MITSUBISHI ELECTRIC

Programmable Controller

MELSEC iQ-R series

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

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

SAFETY PRECAUTIONS

(Read these precautions before using this product.)

Before using this product, please read this manual and the relevant manuals carefully and pay full attention to safety to handle the product correctly.

The precautions given in this manual are concerned with this product only. Refer to the MELSEC iQ-R Module Configuration Manual for a description of the PLC system safety precautions.

In this manual, the safety precautions are classified into two levels: "⚠ WARNING" and "⚠ CAUTION".

⚠ **WARNING**	Indicates that incorrect handling may cause hazardous conditions, resulting in death or severe injury.
⚠ **CAUTION**	Indicates that incorrect handling may cause hazardous conditions, resulting in minor or moderate injury or property damage.

Under some circumstances, failure to observe the precautions given under "⚠ CAUTION" may lead to serious consequences.

Observe the precautions of both levels because they are important for personal and system safety.

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

[Design Precautions]

⚠WARNING

- Configure safety circuits external to the programmable controller to ensure that the entire system operates safely even when a fault occurs in the external power supply or the programmable controller. Failure to do so may result in an accident due to an incorrect output or malfunction.
  (1) Emergency stop circuits, protection circuits, and protective interlock circuits for conflicting operations (such as forward/reverse rotations or upper/lower limit positioning) must be configured external to the programmable controller.
  (2) When the programmable controller detects an abnormal condition, it stops the operation and all outputs are:
    - Turned off if the overcurrent or overvoltage protection of the power supply module is activated.
    - Held or turned off according to the parameter setting if the self-diagnostic function of the CPU module detects an error such as a watchdog timer error.
  (3) All outputs may be turned on if an error occurs in a part, such as an I/O control part, where the CPU module cannot detect any error. To ensure safety operation in such a case, provide a safety mechanism or a fail-safe circuit external to the programmable controller. For a fail-safe circuit example, refer to "General Safety Requirements" in the MELSEC iQ-R Module Configuration Manual.
  (4) Outputs may remain on or off due to a failure of a component such as a relay and transistor in an output circuit. Configure an external circuit for monitoring output signals that could cause a serious accident.
- In an output circuit, when a load current exceeding the rated current or an overcurrent caused by a load short-circuit flows for a long time, it may cause smoke and fire. To prevent this, configure an external safety circuit, such as a fuse.
- Configure a circuit so that the programmable controller is turned on first and then the external power supply. If the external power supply is turned on first, an accident may occur due to an incorrect output or malfunction.
- For the operating status of each station after a communication failure, refer to manuals for the network used. For the manuals, please consult your local Mitsubishi representative. Incorrect output or malfunction due to a communication failure may result in an accident.

[Design Precautions]

⚠WARNING

- When connecting an external device with a CPU module or intelligent function module to modify data of a running programmable controller, configure an interlock circuit in the program to ensure that the entire system will always operate safely. For other forms of control (such as program modification, parameter change, forced output, or operating status change) of a running programmable controller, read the relevant manuals carefully and ensure that the operation is safe before proceeding. Improper operation may damage machines or cause accidents. When a Safety CPU is used, data cannot be modified while the Safety CPU is in SAFETY MODE.
- Especially, when a remote programmable controller is controlled by an external device, immediate action cannot be taken if a problem occurs in the programmable controller due to a communication failure. To prevent this, configure an interlock circuit in the program, and determine corrective actions to be taken between the external device and CPU module in case of a communication failure.
- Do not write any data to the "system area" and "write-protect area" of the buffer memory in the module. Also, do not use any "use prohibited" signals as an output signal from the CPU module to each module. Doing so may cause malfunction of the programmable controller system. For the "system area", "write-protect area", and the "use prohibited" signals, refer to the user's manual for the module used. For areas used for safety communications, they are protected from being written by users, and thus safety communications failure caused by data writing does not occur.
- If a communication cable is disconnected, the network may be unstable, resulting in a communication failure of multiple stations. Configure an interlock circuit in the program to ensure that the entire system will always operate safely even if communications fail. Incorrect output or malfunction due to a communication failure may result in an accident. When safety communications are used, an interlock by the safety station interlock function protects the system from an incorrect output or malfunction.
- Configure safety circuits external to the programmable controller to ensure that the entire system operates safely even when a fault occurs in the external power supply or the programmable controller. Failure to do so may result in an accident due to an incorrect output or malfunction.
  (1) Machine homing is controlled by two kinds of data: a homing direction and a homing speed. Deceleration starts when the proximity dog signal turns on. If an incorrect homing direction is set, motion control may continue without deceleration. To prevent machine damage caused by this, configure an interlock circuit external to the programmable controller.
  (2) When the module detects an error, the motion slows down and stops or the motion rapidly stops, depending on the stop group setting in parameter. Set the parameter to meet the specifications of a positioning control system. In addition, set the homing parameter and positioning data within the specified setting range.
  (3) Outputs may remain on or off, or become undefined due to a failure of a component such as an insulation element and transistor in an output circuit, where the module cannot detect any error. In a system that the incorrect output could cause a serious accident, configure an external circuit for monitoring output signals.
- If safety standards (ex., robot safety rules, etc.,) apply to the system using the module, drive unit and servomotor, make sure that the safety standards are satisfied.
- Construct a safety circuit externally of the module or drive unit if the abnormal operation of the module or drive unit differs from the safety directive operation in the system.

[Design Precautions]

⚠CAUTION

- Do not install the control lines or communication cables together with the main circuit lines or power cables. Keep a distance of 100 mm or more between them. Failure to do so may result in malfunction due to noise.
- During control of an inductive load such as a lamp, heater, or solenoid valve, a large current (approximately ten times greater than normal) may flow when the output is turned from off to on. Therefore, use a module that has a sufficient current rating.
- After the CPU module is powered on or is reset, the time taken to enter the RUN status varies depending on the system configuration, parameter settings, and/or program size. Design circuits so that the entire system will always operate safely, regardless of the time.
- Do not power off the programmable controller or reset the CPU module while the settings are being written. Doing so will make the data in the flash ROM and SD memory card undefined. The values need to be set in the buffer memory and written to the flash ROM and SD memory card again. Doing so also may cause malfunction or failure of the module.

[Security Precautions]

⚠WARNING

- To maintain the security (confidentiality, integrity, and availability) of the programmable controller and the system against unauthorized access, denial-of-service (DoS) attacks, computer viruses, and other cyberattacks from external devices via the network, take appropriate measures such as firewalls, virtual private networks (VPNs), and antivirus solutions.

[Installation Precautions]

⚠WARNING

- Shut off the external power supply (all phases) used in the system before mounting or removing the module. Failure to do so may result in electric shock or cause the module to fail or malfunction.

[Installation Precautions]

⚠CAUTION

- Use the programmable controller in an environment that meets the general specifications in the Safety Guidelines included with the base unit. Failure to do so may result in electric shock, fire, malfunction, or damage to or deterioration of the product.
- To mount a module, place the concave part(s) located at the bottom onto the guide(s) of the base unit, push in the module, and fix it with screw(s). Incorrect interconnection may cause malfunction, failure, or drop of the module.
- To mount a module with no module fixing hook, place the concave part(s) located at the bottom onto the guide(s) of the base unit, push in the module, and fix it with screw(s). Incorrect interconnection may cause malfunction, failure, or drop of the module.
- Tighten the screws within the specified torque range. Undertightening can cause drop of the screw, short circuit, or malfunction. Overtightening can damage the screw and/or module, resulting in drop, short circuit, or malfunction.
- When using an extension cable, connect it to the extension cable connector of the base unit securely. Check the connection for looseness. Poor contact may cause malfunction.
- When using an SD memory card, fully insert it into the SD memory card slot. Check that it is inserted completely. Poor contact may cause malfunction.
- Securely insert an extended SRAM cassette or a battery-less option cassette into the cassette connector of the CPU module. After insertion, close the cassette cover and check that the cassette is inserted completely. Poor contact may cause malfunction.
- Do not directly touch any conductive parts of the module, SD memory card, extended SRAM cassette, battery-less option cassette, or connector. Doing so can cause malfunction or failure of the module.

[Wiring Precautions]

⚠WARNING

- Shut off the external power supply (all phases) used in the system before installation and wiring. Failure to do so may result in electric shock or cause the module to fail or malfunction.
- After installation and wiring, attach a blank cover module (RG60) to each empty slot and an included extension connector protective cover to the unused extension cable connector before powering on the system for operation. Failure to do so may result in electric shock.

[Wiring Precautions]

⚠CAUTION

- Individually ground the FG and LG terminals of the programmable controller with a ground resistance of 100 ohms or less. Failure to do so may result in electric shock or malfunction.
- Use applicable solderless terminals and tighten them within the specified torque range. If any spade solderless terminal is used, it may be disconnected when the terminal screw comes loose, resulting in failure.
- Check the rated voltage and signal layout before wiring to the module, and connect the cables correctly. Connecting a power supply with a different voltage rating or incorrect wiring may cause fire or failure.
- Connectors for external devices must be crimped or pressed with the tool specified by the manufacturer, or must be correctly soldered. Incomplete connections may cause short circuit, fire, or malfunction.
- Securely connect the connector to the module. Poor contact may cause malfunction.
- Do not install the control lines or communication cables together with the main circuit lines or power cables. Keep a distance of 100 mm or more between them. Failure to do so may result in malfunction due to noise.
- Place the cables in a duct or clamp them. If not, dangling cables may swing or inadvertently be pulled, resulting in malfunction or damage to the modules.
  In addition, the weight of the cables may put stress on modules in an environment of strong vibrations and shocks.
  Do not clamp the extension cables with the jacket stripped. Doing so may change the characteristics of the cables, resulting in malfunction.
- Check the interface type and correctly connect the cable. Incorrect wiring (connecting the cable to an incorrect interface) may cause failure of the module and external device.
- Tighten the terminal screws or connector screws within the specified torque range. Undertightening can cause drop of the screw, short circuit, fire, or malfunction. Overtightening can damage the screw and/or module, resulting in drop, short circuit, fire, or malfunction.

[Wiring Precautions]

⚠CAUTION

- When disconnecting the cable from the module, do not pull the cable by the cable part. For the cable with connector, hold the connector part of the cable. For the cable connected to the terminal block, loosen the terminal screw. Pulling the cable connected to the module may result in malfunction or damage to the module or cable.
- Prevent foreign matter such as dust or wire chips from entering the module. Such foreign matter can cause a fire, failure, or malfunction.
- A protective film is attached to the top of the module to prevent foreign matter, such as wire chips, from entering the module during wiring. Do not remove the film during wiring. Remove it for heat dissipation before system operation.
- Programmable controllers must be installed in control panels. Connect the main power supply to the power supply module in the control panel through a relay terminal block. Wiring and replacement of a power supply module must be performed by qualified maintenance personnel with knowledge of protection against electric shock. For wiring, refer to the MELSEC iQ-R Module Configuration Manual.
- For Ethernet cables to be used in the system, select the ones that meet the specifications in the user's manual for the module used. If not, normal data transmission is not guaranteed.

[Startup and Maintenance Precautions]

⚠WARNING

- Do not touch any terminal while power is on. Doing so will cause electric shock or malfunction.
- Correctly connect the battery connector. Do not charge, disassemble, heat, short-circuit, solder, or throw the battery into the fire. Also, do not expose it to liquid or strong shock. Doing so will cause the battery to produce heat, explode, ignite, or leak, resulting in injury and fire.
- Shut off the external power supply (all phases) used in the system before cleaning the module or retightening the terminal screws, connector screws, or module fixing screws. Failure to do so may result in electric shock.

[Startup and Maintenance Precautions]

⚠CAUTION

- When connecting an external device with a CPU module or intelligent function module to modify data of a running programmable controller, configure an interlock circuit in the program to ensure that the entire system will always operate safely. For other forms of control (such as program modification, parameter change, forced output, or operating status change) of a running programmable controller, read the relevant manuals carefully and ensure that the operation is safe before proceeding. Improper operation may damage machines or cause accidents.
- Especially, when a remote programmable controller is controlled by an external device, immediate action cannot be taken if a problem occurs in the programmable controller due to a communication failure. To prevent this, configure an interlock circuit in the program, and determine corrective actions to be taken between the external device and CPU module in case of a communication failure.
- Do not disassemble or modify the modules. Doing so may cause failure, malfunction, injury, or a fire.
- Use any radio communication device such as a cellular phone or PHS (Personal Handy-phone System) more than 25 cm away in all directions from the programmable controller. Failure to do so may cause malfunction.
- Shut off the external power supply (all phases) used in the system before mounting or removing the module. Failure to do so may cause the module to fail or malfunction.
- Tighten the screws within the specified torque range. Undertightening can cause drop of the component or wire, short circuit, or malfunction. Overtightening can damage the screw and/or module, resulting in drop, short circuit, or malfunction.
- After the first use of the product, do not perform each of the following operations more than 50 times (IEC 61131-2/JIS B 3502 compliant).
  Exceeding the limit may cause malfunction.
  - Mounting/removing the module to/from the base unit
  - Inserting/removing the extended SRAM cassette or battery-less option cassette to/from the CPU module
  - Mounting/removing the terminal block to/from the module
- After the first use of the product, do not insert/remove the SD memory card to/from the CPU module more than 500 times. Exceeding the limit may cause malfunction.
- Do not touch the metal terminals on the back side of the SD memory card. Doing so may cause malfunction or failure of the module.
- Do not touch the integrated circuits on the circuit board of an extended SRAM cassette or a batteryless option cassette. Doing so may cause malfunction or failure of the module.
- Do not drop or apply shock to the battery to be installed in the module. Doing so may damage the battery, causing the battery fluid to leak inside the battery. If the battery is dropped or any shock is applied to it, dispose of it without using.
- Startup and maintenance of a control panel must be performed by qualified maintenance personnel with knowledge of protection against electric shock. Lock the control panel so that only qualified maintenance personnel can operate it.
- Before handling the module, touch a conducting object such as a grounded metal to discharge the static electricity from the human body. Failure to do so may cause the module to fail or malfunction.

[Startup and Maintenance Precautions]

⚠CAUTION

- Before testing the operation, set a low speed value for the speed limit parameter so that the operation can be stopped immediately upon occurrence of a hazardous condition.
- Confirm and adjust the program and each parameter before operation. Unpredictable movements may occur depending on the machine.
- When using the absolute position system function, on starting up, and when the module or absolute position motor has been replaced, always perform a homing.
- Before starting the operation, confirm the brake function.
- Do not perform a megger test (insulation resistance measurement) during inspection.
- After maintenance and inspections are completed, confirm that the position detection of the absolute position detection function is correct.
- Lock the control panel and prevent access to those who are not certified to handle or install electric equipment.

[Operating Precautions]

⚠CAUTION

- When changing data and operating status, and modifying program of the running programmable controller from an external device such as a personal computer connected to an intelligent function module, read relevant manuals carefully and ensure the safety before operation. Incorrect change or modification may cause system malfunction, damage to the machines, or accidents.
- Do not power off the programmable controller or reset the CPU module while the setting values in the buffer memory are being written to the flash ROM in the module. Doing so will make the data in the flash ROM and SD memory card undefined. The values need to be set in the buffer memory and written to the flash ROM and SD memory card again. Doing so also may cause malfunction or failure of the module.
- Note that when the reference axis speed is specified for interpolation operation, the speed of the partner axis (2nd, 3rd, or 4th axis) may exceed the speed limit value.
- Do not go near the machine during test operations or during operations such as teaching. Doing so may lead to injuries.

[Disposal Precautions]

⚠CAUTION

- When disposing of this product, treat it as industrial waste.
- When disposing of batteries, separate them from other wastes according to the local regulations. For details on battery regulations in EU member states, refer to the MELSEC iQ-R Module Configuration Manual.

[Transportation Precautions]

⚠CAUTION

- When transporting lithium batteries, follow the transportation regulations. For details on the regulated models, refer to the MELSEC iQ-R Module Configuration Manual.
- The halogens (such as fluorine, chlorine, bromine, and iodine), which are contained in a fumigant used for disinfection and pest control of wood packaging materials, may cause failure of the product. Prevent the entry of fumigant residues into the product or consider other methods (such as heat treatment) instead of fumigation. The disinfection and pest control measures must be applied to unprocessed raw wood.

INTRODUCTION

Thank you for purchasing the Mitsubishi Electric MELSEC iQ-R series programmable controllers.
This manual describes the functions and programming of the relevant products listed below. 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.

- [RD78G]: Symbols indicating that it corresponds to only RD78G
- [RD78GH]: Symbols indicating that it corresponds to only RD78GH
- Underlined variables (AxisName, etc.): Variables defined by users

COMPLIANCE WITH EMC AND LOW VOLTAGE DIRECTIVES

Method of ensuring compliance

To ensure that Mitsubishi programmable controllers maintain EMC and Low Voltage Directives when incorporated into other machinery or equipment, certain measures may be necessary. Please refer to one of the following manuals.
📖MELSEC iQ-R Module Configuration Manual
📖Safety Guidelines (This manual is included with the base unit.)
The CE mark on the side of the programmable controller indicates compliance with EMC and Low Voltage Directives.

Additional measures

To ensure that this product maintains EMC and Low Voltage Directives, please refer to one of the following manuals.
📖MELSEC iQ-R Module Configuration Manual
📖Safety Guidelines (This manual is included with the base unit.)

CONTENTS

SAFETY PRECAUTIONS 1
INTRODUCTION 11
COMPLIANCE WITH EMC AND LOW VOLTAGE DIRECTIVES 11
RELEVANT MANUALS 22
TERMS 23
GENERIC TERMS AND ABBREVIATIONS 25
HOW TO READ THIS MANUAL 26
FUTURE SUPPORT PLANNED 33

PART 1 BASIC SPECIFICATIONS

CHAPTER 1 AXIS SETTING 36

1.1 Axis 36
Relevant variables 36
Maximum number of controlled axes 36
Required settings for axes 37
Axis variables 37
Axis variable initialization timing 38
How to specify in a user program 39
State transition 39
Precautions 40
1.2 Axis Type 41
Relevant variables/FB 41
Real drive axis 44
Real encoder axis 45
Virtual drive axis 50
Virtual encoder axis 50
Virtual linked axis 54
1.3 Connectable Stations 55
Maximum number of stations that can be connected 55
Precaution 55
1.4 Axis Assignment 56
Relevant variables 57
Slave object settings 58
PDO mapping settings 59
Setting procedure 59
Memory usage 59
Slave object list 60
Motion system function/command and slave object 64
1.5 Axes Group 65
Relevant variables/FBs 65
Settings required for axes group 69
Axes group variables 70
Axes group variable initialization timing 70
How to specify in a user program 71
Enabling axes group 71
Axes group enabled 73

Axes group disabled . . . 75
Precautions . . . 76
1.6 Axes Group Assignment . . . 77
Relevant variables/FB . . . 77
Setting with Engineering tool . . . 78
Memory usage . . . 78

CHAPTER 2 AXIS MANAGEMENT FUNCTIONS 79

2.1 Technical Units . . . 79
Relevant variables . . . 79
Setting methods . . . 79
Precautions . . . 82
2.2 Positioning Range . . . 83
Relevant variables . . . 83
Set position . . . 84
Combination with other functions . . . 85
Precautions . . . 85
2.3 Velocity Range . . . 86
Relevant variables . . . 86
Velocity set range . . . 87
2.4 Servo ON/OFF . . . 88
Relevant variables/FB . . . 88
Control details . . . 93
Precautions . . . 94
2.5 Follow Up . . . 95
Relevant variables . . . 95
Disabling follow up . . . 96
Precautions . . . 97
2.6 Absolute Position Control . . . 98
Relevant variables . . . 99
Current position restoration . . . 100
Absolute position data backup . . . 103
Current position restoration for each axis type . . . 105
Precautions . . . 108

CHAPTER 3 BASIC FUNCTIONS 109

3.1 Operation Cycle . . . 109
Relevant variables . . . 109
System basic cycle . . . 109
Operation cycle settings . . . 110
Operation cycle assignment . . . 110
Send/Receive data refresh of network connected devices . . . 111
Motion synchronization station link device auto refresh . . . 113
Buffer memory refresh . . . 113
Axis operation process high-speed mode . . . 114
Operation cycle 31.25 setting [RD78GH] . . . 116
3.2 Add-on Function . . . 118
Relevant variables . . . 118
Add-on library configuration . . . 119
Management of add-on library . . . 120

Loading add-on library . . . 120
3.3 System Memory Settings . . . 121
Relevant variables . . . 121
System memory (RAM) . . . 122
System memory (backup RAM) . . . 123
Precautions . . . 123
3.4 Software Reboot . . . 124
Relevant variables . . . 124
Reset . . . 124
Clear . . . 125
Quick clear . . . 125
Precautions . . . 125

PART 2 MOTION CONTROL

CHAPTER 4 START AND STOP 128

4.1 Start . . . 128
Relevant variables . . . 128
Starting conditions . . . 130
4.2 Retrigger/Continuous Update . . . 131
4.3 Multiple Start (Buffer Mode) . . . 132
Relevant variables . . . 134
Multiple start during single axis operation . . . 135
Multiple start during multiple axes operation . . . 145
Option of buffering FB . . . 152
Near pass . . . 153
Combination with retrigger/continuous update . . . 154
Precautions . . . 159
4.4 Stop . . . 162
Relevant variables/FB . . . 163
Stop cause lists . . . 170
Types of stop processes . . . 172
Order of priority for stop process . . . 175
Overrun at stop cause occurrence . . . 176
Stop during single axis operation . . . 177
Stop during axes group operation . . . 184
4.5 Forced Stop . . . 192
Relevant variables . . . 192
Operation at the forced stop occurrence . . . 194
Operation at the forced stop release . . . 196
Check method . . . 197

CHAPTER 5 HOMING 198

5.1 Overview . . . 198
Relevant variables/FB . . . 199
Homing Request . . . 201
Driver homing method . . . 202
Data set homing method . . . 204
Precautions . . . 204
5.2 Operation Setting for Incompletion of Homing . . . 205

Relevant variables . . . 205
FBs that can be started when homing is incompletion . . . 205
Homing request clear . . . 205
Precautions . . . 205

CHAPTER 6 AXIS CONTROL FUNCTION 206

6.1 **Single Axis Positioning Control . . . 206**
Relevant FBs . . . 206
Absolute Positioning Control . . . 210
Relative Positioning Control . . . 218
6.2 **Single Axis Speed Control . . . 220**
Relevant FB . . . 220
Control details . . . 222
BufferMode . . . 223
6.3 **Single Axis Manual Control . . . 224**
Relevant variables/FB . . . 224
Types of manual control . . . 227
JOG operation . . . 227
6.4 **Multiple Axes Positioning Control . . . 232**
Relevant variables/FBs . . . 232
Operation in multiple axes positioning control . . . 242
Positioning speed in multiple axes positioning control . . . 243
Command unit in the multiple axes positioning control . . . 251
Acceleration/deceleration processing . . . 253
State transition in the multiple axes positioning control . . . 253
Linear interpolation control . . . 254
Circular interpolation control . . . 263

CHAPTER 7 DIRECT CONTROL 275

7.1 **Velocity Control . . . 276**
Relevant variables/FB . . . 276
Control details . . . 280
BufferMode . . . 281
Precautions . . . 286
7.2 **Torque Control . . . 287**
Relevant variables/FB . . . 287
Control details . . . 291
BufferMode . . . 294
Precautions . . . 303

CHAPTER 8 RELEVANT FUNCTIONS to POSITION 304

8.1 **Current Position Change Function . . . 304**
Relevant FBs . . . 304
Position setting range . . . 310
Current position change . . . 310
Trigger current position change . . . 312
Cancel . . . 313
8.2 **Command In-position . . . 315**
Relevant variables . . . 315
Command in-position width check . . . 316

8.3 Software Stroke Limit . . . 317
Relevant variables . . . 317
Definition of movable range . . . 317
Specifications of the software stroke limit check . . . 322
Precautions . . . 322
8.4 Hardware Stroke Limit . . . 323
Relevant variables . . . 323
Hardware stroke limit check details . . . 325
Returning from outside the hardware stroke limit range . . . 326
Confirm that the hardware stroke limit is disabled . . . 326
Precautions . . . 327

CHAPTER 9 RELEVANT FUNCTIONS TO VELOCITY 330

9.1 Acceleration/deceleration Processing Function . . . 330
Relevant variables . . . 332
Acceleration/deceleration method . . . 336
Acceleration/deceleration waveform adjustment when specifying jerk . . . 343
Operation when specifying acceleration/deceleration 0 . . . 345
Acceleration limit value/deceleration limit value . . . 346
Jerk limit value . . . 347
Limit of acceleration time/deceleration time . . . 347
Automatic deceleration . . . 348
Changing control of jerk acceleration/deceleration . . . 349
Acceleration/deceleration method and multiple start . . . 354
9.2 Speed Limit . . . 356
Relevant variables . . . 356
How to set the speed limit function . . . 356
Operation when the speed limit value is exceeded . . . 357
Operation setting when the speed limit value is exceeded . . . 358
9.3 Override Function . . . 359
Relevant variables/FBs . . . 360
How to change control data . . . 366

CHAPTER 10 RELEVANT FUNCTIONS TO TORQUE 371

10.1 Torque limit . . . 371
Relevant variables . . . 371
Slave object map . . . 371
Torque limit value setting range . . . 372
Torque limit maximum value setting range . . . 372
Precautions . . . 372
10.2 Torque Limit Value Change Function . . . 373
Relevant variables/FB . . . 374
How to change control data . . . 380

CHAPTER 11 CONTROL SUB FUNCTIONS 381

11.1 Compensation Function . . . 381
Relevant variables . . . 381
Driver unit conversion function . . . 381
11.2 Command Filter . . . 384
Relevant FBs . . . 384

Smoothing filter . . . 393
Movement direction limit filter . . . 397
Speed limit filter . . . 400
Backlash compensation filter . . . 403
Precautions . . . 411
11.3 Input Variable Change in Execution . . . 412
Relevant variables . . . 412
Target position/movement distance change . . . 413
Set velocity change . . . 416
Acceleration/deceleration and acceleration/deceleration time change . . . 417

CHAPTER 12 COMMON FUNCTIONS 418

12.1 External Signal Selection . . . 418
Relevant variables . . . 418
SIGNAL_SELECT structure . . . 418
TARGET_REF structure . . . 420
12.2 Touch Probe . . . 423
Relevant FBs . . . 423
Touch probe valid . . . 426
Touch probe invalid . . . 433
12.3 Slave Emulation . . . 435
Relevant variables . . . 435
Setting procedures . . . 435
Operation during emulate function . . . 435
Specification of the emulated slave station . . . 436
Precaution . . . 437

CHAPTER 13 SYNCHRONOUS CONTROL 438

13.1 Overview of Single Axis Synchronization Control FBs . . . 438
13.2 Axis Configuration . . . 439
13.3 Master Axis Data Source Selection . . . 440
13.4 Start and Stop Operation . . . 441
13.5 Required Slave Object . . . 442
13.6 Restrictions . . . 442
13.7 Cam Operation . . . 443
Relevant variable/FB . . . 443
Control details . . . 446
Precautions . . . 460
13.8 Gear Operation . . . 461
Relevant FB . . . 461
Control details . . . 463
Precautions . . . 465
13.9 Addition/Subtraction Positioning . . . 466
Relevant FB . . . 466
Control details . . . 469

CHAPTER 14 OPERATION PROFILE FUNCTION 473

14.1 Operation Profile Data . . . 473
Relevant variables/FB . . . 474
Creating operation profile data . . . 488

Operation profile data types . . . 489
Operating operation profile data . . . 502
Control of operation profile data . . . 512
Start mode . . . 514
Sub function of operation profile data control . . . 514
14.2 Operation Profile Data Format . . . 523
Operation profile data details . . . 523

PART 3 CONTROL/OPERATION/MAINTENANCE

CHAPTER 15 LOGGING 530

15.1 Data Logging/Real-Time Monitor . . . 530
Relevant variables . . . 530
Data logging function . . . 531
Real-time monitor . . . 532
Steps until the collected data is saved . . . 532
Logging setting . . . 535
Data to be collected (DATA) . . . 535
Logging setting (LOGGING) . . . 536
Trigger condition (TRIGGERCONDITION) . . . 541
Application function . . . 543
Precautions . . . 545
Folder configuration . . . 547
15.2 Data Logging Format . . . 548
CSV file data specifications . . . 548
JSON File Data Specifications . . . 550
15.3 Logging Data (JSON format) . . . 553
Basic specifications . . . 553
Whole configuration . . . 553
Data specification . . . 554

CHAPTER 16 RAS FUNCTIONS 557

16.1 Execution Time Monitor . . . 557
Relevant variables . . . 558
Control details . . . 559
Processing time over check . . . 561
Errors . . . 562
16.2 History Data . . . 563
Relevant variables . . . 564
Event history function . . . 565
Positioning data history . . . 568
Precautions . . . 568
16.3 Servo System Recorder . . . 569
Relevant variables . . . 570
Control details . . . 571
Restart from the logging interruption . . . 578
How to check the logging data . . . 579
How to delete the logging setting file . . . 582
Precautions . . . 584

CHAPTER 17 MOTION SERVICE PROCESSING 587
17.1 Operation Overview. . . . 587
17.2 Processing Description 588
CHAPTER 18 HOW TO CONTROL FROM CPU MODULE 589
18.1 Specifications of I/O Signals with CPU Modules 589
18.2 Using MCFB 592
Relevant variable. . . . 592
Control details 592
CHAPTER 19 FILE CONTROL 596
19.1 Memory and File 596
Relevant variables. . . . 596
Drive 597
Path setting 599
Operation that can be executed 601
Storage files. . . . 601
Module extended parameters 602
Data backup 603
Operation folder. . . . 604
Precautions 604
19.2 Parameter Read/Write Function 605
Relevant FBs 605
Control Details 609
The response code (SDO Abort Code) 612
Precautions 612
19.3 File Transfer Function 613
Relevant variables 613
Access control setting 614
Functions of file transfer 614
Command format. . . . 615
Log file 619
Precautions 619
19.4 SD Memory Card 620
Relevant variables 620
SD memory card handling 620
Specifications of SD memory card 620
Installation and removal of SD memory card. . . . 620
Precautions 621
CHAPTER 20 SECURITY 622
CHAPTER 21 MODULE SOFTWARE INSTALLATION 624
21.1 Motion System Software Installation 625
Relevant variables 625
Software file configuration of motion system 626
Motion software package file configuration details 626
How to batch install the motion system software. . . . 627
Checking software version 629
21.2 Basic System Software Update Function 631

Software management 631
21.3 **Boot Software Update Function** **634**

CHAPTER 22 TROUBLESHOOTING 637

22.1 LED Control **637**
22.2 Checking Errors and Warnings **638**
Relevant variables 639
Control details 640
Filter setting 640
22.3 Error and Warning Reset **641**
Relevant variables/FB 642
System error reset 646
Axis error reset 649
Axes group error reset 651
22.4 List of Warning Codes **654**
22.5 List of Error Codes **662**
22.6 List of Logging Error Codes **692**
22.7 List of Event Codes **693**

PART 4 LISTS

CHAPTER 23 List of Data Type 696

CHAPTER 24 List of Variables 701

CHAPTER 25 List of Supported Axis Type 716

CHAPTER 26 List of FBs 721

CHAPTER 27 List of Add-on Library 723

CHAPTER 28 List of Boot software 726

APPENDICES 728

Appendix 1 Connectable device to CC-Link IE TSN **728**
Common Items for all devices 728
MR-J5(W)-G (Cyclic synchronous mode) connection method 731
MR-J5(W)-G (Profile mode) connection method 738
How to connect MR-JET-G 743
Relevant functions 744
Precautions 747
Appendix 2 Project Setting Example **748**
Library Registration Procedure 748
Library Update Procedure 750
Precautions 752
Appendix 3 Restrictions by the version **753**

INDEX 760

REVISIONS 762

WARRANTY 763
TRADEMARKS 764

RELEVANT MANUALS

Manual name [manual number]	Description	Available form
MELSEC iQ-R Motion Module User's Manual(Application) [IB-0300411ENG] (This manual)	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(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(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
MELSEC iQ-R Programming Manual (Motion Control Function Blocks) [IB-0300533ENG]	Motion control function blocks, variables, and programming	Print book e-Manual PDF

For programs, refer to the following.

MELSEC iQ-R Programming Manual (Program Design)

This manual does not include information on the network function blocks.

For details on the network function blocks, refer to the following.

MELSEC iQ-R Ethernet/CC-Link IE Function Block Reference

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 using a slave label
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.
Feed machine position	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.
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
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 station that updates a link device in the Motion module
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 syncronization station	A station that performs the synchronous communication among motion control stations
MR-J5-G	MR-J5-_G_(-RJ) 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.
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

Term	Description
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
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.
Return	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 station other than motion syncronization stations
Start	Executing a motion FB for the first time to an axis or an axes group
Switching speed	• The set velocity of the motion control FB in execution when specifying Aborting • The set velocity when the motion control FB in execution 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.
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.

Term	Description
CC-Link IE	A generic terms of the followings. • 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	The abbreviation for the MELSEC iQ-R series CPU module
csp	The abbreviation for cyclic synchronous position mode (One of the control modes for the driver side)
cst	The abbreviation for cyclic synchronous torque mode (One of the control modes for the driver side)
csv	The 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	The abbreviation for homing mode (One of the control modes for the driver side)
MCFB	The abbreviation for Motion Control FB
MR-J5(W)-G	A generic term for MR-J5-_G_(-RJ)/MR-J5W_-_G servo amplifier
PDO	A communication object of CANopen (Process Data Object)
RAS	The abbreviation for Reliability, Availability, and Serviceability. This term refers to usability of automated equipment.
RD78G(H)	A generic term for RD78G_, RD78GH_ (high performance version)
SDO	A communication object of CANopen (Service Data Object)
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 CCLink 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
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	Cyclic transmission and 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

HOW TO READ THIS MANUAL

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

How to use MCFB

Procedure to use MCFB is described as follows.

1. Instance of the MCFB (Global label or local level) is created by engineering tool.
2. Program which sets I/O argument is created for the created MCFB instance.
3. When turning Execute input or Enable input of MCFB, the control will be executed. For details, refer to the specification of each control and "Start" and "Stop".

Point

When one MCFB instance is executed (including during buffering), the multiple axes can not be shared by changing axis input (AXIS_REF input or AXES_GROUP_REF input). The axis input of MCFB can be changed only at the nonexecution. When the axis input is changed in execution, a warning will occur and the change will be ignored. Therefore, the MCFB instance is required to be created for only the numbers of the axis to be controled at the same time.

Operation of this function for each system status

The MCFB can be executed only during RUN.
○: Possible, ×: Not possible

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

FB types

The MCFB is classified according to the operation details and the execution method.

■Motion FB/administrative FB/general FB

The following types are included in the MCFB according to the operation details.

Type	Description
Motion FB	The MCFB that takes an axis or an axes group for the argument and changes the axis status or the axes group status by execution.
Administrative FB	The MCFB 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)
General FB	The MCFB that does not take an axis or an axes group for the argument.

[Combination of FBs]

- The motion FB can be basically executed only one to an axis or an axes group. However, some FBs can be executed at the same time. For details, refer to each FB specifications. For specifications for multiple start, refer to the following.
  ☞ Page 132 Multiple Start (Buffer Mode)
- The administrative FB can basically execute multiple instances to an axis or an axes group at the same time. For the case when starting the multiple administrative FBs, refer to each FB specifications.
- Even if an administrative FB is executed while a motion FB is executed, the axis status or the axes group status will not be basically changed. However, some FBs can cause a specific state transition. For details, refer to each FB specifications.
- The general FB can simultaneously execute multiple instances. Since it is not related to the axis, it does not affect the motion FB or the administrative FB each other.

■Execute type/Enable type

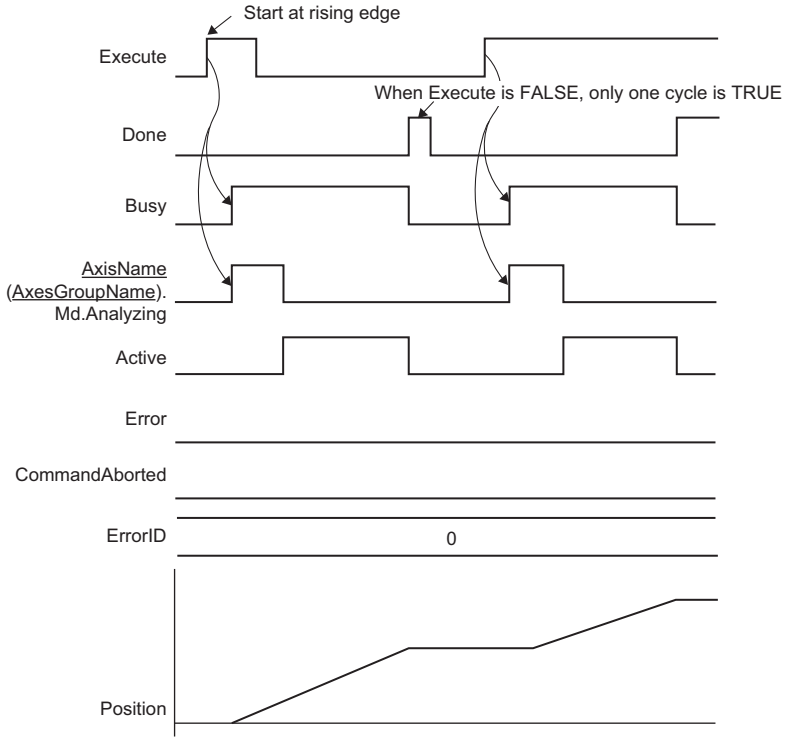
Some MCFBs are executed at an Execute input, and the others are executed at an Enable input. The basic operation of each FB is shown below. However, some FBs have different specifications. For details, refer to each FB specifications.

[Basic operation of Execute type FB]

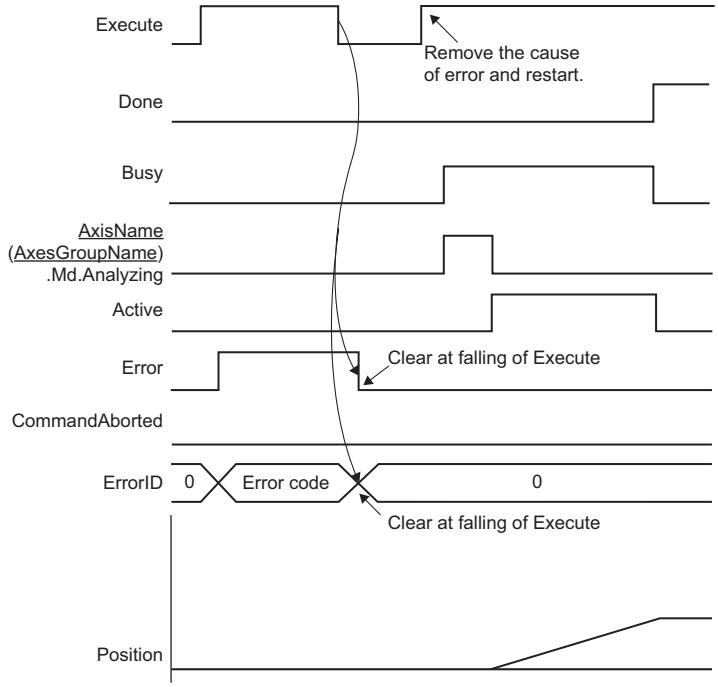
- The Execute type FB reads the input parameters at a rising edge of Execute, then it starts operation. Once operation is started, even if setting FALSE to Execute, the operation will be continued until it is completed.
- When operation is started, only one of outputs among Busy, Done, Error, and CommandAborted, turns TRUE.
- Done, Error, ErrorID, and CommandAborted are reset at a falling edge of Execute. Busy and Active are not affected.
- When changing the input parameter during operation, the change is reflected at restart (retrigger) of Execute or continuous update using the ContinuousUpdate input. For details, refer to the following.
  ☞ Page 131 Retrigger/Continuous Update
- AxisName.Md.Analyzing/AxesGroupName.Md.Analyzing turn TRUE from the rising edge of Busy, AxisName.Md.Analyzing/AxesGroupName.Md.Analyzing turn FALSE after starting the operation.
- When the Execute is used in pulse, Done signal output is outputted in pulse.

Basic operation of Enable type FB is shown below.

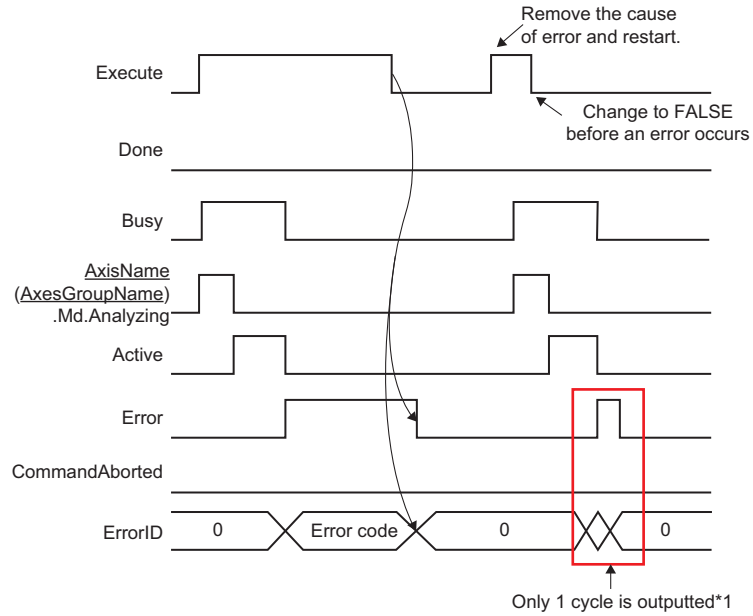
[When the FB is normal]



[When the I/O variables are abnormal]



[When the input variables are abnormal]



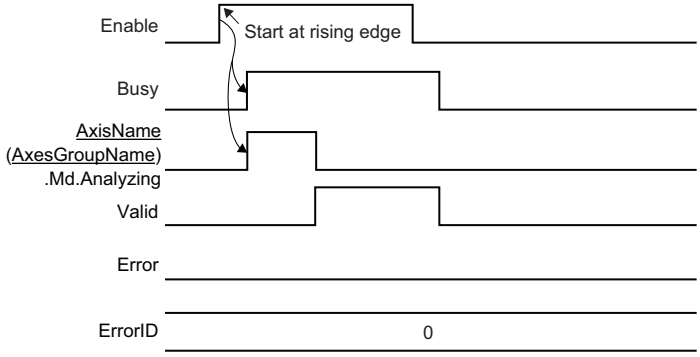
*1 The following operation is carried out when termination condition of FB (Execute = FALSE) is met.
In the case of those FBs which do not related with axis or which do not stop with decelerating, Error turns to TRUE and then ErrorID is outputted for 1 cycle only. In the case of FB which needs deceleration stop, Error turns to TRUE and ErrorID is maintained. At stop completion of axis, Error turns to FALSE and then ErrorID is cleared.

[Basic operation of Enable type FB]

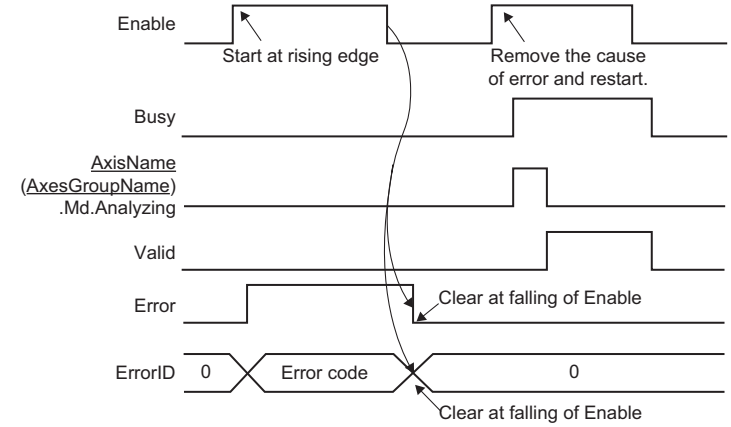
- The Enable type FB is continuously executed while Enable is TRUE.
- Valid indicates that the output is a valid value. After Valid turned FALSE, each output does not change.
- Only one of outputs among Valid/Enabled/Busy, Error, and CommandAborted, turns TRUE.
- AxisName.Md.Analyzing/AxesGroupName.Md.Analyzing turn TRUE from the rising edge of Busy, AxisName.Md.Analyzing/AxesGroupName.Md.Analyzing turn FALSE after starting the operation.

Basic operation of Enable type FB is shown below.

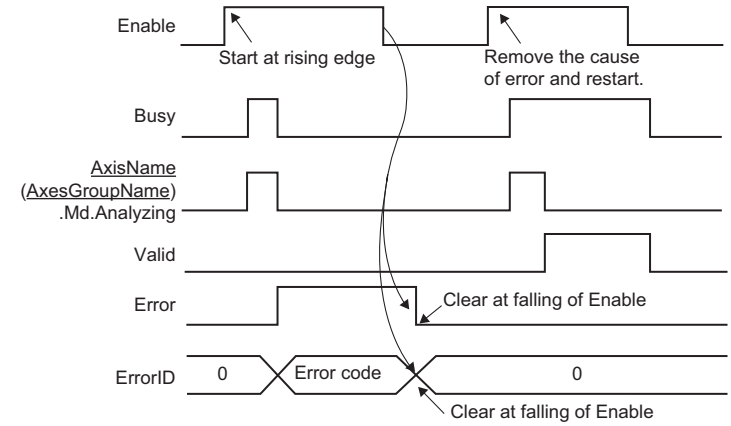
[When the FB is normal]



[When the I/O variables are abnormal]



[When the input variables change is abnormal]



Motion control types

The following types exist in the axis/axes group control that can be executed by the motion MCFB. For details, refer to each function.

Large classification	Middle classification	Small classification	Reference
Axis control	Single axis control	Positioning control	Page 206 Single Axis Positioning Control
		Continuous control	Page 275 DIRECT CONTROL
		Synchronous control	Page 438 SYNCHRONOUS CONTROL Page 384 Command Filter
		Manual control	Page 224 Single Axis Manual Control
		Homing control	Page 198 HOMING
Axes group control	Multiple axes control	Positioning control	

The following shows generic terms for the motion MCFBs which perform each control.

Generic term for FB	Definition
Positioning control FB	A motion MCFB which turns the axis (or axes group) state to "DiscreteMotion" (or "GroupMoving") and moves it to the target position
Continuous control FB	A motion MCFB which turns the axis state to "ContinuousMotion" and controls it continuously
Synchronous control FB	A motion MCFB which has Master (master axis) and Slave (slave axis) of I/O variables and performs synchronous control by turning Slave (slave axis) state to "SynchronizedMotion"
Homing FB	A motion MCFB which turns the axis state to "Homing" and controls continuously

Error processing

If an error occurs while the MCFB is executed, Error turns TRUE and the error code is output to Error ID. At this time, the axis status transits into "1: ErrorStop" and the axes group status transits into "1: GroupErrorStop".

When an available axis transits into the "1: ErrorStop" status, all buffering FBs are interrupted. The interrupted FBs' Error turns TRUE.

After that, an error reset needs to be executed to start the axis/axes group. For how to check an error of the axis/axes group and how to reset the error, refer to the following.

☞ Page 638 Checking Errors and Warnings, ☞ Page 641 Error and Warning Reset

Point

A value same as one of the following (it depends on the control) is output to ErrorID. However, the warning code is not output. For remedy of errors, refer to the following.

☞ Page 637 TROUBLESHOOTING

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

The error codes to be output to MCFB ErrorID are shown below.

Error code	Description
0400H	No response from the Motion module within default time. Execute the FB again.
1C00H	Corresponds to the error code (1800H to 180FH) of the dedicated instruction used within the FB. For the error code of dedicated instruction, 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	

Units used in control

Units of the position, speed, acceleration/deceleration, and jerk that are used in the motion system follow the technical units of the axis to be used. For details, refer to the following.

☞ Page 79 Technical Units

The following types exist in the control value such as the position and the speed that are used in the motion system. For details on the control value related to the position and the speed, refer to the following.

☞ Page 83 Positioning Range, ☞ Page 86 Velocity Range

Name	Meaning
Commanded value	A value based on an input to the MCFB (the target value). Commanded position and commanded speed, etc.
Set value	The curent control value that is generated by motion operation. Set position and set velocity, etc.
Actual value	A value obtained by converting the actual value from the slave device on the real axis into the technical units of the axis. Actual position and actual velocity, etc.

MCFB input/output specifications

The common specifications of MCFB I/O variables are described in this section.

■I/O variable/input variable/output variable

The following shows the variables and their class.

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

For details, refer to specifications of each FB.

■Data type

In this part, the variable type is indicated. For details, refer to the following.

☞ Page 696 List of Data Type

■Enumeration type constant

Enumeration type constant which is used for various parameters, monitor data, or MCFBs actually uses INT type values. On the engineering tool, "enumeration type name_enumerator", INT type global data is available. For details on enumeration type and enumerator, refer to the following.

☞ Page 696 ENUM enumerator

Ex.

The label which is available for MC_BUFFER_MODE type enumerator of mcBuffered on the engineering tool is "MC_BUFFER_MODE_mcBuffered" and Constant "1" is set.

■Omission of input variable

When omitting to input FBs, the default value that is defined for each FB is applied. For details on the default value, refer to specifications of each MCFB.

When omitting to input the speed, etc. in the multiple start FB, the input value of the previous FB is taken over. For details, refer to the following.

☞ Page 132 Multiple Start (Buffer Mode)

■Refresh timing of input/output

Each argument of the FB is refreshed at a 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 the MCFB in the CPU module side, needs to specify the I/O No. of the target Motion module to the I/O variables (ex. AxisName.AxisRef.StartIO) that are used in the MCFB. For details, refer to the following.

☞ Page 589 HOW TO CONTROL FROM CPU MODULE

When using the MCFB in the motion system, the I/O No. needs not to be specified. (The specification is ignored.)

Precautions

- I/O arguments of the MCFB are refreshed at a call timing of the FB, but the cycle that the control is actually executed differs depending on the FB. For example, the motion FB performs the control at the operation cycle regardless of the FB call task (normal/fixed cycle). For details, refer to specifications of each FB.
- The FB imports input values when turning TRUE the execution command (Execute and Enable). Therefore, set input values before turning TRUE the execution command (Execute and Enable). When changing multiple input arguments by a retrigger/continuous update of the FB, change them by the same task with the FB call task to match the import timing of each argument.
- After turning TRUE the execution command (Execute) of the FB, in order to turn FALSE the execution command (Execute), turn it FALSE after the executing (Busy) turned into the TRUE status.
- The motion control function uses the real number data (floating-point data) for the axis current position and the command target position, etc. Therefore, it may include the truncation error. For details about the real number data used for MCFB, refer to "Data Specification Method" in the following manual.

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

  For details about the truncation error by real number data processing in the inside motion control function, refer to the following.

  ☞ Page 83 Positioning Range

Program examples

For details, refer to specifications of each FB.

FUTURE SUPPORT PLANNED

The following model and functions are mentioned in this manual, but it is planned for a future support.

The information in this page might be changed for improvement without prior notice.

Function	Description
Axis assignment	Setting of slave object
Operation cycle	Auto refresh link devices of motion syncronization stations
Send/Receive data refresh of network connected devices	Standard station
Stop	The following setting values of AxesGroupName.Pr.StopMode_ErrorInGroup (Structuring axis operation selection at axis stop by factors) • "4: ServoOffAfterImmediateStop" • "5: ServoOffAfterDecelStop" The following setting values of AxisName.Pr.StopMode_DecelerationCurve/ AxesGroupName.Pr.StopMode_DecelerationCurve (Stop selection at deceleration to stop) • "0: RapidCurve" • "2: ContinueCurve"
Current position change function	MCv_SetPositionTriggered (Trigger current position change) Set position change at control execution during axis operation
Acceleration/deceleration processing function	Acceleration/deceleration time change
External signal selection Touch probe	Data type [OBJ]
Slave emulate	AxisName.Cd.SlaveEmulate_Param (Slave Emulation Parameter) AxisName.Cd.SlaveEmulate_Request (Slave Emulation Request)
Operation profile data	MCv_ProfileSelect (Profile data selection) MCv_ProfileCalc (Profile operation) MCv_ChangeCycleTriggered (Trigger change of current value per cycle) Multidimensional profile Operation profile data control
Cam operation	"2: mcRelative" of StartMode MasterStartDistance "1: TRUE" of MasterAbsolute
Motion system software installation	Motion software package for the RD78G(H) (including files) • Basic system software simulator DLL • Data definition file (axis data, etc.) • Add-on simulator DLL • The engineering tool window definition file for add-on • Boot software binary for network • Bood software simulator DLL for network
Safety communication	All safety communication functions

MEMO

PART 1 BASIC SPECIFICATIONS

1 AXIS SETTING

2 AXIS MANAGEMENT FUNCTIONS

3 BASIC FUNCTIONS

1 AXIS SETTING

1.1 Axis

The target controlled in the motion system is called an axis. Axes are classified into real axes which target the drive units and the I/O devices connected to the network and virtual axes which virtually generate commands and position in the motion system.

Classification	Axis type	Description
Real axis	Real drive axis	Uses the drive unit corresponding to the CiA402 drive profile connected to CC-Link IE TSN. Included in the number of controlled axes.
	Real encoder axis	Generates the current position from the synchronous encoder output pulse connected to the drive unit on the CC-Link IE TSN.
Virtual axis	Virtual drive axis	Virtually generates commands in the motion system. The real drive unit is not used.
	Virtual encoder axis	Generates the current position from the motion system variables. Used as an input axis for the single axis synchronous control.
	Virtual linked axis	Connects FBs of the single axis synchronous control. Only the minimum data required in the single axis synchronous control is defined.

Relevant variables

Variable/Structure name	Name	Details
AxisName.Md.		
AxisStatus	Axis status	Displays the current axis status. -1: Invalid (Axis variable uninitialization/axis parameter error) 0: Disabled (Axis disabled) 1: ErrorStop (Error stop) 2: Stopping (Deceleration stop) 3: Homing (Homing) 4: Standstill (Standby) 5: DiscreteMotion (Positioning operation) 6: ContinuousMotion (Continuous operation) 7: SynchronizedMotion (Synchronous operation)

Maximum number of controlled axes

The maximum number of controlled axes controlled by the motion system is the number of real drive axes. The other type of axes are not counted in the number of axes. For details, refer to the following.

Axis type	RD78G4	RD78G8	RD78G16	RD78G32	RD78G64	RD78GHV	RD78GHW
Real drive axis*1	4 axes	8 axes	16 axes	32 axes	64 axes	128 axes	256 axes
Virtual drive axis	Up to 10000 axes can be set. Settable number of axis varies depending on the system memory capacity setting.*2 Axis assigment method is as follows. • Set on the Engineering tool (Axis setting window)*3						
Virtual linked axis							
Real encoder axis							
Virtual encoder axis							

*1 When a multi-axis drive unit and a general output device are used as multiple axes, the number of those axes is counted.
Ex: The 2-axis drive unit is counted as 2 axes.

*2 For the memory capacity, refer to the following.
☞ Page 59 Memory usage

*3 Up to total sum of 1024 axes can be set on the engineering tool (Axis setting window)

If the axis setting exceeds the maximum number of controlled axes, the warning "Maximum Setting Axis Number Over Warning" (warning code: 0F0BH) is output.

The axis is used as the control axis sequentially which the axis is assigned as the axis variable to the global label and the axis which exceeds the maximum number of the control axis turns the axis status (AxisName.Md.AxisStatus) to "-1: Invalid" and it can not be used for the control.

Required settings for axes

The following items are required to be set in the axis setting window of the engineering tool to set the axes. For axis setting methods, refer to the following.

☞ Page 56 Axis Assignment

Item	Description
Axis name	Sets an arbitrary axis name.
Axis No.	Sets a control axis No. of the motion system.
Axis type	Sets the axis type.
Station address	Sets the station address of a driver device related to the axis. In case of a multi-axis driver, sets the multidrop No. as well.
Absolute position control setting	Sets the absolute position control method of axis.
Control cycle	Sets the cycle to execute control.

Other than above items, additional parameters must be set for each axis type.
For details on parameter settings, refer to the specifications of each axis type.

Axis variables

An axis is generated/initialized by the axis setting in the engineering tool. The set axis is assigned to the global label data as an axis variable.
The axis is defined as an axis variable configured with monitor information axes such as the parameter information, the current position, and the status, etc.
An axis variable data type changes depending on the axis type.
For the setting of the engineering tool, refer to the following.

Navigation window ⇨ "Parameter" ⇨ "Module information" ⇨ Target module ⇨ "Module extended parameter" ⇨ [Motion Control Setting Function]

Navigation window ⇨ "Axis" ⇨ Right-click ⇨ [Add New Data]

Axis variable name (default)

The following names are assigned based on the setting axis No. These can be changed to an arbitrary name. For details, refer to the following.

☞ Page 56 Axis Assignment

Setting axis No.	Engineering tool default axis name
1	Axis0001
2	Axis0002
⋮	⋮
10000	Axis10000

Data type

The following data types are assigned based on the axis type. For details on data type, refer to the relevant variables of each 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 linked axis	AXIS_VIRTUAL_LINK

Each data type has the following members respectively.

Member name	Data type	Description
AxisRef	AXIS_REF	The data structure for input/output of the motion control FBs. The type is fixed regardless of the axis type.
PrConst	Differs by the axis type.	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.
Pr		Stores the axis parameter data. Opens the default value at the axis variable initialization. The fetch timing to the control changes depending on the parameter.
Md		Stores the axis monitor data. Executes the refresh in the fixed cycle for each monitor data.
Cd		Stores the axis control command data. Acquires the latest value every control operation cycle and uses it for the control.

Axis variable initialization timing

Axis variables are initialized at the following timing.

Timing	Processing
Power ON/CPU module reset	Refers to the global label data, and initializes all set axis variables.
PLC READY [Y0] OFF→ON	• Uninitialized axes Refers to the global label data and initializes all axes variables. • Initialized axes Refer to the global label data for the axis parameter data and reload again. When the parameter error occurs at loading, the axis is not deleted. In this case, the ready [X0] does not turn to ON. For the label initialization processing when the PLC READY [Y0] turns OFF to ON, refer to "Label initialization function" in the following manual. MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

In case of a real axis, the device network connection is required to operate an axis after the axis variable is initialized. If the device of the station address has been connected, it must be disconnected and reconnected. (The axes can be emulated without network connection. (☞ Page 435 Slave Emulation))

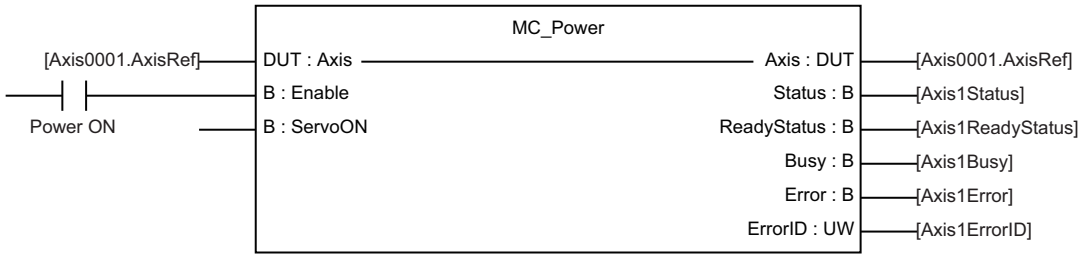
How to specify in a user program

When "AXIS_REF" is described in the input/output setting of the motion control FB, the axis can be specified by setting AxisRef (AxisName.AxisRef) which is the AXIS_REF type member of each axis variable.

Ex.

When executing the MC_Power with the following conditions:

axis type: real drive axis, axis No.: 1, axis name: Axis0001



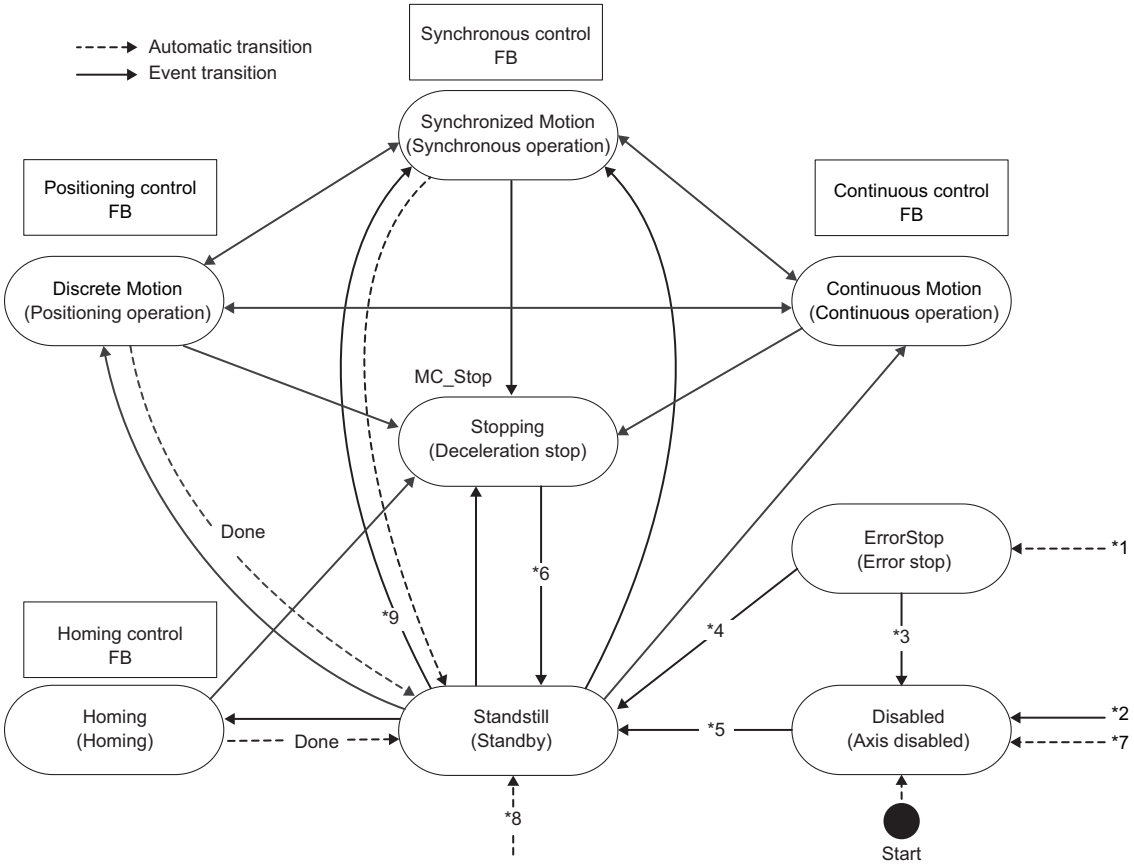
State transition

The following shows the status that a single axis can be in.

The current status can be referred in axis status (AxisName.Md.AxisStatus).

Status	Description
0: Disabled	The status where the axis cannot be used. In case of a real drive axis, this status indicates the servo OFF stop. In case of a real axis, this status occurs also when the slave device is disconnected from the network.
1: ErrorStop	The status where the deceleration stop or stop by error occurrence. In case of a real drive axis, the axis is servo ON or servo OFF depending on the error.
2: Stopping	The status where the deceleration stop has been executed by MC_Stop. After the axis stop is completed, this status continues until the Execute of MC_Stop changes to FALSE. The MCFB cannot be executed in this status.
3: Homing	The status where the homing has been executed in the driver or the motion system by the homing FB.
4: Standstill	The status where the MCFB can be used. In case of a real drive axis, the axis is servo ON stop.
5: DiscreteMotion	The status where the positioning operation has been executed to the target position by the positioning control FB.
6: ContinuousMotion	The status where the operation is not stopped after the processing is completed in the continuous control FB (such as the FB of speed control and torque control).
7: SynchronizedMotion	The status where the axis synchronizes with the master axis by the single synchronous control FB. Transits to this status after the axes group set as the configuration axes changes to the "5: GroupMoving" status.
-1: Invalid	Axes cannot be used in this status due to axis variable uninitialization (parameter error occurrence at an initialization, etc.).

The status transits as shown below according to the start of the MCFB. The state transition may varies depending on the axis type. Refer to the specifications of each axis type.



*1 From any state, transits when an error in the axis occurred.

*2 From any state, transits when MC_Power.Enable = FALSE and there is no error in the axis.
Transits when Enable of the MC_Power = TRUE, ServoON = FALSE, and no error occurs in the axis from Standstill state.
Transits when AxisName.Pr.StopMode_ServoOff is "0: Ignore" and turns Stopping status to ServoON of MC_Power = FALSE.
However, the axis status does not transit until the axis is stopped completely. After that, the status transits to "0: Disabled". For details, refer to the following.
☞ Page 162 Stop

*3 Transits by the status error cancel for the error reset command during the servo OFF state.

*4 Transits by the status error cancel for the error reset command during the servo ON state.

*5 Transits when MC_Power.Enable = TRUE, ServoON = TRUE, and MC_Power.Status = TRUE.

*6 Transits when MC_Stop.Done = TRUE and MC_Stop.Execute = FALSE.

*7 Transits when the forced servo OFF is enabled by the driver from Standstill state.
Transits when a communication with the driver fails from Standstill state.

*8 Transits automatically when the virtual axis is enabled.

*9 Transits when the axes group used as configuration axes turns to the GroupMoving state.

For the classification of the motion FBs, refer to the following.

☞ Page 721 List of FBs

Precautions

The error "Out of Parameter Range (Axis)" (error code: 1D80H) occurs if a parameter error occurs at the axis variable initialization. At this time, the axis status (AxisName.Md.AxisStatus) will be "-1: Invalid". The monitor data, etc. of the axis where "-1: Invalid" occurred is not refreshed. If the axis is specified by the user program, the error "Out of Axis No. Range" (error code: 3400H) occurs.

1.2 Axis Type

Relevant variables/FB

Real drive axis data type AXIS_REAL members (excluding AxisRef)

Variable/Structure name	Name	Details
AxisName.PrConst.		
AxisType	Axis type setting	☞ Page 56 Axis Assignment
AddressOfStation	Station address setting	☞ Page 56 Axis Assignment
PosRestoration_AbsPosBase	Absolute position reference setting	☞ Page 57 Absolute position control setting
OperationCycle	Control cycle setting	☞ Page 56 Axis Assignment
SlaveObject	Slave object data	☞ Page 56 Axis Assignment
AxisName.Md.		
AxisStatus	Axis status	Displays the current axis status. -1: Invalid (Axis variable uninitialization/axis parameter error) 0: Disabled (Axis disabled) 1: ErrorStop (Error stop) 2: Stopping (Deceleration stop) 3: Homing (Homing) 4: Standstill (Standby) 5: DiscreteMotion (Positioning operation) 6: ContinuousMotion (Continuous operation) 7: SynchronizedMotion (Synchronized operation)
UseInGroup	Using axes group	Displays whether the drive unit is used in axes group or not. FALSE: Not used TRUE: Used
Io_PosActualValue	Object data_PosActualValue	Indicates a value of object data_PosActualValue. ☞ Page 56 Axis Assignment
Io_Statusword	Object data_Statusword	Indicates a value of object data_Statusword. ☞ Page 56 Axis Assignment
Io_TargetPos	Object data_TargetPos	Indicates a value of object data_TargetPos. ☞ Page 56 Axis Assignment

Real encoder axis data type AXIS_ENCODER members (excluding AxisRef)

Variable/Structure name	Name	Details
AxisName.PrConst.		
AxisType	Axis type setting	☞ Page 56 Axis Assignment
AddressOfStation	Station address setting	Via the drive unit: The same station address as the real drive axis that the encoder is connected.
Encoder_AxisType	Real encoder axis type setting	Sets the type of the real encoder axis. 1: Via drive unit
Encoder_CounterDisableSignal	Counter disabling signal	☞ Page 95 Follow Up
AxisName.Md.		
AxisStatus	Axis status	Displays the current axis status. 0: Disabled (Axis disabled) 1: ErrorStop (Error stop) 2: Stopping (Deceleration stop) 3: Homing (Homing) 4: Standstill (Standby)
Encoder_CounterDisable	Counter disabled	Turns TRUE while disabling the input from the encoder. FALSE: Counter Enabled TRUE: Counter Disabled
Io_Statusword	Object data_Statusword	Displays the synchronous encoder status emulated by the motion system.
Io_PosActualValue	Object data_PosActualValue	Displays the input pulse value [encoder pulse unit] acquired from the synchronous encoder.
Io_PosEncoderResolution	Object data_PosEncoderResolution	Displays the synchronous encoder resolution. Reads the value from the setting location when the drive unit is connected.

Virtual drive axis data type AXIS_VIRTUAL members (excluding AxisRef)

Variable/Structure name	Name	Details
AxisName.PrConst.		
AxisType	Axis type setting	☞ Page 56 Axis Assignment
PosRestoration_AbsPosBase	Absolute position reference setting	☞ Page 57 Absolute position control setting
OperationCycle	Control cycle setting	☞ Page 56 Axis Assignment
AxisName.Md.		
AxisStatus	Axis status	Displays the current axis status. -1: Invalid (Axis variable uninitialization/axis parameter error) 0: Disabled (Axis disabled) 1: ErrorStop (Error stop) 2: Stopping (Deceleration stop) 3: Homing (Homing) 4: Standstill (Standby) 5: DiscreteMotion (Positioning operation) 6: ContinuousMotion (Continuous operation) 7: SynchronizedMotion (Synchronized operation)
Io_TargetPos	Object data_TargetPos	Displays a value which is converted into the command value in a driver unit from the cumulative current position.

Virtual encoder axis data type AXIS_VIRTUAL_ENCODER members (excluding AxisRef)

Variable/Structure name	Name	Details
AxisName.PrConst.		
AxisType	Axis type setting	☞ Page 56 Axis Assignment
Encoder_CounterDisableSignal	Counter disabling signal	☞ Page 95 Follow Up
Encoder_RingCout_LowerValue	Encoder ring counter lower limit value	Sets the encoder ring counter lower limit value. • one-word PosActualValue: -32768 to 32767*1 • two-word PosActualValue or omits PosActualValue: -2147483648 to 2147483647
Encoder_RingCout_UpperValue	Encoder ring counter upper limit value	Sets the encoder ring counter upper limit value. • one-word PosActualValue: -32768 to 32767*1 • two-word PosActualValue or omits PosActualValue: -2147483648 to 2147483647
SlaveObject	Slave object data	Sets the slave object data
PosActualValue	PosActualValue	Sets the data storing synchronous encoder input. • one-word specified: -32768 to 32767 • two-word specifed: -2147483648 to 2147483647
AxisName.Md.		
AxisStatus	Axis status	Displays the current axis status. 0: Disabled (Axis disabled) 1: ErrorStop (Error stop) 2: Stopping (Deceleration stop) 3: Homing (Homing) 4: Standstill (Standby)
Encoder_Connected	Connection status	Displays the virtual encoder axis connection status. FALSE: Not connected TRUE: Connecting
Encoder_CounterDisable	Counter disabled	Turns TRUE while disabling the input from the encoder. FALSE: Counter Enabled TRUE: Counter Disabled
Io_PosActualValue	Object data_PosActualValue	Displays the value [encoder pulse unit] acquired from the encoder input.
Io_PosEncoderResolution	Object data_PosEncoderResolution	Displays the synchronous encoder resolution.
AxisName.Cd.		
Encoder_Connect	Connection command	The signal for enabling the connection status of the virtual encoder axis input. FALSE: Invalid TRUE: Valid
Encoder_InputValue	Encoder input value	Sets the value sequentially used as the virtual encoder axis input value.

*1 When the out of one-word range value is set, the error "Out of Parameter Range (Axis)" (error code: 1D80H) occurs.

Virtual linked axis data type AXIS_VIRTUAL_LINK members (excluding AxisRef)

Variable/Structure name		Name	Details
AxisName.PrConst.			
	AxisType	Axis type setting	☞ Page 56 Axis Assignment
	PosRestoration_AbsPosBase	Absolute position reference setting	☞ Page 57 Absolute position control setting
	OperationCycle	Control cycle setting	☞ Page 56 Axis Assignment
AxisName.Md.			
	AxisStatus	Axis status	Displays the current axis status. -1: Invalid (Axis variable uninitialization/axis parameter error) 0: Disabled (Axis disabled) 1: ErrorStop (Error stop) 2: Stopping (Deceleration stop) 3: Homing (Homing) 4: Standstill (Standby) 5: DiscreteMotion (Positioning operation) 6: ContinuousMotion (Continuous operation) 7: SynchronizedMotion (Synchronized operation)

Real drive axis

This axis uses a drive unit which is compatible with the CiA402 drive profile connected to CC-Link IE TSN. It is counted as the number of control axes.

Driver control mode transition

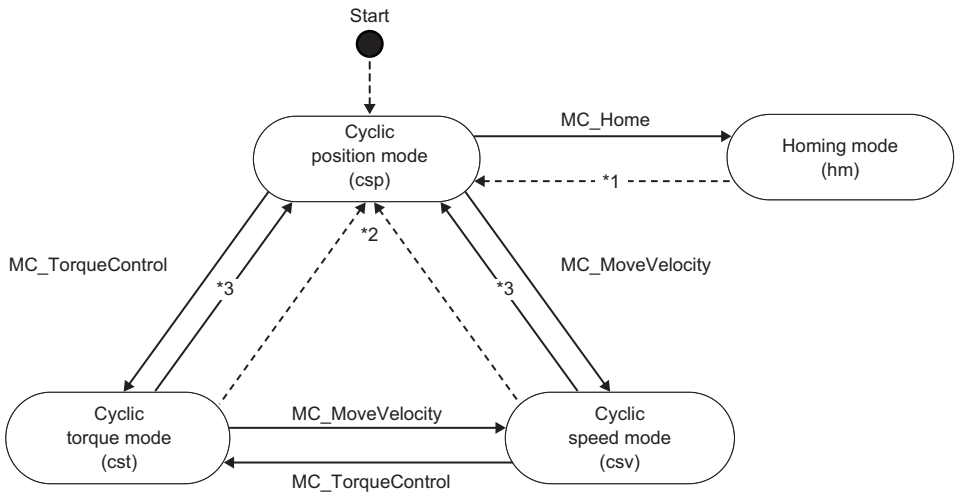
A driver device connected as a real drive axis follows the CiA402 drive protocol, and changes the control mode according to the "Modes of operation" object output from the motion system. The driver inputs the current mode as the "Modes of operation display" object in the motion system and displays it in AxisName.Md.Driver_Mode.

The motion system supports the operations of below modes.

Supported control mode		Description	Reference
Homing mode (hm)	6: hm	Executes homing operation by a driver.	Page 202 Driver homing method
Cyclic position mode (csp)	8: csp	Executes control following the sequential positioning command from the controller in each communication cycle.	Page 275 DIRECT CONTROL
Cyclic velocity mode (csv)	9: csv	Executes control following the velocity command from the controller in each communication cycle.	
Cyclic torque mode (cst)	10: cst	Executes control following the torque command from the controller in each communication cycle.	

The control mode of the driver must be csp when the driver is connected.

Control mode switching of the driver is executed at the same time when MCFB is executed. The state transition is shown below.



*1 Transits after the axis is stopped by homing completion or error occurrence.

*2 Transits at stop completion or error occurrence.

*3 Transits when a command other than MC_MoveVelocity/MC_TorqueControl is being Aborting/Buffered.

Precautions

If the axis parameter has an error when the controller power is turned ON with the driver being connected, the driver is connected as the remote I/O although the axis which has an error is not generated.

In this state, the axis is not generated even if the axis parameter is reviewed or the PLC READY [Y0] is turned OFF to ON again.

For generating the axis, power on the controller and the driver again.

Real encoder axis

This axis generates the current position from the synchronous encoder output pulse connected to the drive unit on CC-Link IE TSN. This will be used at the "single axis synchronous control".

This axis can control the synchronous encoder shown below as the real encoder axis. For required setting items for each real encoder axis specification, refer to the following.

☞ Page 45 Parameter setting items

Real encoder axis type	Description
Via drive unit	Uses the synchronous encoder connected to the drive unit (such as MR-J5(W)-G) corresponding to the scale measurement function as the real encoder axis.

Parameter setting items

Depending on the real encoder axis type, the following items are required to set up in order to use the axis.

○: Necessary, △: Optional, —: Unnecessary, □: Necessary for the same station address real drive axis

Setting item		Real encoder axis type	Reference
		Via drive unit	
Axis No.		○	Page 56 Axis No.
Axis type		○	Page 56 Axis type
Real encoder axis type setting		○	Page 46 Real encoder axis type setting
Station address		○	Page 46 Station address Page 56 Station address
Absolute position control setting		○	Page 57 Absolute position control setting
Control cycle		○	Page 57 Control cycle
Slave object setting			Page 46 Slave object setting Page 58 Slave object settings
	PosActualValue	—	
	Encoder status 2	□	
	Scale ABS counter	□	
	Scale cycle counter	□	
	Scale measurement encoder Resolution	□	
	Scale measurement encoder reception status	□	
Driver unit conversion numerator		○	Page 381 Compensation Function
Driver unit conversion denominator		○	
Position command unit setting		○	Page 79 Technical Units
Velocity command unit setting		○	
Command unit string		○	
Ring counter enabled selection		○	Page 83 Positioning Range
Ring counter upper limit value		△	
Ring counter lower limit value		△	
Counter disabling signal		△	Page 95 Follow Up

Real encoder axis type setting

Sets the synchronous encoder type which will be the origin of generating the real encoder axis input value.

Setting value	Description
1: Via drive unit	Uses the synchronous encoder input value via drive unit connected to the specified drive unit as the synchronous encoder input.

In order to use via drive unit type, set the real drive axis of the same station address and assign the required objects listed below.

☞ Page 46 Slave object setting

For details on the real drive axis setting method, refer to the following.

☞ Page 56 Axis Assignment

Station address

Depending on the real encoder axis type, set the following.

Real encoder axis type	Applicable hardware and software to setup as station address
Via drive unit	Drive unit connected with the synchronous encoder

For details on the setting value, refer to the following.

☞ Page 56 Axis Assignment

Slave object setting

Depending on the real encoder axis type, set the following.

Real encoder axis type	Setting item	Setting required object	Description*1
Via drive unit	Real drive axis of the same station address	Encoder status 2	• Specify the object ID storing the driver scale measurement function setting. bit0: Scale measurement function ABS system correspondent bit1: Scale measurement function (Valid when TRUE) • Reads the value from the setting location when connecting to the driver. When bit1 is FALSE, the error "Drive Module Via Synchronous Encoder Setting Incorrect" (error code: 1A90H) occurs.
		Scale ABS counter*2	• Specify the object ID storing the current synchronous encoder multiple revolution counter. • Setting range is handled as -32768 to 32767.
		Scale cycle counter*2	• Specify the object ID storing the current synchronous encoder position within one revolution. • Setting range is 0 to the range specified by the scale measurement encoder resolution.
		Scale measurement encoder Resolution	• Specify the object ID storing the synchronous encoder resolution. • The synchronous encoder with setting value 0 cannot be connected.
		Scale measurement encoder reception status*2	• Specify the object ID storing the alarm information of the scale measurement encoder. bit0: CPU alarm bit1: LED alarm bit2: Data alarm bit5: Multi-revolution counter alarm bit6: ABS erased alarm (When TRUE, corresponding alarm occurs)

*1 For details on the setting value, refer to the following.
☞ Page 420 TARGET_REF structure

*2 If all of the slave objects are not assigned, the error "Drive Module Via Synchronous Encoder Setting Incorrect" (error code: 1A90H) occurs.

Required functions to use

The required functions to use the real encoder axis are shown below.

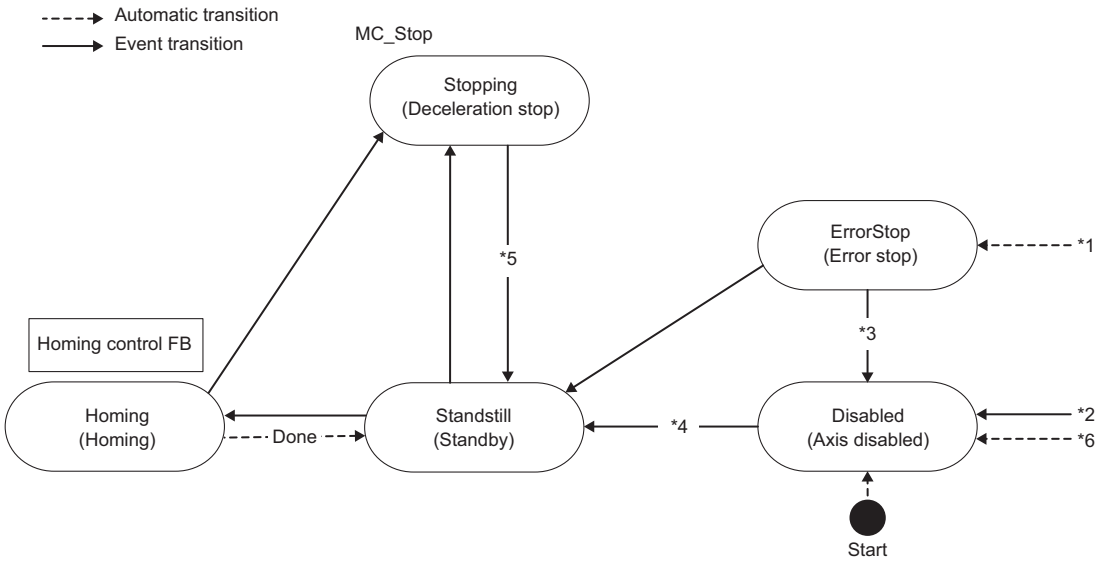
Function		Description	Axis type specific detailed information
Axis assignment		☞ Page 56 Axis Assignment	—
Technical units		☞ Page 79 Technical Units	—
Compensation function	Driver unit conversion function	☞ Page 381 Compensation Function	—
Absolute position control		☞ Page 98 Absolute Position Control	—
Operation cycle		☞ Page 109 Operation Cycle	—
State transition	Single axis state transition	☞ Page 52 Single axis state transition	—
Follow up (counter enabled/counter disabled)		☞ Page 95 Follow Up	—
Servo ON/OFF		☞ Page 88 Servo ON/OFF	—
Ring counter setting		☞ Page 83 Positioning Range	—
Control change function	Current position change function	☞ Page 304 Current Position Change Function	☞ Page 48 Control change function
Command filter	Smoothing filter	☞ Page 384 Command Filter	—
	Movement direction control filter		—

For other functions, refer to each function and check the required slave object for the function. If the axis holds the required slave object, the function is valid.

State transitions

■Single axis state transition

The real encoder axis (via drive unit) transits to the status shown below.



*1 From any state, an error in the axis occurrs.

*2 From any state, Enable of MC_Power/MCv_AllPoweris FALSE and there is no error in the axis.
Enable of MC_Power/MCv_AllPoweris TRUE, ServoON is FALSE and there is no error in the axis from Standstill state.

*3 Transits by the status error cancel for the error reset command during the servo OFF state. (Since no follow up will be done in ErrorStop state, it will be in the current position restoration not executed state when transits state is disabled.)

*4 After the real drive axis of the same station address is connected, it transits when Enable of MC_Power/MCv_AllPower is TRUE, ServoON is TRUE and Status of MC_Poweris TRUE.

*5 Transits when Done of MC_Stop is TRUE and Execute of MC_Stop.Execute is FALSE.

*6 From any state, the real drive axis of the same station address is not connected.

Control change function

When starting the current position change control unconditionally, use MC_SetPosition.

Point

To use the control change function by the real encoder axis, the axis status (AxisName.Md.AxisStatus) of the real encoder axis is required to be set as [4: Standstill] by Servo ON/OFF function.

Setting examples

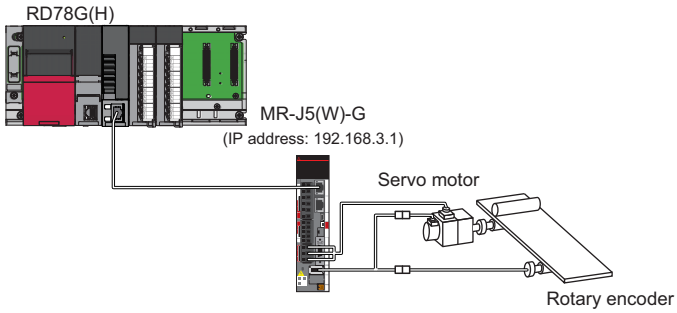
The setting example of the real encoder axis is shown below.

[Via drive unit]

Depending on the connected drive unit or version, available functions and encoders are restricted. For details, refer to the drive unit manuals.

Ex.

To use the real encoder axis1 (MR-J5(W)-G station address 192.168.3.1) scale measurement device as the input of the real encoder axis 2.



■Axis setting

Setting item	Axis setting	
	Real drive axis setting	Real encoder axis setting
Axis No.	1	2
Axis type	0: DriveAxis	2: EncoderAxis
Real encoder axis type setting	Unnecessary	1: Via drive unit
Station address	192.168.3.1	192.168.3.1 (Specify the station address of the drive unit that the synchronous encoder is connected)
Absolute position control setting	-1: Auto	Arbitrary*1
Control cycle	0	0*2

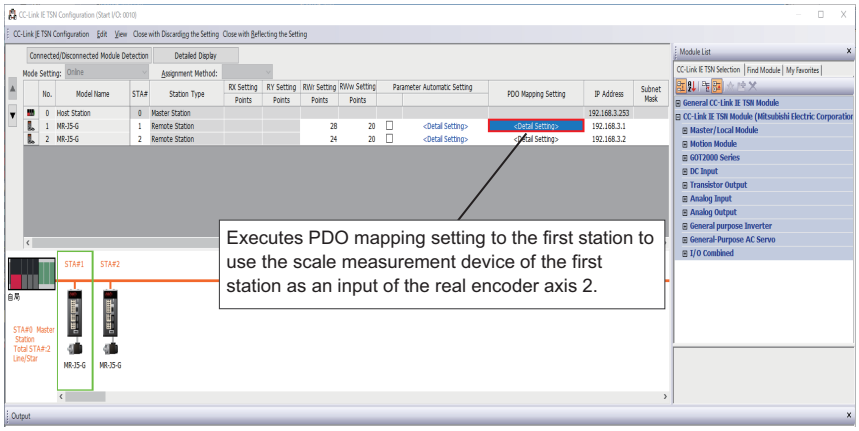
*1 Set the setting that matches with the setting of the synchronous encoder and driver side to be connected.When "-1: Auto" is set, acquires the absolute position system setting from the driver side.

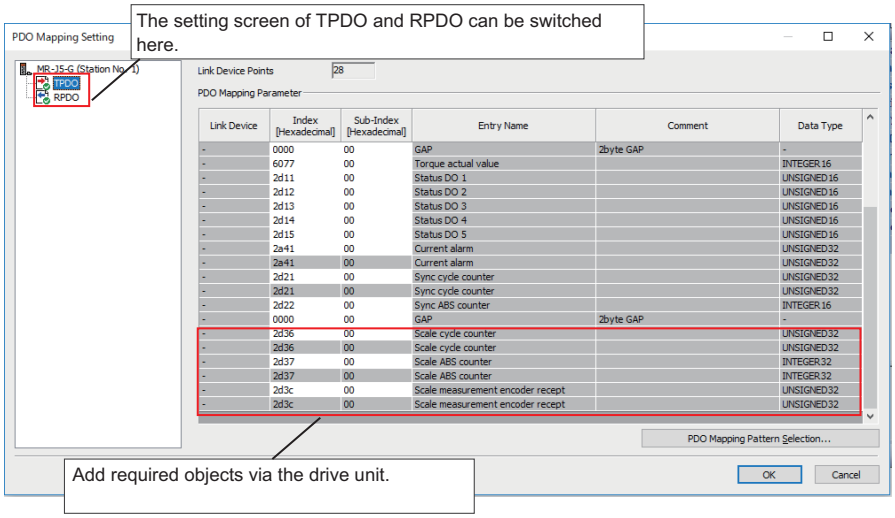
*2 The data refresh cycle of the synchronous encoder is the control cycle of the connected real drive axis and the axis control may not be executed by each data refresh cycle when it differs with the real encoder axis setting. Thus, the setting that the real encoder axis control cycle setting is to be the same cycle setting as the connected real drive axis is recommended.

■PDO mapping setting

Regarding the network configuration setting of GX Works3, set all three objects of the folllowing to the PDO mapping (TxPDO) of the real drive axis 1 (MR-J5(W)-G station address 192.168.3.1).

- Scale cycle counter
- Scale ABS counter
- Scale measurement encoder reception status





■Driver side (MR-J5(W)-G) setting

To enable the scale measurement function on MR-J5(W)-G, set the parameter below. (For MR-J5W-G, only MR-J5W2-G is supported)

For details on the parameter and connection method of the encoder, refer to the driver device manuals.

No.	Abbreviation	Name	Overview
PA22.3*1	**PCS	Scale measurement function selection	Select scale measurement function. Initial setting: 0 (Invalid) "1" (Used in absolute position detect system) "2" (Used in incremental system)

*1 Applicable to MR-J5(W)-G (A4 version) or later

When the applicable real drive axis is not connected,the real encoder axis connection becomes invalid. (It becomes valid by connecting the applicable real drive axis.)

Virtual drive axis

This axis can generate commands virtually in the motion system. A real drive unit is not used. The available commands/functions are partly different from those of the real axis.

Point

Since the virtual drive axis is specialized for generating the position command and it does not simulate a situation that the drive unit is connected, available functions are restricted. For simulating the operation close to the state when the drive unit is connected, use the slave emulation function to the real axis drive. For details, refer to the following.

☞ Page 435 Slave Emulation

Virtual encoder axis

Generates the current position from the values of the motion system variables. Used as an input axis for the single axis synchronous control.

Parameter setting items

The following items are required to set up in order to use the virtual encoder axis.

○: Necessary, △: Optional, —: Unnecessary

Setting item	Virtual encoder axis	Reference
Axis No.	○	Page 56 Axis No.
Axis type	○	Page 56 Axis type
Station address	—	—
Absolute position control setting	○	Page 57 Absolute position control setting
Control cycle	○	Page 57 Control cycle
Slave object setting		
PosActualValue	△	Page 51 Slave object setting
Driver unit conversion numerator	○	Page 381 Compensation Function
Driver unit conversion denominator	○	
Position command unit setting	○	Page 79 Technical Units
Velocity command unit setting	○	
Command unit string	○	
Ring counter enabled selection	○	Page 83 Positioning Range
Ring counter upper limit value	△	
Ring counter lower limit value	△	
Encoder ring counter upper limit value	○	Page 51 Encoder ring counter upper limit value/lower limit value
Encoder ring counter lower limit value	○	
Counter disabling signal	△	Page 95 Follow Up

Slave object setting

Set the following.

Object required setting	Description
PosActualValue	Set the string format data used as the encoder input value.*1 The settable types are only [VAR], [DEV] and [CONST]. If the setting value is not within the valid range, the error "Out of Parameter Range (Axis)" (error code: 1D80H) occurs. When [VAR] type is set, specify the data type to be (INT), (DINT), (WORD) or (DWORD) data. If the data type is other than the above, the error "Out of Parameter Range (Axis)" (error code: 1D80H) occurs. When [DEV] or [CONST] type is set, specify the data type to be (INT), (DINT), (WORD) or (DWORD). If the data type is not specified nor the data type is other than the above, the error "Out of Parameter Range (Axis)" (error code: 1D80H) occurs. When [VAR] type is used, do not specify the local label. Use "AxisName.Cd.Encoder_InputValue" as the encoder input value when omitted. Set the encoder input value sequentially by encoder pulse unit.

*1 For details on the setting value, refer to the following.
☞ Page 420 TARGET_REF structure

Encoder ring counter upper limit value/lower limit value

Sets the upper limit value and lower limit value of the encoder input value.

Set the encoder ring counter lower limit value and encoder ring counter upper limit value corresponds to the lower limit value/upper limit value of the encoder input value. If the range of the encoder input value and "the encoder ring counter lower limit value to the encoder ring counter upper limit value" differ, the encoder input value will not be read correctly.

If the encoder ring counter upper limit value = encoder ring counter lower limit value is set, the encoder input value is processed as the 32-bit counter of "-2147483648 to 2147483647" or as the 16-bit counter of "-32768 to 32767".

If the encoder ring counter lower limit value > the encoder ring counter upper limit value, the error "Out of Encoder Ring Counter Setting Range" (error code: 1AE1H) occurs.

Point

- For the encoder input value, set the cycle counter of "the encoder ring counter lower limit value ≤ the encoder input value ≤ the encoder ring counter upper limit value" as the input value. If the encoder input value is out of the above range, the encoder input value will not be read.
- Set the encoder input value that one operation cycle movement amount meets the following equation. If it does not meet, the actual input movement amount and the movement amount counted by the motion system may not match.
  One operation cycle movement amount <| the encoder ring counter upper limit value - the encoder ring counter lower limit value + 1| / 2
- If the encoder input value is out of range of "the encoder ring counter lower limit value to the encoder ring counter upper limit value" when it is connected to the encoder, the encoder current value will restore as below.
  - Absolute position system: Restore to the backup data value
  - Incremental system: Clamp by the encoder ring counter upper limit value/lower limit value

Required setting to use

The required functions to use the virtual encoder axis are shown below.

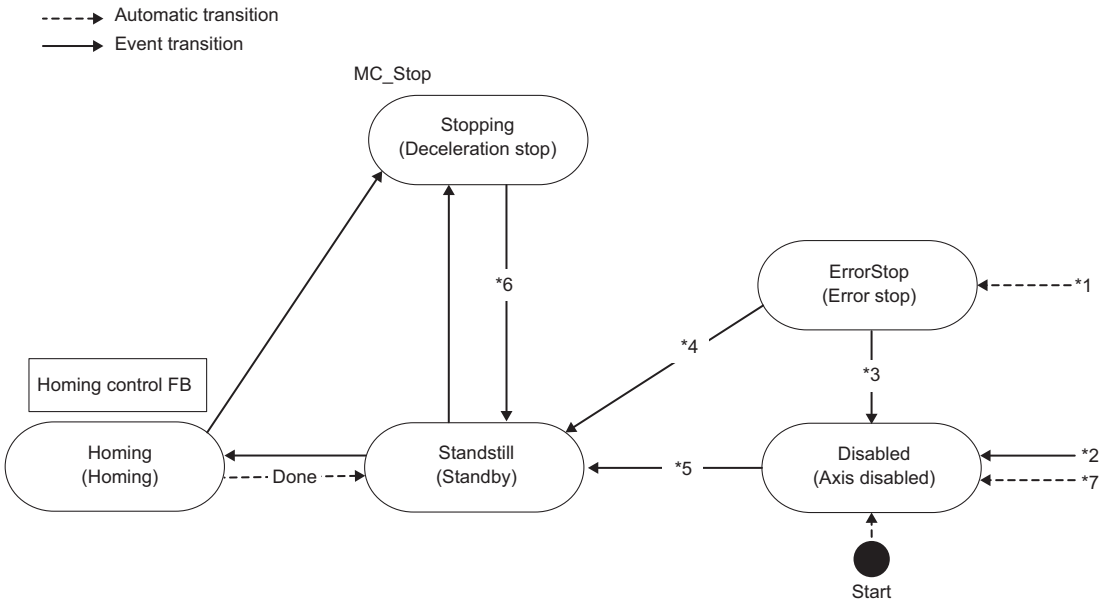
Function		Description	Axis type specific detailed information
Axis assignment		☞ Page 56 Axis Assignment	—
Technical units		☞ Page 79 Technical Units	—
Compensation function	Driver unit conversion function	☞ Page 381 Compensation Function	—
Absolute position control		☞ Page 98 Absolute Position Control	—
Operation cycle		☞ Page 109 Operation Cycle	—
State transition	Single axis state transition	☞ Page 52 Single axis state transition	☞ Page 47 State transitions
Follow up (counter enabled/counter disabled)		☞ Page 95 Follow Up	—
Ring counter setting		☞ Page 83 Positioning Range	—
Control change function	Current position change function	☞ Page 304 Current Position Change Function	☞ Page 48 Control change function
Command filter	Smoothing filter	☞ Page 384 Command Filter	—
	Movementdirection control filter		—

For other functions, refer to each function, check the required slave object for the funtion. If the axis holds the required slave object, the function is valid.

State transition

■Single axis state transition

The virtual encoder axis transits to the following state.



*1 From any state, an error in the axis occurs.

*2 From any state, Enable of MC_Power/MCv_AllPower is FALSE and there is no error in the axis.
Enable of MC_Power/MCv_AllPower is TRUE, ServoON is FALSE and there is no error in the axis from Standstill state.

*3 Transits by the status error cancel for the error reset command during the servo OFF state.

*4 Transits by the status error cancel for the error reset command during the servo ON state.

*5 After AxisNameCd.Encoder_Connect is TRUE, transits when Enable of MC_Power/MCv_AllPower is TRUE, ServoON is TRUE and Status of MC_Power is TRUE.

*6 Transits when Done of MC_Stop is TRUE and Execute of MC_Stop is FALSE.

*7 From any state, transits when AxisName.Cd.Encoder_Connect = FALSE.

Control change function

When starting the current position change control unconditionally, use the MC_SetPosition.

Restriction

The encoder input value will be read by every virtual encoder axis control cycle. When the encoder input value update cycle is slow, the velocity change increases. Use the smoothing filter to smooth the velocity change.

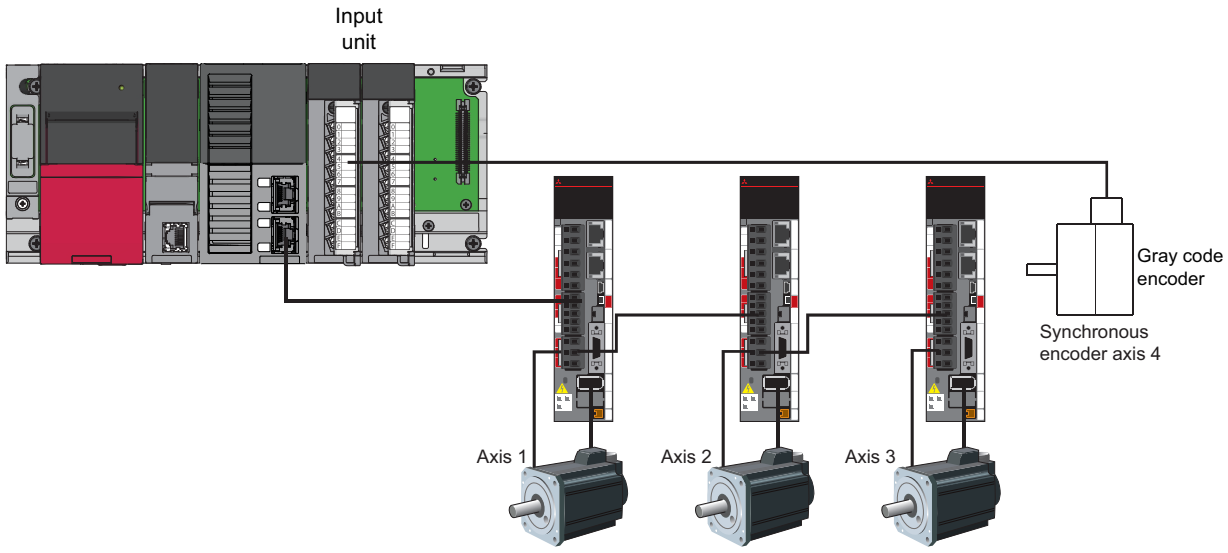
Setting example

The setting example of the virtual encoder axis and the method of use are shown below.

Ex.

When the axis 4 is used as the virtual encoder axis.

(Gray code encoder connected to the input unit: The resolution 4096 [pulse/rev] encoder value is used to control)



■Axis setting

Setting items	Setting value
AxisNo.	4
AxisType	4: VirtualEncoderAxis (Virtual encoder axis)
Absolute position control setting	0: ABSDisabled (Disable absolute position system)
Encoder ring counter upper limit value	4095
Encoder ring counter lower limit value	0
Object data_PosActualValue	[DEV](DINT)G11478000

1. Set AxisName.Cd.Encoder_Connect TRUE.
2. Check AxisName.Md.PosRestoration_Status is "2:RestoredInIncSystem", and AxisName.Md.Encoder_Connected is TRUE.
3. Set MC_Power/MCv_AllPower Enable and DriverON TRUE.
4. Check AxisName.Md.AxisStatus is "4: Standstill".
5. Read the encoder value of the gray code encoder by program, update the device setup on PosActualValue with slave object setting sequentially.
   (The virtual encoder axis will be controlled based on the change amount of PosActualValue.)

Virtual linked axis

This axis connects FBs of single axis synchronous control. Since the minimum data for FB connection is defined, the processing load is reduced compared to the FB connection using a virtual drive axis, etc.
Without axis assignment in the engineering tool, an axis can be temporarily generated during program execution by only declaring the instance of the AXIS_REF structure in the local label. In this case, the motion system discards the axis data generated temporarily when the PLC READY [Y0] is turned ON to OFF. (If axis assignment is executed in the engineering tool, the axis data will be maintained even after the PLC READY [Y0] is turned ON to OFF.)

How to generate an axis

Set it according to axis assignment. For the method, refer to the following.
☞ Page 56 Axis Assignment

Precautions

- Software stroke limit cannot be set in the virtual linked axis. When virtual linked axes are used to connect between each user program of single axis synchronous control, set the software stroke limit in the end axis of connected single axis synchronous control.
- When temporarily generated axis is started, it will take longer than the usual to start the axis because the axis variables are generated in the motion service processing.

1.3 Connectable Stations

This section describes stations that are can be connected to the network.

Maximum number of stations that can be connected

The following shows the maximum number of stations that can be connected to the network.

Item	RD78G4	RD78G8	RD78G16	RD78G32	RD78G64	RD78GHV	RD78GHW
Maximum number of stations that can be connected	120 stations*1*2						

*1 Even if the station is a multi-axis drive unit device (a device which can control two axes or more), the number of station is counted as one station when it is recognized as one station. For details, refer to each drive unit instruction manual.

*2 When slave emulate is carried out, the number of stations is counted according to the connected device setting.

Precaution

An error will not occur even if there is no station and axis in the network setting.

1.4 Axis Assignment

Axes are generated/initialized by setting axis settings on the axis setting window in the engineering tool. The axes set on the axis setting window are assigned to the global label as axis variables. For the setting method, refer to Help of the engineering tool.

Setting items

The following shows setting items according to the axis type.

○: Necessary, —: Unnecessary

Axis type	Axis name	Axis No.	Axis type	Station address	Absolute position control setting	Control cycle
Real drive axis	○	○	○	○	○	○
Real encoder axis	○	○	○	○	○	○
Virtual drive axis	○	○	○	—	○	○
Virtual encoder axis	○	○	○	—	○	○
Virtual linked axis	○	○	○	—	○	○

■Axis name

Set an arbitrarily axis name of the applicable axes within 127 characters.

An instance is generated as the set axis name.

■Axis No.

Axis No. is an identifier to control in the motion system.

If the same No. is set in the multiple axes, the error "Axis No. Setting Duplication Error" (error code: 1A91H) will occur.

Axis setting	Axis No.
User specification	1 to 10000
Vertual linked axis generated automatically	20001 to 30000

■Axis type

Specify an axis type of the axis.

Axis type	Setting value
Real drive axis	0: DriveAxis
Real encoder axis	2: EncodeAxis
Virtual drive axis	3: VirtualDriveAxis
Virtual encoder axis	4: VirtualEncoderAxis
Virtual linked axis	5: VirtualLinkAxis

The data type that an axis variable can take differs depending on the axis type.

For details on data types of each axis type, refer to the following.

☞ Page 36 Axis

■Station address

Specify the station address of the target device to connect a network device as a real axis. Also, specify the axis ID for connecting a device which can control multiple axes with one device (multiple axes device).

These settings are unnecessary to the virtual axis.

For the station address, specify the same format as "Target modification (character string starts with @)" of TARGET_REF structure.

For details on the TARGET_REF structure, refer to the following.

☞ Page 420 TARGET_REF structure

Even though the station address have not been set ("") in the real axis, by setting AxisName.PrConst.SlaveEmulate_Enable TRUE, it enables to use as the slave emulate function. If slave emulate function is not used, the error "Station Address Setting Incorrect" (error code: 1A95H) will be output and the axis will not be ready for operation.

■Absolute position control setting

Specify whether applicable axes are used as the absolute position system or not. For details, refer to the following.

☞ Page 98 Absolute Position Control

■Control cycle

Specify which operation cycle is used for the applicable axis to control. For details, refer to the following.

☞ Page 109 Operation Cycle

Relevant variables

AxisName in the following table indicates the instance name of the structure type for each axis type.

AXIS_REAL: Real drive axis

AXIS_ENCODER: Real encoder axis

AXIS_VIRTUAL: Virtual drive axis

AXIS_VIRTUAL_ENCODER: Virtual encoder axis

AXIS_VIRTUAL_LINK: Virtual linked axis

Variable/Structure name	Name	Details
AxisName.AxisRef.		
AxisNo	Axis No.	Sets the axis No. 0: Not set 1 to 10000: Setting axis No.
AxisName.PrConst.		
AxisType	Axis type setting	Sets the axis type. 0: DriveAxis (Real drive axis) 2: EncoderAxis (Real encoder axis) 3: VirtualDriveAxis (Virtual drive axis) 4: VirtualEncoderAxis (Virtual encoder axis) 5: VirtualLinkAxis (Virtual linked axis)
AddressOfStation	Station address setting	Sets network address of the slave station with a character string.
PosRestoration_AbsPosEnable	Absolute position control setting	Sets the absolute position control setting. 0: ABSDisabled (Disable absolute position system) 1: Enabled (Enable absolute position system) -1: Auto (Automatic setting) (Acquire from connected device)
OperationCycle	Control cycle setting	Sets Control cycle setting. For details, refer to the following. ☞ Page 109 Operation Cycle
AxisName.Md.		
AxisName	Axis name	Displays the axis name.
Io_PosActualValue	Object data_PosActualValue	Indicates a value of object data_PosActualValue.
Io_PosEncoderResolution	Object data_PosEncoderResolution	Indicates a value of object data_PosEncoderResolution.
Io_Statusword	Object data_Statusword	Indicates a value of object data_Statusword.
Io_TargetPos	Object data_TargetPos	Indicates a value of object data_TargetPos.

Slave object settings

In the motion system, the data that sends/receives information to/from the connected device in the motion operation processing is called a slave object.

With slave object settings, the motion system can be customized to adjust to the slave device by setting the data related to the slave device to the slave object.

For one slave object, set as the following.

Setting item	Setting value
Each slave object name	Specifies the target data to set. The following items can be specified. • No setting • Object INDEX Specified by INDEX, sub INDEX, and the number of bytes • Link device • Slave label • Buffer memory • Device • Label • Constant For details on the format to specify the data, refer to the following. ☞ Page 420 TARGET_REF structure

Settable data by the axis type and the slave object is shown below.

Axis type	Slave object attribute*1		Settable setting value in the slave object setting (other than no setting)*2	
	PDO mapping	Access	Setting value	Restriction for combination with other settings
Real drive axis	TPDO	RO	Object INDEX	Only objects registered in the mapping setting for input (the slave to the controller) in the "PDO mapping setting".
	RPDO	RW	Object INDEX	Only objects registered in the mapping setting for output (the controller to the slave) in the "PDO mapping setting".
	No	RO	Object INDEX	—
			Buffer memory device Internal address constant	—
		RO	Object INDEX	—
			Buffer memory device Internal address	—

*1 For details on slave object attribute, refer to the following.
☞ Page 60 Slave object list

*2 When the slave label is specified, refer based on data the applicable label is assigned to.

If unavailable data is set, the error "Mapping Communication Error" (error code: 1A5FH) will occur at the axis variable initialization.

There are slave objects whose settings are necessary depending on the axis type. A valid setting value other than "no setting" must be set to the applicable slave object. (If the setting is illegal, the error "Slave Object Setting Incorrect" (error code: 1A96H) will occur at the axis variable initialization.)

For slave objects whose settings are necessary, refer to the following.

☞ Page 60 Slave object list

When "no setting" is set to the slave object setting, the axis cannot use functions/commands that need the applicable slave object.

Data and information of the setting location are sent/received based on the data type of each slave object in the motion operation processing.

Access	Data size	Description of the motion operation processing
R: Slave → Motion system	(Slave object) < (Setting location data)	Converts the setting location data to data size of a slave object and imports it. Exceeded upper bit information is discarded.
	(Setting location data) ≤ (Slave object)	Converts the setting location data to data type of the slave object and imports it.
W: Motion system → Slave	(Setting location data) < (Slave object)	Stores a slave object value as it is to the the setting location data. After it is stored, shorted upper bit does not change before and after refreshing.
	(Slave object) ≤ (Setting location data)	Converts a value to data size of the setting location data and stores it. Exceeded upper bit information is discarded.

PDO mapping settings

Data (object) that sent/received between the controller and the slave in cyclic communication (PDO communication) must be mapped in advance in order to control each function in the real drive axis.
For details on PDO mapping settings, refer to "Cyclic Transmission" in the following manual.
MELSEC iQ-R Motion Module User's Manual (Network)
Mapping is required for some objects to be connected as real drive axes in the motion system. The following objects must be mapped. (If those are not mapped, the error "Necessary Slave Object Unset" (error code: 1AA8H) will occur.)

Mapping	Required object name
Output (RPDO)	Contorl word
	Target position
	Operation mode
Input (TPDO)	Status word
	Current position
	Operation mode display

Setting procedure

This section describes the procedure to newly set a real drive axis by using the engineering tool.

1. Add slave device settings to the network setting, and set IP address and PDO map information.
2. Select adding axis in the axis setting of the motion system, and set axis information.
3. Double-click the set axis and set the axis parameters.
4. Select the slave object setting, and assign slave objects.
5. Write the parameters in the motion system.

Memory usage

The memory usage differs depending on the axis types. Specify the memory capacity of the axis data according to the axis type and the number of the axis to be used.
The memory usage of each axis type is shown below and these indicate the usage of every axis.

Axis type	Memory usage (K bytes/axis)
Real drive axis	160
Real encoder axis	40
Virtual drive axis	60
Virtual encoder axis	40
Virtual linked axis	40

The memory capacity for axis data is specified by System.PrConst.Addon_Axis.RamSizeMax.
When the axis data memory usage exceeds the capacity at initializing the axis variable, the error "Axis Data Memory Size Over" (error code: 1A5DH) occurs and the axis is not generated. Check the memory capacity and execute the software reboot.
Check the memory capacity of the following add-on library as needed.

- MotionEngine
- NetworkDriver_CCIETSN (for using real drive axis)
- ServoDriver_CANopen (for using real drive axis)
- SignalIO (for using external signal)
- ExternalSignal (for using external signal)

Slave object list

The following shows attributes of slave objects.

Attribute	Description
Data Type	Indicates slave object size.
Access	Indicates whether the object can be read/written or not. • RO: Only read is available. • RW: Read and write are available.
PDO mapping	When the axis type is real drive axis, it indicates whether the mapping to cyclic communication is enabled or not. • No: Mapping to TxPDO and RxPDO is disabled. • TxPDO: Mapping to TxPDO is enabled. • RxPDO: Mapping to RxPDO is enabled. When the axis type is real encoder axis, read shown as below. • No: The motion system will access to the data unperiodically even if the link device is specified. • TxPDO: By specifying the link device inputted from the slave station to the master station, the motion system can read the data from the slave devices in fixed-cycle. • RxPDO: By specifying the link device outputted from the master station to the slave station, the motion system can write the data from the slave devices in fixed-cycle.
Default	Indicates the default value of the slave object for the real drive axis.

The slave object lists are shown below.

Slave object name	DataType	Access	PDO mapping	Default	Description
Supported drive modes	U32	ro	No	65020020	Gets the control mode supported by the driver device.
Modes of operation	I8	rw	RxPDO	60600008	Requires switching the control mode to the driver device.
Modes of operation display	I8	ro	TxPDO	60610008	Gets the control mode of the driver device.
Controlword	U16	rw	RxPDO	60400010	Requires switching the status to the driver device.
Control DI 1	U16	rw	RxPDO	2D010010	Sets the input device in driver device.
Control DI 2	U16	rw	RxPDO	2D020010	Sets the input device in driver device.
Control DI 3	U16	rw	RxPDO	2D030010	Sets the input device in driver device.
Control DI 4	U16	rw	RxPDO	2D040010	Sets the input device in driver device.
Control DI 5	U16	rw	RxPDO	2D050010	Sets the input device in driver device.
Control DI 6	U16	rw	No	2D060010	Sets the input device in driver device.
Control DI 7	U16	rw	No	2D070010	Sets the input device in driver device.
Statusword	U16	ro	TxPDO	2D110010	Gets the status of the driver device.
Status DO 1	U16	ro	TxPDO	2D120010	Sets the output device in driver device.
Status DO 2	U16	ro	TxPDO	2D130010	Sets the output device in driver device.
Status DO 3	U16	ro	TxPDO	2D140010	Sets the output device in driver device.
Status DO 4	U16	ro	TxPDO	2D150010	Sets the output device in driver device.
Status DO 5	U16	ro	TxPDO	2D160010	Sets the output device in driver device.
Status DO 6	U16	ro	No	2D170010	Sets the output device in driver device.
Status DO 7	U16	ro	No	2D110010	Sets the output device in driver device.
Target position	I32	rw	RxPDO	607A0020	The command position to output to the driver device.
Target velocity	I32	rw	RxPDO	60FF0020	The set velocity to output to the driver device.
Target torque	I16	rw	RxPDO	60710010	The command torque to output to the driver device.
Positive torque limit value	U16	rw	RxPDO	60E00010	Sets positive direction torque limit value in the driver device.
Negative torque limit value	U16	rw	RxPDO	60E10010	Sets negative direction torque limit value in the driver device.
Position actual value	I32	ro	TxPDO	60640020	The current value of the driver device.
Velocity actual value	I32	ro	TxPDO	606C0020	The current speed of the driver device.
Following error actual value	I32	ro	TxPDO	60F40020	The droop pulse of the driverdevice.
Torque actual value	I16	ro	TxPDO	60770010	The the current torque of the driver device.
Polarity	U8	rw	No	607E0008	Sets the rotation direction selection of driver device

Slave object name	DataType	Access	PDO mapping	Default	Description
Encoder increments	U32	rw	No	608F0120	Gets the encoder resolution of driver device.
Motor revolutions	U32	rw	No	608F0220	Gets the motor speed of the driver device.
SI unit velocity	U32	rw	No	60A90020	Gets the SI unit velocity of the driver device.
Max motor speed	U32	rw	No	60800020	Gets the maximum speed of the servo motor from the driver device.
Max torque	U16	rw	No	60720010	Gets the maximum torque of the servo motor from the driver device.
Watch dog counter DL	U16	rw	RxPDO	1D010110	Notifies the watch dog counter value to the driver device.
Watch dog counter UL	U16	ro	TxPDO	1D020110	Gets the watch dog counter value to the driver device.
Supported Control DI 1	U16	ro	No	2D000110	Gets the input device supported by the driver device.
Supported Control DI 2	U16	ro	No	2D000210	Gets the input device supported by the driver device.
Supported Control DI 3	U16	ro	No	2D000310	Gets the input device supported by the driver device.
Supported Control DI 4	U16	ro	No	2D000410	Gets the input device supported by the driver device.
Supported Control DI 5	U16	ro	No	2D000510	Gets the input device supported by the driver device.
Supported Control DI 6	U16	ro	No	2D000610	Gets the input device supported by the driver device.
Supported Control DI 7	U16	ro	No	2D000710	Gets the input device supported by the driver device.
Supported Status DO 1	U16	ro	No	2D100110	Gets the output device supported by the driver device.
Supported Status DO 2	U16	ro	No	2D100210	Gets the output device supported by the driver device.
Supported Status DO 3	U16	ro	No	2D100310	Gets the output device supported by the driver device.
Supported Status DO 4	U16	ro	No	2D100410	Gets the output device supported by the driver device.
Supported Status DO 5	U16	ro	No	2D100510	Gets the output device supported by the driver device.
Supported Status DO 6	U16	ro	No	2D100610	Gets the output device supported by the driver device.
Supported Status DO 7	U16	ro	No	2D100710	Gets the output device supported by the driver device.
Home offset	I32	rw	No	607C0020	Sets the difference between zero position and homing position of the machine coordinate system in the driver device.
Home cycle counter	U32	ro	No	2D3D0020	Gets the encoder position within one revolution saved as the home position from the driver device.
Home ABS counter	I16	ro	No	2D3E0010	Gets the encoder multiple revolution counter saved as the home position from the driver device.
Initial position	I32	ro	No	2D3F0020	Gets the current position at power-on from the driver device.
Initial cycle counter	U32	ro	No	2D400020	Gets the encoder position within one revolution at power-on from the driver device.
Initial ABS counter	I16	ro	No	2D410010	Gets the encoder multiple revolution counter at power-on from the driver device.
Max ABS counter	U32	ro	No	2D420020	Gets the maximum value of the encoder multiple revolution counter from the driver device.
Velocity limit value	U32	ro	RxPDO	2D200020	Sets the speed limit value in the driver device.
Encoder status 1	U32	ro	No	2D350120	Gets the encoder status from the driver device.
Encoder status 2	U32	ro	No	2D350220	Gets the scale measurement encoder status from the driver device.

Slave object name	DataType	Access	PDO mapping	Default	Description
Scale measurement encoder resolution	U32	ro	No	2D380020	Gets the scale measurement encoder resolution from the driver device.
Current alarm	U32	ro	TxPDO	2A410020	Gets the occurring alarm from the driver device.
Sync ABS counter	I16	ro	TxPDO	2D220010	Gets the encoder multiple revolution counter from the driver device.
Sync cycle counter	U32	ro	TxPDO	2D210020	Gets the encoder multiple revolution counter from the driver device.
Scale measurement encoder reception status	U32	ro	No	2D3C0020	Gets the encoder position within one revolution from the driver device.
Scale cycle counter	U32	ro	No	2D360020	Gets the position within one revolution of the scale measurement encoder from the driver device.
Scale ABS counter	I16	ro	No	2D370010	Gets the multiple revolution counter of the scale measurement encoder from the driver device.

• Whether or not the setting is required for each axis type

Whether the slave object is available in each axis type or not is as shown below depending on the slave object setting and the PDO mapping setting.

○: Setting possible ... It can be set in the slave object setting of the engineering tool. For a slave object whose PDO mapping attribute is TxPDO or RxPDO in the real drive axis, be sure to set the setting value of the slave object setting to the PDO mapping setting. When the setting is valid, the axis holds the applicable slave object.

×: Setting not possible ... It cannot be set in the slave object setting of the engineering tool. The axis does not hold the applicable slave object.

□: Unnecessary ... It cannot be set in the slave object setting of the engineering tool. However, since it is emulated in the motion system, the axis holds the applicable slave object.

Slave object name	Real drive axis	Real Encoder axis*1	Virtual drive axis	Virtual Encoder axis	Virtual linked axis
Supported drive modes	○	×	×	×	×
Modes of operation	○	×	×	×	×
Modes of operation display	○	×	×	×	×
Controlword	○	×	×	×	×
Control DI 1	○	×	×	×	×
Control DI 2	○	×	×	×	×
Control DI 3	○	×	×	×	×
Control DI 4	○	×	×	×	×
Control DI 5	○	×	×	×	×
Control DI 6	○	×	×	×	×
Control DI 7	○	×	×	×	×
Statusword	○	×	×	×	×
Status DO 1	○	×	×	×	×
Status DO 2	○	×	×	×	×
Status DO 3	○	×	×	×	×
Status DO 4	○	×	×	×	×
Status DO 5	○	×	×	×	×
Status DO 6	○	×	×	×	×
Status DO 7	○	×	×	×	×
Target position	○	×	□	×	×
Target velocity	○	×	×	×	×
Target torque	○	×	×	×	×
Positive torque limit value	○	×	×	×	×
Negative torque limit value	○	×	×	×	×
Position actual value	○	×	□	○	×
Velocity actual value	○	×	×	×	×
Following error actual value	○	×	×	×	×

Slave object name	Real drive axis	Real Encoder axis*1	Virtual drive axis	Virtual Encoder axis	Virtual linked axis
Torque actual value	○	×	×	×	×
Polarity	○	×	×	×	×
Encoder increments	○	×	×	×	×
Motor revolutions	○	×	×	×	×
SI unit velocity	○	×	×	×	×
Max motor speed	○	×	×	×	×
Max torque	○	×	×	×	×
Watch dog counter DL	○	×	×	×	×
Watch dog counter UL	○	×	×	×	×
Supported Control DI 1	○	×	×	×	×
Supported Control DI 2	○	×	×	×	×
Supported Control DI 3	○	×	×	×	×
Supported Control DI 4	○	×	×	×	×
Supported Control DI 5	○	×	×	×	×
Supported Control DI 6	○	×	×	×	×
Supported Control DI 7	○	×	×	×	×
Supported Status DO 1	○	×	×	×	×
Supported Status DO 2	○	×	×	×	×
Supported Status DO 3	○	×	×	×	×
Supported Status DO 4	○	×	×	×	×
Supported Status DO 5	○	×	×	×	×
Supported Status DO 6	○	×	×	×	×
Supported Status DO 7	○	×	×	×	×
Home offset	○	×	×	×	×
Home cycle counter	○	×	×	×	×
Home ABS counter	○	×	×	×	×
Initial position	○	×	×	×	×
Initial cycle counter	○	×	×	×	×
Initial ABS counter	○	×	×	×	×
Max ABS counter	○	×	×	×	×
Velocity limit value	○	×	×	×	×
Encoder status 1	○	×	×	×	×
Encoder status 2	○	×	×	×	×
Scale measurement encoder resolution	○	×	×	×	×
Current alarm	○	×	×	×	×
Sync ABS counter	○	×	×	×	×
Sync cycle counter	○	×	×	×	×
Scale measurement encoder reception status	○	×	×	×	×
Scale cycle counter	○	×	×	×	×
Scale ABS counter	○	×	×	×	×

*1 For the real encoder axis, refer to the following.
☞ Page 36 Axis

Motion system function/command and slave object

If no slave object is registered in the axis of the motion system, there will be errors and restrictions in some functions/commands. Set the required objects as the following.

For a real axis, registered slave objects can be checked with the member variable of AxisName.Md.Io_Object.

For a virtual axis, registered slave objects are fixed by each axis type.

For details on each slave object, refer to the following.

☞ Page 60 Slave object list

◎: PDO mapping is required, ○: PDO mapping is recommended, —: PDO mapping is not required

Function/command	Required slave object	PDO mapping (only real drive axis)	Operation with no object
MC_MoveVelocity	TargetVelocity	◎	Necessary Slave Object Unset Errar
MC_TorqueControl	TargetTorque	◎	Necessary Slave Object Unset Errar
Torque limit function	PositiveTorqueLimitValue NegativeTorqueLimitValue	○	The command is ignored.
MC_Home	HomeOffset	—	Operates with data set method.

1.5 Axes Group

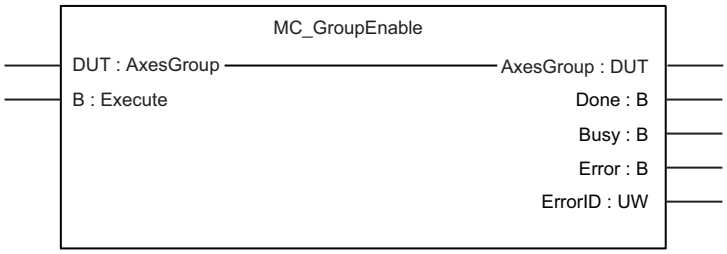
Axes group is used for multiple axes control such as linear interpolation control and 3D linear interpolation control, etc. This section describes an overview of axes group.

The maximum setting number of axes groups depends on the memory. If an axes group cannot be set for memory shortage, the error "Maximum Number of Registrations for Axes Group Over" (error code: 1A60H) will occur. The maximum number of configuration axes of the axes group to be registered is 16.

Relevant variables/FBs

Variable/Structure name	Name	Details
AxesGroupName.AxesGroupRef.		
GroupNo	Axes group No.	Sets axes group No. in the motion system. 0: Not set 1 and later: Setting axes group No.
AxesGroupName.Pr.		
Axis[1..16]	Configuration axis	Sets AxisRef (axis No.) of the axis which configures the axes group.
AxesGroupName.Md.		
GroupStatus	Axes group status	Displays the current axes group status. -1: Invalid (Axes group variable not initialized/Axes group parameter error) 0: GroupDisabled (Axes group disabled) 1: GroupErrorStop (Error stop) 2: GroupStopping (Deceleration stop) 4: GroupStandby (Standby) 5: GroupMoving (Operation)
GroupName	Axes group name	Stores the axes group name.
Axis[1..16]	Configuration axis	Sets AxisRef (axis No.) of the axis which configures the axes group.

MC_GroupEnable

Item	Description
Function overview	Transits specified axes group status from "0: GroupDisabled" to "4: GroupStandby".
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

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

■Function description

This FB transits the axes group status (AxesGroupName.Md.GroupStatus) of the AXES_GROUP_REF structure from "0: GroupDisabled" to "4: GroupStandby".

The FB is executed when Execute is TRUE, and Busy turns TRUE when the processing is normally started.

Done is TRUE and Busy is FALSE when the processing is normally completed.

When an error has occurred in the FB, this FB turns TRUE Error and stores an error code in ErrorID.

For details on error codes, refer to the following.

☞ Page 662 List of Error Codes

For axes group status (AxesGroupName.Md.GroupStatus), refer to the following.

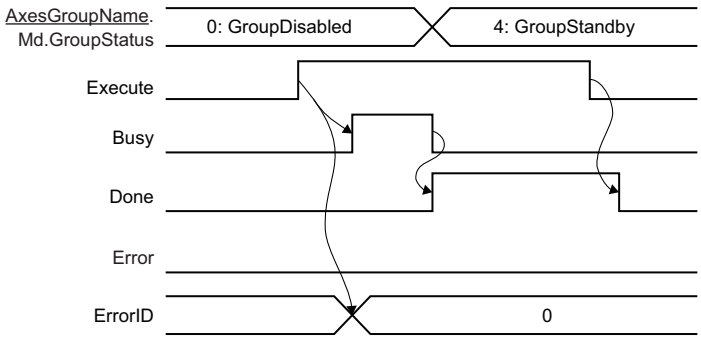
☞ Page 71 Axes group state transition

■Restrictions

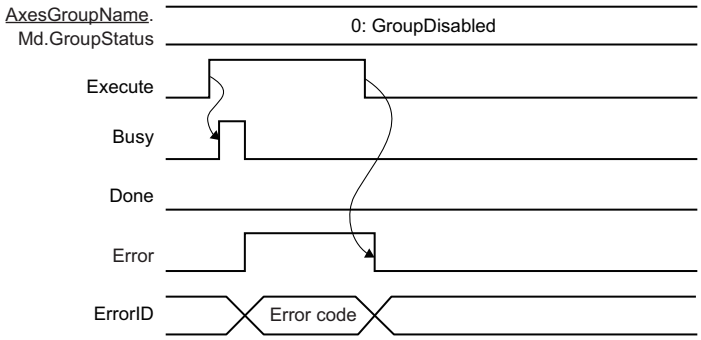
- Axes group No. (GroupNo) of the AXES_GROUP_REF structure must be set in advance before this FB is executed.
- This FB can be executed only when axis status (AxisName.Md.AxisStatus) of all configuration axes is the "4: Standstill" or the "0: Disabled" status.

■Timing chart

[When the FB is normally completed]



[When an error occurs]



■I/O variable

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Axes group information	AxesGroup	AXES_GROUP_REF	↑	—	Omission not possible	☞ Page 706 Axes group variables

■Input variable

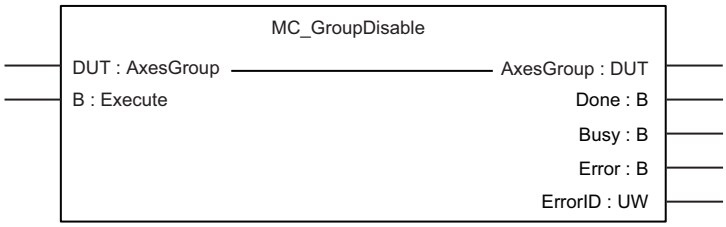
Import ↑: At start

Name	Variable name	Data type	Import	Setting range	Default value	Description
Execute command	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, the FB is executed.

■Output variables

Name	Variable name	Data type	Default value	Description
Execution completion	Done	BOOL	FALSE	Indicates that the status is transited to "4: GroupStandby".
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Error	Error	BOOL	FALSE	When TRUE, it indicates that an error has occurred in the FB.
Error code	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned. For details, refer to the following. ☞ Page 662 List of Error Codes

MC_GroupDisable

Item	Description
Function overview	Transits specified axes group status to "0: GroupDisabled".
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

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

■Function description

This FB transits the axes group status (AxesGroupName.Md.GroupStatus) of the AXES_GROUP_REF structure to "0: GroupDisabled".

The FB is executed when Execute is TRUE, and Busy becomes TRUE when the processing is normally started.

Done becomes TRUE and Busy becomes FALSE when the processing is normally completed.

When an error has occurred in the FB, the Error is changed to TRUE and stores an error code in ErrorID.

For details on error codes, refer to the following.

☞ Page 662 List of Error Codes

For axes group status (AxesGroupName.Md.GroupStatus), refer to the following.

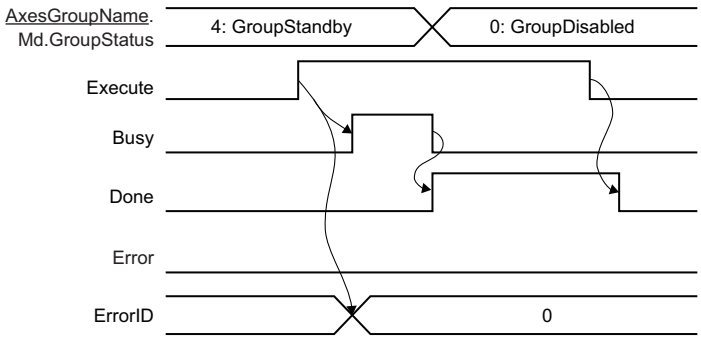
☞ Page 71 Axes group state transition

■Restriction

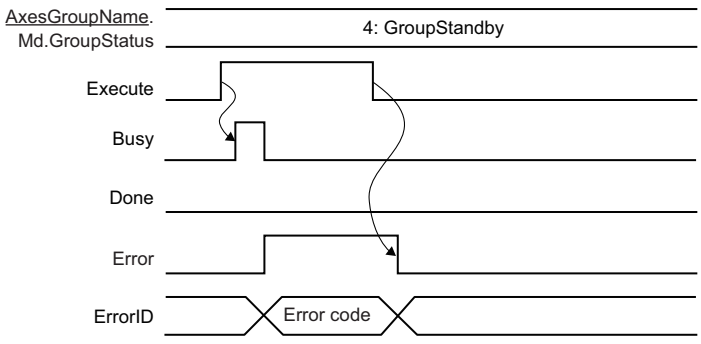
Axes group No. (GroupNo) of the AXES_GROUP_REF structure must be set in advance before this FB is executed.

■Timing chart

[When the FB is normally completed]



[When an error occurs]



■I/O variable

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Axes group information	AxesGroup	AXES_GROUP_REF	↑	—	Omission not possible	☞ Page 706 Axes group variables

■Input variable

Import ↑: At start

Name	Variable name	Data type	Import	Setting range	Default value	Description
Execute command	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, the FB is executed.

■Output variables

Name	Variable name	Data type	Default value	Description
Execution completion	Done	BOOL	FALSE	Indicates that the status is transited to "0: GroupDisabled".
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Error	Error	BOOL	FALSE	When TRUE, it indicates that an error has occurred in the FB.
Error code	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned. For details, refer to the following. ☞ Page 662 List of Error Codes

Settings required for axes group

The following parameters must be set in the engineering tool in order to set axes groups.

Configuration axes can be changed from the program.

For setting methods of axes group, refer to the following.

☞ Page 77 Axes Group Assignment

Item	Description
Axes group No.	Sets axes group No. of the motion system.
Axes group name	Sets an arbitrary axes group name.
Configuration axis	Sets AxisRef (Axis No.) of the axis which configures the axes group.

Axes group variables

Axes group variables are generated by adding the axes group setting in the engineering tool. A generated axes group is assigned to the global label as an axes group variable.
Axes group is defined as an axes group variable configured with monitor information such as parameter information, the current position, and the status.

Axes group variable name (default)

The following names are assigned based on the setting axes group No. It can be changed to an arbitrary name. For details, refer to the following.
☞ Page 77 Axes Group Assignment

Setting axes group No.	Engineering tool default axis name
1	AxesGroup001
2	AxesGroup002
⋮	⋮

Data type

Data types of axes groups are described with the AXES_GROUP type. Data types of axes groups have the following members respectively.

Member name	Data type	Description
AxesGroupRef	AXES_GROUP_REF	Data structure for input/output of the motion control FB
Pr	AXES_GROUP_PRM	Stores parameter data of the axes group. Opens the default value at axes group variable generation. The timing to import to the control changes depending on the parameter.
Md	AXES_GROUP_MONI	Stores monitor data of the axes group. Executes refresh in the fixed cycle for each monitor data.
Cd	AXES_GROUP_CMD	Stores command data for axes group control. Gets the newest value every control operation cycle and uses it for control.

Axes group variable initialization timing

Axes group variables are initialized at the following timings.

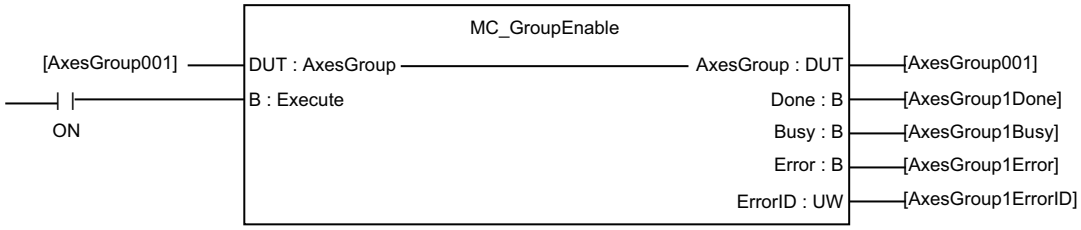
Timing	Processing
Power ON/CPU module reset	Refers to the global label data, and initializes all set axes group variables.
PLC READY [Y0] is turned ON	• Uninitialized axis group Refers to the global label data and initializes all axes group variables. • Initialized axes group Refer to the global label data for the axes group parameter data and import again. However, do not import again the configuration axis. When the parameter error occurs at importing, the axes group is not deleted. At this time, READY [X0] does not turn ON. For the label initialization processing when the PLC READY [Y0] turns OFF to ON, refer to "Label initialization function" in the following manual. 📖MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

How to specify in a user program

When "AXES_GROUP_REF" is written in the input/output setting of the motion control FB, an axes group can be specified by setting the AXES_GROUP_REF type member, AxesGroupRef (AxesGroupName.AxesGroupRef), of the axes group variable.

Ex.

When executing the MC_GroupEnable with the following conditions:
axes group No.: 1, axes group name: AxesGroup001



Enabling axes group

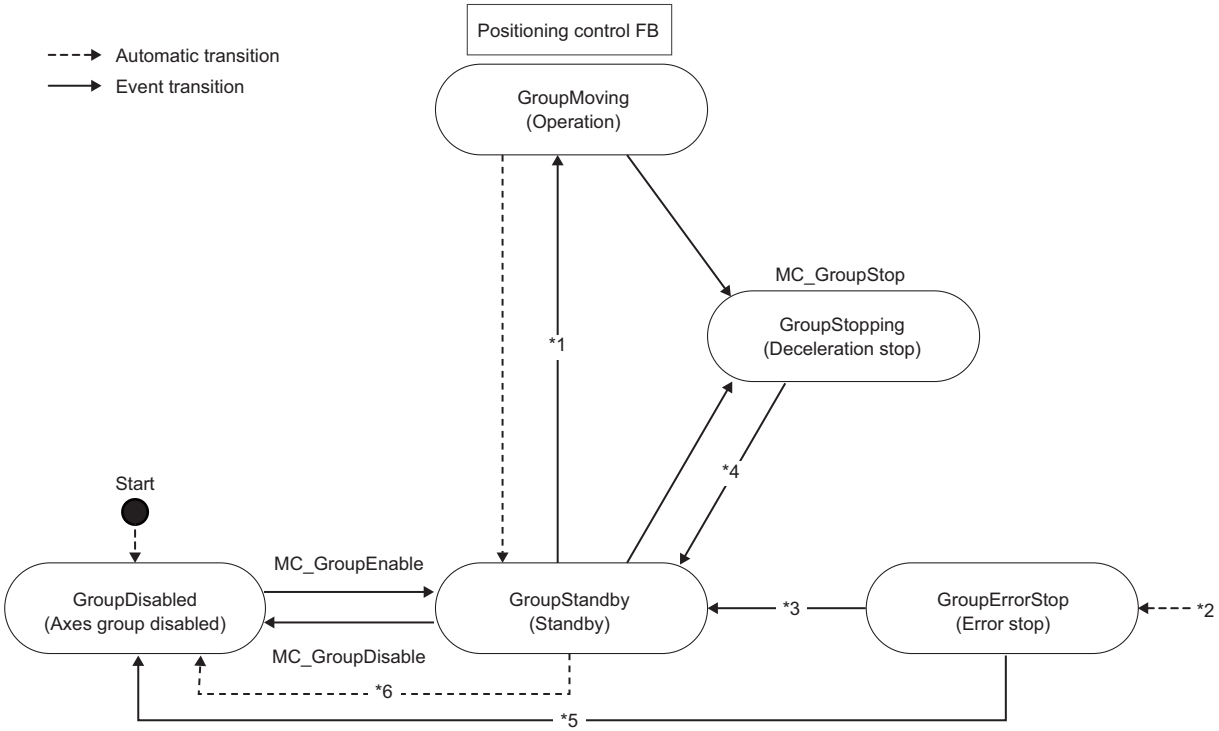
When an axes group is used in a motion FB, the axes group status (AxesGroupName.Md.GroupStatus) must be "4: GroupStandby" by executing MC_GroupEnable to the axes group. The axes group can be started in the motion FB when the axes group status (AxesGroupName.Md.GroupStatus) is "4: GroupStandby" and the axis status (AxisName.Md.AxisStatus) of all configuration axes is "4: Standstill".

Axes group state transition

The following shows status that an axes group can enter.
The current status can be referred with axes group status (AxesGroupName.Md.GroupStatus).
When the axes group is set to "4: GroupStandby", AxisName.Md.UseInGroup of the configuration axes turns TRUE.
When the motion FB is executed for the axis (the axis of which AxisName.Md.UseInGroup is TRUE) used in the axes group, the error "Start Not Possible" (error code: 1AADH) occurs. The axes group should be disabled before executing instructions.

Status	Description
0: GroupDisabled	Axes group is disabled. MCFB cannot be executed in this status. The axis which is being controlled will be stopped immediately to transit this status.
1: GroupErrorStop	Axes group is in the deceleration stop or the stop status by an error occurrence. When the axis status (AxisName.Md.AxisStatus) is set to "1: ErrorStop" in any of the configuration axes, the axes group also enters this status.
2: GroupStopping	The deceleration stop is executed by MC_GroupStop. After the axis stop is completed, this status will be maintained until the Execute of MC_GroupStop turns FALSE. The MCFB cannot be executed in this status.
4: GroupStandby	Axes group is enabled and stand-by. (Not depending on the servo ON/OFF status.)
5: GroupMoving	The positioning control FB is executed in an axes group. When transiting into this status, the axis status (AxisName.Md.AxisStatus) of all configuration axes turn to "7: SynchronizedMotion".

The status is transited as the following figure according to start of the MCFB.



*1 All group operation FB
*2 Transits when an error caused by an axes group occurs in the status other than GroupDisabled.
Transits when an error caused by an configuration axes occurs in the status other than GroupDisabled.
*3 Transits when status error is cleared by the axes group error reset.
*4 Transits when MC_GroupStop.Done = TRUE and MC_GroupStop.Execute = FALSE.
*5 Transits when an axes group is disabled as status error is cleared by carrying out MC_GroupDisable or resetting axes group error.
*6 Transits when a stop of program execution is completed by RUN status → STOP status

For classifications of the motion FB, refer to the following.

☞ Page 721 List of FBs

Axes group enabled

This section describes enabling an axes group.

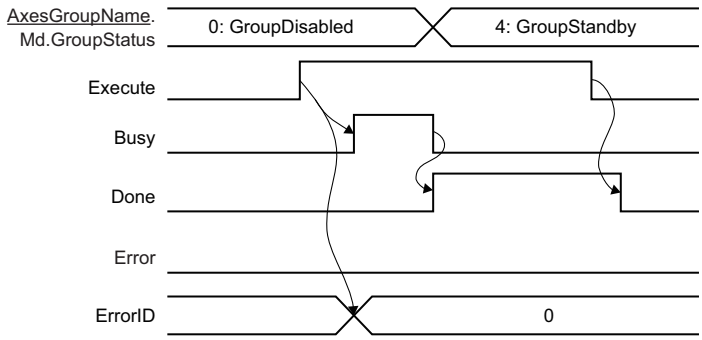
Control details

Specify AxesGroup and change the axes group status (AxesGroupName.Md.GroupStatus) into the "4: GroupStandby" status.

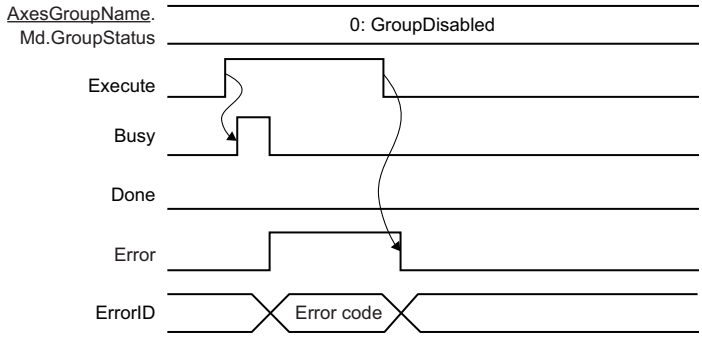
- The axes group can start the motion FB only when the axes group status (AxesGroupName.Md.GroupStatus) is the "4: GroupStandby" status.
- MC_GroupEnable can be executed only when the axis status (AxisName.Md.AxisStatus) of all configuration axes is the "4: Standstill" or the "0: Disabled" status.
- When the axes group status (AxesGroupName.Md.GroupStatus) is set to the "4: GroupStandby" status, AxisName.Md.UseInGroup of the configuration axes becomes TRUE.
- When specifying another axes group including a configuration axis which AxisName.Md.UseInGroup is TRUE and executing MC_GroupEnable, the error "Axes Group Configuration Axis Is in Use" (error code: 3496H) occurs.
- If the error occurs in any of the configuration axes, the axes group status (AxesGroupName.Md.GroupStatus) will be "1: GroupErrorStop".
- If MC_GroupEnable is executed to the axes group which axes group is already valid, Done becomes TRUE and ends.

■Timing chart

[When the operation is normally completed]



[When an error occurs]



■Details of setting items

• Done (Execution completion)

It turns TRUE when the axes group status (AxesGroupName.Md.GroupStatus) has turned to "4: GroupStandby".

• Busy (Executing)

It turns TRUE during FB execution. It turns FALSE after the axes group status (AxesGroupName.Md.GroupStatus) turns to "4: GroupStandby".

• Error (Error)

It turns TRUE when an error occurs in this FB.

• ErrorID (Error code)

It returns error code of the error occurred in this FB.

Precautions

Axes group No. (GroupNo) of the AXES_GROUP_REF structure must be set in advance before this FB is executed.

This FB can be executed only when the axis status (AxisName.Md.AxisStatus) of all configuration axes are "4: Standstill" or "0: Disabled".

Axes group disabled

This section describes disabling an axes group.

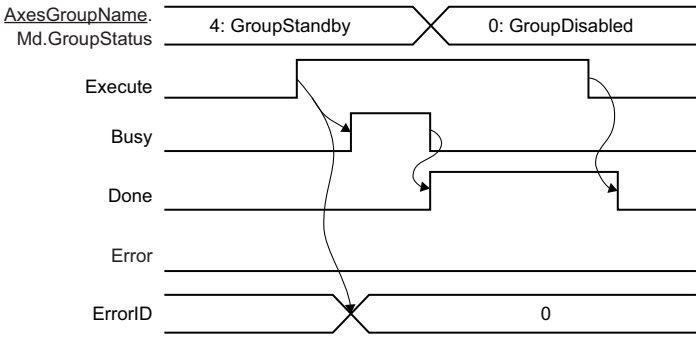
Control details

Specify AxesGroup and change the axes group status (AxesGroupName.Md.GroupStatus) into the "0: GroupDisabled" status.

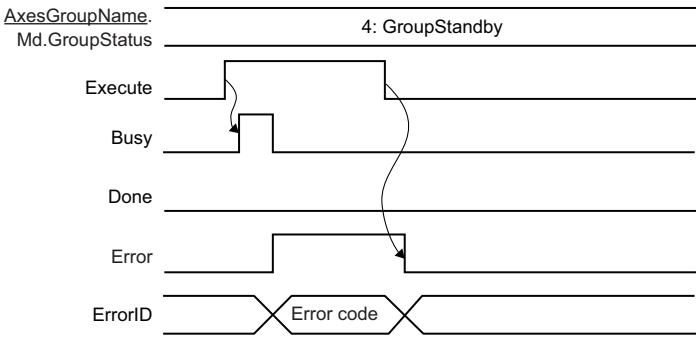
- An axes group cannot execute an operation FB when the axes group status (AxesGroupName.Md.GroupStatus) is the "0: GroupDisabled" status.
- MC_GroupDisable can be executed only when the axes group status (AxesGroupName.Md.GroupStatus) is the "4: GroupStandby" or "1: GroupErrorStop" status. If this FB is executed when the axes group status is "1: GroupErrorStop" and the axes group is in operation, it will transits to "0: GroupDisabled" status after stop completion. If this FB is executed when the axes group status (AxesGroupName.Md.GroupStatus) is other than the "4: GroupStandby" status or "1: GroupErrorStop" status, the error "Axes Group Status Incorrect (Axes Group Disable)" (error code: 1A97H) occurs. The positioning control FB stops. For details on stop processing, refer to the following.
  ☞ Page 162 Stop
- When the axes group status (AxesGroupName.Md.GroupStatus) is set to "0:GroupDisabled", the configuration axes (AxisName.Md.UserInGroup) will be FALSE.
- When MC_GroupDisable is executed to the axes group with invalid axes group, Done becomes TRUE and ends.

■Timing chart

[When the operation is normally completed]



[When an error occurs]



■Details of setting items

• Done (Execution completion)

It turns TRUE when the axes group status (AxesGroupName.Md.GroupStatus) has turned to "0: GroupDisabled".

• Busy (Executing)

It turns TRUE during FB execution. It turns FALSE after the axes group status (AxesGroupName.Md.GroupStatus) turns to "0: GroupDisabled".

• Error (Error)

It turns TRUE when an error occurs in this FB.

• ErrorID (Error code)

It returns error code of the error occurred in this FB.

Precautions

Axes group No. (GroupNo) of the AXES_GROUP_REF structure must be set in advance before this FB is executed.

Precautions

If the parameter error occurs at axes group variable initialization, the error "Out of Parameter Range (Axes Group)" (error code: 1D81H) occurs. In this case, the axes group state (AxesGroupName.Md.GroupStatus) turns to "-1: Invalid". The axes group of which the status is" -1: Invalid", refreshing of the monitor data, etc. is not executed. If the axes group is specified by the user program, the error "Out of Axes Group No. Range" (error code: 3402H) occurs.

1.6 Axes Group Assignment

Axes groups can be set in the engineering tool. For the setting method, refer to Help of the engineering tool.
Based on parameter settings of the axes group, the instance of the axes group is initialized when the power turns ON or when the PLC READY [Y0] turns ON.

Relevant variables/FB

Variable/Structure name	Name	Details
AxesGroupName.AxesGroupRef.		
GroupNo	Axes group No.	Sets the axes group No. 0: No setting 1 to 10000: Setting axes group No.
AxesGroupName.Pr.		
Axis[1..16]	Configuration axis	Sets AxisRef of the axis which configures the axes group.
AxesGroupName.Md.		
NumberOfAxes	Number of configuration axes	Indicates the number of configuration axes of the axes group.
GroupName	Axes group name	Stores the axes group name.
Axis[1..16]	Configuration axis	Stores AxisRef of the axis which configures the axes group.

Setting with Engineering tool

In the axes group setting window of the engineering tool, an axes group can be generated/initialized by setting the axes group setting as a parameter. The axes group set in the axes group setting window is assigned to the global label data as an axes group variable.

Setting items

■Axes group No.

It is an identifier on the motion system control. If the same No. is set in the multiple axes groups, the error "Axes Group No. Setting Duplication Error" (error code: 1A58H) will occur.

Axes group No. setting: 1 to 10000

The axes group whose axes group No. is "0" is regarded as an unset axes group, and it cannot be used for control.

■Axes group name

Set an axes group name of the applicable axes group arbitrarily with up to 127 characters.

An instance is generated as a set axes group name.

■Configuration axis

Set configuration axes of an applicable axes group with AxesGroupName.Pr.Axis[1..16].

Set the AxisName.AxesRef structure of the axis variable for the configuration axes.

Set the configuration axes of the maximum 16 axes in the left-justified format from the configuration axis 1.

The error "No axis" (error code: 1A99H) and the error "Axes Group Configuration Axis Calculation Cycle Incorrect" (error code: 1A59H) occurs for the following cases.

- When there is no configuration axis
- When the operation cycles of the configuration axes are mismatched.

Setting procedures with the engineering tool

The following shows how to create an axes group with the engineering tool.

1. Create an axis, and set the axis name. (A variable instance of the AXIS_* type is generated.)

Navigation window ⇨ "Axis" ⇨ Right click ⇨ [New Data]

2. Create an axes group, and set the axes group No. and the axes group name. (A variable instance of the AXES_GROUP type is generated.)

Navigation window ⇨ "Axis" ⇨ Right click ⇨ [New Data]

3. Add an axis to the configuration axes of the axes group.

[Axes group setting]

Memory usage

Memory usage for the axes group is 60 [K bytes] per an axes group.

The memory capacity for axis data is specified by System.PrConst.Addon_Axis.RamSizeMax.

When the axis data memory usage exceeds the capacity at initializing the axes group variable, the error "Maximum Number of Registrations for Axes Group Over" (error code: 1A60H) occurs and the axes group is not generated. Check the memory capacity and execute the software reboot.

2 AXIS MANAGEMENT FUNCTIONS

2.1 Technical Units

The position command unit and the velocity command unit used in the motion control can be set in the "unit setting". Since the unit can be specified freely depending on the control target, intuitive programming and monitoring are available.

Relevant variables

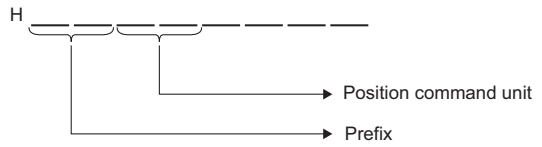
For the setting methods, refer to Help of the engineering tool.

Variable/Structure name	Name	Details
AxisName.Pr. AxesGroupName.Pr.		
Unit_Position	Position command unit	Sets the position command unit to be used in the motion control.
Unit_Velocity	Velocity command unit	Sets the velocity command unit to be used in the motion control.
Unit_PositionString	Position command unit character string	Sets the command unit to be used in the motion control in a character string. It is used when the position unit is set to HFF (an arbitrary unit character string).
AxisName.Md. AxesGroupName.Md.		
Unit_PositionDisplay	Position command unit display	Outputs the position command unit that is being controlled in a character string.
Unit_VelocityDisplay	Velocity command unit display	Outputs the velocity command unit that is being controlled in a character string.

Setting methods

The position command unit and the velocity command unit can be set as the follows.

AxisName(AxesGroupName).Pr.Unit_Position



■Setting value of the position command unit

Setting value	Position command unit
H00	pulse
H01	m
H41	degree
HB4	Revolution
HC0	inch
HFF	An arbitrary unit character string*1

*1 For setting an arbitrary unit character string, set AxisName(AxesGroupName).Pr.Unit_PositionString.

■Setting value of the prefix

When the position command unit setting is HFF (an arbitrary unit character string), this setting is ignored and the control is operated with H00 (1×10^0).

Setting value	Prefix
H00	1×10^0
HFD	1×10^{-3} (m)
HFA	1×10^{-6} (μ)
HF7	1×10^{-9} (n)

Ex.

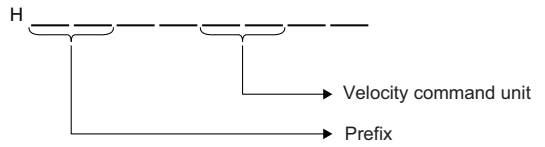
Setting value of AxisName(AxesGroupName).Pr.Unit_Position and display examples of AxisName(AxesGroupName).Md.Unit_PositionDisplay

AxisName(AxesGroupName).Pr.Unit_Position	AxisName(AxesGroupName).Md.Unit_PositionDisplay
H00010000	m
HFD010000	mm
HFA010000	μm
HF7010000	nm
H00B40000	Revolution
H00410000	degree
HFA410000	$\times 10^{-6}$ degree
H00000000	pulse
H00C00000	inch

Point

If "1" is set in the program when AxisName(AxesGroupName).Pr.Unit_Position is HFD010000, the position will be regarded as 1.0 mm.

AxisName(AxesGroupName).Pr.Unit_Velocity



■Setting value of the velocity command unit

Setting value	Velocity command unit
H03	s
H47	min

■Setting value of prefix

Setting value	Prefix
H00	1×10^{0}
HFD	1×10^{-3} (m)
HFA	1×10^{-6} (μ)
HF7	1×10^{-9} (n)

Ex.

Setting value of the AxisName(AxesGroupName).Pr.Unit_Velocity and display examples of AxisName(AxesGroupName).Md.Unit_VelocityDisplay

AxisName(AxesGroupName).Pr.Unit_Velocity	AxisName(AxesGroupName).Md.Unit_VelocityDisplay
H00004700	[Position command unit]/min
HFD000300	[Position command unit]/ms
HFA000300	[Position command unit]/μs
HF7000300	[Position command unit]/ns

Point

If AxisName(AxesGroupName).Pr.Unit_Position is HFD010000 and AxisName(AxesGroupName).Pr.Unit_Velocity is HFD000300, [mm/ms] is displayed on AxisName(AxesGroupName).Md.Unit_VelocityDisplay.
If AxisName(AxesGroupName).Pr.Unit_Position is HFF000000 (An arbitrary unit character string), number of displayable characters from the beginning out of the position command unit character strings is displayed on AxisName(AxesGroupName).Md.Unit_VelocityDisplay.
If "1" is set in the program when AxisName(AxesGroupName).Pr.Unit_Velocity is HFD000300, the velocity will be regarded as 1.0 [position command unit]/ms.

Ex.

Setting value of AxisName(AxesGroupName).Pr.Unit_Position, AxisName(AxesGroupName).Pr.Unit_Velocity and display example of AxisName(AxesGroupName).Md.Unit_VelocityDisplay

AxisName(AxesGroupName).Pr.Unit_Position	AxisName(AxesGroupName).Pr.Unit_Velocity	AxisName(AxesGroupName).Md.Unit_VelocityDisplay
HFD010000	HFD000300	[mm/ms]

Combination of command units during multiple axes positioning control

Set command units of the axes group and the interpolation axis to be matched.

When the command units are mismatched, specify the input value of FB as shown below.

- For the input regarding position in multiple axes positioning control (Position, Distance input, etc.), the value based on the position command unit of the each interpolation axis is required to specify. For details, refer to the following.
  ☞ Page 260 Position (Target position)/Distance (Movement amount)
- For the input of Velocity, Acceleration, Deceleration and Jerk in multiple axes positioning control, the value based on the velocity command unit used for each control is required to specify. For details, refer to the following.
  ☞ Page 260 Velocity (Velocity)
  Velocity command units used for each control are shown below.
  The following operations depend on interpolation controls.

■Linear interpolation control (MCv_MoveLinearInterpolateAbsolute, MCv_MoveLinearInterpolateRelative)

VelocityMode	Operation
0: VectorSpeed (Vector speed)	• The combined movement amount is calculated from the value of the current position and the Position/Distance input. The position command unit of the axes group and the interpolation axis are not affected. • Unit of the axes group is used for the velocity command unit in controling.
1: LongAxisSpeed (Long axis speed)	• The axis whose movement amount value is the largest is regarded as the long axis from the value of the current position and the Position/Distance input. The position command unit of the axes group and the interpolation axis are not affected. • The unit of the long axis is used for the velocity command unit in controlling.Unit of the axes group is not used.
2: ReferenceAxisSpeed (Reference axis speed)	The unit of the reference axis (First element of LinearAxes input) is used for the velocity command unit in controlling.

For details, refer to the following.

☞ Page 254 Linear interpolation control

■Circular interpolation control (MCv_MoveCircularInterpolateAbsolute,MCv_MoveCircularInterpolateRelative)

- Calculates circular path from the value of current position, AuxPoint input, and EndPoint input. The position command unit of the axes group and the the configuration axes are not affected.
- The unit of the axes group is used for the velocity command unit in controlling.
  For details, refer to the following.
  ☞ Page 263 Circular interpolation control
- When the combination of the position command unit between the axes group and the interpolation axis is different, the warning "Position Command Unit Mismatch Warning" (warning code: 0D08H) will occur at start and multiple start.
- When the combination is different between the speed command unit and on-going FB at multiple start, the warning "Speed Command Unit Mismatch Warning" (warning code: 0D1EH) will occur at multiple start.

Ex.

Operation when axes group unit and interpolation axis unit are mismatched

The following shows the operation example when the position/velocity command units of the configuration axis 1 and 2, and axes group in linear interpolation control are different.

- Axis setting

Setting item	Setting details	Setting value
Position/Velocity command unit of the axes group	mm/ms	AxesGroupName.Pr.Unit_Position: HFD010000 AxesGroupName.Pr.Unit_Velocity: HFD000300
Position/Velocity command unit of the configuration axis 1	mm/s	AxisName.Pr.Unit_Position: HFD010000 AxisName.Pr.Unit_Velocity: H00000300
Position/Velocity command unit of the configuration axis 2	degree/min	AxisName.Pr.Unit_Position: HFD010000 AxisName.Pr.Unit_Velocity: H00000300

- FB input (MCv_MoveLinearInterpolateAbsolute, MCv_MoveLinearInterpolateRelative)

Setting item	Setting value
LinearAxes	[1 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0] or [1 2]
Position	[100 100 0 0 0 0 0 0 0 0 0 0 0 0 0 0] or [100 100]
Velocity	1000
VelocityMode	0: VectorSpeed (Vector speed)

- Operation

Regards current position[0].

The position/velocity command unit of the axes group is used as the unit of Velocity.

Combined movement amount = $\sqrt{(100^2 + 100^2)}$ = 141 mm is calculated from the movement amount [100 100].

Each axis movement amount of every operation cycle is calculated from the combined movement amount 141 mm and the vector speed 1000 mm/ms.

The calculated each axis movement amount is commanded as the movement amount in the position command unit of each axis.

Precautions

- When the parameter is out of the range,the initial value is stored in AxisName(AxesGroupName).Md.Unit_PositionDisplay and AxisName(AxesGroupName).Md.Unit_VelocityDisplay. (the default value of position command unit: pulse, velocity command unit: pulse/s)
- Even if AxisName(AxesGroupName).Pr.Unit_Position, AxisName(AxesGroupName).Pr.Unit_Velocity and AxisName(AxesGroupName).Pr.Unit_PositionString is changed, the homing request is not TRUE.
- When the number of characters exceeds 31, the 32 or later character is not displayed in the position/velocity command unit display.
- When the character string is not input, the position command unit is not displayed.

2.2 Positioning Range

"Positioning range" is "-10000000000.0 ≤ positioning range < 10000000000.0". The range of AxisName.Md.SetPosition is determined by AxisName.PrConst.RingCount_UpperValue and AxisName.PrConst.RingCount_LowerValue.

Relevant variables

Variable/Structure name	Name	Details
AxisName.PrConst.		
RingCount_Enable	Ring counter enable selection	Selects the ring counter enable/disable. 0: Disabled*3 1: Enabled
RingCount_UpperValue	Ring counter upper limit value	Sets the ring counter upper limit value.*1 -10000000000.0 to 10000000000.0
RingCount_LowerValue	Ring counter lower limit value	Sets the ring counter lower limit value.*1 -10000000000.0 to 10000000000.0
AxisName.Md.		
SetPosition	Set position	Stores the current position address.*2 • The set position is a value which the cumulative current value is rounded with the ring counter range.
CumulativePosition	Cumulative current position	Stores the current position address. • When the current position is changed, the address is changed to the current position change value.
FeedMachinePosition	Feed machine position	Stores the current position address by the machine coordinate. • The accumulative value from the homing complete position is set. • The address is not changed even if the current position is changed.
CommandedPosition	Commanded position	Stores Position/Distance specified in the positioning FB.
ActualPosition	Actual position	Stores the actual current position. • The value of "the current position × driver unit conversion denominator / numerator" is set. For details, refer to the following. ☞ Page 381 Driver unit conversion function • The actual position is a value which is rounded with the ring counter range.

*1 The value is not fetched when the ring counter is disabled.
*2 The stored value includes an error because a floating-point error occurs.
*3 Positioning range is "-10000000000.0 ≤ positioning range < 10000000000.0".

Set position

AxisName.Md.SetPosition can be set to an arbitrary ring address by setting AxisName.PrConst.RingCount_UpperValue and AxisName.PrConst.RingCount_LowerValue. The ring counter upper/lower limit value can be set within the range of -10000000000.0 to 10000000000.0.

Restrictions

The motion control function uses the real number data (floating-point data) for the axis current position and the command target position, etc. Therefore, it may include the truncation error.

Ex.

When the positioning of relative position specification such as MC_MoveRelative is executed repeatedly, it may cumulative the truncation error between the command current position and the specified movement value.
If the truncation error is a problem, execute the positioning of absolute position specification such as MC_MoveAbsolute.
Besides, the truncation error occurred by the real number data processing internal the function can be suppressed to set the number without after the decimal point to the real number type parameter related to the motion control function.

Ex.

When the truncation error is a problem in the following setting (A), the setting (B) becomes a change candidate.

• Setting (A)

Setting item	Setting detail
Position command unit*1	[mm]
Ring counter upper value	100.3 [mm]
Ring counter lower value	-100.7 [mm]
The minimum scale of the target position/movement amount specified by FB input	0.05 [mm]

• Setting (B)

Setting item	Setting detail
Position command unit*1	[μm]
Ring counter upper value	10300 [μm]
Ring counter lower value	-10700 [μm]
The minimum scale of the target position/movement amount specified by FB input	50 [μm]

*1 For details on the setting method, refer to the following.

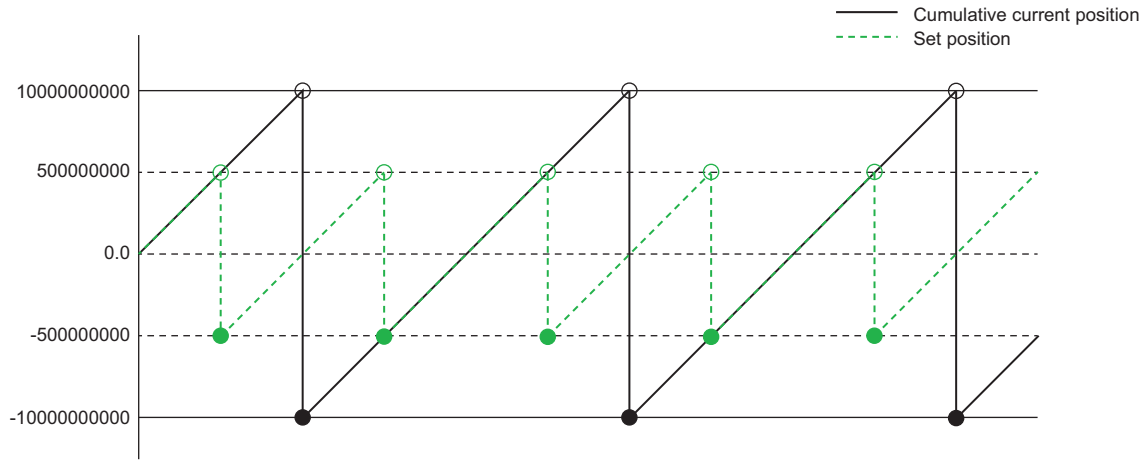
☞ Page 79 Technical Units

Control details

AxisName.Md.SetPosition to be the ring address is calculated based on AxisName.Md.CumulativePosition. The cumulative current position will be the ring address of "-10000000000.0 ≤ cumulative current position < 10000000000.0" regardless of the setting of the ring counter upper/lower value.

Ex.

When the ring counter lower limit value is -500000000.0 and the ring counter upper limit value is 500000000.0, the set position is the ring address of "-500000000.0 ≤ set position < 500000000.0".



The ring counter upper/lower limit value is fetched to the parameter at the timing when the axis is enabled.

Combination with other functions

The available range of positioning differs depending on the type of the positioning control FB.

- Absolute position specification: Within the range of the ring counter
- Relative position specification: Maximum movement amount | 10000000000.0 |

Precautions

When AxisName.PrConst.RingCount_Enable is "1: Enabled", error is checked.

If the ring counter upper/lower limit value is set as follows, an error will occur.

- The ring counter upper/lower limit value is set outside the positioning range.
- The ring counter lower limit value > The ring counter upper limit value
- | The ring counter upper limit value - The ring counter lower limit value | < 2.0

2.3 Velocity Range

Relevant variables

Variable/Structure name		Name	Details
AxisName.Md.			
	SetVelocity	Set velocity	• Stores the set output veocity.*4 • Stores the set velocity of the configuration axes during the axes group operation.*4
	CommandedVelocity	Commanded velocity	• Stores the commanded velocity of the positioning data which is being executed. • Stores "0" after the positioning is completed. • Stores "0" in AxisName.Md.CommandedVelocity of the configuration axes during the axes group operation.
	TargetVelocity	Target velocity	• Stores the actual target velocity whose override and velocity limit value are considered. • Stores "0" after the positioning is completed. • Stores "0" in AxisName.Md.TargetVelocity of the configuration axes during the axes group operation.
	ActualVelocity	Actual velocity	Stores the actual velocity
AxesGroupName.Md.			
	SetVelocity	Set velocity	• Stores the set output veocity during the axes group operation. The value becomes more than "0" as it is regardless of the movement direction of the configuration axes.*1 • Stores "0" after the positioning is completed.
	CommandedVelocity	Commanded velocity	• Stores the commanded velocity of the positioning data.*2 • Stores "0" after the positioning is completed.
	TargetVelocity	Target velocity	• Stores the actual target velocity whose override and velocity limit value are considered.*3 • Stores "0" after the positioning is completed.
	ActualVelocity	Actual velocity	• Stores the actual velocity. It is the vector speed of the actural velocity of the configuration axes. The value becomes more than "0" as it is regardless of the movement direction of the configuration axes. • Stores "0" when the axes group is invalid.

*1 When the reference axis velocity or the long axis velocity is specified in the linear interpolation control, stores the absolute value of the set velocity of the applicable axis.

*2 When the reference axis velocity or the long axis velocity is specified on VelocityMode in the linear interpolation control, stores the commanded velocity of the applicable axis.

*3 When the reference axis velocity or the long axis velocity is specified on VelocityMode in the linear interpolation control, stores the target velocity of the applicable axis.

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

For details on the linear interpolation control, refer to the following.

☞ Page 254 Linear interpolation control

Velocity set range

The following table describes the velocity range that can be specified in each parameter. For the set range when using the override function, refer to the following.

☞ Page 359 Override Function

Name	Range	Condition
Target velocity	0, +0.0001*2 to +2500000000.0*1	When positioning control
	0, ±0.0001*2 to ±2500000000.0*1	When velocity control

*1 If the set velocity is "0" when the FB is executed in the buffer mode, it will be the previous commanded velocity.

*2 The lower limit value of the target velocity is limited by the following restrictions to carry out floating-point operation.

- If the speed that is converted from target velocity to operation cycle is less than 0.00001, the error "Out of Calculation Cycle Converted Speed Range" (error code: 1AE4H) (when changing the velocity, the warning "Out of Calculation Cycle Converted Speed Range Warning" (warning code: 0D2FH)) occurs.
  To improve the precision of floating-point operation, specify the speed that is converted into operation cycle so as not to be less than 0.00001 by changing AxisName(AxesGroupName).Pr.Unit_Position or AxisName(AxesGroupName).Pr.Unit_Velocity.

Ex.

AxisName(AxesGroupName).Pr.Unit_Position H00000000(pulse),
AxisName(AxesGroupName).Pr.Unit_Velocity H00000300(pulse/s)
If target velocity is specified to 0.0001 at operation cycle 0.5 ms
Operation cycle converted speed = 0.0001 pulse/s ÷ 1000 × 0.5 ms = 0.00000005 pulse ...the error "Out of Calculation Cycle Converted Speed Range" (error code: 1AE4H)
If AxisName(AxesGroupName).Pr.Unit_Position is changed to HFD000000(× 10^{-3} pulse) and target velocity is changed to 0.1,
Operation cycle converted speed = 0.1 × 10^{-3} pulse/s ÷ 1000 × 0.5 ms = 0.00005 × 10^{-3} pulse

2.4 Servo ON/OFF

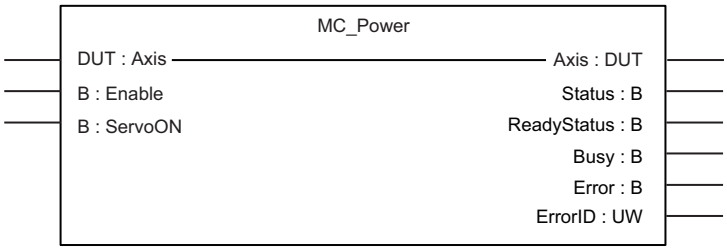
Relevant variables/FB

This function executes servo ON/OFF of the real axis connected to the motion system.

The servo ON enables the operation of the real axis.

Variable/Structure name		Name	Details
AxisName.Md.			
	Driver_ReadyOn	Driver ready ON status	Sets TRUE when the connected driver status (AxisName.Md.Driver_State) is as follows. 5: SwitchedOn 6: OperationEnable 7: QuickStopActive
	Driver_ServoOn	Driver servo ON status	Sets TRUE when the connected driver status (AxisName.Md.Driver_State) is as follows. 6: OperationEnable 7: Quick StopActive

MC_Power

Item	Description
Function overview	Switches a specified axis to the operation possible status.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

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

■Function description

This FB initializes the information of a selected axis and switches the axis status to the operation possible status.

While Enable and the ServoON input are TRUE, the selected axis is switched to the operation possible status.

Busy turns TRUE when the processing is started.

Status turns TRUE when the drive unit has been switched to the operation possible status.

When an error has occurred in the FB, this FB turns Error to TRUE and stores an error code in ErrorID.

For details on error codes, refer to the following.

☞ Page 662 List of Error Codes

The axis status (AxisName.Md.AxisStatus) of the AXIS_MONI structure transits from "0: Disabled" to "4: Standstill".

For the axis status (AxisName.Md.AxisStatus), refer to the following.

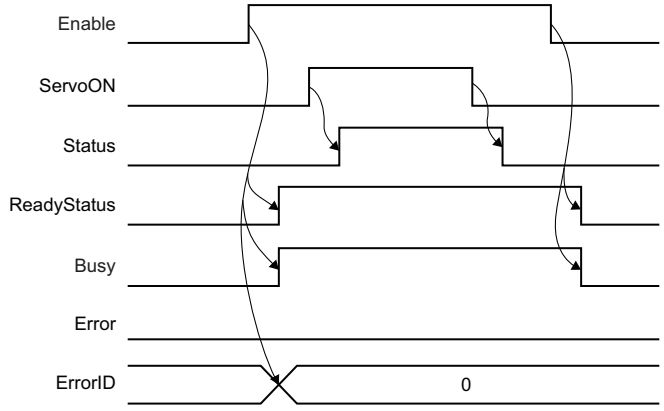
☞ Page 39 State transition

■Restrictions

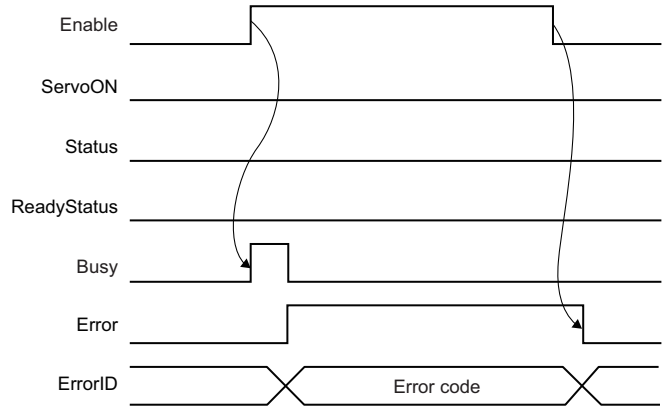
Before executing this FB, set the axis No. (AxisNo) of the AXIS_REF structure.

■Timing chart

[When the FB is normally completed]



[When an error occurs]



■I/O variable

Imput import ⎍: Always

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Axis information	Axis	AXIS_REF	⎍	—	Omission not possible	☞ Page 702 Axis variables For MC_Power, the axis information is established when the command is called for the first time after switching to the PLC READY [Y0] ON.

■Input variables

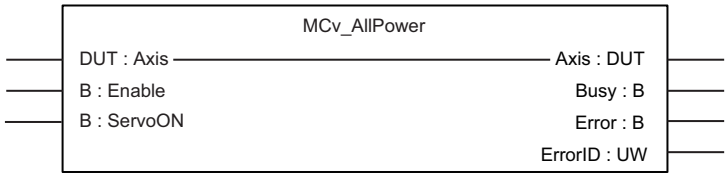
Import ⎍: Always

Name	Variable name	Data type	Import	Setting range	Default value	Description
Enable	Enable	BOOL	⎍	TRUE, FALSE	FALSE	While Enable input is TRUE, the axis control is valid.
Servo ON request	ServoON	BOOL	⎍	TRUE, FALSE	FALSE	Specifies the servo ON request.

■Output variables

Name	Variable name	Data type	Default value	Description
Operable	Status	BOOL	FALSE	Indicates that the servo amplifier is ready for operation.
Ready ON status	ReadyStatus	BOOL	FALSE	Indicates the ready ON/OFF status.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Error	Error	BOOL	FALSE	When TRUE, it indicates that an error has occurred in the FB.
Error ID	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned. For details, refer to the following. ☞ Page 662 List of Error Codes

MCv_AllPower

Item	Description
Function overview	Switches every axis to the operation possible status.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

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

■Function description

This FB initializes the information of every axis and switches the axis status to the operation possible status.

While Enable and the ServoON input are TRUE, every axis is switched to the operation possible status.

Busy turns TRUE when the processing is started.

Ignores the setting of Axis.

When an error has occurred in the FB, this FB turns Error to TRUE and stores an error code in ErrorID.

For details on error codes, refer to the following.

☞ Page 637 TROUBLESHOOTING

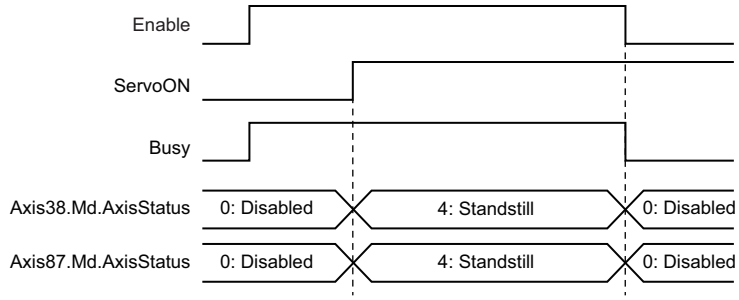
■Restrictions

If the axes specifying MC_Power, MC_Power command is given priority at switching to operable status.

■Timing chart

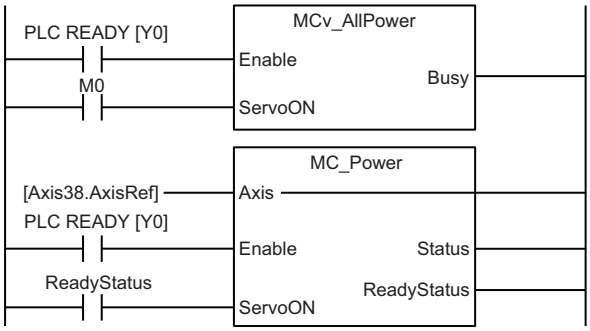
Ex.

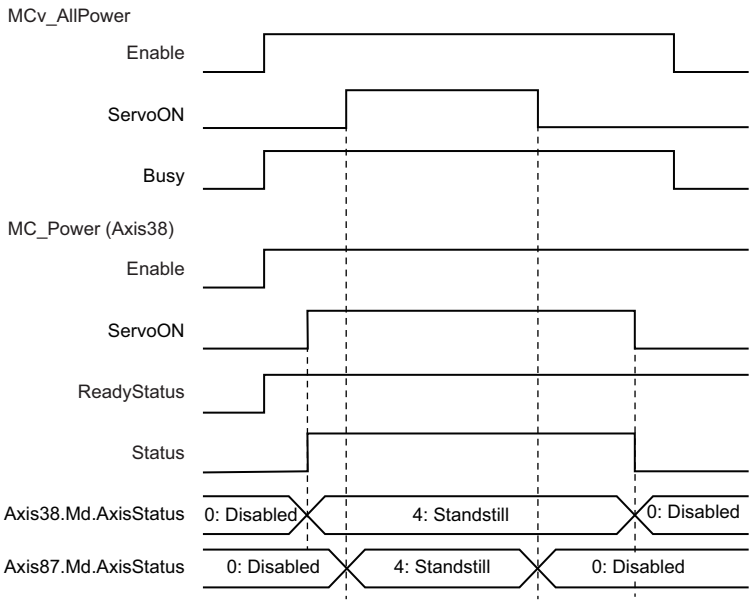
When the axis 38 and 87 are set as real axes



Ex.

When MC_Power is used together to the axis 38





■I/O variable

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Axis information	Axis	AXIS_REF	↑	—	Omission possible	Omission is possible when the FB is used in the motion system side. Ignores the setting value. Set Axis.StartIO when the FB is used in the CPU module side. Ignores Axis.AxisNo.

■Input variables

Import: ⎍: Always

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Enable	Enable	BOOL	⎍	TRUE, FALSE	FALSE	While Enable input is TRUE, the axis control is valid.
Servo ON request	ServoON	BOOL	⎍	TRUE, FALSE	FALSE	Specifies the servo ON request.

■Output variable

Name	Variable name	Data type	Default value	Description
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Error	Error	BOOL	FALSE	When TRUE, it indicates that an error has occurred in the FB.
Error code	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned.

Control details

Servo ON/OFF

The servo ON/OFF of the real axis connected to the motion system is controlled by MC_Power (operation possible) for single axis and MCv_AllPower (all axes operation possible) for all axes.

For the driver status, refer to the following.

☞ Page 94 Control details

■MC_Power input/output and servo ON/OFF status

The servo ON/OFF status and the driver status of the selected axis can be switched as the follows by inputting Enable and ServoON of MC_Power (operation possible).

Input variables		Output variables		Servo ON/OFF status	Driver status (AxisName.Md.Driver_State)
Enable	ServoON	ReadyStatus	Status		
TRUE	ON	ON	ON	Servo ON	6: OperationEnable
	OFF	ON	OFF	Servo OFF	5: SwitchedOn
FALSE	ON	OFF	OFF	Servo OFF	3: SwitchOnDisabled
	OFF	OFF	OFF	Servo OFF	3: SwitchOnDisabled

- If the real axis is rotated by external force during the servo OFF status, follow up processing is performed.
- The servo ON/OFF control can be operated regardless of the control mode. The control mode during the servo OFF status depends on the specification of the driver.
- While the drive unit error occurs, Enable and ServoON are not required to be turned from OFF to ON again since MC_Power/MCv_AllPower has been sent to the driver.

■Input/Output of MCv_AllPower and servo ON/OFF status

The servo ON/OFF status and the driver status of all real axes can be switched as follows by inputting Enable and ServoON of MCv_AllPower.

Input variables		Servo ON/OFF status	Driver status (AxisName.Md.Driver_State)
Enable	ServoON		
TRUE	ON	Servo ON	6: OperationEnable
	OFF	Servo OFF	5: SwitchedOn
FALSE	ON	Servo OFF	3: SwitchOnDisabled
	OFF	Servo OFF	3: SwitchOnDisabled

- If the real axis is rotated by external force during the servo OFF status, follow up processing is performed.
- The servo ON/OFF control can be operated regardless of the control mode. The control mode during the servo OFF status depends on the specification of the driver.
- While the drive unit error occurs, since MC_Power/MCv_AllPower has been sent to the driver, Enable and ServoON are not required to be turned from OFF to ON again.

Point

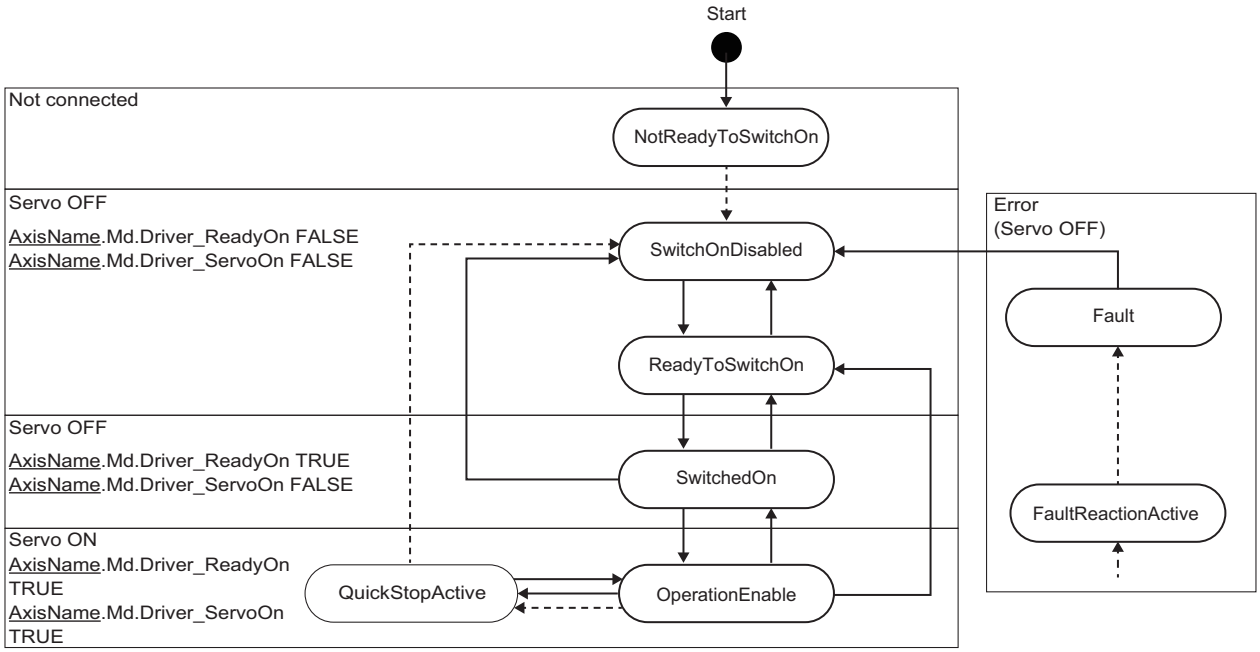
To individually execute servo OFF when using MCv_AllPower, use MC_Power together. When MCv_AllPower and MC_Power are used together, the MC_Power command is given priority.

Driver state transition

The connected driver device as an axis performs operation according to the state transition defined by the CiA402 drive profile shown below. The motion system determines whether the driver is being servo ON or OFF status based on the current driver status.

The current driver status can be monitored by AxisName.Md.Driver_State.

For details on operation in each status, refer to the connected driver manuals.



Precautions

- When MC_Power calls the instruction first after the PLC READY [Y0] is switched to ON, the axis information is determined. Even if the axis information is changed while Enable is FALSE, the change is not reflected.
- Do not set two or more FBs of MC_Power for one axis. The operation is not guaranteed if those are set.
- If MC_Power is executed to the virtual axis, both Status and ReadyStatus are turned ON regardless of the ServoON input.
- The servo ON/OFF status when stopping the program is maintained while the ready [X0] is OFF

2.5 Follow Up

Follow up is the function to reflect the input (current position) from the slave device to the set position of the axis.
In an axis which has the current position (Position actual value) as the slave object, this function can convert the unit of the current position by the driver unit conversion numerator/denominator (electronic gear) and reflect in the set position.

• [For real drive axis]

The current position of the drive unit is converted and reflected in the set position in the servo OFF status. For this processing, even if the current position of the drive unit is moved during servo OFF, the drive unit will not move by the amount of droop pulses and can execute positioning from the stop position.
For the following cases, the follow up is executed when servo status is ON.

• MC_MoveVelocity/MC_TorqueControl are in execution.
• The driver control mode of the driver device is connected by motion system axis control unsupported mode.

[For Real Encoder axis/Virtual Encoder axis]

This function can convert the unit of the current position of the synchronous encoder and reflect it in the set position. By switching the follow up enabled/disabled allows input enabled/disabled from the synchronous encoder.

Operation of this function for each system status

○: Possible, ×: Not possible

Status	Operation availability
STOP	○
RUN	○
Moderate error	○
Major error	×

Relevant variables

Variable/Structure name	Name	Details
AxisName.PrConst.		
Encoder_CounterDisableSignal	Counter disable signal	A signal to switch to the counter disable. Only "HighLevel" or "LowLevel" can be select for Detection (signal detection method). Only "0" can be specified to CompensationTime (Compensation time [s]). "0.0 to 5.0" can be specified to FilterTime (Filter time [s]). If another method is selected, the error "Out of Parameter Range (Axis)" (error code: 1D80H) will occur.
AxisName.Md.		
FollowupDisable	Follow up disabled	Displays the follow up disable status. When it is disabled, the follow up is not executed. FALSE: Follow up enabled TRUE: Follow up disabled
Encoder_CounterDisable	Counter disabled	Turns TRUE while disabling the input from the synchronous encoder. FALSE: Counter enabled TRUE: Counter disabled
Io_PosActualValue	Object data_PosActualValue	Follow up can be used in the axis which has this slave object.
AxisName.Cd.		
FollowupDisable	Followup disabled	Disables follow up. FALSE: Not execute TRUE: Execute the followup disable request
Encoder_CounterDisable	Counter disabled	Switches to the counter disable. FALSE: Not execute TRUE: Execute the counter disable request

Disabling follow up

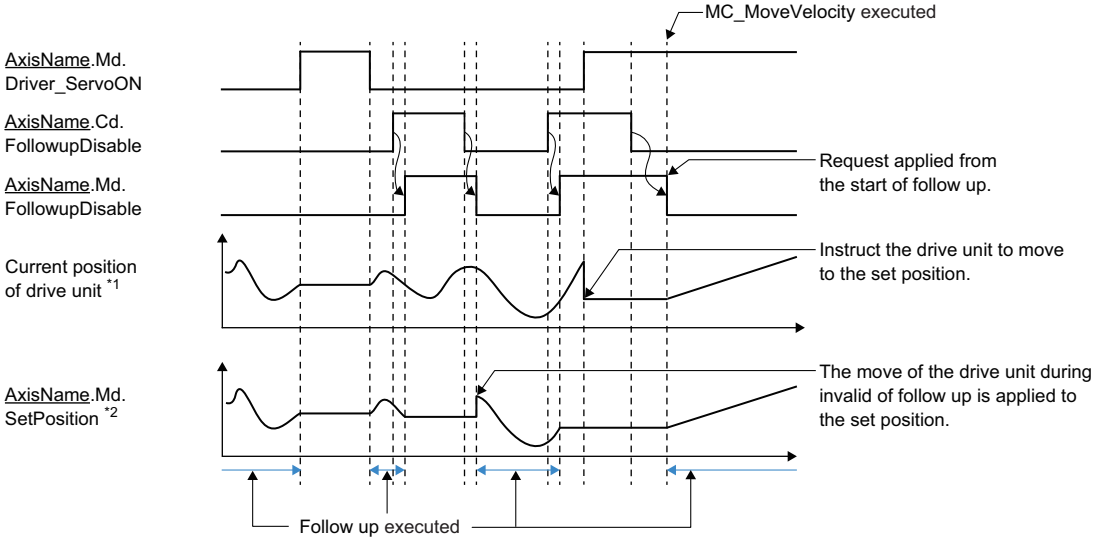
The follow up enabled/disabled can be switched for each axis (default status: enabled). Input from the slave device (the current position) is disabled and not reflected in the set position by switching the follow up to enabled.

Point

That the follow up enabled/disabled is switched is recorded in the event history. Also, for the real drive axis, the servo ON from the follow up disabled status during the servo OFF is recorded in the event history as well. At this time, the difference [driver-based] between the drive unit target position (Target position) and the current position (Position actual value) is also recorded in the event history.

For real drive axis

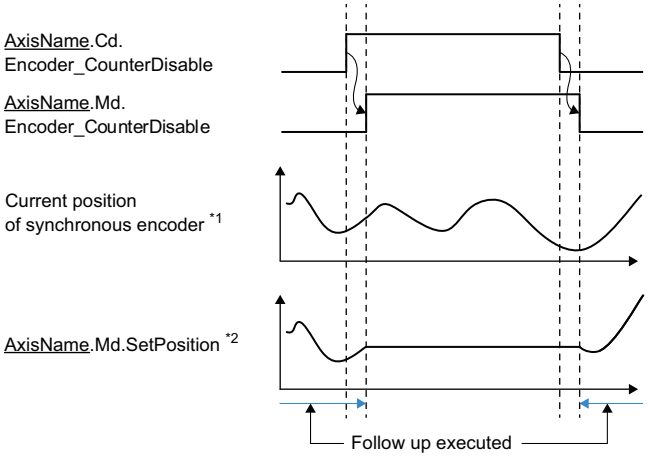
AxisName.Md.FollowupDisable turns TRUE by changing AxisName.Cd.FollowupDisable to TRUE, and the follow up is disabled.



*1 When checking with slave label
*2 When driver unit conversion (Numerator / Denominator) is (1 / 1).

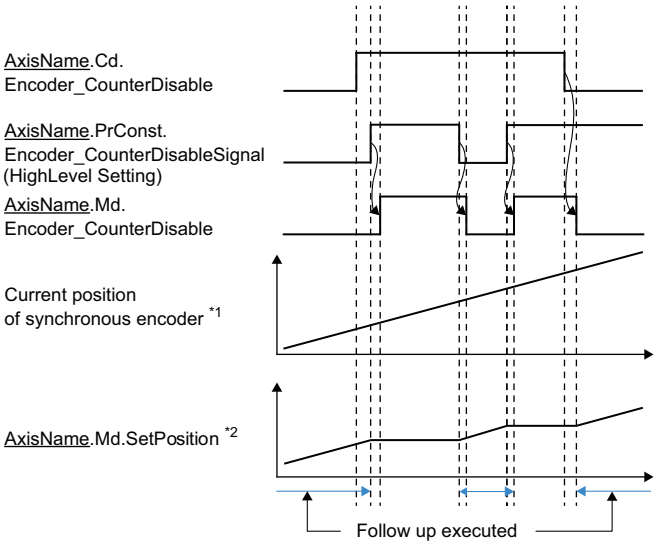
For real encoder axis/virtual encoder axis

If AxisName.PrConst.Encoder_CounterDisableSignal is "no setting", AxisName.Md.Encoder_CounterDisable turns TRUE by setting AxisName.Cd.Encoder_CounterDisable TRUE and the input from the synchronous encoder will be disabled.



*1 When checking with the slave label
*2 When driver unit conversion (Numerator / Denominator) is (1 / 1).

When AxisName.PrConst.Encoder_CounterDisableSignal is set, while AxisName.Cd.Encoder_CounterDisable is TRUE, the counter disabling signal will be valid. AxisName.Cd.Encoder_CounterDisable becomes TRUE while the signal is in detection, and becomes the counter disable. While AxisName.Cd.Encoder_CounterDisable is FALSE, the counter disable signal becomes invalid.



*1 When checking with the slave label
*2 When driver unit conversion (Numerator / Denominator) is (1 / 1).

Precautions

[Only for real drive axes]

- When the follow up is enabled after the current position of the drive unit is moved while the follow up is disabled, the motion system reflects the distance from the drive unit in the set position within one operation cycle. Therefore, when a single axis synchronous control is executed using an applicable axis as the master axis, the command of the movement amount of the set position is transmitted to the slave axis.
- The motion system outputs the distance to the drive unit within one operation cycle for the following cases. Note that a rapid operation may occur in the drive unit.
  When the servo ON occurs next time after the current position of the drive unit is moved in the follow up disabled status during the servo OFF
  When the system stopped in the servo ON status after the control excluding the position loop, such as the speed control, was executed with the follow up disabled status
- The software stroke limit function is invalid during the servo OFF. Therefore, set the software stroke limit in the slave axis or the following axis for safety in order to transmit the command under follow up in the single axis synchronous control to the slave axis.

2.6 Absolute Position Control

This function restores the current position of axes.

Absolute position system

The absolute position system maintains the current position of axes. The absolute position system configuration differs depending on the axis type. For details, refer to the following.

☞ Page 105 Current position restoration for each axis type

The absolute position data used in the absolute position control is maintained inside the motion system as a backup file.

Current position restoration

This function restores the current position of the axis to the previous position when the motion system power is turned ON again or reset or when the slave device is connected.

Whether restoration is carried out with the maintained absolute position data or not can be specified in the absolute position control setting of the axis parameter. For details, refer to the following.

☞ Page 100 Current position restoration

Relevant variables

Variable/Structure name	Name	Details
AxisName.PrConst.		
PosRestoration_AbsPosEnable	Absolute position management setting	Sets the absolute position control setting. 0: ABSDisabled (Disable absolute position system) 1: Enabled (Enable absolute position system) -1: Auto Automatic setting (Acquire from connected device)
PosRestoration_AbsPosBase	Absolute position reference setting	Sets the current position to be the basis at the current position restoration in order to use the absolute position system. 3: FeedMachinePosition (Feed machine position)
AxisName.Pr.		
Homing_Required	Homing required or not	Sets FALSE when the homing is not required. If FALSE is set, the homing request will not turn TRUE at the current position restoration. FALSE: Homing is unnecessary TRUE: Homing is necessary
AxisName.Md.		
Homing_Required	Homing required or not	Displays whether the axis requires the homing or not. If FALSE is set, the homing request will not turn TRUE at the current position restoration. FALSE: Homing is unnecessary TRUE: Homing is necessary
PosRestoration_Status	Current position restoration status	Displays the current position restoration status. 0: NotExecute (Not executed) 1: WaitingRequest (Waiting restoration request) 2: RestoredInIncSystem (Restoration completed by incremental system) 3: RestoredInAbsSystemUnHomed (Restored in absolute position system (Homing uncompleted)) 4: RestoredInAbsSystem (Restored in absolute position system) "0: NotExecute" is displayed when the slave device is disconnected.
Homing_Request	Homing request	The timing of turning TRUE differs depending on the absolute position control setting. For details, refer to the following. ☞ Page 102 Homing request Turns FALSE when the machine homing is completed.
SetPosition	Set position	Stores the current position address. • The set position is a value which rounds the cumulative current value with the ring counter range.
CumulativePosition	Cumulative current position	Stores the current position address. • The address is changed to the current position change value when the current position is changed.
FeedMachinePosition	Feed machine position	Stores the current position address by the machine coordinate. • It is the cumulative current value from the homing completion position. • The address is not change even though the current position is changed.
System.Md.		
BackupRestoreStatus	Backup restore information	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
System.Cd.		
BackupRestore	Backup restore request	Executes backup stop and restored data saving. 0H: No request ABCDH: Backup stop request FFFEH: Restored data saving request

Current position restoration

Specify whether the absolute position system is used for the current position restoration or not in the absolute position control setting.

Absolute position control setting	Setting value	Details
Disable absolute position system	0: ABSDisabled	☞ Page 100 Incremental system
Enable absolute position system	1: Enabled	☞ Page 101 Absolute position system
Automatic setting (Acquire from connected device)	-1: Auto	☞ Page 101 Connected device setting

"-1: Auto" can be set when the connected device has the ENCODER_ABS_STATUS object.
When "-1: Auto" is not set, the connected device side setting and the motion system side setting must be matched. When those are mismatched, normal current position control cannot be executed. (If the connected device has the ENCODER_ABS_STATUS object and the settings are mismatched, the error "Absolute Position Management Setting Mismatch" (error code: 1A9AH) will occur at the current position restoration.)

• Setting example

When the absolute position system is used, select "1: Enabled (absolute position detection system)" for "absolute position detection system selection" (PA03.0) of the servo parameter (basic parameter) in MR-J5(W)-G. In the motion system, select "1: Use the absolute position system" for "absolute position control setting" in the real drive axis which uses MR-J5(W)-G as the connected device.

The current position restoration timing differs depending on the axis type. For details, refer to the following.

☞ Page 105 Current position restoration for each axis type

Point

Whether "incremental system" or "absolute position system" was used for the current position restoration is recorded in the event history.

Incremental system

This system restores the current position without using the absolute position data.

When the current position restoration has been completed, the axis current position (cumulative current position and feed machine position) becomes "0", and the homing request turns TRUE. (The set position may not become "0" depending on the ring counter setting. (☞ Page 83 Positioning Range))

When restoration is completed, the axis current position restoration status is "2: RestoredInIncSystem".

If the machine position establishment with the slave device is not required, set FALSE in the homing request setting. This prevents the homing request from turning TRUE at the restoration.

The absolute position data backup is not executed.

Absolute position system

This system restores the current position using the absolute position data.
At the current position restoration, the axis current position (set position, cumulative current position, and feed machine position) is restored based on backed up absolute position data. When restoration is completed, the axis current position restoration status becomes "4: RestoredInAbsSystem".
However, if the warning "Absolute Position Data Incorrect Warning" (warning code: 0D00H) has been detected at restoration, the axis current position restoration status becomes "3: RestoredInAbsSystemUnHomed". The current position is restored with the same content as the incremental system and the homing request turns TRUE. (The absolute position data is maintained.) At this time, the absolute position system is not enabled until homing is completed. Be sure to execute homing. (When homing has been normally completed, the axis current position restoration status becomes "4: RestoredInAbsSystem".)
When the absolute position system is used, the restoration content differs depending on the axis type. For details, refer to the following.
☞ Page 105 Current position restoration for each axis type
The warning "Absolute Position Data Incorrect Warning" (warning code: 0D00H) occurs in the following case.

- The absolute position data is illegal at the current position restoration by the following causes

  - The axis type has been changed. (The axis type of the absolute position data does not match the axis variable.)
  - The absolute position data is deleted by the memory error, etc.
  - The homing request has been TRUE at backup.
  - The axis driver unit conversion (numerator/denominator) is changed.

- Even though the machine homing has been started, it has not been normally completed.

[Only for real drive axes]

- "Absolute position erased" is detected in the driver side.
- "Polarity (607EH)" b7: position polarity of the driver object is changed.
- "HomeOffset (607CH)" of the driver object is changed.
- The connected driver device is changed.
- The encoder resolution of the driver device is changed.

[Only for real encoder axes (via drive unit)]

- "Absolute position erased" is detected in the scale measurement encoder.
- The encoder resolution of the scale measurement encoder is changed.

[Only for virtual encoder axes]

- The upper limit value/the lower limit value of the encorder ring counter is changed.

Connected device setting

The absolute position system setting is taken from the connected device with the ENCODER_ABS_STATUS object by the SDO communication, and the current position is restored.
Set the absolute position system setting in the connected device side.
If the connected device does not have the ENCODER_ABS_STATUS object, it operates as the incremental system without an error.

Homing request

The homing request will be TRUE in the following cases.

Point

The reason why the homing request became TRUE is recorded in event history.

Absolute position control setting[*1]	Cause		Detection timing[*2]
Common	The system power is turned ON or reset.[*3]		At the system power ON or after reset
	The axis type is changed.		At the current position restoration
	Homing is started. (The homing request flag will not become FALSE unless homing is normally completed.)		At homing start
	The axis driver unit conversion (numerator/denominator) is changed.		At current position restoration At the PLC READY [Y0] ON
	The driver power turns on.[*3] [The real drive axis only]		At current position restoration
Not use the absolute position system	The current position restoration is executed.		At the current position restoration
Use the absolute position system	Since the absolute position data has been erased by the memory error, etc..		At the current position restoration
	Change of the driver and the motor encoder is detected.		
	Only for real drive axes	"Polarity (607EH)" b7: position polarity of the driver object is changed.	
		An electric gear of the driver side is changed.	
		"Absolute position erased" is detected in the driver side.	At the current position restoration and cause occuring
	Only for real encoder axes (via drive unit)	"Absolute position erased" is detected by the scale measurement encoder.	At current position restoration
		The encoder resolution of the scale measurement encoder is changed.	
	Only for virtual encoder axes	The upper limit value/lower limit value of the encorder ring counter is changed.	

*1 For the "-1: Auto" setting, refer based on the connected device setting.

*2 When the homing required setting is FALSE, the homing request will not become TRUE even though the cause is occurring at the current position restoration.

*3 When the system/driver power is OFF when "1: Enabled" is set and the homing is completed at resetting, the homing request does not turn TRUE.

Absolute position data backup

When the absolute position system is used, the absolute position data of each axis is backed up. The absolute position data is backed up linked to the axis variable and the axis type. At the current position restoration, if the setting values do not match with the axis variable and the axis type of the absolute position data, the absolute position data will be discarded. (The warning "Absolute Position Data Incorrect Warning" (warning code: 0D00H) occurs.)
Backup is executed while the homing request is FALSE. (When the homing request becomes TRUE, the absolute position data is discarded.)

- Storage folder and file name

Storage folder: /Ich
File name: abs_axis.bin

- Memory capacity

Memory capacity of the absolute position data differs depending on the axis type. Specify memory capacity of the absolute position data according to the axis type and the number of axes.
Memory capacity per one axis for each axis type is shown below.

Axis type	Memory capacity [byte/axis]
Real drive axis	220
Real encoder axis	170
Virtual drive axis	90
Virtual encoder axis	170
Virtual linked axis	90

Memory capacity used for the absolute position data can be specified in System.PrConst.AddonAbsSystem.BackupRamSizeMax.
If the file size required for backup exceeds specified memory capacity when the current position value of each axis is restored, the error "Insufficient Current Position Backup Capacity" (error code: 1ADFH) occurs and the target axis is not restored completely. Reconsider the memory capacity and carry out software reboot.

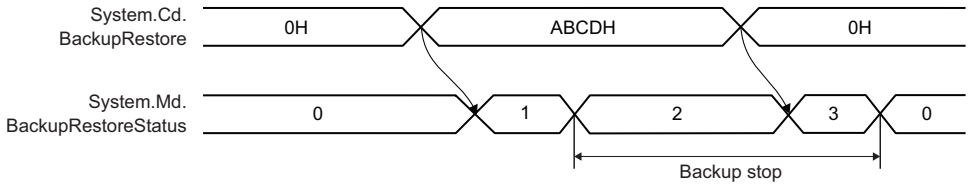
Saving (backup) and restoration (restored) of the absolute position data

The absolute position data maintained inside the motion system can be data saved/restored by saving/restoring functions of the engineering tool, etc. When exchanging the Motion module, restarting time can be shortened by saving/restoring not only programs and parameters but also the absolute position data.
During backup or restoration, the motion system stops backing up the absolute position data. Do not turn OFF the power of the motion system during backup or restoration.
After restring is completed, software reboot is required to enable the restored absolute position data. (The backup is at a standstill and it can not be cleared until software reboot is carried out.)
For real drive axis, the homing request becomes TRUE when the restored absolute position data is different from HomeOffset (home position), the device, and the resolution of the connected driver.

Ex.

When saving absolute position data to an SD memory card

1. Set "ABCDH: backup stop request" in System.Cd.BackupRestore.
2. Confirm System.Md.BackupRestoreStatus changes to "1: backup stop executed" and "2: backup stop".
3. Transfer absolute position data "/lch/axis_abs.bin" to an arbitrary folder in an SD memory card by using file transfer (Execution data backup).
4. Set "0H: No request" to System.Cd.BackupRestore after confirming file transfer (Execution data backup).
5. Confirm System.Md.BackupRestoreStatus changes to "3: backup stop cleared" and "0: backup restore not executed".



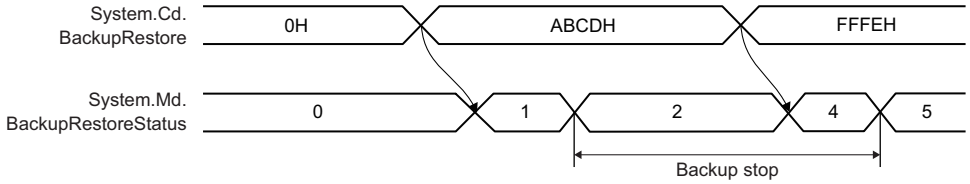
Precautions

The current positon of moved axes after backup stop is not saved.
To execute backup after confirming all axes stop is recommended.

Ex.

When restoring absolute position data saved in an SD memory card

1. Set "ABCDH: backup stop request" to System.Cd.BackupRestore.
2. Confirm System.Md.BackupRestoreStatus changes to "1: backup stop executed" and "2: backup stop".
3. Transfer the "axis_abs.bin" file backuped in an arbitrary folder in an SD memory card by by using file transfer (Execution data backup).
4. Set "FFFEH: Restored data saving request" to "System.Cd.BackupRestore" after confirming file transfer (Execution data backup).
5. Confirm System.Md.BackupRestoreStatus changes to "4: Restored data saving" and "5: Restored data saving completed".
6. Carry out software reboot and enable restored data.



Precautions

After saved data required, the absolute position data of motion system may break if software reboot is carried out or power is supplied again before System.Md.BackupRestoreStatus changes to "5: Restored data saving completed". Carry out restoration again.

Current position restoration for each axis type

This section describes the current position restoration for each axis type.

Real drive axis

Each current position is restored as the following.

Current position	Incremental system	Absolute position system
		PosRestoration_AbsPosBase is "3: FeedMachinePosition"
Feed machine value	Restored in "0".	Differs depending on the restoration method. For details, refer to the following. ☞ Page 98 Absolute position system
Cumulative current position	The same value as restored feed machine position.	
Set position	A value obtained by rounding the cumulative current value with the ring counter range.	

■Current position restoration timing

When an object that constructs the absolute position system to execute the 64-bit current position restoration is required, restoration is automatically executed at the initial communication with the slave device. For details on the absolute position system to execute the 64-bit current position restoration, refer to the following.

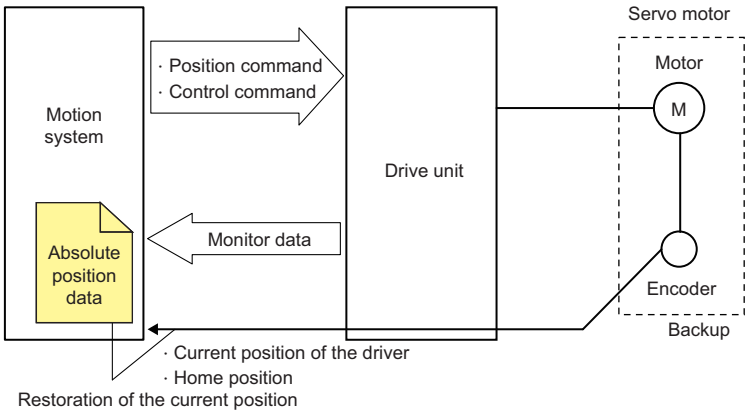
☞ Page 98 Absolute position system

The servo ON status is not established until the current position restoration is completed.

■Absolute position system

This section describes precautions for constructing the absolute position system in a real drive axis.

The following shows the absolute position system configuration when the drive unit is used as the servo amplifier.



A drive unit/servo motor which can detect the absolute position must be used to construct the absolute position system.

For details on the absolute position system setting of the drive unit side, refer to each drive unit manual.

The current position restoration of the absolute position system can be classified into the following three methods.

Current position restoration method	Basis position	Description	Remark
32-bit restoration (Driver current position restoration)	Home position	Correctly executes the current position restoration within the signed 32-bit integral number range [driver-based] from the reference position. Restores based on the driver current position (Position actual value) by using a part of the backup data of the motion system.	—
64-bit restoration (Driver ABS counter position restoration)	Backup position	Correctly executes the current position restoration within up to the signed 64-bit integral number range*1 [driver-based] from the reference position. Restores based on the driver movement amount by using backup data of the motion system.	Some objects are required to execute 64-bit restoration. If the object are not set, 32-bit current position restoration will be executed.
	Home position	Correctly executes the current position restoration within up to the signed 64-bit integral number range*1 [driver-based] from the reference position. Restores based on the driver home position and the position at restoration by using a part of the backup data of the motion system.	Restoration is executed based on the home position when the connected slave device has been changed.

*1 Actual restoration possible range is - ((multiple revolution counter maximum value) / 2 × (encoder resolution)) to ((multiple revolution counter maximum value) / 2 × (encoder resolution) - 1).

Unit of the restoration possible range from the reference position in 64-bit restoration

- Driver-based or motor encoder-based
- For the motor encoder-based unit, the driver side additional electric gear (Gear Ratio) is required for the Motion module to execute the current position restoration. The restoration possible range is (the driver-based unit range) × (Gear Ratio).

Point

In the real drive axis, which method was used for the current position restoration of the absolute position system, "32-bit restoration, 64-bit restoration (backup position), or 64-bit restoration (home position)" is recorded in the event history.

The following shows objects required to construct the absolute position system of 64-bit restoration.

Setting item AxisName.PrConst.SlaveObject(STRUCT(SLAVE_OBJECT_REAL))	Setting description
vHomeCycleCounter (HomeCycleCounter)	Specifies the object ID storing the encoder positon within one revolution at homing.
vHomeAbsCounter (HomeAbsCounter)	Specifies the object ID storing the encoder multiple revolution counter at homing.
vInitialPos (InitialPos)	Specifies the object ID storing Position actual value at connection.
vInitialCycleCounter (InitialCycleCounter)	Specifies the object ID storing the encoder positon within one revolution at connection.
vInitialAbsCounter (InitialAbsCounter)	Specifies the object ID storing the encoder multiple revolution counter at connection.
vMaxAbsCounter (MaxAbsCounter)	Specifies the object ID storing the maximum value of the multiple revolution counter.

If any one of the above parameters is not specified, 32-bit restoration is executed.

Precautions

Even if the above parameters have been specified, an error "Driver Restoration Data Incorrect" (error code: 1AE0H) will occur and the current position restoration may not be completed. Execute reconnecting after correcting the parameter of the drive unit.

[When using MR-J5(W)-G]

The following explains the settings and the current position restoration method when the absolute position system is constructed using MR-J5(W)-G.

- MR-J5(W)-G setting
  Select "1: Enabled (absolute position detection system)" in "Absolute position detection system selection" (PA03.0) of the servo parameter (basic setting). In addition, select "0: Disabled" in "Absolute position counter warning (AL.E3)" selection (PC29.5) of the servo parameter (extension setting).
  Depending on the "Electronic gear numerator (PA06)" and "Electronic gear denominator (PA07)" setting, the error "Driver Restoration Data Incorrect" (error code: 1AE0H) may occur. Correct the following.
  When the resolution of the connected encoder is 2 to the power of n, set the electronic gear of MR-J5(W)-G to be "1 : 1" or multiple of 2 to the power of n.
  When the resolution of the connected encoder is other than 2 to the power of n, set the electronic gear of MR-J5(W)-G to "1 : 1".
- The motion system setting
  Select "-1: Auto" or "1: Enabled" in "Absolute position control setting" in the real drive axis whose connected device is MR-J5(W)-G.
- The current position restoration method
  Execute the 64-bit current position restoration.

Real encoder axis

Each current position is restored as follows.

Current position	Incremental system	Absolute position system
Cumulative current position	Restored in "0".	Restores based on the synchronous encoder movement amount by using backup data of the motion system.
Set position	A value which rounds the restored cumulative current position with the ring counter.	

The restoration method for the absolute position system is depending on "the real encoder axis type" of the axis parameter shown below.

Real encoder axis type	Current position restoration method	Description	Remark
Via drive unit	64-bit current position restoration	Correctly executes the current position restoration within up to the signed 64-bit integral number range*1 [encoder pulse-based] from the backup position.	Only the synchronous encoder connected to the drive unit can be used. For details on the setting, refer to the following. ☞ Page 36 Axis

*1 The actual restoration possible range is - ((multiple revolution counter maximum value) / 2 x (encoder resolution)) to ((multiple revolution counter maximum value) / 2 x (encoder resolution -1).

■Current position restoration timing

Current position restoration is executed after the real drive axis of the same station address is connected.

Virtual drive axis

Each current position is restored as follows.

Current position	Incremental system	Absolute position system
Cumulative current position	Restored in "0".	Restored in the backup data value.
Set position	A value which rounds the restored cumulative current value with the ring counter.	

■Current position restoration timing

Current position restoration is executed at the axis variable initialization. For details on the axis variable initialization timing, refer to the following.

☞ Page 38 Axis variable initialization timing

Virtual encoder axis

Each current position is restored as follows.

Current position	Incremental system	Absolute position system
Cumulative current position	Restored in "0".	Restores the current position based on the movement amount from the backup data to the position at restoration. Correctly executes the current position restoration within the signed 32-bit integral number range [encoder pulse unit] from the backup position.
Set position	A value which rounds the restored cumulative current position with the ring counter.	

■Current position restoration timing

Executes the current position restoration as AxisName.Cd.Encoder_Connect turns TRUE.

Virtual linked axis

Each current position is restored as follows.

Current position	Incremental system	Absolute position system
Cumulative current position	Restored in "0".	Restored in the backup data value.
Set position	A value which rounds the restored cumulative current value with the ring counter.	

■Current position restoration timing

Current position restoration is executed at the axis variable initialization. For details on the axis variable initialization timing, refer to the following.

☞ Page 38 Axis variable initialization timing

Precautions

- In the absolute position control setting, when the connected device side setting and the motion system side setting are mismatched, the absolute position control may not be executed correctly. (If there is the ENCODER_ABS_STATUS object in the connected device, the error "Absolute Position Management Setting Mismatch" (error code: 1A9AH) will occur at current position restoration.)
- If TRUE is set in the homing required setting, address information stored in the motion system is not guaranteed.
- During backup/restoration by the engineering tool, do not write in System.Cd.BackupRestore.

3 BASIC FUNCTIONS

3.1 Operation Cycle

In the motion system, operation processing related to the motion control is performed in the fixed cycle (operation cycle). This section describes operation cycles which can be set in the motion system.

Relevant variables

Variable/Structure name		Name	Details
System.PrConst.			
	OperationCycle[1..1].Cycle	Operation cycle setting	Sets the first operation cycle. The following values can be set. 0: Synchronizes with the basic cycle of network -11: Synchronizes with the basic cycle of network (same as "0") -12: Synchronizes with the middle speed cycle of network -13: Synchronizes with the low speed cycle of network -100256 to -100001: Multiplication by n of the system basic processing (1 to 256)
	BuffermemoryRefreshCycle.Cycle	Buffer memory refresh cycle setting	Sets the buffer memory refresh cycle. The following values can be set. 0: The nth operation cycle ("n" is the slowest operation cycle among operation cycles whose setting is enabled.) -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 -118192 to -110001: Multiplication by n of the first operation cycle (1 to 8192)
	Link_MotionStationRefreshType	Refresh method of send/receive data at motion control station	Sets the send/receive data refresh method of the motion syncronization station.
	FastOperationMode*1	High-speed mode setting	Switch the operation process of the entire system to high-speed mode. 50EFH: High-speed mode Other than the above: Normal mode
AxisName.PrConst.			
	OperationCycle	Control cycle setting	0: Operates at the first operation cycle
	FastOperationMode*1	High-speed mode setting	Switch the operation process of the set axis to high-speed mode.*2 5FE2H: High-speed mode Other than the above: Normal mode
System.Md.			
	SystemBaseCycle_Counter	System basic cycle counter	Counts up every time when executing system basic processing.

*1 In order to switch to the high-speed mode, set both the system parameter and axis parameter to the high-speed mode. If both system parameter and axis parameter are not set to the high-speed mode, it is not switched to the high-speed mode.

*2 Only the real drive axis is supported.

System basic cycle

The system basic cycle is a basic cycle of the motion control such as operation cycle processing. System basic cycle settings are shown below.

System basic cycle is the same as the first operation cycle.

Operation cycle settings

Operation cycle is set in System.PrConst.OperationCycle[1..1].

"Synchronize with the network setting" must be set to control real axes connected to the network.

The following table shows a reference of the operation cycle settings.

Operation cycle	RD78G4	RD78G8	RD78G16	RD78G32	RD78G64	RD78GHV	RD78GHW
31.25 μs	—					—*1	
62.5 μs	1 axis					2 axes*2	
125 μs	4 axes					14 axes	
250 μs	4 axes	8 axes	14 axes			20 axes	
500 μs	4 axes	8 axes	16 axes	20 axes		32 axes	
1.0 ms	4 axes	8 axes	16 axes	32 axes		64 axes	
2.0 ms	4 axes	8 axes	16 axes	32 axes	64 axes	100 axes	
4.0 ms	4 axes	8 axes	16 axes	32 axes	64 axes	128 axes	
8.0 ms	4 axes	8 axes	16 axes	32 axes	64 axes	128 axes	256 axes

*1 Up to 2 axes can be connected by using the high-speed mode.

*2 Up to 4 axes can be connected by using the high-speed mode.

Operation cycle assignment

An operation cycle of the axis can be set by AxisName.PrConst.OperationCycle.

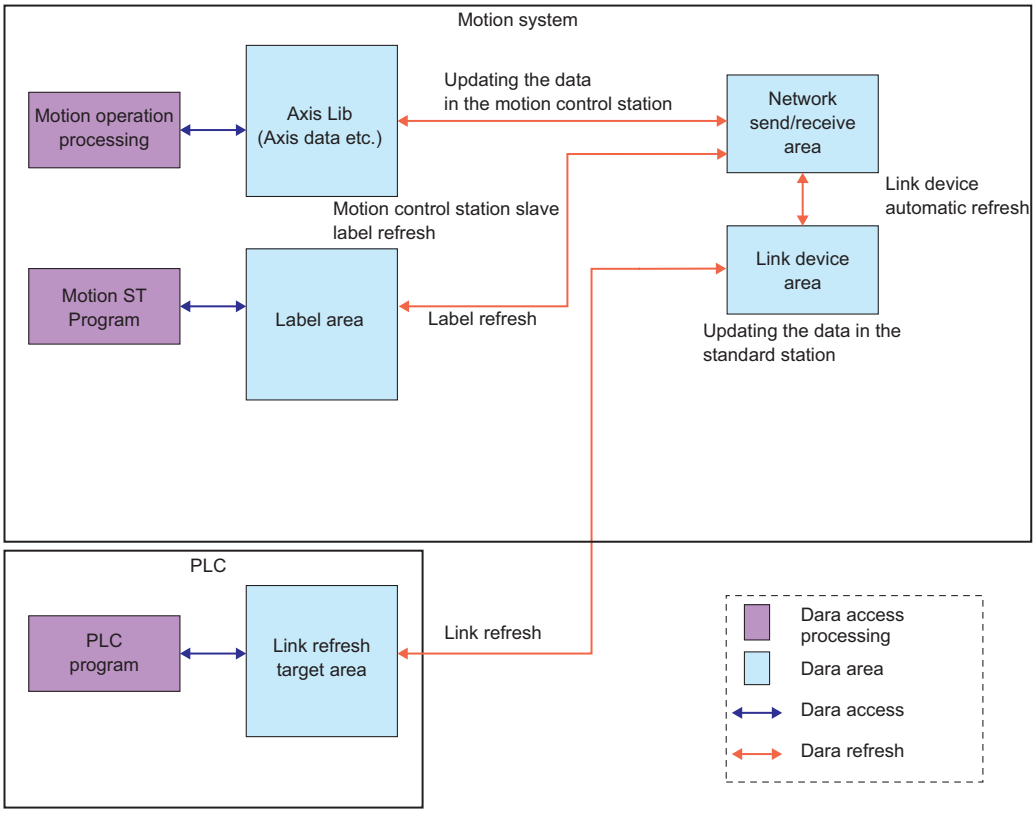
Send/Receive data refresh of network connected devices

Network connected devices are distinguished between stations controlled by the motion system and standard stations, and those data refresh methods are different.

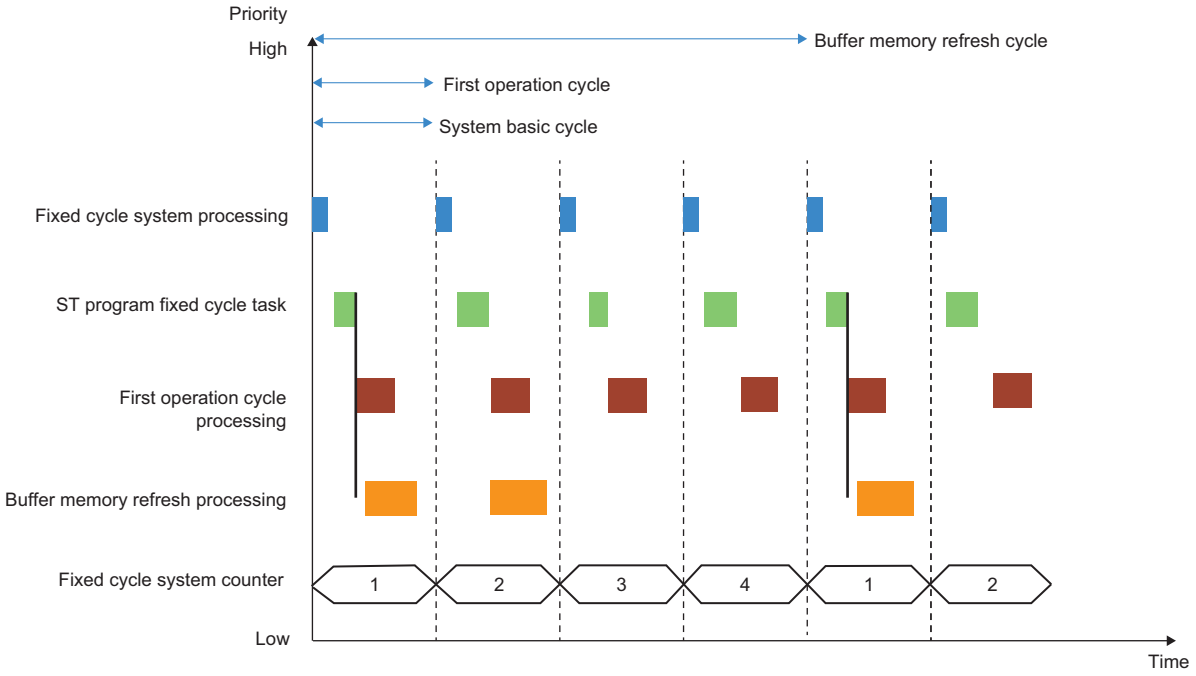
Station	Explanation
Motion syncronization station	• The motion system refreshes data by directly synchronizing the network send/receive with the operation cycle. • The target link device is not refreshed.
Standard station*1	• Stations other than motion syncronization stations are standard stations. • The programmable controller refreshes data. • The link device is refreshed by synchronizing with the programmable controller.

*1 Control by the programmable controller

The following shows the relation of data areas and the refresh.



The internal process timing of the motion system is shown below.



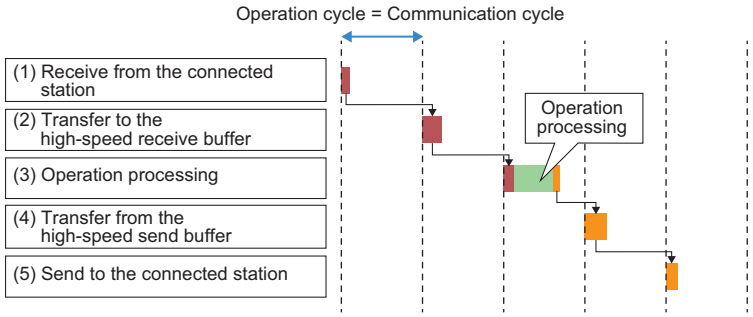
Motion synchronization station

Select a data refresh method from the following for the station which executes the motion control as a real axis. It can be set in System.PrConst.Link_MotionStationRefreshType. It is common in all of the operation cycles.

Motion synchronization station send/receive data refresh method	Description
Operation cycle precedence method	• Data receive, motion operation, and data send are executed in different operation cycles. A high-speed operation cycle can be selected when the motion operation load is high, because these are executed in parallel. • Since the data refresh timing is slower, the responsiveness to feedback is low.

■Operation cycle precedence method

With operation cycle precedence method, data receive, motion operation, and data send are executed in different operation cycles. For the connected device, it takes five cycles to receive a feedback to the data sent from the connected device.



■Slave label

The I/O data which the slave device of motion syncronization stations exchanges with motion system by cyclic communication registers as labels and enables read/write. Refreshing slave labels are carried out in operation cycle. For the using and setting methods of slave label, refer to the following.

☞ Page 728 Connectable device to CC-Link IE TSN

Standard station

The following stations update the data.

- Programmable controller

For details, refer to "Network Synchronous Communication" in the following manual.

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

Motion synchronization station link device auto refresh

Read data (RX, RWr)

Received data is refreshed to the link device. For details, refer to "Cyclic Transmission" in the following manual.

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

Write data (RY, RWw)

Write data to send is refreshed to the link device. It is executed in accordance with the refreshing timing of read data. It is refreshed in the next cycle after the cycle in which the data has been sent to the network.

This function monitors send data, but it cannot write send data.

Buffer memory refresh

Data which are used to issue the MCFB by the following items and to get axis monitor information, etc. are refreshed in the buffer memory refresh cycle.

- Programmable controller

The target of the buffer memory refresh is as follows.

- Area for MCFB
- Area for the public labels (module labels)

Buffer memory refresh cycle

Data used between the motion system and the programmable controller is refreshed in a buffer memory refresh cycle.

This function is set in System.PrConst.BuffermemoryRefreshCycle.

Axis operation process high-speed mode

The following shows the axis operation process high-speed mode.
If it is switched to the high-speed mode, some of the functions will be restricted, but it may be able to communicate with shorter operation cycle.

Axis operation process high-speed mode restrictions

In the high-speed mode, compared to the normal mode, it has the following function restrictions.

• Function restrictions of the motion system

No.	Restriction	Detail
1	Slave object unsettable	Operates by fixed initial setting ([OBJ] setting).
2	PDO mapping cannot be changed	Operates by fixed high-speed mode mapping. Cannot delete, add nor change the mapping objects.
3	A part of axis monitor unusable	The following monitor stops the updates. The slave feedback can be confirmed by using the device labels. Use depending on the system load status. • AxisName.Md.SetAcceleration • AxisName.Md.ActualVelocity • AxisName.Md.ActualPosition (For real encoder axis) • AxisName.Md.Io_TargetPos • AxisName.Md.Io_PosActualValue • AxisName.Md.Io_VelActualValue • AxisName.Md.Io_TargetVelocity • AxisName.Md.Io_TorqueActualValue • AxisName.Md.Io_Statusword • AxisName.Md.DriverError • AxisName.Md.DriverErrorID • AxisName.Md.DriverErrorDetailID • AxisName.Md.VelocityOverride • AxisName.Md.AccelerationOverride • AxisName.Md.JerkOverride • AxisName.Md.TorqueLimit_Positive • AxisName.Md.TorqueLimit_Negative • AxisName.Md.Driver_ReadyOn • AxisName.Md.Driver_ServoOn
4	A part of axis command unusable	The input of the following command will be ignored. • AxisName.Cd.Homing_ClearRequest • AxisName.Cd.TorqueLimit_Positive • AxisName.Cd.TorqueLimit_Negative • AxisName.Cd.AccelerationOverride • AxisName.Cd.VelocityOverride Even if the FB is executed to change the command to be ignored, the change is not carried out.
5	Synchronous control previously used the operation cycle data is unusable	Synchronous control previously used the operation cycle data cannot be used. Use the latest operation cycle data.

• Function restrictions when combined with MR Configurator2

No.	Restriction	Detail
1	Multiple axes graph unusable	Multiple axes function cannot be used. Do not waveform measure including the high-speed mode axis.

Precautions

In the high speed-mode, only the position control can be executed.
If used by direct control (velocity control, torque control), the error "Necessary Slave Object Unset" (error code: 1AA8H) occurs.
Note the following when using the high-speed mode and sift through the function to be used depending on the system load status.

- The operation time increases if the labels are externally exposed, so when using by high-speed mode, set external exposion disabled.
- When using the synchronous control, use it depending on the system load status.
- Executing the ST program in fixed-cycle will increase the operation time, so when using by high-speed mode, set the ST program execution cycle depending on the system load status.

Note the following and adjust the parameter setting value depending on the system load status.

- The operation time increases at the cycle over warning output, so set System.PrConst.OperationCycle[1…].NumOfCycleOverWngDetectTimes to 0, and set the warning "Cycle Over Warning" (warning code: 0F08H) to be not output.
- User needs to confirm the normal operation range of System.PrConst.OperationCycle[1…].NumOfCycleOverErrDetectTimes, and change the value to not be the error "Cycle Over" (error code: 1C80H) and the error "Cycle Over" (error code: 320CH).

Operation cycle 31.25 setting [RD78GH]

By using RD78GH enables to communicate by operation cycle 31.25 μs setting.
The procedure to operate by 31.25 μs is shown below.

Procedure to operate by 31.25 μs

• Enabling add-on

When using by 31.25 μs, start from the minimum configuration and add required add-on depending on the system load status.
The add-ons assumed as minimum configuration are shown below.
Enabled: ○, Disabled: ×

No.	Add-on name	Enabled/Disabled
1	baseSystem	○
2	AbsSystem	×
3	Axis	○
4	ExternalSignal	×
5	FileTransfer	×
6	Logging	×
7	MotionEngine	○
8	MotionControl_AxisFilter	○
9	MotionControl_General	○
10	MotionControl_Sync	○
11	MotionEventHist	×
12	NetworkDriver_CCIETSN	○
13	PackagingApp	×
14	PlcInstruction	○
15	ProfileControl	×
16	Program_ST	○
17	ServoDriver_CANopen	○
18	SignalIO	○
19	ServoSystemRecorder	×

• Parameter setting

Set the axis to be operated by 31.25 μs to the high-speed mode. For details, refer to the following.
☞ Page 109 System basic cycle

Point

The system load status can be confirmed by System.Md.OperationCycle[1...].ProcessingTime.
Do not let System.Md.OperationCycle[1...].ProcessingTime keep exceeding the setting operation cycle.

Precautions

- By minimum configuration add-on, the forced stop command cannot be input from the Motion module.
  Depending on the system load status, adding the add-on ExternalSignal make it usuable. If the operation cycle exceeds by adding this add-on, take an alternative measure such as using the forced stop input from the servo amplifier, not the forced stop command from the Motion module.
- By minimum configuration add-on, the absolute position sytem cannot be used. Depending on the system load status, adding add-on AbsSystem make it usable.
- By minimum configuration add-on, the data logging function cannot be used. Depending on the system load status, adding the add-on Logging make it usable.
- The operation time increases when the number of device labels increases. When the number of axis is 1-axis and using the device labels, adjust the number of device labels depending on the system load status. When the number of axis is 2-axes, do not use the device labels. The operation cycle exceeds at start of function block, but soon the operation time settles within the operation cycle thus the motion control is not affected.

Setting example

The setting examples for operations under the 2-axis interpolation control in the operation cycle 31.25 μs are shown below.

■Configuration

- Real drive axis: MR-J5-G, 2-axes
- Axes group setting: 1
- Device label: No setting

■Parameter setting

- System setting

Parameter name		Setting value
Buffer memory refresh cycle setting		—
	Number of cycle over warning detections	0
	Number of cycle over error detections	20
High-speed mode setting		50EFH
Operation cycle setting		—
	Number of cycle over warning detections	0
	Number of cycle over error detections	20

For the seting value of parameters other than the above, the initial value is applied.

- Axis parameter

Parameter name	Seting value
Absolute position control setting	0: ABSDisabled (Disable absolute position system)
High-speed mode setting	5FE2H
Command in-position width	0.0

For the setting value of parameters other than the above, the initial value is applied.

- Axes group parameter

For the setting value of parameters other than configuration axis, the initial value is applied.

■Program

Executes MCv_AllPower, MC_GroupEnable, and MCv_MoveLinearInterpolateRelative by normal execution program of the motion ST program.

3.2 Add-on Function

This section describes add-on Function.

The motion system functions are configured with basic functions provided by the motion system software and functions provided by the add-on library. The motion system functions can be expanded by installing add-on libraries.

For details on the standard function add-on library, refer to the following.

☞ Page 723 List of Add-on Library

Operation of this function for each system status

○: Possible

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

Relevant variables

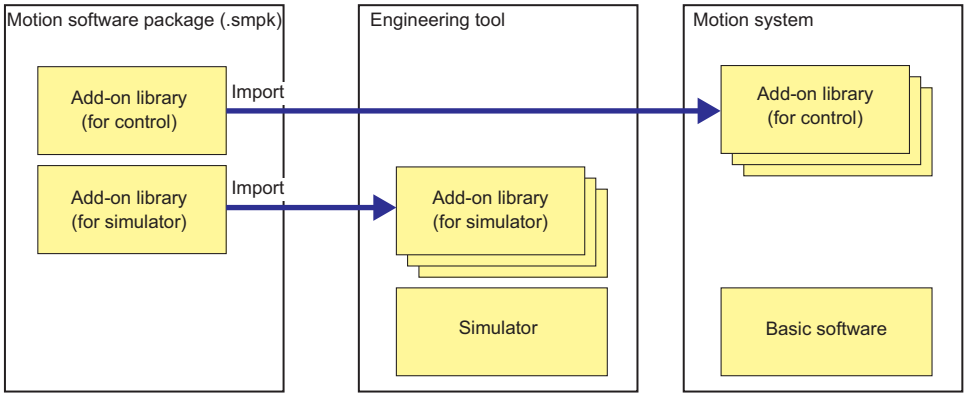
The following shows common variables in each add-on library. Available variables other than the following are different depending on the add-on library. For details, refer to the specifications of each add-on library.

Variable/Structure name		Name	Details
System.PrConst.			
	Addon_AddonLibraryName	Add-on <AddonLibraryName> parameter	ADDON_PARAM structure For add-on used for <AddonLibraryName>, refer to the following.
System.Md.			
	Addon_AddonLibraryName	Add-on <AddonLibraryName> monitor	ADDON_MON structure For add-on used for <AddonLibraryName>, refer to the following.
ADDON_PARAM			
	RamSizeMax	Maximum RAM size	Specifies the maximum amount of system memory (RAM) used in the add-on in K bytes. For details on the setting method, refer to the following. ☞ Page 121 System Memory Settings
	BackupRamSizeMax	Maximum backup RAM size	Specifies the maximum amount of backup RAM used in the add-on in K bytes. For details on the setting method, refer to the following. ☞ Page 121 System Memory Settings
ADDON_MONI			
	RamUsage	RAM usage	Stores the current amount of system memory (RAM) used in the add-on in K bytes.
	RamMaxUsage	RAM usage maximum	Stores the maximum amount of system memory (RAM) used in the add-on in K bytes.
	BackupRamUsage	Backup RAM usage	Stores the current amount of backup used in the add-on in K bytes.
	BackupRamMaxUsage	Backup RAM usage maximum	Stores the maximum amount of backup used in the add-on in K bytes.
	Version	Version	Stores the version information of the add-on.

Add-on library configuration

File configuration

Add-on libraries are included in the motion software package and controlled by the engineering tool. Add-on library functions can be used by installing add-on libraries in the engineering tool and the motion system.



Installation file configuration

The installation file of the software is one file.

For details, refer to the following.

☞ Page 626 Motion software package file configuration details

Provision of installation file

The installation file is provided in the MITSUBISHI ELECTRIC FA Global Website, etc.

The installation file is common between RD78G and RD78GH. (The both BootRoms are packed.)

A motion software package includes the simulator module for the engineering tool. It can simulate the functions including add-on library functions by importing it to the engineering tool. For details on configuration of the motion software package, refer to the following.

☞ Page 626 Software file configuration of motion system

Functions configuration

Depending on the add-on library, one of the following lists or all of those are expanded.

- Axis label, system label, or control data of variables for each function (Pr./Md./Cd.)
- Function block (for the CPU module)
- Function block (for internal ST programs)

Management of add-on library

Installation/Uninstallation

Add-on libraries are installed in the system folder (/sys) of the motion system. The motion system recognizes direct add-on libraries of the system folder (/sys).

For installation of add-on libraries, refer to the following.

☞ Page 625 Motion System Software Installation

For the uninstallation of add-on libraries, refer to Help of the engineering tool.

Enabling/Disabling

Functions which are infrequently used, such as adjustment functions used for starting up the system and the functions for debugging used only when an error occurs, can be disabled and enabled when needed. However, unlike uninstallation, the occupied capacity of the system folder is not reduced.

The disabled add-on library is moved to /sys/disabled.

Enabling/Disabling status is reflected when the motion system is initialized next time.

Amount of memory consumed setting

Amount of memory each add-on library can consume can be changed by the parameter, and the amount of memory between each function can be adjusted flexibly according to the application.

For details on the setting method, refer to the following.

☞ Page 121 System Memory Settings

Examples of adjustable memory:

- Buffer size available in logging
- Operation profile open area size
- Label area size
- ST program area

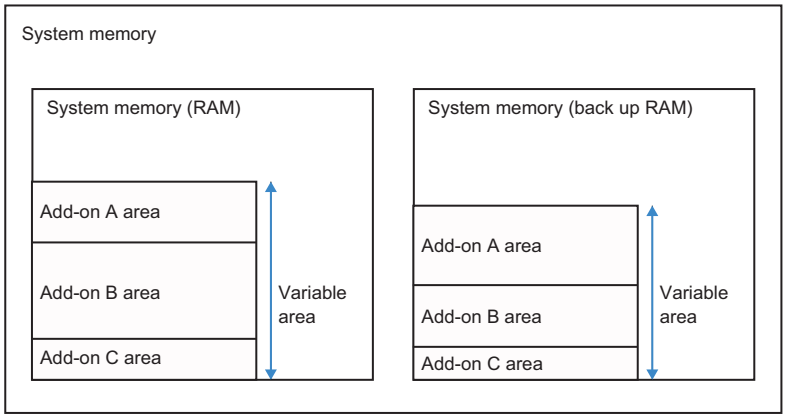
Loading add-on library

Add-on libraries are loaded when the motion system is initialized. When the loading can not be performed because of the followings, the error "Add-on Library Load Error" (error code: 3205H) occurs and the motion system does not turn to RUN.

- The file of add-on library is broken.
- The add-on library with dependencies has not been installed.
- The add-on library that cannot be combined has been installed.
- Inconsistency occurred in the versions of the add-on libraries.
- Inconsistency occurred in the versions of the add-on library and the software of the module.

3.3 System Memory Settings

This section describes setting methods of memory size used in the add-on library in the system memory (RAM) and the system memory (backup RAM).



Relevant variables

The following shows common variables in each add-on library.
Available variables other than the following are different depending on the add-on library.

Variable/Structure name	Name	Details
System.PrConst.		
Addon_AddonLibraryName	Add-on <AddonLibraryName> parameter	ADDON_PARAM structure For add-on used for <AddonLibraryName>, refer to the following. ☞ Page 708 System variables
LabelMemorySize	Label memory size	Stores the label memory size in K bytes.
System.Md.		
Addon_AddonLibraryName	Add-on <AddonLibraryName> monitor	ADDON_MONI structure For add-on used for <AddonLibraryName>, refer to the following. ☞ Page 708 System variables
LabelMemoryFreeSize	Label memory free size	Stores label memory free space in K bytes.
MemoryUsage	System memory usage	ADDON_MON structure
MemorySize	System memory size	ADDON_PARAM structure
ADDON_PARAM		
RamSizeMax	Maximum RAM size	Specifies the maximum amount of system memory (RAM) used in the add-on in K bytes. For details on the setting method, refer to the following. ☞ Page 122 System memory (RAM) usage setting method
BackupRamSizeMax	Maximum backup RAM size	Specifies the maximum amount of system memory (backup RAM) used in the add-on in K bytes. For details on the setting method, refer to the following. ☞ Page 123 System memory (backup RAM) usage setting method
ADDON_MONI		
RamUsage	RAM usage	Stores the current amount of system memory (RAM) used in the add-on bin K bytes.
RamMaxUsage	RAM usage maximum	Stores the maximum amount of system memory (RAM) used in the add-on in K bytes.
BackupRamUsage	Backup RAM usage	Stores the current amount of system memory (backup RAM) used in the add-on in K bytes.
BackupRamMaxUsage	Backup RAM usage maximum	Stores the maximum amount of system memory (backup RAM) used in the add-on in K bytes.

