frequency
leSmoothingFilterPool	Double-precision real number	Filter cumulated value

■FBD/LD program (CPU module side)

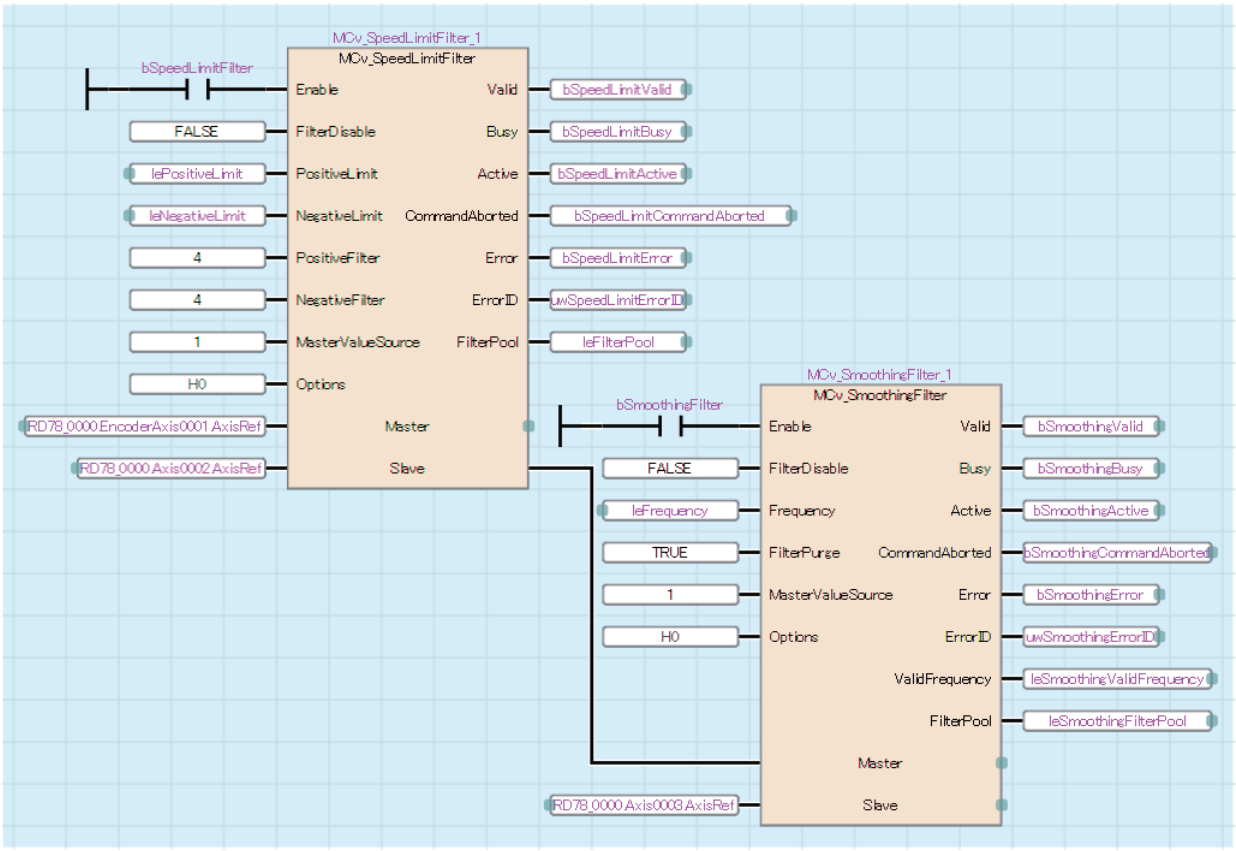
- Positive direction/negative direction limit value



- Frequency



- Speed Limit Filter/Smoothing Filter



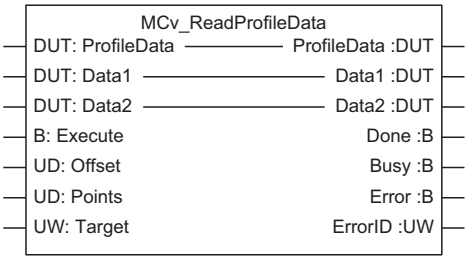
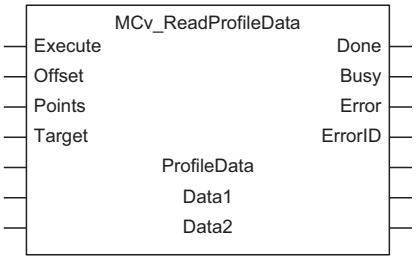
■ST program (Motion module side)

```
//-----Positive direction/negative direction limit value-----
lePositiveLimit:= 100.0;
leNegativeLimit:= 0.0;

//-----Speed Limit Filter-----
MCv_SpeedLimitFilter_1(
   Master:= EncoderAxis0001.AxisRef ,
   Slave:= Axis0002.AxisRef ,
   Enable:= bSpeedLimitFilter ,
   FilterDisable:= FALSE ,
   PositiveLimit:= lePositiveLimit ,
   NegativeLimit:= leNegativeLimit ,
   PositiveFilter:= MC_VELOCITY_LIMIT_MODE__ClampWithoutRamp ,
   NegativeFilter:= MC_VELOCITY_LIMIT_MODE__ClampWithoutRamp ,
   MasterValueSource:= MC_SOURCE__mcSetValue ,
   Options:= H00000000 ,
   Valid=> bSpeedLimitValid ,
   Busy=> bSpeedLimitBusy ,
   Active=> bSpeedLimitActive ,
   CommandAborted=> bSpeedLimitCommandAborted ,
   Error=> bSpeedLimitError ,
   ErrorID=> uwSpeedLimitErrorID ,
   FilterPool=> leFilterPool
);

//-----Frequency-----
leFrequency:= 0.20;

//-----Smoothing Filter-----
MCv_SmoothingFilter_1(
   Master:= MCv_SpeedControl_1.Slave ,
   Slave:= Axis0003.AxisRef ,
   Enable:= bSmoothingFilter ,
   FilterDisable:= FALSE ,
   Frequency:= leFrequency ,
   FilterPurge:= TRUE ,
   MasterValueSource:= MC_SOURCE__mcSetValue ,
   Options:= H00000000 ,
   Valid=> bSmoothingValid ,
   Busy=> bSmoothingBusy ,
   Active=> bSmoothingActive ,
   CommandAborted=> bSmoothingCommandAborted ,
   Error=> bSmoothingError ,
   ErrorID=> uwSmoothingErrorID ,
   ValidFrequency=> leSmoothingValidFrequency ,
   FilterPool=> leSmoothingFilterPool
);
```

3.3 Standard FBs

MCv_ReadProfileData

Reads the specified calculation profile data (cam data) from the open area or file.

Ladder	FBD/LD	ST
		MCv_ReadProfileData(ProfileData:= ?PROFILE_DATA? , Data1:= ?TARGET_REF? , Data2:= ?TARGET_REF? , Execute:= ?BOOL? , Offset:= ?DWORD? , Points:= ?DWORD? , Target:= ?WORD? , Done=> ?BOOL? , Busy=> ?BOOL? , Error=> ?BOOL? , ErrorID=> ?WORD?);

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Profile Read	422	10	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
ProfileData	Profile data	PROFILE_DATA	At start	—	Omission not possible	This variable sets the calculation profile to read. The calculation profile is set with a PROFILE_DATA structure. For details of the PROFILE_DATA structure, refer to the following. ☞ Page 66 PROFILE_DATA Read operation is possible for calculation profile data formats with the structure of the data for reading defined. If specifying a calculation profile that does not correspond to read operation, "Calculation Profile Read/Write Unsupported (error code: 345EH)" occurs.
Data1	Read data 1	TARGET_REF	At start	—	Omission not possible	This variable specifies the structure of the data for reading according to the calculation profile format to read. For details, refer to the following. ☞ Page 422 Write data 1 (Data1)/Write data 2 (Data2)
Data2	Read data 2	TARGET_REF	At start	—	Omission not possible	This variable specifies the structure of the data for reading according to the calculation profile format to read. For details, refer to the following. ☞ Page 422 Write data 1 (Data1)/Write data 2 (Data2)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Start	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MCv_ReadProfileData (Profile Read).
Offset	Offset	DWORD(UDINT)	At start	0 to resolution (2 to 65535)	0	This variable specifies the offset from the top of the calculation profile. If Offset (Offset) exceeds the range of the calculation profile, "Out of Offset Range (error code: 3413H)" occurs and reading will not be performed. When setting a value other than "0" in Offset (Offset), set a value other than "0" in Number of read data (Points). (When "0" is set in Number of read data (Points), "Number of Offset/Read/Write Data Incorrect (error code: 3465H)" occurs.) *: The read operation differs depending on the set values for Offset (Offset) and Number of read data (Points). For details, refer to the following. ☞ Page 409 Offset (Offset) and Number of read data (Points) setting
Points	Number of read data	DWORD(UDINT)	At start	0 to 4294967295	0	This variable specifies the number of data points to read. When Number of read data (Points) exceeds the number of elements of the calculation profile, reading is performed within the number of elements of the calculation profile range. When Interpolation method specification (Interpolate) of the calculation profile is "1: Specifying for each section" or "2: Spline interpolation", set this variable to "0". (When a value other than "0" is set, "Number of Offset/Read/Write Data Incorrect (error code: 3465H)" occurs.)
Target	Read destination	WORD(UINT)	At start	0, 1	0	This variable specifies the read destination of the calculation profile. • 0: Open area • 1: File For details, refer to the following. ☞ Page 415 Read destination (Target)

■Output variables

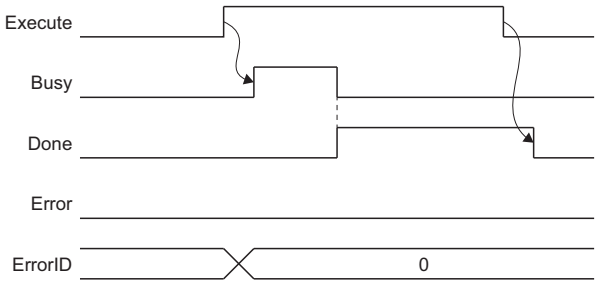
Output variable	Name	Data type	Default value	Description
Done	Completion	BOOL	FALSE	This variable becomes TRUE when the control is completed. This variable changes as shown below based on the status of Start (Execute) at operation completion. ■When Start (Execute) is TRUE This variable remains TRUE until Start (Execute) becomes FALSE. ■When Start (Execute) is FALSE This variable becomes TRUE for one cycle only.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MCv_ReadProfileData (Profile Read) is executed.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. MELSEC iQ-R Motion Module User's Manual (Application)

Processing details

- Performs read operation for the calculation profile using MCv_ReadProfileData (Profile Read).
- The file specified in Profile data (ProfileData) and Read destination (Target), or the calculation profile in the open area are read for the number of points specified in Number of read data (Points) from the data specified with Offset (Offset).
- To read the whole calculation profile, specify "0" in both Offset (Offset) and Number of read data (Points).
- Read data is stored in variables specified in Read data 1 (Data1) and Read data 2 (Data2).

■Timing chart

- When the operation completes normally



- When an error occurs

  For the timing chart when an error occurs, refer to the following.

  ☞ Page 11 Basic operation of Execute command (Execute) type Motion control FBs

■Offset (Offset) and Number of read data (Points) setting

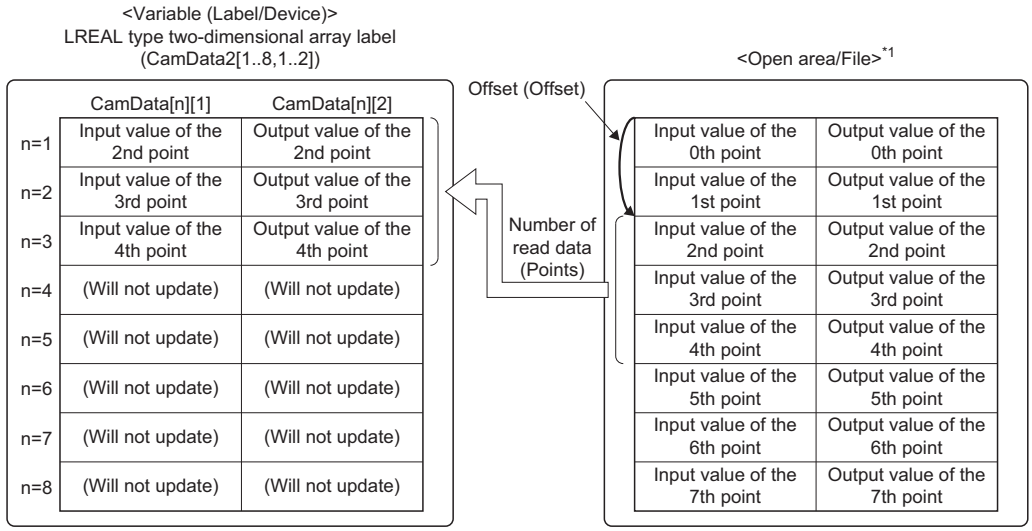
The following operations are based on the values set in Offset (Offset) and Number of read data (Points).

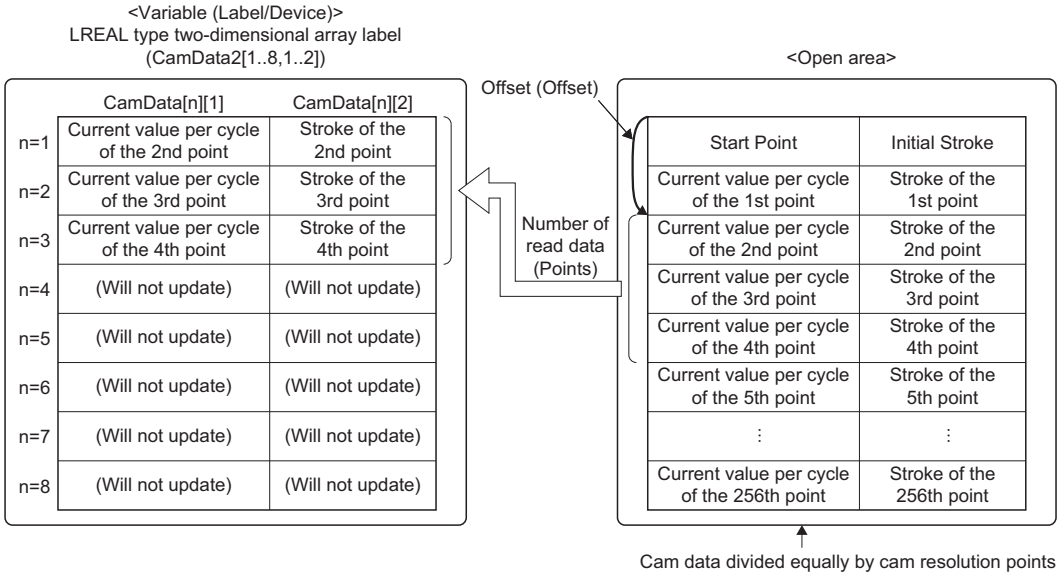
Offset (Offset)	Number of read data (Points)	Size relationship with the number of elements of the calculation profile	Operation
0	0	—	Reads all calculation profile elements.
	Other than 0	Offset (Offset) + Number of read data (Points) ≤ number of elements of the calculation profile	Reads the number of calculation profile elements set in Number of read data (Points) from the top.
	Other than 0	Offset (Offset) + Number of read data (Points) > number of elements of the calculation profile	Reads all calculation profile elements.
Other than 0	0	Offset (Offset) ≤ number of elements of the calculation profile	"Number of Offset/Read/Write Data Incorrect (error code: 3465H)" occurs, and read is not executed.
	Other than 0	Offset (Offset) + Number of read data (Points) ≤ number of elements of the calculation profile	Reads the number of calculation profile elements specified in Number of read data (Points) from Offset (Offset).
	Other than 0	Offset (Offset) + Number of read data (Points) > number of elements of the calculation profile	Reads until the final element of the calculation profile from Offset (Offset).
	—	Offset (Offset) > number of elements of the calculation profile	"Out of Offset Range (error code: 3413H)" occurs, and read does not happen.

The contents of the values set for Offset (Offset) and Number of read data (Points) vary depending on the calculation profile format.

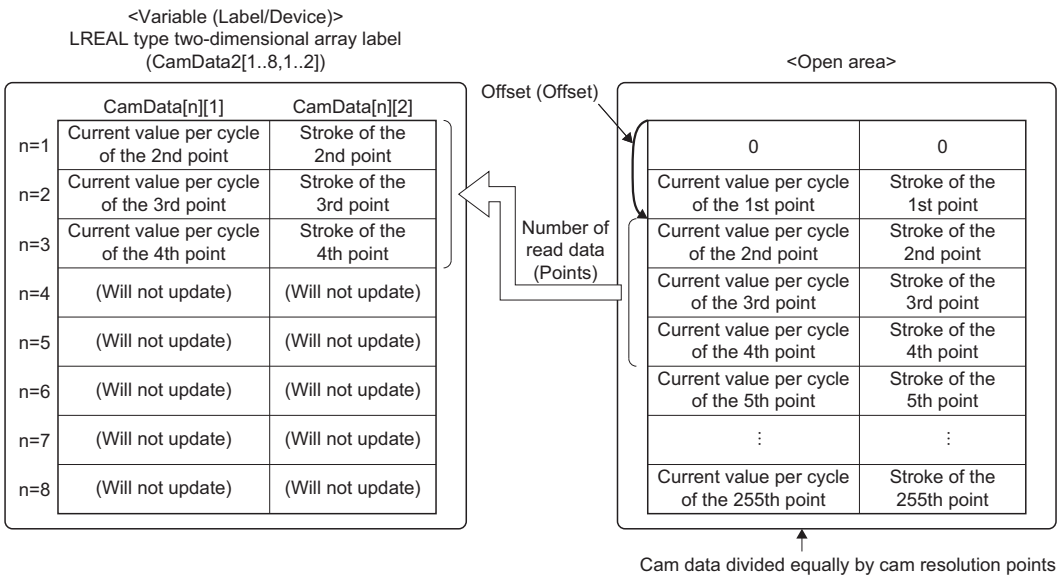
The following describes the contents when "cam data" and "rotary cutter" are set to the calculation profile format.

- When the calculation profile format is "cam data"

Specifying interpolation method (Interpolate)	Description
0: Linear Interpolation	Sets the top of the coordinate to be read in Offset (Offset), and the number of coordinates in Number of read data (Points). The data stored in Read data 1 (Data1) is the PROFILE_CAM_DATA type data when Specifying interpolation method (Interpolate) is "0: Linear interpolation". The data stored in Read data 2 (Data2) is data defined by using set coordinate data (a pair of input value (current value per cycle) and output value (stroke)) in LREAL type two-dimensional array. <Example> Read result of Read data 2 (Data2) when Offset (Offset) is set to "2" and Number of read data (Points) is set to "3". (When the variable set to Read data 2 (Data2) is a "LREAL[1..8, 1..2]" data type "CamData2" label.)  *1: Depends on Read destination (Target).

Specifying interpolation method (Interpolate)	Description
1: Specifying for each section 2: Spline Interpolation	The data that can be read differs depending on the Read destination (Target). ■When Read destination (Target) is "0: Open area" Sets the top of the section to be read in Offset (Offset), and the Number of sections (NumberOfSections) in Number of read data (Points). The data stored in Read data 1 (Data1) is the PROFILE_CAM_DATA type data when Specifying interpolation method (Interpolate) is "0: Linear interpolation". The data stored in Read data 2 (Data2) is data defined by using target cam data divided equally by the cam resolution points (a pair of current value per cycle and stroke) which is defined in LREAL type two-dimensional array. <Example> Read result of Read data 2 (Data2) when Offset (Offset) is set to "2" and Number of read data (Points) is set to "3" and cam resolution is "256". (When the variable set to Read data 2 (Data2) is a "LREAL[1..8, 1..2]" data type "CamData2" label.)  ■When Read destination (Target) is "1: File" Sets Offset (Offset) and Number of read data (Points) to "0", and reads all elements. (When these are set to a setting other than "0", a "Number of Offset/Read/Write Data Incorrect (error code: 3465H)" error occurs.) The data stored in Read data 1 (Data1) is the PROFILE_CAM_DATA type data when Specifying interpolation method (Interpolate) is "1: Specifying for each section" or "2: Spline Interpolation". The data stored in Read data 2 (Data2) is section data defined in PROFILE_CAM_ELEMENT type array.

• When the calculation profile format is "rotary cutter"

Read destination (Target)	Description
0: Open area	Sets the top of the section to be read in Offset (Offset), and the Number of sections (NumberOfSections) in Number of read data (Points). The data stored in Read data 1 (Data1) is the PROFILE_CAM_DATA type data when Specifying interpolation method (Interpolate) is "0: Linear interpolation" and the calculation profile format is "cam data". The data stored in Read data 2 (Data2) is data defined by using target cam data divided equally by the cam resolution points (a pair of current value per cycle and stroke) which is defined in LREAL type two-dimensional array. <Example> Read result of Read data 2 (Data2) when Offset (Offset) is set to "2" and Number of read data (Points) is set to "3". (When the variable set to Read data 2 (Data2) is a "LREAL[1..8, 1..2]" data type "CamData2" label.) 
1: File	Ignores the setting for Offset (Offset) and Number of read data (Points). The data stored in Read data 1 (Data1) is data where the calculation profile format is PROFILE_ROTARY_CUTTER type "rotary cutter". There is no data stored in Read data 2 (Data2).

■Read data 1 (Data1)/Read data 2 (Data2)

Set read data of the calculation profile with a TARGET_REF structure. The [VAR] and [DEV] data types can be used.

For details of the TARGET_REF structure, refer to the following.

☞ Page 64 TARGET_REF (Input signal)

For data types, read data structures of each calculation profile format can be used.

• Read data 1 (Data1)

The calculation profile data formats and data types set in Read data 1 (Data1) are shown below.

Calculation profile data format	Data type	Reference
Cam data	PROFILE_CAM_DATA	☞ Page 68 PROFILE_CAM_DATA
Cam data for rotary cutter	PROFILE_ROTARY_CUTTER	☞ Page 71 PROFILE_ROTARY_CUTTER

Ex.

When reading by using a device or label

· When reading a calculation profile of the cam data format with buffer memory (G11480000 or more)

Data1.Target := "[DEV](PROFILE_CAM_DATA)G11480000"

· When reading a PROFILE_CAM_DATA type label (CamData1)

Data1.Target := "[VAR]CamData1"

• Read data 2 (Data2)

The calculation profile formats and data types set in Read data 2 (Data2) are shown below. The data type of the structure that is set varies depending on the Read destination (Target) and Specifying interpolation method (Interpolate) of the target cam data.

○: Supported, ×: Not supported

Calculation profile format	Read Destination (Target)	Data type	Specifying interpolation method (Interpolate)			Reference
			0: Linear interpolation	1: Specifying for each section	2: Spline interpolation	
Cam data	0: Open area	LREAL[ ]*1	○	○*3	○*3	—
	1: File	LREAL[ ]*1	○	×	×	—
		PROFILE_CAM_ELEMENT [ ]*2	×	○	○*4	☞ Page 70 PROFILE_CAM_ELEMENT
Cam data for rotary cutter	Setting is unnecessary. If set, the setting is ignored.					—

*1 The LREAL type two-dimensional array is set as follows.
LREAL[m..n, o..p]

Item	Description
Number of elements (n-m+1)	Sets the array elements so that they are greater than the number of points that are to be read. When reading the entire calculation profile (Offset (Offset), Number of read data (Points) are set to "0"), set this to the cam data resolution +1/the number of coordinates or more. However, an area bigger than the resolution +1/number of coordinates in Read data 1 (Data1) will not be updated. If the ProfileControl (Calculation profile control) version is "earlier than 1.19", it is necessary to match the number of elements of the specified array to the cam data resolution +1/number of coordinates.
Number of dimensions (p-o+1)	Reads the stroke and current position per cycle for each point. Set the array element to "2". If the array element is set to "3" or more, a read is possible, but as the intended result of the read may not be obtained due to data being moved forward, make sure to set the array element to "2".

*2 The PROFILE_CAM ELEMENT array is set as follows.
PROFILE_CAM_ELEMENT[m..n]

Item	Description
Number of elements (n-m+1)	Sets the array elements so that they are greater than the number of points that are to be read. When reading the entire calculation profile (Offset (Offset), Number of read data (Points) are set to "0"), set this to the cam data Number of sections (NumberOfSections) or more. However, an area bigger than the Number of sections (NumberOfSections) in Read data 1 (Data1) will not be updated. If the ProfileControl (Calculation profile control) version is "earlier than 1.19", it is necessary to match the number of elements of the specified array to the cam data Number of sections (NumberOfSections).

*3 Read as stroke data (a pair of current value per cycle and stroke) opened to the open area. X of 1st point data is the value set in start point, and Y of 1st point data is the value set in initial stroke amount.
Data after the 2nd point is stroke data divided by the resolution. Due to this, the read points in the open area is "resolution+1".

*4 Only used by the end point and stroke.

The following errors in ProfileControl (Calculation profile control) may occur depending on the version used.

ProfileControl (Calculation profile control)	Description
Version earlier than "1.19"	When the number of elements of the specified array does not match the resolution+1/number of coordinates, a "Number of Read/Write Data Mismatch (error code: 3464H)" occurs.
Version "1.19" and later	If the number of the elements of the specified array is smaller than the resolution+1/number of coordinates when reading the entire calculation profile, a "Number of Read/Write Data Mismatch (error code: 3464H)" occurs.

Ex.

When setting LREAL type with Specifying interpolation method (Interpolate)

Specifying interpolation method (Interpolate)	Example setting
0: Linear Interpolation	The following shows examples of setting the two-dimensional device/label coordinate number "100" to Target (Target) of the TARGET_REF structure. ■Buffer memory specification Data2.Target := "[DEV](LREAL[1..100,1..2])G11483000" ■Label specification Data2.Target := "[VAR]CamData2" *: CamData2 declares the "LREAL[1..100,1..2]" label to data type and label name of the global label to "CamData2", and specifies the character string for Target (Target). The read data is as shown below. ・CamData2[n,1]: nth point current position per cycle ・CamData2[n,2]: nth point stroke
1: Specifying for each section 2: Spline Interpolation	The following shows examples of setting the two-dimensional device/label resolution "100" to Target (Target) of the TARGET_REF structure. ■Buffer memory specification TARGET_REF.Target := "[DEV](LREAL[1..101,1..2])G11483000" ■Label specification TARGET_REF.Target := "[VAR]CamData2" *: CamData2 declares the "LREAL[1..101,1..2]" label to data type and label name of the global label to "CamData2", and specifies the character string for Target (Target). The read data will be as shown below. ・CamData2[1,1]: Value set for the start point ・CamData2[1,2]: Value set for the initial stroke amount ・CamData2[n+1,1]: nth point current position per cycle ・CamData2[n+1,2]: nth point stroke *: When specifying labels, local labels can also be set. However, when setting local labels, specify the "POU name" to "@target modification". (☞ Page 64 TARGET_REF (Input signal))

Ex.

For PROFILE_CAM_ELEMENT type

Example setting
The following shows examples of setting the two-dimensional device/label number of sections "10" to Target (Target) of the TARGET_REF structure. ■Buffer memory specification Data2.Target := "[DEV](PROFILE_CAM_ELEMENT[1..10])G11483000" ■Label specification Data2.Target := "[VAR]CamData2" *: CamData2 declares the "PROFILE_CAM_ELEMENT[1..10]" label to data type and label name of the global label to "CamData2", and specifies the character string for Target (Target). The read data is as shown below. ・CamData2[n]: nth point section data (PROFILE_CAM_ELEMENT type) *: When specifying labels, local labels can also be set. However, when setting local labels, specify the "POU name" to "@target modification". (☞ Page 64 TARGET_REF (Input signal))

Point

- If the type of the specified data structure for reading does not match the type that is required for each calculation profile, "Read/Write Data Type Incorrect (error code: 3462H)" occurs.
- For details of the cam data for the calculation profile set in Read data 1 (Data1)/Read data 2 (Data2), refer to the following.
  📖MELSEC iQ-R Motion Module User's Manual (Application)

■Read destination (Target)

Sets the read destination of the calculation profile.

Setting value	Description
0: Open area	• The Profile ID (ID) value of the PROFILE_DATA structure is referred, and the read is performed for the calculation profile that Profile ID No. (Number) of the PROFILE_ID structure indicates in the open area. • If the calculation profile does not exist (unopened), "No Calculation Profile (error code: 340FH)" occurs.
1: File	• The Calculation profile data storage location (Location) value of the PROFILE_DATA structure is referred, and the read is performed for the calculation profile that the file name and the path indicate. • If the calculation profile does not exist, "No Calculation Profile (error code: 340FH)" occurs.

Point

- When Read destination (Target) is "1: File", processing is performed at a cycle with lower priority than the operation cycle. Therefore, it may take time to read depending on the processing details and calculation profile.
- When Read destination (Target) is "0: Open area", the read is performed according to the cycle of the executed program.

Precautions

- The number of data points that can be read at the same time are limited. If the read is not completed in a single FB execution, execute the read over several FB executions.

Program example

In the following program example, "LREAL type (two-dimensional array) format" cam data 1 (ProfileData0001) and "PROFILE_CAM_ELEMENT format" cam data 2 (ProfileData0002) are read from the file.

■Calculation profile

Profile ID	Label name	Data type	Comment
1	ProfileData0001	MC_CAM_REF	Cam data 1
2	ProfileData0002	MC_CAM_REF	Cam data 2

- Settings for calculation profile (cam data)

Item		ProfileData0001	ProfileData0002
Specifying interpolation method		Linear interpolation	Specifying for each section
Resolution		—	256
Length per cycle setting	Unit	degree	degree
	Length per cycle	360.00000	360.00000
Stroke amount	Unit	pulse	%
	Stroke amount	100.0	100.0000000
Cam 1 cycle time		—	1.000[s]
Initial stroke value		—	0.0000000
Cam 1 stroke minimum value		0.00000	—
Cam 1 stroke maximum value		360.00000	—

- ProfileData0001

Section No.	Input value [degree]	Output value [pulse]
1	0.00000	0.0
2	90.00000	60.0
3	180.00000	100.0
4	270.00000	30.0
5	360.00000	0.0

• ProfileData0002

Section No.	Start point [degree]	End point [degree]	Stroke [%]	Cam curve type
1	0.00000	90.00000	60.0000000	Single hypotenuse
2	90.00000	180.00000	100.0000000	Single hypotenuse
3	180.00000	270.00000	30.0000000	Single hypotenuse
4	270.00000	0.00000	0.0000000	Single hypotenuse

■Labels used

Label name	Data type	Comment
MCv_ReadProfileData_1	MCv_ReadProfileData	Profile Read FB 1
ReadData1_1	TARGET_REF	Read data 1 Profile 1
ReadData2_1	TARGET_REF	Read data 2 Profile 1
ReadData1_Lreal	PROFILE_CAM_DATA	Profile data Cam data 1
ReadData2_Lreal	Double-precision real number(0..4,0..1)	Input value/Output value data LREAL type
bReadProfile1	Bit	Profile read command 1
bDone1	Bit	Completion 1
bBusy1	Bit	Executing 1
bError1	Bit	Error
uwErrorID1	Word [unsigned]/bit string [16-bit]	Error code
ProfileData0001*1	PROFILE_DATA	Profile data 0001
MCv_ReadProfileData_2	MCv_ReadProfileData	Profile Read FB 2
ReadData1_2	TARGET_REF	Read data 1 Profile 2
ReadData2_2	TARGET_REF	Read data 2 Profile 2
ReadData1_Element	PROFILE_CAM_DATA	Profile data Cam data 2
ReadData2_Element	PROFILE_CAM_ELEMENT(0..3)	Input value/Output value data PROFILE_CAM_ELEMENT type
bReadProfile2	Bit	Profile read command 2
bDone2	Bit	Completion 2
bBusy2	Bit	Executing 2
bError2	Bit	Error 2
uwErrorID2	Word [unsigned]/bit string [16-bit]	Error code 2
ProfileData0002*1	PROFILE_DATA	Profile data 0002

*1 Only used in FBD/LD programs.

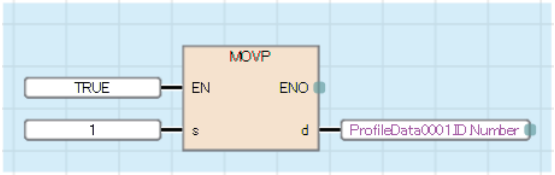
■Global labels used

Label name	Data type	Comment	Public label
G_ProfileCamData*1	PROFILE_CAM_DATA	Profile cam data	Enabled
G_ProfileCamELEMENT*1	PROFILE_CAM_ELEMENT	Profile cam element	Enabled

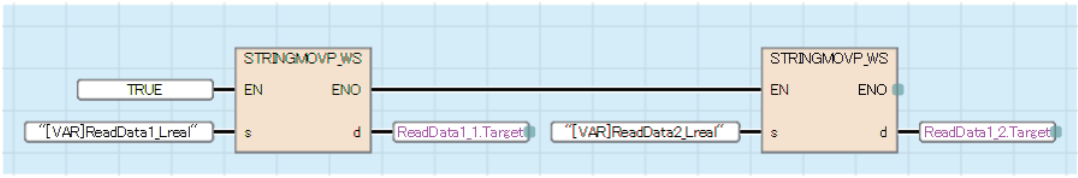
*1 Only used in FBD/LD programs.

■FBD/LD program (CPU module side)

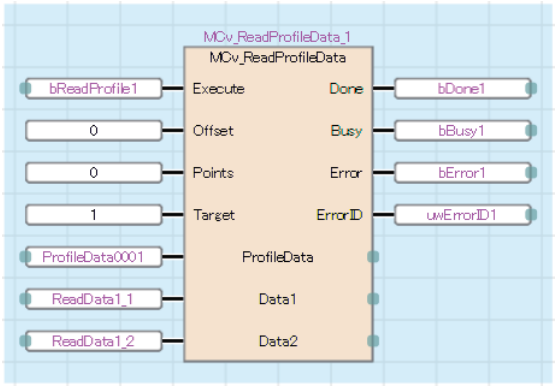
- Profile data 0001



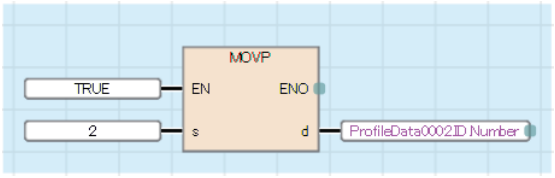
- Read data 1



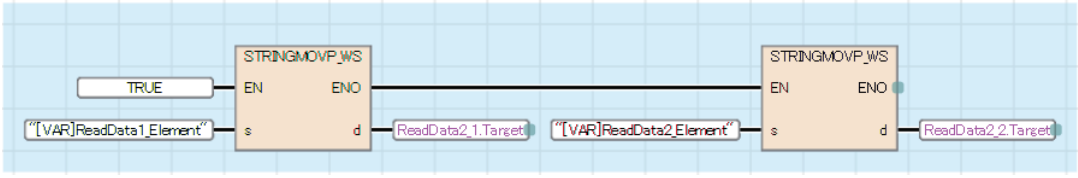
- Profile Read 1 (reads cam data in the LREAL type (two-dimensional array) format)



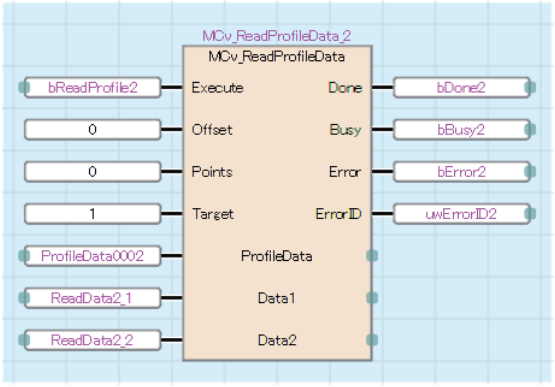
- Profile data 0002



- Read data 2



- Profile Read 2 (reads cam data in the PROFILE_CAM_ELEMENT type format)

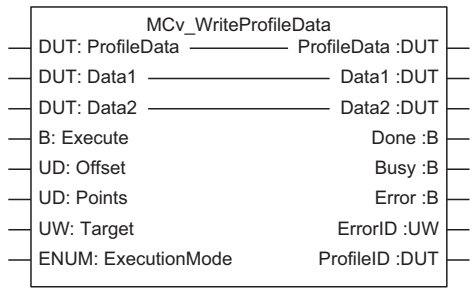
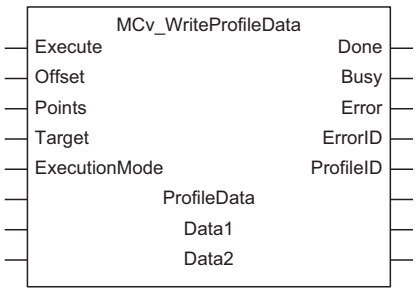


■ST program (Motion module side)

```
//-----Read data 1-----
ReadData1_1.Target := "[VAR]ReadData1_Lreal";
ReadData2_1.Target := "[VAR]ReadData2_Lreal";

//-----Profile Read 1-----
//-----Reads cam data in the LREAL type (two-dimensional array) format)-----
MCv_ReadProfileData_1(
   ProfileData:= ProfileData0002.ProfileData ,
   Data1:= ReadData1_1 ,
   Data2:= ReadData2_1 ,
   Execute:= bReadProfile1 ,
   Offset:= 0 ,
   Points:= 0 ,
   Target:= 1 ,
   Done=> bDone1 ,
   Busy=> bBusy1 ,
   Error=> bError1 ,
   ErrorID=> uwErrorID1
);

//-----Read Data 2-----
ReadData1_2.Target := "[VAR]ReadData1_Element";
ReadData2_2.Target := "[VAR]ReadData2_Element";

//-----Profile Read 2-----
//-----Reads cam data in the PROFILE_CAM_ELEMENT type format-----
MCv_ReadProfileData_2(
   ProfileData:= ProfileData0002.ProfileData ,
   Data1:= ReadData1_2 ,
   Data2:= ReadData2_2 ,
   Execute:= bReadProfile2 ,
   Offset:= 0 ,
   Points:= 0 ,
   Target:= 1 ,
   Done=> bDone2 ,
   Busy=> bBusy2 ,
   Error=> bError2 ,
   ErrorID=> uwErrorID2
);
```

MCv_WriteProfileData

Writes the specified calculation profile data to the open area or file.