System memory (RAM)

System memory (RAM) stores the data used for the control in an add-on library. The following shows the total size of system memory (RAM).

Model	RD78G	RD78GH
System memory (RAM) size	96 [M bytes]	256 [M bytes]

System memory (RAM) usage setting

The amount of system memory (RAM) available in each add-on library is set by System.PrConst.Addon_AddonLibraryName.RamSizeMax. Memory usage between each function can be flexibly adjusted according to the application with this parameter.

System memory (RAM) usage setting method

The maximum amount of system memory (RAM) used in the add-on can be set by the engineering tool.

- The maximum usage of system memory (RAM) can be set in K bytes.
- The set maximum usage is applied at system start.

The current/maximum amount of system memory (RAM) used in the add-on can be monitored by System.Md.Addon_AddonLibraryName.RamUsage/System.Md.Addon_AddonLibraryName.RamMaxUsage respectively. The value is stored in K bytes.

The current/maximum total amount of system memory (RAM) used in all add-ons can be monitored by System.Md.MemoryUsage.RamUsage/System.Md.MemoryUsage.RamMaxUsage respectively. The value is stored in K bytes.

Ex.

Examples of adjustable memory:

- Buffer size available in logging
- Operation profile data open area size
- Axis/variable instance size (It determines the total number of available axes.)
- ST program area (It determines the maximum number of program steps.)

Label memory (RAM) usage setting

Set usable memory (RAM) in label by System.PrConst.LabelMemorySize. Current memory free space in label memory can be checked by System.Md.LabelMemoryFreeSize.

System memory (backup RAM)

System memory (backup RAM) is a file system (latch drive) which stores the backup data used for the control in an add-on library. System memory (backup RAM) size is as shown below.

Model	RD78G	RD78GH
System memory (backup RAM) size	232 [K bytes]	472 [K bytes]

System memory (backup RAM) usage setting

The amount of system memory (backup RAM) available in each add-on library is set by System.PrConst.Addon_AddonLibraryName.BackupRamSizeMax. Memory usage between each function can be flexibly adjusted according to the application with this parameter.

System memory (backup RAM) usage setting method

The maximum amount of system memory (backup RAM) used in the add-on can be set by the engineering tool.

- The maximum usage of system memory (backup RAM) can be set in K bytes.
- The set maximum usage is applied at system start.

The current/maximum amount of system memory (backup RAM) used in the add-on can be monitored by System.Md.Addon_AddonLibraryName.BackupRamUsage/System.Md.Addon_AddonLibraryName.BackupRamMaxUsage respectively. The value is stored in K bytes.

The current/maximum total amount of system memory (backup RAM) used in all add-ons can be monitored by System.Md.MemoryUsage.BackupRamUsage/System.Md.MemoryUsage.BackupRamMaxUsage respectively. The value is stored in K bytes.

Ex.

Examples of adjustable memory:

- Number of ABS axes
- Number of event history storage
- Number of synchronous restoration function axes

Precautions

If the maximum usage of system memory (backup RAM) is changed, the backup data stored in the system memory (backup RAM) will be cleared.

Precautions

- The system memory usage can increase/decrease depending on the usage of add-on libraries. Allow a margin for the maximum memory to use.
- When more system memory capacity is required during RUN, an error or a warning is output for each add-on library. If memory cannot be secured in the function that affects the control, the whole motion system stops as the WDT error.

3.4 Software Reboot

The software reboot (system reset) is executed by writing the reboot command to the control command. When "Clear" is specified at the software reboot command, the system restarts and all data of the system is deleted.
This function is used to fetch such as parameters again and reset the motion system to the factory setting without stopping the system caused by the system power ON again.
The software reboot can execute reset and clear by setting a target specified value in System.Cd.SoftRebootRequest.

Action	Software reboot command	Target	Details
Reset	MOTION_RESET	Motion area	Resets parameters and control data while connection to the network communication is kept.
Clear	MOTION_CLEAR	Motion area	Clears built-in memory.
Quick clear	MOTION_QCLEAR	Motion area	Clears built-in memory quickly.

System.PrConst.SoftRebootEnable setting must be enabled to execute the software reboot. If the software reboot is executed with the setting disabled, the software reboot will not be executed and the warning "Soft Reboot Disabled Warning" (warning code: 0F00H) will be detected.
Register event history (type: operation power ON/reset) after the software reboot.

Relevant variables

Variable/Structure name	Name	Details
System.PrConst.		
SoftReboot_Enable	Soft reboot enabled	Sets the software reboot execution enable/disable. When the setting is disabled, the software reboot is not executed.
System.Cd.		
SoftRebootRequest	Soft reboot request	Executes the software reboot. MOTION_RESET: Resets the motion area only. MOTION_CLEAR: Clears the memory after rebooting the motion area. MOTION_QCLEAR: Clears the memory after rebooting the motion area quickly. Other than above: Not executed. The memory is deleted after the software reboot is completed (after the system restart).

Reset

This function imports parameters, etc., which are imported at system start, again without stopping the whole system. As MOTION_RESET is issued, motion area is reset, however, network area is not reset.
The each type data and the status of the network communication after the software reboot and hardware reboot are shown below.

Item	Software reboot (MOTION_RESET)	Hardware reboot
Label	Sets the default value	Sets the default value
Device	Clear*1	Clear
Program	Equivalent to STOP → RUN	Equivalent to STOP → RUN
Network communication	Watchdog counter update stop*2	Disconnects all stations and returns it after restart
File transmission by script file	Execute	Execute
Base system software/add-on	Reload	Reload
RAM drive	Clear	Clear

*1 The outside data of the reset target is maintained.
*2 The driver alarm may occur. In this case, restart the power of the motion system.
While the reset is operated, the RUN LED turns OFF.

Clear

The following data of the Motion module is deleted after the software reboot is carried out.

- User drive (/rom) storage data
- Latch drive (/lch) storage data
- Security information
- A part of system drive (/sys) storage data
- RAM drive (/ram) storage data

Point

- Write required data such as the add-on libraries, the parameters, and the programs after clear is completed.
- Since the password and key information related to security which are registered in the motion system are also deleted, set them again in needed.

It may take some time to carry out a low-level format of the drive.
While the clear is executed, the RUN LED is flashing (every 500 ms).
After the clear is completed, the RUN LED turns ON. The system starts with the parameters, the programs, and the backup data, etc. deleted.
When the clear is failure, the ERR LED is flashing (every 200 ms). In this case, execute clear again.

Quick clear

- The data of the Motion module is deleted after the software reboot is carried out. That data is the same as the data from the time of "Clear".
- The drive carries out quick format. Quick format can clear the data faster than low-level format ("Clear" is at run time). However, if an access to the file can not be performed normally after quick clear by reason of file system damage, etc., execute a low-level ("Clear") format.
- While the quick clear is executed, the RUN LED is flashing (every 500 ms).
- After the quick clear is completed normally, the RUN LED turns ON. The system starts with the parameters, the programs, and the backup data, etc. deleted.
- When the quick clear is failed, the ERR LED is flashing (every 200 ms). In this case, execute quick clear or clear again.

Precautions

- It is recommended to execute the software reboot during the PLC READY [Y0] OFF.
- When data reset by the software reboot is executed, the variables and status of the slave device change.
- During software reset of the motion area, the communication with the engineering tool and the dedicated instruction issued from the CPU module may be completed with an error.
- Do not turn OFF the power of the motion system while clear or quick clear is executed. An error "Drive Error" (error code: 3207H) may occur along with format failure of the drive (only for baseSystem add-on "Ver. 1.5" or later). In such a case, execute clear again.

MEMO

PART 2 MOTION CONTROL

4 START AND STOP

5 HOMING

6 AXIS CONTROL FUNCTION

7 DIRECT CONTROL

8 RELEVANT FUNCTIONS to POSITION

9 RELEVANT FUNCTIONS TO VELOCITY

10 RELEVANT FUNCTIONS TO TORQUE

11 CONTROL SUB FUNCTIONS

12 COMMON FUNCTIONS

13 SYNCHRONOUS CONTROL

14 OPERATION PROFILE FUNCTION

4 START AND STOP

4.1 Start

This section describes how to start an axis of the motion system.
In the motion system, motion control can be executed by executing motion control function blocks with programs of the control CPU module, such as the PLC CPU and the C language CPU, or the motion system built-in program language such as ST language.
Starting the motion control by executing the motion FB with the axis status (AxisName.Md.AxisStatus) of "4: Standstill" or the axes group status (AxesGroupName.Md.GroupStatus) of "4: GroupStandby" is called "start" of the axis.
Function blocks include the Execute type and the Enable type, and these are started at the rising edge of Execute or Enable which are input variables of the motion control function block.

Relevant variables

Variable/Structure name	Name	Details
AxisName.Md.		
AxisStatus	Axis status	Stores the axis status. 0: Disabled (Axis disabled) 1: ErrorStop (Error stop) 2: Stopping (Deceleration stop) 3: Homing (Homing) 4: Standstill (Standby) 5: DiscreteMotion (Positioning operation) 6: ContinuousMotion (Continuous operation) 7: SynchronizedMotion (Synchronized operation)
UseInGroup	Use in axes group	Displays whether the axis is used in the axes group or not. FALSE: Not used TRUE: Used
Homing_Request	Homing request	Displays TRUE/FALSE of the homing request.
HwStrokeLimit_FlsStatus	Upper limit signal status	Displays TRUE/FALSE of the FLS.
HwStrokeLimit_RlsStatus	Lower limit signal status	Displays TRUE/FALSE of theRLS.
ForcedStop_Released	Forced stop cancelling	Displays forced stop cancel status.
StopStatus	Stop status	Displays TRUE/FALSE of the stop signal (STOP).
AxesGroupName.Md.		
GroupStatus	Axes group status	Stores the axes group status. 0: GroupDisabled (Axes group disabled) 1: GroupErrorStop (Error stop) 2: GroupStopping (Deceleration stop) 4: GroupStandby (Standby) 5: GroupMoving (Operation)
System.Cd.		
SequenceReady	PLC ready	Specifies RUN/STOP of the system. FALSE: STOP TRUE: RUN Turns to TRUE/FALSE automatically by rising/falling of the Motion module [Y0]
System.Md.		
Ready	Ready	Stores ON/OFF of the ready [X0].
Sync	Synchronization flag	Stores ON/OFF of the synchronization flag [X1].

Point

Note the following when switching STOP/RUN by operating System.Cd.SequenceReady.

- When setting System.Cd.SequenceReady to the public label, do not operate [Y0] of the Motion module. STOP/RUN may not be switched correctly.
- The STOP/RUN status of the CPU module and the Motion module do not interlock. Even if the CPU module turns to STOP status with the axis controlled state by MCFB of the CPU module side, the axis control does not stop. Monitor the CPU operating status by the Motion module side program as necessary and stop the axis control. "SB004C: CPU operating status of own station" or "SW004B: CPU status of own station" shows the CPU operating status. For "SB004C: CPU operating status of own station" and "SW004B: CPU status of own station", refer to "List of Link Special Relay (SB)" and "List of Link Special Register (SW)" in the following manual.

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

  When the stop error occurs in the CPU module, the Motion module also turns to STOP automatically.

Starting conditions

To start the control, the following conditions must be satisfied.
The necessary conditions must be incorporated in the program so that the axis is not started when the conditions are not satisfied.
Starting conditions of the axis are mainly controlled by the state transition and the signal state of the axis and axes group.

Starting conditions for motion function blocks of the axis

■State transition

Name	State	Variable name
Axis state	4: Standstill	AxisName.Md.AxisStatus

■Signal state

Signal name		Signal state		Variable name
I/O signal	PLC READY [Y0]	ON	RUN	System.Cd.SequenceReady
	Ready [X0]	ON	Ready [X0]	System.Md.Ready
	Synchronization flag [X1]	ON	Module access enabled	System.Md.Sync
	Use in axes group	FALSE	Not used	AxisName.Md.UseInGroup
	Homing request	FALSE	Homing request FALSE	AxisName.Md.Homing_Request
External signal	Forced stop cancelling	TRUE	Emergency stop input FALSE (Forced stop cancel)	AxisName.Md.ForcedStop_Released
	Upper limit signal status	FALSE	Within the upper limit range	AxisName.Md.HwStrokeLimit_FlsStatus
	Lower limit signal status	FALSE	Within the lower limit range	AxisName.Md.HwStrokeLimit_RlsStatus
	Stop status	FALSE	Stop signal (STOP) FALSE	AxisName.Md.StopStatus

Starting conditions for motion function blocks of the axes group

■State transition

Name	State	Variable name
Axis state	4: Standstill	AxisName.Md.AxisStatus
Axes group state	4: GroupStandby	AxesGroupName.Md.AxesGroupStatus

■Signal state

Signal name		Signal state		Variable name
I/O signal	PLC READY [Y0]	ON	RUN	System.Cd.SequenceReady
	Ready [X0]	ON	Ready [X0]	System.Md.Ready
	Synchronization flag [X1]	ON	Module access enabled	System.Md.SyncFlag
	Homing request	FALSE	Homing request FALSE	AxisName.Md.Homing_Request
External signal	Forced stop cancelling	TRUE	Emergency stop input FALSE (Forced stop cancel)	AxisName.Md.ForcedStop_Released
	Upper limit signal status	FALSE	Within the upper limit range	AxisName.Md.HwStrokeLimit_FlsStatus
	Lower limit signal status	FALSE	Within the lower limit range	AxisName.Md.HwStrokeLimit_RlsStatus
	Stop status	FALSE	Stop signal (STOP) FALSE	AxisName.Md.StopStatus

4.2 Retrigger/Continuous Update

The following methods can be used to change the control of the on-going FB.

- Retrigger (Inputting Execute again)
- Continuous update (ContinuousUpdate)

With a retrigger/continuous update of the FB, input variables can be imported again to the FB instance in execution without interrupting operation.

The fetch cycle of each input variable depends on the longest cycle in the following lists.

- Execution cycle of POU (abbreviation for Program Organization Unit) that calls the FB
- Buffer memory refresh cycle (Only when using the FB in the CPU module side)
- Operation cycle

To change input variables when Execute turns FALSE to TRUE and when ContinuousUpdate turns TRUE, be sure to secure time longer than the cycle.

Also, when a retrigger/continuous update is performed to an instruction being buffered by multiple start of the MCFB, the change will be reflected at the FB switching.

Point

- A retrigger/continuous update which is immediately after execution and immediately before termination of the FB may be ignored.
  Immediately after execution: During AxisName.Md.Analyzing (Analyzing) is TRUE
  Immediately before termination: Immediately before Done turns to TRUE
- Even if a retrigger/continuous update is performed to an instruction being buffered, it may affect on the control in operation.

Retrigger

In an FB whose start condition is Execute, a retrigger can be executed by the re-rising edge of Execute.

For input variables that can be changed by a retrigger, refer to each FB specification.

Continuous update

If ContinuousUpdate is TRUE when Execute turns FALSE to TRUE, input variables will be continuously imported.

For input variables that can be imported, refer to each FB specification.

4.3 Multiple Start (Buffer Mode)

MCFBs can be executed continuously without stopping multiple MCFBs by executing the motion FB of another instance to the axis and the axes group that are executing the motion control FB.

Point

- To execute the MCFB with the axis and the axes group stop status ("4: Standstill" or "4: GroupStandby") is called "start". For details, refer to the following.
  ☞ Page 128 Start
- When the axis operation state and the axes group operation state are as the following, to execute the motion FB of another instance is called "multiple start".
  [Axis state in which multiple start is available (AxisName.Md.AxisStatus)]
  3: Homing (Only MC_Stop is possible)
  5: DiscreteMotion
  6: ContinuousMotion
  7: SynchronizedMotion
  [Multiple Start possible axes group state (AxesGroupName.Md.GroupStatus)]
  5: GroupMoving
- Multiple start of the single axis control FB cannot be executed to an axis operated in the axes group. It will cause the error "Operation FB Issue Error to the Axis during Axes Group Operating" (error code: 1A7CH).

Buffer mode type

The following types can be specified to the buffer mode, and the type that can be specified differs depending on the FB.

Setting value	Buffer mode type	Description
0: mcAborting	Aborting	Interrupts (cancel) the on-going FB and executes next FB immediately. For Aborting during single axis operation, refer to the following. ☞ Page 139 Aborting during single axis operation For Aborting during multiple axes operation, refer to the following. ☞ Page 147 Aborting during multiple axes operation
1: mcBuffered	Buffered	Buffers the next FB on the on-going FB. If the on-going FB has already been buffered, the next FB of the previous FB is buffered. (Up to 2.) Buffering FBs are executed in order after completion of the on-going FB For Buffered during single axis operation, refer to the following. ☞ Page 140 Buffered during single axis operation For Buffered during multiple axes operation, refer to the following. ☞ Page 148 Buffered during multiple axes operation
2: mcBlendingLow	BlendingLow	The lower target velocity between on-going FB and buffering FB is used as the switching speed.*1 For BlendingLow during single axis operation, refer to the following. ☞ Page 144 BlendingLow For BlendingLow during multiple axes operation, refer to the following. ☞ Page 151 BlendingLow
3: mcBlendingPrevious	BlendingPrevious	Target velocity of on-going FB is used as the switching speed.*1 For BlendingPrevious during single axis operation, refer to the following. ☞ Page 142 BlendingPrevious For BlendingPrevious during multiple axes operation, refer to the following. ☞ Page 149 BlendingPrevious
4: mcBlendingNext	BlendingNext	Target velocity of buffering FB is used as the switching speed.*1 For BlendingNext during single axis operation, refer to the following. ☞ Page 143 BlendingNext For BlendingNext during multiple axes operation, refer to the following. ☞ Page 150 BlendingNext
5: mcBlendingHigh	BlendingHigh	The higher target velocity between on-going FB and buffering FB is used as the switching speed.*1 For BlendingHigh during single axis operation, refer to the following. ☞ Page 144 BlendingHigh For BlendingHigh during multiple axes operation, refer to the following. ☞ Page 151 BlendingHigh

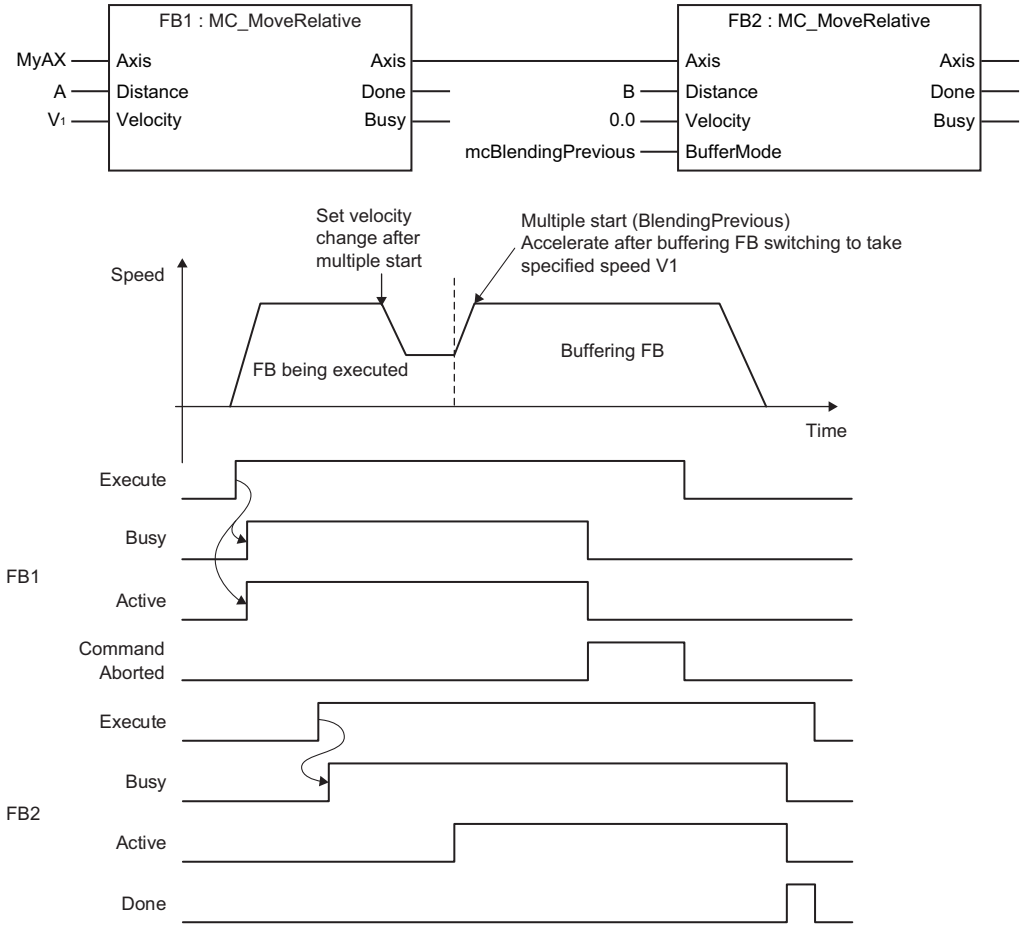
*1 Buffers the next FB on the on-going FB.
If the on-going FB has already been buffered, the next FB of the previous FB is buffered. (Up to 2)
Buffering FBs are executed in order after the on-going FB has reached the target position.
Stop is not executed between on-going FB and buffering FB in this mode.

Point

- Up to two motion FBs can be buffered after multiple start in one axis and an axes group. If multiple start is executed when two FBs have already been multiple started, the error "Warning Starting over Number of Buffering FBs" (warning code: 0D22H) occurs and waits the analysis of the multiple start FB until on-going FB is completed. Even though the warning occurs each multiple start, filter setting can be set not to detect the warning. For details on filter setting, refer to the following.
  ☞ Page 638 Checking Errors and Warnings
  When an error or a stop cause occurred in on-going FB, FBs which are waiting for the analysis are interrupted.
- When the warning "Warning starting over number of buffering FBs" (warning code: 0D22H) occurs, note that not to execute the multiple start until the on-going FB is completed. If the multiple FBs are waiting for the analysis by the multiple start, the next buffering FB may not be in order.
- Since multiple started FBs are executed immediately when Aborting has been specified, the FBs are not buffered. When on-going FBs include a buffering FB, all buffering FBs are interrupted. However, as the FBs which are awaiting analysis are not interrupted, those FBs start after the multiple started FBs are completed by specifying Aborting.
- When an error or a stop cause occurred, all buffering FBs are interrupted (the output CommandAbort turns TRUE).

Command omission at multiple start

When omitting the specified speed, the acceleration specification, and the deceleration specification in an FB that executes multiple start, specified speed of "the previous FB of buffering FB" at multiple start is taken over. If the specified speed input does not exist in "the previous FB of buffering FB", buffering is executed with speed 0.0.



- When FB switching is executed by multiple start, velocity change is executed to the target velocity of next FB. For the velocity waveform operation when the jerk is not specified with 0.0, refer to the following.
  ☞ Page 330 Acceleration/deceleration method
- When multiple FBs are multiple started at the same execution cycle, the order of FBs to be buffered is uncertain. In order to determine the buffering order, execute multiple start after checking that the previous FB is started (Busy = TRUE).

Relevant variables

Variable/Structure name	Name	Details
AxisName.Pr.		
OverrunOperation	Operation setting at overrun	Sets the operation settings when overrun occurs during operation. 1: ImmediateStop (Immediate stop) 2: KeepCurrentAcc (Keep Current Acceleration/Deceleration)
AxisName.Md.		
BufferingFBs	Number of buffering FBs	The number of buffering FBs (0 to 2)
AxesGroupName.Pr.		
OverrunOperation	Operation setting at overrun	Sets the operation settings when overrun occurs during operation. 1: ImmediateStop (Immediate stop)
AxesGroupName.Md.		
BufferingFBs	Number of buffering FBs	The number of buffering FBs (0 to 2)

Multiple start during single axis operation

When axis operation state is as the following, multiple start is enabled by starting a motion FB of another instance. (If the axis operation is in the "4: Standstill" status, it is regarded as "start".)

[Axis state in which multiple start is available (AxisName.Md.AxisStatus)]

3: Homing (Only MC_Stop is possible)

5: DiscreteMotion

6: ContinuousMotion

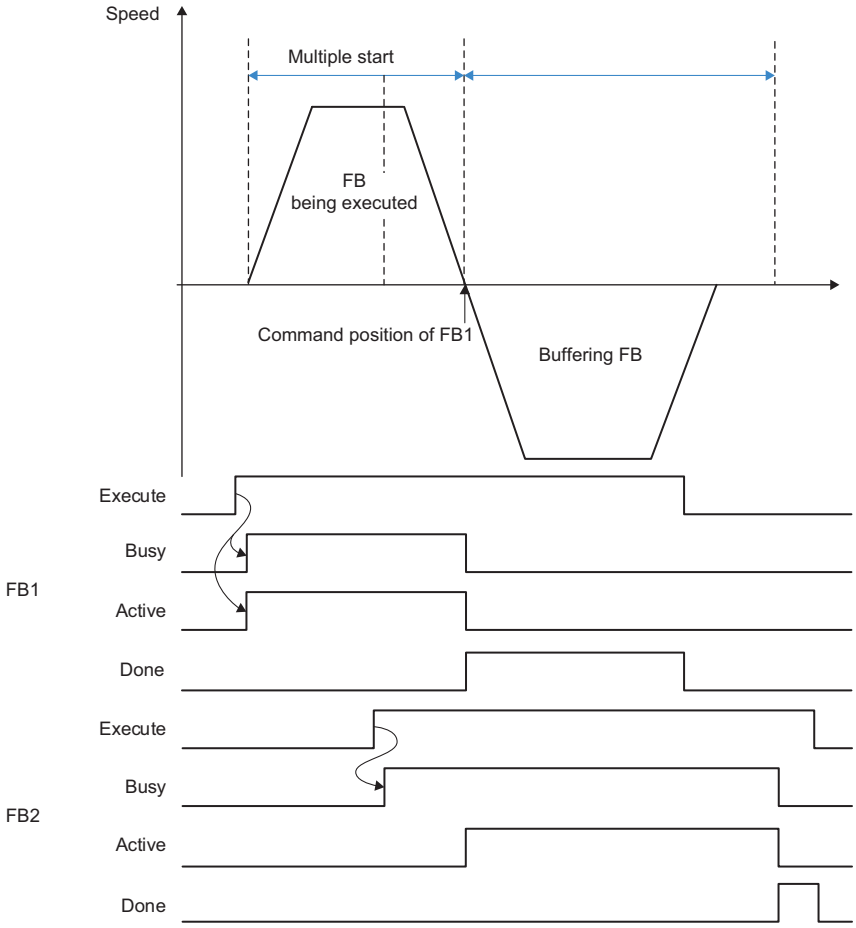
7: SynchronizedMotion

When the buffering FB input includes Direction and "4: mcCurrentDirection" is selected, the previous FB operation direction is taken over.

When the directions are different between on-going FB and next FB in the single operation, the operation differs depending on the selection in case of having reverse rotation permission selection (option bit5). The operation is the same as the operation at reverse rotation permission in case of not having reverse rotation permission selection (option bit5).

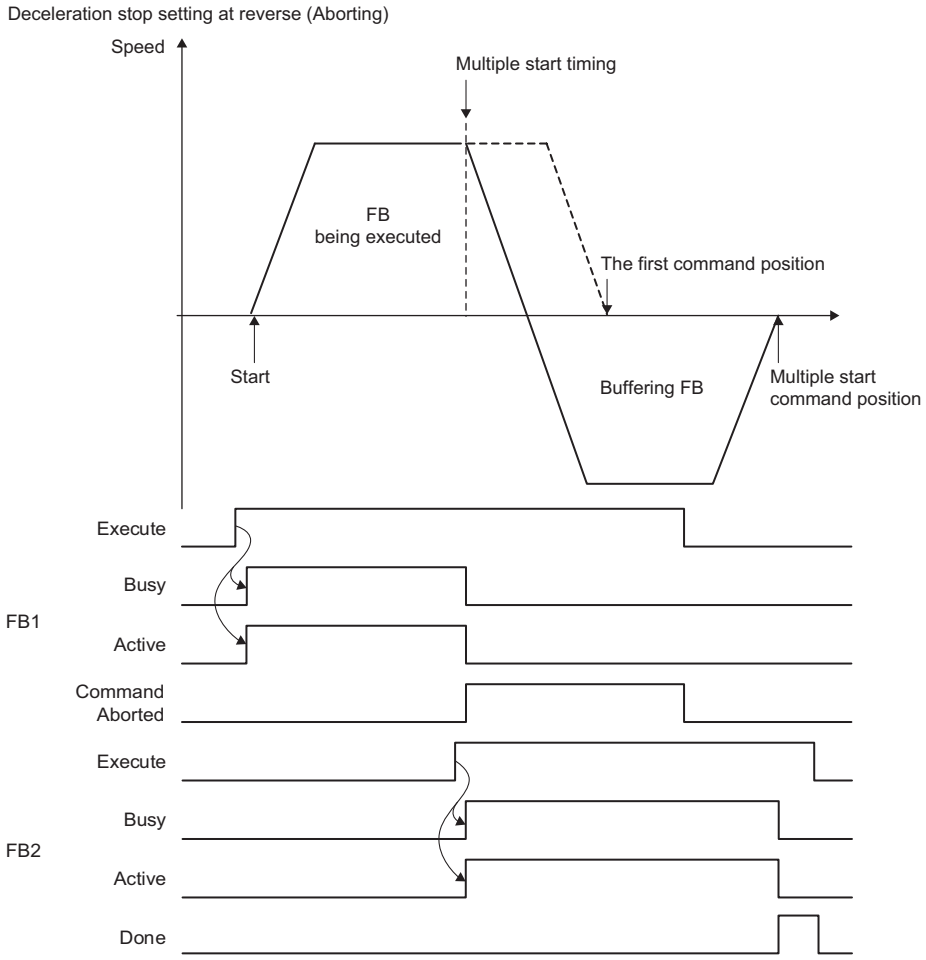
When the direction is reversed at while Blending is specified, a deceleration stop is executed at the target position of the on-going FB, and velocity change is executed to the target velocity of the buffering FB after the on-going FB is switched to the buffering FB. (It is the same operation as Buffered is specified.)

Deceleration stop setting at reverse (BlendingPrevious)



When reverse is enabled

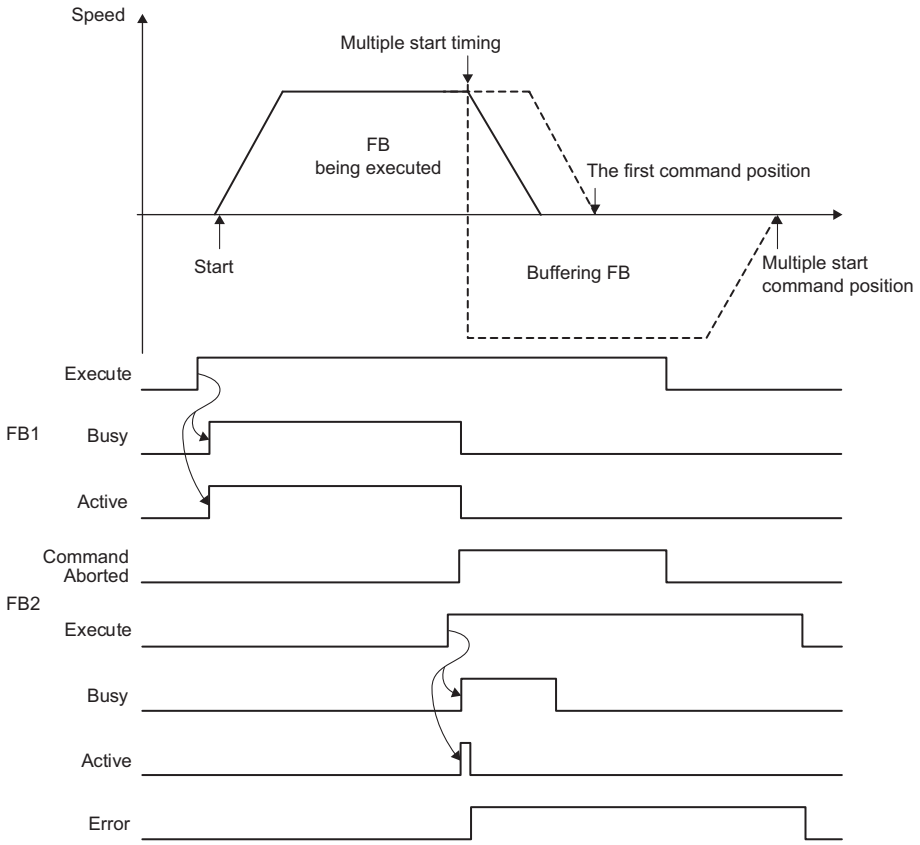
When reverse rotation permission is set in reverse rotation permission selection (option bit5), a deceleration stop is executed once. After the deceleration stop is completed, operation is started to the changed direction.



When reverse is disabled

When reverse rotation no permission is set in reverse rotation permission selection (option bit5), the error "Overrun Error" (error code: 1A7EH) occurs at switching and the axis decelerates to a stop.

Immediate stop setting at reverse



Buffer mode that can be set

The following shows buffer modes that can be specified in the single axis control FB. For operation details on each FB, refer to each FB specifications.

○: Buffer mode setting possible, —: Buffer mode setting not possible (axis error)

FB	Setting possible in FB			Completion output	Setting possible in the later operation FB		
	Aborting	Buffered	Blending		Aborting	Buffered	Blending
MC_Home	—*1	—	—	Done	○*2	—	—
MC_Stop	○	—	—	Done	○*2	—	—
MC_MoveAbsolute	○	○	○	Done	○	○	○
MC_MoveRelative	○	○	○	Done	○	○	○
MCv_SpeedPositionAbsolute	○	○	○	Done	○	○	○
MCv_SpeedPositionRelative	○	○	○	Done	○	○	○
MCv_Jog	—*1	—	—	Done	○	○	—
MCv_SpeedControl	○	○	○	InVelocity	○	○	—
MC_MoveVelocity	○	○	—	InVelocity	○	○	—
MC_TorqueControl	○	○	—	InTorque	○	○	—
MC_CamIn	○	○	—	EndOfProfile	○	○	—
MC_GearIn	○	○	—	InGear	○	○	—
MC_CombineAxes	○	○	—	InSync	○	○	—
MCv_SmoothingFilter	—*1	—	—	—	○*2	—	—
MCv_DirectionFilter	—*1	—	—	—	○*2	—	—
MCv_SpeedLimitFilter	—*1	—	—	—	○*2	—	—
MCv_BacklashCompensationFilter	—*1	—	—	—	○*2	—	—

*1 Since this FB can be started only when the axis state (AxisName.Md.AxisStatus) is "4: Standstill", multiple start is not possible.

*2 Only MC_Stop is possible.

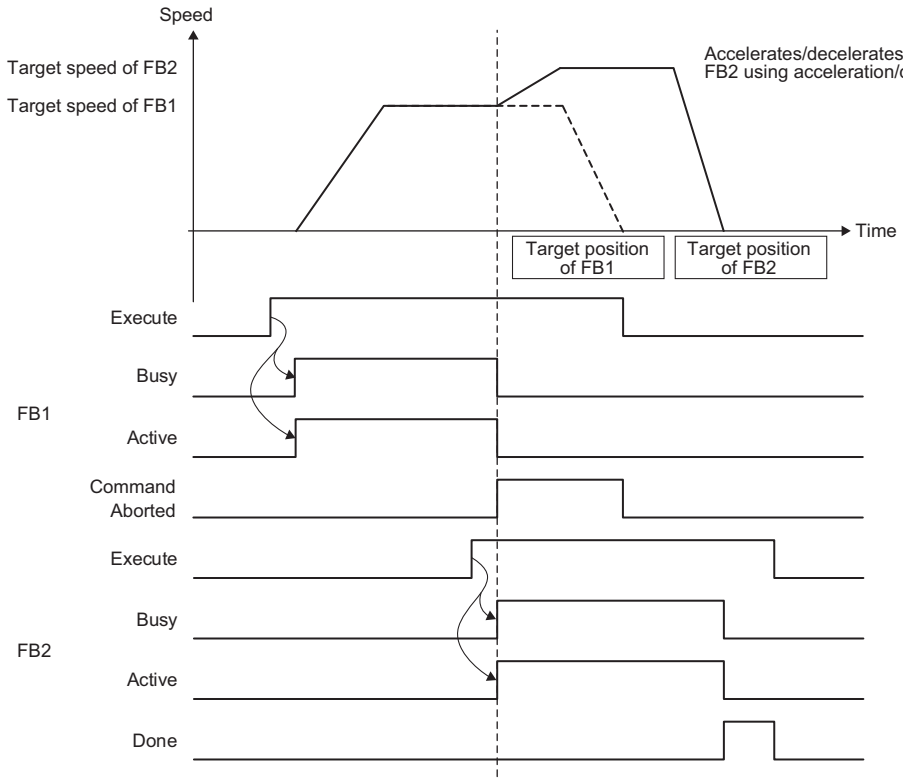
Aborting during single axis operation

On-going FB is interrupted and the next FB is executed.

Since a buffer is not used, an error does not occur even if two of the on-going FBs have already been in buffering, and the FB is executed. (All FBs during buffering are interrupted.)

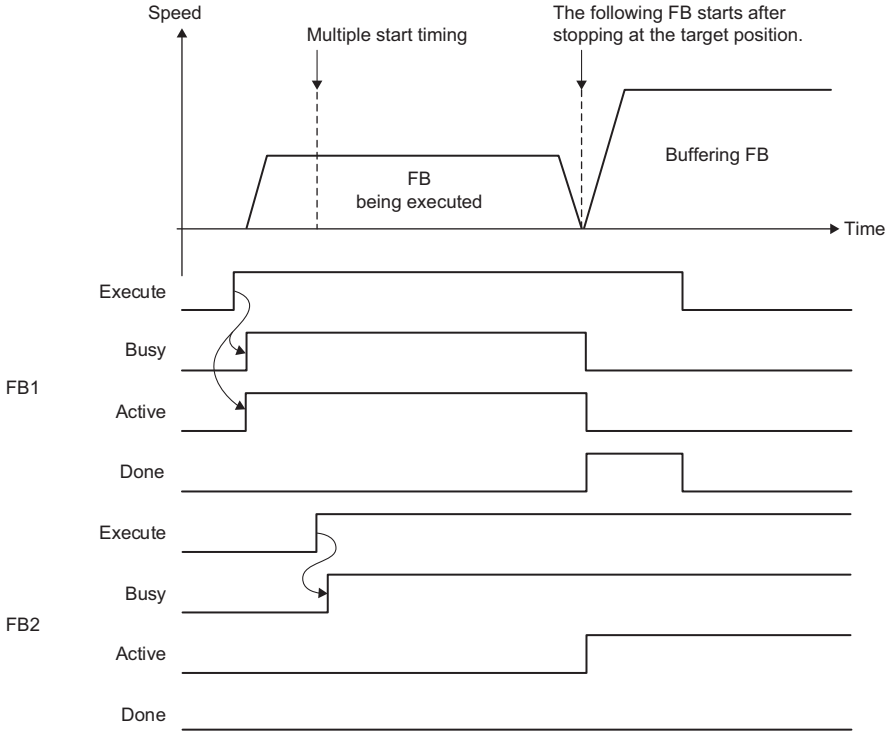
When omitting to specify the buffer mode, this operation is performed.

An operation example when Aborting is executed to the on-going FB1 with the linked start FB2 is shown below.



Buffered during single axis operation

Multiple started FBs are buffered and wait (Busy = TRUE) until the on-going FB is completed. After the on-going FB is normally completed, multiple started FBs are performed in the order of being buffered.



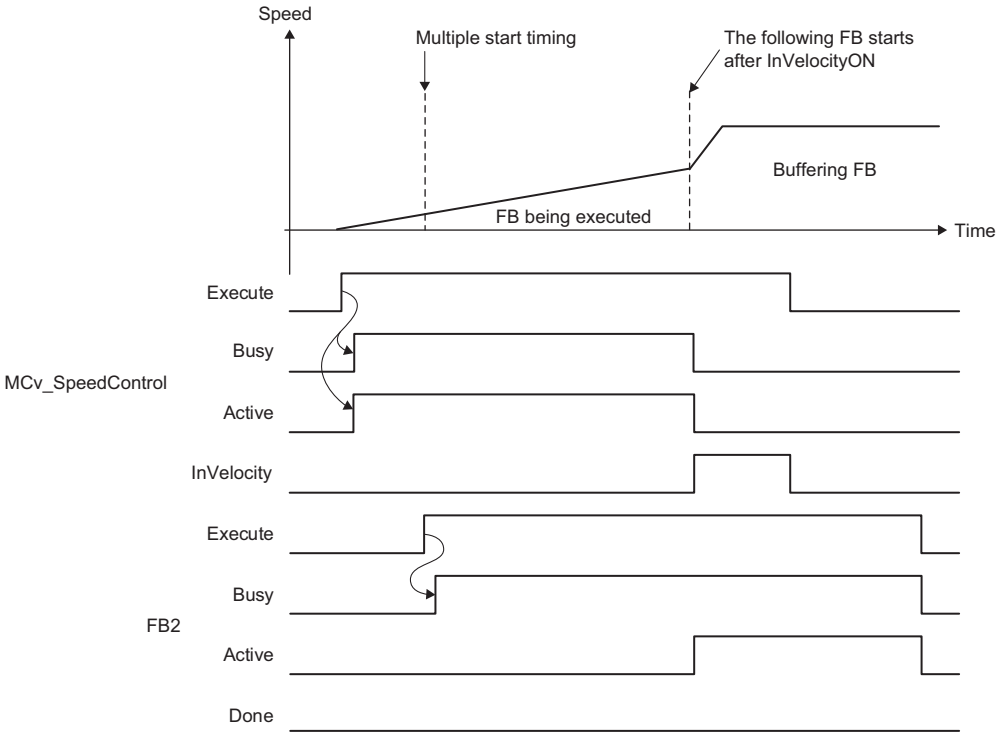
Conditions to determine completion are different depending on the MCFB, and the FB may not be in the stop status with speed 0 at completion. For details, refer to the following.

☞ Page 135 Multiple start during single axis operation

Ex.

When an FB is multiple started with Buffered in MCv_SpeedControl

In MCv_SpeedControl, the FB is determined that it is completed when InVelocity turns TRUE, and the following buffering FB is executed. (The current speed at this time is used as the switching speed.)

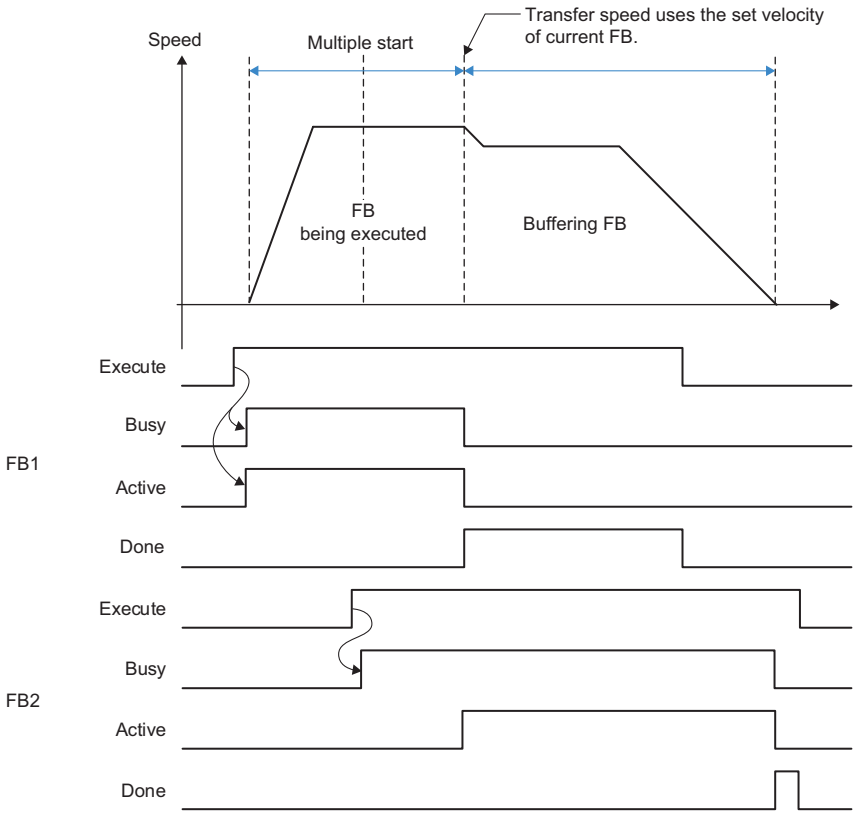


Blending during single axis operation

Multiple started FBs are buffered and wait until the on-going FB is completed. After the on-going FB reached the target position, buffering FBs are performed in order. In this case, the switching speed differs depending on the mode specified in the buffer mode.

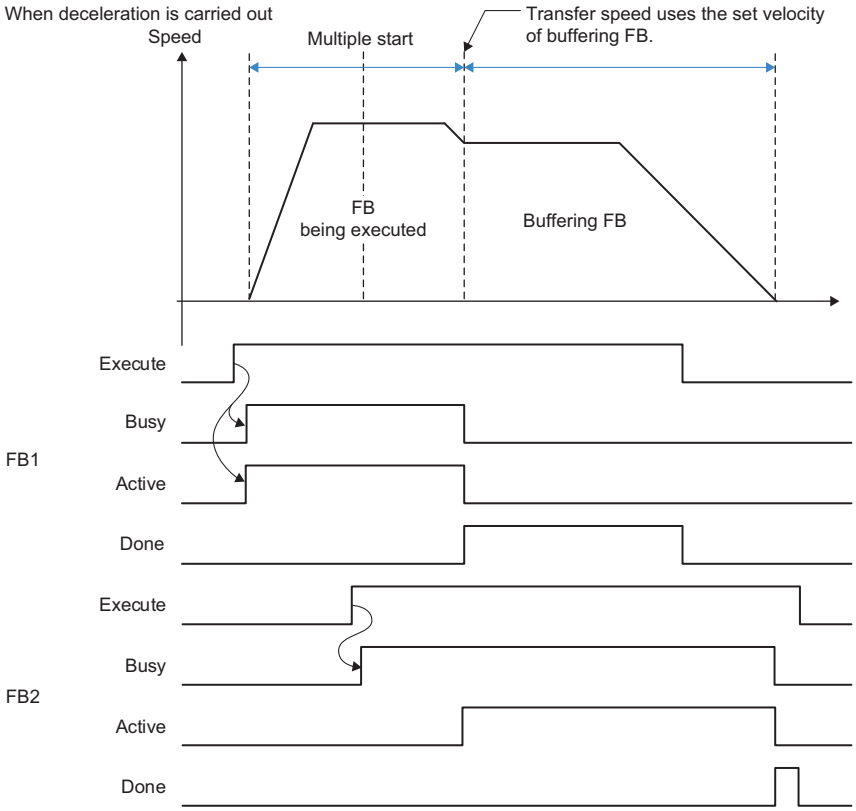
■BlendingPrevious

The FB is operated at the speed of the on-going FB to the target position. After switching to the FB that is during buffering, velocity change is executed to the target velocity of the buffering FB.

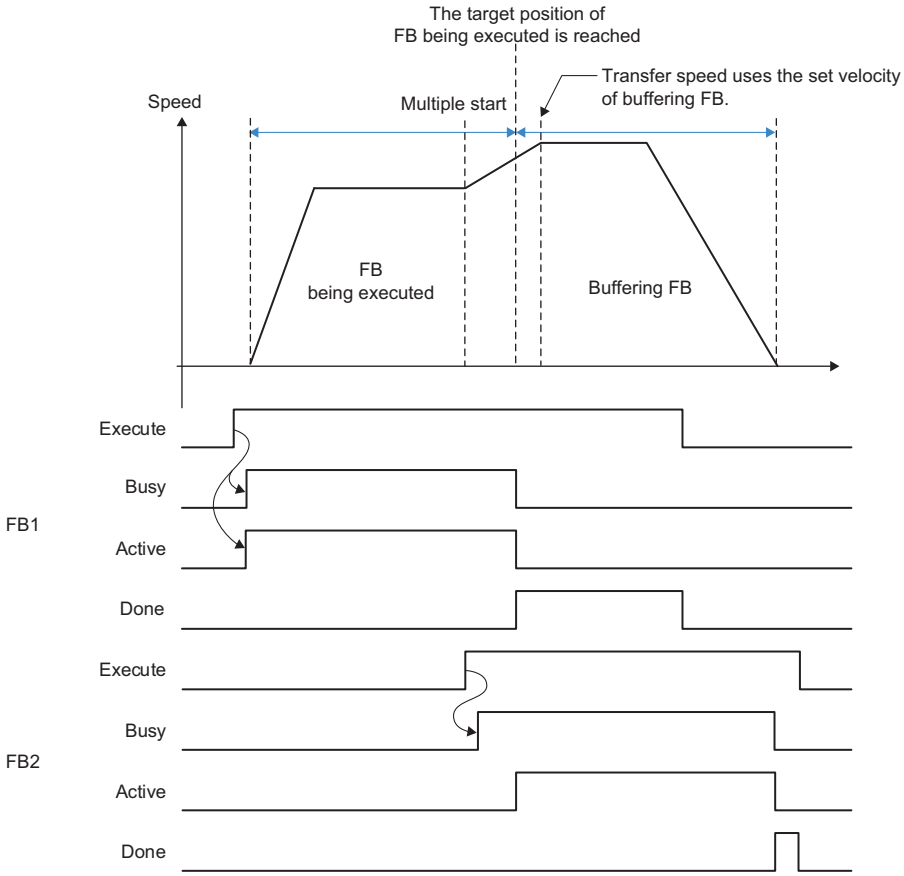


■BlendingNext

The FB is operated so that the velocity becomes the target velocity of the FB in buffering at the target position of the on-going FB.



Depending on the multiple start timing, the movement amount in order to change the speed to the target velocity of the FB in buffering may be in short at the target position of the on-going FB. In this case, velocity change is executed immediately, but the velocity cannot reach the target velocity of the FB that is during buffering at the target position of the on-going FB.



■BlendingLow

The slower target velocity between the on-going FB and the buffering FB is used as the switching speed.
The operation is same as BlendingPrevious when the on-going FB target velocity is slower, and it is same as BlendingNext when the buffering FB target velocity is slower.

■BlendingHigh

The higher target velocity between the on-going FB and the buffering FB is used as the switching speed.
The operation is same as BlendingPrevious when the on-going FB target velocity is higher, and it is same as BlendingNext when the buffering FB target velocity is higher.

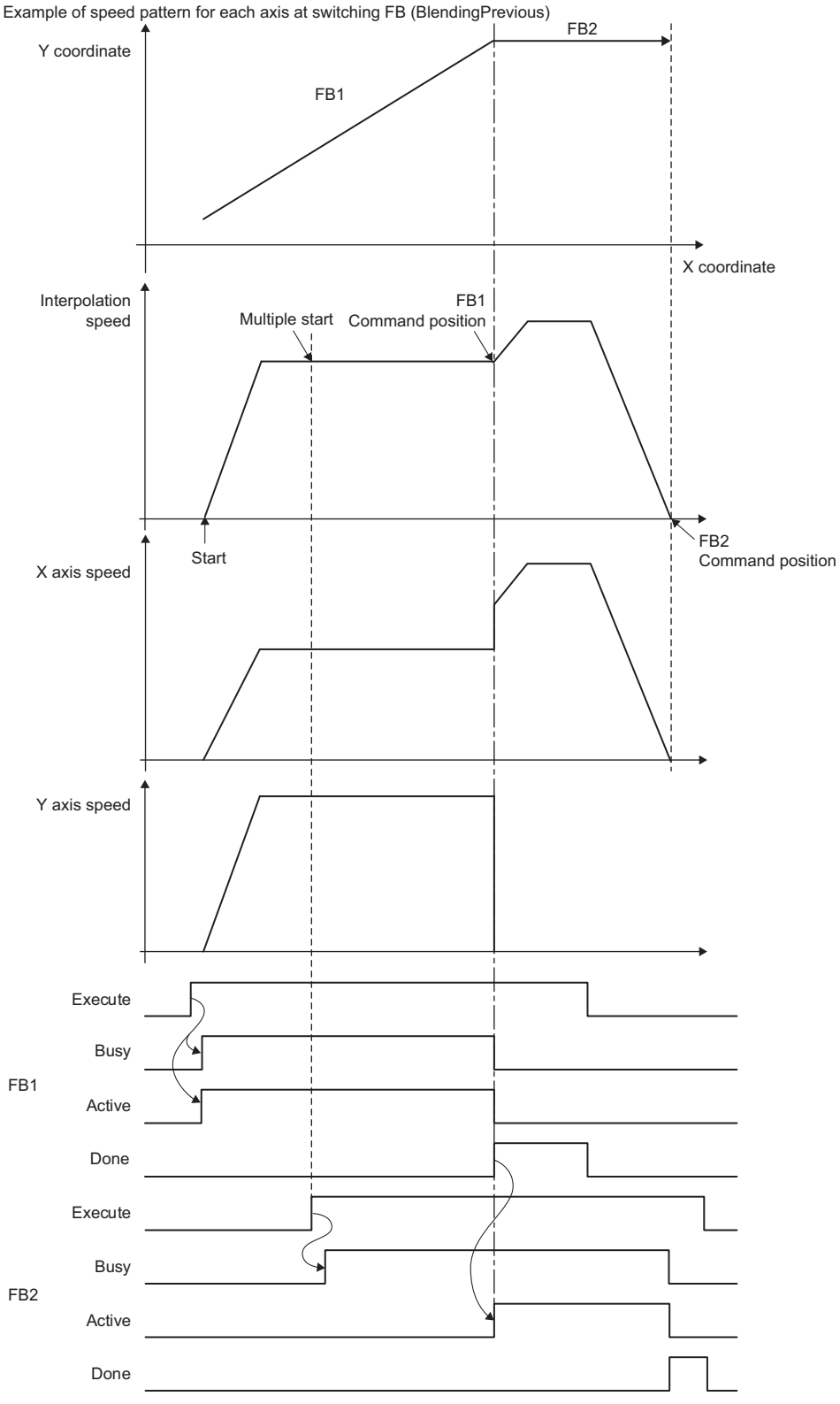
Multiple start during multiple axes operation

When axes group operation state is as the following, multiple start is enabled by executing the motion FB of another instance. (If the axis operation is in the "4: GroupStandby" status, it is regarded as "start".)

[Axes group status that can be multiple started (AxesGroupName.Md.GroupStatus)]

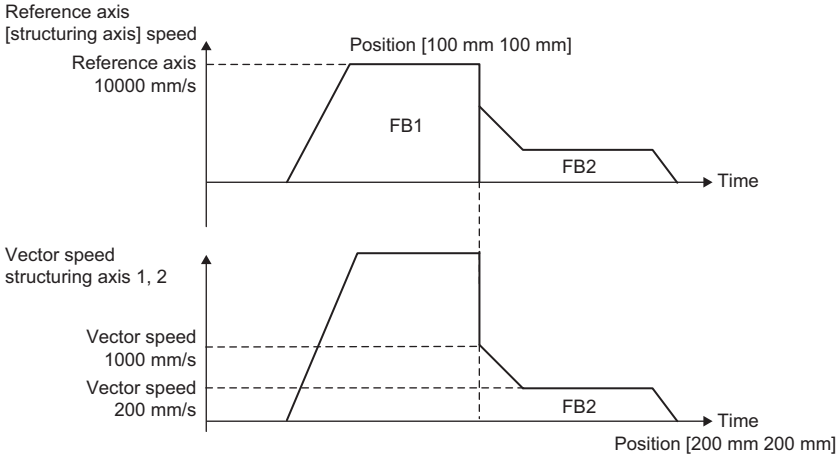
5: GroupMoving

In multiple start during multiple axes operation, the current interpolation speed of the on-going FB is used as the switching speed. Therefore, acceleration/deceleration may not be executed and rapid a velocity change may occur depending on the movement amount and direction of each axis in the buffering FB.



When multiple start by specifying Aborting or Blending, if on-going FB (FB1) or next FB (FB2) has the VelocityMode input and the velocity mode settings are different between FBs, the warning "Multiple Start Speed Mode Specified Mismatch Warning" (warning code: 0D11H) is output because the velocity may change suddenly.
In such a case, FB1 VelocityMode is used in FB1 and FB2 VelocityMode is used in FB2 to control. In this case, the switching speed is controlled using only values. (Unit/speed conversion by VelocityMode is not executed.)

Ex.
Multiple start by specifying BlendingPrevious from FB1 with the reference axis speed to FB2 with the vector speed



The operation is carried out so that the reference axis speed of FB1 becomes the speed set in Velocity (reference axis speed 1000 mm/s). The reference axis speed at the target position of FB1 is regarded as the switching speed and the current vector speed value of FB2 without conversion, and it accelerates and decelerates the vector speed (200 mm/s) set in Velocity of FB2.
Multiple start of the single axis control FB cannot be executed to the axis that is during operation in the axes group. If it is executed, the error "Operation FB Issue Error to the Axis during Axes Group Operating" (error code: 1A7CH) will occur in the target axis, and the axes group FB that is during execution will stop.

Buffer mode that can be set

The following shows buffer modes that can be specified in the multiple axes control FB. For operation details on each FB, refer to each FB specifications.
○: Buffer mode setting possible, —: Buffer mode setting not possible (axes group error)

FB	Setting possible in FB			Completion output	Setting possible in the later operation FB		
	Aborting	Buffered	Blending		Aborting	Buffered	Blending
MC_GroupStop	—*1	—	—	Done	○*2	—	—
MCv_MoveLinearInterpolateAbsolute	○	○	○	Done	○	○	○
MCv_MoveLinearInterpolateRelative	○	○	○	Done	○	○	○
MCv_MoveCircularInterpolateAbsolute	○	○	○	Done	○	○	○
MCv_MoveCircularInterpolateRelative	○	○	○	Done	○	○	○

*1 When it is started during axis operation, the operation will be same as Aborting.
*2 Only MC_GroupStop is possible.

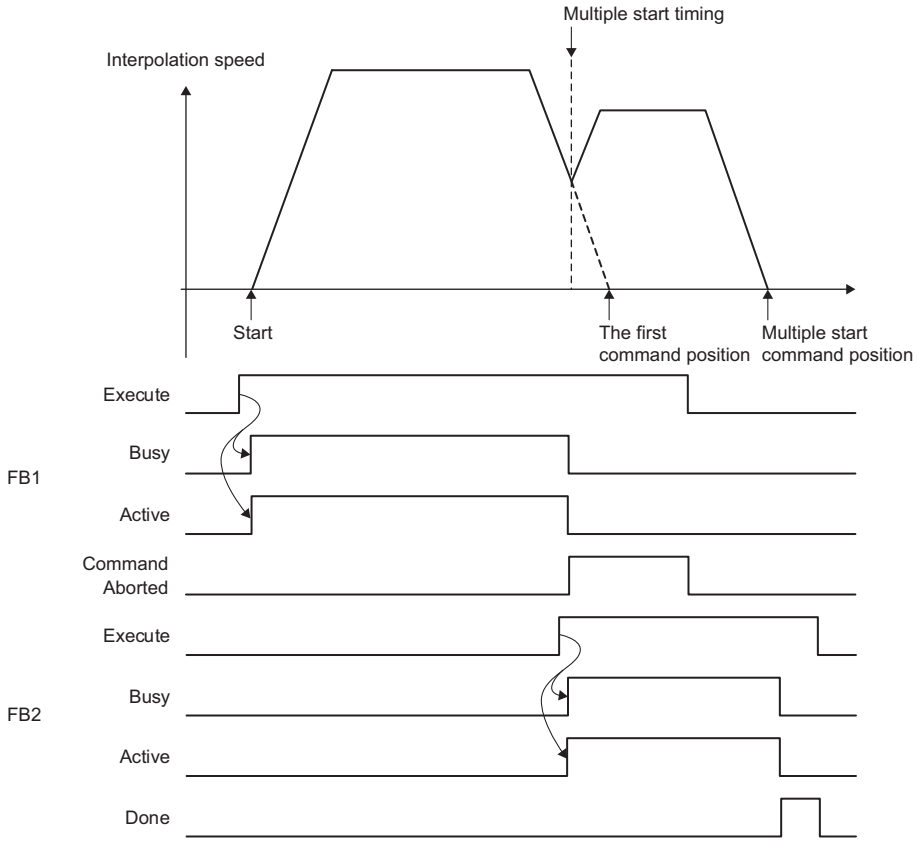
Aborting during multiple axes operation

On-going FB is interrupted and the multiple started FB is executed.

Since a buffer is not used, an error does not occur even if two of the on-going FBs have already been in buffering, and the FB is executed. (All FBs during buffering are interrupted.)

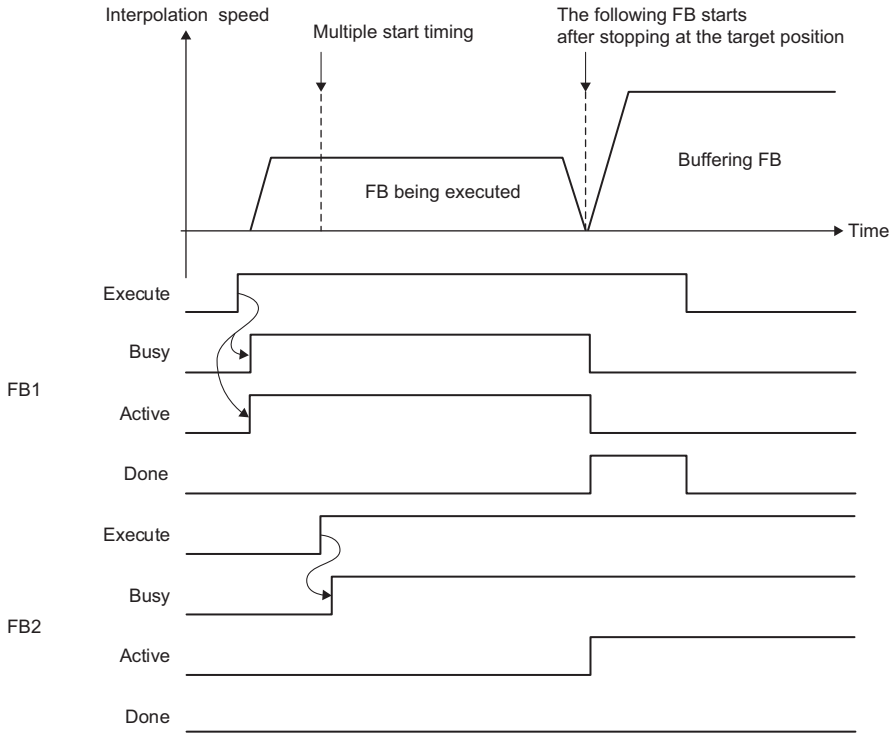
When omitting to specify the buffer mode, this operation is performed.

An operation example when Aborting is executed to the on-going FB1 with the linked start FB2 is shown below.



Buffered during multiple axes operation

Multiple started FBs are buffered and wait until the on-going FB is completed. After the on-going FB is normally completed, multiple started FBs are performed in the order of being buffered.

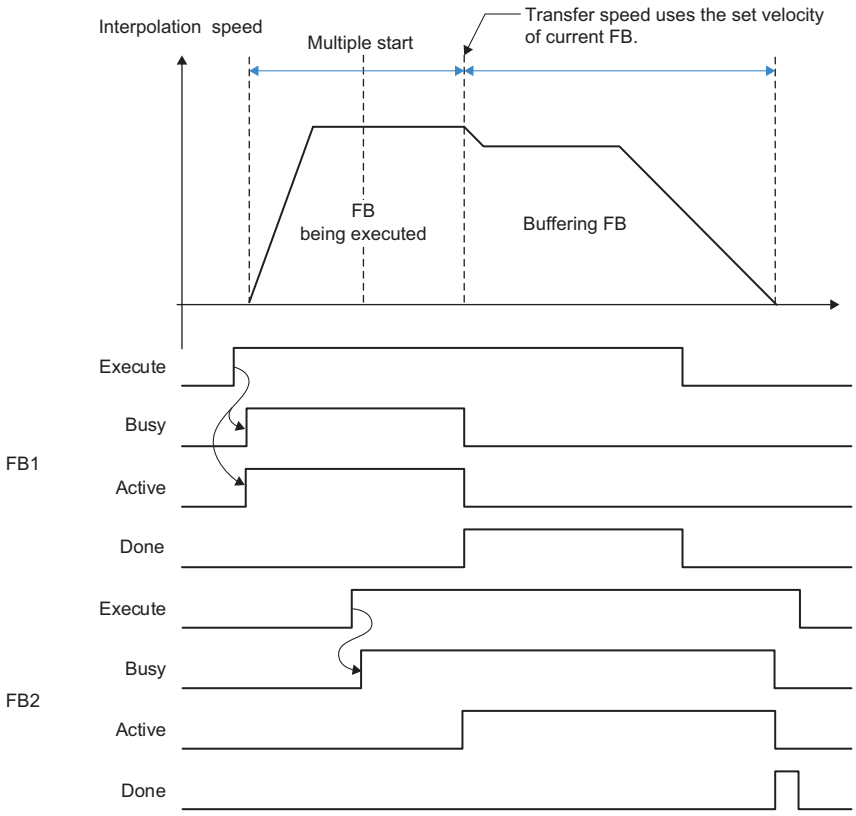


Blending during multiple axes operation

Multiple started FBs are buffered and wait until the on-going FB is completed. After the on-going FB reached the target position, buffering FBs are performed in order. In this case, the switching speed differs depending on the mode specified in the buffer mode.

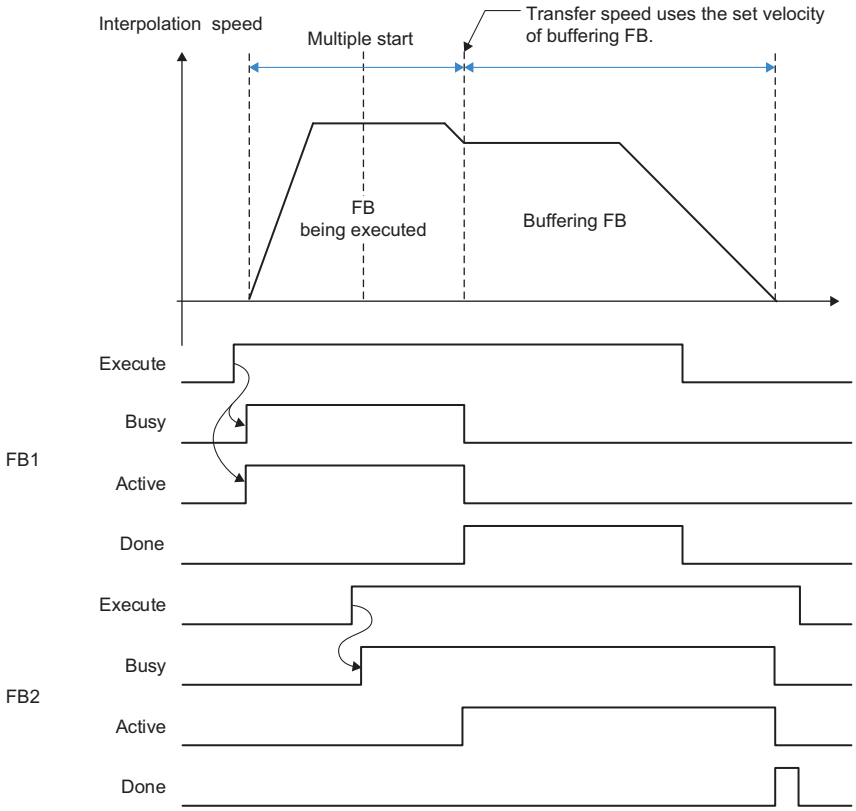
■BlendingPrevious

The FB is operated with the interpolation velocity of on-going FB to the target position. After switching to the FB that is during buffering, velocity change is executed to the target velocity of buffering FB.

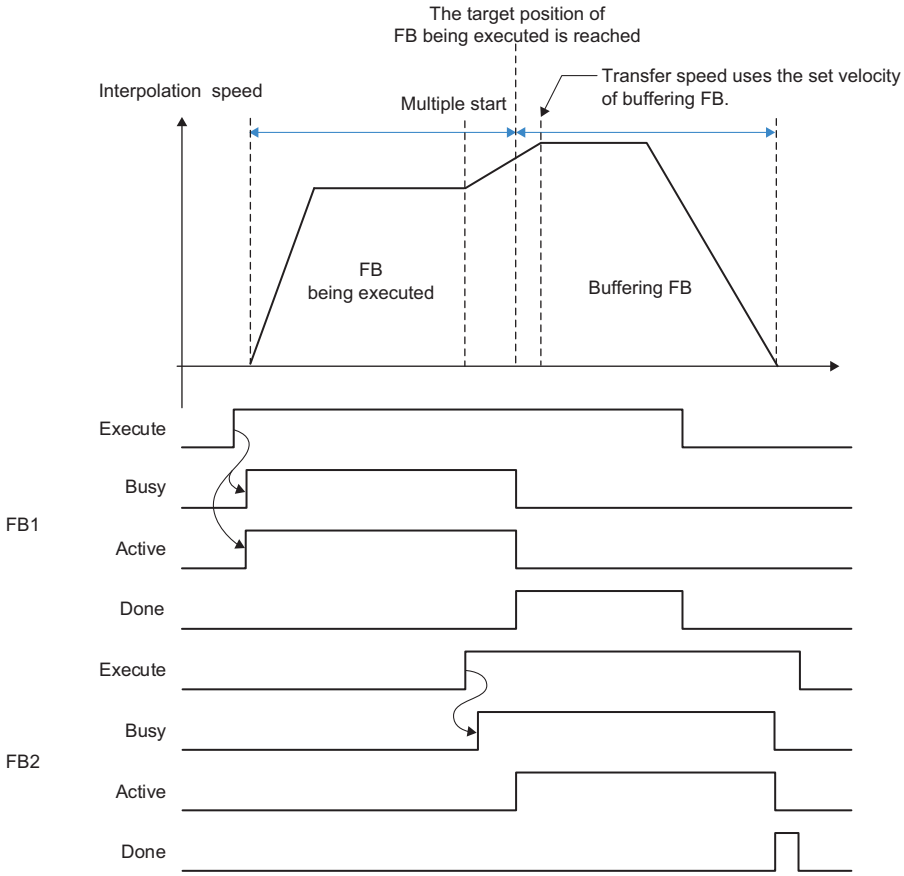


■BlendingNext

The FB is operated so that the interpolation velocity becomes the target velocity of FB that is during buffering at the target position of the on-going FB.



Depending on the multiple start timing, the movement amount to change velocity to the target velocity of the FB that is during buffering at the target position of the on-going FB. In this case, velocity change is executed immediately, but the velocity cannot reach the target velocity of the FB that is during buffering at the target position of the on-going FB.



■BlendingLow

The slower target velocity between the on-going FB and the buffering FB is used as the switching speed.
The operation is same as BlendingPrevious when the on-going FB commanded speed is slower, and it is same as BlendingNext when the buffering FB commanded speed is slower.

■BlendingHigh

The higher target velocity between the on-going FB and the buffering FB is used as the switching speed.
The operation is same as BlendingPrevious when the on-going FB commanded speed is higher, and it is same as BlendingNext when the buffering FB commanded speed is higher.

Option of buffering FB

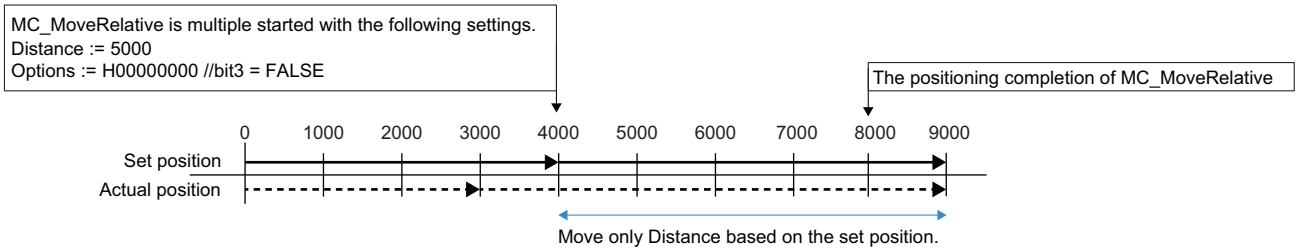
Set the option of the function used for the buffering FB by Options of the buffering FB. For the positioning control FB which can be specified, refer to details for the operation of the each positioning control FB.
The option to be used by the multiple start is shown below.

Positioning selection in the buffer mode (Options bit3)

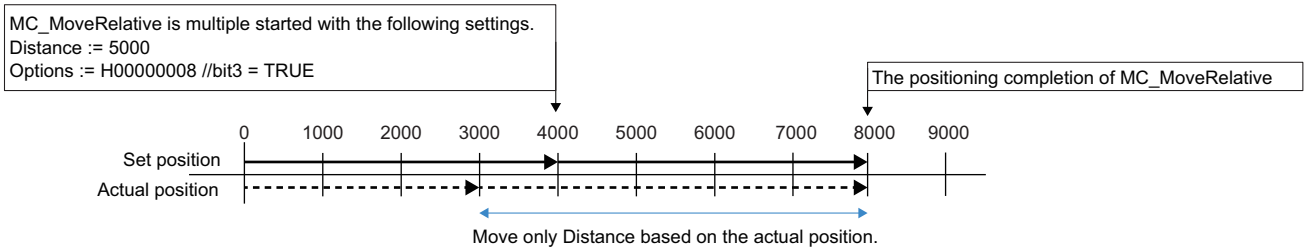
When the relative value positioning control is multiple started, the relative value positioning control from set position or actual position is carried out by the position selection at the multiple start of FB (Options bit 3).
The axis type which does not have the actual position ignores this setting.

Setting value	Description
0	The relative value positioning control from set position
1	The relative value positioning control from actual position

Ex.
When the relative value positioning control from set position (Options bit 3 = 0) is specified



When the relative value positioning control from actual position (Options bit 3 = 1) is specified



Reverse rotation permission selection (Options bit5)

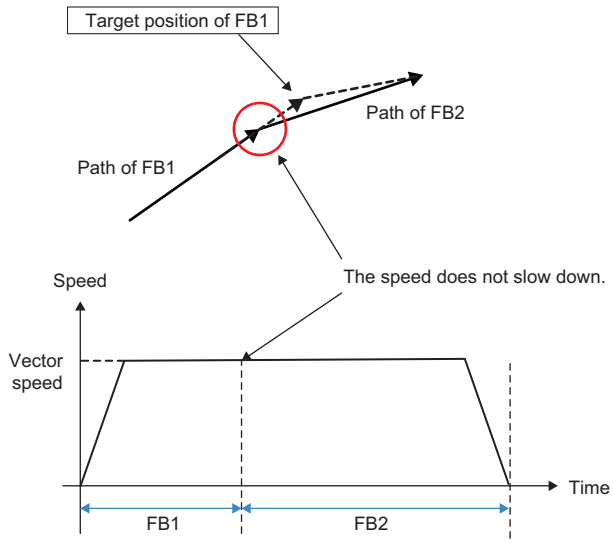
In the single axis operation FB, reverse rotation is allowed when the operation direction differs between the on-going FB and the buffering FB. For details, refer to the following.
☞ Page 135 Multiple start during single axis operation

Setting value	Description
0	Reverse is allowed.
1	Reverse is not allowed.

Near pass

At multiple start by Blending, near pass is performed to suppress machine vibration which occurs at switching of the positioning control FB.
The remainder of the movement amount which occurs at the end of the positioning control FB is used for the next one. Set velocity does not decrease and machine vibration which occurs by the velocity change can be suppressed as the positioning is not carried out for each positioning control FB. Since the alignment is not performed every positioning control FB, the operation is controlled in a path that passes near the position set in the FB.

Ex.
For multiple starting the 2-axes linear interpolation control with Blending (multiple start of FB2 during controlled of FB1)



Combination with retrigger/continuous update

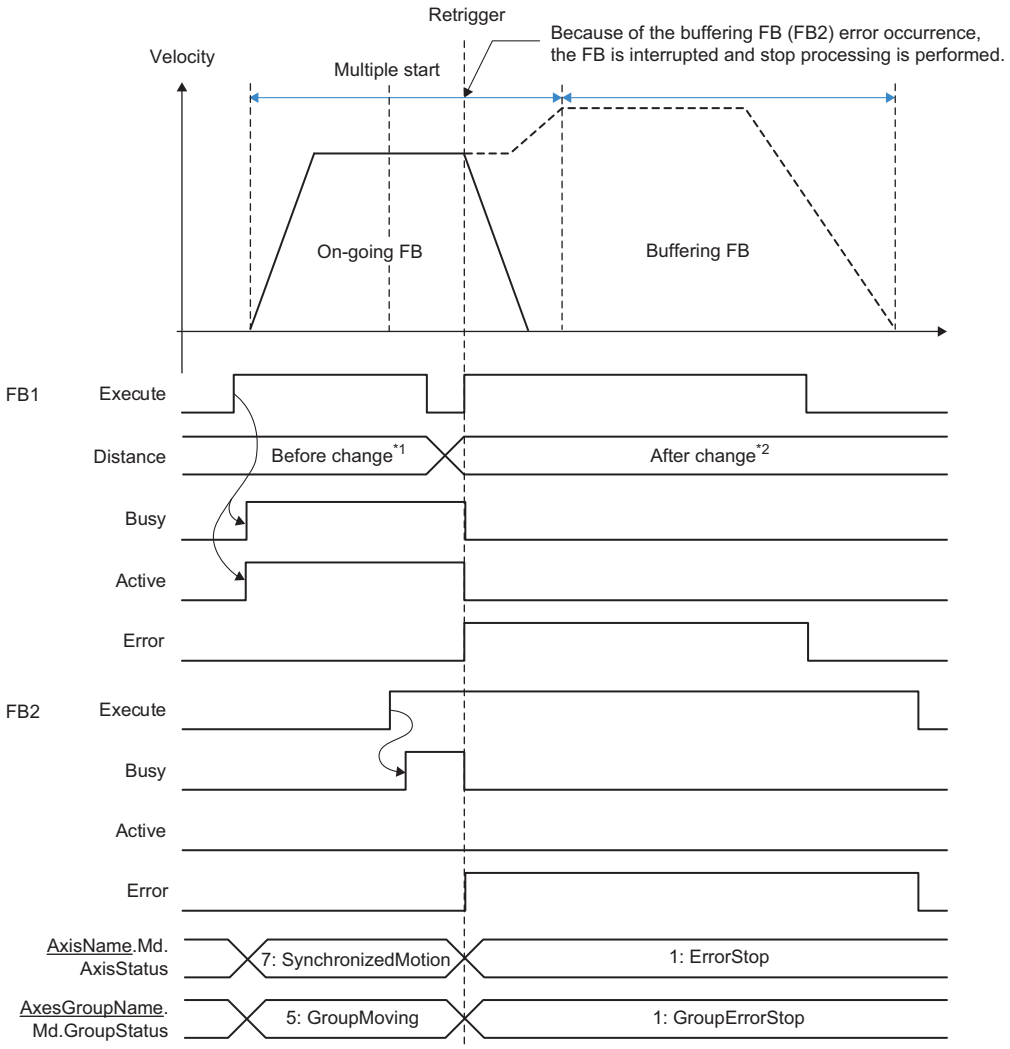
Reanalysis of the buffering FB by retrigger/continuous update

When the target position or the target velocity is changed by retrigger/continuous update for the on-going FB, the reanalysis of the buffering FB may be needed.

As a result of reanalysis, when an error occurs in any of the FBs, the axis and the axes group status turn into Error stop and the on-going FB is interrupted and stopped. (The buffering FB is canceled.)

Ex.

The changed target position is out of the software stroke limit range by retriggering the on-going FB.

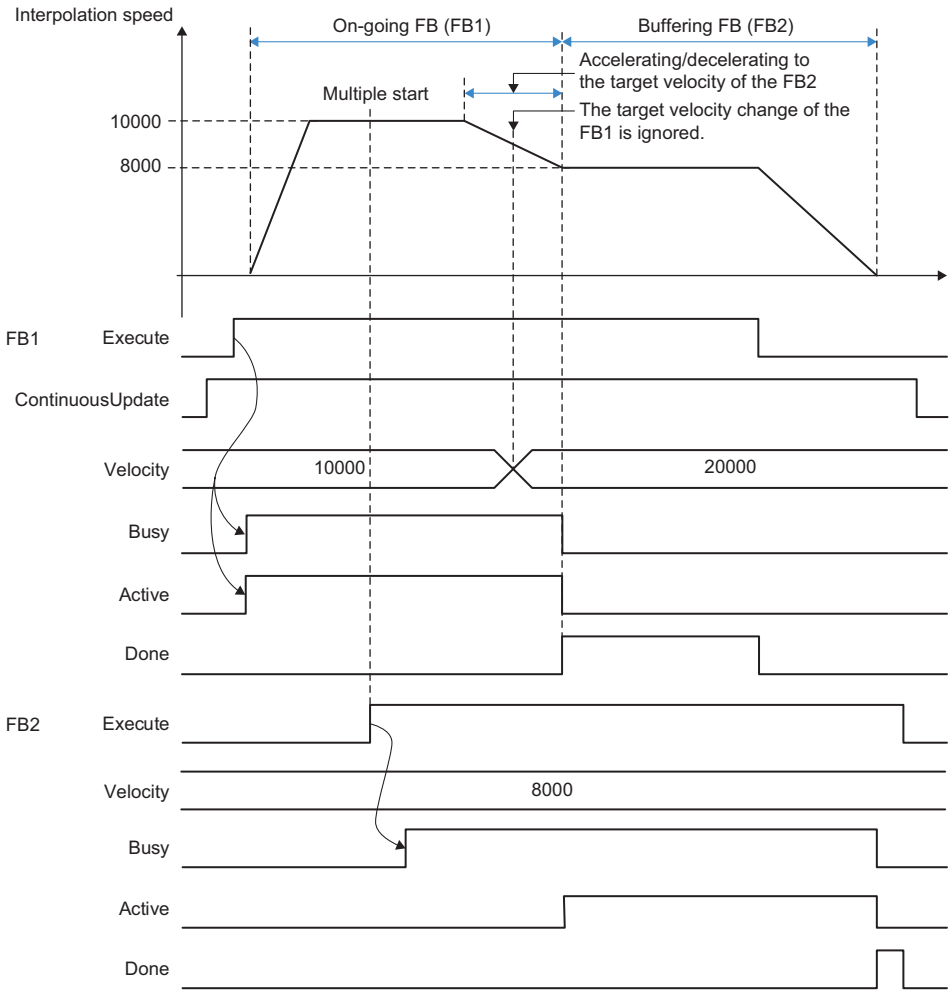


*1 The target position is within the software stroke limit range.

*2 The target position is out of the software stroke limit range.

BlendingNext changes target velocity at accelerating/decelerating

As BlendingNext is specified and the multiple start is executed, while the buffering FB is accelerating/decelerating to its target velocity (transfer speed), the target velocity change of on-going FB is not allowed.



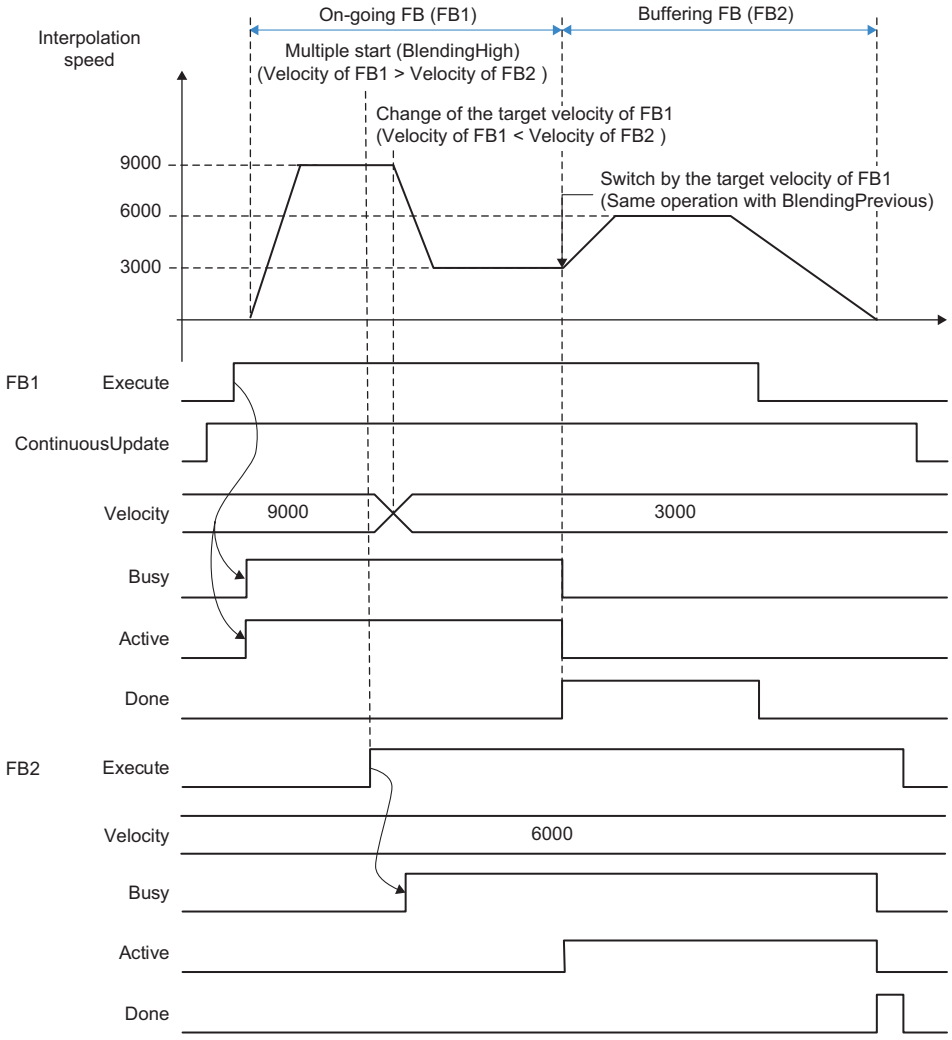
BlendingLow/BlendingHigh changes target velocity after multiple start

Which velocity is used, the velocity of on-going FB or buffering FB, is determined at the multiple start by BlendingLow/BlendingHigh.

Therefore, if the size relation the target velocity between the on-going FB and the buffering FB is changed by target velocity chagne after multiple start, the velocity used for switching is not changed.

Ex.

When the target velocity is changed after multiple start by BlendingHigh

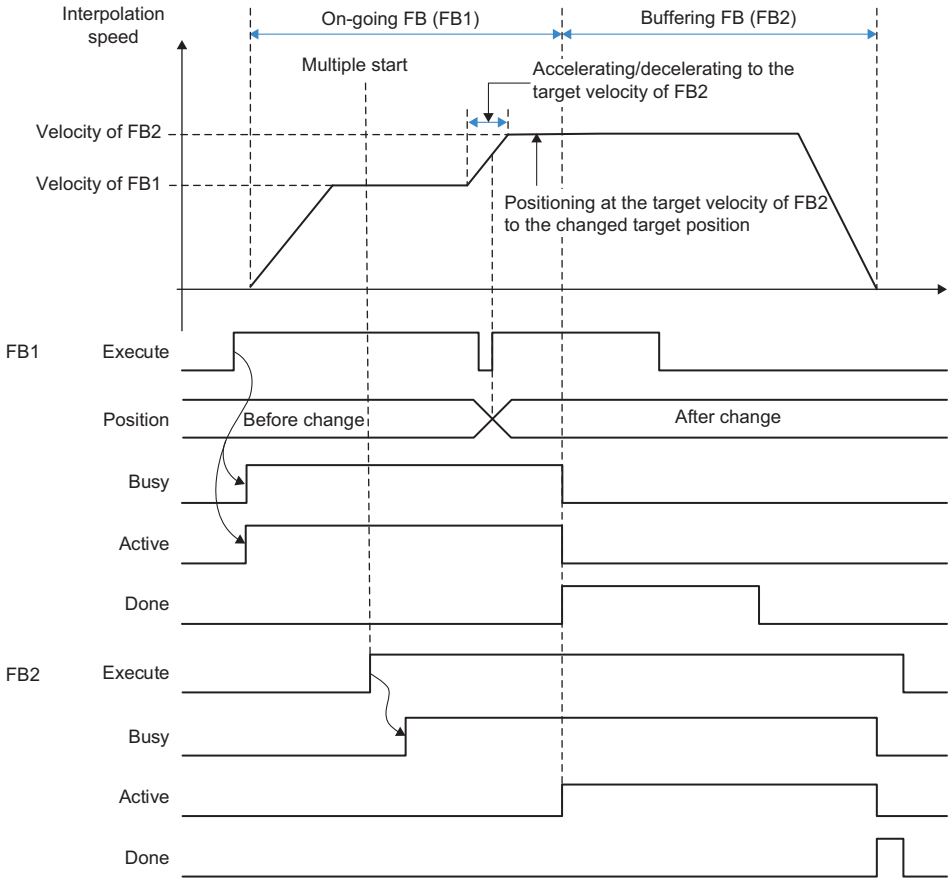


BlendingNext changes target position/movement distance at accelerating/decelerating

As BlendingNext is specified and the multiple start is executed, when the target position/movement distance change during accelerating/decelerating to the target velocity (transfer speed) of the buffering FB is executed, the operation of retrigger and continuous update is different.

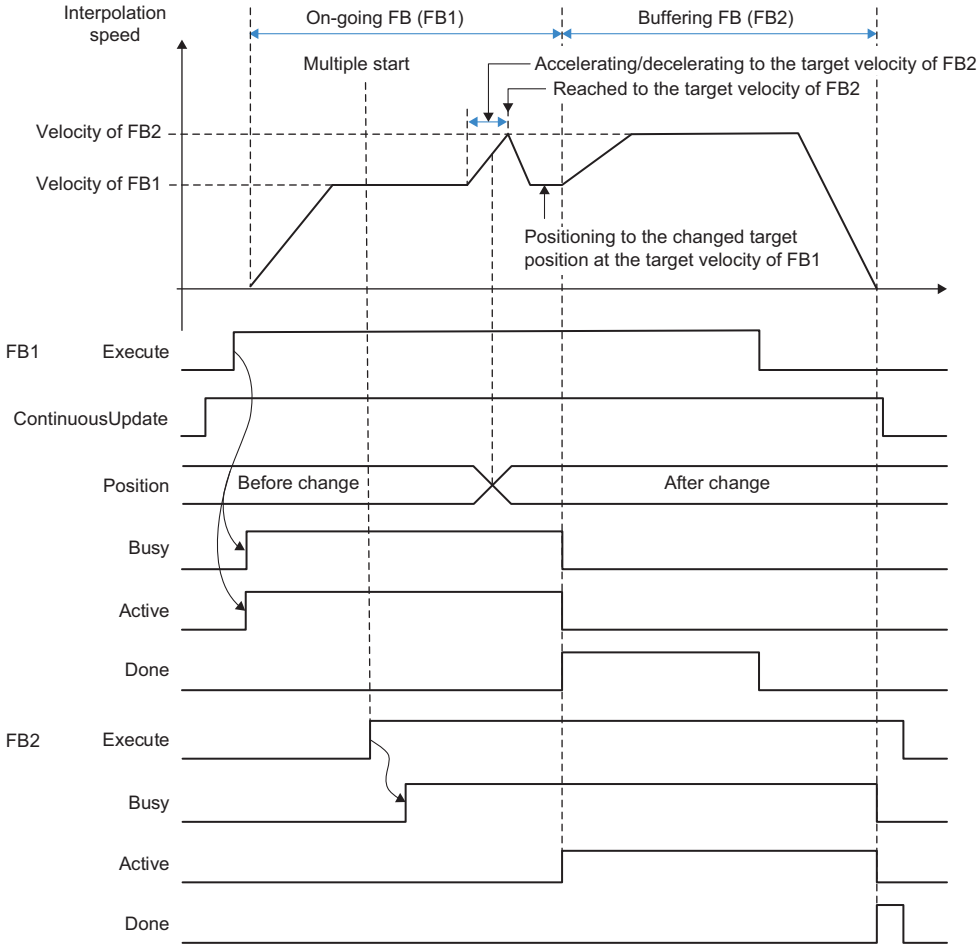
■Target position/movement distance change by retrigger

When the target position/movement distance change is carried out by retrigger, the target velocity (transfer speed) of the buffering FB is maintained and the position to the changed target position/movement distance is fixed.



■Target position/movement distance change by continuous update

When the target position/movement distance change is carried out by continuous update, the velocity accelerates/decelerates to the target velocity (transfer speed) of the buffering FB. After that, it accelerates/decelerates again to the target velocity of on-going FB and positioning to the changed target position/movement distance is fixed.

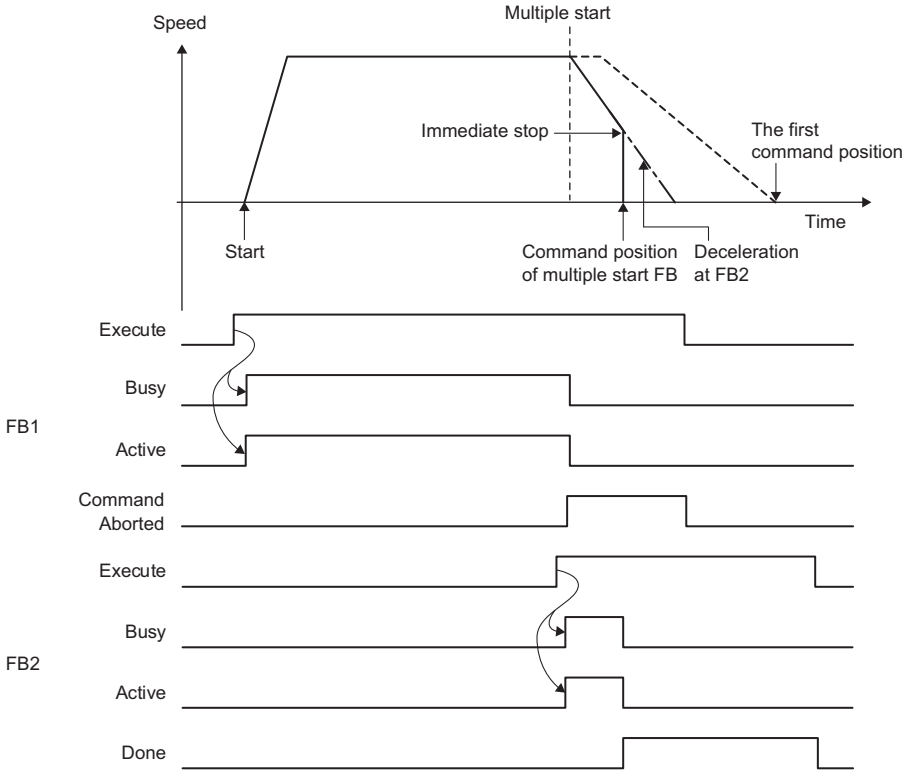


Precautions

When deceleration distance of a buffering FB cannot be secured with the switching speed and the target position is overrun at switching to the buffering FB, automatic deceleration operation is performed according to the AxisName(AxesGroupName).Pr.OverrunOperation settings.

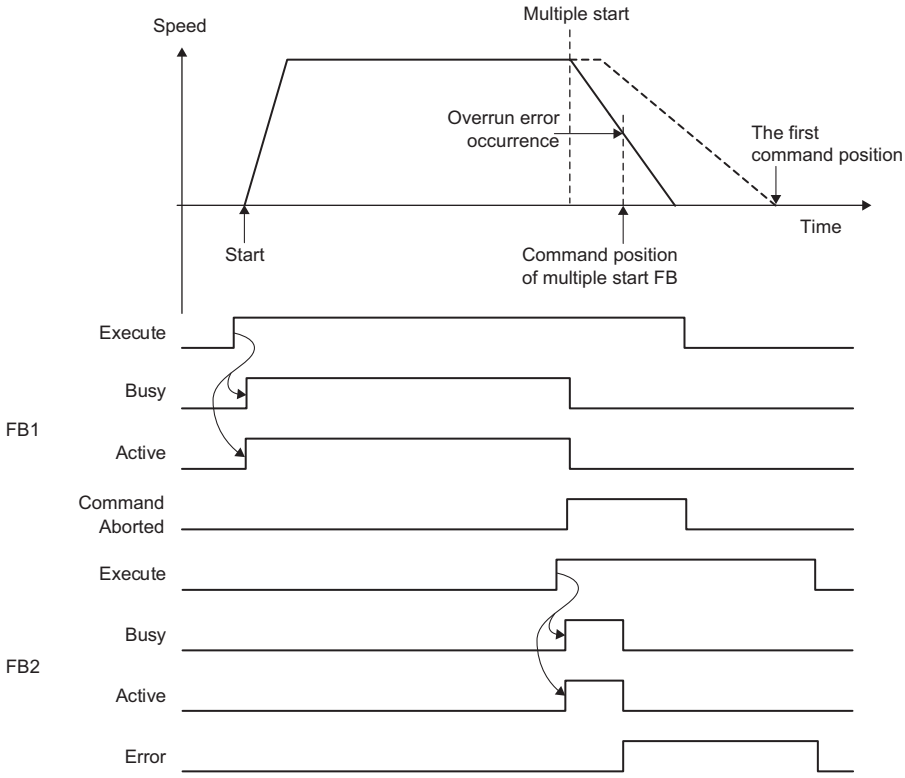
- For "1: ImmediateStop"

The axis starts deceleration immediately, and it stops immediately at reaching the target position after outputting the warning "Overrun Warning" (warning code: 0D10H).



• For "2: KeepCurrentAcc" (Only for single axis)

The axis starts deceleration immediately, and it stops after exceeding the target position. When the stop address is exceeded, the error "Overrun Error" (error code: 1A7EH) is output. However, when the error occurs, the operation stops according to the setting of AxisName.Pr.StopMode_General. To stop the axis after it exceeds the target position, set "2: KeepCurrentAcc" in AxisName.Pr.StopMode_General.



Point

When a new FB is multiple started during deceleration by movement amount shortage, the operation becomes as the following.

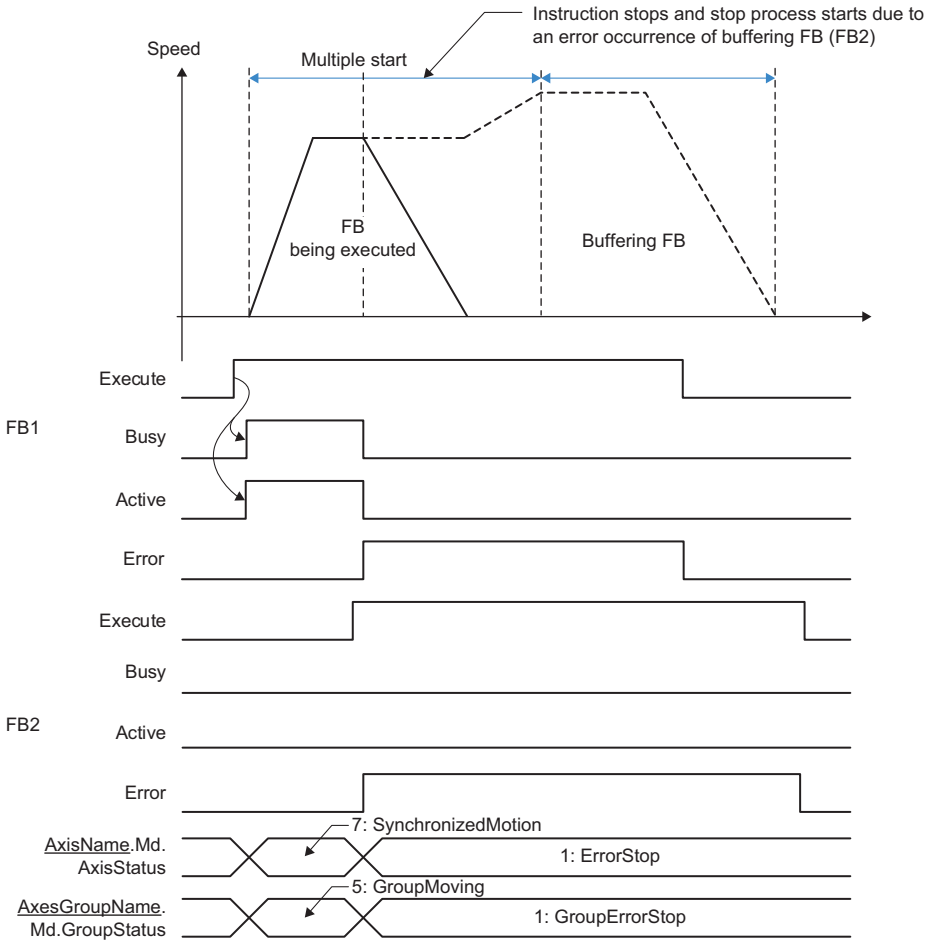
[When AxisName.Pr.OverrunOperation is "2: KeepCurrentAcc" setting]

- If the target position has already been reached when the multiple start happens (when the error "Overrun Error" (error code: 1A7EH) occurs), the multiple started FBs will be disabled.
- If the target position has not been reached when the multiple start happens and if Aborting and Blending are set, deceleration will be canceled. If Buffered is set, the multiple start will be executed but canceled with the error "Overrun Error" (error code: 1A7EH) occurrence.

[When AxisName.Pr.OverrunOperation is other than "2: KeepCurrentAcc" setting]

- If a new FB is multiple started when Aborting and Blending are set, deceleration stop will be canceled.
- If a new FB is multiple started when Buffered is set, automatic deceleration will be continued. The multiple started axes will be executed after the axis stops.

If an error occurs in the following FB analysis processing at multiple start, the axis status (AxisName.Md.AxisStatus) and the axes group status (AxesGroupName.Md.GroupStatus) will be error stop and the on-going FBs will stop.



When the target position or the target velocity is changed at restart or continuous update to on-going FB, the buffering FB may have to be reanalyzed.

If the acceleration/deceleration method settings are different between on-going FB and the next FB, the error "Acceleration/Deceleration Method Mismatched" (error code: 1A0EH) will occur and the FBs will stop. Set the same acceleration/deceleration method in the FBs to be carried out multiple start.

4.4 Stop

This section describes how to stop control.

Each control is stopped in the following cases.

Axis operation stop

- When each control is completed normally
- When the "forced stop input" is turned OFF
- When the drive unit power supply is turned OFF
- When the drive unit network disconnection is detected
- When the drive unit error occurs
- When the forced stop is input to the drive unit
- When "Enable" of MC_Power is FALSE
- When "ServoON" of MC_Power is FALSE
- When the hardware stroke limit upper/lower limit error occurs
- When a CPU module error occurs
- When the "PLC READY [Y0]" is turned OFF
- When the cycle over error occurs
- When a moderate error or a major error of the motion system occurs
- When the software stroke limit upper/lower limit error occurs
- When an axis error is detected
- When "Execute" of MC_Stop is TRUE
- When the "Stop signal" of external input signal is TRUE

Hardware stroke limit upper/lower limit error	Software stroke limit upper/lower limit error
• Error "FLS Signal Detection (at Start)" (error code: 1A2DH) • Error "RLS Signal Detection (at Start)" (error code: 1A2EH) • Error "FLS Signal Detection (Controlling)" (error code: 1A2FH) • Error "RLS Signal Detection (Controlling)" (error code: 1A30H)	• Error "Software Stroke Limit Over (Forward Direction)" (error code: 1A03H) • Error "Software Stroke Limit Over (Reverse Direction)" (error code: 1A04H)

Axes group operation stop

- When each control is completed normally
- When a stop cause occurs in the configuration axis
- When an axes group error is detected
- When "Execute" of MC_GroupStop is TRUE
- When the "Stop signal" of external input signal is TRUE

This section describes stop processing other than when each control in the above cases is completed normally.

For operation details when a stop cause occurs in another function (stop processing of homing, etc.), refer to each function.

Relevant variables/FB

Variable/Structure name	Name	Details
AxisName.Pr.		
StopMode_HwStrokeLimit	Stop selection at hardware stroke limit error occurrence	Selects operation when the hardware stroke limit upper/lower limit error occurs. 1: ImmediateStop (Immediate Stop) (Initial value) 2: KeepCurrentAcc (Keep current acceleration/deceleration) 3: AlternativeAcc (Alternative acceleration/deceleration) (Immediate stop will be executed when executing an instruction without deceleration specification.)
StopMode_SwStrokeLimit	Stop selection at software stroke limit error occurrence	Selects operation when the software stroke limit upper/lower limit error occurs. 1: ImmediateStop (Immediate stop) (Initial value) 2: KeepCurrentAcc (Keep current acceleration/deceleration) 3: AlternativeAcc (Alternative acceleration/deceleration) (Immediate stop will be executed when executing an instruction without deceleration specification.)
StopMode_General	Stop selection at stop by factors	Selects operation when a stop cause for each axis other than the stop cause 1 and 2 occurs. 1: ImmediateStop (Immediate stop) 2: KeepCurrentAcc (Keep current acceleration/deceleration) 3: AlternativeAcc (Alternative acceleration/deceleration) (initial value) (Immediate stop will be executed when executing an instruction without deceleration specification.)
StopMode_Deceleration	Deceleration at stop	Specifies the deceleration at deceleration stop by stop cause occurrence. • When acceleration/deceleration method to specify the acceleration/deceleration Unit: specify with U/s^2. Range: 0.0000, a positive number from 0.0001 to 2,147,483,647 U/s^{2*1} • When acceleration/deceleration method to specify the acceleration/deceleration time Unit: specify with s. Range: 0.0000, a positive number from 0.0001 to 8400.0 s^{*1} If 0 is set, immediately stops regardless of the operation selection at acceleration/deceleration 0 setting.
StopMode_DecelerationCurve	Stop selection at deceleration to stop	Selects operation when a stop cause occurs during deceleration (including a stop cause and automatic deceleration). 0: RapidCurve (Recreate deceleration curve at increased deceleration) 1: OverrideCurve (Recreate deceleration curve) (Initial value) 2: ContinueCurve (Maintain deceleration curve)
StopMode_ServoOff	Process selection at servo OFF command during operation	Selects operation when ServoON of MC_Power turns FALSE during operation. 0: Ignore (Ignore) (Initial value) 4: ServoOffAfterImmediateStop (Servo OFF after immediate stop) 5: ServoOffAfterDecelStop (Servo OFF after deceleration to stop)
OverrunOperation	Operation setting at overrun	Selects operation when stop position is reached during deceleration stop processing by a stop cause. 1: ImmediateStop (Immediate stop) (Initial value) 2: KeepCurrentAcc (Keep current acceleration/deceleration)

Variable/Structure name		Name	Details
	StopSignal	Stop signal	Sets a signal that uses the stop signal (STOP) of the external input signal. StopSignal is a structure of SIGNAL_SELECT type. The specific setting and operation for this signal are below. • StartIO (IO No.) Ignores input values. • Target (Target) If the target is unspecified, it is regarded as invalid signal and it is always the signal undetection status. When unavailable data is set, the error "Out of Parameter Range (Axis)" (error code: 1D80H) is output. • Detection (Signal detection method) The specification is allowed at the following level detection only.When edge detection specification is set, the error "Out of Parameter Range (Axis)" (error code: 1D80H) is output. 0: HighLevel (Detection at TRUE) 1: LowLevel (Detection at FALSE) • CompensationTime (Compensation time) Ignores the input value. • FilterTime (Filter time) Setting range of the filter time is 0.0 to +5.0. When a value outside the range is set, the warning "Out of Filter Time Setting Range of Each Axis Signal Warning" (warning code: 0D24H) is output and the axis operates with 0.0 of the filter time. The fetch timing of the label is at ready ON and the detection timing of the signal is the axis operation cycle. For the details on the SIGNAL_SELECT type, refer to the following. ☞ Page 418 External Signal Selection
	StopOption_DriverTargetIgnored	Driver Command Discard Detection Setting	Select whether to detect the command discard status (Statusword bit12) of the driver module and stop with error during operation of the axis or not. 0: FALSE Detection Disabled 1: TRUE Detection Enabled (Initial value)
AxisName.Md.			
	StopSignal	Stop signal	Displays input status of the stop signal (STOP) of the external input signal. StopSignal is a structure of SIGNAL_SELECT type. The specific output for this signal is below. • StartIO (IO No.) 0 is always displayed. • Target (Target) The fetch result of the parameter is displayed. • Detection (Signal detection method) The fetch result of the parameter is displayed. • CompensationTime (Compensation time) 0.0 is always displayed. • FilterTime (Filter time) The fetch result of the parameter is displayed. For the details on the SIGNAL_SELECT type, refer to the following. ☞ Page 418 External Signal Selection
	StopStatus	Stop status	Displays the input status of stop signal (STOP) among external input signals.
	StopMode_DecelerationCurve	Stop selection at deceleration to stop	Displays input status of the stop processing selection at deceleration stop.
	StopMode_Deceleration	Deceleration at stop	Displays input status of the deceleration at stop.
	StopMode_General	Stop selection at stop by factors	Displays input status of the stop selection at stop cause occurrence.
	StopMode_HwStrokeLimit	Stop selection at H/W stroke limit error occurrence	Displays input status of the stop selection at the hardware stroke limit error occurrence.
	StopMode_ServoOff	Process selection at servo OFF command during operation	Displays input status of the processing selection at servo OFF command during operation.
	StopMode_SwStrokeLimit	Stop selection at S/W stroke limit error occurrence	Displays input status of the stop selection at the software stroke limit error occurrence.
	OverrunOperation	Operation setting at overrun	Displays input status of the overrun operation setting.
	StopOption_DriverTargetIgnored	Driver Command Discard Detection Setting	Displays driver command discard detection setting status
AxesGroupName.Pr.			

Variable/Structure name		Name	Details
	StopMode_General	Stop selection at stop by factors	Selects operation when a stop cause of the axes group occurs. 1: ImmediateStop (Immediate stop) 2: KeepCurrentAcc (Keep current acceleration/deceleration) 3: AlternativeAcc (Alternative acceleration/deceleration) (Initial value) (Immediate stop will be executed when executing an instruction without deceleration specification.)
	StopMode_DecelerationCurve	Stop selection at deceleration to stop	Selects operation when a stop cause occurs during deceleration (including a stop cause and automatic deceleration). 0: RapidCurve (Recreate deceleration curve at increased deceleration) 1: OverrideCurve (Recreate deceleration curve) (Initial value) 2: ContinueCurve (Maintain deceleration curve)
	OverrunOperation	Operation setting at overrun	Selects operation when stop position is reached during deceleration stop processing by a stop cause. If other than "1: ImmediateStop" (Initial value) is set, an error will occur. 1: ImmediateStop (Immediate stop)
	StopMode_ErrorInGroup	Structuring axis operation selection at axis stop by factors	During operation in the axes group, when the driver servo is turned OFF on the configuration axis and an immediate stop axis error occurs, it sets the operation of the axis which an error does not occur. 1: ImmediateStop (Immediate stop) (Default value) 4: ServoOffAfterImmediateStop (Servo OFF after immediate stop)
	StopMode_Deceleration	Deceleration at stop	Specifies the deceleration at deceleration stop by axes group stop cause occurrence. • When acceleration/deceleration method to specify the acceleration/deceleration Unit: Specify with U/s^2. Range: 0.0000, a positive number from 0.0001 to 2147483647.0 U/s^{2*1} • When acceleration/deceleration method to specify the acceleration/deceleration time Unit: Specify with s. Range: 0.000000, a positive number from 0.000001 to 8400.0 s^{*1} If "0" is set, immediate stop will happen regardless of the operation selection at acceleration/deceleration 0 setting.
AxesGroupName.Md.			
	StopMode_Deceleration	Deceleration at stop	Displays input status of the deceleration at stop.
	StopMode_DecelerationCurve	Stop selection at deceleration to stop	Displays input status of the stop processing selection at deceleration stop.
	StopMode_ErrorInGroup	Structuring axis operation selection at axis stop by factors	Displays input status of the configuration axes operation setting at axis stop cause occurrence.
	StopMode_General	Stop selection at stop by factors	Displays input status of the stop selection at stop cause occurrence.
	OverrunOperation	Operation setting at overrun	Displays input status of the overrun operation setting.
System.Pr.			
	StopMode_All	Stop selection at all axes stop by factors	Selects whether to stop immediately or decelerate to stop when an all axes stop cause occurs. 1: ImmediateStop (Immediate stop) 2: KeepCurrentAcc (Keep current acceleration/deceleration) 3: AlternativeAcc (Alternative acceleration/deceleration) (initial value) (Immediate stop will be executed when executing an instruction without deceleration specification.)
	StopMode_AllDeceleration	Deceleration at all axes stop	Specifies the deceleration at deceleration stop when an all axes stop cause occurred. • When acceleration/deceleration methodto specifythe acceleration/deceleration Unit: Specify with U/s^2. Range: 0.0000, a positive number from 0.0001 to 2147483647.0 U/s$^{2\ *1}$ • When acceleration/deceleration method to specify the acceleration/deceleration time Unit: Specify with s. Range: 0.000000, a positive number from 0.000001 to 8400.0 s^{*1} If "0" is set, immediate stop will happen regardless of the operation selection at acceleration/deceleration 0 setting.

*1 Clamp by the upper limit value when the upper limit value is out of the range at stop by factors, and handle as 0 (Immediate stop) when lower limit value is out of the range.

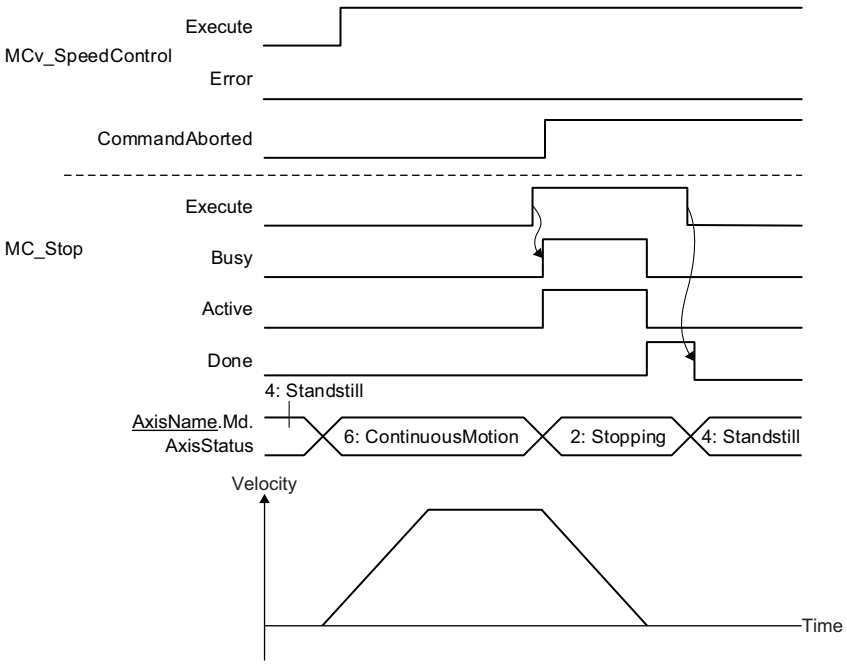
MC_Stop

Item	Description
Function overview	Decelerates specified axis to stop.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

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

■Timing chart

[When the FB is normally completed]



[When an error occurs]

For details when an error occurs, refer to the following.

☞ Page 27 Execute type/Enable type

■I/O variable

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Axis information	Axis	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables

■Input variables

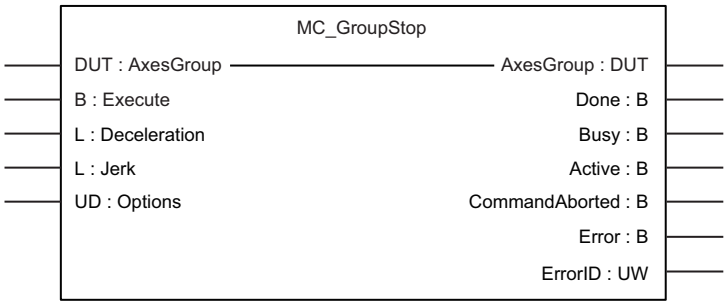
Import ↑: At start, R: Retrigger possible

Name	Variable name	Data type	Import	Setting range	Default value	Description
Execute command	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Deceleration	Deceleration	LREAL	↑/R	☞ Page 335 Input variables to use	0.0	Sets the deceleration.
Jerk (Reserved)	Jerk	LREAL	↑/R	0	0.0	Set "0.0". (If a value other than "0.0" is set, the error "Out of Jerk Range" (error code: 1A13H) will occur.)
Options	Options	DWORD(HEX)	↑	00000000H	00000000H	Set "00000000H". (If a value other than "00000000H" is set, the error "Out of Options Range" (error code: 1A4EH) will occur.)

■Output variables

Name	Variable name	Data type	Default value	Description
Execution completion	Done	BOOL	FALSE	Indicates that the velocity has reached 0.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Controlling	Active	BOOL	FALSE	Indicates that the FB is controlling the axis.
Abortion of execution	CommandAborted	BOOL	FALSE	Indicates that execution is interrupted.
Error	Error	BOOL	FALSE	When TRUE, it indicates that an error has occurred in the FB.
Error code	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned. For details, refer to the following. ☞ Page 662 List of Error Codes

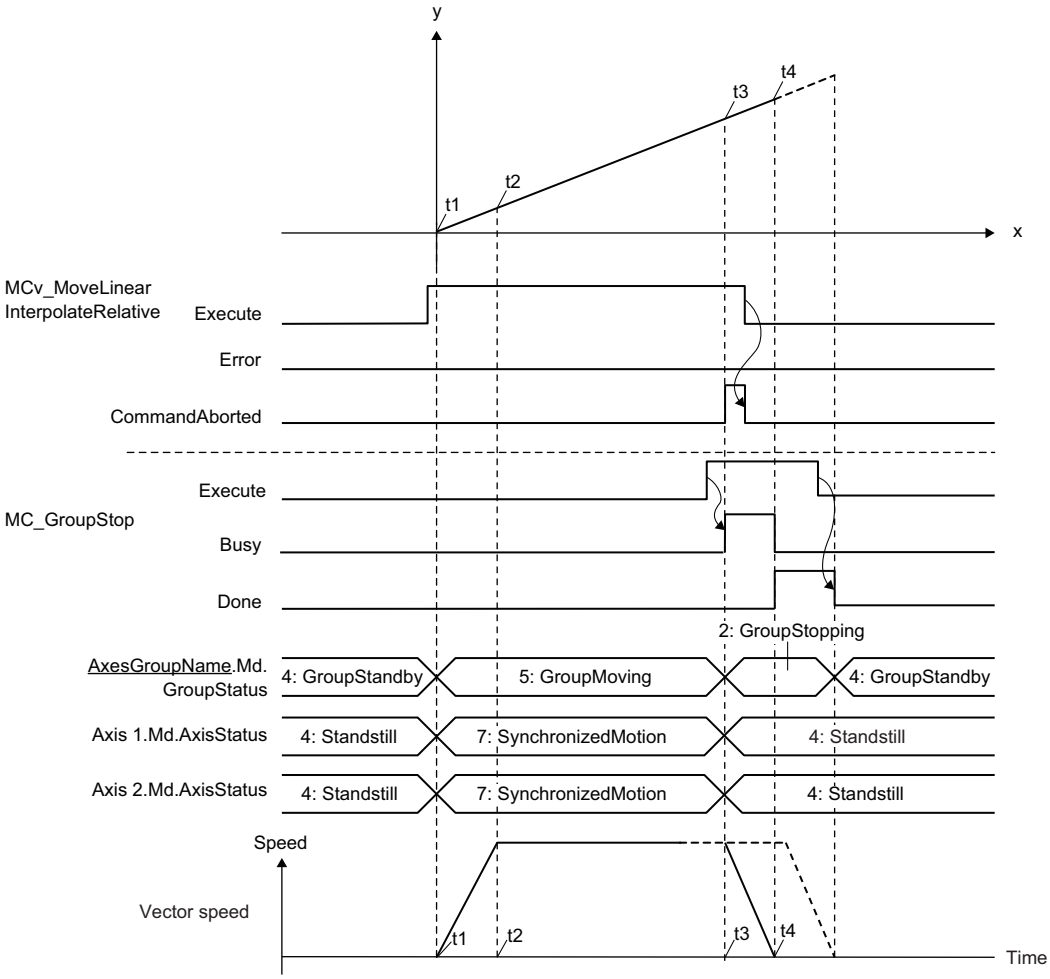
MC_GroupStop

Item	Description
Function overview	Decelerates specified axes group to stop.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

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

■Timing chart

[When the FB is normally completed]



[When an error occurs]

For details when an error occurs, refer to the following.

☞ Page 27 Execute type/Enable type

■I/O variable

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Axes group information	AxesGroup	AXES_GROUP_REF	↑	—	Omission not possible	☞ Page 706 Axes group variables

■Input variables

Import ↑: At start, R: Retrigger possible

Name	Variable name	Data type	Import	Setting range	Default value	Description
Execute command	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Deceleration	Deceleration	LREAL	↑/R	☞ Page 335 Input variables to use	0.0	Sets the deceleration.
Jerk (Reserved)	Jerk	LREAL	↑	0	0.0	Set "0.0". (If a value other than "0.0" is set, the error "Out of Jerk Range" (error code: 1A13H) will occur.)
Options	Options	DWORD(HEX)	↑	00000000H	00000000H	Set "00000000H". (If a value other than "00000000H" is set, the error "Out of Options Range" (error code: 1A4EH) will occur.)

■Output variables

Name	Variable name	Data type	Default value	Description
Execution completion	Done	BOOL	FALSE	Indicates that the velocity has reached 0.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Controlling	Active	BOOL	FALSE	Indicates that the FB is controlling the axis.
Abortion of execution	CommandAborted	BOOL	FALSE	Indicates that execution is interrupted.
Error	Error	BOOL	FALSE	When TRUE, it indicates that an error has occurred in the FB.
Error code	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned. For details, refer to the following. ☞ Page 662 List of Error Codes

Stop cause lists

Stop causes during axis operation

The following shows stop causes during axis operation and the process when each cause occurs.

Cause No.	Stop cause	Stop axis	Axis status (AxisName.Md.AxisStatus)		Stop process[*1]
			During deceleration stop	After stop	
1	"Forced stop input" is FALSE	Each axis/All axes	—	1: ErrorStop	Immediate stop[*2]
2	Drive unit power supply is OFF	Each axis	—	1: ErrorStop	Immediate stop[*2]
	Drive unit network disconnection detection				
	Drive unit error				
	Forced stop input to the drive unit				
	Enable of MC_Power is FALSE				
	ServoON of MC_Power is FALSE (When AxisName.Pr.StopMode_ServoOff is "4: Servo OFF After Immediate Stop")[*3]				
	Drive unit control mode has switched to the control mode which the motion system axis control is not supported.				Immediate stop[*6]
3	Hardware stroke limit upper/lower limit error occurrence	Each axis	1: ErrorStop	1: ErrorStop	Deceleration stop/immediate stop (Follows AxisName.Pr.StopMode_HwStrokeLimit)
4	CPU module error occurrence	All axes	1: ErrorStop	1: ErrorStop	Deceleration stop/immediate stop (Follows System.Pr.StopMode_All)
	PLC READY [Y0] is OFF				
	Cycle over error occurrence				
	Motion system moderate error or major error occurrence				
5	Software stroke limit upper/lower limit error occurrence	Each axis	1: ErrorStop	1: ErrorStop	Deceleration stop/immediate stop (Follows AxisName.Pr.StopMode_SwStrokeLimit)
6	Axis error detection[*4]	Each axis	1: ErrorStop	1: ErrorStop	Deceleration stop/immediate stop (Follows AxisName.Pr.StopMode_General)
7	Execute of MC_Stop is TRUE	Each axis	2: Stopping	2: Stopping[*5]	Deceleration stop (Follows the deceleration set in the FB)
8	"Stop signal" of external input signal is TRUE	Each axis	No change	4: Standstill	Deceleration stop/immediate stop (Follows AxisName.Pr.StopMode_General)

*1 For each operation of the stop process, refer to the following.
☞ Page 172 Stop processes for single axis

*2 The servo turns OFF in the driver side and immediate stop is performed, and also the motion side command is stopped.

*3 Operation differs depending on the setting value selected in AxisName.Pr.StopMode_ServoOff. For details, refer to the following.
☞ Page 177 Stop causes by ServoON input of MC_Power

*4 If an axis error (the error that transits the state into the "1: ErrorStop" status) occurs in the FB which can be linked by the buffer mode, deceleration stop will be performed from the error occurrence.

*5 When Execute of MC_Stop is FALSE at stop completion, the axis status (AxisName.Md.AxisStatus) after stop will be "4: Standstill".

*6 Immediately stops the motion system side command, and follow up on the current position.

Precautions

Provide the emergency stop circuits outside the servo system to prevent cases where danger may result from abnormal operation of the overall system in the event of an external power supply fault or servo system failure.

Stop causes during axes group operation

The following shows stop causes during axes group operation and the process when each cause occurs.

Cause No.	Stop cause		Stop axis	Axes group status (AxesGroupName.Md.GroupStatus)		Stop process[*1]	
				During deceleration stop	After stop		
1	Stop cause occurrence in the configuration axis	Axis error detection	Axes group	1: GroupErrorStop	1: GroupErrorStop	Stop cause occurrence axis	Axes group deceleration stop/immediate stop[*2] (For axes group deceleration stop, it follows AxesGroupName.Pr.StopMode_General)
						Other configuration axes	Axes group deceleration stop/immediate stop (For axes group deceleration stop, it follows AxesGroupName.Pr.StopMode_General)[*3]
2		"Stop signal" of external input signal is TRUE	Axes group	No change	4: GroupStandby	Axes group deceleration stop/immediate stop (Follows AxesGroupName.Pr.StopMode_General)	
3	Axes group error detection[*4]		Axes group	1: GroupErrorStop	1: GroupErrorStop	Axes group deceleration stop/immediate stop (Follows AxesGroupName.Pr.StopMode_General)	
4	Execute of MC_GroupStop is TRUE		Axes group	2: GroupStopping	2: GroupStopping[*5]	Axes group deceleration stop (Follows the deceleration set in the FB)	

*1 For each operation of the stop process, refer to the following.
☞ Page 173 Stop processes for axes group

*2 The servo OFF is executed in the driver side and the axis immediately stops, and also the motion side command is stopped.

*3 When the stop cause occurrence axis is immediately stopped by servo OFF, operation differs depending on the setting selected in AxesGroupName.Pr.StopMode_ErrorInGroup. For details, refer to the following.
☞ Page 184 Stop cause occurrence in the configuration axis

*4 If an axes group error (the error that transits the state into the "1: GroupErrorStop" status) occurs in the FB which can be linked by the buffer mode, automatic deceleration will be performed from the axes group error occurrence.

*5 When Execute of MC_GroupStop is FALSE at stop completion, the axes group status (AxesGroupName.Md.GroupStatus) after stop will be "4: GroupStandby".

Types of stop processes

This section describes types of stop processes for a single axis and an axes group.

Stop processes for single axis

■Deceleration stop

The operation decelerates from the speed during operation to speed 0. The deceleration differs depending on AxisName.Pr.StopMode_General.

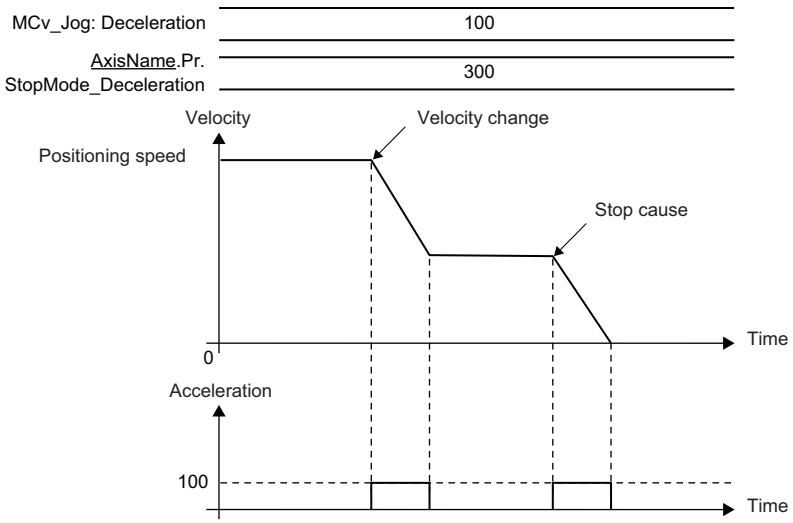
AxisName.Pr.StopMode_General setting value	Deceleration at stop
2: KeepCurrentAcc (Keep current acceleration/deceleration)	Deceleration at instruction during operation
3: AlternativeAcc (Alternative acceleration/deceleration)	AxisName.Pr. StopMode_Deceleration

However, the stop is performed by the MC_Stop execution, the deceleration specified in the Deceleration input is applied. The acceleration/deceleration method and the jerk setting of deceleration stop take over the setting value of the FB which was being executed when stop cause occurred.

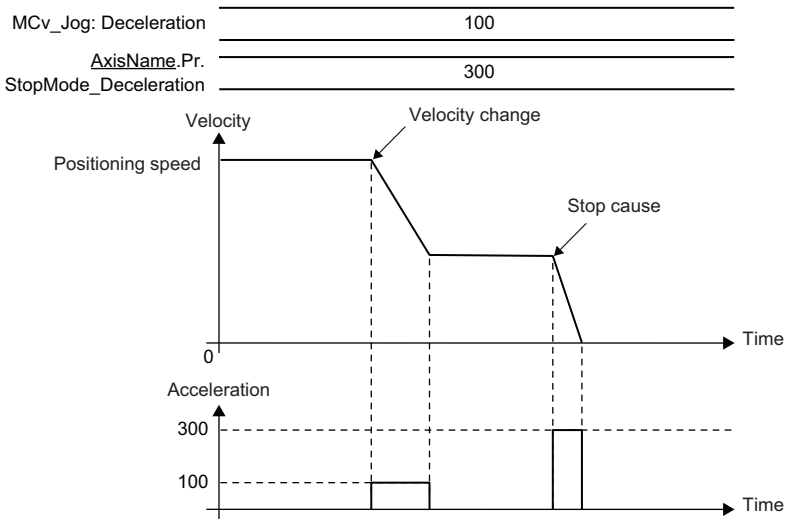
For specifications of each stop method, refer to the following.

☞ Page 330 Acceleration/deceleration Processing Function

- When "2: KeepCurrentAcc" is set to AxisName.Pr.StopMode_General



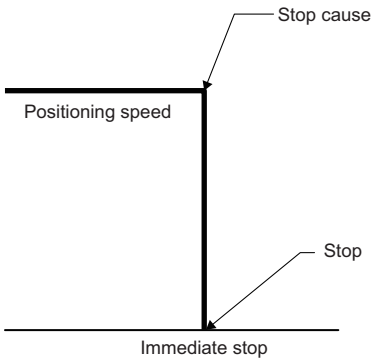
- When "3: AlternativeAcc" is set to AxisName.Pr.StopMode_General



■Immediate stop

The operation does not decelerate and immediately stops the command.

For the stop method of the drive unit, refer to each drive unit manual.



Stop processes for axes group

■Deceleration stop

The operation decelerates from the interpolation speed during operation to the interpolation speed 0. The deceleration differs depending on AxesGroupName.Pr.StopMode_General.

AxesGroupName.Pr.StopMode_General setting value	Deceleration at stop
2: KeepCurrentAcc (Keep current acceleration/deceleration)	Deceleration at instruction during operation
3: AlternativeAcc (Alternative acceleration/deceleration)	AxesGroupName.Pr. StopMode_Deceleration

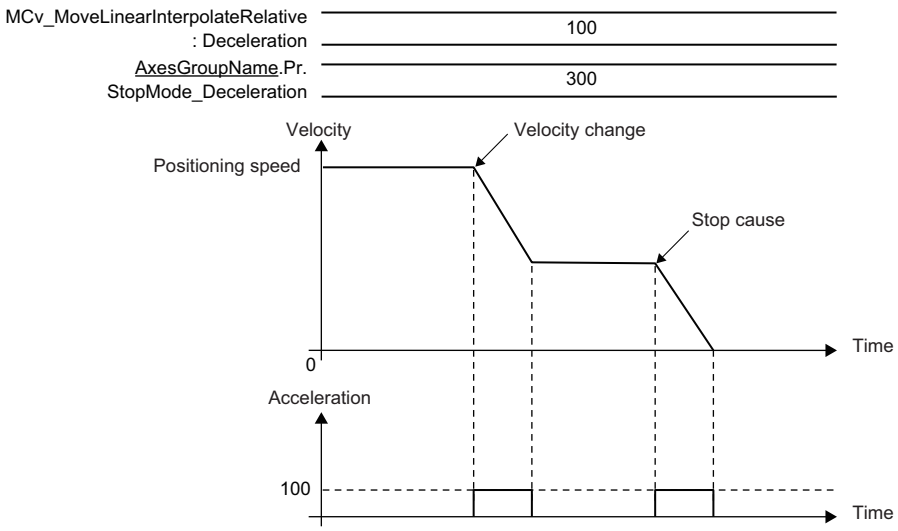
However, the stop is performed by the MC_GroupStop execution, the deceleration specified in the Deceleration input is applied.

The acceleration/deceleration method and the jerk setting of deceleration stop take over the setting value of FB which was being executed when stop cause occurred.

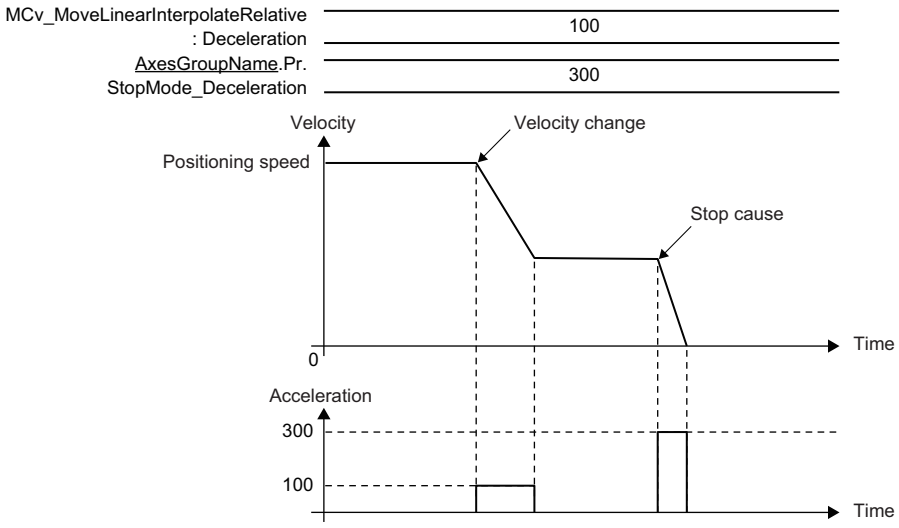
For specifications of each stop method, refer to the following.

☞ Page 330 Acceleration/deceleration Processing Function

- When "2: KeepCurrentAcc" is set to AxesGroupName.Pr.StopMode_General



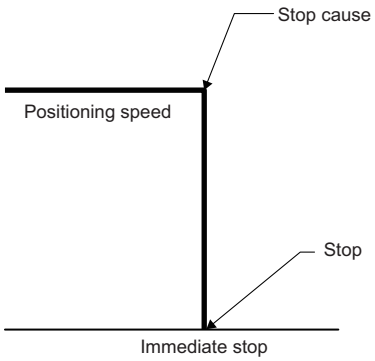
• When "3: AlternativeAcc" is set to AxesGroupName.Pr.StopMode_General



■Immediate stop

The operation does not decelerate and immediately stops the command.

For the stop method of the drive unit, refer to each drive unit manual.



Order of priority for stop process

This section describe processes when multiple stop causes occur.

The order of priority is as shown below based on the stop process of occurred stop cause.

Priority	Stop method
1	Immediate stop
2	Deceleration stop

When a deceleration stop cause occurs again during deceleration stop (including a stop cause and automatic deceleration), the deceleration process differs depending on the following parameter settings for a single axis or an axes group.

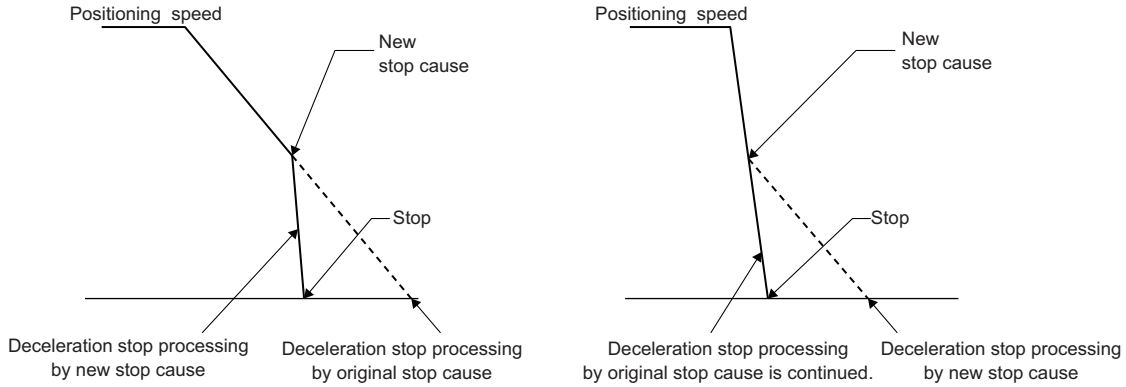
- Single axis: AxisName.Pr.StopMode_DecelerationCurve
- Axes group: AxesGroupName.Pr.StopMode_DecelerationCurve

For AxisName.Pr.StopMode_DecelerationCurve/AxesGroupName.Pr.StopMode_DecelerationCurve, select one of the following three types of operation.

Deceleration curve re-processing for big setting value of the deceleration

0: RapidCurve

A deceleration curve is re-processed when the deceleration setting value of the new stop cause is bigger (when the operation can be stopped earlier).



Deceleration curve re-processing

1: OverrideCurve

A deceleration curve is re-processed from the deceleration of the new stop cause.

During single axis operation, the stop position can be overrun depending on the deceleration and the jerk setting. To avoid overrun, set "1: ImmediateStop" in AxisName.Pr.OverrunOperation.

For operation details, refer to the following.

☞ Page 176 Overrun at stop cause occurrence

Deceleration curve continuation

2: ContinueCurve

The current deceleration curve is continued even if multiple stop causes of deceleration stop occurred.

To stop immediately during deceleration, a stop cause whose stop process is the immediate stop needs to be occurred.

Ex.

Forced stop, Enable of MC_Power is FALSE, etc.

Overrun at stop cause occurrence

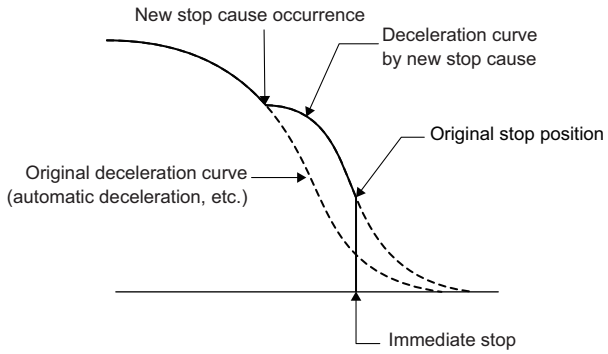
This function selects operation for the case that the positioning address of an instruction which was executed before a stop cause occurs is reached during deceleration stop and immediate stop process.
This function is enabled only when the target position exists at stop cause occurrence.
The following parameters for a single axis or an axes group are need to be set in this function.

- Single axis: AxisName.Pr.OverrunOperation
- Axes group: AxesGroupName.Pr.OverrunOperation

AxisName.Pr.OverrunOperation and AxesGroupName.Pr.OverrunOperation have the following two functions.

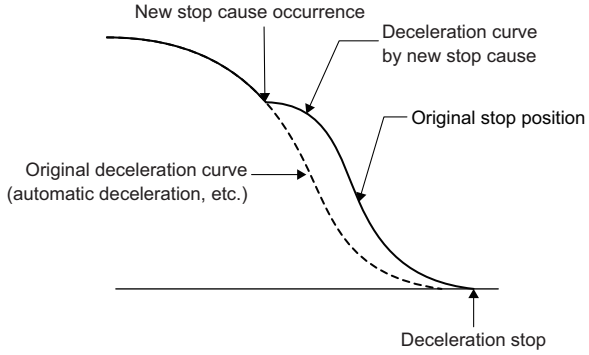
Immediate stop

1: ImmediateStop
The operation is immediately stopped when reaching the original positioning address during deceleration by a stop cause. In this case, the operation does not exceed the positioning address set in the on-going FB.



Keep current acceleration/deceleration (Only for a single axis)

2: KeepCurrentAcc
Deceleration continues even though the operation reached the original deceleration stop position during deceleration by a stop cause. In this case, there is no guarantee that it stops before the positioning address set in the on-going FB.



Stop during single axis operation

This section explains each stop cause.

Stop causes of immediate stop with no condition

When the following stop causes occur, the servo OFF is executed in the drive unit and the axis stops immediately.
For how to stop the drive unit, refer to each drive unit manual.

- "Forced stop input" is turned OFF from an external device
- Drive unit power supply OFF
- Drive unit network disconnection detection
- Drive unit error
- Forced stop input to drive unit
- "Enable" of MC_Power is FALSE

The forced stop input is set in AxisName.Pr.ForcedStop_Signal. For operation details, refer to the following.
☞ Page 192 Forced Stop

Stop causes by ServoON input of MC_Power

■Process selection at servo OFF command during operation

When ServoON of MC_Power is turned FALSE, the stop operation differs depending on the AxisName.Pr.StopMode_ServoOff setting. The processing selection at servo OFF command during operation has the following three functions.

- Ignore

  0: Ignore

During operation, this function ignores the servo OFF command even though it is input and continues on-going operation. If ServoON of MC_Power is turned FALSE when the axis status (AxisName.Md.AxisStatus) of the operation completion axis turns "4: Standstill", servo OFF will be executed. (The axis will not stop.)

- Servo OFF after immediate stop

  4: ServoOffAfterImmediateStop

During operation, the command is immediately stopped by the error "Immediate Stop by Servo OFF During Operation" (error code: 1A3CH) occurrence, and a servo OFF is issued to the driver.

- Servo OFF after deceleration to stop

  5: ServoOffAfterDecelStop

During operation, when receiving the servo OFF command input, the error "Deceleration Stop by Servo OFF During Operation" (error code: 1A3BH) occurs and deceleration stop is executed. If ServoON of MC_Power is FALSE when the operation has reached speed 0, servo OFF will be executed.
The axis type which does not have process selection at servo OFF command during operation operates "4: ServoOffAfterImmediateStop".

■Axis state transition

The axis state changes based on the input combination of Enable and ServoON of MC_Power.

Axis status before input operation (AxisName.Md.AxisStatus)	Input		Processing selection at servo OFF command during operation	Axis status after input operation (AxisName.Md.AxisStatus)
	Enable	ServoON		
4: Standstill (Standby) 0: Disabled (Axis disabled)	TRUE	TRUE	Independent of the setting	4: Standstill (Standby)
		FALSE		0: Disabled (Axis disabled)
	FALSE	TRUE		0: Disabled (Axis disabled)
		FALSE		0: Disabled (Axis disabled)
1: ErrorStop (Error stop)	TRUE	TRUE		1: ErrorStop (Error stop)
		FALSE		1: ErrorStop (Error stop)
	FALSE	TRUE		1: ErrorStop (Error stop)
		FALSE		1: ErrorStop (Error stop)
Other than above	TRUE	TRUE		No transition
		FALSE	0: Ignore (Ignore)	No transition*1*2
			4: ServoOffAfterImmediateStop (Servo OFF after immediate stop)	1: ErrorStop*4 (Error stop)
			5: ServoOffAfterDecelStop (Servo OFF after deceleration to stop)	1: ErrorStop*3 (Error stop)
	FALSE	TRUE	Independent of the setting	1: ErrorStop*4 (Error stop)
		FALSE		1: ErrorStop*4 (Error stop)

*1 It is "0: Disabled" when the axis status (AxisName.Md.AxisStatus) is "4: Standstill" or "2: Stopping" after deceleration stop.

*2 The axis status (AxisName.Md.AxisStatus) will not transit if the axis status (AxisName.Md.AxisStatus) is "1: ErrorStop" after deceleration stop.

*3 If ServoON turns FALSE during operation, the error "Deceleration Stop by Servo OFF During Operation" (error code: 1A3BH) will occur.

*4 When Enable/ServoON turns FALSE during operation, the error "Immediate Stop by Servo OFF During Operation" (error code: 1A3CH) occurs. In this case, on-going FB will be completed or stop immediately, and servo OFF will be executed.

Hardware stroke limit upper/lower limit error occurrence

When the hardware stroke limit upper/lower limit error occurs, the operation stops according to the process selected by AxisName.Pr.StopMode_HwStrokeLimit.

For details, refer to the following.

☞ Page 323 Hardware Stroke Limit

CPU module error occurrence

When the stop error occurs in the control CPU module, PLC READY [Y0] holds ON, and the motion side will be STOP status.

PLC READY [Y0] OFF

When the PLC READY [Y0] is turned OFF, the operation stops according to the process selected by System.Pr.StopMode_All.

Cycle over error occurrence

When the cycle over error occurs in operation cycle processing or buffer memory refresh processing, the operation stops according to the process selected by System.Pr.StopMode_All.

According to the set values of System.PrConst.OperationCycle[1..1].Cycle and System.PrConst.BuffermemoryRefreshCycle.Cycle, different error code is output in every axis.

Setting value	Error
2: MinorError (Minor error)	The error "Cycle Over" (error code: 1C80H)
3: ModerateError (Moderate error)	The error "Module Error During Operation" (error code: 1AB1H)*1

*1 It causes the operation stop as the stop cause at the motion system moderate error or major error occurrence.

Software stroke limit upper/lower limit error occurrence

When the software stroke limit upper/lower limit error occurs, the operation stops according to the process selected by AxisName.Pr.StopMode_SwStrokeLimit.

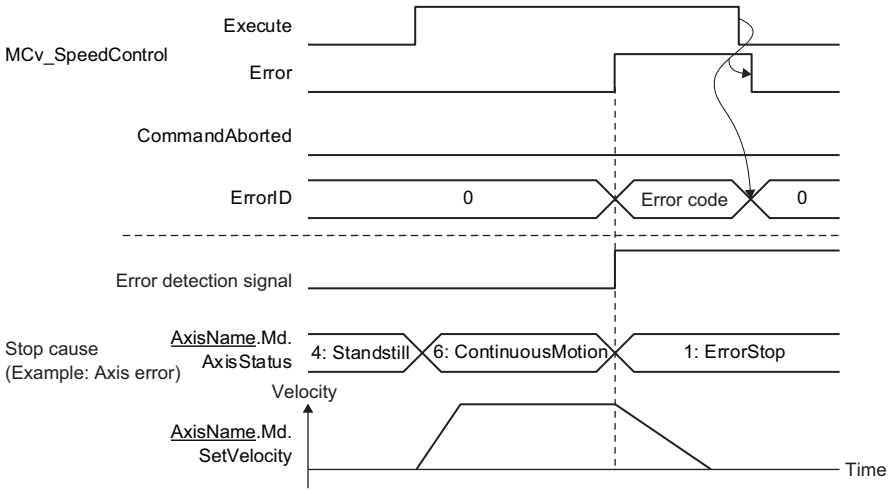
For details, refer to the following.

☞ Page 317 Software Stroke Limit

Axis error detection

When an axis error other than the hardware stroke limit upper/lower limit error, the CPU module error, an error by the PLC READY [Y0] OFF, the error "Cycle Over" (error code: 1C80H), the software stroke limit upper/lower limit error, and an axis error other than the drive unit error is detected, the operation stops according to the process selected by AxisName.Pr.StopMode_General. In this case, the axis status (AxisName.Md.AxisStatus) transits to "1: ErrorStop".

• Operation of the FB that received a stop cause by axis error occurrence



■Driver command discard detection

By setting AxisName.Pr.StopOption_DriverTargetIgnored, when the driver unit Statusword Bit12 turns ON to OFF during axis operation at the real drive axis, the error "Driver Command Discard Detection" (error code: 1AE6H) is output and can stop the command.

When the error "Driver Command Discard Detection" (error code: 1AE6H) occurs, a stop error may occur before and after of its occurrence as the detection factor. Confirm the details before and after of the error with the event history.

When executing the limit detection or forced stop at the driver unit side, the controller side command can be stopped to corresponds to the driver side stop operation. (The Statusword of the connected real drive axis can be monitored by AxisName.Md.Io_Statusword.)

For details, refer to the following.

☞ Page 323 Hardware Stroke Limit

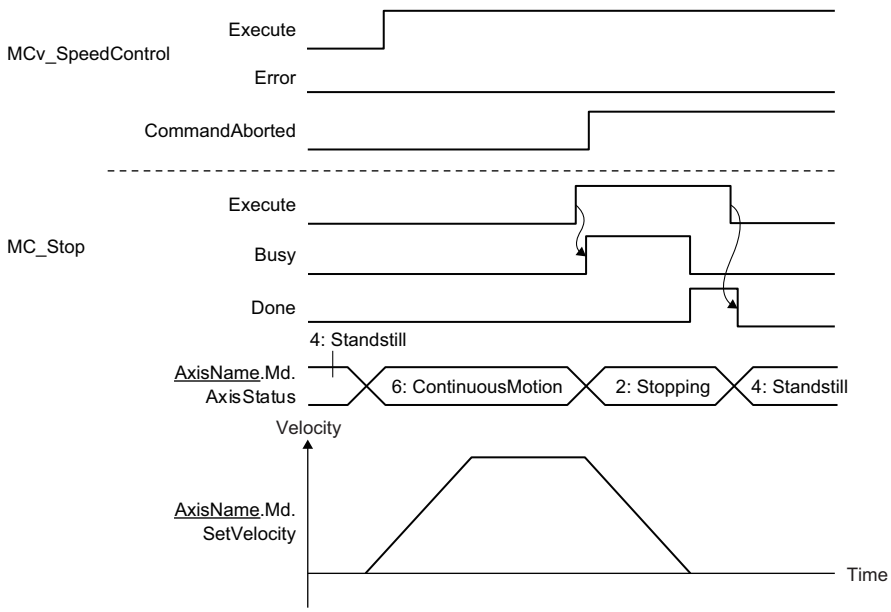
The details of Statusword Bit12 differs depending on the connect driver unit control mode. Also, for Statusword changing conditions etc., refer to the connected device driver unit manuals.

Driver control mode	Statusword[Obj. 6041h]Bit12 abbreviation	Description
Cyclic synchronous position mode (csp)	Target position ignored	0: Target position [Obj. 607Ah] discarding
Cyclic synchronous velocity mode (csv)	Target velocity ignored	0: Target velocity [Obj. 60FFh] discarding
Cyclic synchronous torque mode (cst)	Target torque ignored	0: Target torque [Obj. 6071h] discarding
Continuous operation to torque control mode (ct)	Target torque ignored	0: Target torque [Obj. 6071h] discarding
Other than the above	—	Not checked

MC_Stop (Forced stop)

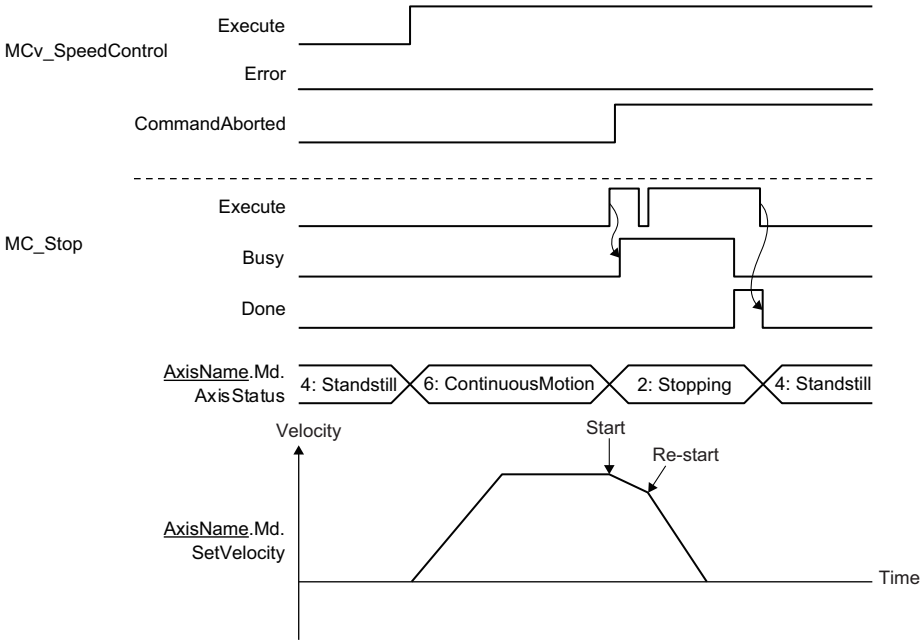
In MC_Stop, the deceleration (Deceleration) is set and on-going instruction is decelerated to a stop.
When MC_Stop is executed, CommandAborted of on-going instruction turns TRUE and the axis status (AxisName.Md.AxisStatus) transits to the "2: Stopping" status. While Execute is TRUE or while the speed does not reach 0, the "2: Stopping" status is maintained. When Done turns TRUE and Execute turns FALSE at stop completion, the axis status changes to the "4: Standstill" status.

■MCv_Stop execution during MC_SpeedControl



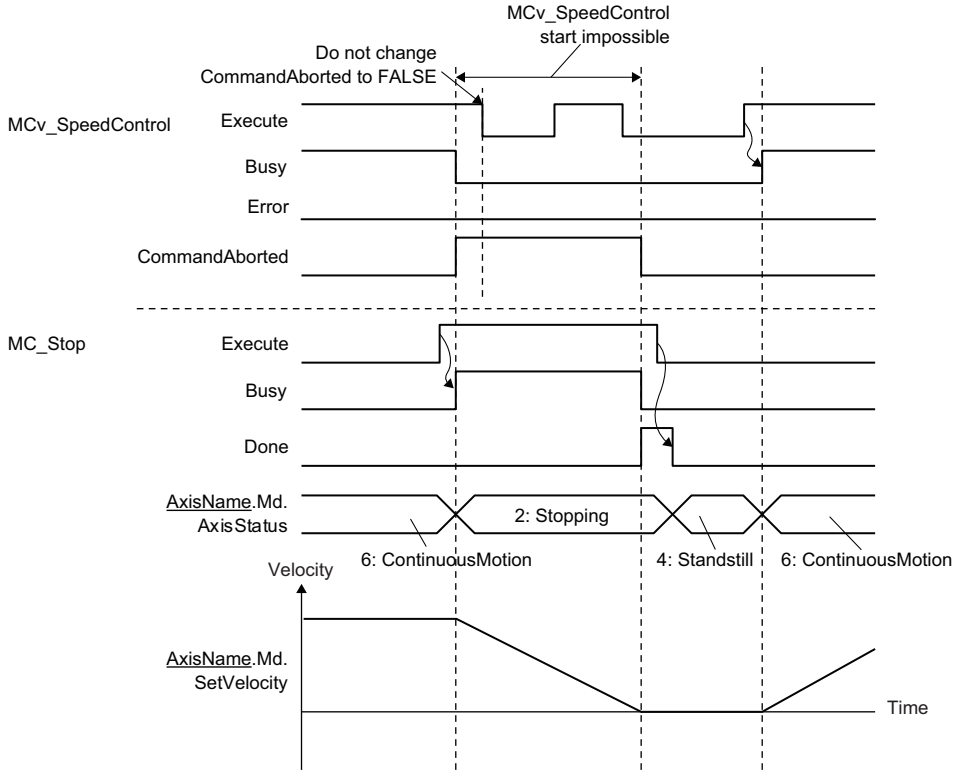
■Operation description

- During deceleration stop by this FB and while Execute (start) of the input variable is TRUE, any operation command other than MC_Stop is not accepted.
- If the deceleration is set to 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 is executed during single axis synchronous control, synchronization to the master axis will be canceled.
- When the setting value of Deceleration is changed and MC_Stop is restarted
  A deceleration stop is executed from the restart point based on Deceleration set in MC_Stop.



- When an FB that operates axes other than MC_Stop is executed while the axis status (AxisName.Md.AxisStatus) is the "2: Stopping" status
  The error "Start Not Possible" (error code: 1AADH) occurs and Error of MC_Stop turns TRUE, and then the axis status (AxisName.Md.AxisStatus) changes to "1: ErrorStop".
- After axis stop, the axis status can be transited to "4: Standstill" by executing AxisName.Cd.ErrorReset.

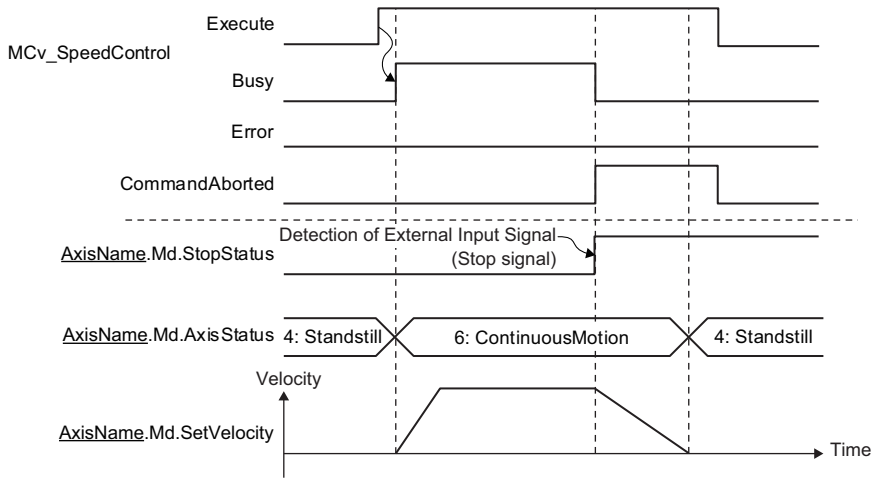
- Even if Execute of command aborted FB by MC_Stop set to FALSE, once the stop operation is started, TRUE of CommandAborted will be continued until it is completed. After it stops completely, changes CommandAborted to FALSE if Execute of command aborted FB is FALSE.



"Stop signal" of external input signal is TRUE

The operation executed now is stopped by turnning the stop signal set in AxisName.Pr.StopSignal TRUE.

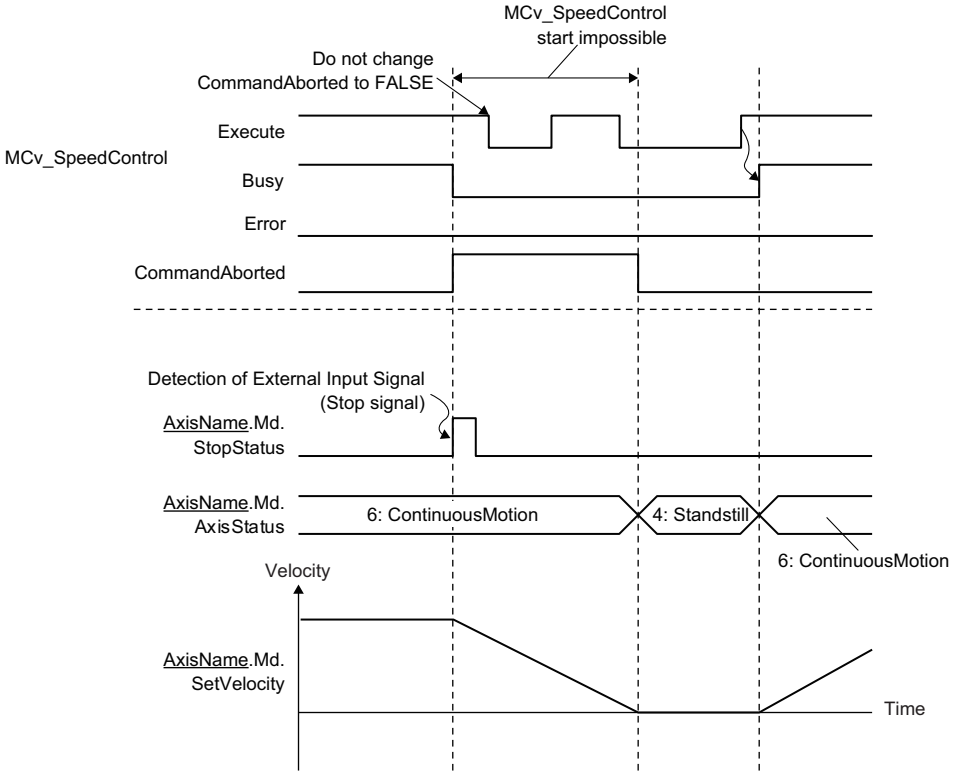
- The operation stops by the operation set in AxisName.Pr.StopMode_General.
- The operation which is executed now turns CommandAborted to TRUE by changing the stop signal to TRUE. When the speed reached 0, the axis status (AxisName.Md.AxisStatus) transits to "4: Standstill".



Point

- When the process is stopped by TRUE of AxisName.Pr.StopSignal, the axis status (AxisName.Md.AxisStatus) after stop will not change.
- The deceleration can be changed by executing MC_Stop while process stop is executed by TRUE of AxisName.Pr.StopSignal.
- The deceleration can be changed when AxisName.Pr.StopSignal is TRUE during MC_Stop execution. In this case, the axis status (AxisName.Md.AxisStatus) after deceleration stop will not change.

- Even if Execute of command aborted FB is set to FALSE by the stop signal, TRUE, once the stop operation is started, TRUE of CommandAborted will be continued until it is completed. After it stops completely, changes CommandAborted to FALSE if Execute of command aborted FB is FALSE.



Stop during axes group operation

This section explains each stop cause.

Stop cause occurrence in the configuration axis

The following shows axes group operation when a stop cause of the configuration axis occurs.

Stop cause occurrence axis				Axes group	
Cause No.	Stop cause	Axis status at stop cause occurrence (AxisName.Md.AxisStatus)	Stop process	Axes group status (AxesGroupName.Md.GroupStatus)	Stop process
1	"Forced stop input" is OFF	1: ErrorStop	Immediate stop*1	1: GroupErrorStop	Stop cause occurrence axis: Immediate stop Other configuration axes: Axes group deceleration stop/immediate stop (Follows AxesGroupName.Pr.StopMode_ErrorInGroup)
2	Drive unit power supply is OFF	1: ErrorStop			
	Drive unit network disconnection detection				
	Drive unit error				
	Forced stop input to the drive unit				
	Enable of MC_Power is FALSE				
	ServoON of MC_Power is FALSE (When AxisName.Pr.StopMode_ServoOff is "4: ServoOffAfterImmediateStop")				
3	Hardware stroke limit upper/lower limit error occurrence	1: ErrorStop	Based on the axes group stop process		Axes group deceleration stop/immediate stop (Follows AxesGroupName.Pr.StopMode_General)
4	CPU module error occurrence	1: ErrorStop			
	PLC READY [Y0] is OFF				
	Cycle over error occurrence				
5	Software stroke limit upper/lower limit error occurrence	1: ErrorStop			
6	Axis error detection	1: ErrorStop			
7	Execute of MC_Stop is TRUE	1: ErrorStop*2			
8	"Stop signal" of external input signal is TRUE	No change		No change	Axes group deceleration stop/immediate stop (Follows AxesGroupName.Pr.StopMode_General)

*1 The servo OFF is executed in the driver side and the axis immediately stops, and also the motion side command is stopped.

*2 If a single axis FB is executed during axes group operation, an axis error will occur and the single axis FB will be disabled. For details, refer to the following.

☞ Page 187 Axis error

When a stop cause has occurred in a configuration axis of the axes group, operation differs depending on the following four patterns.

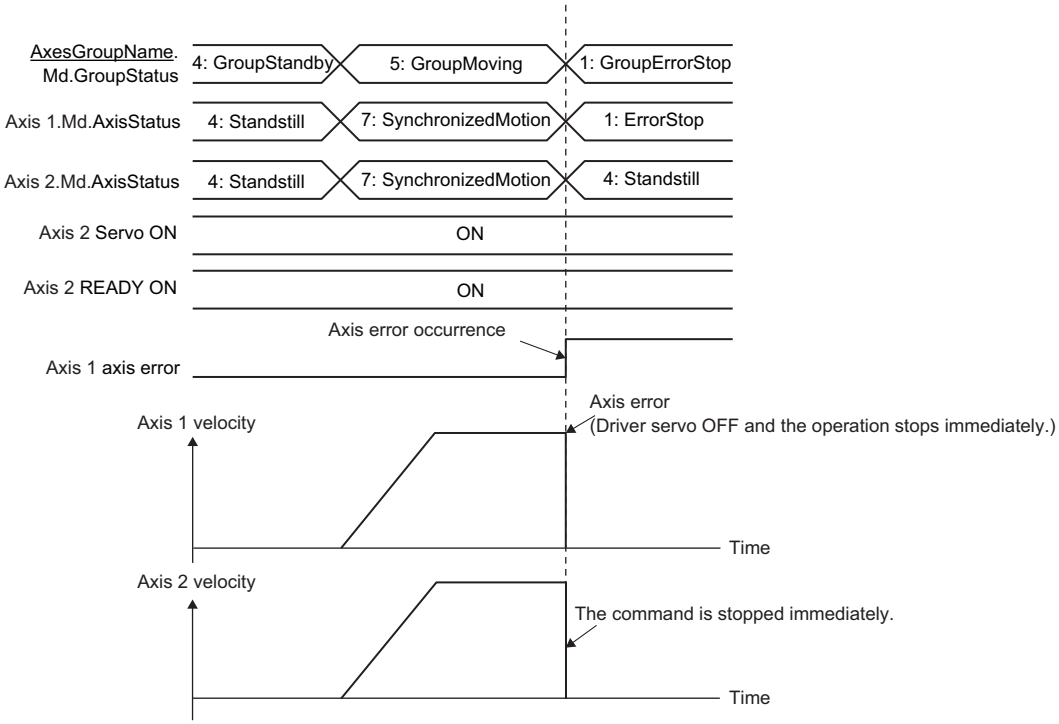
■For an axis error (driver servo OFF and immediate stop)

The error "Stop Cause of Axes Group Configuration Axis" (error code: 1A3DH) occurs in the axes group.
Also, if the driver servo OFF is executed and the configuration axis in which the stop cause occurred stops immediately, the other configuration axes will be operated according to the stop process selected in AxesGroupName.Pr.StopMode_ErrorInGroup.
When a stop cause has occurred in a configuration axis during the axes group operation, one of the following three methods can be selected to stop the other configuration axes.

- Immediate stop

  1: ImmediateStop

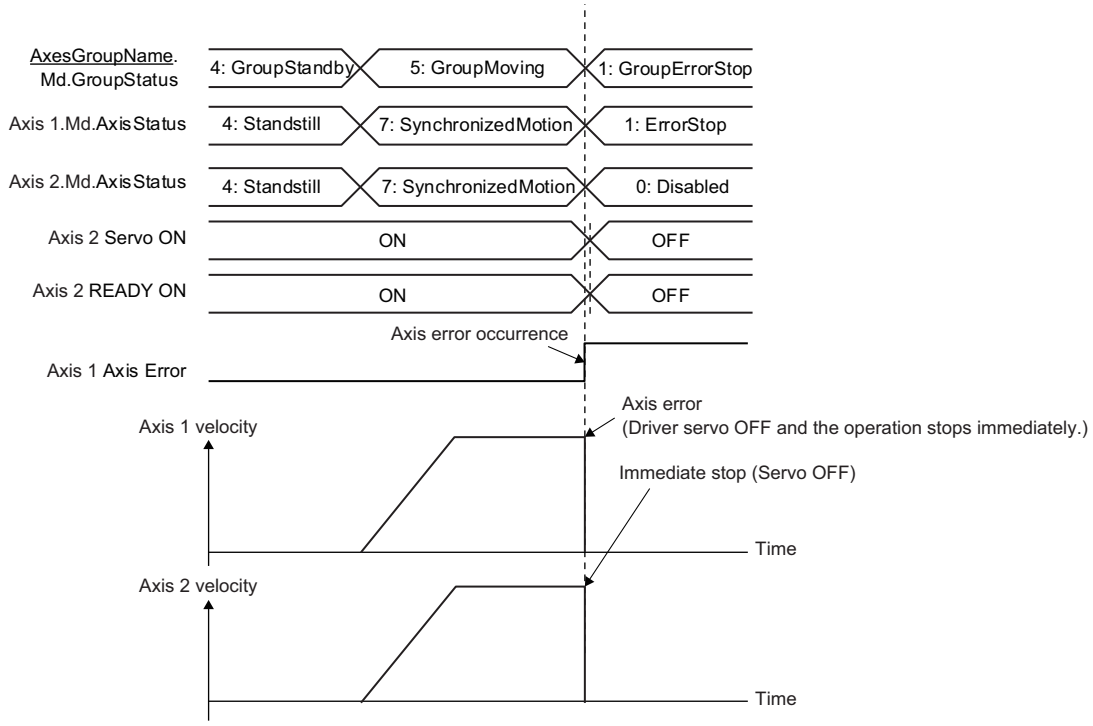
In the axes group configuration axes other than the axis which an axis error has occurred in, a command to the drive unit is stopped.



• Immediate stop and servo OFF

4: ServoOffAfterImmediateStop

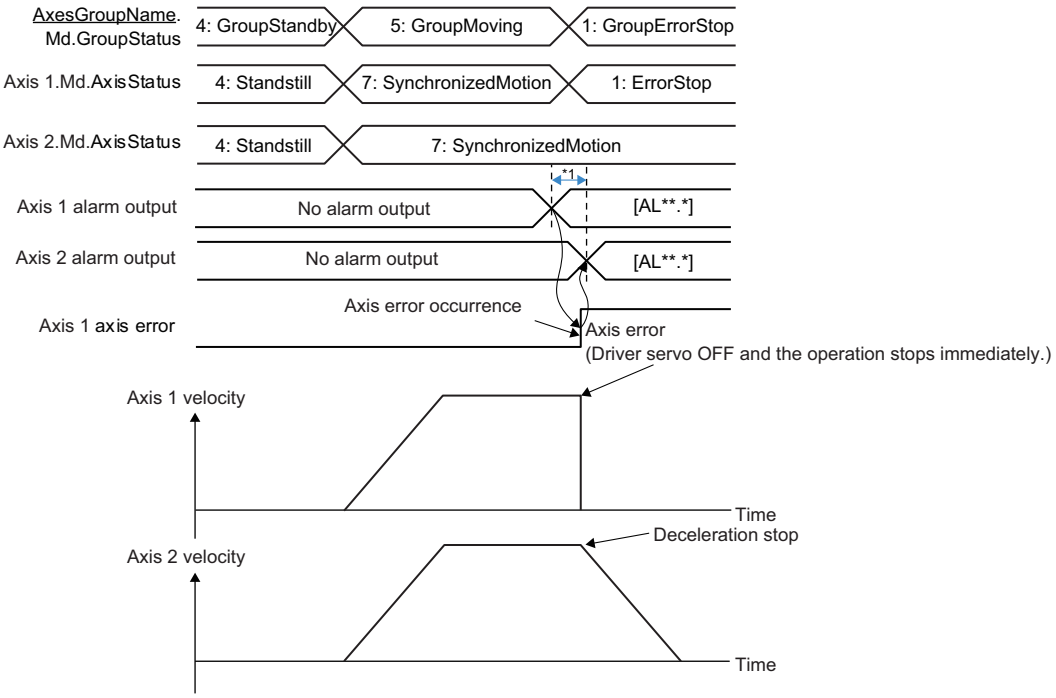
In the axes group configuration axes other than the axis which an axis error has occurred in, a command to the drive unit is stopped, and for real drive axes, the servo OFF command is issued to the drive unit.



• Servo OFF after deceleration to stop

5: ServoOffAfterDecelStop

The axes group configuration axes other than the axis which an axis error has occurred in decelerate to a stop at the deceleration set in AxesGroupName.Pr.StopMode_Deceleration. (The axes during servo OFF will be in the follow up status.)



*1 If a stop cause that performs servo OFF has occurred in the configuration axes, the axis error and the axes group error will be detected in the same operation cycle as the axis error. However, alarm output of the drive unit will be delayed by several cycles to the axis error.

■Axis error

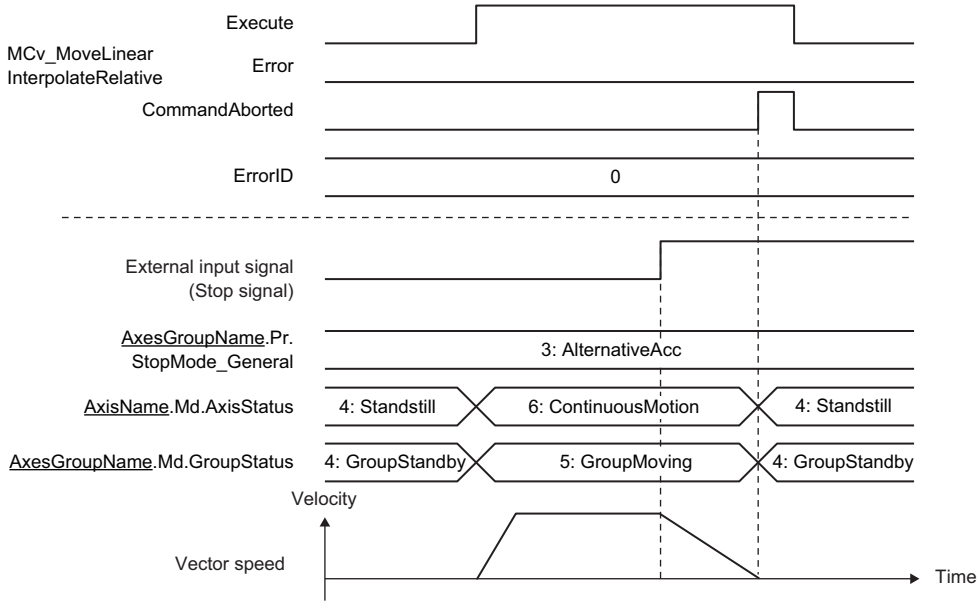
If an axis error has occurred in the configuration axis, the error "Stop Cause of Axes Group Configuration Axis" (error code: 1A3DH) will occur in the axes group. In this case, the axes group will pass through the path and stop according to the AxesGroupName.Pr.StopMode_General setting.

If a single axis control FB is executed to the configuration axis being operated in the axes group, an error will occur in the single axis control FB, and the specified axis will change to the "Instruction issue to the configuration axis being operated in the axes group" and then the "1: ErrorStop" status.

By an error occurrence in the configuration axis, the error "Stop Cause of Axes Group Configuration Axis" (error code: 1A3DH) occurs in the axes group, and the axes group will stop according to the process selected in AxesGroupName.Pr.StopMode_General.

■"Stop signal" of external input signal is TRUE

When AxisName.Pr.StopSignal is turned TRUE to the configuration axis, the axis stops according to the setting of AxesGroupName.Pr.StopMode_General. The axes group will pass through the path and decelerate to a stop. When the speed reaches 0, CommandAborted will turn TRUE, and the axes group status (AxesGroupName.Md.GroupStatus) will transit to "4: GroupStandby". The value set in AxesGroupName.Pr.StopMode_Deceleration will be used for the deceleration at this time.



Point

- When stopping by turning AxisName.Pr.StopSignal to TRUE, the axes group status (AxesGroupName.Md.GroupStatus) after stop will not change.
- When the stop processing is being executed by turning AxisName.Pr.StopSignal to TRUE, the deceleration can be changed by executing MC_GroupStop.
- The deceleration can be changed by turning AxisName.Pr.StopSignal to TRUE during MC_GroupStop execution. In this case, the axes group status (AxesGroupName.Md.GroupStatus) after the deceleration stop will not change.

■Hardware stroke limit error/Software stroke limit error

When any of the following conditions is satisfied, the axes group will immediately stop regardless of the axes group setting. In this case, the error "Stop Cause of Axes Group Configuration Axis" (error code: 1A3DH) occurs in the axes group and its status changes to the "1: GroupErrorStop".

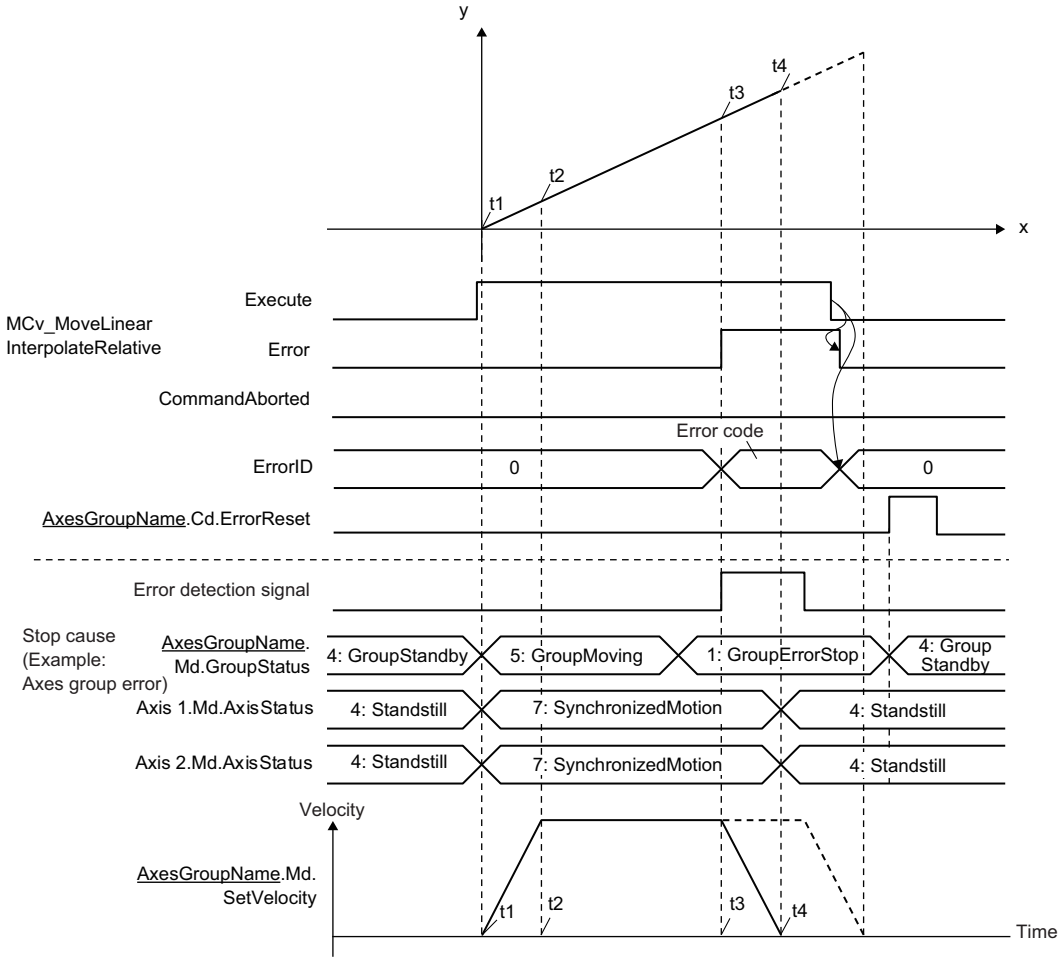
- When the stop cause "Hardware stroke limit error" is occurring in any axis of configuration axes, "1: ImmediateStop" is set to AxisName.Pr.StopMode_HwStrokeLimit in one or more axes.
- When the stop cause "Software stroke limit error" is occurring in any axis of configuration axes, "1: ImmediateStop" is set to AxisName.Pr.StopMode_SwStrokeLimit in one or more axes.

Hardware stroke limit error	Software stroke limit error
• Error "FLS Signal Detection (at Start)" (error code: 1A2DH) • Error "RLS Signal Detection (at Start)" (error code: 1A2EH) • Error "FLS Signal Detection (Controlling)" (error code: 1A2FH) • Error "RLS Signal Detection (Controlling)" (error code: 1A30H)	• Error "Software Stroke Limit Over (Target Position)" (error code: 1A00H) • Error "Software Stroke Limit Over (Start Position)" (error code: 1A01H) • Error "Software Stroke Limit Over (Forward Direction)" (error code: 1A03H) • Error "Software Stroke Limit Over (Reverse Direction)" (error code: 1A04H)

Axes group error detection

When an axes group error other than the "Stop cause occurrence in the configuration axis" is detected, the operation will be stopped according to the process selected by AxesGroupName.Pr.StopMode_General. The value set in AxesGroupName.Pr.StopMode_Deceleration will be used for the deceleration at this time.

- Operation of the FB that received a stop cause by axes group error occurrence

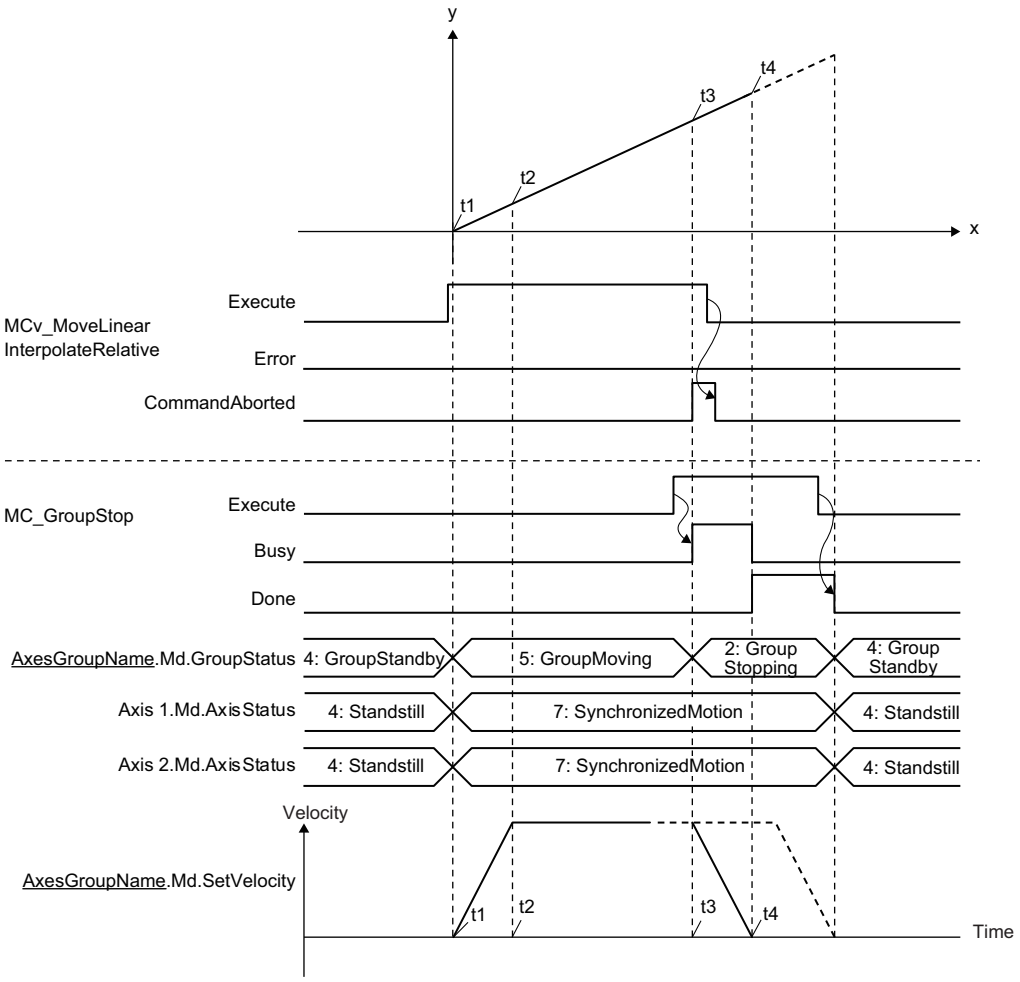


MC_GroupStop (Group forced stop)

In MC_GroupStop, the deceleration (Deceleration) is set and on-going instruction is decelerated to a stop. The axes group passes through the previous operation path and decelerates to a stop.

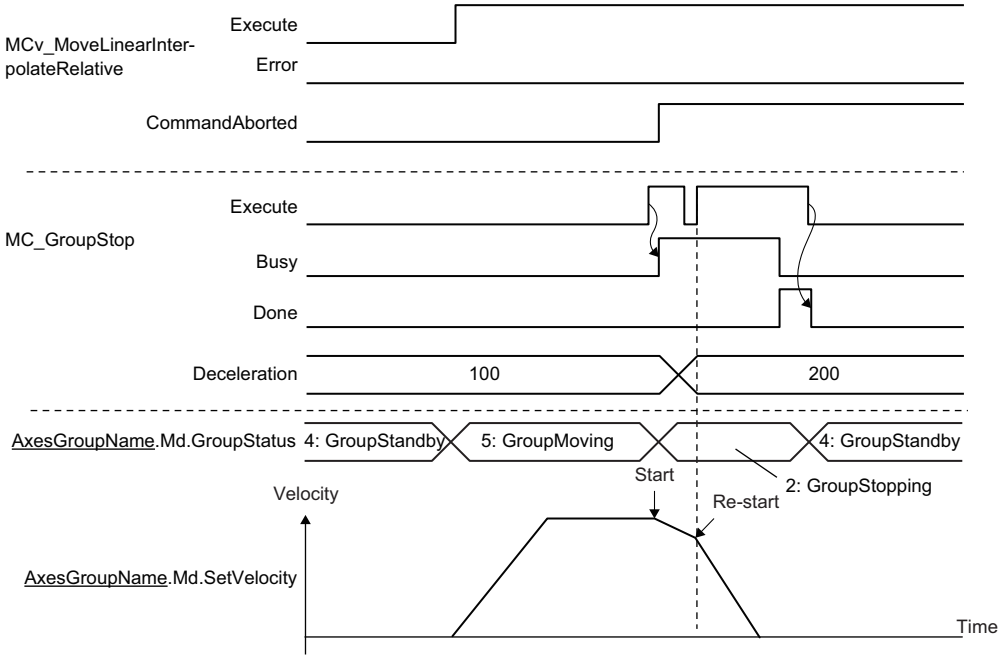
When MC_GroupStop is executed, CommandAborted of on-going instruction turns TRUE and the axes group status (AxesGroupName.Md.GroupStatus) transits to the "2: GroupStopping" status. While Execute is TRUE or while the speed does not reach 0, the "2: GroupStopping" status is maintained. When Done turns TRUE and Execute turns FALSE at stop completion, the axes group status changes to the "4: GroupStandby" status.

■MC_GroupStop execution during MCv_MoveLinearInterpolateRelative



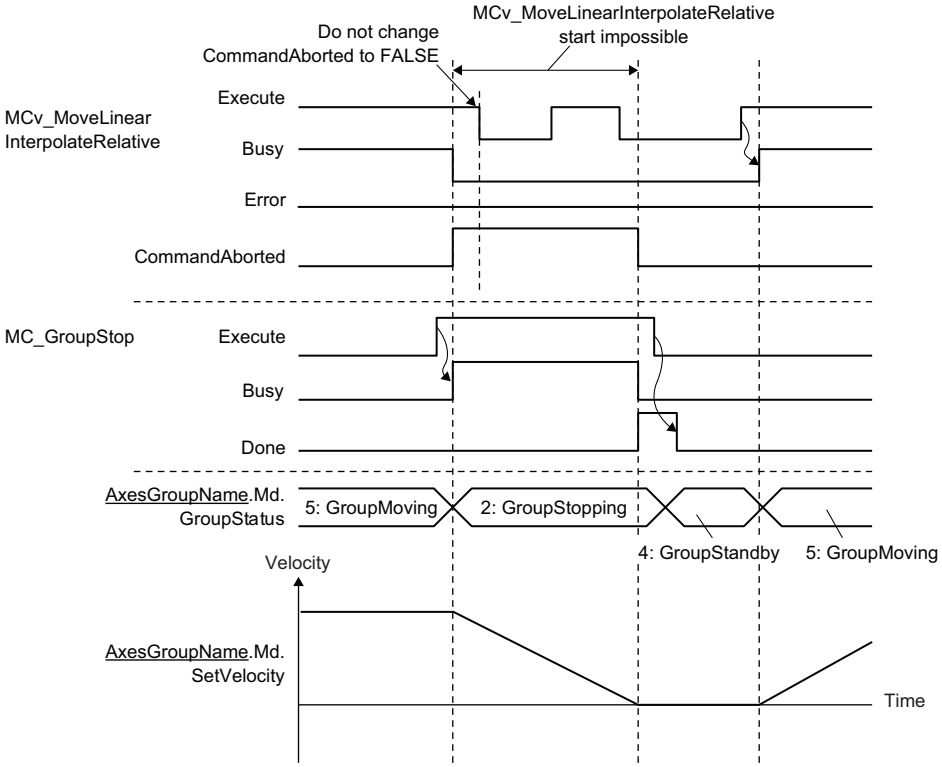
■Operation description

- During deceleration stop by this FB and while Execute (start) of the input variable is TRUE, any operation command other than MC_GroupStop is not accepted.
- If the deceleration is set with 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.
- When the setting value of Deceleration is changed and MC_GroupStop is restarted
  A deceleration stop is executed from the restart point based on Deceleration set in MC_GroupStop and Jerk used in the on-going instruction.



- When an FB that operates axes other than MC_GroupStop is executed while axes group status (AxesGroupName.Md.GroupStatus) is the "2: GroupStopping" status, the error "Start Not Possible" (error code: 1AADH) occurs and Error of MC_GroupStop turns TRUE, and then the axes group status (AxesGroupName.Md.GroupStatus) changes to "1: GroupErrorStop".
- If MC_GroupStop is executed when MC_ MoveDirectAbsolute and MC_ MoveDirectRelative are carried out, the parameter values set in each axis use speed/acceleration/deceleration/jerk.

- Even if Execute of command aborted FB is set to FALSE by MC_GroupStop, once the stop operation is started, TRUE of CommandAborted will be continued until it is completed. After it stops completely, changes CommandAborted to FALSE if Execute of command aborted FB is FALSE.



4.5 Forced Stop

This function stops axes with the forced stop signal.

The axis types which the forced stop is valid are as follows

- The forced stop signal of each axis is valid for the axis type which supports AxisName.Pr.ForcedStop_Signal.
  For its availability of every axis type, refer to the following.
  ☞ Page 716 List of Supported Axis Type
- All axes forced stop signal stops axes of all types including the types which unsupport AxisName.Pr.ForcedStop_Signal

Refer to the manual of the driver unit after the forced stop signal is sent.

⚠WARNING

- When the forced stop is required to be wired, ensure to wire it in the negative logic, and b-contact is recommended.
- When using the forced stop function, confirm the wiring and the settings, and check whether it works or not at startup.

Point

The forced stop function immediately stops the instruction and issues "Quick Stop" to the slave device at the same time. For operation when "Quick Stop" is issued, refer to the manual of the slave device.

Relevant variables

Variable/Structure name	Name	Details
AxisName.Pr.		
ForcedStop_Signal	Forced stop signal	Sets a signal using the each axis forced stop. ForcedStop_Signal is the structure of SIGNAL_SELECT type. The specific setting and the operation of this signal are shown below. • StartIO (IO No.) Ignores the input value. • Target (Target) When the specification is not set, signal disabled is determined and the signal undetection status is always set. When unavailable data is set, the error "Out of Parameter Range (Axis)" (error code: 1D80H) is output. • Detection (Signal detection method) The specification is allowed at the following level detection only. When edge detection specification is set, the error "Out of Parameter Range (Axis)" (error code: 1D80H) is output. 0: HighLevel (Detection at TRUE) 1: LowLevel (Detection at FALSE) • CompensationTime (Compensation time) Ignores the input value. • FilterTime (Filter time) Setting range of the filter time is 0.0 to +5.0. When a value outside the range is set, the warning "Out of Filter Time Setting Range of Each Axis Signal Warning" (warning code: 0D24H) is output and the axis operates with 0.0 of the filter time. The fetch timing of the label is at ready ON, and the detection timing of the signal is the axis operation cycle. For details on SIGNAL_SELECT type, refer to the following. ☞ Page 418 External Signal Selection
AxesGroupName.Pr.		
StopMode_ErrorInGroup	Structuring axis operation selection at axis stop by factors	During operation in the axes group, when the driver servo is turned OFF on the configuration axis and an immediate stop axis error occurs, it sets the operation of the axes other than the axis error occurred. 1: ImmediateStop (Immediate stop) (Default value) 4: ServoOffAfterImmediateStop (Servo OFF after immediate stop) 5: ServoOffAfterDecelStop (Servo OFF after deceleration to stop) (Immediate stop will be executed when executing instructions without deceleration specification.)
AxisName.Md.		
ForcedStop_Released	Forced stop cancelling	FALSE: Forced stop input TRUE (Forced stop) TRUE: Forced stop input FALSE (Forced stop release)

Variable/Structure name		Name	Details
	ForcedStop_Signal	Forced stop signal	Displays the input status of each axis forced stop signal. ForcedStop_Signal is the structure of SIGNAL_SELECT type. The specific output of this signal is shown below. • StartIO (IO No.) 0 is always displayed. • Target (Target) The fetch result of the parameter is displayed. • Detection (Signal detection method) The fetch result of the parameter is displayed. • CompensationTime (Compensation time) 0.0 is always desplayed. • FilterTime (Filter time) The fetch result of the parameter is displayed. For details on SIGNAL_SELECT type, refer to the following. ☞ Page 418 External Signal Selection
System.Pr.			
	ForcedStop_Signal	All axes forced stop signal	Sets a signal using the all axes forced stop. ForcedStop_Signal is the structure of SIGNAL_SELECT type. The specific setting and the operation of this signal are shown below. • StartIO (IO No.) 0 is always displayed. • Target (Target) If the target is unspecified, it is regarded as invalid signal and it is always the signal undetection status. When unavailable data is set, the error "Out of Parameter Range (System)" (error code: 1D82H) is output. • Detection (Signal detection method) The specification is allowed at the following level detection only. When edge detection specification is set, the error "Out of Parameter Range (System)" (error code: 1D82H) is output. 0: HighLevel (Detection at TRUE) 1: LowLevel (Detection at FALSE) • CompensationTime (Compensation time) Ignores the input value. • FilterTime (Filter time) Setting range of the filter time is 0.0 to +5.0. When a value outside the range is set, the warning "Out of Filter Time Setting Range of System Signal Warning" (warning code: 0F0FH) is output and the axis operates with 0.0 of the filter time. The fetch timing of the label is at ready ON, and the detection timing of the signal is the axis operation cycle. For details on SIGNAL_SELECT type, refer to the following. ☞ Page 418 External Signal Selection
System.Md.			
	ForcedStop_Released	Forced stop cancelling	FALSE: All axes forced stop input TRUE (Forced stop) TRUE: All axes forced stop input FALSE (Forced stop release)
	ForcedStop_Signal	All axes forced stop signal	Displays the input status of all axes forced stop signal. ForcedStop_Signal is the structure of SIGNAL_SELECT type. The specific output of this signal is shown below. • StartIO (IO No.) 0 is always displayed. • Target (Target) The fetch result of the parameter is displayed. • Detection (Signal detection method) The fetch result of the parameter is displayed. • CompensationTime (Compensation time) 0.0 is always displayed. • FilterTime (Filter time) The fetch result of the parameter is displayed. For details on SIGNAL_SELECT type, refer to the following. ☞ Page 418 External Signal Selection

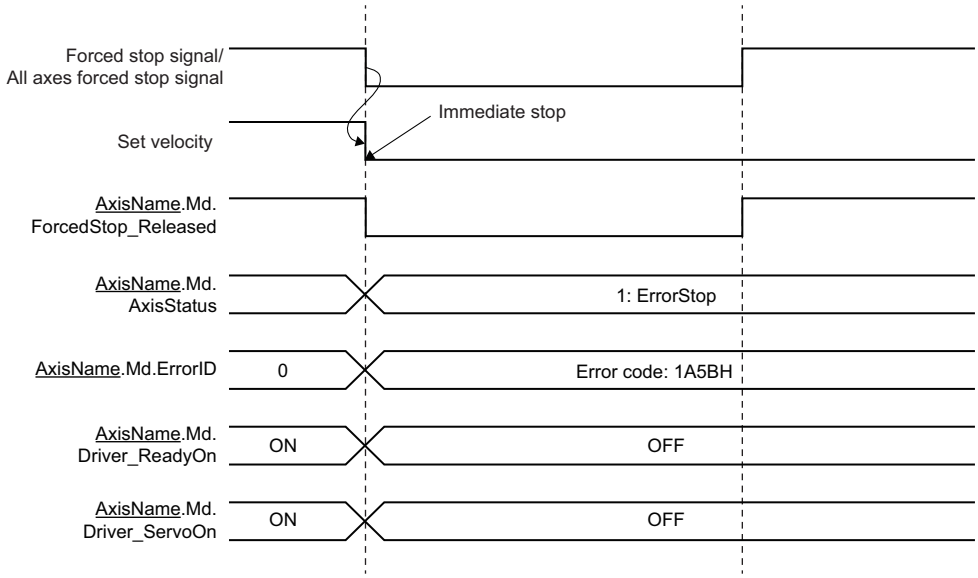
Operation at the forced stop occurrence

The following shows the operation when a forced stop has occurred.

Ex.

The slave device which servo OFF is executed when "Quick Stop" is issued is connected.

During axis operation



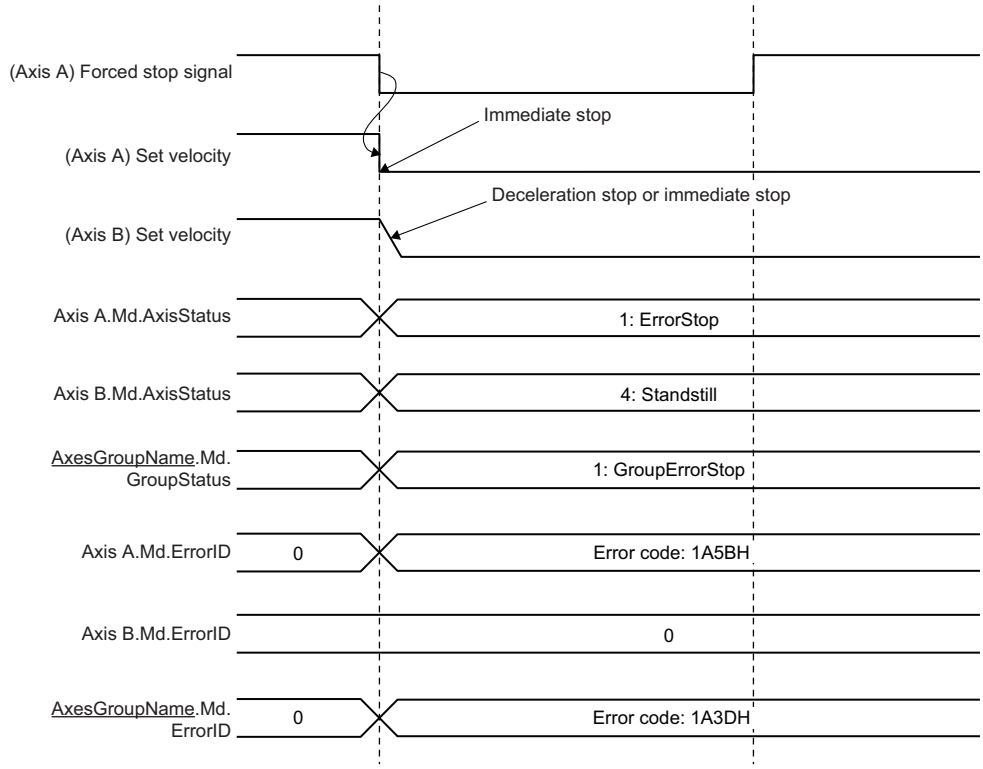
- In the axis that a forced stop occurred, MC_Power is in the error status and the servo OFF is executed. The servo ON will not be executed until the forced stop is released. The error is not output when the axis is not operated.
- In positioning or stopping, the event, "Forced stop detection" is registered in the event history.
  The errar "Forced Stop Status" (error code: 1A5BH) is also registered in positioning.
- If the axis was in the forced stop status at start, the command will not be issued, and it will be in the same "1: ErrorStop" status as the forced stop that is in operation.

During stop

If the axis the forced stop occurred in was during stop, the error "Forced Stop Status" (error code: 1A5BH) will not occur. The axis status (AxisName.Md.AxisStatus) transits according to the driver device operation that received the forced stop signal of the controller.

During axes group operation

The following shows the operation when a forced stop has occurred to the axis A while an axes group configured with the axis A and B is being operated.



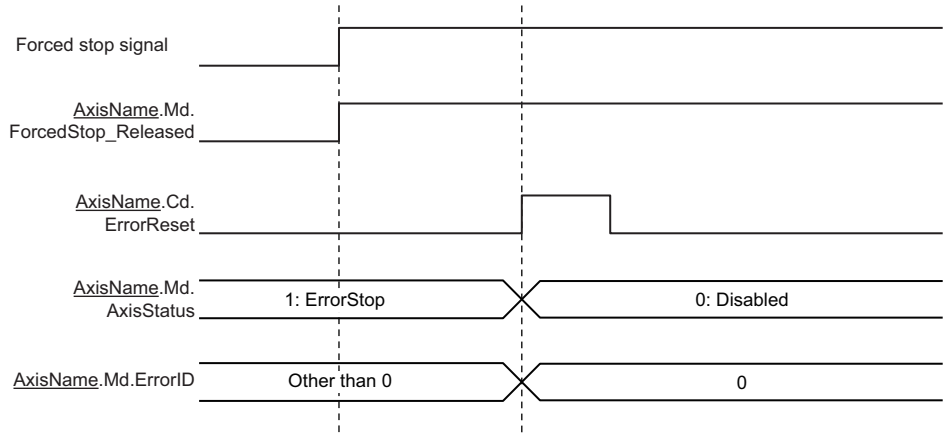
The stop method of the axis B can be set in AxesGroupName.Pr.StopMode_ForcedStopInGroup. For the operation of each setting, refer to the following.

☞ Page 184 Stop cause occurrence in the configuration axis

Operation at the forced stop release

The operation the forced stop is released from the forced stop occurrence status is shown below.

Single axis

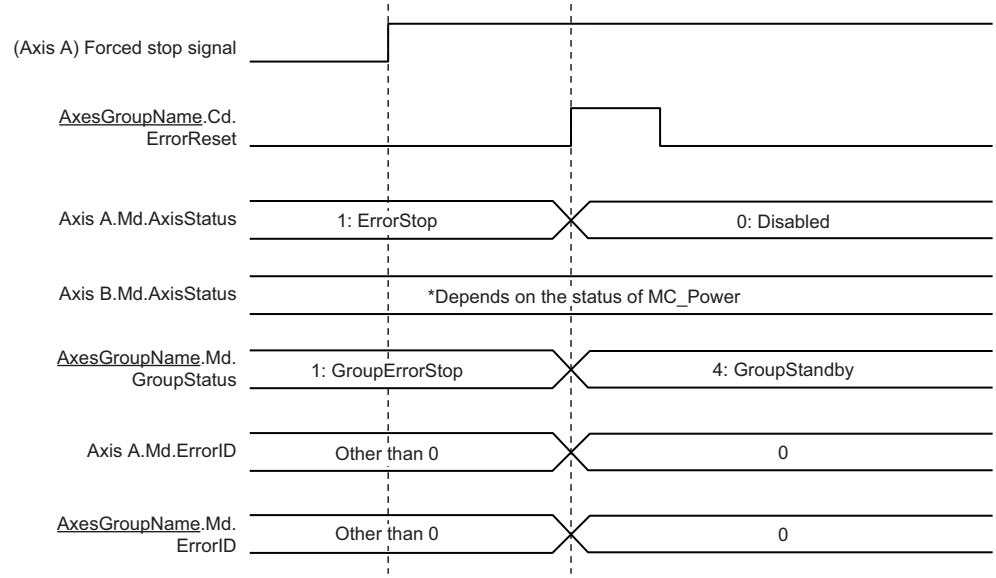


1. Release the forced stop.
The forced stop input changes to "1".
The axis status (AxisName.Md.AxisStatus) is still "1: ErrorStop".

2. Execute the axis error reset.
The axis status (AxisName.Md.AxisStatus) changes to "0: Disabled".

Axes group

The following shows the release operation when a forced stop has occurred to the axis A while an axes group configured with the axis A and B is being operated.



1. Release the forced stop.
The forced stop input changes to "1".
The axes group status (AxesGroupName.Md.GroupStatus) is still "1: GroupErrorStop".

2. Execute the axes group error reset.
The axes group status (AxesGroupName.Md.GroupStatus) changes to "4: GroupStandby", and the error Nos. are cleared in both the axis A and the axes group.

Point

When the forced stop is issued, the servo ON command of MC_Power continues.

Check method

Whether the forced stop input is in the TRUE/FALSE status can be checked with AxisName.Md.ForcedStop_Released.

5 HOMING

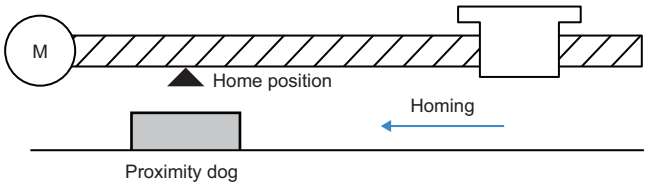
In "homing", a position is established as the starting (or "home position") when carrying out positioning control, and positioning is carried out toward that start point.
It is used to return a machine system at any position other than the home position to the home position when the motion system issues a "homing request" with the power turned ON or others, or after a positioning stop.

Precautions

When using an absolute position system, execute a homig always at the following cases: on starting up and when the controller or absolute position motor has been replaced. Check the homing request signal using the program, etc. before performing the positioning control. Failure to observe this could lead to an accident such as a collision.
The address information stored in the motion system cannot be guaranteed while the AxisName.Md.Homing_Request is TRUE.
AxisName.Md.Homing_Request is FALSE and AxisName.Md.Homing_Complete is TRUE if the homing is executed and is completed normally.

5.1 Overview

In homing, a machine home position is established.
None of the address information stored in the motion system or driver is used at this time.
The position mechanically established after the homing is regarded as the "home position" to be the start point for positioning control.



Homing method at homing starts is "driver homing method" when all of the following conditions are satisfied, or "data set homing method" when the following conditions are not satisfied.

- Axis type is real drive axis
- Driver supports Homing mode
- "HomeOffset (607CH)" is set to a slave object.

Operation of this function for each system status

○: Possible, ×: Not possible

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

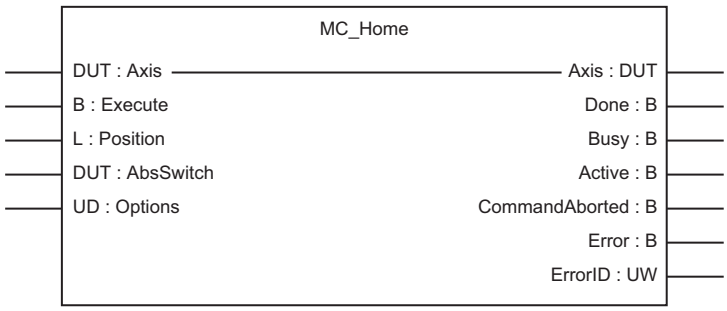
Relevant variables/FB

Variable/Structure name		Name	Details
AxisName.Md			
	Homing_Status	Homing operation status	Stores the homing status of the driver.*1
	Homing_Request	Homing request	Turns TRUE when homing is required and turns FALSE when it is completed. For details, refer to the following. ☞ Page 201 Homing Request
	Homing_Complete	Homing completed	Turns TRUE when homing is normally completed, and it turns FALSE when operation starts.

*1 The following value of homing operation status is stored.

Stored value	Status	Statusword		
		Bit 13	Bit 12	Bit 10
FFFFH	Homing procedure is not in progress	—		
0000H	Homing procedure is in progress	0	0	0
0001H	Homing procedure is interrupted or not started	0	0	1
0002H	Homing is attained, but target is not reached	0	1	0
0003H	Homing procedure is completed successfully	0	1	1
0004H	Homing error occurred, velocity is not 0	1	0	0
0005H	Homing error occurred, velocity is 0	1	0	1

MC_Home

Item	Description
Function overview	Executes homing of the specified axis.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

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

■I/O variable

Input import ↑: Only at start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Axis information	Axis	AXIS_ REF	↑	—	—	☞ Page 702 Axis variables

■Input variables

Import ↑: At start

Name	Variable name	Data type	Import	Setting range	Default value	Description
Execute command	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Target position	Position	LREAL	↑	-10000000000.0 ≤ Setting value < 10000000000.0*1	0.0	Sets the home position address.
Home position switch	AbsSwitch	MC_INPUT_REF	↑	—	—	Sets the proximity dog signal that is transmitted to the slave device in the driver homing method. ☞ Page 418 External Signal Selection Only "HighLevel" or "LowLevel" can be selected for Detection (Signal detection method). Only "0" can be specified for CompensationTime (Compensation time [s]). "0.0 to 5.0" can be specified for FilterTime (Filter time [s]). If the setting value is not within the valid range, an error will occur.
Options	Options	DWORD(HEX)	↑	00000000H	00000000H	Set "00000000H". (If a value other than "00000000H" is set, the error "Out of Options Range" (error code: 1A4EH) will occur.)

*1 When ring counter is enabled, the range is ring counter range.

■Output variables

Name	Variable name	Data type	Default value	Description
Execution completion	Done	BOOL	FALSE	Indicates that the homing is completed.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Controlling	Active	BOOL	FALSE	Indicates that the FB is controlling the axis.
Abortion of execution	CommandAborted	BOOL	FALSE	Indicates that another FB has aborted the execution of the FB.
Error	Error	BOOL	FALSE	When TRUE, it indicates that an error has occurred in the FB.
Error code	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned. For details, refer to the following. ☞ Page 662 List of Error Codes

Homing Request

AxisName.Md.Homing_Request must be TRUE in the motion system, and homing must be executed in the following cases.

Point

The reason of the homing request is TRUE is recorded in event history.

For causes of which AxisName.Md.Homing_Request turns TRUE, refer to the following.

☞ Page 102 Homing request

AxisName.Md.Homing_Request turns FALSE by the completion of the homing.

When homing is not required

Execute a homing request clear in systems that do not require homing. AxisName.Md.Homing_Request can be turned FALSE by doing so. For details, refer to the following.

☞ Page 205 Operation Setting for Incompletion of Homing

Precautions

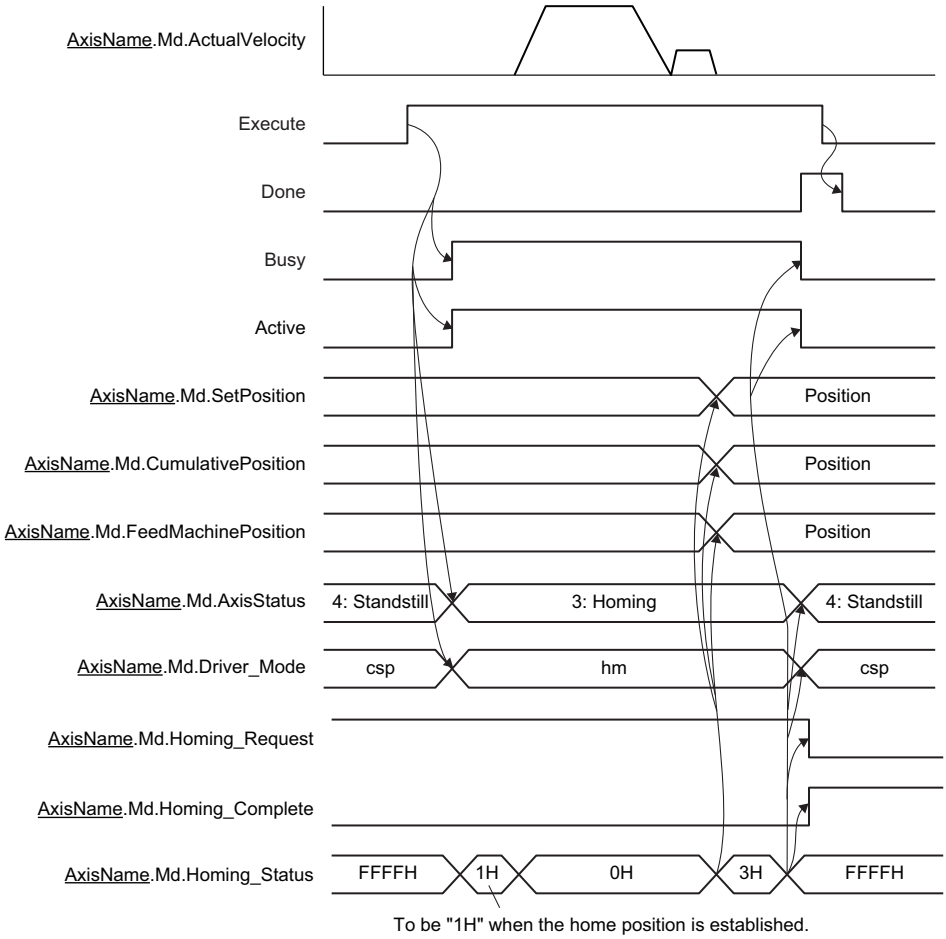
If AxisName.Md.Homing_Request is directly rewritten to FALSE in the program, etc., the homing request in the motion system is not cleared. Make sure that the homing request turns FALSE by using the homing request clear.

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 to change the homing method or each parameter. Refer to the manual of the driver because the homing operation and parameters depend on the specification of the driver.

Timing chart

■Normal operation



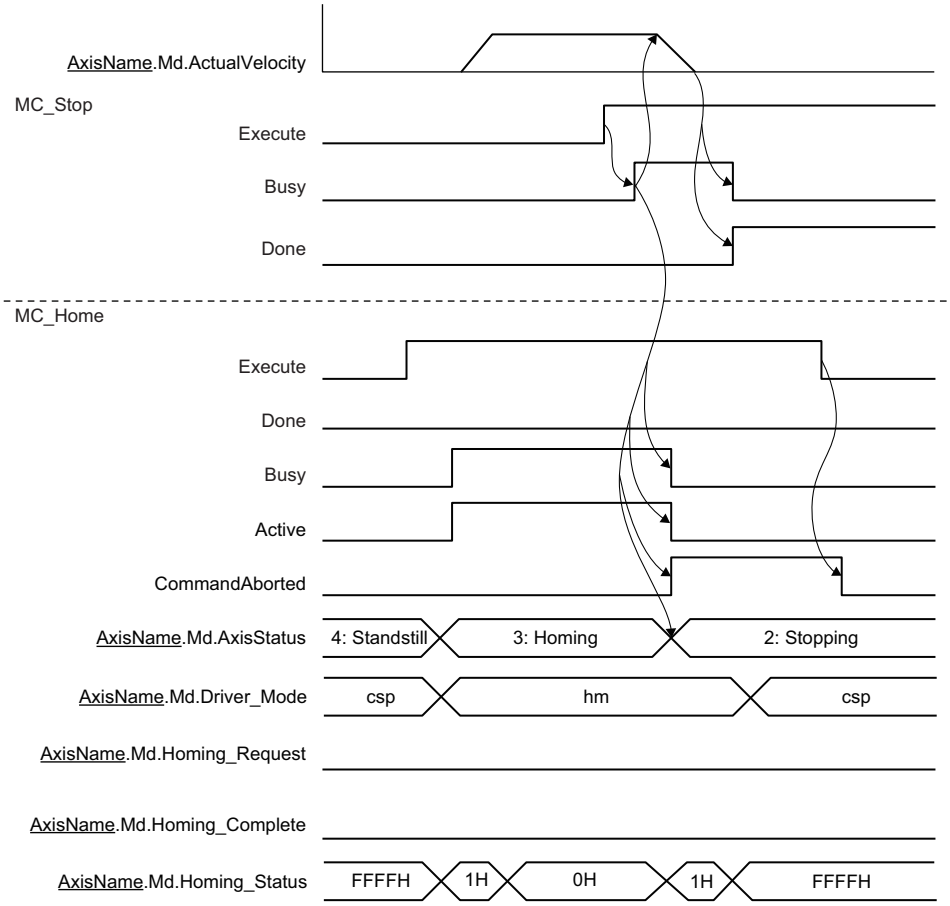
■When an error occurs

The data set method homing is the same.

For details when an error occurs, refer to the following.

☞ Page 27 Execute type/Enable type

■When MC_Stop (forced stop) is executed



- When MC_Stop is turned TRUE at homing, the "HALT" signal is sent to the driver. When using a driver that does not support HALT, use the forced stop because it does not stop with this signal.
- Stop processing at homing depends on the driver specifications. Therefore, Deceleration and Jerk of MC_Stop are ignored.

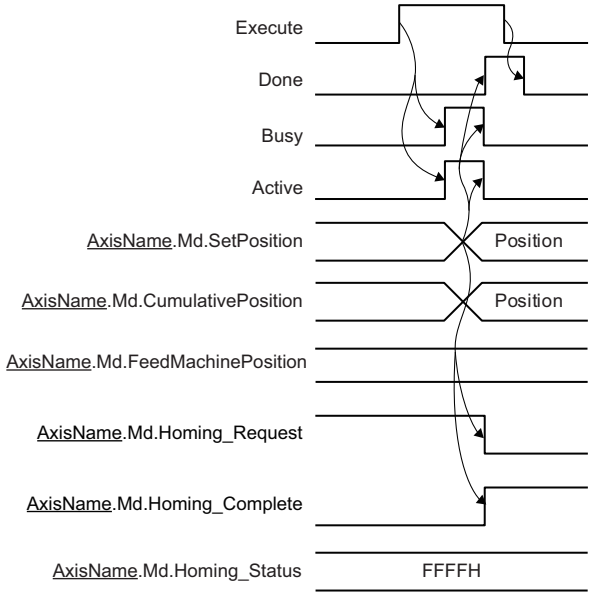
Precautions

- The homing cannot be started during servo-off. Thus, the driver homing method, Homing 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 the AbsSwitch input) 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.
- The set position is not followed up during a driver homing method.
- Do not execute the synchronous control by setting the axis that is in homing as the master axis. Because the set value and the actual value are not updated consecutively, the error "Speed Range Over during Controlling" (error code: 1AE8H) may occur at the slave axis.
- If the following items are not mapped, backup is not guaranteed.
  Home cycle counter (2D3DH)
  Home ABS counter (2D3EH)
- DOG signal inputted in the servo amplifier at MR-J5(W)-G connection can be specified as a home position switch signal. For details on the setting method, refer to the following.
  ☞ Page 733 Using methods
- The hardware stroke limit signal detection is performed during the homing in the Motion module. When the signal is detected, "HALT" signal is sent to the driver. To stop by using the limit switch signal of the driver side, make the hardware stroke limit check of the Motion module side invalid temporarily by turning AxisName.Cd.HwStrokeLimit_Override to DISABLE.
- If ONLY_INSIDE is set in AxisName.Cd.HwStrokeLimit_Override at homing start, the error "Start Not Possible" (error code: 1AADH) occurs and the homing start is not carried out.

Data set homing method

"Data set homing method" is executed to a virtual axis and a real axis that does not have the home position information in the slave device side. It is completed in the motion system, and external signals, etc. are not used. The Position (home position address) at homing is registered in the motion system as the home position, and AxisName.Md.SetPosition and AxisName.Md.CumulativePosition are rewritten to Position (home position address).

Timing chart



Precautions

- FBs other than MC_Stop cannot start during homing.
- The software stroke limit check for the Position (home position address) at homing start is shown below.
  Driver homing method: The software stroke limit check is not carried out.
  Data set homing method: The software stroke limit check is carried out.
- Start position of the software stroke limit check is not carried out at homing start.
- The software stroke limit is not checked during homing.
- The movement direction at homing completion is in positive direction.

Error

Check that the axis status (AxisName.Md.AxisStatus) is "4: Standstill" at MC_Home execution. If it is other than "4: Standstill", an error will occur and the control during operation will stop.

5.2 Operation Setting for Incompletion of Homing

The operation setting for incompletion of homing is provided to select whether the axis is started or not when the homing request, AxisName.Md.Homing_Request is TRUE.
When starting the axis at homing incompletion, whether the axis can startup or not is determined by the executed FB and AxisName.Pr.StartableAtUnhomed.
○: Axis start possible, ×: Axis start impossible

FB	Start permission for incompletion of homing	
	"TRUE: Allowed" and the homing request is TRUE	"FALSE: Not allowed" and the homing request is TRUE
FBs that can be started when homing is incompletion	○*1	○*1
The motion FBs other than the above	○*1	×

*1 There may be restrictions in the operation for incompletion of homing depending on the setting or specifications of the drive unit. For details, refer to each drive unit manual.

Relevant variables

Variable/Structure name	Name	Details
AxisName.Pr.		
StartableAtUnhomed	Start permission at homing uncompleted	Sets whether axis start when homing is incompletion is allowed or not. TRUE: Allowed FALSE: Not allowed
AxisName.Md.		
Homing_Request	Homing request	Turns TRUE when homing is required and turns FALSE when it is completed. For details, refer to the following. ☞ Page 201 Homing Request
AxisName.Cd.		
Homing_ClearRequest	Homing request clear	Turns the homing request to FALSE forcibly. After turning FALSE the homing request, FALSE is set in this variable automatically.

FBs that can be started when homing is incompletion

The following FBs can start the axis regardless of the AxisName.Pr.StartableAtUnhomed and AxisName.Md.Homing_Request status.

FBs that can be started when homing is incompletion
MC_Home (Homing)
MCv_Jog (JOG operation)
MC_Stop (Forced stop)
MC_GroupStop (Group forced stop)
FBs other than the motion FBs

The axis start other than the above cannot start the axis when AxisName.Pr.StartableAtUnhomed is FALSE and AxisName.Md.Homing_Request is TRUE, and the error "Start during Homing Incomplete" (error code: 1A22H) occurs.

Homing request clear

When AxisName.Md.Homing_Request is TRUE, AxisName.Md.Homing_Request can be turned to FALSE forcibly by turning AxisName.Cd.Homing_ClearRequest to TRUE.

Precautions

- In AxisName.Cd.Homing_ClearRequest, when turning AxisName.Md.Homing_Request FALSE and starting the axis, it is operated not based on the home position but based on the current coordinate system.
- Do not turn AxisName.Cd.Homing_ClearRequest TRUE during homing.

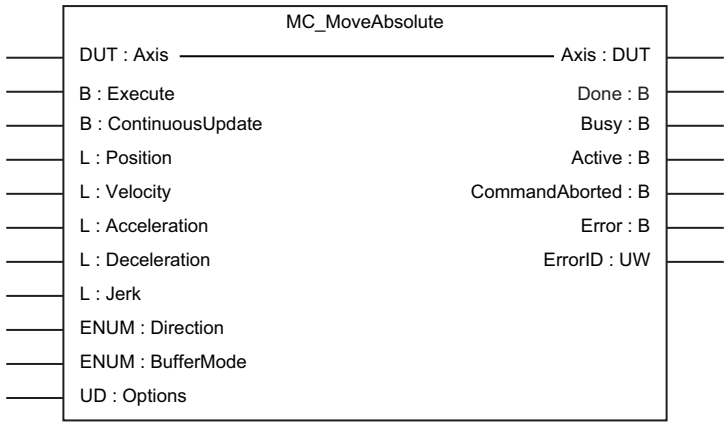
6 AXIS CONTROL FUNCTION

6.1 Single Axis Positioning Control

The details and usage of the single axis positioning controls are explained in this section.
The single axis positioning controls execute positioning to the specified position by using address information.

Relevant FBs

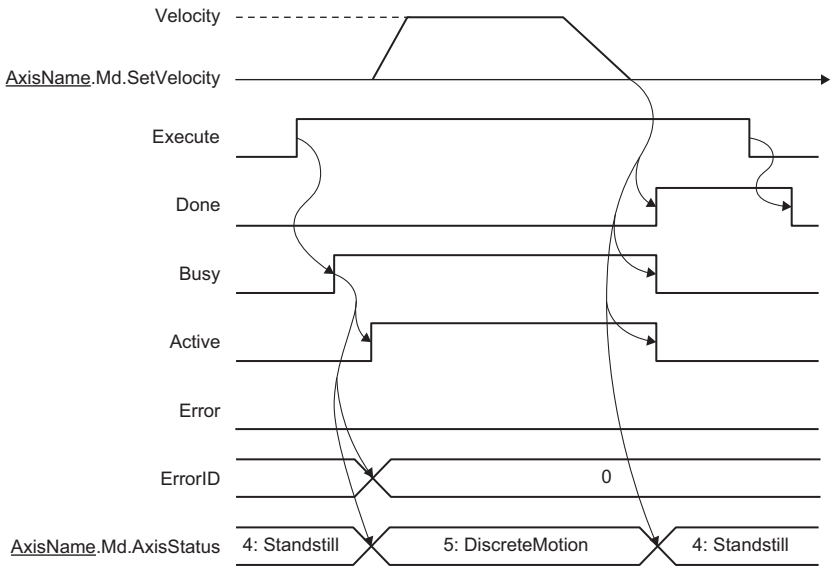
MC_MoveAbsolute

Item	Description
Function overview	Executes positioning after specifying the target position of the absolute position.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

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

■Timing chart

[When the FB is normally completed]



[When an error occurs]

For details when an error occurs, refer to the following.

☞ Page 27 Execute type/Enable type

■I/O variable

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Axis information	Axis	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables

■Input variables

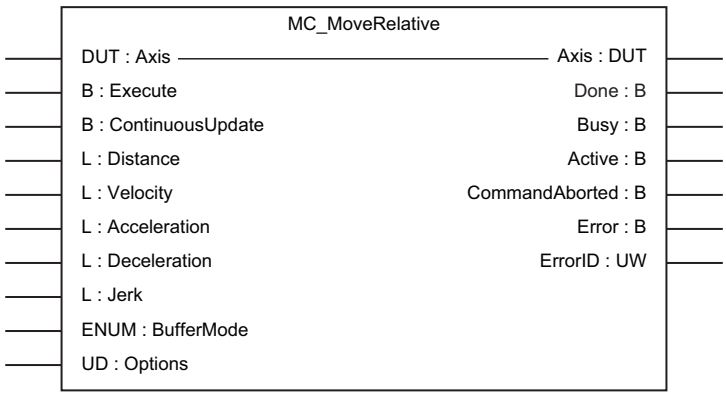
Import ↑: At start, R: Retrigger possible, C: Continuous update possible

Name	Variable name	Data type	Import	Setting range	Default value	Description
Execute command	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Continuous update	ContinuousUpdate	BOOL	↑	TRUE, FALSE	FALSE	During TRUE, it can continuously change the target position, the speed, the acceleration, and the deceleration.
Target position	Position	LREAL	↑/R/C	☞ Page 210 Position (Target position)	0.0	Sets the target position of the absolute position according to unit of axis.
Velocity	Velocity	LREAL	↑/R/C	☞ Page 87 Velocity set range	0.0	Sets the velocity according to unit of axis.
Acceleration	Acceleration	LREAL	↑/R/C	☞ Page 335 Input variables to use	0.0	Sets the acceleration according to unit of axis.
Deceleration	Deceleration	LREAL	↑/R/C	☞ Page 335 Input variables to use	0.0	Sets the deceleration according to unit of axis.
Jerk	Jerk	LREAL	↑	☞ Page 335 Input variables to use	0.0	Sets the jerk according to unit of axis.
Direction selection	Direction	MC_DIRECTION	↑	1 to 3	0	Sets the direction to move from the current position to the target position when the software stroke limit is invalid. 1: mcPositiveDirection (Positive direction) 2: mcNegativeDirection (Negative direction) 3: mcShortestWay (Shortest path)
Buffer mode	BufferMode	MC_BUFFER_MODE	↑	0 to 5	0	Selects the buffer mode. 0: mcAborting (Aborting) 1: mcBuffered (Buffered) 2: mcBlendingLow (BlendingLow) 3: mcBlendingPrevious (BlendingPrevious) 4: mcBlendingNext (BlendingNext) 5: mcBlendingHigh (BlendingHigh)
Options	Options	DWORD(HEX)	↑	☞ Page 217 Options (Options)	00000000H	Sets a function option.

■Output variables

Name	Variable name	Data type	Default value	Description
Execution completion	Done	BOOL	FALSE	During TRUE, it indicates that the output value is valid.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Controlling	Active	BOOL	FALSE	Indicates that the axis is being controlled.
Abortion of execution	CommandAborted	BOOL	FALSE	It indicates that another FB has aborted the execution of the FB. Turns TRUE when this FB is stopped by an error or this FB is started during an error occurence. Turns FALSE when Execute = FALSE.
Error	Error	BOOL	FALSE	When TRUE, it indicates that an error has occurred in the FB.
Error code	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned. For details, refer to the following. ☞ Page 662 List of Error Codes

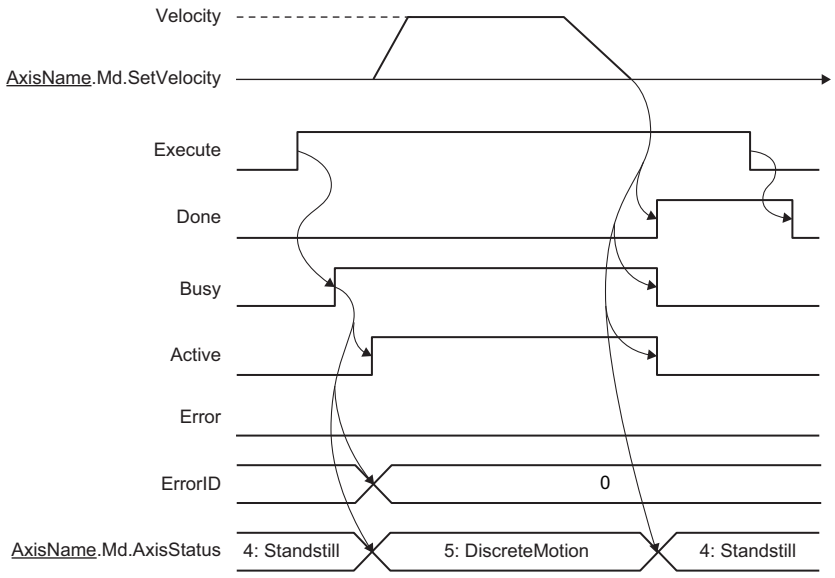
MC_MoveRelative

Item	Description
Function overview	Executes positioning after specifying the movement amount of the relative position.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

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

■Timing chart

[When the FB is normally completed]



[When an error occurs]

For details when an error occurs, refer to the following.

☞ Page 27 Execute type/Enable type

■I/O variable

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Axis information	Axis	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables

■Input variables

Import ↑: At start, R: Retrigger possible, C: Continuous update possible

Name	Variable name	Data type	Import	Setting range	Default value	Description
Execute command	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Continuous update	ContinuousUpdate	BOOL	↑	TRUE, FALSE	FALSE	During TRUE, it can continuously change the movement amount, the speed, the acceleration, and the deceleration.
Movement amount	Distance	LREAL	↑/R/C	☞ Page 85 Combination with other functions	0.0	Sets the relative position according to unit of axis from the current position at start to the end point.
Velocity	Velocity	LREAL	↑/R/C	☞ Page 87 Velocity set range	0.0	Sets the velocity according to unit of axis.
Acceleration	Acceleration	LREAL	↑/R/C	☞ Page 335 Input variables to use	0.0	Sets the acceleration according to unit of axis.
Deceleration	Deceleration	LREAL	↑/R/C	☞ Page 335 Input variables to use	0.0	Sets the deceleration according to unit of axis.
Jerk	Jerk	LREAL	↑	☞ Page 335 Input variables to use	0.0	Sets the jerk according to unit of axis.
Buffer mode	BufferMode	MC_BUFFER_MODE	↑	0 to 5	0	Selects the buffer mode. 0: mcAborting (Aborting) 1: mcBuffered (Buffered) 2: mcBlendingLow (BlendingLow) 3: mcBlendingPrevious (BlendingPrevious) 4: mcBlendingNext (BlendingNext) 5: mcBlendingHigh (BlendingHigh)
Options	Options	DWORD(HEX)	↑	☞ Page 219 Options (Options)	00000000H	Sets a function option.

■Output variables

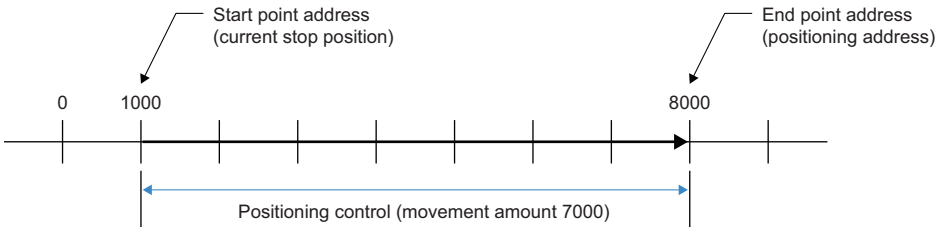
Name	Variable name	Data type	Default value	Description
Execution completion	Done	BOOL	FALSE	When TRUE, it indicates that the output value is valid.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Controlling	Active	BOOL	FALSE	Indicates that the axis is being controlled.
Abortion of execution	CommandAborted	BOOL	FALSE	It indicates that another FB has aborted the execution of the FB. Turns TRUE when this FB is stopped by an error or this FB is started during an error occurence. Turns FALSE when Execute = FALSE.
Error	Error	BOOL	FALSE	When TRUE, it indicates that an error has occurred in the FB.
Error code	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned. For details, refer to the following. ☞ Page 662 List of Error Codes

Absolute Positioning Control

In MC_MoveAbsolute, Position, Velocity, Acceleration, Deceleration, Jerk, Direction, BufferMode, and Options can be set, and positioning is executed from the current position at start (start point address) to the address set in Position (end point address).

Ex.

When the start point address (current stop position) is 1000, and Position is 8000, positioning is carried out in the positive direction for a movement amount of 7000 (8000 - 1000).



Details of setting items

■Position (Target position)

Set the target position of the absolute position. The setting range differs depending on the software stroke limit valid/invalid, Direction, and the setting of the target position specification (Options bit16) exceeding the ring counter.

• When the software stroke limit is valid (AxisName.Md.SwStrokeLimit_Target is other than "-1: Invalid")

Setting range: Ring counter lower limit value ≤ Target position < Ring counter upper limit value

The settable range is "Ring counter lower limit value ≤ Target position ≤ Ring counter upper limit value" regardless of the setting of Direction or the target position specification (Options bit16) exceeding the ring counter. If a value outside the range is set, the error "Out of Target Position Range" (error code: 1A05H) will occur, and the axis will not start.

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

• When the software stroke limit is invalid (AxisName.Md.SwStrokeLimit_Target is "-1: Invalid")

Direction	Options bit 16	Setting range of Position
mcPositiveDirection	0	Ring counter lower limit value ≤ Target position < Ring counter upper limit value
	1	Ring counter lower limit value ≤ Target position < Positioning range upper limit value*1
mcNegativeDirection	0	Ring counter lower limit value ≤ Target position < Ring counter upper limit value
	1	Positioning range lower limit value*1 ≤ Target position < Ring counter upper limit value
mcShortestWay	0	Ring counter lower limit value ≤ Target position < Ring counter upper limit value
	1	

*1 For the positioning range lower/upper limit value, refer to the following.
☞ Page 83 Positioning Range

To set the target position specification (Options bit 16) exceeding ring counter value to "1: Allow" can position in one rotation or more.

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

For details on the direction moving toward the target position, refer to the following.

☞ Page 211 Direction (Direction selection)

■Velocity (Velocity)

Set the set velocity at positioning control. For details, refer to the following.

☞ Page 87 Velocity set range

■Acceleration (Acceleration)

Specify the acceleration. The setting details defer depending on the acceleration/deceleration method. For details, refer to the following.

☞ Page 253 Multiple Axes Positioning Control

■Deceleration (Deceleration)

Specify the deceleration. The setting details defer depending on the acceleration/deceleration method.For details, refer to the following.

☞ Page 253 Multiple Axes Positioning Control

■Jerk (Jerk)

Set Jerk. For details, refer to the following.

☞ Page 330 Acceleration/deceleration Processing Function

■Direction (Direction selection)

This setting is ignored when the software stroke limit is valid. Positioning control in a direction which does not cross over the area outside the software stroke limit range is carried out. However, when the value does not cross over the area outside the software stroke limit range both in the positive direction and the negative direction, positioning control is carried out in the direction close to the target position (the direction that the absolute value of the movement amount is shorter) based on the current position. If the distance is the same between the positive direction and the negative direction, it is carried out in the current direction.

When the software stroke limit is invalid, the positive direction, the negative direction, and the shortest path can be selected, and specify the direction to move from the current position to the target position.

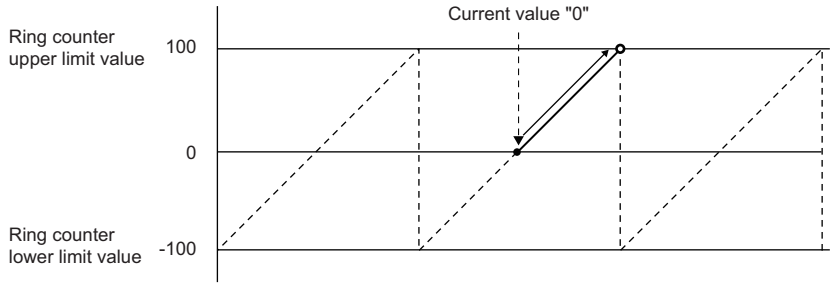
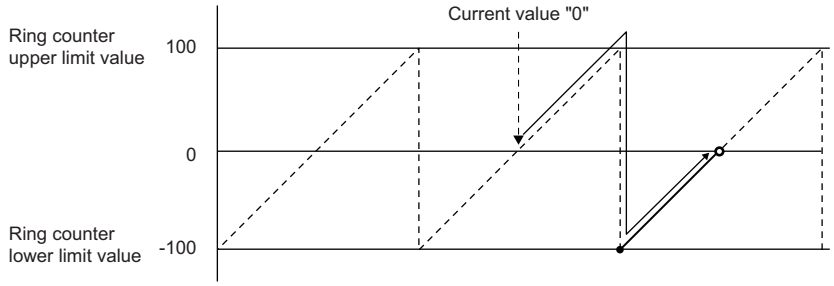
• Positive direction

1: mcPositiveDirection

The positive direction (address increase) positioning is executed from the current position to the target position.

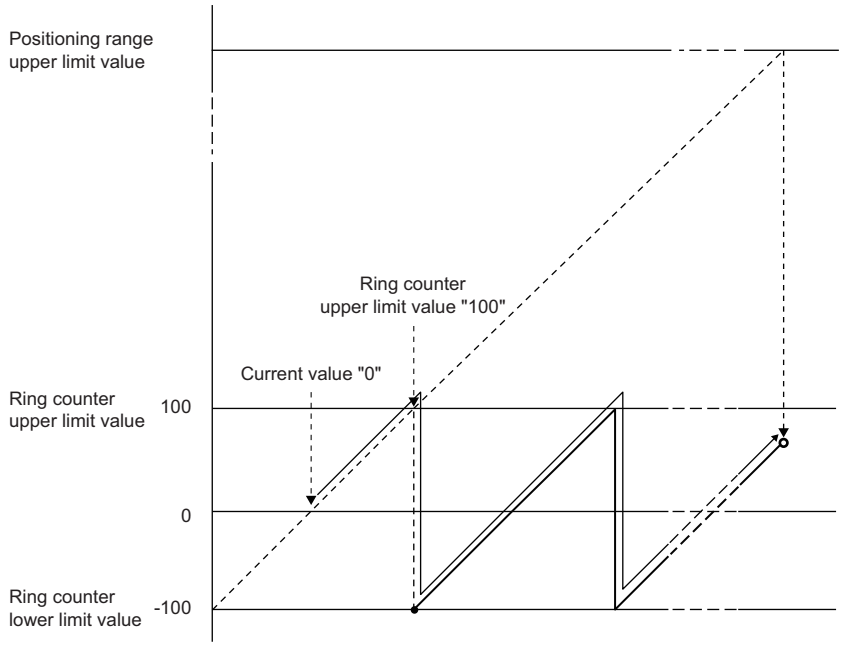
[Setting the target position exceeding the ring counter (Options bit 16)] "0: Not allow"

The setting range of the target position is "Ring counter lower limit value ≤ Target position ≤ Ring counter upper limit value."