Ladder	FBD/LD	ST
		MCv_WriteProfileData(ProfileData:= ?PROFILE_DATA? , Data1:= ?TARGET_REF? , Data2:= ?TARGET_REF? , Execute:= ?BOOL? , Offset:= ?DWORD? , Points:= ?DWORD? , Target:= ?WORD? , ExecutionMode:= ?INT? , Done=> ?BOOL? , Busy=> ?BOOL? , Error=> ?BOOL? , ErrorID=> ?WORD? , ProfileID=> ?PROFILE_ID?);

Name	Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
Profile Write	424	16	Subroutine type	Real-time execution

Setting data

■I/O variables

I/O variable	Name	Data type	Input import	Setting range	Default value	Description
ProfileData	Profile data	PROFILE_DATA	At start	—	Omission not possible	This variable sets the calculation profile to write. The calculation profile is set with a PROFILE_DATA structure. For details of the PROFILE_DATA structure, refer to the following. ☞ Page 66 PROFILE_DATA Write operation is possible for calculation profile formats with the structure of the data for writing defined. If specifying calculation profile that does not correspond to write operation, "Calculation Profile Read/Write Unsupported (error code: 345EH)" occurs.
Data1	Write data 1	TARGET_REF	At start	—	Omission not possible	This variable specifies the structure of the data for writing according to the calculation profile format to write. For details, refer to the following. ☞ Page 422 Write data 1 (Data1)/Write data 2 (Data2)
Data2	Write data 2	TARGET_REF	At start	—	Omission not possible	This variable specifies the structure of the data for writing according to the calculation profile format to write. For details, refer to the following. ☞ Page 422 Write data 1 (Data1)/Write data 2 (Data2)

■Input variables

Input variable	Name	Data type	Input	Setting range	Default value	Description
Execute	Start	BOOL	At start	TRUE, FALSE	FALSE	When this variable is TRUE, executes MCv_WriteProfileData (Profile Write).
Offset	Offset	DWORD(UDINT)	At start	0 to resolution (2 to 65535)	0	This variable specifies the offset from the top of the calculation profile. If Offset (Offset) exceeds the range of the calculation profile, "Out of Offset Range (error code: 3413H)" occurs and writing will not be performed. When setting a value other than "0" in Offset (Offset), set a value other than "0" in Number of write data (Points). (When "0" is set in Number of write data (Points), "Number of Offset/Read/Write Data Incorrect (error code: 3465H)" occurs.) *: The write operation differs depending on the set values for Offset (Offset) and Number of write data (Points). For details, refer to the following. ☞ Page 422 Offset (Offset) and Number of write data (Points) setting
Points	Number of write data	DWORD(UDINT)	At start	0 to 4294967295	0	This variable specifies the number of data points to write. When Number of write data (Points) exceeds the number of elements of the calculation profile, writing is performed within the number of elements of the calculation profile range. When writing a calculation profile and Write designation (Target) is "1: File", set this variable to "0". (When a value other than "0" is set, "Number of Offset/Read/Write Data Incorrect (error code: 3465H)" occurs.)
Target	Write destination	WORD(UINT)	At start	0, 1	0	This variable specifies the write destination of the calculation profile. • 0: Open area • 1: File *: When Write designation (Target) is set to "1: File", set Offset (Offset) and Number of write data (Points) to "0". (When a value other than "0" is set, "Number of Offset/Read/Write Data Incorrect (error code: 3465H)" occurs.) For details, refer to the following. ☞ Page 425 Write destination (Target)
ExecutionMode	Execution mode	INT (MC_EXECUTION_MODE)	At start	0, 1, 3	0	This variable specifies the execution timing of MCv_WriteProfileData (Profile Write). • 0: Execute immediately (mcImmediately) • 1: Execute at completion (mcQueued) • 3: Execute speculatively (mcSpeculatively) For details, refer to the following. ☞ Page 425 Execution mode (ExecutionMode)

■Output variables

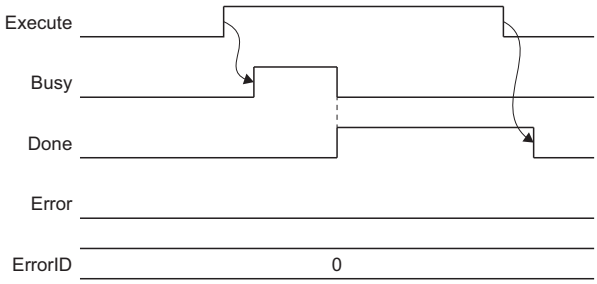
Output variable	Name	Data type	Default value	Description
Done	Completion	BOOL	FALSE	This variable becomes TRUE when the control is completed. This variable changes as shown below based on the status of Start (Execute) at operation completion. ■When Start (Execute) is TRUE This variable remains TRUE until Start (Execute) becomes FALSE. ■When Start (Execute) is FALSE This variable becomes TRUE for one cycle only.
Busy	Executing	BOOL	FALSE	This variable becomes TRUE when MCv_WriteProfileData (Profile Write) is executed.
Error	Error	BOOL	FALSE	This variable becomes TRUE when an error occurs.
ErrorID	Error code	WORD(UINT)	0	When an error occurs, this variable returns the error code. For details of error codes, refer to the following. 📖MELSEC iQ-R Motion Module User's Manual (Application)
ProfileID	Profile ID	PROFILE_ID	0	This variable outputs the profile ID.

Processing details

- Performs write operation for the calculation profile data using MCv_WriteProfileData (Profile Write).
- The file specified in Profile data (ProfileData) and Write destination (Target), or the calculation profile data in the open area are written for the number of points specified in Number of write data (Points) from the data specified with Offset (Offset).
- To write the whole calculation profile data, specify "0" in both Offset (Offset) and Number of write data (Points).
- Write data is stored in variables specified in Write data 1 (Data1) and Write data 2 (Data2).

■Timing chart

- When the operation completes normally



- When an error occurs

  For the timing chart when an error occurs, refer to the following.

  ☞ Page 11 Basic operation of Execute command (Execute) type Motion control FBs

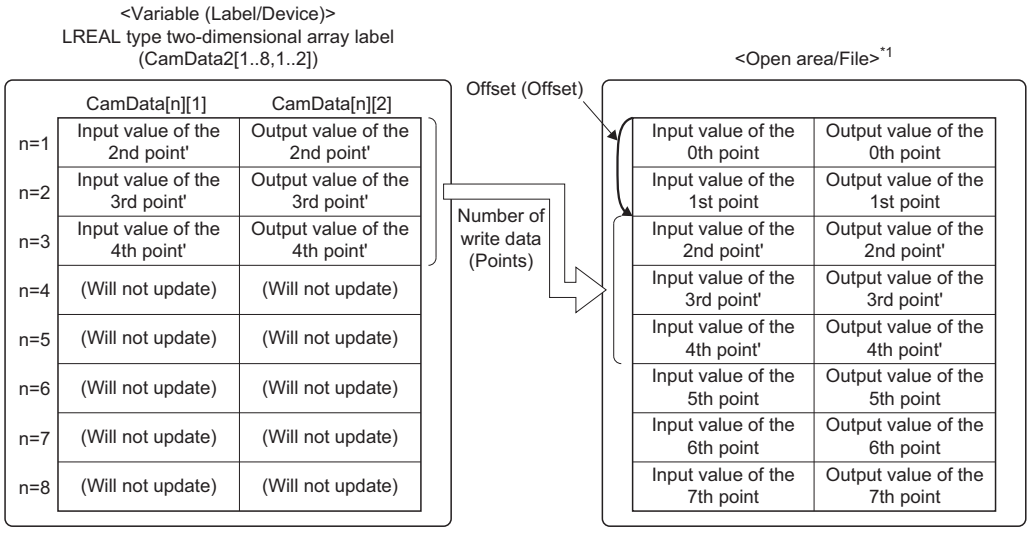
■Offset (Offset) and Number of write data (Points) setting

The following operations are based on the values set in Offset (Offset) and Number of write data (Points).

The contents of the values set for Offset (Offset) and Number of write data (Points) vary depending on the calculation profile format.

The following describes the contents when "cam data" and "rotary cutter" are set to the calculation profile format.

- When the calculation profile format is "cam data"

Specifying interpolation method (Interpolate)	Description
0: Linear Interpolation	Sets the top of the coordinate for write in Offset (Offset), and the number of coordinates in Number of write points (Points). The data stored in Write data 1 (Data1) is the PROFILE_CAM_DATA type data when Specifying interpolation method (Interpolate) is "0: Linear interpolation". The data stored in Write data 2 (Data2) is data defined by using set coordinate data (a pair of input value (current value per cycle) and output value (stroke)) in LREAL type two-dimensional array. <Example> Write result of Write data 2 (Data2) when Offset (Offset) is set to "2" and Number of write data (Points) is set to "3". (When the variable set to Write data 2 (Data2) is a "LREAL[1..8, 1..2]" data type "CamData2" label.)  *1: Depends on Read destination (Target)
1: Specifying for each section 2: Spline Interpolation	Specifying a section for write is not possible. Sets Offset (Offset) and Number of write data (Points) to "0", and writes the entire calculation profile. When these are set to a setting other than "0", a "Number of Offset/Read/Write Data Incorrect (error code: 3465H)" error occurs. The data stored in Write data 1 (Data1) is the PROFILE_CAM_DATA type data when Specifying interpolation method (Interpolate) is "1: Specifying for each section" or "2: Spline Interpolation". The data stored in Write data 2 (Data2) is section data defined in PROFILE_CAM_ELEMENT type array.

- When the calculation profile format is "rotary cutter"

  Specifying a section for write is not possible. This format sets the Offset (Offset) and Number of write data (Points) to "0", and writes the entire calculation profile. The setting input manually for Offset (Offset) and the Number of write data (Points) is ignored. The data stored in Write data 1 (Data1) is data where the calculation profile format is PROFILE_ROTARY_CUTTER type "rotary cutter". Write data 2 (Data2) is not used.

■Write data 1 (Data1)/Write data 2 (Data2)

Set write data of the calculation profile with a TARGET_REF structure. The [VAR] and [DEV] data types can be used.

For details of the TARGET_REF structure, refer to the following.

☞ Page 64 TARGET_REF (Input signal)

For data types, write data structures of each calculation profile format can be used.

- Write data 1 (Data1)

  The calculation profile data formats and data types set in Write data 1 (Data1) are shown below.

Calculation profile data format	Data type	Reference
Cam data	PROFILE_CAM_DATA	☞ Page 68 PROFILE_CAM_DATA
Cam data for rotary cutter	PROFILE_ROTARY_CUTTER	☞ Page 71 PROFILE_ROTARY_CUTTER

Ex.

When writing by using a device or label

· When writing a calculation profile of the cam data format with buffer memory (G11480000 or more)

Data1.Target := "[DEV](PROFILE_CAM_DATA)G11480000"

· When writing a PROFILE_CAM_DATA type label (CamData1)

Data1.Target := "[VAR]CamData1"

- Write data 2 (Data2)

  The calculation profile formats and data types set in Write data 2 (Data2) are shown below. The data type of the structure that is set varies depending on the Write destination (Target) and Specifying interpolation method (Interpolate) of the target cam data.

○: Supported, ×: Not supported

Calculation profile format	Data type	Specifying interpolation method (Interpolate)			Reference
		0: Linear interpolation	1: Specifying for each section	2: Spline interpolation	
Cam data	LREAL[ ]*1	○	×	×	—
	PROFILE_CAM_ELEMENT[ ]*2	×	○	○*3	☞ Page 70 PROFILE_CAM_ELEMENT
Cam data for rotary cutter	Setting is unnecessary. If set, the setting is ignored.				—

*1 The LREAL type two-dimensional array is set as follows.
LREAL[m..n, o..p]

Item	Description
Number of elements (n-m+1)	Sets the array elements so that they are greater than the number of points that are to be read. When writing the entire calculation profile (Offset (Offset), Number of write data (Points) are set to "0"), set this to the cam data resolution +1/number of coordinates or more. However, an area bigger than the resolution +1/number of coordinates in Write data 1 (Data1) will not be updated. If the ProfileControl (Calculation profile control) version is "earlier than 1.19", it is necessary to match the number of elements of the specified array to the cam data resolution +1/number of coordinates.
Number of dimensions (p-o+1)	Writes the stroke and current position per cycle for each point. Set the array element to "2". If the array element is set to "3" or more, a write is possible, but as the intended result of the write may not be obtained due to data being moved forward, make sure to set the array element to "2".

*2 The PROFILE_CAM ELEMENT array is set as follows.
PROFILE_CAM_ELEMENT[m..n]

Item	Description
Number of elements (n-m+1)	Sets the array elements so that they are greater than the number of points that are to be read. When reading the entire calculation profile (Offset (Offset), Number of read data (Points) are set to "0"), set this to the cam data Number of sections (NumberOfSections) or more. However, an area bigger than the Number of sections (NumberOfSections) in Write data 1 (Data1) will not be updated. If the ProfileControl (Calculation profile control) version is "earlier than 1.19", it is necessary to match the number of elements of the specified array to the cam data Number of sections (NumberOfSections).

*3 Only used by the end point and stroke.

The following errors in ProfileControl (Calculation profile control) may occur depending on the version used.

ProfileControl (Calculation profile control)	Description
Version earlier than "1.19"	When the number of elements of the specified array does not match the resolution+1/number of coordinates, a "Number of Read/Write Data Mismatch (error code: 3464H)" occurs.
Version "1.19" and later	If the number of the elements of the specified array is smaller than the resolution+1/number of coordinates when writing the entire calculation profile, a "Number of Read/Write Data Mismatch (error code: 3464H)" occurs.

Ex.

For LREAL type

Specifying interpolation method (Interpolate)	Example setting
0: Linear Interpolation	The following shows examples of setting the two-dimensional device/label coordinate number "100" to Target (Target) of the TARGET_REF structure. ■Buffer memory specification Data2.Target := "[DEV](LREAL[1..100,1..2])G11483000" ■Label specification Data2.Target := "[VAR]CamData2" *: CamData2 declares the "LREAL[1..100,1..2]" label to data type and label name of the global label to "CamData2", and specifies the character string for Target (Target). The write data is as shown below. ・CamData2[n,1]: nth point current position per cycle ・CamData2[n,2]: nth point stroke
1: Specifying for each section 2: Spline Interpolation	The following shows examples of setting the two-dimensional device/label resolution "100" to Target (Target) of the TARGET_REF structure. ■Buffer memory specification TARGET_REF.Target := "[DEV](LREAL[1..101,1..2])G11483000" ■Label specification TARGET_REF.Target := "[VAR]CamData2" *: CamData2 declares the "LREAL[1..101,1..2]" label to data type and label name of the global label to "CamData2", and specifies the character string for Target (Target). The write data will be as shown below. ・CamData2[1,1]: Value set for the start point ・CamData2[1,2]: Value set for the initial stroke amount ・CamData2[n+1,1]: nth point current position per cycle ・CamData2[n+1,2]: nth point stroke *: When specifying labels, local labels can also be set. However, when setting local labels, specify the "POU name" to the "@target modification". (☞ Page 64 TARGET_REF (Input signal))

Ex.

For PROFILE_CAM_ELEMENT type

Example setting
The following shows examples of setting the two-dimensional device/label number of sections "10" to Target (Target) of the TARGET_REF structure. ■Buffer memory specification Data2.Target := "[DEV](PROFILE_CAM_ELEMENT[1..10])G11483000" ■Label specification Data2.Target := "[VAR]CamData2" *: CamData2 declares the "PROFILE_CAM_ELEMENT[1..10]" label to data type and label name of the global label to "CamData2", and specifies the character string for Target (Target). The write data is as shown below. ・CamData2[n]: nth point section data (PROFILE_CAM_ELEMENT type) *: When specifying labels, local labels can also be set. However, when setting local labels, specify the "POU name" to the "@target modification". (☞ Page 64 TARGET_REF (Input signal))

Point

- If the type of the specified data structure for writing does not match the type that is required for each calculation profile, "Read/Write Data Type Incorrect (error code: 3462H)" occurs.
- For details of the cam data for the calculation profile set in Write data 1 (Data1)/Write data 2 (Data2), refer to the following.
  📖MELSEC iQ-R Motion Module User's Manual (Application)
- If the Write destination (Target) of MCv_WriteProfileData (Profile Write) is set to "1: File" with an LREAL type array and then written, the cam data format of Specifying interpolation method is output to a file as "0: Linear Interpolation".

■Write destination (Target)

Sets the write destination of the calculation profile, and writes Write Data 1 (Data1)/Write Data 2 (Data2) and the Profile ID (PROFILE_ID) of the calculation profile open FB.

Setting value	Description
0: Open area	• The Profile ID (ID) value of the PROFILE_DATA structure is referred, and the write is performed for the calculation profile that Profile ID No. (Number) of the PROFILE_ID structure indicates in the open area. ■When "0" is set in Profile ID No. (Number) of the PROFILE_ID structure An available ID is assigned automatically, and the ID is stored in profile ID No. (Number). If an FB with Profile ID No. (Number) specified as "0" is executed multiple times to the same calculation profile, the data will be opened in multiple different open areas. When no ID is available, "Insufficient Calculation Profile ID (error code: 3452H)" will occur and open processing will not be performed. ■When "1 to 60000" is set in Profile ID No. (Number) of the PROFILE_ID structure The set ID is opened without changing the ID. If setting an ID that has been already opened, the open data will be overwritten.
1: File	• The Profile ID (ID) value of the PROFILE_DATA structure is referred, and the write is performed for the calculation profile in the open area that the Profile ID No. (Number) indicates. Following this, the Calculation profile data storage location (Location) value of the PROFILE_DATA structure is referred, and the write is performed for the calculation profile that the File name (Filename) and Folder specification (Path) indicate. With the Profile ID of the calculation profile, the Profile ID (ProfileID) value of the output variable will be written in the file. • If the specified file exists, the file is overwritten. • If the specified file does not exist, a new file will be created and written in "Unicode (UTF-16 Little Endian with BOM)" if the value in both Offset (Offset) and Number of write data (Points) is "0". • When creating a new file, both Offset (Offset) and Number of write data (Points) must be "0", otherwise "Number of Offset/Read/Write Data Incorrect" (error code: 3465H) occurs. (Write to the open area is executed.) • When set to "1: File", set Offset (Offset) and Number of write data (Points) to "0". If a value other than "0" is set, "Number of Offset/Read/Write Data Incorrect" (error code: 3465H) occurs.

Point

- When Write destination (Target) is "1: File", processing is performed at a cycle with lower priority than the operation cycle. Therefore, it may take time to write depending on the processing details and calculation profile.
- When Write destination (Target) is "0: Open area", the write is performed according to the cycle of the executed program.

■Execution mode (ExecutionMode)

Sets the timing for executing MCv_WriteProfileData (Profile Write).

Setting value	Description
0: Execute immediately (mcImmediately)	Immediately writes the contents of the open area. The writing may affect the control in execution. Note that if the calculation profile format and resolution do not match when performing a write during FB execution, "Calculation Profile Operating (error code: 3411H)" occurs.
1: Execute at completion (mcQueued)	Writes after waiting for the FB being executed to complete. When multiple FBs are waiting, the FBs will be executed in order of task priority. FBs of the same priority are executed according to the start order.
3: Execute speculatively (mcSpeculatively)	"Calculation Profile Operating (error code: 3411H)" occurs and the open area is not changed.

Point

When no FB is in execution, the FB is executed immediately regardless of the setting of Execution mode (ExecutionMode). When FBs in execution are accessing the open area, Executing (Busy) of each FB accessing the open area becomes TRUE. Use Execution mode (ExecutionMode) for the interlock as needed.

Precautions

- The number of data points that can be written at the same time are limited. If the write is not completed in a single FB execution, execute the write over several FB executions.
- When the calculation profile data is written in the open area or file area using MCv_WriteProfileData (Profile Write) with the ProfileControl (Calculation profile control) version being "1.9 or earlier", the setting values of Periodic (Periodic), Master axis absolute coordinate (MasterAbsolute) and Slave axis absolute coordinate (SlaveAbsolute) will become FALSE. When changing setting values, follow the procedures below.

1. Write the calculation profile data to the file area.
2. Specify the file written in step 1 using MC_CamTableSelect (Cam Table Selection), and set Periodic (Periodic), Master axis absolute coordinate (MasterAbsolute), and Slave axis absolute coordinate (SlaveAbsolute).

Program example

In the following program example, "LREAL type (two-dimensional array) format" cam data 1 (ProfileData0001) and "PROFILE_CAM_ELEMENT format" cam data 2 (ProfileData0002) are written to the file.

■Calculation profile

Profile ID	Label name	Data type	Comment
1	ProfileData0001	MC_CAM_REF	Cam data 1
2	ProfileData0002	MC_CAM_REF	Cam data 2

- Calculation profile (cam data) settings

Item		ProfileData0001	ProfileData0002
Specifying interpolation method		Linear interpolation	Specifying for each section
Resolution		—	256
Length per cycle setting	Unit	degree	degree
	Length per cycle	360.00000	360.00000
Stroke amount	Unit	pulse	%
	Stroke amount	100.0	100.0000000
Cam 1 cycle time		—	1.000[s]
Initial stroke value		—	0.0000000
Cam 1 stroke minimum value		0.00000	—
Cam 1 stroke maximum value		360.00000	—

- ProfileData0001

Section No.	Input value [degree]	Output value [pulse]
1	0.00000	0.0
2	90.00000	60.0
3	180.00000	100.0
4	270.00000	30.0
5	360.00000	0.0

- ProfileData0002

Section No.	Start point [degree]	End point [degree]	Stroke [%]	Cam curve type
1	0.00000	90.00000	60.0000000	Single hypotenuse
2	90.00000	180.00000	100.0000000	Single hypotenuse
3	180.00000	270.00000	30.0000000	Single hypotenuse
4	270.00000	0.00000	0.0000000	Single hypotenuse

■Labels used

Label name	Data type	Comment
MCv_WriteProfileData_1	MCv_WriteProfileData	Profile Write FB 1
WriteData1_1	TARGET_REF	Write data 1 Profile 1
WriteData2_1	TARGET_REF	Write data 2 Profile 1
WriteData1_Lreal	PROFILE_CAM_DATA	Profile data Cam data 1
WriteData2_Lreal	Double-precision real number(0..4,0..1)	Profile data 1 LREAL type
bWriteProfile1	Bit	Profile write command 1
bDone1	Bit	Completion 1
bBusy1	Bit	Executing1
bError1	Bit	Error
uwErrorID1	Word [unsigned]/bit string [16-bit]	Error code
ProfileID1	PROFILE_ID	Profile ID 1
ProfileData0001*1	PROFILE_DATA	Profile data 0001
MCv_WriteProfileData_2	MCv_WriteProfileData	Profile Write FB 2
WriteData1_2	TARGET_REF	Write data 1 Profile 2
WriteData2_2	TARGET_REF	Write data 2 Profile 2
WriteData1_Element	PROFILE_CAM_DATA	Profile data Cam data 2
WriteData2_Element	PROFILE_CAM_ELEMENT(0..3)	Profile data 2 PROFILE_CAM_ELEMENT type
bWriteProfile2	Bit	Profile write command 2
bDone2	Bit	Completion 2
bBusy2	Bit	Executing2
bError2	Bit	Error 2
uwErrorID2	Word [unsigned]/bit string [16-bit]	Error code 2
ProfileID2	PROFILE_ID	Profile ID 2
ProfileData0002*1	PROFILE_DATA	Profile data 0002

*1 Only used in FBD/LD programs.

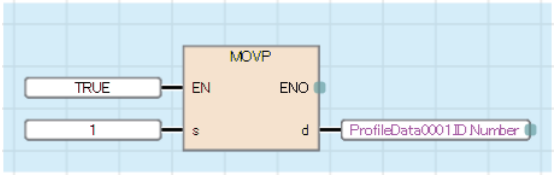
■Global labels used

Label name	Data type	Comment	Public label
G_ProfileCamData*1	PROFILE_CAM_DATA	Profile cam data	Enabled
G_ProfileCamELEMENT*1	PROFILE_CAM_ELEMENT	Profile cam element	Enabled

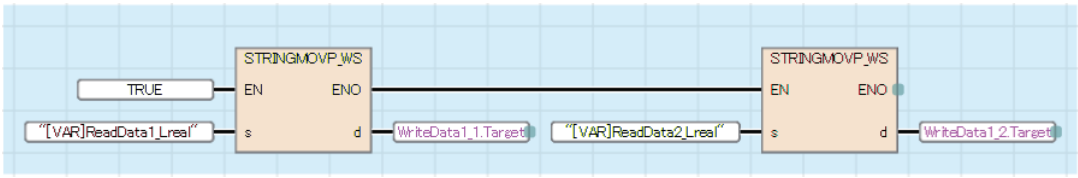
*1 Only used in FBD/LD programs.

■FBD/LD program (CPU module side)

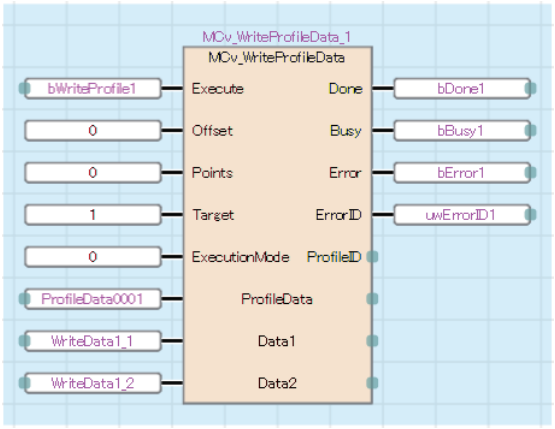
• Profile data 0001



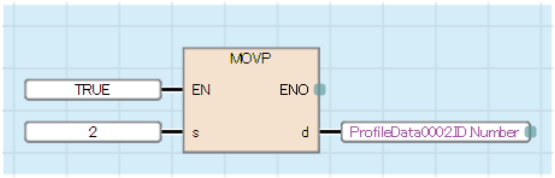
• Write data 1



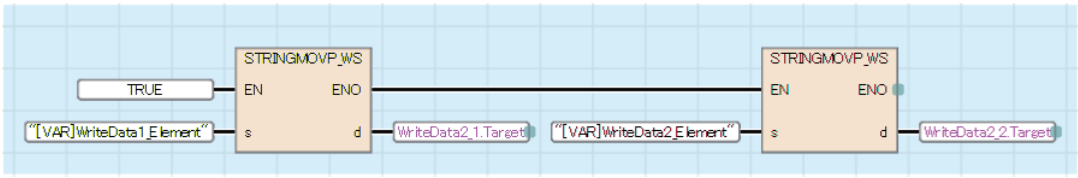
• Profile Write 1 (writes cam data in the LREAL type (two-dimensional array) format)



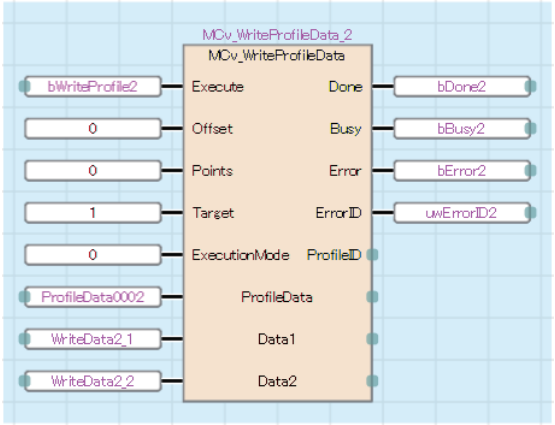
• Profile data 0002



• Write data 2



• Profile Write 2 (writes cam data in the PROFILE_CAM_ELEMENT type format)



■ST program (Motion module side)

```
//-----Write data 1-----
WriteData2_1.Target := "[VAR]WriteData1_Lreal";
WriteData2_2.Target := "[VAR]WriteData2_Lreal";

//-----Profile Write 2-----
//-----Writes cam data in the LREAL type (two-dimensional array) format-----
MCv_WriteProfileData_1(
   ProfileData:= ProfileData0001.ProfileData ,
   Data1:= WriteData1_1 ,
   Data2:= WriteData2_1 ,
   Execute:= bWriteProfile1 ,
   Offset:= 0 ,
   Points:= 0 ,
   Target:= 1 ,
   ExecutionMode:= MC_EXECUTION_MODE__mcImmediately ,
   Done=> bDone1 ,
   Busy=> bBusy1 ,
   Error=> bError1 ,
   ErrorID=> uwErrorID1 ,
   ProfileID=> ProfileID1
);

//-----Write data 2-----
WriteData1_1.Target := "[VAR]WriteData1_Element";
WriteData1_2.Target := "[VAR]WriteData2_Element";

//-----Profile Write 2-----
//-----Writes cam data in the PROFILE_CAM_ELEMENT type format-----
MCv_WriteProfileData_2(
   ProfileData:= ProfileData0001.ProfileData ,
   Data1:= WriteData1_2 ,
   Data2:= WriteData2_2 ,
   Execute:= bWriteProfile2 ,
   Offset:= 0 ,
   Points:= 0 ,
   Target:= 1 ,
   ExecutionMode:= MC_EXECUTION_MODE__mcImmediately ,
   Done=> bDone2 ,
   Busy=> bBusy2 ,
   Error=> bError2 ,
   ErrorID=> uwErrorID2 ,
   ProfileID=> ProfileID2
);
```

4 MOTION MODULE PROGRAMS

This section describes the programs used with the Motion module.

4.1 Motion Module Programs

This section describes the program configuration required to create programs and its contents.

Supported languages

The Motion system can be controlled using the following methods.

- Method that controls from the CPU module
- Method that controls using the Motion system internal program language

The following language can be used for the Motion system internal program language.

- Structured text (ST language)

Variables

The variables used in programs fall into two categories, "Device" and "Label".

Areas related to axis control, such as parameters, monitor data, and control commands, are assigned to the labels (structure) provided by the system.

Type	Description
Device	Variables expressed as a name and number. The name, type, and method of use are defined with the system. ■Devices that are usable by the Motion system • Buffer memory (G) • Remote input (RX) • Remote output (RY) • Remote register (RWw, RWr) • Link special relay (SB) • Link special register (SW)
Label	Variables expressed with a given character string. The label defined with the system for the name, type, and method of use as well as the label defined by the user are available.

Precautions

■Inputting project data

While the power is ON, reload of areas such as label definitions, initial values, and programs is not performed. To reflect changes in the written project, turn the power from OFF to ON.

■Calling Motion control FB instances

Calling a single Motion control FB instance from multiple programs can cause unforeseen operations to occur.

■Character string type label inconsistency

When writing is performed for a character string type label, performing reading/writing from tasks with different execution cycles or from an external device, inconsistency may occur producing an unintended operation result. Correct the accessed cycles so that they are the same or set an interlock when accessing.

■Array elements

When accessing the element defined in an array, access it within the range of the number of elements.

If a constant out of the range defined for the array index is specified, an error will occur when converting the program. If the array index is specified with data other than a constant, an error will not occur when converting the program and processing will be performed by accessing another device.

■Axis type/Axes group type local labels

Defining an axis type or axes group type local label can cause unforeseen operations to occur.

■Device Usage Restrictions

For information about restrictions when using devices, refer to "Direct access from motion built-in program to link devices" in the following manual.

📖 MELSEC iQ-R Motion Module User's Manual (Network)

■Program size limits

The program size limit differs depending on the capacity setting of System memory (RAM) set in Addon Program_ST parameter (Addon_Program_ST). If the system memory is insufficient, "Insufficient Memory Capacity (error code: 350DH)" occurs and the program will not RUN. In this case, change the setting value of System memory (RAM).

Point

The contents of the program file are opened in the internal memory at STOP→RUN. In general, the available area in System memory (RAM) must be equal to 4 to 7 times the program file size (differs depending on the program contents).

4

Execution of programs

The program changes to RUN status when "PLC READY (Y0)" turns ON and the program is executed in accordance with the program execution type and execution procedure settings.

Execution types for programs

Set the timing that program processing is executed for each execution type.

Settable execution types are shown below.

Execution type	Reference
Initial	☞ Page 431 Initial execution type
Normal	☞ Page 431 Normal execution type
Fixed scan	☞ Page 432 Fixed scan execution type
Standby	☞ Page 433 Standby execution type

■Initial execution type

The program is executed only once when "PLC READY (Y0)" changes from OFF→ON.

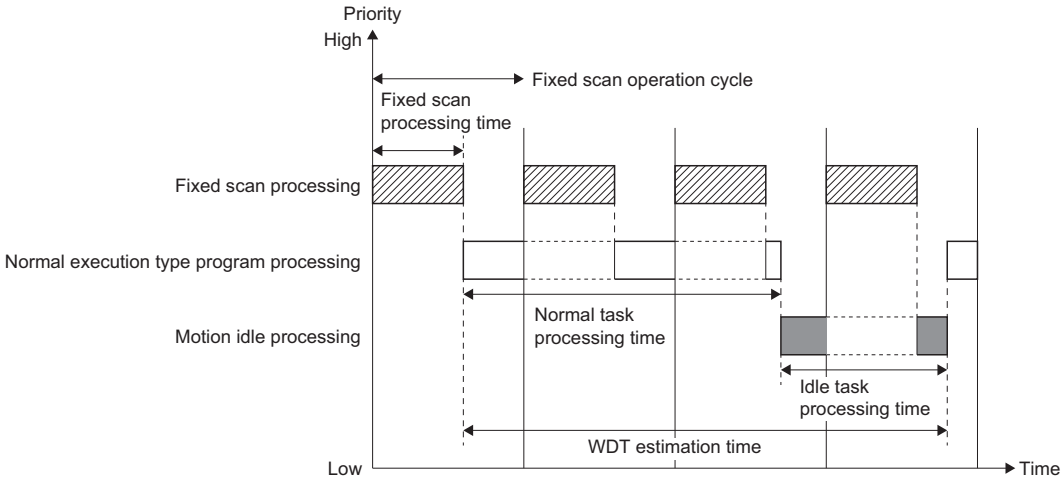
After the initial execution type program completes, a program of a different execution type is executed.

Restriction

For initial execution type programs, do not use a command that requires multiple execution cycles (a command containing a complete device) or a Motion control FB until the program completes.

■Normal execution type

The program is executed by normal task through the Motion system.



Restriction

- While running fixed scan execution type programs, the execution of normal execution type programs is interrupted. However, fixed scan execution type programs can be disabled through a normal execution type program with a Disable Interrupt (DI) instruction. After being disabled through DI instruction, fixed scan execution type programs will remain disabled until an Enable Interrupt (EI) instruction is executed.
  The enabled/disabled status of fixed scan execution type programs can be checked with "EI Flag (System.Md.Program_Ei)".
- When the execution interval of a normal execution type program exceeds 1.0[s], a WDT error occurs.

■Fixed scan execution type

The program is executed at each specified time. The time can be set from that of the first operation cycle up to 60000[ms] (in integer multiples of the first operation cycle).
Processing is prioritized by execution interval length and starts by processing programs with the shortest intervals.

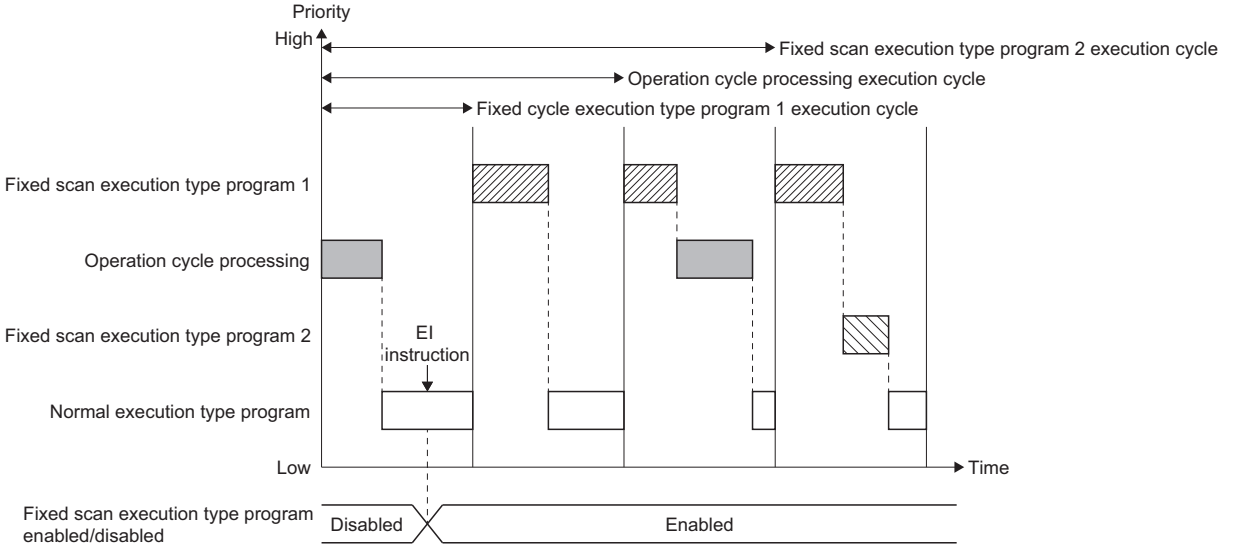
Fixed scan interval	Processing order
Fixed scan interval = Operation cycle	Waits for program execution to complete, then performs operation cycle processing. If the fixed scan timing is during operation cycle processing, the program execution and operation cycle processing are executed simultaneously (in parallel).
Fixed scan interval > Operation cycle	Program execution and operation cycle processing are executed simultaneously (in parallel).

Up to 16 types of fixed scan intervals can be set. Adding fixed scan intervals affects the operation cycle regardless of the length. Because of this, the operation cycle may be exceeded even if there are not 16 types of fixed scan intervals if the operation cycle is short (62.5μs or less).
Program setting numbers with the same interval have no limit.

Point

To execute fixed scan execution type programs, the status must be set to execution enabled status by EI instruction.

Executes the fixed scan program at the start of the next fixed scan after the EI instruction is issued.



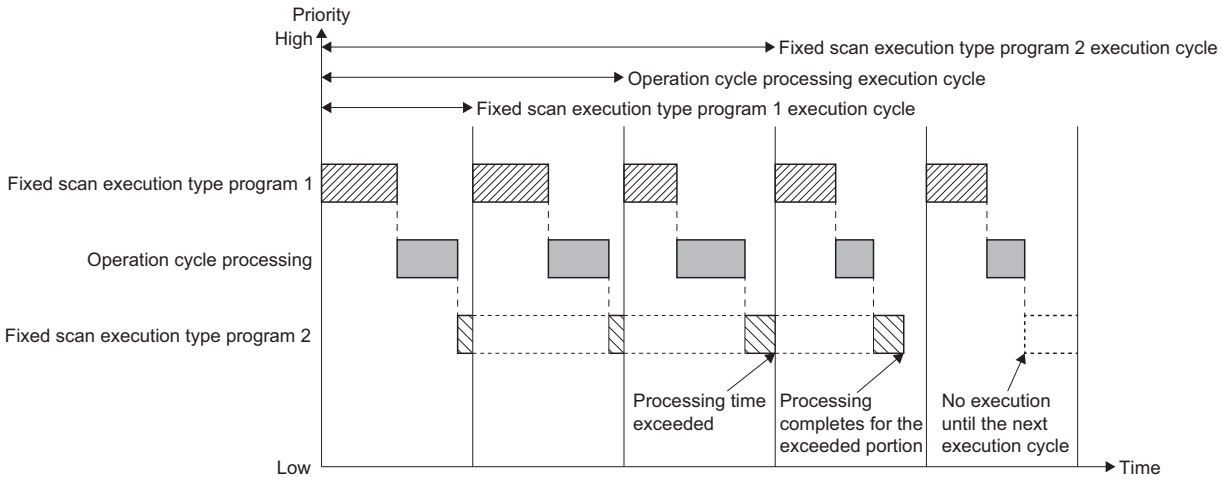
Restriction

Fixed scan type programs cannot be executed if fixed scan execution type programs have been disabled (DI) from a normal execution type program.

- Interruption of processing

  Other high-priority processing can interfere with program execution processing in some cases.

  If processing interference occurs due to interruption while running a fixed scan execution type program, the processing will resume after the interrupt processing completes. In addition, all further execution cycles are ignored until the processing completes if the processing time is exceeded for the fixed scan interval. Execution resumes from the next execution cycle after the processing completes.



Restriction

A warning or error does not occur even if the processing time of a program with a fixed scan interval longer than the operation cycle is exceeded.

When necessary, monitor Execution time (ExecutionTime) of the PROGRAM_INFO structure with the engineering tool.

Point

In some cases, the operation load of fixed scan operation type programs can cause an operation cycle over to occur.

■Standby execution type

The program is executed at execution request.

The program is converted to a normal execution type program and then executed by using a PSCAN instruction from another running program.

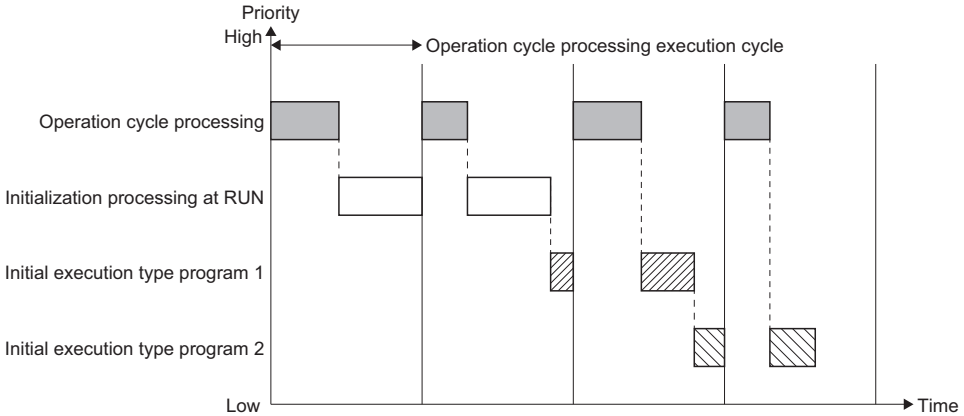
Flow of program execution

The flow of execution for each program is shown below.