Setting	Operation
Current value ≤ Target position < Ring counter upper limit value	
Ring counter lower limit value ≤ Target position < Current value	

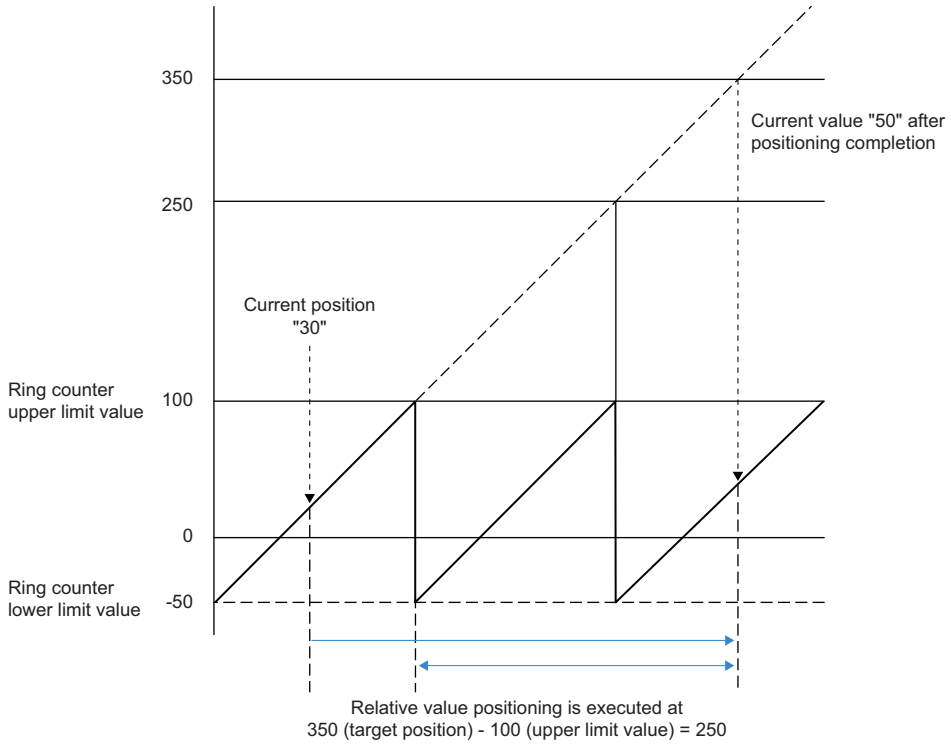
[Setting the target position exceeding the ring counter (Options bit 16)] "1: Allow"

The positioning operation that overflows the ring counter upper limit value can be performed. In this case, the ring counter upper limit value is used as a reference, and positioning is performed with the amount of movement exceeded as the relative movement amount. If the target position is set with the value smaller than the ring counter lower limit value, positioning does not start for the error "Out of Target Position Range" (error code: 1A05H).

Setting	Operation
• Current value ≤ Target position < Ring counter upper limit value • Ring counter lower limit value ≤ Target position < Current value	The same operation as the "Setting the Target position exceeding the ring counter: Not allow"
Ring counter upper limit value ≤ Target position < Positioning range upper limit value	

Ex.

Operation when setting the ring counter upper limit value to "100", the ring counter lower limit value to "-50", the current position to "30", and the target position to "350" (the relative value positioning is executed after exceeding the ring counter upper limit value)



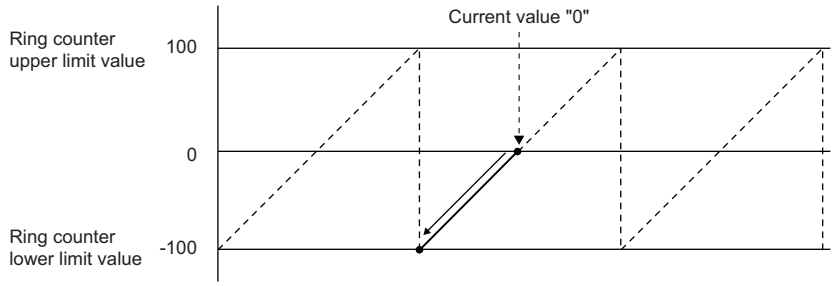
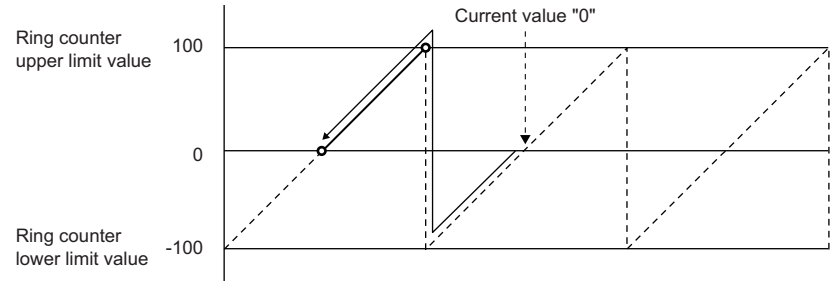
• Negative direction

2. mcNegativeDirection

The positioning is executed from the current position to the target position of negative direction (address decrease).

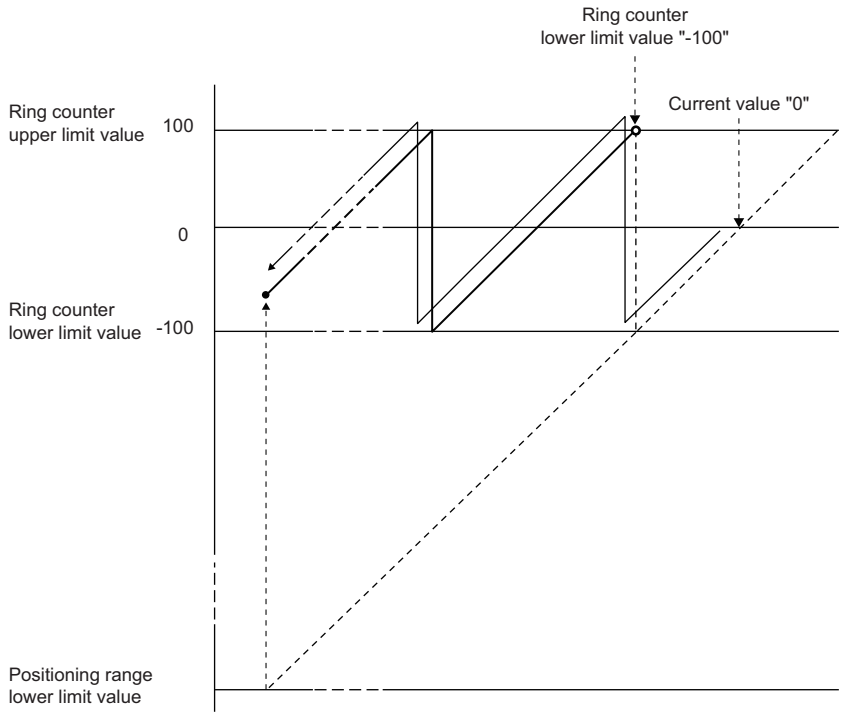
[Setting the target position exceeding the ring counter (Options bit 16)] "0: Not allow"

The setting range of the target position is "Ring counter lower limit value ≤ Target position < Ring counter upper limit value".

Setting	Operation
Ring counter lower limit value ≤ Target position ≤ Current value	
Current value < Target position < Ring counter upper limit value	

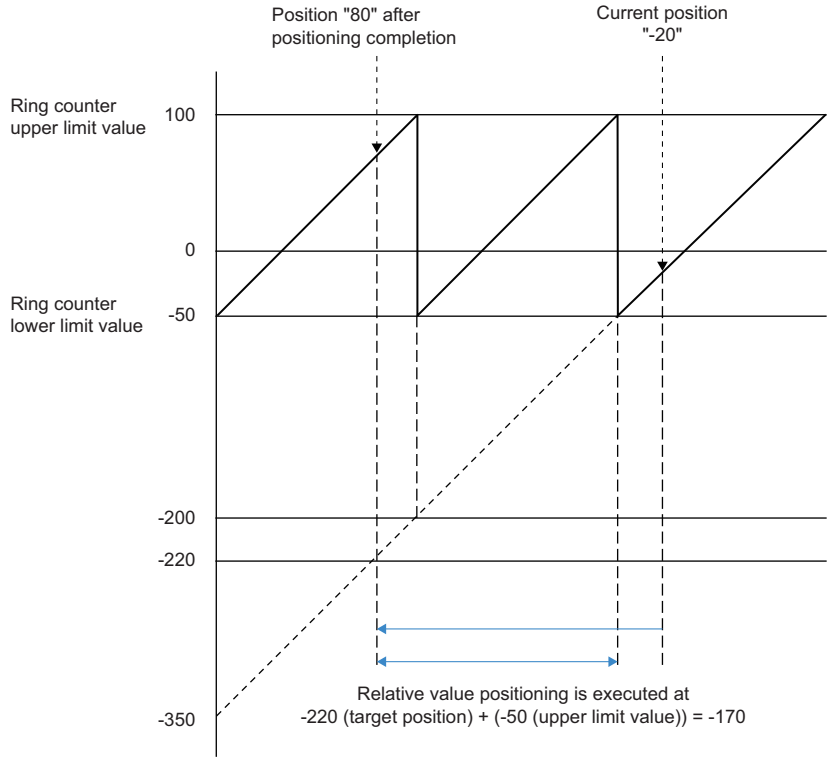
[Setting the target position exceeding the ring counter (Options bit 16)] "1: Allow"

The positioning operation that underflows the ring counter lower limit value can be performed. In this case, the ring counter lower limit value is used as a reference, and positioning is performed with the amount of movement exceeded as the relative movement amount. If the target position is set with the value bigger than the ring counter upper limit value, positioning does not start for the error "Out of Target Position Range" (error code: 1A05H).

Setting	Operation
• Ring counter lower limit value ≤ Target position ≤ Current value • Current value < Target position < Ring counter upper limit value	The same operation as the "Setting the target position exceeding the ring counter: Not allow"
Positioning range lower limit value ≤ Target position < Ring counter lower limit value	

Ex.

Operation when setting the ring counter upper limit value to "100", the ring counter lower limit value to "-50", the current position to "-20", and the target position to "-220"



- Shortest path

3: mcShortestWay

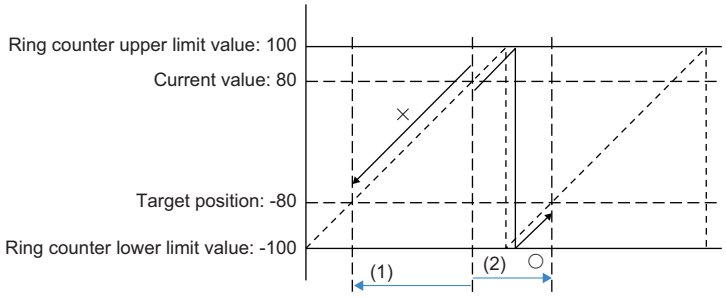
Based on the current position, positioning control is performed in the direction close to the target position (the one with the smaller absolute value of the movement amount). If the distance is the same in the positive direction and the negative direction, it is operated in the current direction.

The setting range of the target position is "Ring counter lower limit value ≤ Target position < Ring counter upper limit value" regardless of the setting of the target position specification (Options bit16) exceeding the ring counter.

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

Ex.

Operation when setting the ring counter upper limit value to "100", the lower limit value to "-100", the current position to "80", and the target position to "-80"



Since the movement amount of (1) is 160 and that of (2) is 40, positioning moves to the positive direction of (2).

■BufferMode (Buffer mode)

Select the buffer mode. Aborting, Buffered, BlendingLow, BlendingPrevious, BlendingNext, and BlendingHigh can be set. For details, refer to the following.

☞ Page 133 Buffer mode type

■Options (Options)

Set the function options that are used in MC_MoveAbsolute by specifying the bit.

The details of bit and the functions are shown below.

Bit	Function description
0 to 2	Acceleration/deceleration method setting 0: mcAccDec (Acceleration/deceleration specification method) 1: mcFixedTime (Acceleration/deceleration time-fixed method) For details, refer to the following. ☞ Page 330 Acceleration/deceleration Processing Function
3 to 4	Empty (Specify "0".)*1
5	Reverse rotation permission selection 0: Allow 1: Not allow It is valid when specifying Aborting of buffer mode or changing target position. For details, refer to the following. ☞ Page 132 Multiple Start (Buffer Mode)☞ Page 413 Target position change
6 to 15	Empty (Specify "0".)*1
16	Target position specification exceeding the ring counter Specifies whether the target position that exceeds the ring counter upper/lower limit value is allowed or not when the software stroke limit is invalid. 0: Not allow 1: Allow
17 to 31	Empty (Specify "0".)*1

*1 If specifying other than "0", the error "Out of Options Range" (error code: 1A4EH) occurs, and the function does not start.

■Done (Execution completion)

It turns TRUE when reaching the target position.

■Busy (Executing)

It is TRUE during the instruction execution. It turns FALSE after reaching the target position.

■Active (Controlling)

It is TRUE while the axis is controlled. It turns FALSE after reaching the target position.

■CommandAborted (Abortion of execution)

It turns TRUE when another instruction is started during the execution of this FB.

■Error (Error)

It turns TRUE when an error is occurring in this FB.

■ErrorID (Error code)

The error code generated in this FB is returned.

Necessary slave object

When using MC_MoveAbsolute, set the following slave object to the axis.

- Target position

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

For details on the slave object setting, refer to the following.

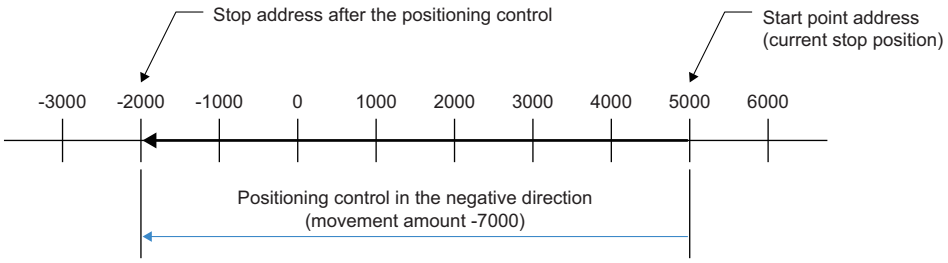
☞ Page 56 Axis Assignment

Relative Positioning Control

In MC_MoveRelative, Distance, Velocity, Acceleration, Deceleration, Jerk, and BufferMode can be set, and positioning of movement amount set in Distance is executed from the current position at start (start point address). The movement direction is determined by the sign of the movement amount. The axis status (AxisName.Md.AxisStatus) becomes "5: DiscreteMotion".

Ex.

When the start point address is 5000, and the movement amount is -7000, positioning is carried out to -2000.



Details of setting items

■Distance (Movement amount)

Set the signed movement amount.

Positive movement amount: Move to positive direction (Address increase direction)

Negative movement amount: Move to negative direction (Address decrease direction)

When movement amount is 0, the axis is not operated, however, the execution completion (Done) will be TRUE.

If "the current value + the movement amount" overflows or underflows the software stroke limit upper/lower limit value when the software stroke limit is valid, the error "Out of Target Position Range" (error code: 1A05H) occurs and the FB does not start.

For details on movement amount range, refer to the following.

☞ Page 83 Positioning Range

■Velocity (Velocity)

Set the set velocity at positioning control. For details, refer to the following.

☞ Page 87 Velocity set range

■Acceleration (Acceleration)

Specify the acceleration. The setting details defer depending on the acceleration/deceleration method. For details, refer to the following.

☞ Page 253 Multiple Axes Positioning Control

■Deceleration (Deceleration)

Specify the deceleration. The setting details defer depending on the acceleration/deceleration method. For details, refer to the following.

☞ Page 253 Multiple Axes Positioning Control

■Jerk (Jerk)

Set Jerk. For details, refer to the following.

☞ Page 330 Acceleration/deceleration Processing Function

■BufferMode (Buffer mode)

Select a buffer mode. Aborting, Buffered, BlendingLow, BlendingPrevious, BlendingNext, and BlendingHigh can be set. For details, refer to the following.

☞ Page 133 Buffer mode type

■Options (Options)

Set the function options that are used in MC_MoveRelative by specifying the bit.

The details of bit and the functions are shown below.

Bit	Function description
0 to 2	Acceleration/deceleration method setting 0: mcAccDec (Acceleration/deceleration specification method) 1: mcFixedTime (Acceleration/deceleration time-fixed method) For details, refer to the following. ☞ Page 330 Acceleration/deceleration Processing Function
3	Position selection during buffer mode 0: Set position 1: Actual current value For details, refer to the following. ☞ Page 132 Multiple Start (Buffer Mode)
4	Empty (Specify "0".)*1
5	Reverse rotation permission selection 0: Allow 1: Not allow It is valid when specifying Aborting of buffer mode or changing target position. For details, refer to the following. ☞ Page 132 Multiple Start (Buffer Mode)☞ Page 413 Target position change
6 to 31	Empty (Specify "0".)*1

*1 If specifying other than "0", the error "Out of Options Range" (error code: 1A4EH) occurs, and the function does not start.

■Done (Execution completion)

It turns TRUE when reaching the target position.

■Busy (Executing)

It is TRUE during the instruction execution. It turns FALSE after reaching the target position.

■Active (Controlling)

It is TRUE while the axis is controlled. It turns FALSE after reaching the target position.

■CommandAborted (Abortion of execution)

It turns TRUE when another instruction is started during the execution of this FB.

■Error (Error)

It turns TRUE when an error is occurring in this FB.

■Error code (ErrorID)

The error code generated in this FB is returned.

Necessary slave object

When using MC_MoveRelative, set the following slave object to the axis.

- Target position

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

For details on the slave object setting, refer to the following.

☞ Page 56 Axis Assignment

Precautions

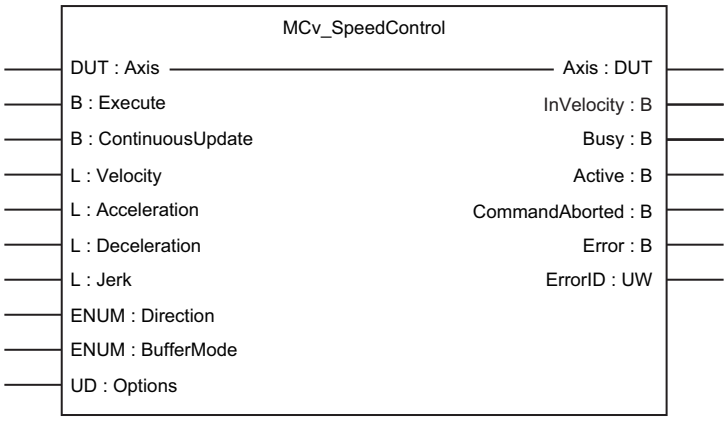
- Even though a FB is restarted or continuously updated, the current position where this FB was started for the first time is the starting address.
- Since it is processed with the floating point type, if the relative value positioning control is repeatedly executed, the movement amount specified may not be reached due to the calculation error.

6.2 Single Axis Speed Control

This section describes operation of the single axis speed control (including position loop).
For speed control excluding position loop, refer to the following.
☞ Page 276 Velocity Control

Relevant FB

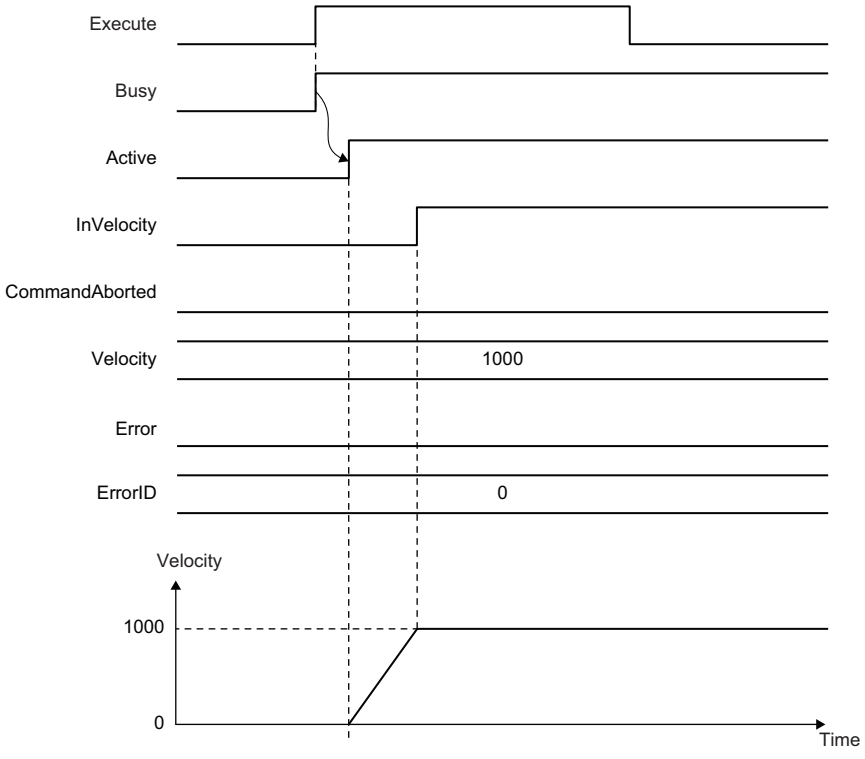
MCv_SpeedControl

Item	Description
Function overview	Executes the speed control including position loop.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
56	8	Subroutine type	Real-time execution

■Timing chart

[When the FB is normally completed]



[When an error occurs]

For details when an error occurs, refer to the following.

☞ Page 27 Execute type/Enable type

■I/O variable

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Axis information	Axis	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables

■Input variables

Import ↑: At start, R: Retrigger possible, C: Continuous update possible

Name	Variable name	Data type	Import	Setting range	Default value	Description
Execute command	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Continuous update	ContinuousUpdate	BOOL	↑	TRUE, FALSE	FALSE	Target velocity, Acceleration, and Deceleration can be continuously changed during TRUE.
Target velocity	Velocity	LREAL	↑/R/C	☞ Page 87 Velocity set range	0.0	Sets target velocity. If the speed is negative, it moves in the negative direction. If it is omitted, the axis is not operated, however, the axis status (AxisName.Md.AxisStatus) will be "6: ContinuousMotion".
Acceleration	Acceleration	LREAL	↑/R/C	☞ Page 335 Input variables to use	0.0	Sets the acceleration.
Deceleration	Deceleration	LREAL	↑/R/C	☞ Page 335 Input variables to use	0.0	Sets thedeceleration.
Jerk	Jerk	LREAL	↑	☞ Page 335 Input variables to use	0.0	Sets thejerk.
Direction selection	Direction	MC_DIRECTION	↑	1 to 2	0	Sets the direction selection. MC_DIRECTION definition can be used. 1: mcPositiveDirection (Positive direction) 2: mcNegativeDirection (Negative direction) When "2: mcNegativeDirection" is selected and Velocity is negative, the motor movement direction will be positive direction. If this setting is omitted, the error "Out of Direction Selection Range" (error code: 1A37H) will occur.
Buffer mode	BufferMode	MC_BUFFER_MODE	↑	0 to 5	0	Selects the buffer mode. 0: mcAborting (Aborting) 1: mcBuffered (Buffered) 2: mcBlendingLow (BlendingLow) 3: mcBlendingPrevious (BlendingPrevious) 4: mcBlendingNext (BlendingNext) 5: mcBlendingHigh (BlendingHigh) For details, refer to the following. ☞ Page 223 BufferMode
Options	Options	DWORD(HEX)	↑	*1	00000000H	Sets function options with bit designation.

*1 For bits and function description, refer to the following table.

Bit	Function description
0 to 2	Acceleration/deceleration method setting 0: mcAccDec Acceleration/deceleration specification method 1: mcFixedTime Acceleration/deceleration time-fixed method For details, refer to the following. ☞ Page 330 Acceleration/deceleration Processing Function
3 to 31	Empty (Specify "0".)*1

*1 If specifying other than "0", the error "Out of Options Range" (error code: 1A4EH) occurs, and the function does not start.

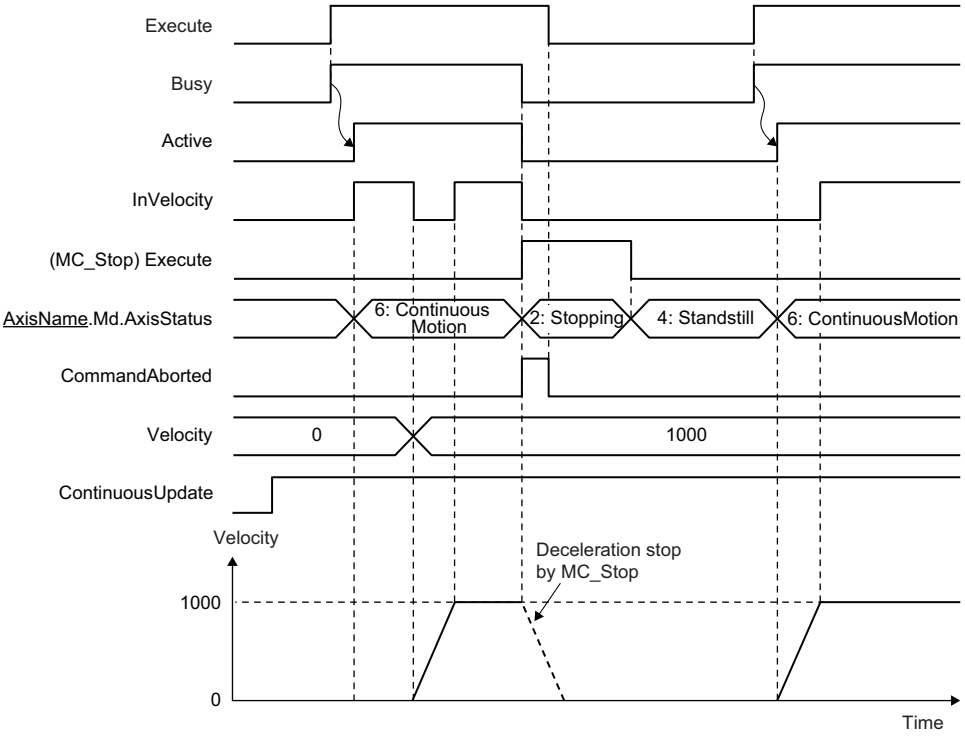
■Output variables

Name	Variable name	Data type	Default value	Description
Target velocity reached	InVelocity	BOOL	FALSE	Indicates that the set velocity has reached the target velocity. After reaching, the speed is maintained until Execute turns FALSE or the control is interrupted. When the target value has been changed due to change at ContinuousUpdate TRUE, it turns FALSE until the set velocity reaches the changed target velocity.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Controlling	Active	BOOL	FALSE	Indicates that the FB is controlling the axis.
Abortion of execution	CommandAborted	BOOL	FALSE	Indicates that another FB has aborted the execution of the FB. Turns TRUE when this FB is stopped by an error or this FB is started during an error occurence. Turns FALSE when Execute is FALSE.
Error	Error	BOOL	FALSE	When TRUE, it indicates that an error has occurred in the FB.
Error code	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned. For details, refer to the following. ☞ Page 662 List of Error Codes

Control details

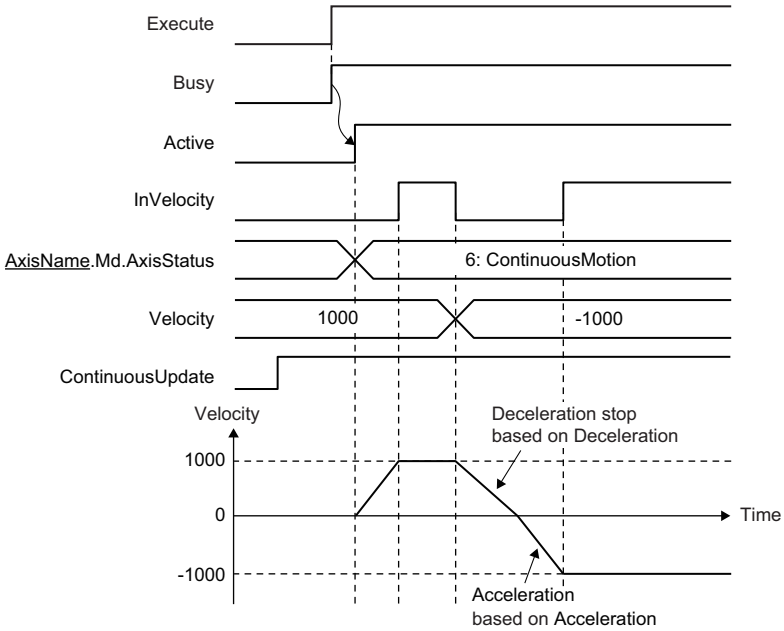
Set the control mode of the driver side to csp and execute the speed control for the specified axis with the specified speed.
MCv_SpeedControl is used to execute. To stop the axis, use MC_Stop or start another operation instruction.
To execute the speed control with csv of the control mode of the driver side, use MC_MoveVelocity, etc.

Timing chart at start and the stop cause occurrence



Timing chart when the operation direction is changed

When the operation direction is changed due to the case that sign of the target velocity is reversed at ContinuousUpdate TRUE, etc., the axis accelerates toward the target velocity once after deceleration stop.



BufferMode

Select a buffer mode. Aborting, Buffered, BlendingLow, BlendingPrevious, BlendingNext, and BlendingHigh can be set.
For details on the operation, refer to the following.

☞ Page 132 Multiple Start (Buffer Mode)

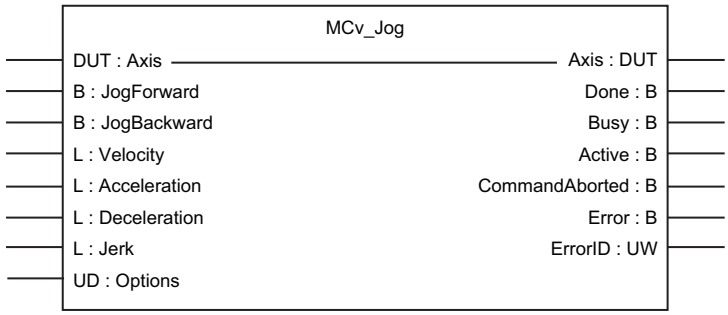
6.3 Single Axis Manual Control

This section describes the single axis manual control.

Relevant variables/FB

Variable/Structure name		Name	Details
AxisName.Cd			
	SwStrokeLimit_Override	Software stroke limit override	Temporarily switches valid/invalid of the software stroke limit check.
	HwStrokeLimit_Override	Hardware stroke limit override	Temporarily switches valid/invalid of the hardware stroke limit check.

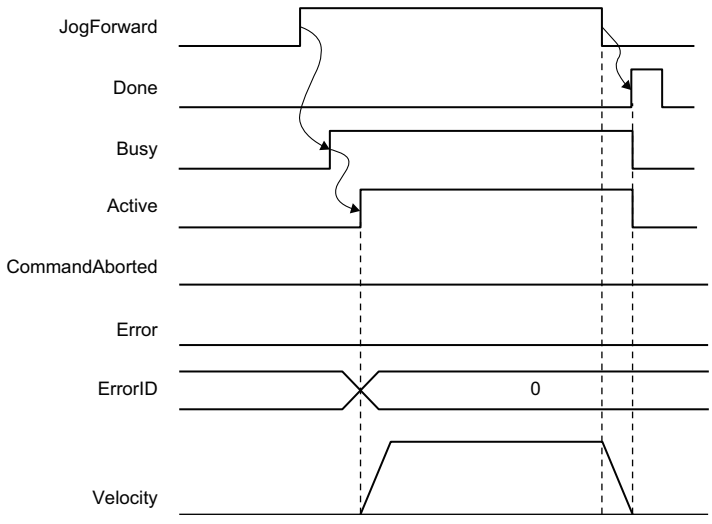
MCv_Jog

Item	Description
Function overview	Executes JOG operation according to the target velocity.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

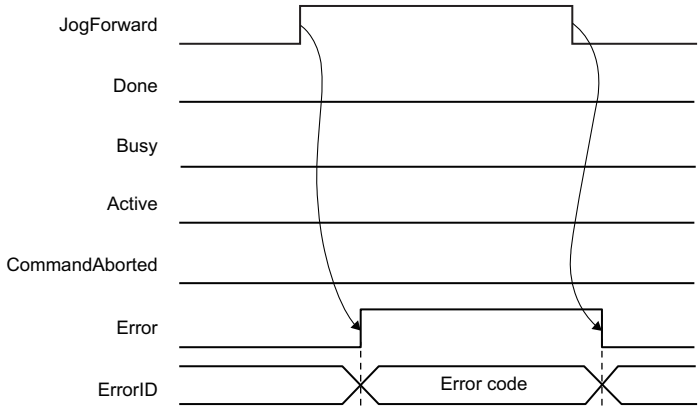
Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
52	8	Subroutine type	Real-time execution

■Timing chart

[When the FB is normally completed]



[When an error occurs]



■I/O variable

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Axis information	Axis	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables

■Input variables

Import ⎍: Always, ↑: At start

Name	Variable name	Data type	Import	Setting range	Default value	Description
Positive rotation JOG command	JogForward	BOOL	⎍	TRUE, FALSE	FALSE	When TRUE, it executes FB.
Reverse rotation JOG command	JogBackward	BOOL	⎍	TRUE, FALSE	FALSE	When TRUE, it executes FB.
Target velocity	Velocity	LREAL	↑	☞ Page 87 Velocity set range	0.0	Sets the set velocity.
Acceleration	Acceleration	LREAL	↑	☞ Page 335 Input variables to use	0.0	Sets the acceleration.
Deceleration	Deceleration	LREAL	↑	☞ Page 335 Input variables to use	0.0	Sets the deceleration.
Jerk	Jerk	LREAL	↑	☞ Page 335 Input variables to use	0.0	Sets the jerk.
Options	Options	DWORD(HEX)	↑	*1	00000000H	Sets a function option with bit specification.

*1 For bits and the function description, refer to the following table.

Bit	Function description
0 to 2	Acceleration/deceleration method setting 0: mcAccDec (Acceleration/deceleration specified method) 1: mcFixedTime (Acceleration/deceleation time constant method) For details, refer to the following. ☞ Page 330 Acceleration/deceleration Processing Function
3 to 31	Empty (Specify "0".)*1

*1 If specifying other than "0", the error "Out of Options Range" (error code: 1A4EH) occurs, and the function does not start.

■Output variables

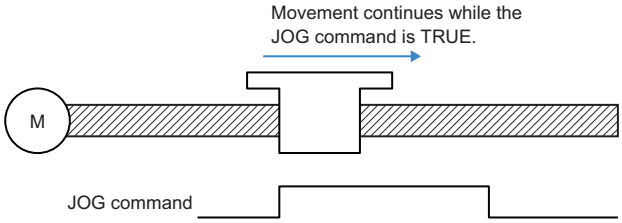
Name	Variable name	Data type	Default value	Description
Execution completion	Done	BOOL	FALSE	When deceleration stop by JOG command OFF is completed, Done bit turns TRUE for only one scan.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Controlling	Active	BOOL	FALSE	Indicates that the FB is controlling the axis.
Abortion of execution	CommandAborted	BOOL	FALSE	Indicates that the execution is interrupted.
Error	Error	BOOL	FALSE	When TRUE, it indicates that an error has occurred in the FB.
Error code	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned. For details, refer to the following. ☞ Page 662 List of Error Codes

Types of manual control

There is the following type of manual control.

JOG operation

JOG operation is a control method in which the machine is moved by only a movement amount (commands are continuously output while the JOG command is TRUE).



JOG operation

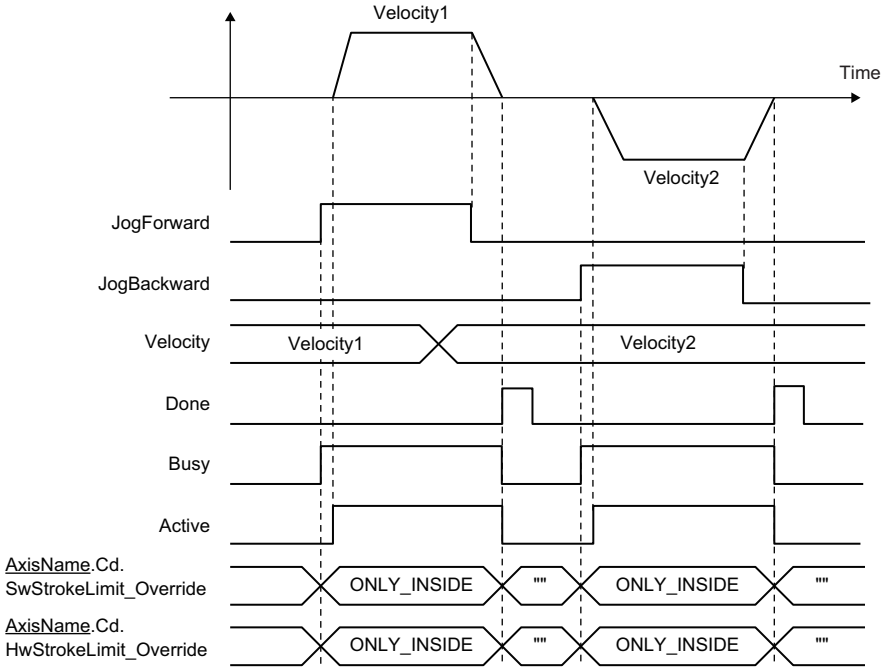
In JOG operation, the command is output from the motion system to the axis while the positive/reverse rotation JOG command is input by using MCv_Jog, and then the axis operates to the specified direction.

Point

- JOG operation can be executed even though the homing is not completed.
- JOG operation can be started from outside to inside of the software stroke limit range. (If the JOG operation is started to outside direction of the software stroke limit range, an error will occur.)
- JOG operation can be started from outside to inside of the hardware stroke limit range. (If the JOG operation is started to outside direction of the hardware stroke limit range, an error will occur.)

Ex.

When a reverse rotation JOG operation is executed after a positive rotation JOG operation



Control details

- JOG operation is carried out with MCv_Jog.
- The target axis is moved to the specified direction during TRUE, by turning JogForward or JogBackward TRUE.
- The axis status (AxisName.Md.AxisStatus) during JOG operation is "6: ContinuousMotion".
- A deceleration stop is performed based on the settings of Deceleration and Jerk, by turning JogForward or JogBackward FALSE.
  The axis status (AxisName.Md.AxisStatus) changes to "4: Standstill" after the deceleration stop is completed.
- When Error turns TRUE during the deceleration by turning JogForward or JogBackward FALSE, Error will be TRUE until JogForward or JogBackward is turned TRUE.
- When another operation instruction is started during a JOG operation, the operation is carried out based on the BufferMode setting.
- When a JOG operation is started during another operation instruction, the start request will be ignored and the warning "Start during Operation Warning" (warning code: 0D01H) will occur. Start a JOG operation when the axis status (AxisName.Md.AxisStatus) is "4: Standstill".
- Use velocity change by override function to change the velocity during JOG operation. For details, refer to the following.
  ☞ Page 359 Override Function

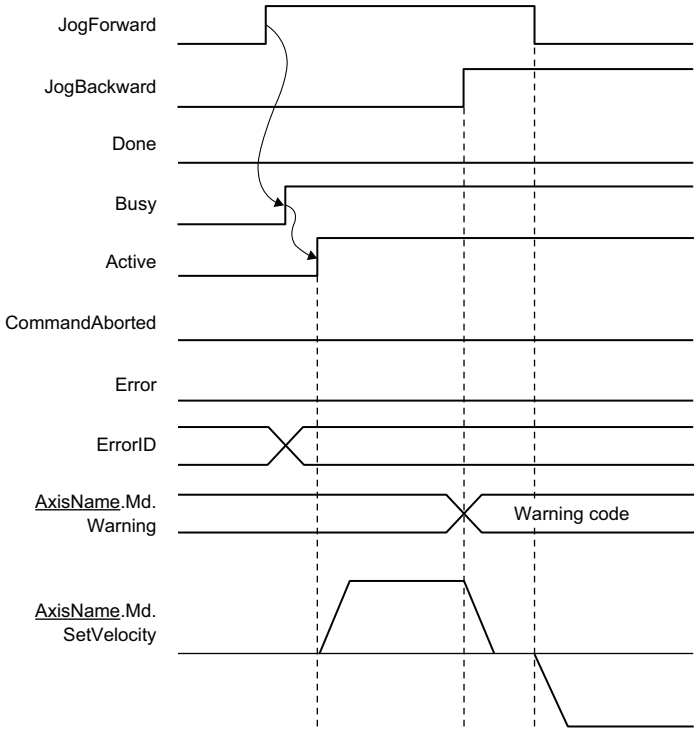
Details of setting items

■JogForward (Positive rotation JOG operation)/JogBackward (Reverse rotation JOG operation)

- Positive rotation JOG operation and reverse rotation JOG operation are started by turning JogForward and JogBackward TRUE respectively, and the target axis is moved to the specified direction during TRUE.
- If JogForward and JogBackward are turned TRUE simultaneously, Busy will turn TRUE, but the warning "JOG Command Input at Both Directions Warning" (warning code: 0D02H) will be detected, CommandAborted turns TRUE, and the JOG operation is not carried out.
  After that, both direction JOG operation turns to FALSE and as only one of the JOG operation turns to TRUE after CommandAborted turns to FALSE, the operation is carried out.
- If the other JOG command of the same FB turns TRUE during a JOG operation, the warning "JOG Command Input at Both Directions Warning" (warning code: 0D02H) will be detected and a deceleration stop will be executed from the point where TRUE is detected. To execute JOG operation, turn TRUE either JogForward or JogBackward.

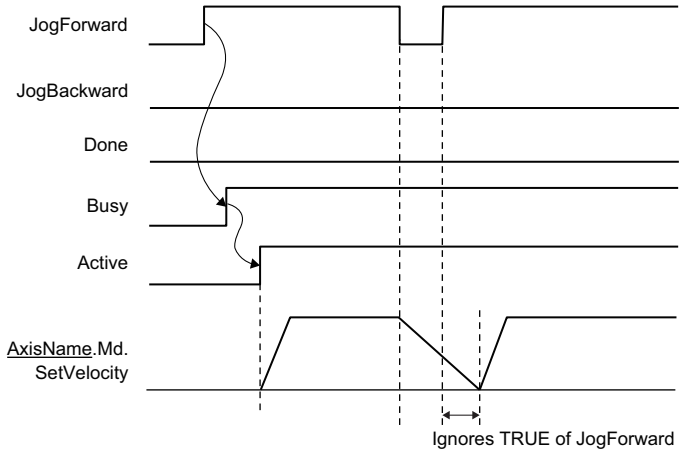
Ex.

When the other JOG command is TRUE during operation



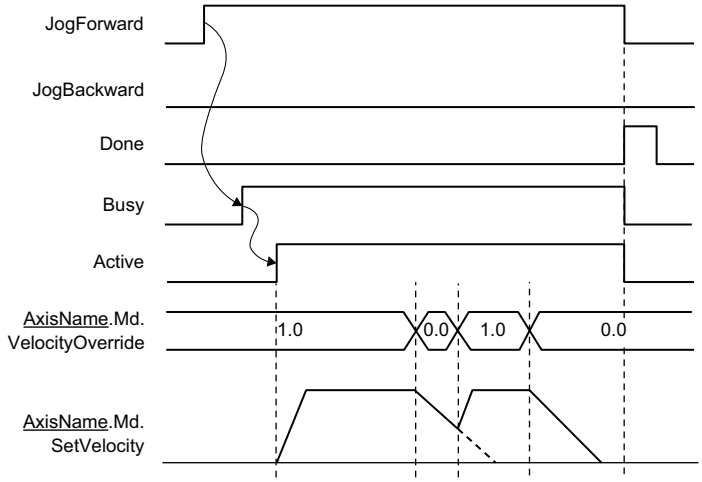
- According to FALSE of JogForward and JogBackward during deceleration, re-acceleration will not be performed even if JOG command turns TRUE. After the deceleration stop is completed, the acceleration will be performed.

Ex.



- Deceleration stop and acceleration are required repeatedly while moving by JOG operation, uses velocity change by override function.
  For details on the override function, refer to the following.
  ☞ Page 359 Override Function

Ex.



■Velocity (Target velocity)/Acceleration (Acceleration)/Deceleration (Deceleration)/Jerk (Jerk)

- Set Target velocity, Acceleration, Deceleration, and Jerk during JOG operation.
- The settings are imported when JOG operation is started by turning JogForward or JogBackward TRUE. A change during JOG operation is not accepted.
- For details on combined operation, refer to the following.
  ☞ Page 87 Velocity set range, ☞ Page 330 Acceleration/deceleration Processing Function

■Options (Options)

- Set the function options to be used. The details of bit and the functions are shown below.

Bit	Function description
0 to 2	Acceleration/deceleration method setting 0: mcAccDec (Acceleration/deceleration specification method) 1: mcFixedTime (Acceleration/deceleration time-fixed method) For details, refer to the following. ☞ Page 330 Acceleration/deceleration Processing Function
3 to 31	Emtpy (Specify "0".)*1

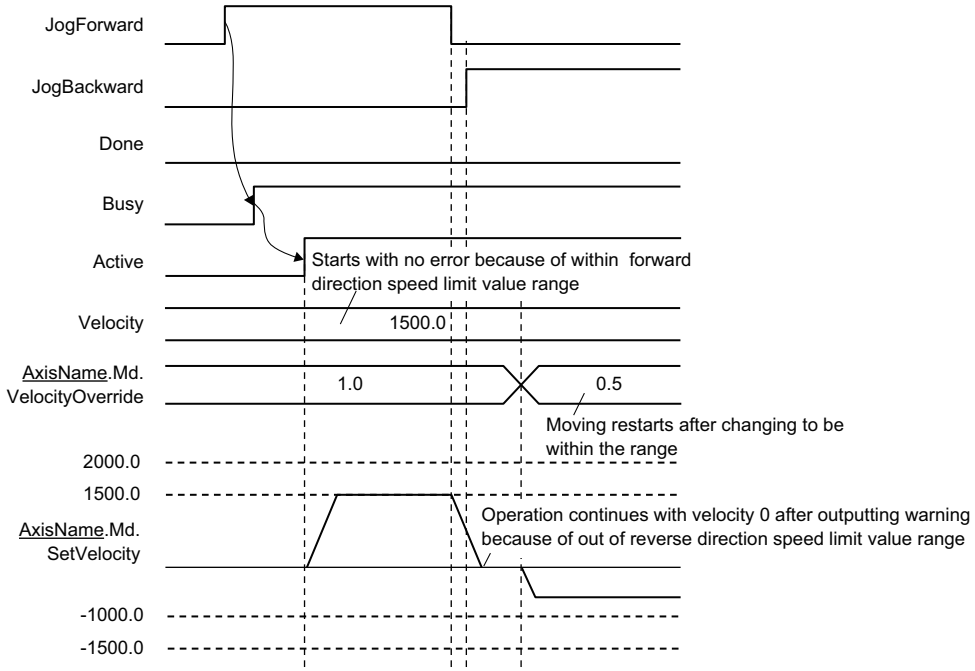
*1 If specifying other than "0", the error "Out of Options Range" (error code: 1A4EH) occrurs and the function does not start.

Precautions

- For safety, set a small value in Velocity first and check the movement, and then gradually increase the value.
- Use the hardware stroke limit function when carrying out JOG operation near the upper or lower limits. If the hardware stroke limit function is not used, the workpiece may exceed the moving range and cause an accident.
- When software stroke limit function is valid and AxisName.Cd.SwStrokeLimit_Override is other than DISABLE or ONLY_INSIDE, rewrites AxisName.Cd.SwStrokeLimit_Override into ONLY_INSIDE during MCv_Jog execution, and rewrites it into "" during JOG operation completion.
- When software stroke limit function is valid and AxisName.Cd.HwStrokeLimit_Override is other than DISABLE or ONLY_INSIDE, rewrites AxisName.Cd.HwStrokeLimit_Override into ONLY_INSIDE during MCv_Jog execution, and rewrites it into "" during JOG operation completion.
- Do not change AxisName.Cd.SwStrokeLimit_Override and AxisName.Cd. HwStrokeLimit_Override during MCv_Jog execution.
- Note that the value of AxisName.Cd.SwStrokeLimit_Override and AxisName.Cd. HwStrokeLimit_Override are retained to the next instruction during multiple start because the next instruction is analyzed during operation when multiple start is executed during JOG operation.
- When moving to negative direction against at start after deceleration stop during JOG operation, and when the target velocity exceeds the velocity limit value, the warning "Speed Limit Value Over Warning on Direction Change" (warning code: 0D20H) occurs. Also, when the acceleration time exceeds 8400 seconds, the warning "Acceleration Time Over Warning on Direction Change" occurs. (The operation continues with velocity 0.) The moving starts after changing control to remove the cause.

Ex.

When setting AxisName.Pr.VelocityLimit_Positive = 2000.0, and AxisName.Pr.VelocityLimit_Negative = 1000.0



6.4 Multiple Axes Positioning Control

The details and usage of the multiple axes positioning controls are explained in this section.
The multiple axes positioning controls use address information and execute positioning to the specified position using interpolation control.

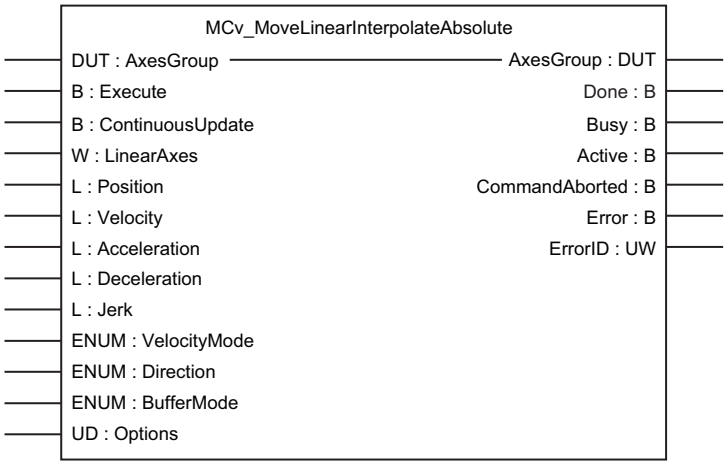
Relevant variables/FBs

Variable/Structure name		Name	Details
AxesGroupName.AxesGroupRef.			
	GroupNo	Axes group No.	Indicates axes group No. 0: No setting From 1 to the maximum number of setting: Set axes group No.
AxesGroupName.Pr.			
	Axis[1..16]	Configuration axis	Sets AxisRef (Axis No.) of the axis configurating the axes group.
	VelocityLimit	Speed limit value	Specifies the speed limit value.
AxesGroupName.Md			
	NumberOfAxes	Number of configuration axes	Indicates the number of configuration axes of the axes group.
	InterpolationAxes	Interpolation axes	Indicates the configuration axis that is executing the interpolation control of the axes group.
	SetVelocity	Set velocity	Indicates the set output velocity of the axes group.

Monitor data in multiple axes positioning control

Configuration axes of the axes group that are used for the multiple axes positioning control can be checked in AxesGroupName.Md.InterpolationAxes.

MCv_MoveLinearInterpolateAbsolute

Item	Description
Function overview	Specifies the target position of the absolute position of specified axes group, and executes positioning by linear interpolation control.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
248	8	Subroutine type	Real-time execution

■I/O variable

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Axes group information	AxesGroup	AXES_GROUP_REF	↑	—	Omission not possible	☞ Page 706 Axes group variables

■Input variables

Import ↑: At start, R: Retrigger possible, C: Continuous update possible

Name	Variable name	Data type	Import	Setting range	Default value	Description
Start	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Continuous update	ContinuousUpdate	BOOL	↑	TRUE, FALSE	FALSE	During TRUE, it can continuously reflect the intput of import "C".
Linear interpolation axis	LinearAxes	INT[0..15]	↑	1 to 16	0	Specifies axes to be used in linear interpolation control from the configuration axes. Specifies the index No. (1 to 16) of the configuration axes in array. When VelocityMode is 2: ReferenceAxisSpeed, the first element of the array is regarded as the reference axis.
Target position	Position	LREAL[0..15]	↑	☞ Page 260 Position (Target position)/ Distance (Movement amount)	0.0	Sets the target position of the absolute position according to unit of axis. It is N-dimensional array data. It is treated as the absolute position of the configuration axis 1, 2, ..., N.
Velocity	Velocity	LREAL	↑/R/C	0, 0.0001 to 2500000000.0	0.0	Sets the speed command value according to unit of axis.
Acceleration	Acceleration	LREAL	↑/R/C	☞ Page 335 Input variables to use	0.0	Sets the acceleration according to unit of axis.
Deceleration	Deceleration	LREAL	↑/R/C	☞ Page 335 Input variables to use	0.0	Sets the deceleration according to unit of axis.
Jerk	Jerk	LREAL	↑	☞ Page 335 Input variables to use	0.0	Sets the jerk according to unit of axis.

Name	Variable name	Data type	Import	Setting range	Default value	Description
Speed mode	VelocityMode	MC_INTERPOLATE_SPEED_MODE	↑	0 to 2	0	Specifies the speed mode of interpolation control. 0: VectorSpeed (Vector speed) 1: LongAxisSpeed (Long axis speed) 2: ReferenceAxisSpeed (Reference axis speed)
Direction selection	Direction	MC_DIRECTION[0..15]	↑	1 to 3	0	Sets the direction selection. It is N-dimensional array data. It is treated as the direction selection of the configuration axis 1, 2, ..., N. 1: mcPositiveDirection (Positive direction) 2: mcNegativeDirection (Negative direction) 3: mcShortestWay (Shortest way)
Buffer mode	BufferMode	MC_BUFFER_MODE	↑	0 to 5	0	Selects the buffer mode. 0: mcAborting (Aborting) 1: mcBuffered (Buffered) 2: mcBlendingLow (BlendingLow) 3: mcBlendingPrevious (BlendingPrevious) 4: mcBlendingNext (BlendingNext) 5: mcBlendingHigh (BlendingHigh) For details, refer to the following. ☞ Page 132 Multiple Start (Buffer Mode)
Options	Options	DWORD(HEX)	↑	*1	00000000H	Sets a function option with bit specification.

*1 For a bit and the function description, refer to the following table.

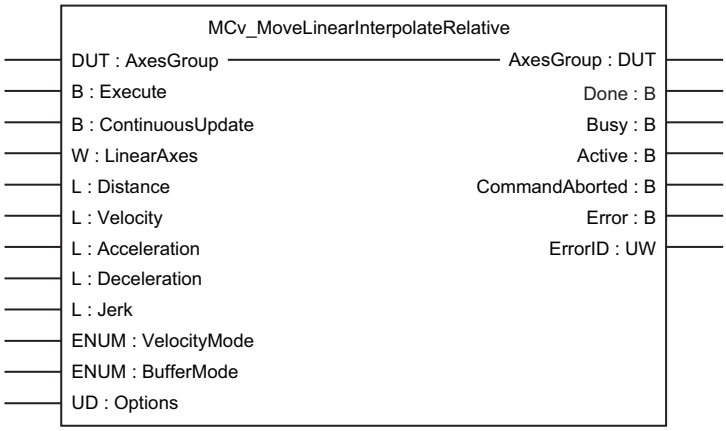
Bit	Function description
0 to 2	Acceleration/deceleration method setting 0: mcAccDec Acceleration/deceleration specification method 1: mcFixedTime Acceleration/deceleration time-fixed method For details, refer to the following. ☞ Page 330 Acceleration/deceleration Processing Function
3 to 15	Empty (Specify "0".)*1
16	The target position specification exceeding the ring counter range Specifies whether to allow the target position that exceeds the ring counter upper/lower limit value or not when the software stroke limit is invalid. 0: Not allow 1: Allow
17 to 31	Empty (Specify "0".)*1

*1 If specifying other than "0", the error "Out of Options Range" (error code: 1A4EH) occurs, and the function does not start.

■Output variables

Name	Variable name	Data type	Default value	Description
Completion	Done	BOOL	FALSE	Indicates that the control is completed. When Execute is TRUE at the operation completion, TRUE remains until Execute turns FALSE. When Execute is FALSE at the operation completion, it becomes TRUE only for one cycle.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Controlling	Active	BOOL	FALSE	Indicates that the axis is being controlled. When multiple FBs are executed for the same axes group, Active of only one FB turns TRUE.
Abortion of execution	CommandAborted	BOOL	FALSE	Indicates that execution of FB has been aborted due to error or multiple start.
Error	Error	BOOL	FALSE	Indicates that an error has occurred.
Error code	ErrorID	WORD (UNIT)	0	The generated error code is returned. For details, refer to the following. ☞ Page 662 List of Error Codes

MCv_MoveLinearInterpolateRelative

Item	Description
Function overview	Specifies the movement amount of the relative position of specified axes group, and executes positioning by linear interpolation control.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
216	8	Subroutine type	Real-time execution

■I/O variable

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Axes groupinformation	AxesGroup	AXES_GROUP_REF	↑	—	Omission not possible	☞ Page 706 Axes group variables

■Input variables

Import ↑: At start, R: Retrigger possible, C: Continuous update possible

Name	Variable name	Data type	Import	Setting range	Default value	Description
Start	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Continuous update	ContinuousUpdate	BOOL	↑	TRUE, FALSE	FALSE	During TRUE, it reflects intput of the import "C".
Linear interpolation axis	LinearAxes	INT[0..15]	↑	1 to 16	0	Specifies axes to be used in linear interpolation control from the configuration axes. Specifies the index No. (1 to 16) of the configuration axes in array. When VelocityMode is 2: ReferenceAxisSpeed, the first element of the array is regarded as the reference axis.
Movement amount	Distance	LREAL[0..15]	↑	☞ Page 260 Position (Target position)/ Distance (Movement amount)	0.0	Sets the relative position according to unit of axis from the current position at start to the end point. It is N-dimensional array data. It is treated as the relative position of the configuration axis 1, 2, ..., N.
Velocity	Velocity	LREAL	↑/R/C	0, 0.0001 to 2500000000.0	0.0	Sets the speed command value according to unit of axis.
Acceleration	Acceleration	LREAL	↑/R/C	☞ Page 335 Input variables to use	0.0	Sets the acceleration according to unit of axis.
Deceleration	Deceleration	LREAL	↑/R/C	☞ Page 335 Input variables to use	0.0	Sets the deceleration according to unit of axis.
Jerk	Jerk	LREAL	↑	☞ Page 335 Input variables to use	0.0	Sets the jerk according to unit of axis.

Name	Variable name	Data type	Import	Setting range	Default value	Description
Speed mode	VelocityMode	MC_INTERPOLATE_SPEED_MODE	↑	0 to 2	0	Specifies the speed mode of interpolation control. 0: VectorSpeed (Vector speed) 1: LongAxisSpeed (Long axis speed) 2: ReferenceAxisSpeed (Reference axis speed)
Buffer mode	BufferMode	MC_BUFFER_MODE	↑	0 to 5	0	Selects the buffer mode. 0: mcAborting (Aborting) 1: mcBuffered (Buffered) 2: mcBlendingLow (BlendingLow) 3: mcBlendingPrevious (BlendingPrevious) 4: mcBlendingNext (BlendingNext) 5: mcBlendingHigh (BlendingHigh) For details, refer to the following. ☞ Page 132 Multiple Start (Buffer Mode)
Options	Options	DWORD(HEX)	↑	*1	00000000H	Sets a function option with bit specification.

*1 For a bit and the function description, refer to the following table.

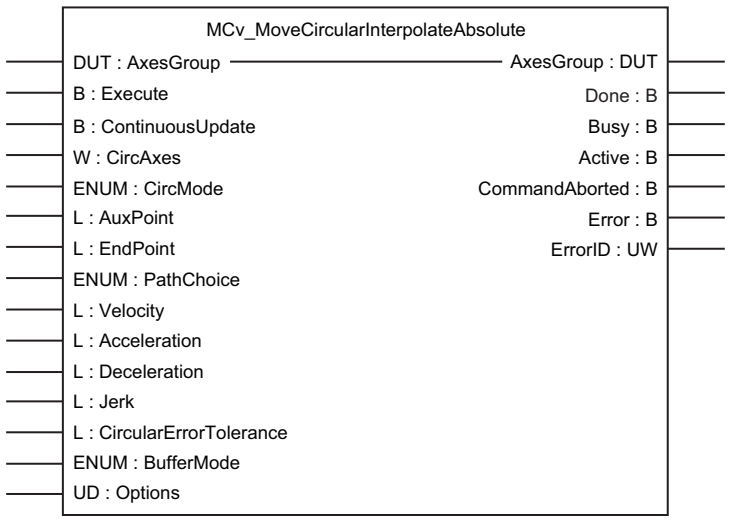
Bit	Function description
0 to 2	Acceleration/deceleration method setting 0: mcAccDec (Acceleration/deceleration specification method) 1: mcFixedTime (Acceleration/deceleration time-fixed method) For details, refer to the following. ☞ Page 330 Acceleration/deceleration Processing Function
3	Position selection during buffer mode 0: Set position 1: Actual position
4 to 31	Empty (Specify "0".)*1

*1 If specifying other than "0", the error "Out of Options Range" (error code: 1A4EH) occurs, and the function does not start.

■Output variables

Name	Variable name	Data type	Default value	Description
Completion	Done	BOOL	FALSE	Indicates that the control is completed. When Execute is TRUE at the operation completion, TRUE remains until Execute turns FALSE. When Execute is FALSE at the operation completion, it becomes TRUE only for one cycle.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Controlling	Active	BOOL	FALSE	Indicates that the axis is being controlled. When multiple FBs are executed for the same axes group, Active of only one FB turns TRUE.
Abortion of execution	CommandAborted	BOOL	FALSE	Indicates that execution of FB has been aborted due to error or multiple start.
Error	Error	BOOL	FALSE	Indicates that an error has occurred.
Error code	ErrorID	WORD (UINT)	0	The generated error code is returned. For details, refer to the following. ☞ Page 662 List of Error Codes

MCv_MoveCircularInterpolateAbsolute

Item	Description
Function overview	Executes positioning by 2-axis circular interpolation by using arbitrary configuration axes of set axes group, and by using the specified end point and the sub point of the absolute position.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
328	8	Subroutine type	Real-time execution

■I/O variable

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Axes group information	AxesGroup	AXES_GROUP_REF	↑	—	Omission not possible	☞ Page 706 Axes group variables

■Input variables

Import ↑: At start, R: Retrigger possible, C: Continuous update possible

Name	Variable name	Data type	Import	Setting range	Default value	Description
Start	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Continuous update	ContinuousUpdate	BOOL	↑	TRUE, FALSE	FALSE	During TRUE, it reflects intput of the import "C".
Circular interpolation axis	CircAxes	INT[0..1]	↑	1 to 16	0	Specifies axes to be used in circular interpolation control from the configuration axes. Specifies the index No. (1 to 16) of the configuration axes in array. The first element of the array is regarded as the reference axis.
Circular interpolation mode	CircMode	MC_CIRC_MODE	↑	0 to 2	0	Sets circular interpolation mode. 0: mcBorder (Border point designation) AuxPoint input defines a point on the arc connecting the start point and the end point. 1: mcCenter (Center point designation) AuxPoint input defines the center point of the circle. 2: mcRadius (Radius designation) AuxPoint input defines the radius of the circle.

Name	Variable name	Data type	Import	Setting range	Default value	Description
Sub point	AuxPoint	LREAL[0..15]	↑	For "0: mcBorder", "1:mcCenter": -10000000000.0 ≤ Seiing value < 10000000000.0[*1] For "2: mcRadius": 0.000001 to 2147483647.0	0.0	It is N-dimentional array data. It is as follows in the circular interpolation mode. For "0: mcBorder" and "1: mcCenter": It is used as the absolute position of the configuration axis 1, 2, …, N. Sets the absolute position depending on the axis unit to the border point and the center point. For "2: mcRadius": The first element is used for the radius, and the second or later elements are ignored.
End point	EndPoint	LREAL[0..15]	↑	-10000000000.0 ≤ Positioning range < 10000000000.0[*1]	0.0	Set the absolute position of end point according to unit of axis. It is N-dimentional array data. It is treated as the absolute position of the configuration axis 1, 2, ..., N.
Path selection	PathChoice	MC_CIRC_PATHCHOICE	↑	0 to 5	0	Sets rotation direction of circular interpolation. 0: mcCW (CW) 1: mcCCW (CCW) 2: mcShortWay (Short way) 3: mcLongWay (Long way) 4: mcCWLongWay (CW long way) 5: mcCCWLongWay (CCW long way) When specifying "0: mcBorder" in CircMode input, the input is ignored. When specifying "2: mcRadius" in CircMode, "0: mcCW" and "1: mcCCW" mean the CW short way and the CCW short way respectively.
Velocity	Velocity	LREAL	↑/R/C	☞ Page 87 Velocity set range	0.0	Sets the speed command value according to unit of axis.
Acceleration	Acceleration	LREAL	↑/R/C	☞ Page 335 Input variables to use	0.0	Sets the acceleration according to unit of axis.
Deceleration	Deceleration	LREAL	↑/R/C	☞ Page 335 Input variables to use	0.0	Sets the deceleration according to unit of axis.
Jerk	Jerk	LREAL	↑	☞ Page 335 Input variables to use	0.0	Sets the jerk according to unit of axis.
Circular interpolation error tolerance	CircularErrorTolerance	LREAL	↑	0.000001 to 100000	100.0	Sets the allowable width of circular interpolation error.
Buffer mode	BufferMode	MC_BUFFER_MODE	↑	0 to 5	0	Selects the buffer mode. 0: mcAborting (Aborting) 1: mcBuffered (Buffered) 2: mcBlendingLow (BlendingLow) 3: mcBlendingPrevious (BlendingPrevious) 4: mcBlendingNext (BlendingNext) 5: mcBlendingHigh (BlendingHigh) For details, refer to the following. ☞ Page 132 Multiple Start (Buffer Mode)
Options	Options	DWORD(HEX)	↑/R	[*2]	00000000H	Sets a function option with bit specification.

*1 The range is ring counter range when the ring counter is enabled.

*2 For a bit and the function description, refer to the following table.

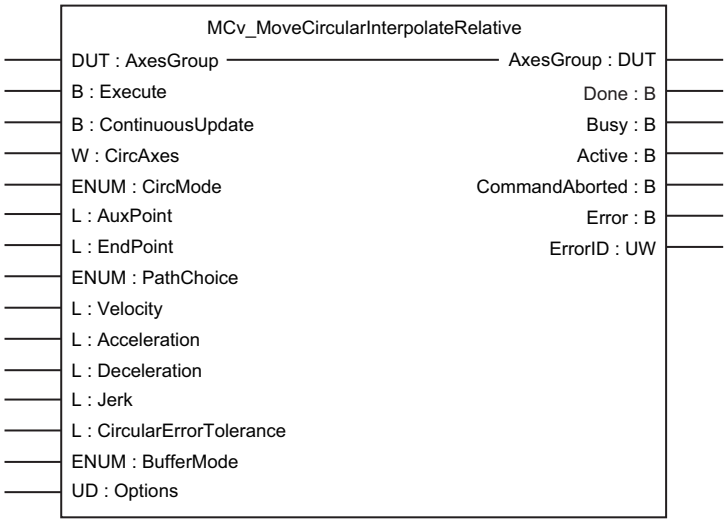
Bit	Function description
0 to 2	Acceleration/deceleration method setting 0: mcAccDec Acceleration/deceleration specification method 1: mcFixedTime Acceleration/deceleration time-fixed method For details, refer to the following. ☞ Page 330 Acceleration/deceleration Processing Function
3 to 31	Empty (Specify "0".)[*1]

*1 If specifying other than "0", the error "Out of Options Range" (error code: 1A4EH) occurs, and the function does not start.

■Output variables

Name	Variable name	Data type	Default value	Description
Completion	Done	BOOL	FALSE	Indicates that the control is completed. When Execute is TRUE at the operation completion, TRUE remains until Execute turns FALSE. When Execute is FALSE at the operation completion, it becomes TRUE only for one cycle.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Controlling	Active	BOOL	FALSE	Indicates that the axis is being controlled. When multiple FBs are executed for the same axes group, Active of only one FB turns TRUE.
Abortion of execution	CommandAborted	BOOL	FALSE	Indicates that execution of FB has been aborted due to error or multiple start.
Error	Error	BOOL	FALSE	Indicates that an error has occurred.
Error code	ErrorID	WORD (UINT)	0	The generated error code is returned. For details, refer to the following. ☞ Page 662 List of Error Codes

MCv_MoveCircularInterpolateRelative

Item	Description
Function overview	Executes positioning by 2-axis circular interpolation by using arbitrary configuration axes of set axes group, and by using the relative position from the specified current position at start to the end point and the sub point.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
328	8	Subroutine type	Real-time execution

■I/O variable

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Axes groupinformation	AxesGroup	AXES_GROUP_REF	↑	—	Omission not possible	☞ Page 706 Axes group variables

■Input variables

Import ↑: At start, R: Retrigger possible, C: Continuous update possible

Name	Variable name	Data type	Import	Setting range	Default value	Description
Start	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Continuous update	ContinuousUpdate	BOOL	↑	TRUE, FALSE	FALSE	During TRUE, it reflects intput of the import "C".
Circular interpolation axis	CircAxes	INT[0..1]	↑	1 to 16	0	Specifies axes to be used in circular interpolation control from the configuration axes. Specifies the index No. (1 to 16) of the configuration axes in array. The first element of the array is regarded as the reference axis.
Circular interpolation mode	CircMode	MC_CIRC_MODE	↑	0 to 2	0	Sets circular interpolation mode. 0: mcBorder (Border point designation) AuxPoint input defines a point on the arc connecting the start point and the end point. 1: mcCenter (Center point designation) AuxPoint input defines the center point of the circle. 2: mcRadius (Radius designation) AuxPoint input defines the radius of the circle.

Name	Variable name	Data type	Import	Setting range	Default value	Description
Sub point	AuxPoint	LREAL[0..15]	↑	For "0: mcBorder", "1:mcCenter": -10000000000.0 to 10000000000.0 For "2: mcRadius": 0.000001 to 2147483647.0	0.0	It is N-dimentional array data. It is as follows in the circular interpolation mode. For "0: mcBorder" and "1: mcCenter": It is used as the relative position of the configuration axis 1, 2, …, N. Sets the relative position depending on the axis unit from current position at start to the border point and the center point. For "2: mcRadius": The first element is used for the radius, and the second or later elements are ignored.
End point	EndPoint	LREAL[0..15]	↑	-10000000000.0 to 10000000000.0	0.0	Set the relative position according to unit of axis from the current position at start to the end point. It is N-dimentional array data. It is treated as the absolute position of the configuration axis 1, 2, ..., N.
Path selection	PathChoice	MC_CIRC_PATHCHOICE	↑	0 to 5	0	Sets rotation direction of circular interpolation. 0: mcCW (CW) 1: mcCCW (CCW) 2: mcShortWay (Short way) 3: mcLongWay (Long way) 4: mcCWLongWay (CW long way) 5: mcCCWLongWay (CCW long way) When specifying "0: mcBorder" in CircMode input, the input is ignored. When specifying "2: mcRadius" in CircMode, "0: mcCW" and "1: mcCCW" mean the CW short way and the CCW short way respectively.
Velocity	Velocity	LREAL	↑/R/C	0, 0.0001 to 2500000000.0	0.0	Sets the speed command value according to unit of axis.
Acceleration	Acceleration	LREAL	↑/R/C	☞ Page 335 Input variables to use	0.0	Sets the acceleration according to unit of axis.
Deceleration	Deceleration	LREAL	↑/R/C	☞ Page 335 Input variables to use	0.0	Sets the deceleration according to unit of axis.
Jerk	Jerk	LREAL	↑	☞ Page 335 Input variables to use	0.0	Sets the jerk according to unit of axis.
Circular interpolation error tolerance	CircularErrorTolerance	LREAL	↑	0.000001 to 100000	100.0	Sets the allowable width of circular interpolation error.
Buffer mode	BufferMode	MC_BUFFER_MODE	↑	0 to 5	0	Selects the buffer mode. 0: mcAborting (Aborting) 1: mcBuffered (Buffered) 2: mcBlendingLow (BlendingLow) 3: mcBlendingPrevious (BlendingPrevious) 4: mcBlendingNext (BlendingNext) 5: mcBlendingHigh (BlendingHigh) For details, refer to the following. ☞ Page 132 Multiple Start (Buffer Mode)
Options	Options	DWORD(HEX)	↑	*1	00000000H	Sets a function option with bit specification.

*1 For a bit and the function description, refer to the following table.

Bit	Function description
0 to 2	Acceleration/deceleration method setting 0: mcAccDec (Acceleration/deceleration specification method) 1: mcFixedTime (Acceleration/deceleration time-fixed method) For details, refer to the following. ☞ Page 330 Acceleration/deceleration Processing Function
3	Position selection during buffer mode 0: Set position 1: Actual position
4 to 31	Empty (Specify "0".)*1

*1 If specifying other than "0", the error "Out of Options Range" (error code: 1A4EH) occurs, and the function does not start.

■Output variables

Name	Variable name	Data type	Default value	Description
Completion	Done	BOOL	FALSE	Indicates that the control is completed. When Execute is TRUE at the operation completion, TRUE remains until Execute turns FALSE. When Execute is FALSE at the operation completion, it becomes TRUE only for one cycle.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Controlling	Active	BOOL	FALSE	Indicates that axis is being controlled. When multiple FBs are executed for the same axes group, Active of only one FB turns TRUE.
Abortion of execution	CommandAborted	BOOL	FALSE	Indicates that execution of FB has been aborted due to error or multiple start.
Error	Error	BOOL	FALSE	Indicates that an error has occurred.
Error code	ErrorID	WORD (UINT)	0	The generated error code is returned. For details, refer to the following. ☞ Page 662 List of Error Codes

Operation in multiple axes positioning control

In the multiple axes positioning control, the positioning control with multiple axes using an axes group can be executed.
To execute the multiple axes positioning control, axes group settings and enabling are required. For details, refer to the following.
☞ Page 65 Axes Group, ☞ Page 77 Axes Group Assignment
The positioning control includes the following types.

Classification of positioning control	Classification of multiple axes	Classification of interpolation	Classification of control
Multiple axes positioning control	Multiple axes interpolation control	Linear interpolation control	Absolute value linear interpolation control
			Relative value linear interpolation control
		Circular interpolation control	Absolute value circular interpolation control
			Relative value circular interpolation control

Positioning speed in multiple axes positioning control

Positioning speed at multiple axes positioning control can be set in the Velocity input of the motion control FB. In the multiple axes positioning control, a control target is controlled at the specified positioning speed.

Linear interpolation control

To specify the positioning speed for the linear interpolation control, the following three methods can be used.

- Vector speed
- Long axis speed
- Reference axis speed

The motion system control methods for each set method are described in the following.

■Vector speed specification

0: VectorSpeed

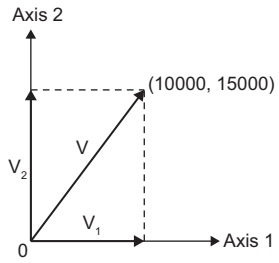
The positioning speed of each axis is calculated from the movement amount of each axis by the motion system based on the positioning speed of the set control target.

The positioning speed of the control target is called the vector speed.

Set the Position input or the Distance input as the movement amount of each axis and set the Velocity input as the vector speed.

Ex.

The following shows the case for 2-axis linear interpolation control.



Program example

Setting item	Setting value
Axis 1 movement amount (D_1)	10000 [pulses]
Axis 2 movement amount (D_2)	15000 [pulses]
Velocity unit of axes group	[s]
Vector speed (V)	7000 [pulses/s]

For the above conditions, the positioning speed of each axis is calculated by the motion system with the following calculation formulas.

Setting item	Setting value
Axis 1 positioning speed	$V_1 = V \times D_1 / \sqrt{(D_1^2 + D_2^2)}$
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 to the Velocity input which is as the vector speed. For details, refer to the following.

☞ Page 356 Speed Limit

■Long axis speed specification

1: LongAxisSpeed

The control is carried out based on the positioning speed (the long axis speed) of the axis whose movement amount is the largest in the addresses set in each axis.

The positioning speed of the other axes is calculated by the motion system from the movement amount of each axis.

Set the Position input or the Distance input as the movement amount of each axis and Velocity input as the long axis speed in the function block for the motion control.

Ex.

The following shows the case for 4-axis linear interpolation control.

Program example

Setting item	Setting value
Axis 1 movement amount (D_1)	10000 [pulses]
Axis 2 movement amount (D_2)	15000 [pulses]
Axis 3 movement amount (D_3)	5000 [pulses]
Axis 4 movement amount (D_4)	20000 [pulses]
Axis 4 velocity unit	[s]
Long axis speed (V)	7000 [pulses/s]

In the above case, the reference axis is the axis 4 whose movement amount is the largest, and the control is carried out based on the positioning speed specified to the axis 4. The positioning speed of the other axes is calculated by the motion system with the following calculation formulas.

Setting item	Setting value
Axis 1 positioning speed	$V_1 = D_1 / D_4 \times V$
Axis 2 positioning speed	$V_2 = D_2 / D_4 \times V$
Axis 3 positioning speed	$V_3 = D_3 / D_4 \times V$

Point

- For the long axis speed specification, set speed limit value is valid for the Velocity input which is as the long axis speed. For details, refer to the following.

☞ Page 356 Speed Limit

- Note that the vector speed when specifying the long axis speed may be larger than the speed limit value.

(Ex.)

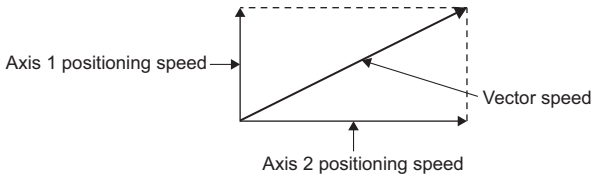
When setting a value like the following in the 2-axis linear interpolation, the vector speed exceeds the speed limit value.

The setting items and the setting values are shown below.

- Axis 1 movement amount: 100 [pulses]
- Axis 2 movement amount: 200 [pulses]
- Long axis speed: 50 [pulses/s]
- Speed limit value: 55 [pulses/s]

In the above case, the reference axis is axis 2 whose movement amount is the largest, and the control is carried out based on the positioning speed specified to the axis 2. The positioning speed and the vector speed of each axis are shown below.

- Axis 1 positioning speed: 100 / 200 × 50 = 25 [pulses/s]
- Axis 2 positioning speed: 50 [pulses/s]
- Vector speed: $\sqrt{(25^2 + 50^2)} = 55.9$ [pulses/s]



The vector speed becomes a value that exceeds the speed limit value 55.

■Reference axis speed specification

2: ReferenceAxisSpeed

The motion system controls the positioning speed of other axes calculating from each axis movement amount based on the positioning speed of the set reference axis (the reference axis speed), and carries out the control.

In MCFB, the top array of the LinearAxes input indicates the reference axis. Also, set the Position input or the Distance input as the movement amount of each axis and set the Velocity input as the single axis speed.

- If "0" is set to the axis movement amount of the axis specified as the reference axis, an error "Reference Axis Movement Amount 0" (error code: 1AABH) occurs.

Ex.

The following shows the case for 4-axis linear interpolation control.

Program example

Setting item	Setting value
Axis 1 movement amount (D_1)	10000 [pulses]
Axis 2 movement amount (D_2)	15000 [pulses]
Axis 3 movement amount (D_3)	5000 [pulses]
Axis 4 movement amount (D_4)	20000 [pulses]
Axis 4 velocity unit	[s]
Reference axis	Axis 4
Reference axis speed (V)	7000 [pulses/s]

In the above case, the reference axis is the axis 4, and control is carried out at the positioning speed specified to the axis 4. The positioning speed of the other axes is calculated by the motion system with the following calculation formula.

Setting item	Setting value
Axis 1 positioning speed	$V_1 = D_1 / D_4 \times V$
Axis 2 positioning speed	$V_2 = D_2 / D_4 \times V$
Axis 3 positioning speed	$V_3 = D_3 / D_4 \times V$

Point

- For the reference axis speed specification, set speed limit value is valid to the Velocity input which is as the reference axis speed. For details, refer to the following.
  ☞ Page 356 Speed Limit
- Note that the positioning speed of the axis whose movement amount is larger than that of the reference axis is larger than the set reference axis speed.

(Ex.)

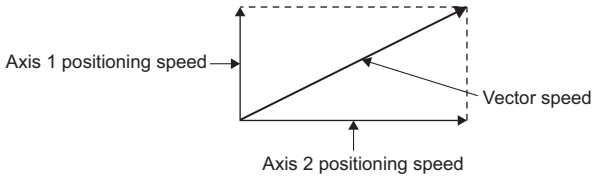
When setting a value like the following in the 2-axis linear interpolation, the positioning speed and the vector speed of the axis 2 exceed the speed limit value.

The setting items and the setting values are shown below.

- Axis 1 movement amount: 100 [pulses]
- Axis 2 movement amount: 200 [pulses]
- Reference axis speed: 50 [pulses/s]
- Reference axis: Axis 1
- Speed limit value: 55 [pulses/s]

In the above case, the reference axis is the axis 1, and control is carried out with the speed limit value set to the axis 1. The positioning speed and the vector speed of each axis are shown below.

- Axis 1 positioning speed: 50 [pulses/s]
- Axis 2 positioning speed: 200 / 100 × 50 = 100 [pulses/s]
- Vector speed: $\sqrt{(50^2 + 100^2)} = 111.8$ [pulses/s]



The positioning speed and the vector speed of the axis 2 become a value that exceeds the speed limit value 55.

Circular interpolation control

For the circular interpolation, the control is performed so that the positioning speed on the circular path becomes the specified speed.

The positioning speed for Blending of the multiple axes positioning control

This section explains operation when selecting BlendingLow, BlendingPrevious, BlendingNext, and BlendingHigh in the BufferMode input of the function block and setting different speed mode to the VelocityMode input of the first function block (FB1) and the second function block (FB2) in the multiple axes positioning control.

- It can start even when VelocityMode is different between FB1 and FB2.
- The control is carried out using the FB1 VelocityMode in FB1 and using the FB2 VelocityMode in FB2.
- During Blending of FB1 and FB2, the speed is not converted by the speed mode. The positioning speed value as the FB1 speed mode is used for Blending in the positioning speed as the FB2 speed mode.
- During Blending, when the VelocityMode of FB1 and FB2 is different, the warning "Multiple Start Speed Mode Specified Mismatch Warning" (warning code: 0D11H) will occur.
- Note that the speed of each axis and the vector speed may rapidly change when VelocityMode of FB1 and FB2 is different during Blending.

Setting value	BufferMode	Description
2: mcBlendingLow	BlendingLow	At the end position of FB1, the speed becomes a lower value of the Velocity input between FB1 and FB2. When the Velocity input of FB2 is lower, at the end position of FB1, it operates with VelcityMode of FB1 and the Velocity setting value of FB2.
3: mcBlendingPrevious	BlendingPrevious	At the end position of FB1, the speed becomes the Velocity input value of FB1. At the end position of FB1, it operates with VelcityMode of FB1 and the Velocity setting value of FB1.
4: mcBlendingNext	BlendingNext	At the end position of FB1, the speed becomes the Velocity input value of FB2. At the end position of FB1, it operates with VelcityMode of FB1 and the Velocity setting value of FB2.
5: mcBlendingHigh	BlendingHigh	At the end position of FB1, the speed becomes a higher value of the Velocity input between FB1 and FB2. When the Velocity input of FB2 is higher, at the end position of FB1, it operates with VelcityMode of FB1 and Velocity of FB2.

■Operation example for BlendingPrevious

Blending from the reference axis speed specification to the vector speed specification is as shown below.

It operates so that the FB1 reference axis speed becomes the speed set to Velocity of FB1 (the reference axis speed) at the FB1 target position. The reference axis speed when reaching the FB1 target position is read as the current vector speed value of FB2 without unit conversion, and it accelerates/decelerates the vector speed set to Velocity of FB2.

- Axis setting

Setting item	Setting value
Position and speed command unit of the configuration axis 1	mm/s
Position and speed command unit of the configuration axis 2	mm/s
Position and speed command unit of the axes group	mm/s

- FB1 input (Command unit: mm/s)

Setting item	Setting value
LinearAxes	[1 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0] or [1 2]
Position	[100 100 0 0 0 0 0 0 0 0 0 0 0 0 0 0] or [100 100]
Velocity	1000
VelocityMode	2: ReferenceAxisSpeed

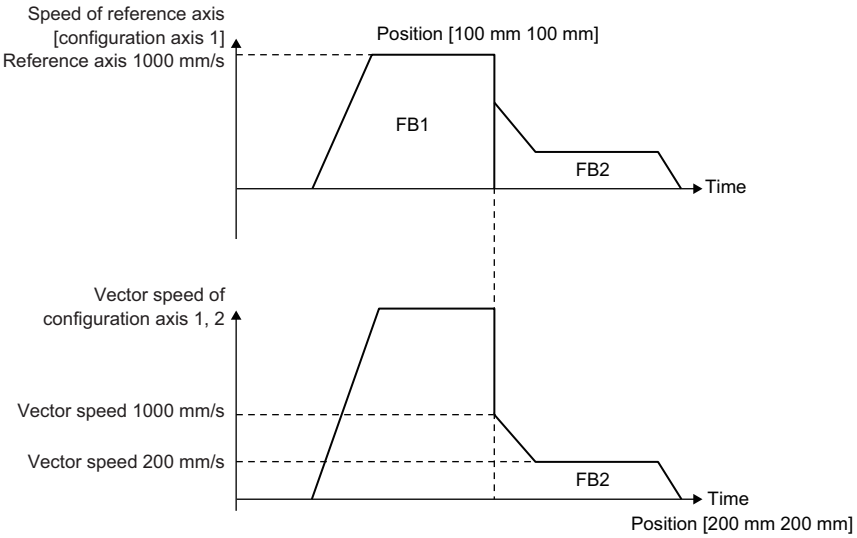
- FB2 input (Command unit: mm/s)

Setting item	Setting value
LinearAxes	[1 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0] or [1 2]
Position	[200 200 0 0 0 0 0 0 0 0 0 0 0 0 0 0] or [200 200]
Velocity	200
VelocityMode	0: VectorSpeed
BufferMode	3: mcBlendingPrevious

Ex.

The following shows an operation example.

- Reaches the target position of FB1 of [100 mm 100 mm] at the reference axis speed of 1000 mm/s of the configuration axis 1 that is the reference axis, 1000 mm/s.
- Starts operation from the current position of FB2 of [100 mm 100 mm] at the vector speed of 1000 mm/s.
- Decelerates to the vector speed of 200 mm/s.
- Performs positioning to the target position of FB2 of [200 mm 200 mm].



■Operation example for BlendingNext

FB1: The reference axis speed → FB2: The vector speed

It operates so that the FB1 reference axis speed becomes the speed set to Velocity of FB2 (the vector speed) at the FB1 target position. The reference axis speed when reaching the FB1 target position is read as the current vector speed value of FB2 without unit conversion, and it accelerates/decelerates the vector speed set to Velocity of FB2.

- Axis setting

Setting item	Setting value
Position and speed command unit of the configuration axis 1	mm/s
Position and speed command unit of the configuration axis 2	mm/s
Position and speed command unit of the axes group	mm/s

- FB1 input (Command unit: mm/s)

Setting item	Setting value
LinearAxes	[1 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0] or [1 2]
Position	[100 100 0 0 0 0 0 0 0 0 0 0 0 0 0 0] or [100 100]
Velocity	1000
VelocityMode	2: ReferenceAxisSpeed

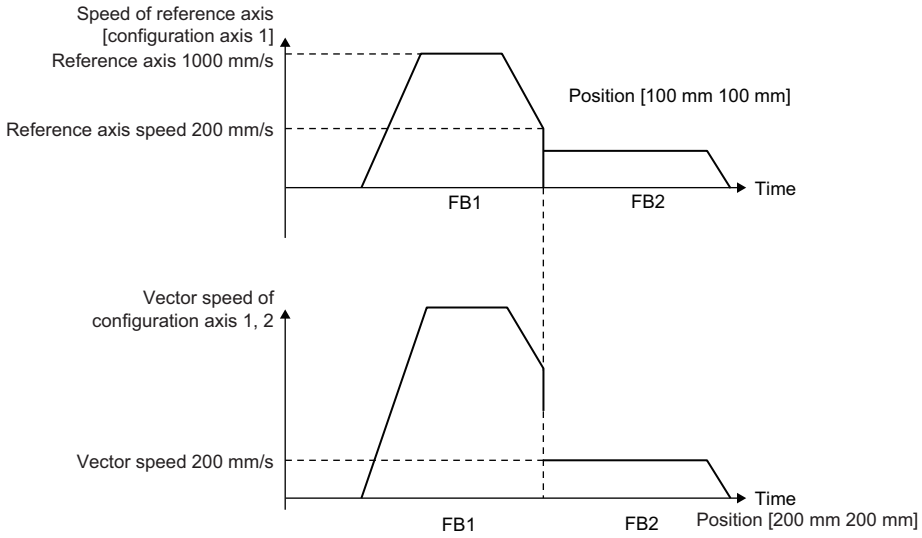
- FB2 input (Command unit: mm/s)

Setting item	Setting value
LinearAxes	[1 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0] or [1 2]
Position	[200 200 0 0 0 0 0 0 0 0 0 0 0 0 0 0] or [200 200]
Velocity	200
VelocityMode	0: VectorSpeed
BufferMode	4: mcBlendingNext

Ex.

The following shows an operation example.

- Operates to the middle of the target position of FB1 of [100 mm 100 mm] at the reference axis speed of the configuration axis 1, 1000 mm/s.
- The configuration axis 1 reaches the target position of FB1 of [100 mm 100 mm] at the reference axis speed of 200 mm/s.
- Starts operation from the current position of FB2 of [100 mm 100 mm] at the vector speed 200 mm/s.
- Performs positioning to the target position of FB2 of [200 mm 200 mm].



Command unit in the multiple axes positioning control

Command unit of the multiple axes positioning control

The command unit in the multiple axes positioning control differs depending on the VelocityMode input of the function block for the motion control. For details and precautions, refer to the following.

☞ Page 79 Technical Units, ☞ Page 145 Multiple start during multiple axes operation

Command unit for Blending of the multiple axes positioning control

This section explains operation when selecting BlendingLow, BlendingPrevious, BlendingNext, and BlendingHigh in the BufferMode input of the function block, and the command unit is different between the first function block (FB1) and the second function block (FB2) in the multiple axes positioning control.

- It can start even when the command unit of FB1 and FB2 is different.
- The control is carried out using the FB1 command unit in FB1 and using the FB2 command unit in FB2.
- During Blending of FB1 and FB2, the speed is not converted by the command unit mode. The positioning speed value as the FB1 command unit is used for Blending in the positioning speed as the FB2 command unit.
- When the command unit of FB1 and FB2 is different, the warning "Speed Command Unit Mismatch Warning" (warning code: 0D1EH) will occur.

Setting value	BufferMode	Description
2: mcBlendingLow	BlendingLow	At the end position of FB1, the speed becomes a lower value of the Velocity input between FB1 and FB2. When the Velocity input of FB2 is lower, the speed at the end position of FB1 is FB1 command unit and the value set in Velocity of FB2.
3: mcBlendingPrevious	BlendingPrevious	At the end position of FB1, the speed becomes the Velocity input value of FB1. The speed at the end position of FB1 is FB1 command unit and the value set in Velocity of FB1.
4: mcBlendingNext	BlendingNext	At the end position of FB1, the speed becomes the Velocity input value of FB2. The speed at the end position of FB1 is FB1 command unit and the value set in Velocity of FB2.
5: mcBlendingHigh	BlendingHigh	At the end position of FB1, the speed becomes a higher value of the Velocity input between FB1 and FB2. When the Velocity input of FB2 is higher, the speed at the end position of FB1 is FB1 command unit and the value set in Velocity of FB2.

■Operation example for BlendingPrevious

During Blending, it can start even when the command unit of FB1 and FB2 is different. Unit conversion is not carried out, and the speed value of FB1 is given as the speed of FB2.

- Axis setting

Setting item	Setting value
Position and speed command unit of the configuration axis 1	mm/s
Position and speed command unit of the configuration axis 2	mm/s
Position and speed command unit of the configuration axis 3	degree/min
Position and speed command unit of the configuration axis 4	degree/min

- FB1 input (Command unit: mm/s from the command unit of the configuration axis 1)

Setting item	Setting value
LinearAxes	[1 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0] or [1 2]
Position	[100 100 0 0 0 0 0 0 0 0 0 0 0 0 0 0] or [100 100]
Velocity	1000
VelocityMode	2: ReferenceAxisSpeed

• FB2 input (Command unit: degree/min from the command unit of the configuration axis 3)

Setting item	Setting value
LinearAxes	[3 4 0 0 0 0 0 0 0 0 0 0 0 0 0 0] or [3 4]
Position	[0 0 200 200 0 0 0 0 0 0 0 0 0 0 0 0] or [0 0 200 200]
Velocity	200
VelocityMode	2: ReferenceAxisSpeed
BufferMode	3: mcBlendingPrevious

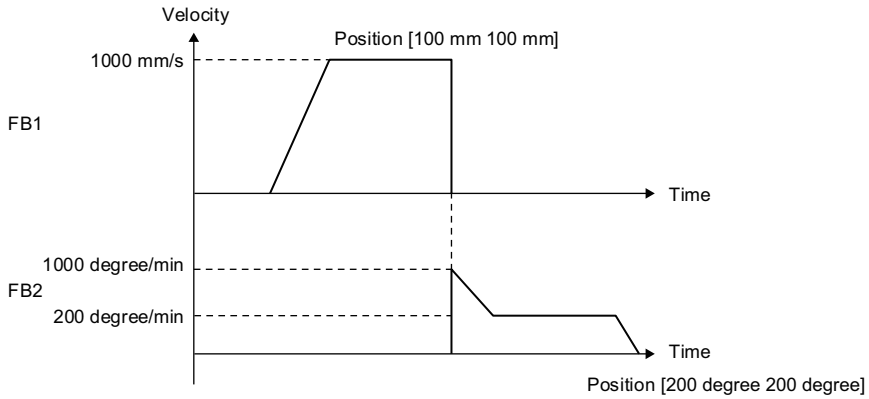
Ex.

The following shows an operation example.

• Reaches the target position of FB1 of [100 mm 100 mm] at the vector speed of 1000 mm/s.

• Starts operation from the current position of FB2 of [0 degree 0 degree] at the vector speed of 1000 degree/min.

• Decelerates to the vector speed of 200 degree/min.



■Operation example for BlendingNext

During Blending, it can start even when the command unit of FB1 and FB2 is different. Unit conversion is not carried out, and the speed value of FB1 is given as the speed of FB2.

• Axis setting

Setting item	Setting value
Position and speed command unit of the configuration axis 1	mm/s
Position and speed command unit of the configuration axis 2	mm/s
Position and speed command unit of the configuration axis 3	degree/min
Position and speed command unit of the configuration axis 4	degree/min

• FB1 input (Command unit: mm/s)

Setting item	Setting value
LinearAxes	[1 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0] or [1 2]
Position	[100 100 0 0 0 0 0 0 0 0 0 0 0 0 0 0] or [100 100]
Velocity	1000
VelocityMode	2: ReferenceAxisSpeed

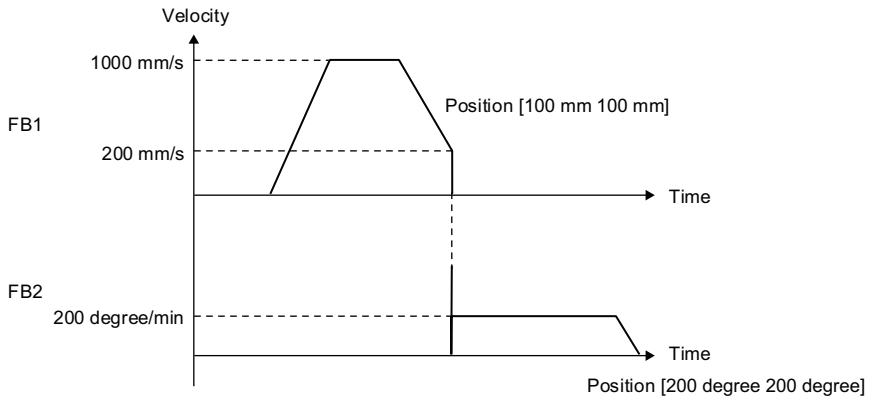
• FB2 input (Command unit: degree/min)

Setting item	Setting value
LinearAxes	[3 4 0 0 0 0 0 0 0 0 0 0 0 0 0 0] or [3 4]
Position	[0 0 200 200 0 0 0 0 0 0 0 0 0 0 0 0] or [0 0 200 200]
Velocity	200
VelocityMode	2: ReferenceAxisSpeed
BufferMode	4: mcBlendingNext

Ex.

The following shows an operation example.

- Operates to the middle of the target position of FB1 of [100 mm 100 mm] at the vector speed of 1000 mm/s.
- Decelerates so that it reaches the target position of FB1 of [100 mm 100 mm] at the vector speed of 200 mm/s.
- Starts operation from the current position of FB2 of [0 degree 0 degree] at the vector speed of 200 degree/min.



Acceleration/deceleration processing

By the VelocityMode input of the function block for the motion control, the relationship between the positioning speed and the acceleration/deceleration processing becomes as shown below.

- Vector speed specification

  0: VectorSpeed

The acceleration/deceleration processing is applied to the vector speed.

- Long axis speed specification

  1: LongAxisSpeed

The acceleration/deceleration processing is applied to the long axis speed.

- Reference axis speed specification

  2: ReferenceAxisSpeed

The acceleration/deceleration processing is applied to the reference axis speed.

There are the following two types of the acceleration/deceleration method. For details on each acceleration/deceleration processing function, refer to the following.

☞ Page 330 Acceleration/deceleration Processing Function

Acceleration/deceleration method	Description
Acceleration/deceleration specification method (Default)	Accelerats and decelerates using the acceleration, the deceleration, and the jerk specified in the FB.
Acceleration/deceleration fixed-time method	Accelerats and decelerates using the acceleration/deceleration time and the jerk specified in the FB regardless of the speed.

State transition in the multiple axes positioning control

The axes group status (AxesGroupName.Md.GroupStatus) of the axes group in the positioning control is "5: GroupMoving".

For the axes group status (AxesGroupName.Md.GroupStatus), refer to the following.

☞ Page 39 State transition

Linear interpolation control

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 maximum 4 axes is performed.

Linear interpolation control includes the following two control methods.

Linear interpolation control	Description
Absolute value linear interpolation control	Specifies the target position of the absolute position, and executes linear interpolation control.
Relative value linear interpolation control	Specifies the relative movement amount from the current position, and executes linear interpolation control.

The absolute value linear interpolation control is executed with MCv_MoveLinearInterpolateAbsolute.

The relative value linear interpolation control is executed with MCv_MoveLinearInterpolateRelative.

Operation chart

The following shows examples of interpolation control for each number of configuration axes used in interpolation control.

■1-axis linear interpolation control (absolute value)

1-axis linear interpolation is executed from the current position at start (start address) to the address set to Position (end address).

Ex.

The control is as shown below for the following conditions.

LinearAxes: [1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]

The current stop address: [1000 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]

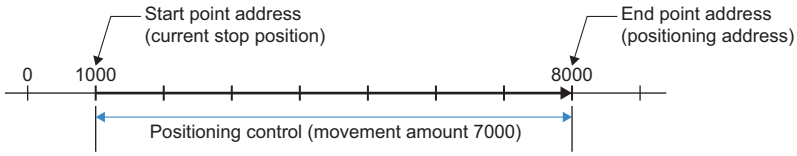
Position: [8000 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]

or

LinearAxes: [1]

The current stop address: [1000]

Position: [8000]

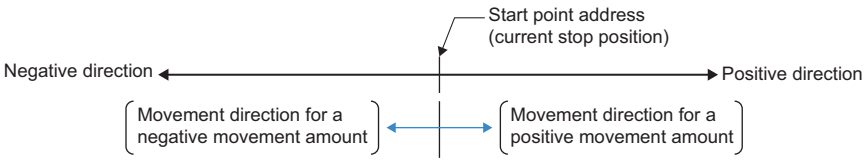


■1-axis linear interpolation control (relative value)

1-axis linear interpolation is executed by the movement amount set to Distance from the current position at start (start address).

The movement direction is determined by a sign (+/-) of the movement amount.

- When the movement direction is positive: Positioning to the positive direction (address increasing direction)
- When the movement direction is negative: Positioning to the negative direction (address decreasing direction)



Ex.

The control is as shown below for the following conditions.

LinearAxes: [1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]

The current stop address: [-3000 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]

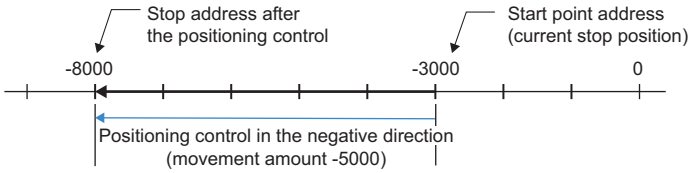
Distance: [-5000 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]

or

LinearAxes: [1]

The current stop address: [-3000]

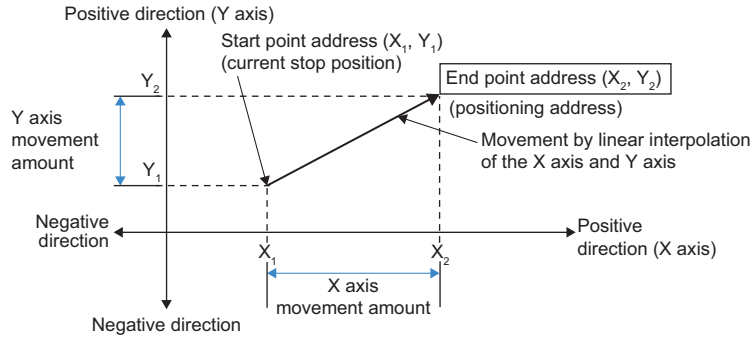
Distance: [-5000]



■2-axis linear interpolation control (absolute value)

2-axis linear interpolation is executed from the current position at start (start address) to the address set to Position (end address).

The movement direction is determined by the current position at start (start address) and the end address of each axis.



Ex.

The control is as shown below for the following conditions.

LinearAxes: [1 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0]

The current stop address: [1000 1000 0 0 0 0 0 0 0 0 0 0 0 0 0 0]

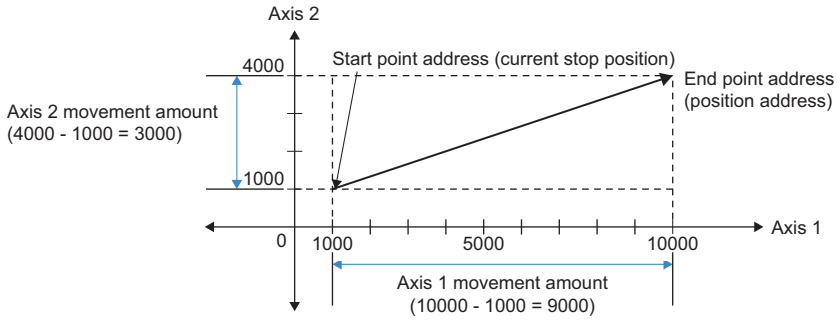
Position: [10000 4000 0 0 0 0 0 0 0 0 0 0 0 0 0 0]

or

LinearAxes: [1 2]

The current stop address: [1000 1000]

Position: [10000 4000]

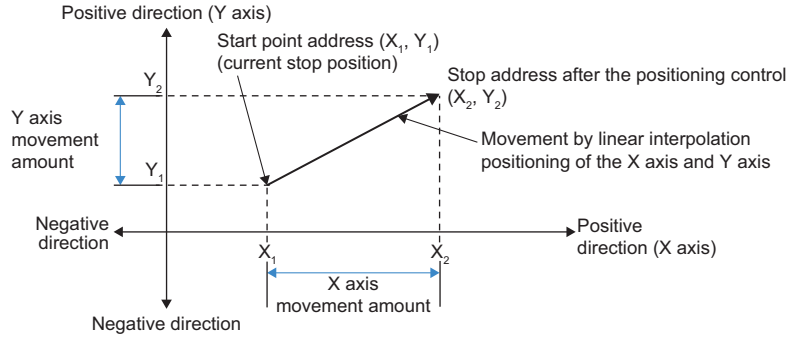


■2-axis linear interpolation control (relative value)

2-axis linear interpolation is executed by the movement amount set to Distance from the current position at start (start address).

The movement direction is determined by a sign (+/-) of the movement amount.

- When the movement direction is positive: Positioning to the positive direction (address increasing direction)
- When the movement direction is negative: Positioning to the negative direction (address decreasing direction)



Ex.

The control is as shown below for the following conditions.

LinearAxes: [1 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0]

The current stop address: [1000 4000 0 0 0 0 0 0 0 0 0 0 0 0 0 0]

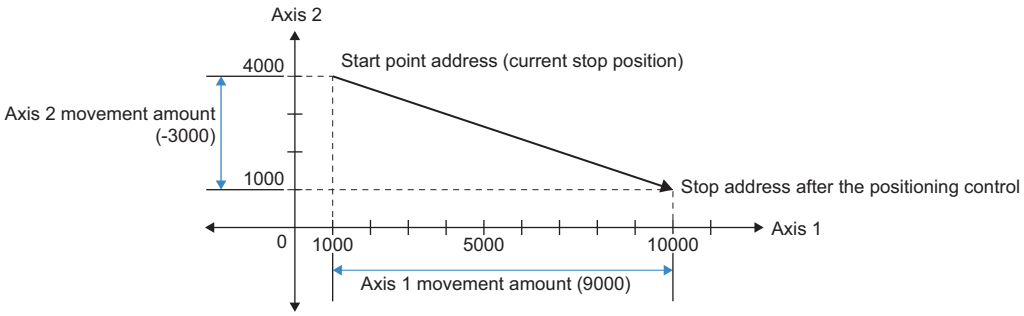
Distance: [9000 -3000 0 0 0 0 0 0 0 0 0 0 0 0 0 0]

or

LinearAxes: [1 2]

The current stop address: [1000 4000]

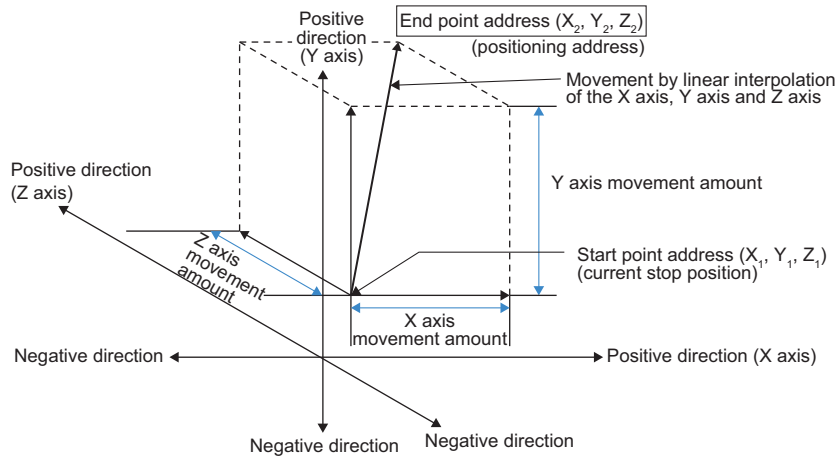
Distance: [9000 -3000]



■3-axis linear interpolation control (absolute value)

3-axis linear interpolation is executed from the current position at start (start address) to the address set to Position.

The movement direction is determined by the current position at start (start address) and the end address of each axis.



Ex.

The control is as shown below for the following conditions.

LinearAxes: [1 2 3 0 0 0 0 0 0 0 0 0 0 0 0 0]

The current stop address: [1000 2000 1000 0 0 0 0 0 0 0 0 0 0 0 0 0]

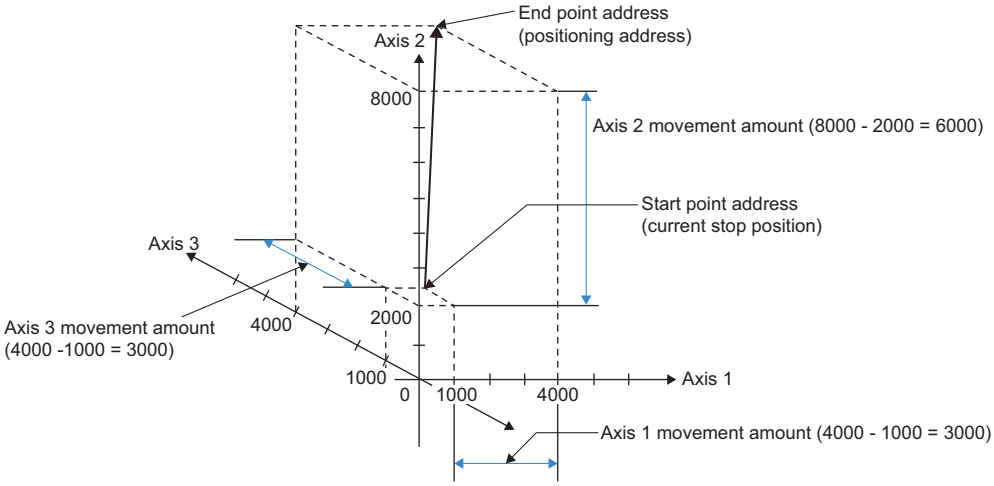
Position: [4000 8000 4000 0 0 0 0 0 0 0 0 0 0 0 0 0]

or

LinearAxes: [1 2 3]

The current stop address: [1000 2000 1000]

Position: [4000 8000 4000]

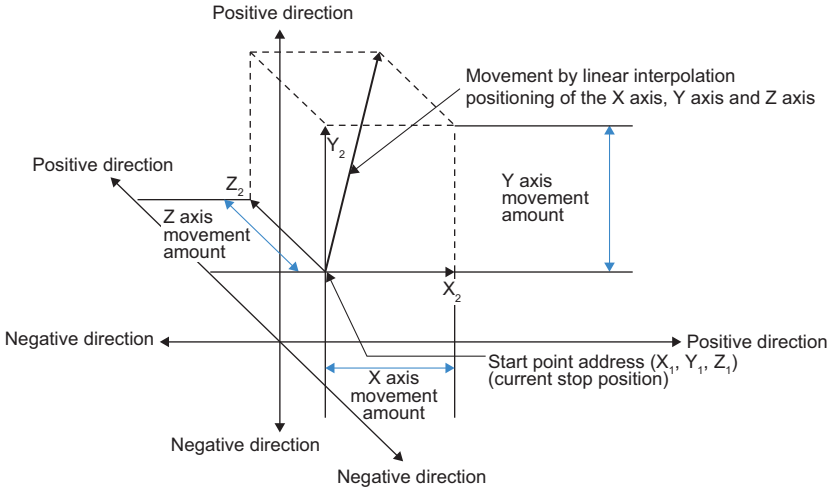


■3-axis linear interpolation control (relative value)

3-axis linear interpolation is executed by the movement amount set to Distance from the current position at start (start address).

The movement direction is determined by a sign (+/-) of the movement amount.

- When the movement direction is positive: Positioning to the positive direction (address increasing direction)
- When the movement direction is negative: Positioning to the negative direction (address decreasing direction)



Ex.

The control is as shown below for the following conditions.

LinearAxes: [1 2 3 0 0 0 0 0 0 0 0 0 0 0 0 0]

The current stop address: [2000 1000 2000 0 0 0 0 0 0 0 0 0 0 0 0 0]

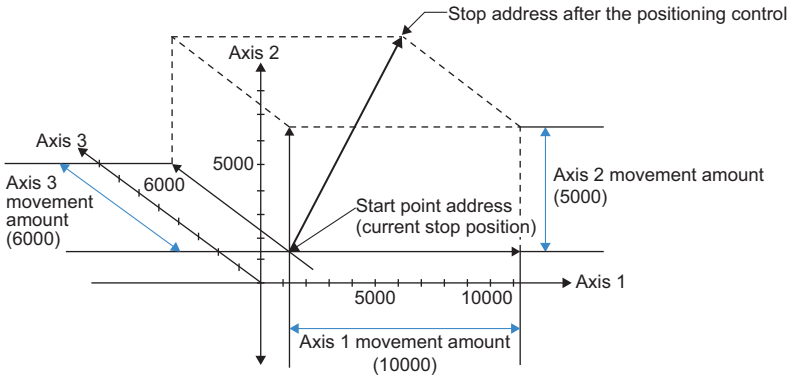
Distance: [10000 5000 6000 0 0 0 0 0 0 0 0 0 0 0 0 0]

or

LinearAxes: [1 2 3]

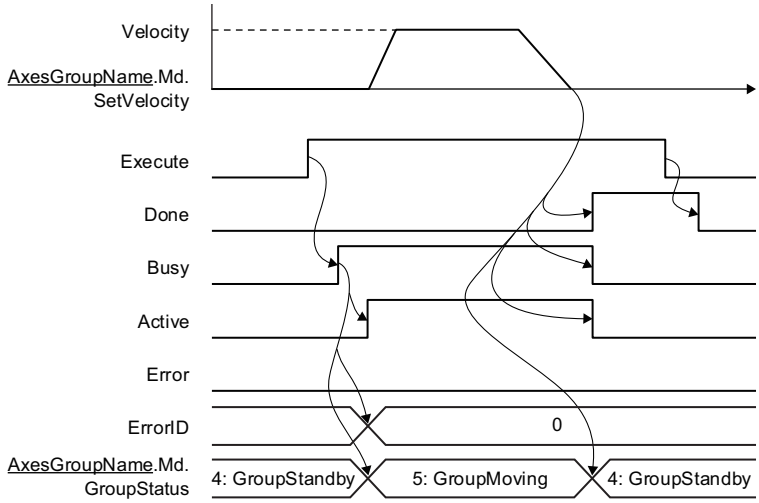
The current stop address: [2000 1000 2000]

Distance: [10000 5000 6000]



Timing chart

■When the FB is normally completed



■When an error occurs

For details when an error occurs, refer to the following.

☞ Page 27 Execute type/Enable type

Details of setting items

■ContinuousUpdate (ContinuousUpdate)

Continuous update of the FB enables to import input variables again without interrupting the operation to the instance in execution.

Input variables that can be change in restart/continuous update are Velocity, Acceleration, and Deceleration.

For details, refer to the following.

☞ Page 131 Retrigger/Continuous Update, ☞ Page 330 RELEVANT FUNCTIONS TO VELOCITY

■LinearAxes (Linear interpolation axis)

In linear interpolation control, linear interpolation can be executed using arbitrary axes from the configuration axes set to the axes group.

Set configuration axes to perform linear interpolation by the LinearAxes input from the configuration axes of the axes group. The LinearAxes input has 16 array elements. For the array, specify the index Nos. (1 to 16: configuration axis 1 to configuration axis 16) of configuration axes to be used for interpolation control by just the number of configuration axes used for control, and specify "0" for the rest. "0" can be omitted.

- The setting number of the linear interpolation axes must be the maximum number of the interpolation axes for interpolation control,4, or less.
- The number of configuration axes to be used for interpolation control must be the number of axes registered to the configuration axes or less.
- When VelocityMode is "2: ReferenceAxisSpeed", the configuration axis of the first element in LinearAxes becomes the reference axis.
- When specifying as follows, the error "Linear Interpolation Axis Setting Incorrect" (error code: 1AB9H) will occur and the axis will not start.

- When specifying the configuration axis where the axis is not set
- When the first element of LinearAxes is "0"
- When specifying the same index No. to LinearAxes in duplicate
- When specifying the number of setting axes of LinearAxes to exceed the maximum number of the interpolation axes

Ex.

When setting configuration axis 2, 3, 4 to the linear interpolation axes

LinearAxes: [2 3 4 0 0 0 0 0 0 0 0 0 0 0 0 0]

or

LinearAxes: [2 3 4]

■Position (Target position)/Distance (Movement amount)

Set the target position of the absolute position or the movement amount from the current position at start by the Position/Distance input. Position/Distance has 16 array elements.

- Position/Distance[0..(N-1)] indicates the target position/movement amount of the configuration axis 1 to N. Set the position to the configuration axes set in LinearAxes.
- The Position/Distance input for the configuration axes which are not set to the linear interpolation axes is ignored.
- The Position/Distance input of axes after all configuration axes set to the linear interpolation axes can be omitted.

Ex.

When setting the configuration axis 2, 3, 4 to the linear interpolation axes, and the target positions of the configuration axis 2, 3, 4 are 2000, 3000, 4000 respectively

LinearAxes: [2 3 4 0 0 0 0 0 0 0 0 0 0 0 0 0]

Position: [0 2000 3000 4000 0 0 0 0 0 0 0 0 0 0 0 0]

or

LinearAxes: [2 3 4]

Position: [0 2000 3000 4000]

■Velocity (Velocity)

Specify the maximum speed of the way. For details, refer to the following.

☞ Page 243 Positioning speed in multiple axes positioning control

■Acceleration (Acceleration)

Specify the acceleration. The setting details differ depending on the acceleration/deceleration method. For details, refer to the following.

☞ Page 330 Acceleration/deceleration Processing Function

■Deceleration (Deceleration)

Specify the deceleration. The setting details differ depending on the acceleration/deceleration method. For details, refer to the following.

☞ Page 330 Acceleration/deceleration Processing Function

■Jerk (Jerk)

Specify the jerk. The setting details differ depending on the acceleration/deceleration method. For details, refer to the following.

☞ Page 330 Acceleration/deceleration Processing Function

■VelocityMode (Velocity mode)

Specify the velocity mode in multiple linear interpolation control. When specifying "2: ReferenceAxisSpeed" in VelocityMode, the reference axis becomes the configuration axis specified to the first elements of the LinearAxes input. For details, refer to the following.

☞ Page 243 Positioning speed in multiple axes positioning control

Setting value	Speed mode name	Description
0: VectorSpeed	Vector speed	Velocity specifies the vector speed.
1: LongAxisSpeed	Long axis speed	Velocity specifies the speed of the long axis.
2: ReferenceAxisSpeed	Reference axis speed	Velocity specifies the speed of the reference axis.

Restriction

- When "0: VectorSpeed" is specified to VelocityMode and the movement value of each axis exceeds "4,294,967,296.0 (= 2^{32})", the error "Out of Linear Movement Amount Range" (error code: 1ADEH) will occur and the axis will not start.
- When Aborting is specified to BufferMode and MCv_MoveLinearInterpolateAbsolute is executed by multiple start, specify "0: VectorSpeed" to VelocityMode. If other than "0: VectorSpeed" is set, the error "Out of Speed Mode Range" (error code: 1A61H) will occur.

■Direction (Direction selection)

It can be selected from the positive direction, the negative direction, and the shortest way. Specify the direction to move from the current position to the target position. Direction has 16 array elements.

- Direction [0..(N-1)] indicates the direction selection of the configuration axis 1 to N. Set the value in the configuration axis set in LinearAxes.
- The direction input of the configuration axis which is not set as the linear interpolation axis is ignored.
- When selecting other than the positive direction, the negative direction, and the shortest way, the error "Out of Direction Selection Range" (error code: 1A37H) will occur and the axis will not start.
- The linear axis where the software stroke limit is valid ignores this setting. For details on operation of each direction selection, refer to the following.

☞ Page 211 Direction (Direction selection)

Setting value	Direction selection name	Description
1: mcPositiveDirection	Positive direction	Executes positioning from the current position to the target position in the positive direction (address increasing).
2: mcNegativeDirection	Negative direction	Executes positioning from the current position to the target position in the negative direction (address decreasing).
3: mcShortestWay	Shortest way	Executes positioning in the direction where the distance between the current position and the target position is close.

■BufferMode (Buffer mode)

Select the buffer mode. Aborting, Buffered, BlendingLow, BlendingPrevious, BlendingNext, and BlendingHigh can be set. For details, refer to the following.

☞ Page 133 Buffer mode type

■Options (Options)

Set a function option with bit specification.

Bit details and the function are as the following table. For details, refer to the following.

☞ Page 330 Acceleration/deceleration Processing Function, ☞ Page 206 Single Axis Positioning Control

Bit	Function description
0 to 2	Acceleration/deceleration method setting 0: mcAccDec (Acceleration/deceleration specification method) 1: mcFixedTime (Acceleration/deceleration time-fixed method) For details, refer to the following. ☞ Page 330 Acceleration/deceleration Processing Function)
3*1	Position selection during buffer mode Specify "0" for absolute value linear interpolation control.*2 0: Set position 1: Actual position For details, refer to the following. ☞ Page 132 Multiple Start (Buffer Mode)
4 to 15	Empty (Specify "0".)*2
16*3	The target position specification exceeding the ring counter value Specifies whether to allow the target position that exceeds the ring counter upper/lower limit value or not when the software stroke limit is invalid. Specify "0" for the relative value linear interpolation control.*2 0: Not allow 1: Allow
17 to 31	Empty (Specify "0".)*2

*1 It is valid only when the relative value linear interpolation control.

*2 When specifying other than "0", the error "Out of Options Range" (error code: 1A4EH) occurs and the function does not start.

*3 It is valid only when the absolute value linear interpolation control.

Circular interpolation control

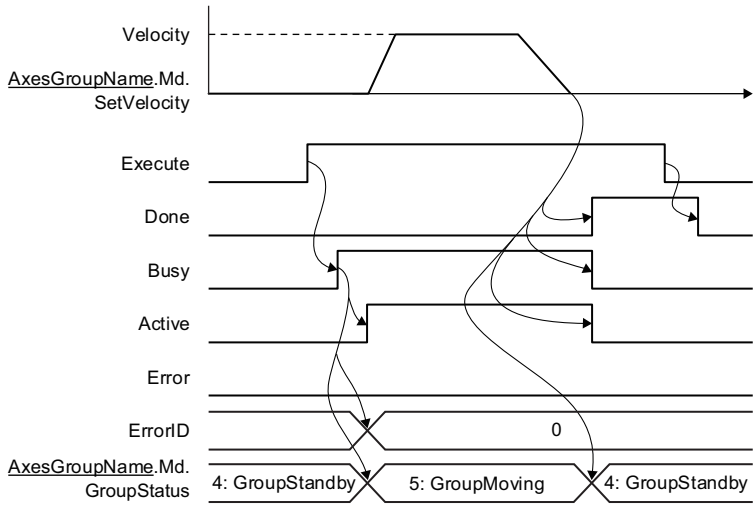
Circular interpolation control specifies an axes group and performs interpolation control so that the path from the start point (move start point) to the end point becomes an arc to a linear axis which is arranged mechanically orthogonally.
In circular interpolation control, interpolation control using arbitrary two axes from the configuration axes set to the axes group.
Three types of the circular interpolation methods, the border point designation, the center point designation, and the radius designation, can be specified for circular interpolation control.
Circular interpolation control has the following two control methods.

Circular interpolation control	Description
Absolute value circular interpolation control	Specifies the end point and the sub point of the absolute position, and executes 2-axis circular interpolation control.
Relative value circular interpolation control	Specifies the end point and the movement amount of the sub point from the current point, and executes 2-axis circular interpolation control.

The absolute value circular interpolation control is executed with MCv_MoveCircularInterpolateAbsolute.
The relative value circular interpolation control is executed with MCv_MoveCircularInterpolateRelative.

Timing chart

■When the FB is normally completed



■When an error occurs

For details when an error occurs, refer to the following.
☞ Page 27 Execute type/Enable type

Details of setting items

■ContinuousUpdate (Continuous update)

Continuous update of the FB enables to import input variables again without interrupting the operation to the instance in execution.

Input variables that can be change in restart/continuous update are Velocity, Acceleration, and Deceleration.

For details, refer to the following.

☞ Page 131 Retrigger/Continuous Update, ☞ Page 330 RELEVANT FUNCTIONS TO VELOCITY

■CircAxes (Circular interpolation axis)

In circular interpolation control, circular interpolation can be executed by using arbitrary two axes from the configuration axes set to the axes group.

Set configuration axes to perform circular interpolation by the CircAxes input from the configuration axes of the axes group.

The CircAxes input has two array elements. For the array, specify the index Nos. (1 to 16: configuration axis 1 to configuration axis 16) of configuration axes to be used for interpolation control. The first element of the array is regarded as the reference axis.

- If a configuration axis which is not set an axis is specified, the error "Circular Interpolation Axis Unset" (error code: 1A62H) occurs.

Ex.

When setting configuration axis 2, 3 to the circular interpolation axes

CircAxes: [2 3]

■CircMode (Circular interpolation mode)

In circular interpolation control, three types of the circular interpolation methods, the border point designation, the center point designation, and the radius designation, can be specified by inputting CircMode. The definition of AuxPoint input differs depending on the CircMode input.

Setting value	Circular interpolation mode name	AuxPoint (Sub point)
0: mcBorder	Border point designation	AuxPoint input defines the border point on the arc connecting the start point and the end point.
1: mcCenter	Center point designation	AuxPoint input defines the center point of the arc.
2: mcRadius	Radius designation	AuxPoint input defines the radius of the arc.

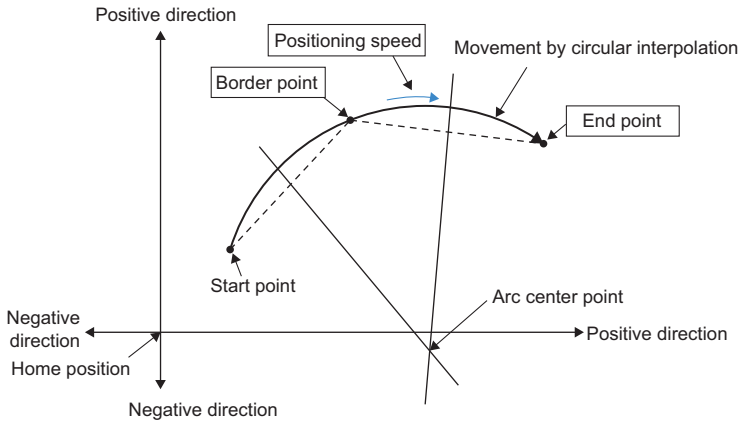
- Circular interpolation control with border point designation

When selecting "0: mcBorder" in the CircMode input, positioning control is carried out in an arc path passing through the specified border point by using two configuration axes specified to CircMode.

• Absolute value circular interpolation control operation with border point designation

In absolute value circular interpolation control with border point designation, positioning is carried out from the current position at start (start point address) to the address set to EndPoint (end point address) in an arc path that passes through the address (border point address) set to AuxPoint.

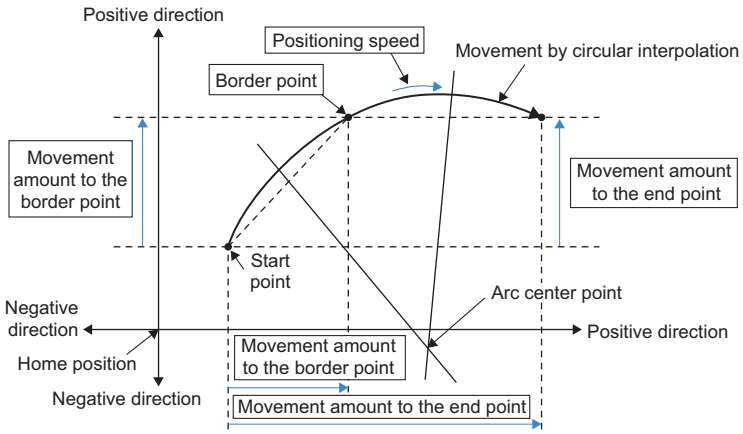
The resulting control path is an arc having as its center the intersection point of perpendicular bisectors of a straight line between the current position at start and the border point address, and a straight line between the border point address and the end point address. However, a perfect circle cannot be drawn with the border point designation. (The end point, the positioning speed, and the border point shown in the following figure indicate the data to be set.)



• Relative value circular interpolation control operation with border point designation

In relative value circular interpolation control with border point designation, positioning is carried out from the current position at start (start point address) to the address of the movement amount set to EndPoint in an arc path (end point address) that passes through the address of the movement amount set to AuxPoint (border point address).

The resulting control path is an arc having as its center the intersection point of perpendicular bisectors of a straight line between the current position at start and the border point address, and a straight line between the border point address and the end point address. (The positioning speed, the border point, the movement amount to the border point, and the movement amount to the end point shown in the following figure indicate the data to be set.)



Restriction

2-axis circular interpolation control 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 is 2147483647.0): The error "Out of Radius Range" (error code: 1A6CH) will occur at positioning start.
- When the border point address exceeds positioning range: The error "Out of Boundary Point Address Range" (error code: 1A64H) will occur.
- When the end point address exceeds positioning range: The error "Out of End Point Address Range" (error code: 1A6DH) will occur.
- When the start point address, border point address, and end point address are in the same straight line: The error "Start Point - Boundary Point - End Point Address Same Straight Line" (error code: 1A6AH) will occur.
- When the start point address is the same as the end point address: The error "Start Point - End Point Address Same Value" (error code: 1A66H) will occur.
- When the start point address is the same as the border point address: The error "Start Point - Boundary Point Address Same Value" (error code: 1A68H) will occur.
- When the end point address is the same as the border point address: The error "End Point - Boundary Point Address Same Value" (error code: 1A69H) will occur.

• Circular interpolation control with center point designation

When selecting "1: mcCenter" in the CircMode input, two configuration axes specified in CircAxes are used to carry out positioning control in an arc path having the specified center point as a center point.

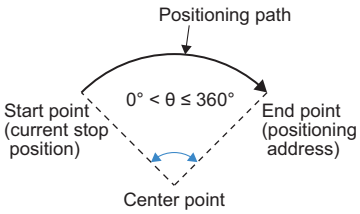
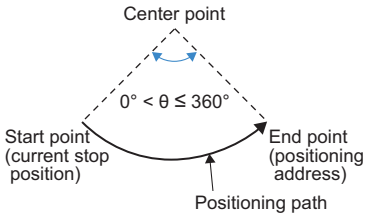
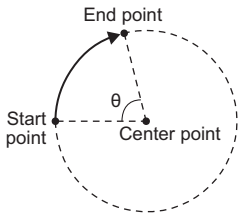
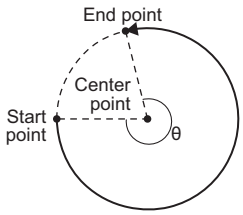
[Circular interpolation path]

The path in the circular interpolation control with center point designation can be set by the PathChoice. The following table shows the setting value, setting description, arc center angle that can be controlled, and path of PathChoice.

When the start point, end point, and center point are on one straight line, and when selecting "2: mcShortWay" or "3: mcLongWay", a semicircle is drawn in CW direction.

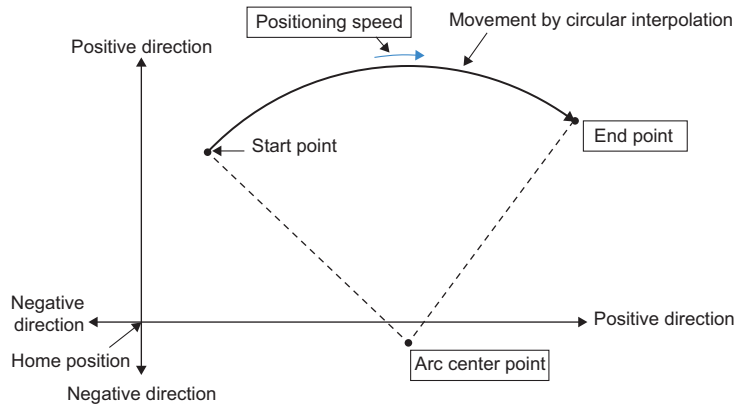
When the start point is same as the end point, and when selecting "2: mcShortWay", an error occurs.

When the start point is same as the end point, and when selecting "3: mcLongWay", a perfect circle is drawn in CW direction.

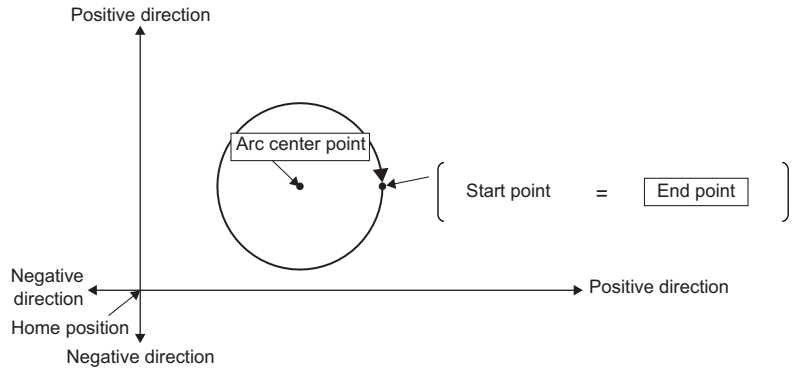
Setting value	Setting description	Arc center angle that can be controlled	Path
0: mcCW	CW	$0° < \theta \leq 360°$	
1: mcCCW	CCW	$0° < \theta \leq 360°$	
2: mcShortWay	Short way • When the start point, end point, and center point are on one straight line, a semicircle is drawn in CW direction. • When the start point is same as the end point, the error "Path Choice Setting Incorrect" (error code: 1AB7H) occurs.	$0° < \theta \leq 180°$	
3: mcLongWay	Long way • When the start point, end point, and center point are on one straight line, a semicircle is drawn in CW direction. • When the start point is same as the end point, a perfect circle is drawn in CW direction.	$180° \leq \theta \leq 360°$	

• Absolute value circular interpolation control operation with center point designation

In absolute value circular interpolation control operation with center point designation, positioning is carried out from the current position at start (start point address) to the end position set to EndPoint in an arc path having the center point set in AuxPoint as its center. (The end point, the positioning speed, and the center point of the arc shown in the following figure indicate the data to be set.)



Positioning of a perfect circle with a radius from the start point address to the arc center point can be carried out by setting the EndPoint to the same address as the start point. (The end point, and the center point of the arc shown in the following figure indicate the data to be set.)



Ex.

To specify EndPoint to the same as the start point, specify the set position of the configuration axes to be used in circular interpolation to the EndPoint input as the following.

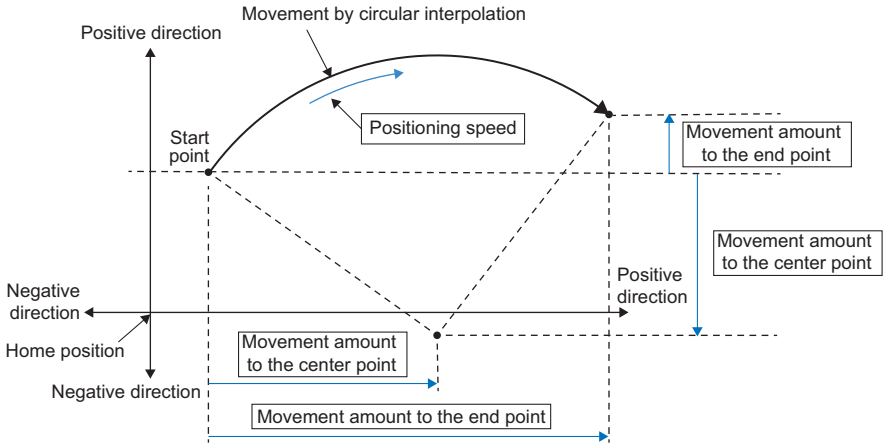
AxesGroupName.Pr.Axis[1] = Axis0001.AxisRef

AxesGroupName.Pr.Axis[2] = Axis0002.AxisRef

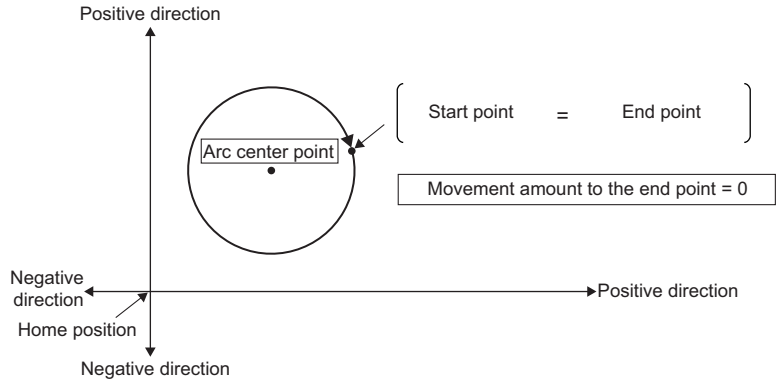
EndPoint = [Axis0001.Md.SetPosition Axis0002.Md.SetPosition]

• Relative value circular interpolation control operation with center point designation

In relative value circular interpolation control operation with center point designation, positioning is carried out from the current position at start (start point address) to the address of the movement amount set to EndPoint (end point address) in an arc path having the address of the movement amount set in AuxPoint (center point address) as its center. (The positioning speed, the movement amount to the center point, and the movement amount to the end point are shown in the following figure indicate the data to be set.)



The end point and the start point have the same address by setting "0" as the movement amount to the end point which is set to EndPoint and the positioning of a complete round with a radius from the start point address to the arc center point can be carried out. (The center point of the arc and the movement mount to the end point = 0 are shown in the following figure indicate the data to be set.)



Restriction

2-axis circular interpolation control cannot be set in the following cases.

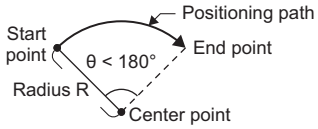
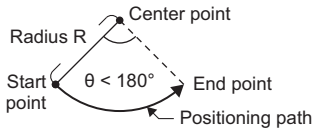
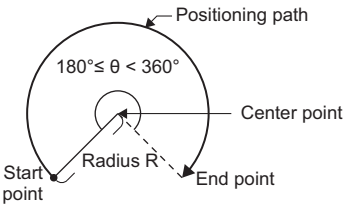
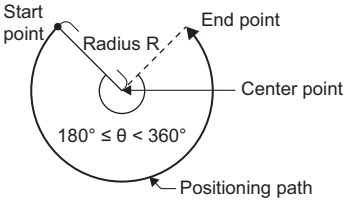
- When the radius exceeds 2147483647.0 (the maximum radius for which 2-axis circular interpolation control is possible is 2147483647.0): The error "Out of Radius Range" (error code: 1A6CH) will occur at positioning start.
- When the center point address exceeds positioning range: The error "Out of Center Point Address Range" (error code: 1A6BH) will occur.
- When the start point address is the same as the center point address: The error "Start Point - Center Point Address Same Value" (error code: 1A65H) will occur.
- When the end point address is the same as the center point address: The error "End Point - Center Point Address Same Value" (error code: 1A67H) will occur.

• Circular interpolation control with radius designation

When selecting "2: mcRadius" in the CircMode input, two configuration axes specified in CircAxes are used to carry out positioning control in an arc path having the specified radius.

[Circular interpolation path]

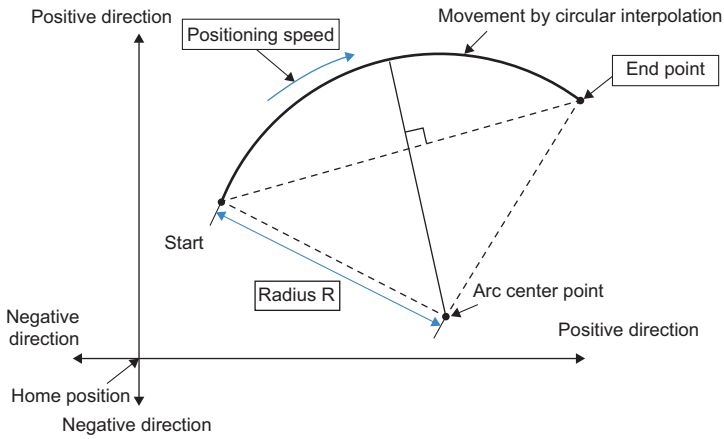
The path in the circular interpolation control with radius designation can be set by the PathChoice. The following table shows the setting value, setting description, arc center angle that can be controlled, and path of PathChoice. When setting "4: mcCWLongWay" and "5: mcCCWLongWay" , and when the start point, end point, and center point are on one straight line, a semicircle is drawn. When setting "0: mcCW" and "1: mcCCW", the error "Path Choice Setting Incorrect" (error code: 1AB7H) will occur and a semicircle can not be drawn.

Setting value	Setting description	Arc center angle that can be controlled	Path
0: mcCW	CW	0° < θ < 180° Short way	
1: mcCCW	CCW	0° < θ < 180° Short way	
4: mcCWLongWay	CW Long way	180° ≤ θ < 360° Long way	
5: mcCCWLongWay	CCW Long way	180° ≤ θ < 360° Long way	

• Absolute value circular interpolation control operation with radius designation

In absolute value circular interpolation control operation with radius designation, positioning is carried out from the current position at start (start point address) to the address set to EndPoint (end address) in an arc path having the radius set in AuxPoint.

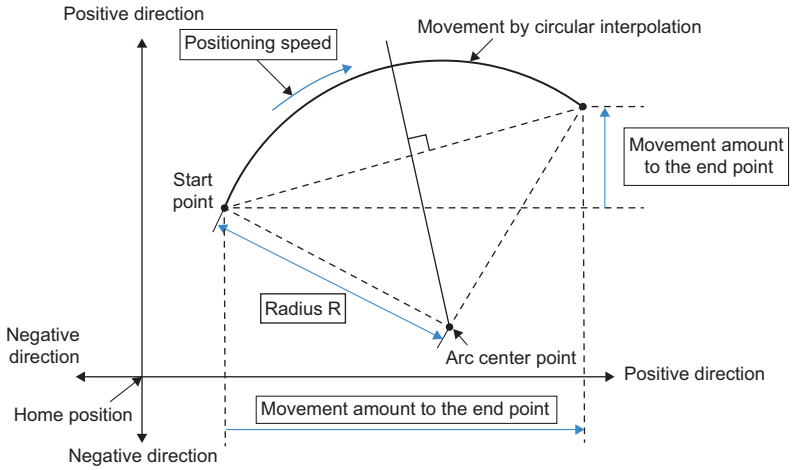
The resulting control path is an arc with the intersection of the perpendicular bisector of the current position at start and the end point address, and the specified radius as the center point. (The end point, positioning speed, and radius R shown in the following figure indicate the data to be set.)



• Relative value circular interpolation control operation with radius designation

In relative value circular interpolation control operation with radius designation, positioning is carried out from the current position at start (start point address) to the address of the movement amount set to EndPoint (end point address) in an arc path having the radius set in AuxPoint.

The resulting control path is an arc with the intersection of the perpendicular bisector of the current position at start and the end point address, and the specified radius as the center point. (The movement amount to the end point, positioning speed, and radius R shown in the following figure indicate the data to be set.)



Restriction

2-axis circular interpolation control cannot be set in the following cases.

- When the radius exceeds 2147483647.0 (the maximum arc for which 2-axis circular interpolation control is possible is radius): The error "Out of Radius Range" (error code: 1A6CH) will occur at positioning start.
- When the end point address exceeds positioning range: The error "Out of End Point Address Range" (error code: 1A6DH) will occur.
- When the start point address is the same as the end point address: The error "Start Point - End Point Address Same Value" (error code: 1A66H) will occur.
- When the distance between the start point address and the end point address is larger than the radius: Since an arc cannot be drawn, the error "Radius Setting Error" (error code: 1A6EH) will occur at positioning start.

■AuxPoint (Sub point)

AuxPoint has 16 array elements. Setting details of AuxPoint differ depending on the circular interpolation mode as shown below.

Setting value	Circular interpolation mode name	AuxPoint (sub point) input setting
0: mcBorder	Border point designation	AuxPoint[0..N] indicates the border point/center point of the configuration axis 1 to N. It is set by the absolute position or the relative position. • AuxPoint inputs for the configuration axes which are not set to the circular interpolation axis are ignored. • AuxPoint inputs after all of the configuration axes which are set to the circular interpolation axis can be omitted.
1: mcCenter	Center point designation	
2: mcRadius	Radius designation	AuxPoint[0] indicates the radius of the arc. • Inputs after AuxPoint[1] are ignored, and these can be omitted.

Ex.

When setting configuration axis 2, 3, 4 to the circular interpolation axes, and the absolute position of the sub point of the configuration axis 2, 3 is 2000, 3000 respectively.

CircAxes: [2 3]

CircMode: Border point designation

AuxPoint: [0 2000 3000 0 0 0 0 0 0 0 0 0 0 0 0 0]

or

AuxPoint: [0 2000 3000]

■EndPoint (End point)

Set the end point position of the absolute position or the movement amount from the current position at start to the end point by the EndPoint input. EndPoint has 16 array elements.

EndPoint[0..N] indicates the end point of the configuration axis 1 to N.

- EndPoint inputs for the configuration axes which are not set to the circular interpolation axis are ignored.
- EndPoint inputs after all of the configuration axes which are set to the circular interpolation axis can be omitted.

■PathChoice (Path selection)

Set rotation direction in circular interpolation. For rotation direction, the first element of the array set in CircAxes (circular interpolation axis) is regarded as the reference axis.

■Velocity (Velocity)

Specify the maximum speed of the way. For details, refer to the following.

☞ Page 243 Positioning speed in multiple axes positioning control

■Acceleration (Acceleration)

Specify the acceleration. The setting details differ depending on the acceleration/deceleration method. For details, refer to the following.

☞ Page 330 Acceleration/deceleration Processing Function

■Deceleration (Deceleration)

Specify the deceleration. The setting details differ depending on the acceleration/deceleration method. For details, refer to the following.

☞ Page 330 Acceleration/deceleration Processing Function

■Jerk (Jerk)

Specify the jerk. The setting details differ depending on the acceleration/deceleration method. For details, refer to the following.

☞ Page 330 Acceleration/deceleration Processing Function

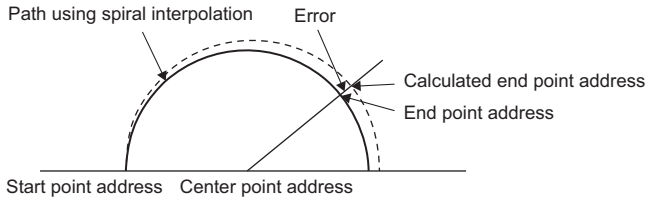
■CircularErrorTolerance (Circular interpolation error tolerance)

In circular interpolation control with center point designation, the arc path calculated from the start position and center position may deviate from the end position set in EndPoint.

CircularErrorTolerance sets allowable width of an error between the calculated arc path and the end position.

- Calculated error ≤ CircularErrorTolerance

Circular interpolation control to the set end point address is carried out while the error compensation is carried out by spiral interpolation.



- Calculated error > CircularErrorTolerance

At the positioning start, the error "Circular Interpolation Error Allowable Value Over" (error code: 1A71H) will occur and the control will not start. The machine will immediately stop if the error is detected during positioning control.

■BufferMode (Buffer mode)

Select the buffer mode. Aborting, Buffered, BlendingLow, BlendingPrevious, BlendingNext, and BlendingHigh can be set. For details, refer to the following.

☞ Page 133 Buffer mode type

■Options (Options)

Set a function option with bit specification.

Bit details and the function are as the following table. For details, refer to the following.

☞ Page 330 Acceleration/deceleration Processing Function

Bit	Function description
0 to 2	Acceleration/deceleration method setting 0: mcAccDec Acceleration/deceleration specification method 1: mcFixedTime Acceleration/deceleration time-fixed method Refer to the following. ☞ Page 330 Acceleration/deceleration Processing Function
3*1	Position selection during buffer mode Specify "0" for absolute value circular interpolation control.*2 0: Set position 1: Actual position For details, refer to the following. ☞ Page 132 Multiple Start (Buffer Mode)
4 to 31	Empty (Specify "0".)*2

*1 It is valid only when relative value circular interpolation controls.

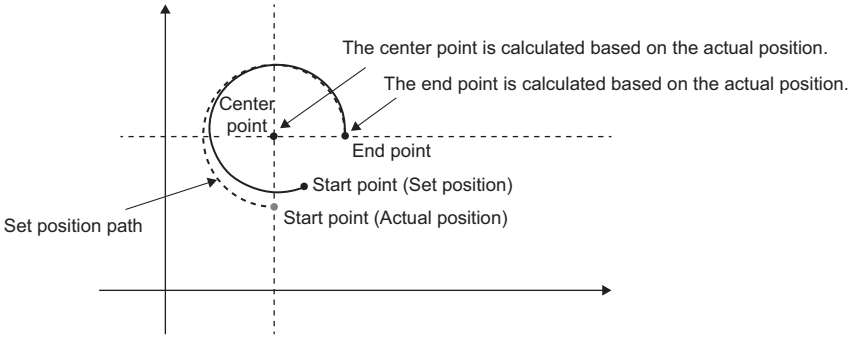
*2 If specifying other than "0", the error "Out of Options Range" (error code: 1A4EH) occurs, and the function does not start.

When "1: Actual position" is selected at position selection during buffer mode in relative value circular interpolation control, the sub point (center point, border point) and the end point is calculated based on the actual position and the set position is used as the start point. The operation of each CircMode (circular interpolation mode) is as follows.

[For the circular interpolation control in the center point designation]

The center point and the end point are calculated based on the actual position.

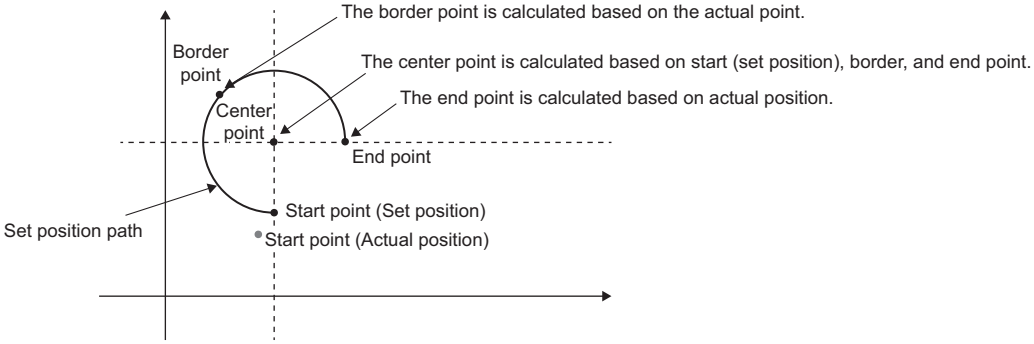
The circular path may be ellipse.



[For the circular interpolation control in the border point designation]

The border point and the end point are calculated based on the actual point.

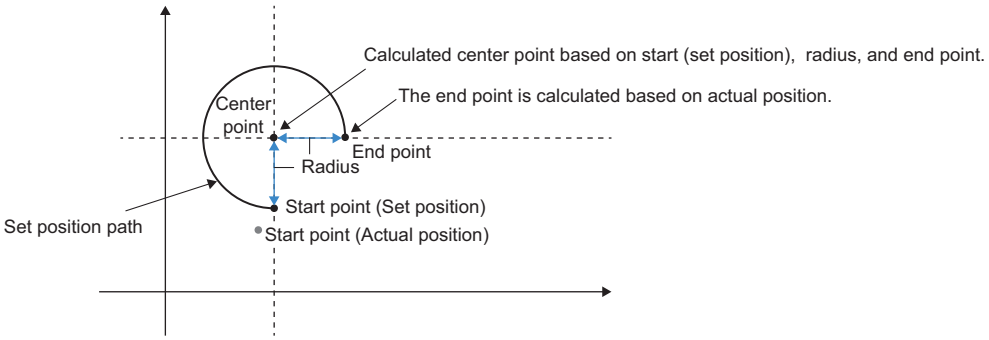
The center point is determined by the start point (set position), the border point, and the end point.



[For the circular interpolation control in the radius designation]

The end point is calculated based on the actual point.

The center point is determined by the start point (set position), the radius, and the end point.



Precautions

- Circular interpolation including an axis whose stroke limit is set to invalid cannot be carried out. The error "Software Stroke Limit Invalid at Circular Interpolation" (error code: 1A72H) will occur, and the operation does not start.
- When the interpolation path exceeds the stroke limit range during interpolation operation, the error "Software Stroke Limit Over (Forward Direction)" (error code: 1A03H) or the error "Software Stroke Limit Over (Reverse Direction)" (error code: 1A04H) will occur, and the operation will decelerate to a stop.

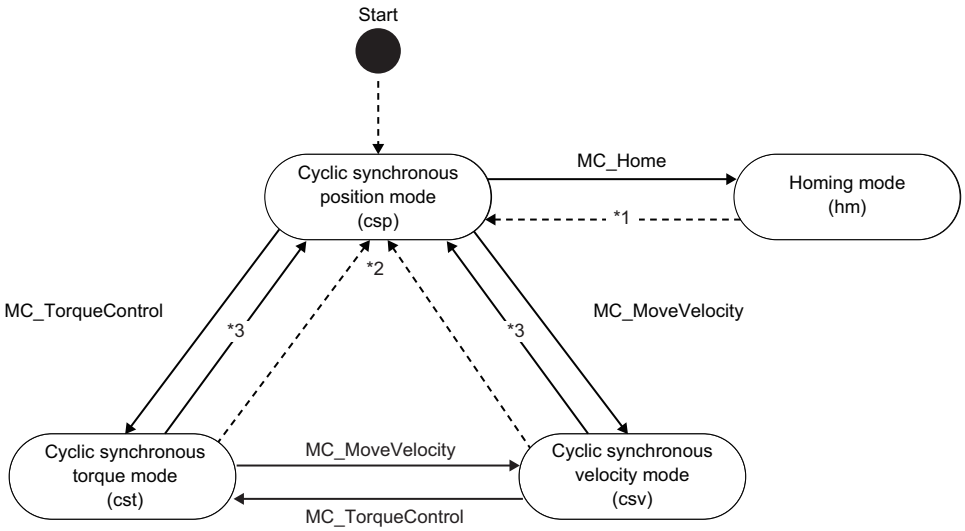
7 DIRECT CONTROL

Direct control includes velocity control and torque control. The operation details of each control is as shown below.

Control Name	Driver control mode	Applicable instruction	Operation details
Velocity control	Cyclic synchronous position mode (csp)	MCv_SpeedControl	This function outputs the commanded position based on the set velocity for each control cycle.
	Cyclic synchronous velocity mode (csv)	MC_MoveVelocity	This function outputs the set velocity for each control cycle. (Excluding the position loop.)
Torque control	Cyclic synchronous torque mode (cst)	MC_TorqueControl	This function outputs the commanded torque for each control cycle. (Excluding the position loop.)

The driver side control mode must be csp at connection.

Switching the driver control mode is simultaneously executed when the MCFB is executed. The following shows the state transition.



*1 Transits after the axis stops by homing completion or error occurrence.

*2 Transits at stop completion or error occurrence.

*3 Transits when Aborting or Buffered is executed to an instruction other than MC_MoveVelocity and MC_TorqueControl.

Point

Switch to the continuous operation to torque control mode for the usage such as pressing a workpiece. Using the continuous operation to torque control mode can switch the operation smoothly from the cyclic synchronous position mode or the cyclic synchronous velocity mode to the pressing operation. Therefore, reducing load to a machine and molding high in quality are provided as the velocity and torque do not change abruptly.

For details, refer to the following.

☞ Page 744 Relevant functions

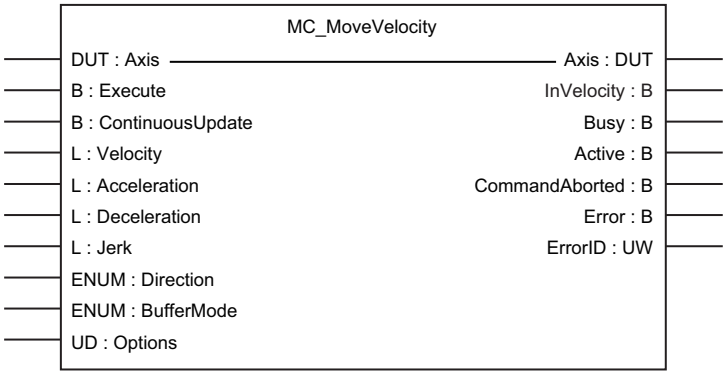
7.1 Velocity Control

In velocity control, the driver control mode is switched to csv, and the control excluding the position loop is executed.

Relevant variables/FB

Variable/Structure name	Name	Details
AxisName.Md.		
SetVelocity	Set velocity	The velocity calculated from the difference of the set position updated by the follow up.
SetAcceleration	Set acceleration	A value calculated from the difference of the set velocity
TargetVelocity	Target velocity	A value set in target velocity (Velocity)
ActualVelocity	Actual velocity	Actual velocity (same as positioning control)
Io_TargetVelocity	Object data_TargetVelocity	The velocity command to send to slave devices.
Io_VelActualValue	Object data_VelActualValue	Feedback of the velocity which is received from slave devices.

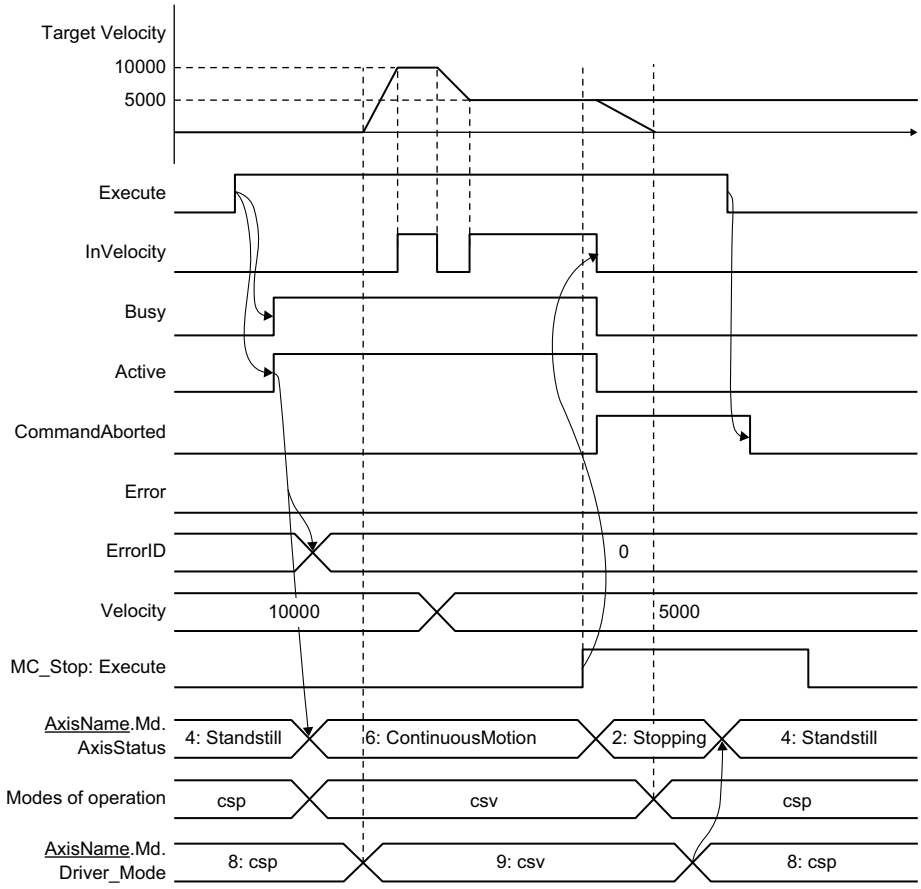
MC_MoveVelocity

Item	Description
Function overview	The driver is switched to the csv, and the velocity control is executed based on the specified target velocity.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
56	8	Subroutine type	Real-time execution

■Timing chart

[When the FB is normally completed]



[When an error occurs]

For details when an error occurs, refer to the following.

☞ Page 27 Execute type/Enable type

■I/O variable

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Axis information	Axis	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables

■Input variables

Import ↑: At start, R: Retrigger possible, C: Continuous update possible

Name	Variable name	Data type	Import	Setting range	Default value	Description
Execute command	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Continuous update	ContinuousUpdate	BOOL	↑	TRUE, FALSE	FALSE	During TRUE, it can continuously change the target velocity, the acceleration, and the deceleration.
Target velocity	Velocity	LREAL	↑/R/C	☞ Page 87 Velocity set range	0.0	Sets the target velocity. When the velocity is negative, the axis moves to the negative direction. When 0.0 is set, the axis does not operate, however, the axis status (AxisName.Md.AxisStatus) changes to "6: ContinuousMotion".
Acceleration	Acceleration	LREAL	↑/R/C	☞ Page 335 Input variables to use	0.0	Sets the acceleration.
Deceleration	Deceleration	LREAL	↑/R/C	☞ Page 335 Input variables to use	0.0	Sets the deceleration.
Jerk	Jerk	LREAL	↑	☞ Page 335 Input variables to use	0.0	Sets the jerk.
Direction selection	Direction	MC_DIRECTION	↑	1 to 2	0	Sets the direction selection. MC_DIRECTION definition can be used. 1: mcPositiveDirection (Positive direction) 2: mcNegativeDirection (Negative direction) When selecting "2: mcNegativeDirection" and Velocity is negative, the movement direction of the motor will be positive.
Buffer mode	BufferMode	MC_BUFFER_MODE	↑	0 to 1	0	Selects the buffer mode. 0: mcAborting (Aborting) 1: mcBuffered (Buffered)
Options	Options	DWORD(HEX)	↑	*1	00000000H	Sets a function option with bit specification.

*1 For a bit and the function description, refer to the following table.

Bit	Function description
0 to 2	Acceleration/deceleration method setting 0: mcAccDec Acceleration/deceleration specification method 1: mcFixedTime Acceleration/deceleration time-fixed method For details, refer to the following. ☞ Page 330 Acceleration/deceleration method
3 to 15	Empty (Specify "0".)*1
16, 17	Velocity initial value selection 0: Set velocity 1: Actual velocity 2: Automatic selection
18 to 31	Empty (Specify "0".)*1

*1 If specifying other than "0", the error "Out of Options Range" (error code: 1A4EH) occurs, and the function does not start.

■Output variables

Name	Variable name	Data type	Default value	Description
Target velocity reached	InVelocity	BOOL	FALSE	Indicates that the set velocity that is calculated by the motion system has reached the target velocity. After reaching, the speed is maintained until Execute turns FALSE or the control is interrupted. When the target velocity has been changed due to change at ContinuousUpdate TRUE, it turns FALSE until the set velocity reaches changed target velocity.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Controlling	Active	BOOL	FALSE	Indicates that the FB is controlling the axis.
Abortion of execution	CommandAborted	BOOL	FALSE	Indicates that another FB has aborted the execution of the FB. Turns TRUE when this FB is stopped by an error or this FB is started during an error occurence. Turns FALSE when Execute is FALSE.
Error	Error	BOOL	FALSE	When TRUE, it indicates that an error has occurred in the FB.
Error code	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned. For details, refer to the following. ☞ Page 662 List of Error Codes

Control details

MC_MoveVelocity switches the driver control mode to the csv (cyclic synchronous velocity mode) and executes control. This function controls the set velocity based on the specified acceleration, deceleration, and jerk. To finish this FB, start MC_Stop.

Velocity initial value selection

Set the "velocity initial value selection (options bit16, 17)" when switching the following control mode.

- The csp (cyclic synchronous position mode)→The csv (cyclic synchronous velocity mode)

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	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 Velocity actual value is not mapped, motor speed received from the drive unit at switching is not used. (Speed initial value selection becomes "0".)

*2 When Velocity actual value is not mapped, velocity commanded to the drive unit immediately after switching becomes "0: Set velocity".

Operation at stop cause occurrence

The following shows stop causes during the csv (cyclic synchronous velocity mode) and the process when each cause occurs.

Cause No.	Stop cause	Axis status (AxisName.Md.AxisStatus)		Stop process
		During deceleration stop	After stop	
1	"Forced stop input" is FALSE	—	1: ErrorStop	Immediate stop*1
2	Drive unit power supply is FALSE	—	1: ErrorStop	Immediate stop*1
	Drive unit network disconnection detection			
	Drive unit			
	Forced stop input to the drive unit			
	Enable of MC_Power is FALSE			
	ServoON of MC_Power is FALSE (When AxisName.Pr.StopMode_ServoOff is "4:ServoOffAfterImmediateStop")*2			
3	Hardware stroke limit upper/lower limit error occurrence	1: ErrorStop	1: ErrorStop	Deceleration stop/Immediate stop (Follows AxisName.Pr.StopMode_HwStrokeLimit) After stopping, switch to the csp.
4	CPU module error occurrence	1: ErrorStop	1: ErrorStop	Deceleration stop/Immediate stop (Follows System.Pr.StopMode_All) After stopping, switch to the csp.
	PLC READY [Y0] is OFF			
5	Software stroke limit upper/lower limit error occurrence	1: ErrorStop	1: ErrorStop	Deceleration stop/Immediate stop (Follows AxisName.Pr.StopMode_SwStrokeLimit) After stopping, switch to the csp.
6	Axis error detection*3	1: ErrorStop	1: ErrorStop	Deceleration stop/Immediate stop (Follows AxisName.Pr.StopMode_General) After stopping, switch to the csp.
7	Execute of MC_Stop is TRUE	2: Stopping	2: Stopping*4	Deceleration stop (Follows the deceleration speed set in the FB) After stopping, switch to the csp.
8	"Stop signal" of external input signal is TRUE	No change	4: Standstill	Deceleration stop/Immediate stop (Follows AxisName.Pr.StopMode_General) After stopping, switch to the csp.

*1 The servo turns OFF in the driver side and immediate stop is performed, and also the motion side command is stopped.

*2 Operation differs depending on the setting value selected in AxisName.Pr.StopMode_ServoOff. For details, refer to the following.
☞ Page 163 Relevant variables/FB

*3 If an axis error (the error that transits the state into the "1: ErrorStop" status) occurs in the FB which can be linked by the buffer mode, deceleration stop will be performed from the error occurrence.

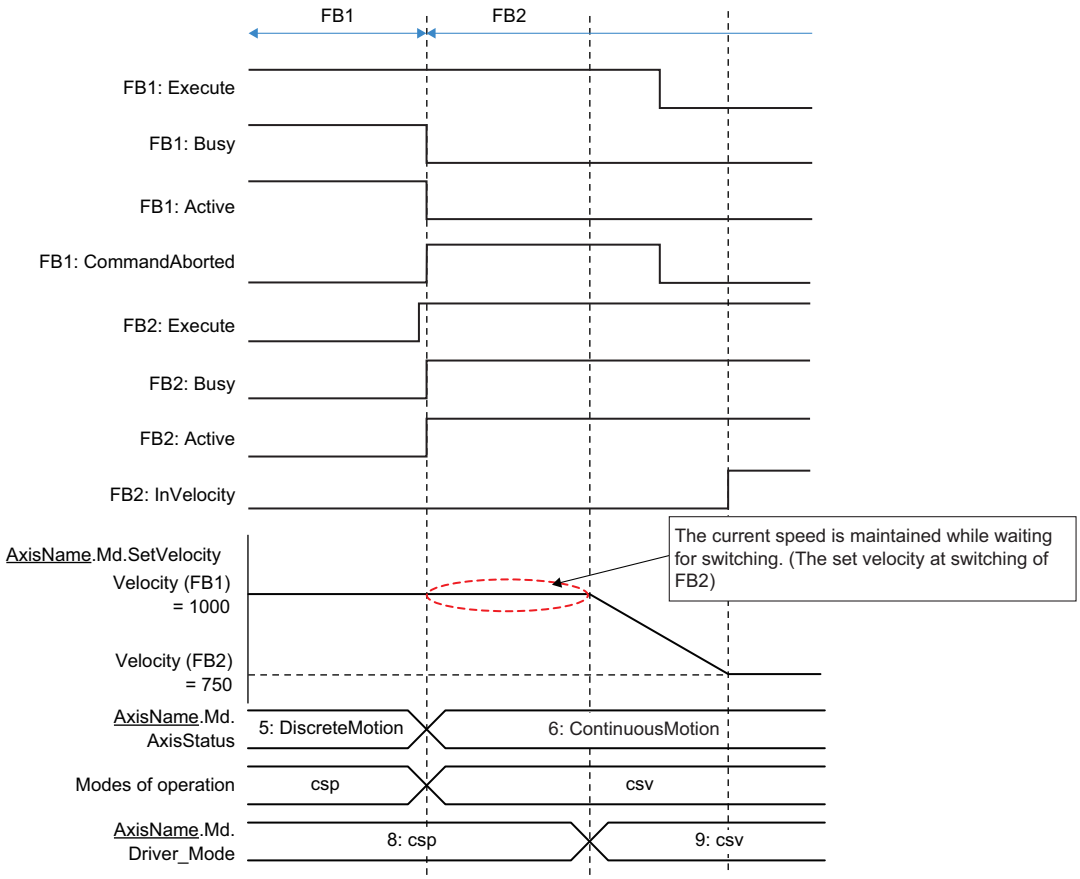
*4 When Execute of MC_Stop is FALSE at stop completion, the axis status (AxisName.Md.AxisStatus) after stop will be "4: Standstill".

When starting this FB during another instruction execution

■Single axis position control (FB1) → Velocity control (FB2)

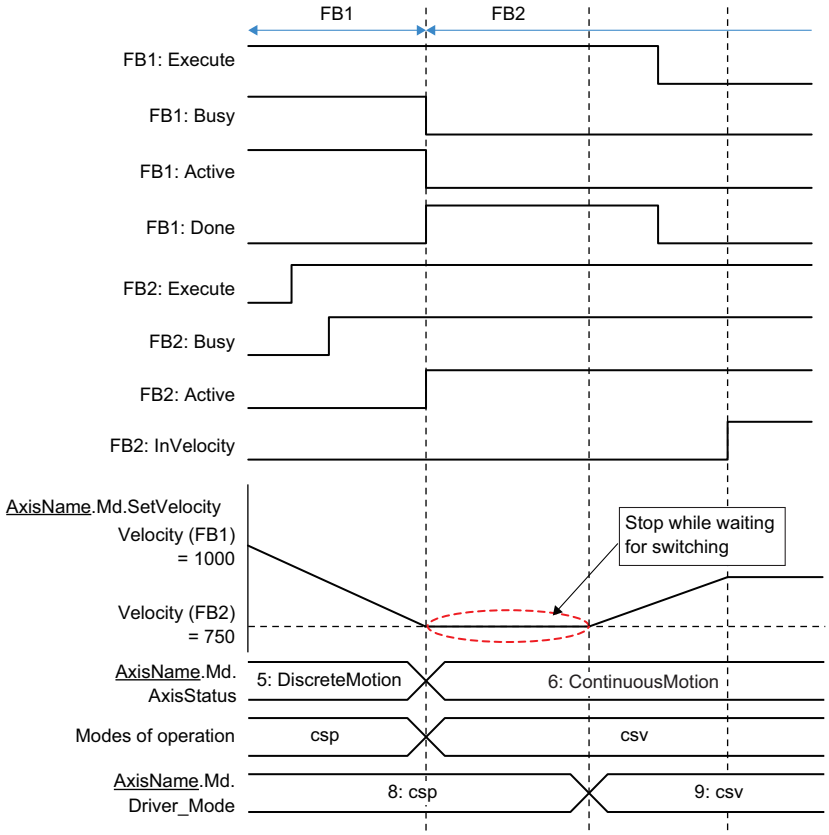
- 0: mcAborting

1. Input variables are analyzed, and a switching request to csv is issued to the driver.
2. Until the driver is switched, the axis is operated at the previous velocity with the velocity control including the position loop.
3. When the driver switches to csv, TargetVelocity is obtained based on the actual velocity and the acceleration/deceleration process is started again.



• 1: mcBuffered

1. Input variables are analyzed, and the axis waits until the previous instruction is completed.
2. When the previous instruction is completed, a switching request to csv is issued to the driver.
3. When the driver switches to csv, the acceleration/deceleration process is started based on the settings of FB2.



When starting another instruction during this FB execution

While this FB is executed, only Aborting and Buffered are supported.

■Velocity control → Operation at positioning start

1. A switching request to csp is issued to the driver.
2. Started positioning control FB is analyzed, and the axis waits until the driver switches to csp. (Even if the axis issues the switching request to csp during operation, switching may not be carried out depending on the driver specifications. If switching to csp is not completed within 1 second, the error "Control Mode Switching Error" (error code: 1A1DH) occurs for the timeout, and the axis stops.)
3. While waiting, velocity control is continued with the set velocity when Execute of FB2 is TRUE.
4. When the driver switches to csp, the positioning control starts.

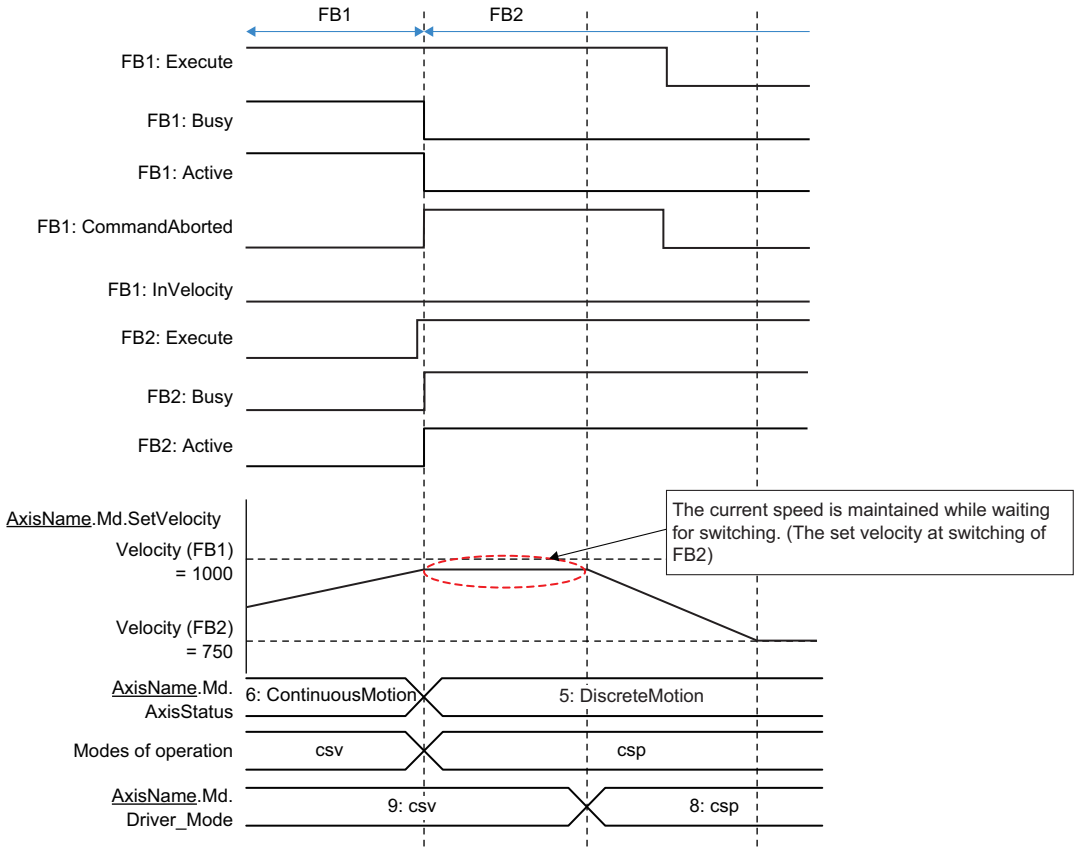
Point

The deceleration distance may not be secured and the target position may be overrun depending on the travel distance switching from the csv to the csp. In this case, the axis operates based on AxisName.Pr.OverrunOperation.
For operation at the overrun occurrence, refer to the following.
☞ Page 132 Multiple Start (Buffer Mode)

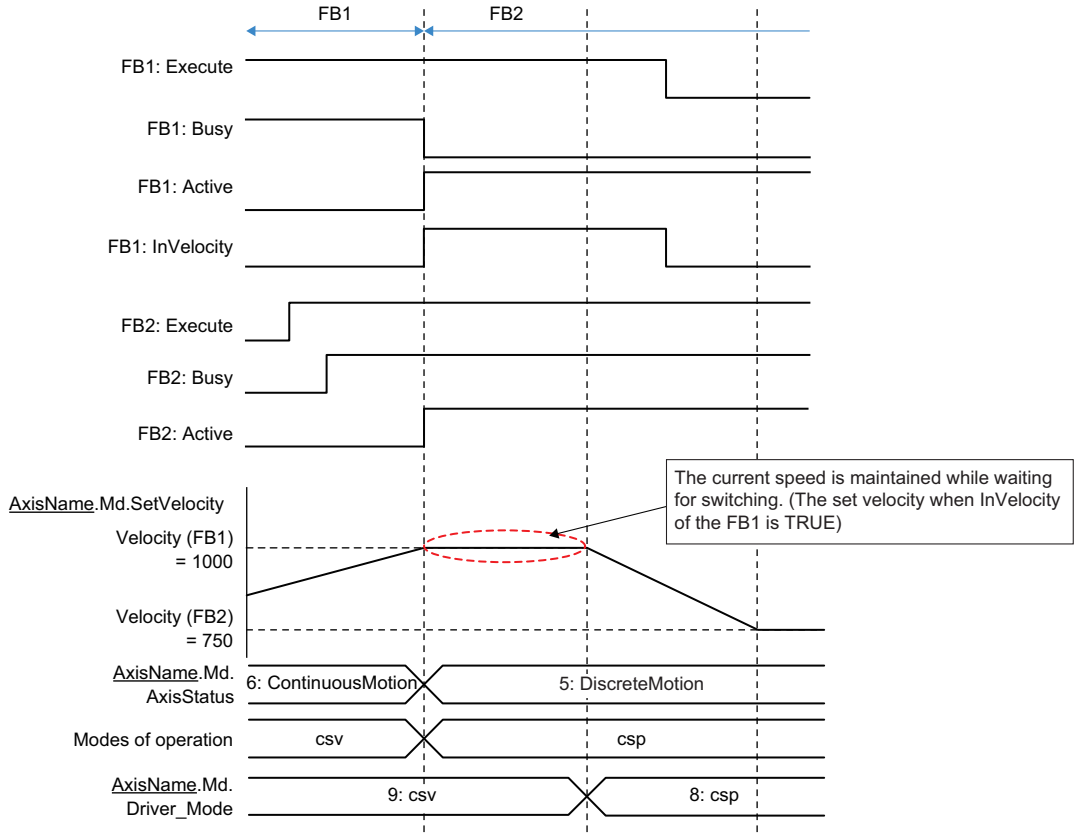
Precautions

If the axis overruns while waiting for switching, it will stop immediately after switching to csp.

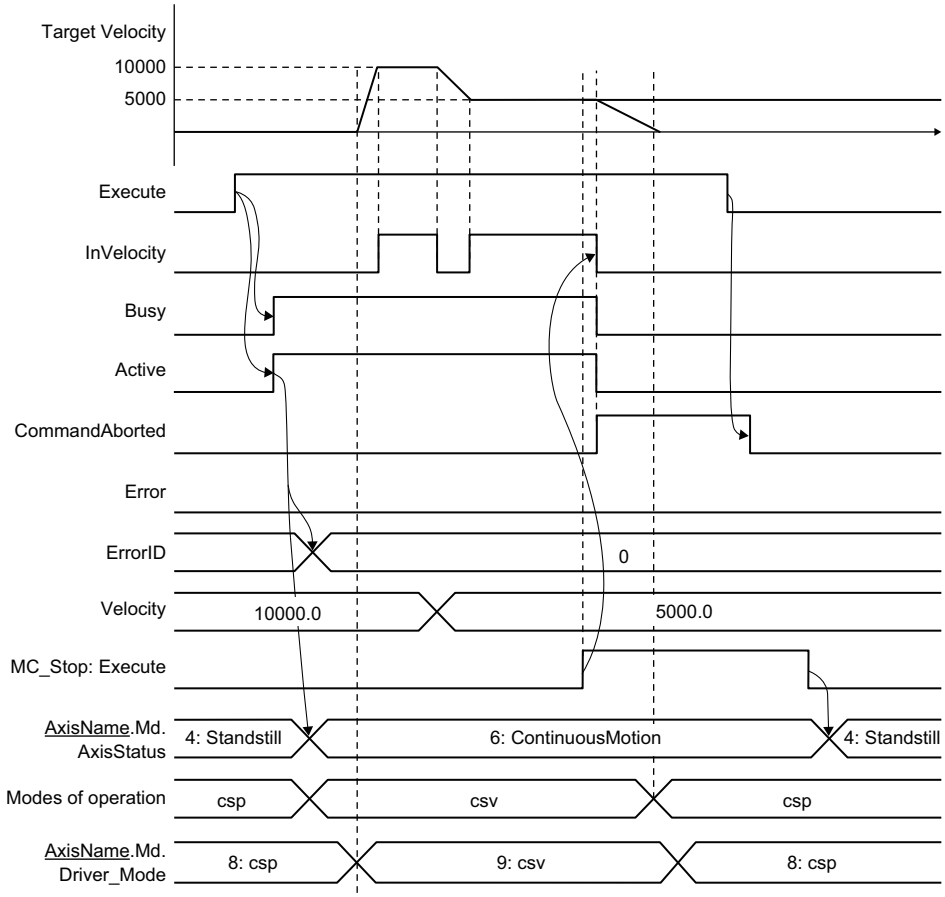
[For Aborting]



[For Buffered]



■Velocity control → Stop instruction (MC_Stop)

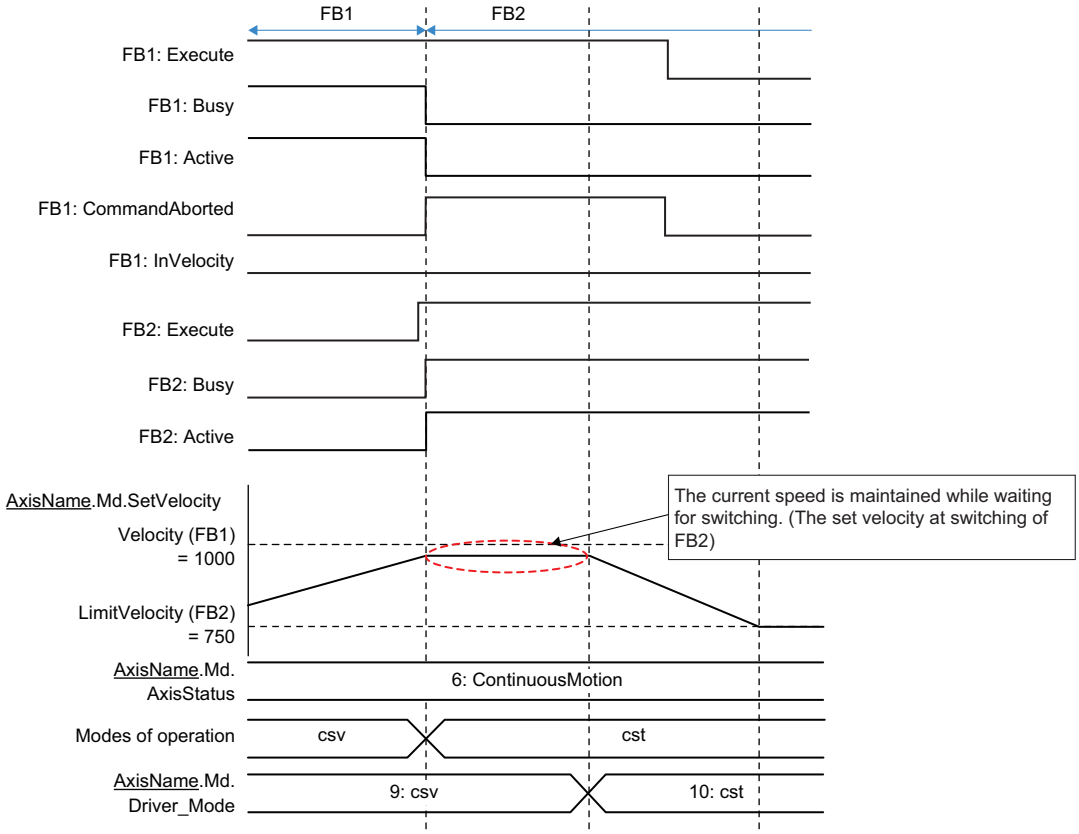


■Velocity control → Torque control

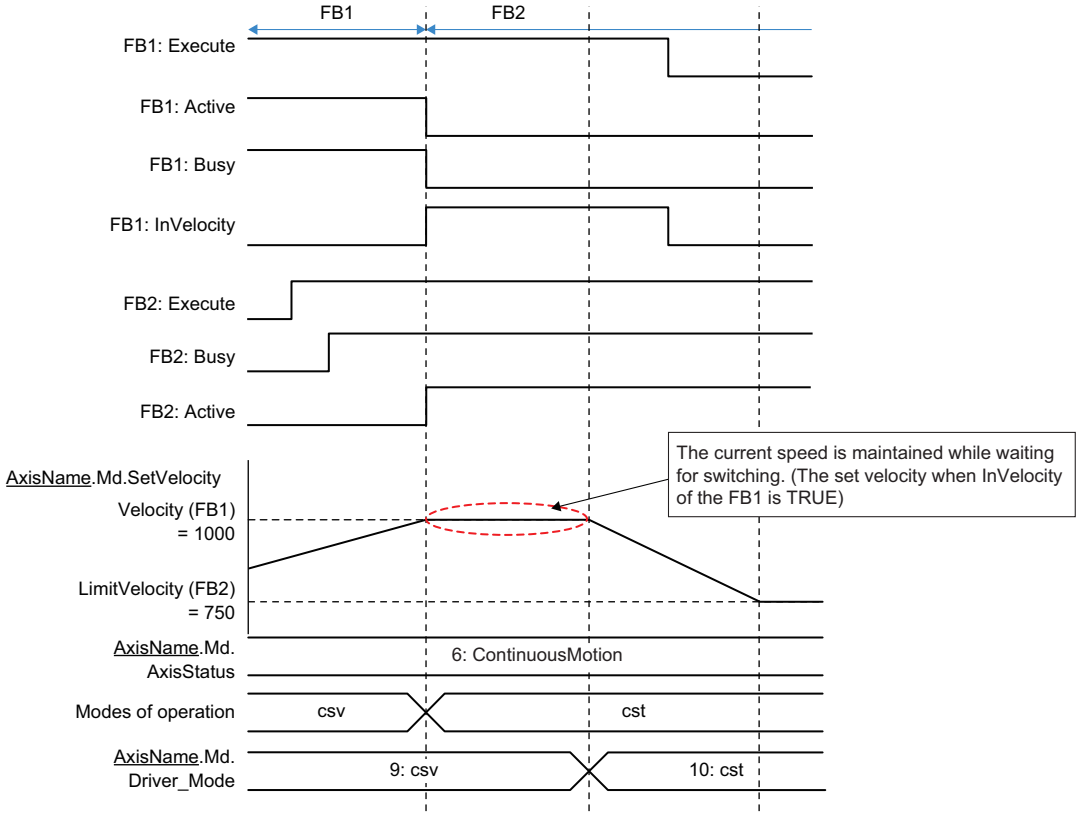
1. Switching request to the cst is executed to the driver.

2. Started torque control is analyzed, and the axis waits until the driver switches to the cst.

[For Aborting]



[For Buffered]



When the operation direction is reversed

When starting this FB and operation direction is reversed during another instruction execution, a deceleration stop is performed once. After the deceleration stop is completed, operation is started to the changed direction.

Precautions

- AxisName.Cd.VelocityOverride and AxisName.Cd.AccelerationOverride are valid.
- The set position and the feed machine position are updated by follow up.
- Switching time to 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 the positioning control FB during the control mode switching. Start the positioning control FB after confirming that AxisName.Md.Driver_Mode is switched to "9: csv".

When using the MR-J5(W)-G

When switching from the csp to the csv or from the csv to the csp without waiting for stop of the motor, note the following.

- Set "ZSP disabled selection at control switching" (PC76.1) of the servo parameter (extension setting) to "1: Disabled" and disable monitoring of zero speed status. However, note that it may cause vibration or impact at control mode switching.
- For the setting value of the servo parameter "Electronic gear numerator" (PA06) and "Electronic gear denominator" (PA07), refer to the following.

☞ Page 731 Setting methods

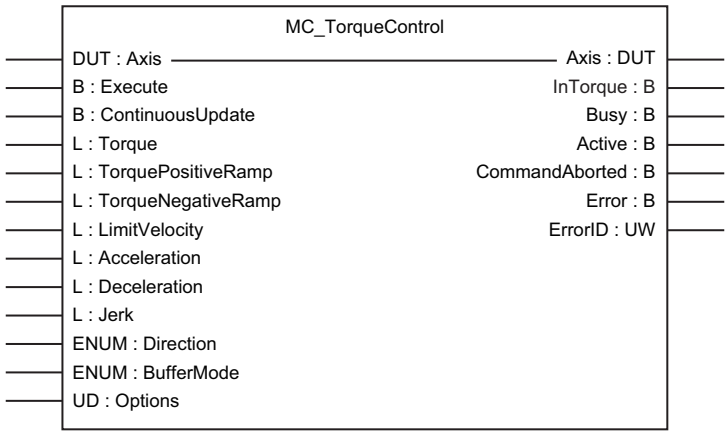
7.2 Torque Control

In torque control, the driver control mode is switched to cst and the control is carried out.

Relevant variables/FB

Variable/Structure name	Name	details
AxisName.Md		
SetVelocity	Set velocity	The velocity calculated from the difference of the set position updated by the follow up.
SetAcceleration	Set acceleration	A value calculated from the difference of the set velocity
TargetVelocity	Target velocity	Stores value set to the limit velocity (LimitVelocity)
ActualVelocity	Actual velocity	Actural velecity (same as positioning control)
Cst_SetTorque	Set torque at torque control	Stores set torque at the cyclic synchronous torque mode
Cst_TargetTorque	Target torque at torque control	Stores target torque at the cyclic synchronous torque mode
Io_TargetVelocity	Object data_TargetVelocity	The velocity command to send to slave devices.
Io_VelActualValue	Object data_VelActualValue	Feedback of the velocity which is received from slave devices.

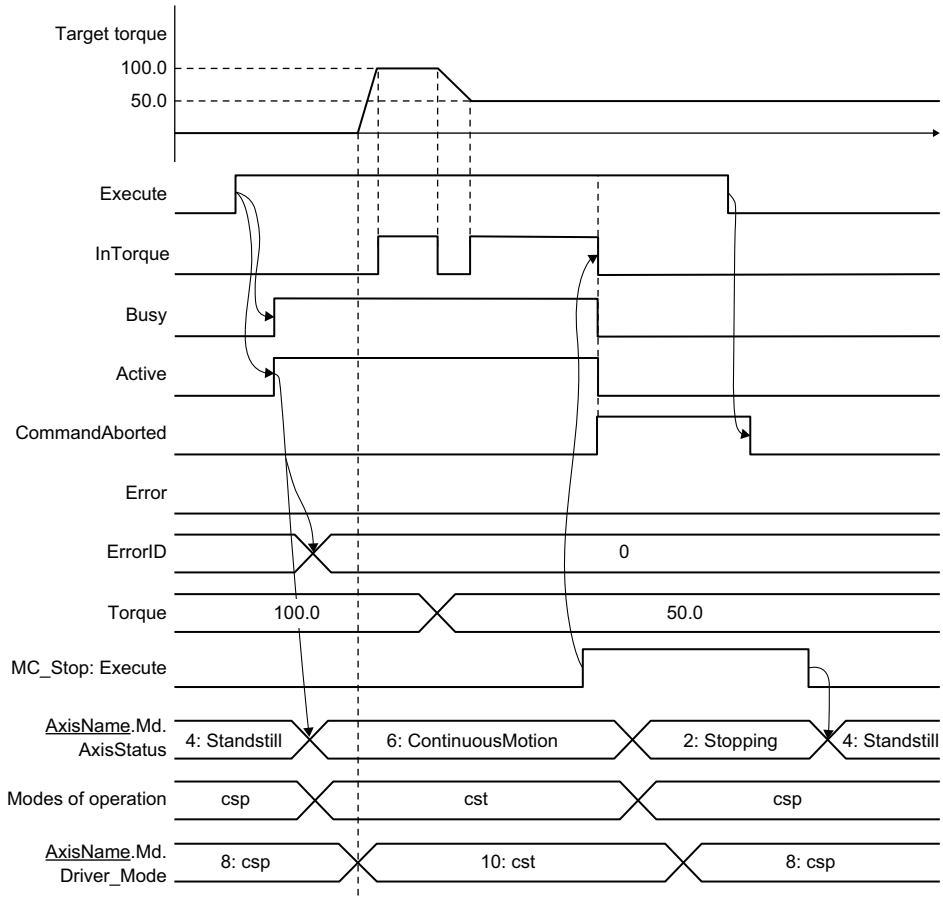
MC_TorqueControl

Item	Description
Function overview	The driver is switched to the cst, and the torque control is executed based on the specified target torque.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
80	8	Subroutine type	Real-time execution

■Timing chart

[When the FB is normally completed]



[When an error occurs]

For details when an error occurs, refer to the following.

☞ Page 27 Execute type/Enable type

■I/O variable

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Axis information	Axis	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables

■Input variables

Import ↑: At start, R: Retrigger possible, C: Continuous update possible

Name	Variable name	Data type	Import	Setting range	Default value	Description
Execute command	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Continuous Update	ContinuousUpdate	BOOL	↑	TRUE, FALSE	FALSE	During TRUE, it can continuously change the target torque, the limit velocity, the torque positive direction ramp, the torque negative direction ramp, the acceleration and the deceleration.
Target torque	Torque	LREAL	↑/R/C	-1000.0 to 1000.0 [%]	0.0	Sets the target torque. Sets the ratio to the rated torque of the servo motor used in percentage. 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 target torque in the positive direction (address increasing direction), set the positive value. To output the target torque in the negative direction (address decreasing direction), set a negative value.
Torque positive direction ramp	TorquePositiveRamp	LREAL	↑/R/C	☞ Page 291 Control details	0.0	☞ Page 291 Control details
Torque negative direction ramp	TorqueNegativeRamp	LREAL	↑/R/C		0.0	
Limit velocity	LimitVelocity	LREAL	↑/R/C	☞ Page 356 Speed Limit	0.0	Sets the velocity limit value at the cyclic synchronous torque mode. For the velocity limit value, refer to the driver manual. [For using MR-J5(W)-G] Uses Velocity limit value as velocity limit value. When the Velocity limit value is not mapped, the velocity command of the controller will not be enabled. (The servo parameter "Speed limit" (PT67) set value is enabled.)
Acceleration	Acceleration	LREAL	↑/R/C	☞ Page 335 Input variables to use	0.0	Sets the acceleration and acceleration/deceleration time until the limit velocity is reached. For details, refer to the following. ☞ Page 330 Acceleration/deceleration Processing Function
Deceleration	Deceleration	LREAL	↑/R/C	☞ Page 335 Input variables to use	0.0	Sets the deceleration until the limit velocity is reached. For details, refer to the following. ☞ Page 330 Acceleration/deceleration Processing Function
Jerk	Jerk	LREAL	↑	☞ Page 335 Input variables to use	0.0	Sets the jerk until the limit velocity is reached. For details, refer to the following. ☞ Page 330 Acceleration/deceleration Processing Function When 0 is set or the setting is omitted, the jerk is not applied.
Direction selection	Direction	MC_DIRECTION	↑	0	0	Set 0. (If a value other than "0" is set, the error "Out of Direction Selection Range" (error code: 1A37H) will occur.)
Buffer mode	BufferMode	MC_BUFFER_MODE	↑	0 to 1	0	Selects the buffer mode. 0: mcAborting (Aborting) 1: mcBuffered (Buffered) For "0: mcAborting", the control mode is immediately switched to the torque control. For "1: mcBuffered", the control mode is switched to the torque control after the previous FB is completed. If the previous FB is MC_TorqueControl, the control mode is switched when InTorque is TRUE.
Options	Options	DWORD(HEX)	↑	*1	00000000H	Sets a function option with bit specification.

*1 For a bit and the function description, refer to the following table.

Bit	Function description
0 to 2	Acceleration/deceleration method setting 0: mcAccDec (Acceleration/deceleration specification method) 1: mcFixedTime (Acceleration/deceleration time-fixed method) For details, refer to the following. ☞ Page 330 Acceleration/deceleration Processing Function
3 to 15	Empty (Specify "0".)*1
16, 17	Torque ramp fuction selection 0: Ramp method 1: Time constant method 2: Fixed time method
18	Torque initial value selection 0: Target torque 1: Actual torque
19	Control mode switching selection*2 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 0: Set velocity 1: Actual velocity 2: Automatic selection For details, refer to the following. ☞ Page 280 Velocity initial value selection
22 to 31	Empty (Specify 0.)*1

*1 If specifying other than "0", the error "Out of Options Range" (error code: 1A4EH) occurs and the function does not start.

*2 When bit20 of Supported drive modes [Obj. 6502h] is TRUE, the mode can be switched to the continuous operation to torque control mode. If "1: ct (continuous operation to torque control mode)" is set when it is FALSE, the error "Driver Control Mode Unsupported" (error code: 1AE9H) occurs.
For details, refer to the following.
☞ Page 745 Continuous operation to torque control mode

*3 It is valid only when "1: ct (continuous operation to torque control mode)" is set for control mode switching selection (Options bit19).

■Output variables

Name	Variable name	Data type	Default value	Description
Target torque reached	InTorque	BOOL	FALSE	Indicates that the target torque has reached the specified value. After the reach, it maintains until Execute turns to FALSE or the control is aborted. If the target torque is changed due to the change etc. when ContinuousUpdate is TRUE, it is FALSE until it reaches to the changed target torque.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Controlling	Active	BOOL	FALSE	Indicates that the FB is controlling the axis.
Abortion of execution	CommandAborted	BOOL	FALSE	Indicates that another FB has aborted the execution of the FB. Turns TRUE when this FB is stopped by an error or this FB is started during an error occurence. Turns FALSE when Execute is FALSE.
Error	Error	BOOL	FALSE	When TRUE, it indicates that an error has occurred in the FB.
Error code	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned. For details, refer to the following. ☞ Page 662 List of Error Codes

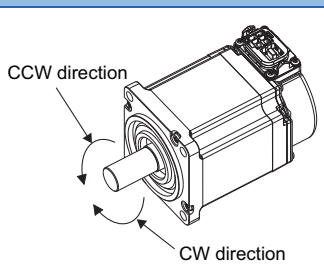
Control details

MC_TorqueControl switches the driver control mode to the cst (cyclic synchronous torque mode) and executes control. This function controls the set torque based on the specified TorquePositiveRamp and TorqueNegativeRamp. To finish this FB, start MC_Stop.

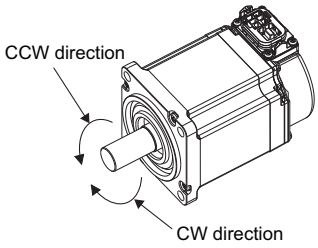
Relation between torque generation direction of servomotor and set torque

When using the MR-J5(W)-G, it differs based on the settings of the servo parameter "Movement direction selection" (PA14) and "Torque POL reflection selection" (PC29.3).

- When the servo parameter (extension setting) "Torque POL reflection selection" (PC29.3) is set to "0: Enabled"

Setting value of "Movement direction selection" (PA14)	Set torque	Torque generation direction of servomotor	
0: Forward rotation (CCW) with the increase of the positioning address	Positive value (positive direction)	CCW direction	
	Negative value (negative direction)	CW direction	
1: Reverse rotation (CW) with the increase of the positioning address	Positive value (positive direction)	CW direction	
	Negative value (negative direction)	CCW direction	

- When the servo parameter (extension setting) "Torque POL reflection selection" (PC29.3) is set to "1: Disabled" (Initial value)

Setting value of "Movement direction selection" (PA14)	Set torque	Torque generation direction of servomotor	
0: Forward rotation (CCW) with the increase of the positioning address	Positive value (positive direction)	CCW direction	
	Negative value (negative direction)	CW direction	
1: Reverse rotation (CW) with the increase of the positioning address	Positive value (positive direction)	CCW direction	
	Negative value (negative direction)	CW direction	

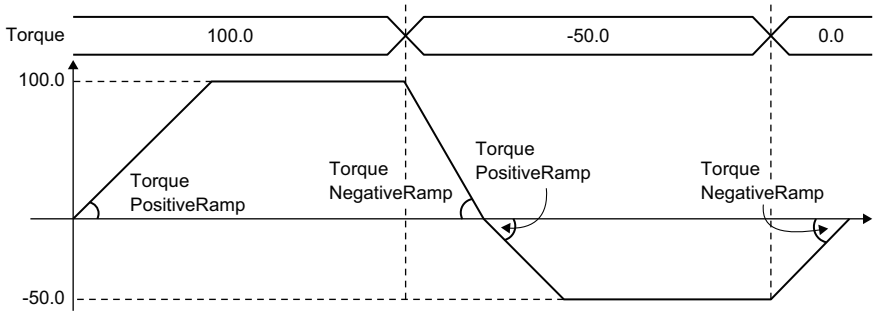
Torque ramp function selection (Options bit16, 17)

TorquePositiveRamp/TorqueNegativeRamp differs depending on the setting values.

- For "0: Ramp method"

Specify the ramp from the current set torque until the target torque is reached.

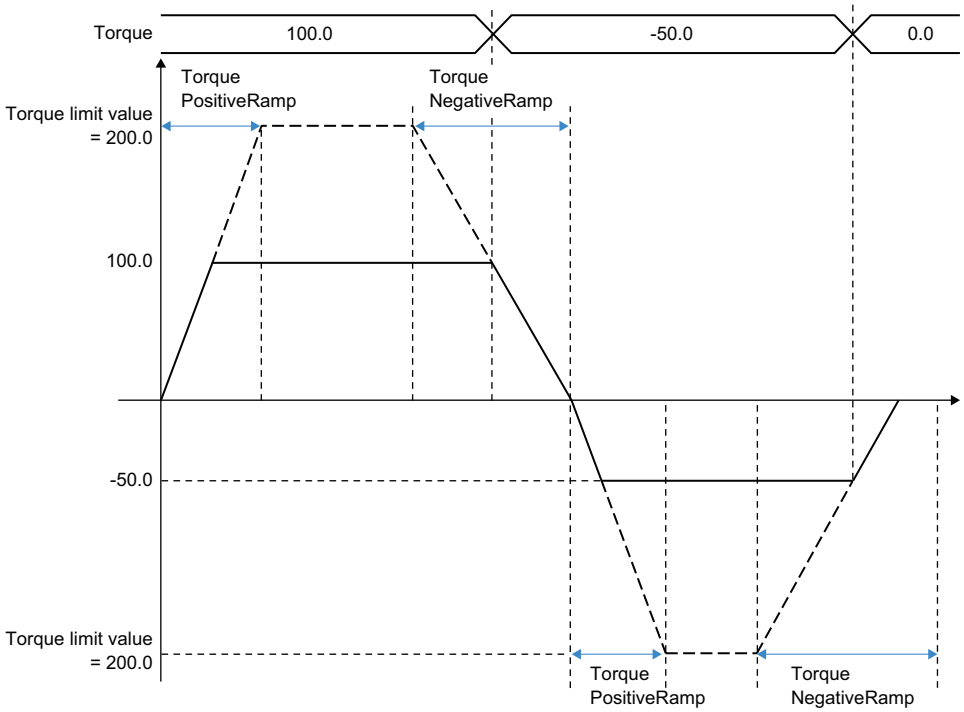
Setting range: 0 to 1000.0 [%/s]



• For "1: Time constant method"

Set the time that the set torque reaches from 0 until the positive/negative torque limit value.

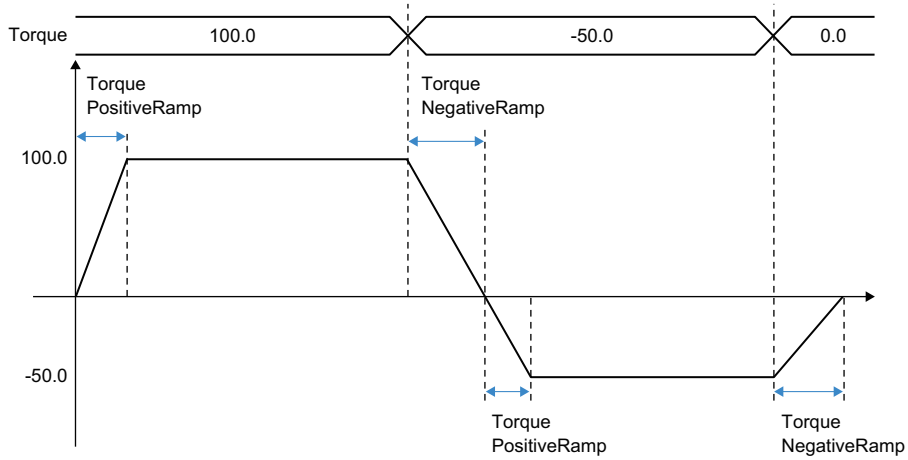
Setting range: 0 to 8400.0 [s]



• For "2: Fixed time method"

Set the time from the current set torque until the target torque is reached.

Setting range: 0 to 8400.0 [s]



When the output torque direction is changed due to the target torque change, the set torque becomes 0 based on the setting value of the torque negative direction ramp. After that, it becomes the target torque based on the setting value of the torque positive direction.

When the torque positive direction ramp and the torque negative direction ramp are specified to 0, the value reaches the target torque in 1 operation cycle.

Torque initial value selection (Options bit18)

Set the torque initial value when switching to the cst (cyclic synchronous torque mode).

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 and torque negative direction ramp immediately after the control mode switching.
1: Actual torque	Torque actual value at switching is set as the set torque.

Point

Normally set "0: Target torque". Set "1: Actual torque" only when switching the control mode without waiting for the servo motor to stop immediately after the command to the motor is completed.

Operation at stop cause occurrence

The following shows stop causes during the cst (cyclic synchronous torque mode) and the process when each cause occurs.

Cause No.	Stop cause	Axis status (AxisName.Md.AxisStatus)	Stop process
		After stop	
1	"Forced stop input" is FALSE	1: ErrorStop	Immediate stop[*1]
2	Drive unit power supply is FALSE	1: ErrorStop	Immediate stop[*1]
	Drive unit network disconnection detection		
	Drive unit		
	Forced stop input to the drive unit		
	Enable of MC_Power is FALSE		
	ServoON of MC_Power is FALSE (When AxisName.Pr.StopMode_ServoOff is "4:ServoOffAfterImmediateStop", "5: ServoOffAfterDecelStop")[*2]		
3	Hardware stroke limit upper/lower limit error occurrence	1: ErrorStop	The limit velocity is changed to 0, and an immediate stop is performed.[*5] After the stop, the control mode is changed to csp.
4	CPU module error occurrence	1: ErrorStop	
	PLC READY [Y0] is OFF		
5	Software stroke limit upper/lower limit error occurrence	1: ErrorStop	
6	Axis error detection[*3]	1: ErrorStop	
7	Execute of MC_Stop is TRUE	2: Stopping[*4]	The limit velocity is changed to 0, and an immediate stop is performed.[*5] After the stop, the control mode is changed to csp. (Not based on the deceleration of MC_Stop)
8	"Stop signal" of external input signal is TRUE	4: Standstill	The limit velocity is changed to 0, and an immediate stop is performed.[*5] After the stop, the control mode is changed to csp.

*1 The servo turns OFF in the driver side and immediate stop is performed, and also the motion side command is stopped.

*2 Operation differs depending on the setting value selected in AxisName.Pr.StopMode_ServoOff. For details, refer to the following.
☞ Page 163 Relevant variables/FB

*3 If an axis error (the error that transits the state into the "1: ErrorStop" status) occurs in the FB which can be linked by the buffer mode, deceleration stop will be performed from the error occurrence.

*4 When "Execute" of MC_Stop is FALSE at stop completion, the axis status (AxisName.Md.AxisStatus) after stop will be "4: Standstill".

*5 The set torque value is not changed. Note that it may take time to reach velocity 0 depending on the torque command value that is currently specified.

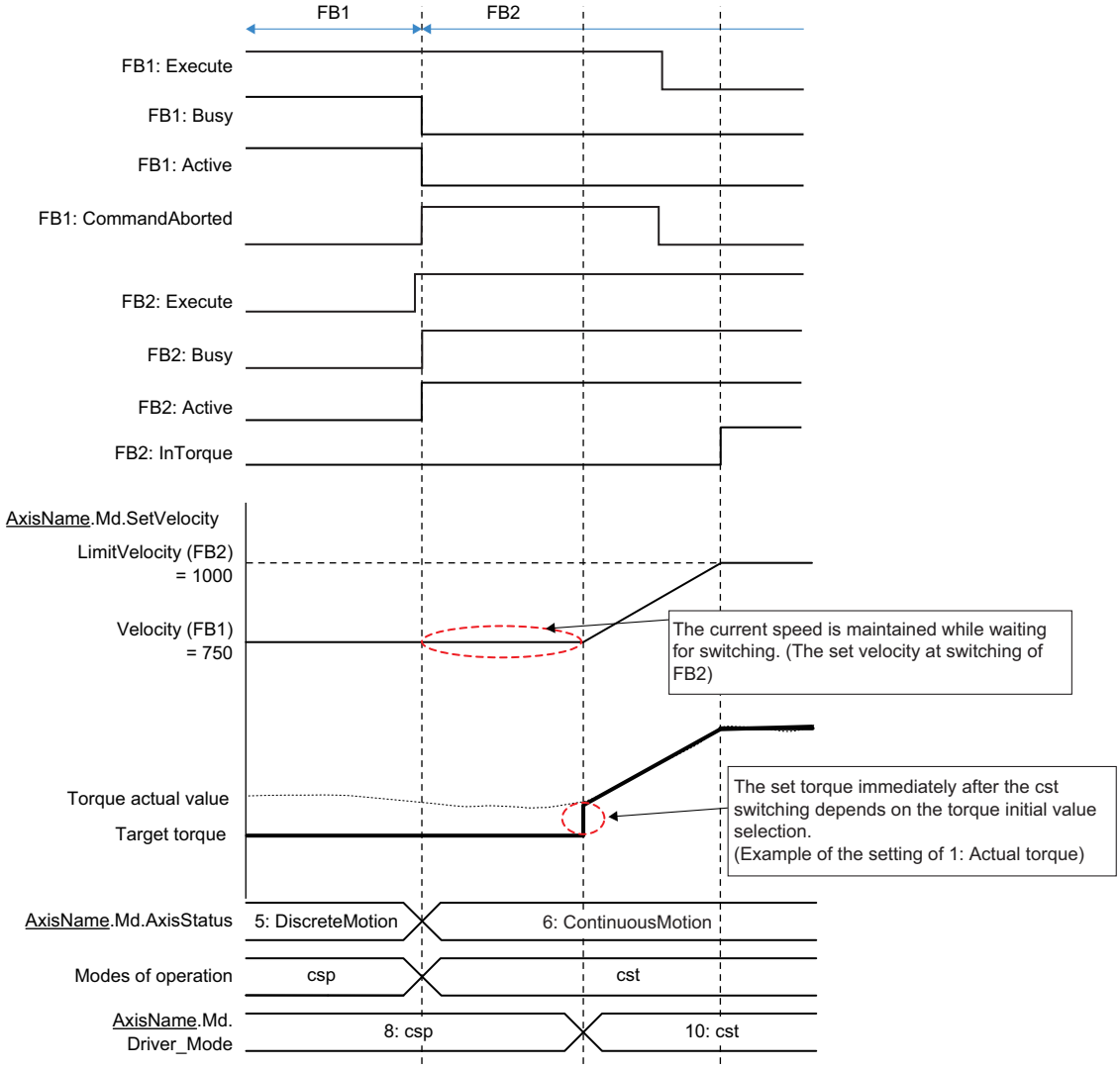
When starting this FB during another instruction execution

While this FB is executed, only Aborting and Buffered are supported.

■Position control (FB1) → Torque control (FB2)

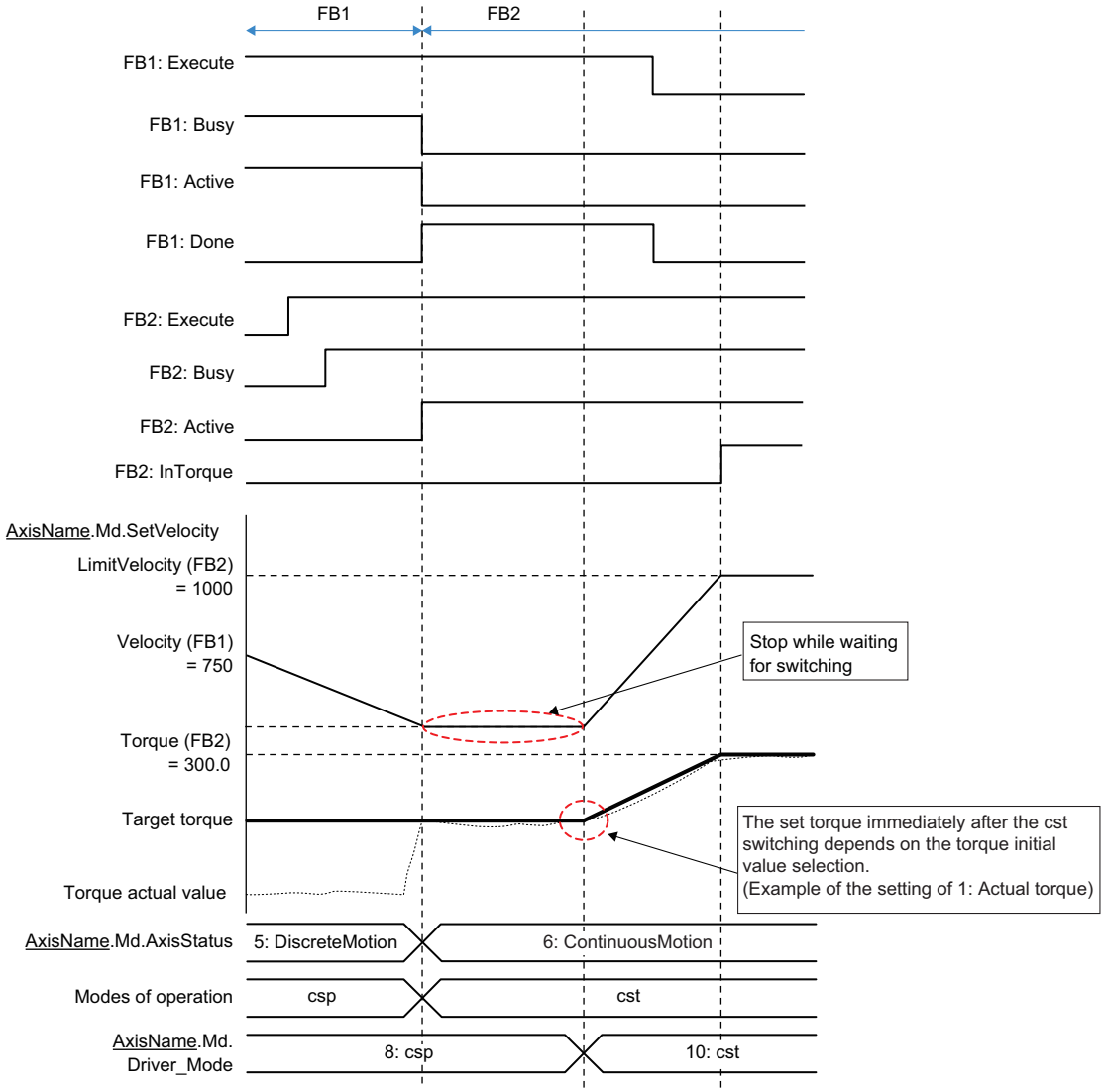
[0: mcAborting]

1. A switching request to cst is issued to the driver.
2. Until the driver is switched, the axis is operated at the previous velocity with the velocity control including the position loop.
3. When the driver switches to csv, the torque increase/decrease from the current actual torque starts and the acceleration/deceleration process of speed limit value from the current set velocity starts.



[1: mcBuffered]

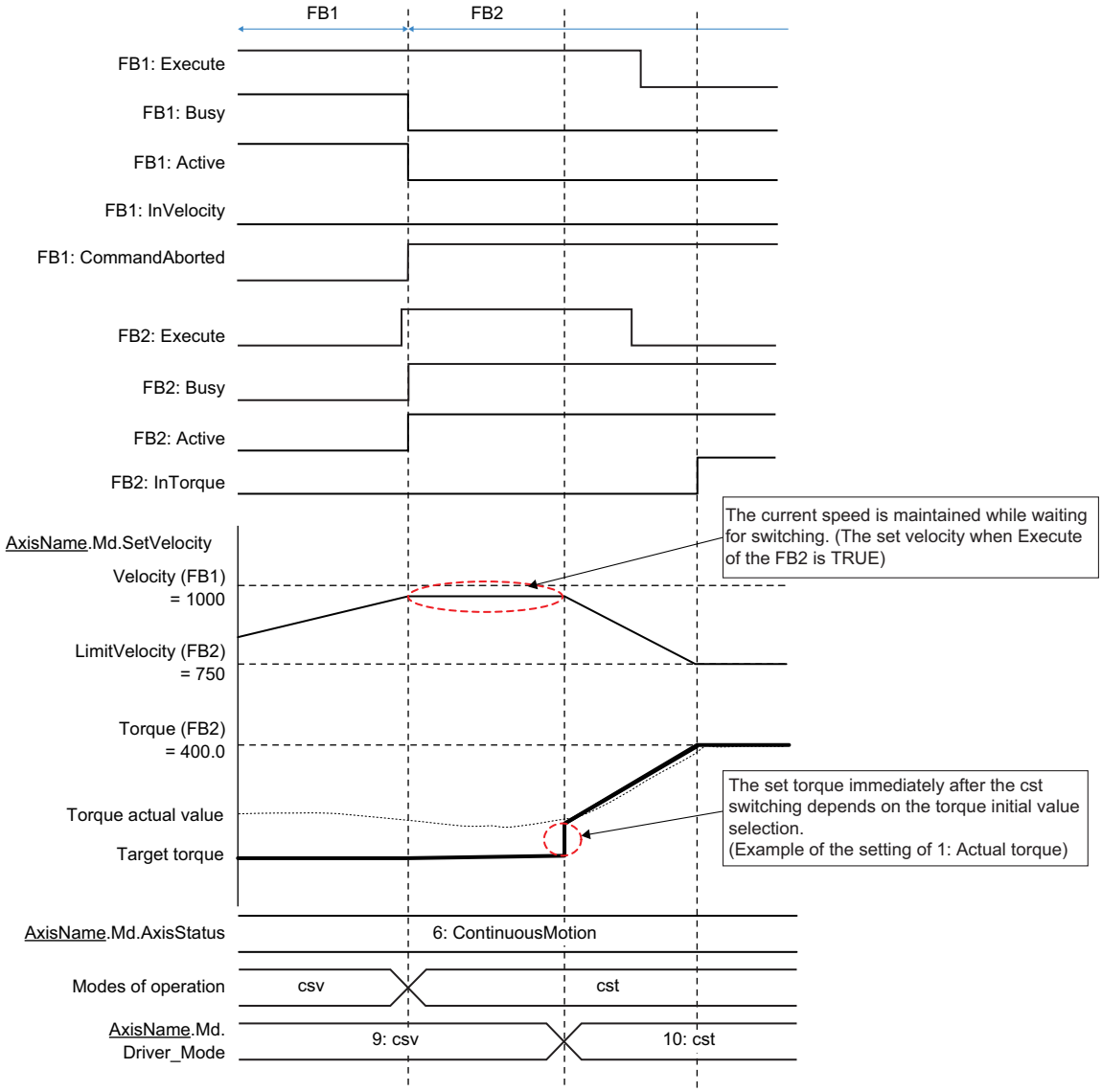
1. When the first instruction is completed, a cst change request is issued to the driver.
2. When the driver mode turns to the cst, the acceleration/deceleration process of the torque and the velocity is started from the current actual torque and the set velocity.



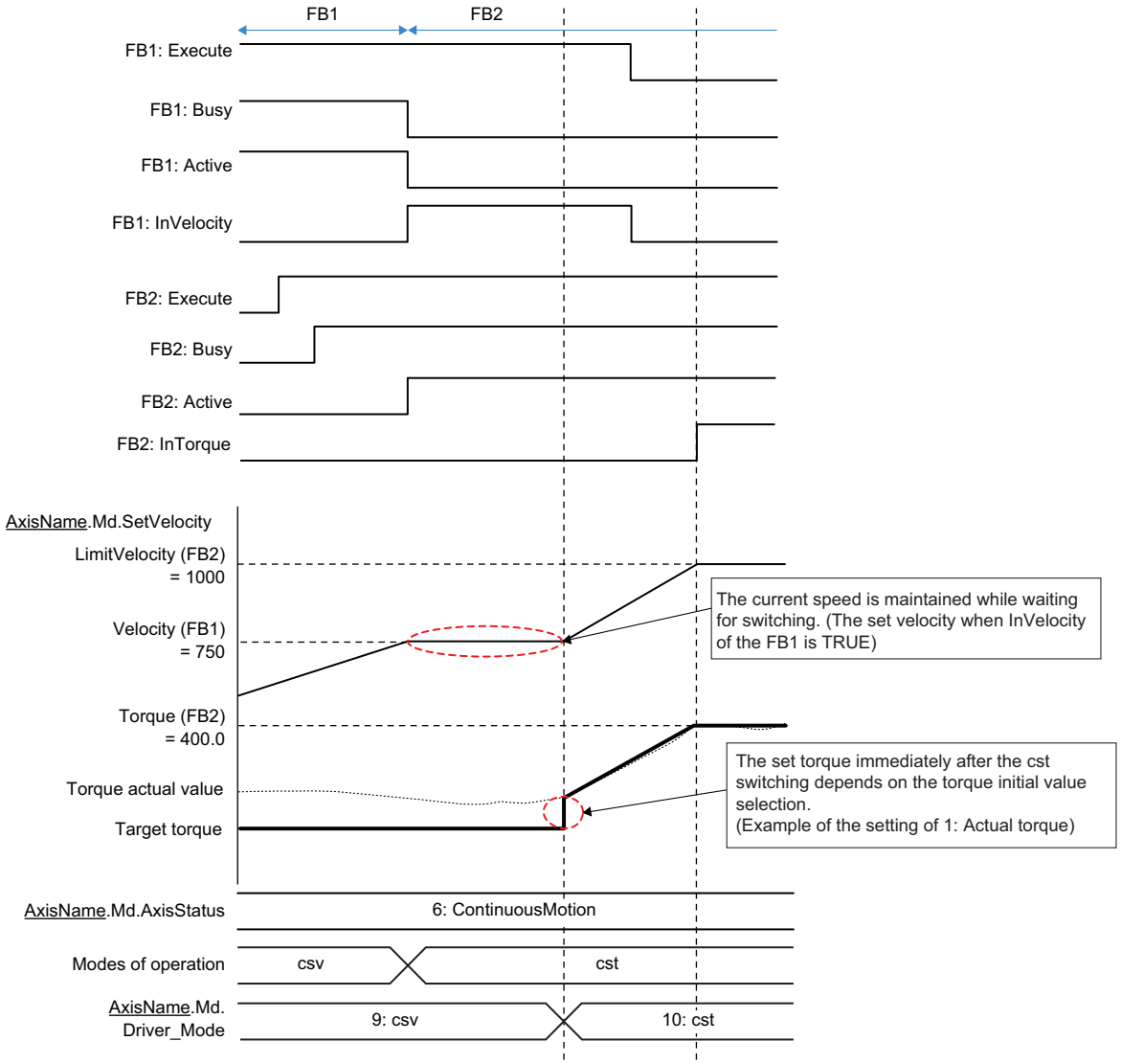
■Velocity control (FB1) → Torque control (FB2)

[0: mcAborting]

1. A cst change request is issued to the driver.
2. While waiting for change, csv is continued.
3. When the driver mode turns cst, the torque increase/decrease is started from the current actual torque, and the acceleration/deceleration process of the velocity from the set velocity is started.

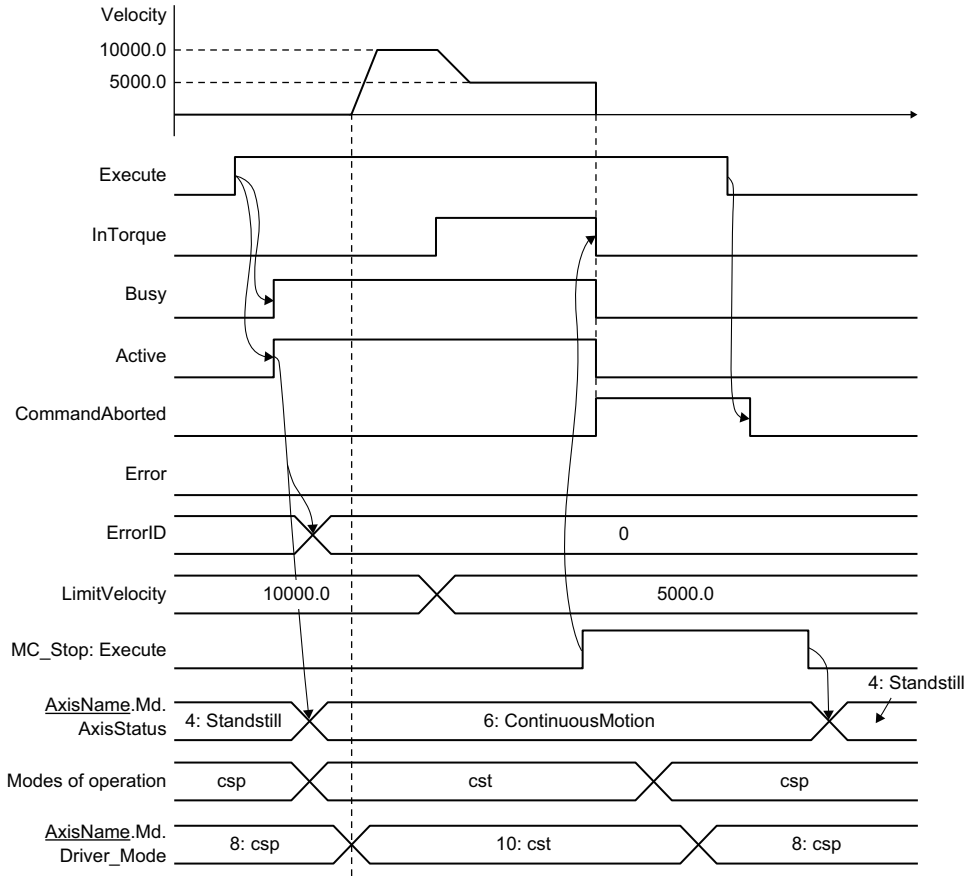


[1: mcBuffered]



When starting another instruction during this FB execution

■Torque control → Stop instruction (MC_Stop)



■Torque control → Positioning (Absolute value/Relative value positioning)

Aborting/Buffered

1. A csp change request is issued to the driver.
2. Started positioning control FB is analyzed, and the axis waits until the driver switches to the csp. (Even if the axis issues the switching request to the csp during operation, switching may not be carried out depending on the driver specifications. If switching to the csp is not completed within 1 second, the error "Control Mode Switching Error" (error code: 1A1DH) occurs due to a timeout and the axis stops.)
3. While waiting for switching, torque control is continued with the torque and the limit velocity at switching.
4. When the driver switches to the csp, the positioning control starts.

Point

The deceleration distance may not be secured and the target position may be overrun depending on the travel distance switching from the csv mode to the csp mode. In this case, the axis operates based on AxisName.Pr.OverrunOperation.

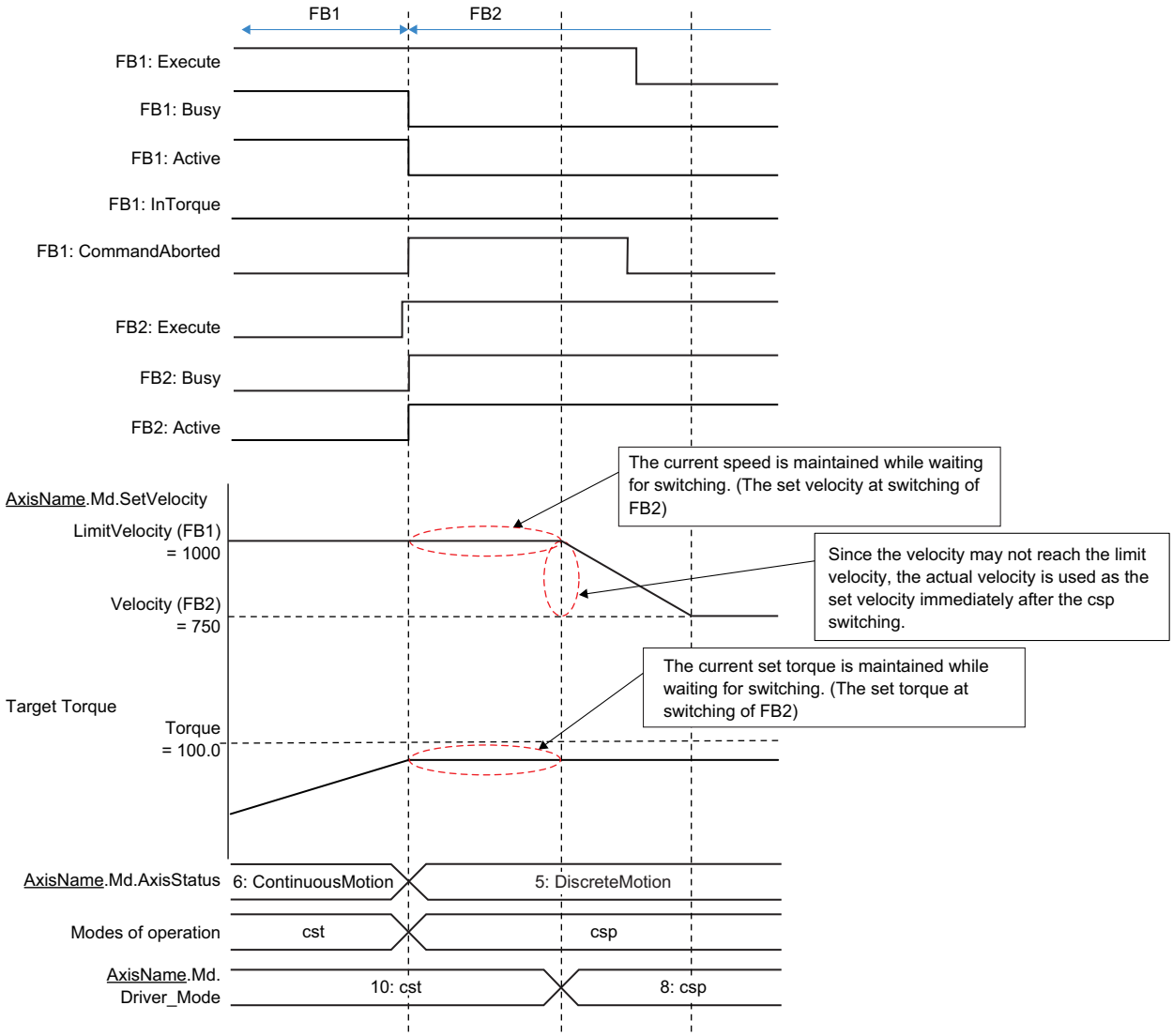
For operation at the overrun occurrence, refer to the following.

☞ Page 132 Multiple Start (Buffer Mode)

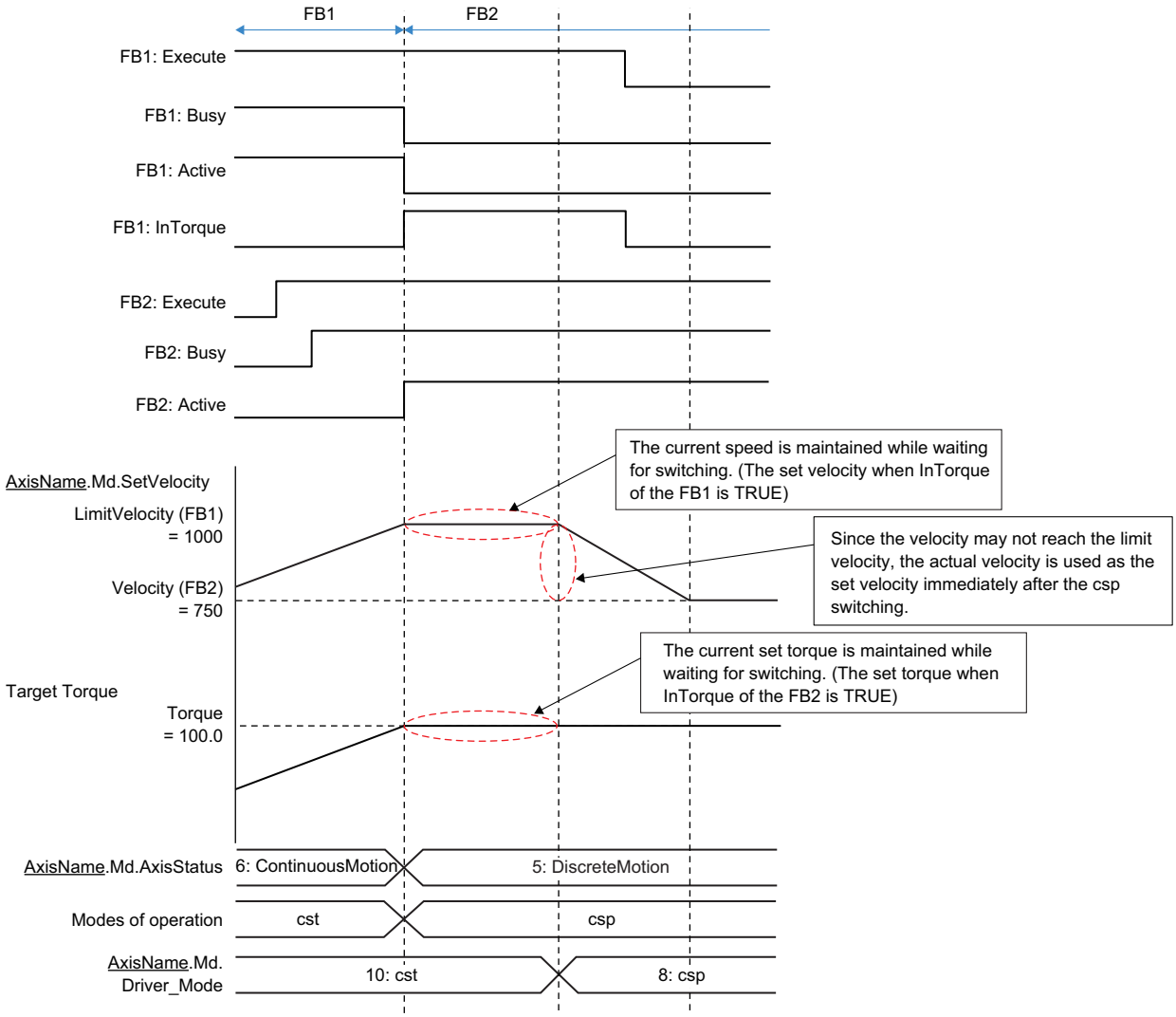
Precautions

If the axis overruns while waiting for switching, it will stop immediately after switching to csp.

[0: mcAborting]



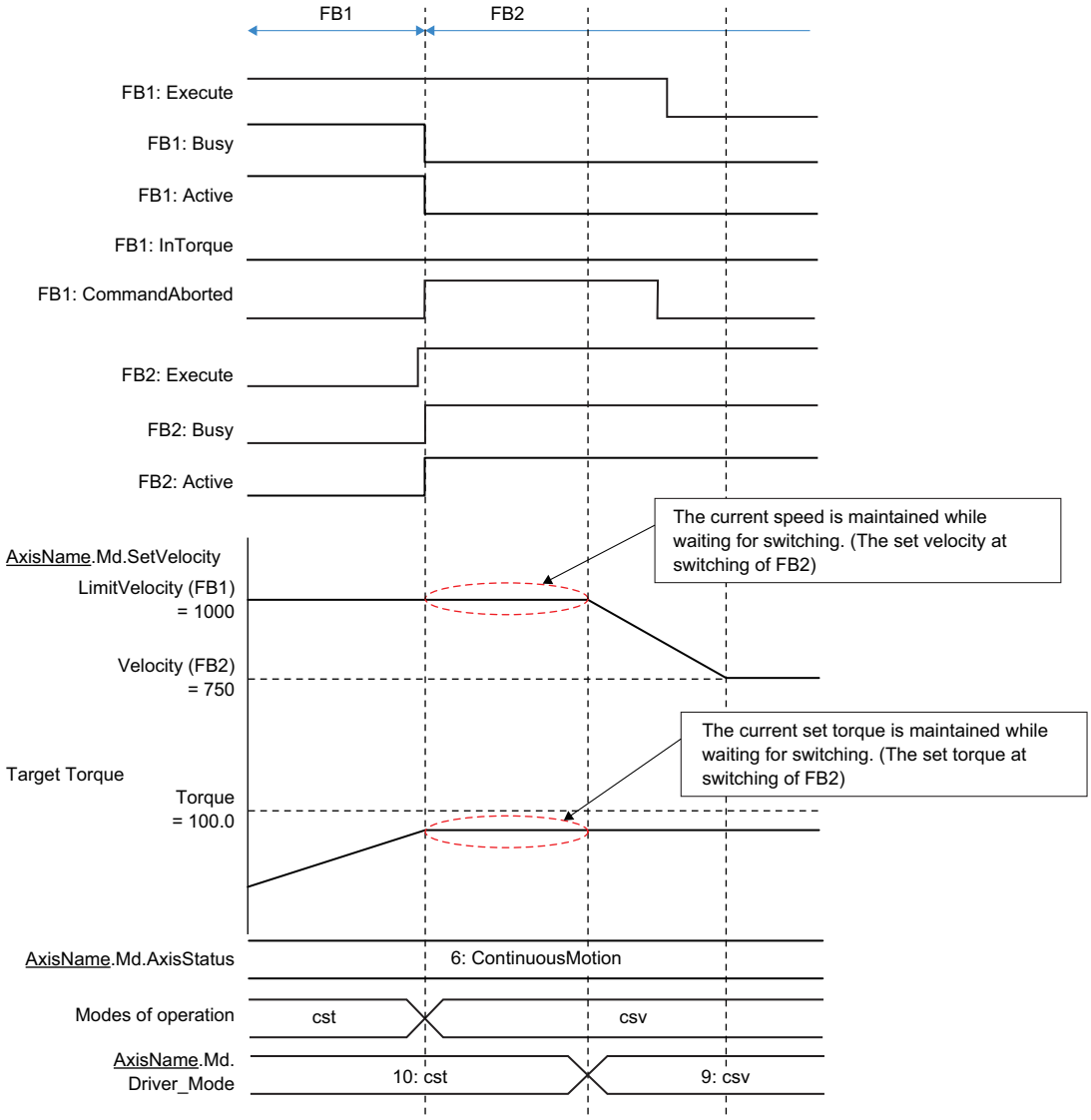
[1: mcBuffered]



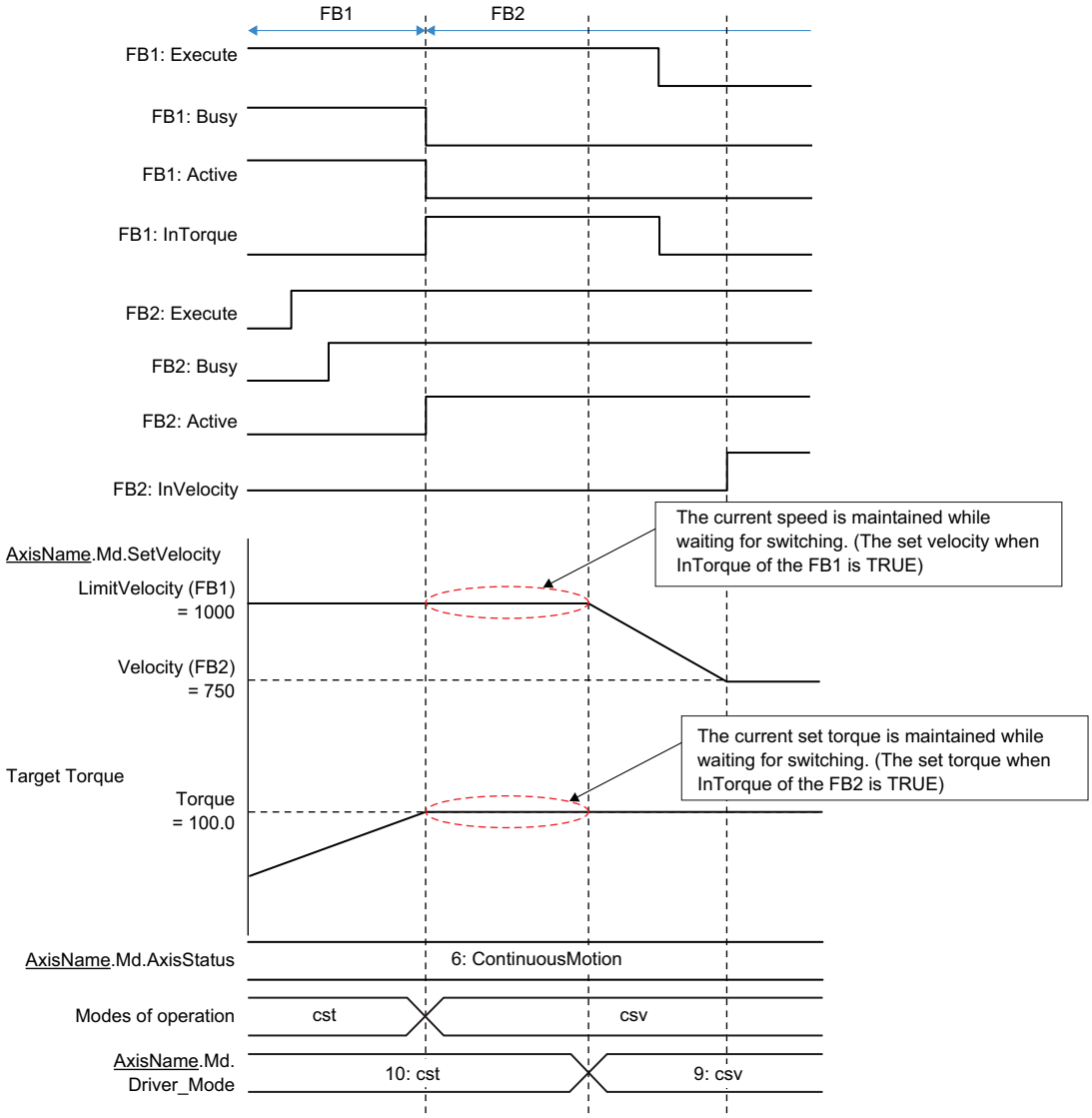
■Torque control (FB1) → Velocity control (FB2)

1. A csv switching request is issued to the driver.
2. Started positioning control FB is analyzed, and the axis waits until the driver switches to the csv. (Even if the axis issues the switching request to the csv during operation, switching may not be carried out depending on the driver specifications. If switching to the csv is not completed within 1 second, the error "Control Mode Switching Error" (error code: 1A1DH) occurs due to a timeout, and the axis will stop.)
3. While waiting for switching, torque control is continued with the torque and the limit velocity at switching.
4. When the driver switches to the csv, velocity control is started.

[0: mcAborting]



[1: mcBuffered]



When the operation direction is reversed

When starting this FB and operation direction is reversed during another instruction execution, reverse the set torque according to the torque positive direction ramp and the torque negative direction ramp.

When starting FB operated in csp mode and operation direction is reversed during this instruction execution, set velocity at csp mode switching is set to 0. After that the operation starts to the changed direction.

Precautions

- AxisName.Cd.VelocityOverride and AxisName.Cd.AccelerationOverride are valid.
- The set position and the feed machine position are updated by follow up.
- Time until the control mode is switched 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 the positioning control FB during the control mode switching. Start the positioning control FB after confirming that AxisName.Md.Driver_Mode is switched to "10: cst".

When using the MR-J5(W)-G

When switching from the csp to the cst or from the cst to the csp without waiting for stop of the motor, note the followings.

- Set "ZSP disabled selection at control switching" (PC76.1) of the servo parameter (extension setting) to "1: Disabled" and disable monitoring of zero speed status. However, note that it may cause vibration or impact at control mode switching.
- For the setting value of the servo parameter "Electronic gear numerator" (PA06) and "Electronic gear denominator" (PA07), refer to the following.

☞ Page 731 Setting methods

Torque limit change during the cst (cyclic synchronous torque mode)

At restart or continuous update, if the target torque is changed to a value bigger than the torque limit value (AxisName.Md.TorqueLimit_Positive, AxisName.Md.TorqueLimit_Negative), the warning "Torque Limit Value Over Warning" (warning code: 0D12H) will occur and the axis operates at the value before change.

During torque control, if AxisName.Cd.TorqueLimit_Positive and AxisName.Cd.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 at 1 cycle.

8 RELEVANT FUNCTIONS to POSITION

8.1 Current Position Change Function

- This function executes control to change the set position to an arbitrary address. The current position change method includes a method of changing the set position at control execution (MC_SetPosition) and a method of changing to the set position corrected at the trigger input signal detection timing (MCv_SetPositionTriggered). (The feed machine position is not changed.)
- The "absolute position set" and the "relative position set" can be specified in the address to change.
- This function can be executed only while the axis status (AxisName.Md.AxisStatus) is "4: Standstill".
- A method of changing the set position at control execution can be performed even though the axis is in operation. However, it cannot be performed while the axes group status (AxesGroupName.Md.GroupStatus) is "5: GroupMoving".
- A current position change to outside the ring counter range and outside the software stroke limit range cannot be executed.
- For operation when executing a current position change to the synchronous control master axis, refer to the following.
☞ Page 438 SYNCHRONOUS CONTROL

Point

When a current position change is completed, that "the current position change is executed" is recorded in the event history.

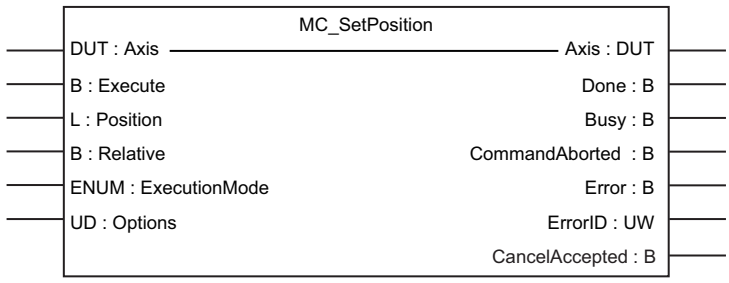
Targets to be changed and corresponding FBs

Target to be changed			Corresponding FB
Position	Current value		MC_SetPosition
			MCv_SetPositionTriggered
	Limit value (Software stroke limit)		—*1
	Limit value (Software stroke limit)	Maximum value of the limit value	—*2

*1 Axis parameters can be changed by MC_WriteParameter. However, it cannot be changed during control.
*2 The maximum value of the limit value can not be changed.

Relevant FBs

MC_SetPosition

Item	Description
Function overview	Changes the current position of the specified axis (set position, actual position).
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

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

■Function description

This FB changes the current position of the specified axis.

- When Relative is TRUE, the current position is changed to the position obtained by the addition of the target position (relative distance) and the current position.
- When Relative is FALSE, the current position is changed to the target position (absolute position).
- This FB is executed when Execute turns TRUE, and Busy turns TRUE when the processing is normally started.
- When the processing is completed and the current position is changed, Done turns TRUE.
- After starting this FB when ExecutionMode turned "1: mcQueued", if the axis status (AxisName.Md.AxisStatus) turns "0: Disabled" / "1: ErrorStop" before processing completion, the error "MC_SetPosition Instruction Error" (error code: 344EH) will occur.
- When an error has occurred in the FB, this FB turns Error to TRUE and stores an error code in ErrorID.
- For details on error codes, refer to the following.

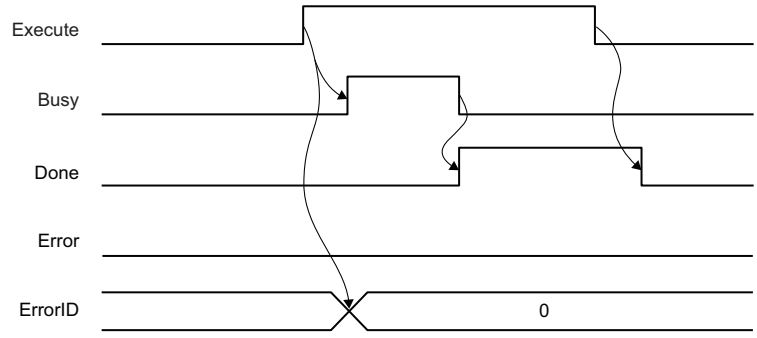
☞ Page 662 List of Error Codes

Precautions

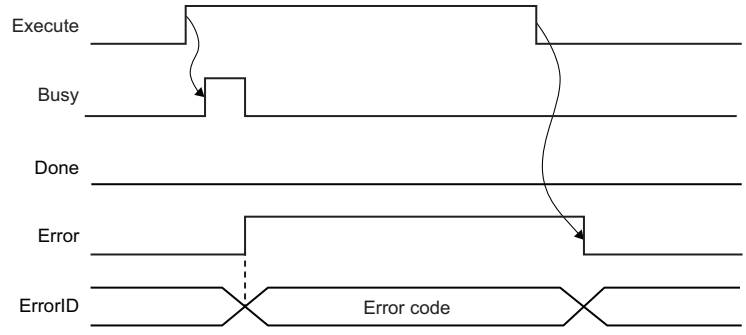
A current position change can be executed during axis operation. If the current position is changed during axis operation, the target position of the on-going FB will not change, but since the current position is changed, the operation to the target value will be changed.

■Timing chart

[When the FB is normally completed]



[When an error occurs]



■I/O variable

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Axis information	Axis	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables

■Input variables

Import ↑: At start

Name	Variable name	Data type	Import	Setting range	Default value	Description
Execute command	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Target position	Position	LREAL	↑	☞ Page 310 Position setting range	0.0	Sets the target position value.
Relative position selection	Relative	BOOL	↑	TRUE, FALSE	FALSE	The current position change with relative distance is set when Relative turns TRUE. The current position change with absolute position is set when Relative is FALSE.
Execution mode	ExecutionMode	MC_EXECUTION_MODE	↑	1 to 3	3	1: mcQueued (Execute at completion) 3: mcSpeculatively (Execute speculatively) When "1: mcQueued", it executes after the previous FB is completed. When "3: mcSpeculatively", it can execute only when there is no on-going FB on the specified axis.
Options	Options	DWORD(HEX)	↑	*1	00000000H	Sets a function option with bit specification.

*1 For a bit and the function description, refer to the following table.

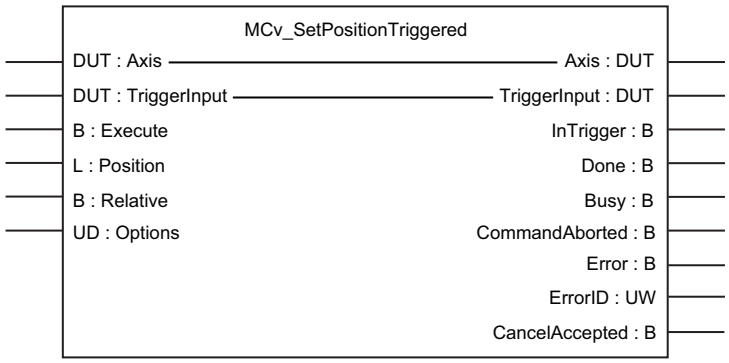
Bit	Function description
0 to 15	Empty (Specifiy "0".)*1
16	Cancel accepted after FB start Specifies whether cancel is allowed or not after FB is started. 0: Not allow 1: Allow
17 to 31	Empty (Specifiy "0".)*1

*1 If specifying other than "0", the error "Out of Options Range" (error code: 1A4EH) occurs, and the function does not start.

■Output variables

Name	Variable name	Data type	Default value	Description
Execution completion	Done	BOOL	FALSE	Indicates that the current position change is completed.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Abortion of execution	CommandAborted	BOOL	FALSE	When TRUE, it indicates that execution is interrupted.
Error	Error	BOOL	FALSE	When TRUE, it indicates that an error has occurred
Error code	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned. For details, refer to the following. ☞ Page 662 List of Error Codes
Cancel acceptance	CancelAccepted	BOOL	FALSE	Indicates that the FB accepted cancel.

MCv_SetPositionTriggered (Future support planned)

Item	Description
Function overview	Changes the current position of specified axis (set position, actual position) to the position compensated at the trigger input signal detection timing.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

Input area points(byte)	Output area points(byte)	Compiling method	FB operation type
—	—	Subroutine type	Real-time execution

■Function description

This FB changes the current position of the specified axis.

- The current position is changed when the TriggerInput signal is detected. When the processing is completed, Done turns TRUE.
- When Relative is TRUE, the current position is changed to the position obtained by the addition of the target position (relative distance) and the current position.
- When Relative is FALSE, the current position is changed to the target position (absolute position).
- This FB is executed when Execute turns TRUE, and Busy turns TRUE when the processing is normally started.
- When an error has occurred in the FB, this FB turns Error to TRUE and stores an error code in ErrorID.
- For details on error codes, refer to the following.

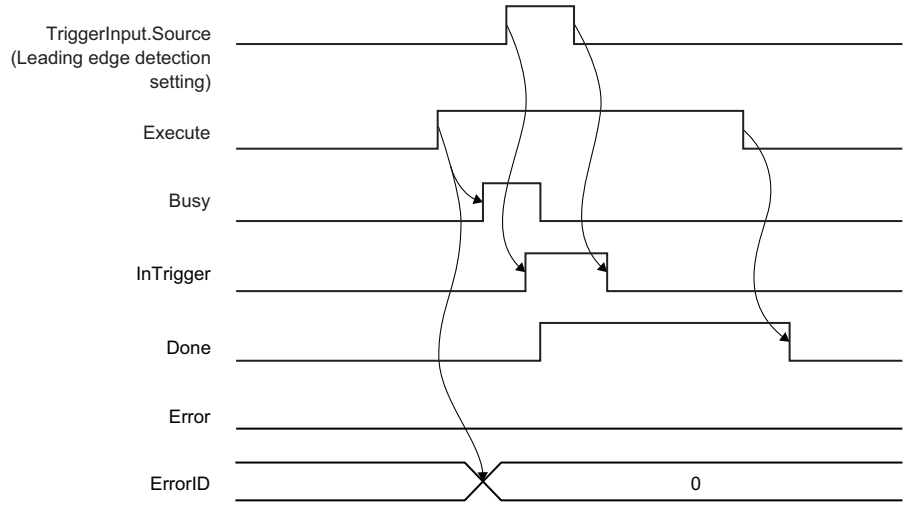
☞ Page 662 List of Error Codes

Precautions

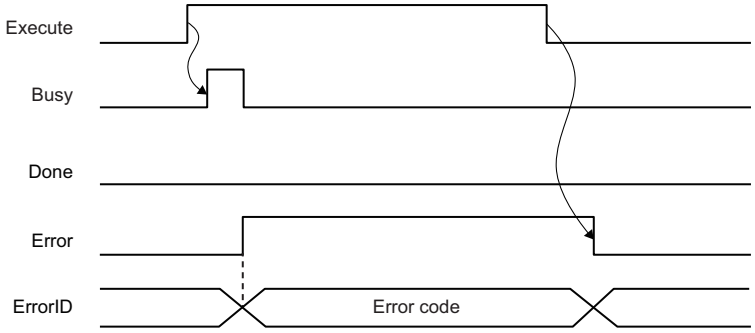
This FB can be executed only when the axis status (AxisName.Md.AxisStatus) is "4: Standstill". If it is other than "4: Standstill", an error occurs and it cannot be executed.

■Timing chart

[When the FB is normally completed]



[When an error occurs]



■I/O variables

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Axis information	Axis	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables
Trigger input signal	TriggerInput	MC_TRIGGER_REF	↑	—	Omission not possible	☞ Page 418 External Signal Selection Only "RisingEdge", "FallingEdge", and "BothEdges" can be selected for Detection (signal detection method). A value within the range of "-5.0 to 5.0" can be specified to CompensationTime (compensation time [s]) when Relative is FALSE. Set "0.0" when Relative is TRUE. If the setting value is not within the valid range, an error will occur.

■Input variables

Import ↑: At start

Name	Variable name	Data type	Import	Setting range	Default value	Description
Execute command	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Target position	Position	LREAL	↑	☞ Page 310 Position setting range	0.0	Sets the target position value.
Relative position selection	Relative	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, it sets the current position change by relative distance. When FALSE, it sets the current position change by absolute distance.
Options	Options	DWORD(HEX)	↑	*1	00000000H	Sets a function option with bit specification.

*1 For a bit and the function description, refer to the following table.

Bit	Function description
0 to 15	Empty (Specifiy "0".)*1
16	Cancel accepted after FB start Specifies whether cancel is allowed or not after FB is started. 0: Not allow 1: Allow
17 to 31	Empty (Specifiy "0".)*1

*1 If specifying other than "0", the error "Out of Options Range" (error code: 1A4EH) occurs, and the function does not start.

■Output variables

Name	Variable name	Data type	Default value	Description
Detecting signal	InTrigger	BOOL	FALSE	Indicates the detection status of the trigger input signal. The signal detection status is output as it is regardless of the correction time set in the trigger input signal.
Execution completion	Done	BOOL	FALSE	Indicates that the current position change is completed.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Abortion of execution	CommandAborted	BOOL	FALSE	When TRUE, it indicates that execution is interrupted.
Error	Error	BOOL	FALSE	When TRUE, it indicates that an error has occurred
Error code	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned. For details, refer to the following. ☞ Page 662 List of Error Codes
Cancel acceptance	CancelAccepted	BOOL	FALSE	Indicates that the FB accepted cancel.

Position setting range

Values that can be specified in Position differs depending on whether Relative is FALSE or TRUE. Specify within each of the following range.

Relative	Position setting range
FALSE	Ring counter lower limit value ≤ Setting value < Ring counter upper limit value
TRUE	-(Ring counter upper limit value - Ring counter lower limit value) / 2 ≤ Setting value ≤ (Ring counter upper limit value - Ring counter lower limit value) /2

Even if a value outside the setting range is set, an error will occur and Position will not be executed when the set position after a current position change is outside the software stroke limit range.

Current position change

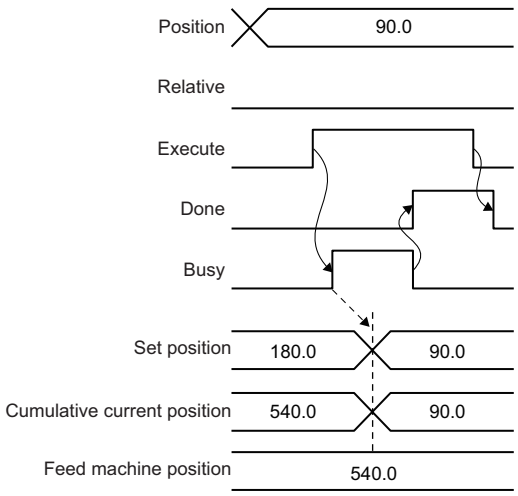
This section describes control details of MC_SetPosition.

Execution when the axis stops

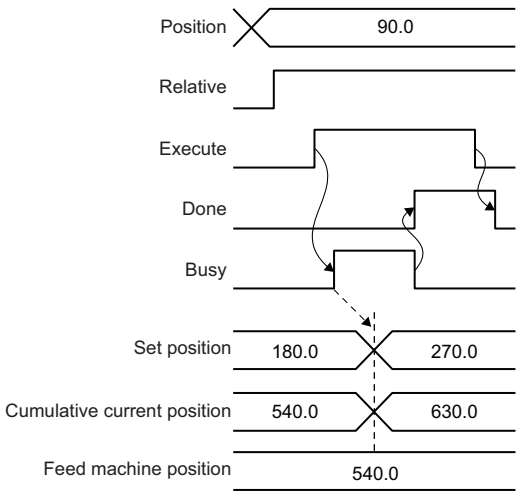
The current position change operation starts at rising edge detection of Execute.
In the current position change, the set position and the cumulative current position are changed to the position specified in Position.
Position is "Relative position set" when Relative is TRUE, and it is "Absolute position set" when Relative is FALSE.

Ex.
When executing the current position change to 90.0 while the axis stops with the ring counter lower limit value of 0.0, the ring counter upper limit value of 360.0, and the set position of 180.0 (the cumulative current position of 540.0).

<When Relative is FALSE>



<When Relative is TRUE>



Execution during axis operation

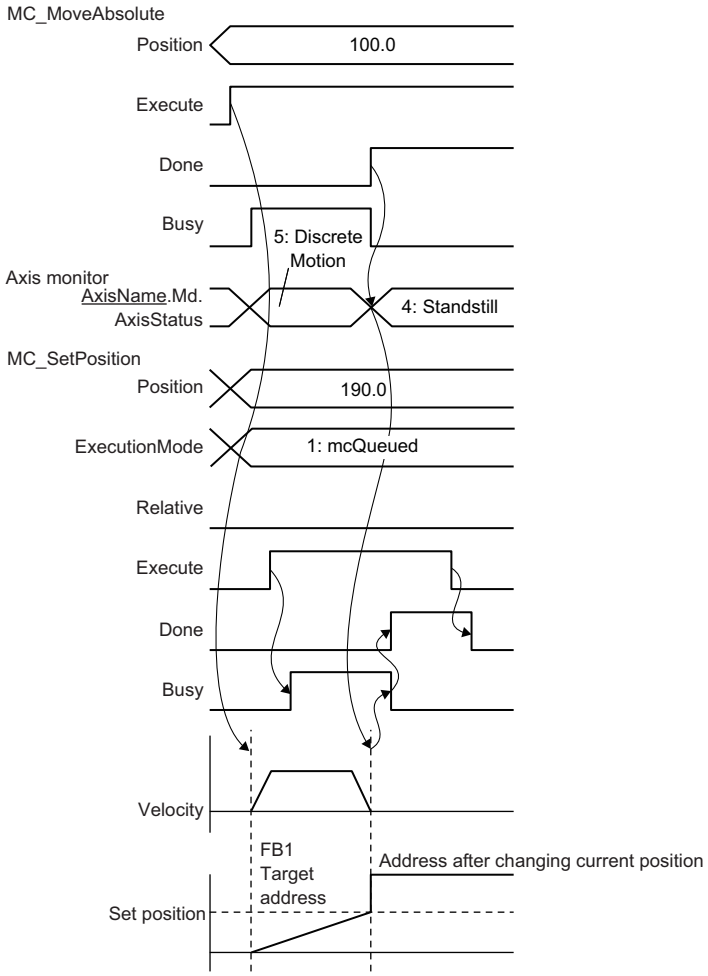
■Execute at completion

When "1: mcQueued" is set in ExecutionMode, this FB is executed after the on-going FB is completed.

Busy turns TURE by rising edge detection of Execute, and this FB waits until the axis status (AxisName.Md.AxisStatus) turns "4: Standstill".

As soon as the axis status (AxisName.Md.AxisStatus) turns "4: Standstill", the current position change operation will start.

Ex.



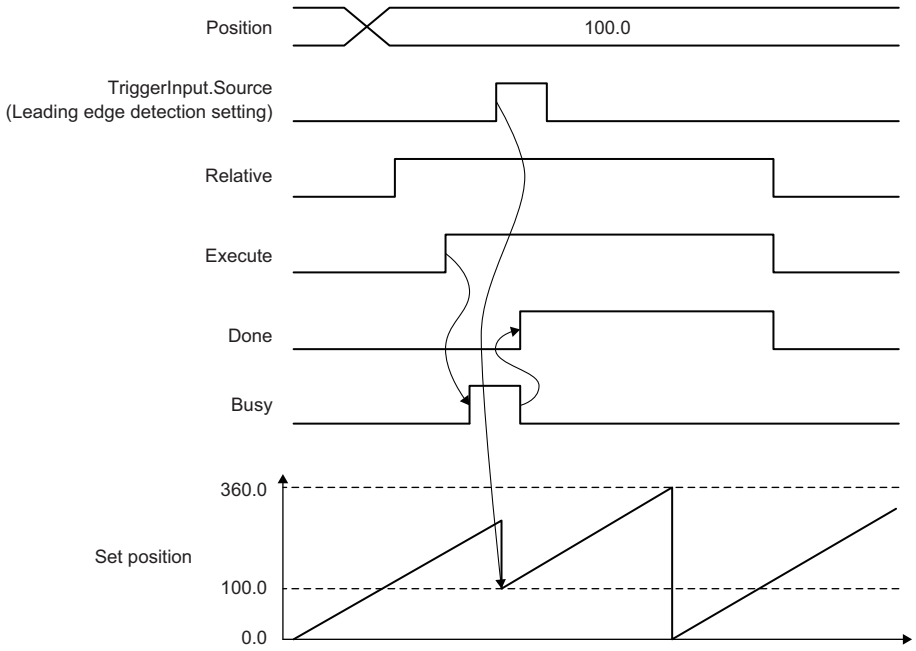
■Speculative execution

When specifying "3: mcSpeculatively" to ExecutionMode, if the axis status (AxisName.Md.AxisStatus) is not "4: Standstill" or if the rising edge of Execute is detected at execution of FB instruction, the error "Axis Status Incorrect Start" (error code: 1A26H) occurs.

Trigger current position change

After rising edge detection of Execute, the current position change operation starts at signal detection of TriggerInput.
When the correction time of TriggerInput is a positive value, a current position change is immediately executed.
When the correction time of TriggerInput is a negative value, this FB waits for operation to start until the correction time has elapsed, and a current position change is executed.
In the current position change, the set position and the cumulative current position are estimated and changed so that the position will be a position set in Position at the signal detection timing of TriggerInput.
Position is "Relative position set" when Relative is TRUE, and it is "Absolute position set" when Relative is FALSE.

Ex.
When a current position change to 100.0 is executed in the real encoder axis during a follow up (during a counter enable) with the ring counter lower limit value of 0.0, the ring counter upper limit value of 360.0, and the axis status (AxisName.Md.AxisStatus) "4: Standstill".



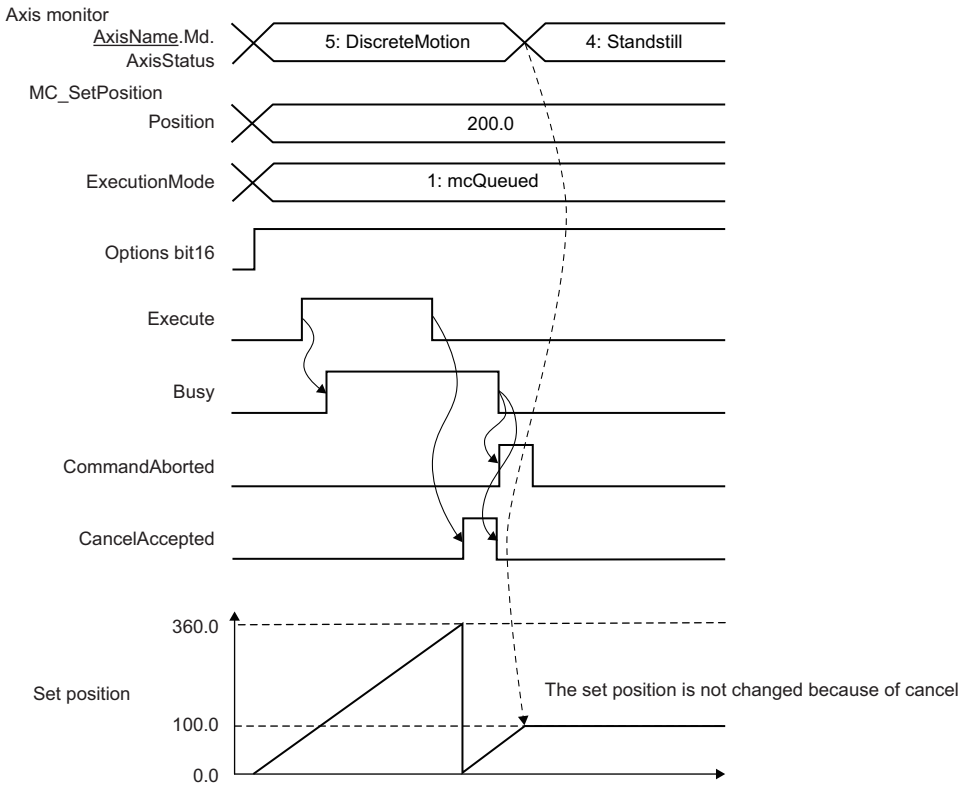
Cancel

Current position change that is in the standby status can be canceled after FB execution.

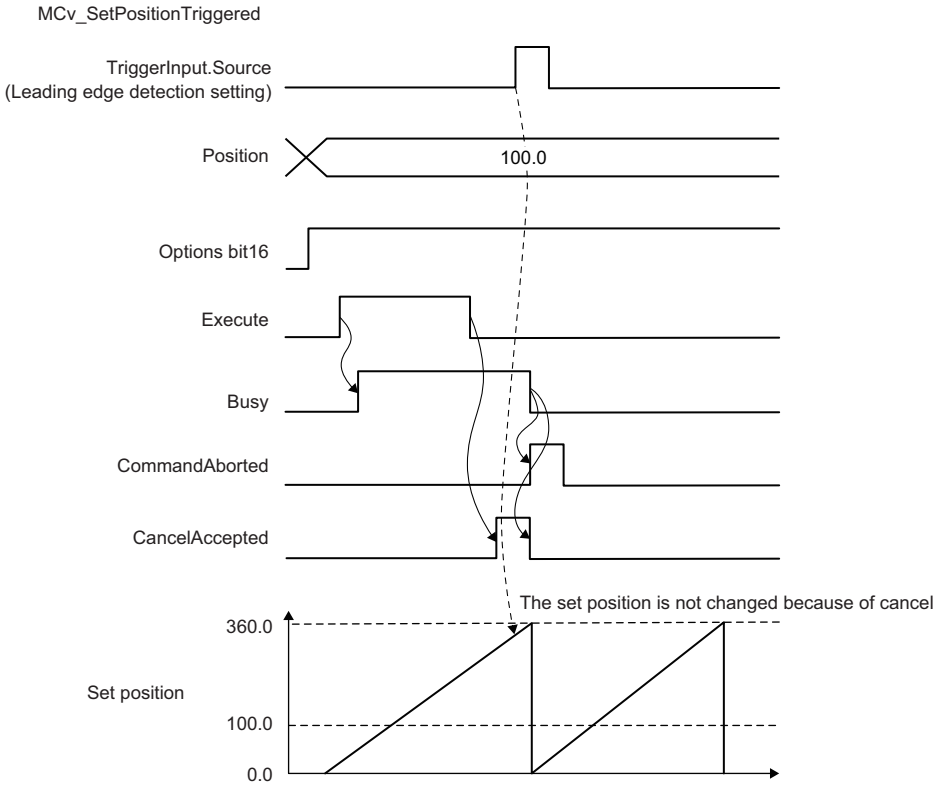
- To enable cancel, execute the FB with the status of "Cancel accepted after FB start" (Options bit16) set to "1: Allowed".
- Cancel is started at rising edge detection of Execute.
- Cancel is accepted only when Busy of the output pin is TRUE.
- When FB accepts cancel, CancelAccepted of the output pin turns TRUE.
- When cancel is completed, CommandAborted of the output pin turns TRUE.
- When cancel was carried out, the set position is not changed.

Ex.

When canceling while the axis status (AxisName.Md.AxisStatus) is waiting for switching to "4: Standstill" after starting MC_SetPosition with "1: mcQueued" in ExecutionMode



When canceling while the axis status is waiting for detecting a trigger signal after starting MCv_SetPositionTriggered



8.2 Command In-position

The command in-position function checks the remaining distance to the target position, and turns the flag to TRUE.
This flag is called the "command in-position flag". The command in-position flag is used as a front-loading signal indicating beforehand the completion of the control.
The command in-position function is valid to all control that requires commanded address specification.
It turns AxisName.Md.CmdInPos to TRUE when the remaining distance to the target position became AxisName.Md.CmdInPos_Width or less.
It turns AxisName.Md.CmdInPos to FALSE when the axis starts and when the FB is switched by multiple start.

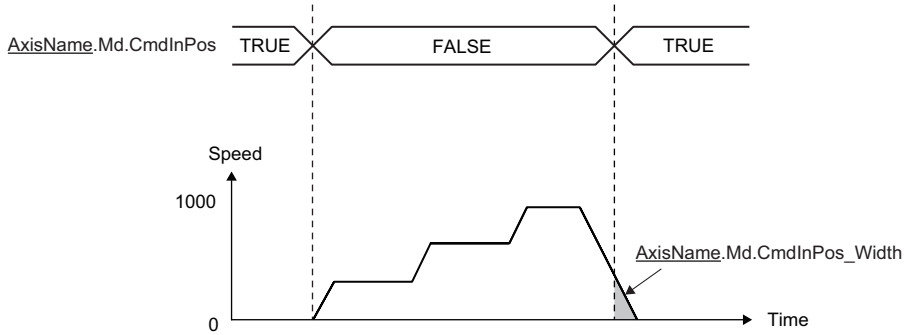
Relevant variables

Variable/Structure name		Name	Details
AxisName.Pr.			
	CmdInPos_Width	Command in-position width	Sets the command in-position width. [Setting range] • 0.0: Function disabled • 0.000000001 to 10,000,000,000.0: Function enabled*1
AxisName.Md.			
	CmdInPos_Width	Command in-position width	Displays the command in-position width. • The value imports the setting value of AxisName.Pr.CmdInPos_Width. • When AxisName.Md.CmdInPos_Width is 0.0, the command in-position function of the axis is disabled.
	CmdInPos	Command in-position	Indicates whether the remaining distance to the target position is AxisName.Md.CmdInPos_Width or less.
AxesGroupName.Pr.			
	CmdInPos_Width	Command in-position width	Sets the command in-position width on the composite axis of the axes group. [Setting range] • 0.0: Function disabled • 0.000000001 to 10,000,000,000.0: Function enabled*1
AxesGroupName.Md.			
	CmdInPos_Width	Command in-position width	Displays the command in-position width on the composite axis of the axes group. • The value imports the setting value of AxesGroupName.Pr.CmdInPos_Width. • When AxesGroupName.Md.CmdInPos_Width is 0.0, the command in-position function of the axes group is disabled.
	CmdInPos	Command in-position	Indicates whether the remaining distance to the target position on the composite axis is AxesGroupName.Pr.CmdInPos_Width or less.

*1 If positive numbers less than 0.000000001 are set, it imports 0.0

Command in-position width check

(Remaining distance) ≤ (AxisName.Md.CmdInPos_Width)



A command in-position width check is carried out every operation cycle.
However, while the axis stops, AxisName.Md.CmdInPos is not refreshed.
When AxisName.Md.CmdInPos_Width is 0.0, a command in-position width check is not carried out.
For an axes group, the remaining distance on the composite axis (line/arc) is checked based on AxesGroupName.Md.CmdInPos_Width.
The flag is stored in AxesGroupName.Md.CmdInPos.
During axes group operation, the command in-position function of the configuration axes is disabled.
AxisName.Md.CmdInPos of the configuration axes is not turned TRUE during axes group operation. (It turned FALSE when the axes group is started.)

8.3 Software Stroke Limit

In the "software stroke limit function", an address is set as the movable range. Even if a movement command to the address outside the setting range is issued, the command will not be executed. By not executing the command outside the movable area, even if an incorrect command or unexpected operation occurs, the system will not operate out of the movable area and damage to the machine can be prevented.
In the motion system, the "set position" and "feed machine position" are used as the addresses indicating the current value. However, in the "software stroke limit function", which address is used to carry out the limit check is designated in AxisName.Pr.SwStrokeLimit_Target.

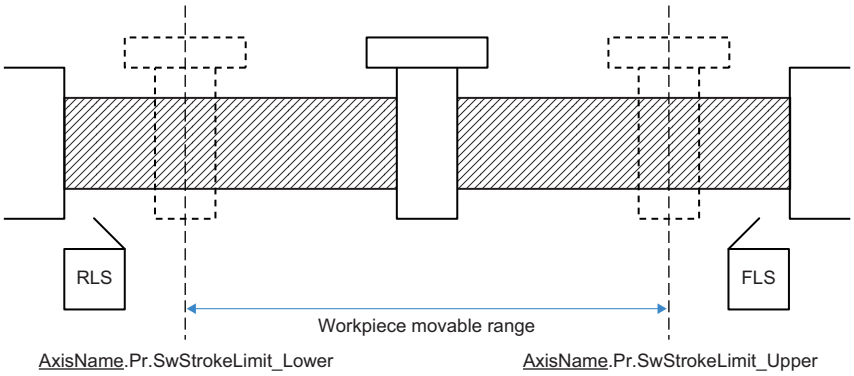
Relevant variables

Variable/Structure name	Name	Details
AxisName.Pr.		
SwStrokeLimit_Upper	Software stroke limit upper value	Sets the software stroke limit upper value.
SwStrokeLimit_Lower	Software stroke limit lower value	Sets the software stroke limit lower value.
SwStrokeLimit_Target	Software stroke limit target	Sets the software stroke limit target.
AxisName.Md.		
SwStrokeLimit_Upper	Software stroke limit upper value	Displays the software stroke limit upper value.
SwStrokeLimit_Lower	Software stroke limit lower value	Displays the software stroke limit lower value.
SwStrokeLimit_Target	Software stroke limit target	Displays the software stroke limit target.
SwStrokeLimit_Override	Software stroke limit override	Displays the software stroke limit valid/invalid status.
AxisName.Cd.		
SwStrokeLimit_Override	Software stroke limit override	Temporarily switches enabled/disabled of the software stroke limit.

Definition of movable range

Movable range differences by selection of set position and feed machine position

The following drawing shows the movable range of the workpiece when the software stroke limit function is used.

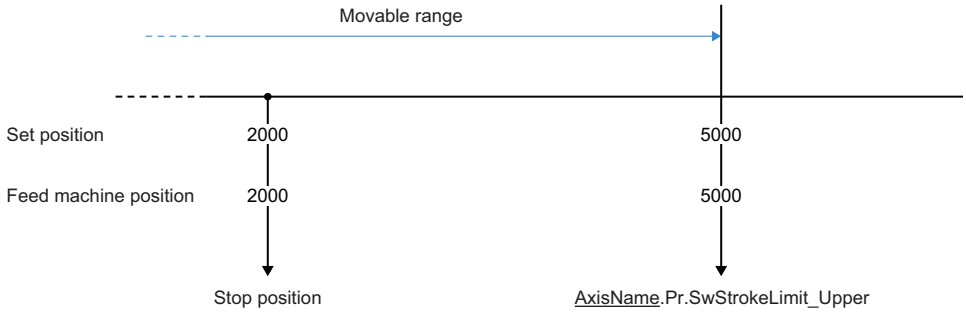


The set position or the feed machine position can be specified in the software stroke limit. (The cumulative current position cannot be specified.)

The following drawing shows the differences in the operation when the set position is used or when the feed machine position is used in the movable range limit check.

• Conditions

Assume the current value is 2000, and AxisName.Pr.SwStrokeLimit_Upper is set to 5000.

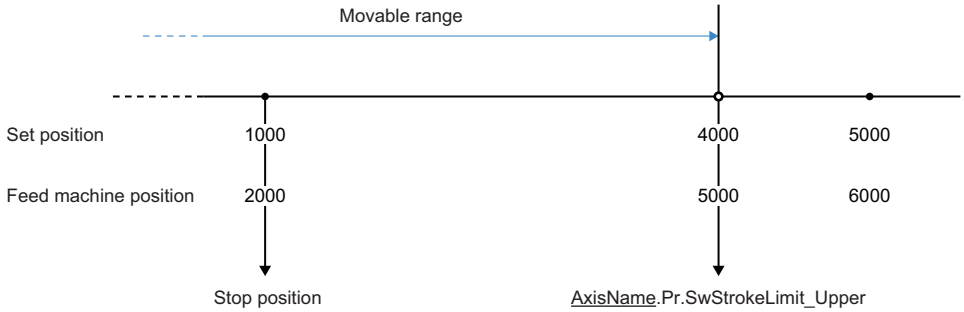


• Current value changing

When the current value is changed by a new current value command from 2000 to 1000, the set position will change to 1000, but the feed machine position will stay the same at 2000.

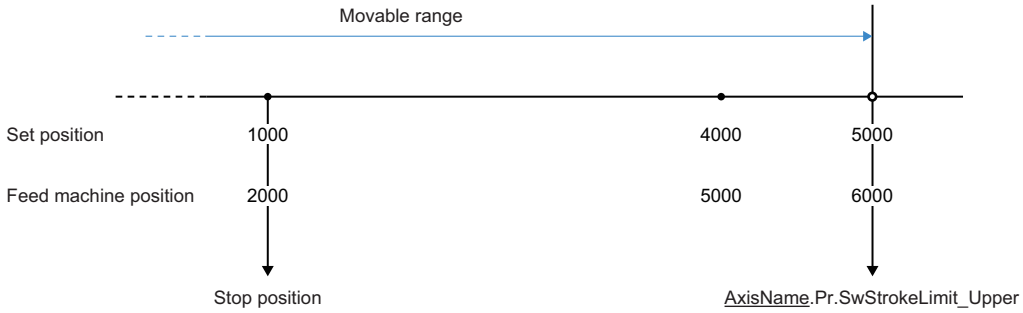
[When the feed machine position is set at the limit]

The set position of 4000 (feed machine position: 5000) becomes the upper stroke limit.



[When the set position is set at the limit]

The set position of 5000 (feed machine position: 6000) becomes the upper stroke limit.



Point

When "3: FeedMachinePosition" is set in AxisName.Pr.SwStrokeLimit_Target, the movable range becomes an absolute range referenced on the home position. When "1: SetPosition" is set, the movable range becomes the relative range from the set position.

■When selecting the set position when the ring counter is valid

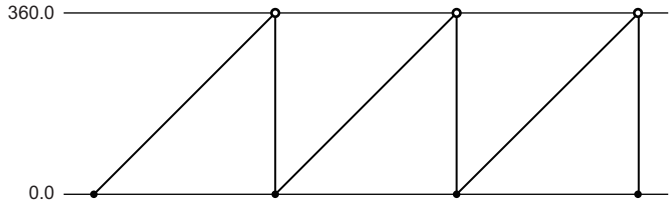
• Set position

The storing range of the set position upper/lower limit value is shown below.

Monitor data	Storing range
AxisName.Md.SetPosition	Ring counter lower value ≤ Set position < Ring counter upper value

Ex.

When the ring counter lower limit value is "0.0" and the ring counter upper limit value is "360.0", the set position address becomes the ring address of "0.0 to 359.999...".



• Setting the software stroke limit

The setting range of the software stroke limit upper/lower value is shown below.

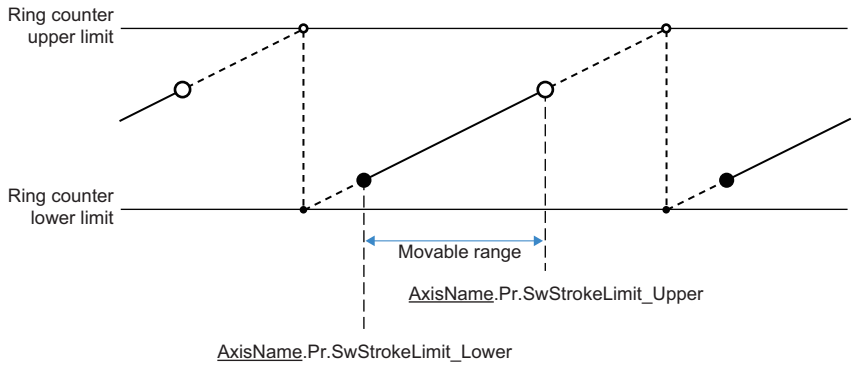
Parameter	Setting range
AxisName.Pr.SwStrokeLimit_Upper	Ring counter lower value < Setting value ≤ Ring counter upper value
AxisName.Pr.SwStrokeLimit_Lower	Ring counter lower value ≤ Setting value < Ring counter upper value

Movable range: AxisName.Pr.SwStrokeLimit_Lower ≤ Set position < AxisName.Pr.SwStrokeLimit_Upper

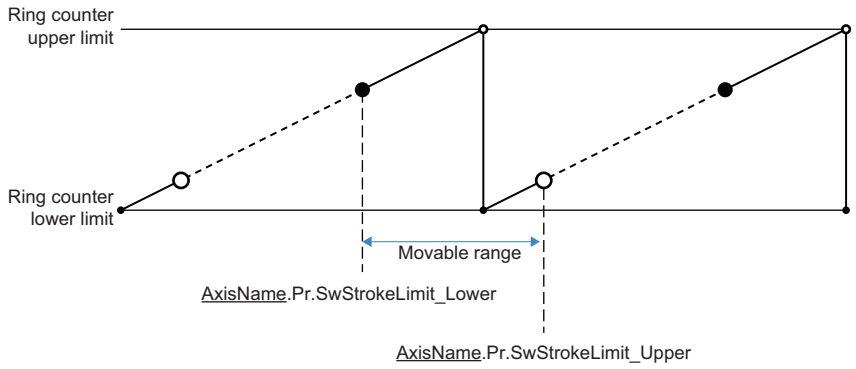
• Details of the movable range

The movable range varies as shown below depending on the relationship between AxisName.Pr.SwStrokeLimit_Upper and AxisName.Pr.SwStrokeLimit_Lower.

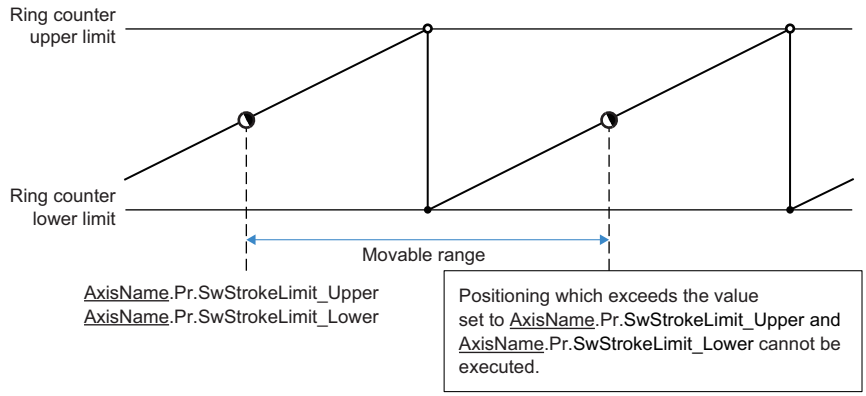
[AxisName.Pr.SwStrokeLimit_Upper > AxisName.Pr.SwStrokeLimit_Lower]



[AxisName.Pr.SwStrokeLimit_Upper < AxisName.Pr.SwStrokeLimit_Lower]



[AxisName.Pr.SwStrokeLimit_Upper = AxisName.Pr.SwStrokeLimit_Lower]



■When selecting the set position or the feed machine position when the ring counter is invalid

• Feed machine position

The storing range of the feed machine position upper/lower limit value are shown below.

Monitor data	Stroring range
AxisName.Md.FeedMachinePosition	10000000000.0 ≤ Feed machine position < 10000000000.0

• Setting the software stroke limit

The setting range of the software stroke limit upper/lower value is shown below.

Parameter	Setting range
AxisName.Pr.SwStrokeLimit_Upper	AxisName.Pr.SwStrokeLimit_Lower < Setting range ≤ 10000000000.0
AxisName.Pr.SwStrokeLimit_Lower	-10000000000.0 ≤ Setting range < AxisName.Pr.SwStrokeLimit_Upper

Movable range: AxisName.Pr.SwStrokeLimit_Lower ≤ Feed machine position < AxisName.Pr.SwStrokeLimit_Upper

Invalidating the software stroke limit

To invalidate the software stroke limit, set "-1: Invalid" in AxisName.Pr.SwStrokeLimit_Target. By setting "-1: Invalid", the software stroke limit check is not executed in all controls regardless of the setting values of AxisName.Pr.SwStrokeLimit_Upper and AxisName.Pr.SwStrokeLimit_Lower.

Returning from outside the software stroke limit range

The axis cannot start outside the software stroke limit range while the software stroke limit is valid.

After setting ONLY_INSIDE or DISABLE in AxisName.Cd.SwStrokeLimit_Override, execute the axis operation to return to the movable range.

AxisName.Cd.SwStrokeLimit_Override is case-sensitive

Point

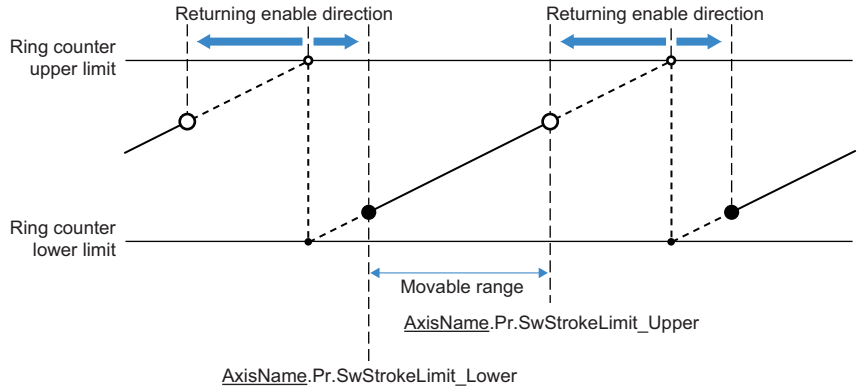
- In AxisName.Cd.SwStrokeLimit_Override, the software stroke limit can be temporarily disabled without changing the settings of AxisName.Pr.SwStrokeLimit_Upper and AxisName.Pr.SwStrokeLimit_Lower.
- In AxisName.Cd.SwStrokeLimit_Override, only the check in the returning enable direction can be disabled from outside the software stroke limit range. By disabling it only in the returning enable direction to the movable range, a motion to unexpected directions can be prevented.
- If the axis operation is started while the software stroke limit is disabled by AxisName.Cd.SwStrokeLimit_Override, that "the axis operation is started while the software stroke limit is disabled by AxisName.Cd.SwStrokeLimit_Override" is recorded in the event history.

■Returning enable direction to the movable range

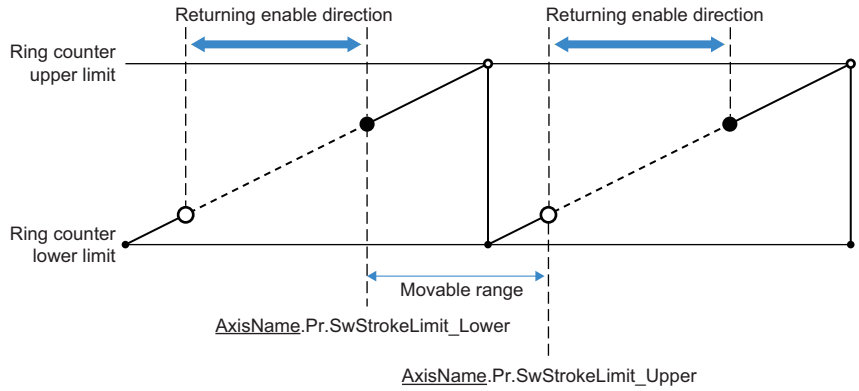
- When setting ONLY_INSIDE in AxisName.Cd.SwStrokeLimit_Override, movement in the returning enable direction from outside the software stroke limit range to the movable range is allowed.
- The returning enable direction from outside the software stroke limit range means the direction in which the axis can return to the movable range without exceeding the ring counter value.
- If the axis operates to the direction to exceed the ring counter value from outside the software stroke limit range, the error "Software Stroke Limit Over (start Position)" (error code: 1A01H) will be detected, and the axis will not start.
- Also, even if it is the returning enable direction, when an address which is outside the software stroke limit range exceeding the movable range is specified as the target position, the error "Software Stroke Limit Over (Target Position)" (error code: 1A00H) will be detected, and the axis will not start.
- To return to the movable range ignoring the returning enable direction, set DISABLE in AxisName.Cd.SwStrokeLimit_Override and carry out a move to the movable range.

■Returning enable direction when setting ONLY_INSIDE in AxisName.Cd.SwStrokeLimit_Override

- AxisName.Pr.SwStrokeLimit_Upper > AxisName.Pr.SwStrokeLimit_Lower



- AxisName.Pr.SwStrokeLimit_Upper < AxisName.Pr.SwStrokeLimit_Lower



Checking the software stroke limit enabled status

- In the monitor AxisName.Md.SwStrokeLimit_Override, the software stroke limit enabled/disabled status can be checked.
- AxisName.Md.SwStrokeLimit_Override is updated when turning the PLC READY [Y0] from OFF to ON and when starting the axis operation.
- Since AxisName.Cd.SwStrokeLimit_Override is restricted by switching of the software stroke limit enabled/disabled status, it performs import at start and updates AxisName.Md.SwStrokeLimit_Override.

AxisName.Pr.SwStrokeLimit_Target	AxisName.Cd.SwStrokeLimit_Override	AxisName.Md.SwStrokeLimit_Override
1: Invalid (Disabled)	—	DISABLE: Check disabled
1: SetPosition (Set position) 3: FeedMachinePosition (Feed machine position)	Other than the following: No disable request DISABLE: Check disabled ONLY_INSIDE: Check disabled only to inside the range direction	Display is blank: No disable request DISABLE: Check disabled ONLY_INSIDE: Check disabled only to inside the range direction

Specifications of the software stroke limit check

The software stroke limit check is carried out at executing the motion FB and during control.

The following table shows the check details of the software stroke limit.

Check details	Check timing	Processing when an error occurs
An error shall occur if the target address is outside the software stroke limit range.	When executing the motion FB which has the "target address"	The error "Software Stroke Limit Over (Target Position)" (error code: 1A00H) is output and the axis does not start.
An error shall occur if the start address is outside the software stroke limit range.	When executing the motion FB	The error "Software Stroke Limit Over (Start Position)" (error code: 1A01H) is output and the axis does not start.
An error shall occur if the current value is outside the software stroke limit range during control.	During control	The error "Software Stroke Limit Over (Forward Direction)" (error code: 1A03H) or the error "Software Stroke Limit Over (Reverse Direction)" (error code: 1A04H) is output and the stop processing is executed.*1

*1 The error occurs when detecting a command to the address outside the software stroke limit range, and the stop processing is executed. To stop within the software stroke limit range, set to immediately stop.

The checking details of the following function is different from the above.

- Homing control

For details, refer to the following.

☞ Page 198 HOMING

Precautions

- Homing must be executed beforehand for the software stroke limit function to function properly.
- If AxisName.Pr.SwStrokeLimit_Upper or AxisName.Pr.SwStrokeLimit_Lower is outside the setting range, the error "Out of Parameter Range" (error code: 1D80H) will be detected.
- During interpolation control, the software stroke limit check is executed to all current values of the interpolation axes. When an error occurs in any axis, all axes will not start.
- When the feed machine position is set as the setting value of AxisName.Pr.SwStrokeLimit_Target to the axis type which does not have AxisName.Md.FeedMachinePosition, the error "Out of Parameter Range" (error code: 1D80H) will be detected.

8.4 Hardware Stroke Limit

In the "hardware stroke limit function", limit switches are set at the upper/lower limit of the physical movable range, and the control is stopped by the input of a signal from the limit switch.
Damage to the machine can be prevented by stopping the control before the upper/lower limit of the physical movable range is reached.

⚠WARNING

- The negative logic (b-contact) is recommended for wiring the hardware stroke limit. If the positive logic (a-contact) is used, a serious accident may occur when disconnection or sensor failure occurs.
- To use the hardware stroke limit function, the add-on ExternalSignal must be valid. In case the add-on is not installed or invalid, the axis does not stop nor an error is not outputted even if the signal is inputted.

For details on the signals that can be used in the hardware stroke limit, refer to the following.
☞ Page 418 External Signal Selection

Relevant variables

Variable/Structure name	Name	Details
AxisName.PrConst.		
HwStrokeLimit_FlsSignal	Upper limit signal	Sets a signal using the upper limit signal (FLS). HwStrokeLimit_FlsSignal is the structure of SIGNAL_SELECT type. The specific setting and the operation of this signal are shown below. • StartIO (IO No.) Ignores the input value. • Target (Target) When the specification is not set, signal disabled is determined and the signal undetection status is always set. When unavailable data is set, the error "Out of Parameter Range (Axis)" (error code: 1D80H) is output. • Detection (Signal detection method) The specification is allowed at the following level detection only. When edge detection specification is set, the error "Out of Parameter Range (Axis)" (error code: 1D80H) is output. 0: HighLevel Detection at TRUE 1: LowLevel Detection at FALSE • CompensationTime (Compensation time) Ignores the input value. • FilterTime (Filter time) Setting range of the filter time is 0.0 to +5.0. When a value outside the range is set, the warning "Out of Filter Time Setting Range of Each Axis Signal Warning" (warning code: 0D24H) is output. The fetch timing of the label is at the system start, and the detection timing of the signal is the axis operation cycle. For details on SIGNAL_SELECT type, refer to the following. ☞ Page 418 SIGNAL_SELECT structure

Variable/Structure name		Name	Details
	HwStrokeLimit_RlsSignal	Lower limit signal	Sets a signal using the lower limit signal (RLS). HwStrokeLimit_RlsSignal is the structure of SIGNAL_SELECT type. The specific setting and the operation of this signal are shown below. • StartIO (IO No.) Ignores the input value. • Target (Target) When the specification is not set, signal disabled is determined and the signal undetection status is always set. When unavailable data is set, the error "Out of Parameter Range (Axis)" (error code: 1D80H) is output. • Detection (Signal detection method) The specification is allowed at the following level detection only. When edge detection specification is set, the error "Out of Parameter Range (Axis)" (error code: 1D80H) is output. 0: HighLevel Detection at TRUE 1: LowLevel Detection at FALSE • CompensationTime (Compensation time) Ignores the input value. • FilterTime (Filter time) Setting range of the filter time is 0.0 to +5.0. When a value outside the range is set, the warning "Out of Filter Time Setting Range of Each Axis Signal Warning" (warning code: 0D24H) is output. The fetch timing of the label is at the system start, and the detection timing of the signal is the axis operation cycle. For details on SIGNAL_SELECT type, refer to the following. ☞ Page 418 SIGNAL_SELECT structure
AxisName.Md.			
	HwStrokeLimit_FlsStatus	Upper limit signal status	Displays the detection status of the FLS signal. • TRUE: Detecting the FLS signal • FALSE: Not detecting the FLS signal Displays the detection state which reflects the signal detection method or the filter time but not the signal input state.
	HwStrokeLimit_RlsStatus	Lower limit signal status	Displays the detection status of the RLS signal. • TRUE: Detecting the RLS signal • FALSE: Not detecting the RLS signal Displays the detection state which reflects the signal detection method or the filter time but not the signal input state.
	HwStrokeLimit_Override	Hardware stroke limit override	Displays the acceptance status of the enabled/disabled switching command of the hardware stroke limit.
AxisName.Cd.			
	HwStrokeLimit_Override	Hardware stroke limit override	Switches the enabled/disabled of the hardware stroke limit.

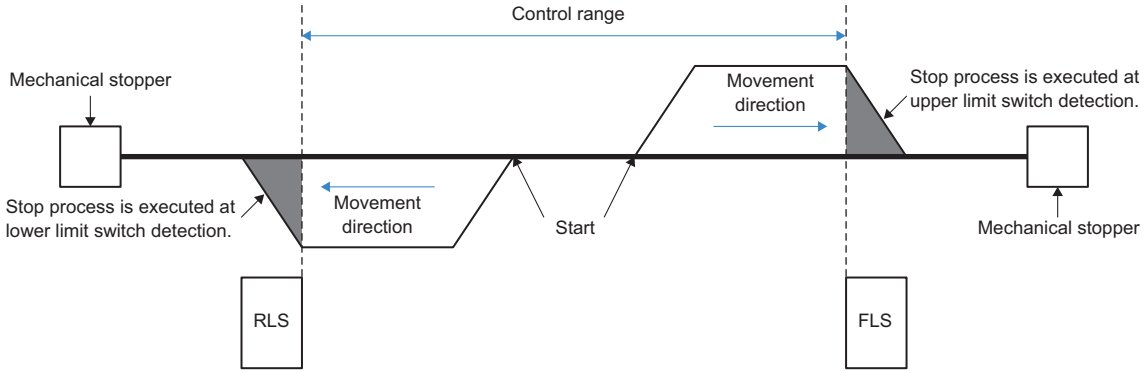
Hardware stroke limit check details

The check of hardware stroke limit range is carried out during executing and controlling the motion FB.

The following table shows the check details of the hardware stroke limit.

Check details	Check timing	Processing when an error occurs
Signal input from the FLS signal is detected at start.	At start	The error "FLS Signal Detection (at Start)" (error code: 1A2DH) is output and the axis does not start.
Signal input from the RLS signal is detected at start.	At start	The error "RLS Signal Detection (at Start)" (error code: 1A2EH) is output and the axis does not start.
Signal input from the FLS signal is detected during control.	During control	The error "FLS Signal Detection (Controlling)" (error code: 1A2FH) is output and the axis stops.*1
Signal input from the RLS signal is detected during control	During control	The error "RLS Signal Detection (Controlling)" (error code: 1A30H) is output and the axis stops.*1

*1 The axis stops according to the stop processing.



Point

While the limit switch is detected, the operation cannot be performed regardless of the detection direction/ movement direction.

Returning from outside the hardware stroke limit range

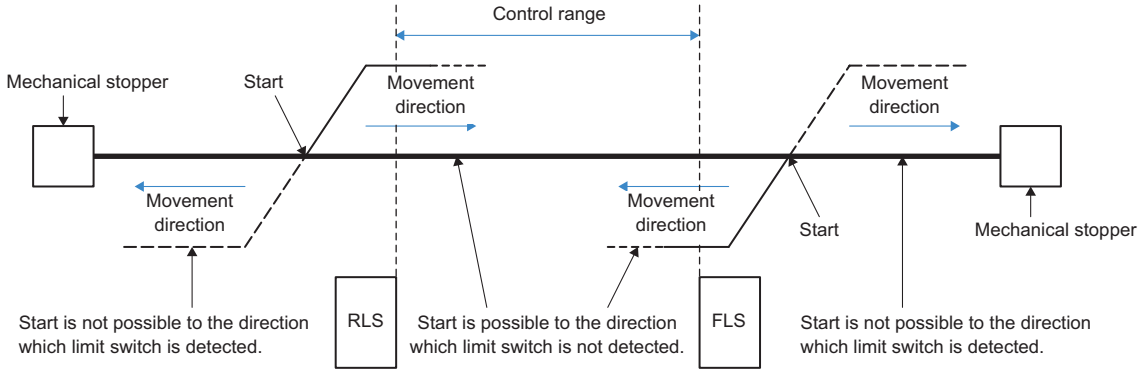
While the limit switch is detected, the operation cannot be performed regardless of the detection direction.
After setting ONLY_INSIDE or DISABLE in AxisName.Cd.HwStrokeLimit_Override, execute the axis operation to return to the controllable range.

Point

- When setting an item other than ONLY_INSIDE and DISABLE in AxisName.Cd.HwStrokeLimit_Override, the axis operates with "No disable request".
  AxisName.Cd.HwStrokeLimit_Override is case-sensitive.
- If the axis operation is started while the hardware stroke limit is disabled by AxisName.Cd.HwStrokeLimit_Override, that "the axis operation is started while the hardware stroke limit is disabled by AxisName.Cd.HwStrokeLimit_Override" is recorded in the event history.

When setting ONLY_INSIDE

The axis can start to the direction in which the limit switch has not been detected.



When setting DISABLE

The axis can start regardless of the limit switch detection status.

Confirm that the hardware stroke limit is disabled

- AxisName.Md.HwStrokeLimit_Override can confirm the state of hardware stroke limit check disabled.
- AxisName.Md.HwStrokeLimit_Override reads the setting value of AxisName.Cd.HwStrokeLimit_Override at executing the motion FB and then it updates as follows.

AxisName.Cd.HwStrokeLimit_Override	AxisName.Md.HwStrokeLimit_Override
DISABLE	DISABLE (Disabling hardware stroke limit check)
ONLY_INSIDE	ONLY_INSIDE (Disabling hardware stroke limit check, only the movement toward the range.)
Other than the above	No display (No disabling request. Hardware stroke limit check is enabled.)

Precautions

- Install the FLS in the direction in which the set position increases, and install the RLS in the direction in which the set position decreases. If inverting the install positions of the FLS/RLS, hardware stroke limit function cannot be operated properly.
- The FLS/RLS signal detected on the Motion module can be sent to the servo amplifier. For details on the setting method at connecting MR-J5(W)-G, refer to the following.
  ☞ Page 733 Using methods

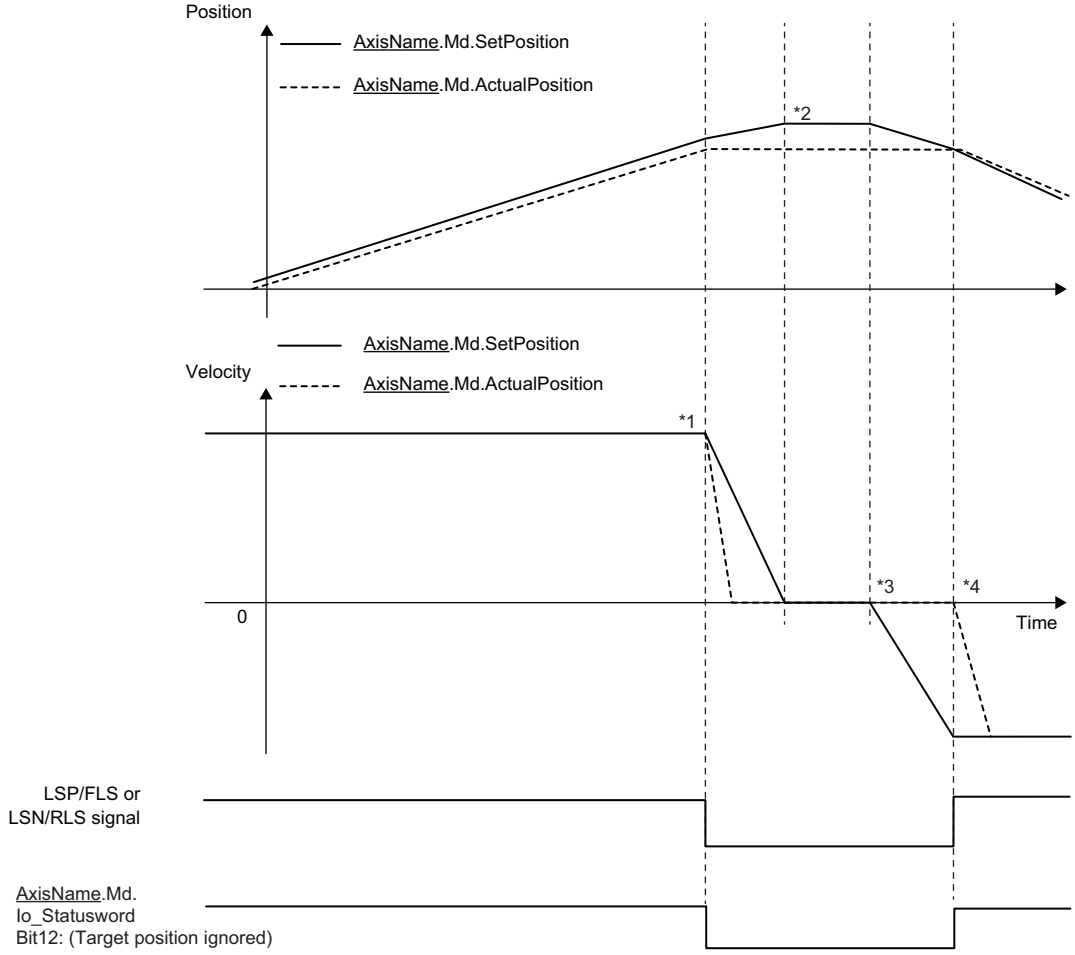
The Motion module sends the signals as "FALSE: Detecting the signal", "TRUE: Not detecting the signal" to the servo amplifier. Therefore, the servo amplifier side must be set by using "the negative logic (b-contact)" to receive the signal.
When labeling the data sent from the Motion module, the sent data is overwritten with the label value and the detection state cannot be sent to the servo amplifier.
To send the data to the servo amplifier, do not label the data.

- The LSP/LSN signal input to the servo amplifier when connecting to MR-J5(W)-G can be set as the hardware stroke limit signal. For details on the setting method, refer to the following.
  ☞ Page 733 Using methods
- Depending on the setting of "Limit switch enabled status selection" (PD41.2) of the servo parameter "Function selection D-4"(PD41) when connecting to MR-J5(W)-G, the operation differs at LSP/FLS or LSN/RLS signal detection.
- When "0: Limit switch always enabled", the drive unit side executes the stop processing, and then turns to servo-lock status (Statusword Bit12: OFF) and the drive unit will ignore the motion system command.
  The motion system can stop the command by enabling AxisName.Pr.StopOption_DriverTargetIgnored.
- When "1: Enabled only for homing mode" or the driver is in the Homing mode, the drive unit side will execute the stop processing. When the driver is in other than the Homing mode, the signal will be ignored so set it to stop by motion system using hardware stroke limit function.

When "Limit switch enabled status selection" (PD41.2) of servo parameter "Function selection D-4" (PD41)is "0: Limit switch always enabled", the stop will be executed ignoring the command from the motion system at limit switch detection. So when using MR-J5(W)-G, it is recommended toset "1: Enabled only for homing mode", and "Sensor input method selection" (PD41.3) to "1: Input from controller".
When setting "0: Limit switch always enabled", the command on the Motion module side needs to be stopped, be sure to enable AxisName.Pr.StopOption_DriverTargetIgnored. The following shows an operation example.

Ex.

Operation example when detecting a limit signal at the drive unit side



*1 When the LSP/FLS or LSN/RLS signal is detected, the stop process is executed by the drive unit. When AxisName.Pr.StopOption_DriverTargetIgnored is enabled, the motion system will detect the error "Driver Command Discard Detection" (error code: 1AE6H) and execute the command stop. For details on the setting, refer to the following.
☞ Page 162 Stop

*2 After completion of the stop process on the motion system side, it stops with the status where AxisName.Md.ActualPosition deviates from AxisName.Md.SetPosition. (The position command from the motion system to the outside of the stroke limit is ignored in the drive unit.)

*3 After stopped and when axis control is performed to the direction toward the inside of the stroke limit, the position command (the set position and feed machine position) and set velocity of the motion system are updated. However, the servo motor does not operate.

*4 When the position command from the motion system becomes "the command position where the LSP/FLS or LSN/RLS signal is detected", the servo motor starts the operation to the inside of the stroke limit.

Point

- To stop holding the relation between the master axis and the slave axis of the axes group and the synchronous control when detecting the stroke limit of the drive unit, set "Limit switch enabled status selection" (PD41.2) of the servo parameter "Function selection D-4" (PD41) to "1: Enabled only for homing mode".
- When the signal is turned ON (limit signal OFF) in the status where the actual position deviates from the position command with the stop by the stroke limit detection of the drive unit, the motor makes rapid movement to the position command of the motion system. Restore the deviation between the actual position and the position command by axis control to the inside of the stroke limit. (Set the servo OFF status, so that restoring the position command to the actual position by follow up is possible. After the restoration, set the servo ON again and move the motor to the inside of the stroke limit with the axis control.)
- When the command position of the motion system stops at the stroke limit side closer than the actual position by the stroke limit detection of the driver unit and the axis control is performed to the inside of the stroke limit, the operation of the actual position follows the command position from the motion system.

9 RELEVANT FUNCTIONS TO VELOCITY

9.1 Acceleration/deceleration Processing Function

The acceleration/deceleration processing function adjusts the acceleration/deceleration of each motion control to the acceleration/deceleration curve suitable for device.

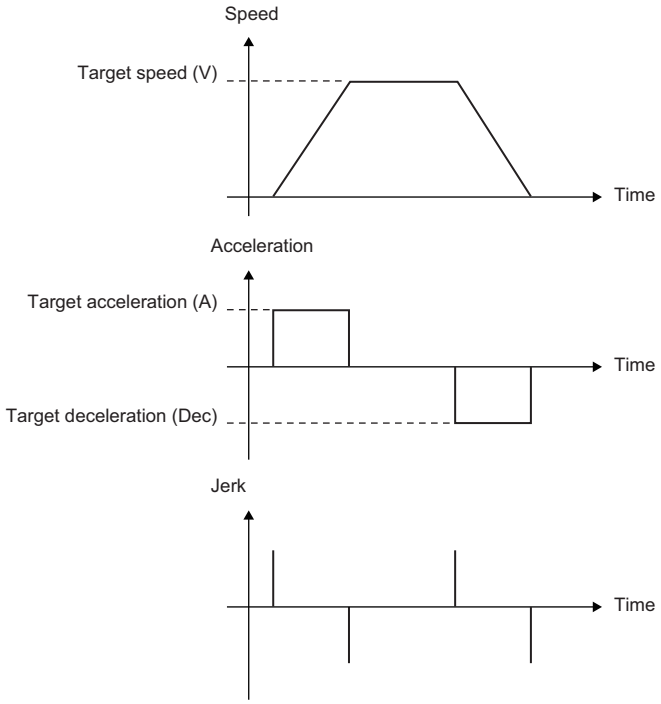
Acceleration/deceleration method

The following methods can be selected for the acceleration/deceleration method. When a value outside the range is set in the acceleration/deceleration method, the error "Out of Acceleration/Deceleration Method Range" (error code: 1AA9H) will occur and the axis will not start.

Acceleration/deceleration method	Description
Acceleration/deceleration specification method (Default)	An acceleration/deceleration method using the acceleration, the deceleration, and the jerk specified in the FB.
Acceleration/deceleration time-fixed method	An acceleration/deceleration method using the acceleration/deceleration time specified in the FB regardless of the speed.

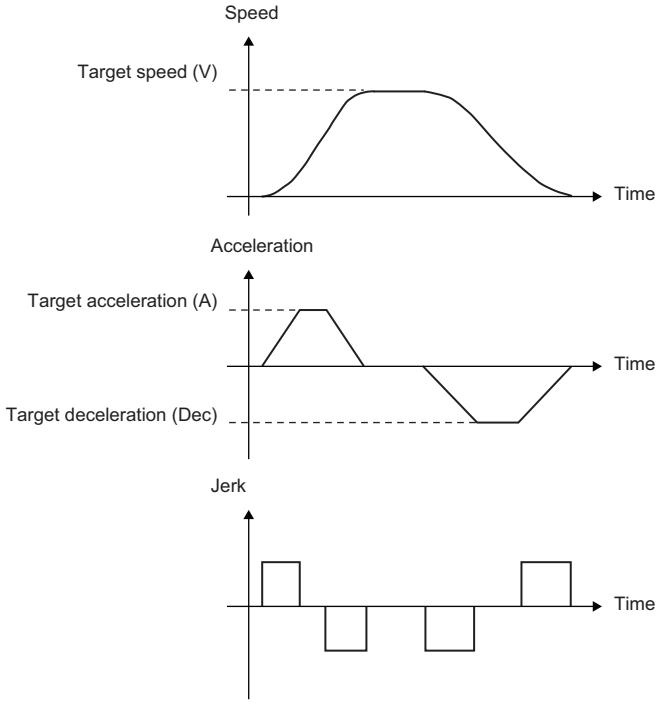
■Trapezoidal acceleration/deceleration

When "0.0" is set in the jerk, the curve is called the trapezoidal acceleration/deceleration. The speed shows a trapezoidal waveform.



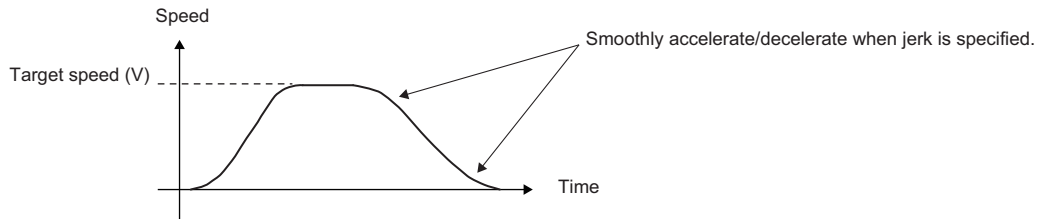
■Jerk acceleration/deceleration

When a value other than "0.0" is set in the jerk, the curve is called the jerk acceleration/deceleration. The speed shows a S-shaped waveform.



Jerk

Jerk is the temporal change ratio of the acceleration or deceleration. The unit is "command unit (U)/s^3". When the jerk is set, since the speed waveform shows a S-shape and the axis can smoothly accelerate and decelerate at the start and the end of acceleration and deceleration, it can reduce the burden on the motor and impact on the machine.



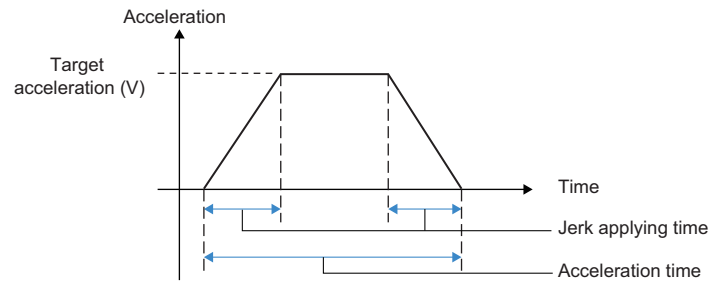
• Jerk applying time/Jerk applying ratio

The sum of the time required to reach the target acceleration after startup and the time from the target acceleration to the acceleration 0 at the end of acceleration is called the jerk applying time. The ratio of jerk applying time to acceleration (deceleration) time is called the jerk applying ratio.

The jerk applying time and the jerk applying ratio can be calculated with the following formula.

Jerk applying time = 2 × Target acceleration ÷ Jerk

Jerk applying ratio = (Jerk applying time ÷ Acceleration time) × 100 [%]



Relevant variables

Variable/Structure name	Name	Details
AxisName.Pr.		
VelocityLimit_Positive	Positive direction speed limit value	Sets the positive direction speed limit value. For details, refer to the following. ☞ Page 356 Speed Limit
VelocityLimit_Negative	Negative direction speed limit value	Sets the negative direction speed limit value. For details, refer to the following. ☞ Page 356 Speed Limit
VelocityLimit_OverOperation	Operation setting at speed limit value exceeded	Sets the operation at speed limit value exceeded. For details, refer to the following. ☞ Page 356 Speed Limit
AccelerationLimit	Acceleration limit value	Sets the acceleration limit value. Range: 0.0000, positive numbers from 0.0001 to 2147483647.0 U/s^2 • When "1: mcFixedTime" is set in the acceleration/deceleration method, it doesn't limit the acceleration. • When "0.0000" is set, it does not limit the acceleration.
DecelerationLimit	Deceleration limit value	Sets the deceleration limit value. Range: 0.0000, positive numbers from 0.0001 to 2147483647.0 U/s^2 • When "1: mcFixedTime" is set in the acceleration/deceleration method, it doesn't limit the the deceleration. • When "0.0000" is set, it does not limit the deceleration.
JerkLimit	Jerk limit value	Sets the jerk limit value. Range: 0.0000, positive numbers from 0.0001 to 2147483647.0 U/s^3 • When "0.0000" is set, it does not limit the jerk.
AccelerationZeroBehavior	Operation selection at start acceleration/deceleration 0	Selects behavior when "0.0" is set as the acceleration, the deceleration, or the acceleration/deceleration time at start. -1: ACCError (Error (Not started)) 1: MaximumAcceleration (Maximum acceleration/deceleration)
OverrunOperation	Operation setting at overrun	Automatically adjusts the deceleration to stop at the target position. 1: ImmediateStop (Immediately stop) 2: KeepCurrentAcc (Keep current acceleration/deceleration)
AxesGroupName.Pr.		
VelocityLimit	Speed limit value	Sets the speed limit value. For details, refer to the following. ☞ Page 356 Speed Limit
AccelerationLimit	Acceleration limit value	Sets the acceleration limit value. Range: 0.0000, positive numbers from 0.0001 to 2147483647.0 U/s^2 • When "1: mcFixedTime" is set in the acceleration/deceleration method, it limits the acceleration calculated from the specified acceleration/deceleration time. • When "0.0000" is set, it does not limit the acceleration.
DecelerationLimit	Deceleration limit value	Sets the deceleration limit value. Range: 0.0000, positive numbers from 0.0001 to 2147483647.0 U/s^2 • When "1: mcFixedTime" is set in the acceleration/deceleration method, it limits the deceleration calculated from the specified acceleration/deceleration time. • When "0.0000" is set, it does not limit the acceleration.
JerkLimit	Jerk limit value	Sets the jerk limit value. Range: 0.0000, positive numbers from 0.0001 to 2147483647.0 U/s^3 • When "0.0000" is set, it does not limit the acceleration.
AccelerationZeroBehavior	Operation selection at start acceleration/deceleration 0	Selects the operation when "0.0" is set as the acceleration, the deceleration, or the acceleration/deceleration time at start. -1: ACCError (Error (Not started)) 1: MaximumAcceleration (Maximum acceleration/deceleration)
OverrunOperation	Operation setting at overrun	Automatically adjusts the deceleration to stop at the target position. 1: ImmediateStop (Immediately stop)
AxisName.Md.		

Variable/Structure name	Name	Details
CommandedAcceleration	Commanded acceleration	Stores the acceleration specified by users. Unit: U/s^2 • When "1: mcFixedTime" is set in the acceleration/deceleration method, it stores the specified acceleration/deceleration time. Unit: s • During axes group operation, stores "0.0" in AxisName.Md.CommandedAcceleration of the configuration axis.
CommandedDeceleration	Commanded deceleration	Stores the deceleration specified by users. Unit: U/s^2 • When "1: mcFixedTime" is set in the acceleration/deceleration method, it is an illegal value. • During axes group operation, stores "0.0" in AxisName.Md.CommandedDeceleration of the configuration axis.
TargetAcceleration	Target acceleration	Stores the acceleration that is maximum during acceleration. Unit: U/s^2 • During axes group operation, stores "0.0" in AxisName.Md.TargetAcceleration of the configuration axis.
TargetDeceleration	Target deceleration	Stores the deceleration that is maximum during deceleration. Unit: U/s^2 • During axes group operation, stores "0.0" in AxisName.Md.TargetDeceleration of the configuration axis.
SetAcceleration	Set acceleration	• Stores the command output acceleration. It is the acceleration/deceleration speed calculated from the difference of set velocity.[*1] Unit: U/s^2 • It indicates accelerating or decelerating by a sign. 0.0: During stop or fixed-speed. "0.0" is stored when the target velocity is reached. Plus sign: During acceleration Minus sign: During deceleration • During axes group operation, stores the set acceleration of the configuration axis.
CommandedJerk	Commanded jerk	Stores the jerk specified by users. Unit: U/s^3 • During axes group operation, stores "0.0" in AxisName.Md.CommandedJerk of the configuration axis.
InVelocity	Target velocity reached	TRUE: Reached FALSE: Not reached • During axes group operation, stores FALSE in AxisName.Md.InVelocity of the configuration axis.
AutoDeceleration	Automatically decelerating	TRUE: During auto deceleration FALSE: Not during auto deceleration • TRUE is stored while the auto deceleration processing is executed. • When the multple start is executed, 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 configuration axis.
AxesGroupName.Md.		
CommandedAcceleration	Commanded acceleration	Stores the acceleration specified by users. Unit: U/s^2 • When "1: mcFixedTime" is set in the acceleration/deceleration method, it stores the acceleration/deceleration time.
CommandedDeceleration	Commanded deceleration	Stores the deceleration specified by users. Unit: U/s^2 • When "1: mcFixedTime" is set in the acceleration/deceleration method, it is an illegal value.
TargetAcceleration	Target acceleration	Stores the acceleration that is maximum during acceleration. Unit: U/s^2
TargetDeceleration	Target deceleration	Stores the deceleration that is maximum during deceleration. Unit: U/s^2

Variable/Structure name		Name	Details
	SetAcceleration	Set current acceleration	• Stores the command output acceleration. It is the acceleration/deceleration speed calculated from the difference of set velocity.*1 Unit: U/s^2 • It indicates accelerating or decelerating by a sign. 0.0: During stop or fixed-speed. "0.0" is stored when the target velocity is reached. Plus sign: During acceleration Minus sign: During deceleration
	CommandedJerk	Commanded jerk	Stores the jerk specified by users. Unit: U/s^3
	InVelocity	Target velocity reached	TRUE: Reached FALSE: Not reached
	AutoDeceleration	Automatically decelerating	TRUE: Automatically decelerating FALSE: Not automatically decelerating • "TRUE" is stored while the auto deceleration processing is executed. • When the multple start is executed, 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.

*1 The Values to be stored include the error, because the floating point type error occurs.

Settings of motion FB

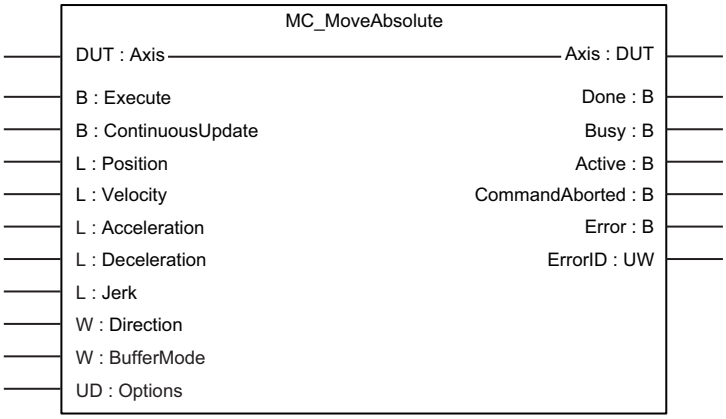
The acceleration/deceleration method can be selected in the UD: Options input of the motion FB.

Set the function options by specifying the bit. The details of bit and the functions are shown below.

Bit	Function description
0 to 2	Acceleration/deceleration method setting 0: mcAccDec (Acceleration/deceleration specification method) 1: mcFixedTime (Acceleration/deceleration time-fixed method) For details, refer to the following. ☞ Page 330 Acceleration/deceleration Processing Function
3 to 15	Empty (Specify "0")*1
16 to 31	The function differs depending on the FB.

*1 If specifying other than "0", the error "Out of Options Range" (error code: 1A4EH) occurs, and the function does not start.

Whether a method can be selected or not differs depending on the FB. For details, refer to the speifications of each FB.



■Input variables to use

Variable/Structure name	Name	Details
Velocity	Velocity	Specifies the target velocity in the FB. Unit: For details, refer to the following. ☞ Page 79 Technical Units Range: For details, refer to the following. ☞ Page 87 Velocity set range
Acceleration	Acceleration and acceleration/ deceleration time	Specifies the acceleration and the acceleration/deceleration time in the FB. • When the acceleration/deceleration method is "0: mcAccDec" Unit: Specify with U/s^2 Range: 0.0000*1, positive numbers from 0.0001 to 2147483647.0 U/s^2 • When the acceleration/deceleration method is "1: mcFixedTime" Unit: Specify with s Range: 0.000000*1, positive numbers from 0.000001 to 8400.0 s
Deceleration	Deceleration	Specifies the deceleration in the FB. • When the acceleration/deceleration method is "0: mcAccDec" Unit: Specify with U/s^2 Range: 0.0000*1, positive numbers from 0.0001 to 2147483647.0 U/s^2 • When the acceleration/deceleration method is "1: mcFixedTime" Not used.
Jerk	Jerk/Jerk ratio	Specifies the jerk in the FB. • When the acceleration/deceleration method is "0: mcAccDec" Unit: Specify with U/s^3 Range: 0.0000, positive numbers from 0.0001 to 2147483647.0 U/s^3 • When the acceleration/deceleration method is "1: mcFixedTime" Not used.
Options	Options	Specifies the option information in the FB.

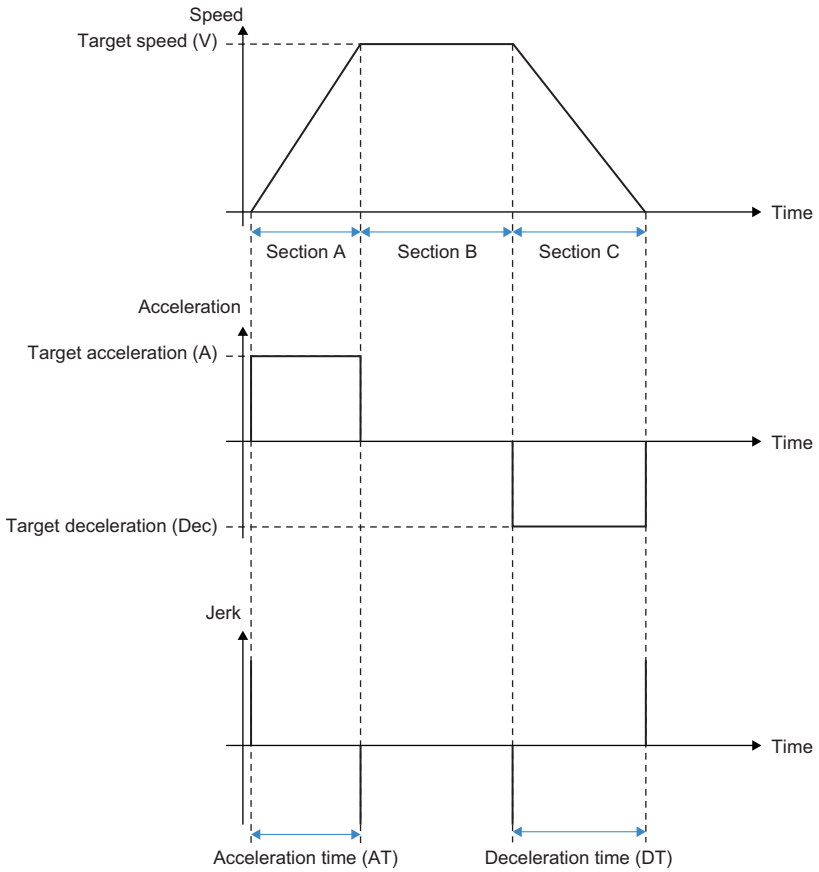
*1 Operation varies according to AxisName.Pr.AccelerationZeroBehavior at start. A change is not accepted during a acceleration change and a deceleration change.

Acceleration/deceleration method

Acceleration/deceleration specification method

When "0: mcAccDec" is selected in the "acceleration/deceleration method setting" (Options bit0 to 2) of the FB, the acceleration, the deceleration, and the jerk can be set.

■Trapezoidal acceleration/deceleration



○: Operation possible, ×: Operation not possible

Section		Process description	Operation	
			Acceleration	Deceleration
A	Acceleration section	Accelerates at the target acceleration.	○	×
B	Fixed speed section	Controls at the target velocity. (Acceleration = 0.0)	×	×
C	Deceleration section	Decelerates at the target deceleration.	×	○

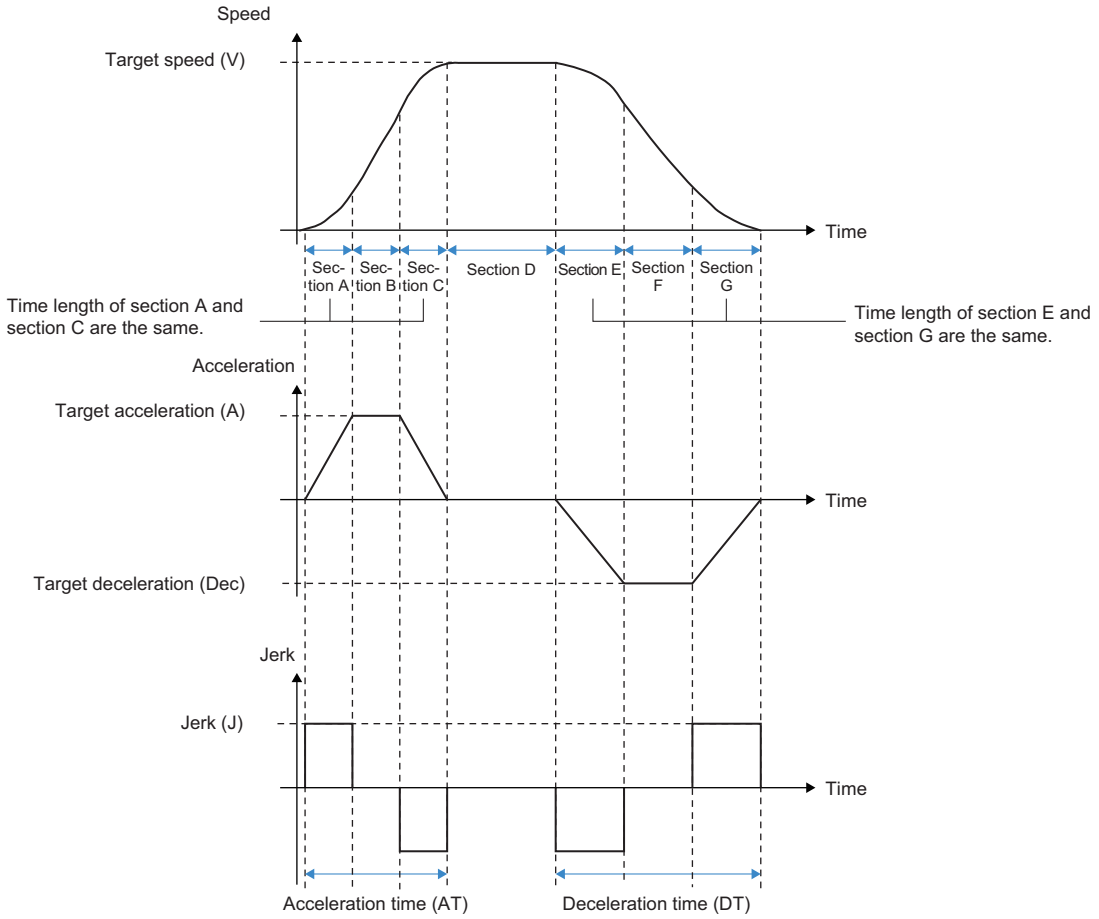
• Acceleration/deceleration time parameter calculation

○: Operation possible, ×: Operation not possible

Type	Item	Abbreviation	Description	Calculation formula	Operation	
					Acceleration	Deceleration
Specified value	Target velocity	V	—	—	○	○
	Target acceleration	A	—	—	○	×
	Target deceleration	Dec	—	—	×	○
Calculated value	Acceleration section time	AT	Time from the speed 0.0 to the target velocity	V ÷ A*1	○	×
	Deceleration section time	DT	Time to decelerate from the target velocity to the speed 0.0	V ÷ Dec*1	×	○

*1 This is the calculation formula when unit of V is [U/s]. When other than [U/s], calculate by converting into [U/s].

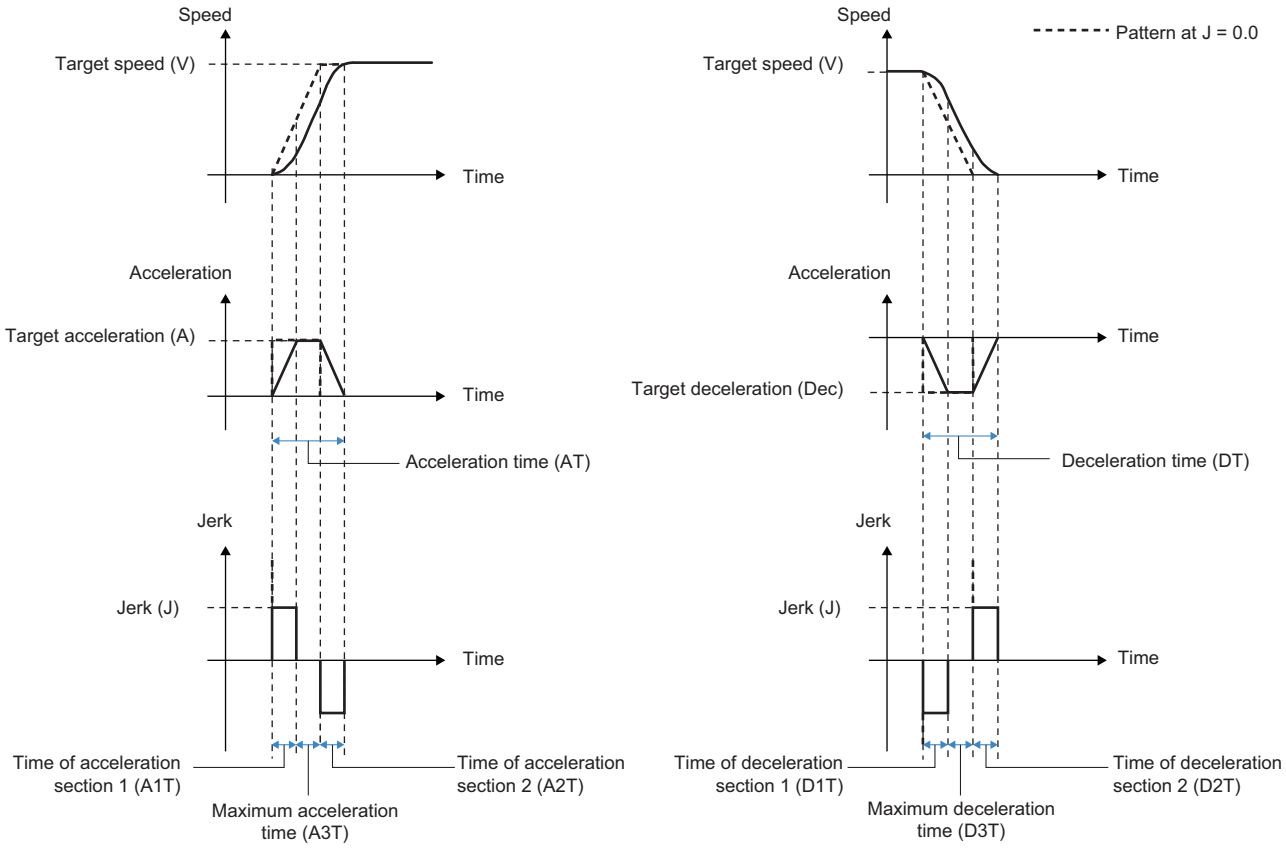
■Jerk acceleration/deceleration



○: Operation possible, ×: Operation not possible

Section		Process description	Operation	
			Acceleration	Deceleration
A	Acceleration section 1	Accelerates by changing (increasing) the acceleration with the specified jerk from acceleration start to target acceleration.	○	×
B	Maximum acceleration section	Accelerates at the target acceleration.	○	×
C	Acceleration section 2	At the end of acceleration, accelerates by changing (decreasing) the acceleration by jerk specified from the target acceleration to acceleration 0.	○	×
D	Fixed speed section	Controls at the target velocity. (Acceleration = 0.0)	×	×
E	Deceleration section 1	Decelerates by changing (increasing) the deceleration with the specified jerk from deceleration start to the target deceleration.	×	○
F	Maximum deceleration section	Decelerates at the target deceleration.	×	○
G	Deceleration section 2	At the end of deceleration, decelerates by changing (decreasing) the deceleration by jerk specified from the target deceleration to deceleration 0.	×	○

• Acceleration/deceleration time parameter calculation



○: Operation possible, ×: Operation not possible

Section	Item	Abbreviation	Description	Calculation formula	Operation	
					Acceleration	Deceleration
Specified value	Target velocity	V	—	—	○	○
	Target acceleration*3	A	—	—	○	×
	Target deceleration*3	Dec	—	—	×	○
	Jerk	J	—	—	○	○
Calculated value	Acceleration section 1 time	A1T	Time obtained by dividing the jerk applying time of the acceleration section by 2	A ÷ J	○	×
	Acceleration section 2 time	A2T	Time obtained by dividing the jerk applying time of the acceleration section by 2	A ÷ J	○	×
	Maximum acceleration section time	A3T	Time of the maximum acceleration section.	(V ÷ A) - (A ÷ J)*2	○	×
	Acceleration time*1	AT	Time from speed 0.0 to the target velocity	A1T + A2T + A3T = (V ÷ A) + (A ÷ J) *2	○	×
	Deceleration section 1 time	D1T	Time obtained by dividing the jerk applying time of the deceleration section by 2	Dec ÷ J	×	○
	Deceleration section 2 time	D2T	Time obtained by dividing the jerk applying time of the deceleration section by 2	Dec ÷ J	×	○
	Maximum deceleration section time	D3T	Time of the maximum deceleration section.	(V ÷ Dec) - (Dec ÷ J)*2	×	○
	Deceleration time*1	DT	Time to decelerate from the target velocity to speed 0.0	D1T + D2T + D3T = (V ÷ Dec) + (Dec ÷ J)*2	×	○

*1 Acceleration time and deceleration time may be limited. For details, refer to the following.
☞ Page 347 Limit of acceleration time/deceleration time

*2 This is the calculation formula when unit of V is [/s]. When other than [U/s], calculate by converting into [U/s].

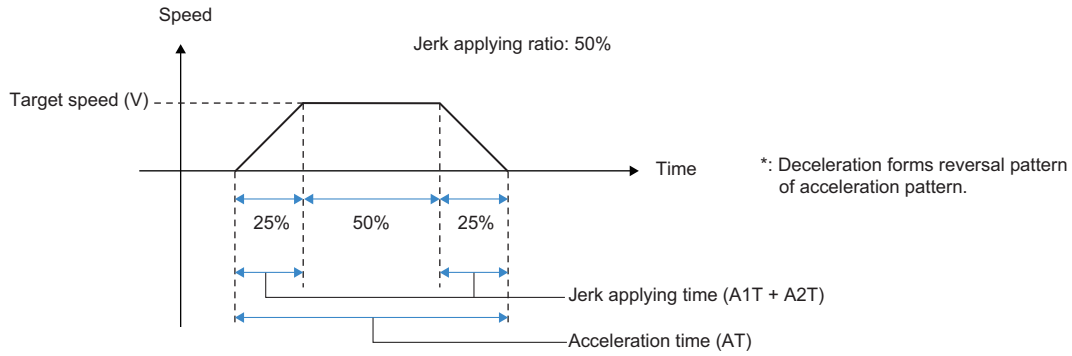
*3 Target acceleration/target deceleration may be adjusted by acceleration/deceleration waveform adjustment. For details, refer to the following.
☞ Page 343 Acceleration/deceleration waveform adjustment when specifying jerk

• Jerk calculation

When the speed and the acceleration of operation are fixed, the jerk can be calculated with the following formula by specifying the jerk applying ratio.

$$\text{Jerk (J)} = \frac{(2 - \text{Jerk applying ratio}) \times (\text{the square of acceleration (A)})}{\text{Jerk applying ratio} \times \text{Speed (V)}}$$

The acceleration section jerk applying ratio is the ratio of the jerk applying time to the acceleration time. The jerk applying time is the sum of the acceleration section 1 time and the acceleration section 2 time.



Ex.

A calculation example of jerk is shown below.

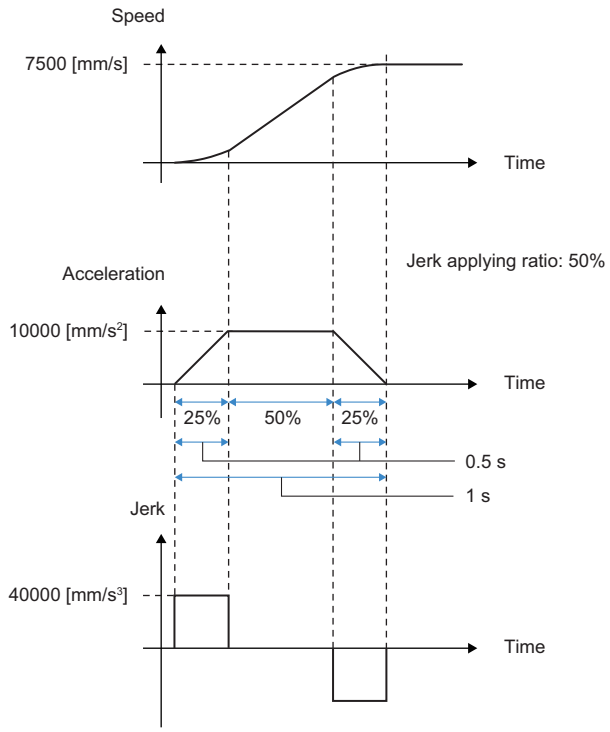
The jerk when the speed is 7500 mm/s, the acceleration is 10000 mm/s^2, and the jerk applying ratio is 0.5 (50%) can be calculated with the following formula.

$$\text{Jerk (J)} = \frac{(2 - 0.5) \times 10000^2}{0.5 \times 7500} = 40000 \text{ [mm/s}^3\text{]}$$

The acceleration time can be calculated when the speed, the acceleration, and the jerk are fixed. For details, refer to the following.

☞ Page 337 Jerk acceleration/deceleration

$$\text{Acceleration time (AT)} = \frac{\text{Speed (V)}}{\text{Acceleration time (AT)}} + \frac{\text{Acceleration time (AT)}}{\text{Jerk (J)}} = \frac{7500}{10000} + \frac{10000}{40000} = 1 \text{ [s]}$$



When the jerk applying ratio is big or the jerk is small, the acceleration time (deceleration time) becomes longer than that of the trapezoidal acceleration/deceleration. When the jerk applying ratio is 100%, the acceleration time (deceleration time) becomes doubled compared with the trapezoidal acceleration/deceleration. When setting the jerk, pay attention to the acceleration time (deceleration time).

Point

Since the speed calculation is based on a given jerk, when the acceleration (deceleration) at FB execution and the target velocity are different from the acceleration (deceleration) used at calculating the jerk, the final acceleration time/deceleration time of control are calculated according to the calculation formula of the acceleration/deceleration time parameter.

If the jerk applying ratio exceeds 100% due to the acceleration (deceleration) or the target velocity at FB execution, the target speed may be reached before the target acceleration is reached.

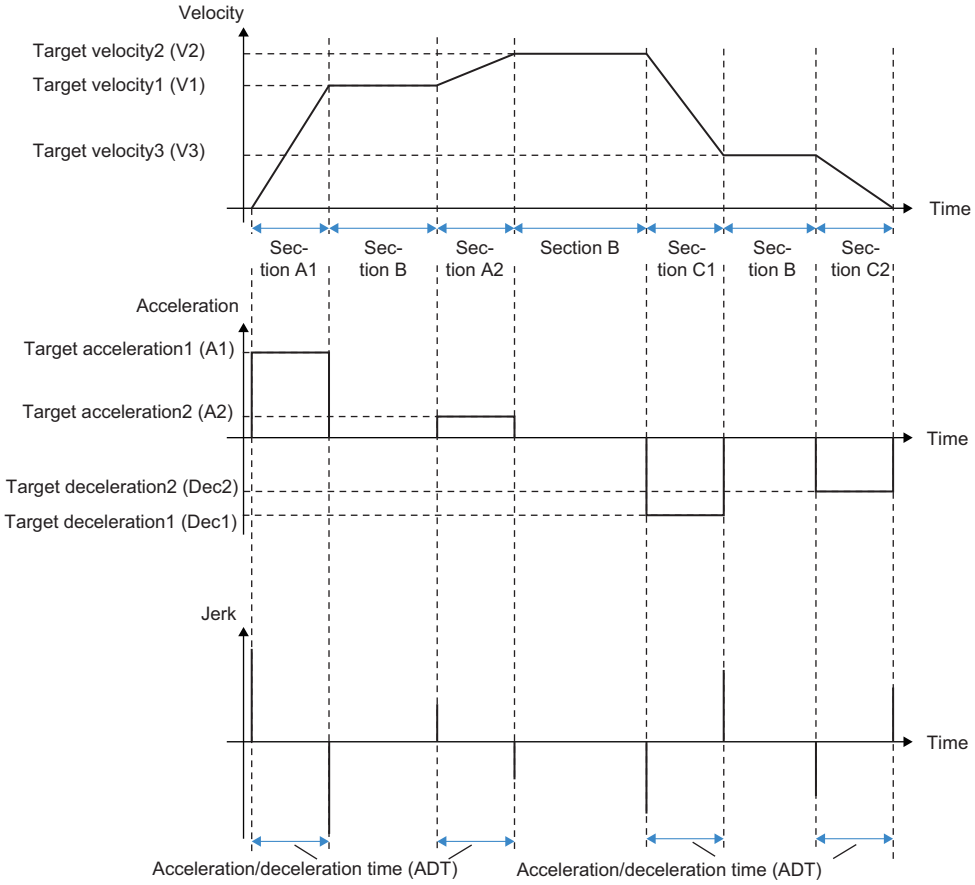
For details, refer to the following.

☞ Page 343 Acceleration/deceleration waveform adjustment when specifying jerk

Acceleration/deceleration time-fixed method

When "1: mcFixedTime" is selected in the "acceleration/deceleration method setting" (Options bit0 to 2) of the FB, the acceleration/deceleration time can be set in Acceleration. Deceleration and jerk are not used.

Calculates acceleration and deceleration from the acceleration/deceleration time at acceleration/deceleration. When a value smaller than one operation cycle other than 0.0000 is set as the acceleration/deceleration time, the target velocity is reached within one operation cycle.



○: Operation possible, ×: Operation not possible

Section		Process description	Operation	
			Acceleration	Deceleration
A	Acceleration section	Accelerates the acceleration time from the set velocity to the target velocity to be the acceleration/deceleration time.	○	×
B	Fixed speed section	Controls at the target velocity. (Acceleration = 0.0)	×	×
C	Deceleration section	Decelerates the deceleration time from the set velocity to the target velocity to be the acceleration/deceleration time.	×	○

• Acceleration/deceleration parameter calculation

○: Operation possible, ×: Operation impossible

Section	Item	Abbreviation	Description	Calculation formula	Operation	
					Acceleration	Deceleration
Specified value	Target velocity	V	—	—	○	○
	Acceleration/deceleration time*4	ADT	—	—	○	○
Calculated value	Current velocity	Vnow	The set velocity at target acceleration and target deceleration calculation	—	○	○
	Target acceleration*1	A	Acceleration from the currentvelocity to the target velocity*2	(V - Vnow)÷ ADT*3	○	×
	Target deceleration*1	Dec	Deceleration to decelerate from the current velocity to the target velocity*2	(V - Vnow)÷ ADT*3	×	○

*1 It may exceed the acceleration limit value or the deceleration limit value. For details, refer to the following. If it exceeds the acceleration/deceleration range, it will be the upper limit value of the acceleration/deceleration.
☞ Page 346 Acceleration limit value/deceleration limit value

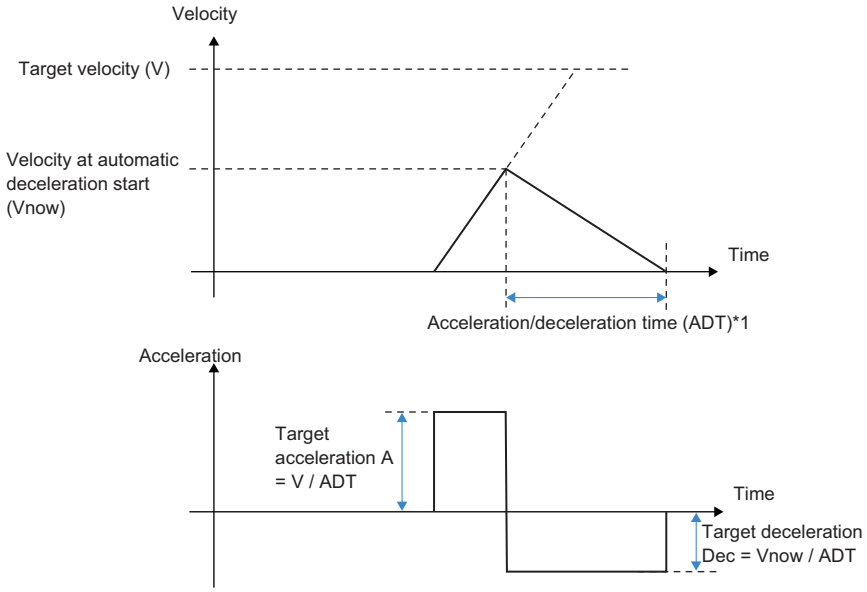
*2 The target velocity will be 0 at stop.

*3 This is the calculation formula when unit of V is [/s]. When other than [U/s], calculate by converting into [U/s].

*4 The deceleration time at automatic deceleration may be longer than the specified acceleration/deceleration time (ADT) due to the compensation of the remaining distance.

When the movement amount which cannot be accelerated up to the target velocity is specified at positioning operation, deceleration stop starts to the target position during acceleration by automatic deceleration. In this case, deceleration is recalculated with the velocity at automatic deceleration start.

Set acceleration/deceleration specification method to set time-fixed acceleration/deceleration at acceleration/deceleration.



Acceleration/deceleration waveform adjustment when specifying jerk

When the jerk acceleration/deceleration (J ≠ 0.0) is specified, the acceleration waveform and the deceleration waveform are adjusted whether the values of the target velocity (V), the target acceleration (A), the target deceleration (Dec), and the jerk (J) satisfy the following condition or not.

For acceleration: $V \geq \frac{A^2}{J}$, For deceleration: $V \geq \frac{Dec^2}{J}$

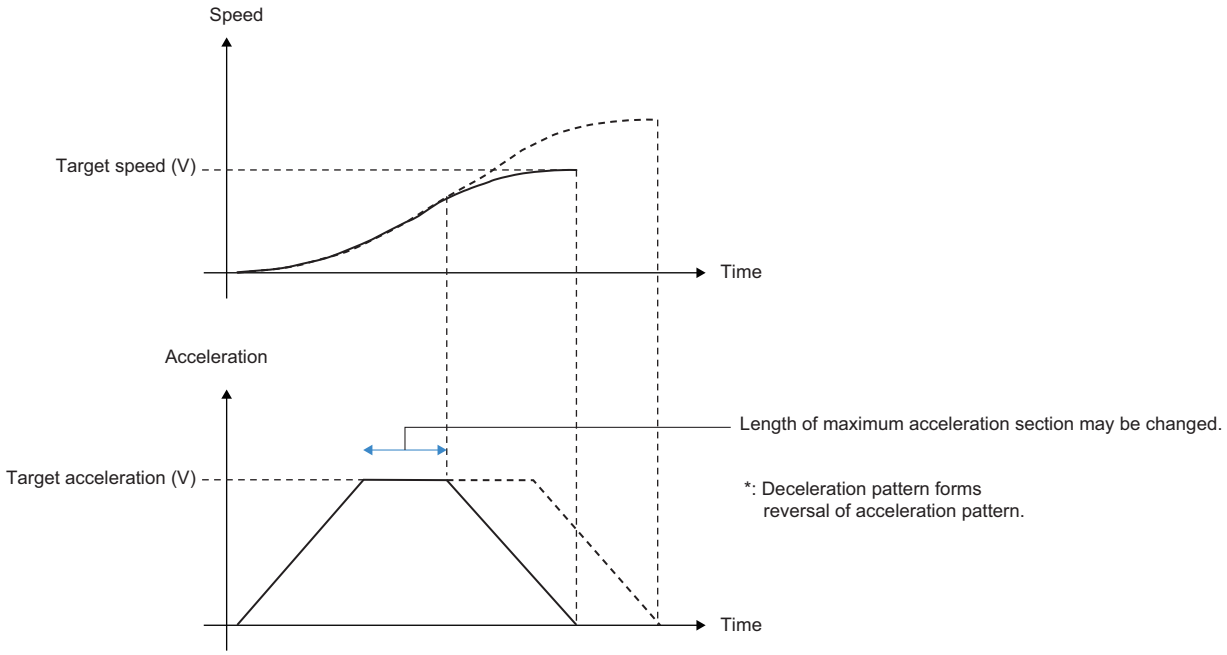
When the condition is satisfied

The target acceleration (deceleration) can be reached.

The maximum acceleration section time and the maximum deceleration section time are adjusted by the speed, the acceleration, the deceleration, and the jerk, and the waveform changes to reach the target velocity at the end of acceleration.

For the acceleration time and the deceleration time, refer to the following.

☞ Page 336 Acceleration/deceleration method

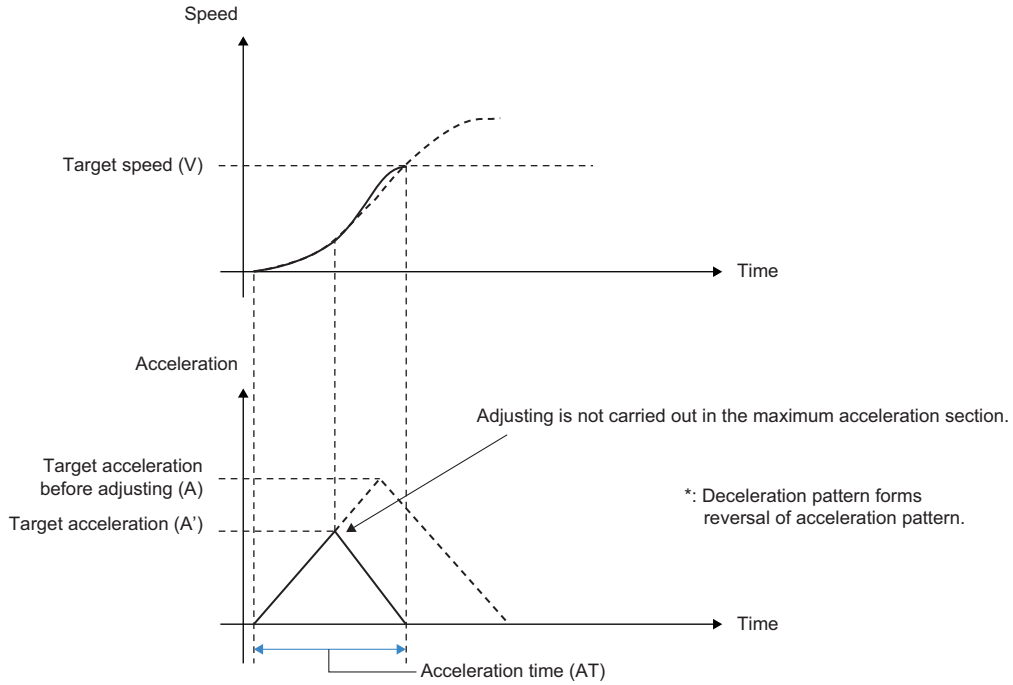


When the condition is not satisfied

When accelerating to the target acceleration, the velocity exceeds the target velocity during acceleration. Therefore, the acceleration is adjusted to reach the target velocity at the end of acceleration. (The velocity does not reach the specified target acceleration (deceleration).)

When the acceleration is adjusted, there is no maximum acceleration section, and the acceleration waveform becomes a triangle. Also, the same adjustment is applied to the deceleration.

Term	Abbreviation	Description
Target acceleration before adjustment	A	Target acceleration before acceleration adjustment
Target acceleration	A'	Target acceleration after acceleration adjustment
Target deceleration before adjustment	Dec	Target deceleration before deceleration adjustment
Target deceleration	Dec'	Target deceleration after deceleration adjustment



The target acceleration (A') and the target deceleration (Dec') can be calculated with the following formula.

$A' = \sqrt{V \times J}$, $Dec' = \sqrt{V \times J}$

The target acceleration (A') and the target deceleration (Dec') after adjustment become smaller than the target acceleration (A) and the target deceleration (Dec) before adjustment.

The acceleration time and the deceleration time can be calculated with the following formula.

Acceleration time (AT) = (V ÷ A') + (A' ÷ J)

Deceleration time (DT) = (V ÷ Dec') + (Dec' ÷ J)

Operation when specifying acceleration/deceleration 0

When "0.0" is specified for the acceleration/deceleration, the operation differs at startup and the rest.
When "1: mcFixedTime" is selected in the "acceleration/deceleration method setting" (Options bit0 to 2) of the FB, the operation is the operation when "0.0" is specified for the acceleration/deceleration time. The operation is the same as specifying "0.0" for acceleration/deceleration.

Specifying 0.0 at start

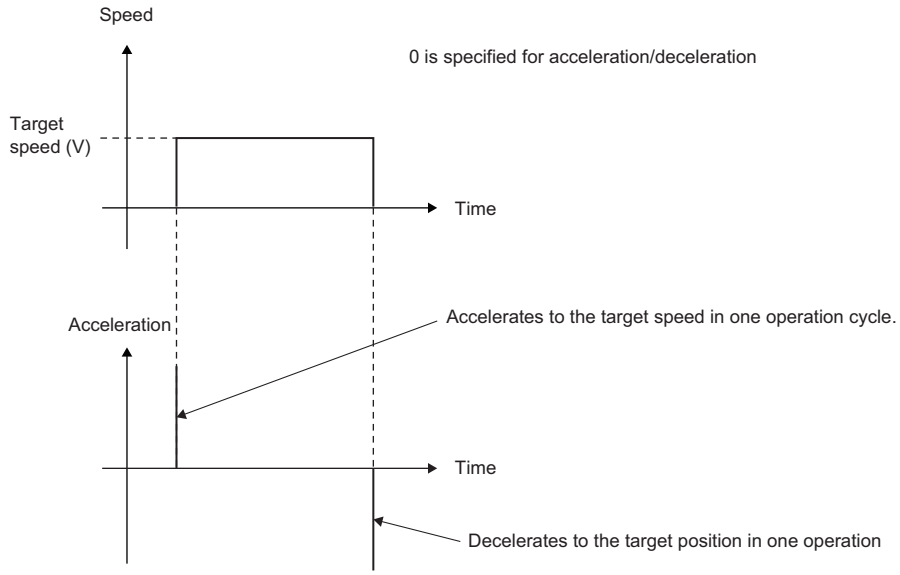
Select the operation when "0.0" is specified for the acceleration and the deceleration in AxisName.Pr.AccelerationZeroBehavior.

■When selecting "-1: ACCError"

The error "Acceleration/Deceleration 0 Specified Operation Error at Start" (error code: 1A0CH) occurs, and the axis does not start.

■When selecting "1: MaximumAcceleration"

If "0.0" is specified for the acceleration or the deceleration, an error will not occur. For acceleration, even if the jerk is set, trapezoidal acceleration/deceleration is operated and the acceleration/deceleration is performed in one operation cycle. In these cases, acceleration/deceleration is performed exceeding the acceleration limit value and deceleration limit value.



Precautions

Pay full attention when setting "1: MaximumAcceleration", because operation is rapid motion.

Point

Operation when specifying acceleration/deceleration 0 is applied to acceleration/deceleration after override.

Specifying 0.0 other than at start

■At the acceleration change/At the deceleration change

When "0.0" is specified at the acceleration change or the deceleration change, the change will not be accepted, and the operation is continued at the target acceleration or the target deceleration before change.

■At multiple start

When "0.0" is specified for the acceleration or the deceleration in the multiple started FB, the target acceleration or the target deceleration of the previous FB is taken over.

Specifying 0.0 at stop processing

In MC_Stop/MC_GroupStop and the deceleration at stop setting, when "0.0" is specified for the deceleration, the axis immediately stops without an acceleration/deceleration.

Acceleration limit value/deceleration limit value

When "0: mcAccDec" is selected in the "acceleration/deceleration method setting" (Options bit0 to 2) of the FB, acceleration/deceleration is limited by the acceleration/deceleration limit value.
When "1: mcFixedTime" is selected in Options of the FB, it is not limited. When "0.0" is specified for the acceleration/deceleration limit value, it is not limited as well.
Applicable range of the acceleration/deceleration limit value for single axis control and multiple axes control is set same as the velocity limit. For details, refer to the following.
☞ Page 356 Speed Limit

Operation

This operation is for the FB which has Acceleration and Deceleration of input variables.
Acceleration/deceleration is limited at the following timing.

■At start

If the speed exceeds the acceleration limit value or the deceleration limit value, the error "Acceleration Limit Value Over" (error code: 1A38H) or the "Deceleration Limit Value Over" (error code: 1A39H) will occur, and the axis will not start.

■At acceleration change/At deceleration change

If the speed exceeds the acceleration limit value or the deceleration limit value by changing the acceleration or the deceleration, the warning "Acceleration Limit Value Over Warning" (warning code: 0D06H) or the warning "Deceleration Limit Value Over Warning" (warning code: 0D07H) will occur, and the change will not be accepted. The acceleration/deceleration operation before change will be continued.

■At multiple start

If the acceleration or the deceleration of the multiple started FB has exceeded the acceleration limit value or the deceleration limit value, the error "Acceleration Limit Value Over" (error code: 1A38H) or the error "Deceleration Limit Value Over" (error code: 1A39H) will occur, and the instruction in execution will be interrupted and stopped. For details, refer to the following.
☞ Page 159 Precautions

■At stop processing

- When decelerating stop by MC_Stop/MC_GroupStop
  When exceeding the deceleration limit value, the error "Deceleration Limit Value Over" (error code: 1A39H) occurs and the axis stops according to AxisName(AxesGroupName).Pr.StopMode_General (Stop selection at stop by factors).
- When decelerating stop by AxisName(AxesGroupName).Pr.StopMode_Deceleration (Deceleration at stop)
  When exceeding the deceleration limit value, the warning "Deceleration Limit Value Over Warning" (warning code: 0D07H) occurs and the target deceleration becomes the deceleration limit value.

Jerk limit value

When the jerk acceleration/deceleration is (J ≠ 0.0), Jerk is limited by a value specified in JerkLimit. When "0.0" is specified for the jerk limit value, the jerk is not limited.
Applicable range and operation of the jerk limit value for single axis control and multiple axes control are set same as the velocity limit. For details, refer to the following.
☞ Page 356 Speed Limit

Operation

This operation is for the FB which has Jerk of input variables.
Jerk is limited at the following timing.

■At start

If the specified jerk at start exceed the jerk limit value, the error "Jerk Limit Value Over" (error code: 1A3AH) will occur, and the axis will not start.

■At multiple start

If the specified jerk in the multiple started FB has exceeded the jerk limit value, the error "Jerk Limit Value Over" (error code: 1A3AH) will occur, and the instruction in execution will be interrupted and the axis will stop. For details, refer to the following.
☞ Page 159 Precautions

Limit of acceleration time/deceleration time

The acceleration time and the deceleration time are limited by 8400 [s].
When "0: mcAccDec" is selected in the "acceleration/deceleration method setting" (Options bit0 to 2) of the FB, the acceleration time and the deceleration time calculated from the acceleration and the deceleration are limited. For a method of calculating the acceleration time and the deceleration time, refer to the following.
☞ Page 336 Acceleration/deceleration specification method

Operation

■At start

If the acceleration time or the deceleration time exceeds 8400 [s], the error "Acceleration Time Over at Start" (error code: 1A0FH) or the error "Deceleration Time Over at Start" (error code: 1A10H) will occur, and the axis will not start.

■At control change

Limit of the acceleration time or the deceleration time may be exceeded when a velocity change, an acceleration change, or a deceleration change is carried out. In this case, the warning "Acceleration Time Limit Over Warning" (warning code: 0D04H) or the warning "Deceleration Time Limit Over Warning" (warning code: 0D05H) occurs, and the control change is not accepted. The acceleration/deceleration operation before change is continued. When a control change is carried out during acceleration or deceleration, time from the issuance of the control change instruction to the end of acceleration or deceleration is limited.

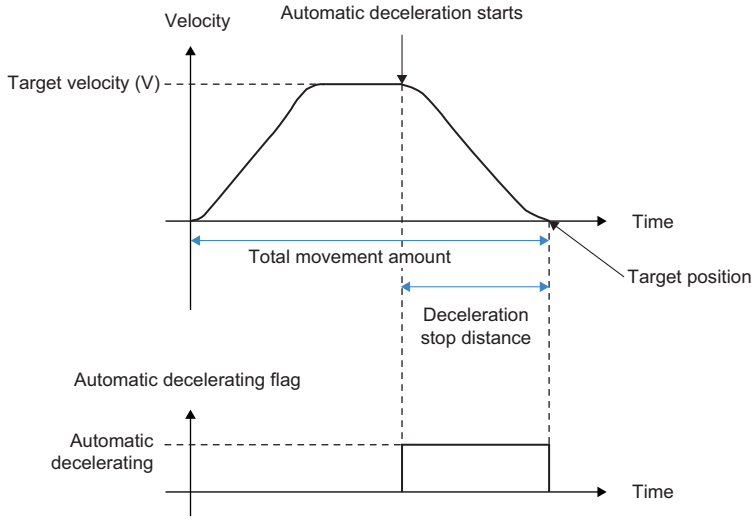
■When the acceleration time/deceleration time is within one operation cycle

When the acceleration time or the deceleration time is less than one operation cycle due to the specified velocity, acceleration, deceleration, and jerk, the speed reaches the target velocity within one operation cycle regardless of the jerk.

Automatic deceleration

Decelerating to a speed of "0.0" to stop at the specified target position is called automatic deceleration.
During automatic deceleration, AxisName.Md.AutoDeceleration is TRUE.
When automatic deceleration is performed, the deceleration may go ahead or behind the target deceleration. Therefore, AxisName.Md.SetAcceleration may exceed AxisName.Md.TargetDeceleration.
Besides, the current speed may be continued due to the compensation of the remaining distance during automatic deceleration of the trapezoidal acceleration/deceleration and at acceleration/deceleration time-fixed method. In this case, SetAcceleration will be 0.

Term	Description
Deceleration stop distance	Distance required to a deceleration stop to the target position
Total movement amount	Movement amount from start of the FB to reaching the target position
Cumulative movement amount	Movement amount from start of the FB to the current position
Next Δ movement amount	Movement amount in the next operation cycle



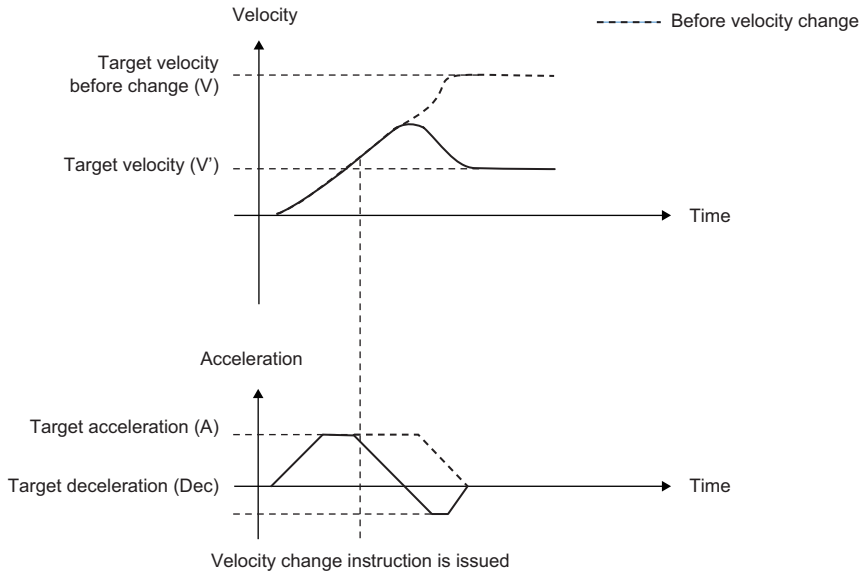
Automatic deceleration starts when the following condition is satisfied.
Deceleration stop distance > Total movement amount - Cumulative movement amount - Next Δ movement amount

- When a multiple start is in execution, automatic deceleration processing is performed when decelerating to the target position during execution of the final positioning point.
- The automatic deceleration flag turns OFF at the deceleration caused by the stop command or the stop cause occurrence.

Changing control of jerk acceleration/deceleration

Acceleration/deceleration operation at changing control immediately changes the acceleration and the deceleration for the trapezoidal acceleration/deceleration (J = 0.0).

For the jerk acceleration/deceleration (J ≠ 0.0), when a control change is carried out during acceleration or deceleration, acceleration/deceleration will be performed taking over the previous acceleration (deceleration). Therefore, if a control change that causes the deceleration operation is executed during acceleration, the operation will continue to accelerate until the acceleration becomes 0 without decelerating immediately.



Term	Description
Velocity change	Indicates the following operations • For retrigger/continuous update the velocity input valiables • For executing velocity override coefficient change
Acceleration change	Indicates the following operations • For retrigger/continuous update the acceleration input valiables • For executing acceleration override coefficient change
Deceleration change	Indicates the following operations • For retrigger/continuous update the deceleration input valiables • For executing acceleration override coefficient change

The following shows the specific operation of the jerk acceleration/deceleration during control change.

Target position change

The specific acceleration/deceleration operation of jerk acceleration/deceleration is shown below. For the basic control description, refer to the following.

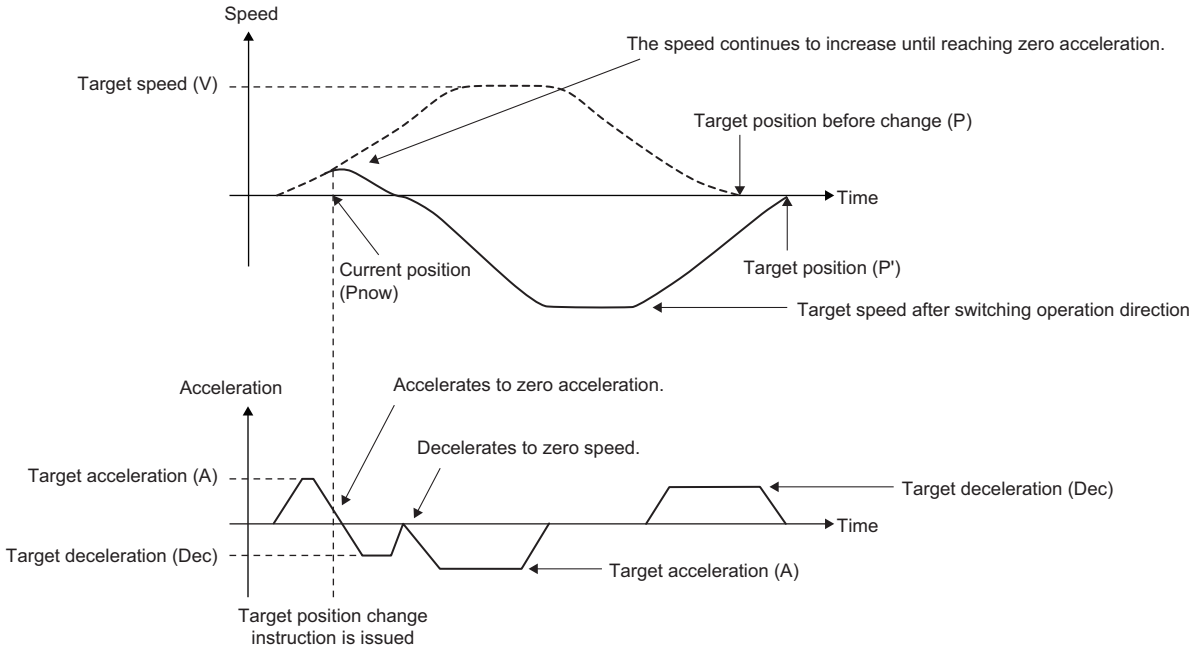
☞ Page 413 Target position/movement distance change

Term	Abbreviation	Description
Current position	Pnow	The position when a target position change instruction is issued
Target position before change	P	The target position before the target position is changed
Target position	P'	The target position after the target position was changed

■When the operation direction is reversed

When the target position after change is specified in the front of the current position, the operation direction is reversed. The axis accelerates to the acceleration of 0, then decelerates to the speed 0. When the speed reaches 0, the operation direction is reversed and acceleration/deceleration is performed to the target position.

■For the acceleration/deceleration specification method



Velocity change

The specific acceleration/deceleration operation of jerk acceleration/deceleration is shown below. For the basic control description, refer to the following.

☞ Page 416 Set velocity change

Term	Abbreviation	Description
Current velocity	Vnow	The velocity when changing the velocity
Target velocity	V	The target velocity before the velocity is changed
Target velocity after change	V'	The target velocity after the velocity was changed

In jerk acceleration/deceleration, in order to smooth acceleration at velocity change, velocity change is carried out taking over the current acceleration before change.

■When not reaching the target acceleration

If the target velocity decreases due to velocity change, the target acceleration may not be reached. Adjust the target acceleration not to exceed the target velocity. For details, refer to the following.

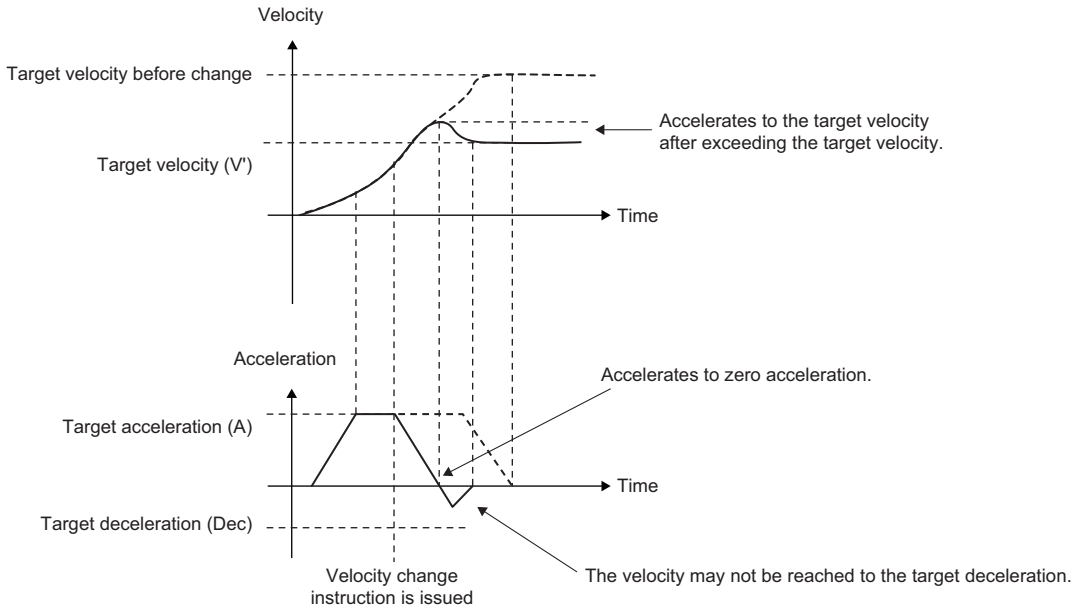
☞ Page 343 Acceleration/deceleration waveform adjustment when specifying jerk

■When exceeding the target acceleration

If speed is changed to a value smaller than the target velocity before change during acceleration, the target velocity may be exceeded. The axis accelerates to acceleration 0 after the velocity change instruction is issued but overrides the target velocity. After reaching the acceleration 0, it accelerates to the target velocity.

■Acceleration/deceleration specification method

After reaching the acceleration 0 due to velocity change during acceleration, the target deceleration may not be reached when decelerating to the target velocity.



■When a target velocity change is performed during automatic deceleration

The velocity change is accepted, however operation in execution is not changed. The target velocity after change is used when re-acceleration is performed by target position change.

Acceleration change and acceleration/deceleration time change

When "0: mcAccDec" is selected in the "acceleration/deceleration method setting" (Options bit0 to 2) of the FB, acceleration change is performed.

When selecting "1: mcFixedTime", acceleration/deceleration time change is performed. The operation when acceleration/deceleration time change is performed is the same as acceleration change to the acceleration calculated from the acceleration/deceleration time.

Term	Abbreviation	Description
Current acceleration	Anow	The acceleration when changing the acceleration
Target acceleration before change	A	Target acceleration before acceleration is changed
Target acceleration	A'	Target acceleration after acceleration was changed

■When the target velocity is not reached

When the target acceleration increases by acceleration change and the target velocity is exceeded at the specified acceleration, the target acceleration will not be reached because the target velocity is given priority. For details, refer to the following.

☞ Page 343 Acceleration/deceleration waveform adjustment when specifying jerk

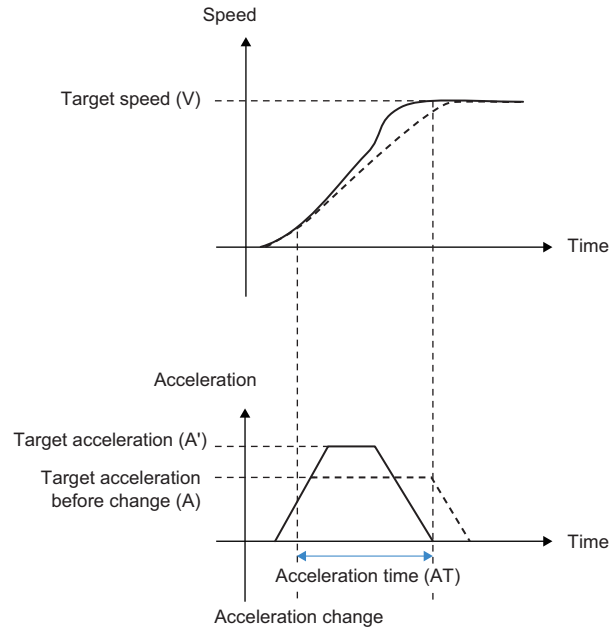
Also, when velocity change and acceleration change are simultaneously performed, the target acceleration may not be reached because the target velocity is given priority.

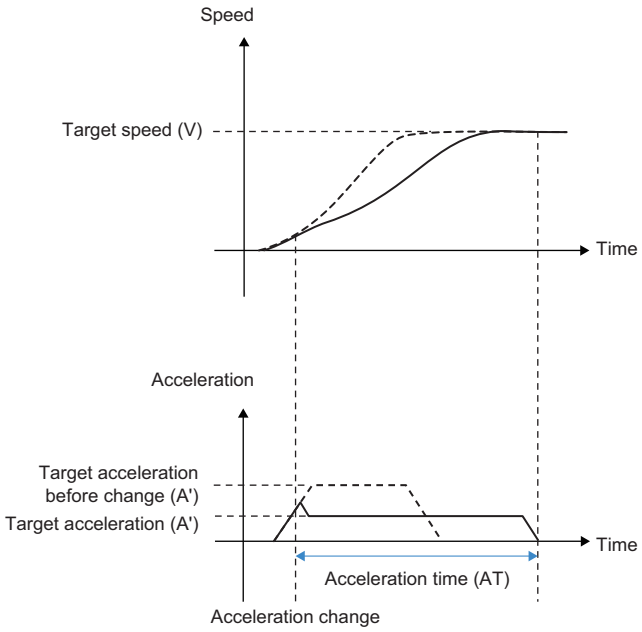
■Acceleration change

The acceleration change pattern can be classified based on the operation status when acceleration change is performed.

[When executing acceleration change in the acceleration section 1]

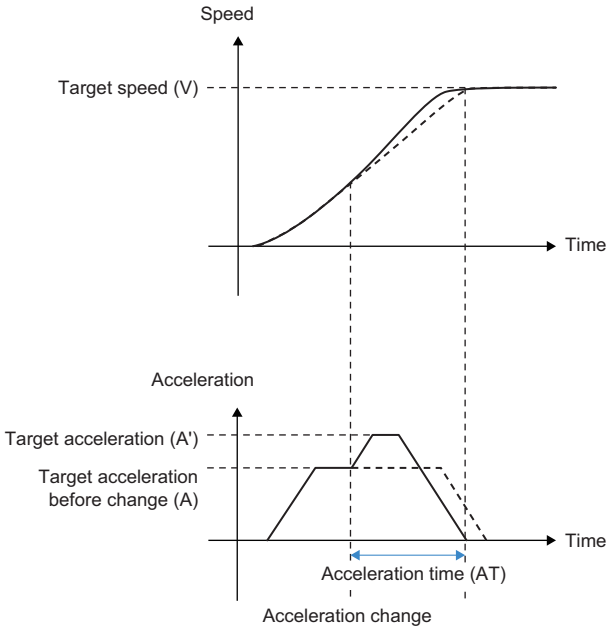
Acceleration is increased or decreased to the target acceleration after change by using jerk, and taking over the speed and the acceleration immediately before the acceleration change.

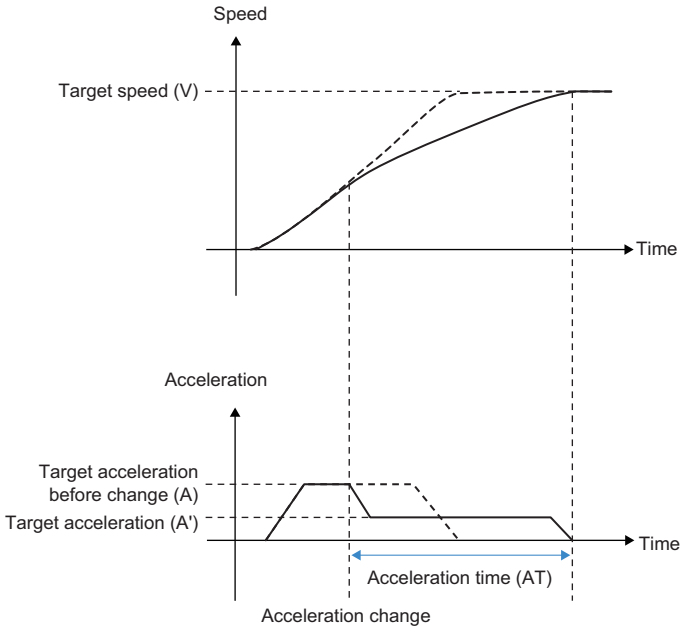




[When executing acceleration change in the maximum acceleration section]

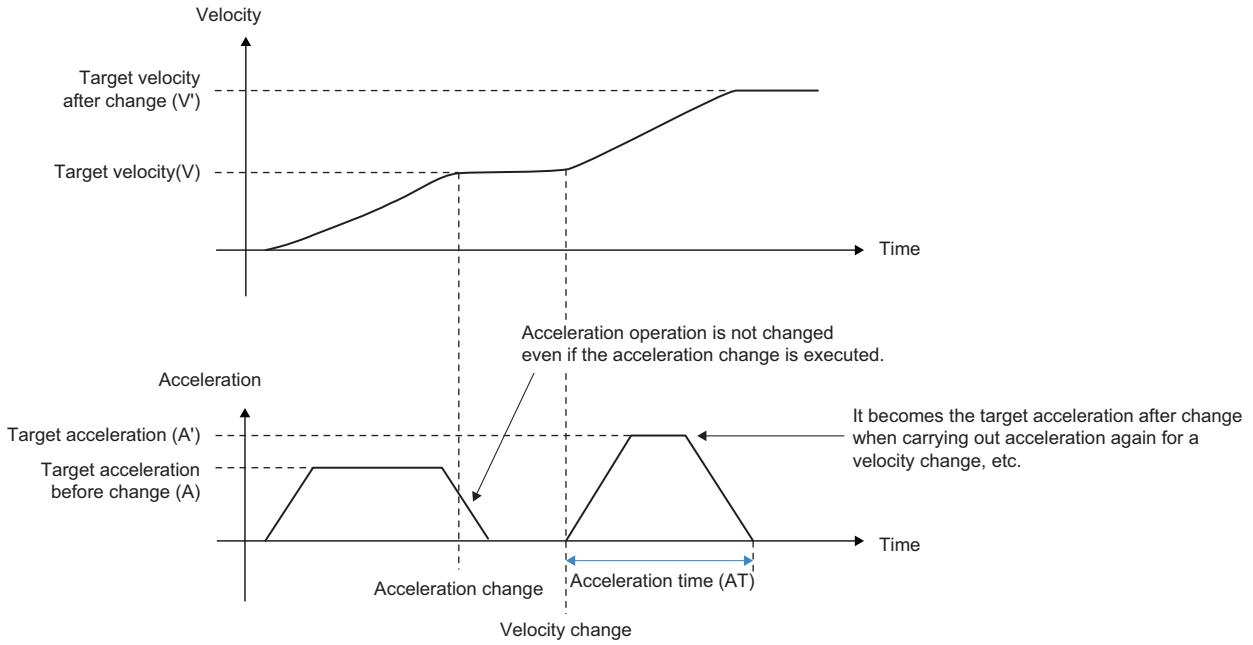
Acceleration is increased or decreased to the target acceleration after change by using jerk, and taking over the speed and the acceleration immediately before the acceleration change.





[When executing acceleration change in the acceleration section 2 or during fixed-speed, deceleration, and automatic deceleration]

When acceleration change is executed in the acceleration section 2, the acceleration change is accepted, however acceleration operation during execution does not change. When re-acceleration is performed by velocity change etc., the axis accelerates with the acceleration after change.



Deceleration change

When "0: mcAccDec" is selected in the "acceleration/deceleration method setting" (Options bit0 to 2) of the FB, deceleration change can be performed.

The following terms are used in deceleration change.

Term	Abbreviation	Description
Current deceleration	DecNow	The deceleration when changing the deceleration
Target deceleration before change	Dec	Target deceleration before deceleration is changed
Target deceleration after change	Dec'	Target deceleration after deceleration was changed

When deceleration change is performed during operation, the target position may be overrun by decreasing the deceleration. In this case, the axis is controlled based on AxisName(AxesGroupName).Pr.OverrunOperation.

[When executing deceleration change in the acceleration section or fixed-speed section]

The deceleration change is accepted, however the operation during execution does not change. When deceleration is performed, the axis decelerates with the deceleration after change.

[When executing deceleration change in deceleration other than the automatic deceleration]

The basic operation is the same as the acceleration change and the acceleration/deceleration time change. Refer to the following.

☞ Page 351 Acceleration change and acceleration/deceleration time change

[When executing deceleration change in the automatic deceleration]

The deceleration change is accepted, however the operation during execution does not change. When re-deceleration is performed after the axis re-accelerates by target velocity change, the axis decelerates with the deceleration after change.

Acceleration/deceleration method and multiple start

This section describes the acceleration/deceleration operation specific to the acceleration/deceleration method. For the basic operation of the buffer mode, refer to the following.

☞ Page 132 Multiple Start (Buffer Mode)

Term	Description
Switching point	The point where FBs are switched
On-going FB	FB that is in execution on the axis and the axes group (FB whose Active is TRUE)
Next FB	FB that multiple starts during operation of the axis and the axes group

Acceleration/deceleration operation

■When the acceleration/deceleration methods are different between the on-going FB and the next FB

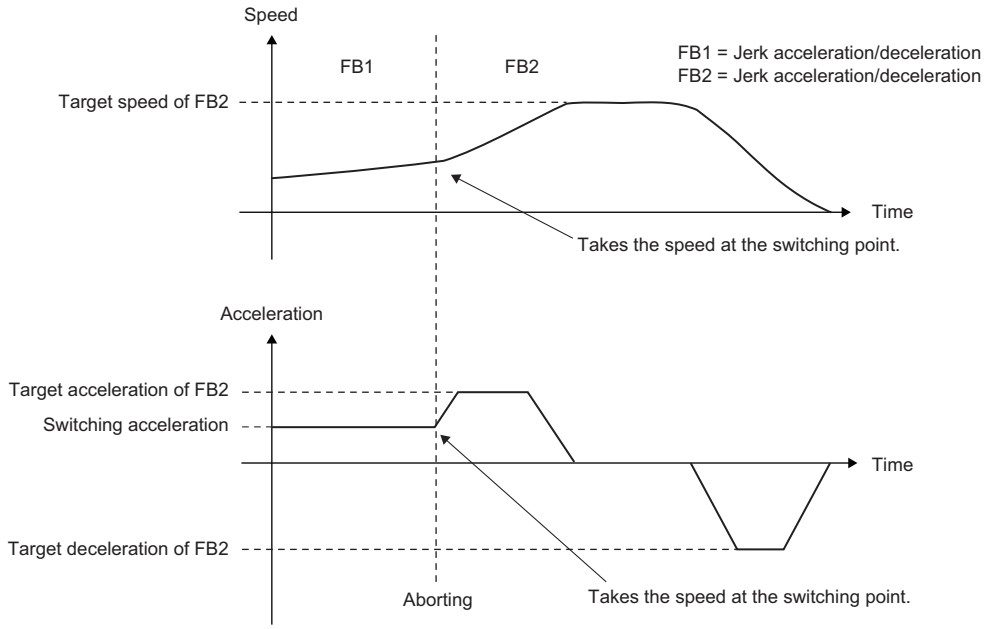
The error "Acceleration/Deceleration Method Mismatched" (error code: 1A0EH) occurs and the axis stops. Specify the same acceleration/deceleration method in the FBs.

■When the acceleration/deceleration methods are the same between the on-going FB and the next FB

- When switching "from the jerk acceleration/deceleration to the another jerk acceleration/deceleration"

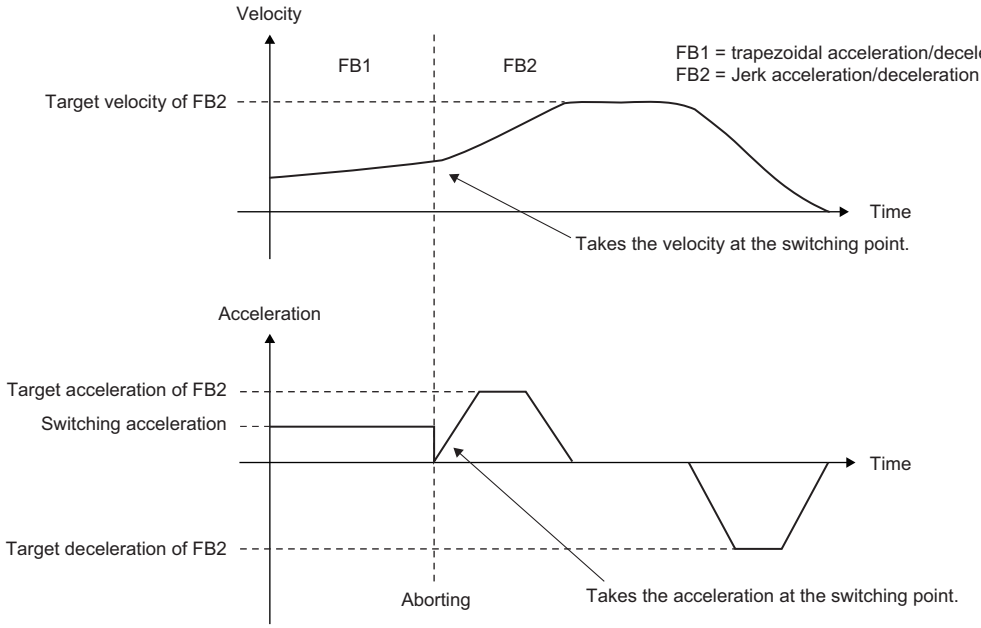
The axis accelerates/decelerates taking over the speed and the acceleration (deceleration) at the switching point.

Also, after reaching the switching point during acceleration (deceleration), the on-going velocity waveform continues in case all settings of the target velocity, the acceleration, the deceleration, and the jerk are the same between the on-going FB and the next FB. If there is a different setting in any of those, the velocity change is performed to the target velocity of the next FB.



- Switching from other than "the jerk acceleration/deceleration to the another jerk acceleration/deceleration"
  In the following case, the acceleration (deceleration) is not taken over at the switching point.

  - Switching from the jerk acceleration/deceleration to the trapezoidal acceleration/deceleration
  - Switching from the trapezoidal acceleration/deceleration to the another trapezoidal acceleration/deceleration
  - Switching from the trapezoidal acceleration/deceleration to the jerk acceleration/deceleration



9.2 Speed Limit

The "speed limit function" limits the set velocity to a value within the speed limit value setting range when the set velocity during control exceeds the speed limit value. By limiting the set velocity within the preset speed limit value setting range, even if an incorrect command or unexpected operation occurs, damage to the machine can be prevented.

Relevant variables

Variable/Structure name		Name	Details
AxisName.Pr.			
	VelocityLimit_Positive	Positive direction speed limit value	Sets the address increasing direction speed limit value for each axis.
	VelocityLimit_Negative	Negative direction speed limit value	Sets the address decreasing direction speed limit value for each axis.
	VelocityLimit_OverOperation	Speed limit value over operation setting	Sets the operation when the speed limit value during control is exceeded. 0: Ignore (Ignore) 3: ImmediateStop (Immediate stop)
AxesGroupName.Pr.			
	VelocityLimit	Speed limit value	Sets the speed limit value of the axes group.

How to set the speed limit function

Speed limit value for each axis

Set the parameters, AxisName.Pr.VelocityLimit_Positive and AxisName.Pr.VelocityLimit_Negative, for each axis.

Setting item	Setting range	Setting details
VelocityLimit_Positive	0.0001 ≤ Setting value ≤ 2500000000.0	Sets the speed limit value (the maximum speed during control) for the positive direction operation.
VelocityLimit_Negative		Sets the speed limit value (the maximum speed during control) for the negative direction operation.

Speed limit value for each axes group

Set the parameter, AxesGroupName.Pr.VelocityLimit, for each axes group.

Setting item	Setting range	Setting details
VelocityLimit	0.0001 ≤ Setting value ≤ 2500000000.0	Sets the speed limit value (the maximum speed during control) of the axes group.

Operation when the speed limit value is exceeded

The operation when specifying the speed exceeding the speed limit value differs depending on single axis control, single axis synchronous control and interpolation control.

For single axis operation

Executes the speed limit to the actual target velocity which the override, etc. are considered for the commanded velocity.

• At start

If the target velocity at start exceeds the speed limit value (AxisName.Pr.VelocityLimit_Positive for the positive direction and AxisName.Pr.VelocityLimit_Negative for the negative direction) of each axis, the error "Speed Limit Value Over" (error code: 1A07H) will occur, and the axis will not start.

• During control

The speed limit is not executed.

• During control (At speed change)

If the speed exceeding the speed limit value of each axis is specified by the override function and input label change in execution, the warning "Speed Limit Value Over Warning" (warning code: 0D03H) will occur, and the operation continues at the speed before change.

For details on the override function and input label change in execution, refer to the following.

☞ Page 359 Override Function, ☞ Page 412 Input Variable Change in Execution

For single axis synchronous control

Executes the speed limit to the set velocity of the slave axis.

• At start

The speed limit is not executed.

• During control

If the set velocity exceeds the speed limit value (AxisName.Pr.VelocityLimit_Positive for the positive direction and AxisName.Pr.VelocityLimit_Negative for the negative direction) of each axis, follow AxisName.Pr.VelocityLimit_OverOperation.

If the absolute value of the set velocity exceeds the maximum speed, the error "Speed Range Over during Controlling" (error code: 1AE8H) will occur and stop the operation.

For interpolation control

For the speed limit at the interpolation control, AxisName.Pr.VelocityLimit_Positive, AxisName.Pr.VelocityLimit_Negative, AxesGroupName.Pr.VelocityLimit are used.

The operation when the target velocity exceeds the speed limit value is as follows.

• At start

Variable to be used	Linear interpolation control			Circular interpolation control
	Vector speed specification	Long axis speed specification	Reference axis speed specification	
AxisName.Pr.VelocityLimit_Positive or AxisName.Pr.VelocityLimit_Negative[*1]	Speed is not restricted.	Long axis speed is restricted by speed limit value of long axis.[*2]	Reference axis speed is restricted by speed limit value of reference axis.[*2]	Speed is not restricted.
AxesGroupName.Pr.VelocityLimit	Speed is restricted for vector speed.[*2]	Speed is not restricted.	Speed is not restricted.	Speed is restricted for vector speed.[*2]

• During control

Variable to be used	Linear interpolation control			Circular interpolation control
	Vector speed specification	Long axis speed specification	Reference axis speed specification	
AxisName.Pr.VelocityLimit_Positive or AxisName.Pr.VelocityLimit_Negative*1	Speed is restricted for set velocity of each configuration axis. When the speed limit value is exceeded, it follows AxisName.Pr.VelocityLimit_OverOperation.			
AxesGroupName.Pr.VelocityLimit	Speed is not restricted.			

• During control (At speed change)

Variable to be used	Linear interpolation control			Circular interpolation control
	Vector speed specification	Long axis speed specification	Reference axis speed specification	
AxisName.Pr.VelocityLimit_Positive or AxisName.Pr.VelocityLimit_Negative*1	Speed is not restricted.	Long axis speed is restricted by speed limit value of long axis.*3	Reference axis speed is restricted by speed limit value of reference axis. *3	Speed is not restricted.
AxesGroupName.Pr.VelocityLimit	*3	Speed is not restricted.	Speed is not restricted.	*3

*1 About either AxisName.Pr.VelocityLimit_Positive or AxisName.Pr.VelocityLimit_Negative is restricted according to the movement direction at the limit value check.
*2 When the speed limit value is exceeded, it will not start. The error "Speed Limit Value Over" (error code: 1A07H) is output.
*3 When the speed limit value is exceeded, the speed change will not accepted. The warning "Speed Limit Value Over Warning" (warning code: 0D03H) warning is output.

Operation setting when the speed limit value is exceeded

The operation when the speed of any axis exceeds the speed limit value during control can be specified in AxisName.Pr.VelocityLimit_OverOperation.

The following two types can be selected for the operation when the speed exceeds the speed limit value.

• 0: Ignore (Ignore)
• 3: ImmediateStop (Immediate stop)

When the multiple axes exceed the limit value, the priority is "3: ImmediateStop" > "0: Ignore".

Control continuation

When "0: Ignore" is specified in AxisName.Pr.VelocityLimit_OverOperation, the warning "Speed Limit Value Over Warning" (warning code: 0D03) 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.

Control stop

When "3: ImmediateStop" is specified in AxisName.Pr.VelocityLimit_OverOperation, the error "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.

9.3 Override Function

- This function sets the factor for the velocity and performs the control to change the target velocity.
- This function sets the factor for the acceleration/deceleration and performs the control to change the target acceleration/target deceleration. (The ratio to the acceleration/deceleration time cannot be set.)
- This function is invalid when using the control without the target velocity and the target acceleration. (Ex. Cam operation of the positioning control)
- When the velocity and the acceleration/deceleration are changed by retrigger or continuous update, the override factor will be added to changed velocity and the acceleration/deceleration.
- To change the override factor, there are a method of using the dedicated FB and a method of using control data.

Change method	Description
Change using a dedicated FB	MC_SetOverride, MC_GroupSetOverride
Change using control data	AxisName.Cd.VelocityOverride, AxesGroupName.Cd.VelocityOverride, etc.

- The override factor can be set in the axis control data and the axes group control data respectively. Only the override factor set in the axis control data is valid for single axis control, and only the override factor set in the axes group control data is valid for axes group control
- The override coefficient does not reflect immediately during deceleration stop after stop causes occur. It reflects at the timing which causes other stop causes.

Point

It is as follows when the value after the velocity override is lower than the lower limit value.

- The velocity is 0 when it is 0 < (The velocity after override) < The lower limit value of the velocity (0.0001).

It is as follows when the value after the accelaration override is higher than the upper limit value.

- The accelaration is the accelaration upper limit value (2147483647.0) when AxisName(AxesGroupName).Pr.AccelerationLImit is 0.
- The decelaration is the decelaration upper limit value (2147483647.0) when AxisName(AxesGroupName).Pr.DecelerationLImit is 0.

It is as follows when the value after the accelaration override is lower than the lower limit value.

[For the acceleration/deceleration method in which acceleration/deceleration is specified]

- The accelaration is 0 when it is 0 < (The deceleration after override) < the deceleration lower limit value (0.0001).
- The decelaration is 0 when it is 0 < (The accelaration after override) < the accelaration lower limit value (0.0001).

[For the acceleration/deceleration method in which acceleration/deceleration time is specified]

- The acceleration/deceleration time is 0 when it is 0 < (The acceleration/deceleration time after override) < the acceleration/deceleration time lower limit value (0.000001).

Targets to be changed and corresponding FBs, control data

Targets to be changed			Corresponding FB	Corresponding control data
Velocity	Current value		MC_MoveVelocity MC_SetOverride MC_GroupSetOverride	AxisName.Cd.VelocityOverride, AxesGroupName.Cd.VelocityOverride
	Limit value		—*1	—*1
	Limit value	Maximum value of the limit value	—*2	—*2

*1 Axis parameters can be changed by MC_WriteParameter. However, it cannot be changed during control.

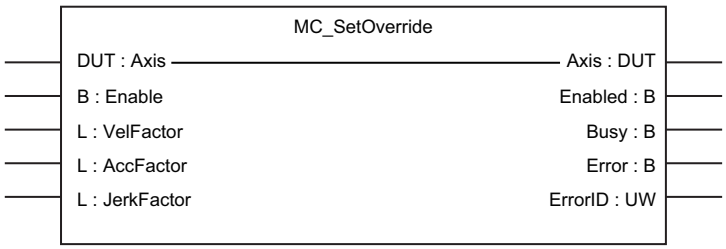
*2 The maximum value of the limit value cannot be changed.

Relevant variables/FBs

Variable/Structure name	Name	Details
AxisName.Cd. AxesGroupName.Cd.		
VelocityOverride	Velocity override factor	Sets the velocity override factor.
AccelerationOverride	Acceleration override factor	Sets the acceleration override factor.
AxisName.Md. AxesGroupName.Md.		
VelocityOverride	Velocity override factor	Sets the velocity override factor.
AccelerationOverride	Acceleration override factor	Sets the acceleration override factor.

The override factor can be changed by using MC_SetOverride and MC_GroupSetOverride.

MC_SetOverride

Item	Description
Function overview	Changes the target velocity, the target acceleration and the target deceleration of the specified axis.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

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

■Function description

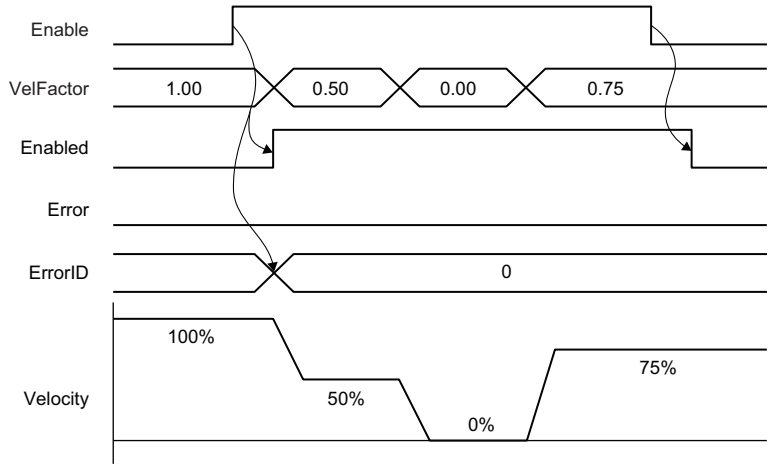
- This FB changes the target velocity, the target acceleration and the target deceleration of the specified axis.
- The values are changed to each value obtained by multiplying the override factor to the target velocity, the target acceleration and the target deceleration respectively.
- The FB is executed when Enable is TRUE. Enabled turns TRUE while the override factor is valid.
- If the override factor value is changed while Enable is TRUE, the new override factor is reflected.
- When an error has occurred in the FB, this FB turns Error to TRUE and stores an error code in ErrorID. For details on error codes, refer to the following.
  ☞ Page 662 List of Error Codes
- When 0.0 is set to the velocity override factor, the axis stops without changing the axis status to "4: Standstill".
- When 0.0 is set to the acceleration override factor, the acceleration override factor is not changed, and the previous acceleration override factor is maintained.

Precautions

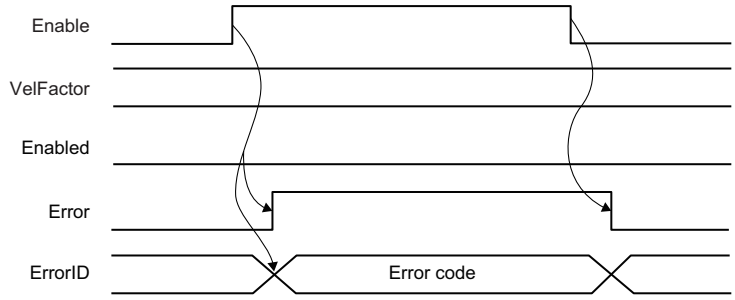
This FB changes AxisName.Cd.VelocityOverride and AxisName.Cd.AccelerationOverride.

■Timing chart

[When the FB is normally completed]



[When an error occurs]



■I/O variable

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Axis information	Axis	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables

■Input variables

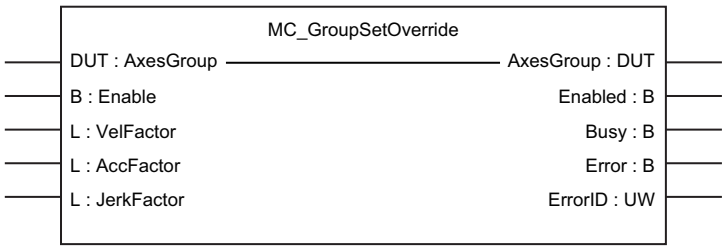
Import ⎍: Always

Name	Variable name	Data type	Import	Setting range	Default value	Description
Enable	Enable	BOOL	⎍	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Velocity override factor	VelFactor	LREAL	⎍	0.0 to 10.0	0.0	While Enable is TRUE, values are always imported. Sets the velocity override factor.
Acceleration override factor	AccFactor	LREAL	⎍	0, 0.01 to 10.0	0.0	While Enable is TRUE, values are always imported. Sets override factor for the acceleration and the deceleration. When "0.0" is set, the acceleration override factor is not changed, and the control is performed with the previous value.
Jerk override factor (Reserved)	JerkFactor	LREAL	⎍	0.0	0.0	Set "0.0". (A value other than "0.0" causes the error "Out of Jerk Override Coefficient (JerkFactor) Range" (error code: 349EH).)

■Output variables

Name	Variable name	Data type	Default value	Description
Enabled	Enabled	BOOL	FALSE	Turns TRUE when the set override value is correct.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Error	Error	BOOL	FALSE	When TRUE, it indicates that an error has occurred
Error code	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned. For details, refer to the following. ☞ Page 662 List of Error Codes

MC_GroupSetOverride

Item	Description
Function overview	Changes the target velocity, the target acceleration and the target deceleration of the specified axes group.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

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

■Function description

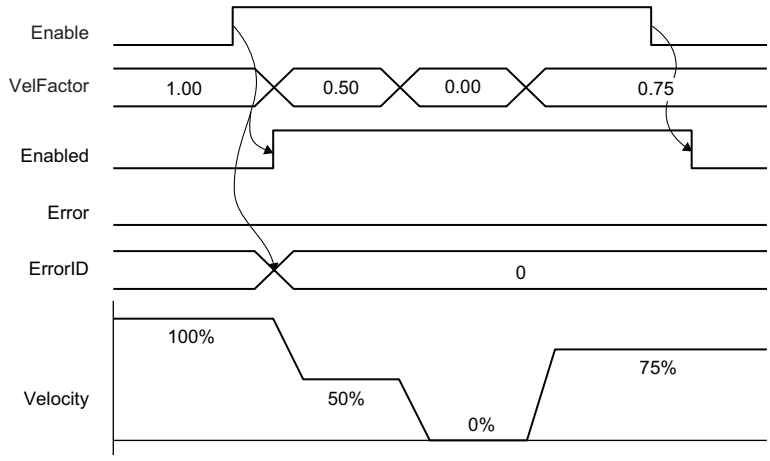
- This FB changes the target velocity, the target acceleration and the target deceleration of the specified axes group.
- The values are changed to each value obtained by multiplying override factor to the velocity, the acceleration and the deceleration respectively.
- The FB is executed when Enable is TRUE. Enabled turns TRUE while the override factor is valid.
- If the override factor value is changed while Enable is TRUE, the new override factor is reflected.
- When an error has occurred in the FB, this FB turns Error to TRUE and stores an error code in ErrorID. For details on error codes, refer to the following.
  ☞ Page 662 List of Error Codes
- When 0.0 is set to the velocity override factor, the axis stops without changing the axis status to "4: Standstill".
- When 0.0 is set to the acceleration override factor, the acceleration override factor is not changed, and the previous acceleration override factor is maintained.

Precautions

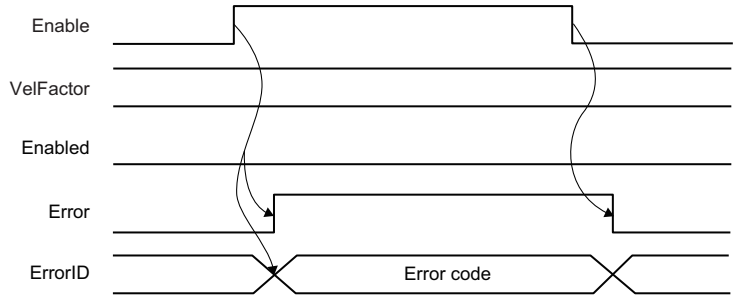
This FB changes AxesGroupName.Cd.VelocityOverride and AxesGroupName.Cd.AccelerationOverride.

■Timing chart

[When the FB is normally completed]



[When an error occurs]



■I/O variable

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Axes group information	AxesGroup	AXES_GROUP_REF	↑	—	Omission not possible	☞ Page 706 Axes group variables

■Input variables

Import ⎍: Always

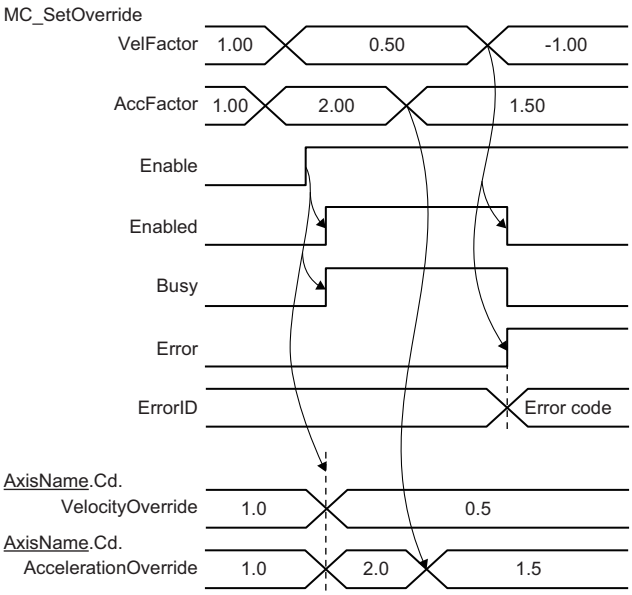
Name	Variable name	Data type	Import	Setting range	Default value	Description
Enable	Enable	BOOL	⎍	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Velocity override factor	VelFactor	LREAL	⎍	0.0 to 10.0	0.0	While Enable is TRUE, values are always imported. Sets the velocity override factor.
Acceleration override factor	AccFactor	LREAL	⎍	0, 0.01 to 10.0	0.0	While Enable is TRUE, values are always imported. Sets override factor for the acceleration and the deceleration. When "0.0" is set, the acceleration override factor is not changed, and the control is performed with the previous value.
Jerk override factor (Reserved)	JerkFactor	LREAL	⎍	0.0	0.0	Set "0.0". (A value other than "0.0" causes the error "Out of Jerk Override Coefficient (JerkFactor) Range" (error code: 349EH).)

■Output variables

Name	Variable name	Data type	Default value	Description
Enabled	Enabled	BOOL	FALSE	Turns TRUE when the set override value is correct.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Error	Error	BOOL	FALSE	When TRUE, it indicates that an error has occurred.
Error code	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned. For details, refer to the following. ☞ Page 662 List of Error Codes

Operation overview

In MC_SetOverride and MC_GroupSetOverride, AxisName(AxesGroupName).Cd.VelocityOverride and AxisName(AxesGroupName).Cd.AccelerationOverride of the axis or the axes group can be changed.



If a value outside the range is set as the override factor, an error will occur in the FB, and subsequent values will not be imported. To import values again, start up Enable again.

Precautions

- Do not arrange two or more MC_SetOverride and MC_GroupSetOverride in the same axis or the same axes group. Operation is not guaranteed when two or more of the FBs are arranged.
- In the single axis control, only the override factor set with the axis control data affects. In the axes group control, only the override factor set with the axes group control data affects.
- Do not operate AxisName.Cd.VelocityOverrid or AxisName.Cd.AccelerationOverride directly during performing MC_SetOverride
- Do not operate AxesGroupName.Cd.VelocityOverride or AxesGroupName.Cd.AccelerationOverride directly during performing MC_GroupSetOverride

How to change control data

Override factor setting range

The following shows each override factor setting range.

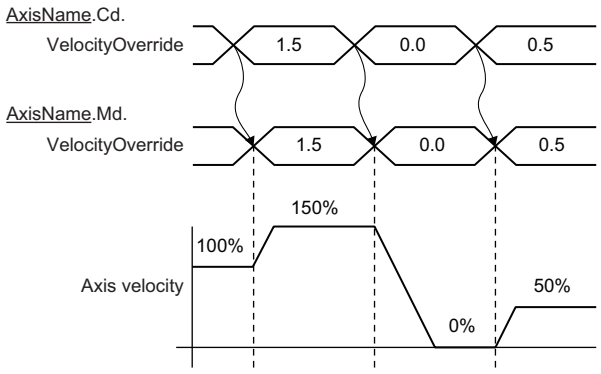
Variable/Structure name		Name	Setting range
AxisName.Cd. AxesGroupName.Cd.			
	VelocityOverride	Velocity override factor	0.0 (0%) ≤ Setting value ≤ 10.0 (1000%)
	AccelerationOverride*1	Acceleration override factor	0.01 (1%) ≤ Setting value ≤ 10.0 (1000%)
	JerkOverride (Reserved)	Jerk override factor	1

*1 Even if the acceleration/deceleration method specifying the acceleration/deceleration time, the override factor will be used to the acceleration/deceleration.

Operation when the velocity override factor is changed

Change the velocity by changing AxisName(AxesGroupName).Cd.VelocityOverride.

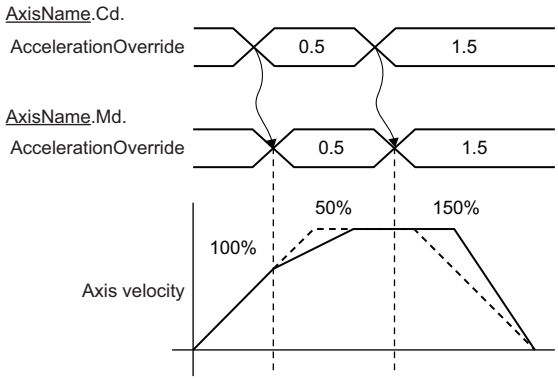
When "0.0" is set in AxisName(AxesGroupName).Cd.VelocityOverride, the axis is stopped without moving to the "4: Standstill" status.



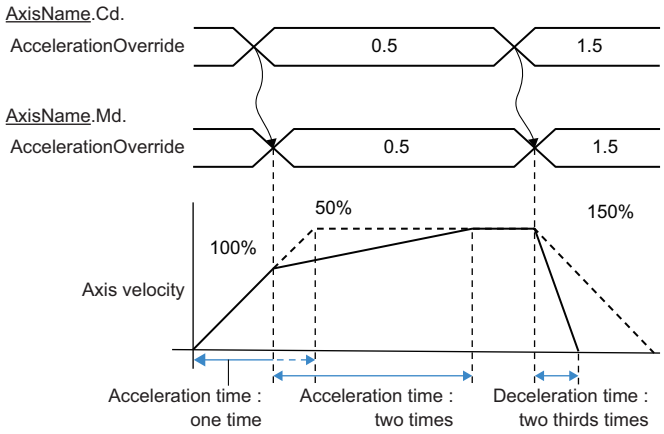
Operation when the acceleration override factor is changed

Change the acceleration/deceleration by changing AxisName(AxesGroupName).Cd.AccelerationOverride.

■Acceleration/deceleration method specifying acceleration/deceleration



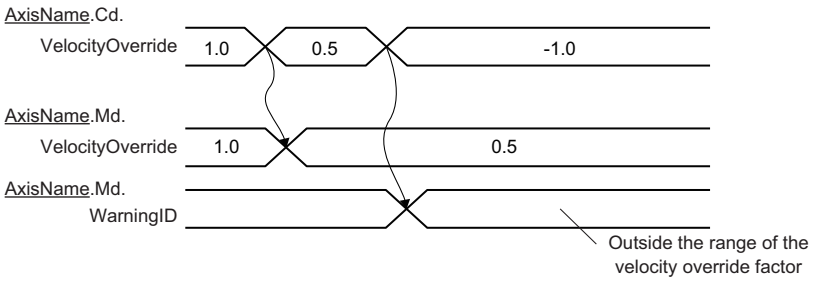
■Acceleration/deceleration method specifying acceleration/deceleration time



Operation when a value outside the override factor setting range is set

If a value outside the setting range is set in each override factor, the following warnings occur, and the override factor will remain unchanged from the previous value.

- The warning "Out of Speed Override Coefficient Range Warning" (warning code: 0D0BH)
- The warning "Out of Acceleration Override Coefficient Range Warning" (warning code: 0D0CH)

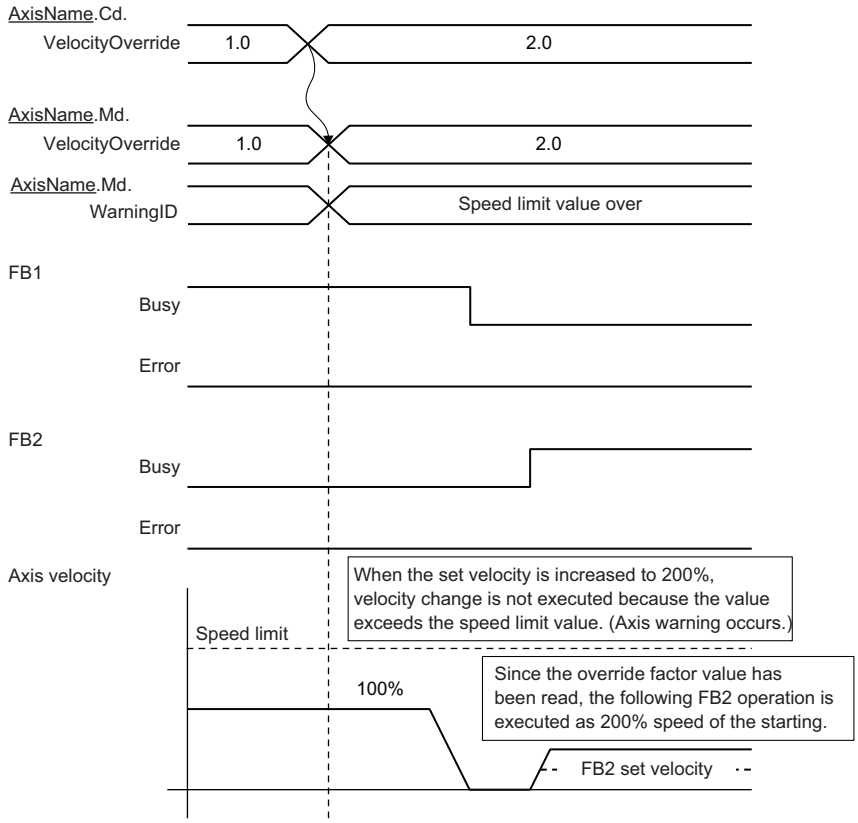


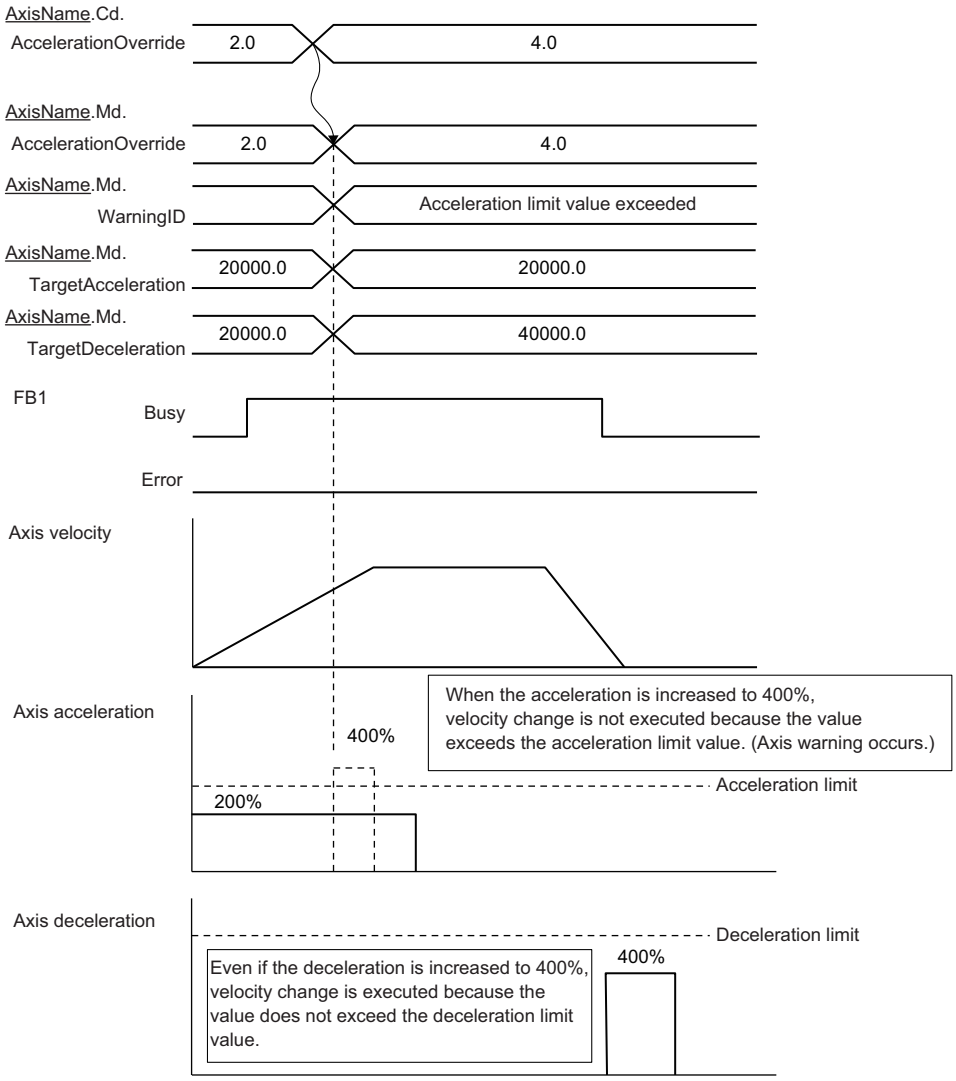
Operation at velocity/acceleration/deceleration limit value over by override

■When velocity limit value over occurs by velocity override.

Velocity over will occur during control execution and the velocity will not change.

When acceleration override factor is changed, the acceleration and deceleration will change. In this case, when acceleration limit value over occurs, the acceleration over will occur and the acceleration will change. When deceleration limit value over occurs, the deceleration over will occur and the deceleration will not change as well. (An error does not occur by a control data change, and the override factor is changed.)



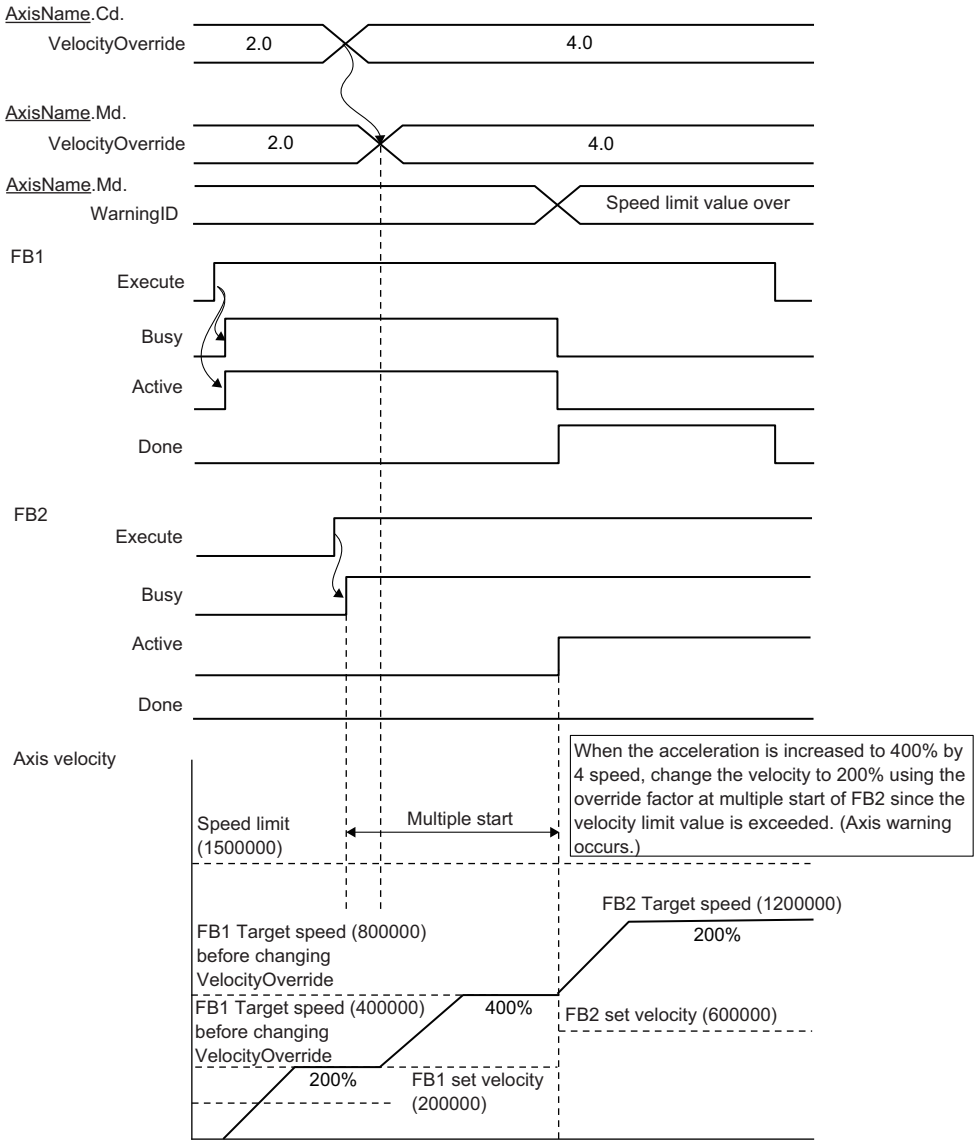

AxisName.Cd.
AccelerationOverride
2.0
4.0
AxisName.Md.
AccelerationOverride
2.0
4.0
AxisName.Md.
WarningID
Acceleration limit value exceeded
AxisName.Md.
TargetAcceleration
20000.0
20000.0
AxisName.Md.
TargetDeceleration
20000.0
40000.0
FB1
Busy
Error
Axis velocity
Axis acceleration
When the acceleration is increased to 400%, velocity change is not executed because the value exceeds the acceleration limit value. (Axis warning occurs.)
400%
Acceleration limit
200%
Axis deceleration
Deceleration limit
400%
Even if the deceleration is increased to 400%, velocity change is executed because the value does not exceed the deceleration limit value.

■When velocity/acceleration/deceleration limit value over occurs at switching of FB by buffer mode

The control change of buffering FB is executed at swiching of FB.

When the control is changed at switching of FB, velocity/acceleration/deceleration limit value over may occur. When velocity/acceleration/deceleration limit value over occurs, velocity/acceleration/deceleration and override factor at multiple start are used to control.

Also when velocity/acceleration/deceleration limit value over occurs at swiching of FB, the import of override factor is executed. If the limit value is not exceeded at the next control change, the override factor after change will be operated.



10 RELEVANT FUNCTIONS TO TORQUE

10.1 Torque limit

The "torque limit function" limits the generated torque to a value within the torque limit value range when the torque generated in the servomotor exceeds the torque limit value.

It controls the operation so that unnecessary force is not applied to the load and machine by limiting the generated torque.

- Set the positive direction (address increment direction) torque limit value and the negative direction (address decrement direction) torque limit value for each axis.
- The torque limit value is used in the "setting of the torque limit value used in the driver" and the "control to suppress the command torque in MC_TorqueControl, etc.".
- The torque limit value can be changed by changing the values of AxisName.Cd.TorqueLimit_Positive and AxisName.Cd.TorqueLimit_Negative. When changing the torque limit value, specifies a value less than or equal to the AxisName.PrConst.TorqueLimit_Max setting value. For details, refer to the following.
  ☞ Page 373 Torque Limit Value Change Function
- The torque limit value that is currently valid can be checked with AxisName.Md.TorqueLimit_Positive and AxisName.Md.TorqueLimit_Negative.
- The torque limit value when connecting a driver, AxisName.Cd.TorqueLimit_Positive and AxisName.Cd.TorqueLimit_Negative are set.

Relevant variables

Variable/Structure name	Name	Details
AxisName.PrConst.		
TorqueLimit_Max	Torque limit maximum value	Sets the maximum value that can be specified as the torque limit value. (Common in the positive direction and the negative direction)
TorqueLimit_PositiveInitial	Positive direction torque limit initial value	Sets the initial value of AxisName.Cd.TorqueLimit_Positive.
TorqueLimit_NegativeInitial	Negative direction torque limit initial value	Sets the initial value of AxisName.Cd.TorqueLimit_Negative.
AxisName.Cd.		
TorqueLimit_Positive	Positive direction torque limit value	Sets the positive direction torque limit value.
TorqueLimit_Negative	Negative direction torque limit value	Sets the negative direction torque limit value.
AxisName.Md.		
TorqueLimit_Positive	Positive direction torque limit value	Displays the positive direction torque limit value that is currently valid.
TorqueLimit_Negative	Negative direction torque limit value	Displays the negative direction torque limit value that is currently valid.

Slave object map

When using the torque limit function in the "setting of the torque limit value used in the driver", assign the torque limit object to the slave object setting. For details, refer to the following.

☞ Page 56 Axis Assignment

Point

- 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.
- If the torque limit object has not been assigned to the slave object map, the torque limit value is imported by the same unit as the command torque, because it is used only in the "control to suppress the command torque in MC_TorqueControl, etc.". Since the torque limit value will not be sent to the driver, refer to each drive unit manual for the torque limit value used by the driver.

Torque limit value setting range

Set the settings of AxisName.Cd.TorqueLimit_Positive and AxisName.Cd.TorqueLimit_Negative for each axis.
The setting values are used in the "setting of the torque limit value used in the driver" and the "control to suppress the command torque in MC_TorqueControl, etc.".
The torque limit value can be changed by changing the setting values. For details, refer to the following.
☞ Page 373 Torque Limit Value Change Function
When the power supply turns ON, the setting values of AxisName.PrConst.TorqueLimit_PositiveInitial and AxisName.PrConst.TorqueLimit_NegativeInitial are stored. In this time, if AxisName.PrConst.TorqueLimit_PositiveInitial and AxisName.PrConst.TorqueLimit_NegativeInitial are set with the larger value than AxisName.PrConst.TorqueLimit_Max, the error "Out of Forward Direction Torque Limit Value Specification Range" (error code: 1A79H) or the error "Out of Reverse Direction Torque Limit Value Specification Range" (error code: 1A7AH) will occur.
If a value outside the range is set during control execution, the warning "Out of Forward Direction Torque Limit Value Specification Range Warning" (warning code: 0D09H) or the warning "Out of Reverse Direction Torque Limit Value Specification Range Warning" (warning code: 0D0AH) will occur.

Control data	Setting range
AxisName.Cd.TorqueLimit_Positive	0.0 (0%) ≤ Setting value ≤ AxisName.PrConst.TorqueLimit_Max
AxisName.Cd.TorqueLimit_Negative	

Torque limit maximum value setting range

Set the settings of AxisName.PrConst.TorqueLimit_Max for each axis.
The torque limit maximum value is used as the upper limit value of the torque limit.
Even if a big value is input erroneously when changing the torque limit value, the change will not be accepted, and erroneous input is prevented. If a value outside the range is set, the error "Out of Parameter Range (Axis)" (error code: 1D80H) will occur.

Parameter	Setting range
AxisName.PrConst.TorqueLimit_Max	0.0 (0%) ≤ Setting value ≤ 1000.0 (1000%)

Point

Set the upper limit value of the torque limit specified in the driver as the torque limit maximum value.
Both the positive direction torque limit value and the negative direction torque limit value are limited by AxisName.PrConst.TorqueLimit_Max.

Precautions

When the axis control is stopped by torque limiting, the droop pulse will remain in the deviation counter. If the load torque is eliminated, operation for the amount of droop pulses will be carried out. Note that the movement might start rapidly as soon as the load torque is eliminated.

10.2 Torque Limit Value Change Function

- To change a torque limit value, there are a method of using the dedicated FB and a method of using control data.

Change method	Description
Change using a dedicated FB	MCv_SetTorqueLimit
Change using control data	AxisName.Cd.TorqueLimit_Positive, AxisName.Cd.TorqueLimit_Negative

- The number of significant digits after the decimal point differs depending on the ID assigned to the slave object map.
- In a torque limit value change, a value cannot be changed to a torque limit value that exceeds AxisName.PrConst.TorqueLimit_Max.

Targets to be changed and corresponding FBs, control data

Target to be changed			Corresponding FB	Corresponding control data
Torque	Current value		MC_TorqueControl	—
	Limit value		MCv_SetTorqueLimit	AxisName.Cd.TorqueLimit_Positive AxisName.Cd.TorqueLimit_Negative
	Limit value	Maximum value of the limit value	—*1	—

*1 Axis parameters can be changed by MC_WriteParameter. However, it cannot be changed during control.

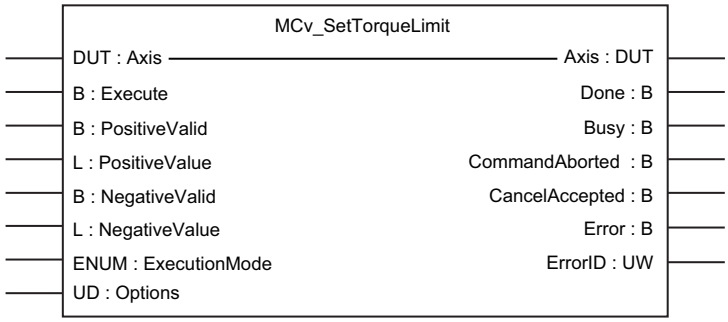
Relevant variables/FB

For variables used in a method of changing control data, refer to the following.

☞ Page 371 Relevant variables

MCv_SetTorqueLimit

To simultaneously change a positive/negative direction torque limit value, use the torque limit value change FB "MCv_SetTorqueLimit".

Item	Description
Function overview	Executes a torque limit value change.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

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

■Function description

- This FB changes a positive/negative direction torque limit value.
- When PositiveValid is TRUE, a positive direction torque limit value is changed to PositiveValue.
- When NegativeValid is TRUE, a negative direction torque limit value is changed to NegativeValue.
- This FB is executed when Execute is TRUE, and Busy turns TRUE when the processing is normally started.
- The torque limit value change timing is set in ExecutionMode.
- When the process is completed and the torque limit value is changed, Done turns TRUE.
- After this FB is started with Execute = "1: mcQueued", if the axis status (AxisName.Md.AxisStatus) turns "0: Disabled"/"1: ErrorStop" before processing completion, the processing will be interrupted and CommandAborted will turn TRUE.
- When an error has occurred in the FB, this FB turns Error to TRUE and stores an error code in ErrorID. For details on error codes, refer to the following.

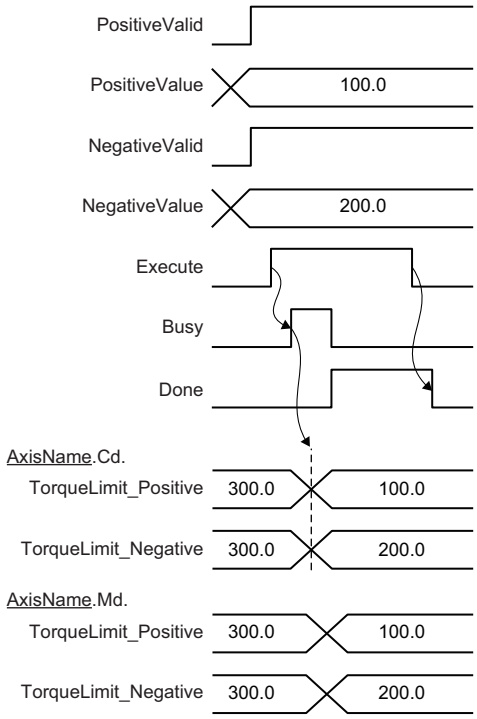
  ☞ Page 662 List of Error Codes
- When both of PositiveValid and NegativeValid are FALSE, the torque limit value will not change and Done will be TRUE.

Precautions

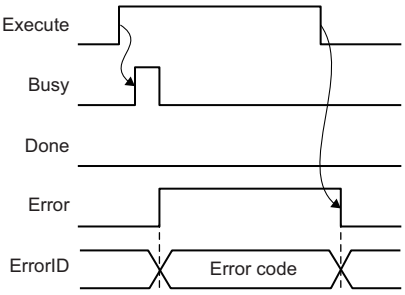
This FB changes AxisName.Cd.TorqueLimit_Positive and AxisName.Cd.TorqueLimit_Negative.

■Timing chart

[When the FB is normally completed]



[When an error occurs]



■I/O variable

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Axis information	Axis	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables

■Input variables

Import ↑: At start, R: Retrigger possible

Name	Variable name	Data type	Import	Setting range	Default value	Description
Execute command	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Positive direction torque limit valid	PositiveValid	BOOL	↑/R	TRUE, FALSE	FALSE	When TRUE, the positive direction torque limit setting value is changed.
Positive direction torque limit value	PositiveValue	LREAL	↑/R	☞ Page 380 Positive/Negative direction torque limit value setting range	0.0	Imports when PositiveValid is TRUE. Sets the positive direction torque limit value.
Negative direction torque limit valid	NegativeValid	BOOL	↑/R	TRUE, FALSE	FALSE	When TRUE, the negative direction torque limit setting value is changed.
Negative direction torque limit value	NegativeValue	LREAL	↑/R	☞ Page 380 Positive/Negative direction torque limit value setting range	0.0	Imports when NegativeValid is TRUE. Sets the negative direction torque limit value.
Execution mode	ExecutionMode	MC_EXECUTION_MODE	↑	0 to 3	0	Specifies the execution timing of the FB. 0: mcImmediately (Execute immediately) 1: mcQueued (Execute at completion) 3: mcSpeculatively (Execute speculatively)
Options	Options	DWORD(HEX)	↑	*1	00000000H	Sets function options with bit designation.

*1 For a bit and the function description, refer to the following table.

Bit	Function description
0 to 15	Empty (Specify "0".)*1
16	Cancel accepted after FB start Specifies whether cancel is allowed or not after FB is started. 0: Not allow 1: Allow
17 to 31	Empty (Specify "0".)*1

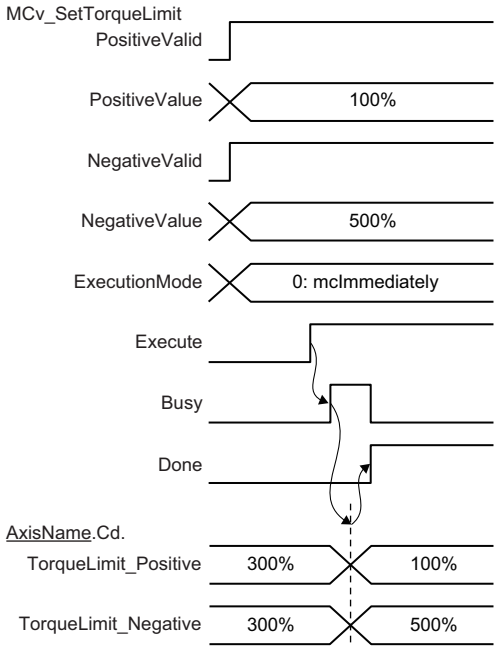
*1 If specifying other than "0", the error "Out of Options Range" (error code: 1A4EH) occurs, and the function does not start.

■Output variables

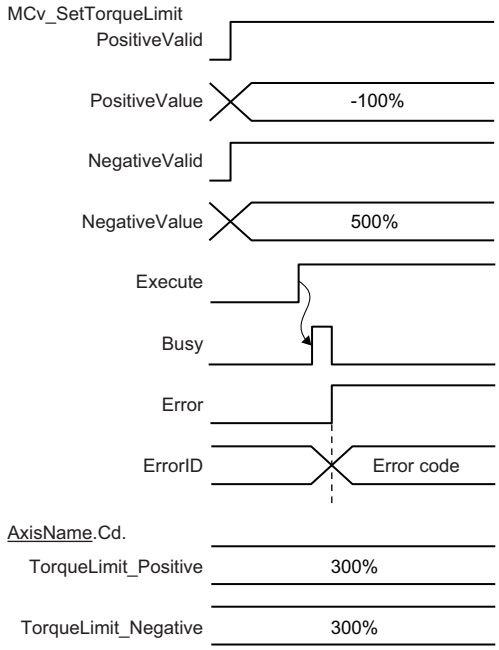
Name	Variable name	Data type	Default value	Description
Execution completion	Done	BOOL	FALSE	Indicates that a torque limit value change is normally completed.
Executing	Busy	BOOL	FALSE	When TRUE, it indicates that the FB is in execution.
Abortion of execution	CommandAborted	BOOL	FALSE	When TRUE, it indicates that execution is interrupted.
Cancel acceptance	CancelAccepted	BOOL	FALSE	Indicates that the FB accepted cancel.
Error	Error	BOOL	FALSE	When TRUE, it indicates that an error has occurred.
Error code	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned. For details, refer to the following. ☞ Page 662 List of Error Codes

■Operation overview

In MCv_SetTorqueLimit, AxisName.Cd.TorqueLimit_Positive and AxisName.Cd.TorqueLimit_Negative are changed.



If a value outside the range is set in PositiveValue or NegativeValue and Execute is turned to TRUE, an error will occur in the FB, and AxisName.Cd.TorqueLimit_Positive and AxisName.Cd.TorqueLimit_Negative will not be changed.



■Details of ExecutionMode

When the torque limit value change FB is executed during axis operation, the change timing differs depending on the ExecutionMode setting value of the input variable.

- Immediate execution

  "0: mcImmediately"

The change is immediately executed regardless of the axis status (AxisName.Md.AxisStatus).

AxisName.Cd.TorqueLimit_Positive and AxisName.Cd.TorqueLimit_Negative are changed at rising edge of Execute.

- Execution after stop

  "1: mcQueued"

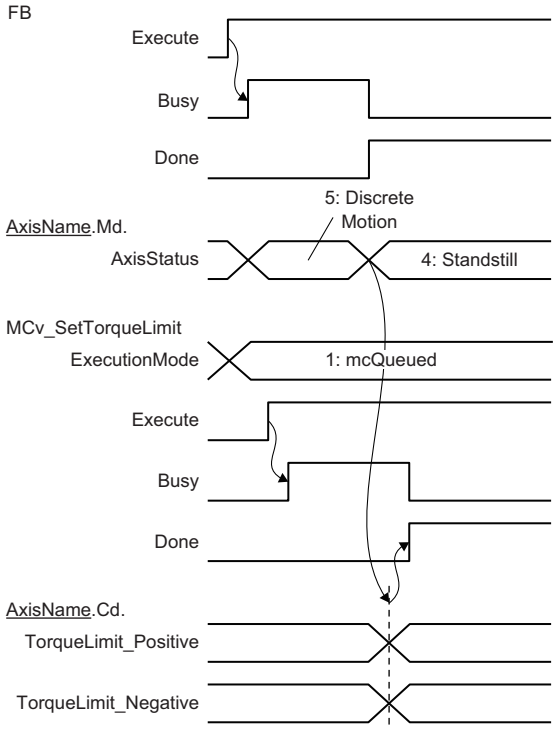
The change is executed after the on-going FB is completed.

Busy turns TRUE at rising edge detection of Execute, and the FB waits until the axis status (AxisName.Md.AxisStatus) turns "4: Standstill".

As soon as the axis status (AxisName.Md.AxisStatus) turns "4: Standstill", AxisName.Cd.TorqueLimit_Positive and AxisName.Cd.TorqueLimit_Negative are changed.

When the axis status (AxisName.Md.AxisStatus) is "4: Standstill" at rising edge detection of Execute, the FB is immediately executed.

If the axis status (AxisName.Md.AxisStatus) turns "0: Disabled"/"1: ErrorStop" before processing is completed, the processing will be interrupted and CommandAborted will turn TRUE.



- Execute speculatively

  "3: mcSpeculatively"

If the axis status (AxisName.Md.AxisStatus) is not "4: Standstill" or if the rising edge of Execute is detected at execution of FB instruction, the error "MCv_SetTorqueLimit Instruction Error" (error code: 3450H) will occur.

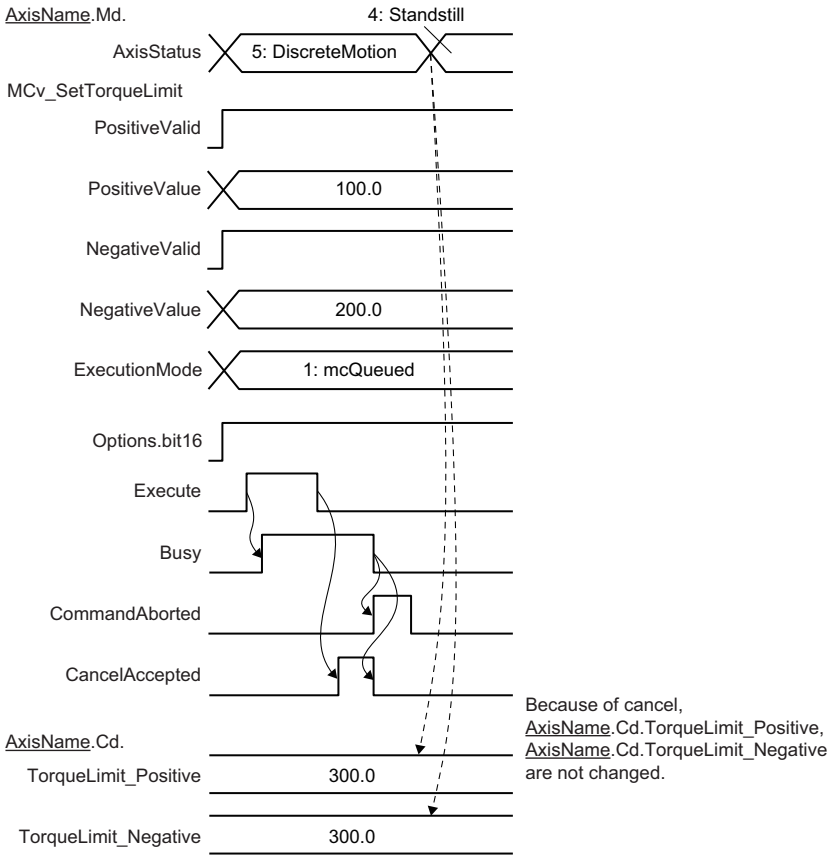
■Cancel

Torque limit value change that is in the standby status after FB execution can be canceled.

- To enable cancel, execute the FB with the status of "Cancel accepted after FB start" (Options bit16) set to "1: Allowed".
- Cancel is started at falling edge detection of Execute.
- Cancel is accepted only while Busy of the output pin is TRUE.
- When FB accepts cancel, CancelAccepted of the output pin will be TRUE.
- When cancel is completed, CommandAborted of the output pin will be TRUE.
- When cancel was carried out, AxisName.Cd.TorqueLimit_Positive and AxisName.Cd.TorqueLimit_Negative are not changed.

Ex.

When canceling while the axis status (AxisName.Md.AxisStatus) is waiting for switching to "4: Standstill" after starting MCv_SetTorqueLimit with "1: mcQueued" of ExecutionMode



How to change control data

Positive/Negative direction torque limit value setting range

The setting range of AxisName.Cd.TorqueLimit_Positive and AxisName.Cd.TorqueLimit_Negative is as the following table.

Control data	Setting range
AxisName.Cd.TorqueLimit_Positive	0.0 (0%) ≤ Setting value ≤ AxisName.PrConst.TorqueLimit_Max
AxisName.Cd.TorqueLimit_Negative	

Torque limit value change timing

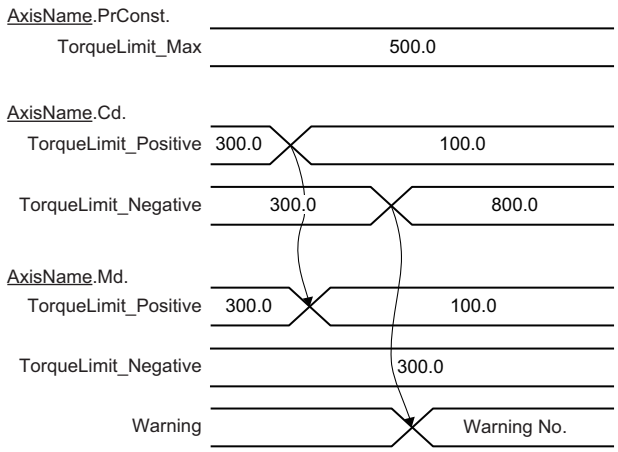
A torque limit value can be changed by changing AxisName.Cd.TorqueLimit_Positive and AxisName.Cd.TorqueLimit_Negative.

If a value outside the setting range is set, the warning "Out of Forward Direction Torque Limit Value Specification Range Warning" (warning code: 0D09H) or "Out of Reverse Direction Torque Limit Value Specification Range Warning" (warning code: 0D0AH) will occur, and the torque limit value will remain unchanged from the previous value.

When the real drive axis is not connected, the warning "Warning Changing Torque Limit Value to Axis not yet Started" (warning code: 0D23H) will occur. At this time, the torque limit value changed when connecting the real drive axis will be enabled because AxisName.Cd.TorqueLimit_Positive and AxisName.Cd.TorqueLimit_Negative are changed.

Ex.

When a warning occurs after AxisName.Cd.TorqueLimit_Negative changes 300.0 to 800.0 in the status where AxisName.PrConst.TorqueLimit_Max = 500.0 is set



11 CONTROL SUB FUNCTIONS

11.1 Compensation Function

Compensation function includes the driver unit conversion function, which can be executed by the parameter settings.

Relevant variables

Variable/Structure name	Name	Details
AxisName.Pr.		
Drive_UnitConvRatioNum	Driver unit conversion numerator	Sets the numerator to convert from the command unit of the motion system to the command unit of the driver.
Drive_UnitConvRatioDen	Driver unit conversion denominator	Sets the denominator to convert from the command unit of the motion system to the command unit of the driver.

Driver unit conversion function

• Real drive axis

"Driver unit conversion function" converts the feed machine position to the driver command value and pass it to the target position (Target position). Also, it converts the current position (Position actual value) of the driver and calculates the actual position.

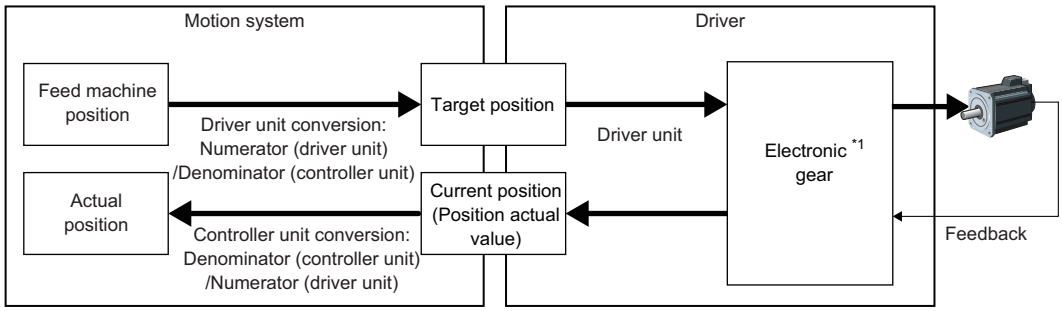
The conversion formulas are shown below.

Target position = Feed machine position (Motion system command value) × Driver unit conversion numerator / Denominator

Actual position = Current position × Driver unit conversion denominator / Numerator

Point

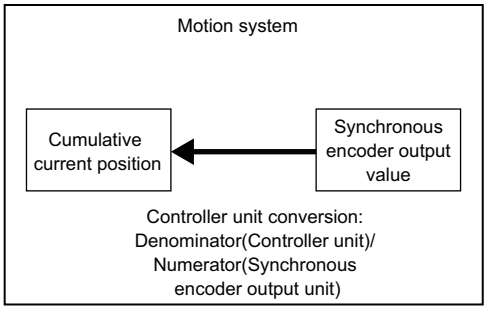
When the driver has the electronic gear, it is recommended to set the ratio to 1:1.



*1 According to the system configuration, if the resolution should be adjusted in the driver side, setting with the electronic gear of the driver side is possible.

• Real encoder axis/Virtual encoder axis

Executes the driver unit conversion to the synchronous encoder output value and calculate the cumulative current position.



Control details

Set AxisName.Pr.Drive_UnitConvRatioNum and AxisName.Pr.Drive_UnitConvRatioDen in order to output the command value to the driver according to the specified movement amount.
The setting examples are shown below.

■For a ball screw

Ex.

When the ball screw pitch is 20 [mm], the controller command unit is [mm], the driver command unit is [pulse], and the motor encoder resolution is 67108864 [pulse/rev]
Set the number of pulses required to rotate a motor once (the encoder resolution) in AxisName.Pr.Drive_UnitConvRatioNum, and set the transfer distance of the machine per motor rotation in AxisName.Pr.Drive_UnitConvRatioDen.
AxisName.Pr.Drive_UnitConvRatioNum = 67108864 [pulses]
AxisName.Pr.Drive_UnitConvRatioDen = 20 [mm]
In this example, the movement amount per pulse is 0.000000298 [mm]. For instance, when the movement amount of 22 [mm] is set, the command will be 73819750.4 [pulses] and a fraction of 0.4 [pulses] will appear. In this case, the movement amount of 67108864 [pulses] is output to the driver, and the fraction is memorized inside the motion system, and positioning is performed by adding this fraction to the movement amount at the next positioning.

■For a linear servo motor

Ex.

When the controller command unit is [μm] and the driver command unit is [pulse], the number of pulses of the linear encoder is calculated with the driver unit conversion numerator and the movement amount is calculated with the driver unit conversion denominator.

$$\text{Linear encoder resolution} = \frac{\text{Number of pulses (driver unit conversion numerator)}}{\text{Movement amount (driver unit conversion denominator)}}$$

Linear encoder resolution: 0.05 [μm]

$$\frac{\text{Number of pulses [pulse]}}{\text{Movement amount [μm]}} = \frac{1}{0.05} = \frac{20}{1.0}$$

AxisName.Pr.Drive_UnitConvRatioNum = 20 [pulses]
AxisName.Pr.Drive_UnitConvRatioDen = 1 [μm]

Point

- Set the driver unit conversion multiplying factor within the following range.
  |The range upper limit of the controller command value × (Driver unit conversion numerator / denominator)| < 8000000000000000000
  For the range upper limit of the controller command value, refer to the following.
  ☞ Page 83 Positioning Range
  If setting a value outside the range, the error "Driver Unit Change Magnification Upper Over" (error code: 1AA4H) will occur.
- A change of the driver unit conversion parameter is also imported when the PLC READY [Y0] is turned ON. Turning ON it in the following status, otherwise, the error "Driver Unit Change Numerator/Denominator Change Timing Incorrect" (error code: 1AA5H) will be output, and the system will be operated with the set value before turning ON the PLC READY [Y0].
  · AxisName.Md.AxisStatus is "0: Disabled"
  · AxisName.Md.FollowupDisable is FALSE

Precautions

- Values less than one which could not be output during machine movement are managed in the motion system, and the cumulated values will be output when the value reaches one pulse or more.
- When machine homing is completed, the function clears the cumulative values of less than one pulse which could not be output to "0".
- When the driver unit conversion numerator/denominator is changed, the homing request flag turns TRUE.
- Double-precision floating-point type variables are used for calculation of the cumulative current value and the target position that is sent to the driver. Therefore, values that can not be expressed within the effective digits of the double precision floating point format (the range of the mantissa part 53 bits) include errors due to rounding processing.
- The target position is calculated by a formula, "cumulative machine position from home position × driver unit conversion numerator / driver unit conversion denominator". Set so that the integral part of the calculation result does not exceed the effective digits range.
  Since the target position sends a value which is obtained by truncating the fractional part of the operation result, errors due to rounding processing is truncated when it is less than or equal to the decimal point. However, when the integer part becomes larger than the effective digit, the command value actually output will also contain rounding error.
- When the cumulative current value per pulse after the driver unit conversion ("1 × driver unit conversion denominator / driver unit conversion numerator") is a small value less than effective digits to the present cumulative current value, the correct command cannot be sent in units of 1 [pulse].
- When the error size is a problem, change the following setting.
  · Reduce the multiplying factor of "driver unit conversion numerator / driver unit conversion denominator".
  For example, when setting 20 [mm] to the machine movement amount per motor rotation, set "20000" in AxisName.Pr.Drive_UnitConvRatioDen, and set [μm] in AxisName.Pr.Unit_Position. (20.0 [mm] → 20000.0 [μm])

11.2 Command Filter

Command filter can perform specified filter processing to inputs of the master axis, and output the result to the slave axis. For example, when setting a virtual drive axis to the master axis and a real drive axis to the slave axis, the result of filter processing to input of the virtual drive axis can be output to the real drive axis.
Command filter includes the following.

Filter type	Overview
Smoothing filter	• Filter function that is used to suppress load-side vibration, such as work-side vibration and base shake. • Frequencies higher than the set frequency can be removed.
Travel direction limit filter	• Filter function that is used to limit the travel direction of the axis. • The input travel value from the master axis can be restricted to one direction. • This helps to avoid reverse operation caused by machine vibration, etc.
Speed limit filter	• Filter function that is used to limit the speed of the axis. • The speed input by the master axis can be limited to the specific speed and output.
Backlash compensation filter	• Filter function that is used to compensate for the backlash amount in the machine system. • Whenever the travel direction changes, outputs the extra commands equivalent to the set up backlash compensation amount.

The command filter control FB is the Enable type of synchronous control FB and has the input variable of FilterDisable (Filter disable) and the output variable of FilterPool (Filter cumulated value).

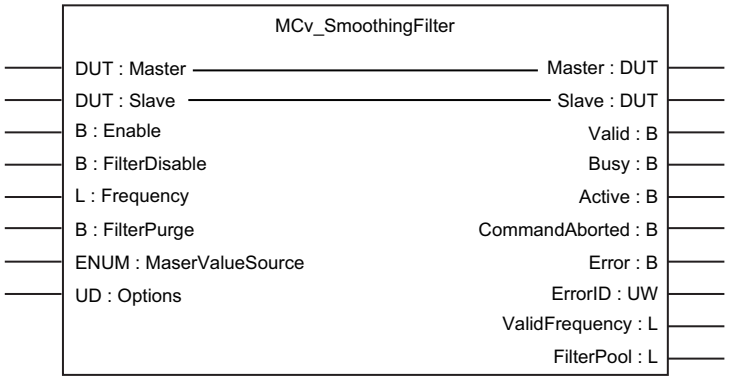
Operation of this function for each system status

○: Possible, ×: Not possible

System status	Function operation
STOP	×
RUN	○
Moderate error	×
Major error	×

Relevant FBs

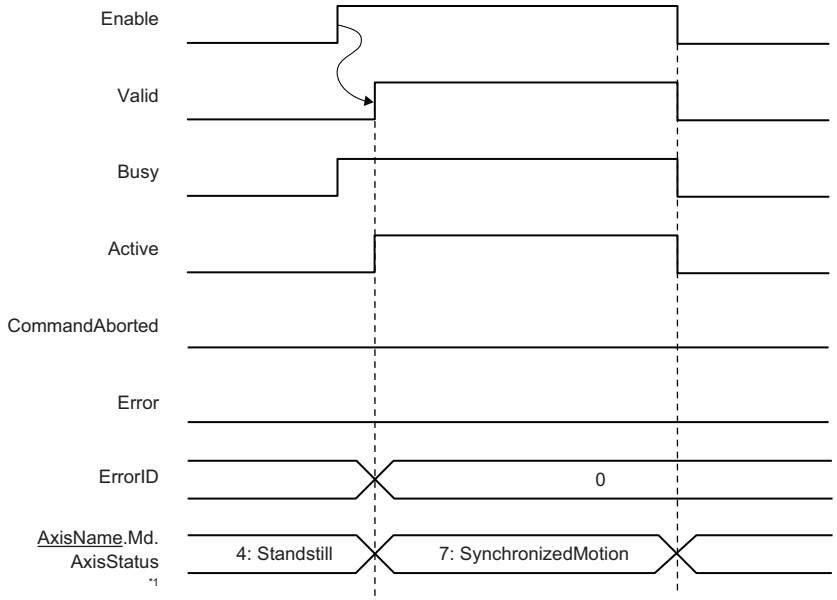
MCv_SmoothingFilter

Item	Description
Function overview	Executes filter processing based on the specified frequency.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
40	24	Subroutine type	Real-time execution

■Timing chart

[When the FB is normally completed]



*1 Indicates the axis status of the slave axis.

[When an error occurs]

For details upon completion with an error, refer to the following.

☞ Page 27 Execute type/Enable type

■I/O variables

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Master axis	Master	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables
Slave axis	Slave	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables

■Input variables

Import ⎍: Always, ↑: At start

Name	Variable name	Data type	Import	Setting range	Default value	Description
Enable	Enable	BOOL	⎍	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Filter disable	FilterDisable	BOOL	⎍	TRUE, FALSE	FALSE	Disables the filter during TRUE, and the master axis input is transmitted to the slave axis as it is.
Frequency	Frequency	LREAL	⎍	0.2 to 250.0 [Hz]	10.0	Sets a frequency that suppresses vibration.
Filter purge	FilterPurge	BOOL	↑	TRUE, FALSE	FALSE	Specifies the end operation with the filter cumulative value.
Master axis data source selection	MasterValueSource	MC_SOURCE	↑	1, 2, 101, 102	1	For details, refer to the following. ☞ Page 440 Master Axis Data Source Selection
Options	Options	DWORD(HEX)	↑	00000000H	00000000H	Set "00000000H". (A value other than "00000000H" causes the error "Out of Options Range" (error code: 1A4EH).)

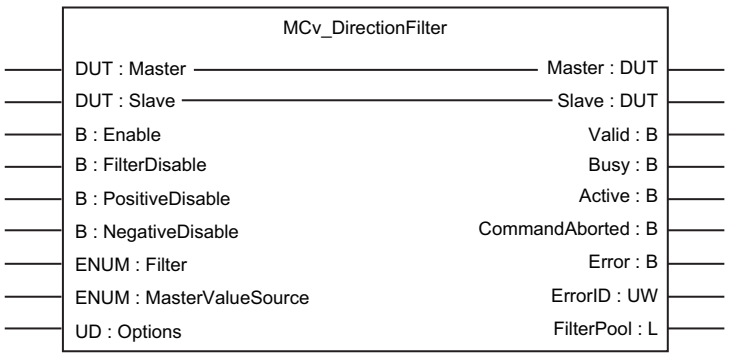
■Output variables

Name	Variable name	Data type	Default value	Description
Output value valid	Valid	BOOL	FALSE	During TRUE, the output value is valid.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Controlling	Active	BOOL	FALSE	Indicates that the FB is controlling the axis.
Abortion of execution	CommandAborted	BOOL	FALSE	Indicates that execution is interrupted.
Error	Error	BOOL	FALSE	When TRUE, it indicates that an error has occurred
Error code	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned. For details, refer to the following. ☞ Page 662 List of Error Codes
Valid frequency	ValidFrequency	LREAL	0.0	Indicates valid frequencies.
Filter cumulated value	FilterPool	LREAL	0.0	Indicates the movement amount cumulated at filter processing.

■Public variables

Name	Variable	Data type	Default value	Description
Disposed filter cumulated value	PurgedFilterPool	LREAL	0.0	Stores disposed FilterPool by turning FilterDisable to TRUE.

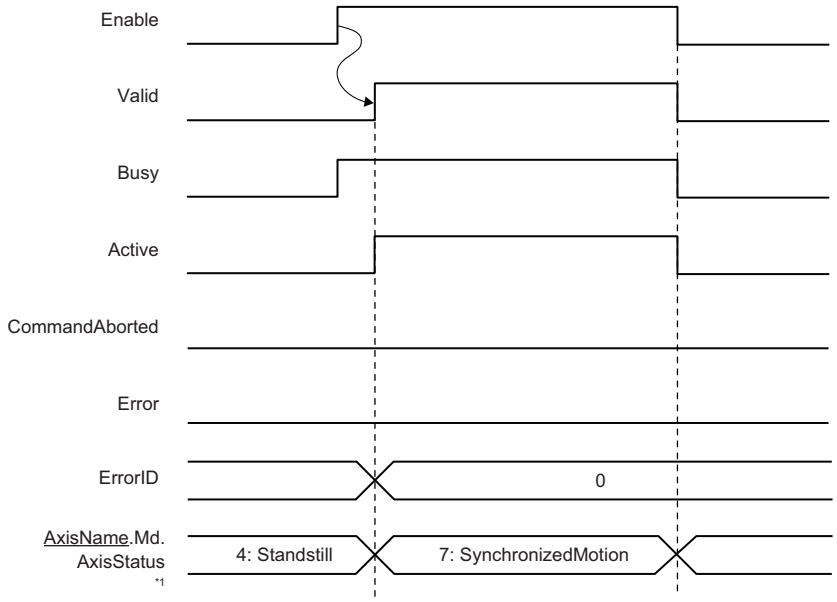
MCv_DirectionFilter

Item	Description
Function overview	Executes filter processing to the travel direction set up to restrict traveling.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

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

■Timing chart

[When the FB is normally completed]



*1 Axis status of the slave axis.

[When an error occurs]

For details when an error occurs, refer to the following.

☞ Page 27 Execute type/Enable type

■I/O variable

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Master axis	Master	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables
Slave axis	Slave	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables

■Input variables

Import ⎍: Always,↑: At start

Name	Variable name	Data type	Import	Setting range	Default value	Description
Enable	Enable	BOOL	⎍	TRUE, FALSE	FALSE	During TRUE, it executes the FB.
Filter disable	FilterDisable	BOOL	⎍	TRUE, FALSE	FALSE	Disables the filter during TRUE, and the master axis input is transmitted to the slave axis as it is.
Positive direction limit	PositiveDisable	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, limit the axis operation to the positive direction.
Negative direction limit	NegativeDisable	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, limit the axis operation to the negative direction.
Filter operation	Filter	MC_VELOCITY_LIMIT_MODE	↑	0, 2, 4	2	Specify the filter operation.
Master axis data source selection	MasterValueSource	MC_SOURCE	↑	1, 2, 101, 102	1	For details, refer to the following. ☞ Page 440 Master Axis Data Source Selection
Options	Options	DWORD(HEX)	↑	00000000H	00000000H	Set "00000000H". (A value other than "00000000H" causes the error "Out of Options Range" (error code: 1A4EH).)

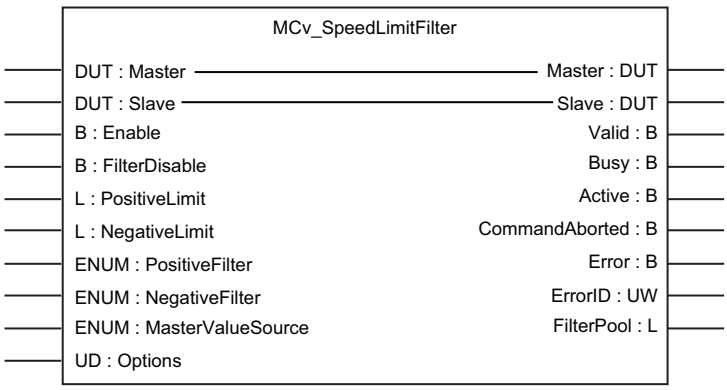
■Output variables

Name	Variable name	Data type	Default value	Description
Output value valid	Valid	BOOL	FALSE	During TRUE, indicates the output value is valid.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Controlling	Active	BOOL	FALSE	Indicates that the FB is controlling the axis.
Abortion of execution	CommandAborted	BOOL	FALSE	Indicates that execution is interrupted.
Error	Error	BOOL	FALSE	When TRUE, it indicates that an error has occurred.
Error code	ErrorID	WORD(UINT)	0	The error code generated in theFB is returned.For details, refer to the following. ☞ Page 662 List of Error Codes
Filter cumulated value	FilterPool	LREAL	0.0	Indicates the movement amount cumulated at filter processing.

■Public variables

Name	Variable name	Data type	Default value	Description
Disposed filter cumulated value	PurgedFilterPool	LREAL	0.0	Stores disposed FilterPool by turning FilterDisable to TRUE.

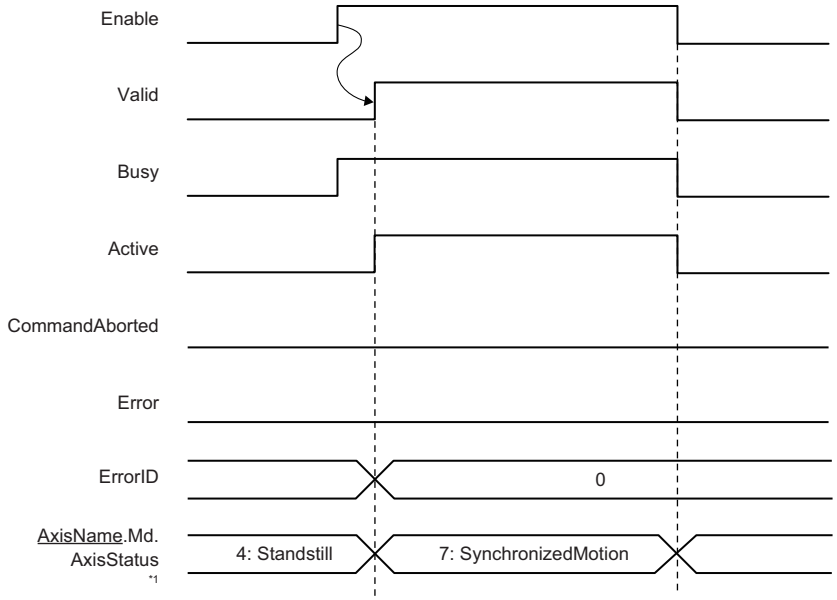
MCv_SpeedLimitFilter

Item	Description
Function overview	Executes filter processing to restrict to the speed of set up limit value.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
52	28	Subroutine type	Real-time execution

■Timing chart

[When the FB is normally completed]



*1 Axis status of the slave axis.

[When an error occurs]

For details when an error occurs, refer to the following.

☞ Page 27 Execute type/Enable type

■I/O variables

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Master axis	Master	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables
Slave axis	Slave	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables

■Input variables

Import ⎍: Always,↑: At start

Name	Variable name	Data type	Imput import	Setting range	Default value	Description
Enable	Enable	BOOL	⎍	TRUE, FALSE	FALSE	During TRUE, it executes the FB.
Filter disable	FilterDisable	BOOL	⎍	TRUE, FALSE	FALSE	Disables the filter during TRUE, and the master axis input is transmitted to the slave axis as it is.
Positive direction limit value	PositiveLimit	LREAL	↑	0, 0.0001to 2500000000.0	0.0	Sets the speed limit value for the positive direction.
Negative direction limit value	NegativeLimit	LREAL	↑	0, 0.0001to 2500000000.0	0.0	Sets the speed limit value for the negative direction.
Positive direction filter operation	PositiveFilter	MC_VELOCITY_LIMIT_MODE	↑	0, 2, 4	2	Sets the filter operation for the direction exceeds the positive direction limit.
Negative direction filter operation	NegativeFilter	MC_VELOCITY_LIMIT_MODE	↑	0, 2, 4	2	Sets the filter operation for the direction exceeds the negative direction limit.
Master axis data source selection	MasterValueSource	MC_SOURCE	↑	1, 2, 101, 102	1	For details, refer to the following. ☞ Page 440 Master Axis Data Source Selection
Options	Options	DWORD(HEX)	↑	00000000H	00000000H	Set "00000000H". (A value other than "00000000H" causes the error "Out of Options Range" (error code: 1A4EH).)

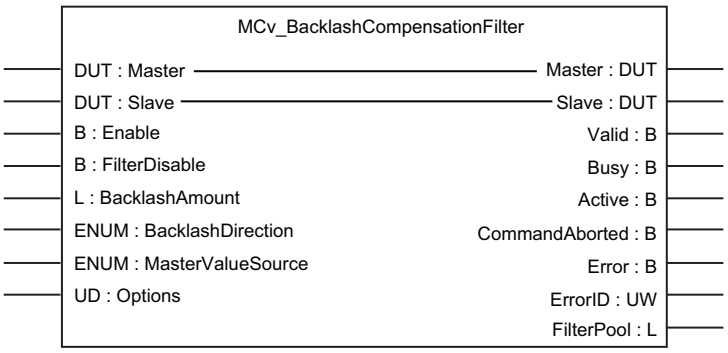
■Output variables

Name	Variable name	Data type	Default value	Description
Output value valid	Valid	BOOL	FALSE	During TRUE, it indicates thatthe output value is valid.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Controlling	Active	BOOL	FALSE	Indicates that the FB is controlling the axis.
Abortion of execution	CommandAborted	BOOL	FALSE	Indicates that execution is interrupted.
Error	Error	BOOL	FALSE	When TRUE, it indicates that an error has occurred.
Error code	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned. For details, refer to the following. ☞ Page 662 List of Error Codes
Filter cumulated value	FilterPool	LREAL	0.0	Indicates the movement amount cumulated at filter processing.

■Public variables

Name	Variable name	Data type	Default value	Description
Disposed filter cumulated value	PurgedFilterPool	LREAL	0.0	Stores disposed FilterPool by turning FilterDisable to TRUE.

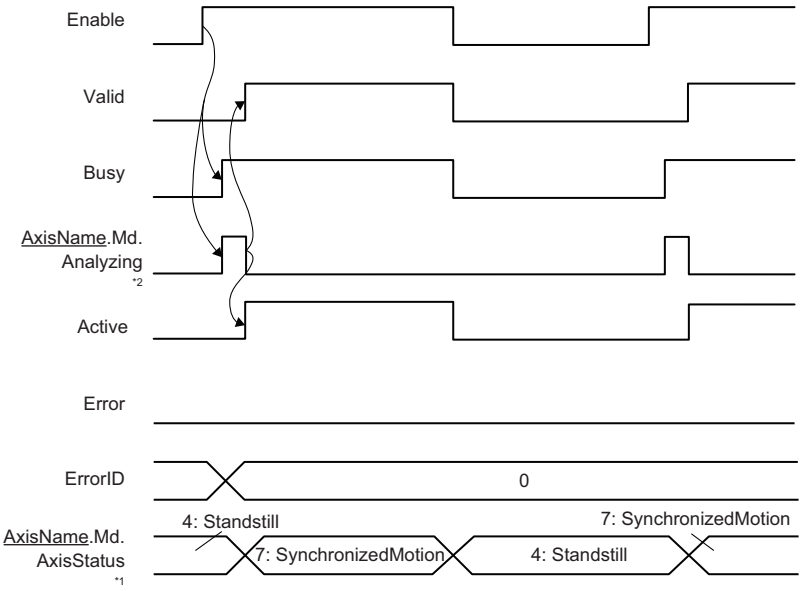
MCv_BacklashCompensationFilter

Item	Description
Function overview	Following the movement direction, executes filter processing to compensate the backlash amount in the machine system.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
40	24	Subroutine type	Real-time execution

■Timing chart

[When the FB is normally completed]



*1 Axis status of the slave axis.

*2 Analyzing the slave axis.

[When an error occurs]

For details when an error occurs, refer to the following.

☞ Page 27 Execute type/Enable type

■I/O variables

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Master axis	Master	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables
Slave axis	Slave	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables

■Input variables

Import ⎍: Always,↑: At start

Name	Variable name	Data type	Import	Setting range	Default value	Description
Enable	Enable	BOOL	⎍	TRUE, FALSE	FALSE	During TRUE, it executes the FB.
Filter disable	FilterDisable	BOOL	⎍	TRUE, FALSE	FALSE	Disables the filter during TRUE, and the master axis input is transmitted to the slave axis as it is.
Backlash amount	BacklashAmount	LREAL	↑	0.0 to 2500000000.0 [U]	0.0	Sets the backlash amount.
Backlash compensation direction	BacklashDirection	MC_DIRECTION	↑	1, 2	0	Sets the backlash compensation direction.
Master axis data source selection	MasterValueSource	MC_SOURCE	↑	1, 101	1	For details, refer to the following. ☞ Page 440 Master Axis Data Source Selection
Options	Options	DWORD(HEX)	↑	00000000H	00000000H	Sets a function option. Set "00000000H". (A value other than "00000000H" causes "Out of Options Range" (error code: 1A4EH).)

■Output variables

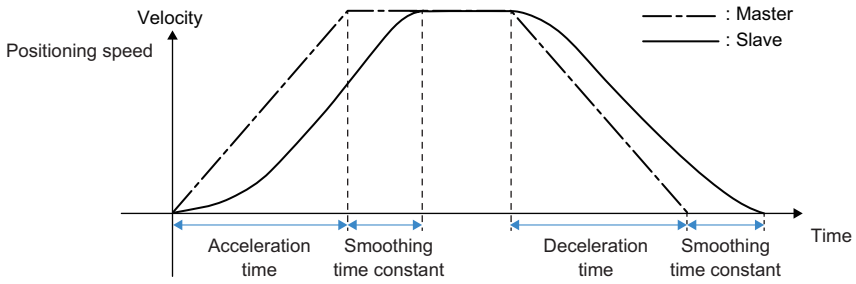
Name	Variable name	Data type	Default value	Description
Output value valid	Valid	BOOL	FALSE	During TRUE, it indicates thatthe output value is valid.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Controlling	Active	BOOL	FALSE	Indicates that the FB is controlling the axis.
Abortion of execution	CommandAborted	BOOL	FALSE	Indicates that execution is interrupted.
Error	Error	BOOL	FALSE	When TRUE, it indicates that an error has occurred
Error code	ErrorID	WORD(UINT)	0	Returns the error code generated in the FB. For details, refer to the following. ☞ Page 662 List of Error Codes
Filter cumulated value	FilterPool	LREAL	0.0	Indicates the movement amount cumulated at filter processing.

■Public variables

Name	Variable name	Data type	Default value	Description
Disposed filter cumulated value	PurgedFilterPool	LREAL	0.0	Stores disposed FilterPool by turning FilterDisable to TRUE.

Smoothing filter

- Smoothing filter removes frequencies higher than the set Frequency, and the entire waveform higher than the setting value becomes a smooth acceleration/deceleration waveform.
- Smoothing filter outputs the result of filter processing that executed to the Master value set in MasterValueSource to Slave.
- The slave axis status (AxisName.Md.AxisStatus) is "7: SynchronizedMotion" during smoothing filter execution.
- The time constant of smoothing is 1/Frequency [s], and the acceleration time and deceleration time become longer by the smoothing time constant. When Frequency is 10 [Hz], the smoothing time constant is 0.1 [s] = 100 [ms].

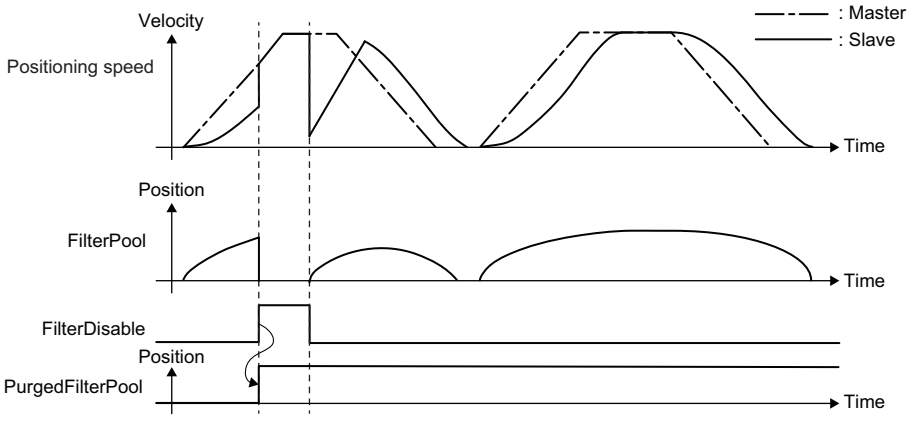


- If a smoothing filter is started while the slave axis is in another operation instruction, the start request will be ignored, and the error "Start Not Possible" (error code: 1AADH) will occur. Start it when the slave axis status (AxisName.Md.AxisStatus) is "4: Standstill".
- MC_Stop is the only function block that can be multiple started during smoothing filter execution.