■Processing for initial execution type programs when "PLC READY (Y0)" is ON

The timing chart below shows the processing executed at "PLC READY (Y0)" ON.

Processing for initial execution type programs is interrupted by operation cycle processing.



When the PLC READY (Y0) is turned from OFF to ON

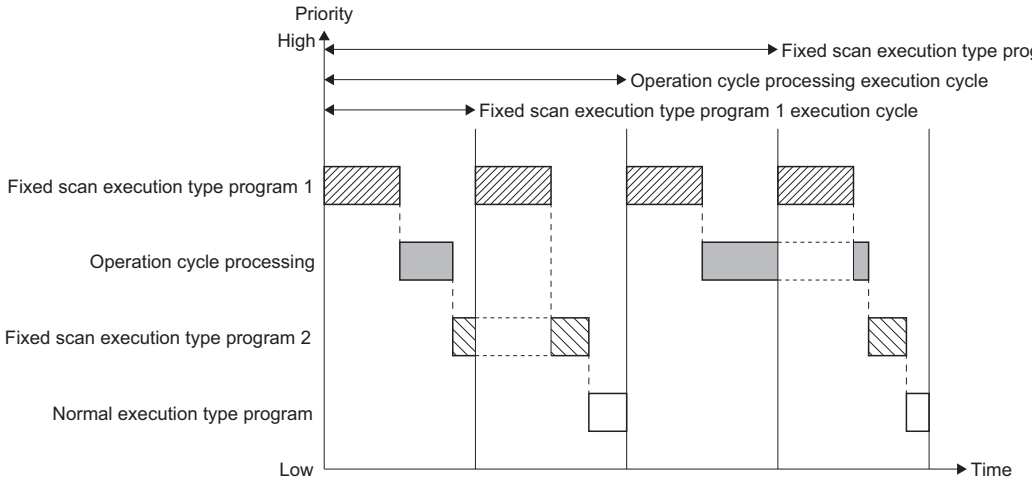
Point

Fixed scan type programs are not executed until the processing for initial execution type programs completes, even if an EI instruction is issued during an initial execution type program.

■Processing at completion of initial execution type programs

The timing chart for each program that operates after initial execution type program processing has fully completed is shown below.

The 3 kinds of fixed cycle processing (fixed scan execution type program 1, fixed scan execution type program 2, and operation cycle processing) are performed with priority given to those with the shortest execution cycles.



Point

Programs of the same execution type/with the same execution cycle settings are executed in order starting with those with the smallest execution order number.

The execution order can be checked with the engineering tool in "Program Execution Setting".

The execution type/execution cycle setting have higher priority than the execution order.

■Initialization processing at power ON

The following processing is performed during label initialization processing when the power turns from OFF to ON.

- Expansion of global label files, structure definition files, and FB definition files
- Configuration and setting of initial values for global labels
- Input of program settings

■Initialization processing at first RUN

The following processing is performed during first RUN initialization processing after the power turns from OFF to ON.

- Expansion of program files and configuration of program local labels.
- Checking of each parameter and program.
- Setting of initial values for labels/FB.

Point

The time required to check parameter settings, programs, and files varies by volume and may cause a delayed response from peripheral devices in some cases.

■Initialization processing at 2nd and subsequent RUN

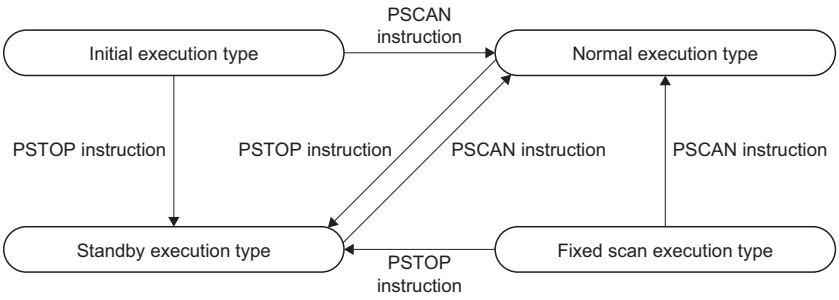
Reinput of each file and label reconfiguration cannot be performed after the power turns from OFF to ON during the 2nd and subsequent RUN. Only processing for the initial value setting is performed.

Refer to label initialization function for details of the initial values that are set. (☞ Page 437 Label initialization function)

Changing the execution type

To change the program execution type while "PLC READY (Y0)" is ON, use the program control command shown below.

- Program scan execution registration (PSCAN)
- Program standby (PSTOP)



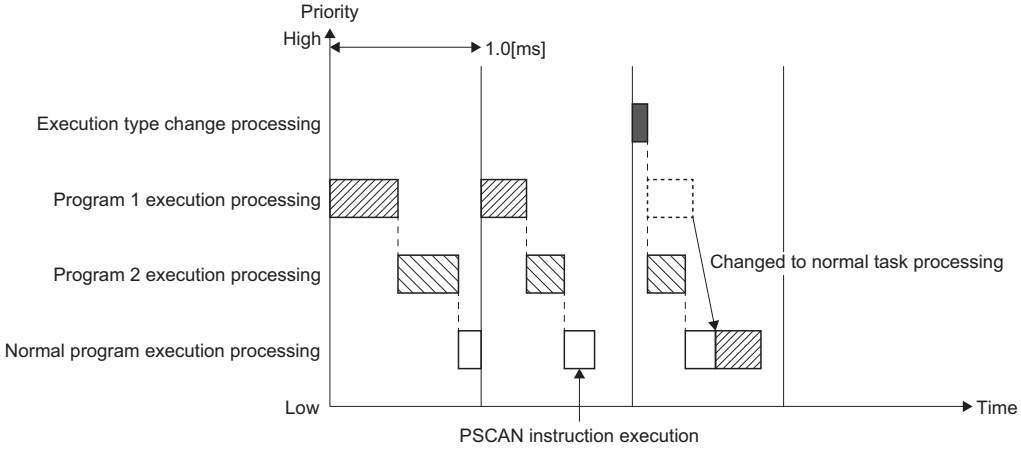
■Timing for changing the execution type

The timing for changing the execution type is shown below.

Execution type	Executed instruction	
	PSCAN	PSTOP
Initial	Changes to normal execution type.	Changes to standby type after program END processing completes.
Normal	No operation.	
Fixed scan	Changes to normal execution type after program END processing completes.	
Standby	Changes to normal execution type.	No operation.

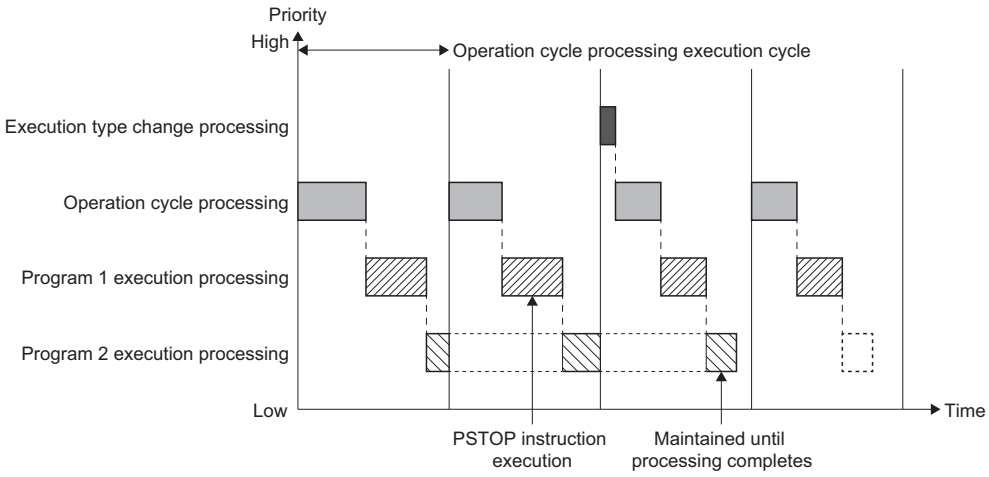
Ex.

When there are 2 fixed scan execution type programs and PSCAN instruction is executed for program 1.



Ex.

When there are 2 normal execution type programs and PSTOP instruction is executed for program 2.



■Precautions

- When changing the execution type of an initial execution type program or a fixed scan execution type program, it is not possible to revert back to the original execution type.
- When multiple instructions are executed in a single program, the last instruction determines the operation.
- The change in execution type is not reflected until processing for the relevant program completes.

Operation at error occurrence

When a stop error occurs during command execution, "Ready (X0)" turns ON→OFF and the program execution stops.
For the following errors, the operation (stop/continue) performed when an error occurs can be set in "Operation setting at error[*1]" in Motion Control Setting Function.

- Invalid device/label/buffer memory specification
- File name specification
- Operation error

*1 "Operation setting at error" is set in basic settings in Motion Control Setting Function. (☞ Page 444 Basic settings)

Label initialization function

For the Motion system, label initialization processing is performed when the power turns OFF→ON and also at STOP→RUN. The execution details for initialization processing are shown below.

Label		Initial value setting provided/ none	Operation for label initialization		
			Power OFF→ON	When STOP→RUN	
				First time	2nd and subsequent time
Motion label	Parameter	—	Set the initial value	No operation (value is stored)	
	Non-parameter	—	Set the initial value	No operation (value is stored)	
User-defined global label		Provided	Set the initial value	No operation (value is stored)	
		None	Set the initial value (=0)	No operation (value is stored)	
User-defined local label		Provided	— (no labels configured)	Set the initial value	No operation (value is stored)
		None	— (no labels configured)	Set the initial value (=0)	No operation (value is stored)

When performing STOP→RUN 2 or more times through settings, setting of the initial value can be performed.

Item	Description	Setting range	Initial value
Label initial value is reflected at STOP→RUN (2nd and subsequent time)	Specify whether or not to set the label initial value at STOP→RUN (2nd and subsequent time). (Values are always reflected the first time.)	• Enable • Disable	Disable

Label memory

The memory area where label instances are allocated is called the label memory. The maximum label memory size is set in Label memory size (System.PrConst.LabelMemorySize).
The current available free space in the label memory can be checked with label memory free size (System.Md.LabelMemoryFreeSize).

4.2 Motion Module Program Creation

Motion module programs are created using the following methods.

- Use Structured Text (ST) language to create an ST program on the Motion module side. (☞ Page 438 Program creation procedure for ST programs)
- Use labels registered to the Motion module on the CPU module side and create a sequence program using ladder language, FBD/LD language, or ST language. (☞ Page 462 Creation procedure for programs using the public label function)

Program creation procedure for ST programs

This section describes the areas that need to be set for the Motion system using an engineering tool as well as the procedure for creating ST programs.

1. Parameter settings

Set the module configuration and the CPU parameters. (☞ Page 439 Parameter settings)

2. Network settings

Set network related parameters in "Module Parameter (Network)".

Set the servo amplifiers to connect using CC-Link IE TSN in "Network Configuration Settings".

(☞ Page 440 Network settings)

3. Module parameter settings

Carry out the setting of RD78G related operations in "Module Parameter (Motion)". (☞ Page 443 Module parameter settings)

4. Module extended parameter settings

Carry out the setting of RD78G axis control related areas using the "Motion Control Setting Function".

- Basic settings (☞ Page 444 Basic settings)
- System settings (☞ Page 445 System settings)
- Axis (☞ Page 448 Axis)
- Axes group (☞ Page 458 Axes group)

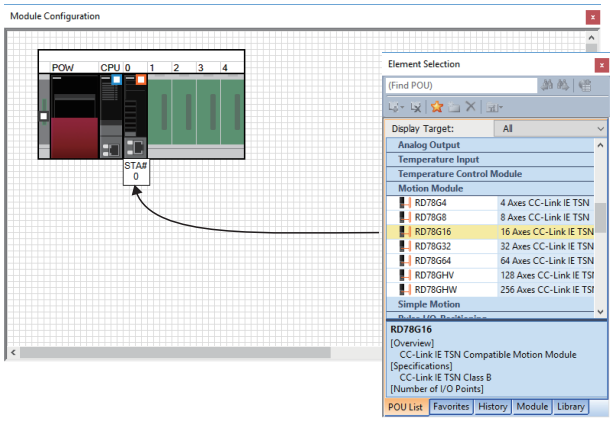
5. ST program creation (☞ Page 460 ST program creation)

Create an ST language program to operate RD78G with using the "Motion Control Setting Function".

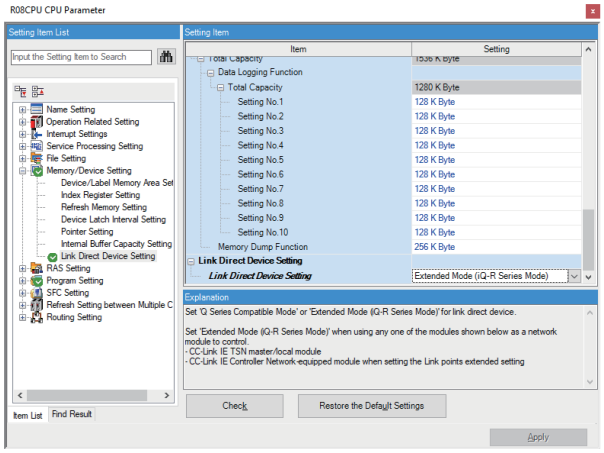
- Inserting Motion control FBs (☞ Page 461 Inserting Motion control FBs)
- Registering labels (☞ Page 461 Registering labels)

Parameter settings

Set the module configuration and the CPU parameters.



1. From the [Navigation window], select [Module Configuration] to open the Module Configuration and add RD78G from the Element Selection window so it can be used.
To fix the module configuration, select [Edit]⇨[Parameter]⇨[Fix] from the menu.

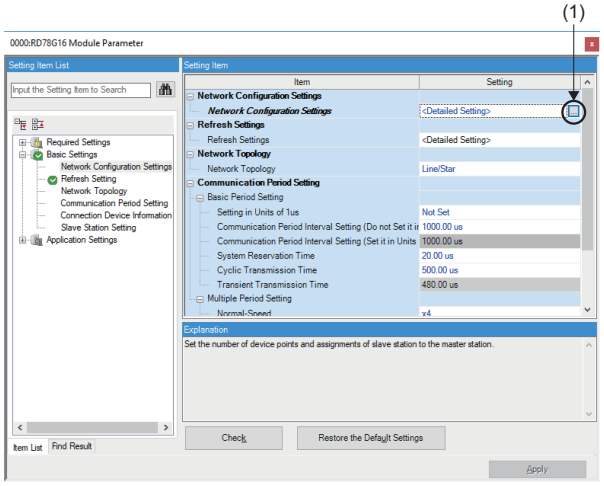


2. From the [Navigation window], select [Parameter]⇨[Module model name*1]⇨[CPU Parameter] to display the parameter editor.
From "Memory/Device Setting"⇨"Link Direct Device Setting", select "Extended Mode (iQ-R Series Mode)" in the pull down list, then click the [Apply] button.

*1: The model name of the set CPU module is displayed.

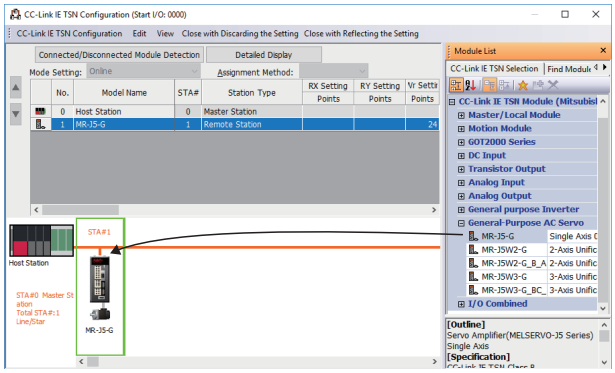
Network settings

Set the module parameters (network).

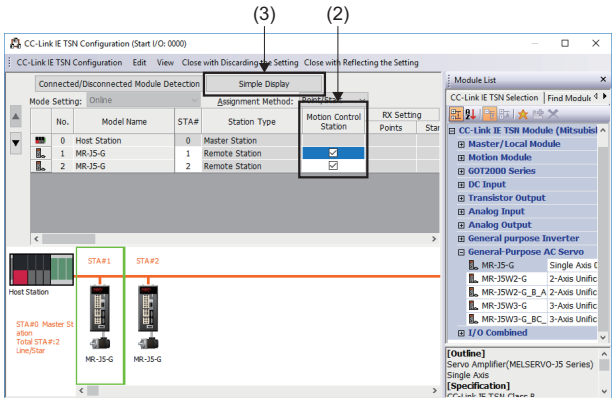


1. From the [Navigation window], select [Parameter]⇨[Module Information]⇨[Module model name*1]⇨[Module Parameter (Network)] to display the parameter editor. Select "Basic Settings"⇨"Network Configuration Settings"⇨"Detailed Setting", then click the [...] button (1).

*1: The model name of the set Motion module is displayed.

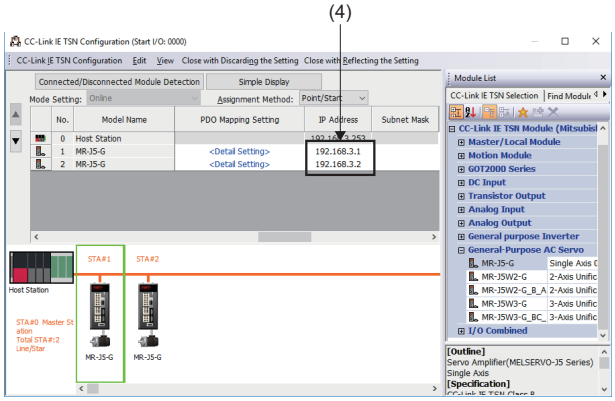


2. The "CC-Link IE TSN Configuration" screen is displayed. From [Module List]⇨[CC-Link IE TSN Device (Mitsubishi Electric Corporation)]⇨"General-Purpose AC Servo", select a servo amplifier and drag and drop it into its assigned position.



3. Place a check in the Motion Control Station (2) check box for each set servo amplifier.

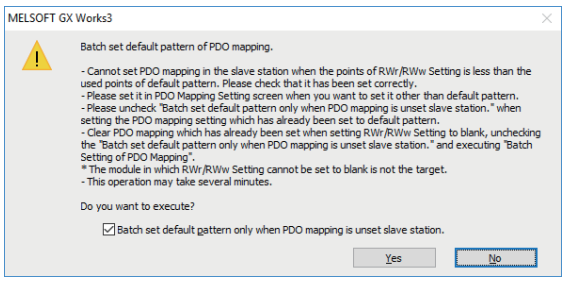
*: If the Motion Control Station is not displayed, click the [Detailed Display] button (3) to switch to the detailed display.



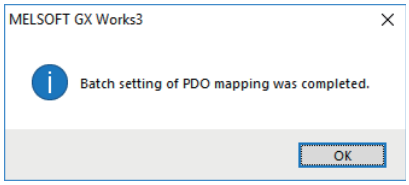
4. Check the IP Address (4) that is assigned to the set servo amplifiers.

*: The IP Address (4) setting is required when creating an axis to set with the Motion Control Setting Function.

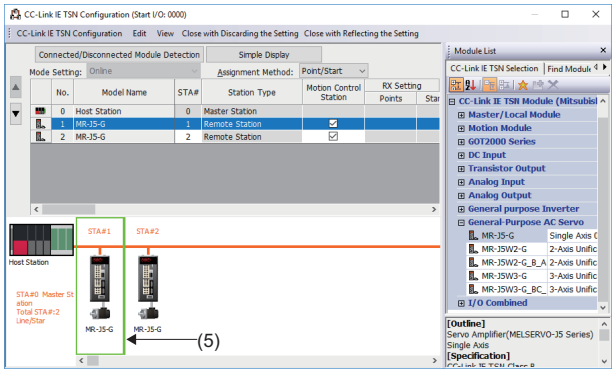
5. When the network configuration is complete, select [CC-Link IE TSN Configuration]⇨[Batch Setting of PDO Mapping] from the menu.



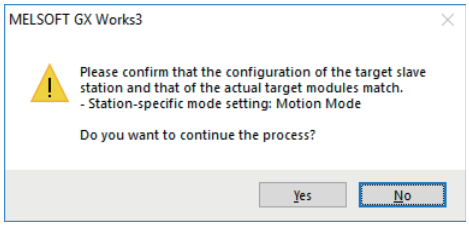
6. When the "Batch set default pattern of PDO mapping." message appears, click the [Yes] button.



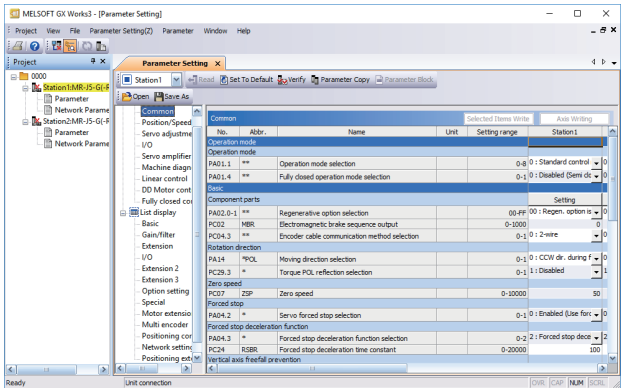
7. The "Batch setting of PDO mapping was completed." message is displayed. Click the [OK] button.



8. From the network configuration, double-click the servo amplifier (5) to be set.

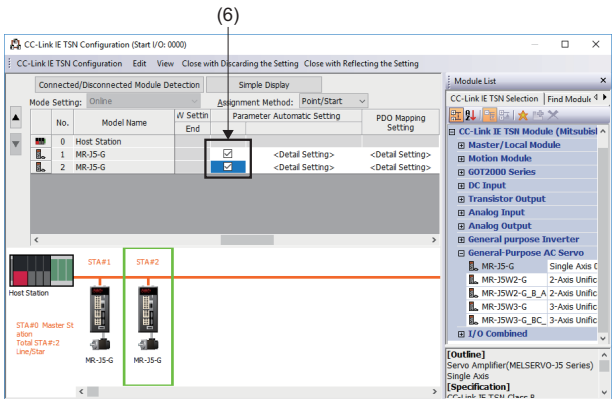


9. When the "Please confirm that the configuration of the target slave station and that of the actual target modules match." message appears, click the [Yes] button.



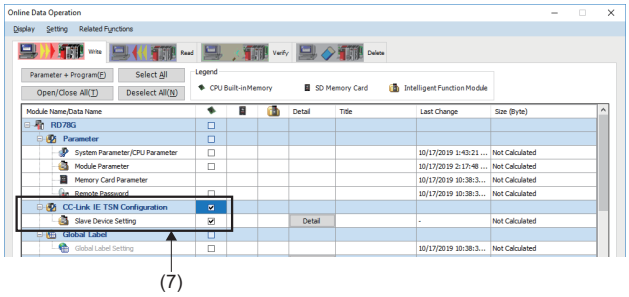
10. The Parameter Setting screen is displayed.
Set the servo parameters for each axis.

11. When the setting of servo parameters is complete, close the Parameter Setting screen.

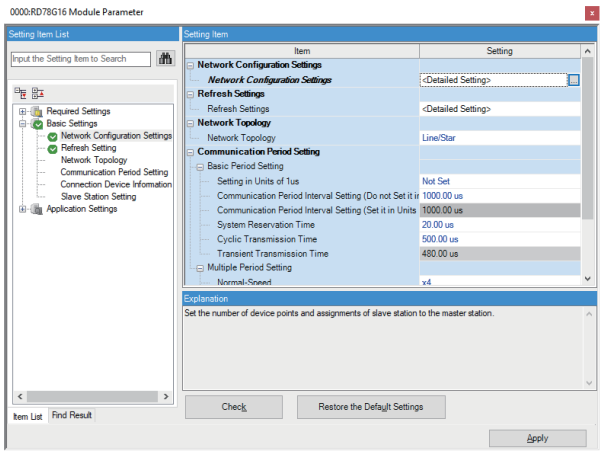


12. Return to the "CC-Link IE TSN Configuration" screen. Place a check in the Parameter Automatic Setting (6) check box.

*: When performing writing of the servo parameters, place a check in "CC-Link IE TSN Configuration" (7) on the "Online Data Operation" screen.



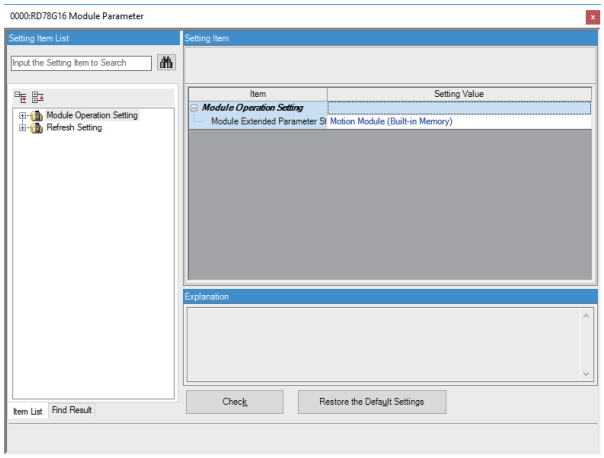
13. Select [Close with Reflecting the Setting], then close the "CC-Link IE TSN Configuration" screen.



14. Click the [Apply] button to complete Module Parameter (Network) settings.

Module parameter settings

Set the module parameters (Motion).



1. From the [Navigation window], select [Parameter]⇨[Module Information]⇨[Module model name*1]⇨[Module Parameter (Motion)] to display the parameter editor.
   Set the parameters in "Module Operation Setting".

*1: The model name of the set Motion module is displayed.

4

Displayed items

■Module operation setting

Item	Setting range	Initial value
Module extended parameter storage target setting	Motion module (built-in memory), Motion module (SD memory card)	Motion module (built-in memory)

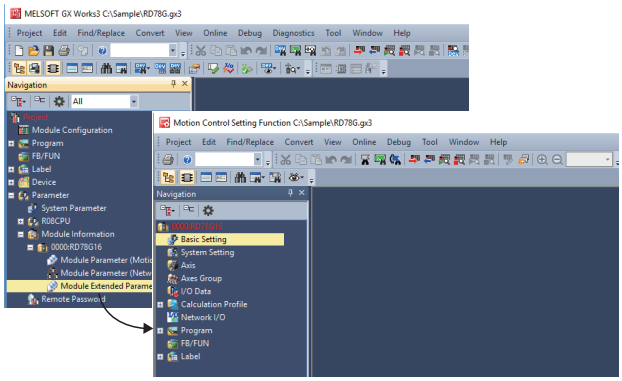
- Module extended parameter storage target setting
  Set the target for module extended parameters to be used for the Motion module. When writing the module extended parameters from GX Works3, write to the target that is specified for this parameter.

Setting	Description
Motion module (built-in memory)	Use the module extended parameter that is written to the flash ROM of the Motion module.
Motion module (SD memory card)	Use the module extended parameter that is written to the SD memory card of the Motion module.

Module extended parameter settings

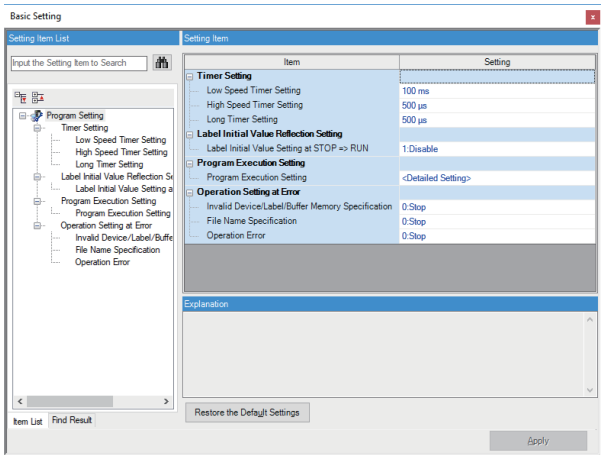
Set the module extended parameters on the Motion Control Setting Function screen.

■Basic settings



1. From the [Navigation window], select [Parameter]⇨[Module Information]⇨[Module model name*1]⇨[Module Extended Parameter] to display the "Motion Control Setting Function" screen.

*1: The model name of the set Motion module is displayed.



2. From the [Navigation window], select [Basic Setting] to display the parameter editor. Set the parameters for the basic settings.
3. When the setting of parameters for the basic settings is complete, click the [Apply] button.

Displayed items

- Program settings

Item		Setting range	Initial value
Timer setting	Low speed timer setting	1 to 10000[ms]	100[ms]
	High speed timer setting	1 to 10000[μs]	500[μs]
	Long timer setting	1 to 1000000[μs]	500[μs]
Label initial value reflection setting	Label initial value setting at STOP→RUN	0: Enable 1: Disable	1: Disable
Program execution setting	Program execution setting	<Detailed setting>*1	—
Operation setting at error	Invalid device/label/buffer memory specification	0: Stop 1: Continue	0: Stop
	File name specification		0: Stop
	Operation error		0: Stop

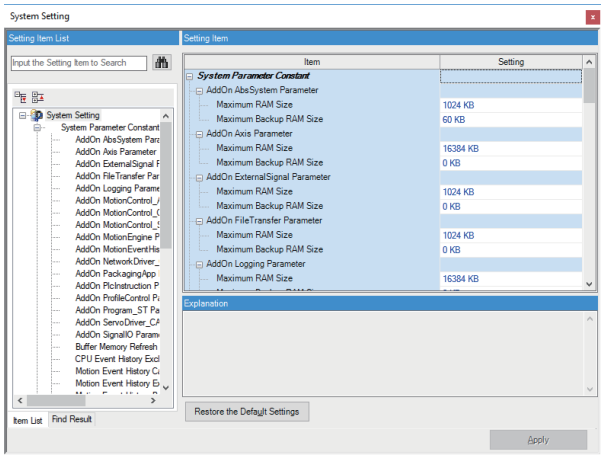
*1 Set when changing the execution order, execution type, and other areas for the programs being used.

Item	Setting description
Execution order	Execute programs in order from 1 to 2048
Program name	The name of the created program is displayed
Execution type	0: Standby 1: Normal 2: Initial 3: Fixed scan

Item			Setting description
Program execution condition*2	Fixed scan interval setting	Unit	1ns, 0.01ms
		Fixed scan interval	■When set to 1ns • RD78G: 62500 to 600000[ns] • RD78GH: 31250 to 600000[ns] ■When set to 0.01ms • 0.25 to 6000[ms]

*2 Can only be set when selecting "3. Fixed scan" as the execution type.

■System settings



1. From the [Navigation window], select [System Setting] to display the parameter editor.
   Set the parameters for the system settings.
2. When the setting of parameters for the system settings is complete, click the [Apply] button.

Displayed items

• System settings

Item [Variable/Structure name]			Setting range	Initial value
System parameter constant	Addon □□ parameter [□□]*1	Maximum RAM size [ADDON_PARAM.RamSizeMax]	—	*1
		Maximum backup RAM size [ADDON_PARAM.BackupRamSizeMax]	—	*1
	Buffer memory refresh cycle setting [System.PrConstBuffermemoryRefreshCycle]	Cycle setting [CYCLE_PARAM.Cycle]	—	0
		Number of cycle over warning detections [CYCLE_PARAM.NumOfCycleOverWngDetectTimes]	0 to 65535	1
		Number of cycle over error detections [CYCLE_PARAM.NumOfCycleOverErrDetectTimes]	0 to 65535	5
		Cycle over error selection [CYCLE_PARAM.CycleOverErrorType]	0: None 1: Warning 2: Minor error 3: Moderate error 4: Major error	3: Moderate error
	CPU event history excluded event [System.PrConst.EventHistoryCpu_Exclude]		—	0x700-0x7ff
	Motion event history capacity [System.PrConst.EventHistoryMotion_Capacity]		1 to 2048[kB]	128[kB]
	Motion event history excluded event [System.PrConst.EventHistoryMotion_Exclude]		—	No setting
	Motion event history path [System.PrConst.EventHistoryMotion_Path]		—	/lch/
	Excluded warning [System.PrConst.ExcludeWarning]		—	No setting
	High-speed mode setting [System.PrConst.FastOperationMode]		50EF: High-speed mode Others: Normal mode	0000

Item [Variable/Structure name]				Setting range	Initial value
System parameter constant	File transfer access control (file in latch drive) [System.PrConst.FileTransfer_AcFile_lch]			Read protected/write protected, Read protected/write permitted, read permitted/write protected, read permitted/write permitted	Read permitted/ write permitted
	File transfer access control (file in RAM drive) [System.PrConst.FileTransfer_AcFile_ram]				Read permitted/ write permitted
	File transfer access control (file in user drive) [System.PrConst.FileTransfer_AcFile_rom]				Read protected/ write permitted
	File transfer access control (file in SD memory card) [System.PrConst.FileTransfer_AcFile_sdc]				Read permitted/ write permitted
	File transfer access control (file in system drive) [System.PrConst.FileTransfer_AcFile_sys]				Read protected/ write permitted
	File transfer access control (label) [System.PrConst.FileTransfer_AcLabel]				Read permitted/ write permitted
	File transfer log capacity [System.PrConst.FileTransfer_LogCapacity]			-1 to 2048[kB]	128[kB]
	Label memory size [System.PrConst.LabelMemorySize]			128 to 8192[kB]	1024[kB]
	Refresh method of send/receive data at Motion control station [System.PrConst.Link_MotionStationRefreshType]			0: Response preferred method 1: Operation cycle preferred method	1: Operation cycle preferred method
	Auto logging enabled [System.PrConst.Logging_AutoLoggingEnable]			0: Disabled 1: Enabled	0: Disabled
	Measurement flag for program information [System.PrConst.MeasuresPrograms]			0: Do not measure 1: Measure	1: Measure
	For manufacturer setting [System.PrConst.MultiThreadMode]			—	0000
	Operation cycle setting [1] [System.PrConst.OperationCycle[1].Cycle]	Cycle setting [CYCLE_PARAM.Cycle]		—	0
		Number of cycle over warning detections [CYCLE_PARAM.NumOfCycleOverWngDetectTimes]		0 to 65535	1
		Number of cycle over error detections [CYCLE_PARAM.NumOfCycleOverErrDetectTimes]		0 to 65535	5
		Cycle over error selection [CYCLE_PARAM.CycleOverErrorType]		0: None 1: Warning 2: Minor error 3: Moderate error 4: Major error	3: Moderate error
	Soft reboot enabled [System.PrConst.SoftReboot_Enable]			0: Disabled 1: Enabled	0: Disabled
System parameter	All axes forced stop signal [System.Pr.ForcedStop_Signal]	Signal [SIGNAL_SELECT.Source]	Target [TARGET_REF.Target]	*2	No setting
		Signal detection method [SIGNAL_SELECT.Detection]		0: Detection at TRUE 1: Detection at FALSE 2: Detection at FALSE→TRUE(rising edge) 3: Detection at TRUE→FALSE(falling edge) 4: Detection at rising edge/falling edge	0: Detection at TRUE
		Compensation time [SIGNAL_SELECT.CompensationTime]		—	0.0[s]
		Filter time [SIGNAL_SELECT.FilterTime]		—	0.0[s]
	Stop selection at all axes stop by factors [System.Pr.StopMode_All]			1: Immediate stop 2: Keep current acceleration/ deceleration 3: Alternative acceleration/ deceleration	3: Alternative acceleration/ deceleration

Item [Variable/Structure name]		Setting range	Initial value
System parameter	Deceleration at all axes stop [System.Pr.StopMode_AllDeceleration]	■When "Acceleration/deceleration method" is Accelerating/ Deceleration specifying method 0.0000, 0.0001 to 2147483647.0 ■When "Acceleration/deceleration method" is Time-fixed acceleration/ deceleration method 0, 0.000000, 0.000001 to 8400.0 *: Immediate stop when "0" is set.	0.0

*1 "□□" refers to the name of each addon parameter or the variable/structure name.

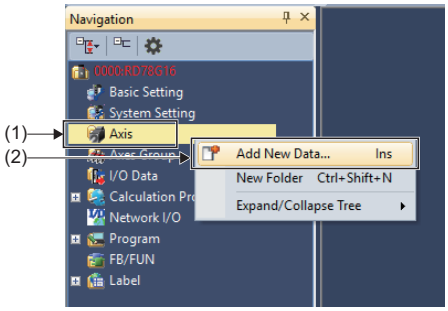
Item [Variable/Structure name]	Initial value	
	Maximum RAM size [ADDON_PARAM.RamSizeMax]	Maximum backup RAM size [ADDON_PARAM.BackupRamSizeMax]
Addon AbsSystem parameter [System.PrConst.Addon_AbsSystem]	1024[kB]	60[kB]
Addon Axis parameter [System.PrConst.Addon_Axis]	16384[kB]	0[kB]
Addon ExternalSignal parameter [System.PrConst.Addon_ExternalSignal]	1024[kB]	0[kB]
Addon FileTransfer parameter [System.PrConst.Addon_FileTransfer]	1024[kB]	0[kB]
Addon Logging parameter [System.PrConst.Addon_Logging]	16384[kB]	0[kB]
Addon MotionControl_AxisFilter parameter [System.PrConst.Addon_MotionControl_AxisFilter]	1024[kB]	0[kB]
Addon MotionControl_General parameter [System.PrConst.Addon_MotionControl_General]	4096[kB]	0[kB]
Addon MotionControl_Sync parameter [System.PrConst.Addon_MotionControl_Sync]	1024[kB]	0[kB]
Addon MotionEngine parameter [System.PrConst.Addon_MotionEngine]	4096[kB]	0[kB]
Addon MotionEventHist parameter [System.PrConst.Addon_MotionEventHist]	1024[kB]	0[kB]
Addon NetworkDriver_CCIETSN parameter [System.PrConst.Addon_NetworkDriver_CCIETSN]	1024[kB]	0[kB]
Addon PackagingApp parameter [System.PrConst.Addon_PackagingApp]	1024[kB]	0[kB]
Addon PlcInstruction parameter [System.PrConst.Addon_PlcInstruction]	1024[kB]	0[kB]
Addon ProfileControl parameter [System.PrConst.Addon_ProfileControl]	16384[kB]	0[kB]
Addon Program_ST parameter [System.PrConst.Addon_Program_ST]	16384[kB]	0[kB]
Addon ServoDriver_CANopen parameter [System.PrConst.Addon_ServoDriver_CANopen]	2048[kB]	0[kB]
Addon ServoSystemRecorder parameter [System.PrConst.Addon_ServoSystemRecorder]	256[kB]	0[kB]
Addon SignalIO parameter [System.PrConst.Addon_SignalIO]	1024[kB]	0[kB]

*2 Set the signal that will be used as the target.

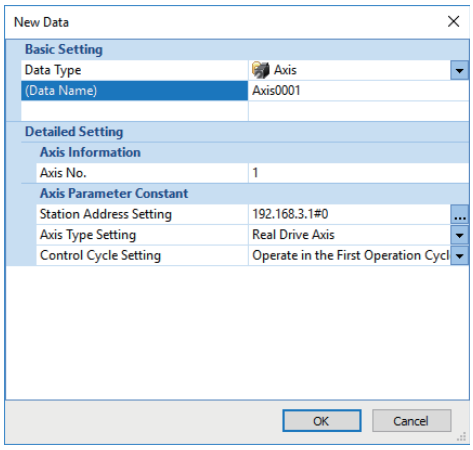
Item	Setting description
Source type	Global label, CANopen object, device, constant
Source data type*3	BOOL, INT, DINT, WORD, DWORD, REAL, LREAL
Source	—

*3 Can only be set when selecting "Device" as the source type.