Details of setting items

■FilterDisable (Filter disable)

- When FilterDisable is FALSE, the axis operates according to the filter setting.
- When FilterDisable is TRUE, the filter is disabled, and Master input is transmitted to Slave.
- When FilterDisable is turned TRUE, FilterPool becomes "0", and stores the disposed FilterPool to the PurgedFilterPool.



■Frequency (Frequency)

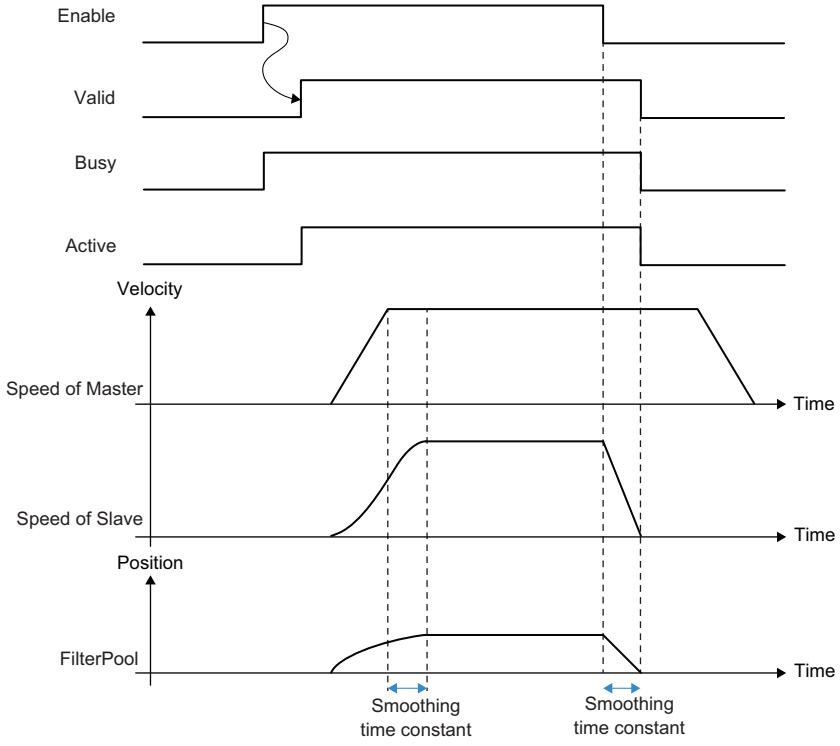
- The following shows the range that can be set in Frequency.

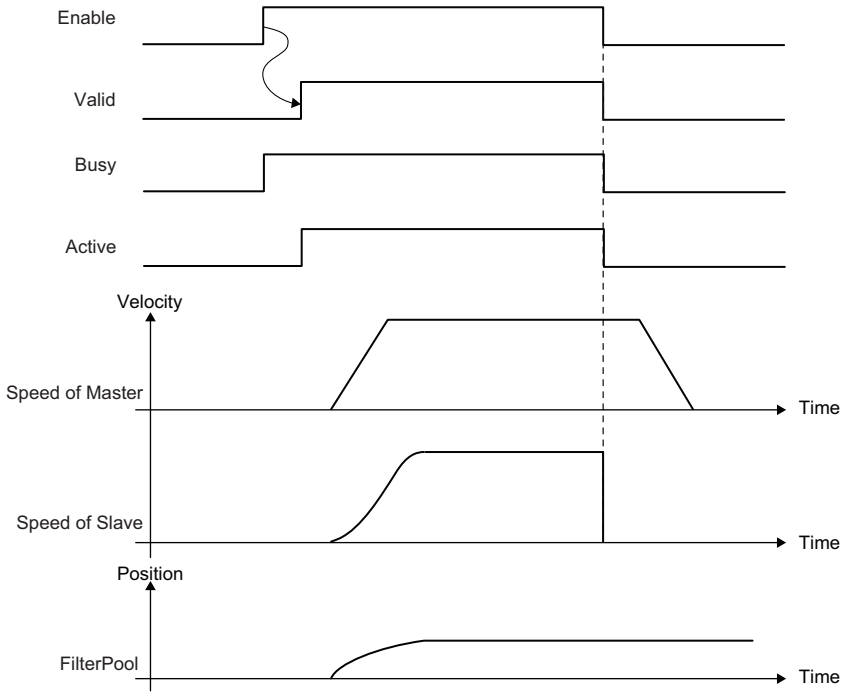
1/Slave axis operation cycle [s]	Setting range
250.00 [Hz] or more	0.20 to 250.00 [Hz]
Less than 250.00 [Hz]	0.20 to (1/Slave axis operation cycle [s]) [Hz]

- If a value outside the range is set in Frequency, the error "Out of Frequency Specification Range" (error code: 1A85H) will occur.
- Valid frequencies can be checked in ValidFrequency.
- To change Frequency, change Frequency when FilterPool is "0". Even if it is changed in a status other than "0", the change will not reflected until FilterPool becomes "0".
- Frequency cannot be changed to a frequency lower than ValidFrequency. If it is set, the error "Out of Frequency Specification Range" (error code: 1A85H) will be output.
- FilterPool may become "0" while the master axis operates with the operation pattern which repeats positive/reverse rotation. When the value of 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, confirm the master axis stop and wait for the filter time constant before changing.

■FilterPurge (Filter purge)

- By FilterPurge setting, operation when Enable turns FALSE can be specified with FilterPool other than "0".

FilterPurge	Operation overview
TRUE	When Enable turns FALSE, the axis operates until FilterPool becomes "0" and stops. 

FilterPurge	Operation overview
FALSE	When Enable turns FALSE, the slave axis immediately stops regardless of the FilterPool status. 

■ValidFrequency (Valid frequency)

- It indicates frequencies which are valid in filter processing.

■FilterPool (Filter cumulated value)

- FilterPool indicates the slave axis movement amount cumulated by filter processing.
- FilterPool is cleared when turning Enable TRUE.
- If the FilterPool value exceeds the positioning range, the warning "Filter Cumulated Value Over Warning" (warning code: 0D13H) will occur. When the warning is detected, the error of FilterPool may be large. For values of the positioning range, refer to the following.

☞ Page 83 Positioning Range

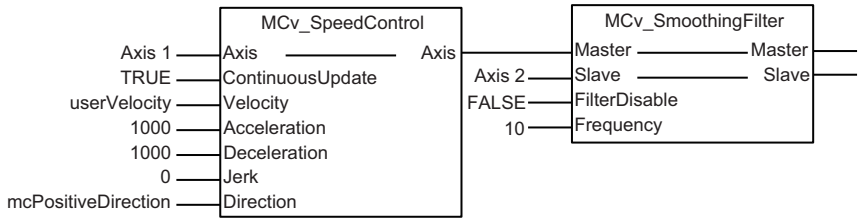
■PurgedFilterPool (Disposed filter cumulated value)

- PurgedFilterPool, when turns FilterDisable to TRUE, displays the disposed FilterPool.
- PurgedFilterPool does not cumulate.
- PurgedFilterPool is 0 cleared at FB start.

Program examples

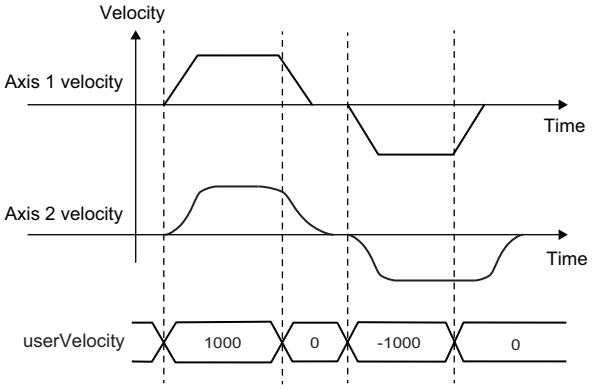
To turn the trapezoidal acceleration/deceleration into smooth acceleration/deceleration waveform

[Program]



[Operation]

As Frequency is set to 10.0, the smoothing time constant will be 0.1 [s].



Precautions

- An operation load increases in smoothing filter as the filter level increases. A rough standard of operation load when the filter level is 5000 is as follows.
  RD78G: Approximately 380 μs
  RD78GH: Approximately 240 μs
- Filter level can be calculated by the following formula. After the decimal point is rounded up/down to the nearest integer. However when the result 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 by the following formula. If memory is insufficient, change maximum RAM size (System.PrConst.Addon_MotionControl_AxisFilter.RamSizeMax) of Addon MotionControl_AxisFilter parameter
  Memory usage = Filter level x 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 change the current value change of absolute position specification.
  When the total sum of the movement amount (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 occurs.

Required objects

For details, refer to the following.

☞ Page 442 Required Slave Object

Movement direction limit filter

- The movement direction limit filter, to the master axis movement, restricts the movement of the slave axis to one direction.
- The movement direction limit filter, outputs the result of filter processing that executed to the Master value set in MasterValueSource to Slave.
- The slave axis status (AxisName.Md.AxisStatus) will be "7: SynchronizedMotion" during movement direction limit filter execution.
- If a movement direction limit filter is started while the slave axis is in another operation instruction, the start request will be ignored, and the error "Start Not Possible" (error code: 1AADH) will occur. Start when the slave axis status (AxisName.Md.AxisStatus) is "4: Standstill".
- MC_Stop is the only function block that can be multiple started during movement direction limit filter execution.
- When turning Enable to FALSE, execute while the slave axis is in stop status. If Enable is turned to FALSE during the slave axis in operation, the slave axis will stop immediately.

Details of setting items

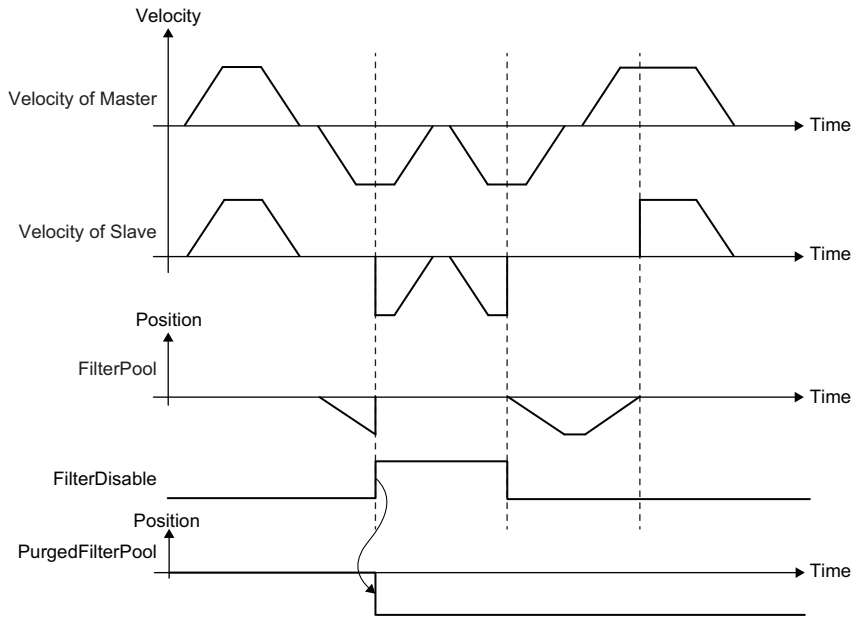
■FilterDisable (Filter disable)

- When FilterDisable is FALSE, the axis operates according to the filter setting.
- When FilterDisable is TRUE, the filter becomes disabled, and Master input is transmitted to Slave as it is.
- When FilterDisable is turned TRUE, FilterPool becomes "0", and stores the disposed FilterPool to the PurgedFilterPool.

Ex.

Examples of the following setting:

Input pin	Setting value
Filter	4: ClampWithoutRamp clamp (Without Ramp at Deceleration)
PositiveDisable	FALSE
NegativeDisable	TRUE

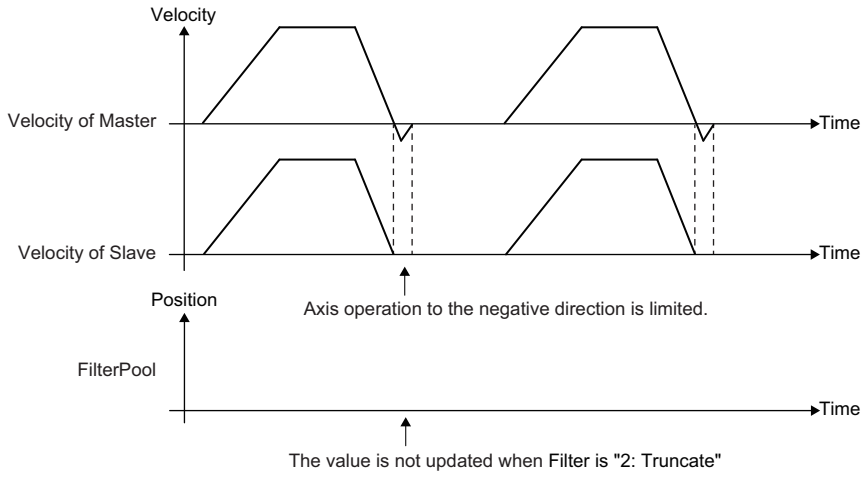
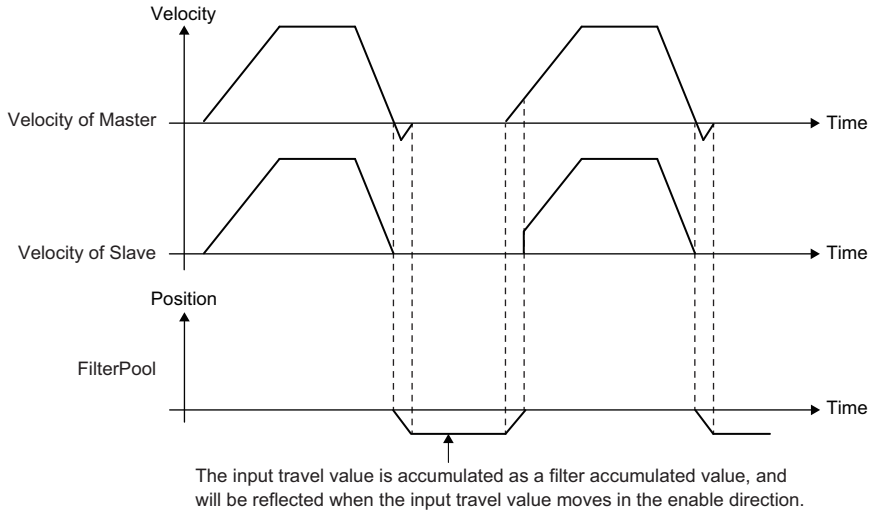


■PositiveDisable (Positive direction limit)/NegativeDisable (Negative direction limit)

- When PositiveDisable is TRUE, the Slave does not operate at Master forward rotation operation.
- When NegativeDisable is TRUE, the Slave does not operate at Master reverse rotation operation.
- When both PositiveDisable and NegativeDisable are set TRUE, the Slave will not operate even if the Master is operated.

■Filter (Filter operation)

• Depending on the Filter setting, set the slave axis operation at control.

Filter	Operation overview
0: Ignore (Ignore)	• Travel direction is not restricted. • FilterPool is not updated
2: Truncate (Truncate)	• The movement amount to the restricted direction is ignored, and not added to FilterPool. • The movement amount to the restricted direction will be truncated, and the position relation between the master axis and the slave axis equivalent of movement amount to the restricted direction will deviate. (Ex.) When PositiveDisable is FALSE, and NegativeDisable is TRUE 
4: ClampWithoutRamp Clamp (Without Ramp at Deceleration)	• Adds movement amount of the restricted direction to FilterPool. • FilterPool, reflects to the reverse direction movement. Thus, even the operation to the restricted direction is repeated, the position relation between the master axis and the slave axis with FilterPool "0" will not deviate. (Ex.) When PositiveDisable is FALSE, and NegativeDisable is TRUE 

• When setting the out of range value to Filter, the error "Out of Filter Operation Specification Range" (error code: 1A86H) is output.

■FilterPool (Filter cumulated value)

• FilterPool indicates the slave axis movement amount cumulated by filter processing.

• When FilterPool status is other than "0", and turns Enable FALSE, FilterPool holds the status at FALSE set. FilterPool, is cleared when turning Enable TRUE next time.

• If FilterPool value exceeds the positioning range, the warning "Filter Cumulated Value Over Warning" (warning code: 0D13H) is output. When the warning is detected, the error of FilterPool may increase. For values of the positioning range, refer to the following.

☞ Page 83 Positioning Range

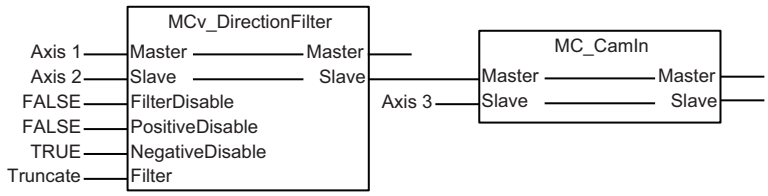
■PurgedFilterPool (Disposed Filter cumulated value)

- PurgedFilterPool, when turns FilterDisable TRUE, displays the disposed FilterPool.
- PurgedFilterPool does not cumulate.
- PurgedFilterPool is 0 cleared at FB start.

Program examples

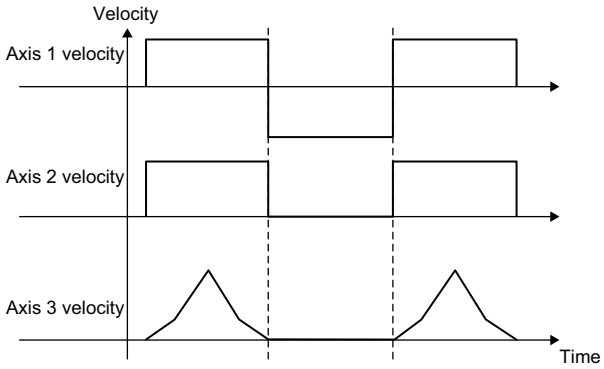
Cam operation direction restriction

[Program]



[Operation]

When MCv_DirectionFilter Master moves to the negative direction, Slave does not move to the negative direction.



Required objects

For details, refer to the following.

☞ Page 442 Required Slave Object

Speed limit filter

- The speed limit filter, to the input velocity of the master axis, sets a specified control value, and outputs the speed of the set control value to the slave axis.
- The speed limit filter, outputs the result of filter processing that executed to the Master value set in MasterValueSource to Slave.
- The slave axis status (AxisName.Md.AxisStatus) will be "7: SynchronizedMotion" during speed limit filter execution.
- If a speed limit filter is started while the slave axis is in another operation instruction, the start request will be ignored, and the error "Start Not Possible" (error code: 1AADH) will occur. Start when the slave axis status (AxisName.Md.AxisStatus is "4: Standstill".
- MC_Stop is the only function block that can be multiple started during speed limit filter execution.
- When turning Enable to FALSE, execute while the slave axis is in stop status. If Enable is turned to FALSE during the slave axis in operation, the slave axis will stop immediately.

Details of setting items

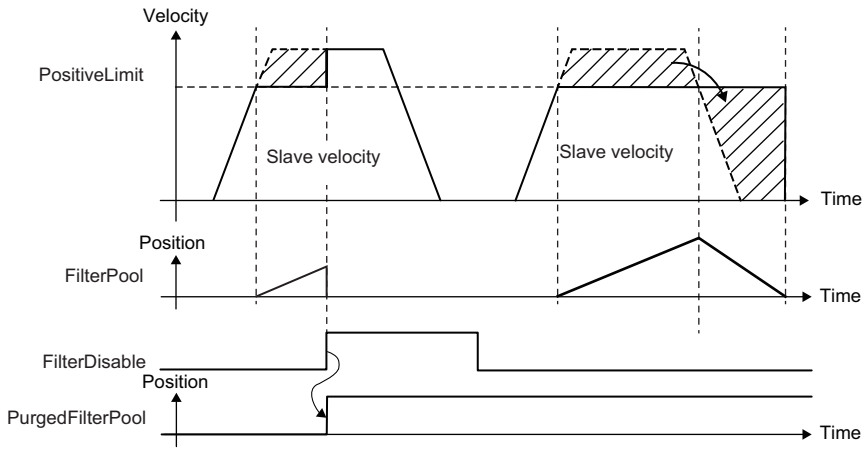
■FilterDisable (Filter disable)

- When FilterDisable is FALSE, the axis operates according to the filter setting.
- When FilterDisable is TRUE, the filter becomes disabled, and Master input is transmitted to Slave.
- When FilterDisable is turned TRUE, FilterPool becomes "0", and stores the disposed FilterPool to PurgedFilterPool.

Ex.

Examples of the following setting:

Input pin	Setting value
PositiveFilter	4: ClampWithoutRamp clamp (Without Ramp at Deceleration)

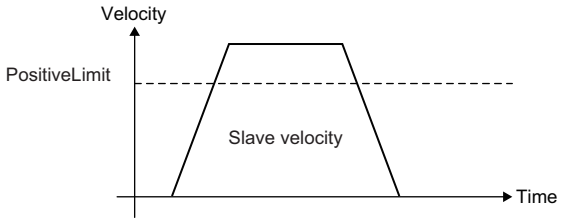
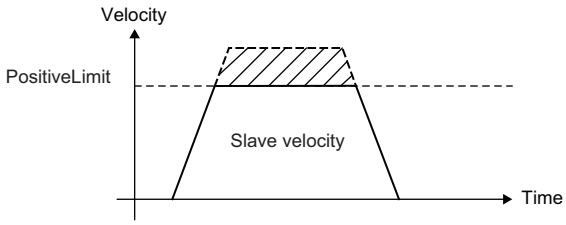
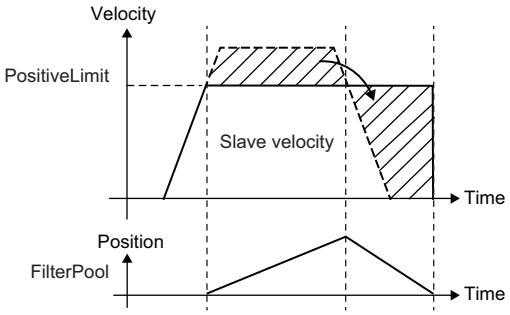
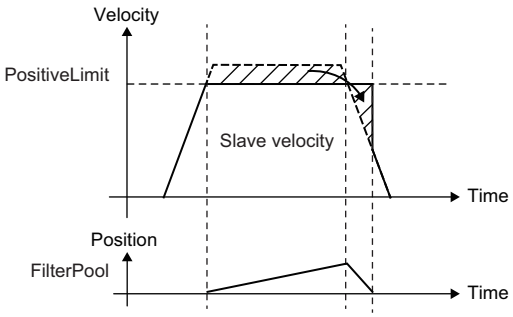


■PositiveLimit (Positive direction limit)/NegativeLimit (Negative direction limit)

- PositiveLimit, to the Master positive direction input speed, sets the upper limit value of the speed output to the Slave.
- NegativeLimit, to the Master negative direction input speed, sets the upper limit value of the speed output to the Slave.
- When "0" is set to the control value, the slave axis does not operate to the applicable direction.
- When out of range value is specified to PositiveLimit/NegativeLimit, the error "Out of Direction Limit Value Specification Range" (error code: 1A87H) is output.

■PositiveFilter (Positive direction filter operation)/NegativeFilter (Negative direction filter operation)

- Depending on PositiveFilter and NegativeFilter setting, set the slave axis operation at control.

Filter operation	Operation overview
0: Ignore (Ignore)	• Speed is not limited. • FilterPool is not updated. 
2: Truncate (Truncate)	• The exceeded control value, is not output to the Slave. • The restricted movement amount will be truncated, and that amount, the position relation between the master axis and the slave axis will deviate. 
4: ClampWithoutRamp (Clamp) (Without Ramp at Deceleration)	• The exceeded control value is cumulated as FilterPool, and output later. FilterPool output does not operates deceleration. • When 0 is set to the control value, all the master axis input to the applicable direction will be cumulated as FilterPool, and will not output later to the slave axis. The cumulated amount will be subtracted by the reverse direction master axis input, and will not operate until FilterPool is 0. [When Filter cumulated value is greater than the decelerated distance]  [When Filter cumulated value is smaller than the decelerated distance] 

- When out of range value is set to PositiveFilter or NegativeFilter, the error "Out of Filter Operation Specification Range" (error code: 1A86H) is output.

■FilterPool (Filter cumulated value)

- FilterPool indicates the slave axis movement amount cumulated by filter processing.
- When FilterPool status is other than "0", and turns Enable FALSE, FilterPool holds the status at FALSE set. FilterPool, is cleared when turning Enable TRUE next time.
- If the FilterPool value exceeds the positioning range, the warning "Filter Cumulated Value Over Warning" (warning code: 0D13H) will occur. When the warning is detected, the error of FilterPool may increase. For values of the positioning range, refer to the following.

☞ Page 83 Positioning Range

■PurgedFilterPool (Disposed filter cumulated value)

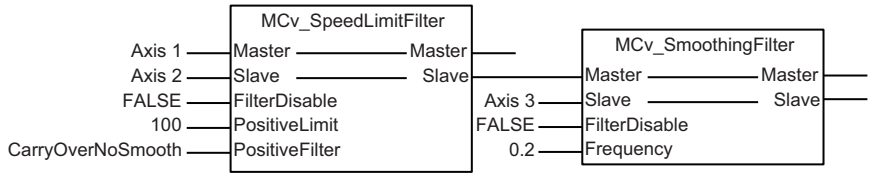
- PurgedFilterPool, when turns FilterDisable TRUE, displays the disposed FilterPool.
- PurgedFilterPool does not cumulate.
- PurgedFilterPool, is 0 cleared at FB start.

Program examples

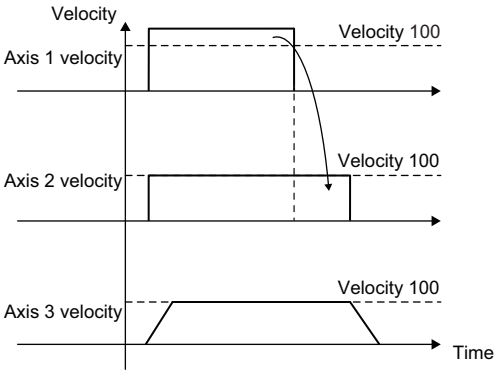
Synchronous encoder speed limit

[Program]

The axis 1 sets the synchronous encoder.



[Operation]



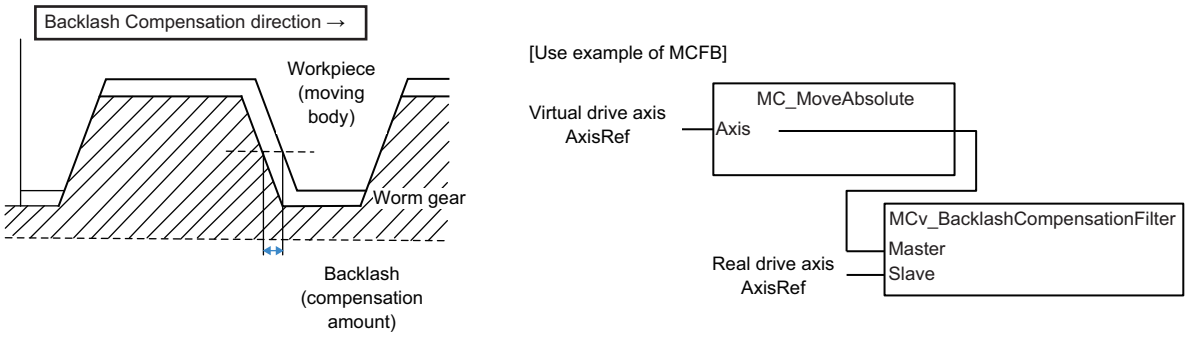
Required objects

For details, refer to the following.

☞ Page 442 Required Slave Object

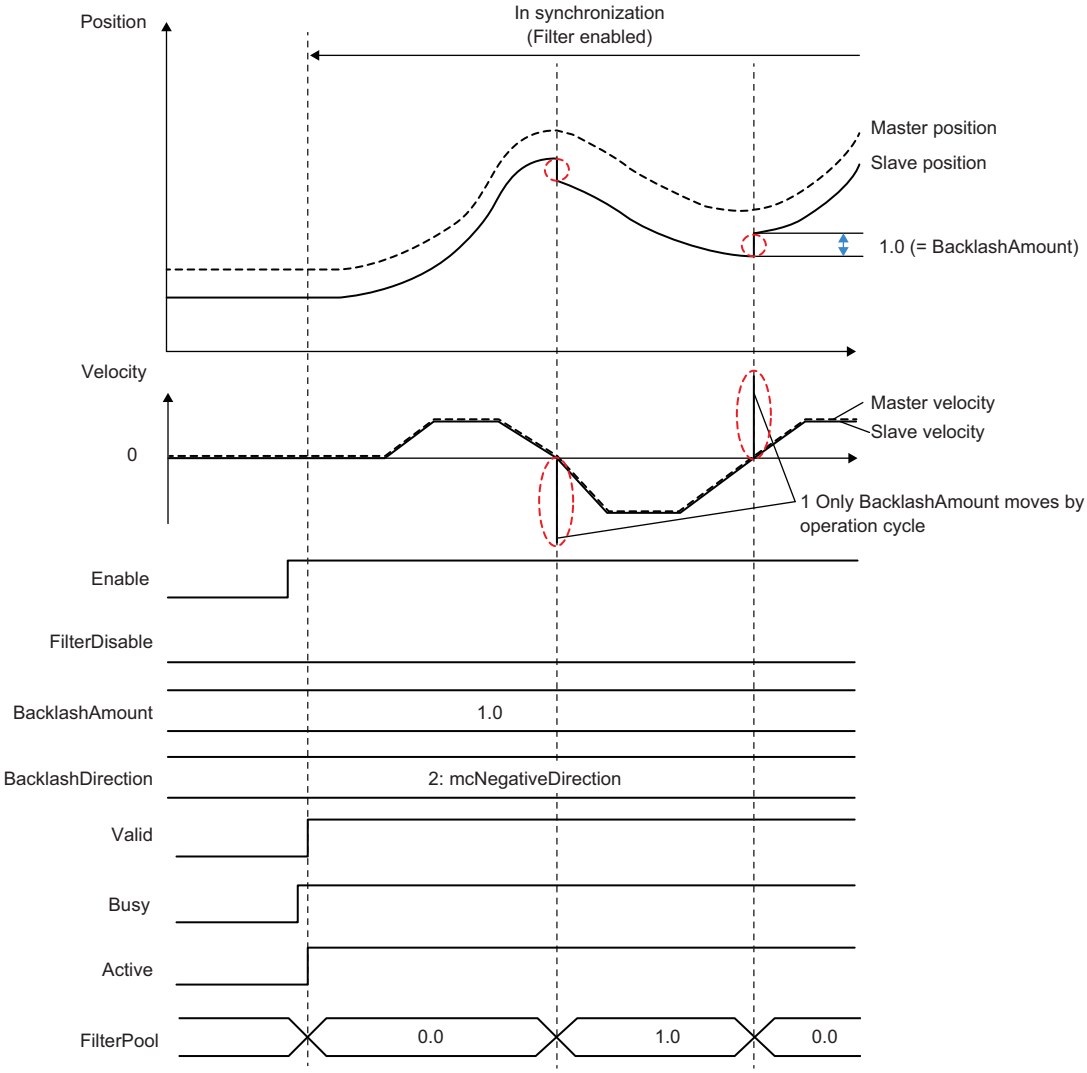
Backlash compensation filter

Backlash compensation filter, is used to compensate the backlash amount in the machine system.



When the travel direction of Master and BacklashDirection are same, Slave synchronizes to Master current direction shifts to the position equivalent to BacklashAmount amount to the BacklashDirection direction. When the travel direction of Mater and BacklashDirection are in reverse, Slave synchronizes to the Master current position.

Whenever Master travel direction changes, extra Slave movement amount occurs only to BacklashAmount, and executes the backlash compensation.



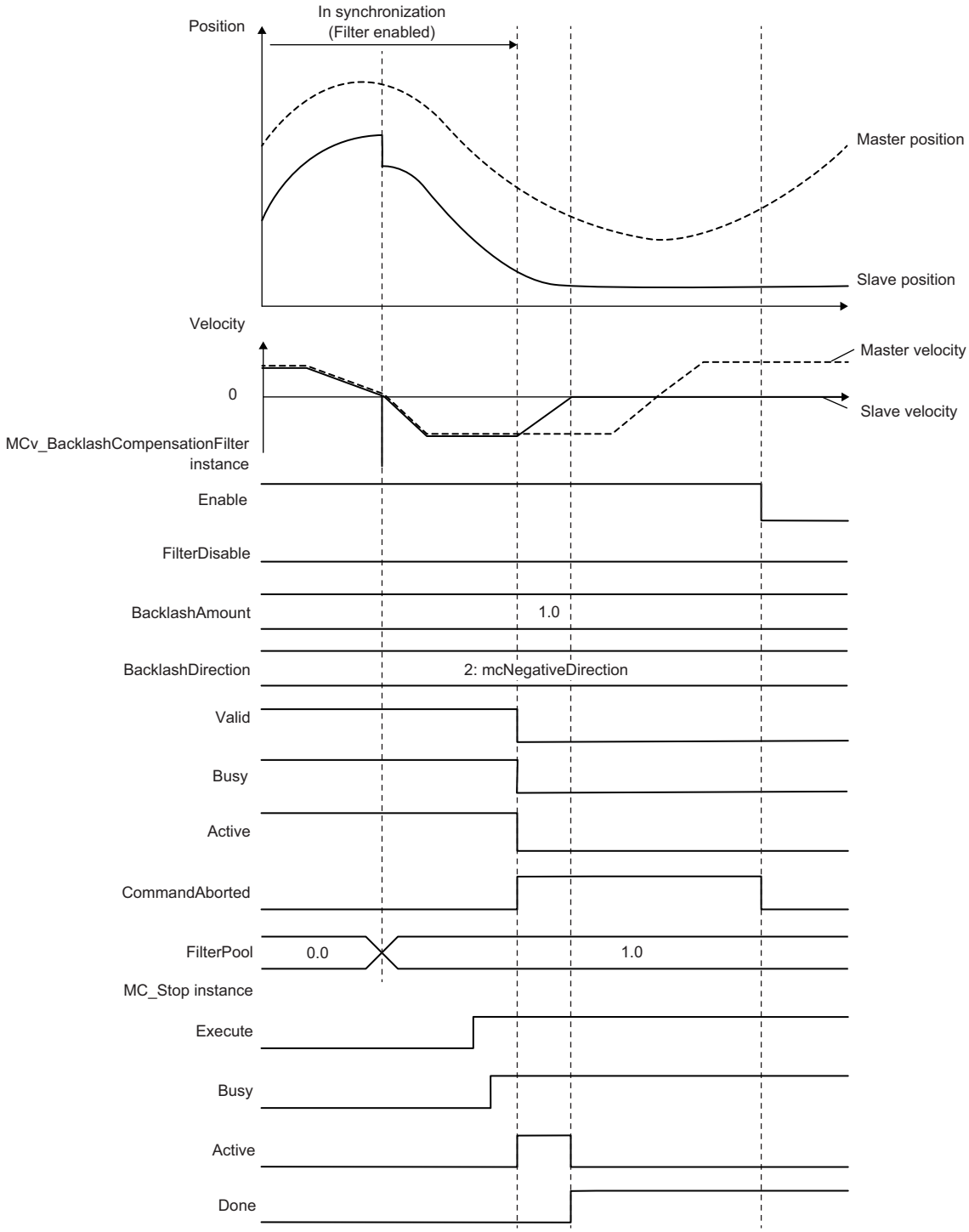
• Stop operation

For details, refer to the following.

☞ Page 441 Start and Stop Operation

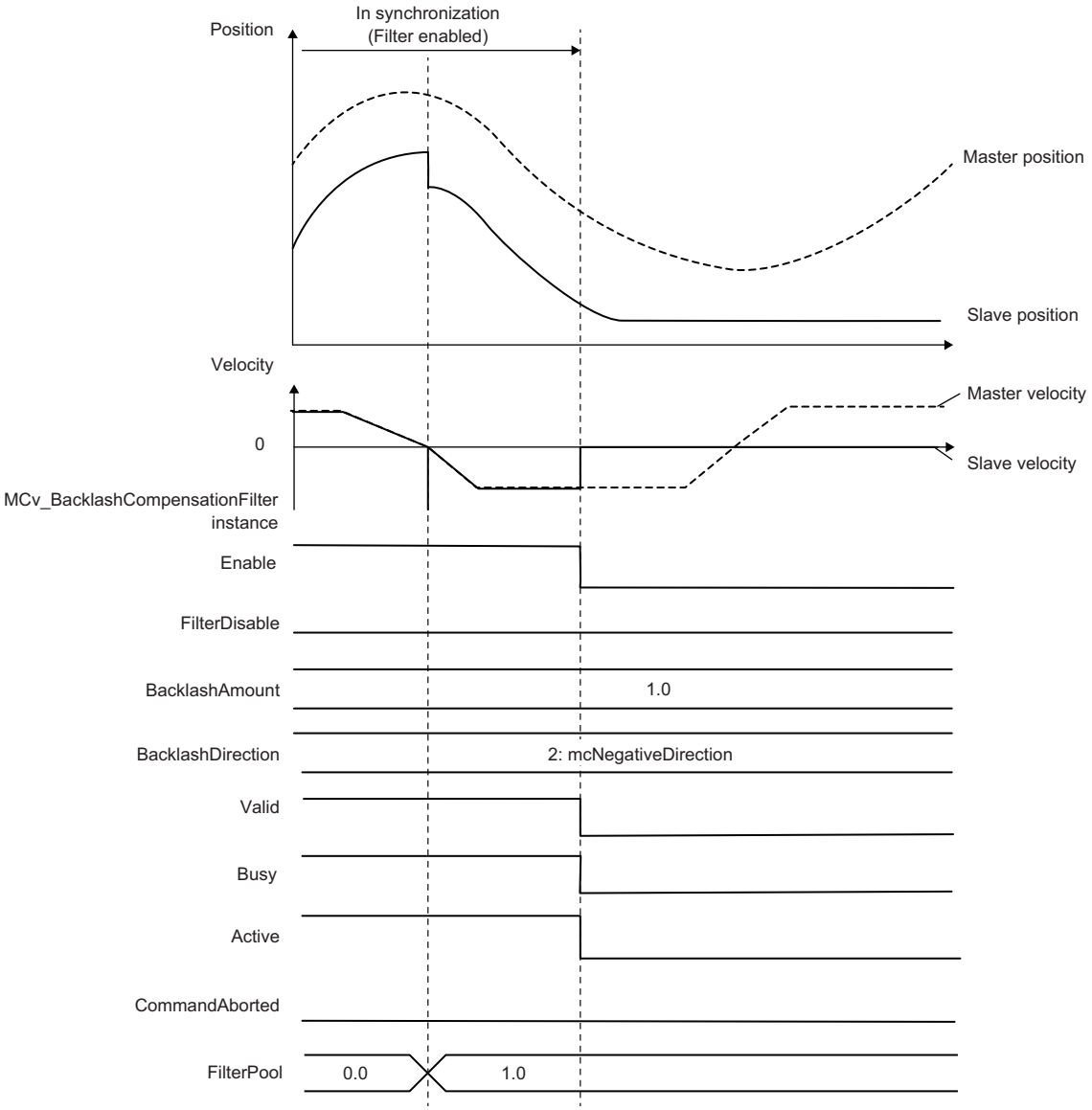
This section describes the spcific operation of the backlash compensation filter.

When stop factor occurs to Slave, Slave starts the stop operation from the synchronous position. The movement amount (FilterPool value) by the compensation will not be restored.



• Operation at Filter disabled (At Enable trailing edge)

Slave stops immediately at the synchronous position when Enable trail edged. The travel amount (FilterPool value) by the compensation will not be restored.



Details of setting items

■FilterDisable (Filter disable)

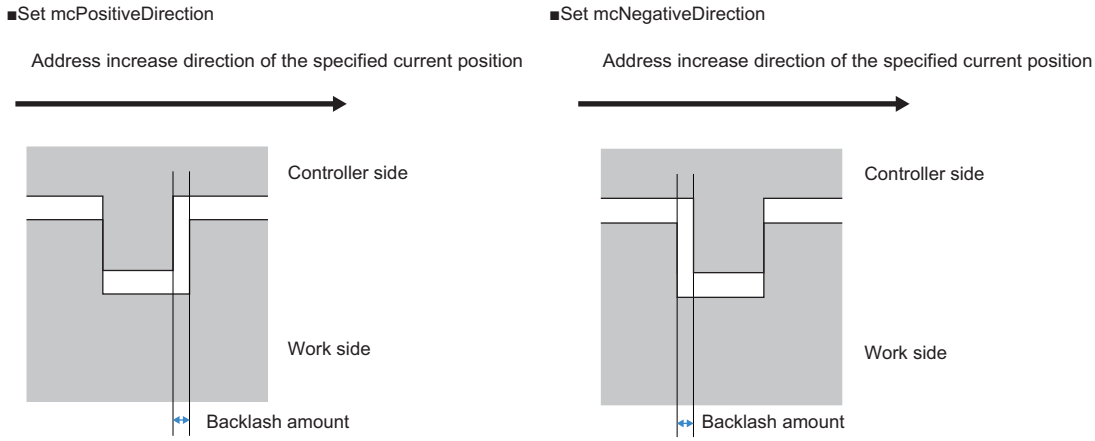
- When FilterDisable is FALSE, the axis operates according to the filter setting.
- When FilterDisable is TRUE, the filter becomes disabled, and Master input is transmitted to Slave.
- When FilterDisable is turned TRUE, FilterPool becomes "0", and stores the disposed FilterPool to the PurgedFilterPool.

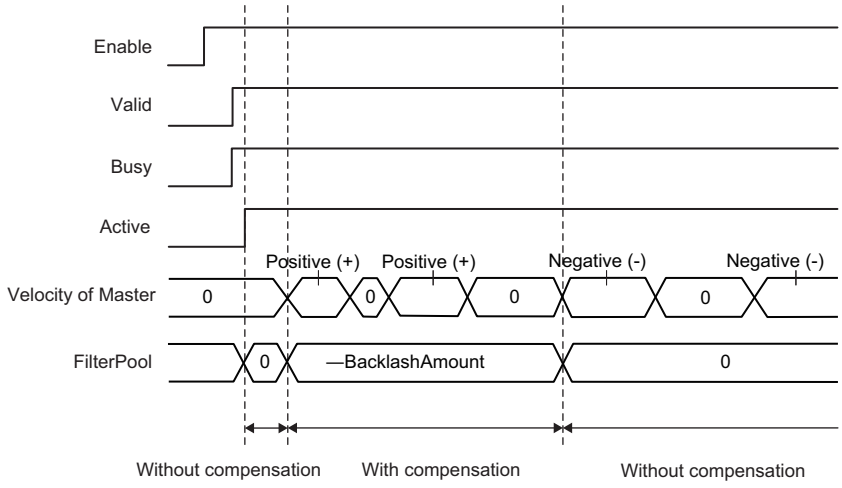
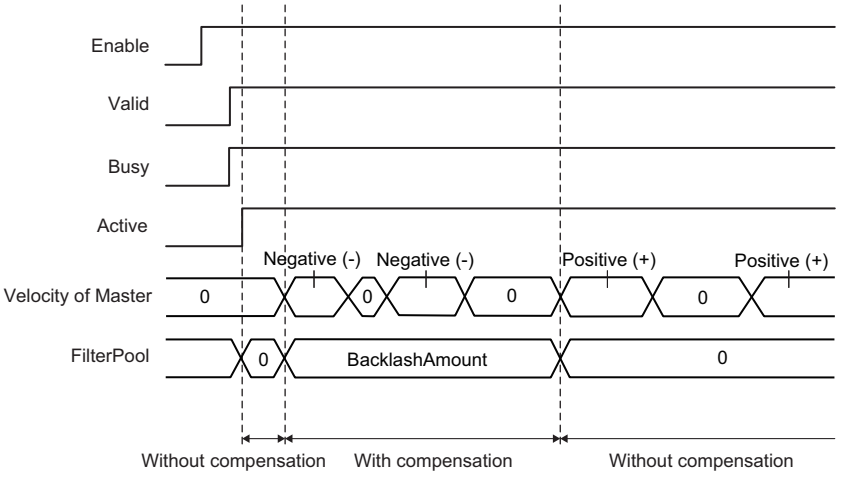
■BacklashAmount (Backlash amount)

- Sets the backlash amount of the machine system.

■BacklashDirection (Backlash compensation direction)

- Specifies the backlash compensation direction. Outputs the compensation amount to Slave when Master movement direction is BacklashDirection. When backlash compensation filter is enabled (Started with FilterDisable FALSE, or trail edged FilterDisable during execution), compensates to the address increase direction of the Master command current position that the backlash status of the machine system is in the following relation.



BacklashDirection	Operation overview
1: mcPositiveDirection (Positive direction)	• When the travel direction of Master is positive direction, Slave synchronizes to Master current direction shifts to the positive direction position equivalent to BacklashAmount. 
2: mcNegativeDirection (Negative direction)	• When the travel direction of Master is negative direction, Slave synchronizes to Master current direction shifts to the negative direction position equivalent to BacklashAmount. 

■FilterPool (Filter cumulate value)

- FilterPool, describes the movement amount of the slave axis cumulated by the filter processing.
  For backlash filter, the reversed value of FilterPool positive/negative will be the compensation amount transmitted to the slave axis.

■Options (Option)

Sets the function option to be used by specifying bit.

For bit and function description, refer to the following.

Bit	Function description
0 to15	Empty (Specify "0".)*1
16	Enable filter initial status selection*1 Selects the initial status of the backlash compensation filter at enable. 0: First compensation
17 to 31	Empty (Specify "0".)*1

*1 If specifying other than "0", the error "Out of Options Range" (error code: 1A4EH) occurs and the function does not start.

- Enable filter initial status selection

Selects the initial status of the backlash compensation filter at enable(When started with FilterDisable FALSE, or trail edged FilterDisable during execution).

Setting value	Description
0: First compensation	Despite Master current direction at filter enable, the filter operation starts following the position relation of Master and Slave current position do not add the backlash compensation.

■PurgedFilterPool (Disposed filter cumulated value)

- PurgedFilterPool, when turns FilterDisable TRUE, displays the disposed FilterPool.
- PurgedFilterPool does not cumulate.
- PurgedFilterPool, is 0 cleared at FB start.

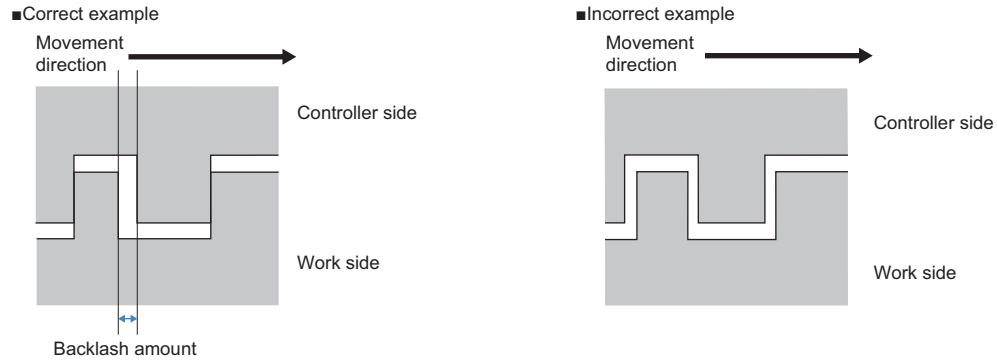
Backlash compensation filter enable procedure

Execute after the machine position is established by homing, or restart the backlash compensation filter after cancelling midway.

Ex.

The example is shown below.

1. Move the real drive axis by homing or JOG operation etc., and keep the status as backlash of the machine system has generated to one direction as an correct example shown below.

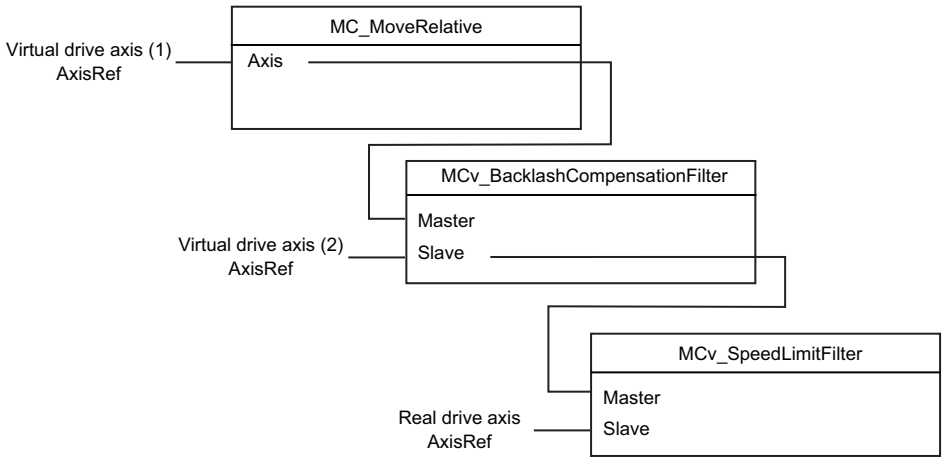


2. Due to the current position change etc., set Master command current position corresponds to the real drive axis command current position.
3. Set the reverse direction of the direction that the real drive axis in Ex1. has moved to, and start the backlash compensation filter. Also start the back part function block if any.

Program example

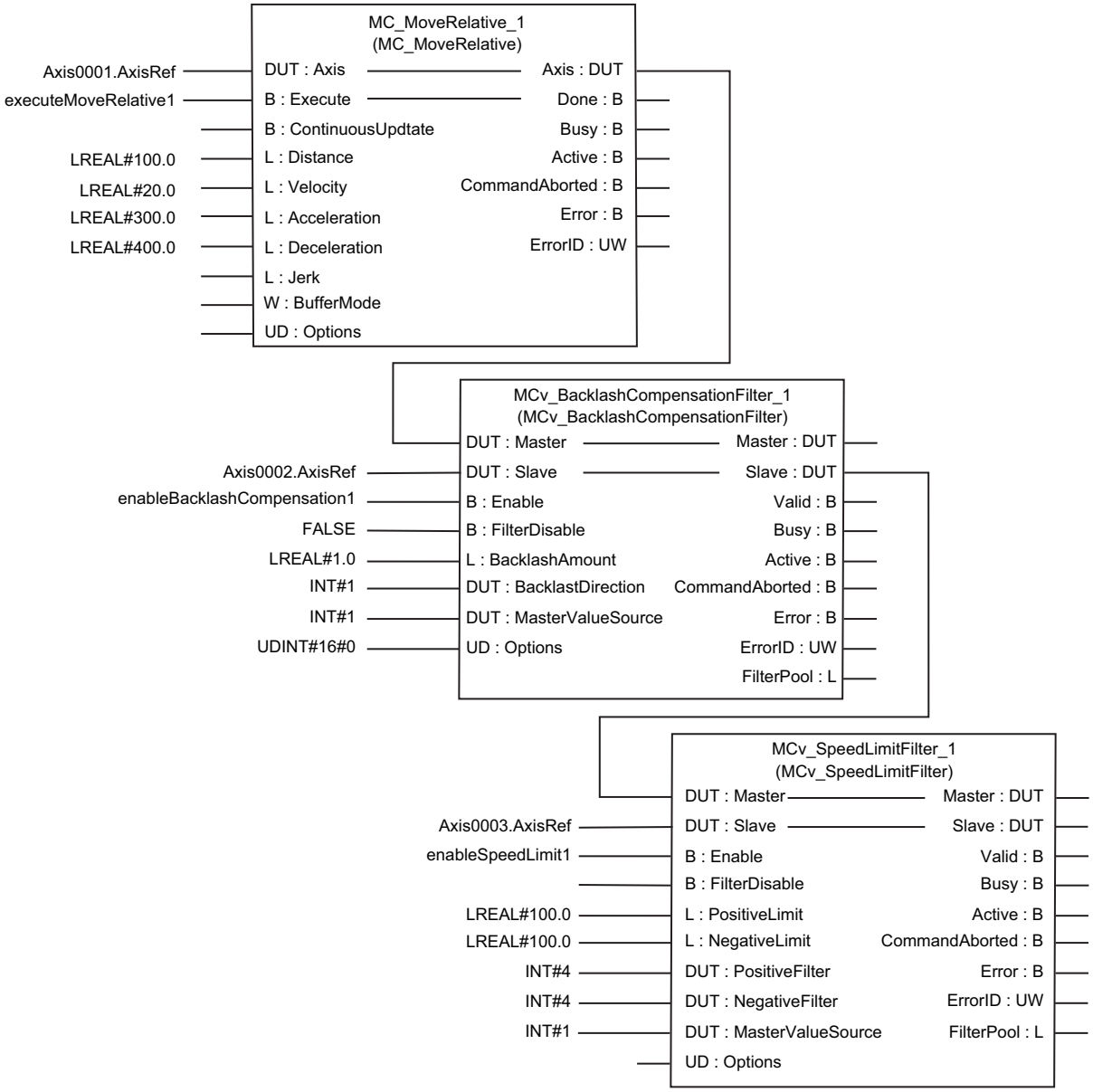
The usage examples of the backlash compensation filter in the program is shown below.

In the example the backlash compensation filter back part is connected with the speed limit filter as configured below.



Axis	Axis name for the program example	Comment
Virtual drive axis (1)	Axis0001	• Executes the motion control such as positioning control, and generates the command to output to the drive device. • Indicates the load-side position (before speed limit).
Virtual drive axis (2)	Axis0002	• When the backlash of the machine and the motor side should be compensated by the backlash compensation filter, it will indicate the motor side position (before speed limit).
Real drive axis	Axis0003	• Indicates the motor side position (after speed limit). • Outputs the command to the drive device.

■Sequence program



The following should be defined by users.

- executeMoveRelative1
- enableBacklashCompensation1
- enableSpeedLimit1

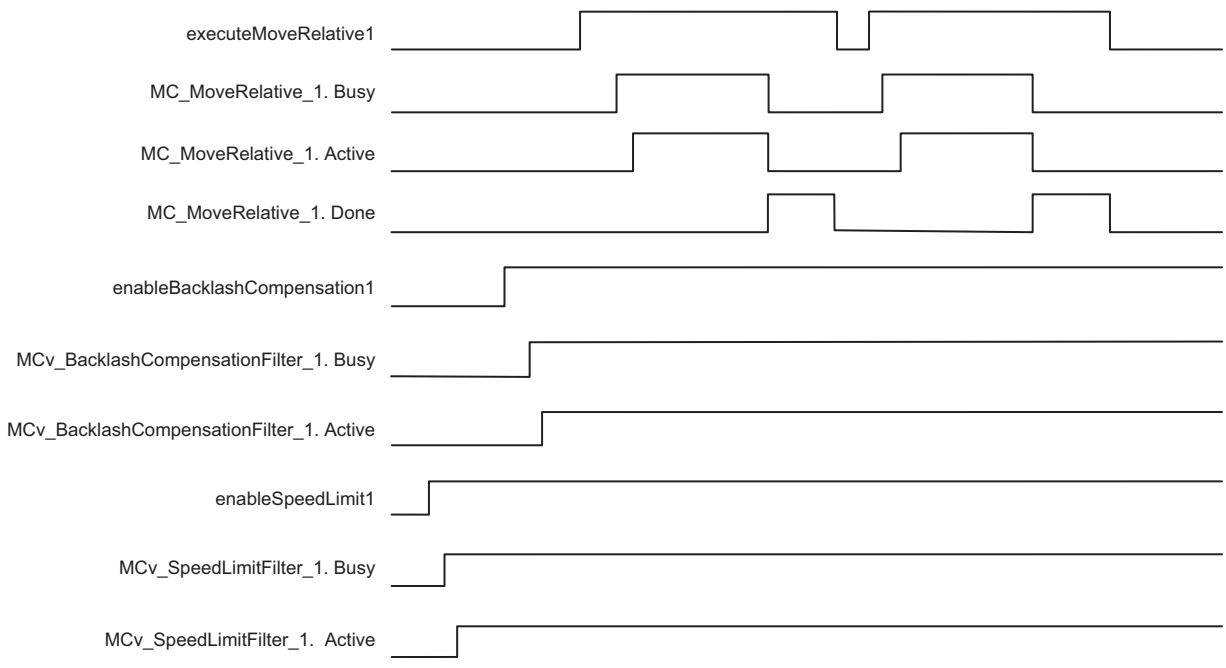
■Structured text (ST language)

```
MC_MoveRelative_1.Axis := Axis0001.AxisRef;
MC_MoveRelative_1.Execute := executeMoveRelative1;
MC_MoveRelative_1.Distance := LREAL#100.0;
MC_MoveRelative_1.Velocity := LREAL#20.0;
MC_MoveRelative_1.Acceleration := LREAL#300.0;
MC_MoveRelative_1.Deceleration := LREAL#400.0;
MC_MoveRelative_1();
MCv_BacklashCompensationFilter_1.Master := MC_MoveRelative_1.Axis;
MCv_BacklashCompensationFilter_1.Slave := Axis0002.AxisRef;
MCv_BacklashCompensationFilter_1.Enable := enableBacklashCompensation1;
MCv_BacklashCompensationFilter_1.FilterDisable := FALSE;
MCv_BacklashCompensationFilter_1.BacklashAmount := LREAL#1.0;
MCv_BacklashCompensationFilter_1.BacklashDirection := INT#1;
MCv_BacklashCompensationFilter_1.MasterValueSource := INT#1;
MCv_BacklashCompensationFilter_1.Options := UDINT#0;
MCv_BacklashCompensationFilter_1();
MCv_SpeedLimitFilter_1.Master := MCv_BacklashCompensationFilter_1.Slave;
MCv_SpeedLimitFilter_1.Slave := Axis0003.AxisRef;
MCv_SpeedLimitFilter_1.Enable := enableSpeedLimit1;
MCv_SpeedLimitFilter_1.PositiveLimit := LREAL#100.0;
MCv_SpeedLimitFilter_1.NegativeLimit := LREAL#100.0;
MCv_SpeedLimitFilter_1.PositiveFilter := INT#4;
MCv_SpeedLimitFilter_1.NegativeFilter := INT#4;
MCv_SpeedLimitFilter_1();
```

The following should be defined by users.

- executeMoveRelative1
- enableBacklashCompensation1
- enableSpeedLimit1

■Timing chart



Precautions

- The backlash compensation is a function which the command direction and the backlash status should not change from the external force etc. The correct compensation will not be executed when used such as below.
  A machine which always weighed by external force unidirectionally such as upper and lower axis.
  A machanism (such as rack and pinion mechanism) which the backlash amount changes depending on the machine position.
- Only Slave BacklashAmount moves by one operation cycle at the time of Master movement direction changes. Depending on the value, it may exceed the command frequency that the speed limit value and the driver unit can accept, connect the speed limit filter or the smoothing filter to the backlash compensation filter back part.
- Slave current position (such as command current position, machine feed position etc.) will be the position that BacklashAmount compensation is added. BacklashAmount compensation will not be added to the Master current position.

Required objects

For details, refer to the following.

☞ Page 442 Required Slave Object

Precautions

- Note that if FilterDisable is switched during operation, Slave velocity may change rapidly.
- After enabling the command filter, when Master command is not transmitted to Slave or the back part synchronous control FB by stop factor occurs to Slave etc., the synchronous position relation deviates. Before enabling the command filter again, execute the synchronous positioning as needed.

11.3 Input Variable Change in Execution

- In the control whose start condition is Execute, data such as Position and Velocity can be re-imported without interrupting the operation by Execute retrigger.
- When ContinuousUpdate is TRUE at start, data such as Position and Velocity are imported at each execution cycle during operation.

This section describes the operation when changing the following input variables as a typical example of input variable change.

- Target position/movement distance
- Set velocity
- Acceleration/deceleration, Acceleration/deceleration time

For details on input variables that can be changed and the operation when changing input variables other than the above, refer to details on each FB.

Relevant variables

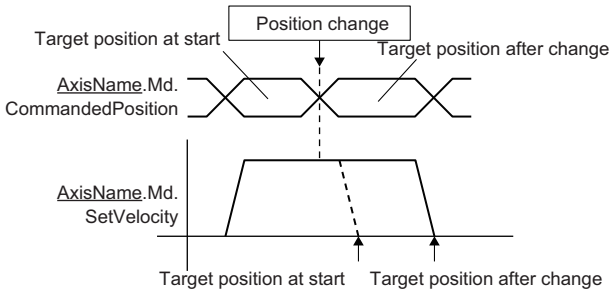
Variable/Structure name	Name	Details
AxisName.Pr.		
OverrunOperation	Operation setting at overrun	Sets operation when an overrun occurs during operation. 1: ImmediateStop (Immediate Stop) 2: KeepCurrentAcc (Keep Current Acceleration/Deceleration)
AxesGroupName.Pr.		
OverrunOperation	Operation setting at overrun	Sets operation when an overrun occurs during operation. 1: ImmediateStop (Immediate Stop)

Target position/movement distance change

This function changes the target position and the movement distance.

Target position change

The target position is changed.

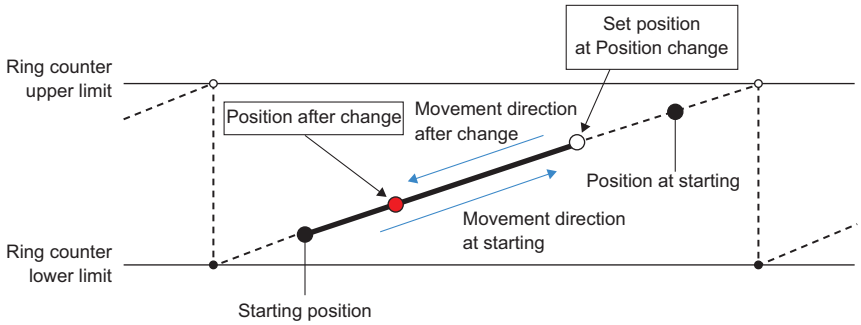


When Direction can be set, the operation when changing the target position differs depending on the setting value.

■For the positive direction/negative direction/current direction

When the target position is changed, the target position will be calculated not from the set position at the change but from the start position and the Direction set direction.

When the target position is changed to between "start position and the set position at the target position change", the operation will be reversed to the target position after change, and the control will be performed according to the selection when having reverse rotation permission selection (Options bit5).



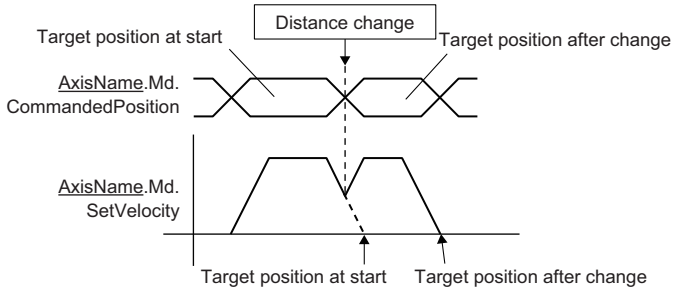
■For the shortest path

When the target position is changed, the operation will be performed in the close direction from the set position at change to the target position after change.

Movement distance change

The movement distance from the start position is changed.

The movement distance after change is calculated not based on the set position at change but based on the start position.



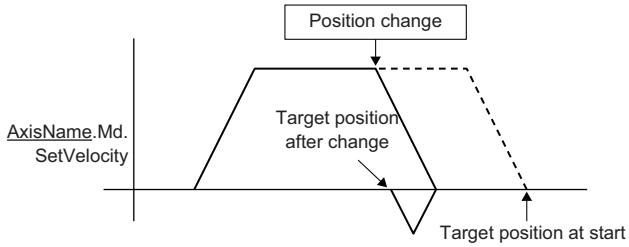
If the movement distance after change is shorter than the moved distance at movement distance change execution, the operation will be reversed, and the control will be performed according to the selection when having reverse rotation permission selection (Options bit5).

The deceleration distance is not kept by the target position/movement distance change

The control is performed according to the selection when having AxisName(AxesGroupName).Pr.OverrunOperation and reverse rotation permission selection (Options bit5).

■When the reverse rotation is allowed

When "0: Allow" is set to the reverse rotation permission selection (Options bit5), a deceleration stop is performed once regardless of the AxisName.Pr.OverrunOperation setting. After the deceleration stop is completed, operation is started to the changed target position.

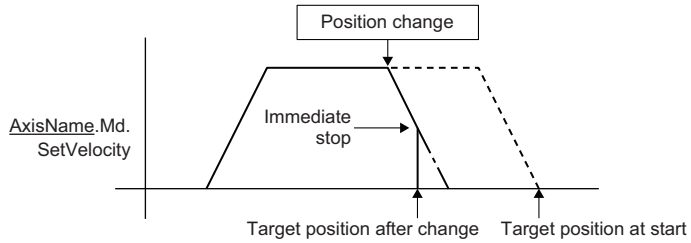


■When the reverse rotation is not allowed

When "1: Not allow" is set to the reverse rotation permission selection (Options bit5), the axis operates according to the AxisName.Pr.OverrunOperation setting.

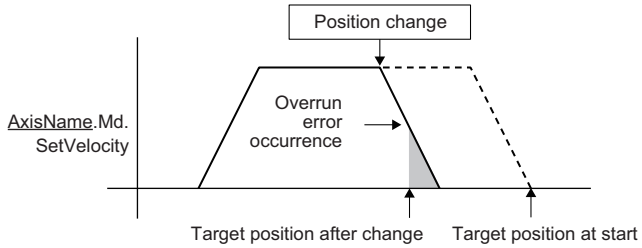
• When "1: ImmediateStop" is set

Deceleration is started, and the axis immediately stops after the warning "Overrun Warning" (warning code: 0D10H) is output when it reaches the stop address.



• When "2: KeepCurrentAcc" is set

The deceleration is started, and the axis exceeds the stop address and stops. When the axis exceeds the stop address, the error "Overrun Error" (error code: 1A7EH) is output.However, the operation stops according to the setting of AxisName.Pr.StopMode_General when the error occurs. To stop the operation with the address which exceeds the stop address, set "2: KeepCurrentAcc" in AxisName.Pr.StopMode_General.

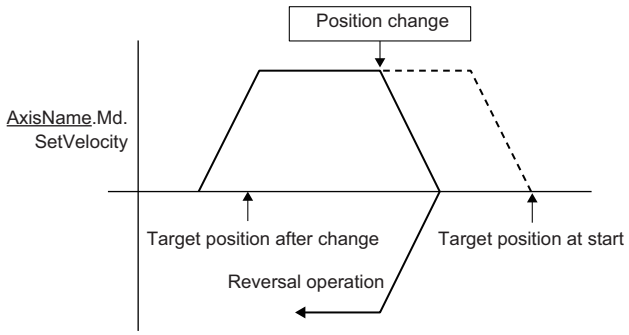


When an operation is reversed by changing the target position/movement distance

When an operation is reversed by changing the target position/movement distance, the axis is controlled according to the selection when having reverse rotation permission selection (Options bit5).

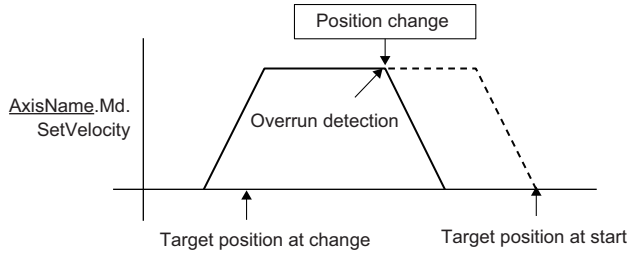
■When the reverse rotation is allowed

When "0: Allow" is set to the reverse rotation permission selection (Options bit5), a deceleration stop is performed once. After the deceleration stop is completed, operation is started to the changed target position.



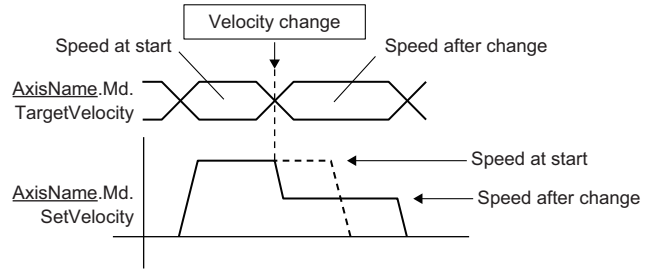
■When the reverse rotation is not allowed

When "1: Not allow" is set to the reverse rotation permission selection (Options bit5), a deceleration stop is performed regardless of the AxisName.Pr.OverrunOperation setting. When the target position/movement distance is changed, the error "Overrun Error" (error code: 1A7EH) is detected.



Set velocity change

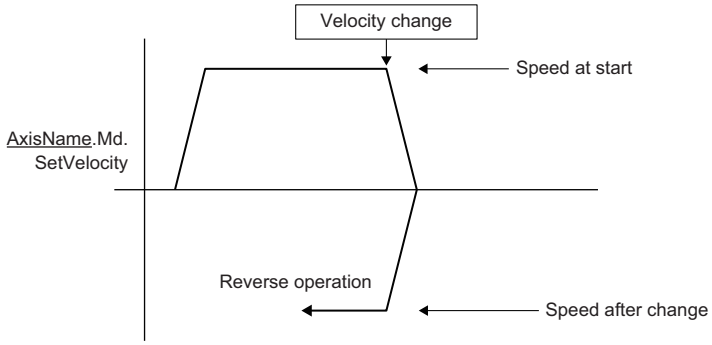
The set velocity is changed according to the acceleration/deceleration.



Set velocity sign inversion during speed control

The negative speed can be set in the speed control.

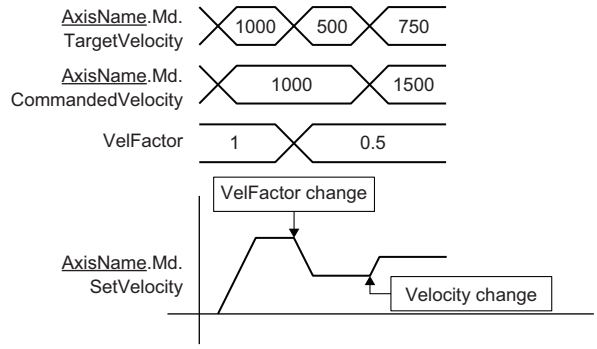
When the sign of set velocity is reversed, the reverse operation will start after a deceleration stop.



Combination with override

The velocity override factor also affects when the speed is changed.

The speed is changed to the speed obtained by adding the velocity override factor to the change speed.



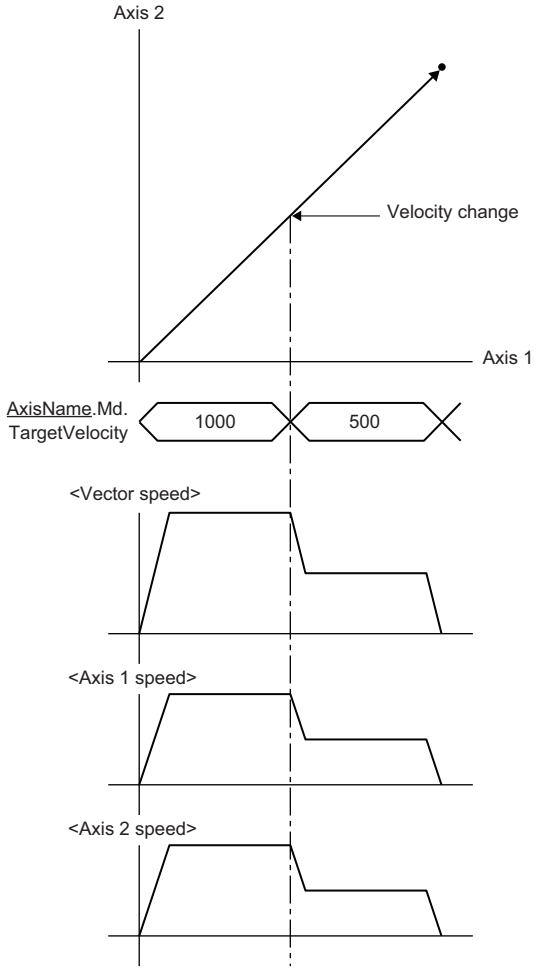
Operation during interpolation control

When the set velocity is changed during the interpolation control, the set velocity is changed according to the acceleration/deceleration method or the acceleration.

Each axis speed is changed according to the velocity mode.

Ex.

When the vector speed is set in the velocity mode



Acceleration/deceleration and acceleration/deceleration time change

This function changes the acceleration/deceleration and the acceleration/deceleration time.

An overrun may occur by changing the acceleration/deceleration and the acceleration/deceleration time.

When an overrun is occurring, the axis is controlled according to AxisName(AxesGroupName).Pr.OverrunOperation.

For operation when changing the acceleration/deceleration and the acceleration/deceleration time, refer to the following.

☞ Page 330 Acceleration/deceleration Processing Function

Precautions

- Maintain the previous value for the input values that are not changed.
- Even if input variables that cannot be changed are changed, any error or warning will not occur, and the change will be ignored and the control will be continued. For input variables that can be changed, refer to details on each FB.

12 COMMON FUNCTIONS

12.1 External Signal Selection

This section describes a setting method of I/O signals which are used in each control.
The following indicates the functions that use external signal selection.
☞ Page 192 Forced Stop
☞ Page 323 Hardware Stroke Limit

Relevant variables

I/O signals are expressed by the TARGET_REF structure. Another structure which has the TARGET_REF structure as a member may be defined depending on the function that uses I/O signals. For example, the MC_TRIGGER_REF structure has the SIGNAL_SELECT structure that indicates the bit I/O data as a member, and furthermore, the SIGNAL_SELECT structure includes the TARGET_REF structure.

Variable/Structure name		Name
MC_TRIGGER_REF		
	Signal (SIGNAL_SELECT structure)	Trigger signal
MC_INPUT_REF		
	Signal (SIGNAL_SELECT structure)	Input signal
MC_OUTPUT_REF		
	Signal (SIGNAL_SELECT structure)	Output signal

For the configuration of each structure, refer to the following.
☞ Page 701 List of Variables
The following shows specifications of structures common for each function, the SIGNAL_SELECT structure and the TARGET_REF structure.

SIGNAL_SELECT structure

This section describes SIGNAL_SELECT structure.

Variable/Structure name		Name
SIGNAL_SELECT		Signal select
	Source	Signal
	Detection	Signal detection method
	CompensationTime	Compensation time
	FilterTime	Filter time

The settings of the BOOL type I/O signals which are used in control are expressed by the SIGNAL_SELECT type structure variables. Details of the SIGNAL_SELECT type structure are shown below.

Signal name

- Specify BOOL type signals which are used in control with the TARGET_REF structure. Data types of [VAR], [DEV], [CONST] and [OBJ] can be used.
- For operation when a signal which cannot be used is set, refer to each function.

Signal Logic

- Specify the signal logic with the MC_SIGNAL_LOGIC enumeration type. For details, refer to the following.

☞ Page 696 ENUM enumerator

■Bit input signals

Setting	Operation
0: HighLevel (Detection at TRUE)	When the corresponding signal is in the TRUE status, it operates as a signal input status. Input signal
1: LowLevel (Detection at FALSE)	When the corresponding signal is in the FALSE status, it operates as a signal input status. Input signal
2: RisingEdge (Detection at FALSE→TRUE (rising edge))	When the corresponding signal turns from FALSE to TRUE, it operates as a signal input timing. Input signal
3: FallingEdge (Detection at TRUE→FALSE (falling edge))	When the corresponding signal turns from TRUE to FALSE, it operates as a signal input timing. Input signal
4: BothEdges (Detection at rising edge/falling edge)	When the corresponding signal turns from FALSE to TRUE and vice versa, it operates as a signal input timing. Input signal

■Bit output signals

Setting	Operation
0: HighLevel (Detection at TRUE)	When the output command is TRUE, the corresponding signal is turned to TRUE. Output command Output signal
1: LowLevel (Detection at FALSE)	When the output command is TRUE, the corresponding signal is turned to FALSE. Output command Output signal
2: RisingEdge (Detection at FALSE→TRUE (rising edge))	Must not be specified.
3: FallingEdge (Detection at TRUE→FALSE (falling edge))	Must not be specified.
4: BothEdges (Detection at rising edge/falling edge)	Must not be specified.

- Selectable logics vary depending on the function. For details, refer to each function.

Compensation time

- Specify the compensation time in seconds.
- Set a positive value to compensate the delay of the input signal, and set a negative value to compensate the lead.
- Set a positive value to delay the output timing of the output signal, and set a negative value to hasten it. When compensated output timing is earlier than the current time, output is immediately carried out.
- Whether the compensation time can be used or not is depending on the function. For details, refer to each function.

Filter time

- Specifies the filter time in seconds to remove chattering of the input signal. During the time that the signal status specifies, it is reflected to the input/output only when the same value is continued. The timing of input/output is delayed for the filter time.
- The availability of the filter time differs according to functions. For details, refer to each function.

TARGET_REF structure

This section describes the TARGET_REF structure.

Variable/Structure name		Name
TARGET_REF		Input signal
	StartIO	IO No.
	Target	Target

Specify a signal which is used in the control in Target of the TARGET_REF structure with the following character string format.
[Classification](Type)Data name.Bit position@Target modification
or
[Classification](Type)WSTRING type label^

Point

Up to 63 characters can be stored in Target. When specifying more than 63 characters string format, specify with the following method.
(1) Declare the WSTRING type global label in the motion system, and store the part of Data name.Bit position@Target modification.
(2) Specify the label name of (1) as the WSTRING type label of TARGET_REF. (To the end, add "^" that indicates reference.)

Classification and data name

Specify the data type in [Classification] and the target data in Data name. The specifying method of the data name varies depending on the classification.

Classification	Target	Specifying method of data name	Example
[OBJ]	CANopen object of the slave device	[OBJ]0xXXXXYYZZ XXXX: Specify the object index. YY: Specify the object sub index. ZZ: Specify the object size (Number of bit). (Ex: For 4 byte object, ZZ = 20H) Specifying the target device may be required in @Target modification. Check the specification of the function.	[OBJ]0x607A0020
[VAR]	Label of the motion system	[VAR]Label name Specify a label in the motion system includes in Label name. When specifying the local label, specify the POU*1 name in @Target modification. For whether the local label can be set or not and the conditions that the setting is enabled, check the specification of the function.	[VAR]ADunit10.OutputEnable
[AXIS]	Axis data	[AXIS]MC_SOURCE Specify a MC_SOURCE enumeration type enumerator in MC_SOURCE. Specifying the target data may be required in @Target modification. Check the specification of the function. Only the data type "LREAL" can be specified. This data cannot be written.	[AXIS]mcSetValue
[DEV]	Device in the motion system (Including buffer memory and link device of the module itself)	[DEV]Device name Specify a device in the module itself in Device name. For devices that can be specified, refer to "Motion Module Program Creation" in the following manual. MELSEC iQ-R Programming Manual (Motion Control Function Blocks) @Target modification cannot be specified. (It is ignored.)	[DEV]RWr10.5 [DEV]G10000.1
[CONST]	Constant	[CONST]Constant [CONST]0xConstant Specify Constant as floating decimal (E format is also available)/decimal integer/hexadecimal integer. The value can be changed for each cycle by specifying the @Target modification. This data cannot be written.	[CONST]1000 [CONST]0x100

*1 POU: Program Organization Unit

■MC_SOURCE

MC_SOURCE enumeration type enumerators are shown below. Check the specifications of the function to be used for the enumerators that can be specified.

○: Can be specified, ×: Cannot be specified

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

Type

It is written when explicitly specifying the data type. For whether the type specification can be used or not, handling when the type is omitted, and type conversion rules, refer to the specifications of the function.

The following types can be specified.

(BOOL)

(INT)

(DINT)

(WORD)

(DWORD)

(REAL)

(LREAL)

When the type of data by itself specified at [Classification] specifies (BOOL) to the data other than BOOL type, it will be handled as bit position .0 is omitted if it is not specifically specified in each function description.

Ex.

[DEV](BOOL)G10000, will be handled as [DEV](BOOL)G10000.0.

Bit position

When .Bit position is added to the data whose [Classification] is [OBJ], [VAR] (only WORD type, DWARD type, INT type, DINT type) and [DEV], the data is dealt with as BOOL type data. From 0 to F can be used for the bit position. When (Type) and .Bit position are simultaneously specified, (Type) is ignored and the data becomes BOOL type data.

Target modification

Specify auxiliary information for specifying data. The details differ depending on [Classification].

Be sure to specify it depending on the function that uses signals. When specifying the target modification for data types and function that do not require it, the specification will be ignored.

Classification	Target modification details		Setting example
[OBJ]	Specify the station address (address information to distinguish slave devices.)		
	IP address specifying	@XXX.XXX.XXX.XXX Write in decimal format.	@ 192.168.3.10
	Network No./station No. specifying	@Network No. - Station No. Write in decimal format.	@ 192-10
	Axis label specifying	@Axis label Write axis label.	@Roller
	Multidrop No.	#Multidrop No. Like multi-axis drive unit, when one station includes multiple logic axes, specify the No. in order to distinguish logic axes. Write in decimal format. Multidrop No. can be omitted. When it is omitted, the device is regarded as a single axis device (for a multiple axes device, it is regarded as specifying 0 in multidrop No.).	@ 192.168.3.10#2
[VAR]	@POU name*1 Use it when specifying the local label. 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.		
[AXIS]	@Position @CumulativePos Specify the target data (position).		@Position @CumulativePos
[CONST]	@+adding value or @-subtracting value Each time referring the value, it is used to change the read value. When the data type is REAL or LREAL, this setting is invalid.		*2

*1 POU: Program Organization Unit

*2 The referred value set as [CONST] (INT) 0 @ -10 will be as 0, -10, -20, -30, ….
The referred value set as [CONST] (BOOL) 0 x 00.3 @ + 1 switch TRUE/FALSE every eight (2^3) cycle. (1 byte value +1 will be added in every cycle and retrieves bit 3 status)
Specifying [CONST] bit can be used from 0 to 7.

Common rule

- Capital letters and lower case letters are not distinguished.
- The space is ignored.

12.2 Touch Probe

This function records (latches) arbitrary data at trigger input signal detection.

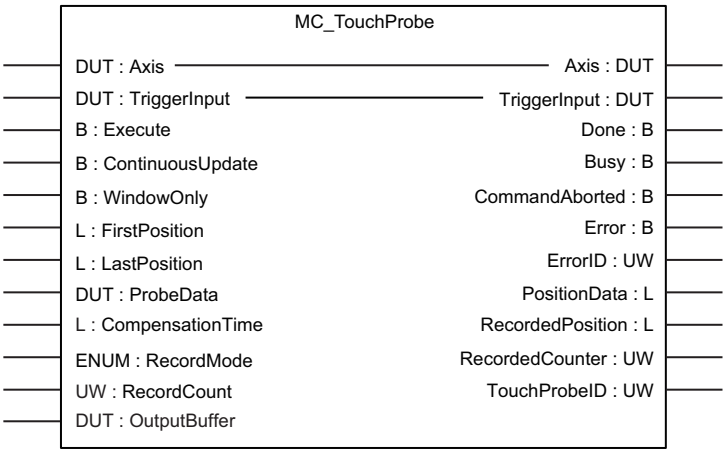
Operation of this function for each system status

○: Possible, ×: Not possible

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

Relevant FBs

MC_TouchProbe

Item	Description
Function overview	Optional data will be recorded due to the trigger event occurence.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

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

■I/O variables

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Axis information	Axis	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables
Trigger input signal	TriggerInput	MC_TRIGGER_REF	↑	—	Omission not possible	☞ Page 418 External Signal Selection

■Input variables

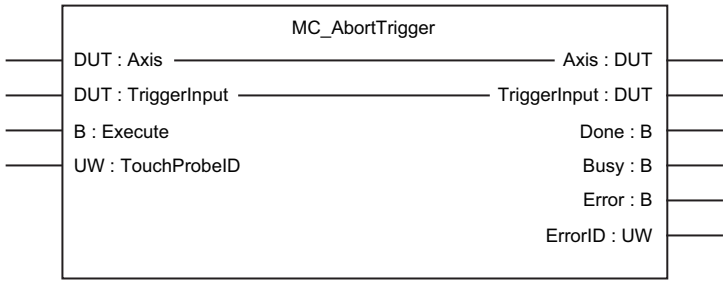
Import ↑: At start, R: Retrigger possible, C: Continuous update possible

Name	Variable name	Data type	Import	Setting range	Default value	Description
Execute command	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Continuous update	ContinuousUpdate	BOOL	↑	TRUE, FALSE	FALSE	During TRUE, it can continuously change the enabled window, the lower limit position, and the upper limit position.
Enabled window	WindowOnly	BOOL	↑/R/C	TRUE, FALSE	FALSE	When it is TRUE, triggers are detected only in the areas defined by lower limit position and upper limit position.
Lower limit position	FirstPosition	LREAL	↑/R/C	—	0.0	Specifies the lower limit position of the trigger event valid area. The valid area includes the lower limit position.
Upper limit position	LastPosition	LREAL	↑/R/C	—	0.0	Specifies the upper limit position of the trigger event valid area. The valid area includes the upper limit position.
Probe data	ProbeData	TARGET_REF	↑	—	—	Sets the probe data. ☞ Page 420 TARGET_REF structure
Compensation time	CompensationTime	LREAL	↑	-5.0 to 5.0 [s]	0.0	Compensates the delay time of the touch probe process. Set a positive value to compensate for a delay.
Latch mode	RecordMode	MC_RECORD_MODE	↑	0 to 2	0	Sets the latch method of the data. 0: OneShot: One shot mode 1: RecordCount: Number of designation times mode 2: RingBuffer: Ring buffer mode
Number of latch times	RecordCount	WORD(UINT)	↑	1 to 65535	1	Sets the number of latch times.
Storage location of latch data	OutputBuffer	TARGET_REF	↑	—	—	Sets the storage location of latch data. When it is omitted, latch data is output only in RecordedPosition.

■Output variables

Name	Variable name	Data type	Default value	Description
Completion	Done	BOOL	FALSE	When TRUE, it indicates the output value is valid.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Abortion of execution	CommandAborted	BOOL	FALSE	Indicates that an execution of FB is aborted by error and multiple start.
Error	Error	BOOL	FALSE	Indicates that an error has occurred in the FB.
Error code	ErrorID	WORD(UINT)	0	The error code generated is returned. For details, refer to the following. ☞ Page 662 List of Error Codes
The current value monitor data	PositionData	LREAL	0.0	Stores the current value of data to be latched.
Latch position	RecordedPosition	LREAL	0.0	Stores the current value when a trigger event is occurring. When latch is executed multiple times, the data is overwritten by the latest data.
Number of latch times	RecordedCounter	WORD(UINT)	0	Stores the number of times that latch is executed.
Probe ID	TouchProbeID	WORD(UINT)	0	Stores the ID unique to this FB.

MC_AbortTrigger

Item	Description
Function overview	Disables the latch in execution.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

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

■I/O variables

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Axis information	Axis	AXIS_REF	↑	—	Omission possible	☞ Page 702 Axis variables
Trigger input signal	TriggerInput	MC_TRIGGER_REF	↑	—	Omission possible	☞ Page 418 External Signal Selection

■Input variables

Import ↑: At start

Name	Variable name	Data type	Import	Setting range	Default value	Description
Start	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Touch probe ID	TouchProbeID	WORD(UINT)	↑	1 to 65535	0	Sets a proper ID of touch probe to invalidate.

■Output variables

Name	Variable name	Data type	Default value	Description
Execution completion	Done	BOOL	FALSE	When TRUE, it indicates the output value is valid.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Error	Error	BOOL	FALSE	When ON, it indicates that an error has occurred in the FB.
Error code	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned. For details, refer to the following. ☞ Page 662 List of Error Codes

Touch probe valid

This is a function that arbitrary data is recorded (latch) at the timing of detecting the trigger input signal.
It is possible to latch only the data within the range by setting the data range to latch.
When this function is used, execute MC_TouchProbe. To stop this function, execute MC_AbortTrigger.
When a trigger event occurs, the operation becomes as shown below.

Operation

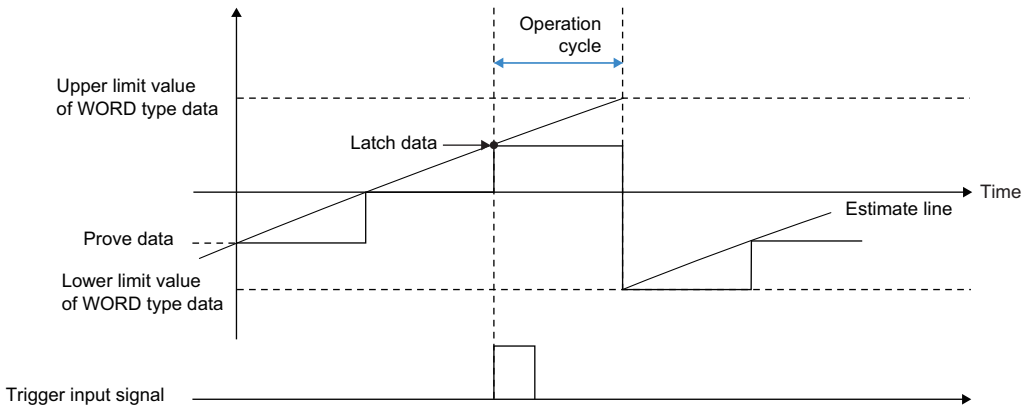
- Calculations for the latched data are estimated at rising edge/falling edge/both direction of the signal set in TriggerInput (indicate trigger input signal as follows).
- When it is WindowOnly = TRUE, whether the latch data at trigger event occurrence is within the range or not is confirmed. Data outside the range are not latched.
- The latch data is stored to the storage location specified RecordedPosition and OutputBuffer according to RecordMode, and RecordedCounter is updated.

■Estimated calculation

The prove data during operation cycle interval is calculated by estimation. The value calculated by estimation at the timing inputted trigger input signal is the latch data. The value is calculated as the figure below.
[When the prove data is the WORD type]

- When using the remote I/O which is not supported with high accuracy

The trigger input timing is operation cycle.



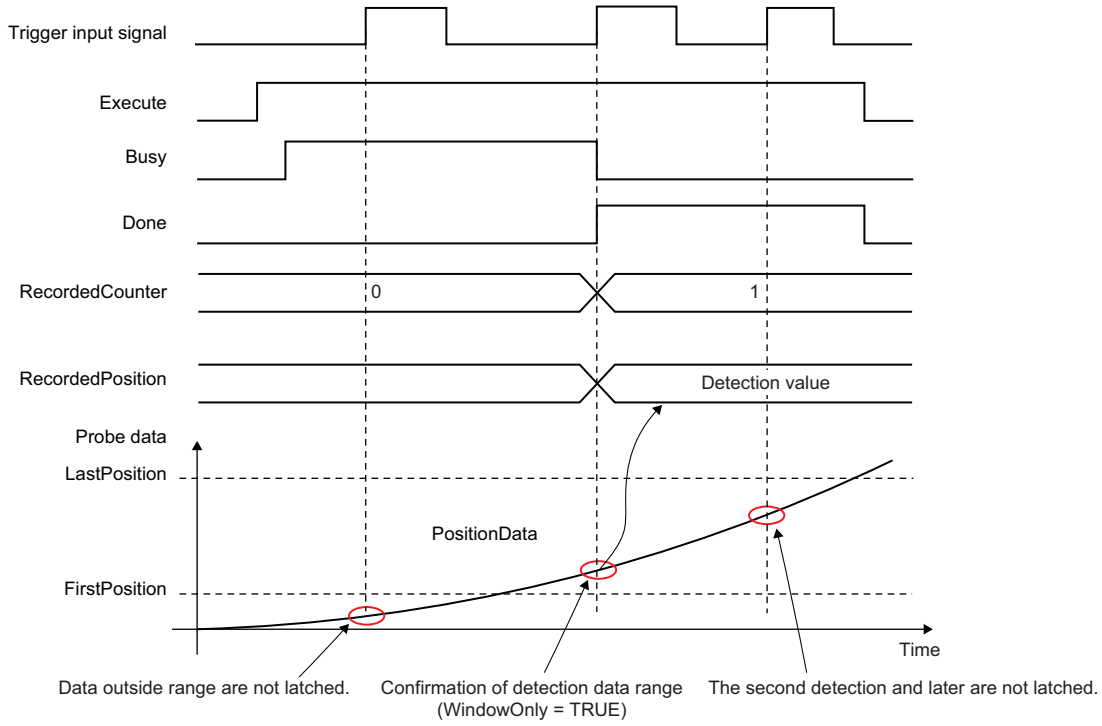
■Timing chart

[When the FB is normally completed]

The following shows examples of latch operation by RecordMode.

Ex.

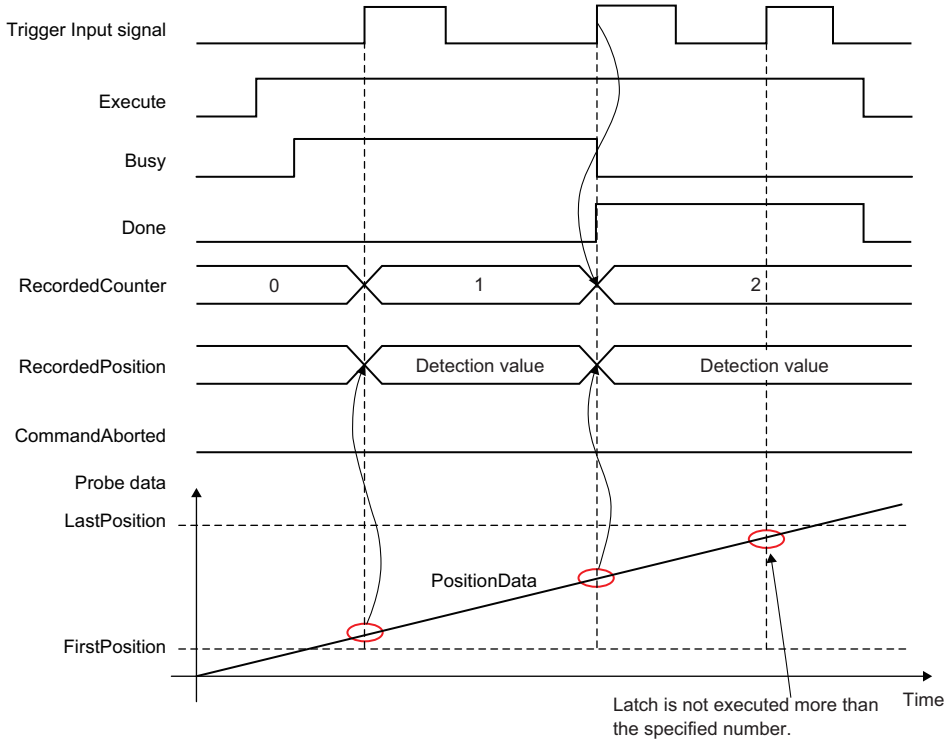
For RecordMode = "0: OneShot"



Ex.

For RecordMode = "1: RecordCount"

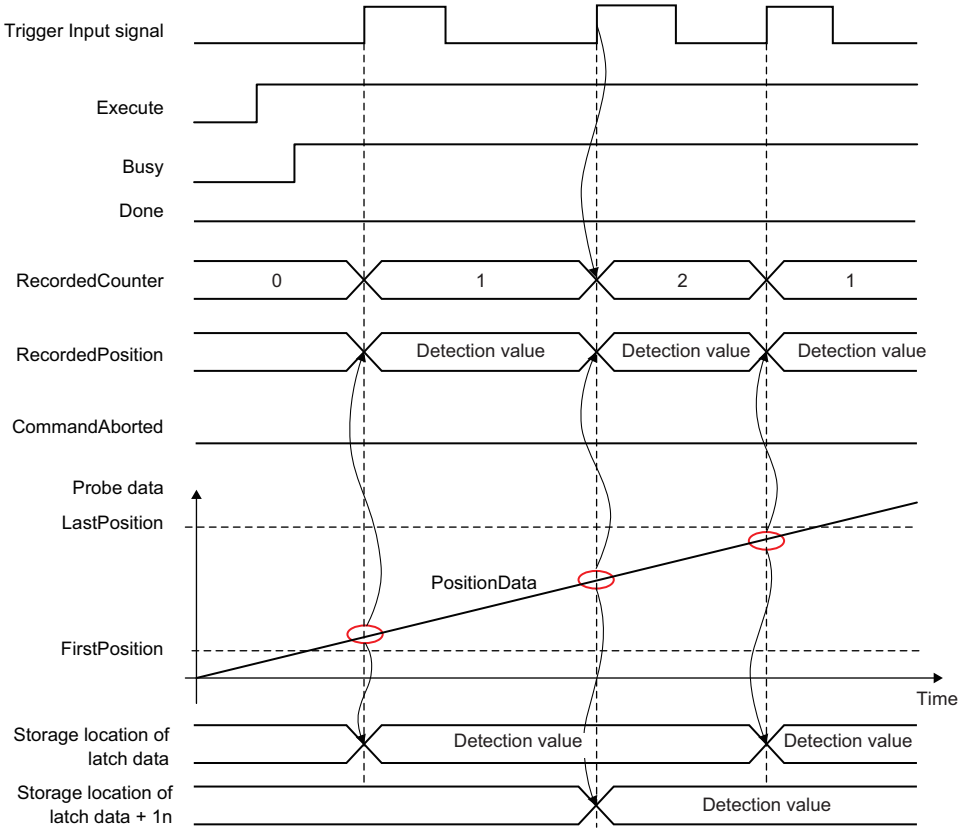
Number of latches: 2



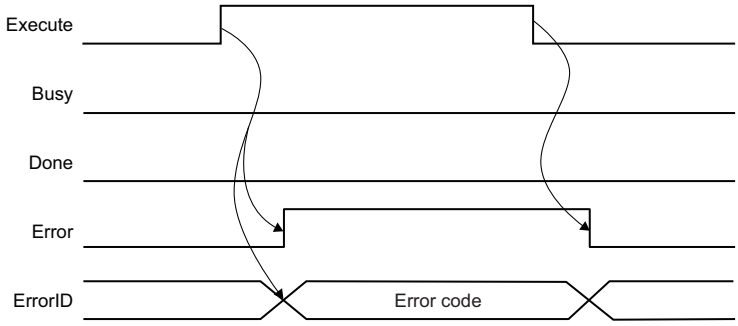
Ex.

For RecordMode = "2: RingBuffer"

Number of latches: 2



[When an error occurs]



Details of setting items

■TriggerInput (trigger input signal)

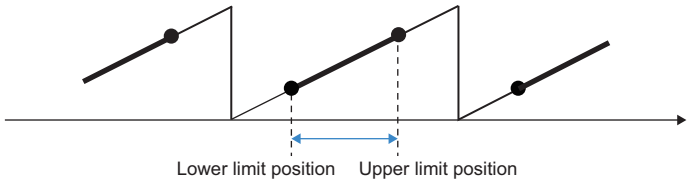
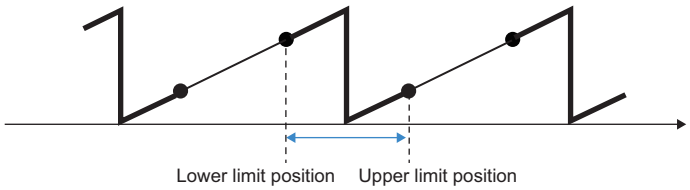
Trigger input signal is specified in MC_TRRIGER_REF structure.

Setting range of TriggerInput.Signal is shown below.

Item		Setting range
TriggerInput.Signal		
	Source.Target	Type which can be specified • BOOL Data type which can be used • [VAR] • [DEV] • [CONST] • [OBJ]
	Detection	2: RisingEdge (Detection at FALSE→TRUE (rising edge)) 3: FallingEdge (Detection at TRUE→FALSE (falling edge)) 4: BothEdges (Detection at rising edge/falling edge)
	CompensationTime	-5.0 to 5.0 [s]
	FilterTime	0 to 5.0 [s]

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

When WindowOnly is valid, triggers are detected only within the range between FirstPosition and LastPosition of the probe data. When it is invalid, the touch probe processing is executed to the all range.

LastPosition, FirstPosition relation	Touch probe processing
LastPosition > FirstPosition	The touch probe processing is executed when the probe data is more than FirstPosition and less than LastPosition. [Operation example] 
LastPosition < FirstPosition	The touch probe processing is executed when the probe data is less than LastPosition or more than FirstPosition. [Operation example] 

■ProbeData (Probe data)

Specify the data to latch by the TARGET_REF structure. The data types of [VAR], [AXIS], [DEV] and [OBJ] can be used.
When specifying [AXIS], the axis data of Axis input is referred. The Target modification of [AXIS] (@Position, etc.) needs to be specified.
When specifying [VAR] or [DEV], Axis is ignored. Specify "(LREAL)" when specifying the data other than LREAL type to TARGET_REF.Target

Ex.
Setting example of TARGET_REF.Target
[DEV](LREAL)G11500000
[VAR]OutputBufferData1(OutputBufferData1 is LREAL type instance)
[VAR](LREAL)OutputBufferData2(OutputBufferData2 is WORD type instance)
The data is updated as the ring counter. The upper limit value or the lower limit value of the ring counter differs depending on the data classification or the data type.
When the data classification is [DEV] or [VAR], the upper limit value or the lower limit value of the ring counter is the maximun value/the minimum value of the data type.

Ex.
The data classification is [AXIS].

Modification	Ring counter upper limit value/lower limit value
@Position	The ring counter upper limit value/lower limit value of the axis
@CumulativePos	The positioning range upper limit value/lower limit value of the axis

The data classification is [VAR].

Label	Ring counter upper limit value/lower limit value
WORD type label	The maximum value/minimum value of the data type Ring counter upper limit value: 65535 Ring counter lower limit value: 0

Restriction

The change amount per 1 operation cycle should meet the following formula. If it does not meet the formula, the actual change amount of the probe data and the latched change amount may not match.
The change amount per 1 operation cycle < |(The ring counter maximun value) - (The ring counter minimum value) + 1| / 2

Precautions

If the operation cycle over occurs after MC_TouchProbe.Busy = TRUE, accurancy of estimated calcuration may deteriorate.

■CompensationTime (Compensation time)

Delay time of a touch probe processing is compensated.Set this item to compensate a proper delay of the touch prove process separately form the compensation time for the trigger input signal. Set the positive value to compensate the delay, and negative value to compensate the process.
Total compensation time is CompensationTime - TriggerInput.Signal.CompensationTime.

Point

When compesating the delay time, the past value is used for latch data estimate calculation. Thus, at the time the compensation time from the Busy = TRUE timing is elapsed, the accuracy will be at the maximum.
The compensation time and the operation cycle consume the memory amount as follows.
Memory consumption = 24 bytes × (total compensation time ÷ operation cycle)

■RecordMode (Latch mode)

Latch method of data is specified by RecordMode.

[RecordMode = "0: OneShot"]

- Latch data is stored only one shot.

[RecordMode = "1: RecordCount"]

- Latch data is stored for the setting value of RecordCount.
- Latch data is stored in the area shifted for the latch data size per latch, to the storage location set in OutputBuffer in front.
- To stop touch probe function before completing latch for RecordCount, execute MC_AbortTrigger.

[RecordMode = "2: RingBuffer"]

- Latch data is stored to the ring buffer for the setting value of RecordCount.The latch operation is always carried out.
- Latch data is stored to the ring buffer for RecordCount after shifted for latch data size per latch, to the storage location set in OutputBuffer in front.
- To stop touch probe function before completing latch for RecordCount, execute MC_AbortTrigger.

Following value is stored for RecordedCounter and RecordedPosition by RecordMode.

RecordMode	RecordedCounter	RecordedPosition
0: OneShot	1 by latch operation.	Stores latch data only one shot.
1: RecordCount	+ 1 by latch operation.	Stores latest latch data by latch operation.
2: RingBuffer	+ 1 by latch operation. Returned 0 when more than RecordCount.	Stores latest latch data by latch operation.

■RecordCount (Number of latch times)

Following value is specified for RecordCount by RecordMode.

RecordMode	RecordCount
0: OneShot	Ignores.
1: RecordCount	Specifies number of latch times of data. When omitting, an error will occur.
2: RingBuffer	Specifies number of ring buffer times of data. When omitting, an error will occur.

■OutputBuffer (Storage location of latch data)

The storage location of latched data can be specified in TARGET_REF structure. Applicable data types are [VAR] and [DEV].

Specify "(LREAL)" when specifying the data other than LREAL type to TARGET_REF.Target LREAL.

RecordMode	storage location
0: OneShot	The area specified by OutputBuffer
1: RecordCount	Following area Area specified by OutputBuffer+ RecordedCounter × Latch data size
2: RingBuffer	

OutputBuffer can be ommitted. When ommitting, the latch data is stored only in RecordedPosition.

A setting example of TARGET_REF.Target is shown below.

Ex.

When RecordMode = OneShot or RecordMode = RecordCount, RingBuffer's RecordCount = 1

[DEV](LREAL)G11500000

[VAR]OutputBufferData1 (OutputBufferData1 is LREAL type instance)

When RecordMode = RecordCount, RingBuffer's RecordCount = 100

[DEV](LREAL[1..100])G11500000

[VAR]OutputBufferData2 (OutputBufferData2 is LREAL[1..400] type instance)

[VAR](LREAL)OutputBufferData3 (OutputBufferData3 is WORD[1..100] type instance)

When the number of elements specified with OutputBuffer is less than the one specified with RecordCount, the error "Latch Data Storage Shortage" (error code: 349FH) occurs. Also, when the out of range of the device and the label is accessed at latch operation, the error "Latch Data Storage Shortage" (error code: 349FH) occurs.

The relation of the setting value and operation of each item is shown below. For details, refer to the following.

☞ Page 426 Operation

RecordMode	RecordCount	OutputBuffer	Detection operation	Data storage location
0: OneShot	Omit/Set	Omit	One shot	RecordedPosition
		Set		RecordedPosition/Buffer
1: RecordCount	Omit	Omit/Set	Error	—
	Set	Omit	Number of setting	RecordedPosition
		Set		RecordedPosition/Buffer
2: RingBuffer	Omit	Omit/Set	Error	—
	Set	Omit	Always	RecordedPosition
		Set		RecordedPosition/Buffer

Precaution

- Refresh of the MCFB I/O argument is carried out at the timing the FB is called, however, the control is executed by using operation cycle regardless of the FB call task (nomal/fixed cycle).
- This FB is executed even if the target axis is not connected or an error occurs.

Touch probe invalid

Touch probe is invalidated.

Control details

MC_TouchProbe which is specified in TouchProbeID is invalidated..

When MC_AbortTrigger is executed, if the specified MC_TouchProbe specified TouchProbeID is not running, the FB will immediately become Done = TRUE.

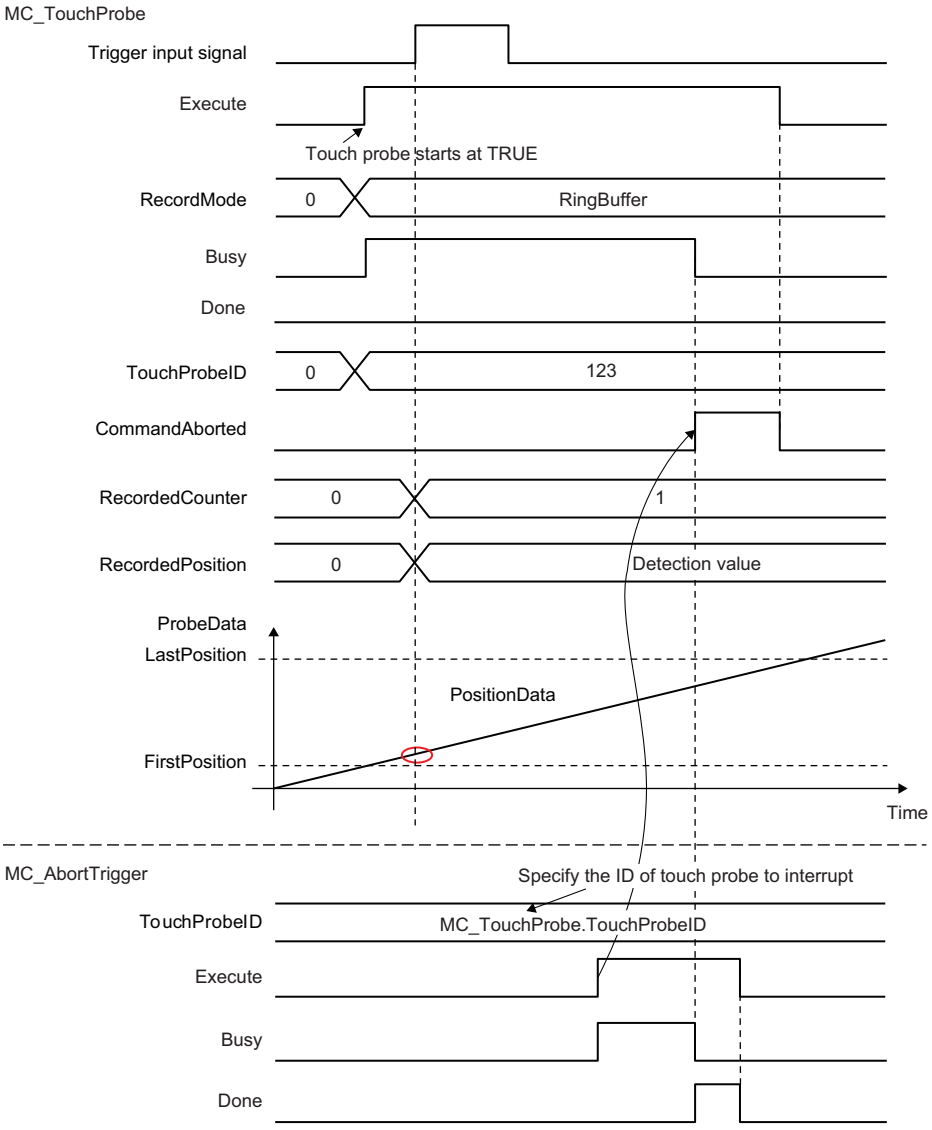
Axis and TriggerInput can be omitted, because those are ignored.

■Timing chart

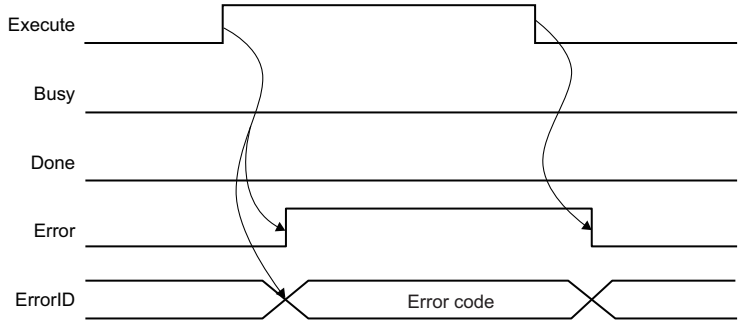
[When the FB is normally completed]

Ex.

For RecordMode = "2: RingBuffer"



[When an error occurs]



Details of setting item

■TouchProbeID (Touch probe ID)

- Sets the proper ID of the touch probe to invalidate. Sets MC_TouchProbe.TouchProbeID after MC_TouchProbe.Execute rising of MC_TouchProbe to invalidate.

12.3 Slave Emulation

Slave emulate function controls axes on the real axis without connecting to the slave station.
When using emulate function, even if the slave station is not connected, the axis can be operated virtually (regarded as it is connected). This function enables to debug the user program at the start-up of the device or verify the positioning operation.
Emulate function can be used on real axes (regardless of the axis type).
Emulate function simulates the cyclic communication and the transient communication of the slave device. The operations unique to the slave device can be simulated by installing emulate models corresponding to each slave device as add-ons.
Emulate function can be used according to the following table.
○: Operation possible, ×: Operation impossible

Module operation mode setting	Emulate function Switches valid/invalid	Start up as an emulate axis
Online mode*1	○	○

*1 For details, refer to "Application Settings" in the following manual.
MELSEC iQ-R Motion Module User's Manual (Network)

Relevant variables

Variable/Structure name	Name	Details
AxisName.PrConst.		
SlaveEmulate_Enable	Slave Emulation Enabled	Sets whether the axis is used as an emulate axis. FALSE: Invalid TRUE: Valid
AxisName.Md.		
SlaveEmulate_Enable	Slave Emulating	Becomes TRUE when driver emulate operation is in execution. FALSE: Invalid TRUE: Valid

Setting procedures

This section explains the procedures to carry out the emulate setting of the real drive axis.

1. Set the real drive axis in the engineering tool, and start up the motion system.
2. Set the real drive axis to use it as the emulate axis in the engineering tool.

Operation during emulate function

The emulate function operation follows the axis parameter setting. The values of the set position and the feed machine position at connection are as shown below.

• When "AxisName.PrConst.PosRestoration_AbsPosEnable" is "-1: Auto" or "0: ABSDisabled"

Both the cumulative current position and the feed machine position are "0". (The set position is according to the ring counter setting)

• When "AxisName.PrConst.PosRestoration_AbsPosEnable" is "1: Enabled"

When the home position has been established, those are the address when the module power was disconnected most recently.
When the home position has not been established, both the cumulative current position and the feed machine position are "0". (The set position is according to the ring counter setting)

Specification of the emulated slave station

While the emulate function is valid, following type of the servo amplifier and servo motor are tentatively connected.

Servo amplifier type	Servo motor type
MR-J5-10G	HK-KT13W

Specification of the emulated MR-J5(W)-G is as follows.

Function		Support	Description
Reception check	WDC check	×	Check of reception WDC (the master station → the slave station) is not carried out.
PDO-related	Emulation of feedback	○	Simulates feedback of MR-J5(W)-G. For details, refer to the following. ☞ Page 436 Feedback data specification
	Variable mapping	×	When the emulate function is valid, the default mapping of MR-J5(W)-G is used. For the default mapping, refer to MR-J5(W)-G manuals.
SLPM-related	Response data simulation	×	SLMP communication simulation is not supported.
Servo alarm-related	Servo alarm [AL.035 Command frequency error] detection	○	Simulates servo alarm [AL.035 Command frequency error] detection
	Alarm reset	○	Simulates alarm reset. (A delay to deactivate the alarm does not occur. The alarm is deactivated immediately.)
Motor type	Standard rotary type	○	Standard motor of MR-J5 (HK-KT13W) is available. A speed unit of the servo amplifier is fixed to 0.01 r/min.
Operation mode	csp	○	Position control operation by csp is available.
	csv	○	Velocity control operation by csv is available.
	cst	○	Torque control operation by cst is available.
	hm	○	Homing by hm is available. The homing method supports the driver homing method (data set method) only.
	ct	○	Continuous operation to torque control by ct is available. (The operation is the same as cst.)
External signal-related	FLS	○	• Input via the controller is only available. (Stop operation of the slave station when the FLS is OFF is not simulated. By FLS detection, the command from the master station stops and then the servo motor stops as well. • In the case of input via the servo amplifier, the input value is always OFF.
	RLS	○	• Input via the controller is only available. (Stop operation of the slave station when the RLS is OFF is not simulated. By RLS detection, the command from the master station stops and then the servo motor stops as well. • In the case of input via the servo amplifier, the input value is always OFF.

Feedback data specification

No.	Object name	Object detail	Support	Remark
1	1D02h/01h: Watch dog counter UL	—	○	—
2	6061h/00h: Modes of operation display	—	○	—
3	6064h/00h: Position actual value	—	○	The value is always the same as the value of Target position.
4	606Ch/00h: Velocity actual value	—	○	—
5	60F4h/00h: Following error actual value	—	○	The value is always 0.
6	6041h/00h: Statusword	—	○	—
7	6077h/00h: Torque actual value	—	○	• The value is always 0 when set to a mode other than cst or ct. • The value is always the same as the value of Target position.
8	2A41h/00h: Current alarm	—	○	—

Precaution

- When executing the all axes emulate switching, it takes time to complete.
- If a motor is rotated when the slave station which is connected actually is set to emulation enabled, as it is switched to disabled, the position information is restored as the axis is recognized as it moves from the position which the axis moves during emulate enabled to the position of the connected slave station during power-OFF.
- In the emulation being disabled, when the slave station is used with the setting which "AxisName.PrConst.PosRestoration_AbsPosEnable" is "-1: Auto" and the connected device uses absolute position system, chenge "AxisName.PrConst.PosRestroration_AbsPosEnable" to "1: Enabled" before switching emulation to enabled. When the emulation is enabled, "AxisName.PrConst.PosRestroration_AbsPosEnable" turns from "-1: Auto" to "0: ABSDisabled", therefore the absolute position is erased next time, the emulation is switched to disabled.
- In the state which the add-on ServoDriver_CANopen is enabled, the axis always starts with the emulation enabled state if the add-on NetworkDriver_CCIETSN is disabled.

13 SYNCHRONOUS CONTROL

13.1 Overview of Single Axis Synchronization Control FBs

Single axis synchronization control FBs enable to control mechanical systems such as gear, speed change gear, and cam with a software by transmitting the position information (command) of Slave that is synchronized with Master.

FB	Control details
MC_CamIn	Executes cam operation.
MC_GearIn	Sets the speed ratio between the master axis and the slave axis, and enters gear operation.
MC_CombineAxes	Combines operations of two axes by a selectable combination method, and outputs it to the third axis.
MCv_ChangeCycle MCv_ChangeCycleTriggered	Changes the cam current value per cycle to the specified value during MC_CamIn control. It is used to interpolate the cam current value per cycle into an arbitrary value.
MCv_******Filter	Carries out the specific filter processing to input of Master, and outputs the result to Slave. ☞ Page 384 Command Filter

13.2 Axis Configuration

The following shows axes that can be specified to Master and Slave in the single axis synchronization control FBs. For the specifications of each axis type, refer to the following.

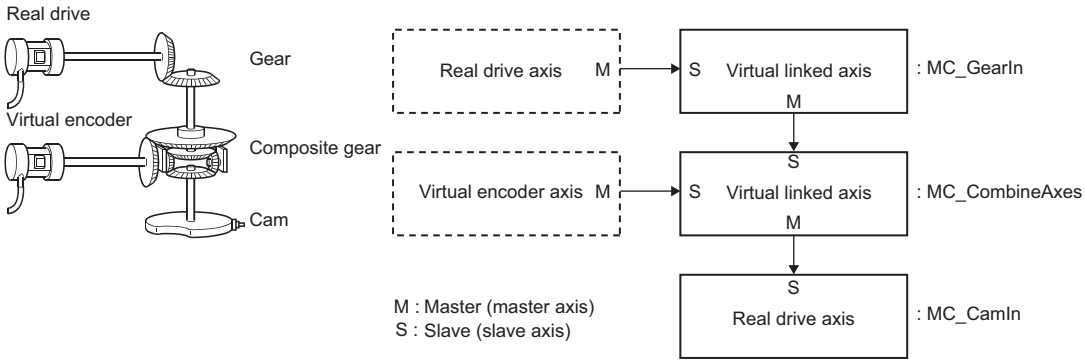
☞ Page 36 Axis, ☞ Page 56 Axis Assignment

Axis type		Used for Master	Used for Slave	Remark
Real axis	Real drive axis	○	○	When control for Motion FB is required, create a configuration that executes function blocks for each virtual linked axis and transmits the result (command) to the real drive axis.
	Real encoder axis	○	×	It is used as Master. When it is used as Slave, the error "Necessary Slave Object Unset" (error code: 1AA8H) occurs and it does not start.
Virtual axis	Virtual drive axis	○	○	Cammands can be mainly created by positioning control, etc.
	Virtual encoder axis	○	×	It is used as Master. When it is used as Slave, the error "Necessary Slave Object Unset" (error code: 1AA8H) occurs and it does not start.
	Virtual linked axis	○	○	It is used as an intermediate axis to transmit a command to the real drive axis. When using multiple motion FBs such as gear, assign virtual linked axes.

A command can be transmitted between axes by linking the axes to the single axis synchronization control FB. Also, to simultaneously execute controls such as gear and cam by executing multiple function blocks, like the following example, link the necessary number of axes and execute the single axis synchronization control FB.

Ex.

When cam control is realized by linking axes with gear from the real drive and adding a command from the virtual encoder to the command



Regardless of the setting of the position command unit of the master axis and the slave axis, synchronous control is execurted based on the value of data set for master axis data source selection.

Ex.

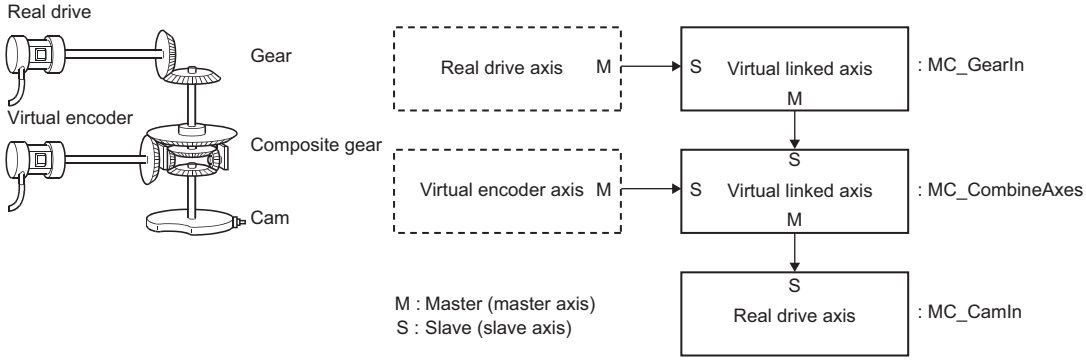
When using the gear (Gear ratio, 1 : 1) under the following setting and moving the master axis for 1 pulse, the slave axis moves 1 degree.

- Position command unit of the master axis: pulse
- Position command unit of the slave axis: degree

13.3 Master Axis Data Source Selection

In the single axis synchronization control FB, a position type of the master axis whose the slave axis executes the single axis synchronous control can be selected by MasterValueSource of each FB.
When the actual position is specified for virtual axis (axis which does not have AxisName.Md.ActualPosition (actual position)), the FB operates by the value as the same with set value.
"101: mcLatestSetValue" and "102: mcLatestActualValue" uses a position of the master axis in the same operation cycle, and the others use a position of the master axis in the previous operation cycle. Using "101: mcLatestSetValue" and "102: mcLatestActualValue" can transmit the multiple MCFB command in the same operation cycle. In this case, set the first execution order and linking order of the linked MCFB to be the same.

Ex.
When transmitting movement amount of the real drive axis and the virtual drive axis to cam to control in the same operation cycle.



When using the above combination of the single synchronous control FB (linked order), set the first execution order as follows.

- First execution order

1. MCFB which moves the master axis (real drive axis) of MC_GearIn
2. MC_GearIn
3. MC_CombineAxes
4. MC_CamIn

- Master axis data source selection of each single synchronous control FB
  "101: mcLatestSetValue"

Point

When "1: mcSetValue" and "101: mcLatestSetValue" are set to MasterValueSource and the master axis becomes servo OFF by the servo alarm, forced stop or disconnection, the amount of value change may be large. This can be prevented by setting "2: mcActualValue" and "102: mcLatestActualValue" to MasterValueSource.

MasterValueSource	Description
1: mcSetValue (Set value)	Uses the commanded position of the master axis in the previous operation cycle.
2: mcActualValue (Actual value)	Uses the actual position of the master axis in the previous operation cycle.
101: mcLatestSetValue (Latest set value)	Uses the commanded position of the master axis in the current operation cycle.
102: mcLatestActualValue (Latest actual value)	Uses the actual position of the master axis in the current operation cycle.

Current position change

When setting "1: mcSetValue" or "101: mcLatestSetValue" to MasterValueSource, even if the current position is changed of master axis, the slave axis does not move because the feed machine position is not changed.

13.4 Start and Stop Operation

Slave continues to control with the command transmitted by the master axis even if a stop cause occurs during execution of the synchronous control function blocks. The axis status of the slave axis does not change.

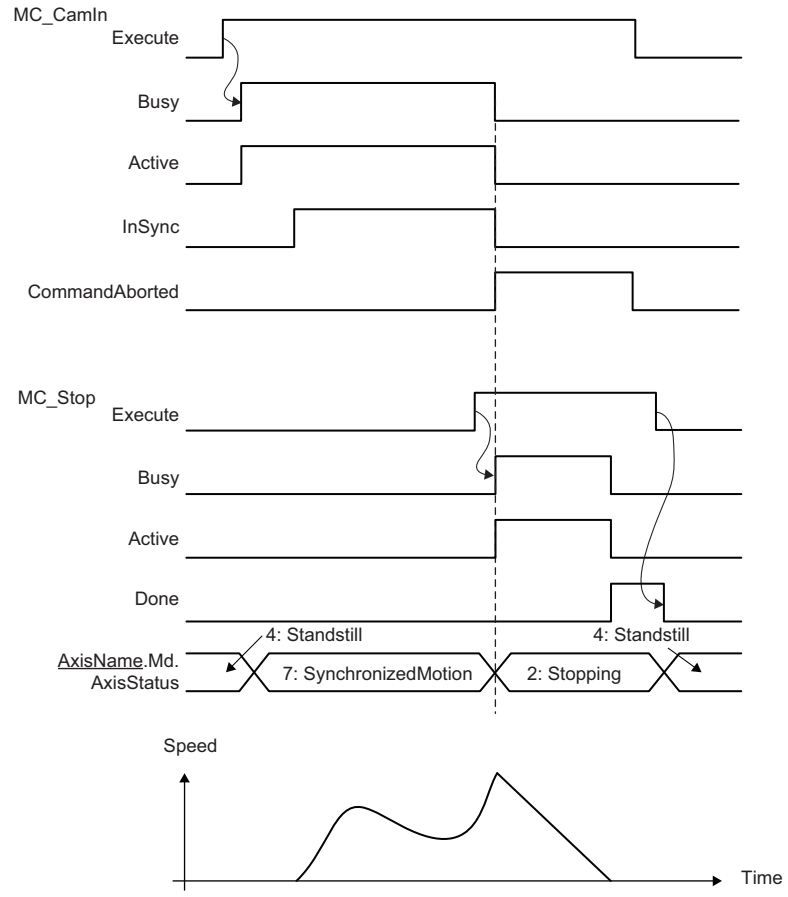
When a stop cause occurs to the slave axis, the axis status transits and the slave axis immediately stops or stops with deceleration according to the stop cause and its stop process. For axis control when a stop cause occurs, refer to the following.

☞ Page 170 Stop causes during axis operation

The master axis is not affected even when the stop cause occurs to the slave axis.

Ex.

When a stop cause (MC_Stop) occurs during execution of the synchronous control FB



Point

When the master axis becomes servo OFF by the servo alarm or forced stop, stopping the operation of the synchronous control FB by using MC_Stop (Forced stop) is recommended because the slave axis may cause unexpected operation.

13.5 Required Slave Object

The following shows the setting condition of the slave object when using it as Slave of the synchronous control FB.

Slave object necessary/not necessary		Starting availability
Target position	**Position actual value**	
Necessary	Not required	Starting is possible
Not necessary	Necessary	Starting is not possible*1
Not necessary	Not necessary	Starting is possible

*1 The error "Necessary Slave Object Unset" (error code: 1AA8H) occurs and it will not start.

13.6 Restrictions

- When Master and Slave are specified to the same axis, the error "Master Axis and Slave Axis No. Duplicated" (error code: 1A3EH) will occur.
- When connecting multiple axes in the synchronous control FB, if the master axis is specified to Slave of the latter function block or more than 257 synchronous control FBs are linked, the error "Master Axis and Slave Axis Cyclic Reference" (error code: 1A3FH) will occur.

13.7 Cam Operation

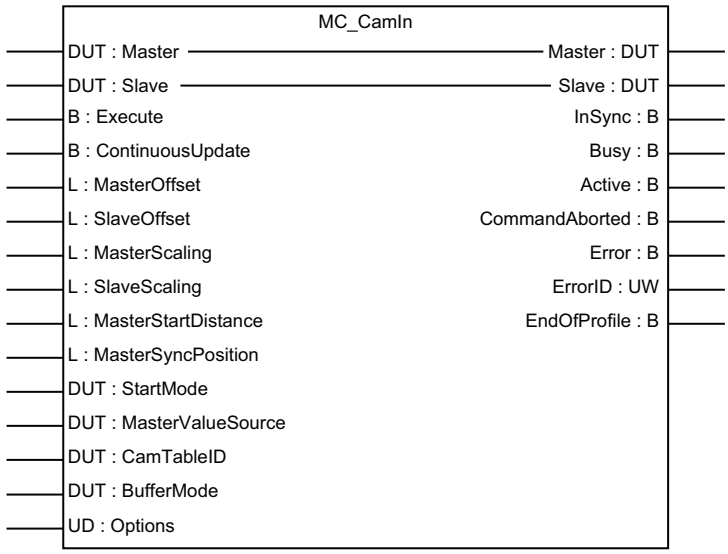
This function operates by synchronizing the slave axis with the master axis based on cam table.
Before using this function, it is necessary to open the cam table to the cam open area and set to a state where it can be controlled. For the open method to the cam open area, and for cam data types that can be controlled in cam operation, refer to the following.

☞ Page 502 Operating operation profile data
☞ Page 489 Operation profile data types

Relevant variable/FB

Variable/Structure name	Name	Details
AxisName.Md.		
ProfileID	Execution Profile ID No.	Stores the executing CamTableID. The update timing is as follows. [Outputs the specified CamTableID] • When executing MC_CamIn • When changing CamTableID The timing differs depending on the setting value of StartMode. [Cleared to "0"] • When EndOfProfile turns TRUE with the setting which Periodic is "FALSE: Non periodic" and Options bit16 is "0: Non periodic" • When stop cause occurs

MC_CamIn

Item	Description
Function overview	Cam operation starts operation according to the specified cam data.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

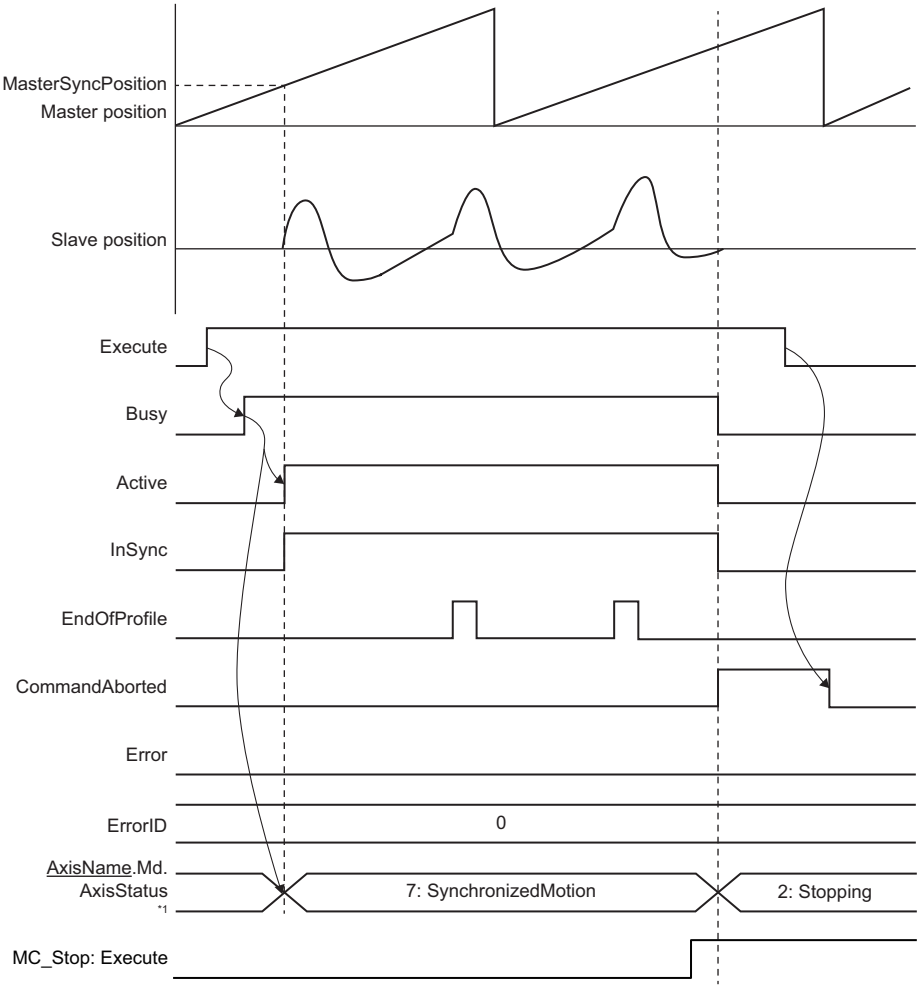
Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
96	48	Subroutine type	Real-time execution

■Timing chart

[When the FB is normally completed]

Ex.

When StartMode is "1: mcAbsolute" and MasterAbsolute is "FALSE: Relative coordinate"



*1 This is the state of the Slave axis.

[When an error occurs]

For details when an error occurs, refer to the following.

☞ Page 27 Execute type/Enable type

■I/O variables

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Master axis	Master	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables
Slave axis	Slave	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables

■Input variables

Import ↑: At start, R: Retrigger possible, C: Continuous update possible

Name	Variable name	Data type	Import	Setting range	Default value	Description
Execute command	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Continuous update	ContinuousUpdate	BOOL	↑	TRUE, FALSE	FALSE	During TRUE, master axis offset, slave axis offset, master axis scaling, slave axis scaling, and cam table ID can be changed continuously.
Master axis offset	MasterOffset	LREAL	↑/R/C	-10000000000.0 to 10000000000.0	0.0	☞ Page 457 MasterOffset (Master axis offset)
Slave axis offset	SlaveOffset	LREAL	↑/R/C	-10000000000.0 to 10000000000.0	0.0	☞ Page 457 SlaveOffset (Slave axis offset)
Master axis scaling	MasterScaling	LREAL	↑/R/C	0.01 to 10.0	1.0	☞ Page 457 MasterScaling (Master axis scaling)
Slave axis scaling	SlaveScaling	LREAL	↑/R/C	0.01 to 10.0	1.0	☞ Page 458 SlaveScaling (Slave axis scaling)
Master axis follow-up distance	MasterStartDistance	LREAL	↑	-10000000000.0 to 10000000000.0	0.0	☞ Page 458 MasterStartDistance (Master axis follow-up distance)
Master axis synchronization start position	MasterSyncPosition	LREAL	↑	-10000000000.0 to 10000000000.0	0.0	☞ Page 458 MasterSyncPosition (Master axis synchronization start position)
Start mode	StartMode	MC_START_MODE	↑	0 to 2	0	☞ Page 458 StartMode (Start mode)
Master axis data source selection	MasterValueSource	MC_SOURCE	↑	1, 2, 101, 102	1	☞ Page 458 MasterValueSource (Master axis data source selection)
Cam table ID	CamTableID	MC_CAM_ID	↑/R/C	1 to 6000	0	☞ Page 458 CamTableID (Cam table ID)
Buffer mode	BufferMode	MC_BUFFER_MODE	↑	0, 1	0	☞ Page 459 BufferMode (Buffer mode)
Options	Options	DWORD(HEX)	↑	For details, refer to the following. ☞ Page 460 Options	00000000H	For details, refer to the following. ☞ Page 460 Options

■Output variables

Name	Variable name	Data type	Default value	Description
In synchronization	InSync	BOOL	FALSE	☞ Page 460 InSync (In synchronization)
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Controlling	Active	BOOL	FALSE	Indicates that the FB is controlling the axis.
Abortion of execution	CommandAborted	BOOL	FALSE	Indicates that another FB has aborted the execution of the FB. When this FB is stopped by an error or when this FB is started up while an error occurs, it turns TRUE. Turns FALSE when Execute turns FALSE.
Error	Error	BOOL	FALSE	When TRUE, it indicates that an error has occurred in the FB.
Error code	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned. For details, refer to the following. ☞ Page 662 List of Error Codes
Cam cycle completion	EndOfProfile	BOOL	FALSE	☞ Page 460 EndOfProfile (Cam cycle completion)

■Public variables

Name	Variable name	Data type	Default value	Description
Current value per cycle	InputsPerCycle	LREAL	0.0	Stores the current value per cycle.
Reference value	Reference	LREAL	0.0	Stores the reference value.
Output value	OutputData	LREAL	0.0	Stores the output value.
Instance ID	InstanceID	WORD(UINT)	0	Instance ID that is automatically set when an instance is created. Used in FB input, etc.

Control details

MC_CamIn sets MasterOffset, SlaveOffset, MasterScaling, SlaveScaling, MasterStartDistance, MasterSyncPosition, StartMode, MasterValueSource, CamTableID, and BufferMode and then executes the cam operation.
To stop the operation, execute MC_Stop.

Starting cam operation

Synchronous timing of the cam operation and reflection timing of the cam control data can be selected by setting of StartMode.

■Cam operation synchronous timing

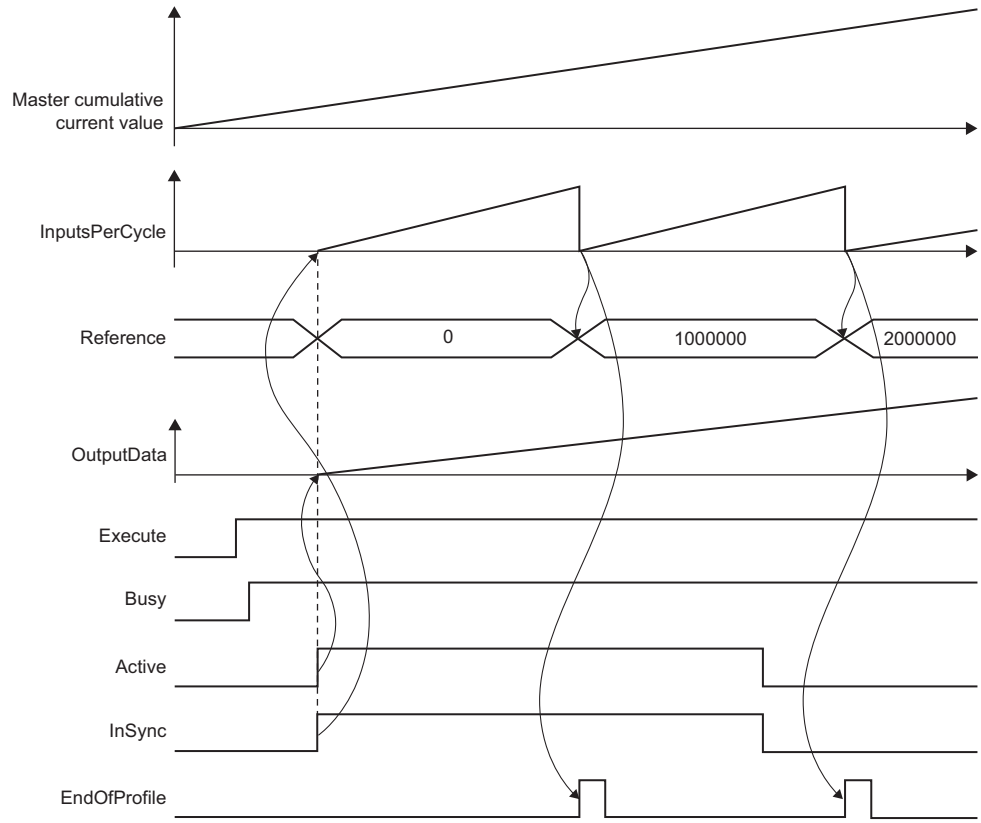
StartMode decides Enabled/Disabled setting of related parameters (MasterSyncPosition and MasterStartDistance) and in this setting, the synchronous timing of MC_CamIn.InputsPerCycle, Reference, and OutputData is as follows.

StartMode	MasterSync Position	MasterStart Distance	Synchronous timing	
			InputsPerCycle	Reference, OutputData
0: mcImmediate (Immediate)	Disabled	Disabled	When executing MC_CamIn	
1: mcAbsolute (Absolute)	Enabled	Disabled	When the cumulative current value of master axis passes MasterSyncPosition	
2: mcRelative (Relative)*1	Enabled	Enabled	When the cumulative current value of master axis passes MasterSyncPosition	When the cumulative current value of master axis passes MasterSyncPosition + MasterStartDistance

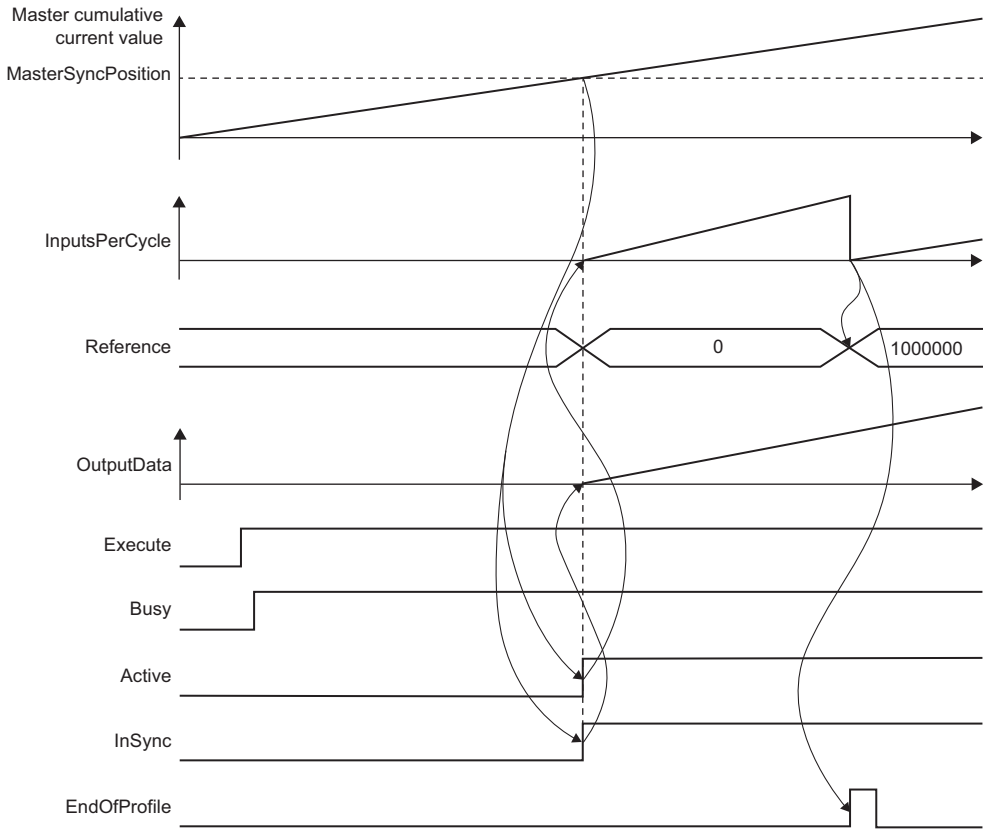
*1 When using this setting, the error "Out of Start Mode Range" (error code: 1A47H) occurs.

After MC_CamIn is executed, Active turns TRUE after MC_CamIn.InputsPerCycle is synchronized, and then InSync turns TRUE after OutputData is synchronized.

■When "0: mcImmediate" is selected for StartMode



■When StartMode is "1: mcAbsolute"



Cam table

The following parameters are set as optional information by MC_CamTableSelect.

Name	Variable name	Specifying interpolation method		
		0: Linear interpolation	1: Specifying for each section	2: Spline interpolation
Periodic	Periodic	○	○	
Input absolute coordinate	MasterAbsolute	○	○	
Output absolute coordinate	SlaveAbsolute	○	○	
Specifying interpolation method	Interpolate	○	○	
Length per cycle	CycleLength	○	○	
Stroke amount	Stroke	○	○	
Start point	StartPoint	×	○	
Initial stroke amount	StartStroke	×	○	

The cam operation when executing MC_CamIn using parameters is shown below.

■Periodic

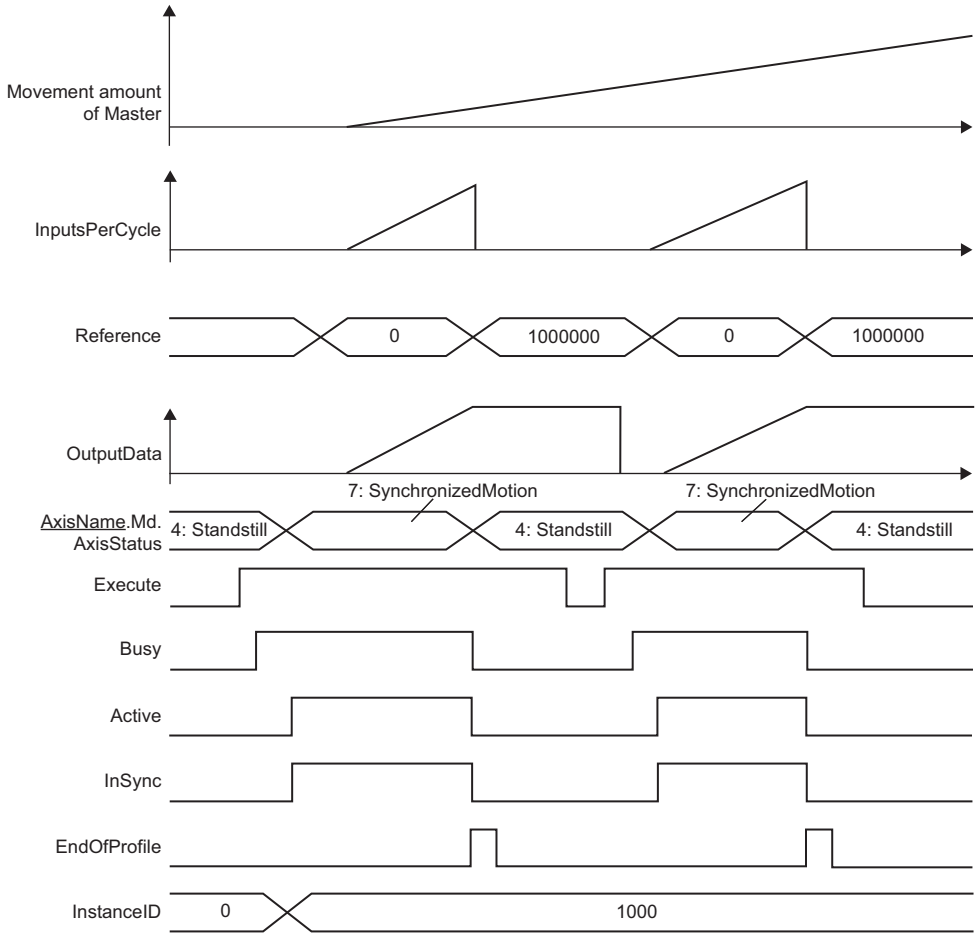
Periodic can set "FALSE: Non periodic" and "TRUE: Periodic". The following explains the operation when executing MC_CamIn.

• Periodic "FALSE: Non periodic" is specified

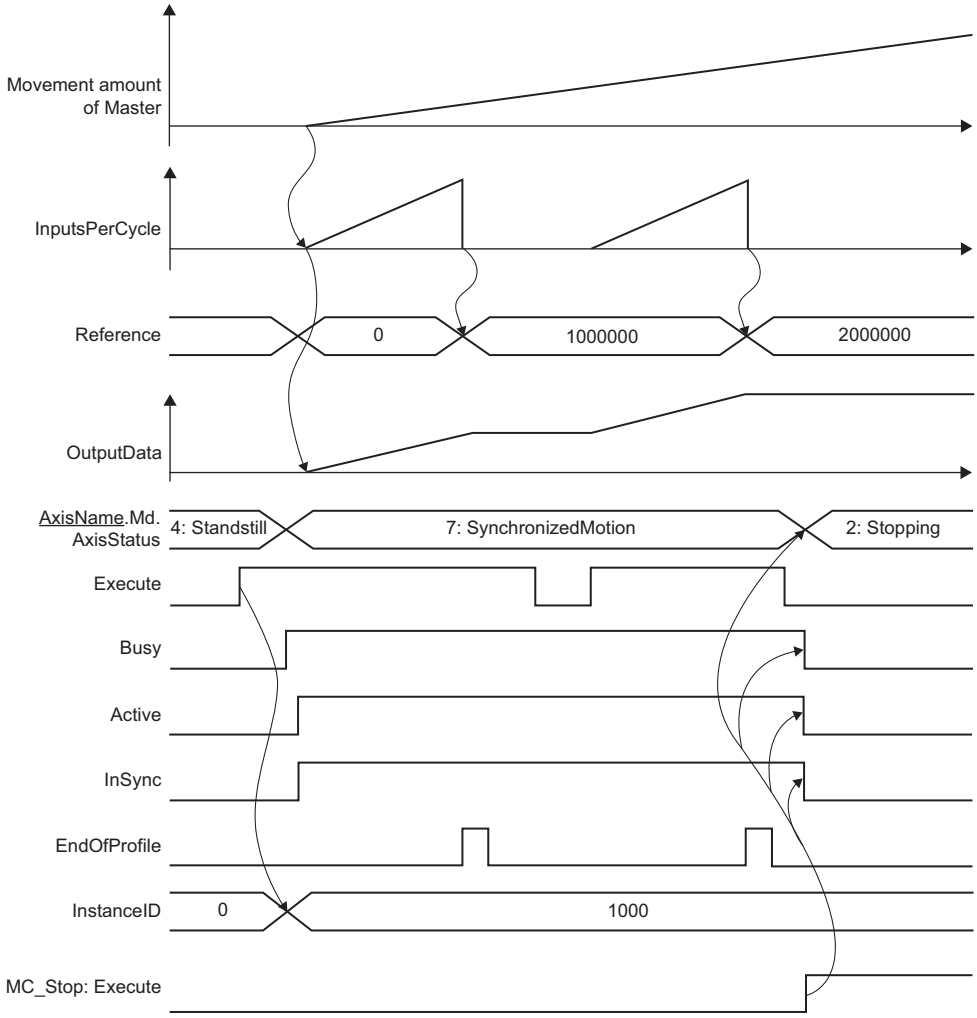
Synchronization with the master axis movement amount for length per cycle is carried out after Active turns TRUE. After that, synchronizing with the master axis movement amount for length per cycle is carried out again by retrigger. This time operation can be selected by Options bit16 as follows.

Options bit16	Operation after synchronizing with the master axis movement amount for length per cycle				Reference and OutputData at retrigger
	Busy	Active	InSync	Axis status	
0: Non periodic	FALSE			Updated to "4: Standstill"	Initialized
1: Periodic	TRUE			Maintains "7: SynchronizedMotion"	Maintained

[When Options bit16 is "0: Non periodic"]



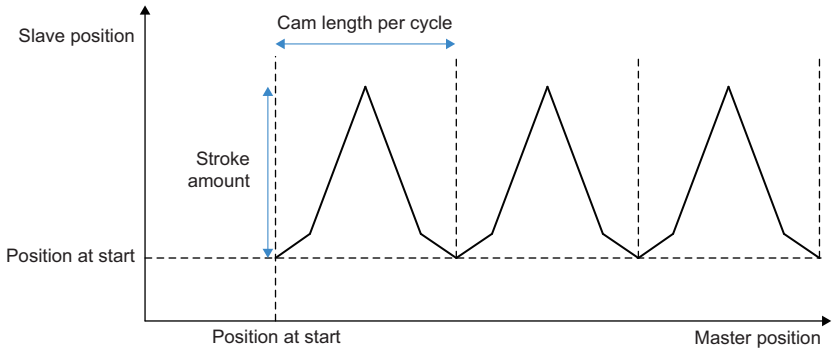
[When Options bit16 is "1: Periodic"]



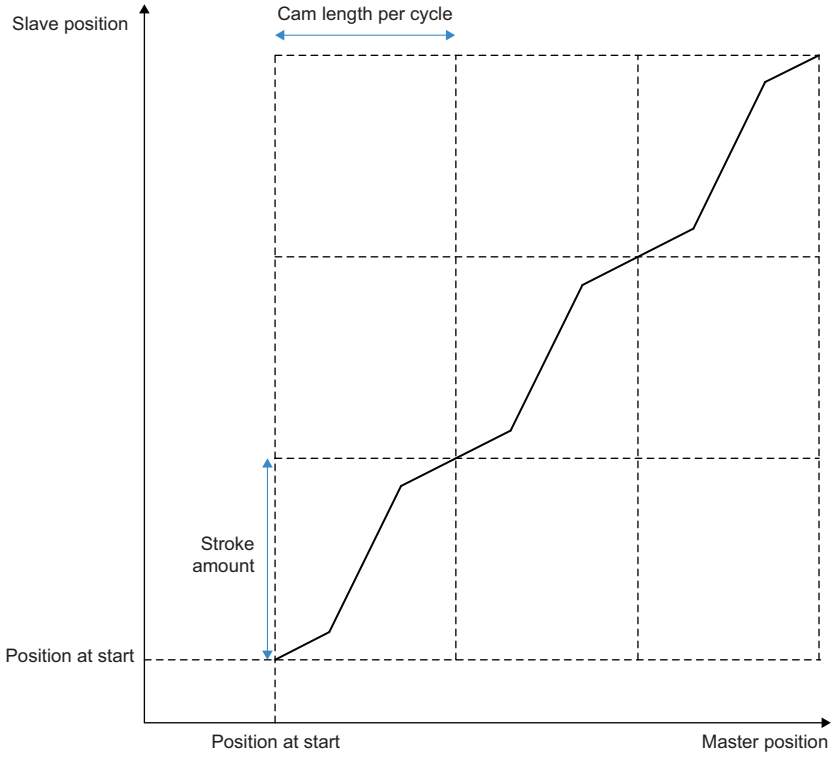
• Periodic "TRUE: Periodic" is specified

Cam table execution is repeated continuously. When the positions of Slave at the cam table start point are the same each time by repeating, cam is operated as a two-way cam. When the set points at the start point and the end point are different, cam is operated as a feed cam.

[Two-way cam operation]



[Feed cam operation]



■Master axis absolute coordinate

The following explains the operation at MasterAbsolute set by MC_CamTableSelect.

• "0: Relative coordinate" is specified to the master axis absolute coordinate

MasterSyncPosition becomes the start point of the cam data. When InSync turns TRUE, cam operation is executed corresponding to the relative movement amount of Master.

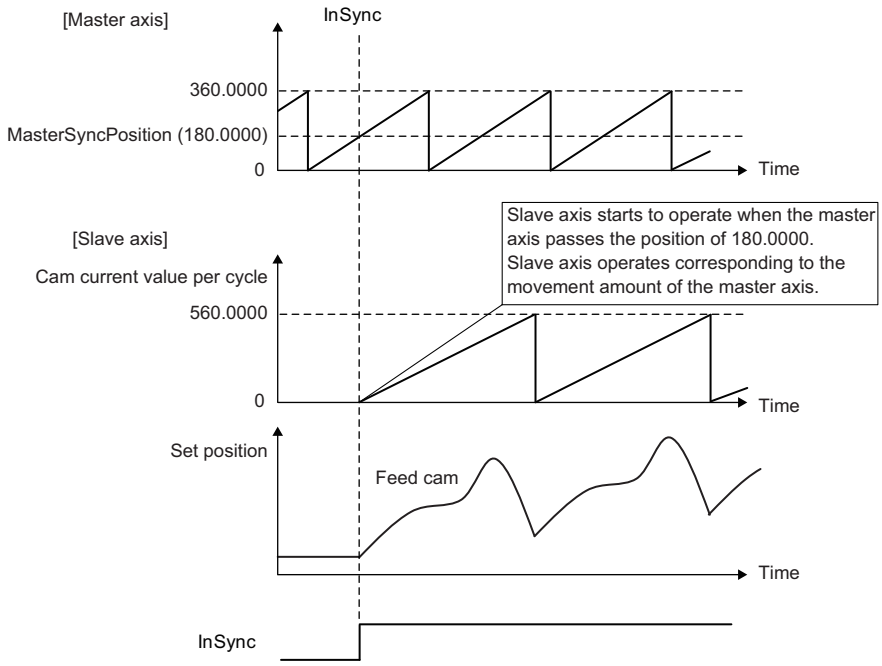
Cam operation is executed continuously, even when ring counters of Master and cam table are not matched.

Ex.

Master axis: Ring counter of the current value: 0 to 360.0000 [degree]

MasterSyncPosition: 180.0000 [degree]

Cam: Cam length per cycle: 540.0000 [degree]



• "1: absolute coordinate" is specified to the master axis absolute coordinate

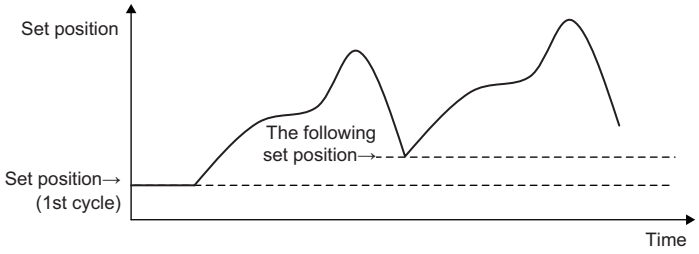
When using this setting, the error "Out of Input Absolute Coordinate Range" (error code: 341DH) occurs.

■Slave axis absolute coordinate

The following explains the operation at SlaveAbsolute set by MC_CamTableSelect.

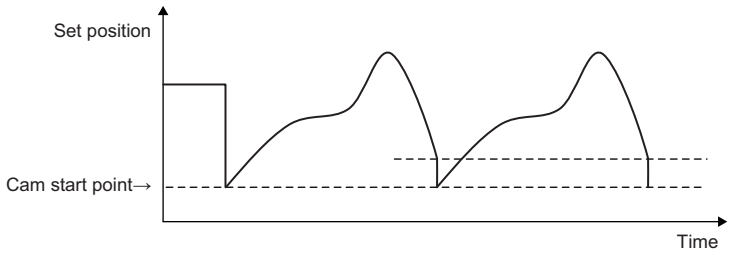
• "0: relative coordinate" is specified to the slave axis absolute coordinate

When InSync turns TRUE, the slave axis starts the operation of the cam profile from the current position (set position).
When the periodic of the cam profile is set to "TRUE: Periodic", the next cycle starts from the stroke position (set position) where the last cycle completed.



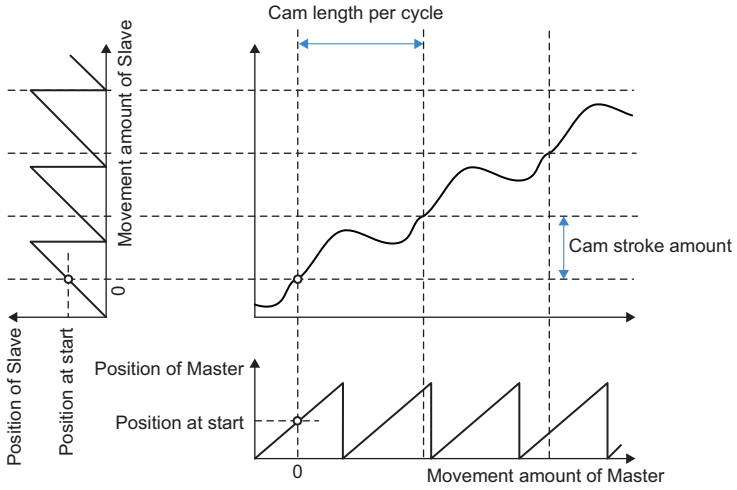
• "1: absolute coordinate" is specified to the slave axis absolute coordinate

Cam is operated to control that the set position of Slave at the point where InSync turns TRUE is set to the start point of the cam profile when starting a cam.
When the periodic of the cam profile is set to "TRUE: Periodic", when the stroke positions of the cam profile differ between the start point and the end point, command is output per operation cycle to return the stroke position back to the first set point when the next cycle starts. If the command current value at this time is large, the position command and speed command to the drive unit become large and that causes the servo alarm [AL. 035 Command frequency error] of the MR-J5(W)-G.

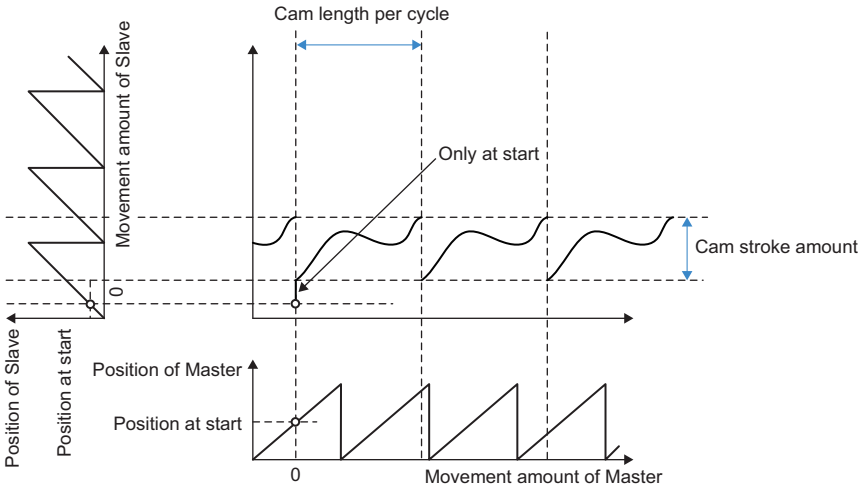


The following explains the relationships of the position between Master and Slave based on the combination of the master axis absolute coordinate and the slave axis absolute coordinate.

• When "0: relative coordinate" is specified to the master axis absolute coordinate and "0: relative coordinate" is specified to the slave axis absolute coordinate



- When "0: relative coordinate" is specified to the master axis absolute coordinate, and "1: absolute coordinate" is specified to the slave axis absolute coordinate



■Length per cycle

Set the input amount that is required for one cycle. For details, refer to the following.

☞ Page 455 MC_CamIn.InputsPerCycle (Cam current value per cycle)

■Stroke amount

Set the stroke amount corresponding to its ratio 100%. For datils, refer to the following.

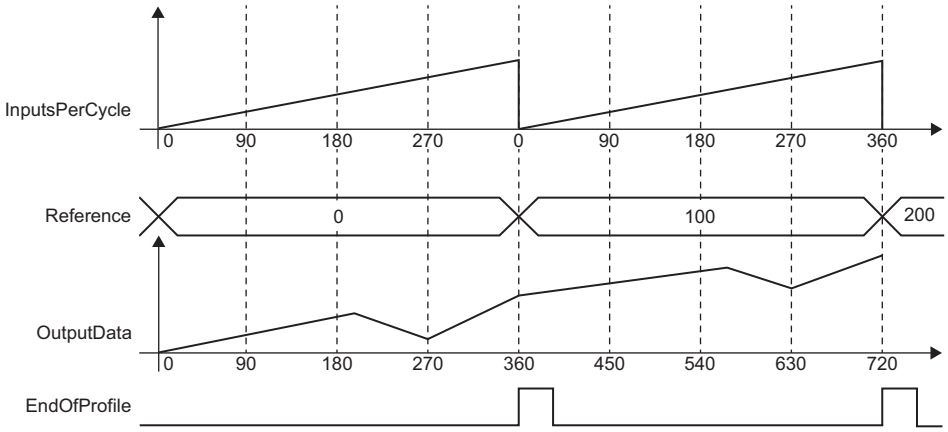
☞ Page 457 MC_CamIn.OutputData (Output value)

■Start point and Initial stroke amount

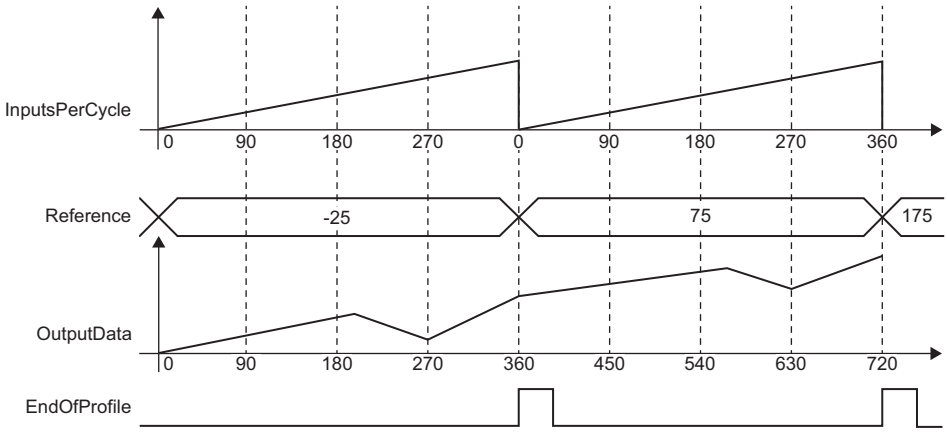
When the specifying interpolation method is "Specifying for each section" and "Spline interpolation", the initial value of MC_CamIn.InputsPerCycle and MC_CamIn.OutputData can be set by using the start point and the initial stroke amount. (When the specifying interpolation method is "0: Linear interpolation", the start point and the initial value is fixed to 0.)
The initial stroke amount is subtracted from Reference not to operate the slave axis by the setting value of the Initial stroke amount.

Parameter	Description
Start point	Sets InputsPerCycle when Active turns to TRUE.
Initial stroke amount	Sets OutputData when InSync turns to TRUE. Subtracts Initial stroke amount from Reference.

[When the start point and the Initial stroke amount are "0"]



[The start point is "0" and the Initial stroke amount is "25"]



Monitor data during cam operation

MC_CamIn can confirm the position relationship of cam data by the data of MC_CamIn.InputsPerCycle, MC_CamIn.Reference, and MC_CamIn.OutputData.

■MC_CamIn.InputsPerCycle (Cam current value per cycle)

The monitor value range is the cam length per cycle range ("0 to (cam length per cycle - 1)") set in the cam table. The initial value is the start point during cam table. For details, refer to the following.

☞ Page 454 Start point and Initial stroke amount

The movement amount of the master axis is reflected after Active turns to TRUE.

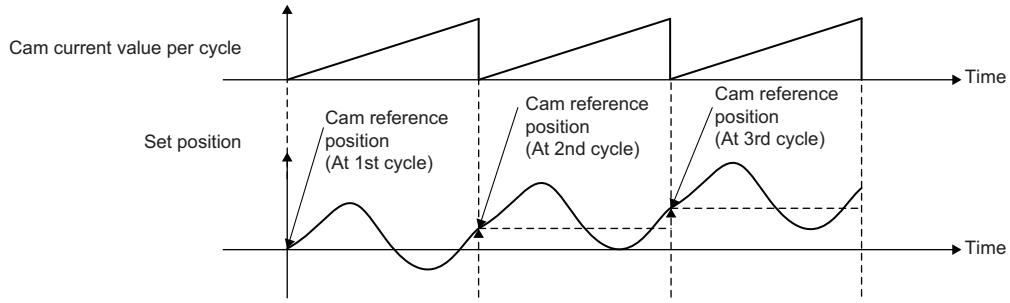
■MC_CamIn.Reference (Cam reference value)

When MC_CamIn is started to operate, cam reference value is calculated by the cam position for start (cam current value per cycle) based on set position where InSync becomes TRUE. The formula for initial value is as follows because the Initial stroke amount during cam table is subtracted.

Cam reference value = 0 - (stroke value when InSync turns TRUE) - the Initial stroke amount

The cam reference position is not updated in two-way cam operation. In a feed cam, the cam reference value is calculated as shown below.

Cam reference value = Original cam reference value + (Stroke value at the last point - Stroke value at the 1st point)



Reference value is updated at the following timing

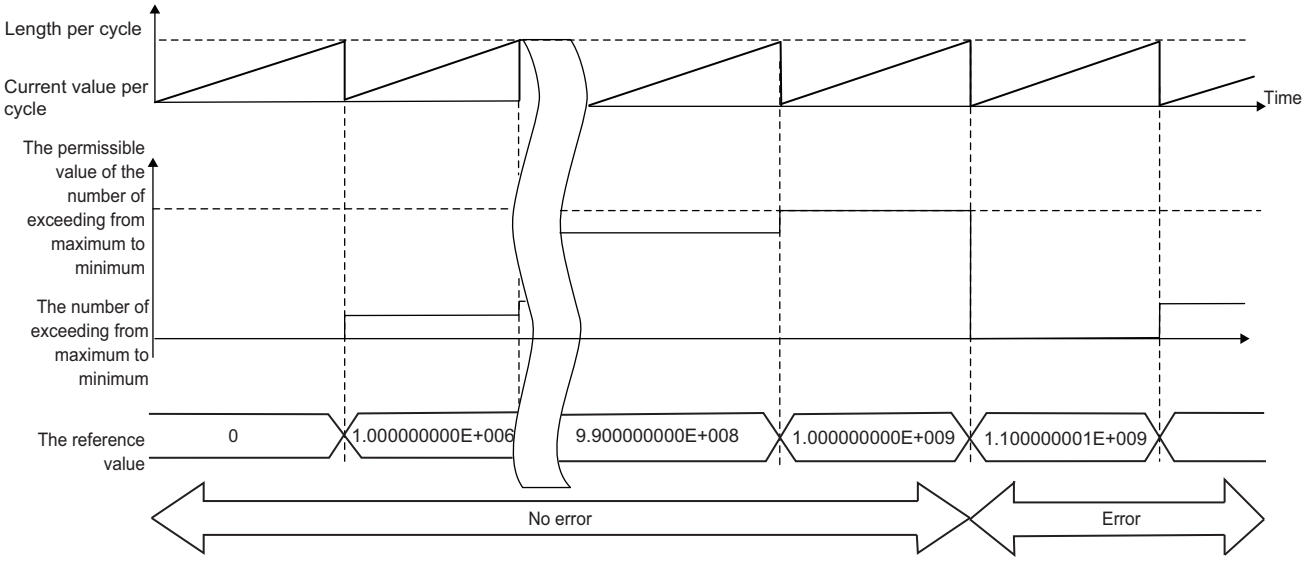
Update timing	The calculation formula of reference value
When InputsPerCycle passes the start point of cam table	Original cam reference value + (Stroke value at the last point - Stroke value at the 1st point)
When changing current value per cycle (MCv_ChangeCycle.Done is TRUE)	Original cam reference value + (Stroke value corresponding to current value per cycle after change × -1)

Restriction

For the reference value error

The following operation may cause the error of reference value.

- Condition 1: When the number of output TRUE of EndOfProfile after InSync turns to TRUE exceeds the permissible value of the number of exceeding from maximum to minimum.
  Although the error does not occur unless the number of exceeding from maximum to minimum per cycle exceeds its permissible value, if the value exceeds the permissible value, the error may occur. (The operation is maintained regardless of the operation error occurrence or not.)



- Condition 2: When the following control parameter includes decimal
  Stroke value at the last point - Stroke value at the 1st point
  Slave axis coefficient
- Condition 3: When changing current value per cycle is repeated in relative specification

When the reference value error becomes large, clear the synchronization status or reset it by executing current value change of absolute position specification.

For the error, refer to the following.

☞ Page 83 Positioning Range

The permissible value of the number of exceeding from maximum to minimum is a variable value calculated with the following formula.

The permissible value of the number of exceeding from maximum to minimum = the limit value of the number of exceeding from maximum to minimum / ((output value at the last point - output value at the 1st point) / the slave axis coefficient)

The limit value of the number of exceeding from maximum to minimum is 10000000000 as the upper limit value of the positioning range.

(e.g.) When the output value at the 1st point = 0, the output value at the last point = 1000000, and the output coefficient = 10, the permissible value of the number of exceeding from maximum to minimum becomes "100000".

■MC_CamIn.OutputData (Output value)

Reference value is the initial stroke amount during cam table. For details, refer to the following.

☞ Page 454 Start point and Initial stroke amount

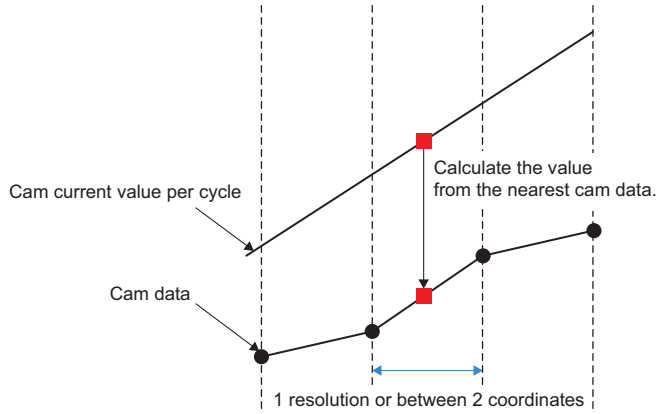
While InSync is TRUE, MC_CamIn.Output is calculated as shown below.

Output value = Cam reference position + (Cam stroke amount × Stroke ratio corresponding to cam current value per cycle)

or

Output value = Cam reference position + (Output value corresponding to cam current value per cycle)

When the cam current value per cycle is in between the cam data, the value is calculated from the nearest cam data.



Details of setting items

■MasterOffset (Master axis offset)

The phase of the master axis can be shifted for the amount of offset by setting MasterOffset. (This does not affect MasterStartDistance or MasterSyncPosition.)

When other than "0" is set at operation start, the phase of master axis is changed to the cam position per cycle which is added the amount of offset for position of the Master at rising edge of InSync. The operation is the same as when changing while InSync is TRUE.

■SlaveOffset (Slave axis offset)

The displacement of the Slave is shifted for amount of offset.

When other than "0" is set before starting up InSync, the command which is added the position of Slave for the amount of offset at rising edge of InSync is output. The operation is the same as when InSync is changed during TRUE.

■MasterScaling (Master axis scaling)

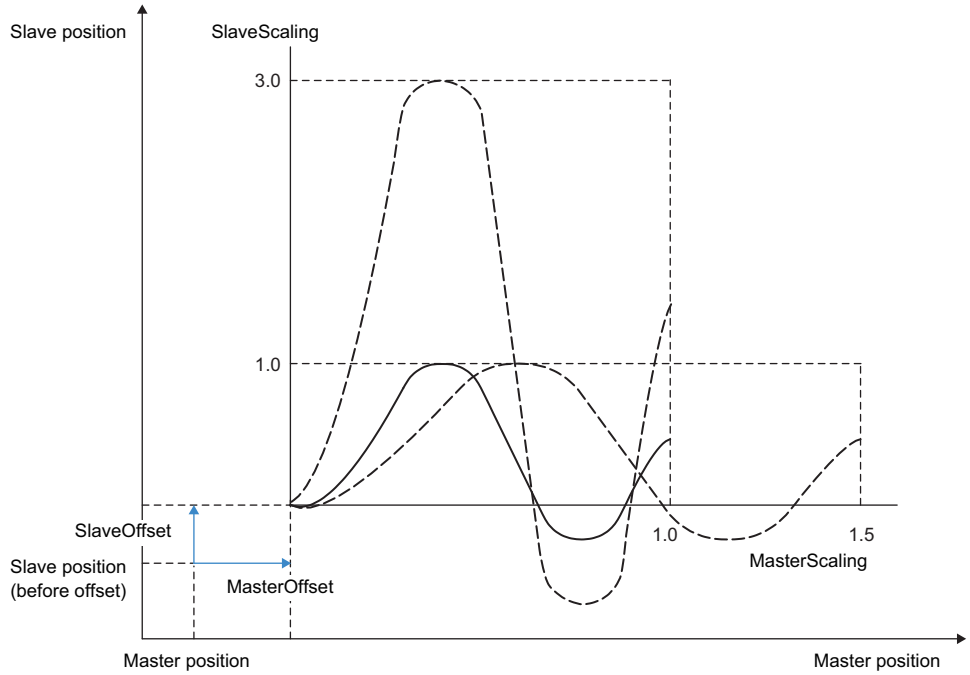
Expand/reduce the cam length per cycle of the cam table.

When changing during synchronization, a command for the moving amount is output to adjust the position of Slave to the position of the cam table.

■SlaveScaling (Slave axis scaling)

Expand/reduce the cam stroke amount of the cam table.

When changing during synchronization, a command for the moving amount is output to adjust the position of Slave to the position of the cam table.



■MasterStartDistance (Master axis follow-up distance)

OutputData sets the position of Master (the relative position from MasterSyncPostion) which starts synchronization. For details, refer to the following.

☞ Page 446 Starting cam operation

■MasterSyncPosition (Master axis synchronization start position)

MC_CamIn.InputsPerCycle sets the position of Master which starts synchronization. For details, refer to the following.

☞ Page 446 Starting cam operation

This setting is not used when "0: mcImmediate" is set to StartMode.

■StartMode (Start mode)

Select "0: mcImmediate"or "1: mcAbsolute". For details, refer to the following.

☞ Page 446 Starting cam operation

When "2: mcRelative" is specified, the error "Out of Start Mode Range" (error code: 1A47H) occurs.

■MasterValueSource (Master axis data source selection)

Select the data source of Master. For details, refer to the following.

☞ Page 440 Master Axis Data Source Selection

■CamTableID (Cam table ID)

Set the ID for cam. Open the cam ID to the open area by MC_CamTableSelect before use.

Set the following.

FB	Name	Variable name
MC_CamTableSelect	Cam tableID	CamTable.ProfileData.ID.Number

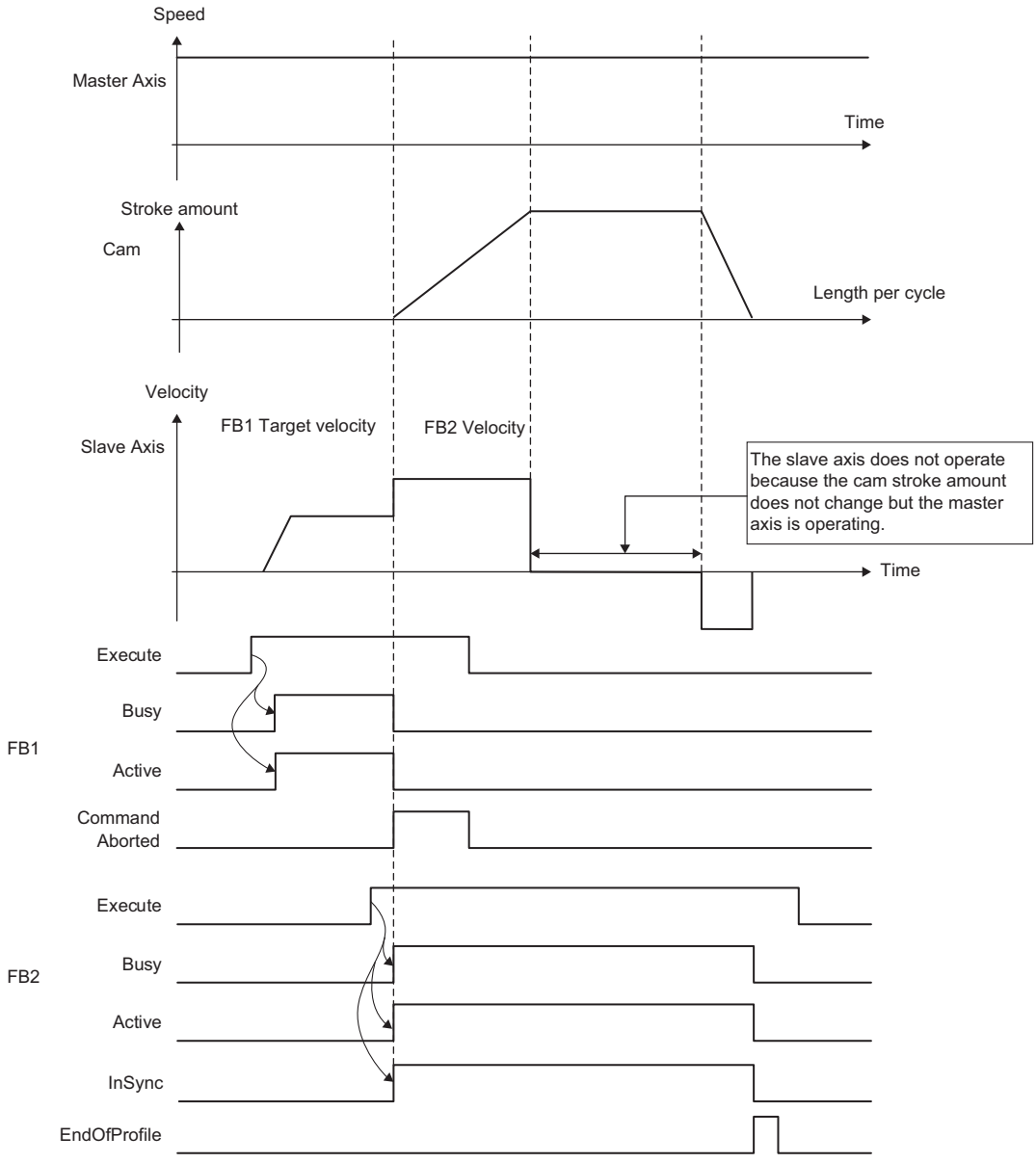
■BufferMode (Buffer mode)

Select the buffer mode. Aborting and Buffered can be set.

When the FB is switched, the speed changes immediately based on the master axis speed and the set cam data as the speed of MC_CamIn follows the synchronizing FB (the master axis). The switching condition at each mode is shown below.

BufferMode	Switching condition
Aborting	Always
Buffered	EndOfProfile = TRUE

An operation example when Aborting is executed in StartMode = "0:mcImmediate" to the on-going FB1 with the linked start FB2 (slave axis: MC_CamIn) is shown below.



The speed of FB2 occurs after InSync turns TRUE.

When StartMode is other than "0: mcImmediate", the speed becomes 0 until it turns from Busy = TRUE to InSync = TRUE because InSync does not turn TRUE immediately after it started.

For details on operation and the combination with the other FBs, refer to the following.

☞ Page 133 Buffer mode type

■Options

Set the function options that are used in MC_CamIn by specifying the bit.

The details of bit and the functions are shown below.

Bit	Function description
0 to15	Empty (Specify "0".)*1
16	Specifying the operation after one cycle operation. Periodic of the cam table specifies the operation after one cycle operation when "FALSE: Non periodic" is set. 0: Non periodic 1: Periodic For detail, refer to the following. ☞ Page 448 Periodic
17 to 31	Empty (Specify "0".)*1

*1 If specifying other than "0", the error "Out of Options Range" (error code: 1A4EH) occurs, and the function does not start.

■InSync (In synchronization)

It turns TRUE when OutputData starts synchronization.

■Busy (Executing)

It turns TRUE during execution of the FB.

■Active (Controlling)

It turns TRUE when MC_CamIn.InputsPerCycle starts synchronization.

■CommandAborted (Abortion of execution)

It turns TRUE if other FBs are started during execution of this FB.

■Error (Error)

It turns TRUE if an error occurs in this FB.

■ErrorID (Error code)

It returns an error code occurred in this FB.

■EndOfProfile (Cam cycle completion)

It turns TRUE for one scan in the program cycle every movement of the length per cycle after Active turns TRUE.

Precautions

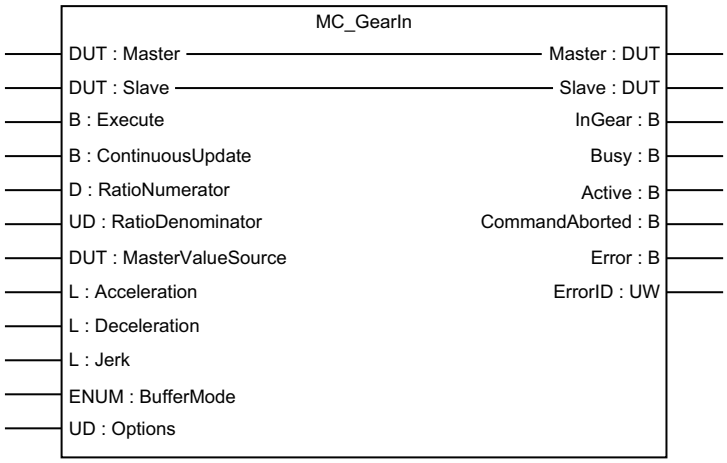
- When other than "0" is set to MasterOffset or SlaveOffset before the rising edge of InSync, the set position of Slave is changed directly at the rising edge of InSync.
  When changing MasterOffset or SlaveOffset while InSync is TRUE, the set position of Slave is immediately changed directly.
  To execute a smoothing process for the movement amount change, use it with MCv_SmoothingFilter.
- Display unit of following items in the axis monitor on GX Works3 is displayed with the setting unit of the axis label name.Pr.Unit_Position.
  Master (Master axis)
  Slave (Slave axis)

13.8 Gear Operation

This function sets the speed ratio between the master axis and the slave axis, and starts gear operation.

Relevant FB

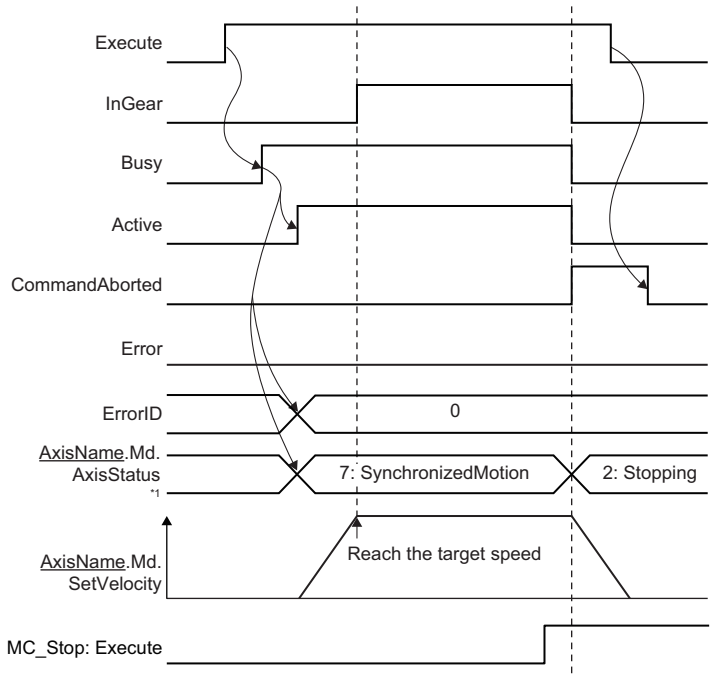
MC_GearIn

Item	Description
Function overview	Gear operation starts operation according to the specified gear ratio.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

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

■Timing chart

[When the FB is normally completed]



*1 This is axis state of Slave axis.

[When an error occurs]

For details when an error occurs, refer to the following.

☞ Page 27 Execute type/Enable type

■I/O variables

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Master axis	Master	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables
Slave axis	Slave	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables

■Input variables

Import ↑: At start, R: Retrigger possible, C: Continuous update possible

Name	Variable name	Data type	Import	Setting range	Default value	Description
Execute command	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Continuous update	ContinuousUpdate	BOOL	↑	TRUE, FALSE	FALSE	During TRUE, it can continuously change the gear ratio numerator, the gear ratio denominator, the acceleration, and the deceleration.
Gear ratio numerator	RatioNumerator	DINT	↑/R/C	-2147483648 to 2147483647	1	☞ Page 464 RatioNumerator (Gear ratio numerator)
Gear ratio denominator	RatioDenominator	DWORD(UDINT)	↑/R/C	1 to 2147483647	1	☞ Page 464 RatioDenominator (Gear ratio denominator)
Master axis data source selection	MasterValueSource	MC_SOURCE	↑	1, 2, 101, 102	1	☞ Page 464 MasterValueSource (Master axis data source selection)
Acceleration	Acceleration	LREAL	↑/R/C	☞ Page 335 Input variables to use	0.0	☞ Page 465 Acceleration (Acceleration)
Deceleration	Deceleration	LREAL	↑/R/C	☞ Page 335 Input variables to use	0.0	☞ Page 465 Deceleration (Deceleration)
Jerk	Jerk	LREAL	↑	☞ Page 335 Input variables to use	0.0	☞ Page 465 Jerk (Jerk)
Buffer mode	BufferMode	MC_BUFFER_MODE	↑	0, 1	0	☞ Page 465 BufferMode (Buffer mode)
Options	Options	DWORD(HEX)	↑	00000000H	00000000H	Sets "00000000H". (A value other than "00000000H"causes the error "Out of Options Range" (error code: 1A4EH).)

■Output variables

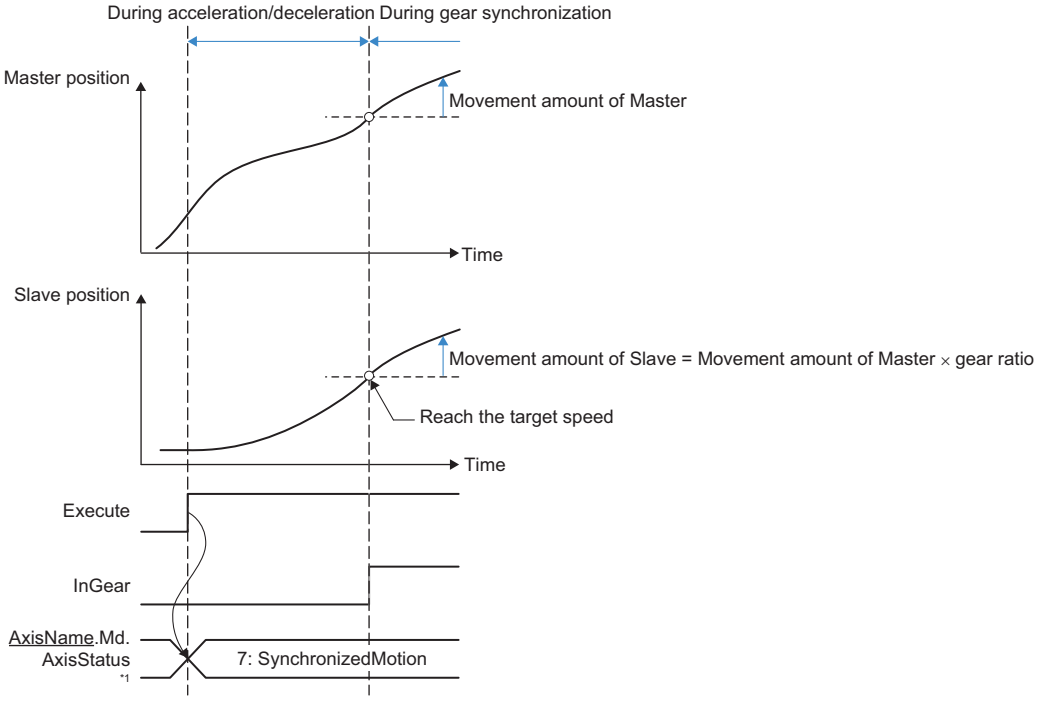
Name	Variable name	Data type	Default value	Description
Gear ratio reached	InGear	BOOL	FALSE	Indicates that the target velocity is reached.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Controlling	Active	BOOL	FALSE	Indicates that the FB is controlling the axis.
Abortion of execution	CommandAborted	BOOL	FALSE	Indicates that another FB has aborted the execution of the FB. When this FB is stopped by an error or when this FB is started up while an error occurs, it turns TRUE. Turns FALSE when Execute turns FALSE.
Error	Error	BOOL	FALSE	When TRUE, it indicates that an error has occurred in the FB.
Error code	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned. For details, refer to the following. ☞ Page 662 List of Error Codes

Control details

In the MC_GearIn, RatioNumerator, RatioDenominator, MasterValueSource, Acceleration, Deceleration, Jerk, and BufferMode are set and the gear operation starts.

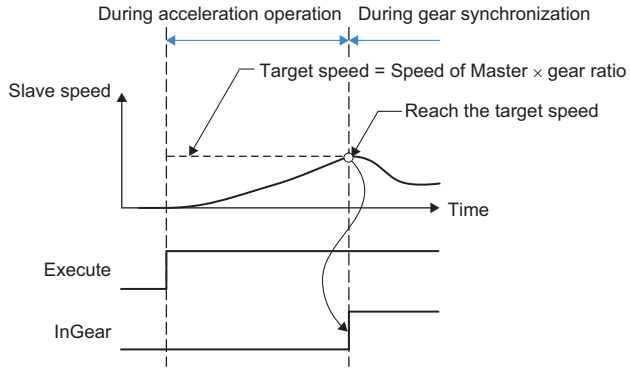
To stop the operation, execute the MC_Stop.

After executing the FB, Slave continues acceleration/deceleration until reaching the target velocity which is a value converted the speed of Master by the gear ratio. After reaching the target velocity, InGear turns TRUE, and Slave is controlled with the speed converted the speed of Master by gear ratio.

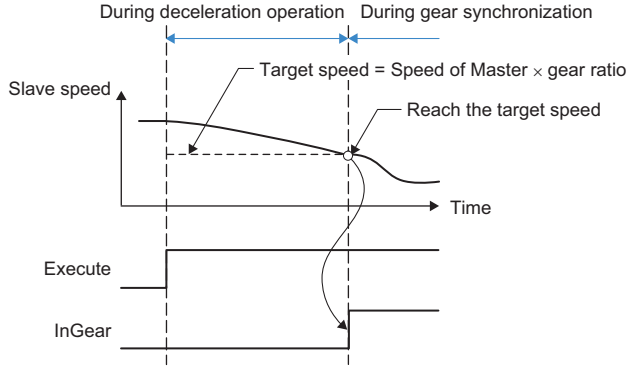


*1 This is axis state of Slave axis.

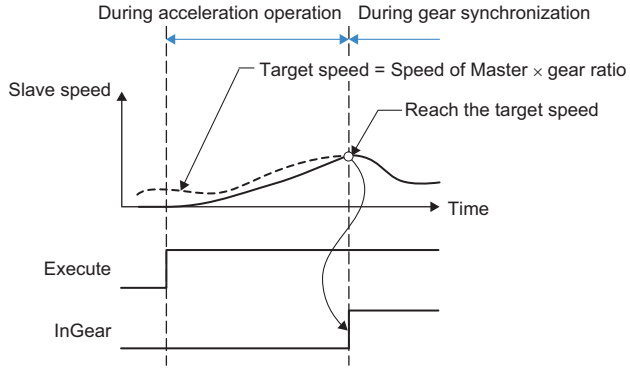
- Accelerate when the speed of Slave is smaller than the target velocity at operation start.



- Decelerate when the speed of Slave is larger than the target velocity at operation start.



• When the speed of Master is changed during acceleration/deceleration, the target velocity is updated.



Details of setting items

■RatioNumerator (Gear ratio numerator)

Set the numerator value for converting the speed of Master and transmitting.

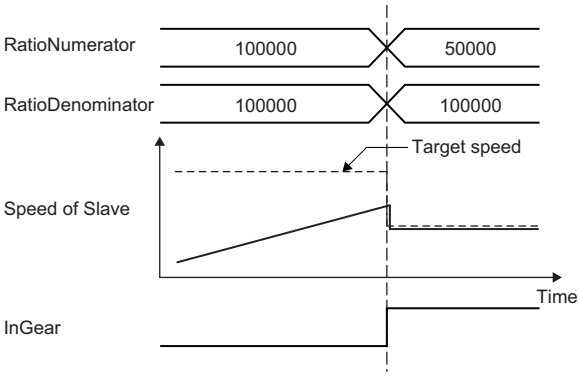
The speed is converted as shown below.

$$\text{Master speed after conversion} = \text{Master speed before conversion} \times \frac{\text{RatioNumerator}}{\text{RatioDenominator}}$$

The speed direction can be reversed and transmitted by setting a negative value in RatioNumerator.

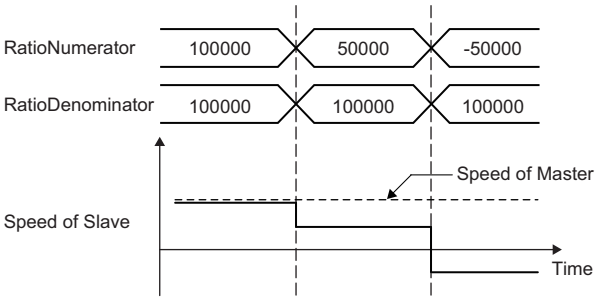
The target velocity is changed if the gear ratio is changed during acceleration/deceleration. Therefore, the speed of Slave is changed directly and the gear operation may be started depends on the change of the target velocity.

• Changing gear ratio during acceleration/deceleration



When the gear ratio is changed during gear synchronization, the speed of Slave is changed directly.

• Changing gear ratio during gear synchronization



■RatioDenominator (Gear ratio denominator)

Set the denominator for converting the movement amount of Master and composing.

Set together with RatioNumerator.

Set a positive value in RatioDenominator.

■MasterValueSource (Master axis data source selection)

Select the data source of Master for transmitting the speed of Master. For details, refer to the following.

☞ Page 440 Master Axis Data Source Selection

■Acceleration (Acceleration)
Set the acceleration for accelerating.
When reaching the target velocity, InGear turns TRUE and Slave is controlled with the speed which converted the speed of Master by the gear ratio.

■Deceleration (Deceleration)
Set the deceleration for decelerating.
When reaching the target velocity, InGear turns TRUE and Slave is controlled with the speed which converted the speed of Master by the gear ratio.

■Jerk (Jerk)
Set Jerk at acceleration/deceleration start.

■BufferMode (Buffer mode)
Select the buffer mode. Aborting and Buffered can be set. For details on operation, refer to the following.
☞ Page 133 Buffer mode type

■InGear (Gear ratio reached)
It turns TRUE when reaching the target velocity.

■Busy (Executing)
It turns TRUE during execution of the FB.

■Active (Controlling)
It turns TRUE during controlling Slave.

■CommandAborted (Abortion of execution)
It turns TRUE if other FBs are started during execution of this FB.

■Error (Error)
It turns TRUE if an error occurs in this FB.

■Error ID (Error code)
It returns an error code occurred in this instruction.

Precautions
- When the gear ratio is changed, the speed of the slave axis is changed directly. To smooth the speed change, use with MCv_SmoothingFilter.
- When Master speed after conversion exceeds the acceleration/deceleration upper limit value during acceleration/deceleration operation, the warning "Acceleration Time Limit Over Warning" (warning code: 0D04H) or the warning "Deceleration Time Limit Over Warning" (warning code: 0D05H) will occur and stops the acceleration/deceleration operation, and continues the operation with the speed when the warning was detected.
  When the above warning occurred, adjust and set Master speed after conversion, Acceleration or Deceleration to not exceed the acceleration/deceleration upper limit value, and restart the acceleration/deceleration operation.

Restriction

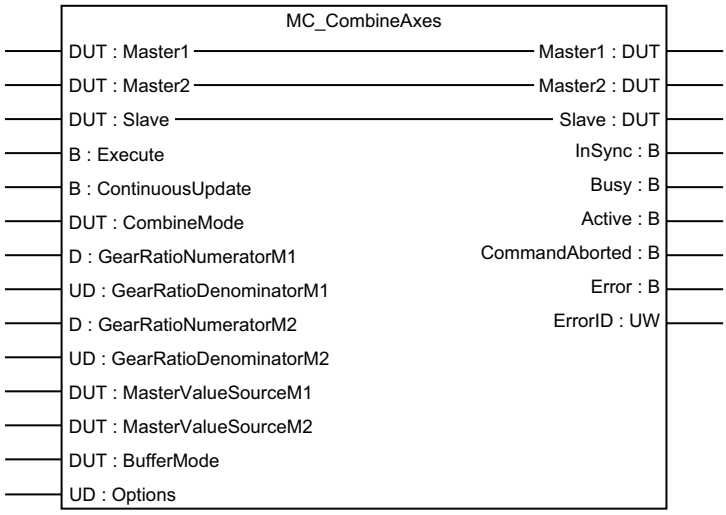
For the error between the master axis and the slave axis at the gear operation
- The error of position between the master axis and the slave axis may increase if the movement amount from the position which InGear of the master axis or the slave axis turns to TRUE exceeds ±10000000000.0. To set the movement range of the master axis and the slave axis within the above stated range, consider the gear ratio and the axis unit setting.
- When the error increases, clear synchronization or change current value specified the absolute position.

13.9 Addition/Subtraction Positioning

This function combines the movement amount of 2 axes and transmits.

Relevant FB

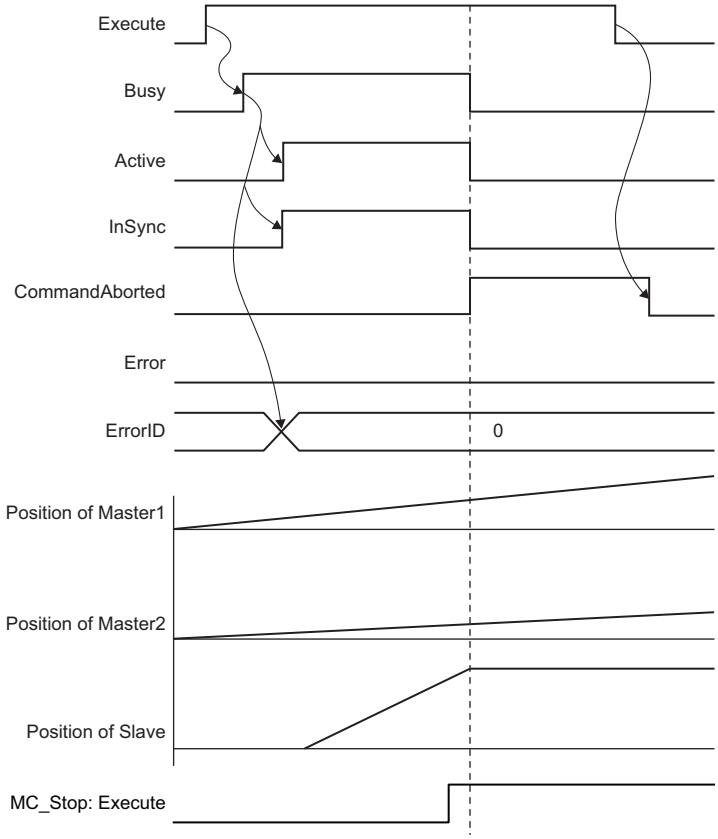
MC_CombineAxes

Item	Description
Function overview	Added or subtracted value which the movement amout of the specified master axis 2-axis is set as the commanded position, and the positioning is executed.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

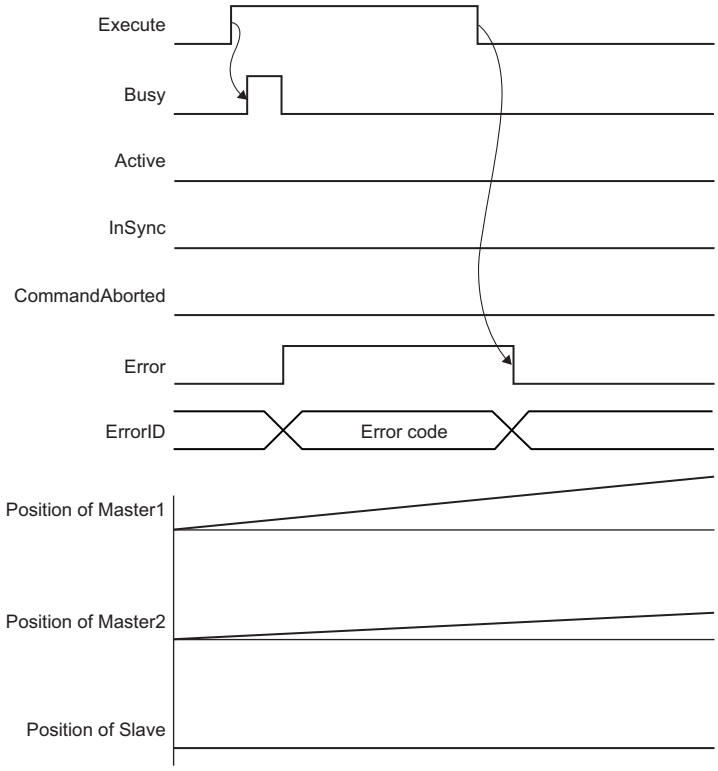
Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
56	8	Subroutine type	Real-time execution

■Timing chart

[When the FB is normally completed]



[When an error occurs]



■I/O variables

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Master axis 1	Master1	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables
Master axis 2	Master2	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables
Slave axis	Slave	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables

■Input variables

Import ↑: At start, R: Retrigger possible, C: Continuous update possible

Name	Variable name	Data type	Import	Setting range	Default value	Description
Execute command	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Continuous update	ContinuousUpdate	BOOL	↑	TRUE, FALSE	FALSE	During TRUE, it can continuously change the addition/subtraction method selection, the master axis 1 gear ratio numerator, the master axis 1 gear ratio denominator, the master axis 2 gear ratio numerator, and the master axis 2 gear ratio denominator.
Addition/ subtraction method selection	CombineMode	MC_COMBINE_MODE	↑/R/C	0, 1	1	☞ Page 470 CombineMode (Addition/ subtraction method selection)
Master axis 1 gear ratio numerator	GearRatioNumeratorM1	DINT	↑/R/C	-2147483648 to 2147483647	1	☞ Page 470 GearRatioNumeratorM1 (Master axis 1 gear ratio numerator)
Master axis 1 gear ratio denominator	GearRatioDenominatorM1	DWORD(UDINT)	↑/R/C	1 to 2147483647	1	☞ Page 470 GearRatioDenominatorM1 (Master axis 1 gear ratio denominator)
Master axis 2 gear ratio numerator	GearRatioNumeratorM2	DINT	↑/R/C	-2147483648 to 2147483647	1	☞ Page 470 GearRatioNumeratorM2 (Master axis 2 gear ratio numerator)
Master axis 2 gear ratio denominator	GearRatioDenominatorM2	DWORD(UDINT)	↑/R/C	1 to 2147483647	1	☞ Page 470 GearRatioDenominatorM2 (Master axis 2 gear ratio denominator)
Master axis 1 data source	MasterValueSourceM1	MC_SOURCE	↑	1, 2, 101, 102	1	☞ Page 471 MasterValueSourceM1 (Master axis 1 data source selection)
Master axis 2 data source	MasterValueSourceM2	MC_SOURCE	↑	1, 2, 101, 102	1	☞ Page 471 MasterValueSourceM2 (Master axis 2 data source selection)
Buffer mode	BufferMode	MC_BUFFER_MODE	↑	0, 1	0	☞ Page 471 BufferMode (Buffer mode)
Options	Options	DWORD(HEX)	↑	00000000H	00000000H	Sets "00000000H". (A value other than "00000000H"causes the error "Out of Options Range" (error code: 1A4EH).)

■Output variables

Name	Variable name	Data type	Default value	Description
In synchronization	InSync	BOOL	FALSE	☞ Page 471 InSync (In synchronization)
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Controlling	Active	BOOL	FALSE	Indicates that the FB is controlling the axis.
Abortion of execution	CommandAborted	BOOL	FALSE	Indicates that another FB has aborted the execution of the FB. When this FB is stopped by an error or when this FB is started up while an error occurs, it turns TRUE. Turns FALSE when Execute turns FALSE.
Error	Error	BOOL	FALSE	When TRUE, it indicates that an error has occurred in the FB.
Error code	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned. For details, refer to the following. ☞ Page 662 List of Error Codes

Control details

In MC_CombineAxes, CombineMode, GearRatioNumeratorM1, GearRatioDenominatorM1, GearRatioNumeratorM2, GearRatioDenominatorM2, MasterValueSourceM1, MasterValueSourceM2 and BufferMode are set and the addition/subtraction positioning is executed.

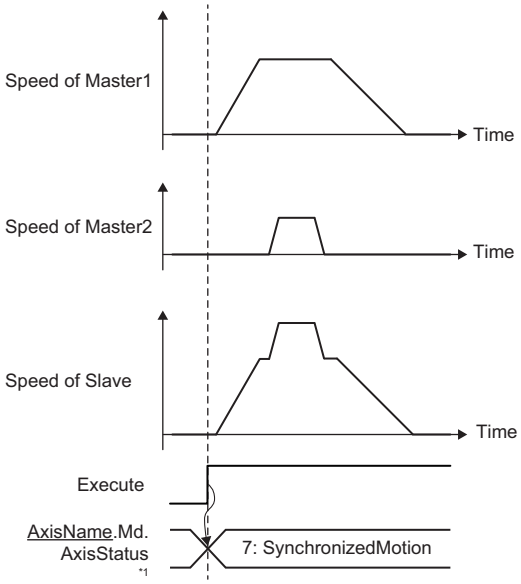
To stop the operation, execute MC_Stop.

The movement amount of Master1 and Master2 are combined. Also, gear ratio can be set to each master axis.

Combining the movement amount calculates adding/subtracting regardless of the unit of Master1 and Master2.

When "0: mcAddAxes" is set to CombineMode

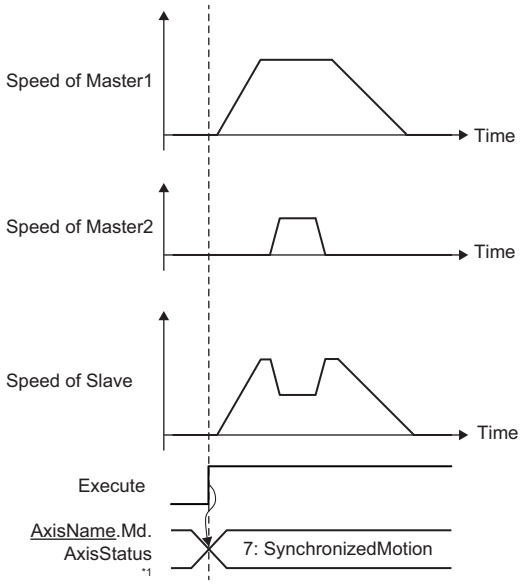
Movement amount of Slave = Master1 + Movement amount of Master2



*1 This is axis state of Slave axis.

When "1: mcSubAxes" is set to CombineMode

Movement amount of Slave = Master1 - Movement amount of Master2



*1 This is axis state of Slave axis.

Details of setting items

■CombineMode (Addition/subtraction method selection)

Set the method to combine the movement amount of Master1 and Master2.

- 0: mcAddAxes

It adds the movement amount of Master1 and Master2 and outputs.

- 1: mcSubAxes

It subtracts the movement amount of Master2 from the movement amount of Master1 and outputs.

■GearRatioNumeratorM1 (Master axis 1 gear ratio numerator)

Set the value of numerator for when converting the movement amount of Master1 and combining.

The movement amount is converted as shown below.

$$\text{Master1 movement amount after conversion} = \text{Master1 movement amount before conversion} \times \frac{\text{GearRatioNumeratorM1}}{\text{GearRatioDenominatorM1}}$$

The movement amount can be reversed and combined by setting a negative value to GearRatioNumeratorM1.

■GearRatioDenominatorM1 (Master axis 1 gear ratio denominator)

Set the value of denominator for when converting the movement amount of Master1 and combining.

It is set with GearRatioNumeratorM1.

Set a positive value to GearRatioDenominatorM1.

■GearRatioNumeratorM2 (Master axis 2 gear ratio numerator)

Set the value of numerator for when converting the movement amount of Master2 and combining.

The movement amount is converted as shown below.

$$\text{Master2 movement amount after conversion} = \text{Master2 movement amount before conversion} \times \frac{\text{GearRatioNumeratorM2}}{\text{GearRatioDenominatorM2}}$$

The movement amount can be reversed and combined by setting a negative value to GearRatioNumeratorM2.

■GearRatioDenominatorM2 (Master axis 2 gear ratio denominator)

Set the value of denominator for when converting the movement amount of Master2 and combining.

It is set with GearRatioNumeratorM2.

Set a positive value to GearRatioDenominatorM2.

■MasterValueSourceM1 (Master axis 1 data source selection)

Select the data source of Master1 which is for the movement amount of Master1 to transmit. For details, refer to the following.

☞ Page 440 Master Axis Data Source Selection

■MasterValueSourceM2 (Master axis 2 data source selection)

Select the data source of Master2 which is for the movement amount of Master2 to transmit. For details, refer to the following.

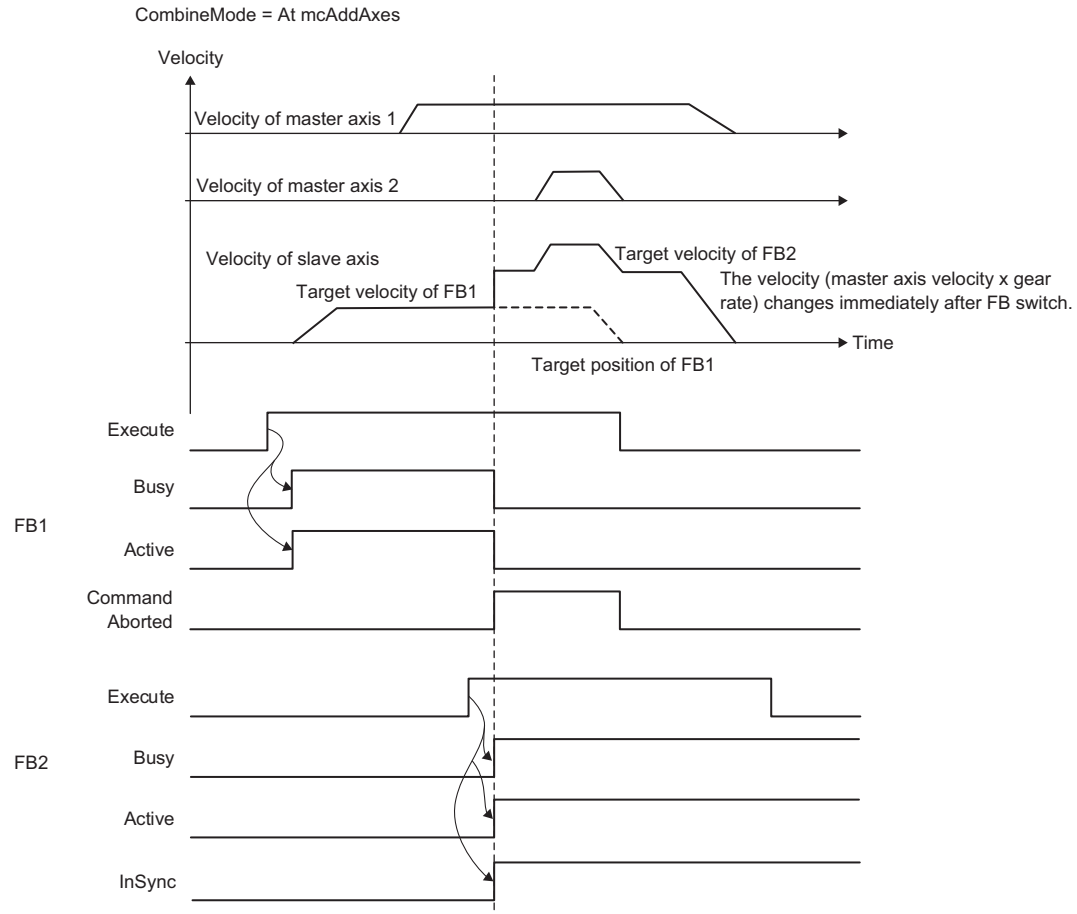
☞ Page 440 Master Axis Data Source Selection

■BufferMode (Buffer mode)

Select the buffer mode. Aborting and Buffered can be set.

When the FB is switched, the speed changes immediately based on the master axis speed and the set gear data as the speed of MC_CombineAxes follows the synchronizing FB (master axis).

An operation example when Aborting is executed to the on-going FB1 with the linked start FB2 (slave axis: MC_CambineAxes) is shown below.



The speed of FB2 occurs when InSync turns TRUE.

For details on operation, refer to the following.

☞ Page 133 Buffer mode type

■InSync (In synchronization)

It turns TRUE when Slave starts synchronizing.

■Busy (Executing)

It turns TRUE during execution of the FB.

■Active (Controlling)

It turns TRUE during controlling Slave.

■CommandAborted (Abortion of execution)

It turns TRUE if other FBs are started during execution of this FB.

■Error (Error)

It turns TRUE if an error occurs in this FB.

■ErrorID (Error code)

It returns an error code occurred in this FB.

Restriction

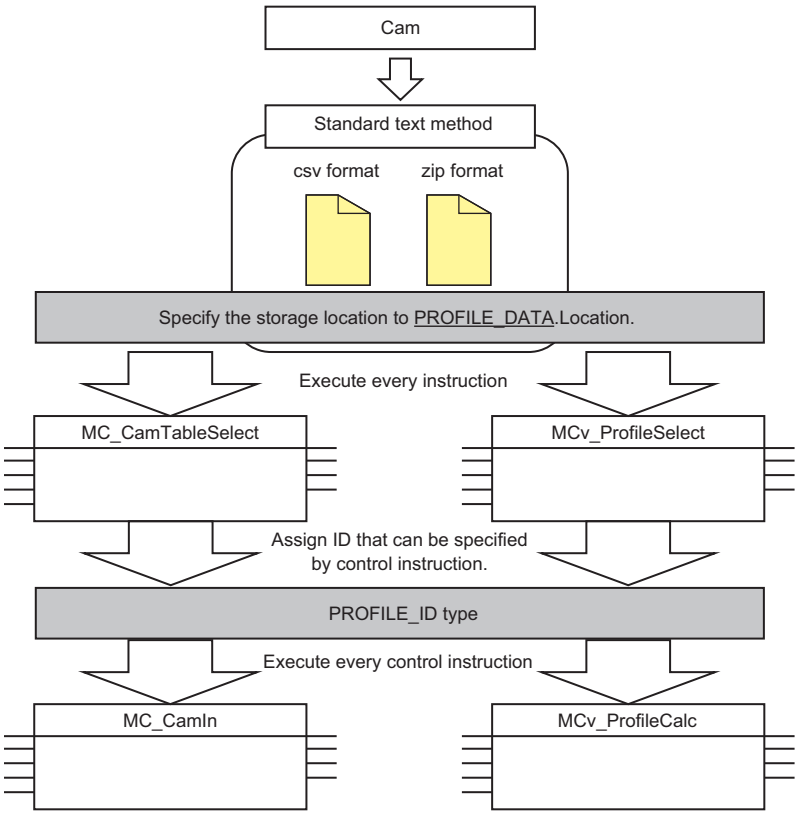
For the error between the master axis and the slave axis at the addition/subtraction positioning

- The error of position between the master axis and the slave axis may increase if the movement amount from the position which InGear of the master axis or the slave axis turns to TRUE exceeds ±10000000000.0. To set the movement range of the master axis and the slave axis within the above stated range, consider the gear ratio and the axis unit setting.
- When the error increases, clear synchronization or change current value specified the absolute position.

14 OPERATION PROFILE FUNCTION

14.1 Operation Profile Data

Waveform data used for control is collectively called operation profile data.
This section describes creating and using methods of operation profile data.
Operation profile data not only can be used in various functions (ex. cam of the single axis synchronous control), but also can perform opening profile for the cam and reading/writing for the cam.



Operation of this function for each system status

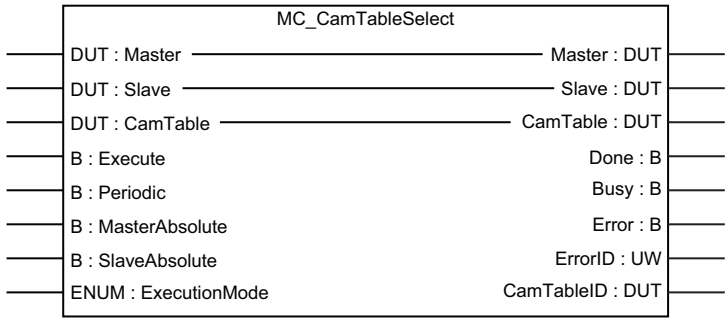
○: Possible, ×: Not possible

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

Relevant variables/FB

Variable/Structure name	Name	Details
PROFILE_DATA		
Location	Calculation profile data storage location	FILE_LOCATION type
ID	Profile ID	PROFILE_ID type
FILE_LOCATION		
FileName	File name	Specifies a file name. (Up to 63 characters including the extension can be set.)
Path	Folder specification	Specifies the folder path in which the file is stored. (Up to 63 characters can be set. Set up to 127 characters (including Null string) in total for FileName and Path.)
PROFILE_ID		
Number	Profile ID No.	Profile ID No. (0 to 60000)
INSTANCE_ID		
Number	Instance ID No.	Instance ID No.

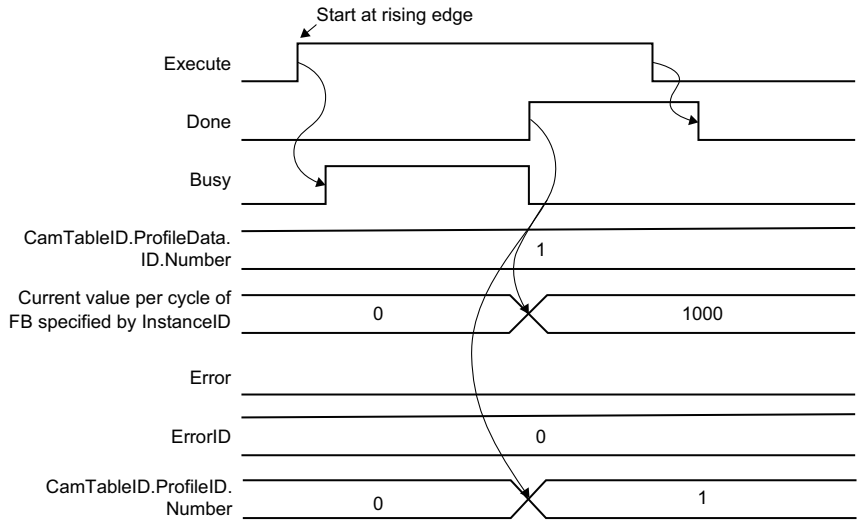
MC_CamTableSelect

Item	Description
Function overview	Stores specified operation profile data in the open area.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

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

■Timing chart

[When the FB is normally completed]



[When an error occurs]

For details when an error occurs, refer to the following.

☞ Page 27 Execute type/Enable type

■I/O variables

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Master axis	Master	AXIS_REF	↑	—	Refer to description	When FB is used on the motion system side, omission is possible. The setting value is ignored. When FB is used on the CPU module side, set Master.StartIO. Master.AxisNo is ignored.
Slave axis	Slave	AXIS_REF	↑	—	Refer to description	When FB is used on the motion system side, omission is possible. The setting value is ignored. When FB is used on the CPU module side, set Slave.StartIO. Slave.AxisNo is ignored.
Cam table	CamTable	MC_CAM_REF	↑	—	Omission not possible	Specifies the operation profile data (cam data).

■Input variables

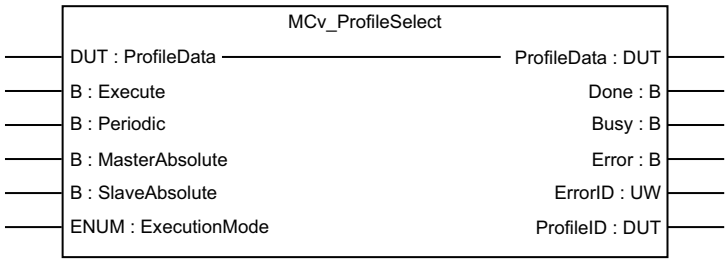
Import ↑: At start

Name	Variable name	Data type	Import	Setting range	Default value	Description
Start	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Periodic	Periodic	BOOL	↑	TRUE, FALSE	FALSE	TRUE: Periodic FALSE: Non periodic
Master axis absolute coordinate	MasterAbsolute	BOOL	↑	TRUE, FALSE	FALSE	TRUE: Absolute coordinate (When it is set, the error "Out of Input Absolute Coordinate Range" (error code: 341DH) occurs.) FALSE: Relative coordinate
Slave axis absolute coordinate	SlaveAbsolute	BOOL	↑	TRUE, FALSE	FALSE	TRUE: Absolute coordinate FALSE: Relative coordinate
Start mode	ExecutionMode	MC_EXECUTION_MODE	↑	0 to 3	0	Specifies the execution timing of the FB. 0: mcImmediately (Execute immediately) 1: mcQueued (Execute at completion) 3: mcSpeculatively (Execute speculatively) For details, refer to the following. ☞ Page 514 Control details

■Output variables

Name	Variable name	Data type	Default value	Description
Completion	Done	BOOL	FALSE	It indicates the control is completed. When Execute is TRUE at the operation completion, it will be TRUE until Execute turns FALSE. When Execute is FALSE at the operation completion, it will be TRUE for only one cycle.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Error	Error	BOOL	FALSE	When TRUE, it indicates that an error has occurred in the FB.
Error code	ErrorID	WORD(UINT)	0	The generated error code is returned. For details, refer to the following. ☞ Page 662 List of Error Codes
Cam table ID	CamTableID	MC_CAM_ID	0	Outputs the profile ID.

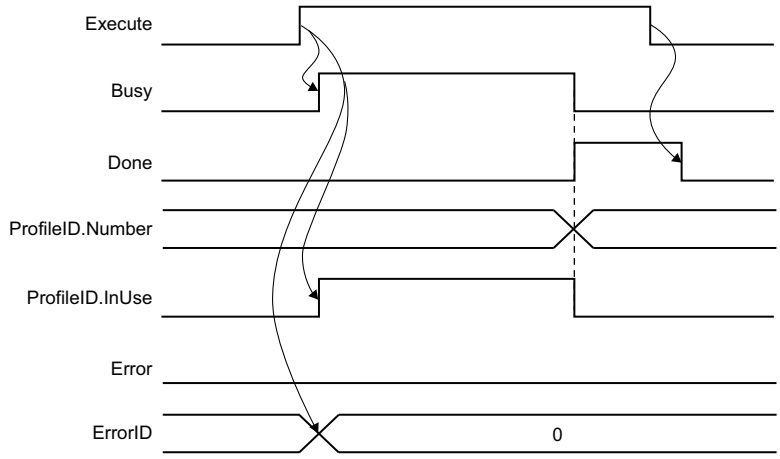
MCv_ProfileSelect (Future support planned)

Item	Description
Function overview	Stores specified operation profile data in the open area.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

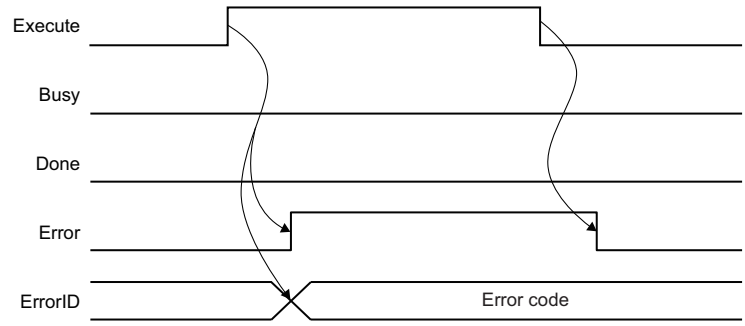
Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
—	—	Subroutine type	Real-time execution

■Timing chart

[When the FB is normally completed]



[When an error occurs]



■I/O variables

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Profile data	ProfileData	PROFILE_DATA	↑	—	Omission not possible	Specifies the operation profile data.

■Input variables

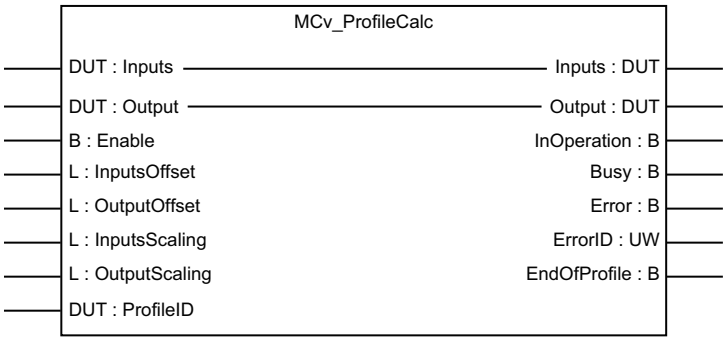
Import ↑: At start

Name	Variable name	Data type	Import	Setting range	Default value	Description
Start	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Periodic	Periodic	BOOL	↑	TRUE, FALSE	FALSE	TRUE: Periodic FALSE: Non periodic
Master axis absolute coordinate	MasterAbsolute	BOOL	↑	TRUE, FALSE	FALSE	TRUE: Absolute coordinate (When it is set, the error "Out of Input Absolute Coordinate Range" (error code: 341DH) occurs.) FALSE: Relative coordinate
Slave axis absolute coordinate	SlaveAbsolute	BOOL	↑	TRUE, FALSE	FALSE	TRUE: Absolute coordinate FALSE: Relative coordinate
Start mode	ExecutionMode	MC_EXECUTION_MODE	↑	0 to 3	0	Specifies the execution timing of the FB. 0: mcImmediately (Execute immediately) 1: mcQueued (Execute at completion) 3: mcSpeculatively (Execute speculatively) For details, refer to the following. ☞ Page 514 Control details

■Output variables

Name	Variable name	Data type	Default value	Description
Completion	Done	BOOL	FALSE	It indicates the control is completed. When Execute is TRUE at the operation completion, it will be TRUE until Execute turns FALSE. When Execute is FALSE at the operation completion, it will be TRUE for only one cycle.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Error	Error	BOOL	FALSE	Indicates that an error has occurred.
Error code	ErrorID	WORD(UINT)	0	The error code generated is returned. For details, refer to the following. ☞ Page 662 List of Error Codes
Profile ID	ProfileID	PROFILE_ID	0	Outputs the profile ID.

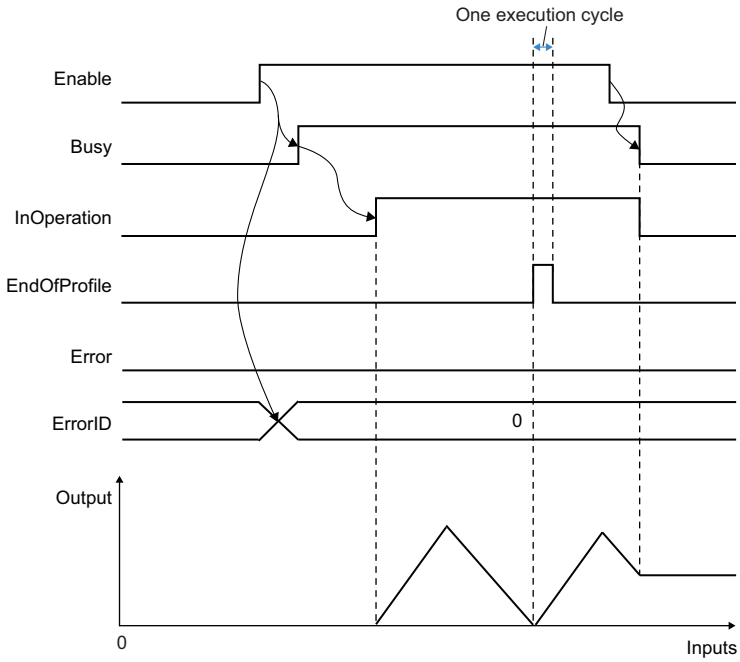
MCv_ProfileCalc (Future support planned)

Item	Description
Function overview	Calculates the ouput data to the input data following the specified operation profile data.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

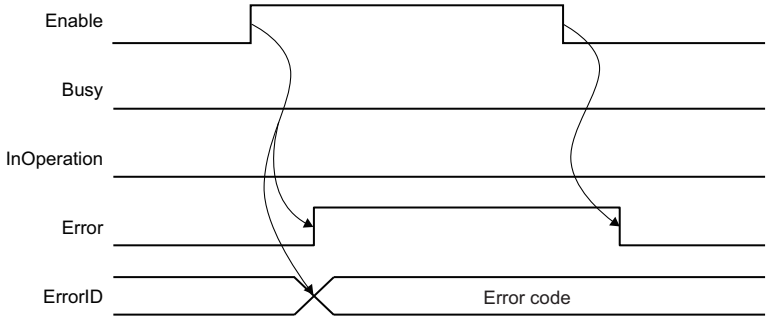
Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
—	—	Subroutine type	Real-time execution

■Timing chart

[When the FB is normally completed]



[When an error occurs]



■I/O variables

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Input data	Inputs	TARGET_REF[0..2]	⎍	—	Omission not possible	Specifies the reference of the input value.
Output data	Output	TARGET_REF	⎍	—	Omission not possible	Specifies the reference of the output value.

■Input variables

Import ⎍: Always

Name	Variable name	Data type	Import	Setting range	Default value	Description
Enable	Enable	BOOL	⎍	TRUE, FALSE	FALSE	During TRUE, it executes the FB.
Input offset	InputsOffset	LREAL[0..2]	⎍	☞ Page 83 Positioning Range	0.0	Shifts the phase of each input by the amount of offset. It can be specified for the number of Inputs elements. (The number exceeding the number of elements is ignored)
Output offset	OutputOffset	LREAL	⎍	☞ Page 83 Positioning Range	0.0	Shifts the phase of each output by the amount of offset.
Input coefficient	InputsScaling	LREAL[0..2]	⎍	0.01 to 10.0	1.0	Widens/narrows the phase of each input. It can be specified for the number of Inputs elements. (The number exceeding the number of elements is ignored)
Output coefficient	OutputScaling	LREAL	⎍	0.01 to 10.0	1.0	Widens/narrows the phase of each output.
Profile ID	ProfileID	PROFILE_ID	⎍	0 to 60000	0	Specifies the profile ID.

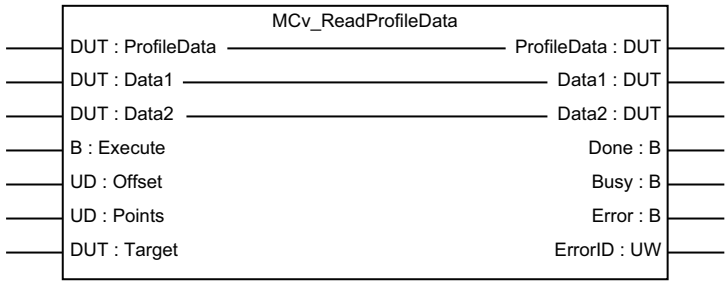
■Output variables

Name	Variable name	Data type	Default value	Description
Execution completion	InOperation	BOOL	FALSE	It indicates the output value is valid (refreshing).
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Error	Error	BOOL	FALSE	Indicates that an error has occurred.
Error code	ErrorID	WORD(UINT)	0	The error code generated is returned. For details, refer to the following. ☞ Page 662 List of Error Codes
Profile cycle completion	EndOfProfile	BOOL	FALSE	When the end edge of the operation profile data is detected, only one execution cycle turns TRUE.

■Public variables

Name	Variable name	Data type	Default value	Description
Current value per cycle	InputsPerCycle	LREAL[0..2]	☞ Page 83 Positioning Range	Variable that the current value per cycle is stored.
Reference value	Reference	LREAL	☞ Page 83 Positioning Range	Variable that the reference value is stored.
Output value	Output	LREAL	☞ Page 83 Positioning Range	Variable that the output value is stored.
Instance ID	InstanceID	WORD(UINT)	0	It is the instance ID. It is automatically set by the system when creating an instance. It is used for inputs of FB, etc.

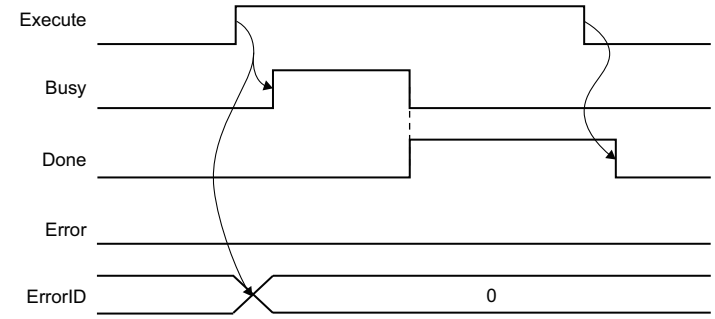
MCv_ReadProfileData

Item	Description
Function overview	Reads the specified operation profile data from the open area or the file.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

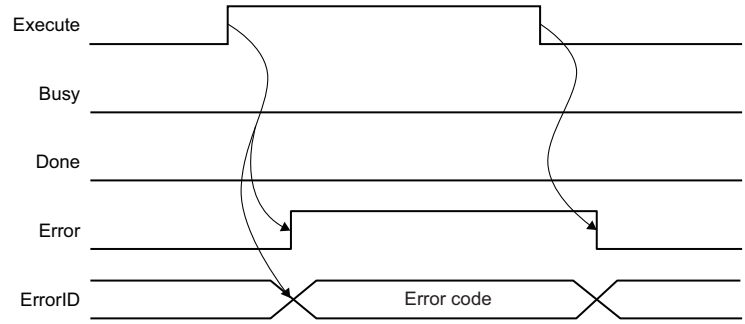
Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
422	10	Subroutine type	Real-time execution

■Timing chart

[When the FB is normally completed]



[When an error occurs]



■I/O variables

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Profile data	ProfileData	PROFILE_DATA	↑	—	Omission not possible	Specifies the operation profile data to read.
Read data 1	Data1	TARGET_REF	↑	—	Omission not possible	Specifies the structure of the data for reading according to the operation profile data format to read.
Read data 2	Data2	TARGET_REF	↑	—	Omission not possible	Specifies the structure of the data for reading according to the operation profile data format to read.

■Input variables

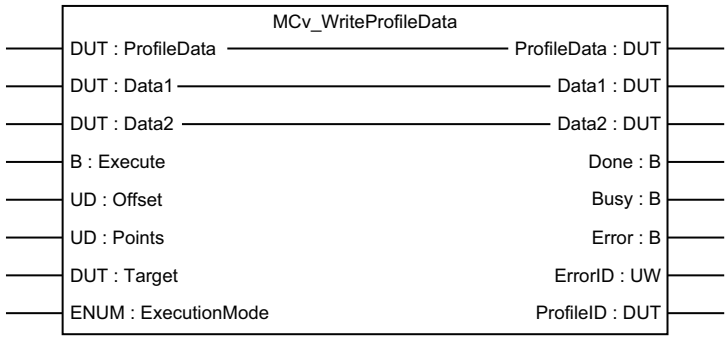
Import ↑: At start

Name	Variable name	Data type	Import	Setting range	Default value	Description
Start	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Offset	Offset	DWORD(UDINT)	↑	—	0	Specifies offset from the top of the operation profile data. To specify a value other than 0 to Offset, also set a value other than 0 to Points. (If a value "0" is set to Points, the error "Number of Offset/Read/Write Data Incorrect" (error code: 3465H) occurs.)
Number of read data	Points	DWORD(UDINT)	↑	—	0	Specifies the number of data points to read. Set 0.
Read destination	Target	WORD(UINT)	↑	0 to 1	0	Specifies the read destination of the operation profile data. 0: OpenArea (Open Area) 1: File (File)

■Output variables

Name	Variable name	Data type	Default value	Description
Completion	Done	BOOL	FALSE	Indicates that the control is completed. When Execute is TRUE at the operation completion, TRUE remains until Execute turns FALSE. When Execute is FALSE at the operation completion, it becomes TRUE only for one cycle.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Error	Error	BOOL	FALSE	Indicates that an error has occurred.
Error code	ErrorID	WORD(UINT)	0	The error code generated is returned. For details, refer to the following. ☞ Page 662 List of Error Codes

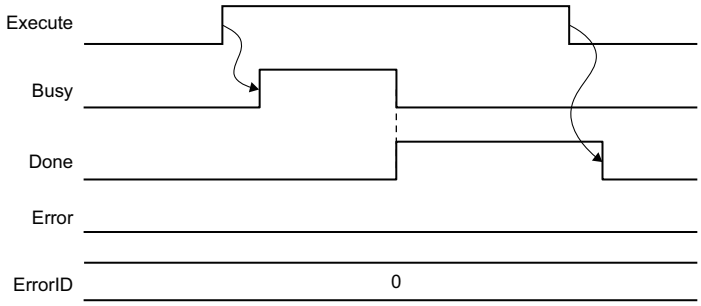
MCv_WriteProfileData

Item	Description
Function overview	Writes the specified operation profile data from the open area or the file.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation
424	16	Subroutine type	Real-time execution

■Timing chart

[When the FB is normally completed]



[When an error occurs]

For details when an error occurs, refer to the following.

☞ Page 27 Execute type/Enable type

■I/O variables

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Profile data	ProfileData	PROFILE_DATA	↑	—	Omission not possible	Specifies the operation profile data to write.
Write data 1	Data1	TARGET_REF	↑	—	Omission not possible	Specifies the structure of the data for writing according to the operation profile data format to write.
Write data 2	Data2	TARGET_REF	↑	—	Omission not possible	Specifies the structure of the data for writing according to the operation profile data format to write.

■Input variables

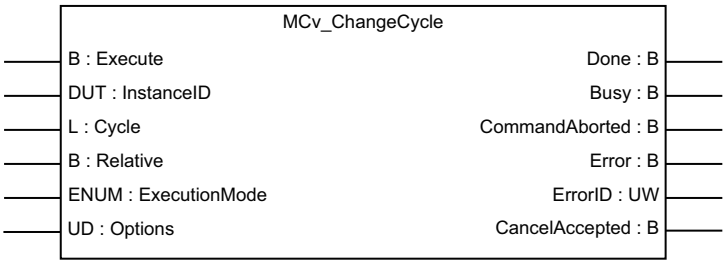
Import ↑: At start

Name	Variable name	Data type	Import	Setting range	Default value	Description
Start	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Offset	Offset	DWORD(UDINT)	↑	—	0	Specifies offset from the top of the operation profile data. To specify a value other than 0 to Offset, also set a value other than 0 to Points. (If a value "0" is set to Points, the error "Number of Offset/Read/Write Data Incorrect" (error code: 3465H) occurs.)
Number of write data	Points	DWORD(UDINT)	↑	—	0	Specifies the number of data points to write.
Write destination	Target	WORD(UINT)	↑	0 to 1	0	Specifies the write destination of the operation profile data. 0: OpenArea (Open Area) 1: File (File) To specify File(file) to the write destination, set Offset and Points 0. (If other than "0" is set, the error "Number of Offset/Read/Write Data Incorrect" (error code: 3465H) occurs.)
Start mode	ExecutionMode	MC_EXECUTION_MODE	↑	0 to 3	0	Specifies the execution timing of the FB. 0: mcImmediately (Execute immediately) 1: mcQueued (Execute at completion) 3: mcSpeculatively (Execute speculatively) For details, refer to the following. ☞ Page 514 Control details

■Output variables

Name	Variable name	Data type	Default value	Description
Completion	Done	BOOL	FALSE	Indicates that the control is completed. When Execute is TRUE at the operation completion, TRUE remains until Execute turns FALSE. When Execute is FALSE at the operation completion, it becomes TRUE only for one cycle.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Error	Error	BOOL	FALSE	Indicates that an error has occurred.
Error code	ErrorID	WORD(UINT)	0	The error code generated is returned. For details, refer to the following. ☞ Page 662 List of Error Codes
Profile ID	ProfileID	PROFILE_ID	0	Outputs profile ID.