■Axis

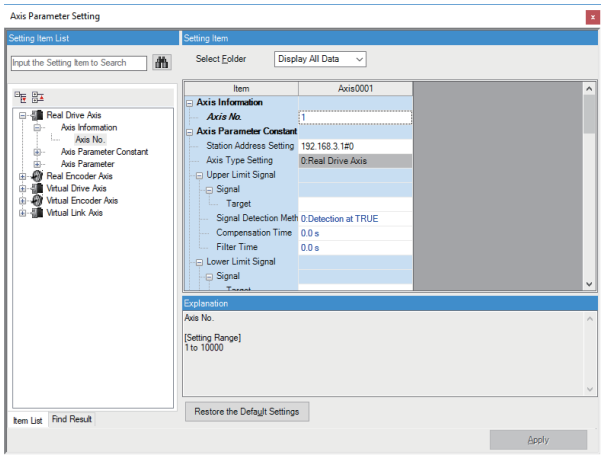


1. From the [Navigation window], right click [Axis] (1), then select [Add New Data] (2).



2. Carry out the setting of the axis to be created, then click the [OK] button.

- Data Name:
  Set the instance name of the axis to be used in the ST program.
- Station Address Setting[*1]:
  When creating a [Real Drive Axis], set the IP address of the servo amplifier and the multi-drop No.[*2] after checking them on the "CC-Link IE TSN Configuration" screen.
- Axis Type Setting:
  Set the type of the axis being created. The following axis types are available.
  · Real drive axis · Real encoder axis
  · Virtual drive axis · Virtual encoder axis
  · Virtual link axis



3. The [Data name[*3]] is added to the [Navigation window]⇨[Axis] and the parameter editor is displayed. Set parameters for the axis.

4. When the setting of parameters for the axis is complete, click the [Apply] button.

*1 Set the values in station address settings as shown below.
· For a single axis device

192.168.3.1
IP address

· For a multi-axis device
<Example> For C-axis of MR-J5W-G

192.168.3.1#2
Multi-drop No.
IP address

*2 Specify the multi-drop No. in order to distinguish logic axes when one station includes multiple logic axes, like the multi-axis drive unit. The multi-drop No. is set as "# + No. (decimal)".
· #0: A-axis
· #1: B-axis
· #2: C-axis

*3 The set data name is displayed.

Displayed items

The following axis types can be set.

• Real drive axis

Item [Variable/Structure name]				Setting range	Initial value
Axis information	Axis No. [AxisName.AxisRef.AxisNo]			1 to 10000	—
Axis parameter constant	Station address setting [AxisName.PrConst.AddressOfStation]			—	No setting
	Axis type setting [AxisName.PrConst.AxisType]			—	0: Real drive axis
	Upper limit signal [AxisName.PrConst.HwStrokeLimit_FlsSignal]	Signal [SIGNAL_SELECT.Source]	Target [TARGET_REF.Target]	*1	No setting
		Signal detection method [SIGNAL_SELECT.Detection]		0: Detection at TRUE 1: Detection at FALSE 2: Detection at FALSE→TRUE (rising edge) 3: Detection at TRUE→FALSE (falling edge) 4: Detection at rising edge/falling edge	0: Detection at TRUE
		Compensation time [SIGNAL_SELECT.CompensationTime]		—	0.0[s]
		Filter time [SIGNAL_SELECT.FilterTime]		—	0.0[s]
	Lower limit signal [AxisName.PrConst.HwStrokeLimit_RlsSignal]	Signal [SIGNAL_SELECT.Source]	Target [TARGET_REF.Target]	*1	No setting
		Signal detection method [SIGNAL_SELECT.Detection]		0: Detection at TRUE 1: Detection at FALSE 2: Detection at FALSE→TRUE (rising edge) 3: Detection at TRUE→FALSE (falling edge) 4: Detection at rising edge/falling edge	0: Detection at TRUE
		Compensation time [SIGNAL_SELECT.CompensationTime]		—	0.0[s]
		Filter time [SIGNAL_SELECT.FilterTime]		—	0.0[s]
	Control cycle setting [AxisName.PrConst.OperationCycle]			0: Operate in the first operation cycle	0: Operate in the first operation cycle
	Absolute position reference setting [AxisName.PrConst.PosRestoration_AbsPosBase]			3: Feed machine position	3: Feed machine position
	Absolute position management setting [AxisName.PrConst.PosRestoration_AbsPosEnable]			0: Disable absolute position system 1: Enable absolute position system -1: Automatic setting (acquire from connected device)	-1: Automatic setting (acquire from connected device)
	Ring counter enabled selection [AxisName.PrConst.RingCount_Enable]			0: Disabled 1: Enabled	0: Disabled
	Ring counter lower limit value [AxisName.PrConst.RingCount_LowerValue]			-1000000000000.0 to 1000000000000.0	-10000000000.0
	Ring counter upper limit value [AxisName.PrConst.RingCount_UpperValue]			-1000000000000.0 to 1000000000000.0	10000000000.0
	Slave emulation enabled [AxisName.PrConst.SlaveEmulate_Enable]			0: Disabled 1: Enabled	0: Disabled
	Torque limit maximum value [AxisName.PrConst.TorqueLimit_Max]			0.0 to 1000.0%	1000.0[%]
	Negative direction torque limit initial value [AxisName.PrConst.TorqueLimit_NegativeInitial]			0.0 to 1000.0%	300.0[%]

Item [Variable/Structure name]				Setting range	Initial value
Axis parameter constant	Positive direction torque limit initial value [AxisName.PrConst.TorqueLimit_PositiveInitial]			0.0 to 1000.0%	300.0[%]
	High-speed mode setting [AxisName.PrConst.FastOperationMode]			5FE2: High-speed mode Others: Normal mode	0000
Axis parameter	Acceleration limit value [AxisName.Pr.AccelerationLimit]			0.0000, 0.0001 to 2147483647.0 *: Limiting is not performed when "0.0000".	2147483647.0
	Operation selection at start acceleration/deceleration 0 [AxisName.Pr.AccelerationZeroBehavior]			-1: Error (not started) 1: Maximum acceleration/deceleration	-1: Error (not started)
	Command in-position width [AxisName.Pr.CmdInPos_Width]			0.0: Function disabled 0.000000001 to 10000000000.0: Function enabled	100.0
	Deceleration limit value [AxisName.Pr.DecelerationLimit]			0.0000, 0.0001 to 2147483647.0 *: Limiting is not performed when "0.0000".	2147483647.0
	Driver unit conversion numerator [AxisName.Pr.Drive_UnitConvRatioNum]			1 to 2147483647	1
	Driver unit conversion denominator [AxisName.Pr.Drive_UnitConvRatioDen]			1 to 2147483647	1
	Forced stop signal [AxisName.Pr.ForcedStop_Signal]	Signal [SIGNAL_SELECT.Source]	Target [TARGET_REF.Target]	*1	No setting
		Signal detection method [SIGNAL_SELECT.Detection]		0: Detection at TRUE 1: Detection at FALSE 2: Detection at FALSE→TRUE (rising edge) 3: Detection at TRUE→FALSE (falling edge) 4: Detection at rising edge/falling edge	0: Detection at TRUE
		Compensation time [SIGNAL_SELECT.CompensationTime]		—	0.0[s]
		Filter time [SIGNAL_SELECT.FilterTime]		—	0.0[s]
	Home position return required or not [AxisName.Pr.Homing_Required]			0: Home position return not required 1: Home position return required	1: Home position return required
	Jerk limit value [AxisName.Pr.JerkLimit]			0.0000, 0.0001 to 2147483647.0 *: Limiting is not performed when "0.0000".	2147483647.0
	Operation setting at overrun [AxisName.Pr.OverrunOperation]			1: Immediate stop 2: Keep current acceleration/deceleration	1: Immediate stop
	Start permission at home position return uncompleted [AxisName.Pr.StartableAtUnhomed]			0: Disabled 1: Enabled	0: Disabled
	Deceleration at stop [AxisName.Pr.StopMode_Deceleration]			■When acceleration/deceleration method is "acceleration/deceleration specification method" 0.0000, 0.0001 to 2147483647.0 ■When acceleration/deceleration method is "acceleration/deceleration time-fixed method" 0.0000, 0.000001 to 8400.0	0.0
	Stop selection at deceleration to stop [AxisName.Pr.StopMode_DecelerationCurve]			1: Recreate deceleration curve	1: Recreate deceleration curve
	Stop selection at stop by factors [AxisName.Pr.StopMode_General]			1: Immediate stop 2: Keep current acceleration/deceleration 3: Alternative acceleration/deceleration	3: Alternative acceleration/deceleration
	Stop selection at hardware stroke limit error occurrence [AxisName.Pr.StopMode_HwStrokeLimit]			1: Immediate stop 2: Keep current acceleration/deceleration 3: Alternative acceleration/deceleration	1: Immediate stop

Item [Variable/Structure name]				Setting range	Initial value
Axis parameter	Process selection at servo OFF command during operation [AxisName.Pr.StopMode_ServoOff]			0: Ignore 4: Servo OFF after immediate stop 5: Servo OFF after deceleration to stop	0: Ignore
	Stop selection at software stroke limit error occurrence [AxisName.Pr.StopMode_SwStrokeLimit]			1: Immediate stop 2: Keep current acceleration/ deceleration 3: Alternative acceleration/deceleration	1: Immediate stop
	Driver command discard detection setting [AxisName.Pr.StopOption_DriverTargetIgnored]			0: Detection disabled 1: Detection enabled	1: Detection enabled
	Stop signal [AxisName.Pr.StopSignal]	Signal [SIGNAL_SELECT.Source]	Target [TARGET_REF.Target]	*1	No setting
		Signal detection method [SIGNAL_SELECT.Detection]		0: Detection at TRUE 1: Detection at FALSE 2: Detection at FALSE→TRUE (rising edge) 3: Detection at TRUE→FALSE (falling edge) 4: Detection at rising edge/falling edge	0: Detection at TRUE
		Compensation time [SIGNAL_SELECT.CompensationTime]		—	0.0[s]
		Filter time [SIGNAL_SELECT.FilterTime]		—	0.0[s]
	Software stroke limit lower value [AxisName.Pr.SwStrokeLimit_Lower]			-1000000000000.0 to 1000000000000.0	-10000000000.0
	Software stroke limit target [AxisName.Pr.SwStrokeLimit_Target]			-1: Invalid 1: Command current position 3: Feed machine position	-1: Invalid
	Software stroke limit upper value [AxisName.Pr.SwStrokeLimit_Upper]			-1000000000000.0 to 1000000000000.0	10000000000.0
	Position command unit [AxisName.Pr.Unit_Position]			pulse, m, mm, um, nm, degree, Revolution, inch, arbitrary unit	pulse
	Position command unit string [AxisName.Pr.Unit_PositionString]			—	No setting
	Speed command unit [AxisName.Pr.Unit_Velocity]			U/s, U/ms, U/us, U/ns, U/min	U/s
	Negative direction speed limit value [AxisName.Pr.VelocityLimit_Negative]			0.0001 to 2500000000.0	2500000000.0
	Operation setting at speed limit value exceeded [AxisName.Pr.VelocityLimit_OverOperation]			0: Ignore 3: Immediate stop	0: Ignore
	Positive direction speed limit value [AxisName.Pr.VelocityLimit_Positive]			0.0001 to 2500000000.0	2500000000.0

*1 Set the signal that will be used as the target.

Item	Setting description
Source type	Global label, CANopen object, device, constant
Source data type*2	BOOL, INT, DINT, WORD, DWORD, REAL, LREAL
Source	—

*2 Can only be set when selecting "Device" as the source type.

• Real encoder axis

Item [Variable/Structure name]				Setting range	Initial value
Axis information	Axis No. [AxisName.AxisRef.AxisNo]			1 to 10000	—
Axis parameter constant	Station address setting [AxisName.PrConst.AddressOfStation]			—	No setting
	Axis type setting [AxisName.PrConst.AxisType]			—	2: Real encoder axis
	Real encoder axis type setting [AxisName.PrConst.Encoder_AxisType]			1: Via drive module	1: Via drive module
	Counter disabling signal [AxisName.PrConst.Encoder_CounterDisableSignal]	Signal [SIGNAL_SELECT.Source]	Target [TARGET_REF.Target]	*1	No setting
		Signal detection method [SIGNAL_SELECT.Detection]		0: Detection at TRUE 1: Detection at FALSE 2: Detection at FALSE→TRUE (rising edge) 3: Detection at TRUE→FALSE (falling edge) 4: Detection at rising edge/falling edge	0: Detection at TRUE
		Compensation time [SIGNAL_SELECT.CompensationTime]		—	0.0[s]
		Filter time [SIGNAL_SELECT.FilterTime]		—	0.0[s]
	Control cycle setting [AxisName.PrConst.OperationCycle]			0: Operate in the first operation cycle	0: Operate in the first operation cycle
	Absolute position management setting [AxisName.PrConst.PosRestoration_AbsPosEnable]			0: Disable absolute position system 1: Enable absolute position system -1: Automatic setting (acquire from connected device)	-1: Automatic setting (acquire from connected device)
	Ring counter enabled selection [AxisName.PrConst.RingCount_Enable]			0: Disabled 1: Enabled	0: Disabled
	Ring counter lower limit value [AxisName.PrConst.RingCount_LowerValue]			-1000000000000.0 to 1000000000000.0	-10000000000.0
	Ring counter upper limit value [AxisName.PrConst.RingCount_UpperValue]			-1000000000000.0 to 1000000000000.0	10000000000.0
	High-speed mode setting [AxisName.PrConst.FastOperationMode]			0000*3	0000
Axis parameter	Driver unit conversion numerator [AxisName.Pr.Drive_UnitConvRatioNum]			1 to 2147483647	1
	Driver unit conversion denominator [AxisName.Pr.Drive_UnitConvRatioDen]			1 to 2147483647	1
	Home position return required or not [AxisName.Pr.Homing_Required]			0: Home position return not required 1: Home position return required	1: Home position return required
	Start position at home position return uncompleted [AxisName.Pr.StartableAtUnhomed]			0: Disabled 1: Enabled	0: Disabled
	Position command unit [AxisName.Pr.Unit_Position]			pulse, m, mm, μm, nm, degree, Revolution, inch, arbitrary unit	pulse
	Position command unit string [AxisName.Pr.Unit_PositionString]			—	No setting
	Speed command unit [AxisName.Pr.Unit_Velocity]			U/s, U/ms, U/μs, U/ns, U/min	U/s

*1 Set the signal that will be used as the target.

Item	Setting description
Source type	Global label, CANopen object, device, constant
Source data type*2	BOOL, INT, DINT, WORD, DWORD, REAL, LREAL
Source	—

*2 Can only be set when selecting "Device" as the source type.
*3 Do not change this setting.

• Virtual drive axis

Item [Variable/Structure name]				Setting range	Initial value
Axis information	Axis No. [AxisName.AxisRef.AxisNo]			1 to 10000	—
Axis parameter constant	Axis type setting [AxisName.PrConst.AxisType]			—	3: Virtual drive axis
	Upper limit signal [AxisName.PrConst.HwStrokeLimit_FlsSignal]	Signal [SIGNAL_SELECT.Source]	Target [TARGET_REF.Target]	*1	No setting
		Signal detection method [SIGNAL_SELECT.Detection]		0: Detection at TRUE 1: Detection at FALSE 2: Detection at FALSE→TRUE (rising edge) 3: Detection at TRUE→FALSE (falling edge) 4: Detection at rising edge/falling edge	0: Detection at TRUE
		Compensation time [SIGNAL_SELECT.CompensationTime]		—	0.0[s]
		Filter time [SIGNAL_SELECT.FilterTime]		—	0.0[s]
	Lower limit signal [AxisName.PrConst.HwStrokeLimit_RlsSignal]	Signal [SIGNAL_SELECT.Source]	Target [TARGET_REF.Target]	*1	No setting
		Signal detection method [SIGNAL_SELECT.Detection]		0: Detection at TRUE 1: Detection at FALSE 2: Detection at FALSE→TRUE (rising edge) 3: Detection at TRUE→FALSE (falling edge) 4: Detection at rising edge/falling edge	0: Detection at TRUE
		Compensation time [SIGNAL_SELECT.CompensationTime]		—	0.0[s]
		Filter time [SIGNAL_SELECT.FilterTime]		—	0.0[s]
	Control cycle setting [AxisName.PrConst.OperationCycle]			0: Operate in the first operation cycle	0: Operate in the first operation cycle
	Absolute position management setting [AxisName.PrConst.PosRestoration_AbsPosEnable]			0: Disable absolute position system 1: Enable absolute position system -1: Automatic setting (acquire from connected device)	0: Disable absolute position system
	Ring counter enabled selection [AxisName.PrConst.RingCount_Enable]			0: Disabled 1: Enabled	0: Disabled
	Ring counter lower limit value [AxisName.PrConst.RingCount_LowerValue]			-1000000000000.0 to 1000000000000.0	-10000000000.0
	Ring counter upper limit value [AxisName.PrConst.RingCount_UpperValue]			-1000000000000.0 to 1000000000000.0	10000000000.0
	High-speed mode setting [AxisName.PrConst.FastOperationMode]			0000*3	0000

Item [Variable/Structure name]				Setting range	Initial value
Axis parameter	Acceleration limit value [AxisName.Pr.AccelerationLimit]			0.0000, 0.0001 to 2147483647.0 *: Limiting is not performed when "0.0000".	2147483647.0
	Operation selection at start acceleration/deceleration 0 [AxisName.Pr.AccelerationZeroBehavior]			-1: Error (not started) 1: Maximum acceleration/ deceleration	-1: Error (not started)
	Command in-position width [AxisName.Pr.CmdInPos_Width]			0.0: Function disabled 0.000000001 to 10000000000.0: Function enabled	100.0
	Deceleration limit value [AxisName.Pr.DecelerationLimit]			0.0000, 0.0001 to 2147483647.0 *: Limiting is not performed when "0.0000".	2147483647.0
	Forced stop signal [AxisName.Pr.ForcedStop_Signal]	Signal [SIGNAL_SELECT.Source]	Target [TARGET_REF.Target]	*1	No setting
		Signal detection method [SIGNAL_SELECT.Detection]		0: Detection at TRUE 1: Detection at FALSE 2: Detection at FALSE→TRUE (rising edge) 3: Detection at TRUE→FALSE (falling edge) 4: Detection at rising edge/falling edge	0: Detection at TRUE
		Compensation time [SIGNAL_SELECT.CompensationTime]		—	0.0[s]
		Filter time [SIGNAL_SELECT.FilterTime]		—	0.0[s]
	Home position return required or not [AxisName.Pr.Homing_Required]			0: Home position return not required 1: Home position return required	1: Home position return required
	Jerk limit value [AxisName.Pr.JerkLimit]			0.0000, 0.0001 to 2147483647.0 *: Limiting is not performed when "0.0000".	2147483647.0
	Operation setting at overrun [AxisName.Pr.OverrunOperation]			1: Immediate stop 2: Keep current acceleration/ deceleration	1: Immediate stop
	Start permission at home position return uncompleted [AxisName.Pr.StartableAtUnhomed]			0: Disabled 1: Enabled	0: Disabled
	Deceleration at stop [AxisName.Pr.StopMode_Deceleration]			■When acceleration/deceleration method is "acceleration/ deceleration specification method" 0.0000, 0.0001 to 2147483647.0 ■When acceleration/deceleration method is "acceleration/ deceleration time-fixed method" 0.0000, 0.000001 to 8400.0	0.0
	Stop selection at deceleration to stop [AxisName.Pr.StopMode_DecelerationCurve]			1: Recreate deceleration curve	1: Recreate deceleration curve
	Stop selection at stop by factors [AxisName.Pr.StopMode_General]			1: Immediate stop 2: Keep current acceleration/ deceleration 3: Alternative acceleration/ deceleration	3: Alternative acceleration/ deceleration
	Stop selection at hardware stroke limit error occurrence [AxisName.Pr.StopMode_HwStrokeLimit]			1: Immediate stop 2: Keep current acceleration/ deceleration 3: Alternative acceleration/ deceleration	1: Immediate stop

Item [Variable/Structure name]				Setting range	Initial value
Axis parameter	Process selection at servo OFF command during operation [AxisName.Pr.StopMode_ServoOff]			0: Ignore 4: Servo OFF after immediate stop 5: Servo OFF after deceleration to stop	0: Ignore
	Stop selection at software stroke limit error occurrence [AxisName.Pr.StopMode_SwStrokeLimit]			1: Immediate stop 2: Keep current acceleration/ deceleration 3: Alternative acceleration/ deceleration	1: Immediate stop
	Stop signal [AxisName.Pr.StopSignal]	Signal [SIGNAL_SELECT.Source]	Target [TARGET_REF.Target]	*1	No setting
		Signal detection method [SIGNAL_SELECT.Detection]		0: Detection at TRUE 1: Detection at FALSE 2: Detection at FALSE→TRUE (rising edge) 3: Detection at TRUE→FALSE (falling edge) 4: Detection at rising edge/falling edge	0: Detection at TRUE
		Compensation time [SIGNAL_SELECT.CompensationTime]		—	0.0[s]
		Filter time [SIGNAL_SELECT.FilterTime]		—	0.0[s]
	Software stroke limit lower value [AxisName.Pr.SwStrokeLimit_Lower]			-1000000000000.0 to 1000000000000.0	-10000000000.0
	Software stroke limit target [AxisName.Pr.SwStrokeLimit_Target]			-1: Invalid 1: Command current position 3: Feed machine position	-1: Invalid
	Software stroke limit upper value [AxisName.Pr.SwStrokeLimit_Upper]			-1000000000000.0 to 1000000000000.0	10000000000.0
	Position command unit [AxisName.Pr.Unit_Position]			pulse, m, mm, um, nm, degree, Revolution, inch, arbitrary unit	pulse
	Position command unit string [AxisName.Pr.Unit_PositionString]			—	No setting
	Speed command unit [AxisName.Pr.Unit_Velocity]			U/s, U/ms, U/us, U/ns, U/min	U/s
	Negative direction speed limit value [AxisName.Pr.VelocityLimit_Negative]			0.0001 to 2500000000.0	2500000000.0
	Operation setting at speed limit value exceeded [AxisName.Pr.VelocityLimit_OverOperation]			0: Ignore 3: Immediate stop	0: Ignore
	Positive direction speed limit value [AxisName.Pr.VelocityLimit_Positive]			0.0001 to 2500000000.0	2500000000.0

*1 Set the signal that will be used as the target.

Item	Setting description
Source type	Global label, CANopen object, device, constant
Source data type*2	BOOL, INT, DINT, WORD, DWORD, REAL, LREAL
Source	—

*2 Can only be set when selecting "Device" as the source type.

*3 Do not change this setting.

• Virtual encoder axis

Item [Variable/Structure name]				Setting range	Initial value
Axis information	Axis No. [AxisName.AxisRef.AxisNo]			1 to 10000	—
Axis parameter constant	Axis type setting [AxisName.PrConst.AxisType]			—	4: Virtual encoder axis
	Counter disabling signal [AxisName.PrConst.Encoder_CounterDisableSignal]	Signal [SIGNAL_SELECT.Source]	Target [TARGET_REF.Target]	*1	No setting
		Signal detection method [SIGNAL_SELECT.Detection]		0: Detection at TRUE 1: Detection at FALSE 2: Detection at FALSE→TRUE (rising edge) 3: Detection at TRUE→FALSE (falling edge) 4: Detection at rising edge/falling edge	0: Detection at TRUE
		Compensation time [SIGNAL_SELECT.CompensationTime]		—	0.0[s]
		Filter time [SIGNAL_SELECT.FilterTime]		—	0.0[s]
	Encoder ring counter lower limit value [AxisName.PrConst.Encoder_RingCout_LowerValue]			■PosActualValue is 1 word -32768 to 32767 ■PosActualValue is 2 words or omitted -2147483648 to 2147483647	0
	Encoder ring counter upper limit value [AxisName.PrConst.Encoder_RingCout_UpperValue]				0
	Control cycle setting [AxisName.PrConst.OperationCycle]			0: Operate in the first operation cycle	0: Operate in the first operation cycle
	Absolute position management setting [AxisName.PrConst.PosRestoration_AbsPosEnable]			0: Disable absolute position system 1: Enable absolute position system -1: Automatic setting (acquire from connected device)	-1: Automatic setting (acquire from connected device)
	Ring counter enabled selection [AxisName.PrConst.RingCount_Enable]			0: Disabled 1: Enabled	0: Disabled
	Ring counter lower limit value [AxisName.PrConst.RingCount_LowerValue]			-1000000000000.0 to 1000000000000.0	-10000000000.0
	Ring counter upper limit value [AxisName.PrConst.RingCount_UpperValue]			-1000000000000.0 to 1000000000000.0	10000000000.0
	Slave object data [AxisName.PrConst.SlaveObject]	PosActualValue [SLAVE_OBJECT_VIRTUAL_ENCODER.PosActualValue]		*1	No setting
	High-speed mode setting [AxisName.PrConst.FastOperationMode]			0000*3	0000
Axis parameter	Driver unit conversion numerator [AxisName.Pr.Drive_UnitConvRatioNum]			1 to 2147483647	1
	Driver unit conversion denominator [AxisName.Pr.Drive_UnitConvRatioDen]			1 to 2147483647	1
	Home position return required or not [AxisName.Pr.Homing_Required]			0: Home position return not required 1: Home position return required	1: Home position return required
	Start position at home position return uncompleted [AxisName.Pr.StartableAtUnhomed]			0: Disabled 1: Enabled	0: Disabled
	Position command unit [AxisName.Pr.Unit_Position]			pulse, m, mm, μm, nm, degree, Revolution, inch, arbitrary unit	pulse
	Position command unit string [AxisName.Pr.Unit_PositionString]			—	No setting
	Speed command unit [AxisName.Pr.Unit_Velocity]			U/s, U/ms, U/μs, U/ns, U/min	U/s

*1 Set the signal that will be used as the target.

Item	Setting description
Source type	Global label, CANopen object, device, constant
Source data type*2	BOOL, INT, DINT, WORD, DWORD, REAL, LREAL
Source	—

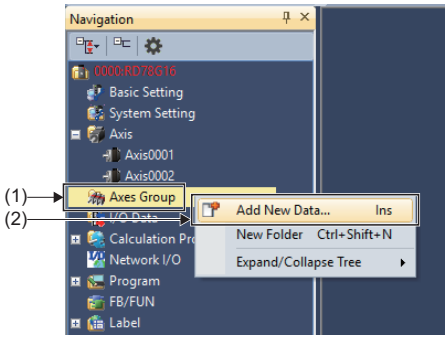
*2 Can only be set when selecting "Device" as the source type.
*3 Do not change this setting.

• Virtual link axis

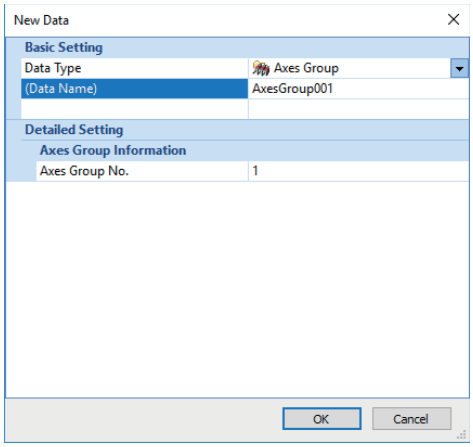
Item [Variable/Structure name]		Setting range	Initial value
Axis information	Axis No. [AxisName.AxisRef.AxisNo]	1 to 10000	—
Axis parameter constant	Axis type setting [AxisName.PrConst.AxisType]	—	5: Virtual link axis
	Control cycle setting [AxisName.PrConst.OperationCycle]	0: Operate in the first operation cycle	0: Operate in the first operation cycle
	Absolute position management setting [AxisName.PrConst.PosRestoration_AbsPosEnable]	0: Disable absolute position system 1: Enable absolute position system -1: Automatic setting (acquire from connected device)	0: Disable absolute position system
	Ring counter enabled selection [AxisName.PrConst.RingCount_Enable]	0: Disabled 1: Enabled	0: Disabled
	Ring counter lower limit value [AxisName.PrConst.RingCount_LowerValue]	-1000000000000.0 to 1000000000000.0	-10000000000.0
	Ring counter upper limit value [AxisName.PrConst.RingCount_UpperValue]	-1000000000000.0 to 1000000000000.0	10000000000.0
	High-speed mode setting [AxisName.PrConst.FastOperationMode]	0000*1	0000
Axis parameter	Home position return required or not [AxisName.Pr.Homing_Required]	0: Home position return not required 1: Home position return required	1: Home position return required
	Start permission at home position return uncompleted [AxisName.Pr.StartableAtUnhomed]	0: Disabled 1: Enabled	0: Disabled
	Position command unit [AxisName.Pr.Unit_Position]	pulse, m, mm, um, nm, degree, Revolution, inch, arbitrary unit	pulse
	Position command unit string [AxisName.Pr.Unit_PositionString]	—	No setting
	Speed command unit [AxisName.Pr.Unit_Velocity]	U/s, U/ms, U/us, U/ns, U/min	U/s

*1 Do not change this setting.

■Axes group

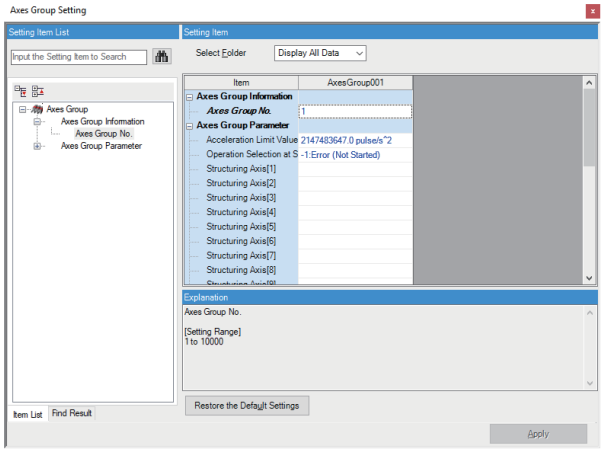


1. From the [Navigation window], right click [Axes Group] (1), then select [Add New Data] (2).



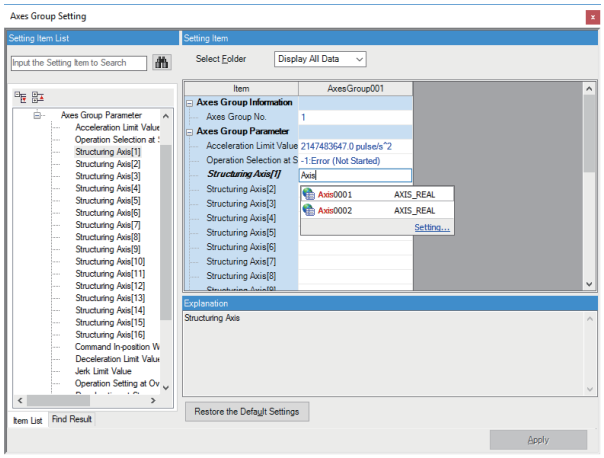
2. Carry out the setting of the axes group to be created, then click the [OK] button.

- Data Name:
  Set the instance name of the axes group to be used in the ST program.



3. The [Data name[*1]] is added to the [Navigation window]⇨[Axes Group] and the parameter editor is displayed.

*1: The set data name is displayed.



4. Set the structuring axes for the created axes group. Set the [Data Name] (the instance name for the axis to be used in the ST program) that was set in the [Navigation window]⇨[Axis].

5. When the setting of parameters for the axes group is complete, click the [Apply] button.

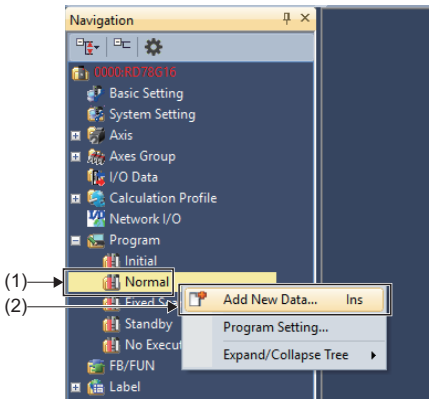
Displayed items

• Axes group

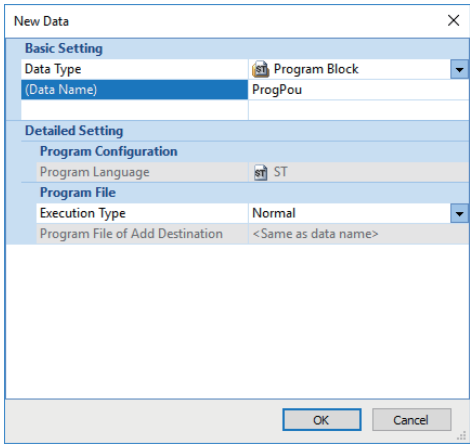
Item [Variable/Structure name]		Setting range	Initial value
Axes group information	Axes group No. [AxesGroupName.AxesGroupRef.GroupNo]	1 to 10000	—
Axes group parameter	Acceleration limit value [AxesGroupName.Pr.AccelerationLimit]	0.0000, 0.0001 to 2147483647.0 *: Limiting is not performed when "0.0000".	2147483647.0
	Operation selection at start acceleration/deceleration 0 [AxesGroupName.Pr.AccelerationZeroBehavior]	-1: Error (not started) 1: Maximum acceleration/ deceleration	-1: Error (not started)
	Structuring axis[☐] [AxesGroupName.Pr.Axis[☐]] *: ☐=1 to 16	—	No setting
	Command in-position width [AxesGroupName.Pr.CmdInPos_Width]	0.0: Function disabled 0.000000001 to 10000000000.0: Function enabled	100.0
	Deceleration limit value [AxesGroupName.Pr.DecelerationLimit]	0.0000, 0.0001 to 2147483647.0 *: Limiting is not performed when "0.0000".	2147483647.0
	Jerk limit value [AxesGroupName.Pr.JerkLimit]	0.0000, 0.0001 to 2147483647.0 *: Limiting is not performed when "0.0000".	2147483647.0
	Operation setting at overrun [AxesGroupName.Pr.OverrunOperation]	1: Immediate stop	1: Immediate stop
	Deceleration at stop [AxesGroupName.Pr.StopMode_Deceleration]	■When acceleration/deceleration method is "acceleration/ deceleration specification method" 0.0000, 0.0001 to 2147483647.0 ■When acceleration/deceleration method is "acceleration/ deceleration time-fixed method" 0.0000, 0.000001 to 8400.0	0.0
	Stop selection at deceleration to stop [AxesGroupName.Pr.StopMode_DecelerationCurve]	1: Recreate deceleration curve	1: Recreate deceleration curve
	Structuring axis operation selection at axis stop by factors [AxesGroupName.Pr.StopMode_ErrorInGroup]	1: Immediate stop	1: Immediate stop
	Stop selection at stop by factors [AxesGroupName.Pr.StopMode_General]	1: Immediate stop 2: Keep current acceleration/ deceleration 3: Alternative acceleration/ deceleration	3: Alternative acceleration/ deceleration
	Position command unit [AxesGroupName.Pr.Unit_Position]	pulse, m, mm, um, nm, degree, Revolution, inch, arbitrary unit	pulse
	Position command unit string [AxesGroupName.Pr.Unit_PositionString]	—	No setting
	Speed command unit [AxesGroupName.Pr.Unit_Velocity]	U/s, U/ms, U/us, U/ns, U/min	U/s
	Speed limit value [AxesGroupName.Pr.VelocityLimit]	0.0001 to 2500000000.0	2500000000.0

ST program creation

Create a program using ST programming language.



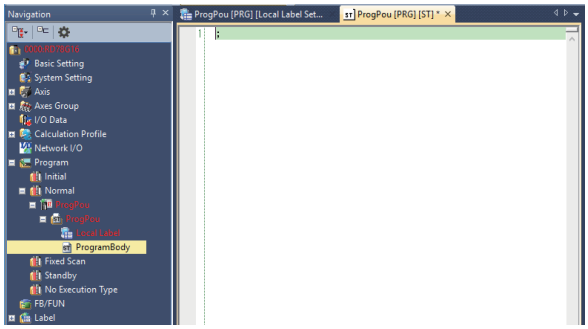
1. From the [Navigation window], right click [Program] (1), then select [Add New Data] (2).



2. Carry out the setting of the program to be created, then click the [OK] button.

- Data Name:
  Set the instance name for the program.
- Execution Type*1:
  Set the execution type for the program.

*1: Refer to execution of programs for details of execution types.(☞ Page 431 Execution of programs)



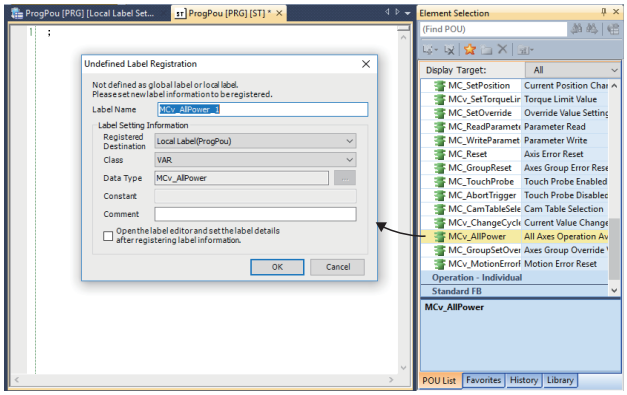
3. [ProgPou]⇨[ProgPou]⇨[Local Label]/[ProgramBody] are added to the [Navigation window]⇨[Program]⇨[Execution type*2] and the program work window is displayed.

4. Create the ST program.

*2: The execution type that was set is displayed as the execution type name.
*3: The program file is created in the tree of the execution type that was set.

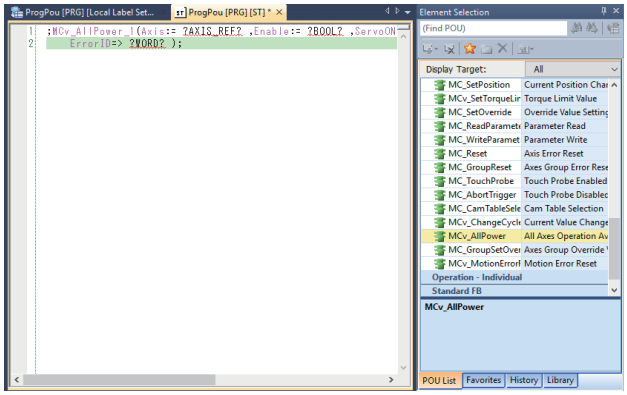
■Inserting Motion control FBs

To create a program, assign Motion control FBs by dragging and dropping them onto the ST program.



1. From the [Navigation window], select [Program]⇨[Execution type*1]⇨[ProgPou]⇨[ProgPou]⇨[ProgramBody] to display the program work window.
2. Select the FB to be inserted from the Element Selection window and drag and drop it into its assigned position to display the "Undefined Label Registration" screen.
3. Set the label name and label setting information, then click the [OK] button.

*1: The name of the created execution type is displayed.

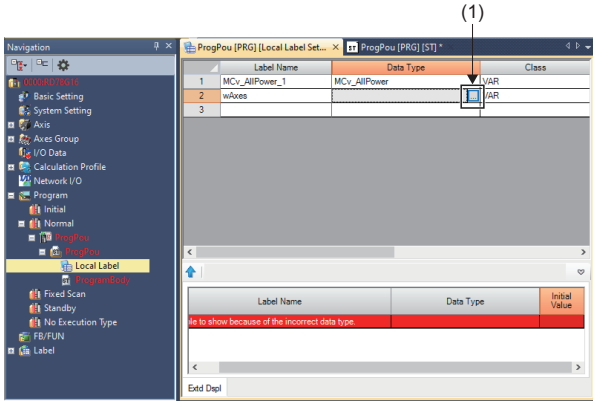


4. The FB is inserted into the program work window.

*2: Inserting a FB from the Element Selection window registers a label using the inserted FB label name into the label setting list.

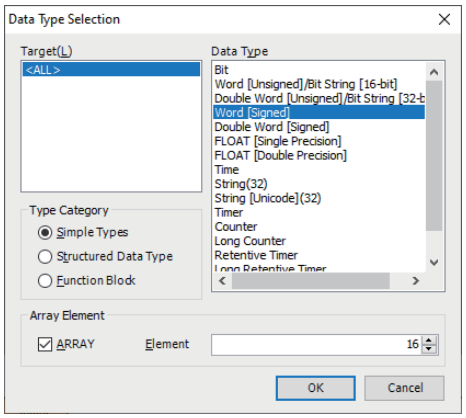
■Registering labels

Register the local labels to be used in the ST program.



1. From the [Navigation window], select [Program]⇨[Execution type*1]⇨[ProgPou]⇨[ProgPou]⇨[Local Label] to display the local label work window.
2. Carry out the setting of label names, data types, and classes, then register them.
3. When setting the data type, click [...](1).

*1: The name of the created execution type is displayed.



4. Perform the settings for the data type to be created, then click the [OK] button.

- Target:
  Sets the category of the created label.
- Data type:
  Sets the data type of the created label.
- Type category:
  Sets the category of the created label.
- Array element:
  When using an array, place a check in ARRAY, then set the number of elements.

Creation procedure for programs using the public label function

This section describes the procedure for creating a sequence program that uses the public label function with an engineering tool.

1. Perform steps 1. to 4. in "Program creation procedure for ST programs".

- Parameter settings (☞ Page 439 Parameter settings)
- Network settings (☞ Page 440 Network settings)
- Module parameter settings (☞ Page 443 Module parameter settings)
- Motion extended parameter settings (☞ Page 444 Module extended parameter settings)

2. Label registration

Register labels used by the Motion module using the "Motion Control Setting Function".
Register labels to be made public to the CPU module side in global labels.
(☞ Page 463 Label registration)

3. Public label settings

Carry out settings so that global labels registered using the "Motion Control Setting Function" are made public to the CPU module side.
(☞ Page 464 Public label settings)

4. Label conversion

Convert labels to fix areas that were set using the "Motion Control Setting Function".
Check the labels that were set, then create the label information to be made public.
(☞ Page 465 Label conversion)

5. Public label reflection

Reflect public label information created using the "Motion Control Setting Function" to the project on the CPU module side.
After reflection completes, the public label information is automatically registered to the CPU module side as module labels.
(☞ Page 467 Public label reflection)

6. Sequence program creation

Create a sequence program using the Motion module labels and Motion control FBs that were made public.

- Inserting public labels (☞ Page 470 Inserting public labels)
- Inserting Motion control FBs (☞ Page 470 Inserting Motion control FBs)

7. Writing the created program

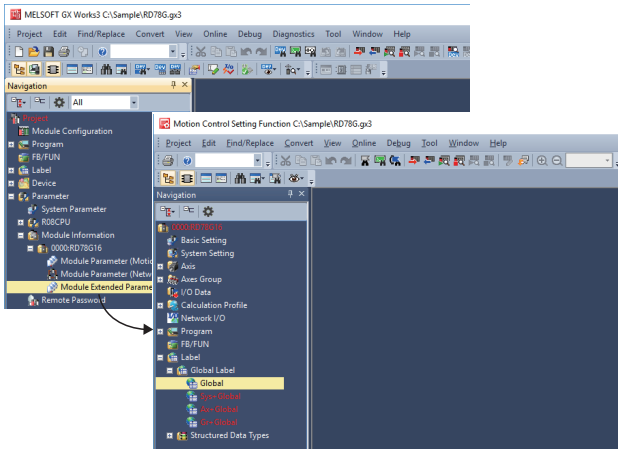
Write the program, global labels, and module parameters created on the CPU module side and Motion module side to both modules.
(☞ Page 471 Writing the created program)

Point

When creating a sequence program using Motion control FBs on the CPU module side, the Motion module FB library must be registered to the library of the engineering tool.
For registering the Motion module FB library, refer to the following.
☞ Page 472 Registering the Motion module FB library

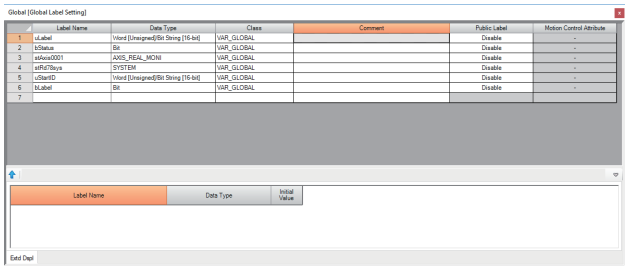
Label registration

Register the labels (global labels) to be used by the Motion module on the Motion Control Setting Function screen.



1. From the [Navigation window], select [Parameter]⇨[Module Information]⇨[Module model name*1]⇨[Module Extended Parameter] to display the "Motion Control Setting Function".

*1: The model name of the set Motion module is displayed.



2. From the [Navigation window], select [Label]⇨[Global Label]⇨[Global*2] to display the label editor for global labels. Register the labels to be used by the Motion module.

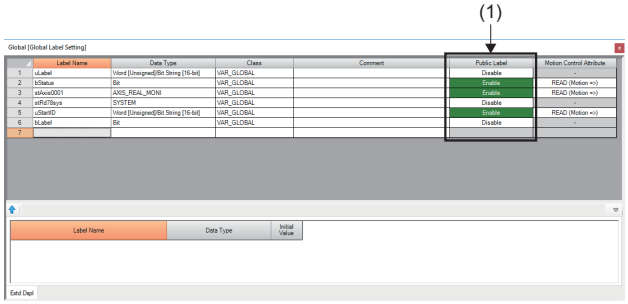
*2: New global label lists can also be created and used to register labels.

Point

Labels registered in global labels are made public to the CPU module side. Labels registered in local labels cannot be made public.

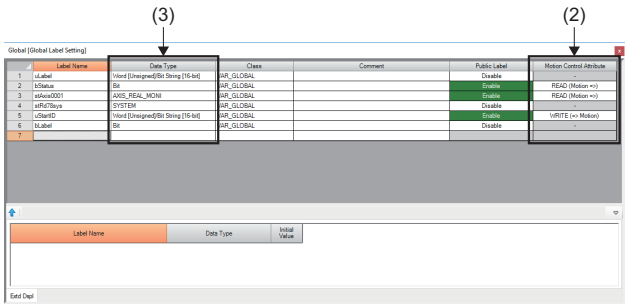
Public label settings

To make registered global labels public to the CPU module side, set the "Public Label" and "Motion Control Attribute".



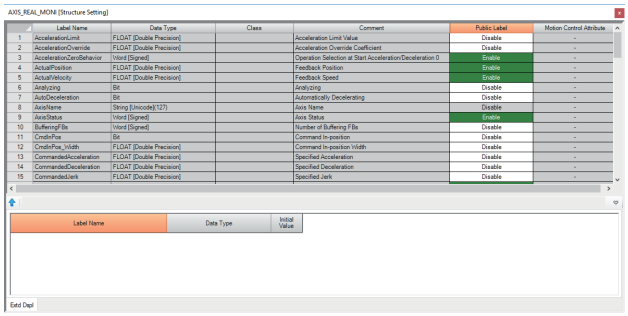
1. In the label editor, set the public label (1) of label names to be made public to the CPU module side to "Enable".

*1: Labels set to certain data types and classes cannot be set as public labels. (☞ Page 464 Conditions for making labels public to the CPU module side)



2. For label names whose public label was set to "Enable", set the refresh direction in Motion Control Attribute (2)*2. When the data type of the set label name is a structured data type, set the refresh direction using the type of structure set in Data Type (3).

*2: Refer to Motion control attribute setting items for the items to set. (☞ Page 465 Motion control attribute setting items)



3. When the data type of the label name was set to a structured data type in Step 2., select [Navigation window]⇨[Label]⇨[Structured Data Types]⇨[Structure name*3] to display the structure label editor. Set the public label and Motion control attribute for label names to be made public to the CPU module side.

*3: The set structure name is displayed.

■Conditions for making labels public to the CPU module side

The conditions for making labels public to the CPU module side are shown below.

○: Setting possible, ×: Setting not possible, △: Setting not possible in some cases

Variable type	Type	Array selection	Public label setting possible/not possible	Remark
Global label	Simple type	None	○	Setting is not possible for the following labels and classes. ■Label • String type labels • Timer type labels • Counter type labels • Long counter type labels • Retentive timer type labels • Long retentive timer type labels • Long timer type labels ■Class • VAR_GLOBAL_CONSTANT class
		Provided	△*1*2	
	Structured data type	None	△*3	
		Provided	△*1*2*4*5	
	Function block (including MCFB)	None	×	
		Provided	×	
Program	—	—	×	
Program block local label	—	—	×	
Structure	—	—	△*3*5	
MCFB structure	—	—	△*6*7	

*1 Individual array elements cannot be set as public labels.
*2 When using a bit-type array, this cannot be set to "Enable". (When using a structure array, that member alone cannot be set to "Enable".)
*3 When using a string type as a member in a structured data type, that member cannot be set to "Enable".
*4 Structured data types that can be made public can be a maximum of 4 layers.
*5 When using a structure array as a member in a structure, that member cannot be set to "Enable".
*6 This is used in some cases by the CPU module side within a PLCopen MCFB program.
*7 When using a string type in an MCFB structure, that particular MCFB structure cannot be set.

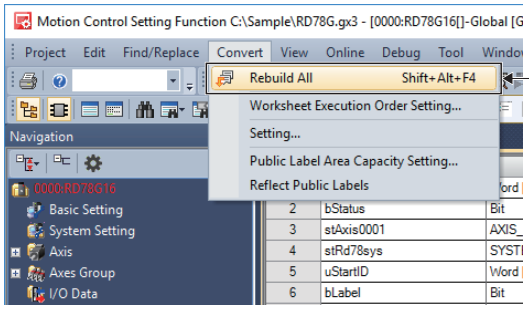
■Motion control attribute setting items

The items to set for the Motion control attribute are shown below.

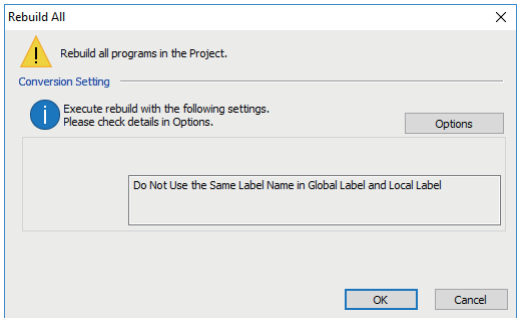
Setting item	Description
READ(Motion⇨)	Read only from the CPU module side. When writing is executed from the CPU module side, the value for the label that was made public is not reflected. (The Motion module side overwrites the value with the value stored on the Motion module.)
WRITE(⇨Motion)	Readable and writable from the CPU module side. When writing is executed from the Motion module side, the value for the label that was made public is not reflected. (The CPU module side overwrites the value with the value stored on the CPU module.)
—	Setting not possible. • This is automatically set by an engineering tool in some cases. • When the data is structured data type, the Motion control attribute setting for each member in the structure is applied.

Label conversion

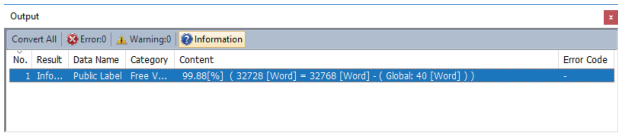
To fix the labels that were set, perform label conversion on the Motion Control Setting Function screen.



1. From the menu, select [Convert]⇨[Rebuild All] (1) to convert the set label data.



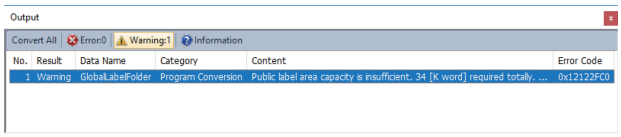
2. When the "Rebuild all programs in the Project." message appears, click the [OK] button.



3. After the "Rebuild All" operation completes normally, the free space in the public label area is displayed in "Information" in the Output window.

*1: If the Output window is not initially displayed, select [Display]⇨[Docking Window]⇨[Output] from the menu to display it.

*2: If a conversion error occurs, correct the setting based on the content of the error, then execute the conversion operation again after all errors have been removed.



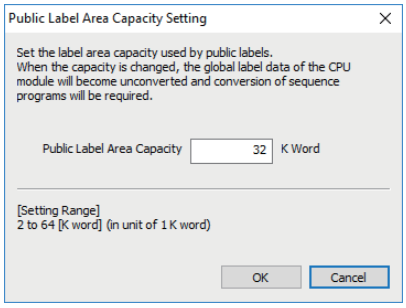
4. If the set public labels exceed the capacity of the public label area, a warning message is displayed in "Warning" in the Output window.
Confirm the label memory size in the Public Label Area Capacity Setting, then change the size so that it is within the label area capacity with the following methods.

- Increase the capacity of the public label area in Public Label Area Capacity Setting. (☞ Page 466 Public Label Area Capacity Setting)
- Reduce the number of public labels set to "Enable".

■Public Label Area Capacity Setting

Set the capacity of the public label area.

The capacity of the public label area can be changed in Public Label Area Capacity Setting when a different capacity is desired, or if the public labels set using "Rebuild All" exceed the capacity of the public label area.



1. From the menu, select [Convert]⇨[Public Label Area Capacity Setting] to display the "Public Label Area Capacity Setting" screen.
   Set the public label area capacity.
2. When the setting is complete, click the [OK] button.

*: If a warning message is displayed in the Output window, confirm the contents of the message, then set the label capacity so that it is within the range.

Point

- Changing the public label area capacity in Public Label Area Capacity Setting will cause the label data to become unconverted. As such, conversion must be executed for the project.
- Reduce the number of labels to be made public if a warning message stating that the public label area capacity exceeds 64k words is displayed during conversion operation.

Displayed items

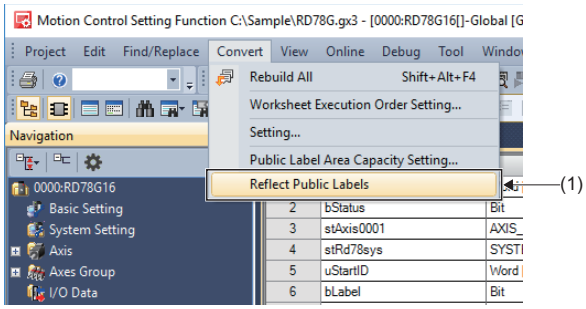
- Public Label Area Capacity Setting

Item	Setting range	Initial value
Public label area capacity	2 to 64k words	32k words

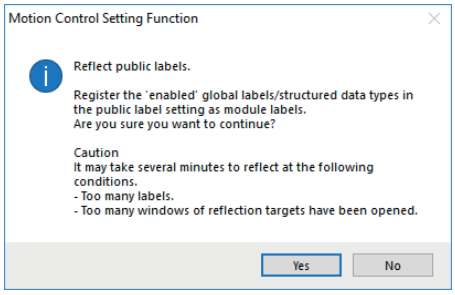
Public label reflection

Reflect the created public label information to the project on the CPU module side from the Motion Control Setting Function screen.

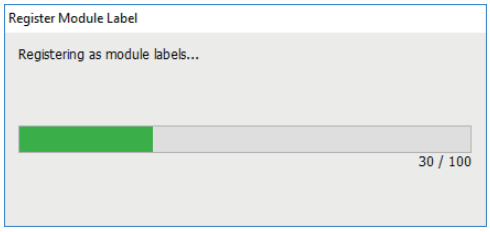
After reflection of the public labels completes, the label information is automatically registered to the CPU module side as module labels.



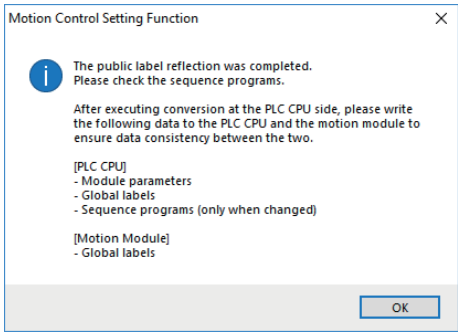
1. From the menu, select [Convert]⇨[Reflect Public Labels] (1).



2. When the "Reflect public labels." message is displayed, click the [Yes] button.



3. Reflect the public labels.
Labels whose public label was set to "Enable" are registered to the CPU module side as module labels.



4. After reflection of the public labels completes, the "The public label reflection was completed." message is displayed.
Click the [OK] button.

Point

Performing public label reflection will register the labels to the CPU module side as module labels and cause the project on the CPU module side to become unconverted. As such, conversion must be executed for the project on the CPU module side.

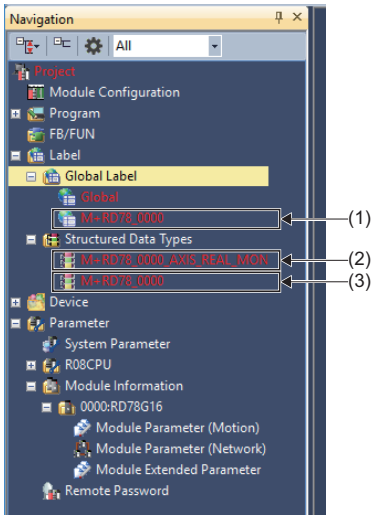
■Project on the CPU module side with public labels reflected

The data that is registered to a project on the CPU module side after reflection of public labels has completed is shown below.

- Project tree

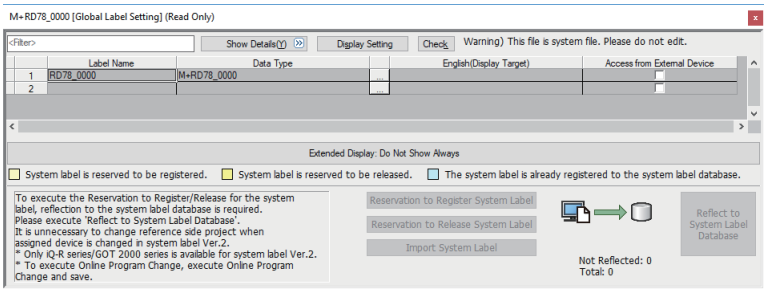
  Global labels and label data for structured data types set to public using the Motion Control Setting Function are added to the global label project tree or the structured data types project tree on the CPU module side.

<Project tree on the CPU module side>



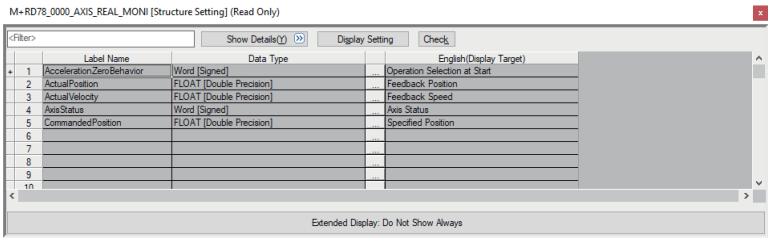
1. Global labels (1)

- The data is created in the global label project tree. (1 data item for each Motion module.)
- The data in the global label list is named "M+RD78_*****"[*1].
- "RD78_*****"[*1] labels are registered to the added global label list.
- The data type of registered labels is a "M+RD78_*****"[*1] structured data type (3).
- The added global label list is read only and cannot be edited.



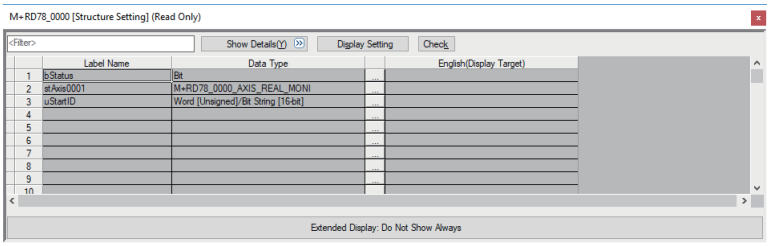
2. Public Motion module structured data types (2)

- Members of structures that were set to public using the Motion Control Setting Function are registered to the CPU module side as structures named "M+RD78_****_####"[*1*2].
- For members of structures, only members whose public label was set to "Enable" using the Motion Control Setting Function are registered.
- The added structure is read only and cannot be edited.



3. Public Motion module global labels (3)

- Label data set to public using the Motion Control Setting Function is registered to the CPU module side as individual members of a structure named "M+RD78_*****"[*1].
- The added structure is read only and cannot be edited.



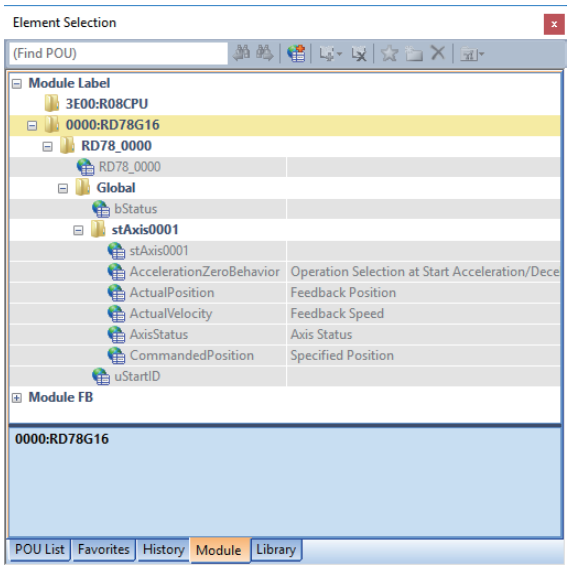
*1 ****=Start I/O No.

*2 ####=Structure name within the Motion Control Setting Function

- Element Selection window

  Motion module labels set to public using the Motion Control Setting Function are displayed as module labels in the Element Selection window.

<Element Selection window on the CPU module side>



- Labels set to public are registered as module labels in the [Module] tab in the Element Selection window.
- The module label data is named "****:RD78☐n" *1*2*3.
- A list of the label data that was set to public using the Motion Control Setting Function is displayed in the "****:RD78☐n" *1*2*3 tree.

*1 ****=Start I/O No.

*2 ☐=Motion module model name
- When set to RD78G: G
- When set to RD78GH: GH

*3 n=Number of Motion module control axes

Point

When using a label that was made public on the CPU module side, specify the label by entering "Module name + Start I/O No." at the start of the label.

<Example>

- When specifying the label of Axis 1(Axis0001)

RD78_0000.Axis0001.AxisRef

→ Start I/O No.

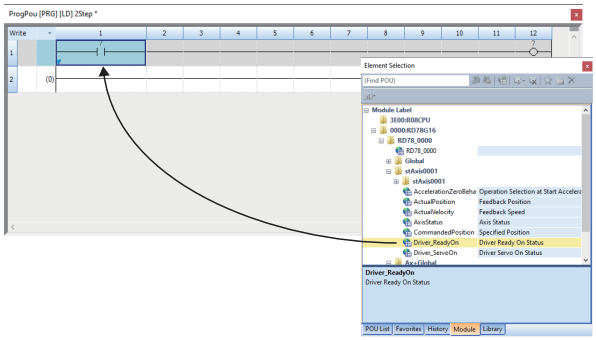
→ Module name

Sequence program creation

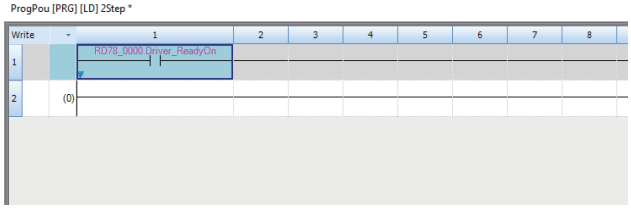
Create a sequence program using the Motion module labels and Motion control FBs that were made public.

■Inserting public labels

Create a program by dragging and dropping public labels registered to the CPU module side and assigning each label on the sequence program.



1. Select the label to be inserted from the Element Selection window, then drag and drop it into its assigned position.

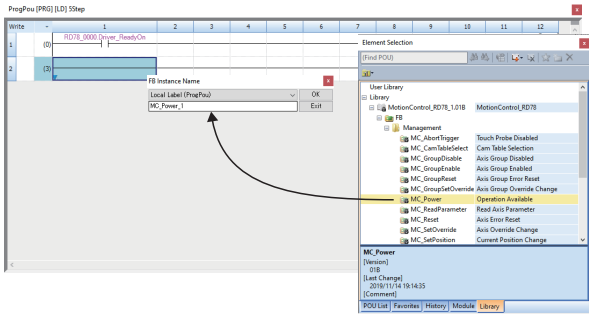


2. The label is inserted into the program work window.

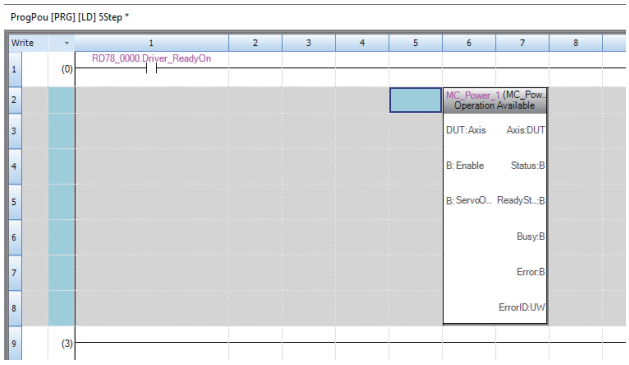
*: The label to be used is registered as a member of the structure, so the instance name of the structure is added.

■Inserting Motion control FBs

Create a program by dragging and dropping Motion control FBs registered to the CPU module side and assigning each FB on the sequence program.



1. Select the FB to be inserted from the Element Selection window and drag and drop it into its assigned position to display the FB instance name entry screen.
2. Set the instance name to be registered, then click the [OK] button.



3. The FB is inserted into the program work window.

*: Inserting an FB from the Element Selection window causes the instance name of the inserted FB to be registered as a label name in the local label setting list.

Point

The Motion module FB library for the Motion control FBs must be registered to the engineering tool library before creating a program.

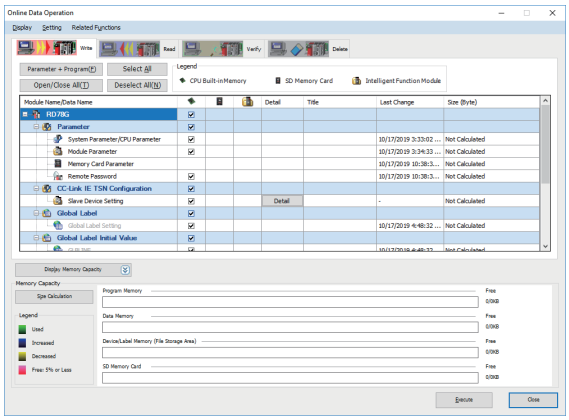
For registering the Motion module FB library, refer to the following.

☞ Page 472 Registering the Motion module FB library

Writing the created program

Write the program, global labels, and module parameters created using on the CPU module side and Motion module side to the CPU module and the Motion module. After executing writing to both modules, perform a reset of the CPU module.

■Writing to the CPU module

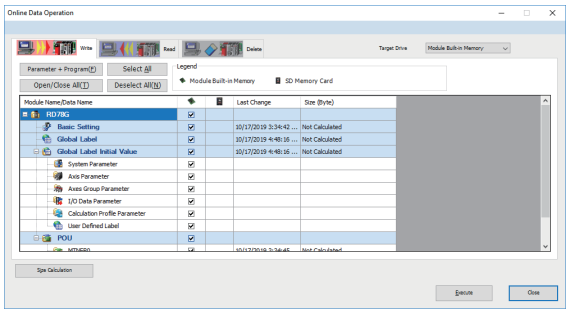


1. From the menu, select [Online]⇨[Write to PLC] to display the "Online Data Operation" screen. Select the data to perform the writing on, then click the [Execute] button. Execute the writing to the PLC CPU.

Point

If the "Reflect Public Labels" operation was executed using the Motion Control Setting Function, be sure to select the "Module Parameter" and "Global Label" data.

■Writing to the Motion module



1. Using the Motion Control Setting Function, select [Online]⇨[Write to Module] from the menu to display the "Online Data Operation" window. Select the data to write, then click the [Execute] button to perform writing to the Motion module.

Point

If the "Reflect Public Labels" operation was executed using the Motion Control Setting Function, be sure to select the "Global Label" data.

⚠Caution

- If a Motion label was made public, be sure to write the data to both the CPU module side and the Motion module side. An unexpected operation may occur If the label data on the CPU module side and the label data on the Motion module side do not match.

Registering the Motion module FB library

When creating a sequence program using Motion control FBs on the CPU module side, the Motion module FB library must be registered to the library of the engineering tool.

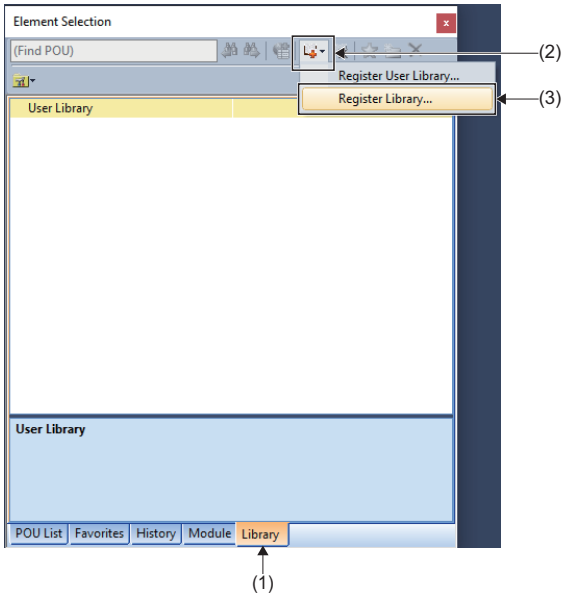
This section describes the procedure for registering the Motion module FB library to the engineering tool.

Point

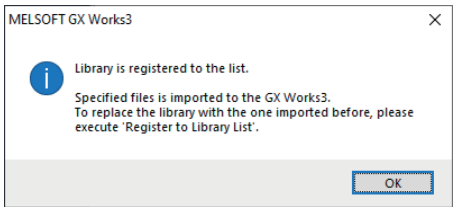
For the Motion module FB library, please consult your local Mitsubishi representative.

Procedure for registering the Motion module FB library

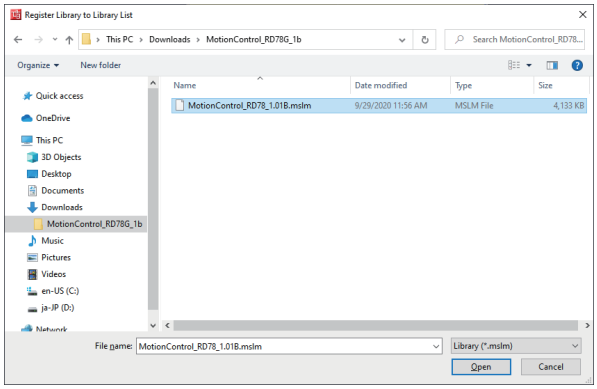
Register the Motion module FB library to the library of the engineering tool.



1. Select the [Library] tab(1) in the Element Selection window.
2. Click [Register to Library List](2)⇨[Register Library](3).

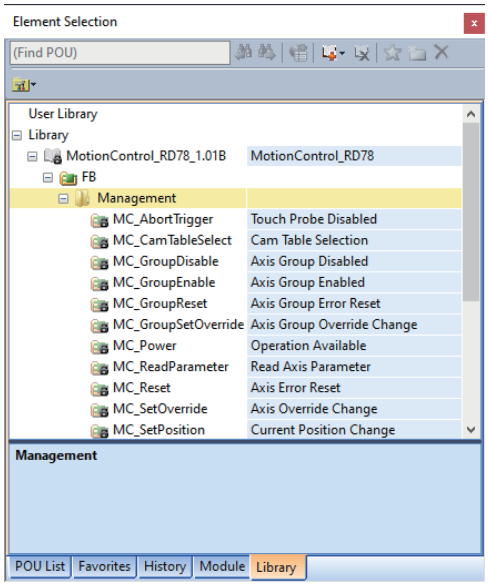


3. When the "Library is registered to the list." window appears, click the [OK] button.



4. The "Register Library to Library List" window appears. Select the file for the Motion module FB library to be registered (MotionControl_RD78_****[*1].mslm), then click [Open].

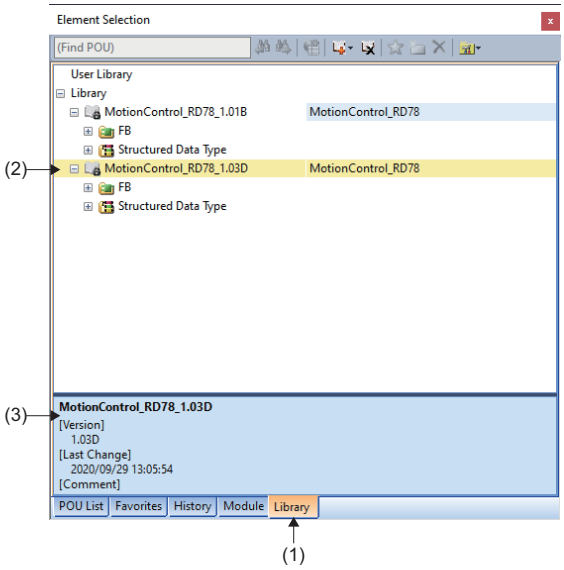
*1: **** = Motion module FB library version



5. The registered Motion module FB library is displayed as a list in the Element Selection window.

Updating the Motion module FB library version

Register the latest version of the Motion module FB library and update the library POU.



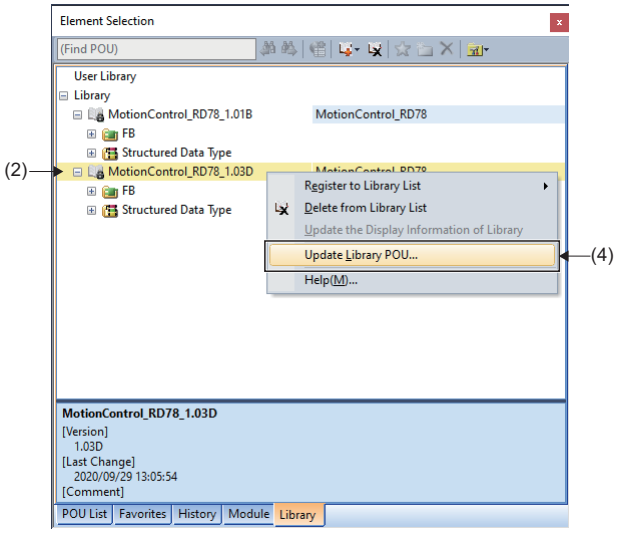
1. Select the [Library] tab(1) in the Element Selection window.

2. Register the lastest version of the Motion module FB library.

*: For the procedure for registering the Motion module FB library, refer to the following.
☞ Page 472 Procedure for registering the Motion module FB library

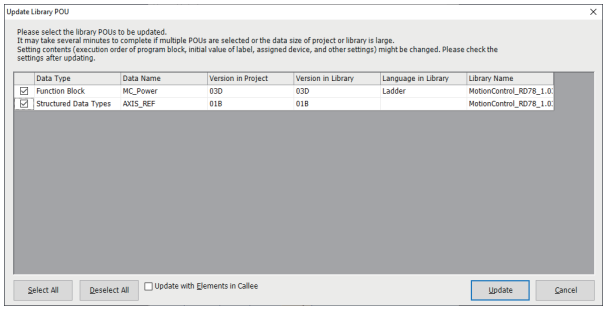
*: To confirm the current version, select a registered library (MotionControl_RD78_****[*1])(2). This will display the library version at the bottom of the Element Selection window (3).

*1: **** = Motion module FB library version



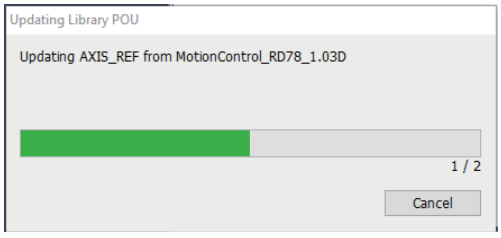
3. Right click the library to be updated (MotionControl_RD78_****[*1])(2), then select [Update Library POU](4).

*1: **** = Motion module FB library version

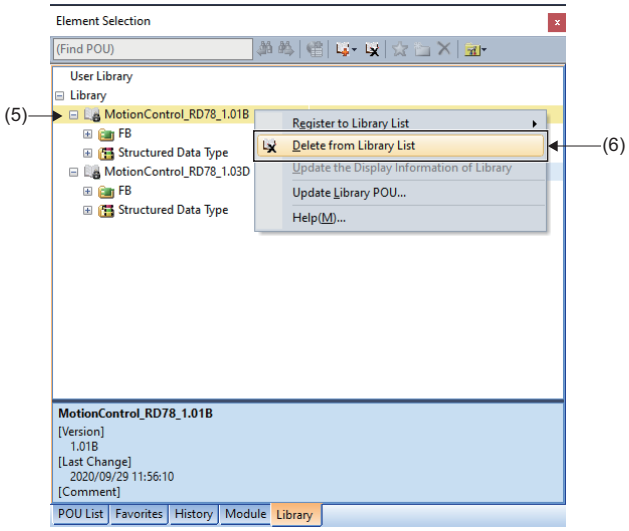


4. The "Update Library POU" screen appears.
A list of the FBs and structures being used in the project is displayed.

5. Place a check next to the FBs and structures that need to be updated.
Click the [Update] button.

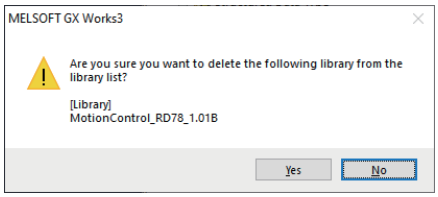


6. Execute the update of the library.
When the update is complete, the "Updating Library POU" screen will close.

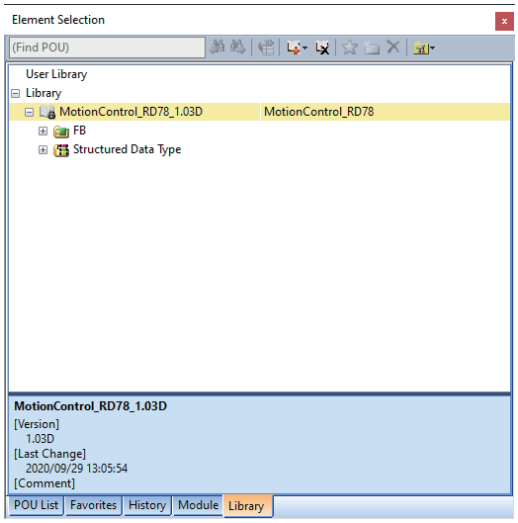


7. To delete the old version of the Motion module FB libary, right click the library to be deleted (MotionControl_RD78_****[*1])(5), then select [Delete from Library List](6).

*1: **** = Motion module FB library version



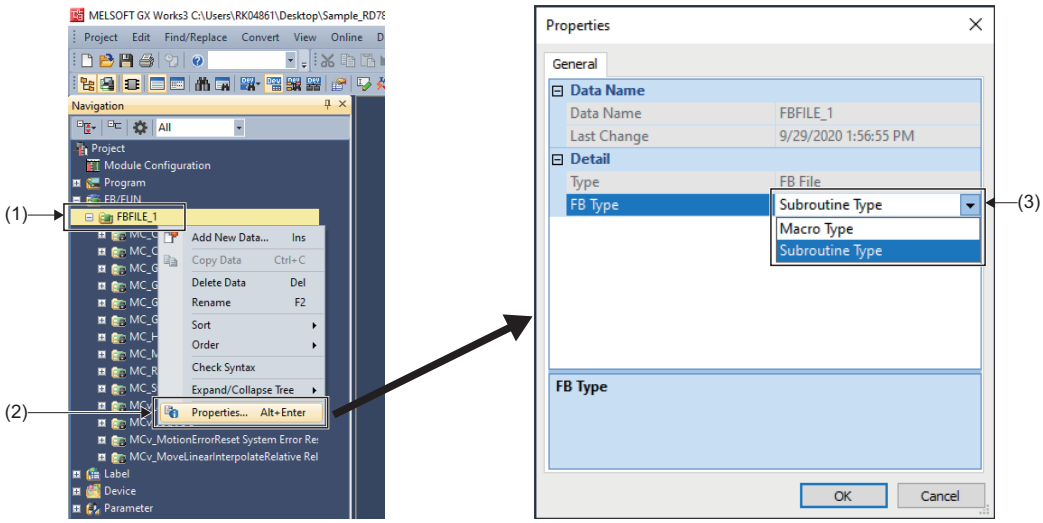
8. When the "Are you sure you want to delete the following library from the library list?" message appears, click the [Yes] button.



9. The old version of the Motion module FB library is deleted from the library list.

■Precautions

- Performing a library update causes the project on the CPU module side to become unconverted. As such, conversion must be executed for the project on the CPU module side.
- When performing a library update, the library update cannot be performed if the FB type is "Macro Type". Change the FB type to "Subroutine Type" before performing the library update.
  To change the FB type, right click [FB/FUN]⇨[FBFILE_1](1) in the navigation window, then select [Properties](2) to display the Properties screen. Change the FB type(3) to "Subroutine Type" from the Properties screen.