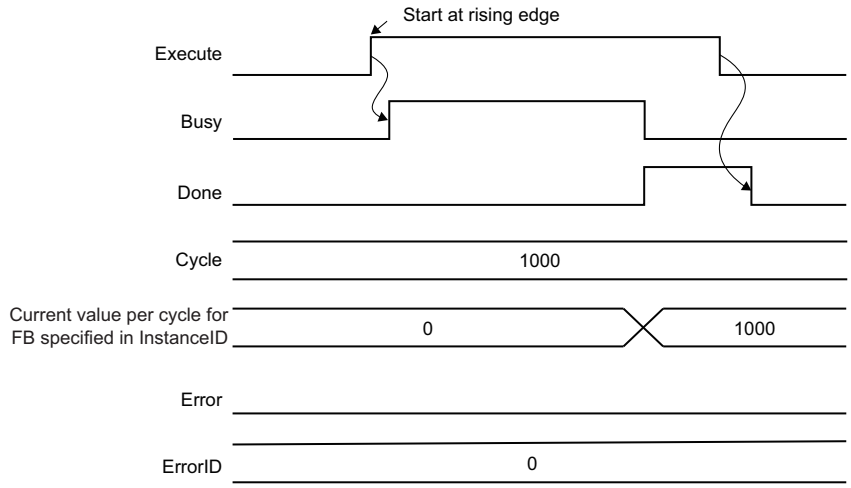
MCv_ChangeCycle

Item	Description
Function overview	Changes the current value per cycle of the specified operation profile data control FB.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

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

■Timing chart

[When the FB is normally completed]



[When an error occurs]

For details when an error occurs, refer to the following.

☞ Page 27 Execute type/Enable type

■Input variables

Import ↑: At start

Name	Variable name	Data type	Import	Setting range	Default value	Description
Start	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Instance ID	InstanceID	INSTANCE_ID	↑	—	0	Specifies the instance ID of FB to change the current value per cycle.
Current value per cycle	Cycle	LREAL[0..2]	↑	—	0.0	Specifies the current value per cycle to change. Specifies the absolute position when Relative = FALSE. Specifies the relative distance when Relative = TRUE.
Relative selection	Relative	BOOL	↑	—	FALSE	TRUE: Relative distance FALSE: Absolute position
Start mode	ExecutionMode	MC_EXECUTION_MODE	↑	0 to 3	0	Specifies the execution timing of the FB. 0: mcImmediately (Execute immediately) 1: mcQueued (Execute at completion) 3: mcSpeculatively (Execute speculatively)
Options	Options	DWORD(HEX)	↑	*1	00000000H	Set the function options by specifying the bit.

*1 For bit and function description, refer to the following.

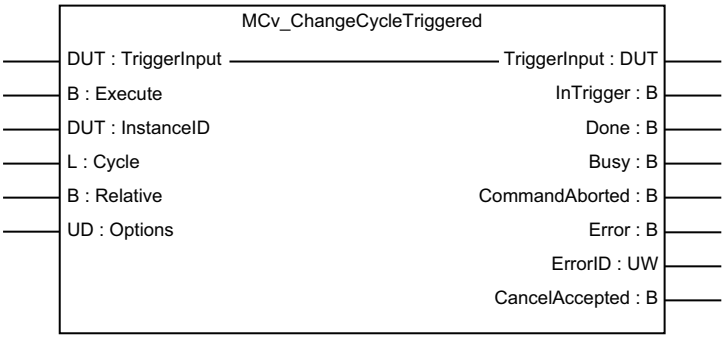
Bit	Function description
0 to 15	Empty (Specify "0".)*1
16	Cancel accepted after FB start Specifies whether cancel is allowed or not after FB is started. 0: Not allowed 1: Allowed
17 to 31	Empty (Specify "0".)*1

*1 If specifying other than "0", the error "Out of Options Range" (error code: 1A4EH) occurs, and the function does not start.

■Output variables

Name	Variable name	Data type	Default value	Description
Completion	Done	BOOL	FALSE	Indicates that the control is completed. When Execute is TRUE at the operation completion, TRUE remains until Execute turns FALSE. When Execute is FALSE at the operation completion, it becomes TRUE only for one cycle.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Abortion of execution	CommandAborted	BOOL	FALSE	Indicates that execution of the FB is interrupted by an error or multiple start, etc.
Error	Error	BOOL	FALSE	Indicates that an error has occurred.
Error code	ErrorID	WORD(UINT)	0	The error code generated is returned. For details, refer to the following. ☞ Page 662 List of Error Codes
Cancel acceptance	CancelAccepted	BOOL	FALSE	Indicates that the FB accepted cancel.

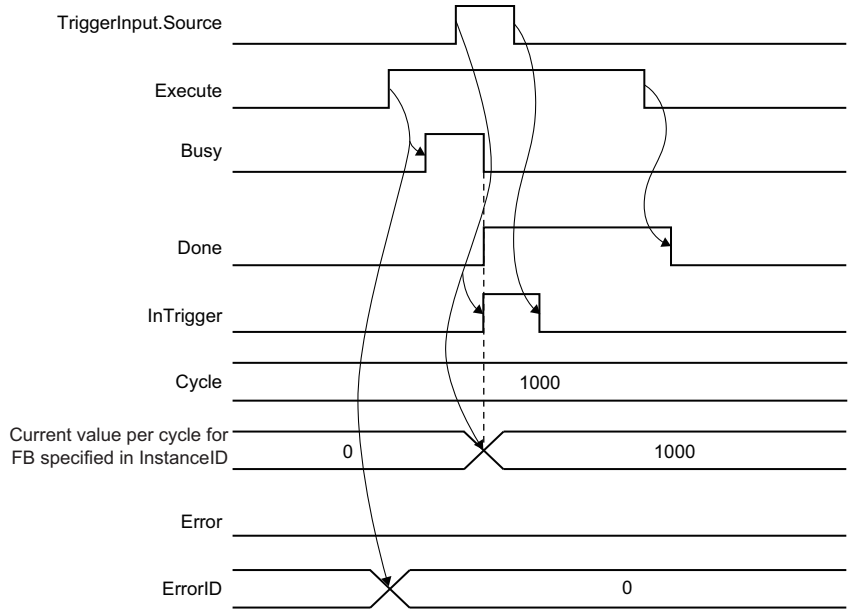
MCv_ChangeCycleTriggered (Future support planned)

Item	Description
Function overview	Changes the current value per cycle of the specified operation profile data control FB to the value which is compensated at the detection timing of the trigger input signal.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

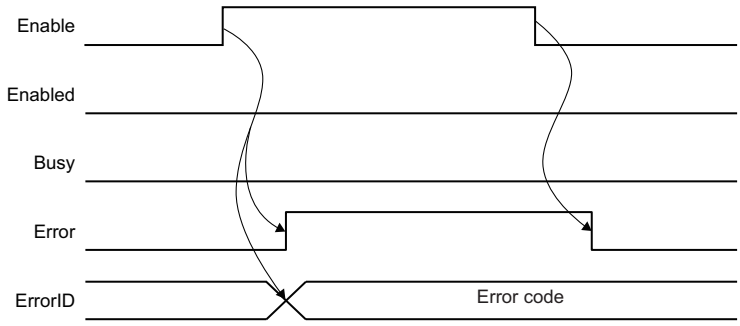
Number of input area points (byte)	Number of output area points (byte)	Compiling method	FB operation type
—	—	Subroutine type	Real-time execution

■Timing chart

[When the FB is normally completed]



[When an error occurs]



■I/O variables

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Trigger input signal	TriggerInput	MC_TRIGGER_REF	↑	—	Omission not possible	Specifies the trigger input signal.

■Input variables

Import ↑: At start

Name	Variable name	Data type	Import	Setting range	Default value	Description
Start	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, it executes the FB.
Instance ID	InstanceID	INSTANCE_ID	↑	—	0	Specifies the instance ID of the FB to change the current value per cycle.
Current value per cycle	Cycle	LREAL[0..2]	↑	—	0.0	Specifies the current value per cycle after change. Specifies the absolute position when Relative = FALSE. Specifies the relative distance when Relative = TRUE.
Relative selection	Relative	BOOL	↑	TRUE, FALSE	FALSE	TRUE: Relative distance FALSE: Absolute position
Options	Options	DWORD(HEX)	↑	*1	00000000H	Set the function options by specifying the bit.

*1 For bit and function description, refer to the following.

Bit	Function description
0 to 15	Empty (Specify "0".)*1
16	Cancel accepted after FB start Specifies whether cancel is allowed or not after FB is started. 0: Not allowed 1: Allowed
17 to 31	Empty (Specify "0".)*1

*1 If specifying other than "0", the error "Out of Options Range" (error code: 1A4EH) occurs, and the function does not start.

■Output variables

Name	Variable name	Data type	Default value	Description
Detecting signal	InTrigger	BOOL	FALSE	Indicates the detection status of the trigger input signal. Outputs the signal detection status as it is regardless of the compensation time that is set to the trigger input signal.
Completion	Done	BOOL	FALSE	Indicates that the control is completed. When Execute is TRUE at the operation completion, TRUE remains until Execute turns FALSE. When Execute is FALSE at the operation completion, it becomes TRUE only for one cycle.
Executing	Busy	BOOL	FALSE	Indicates that the FB is in execution.
Abortion of execution	CommandAborted	BOOL	FALSE	Indicates that execution of the FB is interrupted by an error or multiple start, etc.
Error	Error	BOOL	FALSE	Indicates that an error has occurred.
Error code	ErrorID	WORD(UINT)	0	The error code generated is returned. For details, refer to the following. ☞ Page 662 List of Error Codes
Cancel acceptance	CancelAccepted	BOOL	FALSE	Indicates that the FB accepts cancel.

Creating operation profile data

This section describes storage format and creating method of the operation profile data.

Storage format

The standard text (csv) can be used as the operation profile data in the motion system.

Storage format	Details	Characteristics
Standard text format	Stores the operation profile data as a standard text file (csv file).	The data created in the engineering tool can be stored in this format.

Creating method when not using the engineering tool

When using the operation profile data in the control without using the engineering tool, execute the following procedures.

■When creating the operation profile data by a program

1. Store data (read/write data structure for each type) corresponding to the operation profile data format that will be created on the variables (labels and devices).
2. Set the profile data type (standard/function specific) label as needed. (When storing created operation profile data in the file, set the storage location in Location.)
3. When executing the operation profile data write FB, the control ID can be obtained in the profile ID type (standard/function specific) label.
4. Execute the control FB by using the profile ID type (standard/function specific) label.

■When using the operation profile data created outside

1. Store the operation profile data in an arbitrary place such as a user drive and an SD memory card.
2. Set the storage location in Location of the profile data type (standard/function specific) label.
3. When executing the open FB, the control ID can be obtained in the profile ID type (standard/function specific) label.
4. Execute the control FB by using the profile ID type (standard/function specific) label.

Creating method when using the engineering tool

When using the engineering tool, it can be automatically opened to the motion system when the power turns ON and when the PLC READY [Y0] turns ON by explicitly specifying the profile ID. (When the profile ID is not explicitly specified, it will not be automatically opened.)

Automatically opened operation profile data can be used for control without executing the open FB.

Also, when the operation profile data is created in the engineering tool, the profile data type global label whose label name is the operation profile data name is automatically added, and it can be used in the program. The set description (storage location and ID) is reflected as the initial value of the profile data type.

Point

Profile data type labels created by the engineering tool cannot be edited on the global label editor. Be sure to create and edit it on the operation profile data create window.

The default write destination of the operation profile created by the engineering tool is %PROJECT_ROOT%/calc_profile/, and the storage location is set at PROFILE_DATA.Location.

Files within this folder (including the sub folders) may be overwritten or deleted by the engineering tool operation, so the operation profile created by the program, store in location other than the folder above (including the sub folder).

Operation profile data types

For the function to use operation profile data, there is an operation profile data format that supports the corresponding function. The following table shows operation profile data format and I/O data that can be used to each function.

Function	Operation profile data format	Input data	Output data
Cam	Cam data	Master axis position (selected with MC_SOURCE)	Slave axis position
Profile operation	Multidimensional profile data	Arbitrary data (1 to 3 dimension)	Arbitrary data (1 dimension)

Operation profile data format: Cam data

■Setting item

The relationship of output values to input values is generally defined as the cam data.

A necessary definition as the cam data differs depending on the format. Refer to the following table.

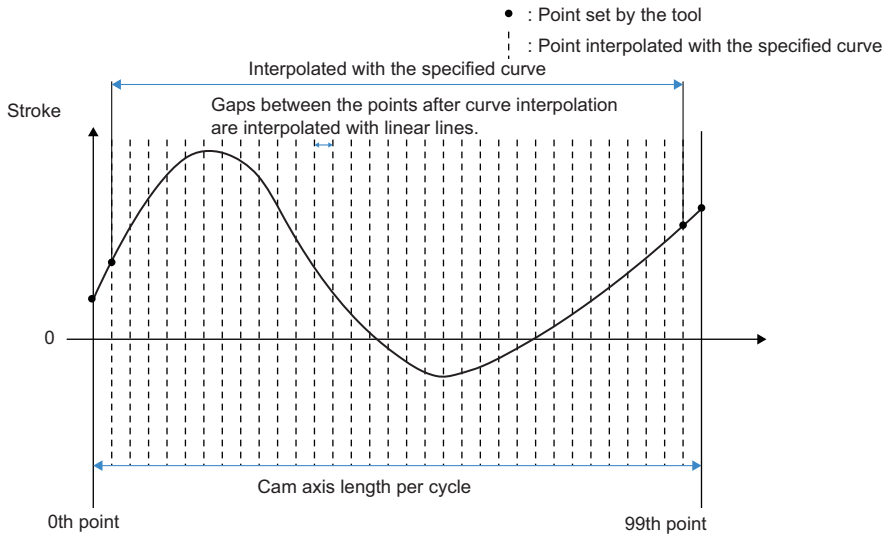
◎: Necessary, ○: Optional, —: Unnecessary

Setting items		Specifying interpolation method		
		0: Linear interpolation	1: Specifying for each section	2: Spline interpolation
Profile ID		○	○	○
Specifying interpolation method		◎	◎	◎
Resolution/coordinate number		◎	◎	◎
Input unit character string		○	○	○
Output unit character string		○	○	○
Start point		—	◎	—
Initial stroke amount		—	◎	—
Initial velocity		—	○	—
Initial acceleration		—	○	—
I/O data		◎	—	—
Parameter	Cam curve types	—	◎	—
	End point	—	◎	◎
	Stroke	—	◎	◎
	Curve applicable range	—	○	—
	Acceleration/ deceleration range compensation	—	○	—
	End point velocity	—	○	—
	End point acceleration	—	○	—
Length per cycle		—	◎	◎
Minimum value per cycle		○	—	—
Maximum value per cycle		○	—	—
Time per cycle		—	○	—
Stroke amount		○	○	○

• Specifying interpolation method

[Specifying for each section]

When "1: Specifying for each section" is set for the specifying interpolation method of the cam data format, gaps between the points are interpolated with the specified curve, and the stroke data that is defined after equally dividing a cam curve per cycle by points of the cam resolution is opened in the open area.



Point

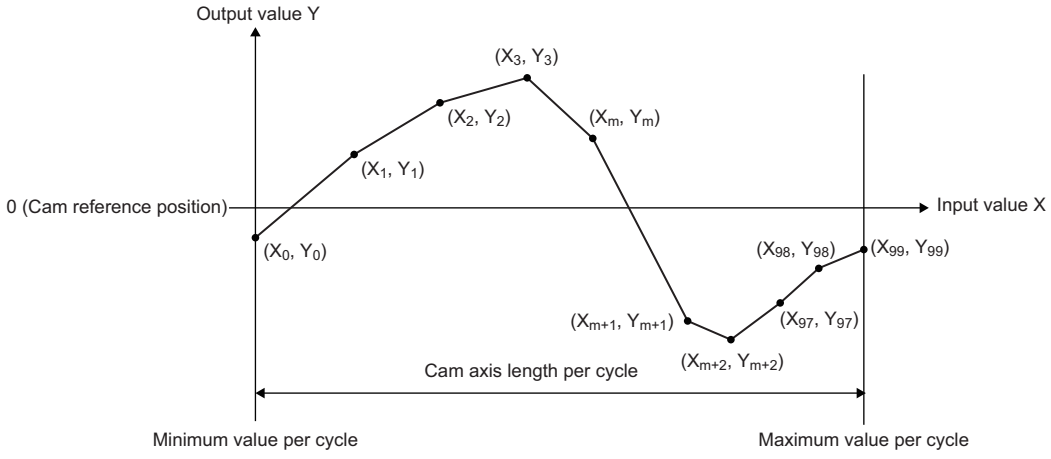
If the number of I/O data is not enough for the set resolution, cam data that is equally divided after interpolating the data in shortage according to the line set by the curve designation will be created.

[Spline interpolation]

When "2: Spline interpolation" is set for the specifying interpolation method of the cam data format, gaps between the points are spline interpolated, and the stroke data that is defined after equally dividing a cam curve per cycle by points of the cam resolution is opened in the open area.

[Linear interpolation]

When "0: Linear interpolation" is set for the specifying interpolation method of the cam data format, the data that is defined for a cam curve per cycle with two or more points is opened in the open area.



• Profile ID

It is a unique ID assigned by an open instruction. The ID opened by the MC_CamTableSelect can be used in the control instruction (MC_CamIn) as the MC_CAM_ID type. ID can be specified with an arbitrary No.

• Resolution/coordinate number

Set the number of divisions for one cam curve. Set the number of data including the 0th point.

• Input/output unit character string

Set a unit of the input/output data with an arbitrary character string. It is handled as a [pulse] when it is omitted. Also, when "%" is specified in the output unit character string, the control that the cam stroke amount is 100% is carried out.

Point

When a unit other than "%" is specified, it is not be used in the control. It is used only in monitoring.

• Start point

Set the start point.

Set "0". If other than "0" is set, the error "Calculation Profile Incorrect" (error code: 3410H) occurs.

• Initial stroke amount

Set the stroke amount for the start point. When the output unit character string is "%", it will be a value corresponding to the 100% of stroke ratio.

• Initial velocity

Set the velocity at the start point.

• Initial acceleration

Set the acceleration at the start point.

• Length per cycle

Set the input amount that is required for one cycle.

• Maximum value per cycle/Minimum value per cycle

Set the maximum/minimum value per cycle. It is valid only when the specifying interpolation method is "0: linear interpolation".

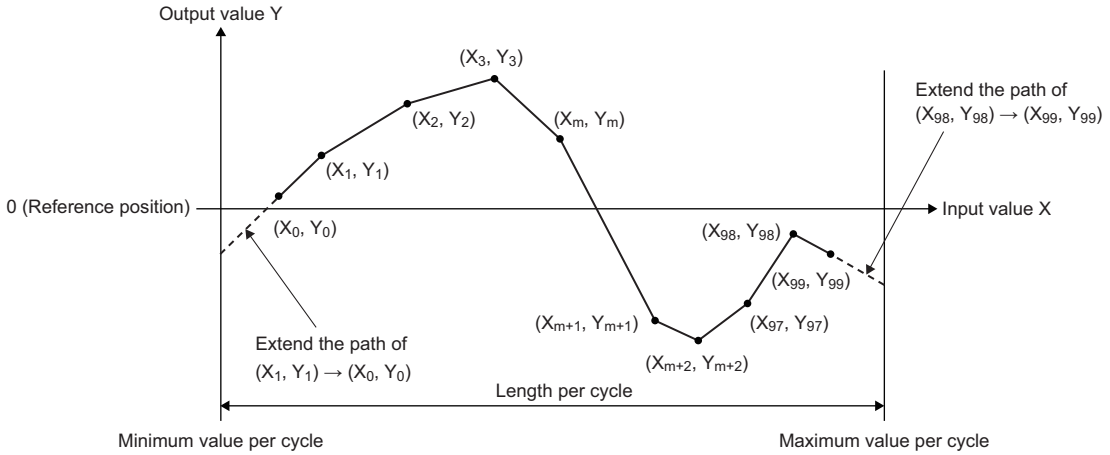
When specifying interpolation method is "0: linear interpolation"

Set "0" to minimum value per cycle. If other than "0" is set to minimum value per cycle, the error "Calculation Profile Incorrect" (error code: 3410H) occurs.

Coordinate data	Description
"Minimum value per cycle < Minimum value of input" "Maximum value of input < Maximum value per cycle"	The control is performed with a line segment created from the coordinates of the nearest two points.
"Minimum value of input < Minimum value per cycle" "Maximum value per cycle < Maximum value of input"	Ignores the coordinate data of outside range of minimum value per cycle to maximum value per cycle

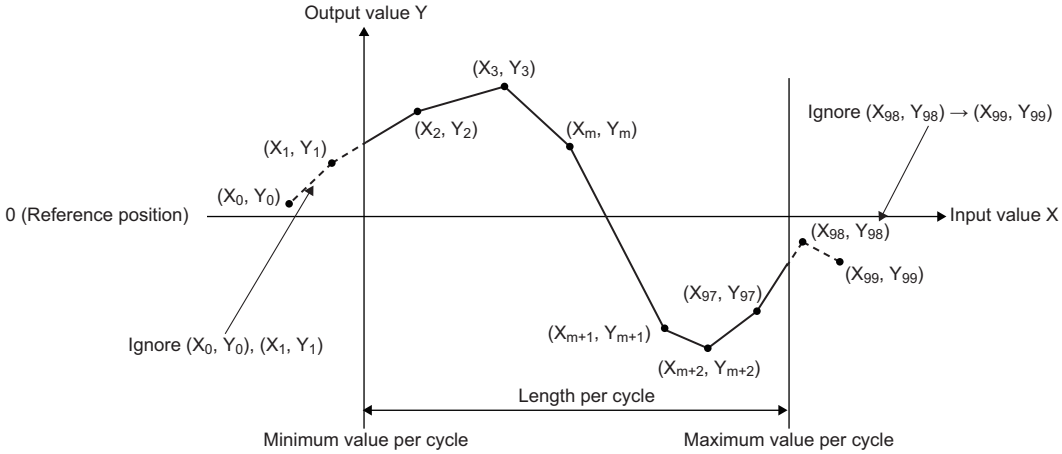
Ex.

For "Minimum value per cycle < Minimum value of input" and "Maximum value of input < Maximum value per cycle", and when resolution is "100"



Ex.

For "Minimum value of input < Minimum value per cycle" or "Maximum value per cycle < Maximum value of input", and when the number of point data is "100"



• Time per cycle

Set time that is required for one cycle.

• Stroke amount

Set the stroke amount. When the output unit character string is "%", it will be a value corresponding to the 100% of stroke ratio.

• I/O data

Set the I/O data. The input data should be set in ascending order.

• End point

Set the end point of each section. End point should be set in ascending order.

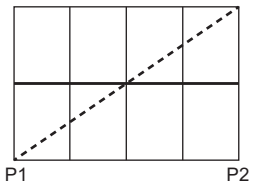
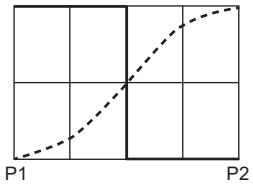
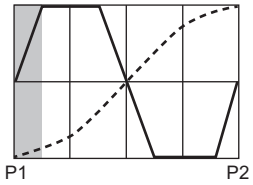
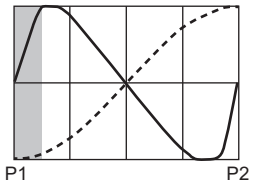
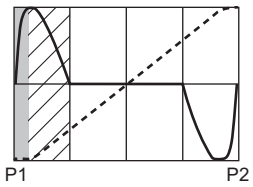
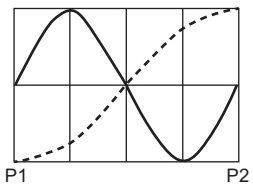
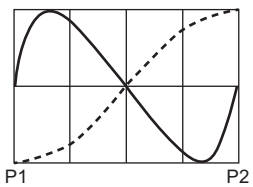
• Stroke

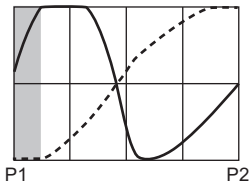
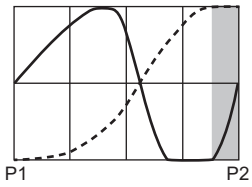
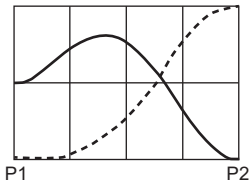
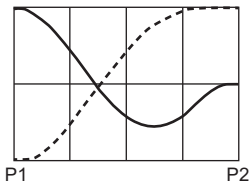
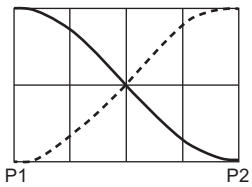
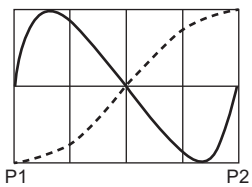
Set the stroke of each section.

• Cam curve types

A cam curve can be selected from the following curves.

- - - - : Stroke ratio ——— : Acceleration : Range L1 : Range L2

Cam curve type (MC_CAM_CURVE_TYPE)			Acceleration curve shape	Curve applicable range (P1 to P2)	Acceleration/ deceleration range compensation ( ): Default value		End point velocity	End point acceleration
Setting value	Cam curve name				Range L1	Range L2		
ConstantSpeed(0)	Constant speed	Discontinuous		0.0 to 1.0	—	—	—	—
ConstantAcceleration(1)	Constant acceleration			0.0 to 1.0	—	—	—	—
DistortedTrapezoid(2)	Distorted trapezoid	Two-dwelling symmetrical		0.0 to 1.0	0.0001 < L1 < 0.2500 (0.125)	—	—	—
DistortedSine(3)	Distorted sine			0.0 to 1.0	0.0001 < L1 < 0.5000 (0.125)	—	—	—
DistortedConstantSpeed(4)	Distorted constant speed			0.0 to 1.0	0.0001 < L1 < 0.1250 (0.0625)	0.0001 < L1 < 0.5000 (0.25)	—	—
Cycloid(5)	Cycloid			0.0 to 1.0	—	—	—	—
FifthCurve(6)	5th Curve			0.0 to 1.0	—	—	—	—

Cam curve type (MC_CAM_CURVE_TYPE)			Acceleration curve shape	Curve applicable range (P1 to P2)	Acceleration/ deceleration range compensation ( ): Default value		End point velocity	End point acceleration
Setting value	Cam curve name				Range L1	Range L2		
Trapecloid(7)	Trapecloid	Two-dwelling asymmetrical		0.0 to 1.0	0.0001 < L1 < 0.2500 (0.125)	—	—	—
Reverse Trapecloid(8)	Reverse trapecloid			0.0 to 1.0	0.0001 < L1 < 0.2500 (0.125)	—	—	—
DoubleHypotenuse(9)	Double hypotenuse	One-dwelling		0.0 to 1.0	—	—	—	—
Reverse DoubleHypotenuse(10)	Reverse double hypotenuse			0.0 to 1.0	—	—	—	—
Single Hypotenuse(11)	Single hypotenuse	Non-dwelling curve		0.0 to 1.0	—	—	—	—
FifthCurve_SpeedDesignation(12)	5th Curve (Adj.)	Two-dwelling symmetrical		0.0 to 1.0	—	—	-2500000000 to 2500000000	-2147483647 to 2147483647

■Read/write data structure

Specify the following structures for [Data1, Data2] of MCv_ReadProfileData/MCv_WriteProfileData.

- Data1

Specify the PROFILE_CAM_DATA type structure.

PROFILE_CAM_DATA			Device offset	Description
Interpolate	Specifying interpolation method	INT	+0	Set the specifying interpolation method of the cam data. 0: Linear interpolation 1: Specifying for each section 2: Spline interpolation
Resolution	Resolution/number of coordinates	DWORD(UDINT)	+2	Set the resolution. Linear interpolation/specifying for each section/spline interpolation 2 to 65535
InputUnitString	Input unit character string	WSTRING[0..31]	+4	—
OutputUnitString	Output unit character string	WSTRING[0..31]	+36	—
StartPoint	Start point*1	LREAL	+68	Set the start point. (Set "0".)
StartStroke	Initial stroke amount*1	LREAL	+72	Set the stroke amount for the start point. (0 ≤ StartStroke < The positive positioning range)
StartVelocity	Initial velocity*3*4	LREAL	76	Set the velocity for the start point. (-2500000000 < StarVelocity < 2500000000)
StartAcceleration	Initial acceleration*3*4	LREAL	80	Set the acceleration for the start point. (-2147483647 < StartAcceleration < 2147483647)
CycleLength	Length per cycle*1	LREAL	+84	Set the input amount that is required for one cycle. (0 < CycleLength < The positive positioning range)
CycleMin	Minimum value per cycle	LREAL	+88	Set the minimum value per cycle. (Set "0".)
CycleMax	Maximum value per cycle	LREAL	+92	Set the maximum value per cycle. (CycleMin < CycleMax < The positive positioning range)
CycleTime	Time per cycle*3	LREAL	+96	Set time per cycle. (0 < CycleTime < 100000.0)
Stroke	Stroke amount	LREAL	+100	Set the stroke amount. (0 < Stroke < The positive positioning range)
NumberOfSections	Number of sections*1	DWORD(UDINT)	+104	Specifying for each section 1 to 360 Spline interpolation 3 to 360
Options	Options*2	DWORD(HEX)	+106	—
			Total size: 108 points	

*1 It is not used in the cam data format (Specifying interpolation method is "0: Linear interpolation"). (It is ignored.)
*2 Specify "0".
*3 Used only when cam curve is 5th Curve (Adj.). (Ignore for other curve)
*4 Set in seconds when the output unit character string is [pulse] and minutes when the output unit character string is other than [pulse].

Ex.

Setting example of TARGET_REF.Target

[DEV](PROFILE_CAM_DATA)G11500000

[VAR]CamData1 (CamData1 is PROFILE_CAM_DATA type instance)

• Data2

For MCv_ReadProfileData, a structure specified in Data2 differs depending on the Target. For MCv_WriteProfileData, it can be executed despite the Target.

If the number of array elements does not match the resolution/coordinate number, the error "Number of Offset/Read/Write Data Incorrect" (error code: 3465H) occurs.

○: Supported, —: Not supported

FB	Target	Structure specified in Data2	Specifying interpolation method		
			Linear interpolation	Specifying for each section	Spline interpolation
MCv_ReadProfileData	0: OpenArea	LREAL[]	○	○*2	○*2
	1: File	LREAL[]	○	—	—
		PROFILE_CAM_ELEMENT[]	—	○	○*1
MCv_WriteProfileData	0: OpenArea	LREAL[]	○	—	—
		PROFILE_CAM_ELEMENT[]	—	○	○*1
	1: File	LREAL[]	○	—	—
		PROFILE_CAM_ELEMENT[]	—	○	○*1

*1 Used only by end point and Stroke

*2 The data of resolution + 1 point (a pair of current value per cycle and stroke) is read/written in the linear interpolation.
In this case, 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.

[LREAL type two-dimensional array]

LREAL[m..n, o..p] Coordinate value[x, y]

The number of elements (n - m + 1), is required as many as the Points arguments of FB. When reading or writing all of the operation profile data (Points = 0), set the same number as PROFILE_CAM_DATA.Resolution.

Set the number of dimensions (p - o + 1) to 2.

To read/write 2 points of LREAL type data equivalent to the number of elements, prepare the device/label areas in same size.

Ex.

When setting the two-dimensional device/label of 100 resolution/coordinate number to TARGET_REF.Target

[Buffer memory specification]

TARGET_REF.Target := "[DEV](LREAL[1..100,1..2])G1150000"

[Label specification]

TARGET_REF.Target := "[VAR]CamData2"

The above "CamData2" declares the label of label name "CamData2" and data type LREAL[1..100,1..2] to the global label, and specifies a character string to Target.

Point

When executing MCv_WriteProfileData whose Target is File in the LREAL type array, the interpolation designation method of the cam data format is output to the file as "0: linear interpolation".

[PROFILE_CAM_ELEMENT type structure array]

PROFILE_CAM_ELEMENT[m..n]*1			Device offset	Description
CurveType	Cam curve type	MC_CAM_CURVE_TYPE	+0	Set the cam curve. (When EndPoint is 0, it is invalid.)
EndPoint	End point*2	LREAL	+4	Set the position to the length per cycle (the current value per cycle). (0 < EndPoint ≤ Length per cycle)
Stroke	Stroke	LREAL	+8	Set the stroke position. (The absolute value of Stroke ≤ Stroke amount)
RangeP1	Curve applicable range (P1)	LREAL	+12	Set the curve applicable range (start point: P1, end point: P2) for the cam curve. (0.0 to 1.0) Set within the range of "P1 < P2". However, for "P1 = P2 = 0", "P1 = 0" and "P2 = 1" are applied.
RangeP2	Curve applicable range (P2)	LREAL	+16	
RangeL1	Acceleration/ deceleration range compensation (Range L1)	LREAL	+20	Set the acceleration/deceleration range (L1, L2) of the cam curve. (0.0001 < L1, L2 < 1.0000) The range that can be set differs depending on the cam curve. For "L1 = L2 = 0.0000", the default value for each cam curve is applied. On the curve that does not use L1 or L2, the setting value is ignored.
RangeL2	Acceleration/ deceleration range compensation (Range L2)	LREAL	+24	
EndVelocity	End point velocity*3	LREAL	+28	Set the end point velocity of the cam curve. (-2500000000 < EndVelocity < 2500000000)
EndAcceleration	End point acceleration*3	LREAL	+32	Set the end point acceleration of the cam curve. (2147483647 < EndAcceleration < 2147483647)
			Total size: 36 points	

*1 The number of elements (n - m + 1), is required as many as the Points arguments of FB. When reading or writing all of the operation profile data (Points = 0), set the same number as PROFILE_CAM_DATA.NumberOfSections.

*2 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.

*3 Setting items used for read/write differ depending on interpolation method specification of the operation profile data. For details, refer to the following.

☞ Page 489 Setting item

Ex.

Setting example of TARGET_REF.Target

[DEV](PROFILE_CAM_ELEMENT [1..10])G11500000

[VAR]CamData2 (CamData2 is dPROFILE_CAM_ELEMENT[1..10] type instance.)

Operation profile data format: Cam data for rotary cutter

■Setting item

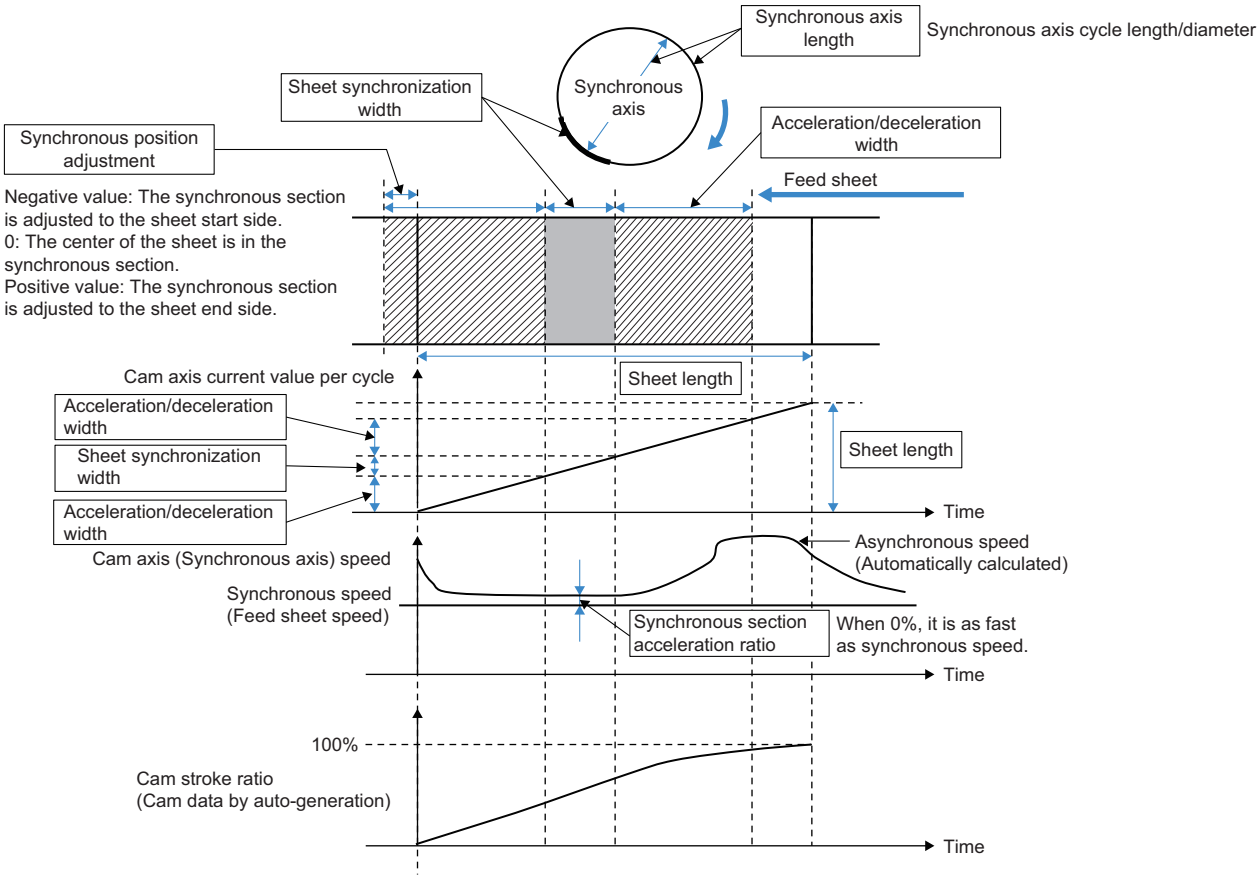
Generate cam data for the rotary cutter. The stroke data that is defined after equally dividing a cam pattern per cycle for rotary cutter by points of the resolution is opened in the open area.

When using this operation profile data format, install add-on PackagingApp.

For a necessary definition as cam data for the rotary cutter, refer to the following table.

◎: Necessary, ○: Optional, —: Unnecessary

Setting item		Cam data for rotary cutter
Profile ID		○
Interpolation method specification		—
Resolution/coordinate number		◎
Input unit character string		○
Output unit character string		—
Start point		—
Initial stroke amount		—
Initial velocity		—
Initial acceleration		—
Parameter	Auto-generation option	◎
	Synchronous section acceleration ratio	◎
	Sheet length	◎
	Sheet synchronization width	◎
	Synchronous axis length	◎
	Synchronous position adjustment	◎
	Acceleration/ deceleration width	◎
	Number of cutter	◎
Length per cycle		—
Minimum value per cycle		—
Maximum value per cycle		—
Time per cycle		—
Stroke amount		— (Fixed with 360)


Synchronous axis length
Synchronous axis cycle length/diameter
Sheet synchronization width
Synchronous axis
Acceleration/deceleration width
Synchronous position adjustment
Feed sheet
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.
Sheet length
Cam axis current value per cycle
Acceleration/deceleration width
Sheet synchronization width
Acceleration/deceleration width
Sheet length
Time
Cam axis (Synchronous axis) speed
Asynchronous speed (Automatically calculated)
Synchronous speed (Feed sheet speed)
Synchronous section acceleration ratio
When 0%, it is as fast as synchronous speed.
Time
100%
Cam stroke ratio (Cam data by auto-generation)
Time

■Read/write data structure

Specify the following structures for Data1 of MCv_ReadProfileData/MCv_WriteProfileData. However, when reading the cam data opened by using the cam data for the rotary cutter from the open area again, it becomes the cam data format. Therefore, use a read/write data structure of the cam data format. For details, refer to the following.

☞ Page 489 Setting item

■Cam for rotary cutter (Specifying a double-precision real number)

• Data1

Specify the PROFILE_ROTARY_CUTTER type structure.

PROFILE_ROTARY_CUTTER			Device offset	Description
Resolution	Resolution	DWORD(UDINT)	+0	Set the resolution for generating the cam. (2 to 32768)
InputUnitString	Input unit character string	WSTRING[0..31]	+2	—
Options	Options	DWORD(HEX)	+34	Specify "0".
AutoGeneration Options	Auto-generation options	WORD(HEX)	+36	• bit0: Acceleration/deceleration method 0: Trapezoidal acceleration/deceleration 1: S-curve acceleration/deceleration • bit1: Synchronous axis length setting 0: Diameter 1: Cycle length • Set "0" for bit2 to 15.
NumberOfCutter	Number of cutter	WORD(UINT)	+37	Set the number of cutter. (1 to 256)
SyncSectionAcc Ratio	Synchronous section acceleration ratio	LREAL	+40	• Set when the synchronous speed in the synchronous section needs to be adjusted. • The speed is "Synchronous speed × (100% + Acceleration ratio)" in the synchronous section. (-50.0 to 50.0)
SheetLength	Sheet length	LREAL	+44	Set the sheet length. (0 < SheetLength < The positive positioning range)
SheetSyncWidth	Sheet synchronization width	LREAL	+48	• Set 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 < SheetSyncWidth < SheetLength) (0 < SheetSyncWidth < Cycle length /NumberOfCutter)
SyncAxisLength	Synchronous axis length	LREAL	+52	• Set the rotary cutter axis length. • When the auto-generation option is set to the diameter, it is calculated as "Cycle length = setting value × π ". • When the auto-generation option is set to the cycle length, it is calculated as "Cycle length = setting value". (0 < Cycle length < The positive positioning range)
SyncPositionAdjustment	Synchronous position adjustment	LREAL	+56	• Set 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. (The absolute value of SyncPositionAdjustment < SheetLength / 2)
AccDecWidth	Acceleration/ deceleration width	LREAL	+60	• Set 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 × AccDecWidth ≤ SheetLength - SheetSyncWidth)
AsyncSpd	Asynchronous speed result	LREAL	+64	When the auto-generation is successfully completed, the asynchronous speed is stored as the ratio of the synchronous speed.
			Total size: 68 points	

• Data2

Specifying is not required. (It is ignored.)

Multidimensional profile data

■Setting item

Multidimensional profile data whose number of input dimensions is two to three can be used.

When using multidimensional profile data, create a csv file according to the operation profile data format, open with the MCv_ProfileSelect, and use the MCv_ProfileCalc.

For a necessary definition as the multidimensional profile data, refer to the following table.

◎: Necessary, ○: Optional, —: Unnecessary

Setting item	Multidimensional profile
Number of input dimensions	◎
Number of output dimensions	—
Profile ID	○
Resolution/coordinate number	◎
Input unit character string	○
Output unit character string	○
I/O data	◎
Length per cycle	◎
Minimum value per cycle	—
Maximum value per cycle	—
Stroke amount	◎
Curve designation	—

• Number of input/output dimensions

Specify the number of input/output dimensions. From two to three can be set for input, and one is fixed for output.

Specify "300: Multidimensional profile format (2 inputs 1 output)" or "310: Multidimensional profile format (3 inputs 1 output)" in the operation profile data format of csv file third line depending on the number of input dimensions.

• Profile ID

It is a unique ID assigned by an open instruction. The ID opened by the MC_CamTableSelect can be used in the control instruction (MC_CamIn) as the MC_CAM_ID type. ID can be specified with an arbitrary No.

• Resolution/coordinate number

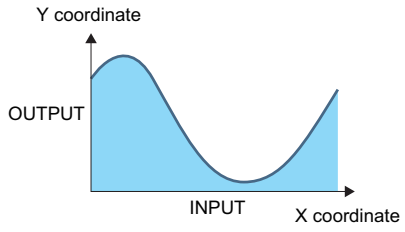
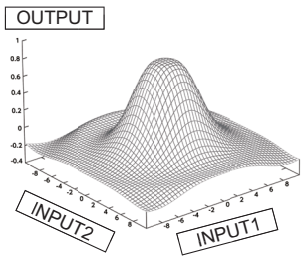
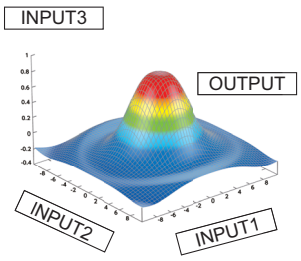
Specify the resolution/coordinate number of the number of input/output dimensions.

• Input/Output unit character string

It can be specified the number of input/output dimensions.

• I/O data

Define the output value to all combinations of the input value.

One-dimensional input — One-dimensional output	Two-dimensional input — One-dimensional output	Three-dimensional input — One-dimensional output
		
X coordinate input →Y coordinate output	X, Y coordinate input →Z coordinate output	X, Y, Z coordinate input → Output

For the operation profile data of one input and one output, define it in the cam data format.

• Stroke amount

Set the stroke amount. When the output unit character string is "%", it will be a value corresponding to the 100% of stroke ratio.

• The minimum/maximum value per cycle

The setting is not required.

• Length per cycle

Set the input amount that is required for one cycle.

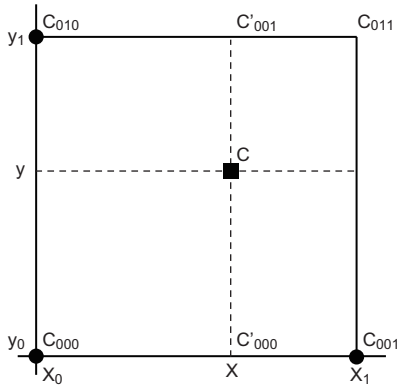
• Curve designation

It is fixed with the linear interpolation.

For the multidimensional profile data, the output value to the input value is calculated by repeating the linear interpolation (linear).

The following figure shows the case of the two-dimensional input.

$C'_0 = \text{linear}(x, x_0, x_1, C_{000}, C_{001})$
$C'_1 = \text{linear}(x, x_0, x_1, C_{010}, C_{011})$
$C = \text{linear}(y, y_0, y_1, C'_0, C'_1)$

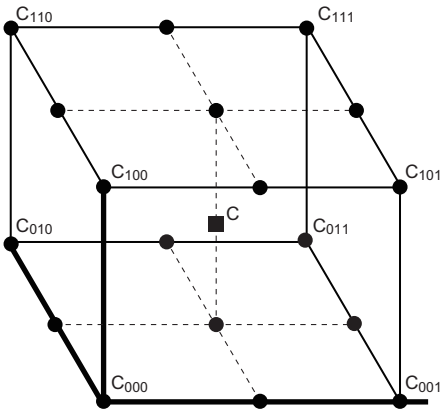


The following figure shows the case of the three-dimensional input.

$C'_{00} = \text{linear}(x, x_0, x_1, C_{000}, C_{001})$
$C'_{01} = \text{linear}(x, x_0, x_1, C_{010}, C_{011})$
$C'_0 = \text{linear}(y, y_0, y_1, C'_{00}, C'_{01})$

$C'_{10} = \text{linear}(x, x_0, x_1, C_{100}, C_{101})$
$C'_{11} = \text{linear}(x, x_0, x_1, C_{110}, C_{111})$
$C'_1 = \text{linear}(y, y_0, y_1, C'_{10}, C'_{11})$

$C = \text{linear}(z, z_0, z_1, C'_0, C'_1)$



Operating operation profile data

Operation profile data uses files and open areas, and performs operation profile data manipulation (open, read/write) and operation profile data controls.

Operation profile data related function		Description
Operation profile data manipulation	Operation profile data open	Performs open of operation profile data from "File" to "Open area".
	Operation profile data read/write	Performs read/write of operation profile data of "File" and "Open area". Uses read/write data structure.
Operation profile data control		Performs control by using operation profile data of "Open area".

Data type of operation profile data

The profile data type structure and the profile ID type structure can be used as the data type to perform operation profile data manipulation/operation profile data controls.

• Files are managed as the profile data type labels.

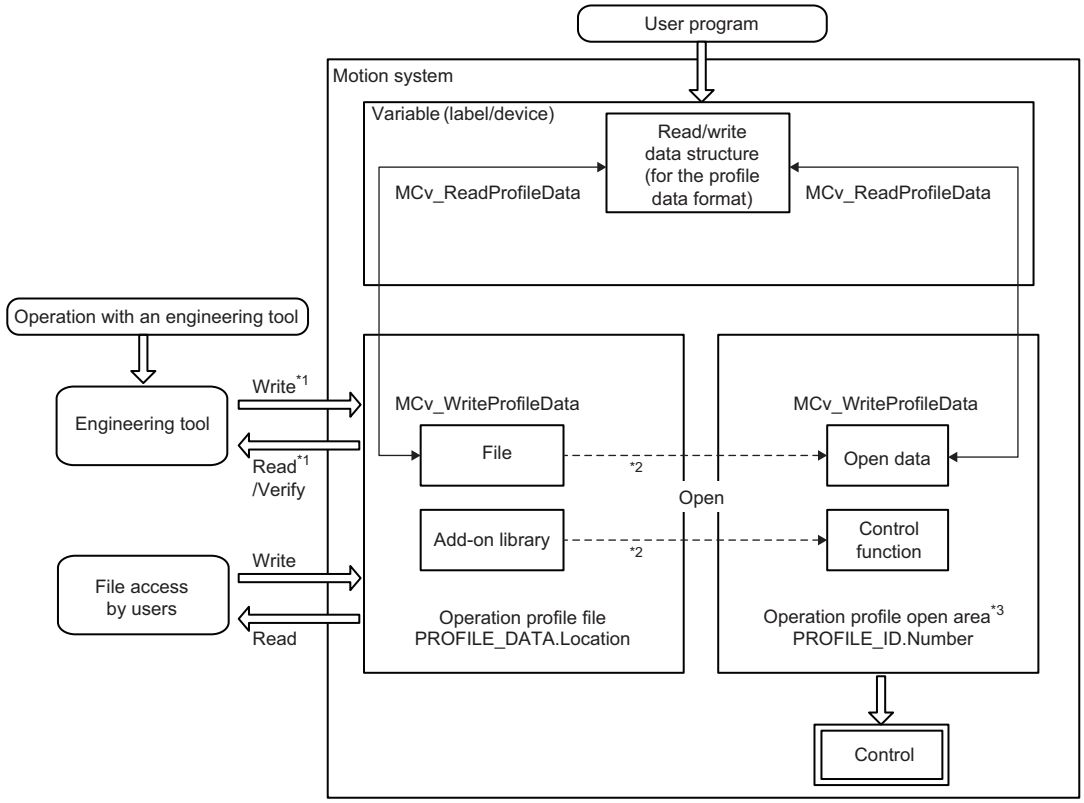
Specify the file path in PROFILE_DATA.Location.

• "Open area" is managed as the profile ID type labels.

When writing operation profile data in the open area by the operation profile data manipulation (open/write), the ID is assigned in the open area by specifying ID of the open area for PROFILE_ID.Number.

Control can be carried out by specifying the profile ID type labels at the operation profile data controls.

Entity of operation profile data	Description	Data type for operation/control	Remark
File	Written by the following operation. • Write from the engineering tool • Operation profile data write execution	Profile data type • General type (PROFILE_DATA type) • Function specific type (MC_CAM_REF type, etc.)	Data is preserved even when the power turns OFF.
Open area	Written by the following operation. • The power turns from OFF to ON • The PLC READY [Y0] turns from OFF to ON • Operation profile data open execution • Operation profile data write execution	Profile ID type • General type (PROFILE_ID type) • Function specific type (MC_CAM_ID type, etc.)	Data actually used in the control is stored. The data is deleted when the power turns OFF.



*1 Operation from the engineering tool is carried out to operation profile data file.

*2 When the power turns ON or when the PLC READY [Y0] turns ON, the operation profile data written by the engineering tool, etc. to which the ID is specified is automatically opened from the operation profile data file to the operation profile data open area.

*3 It is cleared when the system starts up.

Precautions

To hold the change, write file by MCv_WriteProfileData (file specification). Because the change is deleted when the power turns OFF, even if the data is changed in open area by MCv_WriteProfileData (open area specification) during operation.

■Details of profile data type/profile ID type

The profile data type structure and the profile ID type structure include "general-purpose type" which does not limit functions and "function specific type" which is used for the specific functions (FB). "Function specific type" includes "general-purpose type" in the member. Create instances of variable corresponding to the FB argument type to use.

Type name	General-purpose type	Function specific type	Remark
Profile data type	PROFILE_DATA	• MC_CAM_REF	Indicates the operation profile data with the following configurations. • Operation profile data storage location (Location) • Profile ID (ID)
Profile ID type	PROFILE_ID	• MC_CAM_ID	Data type to specify the operation profile data to use in the operation profile data control FB (MC_CamIn, etc.). A unique No. is given by the operation profile data open FB (MC_CamTableSelect, etc.). • Profile ID No. (Number)

The following table shows open FBs and control FBs that can be used to each function type.

Function	Profile data type	Profile ID type	Open FB	Control FB
Cam	MC_CAM_REF	MC_CAM_ID	MC_CamTableSelect*1	MC_CamIn
Profile operation	PROFILE_DATA	PROFILE_ID	MCv_ProfileSelect	MCv_ProfileCalc

*1 It can be opened with MCv_ProfileSelect.

■Operation profile data operation in the engineering tool

When operating from the engineering tool, set operation profile data file reads/writes to files. Operation profile data file cannot read/write to the open area.

■Capacity of operation profile data

Data size that created operation profile data file uses is as shown below.

Operation method	Operation profile data format	Open area
Create with the engineering tool Create with operation profile data manipulation function	Cam data	• When the interpolation method specification is "0: Linear interpolation": Management information (1K bytes) + Coordinate number × 8 × 2 (byte) • When the interpolation method specification is "1: Specifying for each section" and "2: Spline interpolation": Management information (1K bytes) + Resolution × 8 (byte)
	Cam data for rotary cutter	Management information (1K bytes) + Resolution × 8 (byte)
Create csv file directly	Multidimensional profile data	• For "2 inputs 1 output": Management information (1K bytes) + X coordinate number × Y coordinate number × 8 × 3 (byte) • For "3 inputs 1 output": Management information (1K bytes) + X coordinate number × Y coordinate number × Z coordinate number × 8 × 4 (byte)

In operation profile open processing, temporarily twice the data size memory of open area will be used for cam data (interpolation method specification is "1: Specifying for each section" and "2: Spline interpolation") and cam data for rotary cutter, and the same data size memory of open area will be used for other than the above.

Opening operation profile data

Performs open of operation profile data.

Open files specified with PROFILE_DATA.Location in the open area by using operation profile data open FB (MC_CamTableSelect, MCv_ProfileSelect).

- Data opened in the open area is assigned profile IDs expressed in the profile ID type.
- An operation profile data open FB that can be executed differs depending on the defined profile data type. For details, refer to the following.

  ☞ Page 489 Operation profile data types

  MCv_ProfileSelect can be opened regardless of the profile data type.
- When an error has occurred while opening the operation profile data, the open area in processing is discarded.

Precautions

When rewriting the open area during control, the stroke value changes rapidly, and shock may be applied to the machine depending on the waveform pattern. Create operation profile data so that the waveform before and after change does not become discontinuous.

Point

- An open instruction is carried out with motion service processing. It may take time to open depending on the processing details and the operation profile data.
- When there are Periodic, MasterAbsolute, or SlaveAbsolute as an input argument of the open FB, the file setting values are ignored, and the open is carried out referring the setting of the input argument. (The file setting values do not overwrite.)

■Details of setting items

• Master (master axis)/Slave (slave axis)/AxesGroup (axes group)

Ignores them.

• CamTable (cam table)/ProfileData (profile data)

Set the operation profile data with PROFILE_DATA type structure.

The value of PROFILE_DATA.Location is referred, and open is carried out to the operation profile data that the file name and the path indicate.

The ID that PROFILE_DATA.ID indicates is assigned in the open area.

PROFILE_DATA.ID.Number	Operation
When specifying "0"	An available ID is assigned automatically, and the ID is stored in PROFILE_DATA.ID.Number. If an open FB whose PROFILE_DATA.ID.Number is "0" is executed multiple times to the same operation profile data, the data will be opened in multiple different open areas. When there is no available ID, error "Insufficient Calculation Profile ID" (error code: 3452H) will occur and open processing will not be carried out.
When expressly specifying other than "0"	The open is carried out without changing the ID. If specifying an ID that has been already opened, the open data will be overwritten.

The following table shows the relationship between the value of PROFILE_DATA.ID.Number at open and the open location area.

Value of PROFILE_DATA.ID.Number		Open location area
Before open FB execution	After open FB execution	
0	Stores other than 0	New
Other than 0		Overwrite if there exists, new otherwise

If there is no operation profile data specified with PROFILE_DATA.Location, the error "No Calculation Profile" (error code: 340FH) will occur, and the open processing will not be carried out.

If operation profile data specified with PROFILE_DATA.Location is broken or if there is a problem in the format, the error "Operation profile data error" (error code: 3412H) will occur, and the open processing will not be carried out.

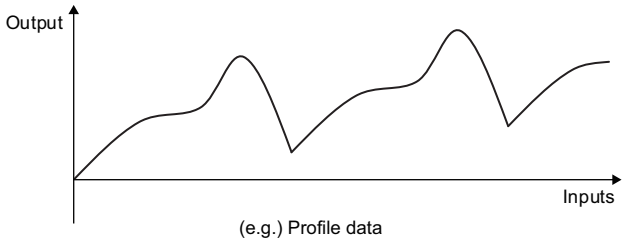
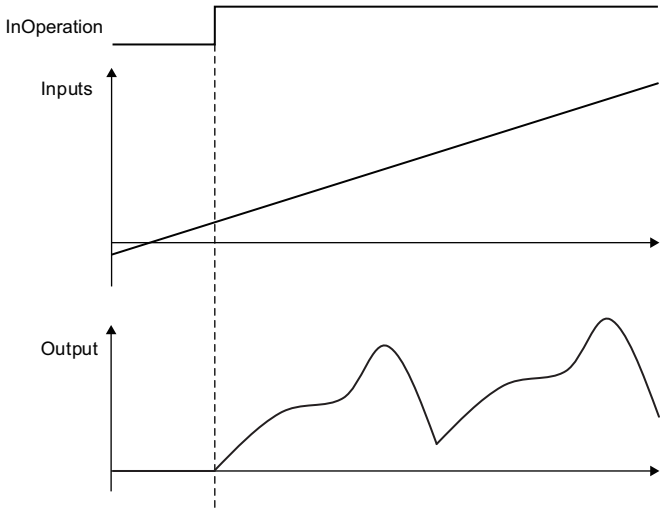
• Periodic (Periodic)

The following explains the operation at operation profile data control execution when Periodic is set.

Periodic	Operation
When FALSE is set	The control ends when it executes until the end of operation profile data.
When TRUE is set	Repeats continuously operation profile data execution.

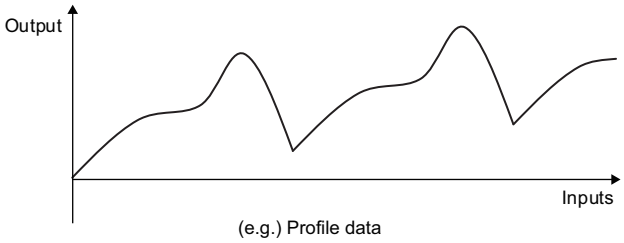
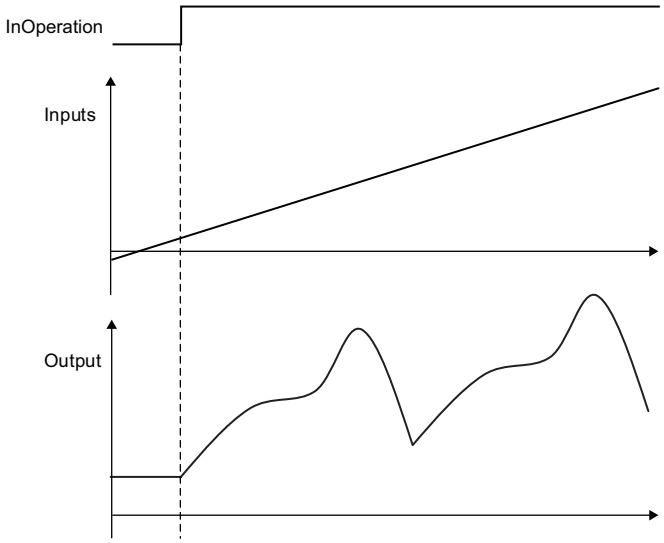
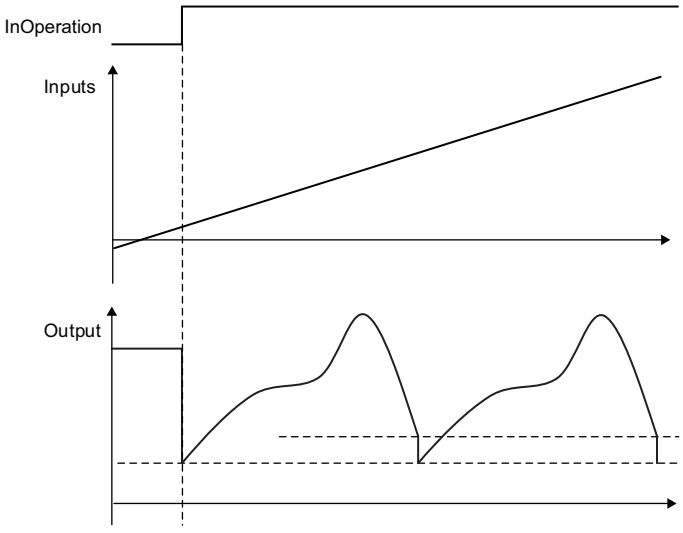
• MasterAbsolute (Master axis (input) absolute coordinate)

The following explains the operation at operation profile data control execution when MasterAbsolute mode is set. (When the input unit is time specification, it is ignored. (Fixed with relative coordinates)

MasterAbsolute	Operation
When FALSE is set	The current Inputs value becomes the start point of the operation profile data. When InOperation turns TRUE, Output is calculated according to the relative movement amount of Inputs.  (e.g.) Profile data 
When TRUE is set	When using this setting, the error "Out of Input Absolute Coordinate Range" (error code: 341DH) occurs.

• SlaveAbsolute (Slave axis (output) absolute coordinate)

The following explains the operation at operation profile data control execution when SlaveAbsolute mode is set.

SlaveAbsolute	Operation
When FALSE is set	When InOperation turns TRUE, Output is calculated based on the current value.  (e.g.) Profile data 
When TRUE is set	Calculates so that Output at the time when InOperation turns TRUE always becomes the start point when one cycle of the operation profile data starts. When there is a limit on the Inputs value, such as the cam data length per cycle, of the operation profile data, and the start point and the end point of the operation profile data are different when Periodic = TRUE (Periodic), the command is output in one operation cycle in order to return to the first Output at the next one cycle start. 

Automatic open of operation profile data

Opens operation profile data automatically.

Operation profile data that satisfy all of the following conditions can be used for control without executing an open FB, because it is automatically opened when the power turns ON.

- Other than "0" is specified to the initial value of PROFILE_DATA.ID.Number.
- The operation profile data is stored in %PROJECT_ROOT%/calc_profile/auto_open, or %PROJECT_ROOT_DRIVE%/calc_profile/auto_open.

Read/write operation profile data

Operates data read/write of operation profile data.

Operates data read/write of file an operation profile data in open area by using MCv_ReadProfileData/MCv_WriteProfileData according to data structure for read/write.

Operation profile data format to which the data structure for read/write is defined can operates read/write operation. For details on the data structure for read/write of each operation profile data format, refer to the following.

☞ Page 489 Operation profile data format: Cam data

When an error has occurred while writing operation profile data, the open area in processing is discarded.

Point

The number of data points that can be read/written at the same time are limited. If read/write is not completed in one time FB execution, carry out it in several times.

■Read operation profile data

Operation profile data can be read by using the MCv_ReadProfileData.

- The file specified in ProfileData and Target, or the operation profile data in open area are read for the number of points specified in Points from data specified with Offset.
- To read the whole operation profile data, specify "0" in both Offset and Points.
- Read data is stored in variables specified in Data1 and Data2.

■Write operation profile data

Operation profile data can be written by using the MCv_WriteProfileData.

- The file specified in ProfileData and Target, or the operation profile data in open area are written for the number of points specified in Points from data specified with Offset.
- To write the whole operation profile data, specify "0" in both Offset and Points.

■Details of setting items

- ProfileData (Profile data)

Set operation profile data for read/write with PROFILE_DATA structure.

Operation profile data format to which the data structure for read/write is defined can operates read/write operation.

If specifying operation profile data that does not correspond to read/write operation, the error "Calculation Profile Read/Write Unsupported" (error code: 345EH) will occur.

- Data1 (Read data 1/Write data 1)/Data2 (Read data 2/Write data 2)

Set read data/write data of operation profile data with TARGET_REF structure.

Data types of [VAR] and [DEV] can be used.

Data type that can be used is the type of the data structure for read/write of each operation profile data. For details, refer to the following.

☞ Page 495 Read/write data structure

Ex.

When reading operation profile data of the cam data format for 10 points of buffer memory (Data1 is G1000 or more and Data2 is G3000 or more)

Data1.Target:= "[DEV](PROFILE_CAM_DATA)G1000"

Data2.Target:= "[DEV](LREAL[1..10, 1..2])G3000"

If the type of the specified data structure for read/write does not match the type that is required for each operation profile data, the error "Read/Write Data Type Incorrect" (error code: 3462H) will occur.

• Offset (Offset)

Specifies offset from the top for operation profile data specified in ProfileData.

If Offset exceeds the range of operation profile data, the error "Out of Offset Range" (error code: 3413H) will occur, and read/write will not be performed.

When setting the value other than 0 to Offset, also set the value other than 0 to Points. (When "0" is set, the error "Number of Offset/Read/Write Data Incorrect" (error code: 3465H) will occur.)

Set 0 to write operation profile when Target = 1 (File). (When other than "0" is set, the error "Number of Offset/Read/Write Data Incorrect" (error code: 3465H) will occur.)

• Points (Number of read data/Number of write data)

Specifies the number of data points to read and the number of data points to write.

When Points exceeds the number of elements of the operation profile data, read/write is carried out within the number of elements of the operation profile data range.

When the interpolation method specification of operation profile is "0: Linear interpolation" and "2: Spline interpolation", set 0. (When other than "0" is set, the error "Number of Offset/Read/Write Data Incorrect" (error code: 3465H) will occur.)

Set 0 to write operation profile when Target = 1 (File). (When other than "0" is set, the error "Number of Offset/Read/Write Data Incorrect" (error code: 3465H) will occur.)

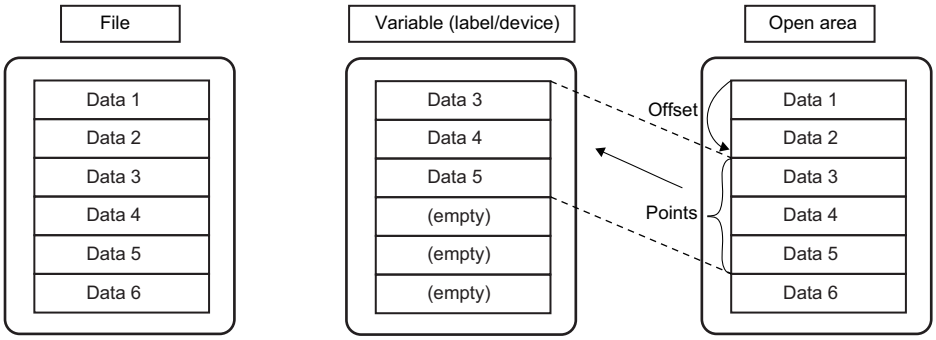
• Target (Read destination/write destination)

Specifies read destination and write destination of the operation profile data.

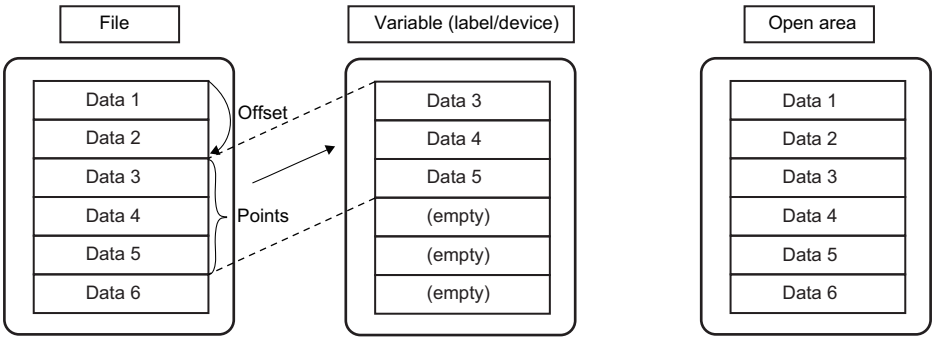
Target	Read operation	Write operation
0: OpenArea	• The value of PROFILE_DATA.ID is referred, and read is carried out to the operation profile data that the ID indicates in open area. • If operation profile data does not exist (unopened), the error "No Calculation Profile" (error code: 340FH) will occur.	The value of PROFILE_DATA.ID is referred, and write is carried out to the operation profile data that the ID indicates in open area. The operation is the same as open of operation profile depending on the specified value to PROFILE_DATA.ID.Number. For details, refer to the following. ☞ Page 506 Details of setting items
1: File	• The value of PROFILE_DATA.Location is referred, and read is carried out to the operation profile data that the file name and the path indicate. • If operation profile data does not exist, the error "No Calculation Profile" (error code: 340FH) will occur.	• The value of PROFILE_DATA.ID is referred, and write is carried out to the operation profile data that the ID indicates in open area. Then The value of PROFILE_DATA.Location is referred, and write is carried out to the operation profile data that the file name and the path indicate. • If the specified file exists, overwrite is carried out. • If the specified file does not exist, a new file will be created when both Offset and Points are "0", and be written in Unicode (UTF-16 Little Endian with BOM). • When creating a new file, both Offset and Points must be "0", otherwise the error "Number of Offset/Read/Write Data Incorrect" (error code: 3465H) will occur. (Write to the open area is executed.)

Ex.

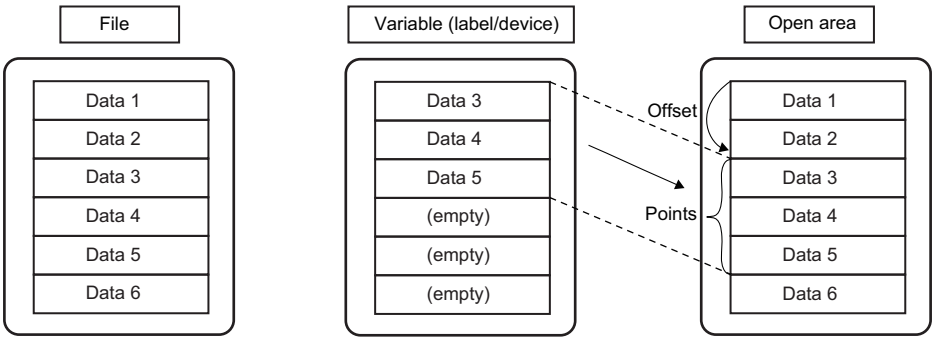
Read process when Target = 0 (OpenArea), Offset = 2 and Points = 3 are specified



Read process when Target = 1 (File), Offset = 2 and Points = 3 are specified



Read process when Target = 0 (OpenArea), Offset = 2 and Points = 3 are specified



Point

- When Target is "1: File", processing is carried out at a cycle with lower priority than the operation cycle. Therefore, it may take time to read/write depending on processing details and operation profile data.
- When Target is "0: OpenArea", read/write is carried out depending on the cycle of executed program.

Precautions

When the operation profile data is written in the open area or file area using MCv_WriteProfileData, the setting value of periodic, master axis (input) absolute coordinate and slave axis (output) absolute coordinate will be FALSE.

Control of operation profile data

Operation profile data is used to control.

Control details

The input data of operation profile data control FB is the PROFILE_ID type and the control is executed based on the specified operation profile data. The operation profile data control FB has the following public variables as the monitor data during control.

- InputsPerCycle (Current value per cycle)
- Reference (Reference value)
- OutputData (Output value)
- InstanceID (Instance ID)

Before executing operation profile data control, open the operation profile data corresponding to each operation profile data control FB in the open area to enable this control.

■Profile operation

Output data for input data is calculated by using multidimensional profile data.

Inputs, Output, InputsOffset, OutputOffset, InputsScaling, OutputScaling and ProfileID are set by using MCv_ProfileCalc, and the control is executed based on multidimensional profile data.

Before using this function, open the multidimensional profile data in the open area of the operation profile data to enable this control.

Output to the signal set in Outputs (Output data indicated as follows) is controled based on the signal set in Inputs (Input data indicatd as follows) and the multidimensional profile data in ProfileID.

■Details of setting items

- Inputs (Input data)

Specify the input data with the array of the number of dimensions of the TARGET_REF structure of the input data.

The data type, [VAR] and [DEV] can be used.

Data type that can be used is LREAL.

- Output (Output data)

Specify the output data with the TARGET_REF structure.

The data type, [VAR] and [DEV] can be used.

Data type that can be used is LREAL.

- InputsOffset (Input offset)/OutputOffset (Output offset)

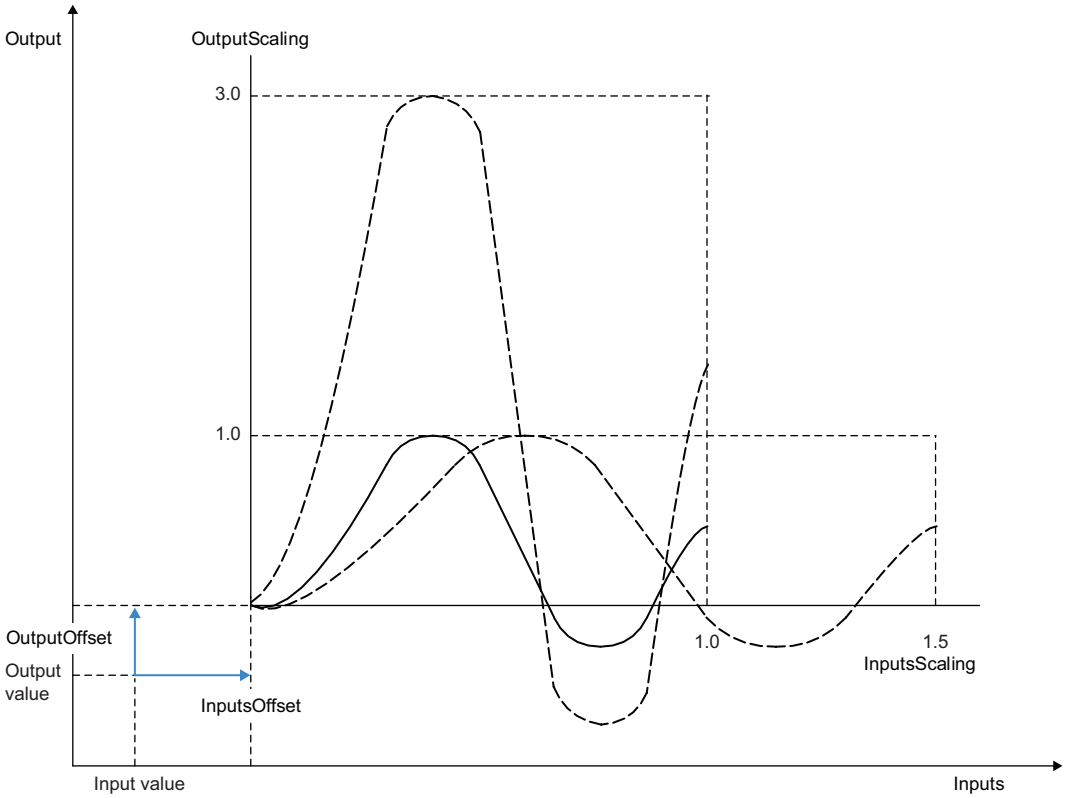
The phase/displacement Inputs/Output can be shifted by setting InputsOffset/OutputOffset.

The phase of input value can be shifted by InputsOffset.

InputsOffset can be specified as an array for each element of Inputs respectively.

• InputsScaling (InputScaling)/OutputScaling (OutputScaling)

The operation profile data can be enlarged or reduced by setting InputsScaling/OutputScaling (OutputScaling).
InputsScaling can be specified as an array for each element of Inputs respectively.



• ProfileID (Profile ID)

Specify the multidimensional profile data ID with the PROFILE_ID structure.

Start mode

The following table shows the operation when simultaneously executing various FBs (open, write, read, control) that access to open areas of operation profile data to the same profile ID.

• ExecutionMode

For the operation profile data open FB and the operation profile data write FB, the operation for simultaneous execution can be specified with ExecutionMode.

○: Execution possible, ×: Error

On-going FB	FB to be executed			
	Open/Write			Read/Control
	mcImmediately	mcQueued	mcSpeculatively	
Open/Write	○	○	×	×
Read/Control	○	○	×	○

[0: mcImmediately]

The description in the open area is immediately changed by an open FB and a write FB. The change may affect the control in execution.

However, if the operation profile format and the resolution do not match when executing open/write during FB execution, the error "Operating operation profile data" (error code: 3411H) occurs.

[1: mcQueued]

Waits execution completion of on-going FB. When multiple FBs are waiting, the FBs will be executed in high priority task. For the same priority, the FBs will be executed in the order of the start.

[3: mcSpeculatively]

The error "Operating operation profile data" (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 specification of ExecutionMode. While the FB (open, write, read, control) that accesses open areas is in execution, Busy of various FBs are TRUE. Use this signal to the interlock as needed.

Sub function of operation profile data control

Current value per cycle change function

Change the current value per cycle of an operation profile data control FB to an arbitrary value.

• Set MC_CamIn.InstanceID of operation profile data control FB (Instance name indicates MC_CamIn as follows) in InstanceID. It is valid to FBs which have MC_CamIn.

• Public variables of MC_CamIn are shown below when current value per cycle change is performed.

Public variable	Updating value	Remark
InputsPerCycle	Value specified in Cycle[]	—
Reference	Stroke value equivalent to Cycle[]	MC_CamIn.Reference is refreshed to fix MC_CamIn.OutputData.
OutputData	Not updated	—

Point

Reference is not refreshed if MC_CamTableSelect.SlaveAbsolute is TRUE (absolute coordinate), but Reference is refreshed to fix OutputData if current value per cycle change is performed.

• There are two methods as follows for current value per cycle change.

Method	Description
MCv_ChangeCycle	Change it to the specified value at control execution.
MCv_ChangeCycleTriggered	Compensate the specified value at the trigger input signal detection timing. Change it to the compensated value.

• For specification of the current value per cycle to be changed, "absolute specification" or "relative specification" can be selected by Relative.

■Current value per cycle change

Uses MCv_ChangeCycle.

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

Executes current value per cycle change at rising edge detection of Execute.

Whether the change of the current value per cycle waits until the operation profile data control FB execution is completed can be selected by ExecutionMode.

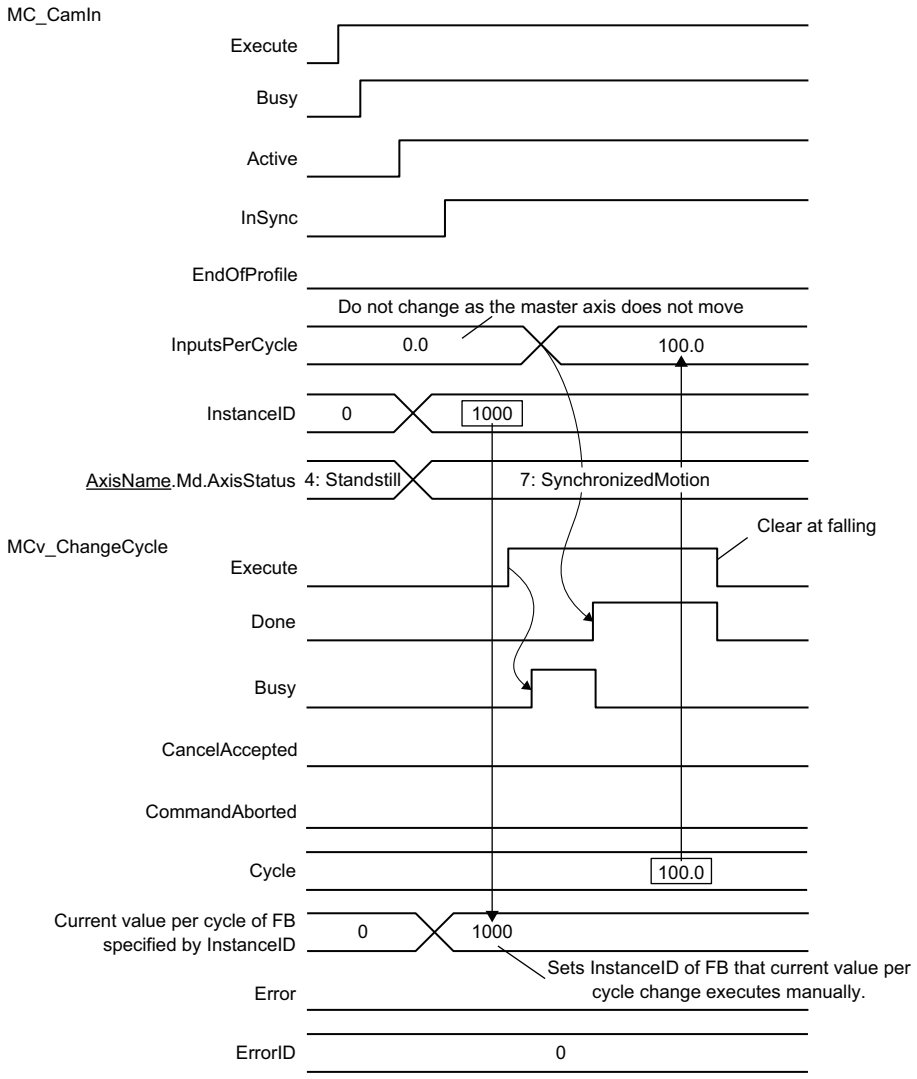
• ExcutionMode (Start mode)

[0: mcImmediately]

Executed immediately at rising edge detection of Execute with MC_CamIn.Busy = TRUE.

Ex.

Sets Cycle = 100.0 when MC_CamIn.Busy = TRUE, and when executing current value per cycle change. (Current value of master axis of MC_CamIn shall not change.)



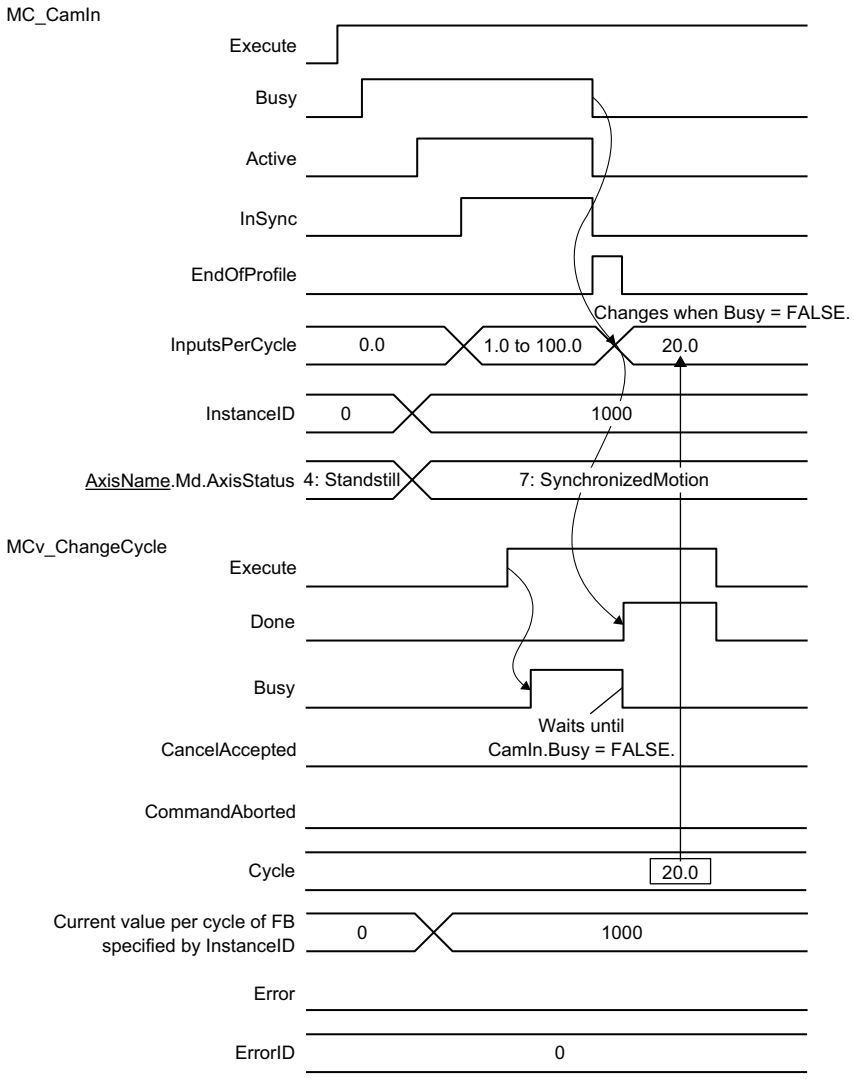
[1: mcQueued]

Becomes Busy = TRUE at rising edge detection of Execute.

Executed after waiting until MC_CamIn.Busy = TRUE turns FALSE.

Ex.

When setting Cycle = 20.0, and executing current value per cycle change.

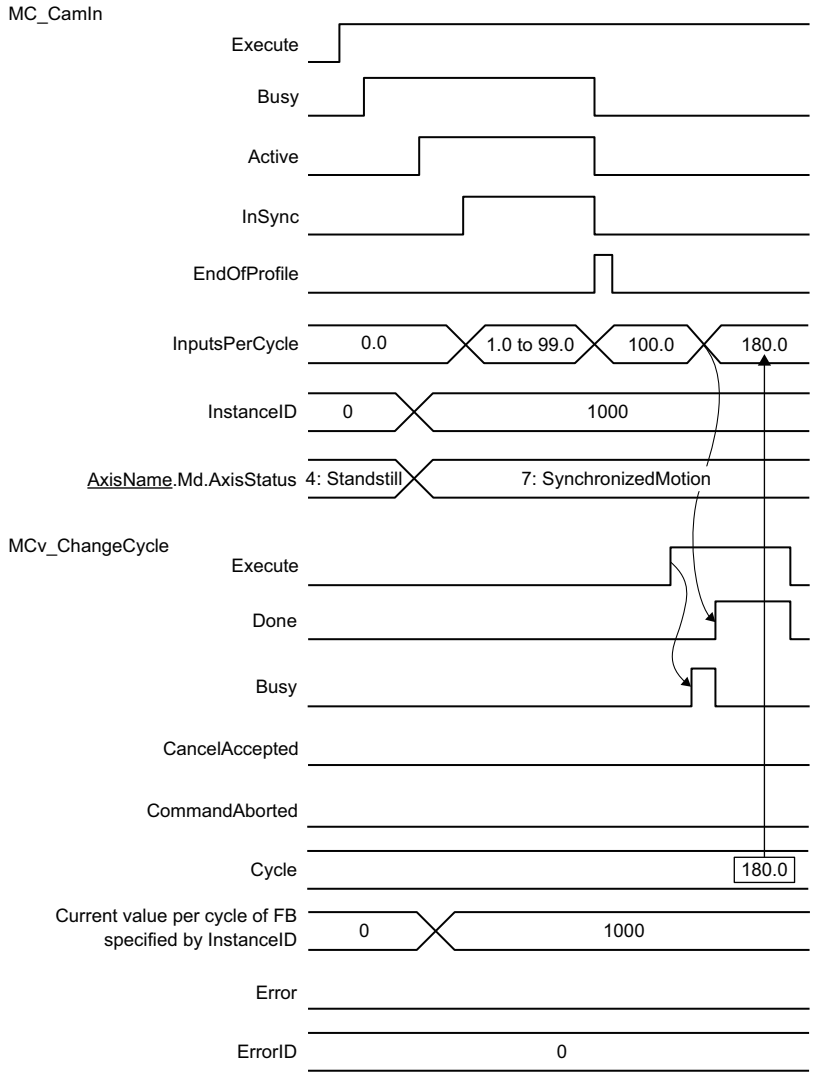


[3: mcSpeculatively]

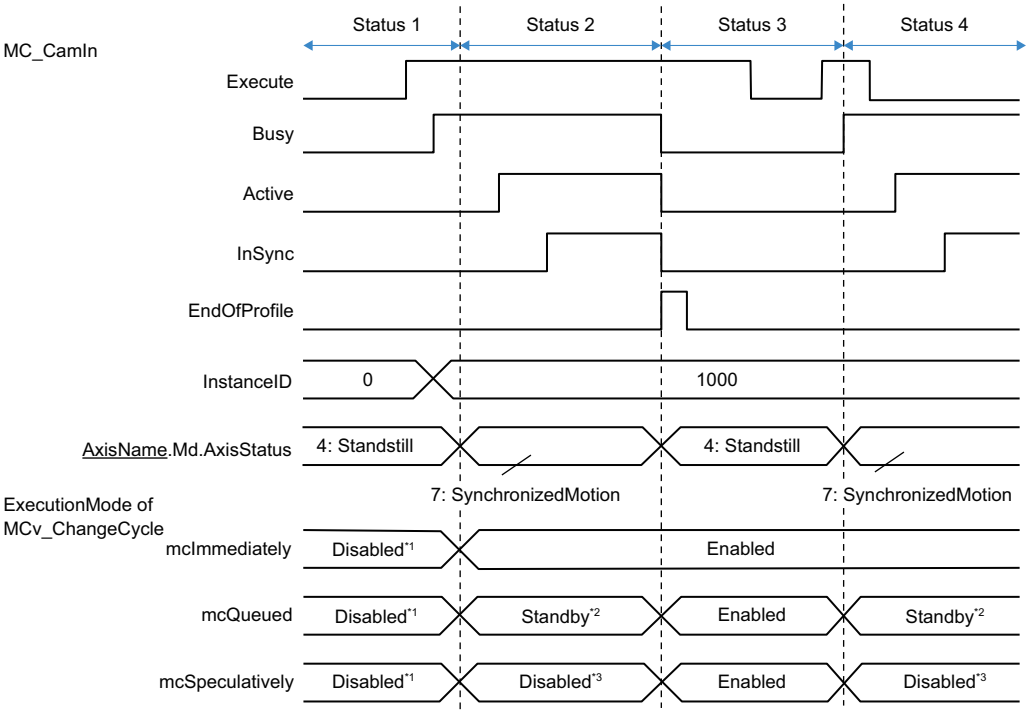
Execute when MC_CamIn.Busy = FALSE.

Ex.

When setting Cycle = 100.0, and executing current value per cycle change



[Current value per cycle change timing by each value]



*1 The error "No Instance ID" (error code: 3468H) occurs.
*2 Standby until AxisName.Md.AxisStatus turns to "4: Standstill".
*3 The error "MCv_ChangeCycle Instruction Error" (error code: 3497H) (Detail Code: 4) occurs.

Status		Remark
Status 1	FB of current value per cycle change target is not started	The error "No Instance ID" (error code: 3468H) occurs regardless of ExecutionMode.
Status 2	FB of current value per cycle change target is starting	*1
Status 3	FB of current value per cycle change target is completed	Current value per cycle change can be executed regardless of ExecutionMode. FB of change target is reflected by execution cycle. When waiting start mode "1: mcQueued" in Status 2, the target FB is reflected at the timing which AxisName.Md.AxisStatus turns to "4: Standstill".
Status 4	FB of current value per cycle change target is after restarting	The operation is the same as Status 2. Repeats after that as Status 3 → Status 4 (Status 2) → Status 3 → ... Current value per cycle change operation is not affected even Execute has TRUE/FALSE differences in Status 2 and Status 4.

*1 Depending on ExecutionMode, the timing is as follows.

Start mode	Change timing
0: mcImmediately	Current value per cycle change can be executed. The execution cycle of target FB is reflected.
1: mcQueued	Standby untill it turns to enable (Status 3). (No error occurs.)
3: mcSpeculatively	The error "MCv_ChangeCycle Instruction Error" (error code: 3497H) (Detail Code: 4) occurs.

■Trigger current value per cycle change

Uses MCv_ChangeCycleTriggered.

Changes the current value per cycle of the specified operation profile data control FB to the value which is compensated at the detection timing of the trigger input signal.

Executes current value per cycle change when the trigger signal is detected after the rising edge detection of Execute.

When TriggerInput.Signal.CompensationTime is a positive value, it is executed immediately.

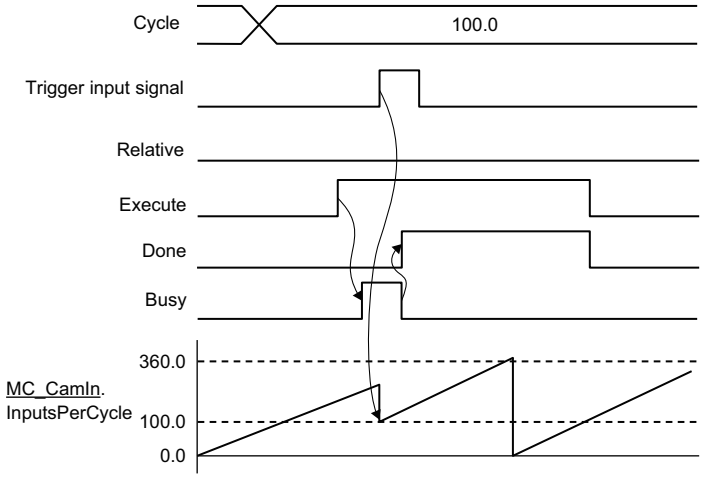
When TriggerInput.Signal.CompensationTime is a negative value, this FB waits for operation to start until the correction time has elapsed, and it is executed.

MC_CamIn.InputsPerCycle is estimated and changed so that the position will be a position set in Cycle at Trigger signal detection timing.

Ex.

Operation example by Trigger input signal

When setting Cycle = 100.0 during MC_CamIn operation, and executing current value per cycle change.



- TriggerInput (Trigger input signal)

Specifies Trigger input signal with the MC_TRIGGER_REF structure.

Setting range of TriggerInput.Signal is shown below.

Item		Setting range
TriggerInput.Signal		
	Source	Type BOOL Data type [VAR] [DEV] [CONST] [OBJ]
	Detection	2: RisingEdge (Detection at FALSE→TRUE (rising edge)) 3: FallingEdge (Detection at TRUE→FALSE (falling edge)) 4: BothEdges (Detection at rising edge/falling edge)
	CompensationTime	When Relative = TRUE: 0.0 When Relative = FALSE: -5.0 to 5.0 [s]

■Details of setting items

• InstanceID (Instance ID)

Specifies the instance ID of operation profile data control FB.

Instance ID is enabled from the controller power turns ON to OFF.

• Cycle (Current value per cycle)

Specifies the current value per cycle to be changed.

When Relative = TRUE: MC_CamIn.InputsPerCycle is changed to the value added Cycle to the current MC_CamIn.InputsPerCycle.

When Relative = FALSE: MC_CamIn.InputsPerCycle is changed to the Cycle value.

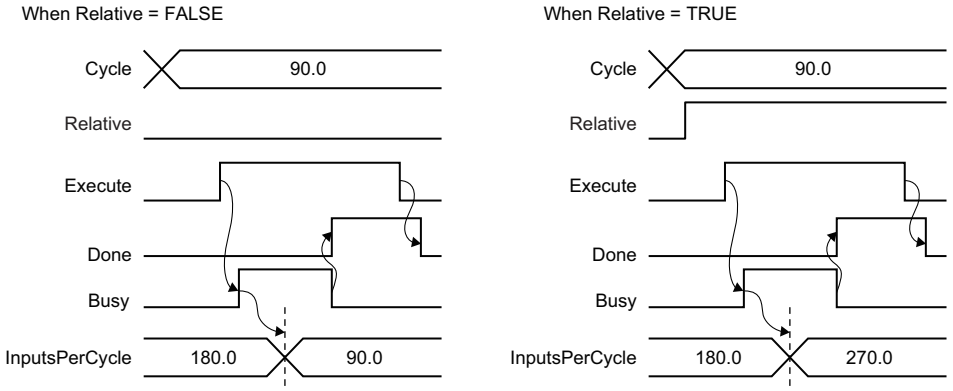
Settable Cycle value differs depending on Relative being FALSE or TRUE. Specify the value within the setting range below. If a value outside the setting range is set, the error "MCv_ChangeCycle Instruction Error" (error code: 3497H) (Detail Code: 1) occurs.

Cycle	Relative	Setting range
[0]	FALSE	0 ≤ Setting value < Length per cycle
	TRUE	-(Length per cycle) / 2 ≤ Setting value ≤ (Length per cycle) / 2
[1], [2]	FALSE/TRUE	0

Ex.

Operation example by Relative

When executing the current value per cycle change of Cycle = 90.0 while it stops with MC_CamIn.InputsPerCycle = 180.0.



• Relative (Relative selection)

When Relative = TRUE, Cycle is "Relative specification", when Relative = FALSE, Cycle is "Absolute specification".

• Options (Options)

Specify the function options.

Bit	Function description
0 to 15	Empty (Specify "0".)*1
16	Cancel accepted after FB start 0: Not allow 1: Allow
17 to 31	Empty (Specify "0".)*1

*1 If specifying other than "0", the error "Out of Options Range" (error code: 1A4EH) occurs, and the function does not start.

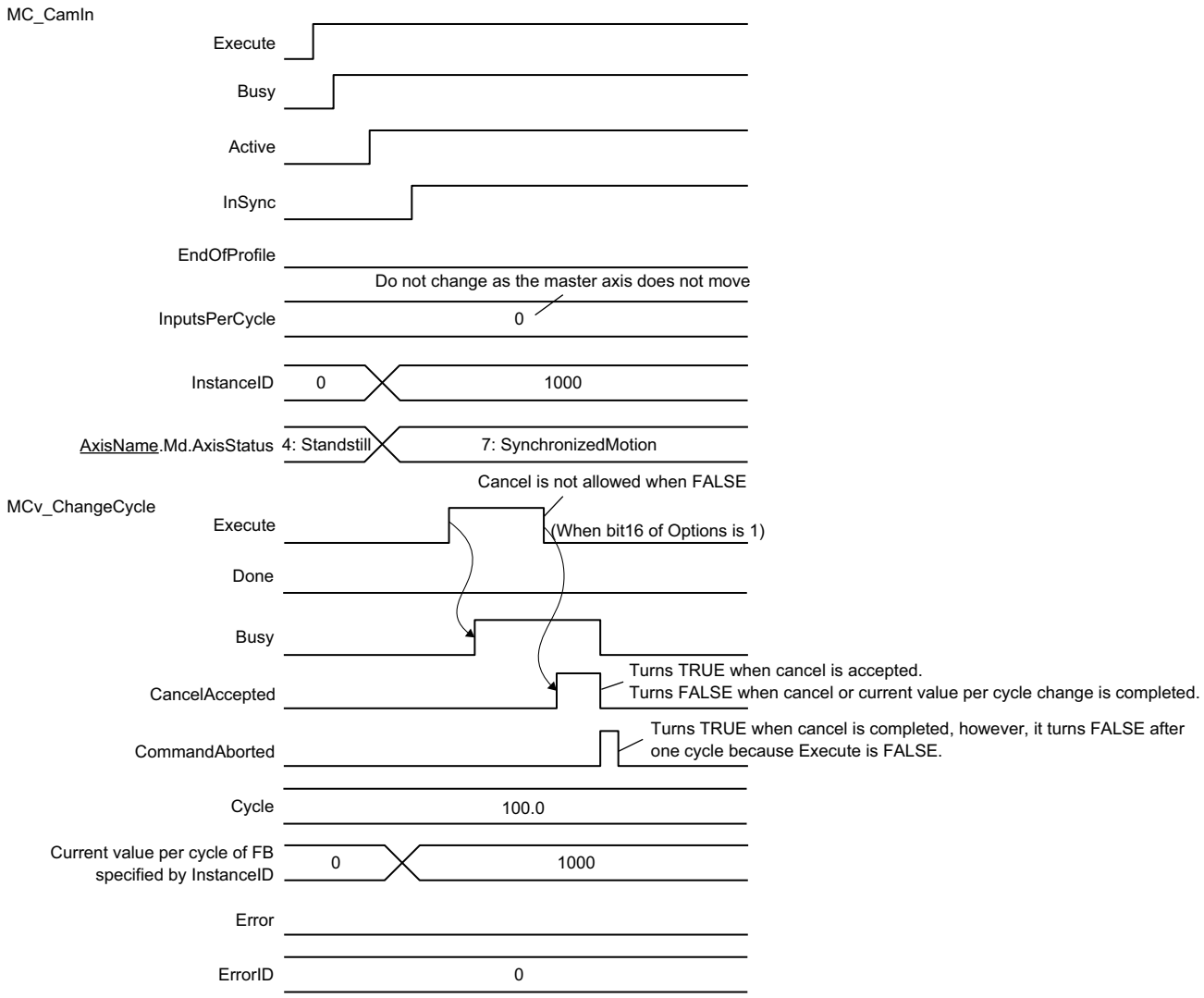
Current value per cycle change that is in the standby status can be canceled after FB execution by cancel enabled setting.

- Cancel is started at falling edge detection of Execute.
- Cancel is accepted only when Busy = TRUE.
- When FB accepts cancel, becomes CancelAccepted = TRUE.
- When cancel is completed, becomes CommandAborted = TRUE.
- When cancel was carried out, InputsPerCycle is not changed.

Ex.

Cancel operation example by MCv_ChangeCycle

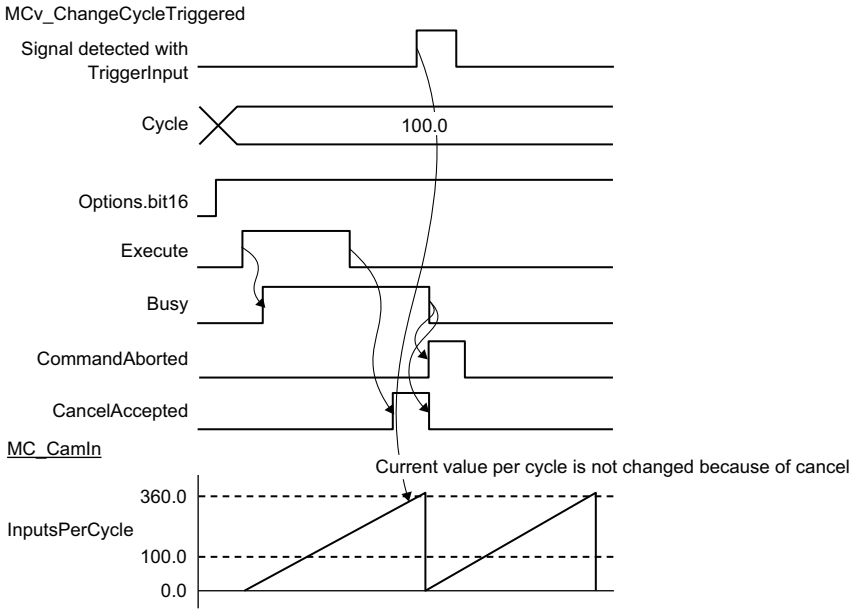
When canceling while MC_CamIn.Busy = TRUE is waiting for switching to FALSE after starting ExecutionMode with "1: mcQueued"



Ex.

Cancel operation example by MCv_SetPositionTriggered

When canceling while it is waiting for detecting a trigger input signal after starting



14.2 Operation Profile Data Format

This section describes csv file format of operation profile data.

Point

- "//" comment lines are not counted among line Nos.
- 8 byte specification data is imported as the double-precision real number type.

Operation profile data details

The basic configuration of profile data is as the following table.

Line No.	Description
1	For system
2	Open information
3	Operation profile data format
4	Resolution/coordinate number
5	Start point, initial stroke amount, options
6	Number of sections
7 or later	Defined for each operation profile data format
N-2*1	Length per cycle
N-1*1	Stroke amount
N*1	Common footer (EOF)
N+1*1	Common footer (CRC)

*1 N is calculated as shown below by specifying interpolation method.
"0: Linear interpolation": 7 (start row of previous step) + (coordinate number × 2)
"1: Specifying for each section": 7 (start row of previous step) + Number of sections
"2: Spline interpolation": 7 (start row of previous step) + Number of sections

The common items as operation profile data are shown below.

○: Omission possible, ×: Omission not possible

Line No.	Column No.	Item	Omission	Setting range (Initial value)	Setting format	Description	Supported file version
2	1	Profile ID	×	☞ Page 473 Operation Profile Data Initial value: 0 (= automatic apply)	Unsigned integer	Specifies Profile ID. (Use only when carrying out automatic open when the power turns ON.)	0x0103
	2	Periodic	○	☞ Page 473 Operation Profile Data	Integer	Specifies periodic.	0x0103
	3	Input absolute coordinate	○	☞ Page 473 Operation Profile Data	Integer	Specifies input absolute coordinate.	0x0103
	4	Output absolute coordinate	○	☞ Page 473 Operation Profile Data	Integer	Specifies output absolute coordinate.	0x0103
3	1	Operation profile data format	×	99: Cam data format	Integer	Specifies operation profile data format.	0x0103
	2	Profile data type	○	0: General profile data 1: Cam data 6: Digital output data format Initial value: 0	Integer	Specifies the profile data type. ☞ Page 473 Operation Profile Data	0x0103
	3	Interpolation method specification	○	0: Linear interpolation 1: Specifying for each section 2: Spline interpolation Initial value: 0	Integer	Specifies the interpolation method between data.	0x0103
4	1	Resolution/ coordinate number	×	☞ Page 473 Operation Profile Data Initial value: 0	Unsigned integer	Sets the resolution or the number of coordinates.	0x0103
5	1	Start point	○	*2 99: Valid for the cam data format (specified for each section, spline interpolation) Initial value: 0	Real number	Specifies the start point.	0x0103
	2	Initial stroke amount	○	*2 99: Valid for the cam data format (specified for each section, spline interpolation) Initial value: 0	Real number	Specifies the initial stroke amount.	0x0103
	3	Initial velocity	○	*2 99: Valid for the cam data format (specified for each section) Initial value: 0	Real number	Specifies the initial velocity	0x0103
	4	Initial accrlrration	○	*2 99: Valid for the cam data format (specified for each section) Initial value: 0	Real number	Specifies the initial accrlrration	0x0103
	5	Options	○	*3 Initial value: 0	Unsigned integer	Defines for each operation profile data format.	0x0103
6	1	Number of sections	×	*3 Initial value: 0	Unsigned integer	Specifies the number of sections for the data format that requires it. Specify "0" for the data format that does not require it.	0x0103

*2 Differs depending on the data format.

*3 Depends on the operation profile data format.

• (N-2)th line: Length per cycle

○: Omission possible

Line No.	Column No.	Item	Omission	Setting range (Initial value)	Setting format	Description	Supported file version
N-2	1	Length per cycle	○	0 or more*1	Real number	Specifies the length per cycle. Specify "0" for the data format that does not require it.	0x0103
	2	Unit setting	○	[Rules] F: An arbitrary unit Initial value: 0x0F For settable units, refer to the following. ☞ Page 79 Technical Units	Unsigned integer	Specifies the unit of the length per cycle. Units that can be specified differ depending on the profile data type. ☞ Page 473 Operation Profile Data	0x0103
	3	Unit character string	○	Initial value: "" ☞ Page 79 Technical Units	Character string	Specifies an arbitrary character string when "an arbitrary unit" is set to the unit setting.	0x0103
	4	Minimum value per cycle	○	*2	Real number	Specifies the minimum value of length per cycle. Specify "0" for the data format that does not require it.	0x0103
	5	Maximum value per cycle	○	*2	Real number	Specifies the maximum value of length per cycle. Specify "0" for the data format that does not require it.	0x0103
	6	Time per cycle	○	0 or more*1	Real number	Specifies the time per cycle. Specify "0" for the data format that does not require it.	0x0103

*1 The maximum value differs depending on the data format.

*2 Differs depending on the data format.

• (N-1)th line: Stroke amount

○: Omission possible

Line No.	Column No.	Item	Omission	Setting range (Initial value)	Setting format	Description	Supported file version
N-1	1	Stroke amount	○	*1	Real number	Specifies the stroke amount. Specify "0" for the data format that does not require it.	0x0103
	2	Unit setting	○	[Rules] F: An arbitrary unit Initial value: 0x0F For settable units, refer to the following. ☞ Page 79 Technical Units	Unsigned integer	Specifies the unit of stroke amount. Units that can be specified differ depending on the profile data type. ☞ Page 473 Operation Profile Data	0x0103
	3	Unit character string	○	Initial value: "" ☞ Page 79 Technical Units	Character string	Specifies an arbitrary character string when "an arbitrary unit" is set to the unit setting.	0x0103

*1 Differs depending on the data format.

Cam data format (Interpolation method specification: "0: Linear interpolation")

The following shows a case for specifying setting value below.

Line No.	Column No.	Item	Setting value
3	1	Operation profile data format	99: Cam data format
	3	Interpolation method specification	0: Linear interpolation

■7th line or later: I/O data

×: Omission not possible

Line No.	Column No.	Item	Omission	Setting range (Initial value)	Setting format	Description	Supported file version
7	1	Input value (0th point)	×	☞ Page 83 Positioning Range	Real number	☞ Page 473 Operation Profile Data	0x0103
8	1	Output value (0th point)	×		Real number		0x0103
9	1	Input value (1st point)	×		Real number		0x0103
⋮							
For the number of coordinates × 2	1	Output value (nth point) (For the number of coordinates)	×	☞ Page 83 Positioning Range	Real number	☞ Page 473 Operation Profile Data	0x0103

Cam data format (Interpolation method specification: "1: Specifying for each section")

The following shows a case for specifying setting value below.

Line No.	Column No.	Item	Setting value
3	3	Interpolation method specification	1: Specifying for each section

■6th line: Number of sections

Specify the number of sections.

■7th line or later: Parameter

○: Omission possible, ×: Omission not possible

Line No.	Column No.	Item	Omission	Setting range (Initial value)	Setting format	Description	Supported file version
7	1	Cam curve types	×	Sets the cam curve with a value. For settable curves, refer to the following. ☞ Page 473 Operation Profile Data	Integer	☞ Page 473 Operation Profile Data	0x0103
	2	End point	×	☞ Page 473 Operation Profile Data	Real number		0x0103
	3	Stroke	×		Real number		0x0103
	4	Curve applicable range (P1)	○		Real number		0x0103
	5	Curve applicable range (P2)	○		Real number		0x0103
	6	Acceleration/ deceleration range compensation (Range L1)	○		Real number		0x0103
	7	Acceleration/ deceleration range compensation (Range L2)	○		Real number		0x0103
	8	End point velocity	○		Real number		0x0103
	9	End point acceleration	○		Real number		0x0103
8	1-7	Same as line No.7					
⋮ (For the number of sections)							

Cam data format (Interpolation method specification: "2: Spline interpolation")

The following shows a case for specifying setting value below.

Line No.	Column No.	Item	Setting value
3	3	Interpolation method specification	2: Spline interpolation

■6th line: Number of sections

Specify the number of sections.

■7th line or later: Parameter

×: Omission not possible

Line No.	Column No.	Item	Omission	Setting range (Initial value)	Setting format	Description	Supported file version
7	1	End point	×	☞ Page 473 Operation Profile Data	Real number	☞ Page 473 Operation Profile Data	0x0103
	2	Stroke	×		Real number		0x0103
8	1-2	Same as line No.7					
⋮ (For the number of sections)							

PART 3 CONTROL/OPERATION/ MAINTENANCE

15 LOGGING

16 RAS FUNCTIONS

17 MOTION SERVICE PROCESSING

18 HOW TO CONTROL FROM CPU MODULE

19 FILE CONTROL

20 SECURITY

21 MODULE SOFTWARE INSTALLATION

22 TROUBLESHOOTING

15 LOGGING

"Logging" is used as a generic term for "Data logging" and "Real-time monitor". This chapter describes how to use data logging and real-time monitor.

15.1 Data Logging/Real-Time Monitor

Operation of this function at each status of system

○: Possible, ×: Not possible

Status	Operation availability
STOP	○
RUN	○
Moderate error	○
Major error	×

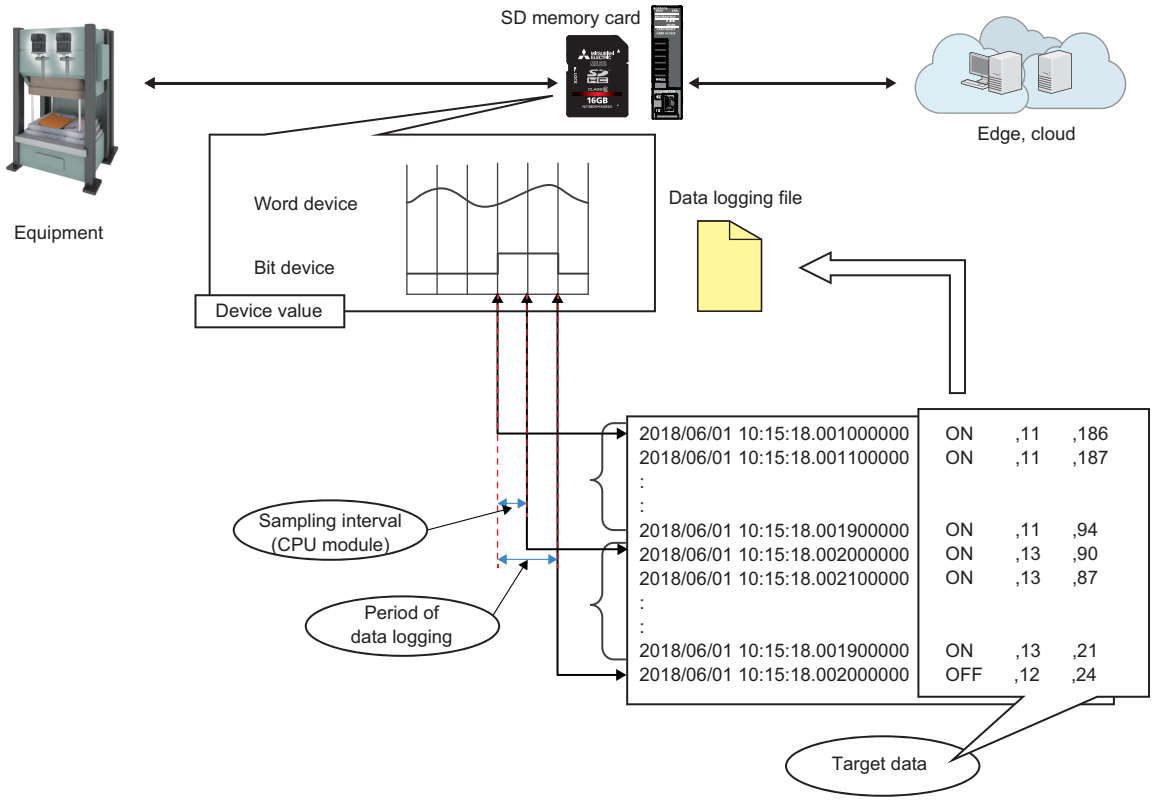
Relevant variables

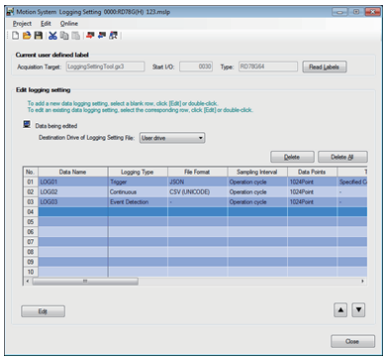
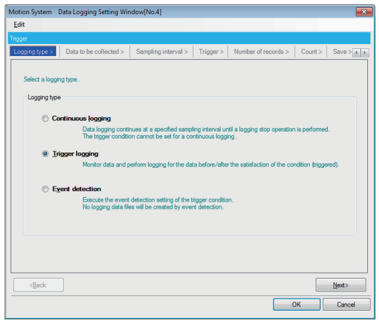
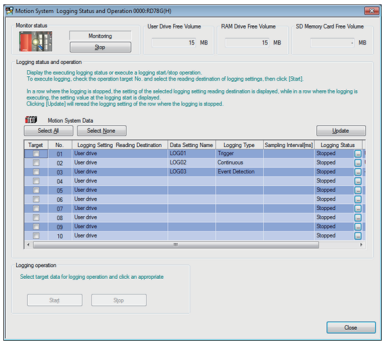
Variable/Structure name	Name	Details
System.LoggingRef[1..10].Cd.*1		
SettingPath	Logging setting path	Sets the storage location path (folder name/file name) of logging setting.
StartLog	Logging request	Requests the execution of logging. FALSE: STOP TRUE: RUN
Trigger	Trigger request	Requests the trigger when "Trigger command" is set for a trigger condition. FALSE: No request TRUE: Request
System.LoggingRef[1..10].Md.*1		
SettingPath	Logging setting file	Indicates the path (file name) of the logging setting file during the logging.
LogDataPath	Logging data file path	Indicates the path (file name) of the logging data file.
LoggingStatus	Logging status	Indicates the logging execution status. FALSE: Stop TRUE: Executing
Triggered	Triggered	Indicates the triggered status.
TriggerCount	Trigger detection count	Indicates the trigger detection count.
TrigerCountIgnored	Trigger ignored count	A trigger occurs again during collecting data after trigger and it indicates the trigger ignored count.
LogDataSavedCount	Number of saved files	Indicates the number of saved data logging files.
TriggerStatus[1..32]	Trigger status	Indicates the status of each trigger condition. FALSE: Trigger condition not established TRUE: Trigger condition established
SaveStatus	Save status	Indicates the save status to files of collected data. 0: UnSave (Unsaved) 1: Saving (Saving) 2: Saved (Saved) 3: FullySaved (Fully saved)
LogType	Logging type	Indicates the logging type in progress.
ErrStatus	Error status	Stores the error code related to logging.
SamplingTime[0..1]	Data collection interval	Indicates the data collection interval in a unit of ns. [0] Sampling interval lower 32 bits [1] Sampling interval upper 32 bits
System.PrConst.		
Logging_AutoLoggingEnable	Auto logging enabled	Sets "Allowed" or "Not allowed" of auto logging function when the logging setting file is stored in the SD memory card. 0: Disabled 1: Enabled

*1 The array elements [1] to [10] correspond to the logging setting 1 to 10.

Data logging function

Data logging function is used to collect the motion system data at a specified interval based on the logging setting (trigger condition or data collection condition) written by the engineering tool and save the result to the data logging file.
The collected data is saved as the logging file. The data logging setting up to 10 can be executed at the same time in the motion system.



Motion system logging setting window	Motion system setting window	Motion system logging state check and operation window
		

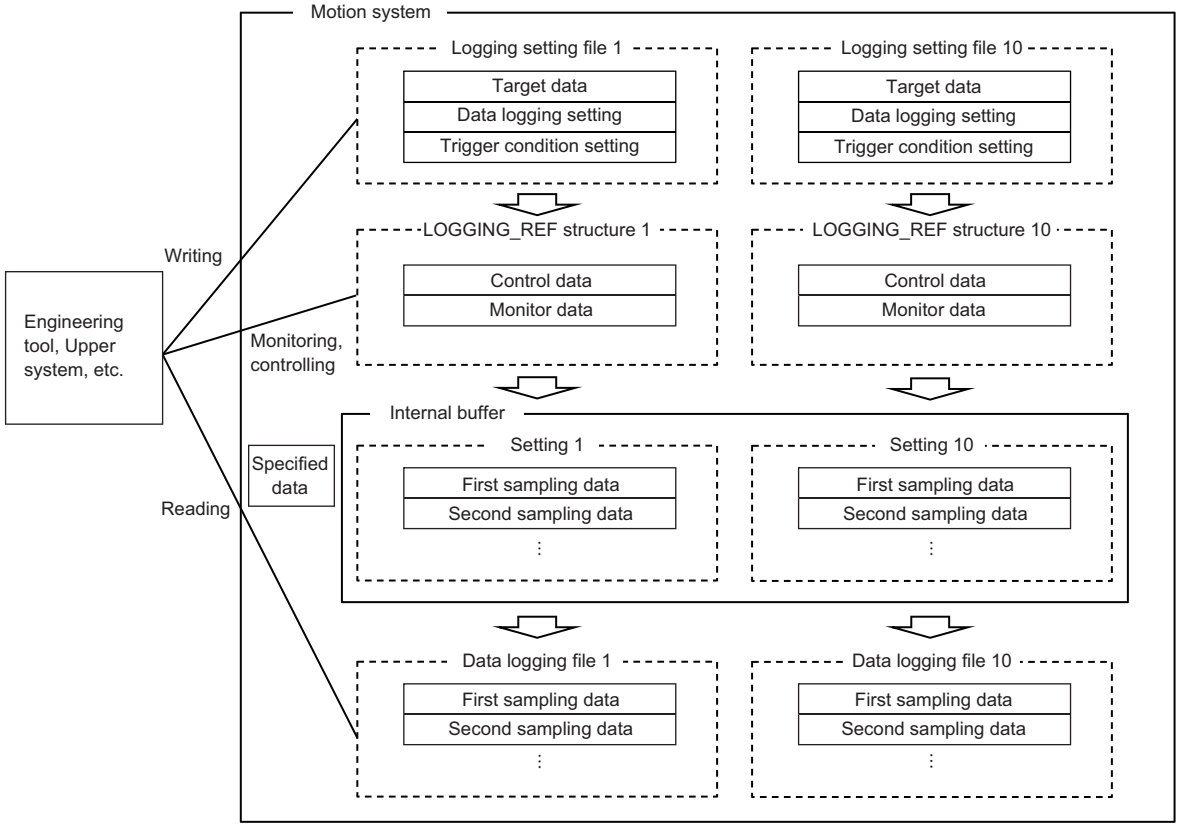
Real-time monitor

Real-time monitor is used to set the data collection and monitor the collected data (waveform display) in real time in the status where the engineering tool has been connected with the motion system.
For details, refer to Help of the engineering tool.

Steps until the collected data is saved

Data logging

The motion system executes data logging according to the description of logging setting files.
The control and monitor of data logging are executed by the LOGGING_REF structure.
The collected data is saved to the internal buffer in real-time processing and it is saved to the file in back-ground processing.

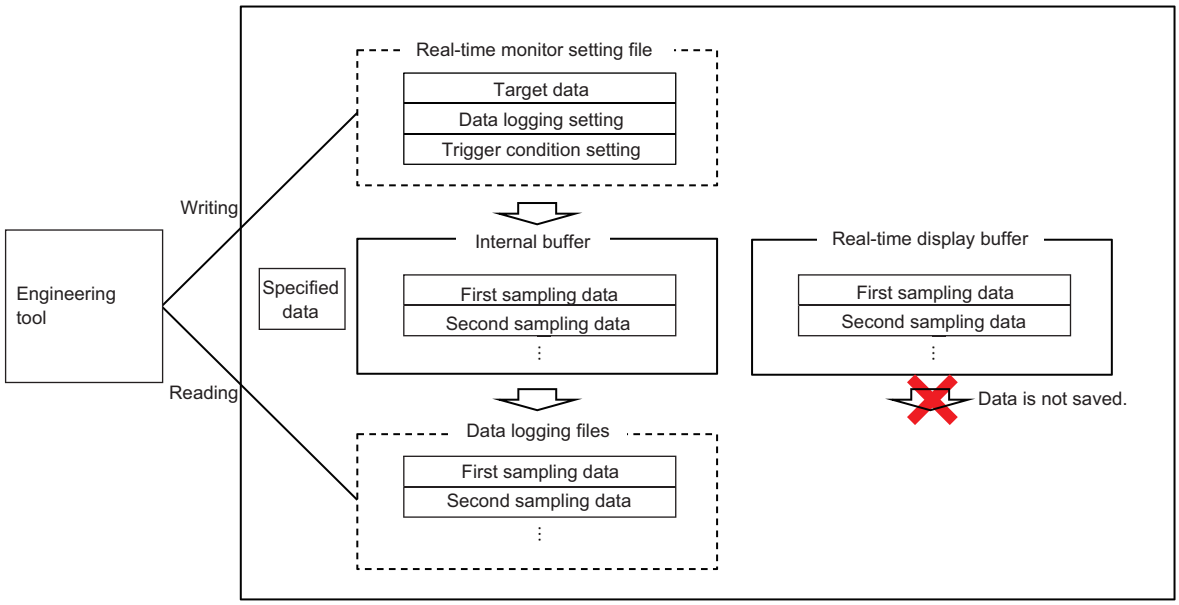


Real-time monitor

Real-time monitor is used in the status where the engineering tool has been connected with the motion system.
The motion system executes the followings according to the description of the real-time monitor setting file set in the engineering tool.

Execution details	Data	Maximum number of points
Data collection to the internal buffer	All of the specified data	Max. 1024 points
Data output to the buffer for the real-time display	Data that are selected as the real-time waveform display target from the specified data	Max. 32 points

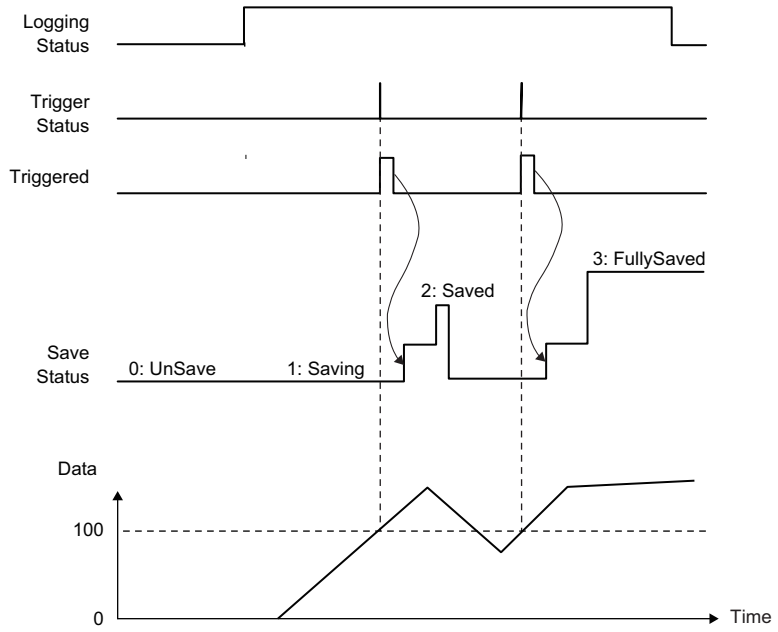
When the data collection is completed, the collected data in the internal buffer is saved and displayed in the engineering tool.
Real-time monitor can start with new setting independently from data logging. Also, it can start based on the setting of the data logging in progress.



Status of logging

The status of logging in progress can be checked by System.LoggingRef[1..10].Md.LoggingStatus, System.LoggingRef[1..10].Md.TriggerStatus[1..32], System.LoggingRef[1..10].Md.Triggered, and System.LoggingRef[1..10].Md.SaveStatus.

The following explains the operation at trigger logging when Data = 100 is set for a "Trigger condition", and 2 is set for a "Number of trigger logging".



It turns TRUE while trigger condition is processed for System.LoggingRef[1..10].Md.TriggerStatus[1..32], and after trigger condition is processed for System.LoggingRef[1..10].Md.Triggered. After the completion of collecting the number of records set to the "number of records after trigger", they both returns to FALSE.

After the completion of saving files, it turns the state of System.LoggingRef[1..10].Md.SaveStatus as follows.

3: FullySaved

- When the number of records specified during continuous logging "Number of file switching records" is all saved in a file.
- When the number of records specified during trigger logging "Number of records after trigger" is all saved in a file, and when the files are saved at the number of specified times at "Number of trigger logging".

2: Saved

- When the file is saved with the state that does not meet the condition of "3: FullySaved".

However, after the status becomes "2: Saved" or "3: FullySaved", the status moves to be "0: UnSave" immediately if the logging is in progress.

Internal buffer

The internal buffer which is used to temporarily store collected data frees up the memory capacity up to the capacity assigned to the logging function by "System memory setting". Adjust the system memory setting according to the number of logging settings and data.

The following shows a rough standard of the required amount of memory for logging.

Minimum required amount per setting (K bytes) = 100 + (The total number of the appropriative words for each data) × R × B

Item	Description
Minimum required amount per setting	It may increase more than the minimum required amount depending on the setting order of the data when different types of word data exist as the target data. For the internal buffer, the number of appropriative words of the smaller type may be rounded up to the number of the appropriative words of the next bigger type. Because the two-word data and the four-word data are set in the two-word border and the four-word border respectively. (Ex.) When setting the order of four-word data, one-word data, four word data, the number of the appropriative words is 12. (The number of the appropriative words for one word data is rounded up to 4.)
Total number of the appropriative words for each data	The number of the appropriative words per one for each data is shown below. Bit: 1 / 16 (one word for 16) Word (signed): 1 Double word (signed): 2 Word (unsigned): 1 Double word (unsigned): 2 Single-precision real number: 2 Double-precision real number: 4
R: Number of records	During continuous logging: R = Number of file switching records (SAVSWICTMNRECNUM) In trigger logging: R = Number of trigger records before trigger (LLNBEFOR) + Number of trigger records after trigger (LLNAFTER)
B: Number of buffer	When the specification of the number of buffer (NUMBUFF) is 2 to 10: B = NUMBUFF When the specification of the number of buffer (NUMBUFF) is 0, and when the number of records R × collection interval is within 1 [s]: B = 4 When within 1 [s] to 5 [s]: B = 3 Other than the above: B = 2

SD memory card replacement

When the storage location of the data logging files is set to an SD memory card, the SD memory card can be replaced by using the SD memory card forced disable function even while data logging is in progress. Only the data saving to the SD memory card is stopped. The data collection keeps working.

Precautions

If a new SD memory card is inserted before the next file saving is started, the logging result is saved to the new SD memory card. If an SD memory card is not inserted before the next file saving is started, the file saving is failure and an error occurs.

Logging setting

The following describes how to set data logging/real-time monitor.

The ( ) of the title indicates the object name of logging setting files (Json format) in this below.

Data to be collected (DATA)

Set data to be collected by logging. The number of data can be set is up to 1024.

Duplicate data records are counted as distinct.

Data name (NAME)

Set the data name to be collected (device, label). For the format specification, refer to the following.

☞ Page 420 TARGET_REF structure

Data type (TYPE)

The following shows data type that can be specified to the data name.

- Bit
- Word [unsigned]
- Double word [unsigned]
- Word [signed]
- Double word [signed]
- Single-precision real number
- Double-precision real number

When the device is specified to the data name, it is required to specify the data type as well. The data type also can be specified by the specification of the NAME object. When both NAME and TYPE are specified, the specification of NAME takes priority.

When the label is specified to the data name, the data type specification is ignored.

Logging setting (LOGGING)

Set the logging operation.

Logging start conditions (STARTCONDITION)

Set a logging start condition.

Logging start conditions	Description
Starts automatically	When the motion system starts*1, System.LoggingRef[1..10].Cd.StartLog becomes TRUE and logging starts automatically. Logging can start and stop after the system starts by System.LoggingRef[1..10].Cd.StartLog.
Stop disabled after automatically start	When the motion system starts*1,System.LoggingRef[1..10].Cd.StartLog becomes TRUE and logging starts automatically. Logging cannot stop manually (System.LoggingRef[1..10].Cd.StartLog changes to FALSE).
Starts by user operation	Logging can start and stop by System.LoggingRef[1..10].Cd.StartLog.

*1 When the system starts, the logging setting file exists as follows, and when the logging start condition is "Starts automatically", the logging starts.
/rom/LOGGING/LOG**/LOGGING.json (** corresponds to the logging setting No.01 to 10.)
When using the logging setting file on the SD memory card, refer to the following.
☞ Page 544 Auto logging

When the logging is stopped, the data in buffer at the time is output to the file.

Collection interval (SMPHSPDTYPE, SMPHSPDTIME)

Set the collection timing of the target data.

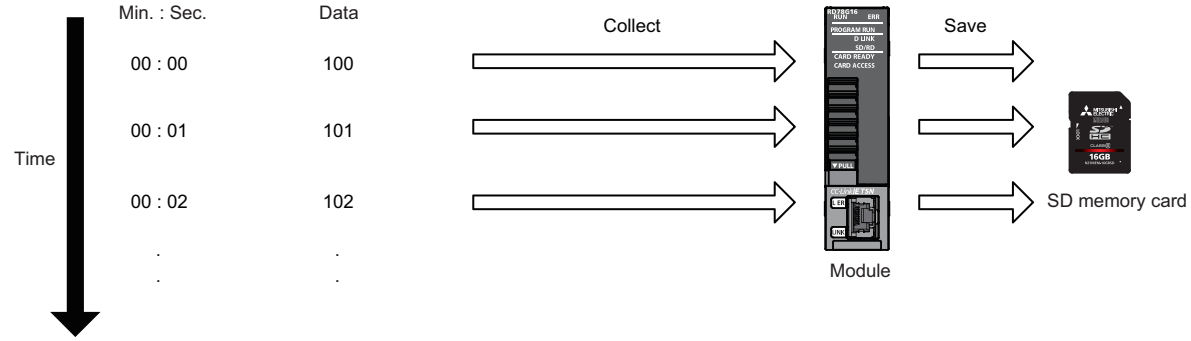
Data collection timing	Description
Operation cycle	Collects data in operation cycle interval. From the first to third operation cycles are available.
Time specification	Collects data at specified time interval (ms). (from 1 to 86400 ms) The closest cycle among cycles obtained by multiplying the first to third operation cycles by an integer becomes the actual collection interval. (When there are multiple options of the cycle, the longer cycle is obtained) (Ex.) When the first operation is 4.0 ms, the second operation is nothing, and the third operation is nothing • When specifying 6.0ms, the actual collection interval is 8.0 ms (the first operation cycle × 2) (Ex.) When the first operation is 0.25 ms, the second operation is 1.0 ms, and the third operation is 4.0 ms • When specifying 1.8ms, the actual interval is 1.75 ms (the first operation cycle × 7) • When specifying 1.9ms, the actual interval is 2.0 ms (the second operation cycle × 2) • When specifying 8.0ms, the actual interval is 8.0 ms (the third operation cycle × 2)

Logging type (TFFLGTP)

Set the logging type to be executed. The following describes available methods of data collection.

■Continuous logging

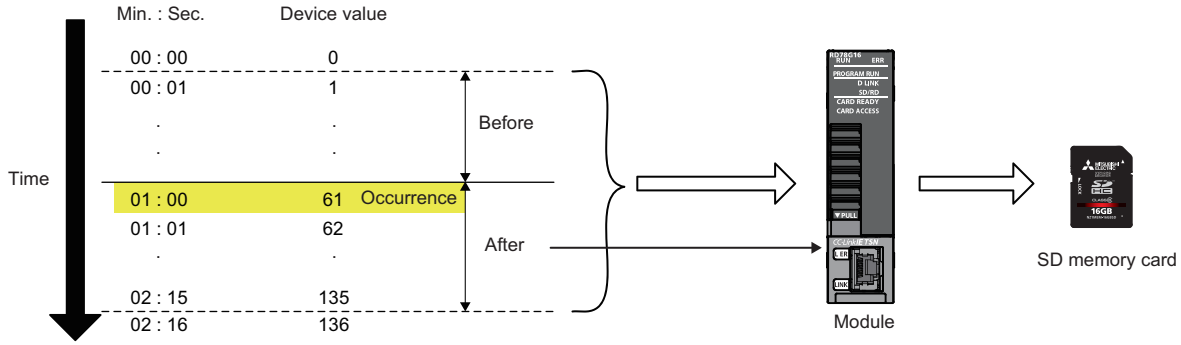
In continuous logging, the module collects specified data continuously and keeps saving the data logging file at file switching timing.



[Operating procedure for continuous logging]

1. Set the logging setting and start logging.
2. Collect data and save the file with the condition specified at "File switching timing".
3. Data collection finishes at the following cases: upon reaching the number of files to be saved specified as part of the "Stop" setting configured in "Operation when exceeds the number of files", or when System.LoggingRef[1..10].Cd.StartLog changes to FALSE.

■Trigger logging



In trigger logging, the module stores specified data in its internal buffer; it extracts a specified number of data records before and after the satisfaction of a trigger condition and saves the extracted data in a data logging file. Note that data collection is performed not only at the specified collection interval or timing but also when a trigger condition is met.

[Operating procedure for trigger logging]

1. Set the logging setting and start logging.
2. Wait until the trigger condition is met. (The trigger condition status can be checked by System.LoggingRef[1..10].Md.Trggered and System.LoggingRef[1..10].Md.TrggerStatus[1..32].)
3. When the trigger condition is met, collect as much data as the number of records specified in the number of records after trigger and save the collected data to the file.
4. System.LoggingRef[1..10].Md.SaveStatus changes to "2: Saved" and "3: FullySaved" when the writing of the collected data to the file is completed.
5. Data collection is completed when the file is saved at the number of times set in "Number of trigger logging" or System.LoggingRef[1..10].Cd.StartLog is changed to FALSE.

Number of trigger logging (TRIGGERTIME)

Set the repeat number of trigger logging. A file is saved each time a trigger is executed once.

Number of trigger logging	Description
1 to 32767 (Number of times specification)	Executes repeatedly at the number of specified times. When exceeding the number of files that specified to the "Number of save files", the setting of "Operation when exceeding the number of files" will be ignored and "Overwrite" will be operated.
0 (Number of save files specification)	Executes repeatedly according to the "Number of save files" and the "Operation when exceeding the number of files". When the file save setting (SAVENABLE) is the "Not save", it executes repeatedly without the limit of number of times.

Number of records (LLNBEFOR, LLNAFTER)

Set the number of records before and after trigger condition occurrence.

Item	Description	Setting range
Number of records (before trigger)	Sets the number of records to be output as pre-trigger record.	0 to 999999
Number of records (after trigger)	Sets the number of records to be logged during and after a trigger occurrence.	1 to 1000000

Point

If there are many records, the free space of the operation folder may be insufficient. The place of the operation folder can be changed. For details, refer to the following.

☞ Page 545 Precautions

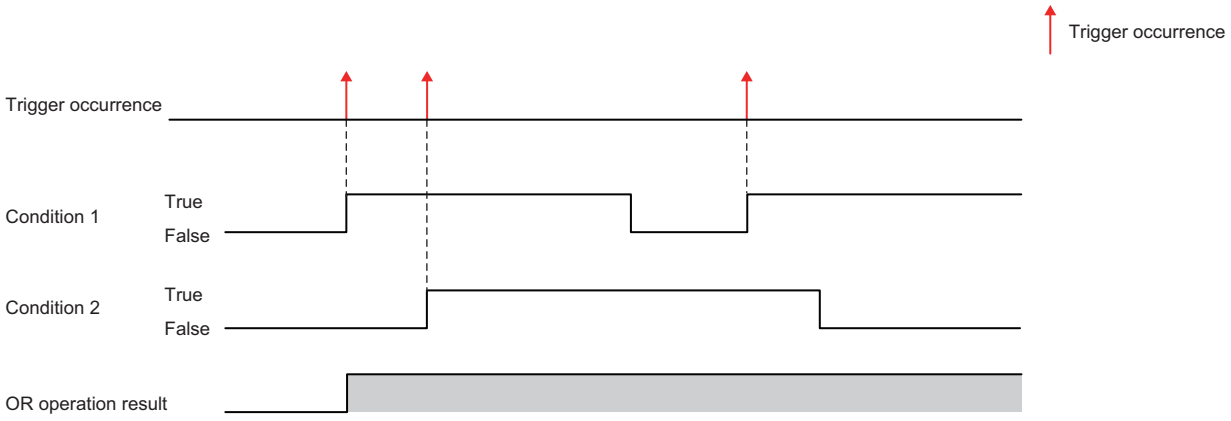
Trigger type (TRGCOMPTYPE)

Set the combination of multiple trigger conditions. One trigger type can be set for one setting.

Trigger type	Number of trigger conditions can be set	Trigger conditions can be specified
OR combine	Up to 32	[The following conditions are regarded as an AND/OR condition is satisfied when a condition is satisfied only.] • Data conditions (Comparison) • Data conditions (Value change) • Fixed cycle [The following conditions are regarded as an AND/OR condition is satisfied after a condition is satisfied once.] • System start • Trigger command • Event history
AND combine	Up to 32	

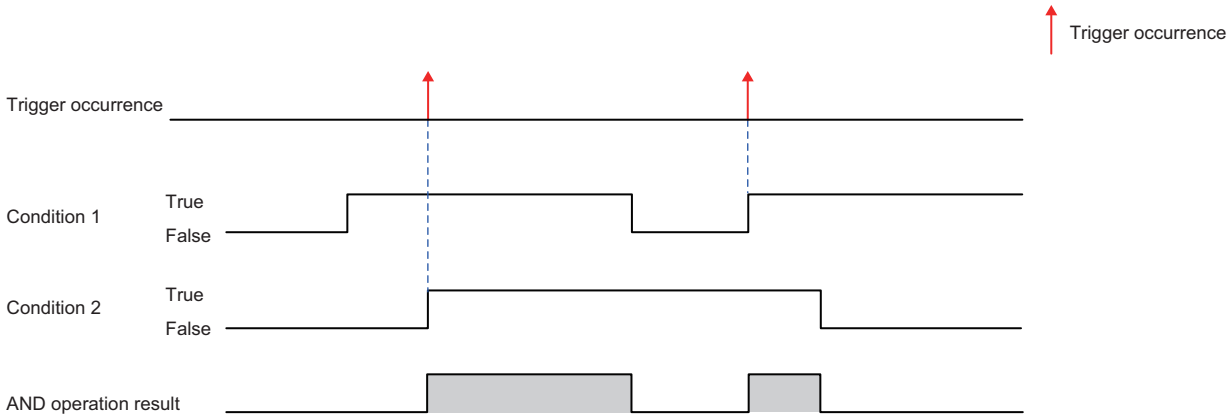
■OR combine

When any of the set conditions is satisfied, the trigger occurs.



■AND combine

When all the set trigger conditions are satisfied, the trigger occurs.



File save setting (SAVENABLE)

Set the saving enabled/disabled of a data logging file. When "Disabled" is set, the collected data file is not saved and only the trigger detection is executed.

File storage location (SAVFLDNAMEDETLFRMT)

Set a folder name of the data logging file storage location. When this setting is empty (""), the file is saved in the same folder as the logging setting file.

The file name is the date (YYYYMMDD)_time(hhmmss).extension.

The file created in the storage location is overwritten when the file with the same file name already exists. It is recommended to specify a different storage location for each file when specifiying the location in this setting and executing multiple logging settings simultaneously.

File format (TFFFILE)

The following storage formats are available for data logging files. For details on the output format, format specifications, and output contents of each file, refer to the data output format.

☞ Page 548 Data Logging Format

File format	Description
CSV (Unicode)	This is a file format which can be opened in generic-purpose application programs such as table calculated software and GX LogViewer.
JSON	This is a file format which is appropriate for data operation on the host system. The file size will be smaller than CSV (Unicode) format. It can also be opened in GX LogViewer.

Point

- To display date and time in the 1st column by unit of 10 ns when opening CSV (Unicode) file by Excel®, import the file in text format, and specify the data format "character string" to the 1st column by text file wizard.
- A time and date data recorded in the data logging file is based on the system time (the time of control CPU module) at data collection. Therefore, the setup collection interval and display time may not match. Check the fixed periodicity (missing or not) of the data in INDEX column (reset to INDEX = 1 when missing occurs) instead of the display time.

Number of files to be saved (SAVFNUM)

Set the maximum number of data logging files (Files of the extension specified with "File format") in the storage location folder.

The setting range of files is 0 to 65535.

When the setting value is greater than or equal to 1, the current number of files is stored in System.LoggingRef[1..10].Md.LogDataSavedCount. The operation when the number of files exceeds the setting value is specified by "Operation when exceeds the number of files".

When the setting value is 0, the maximum number of files is not checked. When the logging starts, System.LoggingRef[1..10].Md.LogDataSavedCount becomes 0, after this the saved number of files is stored.

Point

It may take time to save files if the number of maximum files is large.

Operation when exceeding number of files (SAVFNUMTYPE)

Set the operation when the number of files in the storage location folder exceeds the setting value of "Number of files to be saved".

Setting	Description
Overwrite	When creating the next storage file, the oldest file is deleted, and a new file is created and the logging is continued.
Stop	When trying to create the next save file, the error will occur, and the logging operation will be stopped.

Number of file switching records (SAVSWICTMNRECNUM)

Sets the number of records to switch a file to be saved during continuous logging.

In trigger logging, since a file to be saved is switched automatically when the trigger is satisfied, this setting is ignored.

Number of buffers (NUMBUFF)

Set the number of internal data collection buffers. When the creation interval of the data logging files is short, and the error BUSY occurs, increase this setting value. When the setting value is 0, the number of buffers is determined automatically according to the file creation interval.

For details, refer to the following.

☞ Page 535 Internal buffer

Trigger condition (TRIGGERCONDITION)

Sets a trigger condition. Conditions can be created up to 32 and combined by "Trigger type" setting.

Condition type (TYPE)

Condition		Description
Data condition	Comparison	Data is compared with data or constant and a trigger occurs when a condition is satisfied. (=, ≠, ≥, >, ≤, <)
	Value change	A trigger occurs when a value is changed. (↑, ↓, at change)
Fixed cycle		A trigger occurs every specified cycle (ms).
System start		A trigger occurs at system start.
Trigger command		A trigger occurs when System.LoggingRef[1..10].Cd.Trigger is set to TRUE.
Event history		A trigger occurs when an event of the specified event history occurs.

■Data condition

A trigger occurs when the specified data meets the specified condition.

[Comparison]

- =: When the current value of monitored data is equal to the comparison value
- ≠: When the current value of monitored data is not equal to the comparison value
- ≥: When the current value of monitored data is greater than or equal to the comparison value
- >: When the current value of monitored data is greater than the comparison value
- ≤: When the current value of monitored data is less than or equal to the comparison value
- <: When the current value of monitored data is less than the comparison value

[Value change]

- ↑: When the monitored data turns FALSE to TRUE
- ↓: When the monitored data turns TRUE to FALSE
- At change: When the current value of the monitored data changes

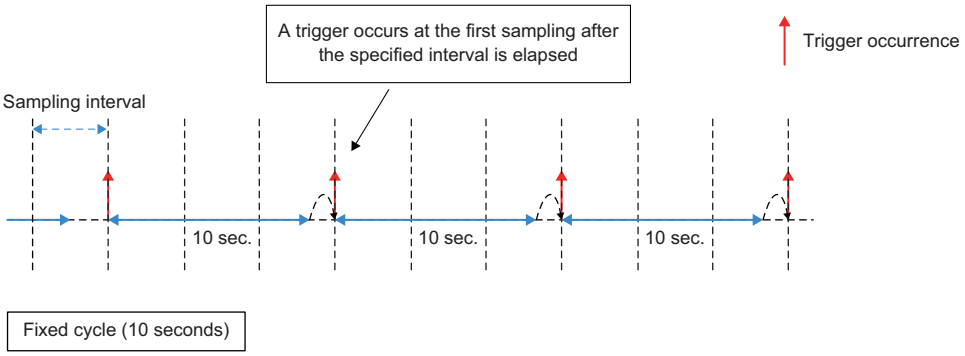
■Fixed Cycle

A trigger occurs in the specified interval after logging starts.

However, when the specified interval in the fixed cycle and the collection timing are not matched, a trigger occurs at the first sampling after the specified fixed cycle interval is elapsed.

The cycle can be specified is 1 to 86400000 [ms].

- When the fixed cycle is specified to 10 seconds



■System start

A trigger occurs at system start (at completion of initialization processing) automatically. When this condition is used, set the start condition of logging setting to "Starts automatically".

■Trigger command

A trigger occurs when System.LoggingRef[1..10].Cd.Trigger is set to TRUE.

■Event history

A trigger occurs when an event of the specified event history occurs.

- Set an event code to be triggered.
- Multiple event codes can be specified using "-" (hyphen) and "," (comma).

Ex.

[When setting event codes from 0x80001000 to 0x80001010 as a trigger condition]

0x80001000-0x80001010

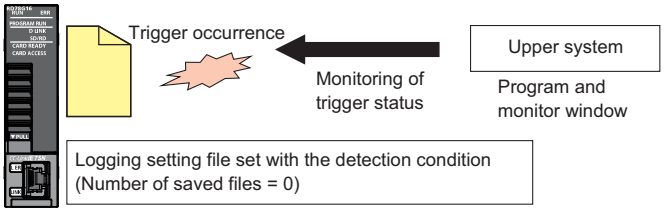
[When setting all event codes as a trigger condition]

0x00000000-0xFFFFFFFF

Event detection

When File save setting is set to "Disabled" in the logging setting, only the trigger detection is executed without saving files. The following can be applied easily by using the trigger detection function of the logging function instead of description of the satisfied judgement processing of various signal conditions (event detection processing) by a user program.

- Set the complicated error condition or program execution status to the trigger condition and utilize it for preventive maintenance.
- Monitor whether the monitor value exceeds the threshold value from the upper system and utilize it for debugging of the program.



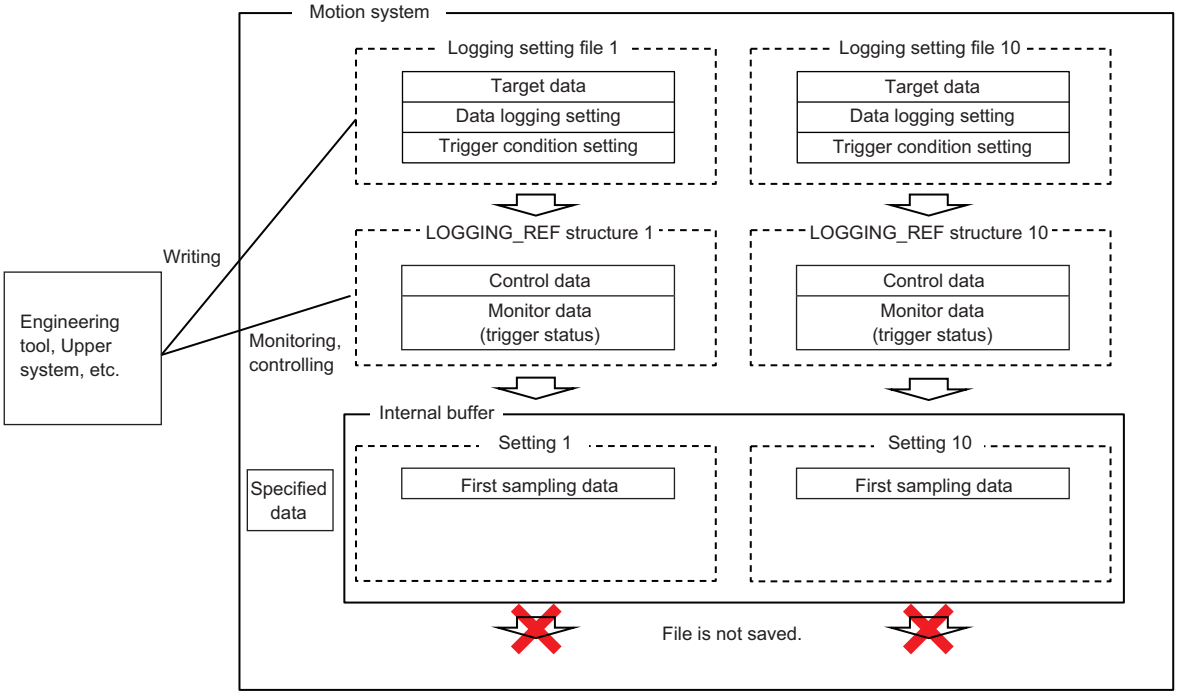
■Procedures of event detection

1. Set as follows in the logging setting and start logging.

- File save setting: Disabled
- Logging type: Trigger logging
- Number of trigger logging: 1 time
- Number of trigger records before trigger: 0
- Number of trigger records after trigger: 1

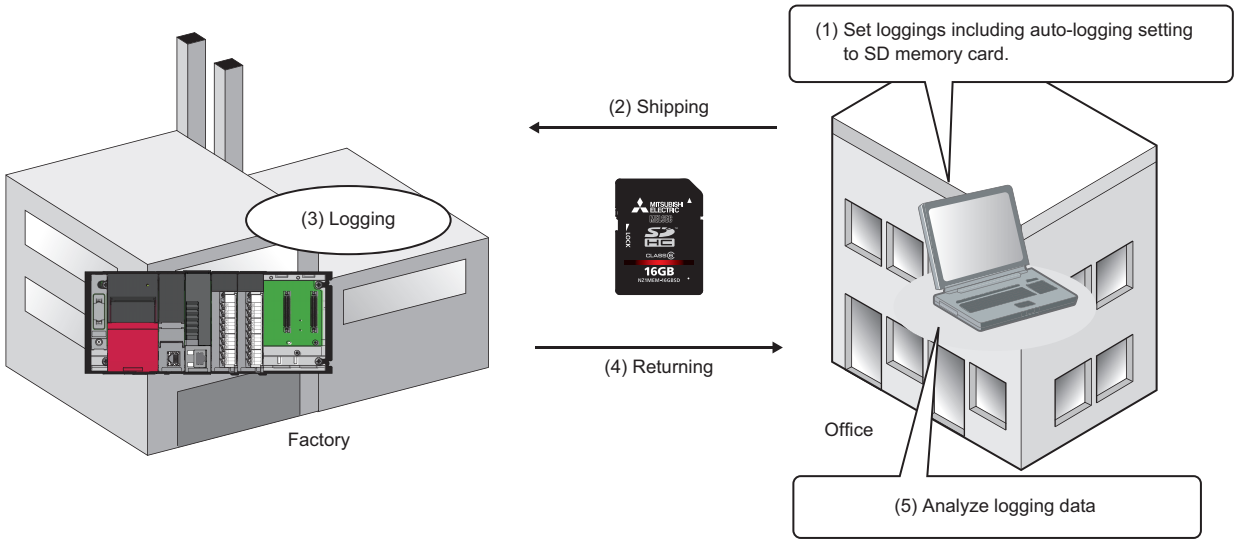
2. When the set trigger condition is detected, System.LoggingRef[1..10].Md.TriggerCount turns 1. This signal is used to control. (From the upper, used for monitoring, program execution condition, etc.)

3. To detect an event continuously, set System.LoggingRef[1..10].Cd.StartLog to FALSE once after a trigger is detected and set to TRUE again.



Auto logging

When inserting an SD memory card, which holds logging setting, into the motion system, the data logging automatically starts based on the logging setting information on the SD memory card.



■How to use auto logging

The following describes how to use auto logging.

1. Set System.PrConst.Logging_AutoLoggingEnable to TRUE in advance and allow the auto logging function.
2. Write the following logging setting file. Set /sdc/LOGGING□□.json or /sdc/LOGGING/LOG□□/LOGGING.json(□□ = 01 to 10) to the writing destination. (When there are both files, the priority is the former.)
   Logging start conditions is set as "Starts automatically".
3. Start the system by inserting the SD memory card prepared in step 2 into the Motion module or insert the Motion module while it is running.
4. System.LoggingRef[1..10].Cd.SettingPath which corresponds to the logging setting file stored in the SD memory card automatically becomes the path of the setting file, System.LoggingRef[1..10].Cd.StartLog becomes TRUE and logging starts. If the logging setting of that number already operates, a new logging does not start and an error occurs. (The logging which has already started is not affected.)

Servo system recorder

The servo system recorder function automatically corrects the position command or actual position at an error occurrence, etc. of the drive unit connected to the servo system by using logging function.

The setting file of the servo system recorder function is automatically generated when using the function. For details, refer to the following.

☞ Page 569 Servo System Recorder

Precautions

The following describes precautions to take when using the logging function.

- When the AND condition is used, the condition at the motion system start is satisfied only once. Therefore, the trigger occurs at the first time only even if the trigger count is set to other than 1.
- The judgement of data condition is executed by the collected data in each set collecting interval.
  When the data condition is specified to the trigger, data is not detected unless the data condition has been satisfied at collecting. Set the condition established time longer than the collecting interval.
- If the 2nd operation cycle is not set when the 2nd operation cycle is selected for the collection interval, the status will be logging error and the data collection will not be executed.
- The write for System.LoggingRef[1..10].Cd.** must not be carried out during the logging operation by the engineering tool.
- After the trigger condition is satisfied, if a new trigger occurs during sampling of records after trigger, a new trigger is ignored. When the sampling of records after trigger is completed and the file is started to be saved, a new trigger is allowed to be received.

The example is shown below.

Ex.

Indicates the operation below.

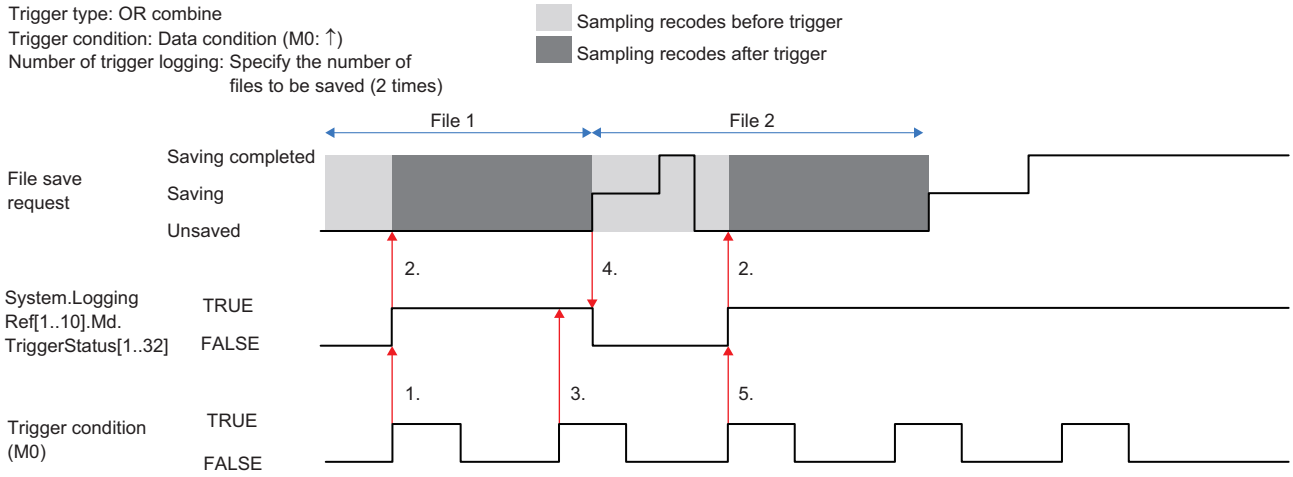
1. System.LoggingRef[1..10].Md.TriggerStatus[1..32] becomes TRUE when the trigger condition is satisfied. (File 1)

2. When the trigger condition is satisfied, sampling records after trigger is executed.

3. The trigger condition occurred during sampling of records after trigger is ignored.

4. When the file save is executed, System.LoggingRef[1..10].Md.TriggerStatus[1..32] becomes FALSE. (Preparation for the next logging operation)

5. System.LoggingRef[1..10].Md.TriggerStatus[1..32] becomes TRUE when the trigger condition is satisfied. (File 2)

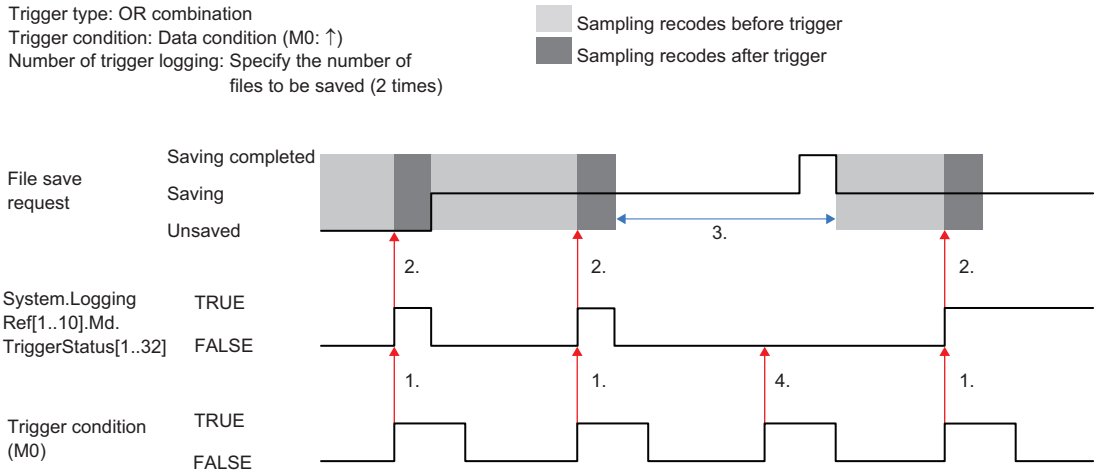


- Saving files can be carried out continuously up to the number specified in the number of buffers (NUMBUFF). If it takes time to save the file when the file size is large and a trigger is established during the time, the trigger is ignored.

Ex.

Indicates the operation for NUMBUFF = 2 below.

1. System.LoggingRef[1..10].Md.TriggerStatus[1..32] becomes TRUE when the trigger condition is satisfied.
2. When the trigger condition is satisfied, sampling records after trigger is executed.
3. The logging whose file size is large is continued 2 times and it takes time to save files.
4. The 3rd trigger established is ignored.



- In logging, the folder which the environmental variable %TEMP% indicates is used as the operation folder. (/ram/temp by default)If the drive free space is insufficient due to many records, etc., consider changing the place of the operation folder. For details, refer to the following.

  ☞ Page 596 FILE CONTROL
- If the file is saved after the data collection, a logging error (error code: 4008H) occurs and stops the logging. By taking following the corrective actions, the error may be avoided or the frequency may be reduced.

- Specify the file storage location of the data logging file to the SD memory card.
- Set file format to JSON.
- Reduce the number of files in the file storage location of the data logging file.
- Set a longer data collection interval.
- Increase the number of buffer. Also, increase the assignment of the system memory (RAM) of the Add-on Logging if necessary.
- Increase the operation time of the motion service processing by executing such as setting longer operation cycle and decreasing the loading of the program execution. (The file saving is processed by the motion service processing.)

Error

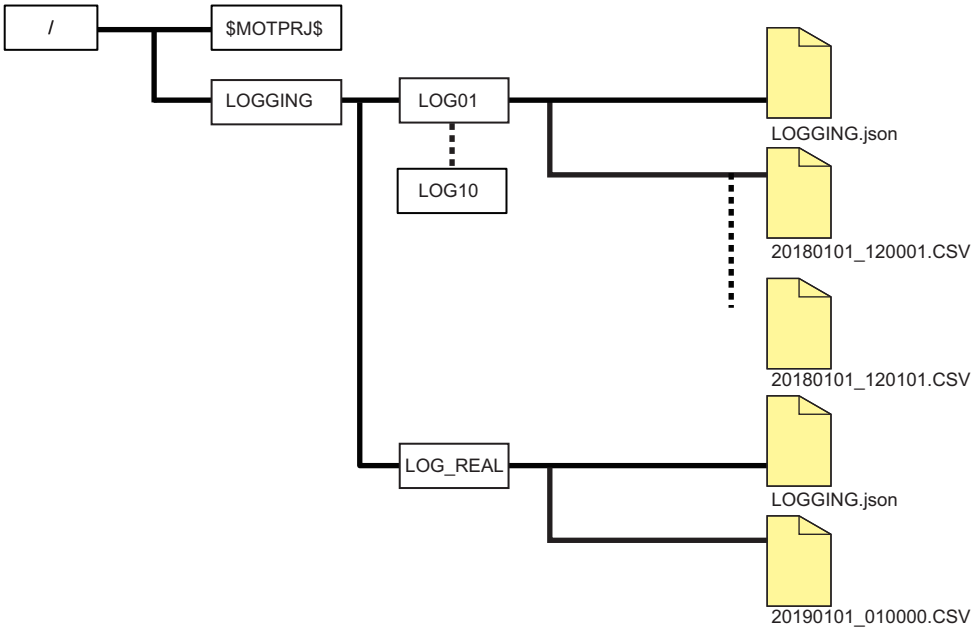
No diagnostic error occurs if an error occurs during logging, and the error event is not recorded in the error history. To check the error status, refer to System.LoggingRef[1..10].Md.ErrStatus.

For the list of logging error codes, refer to the following.

☞ Page 692 List of Logging Error Codes

Folder configuration

The following shows the default folder configuration of logging.



Folder	Data logging: Route of drive/LOGGING/LOG**/ (The ** corresponds to No.1 to 10 of logging setting) Real-time monitor: Route of drive/LOGGING/LOG_REAL/
Logging setting file	LOGGING.json
Data logging file	[SAVFLDNAMEDETLFRMT]/[YYYYMMDD]_[hhmmss].[TFFFILE]

15.2 Data Logging Format

This section describes file format of the data logging.

CSV file data specifications

File information row

Column No.	Column name	Description
1st column	File type	Fixed character "[LOGGING]"
2nd column	Type information_file version	Output file version "RD78G(H)_1" (Initial version)
3rd column	Data type information row number	Number which indicates data type information row: a
4th column	Data name row number	Number which indicates data name row: a + 1
5th column	Data start row number	Number for data starting row: a + 2
6th column	Device comment row number	Number indicates device comment row: a + 3 When the device comment does not exist, it can be omitted.

Setting data information row

The description of the setting file (json) is output. For data details, refer to the followings.

☞ Page 535 Logging setting, ☞ Page 550 JSON File Data Specifications

Data type information row

Data type information row is output in the format of "(Data type)[additional information]".

Column No.	Column name	Description
1st column	Date and time column	Indicates data type of date and time. Fixed character "DATETIME[YYYY/MM/DD hh:mm:ss.ssssssss]"*1
2nd column	Index column	Fixed character "INDEX" which indicates index column Index does not have additional information. [] is not output.
3rd column and later	Data column	Indicates data type of the sampled data.
Last column	Trigger generation information column	The following character string is output regardless of continuous logging/ trigger logging. "TRIGGER["trigger ON string"]"*2 The character string with double quotation "" is fixed character.

*1 Regarding .ssssssss at the end, three digits for information of [ms], three digits for information of [μs], and two digits for information of [ns] are output.

2 "" is output for trigger ON string.

For details of data column are shown below.

Data type of data	Output character for the data type	Description
Bit	BIT	Outputs in the format of "BIT[1;0]".
Word [unsigned]	USHORT	Outputs in the format of "USHORT[DEC.0]".*2
Double word [unsigned]	ULONG	Outputs in the format of "ULONG[DEC.0]".*2
Word [signed]	SHORT	Outputs in the format of "SHORT[DEC.0]".*2
Double word [signed]	LONG	Outputs in the format of "LONG[DEC.0]".*2
Single-precision real number	FLOAT	Outputs in the format of "FLOAT[DEC.7]".*1*2
Double-precision real number	DOUBLE	Outputs in the format of "DOUBLE[DEC.14]".*1*2

*1 The number of digits after the decimal point is not always same as the specified one. (Example) For 1.2345, '1.2345000' is not output.

*2 [DEC.(Number)] shows the numerical value format and the maximum number of fractional part digit. (Example) For [DEC.7], the fractional part digit is seven with decimal notation.

Data name row

Data name row indicates the title character string of each column.

Column No.	Column name	Description
1st column	Date and time column	Indicates title of date and time. Fixed character "TIME[time zone]"
2nd column	Index column	Indicates title of index column. Fixed character "INDEX"
3rd column and later	Data column	Indicates title of data column. Device label (Example: FeedAxis.AxisMntr.Feedcurrent)
Last column	Trigger generation information column	Indicates title of trigger generation information column. Fixed character "Trigger"

Device comment row

The following description is output in the device comment row.

Column No.	Column name	Description
1st column	Date and time column	Blank
2nd column	Index column	Blank
3rd column and later	Data column	Outputs device comment.
Last column	Trigger generation information column	Blank

Data row

Data of each data name is output in the data row.

Column No.	Column name	Description
1st column	Date and time column	Indicates data of date and time. Output in the format of "YYYY/MM/DD hh:mm:ss:ssssssss" with the fixed format.
2nd column	Index column	Indicates data of index. Integer value of the index is output. (When missing, reassigns it from one.)
3rd column and later	Data column	Indicates value of data name. Value according to the data type of data type information row.
Last column	Trigger generation information column	Indicates data of trigger generation information. Character string at occurrence is output in trigger row.

JSON File Data Specifications

Common items (LOGGING)

Item name		Data type	Description
Data logging name	NAME	String	Arbitrary character string Being treated as the setting file for the servo system recorder when it is ServoSystemRecorder or ServoSystemRecorder_*** (Arbitrary character string) For details, refer to the following. ☞ Page 568 Precautions
Logging type*1	TFFLGTP	String	CONT: Continuous logging TRIGGER: Trigger logging
Number of file switching records*1	SAVSWICTMNRECNUM	Number	100 to 100000
Storage location path*1	SAVFLDNAMEDETLFRMT	String	Path name. Blank is available. (Regarded as the same folder of the setting file.)
File format*1	TFFFILE	String	CSV_UTF-16LE: CSV(Unicode)*2 CSV_UTF-16LE_ZIP: ZIP compression CSV(Unicode)*2 JSON: JSON*2 JSON_ZIP: ZIP compression JSON*2
Collecting interval	SMPHSPDTYPE	String	EACHSCAN: Operation cycle TIME: Time specification
Time specification	SMPHSPDTIME	Number	• When SMPHSPDTYPE = EACHSCAN 1 to 3: The nth operation cycle • When SMPHSPDTYPE = TIME 31250 to 86400000000 [ns] Set an unit of ms (floating point) on display
Trigger type (complex condition)	TRGCOMPTYPE	String	OR AND
Number of trigger records before trigger*1	LLNBEFOR	Number	0 to 999999
Number of trigger records after trigger*1	LLNAFTER	Number	1 to 1000000
Total number of rows*1	LLNTOTAL	Number	LLNBEFOR + LLNTOTAL
Number of save files*1	SAVFNUM	Number	0: No upper limits 1 to 65535: Maximum number of files
Operation when exceeds the number of files*1	SAVFNUMTYPE	String	OVERWRITE: Overwrite STOP: Stop
Start condition*2	STARTCONDITION	String	AUTO: Starts automatically CONTINUE: Starts automatically (Stop disabled) MANUAL: Starts by user operation
File save*2	SAVENABLE	Boolean	Saving enabled/disabled of a data logging file TRUE: Save FALSE: Not save
Number of trigger logging*2	TRIGGERTIME	Number	0: Number of save files specification 1 to 32767: Number of times specification

*1 When SAVENABLE = FALSE, setting is not required.

*2 It is a unique setting item for the Motion module (Logger Module does not have this setting).

Trigger condition (TRIGGERCONDITION)

Item name		Data type	Description
No.	NO	Number	1 to 32 (Serial number is not required)
Condition type	TYPE	String	DATACOMP: Data condition (Comparison) DATACHANGE: (Value change) CYCLE: Fixed cycle STARTMODULE: At module start EVENTHISTORY: Event history*1 LOGTRG: At trigger command execution*1
Data name	DATA1	Number	1 to 1024
Condition	DATAOPE	String	EQUAL: = NOTEQUAL: ≠ GREATERTHANEQUAL: ≥ GREATERTHAN: > LESSTHANEQUAL: ≤ LESSTHAN: < RISINGEDGE: ↑*1 FALLINGEDGE: ↓*1
Data/constant	DATA2TYPE	String	DATA CONST
Data name (data name/ constant value)	DATA2	Number	1 to 1024
Constant value (data name/ constant value)	DATA2CONST	String	Constant value Set 0/1 for bit data
Fixed-cycle	CYCL	Number	1000000 to 86400000000000 [ns] Set an unit of ms (floating point) on display
Event code*1	EVENTCODE	String	Event code (0xXXXXXXXX type, Settable in the range)

*1 It is a unique setting item for the Motion module (Logger Module does not have this setting).

Data(DATA)

Item name		Data type	Description
No.	NO	Number	1 to 1024 (Serial number is not required)
Name	NAME	String	Label/device character string
Data name	DATA1	Number	1 to 1024
Data type	TYPE	String	Available only at device specification. (Ignores at label specification) BIT: Bit (BIT) UWORD: Word [unsigned]/Bit column (16 bits) UDWORD: Double word [unsigned]/Bit column (32 bits) SWORD: Word [signed] SDWORD: Double word [signed] FLOAT: Single-precision real number DFLOAT: Double-precision real number SQWORD: Long word [signed]*1 UQWORD: Long word [unsigned]*1
Input method	INPUTMETHOD*1	Number	<For data name restoration by logging setting tool> 0: Directly input (No restoration) 1: Input from label input assistant display (Restores label name) No specification/value other than above: Directly input
Axis type	AXISTYPE*1	String	<For data name restoration by logging setting tool> • Available only at input method "1" "SYSTEM": System "AXIS_REAL": Real drive axis "AXIS_ENCODER": Real encoder axis "AXIS_VIRTUAL": Virtual drive axis "AXIS_VIRTUAL_ENCODER": Virtual encoder axis "AXIS_VIRTUAL_LINK": Virtual linked axis "AXES_GROUP": Axes group No specification/value other than above: Regard as directly input data
Motion unit	MOTIONUNIT*1	String	Specifies the unit information of the motion control data. Outputs the unit information to the logging data file based on this setting value. (Valid only when the label and the instance of the axis, etc. are specified. Other than this, it is ignored.) POSITION: Position VELOCITY: Velocity ACCELERATION: Acceleration JERK: Jerk TORQUE: Torque

*1 It is a unique setting item for the Motion module (Logger Module does not have this setting).

15.3 Logging Data (JSON format)

This section shows the JSON file of the logging result which is compatible with GX LogViewer.

Basic specifications

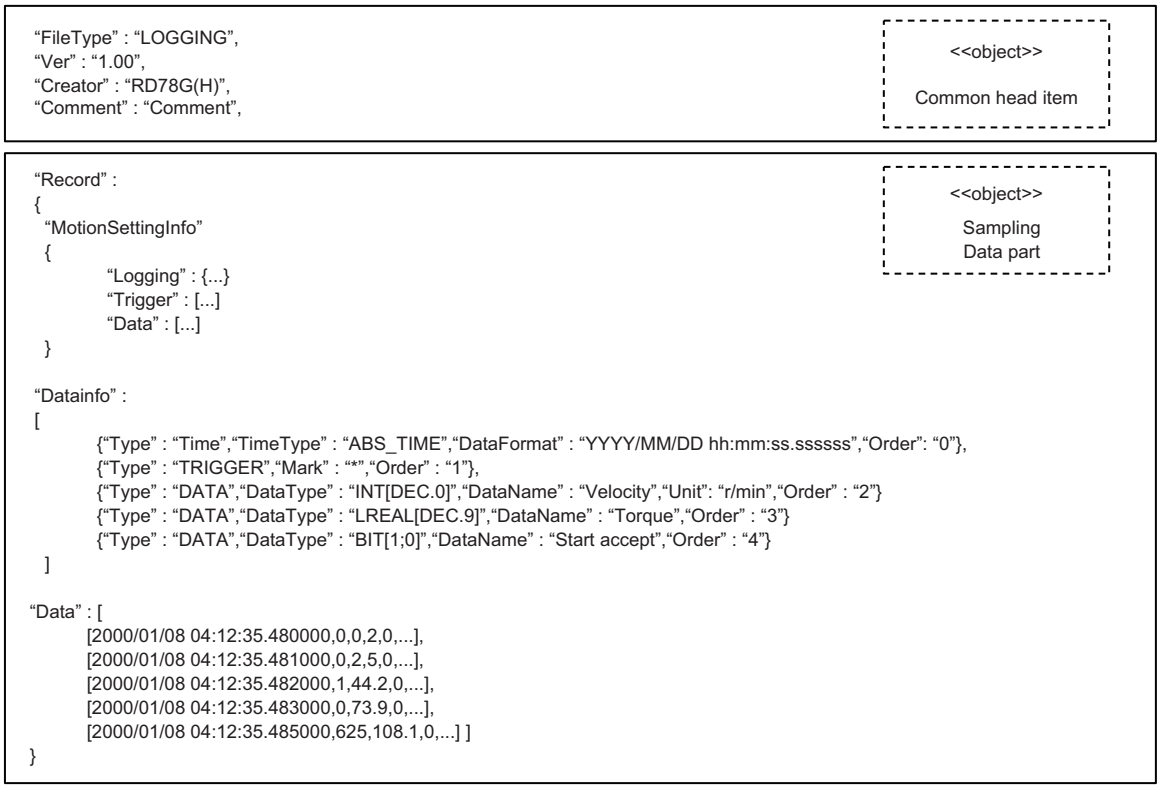
The following JSON files are supported.

- Unicode (UTF-8 with BOM.)
- With or without line break

Whole configuration

The logging setting file configuration is shown below. (The following figure is an image.)

[No group]



The number of data and limitation

GX LogViewer has a limitation to the number of data points and records it can read.

[No group]

- Only 1 record

Item	The muximum number
Number of data points (Datainfo type is DATA)	1024 points (BIT and WORD mixed)
Number of data	1000000 records

However, GX LogViewer cannot read 1024 data points × 1000000 records because much capacity is required. If more than 300M bytes file is read, the error occurs.

Data specification

Common items

Item name	Omission	Key name	Data type	Description
File type	Not possible	FileType	Character string	LOGGING fixed (To make the tool determine the logging data)
File version	Not possible	Ver	Character string	Major version, minor version*1 1.00*2*3
File author*1	Not possible	Creator	Character string	For the Motion module (RD78): RD78G(H) , (a comma) is not allowed.
File comment	Possible	Comment	Character string	Arbitrary character string
Record	Not possible	Record	☞ Page 554 Waveform data (Record)	Waveform data
Status bar color	Possible	StatusBarColor	Color object	0x❶❶❷❷❸❸ ❶❶: Red, ❷❷: Green, ❸❸: Blue Hexadecimal color codes expressed as above

*1 Read is not allowed when the major version of GX LogViewer is other than 1. When its minor version is inconsistent, it reads within the supported range.

*2 This version remains the same even if the device dependent information is changed.

*3 GX LogViewer can read triple-digit minor version.

Waveform data (Record)

Item name	Omission	Key name	Data type	Description
Waveform data name	Not possible	Name	Character string	Arbitrary character string
Waveform data comment	Possible	Comment	Character string	Arbitrary character string (Comment as 1 record)
Device dependent information	Possible	Arbitrary key	☞ Page 556 Device dependent information	Refer to the specification document of each device.
Data column information	Not possible	DataInfo	☞ Page 555 Data column information (dataInfo)	Array of data column information
Data	Not possible	Data	☞ Page 556 Data (Data)	Array of waveform data
Device information	Possible	UnitName	Character string	Outputs when the actual logged device is different with the one outputted on Creator. (If this item does not exists, the device is regarded as the one which is actually logged by Creator.)

Data column information (dataInfo)

Item name	Omission	Key name	Data type	Description
Common				
Type	Not possible	Type	Character string	TIME: Time information TRIGGER: Triggered flag information DATA: Sampling data INDEX: Index information
Order	Arbitrary	Order	Value (starts from 1)	Corresponding column number in the Data part
When the type is time information				
Time information type	Not possible	TimeType	Character string	ABS_TIME: Absolute value
TimeType is ABS_TIME				
Time format	Not possible	DateFormat	Character string	Only when TimeType is absolute value (Ex.) YYYY/MM/DD hh:mm:ss.s
When the type is triggered flag information				
Trigger signal	Not possible	Mark	Character	(Ex.) "*"
When the type is index information (Used for determination of missing data)				
—	—	—	—	—
When the type is sampling data				
Data name	Not possible	DataName	Character string	—
Comment	Possible	DataComment	Character string	—
Data type	Not possible	DataType	Character string	(Corresponding with the type of EdgeCross) BOOL[1;0] <Integer type in decimal format> INT[DEC.0] UINT[DEC.0] DINT[DEC.0] UDINT[DEC.0] LINT[DEC.0]*1 <Integer type in hexadecimal format> UINT[HEX] UDINT[HEX] ULINT[HEX]*1 REAL[DEC.(number of digits after the decimal point)] LREAL[DEC.(number of digits after the decimal point)] REAL[EXP.(number of digits after the decimal point)] LREAL[EXP.(number of digits after the decimal point)] STRING[(data length)]*1 WSTRING[(data length)]*1
Unit	Possible	Unit	Character string	Depents on items
Display color	Possible	LineColor	Value	RGB value

*1 This data format cannot be read by GX LogViewer.

Data (Data)

Array of value column according to data column information

Data type	Display of data type information row	Value range
BOOL type	BOOL[1;0]	1(ON), 0(OFF)
Integer type in decimal format	INT[DEC.0]	-32768 to 32767
	UINT[DEC.0]	0 to 65535
	DINT[DEC.0]	-2147483648 to 2147483647
	UDINT[DEC.0]	0 to 4294967295
	LINT[DEC.0]	-9,223,372,036,847,775,808 to 9,223,372,036,854,775,807
Integer type in hexadecimal format	UINT[HEX]	0 to FFFF
	UDINT[HEX]	0 to FFFFFFFF
	ULINT[HEX]	0 to FFFFFFFFFFFFFFFF
Real number	REAL[DEC.(number of digits after the decimal point)]	-3.40282346E+38 to -1.40129846E-45, 0, 1.40129846E-45 to 3.40282346E+38
	REAL[EXP.(number of digits after the decimal point)]	
	LREAL[DEC.(number of digits after the decimal point)]	-1.7976931348923158E+308 to -4.9406564584124654E-324, 0, 4.9406564584124654E-324 to 1.7976931348923158E+308
	LREAL[EXP.(number of digits after the decimal point)]	
Character string	STRING	—
	WSTRING	—

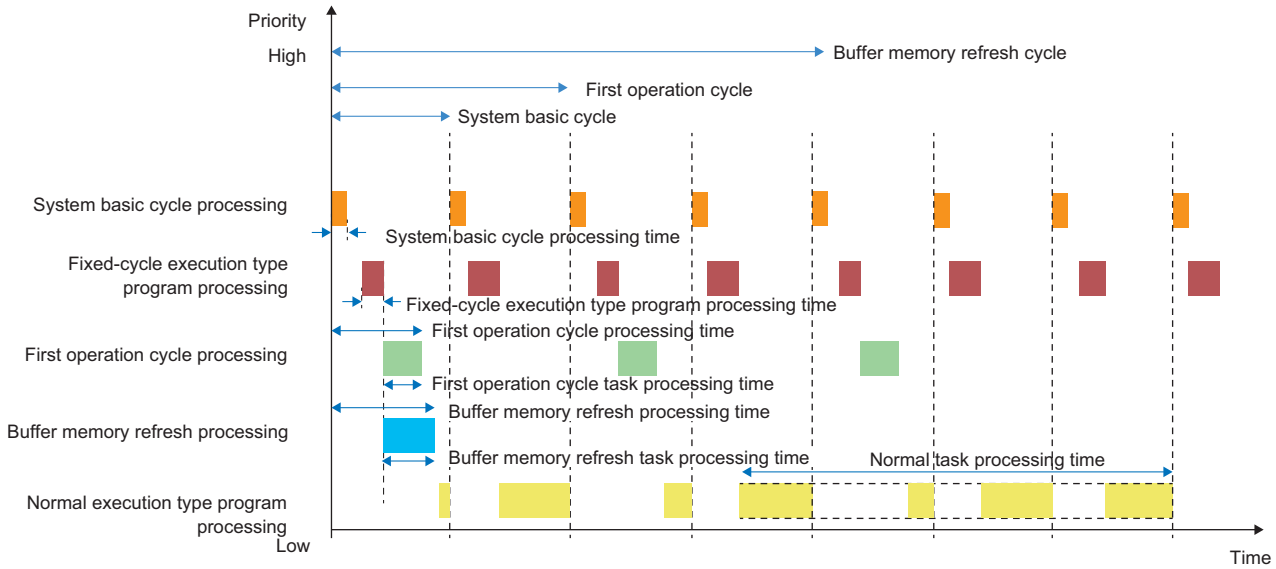
Device dependent information

For the motion system, the key name is MotionSettingInfo.

16 RAS FUNCTIONS

16.1 Execution Time Monitor

This function is used to monitor the operation cycle processing and the normal task execution time.



Relevant variables

Variable/Structure name		Name	Details
System.PrConst.			
	OperationCycle[1..1]	Operation cycle setting	Stores each setting value of first operation cycle. CYCLE_PARAM structure
	BuffermemoryRefreshCycle	Buffer memory refresh cycle setting	Stores each setting value of buffer memory refresh cycle. CYCLE_PARAM structure
System.Md.			
	SystemBaseCycle	System basic cycle monitor	Stores the monitor value of system basic cycle. CYCLE_MONI structure
	OperationCycle[1..1]	Operation cycle monitor	Stores the monitor value of first operation cycle. CYCLE_MONI structure
	BuffermemoryRefreshCycle	Buffer memory refresh cycle monitor	Stores each monitor value of buffer memory refresh cycle. CYCLE_MONI structure
	Program_NormalTaskCycle	Normal task processing time monitor	Stores the monitor value of normal task. CYCLE_MONI structure Limits Cycle: 0 is always stored because the cycle setting is not existed. CycleOver: FALSE is always set because the cycle over detection is not executed.
	WDTerror	WDT error	Becomes TRUE when WDT error is detected.
	SystemBaseCycle_Counter	System basic cycle counter	Stores the system basic cycle counter.
CYCLE_PARAM			
	Cycle	Cycle setting	For details, refer to the following. ☞ Page 109 Operation Cycle
	NumOfCycleOverWngDetectTimes	Number of cycle over warning detections	The number of cycles until the cycle over warning of each cycle is detected can be set. Warning is not detected when 0 is set.
	NumOfCycleOverErrDetectTimes	Number of cycle over error detections	The number of cycles until the cycle over error of each cycle is detected can be set.
	CycleOverErrorType	Cycle over error selection	An assignment for the over error of each cycle can be selected. 2: MinorError (Minor error) 3: ModerateError (Moderate error)
CYCLE_MONI			
	ProcessingTime	Processing time	Stores the processing time of each cycle in a unit of ns.
	MaximumProcessingTime	Maximum processing time	Stores the maximum processing time of each cycle in a unit of ns.
	Cycle	Setting cycle	Stores the cycle setting of each cycle in a unit of ns.
	CycleOver	Cycle over	Becomes TRUE when the cycle over of each cycle is detected.

Control details

System basic cycle

For details on system basic cycle, refer to the following.

☞ Page 109 Operation Cycle

■System basic cycle setting

Stores the system basic cycle setting in System.Md.SystemBaseCycle.Cycle with a unit of nanosecond [ns].

■System basic cycle processing time

Stores the system basic cycle processing time in System.Md.SystemBaseCycle.ProcessingTime with a unit of nanosecond [ns].

■Maximum processing time of system basic cycle

Stores the maximum value of system basic cycle processing time in System.Md.SystemBaseCycle.MaximumProcessingTime with a unit of nanosecond [ns].

■System basic cycle over

For details, refer to the following.

☞ Page 561 System basic cycle over check

Normal task

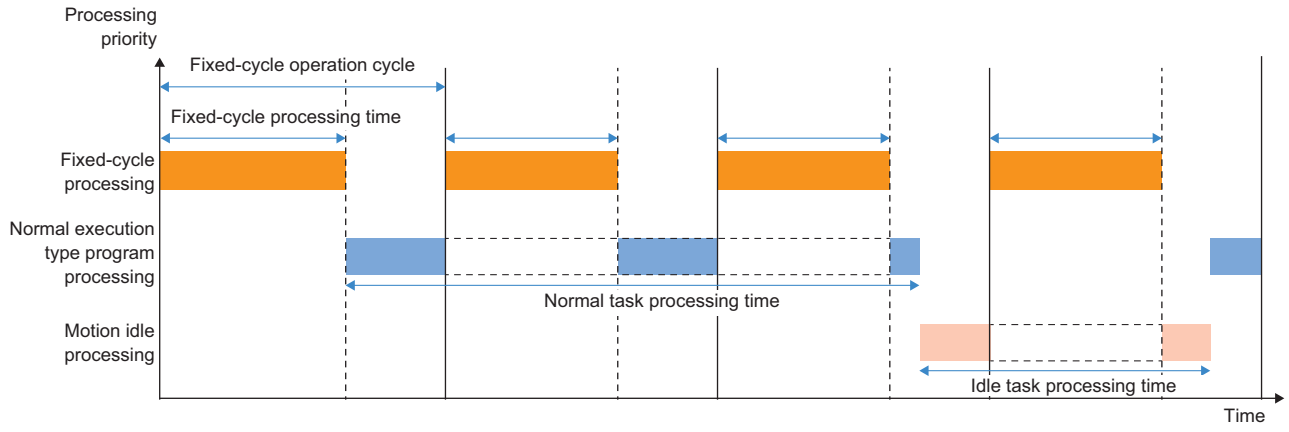
■Normal task processing time

Stores the normal task processing time in System.Md.Program_NormalTaskCycle.ProcessingTime with a unit of nanosecond [ns]. It includes the task processing time whose priority is higher than the normal task.

[Normal execution type program processing]

For details on the processing, refer to "Execution of programs" in the following manual.

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



■Maximum processing time of normal task

Stores the maximum value of the normal task processing time in System.Md.Program_NormalTaskCycle.MaximumProcessingTime with a unit of nanosecond [ns].

First operation cycle

For details, refer to the following.

☞ Page 109 Operation Cycle

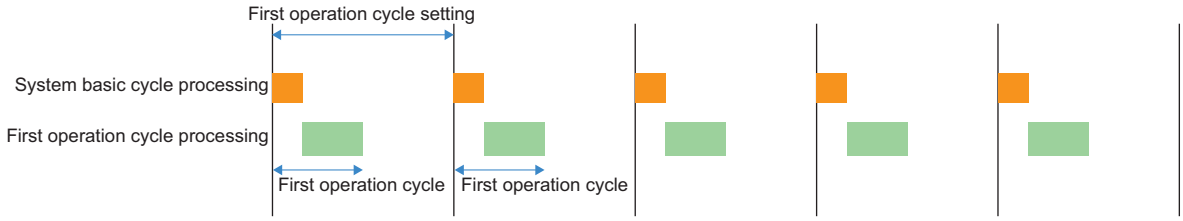
■First operation cycle setting

Stores the first operation cycle setting in System.Md.OperationCycle[1..1].Cycle with a unit of nanosecond [ns].

■First operation cycle processing time

Stores the first operation cycle processing time in System.Md.OperationCycle[1..1].ProcessingTime with a unit of nanosecond [ns].

It includes the task processing time whose priority is higher than the first operation cycle.



■Maximum processing time of first operation cycle

Stores the maximum value of the first operation cycle processing time in System.Md.OperationCycle[1..1].MaximumProcessingTime.

■First operation cycle over

For details, refer to the following.

☞ Page 561 Operation cycle over check

Buffer memory refresh

For details, refer to the following.

☞ Page 109 Operation Cycle

■Buffer memory refresh cycle setting

Stores the buffer memory refresh cycle setting in System.Md.BufferMemoryRefreshCycle.Cycle with a unit of nanosecond [ns].

■Buffer memory refresh cycle processing time

Stores the buffer memory refresh cycle processing time in System.Md.BufferMemoryRefreshCycle.ProcessingTime with a unit of nanosecond [ns]. It includes the task processing time whose priority is higher than the buffer memory refresh cycle.

■Maximum processing time of buffer memory refresh cycle

Stores the maximum value of the buffer memory refresh cycle processing time in System.Md.BufferMemoryRefreshCycle.MaximumProcessingTime with a unit of nanosecond [ns].

■Buffer memory refresh cycle over

For details, refer to the following.

☞ Page 562 Buffer memory refresh cycle over check

Processing time over check

System basic cycle over check

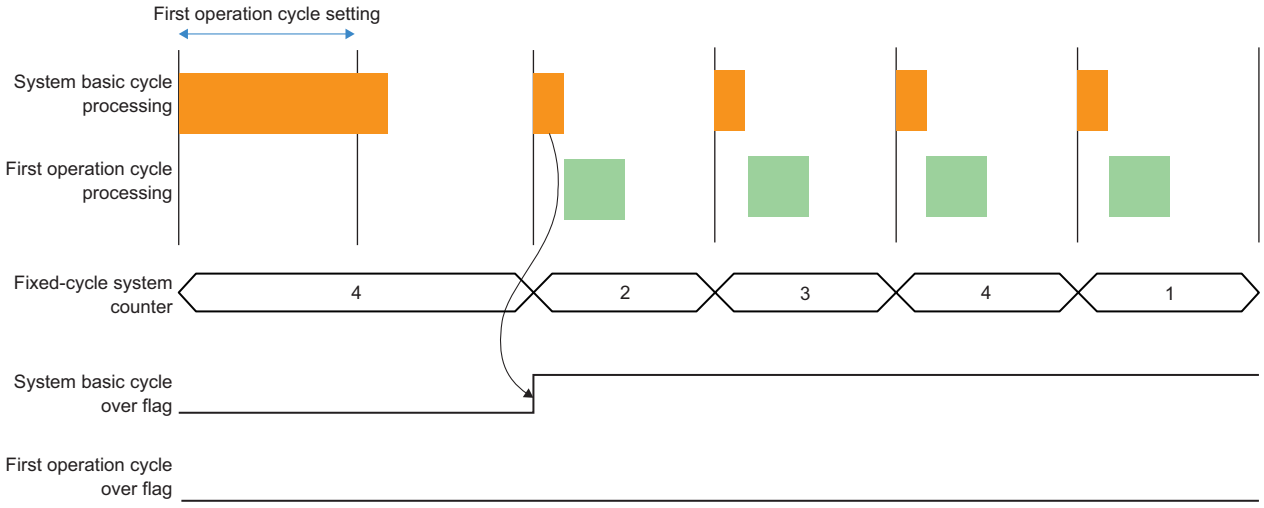
When the system basic cycle processing is not completed before the start of the next system basic cycle processing, the system basic cycle over flag becomes TRUE.

When the system basic cycle over is detected, the number of the fixed-cycle program should be reviewed.

If the system basic cycle exceeds the set cycle at the start of the first operation cycle, a new operation cycle is not executed in this cycle and it is executed in the start timing of the next cycle.

To set the system basic cycle over flag to FALSE, operate one of the followings.

- Power OFF→ON and reset
- Reset by user program

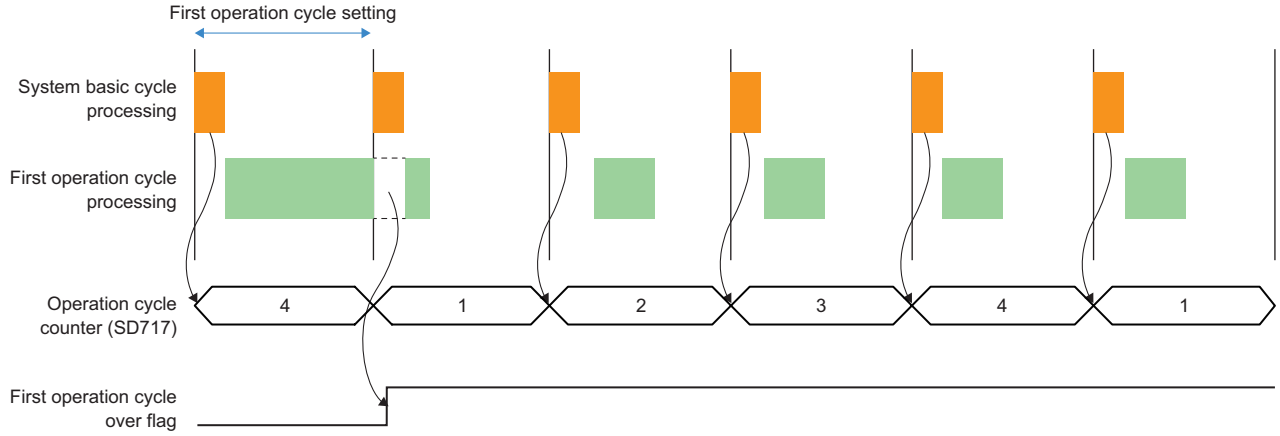


Operation cycle over check

When the first operation cycle has exceeded the set cycle, the first operation cycle over flag becomes TRUE. When the operation cycle over is detected, the number of the fixed-cycle program or the operation cycle setting should be reviewed, or set the greater value for the cycle over detection number or the operation cycle setting.

To set the operation cycle over flag to FALSE, operate the following.

- Power OFF → ON and reset



Point

- When the operation cycle over flag is detected, correct the positioning content or increase the operation cycle.

Buffer memory refresh cycle over check

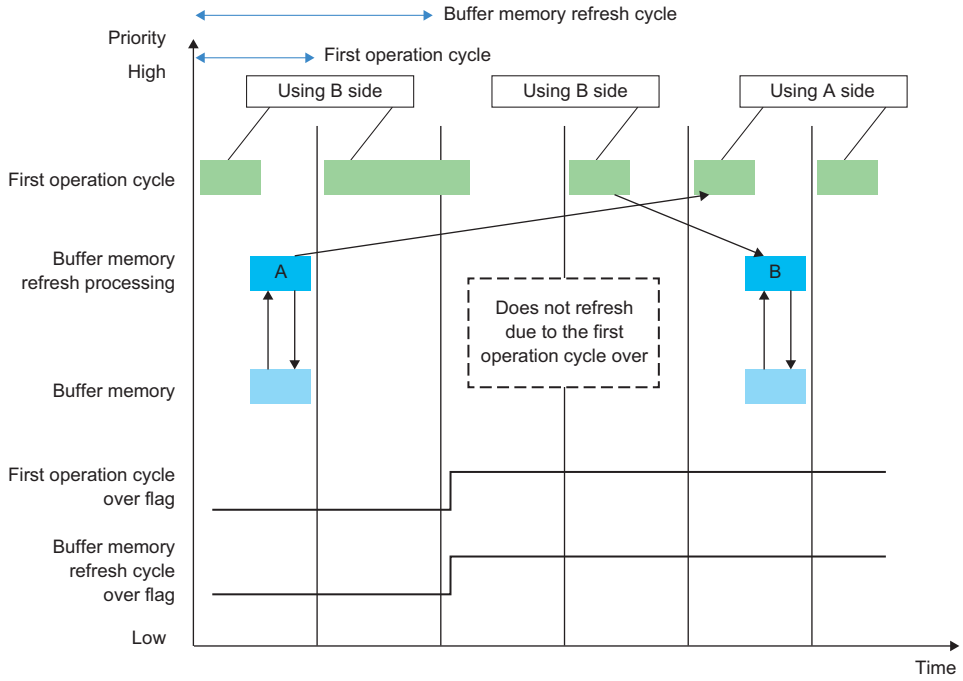
When the buffer memory refresh cycle is not completed before the start of the next cycle, the buffer memory refresh cycle over flag becomes TRUE.

When the buffer memory refresh cycle over is detected, correct the buffer memory refresh number or increase the buffer memory refresh cycle setting or the cycle over error detection number.

Unless one or more of the processing of system basic cycle, first operation cycle, second operation cycle or buffer memory refresh is completed, the processing of buffer memory refresh cycle is skipped in the cycle.

To let the buffer memory refresh cycle over flag be FALSE, perform the following operation.

- Power OFF → ON and reset



Errors

WDT error detection

When the processing is stopped because the operation cycle or other processing is error, a WDT error occurs 1 second later.

When the WDT error is detected, the motion system stops all axes immediately and a moderate error occurs.

Cycle over warning and error detection

When the number of cycle over of each cycle reaches the set cycle over warning/error detection number, the cycle over warning/the error occur.

When the same values are set to cycle over warning detection number and cycle over error detection number, a warning does not occur. And, when 0 is set to the detection number of the cycle over warning, a warning does not occur.

The assignment of over error of each cycle can be selected.

When a minor error is selected, the deceleration stop is executed. When a moderate error is selected, the immediately stop is executed.

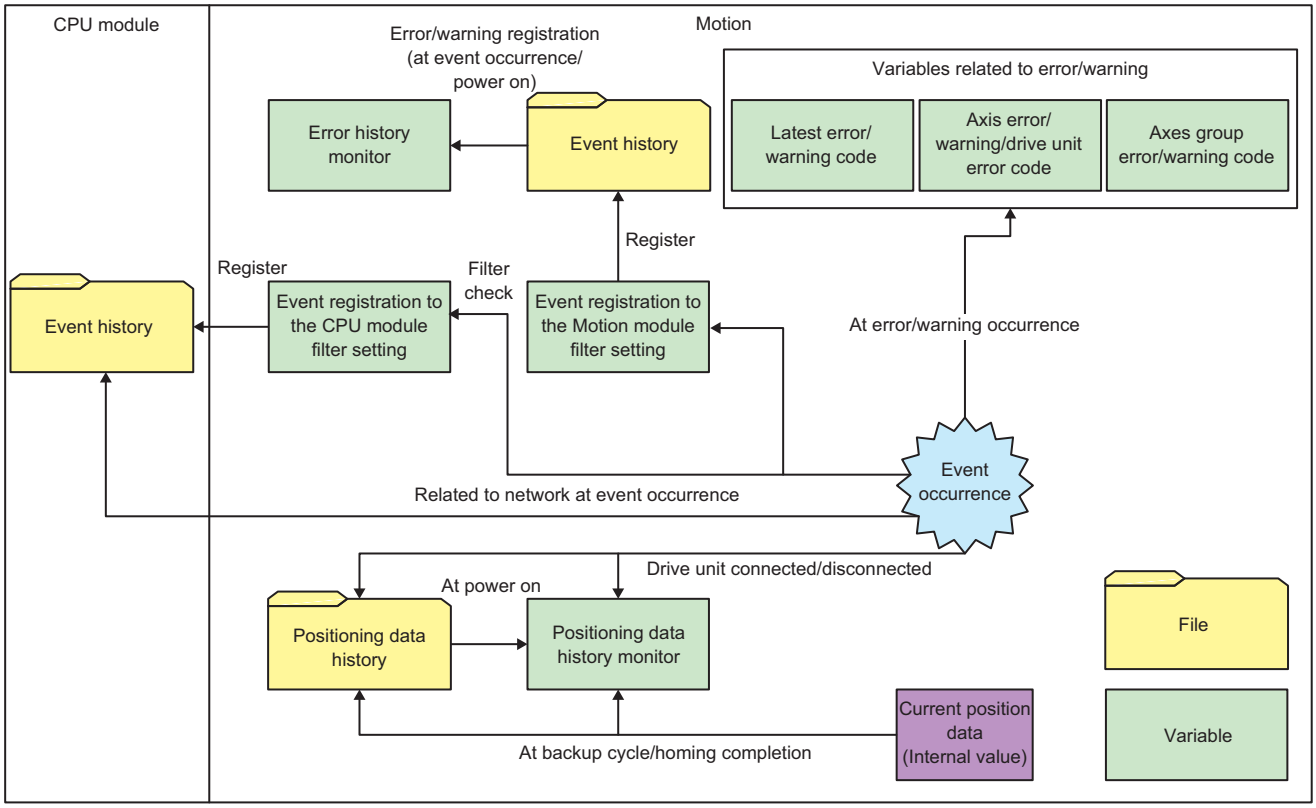
⚠CAUTION

When the deceleration stop cannot be executed because of such as system unstable or hang-ups, a moderate error occurs by WDT error detection. When a minor error is selected, the operation may keep until the WDT error detection. Ensure safety by a user.

16.2 History Data

- History data contains "Event history" and "Positioning data history".
- Positioning data history is used to save such as positioning data of each axis as history at event occurrence related to the absolute position control.
- Event history is used to save the error information and the operation for the module as an event in the CPU module and the motion system. In each event, the records of the CPU module and motion system can be set. The store destination of event records can be set according to event frequency or priority.

Figure related to history data



Relevant variables

Event history filter setting

Variable/Structure name		Name	Details
System.PrConst.			
	EventHistoryMotion_Exclude	Motion event history excluded event	Sets a event code which does not register an event to the motion system.
	EventHistoryMotion_Path	Motion event history path	Sets the store destination of event history file.
	EventHistoryMotion_Capacity	Motion event history capacity	Sets the size of event history file.
	EventHistoryCpu_Exclude	Event excluding CPU event history	Sets a event code which does not register an event to the CPU module.
System.Cd.			
	EventHistoryMotion_Clear	Motion event history clear execution request	Carries out execution requests of the event history clear.
System.Md.			
	EventHistoryMotion_Path	Motion event history path	Stores the store destination (absolute path) of event history file.
	EventHistoryMotion_ClearStatus	Motion event history clear execution status	Displays the execution status of the event history clear. 0: Ready (Waiting for the request) 1: Executing (Executing) 2: Done (Execution completed) -1: Error_ (Error occurrence)

Positioning data history monitor

There is no positioning data history monitor.

Event history function

- This function is used to save the data such as the error detected by the motion system, the operation executed to the module, the event related to the motion control of start and stop as event history.
- The occurrence history of the information such as the saved operation and error can be checked in chronological order.
- The data is saved as event history, so that the history can be checked even if the module turns OFF or resets.
- Using this function enables exploration of failure cause occurred in equipment and device, check of the control data update status of CPU module, and detection of illegal access.
- The event history is always collected regardless of the operation status of Motion module. However, it may not be collected in the cases such as major error of the module, base module error, or cable error.
- The event which occurred in the Motion module and is not set to be filtered is registered in the event history of the control CPU module. For event history of a CPU module, refer to the following.
  MELSEC iQ-R CPU Module User's Manual (Application)

Event occurred in the motion system

The items saved in the event history are shown below.

For details on the event, refer to the following.

☞ Page 693 List of Event Codes

Event type	Category	Details	Event item	Event code
System	Error	Not applicable	Not applicable	—
	Warning	Not applicable	Not applicable	—
	Information	An event on the right is detected in the Motion module.	☞ Page 693 System	00000H to 007FFH
Security	Error	Not applicable	Not applicable	—
	Warning	Not applicable	Not applicable	—
	Information	An event on the right related to the security is detected in the Motion module.	☞ Page 694 Security	10000H to 17FFFH
Operation	Error	Not applicable	Not applicable	—
	Warning	Not applicable	Not applicable	—
	Information	Operation by the user on the right is detected in the motion system.	☞ Page 694 Operation	20000H to 27FFFH

■MCFB execution and stop event

- The MCFB execution event can be monitored.
- Restart and multiple start by buffer mode also can be monitored as an event.
- Monitoring the execution event enables check such as whether each operation has started in intended order at the system startup, etc.
- The stop event when the axis stops caused by stop factor input can be monitored.
- Monitoring the stop event enables check of the stop factor caused when the unintended stop has been executed, etc.

Event history file of the Motion module side

- Event history data of the Motion module side is saved into the drive in the motion system as a file.
- Event history is saved to EVENT.LOG in the storage destination specified by System.PrConst.EventHistoryMotion_Path. The file capacity is specified in System.PrConst.EventHistoryMotion_Capacity in K bytes.
  When System.PrConst.EventHistoryMotion_Path is blank, the event history is saved to the latch drive.
- When the storage destination is the latch drive, the setting of the system memory (backup RAM) size and System.PrConst.EventHistoryMotion_Capacity whichever is smaller amount is the event history file size. To change the file capacity, confirm and change the system memory (backup RAM) size as well. To increase the file capacity, it is recommended to specify the storage destination to an SD memory card. For details on the system memory (backup RAM) size, refer to the following.
  ☞ Page 121 System Memory Settings
- If System.PrConst.EventHistoryMotion_Path and System.PrConst.EventHistoryMotion_Capacity are changed, the event registered before the change cannot refer.
- When saving the event history file fails, the error "Event History File Incorrect" (error code: 1C85H) occurs and the event history file update stops.
- When events frequently occur, the event history storage area in the motion system might be full of event information which has not been collected. If a new event occurs in this state, the motion system discards the event information and outputs the message "*HST.LOSS*" (The event information has been erased.) which indicates the event information has been erased.
- Event history file is updated periodically. When the power of the motion system is turned OFF or the system is reset before saving the event information to the event history file, the unsaved event information is erased.

■Display of the event history

The event history of the Motion module side can be displayed on the engineering tool (The motion control setting function of GX Works3). For the details about the operating procedures and how to interpret the displayed information etc., refer to the following.

📖GX Works3 Operating Manual

■Clearing the event history

[Clearing by the engineering tool]

The event history can be cleared by the engineering tool (The motion control setting function of GX Works3). For the details about the operating procedures, etc., refer to the following.

📖GX Works3 Operating Manual

[Clearing by the event history clear request]

- The event history clear is the function to clear the event history by setting System.Cd.EventHistoryMotion_Clear for TRUE.
- System.Md.EventHistoryMotion_ClearStatus turns "1: Executing" during the event history clear execution.
- System.Md.EventHistoryMotion_ClearStatus turns "2: Done" as the event history clear is completed. When executing the event history clear again, set System.Cd.EventHistoryMotion_Clear for FALSE and execute the clearing after confirming that System.Cd.EventHistoryMotion_Clear turns "0: Ready".
- When the event history clear is failed, System.Md.EventHistoryMotion_ClearStatus turns "-1: Error". When turning to "-1: Error", System.Cd.EventHistoryMotion_Clear set to FALSE temporarily and after a while, carry out the event history clear again after confirming that System.Md.EventHistoryMotion_ClearStatus turns "0: Ready".

The event history file of the CPU module side

The event history data of the CPU module side is saved into the drive of the CPU module as a file.
For the details about the event history of the CPU module, refer to the manual of the CPU module.

■Display of the event history

The event history of the CPU module side can be displayed on the engineering tool (GX Works3). For the details about the operating procedures and how to interpret the displayed information, etc., refer to the following.
GX Works3 Operating Manual

■Clearing the event history

[Clearing by the engineering tool]
The event history can be cleared by the engineering tool (GX Works3). For the details about the operating procedures, etc., refer to the following.
GX Works3 Operating Manual

Filter setting

- The filter setting enables to specify whether the event detected in the motion system is registered in event history.
  For example, when an error is registered to the filter setting, the error processing is executed at the error detection.
  However, the error event is not registered in the event history.
- The following two settings are available for filter setting.

■Filter type

Setting	Details
Event registration in the motion system	Specifies the event code not to be registered in the event history of the motion system at the event detection.
Event registration in the CPU module	Specifies the event code not to be registered in the event history of the CPU module at the event detection.

■Filter setting and event history registration

The following shows the registration result of event history of each module by the filter settings of event registration of the motion system and the CPU module.

Filter setting	Event history
Excluded	Not registered
Other than the above	Registered

■Default value of filter setting

Setting	Filter setting
Event registration in the motion system	None
Event registration in the CPU module	0x700-0x7ff

■Filter setting method

- The filter can be set for each event code.
- The filter setting can be set by the following.

System.PrConst.EventHistoryMotion_Exclude and System.PrConst.EventHistoryCpu_Exclude

Ex.
[For the setting when the event codes 0x80001000 and 0x80001010 are not to be registered in event history]
0x80001000,0x80001010

- The range can be specified by tying event codes using "-" (hyphen).

Ex.
[For the setting when the event codes from 0x80001000 to 0x80001010 are not to be registered in event history]
0x80001000-0x80001010
[For the setting when all event codes are not to be registered in event history]
0x00000000-0xFFFFFFFF
The filter setting is enabled at the power OFF of the motion system, reset, or PLC ready [Y0] ON, after writing in the motion system.

Positioning data history

- The positioning data history of each axis can be monitored by the engineering tool.
- The related positioning data is stored in the history when the following operations are executed.
  At homing completion: 1 time
  At axis connection/axis disconnection: 20 times
- For the positioning data at the axis disconnection, the data at the final backup is set. When the backup data does not exist or the backup data is an error, 0 is set for all.
- The positioning data history is stored in PosHist.bin of the root path.
- The file size of the positioning data history can be calculated with the following formula.
  The positioning data file size (byte) = 110 + (the number of the setting axis × 2126)

Precautions

- The history data in the Motion module can record and monitor the event occurrence time in a unit of ns. However, the event history saved in the CPU module is in a unit of ms (round down 1 ms or less).
- The occurrence order and saved order of events may not match in the event recorded in the event history. When the event history is displayed in the engineering tool, the history can be checked in the event occurrence order by sorting the occurrence time.
- The event which occurs frequently may short the memory life of the event history saving destination. It is recommended that unnecessary events are filtered.
- When the axis setting is not existed at the power ON, the positioning data history data of nonexistent axis is deleted.
- To clear the positioning data history, delete the positioning data history file by file transfer function and turn the power OFF to ON or reset the module.
- When the file version of the positioning data history is different, the warning "Position Data History Update Disabled Warning" (warning code: 0F03H) occurs and the content of the positioning data history file is cleared. After that, the absolute position control data for one time is read, and the positioning data history file is written.
- When the drive memory of root path lacks its capacity, the warning, "Position Data History Update Disabled Warning" (warning code: 0F03H) occurs and the positioning data contents are cleared. After that, the absolute position control data for one time is read and the positioning data file is written. Also, if the memory extremely lacks its capacity and the file can not be created, the positioning data history file is not created.
- The No.1 CPU clock data is the standard for the Motion module. Therefore, the time and date of the event that occurred during initial processing may be different from the actual time and date. Also, the time and date of event occurrence already saved in the event history cannot be corrected even if clock data is received from the CPU module.

16.3 Servo System Recorder

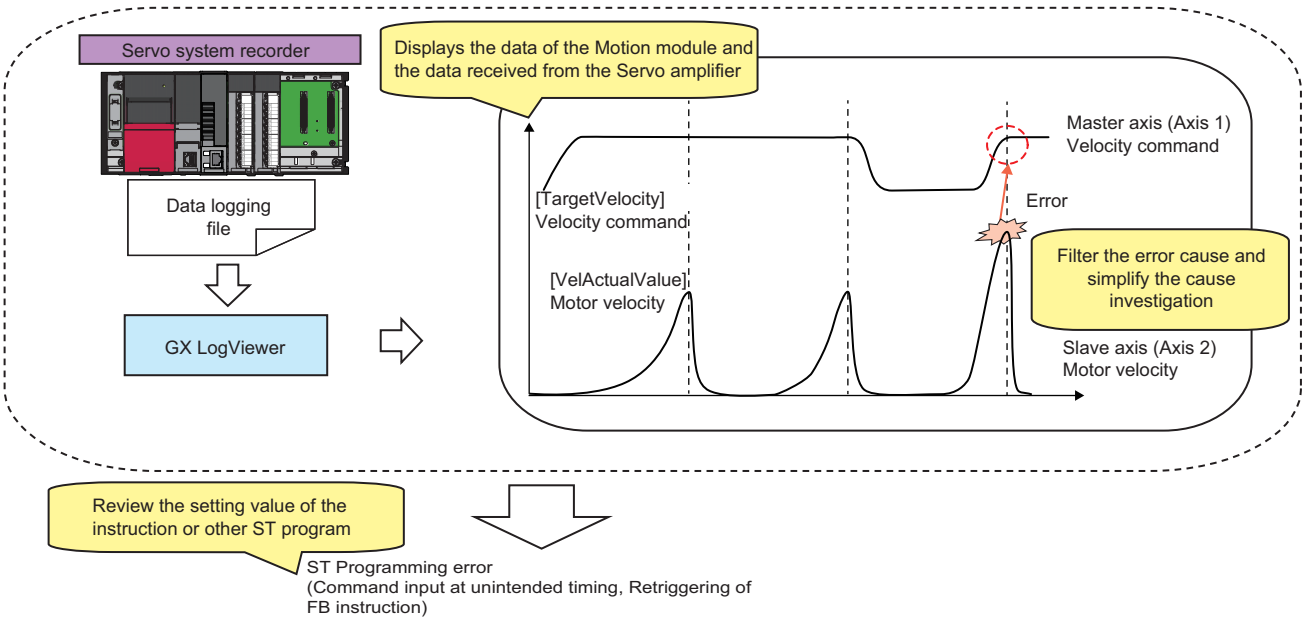
The servo system recorder generates a logging setting file which is optimal for analyzing of the error factors when Add-on ServoSystemRecorder Monitor is enabled. The logging setting file continuously monitors the error status of the motion system and supported slave devices. When an error is detected, the status before and after occurrence is recorded for a certain period of time and saved in the data logging file.

Logging setting file editing and data logging file analysis are possible from the engineering tool.

The collecting target is the real drive axis only. The collected data helps to investigate causes of factor which cannot be solved easily just by information of "Event history".

Ex.

When an error occurs to slave devices which are connected to the slave axis in cam control.



Operation of this function for each system state

○: Possible, △: Possible (Restricted), ×: Not possible

Status	Operation availability
STOP	○
RUN	○
Moderate error	△ (Not possible if ready [X0] does not turn TRUE)
Major error	×

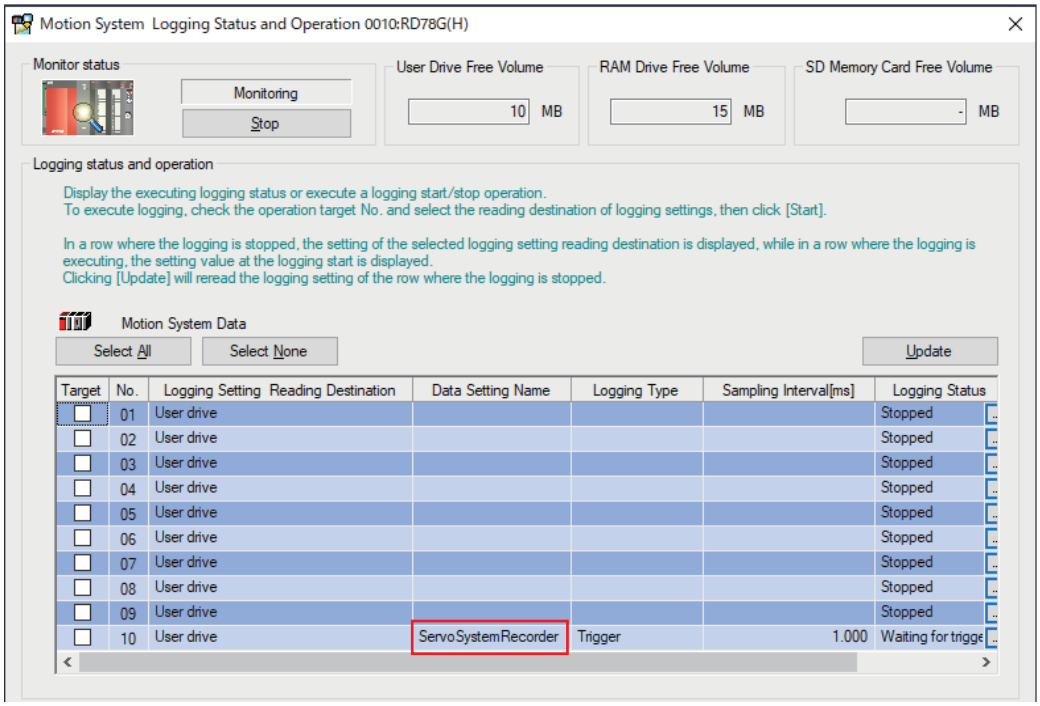
Relevant variables

Variable/Structure name	Name	Details
System.PrConst.		
Addon_ServoSystemRecorder	Add-on ServoSystemRecorder Parameter	ADDON_PARAM structure
Logging_AutoLoggingEnable	Auto Logging Enabled	Sets "Disabled" and "Enabled" of the auto logging function when logging setting files are stored in the SD memory card. 0: Disabled 1: Enabled For details on the auto logging, refer to the following. ☞ Page 544 Auto logging
System.Md.		
Addon_ServoSystemRecorder	Add-on ServoSystemRecorder Monitor	ADDON_MONI structure
ServoSystemRecorder_Enable	Servo System Recorder Enabled	Turns TRUE when the logging collection is operating normally by the servo system recorder.
ADDON_PARAM		
RamSizeMax	Maximum RAM Size	Specifies the maximum amount of system memory (RAM) used in the Add-on ServoSystemRecorder in K bytes. When the operation cycle over is detected during the logging, increase the maximum RAM size of the add-on logging parameter.
BackupRamSizeMax	Maximum Backup RAM Size	Specifies the maximum amount of system memory (backup RAM) used in the Add-on ServoSystemRecorder in K bytes. When backup memory lacks its capacity during the logging, increase the maximum backup RAM size of the add-on logging parameter.
ADDON_MONI		
RamUsage	RAM Usage	Stores current amount of system memory (RAM) used in the Add-on ServoSystemRecorder in K bytes.
RamMaxUsage	RAM Usage Maximum	Stores the maximum amount of system memory (RAM) used in the Add-on ServoSystemRecorder in K bytes.
BackupRamUsage	Backup RAM Usage	Stores the current amount of system memory (backup RAM) used in the Add-on ServoSystemRecorder in K bytes.
BackupRamMaxUsage	Backup RAM Usage Maximum	Stores the maximum amount of system memory (backup RAM) used in the Add-on ServoSystemRecorder in K bytes.
Version	Version	Stores the version information of the Add-on ServoSystemRecorder.

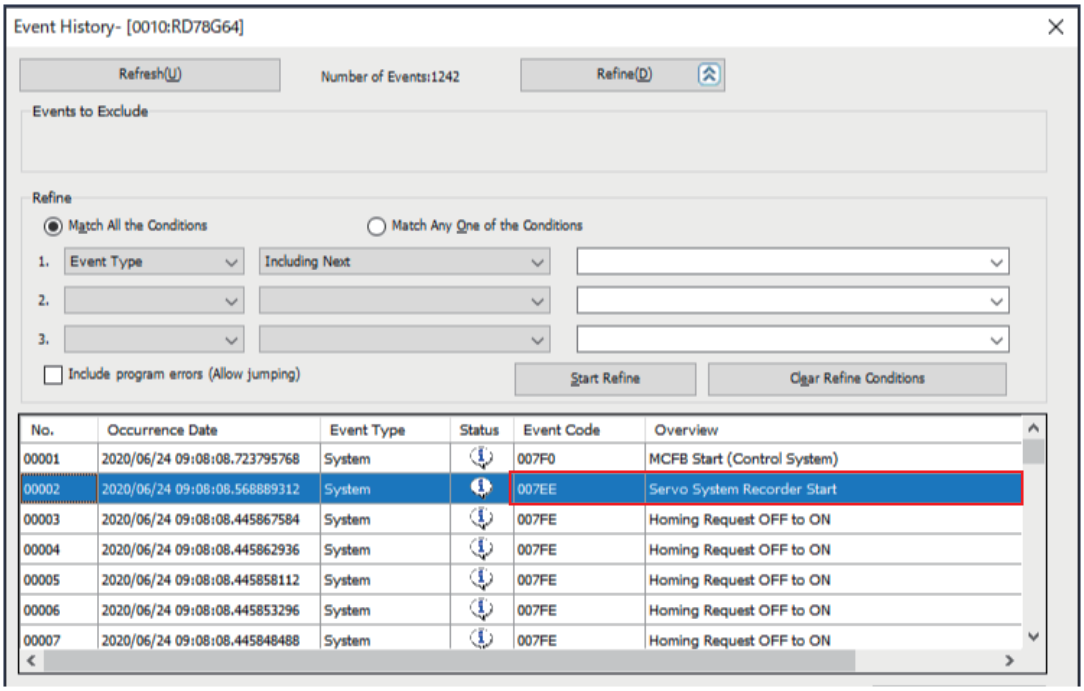
Control details

Logging setting file auto generation

This add-on reads the axis setting in the project at the start of the motion system and automatically generates a logging setting file with collected data reflected with a number of axis setting required for error analysis.
The logging setting file is automatically generated every time the motion system is started so the data will be reflected at the start of the motion system even if the axis setting in the project is changed.
The data logging name which will be added in the file of the generated logging setting file will be "ServoSystemRecorder".
When the name starts with "ServoSystemRecorder", the data in the logging setting file is overwritten every time the motion system is started. When the name starts other than "ServoSystemRecorder", the data in the logging setting file is not be overwritten and the warning "Servo System Recorder Setting File Creation Disabled Warning" (warning code: 0F13H) occurs and the servo system recorder will not start.
The auto-generation can be confirmed if "ServoSystemRecorder" is shown on the "Motion System Logging Status and Operation" window as follow.



Also, the auto-generation completion can be confirmed if "007EE: Servo System Recorder Start" is shown in the event history.



Precautions

- If the real drive axis does not exist in the project, the logging setting file is not generated.
- When starting the motion system without an SD memory card being inserted, the logging setting file is generated in the user drive. When the motion system is restarted with an SD memory card being inserted, both the user drive and the SD memory card will be in the state that the logging setting file is being generated. In this case, the logging is executed by prioritizing the logging setting saved in the SD memory card.

Restriction

If the add-on is disabled, the logging setting file with the data logging name which starts from "ServoSystemRecorder" does not execute the operation. If the file is not needed, delete it.
For how to delete files, refer to the following.
☞ Page 582 How to delete the logging setting file

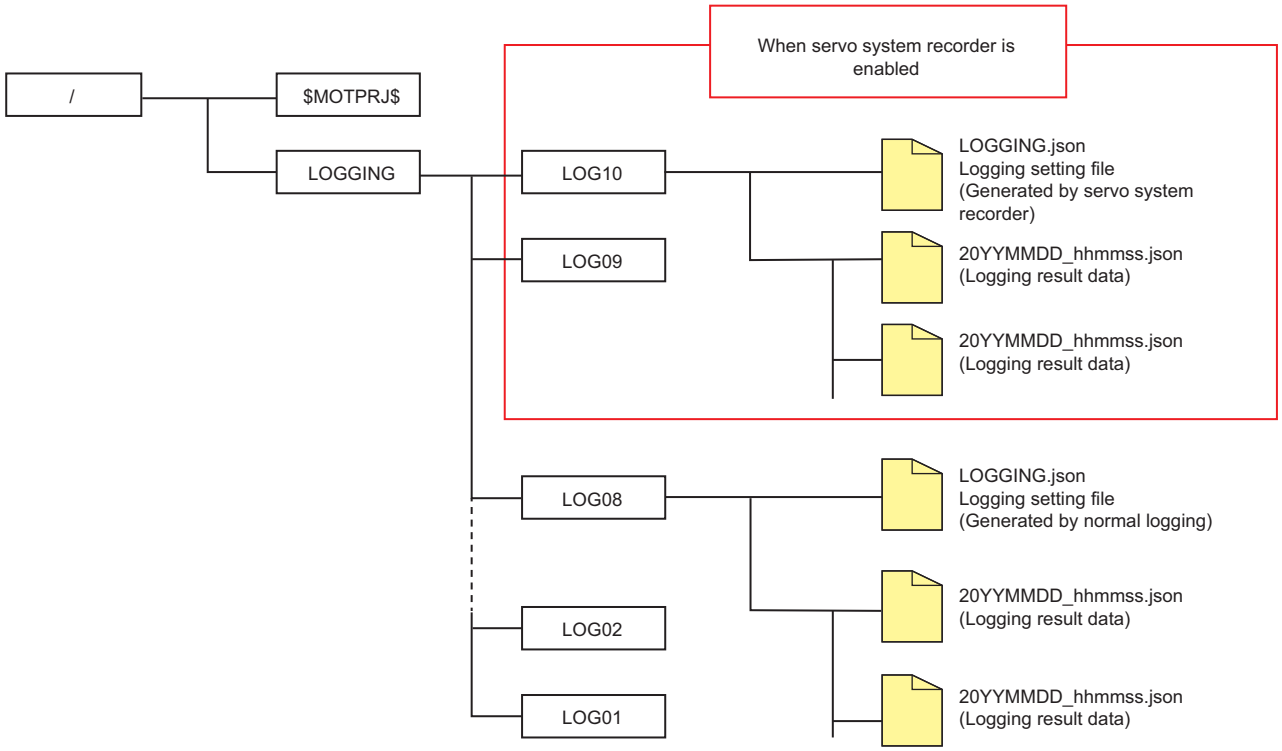
■Storage location

The logging setting file generated by this add-on is stored in an SD memory card or the user drive of the motion system. If the SD memory card is inserted, it is prioritized as a storage location of the logging setting file. (The storage location of the result file will be the same directory of the logging setting file.)
A logging function provides 10 logging setting file available for execution at the same time, and the following settings will be used in this add-on.

Number of real drive axis		Setting
When setting 1 to 128 axes		Logging setting 10 (LOG10)
When setting 129 to 256 axes	For 1 to 128 axes	Logging setting 10 (LOG10)
	For 129 to 256 axes	Logging setting 09 (LOG09)

■Folder configuration

The folder configuration is shown below.



Precautions

- When using this function, do not use the logging setting 10 and 09 by normal logging operation. The logging setting 09 can be used by normal logging operation depending on the number of set axis. In such a case, the motion system does not notify to the slave device.
- When the logging setting 10 and 09 are once created with this function, the logging setting 09 is not deleted even if the number of set axis is changed to 128 axes or less and the motion system is turned on again. Delete the specified logging setting by its tool.

Editing logging setting file

For the logging setting file generated by this add-on, changing the collected data, collection interval (rate), number of collection, and trigger detection position can be edited by the following procedure. In such a case, change the data logging name to "ServoSystemRecorder_***". By doing this, the logging setting file is not overwritten by this add-on even if the motion system power is turned on again.

1. Read the logging setting file (LOG10 or LOG09) in the motion system by logging setting tool.
2. Edit the logging setting file and change the data logging name to "ServoSystemRecorder_***".
   ("***" part in the data logging name can be arbitrarily set by the user)
3. Write from the logging setting tool to the motion system (LOG10 or LOG09).

By editing the logging setting file, adding the target data or deleting the unnecessary data is possible as necessary to the logging setting file generated automatically.
If the operation cycle over occurs, the processing time of the operation cycle can be shortened by reducing the number of target data.
When the edited logging setting file is no longer needed, delete it by logging setting tool. By doing so, the logging setting file is automatically generated by default when the motion system power is turned on.

Precautions

- If the logging setting file is edited, be sure to change the data logging name to "ServoSystemRecorder_***". If the change is not executed, the logging setting file is overwritten by default when the motion system power is turned on.
- If the data logging name starts other character string than "ServoSystemRecorder", the warning "Servo System Recorder Setting File Creation Disabled Warning" (warning code: 0F13H) occurs and the servo system recorder does not start.

Details of logging setting

The specifications of the logging setting generated by the Add-on ServoSystemRecorder are shown below.

• Logging setting file

Item	Specification / Description	
Logging setting file (Uses the last LOG10 and LOG09 from 1 to 10 areas.)	LOG10/LOGGING.json	1 to 128 axes Number of set axis in the motion system (real drive axis) Collected data of 1 to 128 axes
	LOG09/LOGGING.json	129 to 256 axes Number of set axis in the motion system (real drive axis) Collected data of 129 to 256 axes
Logging start condition	Stop disabled after automatically start	
Logging type	Trigger logging (The rows before and after the trigger are output)	
Data storage location	SD memory card (If an SD memory card is not inserted, the storage location of the data is in the user drive. The user drive is the ROM area of the motion system.)	

• Data logging file (result file)

Item	Specification	
Logging data file name	20YYMMDD_hhmmss.json (Date and time at the file being generated.)	
File size	Changes depending on the number of collected data (number of axis) and the data value.	
File format	JSON	
Collection interval	125.00 [μs] (When the operation cycle is 125.00 [μs] or more, the collection interval will be the operation cycle)	
Operation when the file exceeds the number of saved files	Overwrite	
Trigger type	OR combine	
Number of records	Number of real drive axis: 1 to 128	Number of LOG10 records Number of trigger records before trigger: 1434 [records] Number of trigger records after trigger: 614 [records]
	Number of real drive axis: 129 to 256	Number of LOG09 and LOG10 records Number of trigger records before trigger: 717 [records] Number of trigger records after trigger: 307 [records]
Number of saved files	Sets 16 when the storage location of the data is an SD memory card, and 2 for the user drive.	

■Collected data of the servo system recorder

The following data is set automatically. The setting details can be changed by logging setting tool.

○: Set in the collected data, —: Not set in the collected data

Item		Description	Setting number of the number of real drive axis	
			128 or less (LOG10 setting)	129 or more (LOG10 and LOG09 setting)
AxisName.Md.				
	Object data_TargetPosIo_TargetPos	Indicates a value of object data TargetPos (The position command to be sent to the slave device)	○	○
	Object data_ PosActualValueIo_PosActualValue	Indicates a value of object data PosActualValue (The feedback of position to be received from the slave device)	○	○
	Object data_TorqueActualValueIo_TorqueActualValue	Indicates a value of object data TorqueActualValue (The feedback of torque to be received from the slave device)	○	○
	Axis Error Code ErrorID	Displays the error code at axis error occurrence.	○	○
	Axis Status AxisStatus	Displays the current status of the axis. When the axis starts, it transits to the axis status according to the command.	○	○
	Object data_VelActualValueIo_VelActualValue	Indicates a value of object data_ VelActualValue. (The feedback of velocity to be received from the slave device) For the unit, refer to the manual of connected driver.	○	—*1
	Object data_TargetVelocityIo_TargetVelocity	Indicates a value of object data_ TargetVelocity (The velocity command to be sent to the slave device) For the unit, refer to the manual of connected driver.	○	—*1

Item		Description
System.Md.		
	Processing time OperationCycle[1..1].ProcessingTime*2	Stores the processing time of the first operation cycle in a unit of μs.
	Cycle over OperationCycle[1..1].CycleOver*2	Turns TRUE when the cycle over of the first operation cycle is detected.
	Latest motion system error code ErrorID	Stores the latest error code.
	Network error code NetWorkErrorID	Stores the error code when a network error occurs.

*1 To set Io_VelActualValue and Io_TargetVelocity data even when there are 129 axes or more, add them by logging setting tool.

*2 CYCLE_MONI structure

Precautions

When the axis operation process high-speed mode is enabled, the sampled value is always 0 because the following monitors are not updated.

- AxisName.Md.Io_TargetPos (Object data_TargetPos)
- AxisName.Md.Io_PosActualValue (Object data_PosActualValue)
- AxisName.Md.Io_TargetVelocity (Object data_TargetVelocity)
- AxisName.Md.Io_VelActualValue (Object data_VelActualValue)
- AxisName.Md.Io_TorqueActualValue (Object data_TorqueActualValue)

■Trigger condition(TRIGGERCONDITION)

No.	Specification
1	Event history (EVENTHISTORY) • An error occurrence that stops an axis is to be the trigger. For the target range of the events, refer to the following. ☞ Page 577 Target error code to be triggered • An error occurrence in other than real drive axis is to be the trigger as well. • Saves the logging data file every time the event code which is different from the previous code is registered to the event history.
2	At trigger command execution (LOGTRG) The execution of LOGGING_REF.Cd.Trigger is to be the trigger.

■Target error code to be triggered

Error code	Name.	Remark
1A00H to 1BFFH	—	Minor error (Axis error)
1C41H	Watchdog Counter Error	Minor error (Network error)
1C43H	SLMP Communiction Error	Minor error (Network error)
1C45H	SLMP Communiction Error (Timeout)	Minor error (Network error)
1C80H	Cycle Over	Minor error (System error)
2000H to 32FFH	—	Moderate error
3C00H to 3FFFH	—	Major error

If an error which may stop the system occurs and the logging is being unable to continue, the logging data may not be collectable.

Restart from the logging interruption

If an error occurs in the logging while starting the servo system recorder using an SD memory card, the logging is interrupted. By inserting the SD memory card again, the logging can be restarted without turning the motion system power on again.

How to restart logging

1. After clicking the SD memory card access control switch until CARD READY LED turns off, pull out the SD memory card.
2. CARD READY LED turns on as the SD memory card is inserted again.
3. Restart of the logging can be confirmed if the System.Md.ServoSystemRecorder_Enable of the system label is TRUE.

Logging restart conditions

- The servo system recorder is started with using an SD memory card.
- System.PrConst.Logging_AutoLoggingEnable is TRUE.
- The logging setting file of the servo system recorder is saved in LOG10 or LOG09 in an SD memory card.
- The write-protected in an SD memory card is cancelled.
- The error factors occurred in the logging are removed.

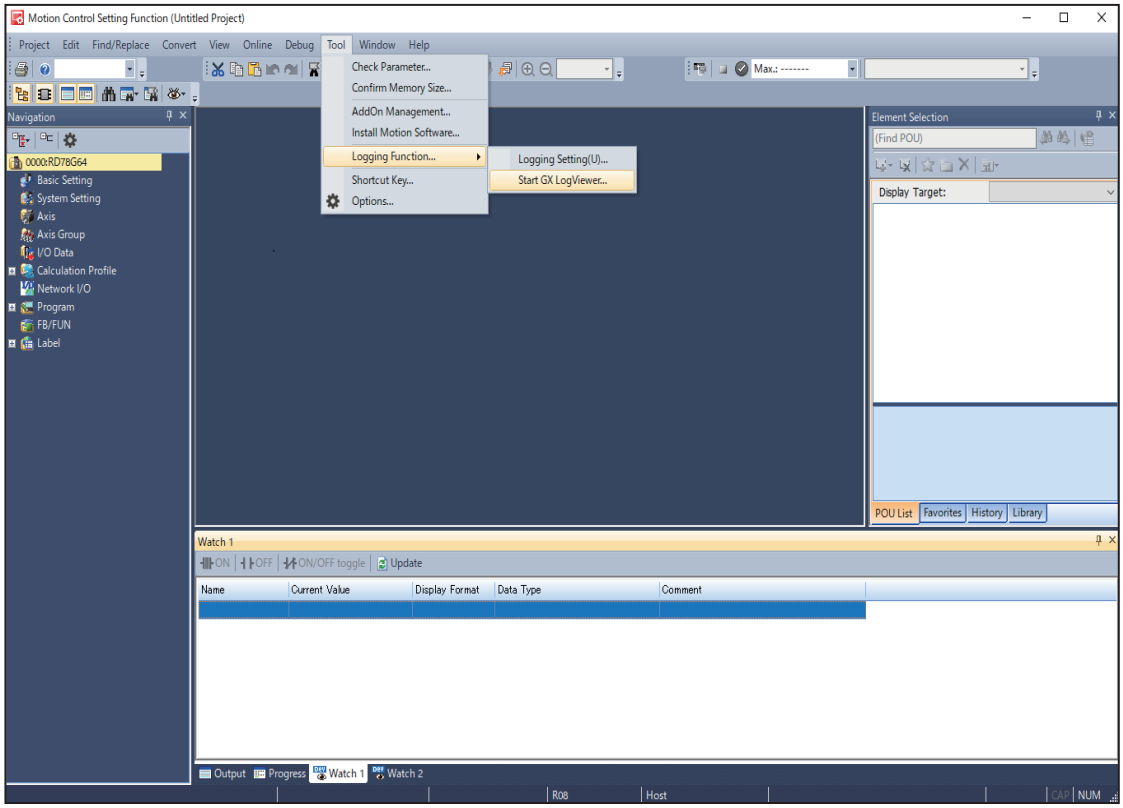
Precautions

- The logging cannot be restarted even if the System.LoggingRef[1..10].Cd.StartLog of the system label is turned TRUE.
- The warning "Servo System Recorder Logging Execution Disabled Warning" (warning code: 0F14H) occurs when the logging in LOG10 or LOG09 is being executed at reinserting the SD memory card and the logging of the servo system recorder does not start.

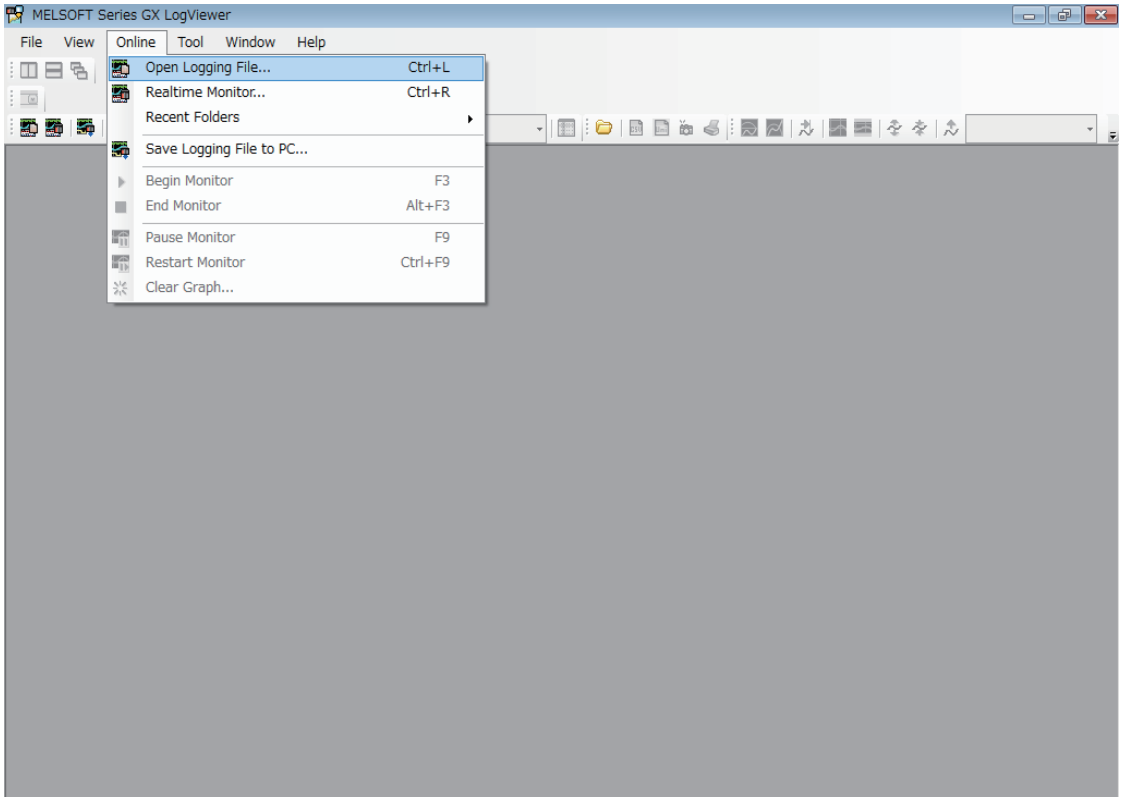
How to check the logging data

[To check the logging data in the motion system]

1. In "Motion Control Setting Function", start GX LogViewer by selecting [Tool] ⇨ [Logging Function] ⇨ [Start GX LogViewer].