Data transfers using the Motion module buffer memory

Data transfers between the CPU module and the Motion module

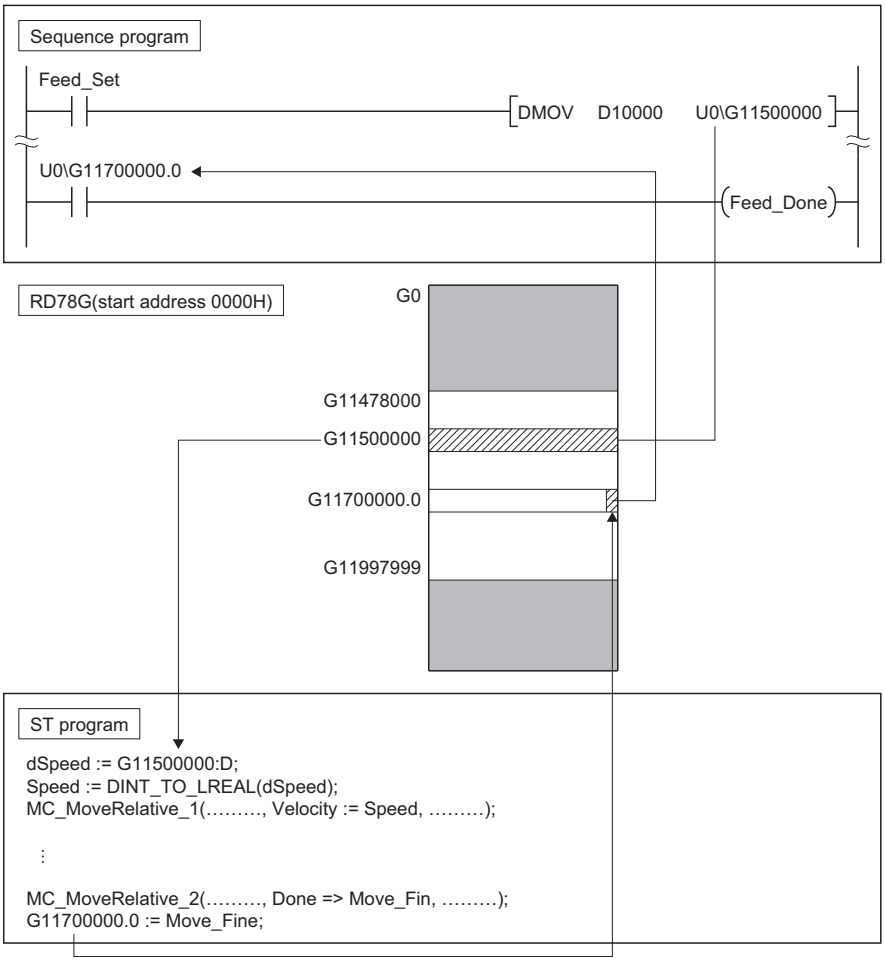
Data can be transferred between the CPU module and the Motion module using the buffer memory of the Motion module.
The devices used for data transfers are shown below.

Module	Device used
CPU module	Un\Gn(module access device)
Motion module	Gn(buffer memory)

The usable buffer memory range for data transfers is shown below. Ensure that the buffer memory used is not outside the usable range.

Setting range
G11478000 to G11997999 (520000 words)

Ex.
For a data transfer that uses the buffer memory of the Motion module



For a program using labels

While label programming can be performed by assigning global labels to the Motion module buffer memory, labels cannot be cross-referenced between the CPU module and the Motion module.

Point

A double-precision real number (LREAL-type) cannot be transferred to the buffer memory by ST instruction when sending data from the CPU module to the Motion module. The reading/writing of double-precision real number-type labels can be performed by using a dedicated instruction.

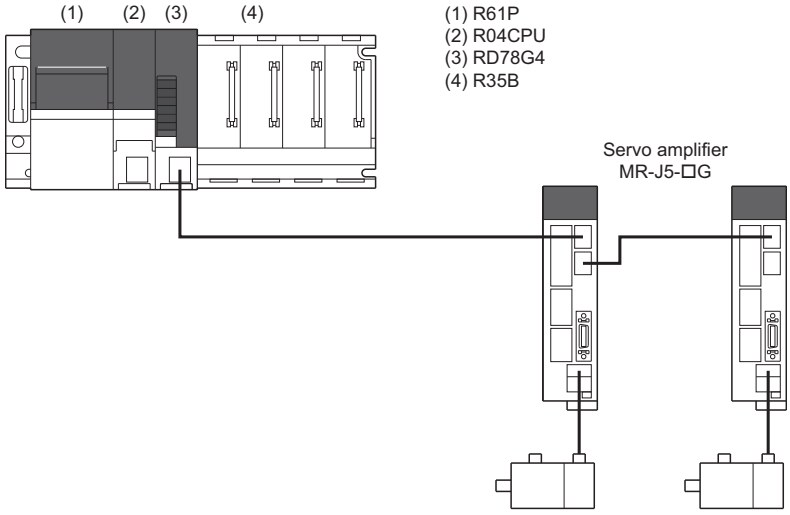
APPENDICES

Appendix 1 FBD/LD Sample Program

A sample program created using FBD/LD language on the CPU module side is shown below.

System configuration

The system configuration used in the sample program is shown below.



Settings in GX Works3

The following settings are performed in GX Works3.

- System parameter
- CPU parameter
- Module parameter (Network)
- Program

System parameter

The following items are set in system parameter.

■Base/power/extension cable setting

Base No.	Base	Slots	Power supply module
Main	R35B	5	R61P

■I/O assignment setting

Slot		Module name	Module status setting	Points	Start XY	CPU module operation at error detection
Base	CPU	R04CPU (host station)	—	—	3E00	—
	0(0-0)	RD78G4	No setting	32 points	0000	Critical: Stop, Moderate: Continue

CPU parameter

The following items are set in CPU parameter.

■Memory/device setting

Item	Setting
Link direct device setting	Extended mode (iQ-R series mode)

■Program setting

Execute order	Program name	Execution type		Refresh group setting
		Type	Detailed setting information	
1	MAIN	Scan	—	(Do not set)

Module parameter (Network)

The following items are set in module parameter (Network).

■Basic setting

• Network configuration settings

Model name	STA#	Station type	Motion control station	RWr setting Points	RWw setting Points	Parameter automatic setting	IP address
Host station	0	Master station	—	—	—	—	192.168.3.253
MR-J5-G	1	Remote station	Checked	24	20	Checked	192.168.3.1
MR-J5-G	2	Remote station	Checked	24	20	Checked	192.168.3.2

Device

The following devices are used in the program.

Device	Comment
X0	READY
X1	Synchronization flag
Y0	PLC READY
SM400	AlwaysON

Global label

The settings for global labels used in the program are shown below.

<Filter> | Easy Display | Display Setting | Check

	Label Name	Data Type		Class	Assign (Device/Label)	Initial Value	Constant	English(Display Target)
1	G_bSRVOFF	Bit		VAR_GLOBAL				Servo OFF
2	G_bJogF1	Bit		VAR_GLOBAL				JOG Positive rotation command Axis0001
3	G_bJogR1	Bit		VAR_GLOBAL				JOG Reverse rotation command Axis0001
4	G_bJogF2	Bit		VAR_GLOBAL				JOG Positive rotation command Axis0002
5	G_bJogR2	Bit		VAR_GLOBAL				JOG Reverse rotation command Axis0002
6	G_bHoming1CMD	Bit		VAR_GLOBAL				Homing command Axis0001
7	G_bHoming2CMD	Bit		VAR_GLOBAL				Homing command Axis0002
8	G_bHoming3CMD	Bit		VAR_GLOBAL				Homing command VirtualAxis0001
9	G_bPosCMD	Bit		VAR_GLOBAL				Single axis positioning start
10	G_bContPosCMD	Bit		VAR_GLOBAL				Single axis continuous positioning start
11	G_bInterpolationCMD	Bit		VAR_GLOBAL				2-axis linear interpolation control start
12	G_bSyncCMD	Bit		VAR_GLOBAL				Synchronous control start
13	G_bErrorReset	Bit		VAR_GLOBAL				Error reset
14	G_bSysErrorReset	Bit		VAR_GLOBAL				System error reset
15	G_leJogVelocity	FLOAT [Double Precision]		VAR_GLOBAL				JOG Velocity
16	G_bHoming1Req	Bit		VAR_GLOBAL				Homing start request Axis0001
17	G_bHoming2Req	Bit		VAR_GLOBAL				Homing start request Axis0002
18	G_bHoming3Req	Bit		VAR_GLOBAL				Homing start request VirtualAxis0001
19	G_bPosReq	Bit		VAR_GLOBAL				Single axis positioning start request
20	G_bContPosReq	Bit		VAR_GLOBAL				Single axis continuous positioning start request
21	G_bInterpolationReq	Bit		VAR_GLOBAL				2-axis linear interpolation control start request
22	G_bSyncReq	Bit		VAR_GLOBAL				Synchronous control start request
23	G_bJog1Busy	Bit		VAR_GLOBAL				JOG operation in progress Axis0001
24	G_bJog2Busy	Bit		VAR_GLOBAL				JOG operation in progress Axis0002

Program

The programs created using FBD/LD language are shown below.

Program file name	Program name	Processing contents	Reference
MAIN	ServoON_Jog	• Performs servo ON for all axes. • Performs JOG operation for Axis 1 (Axis0001) and Axis 2 (Axis0002).	☞ Page 480 ServoON_Jog
	Positioning	• Performs single axis positioning control using the relative value of Axis 1 (Axis0001).	☞ Page 482 Positioning
	ContinuousPositioning	• Performs single axis continuous positioning control using the relative values of Axis 1 (Axis0001) and Axis 2 (Axis0002).	☞ Page 483 ContinuousPositioning
	LinearInterpolation	• Performs 2-axis linear interpolation control for the axes group 1 (AxesGroup001) of Axis 1 (Axis0001) and Axis 2 (Axis0002).	☞ Page 485 LinearInterpolation
	Homing	• Performs homing for Axis 1 (Axis0001), Axis 2 (Axis0002), and Axis 301 (VirtualAxis0001).	☞ Page 488 Homing
	ErrorReset	• Performs error reset of the Motion system for Axis 1 (Axis0001), Axis 2 (Axis0002), Axis 301 (VirtualAxis0001), Axis 401 (LinkAxis0001), Axis 402 (LinkAxis0002), and Axes group 1 (AxesGroup001).	☞ Page 489 ErrorReset
	Synchronous	• Performs synchronous control.	☞ Page 490 Synchronous

■ServoON_Jog

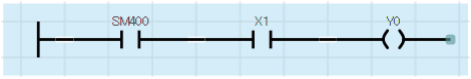
• Local labels

<Filter> | Easy Display | Display Setting | Check

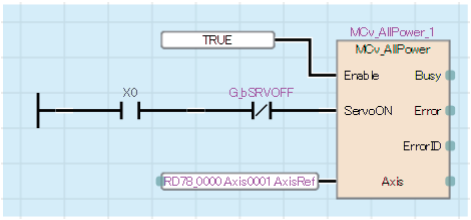
	Label Name	Data Type		Class	Initial Value	Constant	English(Display Target)
1	MCv_AllPower_1	MCv_AllPower	...	VAR			All axes servo ON FB
2	MCv_Jog_1	MCv_Jog	...	VAR			JOG operation FB Axis0001
3	MCv_Jog_2	MCv_Jog	...	VAR			JOG operation FB Axis0002
4	leJogAcceleration	FLOAT [Double Precision]	...	VAR			JOG acceleration
5	leJogDeceleration	FLOAT [Double Precision]	...	VAR			JOG deceleration
6	leJogJerk	FLOAT [Double Precision]	...	VAR			JOG jerk

• ProgramBody

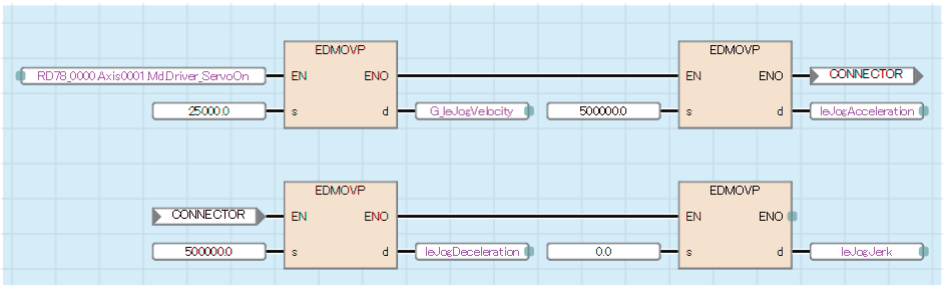
• PLC Ready ON



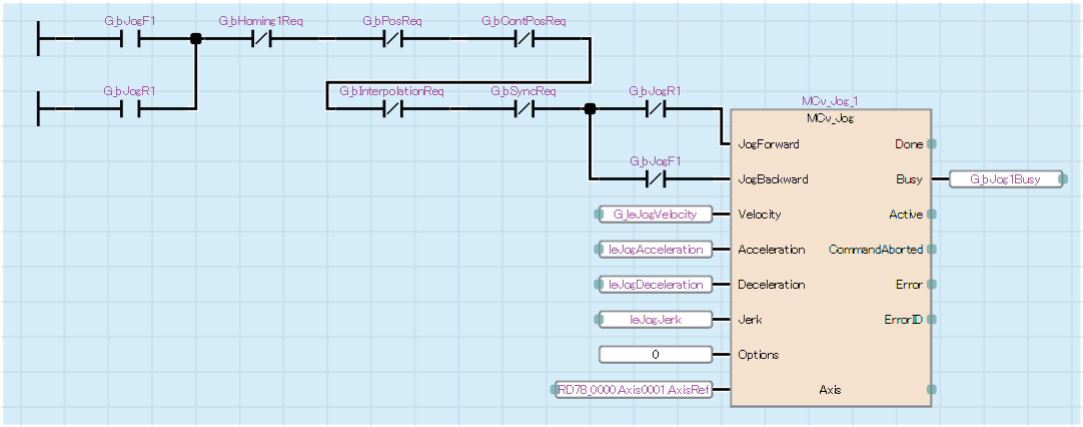
• All axes servo ON



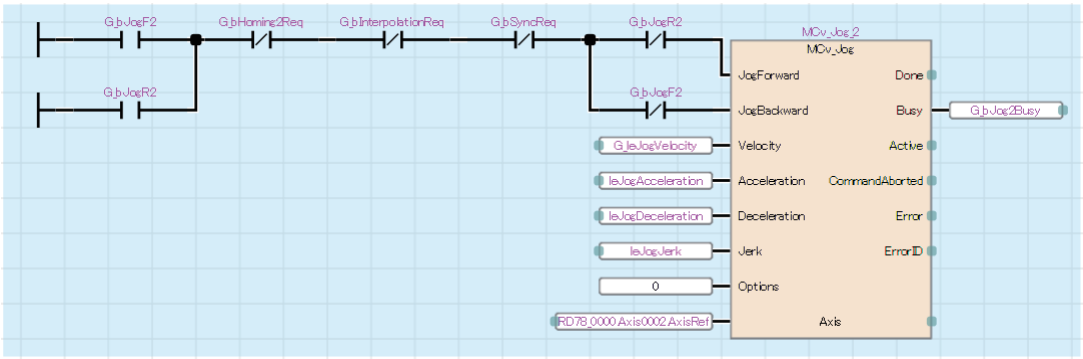
• Data setting for JOG operation



• JOG Axis0001



• JOG Axis0002



A

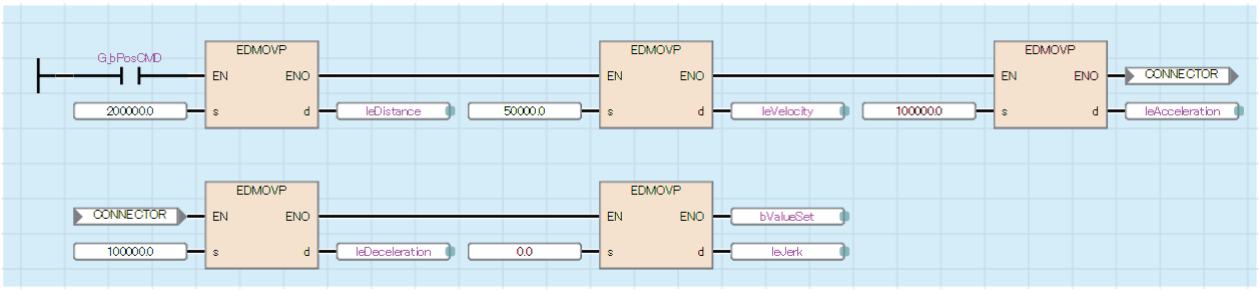
■Positioning

- Local labels

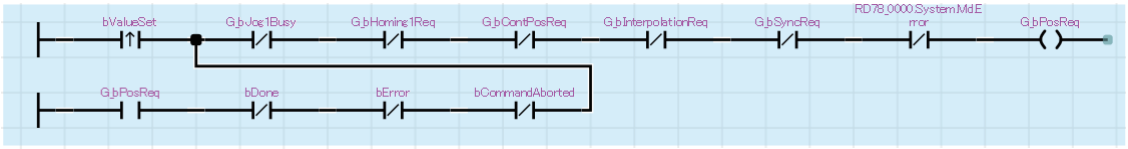
<Filter> | Easy Display | Display Setting | Check

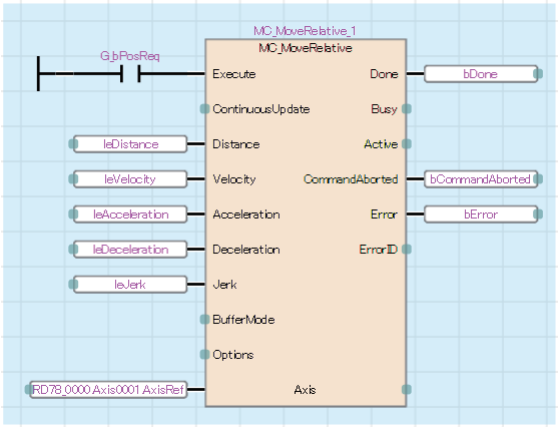
	Label Name	Data Type		Class	Initial Value	Constant	English(Display Target)
1	MC_MoveRelative_1	MC_MoveRelative	...	VAR			Relative value positioning FB
2	leDistance	FLOAT [Double Precision]	...	VAR			Distance
3	leVelocity	FLOAT [Double Precision]	...	VAR			Velocity
4	leAcceleration	FLOAT [Double Precision]	...	VAR			Acceleration
5	leDeceleration	FLOAT [Double Precision]	...	VAR			Deceleration
6	leJerk	FLOAT [Double Precision]	...	VAR			Jerk
7	bDone	Bit	...	VAR			Relative value positioning FB Done output
8	bError	Bit	...	VAR			Relative value positioning FB Error output
9	bValueSet	Bit	...	VAR			Variable set complete
10	bCommandAborted	Bit	...	VAR			Relative value positioning FB CommandAborted output

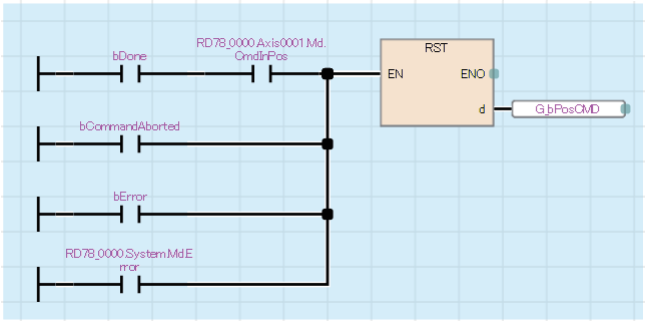
- ProgramBody
  - Data setting for single axis positioning



  - Relative value positioning







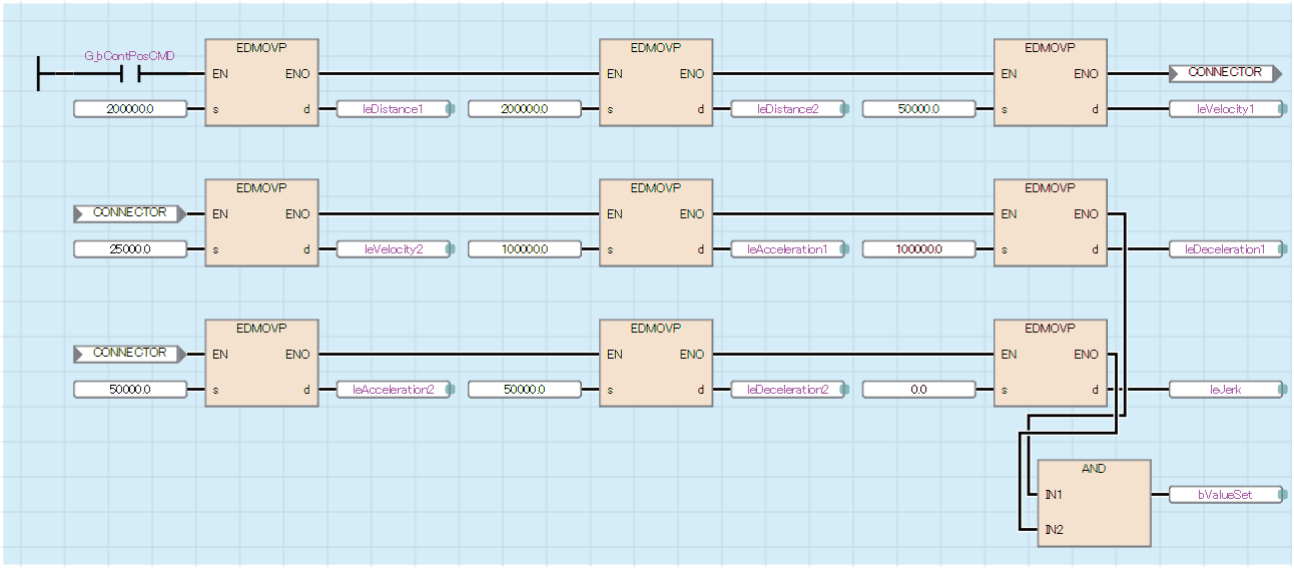
■ContinuousPositioning

- Local labels

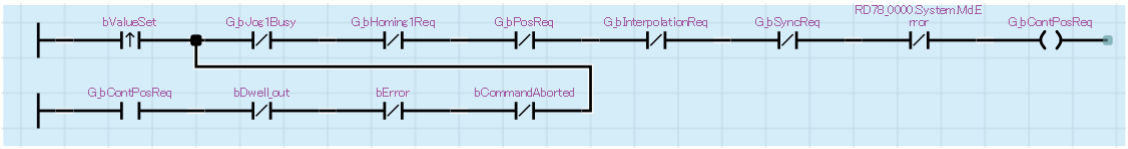
<Filter> | Easy Display | Display Setting | Check

	Label Name	Data Type		Class	Initial Value	Constant	English(Display Target)
1	leDistance1	FLOAT [Double Precision]	...	VAR			Distance1
2	leVelocity1	FLOAT [Double Precision]	...	VAR			Velocity1
3	leDistance2	FLOAT [Double Precision]	...	VAR			Distance2
4	leVelocity2	FLOAT [Double Precision]	...	VAR			Velocity2
5	leAcceleration1	FLOAT [Double Precision]	...	VAR			Acceleration1
6	leDeceleration1	FLOAT [Double Precision]	...	VAR			Deceleration1
7	leAcceleration2	FLOAT [Double Precision]	...	VAR			Acceleration2
8	leDeceleration2	FLOAT [Double Precision]	...	VAR			Deceleration2
9	leJerk	FLOAT [Double Precision]	...	VAR			Jerk
10	MC_MoveRelative_1	MC_MoveRelative	...	VAR			Relative value positioning FB1
11	MC_MoveRelative_2	MC_MoveRelative	...	VAR			Relative value positioning FB2
12	bError	Bit	...	VAR			Relative value positioning FB Error output
13	TON_1	TON	...	VAR			On-delay timer FB
14	bDwell_out	Bit	...	VAR			Timer output
15	bDwell_in	Bit	...	VAR			Timer input
16	bValueSet	Bit	...	VAR			Variable set complete
17	bCommandAborted	Bit	...	VAR			Relative value positioning FB CommandAborted output

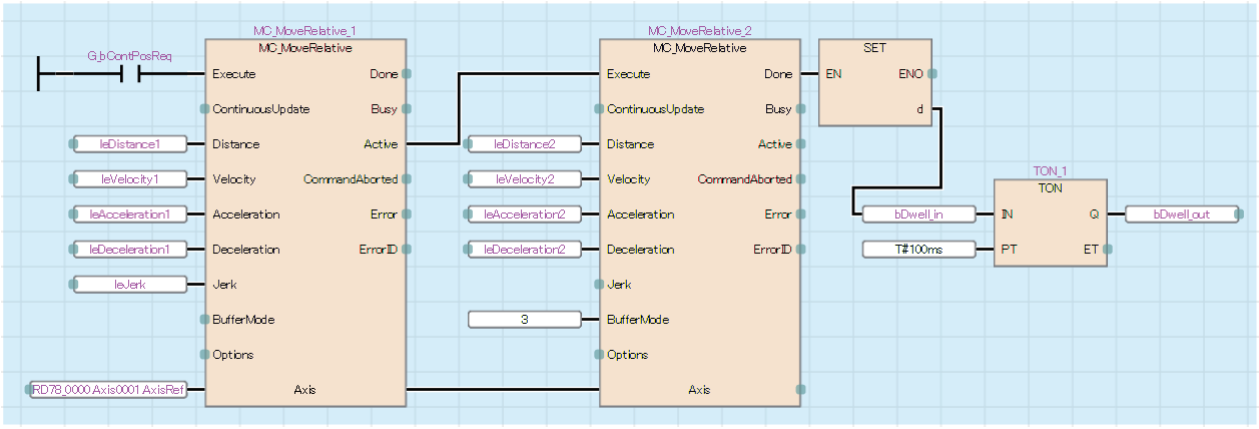
- ProgramBody
  - Data setting for single axis continuous positioning



  - Single axis continuous positioning + Buffer mode + Dwell

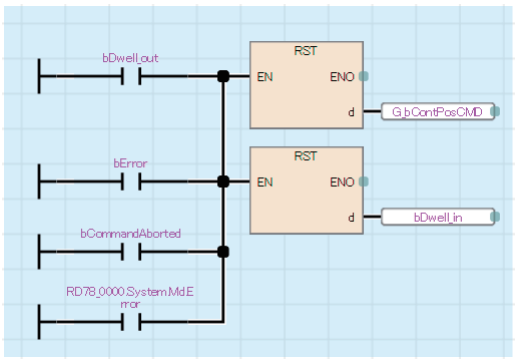


  - Relative value positioning 1/Relative value positioning 2/Dwell

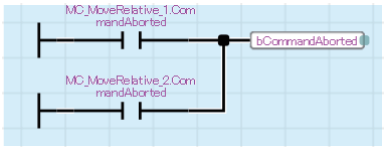


A

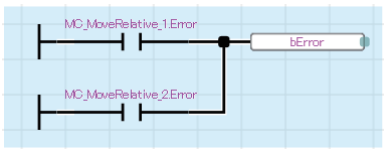
• Resets the positioning start request when positioning is completed or an error occurs



• FB CommandAborted output



• FB Error output



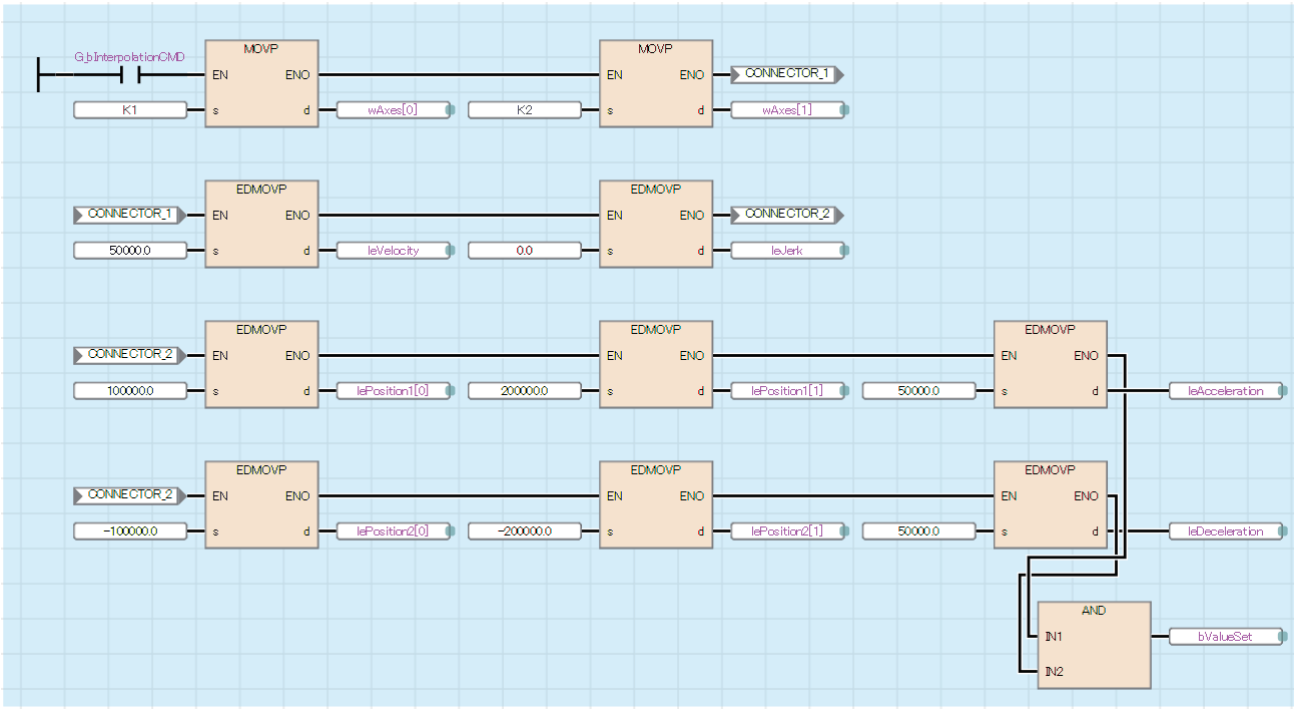
■LinearInterpolation

- Local labels

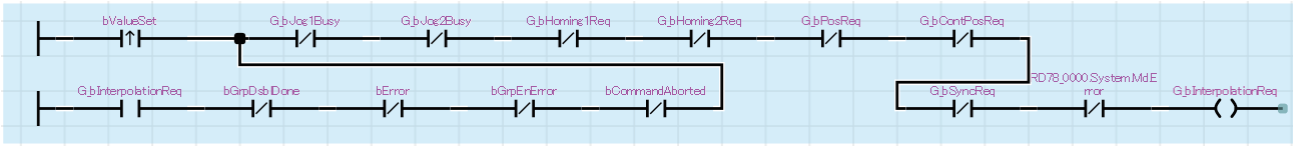
<Filter> | Easy Display | Display Setting | Check

	Label Name	Data Type		Class	Initial Value	Constant	English(Display Target)
1	wAxes	Word [Signed](0..15)	...	VAR			Interpolation axis
2	lePosition1	FLOAT [Double Precision](0..15)	...	VAR			Position data 1
3	lePosition2	FLOAT [Double Precision](0..15)	...	VAR			Position data 2
4	leVelocity	FLOAT [Double Precision]	...	VAR			Velocity
5	leAcceleration	FLOAT [Double Precision]	...	VAR			Acceleration
6	leDeceleration	FLOAT [Double Precision]	...	VAR			Deceleration
7	leJerk	FLOAT [Double Precision]	...	VAR			Jerk
8	MC_GroupEnable_1	MC_GroupEnable	...	VAR			Axes group enable FB
9	MCv_MoveLinearInterpolateRelative_1	MCv_MoveLinearInterpolateRelative	...	VAR			Relative value linear interpolation control FB1
10	MCv_MoveLinearInterpolateRelative_2	MCv_MoveLinearInterpolateRelative	...	VAR			Relative value linear interpolation control FB2
11	MC_GroupDisable_1	MC_GroupDisable	...	VAR			Axes group disable FB
12	bDone2	Bit	...	VAR			Relative value linear interpolation control FB2 Done output
13	bGrpEnError	Bit	...	VAR			Axes group enable FB Error output
14	bError	Bit	...	VAR			Relative value linear interpolation control FB Error output
15	bGrpDsblDone	Bit	...	VAR			Axes group disable FB Done output
16	TON_1	TON	...	VAR			On-delay timer FB
17	bDwell_in	Bit	...	VAR			Timer input
18	bDwell_out	Bit	...	VAR			Timer output
19	bValueSet	Bit	...	VAR			Variable set complete
20	bCommandAborted	Bit	...	VAR			Relative value linear interpolation control FB CommandAborted output
21	bDone_Set	Bit	...	VAR			Done
22	bCommandAborted_Set	Bit	...	VAR			FB abortion of execution

- ProgramBody
  - Data setting for 2-axis linear interpolation control

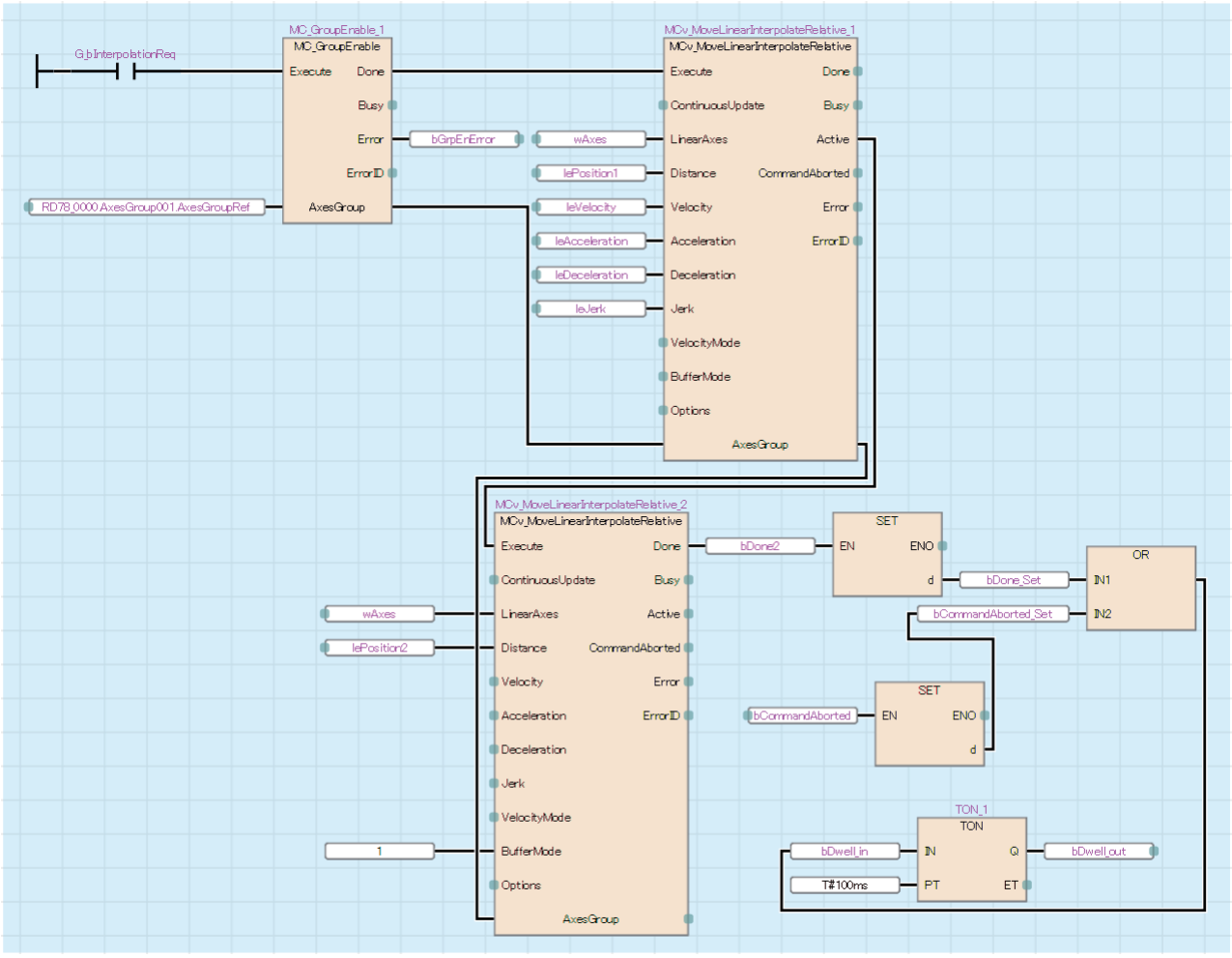


  - Linear interpolation control + Buffer mode + Dwell

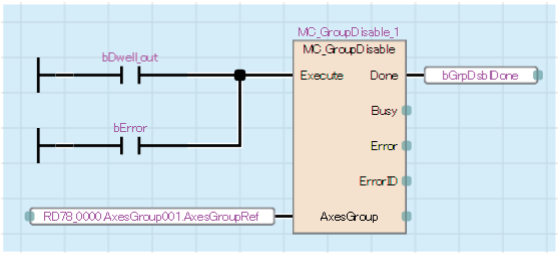


A

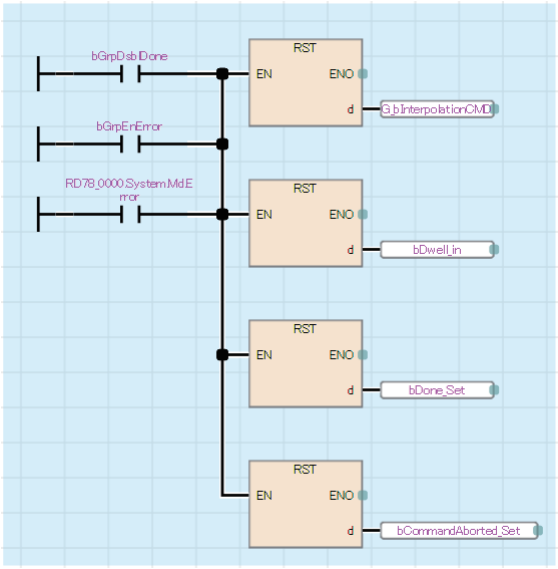
• Axes group enabled/Linear interpolation control 1/Linear interpolation control 2/FB done/FB abortion of execution/Dwell



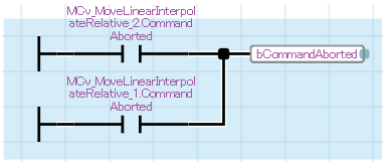
• Axes group disabled



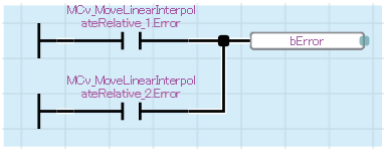
• Resets the start command and timer input