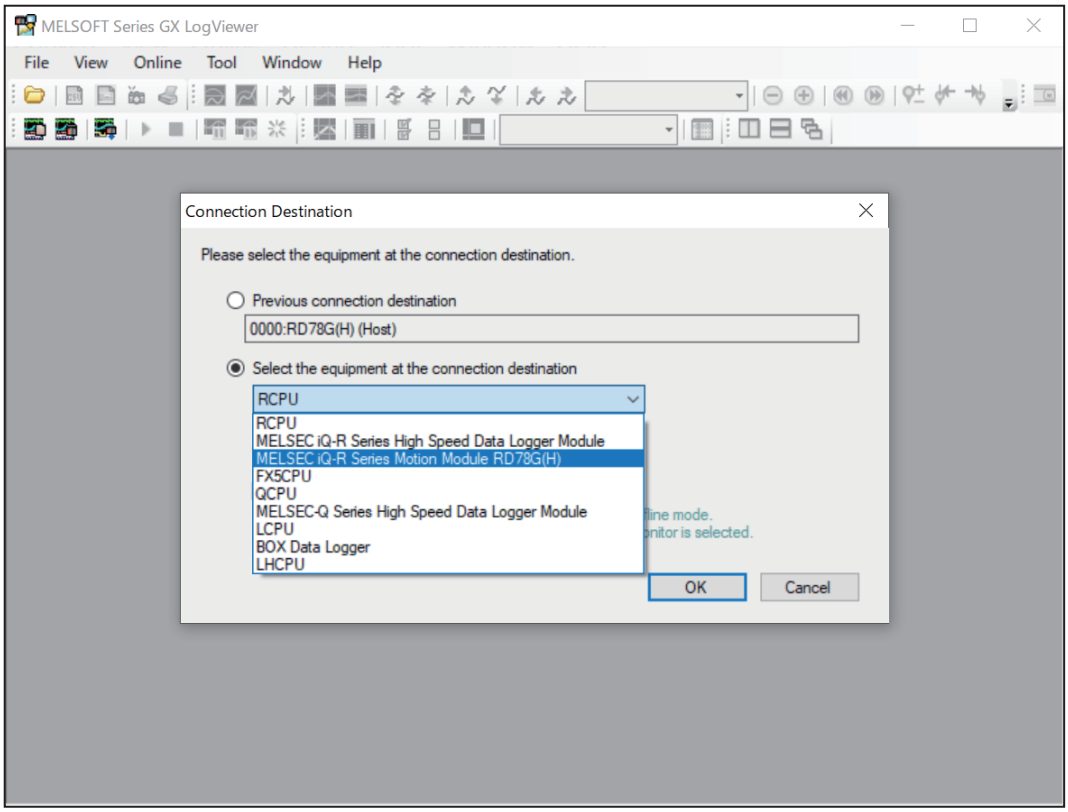


2. In "MELSOFT Series GX LogViewer", select [Online] ⇨ [Open Logging File].

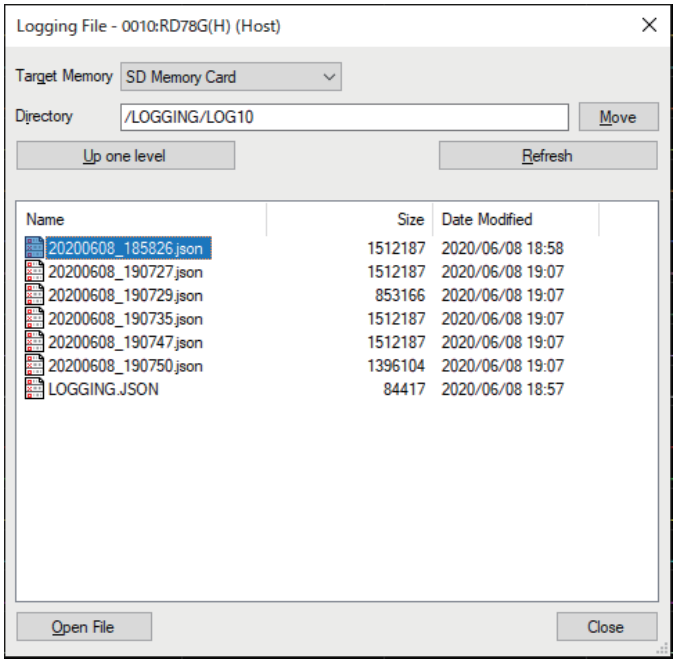


16

3. In the "Connection Destination" window, select "MELSEC iQ-R Series Motion Module RD78G(H)".



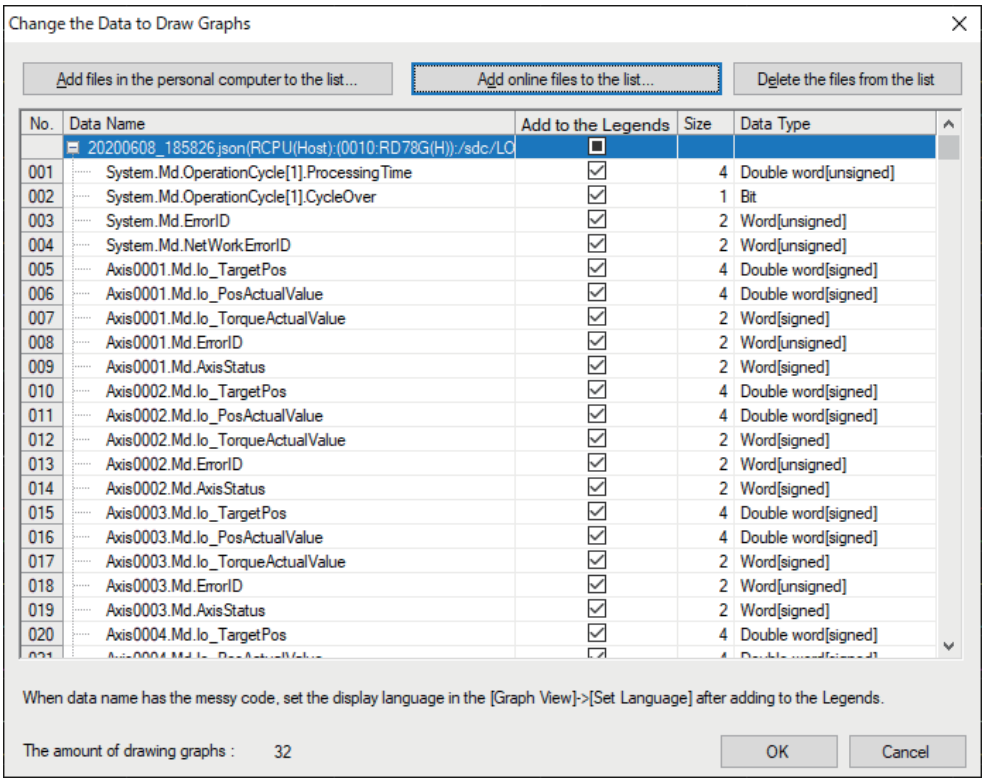
4. In the "Logging File" window, specify the "Target Memory" (When the SD memory card is inserted, the storage location of the data logging file is the SD memory card.), select the logging data to be checked, and click [Open File] button. (Multiple selection is not allowed)



(The following procedure is required when the target axis for logging is 129 axes or more.)

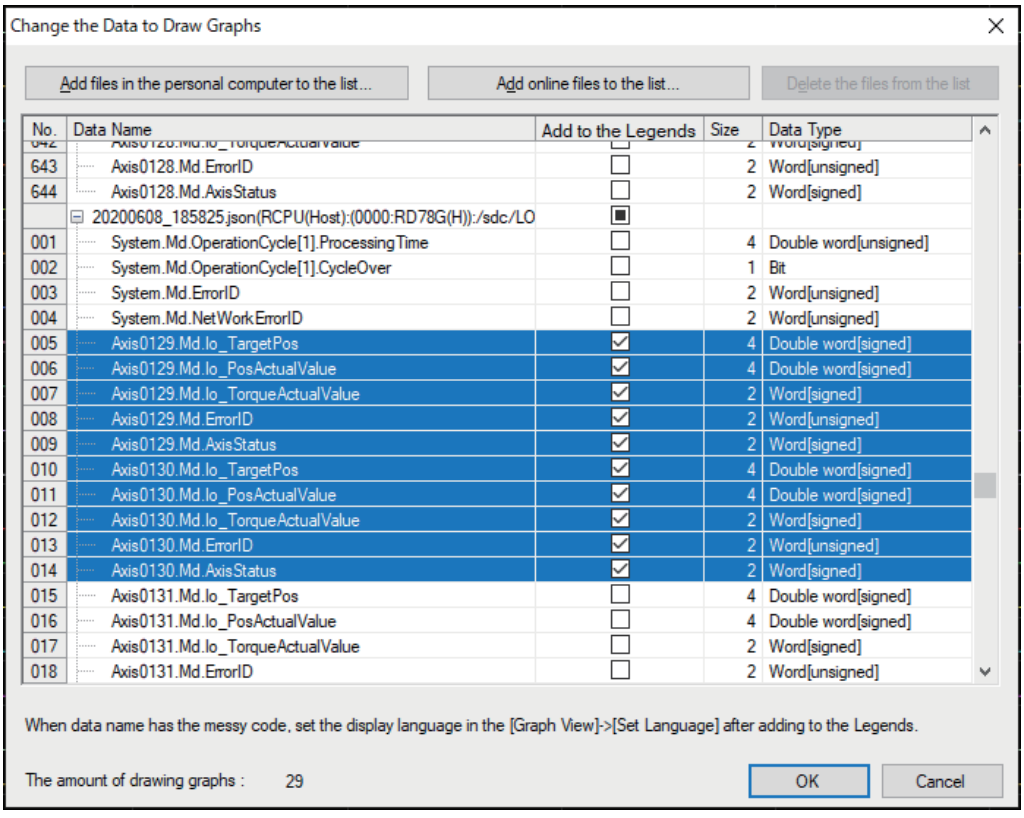
5. In "MELSOFT Series GX LogViewer", select [Graph View] ⇨ [Change the Data to Draw Graphs].

6. In the "Change the Data to Draw Graphs" window, click [Add online files to the list] button.



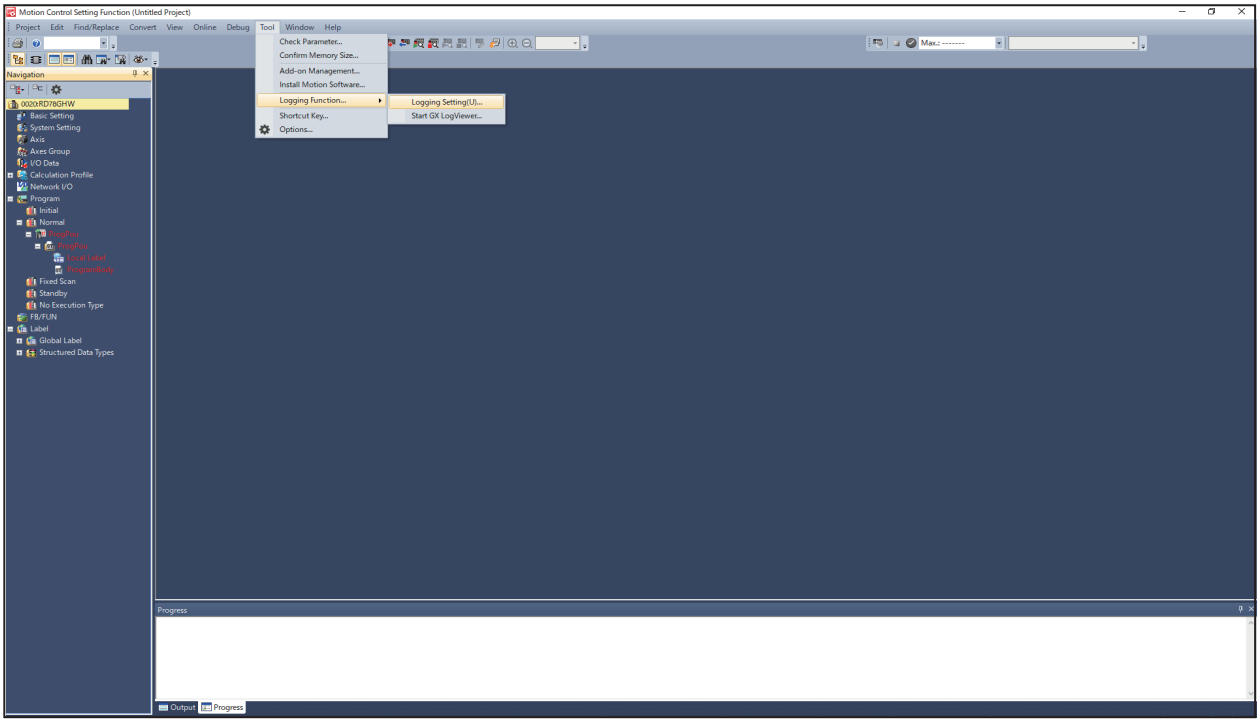
7. Add data logging file by the same procedure as step 3 and 4.

8. Check "Add to the Legends" column in added data logging file.
Up to 32 data are available to draw at the same time. Change the target data of drawing if necessary.

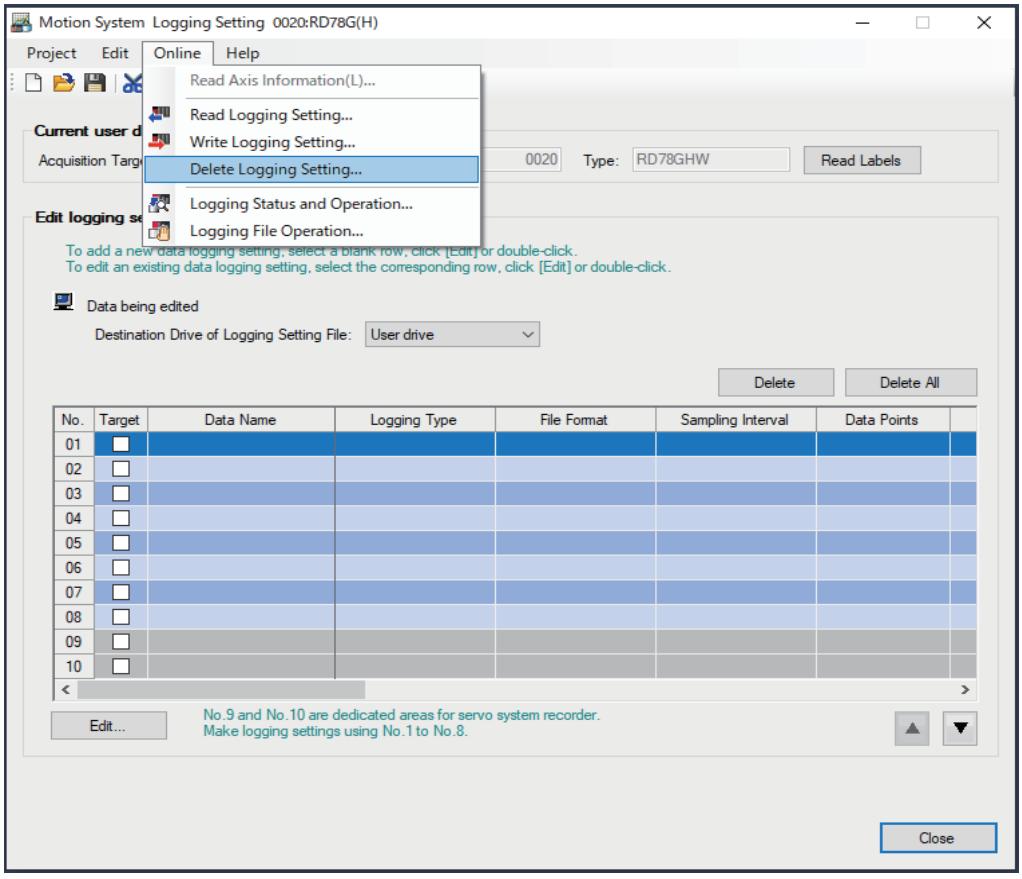


How to delete the logging setting file

1. In the "Motion Control Setting Function" window, select [Tool] ⇨ [Logging function] ⇨ [Logging Setting], and display "Motion System Logging Setting" window.

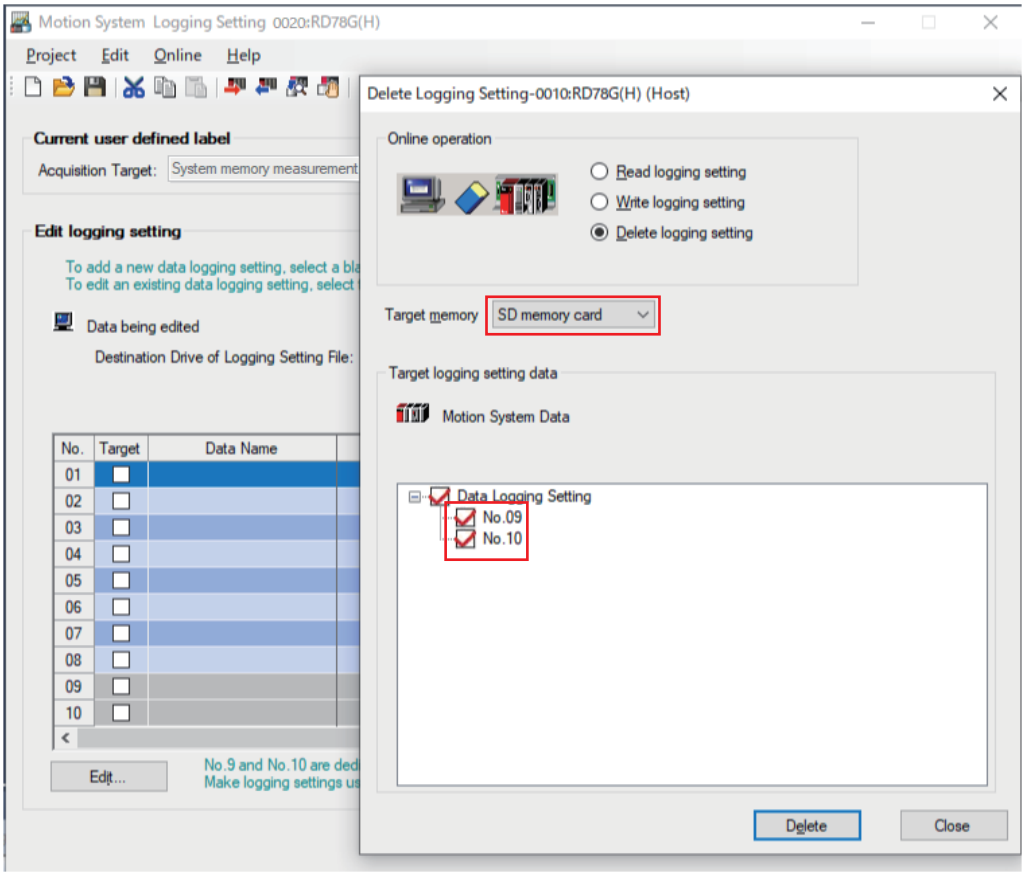


2. In the "Motion System Logging Setting" window, select [Online] ⇨ [Delete Logging Setting].



3. In the "Delete Logging Setting", select the memory which stores the logging setting file to be deleted from the target memory.

4. When the stored logging setting file is displayed, check the logging setting file to be deleted, and click [Delete] botton.

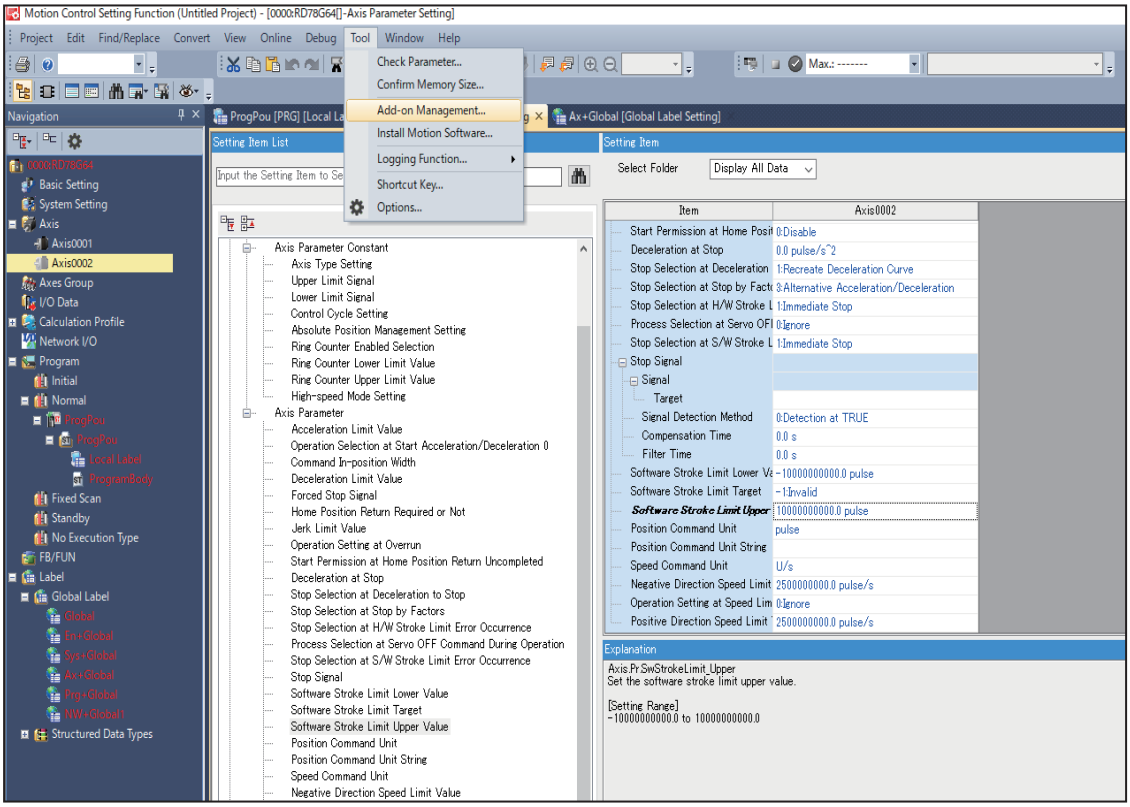


Precautions

If this function is not being used, disable the add-on library setting

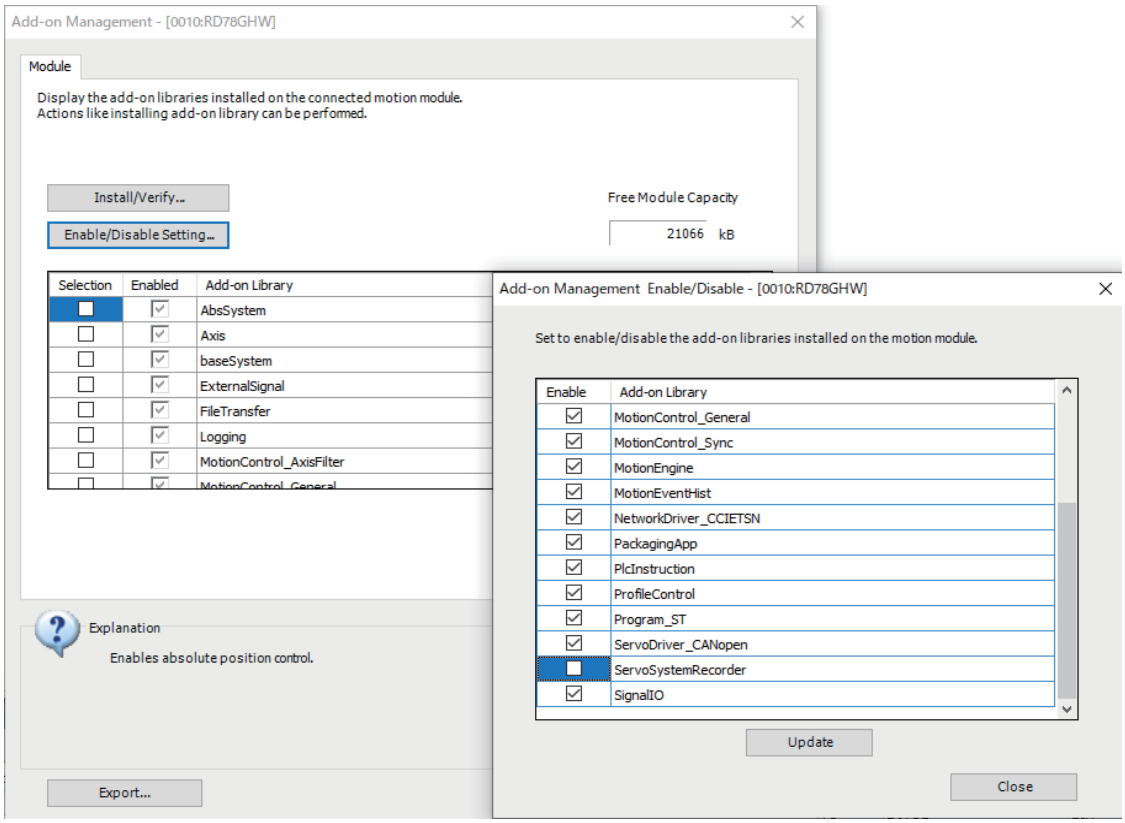
[Set the add-on library to disable setting]

1. In the "Motion Control Setting Function", select [Tool] ⇨ [Add-on Management] and open "Add-on Management".



2. Click [Enable/Disable Setting…] button, and display Add-on Management Enable/Disable window.

3. Uncheck the Enable column in ServoSystemRecorder, and click [Update] button.



When logging is executed by this add-on with all add-ons being enabled, the processing time of the operation cycle increases. A reference of the processing time of the operation cycle is shown below. (The processing time may change depending on the operation cycle and the device being used).

- RD78G64 (when 64 axes are being used): approximately 4.155 [ms]
- RD78GHW (when 256 axes are being used): approximately 4.961 [ms]

Relevant add-ons and versions

The following add-ons are required besides this add-on.

- Logging "Ver.1.7" or later
- Axis "Ver.1.7" or later

When the above add-ons are being disabled while this add-on is enabled, the warning "Servo System Recorder Inconsistency Incorrect Warning" (warning code: 0F10H) occurs.

System memory capacity

The memory capacity is used for generating the logging setting file by this add-on. By logging with generated setting, the Add-on Logging memory will be used. Both memory capacity are shown below.

■Add-on ServoSystemRecorder

[RamUsage]

- RD78G: 256 [K bytes]
- RD78GH: 512 [K bytes]

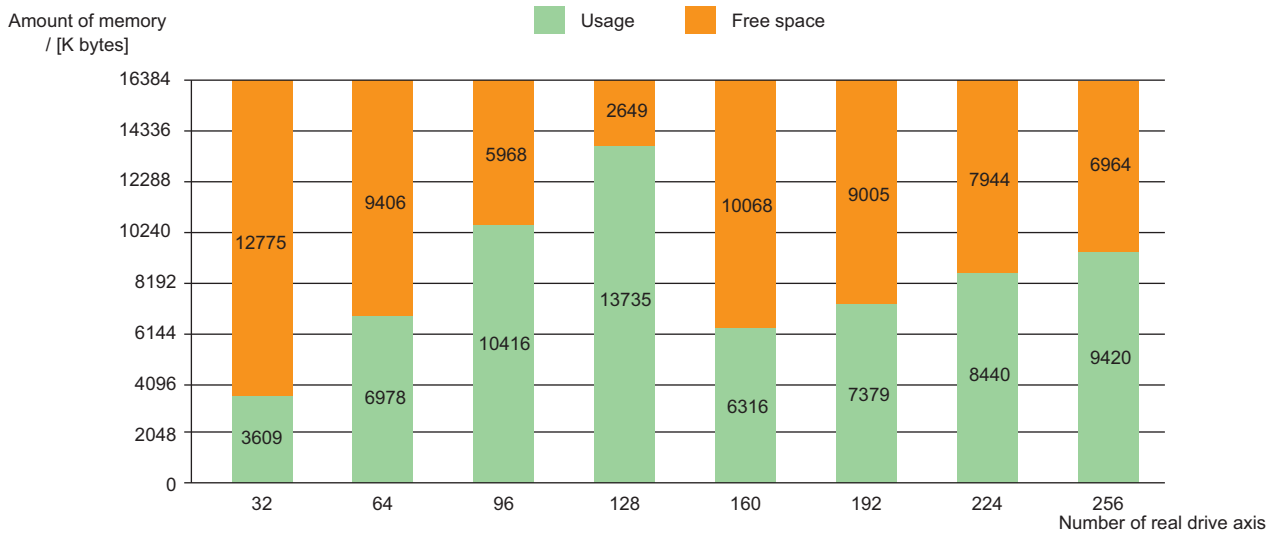
[BackupRamUsage]

- None

■Add-on Logging

[RamUsage]

The memory capacity during the logging differs because on the number of collected data differs depending on the number of set axis setting. An example is shown below.



- The default memory usage setting of the logging is 16384 [K bytes].
- The memory usage is the memory capacity logged in the logging setting file output by the servo system recorder.
- When 129 axes or more are set, the memory capacity is less than setting 128 axes because the number of collected data points and the record points of axis are being reduced.

[BackupRamUsage]

- None

17 MOTION SERVICE PROCESSING

17.1 Operation Overview

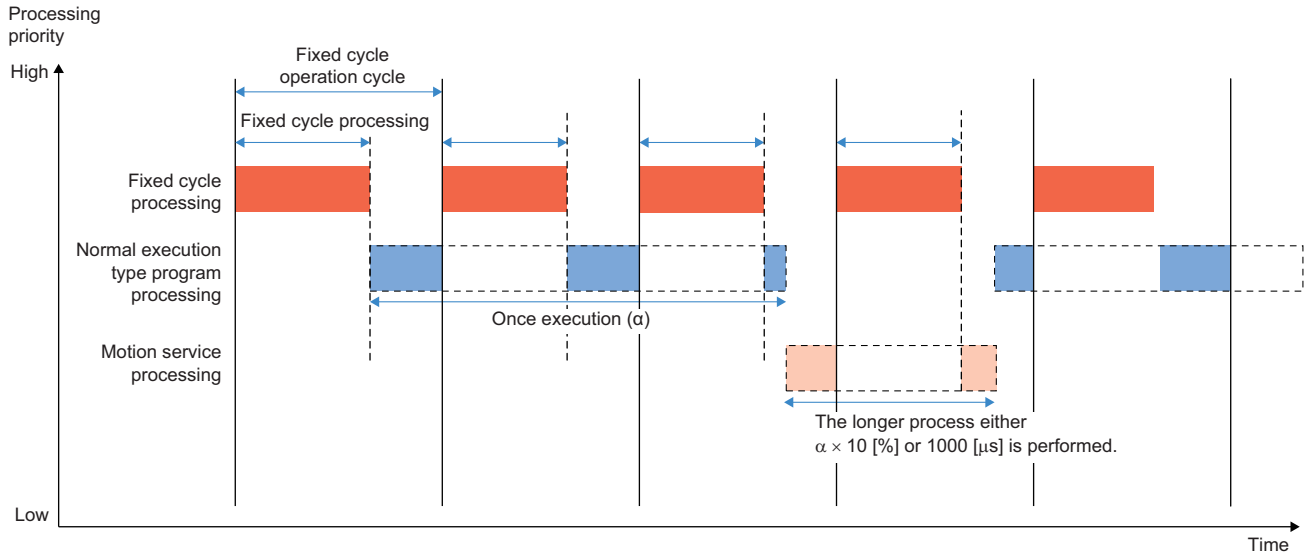
This section describes the motion service processing which executed in the service task.

Motion service processing control

The following shows the operation by the motion service processing setting.

The motion service processing is executed in vacancy time of the fixed cycle processing as well as the normal execution type program.

When all normal execution type program is executed once, the motion service processing is started.



17.2 Processing Description

The following shows the list of the motion service processing.

Applicable function	Description
File access processing	Executes the access processing to the file. The file access processing by a part of FBs is executed in higher priority task than the motion service processing. Refer to the specification of each FB.
MELSOFT communication processing	Executes the communication with the peripheral equipment such as the engineering tool and GOT.

18 HOW TO CONTROL FROM CPU MODULE

18.1 Specifications of I/O Signals with CPU Modules

This section describes how to control the Motion module from the CPU module (control CPU only).
For Motion module program creation procedure, refer to "Motion Module Program Creation" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

Module label

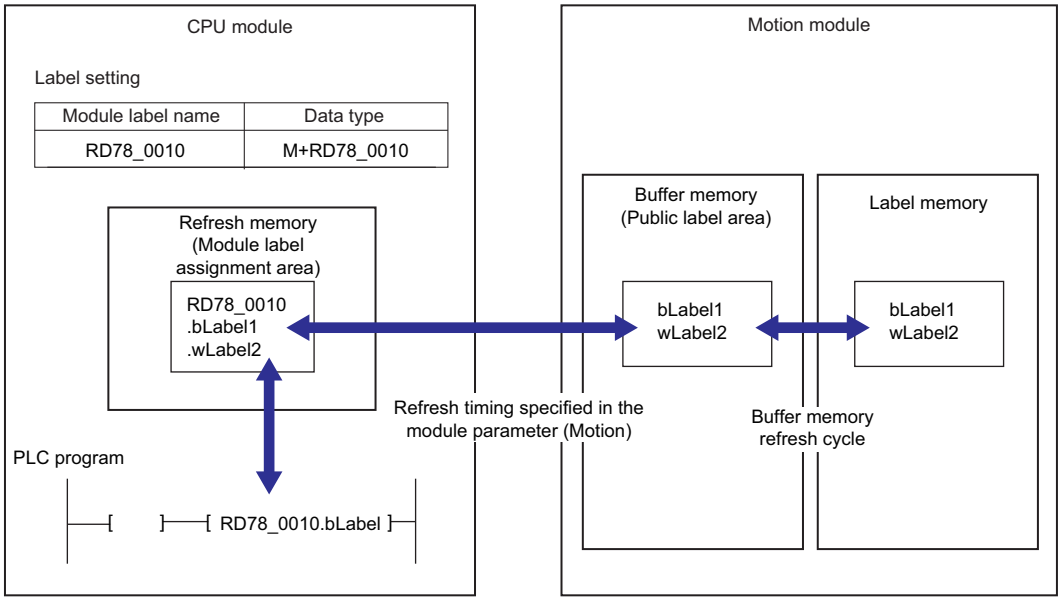
The global label in the Motion module is available as the module label from the CPU module.

■Using method

Follow the procedure below to create the module labels on the engineering tool.

1. In the label editor, set the "public label" of label to be made public as module label to the CPU module to "Enable".
2. Set "Motion Control Attribute" to decide direction of the label refresh.
3. Execute "Public label reflection" operation.

For details, refer to Help of the engineering tool.



When using the public labels, the data refresh is executed using the following memory area. For specifications such as the number of labels to be made public, refer to Help of the engineering tool.

Memory location	Name	Refresh timing
CPU module	Refresh memory (module label assignment area)	Specified in the module parameter (motion)
Motion module	Buffer memory (public label area)	Buffer memory refresh cycle

Point

The global labels to be made public to the CPU module use the buffer memory (area for the labels) of the Motion module. Read/Write directly for the buffer memory other than user setting area must not be carried out in the user program. For the global label size to disclose (the available label size to assign), refer to the following.
GX Works3 Operating Manual
The disclosed labels are refreshed in the "buffer memory refresh cycle". For details, refer to the following.
Page 109 Operation Cycle

■Motion control attribute

The public labels have attributes (= refresh direction).

The label attributes provided by the system can not be changed.

- The CPU module → The Motion module

Writes from the CPU module to the Motion module. In the labels provided by the system, they become the attributes when the parameters and control data (Pr, PrConst, Cd, etc.) are disclosed.

- The Motion module → The CPU module

Writes from the Motion module to the CPU module. In the labels provided by the system, they become the attributes when the monitor data (Md, etc.) are disclosed.

Precautions

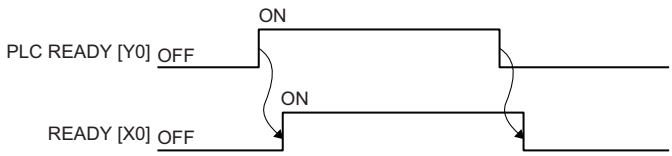
- A string type or a structured data type include a string type is not made public. For other restrictions, refer to the following.
  GX Works3 Operating Manual
- If the different direction from the attribute is written (e.g. The public labels, which attributes are "The CPU module → The Motion module", are written from the Motion program or watch), the values of the refresh-source overwrite the label values with the next buffer memory refresh cycle.
  Therefore, label values (some Cd. etc.) with attribute "The CPU module → The Motion module" which are changed from the motion system cannot confirm the values are changed from the CPU module. Refer to the Md. which is compatible with the Cd. or use the FB which provides the same function without operating the Cd.
- The operation performance is improved and the memory capacity can be reduced by reducing the number of the public lables.
- When the program is carried out with mismatches between the CPU module and the Motion module for the setting of the public labels, the refreshing can not be performed correctly.
  When the setting of the disclosed labels is changed, make sure to carry out the write of the program for both the CPU module and the Motion module.
- The closed labels of the same structure data type can not be substituted for the public labels of the structure including the closed member (and vice versa). Especially, when setting the public labels for the I/O pin (the structure) of the FB, disclose the all member of the public labels.
- When substituting to the structure type labels having the different attribute member such as AXIS_REA type, the member values of the CPU module ⇨ the Motion module attribute are irregular (The values of the refresh-source overwrite the label values with the next buffer memory refresh cycle.)
- When using the public labels, do not execute the software reboot. If it is executed, turn on the power of the motion system again

I/O signals

The Motion module uses 32 points of the input and 32 points of the output for exchanging data with the CPU module.
The following table shows the I/O signals when 0H has been set to the start I/O No. of the Motion module.
The device X is the input signal from the Motion module to the CPU module, and the device Y is the output signal from the CPU module to the Motion module.

Signal direction: Motion module → CPU module		Signal direction: CPU module → Motion module	
Device No.	**Signal name**	**Device No.**	**Signal name**
X0	READY	Y0	PLC READY
X1	Synchronization flag	Y1	Use prohibited
X2	Use prohibited	⋮	
⋮		⋮	
X1F		Y1F	

The ON/OFF timig and conditions of the I/O signals are shown below.

Device No.	Signal name		Description
X0	READY	OFF: Not READY ON: READY	• When the PLC READY [Y0] turns from OFF to ON, the parameter setting range is checked. If no error is found, this signal turns ON. • When the PLC READY [Y0] turns OFF, this signal turns OFF. • When a moderate error or a major error occurs, this signal turns OFF. • This signal is used for interlock in a program, etc. 
X1	Synchronization flag	OFF: Module access disabled ON: Module access enabled	• After the CPU module is turned ON or resetted, this signal turns ON if the access from the CPU module to the Motion module is possible.*1
Y0	PLC READY	OFF: PLC READY OFF ON: PLC READY ON	• This signal notifies the Motion module that the CPU module is normal. It is turned ON/OFF with the PLC program. • When the data (parameter) is changed, the PLC READY [Y0] is turned OFF depending on the parameter. • The following processes are carried out when the PLC READY [Y0] turns from OFF to ON. The parameter setting range is checked. The READY signal [X0] turns ON. • The following processes are carried out when the PLC READY [Y0] turns from ON to OFF. The READY signal [X0] turns OFF. The operating axis stops.

*1 For accessing to the Motion module from the program, necessity of the interlock according to the module synchronization setting of the CPU module and the access destination in the Motion module are shown below.

Access destination	Module synchronization setting	
	"Synchronize" the rising of the module	**"Asynchronous" the rising of the module**
Motion control area	The interlock by X1 is required.	The interlock by X1 is required.
Network control area	The interlock is not required.	The interlock by X1 is required.

Buffer Memory

User setting area can be used for exchanging data between the CPU module and the Motion module.

Start No.	End No.	Name
11478000	11997999	User setting area
11999997	11999997	System (Not available for user)

Do not use the other areas than the above.

18.2 Using MCFB

This section describes how to use the MCFB in PLC programs of the CPU module.

Relevant variable

Variable/Structure name		Name	Details
System.Md.			
	BuffermemoryFreeMcfbArea	Buffer memory MCFB communication area free size	Indicates the number of unused points of the area for MCFB.

Control details

The specifications of variables that are used for the MCFB of the CPU module side are shown below.

• Specifying AXIS_REF and AXES_GROUP_REF

When using the MCFB of the CPU module side, use the followings for the axis information (AXIS_REF) and the axes group information (AXES_GROUP_REF) that are specified for the I/O variables of the MCFB.

AXIS_REF and AXES_GROUP_REF to be specified	Setting
AXIS_REF(UNIT_LABEL.AxisName.AxisRef) or AXES_GROUP_REF(UNIT_LABEL.AxesGroupName.AxesGroupRef) in the axis disclosed in the module labels or the axes group variables are used.	The setting in the sequence program is not required as the members of AXIS_REF and AXES_GROUP_REF (the axis, axes group No., and module I/O No. of the control target) are set.
AXIS_REF and AXES_GROUP_REF that are defined as labels in the CPU module are used.	Set the members (the axis/axes group No. and the module I/O No. to be controlled) of AXIS_REF and AXES_GROUP_REF by using the PLC program.

• Specifying the I/O No.

When using the MCFB of the CPU module side, the start I/O No. of the target Motion module has to be specified for the I/O variables (AxisName.AxisRef.StartIO, etc.) used in the MCFB.

Each MCFB definitely defines one input variable that requires the StartIO setting. For the input variables that require the StartIO setting, refer to the following.

☞ 26 List of FBs

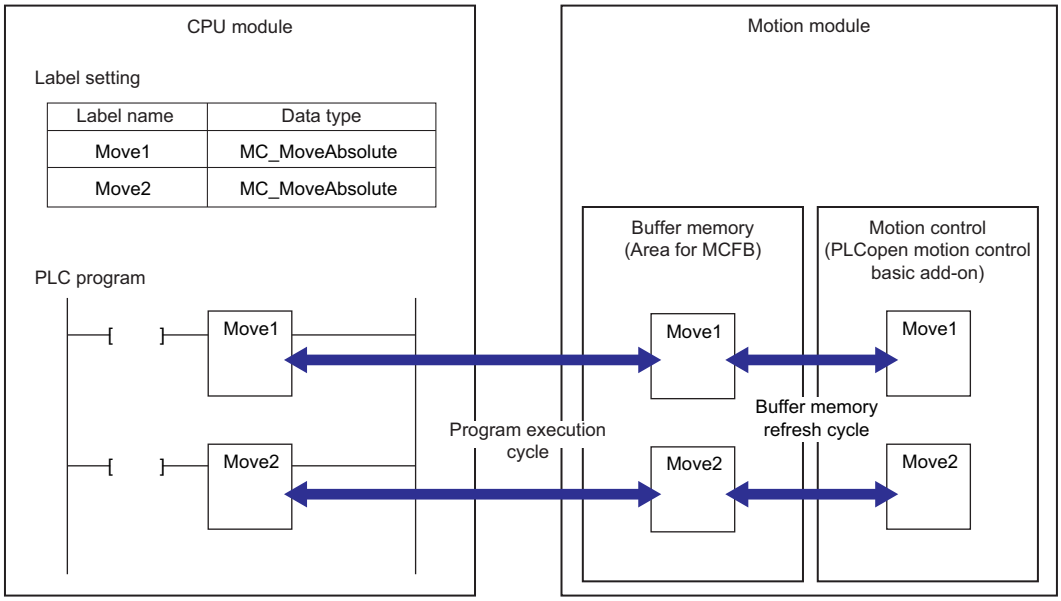
For the MCFB that can input multiple variables whose member includes the StartIO, the setting of the StartIO excluding the corresponding variables is not required. (The values are ignored.)

For the start I/O No. of the target module, refer to UNIT_LABEL.uIO of the module label or directly input the constant.

Be sure to establish the StartIO of each MCFB before the FB is called for the first time.

Generating MCFB instance

Control corresponding to the MCFB can be carried out in the Motion module side by instantiating the MCFB as a global label or a local label on the CPU module and calling the MCFB on the PLC program.



When using the MCFB of the CPU module side, the following memory areas are used. The number of available MCFB instances must not exceed the size of each area.

Memory location	Name	Occupied size	How to check the available size
CPU module	Label memory	☞ Page 721 List of FBs	Check by using the engineering tool
Motion module	Buffer memory (Area for MCFB)		System.Md.BuffermemoryFreeMcfbArea
	Memory for PLCopen motion control basic add-on		System.PrConst.AddonMotionControl_General.RamSizeMax System.Md.AddonMotionControl_General.RamSize

Point

- The buffer memory (area for the MCFB) and the memory for the motion control add-on are hold when the MCFB is called for the first time after the CPU module is set to RUN. If the free space is in short, the error "Insufficient memory capacity" (error code: 349AH) will occur when the MCFB is called for the first time, and the MCFB will not be executed.
- Even if the MCFB of the CPU module side is deleted, the memory on the Motion module side is not released. Therefore, if the MCFB is added or deleted repeatedly by writing during RUN, the memory area of the Motion module side will be in short and the error "Memory shortage" (error code: 349AH) may occur. In this case, reset the CPU module once.

Operating MCFB

• Operation conditions

The conditions (available conditions) to operate the MCFB of the CPU module side are shown below.

CPU module	Motion module	
	RUN	Other than RUN
RUN	Available	Unavailable MCFB in execution is stopped
Other than RUN	Available*1	

*1 This state occurs when System.Cd.SequenceReady is being RUN by operated directly, and the stop error is not occuring in the CPU module.
In this case, execute such as an axis stop in the Motion module program if necessary because the Motion module cannot be controlled from the CPU module. For details, refer to the follwoing.
☞ Page 128 Relevant variables

• Processing order

The processing order of the operation cycle in the Motion module corresponding to the MCFB instance of the CPU module side is the order when the MCFB is called for the first time.

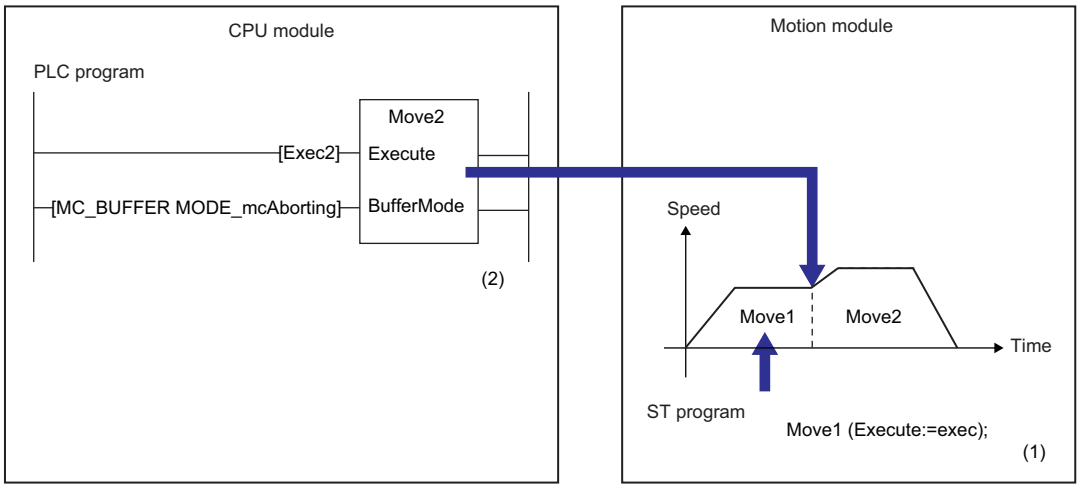
• Mix of the MCFB of CPU module side and the MCFB of the Motion module side.

When the MCFB of the CPU module side and the MCFB of the Motion module side are simultaneously executed to the same axis or axes group, the specifications of the buffer mode and the single axis synchronous control are applied.

Even if the global instance names are the same between the MCFB of the CPU module side and the MCFB of the Motion module side, these are treated as different control FBs in the Motion module.

Ex.

When executing Aborting to the on-going MCFB in the Motion module side by the MCFB of the CPU module side

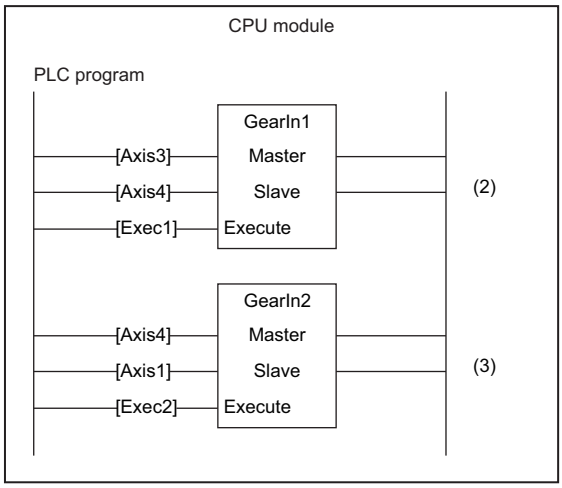


(1) Execute the Move1 of the Motion module side.

(2) When executing the Move2 of the CPU module side while the Move1 is in execution, the program is operated with the buffer mode specified by the Move2.

Ex.

When mixing the MCFBs of the single axis synchronous control in the Motion module side and the CPU module side



Motion module

ST program

```
GearInA(Master:=Axis1, Slave:=Axis2, Execute:=exec);    (1)
GearInB(Master:=Axis2, Slave:=Axis3, Execute:=exec);
```

(1) When executing the GearInA and GearInB of the Motion module side, the Axis2 and Axis3 of the slave axes run.

(2) When executing the GearIn1 of the CPU module side, the Axis4 runs using the slave axis of the GearInB (Axis3) as the master axis.

(3) When attempting to execute the GearIn2 of the CPU module side, the Axis1 to Axis4 make a circular reference. Therefore, an error occurs and the GearIn2 cannot be executed.

- Refresh cycle of I/O variables

The refresh response time of I/O variables for the MCFB of the CPU module side is as shown below.

MCFB call cycle in the CPU module + Buffer memory refresh cycle × 2

Precautions

- Do not use the MCFB in an interrupt program of the CPU module.
- When using the MCFB in the CPU module side, do not execute the software reboot. If it is executed, turn on the power of the motion system again.

Point

Errors (including warnings) of the CPU module side during MCFB execution are output as errors of the Motion module. For the error code output to ErrorID of the MCFB, refer to the following.

☞ Page 30 Error processing

However, when the I/O No. (☞ Page 592 Relevant variable)of the Motion module specified with the MCFB argument is not correct and the Motion module to execute the FB cannot be specified, no operation is performed or the error code is output in the CPU module side.

19 FILE CONTROL

19.1 Memory and File

The programs and parameters used in the motion system control are stored in files. The file storage location can be set freely by parameters of each control and I/O variables of FB.
The motion system built-in memory, the control CPU module memory of the motion system, and an SD memory card can be used as data area for file storage.

Operation of this function at each status of system

○: Possible

Status	Operation availability
STOP	○
RUN	○
Moderate error	○
Major error	○

Relevant variables

Variable/Structure name		Name	Details
System.Md.			
	Storage_rom	User drive information	Stores the information of user drive.
	Storage_sys	System drive information	Stores the information of system drive.
	Storage_ram	RAM drive information	Stores the information of RAM drive.
	Storage_lch	Latch drive information	Stores the information of latch drive.
	Storage_sdc	SD memory card information	Stores the information of SD memory card.
	HwStatus_RomWriteCycle	Flash ROM write count index value	Stores the flash ROM write count index value.

Drive

Drive structure

A generic term for the file storage memory and the SD memory card slot of the motion system is called "Drive". The following shows available drives of the motion system.

○: Available, ×: Not available

Item	User drive	System drive	RAM drive	Latch drive	SD memory card
Usage	• Project data • General data	• Module software • Add-on	Work data	• Absolute position data • Synchronous restoration data	• Project data • General data
Drive identifier	/rom	/sys	/ram	/lch	/sdc
Memory capacity*1	[RD78G] 16M bytes [RD78GH] 64M bytes	[RD78G] 8M bytes [RD78GH] 24M bytes	[RD78G] 16M bytes [RD78GH] 32M bytes	[RD78G] 240k bytes [RD78GH] 480k bytes	maximum 32G bytes
File system	Equivalent to FAT16(VFAT)*2	Equivalent to FAT16(VFAT)*2	FAT16(VFAT)	FAT12(VFAT)*2	SD memory card: FAT16(VFAT) SDHC memory card: FAT32(VFAT)
Format necessity before the first usage	Not necessary*3	Not necessary*3	Not necessary*4	Not necessary*3	Not necessary*5

*1 Total capacity including file management area. The available capacity is smaller.
*2 Power off is protected. (The file system is not damaged even if the power is off during writing.)
*3 Formatting is completed prior to shipping.
*4 Formatting is completed prior to power on/reset. (The file is deleted at power on/reset.)
*5 To format an SD memory card, format it with the CPU module. If formatted with such as the Windows formatting function, an SD memory card may not be able to be used by inserting it into the Motion module.

Point

The drive that can be used may be restricted depending on functions. Also, the function may be restricted depending on the drive to be used. For details, refer to each function.

Drive status

The drive status can be checked by STORAGE_INFORMATION type variable (System.Md.Storage_<drive>) which corresponding to each drive. (The <drive> indicates drive identifier: rom, sys, ram, lch, sdc.)

Variable/Structure name	Description
STORAGE_INFORMATION.Capacity	Stores the drive capacity in a unit of 1K bytes.
STORAGE_INFORMATION.FreeSpace	Stores the drive free space in a unit of 1K bytes.
STORAGE_INFORMATION.Mount	Indicates the drive mounted status. Furthermore, it indicates file access is available when it is set to TRUE.

Memory life

■Life of flash ROM

Since a user drive and system drive use a flash ROM, it has writable times (life). An indication of write times can be monitored by System.Md.HwStatus_RomWriteCycle and data can be written until it reaches to 100,000 times. If the value exceeds 100,000 times, it is registered in the event history.

If the write count index value exceeds 100,000 times, the following symptoms may occur. Replace the motion system.

- Decrease of writing speed to file
- Unable to write data to file

The write count index value may not match the write count by user operation. For example, the write count index value may not increase even if data is written to a file.

Point

- Since the motion system operation is checked at factory acceptance test, the initial value of the write count index is not 0.
- Since a motion system always checks data in the flash ROM, the write count index value may be increased without writing by user operation.

Path setting

The file path (including file name) to be used for control is specified with the STRING or FILE_LOCATION type variable by parameters and I/O variables of FB of each control. The following describes the path specification.

Path specifications

- For the file path, alphabetic characters (a-z, A-Z, 0-9), symbols (!#$%&'()+,-.;=@[]^_`{}~), spaces and directory delimiters (/) are available. Wide characters cannot be used.
- A wild card (*?) is available for a file name depending on the function. Refer to each function for the availability.
- The path length is up to 127 characters.
- The path is not case-sensitive.

■How to set the path

For the path setting, the absolute path and the relative path are available.

- Absolute path

The path start with / is an absolute path. An absolute path must be described with a character string combined with a file name to a drive identifier. For a drive identifier and a file name, refer to the following.

☞ Page 597 Drive, ☞ Page 601 Storage files

Ex.

/sdc/$MOTPRJ$/calc_profile/cam0001.csv

- Relative path

The path not start with / is a relative path. When a relative path is specified, the following directory is the root and the actual file access is performed with a root path + a relative path. The specified root path is shown below.

Module extended parameter storage location setting	Specified root path %PROJECT_ROOT%*1	Specified root drive %PROJECT_ROOT_DRIVE%*1
Motion module (internal memory)	/rom/$MOTPRJ$/	/rom
Motion module (SD memory card)	/sdc/$MOTPRJ$/	/sdc

*1 The root path is stored in the environmental variable PROJECT_ROOT defined with system. Also the drive name including root path (root drive) is stored in the environmental variable PROJECT_ROOT_DRIVE. These can be changed to the arbitrarily path by the pathset command of file transfer function.

Ex.

When the root path is not specified by the pathset command and the module extended parameter storage location setting is set to "Motion module (SD memory card)"

Specify the following path:

Relative path: calc_profile/cam0002.csv

The actual accessed file is as follows.

sdc/$MOTPRJ$/calc_profile/cam0002.csv

Ex.

When the $MOTPRJ$ path has been specified to /sdc/MyMachine/Work002/ by the pathset command and the following path is specified

Relative path: calc_profile/cam0002.csv

The actual accessed file is as follows.

/sdc/MyMachine/Work002/calc_profile/cam0002.csv

As the special relative path, ./ (indicates current directory) and ../ (indicates upper directory) can be used.

Ex.

When the root path is set to /sdc/$MOTPRJ$/ and the following path is specified

Relative path: ../cam0002.csv

The actual accessed file is as follows.

/sdc/cam0002.csv

■Environmental variables

The path includes environmental variables. The format of environmental variables is specified with %Environmental variable name%. Environmental variables are defined with system and users and those can be set by the pathset command of the file transfer function. For details, refer to the following.

☞ Page 613 File Transfer Function

The environmental variables defined with system are as follows.

Environmental variable	Overview	Default value
PROJECT_ROOT	Root directory	/rom/$MOTPRJ$ (Changed depending on the module extended parameter storage location setting)
CALC_PROFILE_DIR	Operation profile storage location	%PROJECT_ROOT%/calc_profile

Ex.

When the environmental variable MY_DIR defined by user is set to /sdc/my_data by the pathset command and the following path is specified

The path including MY_DIR: %MY_DIR%/file.csv

The actual accessed file is as follows.

/sdc/my_data/file.csv

■Drive No.

To access the file in the motion system with the SLMP communication, use the path combined with the "File name" part to the drive identifier corresponding to the "Drive No." part of the SLMP message.

The following shows the correspondence of drive Nos. and drive identifiers.

Drive No.	Motion system
0000H	Root <./>
0001H	—
0002H	SD memory card </sdc>
0003H	—
0004H	User drive </rom>
0005H	System drive </sys>
0006H	—
0007H	—
0008H	—
0009H	—
000AH	RAM drive </ram>
000BH	Latch drive </lch>
000CH	—
000DH	—

Ex.

When setting as follows and the root path is set to /sdc/$MOTPRJ$/

Drive No.: 0000H (root)

File name: calc_profile/cam0002.csv

The actual accessed file is as follows.

/sdc/$MOTPRJ$/calc_profile/cam0002.csv

Ex.

When setting as follows

Drive No.: 0002H (SD memory card)

File name: $MOTPRJ$/calc_profile/cam0002.csv

The actual accessed file is as follows.

/sdc/$MOTPRJ$/calc_profile/cam0002.csv

Operation that can be executed

The following shows operations that can be executed to the drive in the motion system.

- Writing, reading and deleting a file
- Formatting a drive

Point

- When a file attribute is set to read only, writing and deleting are not executed.
- When write protect is set to an SD memory card, writing and deleting are not executed.
- The update time and create time of files are based on the clock data of CPU No.1 module.

Storage files

The programs and parameters to be used to control the motion system are stored in the root path by default. The directory corresponding to the root path is created automatically when writing a file in the root path. For details, refer to the following.

☞ Page 599 Path setting

File lists

The following shows the list of files to be used in the motion system.

■%PROJECT_ROOT%

File name	Usage	Module extended parameter target	Path change with pathset
%PROJECT_ROOT%/			
/prog		○	○
/F000		○	○
*.PFB	FB/FUN	○	○
*.MLB	Member definition	○	○
/P???	Subfolder (??? = 000 to 032)	○	○
*.PRG	Motion program	○	○
*.LBS	Local label setting	○	○
*.LID	Local label default value	○	○
GLBLINF.IFG	Global label setting	○	×
GLBLINF.LBS	Global label setting	○	×
GLBLINF.LID	Global label default value setting	○	×
MTNSTR.MLB	Member definition	○	×
MTNSTR.STR	Structure type definition	○	×
/calc_profile	Operation profile folder	○	○
/PD??	Subfolder (?? = 00 to 03)	○	○
*.csv	Operation profile (csv format)	○	○
/auto_open	Operation profile folder (automatically opened)	○	○
/PD??	Subfolder (?? = 00 to 03)	○	○
*.csv	Operation profile (csv format)	○	○
*.cab	Project control data for engineering tool	○	×
PosHist.bin	Position data history file	×	○
filelog.txt	File transfer log file	×	○
filelog.bak	File transfer log backup file	×	○

■rom

File name			Usage	Module extended parameter target
/LOGGING			Logging folder	×
	/LOG**		Logging setting ** folder	×
		LOGGING**.json	Logging setting file	×
		*.csv	Logging data file (csv format)	×
		*.json	Logging data file (json format)	×
time.json			Time zone information	×

■sys

File name		Usage
/		
	*.smpk	Motion module software
	*.mpk	Motion module add-on
	/disabled	Temporary disabled file

■ram

File name		Usage
AxisList.json		Axis instance list
AxesGroupList.json		Axes group instance list
AddonInstallInfo.json		Add-on installation information list
/temp		Temporary folder
	/LOGGING	Operation folder for logging

■lch

File name	Usage
abs_axis.bin	Axis absolute position data
EVENT.LOG	Motion event history

■sdc

File name		Usage
$MOTPRJ$		
	autorun.txt	File Transfer at boot
LOGGING**.json		Auto logging

Module extended parameters

The programs and the parameters that are used for controlling the motion system are handled as module extended parameters. For the target files of the module extended parameter, refer to the following.

☞ Page 601 Storage files

The module extended parameter storage location can be selected from the below by "Module extended parameter storage location setting" of the module parameter.

Module extended parameter storage location setting	Storage location	Storage file format
Motion system (internal memory)	User drive of motion system (/rom)	Individual file
Motion module (SD memory card)	SD memory card drive of Motion module (/sdc)	Individual file

The module extended parameter storage location setting can be overwritten (override) by specifying PATHSET of the root path using the boot operation file.

Point

- To write the module extended parameter from the engineering tool, specify the writing destination same as the module extended parameter storage location setting. When the different writing destination is specified, the written module extended parameter is not enabled.

Data backup

Data saving/restoration in each drive to the outside the motion system can be executed by user operation using such as file transfer function.
Some data may be saved incorrectly if saving/restoration set procedure is not being followed. Also, when transferring drive data of the Motion module by using file transfer function by user operation, read/write of the target drive needs to be permitted in access control setting in advance. For details, refer to the specifications of each function.
To secure the consistency of the backup data, it is recommended to execute data saving (backup) and restoration (restored) while the drive unit is not connected and PLC ready [Y0] is OFF.

Data saving (backup)

The saving (backup) procedure of all data in the motion system by user operation is shown below.

1. To save the label values, reflect the description of the labels (user defined labels, each type of Pr and PrConst) to the label default value file (specified to store in the root path) by using "Filing" command of the file transfer function.
   For details, refer to the following.
   ☞ Page 617 Filing
2. If function such as the absolute position system which uses latch drive is being used, use System.Cd.BackupRestore to stop the data update of the latch drive.
   For details, refer to the following.
   ☞ Page 103 Saving (backup) and restoration (restored) of the absolute position data
3. Execute saving of all files under the root path, in the latch drive, and in the system drive to the outside using the file transfer function. For details, refer to the following.
   ☞ Page 615 Data copy / Data move
   Note that saving RAM drive is not required as it is used only to store temporary operation files.
4. If the data update of the latch drive is stopped following procedure 2, resume the update using System.Cd.BackupRestore. For details, refer to the following.
   ☞ Page 103 Saving (backup) and restoration (restored) of the absolute position data

Data restoration (restored)

The restoration (restored) procedure of all data in the motion system by user operation is shown below.

1. If function such as the absolute position system which uses latch drive is being used, use System.Cd.BackupRestore to stop the data update of the latch drive. For details, refer to the following.
   ☞ Page 103 Saving (backup) and restoration (restored) of the absolute position data
2. By using the file transfer function, restore files stored outside drive to the original drive. For details, refer to the following.
   ☞ Page 615 Data copy / Data move
3. If the files in the latch drive are changed following procedure 2, execute restored data saving request by using System.Cd.BackupRestore. For details, refer to the following.
   ☞ Page 103 Saving (backup) and restoration (restored) of the absolute position data
4. Reboot (software reboot or hardware reboot) the motion system.

Point

- Data saving/restoration can be executed with the saving/restoration functions of the engineering tool. For details, refer to Help of the engineering tool.
- The Motion module does not hold either network or driver device parameters. To execute saving/restoration, check the specifications of CPU module and the driver device and execute it.

Operation folder

When operating files by each function, a temporary file might be created for operating in the operating file (a folder which is indicated with the environmental variable%TEMP%). The user is not allowed to change and delete the data in the operating folder. (The error might occur in the functions under control.)
The operating folder is %TEMP% = /ram/temp by default. The place for %TEMP% can be changed by the pathset command of file transfer function. For handling large amount of files, change the operating folder to the SD memory card, etc. if the RAM drive lacks its capacity. However, the execution speed for file operating will be slower if other than RAM drive is used.

Precautions

Processing of reading/writing files

- Processing of reading/writing files is executed in the background processing of the motion system. Reading/Writing time is changed depending on the file size and the system load state. Therefore, the system startup time and the switching time from STOP to RUN may be longer.
- Writing time of the user drive may be longer if the free space of the drive decreases.
- The timeout error may occur during reading/writing files due to the communication failure with the engineering tool. In such a case, try the operation after a while.

Power OFF at file operation (including reset)

- While file operation excluding data reading, if the software is reset or the power of the motion system is turned OFF or reset, the file contents are not guaranteed. To prevent data damage, turn OFF the power (including reset) after the file operation is completed.
- Check the consistency of the file system at the motion system initialization (at inserting a card for an SD memory card). It may take time to check when there are many files in the drive.

Access to the same file at the same time

When the following operations are performed via engineering tool or SLMP, an error occurs. Execute the next access after the processing of one request source is completed.

- Execute access (reading/writing) from another request source to the file which is being writing.
- Execute writing from another request source to the file which is being accessing (reading/writing).

Access to the different files from multiple request sources at the same time

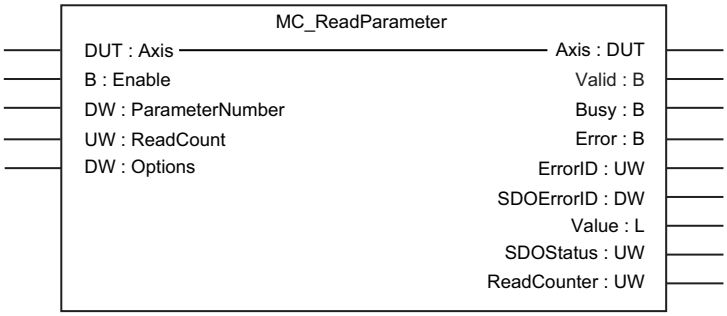
- Access to the different files from multiple request sources at the same time is up to 16 files.

19.2 Parameter Read/Write Function

This function can read/write parameters.
It can read and write objects of the slave devices.

Relevant FBs

MC_ReadParameter

Item	Description
Function overview	Read objects of the slave devices.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

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

■I/O variable

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Axis information	Axis	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables

■Input variables

Import ⎍: Always, ↑: At start

Name	Variable name	Data type	Import	Setting range	Default value	Description
Enable	Enable	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, the FB is executed.
Parameter No.	ParameterNumber	DWORD(HEX)	⎍	00010000H to FFFFFFFFH	00000000H	Indicates the objects of the slave devices.
The number of reading times	ReadCount	WORD(UNIT)	⎍	0 to 65535	0	Indicates the number of reading times of parameters. Carries out continuous read until Enable turns FALSE as 0 is set. Reload is executed at the timing which the Parameter No. is changed.
Options	Options	DWORD(HEX)	⎍	*1	00000000H	Sets function options with bit designation. Reload is executed at the timing which the Parameter No. is changed.

*1 For bits and the function description, refer to the following table.

Bit	Function description
0 to 15	Empty (Specify "0".)*1
16	Specifies unsigned Indicates whether read value of integer data is stored without a sign. 0: signed 1: unsigned
17 to 31	Empty (Specify "0".)*1

*1 If specifying other than "0", the error "Out of Options Range" (error code: 1A4EH) occurs, and the function does not start.

■Output variables

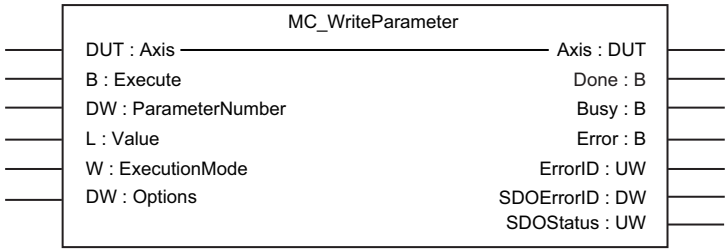
Name	Variable name	Data type	Default value	Description
Output value valid	Valid	BOOL	FALSE	When TRUE, the FB is executed.
Executing	Busy	BOOL	FALSE	Indicates that the FB is executing.
Error	Error	BOOL	FALSE	Indicates that an error has occurred in the FB as it is TRUE.
Error code	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned.
Transient error code	SDOErrorID	DWORD(HEX)	0	The error code (SDO Abort Code) generated by transient communication is returned.
Read value	Value	LREAL	0.0	Outputs the read value from the specified parameters. It is stored as LREAL type in the case that the target parameters are integer data as well.
SDO transfer status	SDOStatus	WORD(UINT)	0	Stores the processing status of transient request.*1
The number of reading times	ReadCounter	WORD(UINT)	0	Stores the number of reading times of parameters. The ring counter is to be "1 to 65535" in the case that "0" is set to ReadCount.

*1 For bits and the function description, refer to the following table.

Bit	Function description
0 to 7	Response object size (byte) Stores object size which the slave devices has responded when the process is completed.
8	The transient transmission acceptance 0: Unexecuted transient transmission 1: Accepting transient transmission
9	Empty (Specify "0".)*1
10	Issuing SLMP 0: Unexecuted SLMP 1: Issuing SLMP
11 to 15	Empty (Specify "0".)*1

*1 If specifying other than "0", the error "Out of Options Range" (error code: 1A4EH) occurs, and the function does not start.

MC_WriteParameter

Item	Description
Function overview	Write objects of the slave devices.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

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

■I/O variable

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Axis information	Axis	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables

■Input variables

Import ↑: At start

Name	Variable name	Data type	Import	Setting range	Default value	Description
Execution command	Execute	BOOL	↑	TRUE, FALSE	FALSE	When TRUE, the FB is executed.
Parameter No.	ParameterNumber	DWORD(HEX)	↑	00010000H to FFFFFFFFH	00000000H	Indicates the objects of the slave devices.
Setting value	Value	LREAL	↑	—	0.0	Indicates a setting value of the specified parameter.
Start mode	ExecutionMode	MC_EXECUTION_MODE	↑	0, 1	0	Indicates writing method of the specified parameter. As it is set to "0: mcImmediately", writes immediately. As it is set to "1: mcQueued", writes after the axis stops.
Options	Options	DWORD(HEX)	↑	*1	00000000H	Set the function options by specifying the bit.

*1 For bits and the function description, refer to the following table.

Bit	Function description
0 to 15	Empty (Specify "0".)*1
16	Specifies unsigned Indicates whether the setting value of integer data is defined as the integer value without a sign. 0: signed 1: unsigned
17 to 31	Empty (Specify "0".)*1

*1 If specifying other than "0", the error "Out of Options Range" (error code: 1A4EH) occurs, and the function does not start.

■Output variables

Name	Variable name	Data type	Default value	Description
Execution completion	Done	BOOL	FALSE	Indicates that writing to the parameter is completed.
Executing	Busy	BOOL	FALSE	Indicates that the FB is executing.
Error	Error	BOOL	FALSE	Indicates that an error has occurred in the FB as it is TRUE.
ErrorID	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned.
Transient error code	SDOErrorID	DWORD(HEX)	0	The error code (SDO Abort Code) generated by transient communication is returned.
SDO transfer status	SDOStatus	WORD(UINT)	0	Stores the processing status of transient request.*1

*1 For bits and the function description, refer to the following table.

Bit	Function description
0 to 7	Response object size (byte) Stores object size which the slave devices has responded when the processing is completed.
8	The transient transmission acceptance 0: Unexecuted transient transmission 1: Accepting transient transmission
9	Empty (Specify "0".)*1
10	Issuing SLMP 0: Unexecuted SLMP 1: Issuing SLMP
11 to 15	Empty (Specify "0".)*1

*1 If specifying other than "0", the error "Out of Options Range" (error code: 1A4EH) occurs, and the function does not start.

Slave object (Transient)

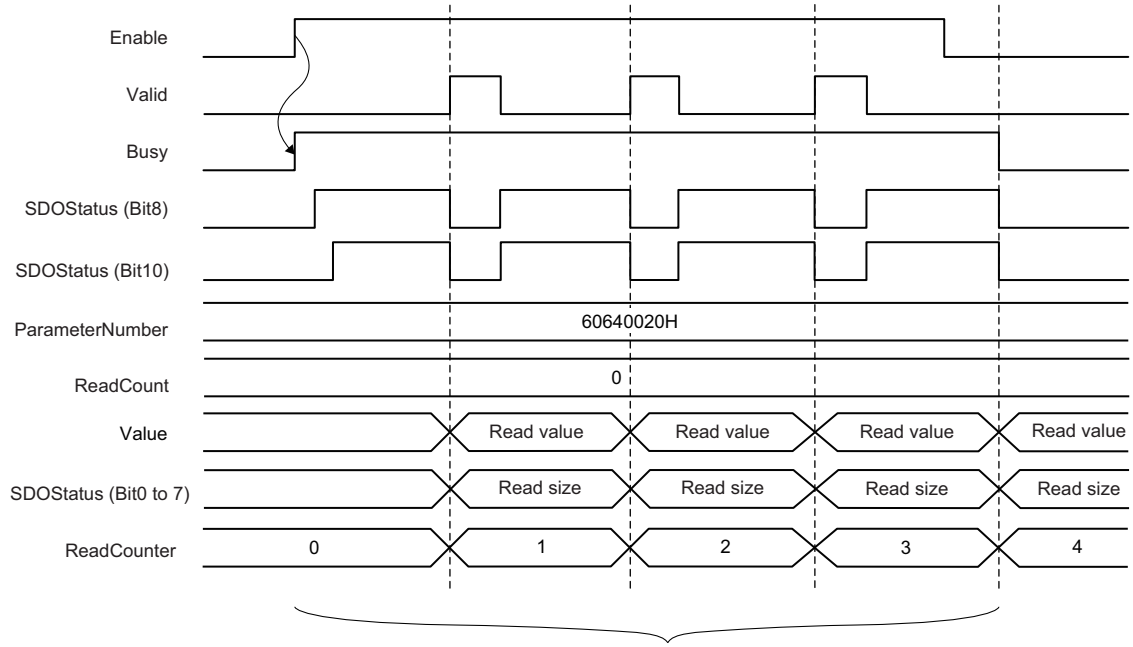
Slave objects can be read/written by indicating 00010000H or later to the parameter No. In this case, object data is sent/received by using the transient transmission function.

When the error (response code (SDO Abort Code)) for transient request is received from the slave device, Error turns TRUE and then the error "SDO Communication Error" (error code: 1800H) is stored to ErrorID and the response code (SDO Abort Code) from slave device is stored to SDOErrorID.

■The timing of send/receive

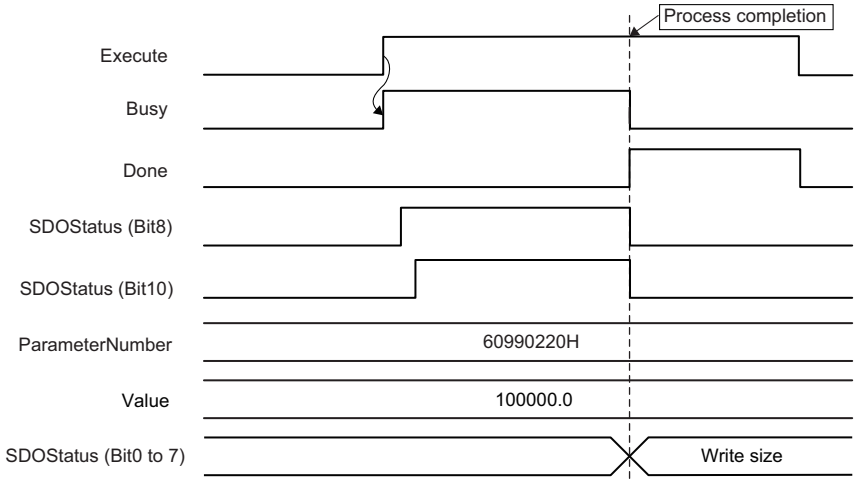
[At successful completion]

- For execuing parameter read



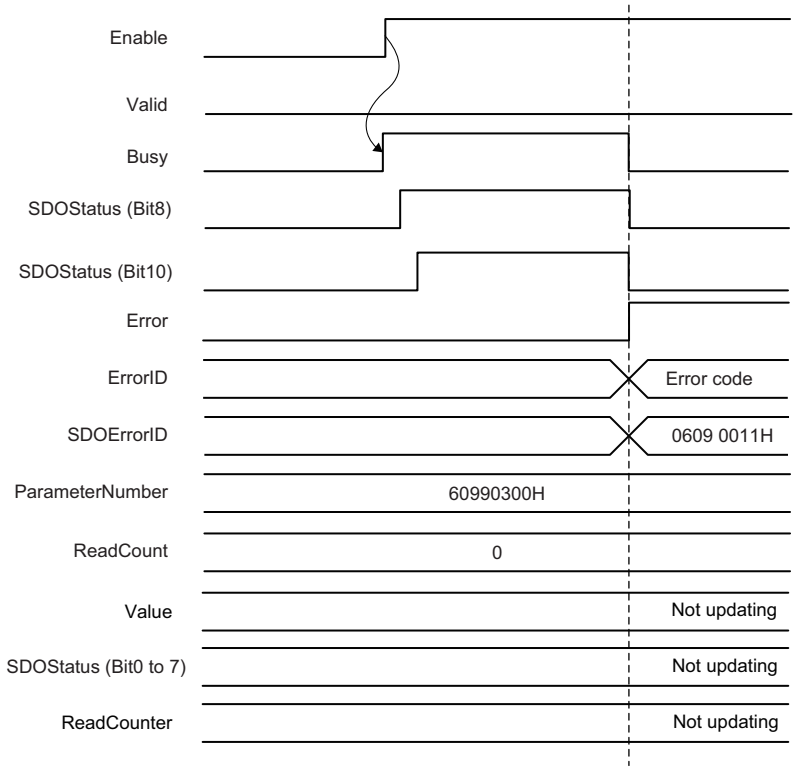
Executes read for a designated number which is indicated with ReadCount until Enable turns FALSE.
(Executes continuously until Enable turns FALSE when ReadCount is set to "0".)

- For execuing parameter write

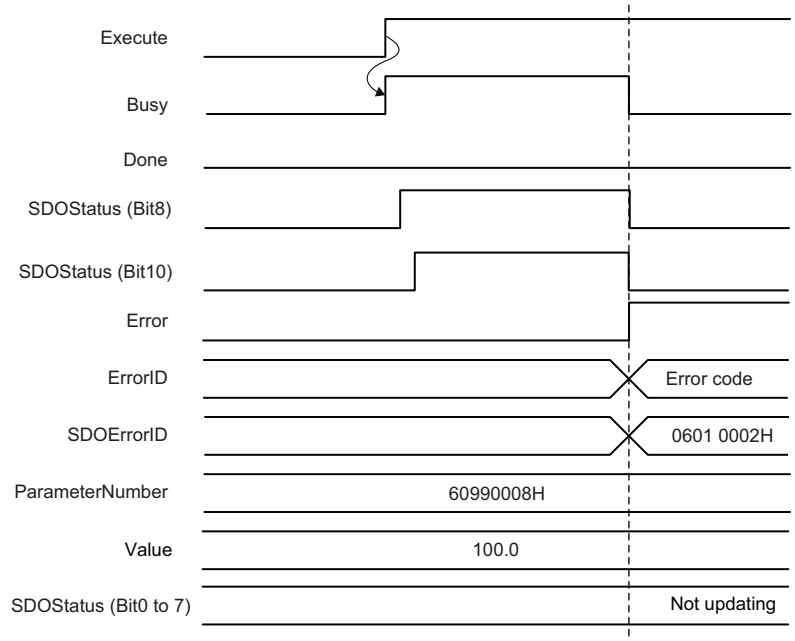


[At unsuccessful completion (SDO communication error has occurred.)]

- For execuing parameter read

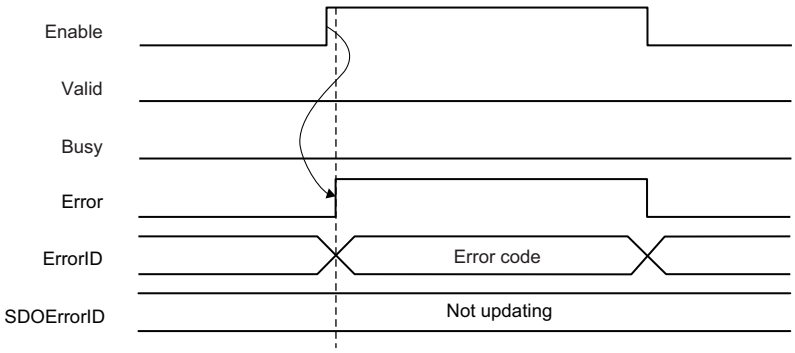


- For execuing parameter write

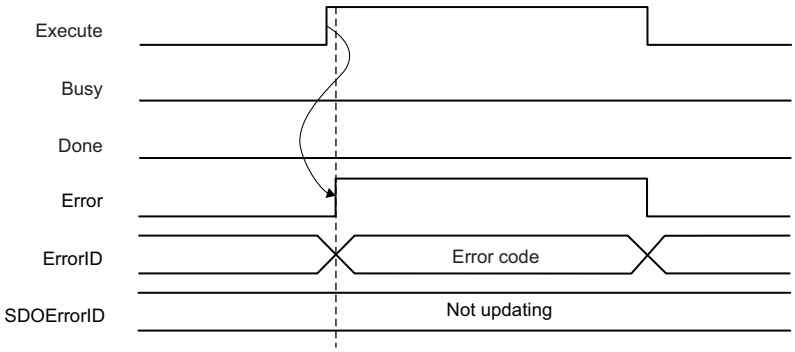


[At unsuccessful completion (the I/O variables and the Input variables are abnormal)]

- For execuing parameter read



- For execuing parameter write



Setting details

■ParameterNumber (parameter No.)

The Parameter No. specified at the parameter read/write is below.

- 00010000H to FFFFFFFFH: the objects of the slave device

For specifying the objects of the slave device, do as follows.

ParameterNumber = XXXXYYZZH

XXXX…Specify the object index.

YY…Specify the object sub index.

ZZ…Specify the object size (Number of bit).

For the size of the objects of the slave device, specify 1 byte / 2 bytes / 4 bytes / 8 bytes in bits.

If "0" is specified, the parameter read is performed with default size and the parameter write is performed with 4 bytes.

If the value other than above is set, the error "Out of Parameter No. Range" (error code: 34A2H) occurs.

When the size specified with ParameterNumber differs from default size of the slave objects, the operations depend on the specifications of the driver.

Ex.

For specifying object index 6099H and UNSIGNED32 object of subindex 02H

For read: Specify "60990200H" (default size).

For write: Specify "60990220H" (the size of 4 bytes)

For details, refer to the manuals of each drive unit.

■ExecutionMode (Start mode)

At executing the parameter write, set the timing of write in ExecutionMode.

If "0: mcImmediately" is specified, write is executed immediately. The axis operations might be affected because the reflection timing depends on each parameter.

If "1: mcQueued" is specified, write is executed after the axis stops (the axis status (AxisName.Md.AxisStatus) is "0: Disabled" or "4: Standstill").

The response code (SDO Abort Code)

The response code by using the transient transmission function when the object data send/receive is unsuccessful is shown below.

SDO Abort Code	Description	Remedy
0601 0000h	Accessed to an unsupported object.	Check Index and subIndex.
0601 0001h	Performed read access to a write-only object.	
0601 0002h	Performed write access to a read-only object.	
0602 0000h	Accessed to an object which is not defined in the object dictionary.	
0604 0041h	Performed the mapping of an object which is not allowed PDO mapping.	Check the mapping data.
0604 0042h	The number and length of the data which is mapped exceed the value defined by the application.	
0609 0011h	Specified SubIndex that does not exist.	Check Index and subIndex.
0609 0030h	Set an invalid parameter value	Check the data value.
0609 0031h	Set the larger value than the parameter range.	
0609 0032h	Set the smaller value than the parameter range.	
0800 0020h	The application cannot transfer or store data.	Check the status of the target device.

Precautions

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

Ex.

Write parameters by specifying two bytes size with MC_WriteParameter.

Setting value (Value(LREAL))	Write value (2 bytes data)	
	Options: bit16 "0: Signed"	Options: bit16 "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

Read parameters by specifying two bytes size with MC_ReadParameter.

Parameter value	Read value (Value(LREAL))	
	Options: bit16 "0: Signed"	Options: bit16 "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

- The changed values by this function are not saved. To use the changed values at the next power supply, save the parameters.
  For slave object devices: Refer to the mannual for slave devices about the parameter saving method.

19.3 File Transfer Function

This function executes file operation and data backup/restore based on the specified command.
This function executes the file operation according to the command description set in System.Cd.FileTransfer_Command by setting TRUE in System.Cd.FileTrasfer_Execute.
While the file transfer is executed, System.Md.FileTransfer_State is set to Executing.
The execution of file transfer is registered in the event history (type: system).

Operation of this function at each status of system

○: Possible

Status	Operation availability
STOP	○
RUN	○
Moderate error	○
Major error	○

Relevant variables

Variable/Structure name	Name	Details
System.PrConst.		
FileTransfer_LogCapacity	File transfer log capacity	Specifies the log file capacity in a unit of k byte.
FileTransfer_AcFile_lch	File transfer access control (File in latch drive)	Specifies the access control to files in the latch drive. For details, refer to the following. ☞ Page 614 Access control setting
FileTransfer_AcFile_ram	File transfer access control (File in RAM drive)	Specifies the access control to files in the RAM drive. For details, refer to the following. ☞ Page 614 Access control setting
FileTransfer_AcFile_rom	File transfer access control (File in user drive)	Specifies the access control to files in the user drive. For details, refer to the following. ☞ Page 614 Access control setting
FileTransfer_AcFile_sdc	File transfer access control (File in SD memory card)	Specifies the access control to files in the SD memory card. For details, refer to the following. ☞ Page 614 Access control setting
FileTransfer_AcFile_sys	File transfer access control (File in system drive)	Specifies the access control to files in the system drive. For details, refer to the following. ☞ Page 614 Access control setting
FileTransfer_AcLabel	File transfer access control (Label)	Specifies the access control to labels. For details, refer to the following. ☞ Page 614 Access control setting
System.Cd.		
FileTransfer_Execute	File transfer execution request	Requests the execution of file transfer. When the file transfer is completed, FALSE is set automatically.
FileTransfer_Command	File transfer command	Specifies the command of file transfer. When the file transfer is completed, it is cleared automatically. For details, refer to the following. ☞ Page 615 Command format
System.Md.		
FileTransfer_State	File transfer executing state	Displays the execution state of file transfer. 0: Ready (Waiting for the request) 1: Executing (Executing) -1: Error_ (Error occurrence)

Access control setting

Set the availability of data transfer by the file transfer function using the engineering tool.

- Set the access permission to each data by the WORD value shown below.

Bit	Meaning
0	Write permission (1: Allowed, 0: Prohibited)
1	Read permission (1: Allowed, 0: Prohibited)
2 to 15	Empty (Specify "0".)*1

*1 If specifying other than "0", the error "Out of Options Range" (error code: 1A4EH) occurs, and the function does not start.

- For details on the access permission required for command execution, refer to specifications of each command.
- The access control setting read at the last is enabled at the execution timing of the file transfer function.

Functions of file transfer

Operations to be specified by the file transfer function are as follows.

Function (Command string)	Operation	Remark
Data copy (copy)	Copies a pecificated file, a file in a specificated folder, or a subfolder.	When a file with the same name exists in the copy destination, the file is overwritten.
Data move (move)	Moves a pecificated file, a file in a specificated folder, or a subfolder.	When a file with the same name exists in the destination, the file is overwritten. The source file is deleted.
Data delete (delete)	Deletes the specified file or the specified folder.	—
Environmental variable specification (pathset)	Specifies the data path to be used by each function. It can read the file stored in another folder and start the system without change of the user drive.	—
Filing (file)	Performs filing of label values.	• When a file with the same name exists in the specified path destination, the file is overwritten. • Uses the operation folder during command execution.
Data compression (compress)	Compresses a pecificated file or a file in a folder. (The name of compressed file is arbitrary.)	• When a compressed file with the same name exists in the destination, the file is overwritten. The source file is not deleted. • Uses the operation folder during command execution.
Data extraction (extract)	Extracts a compressed file to a specified folder.	When a file with the same name exists in the destination, the file is overwritten.
File attribute (attrib)	Changes a file attribute of a specified folder.	—

Command format

Specify the command to be specified by the file transfer function with the following formats.

[Command] [Argument 1] [Argument 2] [Argument 3]

- For details on argument, refer to the description of each command.
- For a delimiter between commands and arguments, write a space.
- Multiple commands can be specified by using a newline code (\\n) as a delimiter. The number of commands to be specified at the same time is not restricted.

Writing example:

```
copy /rom/$MOTPRJ$/* /sdc/$MOTPRJ$/ \\n delete /rom/$MOTPRJ$/
```

- For details on path that can be specified as an argument, refer to the following.
  ☞ Page 599 Path setting
- If a space is included in the path, round the path with double quotation (").
- The path is not case-sensitive.
- When "//" is written before a command, the command is ignored. If the next command exists, the execution continues.
- When the command failed to execute, System.Md.FileTransfer_State turns "Error" and the execution ends.
- When the command format is abnormal, System.Md.FileTransfer_State turns "Error" and the execution ends.
- For the operation to the file set with the read only attribute, refer to the following.
  ☞ Page 601 Operation that can be executed
- When writing commands in ST program, make sure not to include characters which can not be expressed as a character string type constant in the argument or the path, etc. Replacing might be required. For details, refer to the following.
  GX Works3 Operating Manual

Data copy / Data move

■Format

Command	copy or move
Argument 1	• Specify the source data path of the target for copy / move • Wild card characters (*: arbitrary character string, ?: arbitrary single character) enable specification of multiple data. • All data in the specified folder (including the data in the sub directory as well) is copied or moved as the folder is specified. The specified folder itself is not targeted for copy or move. • For copy, read permission to the file is required. • For move, read and write permission to the file is required.
Argument 2	• Specify the folder path of copying or moving destination. Indicates the folder when the path ends in "/" and the file when the path does not end in "/". • As argument 1 is a single file, specifies the move destination folder (ends in "/") or the file (not end in "/"). • As argument 1 is multiple files (including the folder), specifies the move destination folder (ends in "/"). • Write permission to the file is required. • When moved to the same drive, the free space of the drive for the source data size is required.
Argument 3	• Specify whether the output to the log file is enabled. • Note that the argument 3 must not be specified to output the log. • Note that specifying "nolog" not to output the log.

■Writing example

- Copies IFG of the "/rom/$MOTPRJ$/" folder to the "/sdc/$MOTPRJ$/" folder.

```
copy /rom/$MOTPRJ$/param/*.csv /rom*.IFG/sdc/$MOTPRJ$/
```

- Moves all the data in the "/rom/$MOTPRJ$/" folder to the /sdc/$MOTPRJ$/" folder without leaving the log.

```
move /rom/$MOTPRJ$/ /sdc/$MOTPRJ$/ nolog
```

- Renames ProfileData0001.CSV to ProfileData0002.CSV.

```
move /rom/$MOTPRJ$/calc_profile/PD01/ProfileData0001.CSV /rom/$MOTPRJ$/calc_profile/PD01/ProfileData0002.CSV
```

Data delete

■Format

Command	delete
Argument 1	• Specify the data path of the delete target. • Wild card characters (*: arbitrary character string, ?: arbitrary single character) enable specification of multiple data. • All data in the specified folder (including the data in the sub directory as well) is deleted as the folder is specified. • Writing permission to the file is required.
Argument 2	• Specify whether the output to the log file is enabled. • Note that the argument 2 must not be specified to output the log. • Not to output the log, specify "nolog".

■Writing example

• Delete the MTNSTR.STR of "/rom/$MOTPRJ$/" folder.

```
delete /rom/$MOTPRJ$/MTNSTR.STR
```

Environmental variable specification

■Format

Command	pathset
Argument 1	• Specify the environmental variable. • For details on the environmental variable to be specified, refer to the following. ☞ Page 600 Environmental variables • The specified environmental variable has been enabled until the power ON/reset or the software reset of the motion area.
Argument 2	• Specify the file/folder path to be set to the environmental variable. • Specify "/cpu" to use the data of the CPU module. • Wild card characters (*: arbitrary character string, ?: arbitrary single character) enable specification of multiple data.
Argument 3	• Specify whether the output to the log file is enabled. • Note that the argument 3 must not be specified to output the log. • Not to output the log, specify "nolog".

■Writing example

• Specify the root path to the "/sdc/$MOTPRJ$/" folder.

```
pathset PROJECT_ROOT /sdc/$MOTPRJ$/
```

Filing

■Format

Command	file
Argument 1	• Specify the target data for filing. The target data and argument are shown below. Label: label • For the global label of the filing target, refer to the following. ☞ Page 701 List of Variables • Read permission to the target data is required.
Argument 2	• Specify the output destination of the filing data. • Outputting the file with the specified name is allowed as specifying the arbitrarily named file. • For the global label of the filing target, refer to the following. (The label whose attribute is LIST_WRITE_BACK is the filing target.) ☞ Page 701 List of Variables • Write permission to the target data is required.
Argument 3	• Specify whether the output to the log file is enabled. • Note that the argument 3 must not be specified to output the log. • Note that specifying "nolog" not to output the log.

Point

- The label value return to an initial value as the system is rebooted with the different value from the label initial value due to rewrite of the label value. The backuped label value at rebooting can be used as filing the label value (backup to the label default fail) by the file command. An error does not occur if a value out of setting range is written at backup. Do not write the value of the out of setting range. It may cause an unexpected operation of the system.
- When files before the import are specified as the target data, they may cause filing failure (such as when local labels are specified before PLC READY [Y0] is ON). Execute it after the import.
- When replacing the motion system, all the data in the motion system including the absolute position data is backuped and restored (write) again to the motion system by the backup/restore function of the engineering tool.

■Writing example

- Files the global label.
  Files (Backup to the file with a label initial value (GLBLINF.LID)) the label value.

```
file label /rom/$MOTPRJ$/GLBLINF.LID
```

Data compression

■Format

Command	compress
Argument 1	• Specify the target file for compression. • Wild card characters (*: arbitrary character string, ?: arbitrary single character) enable specification of multiple data. • All data in the specified folder (including the data in the sub directory as well) is compressed as the folder is specified. • Read permission to the file is required.
Argument 2	• Specify the path witch includes the file name after compression. • Specifies the following for the compressed file extension depending on the intended use. Zip file format: zip • Write permission to the file is required. • If the file password is set for the file specified by argument 1, compress the file as it is (the file password is set for the file). In this case, the compressed file itself (*.zip or *.prm) is not added the file password.
Argument 3	• Specify whether the output to the log file is enabled. • Note that the argument 3 must not be specified to output the log. • Note that specifying "nolog" not to output the log.

■Writing example

- Compreees the data in the "/rom/$MOTPRJ$/" folder and saves as the compressed file (compress.zip) in the "/sdc/$MOTPRJ$/" folder.

```
compress /rom/$MOTPRJ$/* /sdc/$MOTPRJ$/compress.zip
```

Data extraction

■Format

Command	extract
Argument 1	• Specify the target file for extraction. • Read permission to the file is required. • If the file password, "read protection" is set to the file in the compressed file, authentication is required.
Argument 2	• Specify the path for the extracted folder destination. • Write permission to the file is required. • If the same named file of the extracted destination is the read-only attribute, System.Md.FileTransfer_State turns "Error".
Argument 3	• Specify whether the output to the log file is enabled. • Note that the argument 3 must not be specified to output the log. • Note that specifying "nolog" not to output the log.

■Writing example

• Extracts the compressed file (compress.zip) in the "/sdc/$MOTPRJ$/" folder to the "/rom/$MOTPRJ$/".

```
extract /sdc/$MOTPRJ$/compress.zip /rom/$MOTPRJ$/
```

File attribute

■Format

Command	attrib
Argument 1	Specify the path for the target file which is changed the file attribute.
Argument 2	• Specify the file attribute to be changed. • When setting the read-only attribute, specify "+r". • When clearing the read-only attribute, specify "-r". • This argument cannot be omitted. • Write permission to the file is required.
Argument 3	• Specify whether the output to the log file is enabled. • Note that the argument 3 must not be specified to output the log. • Note that specifying "nolog" not to output the log.

■Writing example

• Makes "/rom/$MOTPRJ$/PRG.PRM" the read-only file.

```
attrib /rom/$MOTPRJ$/PRG.PRM +r
```

Log file

When the file transfer function is executed, the log file (filelog.txt/filelog.bak) is stored in the directory (Default is user root) that the environmental variable FILE_TRANS_LOG indicates at the timing of the execution completion. When the environmental variable is changed by pathset command, the log file is created to the path after change.
The capacity of the log file can be specified by System.PrConst.FileTransfer_LogCapacity and the operation depends on the setting value as follows.

FileTrasfer_LogCapacity (file transfer log capacity) setting	Log create operation
-1	Create a new file in each file transfer execution. The existing filelog.txt is renamed to filelog.bak.
0	Not created.
1 to 2048 [K bytes]	Add writing to a file in each file transfer execution. When filelog.txt exceeds the file capacity, filelog.txt is created after filelog.txt is renamed to filelog.bak.

When the log file (filelog.txt) error is detected, the warning "Log file create disabled warning" (warning code: 0F01H) is output. After the warning is output, the file transfer execution is maintained without log files.

Precautions

- Note that the writing to the file that is being used by the control during PLC ready [Y0] ON may cause unexpected operation. Besides, reading of the file that is being operated by each function may cause lost of the data consistency. Therefore, check the file is not operated and execute the file transfer function.
- The file transfer function is used when the project data is read or written from the engineering tool. While the project data is operated by the engineering tool, note that not write to System.Cd.FileTransfer_Execute and System.Cd.FileTransfer_Command.
- Note that the CPU module must not be reset and that the power must not be turned OFF during the file transmission. If the power is turned OFF or the CPU module is reset during the file transmission and the processing is forced interrupted, the file being transmitted may be broken.
- Some commands such as compression use the operation folder. The error may occur due to the insufficient capacity of the operation folder caused by large files and folder compression, etc. In such a case, change the operation folder to an SD memory card, etc. (The operation folder can be specified by changing the environmental variable %TEMP%).

19.4 SD Memory Card

This section describes functions using an SD memory card.

Operation of this function at each status of system

○: Possible

Status	Operation availability
STOP	○
RUN	○
Moderate error	○
Major error	○

Relevant variables

Variable/Structure name		Name	Details
System.Cd.			
	Storage_sdcRemovalProhibit	SD memory card removal prohibition	Specifies whether the removal of SD memory card is prohibited.
	Storage_sdcForcedDisable	SD memory card force disabled command	Specifies whether the usage of SD memory card forced use stops.
System.Md.			
	Storage_sdc	SD memory card information	Indicates the SD memory card information. STORAGE_INFORMATION structure
	Storage_sdcProtected	SD memory card protected	Indicates whether the write protect switch of the SD memory card is ON/OFF.
	Storage_sdcInserted	SD memory card inserted	Indicates whether the SD memory card is inserted.
	Storage_sdcForcedDisabled	SD memory card force disabled	Indicates whether the usage of SD memory card is being forced stopped.
STORAGE_INFORMATION			
	Capacity	Capacity	Stores the capacity of the SD memory card in a unit of 1K bytes. (Stores the free space after formatting.)
	FreeSpace	Free space	Stores the free space of the SD memory card in a unit of 1k bytes.
	Mount	Mount status	Stores the mount status of the SD memory card.

SD memory card handling

Format an SD memory card which is used with the motion system from the CPU module. If formatting an SD memory card using the function such as the format function of Windows®, an SD memory card may not be able to be used with the motion system.

Data contained in an SD memory card may be corrupted if the power supply is turned OFF, the system is reset, or the SD memory card is removed while the card is being accessed. If the CARD ACCESS LED is ON, always stop access to the SD memory card with the SD memory card access control switch before turning OFF the power supply, resetting, or removing the SD memory card.

Specifications of SD memory card

The SD memory card is handled as the drive in the internal of the motion system and the engineering tool.

For the specifications of SD memory card, refer to the following.

☞ Page 596 Memory and File

Installation and removal of SD memory card

For details, refer to "Installation and Removal of SD Memory Card" in the following manual.

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

Precautions

- If each function accesses to the SD memory card during the SD memory card use stop status, the operation is the same as the operation when the SD memory card is not inserted.
- If the SD memory card is stopped while writing a file to the SD memory card from an external device, file writing may fail. Write the file again after cancelling the SD memory card use stop status.

20 SECURITY

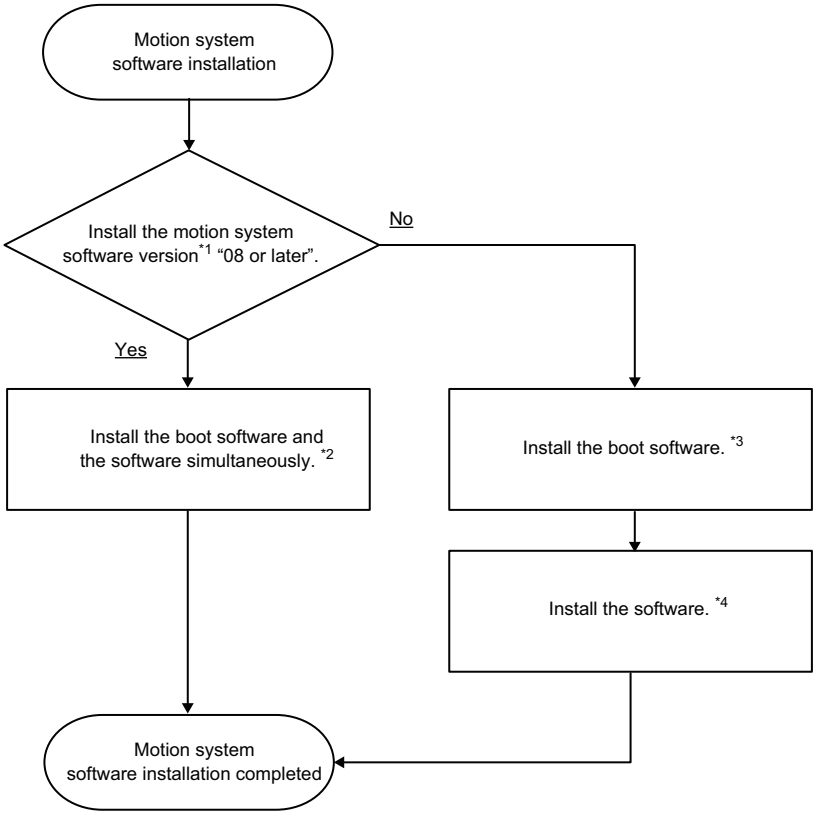
This function serves to protect the user property stored in a personal computer and the user property inside modules in the MELSEC iQ-R series system against threats such as theft, faulty operation, and unauthorized execution due to the unauthorized access by an outsider. Use an appropriate security function according to the purpose as shown in the following table.

Security function	Purpose
IP filter	Blocks access from an invalid IP address by identifying the IP address of an external device connected by Ethernet. Thus, unauthorized access to the system can be prevented. For derails, refer to the CPU module manual.
Remote password	Limits access to just specific communication routes connected by Ethernet. (A password is used.) Thus, unauthorized access from a remote place to the system via Ethernet can be prevented. For derails, refer to the CPU module manual.

MEMO

21 MODULE SOFTWARE INSTALLATION

How to install software of the motion system is explained in this chapter.
To install the motion system software, follow the flow below and execute.



*1 Indicates a version of motion system software to be installed. It is not a version of the installed motion system software.
*2 For details, refer to the following.
☞ Page 627 How to batch install the motion system software
*3 For details, refer to the following.
☞ Page 635 How to update the boot software
*4 For details, refer to the following.
☞ Page 631 How to install the software

21.1 Motion System Software Installation

The software is installed in the motion system when the product is shipped. Therefore, users do not have to install the software. However, when updating or changing to the new software, users have to install the software again. The latest software can be downloaded from MITSUBISHI ELECTRIC FA Global Website.

Point

- Even if carrying out installation, programs, parameters, and absolute position data that are written in the motion system are not rewritten.
- If installation is interrupted by the following operations during installation execution, the file in installation is deleted. Execute installation again.
  "Turning OFF the power of the motion system"
  "Turning RUN/STOP/RESET switch of the CPU module to RESET"
  "Turning OFF the power of the personal computer (when using the engineering tool)"
  "Disconnecting a communication cable from the personal computer (when using the engineering tool)"
  "Removing the SD memory card (when using SD memory cards)"
- The installed software is reflected when the power is turned ON again.
- While installation is executed, installation operation is not accepted by the other installation method. For example, while installation by the SD memory card is executed, installation executed by the engineering tool will cause an error.
- The type name and the version of the motion system software which is installed to the motion system can be checked with the engineering tool.
- The software files and the add-on files are protected by CRC. The consistency of the file is checked when turning on the power of the motion system again, then if an error is detected, the error "Add-on Library Load Error" (error code: 3205H) will occur and the system will not start. Execute installation again to install the correct file.
- The combination of the software version should be matched with the one mentioned in the MITSUBISHI ELECTRIC FA Global Website or reference below.
  ☞ 27 List of Add-on Library
  If the combination of the software version does not match, the error "Add-on Library Load Error" (error code: 3205H) occurs and the system does not start. Execute installation again with the correct combination of the software version.

Relevant variables

Variable/Structure name		Name	Details
System.Md.			
	Version_BootSw	Boot software version	The version of the boot software
	Version_NetworkBootSw	Network boot software version	The version of the network boot software
	Version_BaseSystemSw	Basic system software version	The version of the basic system software

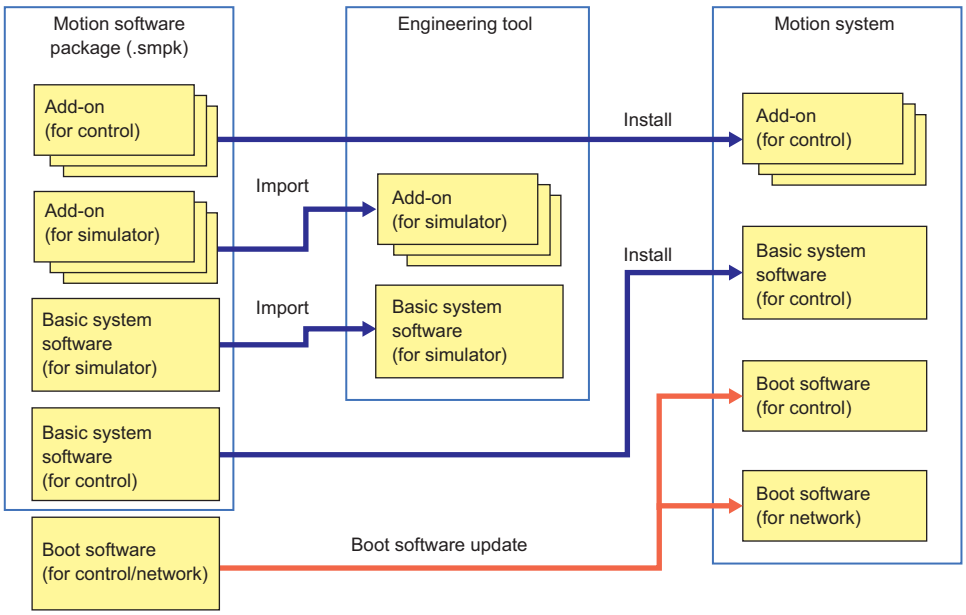
Software file configuration of motion system

The latest version software for the motion system is included in the motion software package, and these are managed by the engineering tool. The latest version functions of the motion system software will be available by installing the motion system software to the engineering tool and the motion system.

The motion system software is saved in the motion software package (.smpk) format.

For file configuration of the add-on that is to be installed in the motion system, refer to the following.

☞ Page 118 Add-on Function



Motion software package file configuration details

The following files are packed in the motion software package. The motion software package is the same file between the RD78G and the RD78GH.

Folder	File name
Motion software package for the RD78G(H)	Basic system software binary
	Basic system software simulator DLL
	Data definition file (axis data, etc.)
	Add-on binary
	Add-on simulator DLL
	The engineering tool window definition file for add-on
	Boot software binary for network
	Bood software simulator DLL for network

The files are opened as the following on the motion system.

Folder	File name
/sys drive	Basic system software binary
	Add-on binary

The boot software is not recognized as a drive.

How to batch install the motion system software

This section describes how to install motion system software downloaded from the MITSUBISHI ELECTRIC FA Global Website.

To install the version of motion system software, "08 or later", install the software and the boot software simultaneously by batch installation.

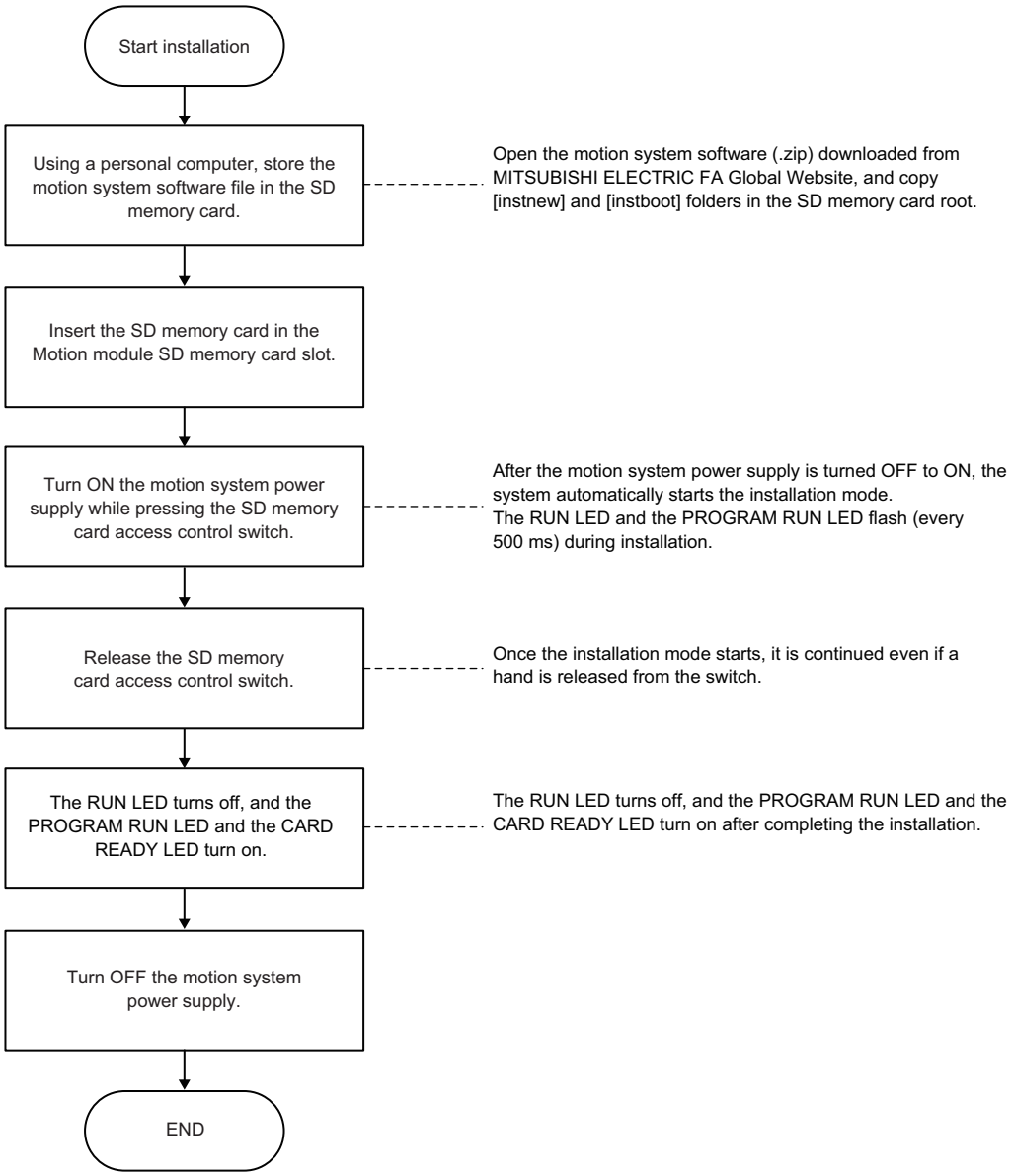
The batch installation can be executed only from the SD memory card. When the batch installation is executed, the seperate installtaion of the software and the boot software is not required.

To install the version of motion system software, "07 or earlier", install the software and the boot software seperately because the batch installation is not supported. For the installation, refer to the following.

☞ Page 631 How to install the software, ☞ Page 635 How to update the boot software

Point

To install the software seperately, update the boot software first, and then install the software.



Point

- Do not install the folder combined [instnew] and [instboot] in the different motion software packages.
- When installation could not be normally completed because an error was detected after installation was started, the RUN LED turns OFF, and the PROGRAM RUN LED and the CARD READY LED are flashing (every 200 ms). After dealing with the error, carry out installation again. When reading the SD memory card is abnormal, check the card.
  When the installation file is abnormal or when it cannot be recognized, check the installation file.
  When the system folder is abnormal, format it with instnew then install the file again.
- The forced stop function of the SD memory card cannot be used during installation execution from the SD memory card. Do not remove the SD memory card during installation.
- The ready [X0] and the synchronization flag [X1] do not turn ON during installation execution from the SD memory card.

□: OFF, ■: ON, ●: Flashing

Motion system status	LED display	Description	Remedy
Normal operation	RUN● P RUN● C RDY□	Flashing (every 500 ms): Executing the installation	Wait until the installation is completed.
	RUN□ P RUN■ C RDY■	Turning on: the installation is normally completed	Turn the power of motion system OFF to ON, and check that the CPU module can be normally turned RUN/STOP.
Operation failure	RUN□ P RUN● C RDY●	Flashing (every 200 ms): the installation is completed with an error.	Turn the power of motion system OFF to ON, and execute the installation again.

Checking software version

The installed software version can be checked with the engineering tool.

Checking the basic system software version

The basic system software version can be checked with System.Md.Version_BaseSystemSw or the engineering tool.

■Checking with the engineering tool

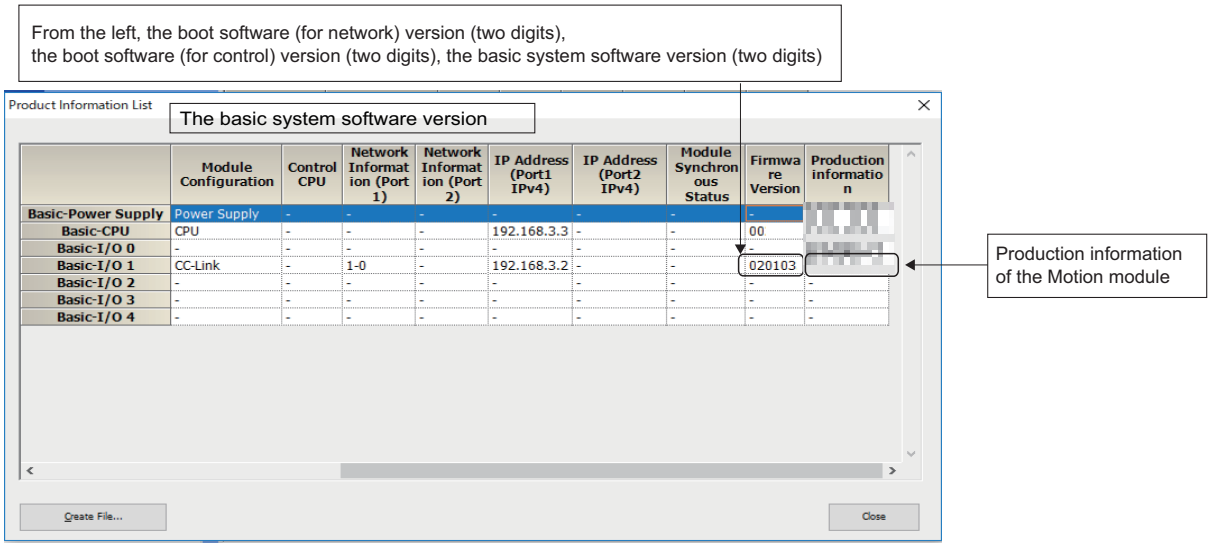
Check on the product information list screen.

[Diagnostics] ⇨ [System monitor] ⇨ [Product Information List]

The basic system software and boot software (for control / for network) versions are displayed.

Point

When the version of motion system software is "1.7 or earlier", only the version (2 digits from the left) of the boot software (for network) is displayed immediately after start (add-on loading) of the Motion module.



Checking the add-on version

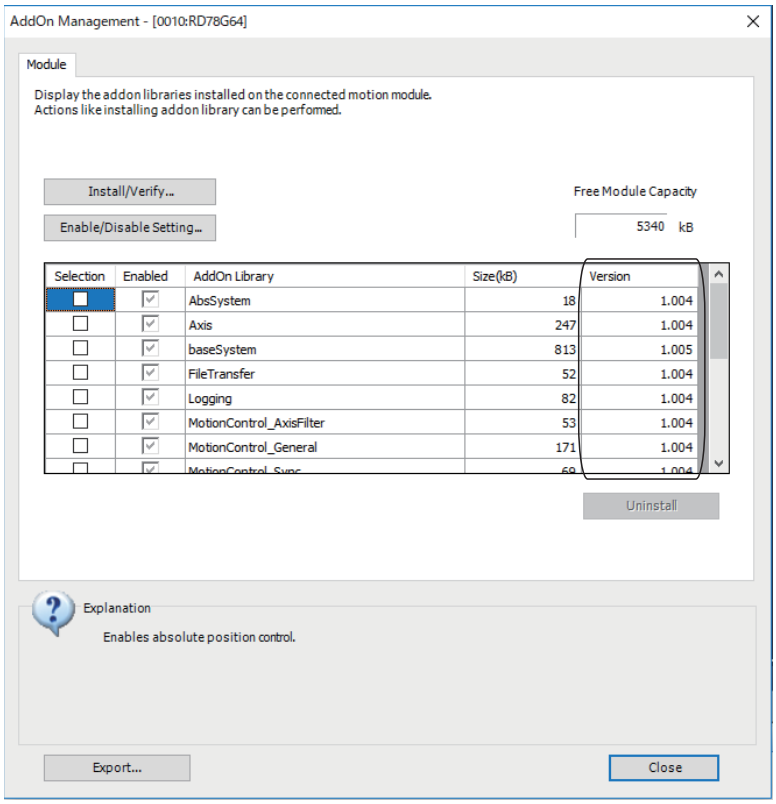
Check the add-on version on the engineering tool.

■Checking with the engineering tool

Check on the add-on management screen.

Navigation window ⇨ "Module Extended Parameter" ⇨ "Tool" ⇨ "AddOn Management..." ⇨ [AddOn Management]

The add-on version is displayed.



Checking the boot software version

The boot software version can be checked with System.Md.Version_BootSw.

21.2 Basic System Software Update Function

Software management

How to install the software

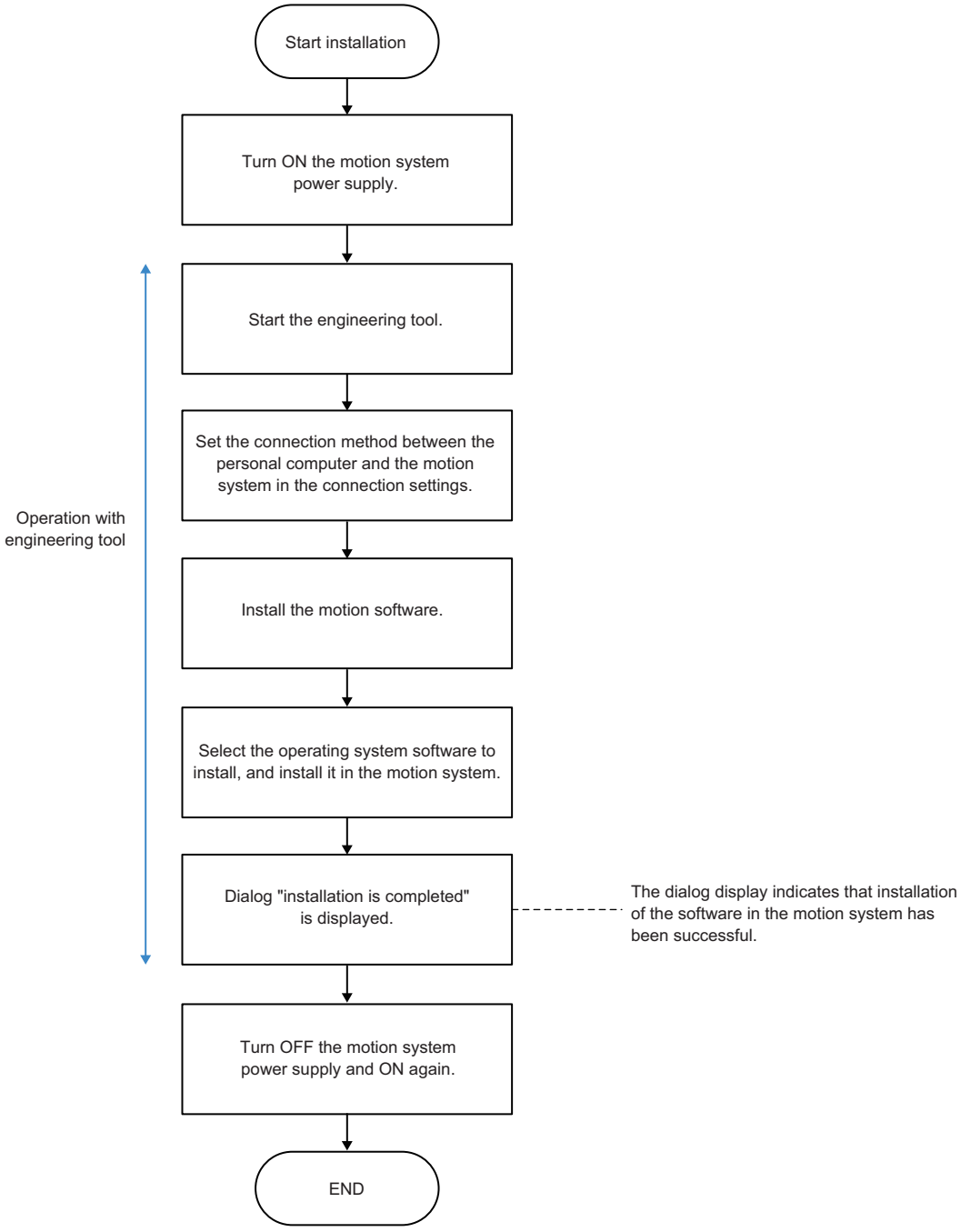
The following explains a procedure for installing the software.

The security function is valid regardless of the installation method. Installation cannot be carried out depending on the setting. For details, refer to the following.

☞ Page 622 SECURITY

■Installation procedure using the engineering tool

The following shows a procedure to install the software by using the engineering tool.



Install the software according to the instructions shown on the screen of the engineering tool. For details, refer to help of the engineering tool.

Point

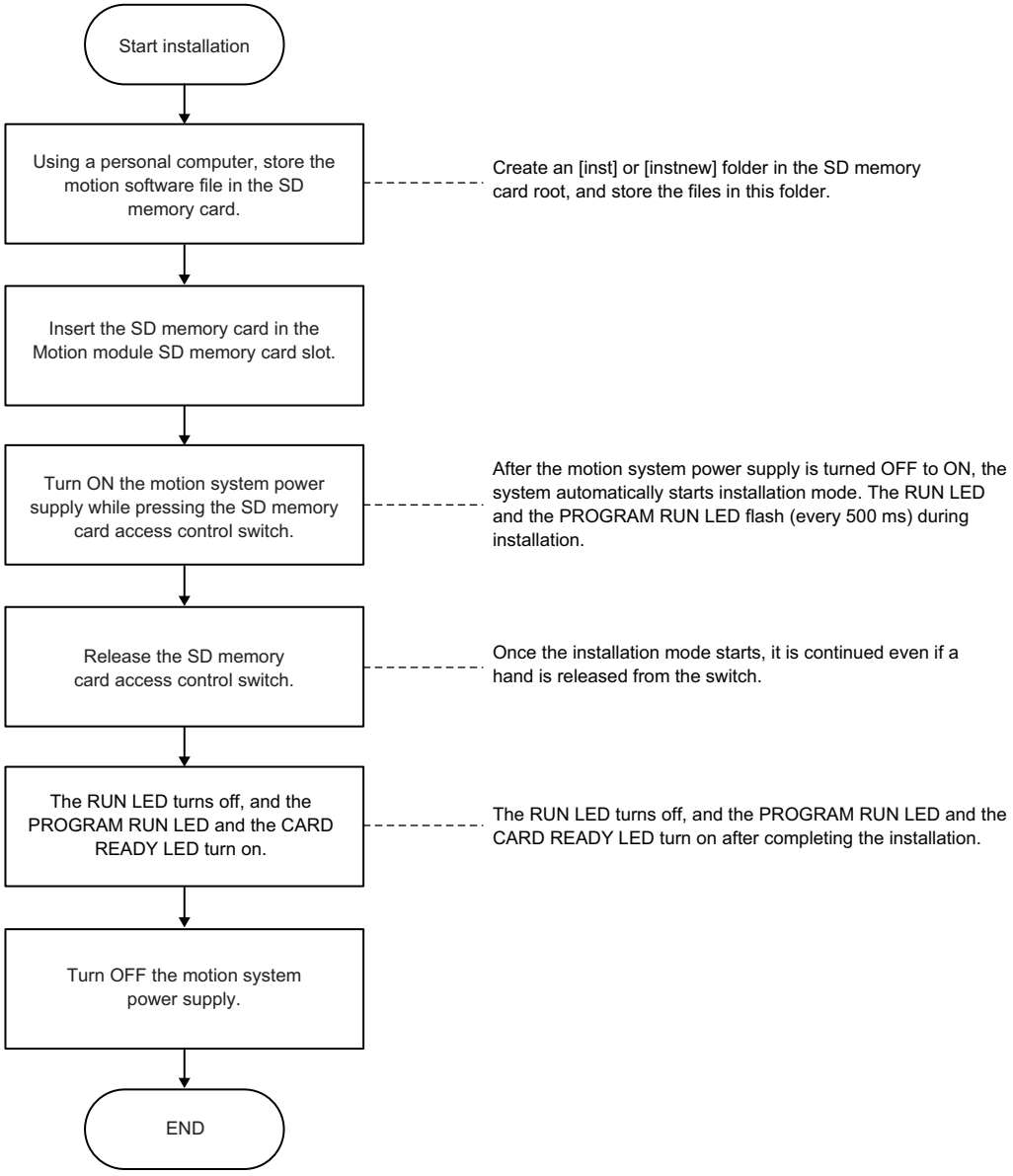
- Each function stored in the motion software package can be selected for the execution of the installation.
- Add-ons which have been installed are not deleted.

■Installation procedure using an SD memory card

The following shows a procedure to install the software by using an SD memory card.

Put the motion software package in the specified folder of the SD memory card.

When selecting files to be installed, create the motion software files to be installed with the engineering tool in advance.



Point

- The motion software creates the [inst] or [instnew] folder in the root of the SD memory card. Store the file in the folder.
  [instnew] folder: When newly installing the file (The file is transmitted after the system drive is formatted.)
  [inst] folder: When updating (overwriting) and when installing an additional module
- When installation could not be normally completed because an error was detected after installation was started, the RUN LED turns OFF, and the PROGRAM RUN LED and the CARD READY LED are flashing (every 200 ms). After dealing with the error, carry out installation again.
  When reading the SD memory card is abnormal, check the SD memory card.
  When the installation file is abnormal or when it cannot be recognized, check the installation file.
  When the system folder is abnormal, format it with instnew then install the file again.
- The forced stop function of the SD memory card cannot be used during installation execution from the SD memory card. Do not remove the SD memory card during installation.
- The ready [X0] and the synchronization flag [X1] do not turn ON during installation execution from the SD memory card.

□: OFF, ■: ON, ●: Flashing

Motion system status	LED display	Description	Remedy
Normal operation	RUN● P RUN● C RDY□	Flashing (every 500 ms): Executing software installation	Wait until software installation is completed.
	RUN□ P RUN■ C RDY■	Turning on: Software installation is normally completed	Turn the power of motion system OFF to ON, and check that the CPU module can be normally turned RUN/STOP.
Operation failure	RUN□ P RUN● C RDY●	Flashing (every 200 ms): Software installation is completed with an error.	Turn the power of motion system OFF to ON, or reset the system, then execute software installation again.

21.3 Boot Software Update Function

This function updates the boot software which starts up the motion system. It is used when the boot software needs to be updated due to the restrictions of the motion software.
The latest boot software can be downloaded from MITSUBISHI ELECTRIC FA Global Website.

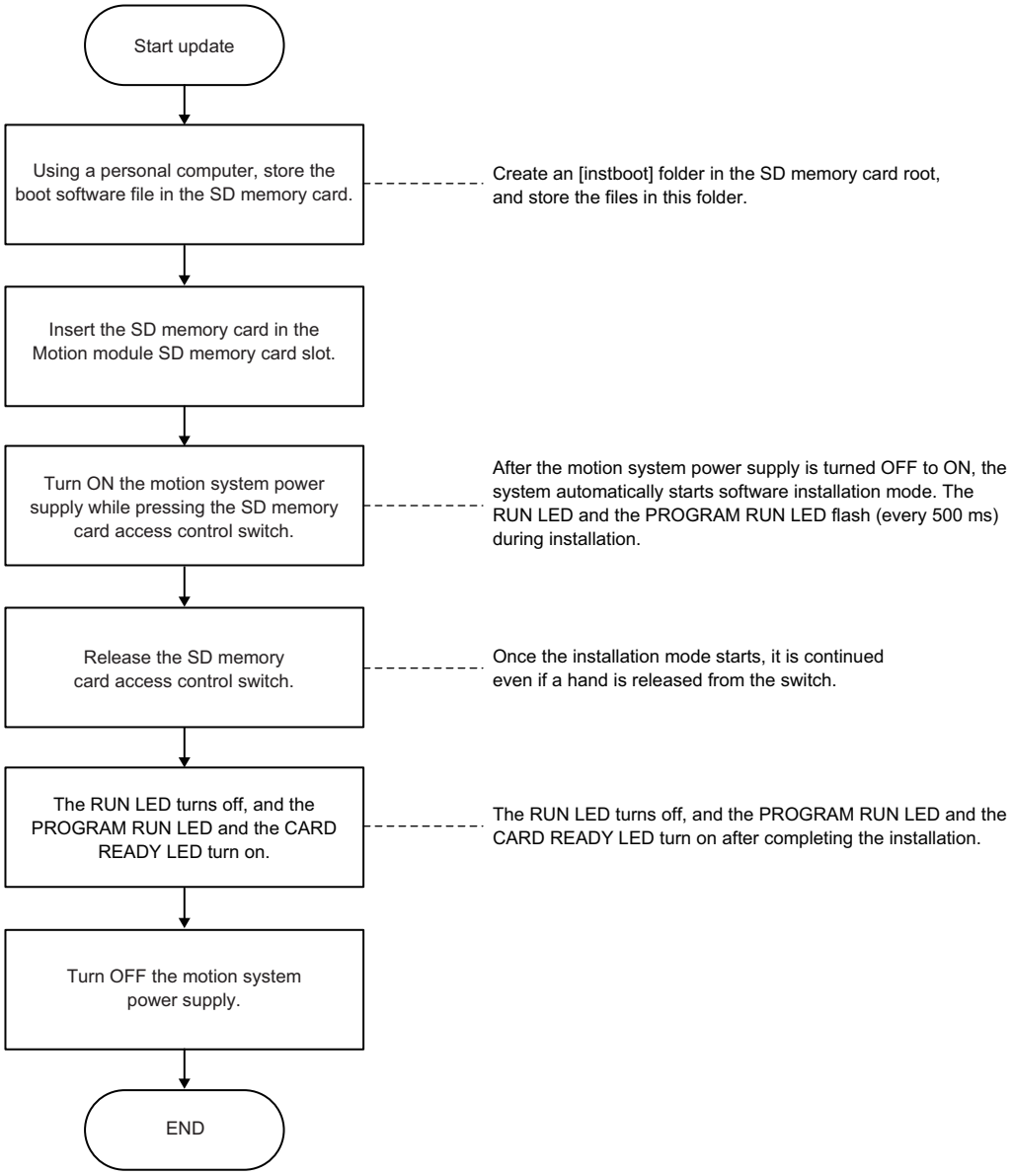
Point

- Even though update is carried out, programs, parameters, and the absolute position data written in the motion system will not be changed.
- If update processing is interrupted during updating execution by the following operation, a file which is being updated will be deleted. Execute update again.
  "Turning OFF the power of the motion system"
  "Turning RUN/STOP/RESET switch of the CPU module to RESET"
  "Removing the SD memory card (when using SD memory cards)"
- The updated software is reflected when the power is turned ON again.
- While update is executed, update operation is not accepted by the other installation method.
- Version of the boot software which is installed to the motion system can be chcked with System.Md.Version_BootSw.
- The boot software is protected by CRC. The consistency of the data is checked when turning on the power of the motion system again, then if an error is detected, the system will not start. Execute update again.

How to update the boot software

■Update procedures using the SD memory card

The following shows a procedure to update the boot software by using an SD memory card.



Point

- The boot software creates the [instboot] folder in the root of the SD memory card. Store the file in the folder.
- When the update did not complete successfully because an error was detected after update of the boot software was started, the RUN LED turns OFF, and the PROGRAM RUN LED and the CARD READY LED are flashing (every 200 ms). Update the software again after clearing the error.
  When reading the SD memory card is abnormal, check the SD memory card.
  When the boot software file is abnormal or when it cannot be recognized, check the boot software file.
- If the boot software updating can not be completed successfully, the RUN LED turns OFF or the ERR LED turns OFF or flashing (every 200 ms) even if the Motion module is powered ON and the Motion module maight not be recognized.
  In this case, insert an SD memory card which the boot software is stored into the slot of the Motion module and power ON the motion system while pressing the SD memory card access control switch. In that condition, release a hand from the SD memory card access control switch after the ERR LED turns flashing (every 200 ms) and the boot software can be recovered by turning OFF to ON the power of the Motion module after waiting for five minutes or longer.
- To install the version "07 or earlier" of the motion system software from the SD memory card, boot software update and software installation cannot be executed simultaneously. When there are both folders, only the boot software update is executed.
- The forced stop function of the SD memory card cannot be used during update execution from the SD memory card. Do not remove the SD memory card during update.
- The ready [X0] and the synchronization flag [X1] do not turn ON during installation execution from the SD memory card.

☐: OFF, ■: ON, ●: Flashing

Motion system status	LED display	Description	Remedy
Normal operation	RUN● P RUN● C RDY☐	Flashing (every 500 ms): Updating the boot software	Wait until boot software update is completed.
	RUN☐ P RUN■ C RDY■	Boot software update is normally completed	Turn the power of the motion system OFF to ON, and check that the motion system can be normally turned RUN/STOP.
Operation failure	RUN☐ P RUN● C RDY●	Flashing (every 200 ms): Boot software update is completed with an error.	Turn the power of the motion system OFF to ON, then execute boot software update again.

The boot software list

For the boot software list, refer to the following.

☞ 28 List of Boot software

22 TROUBLESHOOTING

22.1 LED Control

The status of the motion system can be checked by the following.

The following lists the LED display specifications of the Motion module.

Description differs during software installation. For details, refer to the following.

☞ 21 MODULE SOFTWARE INSTALLATION

Name	Description
RUN LED	Displays the operation status. ON: Operating normally Flashing (every 500 ms): Clear / Quick clearing OFF: Error occurrence*1 or initializing
ERR LED	Displays the error status. OFF: Operating normally ON: Error occurrence*1 Flashing (every 200 ms): Error occurrence*1 Flashing (every 500 ms): A data link faulty station is detected.
PROGRAM RUN LED	Displays the execution status of the built-in program. ON: Executing program OFF: Stopping program
CARD READY LED	Displays the status of an SD memory card. ON: SD memory card is usable Flashing: Preparing OFF: No SD memory card inserted
CARD ACCESS LED	Displays the access status of an SD memory card. ON: Accessing SD memory card OFF: Not accessing SD memory card
D LINK LED	Displays the data link status. ON: Data link (cyclic transmission being performed) Flashing: Data link (cyclic transmission stopped) OFF: Data link not performed (disconnection)
SD/RD LED	Displays the data communication status. Flashing: Communicating data OFF: Not communicating data
L ER LED	Displays the port status. ON: Abnormal data received OFF: Normal data received
LINK LED	Displays the link status. ON: Link-up OFF: Link-down

*1 Error status can be determined by status of the RUN LED and the ERR LED.

RUN LED	ERR LED	Error status	Description
OFF	ON or flickering	Major error	An error such as hardware failure or memory failure. The module stops operating.
ON	Flashing	Moderate error	An error, such as parameter error, which affects module operation. The module stops operating.
ON	ON	Minor error	An error such as communication, positioning control and program failure. The module continues operating.

When multiple errors occur, the error status is displayed in the order of major, moderate, and minor.

22.2 Checking Errors and Warnings

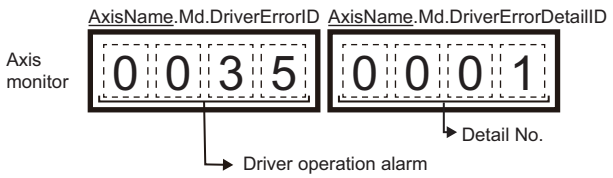
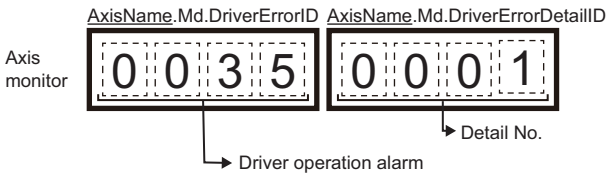
Errors and warnings occurred in the motion system can be checked by the following methods.

Method	Details
Event history of the engineering tool	It can be checked by the event history window of the engineering tool. For details, refer to the following. GX Works3 Operating Manual
Module diagnostic of the engineering tool	It can be checked by the module diagnostic window of the engineering tool. For details, refer to the following. GX Works3 Operating Manual
Motion event history of the engineering tool	It can be checked by the motion event history window of the engineering tool (Motion control setting function). For details, refer to the following. Page 563 History Data
Monitor data (Label)	The detection flag of error/warning and the error code/warning code can be checked by the label of axis, axes group, and system. The CPU module can refer to these labels by the module label or the dedicated instruction.

Error types are as follows.

Error type	Description
System error	Error not related to the axis or FB such as initial failure and bus failure
Axis/Axes group error	Axis related error which is occurred where the setting value of FB input pin is illegal or in control in the FB where the Axis/AxesGroup pin exists

Relevant variables

Variable/Structure name		Name	Details
System.Md			
	Warning	Motion system warning detection	Becomes TRUE at warning occurrence.
	Error	Motion system error detection	Becomes TRUE at error occurrence.
	NetworkError	Network error detection	Becomes TRUE at network error occurrence.
	WarningID	Latest motion system warning code	Stores the latest warning code.
	ErrorID	Latest motion system error code	Stores the latest error code.
	NetworkErrorID	Network error code	Stores the error code at network error occurrence.
AxisName.Md			
	Warning	Axis warning detection	Becomes TRUE at axis warning occurrence.
	Error	Axis error detection	Becomes TRUE at axis error occurrence.
	DriverError	Drive module error detection	Becomes TRUE at drive unit error occurrence.
	WarningID	Axis warning code	Stores the warning code at axis warning occurrence.
	ErrorID	Axis error code	Stores the error code at axis error occurrence.
	DriverErrorID	Drive module error code	Stores upper 16 bits of the slave object [Current alarm] at drive unit error occurrence. (Ex.) For MR-J5(W)-G  Stores [0035H] when servo alarm [AL.035.1_Command frequency error] occurs at drive unit.
	DriverErrorDetailID	Drive Module Error Detail Code	Stores lower 16 bits of the slave object [Current alarm] at drive unit error occurrence. (Ex.) For MR-J5(W)-G  Stores [0001H] when servo alarm [AL.035.1_Command frequency error] occurs at drive unit.
AxesGroupName.Md.			
	Warning	Axes group warning detection	Becomes TRUE at axes group warning occurrence.
	Error	Axes group error detection	Becomes TRUE at axes group error occurrence.
	WarningID	Axes group warning code	Stores the warning code at axes group warning occurrence.
	ErrorID	Axes group error code	Stores the error code at axes group error occurrence.

Control details

The latest error code occurred in motion is output in System.Md.ErrorID.

When multiple errors occur, follow the conditions below whether the error code output to System.Md.ErrorID is updated.

Occurring error	Occurred error					
System.Md.ErrorID	AxisName.Md.ErrorID/AxesGroupName.Md.ErrorID/ System.Md.ErrorID			System.Md.NetworkErrorID		
	Minor error	Moderate error	Major error	Minor error	Moderate error	Major error
No error	Updating	Updating	Updating	Updating	Updating	Updating
Minor error	Updating	Updating	Updating	Not updating	Updating	Updating
Moderate error	Not updating	Updating	Updating	Not updating	Not updating	Updating
Major error	Not updating	Not updating	Updating	Not updating	Not updating	Not updating

Filter setting

- The warning not to be detected can be set by the filter setting in the engineering tool. The error No. cannot be set. (If set, it will be ignored.)
- When the warning which has been registered in the filter setting occurs, the warning processing is executed, however, the data is not stored in the variable.

Variable/Structure name		Name	Details
System.PrConst			
	ExcludeWarning	Excluded warning	Sets the warning which is not to be detected. (Ex.) "": Not set Not detect the warnings of "0x1000, 0x1001": 0x1000, 0x1001. Not detect the warnings from "0x1000-0x1010": 0x1000 to 0x1010.

22.3 Error and Warning Reset

Errors and warnings occurred in the motion system can be reset by the following methods.

Data to be reset

■System error reset

- All errors/warnings

Error type	Description	Variable name
System	Motion system error detection	System.Md.Error
	Latest motion system error code	System.Md.ErrorID
	Motion system warning detection	System.Md.Warning
	Latest motion system warning code	System.Md.WarningID
	Network error detection	System.Md.NetworkError
	Network error code	System.Md.NetworkErrorID
Axis	Axis warning detection	AxisName.Md.Warning
	Axis error code	AxisName.Md.ErrorID
	Axis error detection	AxisName.Md.Error
	Axis warning code	AxisName.Md.WarningID
	Drive module error detection	AxisName.Md.DriverError
	Drive module error code	AxisName.Md.DriverErrorID
	Drive module error detail code	AxisName.Md.DriverErrorDetailID
Axes group	Axes group warning detection	AxesGroupName.Md.Warning
	Axes group warning code	AxesGroupName.Md.WarningID
	Axes group error detection	AxesGroupName.Md.Error
	Axes group error code	AxesGroupName.Md.ErrorID

■Axis error reset

- Axis error/warning

Error type	Description	Variable name
Axis	Axis warning detection	AxisName.Md.Warning
	Axis error code	AxisName.Md.ErrorID
	Axis error detection	AxisName.Md.Error
	Axis warning code	AxisName.Md.WarningID
	Drive module error detection	AxisName.Md.DriverError
	Drive module error code	AxisName.Md.DriveErrorID
	Drive module error detail code	AxisName.Md.DriverErrorDetailID

■Axes group error reset

- Axes group error/warning

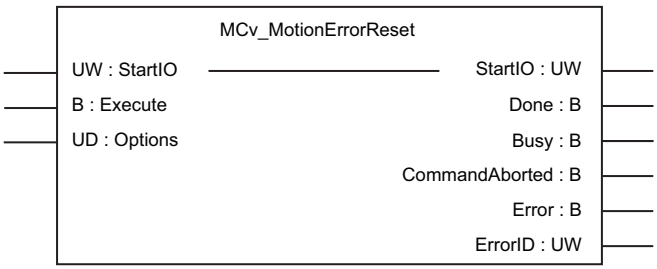
Error type	Description	Variable name
Axes group	Axes group warning detection	AxesGroupName.Md.Warning
	Axes group warning code	AxesGroupName.Md.WarningID
	Axes group error detection	AxesGroupName.Md.Error
	Axes group error code	AxesGroupName.Md.ErrorID
	Axis warning detection*1	AxisName.Md.Warning
	Axis warning code*1	AxisName.Md.WarningID
	Axis error detection*1	AxisName.Md.Error
	Axis error code*1	AxisName.Md.ErrorID
	Drive module error detection*1	AxisName.Md.DriverError
	Drive module error code*1	AxisName.Md.DriverErrorID
	Drive module error detail code*1	AxisName.Md.DriverErrorDetailID

*1 Only axis which belongs to axes group

Relevant variables/FB

Variable/Structure name	Name	Details
System.Cd.		
ErrorReset	System error reset	Resets all errors and warnings.
AxisName.Cd.		
ErrorReset	Axis error reset	Resets axis errors/warnings and drive unit errors.
AxesGroupName.Cd.		
ErrorReset	Axes group error reset	Resets axes group errors/warnings.

MCv_MotionErrorReset

Item	Description
Function overview	Resets all errors and warnings of the motion system.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

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

■I/O variable

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
I/O No.	StartIO	WORD(HEX)	↑	0 to FF0H	Refer to the right description.	It can be omitted when FB is used at motion system side. Setting values are ignored. It cannot be omitted when FB is used at the CPU module side.

■Input variables

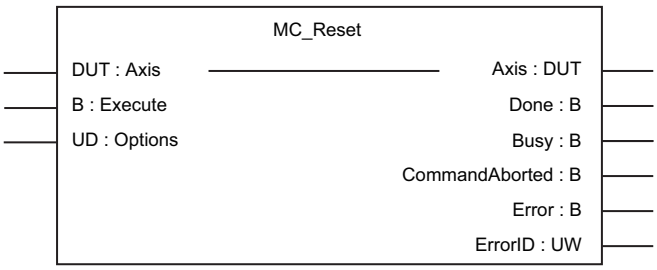
Import ↑: At start

Name	Variable name	Data type	Import	Setting range	Default value	Description
Execution command	Execute	BOOL	↑	TRUE, FALSE	FALSE	Turns TRUE, the FB is activated.
Options	Options	DWORD(HEX)	↑	00000000H	00000000H	Specify "00000000H". (If other than "00000000H" is specified, the error "Out of Options Range" (error code: 1A4EH) occurs.)

■Output variables

Name	Variable name	Data type	Default value	Description
Execution completion	Done	BOOL	FALSE	Indicates that reset has been completed.
Executing	Busy	BOOL	FALSE	Iindicates that the FB is executing.
Abortion of execution	CommandAborted	BOOL	FALSE	Indicates that the command has been aborted because of timeout. Turns FALSE because of Execute = FALSE.
Error	Error	BOOL	FALSE	Indicates that an error has occurred in the FB as it is TRUE.
Error code	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned.

MC_Reset

Item	Description
Function overview	Resets errors and warnings of the axis.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

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

■I/O variable

Input import ↑: At start

Name	Variable name	Data type	Input import	Setting range	Default value	Description
Axis information	Axis	AXIS_REF	↑	—	Omission not possible	☞ Page 702 Axis variables

■Input variables

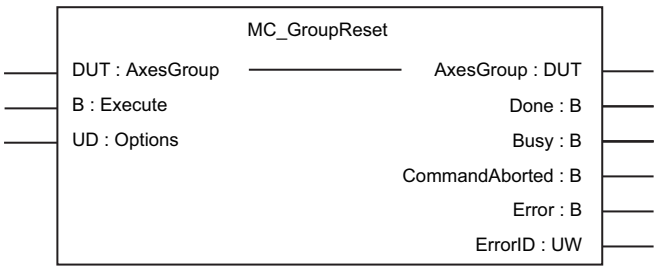
Import ↑: At start

Name	Variable name	Data type	Import	Setting range	Default value	Description
Execution command	Execute	BOOL	↑	TRUE, FALSE	FALSE	Turns TRUE, the FB is activated.
Options	Options	DWORD(HEX)	↑	00000000H	00000000H	Specify "00000000H". (If other than "00000000H" is specified, the error "Out of Options Range" (error code: 1A4EH) occurs.)

■Output variables

Name	Variable name	Data type	Default value	Description
Execution completion	Done	BOOL	FALSE	Indicates that reset has been completed.
Executing	Busy	BOOL	FALSE	Indicates that the FB is executing.
Abortion of execution	CommandAborted	BOOL	FALSE	Indicates that the command has been aborted because of timeout. Turns FALSE because of Execute = FALSE.
Error	Error	BOOL	FALSE	Indicates that an error has occurred in the FB as it is TRUE.
ErrorID	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned.

MC_GroupReset

Item	Description
Function overview	Resets errors and warnings of the axes group and each axis under the axes group.
Symbol [Structured Ladder]	
Applicable hardware and software (Language)	RD78G(H) (Structured Text)

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

■I/O variable

Input import ↑: At start

Name	Variable name	Data type	Imput import	Setting range	Default value	Description
Axes group information	AxesGroup	AXES_GROUP_REF	↑	—	Omission not possible	☞ Page 706 Axes group variables

■Input variables

Import ↑: At start

Name	Variable name	Data type	Import	Setting range	Default value	Description
Execute command	Execute	BOOL	↑	TRUE, FALSE	FALSE	Turns TRUE, the FB is activated.
Options	Options	DWORD(HEX)	↑	00000000H	00000000H	Specify "00000000H". (If other than "00000000H" is specified, the error "Out of Options Range" (error code: 1A4EH) occurs.)

■Output variables

Name	Variable name	Data type	Default value	Description
Execution completion	Done	BOOL	FALSE	Indicates that all axes reset of the axes group has been completed.
Executing	Busy	BOOL	FALSE	Iindicates that the FB is executing.
Abortion of execution	CommandAborted	BOOL	FALSE	Indicates that the command has been aborted because of timeout. Turns FALSE because of Execute = FALSE.
Error	Error	BOOL	FALSE	Indicates that an error has occurred in the FB as it is TRUE.
Error code	ErrorID	WORD(UINT)	0	The error code generated in the FB is returned.

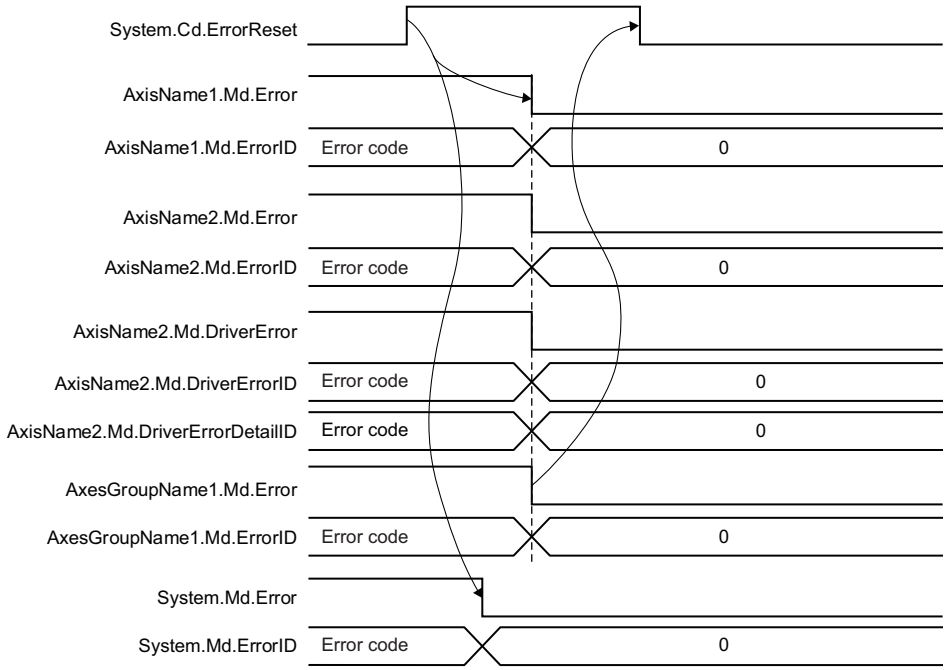
System error reset

- Reset all errors/warnings of the motion system. There are System.Cd.ErrorReset, MCv_MotionErrorReset and clear error button on the module diagnostics window of GX Works3 as a reset method. (SYSTEM.Cd.ErrorReset does not turn TRUE as the reset is performed by clear error button.)

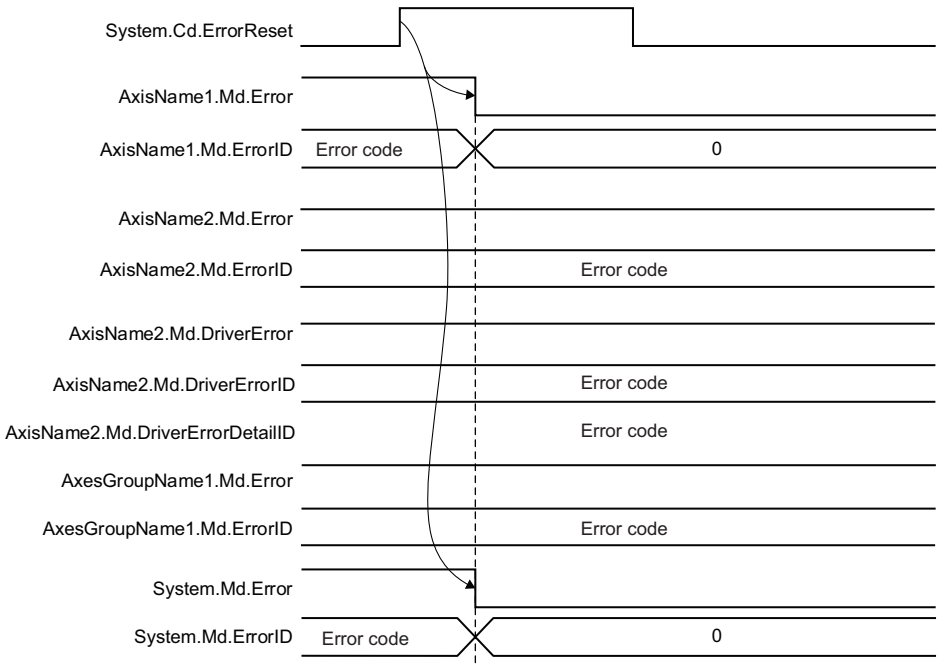
Control details

- Reset is executed at the rising edge detection of System.Cd.ErrorReset.
- When the reset is completed, System.Cd.ErrorReset becomes FALSE.
- If the error reset is not succeeded, System.Cd.ErrorReset remains TRUE.

[When the FB is normally completed] (When the error reset is succeeded)



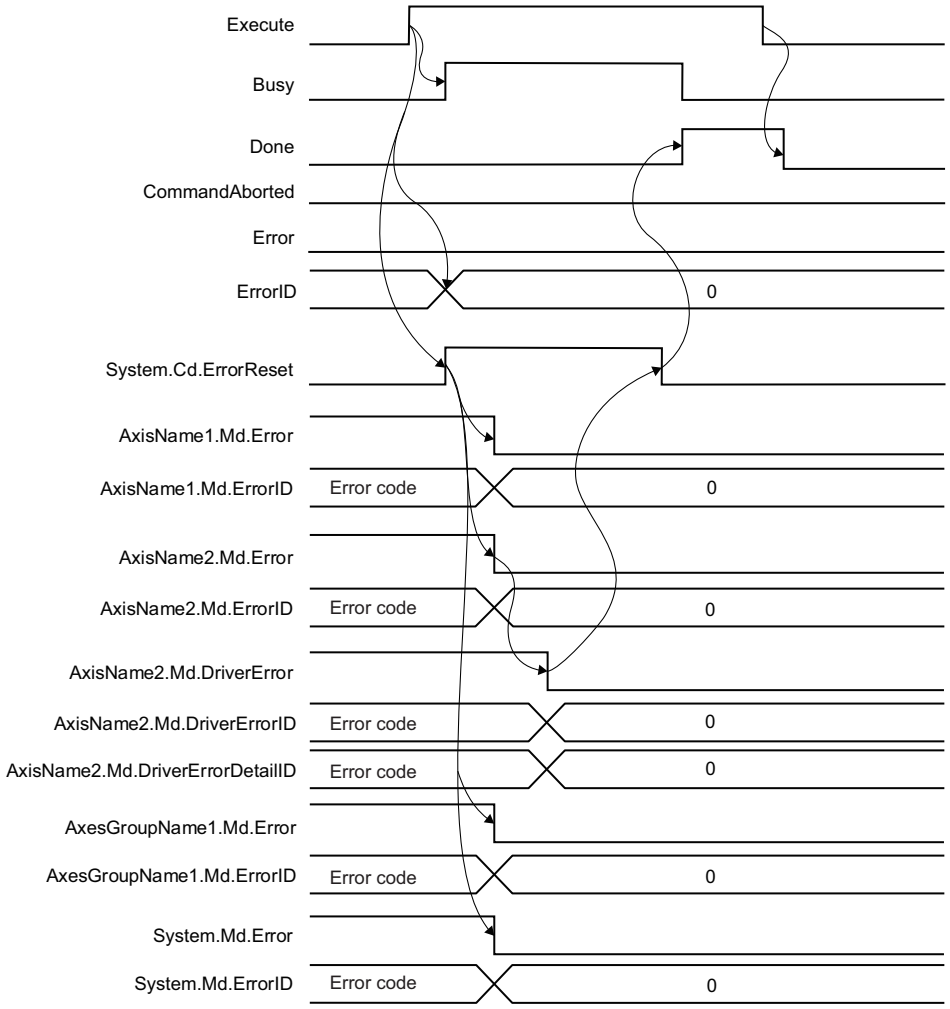
[When an error occurs] (When the error reset is failed)



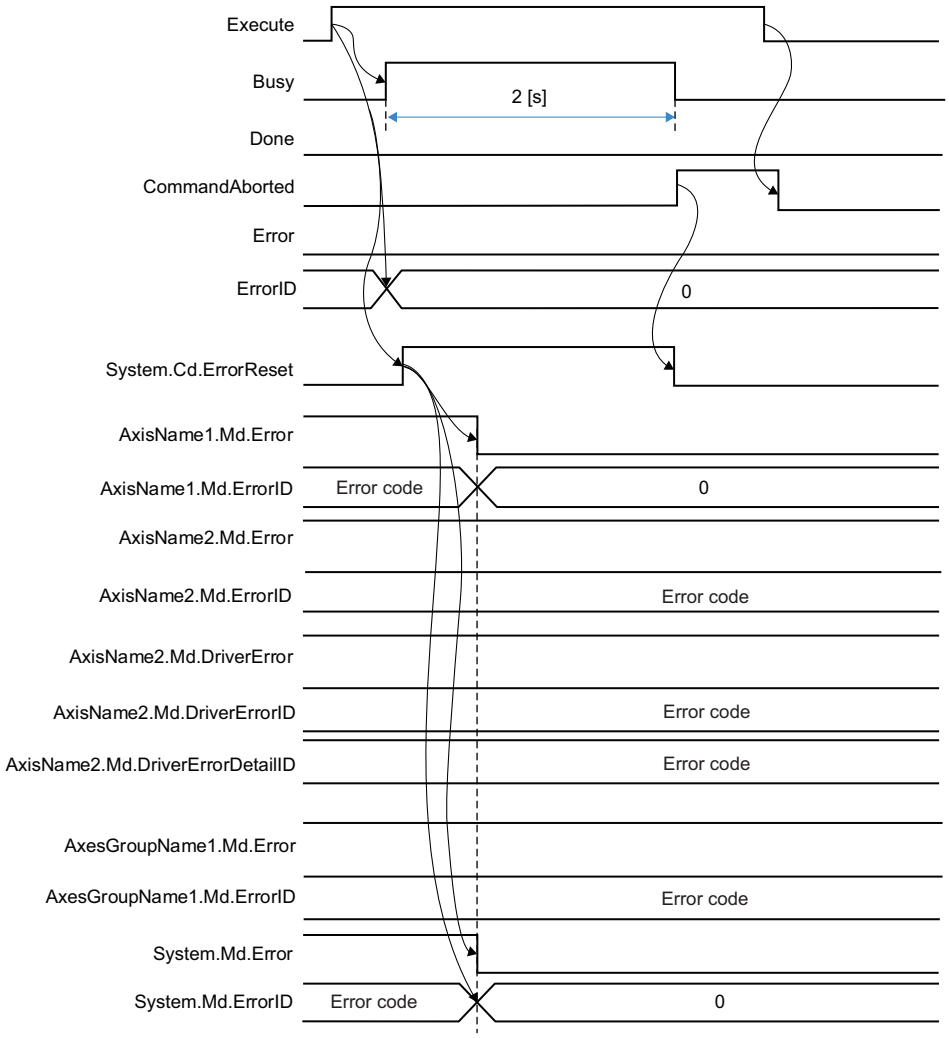
■The motion error reset

- The FB is executed with Execute = TRUE and Busy turns TRUE as the process starts, and then SYSTEM.Cd.ErrorReset turns TRUE.
- Busy turns FALSE and Done turns TRUE as clearing the errors/warnings of the motion system is completed.
- The errors/warnings are not cleared if Execute turns TRUE while the error/warning causes still remain. In this case, CommandAborted turns TRUE and SYSTEM.Cd.ErrorReset turns FALSE if the errors are not cleared within two seconds after command executed.Turn Execute to FALSE temporarily to clear the error/warning causes and turn Execute to TRUE again.

[When the FB is normally completed]



[At timeout]



Precautions

- Do not operate SYSTEM.Cd.ErrorReset directly while SYSTEM.Cd.ErrorReset is executed.
- Use MCv_MotionErrorReset without operating SYSTEM.Cd.ErrorReset to perform the error reset from the CPU module.

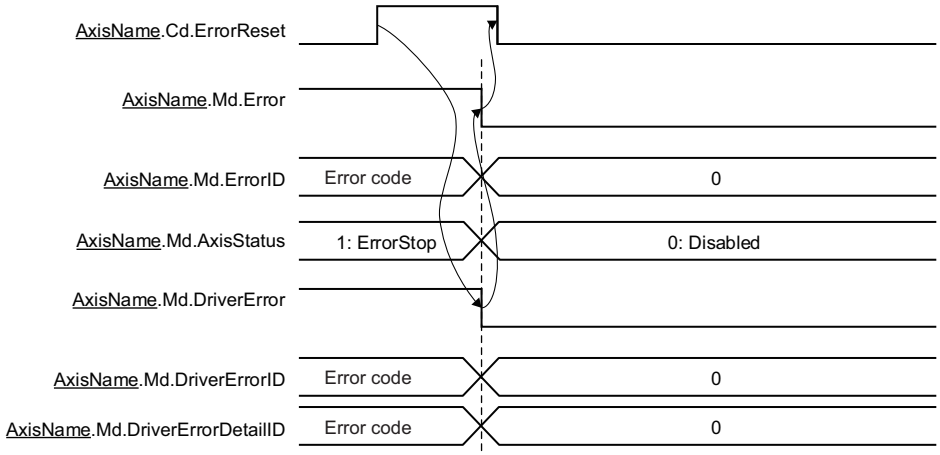
Axis error reset

Reset the error/warning of a single axis and drive unit error by AxisName.Cd.ErrorReset.

Control details

■AxisName.Cd.ErrorReset

- Reset is executed at the rising edge detection of AxisName.Cd.ErrorReset.
- When the reset is completed, AxisName.Cd.ErrorReset becomes FALSE.
- The error reset is not executed if it is executed during axis operation. The error monitor is reset when the axis is stopped completely and AxisName.Cd.ErrorReset is TRUE.

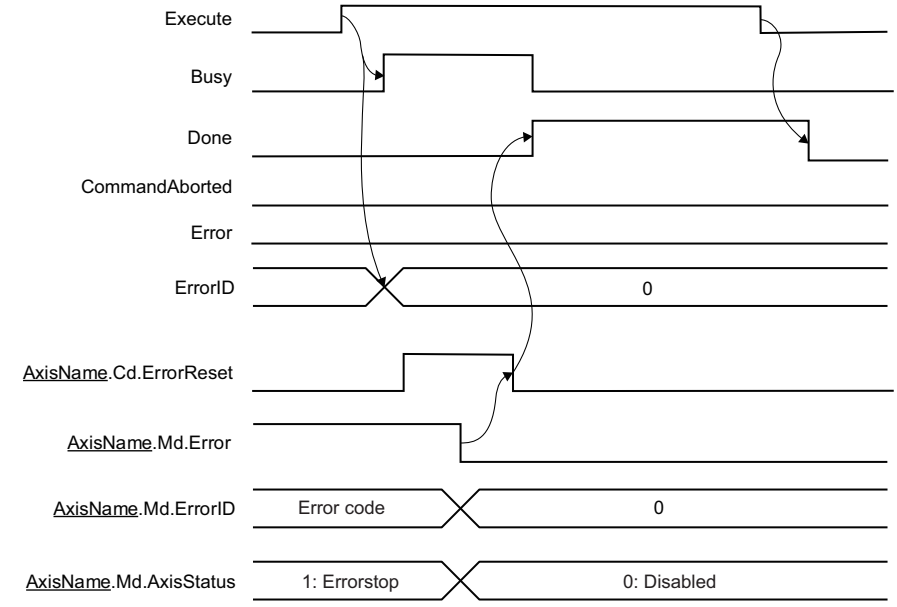


■MC_Reset

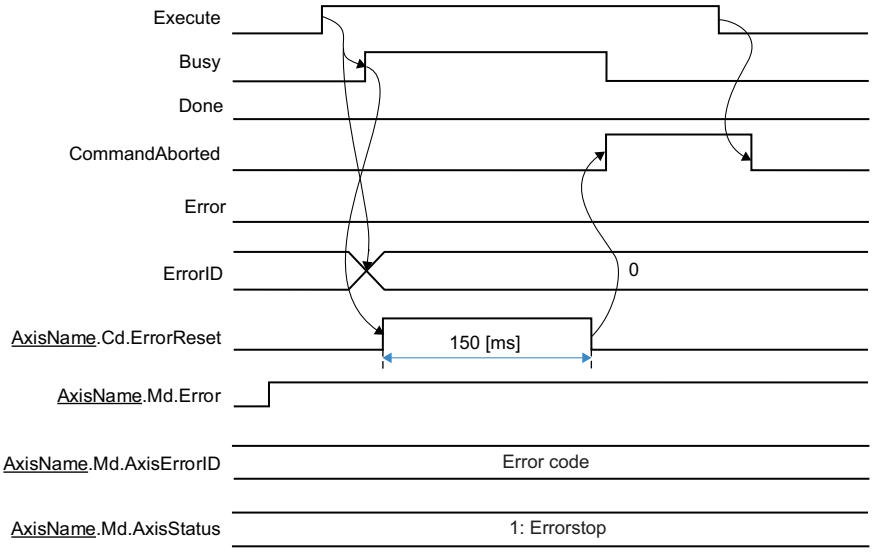
- The FB is executed with Execute = TRUE and Busy turns TRUE as the process starts, and then turn TRUE AxisName.Cd.ErrorReset of the target axis.
- Busy turns FALSE and Done turns TRUE as clearing the errors/warnings of the axis is completed.
- Busy turns FALSE and Done turns TRUE after the warnings are cleared if execution is carried out as the axis status is not ErrorStop.
- The errors/warnings are not cleared if Execute turns TRUE while the error/warning causes of the axis still remain. In this case, CommandAborted turns TRUE. Turn Execute to FALSE temporarily to clear the error/warning causes and turn Execute to TRUE again.
- Error is turned to TRUE and the error code is stored in ErrorID if the error occurs in the FB. For details on error codes, refer to the following.

☞ Page 637 TROUBLESHOOTING

[When the FB is normally completed]



[At timeout]



Precautions

- Do not operate AxisName.Cd.ErrorReset directly while MC_Reset is executed.
- Use MC_Reset without operating AxisName.Cd.ErrorReset to perform the error reset from the CPU module.

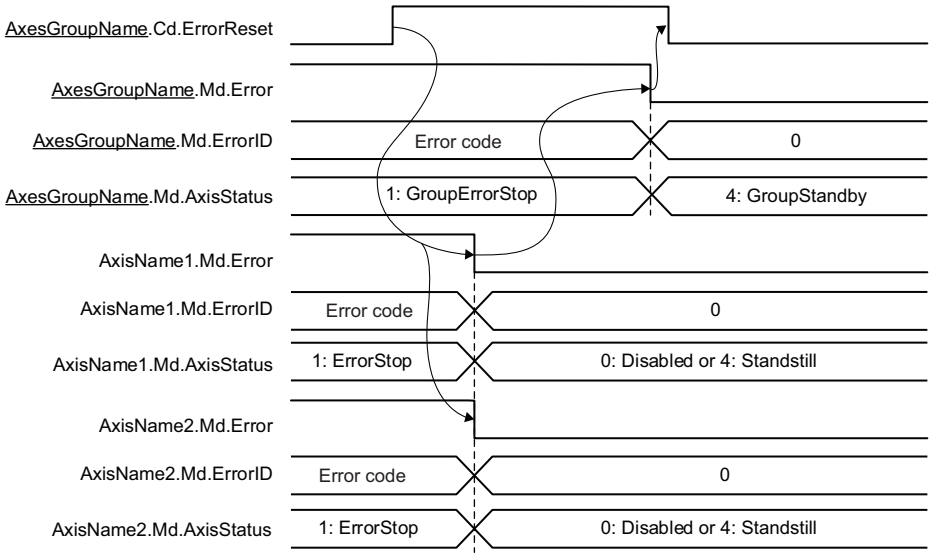
Axes group error reset

Reset the error/warning of the axes group by AxesGroupName.Cd.ErrorReset.
Reset also each axis error which belongs to the axes group in addition to the error/warning of the axes group.

Control details

■AxesGroupName.Cd.ErrorReset

- Reset is executed at the rising edge detection of AxesGroupName.Cd.ErrorReset.
- When the reset is completed, AxesGroupName.Cd.ErrorReset becomes FALSE.
- When the error reset of the configuration axis is not completed, the error reset of the axes group is not completed.
- If the error reset is not succeeded, AxesGroupName.Cd.ErrorReset remains TRUE.
- The error reset is not executed if it is executed during axes group operation. The error monitor is reset when the axes gruop is stopped completely and AxesGroupName.Cd.ErrorReset is TRUE.
- The error reset can be canceled by changing AxesGroupName.Cd.ErrorReset to FALSE.

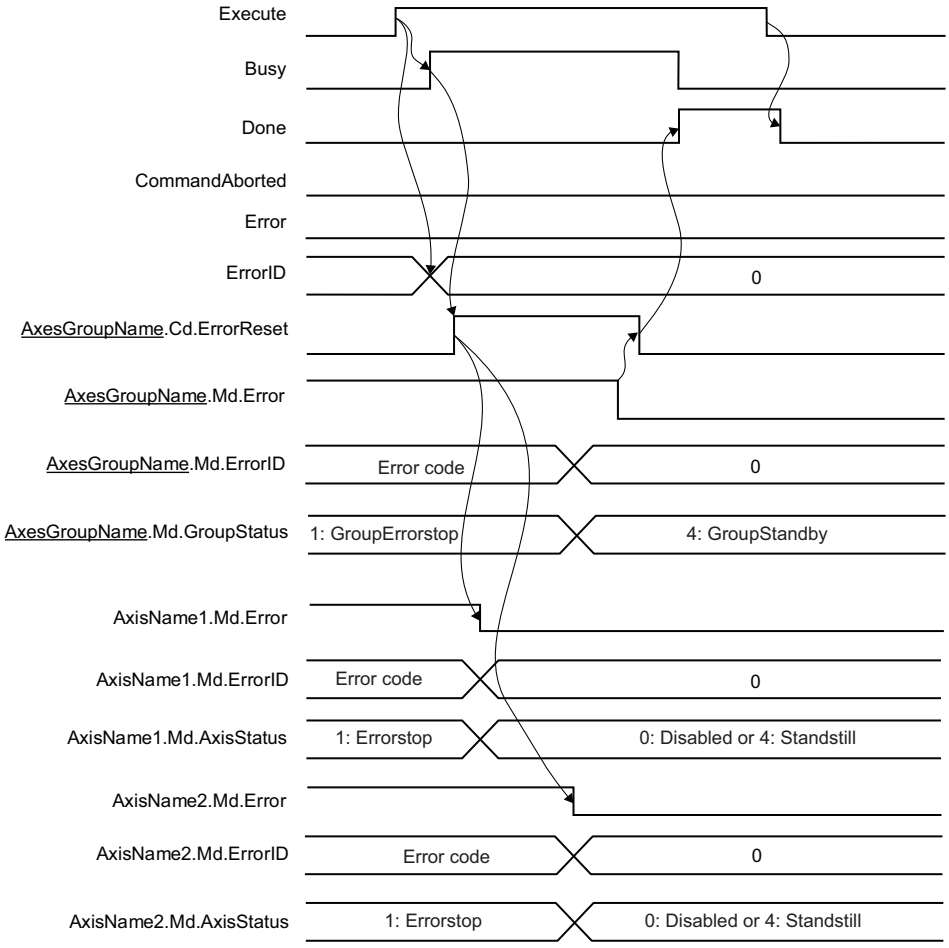


■MC_GroupReset

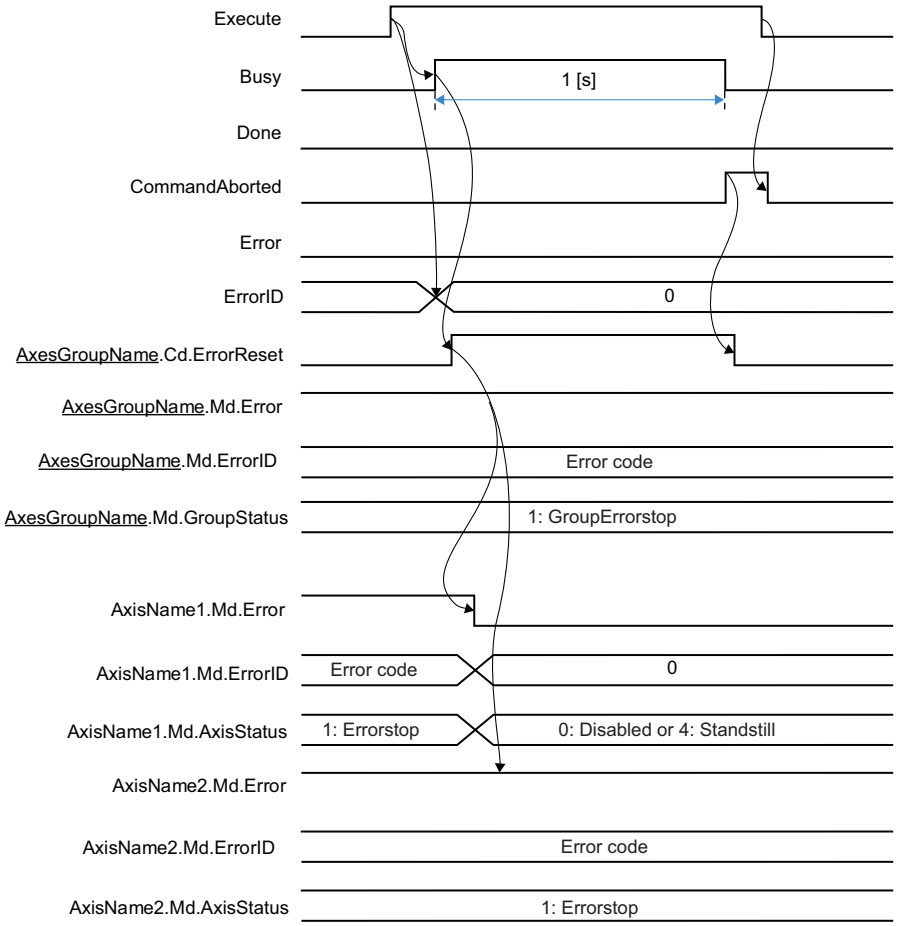
- The FB is executed at detecting a rising Execute and Busy turns TRUE as the process starts, and then turn TRUE AxesGroupName.Cd.ErrorReset of the target axes group.
- Busy turns FALSE and Done turns TRUE as clearing the errors/warnings of the axis and the axes group is completed.
- Busy turns FALSE and Done turns TRUE after the warnings are cleared if execution is carried out as the axes group status is not "1: GroupErrorStop".
- The errors/warnings are not cleared if Execute turns TRUE while the error/warning causes still remain. In this case, CommandAborted turns TRUE and turns FALSE AxesGroupName.Cd.ErrorReset of the target axes group if the errors are not cleared within a second after command executed. Turn Execute to FALSE temporarily to clear the error/warning causes and turn Execute to TRUE again.
- Error is turned to TRUE and the error code is stored in ErrorID if the error occurs in the FB. For details of error codes, refer to the following.

☞ Page 637 TROUBLESHOOTING

[When the FB is normally completed]



[At timeout]



Precautions

- Do not operate AxesGroupName.Cd.ErrorReset directly while MC_GroupReset is executed.
- Use MC_GroupReset without operating AxesGroupName.Cd.ErrorReset to perform the error reset from the CPU module.

22.4 List of Warning Codes

Warning code (Hexadecimal)	Error (System/ axis/axes group)	Warning name	Error detail and cause	Detection timing (At start/ during control	Operation at warning occurrence	Remedy
0D00H	Axis	Absolute Position Data Incorrect Warning	• Absolute position data is incorrect at current position restoration for the following causes. • Axis type was changed. • Absolute position data is erased. • The homing request has been TRUE at backup. • Driver unit conversion of the axis (numerator/denominator) was changed. • Machine homing was started, but not completed properly. [Real drive axis only] • 'Absolute position erased' at driver side was detected. • Driver object'Polarity(607EH)' b7:position polarity was changed. • Driver object 'HomeOffset(607CH)' was changed. • The connected driver device was changed. • The encoder resolution of connected driver device was changed. [Real encoder axis (via drive module) only] • 'Absolute position erased' at scale measurement encoder was detected. • The encoder resolution of scale measurement encoder was changed.	During control	The current position is restored with the same content as the incremental system, and the homing request turns TRUE.	Please execute homing.
0D01H	Axis	Start during Operation Warning	Instruction that cannot start during operation was executed.	During control	Not start after the execution of FB is interrupted.	Start when axis is Standstill.
0D02H	Axis	JOG Command Input at Both Directions Warning	JogForward and JogBackward became TRUE at the same time.	At start During control	• [At start] Not start after the execution of FB is interrupted. • [During control] Decelerates and continues stop/wait until the other JOG command becomes FALSE.	Set it so that JogForward and JogBackward do not become TRUE at the same time.

Warning code (Hexadecimal)	Error (System/axis/axes group)	Warning name	Error detail and cause	Detection timing (At start/during control	Operation at warning occurrence	Remedy
0D03H	Axis/Axes group	Speed Limit Value Over Warning	The target velocity exceeded the following when changing velocity/velocity override coefficient. • AxisName.Pr.VelocityLimit_Positive (Detail Code: 0001H) • AxisName.Pr.VelocityLimit_Negative (Detail Code: 0002H) • AxesGroupName.Pr.VelocityLimit (Detail Code: 0003H) The set velocity exceeded the following when AxisName.Pr.VelocityLimit_Over Operation was 0:Ignore. • AxisName.Pr.VelocityLimit_Positive (Detail Code: 0004H) • AxisName.Pr.VelocityLimit_Negative (Detail Code: 0005H)	At control change During control	[At control change] Continues the operation at the previous speed. [During control] Continues the operation over the speed limit.	[At control change] Set the target velocity within the following. • AxisName.Pr.VelocityLimit_Positive • AxisName.Pr.VelocityLimit_Negative • AxesGroupName.Pr.VelocityLimit [Controlling] Set the velocity within the following. • AxisName.Pr.VelocityLimit_Positive • AxisName.Pr.VelocityLimit_Negative
0D04H	Axis/Axes group	Acceleration Time Limit Over Warning	Acceleration time exceeded 8400 seconds when changing speed/acceleration (acceleration/deceleration time)/speed override coefficient/acceleration override coefficient.	During control	Continues the operation at the previous speed/acceleration (acceleration/deceleration time).	Set the speed/acceleration (acceleration/deceleration time)/speed override coefficient/acceleration override coefficient so that the acceleration time will not exceed 8400 seconds.
0D05H	Axis/Axes group	Deceleration Time Limit Over Warning	Deceleration time exceeded 8400 seconds when changing speed/deceleration (acceleration/deceleration time)/speed override coefficient/acceleration override coefficient.	During control	Continues the operation at the previous speed/deceleration (acceleration/deceleration time).	Set the speed/deceleration (acceleration/deceleration time)/speed override coefficient/acceleration override coefficient so that the deceleration time will not exceed 8400 seconds.
0D06H	Axis/Axes group	Acceleration Limit Value Over Warning	• [Acceleration/deceleration specifying method] Target acceleration exceeded AxisName(AxesGroupName).Pr.AccelerationLimit when changing acceleration/acceleration override coefficient. • [Time-fixed acceleration/deceleration method] Target acceleration exceeded AxisName(AxesGroupName).Pr.AccelerationLimit when changing velocity/acceleration/deceleration time/velocity override coefficient/acceleration override coefficient.	During control	• [Acceleration/deceleration specifying method] Continues the operation at the previous acceleration. • [Time-fixed acceleration/deceleration method] Continues the operation at the previous speed/acceleration.	Set target acceleration so that it does not exceed AxisName(AxesGroupName).Pr.AccelerationLimit.

Warning code (Hexadecimal)	Error (System/axis/axes group)	Warning name	Error detail and cause	Detection timing (At start/during control	Operation at warning occurrence	Remedy
0D07H	Axis/Axes group	Deceleration Limit Value Over Warning	• [Acceleration/deceleration specifying method] Target deceleration exceeded AxisName(AxesGroupName).Pr.DecelerationLimit when changing deceleration/acceleration override coefficient. • [Time-fixed acceleration/deceleration method] Target deceleration exceeded AxisName(AxesGroupName).Pr.DecelerationLimit when changing velocity/acceleration/deceleration time/velocity override coefficient/acceleration override coefficient.	During control	• [Acceleration/deceleration specifying method] Continues the operation at the previous deceleration. • [Time-fixed acceleration/deceleration method] Continues the operation at the previous speed/deceleration.	Set target deceleration so that it does not exceed AxisName(AxesGroupName).Pr.DecelerationLimit.
0D08H	Axis/Axes group	Position Command Unit Mismatch Warning	• Different values have been set to AxisName(AxesGroupName).Pr.Unit_Position for axes group and interpolation axis. • Different values have been set to AxisName.Pr.Unit_Position for master axis and slave axis.	At start / At multiple start	—	• Set the same value to AxisName(AxesGroupName).Pr.Unit_Position for axes group and interpolation axis. • Set the same value to AxisName.Pr.Unit_Position for master axis and slave axis. Set the same value to AxisName(AxesGroupName).Pr.Unit_Position for axes group and interpolation axis.
0D09H	Axis	Out of Forward Direction Torque Limit Value Specification Range Warning	The value is out of range for AxisName.Cd.TorqueLimit_Positive.	During control	Continues the operation with the previous torque limit value.	Set AxisName.Cd.TorqueLimit_Positive within the range.
0D0AH	Axis	Out of Reverse Direction Torque Limit Value Specification Range Warning	The value is out of range for AxisName.Cd.TorqueLimit_Negative.	During control	Continues the operation with the previous torque limit value.	Set AxisName.Cd.TorqueLimit_Negative within the range.
0D0BH	Axis/Axes group	Out of Speed Override Coefficient Range Warning	The value is out of range for AxisName(AxesGroupName).Cd.VelocityOverride.	During control	Continues the operation with the previous value.	Set AxisName(AxesGroupName).Cd.VelocityOverride within the range.
0D0CH	Axis/Axes group	Out of Acceleration Override Coefficient Range Warning	The value is out of range for AxisName(AxesGroupName).Cd.AccelerationOverride.	During control	Continues the operation with the previous value.	Set AxisName(AxesGroupName).Cd.AccelerationOverride within the range.
0D0DH	Axis/Axes group	Out of Position/Distance Setting Range Warning	The value is out of range for Position/Distance.	At control change	Continues the operation with the previous Position/Distance.	Set Position/Distance within the range.
0D0FH	Axis/Axes group	Out of Speed Range Warning	The value is out of range for Velocity.	During control	Continues the operation with the previous speed.	Set Velocity within the range.

Warning code (Hexadecimal)	Error (System/axis/axes group)	Warning name	Error detail and cause	Detection timing (At start/during control	Operation at warning occurrence	Remedy
0D10H	Axis/Axes group	Overrun Warning	Target position turned overrun because the deceleration distance for command output speed was not reached according to either of the following settings. • Target position change during control • Acceleration/deceleration, acceleration/deceleration time change during control • Switching based on multiple start	During control / When switching by multiple start	Stops immediately.	Set Velocity, Position/Distance and Deceleration so that an overrun will not occur.
0D11H	Axis/Axes group	Multiple Start Speed Mode Specified Mismatch Warning	The VelocityMode of executing FB is different from that of multi-started FB.	At multiple start	Operates by "VelocityMode" of each FB.	Set the VelocityModeof executing FB same as that of multi-started FB.
0D12H	Axis	Torque Limit Value Over Warning	• Torque was changed to a larger value than AxisName.Md.TorqueLimit_Positive. (Detail Code: 0001H) • Torque was changed to a larger value than AxisName.Md.TorqueLimit_Negative. (Detail Code: 0002H)	At control change	Continues the operation with the previous target torque.	Set Torque within the following. • AxisName.Md.TorqueLimit_Positive • AxisName.Md.TorqueLimit_Negative
0D13H	Axis	Filter Cumulated Value Over Warning	Filter cumulated value exceeded the positioning range.	During control	Filter cumulated value is clamped at the positioning range, and continues the operation.	Please reduce Master speed.
0D14H	Axis	Current Position Change Axis Status Incorrect Warning	The axis status is not Standstill for the torque input signal detection by executing MCv_SetPositionTriggered.	During control	Current position change is not executed.	Input the torque input signal when the axis status is Standstill.
0D15H	Axis	Current Position Change Execution Disabled Warning	An error occurred when changing the current position.	At start During control	Current position change is not executed.	Check the setting for MC_SetPosition and MCv_SetPositionTriggered.
0D16H	Axis	Out of Torque Setting Range Warning	The value is out of range for Torque.	At control change	Continues the operation with the previous target torque.	Set Torque within the range.
0D17H	Axis/Axes group	Out of Acceleration Range Warning	The value is out of range for Acceleration.	During control	Continues the operation with the previous acceleration.	Set Acceleration within the range.
0D18H	Axis/Axes group	Out of Deceleration Range Warning	The value is out of range for Deceleration.	During control	Continues the operation with the previous deceleration.	Set Deceleration within the range.
0D19H	Axis/Axes group	Out of Acceleration/Deceleration Time Range Warning	The value is out of range for Acceleration.	During control	Continues the operation with the previous acceleration/deceleration time.	Set Acceleration within the range.
0D1BH	Axis	Out of Limit Speed Range Warning	The value is out of range for LimitVelocity.	At control change	Continues the operation with the previous limit velocity.	Set LimitVelocity within the range.
0D1CH	Axis	Out of Torque Forward Direction Ramp Range Warning	The value is out of range for TorquePositiveRamp.	At control change	Continues the operation with the previous torque forward direction ramp.	Set TorquePositiveRamp within the range.
0D1DH	Axis	Out of Torque Reverse Direction Ramp Range Warning	The value is out of range for TorqueNegativeRamp.	At control change	Continues the operation with the previous torque reverse direction ramp.	Set TorqueNegativeRamp within the range.
0D1EH	Axes group	Speed Command Unit Mismatch Warning	Executing FB is different from the speed command unit.	At multiple start	—	Set the same value to AxisName(AxesGroupName).Pr.Unit_Velocity for axes group and interpolation axis.

Warning code (Hexadecimal)	Error (System/axis/axes group)	Warning name	Error detail and cause	Detection timing (At start/during control	Operation at warning occurrence	Remedy
0D1FH	Axis	Driver Warning	Warning occurred in the driver.	During operation	Executes the operation.	Please check the error details and corrective actions according to the content of AxisName.Md.DriverErrorID. (Please refer to the instruction manual of each driver for details of AxisName.Md.DriverErrorID.)
0D20H	Axis	Speed Limit Value Over Warning on Direction Change	The target velocity exceeded the speed limit value on that direction when accelerating again from speed 0 on direction change.	During control	Continues the operation with speed 0.	Make settings so that the target velocity does not exceed speed limit value after direction change.
0D21H	Axis	Out of Final Speed Range during Operation Warning	The value is out of range for EndVelocity when changing to the final target speed.	During control	Continues the operation with the last speed.	Set EndVelocity within the range.
0D22H	Axis/Axes group	Warning Starting over Number of Buffering FBs	The maximum number of FBs which can be buffered has been exceeded with multiple start.	At multiple start	Waits analysis of the buffering FB until on-going FB is completed.	Multiple start is executed so that the maximum number of FBs which can be buffered has not been exceeded.
0D23H	Axis	Warning Changing Torque Limit Value to Axis not yet Started	The torque limit value change request was made to the axis not yet started.	At control change	The torque limit value is not sent to the driver.	Please make the torque limit value change request to the started axis.
0D24H	Axis	Out of FilterTime Setting Range of Each Axis Signal Warning	The value is out of range for FilterTime.	At power ON At ready ON	Continues the operation with FilterTime 0.	Set FilterTime within the range.
0D29H	Axis	Out of Denominator Range at Control Change Warning	The value is out of range for RatioDenominator.	At control change	Continues the operation with the previous gear ratio.	Set RatioDenominator within the range.
0D2CH	Axis/Axes group	Out of Jerk Override Coefficient Range Warning	The value is out of range for AxisName(AxesGroupName).Cd.JerkOverride.	During control	Continues the operation with the previous value.	Set AxisName(AxesGroupName).Cd.JerkOverride within the range.
0D2DH	Axis/Axes group	Out of Position Command Unit Range Warning	The value is out of range for AxisName(AxesGroupName).Pr.Unit_Position.	At power ON At ready ON	Operates as "pulse".	Set AxisName(AxesGroupName).Pr.Unit_Position within the range.
0D2EH	Axis/Axes group	Out of Speed Command Unit Range Warning	The value is out of range for AxisName(AxesGroupName).Pr.Unit_Velocity.	At power ON At ready ON	Operates as "s".	Set AxisName(AxesGroupName).Pr.Unit_Velocity within the range.
0D2FH	Axis/Axes group	Out of Calculation Cycle Converted Speed Range Warning	The speed converted from Velocity by calculation cycle is out of the range.	During control	Continues the operation with the previous speed.	Set Velocity so that the speed converted by calculation cycle is within the range.
0D30H	Axis/Axes group	Out of Deceleration Range at Stop Warning	The value was out of range for AxisName(AxesGroupName).Pr.StopMode_Deceleration or System.Pr.StopMode_AllDeceleration when factors that cause a stop occurred.	During control	• [At the upper limit value and value outside] Decelerates and stops at the upper limit value. • [At the lower limit value and value outside] Stops immediately.	Set AxisName(AxesGroupName).Pr.StopMode_Deceleration or System.Pr.StopMode_AllDeceleration within the range.
0D31H	Axis	Control Mode Switching Disabled Warning	Control mode switching is requested during a switching disabled controlling.	During control	Continues the current control mode.	Please request for switching when control mode is switchable.

Warning code (Hexadecimal)	Error (System/axis/axes group)	Warning name	Error detail and cause	Detection timing (At start/during control	Operation at warning occurrence	Remedy
0D32H	Axis	Acceleration Time Over Warning on Direction Change	The acceleration time exceeded 8400 seconds when accelerating again from speed 0 on direction change.	During control	Continues the operation at speed 0.	Make settings so that the acceleration time after direction change does not exceed 8400 seconds.
0F00H	System	Soft Reboot Disabled Warning	Soft reboot was executed when System.PrConst.SoftReboot_Enable is Disabled.	When executing software reboot	Not execute soft reboot.	Set System.PrConst.SoftRebootEnable as Enabled.
0F01H	System	Log File Create Disabled Warning	An error has been detected when updating log file.	When executing file transfer function	The log file is not created.	• Delete unnecessary files and secure free capacity. • Format the target drive. • Delete log files.
0F02H	System	File Error	• The parameter content is broken. • Major version is unsupported.	When importing file	The file is not imported.	Please check the detailed information (parameter information) and parameter settings.
0F03H	System	Position Data History Update Disabled Warning	• An error was detected when updating position data history. • The memory capacity of add-on MotionEventHist is insufficient.	When updating positioning data history file	The position data history is not updated.	• Delete unnecessary files and secure free capacity. • Format the target drive. • Delete position data history files. • Please check the memory capacity of add-on MotionEventHist.
0F04H	System	File Transfer Command Acquisition Disabled Warning	Failed to get address of System.Cd.FileTransfer_Command.	At system start	Cannot execute the file transfer of the command label.	Start the system again.
0F05H	System	Event Filter Set Failed Warning	• Add-on MotionEventHist is disabled. • The memory capacity of add-on MotionEventHist is insufficient.	When updating filter setting	Continues the operation with the disabled filter setting.	• Enable add-on MotionEventHist. • Please check the memory capacity of add-on MotionEventHist.
0F06H	System	File Transfer Parameter Import Failed Warning	• Failed to access to FRAM area. • The compatibility does not exist in the version of file transfer parameter. • The CRC value of file transfer parameter is abnormal.	At system start	"System.PrConst.FileTransfer_LogCapacity" = 0, and continues the operation with disabled read/write (0×0000) of other acess control.	Start the system again.
0F07H	System	Event History Parameter Incorrect Warning	The value is out of range for System.PrConst.EventHistoryMotion_Path or System.PrConst.EventHistoryMotion_Capacity.	At system start	Operates by the label default value.	Set System.PrConst.EventHistoryMotion_Path and System.PrConst.EventHistoryMotion_Capacity within the range.

Warning code (Hexadecimal)	Error (System/axis/axes group)	Warning name	Error detail and cause	Detection timing (At start/during control	Operation at warning occurrence	Remedy
0F08H	System	Cycle Over Warning	Arithmetic processing or other fixed scan processing is not completed within the set cycle.	During operation	Continues the operation.	• Change the fixed scan interval setting of fixed scan execution type program to a larger value. • Change the following to larger values. • System.PrConst.SystemBaseCycle.Cycle • System.PrConst.OperationCycle[1].Cycle • System.PrConst.BuffermemoryRefreshCycle.Cycle
0F09H	System	File Transfer Execution Request Acquisition Disabled Warning	Failed to get address of System.Cd.FileTransfer_Execute.	At system start	Cannot execute the file transfer of the command label.	Start the system again.
0F0AH	System	File Transfer Execution State Update Disabled Warning	Failed to get address of System.Md.FileTransfer_State.	At system start	Cannot execute the file transfer of the command label.	Start the system again.
0F0BH	Axis	Maximum Setting Axis Number Over Warning	The axis setting which exceeded the maximum control axis number exists.	At ready ON	The axis which is greater than the maximum setting axis number is not recognized as the axis. (Md is not updated)	Set the number of setting axis within the maximum number of control axis.
0F0CH	System	Warning Filter Set Failed Warning	• Add-on MotionEventHist is disabled. • The memory capacity of add-on MotionEventHist is insufficient.	At ready ON	Continues the operation with the disabled filter setting.	• Enable add-on MotionEventHist. • Please check the memory capacity of add-on MotionEventHist.
0F0DH	System	Program Setting Import Disabled Warning	Failed to import the program setting.	At ready ON	The program setting is not imported.	Please write the project to user root path.
0F0EH	System	Module Extended Parameter Storage Target Warning	The project content of the storage target different from module extended parameter storage target setting was changed.	At system start	The written module extended parameter is not enabled.	Please check the module extended parameter storage target setting.
0F0FH	System	Out of FilterTime Setting Range of System Signal Warning	The value is out of range for FilterTime.	At ready ON	Continues the operation with the FillterTime = 0.	Set FilterTime within the range.
0F10H	System	Servo System Recorder Inconsistency Incorrect Warning	• Add-on with dependencies has not been installed. (Detail Code: 1) • Inconsistency occurred in versions with other add-on. (Detail Code: 1) • Inconsistency occurred in versions with engineering tool. (Detail Code: 2)	At power ON	The servo system recorder does not start.	• Please install the software again. • Please install the add-on with dependencies. • Please uninstall the add-on with version inconsistency. • Please execute project writing using the engineering tool that supports servo system recorder.
0F11H	System	Switch For Manufacturer Setting [ON] Detected	Switch 3 or 4 for manufacturer setting is ON.	At system start	Continues the operation with manufacturer support mode.	Set switch 3 or 4 to OFF after turning the power OFF, and then start the system again.
0F12H	System	Event History Filter Setting Error Warning	The event history filter is not set correctly.	At system start	Ignores the incorrect filter settings.	Check the filter setting and start the system again.

Warning code (Hexadecimal)	Error (System/ axis/axes group)	Warning name	Error detail and cause	Detection timing (At start/ during control	Operation at warning occurrence	Remedy
0F13H	System	Servo System Recorder Setting File Creation Disabled Warning	• There is a user-created logging setting file in save destination. (Detail Code: 1) • Memory capacity in save destination is insufficient. (Detail Code: 2) • Cannot output logging setting file because of system error. (Detail Code: 3)	At power ON	Logging setting data which is created in the servo system recorder is not created.	• Please check if there is a user-created setting file in save destination. • Please check the free capacity in save destination. • Please check if any error occurred to other add-on.
0F14H	System	Servo System Recorder Logging Execution Disabled Warning	• Cannot start servo system recorder because logging setting 10 or 9 is being executed. • Cannot start servo system recorder because an error occurred during the logging.	At power ON	The logging does not starts.	• Please stop logging setting 10 or 9 if they are being executed. • Please check the logging error code.

*1 Note that the setting range of the upper limit value/lower limit value differs depending on the setting value of "AxisName.PrConst.RingCount_Enable".

22.5 List of Error Codes

Error code (Hexadecimal)	Error (System/axis/axes group)	Error name	Error detail and cause	Detection timing (At start/during control	Operation at error occurrence	Remedy
1080H	System	Writing to Flash ROM Count Error	The number of writes to the flash ROM has exceeded 100000.	At module start	Not execute the writes to the flash ROM.	Please replace the module.
1800H	System	SDO Communication Error	• An error occurred in the SDO communication. • Issued a large quantity of SLMP requests at a time. • Issued an SLMP request for a slave emulation enabled axis.	During FB execution	The Error of FB turns ON, and the error code of SDO is output to SDOErrorID.	• Remove error factors according to the SDO error code • Reduce the number of SLMP requests issued, or correct the program to not to issue a new request until a response has been returned. • Correct the program to not to issue an SLMP request when slave emulation is enabled.
1A00H	Axis	Software Stroke Limit Over (Target Position)	Value which is out of range of software stroke limit was set in target position.	At start	Not start.	Set the target position within the range of software stroke limit.
1A01H	Axis	Software Stroke Limit Over (Start Position)	• It was started in the position which is out of range of software stroke limit. • It was started in the position out of the range of software stroke limit to the direction where recovery is disabled.	At start	Not start.	Set ONLY_INSIDE or DISABLE to AxisName.Cd.SwStrokeLimit_Override and move to the enabled range.
1A02H	Axis	Start when Software Stroke Limit is Valid	It was started except when AxisName.Pr.SwStrokeLimit_Target is invalid.	At start	Not start.	Set invalid for AxisName.Pr.SwStrokeLimit_Target.
1A03H	Axis	Software Stroke Limit Over (Forward Direction)	The value specified for AxisName.Pr.SwStrokeLimit_Target during control to forward direction would result in a move outside the software stroke limit range.	During control	Stops.	Move inside the range of software stroke limit range.
1A04H	Axis	Software Stroke Limit Over (Reverse Direction)	The value specified for AxisName.Pr.SwStrokeLimit_Target during control to reverse direction would result in a move outside the software stroke limit range.	During control	Stops.	Move inside the range of software stroke limit range.
1A05H	Axis/Axes group	Out of Target Position Range	The value is out of range for Position/Distance.	At start	Not start.	Set Position and Distance within the range.
1A06H	Axis	Out of Limit Speed Range	The value is out of range for LimitVelocity.	At start	Not start.	Set LimitVelocity within the range.
1A07H	Axis/Axes group	Speed Limit Value Over	The target velocity exceeded the following. • AxisName.Pr.VelocityLimit_Positive (Detail Code: 0001H) • AxisName.Pr.VelocityLimit_Negative (Detail Code: 0002H)	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	Set target velocity so that it does not exceed the following. • AxisName.Pr.VelocityLimit_Positive • AxisName.Pr.VelocityLimit_Negative • AxesGroupName.Pr.VelocityLimit

Error code (Hexadecimal)	Error (System/ axis/axes group)	Error name	Error detail and cause	Detection timing (At start/ during control	Operation at error occurrence	Remedy
1A0CH	Axis/Axes group	Acceleration/ Deceleration 0 Specified Operation Error at Start	Started after setting AxisName(AxesGroupName).Pr. AccelerationZeroBehavior to -1: ACCError with Acceleration or Deceleration 0.	At start	Not start.	Make settings so that Acceleration and Deceleration do not become 0.
1A0DH	Axis/Axes group	Out of Acceleration/ Deceleration Time Range	Acceleration was out of range when using the time-fixed acceleration/deceleration method.	At start	Not start.	Set Acceleration within the range.
1A0EH	Axis/Axes group	Acceleration/ Deceleration Method Mismatched	The acceleration/deceleration method of executing FB is different from that of multiple start FB.	At multiple start	Stops the operation.	Set the acceleration/ deceleration method of executing FB same as that of multiple start FB.
1A0FH	Axis	Acceleration Time Over at Start	Acceleration time exceeded 8400 seconds.	At start	Not start.	Set target speed/target acceleration/target deceleration/target jerk so that acceleration time does not exceed 8400 seconds.
1A10H	Axis	Deceleration Time Over at Start	Deceleration time exceeded 8400 seconds.	At start	Not start.	Set target speed/target acceleration/target deceleration/target jerk so that deceleration time does not exceed 8400 seconds.
1A11H	Axis	Out of Acceleration Range	The value is out of range for Acceleration.	At start	Not start.	Set Acceleration within the range.
1A12H	Axis	Out of Deceleration Range	The value is out of range for Deceleration.	At start	Not start.	Set Deceleration within the range.
1A13H	Axis/Axes group	Out of Jerk Range	The value is out of range for Jerk.	At start	Not start.	Set Jerk within the range.
1A17H	Axis	Set Velocity of Converted Driver Velocity Unit out of Range	The set velocity of converted driver velocity unit was out of the range from -2147483648 to 2147483647.	At start	Not start.	Set the velocity of converted driver velocity unit within the setting range.
1A18H	Axis	Out of Speed Initial Value Selection Range	The value is out of range for speed initial value selection of Options.	At start	Not start.	Set speed initial value selection of Options within the range.
1A19H	Axis	Out of Driver Unit Conversion Numerator/ Denominator Range	The value is out of range for driver unit conversion numerator/ denominator.	At start	Not start.	Set driver unit conversion numerator/ denominator within the setting range.
1A1BH	Axis	csv Unsupported Driver	The driver does not support csv.	At start	Not start.	Execute the instruction in the driver supporting csv.
1A1CH	Axis	cst Unsupported Driver	The driver does not support cst.	At start	Not start.	Execute the instruction in the driver supporting cst.
1A1DH	Axis	Control Mode Switching Error	Driver control mode switching did not complete within 1 second after control mode switching command had been issued. During communication initialization, driver control mode did not switch to csp.	During control During initial	Stops. Unable to connect to the driver.	Please check whether error occurred or the setting is incorrect, and take necessary actions.
1A1EH	Axis	Out of Target Torque Range	The value is out of range for Torque.	At start	Not start.	Set Torque within the range.
1A1FH	Axis	Out of Torque Positive Direction Ramp Range	The value is out of range for TorquePositiveRamp.	At start	Not start.	Set TorquePositiveRamp within the range.

Error code (Hexadecimal)	Error (System/axis/axes group)	Error name	Error detail and cause	Detection timing (At start/during control	Operation at error occurrence	Remedy
1A20H	Axis	Out of Torque Negative Direction Ramp Range	The value is out of range for TorqueNegativeRamp.	At start	Not start.	Set TorqueNegativeRamp within the range.
1A22H	Axis	Start during Homing Incomplete	The positioning was started when homing request flag is ON.	At start	Not start.	Start the operation after executing homing.
1A26H	Axis	Axis Status Incorrect Start	The axis is started in the status except when AxisName.Md.AxisStatus is Standstill.	At start	Not start.	Start the axis when AxisName.Md.AxisStatus is Standstill.
1A27H	Axes group	Axes Group Status Incorrect Start	The axes group was started when AxesGroupName.Md.AxesGroupStatus was other than GroupStandby.	At start	Not start.	Start the axes group operation when AxesGroupName.Md.AxesGroupStatus is GroupStandby.
1A28H	Axis	Out of Phase Compensation Advance Time Range	The value is out of range for Time.	At start	Not start.	Set a value within the range for Time.
1A29H	Axis	Out of Phase Compensation Time Constant Range	The value is out of range for TimeConstant.	At start	Not start.	Set a value within the range for TimeConstant.
1A2DH	Axis	FLS Signal Detection (at Start)	The signal input from FLS signal at start was detected.	At start	Not start.	Set ONLY_INSIDE or DISABLE to AxisName.Cd.HwStrokeLimit_Override and move to the controllable range.
1A2EH	Axis	RLS Signal Detection (at Start)	The signal input from RLS signal at start was detected.	At start	Not start.	Set ONLY_INSIDE or DISABLE to AxisName.Cd.HwStrokeLimit_Override and move to the controllable range.
1A2FH	Axis	FLS Signal Detection (Controlling)	The signal input from FLS signal during controlling was detected.	During control	Stops.	Set ONLY_INSIDE or DISABLE to AxisName.Cd.HwStrokeLimit_Override and move to the controllable range.
1A30H	Axis	RLS Signal Detection (Controlling)	The signal input from RLS signal during controlling was detected.	During control	Stops.	Set "ONLY_INSIDE" or "DISABLE" to "AxisName.Cd.HwStrokeLimit_Override" and move to the controllable range.
1A34H	Axis/Axes group	Out of Gear Denominator Range	The value is out of range for RatioDenominator.	At start	Not start.	Set a value within the range for RatioDenominator.
1A35H	Axis	Out of Time Constant Range	The value is out of range for TimeConstant.	At start	Not start.	Set a value within the range for TimeConstant.
1A36H	Axis/Axes group	Speed Limit Value Over during Controlling	The set velocity exceeded the following when AxisName.Pr.VelocityLimit_OverOperation is 3:ImmediateStop. • AxisName.Pr.VelocityLimit_Positive (Detail Code: 0001H) • AxisName.Pr.VelocityLimit_Negative (Detail Code: 0002H)	During control	Operates exeeding "AxisName.Pr.VelocityLimit_Positive" or "AxisName.Pr.VelocityLimit_Negative".	Set the velocity within the following. • AxisName.Pr.VelocityLimit_Positive • AxisName.Pr.VelocityLimit_Negative
1A37H	Axis	Out of Direction Selection Range	The value is out of range for Direction.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	Set Direction within the range.

Error code (Hexadecimal)	Error (System/axis/axes group)	Error name	Error detail and cause	Detection timing (At start/during control	Operation at error occurrence	Remedy
1A38H	Axis	Acceleration Limit Value Over	Target acceleration exceeded AxisName(AxesGroupName).Pr.AccelerationLimit.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	Set target acceleration so that it does not exceed AxisName(AxesGroupName).Pr.AccelerationLimit.
1A39H	Axis	Deceleration Limit Value Over	Target deceleration value exceeded AxisName(AxesGroupName).Pr.DecelerationLimit.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	Set target deceleration so that it does not exceed AxisName(AxesGroupName).Pr.DecelerationLimit.
1A3AH	Axis	Jerk Limit Value Over	Target jerk exceeded AxisName(AxesGroupName).Pr.JerkLimit.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	Set target jerk so that it does not exceed AxisName(AxesGroupName).Pr.JerkLimit.
1A3BH	Axis	Deceleration Stop by Servo OFF During Operation	[0: Servo OFF after deceleration stop is completed] was selected for AxisName.Pr.StopMode_ServoOff, and FALSE was set to ServoOn in MC_Power(MCv_AllPower) during operation.	During control	Deceleration stops.	Set TRUE to ServoOn in MC_Power(MCv_AllPower).
1A3CH	Axis	Immediate Stop by Servo OFF During Operation	• [1: Servo OFF after immediate stop] was selected for AxisName.Pr.StopMode_ServoOff, and FALSE was set to ServoOn in MC_Power(MCv_AllPower) during operation. • FALSE was set to Enable in MC_Power(MCv_AllPower) during operation.	During control	Immediately stops.	Set TRUE to Enable and ServoOn in MC_Power(MCv_AllPower).
1A3DH	Axes group	Stop Cause of Axes Group Configuration Axis	Error occurred in configuration axis.	At start During control	• [At start] Not start. • [During control] Stops.	Please check the error details of configuration axis.
1A3EH	Axis	Master Axis and Slave Axis No. Duplicated	The same axis No. has been set to master axis and slave axis.	At start When executing FB	Not execute FB.	Set different axis No. to master axis and slave axis.
1A3FH	Axis	Master Axis and Slave Axis Cyclic Reference	The master axis is set to the slave axis in the latter part of the FB. The number of connected FBs for synchronization exceeds the upper limit.	At start When executing FB	Not execute FB.	Do not set master axis to the slave axis in the latter part of the FB. Reduce the number of connected FBs for synchronization less than the upper limit.
1A40H	Axis	Out of Cam Table ID Range	The value is out of range for CamTableID.	At start When executing FB	Not execute FB.	Specify the value within the range for CamTableID.
1A41H	Axis	Out of Master Axis Offset Range	The value is out of range for MasterOffset.	At start When executing FB	Not execute FB.	Set a value within the range for MasterOffset.
1A42H	Axis	Out of Slave Axis Range	The value is out of range for SlaveOffset.	At start When executing FB	Not execute FB.	Set a value within the range for SlaveOffset.
1A43H	Axis	Out of Master Axis Coefficient Range	The value is out of range for MasterScaling.	At start When executing FB	Not execute FB.	Set a value within the range for MasterScaling.
1A44H	Axis	Out of Slave Axis Coefficient Range	The value is out of range for SlaveScaling.	At start When executing FB	Not execute FB.	Set a value within the range for SlaveScaling.

Error code (Hexadecimal)	Error (System/axis/axes group)	Error name	Error detail and cause	Detection timing (At start/during control	Operation at error occurrence	Remedy
1A45H	Axis	Out of Master Axis Follow-up Distance Range	The value is out of range for MasterStartDistance.	At start When executing FB	Not execute FB.	Set a value within the range for MasterStartDistance.
1A46H	Axis	Out of Master Axis Synchronization Start Position Range	The value is out of range for MasterSyncPosition.	At start When executing FB	Not execute FB.	Set a value within the range for MasterSyncPosition.
1A47H	Axis	Out of Start Mode Range	The value is out of range for StartMode.	At start When executing FB	Not execute FB.	Set a value within the range for StartMode.
1A48H	Axis/Axes group	Out of Master Axis Data Source Range	The value is out of range for MasterValueSource.	At start When executing FB	Not execute FB.	Set a value within the range for MasterValueSource.
1A49H	Axis/Axes group	Same Master Axis Data Source Selection	An axis selected as Master in a configuration that connects multiple axes with single axis synchronization FB is used as Slave axis of latter FB.	At start When executing FB	Not execute FB.	Check axes specified by Master.
1A4AH	Axis	Out of BufferMode Range	The value is out of range for BufferMode.	At start	Not start.	Set BufferMode within the setting range.
1A4CH	Axis	No Cam Table	The cam table specified in CamTableID does not exist.	At start When executing FB	Not execute FB.	Open cam data to the open area with MC_CamTableSelect or MCv_ProfileSelect.
1A4DH	Axis/Axes group	Out of Speed Range	The value is out of range for Velocity.	At start	Not start.	Set Velocity within the range.
1A4EH	Axis	Out of Options Range	The value is out of range for Options.	At start	Not start.	Set Options within the setting range.
1A4FH	Axis	Same Master Axis 1 Slave Axis Error	The same axis No. was specified to Master1 and Slave.	At start	Not start.	Set different axis No. to Master1 and Slave.
1A50H	Axis	Same Master Axis 2 Slave Axis Error	The same axis No. was specified to Master2 and Slave.	At start	Not start.	Set different axis No. to Master2 and Slave.
1A51H	Axis	Out of Master Axis 1 Gear Denominator Range	The value is out of range for GearRatioDenominatorM1.	At start	Not start.	Set a value within the range for GearRatioDenominatorM1.
1A52H	Axis	Out of Master Axis 2 Gear Denominator Range	The value is out of range for GearRatioDenominatorM2.	At start	Not start.	Set a value within the range for GearRatioDenominatorM2.
1A57H	Axes group	Configuration Axis No. Duplication Error	The duplicated axis number of configuration axis was set.	At power ON / During Configuration Axis Write	• [At power ON] Generating axes group fails. • [During Configuration Axis Write] Continues the configuration axis which is before write.	Set configuration axis so that axis No. does not duplicate.
1A58H	Axes group	Axes Group No. Setting Duplication Error	The axes group No. was duplicated.	At power ON / During Configuration Axis Write	• [At power ON] Generating axes group fails. • [During Configuration Axis Write] Continues the configuration axis which is before write.	Change the axes group No. of axes group where the error was detected.
1A59H	Axes group	Axes Group Configuration Axis Calculation Cycle Incorrect	The calculation cycle of configuration axis in axes group is different.	At power ON / During Configuration Axis Write	• [At power ON] Generating axes group fails. • [During Configuration Axis Write] Continues the configuration axis which is before write.	Set the same calculation cycle in all configuration axes.

Error code (Hexadecimal)	Error (System/axis/axes group)	Error name	Error detail and cause	Detection timing (At start/during control	Operation at error occurrence	Remedy
1A5AH	Axes group	Axes Group Status Incorrect (During Configuration Axis Write)	MCv_WriteAxesInGroup was executed when AxesGroupName.Md.GroupStatus was other than GroupDisabled.	During Configuration Axis Write	Continues the configuration axis before writing. In operation, the operation is stopped.	Execute MCv_WriteAxesInGroup when AxesGroupName.Md.GroupStatus is GroupDisabled.
1A5BH	Axis	Forced Stop Status	Forced stop signal was detected during operating.	When inputting forced stop signal	Stops immediately.	Please cancel the forced stop signal.
1A5DH	System	Axis Data Memory Size Over	The memory size of axis data exceeds the memory capacity set in system.	At power ON	Generating axis fails.	• Please increase the memory capacity set in system. • Please delete unnecessary axes and reduce memory size of axis data.
1A5EH	System	Mapping Size Incorrect Error	PDO size exceeds the settable size.	When connecting driver	The axis set incorrectly is not communicated.	Please check the mapping object.
1A5FH	System	Mapping Communication Error	• An object that cannot be mapped was mapped. • An object that does not exist in the slave object setting was mapped. • Motion mode is set to station-specific mode when AxisName.PrConst.FastOperationMode is 5FE2H (high-speed mode). • The mapping for high-speed mode is changed when AxisName.PrConst.FastOperationMode is 5FE2H (high-speed mode).	When connecting driver	The axis set incorrectly is not communicated.	• Please check the mapping object. • Please check the setting of station-specific mode.
1A60H	System	Maximum Number of Registrations for Axes Group Over	The memory size of axes group data exceeds the memory capacity set in system.	At power ON At ready ON	Generating axes group fails.	Increase the memory capacity set in system. Or, delete unnecessary axes and keep the memory size of axis data within the memory capacity.
1A61H	Axes group	Out of Speed Mode Range	The value is out of range for speed mode.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops.	Correct the speed mode so that it is within the range.
1A62H	Axes group	Circular Interpolation Axis Unset	• A configuration axis not set up was set in CircAxes. • Same index No. was set in CircAxes in duplicate.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	• Set an axis that has been set up as configuration axis in CircAxes. • Set index No. of configuration axes in CircAxes avoiding duplication.
1A63H	Axes group	Out of Circular Interpolation Mode Range	The value is out of range for CircMode.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	Set CircMode within the range.
1A64H	Axes group	Out of Boundary Point Address Range	The value is out of range for boundary point (AuxPoint) at circular interpolation.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	Set boundary point (AuxPoit) within the range.
1A65H	Axes group	Start Point - Center Point Address Same Value	Start point is equal to center point at circular interpolation of center point specification.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	Set center point (AuxPoint) so that start point is not equal to center point.

Error code (Hexadecimal)	Error (System/axis/axes group)	Error name	Error detail and cause	Detection timing (At start/during control	Operation at error occurrence	Remedy
1A66H	Axes group	Start Point - End Point Address Same Value	Start point is equal to end point at circular interpolation.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	Set end point (EndPoint) so that start point is not equal to end point.
1A67H	Axes group	End Point - Center Point Address Same Value	End point is equal to center point at circular interpolation of center point specification.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	Set center point (AuxPoint) or end point (EndPoint) so that end point is not equal to center point.
1A68H	Axes group	Start Point - Boundary Point Address Same Value	Start point is equal to boundary point at circular interpolation of boundary point specification.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	Set boundary point (AuxPoint) so that start point is not equal to boundary point.
1A69H	Axes group	End Point - Boundary Point Address Same Value	End point is equal to boundary point at circular interpolation of boundary point specification.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	Set boundary point (AuxPoint) so that end point is not equal to boundary point.
1A6AH	Axes group	Start Point - Boundary Point - End Point Address Same Straight Line	Start point, end point and boundary point are on the same straight line at circular interpolation of boundary point specification.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	Set boundary point (AuxPoint) so that it is not on the same straight line with start point and end point.
1A6BH	Axes group	Out of Center Point Address Range	The value is out of range for center point (AuxPoint) at circular interpolation.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	Set center point (AuxPoint) within the range.
1A6CH	Axes group	Out of Radius Range	The radius is out of range at circular interpolation.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	Set end point (EndPoint) or auxiliary point (AuxPoint) so that radius is within the range.
1A6DH	Axes group	Out of End Point Address Range	The value is out of range for end point (EndPoint) at circular interpolation.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	Set end point (EndPoint) within the range.
1A6EH	Axes group	Radius Setting Error	End point (EndPoint) and radius (AuxPoint) which were not able to form an arc were set at circular interpolation of radius specification.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	Set them so that the distance between start point - end point (EndPoint) is less than or equal to the radius (AuxPoint) × 2.
1A6FH	Axes group	Out of Path Selection Range	The value is out of range for PathChoice.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	Set PathChoice within the range.
1A70H	Axes group	Out of Circular Interpolation Error Allowable Value Range	The value is out of range for CircularErrorTolerance.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	Set CircularErrorTolerance within the range.
1A71H	Axes group	Circular Interpolation Error Allowable Value Over	The differrence between radius of start point - center point and radius of end point - center point exceeds CircularErrorTolerance at circular interpolation of center point specification.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	Set end point (EndPoint), center point (AuxPoint), and CircularErrorTolerance so that CircularErrorTolerance is not exceeded .
1A72H	Axes group	Software Stroke Limit Invalid at Circular Interpolation	Circular interpolation including an axis where software stroke limit is set to invalid was executed.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	Enable software stroke limit for the axis to execute circular interpolation.
1A73H	Axes group	Out of Pitch Number Range	The value is out of range for Pitch.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	Set Pitch within the range.

Error code (Hexadecimal)	Error (System/axis/axes group)	Error name	Error detail and cause	Detection timing (At start/during control	Operation at error occurrence	Remedy
1A78H	Axis	Out of Signal Selection Range	• The value is out of range for Direction. • The value is out of range for CompensationTime. • The value is out of range for FilterTime.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	• Set Direction within the range. • Set CompensationTime within the range. • Set FilterTime within the range.
1A79H	Axis	Out of Forward Direction Torque Limit Initial Value Specification Range	A value larger than AxisName.PrConst.TorqueLimit_Max was set to AxisName.PrConst.TorqueLimit_PositiveInitial.	At power ON	Generating axis fails.	Set AxisName.TorqueLimit_PositiveInitial within the range of 0 to AxisName.PrConst.TorqueLimit_Max.
1A7AH	Axis	Out of Reverse Direction Torque Limit Initial Value Specification Range	A value larger than AxisName.PrConst.TorqueLimit_Max was set to AxisName.PrConst.TorqueLimit_NegativeInitial.	At power ON	Generating axis fails.	Set AxisName.PrConst.TorqueLimit_NegativeInitial within the range of 0 to AxisName.PrConst.TorqueLimit_Max.
1A7BH	Axes group	Insufficient Movement Amount at Changing Target Address	Movement amount became 0 after changing target position/movement distance at multiple axes interpolation control of reference axis/long axis speed designation.	During control	Stops the operation.	Set target position/movement distance so that movement amount is not insufficient.
1A7CH	Axes group	Operation FB Issue Error to the Axis during Axes Group Operating	Multiple start of single axis control FB was executed for operating axis in axes group.	At start	Stops the operation.	Execute multiple start of multiple axes control FB for the axis of axes group operating.
1A7EH	Axis/Axes group	Overrun Error	• Overrun occurred since the deceleration distance for the output speed was not satisfied after the last positioning address detected. • Positioning direction was inverted when executing multiple start by Aborting. • Positioning direction was inverted when changing target position.	Controlling At multiple start At control change	Stops the operation.	Set Velocity, Position/Distance and Deceleration so that an overrun will not occur.
1A7FH	Axes group	Driver Emulation Switch Error	Driver emulation switch was executed when AxisName.Md.AxisStatus is other than Disable or Standstill.	When executing driver emulate switching	Stops.	Execute driver emulation switch when AxisName.Md.AxisStatus is Disable or Standstill.
1A80H	Axis/Axes group	BufferMode Specification Disabled	Multiple start was executed in combination disable BufferMode.	At multiple start	Stops.	Please specify combinatorial BufferMode.
1A85H	Axis	Out of Frequency Specification Range	The value is out of range for Frequency.	At start	Not start.	Set Frequency within the range.
1A86H	Axis	Out of Filter Operation Specification Range	The value is out of range for Filter.	At start	Not start.	Set Filter within the range.
1A87H	Axis	Out of Direction Limit Value Specification Range	The value is out of range for PositiveLimit and NegativeLimit.	At start	Not start.	Set PositiveLimit and NegativeLimit within the range.
1A8AH	Axis	Out of Overrun Operation Setting Range	The value is out of range for AxisName(AxesGroupName).Pr.OverrunOperation.	At start	Not start.	Set AxisName(AxesGroupName).Pr.OverrunOperation within the range.

Error code (Hexadecimal)	Error (System/axis/axes group)	Error name	Error detail and cause	Detection timing (At start/during control	Operation at error occurrence	Remedy
1A90H	Axis	Drive Module Via Synchronous Encoder Setting Incorrect	• The real drive axis with the same station address as the real encoder axis via drive module is not set. (Detail Code: 0001H) • The scale measurement function of the connected driver is disabled. (Detail Code: 0002H) • The slave object required to be set for the real encoder axis via drive module does not exist. (Detail Code: 0003H) • Scale measurement encoder Resolution is 0 (Detail Code: 0004H) • The slave object required to be set for the real encoder axis via drive module is not mapped to the real drive axis with the same station address. (Detail Code: 0005H) • Absolute position management settings do not match. (Detail Code: 0006H)	At power ON	Generating axis fails.	• Set the real drive axis with the same station address as the real encoder axis via drive module. • Enable scale measurement function of the connected driver. • Check if the slave object required to be set for the real encoder axis via drive module exists. • Map the slave object required to be set for the real encoder axis via drive module to the real drive axis with the same station address. • Connect the encoder whose Scale measurement encoder Resolution is not 0. • Match the absolute position management settings of the real encoder axis and the connected device.
1A91H	Axis	Axis No. Setting Duplication Error	The same axis No. has been set.	At power ON	Generating axis fails.	Set a different axis No.
1A94H	Axis	Station Address Duplication Error	The set of IP address and multiple axes ID of a real drive axis overlaps with that of another.	At power ON	Generating axis fails.	Set real drive axes and sets of IP address and multiple axes ID to one on one.
1A95H	Axis	Station Address Setting Incorrect	There are no specified station address or connection devices in the real axis that not use slave emulation function.	At power ON	Generating axis fails.	• Specify the station address where a device is connected, when slave emulation function is not used.
1A96H	Axis	Slave Object Setting Incorrect	• The signal set in the slave object data (TARGET_REF type) does not exist. • The signal set in the slave object data (TARGET_REF type) can not be used as slave object. • The signal of the slave object data that needs to be set to each axis type is not set • Changed from initial value when AxisName.PrConst.FastOperationMode is 5FE2H (high-speed mode).	At power ON	Generating axis fails.	• Please specify the data that can be set as slave object. • Please set signal for the slave object data that needs to be set to each axis type. • Please check whether it is changed from initial value when setting high-speed mode.
1A97H	Axes group	Axes Group Status Incorrect (Axes Group Disable)	MC_GroupDisable was executed when AxesGroupName.Md.GroupStatus is other than GroupStandby or "GroupErrorStop".	Axes Group Disable	Stops the operation.	Execute MC_GroupDisable in the status where AxesGroupName.Md.GroupStatus is GroupStandby or GroupErrorStop.

Error code (Hexadecimal)	Error (System/ axis/axes group)	Error name	Error detail and cause	Detection timing (At start/ during control	Operation at error occurrence	Remedy
1A98H	Axes group	No Configuration Axis	Configuration axis is not set in axes group.	At power ON / During Configuration Axis Write	• [At power ON] Generating axes group fails. • [During Configuration Axis Write] In operation, the operation is stopped.	Set configuration axis in axes group.
1A99H	Axes group	No axis	The non-existent axis was set in configuration axis.	At power ON / During Configuration Axis Write	• [At power ON] Generating axes group fails. • [During Configuration Axis Write] In operation, the operation is stopped.	Set the existing axis in configuration axis.
1A9AH	Axis	Absolute Position Management Setting Mismatch	The ENCODER_ABS_STATUS object setting in connection devices is not match.	When restoring current position	Current position is not restored.	Please correct the axis and absolute position management setting of connection devices.
1AA3H	Axis	Out of Ring Counter Range	• The setting of ring counter upper limit value and lower limit value exceeds the positioning range. • Ring counter lower limit value > Ring counter upper limit value. • When 0 < |Ring counter upper limit value - Ring counter lower limit value| < 2.0 • The setting value exceeded the positioning range.	At power ON	Generating axis fails.	• Set the ring counter upper limit value and lower limit value within the positioning range. • Set the values so that they meet the following condition: Ring counter lower limit value < Ring counter upper limit value. • Set the values so that they meet the following condition: 2.0 < |Ring counter upper limit value - Ring counter lower limit value|.
1AA4H	Axis	Driver Unit Change Magnification Upper Over	The magnification that exceeded upper limit in the driver unit conversion numerator/ denominator is set.	At ready ON	• Generating axis fails. • The driver unit conversion numerator/ denominator is not changed.	Set values to make sure the calculation result of driver unit conversion numerator/denominator does not exceed 400000000.
1AA5H	Axis	Driver Unit Change Numerator/ Denominator Change Timing Incorrect	The driver unit conversion numerator/denominator setting was changed when follow-up operation can not be executed.	At ready ON	The driver unit conversion numerator/ denominator is not changed.	READY ON when follow-up is possible. Meet both of the following. • AxisName.Md. AxisStatus is Disabled. • AxisName.Md.FollowupDisable is FALSE.
1AA7H	Axis	Add-on Acquisition Failure	Failed to acquire the external signal selection add-on when have 'counter disable signal' setting.	At power ON	Generating axis fails.	Install external signal add-on and enable it.
1AA8H	Axis/Axes group	Necessary Slave Object Unset	Necessary slave object in control command execution is not set to axis (occasion configuration axis of axes group).	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	• Please set necessary object in control command. • Please use single axis synchronization for virtual link axis.

Error code (Hexadecimal)	Error (System/axis/axes group)	Error name	Error detail and cause	Detection timing (At start/during control	Operation at error occurrence	Remedy
1AA9H	Axis/Axes group	Out of Acceleration/Deceleration Method Range	The value is out of range for acceleration/deceleration method of Options.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	Set the acceleration/deceleration method according to the executed control command.
1AAAH	Axis	ABS Reference Point Read Error	Failed to get the data from drive module normally at homing.	During operation	Homing is not completed normally.	• Execute homing again. • When reading or writing parameters of a slave device while executing homing, reduce the number of executions at a time.
1AABH	Axes group	Reference Axis Movement Amount 0	The movement amount of the reference axis is 0 in the linear interpolation for reference axis specification.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	Set the axis of which the movement amount is not 0 to a reference axis.
1AACH	Axis	Out of Torque Lamp Function Selection Range	The value is out of range for torque lamp function selection of Options.	At start	• [At start] Not start. • [At multiple start] Stops the operation.	Set torque lamp function selection of Options within the range.
1AADH	Axis	Start Not Possible	Positioning start not possible. • Moderate Error Detection (Detail Code: 0001H) • PLC READy OFF (Detail Code: 0002H) • Forced Stop Signal ON (Detail Code: 0003H) • Axis Stop Signal ON (Detail Code: 0004H) • Axis/Axes Group Status Incorrect (Detail Code: 0100H) • Axes Group Operating (Detail Code: 0101H) • Configuration Axis Status Incorrect (Detail Code: 0102H)	At start	Not start.	• Set it so that it will not start when PLC READY is OFF. • Set it so that it will not start when READY is OFF. • Check the sequence program which turns PLC READY signal [Y0] ON/OFF. • Check the status of the drive module power supply, the wiring with the drive module and connector status. • Start it after AxesGroupName.Md.GroupStatus changes into GroupStandby. • Start it after AxisName.Md.AxisStatus changes into Standstil • Disable axes group for axes where AxisName.Md.UseInGroup is TRUE.
1AAEH	Axis	Driver Error	An error occurred in the driver.	During operation	Stops immediately.	Please check the error details and corrective actions according to the content of AxisName.Md.DriverErrorID. (Please refer to the instruction manual of each driver for details of AxisName.Md.DriverErrorID.)
1AAFH	Axis	Driver Ready OFF	Driver power OFF during controlling. • Driver power is OFF • Initial processing when driver power is ON • The driver is not installed • Driver error occurs	During operation	Stops immediately.	Please turn on the power supply of driver.

Error code (Hexadecimal)	Error (System/axis/axes group)	Error name	Error detail and cause	Detection timing (At start/during control	Operation at error occurrence	Remedy
1AB0H	Axis	PLC READY OFF During Operation	PLC READY signal [Y0] was turned OFF during operation.	During operation	Stops the operation.	Check the sequence program of which PLC READY signal [Y0] is ON/OFF.
1AB1H	Axis	Module Error During Operation	Moderate error or major error was detected in the Motion module during operation.	During operation	Stops the operation.	Please turn the module power on again.
1AB2H	Axis	Driver Network Disconnection	Driver network disconnection was detected during operation.	During operation	Stops immediately.	Please check the connection cable of driver.
1AB3H	Axis	Executing FB Analysis Disabled	An error was detected in executing FB analysis at target position changing. • Position/Distance after change is out of range. • Target position after change is out of range of software stroke limit.	At control change	Stops the operation.	Set Position/Distance at target position changing within the range.
1AB4H	Axis/Axes group	Buffering FB Analysis Disabled	An error was detected in the buffering FB analysis.	During operation	Stops the operation.	Set buffering FB analysis so that it will not result in an error.
1AB5H	Axis	Ring Counter Invalid Start	Ring counter setting was started in the invalid status.	At start	Not start.	Enable the AxisName.PrConst.RingCount_Enable.
1AB6H	Axis	External Signal String Incorrect	Incorrect string was set in the string of external signal.	At start	Not start.	Set a valid string in the external signal.
1AB7H	Axes group	Path Choice Setting Incorrect	Incorrect value was set in PathChoice.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	Set the value that supports CircMode in the PathChoice.
1AB8H	Axes group	Total Movement Angle 0	The angle between start point - end point became 0 at circular interpolation of boundary point specification or radius specification.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	Set EndPoint or PathChoice so that the angle of start point - end point will not be 0.
1AB9H	Axes group	Linear Interpolation Axis Setting Incorrect	• A configuration axis not set up was set in LinearAxes. • 0 was set in the first element of LinearAxes. • Same index No. was set in LinearAxes in duplicate. • The number of axes set for LinearAxes exceeded the maximum number of interpolation axes.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops the operation.	• Set an axis that has been set up as configuration axis in LinearAxes. • Set a configuration axis in the first element of LinearAxes. • Set index No. of configuration axes in LinearAxes avoiding duplication. • Set the number of axes set for LinearAxes less than the maximum number of interpolation axes.
1ACBH	Axis	External Signal Selection Add-on Acquisition Failure	Failed to acquire the external signal selection add-on.	At power ON At ready ON	The signal is detected continuously.	Install SignalIO.mpk and enable it.
1AD8H	Axis	Out of Master Axis No. Range	The value is out of range for Master.AxisNo.	At start When executing FB	Not execute FB.	• Set a value within the range for Master.AxisNo. • Set the axis No. in which instance is created.

Error code (Hexadecimal)	Error (System/ axis/axes group)	Error name	Error detail and cause	Detection timing (At start/ during control	Operation at error occurrence	Remedy
1AD9H	Axis	Out of Slave Axis No. Range	The value is out of range for Slave.AxisNo.	At start When executing FB	Not execute FB.	• Set a value within the range for Slave.AxisNo. • Set the axis No. in which instance is created.
1ADAH	Axis	Out of BufferMode Range	The value is out of range for BufferMode.	At start When executing FB	Not execute FB.	Set a value within the range for BufferMode.
1ADBH	System	Command Filter Add-on System Memory Shortage	System memory for the add-on is insufficient..	At start At multiple start	Not start.	Please check the System.PrConst.Addon_MotionControl_Axis Filter.RamSizeMax.
1ADCH	System	Add-on Memory Shortage	System memory for add-ons related to operation is insufficient..	At start At multiple start	Not start.	Please check the System.PrConst.Addon_*.RamSizeMax for related add-ons
1ADEH	Axes group	Out of Linear Movement Amount Range	The interpolation axis exists in which movement amount exceeds '4,294,967,296.0 (= 2^{32})' in the linear interpolation for composite speed specification.	At start At multiple start	• [At start] Not start. • [At multiple start] Stops.	Set the value of Position or Distance so that the movement amount of each interpolation axis does not exceed '4,294,967,296.0 (= 2^{32})'.
1ADFH	Axis	Insufficient Current Position Backup Capacity	The necessary memory capacity in backup was not secured when restoring current position.	When restoring current position	Restoring current position is not completed.	Please execute soft reboot after checking the System.PrConst.Addon_AbsSystem.BackpRamSizeMax.
1AE0H	Axis	Driver Restoration Data Incorrect	The parameter of connected driver device was incorrect when restoring current position.	When restoring current position	Restoring current position is not completed.	• Please change the Electronic Gear to a power of two at connected driver side. • Please change the Electronic Gear to 1:1 at connected driver side, if the connected encoder resolution is not a power of two.
1AE1H	Axis	Out of Encoder Ring Counter Setting Range	Encoder Ring counter lower limit value > Encoder Ring counter upper limit value.	At power ON	Generating axis fails.	• Set the values so that they meet the following condition: Encoder Ring counter lower limit value < Encoder Ring counter upper limit value.
1AE2H	Axis	Out of Addition and Subtraction Selection Range	The value is out of range for CombineMode.	At start	Not start.	Set CombineMode within the setting range.
1AE3H	Axis	csp Unsupported Driver	The driver does not support csp.	At initial communicating	Unable to connect to the driver	Please connect a driver that supports csp.
1AE4H	Axis/Axes group	Out of Calculation Cycle Converted Speed Range	The speed converted from Velocity by calculation cycle is out of the range.	At start	Not start.	Set Velocity so that the speed converted by calculation cycle is within the range.
1AE5H	Axis	Communication Error of Connected Encoder	A communication error occurred between the driver <=> encoder.	When connecting During control	• [When connecting] Disable to connect encoder. • [During control] Disconnects encoder.	Pleasecheck the motor and encoder cable.

Error code (Hexadecimal)	Error (System/ axis/axes group)	Error name	Error detail and cause	Detection timing (At start/ during control	Operation at error occurrence	Remedy
1AE6H	Axis	Driver Command Discard Detection	The command discard (Statusword Bit12 OFF) of the connceted driver device was detected during operation.	During operation	Stops the operation.	Please move the command by JOG instruction etc. to the position where the limit signal was detected when using limit signal for driver device.(Please check the specification document of the driver device for details of Statusword)
1AE7H	Axis	Driver Execution Control Mode Incorrect	The control mode of the connected driver device switched to an unsupported mode for the motion system during operation.	During operation	Stops immediately.	Switch the driver control mode to a control mode that is supported by the motion system
1AE8H	Axis/Axes group	Speed Range Over during Controlling	Command current speed exceeded the speed command range.	During operation	Stops the operation.	Set command current speed so that it does not exceed the speed command range.
1AE9H	Axis	Driver Control Mode Unsupported	The driver was switched to an unsupported driver control mode. • ct: Continuous operation to torque control (Detail Code: 0001H)	At start	Not start	Connect a driver that supports the driver control mode to be switched to.
1AEAH	System	Servo Driver Memory Size Over	The memory size of the servo driver exceeds the memory capacity set in system.	When connecting	Unable to connect to the driver	• Please increase the memory capacity set in system. • Please delete unnecessary axes and reduce memory size.
1C40H	System	Network Parameter Error (Total No. of Stations < Station No.)	The station No. of slave stations that connected to the network exceeds the total number of slave stations that has been set in "Network Configuration Setting".	At power ON	The station can not be connected to the drive unit.	Set the station No. of slave stations that connected to the network within the total number of slave stations.
1C41H	Axis	Watchdog Counter Error	A watchdog counter error of driver was detected.	During communication	Stops immediately.	• Please check the communication cycle. • Please check surrounding environment errors such as noise etc., and remove the cause.
1C42H	Axis	Unsupported Driver Connection	An unsupported driver was connected.	When connecting driver	Creating an instance of driver fails.	Please connect a supported driver.
1C43H	Axis	SLMP Communiction Error	An error occurred in SLMP communication.	During communication	[During initial communication] • Unable to connect to the driver	• Please check the slave object setting. • Please check the specification of the driver.
1C44H	System	Network Parameter Error (Station No. Order Error)	The station No. of the slave station to connect to network are not in ascending order for "Network Configuration Settings".	At power ON	The station can not be connected to the drive unit.	Set station No. of the slave station to connect to network in ascending order.