• FB CommandAborted output



• FB Error output



A

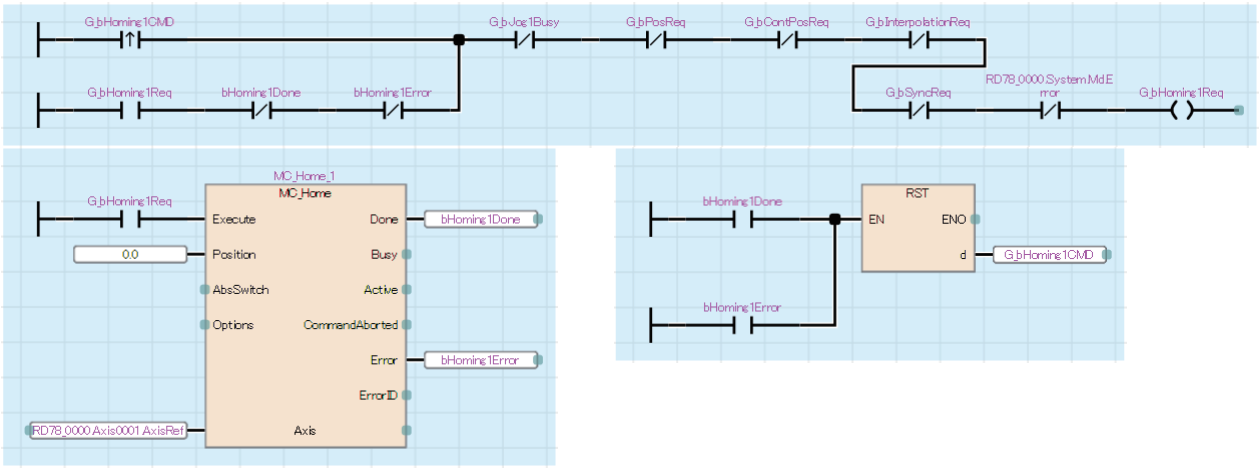
■Homing

- Local labels

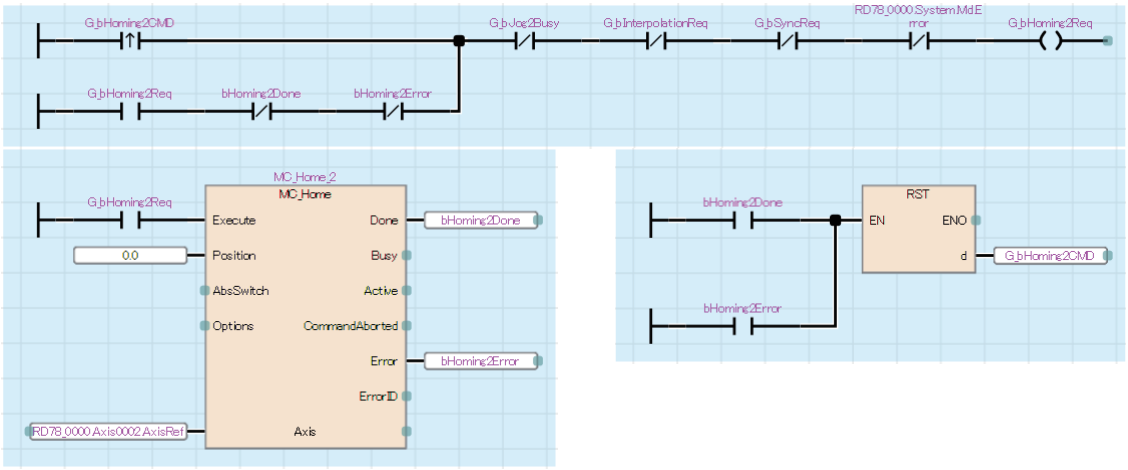
<Filter> | Easy Display | Display Setting | Check

	Label Name	Data Type		Class	Initial Value	Constant	English(Display Target)
1	MC_Home_1	MC_Home	...	VAR			Homing FB1 Axis0001
2	MC_Home_2	MC_Home	...	VAR			Homing FB2 Axis0002
3	MC_Home_3	MC_Home	...	VAR			Homing FB3 VirtualAxis0001
4	bHoming1Done	Bit	...	VAR			Homing FB1 Done output
5	bHoming1Error	Bit	...	VAR			Homing FB1 Error output
6	bHoming2Done	Bit	...	VAR			Homing FB2 Done output
7	bHoming2Error	Bit	...	VAR			Homing FB2 Error output
8	bHoming3Done	Bit	...	VAR			Homing FB3 Done output
9	bHoming3Error	Bit	...	VAR			Homing FB3 Error output

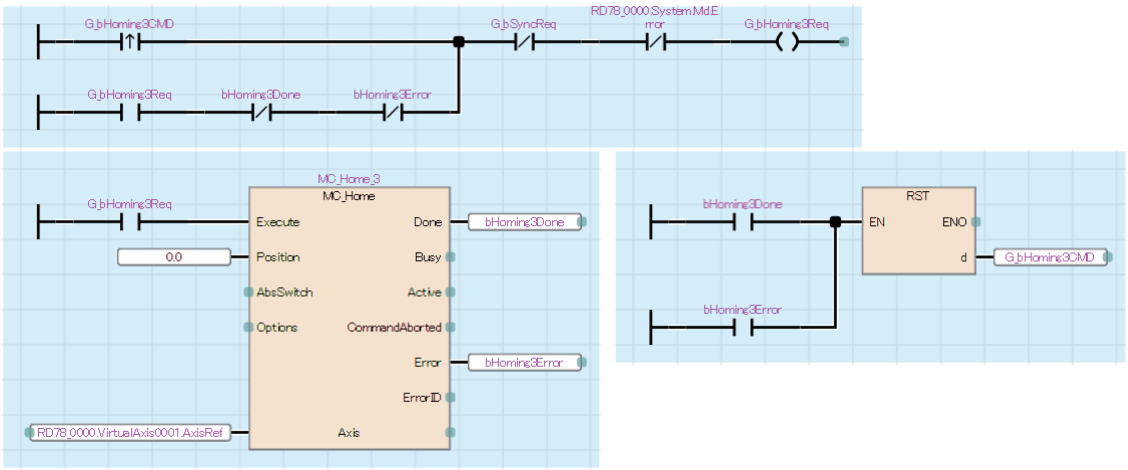
- ProgramBody
  - Axis0001 Homing



  - Axis0002 Homing



  - VirtualAxis0001 Homing



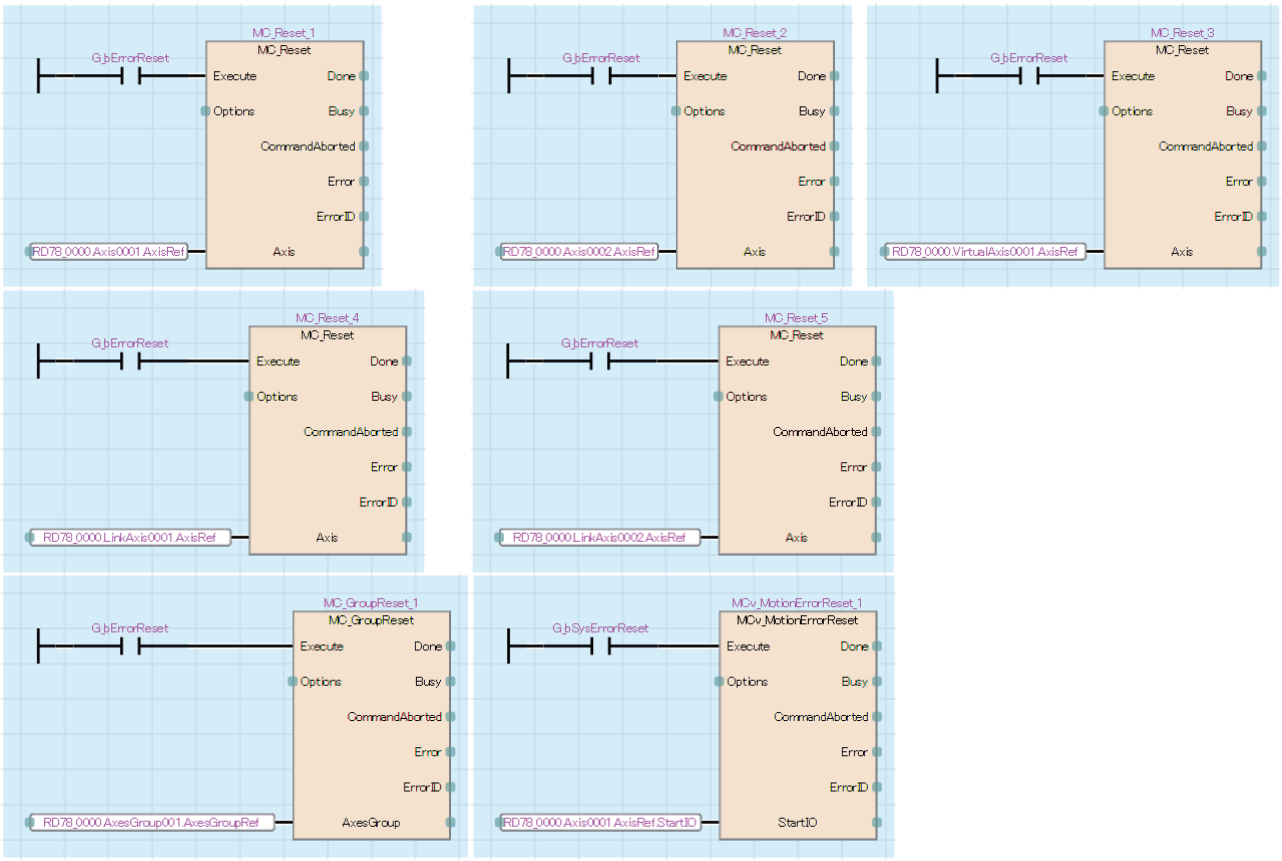
■ErrorReset

- Local labels

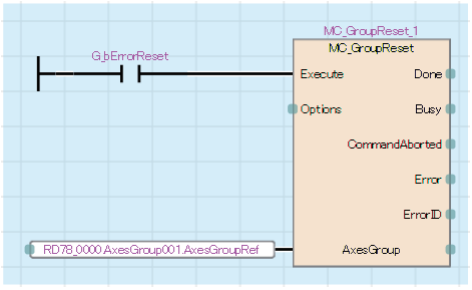
<Filter> | Easy Display | Display Setting | Check

	Label Name	Data Type		Class	Initial Value	Constant	English(Display Target)
1	MC_Reset_1	MC_Reset	...	VAR			Axis error reset FB1 Axis0001
2	MC_Reset_2	MC_Reset	...	VAR			Axis error reset FB2 Axis0002
3	MC_Reset_3	MC_Reset	...	VAR			Axis error reset FB3 VirtualAxis0001
4	MC_Reset_4	MC_Reset	...	VAR			Axis error reset FB4 LinkAxis0001
5	MC_Reset_5	MC_Reset	...	VAR			Axis error reset FB5 LinkAxis0002
6	MC_GroupReset_1	MC_GroupReset	...	VAR			Axes group error reset FB
7	MCv_MotionErrorReset_1	MCv_MotionErrorReset	...	VAR			System error reset FB

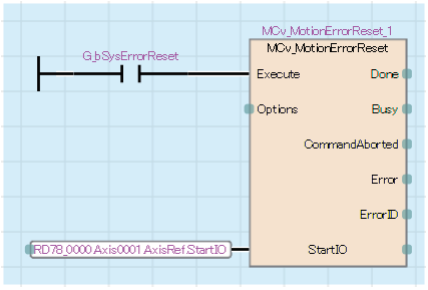
- ProgramBody
  - Axis error reset



  - Axes group error reset



  - System error reset



A

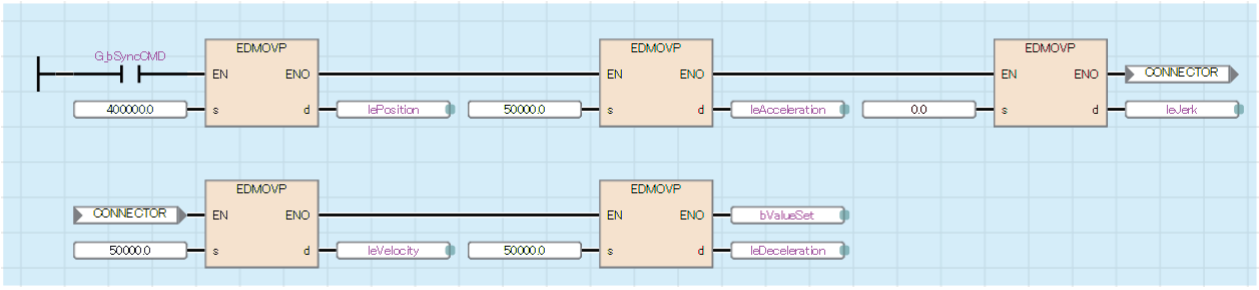
■Synchronous

- Local labels

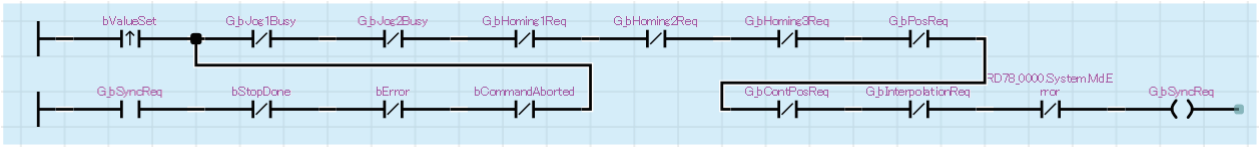
<Filter> | Easy Display | Display Setting | Check

	Label Name	Data Type	Class	Initial Value	Constant	English(Display Target)
1	leVelocity	FLOAT [Double Precision]	VAR			Velocity
2	leAcceleration	FLOAT [Double Precision]	VAR			Acceleration
3	leDeceleration	FLOAT [Double Precision]	VAR			Deceleration
4	leJerk	FLOAT [Double Precision]	VAR			Jerk
5	lePosition	FLOAT [Double Precision]	VAR			Distance
6	MC_GearIn_1	MC_GearIn	VAR			Gear operation FB
7	MC_CombineAxes_1	MC_CombineAxes	VAR			FB combining the motion of two master axes
8	MC_CamIn_1	MC_CamIn	VAR			Cam operation FB
9	CamID	MC_CAM_ID	VAR			Cam ID
10	bInSync	Bit	VAR			Cam operation FB inSync output
11	MC_MoveRelative_1	MC_MoveRelative	VAR			Relative value positioning FB
12	bError	Bit	VAR			Relative value positioning FB Error output
13	bDone1	Bit	VAR			Relative value positioning FB1 Done output
14	MC_MoveRelative_2	MC_MoveRelative	VAR			Relative value positioning FB
15	bDone2	Bit	VAR			Relative value positioning FB2 Done output
16	bStopDone	Bit	VAR			Axis Stop complete
17	bSyncMove	Bit	VAR			Relative value positioning start
18	bValueSet	Bit	VAR			Variable set complete
19	bCommandAborted	Bit	VAR			Relative value positioning FB CommandAborted output
20	MC_Stop_1	MC_Stop	VAR			Axis Stop FB1
21	MC_Stop_2	MC_Stop	VAR			Axis Stop FB2
22	MC_Stop_3	MC_Stop	VAR			Axis Stop FB3

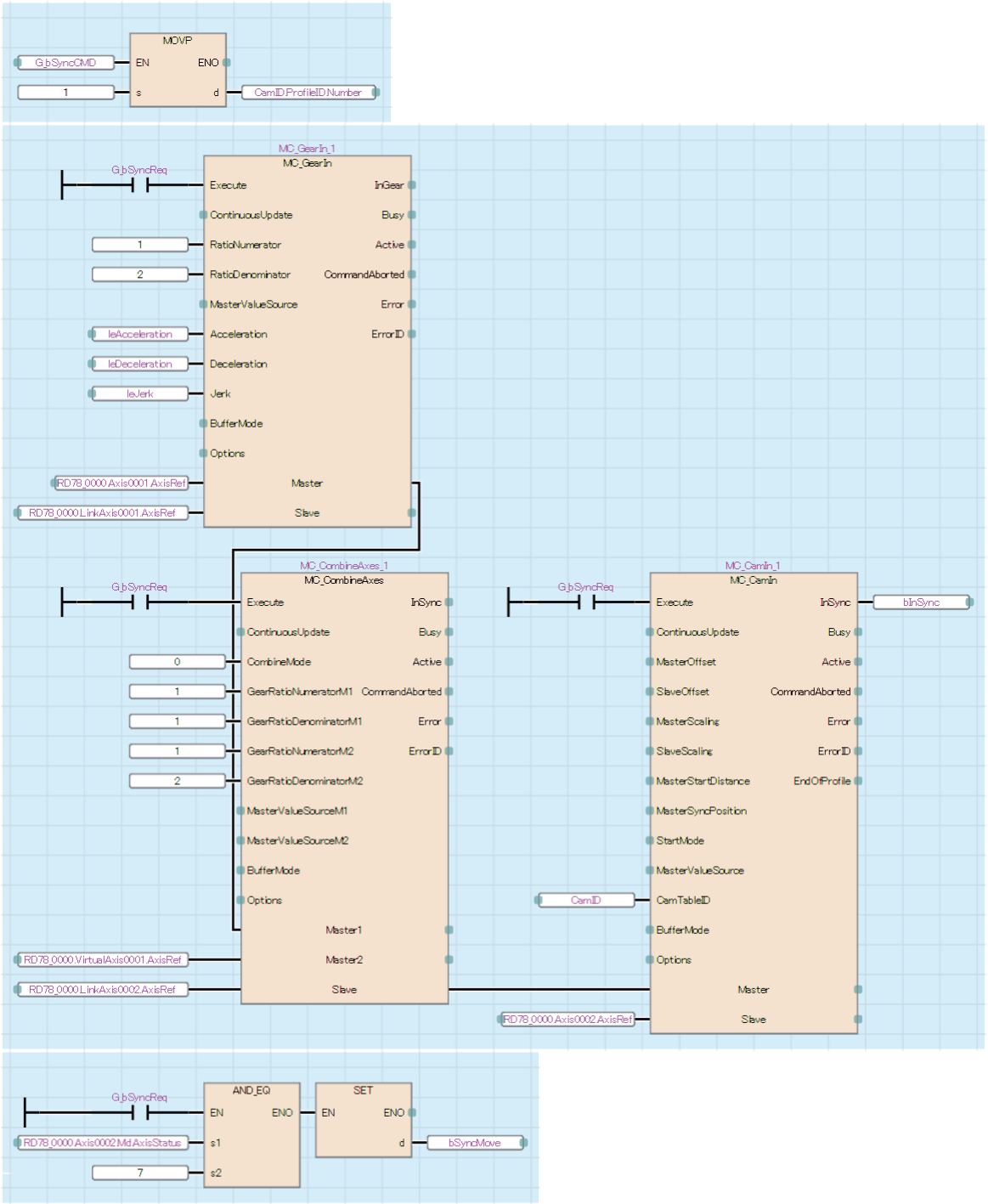
- ProgramBody
  - Data setting for synchronous control



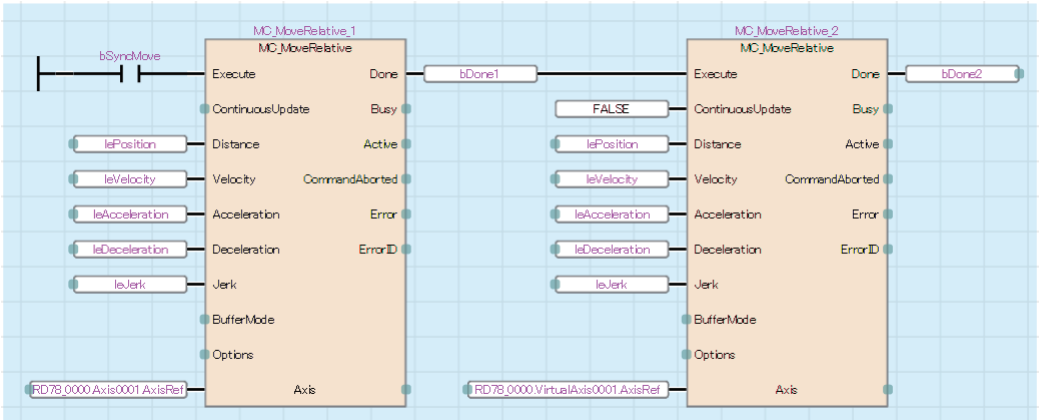
  - Synchronous control



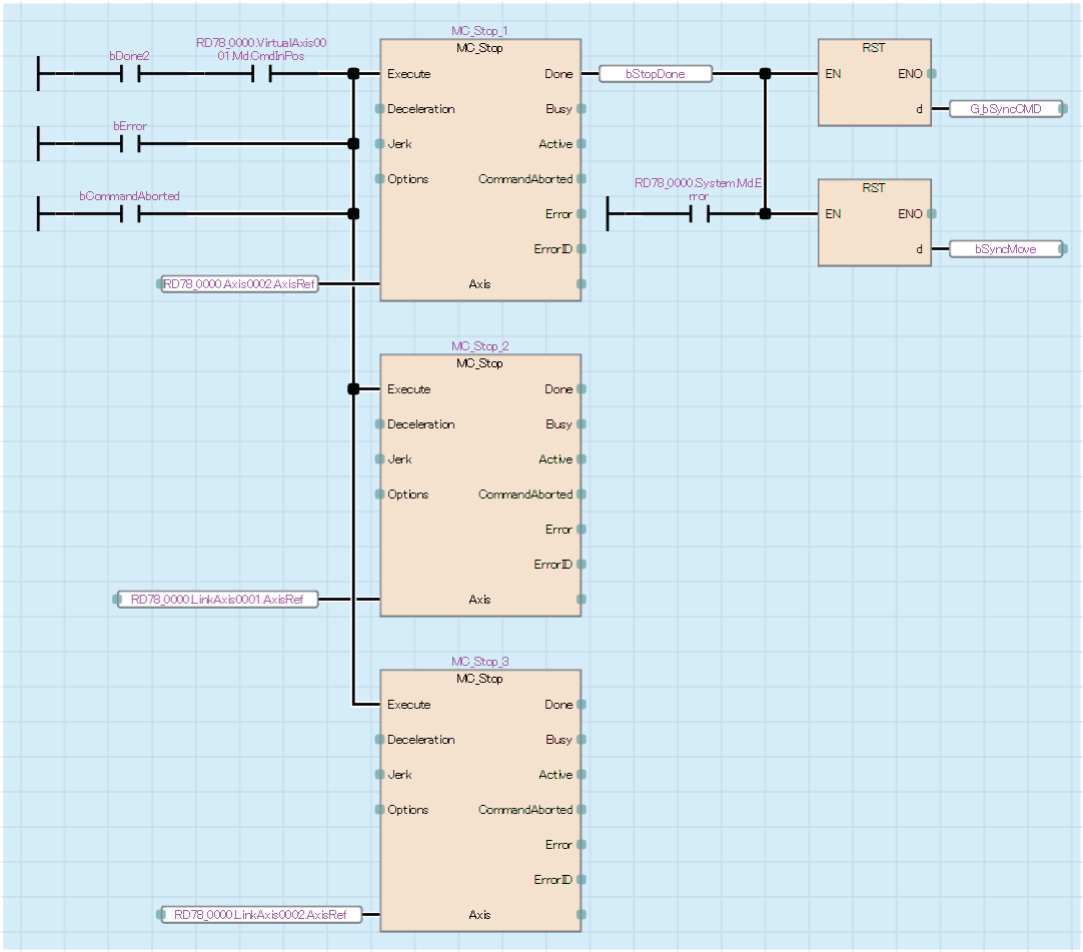
• Gear operation/Combining the motion of two master axes/Cam operation



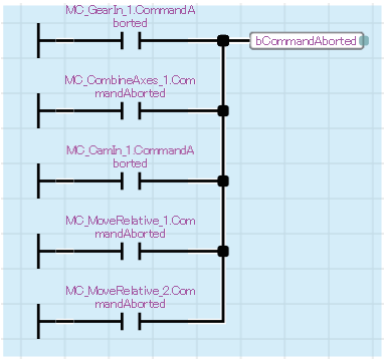
• Relative value positioning real drive axis Axis0001/Relative value positioning VirtualAxis0001



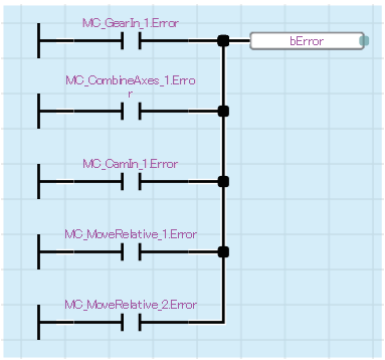
• Synchronization done



• FB CommandAborted output



• FB error output



Settings in Motion Control Setting Function

The following settings are performed in the Motion Control Setting Function.

- Axis
- Axes group
- Global label
- Structure
- Calculation profile

Axis

The settings for the axes used in the program are shown below.

Data name	Title	Axis No.	Axis type setting	Station address
Axis0001	[X_Axis]	1	Real drive axis	192.168.3.1
Axis0002	[Y_Axis]	2	Real drive axis	192.168.3.2
VirtualAxis0001	[Vir_Axis01]	301	Virtual drive axis	—
LinkAxis0001	[Lin_Axis01]	401	Virtual link axis	—
LinkAxis0002	[Lin_Axis02]	402	Virtual link axis	—

■Real drive axis

Item					Axis0001	Axis0002
Axis information	Axis No.				1	2
Axis parameter constant	Station address setting				192.168.3.1	192.168.3.2
	Axis type setting				0: Real drive axis	0: Real drive axis
Axis parameter	Driver unit conversion numerator				67108864[pulse]	67108864[pulse]
	Driver unit conversion denominator				5000[μm]	5000[μm]
	Stop signal	Signal	Target	Source type	Global label	Global label
				Source	[VAR]G_bStopSignalX	[VAR]G_bStopSignalY
	Position command unit				μm	μm

■Virtual drive axis

Item		VirtualAxis0001
Axis information	Axis No.	301
Axis parameter constant	Axis type setting	3: Virtual drive axis
Axis parameter	Position command unit	μm

■Virtual link axis

Item		LinkAxis0001	LinkAxis0002
Axis information	Axis No.	401	402
Axis parameter constant	Axis type setting	5: Virtual link axis	5: Virtual link axis
Axis parameter	Position command unit	μm	μm

Axes group

The settings for the axes group used in the program are shown below.

Item		Setting
Data name		AxesGroup001
Title		[X-Y Table]
Axes group information	Axes group No.	1
Axes group parameter	Structuring axis[1]	Axis0001
	Structuring axis[2]	Axis0002
	Position command unit	μm

A

Global label

The settings for the global labels used in the program are shown below.

Set the public label of global labels that will be used to "Enable".

■Global

	Label Name	Data Type	Class	Initial Value	Constant	Comment	Remark	Public Label	Motion Control Attribute
1	G_bStopSignalX	Bit	VAR_GLOBAL			Stop command Axis0001		Enable	WRITE (=> Motion)
2	G_bStopSignalY	Bit	VAR_GLOBAL			Stop command Axis0002		Enable	WRITE (=> Motion)

■Ax+Global

Label name	Data type	Class	Comment	Public label
Axis0001	AXIS_REAL	VAR_GLOBAL	[X_Axis]	Enable
Axis0002	AXIS_REAL	VAR_GLOBAL	[Y_Axis]	Enable
VirtualAxis0001	AXIS_VIRTUAL	VAR_GLOBAL	[Vir_Axis01]	Enable
LinkAxis0001	AXIS_VIRTUAL_LINK	VAR_GLOBAL	[Lin_Axis01]	Enable
LinkAxis0002	AXIS_VIRTUAL_LINK	VAR_GLOBAL	[Lin_Axis02]	Enable

■Gr+Global

Label name	Data type	Class	Comment	Public label
AxesGroup001	AXES_GROUP	VAR_GLOBAL	[X-Y Table]	Enable

■Sys+Global

Label name	Data type	Class	Comment	Public label
System	SYSTEM	VAR_GLOBAL		Enable

Structure

The settings for the structures used in the program are shown below.

Set the public label of structures that will be used to "Enable".

■AXIS_REAL

Label name	Data type	Comment	Public label
AxisRef	AXIS_REF	Axis information	Enable
Md	AXIS_REAL_MONI	Axis monitor data	Enable

■AXIS_REAL_MONI

Label name	Data type	Comment	Public label
AxisStatus	Word[signed]	Axis status	Enable
CmdInPos	Bit	Command in-position	Enable
Driver_ServoOn	Bit	Driver servo ON status	Enable
Homing_Request	Bit	Home position return request	Enable

■AXIS_VIRTUAL

Label name	Data type	Comment	Public label
AxisRef	AXIS_REF	Axis information	Enable
Md	AXIS_VIRTUAL_MONI	Axis monitor data	Enable

■AXIS_VIRTUAL_MONI

Label name	Data type	Comment	Public label
AxisStatus	Word[signed]	Axis status	Enable
CmdInPos	Bit	Command in-position	Enable
Homing_Request	Bit	Home position return request	Enable

■AXIS_VIRTUAL_LINK

Label name	Data type	Comment	Public label
AxisRef	AXIS_REF	Axis information	Enable
Md	AXIS_VIRTUAL_LINK_MONI	Axis monitor data	Enable

■AXES_GROUP

Label name	Data type	Comment	Public label
AxisGroupRef	AXIS_GROUP_REF	Axes group information	Enable
Md	AXIS_GROUP_MONI	Axes group monitor data	Enable

■AXES_GROUP_MONI

Label name	Data type	Comment	Public label
CmdInPos	Bit	Command in-position	Enable
GroupStatus	Bit	Axes group status	Enable

■SYSTEM

Label name	Data type	Comment	Public label
Md	SYS_MONI	System monitor data	Enable

■SYS_MONI

Label name	Data type	Comment	Public label
Error	Bit	Motion system error detection	Enable

Calculation profile

The settings for calculation profiles (cam data) used in the program are shown below.

Item		Setting value
Data name		ProfileData0001
Title		[Cam #1]
Data format		Cam data
Interpolation method specification		Section interpolation
Expand setting	Auto expand	Yes
	Profile ID	1
	Repetitive operation	Enable
	Master axis absolute coordinate	Disable (Relative coordinate)
	Slave axis absolute coordinate	Enable (Absolute coordinate)
Setting method	Resolution	256
	Length per cycle setting	200000
	Stroke amount	200000
	Cam time setting per cycle	1.000[s]

■Stroke setting

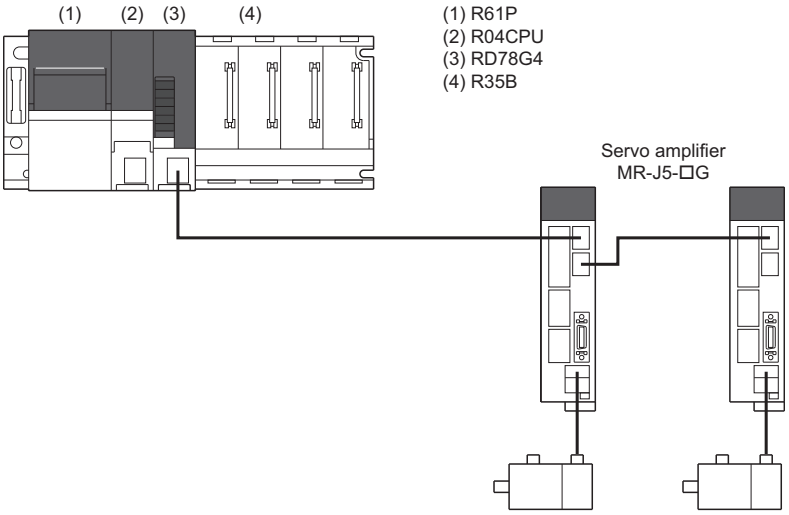
Section No.	Start point	End point	Stroke	Cam curve type
1	0	50000	100000	Single hypotenuse
2	50000	100000	200000	Single hypotenuse
3	100000	150000	50000	Single hypotenuse
4	150000	0	0	Single hypotenuse

Appendix 2 ST Sample Program

A sample program created using ST language on the Motion module side is shown below.

System configuration

The system configuration used in the sample program is shown below.



Settings in GX Works3

The following settings are performed in GX Works3.

- System parameter
- CPU parameter
- Module parameter (Network)
- Program

System parameter

The following items are set in system parameter.

■Base/power/extension cable setting

Base No.	Base	Slots	Power supply module
Main	R35B	5	R61P

■I/O assignment setting

Slot		Module name	Module status setting	Points	Start XY	CPU module operation at error detection
Base	CPU	R04CPU (host station)	—	—	3E00	—
	0(0-0)	RD78G4	No setting	32 points	0000	Critical: Stop, Moderate: Continue

CPU parameter

The following items are set in CPU parameter.

■Memory/device setting

Item	Setting
Link direct device setting	Extended mode (iQ-R series mode)

■Program setting

Execute order	Program name	Execution type		Refresh group setting
		Type	Detailed setting information	
1	MAIN	Scan	—	(Do not set)

Module parameter (Network)

The following items are set in module parameter (Network).

■Basic setting

• Network configuration settings

Model name	STA#	Station type	Motion control station	RWr setting Points	RWw setting Points	Parameter automatic setting	IP address
Host station	0	Master station	—	—	—	—	192.168.3.253
MR-J5-G	1	Remote station	Checked	24	20	Checked	192.168.3.1
MR-J5-G	2	Remote station	Checked	24	20	Checked	192.168.3.2

Device

The following devices are used in the program.

Device	Comment
X0	READY
X1	Synchronization flag
Y0	PLC READY
SM400	AlwaysON
M0	Servo OFF
M1	JOG Positive rotation command Axis0001
M2	JOG Reverse rotation command Axis0001
M3	JOG Positive rotation command Axis0002
M4	JOG Reverse rotation command Axis0002
M10	Homing start request Axis0001
M11	Homing start request Axis0002
M12	Homing start request VirtualAxis0001
M20	Single axis positioning start
M21	Single axis continuous positioning start
M22	2-axis linear interpolation control start
M23	Synchronization control start
M30	Error reset
M31	System error reset

Program

Create a program to start the ST program for the Motion module.

The programs created using ladder language are shown below.

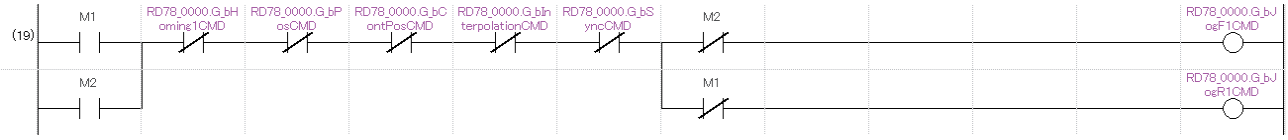
■MAIN

• PLC READY

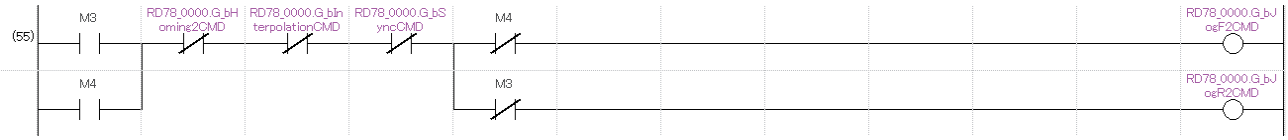
• Servo ON/OFF

• JOG operation

• Axis0001 JOG command

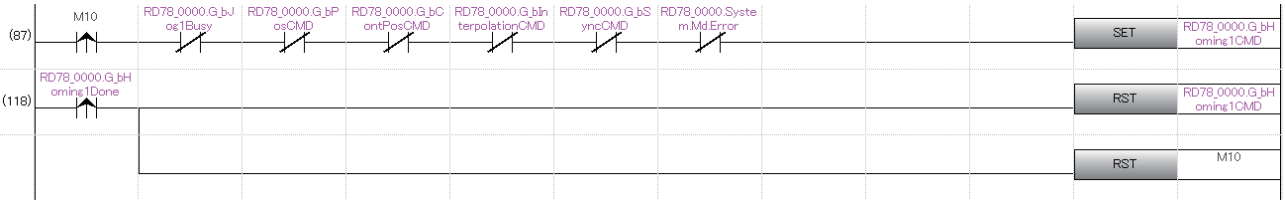


• Axis0002 JOG command

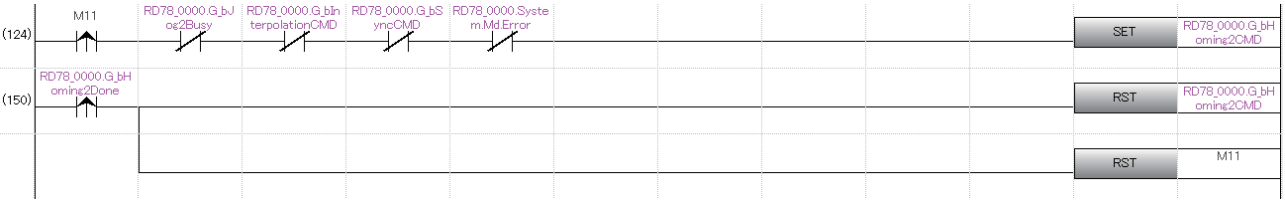


• Homing

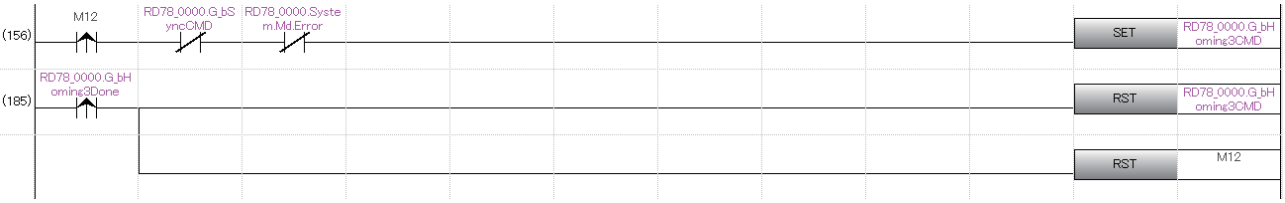
• Axis0001 Homing



• Axis0002 Homing

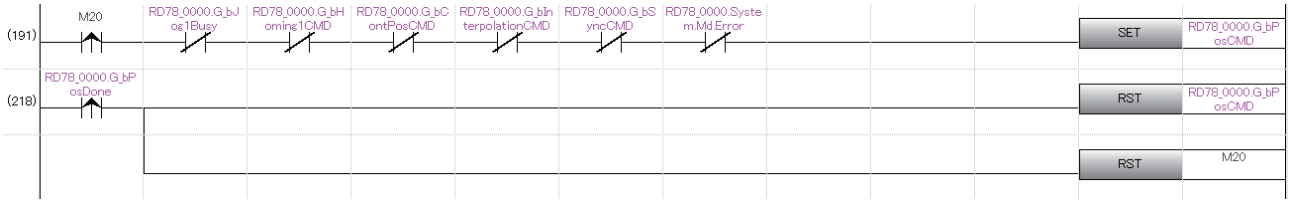


• VirtualAxis0001 Homing

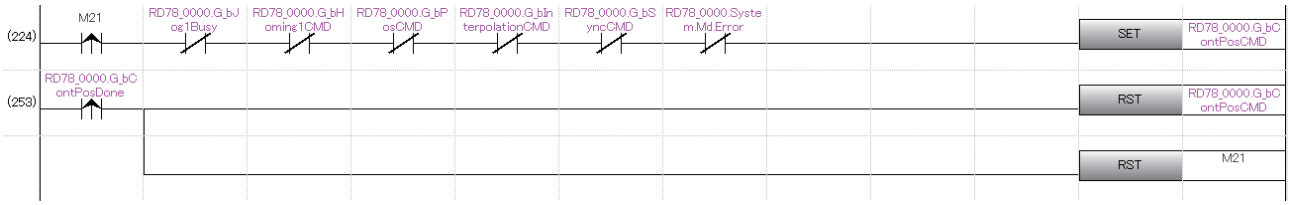


• Program start

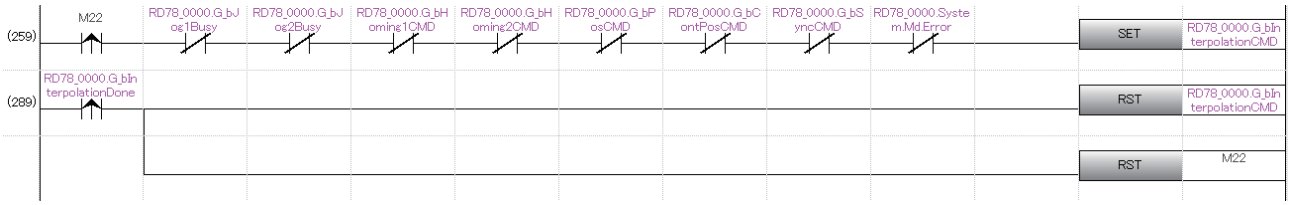
• Single axis positioning control



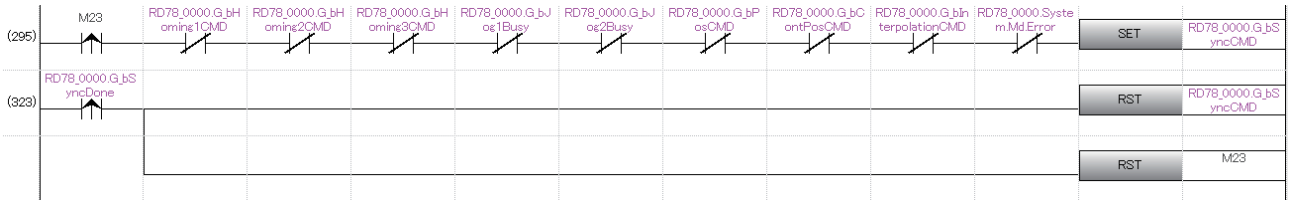
• Single axis continuous positioning control



• 2-axis linear interpolation control

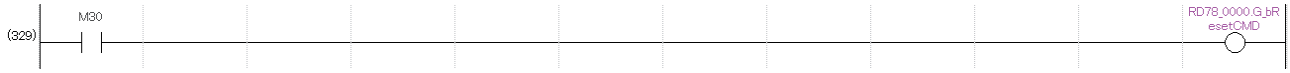


• Synchronous control

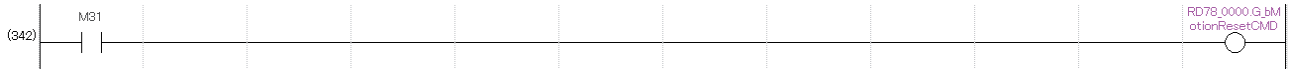


• Error reset

• Error reset



• System error reset



A

Settings in Motion Control Setting Function

The following settings are performed in the Motion Control Setting Function.

- Basic settings
- Axis
- Axes group
- Global label
- Calculation profile
- Program

Basic setting

The following items are set in basic setting.

■Program execution setting

Execution order	Program name	Execution type	Program execution condition
1	ServoON_Jog	1: Normal	—
2	Positioning	1: Normal	—
3	ContinuousPositioning	1: Normal	—
4	LinearInterpolation	1: Normal	—
5	Homing	1: Normal	—
6	ErrorReset	1: Normal	—
7	Synchronous	1: Normal	—

Axis

The settings for the axes used in the program are shown below.

Data name	Title	Axis No.	Axis type setting	Station address
Axis0001	[X_Axis]	1	Real drive axis	192.168.3.1
Axis0002	[Y_Axis]	2	Real drive axis	192.168.3.2
VirtualAxis0001	[Vir_Axis01]	301	Virtual drive axis	—
LinkAxis0001	[Lin_Axis01]	401	Virtual link axis	—
LinkAxis0002	[Lin_Axis02]	402	Virtual link axis	—

■Real drive axis

Item					Axis0001	Axis0002
Axis information	Axis No.				1	2
Axis parameter constant	Station address setting				192.168.3.1	192.168.3.2
	Axis type setting				0: Real drive axis	0: Real drive axis
Axis parameter	Driver unit conversion numerator				67108864[pulse]	67108864[pulse]
	Driver unit conversion denominator				5000[μm]	5000[μm]
	Stop signal	Signal	Target	Source type	Global label	Global label
				Source	[VAR]G_bStopSignalX	[VAR]G_bStopSignalY
	Position command unit				μm	μm

■Virtual drive axis

Item		VirtualAxis0001
Axis information	Axis No.	301
Axis parameter constant	Axis type setting	3: Virtual drive axis
Axis parameter	Position command unit	μm

■Virtual link axis

Item		LinkAxis0001	LinkAxis0002
Axis information	Axis No.	401	402
Axis parameter constant	Axis type setting	5: Virtual link axis	5: Virtual link axis
Axis parameter	Position command unit	μm	μm

Axes group

The settings for the axes group used in the program are shown below.

Item		Setting
Data name		AxesGroup001
Title		[X-Y Table]
Axes group information	Axes group No.	1
Axes group parameter	Structuring axis[1]	Axis0001
	Structuring axis[2]	Axis0002
	Position command unit	μm

Global label

The settings for the global labels used in the program are shown below.

Set the public label of global labels that will be used to "Enable".

■Global

	Label Name	Data Type	Class	Initial Value	Constant	Comment	Remark	Public Label	Motion Control Attribute
1	G_bSRVONCMD	Bit	VAR_GLOBAL			Servo ON/OFF		Enable	WRITE (=> Motion)
2	G_bHoming1CMD	Bit	VAR_GLOBAL			Homing command Axis0001		Enable	WRITE (=> Motion)
3	G_bHoming2CMD	Bit	VAR_GLOBAL			Homing command Axis0002		Enable	WRITE (=> Motion)
4	G_bHoming3CMD	Bit	VAR_GLOBAL			Homing command VirtualAxis001		Enable	WRITE (=> Motion)
5	G_bPosCMD	Bit	VAR_GLOBAL			Single axis positioning start		Enable	WRITE (=> Motion)
6	G_bContPosCMD	Bit	VAR_GLOBAL			Single axis continuous positioning start		Enable	WRITE (=> Motion)
7	G_bInterpolationCMD	Bit	VAR_GLOBAL			2-axis linear interpolation control start		Enable	WRITE (=> Motion)
8	G_bSyncCMD	Bit	VAR_GLOBAL			Synchronous control start		Enable	WRITE (=> Motion)
9	G_bResetCMD	Bit	VAR_GLOBAL			Error reset		Enable	WRITE (=> Motion)
10	G_bMotionResetCMD	Bit	VAR_GLOBAL			System error reset		Enable	WRITE (=> Motion)
11	G_bJogF1CMD	Bit	VAR_GLOBAL			JOG Positive rotation command Axis0001		Enable	WRITE (=> Motion)
12	G_bJogR1CMD	Bit	VAR_GLOBAL			JOG Reverse rotation command Axis0001		Enable	WRITE (=> Motion)
13	G_bJogF2CMD	Bit	VAR_GLOBAL			JOG Positive rotation command Axis0002		Enable	WRITE (=> Motion)
14	G_bJogR2CMD	Bit	VAR_GLOBAL			JOG Reverse rotation command Axis0002		Enable	WRITE (=> Motion)
15	G_leJogVelocity	FLOAT [Double Precision]	VAR_GLOBAL			JOG Velocity		Enable	WRITE (=> Motion)
16	G_bHoming1Done	Bit	VAR_GLOBAL			Homing complete Axis0001		Enable	READ (Motion =>)
17	G_bHoming2Done	Bit	VAR_GLOBAL			Homing complete Axis0002		Enable	READ (Motion =>)
18	G_bHoming3Done	Bit	VAR_GLOBAL			Homing complete VirtualAxis0001		Enable	READ (Motion =>)
19	G_bPosDone	Bit	VAR_GLOBAL			Single axis positioning complete		Enable	READ (Motion =>)
20	G_bContPosDone	Bit	VAR_GLOBAL			Single axis continuous positioning complete		Enable	READ (Motion =>)
21	G_bInterpolationDone	Bit	VAR_GLOBAL			2-axis linear interpolation control complete		Enable	READ (Motion =>)
22	G_bSyncDone	Bit	VAR_GLOBAL			Synchronous control complete		Enable	READ (Motion =>)
23	G_bJog1Busy	Bit	VAR_GLOBAL			JOG operation in progress Axis0001		Enable	READ (Motion =>)
24	G_bJog2Busy	Bit	VAR_GLOBAL			JOG operation in progress Axis0002		Enable	READ (Motion =>)
25	G_bStopSignalX	Bit	VAR_GLOBAL			Stop command Axis0001		Enable	READ (Motion =>)
26	G_bStopSignalY	Bit	VAR_GLOBAL			Stop command Axis0002		Enable	READ (Motion =>)

■Sys+Global

Label name	Data type	Class	Comment	Public label
System	SYSTEM	VAR_GLOBAL		Enable

Structure

The settings for the structures used in the program are shown below.

Set the public label of structures that will be used to "Enable".

■AXIS_REAL

Label name	Data type	Comment	Public label
AxisRef	AXIS_REF	Axis information	Enable

■AXIS_VIRTUAL

Label name	Data type	Comment	Public label
AxisRef	AXIS_REF	Axis information	Enable

■AXIS_VIRTUAL_LINK

Label name	Data type	Comment	Public label
AxisRef	AXIS_REF	Axis information	Enable

■AXES_GROUP

Label name	Data type	Comment	Public label
AxisGroupRef	AXIS_GROUP_REF	Axes group information	Enable

■SYSTEM

Label name	Data type	Comment	Public label
Md	SYS_MONI	System monitor data	Enable

■SYS_MONI

Label name	Data type	Comment	Public label
Error	Bit	Motion system error detection	Enable

Calculation profile

The settings for calculation profiles (cam data) used in the program are shown below.

Item		Setting value
Data name		ProfileData0001
Title		[Cam #1]
Data format		Cam data
Interpolation method specification		Section interpolation
Expand setting	Auto expand	Yes
	Profile ID	1
	Repetitive operation	Enable
	Master axis absolute coordinate	Disable (Relative coordinate)
	Slave axis absolute coordinate	Enable (Absolute coordinate)
Setting method	Resolution	256
	Length per cycle setting	200000
	Stroke amount	200000
	Cam time setting per cycle	1.000[s]

■Stroke setting

Section No.	Start point	End point	Stroke	Cam curve type
1	0	50000	100000	Single hypotenuse
2	50000	100000	200000	Single hypotenuse
3	100000	150000	50000	Single hypotenuse
4	150000	0	0	Single hypotenuse

Program

The programs created using ST language are shown below.

Program name	Processing details	Reference
ServoON_Jog	• Performs servo ON for all axes. • Performs JOG operation for Axis 1 (Axis0001) and Axis 2 (Axis0002).	☞ Page 503 ServoON_Jog
Positioning	• Performs single axis positioning control using the relative value of Axis 1 (Axis0001).	☞ Page 504 Positioning
ContinuousPositioning	• Performs single axis continuous positioning control using the relative values of Axis 1 (Axis0001) and Axis 2 (Axis0002).	☞ Page 505 ContinuousPositioning
LinearInterpolation	• Performs 2-axis linear interpolation control for the axes group (AxesGroup001) of Axis 1 (Axis0001) and Axis 2 (Axis0002).	☞ Page 507 LinearInterpolation
Homing	• Performs homing for Axis 1 (Axis0001), Axis 2 (Axis0002), and Axis 301 (VirtualAxis0001).	☞ Page 509 Homing
ErrorReset	• Performs error reset of the Motion system for Axis 1 (Axis0001), Axis 2 (Axis0002), Axis 301 (VirtualAxis0001), Axis 401 (LinkAxis0001), Axis 402 (LinkAxis0002), and Axes group 1 (AxesGroup001).	☞ Page 510 ErrorReset
Synchronous	• Performs synchronous control.	☞ Page 511 Synchronous

■ServoON_Jog

• Local labels

	Label Name	Data Type	Class	Initial Value	Constant	Comment
1	MCv_AllPower_1	MCv_AllPower	VAR			All axes servo ON FB
2	MCv_Jog_1	MCv_Jog	VAR			JOG operation FB1 Axis0001
3	MCv_Jog_2	MCv_Jog	VAR			JOG operation FB2 Axis0002
4	leJogAcceleration	FLOAT [Double Precision]	VAR			JOG acceleration
5	leJogDeceleration	FLOAT [Double Precision]	VAR			JOG deceleration
6	leJogJerk	FLOAT [Double Precision]	VAR			JOG jerk

• ProgramBody

```
//-----All axes servo ON-----
MCv_AllPower_1(
   Enable:= TRUE ,
   ServoON:= G_bSRVONCMD
);

//-----Data setting for JOG operation-----
IF MCv_AllPower_1.Busy THEN
   leJogAcceleration:= 50000.0;
   leJogDeceleration:= 50000.0;
   leJogJerk:= 0.0;
END_IF;

//-----Axis0001 JOG operation-----
MCv_Jog_1(
   Axis:= Axis0001.AxisRef ,
   JogForward:= G_bJogF1CMD ,
   JogBackward:= G_bJogR1CMD ,
   Velocity:= G_leJogVelocity ,
   Acceleration:= leJogAcceleration ,
   Deceleration:= leJogDeceleration ,
   Jerk:= leJogJerk ,
   BUSY=> G_bJog1Busy
);

//-----Axis0002 JOG operation-----
MCv_Jog_2(
```

```
    Axis:= Axis0002.AxisRef ,
    JogForward:= G_bJogF2CMD ,
    JogBackward:= G_bJogR2CMD ,
    Velocity:= G_leJogVelocity ,
    Acceleration:= leJogAcceleration ,
    Deceleration:= leJogDeceleration ,
    Jerk:= leJogJerk  ,
    BUSY=> G_bJog2Busy
);
```

■Positioning

• Local labels

	Label Name	Data Type	Class	Initial Value	Constant	Comment
1	MC_MoveRelative_1	MC_MoveRelative	VAR			Relative value positioning FB
2	leDistance	FLOAT [Double Precision]	VAR			Distance
3	leVelocity	FLOAT [Double Precision]	VAR			Velocity
4	leAcceleration	FLOAT [Double Precision]	VAR			Acceleration
5	leDeceleration	FLOAT [Double Precision]	VAR			Deceleration
6	leJerk	FLOAT [Double Precision]	VAR			Jerk
7	bDone	Bit	VAR			Relative value positioning FB Done output
8	bBusy	Bit	VAR			Relative value positioning FB Busy output
9	bError	Bit	VAR			Relative value positioning FB Error output
10	bExecute_P	Bit	VAR			Start
11	bCommandAborted	Bit	VAR			Relative value positioning FB CommandAborted output

• ProgramBody

```
//-----Single axis positioning control (data setting, execution, and reset)-----
IF G_bPosCMD THEN;
    leDistance:= 200000.0;
    leVelocity:= 50000.0;
    leAcceleration:= 100000.0;
    leDeceleration:= 100000.0;
    leJerk:= 0.0;
    bExecute_P:= TRUE;
ELSE
    bExecute_P:= FALSE;
END_IF;

//-----Single axis positioning-----
//Relative value positioning
MC_MoveRelative_1(
    Axis:= Axis0001.AxisRef ,
    Execute:= bExecute_P ,
    Distance:= leDistance ,
    Velocity:= leVelocity ,
    Acceleration:= leAcceleration ,
    Deceleration:= leDeceleration ,
    Jerk:= leJerk ,
    Done=> bDone ,
    Busy=> bBusy ,
    CommandAborted=> bCommandAborted ,
    Error=> bError
);

//Transmits positioning complete signal to the PLC CPU
G_bPosDone:= ( bDone & Axis0001.Md.CmdInPos ) OR bCommandAborted OR bError;
```

■ContinuousPositioning

• Local labels

	Label Name	Data Type	Class	Initial Value	Constant	Comment
1	MC_MoveRelative_1	MC_MoveRelative	VAR			Relative value positioning FB1
2	MC_MoveRelative_2	MC_MoveRelative	VAR			Relative value positioning FB2
3	leDistance1	FLOAT [Double Precision]	VAR			Distance1
4	leDistance2	FLOAT [Double Precision]	VAR			Distance2
5	leVelocity1	FLOAT [Double Precision]	VAR			Velocity1
6	leVelocity2	FLOAT [Double Precision]	VAR			Velocity2
7	leAcceleration1	FLOAT [Double Precision]	VAR			Acceleration1
8	leDeceleration1	FLOAT [Double Precision]	VAR			Deceleration1
9	leAcceleration2	FLOAT [Double Precision]	VAR			Acceleration2
10	leDeceleration2	FLOAT [Double Precision]	VAR			Deceleration2
11	leJerk	FLOAT [Double Precision]	VAR			Jerk
12	bBusy1	Bit	VAR			Relative value positioning FB1 Busy output
13	bActive1	Bit	VAR			Relative value positioning FB1 Active output
14	bDone2	Bit	VAR			Relative value positioning FB2 Done output
15	bBusy2	Bit	VAR			Relative value positioning FB2 Busy output
16	TON_1	TON	VAR			On-delay timer FB
17	bDwell_in	Bit	VAR			Timer input
18	bDwell_out	Bit	VAR			Timer output
19	bExecute_CP	Bit	VAR			Start
20	bCommandAborted	Bit	VAR			Relative value positioning FB CommandAborted output
21	bError	Bit	VAR			Relative value positioning FB Error output

• ProgramBody

```
//-----Single axis continuous positioning control (data setting, execution, and reset)-----
IF G_bContPosCMD THEN;
   leDistance1:= 200000.0;
   leDistance2:= 200000.0;
   leVelocity1:= 50000.0;
   leVelocity2:= 25000.0;
   leAcceleration1:= 100000.0;
   leDeceleration1:= 100000.0;
   leAcceleration2:= 50000.0;
   leDeceleration2:= 50000.0;
   leJerk:= 0.0;
   bExecute_CP:= TRUE;
ELSE
   bExecute_CP:= FALSE;
END_IF;

//-----Single axis continuous positioning + Buffer mode + Dwell-----
//Relative value positioning 1
MC_MoveRelative_1(
   Axis:= Axis0001.AxisRef ,
   Execute:= bExecute_CP ,
   Distance:= leDistance1 ,
   Velocity:= leVelocity1 ,
   Acceleration:= leAcceleration1 ,
   Deceleration:= leDeceleration1 ,
   Jerk:= leJerk ,
   Busy=> bBusy1 ,
   Active=> bActive1
);

//Relative value positioning 2
MC_MoveRelative_2(
   Axis:= Axis0001.AxisRef ,
   Execute:= bActive1,
```

```
    Distance:= leDistance2 ,
    Velocity:= leVelocity2 ,
    Acceleration:= leAcceleration2 ,
    Deceleration:= leDeceleration2 ,
    BufferMode:= MC_BUFFER_MODE__mcBlendingPrevious ,
    Done=> bDone2 ,
    Busy=> bBusy2
);

//FB CommandAborted output
bCommandAborted:= MC_MoveRelative_1.CommandAborted OR MC_MoveRelative_2.CommandAborted;

//FB Error output
bError:= MC_MoveRelative_1.Error OR MC_MoveRelative_2.Error;

//Sets timer input with relative value positioning 2 complete signal
SET( bDone2, bDwell_in );

//Dwell
TON_1(
    IN:= bDwell_in ,
    PT:= T#100ms ,
    Q=> bDwell_out
);

//Transmits continuous positioning complete signal to the PLC CPU
G_bContPosDone:= bDwell_out OR bCommandAborted OR bError ;

//Resets timer input when positioning is completed
RST( bDwell_out, bDwell_in );
```

■LinearInterpolation

• Local labels

	Label Name	Data Type	Class	Initial Value	Constant	Comment
1	MCv_MoveLinearInterpolateRelative_1	MCv_MoveLinearInterpolateRelative	VAR			Relative value linear interpolation control FB1
2	MCv_MoveLinearInterpolateRelative_2	MCv_MoveLinearInterpolateRelative	VAR			Relative value linear interpolation control FB2
3	MC_GroupEnable_1	MC_GroupEnable	VAR			Axes group enable FB
4	MC_GroupDisable_1	MC_GroupDisable	VAR			Axes group disable FB
5	wAxes	Word [Signed](0..15)	VAR			Interpolation axis
6	lePosition1	FLOAT [Double Precision](0..15)	VAR			Position data 1
7	lePosition2	FLOAT [Double Precision](0..15)	VAR			Position data 2
8	leVelocity	FLOAT [Double Precision]	VAR			Velocity
9	leAcceleration	FLOAT [Double Precision]	VAR			Acceleration
10	leDeceleration	FLOAT [Double Precision]	VAR			Deceleration
11	leJerk	FLOAT [Double Precision]	VAR			Jerk
12	bGroupEnableDone	Bit	VAR			Axes group enable done
13	bGroupDisableDone	Bit	VAR			Axes group disable done
14	bBusy1	Bit	VAR			Relative value linear interpolation control FB1 Busy output
15	bActive1	Bit	VAR			Relative value linear interpolation control FB1 Active output
16	bDone2	Bit	VAR			Relative value linear interpolation control FB2 Done output
17	bBusy2	Bit	VAR			Relative value linear interpolation control FB2 Busy output
18	TON_1	TON	VAR			On-delay timer FB
19	bDwell_out	Bit	VAR			Timer output
20	bDwell_in	Bit	VAR			Timer input
21	bExcute_LP	Bit	VAR			Start
22	bCommandAborted	Bit	VAR			Relative value linear interpolation control FB CommandAborted output
23	bError	Bit	VAR			Relative value linear interpolation control FB Error output

• ProgramBody

```
//-----2-axis linear interpolation control (data setting, execution, and reset)-----
IF G_bInterpolationCMD THEN;
    wAxes[0]:= 1;
    wAxes[1]:= 2;
    lePosition1[0]:= 100000.0;
    lePosition1[1]:= 200000.0;
    lePosition2[0]:= -100000.0;
    lePosition2[1]:= -200000.0;
    leVelocity:= 50000.0;
    leAcceleration:= 50000.0;
    leDeceleration:= 50000.0;
    leJerk:= 0.0;
    bExcute_LP:= TRUE;
ELSE
    bExcute_LP:= FALSE;
END_IF;

//-----Linear interpolation control + Buffer mode-----
//Axes group enable
MC_GroupEnable_1(
    AxesGroup:= AxesGroup001.AxesGroupRef ,
    Execute:= bExcute_LP ,
    Done=> bGroupEnableDone
);

//Linear interpolation control 1
MCv_MoveLinearInterpolateRelative_1(
    AxesGroup:= AxesGroup001.AxesGroupRef ,
    Execute:= bGroupEnableDone ,
    LinearAxes:= wAxes ,
    Distance:= lePosition1 ,
    Velocity:= leVelocity ,
    Acceleration:= leAcceleration ,
```

```
    Deceleration:= leDeceleration ,
    Jerk:= leJerk ,
    VelocityMode:= MC_INTERPOLATE_SPEED_MODE__VectorSpeed ,
    Busy=> bBusy1 ,
    Active=> bActive1
);

//Linear interpolation control 2
MCv_MoveLinearInterpolateRelative_2(
    AxesGroup:= AxesGroup001.AxesGroupRef ,
    Execute:= bActive1 ,
    LinearAxes:= wAxes ,
    Distance:= lePosition2 ,
    BufferMode:= MC_BUFFER_MODE__mcBuffered ,
    Done=> bDone2 ,
    Busy=> bBusy2
);

//FB CommandAborted output
bCommandAborted:= MCv_MoveLinearInterpolateRelative_1.CommandAborted OR
MCv_MoveLinearInterpolateRelative_2.CommandAborted;

//FB error output
bError:= MCv_MoveLinearInterpolateRelative_1.Error OR MCv_MoveLinearInterpolateRelative_2.Error;

//Sets timer input with linear interpolation control 2 complete signal or FB CommandAborted output
SET( bDone2 OR bCommandAborted, bDwell_in );

//Dwell
TON_1(
     IN:= bDwell_in ,
    PT:= T#100ms ,
    Q=> bDwell_out
);

//Axes group disable
MC_GroupDisable_1(
    AxesGroup:= AxesGroup001.AxesGroupRef,
    Execute:= bDwell_out OR bError,
    Done=> bGroupDisableDone
);

//Transmits linear interpolation control complete signal to the PLC CPU
G_bInterpolationDone:= bDwell_out OR bError;

//Resets timer input when positioning is completed
RST( bDwell_out, bDwell_in );
```

■Homing

• Local labels

	Label Name	Data Type	Class	Initial Value	Constant	Comment
1	MC_Home_1	MC_Home	VAR			Homing FB1 Axis0001
2	MC_Home_2	MC_Home	VAR			Homing FB2 Axis0002
3	MC_Home_3	MC_Home	VAR			Homing FB3 VirtualAxis0001
4	bHoming1Done	Bit	VAR			Homing FB1 Done output
5	bHoming1Error	Bit	VAR			Homing FB1 Error output
6	bHoming2Done	Bit	VAR			Homing FB2 Done output
7	bHoming2Error	Bit	VAR			Homing FB2 Error output
8	bHoming3Done	Bit	VAR			Homing FB3 Done output
9	bHoming3Error	Bit	VAR			Homing FB3 Error output

• ProgramBody

```
//-----Homing Axis0001-----
//Homing
MC_Home_1(
   Axis:= Axis0001.AxisRef ,
   Execute:= G_bHoming1CMD ,
   Position:= 0.0 ,
   Done=> bHoming1Done,
   Error=> bHoming1Error
);

//Transmits homing complete signal to the PLC CPU
G_bHoming1Done:= bHoming1Done OR bHoming1Error;

//-----Homing Axis0002-----
//Homing
MC_Home_2(
   Axis:= Axis0002.AxisRef ,
   Execute:= G_bHoming2CMD ,
   Position:= 0.0 ,
   Done=> bHoming2Done,
   Error=> bHoming2Error
);

//Transmits homing complete signal to the PLC CPU
G_bHoming2Done:= bHoming2Done OR bHoming2Error;

//-----Homing VirtualAxis0001-----
//Homing
MC_Home_3(
   Axis:= VirtualAxis0001.AxisRef ,
   Execute:= G_bHoming3CMD ,
   Position:= 0.0 ,
   Done=> bHoming3Done,
   Error=> bHoming3Error
);

//Transmits homing complete signal to the PLC CPU
G_bHoming3Done:= bHoming3Done OR bHoming3Error;
```

■ErrorReset

• Local labels

	Label Name	Data Type	Class	Initial Value	Constant	Comment
1	MC_Reset_1	MC_Reset	VAR			Axis error reset FB1 Axis0001
2	MC_Reset_2	MC_Reset	VAR			Axis error reset FB2 Axis0002
3	MC_Reset_3	MC_Reset	VAR			Axis error reset FB3 VirtualAxis0001
4	MC_Reset_4	MC_Reset	VAR			Axis error reset FB4 LinkAxis0001
5	MC_Reset_5	MC_Reset	VAR			Axis error reset FB5 LinkAxis0002
6	MC_GroupReset_1	MC_GroupReset	VAR			Axes group error reset FB
7	MCv_MotionErrorReset_1	MCv_MotionErrorReset	VAR			System error reset FB

• ProgramBody

```
// Error reset
MC_Reset_1(
   Axis:= Axis0001.AxisRef ,
   Execute:= G_bResetCMD
);

MC_Reset_2(
   Axis:= Axis0002.AxisRef ,
   Execute:= G_bResetCMD
);

MC_Reset_3(
   Axis:= VirtualAxis0001.AxisRef ,
   Execute:= G_bResetCMD
);

MC_Reset_4(
   Axis:= LinkAxis0001.AxisRef ,
   Execute:= G_bResetCMD
);

MC_Reset_5(
   Axis:= LinkAxis0002.AxisRef ,
   Execute:= G_bResetCMD
);

// Axes group error reset
MC_GroupReset_1(
   AxesGroup:= AxesGroup001.AxesGroupRef ,
   Execute:= G_bResetCMD
);

// Controller error reset
MCv_MotionErrorReset_1(
   Execute:= G_bMotionResetCMD
);
```

■Synchronous

• Local labels

	Label Name	Data Type	Class	Initial Value	Constant	Comment
1	MC_GearIn_1	MC_GearIn	VAR			Gear operation FB
2	MC_CombineAxes_1	MC_CombineAxes	VAR			FB combining the motion of two master axes
3	MC_CamIn_1	MC_CamIn	VAR			Cam operation FB
4	bGearInBusy	Bit	VAR			Gear operation FB Busy output
5	bCombineAxesBusy	Bit	VAR			FB combining the motion of two master axes Busy output
6	bCamInBusy	Bit	VAR			Cam operation FB Busy output
7	MC_MoveRelative_1	MC_MoveRelative	VAR			Relative value positioning FB
8	MC_MoveRelative_2	MC_MoveRelative	VAR			Relative value positioning FB
9	MC_Stop_1	MC_Stop	VAR			Axis Stop FB1
10	MC_Stop_2	MC_Stop	VAR			Axis Stop FB2
11	MC_Stop_3	MC_Stop	VAR			Axis Stop FB3
12	leAcceleration	FLOAT [Double Precision]	VAR			Acceleration
13	leDeceleration	FLOAT [Double Precision]	VAR			Deceleration
14	leJerk	FLOAT [Double Precision]	VAR			Jerk
15	lePosition1	FLOAT [Double Precision]	VAR			Distance
16	leVelocity	FLOAT [Double Precision]	VAR			Velocity
17	bInSync	Bit	VAR			Cam operation FB inSync output
18	bDone1	Bit	VAR			Relative value positioning FB1 Done output
19	bBusy1	Bit	VAR			Relative value positioning FB1 Busy output
20	bDone2	Bit	VAR			Relative value positioning FB2 Done output
21	bBusy2	Bit	VAR			Relative value positioning FB2 Busy output
22	bStopReq1	Bit	VAR			Axis Stop request 1
23	bStopDone1	Bit	VAR			Axis Stop complete 1
24	bStopDone2	Bit	VAR			Axis Stop complete 2
25	bStopDone3	Bit	VAR			Axis Stop complete 3
26	bSyncMoveCMD	Bit	VAR			Relative value positioning start
27	bExecute_S	Bit	VAR			Start
28	bCommandAborted	Bit	VAR			Relative value positioning FB CommandAborted output
29	bError	Bit	VAR			Relative value positioning FB Error output

• ProgramBody

```
//-----Synchronous control (data setting, execution, and reset)-----
IF G_bSyncCMD THEN;
   lePosition1:= 400000.0;
   leVelocity:= 50000.0;
   leAcceleration:= 100000.0;
   leDeceleration:= 100000.0;
   leJerk:= 0.0;
   bExecute_S:= TRUE;
ELSE
   bExecute_S:= FALSE;
END_IF;

//-----Synchronous control-----
//Gear operation
MC_GearIn_1(
   Master:= Axis0001.AxisRef ,
   Slave:= LinkAxis0001.AxisRef ,
   Execute:= bExecute_S ,
   RatioNumerator:= 1 ,
   RatioDenominator:= 2 ,
   MasterValueSource:= MC_SOURCE__mcSetValue ,
   Acceleration:= leAcceleration ,
   Deceleration:= leDeceleration ,
   Jerk:= leJerk ,
   Busy=> bGearInBusy
);

//Combining the motion of two master axes
MC_CombineAxes_1(
```

```
    Master1:= LinkAxis0001.AxisRef ,
    Master2:= VirtualAxis0001.AxisRef ,
    Slave:= LinkAxis0002.AxisRef ,
    Execute:= bExecute_S ,
    CombineMode:= MC_COMBINE_MODE__mcAddAxes ,
    GearRatioNumeratorM1:= 1 ,
    GearRatioDenominatorM1:= 1 ,
    GearRatioNumeratorM2:= 1 ,
    GearRatioDenominatorM2:= 2 ,
    MasterValueSourceM1:= MC_SOURCE__mcSetValue ,
    MasterValueSourceM2:= MC_SOURCE__mcSetValue ,
    Busy=> bCombineAxesBusy
);

//Cam operation
MC_CamIn_1.CamTableID.ProfileID:= ProfileData0001.ProfileData.ID;

MC_CamIn_1(
    Master:= LinkAxis0002.AxisRef ,
    Slave:= Axis0002.AxisRef ,
    Execute:= bExecute_S ,
    MasterOffset:= 0.0 ,
    SlaveOffset:= 0.0 ,
    MasterScaling:= 1.0 ,
    SlaveScaling:= 1.0 ,
    MasterStartDistance:= 0.0 ,
    MasterSyncPosition:= 0.0 ,
    StartMode:= MC_START_MODE__mcImmediate ,
    MasterValueSource:= MC_SOURCE__mcSetValue ,
    InSync=> bInSync ,
    Busy=> bCamInBusy
);

//Input axis start
bSyncMoveCMD:= G_bSyncCMD & ( Axis0002.Md.AxisStatus = MC_AXIS_STATUS__SynchronizedMotion );

//Relative value positioning real drive axis Axis0001
MC_MoveRelative_1(
    Axis:= Axis0001.AxisRef ,
    Execute:= bSyncMoveCMD ,
    Distance:= lePosition1 ,
    Velocity:= leVelocity ,
    Acceleration:= leAcceleration ,
    Deceleration:= leDeceleration ,
    Jerk:= leJerk ,
    Done=> bDone1 ,
    Busy=> bBusy1
);

//Relative value positioning virtual drive axis VirtualAxis0001
MC_MoveRelative_2(
    Axis:= VirtualAxis0001.AxisRef ,
```

```
    Execute:= bDone1 & Axis0001.Md.CmdInPos ,
    Distance:= lePosition1 ,
    Velocity:= leVelocity ,
    Acceleration:= leAcceleration ,
    Deceleration:= leDeceleration ,
    Jerk:= leJerk ,
    Done=> bDone2 ,
    Busy=> bBusy2
);

//FB CommandAborted output
bCommandAborted:= MC_GearIn_1.CommandAborted OR MC_CombineAxes_1.CommandAborted OR
MC_CamIn_1.CommandAborted OR MC_MoveRelative_1.CommandAborted OR MC_MoveRelative_2.CommandAborted;

//FB error output
bError:= MC_GearIn_1.Error OR MC_CombineAxes_1.Error OR MC_CamIn_1.Error OR MC_MoveRelative_1.Error OR
MC_MoveRelative_2.Error;

//Turns axis stop signal ON
bStopReq1:= ( bDone2 & VirtualAxis0001.Md.CmdInPos ) OR bError OR bCommandAborted;

//Synchronization done
MC_Stop_1(
    Axis:= Axis0002.AxisRef ,
    Execute:= bStopReq1 ,
    Done=> bStopDone1
);

MC_Stop_2(
    Axis:= LinkAxis0001.AxisRef ,
    Execute:= bStopReq1 ,
    Done=> bStopDone2
);

MC_Stop_3(
    Axis:= LinkAxis0002.AxisRef ,
    Execute:= bStopReq1 ,
    Done=> bStopDone3
);

//Resets the start request signal when synchronization is completed
RST( bStopDone1 OR System.Md.Error, bSyncMoveCMD );
RST( bStopDone1 OR System.Md.Error, bStopReq1 );

//Transmits synchronous control complete signal to the PLC CPU
G_bSyncDone:= ( bStopDone2 & VirtualAxis0001.Md.CmdInPos ) OR bCommandAborted OR bError;
```

REVISIONS

*The manual number is given on the bottom left of the back cover.

Revision date	*Manual number	Description
October 2020	IB(NA)-0300533ENG-A	First edition
August 2021	IB(NA)-0300533ENG-B	■Added or modified parts TERMS, GENERIC TERMS AND ABBREVIATIONS, Section 1.1, 2.1, 3.1, 3.2, 3.3, 4.1, 4.2, WARRANTY
January 2022	IB(NA)-0300533ENG-C	■Added or modified parts TERMS, Section 1.1, 2.1, 2.3, 3.1, 3.2, 3.3, 4.1, 4.2

Japanese manual number: IB-0300532-D

This manual confers no industrial property rights of any other kind, nor does it confer any patent licenses. Mitsubishi Electric Corporation cannot be held responsible for any problems involving industrial property rights which may occur as a result of using the contents noted in this manual.

© 2020 MITSUBISHI ELECTRIC CORPORATION

WARRANTY

Warranty

1. Warranty period and coverage

We will repair any failure or defect hereinafter referred to as "failure" in our FA equipment hereinafter referred to as the "Product" arisen during warranty period at no charge due to causes for which we are responsible through the distributor from which you purchased the Product or our service provider. However, we will charge the actual cost of dispatching our engineer for an on-site repair work on request by customer in Japan or overseas countries. We are not responsible for any on-site readjustment and/or trial run that may be required after a defective unit are repaired or replaced.

[Term]

For terms of warranty, please contact your original place of purchase.

[Limitations]

(1) You are requested to conduct an initial failure diagnosis by yourself, as a general rule.
It can also be carried out by us or our service company upon your request and the actual cost will be charged. However, it will not be charged if we are responsible for the cause of the failure.

(2) This limited warranty applies only when the condition, method, environment, etc. of use are in compliance with the terms and conditions and instructions that are set forth in the instruction manual and user manual for the Product and the caution label affixed to the Product.

(3) Even during the term of warranty, the repair cost will be charged on you in the following cases;

1. a failure caused by your improper storing or handling, carelessness or negligence, etc., and a failure caused by your hardware or software problem
2. a failure caused by any alteration, etc. to the Product made on your side without our approval
3. a failure which may be regarded as avoidable, if your equipment in which the Product is incorporated is equipped with a safety device required by applicable laws and has any function or structure considered to be indispensable according to a common sense in the industry
4. a failure which may be regarded as avoidable if consumable parts designated in the instruction manual, etc. are duly maintained and replaced
5. any replacement of consumable parts (battery, fan, smoothing capacitor, etc.)
6. a failure caused by external factors such as inevitable accidents, including without limitation fire and abnormal fluctuation of voltage, and acts of God, including without limitation earthquake, lightning and natural disasters
7. a failure generated by an unforeseeable cause with a scientific technology that was not available at the time of the shipment of the Product from our company
8. any other failures which we are not responsible for or which you acknowledge we are not responsible for

2. Term of warranty after the stop of production

(1) We may accept the repair at charge for another seven (7) years after the production of the product is discontinued. The announcement of the stop of production for each model can be seen in our Sales and Service, etc.

(2) Please note that the Product (including its spare parts) cannot be ordered after its stop of production.

3. Service in overseas countries

Our regional FA Center in overseas countries will accept the repair work of the Product. However, the terms and conditions of the repair work may differ depending on each FA Center. Please ask your local FA center for details.

4. Exclusion of loss in opportunity and secondary loss from warranty liability

Regardless of the gratis warranty term, Mitsubishi shall not be liable for compensation to:

(1) Damages caused by any cause found not to be the responsibility of Mitsubishi.

(2) Loss in opportunity, lost profits incurred to the user by Failures of Mitsubishi products.

(3) Special damages and secondary damages whether foreseeable or not, compensation for accidents, and compensation for damages to products other than Mitsubishi products.

(4) Replacement by the user, maintenance of on-site equipment, start-up test run and other tasks.

5. Change of Product specifications

Specifications listed in our catalogs, manuals or technical documents may be changed without notice.

6. Application and use of the Product

(1) For the use of our Motion module, its applications should be those that may not result in a serious damage even if any failure or malfunction occurs in the Motion module, and a backup or fail-safe function should operate on an external system to the Motion module when any failure or malfunction occurs.

(2) Our Motion module is designed and manufactured as a general purpose product for use at general industries.
Therefore, applications substantially influential on the public interest for such as atomic power plants and other power plants of electric power companies, and also which require a special quality assurance system, including applications for railway companies and government or public offices are not recommended, and we assume no responsibility for any failure caused by these applications when used
In addition, applications which may be substantially influential to human lives or properties for such as airlines, medical treatments, railway service, incineration and fuel systems, man-operated material handling equipment, entertainment machines, safety machines, etc. are not recommended, and we assume no responsibility for any failure caused by these applications when used.
We will review the acceptability of the abovementioned applications, if you agree not to require a specific quality for a specific application. Please contact us for consultation.

(3) Mitsubishi shall have no responsibility or liability for any problems involving programmable controller trouble and system trouble caused by DoS attacks, unauthorized access, computer viruses, and other cyberattacks.

TRADEMARKS

Microsoft and Windows are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.
Unicode is either a registered trademark or a trademark of Unicode, Inc. in the United States and other countries.
PLCopen is a registered trademark of PLCopen®.
The company names, system names and product names mentioned in this manual are either registered trademarks or trademarks of their respective companies.
In some cases, trademark symbols such as '™' or '®' are not specified in this manual.

IB(NA)-0300533ENG-C(2201)MEE
MODEL: RD78-P-MCFB-E

MITSUBISHI ELECTRIC CORPORATION

HEAD OFFICE : TOKYO BUILDING, 2-7-3 MARUNOUCHI, CHIYODA-KU, TOKYO 100-8310, JAPAN
NAGOYA WORKS : 1-14 , YADA-MINAMI 5-CHOME , HIGASHI-KU, NAGOYA , JAPAN

When exported from Japan, this manual does not require application to the Ministry of Economy, Trade and Industry for service transaction permission.

Specifications subject to change without notice.