Error code (Hexadecimal)	Error (System/ axis/axes group)	Error name	Error detail and cause	Detection timing (At start/ during control	Operation at error occurrence	Remedy
1C45H	System	SLMP Communiction Error (Timeout)	Response timeout was detected in SLMP communication.	During communication	Disable SLMP communication.	Please check the transient transmission time. It is derived by the following formula.(Transient Transmission Time = Communication Cycle Interval Setting - Cyclic Transmission Time - System Reserved Time)
1C46H	System	Time Synchronization Error	There is a disconnected station where time synchronization is incomplete.	At start communication	The station can not be connected to the drive unit.	• Please change to a short communication cycle. • Please check surrounding environment errors such as noise etc., and remove the cause. • Please connect the stations only set in the network configuration setting.
1C47H	System	Communication Cycle Unsupported Driver Connection	An MR-J5(W)-G that does not support a communication cycle of 31.25/62.5 μs was connected.	At start communication	The station can not be connected to the drive unit.	Please check the version of MR-J5-G series amplifier and apply the latest version.
1C48H	System	PDO Mapping Setting Error	Abnormal response for PDO mapping setting was received from a slave station.	At start communication	The station can not be connected to the drive unit.	Please check the details of the error according to the error code (abort code) and review the PDO mapping setting.
1C80H	System/ Axis	Cycle Over	Arithmetic processing or other fixed scan processing is not completed within the set cycle.	During operation	The axis in operation decelerates and stops.	• Change the fixed scan interval setting of fixed scan execution type program to a larger value. • Change the following to lager values. • System.PrConst.SystemBaseCycle.Cycle • System.PrConst.OperationCycle[1].Cycle • System.PrConst.BuffermemoryRefreshCycle.Cycle
1C81H	System	Cycle Assignment Incorrect	The value is out of range for system basic cycle, the first calculation cycle, buffer memory refresh cycle, or the value which cannot be set was specified.	At power ON	Operates in default operation cycle.	Please set the following within the range. • System.PrConst.SystemBaseCycle.Cycle • System.PrConst.OperationCycle[1].Cycle • System.PrConst.BuffermemoryRefreshCycle.Cycle
1C83H	System	Boot Software File Error	The file of boot software is abnormal.	At power supply	The motion system does not start.	Please install the boot software again
1C84H	System	Saved File Number Over	The saved file reached the upper limit value.	At saving files	The files can not be saved.	Please delete all the saved files.

Error code (Hexadecimal)	Error (System/axis/axes group)	Error name	Error detail and cause	Detection timing (At start/during control	Operation at error occurrence	Remedy
1C85H	System	Event History File Incorrect	• An error was detected in the event history file. • The memory capacity of add-on MotionEventHist is insufficient.	At updating the event history file.	Updating the event history file stops.	• Delete unnecessary files and secure free capacity. • Format the target drive. • Delete event history files. • Please check the memory capacity of add-on MotionEventHist.
1C86H	System	Out of Cycle Over Error Selection Range	The value is out of range for Cycle Over Error Selection.	At power ON	Operates in default cycle over error type.	Please set the following within the range. • System.PrConst.SystemBaseCycle.CycleOverErrorType • System.PrConst.OperationCycle[1].CycleOverErrorType • System.PrConst.BuffermemoryRefreshCycle.CycleOverErrorType
1C87H	System	Label Incorrect	Failed to import the label definition.	At power ON	Interrupts the process to import the labels.	Please write label setting.
1C88H	System	Label Memory Shortage	Label which exceeds the label capacity has been defined.	At power ON At ready ON	Interrupts the process to import the labels.	• Please change the System.Md.LabelMemoryFreeSize to a larger value. • Please check the label usage.
1C8EH	System	Variable Manager Acquisition Failure	Failed to acquire variable manager.	At power ON	The add-on which is discribed in the detailed information is invalid function.	Please re-install the baseSystem.smpk
1C8FH	System	External Signal Selection Add-on Acquisition Failure	Failed to acquire the external signal selection add-on.	At power ON At ready ON	[At power ON] Generating axis fails. [At ready ON] The READY [X0] does not turn ON.	Install SignalIO.mpk and enable it.
1C91H	System	Insufficient Memory Capacity of External Signal Selection Add-on	Cannot create a signal, because the memory capacity of the external signal selection add-on is insufficient.	At power ON At ready ON	[At power ON] Generating axis fails. [At ready ON] The READY [X0] does not turn ON.	Increase the memory capacity of add-on SignalIO.
1C93H	System	Network Driver Memory Size Over	The memory size of the network driver exceeds the memory capacity set in system.	At power ON	SLMP communication error.	• Please increase the memory capacity set in system. • Please delete unnecessary stations and reduce memory size.
1C94H	System	Backup Data Saving Failure	Failed to save the backup data because of insufficient capacity in the Ich drive.	During operation	Files are not saved.	• Please check the free capacity in the Ich drive and delete unnecessary files. • Please check the capacity of motion event histories if the destination to save motion event histories is the Ich drive.

Error code (Hexadecimal)	Error (System/ axis/axes group)	Error name	Error detail and cause	Detection timing (At start/ during control	Operation at error occurrence	Remedy
1C95H	System	Label Instance Error	An error occurred in the corresponding add-on for the label instances.	At power ON At ready ON	Interrupts the process to import the labels.	• Please check the version of the corresponding add-on for the labels. • Please increase the maximum RAM size of the corresponding add-on for the labels.
1C96H	System	Device Label Import Error	• There are inconsistencies in network configuration settings and the settings at device label creation. • Failed to import device labels.	At power ON	Interrupts the process to import the labels.	Please check the settings of network configuration and then create device labels again.
1D80H	System	Out of Parameter Range (Axis)	Parameter is out of the range.	When executing FB At power ON At ready ON	• [When executing FB] The error of FB turns ON and the command is not executed. • [At power ON] Generating axis fails. • [At ready ON] The READY [X0] does not turn ON.	Change the specified value of the parameter in detailed information within the range.
1D81H	System	Out of Parameter Range (Axes Group)	Parameter is out of the range.	When executing FB At power ON At ready ON	• [At FB] The error on FB turns ON and the command is not executed. • [At power ON] Generating axes group fails. • [At ready ON] The READY [X0] does not turn ON.	Change the specified value of the parameter in detailed information within therange.
1D82H	System	Out of Parameter Range (System)	Parameter is out of the range.	When executing FB At power ON At ready ON	• [At FB] The error of FB turns ON and the command is not executed. • [At power ON/At ready ON] The READY [X0] does not turn ON.	Change the specified value of the parameter in detailed information within therange.
1D83H	System	Parameter Write-protected	Wrote in the parameter which can only read.	When executing FB	The error of FB turns ON and the command is not executed.	Change to the parameter that can be written.
3200H	System	File Access Error	The following operations was attempted in the engineering tool and SLMP path. • An access (read/write) from other request source to written file was operated. • A write from other request source to access (read/write) file was operated.	When accessing file	Fails to access the files.	Please operate the next access after one process of request source is completed.

Error code (Hexadecimal)	Error (System/ axis/axes group)	Error name	Error detail and cause	Detection timing (At start/ during control	Operation at error occurrence	Remedy
3203H	System	WDT Error Detected	• Programming It exceeded 2.0 [s] from start to end of the execution. (Initial execution type) The execution time of all normal execution type programs exceeded 1.0 [s]. • Execution time monitor Calculation process or other processes are in abnormal conditions. Processing stopped for more than one second.	During operation	The axis in operation stops immediately.	Programming • Reduce the number of instruction executions of initial execution type and normal execution type programs. Execution time monitor • Change the system basic cycle, the first calculation cycle, or buffer memory refresh cycle to a larger value. • Change the fixed scan interval setting of fixed scan execution type programs to a larger value.
3205H	System	Add-on Library Load Error	Add-on • The file of add-on is broken. • Add-on with dependencies has not been installed. • Add-on that cannot be combined has been installed. • Inconsistency occurred in versions.	At power ON	Add-on The motion system does not start.	Add-on • Please install the software again. • Please install the add-on with dependencies. • Please delete the add-on that cannot be combined. • Please uninstall the add-on with version inconsistency.
3207H	System	Drive Error	Failed to mount the drive Check disk ended with error Verification error of plane management target memory occurred.	At system start	Access to the target drive is not allowed.	Format the target drive.
3208H	System	Insufficient Total System Memory (RAM)	The total number of System.PrConst.Addon_*.RamSizeMax of all add-ons exceeds the total size of system memory (RAM).	At power ON	The motion system does not start.	Please make the total number of System.PrConst.Addon_*.RamSizeMax of all add-ons not exceed the total size of system memory (RAM).
3209H	System	Insufficient Add-on Total System Memory (RAM)	Memory that exceeded System.PrConst.Addon_*.RamSizeMax has been used.	At power ON During operation	• [At power ON] The motion system does not start. • [During operation] WDT error occurs if the function has an effect on control.	Please increase System.PrConst.Addon_*.RamSizeMax.
320AH	System	Insufficient Total System Memory (Backup RAM)	The total number of System.PrConst.Addon_*.BackupRamSizeMax of all add-ons exceeds the total size of system memory (Backup RAM).	At power ON	The motion system does not start.	Please make the total number of System.PrConst.Addon_*.BackupRamSizeMax of all add-ons not exceed the total size of system memory (Backup RAM).
320BH	System	Insufficient Add-on System Memory (Backup RAM)	Memory that exceeded System.PrConst.Addon_*.BackupRamSizeMax has been used.	At power ON During operation	• [At power ON] The motion system does not start. • [During operation] WDT error occurs if the function has an effect on control.	Please increase System.PrConst.Addon_*.BackupRamSizeMax.

Error code (Hexadecimal)	Error (System/ axis/axes group)	Error name	Error detail and cause	Detection timing (At start/ during control	Operation at error occurrence	Remedy
320CH	System	Cycle Over	Arithmetic processing or other fixed scan processing is not completed within the set cycle.	During operation	The axis in operation stops immediately.	• Change the fixed scan interval setting of fixed scan execution type program to a larger value. • Change the following to a lager values. • System.PrConst.SystemBaseCycle.Cycle • System.PrConst.OperationCycle[1].Cycle • System.PrConst.BuffermemoryRefreshCycle.Cycle
320DH	System	Base System Error	• Failed to configure label. • Failed to process initial.	At power ON	The motion system does not start.	• Please install the software again. • Please write the program again.
320EH	System	Network Driver Error	Failed to communicate with Motion Module and Network Module.	At power ON During operation	• The motion system does not start. • The axis in operation stops immediately.	• Please install the software again. • Please write the program again.
320FH	System	Servo Driver Error	Failed to initialize the servo driver.	At power ON During operation	• The motion system does not start. • Creating an instance of driver fails.	Please reinstall the software.
3212H	System	Public Label Address Check Error	Refresh target of public label is incorrect.	At power ON	Interrupts the process to import the labels.	Please install the latest add-on and engineering tool.
3400H	System	Out of Axis No. Range	• The value is out of range for axis No.. • Invalid axis No. has been set.	At start When executing FB	Not execute FB.	• Set the value within the range of 1 to 10000 • Set the axis No. in which instance is created
3401H	System	Axis No. Duplication Setting	The same axis No. has been set to master axis and slave axis.	At start When executing FB	Not execute FB.	Set different axis No. to master axis and slave axis.
3402H	System	Out of Axes Group No. Range	• The value is out of range for axes group No.. • Invalid axes group has been set.	When executing FB	Not execute FB.	• Set the value within the range of 1 to 10000 • Set the axes group No. in which instance is created
3403H	System	Out of Master Axis No. Range	• The value is out of range for master axis No.. • Invalid axis No. has been set.	At start When executing FB	Not execute FB.	• Set the value within the range of 1 to 10000 • Set the axis No. in which instance is created
3404H	System	Out of Slave Axis No. Range	• The value is out of range for slave axis No.. • Invalid axis No. has been set.	At start When executing FB	Not execute FB.	• Set the value within the range of 1 to 10000 • Set the axis No. in which instance is created
3405H	System	Out of Master Axis 1 No. Range	• The value is out of range for axis No. of master axis 1. • An unavailable axis No. is set.	At start When executing FB	Not execute FB.	• Set the value within the range of 1 to 10000 • Set the axis No. in which instance is created

Error code (Hexadecimal)	Error (System/axis/axes group)	Error name	Error detail and cause	Detection timing (At start/during control	Operation at error occurrence	Remedy
3406H	System	Out of Master Axis 2 No. Range	• The value is out of range for axis No. of master axis 2. • Invalid axis No. has been set.	At start When executing FB	Not execute FB.	• Set the value within the range of 1 to 10000 • Set the axis No. in which instance is created
3407H	System	Out of Latch Count Range	The value is out of range for RatchCount.	When executing FB	Not execute FB.	Set the value within the range.
3408H	System	Latch Count Incorrect	RecordCount has not been set when RecodeMode = RecordCount,RingBuffer.	When executing FB	Not execute FB.	Set RecordCount.
3409H	System	Out of Touch Probe ID Range	The value is out of range for TouchProbeID.	When executing FB	Not execute FB.	Set a valid touch probe ID.
340DH	System	Out of Compensation Time Range	The value is out of range for CompensationTime.	When executing FB	Not execute FB.	Set the value within the range of -5.0 to 5.0 [s].
340EH	System	Out of Latch Mode Range	The value is out of range for RatchMode.	When executing FB	Not execute FB.	Set the value within the range for MC_RECORD_MODE.
340FH	System	No Calculation Profile	The calculation profile does not exist.	During operation	The execution of FB is interrupted.	Set the existing calculation profile.
3410H	System	Calculation Profile Incorrect	• The value is out of range for Data or Version. • Add-on that supports the calculation profile format has not been installed.	During operation	The execution of FB is interrupted.	• Set the data within the range. • Install and enable the add-on that supports the calculation profile format.
3411H	System	Calculation Profile Operating	The operation was executed during calculation profile operating.	During operation	The execution of FB is interrupted.	Operate it when Busy of FB is FALSE.
3412H	System	Calculation Profile Error	The calculation profile is broken, or the format is incorrect.	During operation	The execution of FB is interrupted.	Set the calculation profile for a correct format.
3413H	System	Out of Offset Range	Offset exceeds the number of elements of calculation profile (file or open area).	During operation	The execution of FB is interrupted.	Set a value which does not exceed the number of elements of calculation profile (file and open area) in Offset.
3414H	System	Out of Calculation Profile ID Range	The value is out of range for ProfileData.ID.Number or ProfileID.Number.	At start During operation	• [At start] Not execute FB. • [During operation] Operates with previous setting value.	Set a value within the range for ProfileData.ID.Number or ProfileID.Number.
341DH	System	Out of Input Absolute Coordinate Range	The value is out of range for InputsAbsolute.	When executing FB	Not execute FB.	Set the value within the range.
341EH	System	Out of Output Absolute Coordinate Range	The value is out of range for OutputAbsolute.	When executing FB	Not execute FB.	Set the value within the range.
341FH	System	Out of Composite Mode Range	The value is out of range for SynthesizingMode.	When executing FB	Not execute FB.	Set the value within the range.
3421H	System	Out of Input Offset Range	The value is out of range for InputsOffset.	When executing FB	Not execute FB.	Set the value within the range.
3422H	System	Out of Output Offset Range	The value is out of range for OutputOffset.	When executing FB	Not execute FB.	Set the value within the range.
3423H	System	Out of Input Coefficient Range	The value is out of range for InputsScaling.	When executing FB	Not execute FB.	Set the value within the range.
3424H	System	Out of Output Coefficient Range	The value is out of range for OutputScaling.	When executing FB	Not execute FB.	Set the value within the range.
342FH	System	SD Memory Card Unload	The memory card was removed without SD memory card being disabled.	When removing SD memory card	—	Remove SD memory card after SD memory card being disabled.

Error code (Hexadecimal)	Error (System/axis/axes group)	Error name	Error detail and cause	Detection timing (At start/during control	Operation at error occurrence	Remedy
343FH	System	Out of Backlash Amount Range	The value is out of range for BacklashAmount.	When executing FB	Not execute FB.	Set the value within the range.
344EH	System	MC_SetPosition Instruction Error	An error occurred during executing MC_SetPosition for the following causes. • Position is out of range (Detail Code: 0001H) • ExecutionMode is out of the range (Detail Code: 0002H) • The current position after changing is out of range of software stroke limit (Detail Code: 0003H) • Axis status at execution is incorrect (Detail Code: 0004H) • Other MC_SetPosition was executed in the specified axis (Detail Code: 0008H) • Options is out of the range (Detail Code: 0009H)	At start During operation	• Not execute FB. • The execution of FB is interrupted.	• Set Position within the range. • Set ExecutionMode within the range. • Set the values so that the current position after changing is within the software stroke limit range. • Execute it when the axis status is Standstill. • Set Options within the range.
344FH	System	MCv_SetPositionTriggered Instruction Error	An error occurred during executing MCv_SetPositionTriggered for the following causes. • Position is out of range (Detail Code: 0001H) • ExecutionMode is out of range (Detail Code: 0002H) • The current position after changing is out of range of software stroke limit (Detail Code: 0003H) • Axis status at execution is incorrect (Detail Code: 0004H) • TriggerInput is incorrect (Detail Code: 0005H, 0006H) • Add-on SignalIO is not enabled. (Detail Code: 0007H) • Other MCv_SetPositionTriggered was executed in the specified axis (Detail Code: 0008H) • Options is out of range (Detail Code: 0009H)	At start During operation	• Not execute FB. • The execution of FB is interrupted.	• Set Position within the range. • Set ExecutionMode within the range. • Set the values so that the current position after changing is within the software stroke limit range. • Execute it when the axis status is Standstill. • Please check the setting of TriggerInput. • Please enable the add-on SignalIO. • Set Options within the range.
3450H	System	MCv_SetTorqueLimit Instruction Error	An error occurred during executing MCv_SetTorqueLimit forthe following causes. • PositiveValue is out of range (Detail Code: 0001H) • NegativeValue is out of range (Detail Code: 0002H) • ExecutionMode is out of range (Detail Code: 0003H) • Axis type is incorrect (Detail Code: 0004H) • Axis status at execution is incorrect (Detail Code: 0005H) • Other MCv_SetTorqueLimit was executed in the specified axis (Detail Code: 0006H) • Options is out of range (Detail Code: 0007H)	At start During operation	Not execute FB.	• Set PositiveValue within the range. • Set NegativeValue within the range. • Set ExecutionMode within the range. • Execute it for an axis whose axis type is real drive axis. • Execute it when the axis status is Standstill. • Set Options within the range.
3451H	System	Out of Start Mode Range	The value is out of range for ExecutionMode.	At start	Not execute FB.	Set a value within the range for ExecutionMode.

Error code (Hexadecimal)	Error (System/axis/axes group)	Error name	Error detail and cause	Detection timing (At start/during control	Operation at error occurrence	Remedy
3452H	System	Insufficient Calculation Profile ID	The profile ID to be assigned automatically is insufficient.	During operation	The execution of FB is interrupted.	Specify and open the unused profile ID.
3453H	System	Calculation Profile Control Unsupported	Control FB was executed for the unsupported calculation profile.	At start	Not execute FB.	Open the calculation profile which supports the calculation profile control FB in the open area.
345AH	System	Out of Output Offset Range	The value is out of range for OutputOffset.	At start During operation	• [At start] Not execute FB. • [During operation] Operates with previous setting value.	Set OutputOffset within the range.
345BH	System	Out of Input Offset Range	The value is out of range for InputsOffset.	At start During operation	• [At start] Not execute FB. • [During operation] Operates with previous setting value.	Set a value within the range for InputsOffset.
345CH	System	Out of Output Coefficient Range	The value is out of range for OutputScaling.	At start During operation	• [At start] Not execute FB. • [During operation] Operates with previous setting value.	Set a value within the range for OutputScaling.
345DH	System	Out of Input Coefficient Range	The value is out of range for InputsScaling.	At start During operation	• [At start] Not execute FB. • [During operation] Operates with previous setting value.	Set a value within the range for InputsScaling.
345EH	System	Calculation Profile Read/Write Unsupported	The calculation profile which does not support calculation profile read/write is set.	During operation	The execution of FB is interrupted.	Set the calculation profile which supports read/write.
345FH	System	Out of Read Target/Write Target Range	The value is out of range for Target.	During operation	The execution of FB is interrupted.	Set a value within the range for Target.
3460H	System	Read/Write Data Data Type Incorrect	The incorrect data type is set in Data1 or Data2.	During operation	The execution of FB is interrupted.	Set a normal data type in Data1 and Data2.
3461H	System	Read/Write Data Data Name Specification Method Incorrect	The incorrect data name specification method is set in Data1 or Data2.	During operation	The execution of FB is interrupted.	Set a normal data name specification method in Data1 and Data2.
3462H	System	Read/Write Data Type Incorrect	The incorrect type is set in Data1 or Data2. Add-on that supports the calculation profile format has not been installed.	During operation	The execution of FB is interrupted.	Set a normal type in Data1 and Data2. Install and enable the add-on that supports the calculation profile format.
3463H	System	Read/Write Data Target Modification Incorrect	The incorrect target modification is set in Data1 or Data2.	During operation	The execution of FB is interrupted.	Specify a normal target modification in Data1 and Data2.
3464H	System	Number of Read/Write Data Mismatch	Points do not match the number of elements of read/write structured data type.	During operation	The execution of FB is interrupted.	Set Points to match with the number of elements of read/write structured data type.
3465H	System	Number of Offset/Read/Write Data Incorrect	A value other than Offset = 0, Points = 0 is set when creating a new file.	During operation	The execution of FB is interrupted.	Set Offset = 0, Points = 0 when creating a new file.
3466H	System	Out of Current Value per Cycle Range	The value is out of range for Cycle.	During operation	The execution of FB is interrupted.	Set a value within the range for Cycle.
3467H	System	Current Value per Cycle Change Unsupported	FB which does not support the current value per cycle change is set.	At start	Not execute FB.	Set FB which supports the current value per cycle change.
3468H	System	No Instance ID	The set instance ID does not exist.	At start	Not execute FB.	Set the existing instance ID.

Error code (Hexadecimal)	Error (System/axis/axes group)	Error name	Error detail and cause	Detection timing (At start/during control	Operation at error occurrence	Remedy
3470H	System	SD Memory Card is not Inserted	SD memory card is not inserted.	During operation	The execution of FB is interrupted.	Execute it after SD memory card inserted.
3471H	System	SD Memory Card is Write-protected	SD memory card is write-protected.	During operation	The execution of FB is interrupted.	Execute it after cancelling SD memory card write-protected.
3483H	System	Out of Cam Switch ID Range	The value is out of range for Switches.	When executing FB	Not execute FB.	Set the value within the range.
3484H	System	No Digital Cam Switch	The calculation profile specified in Switches does not exist.	When executing FB	Not execute FB.	Set the existing calculation profile.
348CH	System	Out of Valid Track No. Range	The value is out of range for EnableTrack.	When executing FB	Not execute FB.	Set the value within the range.
3492H	System	Calculation Profile Add-on System Memory Shortage	System memory for the add-on is insufficient.	At ready ON	The program stops.	• Please check the System.PrConst.Addon_ProfileControl.RamSizeMax. • Please check the setting to reduce the usage of the open area. • Please reduce the number of FBs to be executed simultaneously.
3493H	System	Calculation Profile Add-on Internal Error	An error occurred in internal of calculation profile add-on.	At ready ON	The program stops.	• Please install the software again. • Please write the program again.
3494H	System	Axis Status Incorrect (When Axes Group is Valid)	MC_GroupEnable was executed when AxisName.Md.AxisStatus of configuration axis is other than Standstill or Disabled.	When Axes Group is Valid	The axes group status does not turn "GroupStandby".	Execute MC_GroupEnable after setting Standstill or Disabled in AxisName.Md.AxisStatus of all configuration axes.
3495H	System	Axes Group Status Incorrect (When Axes Group is Valid)	MC_GroupEnable was executed when AxesGroupName.Md.GroupStatus is other than GroupStandby or GroupDisabled.	When Axes Group is Valid	Not execute FB.	Execute MC_GroupEnable when AxesGroupName.Md.GroupStatus is GroupStandby or GroupDisabled.
3496H	System	Axes Group Configuration Axis Is in Use	MC_GroupEnable was executed for an axes group in which the AxisName.Md.UseInGroup of a configuration axis is TRUE (the configuration axis is in use in another axes group).	When Axes Group is Valid	The axes group status does not turn "GroupStandby"	Disable the axes group in use and execute MC_GroupEnable.
3497H	System	MCv_ChangeCycle Instruction Error	An error occurred during executing MCv_ChangeCycle for the following causes. • Cycle is out of range (Detail Code: 0001H) • ExecutionMode is out of range (Detail Code: 0002H) • Axis status at execution is incorrect (Detail Code: 0004H) • Other MCv_ChangeCycle was executed with the specified instanceID(Detail Code: 0008H) • Options is out of range (Detail Code: 0009H)	At start During operation	• Not execute FB. • The execution of FB is interrupted.	• Set Cycle within the range. • Set ExecutionMode within the range. • Execute it when the axis status is Standstill. • Execute other MCv_ChangeCycle after the current one is completed. • Set Options within the range.

Error code (Hexadecimal)	Error (System/ axis/axes group)	Error name	Error detail and cause	Detection timing (At start/ during control	Operation at error occurrence	Remedy
3498H	System	MCv_ChangeCycleTriggered Instruction Error	An error occurred during executing MCv_ChangeCycleTriggered for the following causes. • Cycle is out of range (Detail Code: 0001H) • ExecutionMode is out of range (Detail Code: 0002H) • Axis status at execution is incorrect (Detail Code: 0004H) • TriggerInput is incorrect (Detail Code: 0005H, 0006H) • Add-on SignalIO is not enabled. (Detail Code: 0007H) • Other MCv_ChangeCycleTriggered was executed with the specified instance ID(Detail Code: 0008H) • Options is out of range (Detail Code: 0009H)	At start During operation	• Not execute FB. • The execution of FB is interrupted.	• Set Cycle within the range. • Set ExecutionMode within the range. • Execute it when the axis status is Standstill. • Please check the setting of TriggerInput. • Please enable the add-on SignalIO. • Execute other MCv_ChangeCycleTriggered after the current one is completed. • Set Options within the range.
3499H	System	Add-on Memory Shortage	System memory for the related add-ons is insufficient.	At start up	Not execute FB.	Please increase System.PrConst.Addon_*.RamSizeMax of related add-ons.
349AH	System	Memory Shortage	The memory capacity in which MCFB instance is created is insufficient.	When executing dedicated instruction	Dedicated instruction is completed with the error.	Check setting data.
349BH	System	Motion Command Execution at STOP	Motion Command Execution was executed when it is in STOP status.	When executing dedicated instruction	Dedicated instruction is completed with the error.	Set module to RUN status.
349CH	System	Out of Velocity Override Coefficient (VelFactor) Range	The value is out of range for VelFactor.	During control	Continues the operation with the previous value.	Set VelFactor within the range.
349DH	System	Out of Acceleration Override Coefficient (AccFactor) Range	The value is out of range for AccFactor.	During control	Continues the operation with the previous value.	Set AccFactor within the range.
349EH	System	Out of Jerk Override Coefficient (JerkFactor) Range	The value is out of range for JerkFactor.	During control	Continues the operation with the previous value.	Set JerkFactor within the range
349FH	System	Latch Data Storage Shortage	• The number of elements specified for OutputBuffer is less than RecordCount. • Access to the outside of device or label range was made.	When executing FB During operation	Not execute FB.	• Set the number of elements specified for OutputBuffer to be more than RecordCount. • Make settings to avoid an access to the outside of the range.
34A0H	System	Out of Backlash Compensation Direction Range	The value is out of range for BacklashDirection.	When executing FB	Not execute FB.	Set the value within the range.
34A1H	System	Parameter Reading And Writing FB Execution Disabled Error	Unable to execute the parameter reading FB or the parameter writing FB.	When executing FB	Not execute FB.	• Check the setting data. • Check the execution count of transient transmission.
34A2H	System	Out of Parameter No. Range	The value is out of range for ParameterNumber.	When executing FB	Not execute FB.	Set the value within the range.
34A3H	System	Out of Option Range	The value is out of range for Options.	When executing FB	Not execute FB	Set the value within the range.

Error code (Hexadecimal)	Error (System/axis/axes group)	Error name	Error detail and cause	Detection timing (At start/during control	Operation at error occurrence	Remedy
34A4H	System	Touch Probe Disabling Unsupported	FB which does not support the touch probe disabling is set.	When executing FB	Not execute FB.	Set FB which supports the touch probe disabling.
34A5H	System	Signal Target Incorrect	Incorrect string is set for the Target of the signal.	When executing FB	Not execute FB.	Set a valid string in the signal Target.
34A6H	System	Out of Detection Setting Range of Signal	The value is out of range for Detection.	When executing FB	Not execute FB.	Set a value within the range for Detection.
34A7H	System	Out of CompensationTime Setting Range of Signal	The value is out of range for CompensationTime.	When executing FB	Not execute FB.	Set a value within the range for CompensationTime.
34A8H	System	Out of FilterTime Setting Range of Signal	The value is out of range for FilterTime.	When executing FB	Not execute FB.	Set FilterTime within the range.
34F8H	System	Device/Label Specification Incorrect	The device or label that cannot be used in the instruction was specified.	When executing instruction	Program stops/keeps Follow the setting below. Operation setting at error: Invalid device/label/buffer memory specification	• Please check and correct error occurrence points. • Please check the public label setting.
34F9H	System	FB/FUN Callee Incorrect	The function block (FB) or function (FUN) specified in the program does not exist.	When executing instruction	The program stops.	• Please check the error occurrence points and correct it. • Please write the program again.
34FAH	System	Label Temporary Area Incorrect	The label temporary area was used incorrectly.	When executing instruction	The program stops.	Please write the program again.
34FBH	System	DI Instruction Nesting Number Error	The number of nesting levels in the DI instruction exceeded its limit (16).	When executing instruction	The program stops.	Please change the number of nesting levels (16 or less).
34FCH	System	FB/FUN Call Nesting Number Error	The number of nesting levels in the function block (FB) or function (FUN) exceeded its limit (32).	When executing instruction	The program stops.	Please change the number of nesting levels (32 or less).
34FDH	System	Pointer execution error	• The target pointer specified in the instruction does not exist. • Exceeded the usable range when the array elements was specified dynamically.	When executing instruction	The program stops.	• Please check the error occurrence points and correct it. • Please write the program again.
34FEH	System	Label Temporary Area Exceeded	The secured label temporary area size exceeded its limit.	When executing instruction	The program stops.	• Please increase the capacity of label area. • Please write the program again.
34FFH	System	Operation Error	Division where the divisor is zero was performed.	When executing instruction	Program stops/continues Follow the setting below. Operation setting at error: Operation error	Please check and correct error occurrence points.
3500H	System	Operation Error	Incorrect data that cannot be converted by using the data conversion instruction was input.	When executing instruction	Program stops/continues Follow the setting below. Operation setting at error: Operation error	Please check and correct error occurrence points.
3501H	System	Operation Error	The operation was performed with the invalid data (denormalized number, NaN (not a number), or ±∞).	When executing instruction	Program stops/continues Follow the setting below. Operation setting at error: Operation error	Please check and correct error occurrence points.
3502H	System	Operation Error	Overflow occurred during calculation.	When executing instruction	Program stops/continues Follow the setting below. Operation setting at error: Operation error	Please check and correct error occurrence points.
3503H	System	File Incorrect	The program file is incorrect. Or, not correctly write the program file.	When executing instruction	Interrupts importing the files.	Please write the program again.

Error code (Hexadecimal)	Error (System/axis/axes group)	Error name	Error detail and cause	Detection timing (At start/during control	Operation at error occurrence	Remedy
3504H	System	Insufficient Memory Capacity	Failed to execute PSCAN/PSTOP instruction because the execution count of PSCAN/PSTOP instruction in 1 scan is too many.	When executing instruction	The program stops.	Please correct it so that it does not execute unnecessary PSCAN/PSTOP instructions after checking the error occurrence points.
3505H	System	Memory Error	The memory jump destination is out of the range for memory assignment.	When executing instruction	The program stops.	Please write the program again.
3506H	System	Device/Label Specification Incorrect	• The device or label specified in the instruction exceeded the usable range. • Insufficient number of array elements.	When executing instruction	Program stops/continues Follow the setting below. Operation setting at error: Invalid device/label/buffer memory specification	Please check and correct error occurrence points.
3507H	System	Operation Error	The data out of the specified range was input.	When executing instruction	Stops/continues Follow the setting below. Operation setting at error: Operation error	Please check and correct error occurrence points.
3508H	System	Operation Error	The operation result is out of the output range. (When the result of the instruction that concatenate character strings exceeded the allowable number of characters)	When executing instruction	Stops/continues Follow the setting below. Operation setting at error: Operation error	Please check and correct error occurrence points.
3509H	System	Program Name Specification Incorrect	The program specified in the instruction is not registered to the program setting.	During END processing	Stops/continues Follow the setting below. Operation setting at error: File name specification	Modify it to a correct program name.
350AH	System	Program Error	The program includes any instruction that cannot be used or decoded.	At ready ON	The program stops.	Please check the corresponding program, and correct it.
350BH	System	END Instruction Error	The END (FEND) instruction does not exist in the program.	At ready ON	The program stops.	• Please check the corresponding program, and correct it. • Please write the program and FB program again.
350CH	System	Program Execution Error	• After the global label setting file was modified, only the global label setting was written. Or, after the global label setting was modified, only the program and FB/FUN were written. (The global label setting file was not written.) • After the FB/FUN was modified, only the FB/FUN was written. Or, after the FB/FUN was modified, only the program and global label setting were written. (The FB/FUN was not written.) • After the buffer memory refresh setting of the global label setting was modified, only the program and FB/FUN were written to PLC. (The global label setting file was not written.)	At ready ON	The program stops.	Please write the program, FB/FUN and global label setting.
350DH	System	Insufficient Memory Capacity	Insufficient capacity of program storage target.	At ready ON	The program stops.	Please check the memory capacity of ST add-on.

Error code (Hexadecimal)	Error (System/ axis/axes group)	Error name	Error detail and cause	Detection timing (At start/ during control	Operation at error occurrence	Remedy
350EH	System	FB FUN Program Error	FB/FUN program configuration is incorrect.	At ready ON	The program stops.	• Please check the corresponding file, and correct it. • Please write the program and FB/FUN again.
350FH	System	Operation Error	The link direct device or buffer memory access device is specified for both (s) and (d) used in the BMOV instruction.	At ready ON	The program stops.	Please check the BMOV instruction of corresponding file, and correct it.
3510H	System	Device/Label Specification Incorrect	• The device or label specified in the instruction exceeded the usable range. • Array was not selected.	At ready ON	The program stops.	Please check the corresponding file, and correct it.
3511H	System	FB FUN Execution Error	Before the FB/FUN program ends, the call source program ended.	At ready ON	The program stops.	• Please check the corresponding file, and correct it. • Please write the program and FB/FUN again.
3512H	System	Parameter Error	The fixed scan program interval exceeded 16 kinds.	At ready ON	The program stops.	Please set the fixed scan program interval within 16 kinds.
3513H	System	Parameter Error	• An incorrect execution type has been specified. • An incorrect execution condition has been specified.	At ready ON	The program stops.	• Please change to a correct execution type. • Please check the fixed scan interval.
3514H	System	Program Error	The execution type that cannot be changed has been specified.	During END processing	The program stops.	Please specify the execution type that can be changed.
3515H	System	MCFB Specification Incorrect	The undefined MCFB exists.	At ready ON	The program stops.	• Please write the program again. • Please check if the necessary add-on has been installed in MCFB execution.
3516H	System	MCFB Execution Error	Unable to execute MCFB.	When executing instruction	The program stops.	• Please check if the label area which is out of range has not been written. • Please write the program again.
3517H	System	ST Add-on System Memory Shortage	System memory for the add-on is insufficient.	At ready ON	The program stops.	Please check the System.PrConst.Addon_Program_ST.RamSizeMax.
3518H	System	ST Add-on Internal Error	An error occurred in ST add-on.	At ready ON	The program stops.	• Please install the software again. • Please write the program again.

Error code (Hexadecimal)	Error (System/ axis/axes group)	Error name	Error detail and cause	Detection timing (At start/ during control	Operation at error occurrence	Remedy
3C00H	System	Hardware Error	An error was detected in the hardware.	During operation	Stops the system.	• Please take measures to reduce noise. • Please reset the CPU module, and run it again. If the same error occurs again even after taking the above, the possible cause is a hardware failure of the CPU module. Please consult your local Mitsubishi representative.
3C01H	System	Hardware Error	An error was detected in the hardware.	During operation	Stops the system.	• Please take measures to reduce noise. • Please reset the CPU module, and run it again. If the same error occurs again even after taking the above, the possible cause is a hardware failure of the CPU module. Please consult your local Mitsubishi representative.
3C02H	System	Hardware Error	An error was detected in the hardware.	During operation	Stops the system.	• Please take measures to reduce noise. • Please reset the CPU module, and run it again. If the same error occurs again even after taking the above, the possible cause is a hardware failure of the CPU module. Please consult your local Mitsubishi representative.
3C03H	System	Hardware Error	An error was detected in the hardware.	During operation	Stops the system.	• Please take measures to reduce noise. • Please reset the CPU module, and run it again. If the same error occurs again even after taking the above, the possible cause is a hardware failure of the CPU module. Please consult your local Mitsubishi representative.

Error code (Hexadecimal)	Error (System/ axis/axes group)	Error name	Error detail and cause	Detection timing (At start/ during control	Operation at error occurrence	Remedy
3C0FH	System	Hardware Error	An error was detected in the hardware.	During operation	Stops the system.	• Please take measures to reduce noise. • Please reset the CPU module, and run it again. If the same error occurs again even after taking the above, the possible cause is a hardware failure of the CPU module. Please consult your local Mitsubishi representative.
3C10H	System	Hardware Error	An error was detected in the hardware.	During operation	Stops the system.	• Please take measures to reduce noise. • Please reset the CPU module, and run it again. If the same error occurs again even after taking the above, the possible cause is a hardware failure of the CPU module. Please consult your local Mitsubishi representative.
3C11H	System	Hardware Error	An error was detected in the hardware.	During operation	Stops the system.	• Please take measures to reduce noise. • Please reset the CPU module, and run it again. If the same error occurs again even after taking the above, the possible cause is a hardware failure of the CPU module. Please consult your local Mitsubishi representative.
3C14H	System	Hardware Error	An error was detected in the hardware.	During operation	Stops the system.	• Please take measures to reduce noise. • Please reset the CPU module, and run it again. If the same error occurs again even after taking the above, the possible cause is a hardware failure of the CPU module. Please consult your local Mitsubishi representative.

Error code (Hexadecimal)	Error (System/ axis/axes group)	Error name	Error detail and cause	Detection timing (At start/ during control	Operation at error occurrence	Remedy
3C2FH	System	Memory Error	An error was detected in the memory.	During operation	Stops the system.	• Please take measures to reduce noise. • Please format the memory. And, please write all files, reset the CPU module, and run it again. If the same error occurs again even after taking the above, the possible cause is a hardware failure of the CPU module. Please consult your local Mitsubishi representative.
3F00H	System	Firmware Version Unsupported	The firmware version is not supported by the CPU module.	At power ON	Stops the system.	Please use a CPU module which supports Motion modules.

22.6 List of Logging Error Codes

Error code (Hexadecimal)	Cause	Remedy
4008H	The system is in the BUSY state.	Execute it again after a certain period of time has elapsed.
4030H	Data logging was started in the following status. • Data logging with an invalid device name was specified.	Check the specified device name.
4053H	An error occurred during writing data to the specified drive.	• Check the specified drive, and then write data again. • Replace the target drive, and then write data again.
4080H	An error in the request/setting data.	Check the specified data and write it again.
40C0H	Specified label name does not exist.	• Check the label name. • Execute again after the module RUN when using local labels.
40C1H	Failed to access in label name because specified array number is larger than array size.	Reduce value of specified array, and specify within array size.
41CCH	• Started the data logging in the status that sub folder for storing the file (folder) to be used in each function cannot be created or accessed. • Unable to create and access file (folder) used in each function during data saving or data logging execution.	Execute it again after checking the file name and sub folder name.
41CDH	• Registered in the status that file (folder) cannot be created or accessed because any of the existing files (folders) has the same name as the file (folder) to be used in each function. • Unable to create and access file (folder) during data saving or data logging execution.	• Do not access the specified file or folder. • Execute it again after checking the file or folder.
41CEH	The specified file is read only and write protected.	The specified file is write protected. Check the attributes and then execute again.
41D0H	No free space in the specified drive, or the number of files in the directory of the specified drive has exceeded the maximum limit.	• Increase the free space in the drive and then execute it again. • Delete the files in the drive and then execute it again.
41D2H	It is unsupported function.	Check if the related add-ons are installed. • The SignalIO add-on is required to use devices or labels. • The MotionEventHist add-on is required to use "Event History" for the trigger condition.
41F4H	Unable to execute request content since operation protected in the system is executed.	Enable auto-logging.
4262H	The number of data points exceeds the maximum (1024 points).	Make the number of data points per setting less than 1024.
4277H	Data logging was started in the following status. • The number of saved files exceeding the set number while the "operation for exceeding file number" was set to "Stop". • The number of files exceeding the number of saved files while the "operation for exceeding file number" was set to "Overwrite".	• Execute it again after deleting the files in the memory for saving the data logging results. • Execute it again after changing the save destination.
4283H	Registered when the number of records exceeded the number of collectable records at the set internal buffer capacity.	• Correct the internal buffer capacity setting. • Reduce the number of records.
4288H	Unable to execute the request because the specified file name has exceeded the maximum number of characters.	Recheck the path name and execute it again.
4289H	An item that cannot be set in data sample condition is being set.	Check the setting of data sampled condition.
4293H	Unable to execute because the total of internal buffer has been exceeded the maximum capacity.	Recheck the total of internal buffer capacity setting and execute it again.
4C00H	There is not enough free space on target memory for result file creation.	Increase free space on target memory, then execute it again.
4C06H	System Error	Confirm specified data and write again.
4C0DH	Tried to start data logging when result file is being transferring.	Start data logging again after data logging file transfer is completed.
4C0FH	The sequence program file or global label setting file was changed when the function for which the label (global label, local label) had been specified as "Sample", "Data" or "Trigger condition" is being executed (the status is "RUN waiting (no collection)", "condition waiting (no collection)", "start waiting (no collection)", "pause", "collecting", "trigger waiting (collecting before trigger)", "collecting after trigger", "saving").	Execute it again.
4C56H	• Failed to read (access) file. • The specified file does not exist. Or the specified sub directory does not exist.	• Check if the file is deleted or not. • Check the file name or sub directory name and then execute it again.
4F01H	An error in the data type.	Recheck the data to be collected.

22.7 List of Event Codes

This section lists the events which occur in CC-Link IE TSN.
System, security, and operation are included in the event types.

System

Event code	Overview	Cause
07ED	Velocity Override '0'	'0' was set to velocity override.
07EE	Servo System Recorder Start	Servo system recorder is started.
07EF	Follow-up Temporarily Invalid Cancellation	Follow-up invalid has been cancelled.
07F0	MCFB Start (Control System)	MCFB was started.
07F1	MCFB Start (Operation System)	MCFB start or continuous start was executed.
07F2	FB Start	FB was started.
07F3	MCFB Stop	MCFB was stopped by stop cause.
07F4	File Transmission Start	File transmission was started by script file/label.
07F5	Start when SW Stroke Limit is Disabled	Start when software stroke limit is invalid by AxisName.Cd.SwStrokeLimit_Override.
07F6	Start when HW Stroke Limit is Invalid	Start when hardware stroke limit is invalid by AxisName.Cd.HwStrokeLimit_Override.
07F7	Forced Stop Detection	Forced stop signal is detection status.
07F8	Event History File Creation	Event history file was created.
07F9	Homing Request Clear	Homing request clear is ON.
07FA	Current Position Change	Current position change was executed.
07FB	Follow-up Temporary Invalid	Follow-up was switched to invalid.
07FC	Servo ON when Follow-up is Invalid	Change it to servo ON from follow-up disabled status of servo OFF.
07FD	Current Position Restoration Completion	Current position restoration was completed.
07FE	Homing Request OFF to ON	Homing request is ON. • Absolute position erase at driver side was detected. [Real drive axis only] (Detail Code: 0000H) • Absolute position data error in Motion module was detected. (Detail Code: 0001H) • Axis type was changed. (Detail Code: 0002H) • 'Polarity(607EH)' b7: position polarity of driver object was changed. (Detail Code: 0003H) • Change of driver or motor encoder was detected (Detail Code: 0004H) • Machine homing was started. (Detail Code: 0005H) • Driver unit conversion (numerator/denominator) of axis was changed. (Detail Code: 0006H) • Machine homing has never been executed after system started. (Detail Code: 0007H) • Current position restoration was executed. (Detail Code: 0007H) • A driver other than the one used at last home position establishment was connected for a virtual axis. (Detail Code: 0008H) • The upper/lower limit value of encoder ring counter was changed. (Detail Code: 0009H)) • Driver object 'HomeOffset(607CH)' was changed. (Detail Code: 000AH)

Security

Event code	Overview	Cause
10100	Security key Registration/Deletion	A security key was registered or deleted.
10200	Remote Password Lock	The lock processing for the remote password was performed.
10201	Remote Password Unlock Successful	The unlock processing for the remote password was completed successfully.
10202	Remote Password Unlock Failed	The unlock processing for the remote password was not completed successfully.
10300	Access from IP restricted with IP filter setting	It was accessed from IP address restricted with the IP filter setting.
10400	File Password Registration/Change/ Deletion Successful	Registration, change, or deletion of a file password was performed and completed successfully.
10401	File Password Registration/Change/ Deletion Failed	Registration, change, or deletion of a file password was performed and was not completed successfully.
10402	File Password Clear Successful	Clear of a file password was performed and completed successfully.
10403	File Password Clear Failed	Clear of a file password was performed and was not completed successfully.

Operation

Event code	Overview	Cause
20010	Module Data Initialization	The parameter data initialization was executed by the user.
20011	Module Data Backup	The executed data backup was executed by the user.
20020	F/W Update	The F/W update was executed by the user.
20030	Online Module Change	The online module change was executed by the user.
20050	Servo Transient Transmission	An object was written with servo transient transmission.
27FFE	Execute Soft Reboot	Soft reboot was executed.
27FFF	Clear Event History	Clear Event History was executed.

PART 4 LISTS

23 List of Data Type

24 List of Variables

25 List of Supported Axis Type

26 List of FBs

27 List of Add-on Library

28 List of Boot software

23 List of Data Type

This section describes the following data types.

- Type
- ENUM enumerator

Type

Data type label	Description
Bit (BOOL)	• Bit type label • Bit-specified word [unsigned]/bit string [16 bits] type label • Bit-specified word [signed] type label • Timer/retentive timer/long timer/long retentive timer type label contact/coil • Counter/long counter type label contact/coil
Word [unsigned] / Bit column [16 bits] (WORD)	• Word [unsigned] / Bit column [16 bits] type label • Current value of timer/retentive timer type label • Current value of counter type label
Double word [unsigned] / Bit column [32 bits] (DWORD)	• Double word [unsigned]/bitstring [32 bits] type label • Current value of long timer / long retentive timer type label • Current value of long counter type label
Word [signed] (INT)	• Word [signed] type label • Current value of timer / retentive timer type label • Current value of counter type label
Double word [signed] (DINT)	• Double word [signed] type label • Current value of long timer / long retentive timer type label • Current value of long counter type label
Single-precision real number (REAL)	Single-precision real type label
Double-precision real number (LREAL)	Double-precision real type label
Time (TIME)	Time type label
Character string (STRING(??))*1	Character string type label
Character string [Unicode] (WSTRING(??))*1	Character string [Unicode] type label

*1 (??) is a value excluding Null.

For details, refer to "Data Specification Method" in the following manual.

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

ENUM enumerator

The type of each ENUM enumerator elements are Word [signed].

Type name/Enumerator	Setting value	Description
MC_BUFFER_MODE		
mcAborting	0	Aborting
mcBuffered	1	Buffered
mcBlendingLow	2	BlendingLow
mcBlendingPrevious	3	BlendingPrevious
mcBlendingNext	4	BlendingNext
mcBlendingHigh	5	BlendingHigh
MC_EXECUTION_MODE		
mcImmediately	0	Execute immediately
mcQueued	1	Execute at completion
mcNextExecute	2	Execute at next start
mcSpeculatively	3	Execute speculatively
MC_COMBINE_MODE		
mcAddAxes	0	Add positions of 2 input axes

Type name/Enumerator	Setting value	Description
mcSubAxes	1	Subtract positions of 2 input axes
MC_DIRECTION		
mcPositiveDirection	1	Positive direction
mcNegativeDirection	2	Negative direction
mcShortestWay	3	Shortest path
mcCurrentDirection	4	Current direction
MC_SOURCE		
mcSetValue	1	Set value
mcActualValue	2	Actual value
mcLatestSetValue	101	Latest set value
mcLatestActualValue	102	Latest actual value
MC_CIRC_MODE		
mcBorder	0	Border point designation
mcCenter	1	Center point designation
mcRadius	2	Radius designation
MC_CIRC_PATHCHOICE		
mcCW	0	CW
mcCCW	1	CCW
mcShortWay	2	Short way
mcLongWay	3	Long way
mcCWLongWay	4	CW long way
mcCCWLongWay	5	CCW long way
MC_START_MODE		
mcImmediate	0	Immediately
mcAbsolute	1	Absolute
mcRelative	2	Relative
MC_AXIS_STATUS		
Invalid	-1	Axis variable uninitialization/axis parameter error
Disabled	0	Axis disabled
ErrorStop	1	Error stop
Stopping	2	Deceleration stop
Homing	3	Homing
Standstill	4	Standby
DiscreteMotion	5	Positioning operation
ContinuousMotion	6	Continuous operation
SynchronizedMotion	7	Synchronous operation
MC_AXES_GROUP_STATUS		
Invalid	-1	Axes group variable not initialized/axes group parameter error
GroupDisabled	0	Axes group disabled
GroupErrorStop	1	Error stop
GroupStopping	2	Deceleration stop
GroupHoming	3	Reserved
GroupStandby	4	Standby
GroupMoving	5	Operation
MC_DRIVE_MODE		
NoModeChange	0	Not changed
pp	1	Reserved
vl	2	Reserved
pv	3	Reserved
tq	4	Reserved
Reserved	5	Reserved
hm	6	hm (Homing)
ip	7	Reserved

Type name/Enumerator	Setting value	Description
csp	8	csp (Positioning control)
csv	9	csv (Velocity control)
cst	10	cst (Torque Control)
cstca	11	Reserved
ct	-104	ct (Continuous operation to torque Control)
MC_VELOCITY_LIMIT_MODE		
Ignore	0	Ignore
ClampWithRamp	1	Clamp
Truncate	2	Truncate
ImmediateStop	3	Immediate stop
ClampWithoutRamp	4	Clamp (Without Ramp at Deceleration)
MC_INTERPOLATE_SPEED_MODE		
VectorSpeed	0	Vector speed
LongAxisSpeed	1	Long axis speed
ReferenceAxisSpeed	2	Reference axis speed
MC_SYNTHESIZING_MODE		
Addition	0	Addition
Substitution	1	Substitution
Combine	2	Combine
Interpolation	3	Interpolation
MC_PROFILE_COMBINE_MODE		
Relative	0	Relative
InputRelative	1	Input relative
OutputRelative	2	Output relative
Absolute	3	Absolute
MC_RECORD_MODE		
OneShot	0	One shot mode
RecordCount	1	Number of designation times mode
RingBuffer	2	Ring buffer mode
MC_CLUTCH_MODE		
Exponent	0	Exponent
Linear	1	Linear
LinearInputFollow	2	Linear (Input follow-up)
MC_CAM_CURVE_TYPE		
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 (Adj.)
MC_AXIS_TYPE		
DriveAxis	0	Real drive axis
EncoderAxis	2	Real encoder axis
VirtualDriveAxis	3	Virtual drive axis
VirtualEncoderAxis	4	Virtual encoder axis
VirtualLinkAxis	5	Virtual linked axis

Type name/Enumerator	Setting value	Description
MC_ENCODER_AXIS_TYPE		
IoModule	0	Via I/O device
Drive	1	Via drive module
MC_ABS_SYSTEM		
ABSDisabled	0	Not use the absolute position control setting
Enabled	1	Use the absolute position control setting
Auto	-1	Automatic setting (Acquire from connected device)
MC_DRIVE_STATE		
NotReadyToSwitchOn	0	Not ready to switch on
Fault	1	Fault
FaultReactionActive	2	Fault reaction active
SwitchOnDisabled	3	Switch on disabled
ReadyToSwitchOn	4	Ready To SwitchOn
SwitchedOn	5	Switched On
OperationEnable	6	Operation enable
QuickStopActive	7	Quick stop active
Invalid	-1	Invalid
MC_DECEL_STOP_MODE		
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		
RapidCurve	0	Recreate deceleration curve at increased deceleration
OverrideCurve	1	Recreate deceleration curve
ContinueCurve	2	Maintain deceleration curve
MC_POS_SOURCE		
Invalid	-1	Invalid
SetPosition	1	Set position
CumulativePosition	2	Cumulative current position
FeedMachinePosition	3	Feed machine position
CommandedPosition	4	Commanded position
ActualPosition	5	Actual position
MC_ACC_ZERO_MODE		
ACCError	-1	Error (Not started)
MaximumAcceleration	1	Maximum acceleration/deceleration
MC_SIGNAL_LOGIC		
HighLevel	0	Detection at TRUE
LowLevel	1	Detection at FALSE
RisingEdge	2	Detection at FALSE → TRUE (leading edge)
FallingEdge	3	Detection at TRUE → FALSE (trailing edge)
BothEdges	4	Detection at leading edge/trailing edge
MC_STATION_REFRESH_MODE		
EmphasisResponse	0	Response preferred method
EmphasisOperationCycle	1	Operation cycle priority method
MC_ERROR_CLASS		
_None	0	None
Warning	1	Warning
MinorError	2	Minor error
ModerateError	3	Moderate error
MajorError	4	Major error

Type name/Enumerator	Setting value	Description
MC_LOGGING_TYPE		
_None	0	None
ContinuousLogging	1	Continuous logging
TriggerLogging	2	Trigger logging
MC_LOGGING_SAVE_STATUS		
UnSave	0	Unsaved
Saving	1	Saving
Saved	2	Saved
FullySaved	3	Fully saved
MC_EXECUTION_STATE		
Ready	0	Waiting for the request
Executing	1	Executing
Done	2	Execution completion
Error_	-1	Error occurrence
MC_ACTIVATION_CMD		
_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		
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		
_None	0	None
Regist	1	Register
TemporaryNoLimitation	10	Temporarily release
Initialize	-1	Initialize
MC_ADDON_LIMIT_STATE		
NoLimitation	0	Not restricted
Limitation	1	Limited
TemporaryNoLimitation	10	Released
Lockout	-1	Lockout
MC_SWITCHING_REQUEST		
NoRequest	0	No request
Enable_	1	Enable
Disable	2	Disable
MC_POS_RESTORATION_STATUS		
NotExecute	0	Not executed
WaitingRequest	1	Waiting restoration request
RestoredInIncSystem	2	Restoration completed by incremental system
RestoredInAbsSystemUnHomed	3	Restored in absolute position system (Homing uncompleted)
RestoredInAbsSystem	4	Restored in absolute position system
MC_TRANSITION_MODE		
TMNone	0	Standard

24 List of Variables

This section contains the following information.

- Axis variables
- Axes group variables
- System variables
- Other structure variables

For the setting method of variables, refer to Help of the engineering tool.

Guide for reference of the variables list is shown below.

- Import

Item	Description	Details
Operation cycle	The timing for executing each operation cycle.	☞ Page 110 Operation cycle settings
Normal	The timing for executing normal task of the motion system.	Refer to "Execution of programs" in the following manual. MELSEC iQ-R Programming Manual (Motion Control Function Blocks)
Logging setting cycle	The timing for correcting the logging target data.	☞ Page 536 Logging setting (LOGGING)
Service	The timing for executing the motion service processing control.	☞ Page 587 Motion service processing control
Immediately	—	—
At start	At FB start	—
At stop	At FB stop	—
At system start	At motion system start.	—
When connecting device	At a slave device connection	—
Ready ON	At the PLC READY [Y0] ON	☞ Page 591 I/O signals
System basic cycle	The timing for executing the system basic cycle processing.	☞ Page 109 System basic cycle
Operation cycle(GroupEnable)	The timing for executing each operation cycle (GroupEnable)	☞ Page 110 Operation cycle settings
When axes group is valid	The timing when the axes group becomes valid.	☞ Page 73 Axes group enabled

- Type

For details on types, refer to the following.

☞ Page 696 Type

- Attribute

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)
NOLIST_SYSTEM	Non-label list (Base generation)

- Parameter type

Display	Description
ID_PRCST	Parameter constant (WRITE(⇨ Motion))
ID_PR	Parameter (WRITE(⇨ Motion))
ID_MD	Monitor data (READ(Motion ⇨))
ID_CD	Command data (WRITE(⇨ Motion))
ID_MDCST	Monitor data (Initial value) (READ(Motion ⇨))

Precautions

- The refresh direction describes labels created automatically by engineering tool.
  For example, if a user creates AXIS_REAL type labels manually, the direction can be specified in instance units. When specifying Write, the attribute for all members will be Write. When specifying Read, members with the attribute Write maintains Write.
- The labels with blank parameter type follow the parameter type of the parent member.

Axis variables

Variable/Structure name	Name	Import	Type	Attribute	Parameter type
AxisName.AxisRef.					
AxisNo	Axis No.	—	WORD(UINT)	LIST_WRITE_BACK	ID_MDCST
StartIO	I/O No. *1	—	WORD(HEX)	LIST_WRITE_BACK	ID_MDCST
AxisName.PrConst.					
AddressOfStation	Station address setting	At system start	WSTRING(63)	LIST_WRITE_BACK	ID_PRCST
AxisType	Axis type setting	At system start	ENUM(MC_AXIS_TYPE)	LIST_WRITE_BACK	ID_PRCST
Encoder_AxisType	Real encoder axis type setting	At system start	ENUM(MC_ENCODER_AXIS_TYPE)	LIST_WRITE_BACK	ID_PRCST
Encoder_CounterDisableSignal	Counter disabling signal	At system start	STRUCT(SIGNAL_SELECT)	LIST_WRITE_BACK	ID_PRCST
Encoder_RingCout_LowerValue	Encoder ring counter lower limit value	At system start	DINT	LIST_WRITE_BACK	ID_PRCST
Encoder_RingCout_UpperValue	Encoder ring counter upper limit value	At system start	DINT	LIST_WRITE_BACK	ID_PRCST
FastOperationMode	High-speed mode setting	At system start	WORD(HEX)	LIST_WRITE_BACK	ID_PRCST
HwStrokeLimit_FlsSignal	Upper limit signal	At system start	STRUCT(SIGNAL_SELECT)	LIST_WRITE_BACK	ID_PRCST
HwStrokeLimit_RlsSignal	Lower limit signal	At system start	STRUCT(SIGNAL_SELECT)	LIST_WRITE_BACK	ID_PRCST
OperationCycle	Control cycle setting	At system start	INT	LIST_WRITE_BACK	ID_PRCST
PositioningRange	Positioning range	At system start	LREAL	LIST_WRITE_BACK	ID_PRCST
PosRestoration_AbsPosBase	Absolute position reference setting	At system start	ENUM(MC_POS_SOURCE)	LIST_WRITE_BACK	ID_PRCST
PosRestoration_AbsPosEnable	Absolute position management setting	At system start	ENUM(MC_ABS_SYSTEM)	LIST_WRITE_BACK	ID_PRCST
RingCount_Enable	Ring counter enabled selection	At system start	BOOL	LIST_WRITE_BACK	ID_PRCST
RingCount_LowerValue	Ring counter lower limit value	At system start	LREAL	LIST_WRITE_BACK	ID_PRCST
RingCount_UpperValue	Ring counter upper limit value	At system start	LREAL	LIST_WRITE_BACK	ID_PRCST
SlaveEmulate_Enable	Slave emulation enabled	At system start	BOOL	LIST_WRITE_BACK	ID_PRCST
SlaveObject	Slave object data	At system start	STRUCT(SLAVE_OBJECT_REAL)	LIST_WRITE_BACK	ID_PRCST
SlaveObject	Slave object data	At system start	STRUCT(SLAVE_OBJECT_ENCODER)	LIST_WRITE_BACK	ID_PRCST
SlaveObject	Slave object data	At system start	STRUCT(SLAVE_OBJECT_VIRTUAL_ENCODER)	LIST_WRITE_BACK	ID_PRCST
TorqueLimit_Max	Torque limit maximum value	At system start	LREAL	LIST_WRITE_BACK	ID_PRCST
TorqueLimit_NegativeInitial	Negative direction torque limit initial value	At system start	LREAL	LIST_WRITE_BACK	ID_PRCST
TorqueLimit_PositiveInitial	Positive direction torque limit initial value	At system start	LREAL	LIST_WRITE_BACK	ID_PRCST
AxisName.Pr.					
AccelerationLimit	Acceleration limit value	Ready ON	LREAL	LIST_WRITE_BACK	ID_PR
AccelerationZeroBehavior	Operation selection at start acceleration/deceleration 0	Ready ON	ENUM(MC_ACC_ZERO_MODE)	LIST_WRITE_BACK	ID_PR
CmdInPos_Width	Command in-position width	Ready ON	LREAL	LIST_WRITE_BACK	ID_PR
DecelerationLimit	Deceleration limit value	Ready ON	LREAL	LIST_WRITE_BACK	ID_PR
Drive_UnitConvRatioNum	Driver unit conversion numerator	Ready ON	DWORD(UDINT)	LIST_WRITE_BACK	ID_PR
Drive_UnitConvRatioDen	Driver unit conversion denominator	Ready ON	DWORD(UDINT)	LIST_WRITE_BACK	ID_PR

Variable/Structure name	Name	Import	Type	Attribute	Parameter type
ForcedStop_Signal	Forced stop signal	Ready ON	STRUCT(SIGNAL_SELECT)	LIST_WRITE_BACK	ID_PR
Homing_Required	Homing required or not	Ready ON	BOOL	LIST_WRITE_BACK	ID_PR
JerkLimit	Jerk limit value	Ready ON	LREAL	LIST_WRITE_BACK	ID_PR
OverrunOperation	Operation setting at overrun	Ready ON	ENUM(MC_DECEL_STOP_MODE)	LIST_WRITE_BACK	ID_PR
StartableAtUnhomed	Start permission at home position return uncompleted	Ready ON	BOOL	LIST_WRITE_BACK	ID_PR
StopMode_Deceleration	Deceleration at stop	Ready ON	LREAL	LIST_WRITE_BACK	ID_PR
StopMode_DecelerationCurve	Stop selection at deceleration to stop	Ready ON	ENUM(MC_STOP_CURVE_MODE)	LIST_WRITE_BACK	ID_PR
StopMode_General	Stop selection at stop by factors	Ready ON	ENUM(MC_DECEL_STOP_MODE)	LIST_WRITE_BACK	ID_PR
StopMode_HwStrokeLimit	Stop selection at hardware stroke limit error occurrence	Ready ON	ENUM(MC_DECEL_STOP_MODE)	LIST_WRITE_BACK	ID_PR
StopMode_ServoOff	Process selection at servo OFF command during operation	Ready ON	ENUM(MC_DECEL_STOP_MODE)	LIST_WRITE_BACK	ID_PR
StopMode_SwStrokeLimit	Stop selection at software stroke limit error occurrence	Ready ON	ENUM(MC_DECEL_STOP_MODE)	LIST_WRITE_BACK	ID_PR
StopOption_DriverTargetIgnored	Driver Command Discard Detection Setting	Ready ON	BOOL	LIST_WRITE_BACK	ID_PR
StopSignal	Stop signal	Ready ON	STRUCT(SIGNAL_SELECT)	LIST_WRITE_BACK	ID_PR
SwStrokeLimit_Lower	Software stroke limit lower value	Ready ON	LREAL	LIST_WRITE_BACK	ID_PR
SwStrokeLimit_Target	Software stroke limit target	Ready ON	ENUM(MC_POS_SOURCE)	LIST_WRITE_BACK	ID_PR
SwStrokeLimit_Upper	Software stroke limit upper value	Ready ON	LREAL	LIST_WRITE_BACK	ID_PR
Unit_Position	Position command unit	Ready ON	DWORD(HEX)	LIST_WRITE_BACK	ID_PR
Unit_PositionString	Position command unit string	Ready ON	WSTRING(31)	LIST_WRITE_BACK	ID_PR
Unit_Velocity	Speed command unit	Ready ON	DWORD(HEX)	LIST_WRITE_BACK	ID_PR
VelocityLimit_Negative	Negative direction speed limit value	Ready ON	LREAL	LIST_WRITE_BACK	ID_PR
VelocityLimit_OverOperation	Operation setting at speed limit value exceeded	Ready ON	ENUM(MC_VELOCITY_LIMIT_MODE)	LIST_WRITE_BACK	ID_PR
VelocityLimit_Positive	Positive direction speed limit value	Ready ON	LREAL	LIST_WRITE_BACK	ID_PR
AxisName.Md.					
AccelerationLimit	Acceleration limit value	Ready ON	LREAL	LIST_READ_ONLY	ID_MD
AccelerationOverride	Acceleration override coefficient	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD
AccelerationZeroBehavior	Operation selection at start acceleration/deceleration 0	Ready ON	ENUM(MC_ACC_ZERO_MODE)	LIST_READ_ONLY	ID_MD
ActualPosition	Feedback position	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD
ActualVelocity	Feedback speed	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD
Analyzing	Analyzing	Operation cycle	BOOL	LIST_READ_ONLY	ID_MD
AutoDeceleration	Automatically decelerating	Operation cycle	BOOL	LIST_READ_ONLY	ID_MD
AxisName	Axis name	At system start	WSTRING(127)	LIST_READ_ONLY	ID_MD
AxisStatus	Axis status	Operation cycle	ENUM(MC_AXIS_STATUS)	LIST_READ_ONLY	ID_MD
BufferingFBs	Number of buffering FBs	At start/At stop	INT	LIST_READ_ONLY	ID_MD
CmdInPos	Command In-position	Operation cycle	BOOL	LIST_READ_ONLY	ID_MD
CmdInPos_Width	Command In-position Width	Ready ON	LREAL	LIST_READ_ONLY	ID_MD
CommandedAcceleration	Specified acceleration	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD
CommandedDeceleration	Specified deceleration	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD

Variable/Structure name	Name	Import	Type	Attribute	Parameter type
CommandedJerk	Specified jerk	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD
CommandedPosition	Specified position	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD
CommandedVelocity	Specified speed	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD
Cst_SetTorque	Command current torque at torque control	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD
Cst_TargetTorque	Target torque at torque control	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD
CumulativePosition	Cumulative current position	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD
DecelerationLimit	Deceleration limit value	Ready ON	LREAL	LIST_READ_ONLY	ID_MD
Drive_RPDO[1..64]	RPDO (Slave device → Motion system) Mapping	When connecting device	DWORD(HEX)	LIST_READ_ONLY	ID_MD
Drive_TPDO[1..64]	TPDO (Motion system → Slave device) Mapping	When connecting device	DWORD(HEX)	LIST_READ_ONLY	ID_MD
Drive_UnitConvRatioNum	Driver unit conversion numerator	Ready ON	DWORD(UDINT)	LIST_READ_ONLY	ID_MD
Drive_UnitConvRatioDen	Driver unit conversion denominator	Ready ON	DWORD(UDINT)	LIST_READ_ONLY	ID_MD
Driver_Mode	Driver control mode	Operation cycle	ENUM(MC_DRIVE_MODE)	LIST_READ_ONLY	ID_MD
Driver_ReadyOn	Driver ready on status	Operation cycle	BOOL	LIST_READ_ONLY	ID_MD
Driver_ServoOn	Driver servo on status	Operation cycle	BOOL	LIST_READ_ONLY	ID_MD
Driver_State	Driver status	Operation cycle	ENUM(MC_DRIVE_STATE)	LIST_READ_ONLY	ID_MD
DriverError	Drive module error detection	Immediately	BOOL	LIST_READ_ONLY	ID_MD
DriverErrorID	Drive module error code	Immediately	WORD(HEX)	LIST_READ_ONLY	ID_MD
DriverErrorDetailID	Drive module error detail code	Immediately	WORD(HEX)	LIST_READ_ONLY	ID_MD
Encoder_Connected	Connection status	Operation cycle	BOOL	LIST_READ_ONLY	ID_MD
Encoder_CounterDisable	Counter disabled	Operation cycle	BOOL	LIST_READ_ONLY	ID_MD
Error	Axis error detection	Immediately	BOOL	LIST_READ_ONLY	ID_MD
ErrorID	Axis error code	Immediately	WORD(HEX)	LIST_READ_ONLY	ID_MD
FeedMachinePosition	Feed function position	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD
FollowupDisable	Follow-up disabled	Operation cycle	BOOL	LIST_READ_ONLY	ID_MD
ForcedStop_Released	Forced stop cancelling	Operation cycle	BOOL	LIST_READ_ONLY	ID_MD
ForcedStop_Signal	Forced stop signal	Ready ON	STRUCT(SIGNAL_SELECT)	LIST_READ_ONLY	ID_MD
Homing_Complete	Homing completed	Immediately	BOOL	LIST_READ_ONLY	ID_MD
Homing_Request	Homing request	Immediately	BOOL	LIST_READ_ONLY	ID_MD
Homing_Required	Homing required or not	Ready ON	BOOL	LIST_READ_ONLY	ID_MD
Homing_Status	Homing operation status	Operation cycle	WORD(HEX)	LIST_READ_ONLY	ID_MD
HwStrokeLimit_FlsStatus	Upper limit signal status	Operation cycle	BOOL	LIST_READ_ONLY	ID_MD
HwStrokeLimit_Override	Hardware stroke limit override	At start	STRING(15)	LIST_READ_ONLY	ID_MD
HwStrokeLimit_RlsStatus	Lower limit signal status	Operation cycle	BOOL	LIST_READ_ONLY	ID_MD
InVelocity	Target speed reached	Operation cycle	BOOL	LIST_READ_ONLY	ID_MD
Io_PosActualValue	Object data_PosActualValue	Operation cycle	DINT	LIST_READ_ONLY	ID_MD
Io_PosEncoderResolution	Object data_PosEncoderResolution	Operation cycle	DWORD(UDINT)	LIST_READ_ONLY	ID_MD
Io_Statusword	Object data_Statueword	Operation cycle	WORD(HEX)	LIST_READ_ONLY	ID_MD
Io_TargetPos	Object data_TargetPos	Operation cycle	DINT	LIST_READ_ONLY	ID_MD
Io_TargetVelocity	Object data_TargetVelocity	Operation cycle	DWORD(UDINT)	LIST_READ_ONLY	ID_MD
Io_VelActualValue	Object data_VelActualValue	Operation cycle	DWORD(UDINT)	LIST_READ_ONLY	ID_MD
Io_TorqueActualValue	Object data_TorqueActualValue	Operation cycle	DWORD(UDINT)	LIST_READ_ONLY	ID_MD
JerkLimit	Jerk limit value	Ready ON	LREAL	LIST_READ_ONLY	ID_MD
OperationCycle	Control cycle	At system start	INT	LIST_READ_ONLY	ID_MD
OverrunOperation	Operation setting at overrun	Ready ON	ENUM(MC_DECEL_STOP_MODE)	LIST_READ_ONLY	ID_MD

Variable/Structure name	Name	Import	Type	Attribute	Parameter type
PosRestoration_Status	Current position restoration status	Service	ENUM(MC_POS_RESTORATION_STATUS)	LIST_READ_ONLY	ID_MD
ProfileID	Execution profile ID No.	Operation cycle	WORD(UINT)	LIST_READ_ONLY	ID_MD
SetAcceleration	Command current acceleration	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD
SetPosition	Command current position	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD
SetVelocity	Set velocity	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD
SlaveEmulate_Enable	Slave emulating	Service	BOOL	LIST_READ_ONLY	ID_MD
StartableAtUnhomed	Start permission at home position return uncompleted	Ready ON	BOOL	LIST_READ_ONLY	ID_MD
StopMode_Deceleration	Stop selection at deceleration to stop	Ready ON	LREAL	LIST_READ_ONLY	ID_MD
StopMode_DecelerationCurve	Stop selection at deceleration to stop	Ready ON	ENUM(MC_STOP_CURVE_MODE)	LIST_READ_ONLY	ID_MD
StopMode_General	Stop selection at stop by factors	Ready ON	ENUM(MC_DECEL_STOP_MODE)	LIST_READ_ONLY	ID_MD
StopMode_HwStrokeLimit	Stop selection at hardware stroke limit error occurrence	Ready ON	ENUM(MC_DECEL_STOP_MODE)	LIST_READ_ONLY	ID_MD
StopMode_ServoOff	Process selection at servo OFF command during operation	Ready ON	ENUM(MC_DECEL_STOP_MODE)	LIST_READ_ONLY	ID_MD
StopMode_SwStrokeLimit	Stop selection at software stroke limit error occurrence	Ready ON	ENUM(MC_DECEL_STOP_MODE)	LIST_READ_ONLY	ID_MD
StopOption_DriverTargetIgnored	Driver Command Discard Detection Setting	Ready ON	BOOL	LIST_READ_ONLY	ID_MD
StopSignal	Stop signal	Ready ON	STRUCT(SIGNAL_SELECT)	LIST_READ_ONLY	ID_MD
StopStatus	Stop status	Operation cycle	BOOL	LIST_READ_ONLY	ID_MD
SwStrokeLimit_Lower	Software stroke limit lower value	Ready ON	LREAL	LIST_READ_ONLY	ID_MD
SwStrokeLimit_Override	Software stroke limit override	At start	STRING(15)	LIST_READ_ONLY	ID_MD
SwStrokeLimit_Target	Software stroke limit target	Ready ON	ENUM(MC_POS_SOURCE)	LIST_READ_ONLY	ID_MD
SwStrokeLimit_Upper	Software stroke limit upper value	Ready ON	LREAL	LIST_READ_ONLY	ID_MD
TargetAcceleration	Target acceleration	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD
TargetDeceleration	Target deceleration	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD
TargetVelocity	Target speed	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD
TorqueLimit_Negative	Negative direction torque limit value	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD
TorqueLimit_Positive	Positive direction torque limit value	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD
Unit_Position	Position command unit	Ready ON	DWORD(HEX)	LIST_READ_ONLY	ID_MD
Unit_PositionDisplay	Position command unit display	Ready ON	WSTRING(31)	LIST_READ_ONLY	ID_MD
Unit_PositionString	Position command unit string	Ready ON	WSTRING(31)	LIST_READ_ONLY	ID_MD
Unit_Velocity	Speed command unit	Ready ON	DWORD(HEX)	LIST_READ_ONLY	ID_MD
Unit_VelocityDisplay	Speed command unit display	Ready ON	WSTRING(31)	LIST_READ_ONLY	ID_MD
UseInGroup	Using axes group	Operation cycle	BOOL	LIST_READ_ONLY	ID_MD
VelocityLimit_Negative	Negative direction speed limit value	Ready ON	LREAL	LIST_READ_ONLY	ID_MD
VelocityLimit_OverOperation	Operation setting at speed limit value exceeded	Ready ON	ENUM(MC_VELOCITY_LIMIT_MODE)	LIST_READ_ONLY	ID_MD
VelocityLimit_Positive	Positive direction speed limit value	Ready ON	LREAL	LIST_READ_ONLY	ID_MD
VelocityOverride	Speed override coefficient	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD
Warning	Axis warning detection	Immediately	BOOL	LIST_READ_ONLY	ID_MD
WarningID	Axis warning code	Immediately	WORD(HEX)	LIST_READ_ONLY	ID_MD

Variable/Structure name	Name	Import	Type	Attribute	Parameter type
AxisName.Cd.					
AccelerationOverride	Acceleration override coefficient	Operation cycle	LREAL	LIST_READ_ONLY	ID_CD
Encoder_Connect	Connection command	Operation cycle	BOOL	LIST_READ_ONLY	ID_CD
Encoder_CounterDisable	Counter disabled	Operation cycle	BOOL	LIST_READ_ONLY	ID_CD
Encoder_InputValue	Encoder input value	Operation cycle	DINT	LIST_READ_ONLY	ID_CD
ErrorReset	Axis error reset	Service	BOOL	LIST_READ_ONLY	ID_CD
FollowupDisable	Follow-up disabled	Operation cycle	BOOL	LIST_READ_ONLY	ID_CD
Homing_ClearRequest	Homing request clear	Operation cycle	BOOL	LIST_READ_ONLY	ID_CD
HwStrokeLimit_Override	Hardware stroke limit override	At start	STRING(15)	LIST_READ_ONLY	ID_CD
SlaveEmulate_Param	Slave emulation parameter	Immediately	STRUCT(TARGET_REF)	LIST_READ_ONLY	ID_CD
SlaveEmulate_Request	Slave emulation request	Service	ENUM(MC_SWITCHING_REQUEST)	LIST_READ_ONLY	ID_CD
SwStrokeLimit_Override	Software stroke limit override	At start	STRING(15)	LIST_READ_ONLY	ID_CD
TorqueLimit_Negative	Negative direction torque limit value	Operation cycle	LREAL	LIST_READ_ONLY	ID_CD
TorqueLimit_Positive	Positive direction torque limit value	Operation cycle	LREAL	LIST_READ_ONLY	ID_CD
VelocityOverride	Speed override coefficient	Operation cycle	LREAL	LIST_READ_ONLY	ID_CD

*1 First 3 digits of the 4 digits written in hexadecimal

Axes group variables

Variable/Structure name	Name	Import	Type	Attribute	Parameter type
AxesGroupName.AxesGroupRef.					
GroupNo	Axes group No.	—	WORD(UINT)	LIST_WRITE_BACK	ID_MDCST
StartIO	I/O No. *1	—	WORD(HEX)	LIST_WRITE_BACK	ID_MDCST
AxesGroupName.Pr.					
AccelerationLimit	Acceleration limit value	Ready ON	LREAL	LIST_WRITE_BACK	ID_PR
AccelerationZeroBehavior	Operation selection at start acceleration/deceleration 0	Ready ON	ENUM(MC_ACC_ZERO_MODE)	LIST_WRITE_BACK	ID_PR
Axis[1..16]	Structuring axis	Ready ON	STRUCT(AXIS_REF)	LIST_WRITE_BACK	ID_PR
CmdInPos_Width	Command In-position width	Ready ON	LREAL	LIST_WRITE_BACK	ID_PR
DecelerationLimit	Deceleration limit value	Ready ON	LREAL	LIST_WRITE_BACK	ID_PR
JerkLimit	Jerk limit value	Ready ON	LREAL	LIST_WRITE_BACK	ID_PR
OverrunOperation	Operation setting at overrun	Ready ON	ENUM(MC_DECEL_STOP_MODE)	LIST_WRITE_BACK	ID_PR
StopMode_Deceleration	Deceleration at stop	Ready ON	LREAL	LIST_WRITE_BACK	ID_PR
StopMode_DecelerationCurve	Stop selection at deceleration to stop	Ready ON	ENUM(MC_STOP_CURVE_MODE)	LIST_WRITE_BACK	ID_PR
StopMode_ErrorInGroup	Structuring axis operation selection at axis stop by factors	Ready ON	ENUM(MC_DECEL_STOP_MODE)	LIST_WRITE_BACK	ID_PR
StopMode_General	Stop selection at stop by factors	Ready ON	ENUM(MC_DECEL_STOP_MODE)	LIST_WRITE_BACK	ID_PR
Unit_Position	Position command unit	When Axes Group is Valid	DWORD(HEX)	LIST_WRITE_BACK	ID_PR
Unit_PositionString	Position command unit string	When Axes Group is Valid	WSTRING(31)	LIST_WRITE_BACK	ID_PR
Unit_Velocity	Speed command unit	When Axes Group is Valid	DWORD(HEX)	LIST_WRITE_BACK	ID_PR
VelocityLimit	Speed limit value	Ready ON	LREAL	LIST_WRITE_BACK	ID_PR
AxesGroupName.Md.					
AccelerationLimit	Acceleration limit value	Ready ON	LREAL	LIST_READ_ONLY	ID_MD

Variable/Structure name	Name	Import	Type	Attribute	Parameter type
AccelerationOverride	Acceleration override coefficient	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD
AccelerationZeroBehavior	Operation selection at start acceleration/deceleration 0	Ready ON	ENUM(MC_ACC_ZERO_MODE)	LIST_READ_ONLY	ID_MD
ActualVelocity	Feedback speed	Operation cycle(GroupEnable)	LREAL	LIST_READ_ONLY	ID_MD
Analyzing	Analyzing	Operation cycle	BOOL	LIST_READ_ONLY	ID_MD
AutoDeceleration	Automatically decelerating	Operation cycle	BOOL	LIST_READ_ONLY	ID_MD
Axis[1..16]	Structuring axis	Immediately	STRUCT(AXIS_REF)	LIST_READ_ONLY	ID_MD
BufferingFBs	Number of buffering FBs	At start/At stop	INT	LIST_READ_ONLY	ID_MD
CmdInPos	Command in-position	Operation cycle	BOOL	LIST_READ_ONLY	ID_MD
CmdInPos_Width	Command in-position width	Ready ON	LREAL	LIST_READ_ONLY	ID_MD
CommandedAcceleration	Specified acceleration	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD
CommandedDeceleration	Specified deceleration	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD
CommandedJerk	Specified jerk	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD
CommandedVelocity	Specified speed	Operation cycle(GroupEnable)	LREAL	LIST_READ_ONLY	ID_MD
DecelerationLimit	Deceleration limit value	Ready ON	LREAL	LIST_READ_ONLY	ID_MD
Error	Axes group error detection	Immediately	BOOL	LIST_READ_ONLY	ID_MD
ErrorID	Axes group error code	Immediately	WORD(HEX)	LIST_READ_ONLY	ID_MD
GroupName	Axes group name	Immediately	WSTRING(127)	LIST_READ_ONLY	ID_MD
GroupStatus	Axes group status	Operation cycle	ENUM(MC_AXES_GROUP_STATUS)	LIST_READ_ONLY	ID_MD
InterpolationAxes	Interpolation axis	Operation cycle	DWORD(HEX)	LIST_READ_ONLY	ID_MD
InVelocity	Target speed reached	Operation cycle	BOOL	LIST_READ_ONLY	ID_MD
JerkLimit	Jerk limit value	Ready ON	LREAL	LIST_READ_ONLY	ID_MD
NumberOfAxes	Number of structuring axes	Immediately	INT	LIST_READ_ONLY	ID_MD
OverrunOperation	Operation setting at overrun	Ready ON	ENUM(MC_DECEL_STOP_MODE)	LIST_READ_ONLY	ID_MD
SetAcceleration	Command current acceleration	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD
SetVelocity	Set velocity	Operation cycle(GroupEnable)	LREAL	LIST_READ_ONLY	ID_MD
StopMode_Deceleration	Deceleration at stop	Ready ON	LREAL	LIST_READ_ONLY	ID_MD
StopMode_DecelerationCurve	Stop selection at deceleration to stop	Ready ON	ENUM(MC_STOP_CURVE_MODE)	LIST_READ_ONLY	ID_MD
StopMode_ErrorInGroup	Structuring axis operation selection at axis stop by factors	Ready ON	ENUM(MC_DECEL_STOP_MODE)	LIST_READ_ONLY	ID_MD
StopMode_General	Stop selection at stop by factors	Ready ON	ENUM(MC_DECEL_STOP_MODE)	LIST_READ_ONLY	ID_MD
TargetAcceleration	Target acceleration	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD
TargetDeceleration	Target deceleration	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD
TargetVelocity	Target speed	Operation cycle(GroupEnable)	LREAL	LIST_READ_ONLY	ID_MD
Unit_Position	Position command unit	When Axes Group is Valid	DWORD(HEX)	LIST_READ_ONLY	ID_MD
Unit_PositionDisplay	Position command unit display	When Axes Group is Valid	WSTRING(31)	LIST_READ_ONLY	ID_MD
Unit_PositionString	Position command unit string	When Axes Group is Valid	WSTRING(31)	LIST_READ_ONLY	ID_MD
Unit_Velocity	Speed command unit	When Axes Group is Valid	DWORD(HEX)	LIST_READ_ONLY	ID_MD

Variable/Structure name	Name	Import	Type	Attribute	Parameter type
Unit_VelocityDisplay	Speed command unit display	When Axes Group is Valid	WSTRING(31)	LIST_READ_ONLY	ID_MD
VelocityLimit	Speed limit value	Ready ON	LREAL	LIST_READ_ONLY	ID_MD
VelocityOverride	Speed override coefficient	Operation cycle	LREAL	LIST_READ_ONLY	ID_MD
Warning	Axes group warning detection	Immediately	BOOL	LIST_READ_ONLY	ID_MD
WarningID	Axes group warning code	Immediately	WORD(HEX)	LIST_READ_ONLY	ID_MD
AxesGroupName.Cd.					
AccelerationOverride	Acceleration override coefficient	Operation cycle	LREAL	LIST_READ_ONLY	ID_CD
ErrorReset	Axes group error reset	Service	BOOL	LIST_READ_ONLY	ID_CD
VelocityOverride	Speed override coefficient	Operation cycle	LREAL	LIST_READ_ONLY	ID_CD

*1 First 3 digits of the 4 digits written in hexadecimal

System variables

Variable/Structure name	Name	Import	Type	Attribute	Parameter type
System.PrConst.					
Addon_AbsSystem	Add-on AbsSystem parameter	At system start	STRUCT(ADDON_PARAM)	LIST_WRITE_BACK	ID_PRCST
Addon_Axis	Add-on Axis parameter	At system start	STRUCT(ADDON_PARAM)	LIST_WRITE_BACK	ID_PRCST
Addon_ExternalSignal	Add-on ExternalSignal parameter	At system start	STRUCT(ADDON_PARAM)	LIST_WRITE_BACK	ID_PRCST
Addon_FileTransfer	Add-on FileTransfer parameter	At system start	STRUCT(ADDON_PARAM)	LIST_WRITE_BACK	ID_PRCST
Addon_Logging	Add-on Logging parameter	At system start	STRUCT(ADDON_PARAM)	LIST_WRITE_BACK	ID_PRCST
Addon_MotionControl_AxisFilter	Add-on MotionControl_AxisFilter parameter	At system start	STRUCT(ADDON_PARAM)	LIST_WRITE_BACK	ID_PRCST
Addon_MotionControl_General	Add-on MotionControl_General parameter	At system start	STRUCT(ADDON_PARAM)	LIST_WRITE_BACK	ID_PRCST
Addon_MotionControl_Sync	Add-on MotionControl_Sync parameter	At system start	STRUCT(ADDON_PARAM)	LIST_WRITE_BACK	ID_PRCST
Addon_MotionEngine	Add-on MotionEngine parameter	At system start	STRUCT(ADDON_PARAM)	LIST_WRITE_BACK	ID_PRCST
Addon_MotionEventHist	Add-on MotionEventHist parameter	At system start	STRUCT(ADDON_PARAM)	LIST_WRITE_BACK	ID_PRCST
Addon_NetworkDriver_CCIETSN	Add-on NetworkDriver_CCIETSN parameter	At system start	STRUCT(ADDON_PARAM)	LIST_WRITE_BACK	ID_PRCST
Addon_PackagingApp	Add-on PackagingApp parameter	At system start	STRUCT(ADDON_PARAM)	LIST_WRITE_BACK	ID_PRCST
Addon_PlcInstruction	Add-on PlcInstruction parameter	At system start	STRUCT(ADDON_PARAM)	LIST_WRITE_BACK	ID_PRCST
Addon_ProfileControl	Add-on ProfileControl parameter	At system start	STRUCT(ADDON_PARAM)	LIST_WRITE_BACK	ID_PRCST
Addon_Program_ST	Add-on Program_ST parameter	At system start	STRUCT(ADDON_PARAM)	LIST_WRITE_BACK	ID_PRCST
Addon_ServoDriver_CANopen	Add-on ServoDriver_CANopen parameter	At system start	STRUCT(ADDON_PARAM)	LIST_WRITE_BACK	ID_PRCST
Addon_ServoSystemRecorder	Add-on ServoSystemRecorder parameter	At system start	STRUCT(ADDON_PARAM)	LIST_WRITE_BACK	ID_PRCST
Addon_SignalIO	Add-on SignalIO parameter	At system start	STRUCT(ADDON_PARAM)	LIST_WRITE_BACK	ID_PRCST
BuffermemoryRefreshCycle	Buffer memory refresh cycle setting	At system start	STRUCT(CYCLE_PARAM)	LIST_WRITE_BACK	ID_PRCST

Variable/Structure name		Name	Import	Type	Attribute	Parameter type
	EventHistoryCpu_Exclude	Event excluding CPU event history	At system start	STRING(255)	LIST_WRITE_BACK	ID_PRCST
	EventHistoryMotion_Capacity	Motion event history capacity	At system start	INT	LIST_WRITE_BACK	ID_PRCST
	EventHistoryMotion_Exclude	Motion event history excluded event	At system start	STRING(255)	LIST_WRITE_BACK	ID_PRCST
	EventHistoryMotion_Path	Motion event history path	At system start	STRING(127)	LIST_WRITE_BACK	ID_PRCST
	ExcludeWarning	Excluded warning	At system start	STRING(255)	LIST_WRITE_BACK	ID_PRCST
	FastOperationMode	High-speed mode setting	At system start	WORD(HEX)	LIST_WRITE_BACK	ID_PRCST
	FileTransfer_AcFile_lch	File transfer access control (File in latch drive)	At system start	WORD(HEX)	LIST_READ_ONLY	ID_PRCST
	FileTransfer_AcFile_ram	File transfer access control (File in RAM drive)	At system start	WORD(HEX)	LIST_READ_ONLY	ID_PRCST
	FileTransfer_AcFile_rom	File transfer access control (File in user drive)	At system start	WORD(HEX)	LIST_READ_ONLY	ID_PRCST
	FileTransfer_AcFile_sdc	File transfer access control (File in SD memory card)	At system start	WORD(HEX)	LIST_READ_ONLY	ID_PRCST
	FileTransfer_AcFile_sys	File transfer access control (File in system drive)	At system start	WORD(HEX)	LIST_READ_ONLY	ID_PRCST
	FileTransfer_AcLabel	File transfer access control (Label)	At system start	WORD(HEX)	LIST_READ_ONLY	ID_PRCST
	FileTransfer_AutorunEnable	File transfer auto execution enabled	At system start	BOOL	LIST_READ_ONLY	ID_PRCST
	FileTransfer_LogCapacity	File transfer log capacity	At system start	INT	LIST_WRITE_BACK	ID_PRCST
	LabelMemorySize	Label memory size	At system start	INT	LIST_WRITE_BACK	ID_PRCST
	Link_MotionStationAutoRefresh	Motion control station link device auto refresh	At system start	BOOL	LIST_WRITE_BACK	ID_PRCST
	Link_MotionStationRefreshType	Refresh method of send/ receive data at motion control station	At system start	ENUM(MC_STATION_REFRESH_MODE)	LIST_WRITE_BACK	ID_PRCST
	Logging_AutoLoggingEnable	Auto logging enabled	At system start	BOOL	LIST_READ_ONLY	ID_PRCST
	MeasuresPrograms	Measurement flag for program information	At system start	BOOL	LIST_WRITE_BACK	ID_PRCST
	OperationCycle[1..1]	Operation cycle setting	At system start	STRUCT(CYCLE_PARAM)	LIST_WRITE_BACK	ID_PRCST
	SoftReboot_Enable	Soft reboot enabled	At system start	BOOL	LIST_WRITE_BACK	ID_PRCST
	MultiThreadMode	For Manufacturer Setting	At system start	WORD(HEX)	LIST_WRITE_BACK	ID_PRCST
System.Pr.						
	ForcedStop_Signal	All axes forced stop signal	Ready ON	STRUCT(SIGNAL_SELECT)	LIST_WRITE_BACK	ID_PR
	StopMode_All	Stop selection at all axes stop by factors	Ready ON	ENUM(MC_DECEL_STOP_MODE)	LIST_WRITE_BACK	ID_PR
	StopMode_AllDeceleration	Deceleration at all axes stop	Ready ON	LREAL	LIST_WRITE_BACK	ID_PR
System.Cd.						
	AddonLimit_Key	Add-on Library configuration restriction key setting	Immediately	STRING(31)	LIST_READ_ONLY	ID_CD
	AddonLimit_Level	Add-on Library configuration restriction level	Immediately	INT	LIST_READ_ONLY	ID_CD
	AddonLimit_Regist	Add-on Library configuration restriction request	Service	ENUM(MC_ADDON_LIMIT_CMD)	LIST_READ_ONLY	ID_CD
	BackupRestore	Backup restore request	Service	WORD(HEX)	LIST_READ_ONLY	ID_CD
	ErrorReset	System error reset	Service	BOOL	NOLIST_SYSTEM	ID_CD
	EventHistoryMotion_Clear	Motion Event History Clear Execution Request	Service	BOOL	LIST_READ_ONLY	D_CD
	FileTransfer_Command	File transfer command	Service	STRING(255)	LIST_READ_ONLY	ID_CD
	FileTransfer_Execute	File transfer execution request	Service	BOOL	LIST_READ_ONLY	ID_CD
	SequenceReady	PLC ready	Service	BOOL	NOLIST_SYSTEM	ID_CD
	SoftRebootRequest	Soft reboot request	Service	STRING(15)	NOLIST_SYSTEM	ID_CD

Variable/Structure name		Name	Import	Type	Attribute	Parameter type
	Storage_sdcForcedDisable	SD memory card force disabled command	Service	BOOL	LIST_READ_ONLY	ID_CD
	Storage_sdcRemovalProhibit	SD memory card removal prohibition	Service	BOOL	LIST_READ_ONLY	ID_CD
System.Md.						
	Addon_AbsSystem	Add-on AbsSystem monitor	Service	STRUCT(ADDON_MONI)	NOLIST_SYSTEM	ID_MD
	Addon_Axis	Add-on Axis monitor	Service	STRUCT(ADDON_MONI)	NOLIST_SYSTEM	ID_MD
	Addon_ExternalSignal	Add-on ExternalSignal monitor	Service	STRUCT(ADDON_MONI)	NOLIST_SYSTEM	ID_MD
	Addon_FileTransfer	Add-on FileTransfer monitor	Service	STRUCT(ADDON_MONI)	NOLIST_SYSTEM	ID_MD
	Addon_Logging	Add-on Logging monitor	Service	STRUCT(ADDON_MONI)	NOLIST_SYSTEM	ID_MD
	Addon_MotionControl_AxisFilter	Add-on MotionControl_AxisFilter monitor	Service	STRUCT(ADDON_MONI)	NOLIST_SYSTEM	ID_MD
	Addon_MotionControl_General	Add-on MotionControl_General monitor	Service	STRUCT(ADDON_MONI)	NOLIST_SYSTEM	ID_MD
	Addon_MotionControl_Sync	Add-on MotionControl_Sync monitor	Service	STRUCT(ADDON_MONI)	NOLIST_SYSTEM	ID_MD
	Addon_MotionEngine	Add-on MotionEngine monitor	Service	STRUCT(ADDON_MONI)	NOLIST_SYSTEM	ID_MD
	Addon_MotionEventHist	Add-on MotionEventHist monitor	Service	STRUCT(ADDON_MONI)	NOLIST_SYSTEM	ID_MD
	Addon_NetworkDriver_CCIETSN	Add-on NetworkDriver_CCIETSN monitor	Service	STRUCT(ADDON_MONI)	NOLIST_SYSTEM	ID_MD
	Addon_PackagingApp	Add-on PackagingApp monitor	Service	STRUCT(ADDON_MONI)	NOLIST_SYSTEM	ID_MD
	Addon_PlcInstruction	Add-on PlcInstruction monitor	Service	STRUCT(ADDON_MONI)	NOLIST_SYSTEM	ID_MD
	Addon_ProfileControl	Add-on ProfileControl monitor	Service	STRUCT(ADDON_MONI)	NOLIST_SYSTEM	ID_MD
	Addon_Program_ST	Add-on Program_ST monitor	Service	STRUCT(ADDON_MONI)	NOLIST_SYSTEM	ID_MD
	Addon_ServoDriver_CANopen	Add-on ServoDriver_CANopen monitor	Service	STRUCT(ADDON_MONI)	NOLIST_SYSTEM	ID_MD
	Addon_ServoSystemRecorder	Add-on ServoSystemRecorder monitor	Service	STRUCT(ADDON_MONI)	NOLIST_SYSTEM	ID_MD
	Addon_SignalIO	Add-on SignalIO monitor	Service	STRUCT(ADDON_MONI)	NOLIST_SYSTEM	ID_MD
	AddonLimit_Level	Add-on Library configuration restriction level	Immediately	INT	NOLIST_SYSTEM	ID_MD
	AddonLimit_State	Add-on Library configuration restriction state	Immediately	ENUM(MC_ADDON_LIMIT_STATE)	NOLIST_SYSTEM	ID_MD
	BackupRestoreStatus	Backup restore information	Service	INT	LIST_READ_ONLY	ID_MD
	BuffermemoryFreeMcfbArea	Buffer memory MCFB Communication area free size	Immediately	DWORD(UDINT)	LIST_READ_ONLY	ID_MD
	BuffermemoryRefreshCycle	Buffer memory refresh cycle monitor	Immediately	STRUCT(CYCLE_MONI)	NOLIST_SYSTEM	ID_MD
	Error	Motion system error detection	Immediately	BOOL	NOLIST_SYSTEM	ID_MD
	ErrorID	Latest motion system error code	Immediately	WORD(HEX)	NOLIST_SYSTEM	ID_MD
	EventHistoryMotion_ClearStatus	Motion event history clear executing state	Service	ENUM(MC_EXECUTION_STATE)	LIST_READ_ONLY	ID_MD
	EventHistoryMotion_Path	Motion event history path	At system start	STRING(127)	NOLIST_SYSTEM	ID_MD

Variable/Structure name	Name	Import	Type	Attribute	Parameter type
FileTransfer_State	File transfer executing state	Service	ENUM(MC_EXECUTION_STATE)	LIST_READ_ONLY	ID_MD
ForcedStop_Released	Forced stop cancelling	Operation cycle	BOOL	LIST_READ_ONLY	ID_MD
ForcedStop_Signal	All axes forced stop signal	Ready ON	STRUCT(SIGNAL_SELECT)	LIST_READ_ONLY	ID_MD
HwStatus_OperationgTime	Cumulative power-on time	Service	DWORD(UDINT)	NOLIST_SYSTEM	ID_MD
HwStatus_RomWriteCycle	Flash ROM write count index value	Service	DWORD(UDINT)	NOLIST_SYSTEM	ID_MD
LabelMemoryFreeSize	Label memory free size	Immediately	INT	LIST_READ_ONLY	ID_MD
MemorySize	System memory size	Service	STRUCT(ADDON_PARAM)	NOLIST_SYSTEM	ID_MD
MemoryUsage	System memory usage	Service	STRUCT(ADDON_MONI)	NOLIST_SYSTEM	ID_MD
NetworkError	Network error detection	Immediately	BOOL	NOLIST_SYSTEM	ID_MD
NetworkErrorID	Network error code	Immediately	WORD(HEX)	NOLIST_SYSTEM	ID_MD
OperationCycle[1..1]	Operation cycle monitor	Immediately	STRUCT(CYCLE_MONI)	NOLIST_SYSTEM	ID_MD
Program_Ei	EI flag	Immediately	BOOL	LIST_READ_ONLY	ID_MD
Program_NormalTaskCycle	Normal task processing time monitor	Normal	STRUCT(CYCLE_MONI)	NOLIST_SYSTEM	ID_MD
Ready	Ready	Service	BOOL	NOLIST_SYSTEM	ID_MD
ServoSystemRecorder_Enable	Servo system recorder enabled	At system start	BOOL	LIST_READ_ONLY	ID_MD
StopMode_All	Stop selection at all axes stop by factors	Ready ON	ENUM(MC_DECEL_STOP_MODE)	LIST_READ_ONLY	ID_MD
StopMode_AllDeceleration	Deceleration at all axes stop	At startReady ON	LREAL	LIST_READ_ONLY	ID_MD
Storage_Ich	Latch drive information	Service	STRUCT(STORAGE_INFORMATION)	NOLIST_SYSTEM	ID_MD
Storage_ram	RAM drive information	Service	STRUCT(STORAGE_INFORMATION)	NOLIST_SYSTEM	ID_MD
Storage_rom	User drive information	Service	STRUCT(STORAGE_INFORMATION)	NOLIST_SYSTEM	ID_MD
Storage_sdc	SD memory card information	Service	STRUCT(STORAGE_INFORMATION)	NOLIST_SYSTEM	ID_MD
Storage_sdcForcedDisabled	SD memory card force disabled	Service	BOOL	NOLIST_SYSTEM	ID_MD
Storage_sdcInserted	SD memory card inserted	Service	BOOL	NOLIST_SYSTEM	ID_MD
Storage_sdcProtected	SD memory card protected	Service	BOOL	NOLIST_SYSTEM	ID_MD
Storage_sys	System drive information	Service	STRUCT(STORAGE_INFORMATION)	NOLIST_SYSTEM	ID_MD
Sync	Synchronization flag	Service	BOOL	NOLIST_SYSTEM	ID_MD
SystemBaseCycle	System basic cycle monitor	Immediately	STRUCT(CYCLE_MONI)	NOLIST_SYSTEM	ID_MD
SystemBaseCycle_Counter	System basic cycle counter	System basic cycle	WORD(UINT)	NOLIST_SYSTEM	ID_MD
Version_BaseSystemSw	Basic system software version	At system start	WORD(HEX)	NOLIST_SYSTEM	ID_MD
Version_BootSw	Boot software version	At system start	WORD(HEX)	NOLIST_SYSTEM	ID_MD
Version_NetworkBootSw	Network boot software version	At system start	WORD(HEX)	NOLIST_SYSTEM	ID_MD
Warning	Motion system warning detection	Immediately	BOOL	NOLIST_SYSTEM	ID_MD
WarningID	Latest motion system warning code	Immediately	WORD(HEX)	NOLIST_SYSTEM	ID_MD
WDTerror	WDT error	Immediately	BOOL	NOLIST_SYSTEM	ID_MD
System.LoggingRef[1..10].Cd.					

Variable/Structure name	Name	Import	Type	Attribute	Parameter type
SettingPath	Logging setting path	Immediately	STRING(127)	LIST_READ_ONLY	ID_CD
StartLog	Logging request	Service	BOOL	LIST_READ_ONLY	ID_CD
Trigger	Trigger request	Logging setting cycle	BOOL	LIST_READ_ONLY	ID_CD
System.LoggingRef[1..10].Md.					
ErrStatus	Error status	Service	WORD(UINT)	LIST_READ_ONLY	ID_MD
LogDataPath	Logging data file path	Service	STRING(127)	LIST_READ_ONLY	ID_MD
LogDataSavedCount	Number of saved files	Service	WORD(UINT)	LIST_READ_ONLY	ID_MD
LoggingStatus	Logging status	Service	BOOL	LIST_READ_ONLY	ID_MD
LogType	Logging type	Logging setting cycle	ENUM(MC_LOGGING_TYPE)	LIST_READ_ONLY	ID_MD
SamplingTime[0..1]	Data collection interval	Service	DWORD(UDINT)	LIST_READ_ONLY	ID_MD
SaveStatus	Save status	Service	ENUM(MC_LOGGING_SAVE_STATUS)	LIST_READ_ONLY	ID_MD
SettingPath	Logging setting file	Immediately	STRING(127)	LIST_READ_ONLY	ID_MD
TriggerCount	Trigger detection count	Logging setting cycle	INT	LIST_READ_ONLY	ID_MD
TrigerCountIgnored	Trigger ignored count	Logging setting cycle	INT	LIST_READ_ONLY	ID_MD
Triggered	Triggered	Logging setting cycle	BOOL	LIST_READ_ONLY	ID_MD
TriggerStatus[1..32]	Trigger status	Logging setting cycle	BOOL	LIST_READ_ONLY	ID_MD

Other structure variables

Variable/Structure name	Name	Type
MC_CAM_REF		
ProfileData	Profile	STRUCT(./PROFILE_DATA)
MC_CAM_ID		
ProfileID	Profile ID	STRUCT(./PROFILE_ID)
MC_TRIGGER_REF		
Signal	Trigger signal	STRUCT(./SIGNAL_SELECT)
MC_INPUT_REF		
Signal	Input signal	STRUCT(./SIGNAL_SELECT)
MC_OUTPUT_REF		
Signal	Output signal	STRUCT(./SIGNAL_SELECT)
TARGET_REF	Input signal	
StartIO	IO No.	WORD(HEX)
Target	Target	WSTRING(63)
SIGNAL_SELECT	Signal select	
Source	Signal	STRUCT(./TARGET_REF)
Detection	Signal detection method	ENUM(MC_SIGNAL_LOGIC)
CompensationTime	Compensation time	LREAL
FilterTime	Filter time	LREAL
PROFILE_DATA		
Location	Calculation profile data storage location	STRUCT(./FILE_LOCATION)
ID	Profile ID	STRUCT(./PROFILE_ID)
FILE_LOCATION		
FileName	File name	STRING(63)
Path	Folder specification	STRING(63)
PROFILE_ID		
Number	Profile ID No.	WORD(UINT)
CYCLE_PARAM		

Variable/Structure name		Name	Type
	Cycle	Cycle setting	DINT
	NumOfCycleOverWngDetectTimes	Number of cycle over warning detections	WORD(UINT)
	NumOfCycleOverErrDetectTimes	Number of cycle over error detections	WORD(UINT)
	CycleOverErrorType	Cycle over error selection	ENUM(MC_ERROR_CLASS)
CYCLE_MONI			
	ProcessingTime	Processing time	DWORD(UDINT)
	MaximumProcessingTime	Maximum processing time	DWORD(UDINT)
	Cycle	Setting cycle	DWORD(UDINT)
	CycleOver	Cycle over	BOOL
ADDON_PARAM			
	RamSizeMax	Maximum RAM size	DWORD(UDINT)
	BackupRamSizeMax	Maximum backup RAM size	DWORD(UDINT)
ADDON_MONI			
	RamUsage	RAM usage	DWORD(UDINT)
	RamMaxUsage	RAM usage maximum	DWORD(UDINT)
	BackupRamUsage	Backup RAM usage	DWORD(UDINT)
	BackupRamMaxUsage	Backup RAM usage maximum	DWORD(UDINT)
	Version	Version	STRING(15)
STORAGE_INFORMATION			
	Capacity	Capacity	DWORD(UDINT)
	FreeSpace	Free space	DWORD(UDINT)
	Mount	Mount status	BOOL
INSTANCE_ID			
	StartIO	IO No.	WORD(HEX)
	Number	Instance ID	WORD(UINT)
PROFILE_CAM_DATA			
	Interpolate	Specifying interpolation method	INT
	Resolution	Resolution	DWORD(UDINT)
	InputUnitString	Input unit string	WSTRING(31)
	OutputUnitString	Output unit string	WSTRING(31)
	StartPoint	Start point	LREAL
	StartStroke	Initial stroke amount	LREAL
	StartVelocity	Initial speed	LREAL
	StartAcceleration	Initial acceleration	LREAL
	CycleLength	Length per cycle	LREAL
	CycleMin	Minimum value per cycle	LREAL
	CycleMax	Maximum value per cycle	LREAL
	CycleTime	Time per cycle	LREAL
	Stroke	Stroke amount	LREAL
	NumberOfSections	Number of sections	WORD(UINT)
	Options	Options	DWORD(HEX)
PROFILE_CAM_ELEMENT			
	CurveType	Cam curve type	ENUM(./MC_CAM_CURVE_TYPE)
	EndPoint	End point	LREAL
	Stroke	Stroke	LREAL
	RangeP1	Curve application range (P1)	LREAL
	RangeP2	Curve application range (P2)	LREAL
	RangeL1	Acceleration/deceleration range compensation (Range L1)	LREAL
	RangeL2	Acceleration/deceleration range compensation (Range L2)	LREAL
	EndVelocity	End point speed	LREAL
	EndAcceLeration	End point acceleration	LREAL
PROFILE_ROTARY_CUTTER			
	Resolution	Resolution	DWORD(UDINT)

Variable/Structure name		Name	Type
	InputUnitString	Input unit string	WSTRING(31)
	Options	Options	DWORD(HEX)
	AutoGenerationOptions	Auto-generation option	WORD(HEX)
	NumberOfCutter	Number of cutter	WORD(UINT)
	SyncSectionAccRatio	Synchronous section acceleration ratio	LREAL
	SheetLength	Sheet length	LREAL
	SheetSyncWidth	Sheet synchronization width	LREAL
	SyncAxisLength	Synchronous axis length	LREAL
	SyncPositionAdjustment	Synchronous position tuning	LREAL
	AccDecWidth	Acceleration/deceleration width	LREAL
	AsyncSpd	Asynchronous speed result	LREAL
PROFILE_MULTIDIMENSIONAL_DATA			
	Dimension	Number of input dimensions	DWORD(UDINT)
	Resolution	Resolution	DWORD[0..2]
	InputUnitString	Input unit string	WSTRING(31)[0..2]
	OutputUnitString	Output unit string	WSTRING(31)[0..2]
	CycleLength	Length per cycle	LREAL[0..2]
	Stroke	Stroke amount	LREAL[0..2]
	Options	Options	DWORD(HEX)
PROFILE_MULTIDIMENSIONAL_ELEMENT			
	Input1	Input value	LREAL
	Input2	Input value	LREAL
	Input3	Input value	LREAL
	Output	Output value	LREAL
PROGRAM_INFO			
	ExecutionTime	Execution time	DWORD(UDINT)
	Number	Execution count	WORD(UINT)
	Status	Status	WORD(HEX)
SLAVE_OBJECT_REAL			
	ControlWord	ControlWord	WSTRING(63)
	EncoderIncrements	EncoderIncrements	WSTRING(63)
	FollowingErrActualValue	FollowingErrActualValue	WSTRING(63)
	GearRatioMotorRevolutions	GearRatioMotorRevolutions	WSTRING(63)
	HomeOffset	HomeOffset	WSTRING(63)
	MaxMotorSpeed	MaxMotorSpeed	WSTRING(63)
	MaxTorque	MaxTorque	WSTRING(63)
	ModesOfOp	ModesOfOp	WSTRING(63)
	ModesOfOpDisp	ModesOfOpDisp	WSTRING(63)
	NegativeTorqueLimitValue	NegativeTorqueLimitValue	WSTRING(63)
	Polarity	Polarity	WSTRING(63)
	PosActualValue	PosActualValue	WSTRING(63)
	PosEncoderResolution	PosEncoderResolution	WSTRING(63)
	PosEncoderResolutionMotorRevolutions	PosEncoderResolutionMotorRevolutions	WSTRING(63)
	PositiveTorqueLimitValue	PositiveTorqueLimitValue	WSTRING(63)
	ShaftRevolutions	ShaftRevolutions	WSTRING(63)
	SiUnitPos	SiUnitPos	WSTRING(63)
	SiUnitVel	SiUnitVel	WSTRING(63)
	StatusWord	StatusWord	WSTRING(63)
	SupportedDriveModes	SupportedDriveModes	WSTRING(63)
	TargetPos	TargetPos	WSTRING(63)
	TargetTorque	TargetTorque	WSTRING(63)
	TargetVelocity	TargetVelocity	WSTRING(63)
	TargetActualValue	TorqueActualValue	WSTRING(63)

Variable/Structure name		Name	Type
	VelActualValue	VelActualValue	WSTRING(63)
	vControlDil1	ControlDi1	WSTRING(63)
	vControlDil2	ControlDi2	WSTRING(63)
	vControlDil3	ControlDi3	WSTRING(63)
	vControlDil4	ControlDi4	WSTRING(63)
	vControlDil5	ControlDi5	WSTRING(63)
	vControlDil6	ControlDi6	WSTRING(63)
	vControlDil7	ControlDi7	WSTRING(63)
	vCurrentAlarm	CurrentAlarm	WSTRING(63)
	vEncoderStatus1	EncoderStatus1	WSTRING(63)
	vHomeAbsCounter	HomeAbsCounter	WSTRING(63)
	vHomeCycleCounter	HomeCycleCounter	WSTRING(63)
	vInitialAbsCounter	InitialAbsCounter	WSTRING(63)
	vInitialCycleCounter	InitialCycleCounter	WSTRING(63)
	vInitialPos	InitialPos	WSTRING(63)
	vMaxAbsCounter	MaxAbsCounter	WSTRING(63)
	vStatusDo1	StatusDo1	WSTRING(63)
	vStatusDo2	StatusDo2	WSTRING(63)
	vStatusDo3	StatusDo3	WSTRING(63)
	vStatusDo4	StatusDo4	WSTRING(63)
	vStatusDo5	StatusDo5	WSTRING(63)
	vStatusDo6	StatusDo6	WSTRING(63)
	vStatusDo7	StatusDo7	WSTRING(63)
	vSupportedControlDi1	SupportedControlDi1	WSTRING(63)
	vSupportedControlDi2	SupportedControlDi2	WSTRING(63)
	vSupportedControlDi3	SupportedControlDi3	WSTRING(63)
	vSupportedControlDi4	SupportedControlDi4	WSTRING(63)
	vSupportedControlDi5	SupportedControlDi5	WSTRING(63)
	vSupportedControlDi6	SupportedControlDi6	WSTRING(63)
	vSupportedControlDi7	SupportedControlDi7	WSTRING(63)
	vSupportedStatusDo1	SupportedStatusDo1	WSTRING(63)
	vSupportedStatusDo2	SupportedStatusDo2	WSTRING(63)
	vSupportedStatusDo3	SupportedStatusDo3	WSTRING(63)
	vSupportedStatusDo4	SupportedStatusDo4	WSTRING(63)
	vSupportedStatusDo5	SupportedStatusDo5	WSTRING(63)
	vSupportedStatusDo6	SupportedStatusDo6	WSTRING(63)
	vSupportedStatusDo7	SupportedStatusDo7	WSTRING(63)
	vSyncAbsCounter	SyncAbsCounter	WSTRING(63)
	vSyncCycleCounter	SyncCycleCounter	WSTRING(63)
	vVelLimitValue	VelLimitValue	WSTRING(63)
	vWatchDogCounterDI	WatchDogCounterDI	WSTRING(63)
	vWatchDogCounterUI	WatchDogCounterUI	WSTRING(63)
	vEncoderStatus2	EncoderStatus2	WSTRING(63)
	vScaleCycleCounte	ScaleCycleCounter	WSTRING(63)
	vScaleAbsCounter	ScaleAbsCounter	WSTRING(63)
	vScaleMeasurementEncoderResolution	ScaleMeasurementEncoderResolution	WSTRING(63)
	vScaleMeasurementEncoderReceptionStatus	ScaleMeasurementEncoderReceptionStatus	WSTRING(63)
SLAVE_OBJECT_ENCODER			
	PosActualValue	PosActualValue	WSTRING(63)
SLAVE_OBJECT_VIRTUAL_ENCODER			
	PosActualValue	PosActualValue	WSTRING(63)

25 List of Supported Axis Type

Variables supported by each axis type are shown below.
○: Supported, ×: Not supported

Axis variables

Variable/Structure name		Name	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 linked axis (AXIS_VIRTUAL_LINK)
AxisName.PrConst.							
	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	×	×	×	○	×
	FastOperationMode	High-speed Mode Setting	○	○	○	○	○
	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	○	×	×	×	×
	SlaveObject	Slave Object Data	×	○	×	×	×
	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	○	×	×	×	×
AxisName.Pr.							
	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	○	○	×	○	×

Variable/Structure name	Name	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 linked axis (AXIS_VIRTUAL_LINK)
Drive_UnitConvRatioDen	Driver Unit Conversion Denominator	○	○	×	○	×
ForcedStop_Signal	Forced Stop Signal	○	×	○	×	×
Homing_Required	Homing Required or Not	○	○	○	○	○
JerkLimit	Jerk Limit Value	○	×	○	×	×
OverrunOperation	Operation Setting at Overrun	○	×	○	×	×
StartableAtUnhomed	Start Permission at Homing 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	Positive Direction Speed Limit Value	○	×	○	×	×
AxisName.Md.						
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	○	×	○	×	×

Variable/Structure name	Name	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 linked axis (AXIS_VIRTUAL_LINK)
CmdInPos	Command In-position	○	×	○	×	×
CmdInPos_Width	Command In-position Width	○	×	○	×	×
CommandedAcceleration	Specified Acceleration	○	×	○	×	×
CommandedDeceleration	Specified Deceleration	○	×	○	×	×
CommandedJerk	Specified Jerk	○	×	○	×	×
CommandedPosition	Specified Position	○	×	○	×	×
CommandedVelocity	Specified Speed	○	×	○	×	×
Cst_SetTorque	Command Current Torque at Torque Control	○	×	×	×	×
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	Homing Completed	○	○	○	○	○
Homing_Request	Homing Request	○	○	○	○	○
Homing_Required	Homing Required or Not	○	○	○	○	○
Homing_Status	Homing Operation Status	○	×	×	×	×
HwStrokeLimit_FlsStatus	Upper Limit Signal Status	○	×	○	×	×
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	○	×	○	×	×

Variable/Structure name	Name	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 linked axis (AXIS_VIRTUAL_LINK)
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	○	×	○	×	×
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	○	○	○	○	○

Variable/Structure name		Name	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 linked axis (AXIS_VIRTUAL_LINK)
	Unit_Velocity	Speed Command Unit	○	○	○	○	○
	Unit_VelocityDisplay	Speed Command Unit Display	○	○	○	○	○
	UseInGroup	Using Axes Group	○	○	○	○	○
	VelocityLimit_Negative	Negative Direction Speed Limit Value	○	×	○	×	×
	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.							
	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	Homing Request Clear	○	○	○	○	○
	HwStrokeLimit_Override	Hardware Stroke Limit Override	○	×	○	×	×
	SlaveEmulate_Param	Slave Emulation Parameter	○	○	×	×	×
	SlaveEmulate_Request	Slave Emulation Request	○	○	×	×	×
	SwStrokeLimit_Override	Software Stroke Limit Override	○	×	○	×	×
	TorqueLimit_Negative	Negative Direction Torque Limit Value	○	×	×	×	×
	TorqueLimit_Positive	Positive Direction Torque Limit Value	○	×	×	×	×
	VelocityOverride	Speed Override Coefficient	○	×	○	×	×

26 List of FBs

Name	Description	Type	Reference
MC_CamIn	Starts cam operation.	MCFB (motion/single)	Page 443 MC_CamIn
MC_CombineAxes	Combines the motion of 2 axes.	MCFB (motion/single)	Page 466 MC_CombineAxes
MC_GearIn	Starts gear operation.	MCFB (motion/single)	Page 461 MC_GearIn
MC_GroupStop	Executes a forced stop for an axes group.	MCFB (motion/single)	Page 168 MC_GroupStop
MC_Home	Executes homing.	MCFB (motion/single)	Page 200 MC_Home
MC_MoveAbsolute	Executes positioning (absolute).	MCFB (motion/single)	Page 206 MC_MoveAbsolute
MC_MoveRelative	Executes positioning (relative).	MCFB (motion/single)	Page 208 MC_MoveRelative
MC_MoveVelocity	Executes speed control.	MCFB (motion/single)	Page 276 MC_MoveVelocity
MCv_SpeedControl	Executes speed control (including position loop).	MCFB (motion/single)	Page 220 MCv_SpeedControl
MC_Stop	Executes a forced stop.	MCFB (motion/single)	Page 166 MC_Stop
MC_TorqueControl	Executes torque control.	MCFB (motion/single)	Page 287 MC_TorqueControl
MCv_BacklashCompensationFilter	Backlash compensation filter	MCFB (motion/single)	Page 391 MCv_BacklashCompensationFilter
MCv_DirectionFilter	Travel direction control filter	MCFB (motion/single)	Page 387 MCv_DirectionFilter
MCv_Jog	Executes JOG operation.	MCFB (motion/single)	Page 224 MCv_Jog
MCv_MoveCircularInterpolateAbsolute	Executes circular interpolation control (absolute).	MCFB (motion/single)	Page 237 MCv_MoveCircularInterpolateAbsolute
MCv_MoveCircularInterpolateRelative	Executes circular interpolation control (relative).	MCFB (motion/single)	Page 240 MCv_MoveCircularInterpolateRelative
MCv_MoveLinearInterpolateAbsolute	Executes linear interpolation control (absolute).	MCFB (motion/single)	Page 233 MCv_MoveLinearInterpolateAbsolute
MCv_MoveLinearInterpolateRelative	Executes linear interpolation control (relative).	MCFB (motion/single)	Page 235 MCv_MoveLinearInterpolateRelative
MCv_SmoothingFilter	Enables smoothing filter.	MCFB (motion/single)	Page 384 MCv_SmoothingFilter
MCv_SpeedLimitFilter	Velocity control filter	MCFB (motion/single)	Page 389 MCv_SpeedLimitFilter
MC_AbortTrigger	Disables the touch probe.	MCFB (motion/single)	Page 425 MC_AbortTrigger
MC_CamTableSelect	Selects cam tables.	MCFB (administrative)	Page 474 MC_CamTableSelect
MC_GroupDisable	Disables an axes group.	MCFB (administrative)	Page 68 MC_GroupDisable
MC_GroupEnable	Enables an axes group.	MCFB (administrative)	Page 66 MC_GroupEnable
MC_GroupReset	Axes group error reset	MCFB (administrative)	Page 645 MC_GroupReset
MC_GroupSetOverride	Sets the values of override for an axes group.	MCFB (administrative)	Page 363 MC_GroupSetOverride
MC_Power	Controls the power stage (ON or OFF) for a single axis.	MCFB (administrative)	Page 88 MC_Power
MC_ReadParameter	Parameter read	MCFB (administrative)	Page 605 MC_ReadParameter
MC_Reset	Axes error reset	MCFB (administrative)	Page 644 MC_Reset
MC_SetOverride	Sets the values of override.	MCFB (administrative)	Page 361 MC_SetOverride
MC_SetPosition	Changes the current position.	MCFB (administrative)	Page 304 MC_SetPosition
MC_TouchProbe	Enables the touch probe.	MCFB (administrative)	Page 423 MC_TouchProbe
MC_WriteParameter	Parameter write	MCFB (administrative)	Page 607 MC_WriteParameter
MCv_AllPower	Controls the power stage (ON or OFF) for all axes.	MCFB (administrative)	Page 91 MCv_AllPower
MCv_ChangeCycle	Changes the current value per cycle.	MCFB (administrative)	Page 484 MCv_ChangeCycle
MCv_ChangeCycleTriggered*1	Changes the current value per cycle.	MCFB (administrative)	Page 486 MCv_ChangeCycleTriggered (Future support planned)
MCv_MotionErrorReset	Motion error reset	MCFB (administrative)	Page 643 MCv_MotionErrorReset
MCv_SetPositionTriggered*1	Compensates the current position at a trigger event.	MCFB (administrative)	Page 307 MCv_SetPositionTriggered (Future support planned)
MCv_SetTorqueLimit	Sets torque limits.	MCFB (administrative)	Page 374 MCv_SetTorqueLimit
MCv_ProfileCalc*1	Calculates profile data.	General FB	Page 478 MCv_ProfileCalc (Future support planned)

Name	Description	Type	Reference
MCv_ProfileSelect*1	Selects profile data.	General FB	Page 476 MCv_ProfileSelect (Future support planned)
MCv_ReadProfileData	Reads profile data.	General FB	Page 480 MCv_ReadProfileData
MCv_WriteProfileData	Writes profile data.	General FB	Page 482 MCv_WriteProfileData

*1 These FBs are planned for a future support.

27 List of Add-on Library

Each add-on version which is compatible with the software version of the Motion module is as follows.

Add-on library name	The software version of the Motion module							
	01	02	03	04	05	06	07	08
baseSystem	1.0.1.0	1.2.1.0	1.3.1.0	1.4.1.0	1.5.1.0	1.6.1.0	1.7.1.0	1.8.1.0
MotionEngine	1.0.1.0	1.1.1.0	1.2.1.0	1.3.1.0	1.4.1.0	1.5.1.0	1.6.1.0	1.7.1.0
Axis	1.0.1.0	1.1.1.0	1.2.1.0	1.3.1.0	1.4.1.0	1.5.1.0	1.6.1.0	1.7.1.0
MotionControl_General	1.0.1.0	1.1.1.0	1.2.1.0	1.3.1.0	1.4.1.0	1.5.1.0	1.6.1.0	1.7.1.0
MotionControl_Sync	1.0.1.0	1.1.1.0	1.2.1.0	1.3.1.0	1.4.1.0	1.5.1.0	1.6.1.0	1.7.1.0
AbsSystem	1.0.1.0	1.1.1.0	1.2.1.0	1.3.1.0	1.4.1.0	1.5.1.0	1.6.1.0	1.7.1.0
ServoDriver_CANopen	1.0.1.0	1.1.1.0	1.2.1.0	1.3.1.0	1.4.1.0	1.5.1.0	1.6.1.0	1.7.1.0
NetworkDriver_CCIETSN	1.0.1.0	1.1.1.0	1.2.1.0	1.3.1.0	1.4.1.0	1.5.1.0	1.6.1.0	1.7.1.0
PlcInstruction	1.0.1.0	1.1.1.0	1.2.1.0	1.3.1.0	1.4.1.0	1.5.1.0	1.6.1.0	1.7.1.0
SignalIO	1.0.1.0	1.1.1.0	1.2.1.0	1.3.1.0	1.4.1.0	1.5.1.0	1.6.1.0	1.7.1.0
ExternalSignal	1.0.1.0	1.1.1.0	1.2.1.0	1.3.1.0	1.4.1.0	1.5.1.0	1.6.1.0	1.7.1.0
Program_ST	1.0.1.0	1.1.1.0	1.2.1.0	1.3.1.0	1.4.1.0	1.5.1.0	1.6.1.0	1.7.1.0
Logging	1.0.1.0	1.1.1.0	1.2.1.0	1.3.1.0	1.4.1.0	1.5.1.0	1.6.1.0	1.7.1.0
MotionEventHist	1.0.1.0	1.1.1.0	1.2.1.0	1.3.1.0	1.4.1.0	1.5.1.0	1.6.1.0	1.7.1.0
FileTransfer	1.0.1.0	1.1.1.0	1.2.1.0	1.3.1.0	1.4.1.0	1.5.1.0	1.6.1.0	1.7.1.0
MotionControl_AxisFilter	1.0.1.0	1.1.1.0	1.2.1.0	1.3.1.0	1.4.1.0	1.5.1.0	1.6.1.0	1.7.1.0
ProfileControl	1.0.1.0	1.1.1.0	1.2.1.0	1.3.1.0	1.4.1.0	1.5.1.0	1.6.1.0	1.7.1.0
PackagingApp	1.0.1.0	1.1.1.0	1.2.1.0	1.3.1.0	1.4.1.0	1.5.1.0	1.6.1.0	1.7.1.0
ServoSystemRecorder	—	—	—	—	—	—	—	1.7.1.0

The details about the add-on library are as follows.

—: No dependent add-on

Add-on library name	Name	Description	Dependent add-on (Necessary)	Dependent add-on (Restricted)
baseSystem	baseSystem	Basic software	—	—
MotionEngine	Motion control basic	Basic add-on for motion control	{"Name":"baseSystem","Later":"1.6","Earlier":"1.6"} {"Name":"Axis","Later":"1.5"}	—
Axis	Axis control	Add-on for axis control	{"Name":"baseSystem","Later":"1.6","Earlier":"1.6"}, {"Name":"MotionEngine","Later":"1.5"}	[AbsSystem] [ServoDriver] [SignalIO] [MotionEventHist]
MotionControl_General	PLCopen motion control single-axis/multi-axis control FB	Enables PLCopen® motion control single-axis/multi-axis control FB.	{"Name":"baseSystem","Later":"1.6","Earlier":"1.6"}, {"Name":"MotionEngine","Later":"1.5"}, {"Name":"Axis","Later":"1.5"}	[SignalIO]
MotionControl_Sync	PLCopen motion control synchronous control	Enables PLCopen® motion control synchronous control FB.	{"Name":"baseSystem","Later":"1.6","Earlier":"1.6"}, {"Name":"MotionEngine","Later":"1.5"}, {"Name":"Axis","Later":"1.5"}, {"Name":"MotionControl_General","Later":"1.5"}	—
AbsSystem	Absolute position control	Enables absolute position control.	{"Name":"baseSystem","Later":"1.6","Earlier":"1.6"}, {"Name":"Axis","Later":"1.5"}	—
ServoDriver_CANopen	Servo driver (CANopen)	CANopen servo driver	{"Name":"baseSystem","Later":"1.6","Earlier":"1.6"}, {"Name":"Axis","Later":"1.5"}, {"Name":"NetworkDriver","Later":"1.5"} {"Name":"SignalIO","Later":"1.5"}	—
NetworkDriver_CCIETSN	Network driver (CC-Link IE TSN)	Network driver of CC-Link IE TSN	{"Name":"baseSystem","Later":"1.6","Earlier":"1.6"}	—
PlcInstruction	External sequence dedicated instructions	Enables external sequence dedicated instructions.	{"Name":"baseSystem","Later":"1.6","Earlier":"1.6"}	[PLCopenMC_General] [ST] [SignalIO]
SignalIO	I/O signal	Provides the signal input/output function.	{"Name":"baseSystem","Later":"1.6","Earlier":"1.6"}	[Axis]
ExternalSignal	External signal	Enables external signals.	{"Name":"baseSystem","Later":"1.6","Earlier":"1.6"}, {"Name":"SignalIO","Later":"1.5"}, {"Name":"Axis","Later":"1.5"}	—
Program_ST	ST language execution	Add-on that executes ST language	{"Name":"baseSystem","Later":"1.6","Earlier":"1.6"}	[PLCopenMC_General]
Logging	Logging	Provides the logging function.	{"Name":"baseSystem","Later":"1.6","Earlier":"1.6"}, {"Name":"SignalIO","Later":"1.5"}	—
MotionEventHist	Motion event history	Add-on for motion event history.	{"Name":"baseSystem","Later":"1.6","Earlier":"1.6"}	—
FileTransfer	File transfer	Provides the file transfer function.	{"Name":"baseSystem","Later":"1.6","Earlier":"1.6"}	—
MotionControl_AxisFilter	Command filter	Enables PLCopen® motion control command filter FB.	{"Name":"baseSystem","Later":"1.6","Earlier":"1.6"}, {"Name":"MotionEngine","Later":"1.5"}, {"Name":"Axis","Later":"1.5"}, {"Name":"MotionControl_General","Later":"1.5"} {"Name":"MotionControl_Sync","Later":"1.5"}	—

Add-on library name	Name	Description	Dependent add-on (Necessary)	Dependent add-on (Restricted)
ProfileControl	Operation profile control	Enables operation profile control.	{"Name":"baseSystem","Later":"1.6","Earlier":"1.6"}, {"Name":"MotionEngine","Later":"1.5"}, {"Name":"Axis","Later":"1.5"}, {"Name":"MotionControl_General","Later":"1.5"} {"Name":"MotionControl_Sync","Later":"1.5"}	[SignalIO]
PackagingApp	Packaging	Provides functions for packaging machines.	{"Name":"baseSystem","Later":"1.6","Earlier":"1.6"}, {"Name":"ProfileControl","Later":"1.5"}	—
ServoSystemRecorder	Servo System Recorder	Provides the servo system recorder function.	{"Name":"baseSystem","Later":"1.8","Earlier":"1.8"}, {"Name":"Logging","Later":"1.7"}, {"Name":"Axis","Later":"1.7"}	[ServoDriver_CANOpen]

Point

baseSystem is necessary.

28 List of Boot software

The versions of each boot which is compatible with the software version of the Motion module are shown below.

Boot name	The software version of the Motion module							
	01	02	03	04	05	06	07	08
Network	1	2	2	2	3	3	3	4
VxWorks	1	1	1	2	3	3	3	4
IPL	1	1	1	1	1	1	1	1

MEMO

APPENDICES

Appendix 1 Connectable device to CC-Link IE TSN

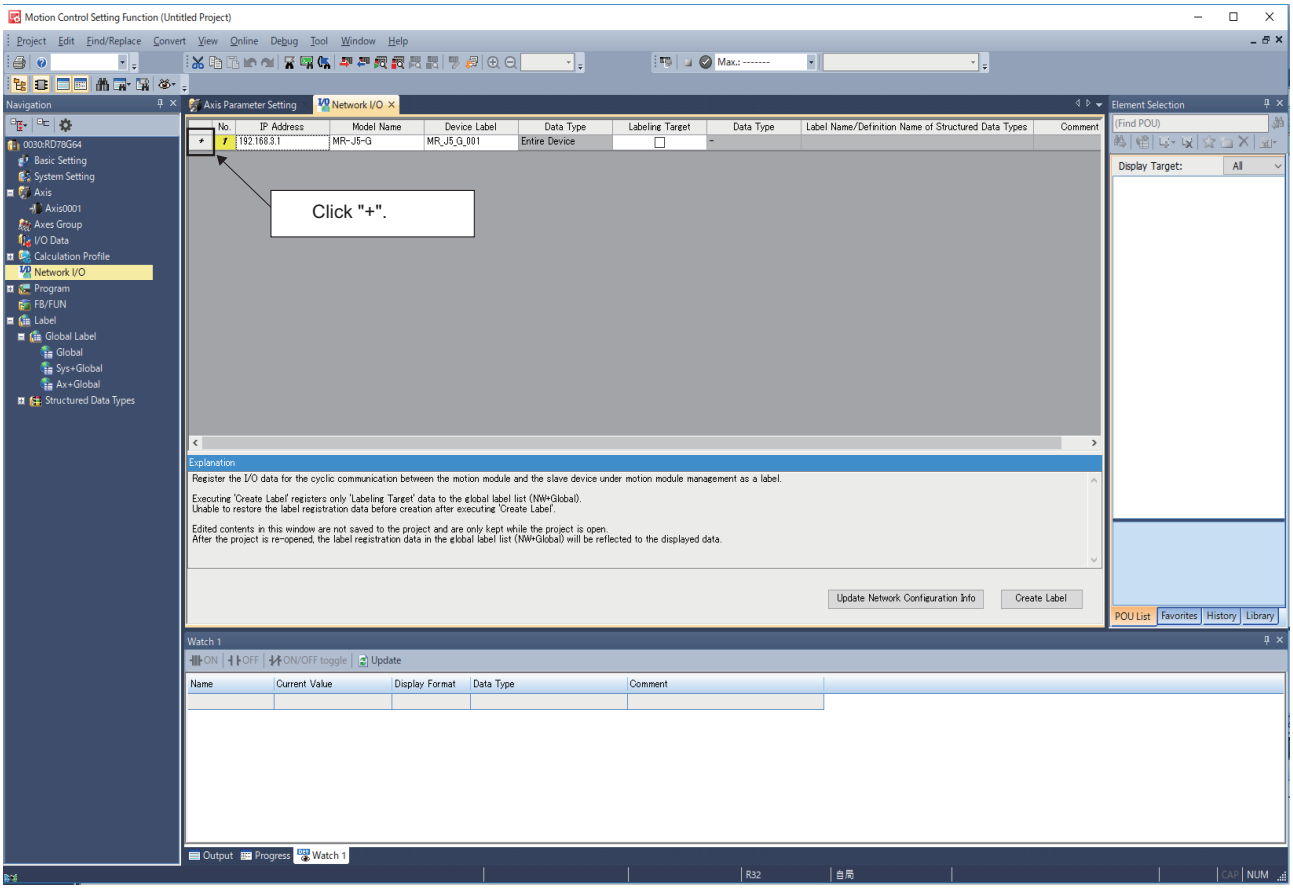
This section describes how to set and use the motion system when a connectable device to CC-Link IE TSN is connected with the Motion module.
For details on the wiring and the parameters, refer to the manual of each device.

Common Items for all devices

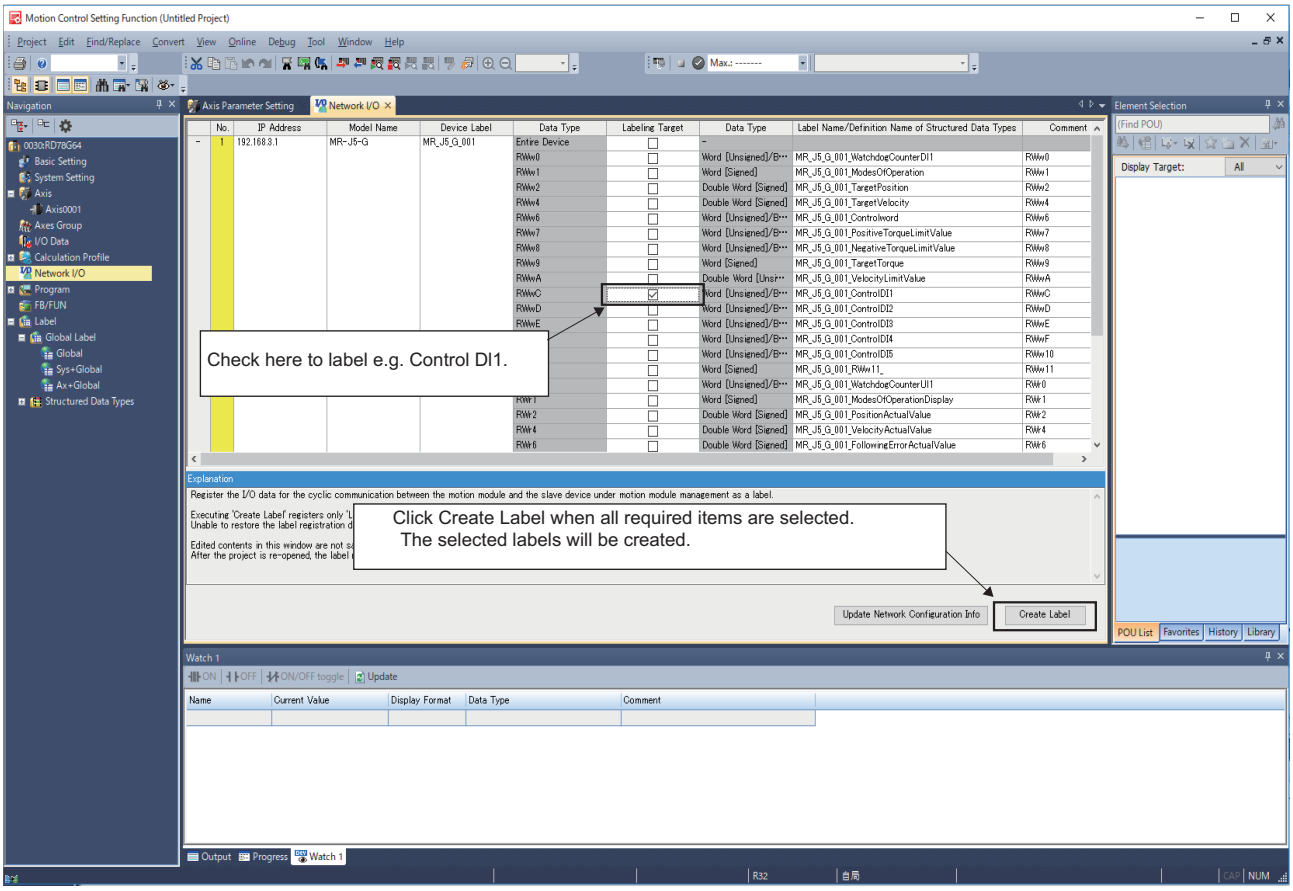
• How to use the device label

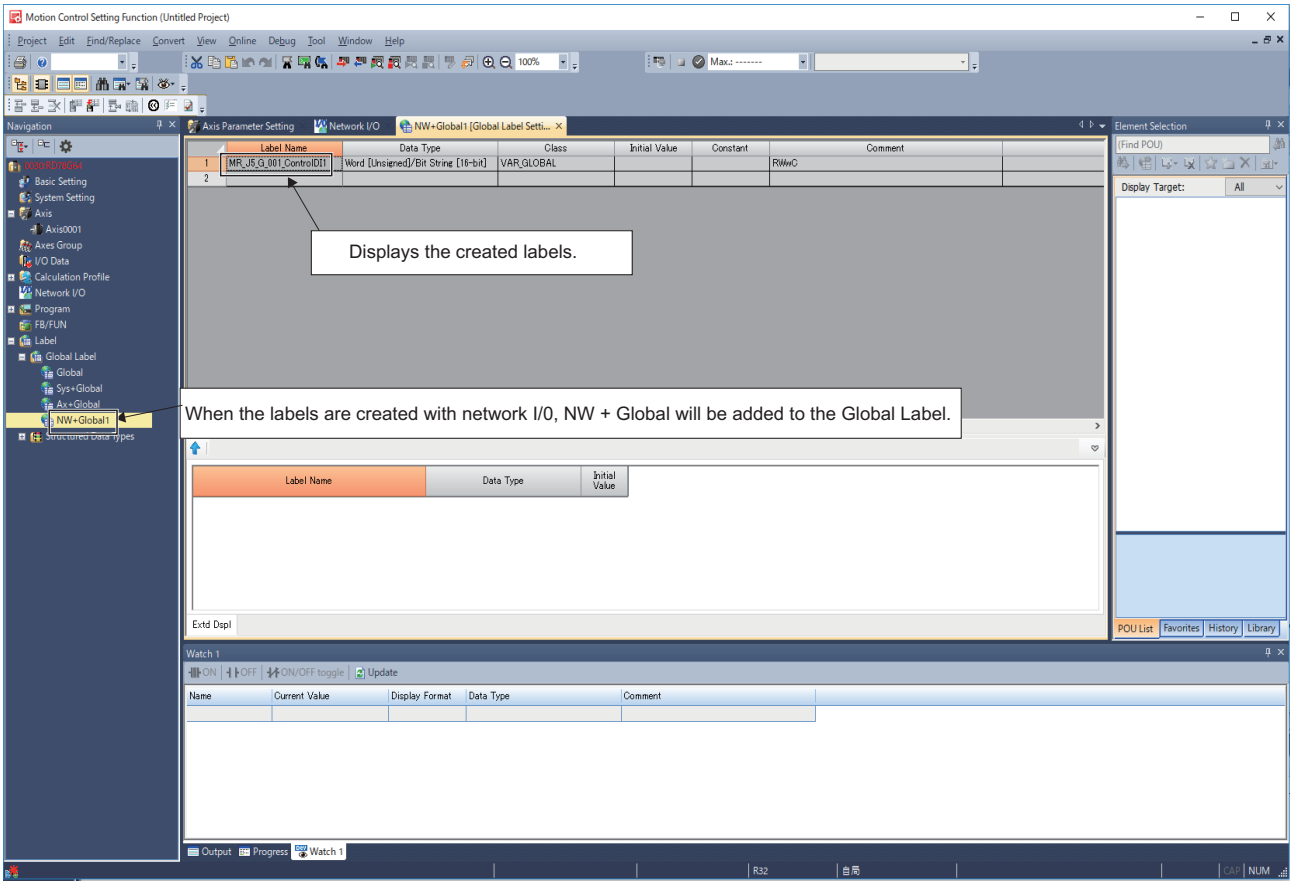
The method of using the device label is shown below. The following shows the MR-J5-G usage example and other device can also use it with the same method.

1. Click the left end "+" of the device, its label is generated, from the devices displayed on Network I/O screen.



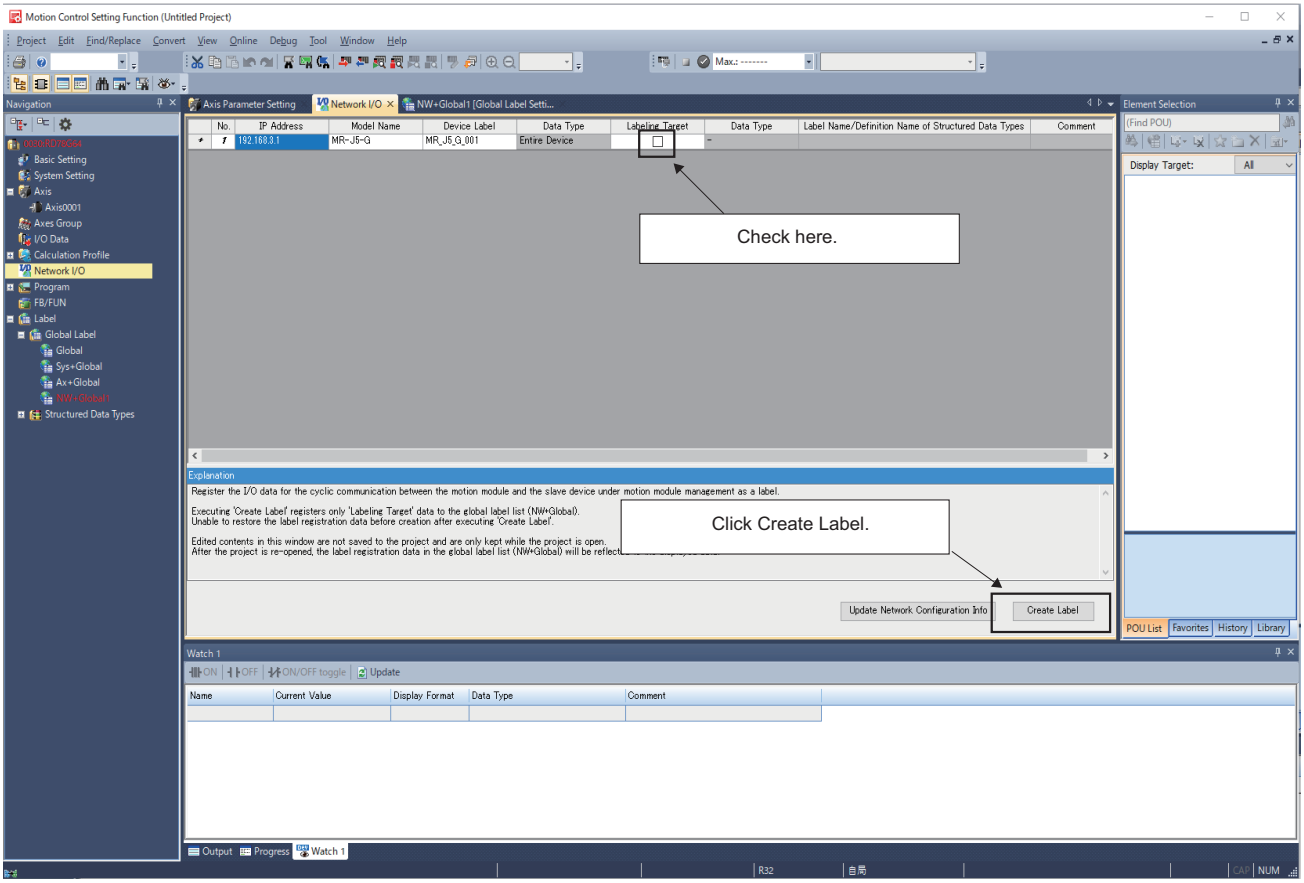
2. Set as the following to label each object.





A

3. Set as follows to label the whole device.



MR-J5(W)-G (Cyclic synchronous mode) connection method

This section describes how to set when connecting MR-J5(W)-G (Cyclic synchronous mode) and use various functions.
For details about wiring and parameters of MR-J5(W)-G, refer to MR-J5(W)-G manuals.

Setting methods

■Parameter setting value to use MR-J5(W)-G

Set the parameters of MR-J5(W)-G as below when executing motion control with MR-J5(W)-G.

No.	Name	Default value	Setting value
PA06	Electronic gear - Numerator	1	[MR-J5(W)-G (version A3) or earlier] Set 1 or 2^n when using for positioning control.*4 Set 1 when using for other than positioning control. [MR-J5(W)-G (version A4) or later] Set 1 or 2^n when using for positioning control.*4 Set as follows when using for other than positioning control. • If PT01.1 is "0h", the setting value is 1 and if PT01.1 is "1h", it is 1 or 2^n.
PA07	Electronic gear - Denominator	1	[MR-J5(W)-G version (version A3) or earlier] Set 1 or 2^n when using for positioning control.*4 Set 1 when using for other than positioning control. [MR-J5(W)-G version (version A4) or later] Set 1 or 2^n when using for positioning control.*4 Set as follows when using for other than positioning control. If PT01.1 is "0h", the setting value is 1 and if PT01.1 is "1h", it is 1 or 2^n.
PC29.5	[AL.0E3 Absolute position counter warning] selection	1h	0h (Available for infinite feed function.) (When using for absolute position system)
PT01.1*1	Velocity / acceleration/ deceleration unit selection	0h	Any setting value can be used for positioning control. For using other than positioning control, set 1h when PA06 and PA07 are other than 1 : 1 (velocity command unit: command unit/s, acceleration/deceleration unit: command unit/s^2) and it is arbitrary when they are 1 : 1.*2
PT15	Software position limit +	0	0*3
PT17	Software position limit -	0	0*3
PV23*1	Speed unit conversion electronic gear - numerator	1	1
PV24*1	Speed unit conversion electronic gear - denominator	1	1

*1 Available since MR-J5(W)-G (version A4)
*2 If "0h" is set for velocity / acceleration/deceleration unit selection, the Velocity actual value[Obj. 606Ch] may not be acquired correctly when other than 1 : 1 is set for electronic gear.
*3 This function is the same as the software stroke limit function of motion system. Therefore, use the function of motion system side to restrict the command position.
*4 Set 1 when the resolution of the encoder connected with MR-J5(W)-G is not multiple of two to the power of n.

For the following parameters, the setting values of the table below are recommended.

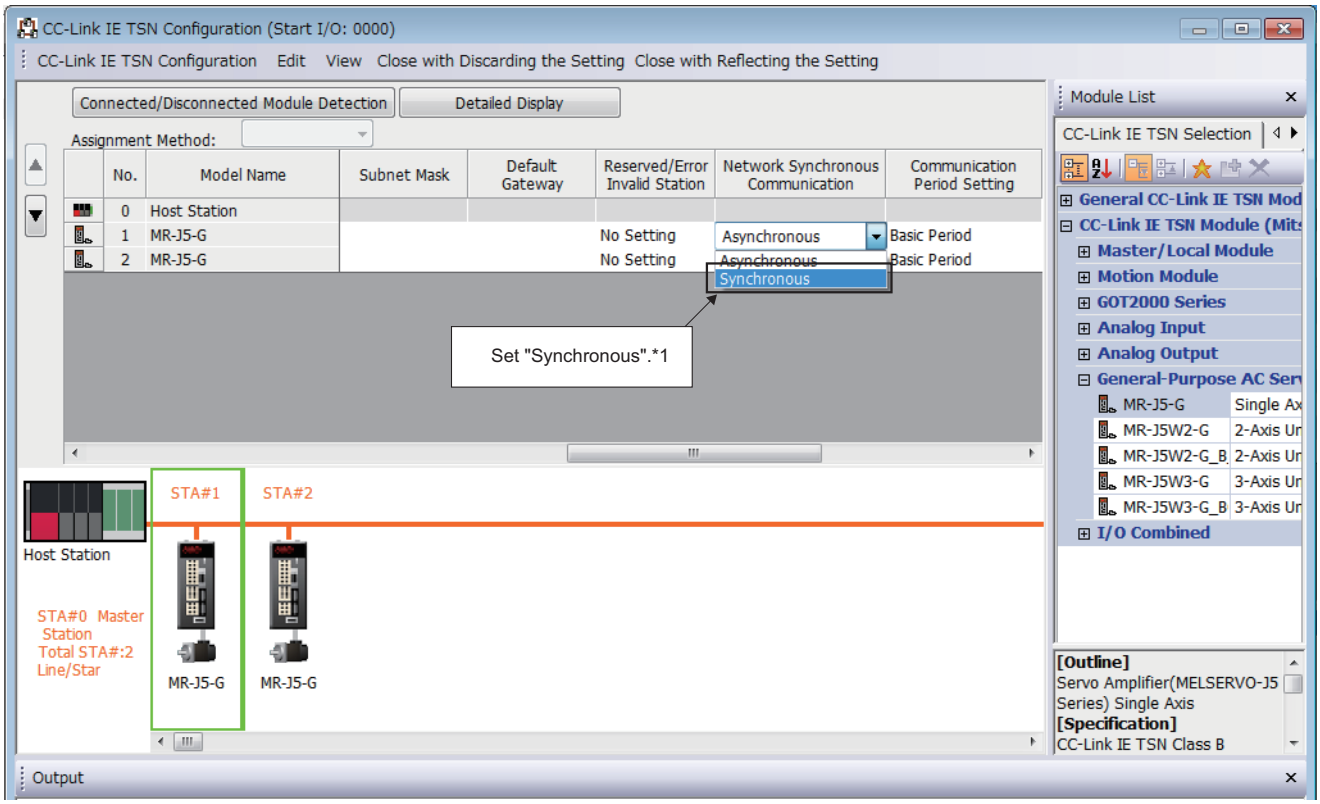
No.	Name	Default value	Overview
PD01	Input signal automatic on selection 1	00000000 1h	000000001h
PD41.2	Limit switch enabled status selection	0h	1h (Enabled only for homing mode)*1
PD41.3	Sensor input method selection	0h	1h (Input from controller (FLS/RLS/DOG))*2
PT29.0	Device input polarity 1	0h	1h: Dog detection with on (When PD41.3 is 1h (Input from the controller (FLS / RLS / DOG))

*1 When "0h" is set, the command of motion system is ignored and the motor stops as the limit switch signal is detected in MR-J5(W)-G side during control. Consider any of the following to stop the system safely.
Method 1:
Invalidates the detected signal of the MR-J5(W)-G side, and detects the limit by the hardware stroke limit function of the motion system side.
Method 2:
Turns AxisName.Pr.StopOption_DriverTargetIgnored valid to stop the command of the motion side when the motor is stopped due to the signal detection of the MR-J5(W)-G side.

*2 When "1h" is set, wire the limit switch to the controller side since the input to MR-J5(W)-G becomes invalid. In the case of wiring to MR-J5(W)-G, the motor does not stop if the limit is detected.
Also, the labeling of the device must not be carried out since [Control DI 5(2D05h)] of the RPDO mapping is used for sending signal to MR-J5(W)-G side.

■Network parameter setting to use MR-J5(W)-G

Set "Synchronous" in the network parameters setting when performing motion control with using MR-J5(W)-G.



*1 The setting, "Synchronous" is available from "Ver. 1.5" or later of baseSystem add-on.
Set "Asynchronous" when using "Ver.1.4" or earlier.

Using methods

■How to operate external signal of MR-J5(W)-G via communication

1. Preparation

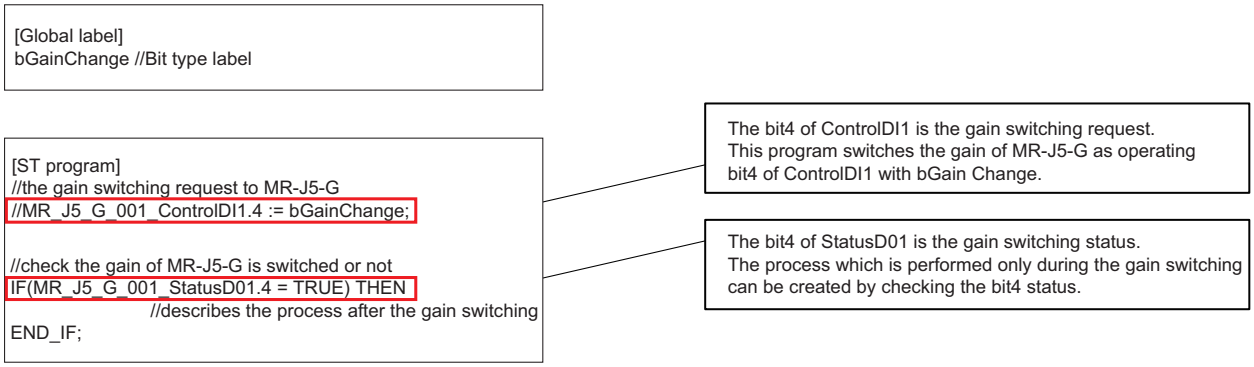
Generates the required objects of the device label according to the procedures described in common items for all devices.
External signals of MR-J5(W)-G are assigned to the objects below.

Object	
Input	Control DI1 to Control DI10
Output	Status DO1 to Status DO10

Also, when using functions of each external signal, some parameters should be changed. For details, refer to MR-J5(W)-G manuals.

2. Using methods

Operating / referring external signals of MR-J5(W)-G are allowed by changing / referring the values of generated labels from the program or watch.
A sample program to carry out the gain switching by using the device label is shown below. (Generate the label previously as Control DI1 and Status DO1 are used for the gain switching.)



A

■The connection with MR-J5(W)-G in high-speed mode

For details on the high-speed mode, refer to the following.

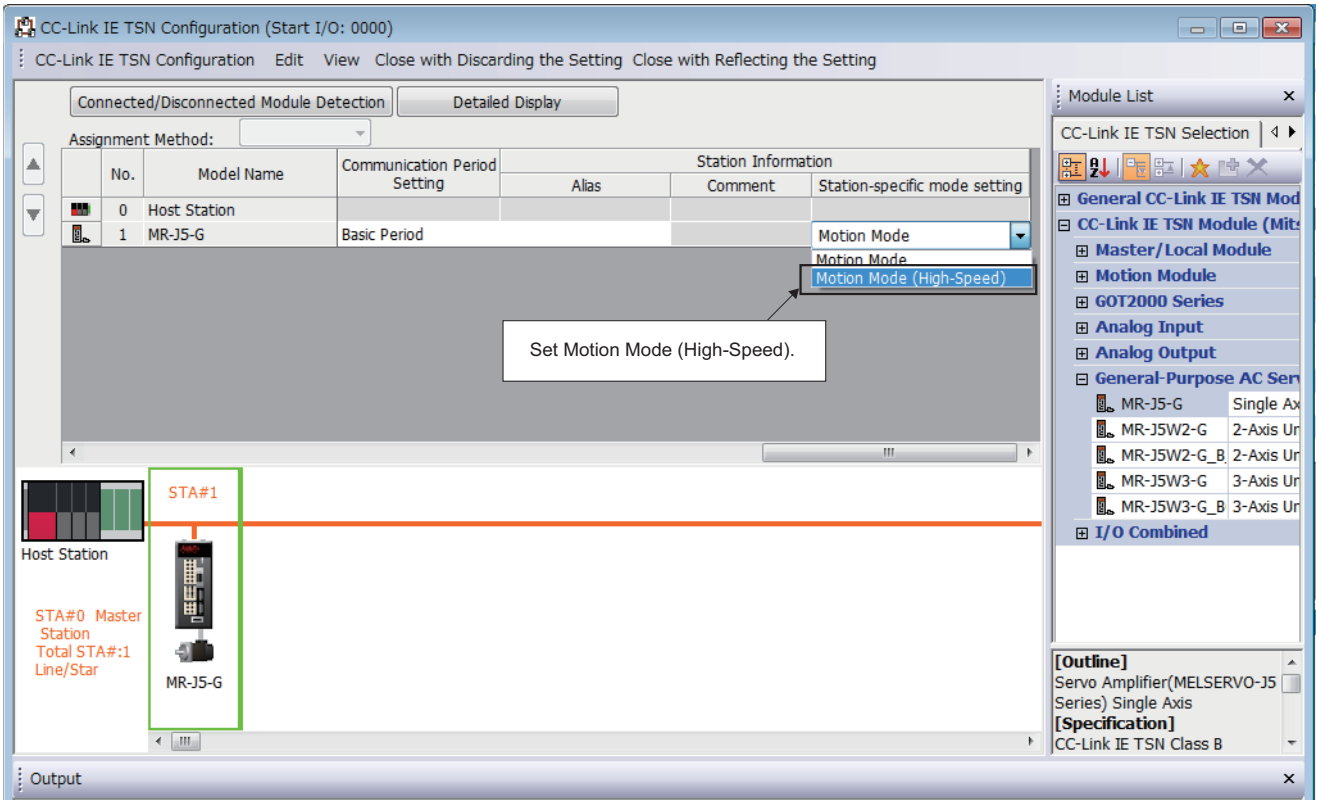
☞ Page 109 Operation Cycle

1. Preparation

[Setting of CPU [GX Works3]]

[Module Parameter (Network)]⇨[Basic settings]⇨[Network Configuration Settings]

Set the station-specific mode to the Motion Mode (High-Speed) on the "CC-Link IE TSN Configuration" screen.



When the Motion Mode (High-Speed) is set, PDO mapping is as follows.

• RPDO

No.	Object name
1	Watch dog counter DL
2	Modes of operation
3	Target position
4	Controlword
5	Control DI4

• TPDO

No.	Object name
1	Watch dog counter UL
2	Modes of operation display
3	Position actual value
4	Statusword
5	Status DO1
6	Status DO4
7	Torque actual value
8	Velocity actual value
9	Current alarm

[Motion system setting [Engineering tool]]

Set to the high-speed mode. For details, refer to the following.

☞ Page 109 Operation Cycle

[MR-J5(W)-G setting [MR Configurator2]]

When connecting with 31.25 μs of communication cycle interval setting, set "High speed mode selection" (PA01.7) to "1h (Enabled)". When connecting with other than 31.25 μs, no specific setting is required.

2. Using method

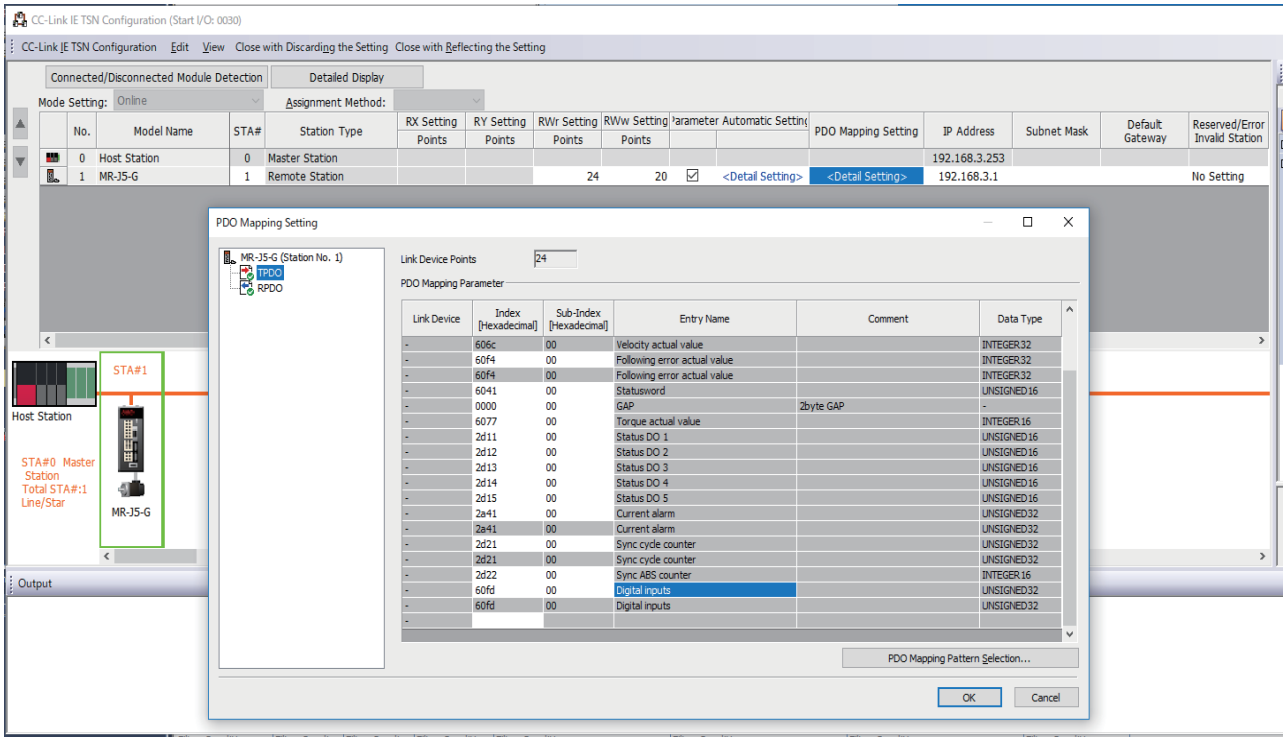
Communication in high speed mode is allowed when MR-J5(W)-G is connected with motion system in the above setting.

■The method for setting (LSP/LSN/DOG) input into MR-J5(W)-G to the upper limit/lower limit and home position switch of the motion system

(LSP/LSN/DOG) input signal of the servo amplifier side can be used as the FLS/RLS, and home position switch signal of the motion system by assigning the object, [Digital inputs(60fdh)] to TPDO mapping of MR-J5(W)-G and creating the device label.

1. Map [Digital inputs(60fdh)] to TPDO.

Add Digital inputs by specifing index: 60fdh and subindex: 00h to TPDO of the target amplifier in Network Configuration Settings on the engineering tool.



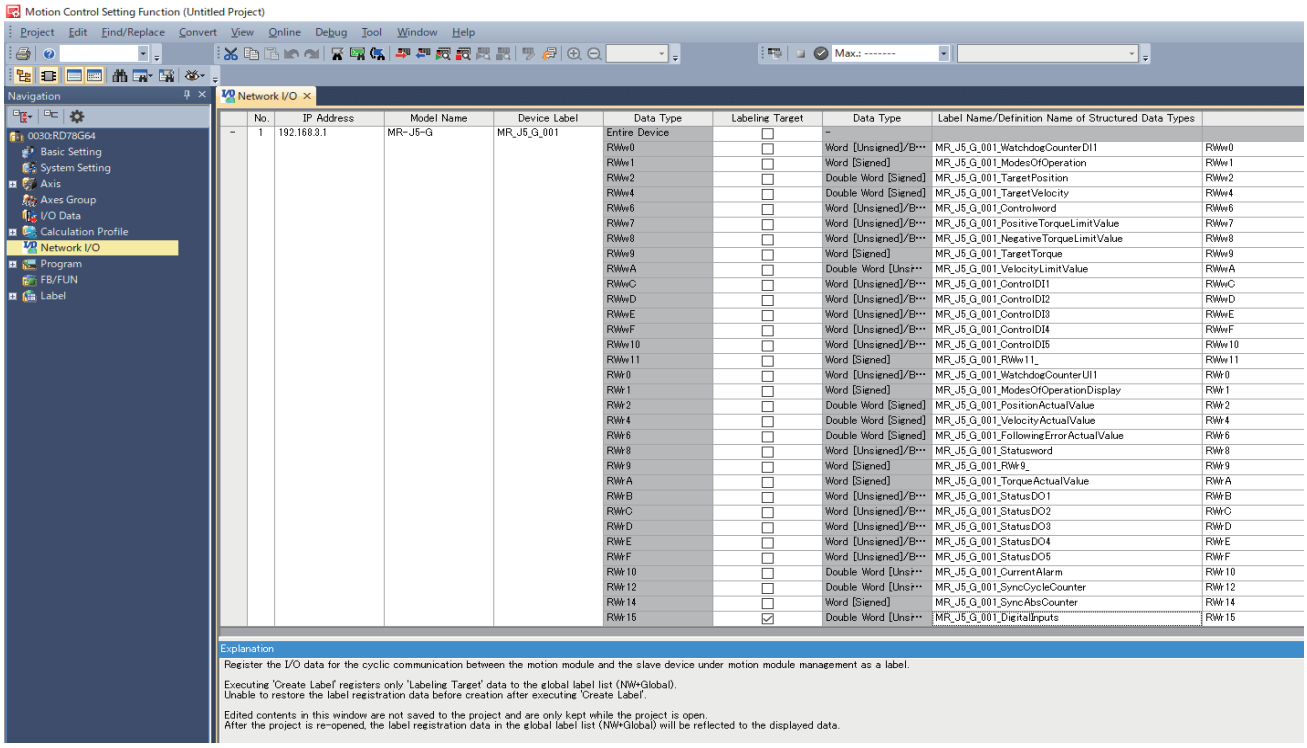
2. Make [Digital inputs(60fdh)] the device label.

For the method of the labeling, refer to the following.

☞ Page 728 Common Items for all devices

Ex.

When the labeling is performed with the name of "MR_J5_G_001_DigitalInputs".

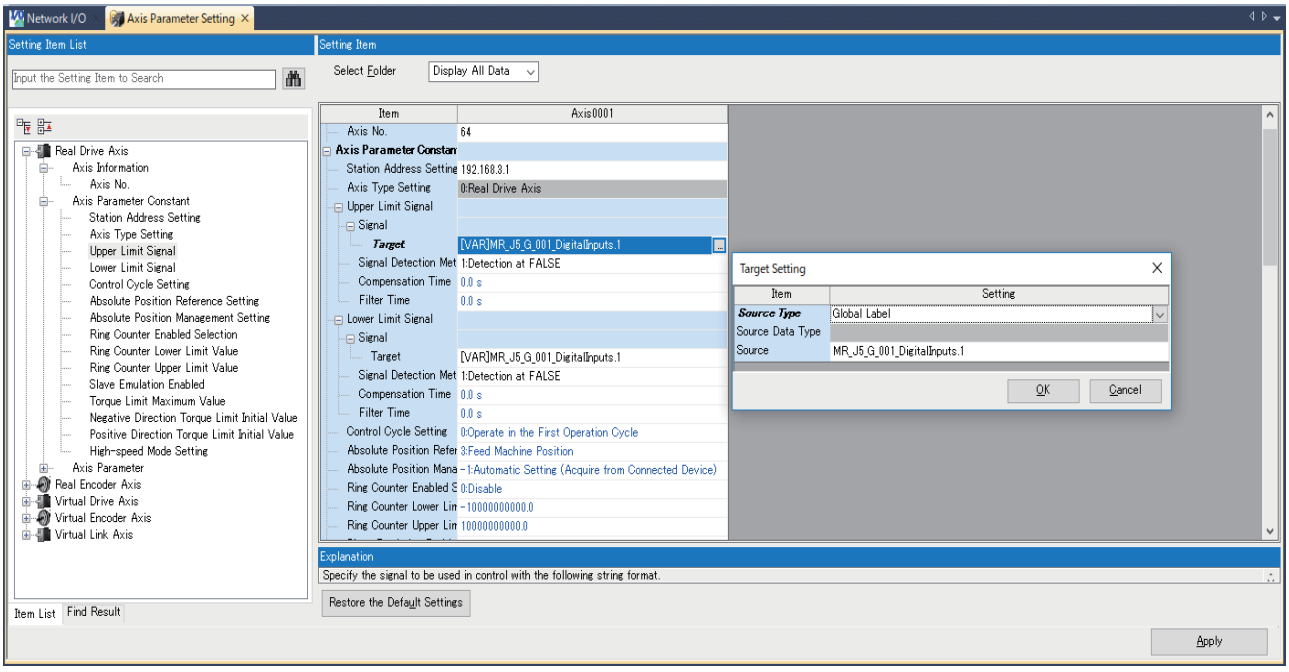


3. Specify the device label to the home position switch of FLS/RLS or MC_Home.

[The upper limit/lower limit setting]

Set the upper limit signal target/lower limit signal target of the axis parameter.

Variable	Setting value (when PA14 is 0)
AxisName.PrConst.HwStrokeLimit_FlsSignal.Source.Target	[VAR]MR_J5_G_001_DigitalInputs.1
AxisName.PrConst.HwStrokeLimit_RlsSignal.Source.Target	[VAR]MR_J5_G_001_DigitalInputs.0



[Specify the DOG signal at homing]

Specify the following to "home position switch" input into MC_Home.

Variable	Setting value (when PA14 is 0)
MC_INPUT_REF.Signal.Source.Target	[VAR]MR_J5_G_001_DigitalInputs.2

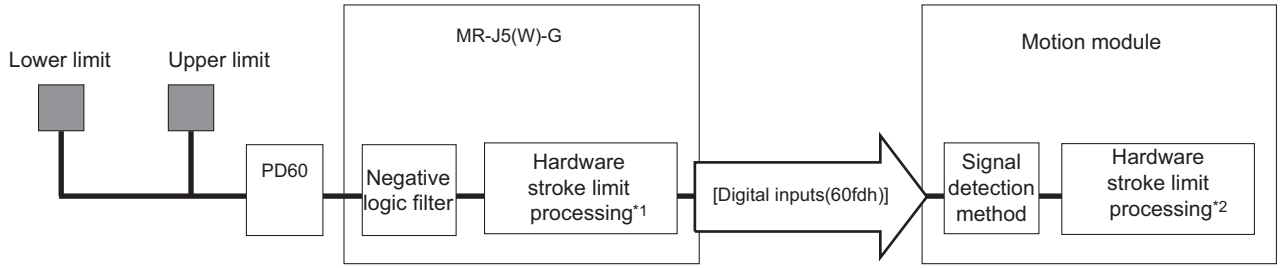
- This setting is not required if "0h: Input from servo amplifier (LSP/LSN/DOG)" is selected in "Sensor input method selection" (PD41.3), input the DOG signal to MRJ5(W)-G directly.
- Set "1: Dog detection with on" to "Device input polarity 1" (PT29.0) if "1h: Input from controller (FLS/RLS/DOG)" is selected in "Sensor input method selection" (PD41.3).

Precautions

- Assign LSP/LSN/DOG signal to "Input device selection 1 to 3" (PD03 to 05) of MR-J5(W)-G side to use the signal of the motion system.
- The polarities of Bit0(Negative limit switch) and Bit1(Positive limit switch) of the object, [Digital inputs(60fdh)] are changed according to the setting of "Travel direction selection" (PA14). For details, refer to the manuals of each drive unit.
- Set the following for signal detection method of the motion system side to be the same polarity of Bit0 (Negative limit switch) and Bit1 (Positive limit switch) of object [Digital inputs(60fdh)] of MR-J5(W)-G. If the setting is different, unintended operation may occur.

Variable	Setting value
AxisName.PrConst.HwStrokeLimit_FlsSignal.Detection	1: LowLevel (Detection at FALSE)
AxisName.PrConst.HwStrokeLimit_RlsSignal.Detection	

- The data flow of object [Digital inputs(60fdh)] is shown below when "0h: Input from servo amplifier (LSP/LSN/DOG)" is selected in "Sensor input method selection" (PD41.3).



Signal input	MR-J5(W)-G			→	[Digital inputs(60fdh)] Bit0 and Bit1	→	Motion module	
	PD60	External input signal logic	Hardware stroke limit error detection*1				External input signal logic setting	Hardware stroke limit error detection*2
ON	0	Negative logic	Not detect	→	TRUE	→	(1: LowLevel (Detection at FALSE))	Not detect
	1		Detect		FALSE			Detect
OFF	0		Detect		FALSE			Detect
	1		Not detect		TRUE			Not detect
ON	0		Not detect	→	FALSE	→	(0: HighLevel (Detection at TRUE))	Not detect
	1		Detect		TRUE			Detect
OFF	0		Detect		TRUE			Detect
	1		Not detect		FALSE			Not detect

*1 When "Limit switch enabled status selection" (PD41.2) is "0: Limit switch always enabled", the error stop is executed in the servo amplifier side not even during homing.

*2 The error stop is executed even in the Motion module during homing. The hardware stroke limit error detection in homing can be temporarily disabled by setting AxisName.Cd.HwStrokeLimit_Override.

Precautions

- When labeling of data sent from the motion system is performed, the send data is overwritten with the label values. Labeling of only necessary objects should be performed because the motion control cannot be performed for a part of the objects such as Target position by the labeling.

The following table shows whether the labeling of the mapped objects in the initial state is possible or not.

○: labeling possible, ×: labeling not possible, △: restrictions apply

Index	SubIndex	Object name	Labeling possible/not possible
1D01h	1	Watchdog counter DL	×
6060h	0	Modes of operation	×
607Ah	0	Target position	×
60FFh	0	Target velocity	×
6040h	0	Controlword	×
60E0h	0	Positive torque limit value	×
60E1h	0	Negative torque limit value	×
6071h	0	Target torque	×
2D20h	0	Velocity limit value	×
2D01h	0	Control DI 1	○
2D02h	0	Control DI 2	○
2D03h	0	Control DI 3	○
2D04h	0	Control DI 4	×
2D05h	0	Control DI 5	△*1

*1 When "Sensor input method selection" (PD41.3) is set to "1h: Input from controller (FLS/RLS/DOG)", the input of FLS and RLS from the motion system can not be sent to the servo amplifier. For making it the device label, change to "0h: Input from servo amplifier (LSP/LSN/DOG)" in "Sensor input method selection" (PD41.3).

- When parameter automatic setting is enabled in the network configuration settings on the engineering tool, according to the communication load condition, the changed parameters might not be reflected in the CPU module as the parameter is changed for the multiple stations at the same time in the communication on MR Configurator2. Change them one by one. Otherwise, carry out writing of the project into the CPU module after starting MR Configurator2 and changing the parameters via GX Works3.
- A servo alarm [AL.035_Command frequency error] may be detected in MR-J5(W)-G when an operation cycle over occurs in the motion system during a motor operation and the command differs greatly before the operation cycle over and after a restoration. Check the program and increase the operation cycle setting or decrease the loading if necessary.
- When connecting MR-J5(W)-G in communication cycle 31.25 μs/62.5 μs, use A6 version or later. If its version is not compatible, the error "Communication Cycle Unsupported Driver Connection" (error code: 1C47H) will occur at connection and it can not be connected with the driver.

MR-J5(W)-G (Profile mode) connection method

This section describes how to set when connecting MR-J5(W)-G (Profile mode) and use various functions.

For details on wiring and parameters of MR-J5(W)-G, refer to MR-J5(W)-G manuals.

The firmware for MR-J5(W)-G which is available with profile mode is shown below.

Model name	Version
MR-J5(W)-G	A4 or later

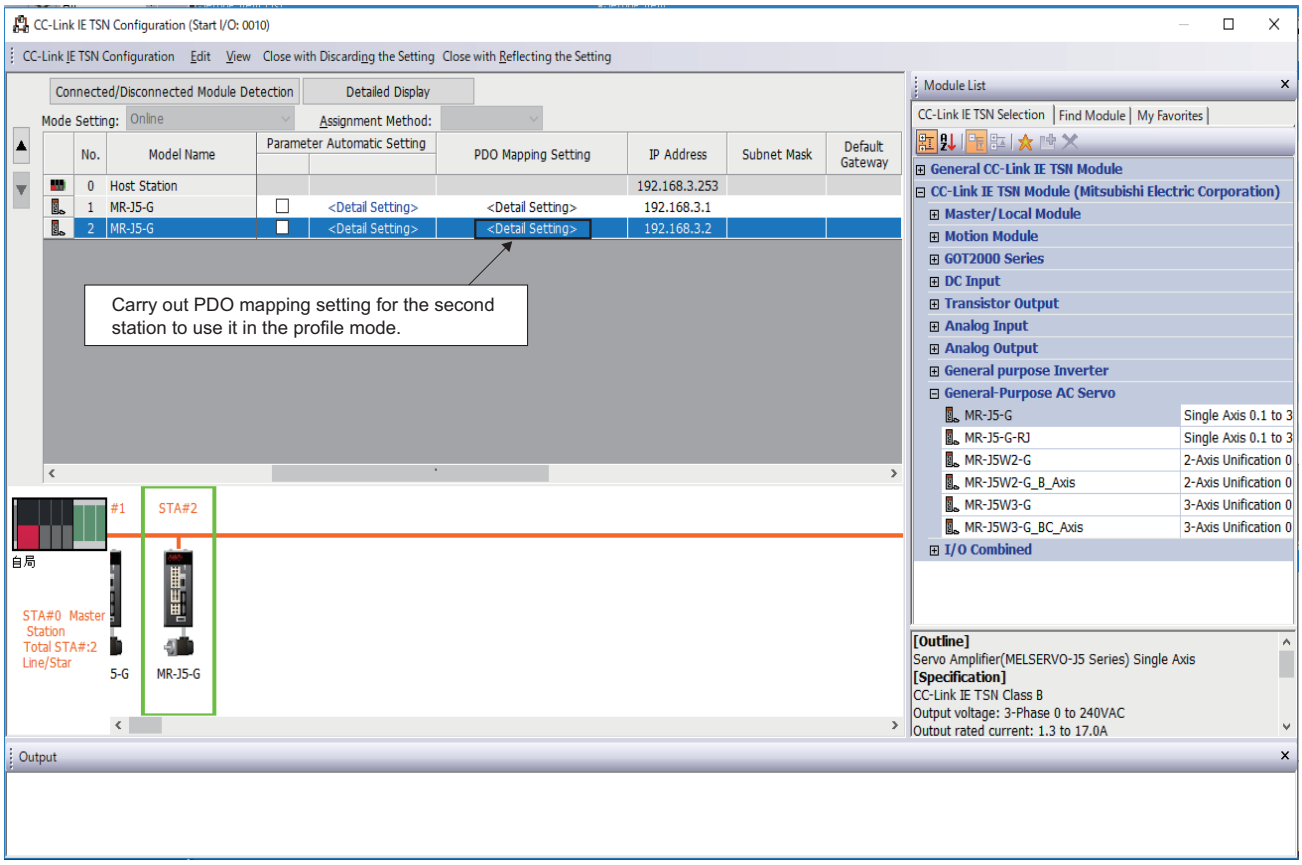
Setting method

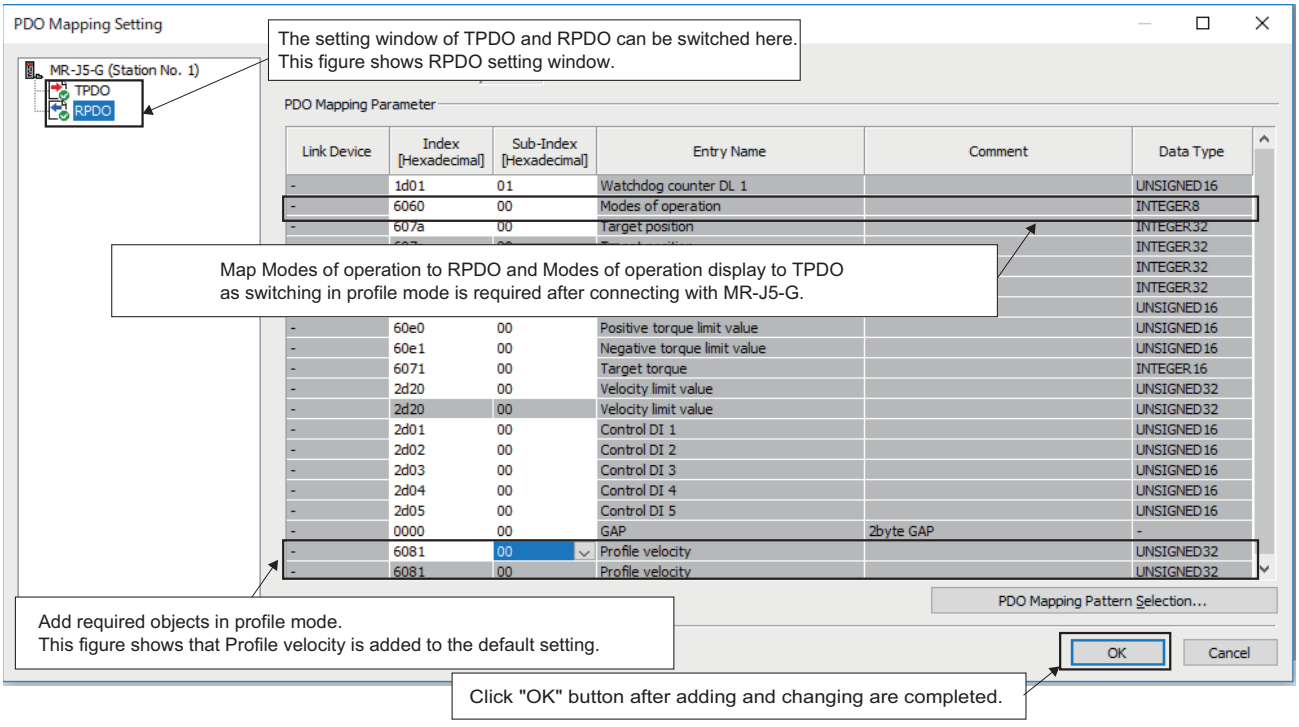
An example is shown below for using two MR-J5-G as the first station (192.168.3.1) set with the cyclic synchronous mode and the second station (192.168.3.2) set in profile mode.

■CPU setting [GX Works3]

[Module Parameter (Network)]⇨[Basic Settings]⇨[Network Configuration Settings]

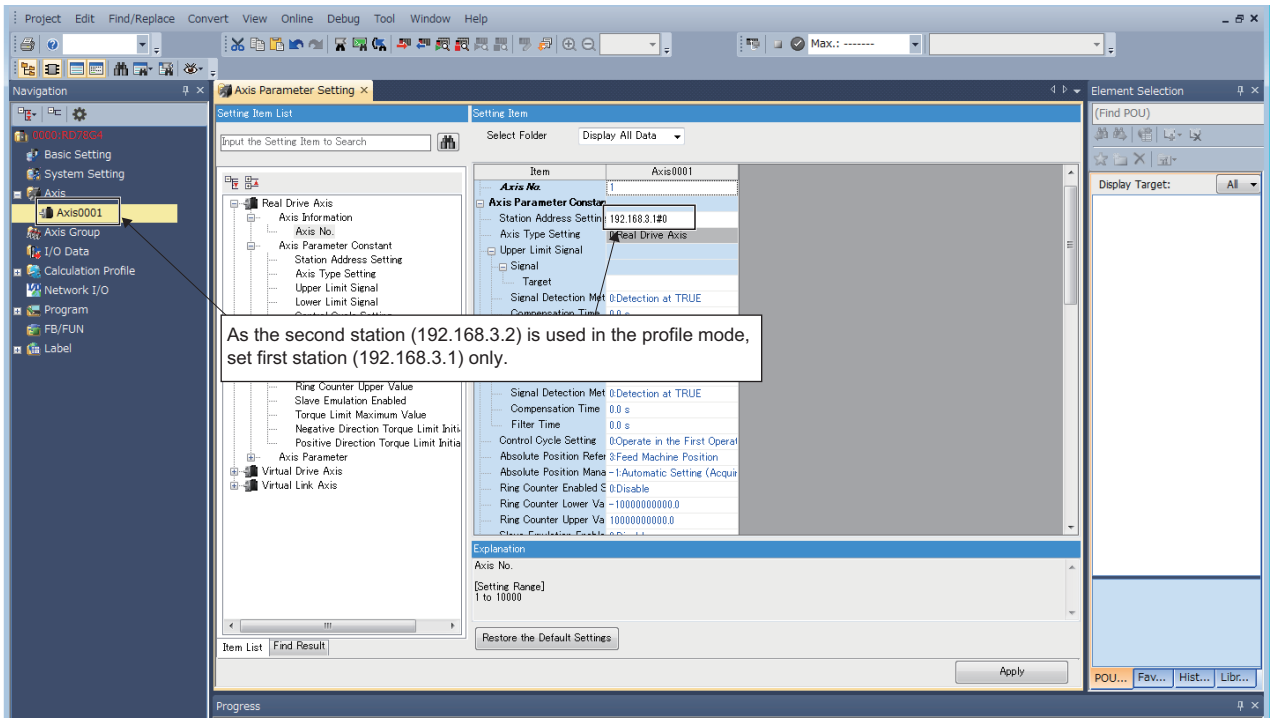
Set PDO mapping and map required objects in profile mode on the "CC-Link IE TSN Configuration" screen.



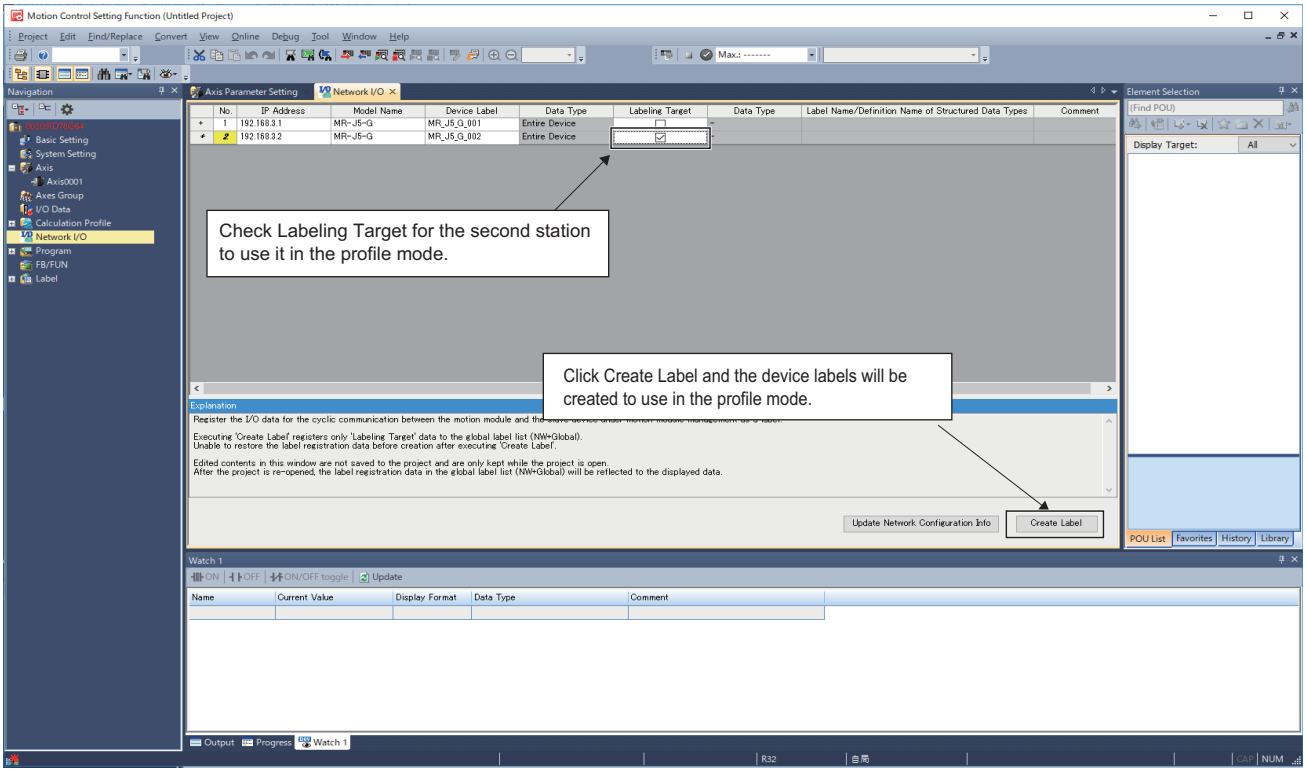


■The motion system setting [Motion control setting function]

The axis to be used in profile mode must not be set.



Generate the device label of MR-J5-G using for the profile mode according to the procedure described in "Common Items for all devices". For using with the profile mode, generate labels of the entire device as the labeling target.

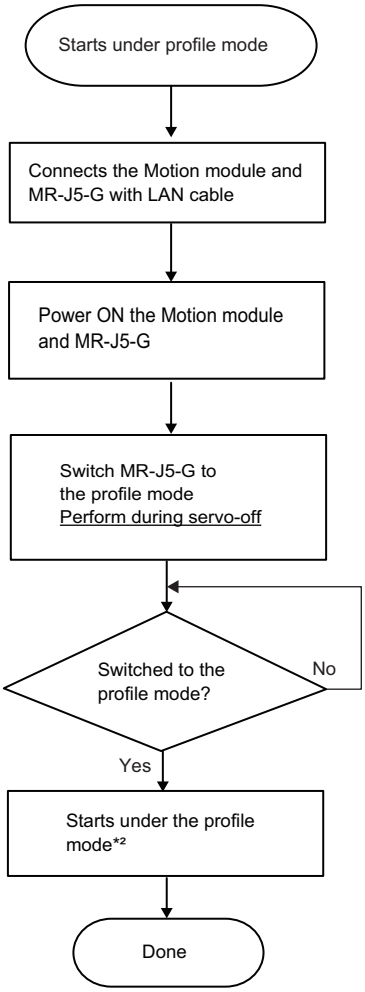


■MR-J5-G setting [MR Configurator2]

No setting is required.

Setting method

The following figure describes the process for driving motors with the profile mode.



MR-J5-G starts with csp mode.
Change MR-J5-G to the profile mode by modifying MR_J5_002_Modes OF Operation*1 from program or watch.
(e.g.: Set the label to "1" for changing to pp mode)

Refer to MR_J5_002_Modes OF OperationDisplay*1 and check the control mode of MR-J5-G is switched.

*1 A label name differs depending on a labeled device. The name described in this process indicates the name for generating the label targeting the second station of MR-J5-G.

*2 When Watch dog counter DL is mapped to the PDO, creating the program to update Watch dog counter DL is required for each communication cycle. (When updating is not performed, the motors do not drive.)
When it is not needed, exclude it from mapping.
When using Watch dog counter DL, set as follows in the program.
Method 1:
Turn bit15 ON.
Method 2:
Add 1 for each communication cycle. (Be sure not to be turned OFF bit15 by adding 1 as bit15 has to be turned ON constantly.)

Precautions

- Do not execute servo-on before switching to the profile mode. An improper operation, such as sudden acceleration of the motors, might occur.
- Do not switch to the cyclic synchronous mode after switching to the profile mode. An improper operation, such as sudden acceleration of the motors, might occur.
- When using with the profile mode, the motion system does not perform the limit check of the command, etc. Carry out safety measures at the user's program or MR-J5(W)-G side.
- Do not switch MR-J5(W)-G set as the axis to the profile mode. In the profile mode and the cyclic synchronous mode after being switched from profile mode again, an improper operation might occur.
- When parameter automatic setting is enabled in the network configuration settings on the engineering tool, according to the communication load condition, the changed parameters might not be reflected in the CPU module as the parameter is changed for the multiple stations at the same time in the communication on MR Configurator2. Change them one by one. Otherwise, carry out writing of the project into the CPU module after starting MR Configurator2 and changing the parameters via GX Works3.
- When connecting MR-J5(W)-G in communication cycle 31.25 μs/62.5 μs, use A6 version or later. If its version is not compatible, the error "Communication Cycle Unsupported Driver Connection" (error code: 1C47H) will occur at connection and it can not be connected with the driver.

How to connect MR-JET-G

This section describes how to set when connecting MR-JET-G and use various functions.
For details on wiring and parameters of MR-JET-G, refer to MR-JET-G manuals.

Setting method

The same as MR-J5(W)-G

How to use

The same as MR-J5(W)-G

Precautions

Do not set the following objects as MR-JET-G is not compatible with them.

Index	SubIndex	Object name	Parameter name
2D35h	2	Encoder status 2	AxisName.PrConst.SlaveObject.vEncoderStatus2
2D36h	0	Scale cycle counter	AxisName.PrConst.SlaveObject.vScaleCycleCounter
2D37h	0	Scale ABS counter	AxisName.PrConst.SlaveObject.vScaleAbsCounter
2D38h	0	Scale measurement encoder resolution	AxisName.PrConst.SlaveObject.vScaleMeasurementEncoderResolution
2D3Ch	0	Scale measurement encoder reception status	AxisName.PrConst.SlaveObject.vScaleMeasurementEncoderReceptionStatus

- MR-JET-G is not compatible with the communication with 31.25 μs and 62.5 μs. Set the communication period interval setting with 125 μs or more.
- When parameter automatic setting is enabled in the network configuration settings on the engineering tool, according to the communication load condition, the changed parameters might not be reflected in the CPU module as the parameter is changed for the multiple stations at the same time in the communication on MR Configurator2. Change them one by one. Otherwise, carry out writing of the project into the CPU module after starting MR Configurator2 and changing the parameters via GX Works3.
- A servo alarm [AL.035_Command frequency error] may be detected in MR-JET-G when an operation cycle over occurs in the motion system during a motor operation and the command differs greatly before the operation cycle over and after a restoration. Check the program and increase the operation cycle setting or decrease the loading if necessary.

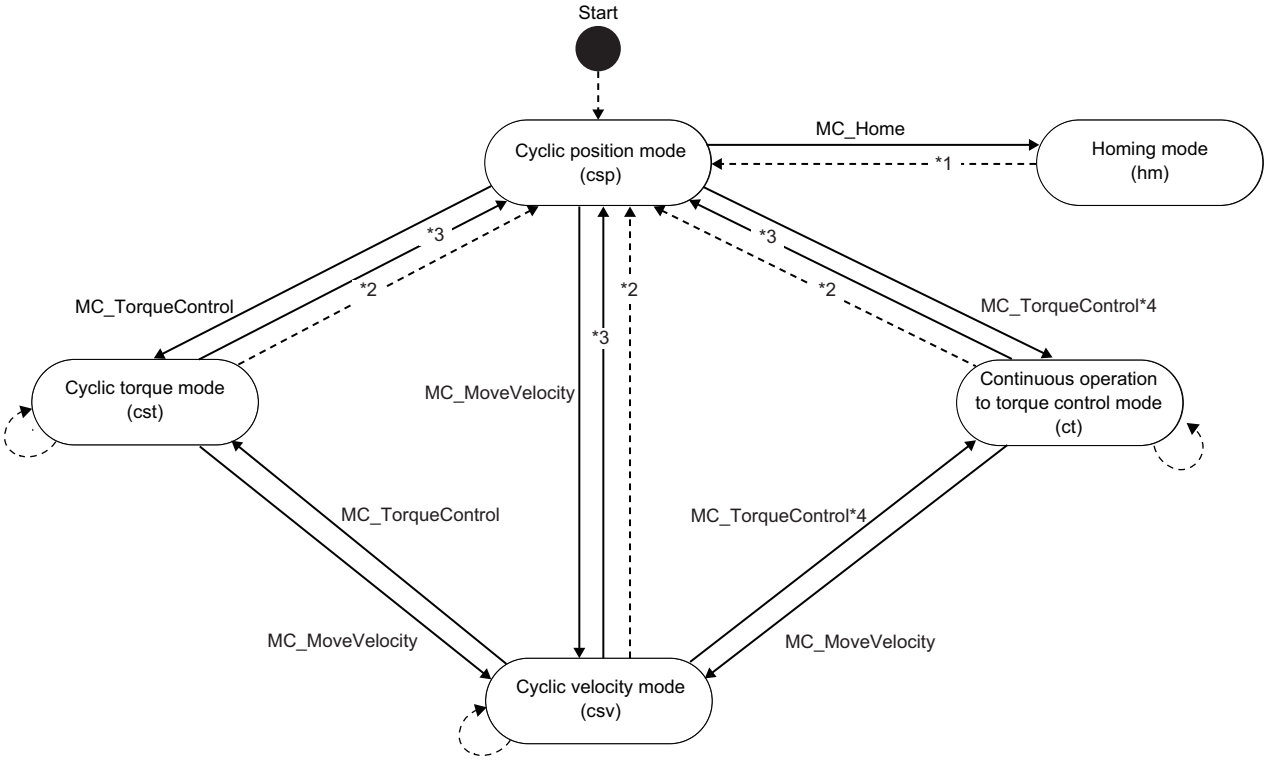
Relevant functions

Driver control mode

The motion system supports the following control modes.

Supported control mode		Description	Parameter name
Homing mode (hm)	6: hm	Executes homing operation by a driver.	Page 202 Driver homing method
Cyclic position mode (csp)	8: csp	Executes control following the sequential positioning command from the controller in each communication cycle.	7 DIRECT CONTROL
Cyclic velocity mode (csv)	9: csv	Executes control following the velocity command from the controller in each communication cycle.	
Cyclic torque mode (cst)	10: cst	Executes control following the torque command from the controller in each communication cycle.	
Continuous operation to torque control mode (ct)	-104: ct	A mode to switch to the continuous operation to torque operation smoothly without stopping from the cyclic position mode or the cyclic velocity mode.	Page 745 Continuous operation to torque control mode

Control mode switching of the driver is executed at the same time when MCFB is executed. The state transition is shown below.



*1 Transits after the axis is stopped by homing completion or error occurrence.
*2 Transits at stop completion or error occurrence.
*3 Transits when a positioning control FB other than MC_MoveVelocity/MC_TorqueControl is being Aborting/Buffered.
*4 For continuous operation to torque control mode (ct) switch, refer to the following.
☞ Page 745 Continuous operation to torque control mode

Precautions

When the control mode does not switch within a second after requesting the control mode switching to the driver, the error "Control Mode Switching Error" (error code: 1A1DH) occurs and the axis stops.

Continuous operation to torque control mode

■Overview

For the drive unit which supports the continuous operation to torque control mode, when MC_TorqueControl is started with selecting "1: continuous operation to torque control mode" by Options bit19, the driver control mode can be switched to the continuous operation to torque control mode (ct) and the driver can be controlled.

The continuous operation to torque control mode is a mode to switch to the continuous operation to torque operation smoothly without stopping from the cyclic position mode in positioning or the cyclic velocity mode in velocity command.

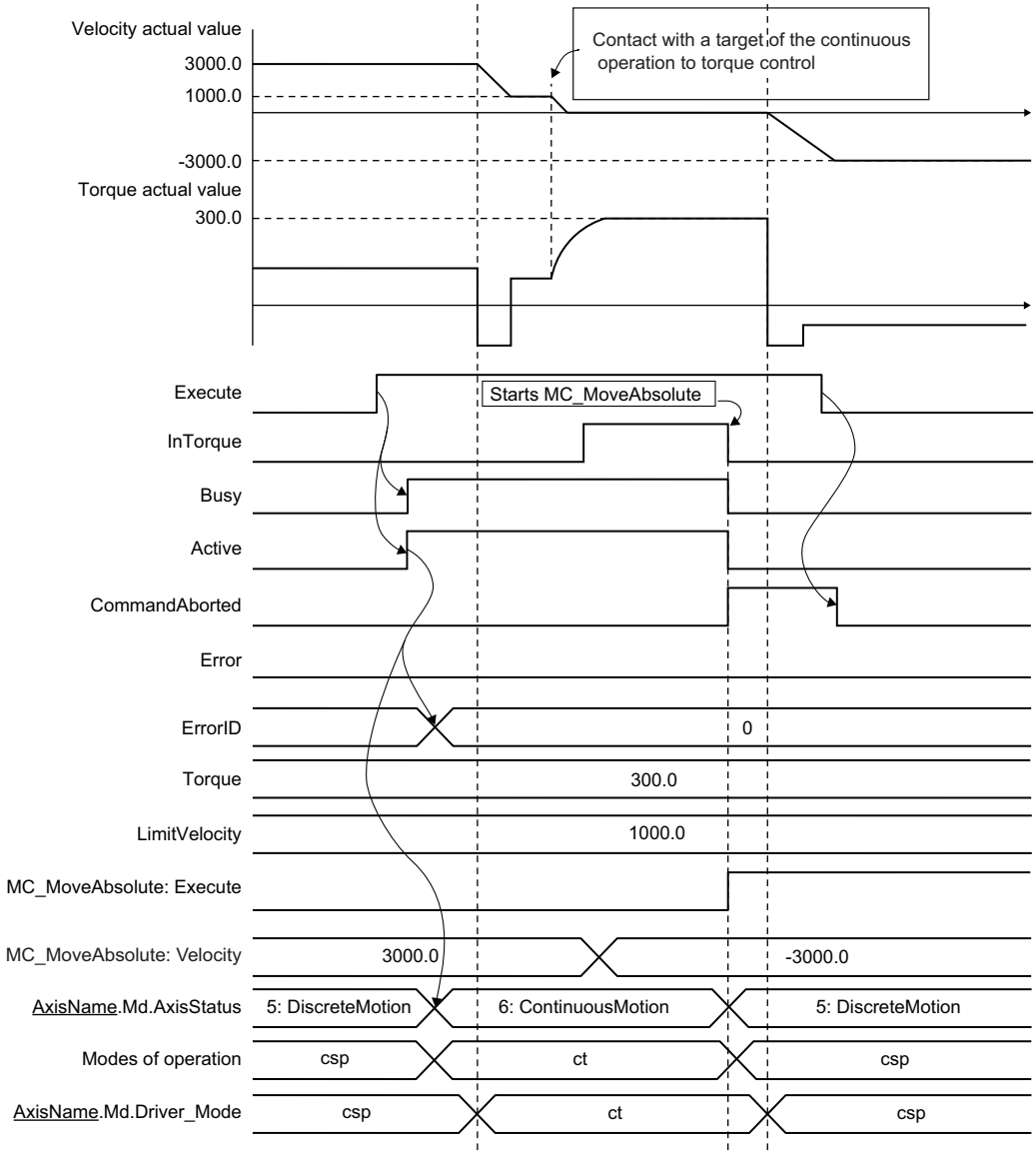
In the continuous operation to torque control mode, the driver executes the torque control with commanded torque set to Torque as accelerating or decelerating to the velocity set to LimitVelocity.

To finish this FB, start MC_Stop.

■Timing chart

[When the FB is normally completed]

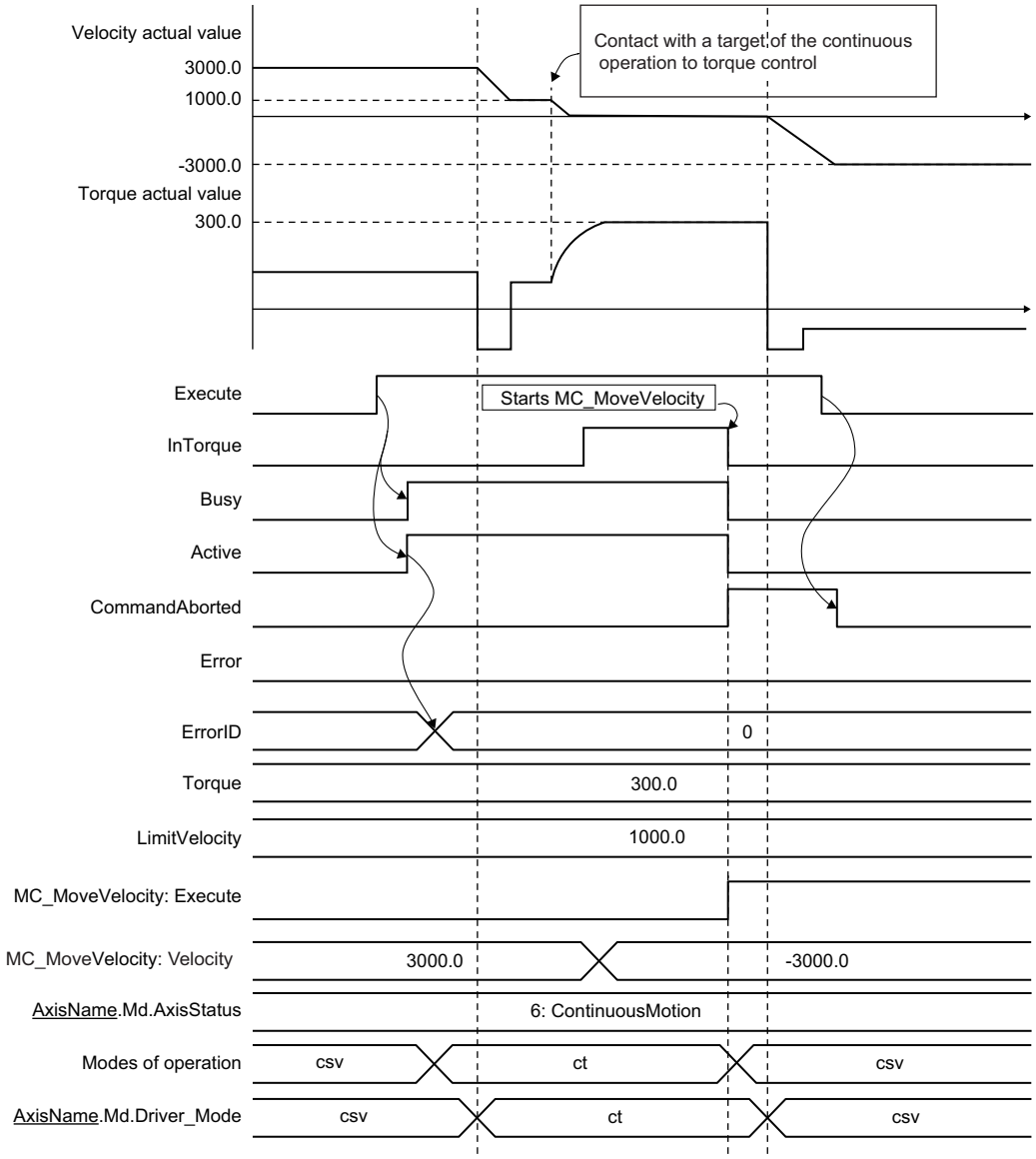
- Operation for "cyclic position mode ⇔ continuous operation to torque control mode switching"



Point

Set "1: actual velocity" to Options bit20 and bit21 to switch to the continuous operation to torque control mode when there is a difference between set velocity and actual velocity such as accelerating and decelerating or the velocity does not reach to the set velocity due to the limited torque.

• Operation for "cyclic velocity mode ⇔continuous operation to torque control mode switching"



Point

A torque command value in the continue operation to torque control mode will be disabled when it is switched from the continuous operation to torque control mode to the cyclic velocity mode.

As shown above, when the drive unit is continuously operates to torque to the continuous operation to torque direction, the torque will be output to the torque limit value when it is switched to the cyclic velocity mode.

In this case, execute any of the following before switching to the cyclic velocity mode.

- Start MC_MoveVelocity in the reverse direction setting to the continuous operation to torque direction and switch to the cyclic velocity mode. (At this time, it is recommended to specify "Acceleration/deceleration specification method" for acceleration/deceleration method.)
- Change the torque limit value to the lower value.

[When an error occurs]

For details, refer to the following.

☞ Page 27 Execute type/Enable type

■Relevant variables/FB

For details, refer to the following.

☞ Page 287 Relevant variables/FB

■Control details

For details, refer to the following.

☞ Page 287 Torque Control

■Precautions

The firmware for connected devices which can use the continuous operation to torque control mode is shown below.

Model name	Version
MR-J5(W)-G	Contact your local Mitsubishi representative.
MR-JET-G	

- "The cyclic torque mode ⇔ the continuous operation to torque control mode switching" cannot be executed when multiple start MC_TorqueControl is executed during MC_TorqueControl start-up. If it is executed, the warning "Control Mode Switching Disabled Warning" (warning code: 0D31H) occurs and operates in the current control mode.
- An actual motor velocity may not reach the limit velocity depending on the machine load during the continuous operation to torque control mode.
- It is recommended to match the direction of the torque command and the velocity command. If they differ, the velocity may decelerate once to 0.
- When switching from the continuous operation to torque control mode to the cyclic position mode, it is recommended to switch after changing the motor to stop status (zero speed status). The zero speed status can be monitored by labeling Status DO2(2D12h) to the device label.
- When switching from the continuous operation to torque control mode to the cyclic position mode without waiting for stop of the motor, set "ZSP disabled selection at control switching" (PC76.1) of the servo parameter to "1: Disabled" and disable monitoring of zero speed status. However, note that it may cause vibration or impact at control mode switching.
- For unavailable functions of the servo amplifier during the continuous operation to torque control mode, refer to the specification of connected driver.

Point

Set the system configuration with an unlimited operation range during the continuous operation to torque control mode as a stroke limit signal of the servo amplifier cannot be used during the continuous operation to torque control mode.
Use the software stroke limit function of Motion module side to restrict the set position.

Safety communication

A safety CPU and the firmware for MR-J5-G-RJ which can combine with RD78G(H) and perform the safety communication are shown below.

Model name	Version
RnSFCPU	☞ Restrictions by the version
MR-J5-G-RJ	Contact your local Mitsubishi representative.

■Precautions for the safety communication with MR-J5-G-RJ

When executing safety communication with MR-J5-G-RJ, the servo warning [AL.5E2.1_Safety communication no connection warning A (safety sub-function)] or [AL.5E2.9_Safety communication no connection warning B (safety sub-function)] occurs in the driver until the safety communication is established. Therefore the warning "Driver Warning" (warning code: 0D1FH) is detected in the motion system at connecting.
When referring the warning as execution conditions of the program, execute an error reset after connecting.

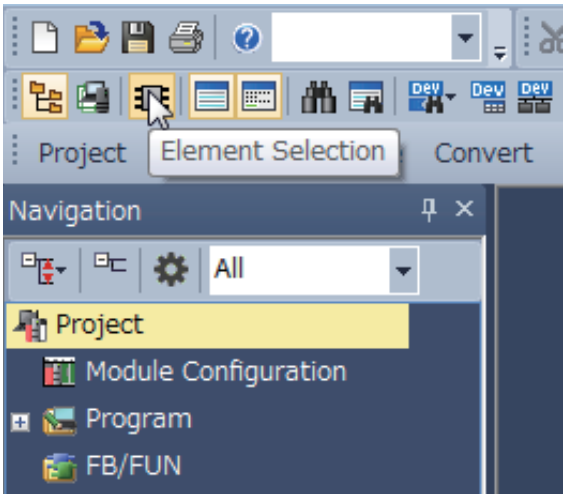
Precautions

When using MR-J5(W)-G with old firmware, the error "Watchdog Counter Error" (error code: 1C41H) may occur at connecting.
When this symptom occurs, update the firmware of MR-J5(W)-G.

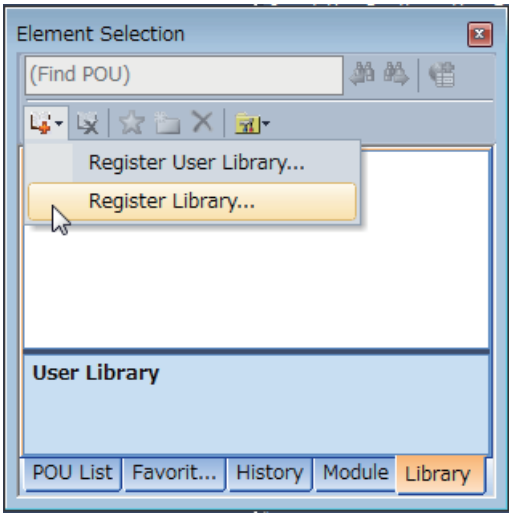
Appendix 2 Project Setting Example

Library Registration Procedure

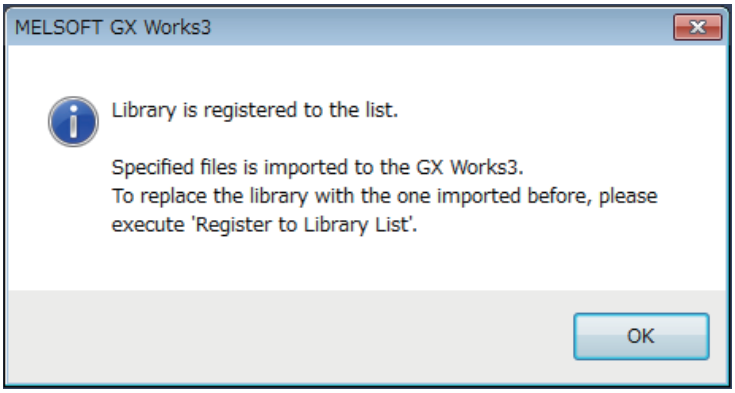
1. For MCFB library, please consult your local Mitsubishi representative.
2. Start GX Works3, select [Project] ⇨ [New], and select a type. Display the Element Selection window.



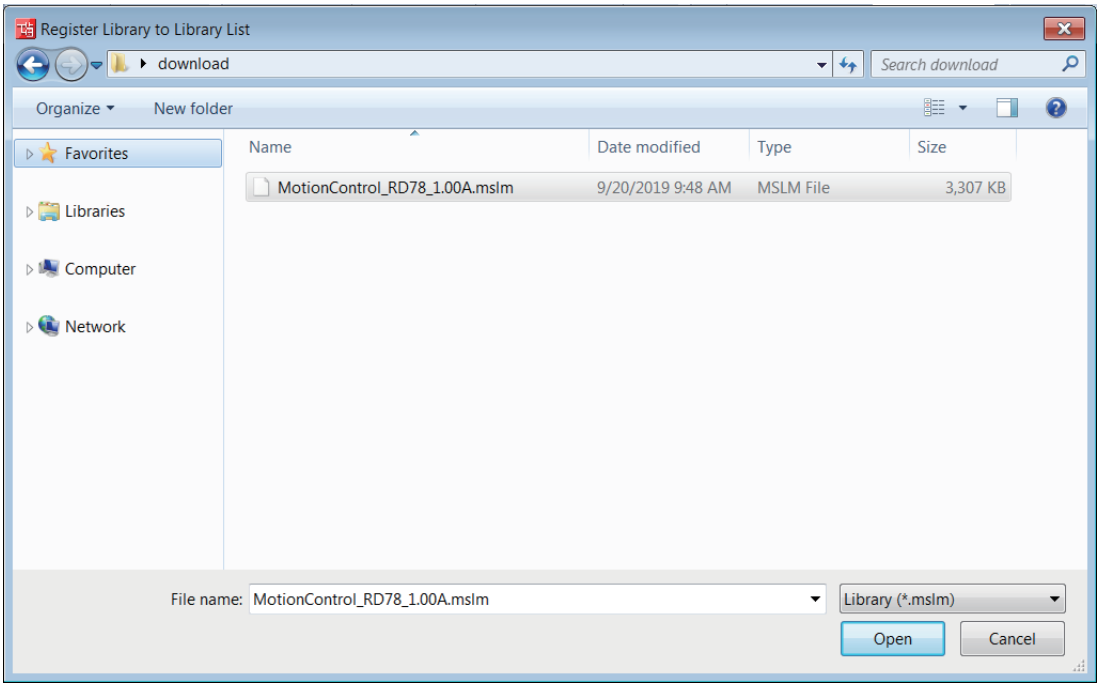
3. Select the Library tag, and select "Register Library..." of the "Register to Library List" icon.



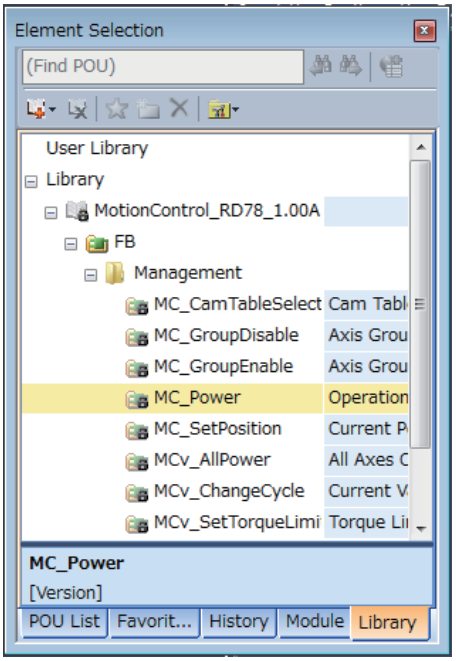
4. The following dialog box appears. Click the [OK] button.



5. The "Register Library to Library List" window appears. Select "MotionControl_****.mslm" and click the [Open] button.



6. Imported FBs are displayed in the Element Selection window.



7. Select an FB in the Element Selection window and drag and drop it into the work window. The FB is added in the Navigation window.

A

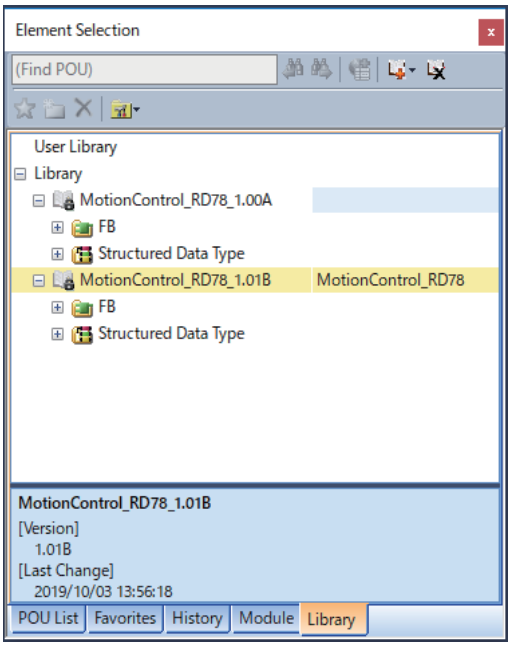
Library Update Procedure

To update the version of the FB library, register the new version of the library according to "Library Registration Procedure" and carry out "Update library element".

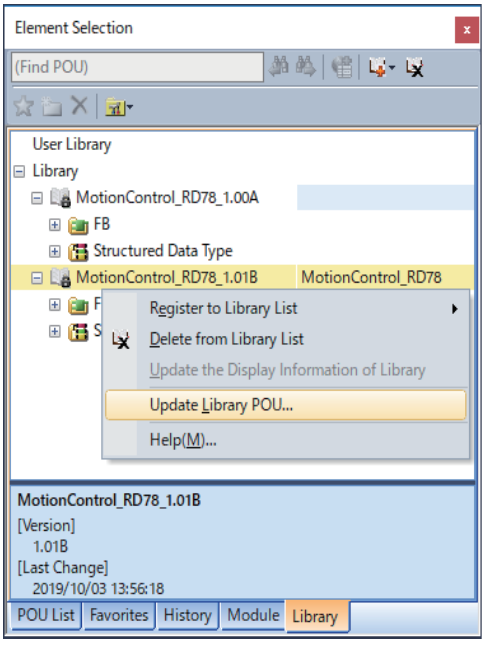
Ex.

Update from Ver.1.00A to Ver.1.01B.

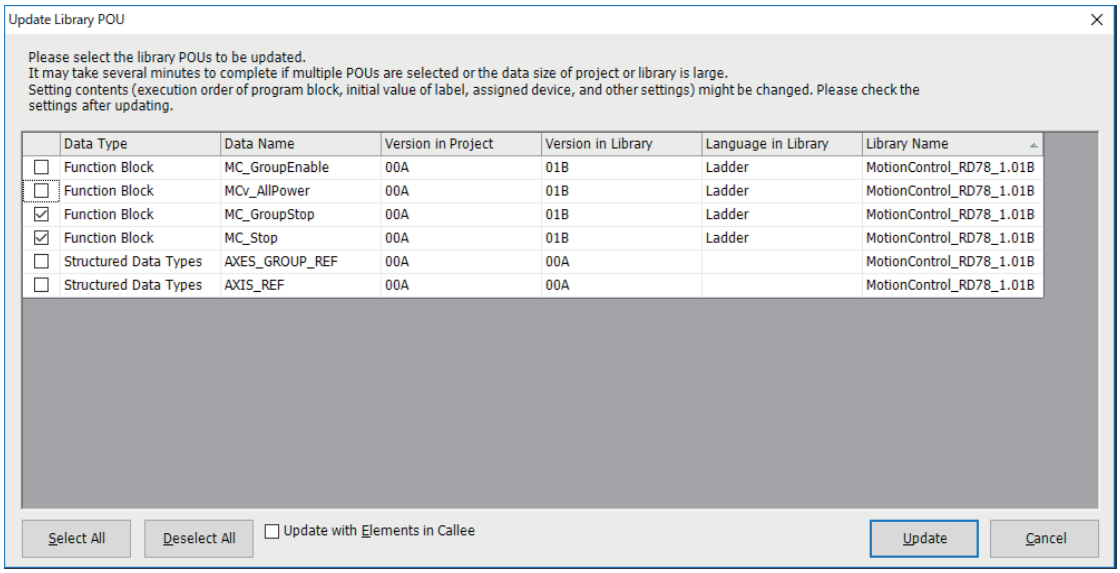
1. Register the new version of the library which is carried out updating according to "Library Registration Procedure". The current library version is shown in the following window.

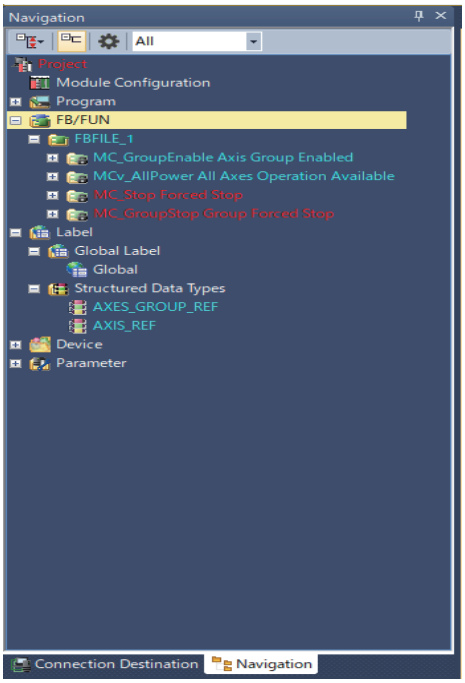


2. Right-click the library name and select "Update library element...".



3. The FB list used for the project is displayed in "Update library element" window.
As the FBs required updating of the version are automatically selected their checkboxes, click the [Update] button.





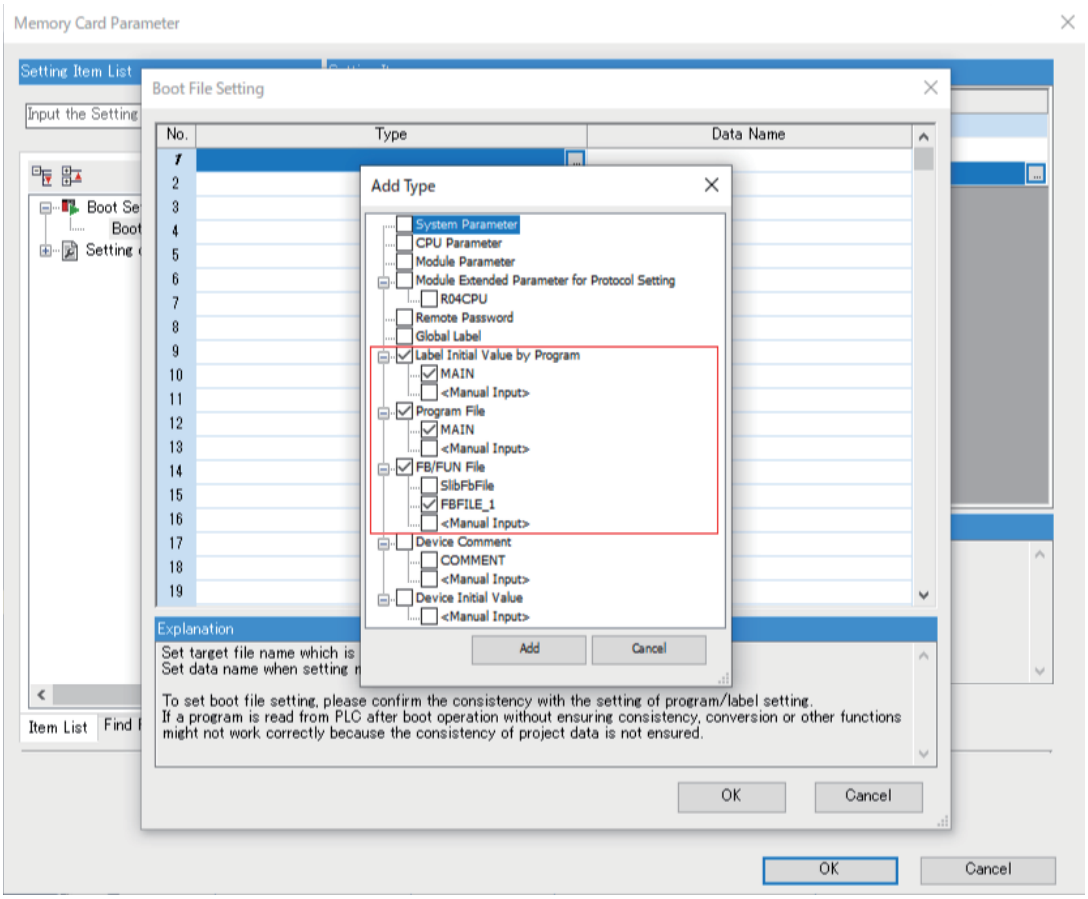
A

Precautions

[When setting program files using FB library in CPU module boot operation]

Each FB in "MotionControl_RD78_****.mslm" uses default value of labels classified by programs.

In CPU module boot operation, when specifying program files using this library FB by boot setting, also specify default value file of labels classified by programs by boot setting.



Appendix 3 Restrictions by the version

The CPU module and engineering tool versions corresponding to each Motion module are shown below.

Model		Version		
		GX Works3	CPU module	MCFB Library
RD78G	Ver. 04 or earlier	1.056J or later	R00CPU/R01CPU/R02CPU: 12 or later R04CPU/R08CPU/R16CPU/R32CPU/ R120CPU: 44 or later R04ENCPU/R08ENCPU/R16ENCPU/ R32ENCPU R120ENCPU: 44 or later	1.00A
	Ver. 05 or later	1.060N or later (The motion control setting function 1.005F or later)		1.00A or later
RD78GH	—	1.060N or later	R00CPU/R01CPU/R02CPU: 14 or later R04CPU/R08CPU/R16CPU/R32CPU/ R120CPU: 46 or later R04ENCPU/R08ENCPU/R16ENCPU/ R32ENCPU/ R120ENCPU: 46 or later	1.00A or later

The engineering tool versions corresponding to MCFB Library are shown below.

Version	
MCFB Library	GX Works3
1.00A	1.056J or later (motion control setting function 1.000A or later)
1.01B or later	1.060N or later (motion control setting function 1.005F or later)

The corresponding CPU module and engineering tool versions have restrictions depending on the functions of the Motion module. Refer to the following function and version combinations.

The add-on versions and software versions of the Motion module combinations are shown below.

☞ Page 723 List of Add-on Library

Point

Other than the above engineering tool does not support the Motion module.

—: There is no restriction by the version.

Item		Detail
Function		Supports communication cycle 31.25 μs, 62.5 μs and 8000.0 μs.
Add-on library		
	baseSystem	1.5
CPU module firmware		—
GX Works3		1.060N
GX Works3 Motion Control FB library		
	—	—
Motion control setting function		—

Item		Detail
Function		Supports network synchronous communication setting.
Add-on library		
	baseSystem	1.5
CPU module firmware		—
GX Works3		—
GX Works3 Motion Control FB library		
	—	—
Motion control setting function		—

Item		Detail
Function		Supports PDO mapping configuration.
Add-on library		
	baseSystem	1.5
	ServoDriver_CANopen	1.4
CPU module firmware		—
GX Works3		1.060N
GX Works3 Motion Control FB library		
	—	—
Motion control setting function		—

Item		Detail
Function		Supports safety communication function.
Add-on library		
	baseSystem	1.8
	NetworkDriver_CCIETSN	1.7
CPU module firmware		R08SFCPU/R16SFCPU/R32SFCPU/R120SFCPU: 20
GX Works3		1.065T
GX Works3 Motion Control FB library		
	—	—
Motion control setting function		—

Item		Detail
Function		Supports communication speed 100 Mbps.
Add-on library		
	baseSystem	1.8
CPU module firmware		—
GX Works3		1.065T
GX Works3 Motion Control FB library		
	—	—
Motion control setting function		—

Item		Detail
Function		Supports up to 120 network stations.
Add-on library		
	baseSystem	1.5
CPU module firmware		—
GX Works3		—
GX Works3 Motion Control FB library		
	—	—
Motion control setting function		—

Item		Detail
Function		Supports general purpose connection of CANopen devices.
Add-on library		
	NetworkDriver_CCIETSN	1.4
CPU module firmware		—
GX Works3		—
GX Works3 Motion Control FB library		
	—	—
Motion control setting function		—

Item		Detail
Function		Supports MR-JET-G.
Add-on library		
	baseSystem	1.5
	ServoDriver_CANopen	1.4
CPU module firmware		—
GX Works3		—
GX Works3 Motion Control FB library		
	—	—
Motion control setting function		—

Item		Detail
Function		Supports partial read and write of operation profile.
Add-on library		
	ProfileControl	1.4
	PackagingApp	1.4
CPU module firmware		—
GX Works3		—
GX Works3 Motion Control FB library		
	—	—
Motion control setting function		—

Item		Detail
Function		Adds file transfer commands (copy, move, file, compress, extract and attrib).
Add-on library		
	FileTransfer	1.4
CPU module firmware		—
GX Works3		—
GX Works3 Motion Control FB library		
	—	—
Motion control setting function		—

Item		Detail
Function		Supports Continuous operation to torque control mode.
Add-on library		
	MotionEngine	1.7
	ServoDriver_CANopen	1.7
CPU module firmware		—
GX Works3		—
GX Works3 Motion Control FB library		
	—	—
Motion control setting function		—

Item		Detail
Function		Adds fast axis operation mode.
Add-on library		
	Axis	1.4
	ServoDriver_CANopen	1.4
	NetworkDriver_CCIETSN	1.4
CPU module firmware		—
GX Works3		—
GX Works3 Motion Control FB library		
	—	—
Motion control setting function		1.005F

Item		Detail
Function		Adds encoder axis types.
Add-on library		
	Axis	1.4
	ServoDriver_CANopen	1.4
CPU module firmware		—
GX Works3		—
GX Works3 Motion Control FB library		
	—	—
Motion control setting function		1.005F

Item		Detail
Function		Adds override function, MC_SetOverride and MC_GroupSetOverride.
Add-on library		
	MotionEngine	1.4
	Axis	1.4
	MotionControl_General	1.4
CPU module firmware		—
GX Works3		—
GX Works3 Motion Control FB library		
	MC_SetOverride	00A
	MC_GroupSetOverride	00A
Motion control setting function		1.005F

Item		Detail
Function		Adds MC_ReadParameter and MC_WriteParameter.
Add-on library		
	Axis	1.4
	MotionControl_General	1.4
CPU module firmware		—
GX Works3		—
GX Works3 Motion Control FB library		
	MC_ReadParameter	00A
	MC_WriteParameter	00A
Motion control setting function		1.005F

Item		Detail
Function		Adds MC_Reset, MC_GroupReset and MCv_MotionErrorReset.
Add-on library		
	MotionControl_General	1.4
CPU module firmware		—
GX Works3		—
GX Works3 Motion Control FB library		
	MCv_MotionErrorReset	00A
	MC_GroupReset	00A
	MC_Reset	00A
Motion control setting function		1.005F

Item		Detail
Function		Adds MC_TouchProbe and MC_AbortTrigger.
Add-on library		
	MotionControl_General	1.4
CPU module firmware		—
GX Works3		—
GX Works3 Motion Control FB library		
	MC_TouchProbe	00A
	MC_AbortTrigger	00A
Motion control setting function		1.005F

Item		Detail
Function		Supports unit label.
Add-on library		
	baseSystem	1.5
	Axis	1.4
	Program_ST	1.4
CPU module firmware		—
GX Works3		1.060N
GX Works3 Motion Control FB library		
	—	—
Motion control setting function		1.005F

Item		Detail
Function		Supports TargetIgnored status of the drive unit.
Add-on library		
	Axis	1.4
	ServoDriver_CANopen	1.4
CPU module firmware		—
GX Works3		—
GX Works3 Motion Control FB library		
	—	—
Motion control setting function		1.005F

Item		Detail
Function		Adds file transfer access control.
Add-on library		
	FileTransfer	1.4
CPU module firmware		—
GX Works3		—
GX Works3 Motion Control FB library		
	—	—
Motion control setting function		1.005F

Item		Detail
Function		Adds MCv_BacklashCompensationFilter, MCv_DirectionFilter and MCv_SpeedLimitFilter.
Add-on library		
	MotionControl_AxisFilter	1.4
CPU module firmware		—
GX Works3		—
GX Works3 Motion Control FB library		
	MCv_BacklashCompensationFilter	00A
	MCv_DirectionFilter	00A
	MCv_SpeedLimitFilter	00A
Motion control setting function		1.005F

Item		Detail
Function		Adds purged pool monitor in MCv_SmoothingFilter.
Add-on library		
	MotionControl_AxisFilter	1.4
CPU module firmware		—
GX Works3		—
GX Works3 Motion Control FB library		
	MCv_SmoothingFilter	01B
Motion control setting function		1.005F

Item		Detail
Function		Supports RD78GH.
Add-on library		
	baseSystem	1.5
CPU module firmware		R00CPU/R01CPU/R02CPU: 14 R04CPU/R08CPU/R16CPU/R32CPU/R120CPU: 46 R04ENCPU/R08ENCPU/R16ENCPU/R32ENCPU/R120ENCPU: 46
GX Works3		1.060N
GX Works3 Motion Control FB library		
	—	—
Motion control setting function		1.005F

Item		Detail
Function		Adds memory usage monitor for BaseSystem.
Add-on library		
	—	—
CPU module firmware		—
GX Works3		—
GX Works3 Motion Control FB library		
	—	—
Motion control setting function		1.005F

Item		Detail
Function		Adds clearing command for motion event history.
Add-on library		
	MotionEventHist	1.4
CPU module firmware		—
GX Works3		—
GX Works3 Motion Control FB library		
	—	—
Motion control setting function		1.005F

Item		Detail
Function		Adds logging realtime monitor.
Add-on library		
	Logging	1.4
CPU module firmware		—
GX Works3		—
GX Works3 Motion Control FB library		
	—	—
Motion control setting function		1.005F

Item		Detail
Function		Supports initial stroke of operation profile.
Add-on library		
	ProfileControl	1.4
CPU module firmware		—
GX Works3		—
GX Works3 Motion Control FB library		
	—	—
Motion control setting function		1.005F
Motion software setting tool		—

Item		Detail
Function		Adds Servo System Recorder.
Add-on library		
	Axis	1.7
	Logging	1.7
CPU module firmware		—
GX Works3		—
GX Works3 Motion Control FB library		
	—	—
Motion control setting function		1.010L
Motion software setting tool		—

Item		Detail
Function		Supports Position data history.
Add-on library		
	Axis	1.7
	Logging	1.7
CPU module firmware		—
GX Works3		—
GX Works3 Motion Control FB library		
	—	—
Motion control setting function		1.010L
Motion software setting tool		—

INDEX

A

Absolute Position Control 98
Absolute Positioning Control 210
Acceleration/deceleration Processing Function . . 330
Addition/Subtraction Positioning 466
Add-on library 119
Axis 36
Axis Type 41

B

Backlash compensation filter 403
Boot Software 634
Buffer memory refresh 113

C

Cam Operation 443
CC-Link IE TSN 728
Circular interpolation control 263
Command format 615
Command In-position 315
Connectable Stations 55
CPU MODULE 589

D

Data logging function 531
DIRECT CONTROL 275
Driver unit conversion 381

E

Engineering tool 78
Event history function 565
External Signal Selection 418

F

Follow Up 95

G

Gear Operation 461

H

Hardware Stroke Limit 323
HOMING 198

J

jerk acceleration/deceleration 349
Jerk limit value 347
JOG operation 227
JSON File Data Specifications 550

L

LED 637
Library 748
Linear interpolation control 254
Log file 619
LOGGING 530

M

MCFB 592
Module extended parameters 602
MOTION SERVICE PROCESSING 587
Motion synchronization station link device auto refresh 113
Motion System Software 625
MR-J5(W)-G 731
Multiple Axes Positioning Control 232
Multiple Start 132

N

Near pass 153

O

Operation Cycle 109
Operation Profile Data 473
Override Function 359

P

PDO mapping 59
Positioning data history 568
Positioning Range 83
Processing time over check 561

R

RAS 557
Real drive axis 44
Real encoder axis 45
Real-time monitor 532
Relative Positioning Control 218

S

Servo ON/OFF 88
Set position 84
SIGNAL_SELECT structure 418
Single Axis Manual Control 224
Single Axis Positioning Control 206
Single Axis Speed Control 220
Single Axis Synchronization Control 438
Slave Emulate 435
Slave object 58
Slave object map 371
Smoothing filter 393
Software Reboot 124
Software Stroke Limit 317
State transition 39
Storage files 601
System basic cycle 109

T

TARGET_REF structure 420
The response code (SDO Abort code) 612
Torque Control 287
Torque limit 371
Touch Probe 423
Trigger condition (TRIGGERCONDITION) 541

V

Virtual drive axis 50
Virtual encoder axis 50
Virtual linked axis 54

I

REVISIONS

*The manual number is given on the bottom left of the back cover.

Revision date	*Manual number	Description
July 2019	IB(NA)-0300411ENG-A	First edition
January 2020	IB(NA)-0300411ENG-B	■Added models RD78GHV, RD78GHW ■Added or modified parts TERMS, GENERIC TERMS AND ABBREVIATIONS, HOW TO READ THIS MANUAL, FUTURE SUPPORT PLANNED,FUTURE SUPPORT PLANNED, Chapter 23, 24, 25, 26, Section 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 3.1, 3.2, 3.3, 4.3, 4.4, 4.5, 5.1, 6.1, 6.2, 6.3, 6.4, 7.1, 7.2, 8.1, 8.4, 9.1, 9.2, 9.3, 10.2, 11.1, 11.2, 12.1, 12.2, 12.3, 13.1, 13.2, 14.2, 14.3, 14.4, 14.5, 14.6, 14.7, 14.8, 14.9, 15.1, 15.2, 16.1, 16.2, 18.1, 18.2, 19.1, 19.2, 19.3, 21.1, 21.3, 22.3, 22.4, 22.5, 22.6, 25, Appendix 1, 2, 3, 4, 5,INDEX, WARRANTY, TRADEMARKS
August 2020	IB(NA)-0300411ENG-C	■Added or modified parts SAFETY PRECAUTIONS, RELEVANT MANUALS, TERMS, GENERIC TERMS AND ABBREVIATIONS, HOW TO READ THIS MANUAL, FUTURE SUPPORT PLANNED, Chapter 7, 23, 24, 25, 26, 27, 28 Section 1.1, 1.2, 1.4, 1.5, 1.6, 2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 3.1, 3.2, 3.3, 3.4, 4.1, 4.2, 4.3, 4.4, 4.5, 5.1, 5.2, 6.1, 6.2, 6.3, 6.4, 7.1, 7.2, 8.1, 8.2, 8.3, 8.4, 9.1, 9.2, 9.3, 10.2, 11.1, 11.2, 11.3, 12.1, 12.2, 12.3, 13.1, 13.2, 13.3, 13.4, 13.5, 13.6, 13.7, 13.8, 13.9, 14.1, 14.2, 15.1, 15.2, 15.3, 16.1, 16.2, 16.3, 17.2, 18.1, 18.2, 19.1, 19.2, 19.3, 19.4, 21.1, 21.2, 21.3, 22.1, 22.2, 22.3, 22.4, 22.5, 22.7, Appendix 1, 2, 3

Japanese manual number: IB-0300410-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.

© 2019 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.

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®.
VxWorks is registered trademarks or trademarks of Wind River Systems, Inc.
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)-0300411ENG-C(2008)MEE
MODEL: RD78-U-APP-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.